SPE-172990-MS

A Multiscale Mortar Method And Two-Stage Preconditioner For Multiphase Flow Using A Global Jacobian Approach

Benjamin Ganis, Kundan Kumar, and Gergina Pencheva, SPE; Mary F. Wheeler, The University of Texas at Austin; Ivan Yotov, University of Pittsburgh

This paper was prepared for presentation at the SPE Large Scale Computing and Big Data Challenges in Reservoir Simulation Conference and Exhibition held in Istanbul, Turkey, 15–17 September 2014.

Abstract

We consider a fully-implicit formulation for two-phase flow in a porous medium with capillarity, gravity, and compressibility in three dimensions. The method is implicit in time and uses the multiscale mortar mixed finite element method for spatial discretization in a non-overlapping domain decomposition context. The interface conditions between subdomains are enforced in terms of Lagrange multiplier variables defined on a mortar space. There are two novelties in our approach: first, we linearize the coupled system of subdomain and mortar variables simultaneously to form a global Jacobian and eliminate variables by taking Schur complements; and second, we adapt a two-stage preconditioning strategy to solve the resulting formulation. The new formulation exploits the local invertibility of mortar block matrices to obtain a system with subdomain variables as unknowns. The two-stage preconditioner uses the Householder transformation to decouple the pressure and saturation variables for each grid block. This allows the elliptic equation for the pressure and hyperbolic equation for the saturation to be solved in a decoupled manner. The algorithm is fully parallel and numerical tests show the low computational costs of eliminating the mortar variables to obtain the final Jacobian matrices. We have demonstrated parallel scalability using our two-stage preconditioner on large-scale heterogeneous two-phase reservoir simulation problems with over 10 million elements and over 1000 processors.

Introduction

This work deals with a novel approach for solving the two-phase fully implicitly in a domain decomposition framework. Driven by the energy security and environmental implications, there has been an intensive effort in developing algorithms for accurate simulations for multiphase flow in porous media. Domain decomposition based methods provide a computationally efficient way to deal with the presence of multiple scales and heterogeneities present in the geological systems (Toselli and Widlund, 2005). Our approach develops a new approach to solve the two-phase flow problem using Multiscale Mortar Mixed Finite Element Method (MMMFEM) (Arbogast, et al. 2000, Arbogast, et al. 2007). This is a non-overlapping domain decomposition based method and adopts a "divide and conquer" strategy in which a reservoir domain is divided into subdomains where independent flow equations are solved with the

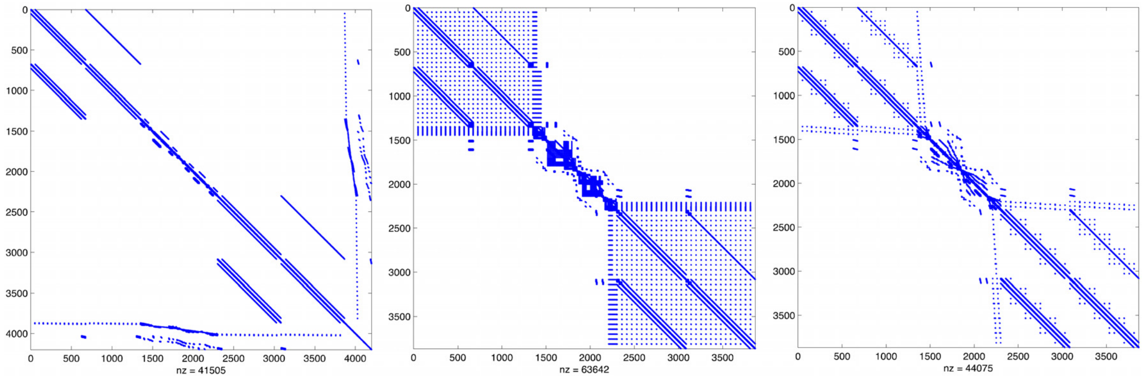


Figure 1—Sparsity pattern of Jacobian matrices for Example 2. From left to right: J_1 matrix for unknowns (δP_o, δN_o, $\delta\Lambda_1$, $\delta\Lambda_2$), J_3 matrix for unknowns (δP_o, δN_o) without mass lumping, and $\tilde{J}3$ matrix for unknowns (δP_o, δN_o) with mass lumping.

coupling achieved through the interface (or mortar) variables. The choice of mixed finite element method allows an accurate computation of flux and ensures local mass conservation (Arbogast et al. 1997, Brezzi and Fortin 1991, Dawson, et al. 1997). A multiscale mortar method provides an independent but fine discretization of subdomains coupled to a coarse mesh on interfaces. The flexibility of discretization allows an adaptive refinement around wells and channels. Moreover, the interfaces may conform to the geological features such as faults and fractures.

Recently, we have developed a global Jacobian approach for solving nonlinear flow equations for both single and two-phase flow models for MMMFEM (Ganis, et al. 2014a, Ganis, et al. 2014b). The main idea in this approach is to linearize the global system in both subdomain and interface variables simultaneously to yield a single Newton iteration. At this stage, we eliminate the flux variables and interface unknowns by taking appropriate Schur complements. As it has been shown in our previous work, this algorithm provides an efficient approach for solving the nonlinear single and two-phase flow equations.

This approach is in contrast with the earlier approach for solving nonlinear flow problems in a non-overlapping domain decomposition framework, where a coarse scale nonlinear interface problem is solved. The interface Jacobian is numerically approximated using finite difference (FD) and at each Newton interface iteration the nonlinear subdomain problem is solved using Newton method (Yotov 2001, Peszynska, et al. 2002, Ganis, et al. 2012). This leads to a nested Newton approach with four nested linear iterations and requires selection of five progressively tighter tolerances for convergence.

