SPE-173231-MS

Efficient Preconditioning for Algebraic Multigrid and Red-Black Ordering in Adaptive-Implicit Black-Oil Simulations

Geoffrey L. Brown and David A. Collins, CMG; Zhangxin Chen, University of Calgary

This paper was prepared for presentation at the SPE Reservoir Simulation Symposium held in Houston, Texas, USA, 23–25 February 2015.

Abstract

Adaptive implicit petroleum reservoir simulations result in huge, often very ill-conditioned linear systems of equations. The full system contains characteristics of both hyperbolic and nearly elliptic sub-systems. Traditional single stage solvers, such as variable degree ILU, are often not efficient at converging nearly elliptic, such as pressure, components. Therefore, multi-stage preconditioning methods, such as the constrained pressure residual (CPR) method, are a popular approach to “divide and conquer” coupled systems. An algebraic multigrid (AMG) method provides a technique to efficiently solve suitably extracted, nearly elliptic sub-systems as a first stage of CPR. An ILU technique can then be used as a second stage.

The primary objective of an efficient two-stage preconditioning strategy consists of extracting nearly elliptic sub-systems that are suitable for an efficient AMG solution while simultaneously ensuring a fast overall convergence of the two-stage approach. This research aims to modify and apply the dynamic row sum preconditioner (DRS) to adaptive-implicit black oil simulation. This extends the original DRS preconditioner (Gries, et al., 2014), which was developed in consideration of individual matrices.

Physical weighting is incorporated into the DRS technique using additional information determined during Jacobian building. The adaptive-implicit formulation produces explicit saturation terms that are fully coupled to pressure from the same location. The DRS method is modified to approximately decouple such saturations so that they may be excluded from the iterative matrix solution.

The multi-coloured ordering of variables is also considered. Such methods have proved efficient by reducing the number of unknowns from the iterative matrix solution. A parallel algorithm using domain decomposition and red-black ordering for the two-stage approach is examined. The effect of increasing the degree of ILU on the boundaries between domains selectively is investigated for the two-stage approach and a single-stage approach. The results of full simulation show significant acceleration using the preconditioned two-stage approach.

Introduction

One of the most computationally intensive tasks of reservoir simulation is the repeated solution of large linear systems resulting from the discretization and linearization of the equations describing fluid flow in porous media. A key difficulty is that the governing equations exhibit mixed parabolic-hyperbolic

character (Aziz and Settari, 1979; Trangenstein and Bell, 1989). In reservoir simulation, the pressure field is essentially parabolic or nearly elliptic with long-range coupling, while the remaining equations (referred to as saturation equations) are hyperbolic or transport-dominated parabolic (Peaceman, 1977; Aziz and Settari, 1979; Trangenstein and Bell, 1989). For fully implicit (FI) time discretization the pressure field of the system is updated simultaneously with the saturation fields. This produces linear systems that are highly coupled. The coupling of pressure and saturation is reduced for IMPES (implicit pressure, explicit saturation) time discretization and adaptive-implicit (AIM) formulations (Forsyth and Sammon, 1986) but is not eliminated. In these formulations saturations remain coupled to pressure locally. Traditional solution techniques, such as incomplete LU factorization (ILU) (Behie and Forsyth, 1984) and nested factorization (Appleyard and Cheshire, 1983) preconditioners of Krylov subspace methods, have been popular because of their robustness. However, they are not the most efficient methods for converging parabolic or elliptic problems. The Constrained Pressure Residual (CPR) method (Wallis, 1983; Wallis, et al., 1985) can achieve improved convergence by targeting the parabolic portion of a linear system as a separate inner stage in a two-stage approach. A two-stage method exploits the fact that ILU techniques are well suited for hyperbolic problems and multilevel techniques, in particular algebraic multigrid (AMG) solvers, are most efficient at solving elliptic and nearly elliptic problems (Ruge and Stüben, 1987; Cleary, et al., 2000; Stüben, 2001). Using AMG as part of a two-stage CPR preconditioner is a common approach in reservoir simulation (Lacroix, et al., 2001; Scheichl, et al., 2003; Cao, et al., 2005; Stüben, et al., 2007).

To efficiently apply a two-stage technique a left preconditioner is often used as a preprocessing step to weaken the coupling between pressure and saturation. Popular decoupling operators include alternate block factorization (ABF) (Bank, et al., 1989) and quasi-IMPES (Lacroix, et al., 2001; Scheichl, et al., 2003). The primary goal of the decoupling processes is to ensure a fast overall convergence rate of the CPR method by reducing the impact of the outer global stage on the inner pressure stage. Indeed, decoupling preconditioners can also be applied as a global left preconditioner for a single-stage method to improve convergence rates. Decoupling preconditioners can aid the convergence of the inner pressure solution as well. They mix all the governing equations of a grid cell ensuring the components most affected by pressure changes are included in the inner stage of CPR. An ideal decoupling preconditioner preserves the algebraic properties of the pressure system for AMG. However, decoupling changes the scaling of equations at different locations in the grid possibly destroying diagonal-dominance properties or producing a strongly non-symmetric pressure system that is not well suited for solution by AMG (Scheichl, et al., 2003). AMG was originally designed to solve linear systems that correspond to discretized elliptic partial differential equations. However, the properties of the pressure fields considered in reservoir simulations are influenced by the physics of porous media flow. In particular, upstream weighting of flow terms may locally modify the properties of nearly elliptic problems to such an extent that the convergence of AMG is adversely affected (Stüben, et al., 2007; Clees and Ganzer, 2010). Gries et al. (2014) developed a dynamic row-sum preconditioner (DRS) to help avoid such problems for AMG solvers in the CPR context. The DRS method scans each matrix rows and determines if it should be included in a simple summation during the creation of a pressure matrix. The DRS preconditioner is of quasi-IMPES type and is based on a model formulation where equations are dimensioned in terms of reservoir volumes. The model formulation described in Gries et al. (2014) uses equations in terms of surface volumes, which could be a potential source of some non-optimal results. Additionally, the DRS preconditioner does not completely decouple IMPES saturation variables and is, therefore, not ideal for AIM time discretization.

The purpose of this work is to modify the DRS preconditioning strategy so that it may be used within reservoir simulators with a wider variety of model formulations and time discretizations. Physical weighting of the rows in the row summation of DRS using additional information determined during Jacobian building is used to apply the preconditioner to a model formulation with equations in terms of surface volumes. The ABF preconditioner is incorporated with a DRS strategy for the application to

IMPES discretization within the adaptive-implicit formulation. The flexibility afforded by the inclusion of the ABF preconditioner allows multi-colored ordering schemes for variable reduction.

