Journal of Computational Physics 259 (2014) 284–303

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

Algebraic multiscale solver for flow in heterogeneous porous media

Yixuan Wang [a,*], Hadi Hajibeygi [b,a], Hamdi A. Tchelepi [a]

[a] Department of Energy Resources Engineering, Stanford University, 367 Panama St., Rm. 065, Stanford, CA 94305-2220, USA
[b] Department of Geoscience and Engineering, Faculty of Civil Engineering and Geosciences, Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands

ARTICLE INFO

Article history:
Received 6 June 2013
Received in revised form 20 November 2013
Accepted 24 November 2013
Available online 6 December 2013

Keywords:
Multiscale methods
Iterative multiscale methods
Algebraic multiscale solver
Scalable linear solvers

ABSTRACT

An Algebraic Multiscale Solver (AMS) for the pressure equations arising from incompressible flow in heterogeneous porous media is described. In addition to the fine-scale system of equations, AMS requires information about the superimposed multiscale (dual and primal) coarse grids. AMS employs a global solver only at the coarse scale and allows for several types of local preconditioners at the fine scale. The convergence properties of AMS are studied for various combinations of global and local stages. These include MultiScale Finite-Element (MSFE) and MultiScale Finite-Volume (MSFV) methods as the global stage, and Correction Functions (CF), Block Incomplete Lower–Upper factorization (BILU), and ILU as local stages. The performance of the different preconditioning options is analyzed for a wide range of challenging test cases. The best overall performance is obtained by combining MSFE and ILU as the global and local preconditioners, respectively, followed by MSFV to ensure local mass conservation. Comparison between AMS and a widely used Algebraic MultiGrid (AMG) solver [1] indicates that AMS is quite efficient. A very important advantage of AMS is that a conservative fine-scale velocity can be constructed after any MSFV stage.

1. Introduction

Numerical simulation of multiphase flow in large-scale heterogeneous reservoirs is computationally demanding. To reduce the computational complexity, several MultiScale (MS) methods have been developed [2–10]. In MS methods, the global fine-scale problem is decomposed into large numbers of local problems. Basis functions, which are numerical solutions of local problems, are used to construct accurate coarse-scale quantities. Once the coarse-scale system is solved, the solution is mapped onto the fine scale using the basis functions. Among the proposed multiscale methods, the Mixed MultiScale Finite-Element (MMSFE) [6,11,5,8] and the MultiScale Finite Volume (MSFV) [7] methods provide locally mass-conservative solutions, which is a crucial property for solving coupled flow and transport problems.

The MSFV method employs locally computed basis functions to construct the coarse-scale system in a finite-volume framework. To obtain a locally conservative velocity field at the fine scale, additional local Neumann problems are constructed over the primal coarse control volumes. Recent developments of the MSFV method include incorporating the effects of compressibility [12,13], gravity and capillary [14], complex wells [15,16], faults [17], fractures [18], three-phase [19] and compositional displacements [20]. Furthermore, the efficiency of the method has been enhanced by adaptive computation of the basis functions for multiphase, time-dependent displacement problems [21–24].

* Corresponding author.
E-mail addresses: yixuanw@stanford.edu (Y. Wang), h.hajibeygi@tudelft.nl (H. Hajibeygi), tchelepi@stanford.edu (H.A. Tchelepi).

0021-9991/$ – see front matter
http://dx.doi.org/10.1016/j.jcp.2013.11.024

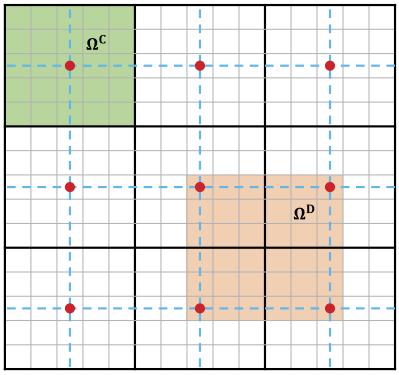


Fig. 1. Primal (bold black) and dual (dashed blue) coarse cells. Fine-cells belonging to a coarse cell (control volume) are shown in green. Fine-cells that belong to a dual coarse cell are highlighted in light orange. The red circles denote the coarse nodes (vertices). (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

For a wide range of heterogeneous test cases, the MSFV results are shown to be in good agreement with reference fine-scale solutions. However, the accuracy of MSFV method suffers from the presence of extreme permeability contrasts (e.g., SPE 10 bottom section [25]) or highly anisotropic problems (e.g., large grid aspect ratios) [26]. To overcome these limitations, the iterative MSFV (i-MSFV) method was introduced [27], where the MSFV errors were systematically reduced with the help of locally computed Correction Functions (CF). The i-MSFV had weak convergence performance for highly heterogeneous and anisotropic problems [28]. The convergence rate was improved significantly by using the MSFE operator (i-MSFE) [29,30]. The benefit of the MSFV operator is that a mass-conservative solution is obtained. Thus, MSFV can be employed at the end of the iterative process to ensure that the approximate solution is conservative.

Both the original (single-pass) and iterative multiscale methods can be formulated in an algebraic manner [13,30]. The algebraic formulation reduces the implementation complexity, especially for problems defined on unstructured grids, and it allows for easy integration of the method into existing reservoir simulators. The Two-Stage Algebraic Multiscale Solver (TAMS) [30] consists of local and global stages. In the global stage, low frequency errors are resolved by a multiscale preconditioner. In the local stage, high frequency errors are resolved using a Block ILU with zero fill-in (BILU) [31] local solver. However, CF was not incorporated into TAMS, and the exact role of CF in the context of multi-stage preconditioning had not been analyzed. In addition, the best choices among the variety of possible local and global stages have not been thoroughly investigated.

In this paper, a general iterative Algebraic Multiscale Solver (AMS) is described. AMS allows for MSFV, or MSFE, as global operators with different types of local boundary conditions, and it allows for many local fine-scale solvers, e.g., BILU and Line-Relaxation (LR) [32]. We show that the CF is an independent local preconditioning stage aimed at resolving high-frequency errors. The effects of the CF local stage on the AMS convergence rate and the overall computational efficiency for several heterogeneous problems are analyzed. To obtain the best combination of methods, we provide systematic performance tests considering different global (MSFV and MSFE) and local (BILU, CF, ILU) stages with different local boundary conditions. Then, the computational efficiency of AMS is compared with an advanced algebraic multigrid solver, SAMG, developed at Fraunhofer SCAI [1].

The paper is organized as follows. First, a general Algebraic Multiscale Solver (AMS) for heterogeneous elliptic problems is developed. Second, the effects of CF are analyzed. Then, numerical results are presented. The conclusions are given in the final section.

2. Algebraic Multiscale Solver (AMS)

In this section, the original MSFV method is reviewed briefly. Then, the Algebraic Multiscale Solver (AMS) is described.

2.1. MultiScale Finite Volume (MSFV) method

The pressure equation for single-phase incompressible flow can be written as

$$\nabla \cdot (\boldsymbol{\lambda} \cdot \nabla p) = \nabla \cdot (\rho g \boldsymbol{\lambda} \cdot \nabla z) + \tilde{q}, \tag{1}$$

where $\boldsymbol{\lambda}$ is the positive-definite mobility tensor, $\tilde{q}$ represents source terms, g is the gravitational acceleration acting in the ∇z direction, and ρ is the density.

The MSFV grid consists of two sets of overlapping coarse grids, namely primal and dual coarse grids, superimposed on the given fine grid (Fig. 1). There are N_C primal coarse cells (control volumes), Ω_i^C $(i \in \{1, \ldots, N_C\})$, and N_D dual-coarse cells (local domains), Ω_j^D $(j \in \{1, \ldots, N_D\})$.

The basis functions in the MSFV and MSFE methods are obtained by solving

$$\begin{cases} \nabla \cdot \left(\boldsymbol{\lambda} \cdot \nabla \phi_j^i\right) = 0 & \text{in } \Omega_j^D \\ \nabla_{\parallel} \cdot \left(\boldsymbol{\lambda} \cdot \nabla \phi_j^i\right)_{\parallel} = 0 & \text{on } \partial\Omega_j^D \\ \phi_j^i(\mathbf{x}_k) = \delta_{ik} & \forall \mathbf{x}_k \in \{1, \ldots, N_C\}, \end{cases} \tag{2}$$

where ϕ_j^i denotes the basis function associated with coarse node i in dual coarse block Ω_j^D [2,7]. The subscript $\parallel$ indicates the vector (or operator) projected along the tangential direction of the dual-coarse cell boundary, $\partial\Omega_j^D$. The CFs [14], which account for fine-scale RHS terms, are local particular solutions, and they are computed as follows:

$$\begin{cases} \nabla \cdot \left(\boldsymbol{\lambda} \cdot \nabla \phi_j^*\right) = \nabla \cdot (\rho g \boldsymbol{\lambda} \cdot \nabla z) + \tilde{q} & \text{in } \Omega_j^D \\ \nabla_{\parallel} \cdot \left(\boldsymbol{\lambda} \cdot \nabla \phi_j^*\right)_{\parallel} = \nabla_{\parallel} \cdot (\rho g \boldsymbol{\lambda} \cdot \nabla z)_{\parallel} + \tilde{q} & \text{on } \partial\Omega_j^D \\ \phi_j^*(\mathbf{x}_k) = 0 & \forall \mathbf{x}_k \in \{1, \ldots, N_C\}, \end{cases} \tag{3}$$

where ϕ_j^* is the CF in dual-coarse block Ω_j^D. Then, the approximate solution p' is obtained by using the superposition expression

$$p \approx p' = \sum_{j=1}^{N_D} \left[\sum_{i=1}^{N_C} \phi_j^i p_C^i + \phi_j^* \right], \tag{4}$$

where p_C^i is the coarse-scale solution at node i. The coarse-scale system is constructed by first substituting Eq. (4) into Eq. (1), and then integrating over the primal coarse control-volumes, Ω_i^C, which after using the divergence theorem can be expressed as

$$A_C p_C = R_C, \tag{5}$$

with

$$A_C(i,j) = -\sum_{d=1}^{N_D} \int_{\partial\Omega_i^C \cap \Omega_d^D} \left(\boldsymbol{\lambda} \cdot \nabla \phi_d^j\right) \cdot \vec{n}\, d\Gamma \tag{6}$$

and

$$R_C(i) = \sum_{d=1}^{N_D} \int_{\partial\Omega_i^C \cap \Omega_d^D} \left(\boldsymbol{\lambda} \cdot \nabla \phi_d^*\right) \cdot \vec{n}\, d\Gamma - \int_{\Omega_i^C} r\, dv, \quad i \in \{1, \ldots, N_C\}, \tag{7}$$

entries. Here, $\vec{n}$ is the unit-normal vector pointing outward, and r represents the RHS of Eq. (1). After solving Eq. (5) for the coarse-scale pressure, p_C, Eq. (4) is used again to obtain an approximate fine-scale solution, p'. While $u_t' = -(\boldsymbol{\lambda} \cdot \nabla p')$ is conservative at the primal coarse scale by construction, it is not conservative at the fine scale. Therefore, additional local Neumann problems on primal coarse control volumes are solved in order to obtain a conservative fine-scale velocity field [27].

2.2. Algebraic Multiscale Solver (AMS)

Eq. (1) discretized on the fine grid, can be written as

$$Ap = q. \tag{8}$$

For 2D problems on structured grids, the dual-coarse grid divides the fine cells into three categories: interior (white), edge (blue), and vertex (red) cells, as illustrated in Fig. 2 [33,34]. As the figure indicates, vertices are the coarse-grid nodes, and the edge cells are located on the boundaries of the dual cells. For 3D problems on structured grids, an additional category is face cells. Finally, internal cells are those that lie inside dual-coarse cells. For simplicity, the framework is described for 2D problems, although the implementation is 3D.

