SPE 51884

A Parallel Multiblock/Multidomain Approach for Reservoir Simulation

Mary F. Wheeler, SPE, Todd Arbogast, SPE, Steven Bryant, Joe Eaton, Qin Lu, Malgorzata Peszynska, The University of Texas at Austin, and Ivan Yotov, University of Pittsburgh

This paper was prepared for presentation at the 1999 SPE Reservoir Simulation Symposium held in Houston, Texas, 14–17 February 1999.

Abstract

Our approach for parallel multiphysics and multiscale simulation uses two levels of domain decomposition: physical and computational. First, the physical domain is decomposed into subdomains or blocks according to the geometry, geology, and physics/chemistry/biology. Each subdomain represents a single physical system, on a reasonable range of scales, such as a black oil region, a compositional region, a region to one side of a fault, or a near-wellbore region. Second, the computations are decomposed on a parallel machine for efficiency. That is, we use a multiblock or macro-hybrid approach, in which we describe a domain as a union of regions or blocks, and employ an appropriate hierarchical model on each block.

This approach allows one to define grids and computations independently on each block. This local grid structure has many advantages. It allows the most efficient and accurate discretization techniques to be employed in each block. The multiblock structure of the algebraic systems allows for the design and use of efficient domain decomposition solvers and preconditioners. Decomposition into independent blocks offers great flexibility in accommodating the shape of the external boundary, the presence of internal features such as faults and wells, and the need to refine a region of the domain in space or time (by treating it as a distinct block); interfacing structured and unstructured grids; and accommodating various models of multiscale and multiphysical phenomena. The resulting grid is not suited to direct application of discretization methods. We use mortar space techniques to impose physically meaningful, mass conservative, flux-matching conditions on the interfaces between blocks.

We present numerical simulations to illustrate several of these decomposition strategies, including the coupling of IMPES and fully implicit models and upscaling by varying the number of degrees of freedom on the block interfaces.

Introduction

Effective production management of oil reservoirs is placing new demands upon the mathematics and science of reservoir simulation. Many fields cannot be adequately described in terms of standard approaches. For example, some fields support flow rates so high that the possibility of non-Darcy flow arises, a situation normally considered only in gas fields. The geology of these fields is complex, exhibiting thin intervals of extremely high productivity, fractures, and layering. Applying traditional dual porosity fracture models to these intervals is not satisfactory, since the usual simplifying assumptions do not hold even approximately. In some cases, the huge areal extent of the fields makes simulation even on very coarse grids a computationally demanding task. In addition to complex geology, different recovery strategies may be required for different regions of the field, such as tertiary recovery in one area and waterflood in another. For almost any field, it may be useful to vary time stepping procedures over the computational domain.

These and other considerations have motivated a search for a new paradigm to describe, model and understand field behavior. Elements of this paradigm include multiscale analysis, multiblock discretizations, advanced solvers[1], and algorithm implementation on parallel platforms. We are implementing these schemes within IPARS[2], a new generation framework for developing and running parallel reservoir simulators. This framework reverses the usual approach to reservoir simulator development. IPARS provides the infrastructure common to most simulators (input/output routines, table lookup routines, well management, etc.) as well as providing all the structure needed for memory management, message passing, visualization, etc. Any physical model (two-phase, black oil, compositional, etc.) can be run under the framework; the developer need only provide routines that compute a timestep for the physical model of interest. Data management and parallel implementation issues are

transparent to the developer, so that new parallel simulators can be built in much shorter time. For example, an IMPES total-velocity formulation of the two-phase flow equations was coded, debugged, tested and running within one month. The IPARS framework is portable, running on platforms such as a single PC (Windows, DOS, Linux), a cluster of PCs under Linux, SGI and IBM workstations, and T3E and SP.

Our approach for parallel multiphysics/multiscale simulation uses the concept of multiple blocks or subdomains, organized as two levels of domain decomposition for parallel computation: physical and computational. First the physical problem is decomposed into regions (i.e., "blocks") in which different physical and/or numerical models are deemed appropriate for the simulation. These blocks are chosen based on the geometry and geology of the reservoir, the placement of wells, the physical, chemical and biological processes assumed to be taking place, and the temporal and spatial scales expected to arise in the solution. In the second stage, the computations are further decomposed on a parallel machine to achieve computational efficiency. In this way we describe the reservoir as a union of blocks. The approach offers great flexibility in accommodating the shape of the external boundary, the presence of internal features such as faults and discrete fractures, the need to refine a region of the domain in space (and thus to treat it as a distinct block), and to accommodate different models in different parts of the domain. That is, we can handle multiscale and multiphysical phenomena in this approach. It is usually preferable to define the grids independently on each block in order to follow large-scale geological features in the subsurface and to allow for the proper spatial resolution. The resulting grid is not suited, however, to direct application of discretization methods.

The use of mortar finite element techniques allows us to formulate numerical models and design software for multiblock domains consistent with the physical and engineering description of the underlying equations. The equations hold with their usual meaning on the subdomains, and use more or less standard numerical models such as IMPES or fully implicit compositional. These are tied together by physically meaningful interface boundary conditions imposed on the blocks. Herein we formulate and demonstrate the applicability of these mortar space decompositions to multiblock and multi-numerical models (by coupling sequential and fully implicit formulations) for two-phase flow in permeable media. Extensions to three phase flows as well as to multimodel/multiphysics are also discussed.

We also address the solution of the nonlinear interface problem arising in the multiblock formulations for multiphase flow. The Jacobian of the interface problem is not readily available for this system, so we have investigated an inexact Newton's method using several GMRES iterations. For solving subdomain problems, we have studied Newton/Krylov strategies based on combining global Newton methods with Krylov subspace iterative methods for solving the nonsymmetric linear systems that arise at each Newton step. These strategies must be parallelizable, as each subdomain problem could be split across a number of processors. Also, we describe some of the major implementation issues.

We provide several computational examples. These include a problem with a high permeability streak, a multinumerics problem involving a random permeability field, an irregular heterogeneous geological domain, and a form of multiblock upscaling.

Multiblock formulation and discretization

For simplicity of presentation we consider a two-phase flow model. In a multiblock formulation the domain $\Omega \subset \mathbf{R}^3$ is decomposed into a series of subdomains Ω_k, $k = 1,...,n_b$. Let $\Gamma_{kl} = \partial\Omega_k \cap \partial\Omega_l$ be the interface between Ω_k and Ω_l. We note that Γ_{kl} does not have to coincide with the entire face of either subdomain block.

