NLA 372 pp: 1--21 (col.fig.: Nil) PROD. TYPE: COM ED: Chandra PAGN: Suresh – SCAN: Venki

NUMERICAL LINEAR ALGEBRA WITH APPLICATIONS
Numer. Linear Algebra Appl. 2004; **10**:000–000 (DOI: 10.1002/nla.372)

A systematic comparison of coupled and distributive smoothing in multigrid for the poroelasticity system

F. J. Gaspar[1], F. J. Lisbona[1], C. W. Oosterlee[2,*,†] and R. Wienands[3]

[1]*Departamento de Mathemática Aplicada, University of Zaragoza, Zaragoza, Spain*
[2]*Department of Applied Mathematical Analysis, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Mekelweg 4, Delft, CD 2628, The Netherlands*
[3]*Fraunhofer Institute for Algorithms and Scientific Computing (SCAI), Sankt Augustin, Germany*

SUMMARY

In this paper, we present efficient multigrid methods for the system of poroelasticity equations discretized on a staggered grid. In particular, we compare two different smoothing approaches with respect to efficiency and robustness. One approach is based on the coupled relaxation philosophy. We introduce 'cell-wise' and 'line-wise' versions of the coupled smoothers. They are compared with a distributive relaxation, that gives us a decoupled system of equations. It can be smoothed equation-wise with basic iterative methods. All smoothing methods are evaluated for the same poroelasticity test problems in which parameters, like the time step, or the Lamé coefficients are varied. Some highly efficient methods result, as is confirmed by the numerical experiments. Copyright © 2004 John Wiley & Sons, Ltd.

KEY WORDS: multigrid; coupled relaxation; decoupled distributive relaxation; comparison; poroelasticity; staggered discretization

1. INTRODUCTION

Multigrid methods are motivated by the fact that many iterative methods, especially if applied to elliptic problems, have a smoothing effect on the error between the exact solution and a numerical approximation. A smooth discrete error can be well represented on a coarser grid, where its approximation is much cheaper. The design of efficient smoothers in multigrid for the iterative solution of *systems* of partial differential equations (PDEs), however, often requires special attention. The relaxation method should smooth the error for all unknowns in the equations (that are possibly of different type) of the system.

A good indication for the appropriate choice of smoother is the system's determinant. If the main operators (or their principal parts) of the determinant lie on the main diagonal of the system's matrix, smoothing is a straightforward matter. In that case, the differential operator

*Correspondence to: C. W. Oosterlee, Department of Applied Mathematical Analysis, Faculty of Electrical Engineering, Mathematics and Computer Science, Delft University of Technology, Mekelweg 4, Delft, CD 2628, The Netherlands.
†E-mail: c.w.oosterlee@ewi.tudelft.nl

Received 15 May 2003
Accepted 15 December 2003

that corresponds to the primary unknown of each equation is the leading operator. Therefore, a simple equation-wise decoupled smoother can efficiently be used. If, however, the main operators in a system are not in the desired position, the choice of efficient smoother needs some care. A first obvious choice in the case of strong off-diagonal operators in the differential system is coupled smoothing: All unknowns in the system at a certain grid point are updated simultaneously.

Additional smoothing difficulties are met, if one of the operators on the system's main diagonal equals zero, or is very close to zero (i.e. with extremely small parameters in front of derivatives). However, for this situation also, different forms of coupled and (distributive) decoupled smoothers exist, which smooth the errors in all the unknowns effectively. The research underlying these smoothers is basically done in the late 1970s and in the early 1980s for incompressible flow problems [1, 2].

Here, we consider multigrid schemes based on coupled and decoupled relaxation for the system of incompressible poroelasticity equations. The system has been discretized on a staggered grid, which is one way to cope with numerical instabilities in the time-dependent process. In the staggered grid arrangement the three primary unknowns, displacements and pressure, are not defined at the same positions on the grid. The equation for the pressure contains a time-dependent divergence operator for the displacements and a Laplace operator for the pressure, possibly with an extremely small parameter in front. Details are given in Section 2.

Decoupled smoothing for this system is found in the distributive framework: Smoothing is applied after a post-conditioning step of the original system. This step transforms the system in such a way that the operators in the determinant of the original system appear on the main diagonal of the transformed system, ready for decoupled smoothing. The distributive, decoupled smoother for poroelasticity has already been introduced in Reference [3]. We repeat it briefly in Section 3.1 and provide efficient smoothers for a term with a biharmonic and a Laplace operator appearing after the transformation.

For coupled smoothing we compare two forms. In one version, three primary unknowns in the staggered arrangement are smoothed and updated simultaneously. In the second version of coupled smoothing, the divergence operator in the third equation is taken into account in a more profound way. A 'cell-wise' relaxation variant is chosen that updates five unknowns at once. Locally, the unknowns in the divergence operator are treated simultaneously. The coupled smoothers are described in Section 3.2.

We will compare coupled and distributive smoothing for the poroelasticity system numerically. Throughout the literature, especially in the description of multigrid for Stokes and incompressible Navier–Stokes equations, one of the two approaches mentioned above has typically been adopted. However, it does not become clear in the papers which relaxation method is to be preferred. In Reference [4], a coupled smoother is compared with distributive smoothers for an incompressible flow problem. It is stated that the coupled smoother is preferable, especially for convection dominating flows. In Reference [5], a distributive smoother is evaluated next to cell- and line-wise versions of the coupled smoother. The coupled smoother comes out best, but in Reference [6] it is concluded that for stratified flow a distributive smoother is to be preferred. An overview paper with many references for these smoothers in computational fluid dynamics problems is Reference [7]. In Reference [8] multigrid has been used for 3D poroelasticity, based on coupled smoothing.

Both smoothing approaches have their advantages and their disadvantages. If a system of equations consists of elliptic and of other, non-elliptic, components, decoupled relaxation easily allows to choose different relaxation methods for the different operators appearing, see, for example, Reference [9]. However, for general systems of equations it is not easy to find a suitable distributive relaxation scheme. Furthermore, the proper treatment of boundary conditions in distributive relaxation may not be trivial, as typically the system's operator is transformed by the smoother but the boundary operator is often not considered. For the system under consideration these problems are not observed. It is further not straightforward to use the concept of a blackbox iterative method, like algebraic multigrid (AMG), in combination with distributive relaxation. Distributive relaxation is based on transformations of the original system, that are not easily extracted from the corresponding matrix. For systems of equations the so-called point-block AMG approach [10] may be naturally used in combination with coupled relaxation. Results of this combination for poroelasticity are, however, not known. The solution methods proposed here rely on geometric multigrid concepts and benefit from insights in the poroelasticity system.

What distinguishes this paper from previous papers on the smoothing topic is that we compare the two tuned, highly efficient smoothing approaches systematically for identical discrete poroelasticity test problems. The other multigrid components, such as the transfer operators and the coarse grid discretization are identical. We will even count the number of floating point operations spent in the respective algorithms for the comparison. Several problem parameters, such as the Lamé coefficients and the time step are then varied. By this, we will gain valuable insight into the behaviour of each smoother. The comparison, presented in Section 4.1, will be performed with respect to efficiency and robustness of the multigrid methods.

We restrict ourselves to Cartesian grids in this paper. This basically covers the consolidation aspect of poroelasticity. In the field of tissue engineering, non-Cartesian grids are needed and a generalization to curvilinear or finite element unstructured grids may then be necessary.

2. DISCRETE POROELASTICITY SYSTEM

The poroelastic model in the classical Biot [11] consolidation theory can be formulated as a system of PDEs for the displacements in x and y directions, u and v, and the pore pressure of the fluid p. They build the solution vector $\boldsymbol{u} = (u, v, p)^{\mathrm{T}}$. The 2D incompressible variant of the system of poroelasticity equations reads

$$\begin{aligned} -(\lambda + 2\mu)u_{xx} - \mu u_{yy} - (\lambda + \mu)v_{xy} + p_x &= 0 \\ -(\lambda + \mu)u_{xy} - \mu v_{xx} - (\lambda + 2\mu)v_{yy} + p_y &= 0 \\ (u_x + v_y)_t - a(p_{xx} + p_{yy}) &= Q \end{aligned} \tag{1}$$

with $\lambda, \mu(\geqslant 0)$ the Lamé coefficients, $a = \kappa/\eta \geqslant 0$ with κ the permeability of the porous medium, η the viscosity of the fluid and Q a source term. The system comes with initial and boundary conditions.