A wirebasket reordered fine-scale system [34] can be expressed as

$$\begin{bmatrix} A_{II} & A_{IE} & 0 \\ A_{EI} & A_{EE} & A_{EV} \\ 0 & A_{VE} & A_{VV} \end{bmatrix} \begin{bmatrix} p_I \\ p_E \\ p_V \end{bmatrix} = \begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix}, \tag{9}$$

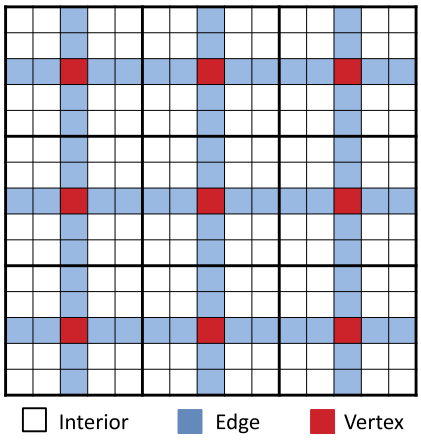


Fig. 2. Ordering of the fine cells based on the imposed dual-coarse grid. Also shown with bold solid lines is the primal coarse grid.

where a local matrix A_{ij} represents the contribution of cell j to the discrete mass conservation equation of cell i. The gravitational source terms (q^G) are separated from the rest of the RHS term, ($\tilde{q}$), because q^G requires special treatment according to Eq. (3). The reordered RHS vector is therefore rewritten as

$$\begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix} = \begin{bmatrix} q_I^G \\ q_E^G \\ q_V^G \end{bmatrix} + \begin{bmatrix} \tilde{q}_I \\ \tilde{q}_E \\ \tilde{q}_V \end{bmatrix} = B \begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix} + (I - B) \begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix}, \tag{10}$$

where B is a diagonal matrix with $B_{ii} = q_i^G / q_i$ entries.

According to Eq. (3), only the tangential component of the gravitational source term for edge cells is considered in the local problems for CF. Therefore, q_E^G is split into tangential $q_{E\parallel}^G$ and normal $q_{E\perp}^G$ components. Thus, the RHS term used for CF can be stated as

$$\begin{bmatrix} q_I \\ q'_E \\ q_V \end{bmatrix} = \begin{bmatrix} q_I^G \\ q_{E\parallel}^G \\ q_V^G \end{bmatrix} + \begin{bmatrix} \tilde{q}_I \\ \tilde{q}_E \\ \tilde{q}_V \end{bmatrix} = \mathcal{E} \begin{bmatrix} q_I^G \\ q_E^G \\ q_V^G \end{bmatrix} + \begin{bmatrix} \tilde{q}_I \\ \tilde{q}_E \\ \tilde{q}_V \end{bmatrix} = (\mathcal{E}B + I - B) \begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix}, \tag{11}$$

where $\mathcal{E}$ is a diagonal matrix with

$$\mathcal{E}_{ii} = \begin{cases} |\vec{n}_{e,i} \cdot \vec{n}_g| & \text{if } i \in \aleph_{edge} \\ 1 & \text{otherwise} \end{cases} \tag{12}$$

entries. Here, $\aleph_{edge}$ is the set of edge cells, $\vec{n}_{e,i}$ is the unit-vector tangent to the edge cells at cell i, and $\vec{n}_g$ is the unit vector parallel to gravitational acceleration. Finally, the RHS terms are expressed as:

$$\begin{bmatrix} q_I \\ q'_E \\ q_V \end{bmatrix} = E \begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix}, \tag{13}$$

where $E = \mathcal{E}B + I - B$.

The matrix entries for interior cells are preserved in the approximate multiscale operator. The stencil for edge cells, however, is modified to reflect the localization assumption. In fact, the only source of error in the multiscale approximation is due to the localization assumption. The reduced problem condition on edge cells is obtained by setting A_{EI} and its corresponding part in A_{EE} to zero. Finally, the multiscale approximate system is expressed as

$$\begin{bmatrix} A_{II} & A_{IE} & 0 \\ 0 & \tilde{A}_{EE} & A_{EV} \\ 0 & 0 & A_C \end{bmatrix} \begin{bmatrix} p'_I \\ p'_E \\ p'_V \end{bmatrix} = \begin{bmatrix} q_I \\ q'_E \\ R_C \end{bmatrix}, \tag{14}$$

where $A_C p'_V = R_C$ is the coarse-scale system that is solved for p'_V, where p'_V corresponds to p_C in Eq. (5). It is clear that the multiscale system is upper-triangular; hence, it is easy to invert. Note that TAMS [30] did not account for q_I and q_E terms. In our AMS framework, we account for RHS terms, including gravitational effects.

Once the coarse system $A_C p'_V = R_C$ is solved, the pressure for the edges and interior cells is obtained using backward substitution, i.e.,

$$p'_E = -\tilde{A}_{EE}^{-1}\left(A_{EV}p'_V - q'_E\right)$$
$$p'_I = -A_{II}^{-1}\left(A_{IE}p'_E - q_I\right) = A_{II}^{-1}\left(A_{IE}\tilde{A}_{EE}^{-1}\left(A_{EV}p'_V - q'_E\right) + q_I\right). \tag{15}$$

Finally, the multiscale approximate solution, p', is expressed as

$$\begin{bmatrix} p'_I \\ p'_E \\ p'_V \end{bmatrix} = \begin{bmatrix} A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1}A_{EV} \\ -\tilde{A}_{EE}^{-1}A_{EV} \\ I_{VV} \end{bmatrix} p'_V + \begin{bmatrix} A_{II}^{-1} & -A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1} & 0 \\ 0 & \tilde{A}_{EE}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_I \\ q'_E \\ q_V \end{bmatrix}, \tag{16}$$

where, I_{VV} is an $N_C \times N_C$ identity matrix. The prolongation operator is defined as

$$\mathcal{P} = G \begin{bmatrix} A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1}A_{EV} \\ -\tilde{A}_{EE}^{-1}A_{EV} \\ I_{VV} \end{bmatrix}, \tag{17}$$

where G is the permutation matrix that transforms the elements from wirebasket ordering into natural ordering. Even though the multiscale formulation accounts for the RHS, those terms do not appear in the prolongation operator. In fact, the CF pressure in natural ordering p^{corr} is expressed as

$$p^{corr} = G \begin{bmatrix} A_{II}^{-1} & -A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1} & 0 \\ 0 & \tilde{A}_{EE}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} q_I \\ q'_E \\ q_V \end{bmatrix}, \tag{18}$$

which indicates that CF solves the same reduced problem for edge cells (in 3D for face cells also) as the basis functions. The last column is zero because the vertices are disconnected from both edge and interior cells. Since the permutation matrix is orthogonal, i.e., $G^T = G^{-1}$, one can write

$$\begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix} = G^T q. \tag{19}$$

Finally, using Eqs. (13), (18), and (19), the CF pressure is related to the original RHS vector as follows

$$p^{corr} = G \begin{bmatrix} A_{II}^{-1} & -A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1} & 0 \\ 0 & \tilde{A}_{EE}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} EG^T q. \tag{20}$$

This equation can be simplified further by defining the correction operator, $\mathcal{C}$, in natural order, i.e.,

$$p^{corr} = \mathcal{C}q, \tag{21}$$

where

$$\mathcal{C} = G \begin{bmatrix} A_{II}^{-1} & -A_{II}^{-1}A_{IE}\tilde{A}_{EE}^{-1} & 0 \\ 0 & \tilde{A}_{EE}^{-1} & 0 \\ 0 & 0 & 0 \end{bmatrix} EG^T. \tag{22}$$

Here, E is an extraction diagonal $N_F \times N_F$ matrix (N_F is the number of fine cells) that includes the gravity term modification for edge cells. The modification of the gravity term at the edge cells is employed only for the first iteration. For the rest of the iteration steps, the RHS term is the residual in fulfillment of the governing equation. Hence, no modification to the RHS is employed,

$$E = \begin{cases} (\mathcal{E} - I)B + I & \nu = 0 \\ I & \nu > 0, \end{cases} \tag{23}$$

where I is the $N_F \times N_F$ identity matrix and ν is the iteration level.

The multiscale approximate solution expressed in Eq. (4) is stated algebraically as

$$p' = \mathcal{P}p'_V + p^{corr}. \tag{24}$$

Once the coarse-scale pressure p'_V is obtained, Eq. (15) provides the edge and interior pressures. To compute p'_V, the following coarse-scale system is constructed and solved

$$A_C p'_V = R_C. \tag{25}$$

Here,

$$A_C = \mathcal{R}A\mathcal{P}, \tag{26}$$

and

$$R_C = \mathcal{R}q - \mathcal{R}Ap^{corr}. \tag{27}$$

The coarse-scale system of Eq. (25) is an algebraic description of Eq. (5). The restriction operator $\mathcal{R}$ is $N_C \times N_F$, and can be based on finite-volume, or finite-element, schemes. For the finite-volume operator, the fine-scale equations in a coarse cell are simply summed up. Therefore, the entries of the MSFV restriction operator are

$$\mathcal{R}(i,j) = \begin{cases} 1 & \text{if } \Omega_j^F \subset \Omega_i^C \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in \{1,\ldots,N_C\}; \forall j \in \{1,\ldots,N_F\}. \tag{28}$$

The condition $\Omega_j^F \subset \Omega_i^C$ is true if the fine cell j (Ω_j^F) belongs to the coarse control volume i (Ω_i^C). The finite-element based restriction operator (MSFE) is the transpose of the prolongation operator, i.e.,

$$\mathcal{R} = \mathcal{P}^T. \tag{29}$$

With the prolongation and restriction operators defined, one can solve the coarse-scale system (MSFV, or MSFE) for p'_V. Then, Eq. (24) is used to prolong the coarse-scale solution back to the fine scale, i.e.,

$$p \approx p' = \left[\mathcal{P}(\mathcal{R}A\mathcal{P})^{-1}\mathcal{R}(I - A\mathcal{C}) + \mathcal{C}\right]q. \tag{30}$$

Finally, one can define the multiscale (MS) preconditioner with CF (which is referred to as MSWC) as

$$M_{mswc}^{-1} = \mathcal{P}(\mathcal{R}A\mathcal{P})^{-1}\mathcal{R}(I - A\mathcal{C}) + \mathcal{C}. \tag{31}$$

This iterative procedure in combination with a fine-scale smoother, e.g., line relaxation [27], or GMRES preconditioner [28] was reported in the literature, where CF played a major role in capturing the fine-scale RHS terms and the residual. However, no detailed study of the computational efficiency of this procedure using different restriction operators with different local boundary conditions and smoothers has been reported. Also, the exact role of CF is unclear. These issues are discussed in the following sections.

Next, we investigate the exact role of the CF in this iterative procedure.

3. Analysis of the correction function

3.1. Independent local stage

After some mathematical manipulation, Eq. (31) can be rewritten as

$$\begin{aligned} M_{mswc}^{-1} &= \mathcal{P}(\mathcal{R}A\mathcal{P})^{-1}\mathcal{R} + \mathcal{C} - \mathcal{P}(\mathcal{R}A\mathcal{P})^{-1}\mathcal{R}A\mathcal{C} \\ &= M_{ms}^{-1} + \mathcal{C} - M_{ms}^{-1}A\mathcal{C}. \end{aligned} \tag{32}$$

In other words, the iterative procedure

$$p^{\nu+1} = p^{\nu} + M_{mswc}^{-1}\left(q - Ap^{\nu}\right) \tag{33}$$

is equivalent to the following two-stage iterative procedure

$$p^{\nu+1/2} = p^{\nu} + \mathcal{C}\left(q - Ap^{\nu}\right) \tag{34}$$

$$p^{\nu+1} = p^{\nu+1/2} + M_{ms}^{-1}\left(q - Ap^{\nu+1/2}\right). \tag{35}$$

The two steps are (1) update the solution with the CF operator; (2) update with the multiscale preconditioner $M_{ms}^{-1} = \mathcal{P}(\mathcal{R}A\mathcal{P})^{-1}\mathcal{R}$, which does not involve CF. Therefore, the operator $\mathcal{C}$ is a totally independent stage that does not affect the MS preconditioner at all. This helps to quantify the impact of CF on the iterative multiscale solution strategy. We show that CF is similar to other standard (local) block preconditioners aimed at high-frequency errors.

