INTERNATIONAL JOURNAL FOR NUMERICAL AND ANALYTICAL METHODS IN GEOMECHANICS
Int. J. Numer. Anal. Meth. Geomech. 2011; **35**:1466–1482
Published online 6 December 2010 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nag.973

Efficient block preconditioners for the coupled equations of pressure and deformation in highly discontinuous media

Joachim Berdal Haga[1,2,*,†], Harald Osnes[1,2] and Hans Petter Langtangen[1,3]

[1] *Scientific Computing Department, Simula Research Laboratory, PO Box 134, N-1325 Lysaker, Norway*
[2] *Department of Mathematics, University of Oslo, Oslo, Norway*
[3] *Department of Informatics, Univeristy of Oslo, Norway*

SUMMARY

Large-scale simulations of flow in deformable porous media require efficient iterative methods for solving the involved systems of linear algebraic equations. Construction of efficient iterative methods is particularly challenging in problems with large jumps in material properties, which is often the case in geological applications, such as basin evolution at regional scales. The success of iterative methods for this type of problems depends strongly on finding effective preconditioners. This paper investigates how the block-structured matrix system arising from single-phase flow in elastic porous media should be preconditioned, in particular for highly discontinuous permeability and significant jumps in elastic properties. The most promising preconditioner combines algebraic multigrid with a Schur complement-based exact block decomposition. The paper compares numerous block preconditioners with the aim of providing guidelines on how to formulate efficient preconditioners. Copyright © 2010 John Wiley & Sons, Ltd.

Received 12 November 2009; Revised 26 July 2010; Accepted 27 July 2010

KEY WORDS: Biot's consolidation; block preconditioners; heterogeneous media; algebraic multigrid; finite elements

INTRODUCTION

Common problems of important industrial and scientific interest in coupled geomechanics include basin modelling, reservoir management, and groundwater depletion. Analysis of such models on a regional scale requires the ability to solve coupled equations with a large number of unknowns, complex geometries and significant spatial variation in the material parameters. To meet the challenge of efficient solution of these models, scalable solvers that are robust with respect to the geometry and discontinuities of realistic problems must be developed. This is addressed in the present paper.

The problem of interest couples single-phase fluid flow with deformation in elastic porous media. This problem is described by a pair of partial differential equations (PDEs), one governing the fluid pressure and one describing the deformation of the porous matrix. Terzaghi [1] developed the original theory of uniaxial soil consolidation, and introduced the ideas of effective stress and the diffusion of fluid pressure by fluid flow. Biot [2] generalized this work to three dimensions and derived the PDEs governing coupling of fluid flow and deformation in linear elastic porous media. The necessity of a hydromechanically coupled formulation has been validated in field and laboratory studies [3–5]; see Neuzil [6] for an overview. A review of modelling of such systems can be found in [7], while [8] offers a comprehensive modern treatment. In this paper, we apply

[*] Correspondence to: Joachim Berdal Haga, Scientific Computing Department, Simula Research Laboratory, PO Box 134, N-1325 Lysaker, Norway.
[†] E-mail: jobh@simula.no

Biot's equations to a series of test cases and study the efficiency of preconditioned iterative solution methods.

In solvers for algebraic systems of equations, such as those arising from discretizations of PDEs, there is a trade-off between robustness and scalability. Direct solvers are generally the most robust with respect to the numerical properties of the equations, and as a result they have become popular in 'difficult' finite element applications. However, they suffer from suboptimal scaling in time and space. The memory requirements in particular grow substantially faster than the number of unknowns in the problem [9]. Furthermore, communication requirements limit parallel scalability [10]. Iterative methods are in contrast highly scalable, but less robust. Their convergence is problem-dependent and sensitive to the parameters of the problem. Even so, their efficiency makes them the only choice for truly large-scale problems.

The number of iterations in Krylov space methods, such as the Conjugate Gradient (CG [11]) or Stabilized Bi-Conjugate Grandient (BiCGStab [12]) methods, for solving a system $\mathscr{A}x=b$ is typically proportional to $\sqrt{\kappa}$, where κ is the condition number of the coefficient matrix $\mathscr{A}$ [13]. By applying a preconditioner $\mathscr{P}^{-1}$ to the system, i.e. solving $\mathscr{P}^{-1}\mathscr{A}x=\mathscr{P}^{-1}b$, one can reduce the condition number and obtain faster convergence. It is in the nature of the finite element method that the condition number of the coefficient matrix increases when the number of unknowns increases—typically, $\kappa\sim\mathcal{O}(h^{-2})$, where h is the characteristic element length [14]. Using a multigrid method as preconditioner, the condition number can in many cases be made independent of the number of unknowns, a property which is referred to as an optimal method because the amount of work per unknown is then independent of the problem size [15].

Multigrid methods have attracted quite some interest as efficient and widely applicable preconditioners [16, 17]. A difficulty with the standard geometric multigrid method is that it needs a hierarchy of coarse grids. This can be difficult to construct in problems with complicated geometries and many internal layers of materials, which is the typical case in geological applications. Algebraic multigrid (AMG [18]) is then a promising alternative, since it relies only on the algebraic structure of the coefficient matrix. Previous studies [19, 20] indicate that AMG preconditioning can remove the dependence of the number of iterations on the number of unknowns when solving the individual PDEs in Biot's model. How AMG can be used to efficiently precondition the coupled systems of equations studied herein is, however, an open question, which we address in the present paper.

There are basically two main categories of preconditioners for coupled systems. The first category addresses the system of algebraic equations that arises from numbering the displacement and pressure degrees of freedom consecutively in each node. Such numberings may minimize the bandwidth for banded solvers or the fill-in for direct sparse solvers. The other category is aimed at systems where all the displacement degrees of freedom are numbered first, followed by the pressure degrees of freedom. This numbering gives rise to a coefficient matrix with a block structure that more directly corresponds to the original system of PDEs (e.g. the first row of blocks corresponds to the first PDE and so forth). Block preconditioners rely on creating separate preconditioners for the individual decoupled equations, and combining these to precondition the coupled system. While simple blockwise methods such as block diagonal (or block Jacobi) preconditioning work well on some coupled problems [21], saddle-point problems (for example) require the application of Schur complement-based methods, owing to non-invertible diagonal blocks. Schur complement-based block preconditioners have also been found to work well on the discretization of Biot's equations [22, 23], although only homogeneous materials were tested. To our knowledge, the efficacy of block preconditioners for Biot's equations with strongly varying material parameters has not been evaluated.

The main physical parameters that influence the evolution of Biot's equations are the elastic parameters and the permeability of the porous matrix. The permeability in particular may exhibit significant jumps of many orders of magnitude in geological applications [8, 24, 25]. This feature may have a severe impact on the performance of numerical methods for solving Biot's equations. Since there is effectively no flow through the low-permeable regions, the use of tailored techniques such as solving for the pressure on only the high-permeable part of the grid is common. In practice, however, this requires either solving for an additional vector variable for the fluid flux in a mixed finite element formulation, or the manipulation of two separate grid solutions for pressure and displacement. Hence, numerical methods that allow for the efficient solution of arbitrary permeability differences without special considerations are attractive.

We assume that the governing differential equations are discretized by a Galerkin finite element method using mixed elements, and study the effect that a large jump in the permeability and a moderate jump in the elastic parameters (consistent with typical geological media) has on the preconditioned iterative solvers. The permeability is parameterized by a factor $\varepsilon \ll 1$, meaning that we basically consider a domain with two types of geological media: one with flow mobility (which is proportional to the permeability) Λ_0 and one with flow mobility $\varepsilon\Lambda_0$. The typical jump in permeability is then described by a factor $1/\varepsilon \gg 1$. The investigations are further extended to the case where the two media have different elastic parameters.

The impact of the jump ε^{-1} on the accuracy of the finite element discretization is not critical as long as the discontinuities are aligned with the element boundaries [26], which we assume in the following. The critical numerical impact of the discontinuities is then on the performance and convergence of solution methods for the coupled linear system $[p\ u]^{\mathsf{T}} = \mathscr{A}^{-1} b$ arising from the discretization, where p and u denote the pressure and displacement solution vectors, respectively.

The present paper studies the numerical convergence of an AMG-preconditioned conjugate gradient-type method applied to the linear system arising from the coupled equations of pressure and displacement in porous media. Our aim is to extend common knowledge from earlier work by investigating a series of test cases and iterative solvers for the coupled problem, with varying degree of discontinuity in the material parameters. We hope that our findings can guide practitioners in how to choose efficient solution methods for large-scale simulations involving coupled geomechanical problems and highly discontinuous media.

