SPE 105832

Algebraic Multigrid Methods (AMG) for the Efficient Solution of Fully Implicit Formulations in Reservoir Simulation

Klaus Stüben, SPE, Tanja Clees, Fraunhofer Institute SCAI, St. Augustin, Germany; Hector Klie, SPE, Bo Lu and Mary F. Wheeler, SPE, Center for Subsurface Modeling, The University of Texas at Austin, USA.

This paper was prepared for presentation at the 2007 SPE Reservoir Simulation Symposium held in Houston, Texas, U.S.A., 26–28 February 2007.

Abstract

A primary challenge for a new generation of reservoir simulators is the accurate description of multiphase flow in highly heterogeneous media and very complex geometries. However, many initiatives in this direction have encountered difficulties in that current solver technology is still insufficient to account for the increasing complexity of coupled linear systems arising in fully implicit formulations. In this respect, a few works have made particular progress in partially exploiting the physics of the problem in the form of two-stage preconditioners.

Two-stage preconditioners are based on the idea that coupled system solutions are mainly determined by the solution of their elliptic components (i.e., pressure). Thus, the procedure consists of extracting and accurately solving pressure subsystems. Residuals associated with this solution are corrected with an additional preconditioning step that recovers part of the global information contained in the original system.

Optimized and highly complex hierarchical methods such as algebraic multigrid (AMG) offer an efficient alternative for solving linear systems that show a "discretely elliptic" nature. When applicable, the major advantage of AMG is its numerical scalability; that is, the numerical work required to solve a given type of matrix problem grows only linearly with the number of variables. Consequently, interest in incorporating AMG methods as basic linear solvers in industrial oil reservoir simulation codes has been steadily increasing for the solution of pressure blocks.

Generally, however, the preconditioner influences the properties of the pressure block to some extent by performing certain algebraic manipulations. Often, the modified pressure blocks are “less favorable” for an efficient treatment by AMG. In this work, we discuss strategies for solving the fully implicit systems that preserve (or generate) the desired ellipticity property required by AMG methods. Additionally, we introduce an iterative coupling scheme as an alternative to fully implicit formulations that is faster and also amenable for AMG implementations. Hence, we demonstrate that our AMG implementation can be applied to efficiently deal with the mixed elliptic-hyperbolic character of these problems. Numerical experiments reveal that the proposed methodology is promising for solving large-scale, complex reservoir problems.

Introduction

The implementation of robust and efficient solvers for fully implicit formulations is one of the main challenges that most simulator developers currently face in the oil industry. The core of the computation at each time step is governed by the successive solution of coupled linear systems (namely, Jacobian systems) that represents the behavior of different physical entities sharing the same discretization element. Generally, these systems are highly nonsymmetric and indefinite. Moreover, the condition number and degree of coupling of these systems may be subject to dramatic changes due to abrupt flow variations induced by the high-heterogeneity and complex well operations during the simulation process. Standard solvers for these systems are still in the early stages of development, despite the intense research activity during the 70’s and 80’s by several oil companies[1-5] and a recent resurgence of interest in the development of a new generation of reservoir simulators[6-12].

Specific preconditioners for fully coupled systems are not frequently encountered in the literature, due in large part to the complexity seen in the contrasting physical behaviors of the primary variables involved: pressures (elliptic or parabolic component) and saturations or concentrations (hyperbolic or convection-dominated component)[13]. Despite the difficulty of these linear systems, there are certain desirable properties associated with each individual coefficient block. Under mild conditions, which are regularly met at a modest time-step size, each of these blocks is irreducible and diagonally dominant. In most cases, diagonal dominance is complemented by the fact

that all off-diagonal entries are negative and the main diagonal entries are all positive. This set of conditions leads to the M-matrix property, which is a desirable property for ensuring convergence of most algebraic iterative methods[14].

One of the main motivations of the present work is to show that exploiting (and preserving) these algebraic properties of each individual block facilitates the development of robust preconditioning techniques. Moreover, corresponding preconditioners may be the result of composing different preconditioning stages to achieve further effectiveness in the solution process. More precisely, if M_0^{-1} is a right preconditioner for the matrix A, further preconditioning may be performed on the system $A_0 = M_0^{-1}A$ by means of, say, a preconditioner M_1^{-1}. The concatenation of these two preconditioners is what is called *two-stage preconditioning*. A generalization of this procedure defines a multi-stage preconditioning method[6,15].

The definition of most two-stage preconditioners in porous media flow has been founded on the basic principle that "pressure governs the solution process"[14]. Loosely speaking, this entails the application of a decoupling operator to facilitate the extraction of the pressure block, followed by the solution of the pressure system itself as the first preconditioning stage. A second stage consists of the application of a preconditioner to recover solution components related to the original coupling or to the non-pressure blocks. In this sense, the two-stage preconditioner has a close resemblance to a coarse-grid iteration where the coarse grid may be seen as the pressure block and the second stage as a smoothing step to correct residuals associated with the pressure solution.

The decoupling process, pressure solution (i.e., the first preconditioning stage), and the second preconditioning stage components must be assembled in such a way that the two-stage preconditioning is as effective as possible. The decoupling should not only aid to weaken the coupling of the pressure block with respect to the other blocks; it is also desirable that the decoupling process have a preconditioning effect on the whole system as well as preserving good algebraic properties for the resulting pressure system. This will facilitate the pressure block solution. In particular, we are aiming at employing optimized and highly complex hierarchical methods, such as algebraic multigrid (AMG)[16,17], for that task.

Inspired by concepts from a wide range of block two-stage preconditioners such as Jacobi, Gauss-Seidel and discrete projection[14], we propose an *iterative coupling scheme* as an alternative to fully-implicit formulations[18]. We have found that AMG is a very attractive option to accelerate iterative coupling computations since the pressure block is readily available in a decoupled form.

The present work explores the following issues:

1. Highlight some conditions for which two-stage preconditioners could be effective in porous media applications.
2. Analyze the potential that two-stage preconditioners based on AMG pressure solutions have in comparison with generalized AMG methods for dealing "directly" with coupled linear systems.
3. Compare AMG solution strategies for fully implicit formulations with those for iterative coupling procedures.

Two-stage Preconditioners

Efforts to develop general and efficient solvers for coupled linear systems with a mixed parabolic, elliptic character have been quite extensive in fluid dynamics applications[14,19]. However, in the setting of porous media flow, the effort has been minimal due to the complexity of characterizing general flow and well operation situations in a reservoir.

Behie and Vinsome appear to be the first to consider combinative preconditioners for generating decoupled preconditioners in reservoir engineering applications[1]. A minor change to the concept that sought to incorporate partial saturation information was later proposed by Behie and Forsyth[20]. A decade later, Wallis proposed a two-stage preconditioning approach motivated by the idea of using an IMPES-like preconditioner suitable for coupled systems arising in fully implicit formulations[21]. This work is particularly inspired by his previous results for constrained pressure residuals (CPR) accelerations[4]. Motivated by the construction of robust physics-based preconditioners for fully-implicit systems, Klie performed a broader analysis and other more robust extensions to the two-stage preconditioner originally given by Wallis[13]. Some of these results are treated by Dawson et al. in the context of Newton-Krylov methods for porous media flow[22].