A heterogeneous case with 100×100 fine and 10×10 coarse cells is considered. The log-normally distributed permeability field with a spherical variogram and dimensionless correlation lengths of $\psi_1 = 0.5$ and $\psi_2 = 0.02$ is used. Also, the variance and mean of $\ln(k)$ are 2 and 3, respectively. As depicted in Fig. 3, the angle between the long correlation length and the vertical domain boundaries is 45°. The pressure at $(1,1)$ and $(100,100)$ is fixed with the values of 1 and 0, respectively. In this case, gravity acts in the y direction with a constant value of $\rho g = 1$. The BILU block size is the same as the size of the dual coarse cells in order to provide the same support as the CF. Fig. 4 shows that both the CF and BILU

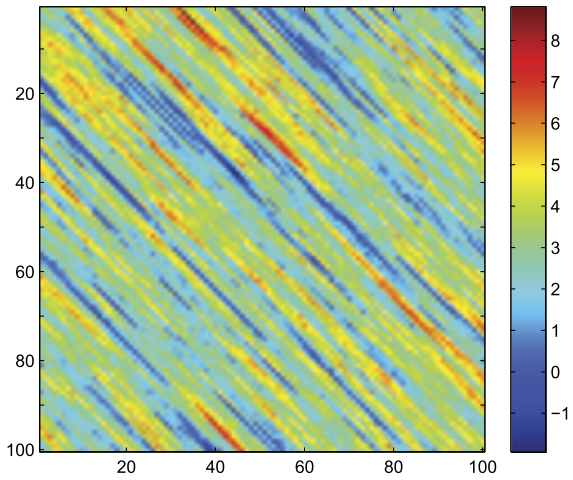


Fig. 3. Natural logarithm of layered permeability field with $\psi_1 = 0.5$ and $\psi_2 = 0.02$. Fine grid size is 100×100 and coarse grid size is 10×10.

improve the original MSFV solution significantly. Also, the MSFV-CF solution is slightly better than the MSFV-BILU solution. Denote p as the pressure solution and p_{fine} as the fine-scale reference, then the pressure solution error norms, defined as $\|p - p_{fine}\|_2 / \|p_{fine}\|_2$, for MSFV, MSFV-CF, and MSFV-BILU are 5.13, 0.16, and 0.28, respectively. The main difference between CF and BILU is the local boundary condition. In this case, the reduced boundary condition captures the gravity effects quite well, if one employs MSFV (with no iterations). Note that as long as MSFV is employed as the final step, local mass conservation on the primal coarse grid is guaranteed regardless of which local preconditioner is used. The choice of the local stage preconditioner is a trade-off between accuracy and computational effort and is investigated in detail in Section 4.

3.2. Spectral analysis: CF is not a convergent local solver

The multiscale stage on its own (without CF or other smoothers) cannot converge because $\text{rank}(M_{ms}^{-1}) \leqslant N_C$, which implies that $\text{rank}(M_{ms}^{-1} A) \leqslant N_C$; i.e., the coarse-scale system cannot span the fine-scale space [35]. In fact, the multiscale stage resolves only low frequency errors. Therefore, local stages (smoothers) are required to remove the high frequency errors. If only CF is considered as the local solver, the two-stage iterative procedure (MSFV-CF) is not convergent. In fact, as shown in [27], MSFV with CF (MSFV-CF) requires another local stage (smoother) to become convergent.

To illustrate the eigenvalue structure of MSFV alone and in combination with local stages, such as the CF and BILU, homogeneous and heterogeneous 2D test cases are considered. The fine and coarse grids are 40×40 and 4×4, respectively. The BILU block size is the same as the size of the dual coarse cells (almost the same support as CF). The pressure is fixed at $(1, 1)$ and $(40, 40)$. For the heterogeneous case, a log-normally distributed permeability field with a spherical variogram (using sequential Gaussian simulations [36]) is generated. The variance and mean are both 4 and the correlation lengths are $1/8$ of domain size in each direction (Fig. 5). For the homogeneous case, the permeability is unity.

Figs. 6 and 7 show that using a multiscale strategy only does not guarantee convergence; however, when it is combined with a local preconditioner such as BILU, all the eigenvalues are inside the unit circle. If the CF is used instead of BILU, i.e., MSFV-CF, some eigenvalues are larger than unity. This is because CF shares the same reduced boundary condition assumption as the basis functions in the MS system. Hence, local errors on dual-coarse block boundaries cannot be removed by the CF. To obtain a convergent iterative scheme using CF, other local preconditioning stages (or GMRES) are required. If BILU is used as an additional local stage, the three-stage AMS (i.e., MSFV-CF-BILU) is convergent. Also, note that the maximum eigenvalue of this three-stage AMS is smaller than that for the two-stage case of MSFV-BILU. Therefore, the three-stage MSFV-CF-BILU converges faster, but each iteration is computationally more expensive.

3.3. Sensitivity to transmissibility contrasts

Another drawback of the CF is that it is very sensitive to large contrasts in the transmissibility field. This is because CF is the solution of lower dimensional problems on the edges of dual-coarse cells with source terms. For example, if non-zero source terms exist on the edge cells between two impermeable regions that cross the boundary, the reduced problem is not solvable. This issue can slow down the convergence rate, and even lead to divergence. To resolve these difficulties, we scale RHS terms of edge cells (face in 3D) using a local factor, i.e.

$$\begin{bmatrix} q_I^m \\ q_E^m \\ q_V^m \end{bmatrix} = E' \begin{bmatrix} q_I \\ q_E \\ q_V \end{bmatrix}, \tag{36}$$

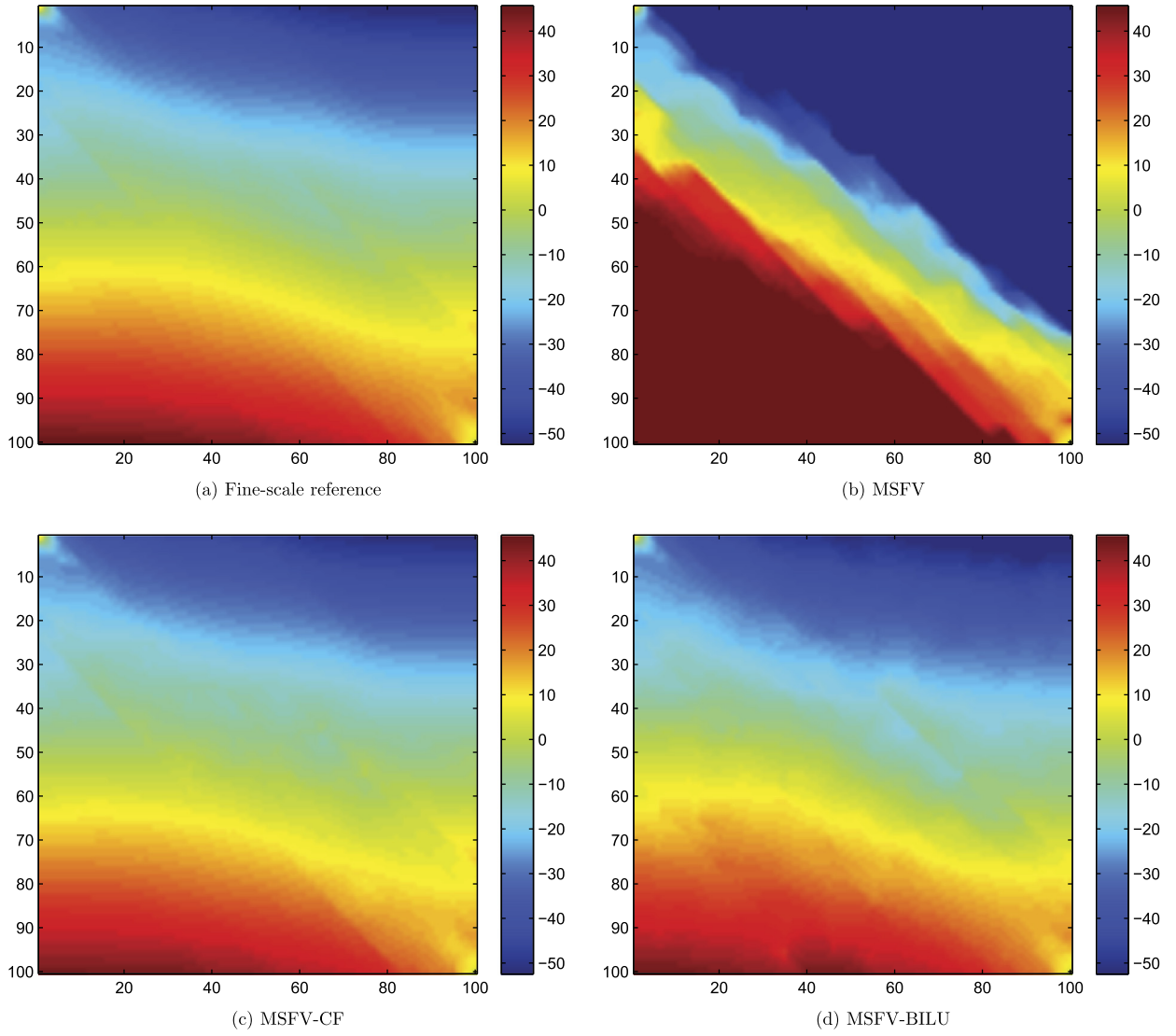


Fig. 4. Comparison between the solutions obtained from fine-scale reference, MSFV, MSFV-CF and MSFV-BILU. Note that all solutions are conservative at the coarse-scale. Furthermore, error norms for MSFV (b), MSFV-CF (c), and MSFV-BILU (d) are 5.13, 0.16 and 0.28, respectively.

where q^m is the modified RHS vector and E' is a diagonal matrix with

$$E'_{ii} = \begin{cases} \frac{T_{e,\min}}{T_{e,\max}} E_{ii} & \text{if } i \in \aleph_{edge} \\ E_{ii} & \text{otherwise} \end{cases} \tag{37}$$

entries. Here, $T_{e,\min}$ and $T_{e,\max}$ are the local minimum and maximum values of the transmissibility at the interfaces along the edge (in 3D, along a face) e. This approach is purely local, and E' can be calculated automatically based on the fine-scale transmissibility field and grid information. The correction function computed with the modified RHS vector q^m is referred to as a modified correction function (MCF), and the effectiveness of MCF is shown in the following sections.

3.4. Computational cost

As shown in the previous subsections, in many settings, CF reduces the number of iterations required to converge. It is important to note that this is beneficial if the gains are worth the additional computational cost of CF. In order to examine the computational cost of the CF stage (original and modified CF), a 3D, log-normally distributed permeability field [36] with 20 realizations is considered (see Fig. 8). The mean and variance of $ln(k)$ are -1 and 4, respectively. The correlation length is 1/8 of the domain size in each direction. The fine-scale and coarse-scale grids consist of $128 \times 128 \times 64$ and $16 \times 16 \times 8$ cells, respectively. Each BILU block contains $4 \times 4 \times 4$ fine cells. The pressure is fixed at the left and right faces with the

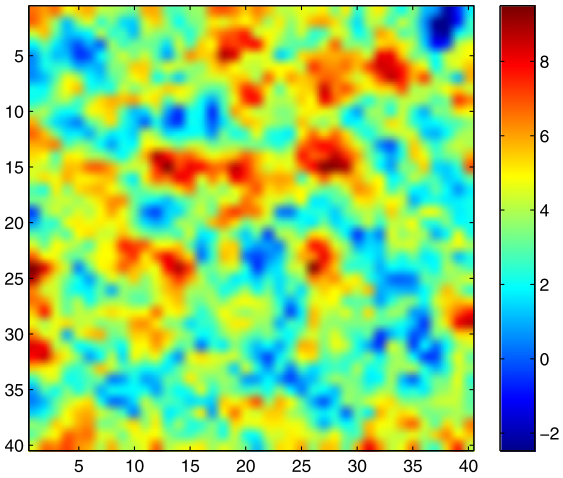


Fig. 5. Natural logarithm of permeability field for spectral analysis.