The paper is organized as follows. First, a brief introduction to the black-oil model under consideration is given followed by a description of the linearization of the model and the creation of the linear system. The next section briefly summarizes the ILU, AMG, CPR, and DRS preconditioning techniques used in this study. Modifications to the DRS preconditioner to extend its applicability are developed. The testing environment for numerical experiments is described along with a short description of the industrial test cases investigated. Numerical results for simulation runs are discussed demonstrating the performance benefits of a modified DRS preconditioner. The effect of red-black ordering and variable reduction via a Schur compliment is explored as is selectively modifying the degree of fill-in for ILU in both the CPR method and as a stand-alone preconditioner. Conclusions and a discussion of ongoing work are provided in the final section.

Model Description

For this study a black-oil model is considered consisting of conservation equations for three components: oil, water, and gas in three phases. The simplifying assumptions of this model are that the phase velocities are given by Darcy's law, the water phase consists entirely of water, oil is present only in the oil phase, and gas can be present in both the gas and oil phases. Following Forsyth & Sammon (1986) the conservation equations for the oil, water, and gas components are:

$$\frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right) + q_o - \nabla \cdot [\boldsymbol{T}_o \nabla \Phi_o] = 0$$

$$\frac{\partial}{\partial t}\left(\frac{\phi S_w}{B_w}\right) + q_w - \nabla \cdot [\boldsymbol{T}_w \nabla \Phi_w] = 0$$

$$\underbrace{\frac{\partial}{\partial t}\left[\phi\left(\frac{S_g}{B_g} + \frac{R_s S_o}{B_o}\right)\right]}_{\text{accumulation term}} + \underbrace{q_g + R_s q_o}_{\text{source term}} - \underbrace{\nabla \cdot \left[\boldsymbol{T}_g \nabla \Phi_g + R_s \boldsymbol{T}_o \nabla \Phi_o\right]}_{\text{flow term}} = 0$$

In the above the transmissibility of phase α is given by $\boldsymbol{T}_\alpha = \left(\frac{k_{r\alpha}}{\mu_\alpha B_\alpha}\boldsymbol{k}\right)$ and the potential of phase α is given by $\nabla\Phi_\alpha = \nabla p_\alpha - \rho_\alpha \bar{\boldsymbol{g}}$. The derivatives with respect to time in the above equations are referred to as accumulation terms, the spatial derivatives as flow terms, and the others as source terms. The reservoir fluid is assumed to fill the pore volume giving the saturation constraint $S_o + S_w + S_g = 1$. The oil pressure is chosen to be the system pressure with the other phase pressures related through capillary pressure terms. The system can be in two states, either under-saturated, when no free gas exists ($S_g = 0$), or saturated, when the pressure of the oil is above the bubble point pressure ($p > p_b$) Well constraint equations are included in the sytem. They contain only one variable, bottom-hole pressure.

These equations are discretized into a finite difference form using a control volume approach, with upstream weighting for the transmissibility terms. For each finite difference cell (grid cell) three primary variables are chosen. In the saturated state oil pressure, water saturation, and oil saturation (p, S_w, S_o) are primary variables and in the undersaturated state oil pressure, water saturation, and bubble point pressure (p, S_w, p_b) are primary variables. The discretization of the accumulation term only relates primary variables of a single grid cell to each other. The discretization of the flow terms relates variables in adjacent grid cells. The equations are discretized in time using an adaptive implicit (AIM) scheme where each grid cell is determined to be either fully implicit (FI) or IMPES (implicit pressure, explicit saturation) (Forsyth and Sammon, 1986). The variables in the flow term are discretized using either the first order implicit backward Euler method or the explicit forward method. The accumulation term is always discretized using the implicit backward Euler method. In the IMPES method the oil pressure is updated implicitly everywhere, while the saturations (or saturation pressures) are updated explicitly in the flow

terms. For the FI method all primary variables are updated implicitly. Thus, saturation variables from IMPES grid cells are only connected to variables from the same grid cell at the update time for accumulation terms. The resulting discrete equations are highly non-linear and stiff. A Newton iteration is used to linearize the problem leading to a series of matrix problems of the type $\boldsymbol{A} \cdot \boldsymbol{\delta x} = \boldsymbol{r}$ where $\boldsymbol{A}$ is the Jacobian matrix, $\boldsymbol{x}$ is a vector consisting of the primary variables for each grid cell which define the current state of the reservoir, and $\boldsymbol{r}$ is the residual vector. These linear systems are continually solved for an update to the primary variables $\boldsymbol{\delta x}$ until the residuals are sufficiently small. The Jacobian matrix can be considered to be a system of block-matrices for the material balance equations of grid cell i with respect to the primary variables of grid cell j. In particular a block matrix is given by

$$\{\boldsymbol{A}\}_{ij} = \begin{bmatrix} A_{o,p} & A_{o,S_w} & A_{o,s} \\ A_{w,p} & A_{w,S_w} & A_{w,s} \\ A_{g,p} & A_{g,S_w} & A_{g,s} \end{bmatrix}$$

where $\{A_{\alpha,\beta}\}_{i,j}$ is the partial derivative of the phase (α) equation of grid cell i with respect to the primary variable β of grid cell j (note, $s = S_o$ or p_b depending on whether grid cell j is saturated).

Preconditioners

Linearization of the discretized model equations via Newton's method results in very large linear systems. The resulting systems are sparse, non-symmetric, and ill-conditioned and, in general, require the majority of the computation time in reservoir simulators. Consequently, iterative linear solvers employing Krylov methods that depend on effective preconditioners are the conventional approach used to accelerate the solution. The Generalized Minimal Residual method (GMRES) and its flexible variant (FGMRES) (Saad, 2003) are the most popular Krylov methods and are used in this work. The performance of a linear solver depends more on the choice of a preconditioner than the particular Krylov method used. One traditional preconditioner is the incomplete LU factorization (ILU) (Behie and Forsyth, 1984; Behie, et al., 1984; Saad, 2003). The particular ILU method of this investigation is variable degree ILU(k) (allowing for k levels of fill-in) and utilizes domain decomposition using vertex separators (Collins, et al., 2003). This differs from most traditional parallel ILU methods which utilize domain decomposition using edge separators and apply a block-Jacobi method to connect domains. The reservoir is partitioned into disjoint sets of grid cells and wells (classes). The classes are further organized into levels where there must be no connection between grid cells of different classes at the same level. Classes at the lowest level make up the majority of the reservoir. Classes at the next level (level-two) are in effect planes of grid cells that separate domains (level-one classes). Classes at higher levels contain grid cells (or well constraint equations) that connect classes at the same level. The degree of ILU can vary for intra-class terms and inter-class terms as well as by level.

