SPE 163608

Preconditioning for Efficiently Applying Algebraic Multigrid in Fully Implicit Reservoir Simulations

Sebastian Gries and Klaus Stüben, SPE, Fraunhofer Institute for Algorithms and Scientific Computing SCAI;
Geoffrey L. Brown, Dingjun Chen and David A. Collins, SPE, Computer Modelling Group Ltd.

This paper was prepared for presentation at the SPE Reservoir Simulation Symposium held in The Woodlands, Texas USA, 18–20 February 2013.

Abstract

Fully implicit petroleum reservoir simulations result in huge, often very ill-conditioned linear systems of equations to solve for different unknowns, for example, pressure and saturations. It is well known that the full system matrix contains both hyperbolic as well as nearly elliptic sub-systems. Since the solution of the coupled system is mainly determined by the solution of their elliptic (typically pressure) components, (CPR-type) two-stage preconditioning methods still belong to the most popular approaches to tackle such coupled systems. After a suitable extraction and decoupling, the numerically most costly step in such two-stage methods consists in solving these elliptic sub-systems. It is known that algebraic multigrid (AMG) provides a technique to solve elliptic linear equations very efficiently. The main advantage of AMG-based solvers – their numerical scalability – makes them particularly efficient for solving huge linear systems.

Depending on the application, the system's properties range from simple to highly indefinite. Unfortunately decoupling pressure and saturation related parts may introduce further difficulties. Consequently, in complex industrial simulations, the application of AMG to elliptic sub-systems might not be straightforward. In fact, an important goal in defining an efficient two-stage preconditioning strategy consists in extracting elliptic sub-systems that are suitable for an efficient AMG solution and, at the same time, ensure a fast overall convergence of the two-stage approach.

The importance of this will be demonstrated for several industrial cases. In particular, some of these cases are very hard to solve by AMG if applied in a standard way.

Preliminary results for a CPR-type coupling of SAMG to CMG's PARASOL, a variable degree variable ordering ILU preconditioner using FGMRES, are compared to using PARASOL by itself. Alternative preconditioning operators will be presented giving elliptic sub-systems which are not only more suitable for applying AMG efficiently but also help accelerate the CPR-type process. Comparisons with one-level iterative methods will show the acceleration by AMG is highly superior. Finally, a strategy is presented that combines all linear solver parts in one single AMG-iteration. In this sense CPR can be seen as a special case of AMG for systems. This, in turn, yields a – formally – very simple but simultaneously very flexible solution approach.

Introduction

Solving the system of linear equations arising in fully and adaptive implicit models (FIM, AIM) in a fast and robust way is one of the key challenges for efficiently applying linear solvers in reservoir simulation. The implicit formulations have the advantage of not being limited by any CFL-like constraint for time steps and thus result in more flexible time steps.

The systems arising in implicit approaches correspond to Jacobian systems resulting from the linearization of the physical mass balance equations. These Jacobian matrices are associated with various different types of unknowns such as pressures, saturations, or concentrations. Temperature or mechanical properties could be part of the system as well, depending on the type of application [6] [21][26].

The highly coupled systems lead to equations of mixed hyperbolic-parabolic type. Parts of the Jacobian matrix related to the pressure unknown have parabolic or elliptic type, whereas parts of the matrix associated with convection are of hyperbolic type [18][21]. One-level iterative methods like incomplete factorizations can deal with the matrices as they are reported to arise in the convection related sub-part [17][20]. However, for the elliptic pressure related part multigrid is known to be more efficient in general [28], especially for large problems.

Consequently the Constrained Pressure Residual (CPR) method [33][34] aims at preconditioning the one-level method by computing a pressure approximation, e.g. by multigrid, in advance to applying a one-level method.

Each component of the CPR method still is influenced by the complex physics arising in reservoir simulation. Especially the pressure equation might be highly heterogeneous due to varying permeability fields. Furthermore the presence of sources and sinks for the flow given by wells or fractures, might introduce further difficulties. These difficulties do not need to be present for the entire simulation, but might appear or disappear when wells are switched on or off.

Algebraic Multigrid methods (AMG) are known to be able to generally deal with complex geometries as well as with varying coefficients and anisotropies [10][28]. Because it does not rely on geometric information, AMG is a popular choice for the pressure preconditioning step in reservoir simulation applications.

AMG originally was designed for solving matrices arising from the discretization of elliptic partial differential equations. However, the pressure matrices arising in reservoir simulations are extracted from Jaciobian matrices and influenced by the complex physics of porous media flow. As a consequence, while AMG in many cases is able to solve the pressure equations in FIM simulations very well, in other cases convergence may seriously suffer from these physical influences. Especially source terms caused by wells and fractures can have distructive influences on the matrix properties from AMG's point of view.

Furthermore the pressure in general should not be seen as an independent unknown when solving the FIM pressure equations. The Jacobian matrix always contains couplings to various other unknowns. Therefore it is a common approach to apply a decoupling operator to the Jacobian matrix before starting the CPR process.

In this work a preconditioning method is described that provides a decoupling in a physical rather than in an algebraic sense by computing a quasi-IMPES pressure equation. Furthermore the pressure matrix' properties with respect to AMG are taken into consideration to ensure a convergence acceleration even in highly ill-conditioned test cases.

This paper is organized as follows: first the basic models and methods are briefly described. Based on industrial test cases of various sizes the acceleration power of AMG is illustrated. Timing results for some full simulation runs will demonstrate the significant performance benefits one can achieve using AMG for accelerating CPR. Additional techniques under development are then described and the dynamic rowsum preconditioner is presented. Decoupling effects are analyzed and a way to use AMG in a CPR-manner for the full system is described. Results focusing on the convergence properties are presented.

Description of the equations

This study focuses on black-oil reservoir simulation [2], however, the principles are applicable for compositional simulations as well. The following conservation equations for three phases (namely oil, gas and water) need to be fulfilled:

$$\begin{aligned}
&\frac{\partial}{\partial t}\left(\frac{\phi S_o}{B_o}\right) + q_o - \nabla\left[\frac{\lambda_o k}{B_o}(\nabla p_o - \rho_o \bar{g})\right] = 0 \\
&\frac{\partial}{\partial t}\left[\phi\left(\frac{R_s S_o}{B_o} + E_g S_g\right)\right] + R_s q_o + q_g - \nabla\left[\frac{R_s \lambda_o k}{B_o}(\nabla p_o - \rho_o \bar{g}) + E_g \lambda_g k(\nabla p_g - \rho_g \bar{g})\right] = 0 \\
&\frac{\partial}{\partial t}\left(\frac{\phi S_w}{B_w}\right) + q_w - \nabla\left[\frac{\lambda_w k}{B_w}(\nabla p_w - \rho_w \bar{g})\right] = 0
\end{aligned} \tag{1}$$

With the following parameters:

B_l = formation volume factor for phase l	p_l = pressure in phase l	t = time
E_g = gas expansion factor	q_l = well term for phase l	λ_l = mobility of phase l
$\bar{g}$ = gravity vector	R_s = solution gas-oil ratio	ρ_l = density of phase l
k = absolute permeability	S_l = saturation of phase l	ϕ_l = porosity

These equations are discretized into finite difference form using a control volume approach with central differences in space and upstream mobilities. Advancement in time is done using an adaptive implicit (AIM) strategy. Three primary variables are chosen for each grid cell. If there is free gas present the primary variables are oil pressure, water saturation, and oil saturation (p_o, S_w, S_o). When no free gas is present, the bubble point pressure replaces the oil saturation (p_o, S_w, p_b). More general details of the model can be found elsewhere [2][14]. In general compositional simulations the number of saturation and concentration unknowns will be larger, depending on the number of components.