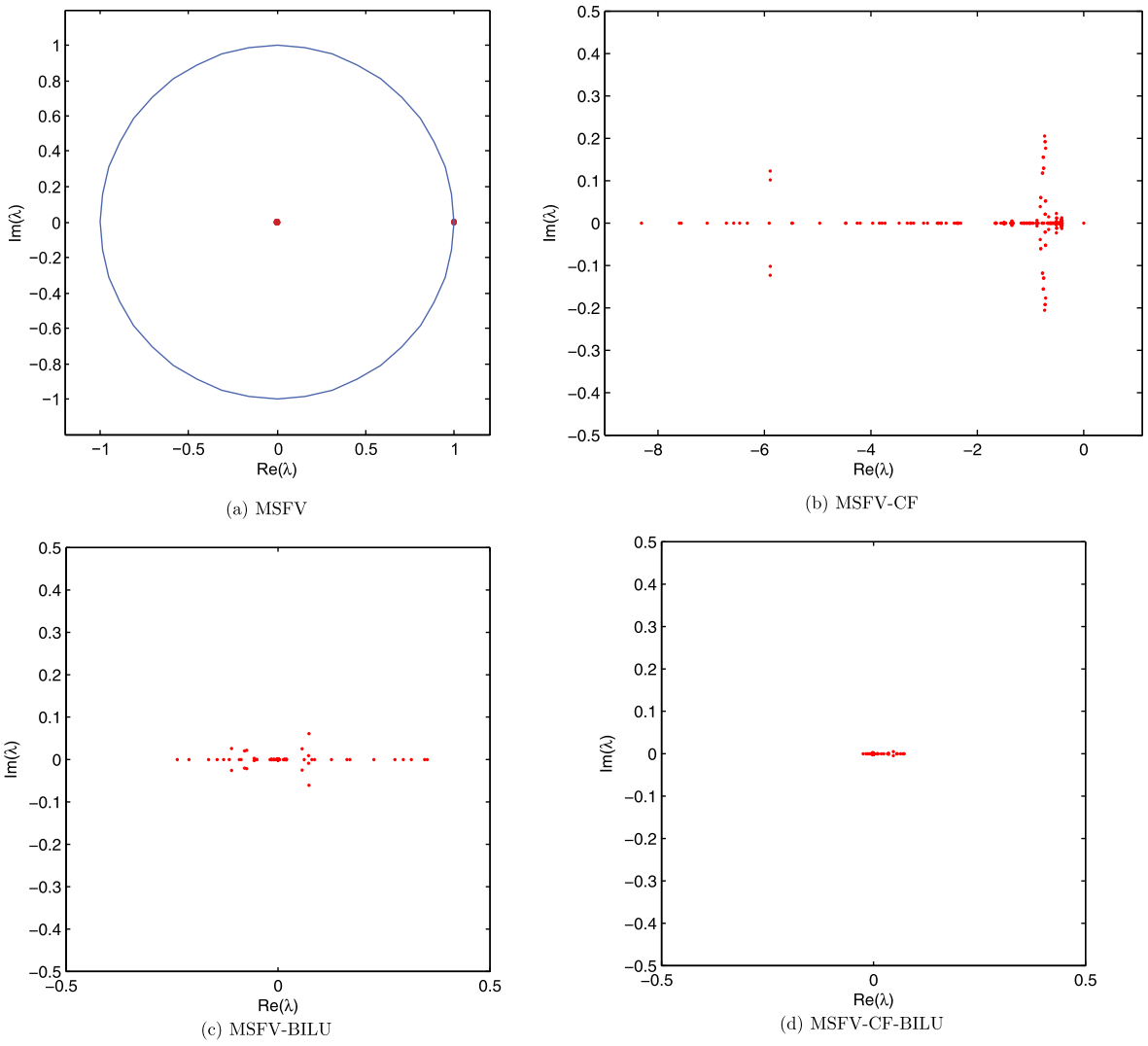


Fig. 6. Eigenvalues of MSFV, MSFV-CF, MSFV-BILU and MSFV-CF-BILU iteration matrices for a simple homogeneous test case.

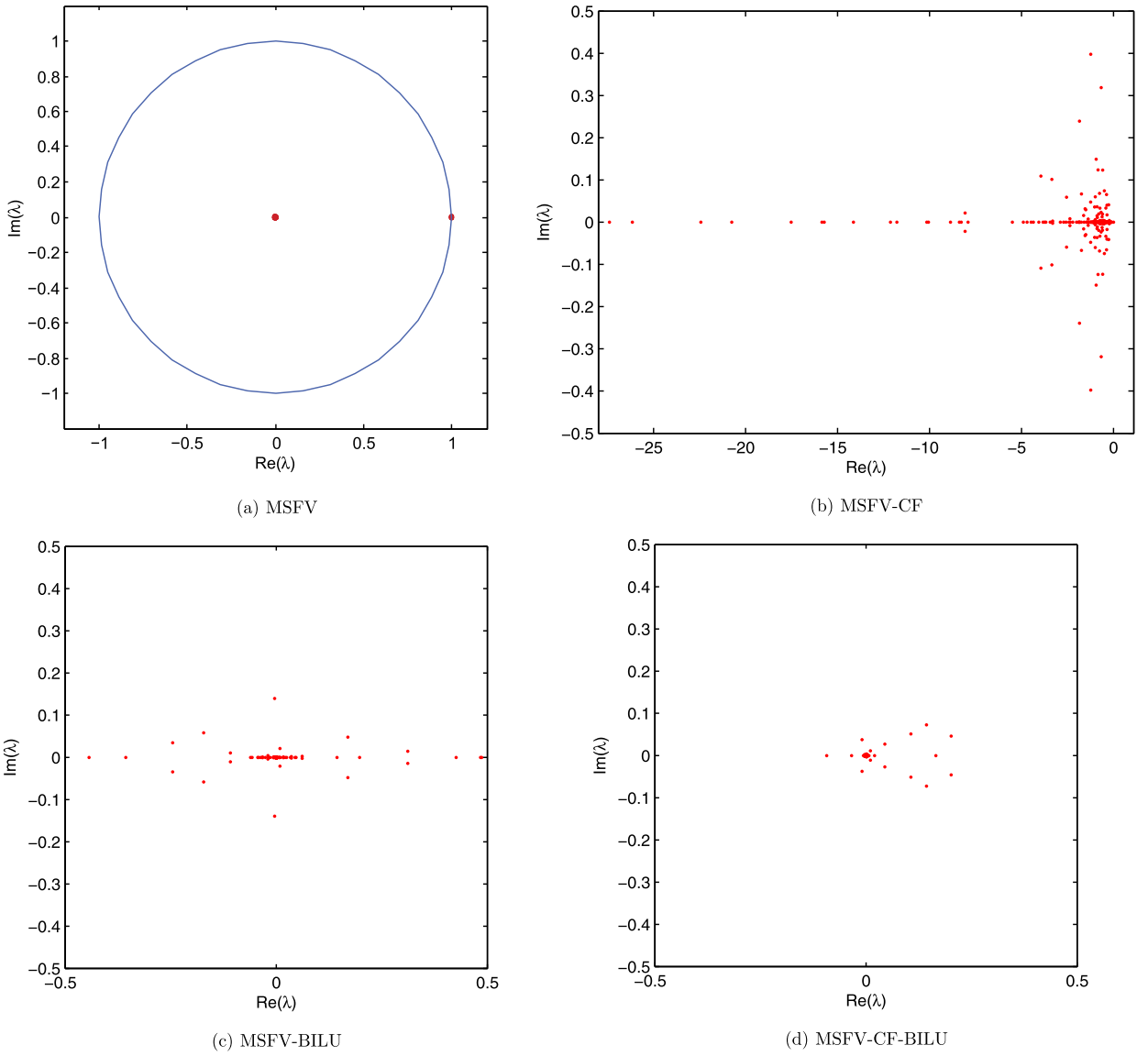


Fig. 7. Eigenvalues of MSFV, MSFV-CF, MSFV-BILU and MSFV-CF-BILU iteration matrices for a heterogeneous test case.

values of 1 and 0, respectively. The iterations are stopped once the reduction in the relative l^2 norm of the residual is ten orders of magnitude (i.e., $\|r_k\|_2/\|r_0\|_2 \leqslant 10^{-10}$). A simple Richardson iterative scheme is used. Table 1 indicates that the three-stage MSFE-CF-BILU with the original CF does not converge. As discussed before, this is due to the high contrasts at the local boundaries. On the other hand, the modification proposed in Eq. (36) for CF leads to a convergent iterative scheme (i.e., MSFE-MCF-BILU). Although the number of iterations is reduced by 19% when the MCF is used, the computational cost of the solution phase (measured in CPU time) is increased by 46%. Further discussion on the efficiency of CF for a wide range of cases is discussed later.

The study presented in this section shows that the CF is an independent local stage solver aimed at high frequency errors. Hence, it can be replaced by other local solvers. Also, CF helps the convergence rate, but it does not offset its additional computational cost (based on total CPU). Moreover, the CF cannot be used as the sole local-stage solver. Additional local or global solvers, e.g., smoothers or GMRES, are required to make the iterative procedure convergent. Furthermore, the CF is sensitive to transmissibility contrasts along edge cells.

Next, systematic numerical tests are provided to find the most effective combination of stages to solve the pressure equation. Also, the research code for AMS is tested against a production-quality SAMG solver.

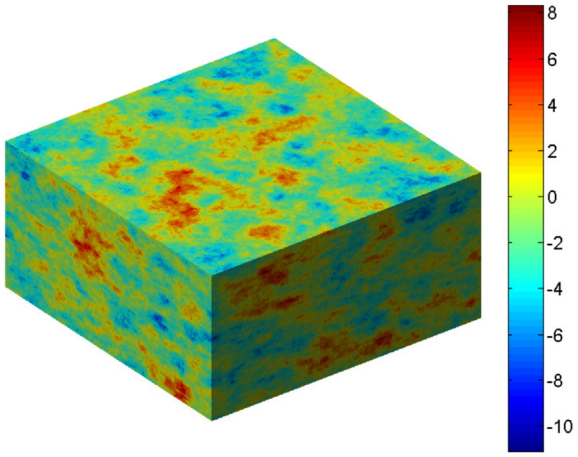


Fig. 8. Natural logarithm of the permeability field for one of the 20 statistically-the-same fields generated to analyze the computational efficiency of CF. The domain consists of $128 \times 128 \times 64$ fine and $16 \times 16 \times 8$ coarse grid cells.

Table 1
Average total simulation time (sec) and number of iteration steps for MSFE-BILU, MSFE-CF-BILU, and MSFE-MCF-BILU AMS solvers. The three stage MSFE-CF-BILU is not convergent due to the CF sensitivity to the contrast. However, if the modified CF (MCF) is used (Eq. (36)), i.e. MSFE-MCF-BILU, the procedure becomes convergent. Results are shown on average for twenty statistically-the-same realizations. Also shown in parentheses are the standard deviations.

	MSFE-BILU	MSFE-CF-BILU	MSFE-MCF-BILU
Iteration steps	45.8 (±3.4)	–	37.0 (±4.4)
Solve time (sec)	20.0 (±1.4)	–	29.0 (±3.4)

Table 2
Five permeability sets (each with 20 equiprobable realizations) are used for the numerical experiments of this section. Layered fields, i.e. sets 1 and 2, are generated for 4 different layering angles, each of which has 20 equiprobable realizations.

