Available online at www.sciencedirect.com
ScienceDirect
Applied Numerical Mathematics 58 (2008) 1933–1950
APPLIED NUMERICAL MATHEMATICS
www.elsevier.com/locate/apnum

Multigrid relaxation methods for systems of saddle point type ☆

C.W. Oosterlee [a,b,*], F.J. Gaspar [c]

[a] *CWI, Center for Mathematics and Computer Science, Amsterdam, the Netherlands*
[b] *Delft University of Technology, Delft Institute of Applied Mathematics (DIAM), the Netherlands*
[c] *Departamento de Mathemática Aplicada, University of Zaragoza, Zaragoza, Spain*

Available online 23 November 2007

Abstract

In this paper, we give an overview of multigrid methods for two systems of equations, namely the Stokes equations and the incompressible poroelasticity equations. We emphasize the saddle point type aspect in these two systems and discuss their discretization on staggered and collocated grids. The basic problem is that of smoothing a system of equations that has a zero (or almost zero) block in the matrix for one of the unknowns. In particular, we discuss the coupled relaxation approach, with its "box-wise" and "line-wise" versions and distributive relaxation, that gives a decoupled system of equations for smoothing. For general systems of equations it is a challenge to design an efficient distributive relaxation scheme. This paper may help in finding one.

PACS: 02.60.Cb; 02.70.-c

Keywords: Multigrid; Coupled relaxation; Distributive relaxation; Stokes equations; Poroelasticity; Staggered grid; Collocated grid; System transformation

1. Introduction

We give an overview of efficient multigrid methods for two systems of equations, namely the Stokes equations and the incompressible poroelasticity equations. We emphasize the saddle point type aspect in the two systems and discuss solution methods for stable discretizations. Furthermore, we discuss transformations of the systems that allow for the most rapid solution methods. The paper contributes to the recently increased interest in saddle point type problems [1].

We combine the discussion of the two unrelated systems of equations in one paper, as for the development of efficient multigrid techniques they show many similarities. At the same time, the differences in the systems may help a reader setting up efficient methods for a different system. Multigrid methods for computational fluid dynamics systems, such as the incompressible Navier–Stokes equations, have been described in detail in [40,44–46]. With the system of poroelasticity equations we have gathered some new insights in the efficient multigrid treatment of saddle

☆ This research has been partially supported by the Dutch government through the national program BSIK: knowledge and research capacity, in the ICT project BRICKS (http://www.bsik-bricks.nl), theme MSV1, and by the INTAS project 03-50-4395, the Spanish project MEC/FEDER MTM 2004-019051 and the Diputación General de Aragón.
* Corresponding author.
E-mail addresses: c.w.oosterlee@tudelft.nl (C.W. Oosterlee), fjgaspar@unizar.es (F.J. Gaspar).

0168-9274/$30.00
doi:10.1016/j.apnum.2007.11.014

point type systems, that we would like to report on here. Furthermore, we have compared the efficiency of various multigrid relaxation techniques and different discretizations for a poroelasticity reference problem.

The stationary Stokes equations represent a linear saddle point PDE system of the following form:

$$\boldsymbol{L}_s\bar{\mathbf{u}} = \begin{pmatrix} -\tilde{\Delta} & \mathcal{G} \\ \mathcal{D} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{u} \\ p \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_1 \\ 0 \end{pmatrix} =: \boldsymbol{f}. \tag{1}$$

Here, $\bar{\mathbf{u}} = (\mathbf{u}, p)$, with $\mathbf{u}$ the velocity components and p the pressure, $\boldsymbol{f}$ is a forcing term, $\tilde{\Delta}$ represents the vector Laplace operator, and $\mathcal{G}$ and $\mathcal{D}$ are the linear gradient and divergence operators, respectively.

We include here the time-dependent incompressible system of poroelasticity equations [2] and its efficient numerical treatment. After a semi-discretization in time, this two-dimensional system can also be written as a block 2×2 system,

$$\boldsymbol{L}_p\bar{\mathbf{u}} = \begin{bmatrix} \mathcal{A} & \mathcal{G} \\ \mathcal{D} & -\tau \cdot \mathcal{B} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ f \end{bmatrix} =: \boldsymbol{f}. \tag{2}$$

Here, $\mathbf{u} = (u, v)$ is the deformation (in the x- and y-directions), p is the pore pressure, $\mathcal{A}$ is the 2D linear elasticity operator, $\mathcal{D}$ is the divergence operator, $\mathcal{G}$ is the gradient, and $\mathcal{B}$ is a Laplace type operator multiplied by the time step τ, and source term $f(\boldsymbol{x}, t)$.

We deliberately re-use the notation for the unknowns in the Stokes equations, as we will focus on the matrices appearing and not so much on physical interpretation.

For very small time steps, the size of the lower diagonal block entry in (2) can become arbitrarily small. So, the system may be viewed as a sort of singularly perturbed system related to

$$\boldsymbol{L}_p\bar{\mathbf{u}} = \begin{bmatrix} \mathcal{A} & \mathcal{G} \\ \mathcal{D} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \boldsymbol{f}. \tag{3}$$

The outline of this paper is as follows. First, a general discussion about multigrid for systems of equations is given in Subsection 1.1. The continuous systems related to (1) and (2) are briefly described in Subsections 1.2, 1.3, after which we discuss different stable discretizations of the two systems: On staggered grids in Section 2, a stable transformation of the systems in Section 3, and on collocated grids in Section 4. These discretizations are accompanied by the corresponding multigrid methods, in which the concepts of coupled and decoupled, distributive relaxation are emphasized. We restrict ourselves to Cartesian grids in this paper.

1.1. Multigrid for systems of equations

We need efficient iterative multigrid solution methods for solving the discrete counterparts of Eqs. (1) and (2):

$$\boldsymbol{L}_h\bar{\boldsymbol{u}}_h = \boldsymbol{f}_h,$$

where $\boldsymbol{L}_h$ represent some discretization of one of the continuous operators (later on we distinguish $\boldsymbol{L}_{s,h}$ from $\boldsymbol{L}_{p,h}$). Multigrid methods are motivated by the fact that many iterative methods, especially if they are applied to elliptic problems, have a smoothing effect on the error between an exact solution and a numerical approximation. A smooth discrete error can be well represented on a coarser grid, where its approximation is much cheaper. We can distinguish between algebraic multigrid (AMG) [31,40] and geometric multigrid [5,8,19,44,40]. In algebraic multigrid no information is used concerning the grid on which the governing partial differential equations are discretized. In geometric multigrid, coarse grids are defined based on a given fine grid, and coarse grid corrections are computed from the systems of equations discretized on the coarse grids. Constructing coarse grids from fine grids by agglomeration of fine grid cells is easy when the fine grid is structured, but not if the fine grid is unstructured. That is where algebraic multigrid becomes useful for the solution of systems of equations. Algebraic methods come, however, with a significant overhead in complexity and complication, especially for systems. The present paper is about geometric multigrid.

The design of efficient relaxation methods for the multigrid solution of *systems* of partial differential equations often requires special attention. The smoother should smooth the error for all unknowns in the equations (that are possibly of different type) of a system. A multigrid method with a suitable relaxation method can be developed on the basis of Fourier analysis [37,4,40,44,49]. Whereas Fourier smoothing analysis is feasible for systems of equations, the

implementation of two-grid analysis is not at all trivial. Available Fourier analysis software [49] is therefore especially helpful for systems of equations. We will not focus on Fourier analysis in this paper, but on the systems determinant.

A good indication for the appropriate choice of relaxation method for a system of equations can be derived from the systems' determinant. If the operators on the main diagonal (or their principal parts) form the determinant, smoothing is a straightforward matter. In that case, the differential operator that corresponds to the primary unknown in each equation is the leading operator. Therefore, a simple equation-wise decoupled relaxation method can efficiently be used. If, however, as is the case for the present systems, the main operators in a system are not in the desired position, the choice of an efficient smoother needs care. A first obvious choice in the case of strong off-diagonal operators in the differential system is *coupled* or *collective* smoothing: All unknowns in the system at a certain grid point are updated simultaneously.

Decoupled smoothing, however, is to be preferred for reasons of efficiency and simplicity. It is typically found in the distributive framework [6,51]. An elegant way to describe distributive relaxation is to introduce a *right preconditioner* in the smoothing procedure [50]. This means that we introduce new variables $\boldsymbol{w}_h$, where $\bar{\boldsymbol{u}}_h = \boldsymbol{C}_h \boldsymbol{w}_h$, and consider the transformed system $\boldsymbol{L}_h \boldsymbol{C}_h \boldsymbol{w}_h = \boldsymbol{f}_h$, with $\boldsymbol{C}_h$ chosen in such a way that the resulting operator $\boldsymbol{L}_h \boldsymbol{C}_h$ is suited for *decoupled* (non-collective) relaxation. This step transforms the system in such a way that, preferably, the operators in the determinant of the original system appear on the main diagonal of the transformed system, ready for decoupled smoothing.

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' $\delta \boldsymbol{w} = (\delta w_u, \delta w_v, \delta w_p)^T$ is computed,

$$\boldsymbol{L}_h \boldsymbol{C}_h \delta \boldsymbol{w}^{m+1} = \boldsymbol{r}_h^m \tag{4}$$

with residual $\boldsymbol{r}_h^m = \boldsymbol{f}_h - \boldsymbol{L}_h \boldsymbol{C}_h \boldsymbol{w}_h^m$. In the *corrector step*, the new approximation for $\bar{\boldsymbol{u}}_h$ is then added to the present approximation as

$$\bar{\boldsymbol{u}}_h^{m+1} = \bar{\boldsymbol{u}}_h^m + \delta \bar{\boldsymbol{u}}_h^{m+1} = \bar{\boldsymbol{u}}_h^m + \boldsymbol{C}_h \delta \boldsymbol{w}^{m+1}. \tag{5}$$

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

The description below enables one to *analyze* the smoothing properties of distributive schemes by means of Fourier analysis.