A '*stationary*' *model operator* $\boldsymbol{L}$ from (1) which is suitable for analysis reads

$$\boldsymbol{L}\boldsymbol{u} = \begin{pmatrix} -(\lambda+2\mu)\partial_{xx} - \mu\partial_{yy} & -(\lambda+\mu)\partial_{xy} & \partial_x \\ -(\lambda+\mu)\partial_{xy} & -\mu\partial_{xx} - (\lambda+2\mu)\partial_{yy} & \partial_y \\ \partial_x & \partial_y & -\tilde{a}\Delta \end{pmatrix} \begin{pmatrix} u \\ v \\ p \end{pmatrix} = \boldsymbol{f} \qquad (2)$$

with Laplace operator $\Delta = \partial_{xx} + \partial_{yy}$. System (2) represents an operator after an implicit (semi-) discretization in time; $\tilde{a}$ equals, for example, $0.5a\delta t$.

From (2) the corresponding determinant reads

$$\det(\boldsymbol{L}) = -\mu\Delta(\tilde{a}(\lambda+2\mu)\Delta^2 - \Delta) \qquad (3)$$

Here, $\Delta^2 = \partial_{xxxx} + 2\partial_{xxyy} + \partial_{yyyy}$ (biharmonic operator). The principal part of $\det(\boldsymbol{L})$ is Δ^m. In the common situation with $\mu,\ \tilde{a},\ \lambda+2\mu>0$, we have $m=3$.

It is obvious that the leading operators in determinant (3) do not appear on system's main diagonal (2). As mentioned in the Introduction, straightforward decoupled smoothing in multigrid will then not lead to efficient geometric multigrid methods. As the parameter $\tilde{a}$ in the main diagonal block of the third equation can become small, in dependence on the time step δt, we have to consider coupled, or distributive relaxation methods for this system.

Another (slight) complication for smoothing arises from the discretization chosen here. The poroelasticity operator (2) suffers from stability difficulties when strong pressure gradients are present. Standard discretizations, like central diffferences on regular meshes or usual finite elements, applied to poroelasticity system (2) suffer from some oscillating behaviour when strong gradients of pressure are present, due to a lack of stability of the methods (the inf–sup condition is not satisfied). To overcome these stability difficulties, a staggered grid was proposed in Reference [12] and employed in Reference [3] for (1), using central differences on a uniform staggered grid with mesh size h. (Staggering is a well-known discretization technique in computational fluid dynamics, in particular for incompressible flow [13, 14], where the third diagonal block in the system equals zero.)

Often in poroelasticity problems pressure values are prescribed at the physical boundary. So, pressure points in the staggered grid should be located at the physical boundary, and the displacement points are then defined at the cell faces, see Figure 1. A divergence operator is naturally approximated by a central discretization of the displacements around the pressure point.

The discretization of each equation, centred around the equation's primary unknown, reads

$$\boldsymbol{L}_h\boldsymbol{u}_h = \begin{pmatrix} -(\lambda+\mu)(\partial_{xx})_h - \mu\Delta_h & -(\lambda+\mu)(\partial_{xy})_{h/2} & (\partial_x)_{h/2} \\ -(\lambda+\mu)(\partial_{xy})_{h/2} & -\mu\Delta_h - (\lambda+\mu)(\partial_{yy})_h & (\partial_y)_{h/2} \\ (\partial_x)_{h/2} & (\partial_y)_{h/2} & -\tilde{a}\Delta_h \end{pmatrix} \begin{pmatrix} u_h \\ v_h \\ p_h \end{pmatrix} = \boldsymbol{f}_h \qquad (4)$$

The following discrete operators on the staggered grid are used in (4) (given in stencil notation):

$$(\partial_x)_{h/2} \mathrel{\hat{=}} \frac{1}{h}[-1 \quad \star \quad 1]_h, \qquad -(\partial_{xx})_h \mathrel{\hat{=}} \frac{1}{h^2}[-1 \quad 2 \quad -1]_h$$

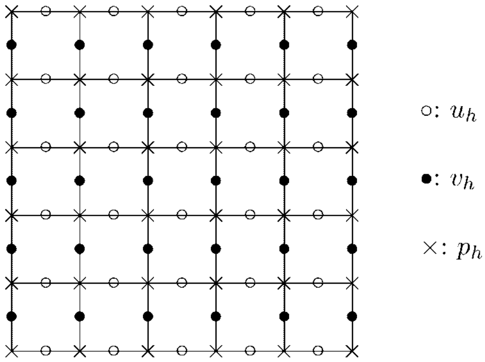

Figure 1. Staggered location of unknowns for poroelasticity.

$$(\partial_{xy})_{h/2} \triangleq \frac{1}{h^2}\begin{bmatrix} -1 & & 1 \\ & \star & \\ 1 & & -1 \end{bmatrix}_h, \qquad -\Delta_h \triangleq \frac{1}{h^2}\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}_h$$

($(\partial_y)_{h/2}$ and $-(\partial_{yy})_h$ are given by analogous stencils.)

The '★' denotes the position on the grid at which the stencil is applied, i.e. ○, • or ×, respectively, in Figure 1.

Furthermore, we choose the Crank–Nicolson time discretization. It is confirmed in Reference [3] that for test problems without singularities we obtain $O(h^2 + \delta t^2)$-accuracy.

Remark: stretching and staggering
We will also use *stretched* staggered grids. Often, boundary layers occur in the beginning of the time-dependent consolidation process, due to pressure boundary conditions. The staggered grid is a remedy for unphysical oscillations near the boundary layer. However, grid stretching serves the same purpose: It may be sufficient to use adequate stretching and a collocated grid to capture a boundary layer well. Here, we use the combination stretching and staggering for evaluating multigrid's robustness.

3. MULTIGRID SOLUTION METHOD

Efficient multigrid solvers for the system of poroelasticity equations discretized on staggered grids are evaluated. We consider both distributive and coupled relaxation methods in the following subsections.

3.1. Distributive relaxation

In order to relax $\boldsymbol{L}_h\boldsymbol{u}_h = \boldsymbol{f}_h$, a ghost variable $\boldsymbol{w}_h$ is used with $\boldsymbol{u}_h = \boldsymbol{C}_h\boldsymbol{w}_h$ and the transformed system $\boldsymbol{L}_h\boldsymbol{C}_h\boldsymbol{w}_h = \boldsymbol{f}_h$ is considered in distributive relaxation [1, 15]. $\boldsymbol{C}_h$ is chosen such that the resulting system $\boldsymbol{L}_h\boldsymbol{C}_h$ is triangular [16]. The transformed system is then suited for *decoupled* smoothing. The distributor, introduced in Reference [3], that fulfils these requirements for poroelasticity reads

$$\boldsymbol{C}_h = \begin{pmatrix} I_h & 0 & -(\partial_x)_{h/2} \\ 0 & I_h & -(\partial_y)_{h/2} \\ (\lambda+\mu)(\partial_x)_{h/2} & (\lambda+\mu)(\partial_y)_{h/2} & -(\lambda+2\mu)\Delta_h \end{pmatrix} \tag{5}$$

with identity I_h. The transformed system reads, combine (4) and (5),

$$\boldsymbol{L}_h\boldsymbol{C}_h = \begin{pmatrix} -\mu\Delta_h & 0 & 0 \\ 0 & -\mu\Delta_h & 0 \\ LC_h^{3,1} & LC_h^{3,2} & \tilde{a}(\lambda+2\mu)\Delta_h^2 - \Delta_h \end{pmatrix} \tag{6}$$

with

$$LC_h^{3,1} = (\partial_x)_{h/2} - \tilde{a}(\lambda+\mu)((\partial_{xxx})_{h/2} + (\partial_{xyy})_{h/2})$$

and

$$LC_h^{3,2} = (\partial_y)_{h/2} - \tilde{a}(\lambda+\mu)((\partial_{xxy})_{h/2} + (\partial_{yyy})_{h/2})$$

where the discrete operators appearing read (in stencil notation),

$$(\partial_x)_{h/2} \triangleq \frac{1}{h}[-1 \quad \star \quad 1]_h, \qquad (\partial_{xxx})_{h/2} \triangleq \frac{1}{h^3}[-1 \quad 3 \quad \star \quad -3 \quad 1]_h$$

$$(\partial_{xxy})_{h/2} \triangleq \frac{1}{h^3}\begin{bmatrix} 1 & -2 & 1 \\ & \star & \\ -1 & 2 & -1 \end{bmatrix}_h, \qquad \Delta_h^2 \triangleq \frac{1}{h^4}\begin{bmatrix} & & 1 & & \\ & 2 & -8 & 2 & \\ 1 & -8 & 20 & -8 & 1 \\ & 2 & -8 & 2 & \\ & & 1 & & \end{bmatrix}_h$$

(The other discrete operators are given by analogous stencils.) Notice that the diagonal elements of the triangular $\boldsymbol{L}_h\boldsymbol{C}_h$ are factors of $\det(\boldsymbol{L}_h)$ (discrete version of (3)), which is a highly desirable feature.

