

Adaptive Algebraic Multiscale Solver for Compressible Flow in Heterogeneous Porous Media

Matei Tene^{a,1,*} Yixuan Wang^{b,2} Hadi Hajibeygi^{a,3}

^a*Department of Geoscience and Engineering, Faculty of Civil Engineering and Geosciences, Delft University of Technology, P.O. Box 5048, 2600 GA Delft, The Netherlands.*

^b*Department of Energy Resources Engineering, Stanford University, 367 Panama St., Rm. 065, Stanford, CA 94305-2220, USA.*

Abstract

This paper presents the development of an Adaptive Algebraic Multiscale Solver for Compressible flow (C-AMS) in heterogeneous porous media. Similar to the recently developed AMS for incompressible (linear) flows [Wang et al., JCP, 2014], C-AMS operates by defining primal and dual-coarse blocks on top of the fine-scale grid. These coarse grids facilitate the construction of a conservative (finite volume) coarse-scale system and the computation of local basis functions, respectively. However, unlike the incompressible (elliptic) case, the choice of equations to solve for basis functions in compressible problems is not trivial. Therefore, several basis function formulations (incompressible and compressible, with and without accumulation) are considered in order to construct an efficient multiscale prolongation operator. As for the restriction operator, C-AMS allows for both multiscale finite volume (MSFV) and finite element (MSFE) methods. Finally, in order to resolve high-frequency errors, fine-scale (pre- and post-) smoother stages are employed. In order to reduce computational expense, the C-AMS operators (prolongation, restriction, and smoothers) are updated adaptively. In addition to this, the linear system in the Newton-Raphson loop is infrequently updated. Systematic numerical experiments are performed to determine the effect of the various options, outlined above, on the C-AMS convergence behaviour. An efficient C-AMS strategy for heterogeneous 3D compressible problems is developed based on overall CPU times. Finally, C-AMS is compared against an industrial-grade Algebraic MultiGrid (AMG) solver. Results of this comparison illustrate that the C-AMS is quite efficient as a nonlinear solver, even when iterated to machine accuracy.

Key words: multiscale methods, compressible flows, heterogeneous porous media, scalable linear solvers, multiscale finite volume method, multiscale finite element method, iterative multiscale methods, algebraic multiscale methods.

1 Introduction

Accurate and efficient simulation of multiphase flow in large-scale heterogeneous natural formations is crucial for a wide range of applications, including hydrocarbon production optimization, risk management of Carbon Capture and Storage, water resource utilizations and geothermal power extractions. Unfortunately, considering the size of the domain along with the high resolution heterogeneity of the geological properties, such numerical simulation is often beyond the computational capacity of traditional reservoir simulators. Therefore, Multiscale Finite Element (MSFE) [1, 2, 3, 4, 5] and Finite Volume (MSFV) [6, 7] methods and their extensions have been developed to resolve this challenge. A comparison of different multiscale methods, based on their original descriptions, has been studied in the literature [8]. MSFV and MSFE methods map a discrete fine-scale system to a much coarser space. In MultiGrid (MG) terminology [9], this map is considered as a special prolongation operator, represented by locally-supported (and adaptively updated) basis functions [10]. The restriction operator is then defined based on either a Finite Element (MSFE), Finite Volume (MSFV), or a combination of both.

MSFV has been applied to a wide range of applications (see, e.g., [10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21]), thus recommending multiscale as a very promising framework for the next-generation reservoir simulators. However, most of these developments, including the state-of-the-art algebraic multiscale formulation (AMS) [19], have focused on the incompressible (linear) flow equations.

When compressibility effects are considered, the pressure equation becomes nonlinear, and its solution requires an iterative procedure involving a parabolic-type linear system of equations [22]. Therefore, the development of an efficient and general algebraic formulation for compressible nonlinear flows is crucial in order to advance the applicability of multiscale methods towards more realistic problems.

The present study introduces the first algebraic multiscale iterative solver for compressible flows in heterogeneous porous media (C-AMS), along with a thorough study of its computational efficiency (CPU time) and convergence behaviour (number of iterations).

In contrast to cases with incompressible flows, the construction of basis functions for compressible flow problems is not straightforward. In the past, incom-

* Corresponding author.

¹ m.tene@tudelft.nl

² yixuanw@stanford.edu

³ h.hajibeygi@tudelft.nl

pressible elliptic [23], compressible elliptic [10, 24], and pressure-independent parabolic [25] basis functions have been considered. However, the literature lacks a systematic study to reveal the benefit of using one option over the other, especially when combined with a fine-scale smoother stage. Moreover, no study of the overall efficiency of the multiscale methods (based on the CPU time measurements) has been done so far for compressible three-dimensional problems.

In order to develop an efficient prolongation operator, in this work, several formulations for basis functions are considered. These basis functions differ from each other in the amount of compressibility involved in their formulation, ranging from incompressible elliptic to compressible parabolic types. In terms of the restriction operator, both MSFE and MSFV are considered, along with the possibility of mixing iterations of the former with those of the latter, allowing C-AMS to benefit from the Symmetric Positive Definite (SPD) property of MSFE and the conservative physically correct solutions of MSFV.

The low-frequency errors are resolved in the global (multiscale) stage of C-AMS, while high-frequency errors are tackled using a second-stage smoother at fine-scale. In this paper, we consider two options for the smoothing stage: the widely used local correction functions with different types of compressibility involved (i.e., more general than the specific pressure-independent operator [25]), as well as ILU(0) [26]. The best C-AMS procedure is determined among these various strategies, on the basis of the CPU time for 3D heterogeneous problems. It is important to note that the setup and linear system population are measured alongside the solve time - a study which has so far not appeared in the previously published compressible multiscale works.

Though C-AMS is a conservative method (i.e., only a few iterations are enough in order to obtain a high-quality approximation of the fine-scale solution), in the benchmark studies of this work, it is iterated until machine accuracy is reached. And, thus, its performance as an exact solver is compared against an industrial-grade Algebraic MultiGrid (AMG) method, SAMG [27]. This comparative study for compressible problems is the first of its kind, and is made possible through the presented algebraic formulation, which allows for easy integration of C-AMS in existing advanced simulation platforms. Numerical results, presented for a wide range of heterogeneous 3D domains, illustrate that the C-AMS is quite efficient for simulation of nonlinear compressible flow problems.

The paper is structured as follows. First, the Compressible Algebraic Multi-scale Solver (C-AMS) is presented, where several options for the prolongation, restriction operators as well as the second-stage solver are considered. Then the adaptive updating of the C-AMS operators are studied, along with the possibility of infrequent linear system updates in the Newton-Raphson loop.

Numerical results are subsequently presented for a wide range of 3D heterogeneous test cases, aimed at determining the optimum strategy. Finally, the C-AMS is compared with an Algebraic MultiGrid Solver (i.e., SAMG) both in terms of the number of iterations and overall CPU time.

2 Compressible Flow in Heterogeneous Porous Media

Single phase compressible flow in porous media, using Darcy's law (without gravity and capillary effects), can be stated as:

$$\frac{\partial}{\partial t}(\phi\rho) - \nabla \cdot (\rho \boldsymbol{\lambda} \cdot \nabla p) = \rho q, \quad (1)$$

where ϕ , ρ , and q are the porosity, density, and the source terms, respectively. Moreover, $\boldsymbol{\lambda} = \mathbf{K}/\mu$ is the fluid mobility with positive-definite permeability tensor, \mathbf{K} , while μ is the fluid viscosity.

The semi-discrete form of this nonlinear flow equation using implicit (Euler-backward) time integration reads

$$\frac{\phi^{n+1}}{\Delta t} - \frac{\phi^n \rho^n}{\Delta t \rho^{n+1}} - \frac{1}{\rho^{n+1}} \nabla \cdot (\rho^{n+1} \boldsymbol{\lambda} \cdot \nabla p^{n+1}) = q, \quad (2)$$

which is linearized as

$$c^\nu (p^{\nu+1} - p^\nu) - \frac{1}{\rho^\nu} \nabla \cdot (\rho^\nu \boldsymbol{\lambda} \cdot \nabla p^{\nu+1}) = b^\nu, \quad (3)$$

where

$$c^\nu = \frac{1}{\Delta t} \left[\left. \frac{\partial \phi}{\partial p} \right|^\nu - \phi^n \left. \frac{\partial}{\partial p} \left(\frac{1}{\rho} \right) \right|^\nu \rho^n \right] \quad (4)$$

and

$$b^\nu = -\frac{\phi^\nu}{\Delta t} + \frac{\phi^n \rho^n}{\Delta t \rho^\nu} + q. \quad (5)$$

The superscripts (ν) and $(\nu + 1)$ denote the old and new Newton-Raphson iteration levels, respectively. Note that, as $(\nu \rightarrow \infty)$, Eq. (3) converges to the nonlinear Eq. (2), and $(p^{\nu+1} - p^\nu) \rightarrow 0$. Therefore, the coefficient c , which is a by-product of the linearization lemma, plays a role only during iterations. This fact opens up the possibility to alter c by computing it based on either p^ν (resulting in c^ν) or p^n (corresponding to c^n) -the pressure at the previous time-step. Each choice can potentially lead to a different convergence behaviour and, thus, computational efficiency.

Algebraically, Eq. (3) can be written for the unknown pressure vector \mathbf{p} as

$$(\mathbf{C}^\nu + \tilde{\mathbf{A}}^\nu)\mathbf{p}^{\nu+1} \equiv \mathbf{A}^\nu\mathbf{p}^{\nu+1} = \mathbf{f}^\nu \equiv \mathbf{b}^\nu + \mathbf{C}^\nu\mathbf{p}^\nu, \quad (6)$$

where \mathbf{C}^ν is a diagonal matrix having $(c_i^\nu dV_i)$ at cell i in its i -th diagonal entry, where dV_i is the volume of cell i . Also, $\tilde{\mathbf{A}}$ is the convective compressible flow matrix, having fine-scale transmissibilities computed on the basis of a finite-volume scheme as entries. Moreover, the vector \mathbf{b}^ν contains the integrated source terms in the fine-scale volumes, i.e., $(q_i dV_i)$. The total Right-Hand-Side (RHS) terms are denoted by the vector \mathbf{f}^ν .