(1) Transform the system $\boldsymbol{L}_h \bar{\boldsymbol{u}}_h = \boldsymbol{f}_h$ to a "simpler" one by a suitable right-preconditioning with an operator $\boldsymbol{C}_h$ (the *distributor*).
(2) Choose a point- or line-wise relaxation process, for each of the equations of the transformed system separately, of the form

$$\boldsymbol{w}_h^{m+1} = \boldsymbol{w}_h^m + \boldsymbol{B}_h (\boldsymbol{f}_h - \boldsymbol{L}_h \boldsymbol{C}_h \boldsymbol{w}_h^m) \tag{6}$$

with $\boldsymbol{B}_h$ being some approximation of the inverse of $\boldsymbol{L}_h \boldsymbol{C}_h$.
(3) Reformulate this relaxation scheme in terms of the original operator and unknowns by using $\bar{\boldsymbol{u}}_h = \boldsymbol{C}_h \boldsymbol{w}_h$:

$$\bar{\boldsymbol{u}}_h^{m+1} = \bar{\boldsymbol{u}}_h^m + \boldsymbol{C}_h \boldsymbol{B}_h (\boldsymbol{f}_h - \boldsymbol{L}_h \bar{\boldsymbol{u}}_h^m). \tag{7}$$

Additional smoothing difficulties are met, if one of the operators on the systems' main diagonal after discretization equals zero or is very close to zero (i.e., with small parameters in front of derivatives). However, also for this situation different forms of coupled and (distributive) decoupled relaxation methods exist, which smooth the errors in all the unknowns effectively. The operator $\boldsymbol{L}_h \boldsymbol{C}_h$ is suited for decoupled relaxation, if $\boldsymbol{C}_h$ is chosen such that the zero (or almost zero) blocks disappear in $\boldsymbol{L}_h \boldsymbol{C}_h$. For the Stokes and for the poroelasticity equations, distributive relaxation methods are described in Subsections 2.2.2 and 2.2.3, respectively.

The research underlying these relaxation methods for incompressible flow problems dates basically back to the late 1970's and early 1980's [6,41]. A distributive, decoupled relaxation method for poroelasticity has been recently introduced in [13].

Coupled and decoupled smoothing approaches have their advantages and disadvantages. If a system of equations consists of elliptic and of other, non-elliptic, components, decoupled relaxation allows to choose different relaxation methods for the different operators appearing. However, for general systems of equations it may not be 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 systems' operator is transformed by the smoother but the boundary operator is sometimes not considered. A system transformation may also increase the order of the PDEs appearing in the system, and then the prescription of additional boundary conditions becomes necessary. A significant difference in cost between coupled and distributive relaxation lies in the line-wise treatment of the unknowns, which may be necessary in the case of stretched grids. The cost of a coupled line-wise relaxation step is substantially higher than of a decoupled line-wise relaxation. The latter can be set up as a tri- (or more) diagonal matrix, whereas in the coupled version all different unknowns at the line need to be updated simultaneously.

1.2. Incompressible Navier–Stokes equations

Here we describe the systems of interest in some more detail. The well-known incompressible Navier–Stokes equations read:

$$\begin{aligned} &\frac{D\mathbf{u}}{Dt} = \rho \boldsymbol{f}_1 - \nabla p + \mu \nabla^2 \mathbf{u}, \\ &\nabla \cdot \mathbf{u} = 0, \end{aligned} \tag{8}$$

where D/Dt is the total derivative, $\boldsymbol{u}$ represents the velocity components in the x-, y- and z-directions, p is the pressure, ρ the density, μ the viscosity of the fluid, and $\boldsymbol{f}_1$ is a body force. Often, the stationary 2D dimensionless form of the incompressible Navier–Stokes equations

$$-\Delta u + Re(uu_x + vu_y) + p_x = f_1, \tag{9}$$

$$-\Delta v + Re(uv_x + vv_y) + p_y = f_2, \tag{9}$$

$$u_x + v_y = 0 \tag{10}$$

is the starting point for the solution of flow problems. Here, $Re := UL/\nu$ is the Reynolds number, U is a characteristic velocity (the unit in terms of which u and v are measured), L is a characteristic length (the unit for x and y) and $\nu = \mu/\rho$ is the kinematic viscosity of the flow.

For $Re = 0$, we have the special case of the *Stokes equations*. The Stokes operator reads

$$\boldsymbol{L}_s = \begin{pmatrix} -\Delta & 0 & \partial_x \\ 0 & -\Delta & \partial_y \\ \partial_x & \partial_y & 0 \end{pmatrix}. \tag{11}$$

The zero block in the third row of the operator hampers a basic numerical treatment of this problem. The determinant of operator (11) is

$$\det(\boldsymbol{L}_s) = \Delta^2,$$

and it is clear that the off-diagonal blocks contribute to this determinant.

For high Reynolds numbers, occurring in many fluid flow applications, thin boundary layers occur, necessitating locally refined grids, giving rise to high mesh aspect ratios (10^4 or even more).

Often, a semi-heuristic turbulence model is used to predict time-averaged flow variables based on a compromise between accuracy, memory requirements and computing time. More complete models such as large-eddy simulation and direct numerical simulation of turbulence [12] are time-dependent and can be efficiently implemented with explicit time-stepping schemes, that do not require multigrid. In these models, however, an elliptic equation for the pressure appears that can easily be handled by classical scalar multigrid methods. In fact, for time-accurate fluid flow the main use of multigrid, for example in many engineering codes, is for some form of the elliptic equation for the pressure. It is, however, possible to benefit more from multigrid, especially for the steady equations, as we will discuss below.

For high Reynolds numbers, the non-linear terms dominate the viscous terms in (8) and the singular perturbation aspect of the system of equations becomes important. Efficient multigrid methods for convection dominating convection–diffusion operators are not straightforwardly defined, see [7,4,40]. In this paper, however, we do not discuss this aspect. We focus on the saddle point aspect, and therefore discuss the Stokes equations in detail.

If the equations are discretized by means of standard central differencing with all unknowns at grid vertices, pressure values are directly coupled only between grid points of distance $2h$. The same is true if all variables are located

at the cell centers. A so-called checkerboard instability arises from this straightforward discretization, leading to unphysical pressure fields. This has given rise to two basically different stable discretization approaches, on a staggered grid and on a collocated grid. For a discussion of the relative merits of staggered and collocated schemes, see, for example, [30,47]. The two approaches will be discussed in separate subsections.

1.3. Incompressible poroelasticity equations

The quasi-static Biot model [2] for soil consolidation can be formulated as a system of partial differential equations for displacements and pressure of the fluid. One assumes the material's solid structure to be linearly elastic, initially homogeneous and isotropic, the strains imposed within the material are small. We denote by $\bar{\boldsymbol{u}} = (u, v, p)^T$ the solution vector, consisting of the displacement vector $\boldsymbol{u} = (u, v)^T$ and pore pressure of the fluid p. The governing equations read

$$-\mu\tilde{\Delta}\mathbf{u} - (\lambda+\mu)\operatorname{grad}\operatorname{div}\mathbf{u} + \alpha\operatorname{grad} p = \mathbf{0}, \quad \mathbf{x}\in\Omega, \tag{12}$$

$$\alpha\frac{\partial}{\partial t}(\operatorname{div}\mathbf{u}) - \frac{\kappa}{\eta}\Delta p = f(\mathbf{x},t), \quad 0 < t \leqslant T, \tag{13}$$

where λ and μ are the Lamé coefficients; κ is the permeability of the porous medium, η the viscosity of the fluid, α is the Biot–Willis constant (set to one here) and $\tilde{\Delta}$ represents the vector Laplace operator. The quantity $\operatorname{div}\mathbf{u}(\mathbf{x},t)$ is the dilatation, i.e. the volume increase rate of the system, a measure of the change in porosity of the soil. Source term $f(\mathbf{x},t)$ represents a forced fluid extraction or injection process.

We assume homogeneous Dirichlet boundary conditions, $\mathbf{u}(\mathbf{x},t) = 0$, $p(\mathbf{x},t) = 0, \mathbf{x}\in\partial\Omega$. The initial state satisfies $\operatorname{div}\mathbf{u}(\mathbf{x},0) = 0, \mathbf{x}\in\Omega$.

The incompressible 2D variant of Biot's consolidation model reads

$$\begin{aligned}
&-(\lambda+2\mu)u_{xx} - \mu u_{yy} - (\lambda+\mu)v_{xy} + p_x = 0,\\
&-(\lambda+\mu)u_{xy} - \mu v_{xx} - (\lambda+2\mu)v_{yy} + p_y = 0,\\
&(u_x + v_y)_t - a(p_{xx} + p_{yy}) = f
\end{aligned} \tag{14}$$

(plus initial and boundary conditions) with $a = \kappa/\eta > 0$. We consider a *model operator* $\boldsymbol{L}_p$,

$$\boldsymbol{L}_p = \begin{pmatrix} -(\lambda+2\mu)\partial_{xx} - \mu\partial_{yy} & -(\lambda+\mu)\partial_{xy} & \partial_x \\ -(\lambda+\mu)\partial_{xy} & -\mu\partial_{xx} - (\lambda+2\mu)\partial_{yy} & \partial_y \\ \partial_x & \partial_y & -\tilde{a}(\partial_{xx}+\partial_{yy}) \end{pmatrix}. \tag{15}$$

$\boldsymbol{L}_p$ can be interpreted as a "stationary variant" of (14), i.e. the operator after an implicit (semi-)discretization in time. For example, in case of Crank–Nicholson time discretization we have $\tilde{a} = 0.5a\tau$. From (15) one may calculate the corresponding determinant:

$$\det(\boldsymbol{L}_p) = -\mu\Delta\big(\tilde{a}(\lambda+2\mu)\Delta^2 - \Delta\big) \tag{16}$$

with Laplace operator Δ and biharmonic operator Δ^2. The principal part of $\det(\boldsymbol{L}_p)$ is Δ^m with m depending on the choice of λ, μ, and $\tilde{a}$. For physical reasons, we always have μ, $\tilde{a}$, $\lambda+2\mu > 0$, yielding $m = 3$. The number of boundary conditions that must accompany $\boldsymbol{L}_p$ is m [4,40].