In detail, the distributive relaxation consists of two steps, the *predictor* and the *corrector*. In the *predictor* step, a new approximation $\delta\boldsymbol{w}^{m+1}$ to the 'ghost variable' $\delta\boldsymbol{w} = (\delta w_u, \delta w_v, \delta w_p)^{\mathrm{T}}$ is computed,

$$\boldsymbol{L}_h\boldsymbol{C}_h\delta\boldsymbol{w}^{m+1} = \boldsymbol{r}_h^m \tag{7}$$

with residual $\boldsymbol{r}_h^m = \boldsymbol{L}_h\boldsymbol{u}_h^m - \boldsymbol{f}_h$.

The first two equations in (6), (7) can be smoothed with an efficient smoother for the Laplace operator. This is typically the well-known red–black Gauss–Seidel relaxation (in 2D and 3D) [17, 18], which is well parallelizable. The corresponding smoothing factor in 2D is 0.063 for two iterations.

The challenging task here is to find a highly efficient smoother for the third equation in (7),

$$(\tilde{a}(\lambda + 2\mu)\Delta_h^2 - \Delta_h)\delta w_p = r_{3,h} \tag{8}$$

(with $r_{3,h}$ composed of the terms $LC_h^{3,1}, LC_h^{3,2}$ and Q).

There are several ways of smoothing the operator in (8). Good smoothing factors are obtained with an overrelaxation parameter ω in red–black Jacobi point relaxation (RB-JAC), as shown in Reference [3]. This is a red–black scheme, where Jacobi is employed within each colour. A suitable overrelaxation parameter for the combination of the two operators, Δ_h^2 and Δ_h, is $\omega = \frac{25}{18} \approx 1.4$ [19]. The smoothing factor for two iterations is bounded by $\frac{1}{4}$ for all mesh sizes and problem parameters [3]. The overrelaxation is performed *after a complete RB-JAC step* in a multi-stage fashion (not-as usual-within a RB-JAC relaxation). This smoother is abbreviated by **dist_bih_rb**.

A multi-stage variant of any arbitrary relaxation S_h is given by

$$\prod_{i=1}^{\tilde{m}} ((1 - \omega_i)I_h + \omega_i S_h)$$

with discrete identity I_h and multi-stage relaxation parameters ω_i $(i = 1, \ldots, \tilde{m})$. As a second variant, we also include a 2-stage version of the RB-JAC method for Equation (8). Suitable multi-stage parameters are in this case $\omega_1 = 2.1$, $\omega_2 = 1$ [19]. This smoother is abbreviated by **dist_bih_ms**.

Next, we consider a different approach for the third equation, that avoids smoothing directly for a biharmonic operator. This approach (similar to Reference [20]) may therefore be more easily applied in the case of curved grids. An efficient smoother is found by splitting the operator in the third equation, as follows

$$-\Delta_h q_h = r_{3,h} \quad (-\tilde{a}(\lambda + 2\mu)\Delta_h + 1)\delta w_p = q_h \tag{9}$$

with extra slack variable q_h. This way, we deal with simple operators of Laplace-type for smoothing. The two operators in (9) can be smoothed with red–black Gauss–Seidel iteration, but also with line-wise Gauss–Seidel relaxation methods. We evaluate the variant with red–black Gauss–Seidel relaxation (**dist_2lp_rb**) and one with alternating line Gauss–Seidel (**dist_2lp_lin**) in the numerical experiments. In an alternating line Gauss–Seidel method, lines in the x and y directions are processed. Of course, line-wise relaxation methods are mainly applicable on structured grids, as we have them here in our applications.

In **dist_2lp_rb**, an underrelaxation parameter $\omega = 0.85$ (obtained experimentally) is necessary for fast convergence of (9). Line-wise Gauss–Seidel relaxation is necessary for satisfactory multigrid convergence in the case of stretched grid test problems (we employ standard geometric grid coarsening). A relaxation of *zebra* line type, in which first all odd numbered lines are processed before all even numbered lines, did not lead to faster multigrid convergence and is therefore not included. Notice that the four distributive variants mainly differ in the treatment of the third scalar equation. The other difference is that the line-wise variant also employs line smoothing for the first two equations.

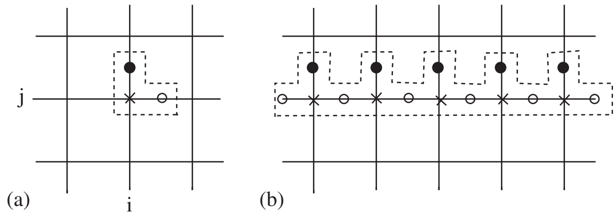

Figure 2. 'Three unknown' coupled relaxation: (a) triad-wise; (b) x-line-wise; $\times$: p_h; $\circ$: u_h; $\bullet$: v_h.

In the *corrector step*, the new approximation for $\boldsymbol{u}_h$ is then added to the present approximation as

$$\boldsymbol{u}_h^{m+1} = \boldsymbol{u}_h^m + \delta \boldsymbol{u}_h^{m+1} = \boldsymbol{u}_h^m + \boldsymbol{C}_h \delta \boldsymbol{w}^{m+1} \qquad (10)$$

This is just a matrix–vector product. The implementation is straightforward.

The distributive relaxation is designed such that its performance should be independent of problem parameters, like the Lamé coefficients or the time step.

Remark: boundary conditions
In distributive smoothing, the order of $\boldsymbol{L}_h\boldsymbol{C}_h$ is higher than the order of $\boldsymbol{L}_h$ and hence boundary conditions for corrections $\delta \boldsymbol{w}$ need to be supplied. There is considerable freedom in selecting the boundary conditions. For our model applications, we can use simple Dirichlet and Neumann boundary conditions for $\delta \boldsymbol{w}$, whenever we prescribe them for $\boldsymbol{u}$. For poroelasticity problems with a prescription of stress components, for example, the proper treatment will depend on the specific boundary condition.

Remark: number of smoothing steps
In principle, it is not necessary to employ the same number of smoothing steps for both operators in (9). In our case, however, using one relaxation step for each operator brings the fastest convergence.

3.2. Coupled relaxation

Straightforward generalization of coupled smoothing with unknowns in a staggered grid arrangement is to relax *triads* of three unknowns, $u_{i,j}$, $v_{i,j}$ and $p_{i,j}$, simultaneously. Figure 2(a) shows a triad.

In the 'triad-wise' variant, a small 3×3-matrix must be solved, for all triads in the staggered grid. It is convenient to consider the *correction* equations

$$\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} \\ a_{2,1} & a_{2,2} & a_{2,3} \\ a_{3,1} & a_{3,2} & a_{3,3} \end{pmatrix} \begin{pmatrix} \delta u_{i,j} \\ \delta v_{i,j} \\ \delta p_{i,j} \end{pmatrix}^{m+1} = \begin{pmatrix} r_{i,j}^1 \\ r_{i,j}^2 \\ r_{i,j}^3 \end{pmatrix}^{m} \qquad (11)$$

where $a_{3,3}$ can be an extremely small entry from the third diagonal block of the system. (During the elimination process $a_{3,3}$ is replaced by larger elements.) In the correction equation setting, it is easily possible to discard certain elements in (11), for example, the elements $a_{1,2}, a_{2,1}$ related to the mixed derivatives. This is not necessary for our applications.