3 Compressible Algebraic Multiscale Solver (C-AMS)

The C-AMS relies on the primal- and dual-coarse grids, which are superimposed on the fine-scale grid (See Fig. 1). There are N_p and N_d coarse and dual-coarse grid cells in a domain with N_f fine-grid cells.

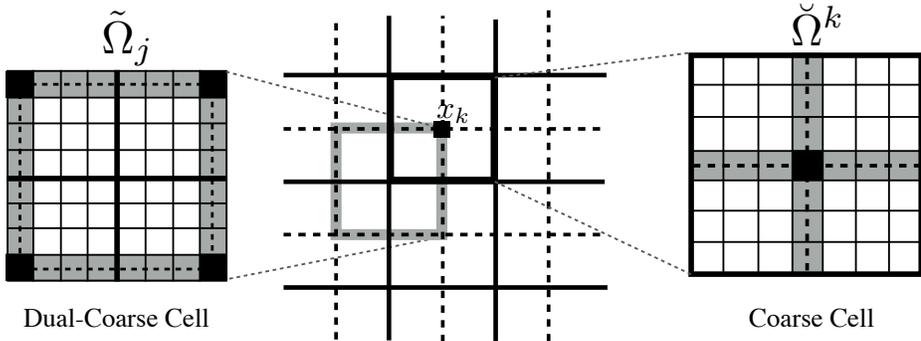


Fig. 1. Multiscale grids imposed on the given fine grid. A primal- and a dual-coarse block are highlighted on the right and left sides, respectively.

The transfer operators between fine-scale and coarse-scale are defined as the multiscale Restriction (\mathbf{R}) and Prolongation (\mathbf{P}). The former is defined based on either Finite Element (MSFE), i.e., $\mathbf{R}^{FE} = \mathbf{P}^T$, or Finite Volume (MSFV), for which \mathbf{R}^{FV} corresponds to the integral over primal-coarse blocks, i.e.,

$$\mathbf{R}^{FV}(i, j) = \begin{cases} 1 & , \text{ if fine-cell } j \text{ is contained in primal-coarse block } i \\ 0 & , \text{ otherwise.} \end{cases} \quad (7)$$

The columns of \mathbf{P} are the basis functions, which are computed on dual-coarse cells (see Fig. 1), subject to simplified boundary conditions (the localization assumption).

In contrast to the (incompressible) AMS, C-AMS can be formulated based on different choices of the basis functions, depending on the level of compressibility involved. The first two types read

$$c^\nu \Phi_{k,h}^{\nu+1} - \frac{1}{\rho^\nu} \nabla \cdot (\rho^\nu \boldsymbol{\lambda} \cdot \nabla \Phi_{k,h}^{\nu+1}) = 0 \quad (8)$$

and

$$-\frac{1}{\rho^\nu} \nabla \cdot (\rho^\nu \boldsymbol{\lambda} \cdot \nabla \Phi_{k,h}^{\nu+1}) = 0, \quad (9)$$

both being pressure dependent (through c and ρ), but different in the sense of the consideration of the accumulation term, c . Alternatively, one can also formulate basis functions using

$$c^n \Phi_{k,h} - \nabla \cdot (\boldsymbol{\lambda} \cdot \nabla \Phi_{k,h}) = 0, \quad (10)$$

or

$$-\nabla \cdot (\boldsymbol{\lambda} \cdot \nabla \Phi_{k,h}) = 0 \quad (11)$$

which are both pressure independent (since c is now based on the pressure from the previous time step). All of these equations are subject to reduced-problem boundary conditions along dual-coarse cell boundaries $\partial\tilde{\Omega}^h$ [19]. One can also obtain the equations for the corresponding four types of local correction functions, Ψ_h , by substituting the corresponding RHS term in Eqs. (8)-(11). As mentioned before, in this work, systematic studies on the basis of the CPU time as well as the number of iterations are performed in order to find the optimum formulation for basis function (i.e., prolongation operator).

The basis functions Φ_k are assembled over dual-coarse cells $\tilde{\Omega}^h$, $\forall h \in \{1, \dots, N_d\}$, i.e., $\Phi_k = \bigcup_{h=1}^{N_d} \Phi_{k,h}$, and, if used, the correction functions are also assembled as $\Psi = \bigcup_{h=1}^{N_d} \Psi_h^\nu$. Fig. 2 illustrates that the basis functions do not form a partition of unity when compressibility effects are included, which is the intrinsic nature of the parabolic compressible equation.

The choices formulated above affect computational efficiency of constructing and updating the multiscale operators. More precisely, while basis functions of Eqs. (8) and (9) depend on pressure (hence, updated adaptively when pressure changes), Eqs. (10) and (11) are pressure independent; thus, they only need to be computed once for single-phase problems (for multi-phase flows, they need to be adaptively updated when local transmissibility changes beyond a prescribed threshold value). While the basis and correction functions from Eq. (10) were previously used [25], the other options are, as of yet, have not been studied.

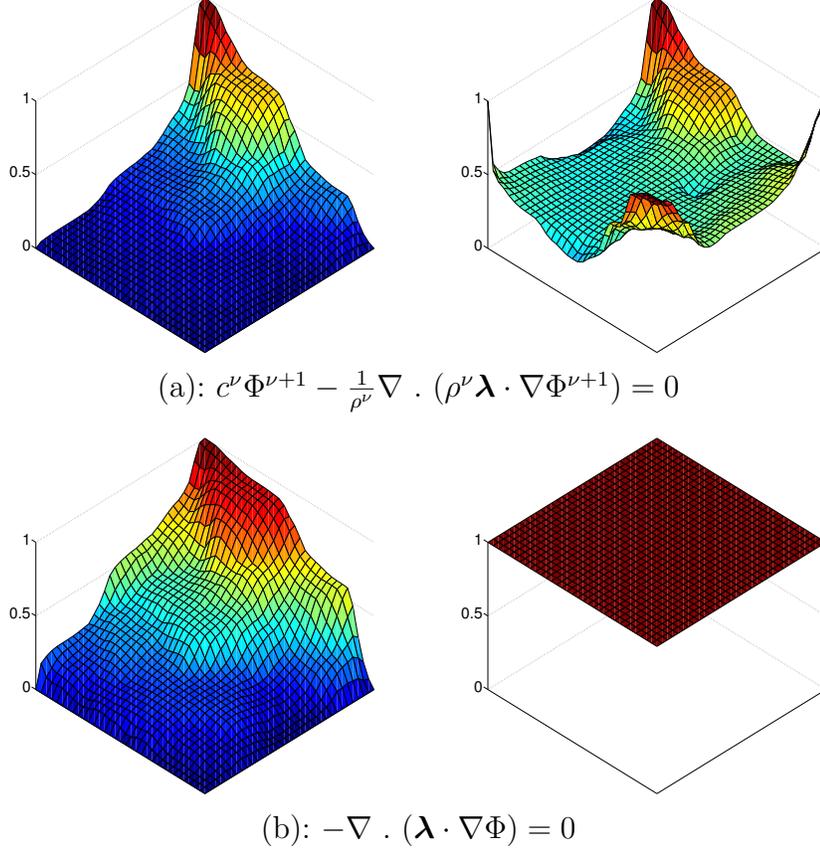


Fig. 2. Two choices of multiscale basis functions in a reference dual-coarse block (left column), Summation of the basis functions over the dual-coarse block (right column), i.e., partition of unity check.

The C-AMS approximates the fine-scale solution \mathbf{p}^ν by \mathbf{p}'^ν using the Prolongation operator \mathbf{P} , which is a matrix of size $N_f \times N_c$, having basis function Φ_k in its k -th column. The map between the coarse ($\check{\mathbf{p}}$) and fine-scale solution (\mathbf{p}') reads

$$\mathbf{p}' = \mathbf{P}\check{\mathbf{p}}. \quad (12)$$

The coarse-scale system is obtained using the restriction operator, \mathbf{R} , as

$$\check{\mathbf{A}}^\nu \check{\mathbf{p}}^{\nu+1} \equiv (\mathbf{R}\mathbf{A}^\nu \mathbf{P})\check{\mathbf{p}}^{\nu+1} = \mathbf{R}\mathbf{f}^\nu, \quad (13)$$

and its solution is prolonged to the fine-scale using Eq. (12), i.e.,

$$\mathbf{p}'^{\nu+1} = \mathbf{P}(\mathbf{R}\mathbf{A}^\nu \mathbf{P})^{-1} \mathbf{R}\mathbf{f}^\nu. \quad (14)$$

In residual form, it reads

$$\delta \mathbf{p}'^{\nu+1} = \mathbf{P}(\mathbf{R}\mathbf{A}^\nu \mathbf{P})^{-1} \mathbf{R}\mathbf{r}^\nu. \quad (15)$$

Here $\mathbf{p}'^{\nu+1} = (\mathbf{p}^\nu + \delta \mathbf{p}'^{\nu+1})$, while $\mathbf{r}^\nu = (\mathbf{f}^\nu - \mathbf{A}^\nu \mathbf{p}^\nu)$ is the fine-scale residual.

Note that all the different options for basis functions can be considered in construction of the prolongation operator.

The C-AMS employs Eq. (14) as the global solver (for resolving low-frequency errors). In addition to the coarse-scale solver, an efficient convergent multiscale solver needs to include a second-stage smoother at fine scale. The smoother accounts for the high-frequency errors, arising from simplified localization conditions, the nonlinearity of the operator, and the complex RHS term. Among the choices for this smoother (block-, line-, or point-wise solvers), the correction functions (CF) and ILU(0) are considered in this work. The C-AMS procedure is finally summarized in Table 1.

- Do until convergence ($\|\epsilon\|_2^{\nu} < e$) achieved (See Eq. (19)) {
1. **Initialize:** update linear system components, \mathbf{A}^{ν} and \mathbf{f}^{ν} , based on \mathbf{p}^{ν}
 2. **Update residual:** $\mathbf{r}^{\nu} = \mathbf{f}^{\nu} - \mathbf{A}^{\nu}\mathbf{p}^{\nu}$
 3. **Adaptively compute Basis Functions:** use either of Eqs. (8)-(11)
 4. **Pre-smoothing Stage:** only if CF is used, apply CF on \mathbf{r}^{ν} and update residual
 5. **Multiscale Stage:** Solve (15) for $\delta\mathbf{p}^{\nu+1/2}$
 6. **Post-smoothing Stage:** smooth $\delta\mathbf{p}^{\nu+1/2}$ for n_s times using a fine-scale iterative solver (here, ILU(0) is used), obtaining $\delta\mathbf{p}^{\nu+1}$
 7. **Update solution:** $\mathbf{p}^{\nu+1} = (\mathbf{p}^{\nu} + \delta\mathbf{p}^{\nu+1/2} + \delta\mathbf{p}^{\nu+1})$
 8. **Update error:** compute ϵ^{ν} , and assign $\mathbf{p}^{\nu} \leftarrow \mathbf{p}^{\nu+1}$
- }

Table 1

C-AMS iteration procedure, converging to \mathbf{p}^{n+1} with tolerance e .

