ON A ROBUST ITERATIVE METHOD FOR HETEROGENEOUS HELMHOLTZ PROBLEMS FOR GEOPHYSICS APPLICATIONS

YOGI A. ERLANGGA, CORNELIS VUIK, AND CORNELIS W. OOSTERLEE

Abstract. In this paper, a robust iterative method for the 2D heterogeneous Helmholtz equation is discussed. Two important ingredients of the method are evaluated, namely the Krylov subspace iterative methods and multigrid based preconditioners. For the Krylov subspace methods we evaluate GM-RES and Bi-CGSTAB. The preconditioner used is the complex shifted Laplace preconditioner [*Erlangga, Vuik, Oosterlee, Appl. Numer. Math. 50(2004) 409–425*] which is approximately solved using multigrid. Numerical examples which mimic geophysical applications are presented.

Key Words. Helmholtz equation, Krylov subspace methods, preconditioner, multigrid

1. Introduction

Wave equation migration is becoming increasingly popular in seismic applications. This migration is currently based on a one-way scheme to allow applications in 3D, in which the full wave equation simulation is simply too expensive. It is already known, however, that one-way wave equations do not correctly image steep events and do not accurately predict the amplitudes of the reflections [12].

In 2D, the linear system obtained from the discretization of the full wave equation in the frequency domain can be efficiently solved with a direct solver and a nested dissection ordering [6]. In 3D, the band size of the linear system becomes too large, which makes the direct method inefficient. As an alternative, iterative methods can be used.

Since 3D problems are our final goal, iterative methods become inevitable. In this paper an evaluation of a robust iterative solver for Helmholtz problems is discussed. The solver mainly consists of two important ingredients: Krylov subspace iterative methods, and a preconditioner including multigrid to accelerate the Krylov subspace iterations.

Krylov subspace methods are chosen because the methods are efficient in terms of memory requirement as compared to direct solvers. Multigrid is used as preconditioner for the Krylov subspace methods. In our applications, however, multigrid is not directly applied to the Helmholtz equation. As already pointed out in [3], high wavenumber problems related to the Helmholtz equation raise difficulties for multigrid in both error smoothing and coarse grid correction, the two main principles of multigrid. Instead, we use multigrid on a Helmholtz-like preconditioner that multigrid can handle it easily. In particular, we consider a Helmholtz operator

with a complex shift. An operator-based preconditioner for the Helmholtz equation is first proposed by Bayliss et. al [1] in the early eighties and solved with multigrid in [8]. Laird and Giles [10] proposed a real positive definite Helmholtz operator (i.e. the same Helmholtz operator but with sign reverse for the zeroth order term) as the preconditioner. Our preconditioner [5] is a complex version of a Helmholtz operator.

This paper is organized as follows. In §2, the Helmholtz equation and preconditioners for iteratively solving it are discussed. Some properties of the preconditioned linear system are explained in §3. Multigrid is briefly discussed in §5. We present numerical examples and some conclusions in §6 and §7, respectively.

2. Helmholtz equation, preconditioner

For a given source function g, we are interested in the solution of the Helmholtz equation

$$(1) \qquad \mathcal{A}\phi := -\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2}\phi - (1 - \alpha i)k^2\phi = g, \text{ in } \Omega \subset \mathbb{R}^d, d = 1, 2, 3,$$

which governs wave propagations in the frequency domain. Here, $\phi = \phi(x_1, x_2, x_3) \in \mathbb{C}$ is usually the pressure wave, and k, the wavenumber, varies in Ω due to spatial variation of local speed of sound, c. This wavenumber is defined as $k = \omega/c$, where ω is the angular frequency related to the source function g. We call the medium "barely attenuative" if $0 < \alpha \ll 1$. In (1), $i = \sqrt{-1}$, the complex identity.

Boundary conditions on $\Gamma = \partial\Omega$ are usually in the form of absorbing boundary condition. There are several mathematical representations to satisfy this condition. In [4] hierarchical, local boundary conditions are proposed. A perfectly matched layer can also be used to ensure absorbing boundary (see [2]). In this paper we use two types of the hierarchical absorbing boundary conditions: (i) the first order formulation, namely

$$(2) \qquad \mathcal{B}_1\phi := \frac{\partial\phi}{\partial\nu} - ik\phi = 0, \qquad \text{on } \Gamma$$

with ν the outward normal direction to the boundary, and (ii) the second order formulation

$$(3) \qquad \mathcal{B}_2\phi := \frac{\partial\phi}{\partial\nu} - ik\phi - \frac{i}{2k}\frac{\partial^2\phi}{\partial\tau^2} = 0,$$

with τ the tangential direction. The second order absorbing condition is more accurate in handling inclined outgoing waves at the boundary than the first order boundary condition, but it requires careful implementation.