The reservoir system is known to have mixed hyperbolic and parabolic (or nearly elliptic) character (Trangenstein and Bell, 1989). Traditional single stage-solvers, such as the above-described ILU, are often not efficient at converging elliptic or nearly elliptic components. This gives rise to a combinative multi-stage method (Behie and Vinsome, 1982) where estimates of the solution of different variable types are found separately as well as an estimate of all unknowns of the whole system. These results are then combined prior to the typical Krylov acceleration process. If the system can be decoupled based on the characteristics of the system more efficient solvers can be applied to each constituent part. In particular, Algebraic Multigrid (AMG) is known to be particularly effective at solving nearly elliptic systems. Detailed information on the AMG technique can be found in the literature (Ruge and Stüben, 1987; Stüben, 2001). AMG computes its hierarchy of grids automatically in a setup-stage before starting the iterative solution process. To compute the setup AMG mimics geometric grids by interpreting the matrix as a graph and distinguishes between strong and weak couplings. Based on this graph, a maximal independent set of nodes is computed taking the particular matrix properties into account. The resulting

set of nodes defines the coarse level using the Galerkin principle (Ruge and Stüben, 1987; Stüben, 2001). The interpolation operators between different levels are defined by computing weights again based on the matrix entries. In the solution stage of AMG the efficient interplay of simple relaxation schemes is used as smoothers and coarse grid corrections based on the hierarchy of levels are exploited.

A variant of the Combinative method, known as the Constrained Pressure Residual (CPR) method, is a two-stage method (Wallis, 1983; Wallis, et al., 1985) that is widely used in reservoir simulation. In CPR, the pressure system is first formed by approximately decoupling pressure from saturation. The remaining couplings of saturation to pressure are neglected and the system is restricted to pressure variable coefficients only. The pressure system is then solved (approximately for CPR like variants) by a suitable technique, usually AMG. This is the first or inner stage of the two-stage preconditioner. The full system residual is updated to correct for the estimate of pressure and then preconditioned using a global preconditioner, in this case the above described ILU method. This is the second or outer stage. Finally, the first and second stage solutions are combined.

The first step of CPR is to approximately decouple the pressure and saturation variables. One such technique is the alternate block factorization (Bank, et al., 1989) used, for example, by (Cao, et al., 2005). In ABF each row of block matrices of the Jacobian matrix is multiplied on the left by the inverse of the block diagonal matrix, $\{\boldsymbol{A}\}_{ii}^{-1}$. This allows the full decopling of primary variables locally, within a grid cell. However, this preconditioner also normalizes the resulting matrix which makes it unsuitable for AMG. The decoupling is crucial for IMPES grid cells where the saturation variables are only coupled locally through the accumulation term. Decoupling them from the rest of the Jacobian allows them to be updated directly at the end of the linear solution. ABF can also be used in conjunction with a Schur complement method to reduce the matrix size using a red-black ordering scheme. Grid cells on the boundaries of each class labeled black and all grid cells that are only connected to black-grid cells are labeled red. Note that wells are included in the red-black scheme. The block rows associated with red cells can then be removed from the iterative matrix solution using a Schur complement. The reordering reduces the matrix size by increasing the matrix complexity. Another popular decoupling method is the quasi-IMPES (Scheichl, et al., 2003) where pressures and saturations are approximately decoupled without the normalization of the matrix.

A dynamic row-sum (DRS) preconditioner has been developed by Gries et al. (2014) based on the approach described by Scheichl et al. (2003) in which pressure derivatives from each equation in a block-row of the Jacobian matrix are summed prior to extraction with CPR. Consequently, a pressure derivative approximating one from a total fluid pressure is generated. The purpose is to include the pressure related behavior from all three phases within each grid cell. The DRS preconditioner modifies this to dynamically select which equation to include in the sum. If the matrix entries from any of the phase equations of a particular grid cell appear to be problematic for solution by AMG they are skipped. In particular, if an equation produces matrix entries which are far from diagonally dominant they are not likely to be solved well using the AMG method and are excluded from the sum. The block sub-matrices of the pressure matrix to be solved with AMG are given as

$$\left\{\widetilde{A_p}\right\}_{i,j} = \boldsymbol{\delta}_i^T \cdot \left\{\boldsymbol{A}_{\alpha,p}\right\}_{i,j} \cdot \boldsymbol{e}_1$$

where $\boldsymbol{\delta}_i^T = \left(\delta_o, \delta_w, \delta_g\right)_i$ is the dynamically chosen summation vector for each grid cell. For each phase α of each block i the term $\delta_{\alpha,i}$ is computed as:

$$\delta_{\alpha,i} = \begin{cases} 0 & \text{if } \dfrac{\left|\left\{A_{\alpha,p}\right\}_{i,i}\right|}{\sum_{j\neq i}^{n_p}\left|\left\{A_{\alpha,p}\right\}_{i,j}\right|} < \varepsilon_{dd} \\ 1 & \text{otherwise} \end{cases}$$

The value of $\delta_{\alpha,i}$ determines the inclusion of derivatives of phase α in the sum. Other conditions are placed on the δ_{α} terms but are of less importance (Scheichl et al. 2003).

Weighted Dynamic Row-Sum

The DRS preconditioner of Gries et al. (2014) applies an un-weighted sum of rows just as in Scheichl et al. (2003). However, in Scheichl et al. (2003) the underlying model equations are in dimensions of reservoir volume whereas in Gries et al. (2014) and in this research they are in dimensions of surface volume. For Scheichl et al. (2003) the summation of matrix entries approximates the elimination of saturation from the governing equations. In this research an un-weighted summation of terms will not eliminate saturation from the governing equations due to the formation volume factors B_{α}. Reservoir volume dimensioned equations are in the canonical form and their summation approximates the behavior of an averaged single-phase fluid at reservoir conditions. To approximate a single-phase fluid for surface volume based equations, physical weights of each equation are necessary. The weighted summation will approximately eliminate saturations from the accumulation term. The weighting parameters are determined at the time of Jacobian building from the reservoir conditions at the end of the previous Newton iteration. The weights required for this formulation include formation volume factors and the solution gas/oil ratio. The new weighted DRS vector becomes:

$$\boldsymbol{\xi} = \left(\delta_o\left[B_o - R_s B_g\right], \delta_w B_w, \delta_g B_g\right)^T$$