In the next section, numerical results for 3D heterogeneous test cases are presented, in order to provide a thorough assessment of the applicability of C-AMS to large-scale problems.

4 Numerical Results

The numerical experiments presented in this section are divided into: (1) finding a proper iterative procedure and multi-stage multiscale components for efficiently capturing the nonlinearity within the flow equation, and (2) systematic performance study by comparing against a commercial algebraic multigrid solver, i.e., SAMG [27]. Note that the second aspect is mainly to provide the computational physics community with an accurate assessment of the convergence properties of the state-of-the-art compressible multiscale solver (i.e., C-AMS). As an advantage over many advanced linear solvers, C-AMS allows

for construction of locally conservative velocity after any MSFV stage. Therefore, for multiphase flow scenarios, only a few C-AMS iterations are necessary to obtain accurate solutions [16].

For the studied numerical experiments of this paper, sets of log-normally distributed permeability fields with spherical variograms are generated by using sequential Gaussian simulations [28]. The variance and mean of natural logarithm of the permeability, i.e., $\ln(k)$, for all test cases are 4 and -1, respectively, unless otherwise is mentioned. Furthermore, the fine-scale grid size and dimensionless correlation lengths in the principle directions, i.e., ψ_1 , ψ_2 and ψ_3 , are provided in Table 2. Each set has 20 statistically-equivalent realizations. The sets with orientation angle of 15° are referred to as the layered fields. Also, the grid aspect ratio α is 1, i.e., $\Delta x/\alpha = \Delta y = \Delta z = 1$ m, unless otherwise is specified.

Permeability Set	1	2	3	4	5	6
Fine-scale grid	64^3	128^3	256^3	64^3	128^3	256^3
ψ_1	0.125	0.125	0.125	0.5	0.5	0.5
ψ_2	0.125	0.125	0.125	0.03	0.03	0.03
ψ_3	0.125	0.125	0.125	0.01	0.06	0.01
Angle between ψ_1 and y direction	patchy			15°		
Variance of $\ln(k)$	4					
Mean of $\ln(k)$	-1					

Table 2

Permeability sets (each with 20 statistically-equivalent realizations) used for numerical experiments of this paper. Layered fields refer to the sets 4-6, in which the orientation angle between ψ_1 and y direction is 15° .

Phase properties and simulation time are described as non-dimensional numbers. The non-dimensional pressure and density are introduced as

$$p^* = \frac{p - p_{east}}{p_{west} - p_{east}}, \quad (16)$$

and

$$\rho^* = \frac{\rho}{\rho_0} = 1 + \eta p^*, \quad (17)$$

respectively, where the coefficient η is set to 1 for all subsequent test cases in this paper.

The p_{west} and p_{east} values of 10^6 and 0 Pa, relative to the Standard (Atmospheric) condition, are considered. These correspond to non-dimensional pressure values of 1 and 0, which are set as Dirichlet conditions at the west and

east boundaries, respectively, for all the cases unless otherwise is mentioned. Also, all the other surfaces are subject to no-flow Neumann conditions.

The non-dimensional time is introduced as $t^* = t/\tau$, where

$$\tau = \frac{\mu\phi L^2}{\bar{K}(p_{west} - p_{east})}. \quad (18)$$

Here, \bar{K} is the average permeability, and L is a length scale of the domain. With the values of 10^6 Pa pressure difference, in-situ viscosity of 2×10^{-6} Pa.s, $\Delta x = 1$ m, $\phi = 0.1$, and \bar{K} value of 10^{-12} m² for homogeneous cases, the τ will be 128 s for problem size of $L = 64$ m in SI units.

The implementation used to obtain the results presented in this paper consists of a single-threaded object-oriented C++ code, and the CPU times were measured on an Intel Xeon E5-1620 v2 quad-core system with 64GB RAM.

4.1 C-AMS: determining the most effective iterative procedure and multi-stage multiscale components

The efficient capturing of the nonlinearity within the iterations is important in designing an efficient multiscale strategy. For the purposes of a conclusive result, in this section, a set of 20 statistically-equivalent patchy fields, i.e., permeability Set 1 from Table 2, is considered. One of the realizations and its corresponding solution at $t^* = 0.4$ are shown in Fig. 3.

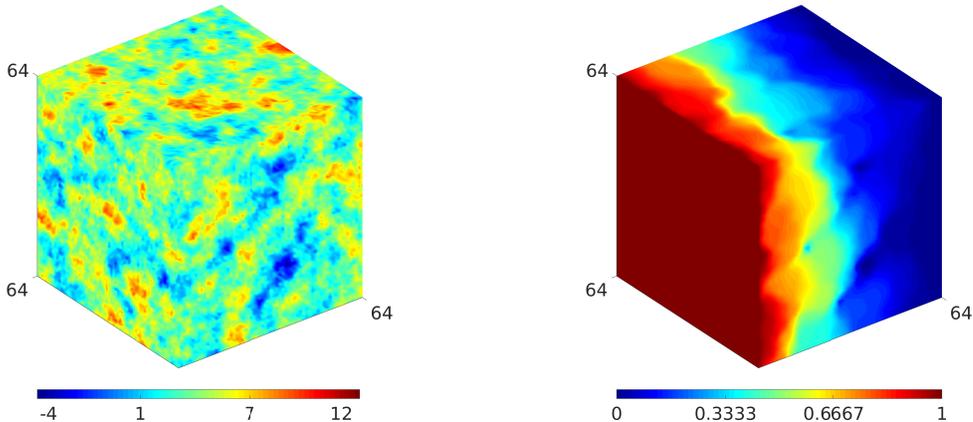


Fig. 3. Natural-log of the permeability (left) and pressure solution after $t^* = 0.4$ (right) corresponding to one of the realization of permeability Set 1 from Table 2.

4.1.1 Nonlinear and linear level updates

In formulating a convergence criterion for the C-AMS, one can express the error of the approximate solution at step ν on the basis of either the linear or nonlinear expressions. According to Eq. (2), the nonlinear error in each grid cell reads

$$\epsilon^\nu = q - \frac{\phi^{n+1}}{\Delta t} + \frac{\phi^n}{\Delta t} \frac{\rho^n S_\alpha^n}{\rho^{n+1}} + \frac{1}{\rho^{n+1}} \nabla \cdot (\rho^{n+1} \boldsymbol{\lambda} \cdot \nabla p^{n+1}), \quad (19)$$

and is assembled in the vector $\boldsymbol{\epsilon}^\nu$, which allows the computation of the *error norm*, $\|\boldsymbol{\epsilon}^\nu\|_2$. On the other hand, the linear-level error is based on the linearized equation (3), which leads to the computation of the *residual norm*, $\|\mathbf{r}^\nu\|_2$.

In order to determine a suitable sequence of the linear and nonlinear stages, the same patchy domain of $64 \times 64 \times 64$ grid cells is considered (Fig. ??), for which the pressure equation is solved using the following solution strategy:

- Do until ($\|\boldsymbol{\epsilon}\|_2 < 10^{-6}$) is reached {
0. Update parameters, linear system matrix and RHS vector based on \mathbf{p}^ν
 1. Solve linear system using the Richardson iterative scheme, preconditioned with one multigrid V-cycle until $\|\mathbf{r}\|_2 < 10^{-6}$
- }

Table 3

Solution strategy used to determine a suitable stopping criterion

The error and residual norms were recorded after each iteration of the Richardson loop and are presented in Fig. 4. Note that the reduction of the residual norm beyond the first few iterations does not contribute to the reduction in the (nonlinear) error norm. Therefore, one could ideally speed up the solution scheme by monitoring the error norm and updating the linear system after its decrease starts to stagnate. However, the computational cost of evaluating the nonlinear equation is roughly the same as that of a linear system update and, thus, much more expensive than the evaluation of the residual norm.

Fig. 4(a) also reveals that the stagnation of the error norm happens roughly after the residual norm has been approximately reduced by 1/10 of its initial value (i.e., immediately after the linear system update). Fig. 4(b) shows the convergence behaviour after implementing this heuristic strategy, which is deemed quite efficient, since the two norms are in agreement. Hence, in the following experiments the same strategy is employed, i.e., for linear level, $\frac{\|\mathbf{r}^i\|_2}{\|\mathbf{r}^0\|_2} < 10^{-1}$ after iteration i of the inner (linear) loop and, for nonlinear level, $\|\boldsymbol{\epsilon}^\nu\|_2 < 10^{-6}$ after iteration ν of the outer (nonlinear) loop are set (see Table 3).

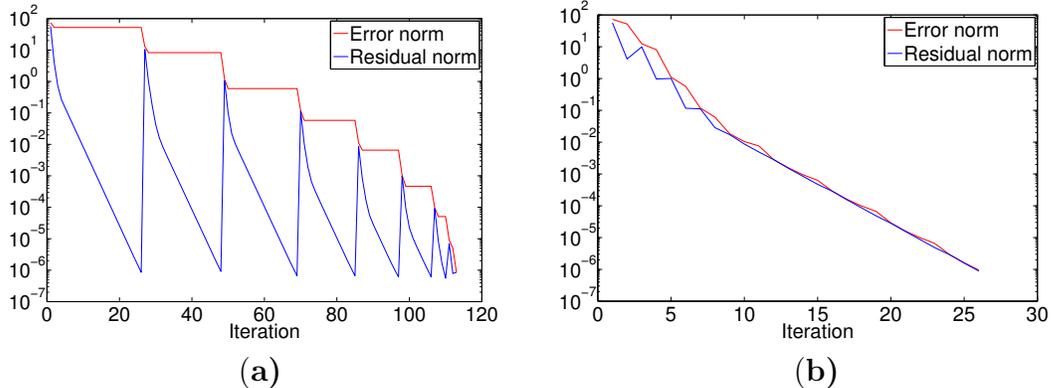


Fig. 4. Error and residual norm histories for one of the realizations of permeability Set 1 from Table 2 over a single time step of $t^* = 0.4$. Shown on the left is the strategy where at each nonlinear stage, the fully converged linear solution is obtained. Shown on the right is the strategy where in each outer (nonlinear) loop the residual is reduced only by one order of magnitude.