Discretization of (1) using finite differences/elements/volumes leads to an indefinite linear system

$$(4) \qquad \boldsymbol{A}\phi = \boldsymbol{g}$$

for large wavenumbers. We use a 5-point finite difference approximation to (1) and (2) (or (3)). Furthermore, only for sufficiently small k the problem is definite. For definite elliptic problems, preconditioned Krylov subspace methods and multigrid are two examples of good solvers and have been widely used. For the Helmholtz equation, both methods, however, are found to be less effective, or even ineffective, if k is large.

For Krylov subspace methods, the methods usually suffer from slow convergence. In this kind of situation the methods rely on preconditioners. Finding good preconditioners for the Helmholtz equation, however, is not a trivial task. Since $\boldsymbol{A}$

of (4) is not an M-matrix, standard ILU factorization may become unstable and can result in an inaccurate approximation for the discrete Helmholtz equation. A non standard ILU factorization is proposed in [7] where the Helmholtz operator is split using parabolic factorization. For constant k, an impressive computational performance is observed. The approach requires optimization parameters, which are dependent on k. The performance of the preconditioner is very sensitive with respect to these parameters. Similarly, [13] proposes operator splitting based on separation of variables. For constant k, this splitting is exact. This is, however, not the case if we allow heterogeneity in Ω. For such the problems, the Krylov subspace iterations show break down.

Elman et al [3] recently proposed a multigrid based preconditioner for the Helmholtz equation. In their approach a non-standard multigrid algorithm is used, based on a mix of Jacobi-type iteration and GMRES. At the finest and coarsest level, the cheap Jacobi-type iteration is used as smoother, while on intermediate levels GMRES is used to reduce the residual. This multigrid algorithm is then used as the preconditioner for GMRES. This approach results in an impressive numerical performance, but is involved.

We propose the following operator as the *preconditioner* for (1) [5]:

$$(5) \qquad \mathcal{M} := -\sum_{j=1}^{d} \frac{\partial^2}{\partial x_j^2} - (\beta_1 + i\beta_2)\, k^2, \ \beta_1, \beta_2 \in \mathbb{R},$$

which is similar to $\mathcal{A}$. To determine the pair (β_1, β_2), the prerequisite condition is that as a preconditioner (5) is easily solvable. Since we will use multigrid to solve (5) and its effectiveness to solve a definite linear system is well known, we require that the operator (5) to be definite. As a consequence we choose β_1 to be non-positive.

3. h-independent property of the preconditioner

In this section, we derive the h-independent property of the preconditioned Helmholtz linear system. Our analysis is based on the simplification that we replace the boundary condition (2) by a Dirichlet boundary condition on Γ.

For simplicity, we use the following 1D Helmholtz problem with constant k:

$$(6) \qquad -\frac{d^2\phi}{dx^2} - k^2\phi = 0, \qquad 0 < x < 1, \qquad \phi(0) = 1 \text{ and } \phi(1) = 0,$$

and the preconditioner operator

$$(7) \qquad \mathcal{M}_{1d} := -\frac{d^2\phi}{dx^2} - (\beta_1 + i\beta_2)k^2\phi.$$

Spectrum. Using the above-mentioned assumption, we find that eigenvalues of the preconditioned linear system can be expressed as

$$(8) \qquad \lambda_n = \frac{k_n^2 - k^2}{k_n^2 + (\beta_1 + i\beta_2)k^2}, \quad k_n = n\pi, \quad n = 1, 2, \dots.$$

For the conjugate gradient method, we know that the convergence rate is determined by the condition number κ; the smaller the condition number is, the faster

the convergence is. We have the following estimate [5]:

$$(9) \qquad |\lambda|^2_{\max} \;=\; \max\left(1, \frac{1}{\beta_1^2 + \beta_2^2}\right),$$

$$(10) \qquad |\lambda|^2_{\min} \;=\; \frac{4}{(1+\beta_1)^2 + \beta_2^2}\left(\frac{\epsilon}{k}\right)^2, \quad 0 < \epsilon \ll 1$$

$$(11) \qquad \kappa^2 \;=\; \begin{cases} \frac{1}{4}\left(1 + \frac{1+2\beta_1}{\beta_1^2+\beta_2^2}\right)(k/\epsilon)^2, & \beta_1^2 + \beta_2^2 \le 1, \\ \frac{1}{4}\left((1+\beta_1)^2 + \beta_2^2\right)(k/\epsilon)^2, & \beta_1^2 + \beta_2^2 \ge 1. \end{cases}$$