Due to pressure and saturation dependent properties, the system (1) is highly non-linear. Newton iteration is used to linearize the problem which requires the solution of an associated Jacobian system. Ordering the system by unknown type it can described as a matrix problem given by:

$$A_{unknown-wise}\begin{pmatrix} p \\ S \end{pmatrix} = \begin{pmatrix} A_{pp} & A_{ps} \\ A_{sp} & A_{ss} \end{pmatrix}\begin{pmatrix} p \\ S \end{pmatrix} = \begin{pmatrix} f_p \\ f_S \end{pmatrix} \tag{2}$$

To describe the preconditioning methods further below, a point-wise notation of the matrix is useful. With n_p being the number of grid points and n_u the number of physical unknowns per point, the Jacobian matrix is ordered into different blocks:

$$A_{point-wise} = \begin{pmatrix} [A]^{1,1} & \cdots & [A]^{1,n_p} \\ \vdots & \ddots & \vdots \\ [A]^{n_p,1} & \cdots & [A]^{n_p,n_p} \end{pmatrix} = \begin{pmatrix} [A_{pp}]^{1,1} & [A_{ps}]^{1,1} & \cdots & [A_{pp}]^{1,n_p} & [A_{ps}]^{1,n_p} \\ [A_{sp}]^{1,1} & [A_{ss}]^{1,1} & & [A_{sp}]^{1,n_p} & [A_{ss}]^{1,n_p} \\ & \vdots & & \ddots & \vdots \\ [A_{pp}]^{n_p,1} & [A_{ps}]^{n_p,1} & & [A_{pp}]^{n_p,n_p} & [A_{ps}]^{n_p,n_p} \\ [A_{sp}]^{n_p,1} & [A_{ss}]^{n_p,1} & \cdots & [A_{sp}]^{n_p,n_p} & [A_{ss}]^{n_p,n_p} \end{pmatrix} \quad (3)$$

Finally each single matrix-entry can be addressed as $[a_{x,y}]^{i,j}$ with x and y defining the position within the point-block i,j:

$$[A_{pp}]^{i,j} = [a_{1,1}]^{i,j} \qquad [A_{ps}]^{i,j} = \begin{pmatrix} [a_{1,2}]^{i,j} & \cdots & [a_{1,n_u}]^{i,j} \end{pmatrix}$$
$$[A_{sp}]^{i,j} = \begin{pmatrix} [a_{2,1}]^{i,j} \\ \vdots \\ [a_{n_u,1}]^{i,j} \end{pmatrix} \quad [A_{ss}]^{i,j} = \begin{pmatrix} [a_{2,2}]^{i,j} & \cdots & [a_{2,n_u}]^{i,j} \\ \vdots & \ddots & \vdots \\ [a_{n_u,2}]^{i,j} & \cdots & [a_{n_u,n_u}]^{i,j} \end{pmatrix} \quad (4)$$

In the following block-brackets *[ ]* will refer to this point-based ordering. In situations, where both orderings are used, an implicit reordering is implied.

The methods to be described are tested with several test cases supplied by CMG. Namely these are:

SPE10 This model is based on the fine grid case for Model 2 from the Tenth SPE Comparative Solution Project [9]. It has 60 x 220 x 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 three-phase, oil-water-gas water injection problem. It is run for 2000 days simulated time.

MX1041 This model consists of 40 five spot patterns in 20 x 2 areal arrangement divided into 10 vertical layers. It has 1041 x 105 x 10 = 1.09 million cells, all active. It is a three-phase oil-water-gas water injection problem and is simulated for 15 years. It has 103 wells.

MX521 This model consists of 90 five spot patterns in 10 x 9 areal arrangement divided into 20 vertical layers. It has 521 x 469 x 20 = 4.88 million cells, all active. It is a three-phase oil-water-gas water injection problem and is simulated for 15 years. It has 200 wells.

TM2T This model is a finer grid version of a client data set. Each cell from the original client data set was split equally, areally into 3x3x1 cells. It has 369 x 1200 x 44 = 1.95 million cells with 1.64 million active. It is a three-phase, oil-water-gas water injection problem. It is run for 3440 days simulated time. It has eleven wells.

CPUTEST This model is based on a client data set. It has 26 x 81 x 16 = 33.7 thousand grid cells with 29.5 thousand active. It is a corner point approximation to a radial-like grid. It is a three-phase, oil-water-gas water and gas injection problem. It is run for approximately 32 years simulated time. It has 47 wells.

Constrained Pressure Residual method

The Constrained Pressure Residual (CPR) [33][34] method is a common strategy for solving the described linear systems of equations. This method, also known as two-stage preconditioning, exploits the different properties of the different parts of the matrix. The basic strategy for CPR's k^{th} iteration is as follows [7][21]:

1. Roughly approximate the pressure: $A_{pp}p' = f_p - A_{ps}S^{k-1}$ (5)
2. Apply an incomplete factorization iteration based on the current solution vector $\begin{pmatrix} p' \\ S^{k-1} \end{pmatrix}$

The pressure is known to drive fluid flows [1] and thus it is used to accelerate the process. Because the pressure related part A_{pp} in the Jacobian is expected to have properties close to the ones of a matrix derived from the discretization of an elliptic PDE, multigrid methods are a common choice for this purpose. These methods are known to be in general very efficient for such kind of matrices [10] [28]. Fraunhofer's SAMG software library [31] will be used in this study.
The saturation related part A_{ss} is usually assumed to be strongly diagonally dominant. Consequently single-level iterative methods like incomplete factorizations are efficient for updating the saturation.

Before applying CPR, it is common to precondition the Jacobian by a matrix L in order to decouple pressure and saturation:

$$L\,A = L\begin{pmatrix} A_{pp} & A_{ps} \\ A_{sp} & A_{ss} \end{pmatrix} = \begin{pmatrix} \tilde{A}_{pp} & \tilde{A}_{ps} \\ \tilde{A}_{sp} & \tilde{A}_{ss} \end{pmatrix} = \tilde{A} \quad (6)$$

This operator aims at minimizing $\|\tilde{A}_{ps}\|$. Consequently the fact that the saturation is an unknown, too, is decoupled from computing the pressure unknown. In order to construct this decoupling operator L, an important observation is exploited: The pressure-saturation couplings $[A_{ps}]^{i,i}$ within the diagonal point-blocks of A are expected to dominate those in off-diagonal

blocks [17][20][26]. Thus the decoupling operator L aims at decoupling the diagonal blocks $[\tilde{A}]^{i,i}$, only. The resulting operator applied to the point-wise ordered matrix will therefore have a block-diagonal structure:

$$L_{point-wise} = \begin{pmatrix} [L]^1 & & 0 \\ & \ddots & \\ 0 & & [L]^{n_p} \end{pmatrix} \quad \text{with} \quad [L]^i \in \mathbb{R}^{n_u \times n_u} \tag{7}$$

Applying any matrix operations on the small $n_u \times n_u$ blocks $[L]^i$ can be expected to be inexpensive. This gives a great flexibility in how to construct L and various approaches are reported in the literature. The Alternate Block Factorization (ABF) [3], developed to decouple general discretized systems of PDEs by a block diagonal scaling, is one option. A more conservative method has been developed based on ABF [30]. Other approaches use Schur complements on the diagonal blocks [6][17] or try to reduce the impact on the pressure matrix by using local QR decompositions [6].

