On the performance of an algebraic multigrid preconditioner for the pressure equation with highly discontinuous media

J. B. Haga,[*12] **H. P. Langtangen,**[13] **B. F. Nielsen**[13] **and H. Osnes**[12]
[1]Simula Research Laboratory
[2]Department of Mathematics, University of Oslo
[3]Department of Informatics, University of Oslo
[*]e-mail: jobh@simula.no

Summary We study the performance of an algebraic multigrid (AMG) preconditioner combined with conjugate gradient-type methods for the pressure equation in the presence of strong discontinuity of the permeability. Such discontinuities are problematic for iterative solvers, due to the dependence of the condition number of the discretized linear system of equations on the magnitude of the discontinuities. We identify two separate causes of such ill-conditioning, and assess how well the AMG preconditioner handles these. For the more difficult problems, we discuss techniques to modify the model to allow iterative solutions.

Introduction

Single-phase, incompressible fluid flow in porous media is described by the equation

$$\nabla \cdot (\Lambda \nabla p) = q, \tag{1}$$

where p is the fluid pressure, Λ is the mobility of the porous medium, and q is a source term modeling injection or extraction of the fluid as well as the effect of body forces, i.e., gravity. In general, Λ is a tensor reflecting the medium's ability to transport fluid in different space directions, and is proportional to the permeability of the porous space. The permeability, and hence Λ, may face significant jumps of several orders of magnitude in geological applications. This feature may have severe impact on the performance of numerical methods for solving Equation (1), which is the topic we investigate in this paper.

We shall assume that Equation (1) is discretized by a Galerkin finite element method with bilinear elements and n nodes in total. Furthermore, we let ϵ denote the typical jump in Λ, meaning that we basically consider two types of geological media: one with scaled permeability of order unity and one with scaled permeability ϵ.

The impact of the jump ϵ on the accuracy of the finite element discretization is not critical as long as the discontinuities are aligned with the element boundaries, which we assume in the following [7, ch. 2.8.2]. The critical numerical impact of discontinuities in Λ is then on the solution of the linear system $\mathrm{A}x = b$ arising from Equation (1). The condition number κ of the coefficient matrix A will behave as $\kappa \sim \epsilon^{-1} n^{2/d}$, where d is the number of space dimensions [7, ch. 2.10.3]. The number of iterations in iterative methods, such as the Conjugate Gradient (CG) method, for solving $\mathrm{A}x = b$ typically depends on $\sqrt{\kappa}$. By applying a preconditioner M to the linear system, i.e., by solving $\mathrm{M}^{-1}\mathrm{A}x = \mathrm{M}^{-1}b$ one can reduce the condition number and obtain faster convergence. Using a multilevel method as preconditioner, the condition number of the coefficient matrix in the preconditioned system can be made independent of n, meaning that the work associated with the iteration

This work has been financially supported by StatoilHydro

method does not increase faster than the number of unknowns in the system [6, ch. 10]. This property is important when dealing with large-scale computations.

In problems with discontinuities of size ϵ, the condition number of A is proportional to ϵ^{-1}, as pointed out above. To our knowledge, preconditioning techniques are not known to remove this unfortunate dependence on ϵ. However, Cai et al. [1] showed that the preconditioned Conjugate Gradient method may be insensitive to small ϵ values if one ensures that the iterations are performed in a suitable subspace, which can be reached through a starting iterate that corresponds to a harmonic function (say a constant) in the part of the domain where the permeability is small (ϵ). As preconditioner, Cai et al. applied a constant-coefficient Laplace operator (an FFT-based solution method).

The algebraic multigrid (AMG) method has recently attracted quite some interest as an efficient and widely applicable preconditioner. A difficulty with standard geometric multigrid 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 typically the case in geological applications of Equation (1). AMG is then a promising alternative. We assume that AMG can remove the dependence of the number of iterations on n, but how AMG treats the dependence on ϵ is an open question.

There are two principal difficulties with low-permeable zones in geological applications. One is the jump in Λ and its effect on linear solvers as described above. Another is that a low-permeable zone may isolate a higher-permeable zone from the Dirichlet boundary conditions. In the limit $\epsilon \rightarrow 0$ it then acts as an impermeable boundary, dividing the original problem into two subproblems. In this limit, one of the subproblems approaches a homogeneous Neumann problem (with $\Lambda \nabla p \cdot n = 0$ on the boundary). This Neumann problem is singular up to an additive constant, which may cause problems in the linear solver. It is known, however, that the choice of initial iterate in practice determines this constant when the linear system is solved by the conjugate gradient method.

The present paper investigates the convergence of an AMG-preconditioned conjugate gradient-type method applied to the linear system arising from Equation (1). Our aim is to extend common knowledge from earlier work by investigating a series of cases. We divide the cases into two main categories: (i) compact low-permeable zones which the fluid can flow around, and (ii) a low-permeable layer spanning the complete domain in horizontal direction, or otherwise isolating a high-permeable region. Case (ii) may arise in geological applications and has the danger of leading to a not well-posed mathematical problem in the limit $\epsilon \rightarrow 0$. How does the efficiency of the iterative method depend on ϵ in this case? And how does it depend on the degree of anisotropy in Λ? Also, in more complicated problems Equation (1) is coupled to other equations and the coefficient matrix is likely to become non-symmetric so that the Conjugate Gradient method cannot be applied. How do methods for non-symmetric systems, such as the BiCGStab and GMRES methods, behave compared to the Conjugate Gradient method in our present test problem?

Numerical experiments

The simulator is implemented in Diffpack [7], which is an object-oriented C++ finite element framework. Diffpack has available a number of well-tested iterative solvers which we have modified to suit our needs, for example to output local convergence results and to recalculate the residual, which is important because the orthogonality (in A-space) of the residual is easily lost when operating near the limits of machine precision. The latter modification is called the *restarted* or *cyclic* CG (or BiCGStab) method [6]. We have also interfaced the Diffpack linear algebra operations with the ML [5] algebraic multigrid / smoothed aggregation library to use as preconditioner, and to the SuperLU [3] sparse direct solver for providing more exact solutions, independently of convergence criteria, when needed.