This work develops an efficient parallel two-stage based preconditioner for fully-implicit multiphase flow using a multiscale mortar discretization with the linear system obtained as a result of global linearization. The existing two-stage preconditioners are applicable for single domain case (Lacroix, et al. 2003, Wallis et al. 1985, Vassilevski 1984, Cao et al. 2005). Therefore, they need to be adapted to our case when mortar variables are also unknowns. In principle, it is possible to treat the entire Jacobian matrix and introduce such a preconditioner to work on the entire matrix. However, the most straightforward way to apply this preconditioner is to take a Schur complement and eliminate the mortar variables and obtain a global system of pressure and saturation. In Figure 1, we show the sparsity pattern and non-zero elements of the Jacobian matrix. The elimination of mortar variables requires an inversion of $J_{\Lambda\Lambda}$ blocks of matrices. As Figure 1 shows these blocks are small in size and are decoupled and hence, a direct inversion may seem an attractive option. However, this causes a non-negligible increase in non-zero elements of the Jacobian resulting matrix. To alleviate this, we use a mass lumping of $J_{\Lambda\Lambda}$, which provides a diagonal matrix making the inversion trivial and improving sparsity of the Schur complement. Furthermore, following Lacroix, et al. 2003, we consider a combinative two-stage preconditioner. A Householder projection is used to decouple the pressure and saturation variables in each grid block followed by a decoupling of pressure block. The second stage gets the feedback from first stage and updates both pressure and saturation by solving the full system with a very loose tolerance.

The contributions of this work are in developing a two-stage based preconditioner and we show its effectiveness in solving large-scale problems. In terms of formulation, the Schur complement obtained as a result of eliminating the Lagrange multipliers is a novelty. We perform several numerical experiments to understand the parameters used in the two stages and we will comment on the insights gained from our experience. Furthermore, the parallel scalability and efficiency as shown in solving a mega size problem (20 million unknowns) demonstrates that this is a matured tool for solving multiphase problems in a domain decomposition framework.

Global Jacobian Algorithms for the Multiscale Mortar Method

We consider flow of two immiscible phases through a porous medium, where phases $\alpha = o$ (oil) and $\alpha = w$ (water) are both assumed to be slightly compressible fluids. The time interval is $[0,T]$, along with a spatial domain, $\Omega \subset \mathbb{R}^d$, $d = 2$ or 3 with boundary $\partial\Omega$, and outward unit normal $\mathbf{n}$. The spatial domain is decomposed into N_Ω non-overlapping subdomains such that $\overline{\Omega} = \cup_{k=1}^{N_\Omega} \overline{\Omega^k}$ and $\Omega^k \cap \Omega^l = \phi$ when $k \neq l$. Subdomain interfaces are denoted by $\Gamma^{kl} = \Gamma^{lk} = \partial\Omega^k \cap \partial\Omega^l$, $\Gamma = \cup \Gamma^{kl}$ and $\boldsymbol{n}^k$ is the outward unit normal to Ω^k.

We consider an expanded mixed formulation for a fully-implicit two-phase system. On subdomain interfaces, Lagrange multipler variables are introduced corresponding to the two phase pressures, and zero flux jump conditions are enforced. The concentration and mobility for phase α is denoted by $n_\alpha = \rho_\alpha S_\alpha$ and $m_\alpha = \rho_\alpha k_{r\alpha}/\mu_\alpha$, respectively. The primary unknowns for the multiscale mortar discretization are subdomain unknowns (p_o, n_o) auxiliary velocities $(\tilde{\boldsymbol{u}}_o, \tilde{\boldsymbol{u}}_w)$, velocities $(\boldsymbol{u}_o, \boldsymbol{u}_w)$, and Lagrange multipliers (λ_o, λ_w).

For each subdomain Ω^k, let $\mathcal{T}_h^k$ be a conforming quasi-uniform finite element partition consisting of rectangular elements or bricks with characteristic mesh size h , which may be different for each k thereby allowing non-matching grids on the subdomain interfaces. Our numerical results use the lowest order Raviart-Thomas-Nedelec spaces $\boldsymbol{V}_h^k \times W_h^k$ on bricks for the flux and the pressure (Raviart and Thomas, 1977, Nédélec, 1980). For each interface Ω^{kl}, we choose an independent quasi-uniform finite element partition $\mathcal{G}_H^{kl}$ with characteristic mesh size H and define the mortar space M_H^{kl} containing either continuous or discontinuous piecewise polynomials of degree larger than one.

For each phase α the global velocity, pressure, and mortar spaces are $\boldsymbol{V}_h = \oplus_{k=1}^{N_\Omega} \boldsymbol{V}_h^k$, $W_h = \oplus_{k=1}^{N_\Omega} W_h^k$, and $M_H = \oplus_{1 \le l < k \le N_\Omega} M_H^{kl}$, respectively. Choose a temporal discretization $0 = t^0 < t^1 < \ldots < t^{N_T} = T$, with $\delta t^n = t^n - t^{n-1}$. To simplify notation we suppress subscripts h and H, as well as time index n except for n_α^{n-1} which denote the known concentrations at the previous time step. Employing the backward Euler method for time integration with the aforementioned discrete spaces for pressure, mass flux, and mortar variables to the two-phase flow system gives the following fully-discrete multiscale mortar expanded mixed finite element system for fully-implicit two-phase flow.

The MMMFEM is stated as follows. For time levels $n = 1, \ldots ,N_T$ find $(\tilde{\boldsymbol{u}}_o^k, \tilde{\boldsymbol{u}}_w^k, \boldsymbol{u}_o^k, \boldsymbol{u}_w^k, p_o^k, n_o^k, \lambda_1^{kl}, \lambda_2^{kl}) \in 4\boldsymbol{V}_h^k \times 2W_h^k \times 2M_H^{kl}$ such that for subdomains Ω^k, $k = 1, \ldots, N_\Omega$,

$$\int_{\Omega^k} \frac{\phi n_\alpha^k - \phi n_\alpha^{n-1}}{\delta t} w \, dx + \int_{\Omega^k} \nabla \cdot \boldsymbol{u}_\alpha^k \, w \, dx - \int_{\Omega^k} q_\alpha w \, dx = 0, \tag{1a}$$