A common goal of all developments is to apply as few changes as possible to the linear system. Especially conserving the 'elliptic-like' properties of the pressure matrix is of importance for the efficient application of multigrid methods in CPR's first stage. Further approaches focusing on this aspect will be described in later chapters.

Algebraic Multigrid (AMG)

Scalable linear solvers are necessary in reservoir simulations because the linear solver is one of the biggest factors with respect to runtime. Multigrid methods provide the required scalability by exploiting a hierarchy of grid levels and thus give a fast reduction of both high and low frequency error components.

Its good numerical scalability and robustness [10] makes AMG an efficient solver especially for large-scale problems. In contrast to geometric multigrid, algebraic multigrid methods do not explicitly rely on any geometric information. The hierarchy of grids is computed purely algebraically in a so-called 'setup' phase before starting the iterative solution process. Therefore the user is not concerned with providing a reasonable hierarchy and the computed coarse levels are moreover adapted to the particular matrix to solve. Thus AMG in practice is known to be able to deal with unsymmetric matrices as well as with heterogeneities and anisotropies.

Therefore AMG is especially interesting for the application in reservoir simulation. Next to their further increasing size, the linear problems to solve in this context are often challenging. This is due to the complex simulated physics, including heterogeneous permeability fields, varying external influences and complex grids. Because of its scalability and robustness [10], AMG in many cases is superior to classical one-level solvers [28].

Finally, because it combines certain techniques, numerous variations of AMG algorithms are possible and thus AMG offers a great flexibility. Nevertheless, because it can be used as a 'black-box' solver with a robust default strategy, in general the integration of AMG to a simulation code is rather easy. Detailed information on the AMG technique can be found in the literature [11][24][29][32].

In computing the setup AMG mimics geometric grids by interpreting the matrix as a graph and distinguishing 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 operator using the Galerkin principle [24][29]. The interpolation operators between different levels are defined by computing weights again based on the concrete matrix entries.
In the solution phase the efficient interplay of simple relaxation schemes and a coarse grid correction is exploited, recursively based on the hierarchy of levels.

The user is not limited to a fixed AMG strategy. For each part of the AMG method various options are available. Even combining different setup algorithms on different levels as well as combinations of different smoothers are possible. This way the basic AMG strategy can be adapted to the user's application, while the strategy itself takes different properties of each particular matrix into consideration.

However, one has to keep in mind AMG originally was developed for matrices of certain classes, especially for weakly diagonally dominant M-matrices as they arise in the discretization of elliptic partial differential equations. In practical applications, especially in industrial contexts, such ideal properties only rarely appear, although they're often close to them. Consequently more robust AMG approaches are being developed, while at the same time it must be ensured the matrices to solve are as suitable for AMG as possible. For fully implicit reservoir simulations this means all information in the Jacobian matrix should be exploited to find a reasonable compromise between the solvability by AMG and the acceleration of CPR.

Effect of AMG on the full simulation

To demonstrate the potential acceleration of convergence speed – and thus runtime – by using AMG in the CPR context, benchmarks with CMG's IMEX simulator have been made.

As reference method to compare with, CMG's sparse linear solver, PARASOL is used with OpenMP parallelization. This is GMRes preconditioned by a variable degree ILU. Parallelization is achieved through the partitioning of the reservoir into disjoint sets of grid cells (classes). The classes are further organized into levels where there must be no flow between grid cells of different classes at the same level [13] [16].

The SAMG software package [31] was coupled to PARASOL via a CPR method (5). To get a first impression of the effect of using AMG, the resulting IMEX-SAMG uses the ABF decoupling method [3]. Additionally, each row is multiplied by the scalar value of the mass or surface volume of the total fluid in the grid cell associated with the row. The number of AMG cycles and the number of ILU steps performed in the first and second preconditioning step, respectively, is crucial for the overall efficiency. A loose relative residual tolerance for the pressure solution was chosen leading to a small number of AMG cycles per outer iteration.

Additionally, a threshold checking mechanism is under development for IMEX-SAMG, to control where AMG preconditioning is used. Two-stage preconditioners are based on the idea that the coupled system solutions are mainly determined by the solution of their nearly elliptic portion (i.e., pressure) for which AMG is very efficient. However, within a simulation run matrices arise in which the hyperbolic portion (non-pressure variables which have a directional dependence) dominates. In these cases a single level technique such as that of standard PARASOL may be more effective to determine a solution.

A simple test has been constructed to determine if AMG should be used for a particular Newton iteration. For each point-block row i the elements in $[A_{ps}]^{i,j}$ $(j=1,\ldots,n_u)$ are compared to the diagonal element of the pressure matrix. If a certain fraction of the matrix is deemed to have "large-off-diagonal" rows, the two-stage preconditioner is not used for that matrix. This method is still in development and default threshold levels for each variable type and the total fraction are being determined. Early tests show the two-stage preconditioner is turned off for matrices where IMEX-SAMG performs poorly and remains on for matrices where IMEX-SAMG outperforms IMEX using the same thresholds for all test problems. However, because it is not finalized, this threshold checking mechanism was not used in the following simulations.

Results are presented here for cases run on a Dell Precision T7500 with two 3.3 GHz Xeon X5680 six core processors and 48GB of 1333MHz DDR2 RAM. For the purposes of this paper, Intel's designations of two six core processors for a total of twelve cores are designated as twelve CPUs. For all test-problems Jacobian matrices were constructed using the adaptive implicit technique, generalized red-black ordering was used and the numerical parameters for IMEX were held constant between IMEX-SAMG runs and IMEX only runs. The number of level one solver classes equals the number of CPUs used for parallel runs. For these red-black ordering cases, ILU(0) was used for intra-class terms and ILU(1) for inter-class terms.

For IMEX-SAMG, SAMG was essentially used with its default parameters. The accelerator was changed to FGMRes and the smoother has been adapted. One sweep of full Gauss-Seidel and full Jacobi was used for pre- and post-smoothing respectively. In both cases an additional Gauss-Seidel F-relaxation sweep was added. As discussed above, a case dependent residual reduction between 0.7 and 0.8 was desired for the pressure.

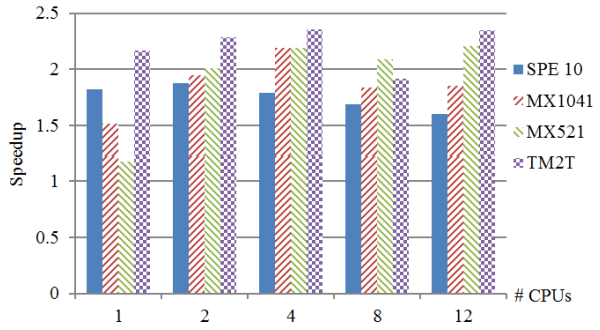

Figure 1: Speedup for the full simulation time by using AMG

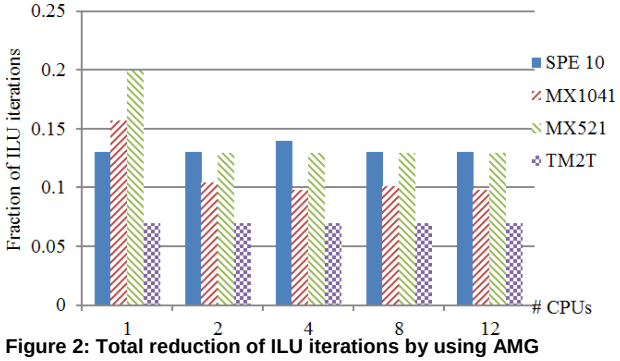

Figure 2: Total reduction of ILU iterations by using AMG

The results in Figure 1 show the significant benefit that can be obtained for the total runtime when using AMG in a CPR manner. Runtimes for these cases were up sped by up to 2.5 times. This, as shown in Figure 2, is essentially because the number of CPR iterations is reduced to around 10% of the number of necessary ILU iterations when not using CPR. The corresponding time saving comes at the cost of the SAMG setup and solution phases. However, the cost of AMG is much smaller than the benefit from saving ILU iterations. The impact of the setup phase can be lessened since much of the setup is reused from matrix to matrix in the simulation. For these cases only one to three AMG cycles on average are necessary per outer iteration.