For eigenvalue calculations, we have employed the `eig` function of Octave [4]. We have found that for extremely small eigenvalues (near machine precision) the results are unreliable, hence we do not report the exact value of eigenvalues below 10^{-14} in the tables that follow.

In the convergence tests, we solve (unless explicitly noted otherwise) the equivalent problem $\mathrm{A}x = 0$, instead of $\mathrm{A}x = b$, with a random initial iterate x_0. Convergence of linear solvers is in general independent of the right-hand side b as long as the initial guess contains all eigenvectors of A, so the convergence measurements are valid even if the equation is different from the original (see for example [6, ch. 3.4]).[1] With this choice of right-hand side, the error norm $\|e_k\|_{L_2}$ is trivially available, since $e_k = x_k$. The convergence criterion used when not otherwise specified is $\|e_k\|_{L_2} < 10^{-20}\|e_0\|_{L_2}$.

Case (i) — Compact subdomains

Many applications deal with compact low-permeable subdomains, where by compact we mean that the low-permeable region does not isolate any high-permeable region from the Dirichlet boundary conditions. Such problems can be dealt with by performing a domain splitting procedure, where the low-permeable regions are excluded from the domain. Assuming that ϵ is sufficiently small, the pressure equation Equation (1) can be solved on just the high-permeable region, with a no-flow condition on the interfaces to the excluded subdomains [8, 9]. Finally, if the pressure solution inside the low-permeable subdomains is required, the pressure equation can then be solved on just the low-permeable subdomains, using the high-permeable solution as boundary conditions. As shown in [8], the error introduced in the flow region by this procedure is on the order of ϵ, and thus quite acceptable when ϵ is small. However, the domain splitting increases implementation complexity, particularly when it is required to handle unstructured grids of highly variable permeability. As an alternative to domain splitting, we therefore look at how the whole-domain problem is handled by AMG-preconditioned conjugate gradient-type solvers.

Figure 1 shows three different cases where low-permeable subdomains are embedded in a high-permeable domain. In case (a), a thin layer (of one element thickness) extends almost across the domain, with a small gap. In case (b), the layer is thicker, and in case (c) the number of thin layers is increased to 4, in effect increasing the distance from the bottom of the domain to the Dirichlet conditions at the top.

The numerical results from these three cases are shown in Table 1, for ϵ varying from 10^{-1} to 10^{-20}. We notice that for the two thin cases, (a) and (c), the condition number of the unpreconditioned coefficient matrix A is bounded. This can be explained by the fact that every node in the low-permeable subdomains Ω_ϵ is in contact with the high-permeable region Ω_0. Still, the condition number grows slowly as the effective distance to the Dirichlet boundary increases. For the thick layer, case (b), the condition number of A is *not* bounded as $\epsilon \rightarrow 0$ (we shall look closer at this behavior in the following section),

[1] This may not always hold in inexact arithmetic.

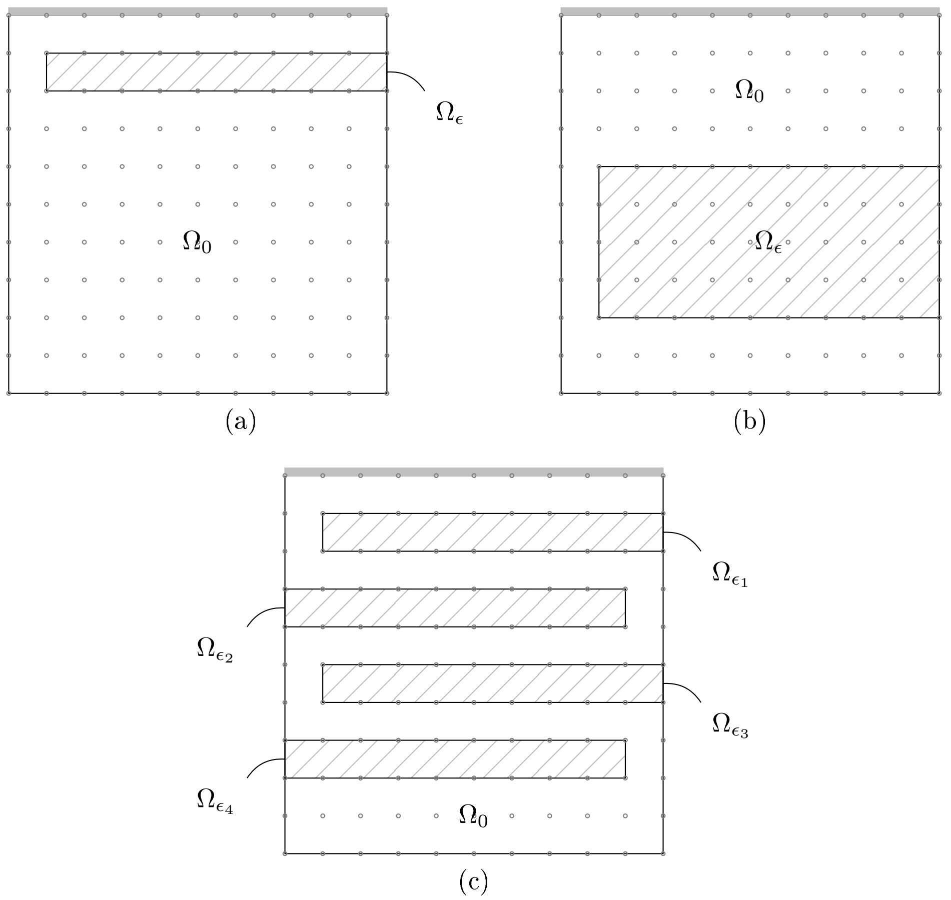


Figure 1: Three cases with compact low-permeable subdomains Ω_ϵ, of varying number and thickness. Dirichlet boundary conditions are in all cases applied to the top of the domain, while the other boundaries have Neumann conditions. The node positions are shown as circles.

but the AMG preconditioner is highly effective for this case, and all three cases converge at a high rate.