Permeability set	1	2	3	4	5
Fine-scale grid	128^3	64^3	128^3	64^3	32^3
ψ_x	0.5	0.5	0.125	0.125	0.125
ψ_y	0.03	0.03	0.125	0.125	0.125
ψ_z	0.06	0.01	0.125	0.125	0.125
Angle between ψ_x and y direction	0°, 15°, 30°, 45°			patchy	
Variance			4		
Mean			−1		

4. Numerical results

In this section, systematic tests are performed to find the best combination of local and global stages. For the following experiments, five sets of log-normally distributed permeability fields with spherical variograms are generated using sequential Gaussian simulations [36]. For all the test cases, the variance and mean of $\ln(k)$ are 4 and -1, respectively. The fine-scale grid size and dimensionless correlation lengths in the x, y, z direction, i.e., ψ_x, ψ_y and ψ_z are shown in Table 2. Each set has 20 equiprobable realizations. For sets 1 and 2, 20 realizations with different orientation angles (Fig. 9) of 0, 15, 30, and 45 degrees are considered. For sets 3, 4 and 5, 20 realizations of patchy domains are used (Fig. 10). In the following experiments, GMRES preconditioned by the AMS is employed as the iterative procedure. The pressure is fixed on the left and right faces with dimensionless values of 1 and 0, respectively. The iterative procedures are performed until the reduction in the relative l^2 norm of the residual is five orders of magnitude (i.e., $\|r_k\|_2/\|r_0\|_2 \leqslant 10^{-5}$).

In the next 3 sections, the coarse-scale restriction, local boundary conditions, and second stage local solver are studied. Finally, in the last section, the AMS efficiency as a linear solver is studied versus SAMG.

4.1. AMS global stage: MSFV versus MSFE

The performance of the two different restriction schemes, namely MSFV and MSFE, is investigated. The permeability of the SPE 10 bottom layer, which has channelized structures is considered (Fig. 11). This permeability is selected because it is very challenging. The fine and coarse grids consist of 220×60 and 22×6, respectively. Each BILU block contains 10×10 fine cells. Also, both FV and FE restriction operators are employed – with and without CF. The pressure is fixed at the left

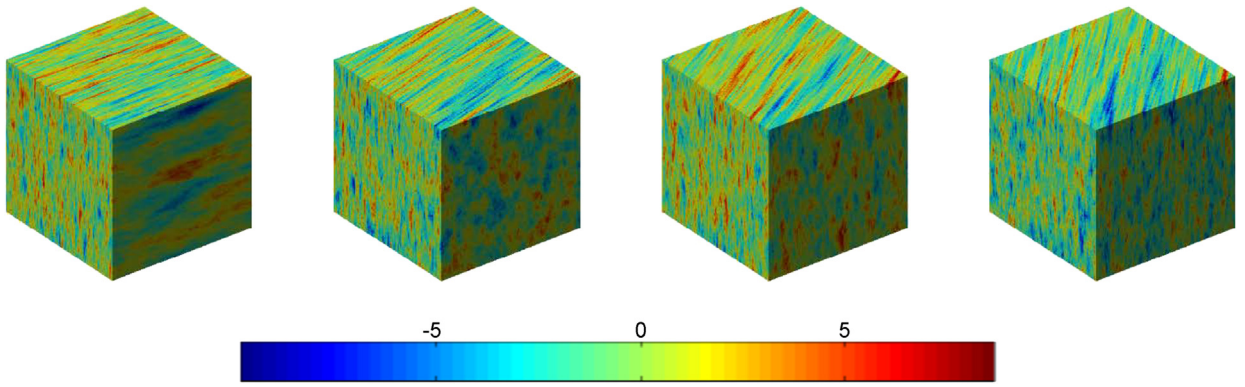


Fig. 9. Natural logarithm of one realization of permeability set 1 with different layering angles of 0°, 15°, 30° and 45° from left to right. For each layering angle, 20 realizations are considered.

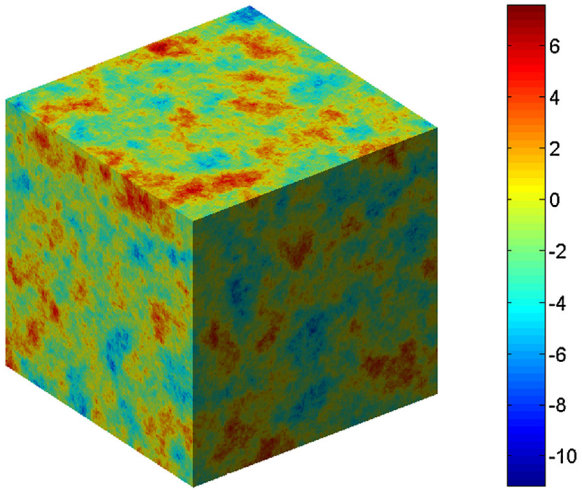


Fig. 10. Natural logarithm of one (out of 20 statistically-the-same) realization of the permeability set 3.

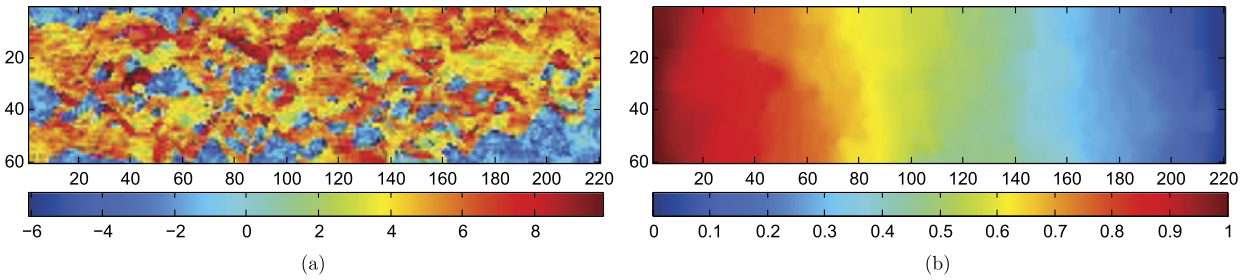


Fig. 11. Natural logarithm of the permeability (a) and fine-scale pressure solution (b) for the SPE 10 bottom layer.

and right sides with the values of 1 and 0, respectively. Fig. 12 shows that if MSFV is used as the global stage solver, AMS does not converge. On the other hand, if MSFE is used, AMS converges efficiently. No GMRES was used to stabilize the iterations. Hence, the inefficient iterations associated with the previously developed i-MSFV [27,30] were mainly due to the weak coarse-scale MSFV operator. The CF is also considered, and it is shown that it does not overcome the difficulties due to the weak coarse-scale operator.

To study the performance of AMS with different global stage solvers, one should consider the total CPU time, not just the iteration numbers. For this reason, permeability sets 1 and 3 are considered. The fine and coarse grids contain $128 \times 128 \times 128$ and $16 \times 16 \times 16$ cells, respectively. ILU is employed as the local preconditioner, and no CF is used. As Fig. 13 shows, the FE coarse-scale operator outperforms the FV one for both permeability sets.

Anisotropic cases are also considered by setting the factor α in $\Delta x = 2\alpha\Delta y = \alpha\Delta z$ relation. One realization from each rotation angle of permeability set 2 and one realization from permeability set 4 are chosen. Both ILU and BILU are employed

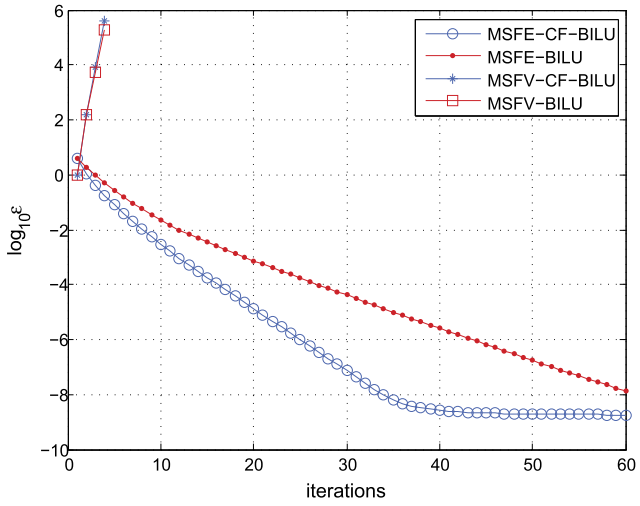


Fig. 12. Iteration histories for AMS with different restriction schemes using the SPE 10 bottom layer.

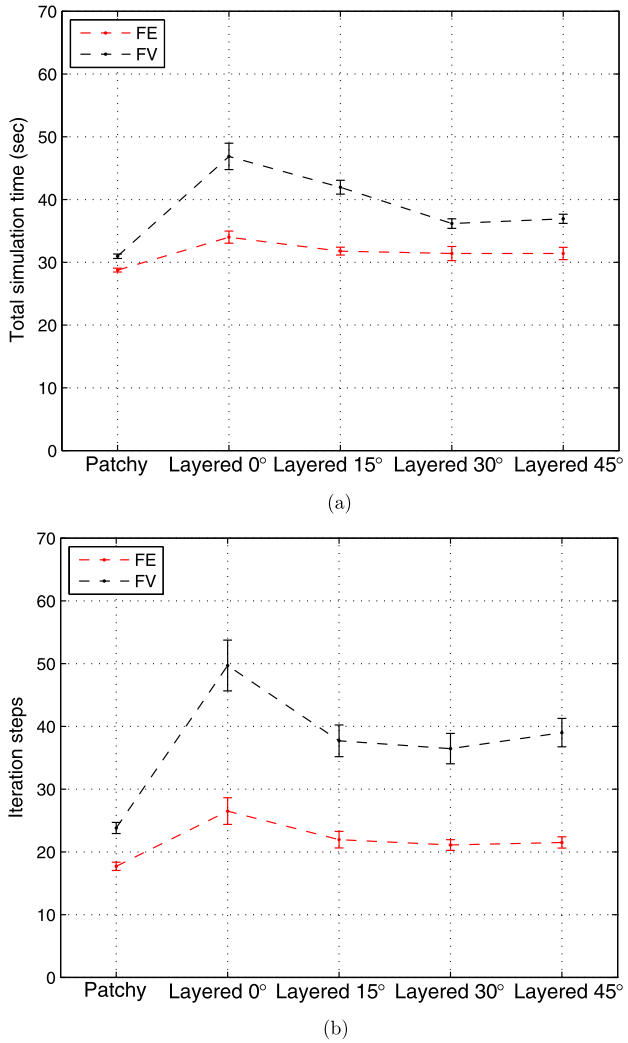
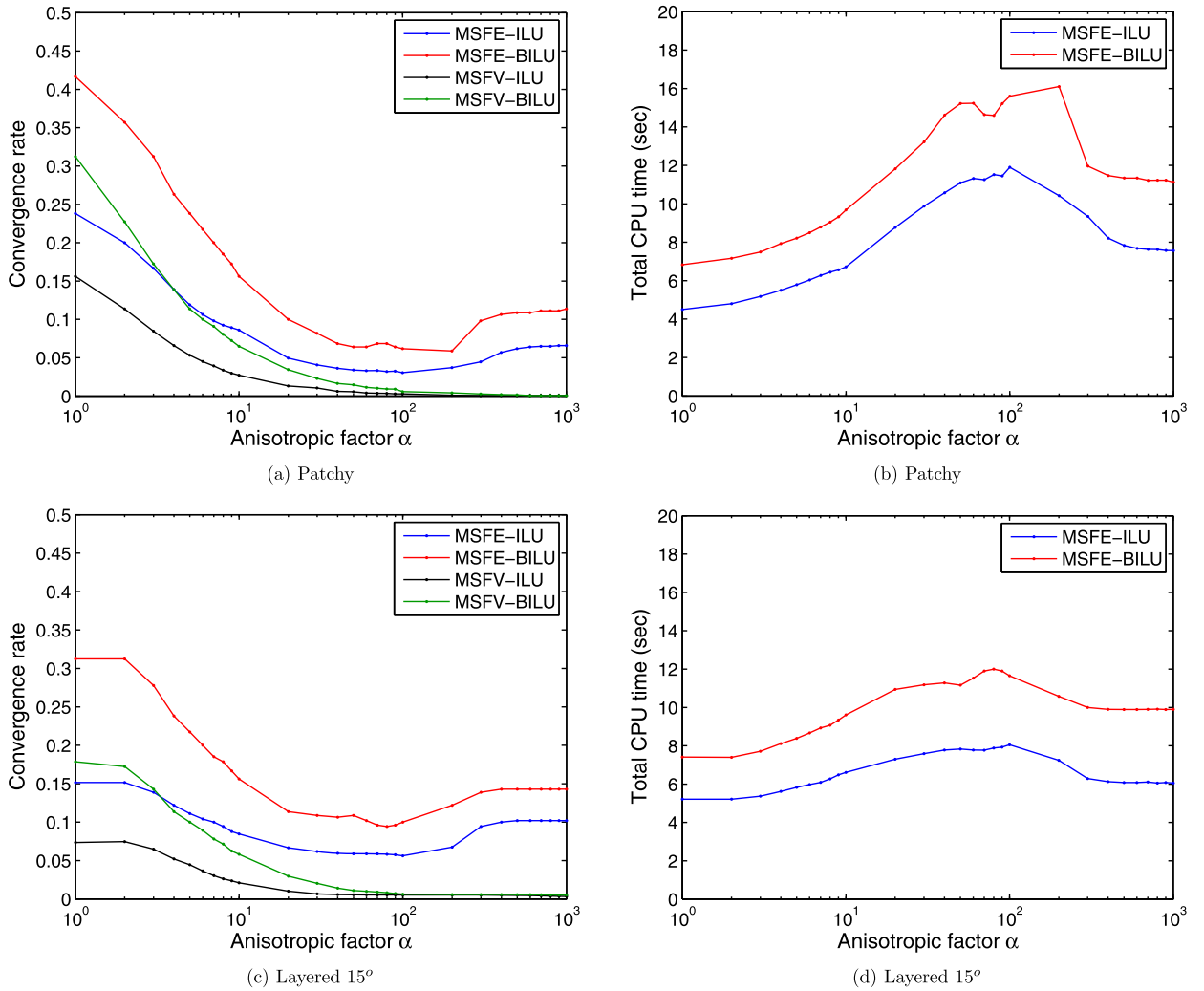