Plots of the increase in iteration count (relative to the serial case) against the number of CPUs are shown in Figure 3 and Figure 4. An increase in iterations is expected for a larger number of CPUs due to the parallelization strategy used for ILU [5]. For most of the test cases IMEX-SAMG has a smaller increase in iteration count than IMEX.

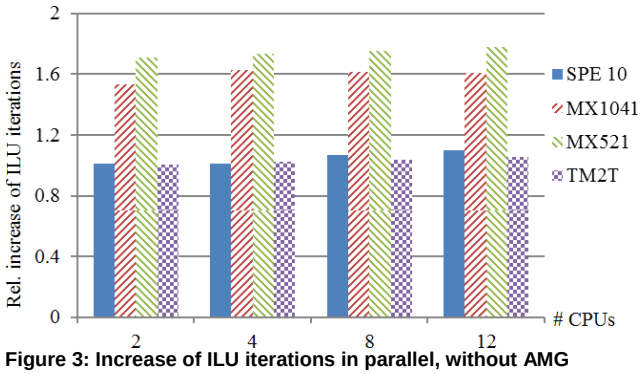

Figure 3: Increase of ILU iterations in parallel, without AMG

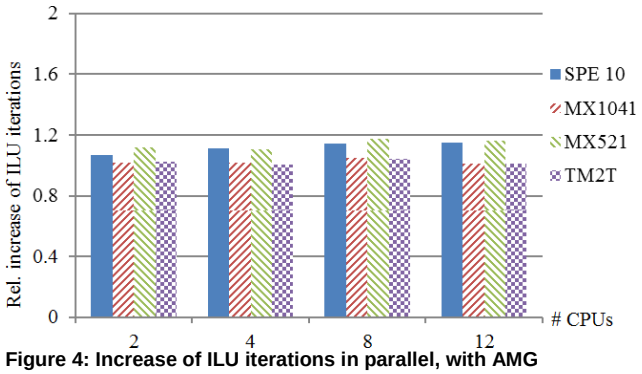

Figure 4: Increase of ILU iterations in parallel, with AMG

These test cases were also run using natural ordering. For these natural ordering cases, ILU(1) was used for intra-class terms and ILU(2) for inter-class terms. The total elapsed times using IMEX and IMEX-SAMG were not as good as those using red-black ordering. The total time spent performing the inner AMG preconditioning step for red-black varied between 0.4 and 0.7of natural. This implies that time cost per grid cell is close to the same for AMG regardless of ordering.

Other data sets have been tested that do not perform as well using IMEX-SAMG. This is predominately due to convergence issues for AMG in CPR's first stage. Consequently, for these cases, CPR does not improve the efficiency. It is therefore desirable to construct a preconditioning method that provides solvability for AMG as well as acceleration of the total CPR process.

Dynamic Rowsum (DRS) preconditioner

The dynamic rowsum preconditioner aims at constructing pressure matrices that are both, solvable by AMG in a robust way, as well as accelerating the CPR process by a significant extent. It is based on the approach described in [26] where – keeping in mind the pressure is the first unknown - the blocks $[L]^i$ (7) in the preconditioning operator L are defined as:

$$[L]^i = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ & 1 & & 0 \\ & & \ddots & \\ 0 & & & 1 \end{pmatrix} \tag{8}$$

The resulting pressure matrix $\tilde{A}_{pp}$ is the sum of all pressure related parts in the Jacobian system. Consequently a pressure equation close to the total pressure equation used in IMPES schemes [12] is generated and thus the method can be considered to give a quasi-IMPES pressure equation.

For solving IMPES-like total pressure equations AMG already is successfully applied [4][15], so AMG should be an efficient choice for the quasi-IMPES pressure equation, as well. In fact, since the Jacobian's 'elliptic parts' are concentrated in $\tilde{A}_{pp}$ [22], this resulting pressure equation can be expected to have comparable properties. Thus no new difficulties from AMG's perspective are introduced by the preconditioner.

However, such difficulties might already have been present in the original Jacobian system. This especially holds for the matrices under consideration, which do not directly result from an elliptic PDE, but are rather extracted parts of Jacobian matrices. If the properties expected by AMG, are violated too much, the interplay between smoothing and coarse grid correction does not work properly anymore and therefore AMG might end up with convergence troubles.

In practice the pressure matrices to solve even suffer from indefiniteness. Figure 5 shows the plot of eigenvalues for A_{pp} in one timestep of case cputest without applying any preconditioning. To focus on eigenvalues with negative real parts – in here about 2% of all eigenvalues - only those around zero are shown. Next to the indefiniteness, there are 11% negative diagonal entries. Their appearance seems to be directly related to the relatively high number of activated wells at the particular time step.

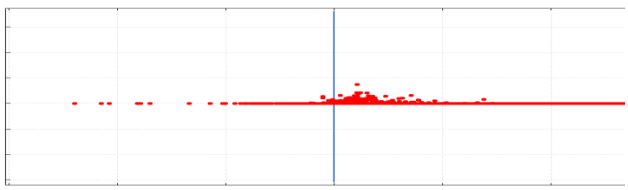

Figure 5: Eigenvalues of pressure matrix in the complex plane, cputest, original matrix. *Re* = 0 is marked by a blue line. Only an excerpt around 0 is shown.

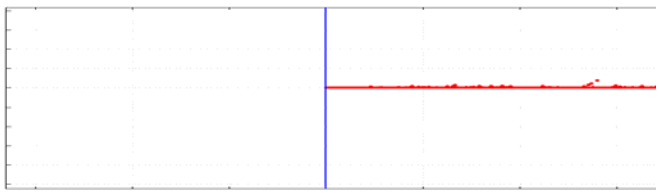

Figure 6: Eigenvalues of pressure matrix in the complex plane, cputest, after dynamic rowsum. *Re* = 0 is marked by a blue line. Only an excerpt around 0 is shown.

Applying AMG directly to this matrix turns out to be a challenge. But even for such ill-conditioned matrices there are fallback strategies to make AMG converge. For this particular matrix a combination of a Schwarz iterative method [27], defect correction [23] and a coarsening strategy actively preventing negative diagonals at coarser levels allows the solution of this matrix using AMG.