This weighting will include pressure behavior from each phase while approximately decoupling pressure from saturation. The weighted DRS method is not applied in the same manner to both FI grid cells and IMPES grid cells. For FI cells the weighted DRS method is applied just as in the original DRS method with no ABF component for pressure aligned rows. For IMPES cells the desired goal is to completely decouple local saturations from pressure to remove them from the matrix calculation. To maintain the same scaling throughout the matrix IMPES diagonal blocks are first multiplied on the left by the weighted DRS vector. The resulting pressure $\{\widetilde{A_p}\}_{i,i}$ value is then used as a scalar multiplier of the pressure aligned row after ABF is applied. The rows aligned with saturation variables are chosen to be decoupled with ABF in both the FI and IMPES cases to ensure numerical stability of the ILU preconditioner in the outer stage of CPR. Assuming that pressure is aligned with the first row of each block-matrix, the resulting weighted DRS preconditioner is a diagonal block matrix $\boldsymbol{G}$, with block diagonals given by:

$$\boldsymbol{G}_{ii}^{\mathrm{FI}} = [\boldsymbol{e}_1\boldsymbol{\xi}_i^T + (\boldsymbol{I} - \boldsymbol{e}_1\boldsymbol{e}_1^T)\{\boldsymbol{A}\}_{ii}^{-1}]$$

$$\boldsymbol{G}_{ii}^{\mathrm{IMPES}} = [\boldsymbol{e}_1\boldsymbol{\xi}_i^T\{\boldsymbol{A}\}_{ii}\boldsymbol{e}_1\boldsymbol{e}_1^T + (\boldsymbol{I} - \boldsymbol{e}_1\boldsymbol{e}_1^T)]\{\boldsymbol{A}\}_{ii}^{-1}$$

The well equations are included in the matrix solution. They contain only one variable, bottom-hole pressure. To weight the well terms they are multiplied by a scalar term that equals the sum of the total reservoir volume of fluid from the source terms in the model equations.

Simulation Data

CMG's IMEX simulator (black-oil) is used to examine the potential acceleration of convergence using the newly developed weighted DRS preconditioner. A comparison of the total simulation (clock) time for several test cases is made to a simulation using a single-stage global ILU preconditioner. The reference method used for comparison is CMG's sparse linear solver, PARASOL, with OpenMP parallelization. Its solution method is GMRES preconditioned first by ABF and then by the above described variable degree ILU.

The weighted DRS method has been implemented within IMEX as a preconditioner for a CPR method. First, the weighted DRS preconditioner is applied to decouple the Jacobian matrix. Next, the SAMG

software package, developed by Fraunhofer SCAI, is used as the first stage of CPR to estimate a solution for pressure. The ILU method of PARASOL is used as the second stage global preconditioner. Finally, the solutions of both stages are combined and used as the iterative solution in the right preconditioned FGMRES method (Saad, 2003). An un-weighted DRS method has also been implemented. It is identical to the weighted DRS method except no weights are applied to the row summations. The un-weighted DRS can be considered a proxy for the original DRS method except that it allows AIM formulations. The number of AMG cycles and the number of ILU steps performed in the first and second preconditioning stages, respectively, is crucial for the overall efficiency. A loose relative residual tolerance for the pressure solution was chosen for each test case leading to a small number of AMG cycles per outer iteration.

The methods above are compared for several test cases described below.

MX521×469: This case consists of 90 five-spot patterns in 10 × 9 areal arrangement divided into 20 vertical layers. It has 521 × 469 × 20 = 4.88 million cells aligned on a Cartesian grid, all active. It is a three-phase (oil/water/gas) water injection problem and is simulated for 15 years. It has 200 wells. The permeabilities and porosities are constant. This simulation is designed so that free gas appears and then increases throughout the simulation.

MX1041×261: This case consists of 100 five spot patterns in 20 × 5 areal arrangement divided into 20 vertical layers. It has 1041 × 261 × 20 = 5.43 million cells aligned on a Cartesian grid, all active. It is a three-phase (oil/water/gas) water injection problem and is simulated for 15 years. It has 226 wells. The five-spot patterns are identical to those of MX521×469.

SPE10: This case is based on the fine grid case for Model 2 from the Tenth SPE Comparative Solution Project (Christie and Blunt, 2001). It has 60 × 220 × 85 = 1.12 million cells, with 1.09 million active. It consists of a single five spot, with a geostatistically generated permeability distribution. It is a two-phase (oil/water) water-injection problem. It is run for 2,000 days simulated time.

TM2T: This case is a fine grid version of client data characterized by an irregular region. Each cell from the original client data set was split equally, areally into 3×3×1 cells. It has 369 × 1200 × 44 = 1.95 million cells with 1.64 million active. It is a three-phase (oil/water/gas) primary-production problem. It is run for 3,440 days simulated time and contains eleven production wells.

SINC: This case is based on client data. It has 275 × 235 × 133 = 8.60 million cells, with 2.64 million active. It is a three-phase (oil/water/gas) water-injection problem. It is run for 37 years of simulated time and contains 1,003 horizontal wells.

FLUV: This case is based on client data. It has 142 × 123 × 56 = 978 thousand cells, with 449 thousand active. It is a three-phase (oil/water/gas) primary-production problem. It is run for 37 years of simulated time and contains 1,326 wells.

Results are presented for cases run on a Dell Power Edge M820 Blade Server with two 2.7 GHz Xeon E5-2697 v2 12 core processors and 128GB of 1866MHz DDR3 RAM. All simulations were run using all 24 cores under Red Hat Enterprise Linux 6 update 5. For all simulation runs unless otherwise stated Jacobian matrices were constructed using the adaptive-implicit technique, generalized red-black ordering was used and the numerical parameters for IMEX were held constant for runs with both preconditioning methods. Each solution is split into 24 level-one solver classes (essentially domains) to fully utilize every core for parallel runs. For red-black ordering cases, ILU(0) was used for intra-class terms and ILU(1) for inter-class terms. For naturally ordered cases, ILU(1) was used for intra-class terms and ILU(2) for inter-class terms.