For $\beta_1 \le 0$, we find that κ is minimal if $\beta_1 = 0$ and $\beta_2 = \pm 1$. We obtain, therefore, a purely imaginary shift to the Laplace operator. From this analysis so far, there should be no difference between choosing positive or negative sign of β_2. Setting $\beta_2 = -1$, however, results in a complex, symmetric positive definite (CSPD) matrix which is more favorable from an iterative method point of view.

With values $\beta = (0, \pm 1)$ we can also conclude that the spectrum is bounded above by one, and this upper bound is independent of k. The lower bound of the spectrum is of order $O(1/k)$. This fact may become problematic as k increases; the smallest eigenvalue move closer to the origin, and this may cause slow convergence in the initial stage of the iteration.

h-**independent property.** From the previous section it appears that the convergence is mainly determined by the smallest eigenvalue. We further extend the analysis on the discrete level to see how this small eigenvalue behaves with respect to the grid size h.

For $k = 0$, the Poisson problem, the eigenvalues of (3) are well known: $\mu_j^c = (j\pi)^2, j = 1, 2, \ldots$. Using the standard central difference method on $N + 1$ grid points and uniform grid size $h = 1/N$, the discrete eigenvalues are given by

$$(12) \qquad \mu_j = \frac{4}{h^2}\left(\sin\frac{\pi h j}{2}\right)^2, \qquad j = 1, \ldots, N.$$

If $\hat{j}$ is such that $\frac{\pi h \hat{j}}{2} \ll 1$ using Taylor expansion we find that $|\mu_j - \mu_j^c| = O(h^2)$ for $j \le \hat{j}$. Therefore, if A_L is the Laplacian part of A, the smallest eigenvalues of the continuous problem can be well approximated by the smallest eigenvalues of A_L.

Suppose now that $k \ne 0$ and $k^2 \ne \mu_j^c$ for all j. For the smallest eigenvalues we have

$$(13) \qquad \lim_{h \to 0}\min_j |\mu_j - k^2| = |\mu_m^c - k^2| \ne 0,$$

where $|\mu_m^c - k^2| = \min_j |\mu_j^c - k^2|$. Combining with (10) we have that

$$(14) \qquad \lim_{h \to 0}\lambda_{\min} = \frac{(\mu_m^c - k^2)^2}{2k^4}.$$

Since the maximal eigenvalues are bounded by 1, we conclude that the condition number, and hence the convergence, is independent of h. Only initially that h influences the convergence.

Remark. This result resembles the analysis given by Manteuffel and Parter in [11] for general elliptic equations preconditioned with another elliptic equation. The result there, however, is based on real-valued and definite operator. Even though the same analysis is not provided in this paper, the result above is in the same line with that in [11].
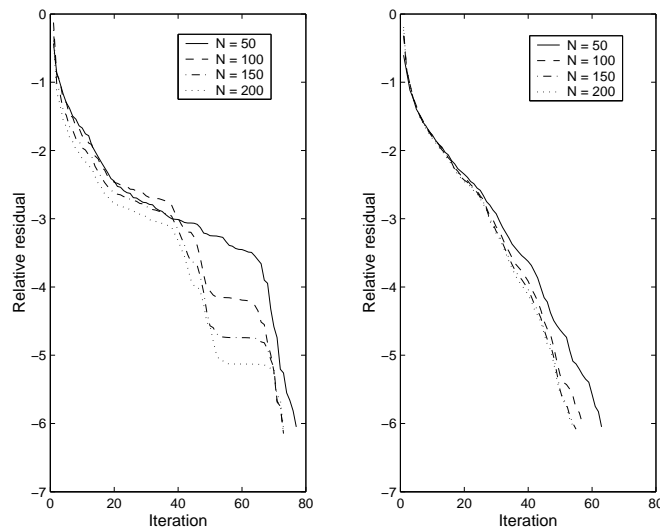
FIGURE 1. Typical GMRES convergence. $k = 40$ and boundary condition is: Dirichlet (left) and absorbing (right).