THE MATHEMATICAL MODEL

The equations describing poroelastic flow and deformation can be derived from the principles of conservation of fluid mass and the balance of forces on the porous matrix. The linear poroelastic can be expressed, in the small-strains regime, as

$$S\dot{p} - \nabla\cdot\mathbf{\Lambda}\nabla p + \alpha\nabla\cdot\dot{\mathbf{u}} = q, \tag{1}$$

$$\nabla(\lambda+\mu)\nabla\cdot\mathbf{u} + \nabla\cdot\mu\nabla\mathbf{u} - \alpha\nabla p = \mathbf{r}. \tag{2}$$

Here, we subsume body forces such as gravitational forces into the right-hand side source terms q and $\mathbf{r}$. The primary variables are p for the fluid pressure and $\mathbf{u}$ for the displacement of the porous medium, S and $\mathbf{\Lambda}$ are the fluid storage coefficient and the flow mobility respectively, α is the Biot–Willis fluid/solid coupling coefficient, and λ and μ are the Lamé elastic parameters.

As pointed out in the introduction, the aim of the present paper is to study the numerical properties of Equations (1) and (2), and how to solve these efficiently with an iterative solver. To that end, we ignore effects that are not essential to these properties. The fluid–solid coupling coefficient α is treated as a constant (in practice it varies between about 0.5 and 1). The fluid storage coefficient S is insignificant compared to the fluid mobility in high-permeable regions. In low-permeable regions it acts as an effective fluid compressibility term, and makes the problem less numerically stiff for short time steps. By dropping this term we try to ensure that the validity of the testing is not compromized by choosing a too short ('easy') time step. The other time-derivative term, $\nabla\cdot\dot{\mathbf{u}}$ in Equation (1), couples the displacement to the pressure and *is* included.

We employ a first-order backward finite difference method in time. Our simplified model problem is thus

$$-\Delta t\nabla\cdot\mathbf{\Lambda}\nabla p + \nabla\cdot\mathbf{u} = q\Delta t + \nabla\cdot\mathbf{u}_{k-1}, \tag{3}$$

$$\nabla(\lambda+\mu)\nabla\cdot\mathbf{u} + \nabla\cdot\mu\nabla\mathbf{u} - \nabla p = \mathbf{r}, \tag{4}$$

DOI: 10.1002/nag

where variables without subscripts are taken to be at the current time step k. Moreover, we restrict $\mathbf{\Lambda}$ to be isotropic, parameterized by $\varepsilon \leqslant 1$, so that $\mathbf{\Lambda} = \Lambda_0 \mathbf{I}$ in the high-permeable region and $\mathbf{\Lambda} = \varepsilon \Lambda_0 \mathbf{I}$ in the low-permeable region, with $\mathbf{I}$ being the identity tensor.

Numerical approximation

We proceed to rewrite Equations (3) and (4) in weak form, using integration by parts to eliminate second derivatives. The following relations must then be satisfied for all test functions π and $\mathbf{w}$ in the domain Ω:

$$\int_\Omega [\Delta t \nabla \pi \cdot \mathbf{\Lambda} \nabla p + \pi \nabla \cdot \mathbf{u}] \, \mathrm{d}\Omega = \int_\Omega [\pi \nabla \cdot \mathbf{u}_{k-1} + \pi q \Delta t] \, \mathrm{d}\Omega - \int_\Gamma \pi f_\mathbf{n} \Delta t \, \mathrm{d}\Gamma, \quad (5)$$

$$\int_\Omega [(\nabla \cdot \mathbf{w})(\lambda + \mu)(\nabla \cdot \mathbf{u}) + \nabla \mathbf{w} : \mu \nabla \mathbf{u} - (\nabla \cdot \mathbf{w}) p] \, \mathrm{d}\Omega = -\int_\Omega \mathbf{w} \cdot \mathbf{r} \, \mathrm{d}\Omega + \int_\Gamma \mathbf{w} \cdot \mathbf{t_n} \, \mathrm{d}\Gamma. \quad (6)$$

The fluid flux $f_\mathbf{n}$ and normal stress $\mathbf{t_n}$ at the boundary Γ appear here as natural boundary conditions.

The discrete finite element approximation follows from solving Equations (5) and (6) in finite-dimensional spaces. In this paper, a piecewise (triangular) continuous quadratic space is used for the deformation and a piecewise continuous linear space is used for the pressure,

$$p, \pi \in P^1(\Omega), \quad \mathbf{u}, \mathbf{w} \in [P^2(\Omega)]^d, \quad (7)$$

with dimensionality $d = 2$. The reason for this mix of spaces is that spurious pressure oscillations can occur in low-permeable regions when the same spaces are used for pressure and deformation [27, 28].

The algebraic system

The algebraic system that results from discretizing Equations (5) and (6) is on the form

$$\mathscr{A} x = b, \quad (8)$$

where $\mathscr{A}$ is the coefficient matrix derived from the left-hand sides of Equations (5) and (6), b is the load vector arising from the right-hand sides, and x is the unknown solution vector. Since this is a coupled system of two equations, the coefficient matrix is a 2×2 block matrix

$$\mathscr{A} = \begin{bmatrix} A_{uu} & A_{up} \\ A_{pu} & A_{pp} \end{bmatrix}, \quad (9)$$

where the subscripts denote the primary variable(s) each block acts upon: A_{pp} couples pressure to pressure, A_{pu} couples displacement to pressure, et cetera. The solution and load vectors are given as $x = [u \;\; p]^\mathsf{T}$ and $b = [b_u \;\; b_p]^\mathsf{T}$. The sign of the equations can be chosen so as to make this a symmetric indefinite problem, which we write as

$$\mathscr{A} = \begin{bmatrix} A & B \\ B^\mathsf{T} & C \end{bmatrix}, \quad (10)$$

where A is symmetric positive definite and C is symmetric negative definite.

BLOCK PRECONDITIONING METHODS

Since the convergence rate of iterative solvers depends on the numerical properties—the condition number in particular, but also the eigenvalue distribution—of the coefficient matrix, a preconditioner is in most cases required to achieve a satisfactory convergence rate. In general, the preconditioner $\mathscr{P}^{-1}$ should be fast to compute and close to $\mathscr{A}^{-1}$, although the latter is not a necessary condition.

DOI: 10.1002/nag

In fact, a better (although somewhat circular) requirement is that it gives $\mathscr{P}^{-1}\mathscr{A}$ a beneficial eigenvalue distribution. For the Krylov family of iterative solvers, the exact meaning of 'beneficial' is somewhat complicated, but having a small number of tight eigenvalue clusters often leads to rapid convergence [29].

We assume for the moment the availability of good preconditioners for the symmetric definite decoupled problems. These can be formed by, e.g. multigrid or incomplete factorization methods; we shall discuss these in a later section. The question then is: how can these be combined to an effective preconditioner for the *coupled* Biot's equations? We briefly present here the motivation for the block preconditioners that are chosen for the numerical experiments.

Given a nonsingular 2×2 block matrix

$$\mathscr{A}=\begin{bmatrix} A & B \\ B^{\mathsf{T}} & C \end{bmatrix}, \tag{11}$$

such as that in Equation (9), we focus on block preconditioners of $\mathscr{A}$, i.e, those that can be written on the form

$$\mathscr{P}^{-1}=\begin{bmatrix} M & N \\ P & Q \end{bmatrix}. \tag{12}$$

For example, the standard block Jacobi and block Gauss–Seidel preconditioners can be expressed as

$$\mathscr{P}_{sJ}^{-1}=\begin{bmatrix} \tilde{A}^{-1} & 0 \\ 0 & \tilde{C}^{-1} \end{bmatrix} \quad \text{and} \quad \mathscr{P}_{\mathrm{sGS}}^{-1}=\begin{bmatrix} \tilde{A}^{-1} & 0 \\ 0 & \tilde{C}^{-1} \end{bmatrix}\begin{bmatrix} I & 0 \\ -B^{\mathsf{T}}\tilde{A}^{-1} & I \end{bmatrix}, \tag{13}$$

respectively, where $\tilde{A}^{-1}$ and $\tilde{C}^{-1}$ are approximations to the inverses of the diagonal blocks in Equation (11), i.e. to the inverses of the decoupled equations.

Furthermore, when A is nonsingular, the associated Schur complement of $\mathscr{A}$ is

$$S=B^{\mathsf{T}}A^{-1}B-C. \tag{14}$$

It is then easily verified that the exact inverse of $\mathscr{A}$ can be written as

$$\mathscr{A}^{-1}=\begin{bmatrix} I & -A^{-1}B \\ 0 & I \end{bmatrix}\begin{bmatrix} A^{-1} & 0 \\ 0 & -S^{-1} \end{bmatrix}\begin{bmatrix} I & 0 \\ -B^{\mathsf{T}}A^{-1} & I \end{bmatrix}, \tag{15}$$

with S defined as in Equation (14). Using this block decomposition as the basis of a preconditioner for symmetric indefinite systems was proposed by Toh *et al.* [23]. Equation (15) can also be viewed as a *symmetric* block Gauss–Seidel preconditioner, where C^{-1} is replaced by $-S^{-1}$ as the $(2,2)$ block. This is seen by comparing Equations (13) and (15). We generalize this observation by defining the preconditioning basis of $\mathscr{A}$ as

$$\mathscr{A}_{\mathrm{prec}}=\begin{bmatrix} A & B \\ B^{\mathsf{T}} & D \end{bmatrix}, \tag{16}$$

where the D block may be replaced by, e.g. the original (C), which leads to the standard block preconditioners in Equation (13); or the negative Schur complement ($-S$), which produces the Schur complement preconditioners based on Equation (15). We have tested preconditioners using both of these bases, as well as one using an ε-capped modification of C, in our numerical experiments.

