Mo A12

A Constrained Pressure Residual Multiscale (CPR-MS) Compositional Solver

M. Cusini* (Politecnico di Milano, Schlumberger), A.A. Lukyanov (Schlumberger), J. Natvig (Schlumberger) & H. Hajibeygi (Delft University of Technology)

SUMMARY

Unconventional Reservoir simulations involve several challenges not only arising from geological heterogeneities, but also from strong nonlinear physical coupling terms. All exiting upscaling and multiscale methods rely on a classical sequential formulation to treat the coupling between the nonlinear flow-transport equations. Unfortunately, the sequential strategies become severely inefficient when the flow and transport equations are strongly coupled. Examples of these cases include compositional displacements, and processes with strong capillarity effects. To extend the applicability of the multiscale methods for these challenging cases, in this paper, we propose a Constrained Pressure Residual Multiscale (CPR-MS) method. In the CPR-MS method, the CPR strategy is used to formulate the pressure equation, the approximate conservative solution of which is obtained by employing a few iterations of the iterative multiscale procedure. Several local- (ILU(k), BILU(k), etc.) and global-stage (Multiscale Finite Volume, MSFV, and Multiscale Finite Element, MSFE) solvers with different localization conditions (Linear BC, Reduced Problem BC, etc.) are employed in order to find an optimum strategy for the highly nonlinear compositional displacements. Numerical results for a wide range of test cases are presented, discussed and future studies are outlined.

Introduction

Mathematical formulation of multicomponent multiphase flow in heterogeneous large-scale porous media entails highly heterogeneous multiscale nonlinear and linear parameters. Development of efficient numerical methods, therefore, depends on correct treatments of both highly heterogeneous multiscale parameters and strongly nonlinear coupling terms. The former challenge has been widely addressed through several studies in the literature, among which the multiscale finite element (MsFEM) and finite volume Methods (MsFVM) cast a promising next-generation simulation framework (Hou and Wu, 1997; Jenny et al., 2003, 2006; Efendiev and Hou, 2009).
The MsFVM has been applied to a wide range of applications involving compressible and three-phase flows (Lunati and Jenny (2008); Zhou and Tchelepi (2007); Lee et al. (2008); Hajibeygi and Jenny (2009, 2011); Hajibeygi and Tchelepi (2014)), complex well configurations (Wolfsteiner et al. (2006); Jenny and Lunati (2009)), and fractures (Hajibeygi et al. (2011)). The method has been also applied for efficient transport solutions (Zhou et al. (2011)) and to reservoir models with unstructured grids (Moyner and Lie (2014)).

Recent advancements in the multiscale methods for reservoir simulation involve scalable conservative iterative solvers (namely, i-MSFV, TAMS, and AMS) for nonlinear and linear problems (Hajibeygi et al., 2008; Lunati et al., 2011; Zhou and Tchelepi, 2012; Hajibeygi and Jenny, 2009; Wang et al., 2014; Kunze et al., 2013; Cortinovis and Jenny, 2014). Note that only a few iterations are required to achieve accurate pressure solutions, used to solve transport equations, because conservative velocity fields can be constructed after any MSFV stage (Hajibeygi et al., 2012).

All multiscale methods listed above treat the coupling between flow and transport equations by adopting a sequential (IMPES- or sequential implicit- type) strategy (Spillette et al., 1973). Consequently, as with many similar multiscale and upscaling approaches, they are efficient when the coupling terms are not strong (Coats et al., 1974). Unfortunately, when the coupling terms between flow and transport equations grow in importance, sequential strategies are not efficient. Strong coupling terms exist, for example, in multiphase formulations with strong capillary and compositional effects. For these cases, fully implicit systems are generally more stable than sequential strategies (Younis, 2009; Cao, 2002). Therefore, there is an increasing demand in the reservoir simulation community to develop multiscale methods for fully implicit systems.

In this work, we devise the first multiscale (MS) method for fully implicit simulations of multiphase flow in highly heterogeneous porous media. Starting from the fully implicit Jacobian, a constrained pressure residual (CPR) preconditioning method (Wallis et al., 1985; Cao et al., 2005) is used to extract the pressure system. This pressure system is then coupled with a second-stage solver for the full residual correction, as is done in the classical CPR-based fully implicit solvers (Wallis et al., 1985; Cao et al., 2005).The innovative development of this work is that the MsFEM or MsFVM method is employed to obtain an approximate solution of the CPR-based pressure system. More precisely, the global solver is based on a multiscale coarse-scale system (either MsFVM or MsFEM), before and after which pres-moothing and post-smoothing steps are applied on the fine-scale solution to eliminate high-frequency errors. This multistage multiscale strategy combines the advantages of multiscale methods with the stability of fully implicit systems for highly coupled problems. As such, being integrated in the fully implicit iterative procedure, the CPR-MS captures the strong nonlinear coupling terms efficiently.