Obviously it is not desirable to investigate such fallback strategies individually per matrix. Actively avoiding difficulties should allow applying AMG in a standard way. For this purpose the preconditioner's blocks (8) are modified to be dynamic:

$$[L]^i = \begin{pmatrix} \delta_1^i & \delta_2^i & \cdots & \delta_{n_u}^i \\ & 1 & & 0 \\ & & \ddots & \\ 0 & & & 1 \end{pmatrix} \tag{9}$$

The entries δ^i_x are computed individually for each point-block i. The physical forces, which the matrix properties suffer from, do not necessarily appear in all phases to the same extent. The resulting different properties are thus exploited to avoid unwanted matrix properties with respect to the applicability of AMG:

$$\delta_x^i = \begin{cases} 0 & if \quad \frac{[a_{x,1}]^{i,i}}{\sum_{j=1,j\neq i}^{n_{points}} \left|[a_{x,1}]^{i,j}\right|} < \varepsilon_{dd} \\ 1 & else \end{cases} \qquad \text{where } 0 \leq \varepsilon_{dd} \leq 1 \tag{10}$$

Only those parts of A are considered for the construction of $\tilde{A}_{pp}$ that are not expected to introduce significant difficulties for efficient application of AMG. In particular, the appearance of negative diagonal entries is prevented. Furthermore, the parameter ε_{dd} controls the severity of the method in terms of the violation of diagonal dominance. In the limiting case ε_{dd}=1 any risk of violating diagonal dominance in $\tilde{A}_{pp}$ is avoided. Experiments showed a trade-off has to be made between the solvability of the matrix by AMG and the necessity to exclude as few rows as possible to obtain a pressure equation close to the one in IMPES schemes.

A further check is introduced for the case ε_{dd} is chosen too small. After all δ^i_x are defined, the diagonal dominance in the resulting ith row of $\tilde{A}_{pp}$ is checked. If it is violated too strongly, the δ^i_x are recomputed with a stricter ε_{dd}. Due to the small size of the block $[L]^i$, this check can be expected to be inexpensive.

If $\delta^i_1 = 0$, then $[L]^i$ – and thus L – will become singular. This is avoided by the following procedure:

- If for an $x > 1$: $\delta^i_x \neq 0$, then in the xth row of $[L]^i$ the 1 on the diagonal has to be moved to the first column.
- Otherwise all δ^i_x are zero. Then the δ^i_x is reset to 1 that is expected to result in as little as possible bad influence on the pressure matrix $\tilde{A}_{pp}$. In order to ensure a regular matrix L, some unwanted influences needs to be accepted.
  In the test cases under consideration this problem occurs at well equations, only, and AMG does not have any problem with it. For more complex cases fallback strategies might be taken into consideration.

Using the dynamic Rowsum method described so far, it is possible to shield AMG from the 'most destructive' physical influences. In Figure 6 the plot of eigenvalues for the same timestep in cputest after applying the dynamic rowsum shows the success – again only the excerpt of the complex plane around 0 is shown. In this particular case there are even no negative eigenvalues remaining in $\tilde{A}_{pp}$.

Next to avoiding unwanted properties in $\tilde{A}_{pp}$, a second dynamic mechanism is introduced. This aims at further improving the overall CPR convergence by monitoring the pressure-saturation couplings. If for a block-row i there exists an $x > 1$, such that:

$$\sum_{j=1}^{n_p} \left|[a_{1,x}]^{i,j}\right| < \epsilon_{ps} \left|[a_{1,1}]^{i,i}\right| \tag{11}$$

then, because the corresponding saturation has a negligible influence on the pressure, this particular pressure-saturation coupling is a weak one. Accounting for it when computing the total pressure equation in grid-block i does not contribute to accelerating the total convergence. But it would add 'noise' to the resulting pressure equation. Moreover one would risk strengthening the already weak pressure-saturation coupling. Therefore, if in a block-row i for $x > 1$ in which equation (11) holds, the corresponding δ^i_x is set to zero - if this would not make L singular.

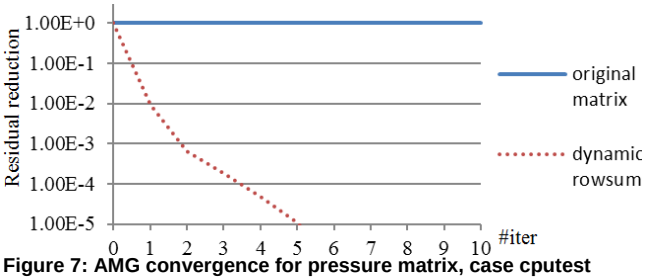

Figure 7: AMG convergence for pressure matrix, case cputest

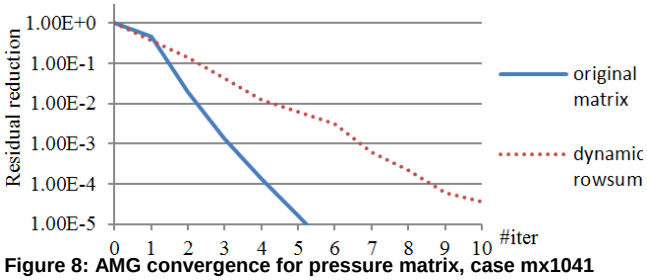

Figure 8: AMG convergence for pressure matrix, case mx1041

In order to analyze the effects of the dynamic rowsum method, the convergence behavior of AMG as a preconditioner for FGMRes is given for the extracted pressure sub-problem. The results in Figure 7 and Figure 8 are showing that AMG is able to solve it in both test cases after the application of the dynamic rowsum method.

Obviously it is desirable for AMG to converge for the given pressure matrix in order to accelerate the global CPR process. Practical experience shows AMG even if it diverges still can accelerate CPR. But that's an application dependent effect one should not rely on. Therefore it is preferable to provide a pressure matrix that AMG is able to handle properly. In fact, using the dynamic rowsum preconditioner, this works for all matrices considered in this study.

The result in Figure 8 is no contradiction, although AMG after applying the dynamic rowsum method converges slower than for the original pressure matrix. But AMG still converges well and even more: the reason for the slower convergence is the introduction of influences by computing the quasi-IMPES total pressure equation. Thus the resulting pressure solution can be expected to accelerate CPR.

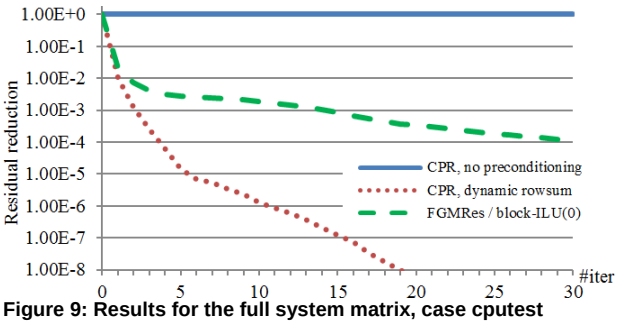

Figure 9: Results for the full system matrix, case cputest

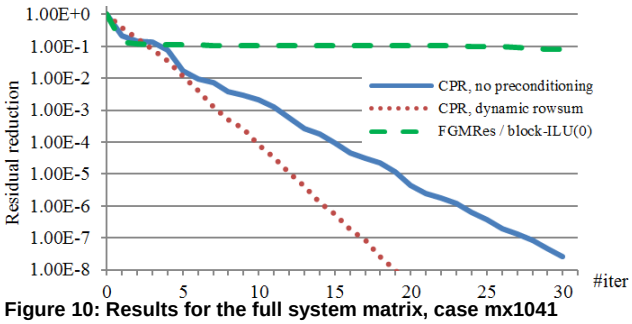

Figure 10: Results for the full system matrix, case mx1041

To focus on the effect of the preconditioner a default AMG strategy without any tuning of parameters is used. For this reason convergence histories are given instead of performance data for representative matrices. FGMRes preconditioned by block-ILU(0) is used as reference to compare with CPR methods. Because a block-structure is available anyway, block-ILU is chosen in this study because of its higher robustness [8]. CPR is furthermore used as preconditioner for FGMRes [25] in the considered implementation.