The implementation of two-stage preconditioners has been increasingly reported in several challenging oil applications[23-25]. Moreover, this technology is currently being considered in initiatives for the development of a new generation of reservoir simulators[6,8].

To further motivate and fix ideas on two-stage preconditioners, consider the coupled system arising from a fully implicit formulation,

$$Ax = \begin{pmatrix} A_{pp} & A_{ps} \\ A_{sp} & A_{ss} \end{pmatrix} \begin{pmatrix} x_p \\ x_s \end{pmatrix} = \begin{pmatrix} b_p \\ b_s \end{pmatrix} = b, \tag{1}$$

with $A_{pp} \in \mathbb{R}^{np \times np}$ representing the pressure block coefficients, $A_{ss} \in \mathbb{R}^{ns \times ns}$ representing the saturation/concentration (non-pressure) block coefficients, and the rectangular blocks $A_{ps} \in \mathbb{R}^{np \times ns}$ and $A_{sp} \in \mathbb{R}^{ns \times np}$ representing the respective coupling coefficients. Unless adaptive techniques are

employed, $ns = k \cdot np$ for $k = 1, 2, \ldots$ and np coincides with the number of active grid cells of the discretized simulation domain. In general, A is highly non-symmetric and indefinite. Moreover, positive eigenvalues come from the A_{pp} block contribution, negative ones usually from the A_{ss} block contribution.

The block A_{pp} has the structure of a purely elliptic problem, whereas the block A_{ss} holds coefficients of a set of coupled (convective-diffusive) parabolic problems. From the algebraic standpoint, the blocks A_{pp} (and the main diagonal blocks of $-A_{ss}$) usually fulfill the following properties: (1) diagonal dominance, (2) positive diagonal and non-positive off-diagonal entries (Z-matrix property), (3) irreducibility[14]. It should be noted that the AMG treatment of negative M-matrices requires appropriate row scaling to be performed before the coarsening process.

Strict diagonal dominance of A_{pp} is generally achieved by the contribution of bottom hole pressures. Also, under small changes of formation volume factors and flow rates between adjacent blocks, we can expect A_{pp} to be nearly symmetric. As a result, A_{pp} may be also positive stable (i.e. the real part of all eigenvalues is positive), which is a relaxed version of symmetric positive definiteness.

It is important to note that many factors may affect this set of desirable algebraic properties. For instance, large time steps or small pore volumes may negatively affect the diagonal dominance of the diagonal blocks of A_{ss} and, therefore, compromise the convergence of the iterative solver. Another negative factor may also come from the well production term contribution in a compressible system. In such a case, diagonal dominance of the pressure and saturation blocks may be lost (i.e., rowsums may become negative), and convergence may not be guaranteed.

Effective extraction of the pressure block is critical to the application of a decoupling procedure that will weaken the original coupling and alternatively provide some extra preconditioning effect to the overall procedure. More precisely, given the system (1), we would like to find *decoupling operators* D_1 and D_2 and transform

$$D_1^{-1} A D_2^{-1} = \tilde{A} = \begin{pmatrix} \tilde{A}_{pp} & \tilde{A}_{ps} \\ \tilde{A}_{sp} & \tilde{A}_{ss} \end{pmatrix}. \qquad (2)$$

Further below we discuss some possible choices for D_1 and D_2 to ensure a set of desirable properties for the implementation of the two-stage preconditioner. Given $\tilde{A}$ and a residual vector, $r = (r_p, r_s)^t$, we define the two-stage preconditioner as follows:

1. Solve the pressure system: $\tilde{A}_{pp} \delta_p = r_p$;
2. Compute the new residual: $\hat{r} = r - \tilde{A} \begin{pmatrix} \delta_p \\ 0 \end{pmatrix}$;
3. Precondition and correct: $\delta = M^{-1} \hat{r} + \begin{pmatrix} \delta_p \\ 0 \end{pmatrix}$.

Here, $\delta = (\delta_p, \delta_s)^t$ denotes the correction obtained after the two stages. Hence, the action of the complete two-stage preconditioner, M_{2s} say, on r can be expressed as

$$\delta = M_{2s}^{-1} r = M^{-1} \left[I - \left(\tilde{A} - M \right) \begin{pmatrix} \tilde{A}_{pp}^{-1} & 0 \\ 0 & 0 \end{pmatrix} \right] r. \qquad (3)$$

Several comments are in order:

1. The computation of the exact inverse of $\tilde{A}_{pp}$ would be unlikely in large-scale settings. An iterative procedure for solving pressures will be necessary to obtain an approximation of this operator. This fact will give rise to a nested iterative procedure for which special care must be taken to tune up inner and outer tolerances.
2. There is no need to have explicit expressions for all terms involved in (3). For the sake of efficiency, the action of the two-stage preconditioner must be based on a concatenation of matrix-vector products and implicit backward and forward substitutions when factorized forms are available. Likewise, specialized (restriction and prolongation) operators can be easily constructed to perform the action of the inverse pressure block or other solution block onto a vector[13].
3. The preconditioner M^{-1} acts as the second stage within the two-stage preconditioning process. Note that, if $M = I$, the two-stage preconditioner reduces to the application of

$$\delta = \begin{pmatrix} \tilde{A}_{pp}^{-1} & 0 \\ \tilde{A}_{sp} \tilde{A}_{pp}^{-1} & I \end{pmatrix} r.$$

This means that the preconditioned correction δ will disregard the saturation block solution and only the coupling between pressures and saturation would enter into the preconditioner process. Hence, a good practice is to allow M to incorporate error solution components associated with the non-pressure block solution. An ILU or block SOR type of iteration may be a reasonable choice in moderate flow situations. In this sense, M plays the role of a global smoother for a coarse-grid iteration.

4. The two-stage preconditioner M_{2s}^{-1}, given by (3), is the inverse of $\tilde{A}$ on the subspace spanned by the columns of $\begin{bmatrix} I & 0 \end{bmatrix}^t$, that is,

$$M_{2s}^{-1}\tilde{A}\begin{pmatrix} I \\ 0 \end{pmatrix} = \begin{pmatrix} I \\ 0 \end{pmatrix}.$$

The aforementioned two-stage preconditioner may be seen as a building block for constructing more robust preconditioning strategies, namely, multi-stage preconditioners[6,13]. For instance, inclusion of saturation/concentrations can be integrated in an additive or multiplicative fashion[14]:

$$v = M_{2s+}^{-1} r = M^{-1}\left[I - \left(\tilde{A} - M\right)\begin{pmatrix} \tilde{A}_{pp}^{-1} & 0 \\ 0 & \tilde{A}_{ss}^{-1} \end{pmatrix}\right] r,$$

$$v = M_{2s*}^{-1} r = M^{-1}\left[I - \left(\tilde{A} - M\right)\begin{pmatrix} \tilde{A}_{pp}^{-1} & -\tilde{A}_{pp}^{-1}\tilde{A}_{ps}\tilde{A}_{ss}^{-1} \\ 0 & \tilde{A}_{ss}^{-1} \end{pmatrix}\right] r.$$

For reasons of efficiency, the saturation/concentrations and coupling blocks may be optionally replaced by an even sparser representation, that is, $\tilde{B}_{ss}^{-1} \approx \tilde{A}_{ss}^{-1}$ and $\tilde{B}_{ps}^{-1} \approx \tilde{A}_{ps}^{-1}$.