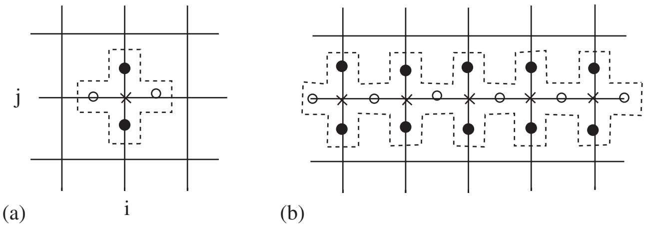

Figure 3. Five unknown coupled relaxation: (a) cell-wise; (b) x-line-wise; $\times$: p_h; $\circ$: u_h; $\bullet$: v_h.

Afterwards, the correction is added to the current approximation, possibly with a relaxation parameter,

$$\mathbf{u}_{i,j}^{m+1} = \mathbf{u}_{i,j}^{m} + \omega \boldsymbol{\delta} \mathbf{u}_{i,j}^{m+1}$$

We use $\omega = 1$.

The triads can be processed in different orderings. An obvious choice for triad numbering is the lexicographic Gauss–Seidel ordering, but also the red–black Gauss–Seidel ordering may be promising for this system. The red–black ordering has advantages over the lexicographic for parallel processing purposes. These variants are abbreviated by **triad_lex** and **triad_rb**, respectively. The relaxation can be performed in triad-wise or in a line-wise fashion if grid anisotropies occur in a test problem. Zebra or lexicographic line Gauss–Seidel ordering is then appropriate. For each line a block tridiagonal matrix has to be inverted. Figure 2b presents the line-wise variant of this coupled smoothing process. We include an alternating line Gauss–Seidel version in the comparison, denoted by **triad_lin**.

It is reported in Reference [2] that the triad smoother is not satisfactorily for incompressible Navier–Stokes equations. A better alternative is a coupled smoother [2], that locally updates all unknowns appearing in the divergence operator in the third equation (4) simultaneously. In practice, this means that instead of the three unknowns (11), five unknowns (pressure $p_{i,j}$, 2 times u_h- and v_h-displacements, $u_{i,j}, u_{i-1,j}, v_{i,j}, v_{i,j-1}$, centred around a pressure point) should be relaxed simultaneously. 'Cell-wise' smoothing is shown in Figure 3(a). A small 5×5-matrix must be inverted for each cell. Notice that the word 'cell' does not relate to a grid cell here, as the unknowns are centred around a pressure point. It is used to distinguish both coupled smoothers. For incompressible Navier–Stokes equations, the corresponding cell-wise relaxation method is sometimes called the 'Vanka smoother' after the author of the first paper [2]. In one smoothing iteration all displacement unknowns are updated twice, whereas pressure unknowns are updated once. This makes an iteration with this smoother more expensive than the triad smoother. For the cell-wise version the orderings can again be lexicographic or red–black. The two variants are abbreviated by **vanka_lex** and **vanka_rb**, respectively.

Figure 3(b) presents the line-wise version of the Vanka smoother. It has been used in the CFD context in References [21, 22]. The line-wise versions can be in lexicographic, zebra line or in alternating line ordering. The block matrices to be inverted are somewhat more involved than those for the line-wise triad smoother. The cost of a coupled line-wise iteration, however, is substantially higher than the cost of distributive line-wise relaxation. An alternating line Gauss–Seidel version is evaluated, denoted by **vanka_lin**.

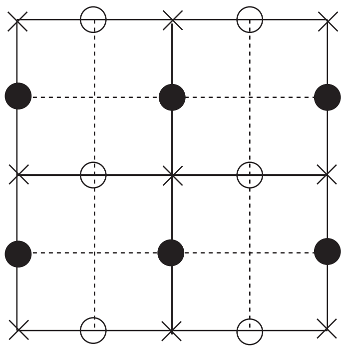

Figure 4. Fine and coarse grid cells and unknowns, ∘: u_H point, •: v_H point, ×: p_H point.

Remark: (coupled line smoothing)
In numerical poroelasticity experiments with stretched grids in Section 4, it is found that the multigrid methods with the coupled line-wise versions only converge satisfactorily, if the terms with mixed derivatives in (4) are not included in the block matrix but placed in the right-hand side instead. Otherwise, due to problems with the diagonally dominance of the block matrices slow convergence is observed.

3.3. Coarse grid correction

In the multigrid method we choose standard geometric grid coarsening on the Cartesian grids, i.e. the sequence of coarse grids is obtained by doubling the mesh size in each space direction. This is indicated by the subscript '$2h$'. An appropriate coarse grid correction consists of geometric transfer operators $R_{h,2h}$, $P_{2h,h}$, and direct coarse grid discretization (i.e. coarse grid analog of $\boldsymbol{L}_h$). For the poroelasticity system there is no benefit in using the Galerkin coarse grid discretization. The Galerkin coarse grid discretization merely results in a larger stencil here. For real-life problems with jumps in the permeability coefficient κ, we may need to reconsider Galerkin coarse grid operators, as they lead to natural coarse grid operators for problems with jumping coefficients.

The transfer operators that act on the different unknowns are dictated by the staggered grid, see Figure 4. At u- and v-grid points we consider 6-point restrictions and at p-grid points a 9-point restriction. In stencil notation they are given by

$$R^u_{h,2h} \triangleq \frac{1}{8}\begin{bmatrix} 1 & & 1 \\ 2 & \star & 2 \\ 1 & & 1 \end{bmatrix}_h, \quad R^v_{h,2h} \triangleq \frac{1}{8}\begin{bmatrix} 1 & 2 & 1 \\ & \star & \\ 1 & 2 & 1 \end{bmatrix}_h, \quad R^p_{h,2h} \triangleq \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h \tag{12}$$

respectively. As the prolongation operators $P^{u/v/p}_{2h,h}$, we apply the usual interpolation operators based on bilinear interpolation of neighbouring coarse grid unknowns in the staggered grid.

3.4. Number of floating point operations

As a measure for the performance of the respective multigrid methods, we count the number of floating point operations (flops) during the iterative solution of the time-dependent poroelasticity test problems. This may give additional insight in the difference in CPU time spent, for example, in coupled and decoupled, in 'point-wise' and line-wise relaxation or in multigrid V- and F-cycles. The number of flops is independent of the hardware on which the problems are solved. For simplicity, we count here additions, multiplications and also divisions as one flop.

4. NUMERICAL EXPERIMENTS

In the numerical experiments, we evaluate the smoothers described. We summarize the abbreviations introduced for the smoothers

dist_bih_rb : distributive, 3rd eq. based on bih. op., red–black Jac. $\omega = 1.4$
dist_bih_ms : distr., 3rd eq. bih. op., ms. red–black Jac. $\omega_1 = 2.1, \omega_2 = 1.0$
dist_2lp_rb : distr., 3rd eq. based on 2 Laplace op., red–black GS, $\omega = 0.85$
dist_2lp_lin : distr., 3rd eq. based on 2 Laplace op., alt. line GS, $\omega = 1.0$

triad_lex : triad-wise coupled, lexicographic GS, $\omega = 1$
triad_rb : triad-wise coupled, red–black GS, $\omega = 1$
triad_lin : triad-line-wise coupled, alt. line GS, $\omega = 1$

vanka_lex : cell-wise coupled, lexicographic GS, $\omega = 1$
vanka_rb : cell-wise coupled, red–black GS, $\omega = 1$
vanka_lin : cell-line-wise coupled, alt. line GS, $\omega = 1$

The measure for convergence is related to the absolute value of the residual after the mth iteration in the maximum norm over the three equations in the system,

$$\text{res}^m = |r_{1,h}| + |r_{2,h}| + |r_{3,h}|$$

The multigrid convergence factor ρ_h presented in the tables below is then given by

$$\rho_h = \sqrt[5]{\frac{\text{res}^m}{\text{res}^{m-5}}} \tag{13}$$

For m the last iteration is chosen before the stopping criterion is met. This quantity is typically somewhat better than the asymptotic convergence factor.