For modified DRS and CPR preconditioned runs, the OpenMP parallelized SAMG was essentially used with its default parameters with only the following changes. Aggressive coarsening was used to construct each level during the setup phase. The full setup phase was performed every time step with only a partial setup done (where Galerkin operators are recalculated) for subsequent Newton iterations. SAMG's built-in sparse direct solver was used for the coarsest level and automated checking of the input matrix

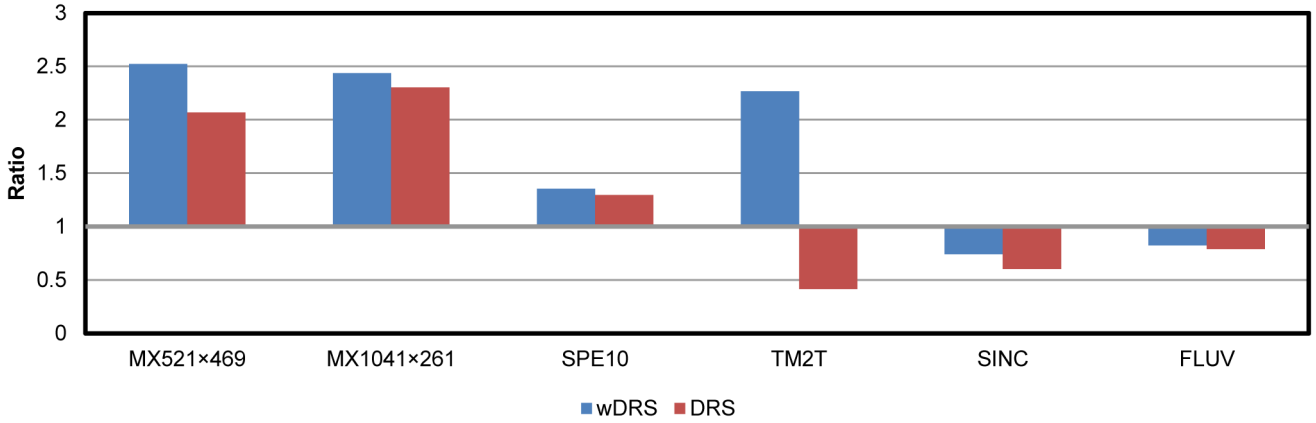

Figure 1—The ratio of the total clock time of simulation for 6 test cases. The weighted DRS and un-weighted DRS + CPR preconditioned methods are compared to the ABF + ILU preconditioned single-stage method. The ratios are given by (ABF+ILU time) / (wDRS+CPR time) and (ABF+ILU time) / (DRS+CPR time).

and residuals was turned off. As discussed above, a case dependent residual reduction between 0.7 and 0.9 was desired for the pressure solution with a maximum of 5 AMG cycles enforced.

Results

Simulations of the six test cases were run using both the weighted DRS (wDRS) and un-weighted DRS with CPR preconditioned solver and a single-stage ABF and ILU preconditioned solver. The total running time of simulation is used to compare the methods since it is of primary concern. Although only the linear solver differs in the methods, minimal drift in the system from, for example, convergence rates, time step size selection, and adaptive-implicit switching can cause other portions of the simulator to take more or less clock time. The accuracy of results of the methods was confirmed by comparing average reservoir pressures and cumulative-production values of oil, water, and gas at the end of the simulation. An estimate of cumulative reservoir fluid production is within 1% for the SPE10, FLUV, and SINC cases and within 0.1% for the others. The average reservoir pressure is within 2% for the FLUV and SINC cases and 0.01% for the others.

The ratio of running times for the preconditioning methods using adaptive-implicit time discretization and red-black ordering of grid cells is shown in Fig. 1. The clock time of the ABF and ILU method is used as the numerator in the ratios meaning that values above one show a speedup in running time and values below one show a longer running time with a two-stage preconditioner.

Improvement in the overall running time by using the weighted DRS method is seen in four of the six test cases. In three of the cases the running time is more than twice as fast. In the SINC and FLUV cases the overall running time is slower using the weighted DRS method. In particular, FLUV took 22% more time and SINC took 35% more time. The weighted DRS preconditioner outperforms the un-weighted DRS preconditioner in all cases. In particular, the TM2T case is 5.5× faster meaning it is now faster than the single-stage method. To further explore the cause of potential speedups and slowdowns convergence rates are examined. The convergence criteria in both solution methods is the same so the primary source of speedup is a reduction in the number of solver-iterations needed to converge each matrix. The average number of solver-iterations for each Newton-iteration (and hence each individual Jacobian matrix) is shown in Fig. 2.

The decrease in solver-iterations using the weighted DRS method in each test case is drastic. No test case took more than an average of 6.1 solver iterations when using the weighted DRS method. It is important to note that the computational cost of a solver-iteration of the two methods is not the same. In the traditional single-stage method a solver-iteration consists only of applying the ILU preconditioner

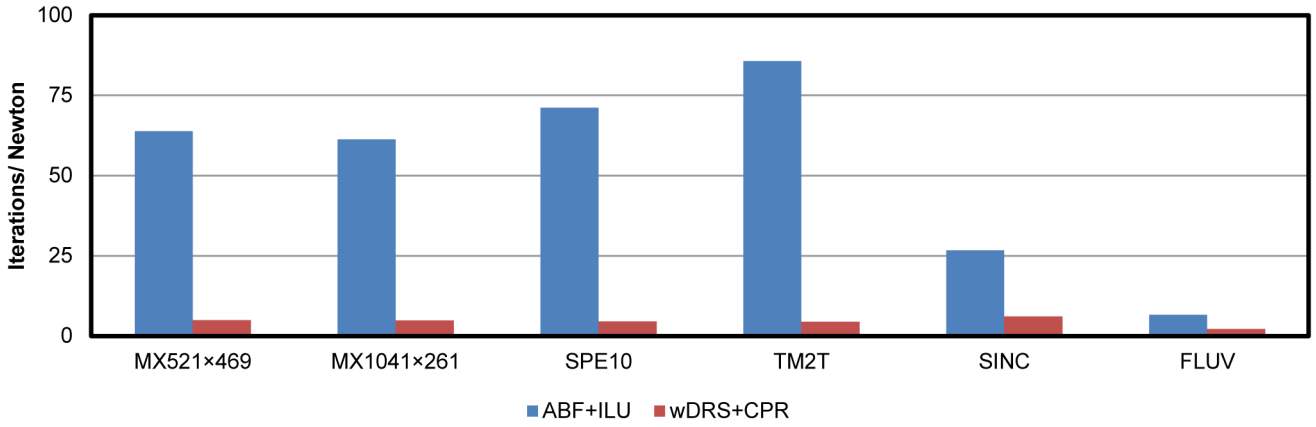

Figure 2—The average number of solver iterations per Newton iteration (and hence matrix) for 6 test cases using both the weighted DRS (wDRS) and CPR preconditioned method and the ABF and ILU preconditioned single-stage method. The combined two-stage iteration is counted as a single iteration for the CPR method.