Table 1 shows the convergence of full GMRES [14] used to solve a 2D Helmholtz problem, with various h. Dirichlet boundary conditions are imposed at the boundaries. A right preconditioner is solved exactly using a direct method. For decreasing values of h these results indicate h-independent convergence. Even though our analysis is based on Dirichlet boundary conditions, the result remains valid numerically for absorbing boundary conditions (see Table 1). Only for high wavenumbers that the convergence is mildly dependent on h. But, as $h \to 0$ the iteration number likely converges to a certain value.

TABLE 1. Number of full GMRES iterations for different grid sizes $h = 1/N$. The problem is 2D with: Dirichlet boundary conditions (left), and absorbing boundary conditions (right). The iteration is terminated after the norm of residual is reduced to 10^{-6}.

$h_x^{-1} = h_y^{-1} = h^{-1}$	Dirichlet k				Absorbing cond. (3) k			
	10	20	30	40	10	20	30	40
50	14	24	42	77	12	23	39	63
100	13	23	43	73	12	23	39	57
150	13	22	41	73	12	23	39	55
200	13	21	41	73	12	23	38	54

Figure 1 shows the convergence of full GMRES for $k = 40$. Even tough the convergence exhibits some stages with slow convergence in the case of Dirichlet boundary condition (left), the convergence is still monotonically decreasing, which is typical for GMRES. Replacing Dirichlet boundary conditions with absorbing boundary conditions results in a more regular convergence behavior (Figure 1: right).

4. Krylov subspace method

In §3, we used GMRES to solve the preconditioned linear system (4). For large problems, however, this algorithm can become expensive due to increasing amount

of work. As the iteration number grows with the increase of k, the GMRES work also increases almost quadratically. Furthermore, the number of vectors to be stored also increase. One practical remedy for GMRES-type algorithms is restarting.

In GMRES(m), where m is the restart parameter, the convergence depends on the choice of m. There is no general rule to choose this parameter. The choice of m can negatively affect the convergence especially if the full GMRES shows a superlinear convergence. For our problem, see Figure 2, the convergence is very suitable for restarting the GMRES iteration if a low wavenumber is used. (For this type of problem, however, restarting GMRES is not necessary). The convergence, however, becomes superlinear as k increases. We can expect that if m is not properly chosen, the overall performance can be even worse. This is what we encounter, see Table 2. In general, restarting GMRES results in a less efficient method for the problem at hand.

TABLE 2. Comparison of GMRES(m) with different restart parameter m. Boundary conditions are as in (2). The number of iterations and CPU time are shown for $k = 40$.

Restart m	∞	5	10	15	25
Iter	57	115	99	97	91
CPU time	66.23	147.91	117.38	112.92	104.58

We also use algorithms based on short recurrence process, like Bi-CGSTAB [16] and COCG [17]. For Bi-CGSTAB, however, one additional matrix/vector multiplication and two preconditioner solves are required per each iteration as compared with one matrix/vector multiplication and one preconditioner solve in GMRES. Nevertheless, for large iteration number Bi-CSGTAB may be more efficient than GMRES. COCG is more attractive, as it requires only one matrix/vector multiplication and one preconditioner solve. COCG, however, can only be used for symmetric matrices. Therefore, it is important that the preconditioned form AM^{-1} (or $M^{-1}A$) is also symmetric. In general, if A and M are symmetric, so is AM^{-1} (or $M^{-1}A$).

TABLE 3. Number of matrix/vector multiplications for a typical 2D case with constant k with absorbing boundary condition (2). 30 gridpoints per wavelength are used. CPU time is shown between parentheses.

k	5	10	20	30	40	50
GMRES	8(0.16)	12(1.71)	23(26.30)	39(160.60)	54(578.99)	76(1801.90)
Bi-CGSTAB	11(0.16)	19(2.34)	37(38.54)	69(268.95)	95(963.87)	115(3106.45)
COCG	8(0.14)	13(1.70)	24(25.86)	44(175.23)	64(653.54)	89(2089.72)

In Table 3 we compare GMRES, Bi-CGSTAB, and COCG for a 2D constant k Helmholtz problem with absorbing conditions (2) at the boundaries. Again, direct methods are used to solve the preconditioner. Here, GMRES is found to be more effective than Bi-CGSTAB in terms of number of matrix/vector multiplications, and slightly wins over COCG. As already mentioned, GMRES, however, has an increase of storage as the number of iterations increases (in the case of higher wavenumber k). (In §6, as we use multigrid to approximate M^{-1}, GMRES proves to be less efficient than Bi-CGSTAB). Also, COCG seems to be more promising