The governing mass conservation equations are imposed on each subdomain Ω_k:

$$\frac{\partial(\phi\rho_\alpha S_\alpha)}{\partial t} + \nabla \cdot \mathbf{U}_\alpha = q_\alpha \quad \text{..... (1)}$$

where $\alpha = w$ (wetting), n (non-wetting) denotes the phase, and

$$\mathbf{U}_\alpha = -\frac{k_\alpha(S_\alpha)K}{\mu_\alpha}\rho_\alpha(\nabla P_\alpha - \rho_\alpha g \nabla D) \quad \text{..... (2)}$$

is the Darcy velocity. On each interface Γ_{kl} the following physically meaningful continuity conditions are imposed:

$$P_\alpha|_{\Omega_k} = P_\alpha|_{\Omega_l}, \quad \text{..... (3)}$$

$$[\mathbf{U}_\alpha \cdot \nu]_{kl} \equiv \mathbf{U}_\alpha|_{\Omega_k} \cdot \nu_k + \mathbf{U}_\alpha|_{\Omega_l} \cdot \nu_l = 0 \quad \text{..... (4)}$$

where ν_k denotes the outward unit normal vector on $\partial\Omega_k$. The above equations are coupled via the volume balance equation and the capillary pressure relation

$$S_w + S_n = 1, \quad p_c(S_w) = P_n - P_w, \quad \text{..... (5)}$$

which are imposed on each Ω_k and Γ_{kl}. We assume that no flow $\mathbf{U}_\alpha \cdot \nu = 0$ is imposed on $\partial\Omega$, although more general types of boundary conditions can be treated.

Discretization spaces. It is important to choose properly the subdomain and interface discretization spaces in order to obtain a stable and accurate scheme. A variant of the mixed method, the expanded mixed method, has been developed for accurate and efficient treatment of irregular domains. The implementation and analysis of the method for single-phase flow have been described in several previous works for single block[3,4,5] and multiblock domains[6,7,8]. The problem is transformed via a coordinate mapping into one on a union of regular computational (reference) grids. After the mapping,

the permeability becomes a full tensor (except in some trivial cases). The mixed method can then be approximated by cell-centered finite differences for the pressure, which is an efficient and highly accurate scheme[3].

To simplify the presentation we describe here only the rectangular reference case. A definition of the discrete spaces on logically rectangular and triangular grids has been given[4,9,10]. Let us denote the rectangular partition of Ω_k by T_k. The lowest order Raviart-Thomas spaces[11] RT_0 are defined on T_k by

$$\tilde{\mathbf{V}}_k = \{\mathbf{v}=(v_1,v_2,v_3): \mathbf{v}|_E = (\alpha_1 x_1+\beta_1, \alpha_2 x_2+\beta_2, \alpha_3 x_3+\beta_3): \alpha_l,\beta_l \in \mathbf{R} \text{ for all } E\in T_k, \text{ and each } v_l \text{ is continuous in the } l\text{th coordinate direction}\},$$

$$\mathbf{V}_k = \{\mathbf{v}\in\tilde{\mathbf{V}}_k : \mathbf{v}\cdot\nu_k = 0 \text{ on } \partial\Omega_k \cap \partial\Omega\},$$

$$W_k = \{w: w|_E = \alpha : \alpha \in \mathbf{R} \text{ for all } E\in T_k\}.$$

To impose the interface matching conditions (3)-(4) we introduce a Lagrange multiplier or mortar finite element space M_{kl} defined on a rectangular grid T_{kl} on Γ_{kl}. In this space we approximate the interface pressures and saturations, and impose weakly normal continuity of fluxes.

If the subdomain grids adjacent to Γ_{kl} are aligned to each other, we take T_{kl} to be the trace of the subdomain grids and define the matching mortar space by

$$M_{kl}^m = \{\mu : \mu|_e = \alpha : \alpha\in\mathbf{R}, \text{ for all } e\in T_{kl}\}.$$

If the grids adjacent to Γ_{kl} are non-matching, the interface grid need not match either of them. Later we impose a mild condition on T_{kl} to guarantee solvability of the numerical scheme. We define our non-matching mortar space on an element $e\in T_{kl}$ by

$$M_{kl}^n(e) = \{\alpha\xi_2\xi_2+\beta\xi_1+\gamma\xi_2+\delta : \alpha,\beta,\gamma,\delta\in\mathbf{R}\},$$

where ξ_l are the coordinate variables on e. Then, for each Γ_{kl}, we give two possibilities for the non-matching mortar space, a discontinuous and a continuous version, as

$$M_{kl}^{n,d} = \{\mu : \mu|_e \in M_{kl}^n(e) \text{ for all } e\in T_{kl}\},$$

$$M_{kl}^{n,c} = \{\mu : \mu|_e \in M_{kl}^n(e) \text{ for all } e\in T_{kl},\ \mu \text{ is continuous on } \Gamma_{kl}\}.$$

We denote by M_{kl} a choice of $M_{kl}^{n,d}$ or $M_{kl}^{n,c}$ (or M_{kl}^m on matching interfaces).

Remark. The usual piece-wise constant Lagrange multiplier space for RT_0 is not a good choice in the case of non-matching grids, since it only provides $O(1)$ flux approximations on the interfaces and a suboptimal global convergence. With the above choice for mortar space, optimal convergence (and, in some cases, superconvergence) is recovered for both pressure and velocity[6,12,13].

The expanded mortar mixed finite element method

Following Arbogast et al.[3], let, for $\alpha = w, n$,

$$\tilde{\mathbf{U}}_\alpha = -\nabla P_\alpha ; \quad (6)$$

then

$$\mathbf{U}_\alpha = \frac{k_\alpha(S_\alpha)K}{\mu_\alpha}\rho_\alpha\left(\tilde{\mathbf{U}}_\alpha + \rho_\alpha g\nabla D\right) \quad (7)$$

Before formulating the method, we note that two of the unknowns can be eliminated using relations (5). Therefore the primary variables can be chosen to be one pressure and one saturation which we denote by P and S.

Let $0 = t_0 < t_1 < t_2 < \ldots$, let $\Delta t^n = t_n - t_{n-1}$, and let $f^n = f(t_n)$.