4.1.2 Adaptive updating of multiscale operators

The previous study described the first adaptive aspect considered in this work, namely, updating the linear system only after the residual norm drops by an order of magnitude. The C-AMS procedure can be further optimized by employing adaptive updates of its multiscale components, i.e., the basis and (if considered) the correction functions. To this end, one has to monitor the changes in the entries of the transmissibility matrix \mathbf{A} and RHS \mathbf{f} between the iteration steps. Fig. 5(c) shows that the adaptive update of the C-AMS basis functions leads to a significant speed-up in terms of CPU time.

Furthermore, the two adaptivity methods (for linear system and local function updates) are combined and shown in Fig. 5(d). Hence, C-AMS will perform its iterations such that it exploits all adaptivity within the multiscale components and the nonlinearity within the flow equation. Note that for this case, the compressible variant from Eq. (8) was used for both basis and correction functions. However, if the incompressible Eqs (10) and (11) are used, then the basis functions do not require updates during iterations. Finally, for this and all the following results (unless otherwise stated), the C-AMS coarsening ratio was taken as $8 \times 8 \times 8$, because it was found efficient (see Subsection 4.1.6).

4.1.3 C-AMS global stage: choice of basis functions

The aim of this study is to determine an optimum choice for the type of basis functions for the C-AMS algorithm. The correction function is computed based on Eq. (8) in all cases (and, hence, updated adaptively with pressure), 20 iterations of ILU(0) are used for smoothing and all possibilities for the basis functions, i.e., Eqs. (8)-(11), are considered. Finally, there is a single

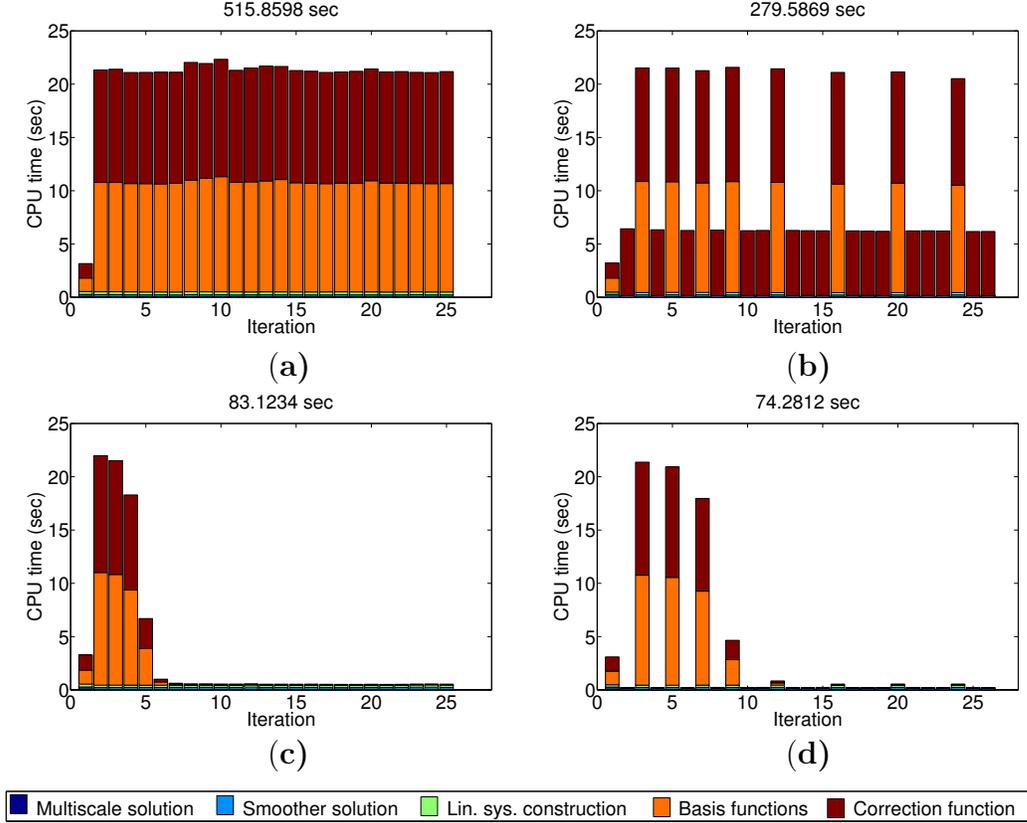


Fig. 5. Effect of different types of adaptivity on the C-AMS performance for the permeability Set 1 from Table 2 after a time step of $t^* = 0.4$: (a): No adaptivity, (b): Linear system update adaptivity only, (c): Multiscale operator update adaptivity only, (d): Fully adaptive, i.e in terms of both linear system and multiscale operator updates.

time step in the simulation, which takes the initial solution at time 0 ($p_0^* = 0$ everywhere) to the solution at time $t^* = 0.4$.

The total CPU time spent in each stage of the solver, as well as the number of iterations (given on top of each bar in Fig. 6), are measured. Also, the success rate of convergence is given inside parentheses beside the average number of iterations.

The results show that including compressibility in the basis functions does not translate into faster convergence and, thus the additional CPU time required to adaptively update them is not justified. In fact, it is more efficient to use the incompressible (pressure independent) basis functions from Eqs. (10) and (11). Also, the inclusion of the accumulation term and the type of Restriction (MSFE or MSFV) does not play an important role for this patchy test case. Note that none of the choices results in 100% successful convergence, even though 20 ILU(0) smoothing iterations have been employed at each iteration.

This can be attributed to the use of correction functions, as investigated in the next paragraph.

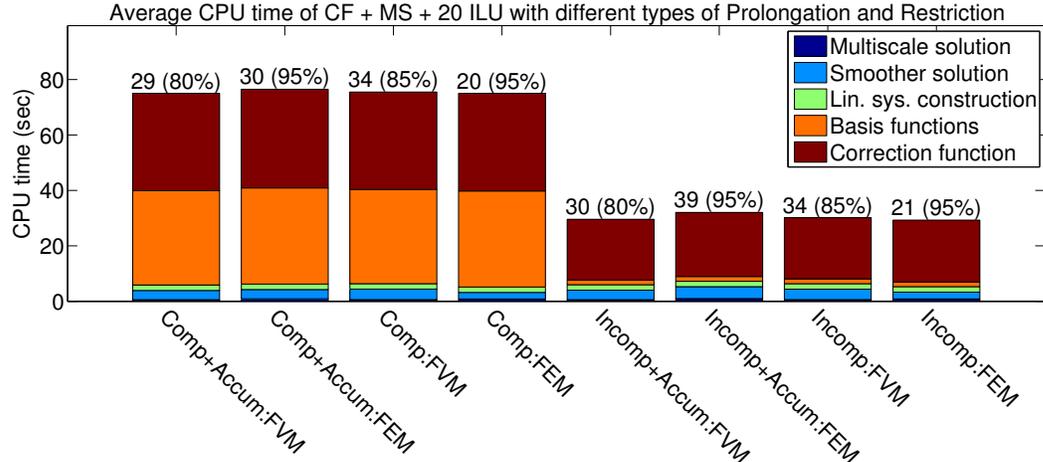


Fig. 6. Effect of the choice of basis function on the C-AMS performance for the 64^3 grid-cell problem after a time step of $t^* = 0.4$. Results are averaged over 20 statistically-equivalent realizations. The number of iterations is shown on top of each bar. The success percentage is also shown in parentheses. Note that all simulations employ correction functions.

4.1.4 C-AMS smoothing stage: choice of correction function

Note that none of the results from the previous test case (Fig. 6) has a 100% success rate. As described in [19], the CF can be seen as an independent stage, the inclusion of which should be seen as an option and not a necessity for convergence. Fig. 7 presents the results of rerunning the previous experiment, this time varying the type of correction function. The plot confirms that eliminating the CF altogether leads to an overall speed-up, and, in addition, a convergence success rate of 100%. As described in [19], this can be explained by the sensitivity of CF to the heterogeneity of the permeability field, which leads to solver instability. Therefore, the CF should not be considered as candidate for the pre-smoothing stage in an efficient C-AMS procedure. Instead, ILU(0) is performed as post-smoother in order to resolve high-frequency errors.

4.1.5 C-AMS smoothing stage: number of smoother iterations

Another variable in the C-AMS framework is the number of smoothing steps (here, ILU(0)) that should be applied in order to obtain the best trade-off between convergence rate and CPU time. The results of several experiments with the optimum choices (i.e., incompressible basis functions and no incorporation of CF) and various numbers of ILU applications are illustrated in Fig. 8. It is clear that with this C-AMS setup, an optimum scenario would

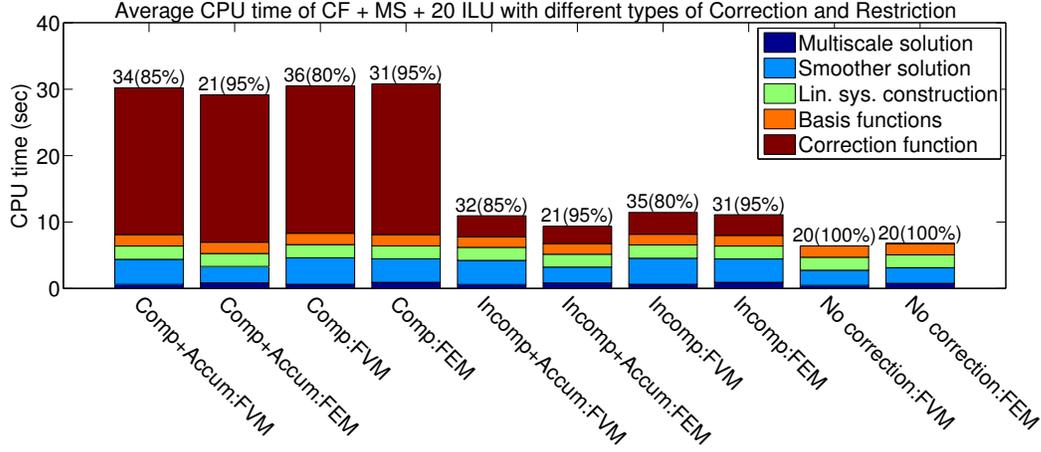


Fig. 7. Effect of the choice of correction function on the CPU time of the multiscale solution on the permeability Set 1 from Table 2 after a time step of $t^* = 0.4$. The number of iterations is shown on top of each bar. Only the last 2 bars on the right correspond to runs in which no correction function was used (i.e., MS + 20 ILU).

be found with 5-10 ILU iterations per second-stage call. Note that all C-AMS runs (without correction functions) converged successfully.