than Bi-CGSTAB. The irregularity of COCG convergence may, however, make it difficult to determine a reliable termination criterion (see Figure 2 for an example of convergence for $k = 40$). A smoother convergence of COCG can be obtained by including a residual smoothing technique [18] in the algorithm. We did not do this.
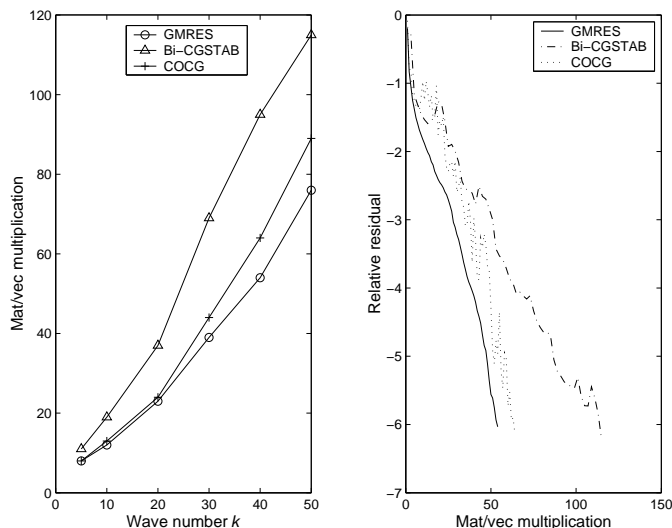

FIGURE 2. Left: Relation between the wavenumber and the number of matrix/vector multiplications with constant k in $\Omega = (0, 1)^2$. Right: Typical convergence history of some Krylov subspace methods. In this figure, the convergence is shown for $k = 40$, $n = 200^2$.

5. Multigrid as preconditioner solver

The preliminary numerical experiments so far have confirmed that using direct solvers for the preconditioner is practically too expensive. In this section we show that multigrid iteration can handle the preconditioner in a more efficient way. An important issue is that the preconditioning matrix derived from (5) is always complex, symmetric and positive definite. For this type of linear systems, multigrid is known to be efficient. The use of multigrid as a solver for this type of matrix is discussed, e.g., in [9]. We refer to [15] for an introduction to multigrid.

Multigrid is based on two principles: error smoothing and coarse grid correction. Starting with a fine grid, basic iterative methods exhibit an error smoothing effect, if appropriately applied. A smooth error can be well approximated on a coarse grid. This leads to a coarse grid correction. On a coarse grid, an iterative method is applied again to reduce the error. So, the same two principles are recursively repeated until the coarsest grid is reached, where the problem can be solved exactly using a direct method or approximately using an iterative method. As the result, the error can be reduced fast, and the amount of work to reach certain error reduction is low because a coarse grid procedure is a cheap procedure.

Iterative methods which are known to have a smoothing effect are damped Jacobi and Gauss-Seidel iteration. The smoothing properties of these types of iteration methods are explained, e.g., in [15]. For coarse grid correction, a widely used coarse grid procedure is the one that based on the Galerkin coarse grid operator defined

as

$$(15) \qquad\qquad M_H := I_h^H M_h I_H^h,$$

where indices h and H are related to the fine and coarse grid. In (15), I_h^H and I_H^h are the transfer operators from the fine to the coarse grid, and vice versa. I_h^H is the restriction operator, which maps fine grid functions to coarse grid functions. I_H^h is the prolongation operator, which maps coarse grid functions to fine grid functions. Here, we use bi-linear interpolation as the prolongator and for the restrictor we set $I_h^H = (I_H^h)^*$, which gives the full weighting operator.

Asymptotic convergence factors of multigrid as a solver for the preconditioning matrix in 2D for different number of pre- and post-smoothing are shown in Table 4. The wavenumber is constant in $\Omega = (0,1)^2$.

TABLE 4. Multigrid convergence factors for a discrete 2D precon-ditioner operator (5) with $\beta_1 = 0$ and $\beta_2 = 1$ in $\Omega = (0,1)^2$. Dirichlet boundary conditions are used at the boundaries.

cycle	n_{pre}	n_{post}	$k = 10$ h^{-1}			$k = 50$ h^{-1}		
			50	100	200	50	100	200
V	1	0	0.592	0.592	0.707	0.576	0.592	0.592
	1	1	0.351	0.438	0.628	0.332	0.351	0.351
F	1	0	0.592	0.592	0.592	0.576	0.592	0.592
	1	1	0.351	0.351	0.351	0.332	0.351	0.351

From Table 4 we see that standard multigrid methods can be used for complex-valued linear systems. We obtain h-independent convergence with the F-cycle, while the V-cycle results in a mildly h-dependent convergence. One pre-smoothing and one post-smoothing also gives better convergent factors than one pre-smoothing and no post-smoothing. We will use the F(1,1)-cycle in our numerical examples in the next section for the preconditioner solve.