Another Schur complement-based preconditioner was evaluated in a homogeneous context by Phoon *et al.* [22], where the Generalized Jacobi preconditioner was defined (in un-inverted form) as

$$\mathscr{P}_{gJ(\alpha)}=\begin{bmatrix} \tilde{A} & 0 \\ 0 & \alpha\tilde{S} \end{bmatrix}, \tag{17}$$

where $\tilde{A}$ and $\tilde{S}$ are approximations to the exact (1,1) block and the Schur complement, respectively. The Generalized Jacobi preconditioner is equivalent to a block Jacobi preconditioner with $D=\alpha S$. Phoon *et al.* argue that while the choice of α is not significant when the exact $(1,1)$ block $\tilde{A}=A$ is used, a negative value for α performs better when a cruder approximation is used. It was shown that this preconditioner leads to an attractive eigenvalue distribution, with three distinct eigenvalue clusters around 1 and $(1\pm\sqrt{1+4/\alpha})/2$, each with diameter of order $\|S^{-1}C\|$. Although this theoretical result depends on the exact inversion of Equation (17), the practical applicability of a diagonal approximation with $\alpha=-4$ was demonstrated.

An interesting question, when utilizing a symmetric preconditioner such as one based on Equation (15), is whether the preconditioned coefficient matrix is positive definite. If it is, then the Conjugate Gradient method can be used instead of indefinite methods such as BiCGStab. We can define the 'approximate identities' generated by $\tilde{A}^{-1}$ and $\tilde{S}^{-1}$ as

$$\tilde{I}_A = \tilde{A}^{-1}A, \tag{18}$$

$$\tilde{I}_S = \tilde{S}^{-1}(B^{\mathsf{T}}\tilde{A}^{-1}B-C). \tag{19}$$

Both approach the identity matrix I (of the appropriate dimension) as $\tilde{A}^{-1}$ and $\tilde{S}^{-1}$ approach the real inverses, and both are symmetric positive definite as long as the single-block preconditioners are. The preconditioned coefficient matrix, which can be written as

$$\mathscr{P}_{\text{gSGS}}^{-1}\mathscr{A}=\begin{bmatrix}\tilde{I}_A & 0\\ 0 & \tilde{I}_S\end{bmatrix}+\begin{bmatrix}\tilde{A}^{-1}B\tilde{S}^{-1}B^{\mathsf{T}} & \tilde{A}^{-1}B\\ -\tilde{S}^{-1}B^{\mathsf{T}} & 0\end{bmatrix}\begin{bmatrix}I-\tilde{I}_A & 0\\ 0 & I-\tilde{I}_S\end{bmatrix}, \tag{20}$$

then also approaches the identity, and the problem is trivially solved. Of more practical interest is under what circumstances Equations (18) and (19) are close enough to the identity such that Equation (20) is ensured to be positive definite. Since the preconditioned matrix is symmetric,[‡] its eigenvalues are on the real axis. The question is whether they are positive. The eigenvalue distribution of Equation (20) (as well as the non-symmetric Gauss–Seidel variant of the same) was analyzed in [23][§]. In particular, it was found that the eigenvalues are not guaranteed to be positive unless all eigenvalues of $\tilde{A}^{-1}A$ are greater or equal to one, which is typically not the case for efficient single-block preconditioners. Hence, $\mathscr{P}_{\text{gSGS}}^{-1}\mathscr{A}$ is not necessarily positive definite; but since all eigenvalues approach unity in the limit of exact single-block preconditioners, it clearly is if these are sufficiently accurate. The utility of transforming a symmetric indefinite system into a positive definite one was demonstrated in [30], wherein a preconditioner was explicitly designed to transform the system of equations into a positive definite one, solvable by Conjugated Gradients.

Computational cost

The computational cost of the preconditioner can be divided in two parts. First, the construction of the preconditioner involves, in addition to the cost of constructing the single-block preconditioners, the creation of the D block of the preconditioning basis in Equation (16). If this involves a modified version of the model equations, the cost is that of an extra finite element assembly. The Schur complement can be very costly to construct, but a reasonable approximation (as we shall see, the one used in this paper) can be created at roughly the cost of three single-block matrix–vector products. This is cheaper than a single iteration of the BiCGStab iterative method.

Second, each application of the block preconditioner results in a number of single-block operations, which is listed in Table I. This cost is incurred twice for each iteration in BiCGStab, or once per iteration with CG. For comparison, the 2 × 2 block BiCGStab iteration also involves eight matrix–vector products (two for each block), twelve vector additions, and eight inner products.

[‡] Strictly speaking, it is the spectrally equivalent matrix $\mathscr{E}^{-1}\mathscr{A}\mathscr{E}^{\mathsf{T}}$ with $\mathscr{P}=\mathscr{E}^{\mathsf{T}}\mathscr{E}$ that is symmetric.

[§] In the reference, these are called the 'constrained' and 'block triangular' preconditioners.

DOI: 10.1002/nag

Table I. Number of applications of the single-block operations for one application of the block preconditioner.

	$\tilde{A}^{-1}x$	$\tilde{D}^{-1}x$	Bx	$x+ay$
Block Jacobi	1	1	0	0
Block Gauss–Seidel	1	1	1	1
Symmetric Block Gauss–Seidel	2	1	2	2

NUMERICAL INVESTIGATIONS

Block preconditioners

In our numerical investigations we compare the performance of ten block preconditioners in combination with the BiCGStab method and one with the CG method. These are selected from the combinations of five different preconditioning bases with three different blocking schemes.

We define the lower-triangular coupling matrix as

$$\mathscr{G}=\begin{bmatrix} I & 0 \\ -B^{\mathsf{T}}\tilde{A}^{-1} & I \end{bmatrix}. \tag{21}$$

The blocking schemes are then, with reference to the definition of $\mathscr{A}_{\text{prec}}$ in Equation (16), the block Jacobi preconditioning scheme,

$$\mathscr{P}_1^{-1}=\begin{bmatrix} \tilde{A}^{-1} & 0 \\ 0 & \tilde{D}^{-1} \end{bmatrix}, \tag{22}$$

where $\tilde{A}^{-1}$ and $\tilde{D}^{-1}$ are (in some sense) close to the real inverses; the block Gauss–Seidel preconditioning scheme

$$\mathscr{P}_2^{-1}=\mathscr{P}_1^{-1}\mathscr{G}; \tag{23}$$

and the symmetric block Gauss–Seidel variant

$$\mathscr{P}_3^{-1}=\mathscr{G}^{\mathsf{T}}\mathscr{P}_1^{-1}\mathscr{G}. \tag{24}$$

Note that when $D=-S$, Equation (15) is approximated by $\mathscr{P}_3^{-1}$.

The (2,2) block in the preconditioning bases are $D=C$ (the ‘standard’ basis), $D=\alpha\tilde{S}$ (approximate Schur complement, or ‘generalized’, basis), and $D=C_{\varepsilon\geqslant 10^{-4}}$ (capped-ε basis). In the latter, the coefficient matrix of a more regular problem, with ε capped to nowhere be smaller than 10^{-4}, is used in the basis. This particular value of ε was chosen after some experimentation.

The selected combinations are then: The standard basis combined with all three blocking schemes; the Schur complement (generalized) basis with $\alpha=-1$, combined with all three blocking schemes; the Schur complement (generalized) basis with $\alpha=1$ and $\alpha=\pm 4$, combined with block Jacobi; and the capped-ε basis combined with block Jacobi (symmetric block Gauss–Seidel was also tested, but it was not observed to bring any advantages over the Jacobi variant). Finally, the $\alpha=-1$ generalized basis with the symmetric Gauss–Seidel scheme is tested in combination with the Conjugate Gradient method. These combinations are summarized in Table II along with their abbreviations.

The single-block preconditioners

The block preconditioners in the previous section depend on the availability of efficient single-block preconditioners $\tilde{A}^{-1}$ and $\tilde{D}^{-1}$. We restrict our attention to preconditioners that have the property of being efficient on massively parallel computers. This rules out incomplete and approximate direct solvers such as the otherwise excellent ILU methods.

Table II. Abbreviations used for the tested preconditioners. These have a three-part structure: The block basis (standard, generalized or capped) in lower case, followed by the preconditioning scheme (Jacobi, Gauss–Seidel or Symmetric Gauss–Seidel), and optionally followed by the variant (the value of α in the generalized Jacobi preconditioners, or the 'cg's postfix where the Conjugate Gradient method is used). With the exception of gSGS/cg, all preconditioners are used with the BiCGStab iterative solver.