Decoupling Operators

Decoupling operators are necessary to weaken the existing coupling between pressure and non-pressure blocks. This operation can be seen as a preprocessing step before proceeding with the rest of the two-stage preconditioning procedure. It is important to mention that the area of decoupling operators has been addressed by some authors with certain spinoffs in the design of two-stage preconditioners[13,23,24,26].

Given system (1) and the operators introduced in (2), we identify five desirable properties to be fulfilled by decoupling operators:

1. $Cond\left(\tilde{A}\right) \le Cond\left(A\right)$, i.e., the decoupling operator reduces the condition number of the full system;
2. $Cond\left(\tilde{A}_{pp}\right) \le Cond(A_{pp})$ and $Cond\left(\tilde{A}_{ss}\right) \le Cond(A_{ss})$, i.e., the condition number of each of the main diagonal blocks should be improved;
3. $\left\| I - \begin{pmatrix} \tilde{A}_{pp}^{-1} & 0 \\ 0 & \tilde{A}_{ss}^{-1} \end{pmatrix}\tilde{A} \right\|_\infty \le \left\| I - \begin{pmatrix} A_{pp}^{-1} & 0 \\ 0 & A_{ss}^{-1} \end{pmatrix} A \right\|_\infty$, i.e., the coupling strength of the blocks of $\tilde{A}$ is weaker than that of the blocks of A;
4. $\tilde{A}_{pp}$ and the main diagonal blocks of $\tilde{A}_{ss}$ should be M-matrices, ensuring that the resulting decoupled blocks are amenable to a convergent iterative solution;
5. The decoupled system $\tilde{A}$ should be computationally inexpensive to obtain.

These conditions may be satisfied at different degrees and are not mutually exclusive. The use of a two-side decoupling strategy with D_1 and D_2 may be convenient to preserve symmetry. However, in most cases reported in the literature, D_2=I and D_1 is defined in terms of the main diagonal of the blocks involved. For instance, for the case $ns = np$ (i.e., a two-phase flow case), we can define D_1 as[13,26,29]

$$D_1 = \begin{pmatrix} D_{pp} & D_{ps} \\ D_{sp} & D_{ss} \end{pmatrix} = \begin{pmatrix} diag(A_{pp}) & diag(A_{ps}) \\ diag(A_{sp}) & diag(A_{ss}) \end{pmatrix} \quad (4)$$

Hence,

$$D_1^{-1}A = \begin{pmatrix} \tilde{A}_{pp} & \tilde{A}_{ps} \\ \tilde{A}_{sp} & \tilde{A}_{ss} \end{pmatrix} = \begin{pmatrix} \Delta^{-1} & 0 \\ 0 & \Delta^{-1} \end{pmatrix} \bullet \begin{pmatrix} D_{ss}A_{pp} - D_{ps}A_{sp} & D_{ss}A_{ps} - D_{ps}A_{ss} \\ D_{pp}A_{sp} - D_{sp}A_{pp} & D_{pp}A_{ss} - D_{sp}A_{ps} \end{pmatrix}$$

with $\Delta = D_{pp}D_{ss} - D_{ps}D_{sp}$. It can be easily checked that the main diagonal entries of $\tilde{A}_{pp}$ and $\tilde{A}_{ss}$ are all equal to 1. Additionally, the main diagonal entries of $\tilde{A}_{ps}$ and $\tilde{A}_{sp}$ are all equal to 0. Therefore, we can expect that the degree of coupling of the off-diagonal blocks has been reduced to some extent

This decoupling operator was proposed by Bank et al.[27] as the *Alternate-Block Factorization (ABF)* procedure and is subject to further analysis in other works[28,29]. The ABF operator was successfully applied by Klie[13] for two-phase flow problems where Properties 1-5 from above were shown to be strongly satisfied under mild conditions (e.g., if blocks A_{ps} and A_{sp} satisfy the M-matrix condition). This approach was also pursued in groundwater applications[26].

Note that the application of this decoupling procedure is simple to implement since it involves a rescaling and linear combination of the underlying data structure holding the coupled matrix. This can be realized by overwriting the underlying data structure and providing additional storage for a few vectors that hold the original block diagonals of A.

Note also that the meaning of the ABF decoupling gets particularly transparent if we consider a point-wise re-ordering of variables. Then,

$$A = \begin{pmatrix} A_{(1,1)} & \cdots & A_{(1,n)} \\ \vdots & \ddots & \vdots \\ A_{(n,1)} & \cdots & A_{(n,n)} \end{pmatrix} \quad (5)$$

with $n=np$ denoting the total number of mesh points and $A_{(i,j)}$ representing the block of couplings between the variables sitting at point i and those sitting at point j. Observing that, with respect to the point-wise numbering,

$$D_1 = \begin{pmatrix} A_{(1,1)} & & \\ & \ddots & \\ & & A_{(n,n)} \end{pmatrix},$$

we immediately see that $D_1^{-1}A$ simply corresponds to a block-diagonal scaling.

Unfortunately, there are two potential drawbacks to the decoupling (4)[24]. In practice, the resulting pressure matrix $\tilde{A}_{pp}$ may be strongly nonsymmetric compared to A_{pp}. This fact may introduce particular difficulties in solving the pressure system with AMG. Another problem comes from the handling of nonlinear residuals within a Newton procedure. The use of left decoupling operators as (4) may lead to oversolving the Jacobian system since the linear stopping criteria should be adequately weighted in terms of D_1.