6. Numerical examples

In this section, we present some numerical results obtained from solving (1), with boundary conditions of the form either (2) or (3). For the main iteration, we use GMRES and Bi-CGSTAB. The preconditioner is (5) with $\beta_1 = 0$ and $\beta_2 = 1$ and is solved with multigrid. In order to reduce CPU time, we do not solve the preconditioner accurately using multigrid. We use only one multigrid iteration. Furthermore, we consider Jacobi iteration as the smoother with relaxation factor $\omega = 0.8$ (or 0.8-JAC).

As already mentioned, for the preconditioned COCG we require that the linear system AM^{-1} to be symmetric. As we use the F(1,1)-cycle multigrid, this condition, however, is not satisfied. Therefore, in this section we do not use COCG.

6.1. Constant wavenumber k. The first example is the same test case as in §3. We first use the first order boundary condition (2) at the boundaries. The numerical performance is presented in Table 5 in terms of matrix/vector multiplications and CPU time.

Since multigrid only approximates M^{-1}, the number of iterations is slightly larger than those in Table 3. CPU time, however, decreases substantially. One fact revealed from the results with multigrid is that GMRES now is less efficient than

Bi-CGSTAB, even though GMRES requires fewer matrix/vector multiplications than Bi-CGSTAB to reach convergence.

As already expected, COCG, which requires AM^{-1} to be symmetric, is found not to be a good method due to the use of the F-cycle. Only for some values of low wavenumber COCG iterations convergence.

TABLE 5. Convergence of GMRES and Bi-CGSTAB used to solve (1) with first order boundary condition (2) and constant k. The number of iterations and CPU time (between parentheses) are shown.

$k =$	5	10	20	30	40	50
GMRES	12(0.01)	15(0.05)	37(1.46)	55(2.44)	74(7.14)	92(16.19)
Bi-CGSTAB	15(0.01)	21(0.05)	47(0.46)	81(2.01)	101(4.76)	121(9.82)

In Table 6, convergence results with the same model problem but with the second order absorbing conditions (3) at the boundaries are shown. This boundary condition affects the computational performance slightly; more iterations are required to reach convergence.

TABLE 6. Convergence of GMRES and Bi-CGSTAB used to solve (1) with first order boundary condition (3) and constant k. The number of iterations and CPU time (between parentheses) are shown.

$k =$	5	10	20	30	40	50
GMRES	18(0.01)	24(0.08)	38(0.52)	64(3.17)	66(6.25)	90(15.74)
Bi-CGSTAB	25(0.01)	35(0.08)	49(0.51)	83(2.21)	99(4.96)	115(9.05)

The solution using the second order absorbing condition is, however, much more preferable than the solution using the first order one, as shown in Figure 3 for $k = 50$. Although the wave velocities are similar, one can distinguish differences in the wave amplitude in Figure 3, which are mainly due to the reflections from the boundaries. The second order absorbing condition provides a better boundary treatment than the first order one, indicated by fewer reflections from the boundaries.

For the next examples we only show convergence results with the second order absorbing condition.

6.2. Layered model. The second example is a layered model in unit domain $\Omega = (0,1)^2$. The wavenumber in Ω varies as follows:

$$(16) \qquad k(x,y) = \begin{cases} \frac{4}{3}k_{reff}, & \text{if } 0 \le y \le \frac{1}{3}, \\ k_{reff}, & \text{if } \frac{1}{3} < y \le \frac{2}{3}, \\ 2k_{reff}, & \text{if } \frac{2}{3} < y \le 1. \end{cases}$$

The solutions for $k_{reff} = 50$ are shown in Figure 4. As already expected, using the second order absorbing condition results in a much reduced reflection from the boundaries, as compared to the first order one.
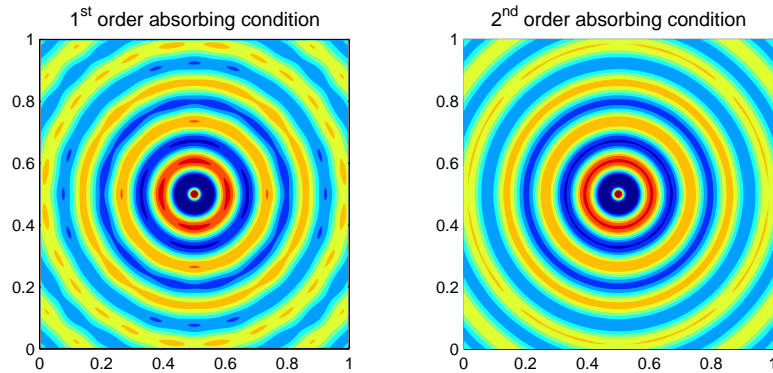
FIGURE 3. Real part of the solution from a 2D constant k problem, with $k = 50$. Left: the first order absorbing condition. Right: the second order absorbing condition (right).