$$\int_{\Omega^k} K^{-1} \, \tilde{\boldsymbol{u}}_\alpha^k \cdot \tilde{\boldsymbol{v}} \, dx - \int_{\Omega^k} p_\alpha^k \nabla \cdot \tilde{\boldsymbol{v}} \, dx - \int_{\Omega^k} \rho_\alpha \boldsymbol{g} \cdot \tilde{\boldsymbol{v}} \, dx + \sum_{l=1, l \neq k}^{N_\Omega} \int_{\Gamma^{kl}} p_\alpha^\Gamma \, \tilde{\boldsymbol{v}} \cdot \boldsymbol{n}^k \, d\sigma = 0, \tag{1b}$$

$$\int_{\Omega^k} \boldsymbol{u}_\alpha^k \cdot \boldsymbol{v} \, dx - \int_{\Omega^k} m_\alpha \, \tilde{\boldsymbol{u}}_\alpha^k \cdot \boldsymbol{v} \, dx = 0, \tag{1c}$$

and for interfaces Γ^{kl}, $1 \le k < l \le N_\Omega$,

$$\int_{\Gamma^{kl}} (\boldsymbol{u}_\alpha^k \cdot \boldsymbol{n}^k + \boldsymbol{u}_\alpha^l \cdot \boldsymbol{n}^l)\eta \, d\sigma = 0, \tag{1d}$$

for all $w \in W_h^k, \tilde{\boldsymbol{v}} \in \boldsymbol{V}_h^k, \boldsymbol{v} \in \boldsymbol{V}_h^k$ and $\eta \in M_H^{kl}$ This system has been given previously in such works as Yotov, 2001 and Peszynska, et al. 2002.

We now express the 8 unknowns as linear combinations of finite element and mortar basis functions, e.g. $p_o^k = \sum_{i=1}^{N_p^k} P_{o,i}^k w_i^k$. This reformulates the problem in terms of unknown coefficient vectors $\tilde{U}_o^k, \tilde{U}_w^k, U_o^k, U_w^k \in \mathbb{R}^{N_u^k}$, $P_o^k, N_o^k \in \mathbb{R}^{N_p^k}$, and $\Lambda_1^{kl}, \Lambda_2^{kl} \in \mathbb{R}^{N_\lambda^{kl}}$. Let $N_u = \sum_{i=1}^{N\Omega} N_u^k$, $N_p = \sum_{i=1}^{N\Omega} N_p^k$, and $N_\lambda = \sum_{1 \le k < l \le N_\Omega} N_\lambda^{kl}$ count the total number of degrees of freedom in the global velocity, pressure, and mortar spaces. The discrete system (1a)–(1d) becomes a set of $(4N_u + 2N_p + 2N_\lambda)$nonlinear equations. Using the global Jacobian approach, it can be solved by a single Newton iteration. This is in contrast to the earlier forward difference approach, which required a more expensive algorithm with two nested Newton iterations.

To form the global Jacobian system, we compute the derivatives of each residual equation with respect to each type of unknown (Ganis, et al. 2014a, 2014b). Entries with two pressure basis functions are evaluated using the midpoint rule, entries with one or more velocity basis functions are evaluated using the trapezoidal midpoint rule (Arbogast, et al. 2007), and entries with mortar basis functions are evaluated using Newton-Cotes rules with accuracy depending on the polynomial degree of the mortar space. We drop certain terms from the Jacobian that are multiplied by the small compressibility constants c_o and c_w as typically done in reservoir simulations with slight compressibility. Terms with phase mobility are upwinded using the accurate “block to block” upwinding approach (Ganis, et al. 2014b). The final step is to form a Schur complement system which eliminates the four velocity variables. Under the aforementioned trapezoidal midpoint rule and our assumption of Cartesian subdomain grids, this forms a cell-centered finite difference method with a two-point flux scheme in the subdomain interior and a larger stencil adjacent to subdomain interfaces.

We define the vector of subdomain variables as $\delta\Theta = [\delta P_o, \delta N_o]$ and the vector of mortar variables as $\delta\Lambda = [\delta\Lambda_1, \delta\Lambda_2]$ Two algorithms have already been described in Ganis, et al. 2014a and Ganis, et al. 2014b: the global Jacobian (GJ) method, and the global Jacobian Schur (GJS) method. In the former method, the linear system for the Newton step has the structure

$$J^1 \begin{bmatrix} \delta\Theta \\ \delta\Lambda \end{bmatrix} = \begin{bmatrix} J_{\Theta\Theta} & J_{\Theta\Lambda} \\ J_{\Lambda\Theta} & J_{\Lambda\Lambda} \end{bmatrix} \begin{bmatrix} \delta\Theta \\ \delta\Lambda \end{bmatrix} = \begin{bmatrix} R_\Theta \\ R_\Lambda \end{bmatrix} = R^1. \tag{2}$$

The latter method further eliminates subdomain unknowns to form a matrix with the structure

$$J^2 \, \delta\Lambda = (J_{\Lambda\Lambda} - J_{\Lambda\Theta} J_{\Theta\Theta}^{-1} J_{\Theta\Lambda}) \, \delta\Lambda = R_\Lambda - J_{\Lambda\Theta} J_{\Theta\Theta}^{-1} R_\Theta = R^2. \tag{3}$$

This is also known as the linearized interface formulation, and we note that the action of $J_{\Theta\Theta}^{-1}$ can be applied by solving linear subdomain problems.