These tests show that unmodified AMG-preconditioned conjugate gradient-type methods are effective for handling compact low-permeable regions, and have the advantage of not requiring splitting of the domain and associated regridding.

Case (ii) – Non-compact (isolating) subdomains

A different situation arises when the low-permeable zone stretches across the whole domain, or a high-permeable region is embedded within a low-permeable zone. In the limit of $\epsilon \to 0$, the problem in fact becomes unphysical, and the inside pressure is undetermined. For $\epsilon > 0$, however, a unique solution exists, but the extreme ill-conditioning is a challenge to iterative solvers. Figure 2 shows three examples where such nearly isolated high-permeable subdomains are present. These are variations of the test cases presented previously, where isolated high-permeable regions are created by extending the layer across the domain in (a) and (c), and by embedding an additional high-permeable region in (b). The isolated high-permeable regions are marked $\Omega_{i>0}$.

The results of solving these cases are shown in Table 2. The contrast from the compact

Table 1: Results for compact low-permeable domains. The 11×11 grid shown in Figure 1 is used. The logarithm of the condition number κ is compared for varying ϵ, along with the number of iterations required by different iterative solvers. Condition numbers below 10^{-14} are not reported exactly.

	$\log \kappa(\mathrm{A})$			$\log \kappa(\mathrm{M}^{-1}\mathrm{A})$			CG it.			BCGS it.			GMRES(10) it.		
ϵ	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
10^{-1}	2.6	2.7	2.8	.15	.12	.35	14	14	18	8	9	12	20	20	30
10^{-2}	3.0	3.1	3.5	.41	.14	.94	16	15	24	10	9	16	20	20	50
10^{-4}	3.1	4.9	3.8	.49	.14	1.2	16	15	26	10	10	18	20	20	60
10^{-8}	3.1	8.9	3.8	.49	.14	1.2	16	16	26	11	11	19	20	20	60
10^{-12}	3.1	13.9	3.8	.49	.14	1.2	16	17	26	11	12	19	20	20	60
10^{-16}	3.1	>14	3.8	.49	.14	1.2	16	19	26	11	13	19	20	20	60
10^{-20}	3.1	>14	3.8	.49	.14	1.2	16	18	26	11	12	18	20	20	60

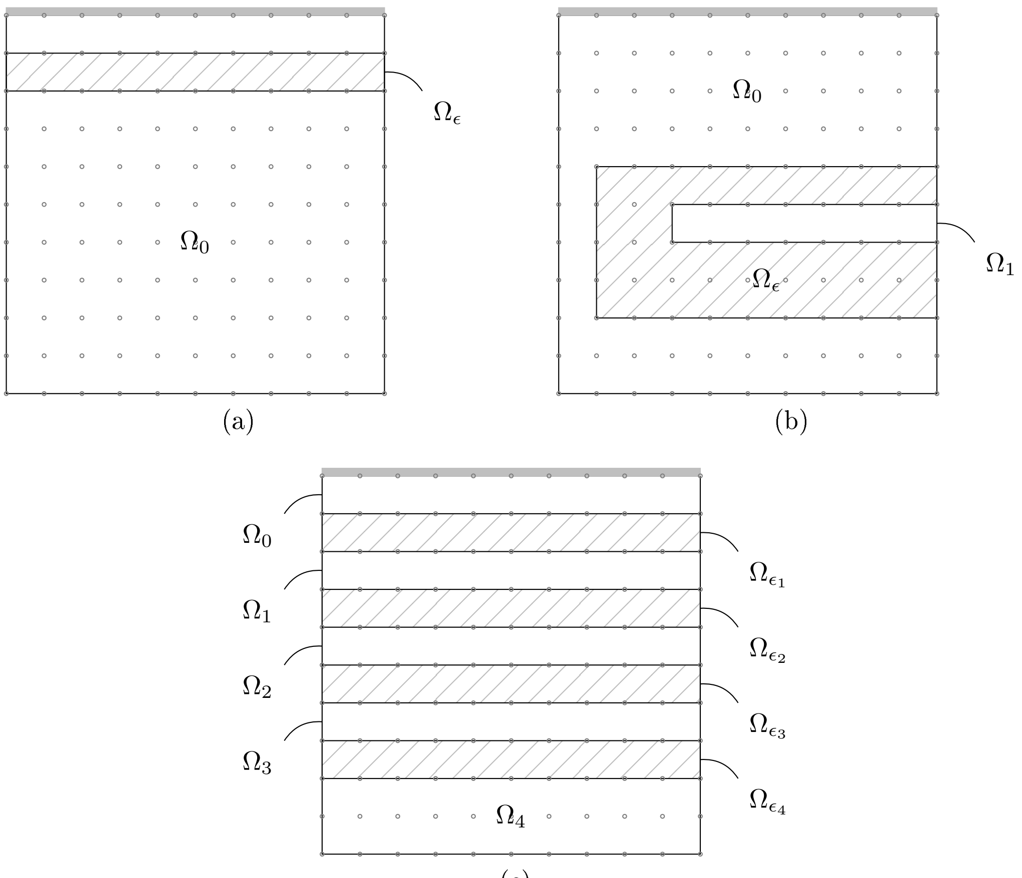
cases in the previous section is stark: Not only is the condition number of A unbounded as $\epsilon \to 0$ in all three cases (a)–(c), but the preconditioner is ineffective in curing it. Nevertheless, the Conjugate Gradient method is able to handle many of these quite ill-conditioned matrices effectively, until the curse of inexact arithmetic takes hold. BiCGStab performs less well. A look at the eigenvalue distributions, shown in Figure 3, is illuminating.

