MODELING GROUNDWATER FLOW AND CONTAMINANT TRANSPORT IN HETEROGENEOUS AQUIFERS: FINAL PROJECT REPORT

Myron B. Allen

1992

Final Report WWRC-92-09

Submitted to

Wyoming Water Resources Center
University of Wyoming

Submitted by

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

Contents of this publication have been reviewed only for editorial and grammatical correctness, not for technical accuracy. The material presented herein resulted from research sponsored by the Applied Mathematical Sciences Program of the U.S. Department of Energy Office of Energy Research and the Wyoming Water Resources Center, however views presented reflect neither a consensus of opinion nor the views and policies of the Wyoming Water Development Commission, Wyoming Water Resources Center, or the University of Wyoming. Explicit findings and implicit interpretations of this document are the sole responsibility of the author(s).

1. TECHNICAL SYNOPSIS

This section of the report is a brief synopsis of the project's scientific aims and accomplishments. The discussion is intended for a technical audience, but it does not assume that readers are specialists in mathematical modeling. The Appendix to this report, summarized in Section 2, consists of published scientific articles that describe the results of the project for specialists.

1.1 Objectives

The main aim of the research was to develop methodologies for modeling simultaneous groundwater flow and contaminant transport in highly heterogeneous aquifers. Most previously existing flow and transport models are based on techniques that, while adequate for nearly homogeneous aquifers, are inappropriate in the presence of significant, fine-scale heterogeneities. The inappropriateness stems from two facts. First, the numerical methods used are inefficient or inaccurate in heterogeneous problems, so that modelers typically sacrifice accuracy in favor of affordability when running the codes. Second, the relationships between actual, fine-scale variations in the media and the parameter values that one should use to represent the media in affordable, coarse-scale models remain poorly understood.

The proposed project had two objectives. The immediate objective was to incorporate recent improvements in numerics to assemble a computer code that is computationally efficient even when the aquifer being modeled is highly heterogeneous. The long-range objective was to use the code to investigate methods for scaling from individual realizations of heterogeneity to ensembles of realizations that are consistent with measured data and, more specifically, to investigate scaling of such parameters as hydraulic conductivity and hydrodynamic dispersion for use in standard flow and transport codes.

1.2 Utility of the research.

The research has utility to groundwater hydrologists who use models to understand the complex flow and transport phenomena affecting groundwater contamination. Natural aquifers can have permeabilities and porosities that exhibit large spatial variations as a consequence of variable depositional environments, diagenesis, and structural events. Among the problems that heterogeneities pose are the following:

- It is extremely difficult to measure heterogeneous aquifer parameters, so the data used in mathematical models often fail to characterize aquifers realistically.

- Even when heterogeneities are known, resolving them numerically often requires the modeler to discretize the aquifer into a large number of very small grid cells. This not only makes the equation sets to be solved large, hence expensive; it also makes them poorly conditioned, hence prone to slow convergence and accumulation of roundoff errors.

- Beyond the effect of small grid cells, heterogeneity itself leads to poor conditioning arising from the fact that important coefficients, such as hydraulic conductivity, can range in value over several orders of magnitude.

- Heterogeneity often occurs at scales that simply cannot be resolved in affordable models. In these cases, there arises the issue of how to scale aquifer parameters to arrive at values that adequately represent the physics in "megascopic" models. Research into this question is in its infancy, since the computational horsepower needed to explore relationships between fine- and coarse-scale models has evolved only recently.

As a consequence of these facts, heterogeneity has been a source of tremendous difficulties in the transfer of modeling technologies from theoretical settings to field applications, where there is an increasing need for reliable predictive tools. The development of robust and efficient numerical techniques is a necessary step, not only for the direct application of models to field studies but also to the more fundamental task of understanding how to incorporate uncertain and sparsely measured geologic data into deterministic computer codes.

The research also has implications for other areas of technology involving underground flow. For example, advances in numerical techniques for flows in heterogeneous porous media simultaneously improve the state of the art in the design of enhanced oil recovery technologies and the simulation of in-situ mining.

1.3 Summary of accomplishments

The project's accomplishments fall into four categories. First, Some effort focused on enhancing the capabilities of a two-dimensional groundwater transport simulator developed in previous work for the Wyoming Water Research Center. This work led to the effective incorporation of a timestepping algorithm based on contaminant paths ("modified method of characteristics") into an existing transport code based on alternating-direction collocation (Allen and Khosravani, 1992) and an adaptive local grid refinement algorithm for this code that allows for fine-scale spatial resolution in regions where contaminant concentrations vary rapidly in space (Curran, submitted).

A second focus for the research was the implementation of an efficient, well-conditioned algorithm for solving the groundwater flow equation using the mixed finite-element method. Proper formulation of the mixed method allows one to solve for groundwater velocities whose accuracies are comparable to those of the computed heads. The method differs from standard finite-element approaches, in that it does not require one to diffferentiate heads numerically to compute velocities — a common procedure that introduces inherent inaccuracies. The new algorithm avoids the poor conditioning (and associated inefficiency) associated with fine-scale, heterogeneous simulations by using an iterative solver based on a multigrid approach (Allen, Ewing, and Lu, 1992). The algorithm has the additional feature that it is readily amenable to parallel processing (Allen and Curran, 1992).

Summaries and overviews of these methodologies for groundwater flow and transport appear in Allen and Ewing (1991) and Allen and Curran (to appear).

A third approach, tailored to a more specialized form of heterogeneity, incorporates a finite-layer technique into models of groundwater flow in highly stratified aquifers. This approach takes advantage of certain simplifying assumptions about the geometry of the heterogeneity to develop a discrete formulation that is suitable for large-scale computing (because of its inherent parallelism) and for microcomputing (because of its ability to decompose large problems into small subproblems). The project addressed both the practical implementation of the method (Smith, Allen, Puckett, and Edgar, 1991; Smith, Allen, Puckett, and Edgar, 1992) and its theoretical basis (Smith, 1992; Smith and Allen, in preparation).

Finally, some effort was devoted to an analysis of standard finite-element techniques for modeling contaminant transport in aquifers characterized by highly heterogeneous adsorption. This analysis was mainly theoretical, although the work generated a computer code that proved useful in testing error estimates derived using abstract methods (Chunyu, 1990).

2. PUBLICATIONS RESULTING FROM THE WORK

The following is a list of refereed journal articles, conference proceedings, doctoral dissertations, and MS papers that resulted from the project. Copies of these documents, except for the dissertation and MS paper, appear in the Appendix. The dissertation is available at the University of Wyoming Science Library and will soon be available from University Microfilms in Ann Arbor, Michigan. The MS paper is available from M.B. Allen, Department of Mathematics, University of Wyoming, (307) 766-4221.

2.1 Refereed journal articles

Allen, M.B., Ewing, R.E., and Lu, P., "Well-conditioned iterative schemes for mixed finite-element models of porous-media flows," *SIAM Journal of Scientific and Statistical Computing 13:3* (1992), 794-814.

Curran, M.C., "An iterative finite-element collocation method for parabolic problems using domain decomposition," submitted to *Numerical Methods for Partial Differential Equations.*

Smith, S.S., Allen, M.B., Puckett, J., and Edgar, T., "The finite layer method for groundwater flow models," *Water Resources Research 28:6* (1992), 1715-1722.

Smith, S.S., and Allen, M.B., "Error analysis of the finite-strip method for parabolic equations," MS in preparation, draft included in Appendix.

2.2 Conference proceedings

Allen, M.B., and Curran, M.C., "A multigrid-based solver for mixed finite-element approximations to groundwater flow," *Computational Methods in Water Resources IX, Vol. I: Numerical Methods in Water Resources,* ed. by T.F. Russell et al., Elsevier Applied Science, London, 1992, 579-585.

Allen, M.B., and Curran, M.C., "Parallelizable methods for modeling flow and transport in heterogeneous porous media," to appear in *Proceedings, Oberwolfach Conference on Porous Media*, June 21-27, 1992, Oberwolfach, Germany, ed. by U. Hornung et al., Birkhauser, Munich.

Allen, M.B., and Ewing, R.E., "Mathematical challenges in groundwater contaminant modeling," *Proceedings, Fourth Annual Meeting of the Wyoming State Section, American Water Resources Association*, Laramie, Wyoming, November 6-7, 1991.

Smith, S.S., Allen, M.B., Puckett, J.A., and Edgar, T.V., "Three-dimensional model of multi-well field using finite-layer models," *Proceedings, Eleventh Annual American Geophysical Union Hydrology Days,* Fort Collins, Colorado, April 2-4, 1991, Colorado State University, Fort Collins, Colorado, 23-34.

2.3 Graduate papers and dissertations

Chunyu, D., "Finite-element methods for contaminant transport with adsorption," M.S. paper, Department of Mathematics, University of Wyoming, Laramie, Wyoming, December, 1990.

Smith, S.S., "Finite-Strip and Finite-Layer Methods: Analysis and Applications to Groundwater Modeling," Ph.D. dissertation, Department of Mathematics, University of Wyoming, Laramie, Wyoming, May, 1992.

3. GRADUATE STUDENT TRAINING

Three graduate students received support from this project. Two of them completed degrees during the course of the project:

- Dongmei Chunyu, M.S., Mathematics, 1990.
- Stanley S. Smith, Ph.D., Mathematics, 1992.

The third student, Azar Khosravani, received an M.S. before the project began, spent a summer working on the project, then transferred to Southern Illinois University, where her husband has a faculty position.

APPENDIX: COPIES OF PUBLICATIONS

Attached are copies of papers that resulted from the project. Also attached is a copy of a Wyoming Water Research Center Research Brief, which summarizes in nontechnical form some aspects of the work.

A Multigrid-Based Solver for Mixed Finite-Element Approximations to Groundwater Flow

Myron B. Allen[1]
Department of Mathematics, University of Wyoming, Laramie, WY 82071, U.S.A.
Mark C. Curran[2]
Applied and Numerical Mathematics Division, Sandia National Laboratories, Albuquerque, NM 87185, U.S.A.

ABSTRACT

Mixed finite-element methods have several features that are attractive in the numerical simulation of groundwater flow. Chief among these is the possibility of computing Darcy velocities whose accuracies are comparable to those of the computed hydraulic heads. Much current research centers on solving the large linear systems that arise from mixed finite-element discretizations. We examine an iterative method that largely overcomes the poor conditioning associated with fine spatial grids and highly variable aquifer properties. The method incorporates a multigrid scheme inside an outer iteration whose convergence rate is independent of grid mesh size and variations in hydraulic conductivity. As we demonstrate, the multigrid algorithm is amenable to effective parallelization on distributed-memory machines, making the overall algorithm a highly efficient one in such computing environments.

[1]The Wyoming Water Research Center partially supported this work through a grant-in-aid

[2]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.

1. INTRODUCTION

The equations governing the steady flow of a single fluid in a two-dimensional porous medium Ω with no gravity drive have the following forms:

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

Here $\mathbf{u} = (u^x, u^y), p$, and f represent the Darcy velocity, hydraulic head, and source term, respectively. In many natural groundwater aquifers, the hydraulic conductivity $K(x, y)$ exhibits irregular variations depending upon the lithology of the host rock. This heterogeneous structure causes many difficulties for numerical modelers, among which are two sources of poor conditioning in linear systems that approximate the differential equations. One source of poor conditioning is the need to use fine spatial grids to resolve the complexities of the medium and the resulting variations in p and $\mathbf{u}$. Another source is the variability in K itself, which affects the coefficients in the matrices of the linear system. These difficulties afflict essentially all discrete approximations to Equations (1).

Among the enormous variety of such methods, mixed finite-element methods have attracted a great deal of attention over the past decade. 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 $h \to 0$ (Douglas et al.[1], Raviart and Thomas [2]). This property stands in contrast to many standard Galerkin and finite-difference formulations, where one first solves for p and then numerically differentiates to compute a less accurate approximation to $\mathbf{u}$. Thus mixed methods are particularly well suited to problems where accurate velocities are critical to the prediction of underground contaminant movements.

This paper examines an iterative scheme for solving the lowest-order mixed finite-element approximations to Equations (1) on rectangular grids. The overall structure of the scheme, analyzed in detail by Allen et al.[3], 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. Rapid execution of this inner iteration is crucial to the efficiency of the scheme. We use a highly parallelizable multigrid method to perform the inner iterations.

Section 2 reviews the mixed finite-element method. Section 3 discusses the iterative scheme, reviews its theoretical properties, and describes the multigrid method used in the inner iteration. Section 4 presents numerical results that indicate the efficiency of the scheme. In Section 5 we briefly draw some conclusions.

2. THE MIXED FINITE-ELEMENT METHOD

Consider Equations (1), subject to the boundary condition $p = 0$ on $\partial\Omega$. To discretize this system via the lowest-order mixed finite-element method, we establish a rectangular grid Δ on Ω having vertical grid lines at $x_0, x_1, \ldots, x_m$ and horizontal grid lines at $y_0, y_1, \ldots, y_n$, as drawn in Figure 1. The mesh size h of Δ is the maximum distance between adjacent grid lines $x = x_i$ or $y = y_j$. With Δ we associate trial spaces[2] Q_x, Q_y, and V for the x-velocity u^x, the y-velocity u^y, and the hydraulic head p, respectively. Functions in Q_x are piecewise linear in x and piecewise constant in y; functions in Q_y are piecewise constant in x and piecewise linear in y, and functions in V are piecewise constant on Δ.

Each of these trial spaces has a finite nodal basis consisting of tensor products of the usual one-dimensional bases for piecewise constant and piecewise linear interpolation. As Figure 1 illustrates, 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 $u^x_{i,j}$ of x-velocity with the midpoint $(x_i, y_{j-1/2})$ of each vertical cell edge, and a nodal value $u^y_{i,j}$ with the midpoint $(x_{i-1/2}, y_j)$ of each horizontal cell edge.

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

$$\begin{aligned} &\int_\Omega \frac{\mathbf{u}_h \cdot \mathbf{v}}{K}\, dx\, dy - \int_\Omega p_h \operatorname{div} \mathbf{v}\, dx\, dy = 0, \quad \forall\, \mathbf{v} \in Q_x \times Q_y, \\ &\int_\Omega (\operatorname{div} \mathbf{u}_h - f) q\, dx\, dy = 0, \qquad\qquad \forall\, q \in V. \end{aligned} \tag{2}$$

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

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

$$\begin{bmatrix} A & N \\ N^T & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix} = \begin{bmatrix} 0 \\ F \end{bmatrix}. \tag{3}$$

Here, U stands for a vector containing the nodal values of the velocities u^x and u^y, and P is a vector containing nodal heads. The block 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 $A^x \in \mathbb{R}^{(m+1)n \times (m+1)n}$ and $A^y \in \mathbb{R}^{m(n+1) \times m(n+1)}$ are tridiagonal, and their entries are integrals involving the variable hydraulic conductivity K. The matrix N has the block structure

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

where $N^x \in \mathbb{R}^{(m+1)n \times mn}$ and $N^y \in \mathbb{R}^{m(n+1) \times mn}$. These blocks reduce to the usual difference approximations to $\partial/\partial x$ and $\partial/\partial y$. The vector $F \in \mathbb{R}^{mn}$ contains integrals involving the source function f. For details concerning the construction of this linear system, we refer readers to Allen et al.[3]

3. AN ITERATIVE SCHEME

We solve the system (3) iteratively, using the following matrix splitting:

$$\begin{bmatrix} D & N \\ N^\top & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix}^{(k+1)} = \begin{bmatrix} 0 \\ F \end{bmatrix} + \begin{bmatrix} D-A & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} U \\ P \end{bmatrix}^{(k)}. \qquad (4)$$

Here, D is a diagonal matrix that can have any of several structures, the simplest of which is just the diagonal part of A. This scheme has convergence rate that is independent of the mesh size h and the variations in hydraulic conductivity K; in fact, each iteration reduces the error by a factor no greater than $\frac{1}{2}$ (see Allen et al.[3]).

Computationally, the scheme (4) requires the following steps:

(*i*) $G^{(k-1)} \leftarrow -F + N^\top D^{-1}(D-A)U^{(k-1)}$.

(*ii*) Solve $N^\top D^{-1} N P^{(k)} = G^{(k-1)}$.

(*iii*) $U^{(k)} \leftarrow D^{-1}(D-A)U^{(k-1)} - D^{-1}NP^{(k)}$.

Steps (*i*) and (*iii*) in this algorithm require only matrix multiplication and addition and hence are quite cheap. Step (*ii*), however, requires more work, since the matrix $N^\top D^{-1} N$ has the same pentadiagonal structure as the usual five-point finite-difference approximation to operators of the form div (K grad).

To execute this step efficiently, we use a multigrid scheme. Instead of solving step (*ii*) exactly, we perform several V-cycles to get an approximate solution for $P^{(k)}$, then move on to step (*iii*). Each V-cycle involves two Gauss-Seidel iterations at each level in a nest $\Delta = \Delta_0 \supset \Delta_1 \supset \cdots \supset \Delta_L$ of grids, ranging from the original grid Δ through coarser subgrids to the coarsest grid Δ_L, then back up to the finest grid Δ. To map the problem from fine grids to coarse grids, we use full weighting as a restriction operator. To map from coarse grids to fine grids, we use bilinear interpolation as a prolongation operator.

One attractive feature of the multigrid scheme is its amenability to parallel processing. Tuminaro and Womble[4] discuss this prospect. By adopting a red-black ordering for the cells in each grid, we can decompose each Gauss-Seidel relaxation sweep into two sets of calculations. In particular, we label 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 can update each of the "red" cells using old values in the "black" cells, then use these updated values in "red" cells

to update each of the "black" cells. Since the calculations for "red" cells in any sweep are independent of each other, we can perform the arithmetic concurrently on a parallel computer. Similarly, the updates for "black" cells are also mutually independent and can be computed concurrently.

This idea works especially well on distributed-memory machines, where it is feasible to have a large number of processors that communicate through message passing instead of accessing a shared memory. In coding the algorithm, we decompose the spatial domain of the problem so that each processor in a parallel machine performs the calculations for a subset of the grid.[4] At any instant during the calculations, a given processor is performing either "red" or "black" updates. The "red" and "black" processors work simultaneously, stopping synchronously to exchange results and "change colors."

4. COMPUTATIONAL PERFORMANCE

Since Allen et al.[3] discuss the performance of the serial precursor to our scheme in the presence of a variety of heterogeneous conductivity fields $K(x, y)$, we focus here on the performance of the parallel version. To assess this performance, we examine the execution time required by our code on a 1024-processor nCUBE 2 having a distributed-memory hypercube archtecture. To gain some appreciation for the degree of parallelism in the code, we investigate the execution time required on subcubes of the machine having dimension 0 (1 processor), 1 (2 processors), ..., 10 (1024 processors).

We base our assessment on the notion of *scaled speedup*. To gauge the scaled speedup, we run problems of proportionately larger size on larger subcubes. Specifically, we run a problem in which we assign the work associated with a 32×32-cell grid with each processor of a subcube of the machine. Thus we use a 32×32-cell grid on the one-processor subcube, one involving a 64×32-cell grid on the two-processor subcube, one involving a 64×64-cell grid on the four-processor subcube, and so forth, eventually running a problem involving a 1024×1024 grid on the 1024-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, such as setup time, interfere with this ideal relationship.

Table 1 lists the timings for the runs. The table shows the total execution times, along with the times associated with problem setup (e.g. initialization and matrix assembly) and interprocessor communication, for various subcubes of the machine. The subcubes range in size from dimension zero (one processor operating on a 1024-cell grid) to dimension 10 (1024 processors acting on a 1,048,576-cell grid). Each run represents 20 outer iterations of the scheme (4), each iteration of which requires five V-cycles in step (ii). The times listed in the second through fourth columns are averages over all

processors, while the times listed in the last column are the maximum times over all processors and thus more closely reflect the apparent execution time observed by a user. These timings suggest that the algorithm possesses excellent parallelism in addition to its good performance in the presence of heterogeneities and fine grids.

Table 1: RUNTIMES (SECONDS) FOR SCALED PROBLEMS ON THE nCUBE 2.

Number of processors	Setup time	Communication time	Average total time	Maximum total time
1	6.515	0.205	41.119	41.119
2	6.298	1.783	43.131	43.133
4	6.109	3.694	45.389	45.454
8	6.114	4.640	46.607	46.647
16	6.150	5.774	48.088	48.160
32	6.223	5.832	48.357	48.526
64	6.371	5.907	48.681	48.998
128	6.671	5.949	49.096	49.721
256	7.290	5.979	49.789	51.035
512	8.565	6.028	51.149	53.693
1024	11.141	6.077	53.795	58.981

5. CONCLUSIONS

Our algorithm appears to promise excellent opportunities for parallel computing as well as a reasonable way to overcome some of the numerical difficulties associated with heterogeneities. Given this promise, we see our next task as the extension of the method to time-dependent and nonlinear problems, which have more general applicability to underground contaminant modeling.

ACKNOWLEDGMENT

The authors thank Dick Ewing, whose insights guided much of our work.

REFERENCES

1. Douglas, J., Ewing, R.E., and Wheeler, M.F., "The Approximation of the Pressure by a Mixed Method in the Simulation of Miscible Displacement," *R.A.I.R.O. Analyse Numerique 17*, pp. 17-33, 1983.

2. Raviart, P.A., and Thomas, J.M., "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, 1977.

3. Allen, M.B., Ewing, R.E., and Lu, P., "Well Conditioned Iterative Schemes for Mixed Finite-Element Models of Porous-Media Flow," to appear in *SIAM Jour. Sci. Stat. Comp.*

4. Tuminaro, R.S., and Womble, D.E., "Analysis of the Multigrid FMV Cycle on Large-Scale Parallel Machines," to appear in *SIAM Jour. Sci. Stat. Comp.*

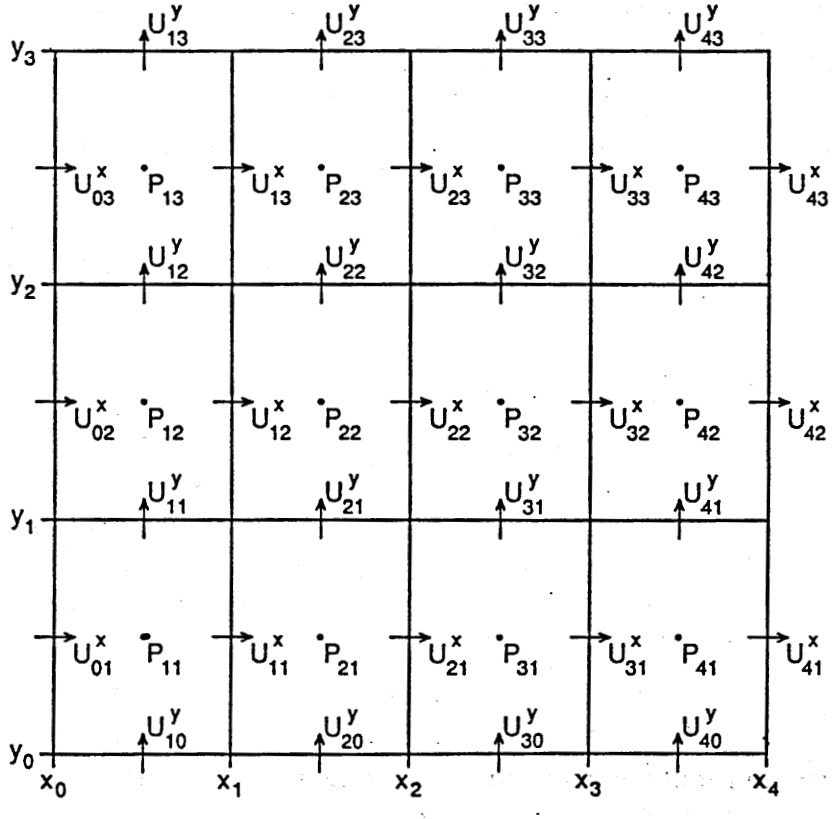

Figure 1. Sample grid for the mixed finite-element method, showing nodes for the hydraulic head and the x- and y-velocities.

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 solwly. 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 "embarassingly 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 $h \to 0$ (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

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

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 $x = x_0, x_1, \ldots, x_m$ and horizontal grid lines at $y = y_0, y_1, \ldots, y_N$. 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 $u^x_{i,j}$ of x-velocity with the midpoint $(x_i, y_{j-1/2})$ of each vertical cell edge, and a nodal value $u^y_{i,j}$ 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

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

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:

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

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:

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

Here, D is a diagonal matrix, the simplest effective structure for which is the 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) $G^{(k-1)} \leftarrow -F + N^\top D^{-1}(D-A)U^{(k-1)}$.

(ii) Solve $N^\top D^{-1} N P^{(k)} = G^{(k-1)}$.

(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 $N^\top D^{-1} N$ 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 and "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 $\Delta = \Delta_0 \supset \Delta_1 \supset \cdots \supset \Delta_L$ 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 $\Omega = (0,1) \times (0,1)$:

$$
\begin{aligned}
K_{\mathrm{I}}(x,y) &= 1; \\
K_{\mathrm{II}}(x,y) &= e^{-x-y}; \\
K_{\mathrm{III}}(x,y) &= \begin{cases} 1, & \text{if } x < y, \\ 0.1, & \text{if } x \geq y; \end{cases} \\
K_{\mathrm{IV}}(x,y) &= K_{\mathrm{II}}(x,y) \cdot K_{\mathrm{III}}(x,y); \\
K_{\mathrm{V}}(x,y) &= \begin{cases} 1, & \text{if } x < y, \\ 0.01, & \text{if } x \geq y. \end{cases}
\end{aligned}
$$

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.

	Grid mesh h				
Coefficient	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 archtecture. 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 communciation 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

$$D_t c - \nabla \cdot (D_H \nabla c) = 0 . \tag{6}$$

Consider the following initial-boundary-value problem:

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

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:

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

These equations determine the evolution of the unknown coefficents 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:

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

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

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

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, Chigago, 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).

Proceedings, Fourth Annual Meeting of the Wyoming State Section, American Water Resources Section, Laramie, Wyoming, November 6-7, 1991.

Mathematical Challenges in Groundwater Contaminant Modeling

by Myron B. Allen and Richard E. Ewing
Institute for Scientific Computation
P.O. Box 3036 University of Wyoming
Laramie, WY 82071

The use of computer models to simulate groundwater flow and contaminant transport has burgeoned in the past few years. There are good reasons for this phenomenon: Natural aquifers tend to have complicated geometries and highly variable rock properties, and there is a pressing societal need for quantitative predictions of contaminant movements in these complex geologic settings. Computer models offer the only realistic hope for meeting this need.

Despite the apparent power of computer models, many technical problems conspire to reduce their accuracy in field studies. Obvious to most water resources professionals are difficulties associated with aquifer characterization and the "garbage in, garbage out" syndrome. More subtle, however, are several mathematical issues that require adequate resolution before we can expect realistic aquifer simulations. This abstract is a brief summary of our research into these issues.

Three concepts are common to all numerical models of underground flows. First, one must make some assumptions about the physics and chemistry of the flows. These assumptions give rise to complicated and often nonlinear sets of partial differential equations that govern fluid velocities, movements and fates of contaminant plumes, and other variables of interest. Second, to solve the governing equations, one must approximate them, usually by converting the differential equations to discrete algebraic analogs. Among the most common "discretization" methods are finite-difference and finite-element techniques. These methods partition the aquifer into grid cells or nodes, associating with each cell or node algebraic equations analogous to the mass or momentum balance for that zone. The results are systems of algebraic equations, characterized by matrices that can have tens of thousands or even millions of entries. Third, given such large matrix analogs of the original flow and transport equations, one must devise efficient ways to solve them on digital computers.

Some of our work focuses on the first phase of the modeling enterprise, the derivation of governing equations. Although the physics of flows in porous media are well established at small scales, they are poorly understood at scales where the natural heterogeneities of the rock matrix are prominent. Such heterogeneities arise from variations in depositional environment, diagenetic changes in the pore geometry of the rock, and structural events that cause fracturing and faulting. To the modeler, these heterogeneities pose a severe challenge: How can we scale our knowledge, gained from measurements on cores, well tests, and wireline data, to the scale of typical grid cells?

As an example of the utility of numerical models in answering scaling problems, consider the small-scale fingering and channeling of water-soluble contaminants through an aquifer that has high-conductivity streaks distributed irregularly in space. Capturing the precise geometry of such plumes in a model is typically infeasible: It simply requires too much fine-scale knowledge of an aquifer's properties, and this knowledge is expensive even in bench-scale studies. However, one can use numerical models to investigate connections between well understood, small-scale physics and the large-scale movement of plumes in the presence of heterogeneities. We have explored techniques for modeling the average behavior of such plumes by incorporating "effective hydrodynamic dispersivities" in the governing equations. To incorporate geologic and petrologic information into the calculation of the new effective dispersivity parameters, though, we need help from engineers and hydrogeologists, who have detailed knowledge of the types of measurements that are feasible and a sense of the statistical structure of the conductivity fields that occur in particular formations.

We have also devoted considerable effort to the development of finite-difference and finite-element approximations to the equations governing groundwater flow and contaminant transport. For example, we have explored the calculation of accurate fluid velocities from the groundwater flow equations, the resolution of steep concentration gradients in moving contaminant plumes, and the efficient discretization of multiphase flows, such as those that occur beneath leaking gasoline tanks, TCE spills, and other nonaqueous liquid sources.

Among the most promising methods for approximating the groundwater flow equation are *mixed finite-element methods.* These methods solve the coupled system comprising the mass balance for water and Darcy's law. By choosing appropriate shape functions, one can generate approximate solutions for the water velocity having the same order of accuracy as the approximate hydraulic head. In contrast, standard finite-element and finite-difference methods, which differentiate numerical heads to compute Darcy velocities, yield approximate velocities that are less accurate than the heads and therefore less useful in modeling contaminant transport.

In the realm of transport equations, we have focused much of our attention on cases where advective transport dominates the effects of hydrodynamic dispersion — a case of prime interest in many sandstone and unconsolidated aquifers. Plumes in this regime tend to have persistent, steep concentration gradients that are difficult to resolve numerically with coarse-celled grids. One strategy that we have used to overcome this difficulty is the use of *adaptive local grid refinement.* The idea is to assign smaller grid cells to regions of the plume needing greater numerical resolution. However, the fact that the plume is moving makes implementation of the idea on the computer a delicate task. Among the algorithmic difficulties that we have tried to address are the disruption of efficient matrix structures associated with regular, coarse grids and the poor numerical conditioning that results from the use of cells

having widely disparate sizes.

We have employed a variety of other techniques in this arena. For example, it is possible to adopt a "hybrid" coordinate system in discretizing the contaminant transport equation, measuring temporal rates of concentration change along the paths of fluid particles, not at fixed spatial points. This *modified method of characteristics* allows more accurate timestepping than the usual formulation. Also, we have investigated the use of *finite-element collocation*, a high-accuracy discretization technique, to reduce the numerical smearing associated with many low-order finite-element and finite-difference methods.

In modeling multiphase flows, we have developed a variety of improvements to the standard discretizations. Among these are mass-conserving formulations of the time derivatives in vadose-zone flows, splittings of the nonlinear fractional flow in saturation equations to facilitate the use of the modified method of characteristics, and the analysis of finite-element methods in the mathematically difficult case when capillary pressure gradients are negligible or degenerate. By no means has our work settled all of the important issues in this class of flows. Nonaqueous-phase contaminant flow promises to remain a significant challenge for modelers and engineers for years to come.

Finally, our research has led to the development of several new approaches for solving the large matrix equations associated with discretizations of the governing equations. For example, we have examined iterative schemes for solving the mixed finite-element equations that use *conjugate-gradient* and *multigrid* techniques to overcome the slow convergence associated with highly heterogeneous conductivity fields. We have also explored *alternating-direction methods* for decomposing multidimensional problems to one-dimensional structures that can be solved efficiently on parallel-processing computers. We have also developed efficient ways to decompose locally refined grids into coupled coarse-grid problems and fine-grid problems, thereby overcoming the disruption of regular coarse-grid structures and the conditioning problems associated with local grid refinement.

Mathematicians often unwittingly give the impression that numerical problems associated with groundwater modeling are under control and that the remaining difficulties are attributable to poor input data. However, poor data constitute only part of the problem. Many of the standard numerical techniques are blunt instruments in the presence of the mathematically difficult features of groundwater flow and transport. We aim to sharpen these instruments.

ACKNOWLEDGMENTS: The Wyoming Water Research Center supported much of our work. We have also received support from a variety of other sources, including NSF, ONR, EPA, and the Enhanced Oil Recovery Institute.

SIAM J. SCI. STAT. COMPUT.
Vol. 13, No. 3, pp. 794–814, May 1992
© 1992 Society for Industrial and Applied Mathematics
011

WELL-CONDITIONED ITERATIVE SCHEMES FOR MIXED FINITE-ELEMENT MODELS OF POROUS-MEDIA FLOWS*

MYRON B. ALLEN†, RICHARD E. EWING†, AND PENG LU‡

Abstract. Mixed finite-element methods are attractive for modeling flows in porous media since they can yield pressures and velocities having comparable accuracy. In solving the resulting discrete equations, however, poor matrix conditioning can arise both from spatial heterogeneity in the medium and from the fine grids needed to resolve that heterogeneity. This paper presents two iterative schemes that overcome these sources of poor conditioning. The first scheme overcomes poor conditioning resulting from the use of fine grids. The idea behind the scheme is to use spectral information about the matrix associated with the discrete version of Darcy's law to precondition the velocity equations, employing a multigrid method to solve mass-balance equations for pressure or head. This scheme still exhibits slow convergence when the permeability or hydraulic conductivity is highly variable in space. The second scheme, based on the first, uses mass lumping to precondition the Darcy equations, thus requiring more work per iteration and minor modifications to the multigrid algorithm. However, the scheme is insensitive to heterogeneities. The overall approach should also be useful in such applications as electric field simulation and heat transfer modeling when the media in question have spatially variable material properties.

Key words. mixed finite elements, iterative solution schemes, heterogeneous porous media

AMS(MOS) subject classification. 65

1. Introduction.

We consider methods for solving discrete approximations to the equations governing single-fluid flow in a porous medium. If the flow is steady and two-dimensional with no gravity drive, Darcy's law and the mass balance take the following forms:

$$\begin{aligned} \mathbf{u} &= -K \operatorname{grad} p \quad \text{in } \Omega, \\ \operatorname{div} \mathbf{u} &= f \quad \text{in } \Omega. \end{aligned} \tag{1.1}$$

Here $\mathbf{u}, p$, and f represent the Darcy velocity, pressure, and source term, respectively. For simplicity, we take the spatial domain to be a square, scaled so that $\Omega = (0,1) \times (0,1)$. The coefficient $K(x,y)$ is the *mobility*, defined as the ratio of the permeability of the porous medium to the dynamic viscosity of the fluid. In applications to underground flows, the structure of K may be quite complex, depending on the lithology of the porous medium and the composition of the fluid. We assume, however, that this ratio is bounded and integrable on $\bar{\Omega}$ and satisfies $K \geq K_{\text{inf}} > 0$. We impose the boundary condition $p = 0$ on $\partial\Omega$, so that p effectively represents the deviation in pressure from a reference value known along $\partial\Omega$.