	Standard	Capped	Generalized	
	$D=C$	$D=C_{\varepsilon\geqslant c}$	$D=-\tilde{S}$	$D=\alpha\tilde{S}$
Block Jacobi	sJ	cJ	gJ(−1)	gJ(1), gJ(±4)
Block Gauss–Seidel	sGS		gGS	
Symmetric Block Gauss–Seidel	sSGS		gSGS	
(... with Conjugated Gradients)			gSGS/cg	

Adams [19] found algebraic multigrid (AMG) to behave very well on problems of elastic deformation, even in the presence of strong material discontinuities. In particular, the smoothed aggregation method [31, 32] was considered to be the overall superior AMG method for elasticity problems. The present authors likewise found AMG to be a nearly optimal preconditioner for the discontinuous Poisson pressure problem, as long as the low-permeable regions do not completely isolate any high-permeable regions [20]. In the limit of $\varepsilon\rightarrow 0$, such isolation would in fact create a physically indeterminate problem. When coupled with deformation of the solid matrix, however, the problem becomes well-posed both physically and—as we shall see—numerically.

In the light of these earlier results, and the fact that AMG has been shown to scale very well in parallel, to at least thousands of processors [17, 33–35], we have chosen to use AMG for both the pressure and the displacement equation. As for the other preconditioning bases, both αS and $C_{\varepsilon\geqslant 10^{-4}}$ are modifications of the single block in the preconditioning basis associated with the pressure equation, and AMG is used also to approximate the inverses of these.

Approximating the Schur complement

The Schur complement in Equation (14) is a dense matrix, and as such it is neither feasible nor desirable to compute. While a number of sparse approximations to S are possible, one approximation that is very fast to compute[¶] is

$$\tilde{S}_1=\mathrm{diag}(B^{\mathsf{T}}(\mathrm{diag}A)^{-1}B)-C. \tag{25}$$

This is the approximation used in the numerical experiments in this paper. The entries of $\tilde{S}$ are simply $\tilde{S}_{ij}=\delta_{ij}\sum_k (A_{kk})^{-1}(B_{ki})^2-C_{ij}$. When the matrices are stored in the CRS (compressed row storage) representation, this makes the calculation extremely cheap: a sequential traversal of three matrices plus arbitrary accesses into the diagonal of A.

More accurate approximations to the Schur complement can be calculated. Toh *et al.* [23] evaluated a number of approximations in the context of iterative solution of Biot's equations, and found the simple approximations to be effective. This matches our experience: In addition to the approximation in Equation (25), we also looked at a slightly more accurate variation,

$$\tilde{S}_2=B^{\mathsf{T}}(\mathrm{diag}A)^{-1}B-C, \tag{26}$$

but no improvement was observed (the performance in initial testing was in fact slightly worse). Other variants, such as using a sparse approximate inverse of A in the triple matrix product, are also possible.

The action of $\tilde{S}^{-1}$ on a vector v can, however, also be approximated by an inner iterative solution of $\tilde{S}x=v$, in which case $\tilde{S}$ need not be formed explicitly. For example, the Conjugate Gradient method can be employed with $\tilde{S}_3=B^{\mathsf{T}}\tilde{A}^{-1}B-C$. We notice from Equation (19) that it is in fact

[¶] In particular, this matrix can be calculated with minimal or no interprocess communication on a parallel computer.

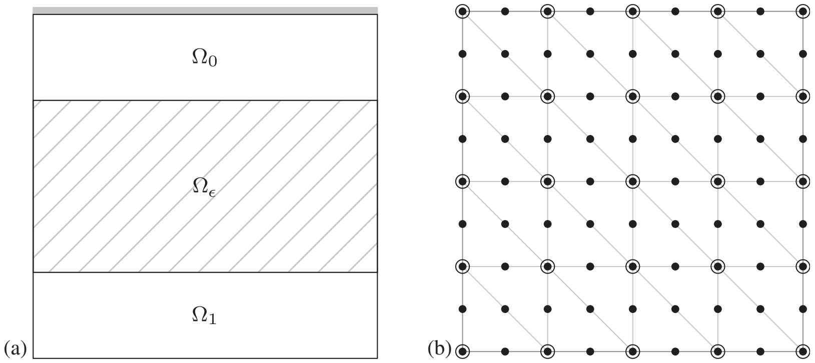

Figure 1. The domain (a) and the mesh (b). Ω_ε marks the low-permeable region. Pressure nodes are shown as circles (○), displacement nodes as dots (●). The mesh is the smallest regular P^2–P^1 mesh that aligns the element boundaries with the discontinuities ($N=9$).

better if $\tilde{S}$ approximates $B^{\mathsf{T}}\tilde{A}^{-1}B-C$ rather than the exact Schur complement. We have not seen the need to include this procedure in our test, so it is mentioned here only for completeness.

Implementation

We have implemented the finite element discretization, block preconditioners and linear solvers using the Diffpack C++ framework [36], somewhat modified for our needs. The single-block AMG preconditioners are from the ML package for smoothed aggregation [37], which is part of Trilinos [38].

Test geometry

Figure 1 shows the two-dimensional domain of the test problems. For the pressure variable, we use essential boundary conditions at the top of the domain (specified pressure) and natural boundary conditions at the bottom and sides (no-flow condition). The displacement boundary conditions are essential at the bottom (fixed position) and natural at the top (specified traction force). At the sides the horizontal displacement components are zero.

It should be noted that when $\varepsilon \to 0$, the decoupled pressure equation is ill-posed because Ω_1 in Figure 1(a) becomes an isolated subdomain with indeterminate pressure because of the pure Neumann conditions. When coupled to deformation, however, the problem is well-posed.

Convergence criterion

We have in our earlier work observed that a convergence criterion based on the residual in iteration k, $r_k = b - \mathscr{A}x_k$, may be misleading when $\mathscr{A}$ is severely ill-conditioned, owing to some components of x being $\kappa(\mathscr{A})$ times more influential than others [20]. This problem is exacerbated when pushing against the limits of machine precision, as may happen when parameters vary by more than ten orders of magnitude. Hence, in the convergence tests in the present paper we exploit an established property of iterative solvers: their rate of convergence is independent of the right-hand side b as long as the initial guess contains all eigenvectors of $\mathscr{A}$ [13, chapter 3.4].

For this reason we have chosen to solve the modified problem $\mathscr{A}x=0$, instead of $\mathscr{A}x=b$, together with a random initial solution vector x_0. With this choice of right-hand side, the error norm $\|e_k\|_{\ell_2}$ is trivially available, since $e_k = x_k$. The convergence criterion is $\|e_k\|_{\ell_2} < 10^{-6}\|e_0\|_{\ell_2}$. We also note that due to this testing procedure, the exact value of any boundary condition is irrelevant, since these values go into the b vector. The only relevant information is whether or not they are essential, since the presence of an essential boundary condition at a node is reflected by a modification to $\mathscr{A}$.

All the reported iteration counts are from at least five runs using different random initial guesses. In the graphs, the mean and range of the results are shown.

On the order of iterative methods

We often refer to the order of an iterative solution method, or the order of a preconditioner (in combination with an iterative method). As mentioned in the introduction, the number of iterations to solve a linear system to a given accuracy with conjugate gradient-type methods is proportional to $\sqrt{\kappa}$, where $\kappa(\mathscr{P}^{-1}\mathscr{A})$ is the condition number of the preconditioned coefficient matrix. For discretizations of the finite element methods, $\kappa(\mathscr{A})\sim\mathcal{O}(h^{-2})$, where h is the length scale of the elements. The number of iterations of an iterative method for this unpreconditioned coefficient matrix is then of *order* $\mathcal{O}(h^{-1})\sim\mathcal{O}(N)$, since $N\sim h^{-1}$ in the present paper denotes the number of nodes in each space direction.

In general, we assume that the number of iterations to reduce the error by a fixed factor can be modelled as

$$n\sim aN^{p}, \tag{27}$$

where the multiplicative factor a and the exponent p of the order may depend on the geometry and mesh, the heterogeneity of the material parameters, boundary conditions, and so on; but not on N. By *optimal order* (*with respect to* N) we mean that $p=0$, and hence that the number of iterations is independent of N. A method which is *optimal with respect to* ε may have $p>0$, but the number of iterations is independent of ε. Finally, a weaker (but still attractive) property is having a *growth rate that is independent of* ε; that is, p does not depend on ε even if a does.

Performance of the fully coupled solver with uniform elastic parameters

In the first group of experiments with the fully coupled solver, the elastic parameters are held constant throughout the domain, while the permeability has a discontinuous jump of up to 16 orders of magnitude ($\varepsilon=10^{0},\dots,10^{-16}$). The time step and fluid mobility are scaled such that $\Lambda_0\Delta t=1$, and the elastic parameters are $\lambda=114$ and $\mu=455$ (corresponding to Young's modulus $E=10^{3}$ and Poisson's ratio $\nu=0.1$).