In the following sections, we report on the multigrid convergence of the numerical experiments. Corresponding analysis results based on Fourier analysis are available for some of the smoothers, but they will be presented elsewhere. The analysis results agree well with our numerical convergence, which indicates that our straightforward boundary (and near-boundary) treatment in the smoothing methods does not influence the convergence negatively.

4.1. Multigrid convergence for first model problem

Some analytical reference solutions are known in the literature [23] for (1) in dimensionless form, where scaling has taken place with respect to a characteristic length of the medium ℓ, Lamé constants $\lambda + 2\mu$, time scale t_0 and a.

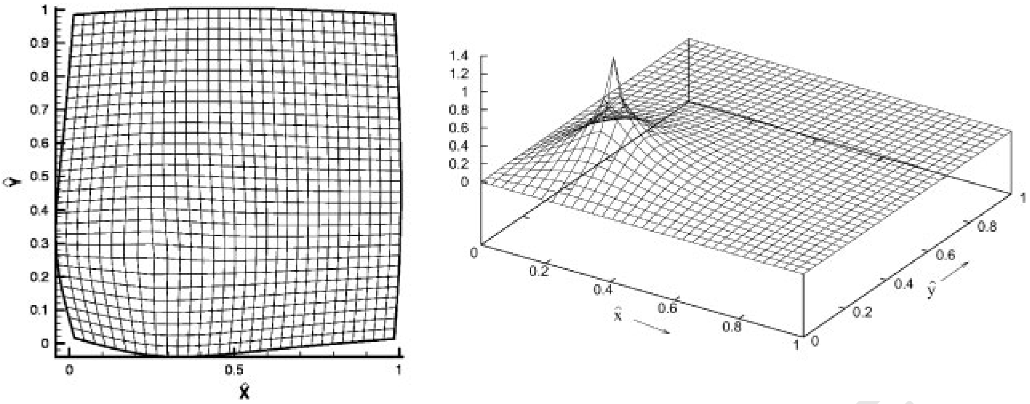

Figure 5. Numerical solution for displacement and pressure for 2D poroelasticity reference problem, 32^2-grid.

By choosing a unit squared domain, a source term $Q = 2\sin\hat{t}\cdot\delta_{0.25,0.25}$ ($\hat{t} = (\lambda + 2\mu)at$, δ is the Kronecker delta function), the following boundary and initial conditions:

$$\text{at } y = \{0, 1\}, \quad u = 0, \quad \partial v/\partial y = 0$$

$$\text{at } x = \{0, 1\}, \quad v = 0, \quad \partial u/\partial x = 0$$

and pressure $p = 0$ at the boundaries, we can mimic the dimensionless situation. In this case, the solution can be written as an infinite series [23]. An interesting feature is that this solution is independent of the Lamé coefficients. Figure 5 shows for this setting the computed displacement and pressure solution at time $\hat{t} = \pi/2$. The solution resembles the exact solution in Reference [23] very well, see also [3]. $O(h^2 + \delta t^2)$ accuracy is observed for the displacements, and, asymptotically, for the pressure too (despite the occurrence of the delta function which usually influences the numerical accuracy negatively) [3].

This reference problem is solved with multigrid. In the various multigrid methods compared here, only the smoother changes. We start the evaluation with a basic form of the equations, by choosing the Lamé coefficients in (4) as $\lambda = 0, \mu = \frac{1}{2}$ and coefficient $\tilde{a} = 0.5a\delta t = 5 \times 10^{-3}$ ($a = 1, \delta t = 10^{-2}$). We consider here the multigrid convergence in the first time step with different mesh sizes ranging from $h = \frac{1}{64}$ to $\frac{1}{256}$. These convergence statistics are representative for all other time steps.

Table I shows the V(1,1)- and F(1,1)-cycle results for the four variants of distributive relaxation, whereas Table II compares the four coupled relaxation methods. They present the convergence factor ρ_h (13), the number of iterations to reach the stopping criterion in brackets, and correspondingly the CPU time in seconds needed for this first time step. The stopping criterion is chosen as the absolute residual over all unknowns to be less than 10^{-9}. This criterion is too severe for realistic applications, but well-suited for our investigation of the multigrid convergence. The PC used for the timing results is a Pentium IV with 2.6 Mhz. A matrix–free version of multigrid is used; the CPU times include the time for computing the operator elements.

Table I. V(1,1)- and F(1,1)-multigrid convergence factors with *distributive smoothing*; in brackets, the average number of iterations for a residual reduction to 10^{-9}, and the corresponding CPU time in econds.

		Smoother			
Cycle	Grid	**dist_bih_rb**	**dist_bih_ms**	**dist_2lp_rb**	**dist_2lp_lin**
V(1,1)	256^2	0.24 (16) 49″	0.16 (13) 37″	0.20 (15) 42″	0.15 (12) 33″
	128^2	0.23 (15) 12″	0.15 (12) 9″	0.19 (14) 10″	0.15 (11) 8″
	64^2	0.21 (13) 3″	0.15 (11) 2″	0.18 (13) 2″	0.15 (11) 2″
F(1,1)	256^2	0.15 (13) 51″	0.10 (11) 42″	0.12 (12) 44″	0.10 (11) 41″
	128^2	0.14 (13) 13″	0.10 (11) 10″	0.12 (11) 10″	0.10 (10) 10″
	64^2	0.13 (12) 4″	0.10 (11) 2″	0.11 (10) 2″	0.10 (9) 2″

Table II. V(1,1)- and F(1,1)-multigrid convergence factors with *coupled smoothing*; in brackets, the average number of iterations for a residual reduction to 10^{-9}, and the corresponding CPU time in seconds.

		Smoother			
Cycle	Grid	**triad_lex**	**triad_rb**	**vanka_lex**	**vanka_rb**
V(1,1)	256^2	0.24 (16) 62″	0.23 (16) 62″	0.17 (14) 83″	0.10 (11) 65″
	128^2	0.24 (16) 16″	0.23 (15) 15″	0.17 (13) 20″	0.10 (10) 15″
	64^2	0.20 (15) 3″	0.22 (14) 3″	0.17 (12) 5″	0.10 (10) 4″
F(1,1)	256^2	0.20 (15) 76″	0.17 (13) 66″	0.17 (14) 111″	0.07 (10) 78″
	128^2	0.20 (15) 19″	0.17 (13) 18″	0.17 (13) 27″	0.07 (9) 18″
	64^2	0.20 (15) 5″	0.17 (12) 4″	0.17 (12) 6″	0.07 (9) 5″

An h-independent convergence can be observed in the tables for all variants of distributive and coupled relaxation for these problem parameters for the F-cycle. Especially, the CPU time results of the distributive variants are very satisfactory. The fastest method is the alternating line smoother based on the splitting into two Laplace-type operators. This may be somewhat surprising for a problem without any grid anisotropies. The multi-stage variant **dist_bih_ms** is equally fast. It is, however, slower than we would expect from Fourier *smoothing* analysis results. In the case of two smoothing iterations, the smoothing factor of **dist_bih_ms** is 0.025 (for **dist_bih_rb** it is 0.25). However, a Fourier two-grid analysis already shows that these excellent smoothing factors cannot be maintained in a two-grid method. The corresponding Fourier two-grid factors are for **dist_bih_ms** 0.13 and for **dist_bih_rb** 0.24. The results in Table I are somewhat better than these asymptotic two-grid Fourier analysis factors.

The CPU time of the coupled smoothing methods in Table II is somewhat higher than of the methods from Table I, although a similar number of iterations is needed to meet the convergence criterion. Among the coupled smoothers, the Vanka smoothers are more expensive than the triad smoothers. This is because the Vanka smoother treats each displacement unknown twice. An interesting observation is that the convergence with the Vanka smoother is improved by red–black relaxation of the cells, whereas a red–black relaxation of triads does not improve the multigrid convergence.