within the GMRES acceleration. For the weighted DRS and CPR method the AMG setup stage is performed for each matrix. Some time savings can be gained by only performing partial AMG setups where the hierarchy of levels is unchanged and only the Galerkin transfer operations are recalculated. Partial AMG setups are used in these simulations. For each solver-iteration a solution for the pressure is calculated using AMG cycles. Although the use of AMG adds the majority of additional time per solver-iteration there are some additional computational costs. The weighted DRS preconditioner itself needs to be constructed requiring an extra scan of the Jacobian matrix prior to solution and each solver-iteration includes an update to the residual, requiring an additional matrix multiplication, and the combination of the two solutions. Clearly, to achieve a reduction in total running time by improving the convergence rates requires a substantial reduction in average solver iterations. The test cases SINC and FLUV already have the lowest average solver-iterations using the single-stage method, 26.7 for SINC and 6.5 for FLUV. Essentially, improvement in convergence does not compensate for the overhead required when using the weighted DRS method in these two cases.

The modified DRS method performs a full decoupling of saturation variables from IMPES grid cells so that they may be excluded from the iterative matrix solution. This allows the DRS method to be applied in the case of adaptive-implicit time discretization. Additionally, variables from entire grid cells can be excluded from the iterative matrix solution with the use of red-black ordering and the elimination of red labeled grid cells from the matrix. The results for the six test cases for simulations run with adaptive-implicit (AI) time discretization with both natural (NA) and red-black (RB) ordering schemes are shown in Fig. 3. The plot is of the total clock time normalized by the adaptive-implicit, red-black (AI+RB) run.

In each test case using a red-black ordering scheme reduced the total running time by up to as much as 22% (MX1041×261 case). Using the adaptive-implicit formulation with red-black ordering has the additional benefit of requiring less storage for the matrix.

The ILU preconditioner is flexible in the choice of fill-in degree. The standard method for red-black ordering cases uses ILU(0) for matrix terms connecting grid cells within the same class and ILU(1) for terms connecting two classes. These connections between classes are connections between domains (level-one classes) and separator planes (level-two classes) or connections to higher level classes. The increase in fill-in degree on the boundaries helps improve the convergence for parallel runs where there is significant flow across the boundary. To further improve the solution along boundaries a method using a higher degree ILU along boundaries (referred to as RBB) is used. The RBB method uses the Parallel Degree Red Black Boundary (PDEGRBB) option in IMEX. Red-black ordering is implemented so that class boundary grid cells are all labeled black so that red cells may be pre-eliminated in a simple fashion

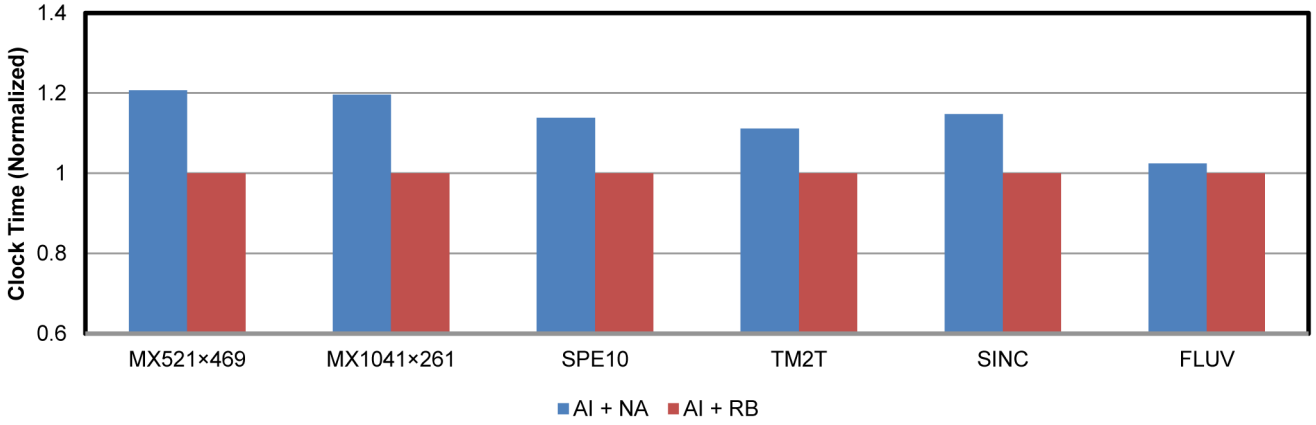

Figure 3—The total clock time for 6 test cases run with the weighted DRS and CPR preconditioners. Each test case is run using both a natural (NA) order scheme and a red-black (RB) ordering scheme solving only for black variables with the iterative solver. The total simulation times are normalized using the time of the adaptive implicit with red-black ordering run (AI + RB).

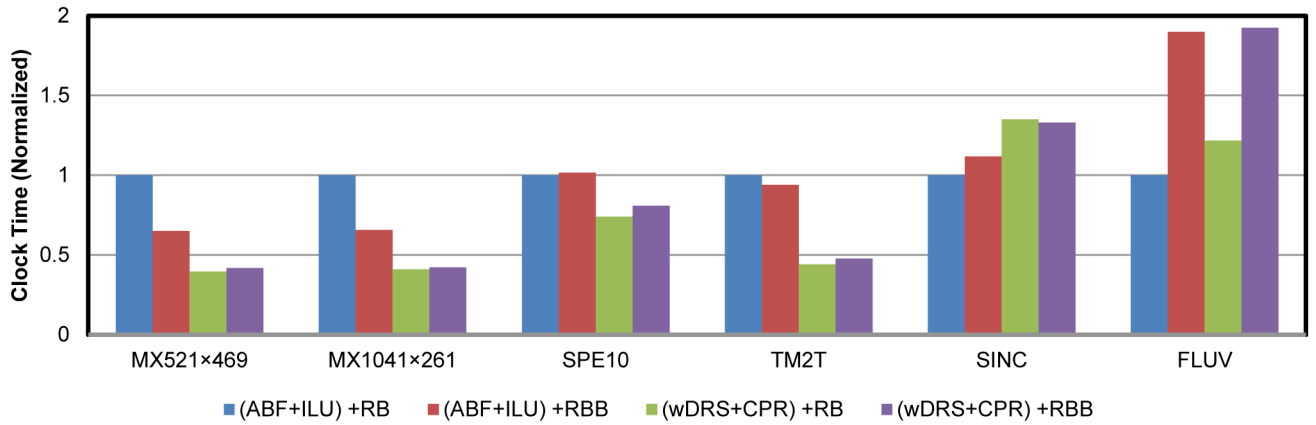

Figure 4—The total clock time of simulation for six test cases run with the single-stage (ABF+ILU) preconditioned method and the two-stage (wDRS+CPR) preconditioned method. The methods are run using red-black ordering with the standard ILU degree choices (RB) and a higher degree ILU for entries on the boundary (RBB).

in parallel. This means that boundary cells for the lowest level class and all cells in higher level classes are black. Therefore, the boundary regions are similar to a natural ordering case. Thus, as done for natural ordering (Gries et al., 2014), inter-class terms are increased from ILU(0) to ILU(2) and intra-class terms involving at least one cell on a class boundary are increased from ILU(0) to ILU(1) for the RBB method.