The one-level method was able to solve all given test cases. CPR, however, was not always able to exploit the pressure preconditioning by AMG if it was applied to the original matrix. In extremely ill-conditioned cases, AMG's divergence even prevented total convergence.

Using the dynamic rowsum preconditioner in all test cases under consideration has given pressure sub-problems that AMG can solve without difficulties. This pressure solution secondly is meaningful in terms of the full Jacobian matrix and thus speeds up the total convergence.

Decoupling operator

The algebraic decoupling of pressure and saturation, i.e. minimizing $\|\tilde{A}_{ps}\|$, has not been considered to this point. To perform the decoupling, two alternative right preconditioning methods are presented. The right preconditioner R is constructed analogously to the left one (7):

$$\hat{A} = \tilde{A}\,R \quad \text{with} \quad R_{point-wise} = \begin{pmatrix} [R]^1 & & 0 \\ & \ddots & \\ 0 & & [R]^{n_p} \end{pmatrix} \quad \text{and} \quad [R]^i \in \mathbb{R}^{n_u \times n_u} \tag{12}$$

The first approach, the quasi Alternate Block Factorization (qABF), constructs $[R]^i$ for block-row i by inverting small matrices V based on the diagonal blocks $[\tilde{A}]^{i,i}$:

$$V = \begin{pmatrix} \tilde{A}_{pp} & \tilde{A}_{ps} \\ 0 & \tilde{A}_{ss} \end{pmatrix} \quad \Rightarrow \quad V^{-1} = \begin{pmatrix} S & T \\ 0 & U \end{pmatrix} \quad \text{where } S \in \mathbb{R}, T \in \mathbb{R}^{1 \times n_u}, U \in \mathbb{R}^{n_u \times n_u} \tag{13}$$

If V is not regular and the LU-decomposition fails, $[R]^i$ is simply set to the unity matrix. Otherwise $[R]^i$ is defined as:

$$[R]^i = \begin{pmatrix} 1 & T \\ 0 & U \end{pmatrix} \tag{14}$$

It can easily be shown the resulting operator gives a pressure-saturation decoupling in the diagonal blocks. However, there is one disadvantage; after the application of the right qABF, new off-diagonal entries might have been introduced in $\hat{A}_{ss}$. It is guaranteed no new non-zero blocks are introduced, but new entries might occur in those already existing blocks i,j where $[\hat{A}_{sp}]^{i,j} \neq 0$. This is therefore not a problem for block-ILU.

However, a second alternative is presented, the saturation column eliminating method (SCE). This method will not be able to perfectly decouple pressure and saturation – even not in the diagonal block, but it will not mix pressure and saturation related columns and thus reduces the risk of additional fill-in for the matrix.
As a first step the residual vector $r = Au - f$ is computed. Afterwards the saturation-unknown z with the smallest residual is chosen. The construction of $[R]^i$ again is based on the diagonal block $[\tilde{A}]^{i,i}$:

$$R^i = \begin{pmatrix} 1 & 0 & \cdots & & \cdots & 0 \\ 0 & 1 & & & & \\ \vdots & & \ddots & & & \\ & & & 1 & & \\ 0 & -\frac{\tilde{a}^{i,i}_{1,1}}{\tilde{a}^{i,i}_{1,z}} & \cdots & -\frac{\tilde{a}^{i,i}_{1,z-1}}{\tilde{a}^{i,i}_{1,z}} & 1 & -\frac{\tilde{a}^{i,i}_{1,z+1}}{\tilde{a}^{i,i}_{1,z}} & \cdots & -\frac{\tilde{a}^{i,i}_{1,n_u}}{\tilde{a}^{i,i}_{1,z}} \\ \vdots & & & & 1 & & & \\ 0 & & & & & & & 1 \end{pmatrix} \leftarrow z^{th}\ row \tag{15}$$

After the application of the SCE operator, the pressure in the diagonal is decoupled from all saturations except for the zth one. But because this coupling corresponds to the smallest residual, this is accepted. Due to off-diagonal entries in A_{ps} a perfect decoupling (meaning $\hat{A}_{ps} = 0$) will usually not be possible anyway.

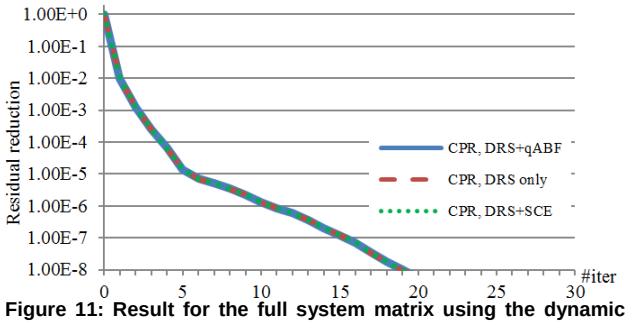

Figure 11: Result for the full system matrix using the dynamic rowsum (DRS) method and varying the decoupling, case cputest

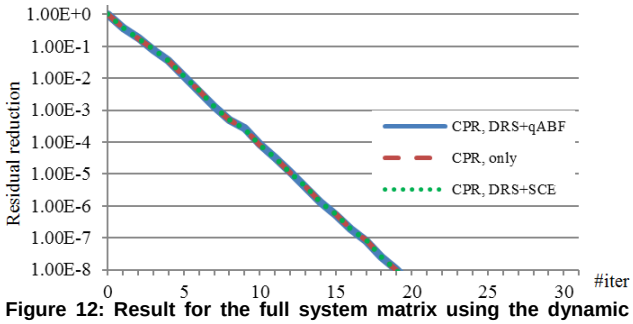

Figure 12: Result for the full system matrix using the dynamic rowsum (DRS) method and varying the decoupling, case mx1041

The results in Figure 11 and Figure 12 indicate that for the given test matrices generating quasi-IMPES pressure equations is more important than an algebraic decoupling of pressure and saturation. The decoupling in the diagonal entries of $\hat{A}_{ps}$ for both methods does not give a further advantage to the CPR process. For the given matrices this seems to be due to relatively large off-diagonal pressure saturation couplings.

Remarks:

- In the literature it is common to use a left preconditioning operator for decoupling purposes. Right preconditioners influence the matrix' scaling and might introduce new non-zero entries [19]. However, with the proposed methods, the scaling is influenced for the saturation related columns, only. Since they are expected to be highly diagonally dominant, major problems for the incomplete factorizations are not expected.
  The introduction of new non-zero entries is possible for existing non-zero blocks, only. Thus, block-ILU should not suffer from it. Moreover, the SCE operator reduces the risk to the same level one has when using decoupling operators from the left.
- Although the algebraic decoupling does not have a significant influence for the test cases under consideration, there are various cases where it might pay-off. Especially if the pressure-saturation couplings in off-diagonal blocks $[A]^{i,j}$ are really small, experiments show a further improvement in the convergence.