Numerical results, presented for heterogeneous three-dimensional cases, illustrate the efficiency of the CPR-based multiscale solver. The CPR-MS is an important development that puts the next-generation multiscale-based reservoir simulators into the context of strong nonlinear physics.

CPR-based Fully Implicit Reservoir Simulation

Mathematical formulation of multicomponent/multiphase fluid flow in hydrocarbons reservoirs leads to a nonlinear coupled system of partial differential equations (PDEs) (Aziz et al., 2005). To solve the strongly nonlinear and coupled system of discretized equations arising from the spatial and temporal discretization of the governing equations, a global linearization Newton-Raphson method is used (Aziz et al., 2005). This global linearization results in sparse, large, linear systems of equations, i.e.,

$$\mathbf{J}\Delta X = \begin{bmatrix} \mathbf{J}_{pp} & \mathbf{J}_{ps} \\ \mathbf{J}_{sp} & \mathbf{J}_{ss} \end{bmatrix} \begin{bmatrix} \Delta x_p \\ \Delta x_s \end{bmatrix} = r = \begin{bmatrix} r_p \\ r_s \end{bmatrix}, \tag{1}$$

where $\mathbf{J}_{pp}$ is the pressure block coefficients, $\mathbf{J}_{ss}$ is the nonpressure block (saturations or components' molar fractions) coefficients, $\mathbf{J}_{ps}$ and $\mathbf{J}_{sp}$ represent the respective coupling coefficients. Δx_p and Δx_s are the increments of pressure (primary variables) and of the other (secondary) variables (e.g., saturations and components' molar fractions). In many practical scenarios, the linear solver step represents a dominant part of the whole computational time. It is a well-known fact that performance of iterative linear solvers strongly depends on robust and fast preconditioners, amenable for massive parallelization. Additionally, for realistic-scale problems, the memory requirement to store the sparse Jacobian linear systems becomes an important issue.

Linear solvers generally used in reservoir simulations consist of preconditioned Krylov subspace methods such as ORTHOMIN with nested factorization as a preconditioner (Appleyard, 1983). A significant improvement in preconditioning strategy of linear systems arising from reservoir simulations was made by introducing the CPR method (Wallis, 1983; Wallis et al., 1985). The CPR preconditioner acknowledges the fact that Eq. 1 is of a mixed parabolic- (or elliptic-) hyperbolic type. By targeting the parabolic part of the system (or elliptic for incompressible flow) as a separate inner stage, the CPR method improves the convergence rate for the full linear system, based on a pressure predictor-corrector strategy for each linear step. The pressure equation is constructed by an IMPES-like reduction of Eq. 1 in which both sides are multiplied by a matrix of the form (i.e., Schur complement with the matrix $\mathbf{J}_{ss}$)

$$\mathbf{M} = \begin{bmatrix} \mathbf{I} & -\mathbf{Q} \\ \mathbf{0} & \mathbf{I} \end{bmatrix} \tag{2}$$

to obtain

$$\begin{bmatrix} \mathbf{J}^*_{pp} & \mathbf{J}^*_{ps} \\ \mathbf{J}_{sp} & \mathbf{J}_{ss} \end{bmatrix} \begin{bmatrix} \Delta x_p \\ \Delta x_s \end{bmatrix} = \begin{bmatrix} r^*_p \\ r_s \end{bmatrix} \tag{3}$$

where

$$\begin{aligned} \mathbf{J}^*_{pp} &= \mathbf{J}_{pp} - \mathbf{Q}\mathbf{J}_{sp}, \\ \mathbf{J}^*_{ps} &= \mathbf{J}_{ps} - \mathbf{Q}\mathbf{J}_{ss}, \\ r^*_p &= r_p - \mathbf{Q}r_s. \end{aligned}$$