In this work, we describe a new global Jacobian algorithm where mortar variables are eliminated, referred to as a coupled subdomain formulation. This system has the structure

$$J^3 \, \delta\Theta = (J_{\Theta\Theta} - J_{\Theta\Lambda} J_{\Lambda\Lambda}^{-1} J_{\Lambda\Theta}) \, \delta\Theta = R_\Theta - J_{\Theta\Lambda} J_{\Lambda\Lambda}^{-1} R_\Lambda = R^3. \tag{4}$$

The algorithm to solve system (4) is described in Algorithm 1. Here the matrix inverse $J_{\Lambda\Lambda}^{-1}$ can be formed explicitly using sparse LU decomposition since it is extremely small and has completely decoupled diagonal blocks corresponding to each separate mortar interface and fluid phase. Alternatively, one may employ mass lumping to form the matrix inverse $\tilde{J}_{\Lambda\Lambda}^{-1}$, which in our experience does not hurt linear or nonlinear convergence, and makes a big difference in runtime for large scale problems. Figure 1 shows an example sparsity pattern for a non-matching grid problem: matrix J^1 shows how subdomain variables are coupled to mortar variables for elements adjacent to interfaces; matrix J^3 shows the new

```
Given initial condition $[\Theta^0, \Lambda^0]$,
for time steps $n = 1, \dots, N_T$ do
    $[\Theta^{n,0}, \Lambda^{n,0}] = [\Theta^{n-1}, \Lambda^{n-1}]$.
    for Newton steps $k = 0, \dots, NEWT\_MAX$ do
        Form residual $[R^1]^{n,k}$.
        if $(\|[R^1]^{n,k}\| < NEWT\_TOL)$ then break $k$-loop
        Form Jacobian $[J^1]^{n,k}$.
        Explicitly calculate $J_{\Lambda\Lambda}^{-1}$ using sparse LU (or $\tilde{J}_{\Lambda\Lambda}^{-1}$ using mass lumping).
        Form Jacobian $J^3$ and residual $R^3$ using (4).
        $\delta\Theta = \mathbf{gmres}(J^3, R^3)$ using the two-stage preconditioner $M^{-1}$ (see Algorithm 2).
        Calculate $\delta\Lambda$ using $J_{\Lambda\Lambda}^{-1}$ (or $\tilde{J}_{\Lambda\Lambda}^{-1}$).
        Add Newton increment $[\Theta^{n,k+1}, \Lambda^{n,k+1}] = [\Theta^{n,k} + \delta\Theta, \Lambda^{n,k} + \delta\Lambda]$.
    end for
    Save previous solution for next time step $[\Theta^n, \Lambda^n] = [\Theta^{n,k}, \Lambda^{n,k}]$.
end for
```

Algorithm 1.—A global Jacobian method to solve the MMMFEM using a Schur complement system for the δΘ variable.

Schur complement formulation where subdomains are directly coupled to each other adjacent to interfaces after elimination of the mortar variable; and matrix $\tilde{J}^3$ shows how the fullness can be decreased using mass lumping.

A Two-Stage Preconditoner

The goal of this section is to develop a two-stage preconditioner for system (4). Note the first N_P rows of the matrix J_3 group the unknowns for δP_O together by increasing element and then increasing subdomain. The second set of N_P rows follow the same ordering for the variable δN_0. We therefore must introduce the permutation matrix P that reorders unknowns $(\delta P_0, \delta N_0)_i$ together for each element i by order of increasing element then increasing subdomain. Consider the following example in Figure 2. The row order for matrix J would be $\delta[P_O^1, P_O^2, \dots, P_O^{13}, N_O^1, N_O^2, \dots, N_O^{13}]$, while the row order for matrix PJ would be $\delta[P_O^1, N_O^1, P_O^2, N_O^2 \dots, P_O^{13}, N_O^{13}]$.

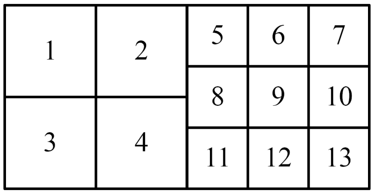


Figure 2.—Example of an element ordering for two subdomains with non-matching grids in two dimensions.

We first transform (4) using the Householder decoupling technique. For the fully-implicit two-phase system, consider the 2 × 2 diagonal blocks of the matrix PJ^3. Since variables are grouped together by element, call this matrix A = blockdiag(A_i), for i = 1, . . ., N_P. We perform a QL factorization on these blocks to decouple primary phase pressure from the other variables on the element level such that $Q_i^T A_i = L_i$, where Q_i is an orthogonal matrix and L_i is a lower triangular matrix. Then we define matrix Q = blockdiag(Q_i), and note that $(Q^T P J^3)$ will have lower triangular matrices (L_i) on its block diagonal. To complete the Householder step, we permute the matrix rows back to their original order by

$$(P^{-1} Q^T P J^3)\, \delta\Theta = P^{-1} Q^T P\, R^3 \tag{5}$$

$$\Leftrightarrow H\, \delta\Theta = \begin{bmatrix} H_{P_O P_O} & H_{P_O N_O} \\ H_{N_O P_O} & H_{N_O N_O} \end{bmatrix} \begin{bmatrix} \delta P_O \\ \delta N_O \end{bmatrix} = \begin{bmatrix} b_{P_O} \\ b_{N_O} \end{bmatrix} = b.$$

The two-stage preconditioned system will seek to solve $M^{-1} H\, \delta\Theta = M^{-1} b$ with GMRES (Saad and Schultz, 1986). The action of M^{-1} is summarized in Algorithm 2. Note the Householder transformed matrix H was specially formed to decouple the δP_0 unknown from the δN_0 unknown along the main diagonal of 2 × 2 blocks. The off-diagonal coupling of δP_0 to δN_0 still remains, but has been observed to be very weak in a single domain system (Lacroix, et al. 2003). Step 1 of the preconditioner solves the decoupled pressure system using preconditioner M_{1S}^{-1} , which is generally taken to be quite strong using e.g. algebraic multigrid with an incomplete LU smoother. Step 2 of the preconditioner computes a new

```
1. Solve the pressure equation $Y_{P_O} = \mathbf{gmres}(H_{P_O P_O}, Z_{P_O})$ with preconditioner $M_{1S}^{-1}$ to a specified tolerance.
2. Update the linear residual $R = Z - H[Y_{P_O}, 0]$.
3. Solve the second stage equation $Y = \mathbf{gmres}(H, R) + [Y_{P_O}, 0]$ with preconditioner $M_{2S}^{-1}$ to a specified tolerance.
```

Algorithm 2.—Computing the action of two-stage preconditioner $Y = M^{-1}Z$. This is used inside the outer preconditioned GMRES iteration given in Algorithm 1.

residual R using the solution from Step 1 to have a combinative effect, as opposed to a purely decoupled two-stage preconditioner. Finally, Step 3 solves the second stage system with the new residual R using the preconditioner M_{2S}^{-1} , which is generally taken to be quite weak using e.g. several Gauss-Seidel iterations. This is because a weak second stage solution usually does not drastically hurt the convergence of the outer GMRES iteration in Algorithm 1.