In the backward Euler multiblock expanded mixed finite element approximation of (1)-(5) we seek, for $1 \le k < l \le n_b$ and $n = 1,2,3\ldots$, $\mathbf{U}_{h,\alpha}^n|_{\Omega_k} \in \mathbf{V}_k$, $\tilde{\mathbf{U}}_{h,\alpha}^n|_{\Omega_k} \in \tilde{\mathbf{V}}_k$, $P_h^n|_{\Omega_k} \in W_k$, $S_h^n|_{\Omega_k} \in W_k$, $\bar{P}_h^n|_{\Gamma_{kl}} \in M_{kl}$, and $\bar{S}_h^n|_{\Gamma_{kl}} \in M_{kl}$ such that, for $\alpha = w$ and n,

$$\int_{\Omega_k}\frac{S_{h,\alpha}^n - S_{h,\alpha}^{n-1}}{\Delta t^n} w\,dx + \int_{\Omega_k}\nabla\cdot\mathbf{U}_{h,\alpha}^n w\,dx = \int_{\Omega_k} q_\alpha w\,dx, \quad w\in W_k \quad (8)$$

$$\int_{\Omega_k}\tilde{\mathbf{U}}_{h,\alpha}^n\cdot\mathbf{v}\,dx = \int_{\Omega_k} P_{h,\alpha}^n\nabla\cdot\mathbf{v}\,dx - \int_{\partial\Omega_k\setminus\partial\Omega}\bar{P}_{h,\alpha}^n\mathbf{v}\cdot\nu_k\,d\sigma, \quad \mathbf{v}\in\mathbf{V}_k \quad (9)$$

$$\int_{\Omega_k}\mathbf{U}_{h,\alpha}^n\cdot\tilde{\mathbf{v}}\,dx = \int_{\Omega_k}\frac{k_{h,\alpha}^n K}{\mu_{h,\alpha}}\rho_{h,\alpha}^n\left(\tilde{\mathbf{U}}_{h,\alpha}^n + \rho_{h,\alpha}^n g\nabla D\right)\cdot\tilde{\mathbf{v}}\,dx, \quad \tilde{\mathbf{v}}\in\tilde{\mathbf{V}}_k \quad (10)$$

$$\int_{\Gamma_{kl}}\left[\mathbf{U}_{h,\alpha}^n\cdot\nu\right]_{kl}\mu\,d\sigma = 0, \quad \mu\in M_{kl} \quad (11)$$

Here $k_{h,\alpha}^n$ and $\rho_{h,\alpha}^n \in W_k$ are given functions of the subdomain primary variables P_h^n and S_h^n. The mortar functions $\bar{P}_{h,\alpha}^n$ can be computed using (5), given the mortar primary variables $\bar{P}_h^n$ and $\bar{S}_h^n$.

Remark. Introducing the pressure gradients $\tilde{\mathbf{U}}_\alpha$ in the expanded mixed method allows for proper handling of the degenerate relative permeability $k_\alpha(S_\alpha)$ (for $S_\alpha = 0$) in (9)-(10). It also allows, even for a full permeability tensor K, to accurately approximate the mixed method on each subdomain by cell-centered finite differences for P_h and S_h. This is achieved by approximating the vector integrals in (9) and (10) by a trapezoidal quadrature rule and eliminating $\tilde{\mathbf{U}}_{h,\alpha}$ and $\mathbf{U}_{h,\alpha}$ from the system[3,4,5].

Remark. A necessary condition for solvability of the scheme is that, for any $\mu \in M_{kl}$,

$$Q_k \mu = Q_l \mu = 0 \;\Rightarrow\; \mu = 0, \quad \text{.....(12)}$$

where Q_k is the L^2-projection onto $\mathbf{V}_k \cdot \nu_k$. This is not a very restrictive condition and requires that the mortar grid is not too fine compared to the subdomain grids. One choice that satisfies this condition for both continuous and discontinuous mortars is to take the trace of either subdomain grid and coarsen it by two in each direction[6,12].

Multinumerics. A key advantage of the multiblock approach is the ease with which different numerical schemes can be used simultaneously in different parts of the domain. The structure of the interface problem is essentially invariant with respect to the solution scheme in the subdomains. The only complication arises when different primary variables are used in different subdomains. In such cases an additional layer of translation is required between the subdomain and the mortar space, but this raises no fundamental difficulties. The example presented below joins an IMPES scheme for a total-velocity formulation of the two-phase flow equations and a fully implicit scheme for the traditional two-phase flow equations.

Extension to three-phase flow. The extension of this formulation to a third flowing phase is straightforward. An additional mass balance is formed for the third phase; in the case of black oil, this third (gas) phase includes terms involving oil phase flux and saturation to account for gas dissolved in the oil. The fact that the gas phase may appear or disappear raises familiar numerical complications in the subdomains that must also be considered in the interfaces. In particular, one must allow for the possibility that a free gas phase exists only on one side of the interface. This requires care in choosing primary variables in the mortar space and in computing fluxes into and out of the interface. Our current implementation uses oil pressure, oil concentration and gas concentration as primary variables.

Solution of the interface problem

Algorithm.

To solve the discrete system (8)-(11) at each time step, we reduce it to an interface problem in the mortar space. This approach is based on a domain decomposition algorithm for single-phase flow developed originally for conforming grids[14], and later generalized to non-matching grids coupled with mortars[6].

Reduction to interface problem. Let $M_h = \bigoplus_{1 \le k < l \le n_b} M_{kl}$ denote the mortar space on $\Gamma = \cup_{1 \le k < l \le n_b} \Gamma_{kl}$. We define a non-linear bivariate interface functional $B^n : (M_h \times M_h) \times (M_h \times M_h) \to \mathbf{R}$ that quantifies non-conservation of flux as follows. For $(\bar{P}_h^n, \bar{S}_h^n) \in M_h \times M_h$ and $(\mu_1, \mu_2) \in M_h \times M_h$, let

$$B^n(P_h^n, S_h^n; \mu_1, \mu_2) = \sum_{1 \le k < l \le n_b} \int_{\Gamma_{kl}} \left(\left[\mathbf{U}_{h,w}^n(\bar{P}_h^n, \bar{S}_h^n) \cdot \nu \right]_{kl} \mu_1 + \left[\mathbf{U}_{h,n}^n(\bar{P}_h^n, \bar{S}_h^n) \cdot \nu \right]_{kl} \mu_2 \right) d\sigma \quad \text{....(13)}$$

where $\left(S_h^n(\bar{P}_h^n, \bar{S}_h^n), \mathbf{U}_{h,\alpha}^n(\bar{P}_h^n, \bar{S}_h^n) \right)$ are solutions to the n_b subdomain problems (8)-(10) with fixed boundary data $\bar{P}_{h,\alpha}^n(\bar{P}_h^n, \bar{S}_h^n)$, given by (5).

Define a non-linear interface operator $B^n : M_h \times M_h \to M_h \times M_h$ by

$$\left\langle B^n(\bar{P}_h^n, \bar{S}_h^n), (\mu_1, \mu_2) \right\rangle = B^n\left((\bar{P}_h^n, \bar{S}_h^n); (\mu_1, \mu_2) \right), \quad (\mu_1, \mu_2) \in M_h \times M_h \quad \text{.....(14)}$$

where $\langle \cdot, \cdot \rangle$ is the L^2-inner product on $M_h \times M_h$. It is now easy to see that the solution to (8)-(11) equals $\left(\bar{P}_h^n, \bar{S}_h^n, S_h^n(\bar{P}_h^n, \bar{S}_h^n), \mathbf{U}_{h,\alpha}^n(\bar{P}_h^n, \bar{S}_h^n) \right)$, where $(\bar{P}_h^n, \bar{S}_h^n) \in M_h \times M_h$ solves

$$B^n(\bar{P}_h^n, \bar{S}_h^n) = 0; \quad \text{.....(15)}$$

that is, the jumps in the fluxes vanish.