Physically, when a load is applied in a poroelasticity problem, the pressure suddenly increases and a boundary layer appears in the early stages of the time-dependent process. In the case of an unstable discretization, unphysical oscillations can occur in the first time steps of the solution. After this phase, the solution shows a much smoother behaviour. The time-dependent operator (15) suffers from stability difficulties. The coefficient in the $L_p^{3,3}$-block in (15) is typically, depending on the time step, very small. In order to avoid oscillating solutions, the discretization has to be designed with care.

2. Classical approach: Staggered grid discretization and multigrid treatment

One remedy for the instability in the case of the Stokes and the incompressible poroelasticity equations is to use a *staggered* distribution of unknowns [20] instead of a non-staggered one. The three primary unknowns, u, v and p, are

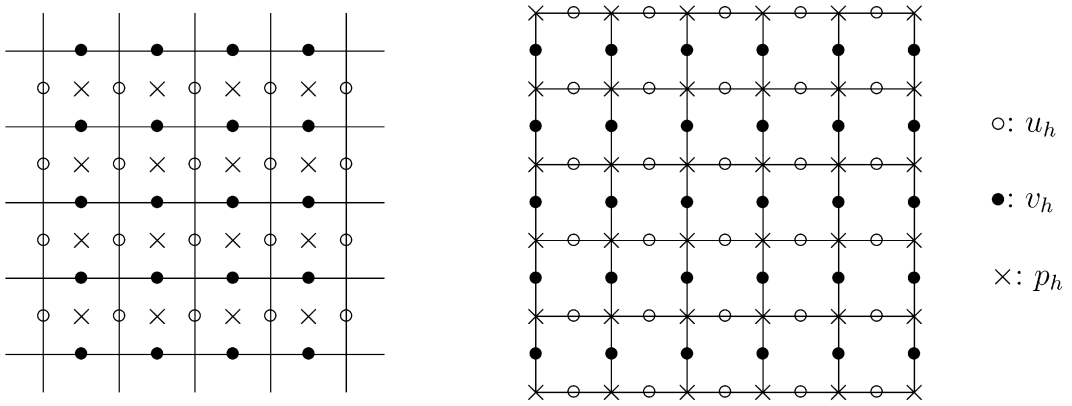

Fig. 1. Staggered grid location of unknowns for Stokes equations (left picture) and for incompressible poroelasticity equations (right picture).

defined at different positions on the grid. On a staggered grid, discretization employs nearest neighbor central finite differences.

2.1. Staggered grid discretization

In a staggered arrangement for the incompressible Stokes equations, the discrete pressure unknowns p_h are defined at cell centers (the $\times$-points), and the discrete values of u_h and v_h are located at the grid cell faces in the $\circ$- and $\bullet$-points, respectively, see Fig. 1 (left side).

The discrete analog of the continuity equation, $u_x + v_y = 0$, is defined in the $\times$-points and the discrete x- and y-momentum equations are located in the $\circ$- and the $\bullet$-points, respectively. Altogether, the staggered discretization of the *Stokes* equations leads to the system

$$\boldsymbol{L}_{s,h}\bar{\boldsymbol{u}}_h = \begin{pmatrix} -\Delta_h & 0 & (\partial_x)_h \\ 0 & -\Delta_h & (\partial_y)_h \\ (\partial_x)_h & (\partial_y)_h & 0 \end{pmatrix} \begin{pmatrix} u_h \\ v_h \\ p_h \end{pmatrix} = \begin{pmatrix} f_1 \\ f_2 \\ 0 \end{pmatrix}. \tag{17}$$

The determinant of the discrete operator is $\det(\boldsymbol{L}_{s,h}) = (\Delta_h)^2$:

The staggered approach leads to a natural discrete form of the continuity equation on a Cartesian grid: In accordance with the continuous problem, pressure values are not needed at physical boundaries.

On more general, curvilinear or unstructured grids, accurate staggering is much more complicated, as grid-oriented velocity components need to be employed. No special measures are required, but the scheme must be formulated carefully to maintain accuracy on rough grids; see [47].

Also the poroelasticity system has been discretized on a staggered grid to cope with the numerical instabilities in the time-dependent process. Pressure points in the staggered grid are located at the physical boundary, and the displacement points are defined at the cell faces, as in poroelasticity applications pressure is often prescribed at the boundary. Notice further that the "3, 3"-block in the poroelasticity system is not equal to zero.

The divergence operator is naturally approximated by a central discretization of the displacements in a cell. A staggered grid suitable for poroelasticity is therefore slightly different from the one previously discussed, see Fig. 1 (right side).

The discretization of each equation, centered around the equation's primary unknown, reads in this case

$$\boldsymbol{L}_{p,h}\bar{\boldsymbol{u}}_h = \begin{pmatrix} -(\lambda+\mu)(\partial_{xx})_h - \mu\Delta_h & -(\lambda+\mu)(\partial_{xy})_h & (\partial_x)_h \\ -(\lambda+\mu)(\partial_{xy})_h & -\mu\Delta_h - (\lambda+\mu)(\partial_{yy})_h & (\partial_y)_h \\ (\partial_x)_h & (\partial_y)_h & -\tilde{a}\Delta_h \end{pmatrix} \begin{pmatrix} u_h \\ v_h \\ p_h \end{pmatrix} = \boldsymbol{f}_h, \tag{18}$$

with $\boldsymbol{f}_h = (0, 0, f_h)^T$, and the subscripts denote central discrete operators on the staggered grid. The determinant of the discrete version of (15) is given by

$$\det(\boldsymbol{L}_{p,h}) = -\mu\Delta_h\big(\tilde{a}(\lambda+2\mu)\Delta_h^2 - \Delta_h\big), \tag{19}$$

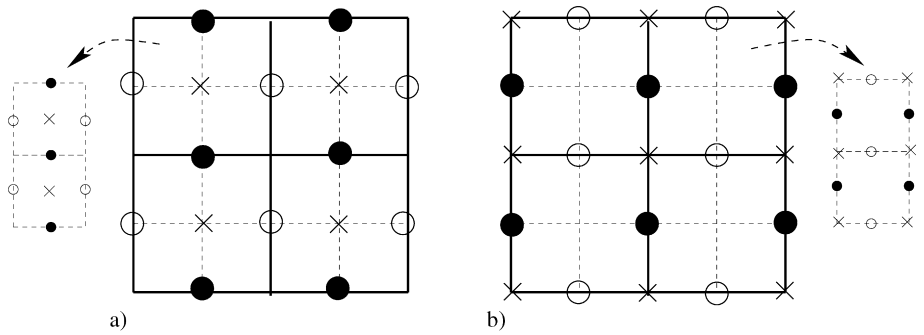

Fig. 2. Fine and coarse grid cells and staggered grid unknowns, ○: u_H point, •: v_H point, ×: p_H point; (a) for Stokes, (b) for poroelasticity.

where the discrete Laplacian and the discrete biharmonic operator are represented by the following stencils

$$-\Delta_h \mathrel{\hat{=}} \frac{1}{h^2}\begin{bmatrix} & -1 & \\ -1 & 4 & -1 \\ & -1 & \end{bmatrix}_h, \qquad \Delta_h^2 \mathrel{\hat{=}} \frac{1}{h^4}\begin{bmatrix} & & 1 & & \\ & 2 & -8 & 2 & \\ 1 & -8 & 20 & -8 & 1 \\ & 2 & -8 & 2 & \\ & & 1 & & \end{bmatrix}_h. \tag{20}$$

Stability of this discretization has been proved rigorously in [16].

2.2. Multigrid on staggered grids

In this context, efficient multigrid solvers for the system of equations discretized on staggered grids are necessary. In multigrid methods for Cartesian staggered grid discretizations one chooses standard geometric grid coarsening, i.e., the sequence of coarse grids is obtained by doubling the mesh size in each space direction. An appropriate coarse grid correction consists of geometric transfer operators $R_{h,2h}$, $P_{2h,h}$, and a direct coarse grid discretization (i.e., a coarse grid analog of $\boldsymbol{L}_h$).

There is no real benefit in using the Galerkin coarse grid discretization. The Galerkin coarse grid discretization merely results in a larger stencil here. For real-life poroelasticity problems with jumps in the permeability coefficient κ, however, we may need to reconsider Galerkin coarse grid operators, as they lead to natural coarse grid operators for problems with discontinuous coefficients.

The transfer operators that act on the different unknowns are dictated by the staggered grid, see Fig. 2. For the Stokes and poroelasticity equations, the transfer operators are very similar: At u- and v-grid points one considers 6-point restrictions and at p-grid points a 4-point cell-centered restriction for Stokes and a 9-point vertex-centered full-weighting restriction for poroelasticity. In stencil notation the restriction operators *for poroelasticity* are given by

$$R^u_{h,2h} \mathrel{\hat{=}} \frac{1}{8}\begin{bmatrix} 1 & & 1 \\ 2 & \star & 2 \\ 1 & & 1 \end{bmatrix}_h, \quad R^v_{h,2h} \mathrel{\hat{=}} \frac{1}{8}\begin{bmatrix} 1 & 2 & 1 \\ & \star & \\ 1 & 2 & 1 \end{bmatrix}_h, \quad R^p_{h,2h} \mathrel{\hat{=}} \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}_h, \tag{21}$$

respectively. The restriction operator for u in the staggered Stokes case resembles the operator for v in poroelasticity and vice versa. As the prolongation operators $P^{u/v/p}_{2h,h}$, one applies the usual interpolation operators based on bilinear interpolation of neighboring coarse grid unknowns in the staggered grid.

The interesting multigrid aspect lies in the choice of relaxation method.

2.2.1. Coupled relaxation

Coupled smoothing on staggered grids basically comes in 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 in (17) and (18) is taken into account in a more sophisticated way.