To simplify the Schur complement procedure, it is assumed that

$$\mathbf{Q} = \mathbf{J}_{ps}\mathbf{J}^{-1}_{ss} \approx Q \cdot \mathbf{I} \quad \text{with} \quad Q = \text{colsum}(\mathbf{J}_{ps}) \cdot \text{colsum}(\mathbf{J}_{ss})^{-1} \tag{4}$$

where matrices $\mathbf{J}_{ps}$ and $\mathbf{J}_{ss}$ are replaced by vectors whose elements are the sums of each column. This implies that $\mathbf{J}^*_{ps}$ is approximately equal to zero, i.e.,

$$\mathbf{J}^*_{pp}\Delta x_p \approx r^*_p. \tag{5}$$

or that the pressure matrix can be extracted as follows:

$$\mathbf{J}^*_{pp} = \mathbf{C}^T\mathbf{M}\mathbf{J}\mathbf{C}, \quad \mathbf{C}^T = \begin{bmatrix} \mathbf{I} & \mathbf{0} \end{bmatrix}. \tag{6}$$

The effectiveness of CPR methods depends, to a large extent, on the quality of the pressure matrix $\mathbf{J}^*_{pp} = \mathbf{C}^T\mathbf{MJC}$. Several methods can be used to extract $\mathbf{J}^*_{pp}$, such as quasi-IMPES and true-IMPES. For the quasi-IMPES scheme, we ignore the off-diagonal $\mathbf{J}_{ps}$ terms and eliminate the diagonal part of $\mathbf{J}_{ps}$ using the inverse of the block-diagonal of $\mathbf{J}_{ss}$.

Equation (5) is the pressure equation. It forms an approximation of the parabolic (or elliptic for incompressible flow) part of the discrete operator and is considered separately from the remaining hyperbolic part.

The linear system (Eq. 1) is solved by GMRES (Saad and Schultz, 1986) preconditioned by the CPR method. The preconditioner consists of two complementary stages. The first stage is used to approximate the pressure solution by commonly applying algebraic multigrid (AMG) method to Eq. (5). In the second stage, another preconditioner (e.g., the ILU(0) method) is applied to the full system of Eq. (1) whereby the first-stage pressure approximation is used as the initial guess. The entire process for a given timestep is schematically illustrated in Figure 1.

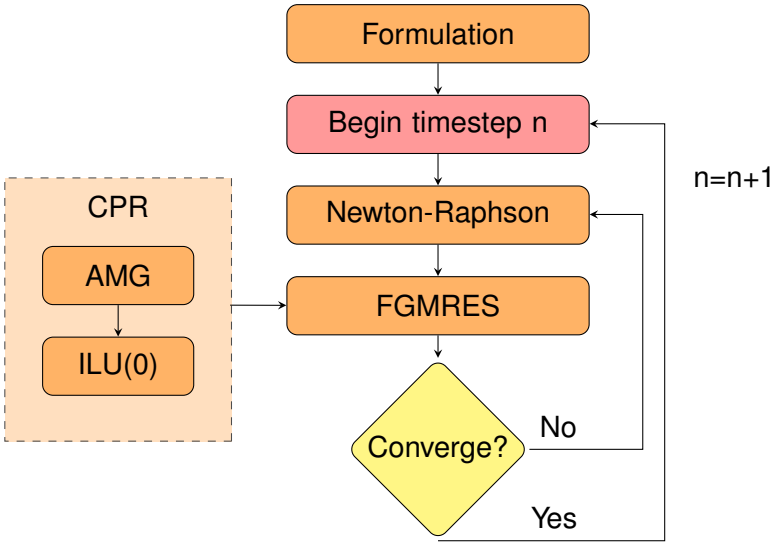

Figure 1 *Reservoir simulator framework: GMRES preconditioned by a two-stage CPR preconditioner.*

A linear solver based on CPR-AMG preconditioning is extremely effective in terms of algorithmic efficiency (Cao et al., 2005; Fung and Dogru, 2007). However, there are still challenges to overcome in implementing a near-ideal scalable AMG solver. Also, the second stage of CPR preconditioning often includes a variant of an incomplete LU factorization, which again is nontrivial to parallelize. As a result, the linear solver is still some way from near-ideal scalability. In the following section, the CPR-MS solver is described.

Multiscale Basis Functions

Analogously to finite element methods, multiscale methods use an approximation space that is spanned by basis functions in overlapping subdomains forming a *dual grid*. In the present paper, we use multiscale finite element basis functions to create an approximation space for pressure of much lower dimension than the original simulation grid. The approximation space is constructed using a discretization for pressure to capture the effect of fine-scale heterogeneity locally.

In the MsFEM and MsFVM methods, the discretization for pressure basis functions is obtained from sequential formulation. The pressure residual is formed as in the IMPES formulation by eliminating the unknown saturation from the mass balance equations. The pressure matrix is the Jacobian of the pressure residual. Although the IMPES pressure discretization differs from CPR, it has been used with success to precondition the fully implicit system (Watts, 1999).