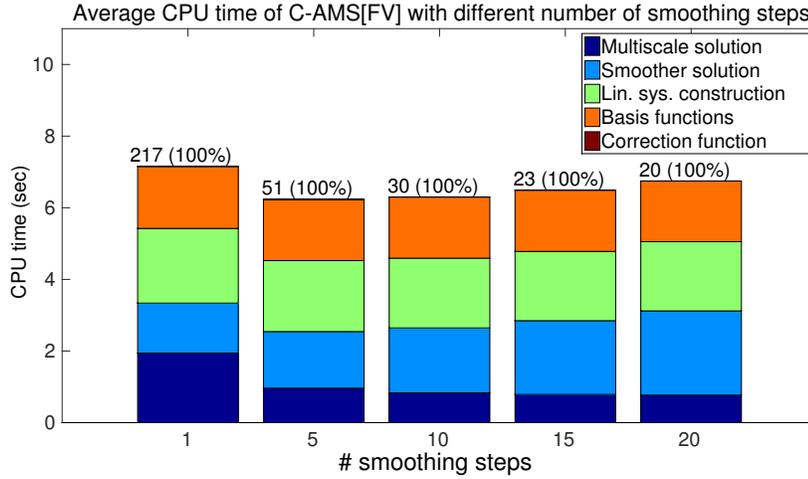


Fig. 8. Effect of the number of ILU smoothing steps on the C-AMS[FV] performance for the permeability Set 1 from Table 2 (grid aspect ratio is 1) after a time step of $t^* = 0.4$. The number of iterations is shown on top of each bar, with convergence success rate inside parentheses. Note that excluding CF leads to 100% success rate for all scenarios.

4.1.6 C-AMS sensitivity to coarsening ratio: trade-off between size of coarse system and local problem cost

The coarsening factors used in this paper were found to be optimal after a careful study of the C-AMS sensitivity with the coarsening ratio. As for a thorough study of the new C-AMS solver, it is important to illustrate also its

sensitivity with change of coarse-scale system size (and thus the coarsening ratio). This important fact is studied and shown in Figs. 9-11 for patchy fields. Not that for the cases studied in this paper, the optimum overall CPU times were obtained with coarse-grid cells with the size of (approximately) the square-root of the domain length in each direction.

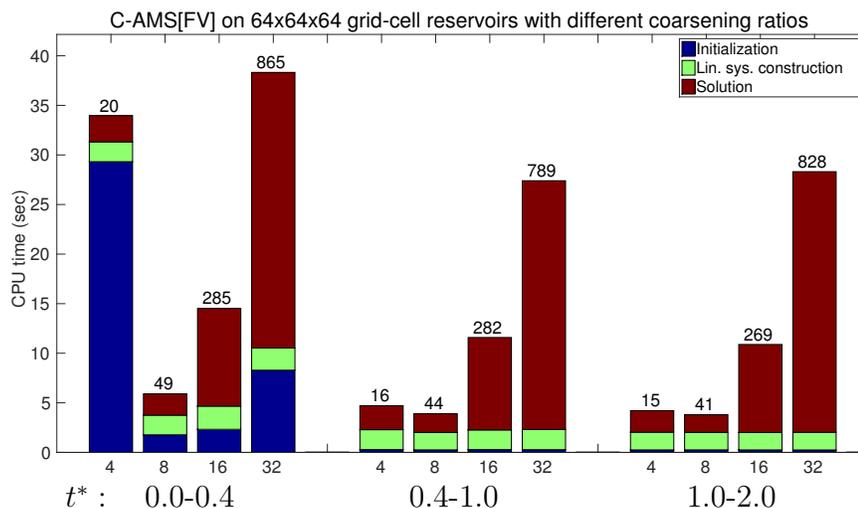


Fig. 9. Patchy fields: Averaged CPU time (over 20 realizations) of C-AMS[FV] for different coarsening ratios for the permeability Set 1 from Table 2. Results support the use of coarsening ratio of 8^3 . A similar behaviour was observed with the FE restriction operator.

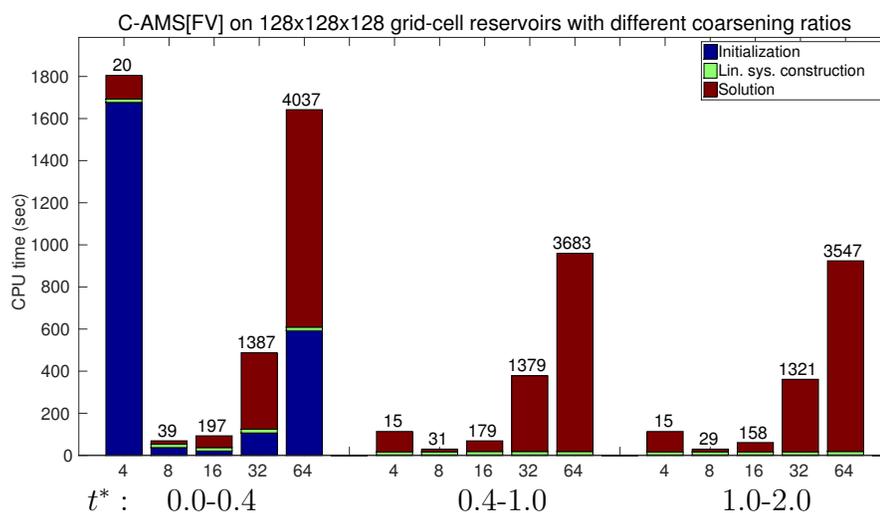


Fig. 10. Patchy fields: Averaged CPU time (over 20 realizations) comparison of C-AMS [FV] for different coarsening ratios for the permeability Set 2 from Table 2. Results support the use of coarsening ratio of 8^3 . A similar behaviour was observed with the FE restriction operator.

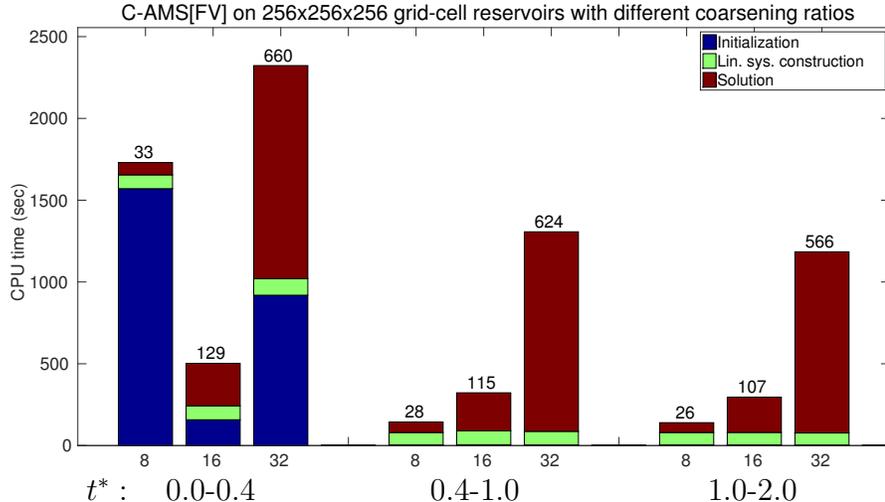


Fig. 11. Patchy fields: Averaged CPU time (over 20 realizations) comparison of C-AMS[FV] for different coarsening ratios for the permeability Set 3 from Table 2. A coarsening ratio of 16^3 offers the best balance between initialization (basis function computation) and solution time, while 8^3 results in a more expensive initialization but faster convergence in subsequent time-steps. A similar behaviour was observed with the FE restriction operator.

4.2 C-AMS benchmark versus SAMG

On the basis of the previously presented studies, the optimal C-AMS strategy includes a global multiscale stage using incompressible basis functions (Eq. (11)), accompanied by 5 iterations of ILU for post-smoothing. In this subsection, C-AMS is compared against SAMG for three sets of different test cases: (1) the heterogeneous domains of different sizes from Table 2; (2): permeability Set 1 from Table 2 with stretched grids and line-source terms; and, (3): permeability Set 1 from Table 2 with different $\ln(k)$ variances (i.e., permeability contrasts).

In all the presentd experiments, SAMG is called to perform a single V-cycle, repeatedly in a Richardson loop. Its adaptivity is controlled manually, i.e., at the beginning of each Newton-Raphson outer iteration, SAMG is allowed to update its Galerkin operators. On the other hand, during linear iterations, SAMG is instructed to reuse its previous grids and operators. For the test cases considered here, this approach was found more efficient (by a factor in excess of 2) than the *automatic solver control* described in [27], In all other aspects, SAMG has been used as a black-box commercial solver.

4.2.1 Test case 1: heterogeneous domains of different sizes from Table 2

In this subsection, C-AMS is compared against the SAMG algebraic multigrid solver for both patchy and layered permeability fields of Table 2 over 3 consecutive time steps. The time-lapse pressure solution for one patchy and one layered sample are shown in Fig. 12, illustrating the propagation of the signal from the western face through the entire domain.

Figs. 13 and 14 show the number of iterations and CPU time at 3 consecutive non-dimensional times for different problem Sets 1, 2, 4, and 5 from Table 2. Note that C-AMS with FV-based restriction operator did not converge in some of the test cases, while the FE-based variant achieved 100% success rate due to its SPD property. Therefore, an ideal solution strategy would use MSFE to converge to the desired level of accuracy and then employ a single MSFV sweep, in order to ensure mass conservation [16].

In addition, Figs. 15 illustrate CPU time (vertical axis) and the total number of iterations (on top of each column), for permeability Sets 3 and 6 from Table 2, with 8^3 and 16^3 coarsening ratios.