Numerical Results

The numerical results obtained in this paper were generated on the stampede supercomputer at Texas Advanced Computing Center. The parallelism is implemented using MPI where each subdomain is assigned to a separate CPU core. Our convention is that the positive x −coordinate represents the vertical direction, and gravitational acceleration is taken into account. All wells are oriented in the vertical direction, completed through the entire reservoir depth, and have diameter 0.5 [ft]. Unless otherwise specified, the capillary pressure, relative oil permeability, and relative water permeability curves use the following J-Leverett and Brooks-Corey relationships

$$p_c(s_w) = p_d s_e^{-\frac{1}{\lambda}}, \tag{6a}$$

$$\mathrm{k_{ro}(s_w)} = (1 - \mathrm{s_e})^2 \left(1 - \mathrm{s_e}^{\frac{2+\lambda}{\lambda}}\right), \tag{6b}$$

$$\mathrm{k_{rw}(s_w)} = \mathrm{s_e}^{\frac{2+3\,\lambda}{\lambda}}, \tag{6c}$$

where the effective saturation $s_e\ \varepsilon(0,1)$ is defined as

$$s_e = \frac{s_w - s_{rw}}{1 - s_{rw} - s_{ro}}. \tag{6d}$$

Example 1. Parallel Scaling for the SPE10 Benchmark Problem.

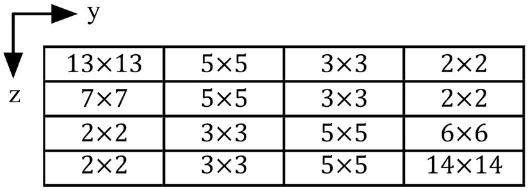
In this example, we show our method is capable of simulating two-phase flow with data from the challenging SPE10 benchmark problem (Christie and Blunt, 2001).

The domain size is 170 × 1200 × 2200 [ft] at a depth of 12000 [ft]. This domain is divided into N_Ω subdomains of roughly the same size. The mortar grids are piecewise constant, coincident with the trace of both adjacent subdomains. The final simulation time is $T = 5$ [days] and an adaptive time step is taken as follows: the initial time step size is $\delta t^1 = 1e - 2$ [days], the time step multipler is 1.25, the time step cut factor is 0.75, the minimum time step is $\delta t_{\min} = 1e - 6$ [days], and the maximum time step size is $t_{\min} = 1e - 1$ [days]. The absolute permeability, porosity, and relative permeability curves are taken from the SPE10 dataset (Christie and Blunt, 2001). The fluid compressibilities are $c_o = 4.2e - 5$ [1/psi] and $c_w = 3.1e - 6$ [1/psi], the fluid densities are $\rho_o^{ref} = 53$ [lb/ft^3] and $\rho_w^{ref} = 64$ [lb/ft^3], and fluid viscosities are $\mu_o = 3$ [cp] and $\mu_w = 0.3$ [cp]. The capillary pressure is zero in this case. A hydrostatic initial condition is assumed with initial oil pressure $p_o(0) = 6000$ [psi] and initial water saturation $s_w(0) = 0.201$ at a reference depth of 12170 [ft]. No-flow boundary conditions are assumed on $\partial\Omega$. There is one injection and one production well in a quarter five-spot well pattern, with bottom hole pressures specified at 8000 [psi] and 4000 [psi], respectively.

Table 1—Strong parallel scaling results for Example 1.

CPU cores/Subdomains	Total CPU time	Total Newton Steps Taken	Avg. Outer GMRES Iter. per Newton step	Time Step Cuts
1 × 1 × 1 = 1	8331.79	51	4.88	0
1 × 1 × 2 = 2	4675.22	51	5.00	0
1 × 1 × 4 = 4	3102.14	52	5.65	1
1 × 2 × 4 = 8	2727.95	51	5.04	0
1 × 2 × 8 = 16	1216.14	52	5.71	1
1 × 4 × 8 = 32	517.69	51	5.02	0
1 × 4 × 16 = 64	618.41	109	5.71	2

The Newton tolerance is $1e-4$ with a maximum of 5 iterations, and the outer GMRES tolerance is $1e-5$, has a maximum of 20 iterations with no restarts, and uses the two-stage preconditioner. The pressure solve uses GMRES with tolerance $1e-6$, has a maximum of 100 iterations with a restart every 20 iterations. M_{1S}^{-1} is an algebraic multigrid V-cycle with an ILU(0) smoother with 1 sweep, and the coarsest level has size less than or equal to 1000 × 1000 and uses a Sparse LU direct solver. The second stage solve uses GMRES with tolerance $1e-3$, has a maximum of 20 iterations no restarts. M_{2S}^{-1} is an algebraic multigrid V-cycle with an ILU(0) smoother with 1 sweep, and the coarsest level has size less than or equal to 1000 × 1000 and uses a Sparse LU direct solver.



13×13	5×5	3×3	2×2
7×7	5×5	3×3	2×2
2×2	3×3	5×5	6×6
2×2	3×3	5×5	14×14

Figure 3.—Areal grid dimensions for subdomains in Example 2.

Table 1 summarizes the statistics on iteration counts and CPU times for Example 1. We have used algebraic multigrid preconditioners in both pressure and second stage solves because of the highly heterogeneous nature of the dataset. Strong parallel scaling was achieved upto 64 CPU cores. Note that since we take 1 CPU core per subdomain, we are actually changing the size and condition number of the problem as we increase the number of processors. This is in contrast to data decomposition on a single domain, which would have the same system for varying number of processors. Our domain decomposition approach in concert with our two-stage preconditioner was stable in this problem upto 32 processors. At 64 processors, the number of Newton steps increased significantly causing an increase in Total CPU time, although the two-stage preconditioner remained effective.

Example 2. A Multiscale Problem on Non-matching Grids

In this example, we highlight the multiscale modeling capabilities of the MMMFEM on non-matching grids.