Fig. 13. Comparison of (a) total simulation time and (b) iteration steps for FE and FV global solvers (i.e., restriction operator) on layered and patchy permeability fields over 20 different realizations. Also shown in error bars are the standard deviations. Clearly, FE restriction operator outperforms the FV one.



(a) Patchy (b) Patchy

(c) Layered 15^o (d) Layered 15^o

Fig. 14. Comparison of MSFV and MSFE restriction operators for anisotropic patchy and layered permeability fields (one realization for each) with different local solvers (BILU and ILU). The convergence rate and total simulation time (sec) are illustrated on the left and right columns, respectively. For layered systems with orientation angles of 0°, 30° and 45°. The results are similar to the layered 15° case; therefore, they are not shown here.

as local preconditioners, and the results are compared. The coarse grid is $8 \times 8 \times 8$ and each BILU block contains $4 \times 4 \times 4$ fine cells. The convergence rate is defined as the inverse of the number of iterations required for reducing the error by five orders of magnitude. As shown in Fig. 14, the convergence rate of MSFV decreases monotonically as α increases. However, with the MSFE coarse-scale operator, the convergence rate stagnates once α is greater than a certain value. The main message here is that MSFE clearly outperforms MSFV for highly heterogeneous anisotropic problems. Note that although BILU has better convergence rates compared with ILU, it has a more expensive setup phase and entails more operations per iteration. Our observation with using the linear relation (LR) also showed that blocks of the size $b \times 1$ (corresponding to LR lines) have higher setup time and more expensive operations per iteration compared with ILU.

For the above mentioned anisotropic test cases, the coarsening factor is chosen as $8 \times 8 \times 8$ because this coarse-grid size leads to the most computationally efficient performance for both FE and FV restriction schemes. For permeability sets 2 and 4, four different coarse-grid sizes are generated: $16 \times 16 \times 16$, $8 \times 8 \times 8$, $4 \times 4 \times 4$ and $2 \times 2 \times 2$. Also, ILU is employed as the local preconditioner. As shown in Fig. 15, considering the total CPU time, the trade-off between the convergence rate and computational cost is achieved at a coarsening factor of eight in each direction. Generally, it is a good choice to chose the coarsening factor in each direction nearly the square root of the number of cells in that direction.

4.2. AMS global stage: local boundary conditions

The effects of different local boundary conditions (BC), i.e. reduced BC and linear BC, are studied here. Permeability sets 2 and 4 are considered. The coarse grid size is $8 \times 8 \times 8$, and ILU is employed as the local preconditioner. Fig. 16 shows that for patchy domains, the reduced and linear BC have similar performance. For layered permeability fields, however, the linear

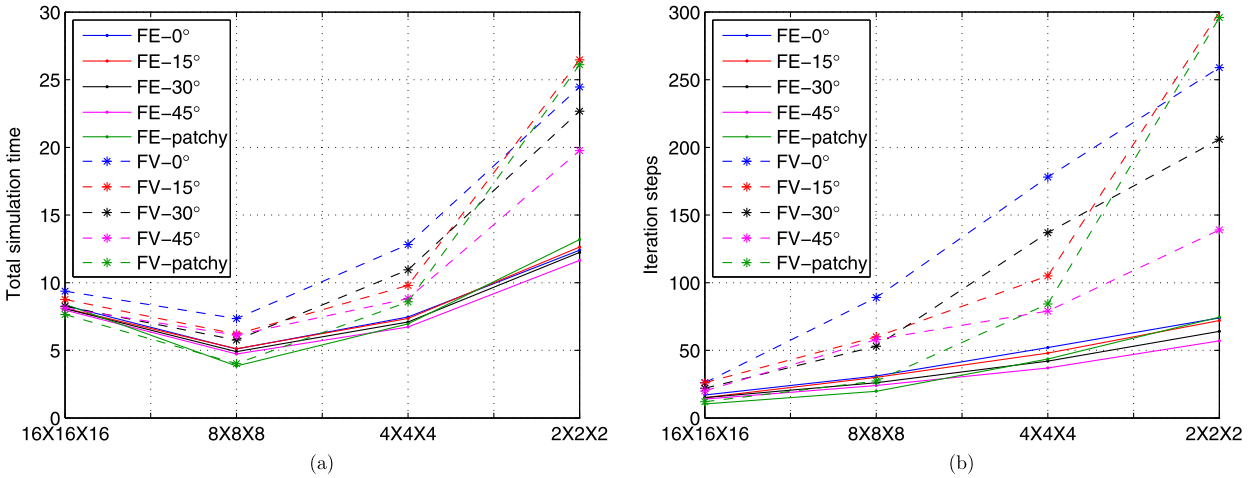


Fig. 15. Coarse grid size effect on average (a) total simulation time and (b) iteration steps for permeability set 2 and 4.

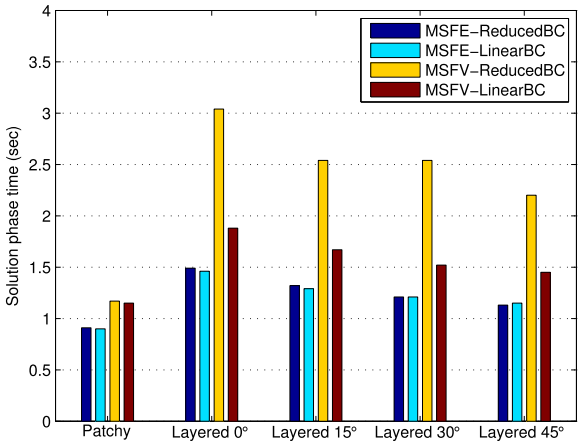


Fig. 16. Solution phase time (i.e. excluding setup time) averaged over 20 equiprobable realizations for MSFV and MSFE restriction schemes with linear and reduced boundary conditions.

BC improves the computational efficiency of MSFV. However, the improved MSFV with linear BC is still not competitive with MSFE. Clearly, the linear BC is advised for MSFV, while for MSFE the reduced problem is as efficient as the linear BC.

Next, we employ the full SPE 10 3D case (Fig. 17). The domain consists of $60 \times 220 \times 85$ fine and $6 \times 22 \times 17$ coarse grid cells. Also, each BILU block contains $4 \times 4 \times 5$ fine cells. The pressure is fixed at the left and right faces with the values of 1 and 0, respectively. In this case, AMS is used as a preconditioner for GMRES. The AMS performance with reduced and linear boundary conditions for both MSFV and MSFE is shown in Fig. 18. From this figure, it is clear that the MSFE with the reduced-problem boundary condition is more efficient that the other options. For MSFV, the linear boundary condition is better than the reduced boundary condition. In fact the MSFV-BILU-ReducedBC does not lead to a convergent iterative scheme for this challenging test case. Also, note that when the linear BC is used, MSFE and MSFV have comparable performance. However, MSFE with the reduced-problem BC outperforms the linear BC. To obtain an estimate about the efficiency of the AMS for this test case, Fig. 18 also shows the CPU time for MSFE-BILU and MSFE-ILU iterative procedures with the reduced boundary conditions, from which it is also clear that MSFE-ILU is more efficient.

4.3. AMS local stage

Overall, MSFE with the reduced boundary condition is found to be the most efficient global-stage solver. Next, we investigate which local stage preconditioner is the best overall choice. BILU is used as the second stage preconditioner in TAMS [35]. Here, ILU is employed as the local preconditioner. Based on our experiments, the solution time of BILU and ILU are comparable; however, ILU has minimal setup time compared with BILU. Hence, overall, ILU outperforms BILU in terms of CPU time. A comparison between ILU and BILU is performed for permeability sets 1 and 3. The coarse grid and BILU blocks are $16 \times 16 \times 16$ and $4 \times 4 \times 4$, respectively. Fig. 19 shows the although ILU employs many iterations to converge, its total CPU time is much less than that of BILU for all the studied cases.

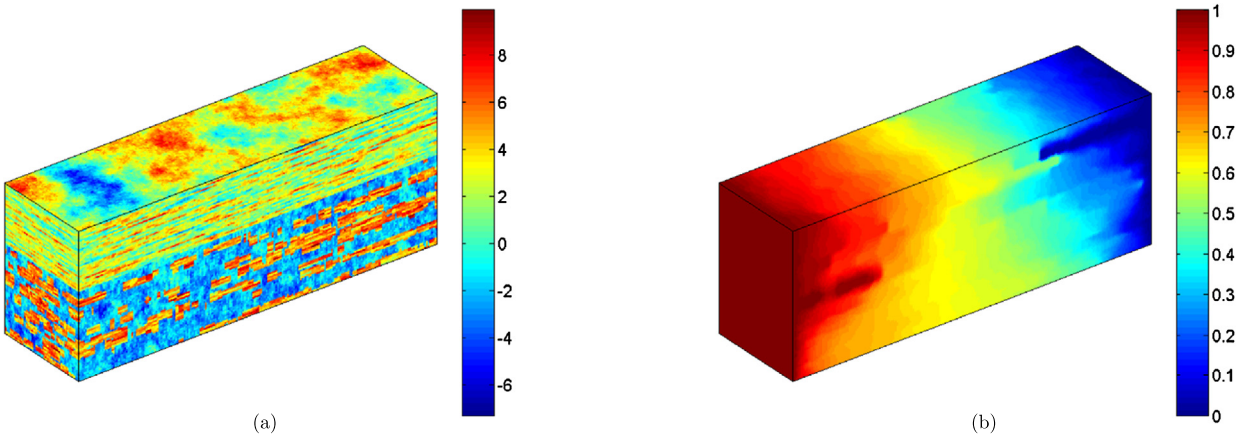
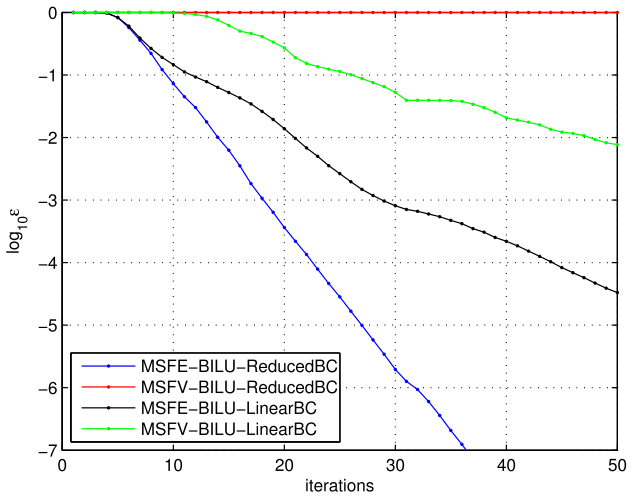


Fig. 17. (a) Natural logarithm of the permeability and (b) pressure solution for the full SPE 10 case. The grid contains $60 \times 220 \times 85$ fine and $6 \times 22 \times 17$ coarse cells.