A more conservative decoupling strategy (also known as quasi-IMPES) preserves most of the original structure of the coupled system[23,24]. More precisely, it defines the decoupling operator as:

$$D_1 = \begin{pmatrix} I & D_{ps} \\ 0 & D_{ss} \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & D_{ss}^{-1} \end{pmatrix} = \begin{pmatrix} I & D_{ps}D_{ss}^{-1} \\ 0 & I \end{pmatrix}.$$

Thus,

$$D_1^{-1}A = \begin{pmatrix} I & -D_{ps}D_{ss}^{-1} \\ 0 & I \end{pmatrix} \begin{pmatrix} A_{pp} & A_{ps} \\ A_{sp} & A_{ss} \end{pmatrix} = \begin{pmatrix} A_{pp} - D_{ps}D_{ss}^{-1}A_{sp} & A_{ps} - D_{ps}D_{ss}^{-1}A_{ss} \\ A_{sp} & A_{ss} \end{pmatrix}.$$

Clearly, this operation preserves the coefficients of the saturation equation. Additionally, the pressure equation rows are not directly scaled but it would comprise contribution from the A_{sp} block which generally has the M-matrix property. Yet, the main diagonal of the resulting $\tilde{A}_{ps}$ has been zeroed out. Also, this form is less costly to perform than the ABF decoupling. As a disadvantage, Properties 1-3 stated above are weakly fulfilled[13] and a stronger preconditioning effect may be demanded from the two-stage preconditioner, namely, the M^{-1} component of (3). For illustration, note that if

$$\tilde{A}_{pp}^{QI} = A_{pp} - D_{ps}D_{ss}^{-1}A_{sp} \text{ and}$$
$$\tilde{A}_{pp}^{ABF} = \Delta^{-1}\left(D_{ss}A_{pp} - D_{ps}A_{sp}\right)$$

denote the resulting pressure systems after having applied the quasi-IMPES and ABF decoupling operator, respectively, then

$$\tilde{A}_{pp}^{ABF} = \Delta^{-1}D_{ss}\tilde{A}_{pp}^{QI}.$$

Therefore, the ABF decoupling performs a more aggressive row scaling than the quasi-IMPES decoupling operator, thus generating a stronger nonsymmetric pressure system. However, in many practical situations[13] we can find that $Cond\left(\tilde{A}_{pp}^{ABF}\right) < Cond\left(\tilde{A}_{pp}^{QI}\right)$.

This illustrates the type of tradeoff that must be carefully evaluated for solving the resulting pressure system and constructing the two-stage preconditioning strategy.

Algebraic Multigrid (AMG) for Scalar and Coupled Applications

Scalable linear solvers are required to efficiently perform large-scale numerical simulations. Scalability, however, requires hierarchical algorithms which ensure a rapid reduction of both short and long range error components. A breakthrough, and certainly one of the most important advances during the last three decades, was due to the (geometric) multigrid principle[30].

Algebraic multigrid (AMG) solvers attempt to combine the advantages of geometric multigrid with those of easy-to-use plug-in solvers[32]. In contrast to geometric multigrid, which operates on a pre-defined hierarchy of grids, AMG directly operates on the linear matrix problem which corresponds to a given discretization. Since the explicit construction of a multilevel hierarchy is part of the AMG algorithm (i.e., is invisible to a user), a corresponding solver is easy to integrate into existing simulation packages. For these reasons, algebraic multigrid solvers have become quite popular as the basic solver in various industrial simulation codes. In reservoir simulation, this was driven by ever increasing problem sizes, complex structures, heterogeneities, multiphase flows and wells which made clear the limits of classical one-level solvers.

AMG automates the complete process of creating a multilevel hierarchy. Based only on the concept of strength of connectivity between variables (in the simplest case defined by the size and sign of matrix coefficients) and by exploiting the so-called Galerkin-principle, classical AMG[31,32,16] directly mimics geometric multigrid. Sets of coarse-level variables are recursively defined by requiring the fine-to-coarse connectivity to be as strong as possible under the constraint that the sets of coarse-level variables themselves are maximally independent (w.r.t. their *strong* couplings). Interpolation between levels is recursively defined based on the respective matrices with weights being proportional to the

matrix entries, a_{ij} [†].

The original AMG approach is effectively restricted to particular classes of problems, an important one being the class of linear algebraic systems with (approximately) weakly diagonally dominant M-matrices. Problems like this widely occur in connection with discretized *scalar* second-order elliptic PDEs. In such cases, AMG is very mature and can handle millions of variables much more efficiently than any one-level method and is especially suited for unstructured grids.

However, most important industrial as well as scientific applications are far from being of the appropriate type and require generalized AMG approaches. More general matrix problems do not only occur in connection with *coupled* PDE systems, also *scalar* PDE applications lead to strict M-matrices only in simple situations. Often large positive off-diagonal entries occur, or particular rows are far from being weakly diagonally dominant. More sophisticated AMG methods are needed to make the overall approach more generally applicable and, through this, make it more interesting for industrial applications.

During the last years, systematic extensions of the classical AMG approach have been investigated. In particular, starting from early studies (mainly for semiconductor process and device applications)[33,34], a general AMG framework has been developed which is highly flexible in exploiting additional (e.g. user-provided) information in order to adjust its algorithmic components to specific requirements of a given problem class[17]. Most features of this framework have been realized in the software library SAMG[35]. The basic idea relies on the introduction of some auxiliary (sparse) control matrix, the so-called "primary" matrix, generally denoted by *P*: Rather than defining the connectivity between variables based directly on the entries of the given matrix *A*, it is defined via the entries of *P*.

Clearly, in case of *scalar* PDEs and dicretization matrices which are (approximately) M-matrices, the standard choice would be *P*=*A*, that is, *P* represents the connectivity structure between variables as provided directly by the user. In more general scalar applications, one may, for instance, define *P* based on geometric distances of the variables (if known), or on alternate (for instance, less accurate but simpler) discretizations. Interpolation between levels may or may not be defined based on the matrix entries of *P* rather than those of *A*. Obviously, there are various possible combinations and which one is the best, strongly depends on certain characteristics of a given application.

In the following, we consider *coupled PDE systems*. To be more specific, assume that we want to solve for *nf* physical functions such as, for instance, the pressure, the saturation of a particular species, or a velocity component. (For historical reasons, in the context of AMG, we usually call such functions "unknowns".) Often, the solution of such coupled PDE systems is done in stages, each of which requires a scalar sub-problem to be solved and for which scalar AMG approaches can immediately be applied. Whether or not such an "indirect" solution approach is efficient, depends strongly on the situation. However, in many cases it will be more efficient to treat the coupled system by some "direct" AMG approach. There are various ways to do this, some ideas are outlined in the following.

"Unknown-based" AMG. Assuming the variables to be ordered unknown-wise, the discretization matrix has the form

$$A = \begin{pmatrix} A_{[1,1]} & \cdots & A_{[1,nf]} \\ \vdots & \ddots & \vdots \\ A_{[nf,1]} & \cdots & A_{[nf,nf]} \end{pmatrix},$$

and a general primary matrix would look like

$$P = \begin{pmatrix} P_1 & & \\ & \ddots & \\ & & P_{nf} \end{pmatrix}.$$

Analogously to the scalar case, the connectivity structure reflected by the auxiliary matrix P_i is used to coarsen the *i*-th unknown. Moreover, interpolation is kept separate for the different unknowns, that is, variables corresponding to the *i*-th unknown, say, are interpolated from variables of the same type only. Concrete weights may, for example, be based on the entries of P_i, $A_{[i,i]}$ or on geometric distances (if available).

