On a complex Shifted-Laplace preconditioner for high wavenumber heterogeneous Helmholtz problems *

Y.A. Erlangga†, C. Vuik, C.W. Oosterlee

Department of Applied Mathematical Analysis, Delft University of Technology, Mekelweg 4, 2628 CD, Delft, The Netherlands

Abstract

In [3] we propose a concept for a preconditioner to improve the convergence rate of Krylov subspace methods applied to the Helmholtz equation. The preconditioner is called the Shifted-Laplace preconditioner. In this paper, we present some numerical experiments using different ways to solve the preconditioner for heterogeneous, high wavenumber Helmholtz problems. Although the preconditioner is solved iteratively, the convergence is much faster than that of a standard ILU preconditioner based on the original matrix A.

1 Introduction

We are concerned with solutions of the following boundary value problem

$$\mathcal{L}u \equiv \left(\partial_{xx} + \partial_{yy} + k^2(x,y)\right) u = f \text{ in } \Omega \in \mathbb{R}^2, \quad (1)$$

$$\lim_{r\to\infty} \sqrt{r}\left(\frac{\partial u}{\partial n} - iku\right) = 0 \text{ on } \Gamma = \partial\Omega, \quad (2)$$

– the Helmholtz problem – which governs wave propagations/scattering in a 2-d heterogeneous medium, with $k = k(x,y)$ is the wavenumber. This kind of problem arises in many applications, e.g. aeronautics, geophysics and electromagnetics. In particular we are interested in solutions of geophysical problems and solve the system (1)–(2) numerically. In geophysical problems, the boundary of the domain is usually simple (e.g. a rectangular shape in 2-d). In such

*This research is financially supported by the *Dutch Ministry of Economic Affairs* under the project BTS01044 "Rigorous modelling of 3D wave propagation in inhomogeneous media for geophysical and optical problems"

†PhD student. E-mail address: y.a.erlangga@math.tudelft.nl

a domain, finite differencing is easy to be done and accurate solutions can be obtained by using a high order scheme.

Our primary iterative solution method is the Krylov subspace method applied to the discretized formulation of (1)–(2). The algorithm is in general easy to implement, requiring only matrix/vector multiplication(s) and some vector updates.

In the original setting, a Krylov subspace method usually faces slow convergence when it is used to solve the Helmholtz equation. There are at least two reasons for this slow convergence: (1) the indefiniteness of the system to solve, and (2) an ill-condition system characterized by an extremely large condition number. A standard remedy to enhance the convergence is by incorporating a preconditioner so that the preconditioned system is better conditioned than the original one. Finding an effective preconditioner for the Helmholtz equation, however, is not an easy task. Standard preconditioners, e.g. based on ILU or approximate inverse (AI), do not seem to work quite effectively for high wavenumbers, see, e.g., the recent study in [5]. For this type of preconditioner, break down seems to occur even for small wavenumbers. As geophysical applications demand high wavenumbers, this type of preconditioner may become inattractive.

Several authors have proposed special preconditioners for the Helmholtz problems, e.g. [2, 8, 10]. In [3], we propose a preconditioner, the so-called Shifted-Laplace, which can be considered as a generalization of the preconditioners introduced in [1, 6]. The preconditioner is built based on the operator

$$\partial_{xx} + \partial_{yy} - (\alpha + i\beta)k^2, \quad \alpha, \beta \geq 0. \tag{3}$$

For fastest convergence, we found that $\alpha = 0$ and $\beta = 1$ results in an optimal condition number within this class of preconditioners. Furthermore, our numerical examples do not show any breakdown, see [3] for more details.

In [3] this fastest convergence was, however, achieved with exact inversion of the preconditioner. This is a difficult task and time consuming. (Solving directly the original problem using direct methods is much cheaper than solving exactly the preconditioner several times during iteration).

The main goal of this paper is to compare *approximate* methods for solving the Shifted-Laplace preconditioner. We focus on two classes of methods for this purpose: incomplete LU (ILU) and multigrid. As the discretization of (3) leads to a complex, symmetric positive definite system, ILU and multigrid should be the methods of choice. We expect to gain much in computational time.