The convergence results are shown in Table 7 for GMRES and Bi-CGSTAB. In terms of matrix/vector multiplications, GMRES is somewhat better than Bi-CGSTAB. With respect to CPU time, however, Bi-CGSTAB is faster than GMRES.

TABLE 7. Convergence of GMRES and Bi-CGSTAB from the 2D layered problem with second order absorbing conditions (3).The number of iterations and CPU time (between parentheses) are shown.

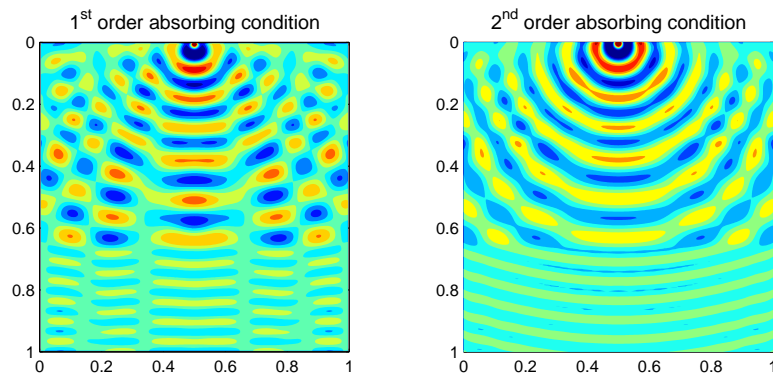| $k_{reff} =$	5	10	20	30	40	50
GMRES	25(0.02)	40(0.14)	69(1.16)	99(5.99)	116(14.58)	145(33.94)
Bi-CGSTAB	33(0.02)	55(0.13)	87(0.85)	125(3.26)	143(6.77)	177(13.91)

FIGURE 4. Real part of the solution from a 2D layered problem with $k_{reff} = 50$. Left: the first order absorbing condition. Right: the second order absorbing condition.

6.3. Cross-well: a guided wave. The last example is from a wave guide model in a physical domain $\Omega = (0, 130) \times (0, 150)$ m^2. This model mimics a cross-well situation, where guided wave propagation occurs. A source is positioned at the depth of 60 meter inside a low velocity zone (see Figure 5). Instead of using wavenumber, the source is determined in terms of wave frequency, f, which is related to k as $k = 2\pi f/c$, with c the local speed of sound (in ms^{-1}). The solutions are also shown in Figure 5, for the two boundary conditions. From this figure, we can see that most of the energy is inside the low velocity layer and creates a guided wave.
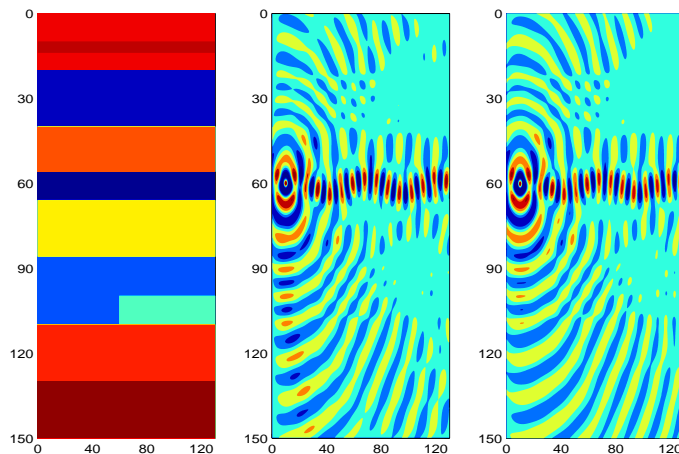

FIGURE 5. Real part of the solution of the 2D guided wave problem using the first order absorbing condition (mid), and using the second order absorbing condition (right). The frequency is 300 Hz.

Table 8 shows numerical performance of Bi-CGSTAB with 650×750 grid points, where the second order absorbing condition is imposed at the boundaries. We omit the computation using GMRES because of memory requirements. For various frequencies the method converges satisfactorily to the specified accuracy.