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. Fig. 3(a) shows a triad for the Stokes equations; Fig. 3(c)

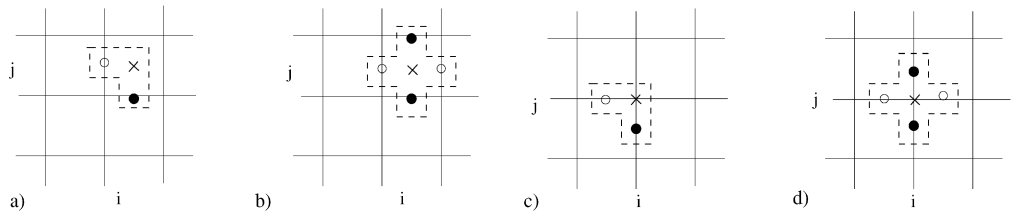

Fig. 3. Coupled relaxation: (a) triad-wise for Stokes, (b) box-wise for Stokes, (c) triad-wise for poroelasticity, (d) box-wise for poroelasticity; $\times$: p_h, $\circ$ u_h, $\bullet$: v_h.

for poroelasticity. In the "triad-wise" variant, a small 3×3-matrix must be solved, for all triads in the staggered grid. However, it is reported in [41] that the triad smoother is not satisfactory for Stokes equations. A better alternative is a coupled smoother, that locally updates all unknowns appearing in the divergence operator in the third equation simultaneously. In practice, this means that instead of the three unknowns in a triad, five unknowns (pressure $p_{i,j}$, 2 times u_h- and v_h-unknowns, $u_{i,j}, u_{i+1,j}, v_{i,j}, v_{i,j+1}$, centered around a pressure point) should be relaxed simultaneously. This "box-wise" smoothing is shown in Fig. 3(b) for the Stokes equations, and in Fig. 3(d) for poroelasticity. A small 5×5-matrix must be inverted for each box. It is convenient to consider the *correction* equations

$$\begin{pmatrix} a_{1,1} & a_{1,2} & a_{1,3} & a_{1,4} & a_{1,5} \\ a_{2,1} & a_{2,2} & a_{2,3} & a_{2,4} & a_{2,5} \\ a_{3,1} & a_{3,2} & a_{3,3} & a_{3,4} & a_{3,5} \\ a_{4,1} & a_{4,2} & a_{4,3} & a_{4,4} & a_{4,5} \\ a_{5,1} & a_{5,2} & a_{5,3} & a_{5,4} & a_{5,5} \end{pmatrix} \begin{pmatrix} \delta u_{i,j} \\ \delta v_{i,j} \\ \delta u_{i+1,j} \\ \delta v_{i,j+1} \\ \delta p_{i,j} \end{pmatrix}^{m+1} = \begin{pmatrix} r^1_{i,j} \\ r^2_{i,j} \\ r^1_{i+1,j} \\ r^2_{i,j+1} \\ r^3_{i,j} \end{pmatrix}^{m}, \tag{22}$$

where $\boldsymbol{\delta}\bar{\mathbf{u}}^{m+1}_{i,j} = \bar{\mathbf{u}}^{m+1}_{i,j} - \bar{\mathbf{u}}^{m}_{i,j}$ and $(r^l_{i,j})^m$ represents the residual of equation l ($l = 1, 2, 3$) of the system corresponding to node (i, j). Afterwards, the correction is added to the current approximation, possibly with a relaxation parameter,

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

Coefficient $a_{5,5}$ is zero for the staggered Stokes discretization and it is a very small entry for the poroelasticity system. During the elimination process $a_{5,5}$ is replaced by larger elements, though. In the correction equation setting, it may be possible to discard certain elements in (22), for example, the elements $a_{1,2}, a_{2,1}$ related to the mixed derivatives. For the incompressible Navier–Stokes equations, the box-wise relaxation method is called the "Vanka smoother" after the author of the first paper [41]. In one iteration all displacement unknowns are updated twice, whereas pressure unknowns are updated once.

The boxes can be processed in different orderings. An obvious choice for box numbering is the lexicographic Gauss–Seidel ordering, but also the red-black Gauss–Seidel ordering may be chosen. For the Stokes equations both numbering choices are perfectly fine. However, for incompressible Navier–Stokes flow problems at high Reynolds numbers, the processing order of the boxes in the coupled relaxation should also take the convection aspect in the momentum equations into consideration, as all equations are relaxed simultaneously. In that case, a lexicographical ordering of boxes, in alternating directions, is often to be preferred.

The relaxation can be performed in box-wise or in a line-wise fashion if grid anisotropies occur in a problem. Zebra or lexicographic line Gauss–Seidel ordering is then appropriate. It has been used in the CFD context in [25,39]. For each line a block tridiagonal matrix has to be inverted. Fig. 4(a) presents the line-wise variant of this coupled smoothing process for the Stokes equations, and Fig. 4(b) for the poroelasticity equations. In [26,13] multigrid has also been used for staggered poroelasticity, based on coupled smoothing. The cost of a coupled line-wise iteration is, however, substantial.

2.2.2. Distributive relaxation, Stokes equations

Whereas box relaxation is clearly of collective type, so-called distributive relaxation has been proposed in the literature about decoupled relaxation schemes. For a theoretical description and corresponding analysis, we refer to [50,51].

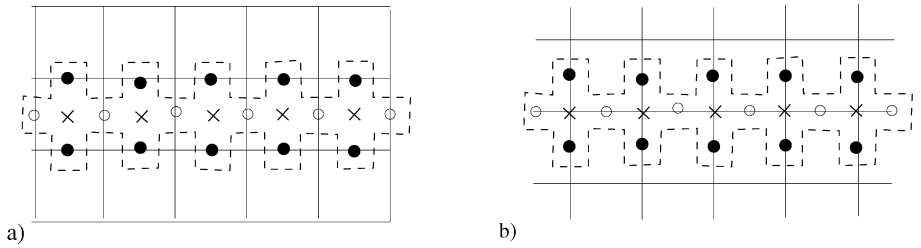

Fig. 4. x-Line-box-wise coupled relaxation: (a) for the Stokes equations, (b) for poroelasticity, $\times$: p_h, $\circ$ u_h, $\bullet$: v_h.

For the Stokes and incompressible Navier–Stokes equations, the first relaxation method of distributive type was introduced in the late seventies by Brandt and Dinar [6]. The smoother was called distributive Gauss–Seidel, DGS. In matrix form it can be represented by the preconditioner

$$\boldsymbol{C}_h = \begin{pmatrix} I_h & 0 & -(\partial_x)_h \\ 0 & I_h & -(\partial_y)_h \\ 0 & 0 & -\Delta_h \end{pmatrix}. \tag{23}$$

The corrections are obtained by a standard lexicographic Gauss–Seidel type relaxation with respect to the operator

$$\boldsymbol{L}_{s,h}\boldsymbol{C}_h = \begin{pmatrix} -\Delta_h & 0 & 0 \\ 0 & -\Delta_h & 0 \\ (\partial_x)_h & (\partial_y)_h & -\Delta_h \end{pmatrix}. \tag{24}$$

For DGS, the Fourier smoothing analysis can be performed in terms of the product operator $\boldsymbol{L}_{s,h}\boldsymbol{C}_h$. Here, the smoothing properties only depend on the diagonal terms Δ_h. The DGS smoothing factor μ equals that of point-wise lexicographic Gauss–Seidel smoothing for $\boldsymbol{L}_{s,h}\boldsymbol{C}_h$ and thus is equal to $\mu(-\Delta_h) = 0.5$ [6]. A detailed two-grid analysis can be found in [24], where also stretched grids are included in the discussion.

2.2.2.1. Pressure correction relaxation. The SIMPLE algorithm [28] (Semi-Implicit Method for Pressure-Linked equations) is another example of a distributive scheme. It belongs to the class of the so-called *pressure correction schemes*, which have been used as smoothers for example, in [33]. For Stokes equations, they are obtained by using a distributor of the form

$$\boldsymbol{C}_h = \begin{pmatrix} I_h & 0 & -\hat{\Delta}_h^{-1}(\partial_x)_h \\ 0 & I_h & -\hat{\Delta}_h^{-1}(\partial_y)_h \\ 0 & 0 & I_h \end{pmatrix} \tag{25}$$

with $-\hat{\Delta}_h$ being an approximation of $-\Delta_h$. A suitably chosen relaxation scheme is then employed for the resulting product system $\boldsymbol{L}_h\boldsymbol{C}_h$ (e.g. line Gauss–Seidel successively for velocity and pressure). Notice that this $\boldsymbol{C}_h$ does not lead to a fully decoupled system $\boldsymbol{L}_h\boldsymbol{C}_h$.

Smoothing analysis results for pressure correction schemes are available in [33]. The resulting smoothing factor is $\mu = 0.6$ for Stokes equations. In [3], the pressure correction smoothers are, however, not recommended as smoothers for the stationary incompressible Navier–Stokes equations.

2.2.2.2. Discussion. Box relaxation and DGS work equally well for low Reynolds numbers. Box relaxation is, however, somewhat more expensive than DGS. For higher Reynolds numbers, different distributive schemes for smoothing $\boldsymbol{L}_h\boldsymbol{C}_h$ are at hand. One can choose for line-wise Gauss–Seidel relaxation, or for ILU-type smoothing [52], that makes this smoother certainly more robust. It is even possible to choose for flow following or alternating line-wise smoothing for the convection–diffusion part for the incompressible Navier–Stokes equations. 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 [35], 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 dominated flows. In [36], a distributive smoother is evaluated together with box- and line-wise versions of the coupled smoother.

The coupled smoother comes out best, but in [27] 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 [46]. In general, we argue that on the basis of an analysis of the $\boldsymbol{L}_h\boldsymbol{C}_h$-system, possibly in combination with special relaxations at domain boundaries, distributive relaxation should be preferred.