Scientists modeling contaminant flows in groundwater or solvent flows in oil reservoirs often need accurate finite-element approximations of $\mathbf{u}$ and p simultaneously. For this reason, mixed finite-element methods for solving the system (1.1) are particularly attractive, since they can yield approximations to $\mathbf{u}$ and p that have comparable accuracy [1], [5], [9]. The key to achieving such approximations is the use of appropriate piecewise polynomial trial spaces, such as those proposed by

* Received by the editors April 12, 1990; accepted for publication (in revised form) February 20, 1991. This research was supported by National Science Foundation grant RII-8610680, Office of Naval Research grant 0014-88-K-0370, and the Wyoming Water Reseach Center.
† Department of Mathematics, University of Wyoming, Laramie, Wyoming 82071-3036.
‡ Department of Mathematics, University of Georgia, Athens, Georgia 30605.

Raviart and Thomas [11]. As we review in §2, if we use the lowest-degree Raviart–Thomas spaces, the mixed formulation yields systems of discrete equations that have the form

$$\begin{aligned} AU + NP &= 0, \\ N^T U &= F. \end{aligned} \tag{1.2}$$

Here, U and P signify vectors containing nodal values of the trial functions for $\mathbf{u}$ and p, defined on a grid over Ω, and A and N are matrices. As we illustrate below, the matrix A contains information about the spatially varying material property K, while N and N^T are essentially finite-difference matrices.

Equations (1.2) can be quite difficult to solve efficiently, for the following reasons. When K varies over short distances, accurate finite-element approximations require fine grids on Ω. For example, one might choose grids fine enough to allow reasonable approximations of K by piecewise constant functions. Fine grids, however, typically yield poorly conditioned matrix equations. For classical stationary iterative schemes, this increase in the condition number of the system leads to slow convergence, no matter how "nice" K may be [2, §4.11]. The problem is compounded whenever K exhibits large spatial variations, as can occur near lithologic changes in the porous medium or sharp contacts between fluids of different viscosity. In such problems, as we shall demonstrate, the poor conditioning associated with spatial variability typically aggravates that associated with the fine grids needed to resolve the physics of the problem. Thus, in problems with significant material heterogeneity, methods that are relatively insensitive to these two sources of poor conditioning can have considerable utility.

In this paper we discuss two iterative schemes for the mixed-method equations (1.2). The first scheme possesses convergence rates that are independent of the fineness of the grid. The second scheme, derived from the first, also overcomes the sensitivity to the spatial structure of K, at the expense of somewhat more computation per iteration. Briefly, the first scheme proceeds as follows: Let $(U^{(0)}, P^{(0)})$ be initial guesses for the value of (U, P). Then the kth iterate for (U, P) is the solution of

$$\begin{pmatrix} \omega I & N \\ N^T & 0 \end{pmatrix} \begin{pmatrix} U^{(k)} \\ P^{(k)} \end{pmatrix} = \begin{pmatrix} 0 \\ F \end{pmatrix} + \begin{pmatrix} \omega I - A & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} U^{(k-1)} \\ P^{(k-1)} \end{pmatrix}, \tag{1.3}$$

where I stands for the identity matrix and ω signifies a parameter, discussed below, that is related to the spectral radius $\rho(A)$ of A. For each iteration level k, the main computational work in (1.3) is to solve a linear system of the form $(\omega^{-1} N^T N) P^{(k)} = G^{(k-1)}$. However, the matrix $\omega^{-1} N^T N$ remains vulnerable to the poor conditioning associated with fine grids. We overcome this difficulty by using a multigrid scheme to solve for $P^{(k)}$, thereby greatly reducing the computational work in each iteration.

An interesting feature of this approach is that $N^T N$ is essentially the matrix associated with the five-point difference approximation to the Laplace operator with Dirichlet boundary conditions. Hence, the multigrid portion of the scheme does not encounter the variable coefficient, and the algorithm is particularly simple. The price paid for this simplicity, as we shall see, is sensitivity to the poor conditioning associated with spatial variability.

To overcome this second source of trouble, we modify the first scheme to get new ones of the form

$$\begin{pmatrix} D & N \\ N^T & 0 \end{pmatrix} \begin{pmatrix} U^{(k)} \\ P^{(k)} \end{pmatrix} = \begin{pmatrix} 0 \\ F \end{pmatrix} + \begin{pmatrix} D - A & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} U^{(k-1)} \\ P^{(k-1)} \end{pmatrix}, \tag{1.4}$$

where D denotes a diagonal matrix that we compute from A. This new class of schemes requires us to invert $N^T DN$, which we again do using a multigrid method to preserve h-independence of the convergence rate. While the multigrid method must now accommodate spatially varying coefficients, the overall scheme possesses the advantage that its convergence rate is independent of the spatial structure of K, provided K is piecewise constant on the grids of interest.

Our paper has the following format. In §2 we review the mixed finite-element method that we use. Section 3 describes the first iterative scheme in more detail and analyzes its convergence. In §4 we discuss the application of multigrid ideas to the first scheme. Much of the motivation and groundwork for the second class of iterative schemes resides in §§3 and 4. In §5 we present some numerical results for this algorithm. Section 6 describes the modifications necessary to produce the second class of iterative schemes and presents numerical results illustrating good convergence rates even in the presence of heterogeneities.

2. A mixed finite-element method. We begin with a brief review of the mixed finite-element method, following the notation of Ewing and Wheeler [8]. Let $H(\text{div}, \Omega) = \{\mathbf{v} \in L^2(\Omega) \times L^2(\Omega) : \text{div } \mathbf{v} \in L^2(\Omega)\}$. The variational form for (1.1) is as follows: Find a pair $(\mathbf{u}, p) \in H(\text{div}, \Omega) \times L^2(\Omega)$ such that

$$\int_\Omega \frac{\mathbf{u} \cdot \mathbf{v}}{K} dx\, dy - \int_\Omega p \text{ div } \mathbf{v}\, dx\, dy = 0 \quad \forall\, \mathbf{v} \in H(\text{div}, \Omega),$$

(2.1)

$$\int_\Omega (\text{div } \mathbf{u} - f) q\, dx\, dy = 0 \quad \forall\, q \in L^2(\Omega).$$

By our assumptions on K, there exist constants $K_{\text{inf}}, K_{\text{sup}}$ such that $0 < K_{\text{inf}} \leq K \leq K_{\text{sup}}$. Implicit in these equations is also the assumption that K^{-1} is integrable on $\bar{\Omega}$.

To discretize the system (2.1), let $\Delta_x = \{0 = x_0 < x_1 < \cdots < x_m = 1\}$ be a set of points on the x-axis and $\Delta_y = \{0 = y_0 < y_1 < \cdots < y_n = 1\}$ a set of points on the y-axis. Let $\Delta_h = \Delta_x \times \Delta_y$ be the rectangular grid on Ω with nodes $\{(x_i, y_j)\}_{i=0,j=0}^{m,n}$. The mesh of this grid is

$$h = \max_{i,j} \{x_i - x_{i-1},\, y_j - y_{j-1}\}.$$

We assume throughout the paper that Δ_x and Δ_y are quasi-uniform in the sense that $x_i - x_{i-1} \geq \alpha h$ and $y_j - y_{j-1} \geq \alpha h$ for some fixed $\alpha \in (0, 1)$. With Δ_h we associate a finite-element subspace $\mathbf{Q}_h \times V_h$ of $H(\text{div}, \Omega) \times L^2(\Omega)$. The "velocity space" is $\mathbf{Q}_h = Q_h^x \times Q_h^y$, where Q_h^x and Q_h^y are both tensor-product spaces of one-dimensional, finite-element spaces. In particular, we use the lowest-order Raviart–Thomas spaces in which Q_h^x contains functions that are piecewise linear and continuous on Δ_x and piecewise constant on Δ_y. Similarly, Q_h^y contains functions that are piecewise linear and continuous on Δ_y and piecewise constant on Δ_x. The "pressure space" V_h consists of functions that are piecewise constant on Δ_h.

Given these approximating spaces, the corresponding mixed finite-element method for solving (2.1) is as follows: Find a pair $(\mathbf{u}_h, p_h) \in \mathbf{Q}_h \times V_h$ such that

$$\int_\Omega \frac{\mathbf{u}_h \cdot \mathbf{v}_h}{K} dx\, dy - \int_\Omega p_h \text{ div } \mathbf{v}_h dx\, dy = 0 \qquad \forall\, \mathbf{v}_h \in \mathbf{Q}_h,$$

(2.2)

$$\int_\Omega (\text{div } \mathbf{u}_h - f) q_h dx\, dy = 0 \qquad \forall\, q_h \in V_h.$$

This finite-element discretization yields approximations $\mathbf{u}_h$ and p_h whose global errors are both $O(h)$ in the norm $\|\cdot\|_{L^2(\Omega)}$. Ewing, Lazarov, and Wang [6] also prove superconvergence results that guarantee smaller errors at special points in Ω. This phenomenon appears in our numerical examples in §5. In contrast, standard approaches solve for approximations to p and then numerically differentiate to compute $\mathbf{u} = -K \operatorname{grad} p$, thereby losing an order of accuracy in the velocity field [1].

To see the linear algebraic equations implied by (2.2), suppose $\mathbf{u}_h$ and p_h have the expansions

$$\mathbf{u}_h(x,y) = \left(\sum_{i=0}^{m} \sum_{j=1}^{n} U^x_{i,j} \phi^x_{i,j}(x,y), \sum_{i=1}^{m} \sum_{j=0}^{n} U^y_{i,j} \phi^y_{i,j}(x,y) \right),$$

$$p_h(x,y) = \sum_{i=1}^{m} \sum_{j=1}^{n} P_{i,j} \psi_{i,j}(x,y).$$

Here, $\phi^x_{i,j}$, $\phi^y_{i,j}$, and $\psi_{i,j}$ signify elements in the standard nodal bases for Q^x_h, Q^y_h, and V_h. Define the column vectors $U \in \mathbb{R}^{2mn+m+n}$, $P \in \mathbb{R}^{mn}$ containing the nodal unknowns as follows:

$$U^T = (U^x_{0,1}, U^x_{1,1}, \cdots, U^x_{m,1}, \cdots, U^x_{0,n}, U^x_{1,n}, \cdots, U^x_{m,n},$$

$$U^y_{1,0}, U^y_{1,1}, \cdots, U^y_{1,n}, \cdots, U^y_{m,0}, U^y_{m,1}, \cdots, U^y_{m,n}), \tag{2.3}$$

$$P^T = (P_{1,1}, P_{2,1}, \cdots, P_{m,1}, \cdots, P_{1,n}, P_{2,n}, \cdots, P_{m,n}).$$

Figure 1 shows how to associate these coefficients with nodes on a spatial grid Δ_h with $m = 4$, $n = 3$.

With these bases, the problem (2.2) has a matrix representation of the form

$$\begin{pmatrix} A & N \\ N^T & 0 \end{pmatrix} \begin{pmatrix} U \\ P \end{pmatrix} = \begin{pmatrix} 0 \\ F \end{pmatrix}. \tag{2.4}$$

Here A is a symmetric, positive definite matrix having the block structure

$$A = \begin{pmatrix} A^x & 0 \\ 0 & A^y \end{pmatrix},$$

in which $A^x \in \mathbb{R}^{(m+1)n \times (m+1)n}$ and $A^y \in \mathbb{R}^{m(n+1) \times m(n+1)}$ have entries of the form

$$\int_\Omega \frac{\phi^x_{i,j} \phi^x_{k,\ell}}{K} dx\, dy, \qquad \int_\Omega \frac{\phi^y_{i,j} \phi^y_{k,\ell}}{K} dx\, dy,$$

respectively. Note that these entries contain information about the spatially varying coefficient K. The matrix N has the block structure

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

where $N^x \in \mathbb{R}^{(m+1)n \times mn}$ and $N^y \in \mathbb{R}^{m(n+1) \times mn}$ have entries given, respectively, by

$$\int_\Omega \psi_{i,j} \frac{\partial \phi^x_{k,\ell}}{\partial x} dx\, dy, \qquad \int_\Omega \psi_{i,j} \frac{\partial \phi^y_{k,\ell}}{\partial y} dx\, dy.$$

By calculating these integrals, one readily confirms that N^x and N^y reduce to the usual difference approximations to $\partial/\partial x$ and $\partial/\partial y$. The vector $F \in \mathbb{R}^{mn}$ has entries given by the integrals $\int_\Omega f \psi_{i,j} dx\, dy$. The appendix to this paper gives more detail on the construction of A and N.

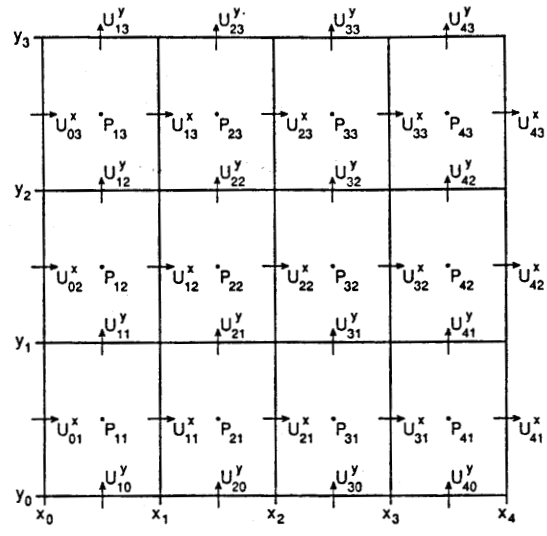

FIG. 1. *Sample* 4×3 *rectangular grid on* $\Omega = (0,1) \times (0,1)$, *showing locations of the nodal unknowns in the velocity and pressure trial functions.*

3. An h-independent iterative method. Our first iterative scheme for solving the discrete system (2.4) is as follows.

ALGORITHM 1. Beginning with initial guess $\left(U^{(0)}, P^{(0)}\right)^T$ for (U,P), the kth iterate $\left(U^{(k)}, P^{(k)}\right)^T$ is the solution of

$$\begin{pmatrix} \omega I & N \\ N^T & 0 \end{pmatrix} \begin{pmatrix} U^{(k)} \\ P^{(k)} \end{pmatrix} = \begin{pmatrix} 0 \\ F \end{pmatrix} + \begin{pmatrix} \omega I - A & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} U^{(k-1)} \\ P^{(k-1)} \end{pmatrix}, \tag{3.1}$$

where $I \in \mathbb{R}^{(2mn+m+n)\times(2mn+m+n)}$ is the identity matrix and ω is a parameter chosen to satisfy $\omega \geq \rho(A)$.

Here, $\rho(A)$ denotes the spectral radius of the matrix A. Later in this section we discuss a practical way to pick ω that does not require detailed knowledge of the spectrum of A.

Computationally, Algorithm 1 has the following compact form: Given an initial guess $\left(U^{(0)}, P^{(0)}\right)^T$, compute $\left(U^{(k)}, P^{(k)}\right)^T$ by executing three steps:

$$\text{(i) } G^{(k-1)} \leftarrow -F + \omega^{-1} N^T (\omega I - A) U^{(k-1)}, \tag{3.2}$$

$$\text{(ii) Solve } \omega^{-1} N^T N P^{(k)} = G^{(k-1)}, \tag{3.3}$$

$$\text{(iii) } \omega U^{(k)} \leftarrow (\omega I - A) U^{(k-1)} - N P^{(k)}. \tag{3.4}$$

In each iteration, the main computational work is to solve for $P^{(k)} = \omega (N^T N)^{-1} G^{(k-1)}$. An easy calculation shows that the matrix $\omega^{-1}(N^T N)$ is positive definite, being pro-

portional to the standard five-point, finite-difference Laplace operator applied to $P^{(k)}$. Therefore, we expect the numerical solution for $P^{(k)}$ using stationary iterative methods to be plagued by poor conditioning when the grid mesh h is small.

This observation leads us to use a multigrid scheme to get approximations to $P^{(k)}$. (In fact, any fast solver for the five-point discrete Laplacian operator would be appropriate here.) Such a device preserves the h-independence of the overall scheme's convergence rate. We discuss this facet of the algorithm in more detail in the next section. For now let us analyze the convergence properties of the overall iterative scheme, assuming an efficient "black-box" solver for $P^{(k)}$.

We begin by writing (3.1) as a stationary iterative scheme

$$\begin{pmatrix} U^{(k)} \\ P^{(k)} \end{pmatrix} = L + M \begin{pmatrix} U^{(k-1)} \\ P^{(k-1)} \end{pmatrix}, \tag{3.5}$$

where

$$L = \begin{pmatrix} \omega I & N \\ N^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} 0 \\ F \end{pmatrix},$$

$$M = \begin{pmatrix} \omega I & N \\ N^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} \omega I - A & 0 \\ 0 & 0 \end{pmatrix}.$$

The convergence of Algorithm 1 depends on the spectral radius of the matrix M, for which the following proposition gives a bound.

PROPOSITION 3.1. *Let*

$$0 < \lambda_{\min} \leq \cdots \leq \lambda_{\max} \tag{3.6}$$

be the eigenvalues of the matrix A, and let $\omega \geq \lambda_{\max}$. Then the spectral radius of M obeys the estimate

$$\rho(M) \leq 1 - \frac{\lambda_{\min}}{\omega}. \tag{3.7}$$

Proof. Let $\lambda \neq 0$ be an eigenvalue of M with eigenvector $(U_\lambda, P_\lambda)^T$. Thus

$$M \begin{pmatrix} U_\lambda \\ P_\lambda \end{pmatrix} \equiv \begin{pmatrix} \omega I & N \\ N^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} \omega I - A & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} U_\lambda \\ P_\lambda \end{pmatrix} = \lambda \begin{pmatrix} U_\lambda \\ P_\lambda \end{pmatrix}, \tag{3.8}$$

so

$$(\omega I - A)U_\lambda = \lambda(\omega U_\lambda + NP_\lambda), \tag{3.9a}$$

$$0 = \lambda N^T U_\lambda. \tag{3.9b}$$

Since $(U_\lambda, P_\lambda)^T \neq 0$, (3.9a) shows that $U_\lambda \neq 0$; however, U_λ may be complex. Let U_λ^H denote its Hermitian conjugate. If we multiply (3.9a) by U_λ^H, observe that N is a real matrix, and apply (3.9b), we obtain

$$\begin{aligned} U_\lambda^H(\omega I - A)U_\lambda &= \lambda\omega U_\lambda^H U_\lambda + \lambda(N^T U_\lambda)^H P_\lambda \\ &= \lambda\omega U_\lambda^H U_\lambda. \end{aligned}$$

This equation allows us to conclude that

$$0 < |\lambda| = \left| \frac{U_\lambda^H (I - \omega^{-1} A) U_\lambda}{U_\lambda^H U_\lambda} \right| \leq \rho(I - \omega^{-1} A),$$

which implies

$$\rho(M) \leq \rho(I - \omega^{-1} A). \tag{3.10}$$

Also, by (3.6) and the fact that $\omega \geq \lambda_{\max}$, we have

$$\rho(I - \omega^{-1} A) \leq 1 - \frac{\lambda_{\min}}{\omega}.$$

These last two inequalities imply the desired bound (3.7). □

If we choose $\omega = \lambda_{\max} = \rho(A)$, then the estimate (3.7) for the spectral radius of the iteration matrix M becomes

$$\rho(M) \leq 1 - \frac{\lambda_{\min}}{\lambda_{\max}}.$$

To estimate $\lambda_{\min}/\lambda_{\max}$, the following proposition is helpful.

PROPOSITION 3.2. *For the matrix A appearing in* (2.4), *there exist constants k_0 and k_1, independent of h, such that*

$$k_0 h^2 U^T U \leq U^T A U \leq k_1 h^2 U^T U. \tag{3.11}$$

Proof. The representation of $\mathbf{u}_h$ given in (2.3) leads to the identity

$$U^T A U = \int_\Omega \frac{1}{K} |\mathbf{u}_h|^2 dx\, dy = \sum_{i=1}^{m} \sum_{j=1}^{n} \int_{\Omega_{i,j}} \frac{1}{K} |\mathbf{u}_h|^2 dx\, dy,$$

where $\Omega_{i,j} = (x_{i-1}, x_i) \times (y_{j-1}, y_j)$. Since K is bounded and integrable on $\Omega_{i,j}$, the mean value theorem for integrals [10, pp. 184–185] guarantees the existence of a number $K_{i,j}$, satisfying $\inf_{\Omega_{i,j}} K \leq K_{i,j} \leq \sup_{\Omega_{i,j}} K$, such that

$$\int_{\Omega_{i,j}} \frac{1}{K} |\mathbf{u}_h|^2 dx\, dy = \frac{1}{K_{i,j}} \int_{\Omega_{i,j}} |\mathbf{u}_h|^2 dx\, dy.$$

(If K^{-1} is continuous on $\bar{\Omega}_{i,j}$, then K^{-1} actually assumes the value $K_{i,j}^{-1}$ somewhere on $\Omega_{i,j}$.) Calculating the last integral using our basis for $\mathbf{Q}_h$, we get

$$U^T A U = \sum_{i=1}^{m} \sum_{j=1}^{n} \frac{a_{ij}}{6K_{ij}} \left[\begin{pmatrix} U^x_{i-1,j} \\ U^x_{i,j} \end{pmatrix}^T \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} U^x_{i-1,j} \\ U^x_{i,j} \end{pmatrix} \right.$$

$$\left. + \begin{pmatrix} U^y_{i,j-1} \\ U^y_{i,j} \end{pmatrix}^T \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} U^y_{i,j-1} \\ U^y_{i,j} \end{pmatrix} \right],$$

where a_{ij} signifies the area of $\Omega_{i,j}$. To simplify notation, we notice that the 2×2 matrix appearing in each term of this sum is positive definite. This observation allows us to define a new norm on $\mathbb{R}^2$ as follows:

$$\left\| \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \right\|_A^2 = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix}^T \begin{pmatrix} 2 & 1 \\ 1 & 2 \end{pmatrix} \begin{pmatrix} U_1 \\ U_2 \end{pmatrix}.$$

If $\|\cdot\|_2$ denotes the usual Euclidean norm on $\mathbb{R}^2$, then it is easy to check that $\|\cdot\|_2^2 \le \|\cdot\|_A^2 \le 3\|\cdot\|_2^2$. In terms of the new norm,

$$U^T A U = \sum_{i=1}^{m}\sum_{j=1}^{n} \frac{a_{ij}}{6K_{ij}} \left[\left\| \begin{pmatrix} U^x_{i-1,j} \\ U^x_{i,j} \end{pmatrix} \right\|_A^2 + \left\| \begin{pmatrix} U^y_{i,j-1} \\ U^y_{i,j} \end{pmatrix} \right\|_A^2 \right].$$

The quantity $U^T U$ is easier to calculate:

$$U^T U = \sum_{i=0}^{m}\sum_{j=1}^{n} |U^x_{ij}|^2 + \sum_{i=1}^{m}\sum_{j=0}^{n} |U^y_{ij}|^2. \tag{3.12}$$

Now we use the bounds on K and the quasi uniformity of Δ_h to observe that

$$\begin{aligned} U^T A U &\ge \frac{\alpha^2 h^2}{6K_{\sup}} \sum_{i=1}^{m}\sum_{j=1}^{n} \left[\left\| \begin{pmatrix} U^x_{i-1,j} \\ U^x_{i,j} \end{pmatrix} \right\|_A^2 + \left\| \begin{pmatrix} U^y_{i,j-1} \\ U^y_{i,j} \end{pmatrix} \right\|_A^2 \right] \\ &\ge \frac{\alpha^2 h^2}{6K_{\sup}} \sum_{i=1}^{m}\sum_{j=1}^{n} \left[\left\| \begin{pmatrix} U^x_{i-1,j} \\ U^x_{i,j} \end{pmatrix} \right\|_2^2 + \left\| \begin{pmatrix} U^y_{i,j-1} \\ U^y_{i,j} \end{pmatrix} \right\|_2^2 \right] \\ &\ge \frac{\alpha^2 h^2}{6K_{\sup}} \left[\sum_{i=0}^{m}\sum_{j=1}^{n} \left\| \begin{pmatrix} 0 \\ U^x_{i,j} \end{pmatrix} \right\|_2^2 + \sum_{i=1}^{m}\sum_{j=0}^{n} \left\| \begin{pmatrix} 0 \\ U^y_{i,j} \end{pmatrix} \right\|_2^2 \right] \\ &= \frac{\alpha^2 h^2}{6K_{\sup}} U^T U. \end{aligned}$$

This observation establishes the first inequality in (3.11), since we can take $k_0 = \alpha^2/6K_{\sup}$. To prove the second inequality in (3.11), we rewrite (3.12) as follows:

$$\begin{aligned} U^T U = \frac{1}{2} &\sum_{i=0}^{m}\sum_{j=0}^{n} \left[\left\| \begin{pmatrix} 0 \\ U^x_{i,j} \end{pmatrix} \right\|_2^2 + \left\| \begin{pmatrix} 0 \\ U^y_{i,j} \end{pmatrix} \right\|_2^2 \right] \\ &+ \frac{1}{2} \sum_{i=1}^{m+1}\sum_{j=1}^{n+1} \left[\left\| \begin{pmatrix} U^x_{i-1,j} \\ 0 \end{pmatrix} \right\|_2^2 + \left\| \begin{pmatrix} U^y_{i,j-1} \\ 0 \end{pmatrix} \right\|_2^2 \right], \end{aligned}$$

where we agree that $U^x_{ij} = 0$ if either $j = 0$ or $j = n+1$, and $U^y_{ij} = 0$ if either $i = 0$ or $i = m+1$. Hence,

$$\begin{aligned} U^T U &\ge \frac{1}{2} \sum_{i=1}^{m}\sum_{j=1}^{n} \left[\left\| \begin{pmatrix} U^x_{i-1,j} \\ U^x_{i,j} \end{pmatrix} \right\|_2^2 + \left\| \begin{pmatrix} U^y_{i,j-1} \\ U^y_{i,j} \end{pmatrix} \right\|_2^2 \right] \\ &\ge \frac{1}{6} \sum_{i=1}^{m}\sum_{j=1}^{n} \left[\left\| \begin{pmatrix} U^x_{i-1,j} \\ U^x_{i,j} \end{pmatrix} \right\|_A^2 + \left\| \begin{pmatrix} U^y_{i,j-1} \\ U^y_{i,j} \end{pmatrix} \right\|_A^2 \right] \\ &\ge \frac{K_{\inf}}{h^2} U^T A U. \end{aligned}$$

We conclude that $U^T A U \le k_1 h^2 U^T U$, where $k_1 = 1/K_{\inf}$. □

If we apply Proposition 3.2 to the case when U is an eigenvector of A associated with the eigenvalue $\lambda_{\min}$ or $\lambda_{\max}$, respectively, we find that $\lambda_{\min} \geq \alpha^2 h^2 / 6K_{\sup}$ and $\lambda_{\max} \leq h^2 / K_{\inf}$. Therefore, provided we choose $\omega \geq \lambda_{\max}$ in Algorithm 1, the spectral radius of our iteration matrix M obeys the bound

$$\rho(M) \leq 1 - \frac{\alpha^2 K_{\inf}}{6K_{\sup}}. \tag{3.13}$$

Notice that the right side of this inequality is a constant independent of h. This is the sense in which the convergence rate of Algorithm 1 is independent of h.

Two remarks about the practical implications of the estimate (3.13) are in order. First, the bound on $\rho(M)$ depends strongly on the nature of the coefficient $K(x, y)$. In particular, if $K_{\inf}/K_{\sup}$ is very small, reflecting a high degree of heterogeneity in the physical problem, then we can expect the actual convergence of the algorithm to be slow, albeit independent of grid mesh. Several examples in §5 confirm this expectation. Second, even though the bound (3.13) suggests choosing $\omega = \lambda_{\max}$ to accelerate iterative convergence, this choice is impractical owing to the expense of calculating $\lambda_{\max}$. In practice, we typically pick $\omega = \|A\|_\infty \geq \lambda_{\max}$. This choice is easily computable as the maximum row sum of A, and it preserves h-independence of convergence rate, even though it may be theoretically nonoptimal.

4. Application of a multigrid solver. As we have mentioned, the computation of the pressure iterate $P^{(k)}$ in step (ii) of Algorithm 1 is inefficient if we use direct schemes or classical stationary iterative methods on fine grids. However, the fact that $\omega^{-1} N^T N$ is essentially the finite-difference Laplacian operator motivates us to reduce the computational work for each iteration by calculating an approximation to the kth pressure iterate by using several cycles of a multigrid method on the system (3.3). We refer the reader to [3] for a discussion of the multigrid approach and for a Fortran code applicable in the context of our problem. The modified scheme is as follows.

ALGORITHM 2. Begin with an initial guess $\left(U^{(0)}, P^{(0)}\right)^T$, and suppose that we have computed $\left(U^{(k-1)}, P^{(k-1)}\right)^T$. Compute a new approximation $\left(U^{(k)}, P^{(k)}\right)^T$ using the following steps:

1. Compute the residual,

$$G^{(k-1)} \leftarrow -F + N^T (I - \omega^{-1} A) U^{(k-1)}. \tag{4.1}$$

2. Let $\hat{P}^{(k)}$ denote the exact solution of the problem

$$\omega^{-1} N^T N \hat{P}^{(k)} = G^{(k-1)}. \tag{4.2}$$

   Calculate an approximation $P^{(k)}$ of $\hat{P}^{(k)}$ by applying r cycles of the multigrid algorithm [3] to (4.2), using $P^{(k-1)}$ as initial guess. (We discuss the choice of r below.)
3. Compute $U^{(k)}$ as in Algorithm 1:

$$\omega U^{(k)} \leftarrow (\omega I - A) U^{(k-1)} - N P^{(k)}. \tag{4.3}$$

Multigrid methods for solving elliptic problems have an advantage that is quite relevant to the conditioning problems associated with fine grids: Each cycle has a convergence rate that is independent of h [4, Chap. 4]. Therefore, we need only show

that we can choose a *fixed* number r of multigrid cycles such that each iteration of Algorithm 2 reduces the error norm by an appropriate factor close to $\rho(M)$. We do this in Proposition 4.1. Since the factor is independent of h, Algorithm 2 has convergence rate independent of h.

We begin by defining norms on the "pressure" and "velocity" spaces that will make the proof easier. Any $p_h \in V_h$ has a representation

$$p_h(x,y) = \sum_{i,j} P_{i,j}\psi_{i,j}(x,y).$$

Taking advantage of the fact that N^TN is positive definite, we compute a norm of the vector

$$P = (P_{1,1}, P_{2,1}, \cdots, P_{m,1}, \cdots, P_{1,n}, P_{2,n}, \cdots, P_{m,n})^T$$

by setting $||P||_h^2 = P^T(\omega^{-1}N^TN)P$. On the other hand, any $\mathbf{u}_h \in \mathbf{Q}_h$ has a representation

$$\mathbf{u}_h(x,y) = \left(\sum_{i,j} U^x_{i,j}\phi^x_{i,j}(x,y), \sum_{i,j} U^y_{i,j}\phi^y_{i,j}(x,y)\right).$$

We compute a norm of the vector

$$U = \big(U^x_{0,1}, U^x_{1,1}, \cdots, U^x_{m,1}, \cdots, U^x_{0,n}, U^x_{1,n}, \cdots, U^x_{m,n},$$
$$U^y_{1,0}, U^y_{1,1}, \cdots, U^y_{1,n}, \cdots, U^y_{m,0}, U^y_{m,1}, \cdots, U^y_{m,n}\big)^T$$

by setting $||U||_\omega^2 = \omega U^TU$.

The norm $\|\cdot\|_\omega$ is just a scalar multiple of the Euclidean distance function $\|\cdot\|_2$, and since ω is a constant related to $\rho(A)$, $\|\cdot\|_\omega$ is actually a discrete analog of the Euclidean norm $\|\cdot\|_{L^2(\Omega)\times L^2(\Omega)}$ on the velocity space by Proposition 3.2. This norm is appropriate for measuring the convergence of velocity iterates $U^{(k)}$ to the true discrete approximation U. Also, since N^TN is just the positive definite matrix associated with the five-point difference approximation to the Laplace operator, the norm $\|\cdot\|_h$ is appropriate for measuring the rapidity with which the pressure iterates satisfy the discrete pressure equation (3.3) as the iterations progress. Ultimately, we want to relate our results to more familiar norms such as $\|\cdot\|_2$ and $\|\cdot\|_\infty$; for this step we shall rely on the equivalence of norms for finite-dimensional Euclidean spaces.

In the following proposition, we assume $\nu = \rho(I-\omega^{-1}A) < 1$. Thus ν is an upper bound on $\rho(M)$. Suppose the multigrid iteration used to approximate $\hat{P}^{(k)}$ in step (ii) of Algorithm 1 has convergence rate $\mu \in (0,1)$. This implies that, after r multigrid cycles for $P^{(k)}$ using $P^{(k-1)}$ as initial guess,

$$\left|\left|\hat{P}^{(k)} - P^{(k)}\right|\right|_h \le \mu^r \left|\left|\hat{P}^{(k)} - P^{(k-1)}\right|\right|_h. \tag{4.4}$$

PROPOSITION 4.1. *For any $\nu' \in (\nu, 1)$, there exists a number r of multigrid cycles such that*

$$\left|\left|P - P^{(k)}\right|\right|_h + \left|\left|U - U^{(k)}\right|\right|_\omega \le \nu'\left(\left|\left|P - P^{(k-1)}\right|\right|_h + \left|\left|U - U^{(k-1)}\right|\right|_\omega\right),$$

where (P,U) is the solution of the problem (2.4) *and $\left(P^{(k)},U^{(k)}\right)$ is the approximation to (P,U) produced by the kth iteration of Algorithm* 2.