Iterative solution. We solve the system of nonlinear equations on the interface (15) by an inexact Newton method. Let $\lambda = (\bar{P}, \bar{S})$. One Newton step, $\delta\lambda$, is computed by a forward difference GMRES iteration for solving $B^{n\prime}(\lambda)\delta\lambda = -B^n(\lambda)$. On each GMRES iteration, the action of the Jacobian $B^{n\prime}(\lambda)$ on a vector μ is approximated by the forward difference

$$\|\bar{P}_h^n\| \frac{B_1^n\left(\bar{P}_h^n + \eta\|\bar{P}_h^n\|\mu_1/\|\mu_1\|, \bar{S}_h^n + \eta\|\bar{S}_h^n\|\mu_2/\|\mu_2\|\right) - B_1^n(\bar{P}_h^n, \bar{S}_h^n)}{\eta\sqrt{\|\bar{P}_h^n\|^2 + \|\bar{S}_h^n\|^2}}$$

and

$$\|\bar{S}_h^n\| \frac{B_2^n\left(\bar{P}_h^n + \eta\|\bar{P}_h^n\|\mu_1/\|\mu_1\|, \bar{S}_h^n + \eta\|\bar{S}_h^n\|\mu_2/\|\mu_2\|\right) - B_2^n(\bar{P}_h^n, \bar{S}_h^n)}{\eta\sqrt{\|\bar{P}_h^n\|^2 + \|\bar{S}_h^n\|^2}}$$

We take $\eta = \sqrt{\varepsilon}$, where ε is the nonlinear tolerance for evaluation of B^n. Details of the inexact Newton-GMRES algorithm are available elsewhere[15].

Note that each GMRES iteration only requires one evaluation of the nonlinear operator B^n. The evaluation of B^n involves solving subdomain problems (8)-(10) in parallel and two inexpensive projection steps, one from the mortar grid onto the local subdomain grids and another from the local grids onto the mortar grid. Because the perturbation η is very small,

the subdomain solution with boundary data $\left(\overline{P}_h^{\,n}, \overline{S}_h^{\,n}\right)$ is a very good initial guess for solving subdomain problems with boundary data $\left(\overline{P}, \overline{S}\right) \equiv \left(\overline{P}_h^{\,n} + \eta \left\| \overline{P}_h^{\,n} \right\| \mu_1 / \left\| \mu_1 \right\|, \overline{S}_h^{\,n} + \eta \left\| \overline{S}_h^{\,n} \right\| \mu_2 / \left\| \mu_2 \right\| \right)$. As a result it usually takes only one nonlinear subdomain iteration to evaluate $B^n\left(\overline{P}, \overline{S}\right)$.

Implementation.
The algorithm described above has been implemented under the IPARS framework[2] (see Introduction).

In our approach the physical models in each block operate independently of each other using available parallel processors. The subdomain code is tightly connected to the interface data and algorithm. Of course, a less integrated approach is possible. The use of mortar spaces allows for coupling of independent modules representing either different numerical or physical models; however, the choice of implementation strategy affects the overall efficiency of the code.

Our implementation consists of three modules: the IPARS physical model code, the MACE/DAGH library handling the mortar (interface) data structures and their parallelism, and the IPARS MultiBlock Library providing the projection mappings, the interface algorithm, and the links between the subdomain data and mortar data.

The input file for the multiblock problem contains descriptions of the subdomain problems as well as a description of the interfaces. Included are all geometrical and geophysical data, a choice of a physical/numerical model to describe the flow and transport in a given block, and the interface variables λ used as a boundary condition for the subdomain problems. For example, consider two blocks, one denoted H and simulated by a two-phase implicit code coupled to a second block (G) simulated by a two-phase IMPES code. In this case the primary unknowns are oil pressure and oil concentration on both subdomains. We define then λ to be the interface (mortar) values of the primary unknown, so $\lambda_H = \left(\overline{P}_o^{\,H}, \overline{C}_o^{\,H}\right)$ in block H and $\lambda_G = \left(\overline{P}_o^{\,G}, \overline{C}_o^{\,G}\right)$ are the Dirichlet boundary values in the blocks H and G, respectively. We also define $B(\lambda) = B(\lambda_G) - B(\lambda_H)$ to be the jump in the fluxes across the interface, with $B(\lambda_H)$ = (FluxOIL, FluxWATER) in block H and $B(\lambda_G)$ = (FluxOIL, FluxWATER) in block G.

These definitions are made in each subdomain independently. The physical/numerical model associated with one block may change but this change would not affect the other blocks.

The computation of $B(\lambda)$ for a given current guess λ needed for the quasi-Newton solver involves the following three steps executed by the IPARS MultiBlock Library and the IPARS time step routines.

1. Given λ in the mortar space, find λ_H and λ_G by projecting the mortars to the boundary of the H and G blocks.

2. (IPARS time step routine) Solve the subdomain problem in blocks H and G with the given Dirichlet boundary values λ_H and λ_G to find new values of the primary unknowns and fluxes $B(\lambda_H)$ and $B(\lambda_G)$ across the interface.

3. Given $B(\lambda_G)$ and $B(\lambda_H)$, find their projections into the mortar space and compute the jumps in FluxOIL and FluxWATER $B(\lambda) = B(\lambda_G) - B(\lambda_H)$ across the interface in the mortar space.

The definition for a different physical model involves a change of $B(\lambda)$ and possibly λ. For example, for a black oil model the primary unknowns could be water pressure, oil concentration and gas concentration with fluxes of oil, water, and gas constituting $B(\lambda)$. Several interesting questions arise if different primary unknowns are chosen in different subdomains. If the number of primary unknowns is the same on both sides of the interface, we merely add equations to treat the interface quantities in a consistent manner. For example, suppose oil saturation, rather than oil concentration, is the primary unknown on the G block. Then the mortar concentration $\overline{C}_o$ may come from the projected subdomain values of saturation multiplied by the density coming from interface pressures $\overline{P}_o$. Alternatively, $\overline{C}_o$ may be projected directly from concentration values C_o computed in the subdomain. If the number of primary unknowns on either side of the interface is not equal, then the code has to make sure that the fluxes of the component which is absent on one side are zero for the time of simulation in order to ensure correct physics. Otherwise, a change in model is indicated.

Parallelism. As mentioned above, our IPARS multiblock implementation is tightly coupled and can use any number of processors. The partitioning of subdomain data and of interface data is done independently, adding the flexibility necessary for load balancing. Memory management routines handle each block separately. In particular, each block can be distributed among a number of processors and so the subdomain solvers are themselves parallel. IPARS handles the communication between ghost cells for processors that share a physical variable. Global sums, maxima, etc. are realized using multiple MPI Communicators. Variables are used in computations only in the relevant subdomains, but variables common to several subdomains may need to be interpreted together for the needs of output and visualization.