Since, independent of the concrete choice of *P*, different unknowns are coarsened and interpolated separately[‡], this kind of approach is called the "unknown-based" approach. Clearly, for this approach to make sense, P_i should reflect the physical connectivity between variables corresponding to the *i*-th unknown reasonably well. For instance, for all diagonal blocks of *A* which are close to being weakly diagonally dominant M-matrices, a standard choice would be $P_i = A_{[i,i]}$. However, analogous to the scalar case, various other choices are possible.

Unknown-based AMG approaches are both simple and powerful for many classes of applications, at least if the cross-couplings between unknowns do not exceed a certain strength.

[†] The whole process of coarsening is defined recursively. For simplicity, here and in the following, we refer only to the very first step of coarsening a given problem. Hence, all indices regarding coarser levels are omitted.

[‡] We point out that this does *not* mean that all unknowns are treated independently of each other. In fact, on each level of the multigrid hierarchy the Galerkin operators preserve cross-couplings between different unknowns.

"Point-based" AMG. In more complex situations, so-called "point-based" AMG approaches may be more promising. For some rough description, we assume the set of variables to be re-numbered point-wise, resulting in the representation of A as shown in (5). We require all unknowns to be defined at the same grid (i.e., no staggered grids allowed). However, not all unknowns need to be defined at each point, allowing for adaptive strategies such as those realized in the AIM approach[36].

In contrast to unknown-based approaches, point-based ones are controlled by primary matrices P which are defined on the level of "points" rather than variables, and all unknowns are coarsened based on that same P. Hence, the connectivity structure defined by P should represent the connectivity of *all* unknowns in the given PDE system sufficiently well.

Often, the connectivity inherent to one of the given PDE system's unknowns, k say, can be regarded as being representative also for the other unknowns of the full system. In such cases, provided that the k-th unknown is defined at each point of the mesh, a possible choice is

$$P = A_{[k,k]} \ .$$

In reservoir modelling, the pressure matrix, A_{pp}, can often be used for that purpose. In other applications, depending on certain characteristics of the class of PDEs under consideration, potential choices are $P = (p_{ij})$ with, e.g.,

$$p_{ij} = -\| A_{(i,j)} \| \quad \text{or} \quad p_{ij} = -1/dist(i,j)^2$$

for $i \neq j$ and

$$p_{ii} = -\sum_{j \neq i} p_{ij} \ .$$

While the first choice leads to what is sometimes called „block approach", the second choice (requiring the coordinates of points to be available) is closely related to geometric coarsening. Finally, one can imagine that P is defined based on some natural physical quantity for which there is no obvious equation contained in the given system of PDEs. An example of such a situation would be the pressure in the context of the Navier-Stokes equations.

Once a primary matrix has been selected, there are again various ways to define interpolation. In particular, interpolation may be different for each physical unknown (e.g., based on the original matrix blocks $A_{[k,k]}$), it may be the same for each unknown (e.g., based on the primary matrix P or on coordinates), or it may be defined based on the pointwise block couplings.

Primary matrices are either defined internally to SAMG (i.e. is constructed automatically) or they are user-provided. The latter option makes particular sense in applications where SAMG cannot construct a reasonable primary matrix automatically, based solely on algebraic information contained in the given linear system. However, in many such cases, a user may still be able to define a reasonable matrix himself, based on the underlying physics.

A general framework. Formally, by combining the above ideas related to coarsening and interpolation, one obtains a very flexible and general framework to define concrete AMG approaches which can be adapted to the specific requirements of many PDE systems. It seems clear, however, that none of these algorithms will work satisfactorily for *all* systems of PDEs. Instead, different approaches may be required for different classes of applications, for instance, from fluid dynamics, oil reservoir simulation and semiconductor process and device simulation.

Independent of this, a principle advantage of point-based approaches (compared to unknown-based ones) is that they are much more flexible in taking strong cross-couplings between unknowns into account. In particular, smoothing can be performed in a point block sense (e.g. block Gauss-Seidel or block ILU). Furthermore, if required, interpolation can also be performed block-wise as naturally induced by the system's point block coupling.

Although AMG has become fairly mature for many different classes of problems, we point out that there are still various open questions. The general approach is still continuously being enhanced and extended to cover more and more applications.

Iterative Coupling

Many new techniques have been developed to cover the efficiency gap between fully implicit methods and IMPES formulations. *Iterative coupling* is one of these methods. The main objective of the iterative coupling method is to achieve both accuracy and efficiency at the same time.

By means of the iterative coupling we avoid solving pressure and saturation simultaneously as in fully implicit formulations. On the other hand, iterative coupling differs from sequential methods in that saturations are not solved implicitly.

Iterative coupling is an operator-splitting technique that decouples the multiphase system into pressure and saturation equations. At each time step a series of iterations are computed that involve solving both pressure and a linearized saturation equation using specific tolerances that are iteration-dependent and sequential. Following convergence of an iteration, phase concentrations and mass balances are checked to determine if a time-step convergence is satisfied. If not, nonlinear coefficients are updated and iteration tolerances are tightened. The sequential iteration is then repeated. This concept is illustrated in Figure 1.

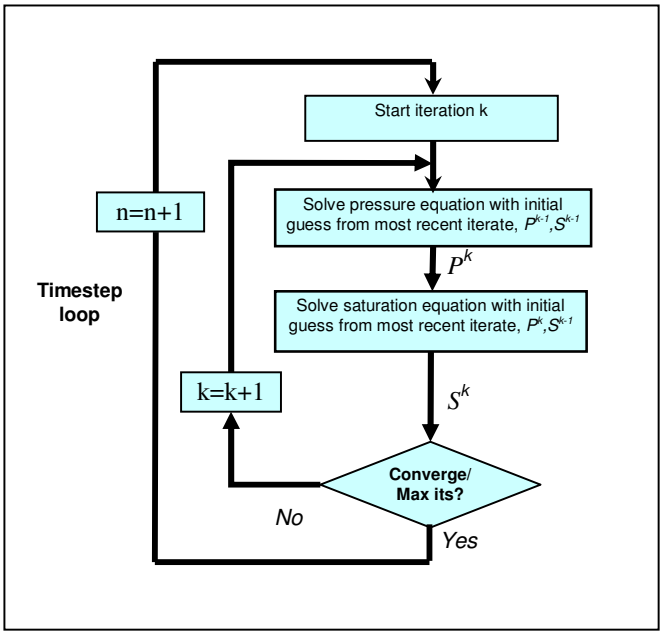

Fig. 1: The iterative coupling approach for solving pressures and saturations

Figure 2 shows how the iterative coupling reproduces very well the production response for the SPE 10th Comparative Project[37] using the fully implicit formulations of IPARS[38] and Eclipse.