This paper is organized as follows. In Section 2 we discuss briefly the fourth order discretization used. Krylov subspace methods and the preconditioner are discussed in Section 3. In Section 4, methods to solve the preconditioner are explained. We present numerical results in Section 5.

2 Discretization

We first discretize (1)–(2). For this purpose we use the fourth order finite difference discretization derived from the Padé approximation of the differential operator. For example, we can write the Padé approximation of ∂_{xx} as

$$\partial_{xx} \approx \frac{D_{xx}}{1 + (h^2/12)D_{xx}}, \tag{4}$$

and similar for ∂_{yy}. Substituting (4) into (1) yields

$$-\left(\frac{D_{xx}}{1 + (h^2/12)D_{xx}}u_{i,j} + \frac{D_{xx}}{1 + (h^2/12)D_{xx}} + k^2\right) u_{i,j} = f_{i,j}. \tag{5}$$

To minimize dispersion and anisotropy, generalization is proposed in [4] by introducing a coefficient γ:

$$\begin{aligned} - & \left(D_{xx} + D_{yy} + \frac{h^2}{6}D_{xx}D_{yy}\right) u_{i,j} \\ - & k^2\left(1 + \frac{h^2}{12}(D_{xx} + D_{yy}) + \gamma\frac{h^2}{144}D_{xx}D_{yy}\right) u_{i,j} = \\ - & \left(1 + \frac{h^2}{12}(D_{xx} + D_{yy}) + \gamma\frac{h^4}{144}D_{xx}D_{yy}\right) f_{i,j}. \end{aligned} \tag{6}$$

The value γ is chosen such that the dispersion or anisotropy are minimal. In (6), the operator $D_{xx}D_{yy}$ reads

$$D_{xx}D_{yy} = \frac{1}{h^4}\left(\sigma_c - 2\sigma_s + 4\phi_{i,j}\right) \tag{7}$$

with

$$\sigma_c = \phi_{i-1,j-1} + \phi_{i-1,j+1} + \phi_{i+1,j-1} + \phi_{i+1,j+1}, \tag{8}$$

the sum of the corner points, and

$$\sigma_s = \phi_{i,j-1} + \phi_{i,j+1} + \phi_{i+1,j} + \phi_{i,j-1}, \tag{9}$$

the sum of mid-side points. This finally gives a compact, nine-point stencil with the high order property. Analysis in [4] shows that setting $\gamma = \frac{2}{5}$ or $\gamma = \frac{14}{5}$ results in schemes with minimal dispersion or anisotropy, respectively. We use the fourth order scheme to reduce the number of grid points needed per wavelength to approximate high wavenumber solutions satisfactorily.

The resulting linear system

$$Ax = b, \;\; A \in \mathbb{C}^{N\times N}, \;\; x, b \in \mathbb{C}^{n\times 1} \tag{10}$$

is in general indefinite for sufficiently large k.

3 Krylov subspace methods

3.1 General

In Krylov subspace methods we construct iterants in the subspace

$$\mathcal{K}^j(A, r_0) = \text{span}\{r_0, Ar_0, A^2 r_0, \ldots, A^{j-1} r_0\}, \tag{11}$$

where $\mathcal{K}^j(A, r_0)$ is the j-th Krylov subspace associated with A and r_0, $r_0 = b - Ax_0$ is the initial residual. There are many ways to construct the Krylov subspace, which leads to different algorithms. For a complete survey, see [11].

Since we are interested in a solution method which is less stringent in the choice of preconditioners, we use algorithms for unsymmetric matrices instead. Some advanced algorithms are Bi-CGSTAB [13] and GMRES [12]. We choose Bi-CGSTAB which, despite requiring two matrix-vector multiplications, can preserve a constant amount of work and storage per iteration.

If the linear system is badly conditioned, indicated by a large condition number κ, Krylov subspace methods typically converge very slowly and may break down in some cases. As a remedy one preconditions the linear system, so that the preconditioned linear system becomes better-conditioned. By preconditioning one solves the equivalent linear system

$$M_1^{-1} A M_2^{-1} \tilde{x} = M_1^{-1} b, \;\; \tilde{x} = M_2 x, \tag{12}$$

where $M = M_1 M_2$ is the preconditioning matrix. To reduce the number of preconditioner solves, in our applications only right preconditioning is used.