- The proposed decoupling strategies will not influence the pressure equation at all. The success achieved by the DRS method is thus not at risk. This is only possible when decoupling from the right, otherwise the pressure sub-matrix is always influenced.
- Applying the decoupling in advance to the dynamic rowsum method to generate more weak pressure saturation couplings does also not speed up the convergence. This seems to be because of the saturation-pressure couplings $\tilde{A}_{sp}$ that are not removed by the decoupling. It was already suggested in [26] that these have a non-negligible influence.

Solving the full Jacobian system by AMG

Modern AMG implementations are not only able to deal with matrices arising from the discretization of one single partial differential equation, but they can also handle matrices resulting from coupled systems of PDEs [11][31]. Of course the different blocks of physical unknowns of these matrices might have different properties and of course the linear solver must not ignore these differences. Therefore the application of AMG is not straight-forward. However, it is desired to apply AMG to the full system instead of using operator splitting approaches. If the full system is provided to AMG, couplings between different physical unknowns are present at coarser levels, too. Consequently all fine-level techniques can be recursively applied to the coarse levels.

The **unknown-based** AMG approach applies an individual coarsening strategy to the different n_u unknowns. A natural choice is to coarsen the diagonal blocks of an unknown-wise ordered matrix (2) individually. However, depending on the demands of a particular application, further options are conceivable. The coarsening applied to each unknown is equivalent to the one AMG would apply to a matrix resulting from a scalar PDE. This also holds for the interpolation. Different properties of different physical unknowns are therefore taken into account instead of mixing them up. Consequently, using the unknown-based approach, it is even possible to couple physical unknowns defined on different grids. The coupling between these unknowns still will be considered at coarser levels.

The **point-based** AMG is a further alternative. It assumes all unknowns to be discretized on the same grid, but still not all unknowns do need to be present at all grid elements. This is especially important for reservoir simulations where phases might appear and disappear in certain grid elements over time.

The coarsening is not computed based on the given matrix directly. Instead a so-called primary matrix P is constructed for this purpose. This way AMG again tries to mimic geometric grid hierarchies by constructing a coarsening in the block-structure represented by P. There are various options to define this structure that should represent the connectivity of all physical unknowns sufficiently. Geometric information like the distance between points can be taken into consideration. But the primary matrix again can be constructed based on algebraic information found in only the matrix itself. Further details can be found in [11]. In applications where the connectivity structure is implicitly given by one of the physical unknowns, e.g. by a driving force like pressure, it is often reasonable to define P based on this unknown.

For the Jacobian matrices arising in fully implicit reservoir simulations choosing the pressure unknown to define the connectivity structure thus would be a natural choice. The driving properties of the pressure are the motivation behind the CPR method, as well.

In fact CPR can be seen as a special case of AMG for systems of coupled PDEs: Because of the flexibility in how to combine the multigrid method's parts, it is possible to define coarsening and smoothers that mimic CPR. If the coarsening is constructed based on the pressure unknown, only the smoothing operators at the finest level have to be adapted to mimic the CPR method's second stage. Thus, incomplete factorizations are used as post-smoother on this level, merging both stages of CPR. This approach has some advantages:

- The total number of operations decreases by merging post-smoothing and CPR's second stage.
- Pressure and saturation at the finest level are updated simultaneously. This prevents AMG from computing a pressure based on a saturation that is outdated within the next step.
- Serial simulation codes can even exploit more parallelism from external linear solver libraries like XSAMG. Because the linear solver is known to take a significant part of the runtime, this can help speeding up simulations without parallelizing the simulation code itself.

Because the coarsening still is based on the pressure-related part of the matrix, the provided matrix nevertheless needs to be solvable by AMG. Therefore methods like the dynamic rowsum preconditioner still are necessary. However, it can be an option to define the primary matrix P based on the DRS method instead of physically changing the Jacobian matrix. Moreover, because pressure and saturation at the finest level are updated simultaneously, the algebraic decoupling is of smaller importance.

Finally, combining the entire linear solution part in an AMG iteration gives a higher flexibility. The full set of available AMG methods is available and thus various types of AMG cycles are possible. This of course also includes exploiting information obtained during the preconditioning stage by the AMG method.

Furthermore, the method is expandable. More complex reservoir simulations might deal with additional unknowns. For those described by elliptic PDEs, AMG again could be an efficient option for the resulting sub-systems in the Jacobian. Instead of solving for these unknowns in a segregated approach, applying AMG to the full system again combines the linear solution in one single iterative scheme.

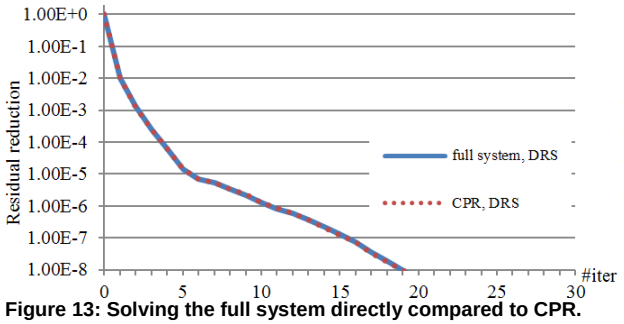

Figure 13: Solving the full system directly compared to CPR. Both using the dynamic rowsum technique, case cputest

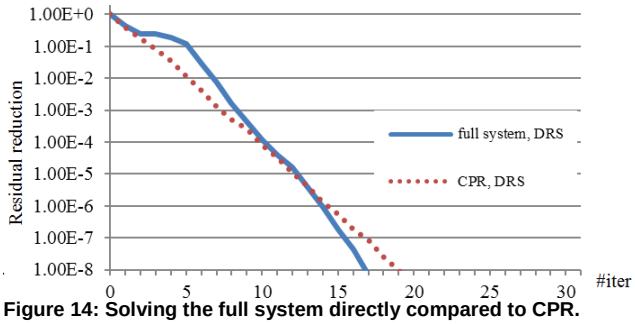

Figure 14: Solving the full system directly compared to CPR. Both using the dynamic rowsum technique, case mx1041

For the results presented in Figure 13 and Figure 14 the full system AMG has been constructed in a way to mimic CPR. Thus the convergence behavior is comparable between both methods. However, small advantages due to merging post-smoothing and updating the saturation are already visible.

Summary and Conclusions

The results obtained within this study show the importance of finding a pressure matrix that is sufficiently solvable by AMG and at the same time is meaningful in terms of the full system to solve.

A CPR-like technique using SAMG to solve the pressure and PARASOL's ILU-GMRes solver in IMEX was developed. Overall speedups on four problems were obtained due to substantial reductions in the number of linear solver iterations. For these four test problems, running in serial and with up to twelve threads, it was found that using red-black ILU with SAMG was faster than using natural ordering ILU with SAMG. For parallel cases, the increase in the number of linear solver iterations was reduced compared to running red/black ILU by itself. A method was developed to automatically revert to ILU rather than CPR for appropriate matrices.

New techniques were developed to handle more difficult matrices in a CPR-like context as described below. These techniques have not yet been implemented for IMEX-SAMG.

The presented DRS preconditioning method is able to construct a quasi-IMPES pressure matrix that is efficiently solvable by AMG – even if the original system was highly ill-conditioned. Therefore the CPR convergence could have been drastically accelerated using AMG. Experiments furthermore showed that algebraic decoupling of pressure and saturation is of minor importance as long as the off-diagonal couplings of both unknowns are too large. These results hold for all matrices under consideration.