Note that, except for the first time-step, when all the basis functions are fully computed, C-AMS has a slight edge over SAMG, mainly due to its adaptivity and relatively inexpensive iterations. The initialization cost of C-AMS is particularly high in the 256^3 case, due to the large number of linear systems (solved with a direct solver) needed for the basis functions. It is clear from Fig. 15 that with larger primal-coarse blocks C-AMS requires less setup time, but more iterations to converge. Note that all performance studies presented in this paper are for single-process computations.

Since reservoir simulators are typically run for many time-steps, the high initialization time of C-AMS is outweighed by the efficiency gained in subsequent steps. Moreover, given the local support of the basis functions, this initialization can be greatly improved through parallel processing. Furthermore, only a few multiscale iteration may prove necessary to obtain an accurate approximation of the pressure solution in each time step for multi-phase flow problems.

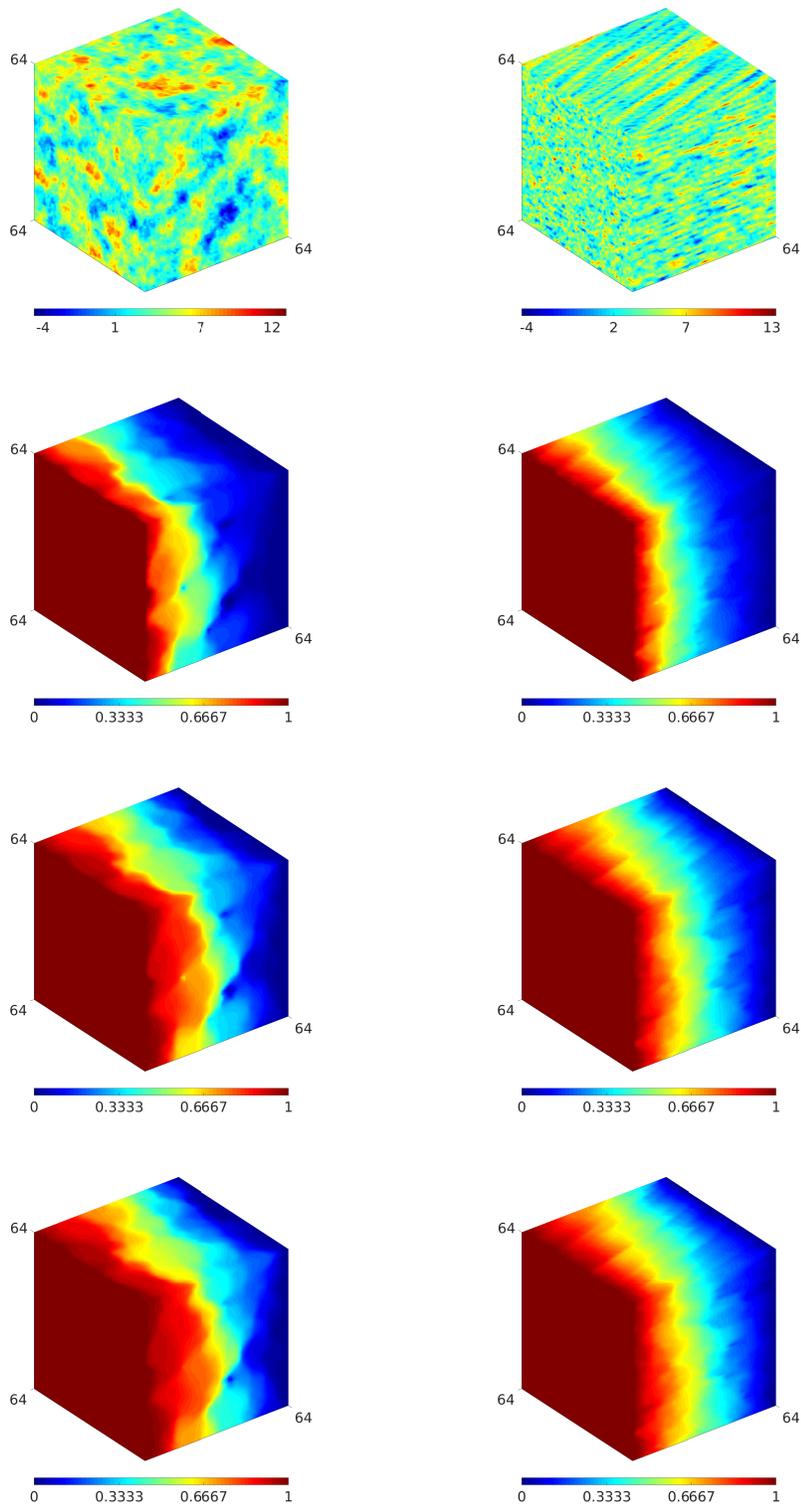


Fig. 12. Pressure solution on one of the realizations of permeability Sets 1 (left) and 4 (right) from Table 2 at $t^* = 0.4, 1.0,$ and 2.0 from top to bottom, respectively.

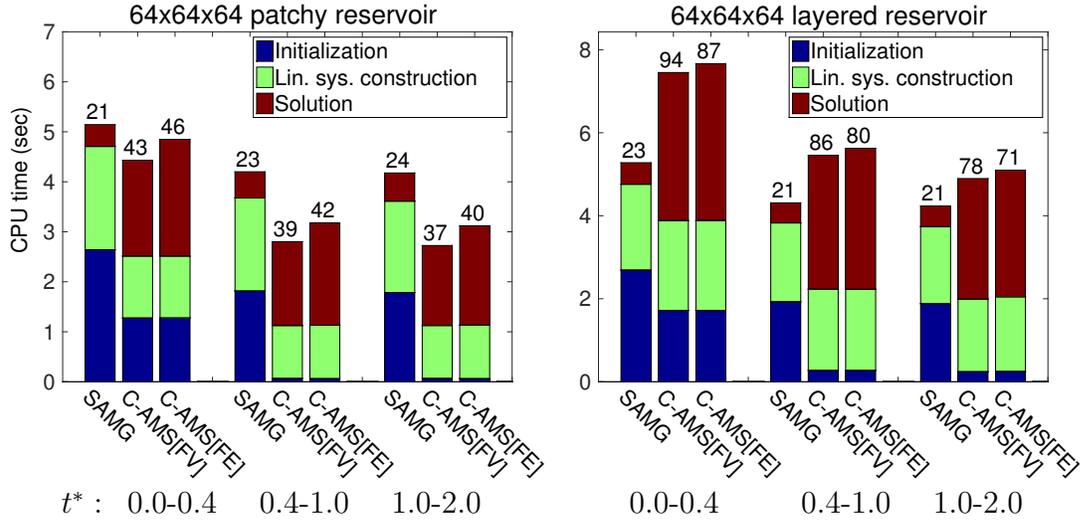


Fig. 13. Averaged CPU time (over 20 realizations) comparison between the C-AMS and SAMG solvers on permeability Sets 1 (a) and 4 (b) from Table 2 over 3 successive time-steps. The coarsening ratio of C-AMS is 8^3 . Moreover, C-AMS employs 5 ILU smoothing steps per iteration. The number of iterations is given on top of each bar.

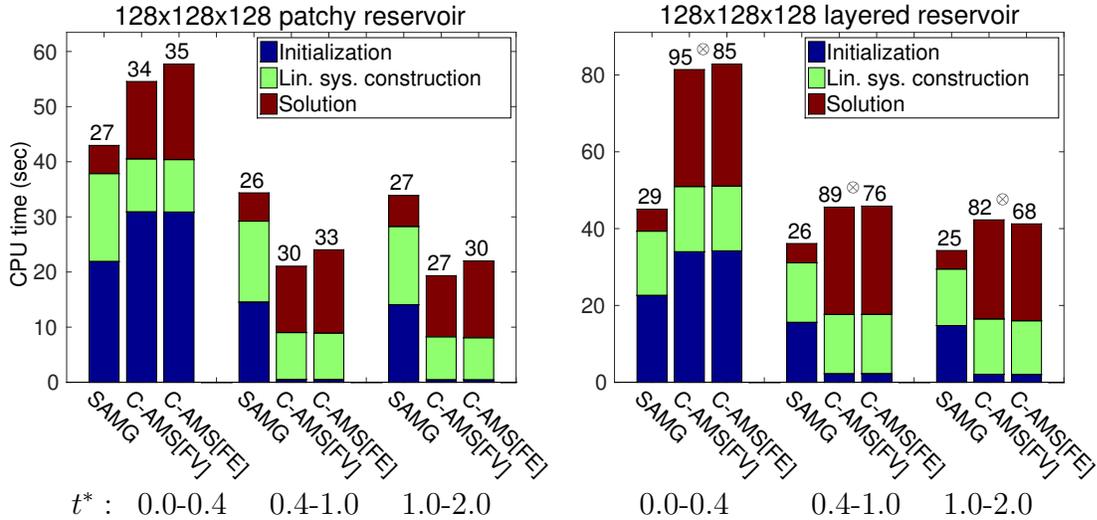


Fig. 14. Averaged CPU time (over 20 realizations) comparison between the C-AMS and SAMG solvers on permeability Sets 2 (a) and 5 (b) from Table 2 over 3 successive time steps. C-AMS employs the coarsening ratio of 8^3 , along with 5 ILU smoothing steps per iteration. The number of iterations is given on top of each bar. The \otimes symbol signifies convergence success rate of 72% when FV-based restriction operator was employed.