For example, Figure 11 (see Multinumerics computational example below) depicts three blocks distributed among three processors. In this example different numerical codes handling two-phase flow are assigned to different subdomains. The

processors need to exchange only the values relevant to the code that is active on the particular block.

Performance. For a given implementation, the performance of the multiblock code depends on the convergence of the interface algorithm. Several auxiliary parameters affect this convergence. These are the interface time step, constants in the approximation of the interface Jacobian by a forward difference quotient, the parameters controlling the Newton-Krylov algorithm, choice of a preconditioner and many others, including all elements which affect the performance of the subdomain solvers. Of course, the most important is the selection of an appropriate physical model and effective numerical algorithm. All these can be varied depending on the conditions in the reservoir.

Computational results

Two-phase flow through a high-permeability streak. In this example we illustrate the flexibility of the multiblock approach by modeling displacement of oil by water through a vertical cross-section. The high-permeability streak, akin to a fracture, is approximated by a union of three layers, two horizontal and one vertical. Permeability in the streak is two orders of magnitude higher than in the rest of the reservoir. The layers are represented by three separate blocks which are discretized on a much finer grid compared to the rest of the domain (see Figure 2). Overall there are five blocks and six mortars.

The initial water saturation is $S_w = 0.2$. Water is injected from the left and a production well is placed along the right edge. The problem is relatively difficult to simulate, since water flows preferentially through the streak and against gravity in the vertical segment. Oil pressure contours and the velocity vector field after 158 days of injection are shown in Figure 1. The oil concentration profile at that time is given in Figure 2. The fine grid along the streak allows for a better resolution of the high velocities in this region. Continuity of normal fluxes across the interfaces is properly imposed using piecewise linear mortars, as can be seen from the zoom on the velocity field in Figure 3.

This example suggests that a fracture or fault can be specified as a subdomain, incorporating the physics of flow in a fracture, and coupled to the rest of the reservoir or aquifer through boundary conditions on both faces. We are investigating the extension of these ideas to networks of fractures.

Multiblock 3D reservoir example. Let us now extend these ideas to a two-phase, 3D reservoir. Consider the horseshoe-shaped reservoir shown in Figure 4. The areal dimensions of the reservoir are 1760 ft by 920 ft; the formation is 13 ft thick. An injection well is located in each wing of the horseshoe, and three production wells are distributed in the bend between the injectors. The grid is refined in order to improve resolution around the wells, with a total of about 4200 gridblocks. The bend of the horseshoe has been faulted via uplift, and we define one block on either side of the fault. The grid in each block follows the original permeability layers, including two thin low-permeability streaks that extend throughout each block. The multiblock approach described above readily accommodates the non-matching grids, so the geology of such a reservoir is captured straightforwardly.

All wells were run at constant pressure, and the pressures after 960 days of production are shown in Figure 5. The corresponding distribution of water saturation is shown in Figure 6. Oil along the sides of the horseshoe, in the injection block, is being bypassed, and the advance of water into the bottom of the uplifted production block is evident. The water-oil ratios at the three production wells plotted in Figure 7 show the preferential advance of water along the right-hand lobe of the structure, toward the #5 Producer.

Multinumerics (Fully implicit coupled to IMPES). Here the results of a multi-model simulation with different numerical models coupled across the interface are presented.

Specifically, we consider a U-shaped reservoir 24 ft thick with areal dimensions 1200 ft x 400 ft consisting of three blocks. An injector is in the corner of one block, a producer in the opposite corner of the third block, and the middle block is shifted laterally to form the U-shape. Nonmatching grids across the two interfaces are coupled by the mortar spaces. The permeability field is random and anisotropic with a 10:1 ratio of horizontal to vertical permeability and a set of layers of higher mean permeability cutting through the reservoir as shown in Figure 8. The average vertical permeabilities in the layers in the first block are 8.2, 6.9, 123.0, and 8.9 md, from the top down. The bottom of the injection well is in the highly permeable layer. The results of the simulation after 1500 days are shown in Figures 9 and 10. Note the configuration of the contours in the vicinity of the corners and the relation between the concentrations and permeabilities. The preferential flow along the high permeability layer is evident, and the lower pressure gradient in the block corners results in higher oil saturations there.

The variation in permeability makes the problem more difficult for the interface algorithm: for a given time step the interface algorithm may need twice as many iterations to converge than in the case of a homogeneous but layered permeability field when the same convergence criteria and algorithm parameters are used. However, each interface iteration can be made less costly by adapting the numerical method on a given subdomain or block to the difficulties anticipated. In our example, the flow around the wells in the outside blocks is described in a fully implicit way with oil pressure and oil concentration chosen as primary unknowns and the Newton iterations solved by a GMRES solver. With no wells present in the middle block, the computations there are handled by the IMPES-like code based on the sequential formulation of the two-phase flow equations with an

inexpensive PCG solver of the pressure and saturation equations. This is motivated by the relatively small variation in the oil pressure field in the middle block. Note that the time step chosen for the middle block IMPES code is not subject to the usual constraints since the code does not have to handle any wells there. In general, the time step size on the subdomains need not be identical across the interface and may be controlled only from above by the time step on the interface. The assignment of the numerical methods to the nonrectangular subdomains is pictured in Figure 11.

Mortar adaptivity

Adapting mortar degrees of freedom may result in substantial reduction of the cost for solving the interface problem. The solvability condition (12) does not preclude using mortar grids much coarser than the subdomain grids. One must expect, however, a certain loss of accuracy with the coarsening of the interface grids. In the following example we study how the reduction of mortar degrees of freedom affects the number of interface iterations and the flux discretization error on the interface. Similar ideas have been explored by Dorr[16]. We solve a single phase flow problem on a 32×32×32 domain with a highly correlated log-normal permeability field and one injection and three production wells at the corners. A 2×2×2 domain decomposition is employed. The results of the experiment are shown in Figure 12. The traces of subdomain grids on each interface are 16×16; 256 mortar degrees of freedom is equivalent to exact matching of the fluxes. We report the number of conjugate gradient iterations (no preconditioning) and relative flux L^2-error on the interface for several levels of coarsening of the mortar grids and for three different types of mortars. The error for the piecewise constant mortars grows very rapidly, indicating that this is not a good choice. This is consistent with our theoretical results[12]. The two bilinear cases behave similarly, although the continuous case performs somewhat better. In this case, the number of mortar degrees of freedom, and consequently the number of interface iterations, can be reduced by a factor of two, with the relative flux error still being under ten percent. Moreover, the global relative error is even smaller, as the solution is superconvergent away from the interfaces.