Let the dual coarse grid (Figure 2) be a partition, $\mathscr{D} = \{D_1 \ldots, D_M\}$, of the main grid into M overlapping subdomains such that the overlap yields a discrete 3D mesh: each grid cell is classified as an *inner cell* if it is not part of the overlap between subdomains. The overlapping region between subdomains is partitioned into a set of *face* subdomains that form an overlap only between two partitions; the remaining cells (wirebasket) are partitioned into *edge* subdomains connected by *corner cells* (Jenny et al., 2003).

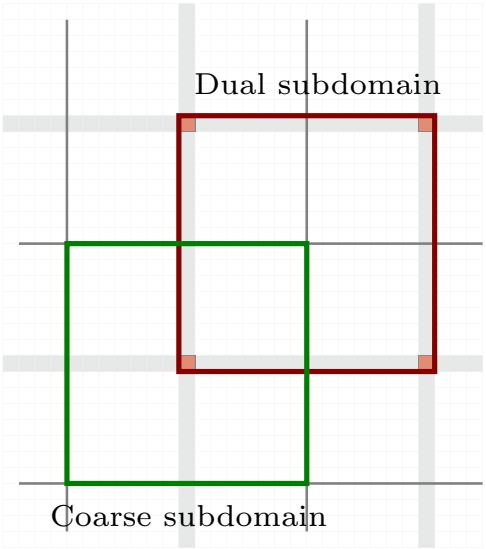

Figure 2 *For the MsFE basis functions, the simulation grid is partitioned into overlapping subdomains D_i (red) where the cells in the overlap region (gray) form a coarse mesh. The MsFV method uses a nonoverlapping partition (green).*

For each subdomain D_k we compute basis functions ϕ_i^k for each corner cell $i \in D_k$ with the properties that $\phi_{ij}^k = \delta_{ij}$ for corner cells j. Each basis function ϕ_i^k is a discrete solution to a homogeneous pressure equation in the interior of D_k, and solves a reduced pressure equation on the face and edge subdomains on the boundary of D_k (Hou and Wu, 1997). Alternatively, the basis function can be constructed using meshless multiscale methods (MS-MMPFA) (Lukyanov, 2010, 2012). The discrete subdomain basis functions created in this way are continuous across subdomain boundaries and, for a pure incompressible problem without any source term inside the subdomains, we have by construction

$$\sum_{i=1}^{n} \phi_i^k = 1 \quad \text{(where n is the number of corners in } D_k\text{)}, \tag{7}$$

for each subdomain $D_k \in \mathscr{D}$. Thus, the coarse-scale approximation space includes constant functions. In the following, we will use the multiscale basis functions as an interpolation operator. By collection

of all basis functions into a matrix $\mathbf{P}$, we obtain a linear operator that interpolates pressures x_p^c in the wirebasket corner cells to the whole grid:

$$x_p = \mathbf{P} x_p^c. \tag{8}$$

We will also need restriction operators $\mathbf{R}$ to form a linear system for the pressure in the wirebasket corner cells. Herein, we consider two different restriction operators. One is to use $\mathbf{R_{FE}} = \mathbf{P}^\mathsf{T}$. The application of $\mathbf{R_{FE}}$ to a vector x yields weighted sum of x for cells near each wirebasket corner cell. Applied to a linear system, it yields a classical Galerkin-type coarse-scale approximation. The other restriction operator we consider is a finite-volume type restriction $\mathbf{R_{FV}}$: For a nonoverlapping decomposition $\{\Omega_1, \ldots \Omega_N\}$ (Figure 2) of the simulation grid, dual to $\mathscr{D}$ in the sense that each subdomain Ω_i contains one (and only one) wirebasket corner cell, the $\mathbf{R_{FV}}$ is defined by

$$(\mathrm{R_{FV}})_{ij} = \begin{cases} 1 & \text{if cell } j \in \Omega_i, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Constrained Pressure Residual Multiscale (CPR-MS) Solver

The CPR-MS algorithm combines both steps of the CPR-AMG method (Cao et al., 2005) and of multiscale methods.