We consider first the eigenvalues for the compact subdomains. The eigenvalues of the original operator A is shown as a solid red line. In cases (a) and (c), these eigenvalues are nearly independent of ϵ. This is to be expected, since there are no nodes in these cases that are surrounded by a low-permeable region. In case (b), however, 27 eigenvalues are of order ϵ. This is exactly the number of nodes (9×3) that are inside the low-permeable region Ω_ϵ. This case demonstrates the effectiveness of the AMG preconditioner for this class of problems: The problematic eigenvalues are perfectly canceled in $\mathrm{M}^{-1}\mathrm{A}$, shown in the figure as a solid green line.

Next we consider the cases of non-compact subdomains, where we have seen that the preconditioned solvers fail to converge for small ϵ. These are shown as dotted lines; red for A, green for $\mathrm{M}^{-1}\mathrm{A}$. In case (a), A has one eigenvalue of order ϵ. In case (c) it has four, one for each isolated high-permeable region. Neither of these are cancelled in $\mathrm{M}^{-1}\mathrm{A}$. Furthermore, of the 12 problematic eigenvalues in case (b), 11 are caused by nodes inside the low-permeable domain Ω_ϵ, and one by the embedded high-permeable domain Ω_1. This count has been verified by running the experiment at different resolutions. The preconditioned operator, however, has exactly one eigenvalue of order ϵ. This confirms that there are indeed two separate issues: One is that of nodes inside low-permeable regions, which is handled perfectly well by the AMG operator, while the other is that of ill-posed Neumann subproblems on isolated high-permeable regions, which is not.

The Conjugate Gradient method is known to handle a small number of extreme eigenvalues well, a phenomenon known as superconvergence [6, ch. 9.43], which explains why it is still rather effective for moderate values of ϵ. The non-symmetric variants BiCGStab and GMRES, meanwhile, seem to handle a single runaway eigenvalue well, but get into trouble when there is more than one.

It is interesting to note that even in the cases where CG/BiCGStab does not converge in



(a)

(b)

(c)

Figure 2: Three test cases where high-permeable subdomains $\Omega_{i>0}$ are isolated from the Dirichlet boundary conditions by one or more low-permeable subdomains Ω_ϵ. In the limit of $\epsilon = 0$, these problems are ill-posed, but a unique solution exists for any $\epsilon > 0$. All cases have Dirichlet conditions on the top boundary. The node positions are shown as circles.

the full domain, it does exhibit fast local convergence in the connected high-permeable region Ω_0. Furthermore, in the non-connected high-permeable regions, $\Omega_{i>0}$, the computed solution is constant (although not the correct constant). Local convergence is actually significantly faster in the cases where global convergence fails, but we have no cogent explanation for this observation.

The error

So far we have looked at the properties — and thus the solvability — of the coefficient matrix from a solution-independent point of view. Assuming that the preconditioned coefficient matrix has a solvable structure (eigenvalue distribution), we now look at the actual error in the solution. As the jumps in Λ approaches machine precision,[2] we can no longer have confidence that even a direct solver can give an accurate solution. Another difficulty is that of satisfying a residual-based convergence criterion for iterative solvers. In the energy norm, values inside low-permeable zones is weighted by ϵ, meaning that the

[2] We can define the machine precision limit as the ratio at which a number y becomes additive zero with respect to a floating-point number x, such that $x + y = x$. In double precision, this happens when $y/x \leq 2^{-53} \approx 1.1 \cdot 10^{-16}$.