Furthermore, the V-cycle seems sufficient for this problem; it is fastest and shows (almost) h-independent behaviour. From this single experiment, we cannot yet distinguish clearly between methods.

4.2. Smaller time steps

We keep the problem formulation of the previous section, but vary systematically some problem parameters. The first parameter that varies in this section is the time step δt. (The Lamé coefficients are kept at $\lambda = 0, \mu = \frac{1}{2}$.) The time step is chosen smaller: It ranges now from 10^{-3} to 10^{-6}. This affects $\tilde{\alpha}$ in (2). In the case of pressure boundary layers in the initial stage of a consolidation process, such small time steps are realistic. The grid size is set to 256^2. As the coefficient in the third diagonal block of the poroelasticity system now tends to zero, we expect that the triad smoother may fail to converge (as observed for incompressible Navier–Stokes in Reference [2]). Figure 6 presents convergence plots for V(1,1)- and F(1,1)-cycles with the three smoothers **triad_rb** (Figures 6(a) and 6(b)), **vanka_rb** (Figures 6(c) and 6(d)) and **dist_2lp_rb** (Figures 6(e) and 6(f)). The other variants did not lead to other convergence tendencies. Within the figures the time step is varied. Figure 6(a) contains $\delta t = 10^{-5}$, instead of $\delta t = 10^{-6}$ in the other pictures. This is because $\delta t = 10^{-5}$ is the first time step for which divergence with the triad smoother in the V(1,1)-cycle is observed. Note that the y-axis is in a logarithmic scale. Figure 6 shows that both coupled smoothers are more sensitive to the variation of the time step than the distributive smoother: The convergence slope in Figures 6(e) and 6(f) is independent of δt. The robustness of the triad smoother is clearly limited, as in Figures 6(a) and 6(b) the multigrid convergence degrades severely for extremely small time steps. The convergence of the Vanka smoother is also sensitive with respect to variations in δt: smaller δt leads to slower convergence. From this experiment, the distributive relaxation is to be preferred for extremely small time steps.

4.3. Variation of Lamé coefficients

In this section, we vary the Lamé coefficients and investigate their effect on the multigrid convergence. The grid for these tests is 256^2; time step $\delta t = 0.1$. Two cases are compared: $\lambda = \mu = 1$ and $\lambda = 10^3$, $\mu = 10^4$. Table III presents for both, the distributive and the coupled, relaxation methods V(1,1)- and F(1,1)-cycle multigrid convergence in the first time step. Next to convergence factor ρ_h (13), the number of iterations to reduce the absolute value of the residual to less than 10^{-9} is shown. As expected [3], Table III shows that the multigrid methods with distributive relaxation converge at the same speed, independent of the size of the Lamé coefficients. It is also very similar to the convergence in Table I. The convergence with the triad smoother depends on the Lamé coefficients ($\lambda = \mu = 1$ converges slower than the second case); for the Vanka smoother this is not the case. Overall, all results are impressive for such a complicated system.

Table IV presents, in addition, the number of flops to reduce the residual to the more realistic value of 10^{-5} for all methods. The Lamé coefficients are set to $\lambda = 10^3$, $\mu = 10^4$, the grid is 256^2, $\delta t = 0.1$. The total number of flops is given for one and for five time steps. Table IV confirms the convergence tendency from Table III for $\lambda = 10^3$, $\mu = 10^4$. Again the V(1,1)-cycle in the distributive relaxation **dist_2lp_lin** performs best. The multi-stage distributive smoother also performs very well. For this parameter set, however, the red–black version of the triad smoother converges well. The Vanka smoother is more expensive. Furthermore,

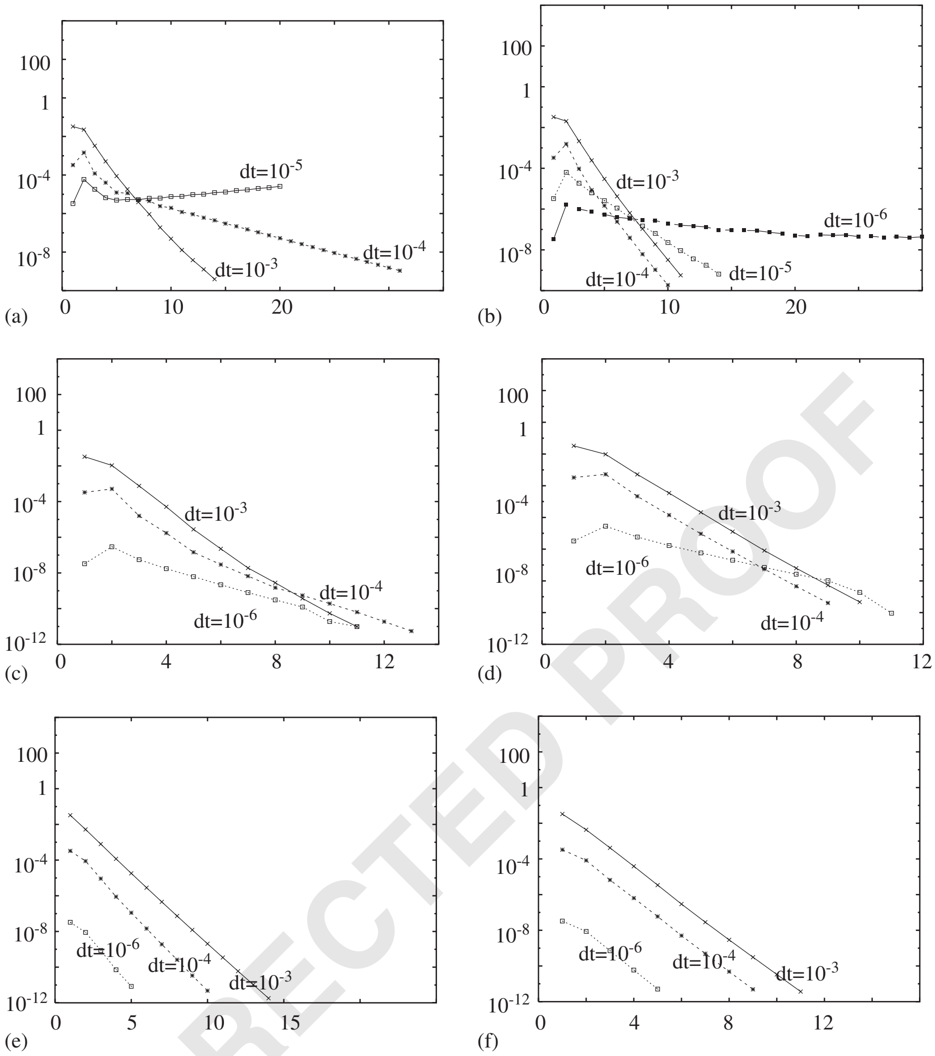

Figure 6. Multigrid convergence for very small time steps: (a) V-cycle, triad smoother; (b) F-cycle, triad smoother; (c) V-cycle, Vanka smoother; (d) F-cycle, Vanka smoother; (e) V-cycle, distributive smoother; (f) F-cycle, distributive smoother.

the number of flops needed for five time steps is about five times the number for one time step. This indicates that the convergence in the first time step is a representative measure for the total process.

4.4. *Problem with realistic parameters*

Next, we evaluate a poroelasticity test problem with more realistic parameters. These are the following. The domain size is larger, $\Omega = (-50, 50) \times (0, 100)$; the Lamé coefficients are $\lambda = 8333$, $\mu = 12500$; the porosity is $a = \kappa/\eta = 10^{-6}$. As the boundary conditions for the pres-

Table III. 256^2-grid multigrid convergence factors with distributive and coupled smoothing, variation in Lamé coefficients.