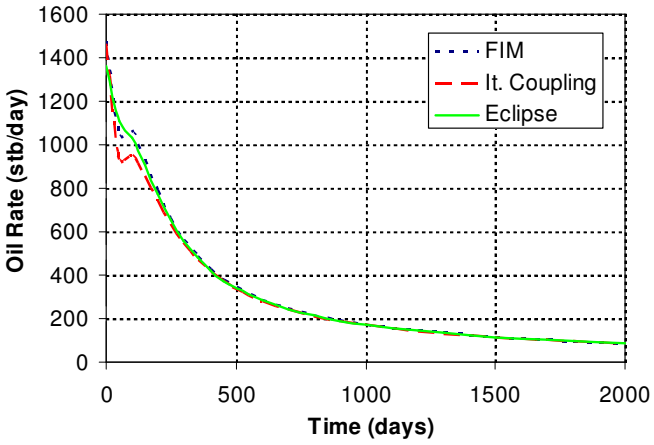

Fig. 2: Production response using fully implicit and iterative coupling implementation in IPARS against Eclipse for the SPE 10th Comparative project

Numerical Experiments

In the following we consider five test cases based on different sections and upscaled versions of the SPE 10th Comparative Project[37]:

1. 60x220 (1st slice of the non-fluvial reservoir, Tarbert formation);
2. 60x220 (1st slice of the fluvial reservoir, Upper Ness formation);
3. 17x12x44 (upscaled version of the original data, including Tarbert and Upper Ness formations);
4. 25x15x55 (Upper Ness formation, upscaled version);
5. 50x30x110 (Upper Ness formation, upscaled version).

For all cases we define one water injection well at the center of the reservoir, and four production wells at the four corners of the reservoir. All are bottomhole pressure specified. For each of the considered problem sizes, we performed tests for an oil-water and a black-oil system. For the black-oil system, the PVT and saturation-dependent curves data were adapted from the SPE 9th Comparative Project.

We compare four different solver approaches:

1. **AMG-FIMS**: Direct use of AMG as a solver for the coupled system (1). More precisely, a point-based approach is used, with the primary matrix given by the pressure block, A_{pp}. Correspondingly, the matrix is reordered in a point-block fashion (5).
2. **AMG-FIM2SP**: AMG is used as a solver just for the pressure system $\tilde{A}_{pp}$ resulting from the decoupling and restriction process of the two-stage preconditioner given by (3). In the tests here, we apply the quasi-IMPES decoupling (to favour AMG convergence) and define M^{-1} by one LSOR iteration[23].
3. **AMG-ICS**: AMG is used for solving the pressure system resulting from the iterative coupling. Saturations are solved by GMRES preconditioned with line SOR.
4. **AMG-ICP**: AMG is used as a preconditioner for the pressure system resulting from the iterative coupling, accelerated by a conjugate gradient method. Saturations are solved by GMRES preconditioned with line SOR.

While for oil-water simulations all four solver approaches have been implemented and compared with each other, for black-oil simulations we have currently implemented only the first two approaches.

The AMG employed here is the SAMG software developed at the Fraunhofer Institute for Algorithms and Scientific Computing[35]. SAMG was added to the suite of different solvers already provided by IPARS[38]. We consider the following default settings for SAMG[35] in all cases that were analyzed:

- Pre- and post-smoothing steps at each level consisting of one Gauss-Seidel iteration;
- V-cycle coarsening/refinement pattern consisting of a maximum of 25 levels;
- Sparse Gaussian elimination at the very coarsest level.

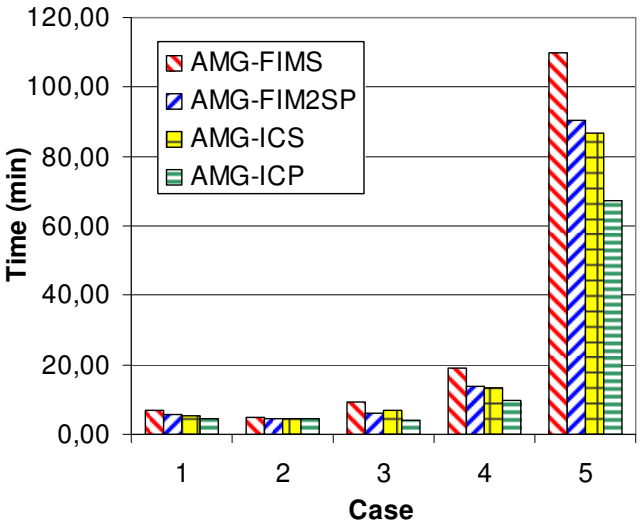

Fig. 3: Timings for oil-water simulations using AMG-based solver approaches

Figure 3 summarizes timings for oil-water simulations in the Cases 1-5. Simulations were carried out for 2000 days except for Case 5 which was simulated for 500 days. We first observe that, for all test cases and solver approaches considered here, AMG is very efficient as a basic linear solver. For multiphase fully implicit formulations, the two-stage preconditioning seems to offer a slightly more efficient possibility as compared with directly tackling the solution of the coupled linear system with AMG. The results of the iterative coupling approach demonstrate that this approach is even more promising than the other ones. This is, in particular, true for the AMG-ICP variant. The results in Figure 4 clearly show the substantial gain in performance in switching from multilevel ILU[23,39] to AMG.

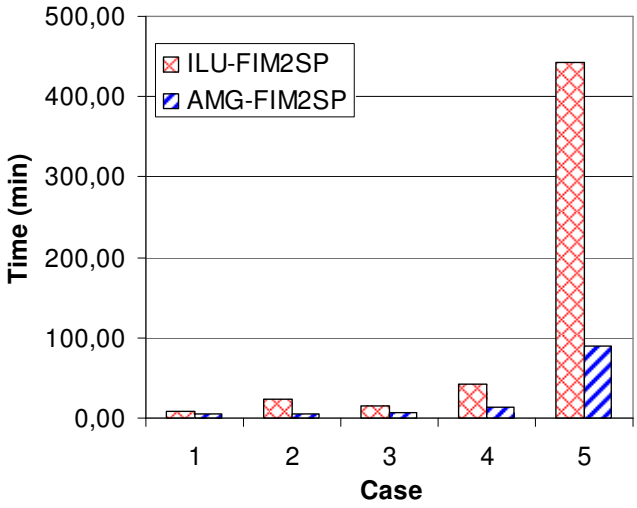

Fig. 4: Timings for the two-stage preconditioning approach comparing AMG with multilevel ILU for solving the pressure

Figure 5 shows timings for the black-oil case. There appears to be no striking difference between applying AMG directly to the coupled system and applying it to the pressure solution in the two-stage process, with a slight preference for the latter.