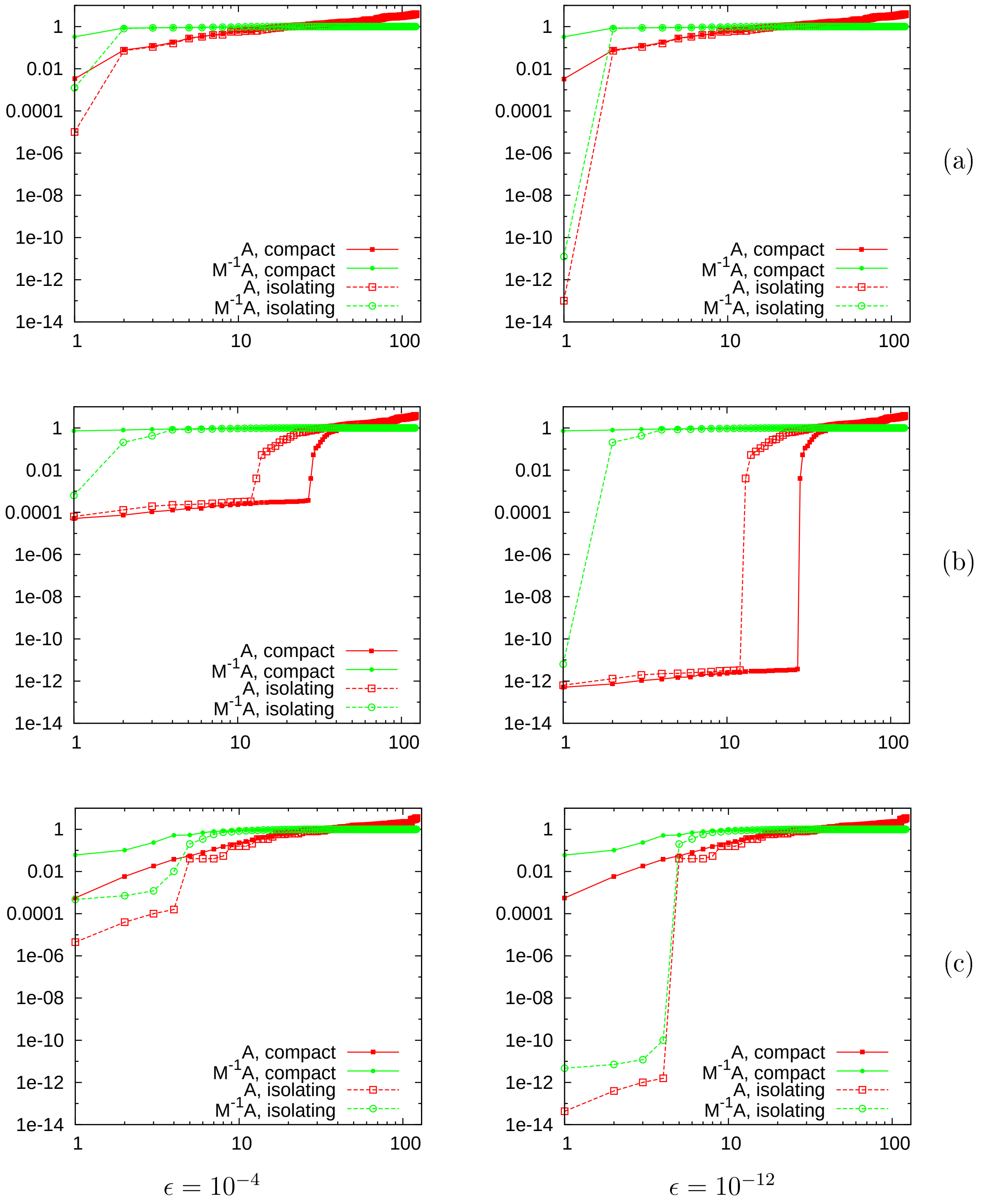


Figure 3: The eigenvalues of cases (a)–(c) (top to bottom), ordered by magnitude. The x axis is thus the index of the eigenvalue. Two different values of ϵ are shown (left/right). The eigenvalues of the original operator A are shown in red (squares), the preconditioned operator $\mathrm{M}^{-1}\mathrm{A}$ in green (circles). The compact cases (from Figure 1) are solid lines (filled symbols), non-compact (from Figure 2) are dotted lines (open symbols).

Table 2: Results for low-permeable domains which isolate high-permeable subdomains from the Dirichlet boundary. The 11×11 grid shown in Figure 1 is used. The logarithm of the condition number κ is compared for varying ϵ, along with the number of iterations required by different iterative solvers. Non-convergence is denoted as "—", and condition numbers below 10^{-14} are not reported exactly.

	$\log\kappa(\mathrm{A})$			$\log\kappa(\mathrm{M}^{-1}\mathrm{A})$			CG it.			BCGS it.			GMRES(10)		
ϵ	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
10^{-1}	2.7	2.7	2.9	.20	.44	.42	15	17	19	9	11	13	20	20	30
10^{-2}	3.6	3.0	3.9	.94	1.2	1.3	17	22	31	13	14	24	20	30	100
10^{-4}	5.6	4.8	5.9	2.9	3.3	3.3	20	24	46	13	18	53	30	40	550
10^{-8}	9.6	8.8	9.9	6.9	7.3	7.3	31	38	120	15	20	—	110	130	—
10^{-12}	13.6	12.8	13.9	10.9	11.3	11.3	77	87	387	28	34	—	—	—	—
10^{-16}	>14	>14	>14	>14	>14	>14	—	—	—	—	—	—	—	—	—

L_2-error in these regions may be roughly ϵ^{-1} times greater than the error in energy norm. This severely retards our ability to recognize when convergence has been achieved.

We have looked at two variations of the simplest non-compact case, (ii-a), which is nevertheless extremely hard to solve accurately for the large isolated region. In the case denoted "Dir-Neu", an inflow boundary condition is applied to the bottom edge, while the top edge has prescribed pressure. When ϵ becomes small, this is an unstable (or sensitive) problem: A flux of the order of ϵ induces a pressure of the order of unity. The other case, denoted "Dir-Dir", is the same physical problem but mirrored along the bottom so that it extends to twice the size in the vertical direction, with one barrier near the top and one barrier near the bottom. Prescribed pressure is applied to both the top and the bottom boundaries, such that a flux equivalent to the Dir-Neu case is created. In both cases we compare the solution found by a direct solver and the solution found by CG to the analytical solution. BiCGStab is not included in the table, but was found to behave substantially similar to CG.

For the Dir-Neu case, we set $p = 0$ at the top boundary and $\Lambda\nabla p \cdot n = 10\epsilon$ at the bottom boundary. The analytical solution is then

$$\hat{p}_{\mathrm{N}}(z;\epsilon) = \begin{cases} 1 + (9 - 10z)\epsilon & 0.0 \le z \le 0.8, \\ \epsilon + (9 - 10z) & 0.8 < z \le 0.9, \\ (10 - 10z)\epsilon & 0.9 < z \le 1.0. \end{cases} \tag{2}$$

The Dir-Dir case has prescribed pressure $p = 2\hat{p}_{\mathrm{N}}(0;\epsilon)$ at the bottom boundary, and has the analytical solution

$$\hat{p}_{\mathrm{D}}(z;\epsilon) = \begin{cases} \hat{p}_{\mathrm{N}}(z;\epsilon) & z \ge 0, \\ 2\hat{p}_{\mathrm{N}}(0;\epsilon) - \hat{p}_{\mathrm{N}}(-z;\epsilon) & z < 0. \end{cases} \tag{3}$$

Table 3 summarizes the present experiments. The direct solution degrades as we approach 10^{-16}. So does the CG solution, but not worse than the direct one. In the Dir-Neu case, the residual convergence is highly dependent on the initial iterate. The error, however, is reduced at a similar rate for both choices of initial iterate.

Table 3: The error compared to the analytical solution, for a direct solver and for an AMG-preconditioned Conjugate Gradient solver. For the latter, random and zero initial iterate x_0 are considered. Logarithm of the error is shown, along with the number of iterations of the iterative solver. Non-convergence is denoted as "—".