In fact, a fully implicit system is solved with a CPR-type two-stage preconditioners where an MS method is used for solving the pressure system extracted by the CPR decoupling operator.
Starting from the Jacobian matrix $\mathbf{J}$ built with a fully implicit formulation, the whole solution strategy can be summarized in six main steps:

1. Perform CPR preconditioning. The pressure matrix is extracted $\mathbf{A}_{cpr} = \mathbf{C}^T\mathbf{MJC}$ and the residual is restricted to a pressure residual $r_p = \mathbf{C}^T r$.
2. Construct the coarse pressure system using restriction ($\mathbf{R}$) and prolongation ($\mathbf{P}$) operators built with the basis functions computed either with a finite element or finite volume multiscale method: $\mathbf{A}_{cpr}^M = \mathbf{R}\mathbf{A}_{cpr}\mathbf{P}$ and $r_p^M = \mathbf{R} \cdot r_p$.
3. Solve the coarse pressure system and prolongate its solution to the fine scale: $\mathbf{A}_{cpr}^M \cdot \Delta x_p^c = r_p^M$ and $\Delta x_p = \mathbf{P} \cdot \Delta x_p^c$.
4. Correct the full system residual using the fine scale solution: $r^* = r - \mathbf{J} \cdot (\mathbf{C} \cdot \Delta x_p)$.
5. Solve the corrected full system with a second stage preconditioner: $\Delta x^* = \mathbf{M}^{-1} \cdot r^*$.
6. Combine the two solutions to build the full system solution: $\Delta x = \Delta x^* + \mathbf{C} \cdot \Delta x_p$.

Steps 1, 5, and 6 are exactly the same as those shown in the original CPR-type solver already implemented in a modern reservoir simulator. Steps 2 and 3 are typical of multiscale methods. Thus, the original two-stage preconditioning structure of the solution strategy shown in Cao et al. (2005) has not been changed, but a different choice has been made for the solution of the pressure system.

Basis functions, needed for the construction of MS restriction ($\mathbf{R}$) and prolongation ($\mathbf{P}$) operators, are computed at the beginning of each timestep and kept constant throughout the Newton loop. Updating them at the end of each Newton iteration may be useful when large timestep sizes are used.

The coarse system can be solved (step 3) with different strategies, and various were tried in order to find the optimal one. The choice of the solver for the coarse pressure system is strictly related to the

size of the coarse pressure matrix; this depends both on the size of the model and on the choice of the coarsening factors. If the size of the coarse system is small enough, a direct solver can be used; for big cases, to obtain a small, coarse pressure matrix, very large coarsening factors must be used. This may slow down the computation of basis functions as a direct solver is usually used for this step; to avoid this slow down, an iterative method should be used for the basis functions computation. The advantage of doing that is that basis functions calculation is only done once per each timestep whereas a different coarse system is solved for each Newton iteration. On the other hand, increasing the coarsening factors too much may affect the accuracy of our pressure solution, and the number of nonlinear iterations may grow.

Following the structure of a multigrid solver, pre- and post smoothing steps on the fine-scale pressure solution were introduced. The fine-scale pressure solution is smoothed using either Gauss Seidel (GS) or ILU(0) before and after step 3.

Examples

Tests were carried out on the test case SPE9; this is a 3-phase (water, gas, and oil) black oil case without any capillary effect. The fine grid is $144 \times 150 \times 30$, for a total of 648 000 cells (Figure 3).

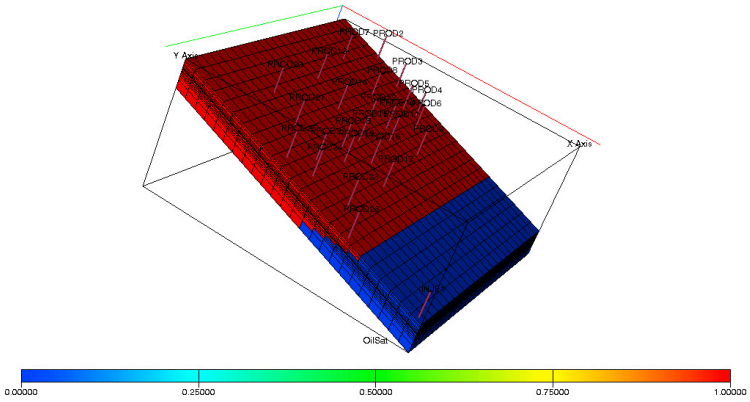

Figure 3 *Test case SPE9.*

Results and performance were compared with the current version of a modern reservoir simulator which uses CPR preconditioning with a 1 V-cycle AMG for solving the pressure system and ILU(0) as a second-stage preconditioner. Thus, the solution obtained with the current version of a modern reservoir simulator was used as a reference. The same second-stage preconditioner was used so that the only difference is the solution strategy of the pressure system. Two different values for the minimum timestep size were used which resulted in two completely different timestep selections throughout the simulation. The maximum timestep size was always kept the same.

All results presented were obtained in serial runs. The algorithm should be highly parallelizable and its potential should be even greater for parallel runs.