The plot in Fig. 4 shows clock times for the six test cases run with both the default (RB) choice of ILU degree (Gries et al., 2014) and the increased degree method (RBB) in both the ABF and ILU preconditioned single-stage solver and the weighted DRS and CPR preconditioned solver. Note that the CPR method used the variable degree ILU as the second stage preconditioner. The clock times are normalized using the default choice of ILU degree for the ABF and ILU preconditioned single-stage simulation running time.

In the two pattern flood cases (MX521×469 and MX1041×261) and TM2T the simulation time is reduced using the RBB set of degrees for the single-stage method. In the case of the pattern floods this is partially due to the symmetric nature of the repeated patterns. The resulting matrix entries maintain a certain symmetry that is more easily solved using higher degree ILU decomposition. In the other cases the computational cost of increasing the degree of ILU, even selectively, either does not significantly reduce the simulation time or in the case of FLUV, which has the fewest number of planes per level-one class, takes much longer. The average number of solver-iterations for each Newton-iteration in these simulations is shown in Fig. 5.

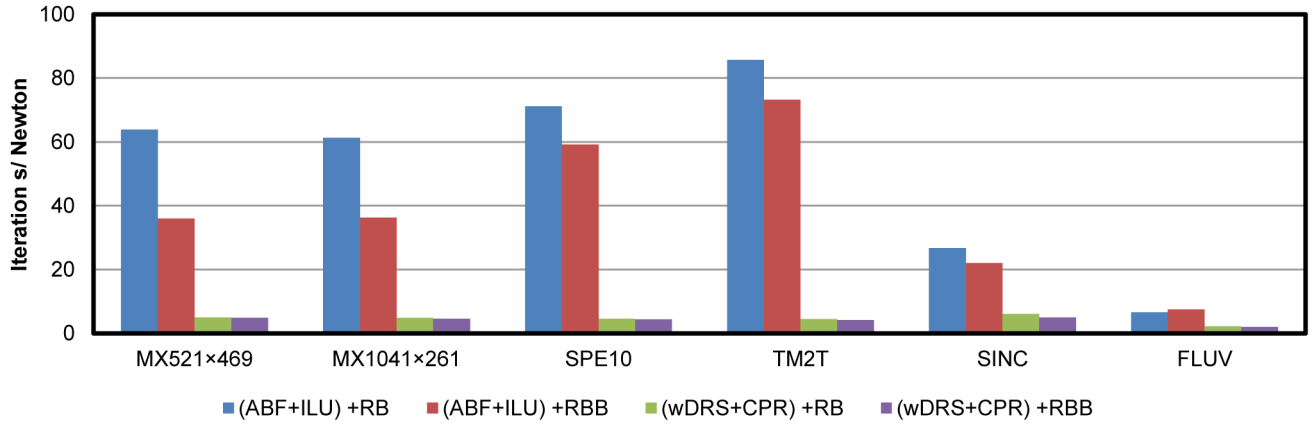

Figure 5—The average number of solver iterations per Newton iteration for 6 test cases run with the single-stage (ABF+ILU) preconditioned method and the two-stage (wDRS+CPR) preconditioned method. The methods are run using red-black ordering with the standard ILU degree choices (RB) and a higher degree ILU for entries on the boundary (RBB). The combined two-stage iteration is counted as a single iteration for the CPR method.

Only the two pattern flood cases show a significant reduction in the average number of solver-iterations per Newton-iteration. In both cases more than 25 fewer solver-iterations (or a reduction of over 40%) are required for the same convergence. Again, this is likely due to favorable elimination of matrix entries at higher orders due to the symmetry of the systems. In the case of FLUV the average number of solver-iterations is low even for the RB choice of degree for the ILU only case and is not further reduced for RBB cases leading to an increase in simulation time. Since FLUV converges so quickly using the simplest method, the additions of higher degree ILU or the weighted DRS preconditioner are not beneficial.

Conclusions

Common approaches used to precondition linear systems solved with a CPR method aim to approximately decouple pressure and saturation. In some cases decoupling can produce a pressure matrix with properties that are not well suited to solution using AMG. The dynamic row-sum preconditioner aims to construct a pressure matrix that is well suited to AMG and at the same time accelerating the CPR process. A modification to the DRS preconditioner has been developed in this work which allows for the application to a wider variety of model formulations. In particular, the new weighted DRS method can be applied in situations where the model equations are dimensioned in terms of surface volumes rather than reservoir volumes. Additionally, IMPES and AIM time discretizations can be used with a modified DRS preconditioner.

The use of the weighted DRS preconditioner as a preprocessor to a CPR method leads to a substantial reduction in simulation time from single-stage solvers in many cases. The weighted DRS preconditioner outperforms the un-weighted DRS preconditioner for surface-volume based formulations. The time reduction is due to a large decrease in solver-iterations per matrix solution required for convergence. Varying the degree of fill for an ILU preconditioner selectively can further improve the convergence rates of certain cases. This is the case for single-stage solvers as well as the CPR preconditioned solver.

Further research will focus on improving the simulation time in all cases by only selectively applying the DRS preconditioner. During the scan of a Jacobian matrix for the construction of the DRS preconditioner, if the matrix is determined to have a fraction of rows where the coupling between saturation and pressure exceeds a certain threshold the two-stage preconditioner is not used for that matrix. Default threshold levels are being determined. The weighted DRS preconditioner is being extended to more complex reservoir simulation models. In particular, thermal models will be examined.

Acknowledgements

The authors would like to thank Sebastian Gries and Klaus Stüben of Fraunhofer SCAI for their support with the SAMG software and their help with the development of the DRS preconditioner.

This work has been performed as part of Geoffrey Brown's ongoing PhD project at the University of Calgary with the support of the Reservoir Simulation Group.