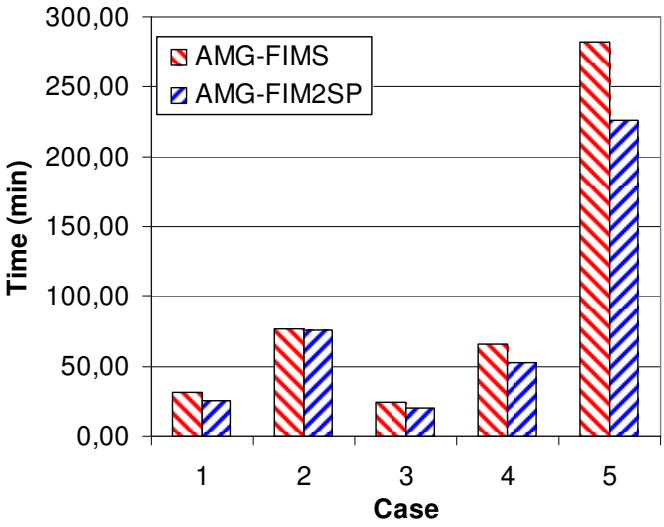

Fig. 5: Timings for black-oil simulations

Conclusions and Outlook

The numerical experiments compare the performance of four strategies all of which use AMG as a basic solver component. The results are very promising and seem to indicate that the "direct" application of AMG (point-based approach AMG-FIMS) is the least efficient.

However, more testing is certainly needed to draw some well-founded and final conclusion. In particular, the influence of wells, time step size, DPDP or otherwise physically more involved situations need carefully to be investigated. In fact, some recent studies with AMG have demonstrated the particular potential and effectiveness of direct AMG approaches in dealing with more complex, real-life reservoir simulations[36]. In these cases, more powerful and robust AMG approaches have turned out to be beneficial.

To be more specific, we want to recall that the effectivity of two-stage preconditioners is strongly tied to the decoupling, the pressure solution (i.e., the properties of the matrix block $\tilde{A}_{pp}$) and the preconditioner M^{-1}. Clearly, as long as $\tilde{A}_{pp}$ is "sufficiently elliptic" (so that standard scalar AMG for computing the pressure works effectively), and a simple and cheap *one-level* preconditioner is sufficient to ensure rapid convergence of the overall two-stage process, AMG-FIMS can hardly be more efficient than the two-stage process (AMG-FIM2SP). In fact, this is observed in all cases tested her: already in its simplest form (with just plain Gauss-Seidel smoothing) scalar AMG works efficiently as a pressure solver, and the overall pre-conditioner corresponds to just a single step of LSOR. Hence, it is not really surprising that AMG-

FIMS is slightly more expensive than AMG-FIM2SP.

However, this cannot always be expected to be true. In general, stronger and more costly preconditioners M^{-1} as well as more robust smoothers will be required. Most importantly, however, $\tilde{A}_{pp}$ may have properties which are "less favorable" for an efficient treatment by AMG. In fact, even if the original pressure matrix A_{pp} is perfectly elliptic, algebraic manipulations such as those done in the decoupling process, may negatively influence the ellipticity to an extent which may cause AMG to perform less efficiently. Depending on the situation (in particular, the well production term contribution), row sums may get increasingly negative and, eventually, $\tilde{A}_{pp}$ may become indefinite (with one or more negative eigenvalues). Analysis of the interplay between all relevant components deserves further efforts over a wide range of multiphase flow reservoir situations. The risk of a substantial performance drop can certainly be reduced by a careful choice of the decoupling. However, a rigorous analysis, taking all aspects into account, seems fairly difficult. It is known that even simple re-scalings (multiplying by a diagonal matrix from the left) may substantially influence AMG's performance. Re-scalings from the right are even much more dangerous since they may completely change the elliptic character of a matrix.

The fact that direct AMG approaches do not perform any algebraic manipulations on a given system makes them particularly interesting and promising compared to two-stage approaches. In fact, direct AMG is always applied to the unmodified system, with the coarsening process being driven by the fully elliptic pressure block A_{pp}. Moreover, for various reasons, convergence of the direct AMG approach is expected to be less dependent on strong cross-couplings between pressure and saturation. This is due to its hierarchical nature combined with its potential in employing strong block-wise smoothing and interpolation.

The authors are interested in pursuing the following research paths:

1. Define adaptive and physically-based strategies for the use of two-stage preconditioners. This goal is motivated by the fact that the strength of each of the preconditioning components described herein can be controlled in accordance with the timestep.
2. Perform a more detailed analysis of decoupling strategies that may be more amenable for the efficiency of AMG pressure solutions.
3. Perform a more detailed analysis of direct AMG approaches for more complex physical situations, larger time steps, etc. The recent results of Clees and Ganzer[36] will serve as a starting point.
4. Perform analysis on parallel and multicore architectures for exploiting different degree of granularity for two-stage preconditioners and iterative coupling.
5. Analysis of iterative coupling in the setting of multiblock implementations that may involve the simulation of different scales, physics and numerical formulations in different regions of the reservoir domain.

Acknowledgements

The authors would like to thank Tareq Al-Shaalan from Aramco for sharing valuable discussions on two-stage preconditioning methods.

References

1. Behie G. and Vinsome, P.: *"Block Iterative Methods for Fully Implicit Reservoir Simulation"*, *SPEJ*, pp 658-668, 1982.
2. Behie, A. and Forsyth, P.A.: *"Multigrid Solution of the Pressure Equation in Reservor Simulation"*, *SPE AIME* SPE 10492, pp. 623-636, 1983.
3. Dendy Jr., J.E., McCormick, S.F., Ruge, J.W., Russell, T.F. and Schaffer, S.: *"Multigrid Methods for Three-Dimensional Petroleum Reservoir Simulation"*, SPE Symposium on Reservoir Simulation, SPE 18409, Houston, TX, Feb. 6-8, 1989.
4. Wallis, J.R., Kendall, R.P. and Little, L.E.: *"Constrained Residual Acceleration of Conjugate Residual Methods"*, SPE Reservoir Simulation Symposium, SPE 13536, Dallas, Texas, Feb 10-13, 1985.
5. Mattax, C. and Dalton, R.: *"Reservoir Simulation"*, SPE Monograph Series, 1990.
6. Cao, H., Tchelepi, H.A. and Wallis, J.: *"Parallel Scalable Unstructured CPR-type Linear Solver for Reservoir Simulation"*, SPE Annual Technical Conference and Exhibition, SPE 96809, Dallas, Texas, Oct. 9-12, 2005.
7. Dogru, A.H., Sunaidi, H.A., Fung, L.S., Habiballah, W.A., Al-Zamel, N. and Li, K.G.: *"A Parallel Simulator for Large-Scale Reservoir Simulation"*, SPE Reservoir Evaluation and Engineering, SPE N. 75805-PA,Vol.5, N. 1, pp. 11-23. 2002.
8. DeBaun, D. et al.: *"An Extensible Architecture for Next Generation Scalable Parallel Reservoir Simulation"*, SPE Reservoir Simulation Symposium, SPE 93274, The Woodlands, Texas, Jan 31-Feb 2, 2005.
9. Watts J.W. and Shaw, J.S.: *"New Method for Solving Implicit Reservoir Simulation Matrix Equation"*, SPE Reservoir Simulation Symposium, SPE 93068-MS, The Woodlands, Texas, Jan 31-Feb 2, 2005.
10. Fung, L.K, Su, H., Tan, C.T., Hemanthkumar, K. and Pita, J.: *"Fully-Implicit Fully-coupled Well Model for Parallel Mega-Cell Reservoir Simulation"*, SPE Technical Symposium of Saudi Arabia Section, SPE 106331, Dhahran, Saudi Arabia, May 14-16, 2005.
11. Fjerstad, P.A., Sikandar, A.S., Cao, H. and Liu, J.: *"Next Generation Parallel Computing for Large-Scale Reservoir Simulation"*, SPE International Improved Oil Recovery Conference in Asia Pacific, SPE 97358, Kuala Lampur, Malaysia, Dec. 2005.
12. Becker, B.L., Usadi, A.K., Ray, M.B. and Diyankov, O.V.: *"Next Generation Reservoir Simulation Using Russian Linear Solvers"*, SPE Russian Oil and Gas Technical Conference and Exhibition, SPE 103578-MS, Moscow, Russia, Oct 3-6, 2006.
13. Klie, H.: *"Krylov-Secant Methods for Solving Large Scale Systems of Coupled Nonlinear Parabolic Equations"*, PhD thesis, Dept. of Computational and Applied Mathematics, Rice University, Houston, TX, 1996.
14. Axelsson, O.: *"Iterative Solution Methods"*, Cambridge University Press, 1994.