Performance with constant permeability. The constant-parameter Biot's equations, with $\varepsilon=10^{0}$ and uniform elasticity, seem simple to solve. If AMG can solve or precondition the separate equations nearly optimally—which seems to be the case, at least in idealized cases [20, 33]—then one might expect the same to be the case for the fully coupled problem with the application of an equally simple block preconditioner. Yet, as seen in Figure 2, this is not necessarily the case. The (nearly) optimal order, where the number of iterations is independent of problem size, is seen only when the domain is discretized with equal polynomial order quadrilateral ($Q^{1}-Q^{1}$) elements. These elements are, however, less attractive for other reasons; equal-order elements are susceptible to pressure oscillations in permeability interfaces, and quadrilaterals are less flexible with respect to unstructured geometries than triangular elements. When triangular or mixed elements are used, the order is slightly below $\sqrt{N}$. This is still a major improvement over the expected order N of the unpreconditioned or diagonally scaled finite element method. For two-dimensional problems, it means that the number of unknowns can be increased at least 16 times for a doubling in the number of iterations, whereas using diagonal scaling it can only be increased fourfold.

The figure shows convergence data for the block Jacobi (sJ) preconditioner, but as seen in Table III(b) similar rates are seen with the other preconditioners for the $P^{2}-P^{1}$ space.

Performance with moderate jumps in permeability. As long as the jumps in permeability are of moderate size, $\varepsilon\geqslant10^{-4}$, the problem is numerically well behaved. Figure 3(a) shows the convergence behavior of different block preconditioners under these conditions. In fact, some of the preconditioned solvers initially have *decreasing* order as ε gets smaller (most easily seen by comparing columns one through three in Table III(b)). This is, however, a small effect, and not significant compared to the increase in the number of iterations observed in Figure 3(a).

Performance of the fully coupled solver with severe jumps in permeability. When the discontinuities become more severe, with $\varepsilon<10^{-4}$, several of the preconditioners fail to converge, as shown in

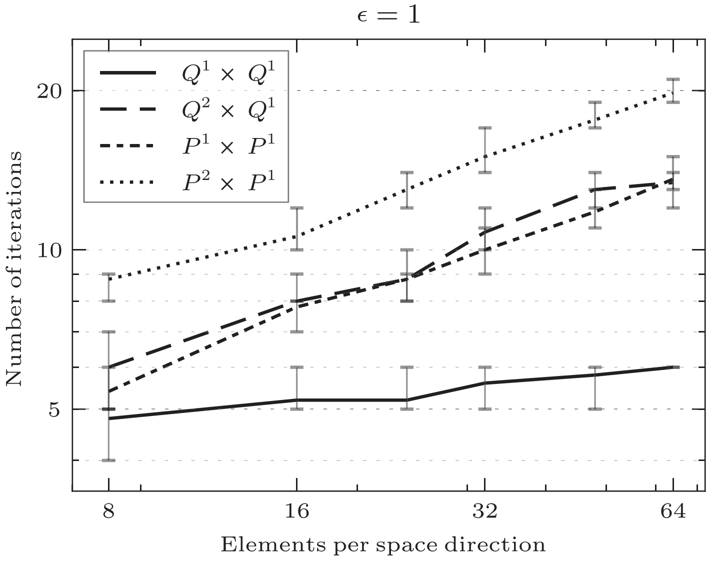

Figure 2. Iteration count for the homogeneous-domain problem. The sJ preconditioner was used. Q denotes quadrilaterals and P denotes simplices of a given polynomial order.

Table III. Performance of the iterative solvers with the various preconditioners listed in Table II, with uniform and discontinuous elastic parameters.

	Uniform elastic parameters					Discontinuous elastic parameters				
$\varepsilon \rightarrow$	10^0	10^{-4}	10^{-8}	10^{-12}	10^{-16}	10^0	10^{-4}	10^{-8}	10^{-12}	10^{-16}
(*a*) *Average number of iterations at* $N = 65$										
sJ	18	35	—	—	—	36	57	178	—	—
sGS	19	34	486	—	—	37	56	80	—	—
sSGS	19	32	621	—	—	37	59	95	—	—
cJ	*Same as sJ*		323	313	326	*Same as sJ*		579	—	—
gGS	19	36	72	71	75	38	57	79	116	109
gSGS	19	35	67	66	66	37	56	70	104	108
gSGS/cg	26	45	89	91	91	46	65	96	146	153
gJ(−1)	18	35	—	—	—	36	63	177	—	—
gJ(−4)	19	38	1279	1545	—	36	61	152	—	—
gJ(1)	60	103	195	213	207	119	173	177	348	360
gJ(4)	76	122	195	204	199	134	185	210	327	409
(*b*) *Order of convergence* (*p in Equation* (27)) *calculated from* $N = 33 \dots 65$										
sJ	0.41	0.31	—	—	—	0.35	0.45	-0.10	—	—
sGS	0.33	0.48	0.18	—	—	0.41	0.34	0.05	—	—
sSGS	0.43	0.29	0.33	—	—	0.37	0.44	0.09	—	—
cJ	*Same as sJ*		1.00	0.86	1.00	*Same as sJ*		-0.14	—	—
gGS	0.47	0.47	0.57	0.49	0.60	0.40	0.41	0.47	0.58	0.54
gSGS	0.37	0.46	0.51	0.48	0.43	0.35	0.38	0.26	0.46	0.52
gSGS/cg	0.40	0.49	0.42	0.45	0.44	0.37	0.36	0.27	0.42	0.50
gJ(−1)	0.38	0.30	—	—	—	0.33	0.59	0.08	—	—
gJ(−4)	0.42	0.36	2.98	2.20	—	0.38	0.48	0.81	—	—
gJ(1)	0.52	0.54	0.46	0.50	0.45	0.53	0.49	0.26	0.47	0.54
gJ(4)	0.64	0.66	0.51	0.53	0.53	0.70	0.57	0.48	0.50	0.91

Failure to converge is indicated by '—'.

Figure 3(b). The first to diverge are the standard and generalized Jacobi preconditioners sJ and gJ(−1), which drop out at $\varepsilon = 10^{-8}$ (hence these are not plotted in this figure). The Gauss–Seidel preconditioners are better, but when ε goes below 10^{-8}, the standard-basis variants sGS and sSGS also fail. At $\varepsilon = 10^{-16}$, the gJ(−4) preconditioner does not converge on the finest grid ($N = 65$).