As an addition to the condition that the preconditioned system should be better-conditioned than the original one, the preconditioner must also be easy to solve. For example, a preconditioner in the form of an incomplete LU factorization is easy to solve using backward-forward substitution and this solution process requires only $\mathcal{O}(N)$ operations.

3.2 Preconditioner

We distinguish two classes of preconditioners for the Helmholtz equation. The first one is classified as "matrix-based". Examples of this class are incomplete LU (ILU) and approximate inverse (AI), applied to matrix A resulting from the Helmholtz discretization. ILU preconditioners, however, may require extra storage due to fill-in and this storage requirement can exceed that for storing A. A recent study on the application of ILU preconditioners for the Helmholtz equation is given in [5]. The results, however, are not satisfactorily for high wavenumbers, indicated by a stagnation of the methods.

The second class is the so-called "operator based" preconditioners. In this class, the preconditioner is determined on the continuous level (and not in the matrix level). Within this class are Analytic ILU (or AILU) [2], the Separation of Variables preconditioner [10], and the well-known Laplace preconditioner [1]. Preconditioning using the Laplace operator proposed in [1] is further generalized

for the Helmholtz equation in [6] by adding a positive zero-th order term with the constant k^2, which results in a positive definite preconditioning operator. In [3] we propose a further generalization in this class by adding an imaginary shift to the zero-th order term for improved convergence. We call this class of the preconditioners " Shifted-Laplace preconditioners". Our variant takes the form

$$\mathcal{M} = \partial_{xx}u + \partial_{yy}u - ik^2u, \tag{13}$$

where $i^2 = -1$, the complex identity. A first motivation for this preconditioner can be given by means of a spectral analysis of the homogeneous 1-d case.

Consider the homogeneous 1-d Helmholtz equation in $\Omega = (0, 1)$ with Dirichlet boundary at both ends. For the preconditioner, we use the 1-d version of (13). The eigenvalue problem related to the preconditioned system can be expressed in continuous form as

$$\left(\frac{d^2}{dx^2} + k^2\right) u_v = \lambda \left(\frac{d^2}{dx^2} - ik^2\right) u_v, \tag{14}$$

where u_v is an eigenfunction, and λ is the eigenvalue. After some arithmetic we find that

$$\lambda_n = \frac{k_n^2 - k^2}{k_n^2 + ik^2}, \;\; k_n = n\pi, n \in \mathbb{Z}. \tag{15}$$

If one considers the normal equations representation of (13), the eigenvalue becomes

$$|\lambda_n|^2 = \frac{(k_n^2 - k^2)^2}{k_n^4 + k^4}. \tag{16}$$

For an indefinite system, the latter is easier to be used for the convergence analysis than (15).

Assuming a finite k, one finds that

$$\lim_{n\to\infty} |\lambda_n|^2 = \lim_{n\to 0} |\lambda_n|^2 = 1. \tag{17}$$

Furthemore, for finite k_n,

$$\lim_{k\to\infty} |\lambda_n|^2 = 1. \tag{18}$$

Therefore, in these kinds of situations the eigenvalues are bounded above by one. Consider the condition that for the smallest eigenvalue $|\lambda_n|^2 = k^2 + \epsilon$ holds. Subsituting this relation into (15) and omitting terms with ϵ^2 one finds

$$|\lambda_{\min}|^2 = 2\left(\frac{\epsilon}{k}\right)^2. \tag{19}$$

Hence, the condition number is $\kappa = \frac{1}{2}(k/\epsilon)^2$. A similar analysis also holds for the discrete case and in higher dimensions, see [3].

We summarize some spectral properties of the preconditioned system (i.e. $\mathcal{M}^{-1}A$), related to the normal equations formulation: (1) the spectrum is bounded above by one, and (2) the spectrum has a lower bound which is of order $\mathcal{O}(\epsilon^2/k^2)$. Also we can determine from the analysis that the convergence becomes slower as k increases.