Proof. Suppose we compute $\hat{U}^{(k)}$ according to (3.4) with the exact (nonmultigrid) pressure iterate $\hat{P}^{(k)}$. Thus,

$$\omega\hat{U}^{(k)} = (\omega I - A)U^{(k-1)} - N\hat{P}^{(k)}, \tag{4.5}$$

where $\hat{P}^{(k)}$ satisfies (4.2). Then from (2.4), (4.1), (4.2), and (4.5), we have

$$\omega\left(U-\hat{U}^{(k)}\right) + N\left(P-\hat{P}^{(k)}\right) = (\omega I - A)\left(U-U^{(k-1)}\right), \tag{4.6}$$

$$N^T\left(U-\hat{U}^{(k)}\right) = 0. \tag{4.7}$$

Multiplying (4.6) by $\left(U-\hat{U}^{(k)}\right)^T$ and using the identity (4.7), we get

$$\begin{aligned}\left\|U-\hat{U}^{(k)}\right\|_\omega^2 &= \left(U-\hat{U}^{(k)}\right)^T(\omega I - A)\left(U-U^{(k-1)}\right)\\ &\le \left\|U-\hat{U}^{(k)}\right\|_\omega \left\|(I-\omega^{-1}A)\left(U-U^{(k-1)}\right)\right\|_\omega\\ &\le \rho(I-\omega^{-1}A)\left\|U-\hat{U}^{(k)}\right\|_\omega \left\|U-U^{(k-1)}\right\|_\omega.\end{aligned}$$

Therefore, the velocity iterates obey the estimate

$$\left\|U-\hat{U}^{(k)}\right\|_\omega \le \nu\left\|U-U^{(k-1)}\right\|_\omega.$$

Similarly, multiplying (4.6) by $\left[\omega^{-1}N\left(P-\hat{P}^{(k)}\right)\right]^T$, we get

$$\begin{aligned}\left\|P-\hat{P}^{(k)}\right\|_h^2 &= \left(P-\hat{P}^{(k)}\right)^T N^T\omega^{-1}(\omega I - A)\left(U-U^{(k-1)}\right)\\ &\le \left\|\omega^{-1}N\left(P-\hat{P}^{(k)}\right)\right\|_\omega \left\|(I-\omega^{-1}A)\left(U-U^{(k-1)}\right)\right\|_\omega\\ &\le \left\|P-\hat{P}^{(k)}\right\|_h \rho(I-\omega^{-1}A)\left\|U-U^{(k-1)}\right\|_\omega.\end{aligned}$$

Hence, the pressure iterates obey the bound

$$\left\|P-\hat{P}^{(k)}\right\|_h \le \nu\left\|U-U^{(k-1)}\right\|_\omega.$$

Now we derive bounds on $\left\|P-P^{(k)}\right\|_h$ and $\left\|U-U^{(k)}\right\|_\omega$ in terms of their values at the previous iterative level. For $\left\|P-P^{(k)}\right\|_h$, we use the triangle equality and the multigrid estimate (4.4) to get

$$\begin{aligned}\left\|P-P^{(k)}\right\|_h &\le \left\|P-\hat{P}^{(k)}\right\|_h + \left\|\hat{P}^{(k)}-P^{(k)}\right\|_h\\ &\le \left\|P-\hat{P}^{(k)}\right\|_h + \mu^r\left\|\hat{P}^{(k)}-P^{(k-1)}\right\|_h \qquad (4.8)\\ &\le \left\|P-\hat{P}^{(k)}\right\|_h + \mu^r\left(\left\|P-\hat{P}^{(k)}\right\|_h + \left\|P-P^{(k-1)}\right\|_h\right).\end{aligned}$$

But the original iterative scheme (3.5) implies that

$$\begin{pmatrix} U - \hat{U}^{(k)} \\ P - \hat{P}^{(k)} \end{pmatrix} = M \begin{pmatrix} U - U^{(k-1)} \\ P - P^{(k-1)} \end{pmatrix}.$$

So, in light of the inequality (3.1) bounding $\rho(M)$ by ν, we have

$$\left|\left|\hat{P} - P^{(k)}\right|\right|_h \leq \rho(M) \left|\left|P - P^{(k-1)}\right|\right|_h \leq \nu \left|\left|P - P^{(k-1)}\right|\right|_h.$$

This inequality allows us to simplify (4.8), getting

$$\left|\left|P - P^{(k)}\right|\right|_h \leq (\nu + \mu^r + \nu\mu^r) \left|\left|P - P^{(k-1)}\right|\right|_h. \tag{4.9}$$

Turning to $||U - U^{(k)}||_\omega$, we use (4.3), multiplied by ω^{-1}, to write

$$\left(U - U^{(k)}\right) = (I - \omega^{-1}A)\left(U - U^{(k-1)}\right) + \omega^{-1}N\left(P - P^{(k)}\right).$$

This identity implies that

$$\begin{aligned} ||U - U^{(k)}||_\omega &\leq ||(I - \omega^{-1}A)\left(U - U^{(k-1)}\right)||_\omega + ||\omega^{-1}N\left(P - P^{(k)}\right)||_\omega \\ &\leq \nu ||U - U^{(k-1)}||_\omega + ||P - P^{(k)}||_h \\ &\leq \nu ||U - U^{(k-1)}||_\omega + (\nu + \mu^r + \nu\mu^r) ||P - P^{(k-1)}||_h \\ &\leq (\nu + \mu^r + \nu\mu^r)\left(||P - P^{(k-1)}||_h + ||U - U^{(k-1)}||_\omega\right). \end{aligned} \tag{4.10}$$

Combining the inequalities (4.9) and (4.10), we get

$$\begin{aligned} &||P - P^{(k)}||_h + ||U - U^{(k)}||_\omega \\ &\quad \leq (\nu + \mu^r + \nu\mu^r)\left(||P - P^{(k-1)}||_h + ||U - U^{(k-1)}||_\omega\right). \end{aligned}$$

Since $\mu < 1$, $\mu^r + \nu\mu^r \to 0$ as $r \to \infty$. We can therefore choose r large enough so that $\nu + \mu^r + \nu\mu^r + \nu \leq \nu' < 1$. In this way,

$$\left|\left|P - P^{(k)}\right|\right|_h + \left|\left|U - U^{(k)}\right|\right|_\omega \leq \nu'\left(\left|\left|P - P^{(k-1)}\right|\right|_h + \left|\left|U - U^{(k-1)}\right|\right|_\omega\right). \qquad \square$$

In view of the norm equivalence mentioned earlier, Proposition 4.1 leads us to expect that, if we choose ω as prescribed in §3, then the computed convergence rate

$$\bar{\mu} = \lim_{k \to \infty} \left[\frac{\|P - P^{(k)}\|_\infty + \|U - U^{(k)}\|_\infty}{\|P - P^{(0)}\|_\infty + \|U - U^{(0)}\|_\infty}\right]^{1/k} \tag{4.11}$$

should be a constant independent of h as $h \to 0$. In fact, for "generic" initial guesses, the contribution from the eigenvector associated with the largest magnitude eigenvalue of M will eventually dominate the error. We therefore expect $\bar{\mu}$ to give good approximations to $\rho(M)$ in computational practice [2, p. 129].

5. Numerical examples of h-independence. To test our results, we apply Algorithm 2 to several versions of the following boundary-value problem:

$$\begin{aligned} -\operatorname{div}[K(x,y)\operatorname{grad} p(x,y)] &= f(x,y), \qquad (x,y) \in \Omega, \\ p(x,y) &= 0, \qquad (x,y) \in \partial\Omega. \end{aligned} \tag{5.1}$$

We use the lowest-order, mixed finite-element method on grids with $h = 2^{-\ell}$, where $\ell = 4, 5, 6, 7, 8$. Each iteration of the solution scheme includes $r = 2$ V-cycles of the multigrid algorithm described in [3], where the coarsest grid in each cycle has mesh 2^{-1}, and the finest has mesh $2^{-\ell}$. We use the following realizations of the coefficient $K(x,y)$:

$$K_{\mathrm{I}}(x,y) = 1,$$

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

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

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

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

To confirm the convergence properties of the mixed finite-element method as $h \to 0$, we examine the exact and numerical solutions to (5.1) using $K = K_{\mathrm{II}}$ and taking $f(x,y)$ to be the function that results when the solution is $p(x,y) = x(1-x)\sin(\pi y) + y(1-y)\sin(\pi x)$. We compute the nodal error indicators $\|U_{\text{exact}} - U\|_\infty$ and $\|P_{\text{exact}} - P\|_\infty$, where U_{exact} and P_{exact} stand for the vectors of nodal values of the exact solutions $\mathbf{u}$ and p, and U and P are vectors containing nodal values of the finite-element approximations on a uniform grid of mesh h. Figure 2 shows plots of $\log \|U_{\text{exact}} - U\|_\infty$ and $\log \|P_{\text{exact}} - P\|_\infty$ versus $\log h$ having least-squares slopes of 1.899 and 2.000, respectively. These results suggest that the nodal values of U and P are accurate to $O(h^2)$, corroborating the equal-order accuracy available in the Raviart–Thomas subspaces and indicating superconvergent nodal values in accordance with the work of Ewing, Lazarov, and Wang [6].

To check the convergence properties of the iterative scheme, we examine the behavior of the ratio $\bar{\mu}$, defined in (4.11), for each of the choices of K. Our results, shown in Fig. 3, support the expectation that, as $h \to 0$, the convergence rate of the scheme tends to a constant independent of h. Notice however that, as K exhibits more spatial variation, the convergence of the algorithm becomes slower. Any effects of variability in K on the conditioning of the discrete equations still influence this first algorithm; the only effects of poor conditioning that we have eliminated so far are those associated with grid refinement.

6. Modified schemes for heterogeneous media. To mitigate the difficulties associated with spatial variability, we modify the first iterative scheme (3.1) to get a class of new schemes having the following form.

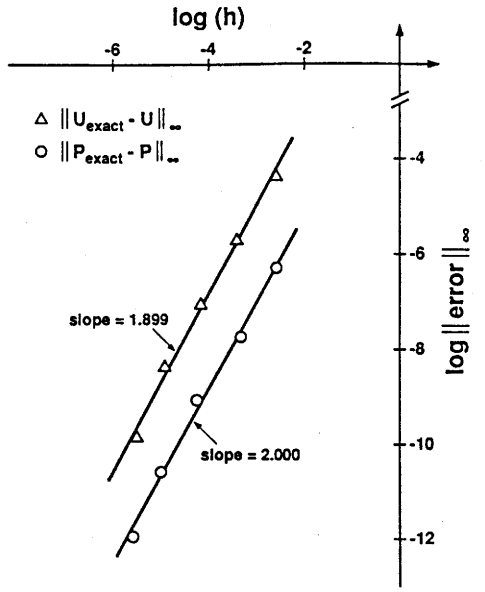

FIG. 2. *Convergence plot for the mixed finite-element scheme for Poisson's equation, using lowest-order Raviart–Thomas trial spaces. The plots demonstrate the rate of decrease in the nodal errors as* $h \to 0$.

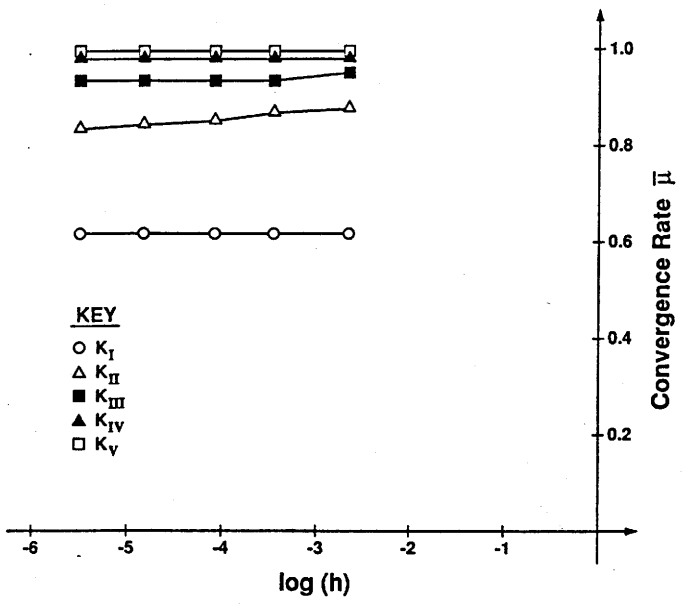

FIG. 3. *Rate of convergence* $\bar{\mu}$ *versus grid mesh* h *for Algorithm 2, using the various choices of coefficient* $K(x, y)$.

ALGORITHM 3. Given initial guess $\left(U^{(0)}, P^{(0)}\right)^T$, the kth iterate $\left(U^{(k)}, P^{(k)}\right)^T$ is the solution of

$$\begin{pmatrix} D & N \\ N^T & 0 \end{pmatrix} \begin{pmatrix} U^{(k)} \\ P^{(k)} \end{pmatrix} = \begin{pmatrix} 0 \\ F \end{pmatrix} + \begin{pmatrix} D-A & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} U^{(k-1)} \\ P^{(k-1)} \end{pmatrix}. \tag{6.1}$$

Here, the "preconditioning" matrix $D \in \mathbb{R}^{(2mn+m+n)\times(2mn+m+n)}$ is a diagonal matrix whose choice we discuss below.

When we construct D properly, the iteration matrix

$$M = \begin{pmatrix} D & N \\ N^T & 0 \end{pmatrix}^{-1} \begin{pmatrix} D-A & 0 \\ 0 & 0 \end{pmatrix} \tag{6.2}$$

has spectral radius that is independent of both h and the structure of K. The price we pay for this benefit is apparent in the computational form of the new algorithm:

$$\text{(i) } G^{(k-1)} \leftarrow -F + N^T D^{-1}(D-A)U^{(k-1)}, \tag{6.3}$$

$$\text{(ii) Solve } N^T D^{-1} N P^{(k)} = G^{(k-1)}, \tag{6.4}$$

$$\text{(iii) } U^{(k)} \leftarrow D^{-1}(D-A)U^{(k-1)} - D^{-1}NP^{(k)}. \tag{6.5}$$

In contrast to (3.3), solving for $P^{(k)}$ in the new scheme calls for the inversion of $N^T D^{-1} N$ instead of $N^T N$. Therefore, we must modify the multigrid segment of the algorithm to accommodate variable coefficients. As we discuss, this modification is fairly easy to make. This section establishes criteria for the construction of D, gives two examples that satisfy these criteria, comments on the multigrid solver used, and presents computational results.

As with the original scheme presented in §3, the key to the convergence of the new scheme is the spectral radius of the iteration matrix M defined in (6.2). The following proposition gives sufficient conditions under which $\rho(M) < 1$.

PROPOSITION 6.1. *Suppose D is a diagonal matrix with positive entries on the diagonal, and suppose there exist constants $b_1, b_2 \in (0,1)$ such that*

$$b_1 \le \frac{U^H A U}{U^H D U} \le 2 - b_2$$

for all vectors $U \in \mathbb{C}^{(m+1)n+m(n+1)}$. Then the iteration matrix M defined in (6.2) *satisfies*

$$0 < \rho(M) \le \max\{1-b_1, 1-b_2\} < 1. \tag{6.6}$$

Proof. Let $\lambda \neq 0$ be an eigenvalue of M with associated eigenvector $(U_\lambda, P_\lambda)^T$, as in Proposition 3.1. Then steps similar to those yielding (3.9) show that

$$\begin{aligned} (D-A)U_\lambda &= \lambda(DU_\lambda + NP_\lambda), \\ 0 &= \lambda N^T U_\lambda. \end{aligned}$$

Thus $U_\lambda^H(D-A)U_\lambda = \lambda U_\lambda^H D U_\lambda$, which is nonzero since D is positive definite. Therefore,

$$|\lambda| = \left|1 - \frac{U_\lambda^H A U_\lambda}{U_\lambda^H D U_\lambda}\right|.$$

Hence, using the hypothesized bounds on $U_\lambda^H AU_\lambda / U_\lambda^H DU_\lambda$, we have the desired inequalities (6.6). □

To use this proposition, we need estimates on $U^H AU$. Given the structure of A as shown in the Appendix, one can calculate a useful expression for $U^H AU$, assuming $U \in \mathbb{C}^{(m+1)n+m(n+1)}$ has the form $(U^x, U^y)^T$ indicated in (2.3). In particular,

$$U^H AU = \frac{1}{3}S(U) + \frac{1}{6}R(U),$$

where, in the notation of the Appendix,

$$S(U) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left(T_{i,j}^{\mathrm{I}}|U_{i-1,j}^x|^2 + T_{i,j}^{\mathrm{III}}|U_{i,j}^x|^2 + T_{i,j}^{\mathrm{IV}}|U_{i,j-1}^y|^2 + T_{i,j}^{\mathrm{VI}}|U_{i,j}^y|^2\right),$$

$$R(U) = \sum_{i=1}^{m}\sum_{j=1}^{n}\left[T_{i,j}^{\mathrm{II}}\left(\bar{U}_{i,j}^x U_{i-1,j}^x + \bar{U}_{i-1,j}^x U_{i,j}^x\right) + T_{i,j}^{\mathrm{V}}\left(\bar{U}_{i,j}^y U_{i,j-1}^y + \bar{U}_{i,j-1}^y U_{i,j}^y\right)\right].$$

Here, $\bar{z}$ denotes the complex conjugate of z. The coefficients $T_{i,j}^{\mathrm{I}}, \cdots, T_{i,j}^{\mathrm{VI}}$ appearing in these expressions are values depending on $K(x,y)$ and arising from applications of the mean value theorem for integrals over each cell $\bar{\Omega}_{i,j}$ in the finite-element grid Δ_h. By using the inequality $|w|^2 + |z|^2 \geq -2|w||z|$, we can estimate $R(U)$ as follows:

(6.7)

$$-2\sum_{i=1}^{m}\sum_{j=1}^{n}\left(T_{i,j}^{\mathrm{II}}|U_{i,j}^x||U_{i-1,j}^x| + T_{i,j}^{\mathrm{V}}|U_{i,j}^y||U_{i,j-1}^y|\right) \leq R(U)$$

$$\leq \sum_{i=1}^{m}\sum_{j=1}^{n}\left(T_{i,j}^{\mathrm{II}}|U_{i-1,j}^x|^2 + T_{i,j}^{\mathrm{II}}|U_{i,j}^x|^2 + T_{i,j}^{\mathrm{V}}|U_{i,j-1}^y|^2 + T_{i,j}^{\mathrm{V}}|U_{i,j}^y|^2\right).$$

In general, the estimates $0 < K_{\mathrm{inf}} \leq K \leq K_{\mathrm{sup}}$ may be too coarse to provide enough control on the coefficients $T_{i,j}^{\mathrm{I}}, \cdots, T_{i,j}^{\mathrm{VI}}$ for constructing a reasonable preconditioner D. Strictly speaking, the necessary level of control will be available only if we have information about the *local* variation of K on each cell $\bar{\Omega}_{i,j}$.

In practice, however, we rarely have such fine-scale knowledge of K, and even if we did we would not try to use it in calculating the Galerkin integrals $\int_\Omega K^{-1}\mathbf{u}\cdot\mathbf{v}\,dx\,dy$ exactly. Instead, most practical codes use approximate quadrature schemes that effectively treat K^{-1} as piecewise polynomial. In fact, as we suggested in §1, for sufficiently fine grids it is reasonable to treat K^{-1} as piecewise constant. In such applications, we can use the second inequality in (6.7), together with the identities $T_{i,j}^{\mathrm{II}} = T_{i,j}^{\mathrm{V}} = T_{i,j}$, to show that

$$U^H AU = \frac{1}{3}S(U) + \frac{1}{6}R(U) \leq \frac{1}{2}S(U).$$

Similarly, the first inequality in (6.7), together with the identities $T_{i,j}^{\mathrm{I}} = T_{i,j}^{\mathrm{III}} = T_{i,j}^{\mathrm{IV}} =$

$T^{\mathrm{VI}}_{i,j} = T_{i,j}$, shows that

$$\begin{aligned} U^H AU &= \frac{1}{6}S(U) + \frac{1}{6}[S(U) + R(U)] \\ &\geq \frac{1}{6}S(U) + \frac{1}{6}\sum_{i=1}^{m}\sum_{j=1}^{n} T_{i,j}\left[\left(|U^x_{i-1,j}| - |U^x_{i,j}|\right)^2 + \left(|U^y_{i,j-1}| - |U^y_{i,j}|\right)^2\right] \\ &\geq \frac{1}{6}S(U). \end{aligned}$$

In summary, $\frac{1}{6}S(U) \leq U^H AU \leq \frac{1}{2}S(U)$ whenever K is piecewise constant on the grid Δ_h.

Now consider the choice $D = \frac{2}{3}\mathrm{lump}(A)$, where

$$[\mathrm{lump}(A)]_{i,j} = \begin{cases} 0 & \text{if } i \neq j, \\ \displaystyle\sum_j A_{i,j} & \text{if } i = j. \end{cases}$$

This is the matrix that results when we add entries along each row of A and assign the sum to the diagonal entry in that row. Gonzales and Wheeler [9] use this "mass lumping" idea to improve conditioning in mixed finite-element discretizations of petroleum reservoir problems. This choice of D is also a simple instance of a preconditioner developed in [7] for other iterative schemes. It is a straightforward matter to show that, when K is piecewise constant, $U^H \mathrm{lump}(A) U = \frac{1}{2}S(U)$, so $U^H DU = \frac{1}{3}S(U)$. As a consequence,

$$b_1 = \frac{1}{2} \leq \frac{U^H AU}{U^H DU} \leq \frac{3}{2} = 2 - b_2.$$

Therefore, by Proposition 6.1, $\rho(M) \leq \frac{1}{2}$, and the iterative scheme converges with a rate independent of h and K. According to our remarks at the end of §4, we expect the ratio of error norms between successive iterates to approach $\frac{1}{2}$ as the iteration counter $k \to \infty$.

As an even simpler example, consider the choice $D = \mathrm{diag}(A)$, where

$$[\mathrm{diag}(A)]_{i,j} = \begin{cases} 0 & \text{if } i \neq j, \\ A_{i,i} & \text{if } i = j, \end{cases}$$

is the matrix A stripped of its off-diagonal entries. This choice has the attractive feature that it is trivial to compute from A. With D defined in this way, we once again find that $U^H DU = \frac{1}{3}S(U)$ when K is piecewise constant on Δ_h. Therefore, $\rho(M) \leq \frac{1}{2}$, and this iterative scheme also converges with a rate independent of h and K.

Either choice of D requires us to solve a matrix equation of the form

$$N^T D^{-1} N P^{(k)} = G^{(k-1)}$$

at each iteration. To do this, we use two cycles of a multigrid scheme in which the Jacobi iteration is the smoother, the coarse-to-fine interpolation is bilinear, and the fine-to-coarse restriction is accomplished using half-injection [4, p. 65]. This scheme preserves the h-independence of the overall algorithm's convergence rate and appears

TABLE 1
Convergence rates for various coefficients and grids.

	Grid Mesh h				
Coefficient	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 handle the variable coefficient K effectively. Alternative multigrid implementations are certainly possible here.

To test the convergence rate of Algorithm 3, we apply it to the boundary-value problems described in §5, using the preconditioner $D = \frac{2}{3}\mathrm{lump}(A)$. Table 1 shows values of the convergence rate $\bar{\mu}$ computed for each choice of coefficient K, for each of five different values of the grid mesh h. All of the tabulated values are very close to the spectral radius estimate $\rho(M) \leq \frac{1}{2}$. We conclude that this scheme converges at a rate independent of both grid mesh h and the heterogeneity reflected in the mobility coefficient K.

7. Conclusions. Poor conditioning associated with heterogeneity and fine spatial grids is a common problem. While this paper focuses on steady flows in porous media, similar equations and results apply in other fields. Two obvious applications for (1.1) arise in heat transfer, where temperature plays the role of pressure and heat flux plays the role of the Darcy velocity, and in electrostatics, where the electric potential and the electric field serve as the analogs of pressure and Darcy velocity, respectively. In either case, mixed finite-element methods can give useful approximations. However, heterogeneity, either in the thermal diffusivity or in the dielectric coefficient, can lead to poor conditioning in precisely the same way as it does for porous media. One virtue of the mixed finite-element formulation is that it permits us to attack the two sources of poor conditioning separately, exploiting multigrid ideas to reduce the sensitivity to fine grids and using spectral information associated with the material coefficient to reduce the sensitivity to heterogeneity.

Appendix: Matrix structure of the finite-element equations. The mixed finite-element equations (2.2) give rise to integral equations having the following forms. For the x-velocity equation,

$$\int_\Omega \left(K^{-1} u_h^x \, \phi_{i,j}^x - p_h \frac{\partial \phi_{i,j}^x}{\partial x} \right) dx\, dy = 0, \quad i = 0, \cdots, m, \quad j = 1, \cdots, n.$$

For the y-velocity equation,

$$\int_\Omega \left(K^{-1} u_h^y \, \phi_{i,j}^y - p_h \frac{\partial \phi_{i,j}^y}{\partial y} \right) dx\, dy = 0, \quad i = 1, \cdots, m, \quad j = 0, \cdots, n.$$

For the mass balance,

$$\int_\Omega \left(\frac{\partial u_h^x}{\partial x} + \frac{\partial u_h^y}{\partial y} - f \right) \psi_{i,j} dx\, dy = 0, \quad i = 1, \cdots, m, \quad j = 1, \cdots, n.$$

The following integrals appearing in these expressions involve no spatially varying coefficients and hence are easy to compute using the bases for $\mathbf{Q}_h$ and V_h:

$$\int_\Omega p_h \frac{\partial \phi_{i,j}^x}{\partial x} dx\, dy, \quad \int_\Omega p_h \frac{\partial \phi_{i,j}^y}{\partial y} dx\, dy, \quad \int_\Omega \frac{\partial u_h^x}{\partial x} \psi_{i,j} dx\, dy, \quad \int_\Omega \frac{\partial u_h^y}{\partial y} \psi_{i,j} dx\, dy.$$

However, the remaining integrals involve the spatially varying functions $K^{-1}(x,y)$ and $f(x,y)$. We compute these integrals using the mean value theorem for integrals [10, pp. 184–185] as follows: Since K^{-1} is bounded and integrable on each cell $\overline{\Omega}_{i,j}$, there exist numbers $T_{i,j}^{\mathrm{I}}, T_{i,j}^{\mathrm{II}}, T_{i,j}^{\mathrm{III}}$ such that

$$\int_\Omega K^{-1}\, \phi_{s,t}^x\, \phi_{i,j}^x\, dx\, dy = \begin{cases} T_{i,j}^{\mathrm{II}}/6, & t = j, \quad s = i-1; \\ (T_{i,j}^{\mathrm{I}} + T_{i+1,j}^{\mathrm{III}})/3, & t = j, \quad s = i; \\ T_{i+1,j}^{\mathrm{II}}/6, & t = j, \quad s = i+1\,. \end{cases}$$

Here, $T_{i,j}^{\mathrm{I}}/[(x_i - x_{i-1})(y_j - y_{j-1})]$ is a number lying between the upper and lower bounds of K^{-1} on the cell $\overline{\Omega}_{i,j}$, and similarly for $T_{i,j}^{\mathrm{II}}$ and $T_{i,j}^{\mathrm{III}}$. Analogous calculations show that

$$\int_\Omega K^{-1}\, \phi_{s,t}^y\, \phi_{i,j}^y\, dx\, dy = \begin{cases} T_{i,j}^{\mathrm{V}}/6, & t = j-1, \quad s = i; \\ (T_{i,j}^{\mathrm{IV}} + T_{i,j+1}^{\mathrm{VI}})/3, & t = j, \quad s = i; \\ T_{i,j+1}^{\mathrm{V}}/6, & t = j+1, \quad s = i\,. \end{cases}$$

The calculations of $\int_\Omega f\, \psi_{i,j}\, dx\, dy$ can proceed similarly.

Now let us adopt the following orderings for the vectors of unknown nodal coefficients:

$$U^x = \begin{bmatrix} U_{0,1}^x \\ \vdots \\ U_{m,1}^x \\ \vdots \\ U_{0,n}^x \\ \vdots \\ U_{m,n}^x \end{bmatrix}, \quad U^y = \begin{bmatrix} U_{1,0}^y \\ \vdots \\ U_{1,n}^y \\ \vdots \\ U_{m,0}^y \\ \vdots \\ U_{m,n}^y \end{bmatrix}, \quad P = \begin{bmatrix} P_{1,1} \\ \vdots \\ P_{m,1} \\ \vdots \\ P_{1,n} \\ \vdots \\ P_{m,n} \end{bmatrix}.$$

Then the entire algebraic system arising from (2.2) has the structure

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

Here,

$$A^x = \begin{bmatrix} A_1^x & & \\ & \ddots & \\ & & A_n^x \end{bmatrix} \in \mathbb{R}^{(m+1)n \times (m+1)n},$$

where each block $A_j^x \in \mathbb{R}^{(m+1)\times(m+1)}$ has the tridiagonal structure

$$A_j^x = \frac{1}{6}\begin{bmatrix} 2T_{1,j}^{\mathrm{III}} & T_{1,j}^{\mathrm{II}} & & & \\ T_{1,j}^{\mathrm{II}} & 2(T_{1,j}^{\mathrm{III}} + T_{2,j}^{\mathrm{I}}) & T_{2,j}^{\mathrm{II}} & & \\ & & \ddots & & \\ & & & T_{m,j}^{\mathrm{II}} & 2T_{m,j}^{\mathrm{III}} \end{bmatrix}.$$

Similarly,

$$A^y = \begin{bmatrix} A_1^y & & \\ & \ddots & \\ & & A_m^y \end{bmatrix} \in \mathbb{R}^{m(n+1)\times m(n+1)},$$

where each block $A_i^y \in \mathbb{R}^{(n+1)\times(n+1)}$ has the tridiagonal form

$$A_i^y = \frac{1}{6}\begin{bmatrix} 2T_{i,1}^{\mathrm{VI}} & T_{i,1}^{\mathrm{V}} & & & \\ T_{i,1}^{\mathrm{V}} & 2(T_{i,1}^{\mathrm{VI}} + T_{i,2}^{\mathrm{IV}}) & T_{i,2}^{\mathrm{V}} & & \\ & & \ddots & & \\ & & & T_{i,n}^{\mathrm{V}} & 2T_{i,n}^{\mathrm{VI}} \end{bmatrix}.$$

Finally, the two "difference" matrices N^x and N^y have the following structures:

$$N^x = \begin{bmatrix} N_1^x & & \\ & \ddots & \\ & & N_n^x \end{bmatrix} \in \mathbb{R}^{n(m+1)\times nm},$$

where

$$N_j^x = (y_j - y_{j-1}) \begin{bmatrix} 1 & & & & \\ -1 & 1 & & & \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ & & & & 1 \\ & & & & -1 \end{bmatrix} \in \mathbb{R}^{(m+1)\times m},$$

while

$$N^y = \begin{bmatrix} N_{1,1}^y & \cdots & N_{1,n}^y \\ \vdots & & \\ N_{m,1}^y & \cdots & N_{m,n}^y \end{bmatrix} \in \mathbb{R}^{(n+1)m\times nm},$$

where

$$N_{i,j}^{y} = (x_i - x_{i-1}) \begin{bmatrix} & \vdots & \\ \cdots & 1 & \cdots \\ \cdots & -1 & \cdots \\ & \vdots & \end{bmatrix} \begin{matrix} \\ \leftarrow \text{ row } j \\ \leftarrow \text{ row } j+1 \\ \\ \uparrow \\ \text{column } i \end{matrix} \in \mathbb{R}^{(n+1)\times m}.$$

REFERENCES

[1] M. B. ALLEN, R. E. EWING, AND J. V. KOEBBE, *Mixed finite-element methods for computing groundwater velocities*, Numer. Meth. P.D.E., 3 (1985), pp. 195–207.

[2] G. BIRKHOFF AND R. E. LYNCH, *Numerical Solution of Elliptic Problems*, Society for Industrial and Applied Mathematics, Philadelphia, 1984.

[3] A. BRANDT, *Multi-level adaptive solutions to boundary-value problems*, Math. Comp., 31 (1977), pp. 333–390.

[4] W. L. BRIGGS, *A Multigrid Tutorial*, Society for Industrial and Applied Mathematics, Philadelphia, 1987.

[5] J. DOUGLAS, R. E. EWING, AND M. F. WHEELER, *The approximation of the pressure by a mixed method in the simulation of miscible displacement*, RAIRO Anal. Numér., 17 (1983), pp. 17–33.

[6] R. E. EWING, R. D. LAZAROV, AND J. WANG, *Superconvergence of the velocities along the Gaussian lines in mixed finite element methods*, SIAM J. Numer. Anal., 28 (1991), pp. 1015–1029.

[7] R. E. EWING, R. D. LAZAROV, P. LU, AND P. S. VASSILEVSKI, *Preconditioning indefinite systems arising from the mixed finite-element discretization of second-order elliptic systems*, in Preconditioned Conjugate Gradient Methods, Lecture Notes in Mathematics 1457, O. Axelsson and L. Kolotilina, eds., Springer-Verlag, Berlin, 1990, pp. 280–343.

[8] R. E. EWING AND M. F. WHEELER, *Computational aspects of mixed finite element methods*, in Numerical Methods for Scientific Computing, R. S. Stepelman, ed., North-Holland, Amsterdam, 1983, pp. 163–172.

[9] R. GONZALES AND M. F. WHEELER, *Mixed finite element methods for petroleum reservoir engineering problems*, in Proceedings, Sixth International Conference on Computing Methods in Applied Sciences and Engineering, INRIA, Versailles, France, 1983, North-Holland, Amsterdam, 1984, pp. 639–658.

[10] M. E. MUNROE, *Introduction to Measure and Integration*, Addison-Wesley, Cambridge, MA, 1953.

[11] P. A. RAVIART AND J. M. THOMAS, *A mixed finite element method for 2nd order elliptic problems*, in Mathematical Aspects of Finite Element Methods, Lecture Notes in Mathematics 606, I. Galligani and E. Magenes, eds., Springer-Verlag, Berlin, 1977, pp. 292–315.

submitted to NMPDE Nov 91

An Iterative Finite-Element Collocation Method for Parabolic Problems using Domain Decomposition

Mark C. Curran
Sandia National Laboratories
Albuquerque, NM 87185-5800

Abstract

Advection-dominated flows occur widely in the transport of groundwater contaminants, the movements of fluids in enhanced oil recovery projects, and many other contexts. In numerical models of such flows, adaptive local grid refinement is a conceptually attractive approach for resolving the sharp fronts or layers that tend to characterize the solutions. However, this approach can be difficult to implement in practice. A domain decomposition method developed by Bramble, Ewing, Pasciak, and Schatz, known as the BEPS method, overcomes many of the difficulties. We demonstrate the applicability of the iterative BEPS ideas to finite-element collocation on trial spaces of piecewise Hermite bicubics. The resulting scheme allows one to refine selected parts of a spatial grid without destroying algebraic efficiencies associated with the original coarse grid. We apply the method to two dimensional time-dependent advection-diffusion problems.

1. Introduction

The purpose of this paper is to investigate a numerical scheme for solving highly advective fluid-flow problems. The difficulty with these problems is that they tend to form transient, localized regions where the solution exhibits rapid variation. These regions are typically called shocks or fronts; they may persist through much of the time domain.

Tracking the location of fronts and numerically resolving the solution near fronts are two important aspects of the problem. Tracking the front requires knowledge of the spatial rate of change of the solution, or the gradient. This suggests the use of a numerical scheme which also solves for the gradient of the solution. Better numerical resolution near fronts involves increasing the number of degrees of freedom within and near the front. This suggests a scheme which can adaptively introduce additional unknowns where necessary.

First, to obtain high-order spatial accuracy while solving for the gradient as well as the solution, finite-element collocation on piecewise Hermite cubics

is used. Second, an adaptive grid refinement scheme is implemented which decouples the regions possessing added degrees of freedom from the original, unrefined problem by domain-decomposition techniques. This preserves the efficient structure of the original problem and provides a method for transfer of information between regions containing fronts and outlying areas where the desired solution is more regular.