Upscaling via Multiblock

The reduction of mortar degrees of freedom can be viewed as an upscaling procedure. Standard upscaling techniques compute effective permeabilities on coarse grids. It is usually difficult to estimate the error associated with the upscaling process. Here we compute, in a sense, an effective flow field, and the flux jump is a good indication of the numerical error.

If only a single bilinear mortar is used on each interface, we have a two scale problem, where the solution is computed locally on the fine scale and fluxes match on the coarse (subdomain) scale. One can view the solution as a sum of a coarse grid solution and a local fine grid correction[17,18]. For example, we have solved the single phase flow equation with a log-normal permeability field using very coarse mortar interfaces. Using a single linear mortar to match the fluxes on each interface of a coarse 4×4 grid, the solution is very similar to the fine-scale (32×32 grid), single-block solution. A similar procedure using constant rather than linear mortars produced highly inaccurate results.

Conclusions

A domain decomposition procedure for treating multiphysics and multiscale simulation with multinumerical models has been formulated, and computational examples presented. In this approach it is important to first decompose the physical problem into subdomains (i.e., blocks) over which a single, appropriate hierachical model describes the relevant physics, chemistry, geology, etc., on relevant time and space scales, and then to further decompose the computations for efficient solution on a parallel computer. The use of mortar space techniques provides a flexible, accurate, and efficient way to synthesize the subdomain problems back into the desired global problem. Mortar spaces also provide a paradigm for linear and nonlinear system solution techniques (domain decomposition), and allow for incorporation of non-matching grids and certain upscaling techniques.

Acknowledgments

John A. Wheeler, Manish Parashar, Carter Edwards, and Srinivas Chippada contributed to IPARS code development. DOE, NGOTP, NSF (KDI grant DMS 9873326), and the Industrial Affiliates of the Center for Subsurface Modeling supported this work.

Nomenclature

B = *interface operator*
B = *bivariate interface operator*
D = *depth*
E = *3D grid cell or element*
e = *2D mortar grid cell or element (face)*
g = *gravitational constant*
h = *(as subscript) approximated quantity*
K = *absolute permeability tensor*
k_α = *relative permeability of phase* α
M_{kl} = *discrete space of mortar scalar functions*
M_h = *discrete space of mortar scalar functions on all interfaces*
n_b = *number of blocks or subdomains in the multi-block grid*
P_α = *pressure of phase* α
$\bar{P}$ = *interface (mortar) pressure*
p_c = *capillary pressure*
q = *well flow rates (sources/sinks)*
Q_k = *L^2-projection onto* $\mathbf{V}_k \cdot \nu_k$
$\mathbf{R}$ = *real number*
S_α = *saturation of phase* α
$\bar{S}$ = *interface (mortar) saturation*
$\mathbf{T}_k$ = *collection of 3D grid cells on* Ω_k
$\mathbf{T}_{kl}$ = *collection of 2D grid cells on* Γ_{kl}
t = *time*

t_l = lth *time level*
$\mathbf{U}_\alpha$ = *phase Darcy velocity*
$\tilde{\mathbf{U}}$ = *pressure gradient*
$\mathbf{V}_k$ = *discrete space of vector functions with outer boundary condition imposed*
$\tilde{\mathbf{V}}_k$ = *discrete space of vector functions*
$\mathbf{v}$ = *discrete space vector test function*
$\tilde{\mathbf{v}}$ = *discrete space vector test function*
W_k = *discrete space of scalar functions*
w = *discrete space scalar test function*
x = *3D position vector*

α = *phase subscript (o, w, or g)*
Γ_{kl} = *the interface between the kth and lth blocks*
Δt^n = nth *time step*
δ = *Newton change*
η = *forward difference parameter*
ξ = *2D position vector in interface*
λ = *vector of mortar interface primary unknowns*
μ = *discrete space mortar scalar test function*
μ_α = *phase viscosity*
ν = *outer unit normal vector*
ρ = *density*
ϕ = *porosity*
Ω = *the reservoir domain*
Ω_k = *the k*th *block or subdomain*
$\partial\Omega$ = *boundary of* Ω

$d\sigma$ = *differential of surface area*
dx = *differential of volume*
$[\bullet]_{kl}$ = *jump (difference in values on two sides of an interface)*

FIGURES

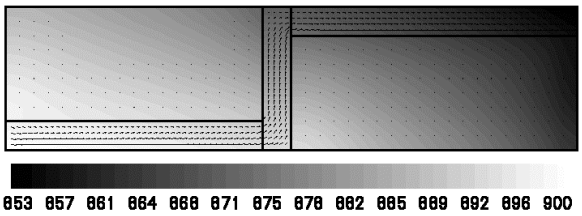

Figure 1. Displacement of oil by water (injected at left face) in a medium containing a high permeability channel. The mesh in the domains representing the streak is finer than in the domains representing the rest of the medium. Shading and arrows indicate oil phase pressure and velocity after 158 days.

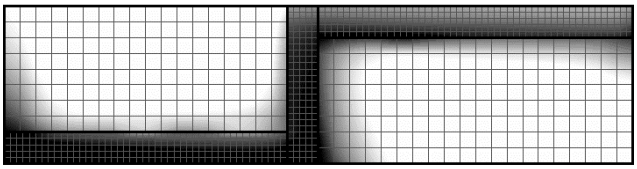

Figure 2. Shading shows the oil saturation (light=high S_o, dark=low S_o) after 158 days in the same simulation as Fig. 1. Preferential displacement within the streak is evident.

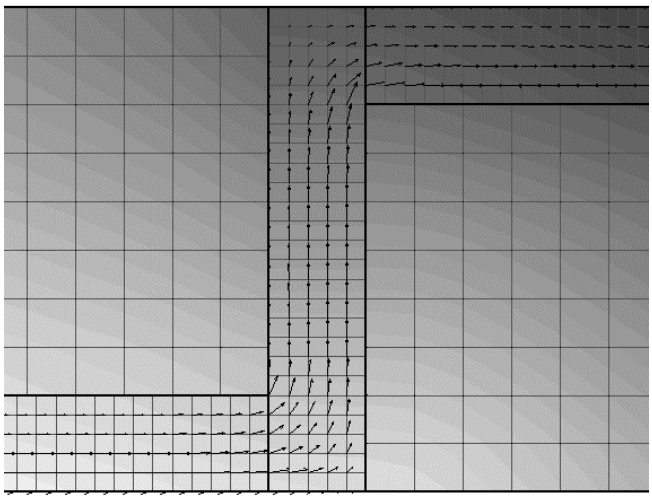

Figure 3. Detail from Fig. 1 illustrates continuity of fluxes between blocks maintained by the mortar.