2.2.3. Distributive relaxation, poroelasticity equations

Here, we consider decoupled relaxation for the system of incompressible poroelasticity equations discretized on the staggered grid, see also [13,48]. In order to relax $\boldsymbol{L}_{p,h}\bar{\boldsymbol{u}}_h = \boldsymbol{f}_h$, a new variable $\boldsymbol{w}_h$ defined by $\bar{\boldsymbol{u}}_h = \boldsymbol{C}_h\boldsymbol{w}_h$ was introduced and system $\boldsymbol{L}_{p,h}\boldsymbol{C}_h\boldsymbol{w}_h = \boldsymbol{f}_h$ has been considered in smoothing. The resulting system is suited for *decoupled* smoothing, i.e., each equation can be treated separately. The distributor reads [48]

$$\boldsymbol{C}_h = \begin{pmatrix} I_h & 0 & -(\partial_x)_h \\ 0 & I_h & -(\partial_y)_h \\ (\lambda+\mu)(\partial_x)_h & (\lambda+\mu)(\partial_y)_h & -(\lambda+2\mu)\Delta_h \end{pmatrix} \tag{26}$$

with identity I_h. Then, the system reads

$$\boldsymbol{L}_{p,h}\boldsymbol{C}_h = \begin{pmatrix} -\mu\Delta_h & 0 & 0 \\ 0 & -\mu\Delta_h & 0 \\ LC_h^{3,1} & LC_h^{3,2} & \tilde{a}(\lambda+2\mu)\Delta_h^2 - \Delta_h \end{pmatrix} \quad \text{with} \tag{27}$$

$$LC_h^{3,1} = (\partial_x)_h - \tilde{a}(\lambda+\mu)\big((\partial_{xxx})_h + (\partial_{xyy})_h\big), \tag{28}$$

$$LC_h^{3,2} = (\partial_y)_h - \tilde{a}(\lambda+\mu)\big((\partial_{xxy})_h + (\partial_{yyy})_h\big), \tag{29}$$

where the discrete operators with third derivatives read (in stencil notation),

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

$$(\partial_{xxy})_h \triangleq \frac{1}{h^3}\begin{bmatrix} 1 & -2 & 1 \\ & \star & \\ -1 & 2 & -1 \end{bmatrix}_h.$$

The other discrete operators are given by analogous stencils. Notice that the diagonal elements of the triangular $\boldsymbol{L}_{p,h}\boldsymbol{C}_h$ are factors of $\det(\boldsymbol{L}_{p,h})$, which is a highly desirable feature. Equation-wise smoothing for (27) is now an option, starting with the first equation in the system, etc.

The first two equations in (4) 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) [37,40], which is efficiently parallelizable.

The challenging task here is to find a highly efficient smoother for the third equation in (4),

$$\big(\tilde{a}(\lambda+2\mu)\Delta_h^2 - \Delta_h\big)\delta w_p = r_{3,h} \tag{30}$$

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

There are several ways of smoothing the operator in (30). Good smoothing factors are obtained with an overrelaxation parameter ω in red-black Jacobi point relaxation, as shown in [13].

A different approach for the third equation, that avoids smoothing directly for a biharmonic operator is found by splitting the operator in the third equation, as follows

$$-\Delta_h q_h = r_{3,h}, \qquad \big(-\tilde{a}(\lambda+2\mu)\Delta_h + I_h\big)\delta w_p = q_h \tag{31}$$

with an extra slack variable q_h, and discrete identity I_h. This way, we deal with simple operators of Laplace-type for smoothing. The two operators in (31) can be smoothed again with red-black Gauss–Seidel iteration, but also with line-wise Gauss–Seidel relaxation methods. Of course, line-wise relaxation methods are mainly applicable on structured grids, as we have them here in our applications. Alternatives for line-wise relaxation, that are also useful for unstructured grid applications, are found in the incomplete ILU decomposition based smoothers [52].

The distributive relaxation is designed such that its performance should be independent of problem parameters, like the Lamé coefficients or the time step. The multigrid scheme works very well, with convergence factors less than 0.2 for a wide range of problem parameters [13,48]. In fact, we have shown in [13] that distributive relaxation, with line-wise smoothing for the resulting scalar equations, gives robust and fast multigrid convergence for a variety of problems. It is more robust than coupled relaxation, especially when very small time steps are involved.

3. Transformed system of equations and multigrid treatment

It is possible to reformulate the systems under consideration in such a way that standard collective relaxations become appropriate for second order accurate central discretizations. The applicability of these reformulations is typically limited to certain classes of problems, like problems with constant problem coefficients, sufficiently smooth solutions, etc., because they are based on operator manipulations that hold for constant parameters. An interesting aspect is that the transformed systems can be discretized on a collocated grid in a straightforward way, since a *stable discretization* results. This is explained below.

An obvious question leading to the transformed systems of equations is the following: If a transformation of the form $\boldsymbol{L}_h\boldsymbol{C}_h$ (4) gives an operator that can be relaxed well, why not *solve* for this right-preconditioned system rather than only apply relaxation? Especially if the transformation leads to (almost) decoupled systems this may be an interesting alternative. The strategy followed here is that we will *first* transform a system and then discretize (so subscript h refers to discretization of $\boldsymbol{LC}$ then).

A difficulty when solving $\boldsymbol{LCw} = \boldsymbol{f}$ may arise at the boundaries, due to the transformation of $\boldsymbol{u}$ to $\boldsymbol{w}$, since the boundary conditions must be changed accordingly. This can be avoided by adopting a *left-preconditioner* instead. The iterative solution procedure for the transformed system of equations can then be interpreted as a left distributor for

$$\boldsymbol{CL}\bar{\boldsymbol{u}} = \boldsymbol{Cf}.$$

Notice that also the preconditioning method proposed by Elman, Wathen and Silvester, see, for example, [11] falls into these categories of left- and right-preconditioning. An automated approach for transforming systems of equations in order to obtain new formulations that are more accessible to numerical solution is described in [18].

3.1. Transformed Stokes and Navier–Stokes equations

3.1.1. SIMPLE

A well-known *classical solver* for the incompressible Navier–Stokes equations is the so-called SIMPLE algorithm [28]. As mentioned before, this is a right-preconditioned system. The SIMPLE and related algorithms are iterative solvers, which treat the momentum equations and a "pressure equation" separately in an "outer" iteration. Within this iteration, the pressure is updated using a Poisson-type equation. For such single grid codes with an outer iteration, it is possible to replace the most time-consuming component of the solver, i.e. the solution of the Poisson-like equation for the pressure, by an efficient multigrid solver. Although this approach does reduce the computing times, the overall convergence will be unchanged since the outer SIMPLE iteration is not accelerated, and the resulting system is not decoupled. On the other hand, the SIMPLE approach allows a relatively straightforward numerical solution of more general and complicated systems than the Navier–Stokes equations. It is, for example, relatively easy to add turbulence equations in the SIMPLE framework.

3.1.2. Streamfunction–vorticity

Another possibility to reformulate the incompressible Navier–Stokes equations, particularly suited in 2D, is the so-called streamfunction–vorticity formulation. Here, the unknowns to be solved for are the vorticity, $\boldsymbol{\omega} = \nabla \times \boldsymbol{u}$, and the streamfunction, Ψ, where $u = \partial\Psi/\partial y$, $v = -\partial\Psi/\partial x$. In this formulation, the solution of the system reduces to the solution of two Poisson-like equations for the streamfunction and the vorticity, respectively. Multigrid methods have been developed which are appropriate even for high Reynolds numbers (see, for example, [17]). In 3D, however, this approach becomes more complicated and looses its attractive features, as it requires the determination of 6 unknown functions (three components of the vorticity and of the velocity).

3.1.3. Efficient transformation

As an example of a transformation into a system that can efficiently be solved, we consider the following reformulation of the Stokes system. An interesting aspect is that it is not necessary to discretize on a staggered grid for a stable discretization, because the equations of the transformed system are decoupled and one can apply standard schemes to obtain M-matrices. For simplicity, we consider a Stokes problem with homogeneous Dirichlet boundary conditions:

$$\begin{aligned}(\mathbf{S}):\quad & -\tilde{\Delta}\mathbf{u} + \mathcal{G}p = \mathbf{f} \quad \text{in } \Omega,\\ & \mathcal{D}\mathbf{u} = 0 \quad \text{in } \Omega.\end{aligned}$$

Using the identities

$$\mathcal{D}\mathcal{G} = \Delta, \qquad \mathcal{D}\tilde{\Delta} = \Delta\mathcal{D}, \tag{32}$$

it is easy to verify that problem (**S**) is equivalent to the following transformed problem (assuming sufficient regularity of the solution)

$$\begin{aligned}(\mathbf{S^t}): \quad & -\tilde{\Delta}\mathbf{u} + \mathcal{G}p = \mathbf{f} \quad \text{in } \Omega,\\ & \Delta p = \mathcal{D}\mathbf{f} \quad \text{in } \Omega,\\ & \mathcal{D}\mathbf{u} = 0 \quad \text{on } \partial\Omega.\end{aligned}$$

If we now consider a collocated placement for the unknowns and central second order discretizations to approximate the operators $\mathcal{G}$ and $\mathcal{D}$ the following discrete problems are obtained respectively

$$(\mathbf{S_h}): \quad \begin{pmatrix} -\Delta_h & 0 & (\partial_x)_h \\ 0 & -\Delta_h & (\partial_y)_h \\ (\partial_x)_h & (\partial_y)_h & 0 \end{pmatrix}, \qquad (\mathbf{S_h^t}): \quad \begin{pmatrix} -\Delta_h & 0 & (\partial_x)_h \\ 0 & -\Delta_h & (\partial_y)_h \\ 0 & 0 & -\Delta_h \end{pmatrix}.$$

Notice that the collocated grid provides us an unstable discretization if it is used for the original problem (**S**), but a stable solution after discretization of the transformed problem ($\mathbf{S^t}$). The reason is that the discrete operators do not satisfy (32), in particular equality $D_h G_h = \Delta_h$ is not verified on a collocated grid. Whereas the discrete operators defined on the staggered grids indeed satisfy (32), the corresponding discrete operators on the collocated grid are not *mimetic* [21,22], see also [43], i.e., they do not mimic the differential operators well enough. Therefore, the collocated grid discretization can be stable for problem ($\mathbf{S^t}$), but unstable for an equivalent continuous problem (**S**).