The actual implementation of a local grid refinement scheme in two or more dimensions creates another problem. Heuristically, the idea of adding extra degrees of freedom only where needed is an intriguing one. Numerically however, it is quite difficult to administer. This is because the use of additional degrees of freedom implies additional unknowns in the numerical system and hence additional equations to be solved. The relationship between these unknowns and the equations in which they appear can severely disrupt the linear algebraic structure of the original coarse-grid system, for which we frequently have very efficient solution techniques. In this paper, alternating-direction schemes are the solvers of choice. Local grid refinement would disrupt the structure such that alternating direction methods could not be used. This is why domain decomposition techniques will be employed to decouple regions of refinement from the original system. A block Gauss-Siedel-like algorithm is incorporated to transfer information between regions of refinement and the original coarse-grid system. This approach allows the use of a very efficient matrix solver on both the original system and the refined systems, despite the algebraic disruption associated with the added degrees of freedom.

To adaptively refine a grid there must first be some criteria for choosing h locally. In the advection-diffusion equation, h can be chosen to keep the grid Peclet number $\mathcal{P} = \mathbf{vL/D}$ near sharp fronts below some tolerance to eliminate spurious oscillations in the numerical solution. Typically, when adaptive refinement schemes have been proposed, a very complex data structure is needed to keep track of the evolving nature of the numerical solution from time step to time step due to the relocation of numerical degrees of freedom. Thus an important aspect of the grid refinement problem is the development of computationally efficient algorithms to implement the refinement scheme. BEPS-like methods hold continuing promise in this regard.

This paper is limited in scope primarily to the development and coding of computationally efficient algorithms. The gridding criteria to be used are heuristically motivated and largely based on polynomial approximation theory. Problems considered do not involve highly complex frontal interactions. In complex applications, the use of a variation of the BEPS method called overlapping domain decomposition is foreseen, as developed in [5] for Galerkin finite elements.

The new contributions are in Sections 3-5, which discuss domain decomposition, numerical results for grid refinement in two dimensions, and computationally obtained convergence rates respectively. The main novel idea is the extension of a BEPS-like preconditioned domain decomposition technique to finite-element collocation. This extension is most profitable in two space dimensions (see [1] for one dimensional case), where it allows us to use an alternating-direction solution procedure in spite of the irregular geometry of the locally refined grids. The BEPS method was first developed for Galerkin finite-element formulations and later was applied to cell-centered finite differences. This paper presents the first application of the BEPS ideas to collocation on Hermite cubics.

The organization of this paper is as follows: in Section 2, alternating-direction collocation (ADC) is introduced and discussed. This is an operator-

splitting technique which converts a time-dependent, multidimensional problem into a set of one-dimensional problems. In Section 3 a new approach for local grid refinement using a BEPS-like preconditioned iterative method applied to multidimensional problems is discussed. This method uses alternating-direction techniques to solve equations involving coarse- and fine-grid matrices. Thus the domain-decomposition approach allows us to achieve substantial computational efficiencies on both the coarse-grid and fine-grid problems, whereas no such efficiencies seem to be available for the composite problem. Section 4 discusses some numerical experiments of the method and Section 5 presents computationally observed convergence rates and error estimates. Conclusions are discussed in Section 6.

2. Alternating-Direction Collocation

Alternating-direction (AD) methods have been of interest in the numerical solution of partial differential equations since their introduction in 1955 by Peaceman and Rachford [20]. In this paper, the amenability of ADC to implementation on parallel-architecture computers will be noted to some degree. In 1970 Douglas and Dupont [12] developed an alternating-direction Galerkin method, variants of which have attracted the attention of several authors, including Dendy and Fairweather [13] and Hayes and Krishnamachari [18]. Analogous alternating-direction collocation methods have also appeared in several papers, including those by Bangia et al. [3], Chang and Finlayson [10], Hayes [17], Celia et al. [8], Celia [7], and Celia and Pinder [9]. Another approach similar to ADC, pioneered by Guarnaccia [16], involves an iterative technique which can be used to solve problems involving a tensorial form for diffusion since it does not corrupt the cross derivative terms, as happens in standard ADC formulations. This paper does not consider tensor diffusion.

We examine Celia's ADC for the two-dimensional advection-diffusion equation for solute transport in a known velocity field. Of interest here are algorithmic features of ADC that enhance its efficiency in comparison with standard two-dimensional collocation.

The aim of ADC is to modify the ordinary two-dimensional collocation procedure via an operator splitting. This splitting reduces the discrete problem to one involving a sequence of matrix equations, each of which has the same sparse structure as the one-dimensional collocation system. The following description of this splitting approach is essentially a review of the development presented by Celia and Pinder in [9].

We begin the discussion by first presenting the advection-diffusion equation for two space dimensions and discretizing in time using a variably impicit finite-difference approximation. The advection-diffusion equation is

$$\frac{\partial u}{\partial t} + \mathcal{L}_x u + \mathcal{L}_y u = 0, \tag{1}$$

where

$$\mathcal{L}_x = v_x \partial/\partial x - \partial(D\partial/\partial x)/\partial x,$$

and similarly,

$$\mathcal{L}_y = v_y \partial/\partial y - \partial(D\partial/\partial y)/\partial y.$$

Here, $\mathbf{v} = (v_x, v_y)$ is a velocity field which we assume is known, and D is a diffusion coefficient. We formulate this problem on some space-time domain $\Omega \times J$, where $\Omega = (a, b) \times (c, d)$ is a rectangle in the x, y plane and $J = (t_0, T)$ is

an interval of time. For existence and uniqueness of solutions we need to specify an initial condition at time $t = t_0$ and some type of boundary condition on $\partial\Omega$ for u and the derivatives of u, such as $u = 0$ for all $(x, y) \in \partial\Omega$.

We form a semidiscrete version of Equation 1 by employing a variably weighted, implicit finite-difference approximation in time:

$$u^{n+1} - u^n + \Delta t(\mathcal{L}_x + \mathcal{L}_y)u^{n+\theta} = 0. \tag{2}$$

Here, Δt is the time step and θ is the time stepping parameter with $u^{n+\theta}$ given by $u^{n+\theta} = (1-\theta)u^n + \theta u^{n+1}$. With $\theta = 0.5$ we obtain the Crank-Nicolson method which has truncation error which is $\mathcal{O}((\Delta t)^2)$.

Next, we perturb Equation 2 by a term that is $\mathcal{O}((\Delta t)^3)$ (and hence preserves consistency and stability of the approximation) to get

$$u^{n+1} - u^n + \Delta t(\mathcal{L}_x + \mathcal{L}_y)u^{n+\theta} + \Delta t^2\theta^2(\mathcal{L}_x\mathcal{L}_y)(u^{n+1} - u^n) = 0. \tag{3}$$

(Reference [9] treats the advection-diffusion equation in a slightly different fashion, splitting only the diffusive part of the spatial operator.) Rearranging Equation (3) and factoring gives

$$(1 + \Delta t\theta\mathcal{L}_y)(1 + \Delta t\theta\mathcal{L}_x)(u^{n+1} - u^n) = -\Delta t(\mathcal{L}_x + \mathcal{L}_y)u^n.$$

Conceptually, we can solve $(1+\Delta t\theta\mathcal{L}_y)z = -\Delta t(\mathcal{L}_x+\mathcal{L}_y)u^n$ for the intermediate unknown z, then solve $(1 + \Delta t\theta\mathcal{L}_x)(u^{n+1} - u^n) = z$ for the time increment in $\hat{u} = u^{n+1} - u^n$.

To see how this works algebraically, notice that substituting Hermite bicubic trial functions for $\hat{u}$ and collocating produces a matrix equation $\mathbf{K}\mathbf{u}^{n+1} = \mathbf{r}^n$, where $\mathbf{u}^{n+1}$ is the vector of time increments for the unknown nodal coefficients of $\hat{u}^{n+1}$. Consider a typical entry of the matrix $\mathbf{K}$:

$$\left\{\left[1 + \Delta t\theta(\mathcal{L}_x + \mathcal{L}_y) + \Delta t^2\theta^2(\mathcal{L}_x\mathcal{L}_y)\right] H_{ij\ell m}\right\}(\bar{x}_p, \bar{y}_q), \tag{4}$$

where $H_{ij\ell m}$ is some basis function in the tensor-product interpolation space and $(\bar{x}_p, \bar{y}_q)$ are the coordinates of the collocation points. Each $H_{ij\ell m}(x, y) = H_{i\ell}(x)H_{jm}(y)$, so we can expand the expression (4) and factor it to get

$$[H_{i\ell}(\bar{x}_p) + \Delta t\theta(\mathcal{L}_x H_{i\ell})(\bar{x}_p)] \cdot [H_{jm}(\bar{y}_q) + \Delta t\theta(\mathcal{L}_y H_{jm})(\bar{y}_q)].$$

This factoring of each matrix entry, together with Celia's scheme for numbering and renumbering equations and unknowns, allows us to factor the entire matrix equation at each time level. If we number the equations and unknowns "vertically," that is, lexicographically along the lines $x = \bar{x}_p$, as shown in Figure 1a, then the $4N_xN_y \times 4N_xN_y$ matrix $\mathbf{K}$ factors as follows:

$$\mathbf{K} = \mathbf{Y}\mathbf{X} = \begin{bmatrix} \mathbf{Y}_{1,1} & & \\ & \ddots & \\ & & \mathbf{Y}_{2N_x,2N_x} \end{bmatrix} \begin{bmatrix} \mathbf{X}_{1,1} & \cdots & \mathbf{X}_{1,2N_y} \\ \vdots & & \vdots \\ \mathbf{X}_{2N_y,1} & \cdots & \mathbf{X}_{2N_y,2N_y} \end{bmatrix}.$$

Each $2N_y \times 2N_y$ block $\mathbf{Y}_{p,p}$ has the five-band structure of a one-dimensional collocation matrix. Moreover, The entries in $\mathbf{Y}_{p,p}$ depend only on the y-coordinates of collocation points.

Now consider the matrix $\mathbf{X}$. If we switch to the "horizontal" numbering scheme for equations and unknowns, illustrated in Figure 1b, then $\mathbf{X}$ transforms to a block-diagonal matrix that we denote as follows:

$$\mathbf{X}^* = \begin{bmatrix} \mathbf{X}^*_{1,1} & & \\ & \ddots & \\ & & \mathbf{X}^*_{2N_y,2N_y} \end{bmatrix}.$$

(The superscript * is used to indicate the result of switching to the "horizontal" numbering scheme. In computational practice, the renumbering can be accomplished without any computation by using double integer indices for grid variables and switching the order of the DO loops over the indices.) Again, each $2N_x \times 2N_x$ block $\mathbf{X}^*_{q,q}$ has the five-band structure. It should be noted here that the usual ADI scheme is an iterative process but here we are solving the equations exactly at each step.

In light of these observations, we can solve the two-dimensional matrix equation $\mathbf{K}\mathbf{u}^{n+1} = \mathbf{r}^n$ by the following procedure.

1. Adopt the "vertical" numbering scheme, and solve $\mathbf{Y}\mathbf{z} = \mathbf{r}^n$ for the intermediate vector $\mathbf{z}$ by solving the independent problems $\mathbf{Y}_{p,p}\mathbf{z}_p = \mathbf{r}^n_p$, $p = 1, \ldots, 2N_x$.

2. Renumber according to the "horizontal" scheme, converting $\mathbf{z}$ to the reordered vector $\mathbf{z}^*$. This renumbering, accomplished by simple DO loop inversion, transforms $\mathbf{X}$ to the block-diagonal form $\mathbf{X}^*$.

3. Solve $\mathbf{X}^*\mathbf{u}^{n+1} = \mathbf{z}^*$ for the desired time increments by solving the independent systems $\mathbf{X}^*_{q,q}\mathbf{u}^{n+1}_q = \mathbf{z}^*_q$, $q = 1, \ldots, 2N_y$.

Thus each time step involves the solution of matrix equations that are at worst one-dimensional in structure.

To examine the efficiency to be gained by the splitting scheme, let us assume that $N_x = N_y = N$. In the fully two-dimensional matrix problem $\mathbf{K}\mathbf{u}^{n+1} = \mathbf{r}^n$, there are then $4N^2$ unknowns, and the matrix $\mathbf{K}$ is asymmetric. If we order equations and unknowns to allow for row reduction without pivoting, $\mathbf{K}$ has a bandwidth $B_2 = 8N + 16$ (see Frind and Pinder, [15]). Assuming that row reduction accounts for the bulk of the computational work in the sparse matrix solver used, we expect the operation count for solving the fully two-dimensional equations at each time step to be roughly $4N^2B_2^2 = 256N^4$ for large N. By contrast, ADC calls for the solution of $4N$ matrix equations of bandwidth $B_1 = 5$ and order $2N$ at each time level. Thus an upper bound for the number of arithmetic operations required in the row reductions for ADC is $4N(2NB_1^2) = 200N^2$. By writing a row reduction scheme tailored to the zero structure of one-dimensional collocation matrices, one can reduce this operation count somewhat.

Furthermore, each of the "one-dimensional" systems in step 1 of ADC is independent of any other. Therefore the solution of these systems can be done

concurrently. Step 3 can also be completed in parallel for the same reason. The inherent parallelism of ADC is explored in [2].

3. Domain Decomposition for Two-Dimensional Collocation

In this section we develop the domain decomposition techniques needed to apply an adaptive gridding scheme in two space dimensions. We extend the ideas of [1] and use the alternating direction methods of Section 2 for the matrix solvers.

We begin by defining the necessary function spaces needed for the domain decomposition. We also describe the various sets of collocation points used. First consider the rectangular spatial domain Ω with associated regular coarse grid $\Delta_0 = \Delta_x \times \Delta_y$ where $\Delta_x = \{a = x_0 < x_1 < \cdots < x_{N_x} = b\}$ and $\Delta_y = \{c = y_0 < y_1 < \cdots < y_{N_y} = d\}$. Define the space of Hermite bicubics on Δ_0 as:

$$\mathcal{M}(\Delta_0) = \{f \in C^1(\Omega) : f \text{ is bicubic on each element formed by } \Delta_0\}.$$

We also define a set of collocation points for this grid to be the set of points whose coordinates are the two-point Gauss quadrature abscissae for each of the intervals in Δ_x and Δ_y. Thus we have a set K_0 of $4(N_x - 1)(N_y - 1)$ collocation points in Ω.

The spatial domain Ω is decomposed into two disjoint subdomains Ω_1 and Ω_2, where the boundary between Ω_1 and Ω_2 lies strictly along coarse-grid lines in Δ_0. For ease of discussion we will assume Ω_2 to be a rectangular region, say $\Omega_2 = (x_I, x_J) \times (y_K, y_L)$, where $0 \leq I < J \leq N_x$ and $0 \leq K < L \leq N_y$. A typical spatial domain decomposed into two subdomains, along with computational grid, is pictured in Figure 2.

Now consider solving the advection-diffusion problem (2) on the computational domain in Figure 2. Suppose the solution has some type of local behavior in Ω_2, such as steep gradients or shock-like fronts, which cannot be accurately resolved without grid refinement. Instead of globally refining the entire domain Ω we refine only locally in the subdomain Ω_2 by adding nodes in the x-direction and in the y-direction in a uniform way, so that we now have N_x^r elements in the x-direction and N_y^r elements in the y-direction in Ω_2. The composite grid formed in this manner is shown in Figure 3. This new fine grid in Ω_2 is denoted by Δ_2. The composite grid is $\Delta = \Delta_0 \cup \Delta_2$, and the portion of the original coarse grid that lies in the unrefined region Ω_1 is denoted by Δ_1.

The space of Hermite bicubics in which we seek a solution is defined on the composite grid Δ as follows:

$$\mathcal{M}(\Delta) = \{f \in C^1(\Omega) : f \text{ is bicubic on every element formed by } \Delta\}.$$

It is of interest to note that the previously defined space $\mathcal{M}(\Delta_0)$ is a subspace of $\mathcal{M}(\Delta)$. We also need two other subspaces of $\mathcal{M}(\Delta)$. The first is the subspace of $\mathcal{M}(\Delta)$ containing only those functions whose support lies entirely in Ω_2. This space is denoted by $\mathcal{M}_0(\Delta_2)$ and is defined as follows:

$$\mathcal{M}_0(\Delta_2) = \{f \in C_0^1(\Omega_2) : f \text{ is bicubic on each element formed by } \Delta_2\}.$$

It is important to restate that the functions in $\mathcal{M}_0(\Delta_2)$ are identically zero on the boundary of Ω_2. Δ_2 contains $(N_x^r - 1)(N_y^r - 1)$ interior nodes called

refinement nodes or fine grid nodes. Note that Hermite bicubics have four degrees of freedom at each node. Therefore there are $4(N_x^r-1)(N_y^r-1)$ degrees of freedom for functions in $\mathcal{M}_0(\Delta_2)$, since they are already completely determined on the boundary. Therefore there are $4(N_x^r-1)(N_y^r-1)$ basis functions for the space $\mathcal{M}_0(\Delta_2)$. They are denoted by H_{ij00}^r, H_{ij10}^r, H_{ij01}^r, and H_{ij11}^r, where $i=1,2,...,N_x^r-1$ and $j=1,2,...,N_y^r-1$.

The final subspace of $\mathcal{M}(\Delta)$ is also a subspace of $\mathcal{M}(\Delta_0)$ and is denoted by $\mathcal{M}_0(\Delta_0)$. This last space contains the functions of $\mathcal{M}(\Delta_0)$ whose supports are contained in the union of Ω_1 with the adjacent layer of coarse-grid elements in Ω_2. These functions are identically zero at all the original coarse-grid nodes that now lie in the interior of Ω_2. The basis for $\mathcal{M}_0(\Delta_0)$ is the subset of the basis for $\mathcal{M}(\Delta_0)$ containing only those basis functions centered at nodes in Δ_1. In fact, $\mathcal{M}_0(\Delta_0)$ is such that any function $f \in \mathcal{M}(\Delta)$ can be written as a unique linear combination of functions f_1 and f_2, where $f_1 \in \mathcal{M}_0(\Delta_0)$ and $f_2 \in \mathcal{M}_0(\Delta_2)$. Furthermore, the basis for $\mathcal{M}(\Delta)$ is the union of the bases for $\mathcal{M}_0(\Delta_2)$ and $\mathcal{M}_0(\Delta_0)$.

In summary, we have defined four function spaces. The principal space $\mathcal{M}(\Delta)$ is the space of Hermite bicubics on the composite grid Δ. $\mathcal{M}(\Delta_0)$ is the subspace of $\mathcal{M}(\Delta)$ containing the functions that are Hermite bicubic on the original coarse grid Δ_0. $\mathcal{M}_0(\Delta_0)$ is the subspace of $\mathcal{M}(\Delta_0)$ containing functions which are identically zero at all original coarse-grid nodes in $\Delta_0 \backslash \Delta_1$. Finally, $\mathcal{M}_0(\Delta_2)$ is the subspace of $\mathcal{M}(\Delta)$ containing functions whose support lies entirely in Ω_2.

Next we need to describe the various sets of collocation points associated with these function spaces. We have already mentioned K_0, which is the collection of 2×2 Gauss points associated with Δ_0. The subset of K_0 of points which lie in Ω_1 is denoted by K_1. We now consider the collection of collocation points for the fine grid Δ_2. The coordinates of these points are just the coordinates of the two-point Gauss quadrature absciscae for the intervals $[x_{i-1}^r, x_i^r]$ and $[y_{j-1}^r, y_j^r]$, where $i=1,2,...,N_x^r$, and $j=1,2,...,N_y^r$. This is a set of $4N_x^rN_y^r$ points, which is $4(N_x^r+N_y^r-1)$ points too many to completely determine a function in $\mathcal{M}_0(\Delta_2)$. Let K_r be the subset obtained by removing the points nearest to $\partial\Omega_2$. That outer ring of collocation points is precisely $4(N_x^r+N_y^r-1)$ in number, which suffices to correct the surplus just mentioned. Justification for deleting these points is found by noting that functions in $\mathcal{M}_0(\Delta_2)$ are already completely determined on $\partial\Omega_2$, and hence collocation points nearest $\partial\Omega_2$ are superfluous. Another subset needed are the points in that outer ring which are nearest to coarse-grid points in Δ_1 lying on the boundary between Ω_1 and Ω_2. We denote this set of collocation points by K_∂^*.

Two sets of collocation points which are only needed to facilitate the discussion of the solution procedure are a separation of original coarse-grid collocation points from K_0 which lie in Ω_2. The first subset contains those points which lie in Ω_2, except the outer ring of points nearest to $\partial\Omega_2$. This set is denoted K_2, with the remainder of the coarse-grid collocation points in Ω_2 being designated K_∂, since they are located just inside the boundary of Ω_2. Note that there is a natural one-to-one correspondence between points in K_∂ and points in K_∂^*. We use this correspondence in the next section

Finally, the set of collocation points for the composite grid is a union of three of the sets previously defined, namely $K = K_1 \cup K_\partial^* \cup K_r$. Figure 4 depicts the

collocation points K for the composite grid as well as the set K_1, K_∂^*, K_r, K_2, and K_∂. One way of describing K is by associating with each node of the composite grid Δ four collocation points, one in each of the four elements for which the node is a vertex.

One final note: The fine-grid nodes that appear to be added along the boundary of Ω_2 are not associated with unknown degrees of freedom but are necessarily slave nodes to preserve the continuous differentiability of functions in $\mathcal{M}(\Delta)$. In other words, the values of u and its derivatives are determined at these nodes by interpolation from the nearest coarse-grid nodes.

To motivate and describe the BEPS-like preconditioner, we consider a certain decomposition of functions in $\mathcal{M}(\Delta)$. If we wish to solve an operator equation of the form $\mathcal{L}u = 0$ on Δ, then we decompose $u \in \mathcal{M}(\Delta)$ as $u = u_r + u_1$, where $u_r \in \mathcal{M}_0(\Delta_2)$ and u_1 satisfies

$$\mathcal{L}u_1(\overline{\mathbf{x}}_k) = 0 \text{ for all } \overline{\mathbf{x}}_k \in K_r,$$

$$u_1 = 0 \text{ on } \partial\Omega,$$

and

$$\mathcal{L}\hat{u}_r(\overline{\mathbf{x}}) = f(\overline{\mathbf{x}}), \quad \text{for each } \overline{\mathbf{x}} \in K_r.$$

This problem uniquely determines u_1 on $\overline{\Omega}_2$. In Ω_1, the function u_1 is identically equal to u. Thus, this decomposition is unique for functions in $\mathcal{M}(\Delta)$.

We consider as our differential operator the fully implicit, temporally discrete advection-diffusion model. Throughout this section, we use the notation

$$\mathcal{L}u^{n+1} = u^{n+1} - u^n + k\mathbf{v} \cdot \nabla u^{n+1} - k\nabla(D\nabla u^{n+1}).$$

One important aspect of the BEPS iteration is that it utilizes the original coarse grid operator to invert the composite operator. Communication between the coarse- and fine-grid problems is achieved through manipulation of the right-hand side vectors in the matrix solution processes. The end result is a technique which can utilize efficient solvers for both the coarse-grid and fine-grid problems with very little rewriting of code.

The problem we wish to solve is a finite-element collocation approximation to

$$\mathcal{L}u^{n+1}(x,y) = f(x,y), \quad \forall\,(x,y) \in \Omega.$$

We assume the boundary data

$$u^{n+1}(x,y,t) = 0, \quad \forall\,(x,y) \in \partial\Omega, \ \ t \in (t_0,T),$$

and some initial condition:

$$u^0(x,y,t_0) = u_0(x,y), \quad \forall\,(x,y) \in \Omega.$$

Here, u_0 is the interpolant of the initial function in the space $\mathcal{M}(\Delta)$. We search for solutions in the space $\mathcal{M}(\Delta)$ such that the residual $\mathcal{L}u^{n+1}(\bar{\mathbf{x}}_k) - f(\bar{\mathbf{x}}_k)$ is equal to zero at all collocation points $\bar{\mathbf{x}}_k \in K$. Recall that the basis for $\mathcal{M}(\Delta)$ is the union of the bases for $\mathcal{M}_0(\Delta_2)$ and $\mathcal{M}_0(\Delta_0)$, and as such it is not a nodal basis. In particular, basis functions for $\mathcal{M}_0(\Delta_1)$ that are centered at nodes on $\partial\Omega_2$ can have nonzero values over all of the fine-grid elements in the first row of coarse-grid elements in Ω_2. Assembling the matrix for this system can

be an arduous task, since equations at those fine-grid collocation points will not only involve the sixteen fine-grid basis functions that have support on that particular fine-grid element but could also involve as many as sixteen coarse-grid basis functions. Also, the composite system does not lend itself to solution by alternating-direction collocation, because the irregular composite mesh prevents us from factoring the composite matrix. So not only is the composite matrix going to be difficult to construct, but also it will have to be done for a full two-dimensional collocation system.

On the other hand, the BEPS method for finite-element collocation separates the coarse- and fine-grid problems and uses the original coarse-grid matrix when solving the coarse-grid part of the composite-grid problem. Hence, alternating-direction methods can still be used to solve the coarse-grid problem. Similarly, the fine-grid problem involves only fine-grid Hermite basis functions and fine-grid collocation points in the matrix formation, so that alternating-direction collocation can be used to solve the fine-grid problem as well. Thus we completely avoid having to form a composite matrix for a full two-dimensional collocation system.

The following is a description, in operator notation, of our BEPS-type method for finite-element collocation. The numerical scheme for solving the composite system is in some respects similar to the block Gauss-Seidel (BGS) method. It differs in that, during each iteration, a fine, coarse and then another fine-grid solution is computed, as opposed to BGS, in which one would compute just a fine- and a coarse-grid solution. The benefit of the BEPS-like methods over BGS are twofold. First, there is the advantage of more efficient matrix formulation, discussed in the last paragraph. Also, with the added fine-grid iteration and the reorganization of collocation points in the coarse-grid solve, BEPS methods tend to converge at a substantially faster rate than do BGS methods.

The solution procedure at each time step begins with a three-step initialization. This initialization starts by computing an initial solution involving the operator associated with the fine grid:

$$\mathcal{L}\hat{u}_r^0(\overline{\mathbf{x}}) = f(\overline{\mathbf{x}}), \qquad \text{for each } \overline{\mathbf{x}} \in K_r,$$

where $\hat{u}_r^0 \in \mathcal{M}_0(\Delta_2)$. The next step in the initialization involves the operator associated with the coarse grid, using a right-hand side modified by the new fine-grid information. Here, instead of using the composite operator, the original coarse-grid operator is used. However, the right-hand side is evaluated as it would be in the composite system. Thus we solve for $\tilde{u}_c^0 \in \mathcal{M}(\Delta_0)$ such that

$$\begin{array}{lll} \mathcal{L}\tilde{u}_c^0(\overline{\mathbf{x}}) & = f(\overline{\mathbf{x}}), & \forall \overline{\mathbf{x}} \in K_1, \\ \mathcal{L}\tilde{u}_c^0(\hat{\mathbf{x}}) & = f(\overline{\mathbf{x}}) - \mathcal{L}\hat{u}_r^0(\overline{\mathbf{x}}), & \forall \hat{\mathbf{x}} \in K_\partial, \\ & & \quad \text{and each corresponding } \overline{\mathbf{x}} \in K_\partial^*, \\ \mathcal{L}\tilde{u}_c^0(\overline{\mathbf{x}}) & = 0, & \forall \overline{\mathbf{x}} \in K_2, \end{array} \tag{5}$$

where $\hat{\mathbf{x}} \in K_\partial$ is the coarse-grid collocation point associated with the fine-grid collocation point $\overline{\mathbf{x}} \in K_\partial^*$ under the one-to-one correspondence described in the previous section. So indeed Equations (5) involve the coarse-grid operator, since $\tilde{u}_c^0$ is a coarse-grid Hermite and all of the function evaluations on the left side of (5) are made at coarse-grid collocation points. Once $\tilde{u}_c^0 \in \mathcal{M}(\Delta_0)$ is known, we form the restriction of $\tilde{u}_c^0$ to the subspace $\mathcal{M}_0(\Delta_0)$, denoted by $\hat{u}_c^0$.

The third step of the initialization is another solution involving the fine-grid operator to adjust the initial solution on the fine-grid. We solve for $\hat{u}_f^0 \in \mathcal{M}_0(\Delta_2)$ such that

$$\mathcal{L}\hat{u}_f^0(\overline{\mathbf{x}}) = -\mathcal{L}\hat{u}_c^0(\overline{\mathbf{x}}) \qquad \forall \overline{\mathbf{x}} \in K_r.$$

Thus the initial composite solution is formed by summing the parts $\hat{u}^0 = \hat{u}_r^0 + \hat{u}_1^0$, where $\hat{u}_1^0 = \hat{u}_c^0 + \hat{u}_f^0$. This completes the initialization of the unknown $\hat{u}$ at the new time level.

Next, the residual g is computed and an iteration procedure defined below is used to reduce $\|g\|_\infty$ to some predetermined tolerance. At any iteration level m, the residual g is evaluated at the collocation points of the composite system as follows:

$$\begin{aligned} g^m(\overline{\mathbf{x}}) &= f(\overline{\mathbf{x}}) - \mathcal{L}\hat{u}_c^m(\overline{\mathbf{x}}), && \forall \overline{\mathbf{x}} \in K_1 \\ g^m(\overline{\mathbf{x}}) &= f(\overline{\mathbf{x}}) - \mathcal{L}\hat{u}_c^m(\overline{\mathbf{x}}) - \mathcal{L}\hat{u}_r^m(\overline{\mathbf{x}}) - \mathcal{L}\hat{u}_f^m(\overline{\mathbf{x}}), && \forall \overline{\mathbf{x}} \in K_\partial^* \cup K_r. \end{aligned} \tag{6}$$

Here we are using the perturbed time-discrete or alternating-direction operator to evaluate the residual. Another possiblity suggested by Ewing [14], which increases the convergence rate of the iteration for Galerkin finite element approaches, is to use the actual unperturbed composite operator for collocation in evaluating the residual after each iteration. Thus, as the residual is driven to zero in the iteration, the ADC splitting error would also be driven to zero. We have not pursued this idea computationally. If exact methods are used to solve the matrix equations at each step, then the only nonzero values of the residual appearing in this list occur at points in K_∂^*. If the residual is small enough in norm, the iteration procedure is halted and the solution for the new time level is saved.

Otherwise, with the residual now known, the following sequence of steps is repeated, solving for an iterative correction $\hat{w}^{m+1} \in \mathcal{M}(\Delta)$, until the solution has converged. The first step involves the fine-grid operator: Find $\hat{w}_r^{m+1} \in \mathcal{M}_0(\Delta_2)$ such that

$$\mathcal{L}\hat{w}_r^{m+1}(\overline{\mathbf{x}}) = g^m(\overline{\mathbf{x}}), \quad \forall \overline{\mathbf{x}} \in K_r,$$

The next step involves the coarse-grid operator with the residual modified by the new fine-grid information. Here again, instead of using the composite operator, the original coarse-grid operator is used. Thus we solve for $\tilde{w}_c^{m+1} \in \mathcal{M}(\Delta_0)$ such that:

$$\begin{aligned} \mathcal{L}\tilde{w}_c^{m+1}(\overline{\mathbf{x}}) &= g^m(\overline{\mathbf{x}}), && \forall \overline{\mathbf{x}} \in K_1, \\ \mathcal{L}\tilde{w}_c^{m+1}(\hat{\mathbf{x}}) &= g^m(\overline{\mathbf{x}}) - \mathcal{L}\hat{w}_r^{m+1}(\overline{\mathbf{x}}), && \forall \hat{\mathbf{x}} \in K_\partial, \\ & && \text{and each corresponding } \overline{\mathbf{x}} \in K_\partial^*, \\ \mathcal{L}\tilde{w}_c^{m+1}(\overline{\mathbf{x}}) &= 0, && \forall \overline{\mathbf{x}} \in K_2. \end{aligned} \tag{7}$$

Once $\tilde{w}_c^{m+1} \in \mathcal{M}(\Delta_0)$ is known we form the restriction of $\tilde{w}_c^{m+1}$ to the subspace $\mathcal{M}_0(\Delta_0)$, denoted by $\hat{w}_c^{m+1}$.

The last step of the iteration to reduce the residual is another solution, involving the fine-grid operator, to correct the solution on the fine grid Δ_2. We solve for $\hat{w}_f^{m+1} \in \mathcal{M}_0(\Delta_2)$ such that

$$\mathcal{L}\hat{w}_f^{m+1}(\overline{\mathbf{x}}) = -\mathcal{L}\hat{w}_c^{m+1}(\overline{\mathbf{x}}), \qquad \forall \overline{\mathbf{x}} \in K_r. \tag{8}$$

Finally, with the composite iterative correction $\hat{w}^{m+1} = \hat{w}_c^{m+1} + \hat{w}_f^{m+1} + \hat{w}_r^{m+1}$ completely determined, we update the solution by adding $\hat{w}^{m+1}$ to $\hat{u}^m$ to get

$$\hat{u}_{m+1} = (\hat{u}_c^m + \hat{w}_c^{m+1}) + (\hat{u}_f^m + \hat{w}_f^{m+1}) + (\hat{u}_r^m + \hat{w}_r^{m+1})$$

and return to the step where the residual is evaluated.

4. Local Refinement for the Rotating Plume Problem

The rotating plume problem is a purely advective one in which the velocity is known a priori in the domain $(-1,1) \times (-1,1)$. Since diffusion is not present, the profile of the solution at any given time will be a translation of the initial profile.

Example 1

In our first example of the rotating plume problem, the centroid of the plume is located at $(x_0, y_0) = (0.0, -0.4)$ and the standard deviation of the Gauss hill is $\sigma = 0.066$. The coarse grid has mesh size $\Delta x^c = \Delta y^c = 0.2$ with the fine grid having a mesh size of $\Delta x^f = \Delta y^f = 0.05$. Thus, this is an example of a 4×4 refinement. The refinement strategy used here is purely problem dependent in that the element containing the peak of the Gauss hill is refined along with all eight adjacent elements. Therefore we have a 12×12 fine-grid patch contained in a 3×3 coarse-grid element patch. Figure 5 shows the contours for the numerical solution at times $t = 0.0$, 0.2, 0.4, 0.6 and 0.8. This numerical solution was computed with an iteration parameter $\gamma = 1.0$ using Crank-Nicolson time stepping with $\Delta t = 0.0005$. As in the one-dimensional case, one can search for different values of the iteration parameter γ to obtain faster convergence. We discuss this idea briefly in the next section.