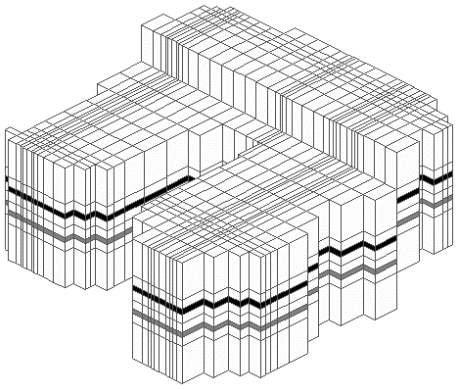

Figure 4. Permeability layers, fault-block structure and irregular mesh for horseshoe reservoir example. The darker layers have low permeability; the anisotropy is k_v/k_h = .15.

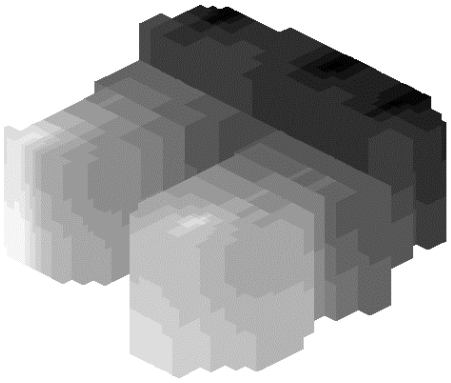

Figure 5. Pressure distribution after 960 days production in the horseshoe reservoir example.

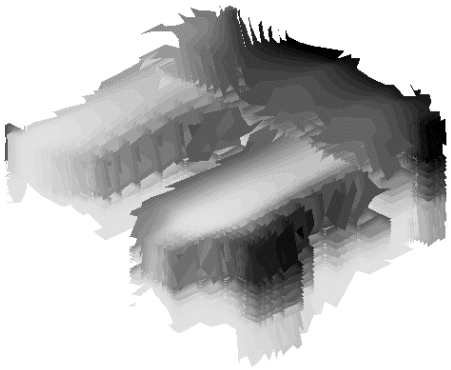

Figure 6. Water saturation contours after 960 days production in the horseshoe reservoir example.

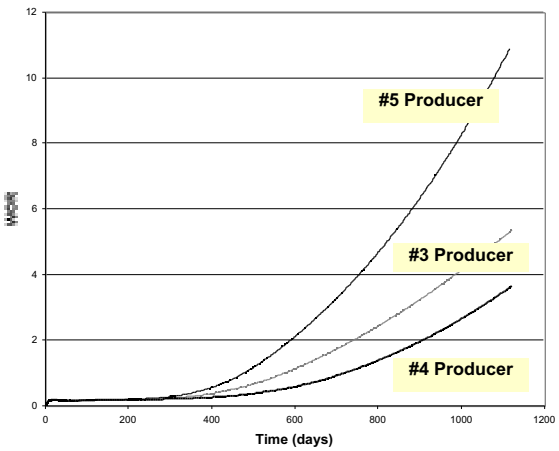

Figure 7. Water-oil ratios for the horseshoe reservoir example show the early breakthrough in the #5 Producer, which is closest to the injection wells, and the later arrival at the #4 Producer located in the middle of the production block.

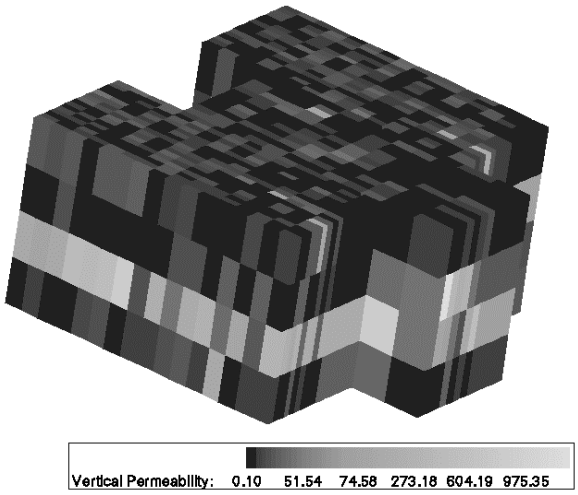

Figure 8. Random permeability field in a three-block reservoir example in which different numerical schemes are used in the blocks.

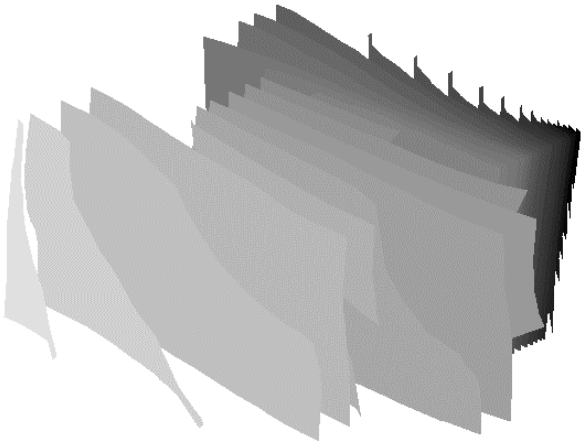

Figure 9. Pressure contours in the multimodel example after 1500 days injection.

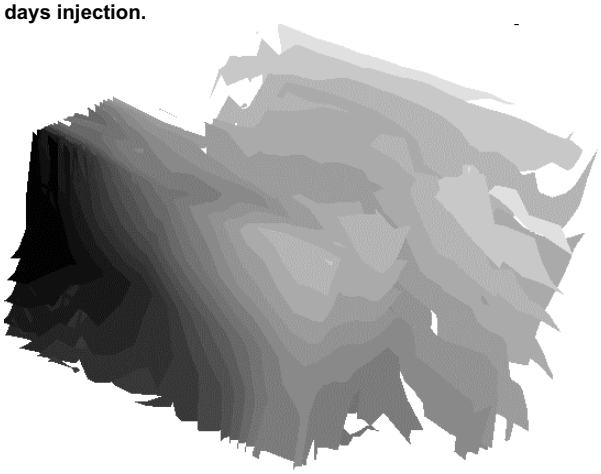

Figure 10. Oil concentrations (light=high S_o, dark=low) in the multimodel example after 1500 days injection. Bypassed oil is apparent in the top right corner of the injection block and the middle block.

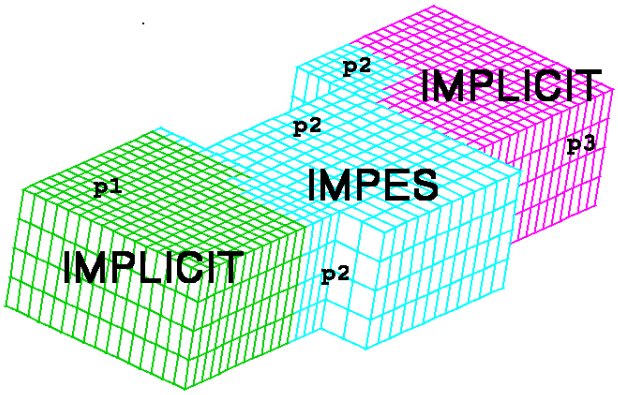

Figure 11. Numerical schemes and grids used in the multimodel example.