Cycle	λ, μ	Smoother			
		dist_bih_rb	**dist_bih_ms**	**dist_2lp_rb**	**dist_2lp_lin**
V(1,1)	1,1	0.25 (20)	0.21 (17)	0.21 (19)	0.18 (15)
F(1,1)	1,1	0.19 (16)	0.10 (13)	0.11 (14)	0.10 (13)
V(1,1)	$10^3, 10^4$	0.25 (20)	0.21 (17)	0.21 (19)	0.18 (15)
F(1,1)	$10^3, 10^4$	0.19 (16)	0.10 (13)	0.12 (14)	0.10 (13)
		triad_lex	**triad_rb**	**vanka_lex**	**vanka_rb**
V(1,1)	1,1	0.35 (23)	0.38 (24)	0.16 (16)	0.12 (13)
F(1,1)	1,1	0.34 (21)	0.30 (19)	0.18 (16)	0.08 (11)
V(1,1)	$10^3, 10^4$	0.27 (20)	0.27 (19)	0.16 (16)	0.11 (13)
F(1,1)	$10^3, 10^4$	0.23 (17)	0.18 (15)	0.18 (16)	0.08 (11)

Table IV. Number of flops ($\times 10^9$) to reach $\text{res}^m \leqslant 10^{-5}$ for one and five time steps, with distributive and coupled smoothing, $\lambda = 10^3$, $\mu = 10^4$.

Cycle	No. steps	Smoother			
		dist_bih_rb	**dist_bih_ms**	**dist_2lp_rb**	**dist_2lp_lin**
V(1,1)	1	1.35	1.40	1.74	1.32
	5	6.15	6.33	7.80	6.01
F(1,1)	1	1.63	1.50	1.74	1.55
	5	7.51	6.93	7.92	7.01
		triad_lex	**triad_rb**	**vanka_lex**	**vanka_rb**
V(1,1)	1	1.42	1.42	2.17	1.53
	5	6.48	6.23	9.79	7.00
F(1,1)	1	1.73	1.39	2.90	2.04
	5	7.95	6.26	11.3	9.05

sure, we set

$$p = \begin{cases} 1 & \text{on } \Gamma_1\text{:}\ \ |x| \leqslant 20,\ \ y = 100, \\ 0 & \text{on } \Gamma \backslash \Gamma_1 \end{cases}$$

The boundary conditions for the displacements are identical to the ones prescribed in Section 4.1. The grid size varies between $\frac{1}{64}$ and $\frac{1}{256}$; the time step is fixed, $\delta t = 1.0$. We present the V(1,1)-cycle convergence by means of the convergence factor and, in brackets, the number of iterations to reach the stopping criterion for some selected relaxation methods. The stopping criterion per time step was chosen as the absolute residual to be again less than 10^{-9}. Further, the CPU time spent until convergence is presented (Table V). From this table, the superiority of **dist_2lp_lin** among the ones presented becomes obvious. The triad smoother

Table V. V(1, 1) multigrid convergence factors, and, in brackets, the average number of iterations per time step, and CPU time (s) for different smoothers.

Smoother	**triad_lex**	**dist_2lp_lin**	**dist_2lp_rb**	**vanka_lex**
256^2	>50	0.08 (8) 22″	0.20 (14) 46″	0.45 (22) 144″
128^2	>50	0.08 (8) 5″	0.20 (14) 11″	0.54 (27) 44″
64^2	>50	0.08 (8) 1″	0.19 (13) 3″	0.50 (26) 11″

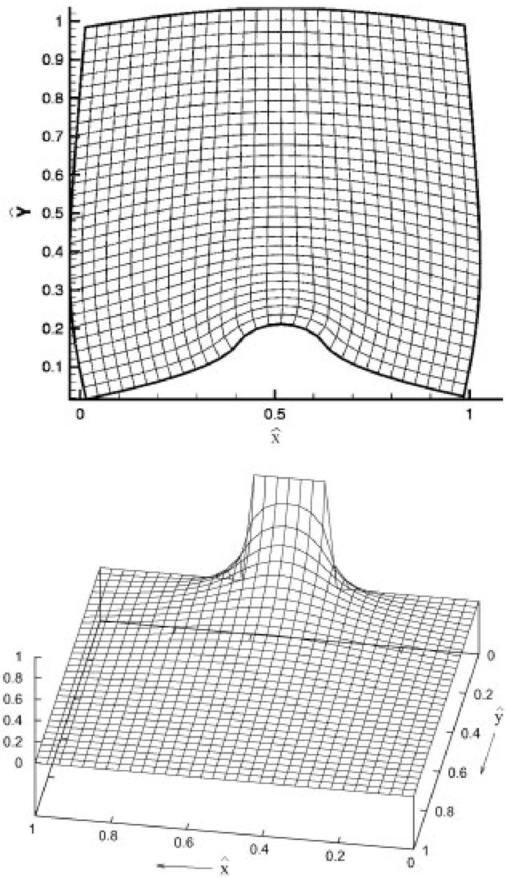

Figure 7. Numerical solution for displacement and pressure (in different orientations) for the 2D poroelasticity reference problem of Section 4.5.

fails to converge within 50 iterations. The red–black versions of coupled smoothing led to worse convergence. The other distributive smoothers performed well, as expected.

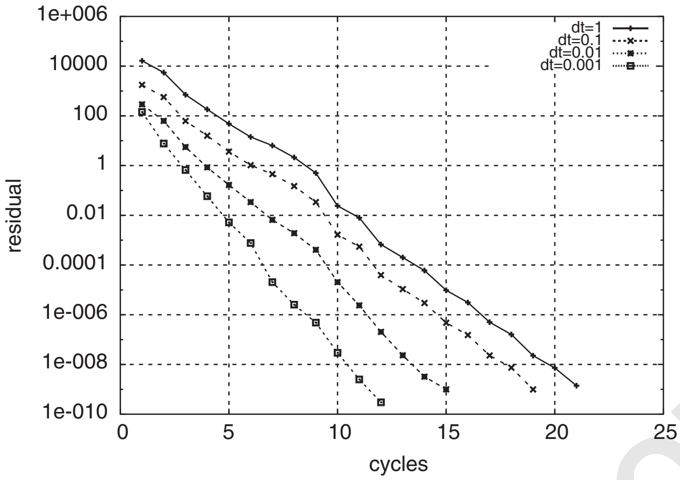

Figure 8. Multigrid convergence with **dist_2lp_lin** on a 16×128 grid, and varying time step.

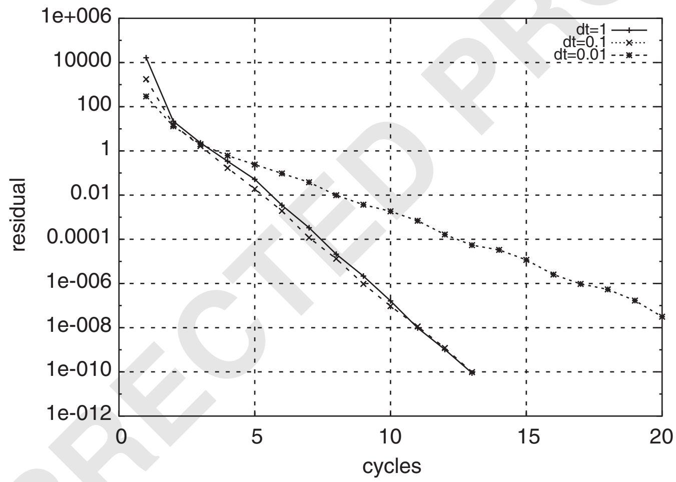

Figure 9. Multigrid convergence with **triad_lin** on a 16×128 grid, and varying time step.

4.5. Anisotropic grids; grid stretching

Another analytic solution is obtained with source term $Q = 0$ and a non-zero pressure boundary condition prescribed on the lower edge, as

$$p(x, y = 0, \hat{t}) = (H(x - 0.4) - H(x - 0.6)) \sin \hat{t}, \quad \hat{t} = (\lambda + 2\mu)at$$

with $H(\cdot)$ the Heaviside function. The displacement boundary conditions are as in Section 4.1. Also in this case, an analytic solution is obtained [23]. The numerical solution is depicted in Figure 7. In the solution a rapidly varying pressure at the lower boundary can be observed.