Example 2

To demonstrate the efficiency of the grid refinement scheme we examine a very large problem using globally fine grids with alternating-direction collocation and a locally refined problem using our new technique. In this example of the rotating plume, the same initial data as in our first example is used. The globally fine mesh size is $\Delta x = \Delta y = 0.01$, and the time step is $\Delta t = .0005$. Thus we are solving for 160000 unknowns per time step on 2000 time steps. This solution procedure thus takes approximately 8×10^6 operations per time step. It is interesting to note if full two-dimensional collocation were used instead of alternating-direction collocation the solution at each time step would involve on the order of 4×10^{11} operations. We compare this problem with a locally refined version with $\Delta x^c = \Delta y^c = 0.1$ and $\Delta x^r = \Delta y^r = 0.01$. Thus we are using 10×10 refinement in each coarse-grid element on a 20×20-element coarse grid. We refine a 4×4 patch of the coarse grid. At each iteration of the algorithm, we are solving one fine-grid problem on 40×40-element grids with 6400 unknowns. Thus, each fine-grid solution takes approximately 3.2×10^5 operations. Also, one coarse-grid problem on a 20×20-element grid involving 1600 unknowns is solved. This coarse-grid solution requires on the order of 8×10^4 operations. Therefore, each iteration involves a total of 8000 unknowns which requires approximately 4×10^5 operations to compute. Hence, each iteration of the local refinement algorithm (typically 3-7 are necessary) requires five percent of the computational effort needed for a complete global fine-grid solution.

Table 1: Discrete $\mathcal{L}^\infty$ norm of residual with $\gamma = 0.8$ at time $t = 1.0$.

m	$\|g^m\|_\infty$	$\ln(\|g^m\|_\infty)$	slope
1	0.121	-2.11	1.03
2	5.798×10^{-5}	-9.75	0.97
3	2.778×10^{-8}	-17.40	1.05

Table 2: Discrete $\mathcal{L}^\infty$ norm of residual with $\gamma = 1.0$ at time $t = 1.0$.

m	$\|g^m\|_\infty$	$\ln(\|g^m\|_\infty)$
1	8.94	2.19
2	0.317	-1.15
3	1.123×10^{-2}	-4.49
4	3.977×10^{-4}	-7.83
5	1.409×10^{-5}	-11.17
6	4.991×10^{-7}	-14.51
7	1.768×10^{-8}	-17.85

5. Discussion

In this section we discuss error estimates and convergence rates that were found computationally. Also, comparisons are made of the convergence rates for the BEPS iteration for differing values of the iteration parameter γ. Experimentally, for the rotating plume problem on the given grid, an optimal iteration parameter of $\gamma = 0.778$ is found. A theory for predicting optimal values of γ in general would be helpful here. We also investigate the apparent convergence rate of the iterative scheme.

Tables 1 and 2 contain values for the norm of the residual after each iteration and the natural logarithm of the norm for the rotating plume problem in Example 1. Residuals are given after one complete rotation of the plume at time $t = 1.0$. Table 1 shows the results for an iteration parameter of $\gamma = 0.8$. Table 2 shows the results when no scaling of the iteration is done, in other words $\gamma = 1.0$.

In Table 1 the value of the iteration parameter is close to the optimal iteration parameter $\gamma = 0.778$ and the BEPS iteration converges in only three steps with a tolerance of 5.0×10^{-8}. If scaling of the iteration is not done in Example 1, the BEPS iteration takes seven steps to converge with the same tolerance as shown in Table 2. In both cases the BEPS iteration appears to converge linearly.

The convergence of the numerical solution to the true solution as the level of refinement increases is also investigated computationally. In Table 3 the $\mathcal{L}_\infty$ error at time $t = 1.0$ is tabulated for various levels of refinement. Here, h_r is equal to the fine grid mesh size in the x and y directions. A 20×20-element coarse grid is used for this example of the rotating plume.

Taking the data from Table 3 and computing the line using least squares it is found that the the error is roughly $\mathcal{O}(h_r^{3.41})$. This represents a slight loss in

Table 3: $\mathcal{L}_\infty$ norm of the error at time $t = 1.0$.

	h_r	$\|e\|_\infty$	$\ln(\|e\|_\infty)$
2×2	0.05	8.34×10^{-2}	-2.48
4×4	0.025	7.91×10^{-3}	-4.89
8×8	0.0125	7.03×10^{-4}	-7.26

the convergence rate compared with the $\mathcal{O}(h^4)$ error estimates that hold for the collocation scheme without local grid refinement.

6. Conclusions

This paper presents a numerical scheme for solving the advection-diffusion equation using finite-element collocation and domain-decomposition techniques with adaptive local grid refinement. On the basis of computational experience, it appears that BEPS-like algorithms, which allow one to decouple composite-grid sytems into seperate coarse- and fine-grid problems, hold promise for collocation on Hermite bicubic trial spaces.

While this computational experience is important practically, from a theoretical point of view much further work needs to be done. For example, it would be desirable to show that the preconditioner for these methods is related to the original composite matrix by some bounds on the maximum and minimum eigenvalues. Another important result but a very difficult one to obtain is an analytical way to compute the optimum scaling parameter γ for the BEPS-like iteration. For as we have seen, proper scaling of the iteration can dramatically decrease the number of iterations necessary for convergence at each time level, even though we obtained reasonable convergence with the "naive" choice $\gamma = 1$. Also, even though computationally found error estimates are promising, rigorous theoretical error estimates for the method need to be developed. Extension to nonlinear problems or coupled systems of partial differential equations is another useful avenue for further research.

Extension of the method to three-dimensional problems should be considered. Also, techniques should be studied for extending the local grid refinement ideas presented here to nonrectangular regions. Two methods for possibly doing this are by the inclusion of overlapping domain-decomposition ideas or methods where abutting regions are refined and problems along their boundaries need to be solved as well.

References

[1] M.C. Curran and M.B. Allen, *A Domain Decomposition approach to adaptive grid refinement in finite-element collocation*, Numer. Meth. P.D.E., (to appear).

[2] M. B. Allen and M. C. Curran, *Adaptive local grid refinement algorithms for finite-element collocation*, Numer. Meth. P.D.E., 5(1989), 105-117.

[3] Bangia, V.K., C. Bennett, and A. Reynolds, *Alternating direction collocation for simulating reservoir performance*, presented at the 53rd Annual Fall Conference, Society of Petroleum Engineers, Houston, 1978.

[4] J.H. Bramble, R.E. Ewing, J.E. Pasciak, and A.H. Schatz, *A preconditioning technique for the efficient solution of problems with local grid refinement*, Comp. Meth. Appl. Mech. Eng., 67(1988), 149-159.

[5] J.H. Bramble, R.E. Ewing, R.R. Pareshkevov, and J.E. Pasciak, *Domain decomposition methods for problems with partial refinement*, SIAM J. Sci. Statist. Comput., (to appear).

[6] J.H. Bramble, J.E. Pasciak, and A.H. Schatz, *The construction of preconditioners for elliptic problems by substructuring*, I, Math. Comp. 47(1986), 103-134.

[7] M.A. Celia, *Collocation on Deformed Finite Elements and Alternating Direction Collocation Methods*, Ph.D. Dissertation, Princeton University, 1983.

[8] M.A. Celia, G.F. Pinder, and L.J. Hayes, *Alternating direction collocation simulation of the transport equation*, Proceedings Third Int. Conf. Finite Elements in Water Resources, S.Y. Wang *et al.*, Eds., University of Mississippi, Oxford, MS, 1980, 3.36–3.48.

[9] M.A. Celia and G.F. Pinder, *Analysis of alternating-direction methods for parabolic equations*, Numer. Meth. P.D.E., 1(1985), 57-70.

[10] P.W. Chang and B.A. Finlayson, *Orthogonal collocation on finite elements for elliptic equations*, Math. Comp. Simulation, (1978), 83–92.

[11] M. C. Curran and M. B. Allen, *Parallel computing for solute transport models via alternating direction collocation*, Adv. Wat. Res., 13:2(1990), 70–75.

[12] J. Douglas Jr. and T. Dupont, *Alternating-direction galerkin methods on rectangles*, Numerical Solution of Partial Differential Equations, Synspade 1970, B. Hubbard, Ed., Academic, New York, 2(1971), 133–214.

[13] J. Dendy and G. Fairweather, *Alternating-direction galerkin schemes for parabolic and hyperbolic problems on rectangular polygons*, SIAM J. Numer. Anal., 2(1975), 144–163.

[14] R.E. Ewing, discussions on August 24, 1990.

[15] E.O. Frind and G.F. Pinder, *A collocation finite element method for potential problems in irregular domains*, Int. J. Numer. Meth. Eng., 14(1979), 681-701.

[16] J.F. Guarnaccia and G.F. Pinder, *A parallel collocation based algorithm for the generalized transport equation*, Applications of supercomputers in engineering: Fluid flow and stress analysis applications, C.A. Brebia and A. Peters, Eds., Elsevier and Computational Mechanics Publications, 1989.

[17] L.J. Hayes, *An alternating-direction collocation method for finite element approximations on rectangles*, Comput. Math. Appl., 6(1980), 45–50.

[18] L.J. Hayes, and S.V. Krishnamachari, *Alternating direction along flow lines in a fluid flow problem*, Comp. Meth. App. Mech. and Eng., 47(1984).

[19] O.K. Jensen, and B.A. Finlayson, *Oscillation limits for weighted residual methods applied to convective diffusion equations*, Int. J. Numer. Meth. Eng., 15(1980), 1681–1689.

[20] D.W. Peaceman, and H.H. Rachford, *The numerical solution of parabolic and elliptic equations*, SIAM J., 3(1955), 28–41.

WATER RESOURCES RESEARCH, VOL. 28, NO. 6, PAGES 1715–1722, JUNE 1992

The Finite Layer Method for Groundwater Flow Models

STANLEY S. SMITH AND MYRON B. ALLEN

Department of Mathematics, University of Wyoming, Laramie

JAY PUCKETT AND THOMAS EDGAR

Department of Civil Engineering, University of Wyoming, Laramie

The finite layer method (FLM) is an extension of the finite strip method familiar in structural engineering. The idea behind the method is to discretize two space dimensions using truncated Fourier series, approximating variations in the third via finite elements. The eigenfunctions used in the Fourier expansions are orthogonal, and, consequently, the Galerkin integrations decouple the weighted residual equations associated with different Fourier modes. The method therefore reduces three-dimensional problems to sets of independent matrix equations that one can solve either sequentially on a microcomputer or concurrently on a parallel processor. The latter capability makes the method suitable for such computationally intensive applications as optimization and inverse problems. Four groundwater flow applications are presented to demonstrate the effectiveness of FLM as a forward solver.

1. INTRODUCTION

The finite layer method (FLM) is a numerical method that shows promise for modeling many aquifer flow problems. The idea behind the method is to discretize one dimension of the spatial domain using finite elements, approximating variations in the other two dimensions using truncated Fourier series. For problems having sufficient geometric simplicity this approach avoids much of the expense associated with three-dimensional finite elements. When the Fourier series is composed of orthogonal eigenfunctions, the finite element integration decouples the equation sets for different Fourier modes, and it is therefore possible to solve many small, simultaneous matrix equations in parallel. This inherent parallelism can be especially important when it is necessary to execute a flow model iteratively, as in parameter identification and optimization studies. This paper examines the application of the FLM to several problems of interest to groundwater hydrologists.

Much of the literature relevant to the FLM concerns its predecessor, the finite strip method (FSM) [*Cheung*, 1976], and applications to structural engineering. The FSM uses truncated Fourier series to discretize problems along one coordinate axis instead of two. *Puckett and Wiseman* [1987] review the literature on the FSM pertaining to structural analysis. The FLM itself has received some attention, for example, in the analysis of elastic, horizontally layered foundations [*Cheung and Fan*, 1979]. It is also possible to extend the FLM to problems with infinite layers having finite thickness. *Rowe and Booker* [1982] apply this technique to elastic soils, as do *Small and Booker* [1984*a*]. *Booker and Small* [1982*a*, *b*, 1986] also use this approach to model soil consolidation and surface deformation accompanying the extraction of water [*Small and Booker*, 1984*b*]. *Slattery* [1986] and, subsequently, *Puckett and Schmidt* [1990] utilize the FSM to obtain head distributions in two-dimensional well drawdown models.

Copyright 1992 by the American Geophysical Union.

Paper number 92WR00425.
0043-1397/92/92WR-00425$05.00

One way to think of the FLM is as a quasi-analytic method, in which one incorporates analytic information about the initial boundary value problem (in this case, the eigenfunctions of the spatial operator) into the numerical approximation. Other quasi-analytic methods, similar in spirit but different in detail, have appeared in the water resources literature, including the finite analytic method [*Hwang et al.*, 1985] and the Laplace transform Galerkin method [*Sudicky*, 1989], among others. The FLM is also related to the spectral method [*Gottlieb and Orszag*, 1977]. The two methods share the idea of approximating spatial variations using truncated series of eigenfunctions. Where they differ is in the use of finite element approximations to discretize problems along one of the three spatial coordinates in the FLM. This device facilitates the simulation of certain geometrically simple heterogeneities, such as those occurring in stratified sedimentary basins.

In this paper we present the formulation of the FLM, discuss several coding aspects of the method, and demonstrate its application to four problems. The first problem involves a fully penetrating well; the second involves injection of water at a single point in the aquifer; the third is a three-dimensional model of a leaky aquifer; and the fourth is a model of a multiwell field. We do not present a full error analysis for the method, which is logically the subject of another article. Such an analysis would clarify the trade-offs between accuracy and computational effort, both in the choice of Fourier discretizations and in the finite element gridding.

2. FORMULATION OF THE FINITE LAYER METHOD

The FLM rests on certain geometric assumptions about the problem's spatial domain $\mathcal{D}$. In particular, we consider $\mathcal{D}$ to be a rectangular parallelepiped consisting of a saturated, confined aquifer in which Darcy's law applies. We assume that the coordinate axes coincide with the principal directions of the hydraulic conductivity tensor and that principal hydraulic conductivities K_x, K_y, and K_z vary only with elevation z above datum. These assumptions are reasonable

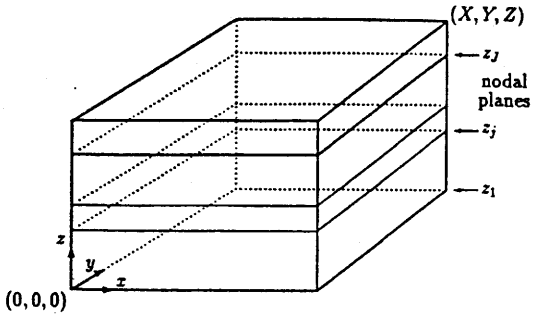

Fig. 1. Typical rectangular parallelepiped domain $\mathcal{D}$.

in many sedimentary formations where the bedding planes are nearly parallel. Figure 1 depicts a typical domain $\mathcal{D}$ with dimensions X, Y, and Z.

We begin by establishing the boundary value problem to be solved and define the differential operator $L[\ ]$ as follows:

$$L[h] = -K_x \frac{\partial^2 h}{\partial x^2} - K_y \frac{\partial^2 h}{\partial y^2} - \frac{\partial}{\partial z}\left(K_z \frac{\partial h}{\partial z}\right) + S_s \frac{\partial h}{\partial t}. \quad (1)$$

Here S_s is the specific storage. Under our assumptions the equation governing the head $h(x, y, z, t)$ is

$$L[h] + F = 0, \quad (2)$$

where the prescribed forcing function $F(x, y, z, t)$ gives the rate of water withdrawal per unit volume of porous medium. We refer readers to *Huyakorn and Pinder* [1983] and *Walton* [1970] for the derivation of (2).

When the withdrawal (or injection) occurs at a point sink or source, one can take F to be a possibly time-dependent multiple of the Dirac δ distribution. Superpositions of such distributions, centered at different spatial points, are also possible, as are more general functional forms. As with most discrete methods, the FLM has a limited ability to capture the steep head gradients that occur near point sources and sinks. In the examples discussed below the FLM produces results that appear reasonable, but for more accuracy one might employ some special technique, such as singularity removal [*Lowry et al.*, 1989] to improve the approximations.

In our first two test problems below we use the initial condition $h(x, y, z, 0) = 0$ and impose no-flow conditions ($\partial h/\partial z = 0$) on the two horizontal planes representing the impermeable confining layers. In the third problem we impose the condition $h = 0$ at the top of the semipermeable aquitard and a no-flow condition at the bottom of the aquifer. In all three problems we impose the condition $h = 0$ at the vertical planes $x = 0$, $x = X$, $y = 0$, and $y = Y$.

To discretize these problems, we divide the domain $\mathcal{D}$ into J layers that are normal to the z axis. The jth layer has thickness $(\Delta z)_j$, and the aquifer characteristics remain constant within each layer; however, they may vary from layer to layer. Each layer j is bounded above and below by nodal planes $z = z_j$ and $z = z_{j+1}$, so that $(\Delta z)_j = z_{j+1} - z_j$.

At any time t we represent the hydraulic head $h(x, y, z_j, t)$ on a single nodal plane $z = z_j$ by a function $h_j(x, y, t)$ satisfying the prescribed conditions $h_j(0, y, t) = h_j(X, y, t) = h_j(x, 0, t) = h_j(x, Y, t) = 0$ at the vertical boundaries. These conditions allow an exact representation of the (x, y) variations in h_j as a double Fourier sine series, in which the Fourier coefficients Φ_{mnj} are time-dependent:

$$h_j(x, y, t) = \sum_{m=1}^{\infty} \sum_{n=1}^{\infty} \Phi_{mnj}(t) G_{mn}(x, y). \quad (3)$$

Here

$$G_{mn}(x, y) = \sin(n\pi x/X) \sin(m\pi y/Y). \quad (4)$$

For the numerical method we truncate this series, getting an approximation

$$h_j(x, y, t) \approx \sum_{m=1}^{M} \sum_{n=1}^{N} \Phi_{mj}(t) G_{mn}(x, y), \quad (5)$$

where M and N are determined by the level of accuracy desired.

To define the vertical variation of the approximate head $\hat{h}$, we linearly interpolate between nodal planes:

$$\hat{h}(x, y, z, t) = \sum_{j=1}^{J+1} \left[\sum_{m=1}^{M} \sum_{n=1}^{N} \Phi_{mnj}(t) G_{mn}(x, y) \right] N_j(z). \quad (6)$$

Here the functions $N_j(z)$ are standard linear shape functions in the z direction:

$$N_j(z) = 0 \qquad z \geq z_{j+1} \text{ or } z \leq z_{j-1},$$
$$N_j(z) = (z - z_{j-1})/(\Delta z)_{j-1} \qquad z_{j-1} < z \leq z_j, \quad (7)$$
$$N_j(z) = (z_{j+1} - z)/(\Delta z)_j \qquad z_j < z \leq z_{j+1}.$$

To determine the unknown coefficients Φ_{mnj}, we develop a linear system of ordinary differential equations in time by using the following weighted residual equations:

$$\iiint_{\mathcal{D}} \{L[\hat{h}] + F\} w_{m'n'i}\, dx\, dy\, dz = 0. \quad (8)$$

We use as weight functions the shape functions associated with the unknown coefficients Φ_{mnj}, namely,

$$w_{m'n'i}(x, y, z) = N_i(z) G_{m'n'}(x, y). \quad (9)$$

If we interchange the operations of differentiation and integration with the finite summation implicit in $\hat{h}$, (8) becomes

$$\sum_{j=1}^{J+1} \sum_{m=1}^{M} \sum_{n=1}^{N} \iiint_{\mathcal{D}} \{L[\Phi_{mnj} N_j G_{mn}] + F\} \cdot N_i G_{m'n'}\, dx\, dy\, dz = 0. \quad (10)$$

We now integrate by parts to shift one order of differentiation from $\hat{h}$ to the weight function $N_i G_{m'n'}$. In doing so, we simplify matters by observing that the eigenfunctions $G_{mn}(x, y)$ obey orthogonality relationships guaranteeing that, whenever $m \neq m'$ or $n \neq n'$,

$$\int_0^Y \int_0^X G_{mn} G_{m'n'}\, dx\, dy = 0, \quad (11a)$$

$$\int_0^Y \int_0^X \frac{\partial^2 G_{mn}}{\partial x^2} G_{m'n'}\, dx\, dy = 0, \tag{11b}$$

$$\int_0^Y \int_0^X \frac{\partial^2 G_{mn}}{\partial y^2} G_{m'n'}\, dx\, dy = 0. \tag{11c}$$

Therefore the only terms that survive the integration and summation in (10) are those for which $m = m'$ and $n = n'$, and we get

$$\sum_{j=1}^{J+1} \left[- \Phi_{mnj} \int_0^Z K_x N_j N_i\, dz \int_0^Y \int_0^X \frac{\partial^2 G_{mn}}{\partial x^2} G_{mn}\, dx\, dy \right.$$

$$- \Phi_{mnj} \int_0^Z K_y N_j N_i\, dz \int_0^Y \int_0^X \frac{\partial^2 G_{mn}}{\partial y^2} G_{mn}\, dx\, dy$$

$$- \Phi_{mnj} K_z \frac{\partial N_j}{\partial z} N_i \Big|_0^Z \int_0^Y \int_0^X G_{mn}^2\, dx\, dy$$

$$+ \Phi_{mnj} \int_0^Z K_z \left(\frac{\partial N_j}{\partial z}\right)^2 dz \int_0^Y \int_0^X G_{mn}^2\, dx\, dy$$

$$\left. + \frac{\partial \Phi_{mnj}}{\partial t} \int_0^Z S_s N_j N_i\, dz \int_0^Y \int_0^X G_{mn}^2\, dx\, dy \right]$$

$$+ Q_{mni} = 0. \tag{12}$$

Here

$$Q_{mni} = \iiint_{\mathcal{L}} F N_i G_{mn}\, dx\, dy\, dz. \tag{13}$$

One equation of the form (12) holds for each distinct triple (i, m, n) of indices associated with a weight function. For simplicity, we represent the forcing function F by a constant multiple of the Dirac δ distribution $\delta(x, y)$).

As with the usual finite element method using piecewise linear basis functions, terms in (12) for which $|i - j| \geq 2$ vanish, yielding tridiagonal systems with unknowns Φ_{mnj}. If the bottom (or top) of the aquifer is a no-flow boundary, the contributions at $z = 0$ (or $z = Z$) that arise from the integration by parts also vanish. Moreover, owing to the orthogonality relations in (11), each Fourier mode (m, n) has its own matrix equation:

$$[M]_{mn}\Phi_{mn} + [B]_{mn}\, d\Phi_{mn}/dt + \mathbf{Q}_{mn} = \mathbf{0}, \tag{14}$$

where $[M]_{mn}$ and $[B]_{mn}$ are tridiagonal matrices, and Φ_{mn} and $\mathbf{Q}_{mn}$ are vectors with components Φ_{mnj} and Q_{mnj}, $j = 1, \cdots, N + 1$, respectively. The typical $[M]_{mn}$ and $[B]_{mn}$ tridiagonal entries for a specific layer j (where $1 \leq j \leq J$) are as follows:

$$m_{j1} = \frac{XY}{4}\left\{\left[(K_x)_j\left(\frac{n\pi}{X}\right)^2 + (K_y)_j\left(\frac{m\pi}{Y}\right)^2\right]\frac{(\Delta z)_j}{3} + \frac{(K_z)_j}{(\Delta z)_j}\right\},$$

$$m_{j2} = \frac{XY}{4}\left\{\left[(K_x)_j\left(\frac{n\pi}{X}\right)^2 + (K_y)_j\left(\frac{m\pi}{Y}\right)^2\right]\frac{(\Delta z)_j}{6} - \frac{(K_z)_j}{(\Delta z)_j}\right\},$$

$$m_{j3} = m_{j2},$$

$$m_{j4} = m_{j1},$$

$$b_{j1} = XYS_s(\Delta z)_j/12,$$

$$b_{j2} = b_{j1}/2,$$

$$b_{j3} = b_{j2},$$

$$b_{j4} = b_{j1},$$

Figure 2 depicts how $[M]_{mn}$ and $[B]_{mn}$ are assembled and what entries the 2×2 matrices have.

We approximate the time derivative by a simple difference scheme in Φ:

$$\left.\frac{d\Phi_{mn}}{dt}\right|^{k+\theta} \approx \frac{\Phi_{mn}^{k+1} - \Phi_{mn}^{k}}{\Delta t};$$
$$\Phi_{mn}^{k+\theta} \approx \theta\Phi_{mn}^{k+1} + (1 - \theta)\Phi_{mn}^{k}. \tag{15}$$

Here k indexes the most recent time level at which Φ_{mn} is known, and $k + 1$ indexes the next time level. We represent the time increment between these two levels by Δt and use θ to denote a weighting parameter, discussed shortly. The temporally discrete system therefore becomes

$$\left\{\theta[M]_{mn} + \frac{1}{\Delta t}[B]_{mn}\right\}\Phi_{mn}^{k+1}$$
$$= \left\{\frac{1}{\Delta t}[B]_{mn} - (1 - \theta)[M]_{mn}\right\}\Phi_{mn}^{k} - \mathbf{Q}_{mn}^{k}. \tag{16}$$

Choosing various values of $\theta \in [0, 1]$ yields various temporal weightings of the scheme, with $\theta = 0$ giving an explicit scheme and $\theta = 1$ yielding a fully implicit scheme. We use $\theta = 1/2$, which corresponds to the familiar Crank-Nicolson scheme. This scheme is unconditionally stable and is second-order accurate in t.

3. CODING CONSIDERATIONS

Together with initial conditions and boundary conditions, the model requires the following information: layer-dependent variables, constant within each layer or nodal plane; mode-dependent variables, constant for each Fourier

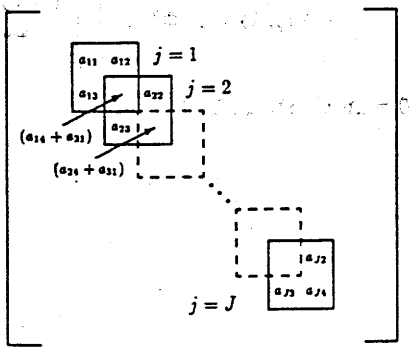

Fig. 2. Matrix assembly for tridiagonal matrices $[M]_{mn}$ and $[B]_{mn}$.

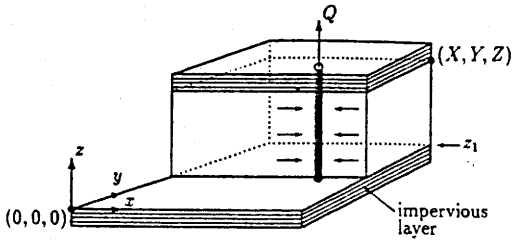

Fig. 3. Geometry of the fully penetrating well.

component; variables characterizing sources (F), and timing variables.

The layer-dependent variables include the number of layers J, the dimensions of each layer, and the conductivities and specific storage of each layer. Variables associated with the Fourier modes include the indices M and N at which the two-dimensional series will be truncated and a matrix [Φ] in which to store Fourier coefficients for each nodal plane.The initial value of [Φ] reflects the initial condition of the aquifer. The variables needed to characterize sources include well locations and volumetric flow rates between nodal planes. The timing variables include the total time t_{total}, the time step Δt, and the temporal weighting parameter θ.

The FLM has advantages in both small-scale and large-scale computing environments. Because the method reduces three-dimensional problems to sets of one-dimensional problems, one can often use a microcomputer to model large, three-dimensional aquifers that would otherwise require too much memory. On the other hand, since the one-dimensional problems are uncoupled, the method is also very adaptable to parallel computing environments. We discuss this possibility further in section 4.

4. Test Problems and Results

We examine four test problems. The first two problems have exact solutions in ideal cases, when the sources have infinitesimal radius and the aquifers have infinite areal extent. The third problem has no exact solution, but there is a classical, closed-form solution that is available if we accept certain simplifying assumptions. The exact solution for the first problem can be used with superposition to obtain an exact solution for the fourth case.

Single, Fully Penetrating Well

Figure 3 depicts a fully penetrating well with a constant discharge rate Q and horizontal flow within the aquifer, and Table 1 summarizes the parameters defining the problem.

TABLE 1. Input Data for the Fully Penetrating Well Problem

	Definition
Depth of aquifer	$Z = 100$ m
Plan dimensions	$X = Y = 1280$ m
Well location	$(x_s, y_s) = (640, 640)$
Hydraulic conductivity	$K = 4$ m/d
Specific storage	$S_s = 1.6 \times 10^{-6}$/m
Discharge rate	$Q = -1257$ m^3/d
Number of modes	$M = N = 32$
Number of layers	$J = 1$
Time step	$\Delta t = 0.001$ day
Total time	$t_{\text{total}} = 0.02$ day

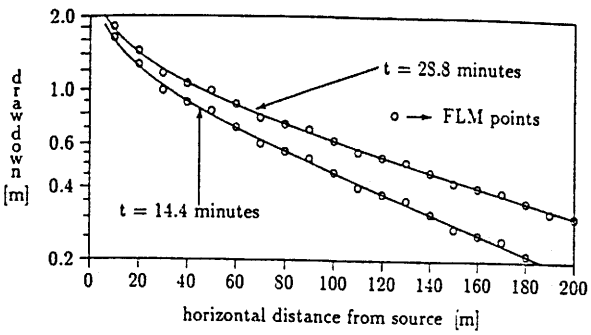

Fig. 4. Hydraulic head h versus distance r from the single, fully penetrating well. Solid curves depict the classical, one-dimensional radial solution.

The exact solution that we use for comparison is a similarity solution for a line source having infinitesimal radius in a one-dimensional, radial problem, where $r = (x^2 + y^2)^{1/2}$ is the distance from the well. Walton [1970] gives this exact solution as

$$h(r, t) = \frac{Q}{4\pi KZ}\left[-\gamma - \ln u + \sum_{n=1}^{\infty} \frac{(-u)^n}{n(n!)}\right], \quad (17)$$

where $u = (r^2 S_s Z)/(4KZt)$ is the similarity variable and $\gamma \approx 0.5772$ is the Euler constant.

In the numerical model we keep t_{total} small and use large values for X and Y to reduce the influence of the zero-head boundary, since the similarity solution applies to a domain of infinite areal extent. As Figure 4 indicates, the FLM approximation in this case is essentially indistinguishable from the similarity solution.

Point Source Injection

The primary purpose of this test problem is to demonstrate the ability of the layers to model vertical gradients in head. Using a specific storage $S_s = 1.0$ facilitates comparison of the results to the corresponding problem in heat conduction. Figure 5 depicts a point source injection well with a constant injection rate Q, corresponding to a well screened over a small vertical interval. Table 2 summarizes the parameters used to define a sample problem for this geometry. The layer thickness varies from 0.1 to 1.5 m, where we concentrate a large number of layers at and above the point source. The exact solution used for comparison represents radial flow from a point source in a domain having infinite areal extent. *Carslaw and Jaeger* [1959] give this solution as

$$h(r, t) = (Q/4\pi Kr)\ \text{erfc}\ [r/(4Kt)^{1/2}]. \quad (18)$$

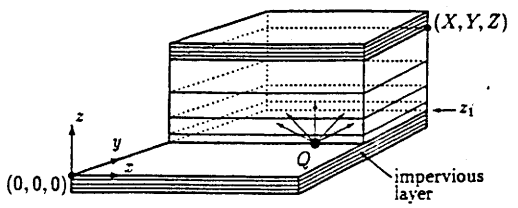

Fig. 5. Geometry of the point source injection well.

TABLE 2. Input Data for the Point Source Injection Well

	Definition
Depth of aquifer	$Z = 32$ m
Plan dimensions	$X = Y = 64$ m
Line source location	$(x_s, y_s) = (32, 32)$
Hydraulic conductivity	$K = 195.3$ m/d
Specific storage	$S_s = 1$ per meter
Injection rate	$Q = 2000$ m^3/d
Number of modes	$M = N = 64$
Number of layers	$J = 50$
Layer thickness	0.1–1.5 m
Time step	$\Delta t = 0.001$ day
Total time	$t_{total} = 0.04$ day

We use a Chebyshev approximation to erfc [see *Press et al.*, 1988]. As in the first sample problem, we keep t_{total} small to avoid the influence of the computational boundaries in the FLM model.

We compare the exact solution with the FLM approximation along two directions from the point source: one on the nodal plane normal to the z axis and one parallel to the z axis. Figures 6 and 7 show these comparisons. As with the fully penetrating well, the FLM gives a good approximation to the exact solution except near the well bore. The discrepancy for $r < 1/2$ m is attributable to the assumption in the exact solution that the source has infinitesimal radius, which implies that the exact solution is unbounded as $r \to 0$. The pressure near the point source remains finite in the FLM solution.

Single Well in a Leaky Aquifer

As a third example we use the FLM to simulate unsteady radial flow in a leaky, isotropic, confined aquifer where a fully penetrating well discharges at a constant rate, as shown in Figure 8. We present two separate runs to illustrate the effectiveness of the FLM model. Table 3 contains the parameters defining them. *Walton* [1970] provides a classical one-dimensional radial solution for this problem, again assuming a well having infinitesimal radius in an aquifer of infinite radial extent:

$$h(r, t) = (Q/4\pi K_A Z_A) W(u, B). \qquad (19)$$

Here K_A and Z_A are the conductivity and depth, respectively, of the aquifer. The well function $W(u, B)$ is represented by the integral

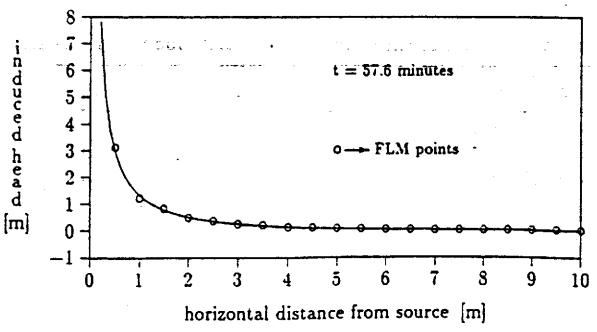

Fig. 6. Numerical and classical solutions for point source injection plotted along the horizontal line $((x, y, z) = (x, y_s, 0))$. The one-dimensional classical solution is depicted by the solid curve.

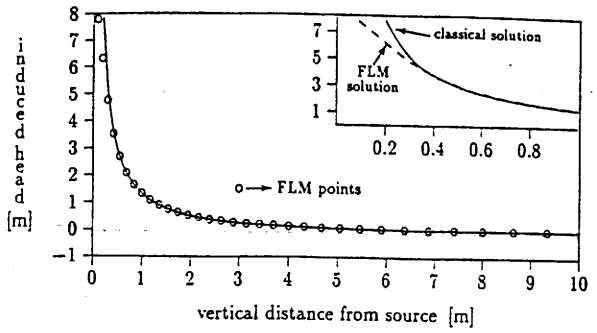

Fig. 7. Numerical and classical solutions for point source injection plotted along the vertical line $((x, y, z) = (x_s, y_s, z))$. The one-dimensional classical solution is depicted by the solid curve. The inset compares the solutions close to the source.

$$W(u, B) = \int_u^\infty \frac{e^{-y}}{y} \exp\left(\frac{-r^2 K_T}{4K_A Z_A Z_T y}\right) dy, \qquad (20)$$

where K_T and Z_T stand for the conductivity and depth of the aquitard. To derive this solution, one must assume that the vertical component of water velocity vanishes in the aquifer. Thus the classical solution unrealistically requires flow lines to be refracted instantaneously from vertical to horizontal as they cross the aquitard-aquifer interface. The classical solution also incorporates the assumption that water is not released from storage in the aquitard. Since $S_s = 0$ in the aquitard, the drawdown varies linearly with elevation, and the vertical velocity is independent of z in the aquitard. As we argue below, the numerical solutions depict more realistic values of the drawdown, capturing a vertical component of velocity in the aquifer and a changing vertical component of velocity in the semipermeable aquitard at early times. As time proceeds, the numerical model approaches the classical solution as expected.