The domain size is 10 × 100 × 100 [ft], divided into $N_{\Omega} = 1 \times 4 \times 4 = 16$ subdomains. Each subdomain has a uniform grid in the areal dimensions with dimensions given in Figure 3 illustrating the refinement around wells. The number of vertical layers is obtained by dividing by a factor of 3. The mortar grids are piecewise constant grids taken to be the trace of the adjacent coarsest subdomain. The final simulation time is $T = 100$ [days] and a uniform time step of $\delta t = 0.1$ [days] is used. The absolute permeability is $K = \text{diag}(100{,}200{,}200)$ [md], porosity is $\phi = 0.2$, fluid compressibilities are $c_o = 4e-5$ [1/psi] and $c_w = 3.3e-6$ [1/psi], the fluid densities are $\rho_o^{ref} = 53$ [lb/ft^3] and $\rho_w^{ref} = 64$ [lb/ft^3], and fluid viscosities are $\mu_o = 2$ [cp] and $\mu_w = 0.5$ [cp]. The capillary pressure and relative permeability curves follow (6a)–(6c) with residual water saturation $s_{rw} = 0.2$, residual oil saturation $s_{ro} = 0.2$, grain size parameter $\lambda = 2$, and entry pressure $p_d = 10$[psi]. A hydrostatic initial condition is assumed with initial oil pressure $p_o(0) = 500$ [psi] and initial water saturation $s_w(0) = 0.22$ at a reference depth of 10 [ft]. No-flow boundary conditions are assumed on $\partial\Omega$. There is one injection and one production well in a quarter five-spot well pattern, with bottom hole pressures specified at 520 [psi] and 480 [psi], respectively.

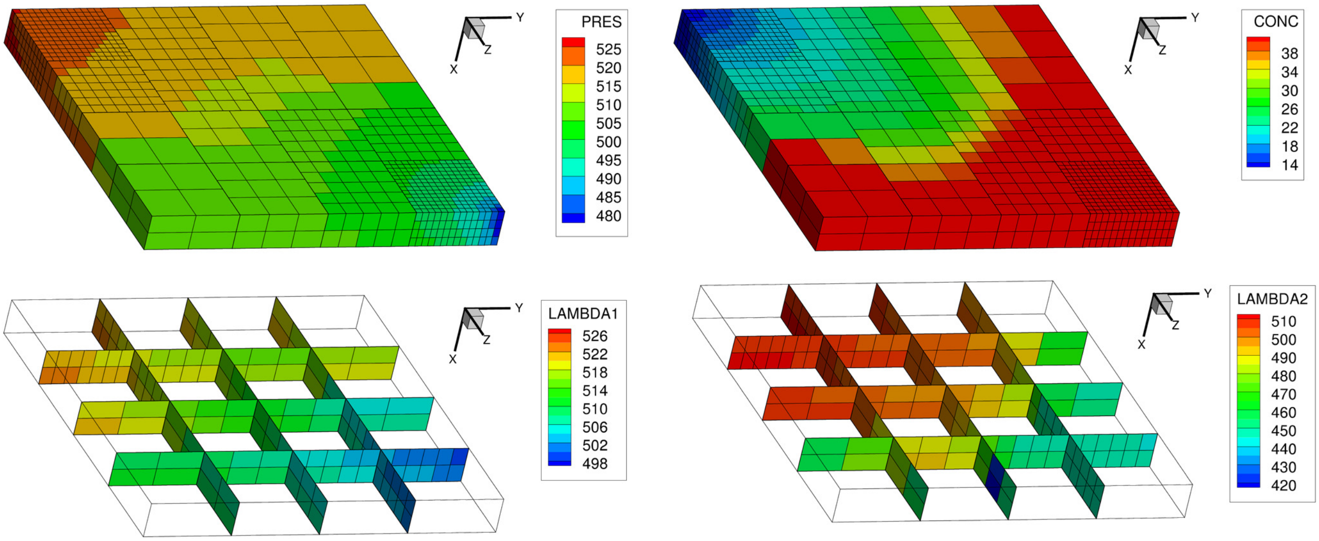


Figure 4—Simulation results at time t = 100 days showing four independent solution variables: subdomain variable P_o (top left), subdomain variable N_o (top right), mortar variable Λ_1 (bottom left), and mortar variable Λ_2 bottom right).

Table 2—Iteration counts (left) and timing statistics (right) for Example 2.

		Matrix assembly time	18.72
Total time steps	1001	Outer GMRES time	91.68
Total Newton iterations	1013	Householder decoupling time	3.51
Total outer GMRES iterations	14867	Pressure solve GMRES time	45.94
Average GMRES iterations per Newton step	14.68	Second stage GMRES time	32.26
Average Newton iterations per time step	1.01	Mass lumping time	0.06
Total time step cuts	0	Matrix-matrix multiply time	0.54
		Total CPU time	118.89

The Newton tolerance is $1e-4$ with a maximum of 20 iterations, and the outer GMRES tolerance is $1e-6$, has a maximum of 50 iterations, restarts after every 20 iterations, and uses the two-stage preconditioner. The pressure solve uses GMRES with tolerance $1e-3$, has a maximum of 20 iterations with no restarts. M_{1S}^{-1} is an algebraic multigrid V-cycle with an ILU(0) smoother with 1 sweep, and the coarsest level has size less than or equal to 1000 × 1000 and uses a Sparse LU direct solver. The second stage preconditioner M_{2S}^{-1} is a simple application of 5 Gauss-Seidel iterations.

Figure 4 shows the simulation results at the final time, and Table 2 shows statistics on iteration counts and CPU times. The effect of the two-stage preconditioner caused the outer GMRES iteration to take on average 14.44 iterations per Newton step. We note that this number can be reduced to 2–3 iterations with a stronger M_{2S}^{-1} preconditioner, but Total CPU time would increase. For optimal CPU time, the best practice is to choose the weakest second stage as possible, which still allows the outer GMRES iteration to converge.