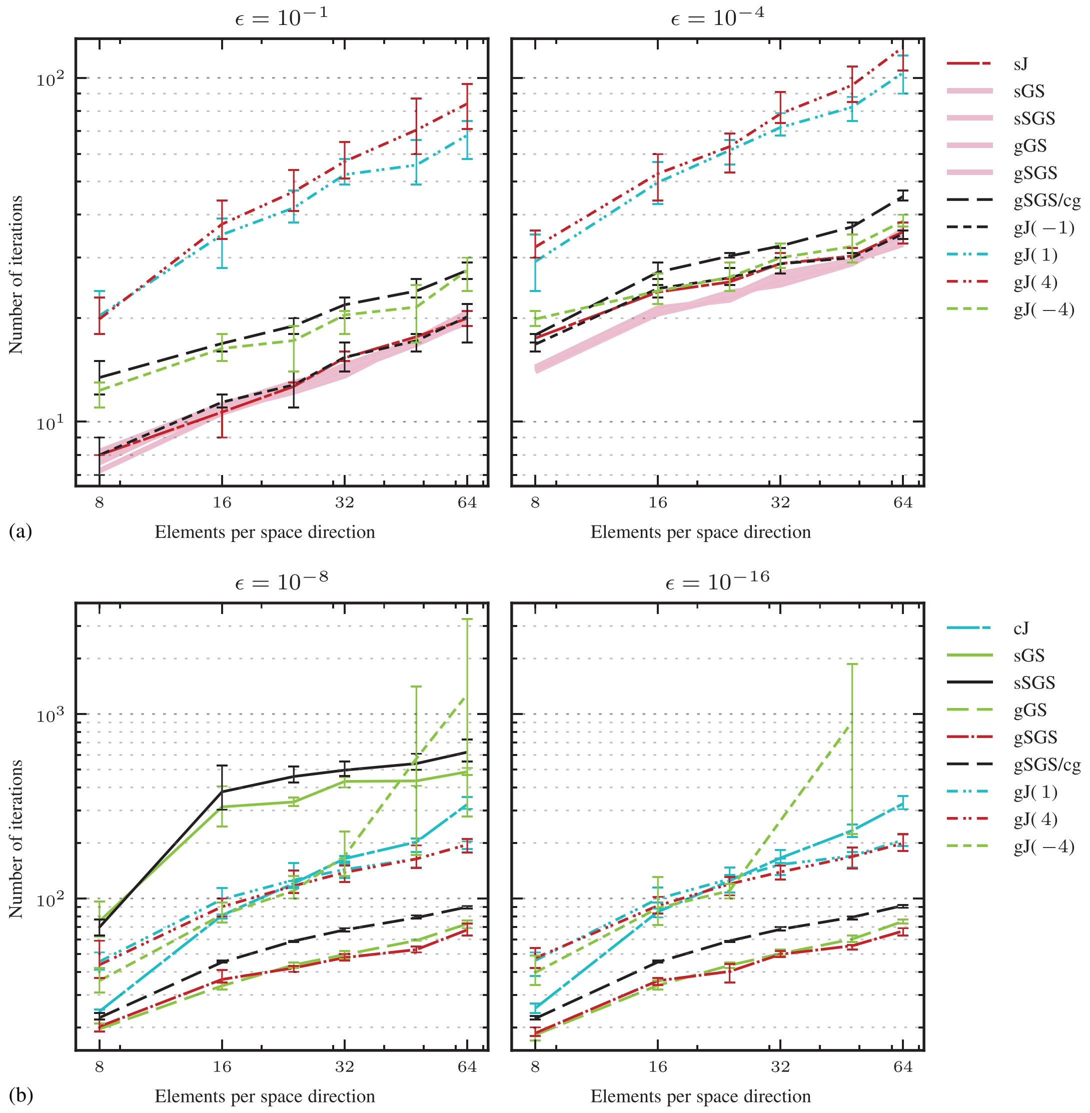

Figure 3. Number of iterations to reach convergence ($|e_k|<10^{-6}|e_0|$) for the model problem with uniform elastic parameters. (a) Low-to-moderate permeability contrast. The Gauss–Seidel methods overlap, and are drawn in gray for legibility and (b) Severe permeability contrast. In (a), all preconditioners except for gJ($\alpha>0$) show a growth rate of roughly $N^{0.3}$–$N^{0.4}$, with N being the number of displacement nodes in each space direction. At $\varepsilon=10^{-8}$ (lower left), the sGS and sSGS preconditioners show a surprisingly low growth rate as N increases, but with a large constant factor. When the discontinuities are even stronger (lower right), these variants fail to converge at all. The Schur variants (gGS, gSGS and gJ(1)/gJ(4)) show a growth rate of about $N^{0.5}$ for both values of ε, while the cJ preconditioner exhibits linear growth.

In short, the story told in Figure 3 is that the generalized Gauss–Seidel (gGS and gSGS) preconditioners perform consistently well (the latter also with Conjugated Gradients), with both a low number of iterations and a low growth rate. The gJ(1)/gJ(4) preconditioners also exhibit a low rate of growth, and their higher absolute iteration count is at least partly offset by a lower computational cost per iteration.

Discontinuities in both permeability and elastic parameters

In the experiments we have looked at so far, the elastic parameters have been constant throughout the domain. We now proceed to investigate the effect of discontinuous elastic material parameters. This is a more realistic case of two different geological materials. The parameters of the (soft,

DOI: 10.1002/nag

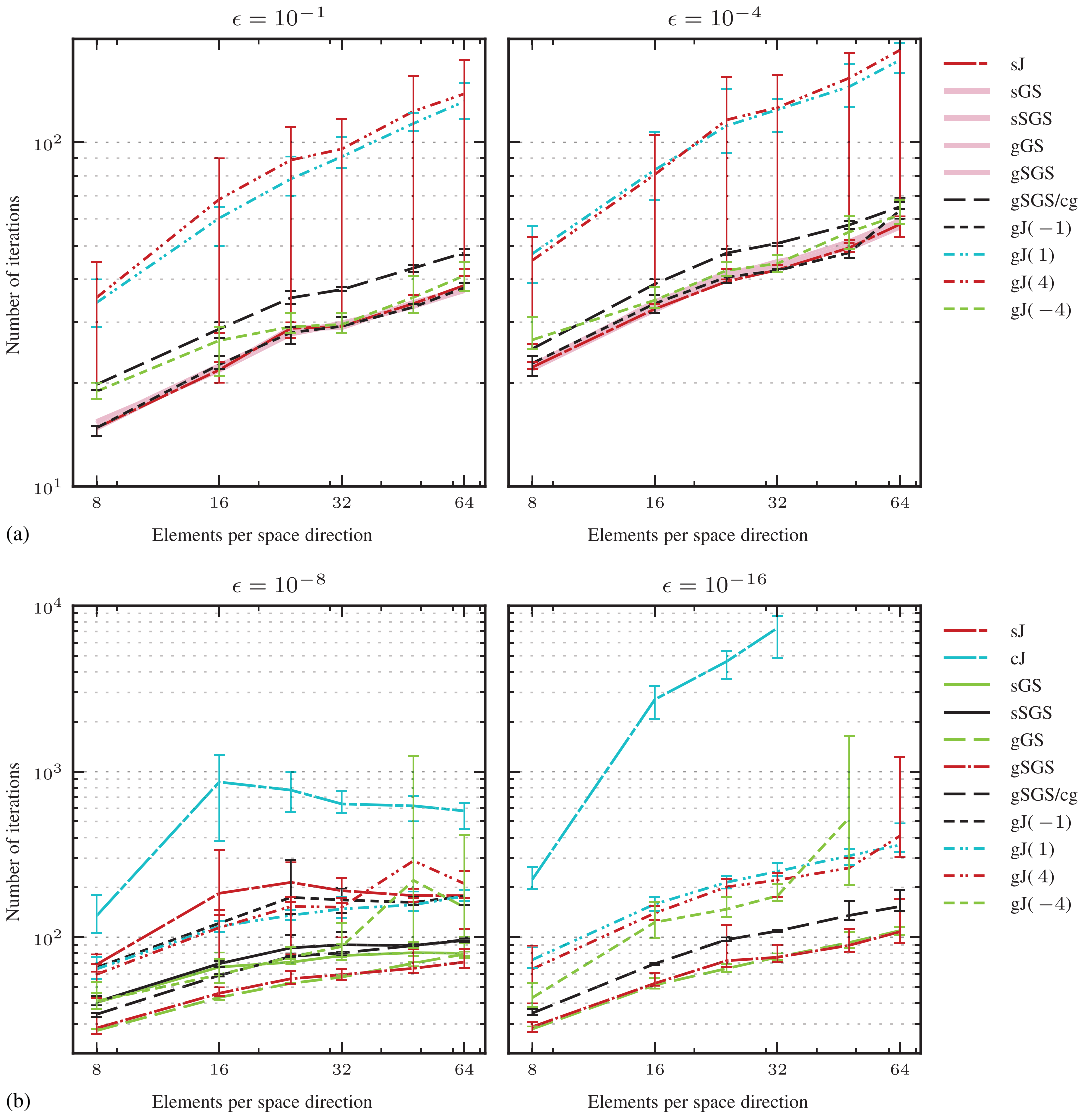

Figure 4. Iterations to reach convergence for the model problem with discontinuous elastic parameters. (a) Low-to-moderate permeability contrast. The Gauss–Seidel methods overlap, and are drawn in gray for legibility and (b) Severe permeability contrast. With moderate permeability contrasts, (a), the tested preconditioners show a growth rate of roughly $\sqrt{N}$, with N being the number of divisions in each space direction. When the contrasts are stronger, (b), the picture is more complicated; but the generalized Gauss–Seidel preconditioners remain efficient.

high-permeable) surrounding subdomain Ω_0 are the same as in the constant-parameter case. Inside Ω_ε, the scaled elastic parameters are $\lambda = 1.43 \times 10^5$ and $\mu = 3.57 \times 10^4$, corresponding to Young's modulus $E = 10^5$ and Poisson's ratio $v = 0.4$.

Performance with moderate jumps in permeability. Figure 4(a) shows the results for a moderate discontinuity in permeability. The general behavior of the preconditioners is quite similar to the constant-elasticity case, differing mostly by a multiplicative factor (on average, the number of iterations is about doubled). Except for the $\alpha > 0$ generalized Jacobi variants, all preconditioners perform equally well, with a growth rate in the range $N^{0.3} - N^{0.4}$ (see Table III(b)). This demonstrates that heterogeneity in the elastic parameters is not in itself a major difficulty with these block preconditioners.

DOI: 10.1002/nag

Performance with severe jumps in permeability. When the permeability contrasts are strengthened, however, we see some changes relative to the constant-elasticity case. This is shown in Figure 4(b) (compared with Figure 3(b)). Four of the preconditioners have the same behavior as they did with uniform elasticity. These are the sJ and gJ(-1) Jacobi-scheme methods, which fail, and the gGS and gSGS generalized Gauss–Seidel methods, which converge robustly. But the remaining preconditioners behave differently in the problem with discontinuous elastic parameters.