Figure 9 and Figure 10 summarize the first run. Figure 9 shows the drawdown in the classical solution and in the numerical solution generated by the FLM at a radius of 50 m from the source at two time intervals. Figure 10 shows the corresponding values of vertical velocity in the aquitard. The vertical velocity in the aquifer is essentially constant at about 0.001 m/d. The FLM solution at $t = 2.88$ min illustrates the effects of storage in the semipermeable layer, which the classical model cannot capture. Figure 11 depicts the results of the second run in a log-log format, at an elevation of 15 m. These results are representative of those obtainable from the classical solution. However, the FLM method allows one to distinguish well function values asso-

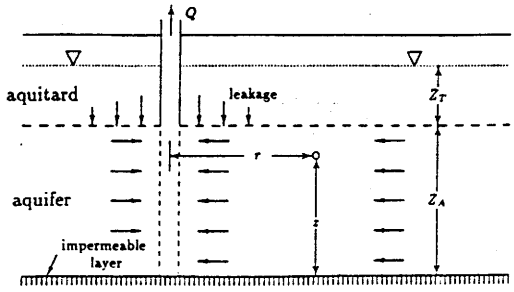

Fig. 8. Geometry of a single well in a leaky aquifer.

TABLE 3. Input Data for the Single Well in a Leaky Aquifer

	Definition
Total depth	$Z = Z_A + Z_T = 80$ m
Aquifer depth	$Z_A = 60$ m
Aquitard depth	$Z_T = 20$ m
Plan dimensions	
Run 1	$X = Y = 1280$ m
Run 2	$X = Y = 3200$ m
Well location	
Run 1	$(x_s, y_s) = (640, 640)$
Run 2	$(x_s, y_s) = (1600, 1600)$
Aquifer conductivity	$K_A = 25$ m/d
Aquitard conductivity	$K_T = 0.12$ m/d
Aquifer specific storage	$S_{sA} = 2.0 \times 10^{-6}$/m
Aquitard specific storage	$S_{sT} = 1.5 \times 10^{-6}$/m
Discharge rate	$Q = 18.850$ m^3/d
Number of modes	$M = N = 64$
Number of layers	$J = 120$
Layer thickness	0.1–15 m
Time step	$\Delta t = 0.0001$–0.001 day

ciated with different elevations within the aquifer, which the classical solution does not. The inset in Figure 11 shows the well function values at different elevations, 15 and 59 m.

Multiwell Field

The primary purpose of the fourth test case is to demonstrate the ability of the FLM to model a multiwell field. Our example has three fully penetrating wells. The first well discharges at a constant rate starting at $t = 0$. The second and third wells inject at constant rates starting at $t = 0.002$ day. Table 4 summarizes the parameters defining the problem. The exact solution that we use for comparison is a superposition of similarity solutions like those used for the first problem.

In the numerical model we keep t_{total} small and use large values for X and Y to reduce the influence of the zero-head boundary, since the similarity solution applies to a domain of infinite areal extent. We compare numerical and exact solutions along the transect $y = 600$ m, which passes close to the three wells. As Figure 12 indicates, the FLM approximation for the case $M = N = 32$ shows virtually no spurious oscillations, being essentially indistinguishable from the similarity solution. At the coarser level of Fourier discretization in which $M = N = 16$, the numerical solution is still reasonable, but some overshooting and oscillations, attributable to the Gibbs phenomenon, are apparent.

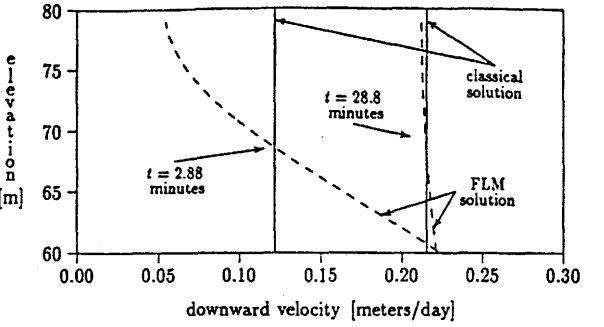

Fig. 10. Vertical velocity in aquitard for the leaky aquifer problem at 50 m from the well, shown at two different times. The solid curve depicts the one-dimensional radial solution, and the dashed curves depict the FLM solution.

Parallelization

Although one can run all of our test problems on a personal computer by sequentially solving the tridiagonal matrix equations for the Fourier modes, it is noteworthy that our code is also amenable to parallel processing. To demonstrate this fact, we present results of the second test problem run on an Alliant FX/8 computer having a shared memory and eight vector processors. Parallelization in a FLM model consists of sending distinct tridiagonal systems to different processors, which then execute the solution algorithm concurrently until all Fourier modes have been computed.

To quantify the efficiency of the parallelization, we examine the CPU time required to solve problems using different numbers p of processors. For each value of p the speedup S_p is the ratio of the time taken by one processor in solving the problem to the time required for p processors. For an ideally parallel algorithm a plot of S_p versus p, called a speedup curve, yields a line having unit slope. In practice, the need for processors to transfer information among themselves prohibits this ideal case, and speedup curves having average slope greater than 0.7 typically indicate excellent parallelism. Figure 13 shows the speedup curve for the second test problem, where $M = N = 64$. The ideal curve is represented by the top curve and has unit slope. The CPU time ratio which was required for just the FLM parallel algorithms is depicted by the lower curve and has a slope of approximately 0.8. For much larger values of M and N we expect the speedups to be somewhat less favorable on shared-memory machines because of computational overhead asso-

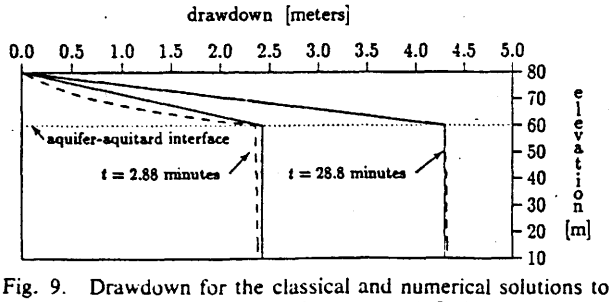

Fig. 9. Drawdown for the classical and numerical solutions to the leaky aquifer problem at 50 m from the well. Solid curves depict the one-dimensional radial solution, and the dashed curves depict the FLM solution.

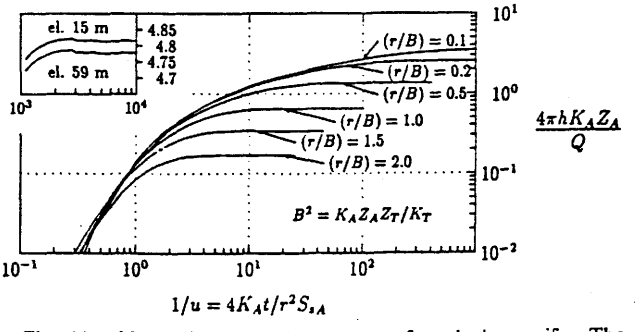

Fig. 11. Normalized drawdown curves for a leaky aquifer. The inset shows the drawdowns at two different depths, as predicted by the FLM model.

TABLE 4. Input Data for the Multiwell Problem

	Definition
Depth of aquifer	$Z = 100$ m
Plan dimensions	$X = Y = 1280$ m
Location	
Well 1	$(x_s, y_s) = (640, 640)$
Well 2	$(x_s, y_s) = (480, 560)$
Well 3	$(x_s, y_s) = (640, 440)$
Hydraulic conductivity	$K = 4$ m/d
Specific storage	$S_s = 1.6 \times 10^{-6}$/m
Discharge rate	
Well 1	$Q = -1257$ m^3/d
Injection rate	
Well 2	$Q = +1000$ m^3/d
Well 3	$Q = +257$ m^3/d
Number of modes	
Run 1	$M = N = 16$
Run 2	$M = N = 32$
Start time	
Well 1	$t = 0.0$
Well 2	$t = 0.002$ day
Well 3	$t = 0.002$ day
Number of layers	$J = 1$
Time step	$\Delta t = 0.001$ day
Total time	$t_{total} = 0.02$ day

ciated with the retrieval of data from cache. For such large-scale problems it is likely that distributed memory machines offer a more effective parallel environment.

5. Conclusions

The FLM offers a numerical approach for modeling aquifer problems having reasonably regular, layered geometry. The method's attractiveness stems from its ability to capture three-dimensional aspects of aquifer behavior in a highly parallelizable fashion, without the intensive computational requirements associated with fully three-dimensional matrices arising in traditional finite element methods. Of course, for complicated heterogeneities the simplified geometry assumed by the FLM is inadequate, and fully three-dimensional models are needed.

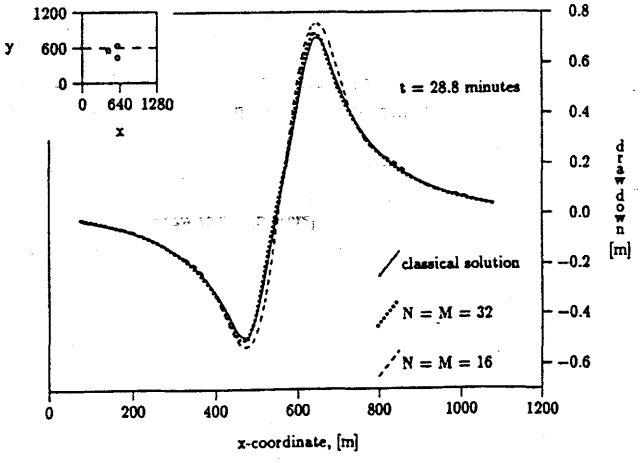

Fig. 12. Drawdown along the transect $(x, y, z) = (x, 600, z)$ for the multiwell problem. Shown are the exact solution and numerical solutions for two different Fourier discretizations. The inset shows the location of the transect (dashed line) with respect to the three wells.

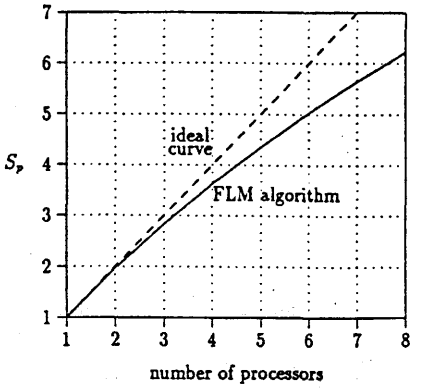

Fig. 13. Speedup curve for FLM model on Alliant FX/8 parallel computer.

We see tremendous potential for the FLM in developing rapidly executable models of groundwater flow. The method's inherent parallelism may make it an attractive choice for applications that require repeated execution, since iteratively running such standard flow codes as MODFLOW [*McDonald and Harbaugh*, 1984] can be prohibitively slow. This advantage can be especially important, for example, in optimization studies and inverse problems.

Notation

Dimensions appear in square brackets.

$[B]_{mn}$ finite element stiffness matrix for Fourier mode mn.
$\mathcal{D}$ three-dimensional domain, $(0, X) \times (0, Y) \times (0, Z)$.
F forcing function $[1/T]$.
$G_{mn}(x, y)$ double sine or cosine function.
h hydraulic head $[L]$.
$\hat{h}$ trial function for hydraulic head $[L]$.
h_j hydraulic head on nodal plane j $[L]$.
i, j nodal plane subscripts; $1 \le i, j \le J + 1$.
J number of layers.
$k, k + 1$ time level superscripts, old and new, respectively.
K_x hydraulic conductivity in the x direction $[L/T]$.
$L[\ ]$ differential operator for transient groundwater flow.
$[M]_{mn}$ finite element mass matrix for a specific Fourier mode, mn.
m, n Fourier mode subscripts.
M, N truncation levels for Fourier series; $1 \le m \le M$ and $1 \le n \le N$.
$N_j(z)$ linear shape function.
p number of processors.
$\mathbf{Q}_{mn}$ forcing vector, equal to $(Q_{mn1}, Q_{mn2}, \cdots, Q_{mn(N+1)})^T$.
Q_{mnj} variational form of forcing function.
r radial distance from line and point sources $[L]$.
S_p speedup.
S_s specific storage $[L^{-1}]$.
t time $[T]$.
t_{total} total time of simulation $[T]$.

u similarity variable.
$W(u, B)$ well function.
x, y, z spatial coordinates (z is elevation above datum) [L].
X, Y, Z dimensions of finite spatial domain [L].
x_s, y_s, z_s coordinate of point source or line source [L].
z_i elevation of layer i; $1 \le i \le J + 1$ [L].
Δt time step [T].
$(\Delta z)_i$ thickness of layer [L].
$[\Phi]$ matrix composed of vectors Φ_{mn}.
Φ_{mn} vector of Fourier coefficients, equal to $(\Phi_{mn1}, \Phi_{mn2}, \cdots, \Phi_{mn(N+1)})^T$.
$\Phi_{mnj}(t)$ Fourier coefficient for nodal plane j.
θ temporal weighting parameter; $0 \le \theta \le 1$.

Acknowledgments. The Wyoming Water Research Center supported this work through a grant in aid. We also received support from NSF grant RII-8610680 and ONR grant 0014-88-K-0370.

REFERENCES

Booker, J. R., and J. C. Small, Finite layer analysis of consolidation, I, *Int. J. Numer. Anal. Methods Geomech.*, *6*(2), 151–171, 1982*a*.

Booker, J. R., and J. C. Small, Finite layer analysis of consolidation, II, *Int. J. Numer. Anal. Methods Geomech.*, *6*(2), 173–194, 1982*b*.

Booker, J. R., and J. C. Small, Finite layer analysis of viscoelastic layered materials, *Int. J. Numer. Anal. Methods Geomech.*, *10*(4), 415–430, 1986.

Carslaw, H. S., and J. C. Jaeger, *Conduction of Heat in Solids*, 2nd ed., Oxford University Press, New York, 1959.

Cheung, Y.-K., *Finite Strip Method in Structural Mechanics*, Pergamon, New York, 1976.

Cheung, Y.-K., and S. C. Fan, Analysis of pavements and layered foundations by finite layer method, in *Proceedings of the Third International Conference on Numerical Methods in Geomechanics*, pp. 1129–1135, edited by W. Wittke, A. A. Balkema, Rotterdam, Netherlands, 1979.

Gottlieb, D., and S. A. Orszag, *Numerical Analysis of Spectral Methods: Theory and Applications*, Society for Industrial and Applied Mathematics, Philadelphia, Pa., 1977.

Huyakorn, P. S., and G. F. Pinder, *Computational Methods in Subsurface Flow*, Academic, San Diego, Calif., 1983.

Hwang, J. S., C. J. Chen, M. Sheikhoslami, and B. K. Panigrahi, Finite analytic solution for two-dimensional groundwater solute transport, *Water Resour. Res.*, *21*(9), 1354–1360, 1985.

Lowry, T., M. B. Allen, and P. N. Shive, Singularity removal: A refinement of resistivity modeling techniques, *Geophysics*, *54*(6), 766–774, 1989.

McDonald, M. G., and A. W. Harbaugh, A modular three-dimensional finite-difference ground-water flow model, *U.S. Geol. Surv. Open File Rep.*, *83-875*, 1984.

Press, W. H., B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C, The Art of Scientific Computing*, Cambridge University Press, New York, 1988.

Puckett, J. A., and R. J. Schmidt, Finite strip method for groundwater modeling in a parallel computing environment, *Eng. Comp.*, *7*(2), 1990.

Puckett, J. A., and D. L. Wiseman, Recent developments in the finite strip methods, paper presented at the Structures Congress, Am. Soc. Civ. Eng., Orlando, Fla., Aug. 1987.

Rowe, R. K., and J. R. Booker, Finite layer analysis of nonhomogeneous soils, *J. Eng. Mech. Div., Am. Soc. Civ. Eng.*, *108*(EM1), 115–132, 1982.

Slattery, J. E., *The Finite Strip Method in Groundwater Hydrology*, M.S. thesis, Colorado State Univ., Fort Collins, 1986.

Small, J. C., and J. R. Booker, Finite layer analysis of layered elastic materials using a flexibility approach, 1, Strip loadings, *Int. J. Numer. Methods Eng.*, *20*(6), 1025–1037, 1984*a*.

Small, J. C., and J. R. Booker, Surface deformation of layered soil deposits due to extraction of water, in *Ninth Australasian Conference on the Mechanics of Structures and Materials*, vol. 9, pp. 33–38, University of Sydney, School of Civil and Mining Engineering, Sydney, Australia, 1984*b*.

Sudicky, E. A., The Laplace transform Galerkin technique: A time-continuous finite element theory and application to mass transport in groundwater, *Water Resour. Res.*, *25*(8), 1833–1846, 1989.

Walton, W. C., *Groundwater Resource Evaluation*, McGraw-Hill, New York, 1970.

M. B. Allen and S. S. Smith, Department of Mathematics, University of Wyoming, Box 3036, Laramie, WY 82071.

T. Edgar and J. Puckett, Department of Civil Engineering, University of Wyoming, Box 3295, Laramie, WY 82071.

(Received April 1, 1991;
revised February 5, 1992;
accepted February 13, 1992.)

Error Analysis of the Finite-Strip Method for Parabolic Equations

Stanley S. Smith
Department of Mathematics
Black Hills State University
Spearfish, SD 57799

Myron B. Allen
Department of Mathematics
University of Wyoming
Laramie, WY 82071

September 9, 1992

Keywords

Finite strip method, groundwater flow models, heat equation, semianalytic methods, spectral methods.

Acknowledgments

The Wyoming Water Research Center supported this work through a grant-in-aid. We also received support from NSF grant RII-8610680 and ONR grant 0014-88-K-0370. The authors thank Professors Jay Puckett and Thomas Edgar of the Department of Civil Engineering, University of Wyoming, for their practical insights into the FSM.

1 Introduction

The finite-strip method (FSM) is a hybrid of the finite-element and spectral methods. Its typical applications are in the numerical solution of partial differential equations in two spatial variables, especially in problems that are geometrically regular in one coordinate direction. Owing to its unusual efficiency, the technique is a familiar one in structural mechanics [3]. It is also useful in models of stratified groundwater flow [7, 10]. A three-dimensional extension of the method, the finite-layer method, has utility in groundwater flow models [11, 12] as well as in other applications. This paper presents an error analysis for the the FSM applied to time-dependent, parabolic partial differential equations. We also indicate how to extend the analysis to the finite-layer method.

The FSM generates an approximate solution that, at each time level, belongs to a peculiar finite-element trial space. This space consists of functions that are piecewise polynomial in the z-direction and are truncated Fourier series in the x-direction. The space has a tensor-product basis, each element of which is a product of two types of one-dimensional basis functions. The first type is associated with traditional finite-element techniques. We partition the z-dimension of the spatial domain by a grid and define piecewise polynomials, such as standard piecewise linear basis functions $\ell_j(z)$, over the grid. The basis functions used for the x-dimension are the trigonometric functions associated with spectral methods [1]. If $\omega_m(x)$ represents a typical element of the trigonometric basis, indexed by the Fourier mode number m, then a typical basis function of the trial space for the FSM has the form $\omega_m(x)\ell_j(z)$. Section 3 discusses this basis in more depth.

We discretize a given initial-boundary-value problem in space by using a Galerkin formulation [8], in which basis functions $\omega_m(x)\ell_j(z)$ serve as weight functions in the weighted-residual equations. We discretize in time using finite differences. Section 4 outlines this formulation in more detail.

In problems having sufficient geometric regularity, the FSM has several computational advantages over traditional finite-element and spectral methods. Chief among these is the fact that it yields a sparse linear system to solve for each Fourier mode of the approximate solution. As discussed briefly in Section 4, the matrix equations for different modes are independent and therefore are amenable to parallel processing.

Several other papers [7, 11, 12] discuss such computational matters in detail. This paper focuses on the analysis of the FSM.

The key question in the error analysis is the following: How does the error in the FSM solution decay as we refine the mesh size h of the finite-element grid in z or increase the number $2M + 1$ of Fourier modes used in x? Our development shows that, when the trial function is piecewise linear in z, the FSM error is $\mathcal{O}(h^2 + M^{-r})$. Here, the exponent $r \geq 2$ increases with the smoothness of the exact solution in the x-direction.

Our paper is organized as follows. Section 2 describes the physical problem of interest and the mathematical assumptions and notation. Section 3 discusses the FSM trial space, and Section 4 describes the FSM formulation. Section 5 estimates the approximation error associated with interpolation and projection maps into the trial space. Using these estimates, Section 6 derives an L^2 estimate of the difference between the approximate FSM solution and the exact solution. This error estimate is then verified computationally in Section 7. In Section 8 we sketch the extension of the analysis to the finite-layer method.

2 The Physical Problem and Notation

Our analysis involves a two-dimensional generalization of the heat equation. Consider a rectangular spatial domain $\Omega := (-\pi, \pi) \times (0, 1)$ with homogeneous Dirichlet boundary conditions and coefficients that vary with z:

$$\left.\begin{aligned} S(z)\partial_t u - K_x(z)\partial_x^2 u - \partial_z[K_z(z)\partial_z u] = f(x,z,t), \quad (x,y) \in \Omega, \quad t \in (0,T] \\ u(x,z,t) = 0, \quad (x,z) \in \partial\Omega, \quad t \in [0,T] \\ u(x,z,0) = u^0(x,z), \quad (x,z) \in \Omega. \end{aligned}\right\} \tag{1}$$

Here, $\partial_x u := \partial u/\partial x$, $\partial_x^2 u := \partial^2 u/\partial x^2$, and so forth. We adopt the following notation to describe the spatial domain: $X := (-\pi, \pi)$; $Z := (0, 1)$; $\Omega := X \times Z$. Also, $\partial\Omega$ denotes the boundary of Ω.

The problem (1) occurs in several applications. In two-dimensional saturated groundwater flow, the coefficient $S(z)$ represents specific storage. The coefficients $K_x(z)$ and $K_z(z)$ in this context denote hydraulic conductivities in the x- and z-directions,

respectively. Huyakorn and Pinder [5], for example, discuss this application in detail. All three coefficients may vary with the vertical coordinate z, as occurs in horizontally uniform sedimentary beds. The function $f(x,z,t)$ accounts for sources, and $u(x,z,t)$ represents the unknown hydraulic head. The boundary-value problem (1) also has applications to conductive heat flow. For a two-dimensional, layered composite slab, $S(z) = 1.0$; $K_x(z)$ and $K_z(z)$ stand for thermal diffusivities, and $u(x,z,t)$ represents temperature. In realistic problems, it is generally necessary to rescale the domain $\Omega = (-\pi,\pi) \times (0,1)$ to physical dimensions. Linear scalings may change the multiplicative constants in our error estimates but do not affect their asymptotic orders.

We assume that K_x and K_z, and S are piecewise constant with respect to z. We also assume that they are positive, bounded away from zero, and bounded above:

$$0 < c \le K_x(z) \le K, \tag{2}$$

$$0 < c \le K_z(z) \le K, \tag{3}$$

$$0 < s \le S(z) \le S^*. \tag{4}$$

We assume that the forcing function f and the initial condition u^0 are smooth enough to guarantee that the solution $u(x,z,t)$ exists, is unique, and depends continuously on these data.

We use a variety of normed function spaces in our analysis. Denote by $L^2(\Omega)$ the space of square-integrable, complex-valued functions defined on Ω. The quantity

$$\|v\|^2_{L^2(\Omega)} = \int_\Omega |v|^2 \, dx \, dz. \tag{5}$$

defines the standard norm on this space. Here, $|v(x,z)|^2 := v(x,z)\overline{v(x,z)}$, the overbar indicating complex conjugation. We use analogous notation for the one-dimensional domains X and Z. For example, the space of square-integrable functions on X is $L^2(X)$, and the corresponding norm is

$$\|v\|^2_{L^2(X)} = \int_X |v|^2 \, dx. \tag{6}$$

Given $v \in L^2(\Omega)$, $v(x, \cdot\,)$ represents a family of functions in $L^2(Z)$ (that is, functions of z), where x is a parameter. Similarly, $v(\cdot\,, z)$ represents a family of functions in $L^2(X)$ indexed by the parameter z. Thus $\|v(x, \cdot\,)\|_{L^2(Z)}$ represents a function in

$L^2(X)$. We sometimes abbreviate this function by writing $\|v\|_{L^2(Z)}$. Likewise, when $v \in L^2(\Omega)$, $\|v\|_{L^2(X)}$ serves as shorthand for the function $\|v(\,\cdot\,, z)\|_{L^2(X)}$.

We denote by $\langle\,\cdot\,,\,\cdot\,\rangle$ the inner product associated with $L^2(\Omega)$. In working with this inner product we occasionally employ Fubini's Theorem (see Royden, [9]) and interchange the order of integration. Thus, if $v_1, v_2 \in L^2(\Omega)$, then

$$\langle v_1(x,z), v_2(x,z)\rangle = \int_X \int_Z v_1(x,z)\overline{v_2(x,z)}\, dz\, dx = \int_Z \int_X v_1(x,z)\overline{v_2(x,z)}\, dx\, dz. \quad (7)$$

We define Sobolev spaces associated with X and Z and then use these definitions to define function spaces over the two-dimensional domain Ω. The Sobolev spaces $H^2(Z)$, $H_0^2(Z)$, and $H_p^r(X)$ are defined in the usual way:

$$H^2(Z) := \left\{v \in L^2(X) : \partial_z^a v \in L^2(Z), \quad \text{for} \quad 0 \le a \le 2\right\} \quad (8)$$

$$H_0^2(Z) := \left\{v \in H^2(Z) : v(0) = v(1) = 0\right\} \quad (9)$$

$$H_p^r(X) := \left\{v \in L^2(X) : \partial_x^a v \in L^2(Z) \quad \text{and is periodic for} \quad 0 \le a \le r\right\}. \quad (10)$$

Following Canuto et al. [1], we define the nonisotropic Hilbert space $H_p^{r,2}(\Omega)$ as the space containing all functions $v \in L^2(\Omega)$ such that

$$\int_X \sum_{\alpha=0}^{2} \|\partial_z^\alpha v\|_{L^2(Z)}^2 dx < \infty \quad (11)$$

and

$$\int_Z \sum_{\alpha=0}^{r} \|\partial_x^\alpha v\|_{L^2(X)}^2 dz < \infty. \quad (12)$$

We assume that $r \ge 1$, and we denote by $\mathcal{H}$ the space containing functions $v \in H_p^{(r+1),2}(\Omega)$ such that $\partial_z^2 \partial_x v \in L^2(\Omega)$ and $v(x,z) = 0$ when $(x,z) \in \partial\Omega$.

3 The Finite-Strip Trial Space

What distinguishes the FSM from other weighted-residual techniques is its trial space. This space, $\tilde{\mathcal{H}}$, is a finite-dimensional subspace of $\mathcal{H}$ whose standard basis contains products $\omega_m(x)\ell_j(z)$ of functions defined on X and Z. For the functions $\ell_j(z)$, we use basis functions for piecewise linear interpolation over a grid defined on Z. Trigonometric

functions, defined below, serve as the basis functions $\omega_m(x)$ defined on X. We now describe this trial space in detail.

The piecewise linear basis $\{\ell_j(z)\}_{j=1}^{J-1}$ requires that Z be partitioned by a grid. Figure 3 depicts the nodal lines associated with the grid $0 = z_0 < z_1 < \cdots < z_J = 1$. We demand that the grid contain all loci of the jump discontinuities in the coefficients K_x, K_z, and S. The mesh size of this grid is

$$h = \max_{j=1,\ldots,J} |z_j - z_{j-1}|. \tag{13}$$

A typical piecewise linear basis function, shown in Figure 3, has local support and satisfies the conditions

$$\ell_j(z_i) = \begin{cases} 0 & i \neq j \\ 1 & i = j \end{cases} \qquad i = 0, 1, \ldots, J, \quad j = 1, 2, \ldots, J-1. \tag{14}$$

These functions span a $(J-1)$-dimensional subspace $\mathcal{V}$ of $L^2(Z)$, namely,

$$\mathcal{V} = \left\{ v \in L^2(Z) : v(z) = \sum_{j=1}^{J-1} v_j \ell_j(z) \right\}. \tag{15}$$

Thus $\mathcal{V}$ contains all functions that are piecewise linear with respect to the given grid and that vanish at the endpoints $z_0 = 0$ and $z_1 = 1$.

The basis for approximation along the horizontal direction consists of trigonometric functions associated with truncated Fourier series on X. Figure 3 depicts one such function. Although Fourier sine-cosine series are typically used in FSM computations, for succinctness we use the complex exponential form. Letting $i^2 = -1$, we have

$$v(x) = \sum_{m=-\infty}^{\infty} \hat{v}_m \omega_m(x) \tag{16}$$

Here, $\omega_m(x) := \exp(imx)$, and $\hat{v}_m$ denotes the Fourier coefficient,

$$\hat{v}_m := \frac{1}{2\pi} \int_X v(x) \overline{\omega_m(x)} dx. \tag{17}$$

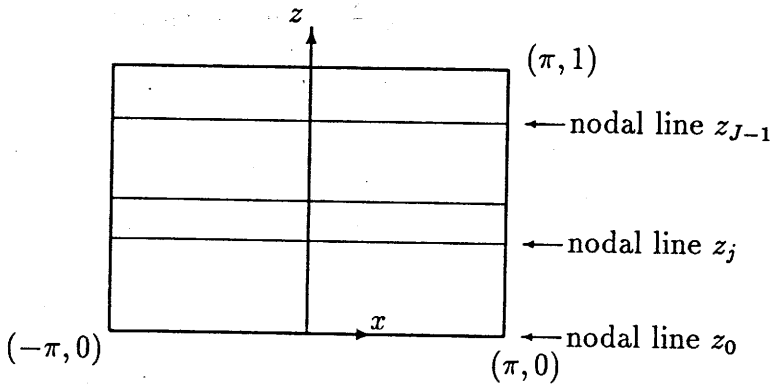

Figure 1: Rectangular domain, Ω.

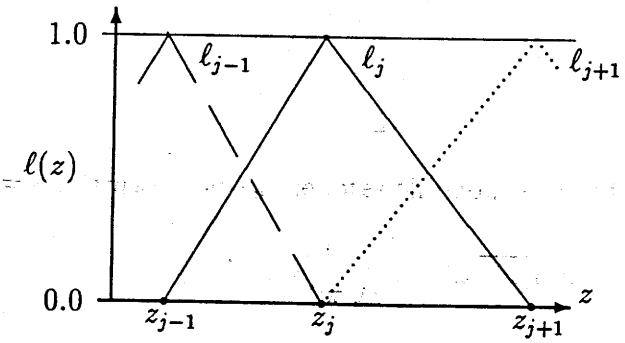

Figure 2: Linear Basis, $\ell_j(z)$.

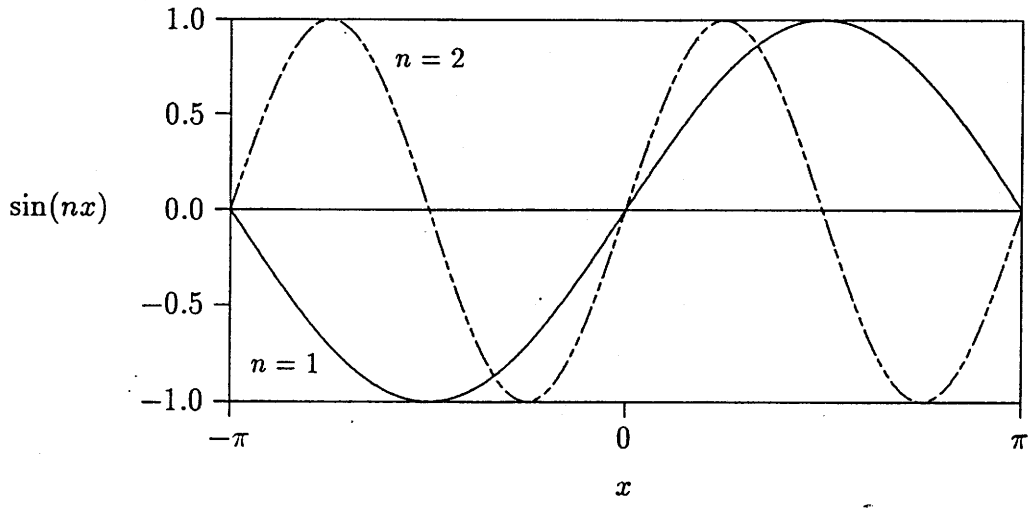

Figure 3: Global Basis Functions, $\{\sin nx\}_{n=1}^{2}$.

We denote by $\mathcal{U}$ the following $(2M+1)$-dimensional subspace of $L^2(X)$:

$$\mathcal{U} := \Big\{ v \in L^2(X) : \hat{v}_m = 0 \text{ for } |m| > M \Big\}. \tag{18}$$

Thus $\mathcal{U}$ contains all Fourier series on X that are truncated at mode number M.

Functions in the trial space $\tilde{\mathcal{H}}$ are bilinear combinations of basis functions associated with $\mathcal{U}$ and $\mathcal{V}$, that is

$$\tilde{\mathcal{H}} = \left\{ v \in \mathcal{H} : v = \sum_{j=1}^{J-1} \sum_{|m| \le M} v_{m,j}\, \omega_m(x) \ell_j(z) \right\} = \mathcal{U} \otimes \mathcal{V}. \tag{19}$$

Functions in $\tilde{\mathcal{H}}$ are thus piecewise linear in z and vary as truncated Fourier series in x. The dimension of $\tilde{\mathcal{H}}$ is therefore $(J-1)\cdot(2M+1)$.

4 Formulation of the FSM

The FSM arises from the following weak form of the exact problem (1): Find a one-parameter family $u(\,\cdot\,,\,\cdot\,,t)$ in $\mathcal{H}$ such that, for all test functions $w \in \mathcal{H}$ and all times $t \in (0,T]$,

$$\langle S\partial_t u, w\rangle + \langle K_x \partial_x u, \partial_x w\rangle + \langle K_z \partial_z u, \partial_z w\rangle = \langle f, w\rangle. \tag{20}$$

To discretize this problem in space, we restrict $u(\cdot,\cdot,t)$ and w to a finite-dimensional subspace of $\mathcal{H}$: Find a one-parameter family of functions $\tilde{u}(\,\cdot\,,\,\cdot\,,t)$ in $\tilde{\mathcal{H}}$ such that, for all $w \in \tilde{\mathcal{H}}$ and all $t \in (0,T]$,

$$\langle S\partial_t \tilde{u}, w\rangle + \langle K_x \partial_x \tilde{u}, \partial_x w\rangle + \langle K_z \partial_z \tilde{u}, \partial_z w\rangle = \langle f, w\rangle. \tag{21}$$

This condition yields a set of $(J-1)\cdot(2M+1)$ ordinary differential equations in time.