The operators to be inverted for the discrete transformed system above are *scalar Laplace type operators*, for which geometric multigrid methods for scalar equations converge extremely well. In particular, multigrid method with highest efficiency, based on a red-black point-wise Gauss–Seidel smoother, and well-known choices for the remaining multigrid components [40] can be used. These include the direct coarse grid discretization of the transformed system of equations, full weighting and bilinear interpolation, as the restriction and prolongation operators, respectively.

An equivalent incompressible Navier–Stokes transformation is defined as follows: Assume that Ω is a bounded domain in $\mathbb{R}^2$ and that velocities u, v and pressure p are sufficiently smooth. Then the system

$$\begin{aligned} & -\Delta u + Re(vu_y - uv_y) + p_x = 0 \quad (\Omega),\\ & -\Delta v + Re(uv_x - vu_x) + p_y = 0 \quad (\Omega),\\ & \Delta p + 2Re(v_x u_y - u_x v_y) = 0 \quad (\Omega),\\ & u_x + v_y = 0 \quad (\partial\Omega),\end{aligned}$$

is equivalent to the original system (9). Two boundary conditions are necessary for the solution of the original equations. The new system requires three, one of them being the continuity equation on the boundary. The generalization to 3D is trivial. This transformation has been proposed in [32], and has recently been evaluated in [38]. Notice that the original primary unknowns are used after the transformation.

A good smoothing algorithm for the resulting discrete problem can be found easily: The principal part of each difference equation is characterized by the Δ_h-operator. The three equations are coupled by lower order terms. So, a standard collective point relaxation is suitable, which updates the three variables u_h, v_h and p_h simultaneously by the solution of a 3×3 systems in every grid point. On a square grid the corresponding collective lexicographical Gauss–Seidel relaxation gives smoothing factors of $\mu = 0.5$ for the Stokes problem. As the discretization takes place on a collocated grid, the other multigrid components related to the coarse grid correction are essentially basic.

3.2. Transformed system of poroelasticity equations

Here, we propose a poroelasticity system transformation $\boldsymbol{C}\boldsymbol{L}_p\bar{\boldsymbol{u}} = \boldsymbol{C}\boldsymbol{f}$, that enables the use of the most efficient (scalar) multigrid methods for iterative solution. We have found in [14] that this approach is more efficient than the staggered or the collocated multigrid treatment of the complete system of equations. However, the system transforma-

tion presented may not be easily performed in the case of heterogeneous media, in which coefficients are not constant. With

$$\boldsymbol{C} = \begin{pmatrix} I & 0 & (\lambda+\mu)\partial_x \\ 0 & I & (\lambda+\mu)\partial_y \\ -\partial_x & -\partial_y & -(\lambda+2\mu)\Delta \end{pmatrix}, \tag{33}$$

we obtain

$$(\mathbf{P^t}) \quad \boldsymbol{C}\boldsymbol{L}_p = \begin{pmatrix} -\mu\Delta & 0 & LC^{1,3} \\ 0 & -\mu\Delta & LC^{2,3} \\ 0 & 0 & \tilde{a}(\lambda+2\mu)\Delta^2 - \Delta \end{pmatrix},$$

with $LC^{1,3}$ and $LC^{2,3}$ continuous versions of the $LC^{3,1}$- and $LC^{3,2}$-terms in (28), (29), and

$$\boldsymbol{C}\boldsymbol{f} = \begin{pmatrix} (\lambda+\mu)\partial_x f \\ (\lambda+\mu)\partial_y f \\ (\lambda+2\mu)\Delta f \end{pmatrix}.$$

Here, a semi-discretization in time has already taken place, as in operator (15). Discrete operator ($\mathbf{P^t_h}$) represents the discretization on a collocated grid.

We end up with an upper triangular discrete system. In a first step then, the last equation should be updated after which the other two equations may be treated. Operator $\tilde{a}(\lambda+2\mu)\Delta_h^2 - \Delta_h$ is split into

$$\big(-\tilde{a}(\lambda+2\mu)\Delta_h + I_h\big)q_h = \tilde{f}_h, \qquad -\Delta_h p_h = q_h,$$

so that we only deal with Laplace type operators.

We describe this algebraic transformation more concretely in vector notation for problem (12), (13). Firstly, applying the divergence operator to (12) and the operator $(\lambda+2\mu)\Delta$ to (13), adding the resulting equations and taking into account the equality $-(\lambda+2\mu)\Delta\,\mathrm{div} = \mathrm{div}(-\mu\tilde{\Delta} - (\lambda+\mu)\,\mathrm{grad}\,\mathrm{div})$, we obtain

$$-\Delta\frac{\partial p}{\partial t} + (\lambda+2\mu)\frac{\kappa}{\eta}\Delta^2 p = -(\lambda+2\mu)\Delta f.$$

Secondly, by applying operator $(\lambda+\mu)\,\mathrm{grad}$ to (13) and by adding the resulting equation to (12) we get

$$-\mu\tilde{\Delta}\frac{\partial \mathbf{u}}{\partial t} + \mathrm{grad}\,\frac{\partial p}{\partial t} - (\lambda+\mu)\frac{\kappa}{\eta}\,\mathrm{grad}\,\Delta p = (\lambda+\mu)\,\mathrm{grad}\,f.$$

With the new variables $q = -\Delta p$ and $\mathbf{v} = \frac{\partial \mathbf{u}}{\partial t}$, we obtain the following transformed system:

$$-\mu\tilde{\Delta}\mathbf{v} + \mathrm{grad}\,\frac{\partial p}{\partial t} + (\lambda+\mu)\frac{\kappa}{\eta}\,\mathrm{grad}\,q = (\lambda+\mu)\,\mathrm{grad}\,f, \tag{34}$$

$$q + \Delta p = 0, \tag{35}$$

$$\frac{\partial q}{\partial t} - (\lambda+2\mu)\frac{\kappa}{\eta}\Delta q = -(\lambda+2\mu)\Delta f, \tag{36}$$

$$\mathbf{v} = \mathbf{0}, \quad p = 0, \quad \mathrm{div}\,\mathbf{v} + \frac{\kappa}{\eta}q = f \quad \text{on } \partial\Omega, \tag{37}$$

plus initial conditions.

Note that problem (34)–(37) is coupled over the boundary of the domain. Let us consider a semi-discretization in time with time step τ. Assuming that $\mathbf{v}^m$, p^m and q^m at time point m are known, the following iterative scheme is proposed:

(1) Solve

$$\begin{cases} \left(\frac{q^{m+1}-q^m}{\tau}\right) - (\lambda+2\mu)\frac{\kappa}{\eta}\Delta q^{m+1} = -(\lambda+2\mu)\Delta f^{m+1} & \text{in } \Omega, \\ \mathrm{div}\,\mathbf{v}^m + \frac{\kappa}{\eta}q^{m+1} = f^{m+1} & \text{on } \partial\Omega. \end{cases}$$

(2) Solve

$$\begin{cases} -\Delta p^{m+1} = q^{m+1} & \text{in } \Omega, \\ p^{m+1} = 0 & \text{on } \partial\Omega. \end{cases}$$

(3) Solve

$$\begin{cases} -\mu\tilde{\Delta}\mathbf{v}^{m+1} = \operatorname{grad}((\lambda+\mu) f^{m+1} - \frac{p^{m+1}-p^m}{\tau} - (\lambda+\mu)\frac{\kappa}{\eta} q^{m+1}) & \text{in } \Omega, \\ \mathbf{v}^{m+1} = \mathbf{0} & \text{on } \partial\Omega. \end{cases}$$

This is as in ($\mathbf{P^t}$). Notice that the boundary condition in step (1) is lagging behind one time step. Without additional iteration the scheme will be only of $\mathcal{O}(\tau)$.

Interesting aspects of this decoupled system are that stable numerical solutions are obtained on a standard collocated grid. Secondly, the operators to be inverted in the algorithm above are again only scalar Laplace type operators, for which we can simply choose four times a highly efficient scalar multigrid algorithm that works well for discrete Laplace type operators. Well-known choices for the multigrid components [40] can be used for all choices of λ, μ, and $\tilde{a}$. In accordance with classical multigrid theory, we observe multigrid convergence factors that are less than 0.1 for a variety of poroelastic problems. In the literature this is often called "textbook multigrid efficiency". Very satisfactory solution times have been obtained, that are about 10 times faster than the iterative solution of the coupled system, discretized on a staggered grid.

The relation between the original and transformed systems and their discrete counterparts suggests us a way to obtain stable schemes on collocated grids for the original problem, if we need to deal with the full system. First of all, we transform the continuous problem, then we discretize this transformed problem, and finally we derive a "perturbed discrete variant" of the original problem, based on these transformations. This insight was used to obtain a stable discrete scheme for the poroelasticity equations in the 1D case, see [14]. In fact, there the discrete perturbed problem and the discrete transformed problem were equivalent, and it can be seen that an added perturbation term was the optimal choice with respect to both accuracy and stability. In 2D and 3D, or in the case of varying coefficients, it is difficult to find a perturbation term that is identical to an equivalent discretization of a transformed problem. But anyway, this insight can be useful to find stable schemes on collocated grids. The stabilizing term in Subsection 4.2 originates from these considerations.

4. Collocated grids

For several industrial problems it is desirable to work with the original system of equations, where, preferably, discretization can take place on a non-staggered grid. Non-staggered discretizations are more convenient in connection with general non-uniform, non-orthogonal curvilinear or unstructured grids. However, stabilization measures are required. With a collocated (i.e., non-staggered) placement, Cartesian components for the unknowns can be used, instead of the grid-oriented ones in the case of staggered grids. A detailed discussion on the advantages and disadvantages of staggered and non-staggered grids on curvilinear grids can, for example, be found in [30].

The choice of staggered or non-staggered discretization depends on the application at hand. As long as one is dealing with rather simple geometries and orthogonal grids, the staggered discretization is superior to the non-staggered one. However, real life applications are often defined by complicated geometries.