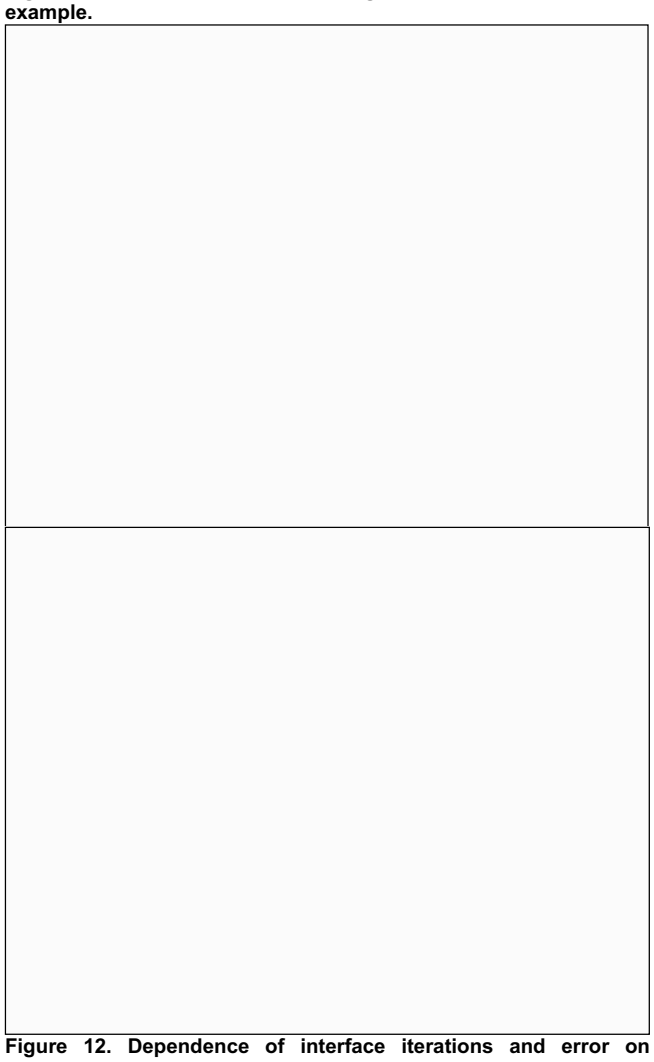

Figure 12. Dependence of interface iterations and error on number of interface degrees of freedom; mortar 1--continuous piecewise bilinears, mortar 2--discontinuous piecewise bilinears, mortar 3--piecewise constants.

References

[1] Y.-A. Kuznetzov and M. F. Wheeler: "Optimal order substructuring preconditioners for mixed finite element methods on nonmatching grids," *East-West J. Numer. Math.*, pp. 127-143, 1995.

[2] P. Wang, I. Yotov, M. Wheeler, T. Arbogast, C. Dawson, M. Parashar, K. Sepehrnoori: "A new generation EOS compositional reservoir simulator: Part I--Formulation and discretization," SPE 37979, 14th SPE Symposium on Reservoir Simulation, Dallas, June 8-11, 1997.

[3] T. Arbogast, M. F. Wheeler and I. Yotov: "Mixed finite elements for elliptic problems with tensor coefficients as cell-centered finite differences," *SIAM J. Numer. Anal.,* **34,** 2, 828-852, 1997.

[4] T. Arbogast, C. N. Dawson, P. T. Keenan, M. F. Wheeler and I. Yotov: "Enhanced cell-centered finite differences for elliptic equations on general geometry," *SIAM J. Sci. Comp.*, **19**, 404-425, 1998.

[5] T. Arbogast, P. T. Keenan, M. F. Wheeler and I. Yotov: "Logically rectangular mixed methods for Darcy flow on general geometry," SPE 29099, 13th SPE Symposium on Reservoir Simulation, San Antonio, Texas, 51-59, 1995.

[6] I. Yotov: *Mixed finite element methods for flow in porous media*, PhD Thesis, Rice University, Houston, Texas, 1996.

[7] T. Arbogast, M. F. Wheeler and I. Yotov: "Logically rectangular mixed methods for flow in irregular heterogeneous domains," *Computational Methods in Water Resources XI,* A. A. Aldama et al., eds., Computational Mech. Publ., Southampton, 621-628, 1996.

[8] I. Yotov: "Mortar mixed finite element methods on irregular multiblock domains," *Iterative Methods in Scientific Computation, IMACS series Comp. Appl. Math.,* J. Wang et al., eds., **4,** IMACS, 239-244, 1998.

[9] J. M. Thomas: *These de Doctorat d'etat,* l'Universite Pierre et Marie Curie, 1977.

[10] F. Brezzi and M. Fortin: *Mixed and hybrid finite element methods,* Springer-Verlag, New York, 1991.

[11] R. A. Raviart and J. M. Thomas: "A mixed finite element method for 2nd order elliptic problems," in *Mathematical Aspects of the Finite Element Method, Lecture Notes in Mathematics* **606,** Springer-Verlag, New York, 292-315, 1977.

[12] T. Arbogast, L. C. Cowsar, M. F. Wheeler and I. Yotov: "Mixed finite element methods on non-matching multiblock grids," to appear, *SIAM J. Num. Anal.*

[13] I. Yotov: "A mixed finite element discretization on non-matching multiblock grids for a degenerate parabolic equation arising in porous media flow," *East-West J. Numer. Math.* **5**, 211-230, 1997.

[14] R. Glowinski and M. F. Wheeler: "Domain decomposition and mixed finite element methods for elliptic problems," *First International Symposium on Domain Decomposition Methods for Partial Differential Equations*, R. Glowinski et al., eds., SIAM, Philadelphia, 144-172, 1988.

[15] C. T. Kelley: *Iterative methods for linear and nonlinear equations,* SIAM, Philadelphia, 1995.

[16] M. R. Dorr: "On the discretization of interdomain coupling in elliptic boundary value problems," *Second International Symposium on Domain Decomposition Methods*, T.F. Chan et al., eds., SIAM, Philadelphia, 17-37, 1989.

[17] T. Arbogast, S. E. Minkoff, and P. T. Keenan: "An operator-based approach to upscaling the pressure equation," in *Computational Methods in Water Resources XII*, v. 1, V. N. Burganos et al.,

eds., Computational Mech. Publ., Southampton, U.K., 405-412, 1998.

[18] N. Moes, J. T. Oden and K. Vemaganti: "A two-scale strategy and *a posteriori* error estimation for modeling heterogeneous structures," *On new advanced in adaptive computational methods in mechanics*, Elsevier, 1998.