15. Chen, K.: *"Matrix Preconditioning Techniques and Applications"*, Cambridge University Press, 2005.
16. Stüben, K.: *"An Introduction to Algebraic Multigrid"*, Appendix in the book „Multigrid" by Trottenberg, U., Oosterlee, C.W., Schüller, A., Academic Press, 2001, pp. 413-532.
17. Clees, T.: *"AMG Strategies for PDE Systems with Applications in Industrial Semiconductor Simulation"*, Ph.D. Thesis, University of Cologne, Nov 2004; Shaker, Aachen, Germany, 2005.
18. Lu, B.: Iterative Coupling Techniques for Reservoir Simulation with Multiphase Flow. Ph.D. Thesis, Center for Subsurface Modeling, ICES, The University of Texas at Austin, 2006 (in progress).
19. Elman H., Silvester, D. and Wathen, A.: *"Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics"*, Oxford Science Publications, 2005.
20. Behie, G. and Forsyth, P.: *"Incomplete Factorization Methods for Fully-Implicit Simulation of Enhanced Oil Recovery"*, SIAM J. Sci. Statist. Comput., Vol. 5, pp. 543-561, 1984.
21. Wallis, J.: "Two-step Preconditioning", Private Communication, 1993.
22. Dawson, C., Klíe, H., Wheeler, M.F. and Woodward, C.: *"A Parallel, Implicit, Cell-Centered Method for Two-Phase Flow with a Preconditioned Newton-Krylov Solver"*, Comp. Geosciences 1, pp. 215–249, 1997.
23. Lacroix, S., Vassileski, Yu., Wheeler, J. and Wheeler, M.F.: *"Iterative Solution Methods for Modeling Multiphase Flow in Porous Media Fully-Implicitly"*, SIAM J. Sci. Comput. 25, pp.905-926, 2003.
24. Scheichl, R., Masson, R. and Wendebourg, J.: *"Decoupling and Block Preconditioning for Sedimentary Basin Simulations"*, Comp. Geosciences 7, pp.295-318, 2003.
25. Monteagudo, J.E.P. and Firoozabadi, A.: *"Numerical Simulation of Water Injection in Disconnected and Connected Fracture Media Using Jacobian-Free Fully Implicit Control Volume Method"*, SPE/DOE Fourteenth Symposium on Improved Oil Recovery, SPE 89449, Tulsa, Oklahoma, April 17-21, 2004.
26. Bastian, P. and Helmig, R.: *"Efficient Fully-coupled solution Techniques For Two-Phase Flow in Porous Media: Parallel Multigrid Solution and Large Computations"*, Advances in Water Res. 23, pp. 199-216, 1999.
27. Bank, R., Chan, T., Coughan, W. and Smith, K.: *"The Alternate-Block-Factorization Procedure for Systems of Partial Differential Equations"*, BIT 29, pp. 938-954, 1989.
28. Wang, Z.Y., Wu, K.C. and Dutton, R.W.: *"An Approach to Construct Preconditioning Matrices for Block Iteration of Linear Equations"*, IEEE Trans. on Computer Aided Design 11, pp. 1334-1343, 1992.
29. Fan, Q., Forsyth, P.A., McMacken, J.R.F. and Tang, W.P.; *"Performance Issues for Iterative Solvers in Device Simulation"*, SIAM J. Sci. Comp. 17, pp. 100-117, 1996.
30. Trottenberg, U., Oosterlee, C.W., Schüller, A.: *„Multigrid"*, Academic Press, 2001.
31. Ruge, J.W. and Stüben. K.: *"Algebraic Multigrid (AMG)"* in "Multigrid Methods" (McCormick, S.F., ed.), SIAM, Frontiers in Applied Mathematics, Vol. 5, Philadelphia, 1986.
32. Stüben, K.: *"Review of Algebraic Multigrid"*, J. of Comp. and Appl. Math. 128, pp. 281-309, 2001.
33. Füllenbach, T. and Stüben, K.: *"Algebraic multigrid for selected PDE systems"*, Proceedings of the Fourth European Conference on Elliptic and Parabolic Problems, Rolduc (The Netherlands) and Gaeta (Italy), 2001. World Scientific, New Jersey, London, pp. 399-410, 2002.
34. Clees, T. and Stüben, K.: *"Algebraic multigrid for industrial semiconductor device simulation"*, Proceedings of the First International Conference on Challenges in Scientific Computing, Berlin, Germany, Oct 2-5, 2002. Lecture Notes in Computational Science and Engineering, Springer, Heidelberg, Berlin, 2003.
35. Stüben, K. and Clees, T.. *"SAMG User's Manual"*, Release 22c, Fraunhofer Institute SCAI, June 2005. Can be downloaded from http://www.scai.fhg.de/samg.
36. Clees, T. and Ganzer, L.: *"An Efficient Algebraic Multi-Grid Solver Strategy for Adaptive Implicit Methods in Oil Reservoir Simulation"* paper SPE 105789 presented at the 2007 SPE Reservoir Simulation Symposium, Houston, TX, Feb. 26-28.
37. Christie M.A. and Blunt, M.J.: *"Tenth SPE Comparative Solution Project: A Comparison of Upscaling Techniques"*, SPE Reservoir Simulation Symposium, SPE 72469, Houston, Feb 11-14, 2001.
38. Wheeler, J.: "IPARS User's Manual", *Tech Report CSM*, ICES, The University of Texas at Austin, Austin, TX, 2000.
39. Bank, R. and Wagner, Ch.: *"Multilevel ILU decomposition"* Numer. Math. 82, No.4, pp. 543-576, 1999.