The possibility of using system-AMG approaches alternatively to CPR furthermore allows solving matrices in more challenging simulations with a coupled method. There individual strategies, taking the different properties of different unknowns into account, can be applied without the necessity of segregating the different unknowns in the process of the linear solver.

Further research will focus on exploiting the properties of the system-AMG approaches. First, the integration of the preconditioning methods into AMG's setup phase shall be done. This way AMG's setup can benefit from the information obtained while preconditioning. Second, the AMG-cycle can be constructed depending on matrix properties as well. Smoothers can be varied depending on the interplay between pressure and saturation.

Finally, Jacobian matrices arising in reservoir simulations taking further physical effects into account shall be considered. For these matrices the system-AMG approaches may give even more advantages by allowing for a coupled solution.

Acknowledgements

This work has been done as part of an ongoing PhD project at the University of Cologne.
The authors would like to acknowledge the support of the Reservoir Simulation Group at the University of Calgary.

References

[1] Axelsson, O.: "Iterative Solution Methods," Cambridge University Press, 1994

[2] Aziz, K.; Settari, A.: Petroleum Reservoir Simulation, 1979

[3] Bank, R.; Chan, T.; Coughran, W. Smith, R.: The Alternate-Block-Factorization Procedure for Systems of Partial Differential Equations, *BIT Computer Science and Numerical Mathematics,* Lawrence*,* 1989

[4] Batycky, R.P.; Förster, M.; Thiele, M. R.; Stüben, K.: Parallelization of a Commercial Streamline Simulator and Performance on Practical Models, *SPE Reservoir Simulation Symposium,* 2009

[5] Bridson, R.; Tang, W.: A Structural Diagnosis of Some IC Orderings, *SIAM J. Sci. Comput.,* **22**, (5), pp 1527–1532, 2001

[6] Cao, H: Development of Techniques for General Purpose Simulators, PhD thesis, Stanford University 2002

[7] Cao, H.; Tchelepi, H.; Wallis, J.: Parallel Scalable Unstructured CPR-type Linear Solver for Reservoir Simulation, *SPE Annual Technical Conference and Exhibition,* 9-12 October 2005, Dallas, Tx

[8] Chow, E.; Saad, Y.: Experimental Study of ILU Preconditioners for Indefinite Matrices, *Journal of Computational and Applied Mathematics,* **86**, (2), pp. 387-414, 1997

[9] Christie, M.A.; Blunt, M.J.: Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques, *SPE Reservoir Simulation Symposium*, 2001

[10] Cleary, A.; Falgout, R.; Henson, V.; Jones, J.; Manteuffel, T.; Miranda, G.; Ruge, J.; McCormick, S.: Robustness And Scalability of Algebraic Multigrid, *SIAM J. Sci. Comput,* **21**, pp. 1886-1908, 1998

[11] Clees, T.: AMG Strategies for PDE Systems with Applications in Industrial Semiconductor Simulation, Shaker Verlag, 2005

[12] Coats, K.: A Note om IMPES and Some IMPES-Based Simulation Models, *SPE Journal,* Vol. 5, (3), pp. 245-251, 2000

[13] Collins, D.A.; Grabenstetter, J.E.; Sammon, P.H.: A Shared-Memory Parallel Black-Oil Simulator with a Parallel ILU Linear Solver, *SPE Reservoir Simulation Symposium*, 2003

[14] Forsyth, P.A.; Sammon, P.H. 1986. Practical Considerations for Adaptive Implicit Methods in Reservoir Simulation. *Journal of Computational Physics* **62** (2): 265-281.

[15] Geiger, S.; Huangfu, Q.; Reid, F.; Matthäi, S.; Coumou, D.; Belayneh, M.; Fricke, C.; Schmid, K.: Massively Parallel Sector Scale Discrete Fracture and Matrix Simulations, *SPE Reservoir Simulation Symposium,* 2009

[16] Hendrikson, B.; Rothberg, E.: Improving the Run Time and Quality of Nested Dissection Ordering, *SIAM J. Sci. Comput*. Vol. 20, (2), pp. 468-489, 1999

[17] Jiang, Y.: Techniques for Modeling Complex Reservoirs and Advanced Wells, PhD Thesis, Stanford University, 2007

[18] Klie, H.: Krylov-Secant Methods for Solving Large Scale Systems of Coupled Nonlinear Parabolic Equations, Rice University, Houston, Tx, 1996

[19] Klie, H.; Ramé, M.; Wheeler, M.: Two-stage preconditioners for inexact Newton methods in multi-phase reservoir simulation, *CRPC Report,* Rice University, Houston, Tx*,* 1996

[20] Lacroix, S.; Vassilevski, Y.; Wheeler, M.: Iterative Solvers of the Implicit Parallel Accurate Reservoir Simulator (IPARS), I: Single Processor Case, *Numer. Linear Algebra Appl,* 2000

[21] Lacroix, S.; Vassilevski, Y.; Wheeler, J.; Wheeler, M.: Iterative Solution Methods for Modeling Multiphase Flow in Porous Media Fully Implicitly, *SIAM Journal on Scientific Computing,* **25**, pp. 905-926, 2003

[22] Masson, R.; Quandalle, P.; Requena, S.; Scheichl, R.: Parallel Preconditioning for Sedimentary Basin Simulations, *Lecture Notes in Computer Science,* Vol. 2907/2004, pp. 59-81, 2004

[23] Naumovich, A.; Förster, M.; Dwight, R.: Algebraic Multigrid within Deffect Correction for the Linearized Euler Equations, *Numerical Linear Algebra with Applications,* **17**, (2-3), pp. 307-324, 2010

[24] Ruge, J.; Stüben, K.: Algebraic Multigrid (AMG), in *Multigrid Methods (McCormick, S.F. ed.)*, Philadelphia, SIAM, Forntiers in Applied Mathematics, **5**, 1986

[25] Saad, Y.: Iterative Methods for Sparse Linear Systems, PWS Publishing Company, Boston, 1996

[26] Scheichl, R.; Masson, R.; Wendebourg, J: Decoupling and Block Preconditioning for Sedimentary Basin Simulations, *Computational Geosciences,* **7**, pp. 295-318, 2003

[27] Schwarz, H.: Über einen Grenzübergang durch alternierendes Verfahren, *Vierteljahrsschrift der Naturforschenden Gesellschaft in Zürich,* **15**, pp. 272-286, 1870

[28] Stüben, K.: A Review of Algebraic Multigrid, German National Research Center for Information Technology (GMD), 1999

[29] Stüben, K.: An Introduction to Algebraic Multigrid, in [32], pp. 413-532

[30] Stüben, K., Clees, T.; Klie, H.; Lu, B.; Wheeler, M.: Algebraic Multigrid Methods (AMG) for the Efficient Solution of Fully Implicit Formulations in Reservoir Simulation, *SPE Reservoir Simulation Symposium*, 2007

[31] Stüben, K.: SAMG User's Manual, *Fraunhofer SCAI*, 2012

[32] Trottenberg, U.; Oosterlee, C.; Schüller, A.: Multigrid, Academic Press, 2001

[33] Wallis, J.: Incomplete Gaussian Elimination as a Preconditioning for Generalized Conjugate Gradient Acceleration, *SPE Reservoir Simulation Symposium,* 1983

[34] Wallis, J.; Kendall, R.; Little, T.; Nolen, J.: Constrained Residual Acceleration of Conjugate Residual Methods, *SPE Reservoir Simulation Symposium,* 1985