4.1. Artificial pressure terms

In order to rule out spurious pressure modes, usually the continuity equation is perturbed by terms involving pressure:

$$\operatorname{div}\mathbf{u} + \varepsilon p = 0, \tag{38}$$

$$\operatorname{div}\mathbf{u} - \varepsilon\Delta p = 0, \tag{39}$$

$$\operatorname{div}\mathbf{u} + \varepsilon\partial_t p = 0, \tag{40}$$

$$\operatorname{div}\mathbf{u} - \varepsilon\partial_t\Delta p = 0, \tag{41}$$

where the parameter ε is taken as h^α or τ^α. Approach (38) corresponds to the classical penalty method, stabilization (40) to the compressibility method and it is related to the projection method of Chorin [9] and (41), which also corresponds to a compressibility method [34], and can be interpreted as the popular Van Kan's method [42]. Approach (39) was especially derived with the purpose of stabilizing the pressure oscillations. It is most widespread in engineering practice. The Pressure-Weighted Interpolation method (PWI) of [29], or a Roe-type flux difference splitting [10] falls into this category. Stabilization for collocated discretizations can thus be achieved by adding an artificial elliptic pressure term to the continuity equation of the following form ($\varepsilon = h^2\tilde{\epsilon}$):

$$\boldsymbol{L}_{s,h}(\tilde{\epsilon}) = \begin{pmatrix} -\Delta_h & 0 & \partial_{x,h} \\ 0 & -\Delta_h & \partial_{y,h} \\ \partial_{x,h} & \partial_{y,h} & -\tilde{\epsilon}h^2\Delta_h \end{pmatrix}. \tag{42}$$

The stabilizing terms replace the zero block. For $h \to 0$, the artificial pressure term tends to 0. Since this term is proportional to h^2, second order accuracy is maintained if all other terms in the system are discretized with second order accuracy. For Stokes equations and for low Reynolds numbers, central second order discretizations can be employed in a straightforward way when using the artificial pressure term.

The addition of an artificial pressure term is implicitly also used in other Stokes or Navier–Stokes discretizations. For instance, the SIMPLE method for non-staggered grid discretizations as discussed in [12] can be seen to implicitly solve a discrete continuity equation augmented by an artificial pressure term of the form (42) with $\tilde{\epsilon} = 1/8$. In the flux-difference splitting of [10], a stabilizing pressure term is introduced in the discrete continuity equation in a natural way. This scheme leads to a so-called vector-positive discretization. It makes Gauss–Seidel type relaxation possible in a collective or a decoupled formulation. Also in the framework of (stabilized) finite element approximations, artificial pressure terms are quite common.

We obtain

$$\det\bigl(\boldsymbol{L}_h(\tilde{\epsilon})\bigr) = -\Delta_h(\tilde{\epsilon}h^2\Delta_h\Delta_h - \Delta_{2h}), \tag{43}$$

with Δ_{2h} denoting the Laplacian discretized on a grid with mesh size $2h$. Parameter $\tilde{\epsilon}$ has to be chosen sufficiently small to maintain a good accuracy but also sufficiently large so that the discrete system presents stable solutions; $\tilde{\epsilon} = 1/16$ [23], for example, turns out to be a reasonable choice in practice, with respect to both accuracy and stability. The stability is satisfactory for $\tilde{\epsilon} \geqslant 1/16$.

4.2. Collocated poroelasticity discretization

We deal with the incompressible poroelasticity equations, in which *a stabilization term* $\varepsilon\partial\Delta p/\partial t$ [15], with $\varepsilon = h^2/4(\lambda + 2\mu)$ is added:

$$-\mu\tilde{\Delta}\mathbf{u} - (\lambda+\mu)\operatorname{grad}\operatorname{div}\mathbf{u} + \operatorname{grad} p = 0, \tag{44}$$

$$\frac{\partial}{\partial t}(\operatorname{div}\mathbf{u}) - \frac{\kappa}{\eta}\Delta p - \varepsilon\frac{\partial}{\partial t}\Delta p = f(\mathbf{x}, t). \tag{45}$$

It needs to be supplied by an initial pressure $p(\mathbf{x}, 0) = 0$.

This problem, i.e., the original system of equations in which a stabilization term is added, can be discretized on a collocated grid with central differences.

We approximate the elasticity part, $-\mu\tilde{\Delta} - (\lambda+\mu)\operatorname{grad}\operatorname{div}$, by the difference operator

$$A_h = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}, \quad A_{11} = -\mu\Delta_h - (\lambda+\mu)\partial_{h,xx},$$
$$A_{12} = A_{21} = -\frac{\lambda+\mu}{h^2}\begin{bmatrix} 1 & & -1 \\ & * & \\ -1 & & 1 \end{bmatrix}_h, \quad A_{22} = -\mu\Delta_h y - (\lambda+\mu)\partial_{h,yy}, \tag{46}$$

where the usual five-point stencil approximation of the Laplace operator is used. To approximate the coupling terms, grad p and div $\mathbf{u}$, we also use second order central difference approximations.

After the spatial approximation of (44), (45) we arrive at a Cauchy problem for the system of differential-difference equations

$$A_h \mathbf{u}_h + G_h p_h = 0, \tag{47}$$

$$\frac{d}{dt}\left(D_h \mathbf{u}_h - \frac{h^2}{4(\lambda+2\mu)} \Delta_h p_h \right) - \frac{\kappa}{\eta} \Delta_h p_h = f_h(\mathbf{x}, t), \tag{48}$$

with initial conditions $D_h \mathbf{u}_h(0) = 0$, $-\Delta_h p_h(0) = 0$. We construct a simple difference scheme in time for the approximation of the solution of the Cauchy problem. We use a uniform grid for time discretization with step-size $\tau > 0$, with $\bar{\boldsymbol{u}}_h^m(\mathbf{x}) = \bar{\boldsymbol{u}}_h(\mathbf{x}, t_m)$, the implicit Euler scheme reads

$$A_h \mathbf{u}_h^{m+1} + G_h p_h^{m+1} = 0, \tag{49}$$

$$\frac{D_h \mathbf{u}_h^{m+1} - D_h \mathbf{u}_h^m}{\tau} - \frac{h^2}{4(\lambda+2\mu)} \frac{\Delta_h p_h^{m+1} - \Delta_h p_h^m}{\tau} - \frac{\kappa}{\eta} \Delta_h p_h^{m+1} = f_h^{m+1}, \tag{50}$$

where the initial condition $D_h \mathbf{u}_h^0$ has already been incorporated. Other discretizations can be obtained if, for example, two-level weighted schemes are used for time stepping. Stability and second order accuracy of this discretization are shown in [15]. There it has also been shown that a stabilization $\varepsilon \partial \Delta p / \partial t$ with $\varepsilon = h^2/4$, so without the term $\lambda + 2\mu$, does not bring discrete solutions with satisfactorily accuracy.

4.3. Relaxation on collocated grids

When discretizing the Stokes or the poroelasticity equations with standard second order discretizations and the artificial pressure term as discussed in Subsections 4.1, 4.2 the development of efficient relaxation schemes is not straightforward. Fourier smoothing analysis shows that the smoothing factors of standard *collective* point relaxations are not satisfactory. One possibility to overcome this problem is to extend the idea of box relaxation, which has proved to be a suitable smoother in the case of staggered discretizations, to the non-staggered case [23].

In Cartesian coordinates, box relaxation is defined essentially in the same way as in the staggered case, except that the side lengths of the boxes are now twice as large (see left side of Fig. 5). For each box, again a 5×5 system is solved, using the respective 4 momentum equations along the edges and the continuity equation in the center of the box.

As in the staggered case, box relaxation should be replaced by a corresponding box-line version, if significant anisotropies occur. For example, in the case of box x-line relaxation, this means that all unknowns marked in Fig. 5(right picture) are updated collectively. For each line then a block tridiagonal matrix has to be inverted.

The smoothing factor, μ, of lexicographic box relaxation for the Stokes operator on a non-staggered grid with $\tilde{\epsilon} = 1/16$ is $\mu = 0.63$ [23]. For box-line relaxation one obtains $\mu = 0.56$. Using underrelaxation for the pressure, these smoothing factors can be improved to values below 0.5 [23].

In [15] an efficient multigrid method is constructed for consolidation type poroelasticity problems on Cartesian grids. This is done on the basis of box relaxation; its robust variant being the alternating box-line relaxation. It has been shown that box-line relaxation is robust for a wide range of values of the Lamé coefficients, if some form of overrelaxation is employed. Even for realistic poroelasticity problems with extremely small time steps and Lamé coefficients defined so that the Poisson ratio is close to 0.5 (the incompressible limit), box line relaxation provides satisfactory multigrid convergence. So, box relaxation is also of importance for multigrid efficiency on collocated grids. The resulting efficiency is of the same order as observed with staggered grids and coupled relaxation.

Although the distributive relaxation schemes have originally been developed for discretizations on staggered grids, as described in Subsections 2.2.2, 2.2.3, these schemes can also be used on non-staggered grids. However, here it

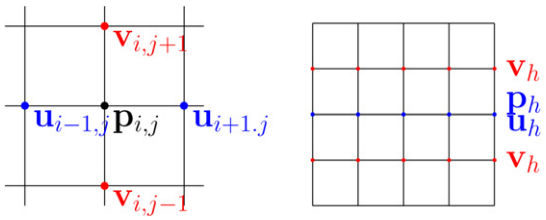

Fig. 5. Left: Five unknowns centered around a pressure point updated simultaneously; Right: Unknowns updated simultaneously by box x-line relaxation.

should be noted that the discrete operators on the collocated grids do not satisfy the important relation $D_h G_h = \Delta_h$. This may have a negative impact on the smoothing properties of distributive relaxation in the case of collocated grids.