4 Solution method

The preconditioner is promising from theoretical point of view. As we have shown in [3], an exact inverse of this preconditioner can effectively accelerate the convergence for heterogeneous, high wavenumber Helmholtz problems. We have, however, use direct methods to solve the preconditioner. As the direct methods are too expensive, especially in 3-d, this version of the preconditioner seems impracticable. In this paper, we give some numerical results verifying that approximately inverting the preconditioner by cheaper iteration still results in fast convergence and w.r.t. computational time is much more acceptable.

It is important to note that, whereas matrix A has been set up with a high order discretization, for M it appears to be sufficient to keep a second order discretization of (13). This is beneficial for the iterative solutions.

4.1 Incomplete LU

The first method we use to approximate the inverse of the preconditioner is the ILU factorization. In this case, we construct the LU factors based on M and *not* A. This is somewhat uncommon as one usually tries to build LU as an approximation to A. As M is not an approximation of A, but of (13), neither is LU if constructed based on M.

An ILU factorization is obtained from an approximate Gaussian elimination and, by nature, introduces fill-in. This fill-in will require extra storage. The simplest ILU factorization is ILU(0), obtained from standard LU factorization by dropping all fill-in elements. In many situations, however, ILU(0) is not sufficiently accurate. A general ILU(p) factorization is obtained by the same procedures but by dropping the fill-in elements according to the "levels" in the elimination process.

In this study, we limit ourselves to ILU(0) and ILU(1). Furthermore, since the preconditioning matrix is built by using finite differencing, the LU factors can be constructed based on the stencil [11] and not on the Gaussian elimination. This procedure offers a cheaper initialization process.

4.2 Multigrid

One advantage of the Shifted-Laplace preconditioner (13) is that the preconditioning matrix is complex, symmetric positive definite. For this type of matrices, iterative solution using (algebraic) multigrid has been reported in [7]. In our case, however, geometric multigrid is sufficient.

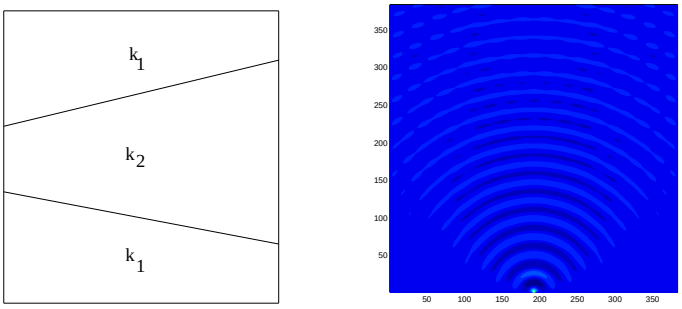

Figure 1: *Domain with transparent wedge (left) and solution ($\Re(u)$) for $k = 100$ (right)*

For multigrid, we use one of the multigrid algorithms discussed in [9], called $MG1$. Prolongation operator P is bi-linear interpolation. For the restriction operator, the full weighting, i.e. transpose of the prologation is used. Coarse grid discretization is done with Galerkin coarsening. For the smoother, red-black Gauss-Seidel is used. It is able to handle, however, a complex main diagonal. For solving the linear system, the V-cycle with one pre- and post-smoothing is used (V(1,1)). In our current implementation, there are no special enhancements incorporated to make the multigrid algorithm especially efficient for complex-valued matrices.

5 Numerical experiments

We present numerical results for the following two problems.

Problem 1. We first consider a very simple test case which mimics a geophysical problem in homogeneous medium. The domain is a unit square $\Omega = (0,1) \times (0,1)$. The wavenumber varies between 10 and 100,200. In real situations, the last two wavenumbers can be considered as problems with "medium frequencies". A source term is included at the center of the upper surface.

Problem 2. In this case, a domain with a transparent wedge (see Figure 1) is considered for various wavenumbers with $k_2 = 2k_1$. As in Problem 1, a source is generated at the center of upper surface. The wavenumbers here are as in Problem 1. A picture of the solution for this problem is also presented in Figure 1.