Instead of solving these ordinary differential equations, we use a temporally discrete approximation. We replace the function $\tilde{u}(x,z,t)$ by a sequence of functions $\tilde{u}^k(x,z) \approx \tilde{u}(x,z,k\tau)$ in $\tilde{\mathcal{H}}$. Here, τ represents the time step. Similarly, $u^k(x,z)$ signifies the exact solution value $u(x,z,k\tau)$. To solve for $\tilde{u}^k(x,z)$, we introduce the backward difference scheme

$$\left\langle S\frac{\tilde{u}^k - \tilde{u}^{k-1}}{\tau}, w \right\rangle + \left\langle K_x \partial_x \tilde{u}^k, \partial_x w \right\rangle + \left\langle K_z \partial_z \tilde{u}^k, \partial_z w \right\rangle = \left\langle f^k, w \right\rangle. \tag{22}$$

Since $\tilde{u}^k$ has the form

$$\tilde{u}^k(x,z,t) = \sum_{j=1}^{J-1} \sum_{|m| \le M} \Phi^k_{m,j} \omega_m(x) \ell_j(z), \tag{23}$$

our objective is to determine the coefficients $\Phi^k_{m,j}$ at each time level k. To start the calculations, we must choose an appropriate initial function $\tilde{u}^0(x,z)$. In practice, we project the exact initial condition $u^0(x,z)$ into the trial space $\tilde{\mathcal{H}}$ using projection operators defined in the next section.

We determine the unknown coefficients $\Phi^k_{m,j}$ at time level k by solving linear systems obtained using the basis functions $\ell_j(z)\omega_m(x)$ as weight functions w. If we order the weighted-residual equations lexicographically according to the index pairs (m,j), then the choice of the linear basis functions $\ell_j(z)$ for the vertical dimension implies that the linear system is tridiagonal. Our assumptions that K_x and K_z are strictly positive and bounded guarantee that the system is symmetric and positive definite and hence nonsingular at each time level. The system therefore generates a unique sequence $\tilde{u}^k$ in $\tilde{\mathcal{H}}$.

One benefit of the FSM is its efficiency in parallel computing environments. This benefit owes its existence to the orthogonality of the trigonometric basis $\{\omega_m(x)\}_{|m|\le M}$:

$$\frac{1}{2\pi}\int_X \omega_n\overline{\omega_m}\,dx = \begin{cases} 0 & \text{for } m \neq n \\ 1 & \text{for } m = n. \end{cases} \tag{24}$$

We also have

$$\int_X \partial_x\omega_n\overline{\partial_x\omega_m}dx = \int_X mn\,\omega_n\,\overline{\omega_m}\,dx = \begin{cases} 2\pi m^2 & \text{for } m = n \\ 0 & \text{for } m \neq n. \end{cases} \tag{25}$$

Thus the tridiagonal system to be solved at each time level decouples into $(2M+1)$ independent matrix equations of size $J-1$, one system for each Fourier mode. This decoupling allows one to solve for distinct Fourier modes in parallel, as demonstrated computationally in [7, 11, 12].

5 Approximation Error Estimates

In this section, we review error estimates for interpolation and projection into the trial space $\tilde{\mathcal{H}}$. We use these estimates in the error analysis presented later.

Define the interpolation map $\mathcal{I} : L^2(Z) \to \mathcal{V}$ as follows:

$$(\mathcal{I}v)(z) := \sum_{j=1}^{J-1} v(z_j)\ell_j(z). \tag{26}$$

For functions $v \in \mathcal{H}$, we extend this map in the straightforward way:

$$(\mathcal{I}v)(x,z) := \sum_{j=1}^{J-1} v(x,z_j)\ell_j(z). \tag{27}$$

We denote by $\mathcal{P} : L^2(X) \to \mathcal{U}$ the projection that truncates Fourier series to $2M+1$ terms. Provided that $M \geq 1$, we have

$$(\mathcal{P}v)(x) := \sum_{|m|\leq M} \hat{v}_m \omega_m(x). \tag{28}$$

Again, extension to functions of two variables is straightforward: For $v \in \mathcal{H}$,

$$(\mathcal{P}v)(x,z) := \sum_{|m|\leq M} \hat{v}_m(z)\omega_m(x), \tag{29}$$

where $\hat{v}_m(z) := (2\pi)^{-1} \int_X v(x,z)\omega_m(x)\,dx$.

Composition of these maps yields the *approximation map* $\mathcal{IP} : \mathcal{H} \to \tilde{\mathcal{H}}$. For $v \in \mathcal{H}$,

$$(\mathcal{IP}v)(x,z) = \sum_{j=1}^{J-1} \sum_{|m|\leq M} \hat{v}_m(z_j)\ell_j(z)\omega_m(x). \tag{30}$$

In estimating the FSM error $\left\|u^k - \tilde{u}^k\right\|_{L^2(\Omega)}$ in the next section, we need an estimate of $\|v - \mathcal{IP}v\|_{L^2(\Omega)}$, which we call the *approximation error*. To develop this estimate, we first discuss the errors associated with $\mathcal{I}$ and $\mathcal{P}$. Strang and Fix [13] show that the interpolation error for $v \in H^2(Z)$ obeys the bound

$$\|v - \mathcal{I}v\|_{L^2(Z)} \leq \pi^{-2}h^2 \left\|\partial_z^2 v\right\|_{L^2(Z)}. \tag{31}$$

Analogous estimates exist for the projection error associated with $\mathcal{P}$. If $v \in H_p^{r,2}(\Omega)$, where $r \geq 1$ is an integer, then

$$\|v - \mathcal{P}v\|_{L^2(\Omega)} \leq \frac{\sqrt{2\pi}}{M^r} \|\partial_x^r v\|_{L^2(\Omega)}. \tag{32}$$

Canuto et al. [2] outline a proof of this estimate, which we detail in Lemma 10 of the Appendix.

We now prove two lemmas giving an estimate of $\|v - \mathcal{IP}v\|_{L^2(\Omega)}$. In the proofs, we indicate parenthetically the steps where we use the Parseval equality, the Bessel inequality [6], and Fubini's theorem [9]. The first lemma estimates the interpolation error when we apply $\mathcal{I}$ to the truncated Fourier series $\mathcal{P}v$.

Lemma 1 *If $v \in \mathcal{H}$, then*

$$\|\mathcal{P}v - \mathcal{I}\mathcal{P}v\|_{L^2(\Omega)} \leq \sqrt{\frac{2}{\pi^3}}\, h^2 \left\|\partial_z^2 v\right\|_{L^2(\Omega)}. \tag{33}$$

Proof: Using the definition of $\|\cdot\|_{L^2(\Omega)}$, we have

$$\begin{aligned}
\|\mathcal{P}v - \mathcal{I}\mathcal{P}v\|^2_{L^2(\Omega)} &= \int_X \int_Z |\mathcal{P}v(x,z) - \mathcal{I}\mathcal{P}v(x,z)|^2 \, dz \, dx \\
&= \int_X \|\mathcal{P}v(x,\cdot) - \mathcal{I}\mathcal{P}v(x,\cdot)\|^2_{L^2(Z)} dx \\
\text{(Equation (31))} \quad &\leq \int_X \left(\pi^{-2}h^2\right)^2 \left\|\partial_z^2 \mathcal{P}v(x,\cdot)\right\|^2_{L^2(Z)} dx \\
&= \left(\pi^{-2}h^2\right)^2 \int_X \int_Z \left| \sum_{|m|\leq M} \partial_z^2 \hat{v}_m(z)\omega_m(x) \right|^2 dz \, dx \\
\text{(Fubini's Theorem)} \quad &= \left(\pi^{-2}h^2\right)^2 \int_Z \int_X \left| \sum_{|m|\leq M} \partial_z^2 \hat{v}_m(z)\omega_m(x) \right|^2 dx \, dz \\
\text{(orthogonality)} \quad &= \frac{2h^4}{\pi^3} \int_Z \sum_{|m|\leq M} \left|\partial_z^2 \hat{v}_m(z)\right|^2 dz \\
\text{(Bessel inequality)} \quad &\leq \frac{2h^4}{\pi^3} \int_Z \left\|\partial_z^2 v(\,\cdot\,,z)\right\|^2_{L^2(X)} dz \\
&= \frac{2h^4}{\pi^3} \left\|\partial_z^2 v\right\|^2_{L^2(\Omega)}.
\end{aligned}$$

■

When we combine Equation (32) and Lemma 1 using the triangle inequality, we get an estimate of the approximation error:

Lemma 2 *If $v \in \mathcal{H}$, then*

$$\|v - \mathcal{I}\mathcal{P}v\|_{L^2(\Omega)} \leq \frac{\sqrt{2\pi}}{M^r} \|\partial_x^r v\|_{L^2(\Omega)} + \sqrt{\frac{2}{\pi^3}}\, h^2 \left\|\partial_z^2 v\right\|_{L^2(\Omega)}. \tag{34}$$

Proof: The triangle inequality gives

$$\|v - \mathcal{I}\mathcal{P}v\|_{L^2(\Omega)} \leq \|v - \mathcal{P}v\|_{L^2(\Omega)} + \|\mathcal{P}v - \mathcal{I}\mathcal{P}v\|_{L^2(\Omega)}.$$

The desired result follows from the estimates (32) and (33). ■

(Canuto, Maday, and Quarteroni [1] obtain a comparable estimate.)

6 Error Analysis of the FSM

We now estimate the difference between the exact solution $u^k(x,z)$ of Problem (1) and the approximate solution $\tilde{u}^k(x,z)$ generated by the FSM. We begin by defining three error components:

$$e^k := u^k - \tilde{u}^k \tag{35}$$

$$\eta^k := u^k - \mathcal{I}\mathcal{P}u^k \tag{36}$$

$$\xi^k := \mathcal{I}\mathcal{P}u^k - \tilde{u}^k. \tag{37}$$

The objective is to estimate $\left\|e^k\right\|_{L^2(\Omega)}$. Since $e^k = \eta^k + \xi^k$, the triangle inequality yields

$$\left\|e^k\right\|_{L^2(\Omega)} \leq \left\|\eta^k\right\|_{L^2(\Omega)} + \left\|\xi^k\right\|_{L^2(\Omega)}. \tag{38}$$

Lemma 2 provides an estimate for $\left\|\eta^k\right\|_{L^2(\Omega)}$, so an estimate for $\left\|\xi^k\right\|_{L^2(\Omega)}$ will suffice to bound $\left\|e^k\right\|_{L^2(\Omega)}$.

Our development proceeds by the following plan: We first derive an equation using ξ^k as the test function in the fully discretized weak formulation, Equation (22). We then obtain estimates for individual terms in this equation. Finally, we apply a discrete form of Gronwall's lemma to yield the desired estimate for $\|\xi^k\|_{L^2(\Omega)}$.

We start by restricting the weight function w to $\tilde{\mathcal{H}}$ in Equation (20) and subtract Equation (22) from it. We also add the quantity

$$\left\langle S\frac{u^k - u^{k-1}}{\tau}, w\right\rangle$$

to both sides of the resulting sum. It follows that, for all test functions $w \in \tilde{\mathcal{H}}$ and all time levels $k \in (0, T/\tau]$,

$$\left\langle S\frac{e^k - e^{k-1}}{\tau}, w\right\rangle + \left\langle K_x\partial_x e^k, \partial_x w\right\rangle + \left\langle K_z\partial_z e^k, \partial_z w\right\rangle$$

$$= \left\langle S\left(\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k\right), w\right\rangle. \tag{39}$$

Since $e^k = \eta^k + \xi^k$, we may rearrange Equation (39) to get

$$\left\langle S\frac{\xi^k - \xi^{k-1}}{\tau}, w\right\rangle + \left\langle K_x\partial_x \xi^k, \partial_x w\right\rangle + \left\langle K_z\partial_z \xi^k, \partial_z w\right\rangle$$

$$= \left\langle S\left(\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k\right), w\right\rangle - \left\langle S\frac{\eta^k - \eta^{k-1}}{\tau}, w\right\rangle$$

$$- \left\langle K_x \partial_x \eta^k, \partial_x w\right\rangle - \left\langle K_z \partial_z \eta^k, \partial_z w\right\rangle. \tag{40}$$

Setting $w = \xi^k$ and multiplying through by τ yields

$$\begin{aligned}
\left\langle S\xi^k, \xi^k\right\rangle \; &- \left\langle S\xi^{k-1}, \xi^k\right\rangle + \tau\left\langle K_x \partial_x \xi^k, \partial_x \xi^k\right\rangle + \tau\left\langle K_z \partial_z \xi^k, \partial_z \xi^k\right\rangle \\
&= \tau\left\langle S\left(\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k\right), \xi^k\right\rangle - \left\langle S\left(\eta^k - \eta^{k-1}\right), \xi^k\right\rangle \\
&\quad - \tau\left\langle K_x \partial_x \eta^k, \partial_x \xi^k\right\rangle - \tau\left\langle K_z \partial_z \eta^k, \partial_z \xi^k\right\rangle \\
&\leq \tau\left\langle S\left(\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k\right), \xi^k\right\rangle - \left\langle S\left(\eta^k - \eta^{k-1}\right), \xi^k\right\rangle \\
&\quad + \tau\left|\left\langle K_x \partial_x \eta^k, \partial_x \xi^k\right\rangle\right| + \tau\left|\left\langle K_z \partial_z \eta^k, \partial_z \xi^k\right\rangle\right|.
\end{aligned} \tag{41}$$

We now analyze individual terms in Equation (41), beginning with $\left\langle S\xi^{k-1}, \xi^k\right\rangle$. The inequality $2\langle a, b\rangle \leq \langle a, a\rangle + \langle b, b\rangle$ and the assumption that $0 < S$ imply that

$$\left|\left\langle S\xi^{k-1}, \xi^k\right\rangle\right| \leq \frac{1}{2}\left\langle S\xi^{k-1}, \xi^{k-1}\right\rangle + \frac{1}{2}\left\langle S\xi^k, \xi^k\right\rangle. \tag{42}$$

Next we obtain an estimate for $\tau\left\langle K_x \partial_x \eta^k, \partial_x \xi^k\right\rangle$. Using the inequality $2\langle a, b\rangle \leq \langle a, a\rangle + \langle b, b\rangle$, the definition of η, and the assumption that $0 < K_x \leq K$, we find that

$$\begin{aligned}
\left\langle K_x \partial_x \eta^k, \partial_x \xi^k\right\rangle \; &\leq \frac{1}{2}\left\langle K_x\left[\partial_x u^k - \mathcal{IP}(\partial_x u^k)\right], \partial_x u^k - \mathcal{IP}(\partial_x u^k)\right\rangle \\
&\quad + \frac{1}{2}\langle K_x \partial_x \xi^k, \partial_x \xi^k\rangle \\
&\leq \frac{K}{2}\left\|\partial_x u^k - \mathcal{IP}(\partial_x u^k)\right\|^2_{L^2(\Omega)} + \frac{1}{2}\langle K_x \partial_x \xi^k, \partial_x \xi^k\rangle.
\end{aligned} \tag{43}$$

Applying Lemma 2 then yields

$$\begin{aligned}
\left\langle K_x \partial_x \eta^k, \partial_x \xi^k\right\rangle \; &\leq \frac{K}{2}\left(\frac{\sqrt{2\pi}}{M^r}\left\|\partial_x^{r+1} u^k\right\|_{L^2(\Omega)} + h^2\sqrt{\frac{2}{\pi^3}}\left\|\partial_z^2 \partial_x u^k\right\|_{L^2(\Omega)}\right)^2 \\
&\quad + \frac{1}{2}\left\langle K_x \partial_x \xi^k, \partial_x \xi^k\right\rangle.
\end{aligned} \tag{44}$$

Since $t \in (0,T]$, this last inequality allows us to deduce that

$$\left\langle K_x \partial_x \eta^k, \partial_x \xi^k \right\rangle \leq \frac{1}{2}\left(M^{-r}\Gamma_1 + h^2\Gamma_2\right)^2 + \frac{1}{2}\left\langle K_x \partial_x \xi^k, \partial_x \xi^k \right\rangle, \tag{45}$$

where

$$\Gamma_1 := \sup_{t\in(0,T]} \sqrt{2\pi K}\left\|\partial_x^{r+1} u^k\right\|_{L^2(\Omega)} \tag{46}$$

$$\Gamma_2 := \sup_{t\in(0,T]} \sqrt{\frac{2K}{\pi^3}}\left\|\partial_z^2 \partial_x u^k\right\|_{L^2(\Omega)} \tag{47}$$

Although the term $\left\langle K_z \partial_z \eta^k, \partial_z \xi^k \right\rangle$ may be analyzed similarly, we use a different approach to show that it vanishes. For any node z_j on the z-axis,

$$\begin{aligned} \eta^k(x, z_j) &= \sum_{m=-\infty}^{\infty} \hat{u}_m(z_j)\omega_m(x) - \sum_{|m|\leq M} \hat{u}_m(z_j)\omega_m(x) \\ &= \sum_{|m|>M} \hat{u}_m(z_j)\omega_m(x). \end{aligned} \tag{48}$$

Using the expansion (23) of $\tilde{u}^k \in \tilde{\mathcal{H}}$, we write the quantity ξ^k as follows:

$$\begin{aligned} \xi^k(x,z) &= \mathcal{I}\mathcal{P}u^k(x,z) - \tilde{u}^k(x,z) \\ &= -\sum_{j=0}^{J} \sum_{|m|\leq M} [\Phi_{m,j}^k - \hat{u}_m^k(z_j)]\omega_m(x)\ell_j(z). \end{aligned}$$

Differentiation with respect to z yields

$$\partial_z \xi^k(x,z) = -\sum_{|m|\leq M} C_{m,j}^k \omega_m(x),$$

for any $z \in (z_{j-1}, z_j)$. Here,

$$C_{m,j}^k := \frac{\left[\Phi_{m,j}^k - \tilde{u}_m^k(z_j)\right] - \left[\Phi_{m,j-1}^k - \tilde{u}_m^k(z_{j-1})\right]}{z_j - z_{j-1}}.$$

Because the value of $K_z(z)$ is a constant $K_{z,j}$ for $z \in (z_{j-1}, z_j)$, the integral over Z in $\left\langle K_z \partial_z \eta^k, \partial_z \xi^k \right\rangle$ decomposes into a sum over the intervals formed by the finite-element grid. Using Equation (48), we get

$$\begin{aligned} \left\langle K_z \partial_z \eta^k, \partial_z \xi^k \right\rangle &= -\int_X \sum_{j=1}^{J} K_{2,j} \int_{z_{j-1}}^{z_j} \sum_{|m|\leq M} C_{m,j}^k \omega_m(x)\, \overline{\partial_z \eta^k}\, dz\, dx \\ &= -\int_X \sum_{j=1}^{J} K_{2,j} \sum_{|m|\leq M} C_{m,j}^k \omega_m(x) \int_{z_{j-1}}^{z_j} \overline{\partial_z \eta^k}\, dz\, dx \\ &= -\sum_{j=1}^{J} K_{2,j} \int_X \left(\sum_{|m|\leq M} C_{m,j}^k \omega_m(x)\right)\left(\sum_{|m|>M} d_{m,j}^k \omega_m(x)\right) dx, \end{aligned}$$

where

$$d_{m,j}^k := \overline{\hat{u}_m^k(z_j) - \hat{u}_m^k(z_{j-1})}.$$

The orthogonality property (25) then implies that

$$\left\langle K_z \partial_z \eta^k, \partial_z \xi^k \right\rangle = 0, \tag{49}$$

as claimed.

In addition, since K_x and K_z are positive, the third and fourth terms on the left side of Equation (41) are nonnegative:

$$0 \leq \tau \left\langle K_x \partial_x \xi^k, \partial_x \xi^k \right\rangle, \tag{50}$$

and

$$0 \leq \tau \left\langle K_z \partial_z \xi^k, \partial_z \xi^k \right\rangle. \tag{51}$$

Incorporating the estimates (42) through (51) into Equation (41) yields the inequality

$$\begin{aligned} &\frac{1}{2}\left\langle S\xi^k, \xi^k \right\rangle - \frac{1}{2}\left\langle S\xi^{k-1}, \xi^{k-1} \right\rangle \\ &\leq \tau \left| \left\langle S\left(\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k \right), \xi^k \right\rangle \right| + \left| \left\langle S\left(\eta^k - \eta^{k-1} \right), \xi^k \right\rangle \right| \\ &\quad + \frac{\tau}{2}\left(M^{-r}\Gamma_1 + h^2\Gamma_2 \right)^2. \end{aligned} \tag{52}$$

We now estimate the first two terms on the right side of Equation (52). Lemmas 3 through 5 concern the first term on the right, which involves the truncation error associated with the timestepping scheme.

Lemma 3 *Let $u^k \in \mathcal{H}$ for $0 \leq k \leq T/\tau$. Then for all $(x,z) \in \Omega$,*

$$\frac{u^k(x,z) - u^{k-1}(x,z)}{\tau} - \partial_t u^k(x,z) = \frac{-1}{\tau} \int_{(k-1)\tau}^{k\tau} \left[t - (k-1)\tau \right] \partial_t^2 u(x,z,t)\, dt. \tag{53}$$

Proof: The Fundamental Theorem of Calculus and integration by parts yield

$$\begin{aligned} u^k(x,z) - u^{k-1}(x,z) &= \int_{(k-1)\tau}^{k\tau} \partial_t u(x,z,t) dt \\ &= \left[t - (k-1)\tau \right] \partial_t u(x,z,t) \Big|_{t=(k-1)\tau}^{t=k\tau} \\ &\quad - \int_{(k-1)\tau}^{k\tau} \left[t - (k-1)\tau \right] \partial_t^2 u(x,z,t)\, dt \end{aligned}$$

$$= [k\tau - (k-1)\tau]\,\partial_t u^k(x,z) - \int_{(k-1)\tau}^{k\tau} [t-(k-1)\tau]\,\partial_t^2 u(x,z,t)\,dt.$$

The desired result follows upon rearrangement. ■

Lemma 4 *Let $u^k \in \mathcal{H}$ for $0 \le k \le T/\tau$. Then*

$$\left\| \partial_t u^k - \frac{u^k - u^{k-1}}{\tau} \right\|_{L^2(\Omega)}^2 \le \frac{\tau}{3} \int_{(k-1)\tau}^{k\tau} \left\| \partial_t^2 u \right\|_{L^2(\Omega)}^2 dt. \tag{54}$$

Proof: Lemma 3 and the Cauchy-Schwarz inequality imply that

$$\left\| \partial_t u^k - \frac{u^k - u^{k-1}}{\tau} \right\|_{L^2(\Omega)}^2$$

$$\le \frac{1}{\tau^2} \left\| \int_{(k-1)\tau}^{k\tau} [t-(k-1)\tau]\,\partial_t^2 u(x,z,t)dt \right\|_{L^2(\Omega)}^2$$

$$\le \frac{1}{\tau^2} \left\| \sqrt{\int_{(k-1)\tau}^{k\tau} [t-(k-1)\tau]^2\,dt} \sqrt{\int_{(k-1)\tau}^{k\tau} (\partial_t^2 u)^2 dt} \right\|_{L^2(\Omega)}^2$$

$$= \frac{1}{\tau^2} \int_{\Omega_2} \left[\int_{(k-1)\tau}^{k\tau} [t-(k-1)\tau]^2\,dt \int_{(k-1)\tau}^{k\tau} (\partial_t^2 u)^2 dt \right] dx\,dz$$

$$= \frac{1}{\tau^2} \int_{\Omega_2} \frac{\tau^3}{3} \int_{(k-1)\tau}^{k\tau} (\partial_t^2 u)^2 dt\,dx\,dz$$

$$= \frac{\tau}{3} \int_{(k-1)\tau}^{k\tau} \left\| \partial_t^2 u \right\|_{L^2(\Omega)}^2 dt.$$

The last step follows from Fubini's theorem. ■

Lemma 5 *Let $u^k \in \mathcal{H}$ for $0 \le k \le T/\tau$. Then*

$$\tau \left| \left\langle S\left(\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k \right), \xi^k \right\rangle \right| \le \frac{S^* \tau^2}{6} \int_{(k-1)\tau}^{k\tau} \left\| \partial_t^2 u \right\|_{L^2(\Omega)}^2 dt + \frac{S^* \tau}{2} \left\langle \xi^k, \xi^k \right\rangle. \tag{55}$$

Proof: The assumption that $0 < S(z) \le S^*$ and the inequality $2\langle a,b\rangle \le \langle a,a\rangle + \langle b,b\rangle$ imply that

$$\tau \left| \left\langle S\left[\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k \right], \xi^k \right\rangle \right| \le S^* \tau \left| \left\langle \left[\frac{u^k - u^{k-1}}{\tau} - \partial_t u^k \right], \xi^k \right\rangle \right|$$

$$\le \frac{S^* \tau}{2} \left\| \frac{u^k - u^{k-1}}{\tau} - \partial_t u^k \right\|_{L^2(\Omega)}^2 + \frac{S^* \tau}{2} \left\langle \xi^k, \xi^k \right\rangle.$$

The desired result follows from Lemma 4. ■

We now analyze the second term on the right side of Equation (52).

Lemma 6 . *If η^k and ξ^k are as defined in Equations (36) and (37), then*

$$\left|\left\langle S\left(\eta^k-\eta^{k-1}\right),\xi^k\right\rangle\right| \le \frac{S^*}{2}\int_{(k-1)\tau}^{k\tau}\|\partial_t\eta\|^2_{L^2(\Omega)}dt+\frac{S^*\tau}{2}\left\langle\xi^k,\xi^k\right\rangle. \tag{56}$$

Proof: The Cauchy-Schwarz inequality, the assumption that $S(z)\le S^*$, and the inequality $2\langle a,b\rangle \le \langle a,a\rangle+\langle b,b\rangle$ yield

$$\begin{aligned}
\left|\left\langle S\left(\eta^k-\eta^{k-1}\right),\xi^k\right\rangle\right| &= \left|\left\langle S\int_{(k-1)\tau}^{k\tau}\partial_t\eta(x,z,t)dt,\xi^k\right\rangle\right| \\
&\le \left\langle S\sqrt{\int_{(k-1)\tau}^{k\tau}dt}\sqrt{\int_{(k-1)\tau)}^{k\tau}(\partial_t\eta)^2dt},|\xi^k|\right\rangle \\
&= \left\langle S\sqrt{\int_{(k-1)\tau}^{k\tau}(\partial_t\eta)^2\,dt},\sqrt{\tau}|\xi^k|\right\rangle \\
&\le \frac{S^*}{2}\int_\Omega\int_{(k-1)\tau}^{k\tau}(\partial_t\eta)^2\,dt\,dx\,dz+\frac{S^*\tau}{2}\left\langle\xi^k,\xi^k\right\rangle \\
\text{(Fubini's Theorem)} \quad &= \frac{S^*}{2}\int_{(k-1)\tau}^{k\tau}\|\partial_t\eta\|^2_{L^2(\Omega)}dt+\frac{S^*\tau}{2}\left\langle\xi^k,\xi^k\right\rangle. \quad ■
\end{aligned}$$

Application of Lemma 5 and Lemma 6 to Equation (52) now produces the inequality

$$\begin{aligned}
&\frac{1}{2}\left\langle S\xi^k,\xi^k\right\rangle-\frac{1}{2}\left\langle S\xi^{k-1},\xi^{k-1}\right\rangle \\
&\le S^*\tau\left\langle\xi^k,\xi^k\right\rangle+\frac{S^*\tau^2}{6}\int_{(k-1)\tau}^{k\tau}\left\|\partial_t^2u\right\|^2_{L^2(\Omega)}dt \\
&\quad+\frac{S^*}{2}\int_{(k-1)\tau}^{k\tau}\|\partial_t\eta\|^2_{L^2(\Omega)}dt+\frac{\tau}{2}\left(M^{-r}\Gamma_1+h^2\Gamma_2\right)^2.
\end{aligned} \tag{57}$$

We now make three observations to prepare for the application of the discrete Gronwall lemma. First, if p is any positive integer such that $p\tau\le T$ and if we sum Equation (57) from $k=1$ through $k=p$, then we obtain the inequality

$$\begin{aligned}
&\frac{1}{2}\left(\langle S\xi^p,\xi^p\rangle-\left\langle S\xi^0,\xi^0\right\rangle\right) \\
&\le S^*\tau\sum_{k=1}^{p}\left\langle\xi^k,\xi^k\right\rangle+\frac{S^*\tau^2}{6}\int_0^{pk}\left\|\partial_t^2u\right\|^2_{L^2(\Omega)}dt \\
&\quad+\frac{S^*}{2}\int_0^{pk}\|\partial_t\eta\|^2_{L^2(\Omega)}dt+\sum_{k=1}^{p}\frac{\tau}{2}\left(M^{-r}\Gamma_1+h^2\Gamma_2\right)^2.
\end{aligned} \tag{58}$$

Let us use the numerical initial condition $\tilde{u}^0 = \mathcal{I}\mathcal{P}u^0$, so that $\xi^0 = 0$ and thus $\left\langle \xi^0, \xi^0 \right\rangle = 0$ and $\left\langle S\xi^0, \xi^0 \right\rangle = 0$. In this case, we can multiply Equation (58) by 2 and extend the integrations to the full time interval $(0, T]$ to get

$$\langle S\xi^p, \xi^p \rangle \leq 2S^*\tau \sum_{k=0}^{p} \left\langle \xi^k, \xi^k \right\rangle + \beta', \tag{59}$$

where

$$\beta' := \frac{S^*\tau^2}{3} \int_0^T \left\| \partial_t^2 u \right\|_{L^2(\Omega)}^2 dt + S^* \int_0^T \|\partial_t \eta\|_{L^2(\Omega)}^2 dt + T\left(M^{-r}\Gamma_1 + h^2\Gamma_2\right)^2. \tag{60}$$

Second, the Fourier series for $\partial_t u$ may be written in terms of the Fourier coefficients of u. In particular, if $u(\,\cdot\,,\,\cdot\,,t) \in \mathcal{H}$ for $t \in (0, T]$, then

$$u(x, z, t) = \sum_{m=-\infty}^{\infty} \hat{u}_m(z, t)\omega_m(x),$$

and the series converges uniformly. Therefore,

$$\partial_t u(x, z, t) = \sum_{m=-\infty}^{\infty} \partial_t \hat{u}_m(z, t)\omega_m(x).$$

Thus we can estimate the term in (60) involving $\partial_t \eta$ using Lemma 2:

$$\|\partial_t \eta\|_{L^2(\Omega)} \leq \frac{\sqrt{2\pi}}{M^r} \|\partial_t \partial_x^r u\|_{L^2(\Omega)} + \sqrt{\frac{2}{\pi^3}} h^2 \left\| \partial_t \partial_z^2 u \right\|_{L^2(\Omega)}. \tag{61}$$

Third, utilizing the assumption that $0 < s \leq S$, we can move the last term of the sum in Equation (59) to the left side, getting

$$(s - 2S^*\tau)\langle \xi^p, \xi^p \rangle \leq 2S^*\tau \sum_{k=0}^{p-1} \left\langle \xi^k, \xi^k \right\rangle + \beta'. \tag{62}$$

Let us choose the time step τ small enough so that $s - 2S^*\tau > 0$. Defining

$$\lambda := \frac{2S^*\tau}{s - 2S^*\tau} \tag{63}$$

and

$$\beta := \frac{\beta'}{s - 2S^*\tau}, \tag{64}$$

we obtain

$$\langle \xi^p, \xi^p \rangle \leq \lambda\tau \sum_{k=0}^{p-1} \left\langle \xi^k, \xi^k \right\rangle + \beta. \tag{65}$$

We now use a discrete form of Gronwall's lemma (reviewed in the Appendix) to establish the estimate on $\|\xi^k\|_{L^2(\Omega)}$. If p is any integer such that $p\tau \le T$, then

$$\left\|\xi^k\right\|^2_{L^2(\Omega)} \le \beta e^{\lambda T}, \qquad \text{for } k = 0, 1, \ldots, P. \tag{66}$$

Finally, the main error estimate for the FSM results when we use the estimate (66) in the triangle inequality (38):

Theorem 1 *(FSM Error). Let $u(\,\cdot\,,\,\cdot\,,t) \in \mathcal{H}$ satisfy the initial-boundary-value problem (1) for $t \in (0,T)$. Let $\{\tilde{u}^k\}$ be a sequence of functions in $\tilde{\mathcal{H}}$ determined using the FSM, Equation (22). If p is any integer such that $p\tau \le T$, then, for time levels $k = 0, 1, \ldots, p$,*

$$\begin{aligned}\left\|\tilde{u}^k - u^k\right\|_{L^2(\Omega)} \;&\le\; \frac{\sqrt{2\pi}}{M^r}\left\|\partial_x^r u^k\right\|_{L^2(\Omega)} \\ &\quad + \sqrt{\frac{2}{\pi^3}}\, h^2 \left\|\partial_z^2 u^k\right\|_{L^2(\Omega)} + \sqrt{\beta e^{\lambda T}}.\end{aligned} \tag{67}$$

Here,

$$\lambda \;:=\; \frac{2S^*\tau}{s - 2S^*\tau},$$

$$\beta \;:=\; \frac{\beta'}{s - 2S^*\tau},$$

$$\begin{aligned}\beta' \;&:=\; S^* \int_0^T \|\partial_t \eta\|^2_{L^2(\Omega)} dt + \frac{S^*\tau^2}{3}\int_0^T \left\|\partial_t^2 u\right\|^2_{L^2(\Omega)} dt + T\left(M^{-r}\Gamma_1 + h^2\Gamma_2\right)^2 \\ &\le\; S^* \int_0^T \left[\frac{\sqrt{2\pi}}{M^r}\|\partial_t \partial_x^r u\|^2_{L^2(\Omega)} + \sqrt{\frac{2}{\pi^3}}\, h^2 \left\|\partial_t \partial_z^2 u\right\|^2_{L^2(\Omega)}\right]^2 dt + \frac{S^*\tau^2}{3}\int_0^T \left\|\partial_t^2 u\right\|^2_{L^2(\Omega)} dt \\ &\quad + T\left(M^{-r} \sup_{t\in(0,T]} \sqrt{2K\pi}\left\|\partial_x^{r+1} u\right\|_{L^2(\Omega)} + h^2 \sup_{t\in(0,T]} \sqrt{\frac{2K}{\pi^3}}\left\|\partial_z^2 \partial_x u\right\|_{L^2(\Omega)}\right)^2\end{aligned}$$

This theorem asserts that the L^2 error in the backward-Euler FSM applied to the problem (1) is $\mathcal{O}(M^{-r} + h^2 + \tau)$. Here, r is the degree of smoothness of the exact solution in the x-direction. The order of the estimate, M^{-r} in the Fourier direction and h^2 in the finite-element direction, remains unchanged if we scale the spatial domain to a more general rectangle $\Omega = (a,b) \times (c,d)$. In particular, the FSM converges in the sense that $\left\|\tilde{u}^k - u^k\right\|_{L^2(\Omega)} \to 0$ as $\max\{h, M^{-1}, \tau\} \to 0$.