AMS strategy	MSFE-BILU	MSFE-ILU
Setup phase time	26.91	13.24
Solution phase time	18.01	17.66
Total simulation time	44.92	30.90
Iteration steps	26	82

Fig. 18. Iteration histories for MSFV and MSFE with linear and reduced boundary conditions (left). Also shown on right is the CPU time (sec) and iteration steps for MSFE-BILU and MSFE-ILU preconditioned by GMRES. Iterations are stopped when the relative l^2 norm of the residual is reduced by five orders of magnitude.

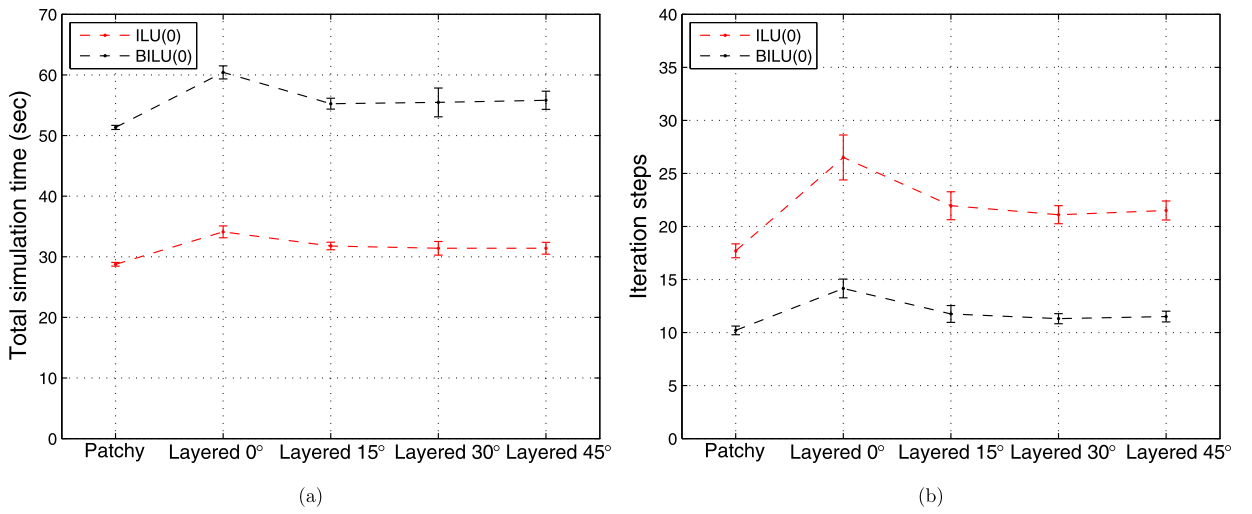


Fig. 19. The average and error bar plots of (a) total simulation time and (b) iteration steps for BILU and ILU comparison on layered and patchy permeability fields.

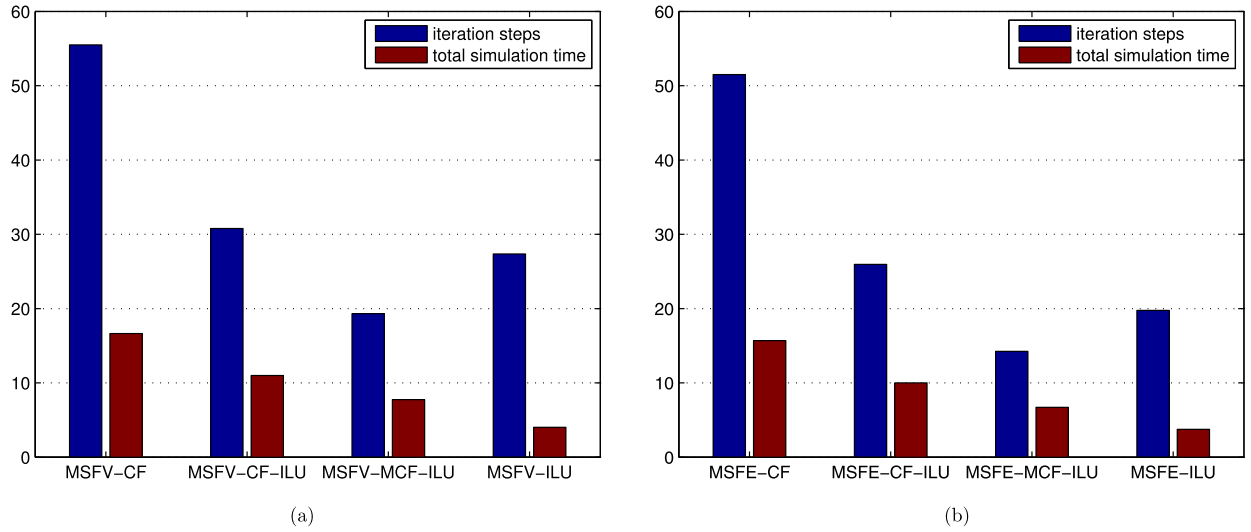


Fig. 20. Iteration steps and total simulation time (sec) for GMRES preconditioned by the MSFV (a) and MSFE (b) with CF, MCF, and ILU. Results are averaged over 20 realizations of patchy permeability field of set 4.

Finally, in order to compare the efficiency of the iterative procedure including CF and the proposed modified CF (MCF) with ILU, permeability set 4 is considered. The fine and coarse grids contain $64 \times 64 \times 64$ and $8 \times 8 \times 8$ cells, respectively. GMRES is also used in this case, so that MSFV-CF is convergent. Fig. 20 shows that MSFV-ILU outperforms the other cases where CF is used as the second, or third, stage solver. Due to the sensitivity of CF to high permeability contrasts, MSFV-CF consumes a lot of iterations. The iterations converge faster when ILU is used as an additional stage, and the efficiency will be further improved if the proposed modified CF (MCF) is used instead of the original CF, i.e. MSFV-MCF-ILU. Nevertheless, MSFV-ILU is still the most efficient combination. Similar results are obtained for MSFE. The conclusion is that ILU is more efficient than CF and MCF.

On the basis of what we presented above, it is found that MSFE with ILU as global and local stage solvers, respectively, lead to an efficient iterative multiscale solver. For the local stage CF, MCF, ILU, and BILU were studied in this paper. Line relaxation was also tested by setting the block sizes as lines. Among these choices, ILU was found to be the most efficient choice in terms of total CPU time. Of course, several other choices for the local stage could be considered and studied. However, our main message is that MSFE outperforms MSFV for both linear and reduced local BC for the test cases we studied here. Also, we found that the CF does not add significant improvements to the multiscale procedure.

Next, an optimum AMS procedure (on the basis of the presented study), i.e., GMRES preconditioned by the MSFE-ILU is tested against SAMG [1].

4.4. AMS vs. AMG

To investigate the efficiency of AMS compared with SAMG, which is widely used in the community, permeability set 5 (patchy field) is considered. As shown in Table 2, this problem set consists of 32^3 fine cells. To increase the size of the domain, but keeping the same permeability statistics, a refinement procedure is employed, such that each grid cell is divided into 8 cells in each refinement step (split into two in each direction). Employing this refinement procedure, four grid sets are generated with 32^3, 64^3, 128^3 and 256^3 fine cells. For all the problem sizes, the coarsening factor is kept constant with the value of $8 \times 8 \times 8$. Dirichlet boundary conditions are employed on the left and right faces with the values of 1 and 0, respectively. No-flow boundary condition is applied on all other faces. Iterations are performed until the relative l^2 norm of the residual is reduced by five orders of magnitude. The MSFE (with reduced boundary condition) and ILU are used as global and local solvers for AMS. The SAMG library is obtained from Fraunhofer Institute SCAI, release version 25a1 of December 2010 [1]. It employs a single stand-alone V-cycle with a convergence tolerance of 0.1 for the relative residual reduction and one Gauss–Seidel C-relaxation sweep as pre- and post-smoothing steps on each level. Also, the coarsest system is solved by a direct solver (Gauss elimination). The CPU time and number of iterations for SAMG and AMS are presented in Tables 3 and 4, respectively. It is clear that the two methods perform similarly for this test case. Also, Table 4 indicates that the AMS convergence rate is independent of problem size, i.e., similar to SAMG, the AMS is a scalable solver. This fact is further illustrated in Fig. 21(a), where the computational times for different problem sizes are normalized with that of the 32^3 case and plotted for setup and solution phases. Note that Fig. 21(b) shows SAMG is slightly above the ideal line for the setup phase, which reflects its complex coarsening strategy.

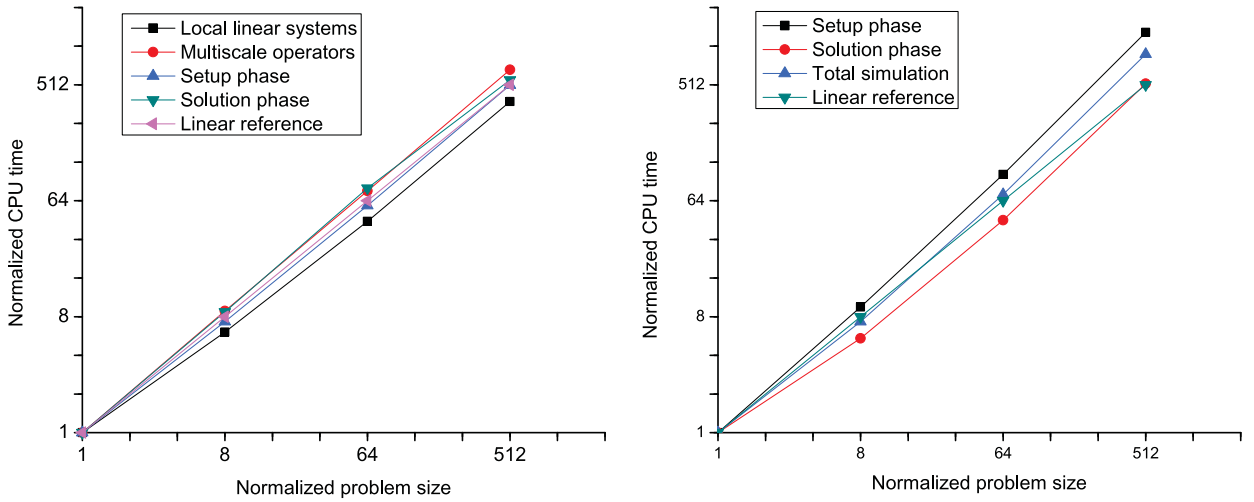
The performance of both AMS and SAMG is also tested and compared for permeability sets 1 and 3. The coarse grid consists of $16 \times 16 \times 16$ cells, and the same strategy for AMS, i.e., MSFE with reduced BC as global and ILU as local stages, is chosen. Note that the permeability set 1 is a layered field, for which the Cartesian coarse grid is still used in the AMS

Table 3
CPU time (sec) and iteration steps for SAMG on a patchy domain.

Problem size	$32 \times 32 \times 32$	$64 \times 64 \times 64$	$128 \times 128 \times 128$	$256 \times 256 \times 256$
Setup phase	0.12	1.14	12.33	157.41
Solution phase	0.14	0.76	6.33	73.02
Total	0.26	1.90	18.66	230.43
GMRES iterations	6	7	9	10

Table 4
CPU time (sec) and iteration steps for AMS on a patchy domain.