Results

First set of runs

For the first set of runs a minimum timestep size of 15 days was used. Finite element (FE) basis functions were used. The coarsening factors for the three directions were $(3,5,5)$, which resulted in a coarse grid of $48 \times 30 \times 6$, for a total of 8640 cells. With coarse pressure matrix of such a size, the direct methods

are very slow. For this reason, no results are presented for runs that use a direct solver on the coarse pressure system.

Table 1 *Settings of all runs for a minimum timestep size of* 15 *days*

Runs	Presmoothing	Post-smoothing	Coarse Solver
Run 1	GS	GS	AMG
Run 2	GS	GS	GMRES-AMG
Run 3	GS	GS	AMG
Run 4	none	GS	AMG

Before proceeding with the analysis of the performance in terms of number of linear iterations and CPU time of the CPR-MS solver, it is important to make sure that the solution does not change. All results were identical to the reference solution, as shown in Figure 4.

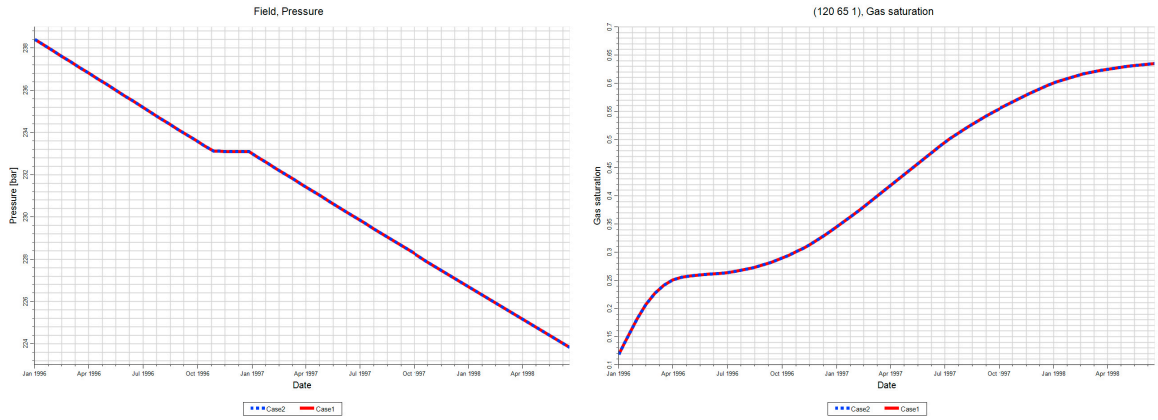

Figure 4 *Field pressure and gas block saturation solutions for CPR-AMG (Case 1) and CPR-MS (Case 2), showing that the two solutions match perfectly.*

Table 2 shows the results in terms of performance (number of iterations and CPU time) for the 4 runs and for the current modern reservoir simulator configuration (CPR-AMG).

Table 2 *Results of all runs for a minimum timestep size of 15 days*

	Number of iterations		CPU time (s)	
Runs	Nonlinear iter.	Linear iter.	Linear solver	Total
Run 1	369	2872	1615	2699
Run 2	369	2845	1119	2203
Run 3	371	3327	1233	2309
Run 4	371	3649	2026	3114
CPR-AMG	369	2187	1571	2537

We remark that the number of nonlinear iterations is almost constant. For run 3 and run 4 the heuristic functionality, which already exists in the simulator, was used; it keeps AMG restriction and prolongation operators constant for a certain number of iterations allowing a speed-up of the setup phase of the AMG solver.
The CPR-MS algorithm needs a greater number of linear iterations than the original CPR-AMG; this was actually expected as MS methods usually require a great number of linear iterations. Pre- and post-smoothing help limit the growth in the number of linear iterations.
Table 3 summarizes the relative difference in CPU time between each run and the reference CPR-AMG run.

Table 3 $\Delta\% = \frac{\text{Time}_{\text{run}} - \text{Time}_{\text{CPR-AMG}}}{\text{Time}_{\text{CPR-AMG}}} \cdot 100$

Runs	**Δ%**
Run 1	+ 6.38
Run 2	**-13.17**
Run 3	**-8.99**
Run 4	+22.74

Comparing the results of runs 3 and 4, we notice that introducing a presmoothing considerably reduces the number of linear iterations, and it allows a substantial speedup. Using a three-iteration GMRES cycle plus AMG is the most effective choice. If the correct settings are chosen, CPR-MS results are more than 10 % faster than CPR-AMG results.