All computations are performed on an Intel PIII 550 Mhz processor with 256 Mb of RAM. The code is compiled with GNU F77. To limit the storage, the matrix A is not stored and is recomputed whenever necessary. For ILU(0), only the diagonal entries are stored whereas the remaining parts are recomputed whenever needed. For ILU(1), the diagonal and two off-diagonal entries are

stored. As the termination criteria, we use $\|r/r_0\|_2 = \|(b - Ax)/(b - Ax_0)\|_2 \leq 10^{-6}$.

Numerical results for Problem 1 are shown in Table 1 with, both, the number of iterations and the (total) CPU time. Approximate preconditioners (either based on A or M) improve the convergence and computational time considerably. Preconditioning based on M, however, results in better numerical performance as compared to a preconditioner based on A. With multigrid applied to M the performance is even better. The number of iterations is reduced by almost factor of 40 as compared to the unpreconditioned case and this factor tends to become larger for higher k (we did not compute the solution for the unpreconditioned cases with $k = 100$ and 200). This indicates the effectiveness of the Shifted-Laplace preconditioner. Furthermore, the Shifted-Laplace preconditioner handled with one multigrid iteration also reduces CPU time significantly. However, compared to the simple-to-construct ILU(M,1) (meaning ILU(1) based on M) this CPU time is still comparable. This can be understood from the large (complex) diagonal elements in M for high wavenumbers.

From Table 1 it is also seen that for M there does not seem to be a breakdown. The increase in number of iterations for large problems behaves well, with approximately a linear increase with respect to k. Also, the convergence becomes faster as M is solved more accurately.

Table 1: *Numerical results from Problem 1. Number of iterations and (in brackets) CPU time in sec. are shown for various k.*

k	10	20	30	40	50	100
grid	32^2	64^2	96^2	128^2	192^2	384^2
No-Prec	154(0.4)	1028(12)	1922(67)	2715(164)	4901(628)	–
ILU(A,0)	69(0.25)	316(7.9)	473(25)	break	1165(227)	–
ILU(A,1)	37(0.21)	126(3.6)	215(13)	333(35)	551(125)	–
ILU(M,0)	69(0.31)	240(6.1)	391(21)	594(53)	899(176)	1837(1440)
ILU(M,1)	37(0.17)	117(3.3)	195(12)	295(30)	489(110)	838(746)
MG(V(1,1))	18(0.22)	40(2.8)	62(10)	84(25)	107(75)	206(595)

– not computed

Figure 2 shows a comparison of the convergence history for Problem 1 for $k = 30$. This convergence behavior is representative for most of our computations at different wavenumbers for Problems 1 and 2.

Table 2 shows numerical results for Problem 2. The results are in general analogous to those for Problem 1. As Problem 2 is more difficult to solve, the number of iterations increases. A preconditioner solve using multigrid seems to result in a more robust method than using ILU (the number of iterations with ILU grows much faster than that with multigrid) for this heterogeneous problem. This suggests the importance of a more accurate inversion of M in the case of heterogeneous media. Using ILU based on A in the case of heterogeneous problem does not seem to be attractive.

Actually, we have also solved the $k = 200$ case on a 512^2 grid with the multigrid preconditioner based on M. Problem 1 takes 376 iterations and 1921 seconds; for Problem 2, 553 iterations and 2867 seconds.

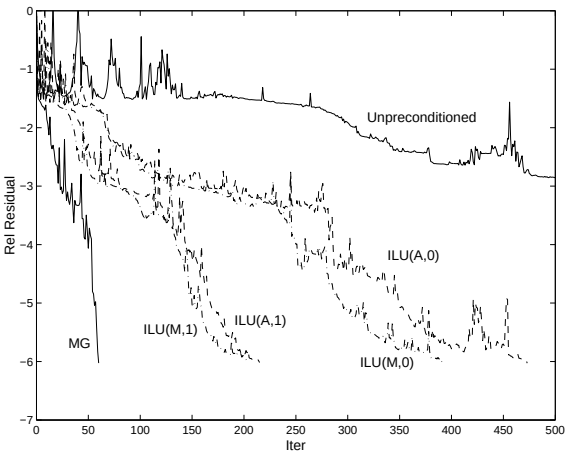

Figure 2: *Convergence history for Problem 1.* $k = 30$

Table 2: *Numerical results from Problem 2. Number of iterations and (in brackets) CPU time in sec. are shown for various* k_2 *and* $k_2 = 2k_1$.