Problem size	$32 \times 32 \times 32$	$64 \times 64 \times 64$	$128 \times 128 \times 128$	$256 \times 256 \times 256$
Setup phase	0.29	2.12	17.00	147.96
Solution phase	0.10	0.87	8.01	55.91
Total	0.39	2.99	25.01	203.87
GMRES iterations	22	21	21	18



(a) AMS (b) SAMG

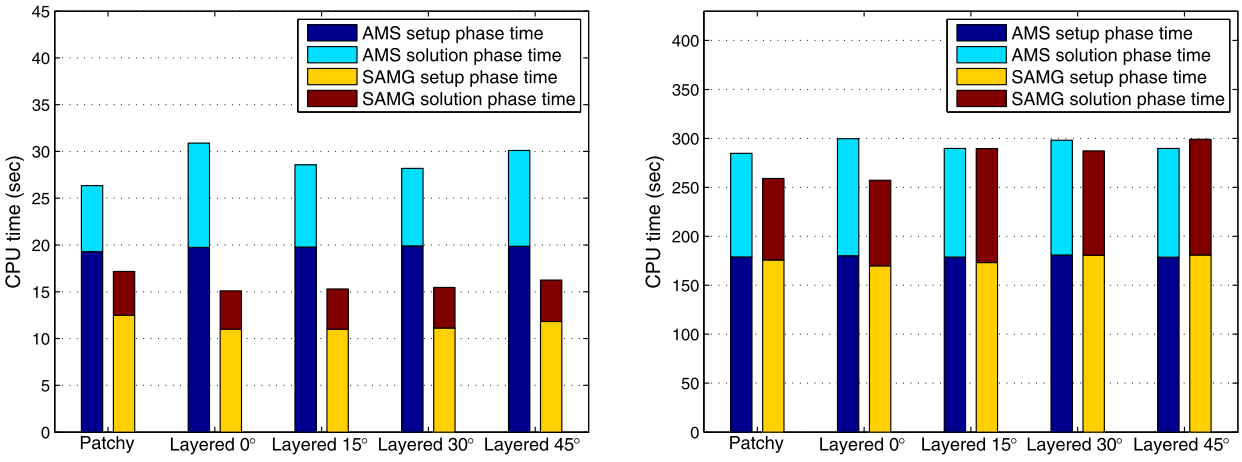
Fig. 21. Scalability analysis of AMS and SAMG.

coarse scale solver. It is clear from Fig. 22 that SAMG outperforms AMS. The difference between the two is more severe for the layered field, which demonstrates clearly the coarsening strategy of AMS needs to be improved. In general, having an AMS solver that is competitive with SAMG is important for the following reasons. First, it is clear that both AMS and SAMG have a considerable setup time. For time-dependent problems, AMS benefits from adaptive updating of the basis functions. Second, AMS is a mass conservative iterative solver when the MSFV operator is employed as the last step. This has been studied for i-MSFV [37,24]. A more general study of the adaptive AMS for multiphase nonlinear problems is a topic of ongoing investigation.

Note that the AMS used in this paper employs a structured Cartesian coarse grid, which is not efficient specially for layered permeability fields. Improving the coarse-grid geometry in order to account for the fine-scale transmissibility field is also a topic of ongoing research.

5. Conclusion

In this paper, a general Algebraic Multiscale Solver (AMS) for the pressure equation was developed. We analyzed the role of the Correction Function (CF) in the context of AMS, and we showed that the CF can be seen as an independent local stage aimed at high frequency errors. As a local preconditioner, CF helps to capture the fine-scale RHS (and residual) and accelerates the overall convergence rate. However, – on average – the gain in convergence rate does not compensate for the additional computational cost. Also, CF must be combined with other solvers (or smoothers) to guarantee convergence. Furthermore, the CF with the reduced boundary condition is sensitive to transmissibility contrasts. A modification to the CF is proposed and the improvement of the modification was studied numerically. In general, other preconditioners, such as ILU, are found to be more efficient than CF and MCF. Note that AMS with any combination of local and global stage solvers allows for reconstruction of a conservative velocity field, if an MSFV global stage is applied as the last step. The performance of AMS with many combinations of local and global solvers was systematically tested for several problems



(a) Fine-scale grid is $128 \times 128 \times 128$

(b) Fine-scale size is $256 \times 256 \times 256$

Fig. 22. Total, setup, and simulation times (sec) of AMS and SAMG as linear solvers for permeability sets 1 and 3. Results are averaged over 20 statistically-the-same realizations for each case.

(each with 20 different realizations). For several highly heterogeneous anisotropic problems, the MSFE restriction operator was found to be superior to the MSFV one. Overall, the best AMS is MSFE with reduced problem BC along with ILU. The performance of AMS is comparable to the state-of-the-art algebraic multigrid solver (SAMG) preconditioner. Our results show that AMS is very efficient, especially if it is used as a multiscale approximate (but conservative) solver in a nonlinear reservoir simulation context. Note that all studies presented in this work were done on single-processor machines. AMS is amenable for massive parallel computations of the setup phase, since basis functions are computed independently. For the local-stage solver, an efficient and robust solver for parallel computations is needed. ILU(0) is found to be efficient for our single-processing computations. However, it may not be an efficient solver for parallel computations. Detailed investigation of the proper components for local and global stages is the subject of future studies.

Acknowledgements

Financial Support from the Petroleum Institute (PI) and ADNOC is gratefully acknowledged. The authors thank Dr. Hui Zhou and Dr. Yifan Zhou for many helpful discussions.

References

[1] K. Stuben, SAMG User's Manual, Fraunhofer Institute SCAI, 2010.
[2] T. Hou, X.H. Wu, A multiscale finite element method for elliptic problems incomposite materials and porous media, J. Comput. Phys. 134 (1997) 169–189.
[3] Y. Efendiev, V. Ginting, T. Hou, R. Ewing, Convergence of a nonconforming multiscale finite element method, SIAM J. Numer. Anal. 37 (3) (2000) 888–910.
[4] J. Aarnes, T.Y. Hou, Multiscale domain decomposition methods for elliptic problems with high aspect ratios, Acta Math. Appl. 18 (1) (2002) 63–76.
[5] T. Arbogast, S.L. Bryant, A two-scale numerical subgrid technique for waterflood simulations, SPE J. 7 (2002) 446–457.
[6] Z. Chen, T. Hou, A mixed finite element method for elliptic problems with rapidly oscillating coefficients, Math. Comput. 72 (2003) 541–576.
[7] P. Jenny, S.H. Lee, H.A. Tchelepi, Multi-scale finite-volume method for elliptic problems in subsurface flow simulation, J. Comput. Phys. 187 (2003) 47–67.
[8] J.E. Aarnes, On the use of a mixed multiscale finite element method for greater flexibility and increased speed or improved accuracy in reservoir simulation, Multiscale Model. Simul. 2 (3) (2004) 421–439.
[9] J.E. Aarnes, V. Kippe, K.A. Lie, Mixed multiscale finite elements and streamline methods for reservoir simulation of large geomodels, Adv. Water Resour. 28 (3) (2005) 257–271.
[10] Y. Efendiev, T.Y. Hou, Multiscale Finite Element Methods: Theory and Applications, Springer, 2009.
[11] T. Arbogast, Implementation of a locally conservative numerical subgrid upscaling scheme for two-phase Darcy flow, Comput. Geosci. 6 (2002) 453–481.
[12] H. Hajibeygi, P. Jenny, Multiscale finite-volume method for parabolic problems arising from compressible multiphase flow in porous media, J. Comput. Phys. 228 (2009) 5129–5147.
[13] H. Zhou, H.A. Tchelepi, Operator-based multi-scale method for compressible flow, SPE J. 13 (2008) 267–273.
[14] I. Lunati, P. Jenny, Multiscale finite-volume method for density-driven flow in porous media, Comput. Geosci. 12 (3) (2008) 337–350.
[15] C. Wolfsteiner, S.H. Lee, H.A. Tchelepi, Well modeling in the multiscale finite volume method for subsurface flow simulation, SIAM Multiscale Model. Simul. 5 (3) (2006) 900–917.
[16] P. Jenny, I. Lunati, Modeling complex wells with the multi-scale finite volume method, J. Comput. Phys. 228 (2009) 687–702.
[17] H. Hajibeygi, R. Deb, P. Jenny, Multiscale finite volume method for non-conformal coarse grids arising from faulted porous media, in: SPE 142205-MS, 21–23 Feb., The Woodlands, Texas, 2011.
[18] H. Hajibeygi, D. Karvounis, P. Jenny, A hierarchical fracture model for the iterative multiscale finite volume method, J. Comput. Phys. 230 (24) (2011) 8729–8743.

[19] S.H. Lee, C. Wolfsteiner, H.A. Tchelepi, A multiscale finite-volume method for multiphase flow in porous media: Black oil formulation of compressible, three phase flow with gravity, Comput. Geosci. 12 (2008) 351–366.
[20] H. Hajibeygi, H.A. Tchelepi, Compositional multiscale finite volume formulation, SPE J. (2013), http://dx.doi.org/10.2118/163664-PA, SPE 163664-PA, in press.
[21] P. Jenny, S.H. Lee, H.A. Tchelepi, Adaptive multiscale finite volume method for multi-phase flow and transport, SIAM Multiscale Model. Simul. 3 (1) (2004) 50–64.
[22] P. Jenny, S.H. Lee, H.A. Tchelepi, Adaptive fully implicit multi-scale finite-volume method for multi-phase flow and transport in heterogeneous porous media, J. Comput. Phys. 217 (2006) 627–641.
[23] H. Zhou, S.H. Lee, H.A. Tchelepi, Multiscale finite-volume formulation for saturation equations, SPE J. 17 (1) (2011) 198–211.
[24] H. Hajibeygi, P. Jenny, Adaptive iterative multiscale finite volume method, J. Comput. Phys. 230 (2011) 628–643.
[25] M.A. Christie, M.J. Blunt, Tenth SPE comparative solution project: A comparison of upscaling techniques, SPE Reserv. Eval. Eng. 4 (2001) 308–317.
[26] V. Kippe, J.E. Aarnes, K.A. Lie, A comparison of multiscale methods for elliptic problems in porous media flow, Comput. Geosci. 12 (3) (2008) 377–398.
[27] H. Hajibeygi, G. Bonfigli, M.A. Hesse, P. Jenny, Iterative multiscale finite volume method, J. Comput. Phys. 227 (2008) 8604–8621.
[28] I. Lunati, M. Tyagi, S.H. Lee, An iterative multiscale finite volume algorithm converging to the exact solution, J. Comput. Phys. 230 (2011) 1849–1864.
[29] G. Bonfigli, P. Jenny, Recent developments in the multi-scale-finite-volume procedure, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), vol. 5910, 2010, pp. 124–131.
[30] H. Zhou, H.A. Tchelepi, Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models, SPE J. SPE-141473-PA (2012).
[31] E. Chow, M.A. Heroux, BPKIT reference manual, Supercomputing Institute Technical Report UMSI 96/183, University of Minnesota, 1996.
[32] D. Peaceman, H. Rachford, The numerical solution of elliptic and parabolic differential equations, J. SIAM 3 (1955) 28–41.
[33] B. Smith, P. Bjorstad, W. Gropp, Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations, Cambridge University Press, 1996.
[34] J. Wallis, H.A. Tchelepi, Apparatus, method and system for improved reservoir simulation using an algebraic cascading class linear solver, U.S. Patent No. 7684967, 2010.
[35] H. Zhou, Algebraic multiscale finite-volume methods for reservoir simulation, PhD thesis, Stanford University, 2010.
[36] N. Remy, A. Boucher, J. Wu, Applied Geostatistics with SGeMS: A User's Guide, Cambridge University Press, New York, 2009.
[37] H. Hajibeygi, S.H. Lee, I. Lunati, Accurate and efficient simulation of multiphase flow in a heterogeneous reservoir with error estimate and control in the multiscale finite-volume framework, SPE J. 17 (4) (2012) 1071–1083, SPE-141954-PA.