At $\varepsilon = 10^{-8}$, the standard Gauss–Seidel preconditioners, sGS and sSGS, perform very well, while the capped-ε Jacobi method (cJ) actually converges *faster* as N grows (although the number of iterations is still much higher than for the other methods). All these methods were among the worst performers with the same value of ε and uniform elastic parameters. These anomalies disappear in the most discontinuous case, where $\varepsilon = 10^{-16}$; here, the standard basis (sGS and sSGS) methods do not converge at all, and the cJ and gJ(-4) methods fail for large N. The latter result is in line with its performance in the continuous-elasticity case, Figure 3(b). While the good result at $\varepsilon = 10^{-8}$ is surprising, it has little practical significance since the effect appears to be a result of particular combinations of parameters.

We note that the only four preconditioners that achieve convergence for all values of ε are the same that performed best in the constant-elasticity test: The positive-α generalized Jacobi methods gJ(1) and gJ(4), and the generalized Gauss–Seidel methods gGS and gSGS (with either BiCGStab or CG iterations). The high sensitivity of gJ(4) to the initial vector, seen most clearly in Figure 4(a) as a large variance in the results, can be construed either as a warning flag, or as a sign that it can potentially be more efficient if certain (unidentified) modes are not present in the initial guess.

The orders of the different methods, when used with discontinuous elastic material parameters, is given on the right side of Table III(b). The gSGS method does not go significantly above $\mathcal{O}(N^{0.5})$ in any of the tests—a remarkably robust result.

Summary of experimental results

Figure 5(a) summarizes the performance of the successful preconditioners for the largest problem size, $N = 65$. The ones that fail to converge in one or more of the tests are similarly shown in Figure 5(b). It is clear that when $\varepsilon \geqslant 10^{-4}$, it does not matter much which preconditioner is chosen; they all converge, and with the exception of the generalized Jacobi preconditioners gJ(1)/gJ(4) they are equally effective. When the permeability jump becomes larger, however, there are only four preconditioners that converge consistently with every combination of material parameters: the generalized Gauss–Seidel methods gGS/gSGS, and the generalized Jacobi methods gJ(1)/gJ(4), again with gJ(α) being least efficient. Additionally, gSGS/cg (which is solved with Conjugated Gradients) performs well in all cases. Although the number of iterations is higher for this method, the cost per iteration is lower than with BiCGStab, resulting in faster overall performance.

We further note that:

- The standard-basis family of block methods (sJ, sGS and sSGS) does not work well for this problem.
- The Generalized Jacobi family of block preconditioners is unstable with negative α, even though these are more efficient with less severe discontinuities. Positive α is stable, but requires a large number of BiCGStab iterations to converge. The magnitude of α seems to be of less importance, although the variance is much higher with $\alpha = 4$ than with $\alpha = 1$.
- The capped-ε cJ preconditioner is stable, although inefficient for large permeability jumps, when the elastic parameters are uniform; but it fails for large jumps in the discontinuous-elasticity cases.
- The generalized symmetric Gauss–Seidel (gSGS) block preconditioner performs well in all cases.
- The gGS block preconditioner, which is a simplified variant of gSGS, performs almost as well (but fails to preserve symmetry, limiting the choice of iterative solver).
- The gSGS block preconditioner with sufficiently accurate single-block preconditioners transforms the problem into a symmetric positive definite one, which can be solved by the

DOI: 10.1002/nag

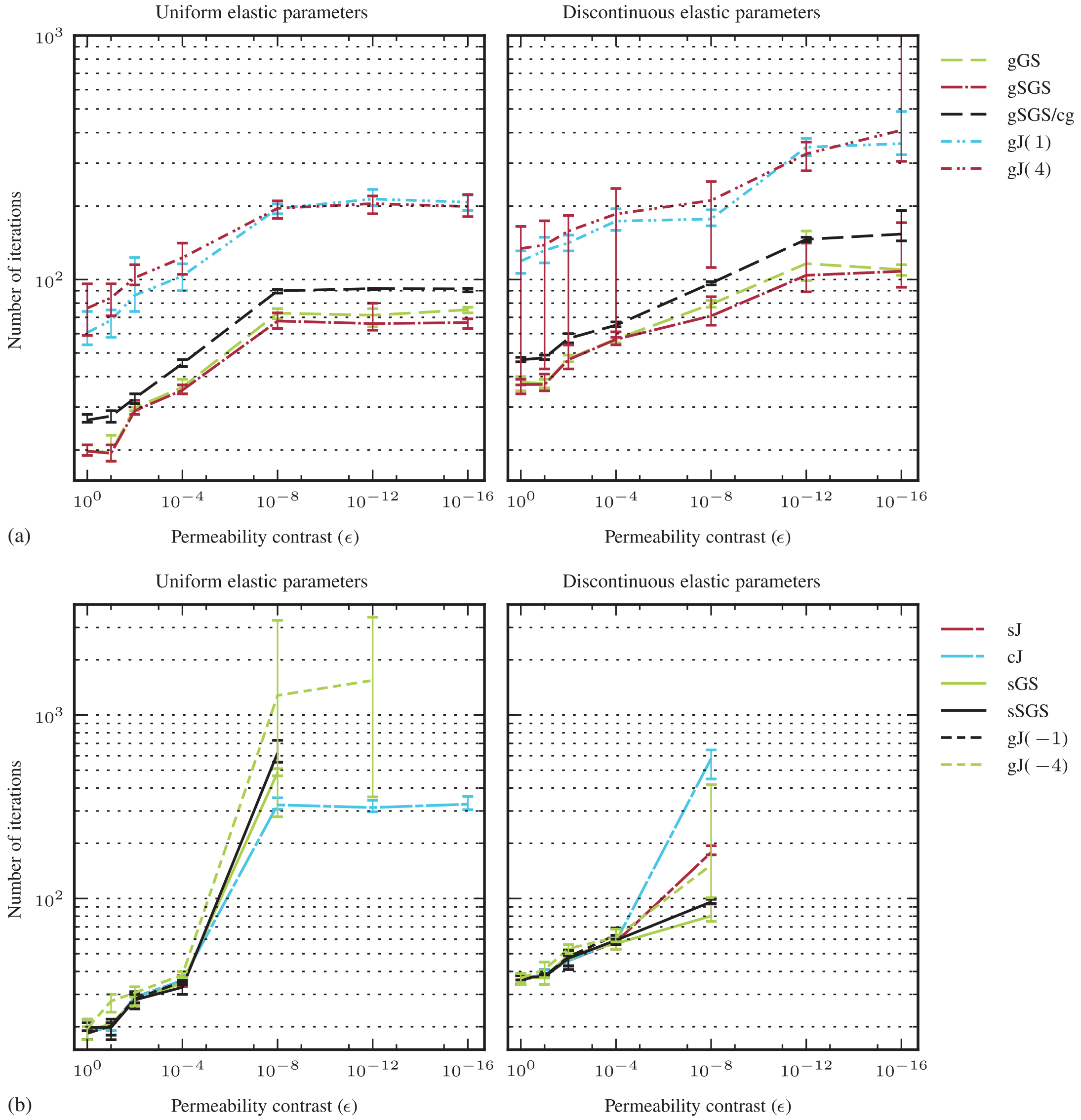

Figure 5. The ε-dependence of the preconditioners with $N = 65$ displacement nodes in each space direction. The upper plots show the methods that converged in all tests, while those that failed to converge for some combination of parameters are shown at the bottom.

Conjugated Gradient method. This combination is denoted as the gSGS/cg method. The AMG method combined with a cheap approximation of the Schur complement is sufficiently accurate for the model problems presented in this paper.

CONCLUDING REMARKS

The iterative solution of large-scale problems in geomechanics requires efficient and robust preconditioners. While a number of preconditioners for Biot's equation (and similar symmetric indefinite problems) have been put forth in the literature, their performance with highly discontinuous permeability has to our knowledge not previously been systematically evaluated. This paper evaluates several block preconditioners for this problem in the presence of severe jumps in the material parameters.

DOI: 10.1002/nag

Our investigations reveal that discontinuous material parameters, which are present in many realistic geological scenarios, pose a serious challenge for iterative solution methods. Indeed, some seemingly attractive methods converge very slowly, or fail to converge, on a model problem when the heterogeneities are sufficiently strong. These include the standard block Jacobi and block Gauss–Seidel preconditioners [21], as well as the generalized Jacobi block preconditioner [22] with $\alpha<0$. The generalized Jacobi block preconditioner with $\alpha>0$ does, however, converge at an acceptable rate.

Using Algebraic Multigrid as the single-block preconditioners and a cheap approximation to the Schur complement, we identify two block preconditioners that perform consistently well on Biot's equation with severe jumps in permeability and discontinuous elastic parameters. These two, one symmetric and one non-symmetric variant of the generalized Gauss–Seidel method, are (in our interpretation) based on an exact blockwise inversion of the coupled equations. The performance of these preconditioners is very good, with a number of BiCGStab iterations, which is about one third of the generalized Jacobi preconditioner with $\alpha>0$. Furthermore, the symmetric variant leads (under certain assumptions) to a symmetric positive definite problem, which can be solved by the Conjugate Gradient method.