k_2	10	20	30	40	50	100
grid	32^2	64^2	96^2	128^2	192^2	384^2
No-Prec	201(0.56)	1028(12)	3464(125)	5170(316)	–	–
ILU(A,0)	55(0.36)	348(9)	831(48)	1484(131)	2344(498)	–
ILU(A,1)	26(0.14)	126(4)	317(20)	577(62)	894(207)	–
ILU(M,0)	57(0.29)	213(8)	784(44)	1289(122)	2072(451)	–
ILU(M,1)	28(0.28)	116(4)	244(16)	443(48)	763(191)	2021(1875)
MG(V(1,1))	13(0.21)	38(3)	68(12)	94(28)	115(82)	252(850)

– not computed

6 Conclusion

In this paper, Krylov subspace iterative methods applied to the Helmholtz problem have been presented with the complex Shifted-Laplace operator used as the preconditioner. This preconditioner is very effective to enhance the convergence of the Krylov subspace methods for our target problem: high wavenumber, heterogeneous Helmholtz problems. Two methods have been used to solve the preconditioner: ILU and multigrid. For the preconditioner solves, it is concluded from numerical results that multigrid is more effective and efficient. The effectiveness of multigrid becomes more prominent for cases involving heterogeneous media. However, constructing ILU factors from the Shifted-Laplace operator improves the numerical performance as compared to constructing ILU from the original system.

So far, only a standard multigrid solver with a straightforward extension to complex-valued matrices has been used. In a future paper, a more efficient multigrid method for complex-valued matrices will be investigated to further

increase the computational efficiency.

References

[1] A. Bayliss, C.I. Goldstein, E. Turkel, An iterative method for Helmholtz equation, J. Comput. Phy., 49 (1983), pp. 443–457.

[2] M.J. Gander, F. Nataf, AILU for Helmholtz problems: a new preconditioner based on the analytical parabolic factorization, J. Comput. Acoustics, 9(4) (2001), pp. 1499–1506.

[3] Y.A. Erlangga, C. Vuik, C.W. Oosterlee, On a class of preconditioners for the Helmholtz equation, to appear in Appl. Numer. Math.

[4] I. Harari, E. Turkel, Accurate finite difference methods for time-harmonic wave propagation, J. Comput. Phys., 119 (1995), pp. 252–270.

[5] R. Kechroud, A. Soulaimani, Y. Saad, Preconditioning techniques for the solution of the Helmholtz equation by the finite element method, in *2003 Workshop in wave phenomena in physics and engineering: new models, algorithms and Applications, May 18-21, 2003*, eds. Kumar et al., Springer Verlag, Berlin, 2003.

[6] A.L. Laird, Preconditioned iterative solution of the 2D Helmholtz equation, Report, St. Hugh's College, UK,2001.

[7] D. Lahaye, H. De Gersem, S. Vandewalle, K. Hameyer, Algebraic multigrid for complex symmetric systems, IEEE Trans. Magn. 36(4) (2000), pp. 1535–1538.

[8] M.M. Monga Made, Incomplete factorization-based preconditionings for solving the Helmholtz equation, Int. J. Numer. Meth. Engng., 50 (2001), pp. 1077–1101.

[9] C.W. Oosterlee, T. Washio, An evaluation of parallel multigrid as a solver and a preconditioner for singularly perturbed problems, SIAM. J. Sci. Comput., 19(1) (1998), pp. 87–110.

[10] R.E. Plessix, W.A. Mulder, Separation-of-variables as a preconditioner for an iterative Helmholtz solver, Appl. Num. Math., 44 (2003), pp. 385–400.

[11] Y. Saad, Iterative methods for sparse linear systems, 2^{nd} edition, SIAM, 2003.

[12] Y. Saad, M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM J. Sci. Stat. Comput., 7(12) (1986), pp. 856–869.

[13] H.A. van der Vorst, Bi-CGStab: A fast and smoothly converging variant of BiCG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comput., 13(2) (1992), pp. 631–644.