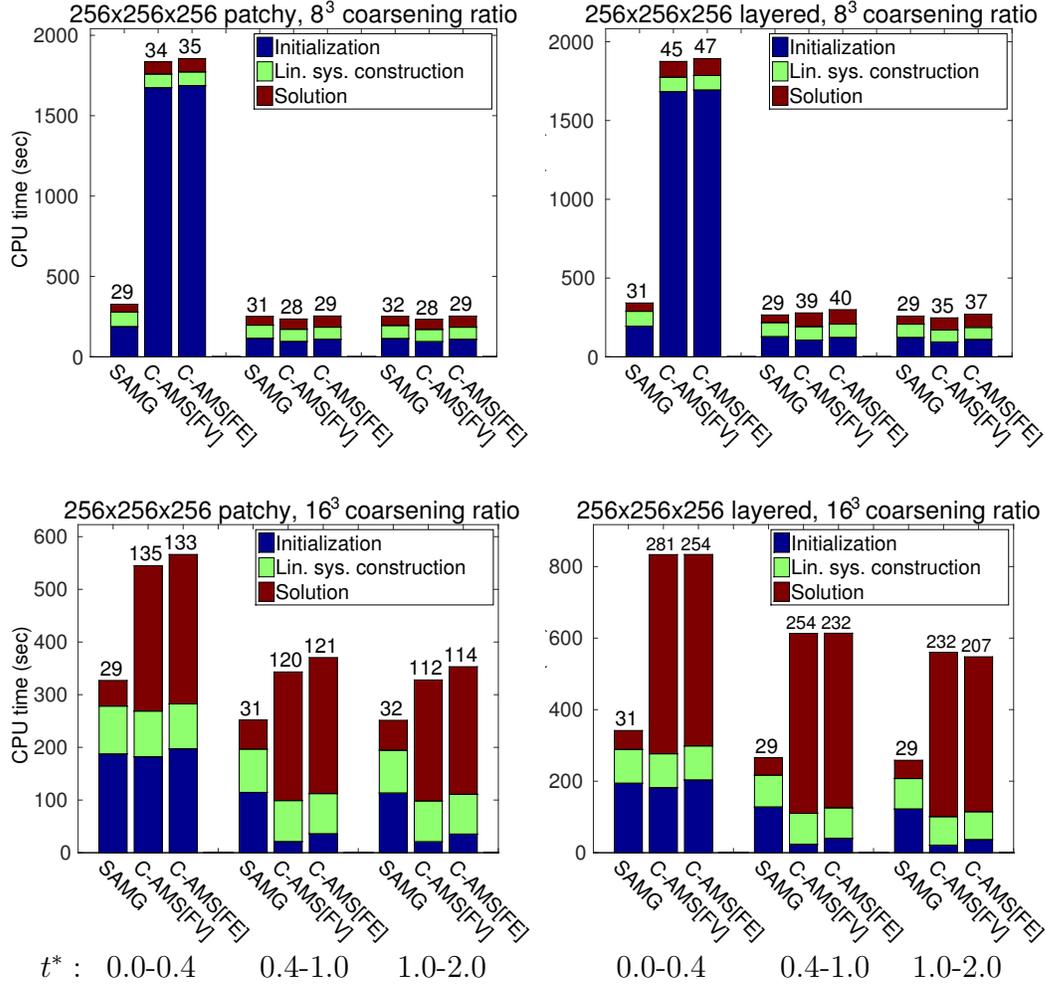


Fig. 15. Averaged CPU time (over 20 realizations) comparison between the C-AMS and SAMG solvers on permeability Sets 3 (left column) and 6 (right column) from Table 2 over 3 successive time-steps. Different coarsening ratios of 8^3 (top row) and 16^3 (bottom row) are considered for C-AMS. Moreover, C-AMS employs 5 ILU smoothing steps per iteration. The number of iterations is given on top of each bar.

4.2.2 Test Case 2: stretched grids with line-source terms

To study the effect of anisotropic permeability fields along with radial injection flow pattern, the permeability Set 1 from Table 2 is considered. The settings are all the same as previous test cases, except the following items. Dirichlet boundary conditions are set at the centers of two vertical sets of fine-scale grid cells: one from $(1,1,1)$ to $(1,1,64)$ and the other from $(64,64,1)$ to $(64,64,64)$ with the values of 1 and 0, respectively. In addition, grid aspect ratios of $\alpha = 1, 5,$ and 10 are considered (Note that $\Delta x/\alpha = \Delta y = \Delta z$). The non-dimensional time is calculated using αL as the characteristic length. Figure 16 illustrates the pressure solutions for one of the permeability realizations after the first time step $t^* = 0.4$.

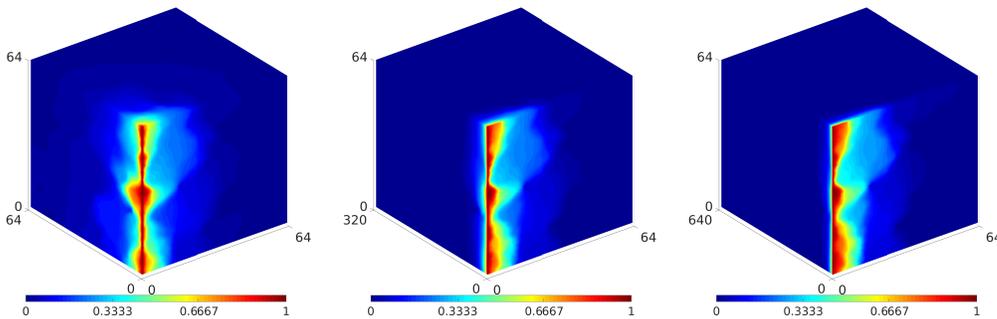


Fig. 16. Converged pressure solution for one of the realizations of permeability Sets 1 with grid aspect ratio $\alpha = 1, 5,$ and $10,$ respectively from left to right, after one time step $t^* = 0.4$. Dirichlet boundary conditions are set at the centers of two vertical sets of fine-scale grid cells: one from $(1,1,1)$ to $(1,1,64)$ and the other from $(64,64,1)$ to $(64,64,64)$ with the values of 1 and 0, respectively.

The performance of C-AMS[FE] and SAMG are presented in Fig. 17. In contrast to C-AMS[FE], the C-AMS[FV] (not shown) did not lead to 100% convergence success. However, for those C-AMS[FV] successful runs, similar CPU times as in C-AMS[FE] were observed.

Results shown in Fig. 17 are obtained with the C-AMS coarsening ratios of $8 \times 8 \times 8,$ $2 \times 8 \times 8,$ and $2 \times 8 \times 8$ for the cases of $\alpha = 1, 5,$ and $10,$ respectively. Note that as shown in Fig. 16, the anisotropic transmissibility (caused by stretched grid effect) would further motivate the use of enhanced coarse-grid geometries for C-AMS. Such a strategy is well developed in algebraic multigrid community, and is the subject of our future studies.

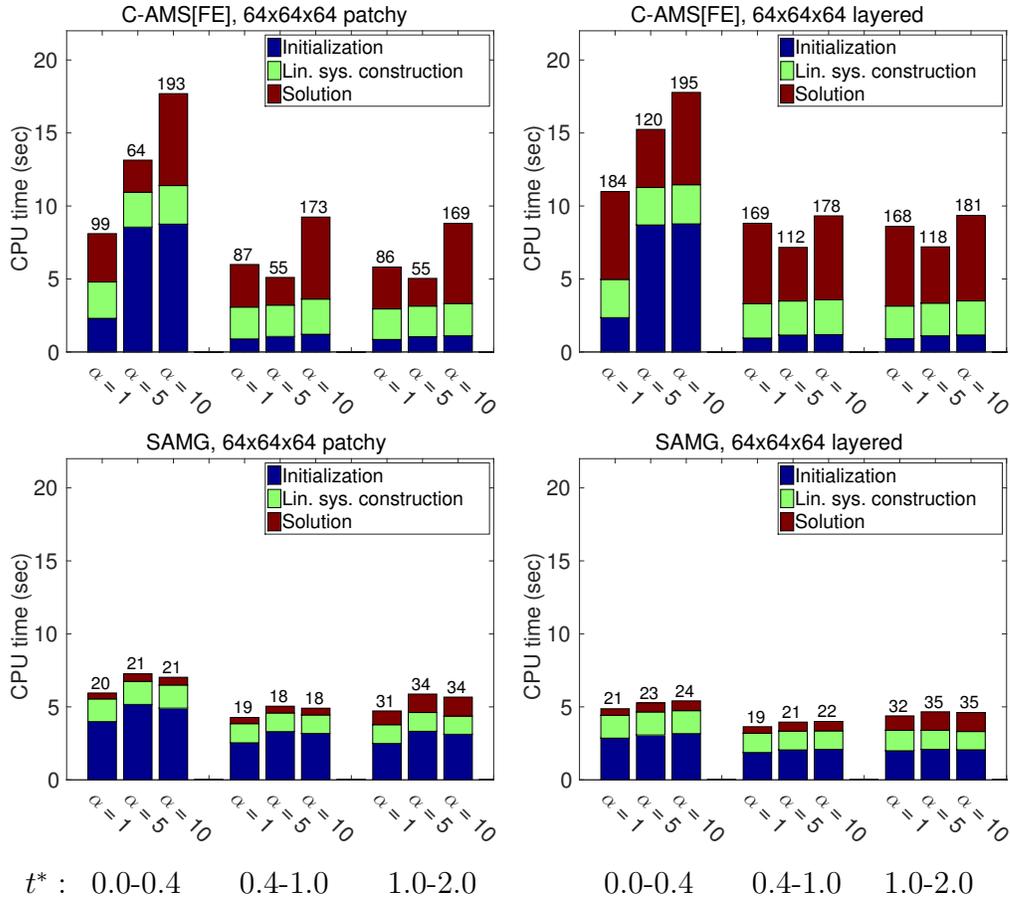


Fig. 17. Performance of C-AMS (top) and SAMG (bottom) for permeability Set 1 from Table 2 for different grid aspect ratios α in $\Delta x/\alpha = \Delta y = \Delta z$ for three successive time steps. Pressure solutions for the first time step is shown for one of the realizations in Fig. 16.

4.2.3 Test Case 3: effect of permeability contrast

To study the effect of permeability contrast, permeability Set 1 from Table 2 is considered with different $\ln(k)$ variances of $\sigma = 2, 4,$ and 8 . Note that the so-far studied cases were for variance 4, as described in Table 2. The settings are all the same as the default test cases, i.e., Dirichlet conditions are set at the east and west faces with no-flow condition everywhere else. Figure 18 illustrates the performances of C-AMS[FE] and SAMG for this test case. Note that the C-AMS requires more iterations when the permeability contrast is increased. To improve its performance, one can consider enriched multiscale strategies which are based on local spectral analysis [29], and modified permeability field (with less contrast) for calculation of basis functions [30]. Note that the success rates of C-AMS[FV] (not shown) were 90% (patchy, $\sigma = 2$), 95% (patchy, $\sigma = 8$) and 40% (layered, $\sigma = 8$). For the successful runs, the CPU times of C-AMS[FV] were comparable with C-AMS[FE].