	Dir-Dir					Dir-Neu				
	LU	CG, $x_0 = 0$		CG, $x_0 = \text{rnd}$		LU	CG, $x_0 = 0$		CG, $x_0 = \text{rnd}$	
ϵ	$\log\|e\|_{L_2}$	$\log\|e\|_{L_2}$	it.	$\log\|e\|_{L_2}$	it.	$\log\|e\|_{L_2}$	$\log\|e\|_{L_2}$	it.	$\log\|e\|_{L_2}$	it.
10^{-4}	-11.0	-10.9	13	-11.1	14	-10.9	-10.5	11	-11.1	13
10^{-8}	-7.1	-6.9	16	-7.0	16	-6.9	—	—	-7.2	15
10^{-12}	-3.5	-3.0	20	-2.9	20	-2.9	—	—	-2.9	18
10^{-13}	-2.0	-1.9	21	-2.0	21	-1.9	—	—	-2.0	19
10^{-14}	-2.1	$-.76$	22	$-.06$	11	$-.92$	—	—	$-.96$	20
10^{-15}	2.6	$-.07$	10	$-.06$	11	1.7	—	—	$-.05$	10

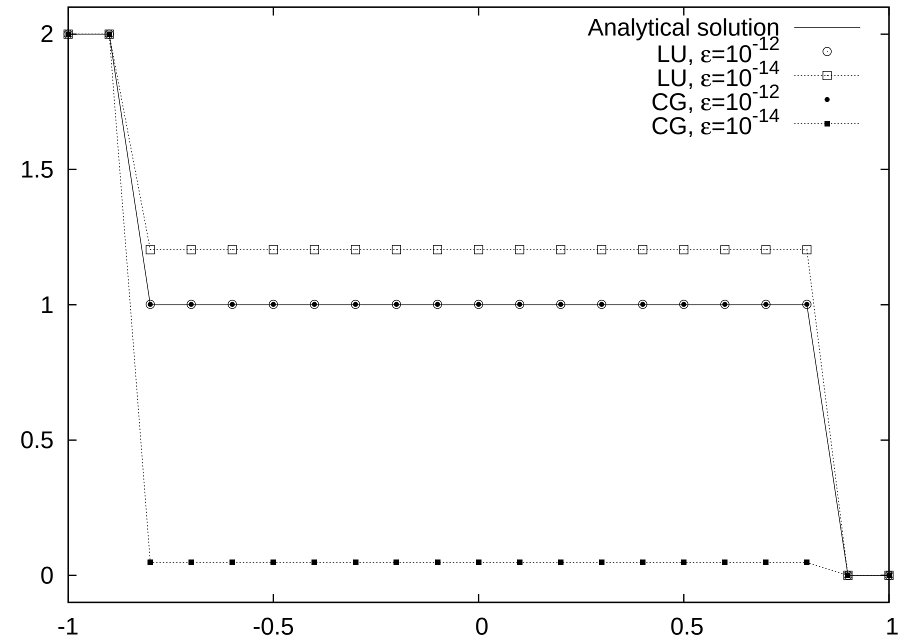


Figure 4: LU (open symbols) and CG (filled symbols) solutions of the Dir-Dir problem for two values of ϵ. The two analytical solutions (solid line) are indistinguishable. When $\epsilon \lesssim 10^{-12}$, the pressure constant in the central region in effect becomes undetermined.

We conclude that for problems with compact low-permeable regions, AMG-preconditioned conjugate gradient-type methods are effective. The pressure inside the low-permeable regions may or may not be accurate, but in either case the solver converges fine and any error in the pressure inside Ω_ϵ does not impact the flow materially. For non-compact regions, we have an unstable system which may have a large error in the embedded nearly "pure Neumann" high-permeable regions (see for example Figure 4). Again, this has only a negligible impact on flow. An example of this is shown in Figure 5, where the pressure distribution inside the (non-compact) low-permeable region does not change the flow at all. The main problem is then the impact it has on the convergence of the iterative solver. One way to ameliorate the effect it has on convergence, if we assert that the exact pressure solution is ill-defined and of little interest, is to impose a solution on it by way of regularization.

One such scheme applies a small modification to Equation (1), in order to stabilize it when it would otherwise be ill-conditioned. The modified equation is

$$\nabla \cdot (\Lambda \nabla p) - \delta p = q, \tag{4}$$

where δ is a small parameter, possibly varying in space. The effect of this modification is to impose an additional restriction on an otherwise underdetermined problem, pulling it towards the lowest-norm solution. We let δ be 10^{-5} in the isolated regions $\Omega_{i>0}$ and 0 outside, and look at the eigenvalue distribution of the preconditioned system in Table 4. While the condition number of A is still unbounded for case (b), the preconditioner is now effective in bounding the condition number of the problem, although at a rather high value (on the order of δ^{-1}). Comparing with Table 2, we see that this stabilizing term is about equivalent, in terms of condition number, to limiting ϵ to 10^{-5}–10^{-7}, depending on the number of isolated domains. However, we see that BiCGStab still struggles in case (c), when there are multiple extreme eigenvalues, like we observed previously in Table 2. By increasing δ we can make BiCGStab converge, at the cost of further loss of accuracy.

The two techniques — ϵ-limiting and regularization — have fundamentally different goals: With the former, we allow some extra flow in order to stabilize (and thus get a reasonably accurate approximation of) the pressure; with the latter, we sacrifice the accuracy in pressure inside the low-permeable regions in order not to disturb the flow. A more drastic way of achieving accurate flow at the expense of pressure accuracy is to make the isolated high-permeable regions *less* permeable, thus entering into the compact problem domain, which we have seen is handled very well by the AMG preconditioner. Which approach is preferable depends on the application.

Miscellaneous results

We have assumed convergence to be independent of the resolution n, owing to known properties of the multigrid preconditioner. These properties may not hold, however, when $\mathrm{M} \approx \nabla \cdot \Lambda \nabla$, which we use here in order to remove the ϵ-dependence of the condition number κ, instead of the more well-studied Laplace operator $\mathrm{M}_\Delta \approx \nabla^2$. There are also questions of the influence of anisotropy, triangular elements, 3D problems, et cetera. While the scope of this paper does not allow a full investigation of these questions, we have gathered a number of suggestive results in Table 5. The main impression is that these variants, while they may increase the condition number and iteration count, do so in a limited way. (Note that the jump in condition number in case V is explained by the

Table 4: Results for the regularized equation, where $\delta = 10^{-5}$ is applied to the isolated permeable region(s) $\Omega_{i>0}$.