TABLE 8. Convergence results of Bi-CGSTAB from the 2D guided wave problem. The second order absorbing condition is used

f (Hz)	50	100	200	300
Iter	44	120	118	155

7. Conclusion

An iterative solution method for the heterogeneous Helmholtz equation is described and numerical examples have been shown to indicate the performance of the method. The method is based on a Krylov subspace iterative method and a multigrid based preconditioner. Two Krylov subspace methods have been studied: Bi-CGSTAB and GMRES. Even though Bi-CGSTAB requires more matrix/vector multiplications than GMRES, it is more efficient from a practical point of view.

Standard multigrid methods are found to be extendable to the complex-valued linear system (where in our case is the preconditioning matrix). Using multigrid

to solve the preconditioner, one loses only some iteration numbers, but enables to reduce CPU time substantially.

The method is also extendable to heterogeneous Helmholtz problems. From our numerical results, the method shows acceptable performance, without any potential of breakdown.

References

[1] A. Bayliss, C.I. Goldstein, E. Turkel, An iterative method for Helmholtz equation, J. Comput. Phys. 49 (1990), pp. 323–352.

[2] J.-P. Berenger, A perfectly matched layer for the absorption of electromagnetic waves, J. Comput. Phys. 114 (1994), pp. 185-200.

[3] H.R. Elman, O.G. Ernst, D.P. O'Leary, A multigrid method enhanced by Krylov subspace iteration for discrete Helmholtz equations, SIAM J. Sci. Comput. 23 (2001), pp. 1291–1315.

[4] B. Engquist, A. Majda, Absorbing boundary conditions for the numerical for the numerical simulation of waves, Math. Comp. 31 (1977), pp. 261–263

[5] Y.A. Erlangga, C. Vuik, C.W. Oosterlee, On a class of preconditioners for solving the Helmholtz equation, Appl. Numer. Math. 50(2004), pp. 409–425

[6] Yogi Erlangga, Kees Vuik, Kees Oosterlee, Rene-Edouard Plessix, Wim Mulder, A robust iterative solver for the two-way wave equation based on a complex shifted Laplace preconditioner, SEG Expanded Abstract 23, Denver, Colorado, 2004, pp. 1897–1900.

[7] M.J. Gander, F. Nataf, AILU for Helmholtz problems: A new preconditioner based on the analytic parabolic factorization, J. Comput. Acoustics 9(4) (2001), pp. 1499–1509.

[8] J. Gozani, A. Nachson, E. Turkel, Conjugate gradient coupled with multi-grid for an indefinite problem, in: R. R. Vichnevetsky, R.S. Tepelman (Eds.), in: Advances in Computer Methods for Partial Differential equations, Vol. V, IMACS, New Brunswick, NJ, 1984, pp. 425–427.

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

[10] A.L. Laird, M.B. Giles, Preconditioned iterative solution of the 2D Helmholtz equation, Report No. 02/12, Oxford Comp. Lab. Oxford, UK, 2002.

[11] T.A. Manteuffel, S.V. Parter, Preconditioning and Boundary Conditions, SIAM J. Numer. Anal. 27(3) (1990) 656–694

[12] W.A. Mulder, R.E. Plessix, One-way and two-way wave migration, SEG Annual Meeting, 2003, pp. 881–884

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

[14] Y. Saad, M.H. Schultz, GMRES: A generalized minimal residual algorithm for solving non-symmetric linear systems, SIAM J. Sci. Stat. Comput. 7 (2) (1986), pp. 856–869.

[15] U. Trottenberg, C.W. Oosterlee, A. Schüller, Multigrid, Academic Press, London, 2001.

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

[17] H.A. van der Vorst, J.B.M. Melissen, A Petrov-Galerkin type method for solving $Ax = b$, where A is symmetric complex, IEEE Trans. Magnetics 26(2) (1990), pp. 706–708.

[18] L. Zhou, H.F. Walker, Residual smoothing techniques for iterative methods, SIAM J. Sci. Comput. 15(2) (1994), pp. 297–312.

Currently at Department of Aerospace Engineering, Institute of Technology at Bandung, Ganesha 10, Bandung 40132, Indonesia.
    E-mail: `Y.A.Erlangga@tudelft.nl`

Delft Institute of Applied Mathematics, Delft University of Technology, Mekelweg 4, 2628 CD Delft, The Netherlands.
    E-mail: `C.Vuik@tudelft.nl and C.W.Oosterlee@tudelft.nl`