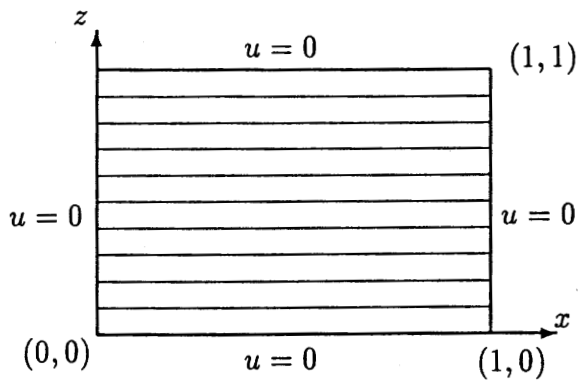

Figure 4: Partition of domain into strips.

7 Computational Results

We test Theorem 1 computationally with a dimensionless quenching problem from the classical theory of heat transfer. We solve the following model problem on $\Omega = (0,1) \times (0,1)$ with the FSM:

$$\left.\begin{aligned} \partial_t u = K(\partial_x^2 u + \partial_z^2 u), \quad (x,z) \in \Omega,\, t \in (0,T], \\ u(x,z,0) = u^0 = 1, \quad (x,z) \in \Omega, \\ u(x,z,t) = 0, \quad (x,z) \in \partial\Omega, \quad t > 0. \end{aligned}\right\} \tag{68}$$

We use a uniform finite-element grid on Z, the mesh size of which varies among different tests, as discussed below. Figure 7 depicts the decomposition of the domain Ω into strips.

The exact solution to the problem (68) has a double Fourier series:

$$u(x,z,k\tau) = \sum_{m=1,3,5\ldots}^{\infty} \frac{4e^{-Kk\tau(\pi m)^2}}{m\pi} \sin(m\pi x) \sum_{n=1,3,5\ldots}^{\infty} \frac{4e^{-Kk\tau(\pi n)^2}}{n\pi} \sin(n\pi z). \tag{69}$$

The symmetry of the problem implies that only odd-numbered Fourier modes have nonzero amplitudes This solution is continuously differentiable to all orders in both x and z for $t > 0$ [4, Chapter 4].

Using the exact solution, we compute the error term by term as follows:

$$\left\|u^k - \tilde{u}^k\right\|^2_{L^2(\Omega)} = \left\|u^k\right\|^2_{L^2(\Omega)} - 2\left\langle u^k, \tilde{u}^k\right\rangle + \left\|\tilde{u}^k\right\|^2_{L^2(\Omega)}. \tag{70}$$

Orthogonality implies that the first term on the right side of this expansion collapses to the infinite sum

$$\left\|u^k\right\|^2_{L^2(\Omega)} = \sum_{n=1,3,5\ldots}^{\infty} \left[\frac{e^{-Kk\tau(\pi n)^2}}{n}\right]^2 \approx \sum_{n=1,3,5\ldots}^{N} \left[\frac{e^{-Kk\tau(\pi n)^2}}{n}\right]^2. \tag{71}$$

Here, N is a positive integer at which we truncate the series in the computations. To determine an appropriate value of N, we observe that $e^{-Kk\tau(\pi n)^2}$ decays quickly with n. We pick N such that $e^{-Kk\tau(\pi N)^2} \leq 10^{-10}$, or

$$N \geq \sqrt{\frac{10\log_e 10}{\pi^2 K k\tau}}, \tag{72}$$

for $1 \leq k \leq T/\tau$. We also use the same value of N for the truncated series that arises from the second term on the right side of (70):

$$\left\langle u^k, \tilde{u}^k\right\rangle \approx \frac{8}{\pi^2} \sum_{m=1,3,5\ldots}^{M} \left\{\sum_{j=1}^{J-1} \frac{e^{-Kk\tau(\pi m)^2}}{m} \Phi^k_{m,j} \left[\sum_{n=1,3,5\ldots}^{N} \frac{e^{-Kk\tau(\pi n)^2}}{n} \left(\frac{1}{n\pi}\right)^2 \frac{c_j}{h}\right]\right\}, \tag{73}$$

where $c_j := 2\sin(n\pi z_j) - \sin(n\pi z_{j-1}) - \sin(n\pi z_{j+1})$. We use all the terms of $\tilde{u}^k$ to calculate its norm. Owing to orthogonality, mixed products of modes do not survive integration, and we obtain

$$\left\|\tilde{u}^k\right\|^2_{L^2(\Omega)} = \frac{1}{2} \sum_{m=1,3,5\ldots}^{M} \sum_{j=1}^{J-1} h \left(\frac{\Phi_{m,j-1}\Phi_{m,j}}{6} + \frac{2\Phi^2_{m,j}}{3} + \frac{\Phi_{m,j+1)}\Phi_{m,j}}{6}\right). \tag{74}$$

Since $u^k \in H^{r,2}_p(\Omega)$ for all $r \geq 1$, Theorem 1 indicates that $\left\|u^k - \tilde{u}^k\right\|_{L^2(\Omega)} = \mathcal{O}(h^2 + M^{-r} + \tau)$ for all $r \geq 1$. The idea behind the following tests is to generate numerical solutions using an extremely small time step τ and to plot $\log\left\|u^k - \tilde{u}^k\right\|_{L^2(\Omega)}$ versus $\log h$ and $\log M^{-1}$. The slopes of the resulting plots should confirm Theorem 1.

The first computational test considers the effect of varying the finite-element mesh size h. The parameters for this test are summarized in Table 7. We use $K = .02$ and a final time $T = 0.5$. To make the timestepping error negligible, we choose $\tau = 0.0005$. To render the $\mathcal{O}(M^{-r})$ error terms negligible, we choose $M = 65$ for the total number of Fourier modes. However, only the 32 odd-numbered modes contribute to the expansion of $\tilde{u}^k$. With this fixed value of M, we vary h from 1/2 to 1/28. Figure 7 depicts the results. The graph indicates that, as h shrinks, the FSM error is indeed $\mathcal{O}(h^2)$.

Table 1: Parameter Summary for Test 1 (varying h).

Diffusivity:	K	=	0.02
Output time:	T	=	0.5
Time step:	τ	=	0.0005

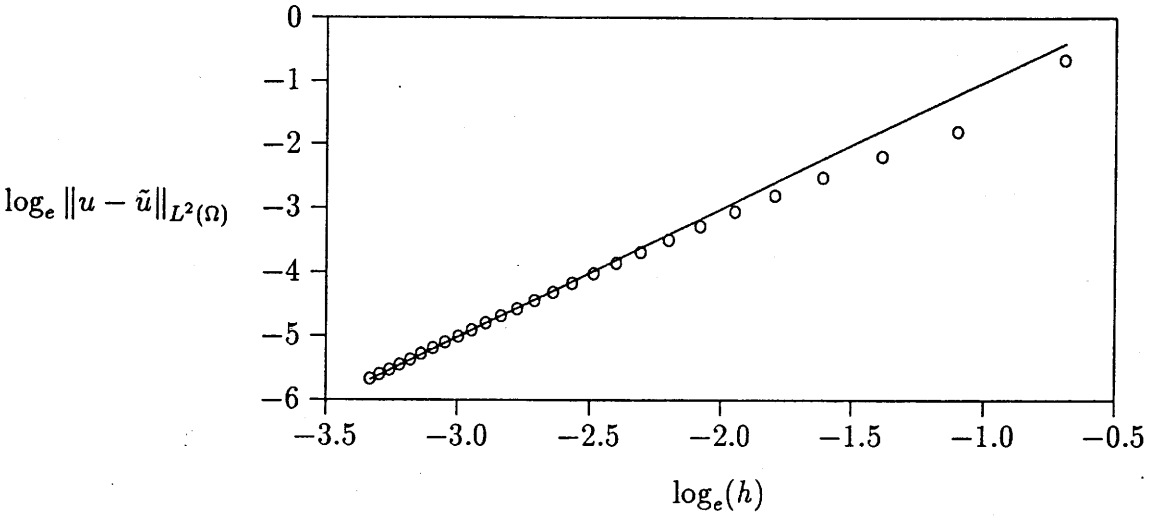

Figure 5: Convergence plot for changing mesh size h.

Next we examine the effect of varying the total number M of Fourier modes. In this test problem, the Fourier coefficients decay rapidly as t increases. While this phenomenon is beneficial in computational practice, in numerical testing it requires us to look at early solutions to distinguish the FSM error from errors associated with finite machine precision. Table 7 summarizes the parameters of this test. We present results for $t = 0.03, 0.1,$ and 0.3. To render the $\mathcal{O}(h^2)$ portion of the error negligible, we fix $h = 0.002$.

The efficiency of the FSM becomes apparent in computations of this magnitude. At each time level, the problem decouples into 32 separate tridiagonal problems, each of which determines 499 values $\Phi^k_{m,j}$, $j = 1, 2, \ldots, 499$, for a distinct mode number m. Also calculated for each mode, using results of the lower-numbered modes, is the error, $\left\| u^k - \tilde{u}^k \right\|_{L^2(\Omega)}$. To exploit the increasing smoothness of the solution in time, we increase the size of the time step, τ, as the calculations progress. Specifically, τ ranges from 0.0001 initially to a maximum value of 0.0025, which is still small enough to keep the timestepping error negligible. Figure 7 shows a convergence plot of the errors computed for the three output times. The plot indicates convergence beyond all orders in r, until the machine's precision limits have been reached. This result is consistent with the fact that the exact solution in this test problem is smooth in x, belonging to $H^{r,2}_p(\Omega)$ for all $r \geq 1$.

These computational tests verify that it is possible in practice to obtain $\mathcal{O}(M^{-r}+h^2)$ errors using the FSM, in accordance with Theorem 1.

Table 2: Parameter Summary for Test 2 (varying M).

Diffusivity: $K = 0.02$		
Time Data:		
number of steps	time step τ	total time t
100	0.00010	0.01
80	0.00025	0.03*
80	0.00025	0.05
100	0.00050	0.10*
80	0.00250	0.30*

* results included in Figure 7.

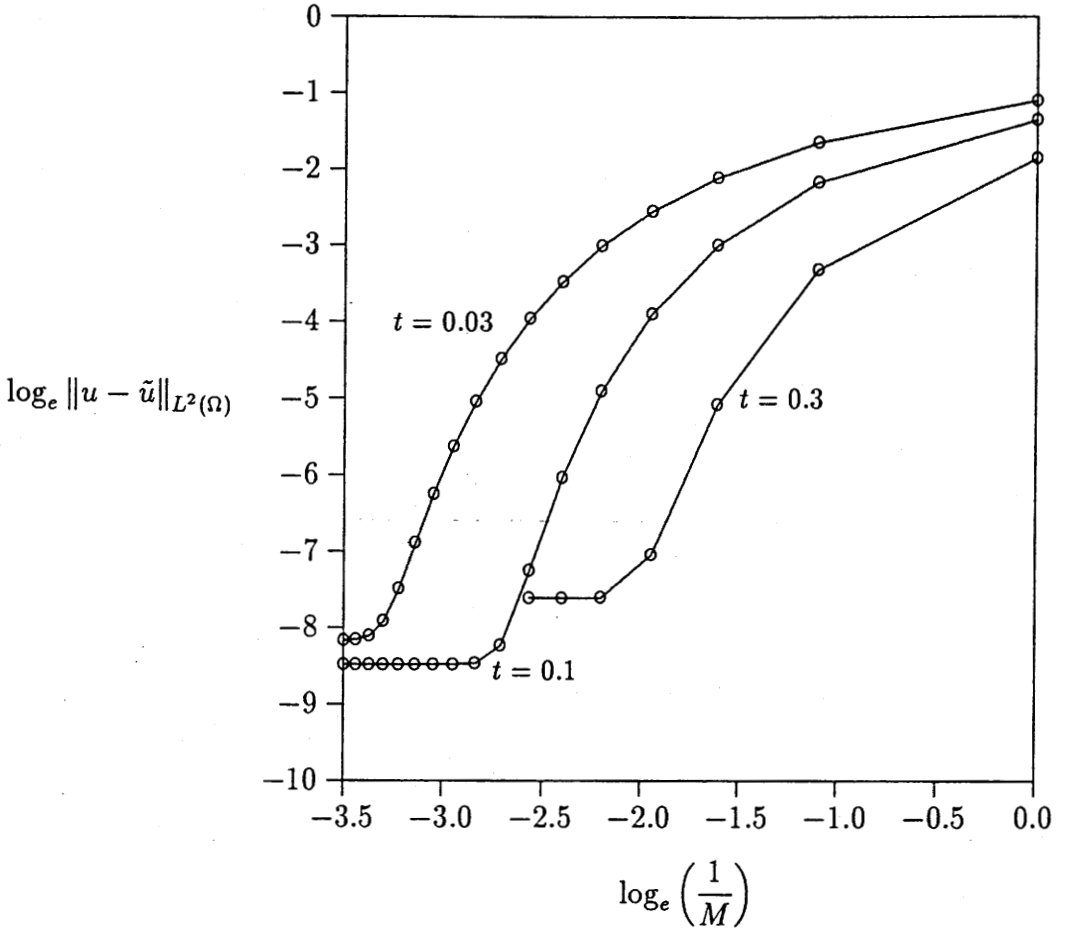

Figure 6: Convergence plot for changing number M of Fourier modes

8 Extension to the Finite-Layer Method

It is possible to extend the error estimate of Theorem 1 to problems on three-dimensional domains $\Omega = X \times Y \times Z$ in a straightforward way. We now sketch this extension. By analogy with the FSM, we consider problems that are geometrically regular and periodic in x and y. Consider the following initial-boundary-value problem:

$$\left.\begin{aligned} S\partial_t u - K_x\partial_x^2 u - K_y\partial_y^2 u - \partial_z(K_z\partial_z u) = f, \quad \text{on} \quad \Omega \times (0,T] \\ u(x,y,z,t) = 0, \quad (x,y,z) \in \partial\Omega, \quad t \in [0,T] \\ u(x,y,z,0) = u^0(x,y,z), \quad (x,y,z) \in \Omega. \end{aligned}\right\} \tag{75}$$

Here, the coefficients S, K_x, K_y, and K_z vary as functions of z and obey bounds similar to those given in the inequalities (2), (3), and (4).

Discretization in the finite-layer method is analogous to that used in the FSM. To discretize the problem in the z-direction, we again use the piecewise linear basis functions $\{\ell_j(z)\}_{j=1}^{J-1}$. For the x- and y-directions, we use truncated Fourier series. The exponential basis functions in this case have the following form:

$$\omega_{m,n}(x,y) := \omega_m(x)\,\omega_n(y) = e^{i(mx+ny)}. \tag{76}$$

By orthogonality, we have

$$\frac{1}{4\pi^2}\int_{X\times Y} \omega_{nm}\overline{\omega_{m',n'}}dx = \begin{cases} 0 & \text{for } m \neq m' \text{ or } n \neq n' \\ 1 & \text{for } m = m' \text{ and } n = n'. \end{cases} \tag{77}$$

We again use backward differences to approximate time derivatives.

The appropriate nonisotropic Hilbert space $H_p^{r,q,2}(\Omega)$ in this setting contains all $v \in L^2(\Omega)$ such that

$$\int_{X\times Y} \sum_{a=0}^{2} \|\partial_z^a v\|_{L^2(Z)}^2 dx\, dy < \infty, \tag{78}$$

$$\int_{Z\times Y} \sum_{a=0}^{r} \|\partial_x^a v\|_{L^2(X)}^2 dz\, dy < \infty, \tag{79}$$

and

$$\int_{Z\times X} \sum_{a=0}^{q} \left\|\partial_y^a v\right\|_{L^2(Y)}^2 dz\, dx < \infty. \tag{80}$$

By analogy with the FSM, the space $\mathcal{H}$ contains all functions $v \in H_p^{r,q,2}(\Omega)$ for which $\partial_z^2\partial_x v, \partial_z^2\partial_y v, \partial_x^r\partial_y^q v, \partial_x^{r+1}\partial_y^q v, \partial_x^r\partial_y^{q+1} v \in L^2(\Omega)$ and v vanishes on $\partial\Omega$. The trial

space $\tilde{\mathcal{H}}$ is the span of the tensor-product basis functions $\ell_j(z)\omega_{m,n}(x,y)$, where $j = 1,2,\ldots,J-1$, $|m| \le M$, and $|n| \le N$. The interpolation operator $\mathcal{I}$ is analogous to that used in the analysis of the FSM. The projection $\mathcal{P}$ in this context truncates double Fourier series:

$$(\mathcal{P}v)(x,y) := \sum_{|m|\le M} \sum_{|n|\le N} \hat{v}_{m,n}\omega_{m,n}(x,y), \tag{81}$$

where

$$\hat{v}_{m,n} := \frac{1}{4\pi^2}\int_{X\times Y} v(x,y)\omega_{m,n}(x,y)\,dx\,dy. \tag{82}$$

When we extend $\mathcal{P}$ to functions $v \in \mathcal{H}$, we have a projection error estimate comparable to Equation (32).

Lemma 7 . *If $v \in \mathcal{H}$, then*

$$\|v - \mathcal{P}v\|_{L^2(\Omega)} \le \frac{2\pi}{M^r N^q}\left\|\partial_x^r\partial_y^q v\right\|_{L^2(\Omega)}. \tag{83}$$

Proof:

$$\begin{aligned}
\|v-\mathcal{P}v\|^2_{L^2(\Omega)} &= \int_Z\int_Y\int_X |v(x,y,z)-\mathcal{P}v(x,y,z)|^2\,dx\,dy\,dz \\
&= \int_Z \|v(\,\cdot\,,z)-\mathcal{P}v(\,\cdot\,,z)\|^2_{L^2(X\times Y)}dz \\
\text{(Parseval equality)} \quad &= 4\pi^2\int_Z \sum_{|m|>M}\sum_{|n|>N} |\hat{v}_{m,n}(z)|^2\,dz \\
&= 4\pi^2\int_Z \sum_{|m|>M}\sum_{|n|>N} \left(m^2\right)^{-r+r}\left(n^2\right)^{-q+q}|\hat{v}_{m,n}(z)|^2\,dz \\
&\le \frac{4\pi^2}{M^{2r}N^{2q}}\int_Z \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} |m^r n^q \hat{v}_{m,n}(z)|^2\,dz \\
&= \frac{4\pi^2}{M^{2r}N^{2q}}\int_Z \sum_{m=-\infty}^{\infty}\sum_{n=-\infty}^{\infty} \left|\widehat{(\partial_x^r\partial_y^q v)}_{m,n}(z)\right|^2 dz \\
\text{(Parseval equality)} \quad &= \frac{4\pi^2}{M^{2r}N^{2q}}\int_Z \left\|\partial_x^r\partial_y^q v\right\|^2_{L^2(X\times Y)}dz \\
&= \frac{4\pi^2}{M^{2r}N^{2q}}\left\|\partial_x^r\partial_y^q v\right\|^2_{L^2(\Omega)}. \qquad \blacksquare
\end{aligned}$$

We now state the approximation error estimate corresponding to Lemma 1 and Lemma 2. The proofs of the next two lemmas are identical to those of the earlier lemmas, except for the following changes: Integrations over X become integrations

over $X \times Y$; the basis function $\omega_m(x)$ is replaced by $\omega_{m,n}(x,y)$; and the sums over m are replaced by double sums over m and n.

Lemma 8 . *If $v \in \mathcal{H}$, then*

$$\|\mathcal{P}v - \mathcal{I}\mathcal{P}v\|_{L^2(\Omega)} \leq \frac{2h^2}{\pi}\left\|\partial_z^2 v\right\|_{L^2(\Omega)}. \tag{84}$$

Lemma 9 *If $v \in \mathcal{H}$, then*

$$\|v - \mathcal{I}\mathcal{P}v\|_{L^2(\Omega)} \leq \frac{2\pi}{M^r N^q}\left\|\partial_x^r \partial_y^q v\right\|_{L^2(\Omega)} + \frac{2h^2}{\pi}\left\|\partial_z^2 v\right\|_{L^2(\Omega)}. \tag{85}$$

We obtain an error estimate for the finite-layer method by a sequence of arguments analogous to those leading to Theorem 1, incorporating the following changes:

1. Replace integration over X by integration over $X \times Y$.
2. Replace sums over m by double sums over m and n and use the respective truncation limits M and N where appropriate.
3. Manipulate the term $\left\langle K_y \partial_y \tilde{u}^k, \partial_y w \right\rangle$ in the error equation in a manner identical to that used for the term $\left\langle K_x \partial_x \tilde{u}^k, \partial_x w \right\rangle$ in the FSM analysis.

The following theorem results.

Theorem 2 *(Finite-Layer Error). Let $u^k \in \mathcal{H}$ denote the solution to the problem (75) at $t = k\tau$, and let Let $\tilde{u}^k \in \tilde{\mathcal{H}}$ be the corresponding solution to the finite-layer method. If p is any integer such that $p\tau \leq T$, then*

$$\begin{aligned}\left\|\tilde{u}^k - u^k\right\|_{L^2(\Omega)} \;\leq\; & \frac{2\pi}{M^r N^q}\left\|\partial_x^r \partial_y^q u\right\|_{L^2(\Omega)} \\ & + \frac{2h^2}{\pi}\left\|\partial_z^2 u\right\|_{L^2(\Omega)} + \sqrt{\beta e^{\lambda T}},\end{aligned} \tag{86}$$

where

$$\lambda = \frac{2S^*\tau}{s - 2S^*\tau},$$

$$\begin{aligned}
\beta \leq\ & \frac{S^*}{s - 2S^*\tau}\int_0^T \left[\frac{2\pi}{M^r N^q}\left\|\partial_t \partial_x^r \partial_y^q u\right\|^2_{L^2(\Omega)} + \frac{2h^2}{\pi}\left\|\partial_t \partial_z^2 u\right\|^2_{L^2(\Omega)}\right]^2 dt \\
& + \frac{S^*\tau^2}{3(s - 2S^*\tau)}\int_0^T \left\|\partial_t^2 u\right\|^2_{L^2(\Omega)} dt \\
& + \frac{TK}{s - 2S^*\tau}\left[\left(\sup_{t\in(0,T]} \frac{2\pi}{M^r N^q}\left\|\partial_x^{r+1}\partial_y^q u^k\right\|_{L^2(\Omega)} + \sup_{t\in(0,T]} \frac{2h^2}{\pi}\left\|\partial_z^2 \partial_x u^k\right\|_{L^2(\Omega)}\right)^2\right. \\
& \left. + \left(\sup_{t\in(0,T]} \frac{2\pi}{M^r N^q}\left\|\partial_x^r \partial_y^{q+1} u^k\right\|_{L^2(\Omega)} + \sup_{t\in(0,T]} \frac{2h^2}{\pi}\left\|\partial_z^2 \partial_y u^k\right\|_{L^2(\Omega)}\right)^2\right]
\end{aligned}$$

Thus the error is $\mathcal{O}(M^{-r} + N^{-q} + h^2 + \tau)$, in close analogy with the error estimate of Theorem 1.

Appendix

Projection Error

Lemma 10 . *Let r and M be positive integers, and let $v \in H_p^{r,2}(\Omega)$. Then*

$$\|v - \mathcal{P}v\|_{L^2(\Omega)} \leq \frac{\sqrt{2\pi}}{M^r} \|\partial_x^r v\|_{L^2(\Omega)}. \tag{87}$$

Proof: By definition,

$$\begin{aligned}
\|v - \mathcal{P}v\|_{L^2(\Omega)}^2 &= \int_Z \int_X |v(x,z) - (\mathcal{P}v)(x,z)|^2 \, dx \, dz \\
&= \int_Z \|v(\,\cdot\,,z) - (\mathcal{P}v)(\,\cdot\,,z)\|_{L^2(X)}^2 dz \\
\text{(Parseval equality)} \quad &= 2\pi \int_Z \sum_{m>M} |\hat{v}_m(z)|^2 \, dz \\
&= 2\pi \int_Z \sum_{m>M} \left(m^2\right)^{-r+r} |\hat{v}_m(z)|^2 \, dz \\
&\leq \frac{2\pi}{M^{2r}} \int_Z \sum_{m=-\infty}^{\infty} |m^r \hat{v}_m(z)|^2 \, dz \\
&= \frac{2\pi}{M^{2r}} \int_Z \sum_{m=-\infty}^{\infty} \left|\widehat{(\partial_x^r v)}_m(z)\right|^2 dz \\
\text{(Parseval equality)} \quad &= \frac{2\pi}{M^{2r}} \int_Z \|\partial_x^r v\|_{L^2(X)}^2 dz \\
&= \frac{2\pi}{M^{2r}} \|\partial_x^r v\|_{L^2(\Omega)}^2 \qquad \blacksquare
\end{aligned}$$

Discrete Form of Gronwall's Lemma

Lemma 11 *Suppose that the real sequence* $\{V_k\}_{k=0}^{P}$ *satisfies the inequality*

$$|V_k| \leq \beta + \lambda\tau \sum_{j=0}^{k-1} |V_j|, \qquad \text{for } k = 0, 1, \cdots, P;$$

where λ, β, *and* τ *are nonnegative real numbers. Then*

$$|V_k| \leq \beta e^{(\lambda P \tau}, \qquad \text{for } k = 0, 1, \cdots, P.$$

Proof: Define the sequence $\{Z_k\}_{k=0}^{P}$ by

$$Z_k = \beta + \lambda\tau \sum_{j=0}^{k} |V_j|.$$

The definition of Z_k and the inequality of the hypothesis imply that

$$Z_0 = \beta + \lambda\tau|V_0| \leq \beta + \lambda\tau\beta,$$

and

$$Z_j - Z_{j-1} = \lambda\tau|V_j| \leq \lambda\tau Z_{j-1}, \qquad \text{for } j = 1, 2, \cdots, P;$$

that is,

$$\begin{aligned} Z_0 &\leq (1 + \lambda\tau)\beta \\ Z_j &\leq (1 + \lambda\tau)Z_{j-1}, \qquad \text{for } j = 1, 2, \cdots, P. \end{aligned}$$

Apply the above result $k-1$ times. Since $(1+\lambda\tau)^k \leq e^{\lambda P\tau}$ for any integer k, $0 \leq k \leq P$, we have

$$Z_{k-1} \leq (1 + \lambda\tau)Z_{k-2} \leq \cdots \leq (1 + \lambda\tau)^{k-1} Z_0 \leq (1 + \lambda\tau)^k \beta \leq \beta e^{\lambda P\tau},$$

or,

$$\beta + \lambda\tau \sum_{j=0}^{k-1} |V_j| \leq \beta e^{(\lambda P\tau)}, \quad k = 1, 2, \cdots, P.$$

The inequality in the hypothesis implies the desired result. ■

References

[1] Claudio Canuto, Y. Maday, and Alfio Quarteroni. Analysis of the Combined Finite Element and Fourier Interpolation. *Numerische Mathematik*, **39**, 205-220, 1982.

[2] Claudio Canuto, M. Yousuff Hussaini, Alfio Quarteroni, and Thomas A. Zang. *Spectral Methods in Fluid Dynamics.* Springer-Verlag New York Inc., New York, 1988.

[3] Yau-kai Cheung. *Finite Strip Method in Structural Analysis.* Pergamon Press, Oxford, England, 1976.

[4] G. Folland. *Introduction to Partial Differential Equations.* Princeton University Press, Princeton, 1976.

[5] Huyakorn, Peter S., and George F. Pinder. *Computational Methods in Subsurface Flow.* New York: Academic Press, Inc., 1983.

[6] Erwin Kreyszig. *Introductory Functional Analysis with Applications.* John Wiley & Sons, New York, Wiley Classics Library edition, 1989.

[7] Jay A. Puckett and R. J. Schmidt. Finite strip method for groundwater modeling in a parallel computing environment. *Engineering Computation*, Vol.7, No. 2, Jun 1990.

[8] Karel Rektorys. *Variational Methods in Mathematics, Science and Engineering.* D. Reidel Publishing Company, Boston, second edition, 1980.

[9] H. L. Royden. *Real Analysis.* MacMillan Publishing Co., Inc, New York, second edition, 1968.

[10] James E. Slattery. *The Finite Strip Method in Groundwater Hydrology.* Master's thesis, Colorado State University, Fort Collins, Colorado, 1986.

[11] S. S. Smith, M. B. Allen, J. A. Puckett and T. Edgar. Three Dimensional Model of Multi-Well Field Using Finite Layer Methods. In *Proceedings of Eleventh Annual American Geophysical Union Hydrology Days*, pages 23-34, Hydrology Days Pulications, Fort Collins, 1991. Conference Location: Colorado State University, Fort Collins, Colorado. Conference Date: apr 2-4, 1991.

[12] S. S. Smith, M. B. Allen, J. A. Puckett and T. Edgar. The Finite-Layer Method for Groundwater Flow Models. *Water Resources Research*, 1992.

[13] Gilbert Strang and George J. Fix. *An Analysis of the Finite Element Method.* Prentice-Hall, Inc., New Jersey, 1973.

RESEARCH BRIEFS

Wyoming Water Resources Center

Technical RB92-03

MODELING GROUNDWATER FLOW AND CONTAMINANT TRANSPORT

Investigator Myron B. Allen, Department of Mathematics, University of Wyoming.

Purpose Computer models of groundwater flow and contaminant transport are important tools in designing aquifer cleanup schemes. Models also aid in the investigation of poorly understood aspects of underground flows. Examples include the spread of contaminants by random velocity variations and the effects of measurement uncertainty on model predictions. Both applications --- the practical and the theoretical --- require more efficiency and accuracy than standard models provide. This research focuses on better numerical methods for such models.

Methods Groundwater models are based on differential equations. The solutions to these equations give the water velocity, pressure (or head), and contaminant concentrations. For real aquifers, it is usually impossible to solve the equations exactly. Instead, engineers use complex computer codes to generate approximate solutions. Most codes employ finite differences or finite elements, which partition aquifers into cells and compute local mass and momentum balances. These discrete techniques require billions of arithmetic operations, taking hours or days to run on supercomputers.

Careful studies, involving the world's most thoroughly measured contamination sites, show that standard models are often too inaccurate to be realistic, even when good site data exist. Especially troublesome are heterogeneous sites, where rock properties vary over several orders of magnitude.

Two approaches help overcome these difficulties. First, proper discrete techniques can maximize the accuracy available for a given number of arithmetic operations. Second, new solution algorithms can reduce numerical sensitivity to heterogeneity. Also, well designed algorithms can exploit the emerging generation of fast, parallel-architecture computers.

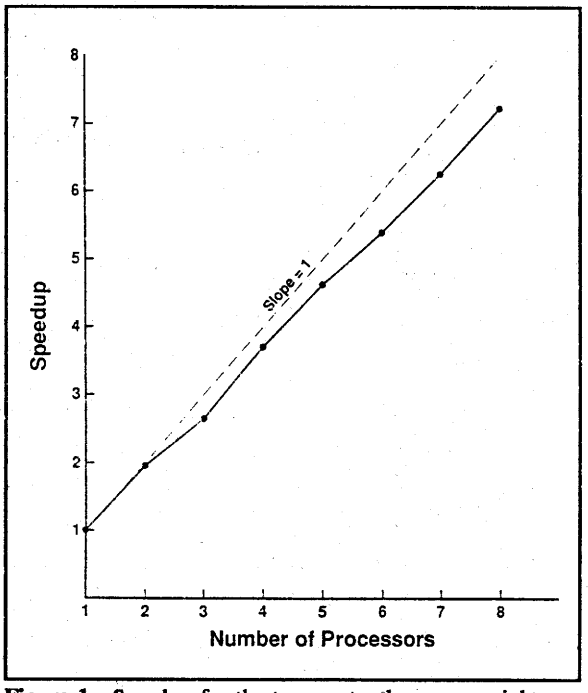

Figure 1. Speedup for the transport solver on an eight-processor computer. Ideal parallelization corresponds to a slope of 1.

Results For the flow equations a technique called **mixed finite elements** produces accurate water velocities, unlike standard methods that numerically differentiate accurate heads to compute inaccurate velocities. A **multigrid** algorithm solves the approximations rapidly

and in parallel, even when heterogeneity forces modelers to use extremely small cells for resolution.

A technique called **alternating-direction collocation** yields highly accurate solutions to the transport equation. The technique works well on parallel machines, and it admits embellishments that make it suitable for high-velocity flows. Among these are timestepping along natural contaminant paths (**method of characteristics**) and adaptively locating small cells to resolve moving contaminant fronts (**local grid refinement**).

Future Work One can extend the techniques to accommodate more complicated physics. Of special interest are tensor rock properties and the nonlinearities associated with nonaqueous contaminants. Other research involves new scaling rules for aquifer properties. These rules translate hydrogeologic data to the scale of cells used in field-wide models.

Publications

Allen, M.B., and M.C. Curran. 1992. "A multigrid-based solver for mixed finite-elementapproximations to groundwater flow", in Numerical Methods in Water Resources, ed. by T. Russell et al.Elsevier, London. pp579-585.

Allen, M.B., R.E. Ewing, and P. Lu. 1992. "Well conditioned iterative schemes for mixed finite-element models of porous-media flow". Journal of Society for Industrial and Applied Mathematics. 13(3):794-814.

Allen, M.B., and A. Khosravani. "Solute transport via alternating-direction collocation with a modified method of characteristics". Advanced Water Resources. (forthcoming).

S. Smith, M. Allen, J. Puckett, and T. Edgar. 1992. "The finite-layer method for groundwater flow models". Water Resources Research. 28(6):1715-1722.

The Wyoming Water Resources Center publishes two series of Research Briefs. The more technical series is designated by Technical in front of the Research Brief publication number.

For further information on this or other research projects or for a list of WWRC publications, telephone or write:

Wyoming Water Resources Center
P.O. Box 3067, University of Wyoming
Laramie WY 82071-3067
(307) 766-2143; FAX (307) 766-3718

WWRC RESEARCH BRIEFS are published by the Wyoming Water Resources Center with funds provided in part by the US Geological Survey, Department of Interior, as authorized by the Water Resources Research Act of 1984. The research on which this report is based was financed in part by the US Geological Survey, Department of Interior and Wyoming Water Resources Center. The views expressed do not necessarily represent those of the Department of Interior or the WWRC. Persons seeking admission, employment, or access to programs at the University of Wyoming shall be considered without regard to race, color, national origin, sex, age, religion, political belief, handicap, or veteran status.

UNIVERSITY OF WYOMING

Technical RB92-03

recycled paper