	$\log\kappa(\mathrm{A})$			$\log\kappa(\mathrm{M}^{-1}\mathrm{A})$			CG it.			BCGS it.		
ϵ	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
10^{-1}	2.7	2.7	2.9	.20	.44	.06	15	17	19	9	11	13
10^{-2}	3.6	3.0	3.9	.94	1.2	.42	17	22	31	10	14	25
10^{-4}	5.6	4.8	5.9	3.9	3.2	3.3	19	24	49	12	17	59
10^{-8}	7.7	8.5	7.9	5.0	6.4	6.5	31	37	81	15	21	—
10^{-12}	7.7	12.5	7.9	5.0	6.4	6.6	56	29	94	14	20	—
10^{-16}	7.7	>14	7.9	5.0	6.4	6.6	56	30	95	14	21	—
10^{-20}	7.7	>14	7.9	5.0	6.4	6.6	47	37	100	16	19	—

Table 5: Results for miscellaneous variations of the compact case (i). I—the base case, II—triangular elements, III—anisotropic in low-permeable region, $\Lambda_x = 100\Lambda_y$, IV—anisotropic in high-permeable region, $\Lambda_x = 100\Lambda_y$, V—biquadratic elements with 2× resolution, VI—5× resolution, VII—3D. In all cases, $\epsilon = 10^{-12}$.

	$\log\kappa(\mathrm{A})$			$\log\kappa(\mathrm{M}^{-1}\mathrm{A})$			CG it.			BCGS it.		
	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)	(a)	(b)	(c)
I	3.1	13.9	3.8	.49	.14	1.2	16	17	26	11	12	20
II	3.3	13.2	4.1	.89	.41	1.4	19	18	33	19	11	27
III	3.1	12.1	3.8	.49	.80	1.3	17	28	26	10	20	21
IV	4.8	>14	4.9	2.0	1.8	2.1	54	50	63	40	35	50
V	12.7	13.7	12.5	1.1	1.0	1.4	30	35	28	18	22	19
VI	13.1	>14	>14	.62	.33	.33	18	22	33	12	13	20
VII	4.1	14.0	13.1	.50	.17	1.3	17	18	34	10	12	23

additional nodes inside the thin layers in (a) and (c), making these cases more similar to case (b).) The only exception is when the high-permeable material is made anisotropic. However, this happens already at $\epsilon = 1$ (not shown in the table), i.e., even when the permeability is constant. Anisotropy is a known difficult case for multilevel methods, which may be treated by techniques such as semicoarsening [10] or line-smoothing [2]. Hence, the main conclusions from previous sections remain unchanged, although the exact limits of convergence may change.

Finally, we look at the result of setting the initial iterate to a harmonic function inside the low-permeable regions. In [1], Cai et al. argue that a harmonic function inside the low-permeable region stays in the subspace where it does not see the effect of ϵ, when an inverse Laplace operator is used as preconditioner, and hence that convergence is independent of ϵ as long as the starting iterate is chosen correctly. Table 6 shows the convergence results for three different choices of starting iterate, using unpreconditioned CG, CG preconditioned with an inexact AMG inverse of the Laplace operator, and the AMG approximation of the actual operator. We note that the unpreconditioned case is insensitive to whether the starting iterate is random or constant, except in the case where that constant is 0 (which is the correct solution). The AMG approximation of the

Table 6: Number of iterations to solve $Ax = 0$ to tolerance $\|e\|_{L_2} < 10^{-10}$ for case (i-b). Unpreconditioned CG, inexact Laplace-preconditioned CG, and CG preconditioned by the inexact operator A are tested with different starting iterates x_0. These are: Random everywhere; random in Ω_0 and zero in Ω_ϵ (including boundary); and random on Ω_0 and a non-zero constant in Ω_ϵ.

	None			AMG/Laplace			AMG		
ϵ	rnd	rnd|0	rnd|C	rnd	rnd|0	rnd|C	rnd	rnd|0	rnd|C
10^{-1}	81	80	81	25	22	22	8	8	8
10^{-2}	132	125	129	34	31	31	9	8	9
10^{-4}	227	185	227	49	43	43	9	9	9
10^{-8}	376	267	373	77	65	65	10	10	10
10^{-12}	552	51	551	106	91	91	12	11	12
10^{-16}	727	51	725	132	114	112	11	11	13
10^{-20}	901	52	899	161	138	138	10	12	13