Second set of runs

For the *second set of runs* a minimum timestep size of 1 day was used. This was the only difference between the two sets of runs; in fact, the same coarsening factors and FE restriction and prolongation operators were used. The heuristic function was used for runs 8 and 9.

Table 4 *Settings of all runs for a minimum timestep size of 1 day*

Runs	**presmoothing**	**Post-smoothing**	**Coarse Solver**
Run 6	GS	GS	GMRES-AMG
Run 7	none	GS	AMG
Run 8	none	GS	AMG
Run 9	GS	GS	AMG

Even for this set of runs, the CPR-MS solver solutions are perfectly matching with the ones provided by the reservoir simulator used as a reference (Table 5). As expected, the number of linear iterations for CPR-MS is much greater than the number for CPR-AMG.

Table 5 *Results of all runs for a minimum timestep size of 1 day*

	Number of iterations		**CPU time (s)**	
Runs	**Non linear iter.**	**Linear iter.**	**Linear Solver**	**total**
Run 6	711	4210	1718	3887
Run 7	709	4804	3301	5576
Run 8	708	5254	1911	4085
Run 9	709	4658	3212	5472
CPR-AMG	706	2868	2718	4666

Results for the smaller timestep size (Table 6) seem to confirm what observed for the larger one. Using presmoothing in this case does not seem to be as effective as in the previous one; in fact, by comparing runs 8 and 9, in which the only difference is the presence of the presmoother, we observe that using a presmoother slows down the simulation instead of speeding it up. A GMRES solver preconditioned by AMG is still the best choice for the coarse pressure system solver.

Table 6 $\Delta\% = \frac{\text{Time}_{\text{run}} - \text{Time}_{\text{CPR-AMG}}}{\text{Time}_{\text{CPR-AMG}}} \cdot 100$

Runs	**Δ%**
Run 6	**- 16.69**
Run 7	+19.5
Run 8	**-12.45**
Run 9	+17.27

Conclusions

Combining the fully implicit system with multiscale seems to present great potential; results obtained with single processor simulations are encouraging as the CPR-MS solver, in many cases, is faster than the current CPR-AMG solver. Multiprocessor simulations are expected to give even better results as computation of basis functions should be highly scalable.

Acknowledgments

This work was sponsored by Chevron/Schlumberger INTERSECT* Technology Alliance and the Schlumberger Abingdon Technology Centre. The authors gratefully thank the INTERSECT simulator Engine team and the Research & Prototyping team.

*Mark of Schlumberger; INTERSECT software is a joint product collaboration of Chevron, Total and Schlumberger.

References

Appleyard, J.R. [1983] Nested factorization. *SPE Reservoir Simulation Symposium, San Francisco, California*.
Aziz, K., Durlofsky, L. and Tchelepi, H. [2005] *Notes on Petroleum Reservoir Simulation*. Department of Petroleum Engineering, School of Earth Sciences, Stanford University, California, USA.
Cao, H. [2002] *Development of Techniques for General Purpose Simulation*. Phd thesis, Stanford University, USA.
Cao, H., Tchelepi, H.A., Wallis, J.R. and Yardumian, H. [2005] Constrained residual acceleration of conjugate residual methods. *SPE paper 96809, SPE Annual Technical Conference and Exhibition, Dallas, Texas, USA*, doi: doi:10.2118/96809-MS.
Coats, K., Chu, W.G.C. and Marcum, B. [1974] Three-dimensional simulation of steamflooding. *SPE J.*, **14**(6), 573–592.
Cortinovis, D. and Jenny, P. [2014] Iterative galerkin-enriched multiscale finite-volume method. *J. Comp. Phys., under review*.
Efendiev, Y. and Hou, T.Y. [2009] *Multiscale Finite Element Methods: Theory and Applications*. Springer.
Fung, L.S.K. and Dogru, A.H. [2007] Parallel unstructered solver methods for complex giant reservoir simulation. *SPE paper 106237, SPE Reservoir Simulation Symposium, Houston, Texas, USA*.
Hajibeygi, H., Bonfigli, G., Hesse, M. and Jenny, P. [2008] Iterative multiscale finite-volume method. *J. Comput. Phys.*, **227**, 8604–8621.
Hajibeygi, H. and Jenny, P. [2009] Multiscale finite-volume method for parabolic problems arising from compressible multiphase flow in porous media. *J. Comput. Phys.*, **228**, 5129–5147.
Hajibeygi, H. and Jenny, P. [2011] Adaptive iterative multiscale finite volume method. *J. Comput. Phys.*, **230**(3), 628–643.
Hajibeygi, H., Karvounis, D. and Jenny, P. [2011] A hierarchical fracture model for the iterative multiscale finite volume method. *J. Comput. Phys.*, **230**(24), 8729–8743.
Hajibeygi, H., Lee, S.H. and Lunati, I. [2012] 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**(4), 1071–1083.