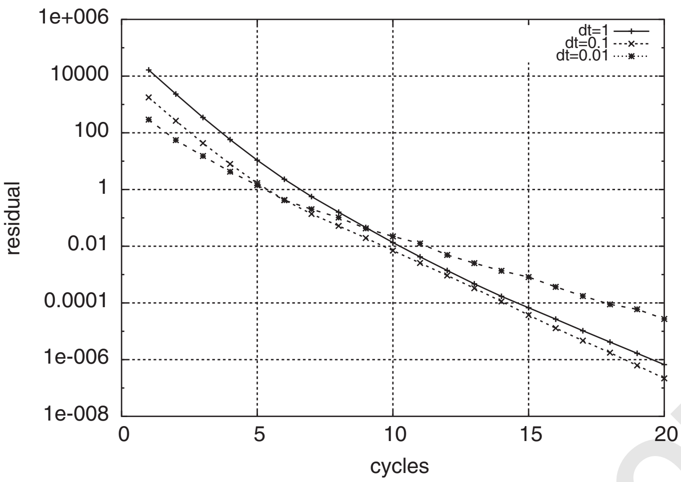

Figure 10. Multigrid convergence with **vanka_lin** on a 16×128 grid, and varying time step.

This test case serves as the evaluation of stretched grids in order to capture the pressure gradient accurately. The grids are chosen such that the line-wise versions of the distributive and coupled relaxation methods are most favourable. The computational domain $\Omega = (0,1)^2$ is discretized with 16×128 grid cells. We use four grids in the multigrid solver. Figure 8 presents the convergence for **dist_2lp_lin**, the distributive alternating line relaxation. The parameters used are $\lambda = \mu = a = 1$. The time step is varied; the plot presents the convergence with different time steps. Figure 8 shows a very satisfactory convergence with distributive relaxation for all values of δt. Figure 9 shows the corresponding convergence with the alternating line-wise triad smoother **triad_lin**. It can be observed in Figure 9 that also the line-wise version of the triad smoother is very sensitive to the size of the time step. For extremely small steps, the method no longer converges. For larger time steps, however, the convergence is satisfactory.

Figure 10 then shows the multigrid convergence with the coupled Vanka alternating line-wise smoother **vanka_lin**. For very small time steps, $\delta t = 0.001$, for example, also this line-wise version does not converge. Obviously, the line-wise distributive smoother is to be preferred, as it is most robust.

With respect to the computational costs of the different line-wise smoothers, the distributive smoother is clearly to be preferred. The coupled line-wise smoothers are at least 1.5 times more expensive than the distributive version.

In the distributive version, only tridiagonal systems need to be solved, whereas in the coupled line-wise smoothers more complicated block matrices must be inverted.

5. CONCLUSIONS

We evaluate multigrid solution methods for a fast solution of the incompressible poroelasticity equations. For stability reasons, a staggered grid discretization has been adopted.

For the system, we have compared distributive relaxation methods with two variants of coupled smoothing, triad-wise and cell-wise. The other multigrid components are based on standard grid coarsening, geometric transfer operators and a direct coarse grid discretization.

From the various systematic multigrid tests, in which many parameters have been varied, the methods based on distributive relaxation are the favourites. They are most efficient, and they are robust. The convergence of the methods based on distributive relaxation, especially of the multi-stage variant for the operator with a biharmonic term, and of the alternating line relaxation for the split operator, are highly efficient, insensitive to changes in the time step, or the Lamé coefficients. For implementation on parallel computers with distributed memory, the multi-stage variant is most easily parallelizable. As the distributive variant based on line smoothing is also able to deal with stretched grids, this method is to be preferred.

The coupled triad smoothers are not robust with respect to extremely small time steps. The coupled smoothers of Vanka-type are more robust, but most often more expensive than the distributive relaxation methods. This is especially true for the line-wise versions of coupled smoothing.

In this paper we have evaluated the asymptotic multigrid convergence of some highly efficient multigrid variants. For the most efficient methods the convergence is close to 0.10 for all the test problems presented. These excellent multigrid convergence factors are also the basis for full multigrid (FMG) methods, in which the iteration starts on the coarsest grid and, after reaching the finest grid, only one additional cycle is necessary to obtain the desired accuracy of the solution. For time-dependent problems full multigrid methods (i.e. starting each time step on the coarse grid) are somewhat artificial as a good starting guess, that is the solution of the previous time step, exists. But, in principle with the convergence factors presented, full multigrid techniques may provide even more efficient solvers for our problems.

REFERENCES

1. Brandt A, Dinar N. Multigrid solutions to elliptic flow problems. In Parter S (ed.). *Numerical Methods for Partial Differential Equations*. Academic Press: New York, 1979; 53–147.
2. Vanka SP. Block-implicit multigrid solution of Navier–Stokes equations in primitive variables. *Journal of Computational Physics* 1986; **65**:138–158.
3. Gaspar FJ, Lisbona FJ, Oosterlee CW, Wienands R. *An Efficient Multigrid Solver Based on Distributive Smoothing for Poroelasticity Equations*, submitted for publication.
4. Sivaloganathan S. The use of local mode analysis in the design and comparison of multigrid methods. *Computer Physics Communications* 1991; **65**:246–252.
5. Sockol P. Multigrid solution of the Navier–Stokes equations on highly stretched grids. *International Journal for Numerical Methods in Fluids* 1993; **17**:543–566.
6. Paisley MF, Bhatti NM. Comparison of multigrid methods for neutral and stably stratified flows over two-dimensional obstacles. *Journal of Computational Physics* 1998; **142**:581–610.
7. Wesseling P, Oosterlee CW. Geometric multigrid with applications to computational fluid dynamics. *Journal of Computational and Applied Mathematics* 2001; **128**:311–334.
8. Osorio JG, Chen H-Y, Teufel LW. Numerical simulation of the impact of flow–induced geomechanical response on the productivity of stress-sensitive reservoirs. *Society of Petroleum Engineers SPE* 1999; **51929**:1–15.
9. Brandt A, Yavneh I. On multigrid solution of high-Reynolds incompressible entering flows. *Journal of Computational Physics* 1992; **101**:151–164.
10. Ruge JW, Stüben K. Algebraic multigrid (AMG). In *Multigrid Methods*, *Frontiers in Applied Math*, Mc Cormick SF (ed.). SIAM: Philadelphia, 1987; 73–130.
11. Biot MA. General theory of three dimensional consolidation. *Journal of Applied Physics* 1941; **12**:155–164.
12. Gaspar FJ, Lisbona FJ, Vabishchevich PN. A finite difference analysis of Biot's consolidation model. *Application of Numerical Mathematics* 2003; **44**:487–506.
13. Harlow FH, Welch JE. Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface. *Physics of Fluids* 1965; **8**:2182–2189.

14. Wesseling P. *Principles of Computational Fluid Dynamics*. Springer: Berlin, 2001.
15. Wittum G. Multi-grid methods for Stokes and Navier–Stokes equations with transforming smoothers: algorithms and numerical results. *Numerische Mathematik* 1999; **54**:543–563.
16. Brandt A. *Multigrid Techniques*: *1984 Guide with Applications to Fluid Dynamics*, GMD-Studie No. 85. Sankt Augustin: Germany, 1984.
17. Stüben K, Trottenberg U. Multigrid methods: fundamental algorithms, model problem analysis and applications. In *Multigrid Methods*, *Lecture Notes in Mathematics*, Hackbusch W, Trottenberg U (eds), vol. 960. Springer: Berlin, 1982; 1–176.
18. Trottenberg U, Oosterlee CW, Schüller A. *Multigrid*. Academic Press: New York, 2001.
19. Wienands R. Extended local fourier analysis for multigrid: optimal smoothing, coarse grid correction and preconditioning. *Ph.D. Thesis*, University of Cologne, Germany, 2001.
20. Livne OE, Brandt A. Local mode analysis of multicolour and composite relaxation schemes. *Computational Mathematics and Applications*, 2003; (in press).
21. Oosterlee CW, Wesseling P. A robust multigrid method for a discretization of the incompressible Navier–Stokes equations in general coordinates. *Impact of Computers in Science and Engineering* 1993; **5**:128–151.
22. Thompson MC, Ferziger JH. An adaptive multigrid technique for the incompressible Navier–Stokes equations. *Journal of Computational Physics* 1989; **82**:94–121.
23. Barry SI, Mercer GN. Exact solutions for 2D time dependent flow and deformation within a poroelastic medium. *Journal of Applied Mechanics (ASME)* 1999; **66**:536–540.