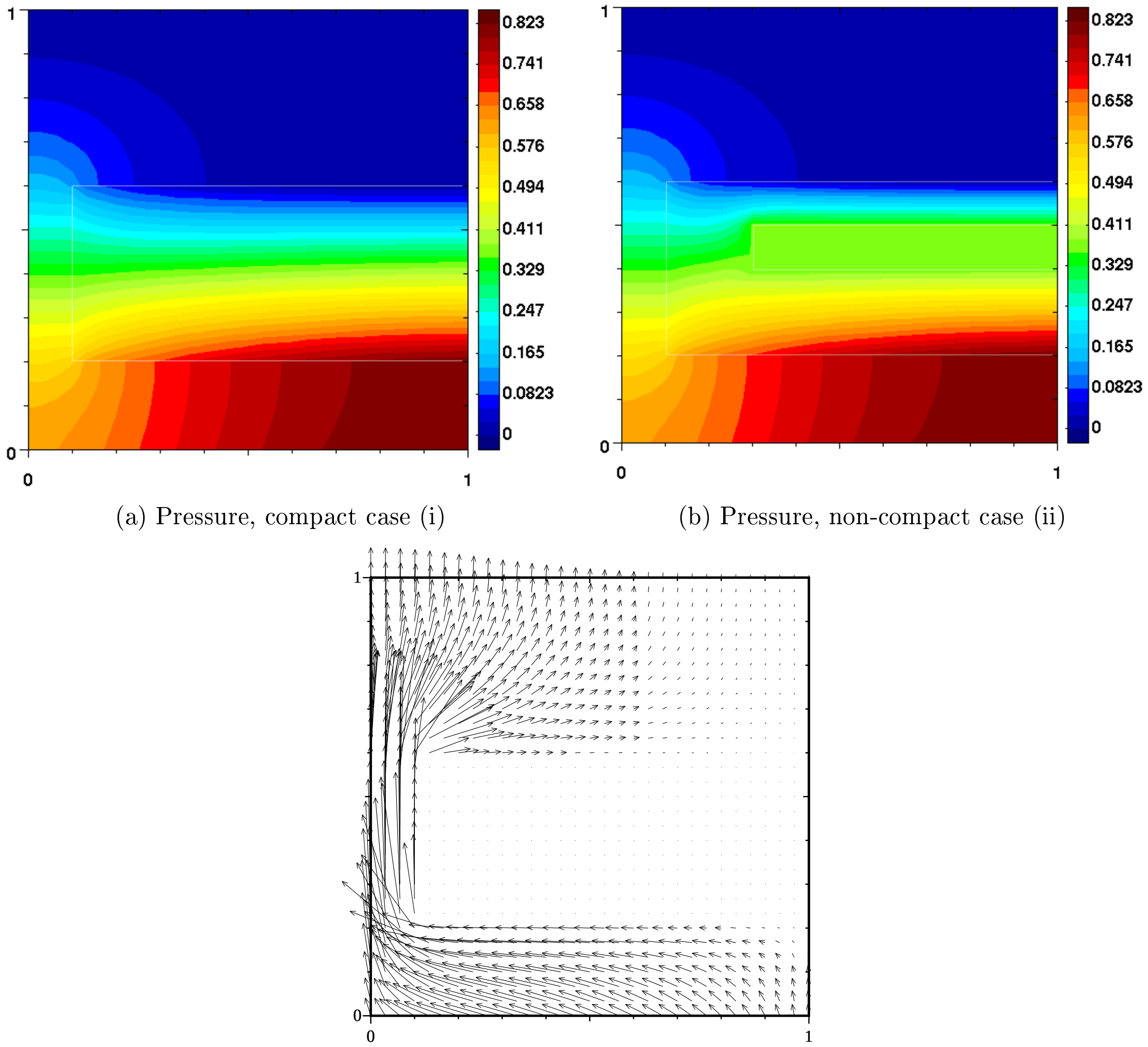
actual operator, meanwhile, converges quickly whatever starting iterate is used. As for the inexact Laplace operator, we see that it converges faster when a harmonic function is used, but it is *not* fully independent of ϵ. This may be because the AMG Laplace inverse is not a good enough approximation to keep the iterates confined to the desired subspace.

Conclusion

We have investigated the performance of conjugate gradient-type iterative solvers, preconditioned by an algebraic multigrid method, on the pressure equation with highly discontinuous permeability. Our results show that this combination is very effective when dealing with the common case of compact low-permeable subregions, relieving the need to perform complicated domain splitting operations to deal with such regions numerically. Furthermore, we show results for a class of problems which can not be handled by the domain splitting technique, namely that of high-permeable regions which are isolated from the Dirichlet boundary by one or more low-permeable regions. This class of problems causes significant problems, both in accuracy and in convergence, as the permeability jumps approach the limits of machine precision. We have briefly discussed different stabilization schemes and compared their various trade-offs. Depending on whether we are primarily interested in accuracy in the flow solution or in the pressure solution, different stabilization techniques are needed.

The findings in this paper are important for efficient numerical solution of large-scale porous media flow problems, in particular because the tested challenging geometric configurations mimic features that are often encountered in real-world geological applications.

Acknowledgements The authors would like to thank Kent-Andre Mardal, who has been of invaluable assistance when dealing with the subtler issues of iterative solver convergence. We also thank StatoilHydro for their support, both financial and in access to their geological models and expertise. This work is supported by a Center of Excellence grant from the Norwegian Research Council to Center for Biomedical Computing at Simula Research Laboratory.



(a) Pressure, compact case (i)

(b) Pressure, non-compact case (ii)

(c) Flow, both cases

Figure 5: Pressure and flow in and around a very nearly impermeable obstacle. The calculated solution of the thick-obstacle case, (i-b) and (ii-b) in figures 1 and 2, for $\epsilon = 10^{-20}$, is shown. The flow is calculated from the relation $v = -\Lambda \nabla p$, and is indistinguishable for the two cases.

References

[1] X.Cai, B. F.Nielsen and A.Tveito An analysis of a preconditioner for the discretized pressure equation arising in reservoir simulation *IMA Journal of Numerical Analysis*, **vol.19**(2), 291, 1999.

[2] E.Chow, R. D.Falgout, J. J.Hu, R.Tuminaro and U. M.Yang A survey of parallelization techniques for multigrid solvers *Parallel Processing for Scientific Computing*, page 179, 2006.

[3] J. W.Demmel, S. C.Eisenstat, J. R.Gilbert, X. S.Li and J. W. H.Liu A supernodal approach to sparse partial pivoting *SIAM J. Matrix Analysis and Applications*, **vol.20**(3), 720–755, 1999.

[4] J. W.Eaton *GNU Octave Manual* Network Theory Limited, 2002.

[5] M. W.Gee, C. M.Siefert, J. J.Hu, R. S.Tuminaro and M. G.Sala ML 5.0 smoothed aggregation user's guide Technical Report SAND2006-2649, Sandia National Laboratories, 2006.

[6] W.Hackbush *Iterative Solution of Large Sparse Systems of Equations* Springer-Verlag, 1995.

[7] H. P.Langtangen *Computational Partial Differential Equations: Numerical Methods and Diffpack Programming* Springer, 2nd edition, 2003.

[8] B. F.Nielsen and A.Tveito On the approximation of the solution of the pressure equation by changing the domain *SIAM Journal on Applied Mathematics*, pages 15–33, 1997.

[9] D. W.Peaceman *Fundamentals of numerical reservoir simulation* Elsevier Science, 1991.

[10] S.Schaffer A semicoarsening multigrid method for elliptic partial differential equations with highly discontinuous and anisotropic coefficients *SIAM Journal on Scientific Computing*, **vol.20**, 228, 1998.