Example 3. Simulation of a Heterogeneous 10 Million Element Problem Using 1024 Processors

In this example, we show numerical results for a large scale problem with 10 million elements, and show the efficiency of our two-stage preconditioner when using 1024 processors.

The absolute permeability field K is heterogeneous, where each vertical layer is generated from independent realizations of a scalar 2D stochastic geostatistical model. A Karhunen-Loève (KL) expansion for the mean removed log permeability Y is computed from the specified covariance function

$$C_Y(\boldsymbol{x},\bar{\boldsymbol{x}}) = \sigma_Y^2 \exp\left[-\frac{|x_1-\bar{x}_1|}{\eta_1} - \frac{|x_2-\bar{x}_2|}{\eta_2}\right]. \tag{7}$$

Table 3—Iteration counts (left) and timing statistics (right) for Example 3.

		Matrix assembly time	86.04
Total time steps	1007	Outer GMRES time	8459.16
Total Newton iterations	1007	Householder decoupling time	42.25
Total outer GMRES iterations	2449	Pressure solve GMRES time	1394.55
Average GMRES iterations per Newton step	2.43	Second stage GMRES time	3340.99
Average Newton iterations per time step	1.00	Mass lumping time	0.05
Total time step cuts	0	Matrix-matrix multiply time	1206.87
		Total CPU time	8571.76

The parameters are correlation lengths $\eta_1 = 17$ [ft] and $\eta_2 = 30$ [ft], variance $\sigma_Y = 5$ [log(md)], mean $\bar{Y} = 2.1$ [log(md)], and the series was truncated after 400 terms. For complete details on the calculation of eigenvalues and eigenfunctions for this series, see e.g. Zhang and Lu 2004. The exponential of this series becomes the x component of the diagonal permeability tensor, and the y and z permeability components are multiplied by a factor of 2.

The domain size is $100 \times 1700 \times 3000$ [ft] with a uniform grid of $80 \times 272 \times 480 \approx 10.44$ million elements, evenly divided into $N_\Omega = 4 \times 16 \times 16 = 1024$ subdomains. The mortar grids are piecewise constant, coincident with the trace of both adjacent subdomains. The final simulation time is $T = 100$ [days] and an adaptive time step is taken as follows: the initial time step size is $\delta t^1 = 0.1$ [days], the time step multipler is 1.25, the time step cut factor is 0.75, the minimum time step is $\delta t_{\min} = 1e - 8$ [days], and the maximum time step size is $t_{\min} = 1\mathrm{e} - 1$ [days]. The porosity is $\phi = 0.2$; , fluid compressibilities are $c_o = 4e - 5$ [1/psi] and $c_w = 3.3e - 6$ [1/psi], the fluid densities are $\rho_o^{ref} = 56.03$ [lb/ft^3] and $\rho_w^{ref} = 62.34$ [lb/ft^3], and fluid viscosities are $\mu_o = 2$ [cp] and $\mu_w = 0.5$ [cp]. The capillary pressure and relative permeability curves follow (6a)–(6c) with residual water saturation $s_{rw} = 0.2$, residual oil saturation $s_{ro} = 0.2$, grain size parameter $\lambda = 2$, and entry pressure $p_d = 10$ [psi]. A hydrostatic initial condition is assumed with initial oil pressure $p_o(0) = 6000$ [psi] and initial water saturation $s_w(0) = 0.22$ at a reference depth of 20 [ft]. No-flow boundary conditions are assumed on $\partial\Omega$. There is one injection and one production well in a quarter five-spot well pattern, with bottom hole pressures specified at 8000 [psi] and 4000 [psi], respectively.

The Newton tolerance is $1e - 4$ with a maximum of 5 iterations, and the outer GMRES tolerance is $1e - 5$, has a maximum of 5 iterations with no restarts, and uses the two-stage preconditioner. The pressure solve uses GMRES with tolerance $1e - 3$, has a maximum of 20 iterations with no restarts. M_{1S}^{-1} is an algebraic multigrid V-cycle with an ILU(0) smoother with 1 sweep, and the coarsest level has size less than or equal to 1000×1000 and uses a Sparse LU direct solver. In this case, the second stage solve and M_{2S}^{-1} preconditoner are taken to be the same as the pressure solve and M_{1S}^{-1} preconditioner.

Table 3 summarizes the statistics on iteration counts and CPU times for Example 3. The effect of the two-stage preconditioner caused the outer GMRES iteration to take on average 2.43 iterations per Newton step. This number is much smaller than Example 2 because the second stage solve was taken to be much stronger. It was necessary to do this for this problem, because convergence for the outer GMRES iteration was lost when using a weaker second stage. We have found this behavior to be typical for very large scale heterogeneous problems.

Conclusions

In this work we have presented an efficient parallel implementation of a multiscale mortar mixed finite element method for fully-implicit two-phase flow using a global Jacobian approach, ensuring local mass conservation and accurate flux. The algorithm eliminates both velocities and Lagrange multiplier variables. A two-stage precondioning strategy was developed and demonstrated in numerical results. The two-stage preconditioner decouples the pressure solve and the saturation update for a fully-implicit method. These individual components are solved by using appropriate solvers, such as multigrid for pressure with ILU as a preconditioner and a Gauss-Seidel solver for the second stage with a loose tolerance. Examples show grid refinement can be done near regions of interest leading to subdomains with independent non-matching discretizations, and that our method is capable of solving a challenging industrial benchmark case.

The linear systems arising from mortar discretizations of fully-implicit two-phase flow with gravity, capillarity, and compressibility in three dimensions are very challenging. The two-stage precondioner that we have developed provides an efficient way to solve these linear systems. In one of our examples, we were able to achieve an average of 2.3 GMRES iterations per Newton step for a problem on the order of 20 million unknowns, and were able to utilize over 1000 processors in parallel.

Acknowledgements

This material is based upon work supported by DOE grant ER25617 and the Center for Frontiers of Subsurface Energy Security, an Energy Frontier Research Center funded by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences under Award Number DE-SC0001114. The authors acknowledge the Texas Advanced Computing Center (TACC) at The University of Texas at Austin for providing HPC resources that have contributed to the research results reported within this paper.