Nomenclature

B	= Formation volume factor
e_1	= Unit vector $(1,0,0)^T$
$\bar{g}$	= Gravity vector
I	= Identity matrix
k	= Absolute permeability vector
k_r	= Relative permeability
p	= Pressure
q	= Source (well) term
R_s	= Solution gas/oil ratio
S	= Saturation
t	= Time
T	= Transmissibility
μ	= Viscosity
ρ	= Density
ϕ	= Porosity
Φ	= Potential

Subscripts

g	= Gas phase
i	= Grid block number
j	= Grid block neighbor
o	= Oil phase
w	= Water phase

References

Appleyard, J.R., and Cheshire, I.M. 1983. Nested Factorization. In SPE Reservoir Simulation Symposium, 15-18 November, San Francisco, California, 315–324. SPE–12264–MS. http://dx.doi.org/10.2118/12264-MS.

Aziz, K., and Settari, A. 1979. *Petroleum Reservoir Simulation*. Applied Science Publishers.

Bank, R.E., Chan, T.F., Coughran, W.M., et alet al. 1989. The Alternate-Block-Factorization Procedure for Systems of Partial Differential Equations. *BIT* **29** (4): 938–54. http://dx.doi.org/10.1007/BF01932753.

Behie, A., Collins, D.A., and Forsyth, P.A. 1984. Incomplete Factorization Methods for Three-Dimensional Non-Symmetric Problems. *Computer Methods in Applied Mechanics and Engineering* **42** (3): 287–99. http://dx.doi.org/10.1016/0045-7825(84)90010-0.

Behie, A., and Forsyth, P.A. 1984. Incomplete Factorization Methods for Fully Implicit Simulation of Enhanced Oil Recovery. *SIAM Journal on Scientific and Statistical Computing* **5** (3): 543–61. http://dx.doi.org/10.1137/0905040.

Behie, A., and Vinsome, P.K.W. 1982. Block Iterative Methods for Fully Implicit Reservoir Simulation. *SOC. PET. ENGRS. J.* **22** (5): 658–68.

Cao, H., Tchelepi, H.A., Wallis, J.R., et alet al. 2005. Parallel Scalable Unstructured CPR-Type Linear Solver for Reservoir Simulation. In SPE Annual Technical Conference and Exhibition, 9-12 October, Dallas, Texas, SPE–96809–MS. http://dx.doi.org/10.2118/96809-MS.

Christie, M.A., and Blunt, M.J. 2001. Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques. *SPE Reservoir Evaluation & Engineering* **4** (04): 308–317. SPE–72469–PA. http://dx.doi.org/10.2118/72469-PA.

Cleary, A.J., Falgout, R.D., Henson, V.E., et alet al. 2000. Robustness and Scalability of Algebraic Multigrid. *SIAM Journal on Scientific Computing* **21** (5): 1886–1908.

Clees, T., and Ganzer, L. 2010. An Efficient Algebraic Multigrid Solver Strategy for Adaptive Implicit Methods in Oil-Reservoir Simulation. *SPE Journal* **15** (3): 670–81. SPE–105789–PA. http://dx.doi.org/10.2118/105789-PA.

Collins, D.A., Grabenstetter, J.E., and Sammon, P.H. 2003. A Shared-Memory Parallel Black-Oil Simulator with a Parallel ILU Linear Solver. In SPE Reservoir Simulation Symposium, 3-5 February, Houston, Texas, SPE–79713–MS. http://dx.doi.org/10.2118/79713-MS.

Forsyth, P.A., and Sammon, P.H. 1986. Practical Considerations for Adaptive Implicit Methods in Reservoir Simulation. *Journal of Computational Physics* **62** (2): 265–81. http://dx.doi.org/10.1016/0021-9991(86)90127-0.

Gries, S., Stüben, K., Brown, G.L., et alet al. 2014. Preconditioning for Efficiently Applying Algebraic Multigrid in Fully Implicit Reservoir Simulations. *SPE Journal* **19** (04): 726–736. SPE–163608–PA. http://dx.doi.org/10.2118/163608-PA.

Lacroix, S., Vassilevski, Y.V., and Wheeler, M.F. 2001. Decoupling Preconditioners in the Implicit Parallel Accurate Reservoir Simulator (IPARS). *Numerical Linear Algebra with Applications* **8** (8): 537–49. http://dx.doi.org/10.1002/nla.264.

Peaceman, D.W. 1977. *Fundamentals of Numerical Reservoir Simulation*. Developments in Petroleum Science. Elsevier.

Ruge, J.W., and Stüben, K. 1987. Algebraic Multigrid. In *Multigrid Methods,* edited by SF McCormick, **3**:73–130. Frontiers in Applied Mathematics. Society for Industrial and Applied Mathematics. http://dx.doi.org/doi:10.1137/1.9781611971057.ch4.

Saad, Y. 2003. *Iterative Methods for Sparse Linear Systems*. Society for Industrial and Applied Mathematics. http://dx.doi.org/10.2113/gsjfr.6.1.30.

Scheichl, R., Masson, R., and Wendebourg, J. 2003. Decoupling and Block Preconditioning for Sedimentary Basin Simulations. *Computational Geosciences* **7** (4): 295–318. http://dx.doi.org/10.1023/B:COMG.0000005244.61636.4e.

Stüben, K. 2001. A Review of Algebraic Multigrid. *Journal of Computational and Applied Mathematics* **128** (1-2): 281–309.

Stüben, K., Clees, T., Klie, H., et alet al. 2007. Algebraic Multigrid Methods (AMG) for the Efficient Solution of Fully Implicit Formulations in Reservoir Simulation. In SPE Reservoir Simulation Symposium, 26-28 February, Houston, Texas, SPE–105832–MS. http://dx.doi.org/10.2118/105832-MS.

Trangenstein, J.A., and Bell, J.B. 1989. Mathematical Structure of the Black-Oil Model for Petroleum Reservoir Simulation. *SIAM Journal on Applied Mathematics* **49** (3): 749–83. http://dx.doi.org/10.1137/0149044.

Wallis, J.R. 1983. Incomplete Gaussian Elimination as a Preconditioning for Generalized Conjugate Gradient Acceleration. In SPE Reservoir Simulation Symposium, 15-18 November, San Francisco, California, SPE–12265–MS. http://dx.doi.org/10.2118/12265-MS.

Wallis, J.R., Kendall, R.P., and Little, T.E. 1985. Constrained Residual Acceleration of Conjugate Residual Methods. In SPE Reservoir Simulation Symposium, 10-13 February, Dallas, Texas, SPE–13536–MS. Society of Petroleum Engineers. http://dx.doi.org/10.2118/13536-MS.