Given that AMG preconditioners have shown themselves to scale to massively parallel computers [33, 34], and that the methods presented herein only have minor parallel communication requirements beyond those of AMG, we anticipate that this combined block preconditioner is equally scalable. This assertion must however be investigated in more detail, which will be performed in a forthcoming paper.

Moreover, owing to its construction from an exact decomposition, we believe that the generalized symmetric Gauss–Seidel preconditioner is widely useful for general difficult coupled problems where the single blocks A and S are individually invertible.

ACKNOWLEDGEMENTS

We thank Kent-Andre Mardal and Xing Cai at Simula Research Laboratory for their advice and support. We also thank Statoil for their support, both financial and in access to their geological models and expertize. This work was supported by a Center of Excellence grant from the Norwegian Research Council to Center for Biomedical Computing at Simula Research Laboratory.

REFERENCES

1. Terzaghi K. Die berechnung der durchlassigkeitsziffer des Tones aus dem verlauf der hydrodynamischen spannungserscheinungen. *Sitzungsberichte der Akademie der Wissenschaften in Wien*, *Mathematisch-Naturwissenschaftliche Klasse*, *Abteilung IIa* 1923; **132**:125–138.
2. Biot MA. General theory of three-dimensional consolidation. *Journal of Applied Physics* 1941; **12**(2):155–164. DOI: 10.1063/1.1712886.
3. Boutéca M, Bary D, Piau J, Kessler N, Boisson M, Fourmaintraux D. Contribution of poroelasticity to reservoir engineering: lab experiments, application to core decompression and implication in HP-HT reservoirs depletion. *Rock Mechanics in Petroleum Engineering* 1994; DOI: 10.2118/28093-MS.
4. Lewis RW. Coupling of fluid flow and deformation in underground formations. *Journal of Engineering Mechanics* 2002; **128**:779. DOI: 10.1061/(ASCE)0733-9399(2002)128:7(779).
5. Lewis RW, Makurat A, Pao WKS. Fully coupled modeling of seabed subsidence and reservoir compaction of north sea oil fields. *Hydrogeology Journal* 2003; **11**(1):142–161.
6. Neuzil CE. Hydromechanical coupling in geologic processes. *Hydrogeology Journal* 2003; **11**:41–83.
7. Showalter RE. Diffusion in deforming porous media. *Dynamics of Continuous Discrete and Impulsive Systems-Series A-Mathematical Analysis* 2003; **10**(5):661–678.
8. Wang HF. *Theory of Linear Poroelasticity with Applications to Geomechanics and Hydrogeology*. Princeton University Press: Princeton, 2000.
9. George A, Ng E. On the complexity of sparse QR and LU factorization of finite-element matrices. *SIAM Journal on Scientific and Statistical Computing* 1988; **9**:849. DOI: 10.1137/0909057.
10. Doi S, Washio T. Ordering strategies and related techniques to overcome the trade-off between parallelism and convergence in incomplete factorizations. *Parallel Computing* 1999; **25**:1995–2014. DOI: 10.1016/S0167-8191(99)00064-2.
11. Hestenes MR, Stiefel E. Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards* 1952; **49**(6):409–436.

DOI: 10.1002/nag

12. van der Vorst HA. Bi-CGSTAB: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1992; **13**(2):631–644.
13. Hackbush W. *Iterative Solution of Large Sparse Systems of Equations*. Springer: Berlin, 1995.
14. Strang WG, Fix GJ. *Analysis of the Finite Element Method*. Prentice-Hall: Englewood Cliffs, NJ, 1973.
15. Bank RE, Dupont T. An optimal order process for solving finite element equations. *Mathematics of Computation* 1981; 35–51. DOI: 10.2307/2007724.
16. Benzi M. Preconditioning techniques for large linear systems: a survey. *Journal of Computational Physics* 2002; **182**(2):418–477. DOI: 10.1006/jcph.2002.7176.
17. Yang UM. Parallel algebraic multigrid methods—high performance preconditioners. *Numerical Solution of Partial Differential Equations on Parallel Computers*, Bruaset AM, Tveito A (eds). Springer: Berlin, 2006; 209–236. DOI: 10.1007/3-540-31619-1_6.
18. Ruge JW, Stuben K. Algebraic multigrid. In *Multigrid Methods*, vol. 3, McCormick SF (ed.). SIAM: Philadelphia, PA, 1987; 73–130.
19. Adams M. Evaluation of three unstructured multigrid methods on 3D finite element problems in solid mechanics. *International Journal of Numerical Methods in Engineering* 2002; **55**:519–534. DOI: 10.1002/nme.506.
20. Haga JB, Langtangen HP, Nielsen BF, Osnes H. On the performance of an algebraic multigrid preconditioner for the pressure equation with highly discontinuous media. *Proceedings of MekIT'09*, Skallerud B, Andersson HI (eds). NTNU: Tapir, 2009; 191–204.
21. Mardal KA, Sundnes J, Langtangen HP, Tveito A. Systems of PDEs and block preconditioning. *Advanced Topics in Computational Partial Differential Equations*, Langtangen HP, Tveito A (eds). Springer: Berlin, 2003; 199–236.
22. Phoon KK, Toh KC, Chan SH, Lee FH. An efficient diagonal preconditioner for finite element solution of Biot's consolidation equations. *International Journal for Numerical Methods in Engineering* 2002; **55**:377–400. DOI: 10.1002/nme.500.
23. Toh KC, Phoon KK, Chan SH. Block preconditioners for symmetric indefinite linear systems. *International Journal for Numerical Methods in Engineering* 2004; **60**:1361–1381. DOI: 10.1002/nme.982.
24. Bear J. *Dynamics of Fluids in Porous Media*. Dover: New York, 1988. DOI: 10.1097/00010694-197508000-00022.
25. Neuzil CE. How permeable are clays and shales? *Water Resources Research* 1994; **30**(2):145–150. DOI: 10.1029/93WR02930.
26. Nielsen BF. Finite element discretizations of elliptic problems in the presence of arbitrarily small ellipticity: an error analysis. *SIAM Journal on Numerical Analysis* 1999; 368–392. DOI: 10.1137/S0036142997319431.
27. Liu R, Wheeler MF, Dawson CN. Discontinuous Galerkin finite element solution for poromechanics. *Ph.D. Thesis*, University of Texas at Austin, 2006.
28. Phillips PJ, Wheeler MF. Overcoming the problem of locking in linear elasticity and poroelasticity: an heuristic approach. *Computational Geosciences* 2009; **13**(1):5–12. DOI: 10.1007/s10596-008-9114-x.
29. Shewchuk JR. An introduction to the conjugate gradient method without the agonizing pain. *Technical Report*, Carnegie Mellon University, Pittsburgh, PA, U.S.A., 1994.
30. Bramble JH, Pasciak JE. A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems. *Mathematics of Computation* 1988; **50**(181):1–17. DOI: 10.2307/2007912.
31. Tuminaro RS, Tong C. Parallel smoothed aggregation multigrid: aggregation strategies on massively parallel machines. *Proceedings of the 2000 ACM/IEEE Conference on Supercomputing*. IEEE Computer Society: Silver Spring, MD, 2000; DOI: 10.1109/SC.2000.10008.
32. Brezina M, Falgout R, MacLachlan S, Manteuffel T, McCormick S, Ruge J. Adaptive smoothed aggregation (αSA). *SIAM Journal on Scientific Computing* 2004; **25**(6):1896–1920. DOI: 10.1137/S1064827502418598.
33. Adams MF, Bayraktar HH, Keaveny TM, Papadopoulos P. Ultrascalable implicit finite element analyses in solid mechanics with over a half a billion degrees of freedom. *Proceedings of the ACM/IEEE Conference on Supercomputing* (*SC2004*). IEEE Computer Society: Silver Spring, MD, 2004; 34.
34. Joubert W, Cullum J. Scalable algebraic multigrid on 3500 processors. *Electronic Transactions on Numerical Analysis* 2006; **23**:105–128.
35. Chow E, Falgout RD, Hu JJ, Tuminaro RS, Yang UM. A survey of parallelization techniques for multigrid solvers. In *Parallel Processing for Scientific Computing*, Heroux MA, Raghavan P, Simon HD (eds). Society for Industrial and Applied Mathematics (SIAM), 2006; 179–202.
36. http://www.diffpack.com/, library for numerical solution of PDEs from inuTech GmbH.
37. Gee MW, Siefert CM, Hu JJ, Tuminaro RS, Sala MG. ML 5.0 smoothed aggregation user's guide. *Technical Report SAND2006-2649*, Sandia National Laboratories 2006. Available from: http://software.sandia.gov/trilinos/packages/ml/.
38. Heroux MA, Bartlett RA, Howle VE, Hoekstra RJ, Hu JJ, Kolda TG, Lehoucq RB, Long KR, Pawlowski RP, Phipps ET *et al.* An overview of the Trilinos project. *ACM Transactions on Mathematical Software* 2005; **31**(3):397–423. DOI: 10.1145/1089014.1089021.