5. Conclusions

In this paper we have given an overview of multigrid methods for saddle point type problems originating from the Stokes equations and the incompressible poroelasticity equations. We have emphasized that both systems of equations exhibit similar features with respect to their multigrid treatment. The choice of multigrid method depends strongly on the type of discretization and the problem formulation employed: Departing from the classical treatment of these problems on staggered grids, where the development of efficient relaxation methods appears to be the main issue, we reached stable discretizations on collocated grids via the systems transformation approach. Coupled and decoupled distributive relaxation methods are described in detail. If one is able to apply a systems' transformation that leads to an "almost" decoupled transformed system of equations, the most efficient scalar multigrid methods can be used at the highest efficiency. The development of an efficient distributive relaxation scheme comes second best. Coupled relaxation is a very satisfactory alternative, that can be made "blackbox" for certain problem classes. For poroelasticity on *staggered grids*, coupled relaxation was not fully robust, especially not regarding the choice of the time step, whereas distributive relaxation on the staggered grid was robust for all values of the problem parameters. On the *collocated grid*, the box line relaxation performed best.

References

[1] M. Benzi, G.H. Golub, J. Liesen, Numerical solution of saddle point problems, Acta Numerica (2005) 1–137.
[2] M.A. Biot, General theory of three dimensional consolidation, J. Appl. Phys. 12 (1941) 155–164.
[3] D. Braess, R. Sarazin, An efficient smoother for the Stokes problem, Appl. Numer. Math. 23 (1997) 3–19.
[4] A. Brandt, Multigrid techniques: 1984 guide with applications to fluid dynamics, GMD-Studie Nr. 85, Sankt Augustin, Germany, 1984.
[5] A. Brandt, Multi-level adaptive solutions to boundary-value problems, Math. Comp. 31 (1977) 333–390.
[6] A. Brandt, N. Dinar, Multigrid solutions to elliptic flow problems, in: S. Parter (Ed.), Numerical Methods for Partial Differential Equations, Academic Press, New York, 1979, pp. 53–147.
[7] A. Brandt, I. Yavneh, Accelerated multigrid convergence and high-Reynolds recirculating flows, SIAM J. Sci. Comput. 14 (1993) 607–626.
[8] W. Briggs, V. Emden Henson, St.F. McCormick, A Multigrid Tutorial, second ed., SIAM, Philadelphia, 2000.
[9] A.J. Chorin, A numerical method for solving incompressible viscous flow problems, J. Comput. Phys. 2 (1967) 12–26.
[10] E. Dick, J. Linden, A multigrid method for steady incompressible Navier–Stokes equations based on flux difference splitting, Int. J. Numer. Methods Fluids 14 (1992) 1311–1323.
[11] H. Elman, D. Silvester, A. Wathen, Finite Elements and Fast Iterative Solvers with Applications in Incompressible Fluid Dynamics, Oxford Science Publ., Oxford, 2005.
[12] J.H. Ferziger, M. Perić, Computational Methods for Fluid Dynamics, Springer, Berlin, Germany, 2002.
[13] F.J. Gaspar, F.J. Lisbona, C.W. Oosterlee, R. Wienands, A systematic comparison of coupled and distributive smoothing in multigrid for the poroelasticity system, Numer. Linear Algebra Appl. 11 (2004) 93–113.
[14] F.J. Gaspar, F.J. Lisbona, C.W. Oosterlee, P.N. Vabishchevich, An efficient multigrid solver for a reformulated version of the poroelasticity system, Comput. Methods Appl. Mech. Engrg. 196 (2007) 1447–1457.
[15] F.J. Gaspar, F.J. Lisbona, C.W. Oosterlee, A stabilized difference scheme for deformable porous media and its numerical resolution by multigrid methods, Computing and Visualization in Science (2008), in press.
[16] F.J. Gaspar, F.J. Lisbona, P.N. Vabishchevich, Staggered grid discretizations for the quasi-static Biot's consolidation problem, Appl. Numer. Math. 56 (6) (2006) 888–898.
[17] U. Ghia, K.N. Ghia, C.T. Shin, High-*Re* solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.
[18] Y. Gil, Z. Gutterman, S. Onn, I. Yavneh, Automated transformations for PDE systems with application to multigrid solvers, SIAM J. Sci. Comput. 24 (2002) 886–904.
[19] W. Hackbusch, Multi-Grid Methods and Applications, Springer, Berlin, 1985.
[20] F.H. Harlow, J.E. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with a free surface, Phys. Fluids 8 (1965) 2182–2189.
[21] J.M. Hyman, M. Shashkov, Adjoint operators for the natural discretizations of the divergence, Gradient and curl on logically rectangular grids, Appl. Numer. Math. 25 (1997) 413–442.
[22] J.M. Hyman, M. Shashkov, S. Steinberg, The effect of inner products for discrete vector fields on the accuracy of mimetic finite difference methods, Comput. Math. Appl. 42 (2001) 1527–1547.
[23] J. Linden, B. Steckel, K. Stüben, Parallel multigrid solution of the Navier–Stokes equations on general 2D-domains, Parallel Comp. 7 (1988) 461–475.
[24] A. Niestegge, K. Witsch, Analysis of a multigrid Stokes solver, Appl. Math Comp. 35 (1990) 291–303.

[25] C.W. Oosterlee, P. Wesseling, A robust multigrid method for a discretization of the incompressible Navier–Stokes equations in general coordinates, Impact Comp. in Science Engrg. 5 (1993) 128–151.
[26] J.G. Osorio, H.-Y. Chen, L.W. Teufel, Numerical simulation of the impact of flow-induced geomechanical response on the productivity of stress-sensitive reservoirs, in: Society of Petroleum Engineers (SPE) 51929, 1999, pp. 1–15.
[27] M.F. Paisley, N.M. Bhatti, Comparison of multigrid methods for neutral and stably stratified flows over two-dimensional obstacles, J. Comput. Phys. 142 (1998) 581–610.
[28] S.V. Patankar, D.B. Spalding, A calculation procedure for heat and mass transfer in three-dimensional parabolic flows, Int. J. Heat Mass Transfer 15 (1972) 1787–1806.
[29] C.M. Rhie, W.L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA J. 21 (1983) 1525–1532.
[30] W. Rodi, S. Majumdar, B. Schönung, Finite volume methods for two-dimensional incompressible flow problems with complex boundaries, Comput. Methods Appl. Mech. Engrg. 75 (1989) 369–392.
[31] J.W. Ruge, K. Stüben, Algebraic Multigrid (AMG), in: S.F. McCormick (Ed.), Multigrid Methods, in: Frontiers in Applied Math., SIAM, Philadelphia, 1987, pp. 73–130.
[32] A. Schüller, A multigrid algorithm for the incompressible Navier–Stokes equations, in: W. Hackbusch, R. Rannacher (Eds.), Num. Treatment of Navier–Stokes Equations, in: Notes on Numerical Fluid Mechanics, vol. 30, Vieweg, Braunschweig, 1990, pp. 124–133.
[33] G.J. Shaw, S. Sivaloganathan, On the smoothing of the SIMPLE pressure correction algorithm, Int. J. Numer. Methods Fluids 8 (1988) 441–462.
[34] J. Shen, On error estimates of projection methods for the Navier–Stokes equations: second-order schemes, Math. Comp. 65 (1996) 1039–1065.
[35] S. Sivaloganathan, The use of local mode analysis in the design and comparison of multigrid methods, Comput. Phys. Comm. 65 (1991) 246–252.
[36] P. Sockol, Multigrid solution of the Navier–Stokes equations on highly stretched grids, Int. J. Numer. Methods Fluids 17 (1993) 543–566.
[37] K. Stüben, U. Trottenberg, Multigrid methods: Fundamental algorithms, model problem analysis and applications, in: W. Hackbusch, U. Trottenberg (Eds.), Multigrid Methods, in: Lecture Notes in Math., vol. 960, Springer, Berlin, 1982, pp. 1–176.
[38] R.C. Swanson, Evaluation of a multigrid scheme for the incompressible Navier–Stokes equations, NASA/TM-2004-213026, May 2004.
[39] M.C. Thompson, J.H. Ferziger, An adaptive multigrid technique for the incompressible Navier–Stokes equations, J. Comput. Phys. 82 (1989) 94–121.
[40] U. Trottenberg, C.W. Oosterlee, A. Schüller, Multigrid, Academic Press, New York, 2001.
[41] S.P. Vanka, Block-implicit multigrid solution of Navier–Stokes equations in primitive variables, J. Comput. Phys. 65 (1986) 138–158.
[42] J. van Kan, A second-order accurate pressure-correction scheme for viscous incompressible flow, SIAM J. Sci. Comput. 7 (1986) 870–891.
[43] R.W.C.P. Verstappen, A.E.P. Veldman, Symmetry-preserving discretization of turbulent flow, J. Comput. Phys. 187 (2003) 343–368.
[44] P. Wesseling, An Introduction to Multigrid Methods, Wiley, Chichester, 1992.
[45] P. Wesseling, Principles of Computational Fluid Dynamics, Springer, Berlin, 2001.
[46] P. Wesseling, C.W. Oosterlee, Geometric multigrid with applications to computational fluid dynamics, J. Comp. Appl. Math. 128 (2001) 311–334.
[47] P. Wesseling, A. Segal, C.G.M. Kassels, Computing flows on general three-dimensional nonsmooth staggered grids, J. Comput. Phys. 149 (1999) 333–362.
[48] R. Wienands, F.J. Gaspar, F.J. Lisbona, C.W. Oosterlee, An efficient multigrid solver based on distributive smoothing for poroelasticity equations, Computing 73 (2004) 99–119.
[49] R. Wienands, W. Joppich, Practical Fourier Analysis for Multigrid Methods, Chapman & Hall, 2005.
[50] G. Wittum, Multi-grid methods for Stokes and Navier–Stokes equations with transforming smoothers: algorithms and numerical results, Numer. Math. 54 (1989) 543–563.
[51] G. Wittum, On the convergence of multi-grid methods with transforming smoothers, Numer. Math. 57 (1990) 15–38.
[52] G. Wittum, On the robustness of ILU smoothing, SIAM J. Sci. Comput. 10 (1989) 699–717.