Nomenclature

FEM	: Finite Element Method
RTN	: Raviart-Thomas-Nedelec
MMMFEM	: Multiscale Mortar Mixed Finite Element Method
FD	: Finite Difference
GJ	: Global Jacobian
GJS	=: Global Jacobian Schur
GMRES	=: Generalized Minimum Risidual
α	= Phase α=o, w, oil or water
Ω	= Reservoir domain
$\partial\Omega$	= Reservoir external boundary
Ω^k	= k-th Subdomain
Γ^{kl}	= Interface between k-th and l-th subdomains
t	= Simulation Time [day]
T	= Final Simulation Time [day]
ϕ	= Porosity
s_α	= Saturation of phase α
s_e	= Effective saturation of phase α
p_α	= Pressure of phase α [psi]
p_c	= Capillary pressure [psi]
p_d	= Entry pressure in Brooks-Corey relationship
n_α	= Concentration of phase α [lb/ft^3]
p_α	= Density of phase α [lb/ft^3]
N_p	= Total degrees of freedom for pressure unknowns
N_λ	= Total degrees of freedom for Lagrange multiplier unknowns
P_o	= Coefficient unknown vector for oil pressure
N_o	= Coefficient unknown vector for oil concentration
Λ_α	= Coefficient unknown vector for Lagrange multipliers for phase α

J_1	= Jacobian matrix obtained using global Jacobian
J_2	= Jacobian matrix obtained after eliminating subdomain unknowns
J_3	= Jacobian matrix obtained after eliminating interface unknowns
δP_o	= Newton update for oil pressure
δN_o	= Newton update for oil concentration
$\delta\Lambda$	= Newton update for Lagrange multipliers
$\delta\Theta$	= Newton update for subdomain unknowns
$\mathbf{u}_\alpha$	= Darcy flux of phase α [lb/(ft^2-day)]
$\tilde{\mathbf{u}}_\alpha$	= Auxiliary flux of phase α [lb/ft]
λ_α	= Lagrange multiplier corresponding to phase α
m_α	= Mobility of phase α
μ_α	= Viscosity of phase[cp] α
c_α	= Compressibility of phase α [psi^{-1}]
$k_{r\alpha}$	= Relative permeability of phase α
q_α	= Source or sink term for phase α [lb/(ft^3-day)]
K	= Absolute permeability [Darcy]
$\mathbf{g}$	= Acceleration due to gravity [ft-day^{-2}]

References

Arbogast, T., Cowsar, L.C., Wheeler, M.F. and Yotov, I. 2000. Mixed finite element methods on nonmatching multiblock grids. *SIAM Journal on Numerical Analysis* **37** (4): 1295–1315.

Arbogast, T., Pencheva, G., Wheeler, M.F., and Yotov, I. 2007. A multiscale mortar mixed finite element method. *Multiscale Modeling & Simulation* **6** (1): 319–346.

Arbogast, T., Wheeler, M.F., and Yotov, I. 1997. Mixed finite elements for elliptic problems with tensor coefficients as cell-centered finite differences. *SIAM Journal on Numerical Analysis* **34** (2): 828–852.

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

Cao, H., Tchelepi, H.A., Wallis, J.R., et al. 2005. Parallel scalable unstructured cpr-type linear solver for reservoir simulation. In *SPE Annual Technical Conference and Exhibition*. Society of Petroleum Engineers.

Dawson, C.N., Klíe, H., Wheeler, M.F., and Woodward, C.S. 1997. A parallel, implicit, cell-centered method for two-phase flow with a preconditioned Newton-Krylov solver. *Computational Geosciences* **1** (3–4): 215–249.

Ganis, B., Juntunen, M., Pencheva, G., Wheeler, M.F., and Yotov, I. 2014. A global Jacobian method for mortar discretizations of nonlinear porous media flows. *SIAM Journal on Scientific Computation* **36** (2): A522–A542.

Ganis, B., Kumar, K., Pencheva, G., Wheeler, M.F., and Yotov, I. 2014. A global Jacobian method for mortar discretizations of a fully-implicit two-phase flow model. *Multiscale Modeling & Simulation, accepted*.

Ganis, B., Pencheva, G., Wheeler, M.F., Wildey, T., and Yotov, I. 2012. A frozen Jacobian multiscale mortar preconditioner for nonlinear interface operators. *Multiscale Modeling & Simulation,* **10** (3): 853–873.

Lacroix, S., Vassilevski, Y., Wheeler, J., and Wheeler, M.F. 2003. Iterative solution methods for modeling multiphase flow in porous media fully implicitly. *SIAM Journal on Scientific Computation* **25** (3): 905–926.

Nédélec, J.-C. 1980. Mixed finite elements in $\mathbb{R}^3$. *Numerische Mathematik,* **35** (3): 315–341.

Peszynska, M., Wheeler, M.F., and Yotov, I. 2002. Mortar upscaling for multiphase flow in porous media. *Computational Geosciences* **6** (1): 73–100.

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

Saad, Y., and Schultz, M. H. 1986. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on scientific and statistical computing,* **7** (3): 856–869.

Toselli, A. and Widlund, O.B. 2005. *Domain decomposition methods–algorithms and theory,* volume **34**. Heidelberg, Germany: Springer Verlag.

Vassilevski, P.S. 1984. Fast algorithm for solving a linear algebraic problem with separable variables. *Dokladi Na Bolgarskata Akademiya Na Naukite,* **37** (3): 305–308.

Wallis, J.R., Kendall, R.P., and Little, T.E. 1985. Constrained residual acceleration of conjugate residual methods. In *SPE Reservoir Simulation Symposium,* SPE 13536.

Yotov, I. 2001. A multilevel Newton–Krylov interface solver for multiphysics couplings of flow in porous media. *Numerical Linear Algebra and Applications,* **8** (8): 551–570.

Zhang, D., Lu, Z. 2004. An efficient, high-order perturbation approach for flow in random porous media via Karhunen-Loève and polynomial expansions. *Journal of Computational Physics* **194** (2): 773–794.