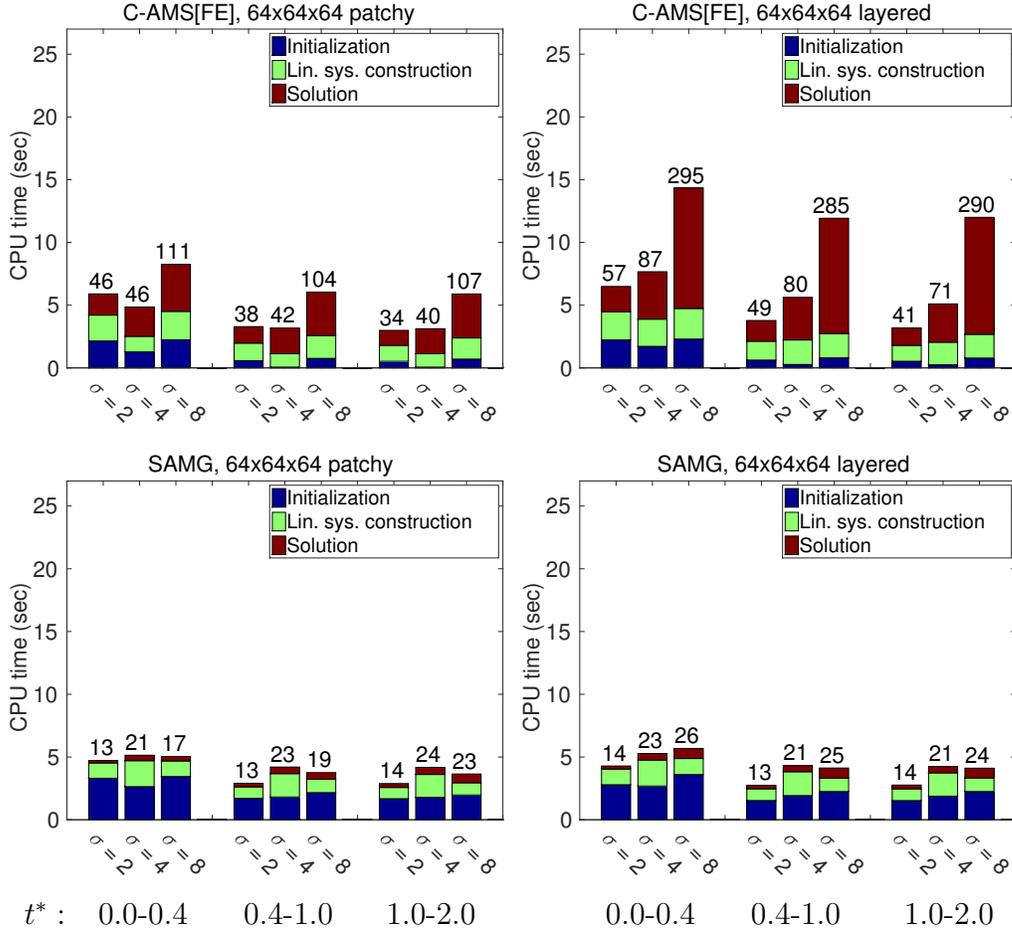


Fig. 18. Averaged CPU time comparison between C-AMS (top) and SAMG (bottom) for permeability Set 1 from Table 2 for different $\ln(k)$ variances of $\sigma = 2, 4$ and 5 .

5 Conclusions

Algebraic Multiscale Solver for Compressible flows (C-AMS) in heterogeneous porous media was introduced. Its algebraic formulation benefits from adaptivity, both in terms of the infrequent updating of the linearized system and from the selective update of the basis functions used to construct the prolongation operator.

Extensive numerical experiments on heterogeneous patchy and layered reservoirs revealed that the most efficient strategy is to use basis functions with incompressible advection terms, paired with 5 iterations of ILU(0) for post-smoothing.

Finally, several benchmark studies were presented, where the developed C-AMS research simulator was compared with an industrial-grade multigrid solver, i.e., SAMG. The results show that C-AMS is a competitive solver, especially in experiments that involve the simulation of a large number of time steps. The only drawback is the relatively high initialization time, which can be reduced by choosing an appropriate coarsening strategy or by running the basis function updates in parallel [31]. Moreover, due to its conservative property, C-AMS requires only a few iterations per time step to obtain a good quality approximation of the pressure solution for practical purposes. Systematic error estimate analyses for 3D multiphase simulations are a subject of ongoing research and, in addition, the C-AMS performance can be further extended by enrichment of the multiscale operators [29, 32, 20], and enriched coarse grid geometries on the basis of the underlying fine-scale transmissibility. Both are subjects of our future studies.

6 Acknowledgements

We would like to acknowledge the financial support from the Chevron/Schlumberger Intersect Alliance Technology and Schlumberger Petroleum Services CV during Matei Țene's scientific visit at TU Delft between November - February 2014. Since March 2014, Matei Țene is a PhD Research Assistant at TU Delft, sponsored by PI/ADNOC. The authors also thank Prof. Hamdi Tchelepi of Stanford University for the many helpful discussions.

I as

References

- [1] T. Y. Hou, X.-H. Wu, A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134 (1997) 169–189.
- [2] T. Y. Hou, X.-H. Wu, Z. Cai, Convergence of a multiscale finite element method for elliptic problems with rapidly oscillating coefficients. *Math. Comp.*, 68 (1999) 913–943.
- [3] Y. Efendiev, T. Y. Hou, Multiscale Finite Element Methods: Theory and Applications. Springer, 2009.
- [4] Y. Efendiev, V. Ginting, T. Hou, R. Ewing, Convergence of a nonconforming multiscale finite element method. *SIAM J. NUMER. ANAL.*, 37 (2000) 888–910.
- [5] J. Aarnes, T. Y. Hou, Multiscale domain decomposition methods for elliptic problems with high aspect ratios. *Acta Math. Appl.*, 18 (2002) 63–76.
- [6] 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.
- [7] 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.
- [8] V. Kippe, J. E. Aarnes, K.-A. Lie, A comparison of multiscale methods for elliptic problems in porous media flow. *Comput. Geosci.*, 12 (2008) 377–398.
- [9] U. Trottenberg, C.W. Oosterlee, A. Schueller, *Multigrid*. Elsevier Academic Press, 2001.
- [10] H. Zhou, H. A. Tchelepi. Operator based multiscale method for compressible flow. *SPE Journal*, 13 (2008) 267–273.
- [11] H. Zhou, H. A. Tchelepi, Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models. *SPE J.*, *SPE 141473-PA*, 17 (2012) 523–539.
- [12] I. Lunati, P. Jenny, Multiscale finite-volume method for density-driven flow in porous media. *Comput. Geosci.*, 12 (2008) 337–350.
- [13] S.H. Lee, H. Zhou, H. Techelpi, Adaptive multiscale finite-volume method for nonlinear multiphase transport in heterogeneous formations. *J. Comput. Phys.*, 228 (2009) 9036–9058.
- [14] H. Hajibeygi, G. Bonfigli, M.A. Hesse, P. Jenny, Iterative multiscale finite-volume method. *J. Comput. Phys.*, 227 (2008) 8604–8621.
- [15] H. Hajibeygi, D. Karvounis, P. Jenny, A hierarchical fracture model for the iterative multiscale finite volume method. *J. Comput. Phys.*, 230 (2011) 8729–8743.

- [16] H. Hajibeygi, S. H. Lee, I. Lunati, Accurate and efficient simulation of multiphase flow in a heterogeneous reservoir by using error estimate and control in the multiscale finite-volume framework. *SPE Journal*, 17 (2012) 1071–1083.
- [17] 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 (2006) 900–917.
- [18] H. Hajibeygi, H. A. Tchelepi, Compositional multiscale finite-volume formulation. *SPE Journal*, 19 (2014) 316–326.
- [19] Y. Wang, H. Hajibeygi, H. A. Tchelepi, Algebraic multiscale linear solver for heterogeneous elliptic problems. *Journal of Computational Physics*, 259 (2014) 284–303.
- [20] D. Cortinovis, Patrick Jenny, Iterative galerkin-enriched multiscale finite-volume method. *J. Comp. Phys.*, 277 (2014) 248–267.
- [21] O. Moyner, K.A. Lie, A multiscale two-point flux-approximation method. *J. Comput. Phys.*, 275 (2014) 273–293.
- [22] K. Aziz, A. Settari, *Petroleum Reservoir Simulation*. Blitzprint Ltd., Cagary, Alberta, 1979.
- [23] I. Lunati, P. Jenny, Multiscale finite-volume method for compressible multiphase flow in porous media. *J. Comput. Phys.*, 216 (2006) 616–636.
- [24] S. H. Lee, C. Wolfsteiner, H. A. Tchelepi, Multiscale finite-volume formulation for multiphase flow in porous media: black oil formulation of compressible, three-phase flow with gravity. *Comput. Geosci.*, 12 (2008) 351–366.
- [25] 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.
- [26] Y. Saad, *Iterative Methods for Sparse Linear Systems*. SIAM, Philadelphia, USA, 2003.
- [27] K. Stuben. *SAMG User’s Manual*. Fraunhofer Institute SCAI, 2010.
- [28] N. Remy, A. Boucher, J. Wu, *Applied geostatistics with SGeMS : a user’s guide*. Cambridge University Press, New York, 2009.
- [29] Y. Efendiev, J. Galvis, X.H. Wu, Multiscale finite element methods for high-contrast problems using local spectral basis functions. *J. Comput. Phys.*, 230 (2011) 937–955.
- [30] G. Bonfigli, P. Jenny, An efficient multi-scale poisson solver for the incompressible navier-stokes equations with immersed boundaries. *J. Comput. Phys.*, 228 (2009) 4568–4587.
- [31] A. M. Manea, J. Sewall, and H. A. Tchelepi, Parallel multiscale linear solver for highly detailed reservoir models. *Proceedings of SPE RSS*, (2015) 1–26, DOI: 10.2118/173259-MS.

- [32] V. Dolean, P. Jolivet, F. Nataf, N. Spillane, H. Xiang, Two-level domain decomposition methods for highly heterogeneous darcy equations. connections with multiscale methods. Oil & Gas Science and Technology - Revue d'IFP Energies nouvelles, DOI: 10.2516/ogst/2013206, 2014.