Hajibeygi, H. and Tchelepi, H.A. [2014] Compositional multiscale finite-volume formulation. *SPE Journal*, **19**(2), 316–326.
Hou, T.Y. and Wu, X.H. [1997] A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, **134**, 169–189.
Jenny, P., Lee, S.H. and Tchelepi, H.A. [2003] Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *J. Comput. Phys.*, **187**, 47–67.
Jenny, P., Lee, S.H. and Tchelepi, H.A. [2006] Adaptive fully implicit multi-scale finite-volume method for multiphase flow and transport in heterogeneous porous media. *J. Comput. Phys.*, **217**, 627–641.
Jenny, P. and Lunati, I. [2009] Modeling complex wells with the multi-scale finite volume method. *J. Comput. Phys.*, **228**, 687–702.
Kunze, R., Lunati, I. and Lee, S.H. [2013] A multilevel multiscale finite-volume method. *J. Comput. Phys.*, **225**, 502–520.
Lee, S.H., Wolfsteiner, C. and Tchelepi, H.A. [2008] Multiscale finite-volume formulation for multiphase flow in porous media: black oil formulation of compressible, three-phase flow with gravity. *Comput. Geosci.*, **12**(3), 351–366.
Lukyanov, A.A. [2010] Meshless upscaling method and its application to a fluid flow in porous media. *ECMOR XII. Oxford. UK*.
Lukyanov, A.A. [2012] Adaptive fully implicit multi-scale meshless multi-point flux method for fluid flow in heterogeneous porous media. *ECMOR XIII. Biarritz. France*.
Lunati, I. and Jenny, P. [2008] Multiscale finite-volume method for density-driven flow in porous media. *Comput. Geosci.*, **12**(3), 337–350.
Lunati, I., Lee, S. and Tyagi, M. [2011] An iterative multiscale finite volume algorithm converging to exact solution. *J. of Comp. Phys.*, **230**(5), 1849–1864.
Moyner, O. and Lie, K.A. [2014] The multiscale finite-volume method on stratigraphic grids. *SPE Journal*, **in press**, doi:http://dx.doi.org/10.2118/163649-PA.
Saad, Y. and Schultz, M. [1986] Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, **7**(3), 856–869.
Spillette, A.G., Hillestad, J.G. and Stone, H.L. [1973] A high-stability sequential solution approach to reservoir simulation. *SPE J.*, sPE 4542.
Wallis, J.R., Kendall, R.P., Little, T.E. and Nolen, J.S. [1985] Constrained residual acceleration of conjugate residual methods. *SPE Reservoir Simulation Symposium*, **doi:10.2118/13536-MS**.
Wallis, J. [1983] Incomplete gaussian elimination as a preconditioning for generalized conjugate gradient acceleration. *SPE Reservoir Simulation Symposium*, doi:doi:10.2118/12265-MS.
Wang, Y., Hajibeygi, H. and Tchelepi, H.A. [2014] Algebraic multiscale linear solver for heterogeneous elliptic problems. *Journal of Computational Physics*, **259**, 284–303.
Watts, J.W. [1999] A total velocity sequential preconditioner for solving implicit reservoir simulation matrix equations. *SPE paper 51909, SPE Reservoir Simulation Symposium, Houston, Texas, USA*.
Wolfsteiner, C., Lee, S.H. and Tchelepi, H.A. [2006] Well modeling in the multiscale finite volume method for subsurface flow simulation. *SIAM Multiscale Model. Simul.*, **5**(3), 900–917.
Younis, R. [2009] *Advances in Modern Computational Methods for Nonlinear Problems; A Generic Efficient Automatic Differentiation Framework, and Nonlinear Solvers That Converge All The Time*. Phd thesis, Stanford University, USA.
Zhou, H., Lee, S. and Tchelepi, H. [2011] Multiscale finite-volume formulation for saturation equations. *SPE J.*, **17**(1), 198–211.
Zhou, H. and Tchelepi, H.A. [2007] Operator based multiscale method for compressible flow. *SPE 106254 presented at the SPE Reservoir Simulation Symposium, Houston, TX, USA*.
Zhou, H. and Tchelepi, H.A. [2012] Two-stage algebraic multiscale linear solver for highly heterogeneous reservoir models. *SPE J., SPE 141473-PA*, **17**(2), 523–539.