Research Article

Comparison of Algebraic Multigrid Preconditioners for Solving Helmholtz Equations

Dandan Chen, Ting-Zhu Huang, and Liang Li

School of Mathematical Sciences, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China

Correspondence should be addressed to Ting-Zhu Huang, tingzhuhuang@126.com

An algebraic multigrid (AMG) with aggregation technique to coarsen is applied to construct a better preconditioner for solving Helmholtz equations in this paper. The solution process consists of constructing the preconditioner by AMG and solving the preconditioned Helmholtz problems by Krylov subspace methods. In the setup process of AMG, we employ the double pairwise aggregation (DPA) scheme firstly proposed by Y. Notay (2006) as the coarsening method. We compare it with the smoothed aggregation algebraic multigrid and meanwhile show shifted Laplacian preconditioners. According to numerical results, we find that DPA algorithm is a good choice in AMG for Helmholtz equations in reducing time and memory. Spectral estimation of system preconditioned by the three methods and the influence of second-order and fourth-order accurate discretizations on the three techniques are also considered.

1. Introduction

In this paper, the time-harmonic wave equation in 2D homogeneous media is solved numerically. The essential equation is Helmholtz equation, which governs wave scattering and propagation phenomena arising in acoustic problems in many areas, such as geophysics, aeronautics, and optical problems. In particular, we are in search of solutions of the Helmholtz equation discretized by the finite difference method. The discrete problem becomes extremely large for very high wavenumber, because the number of gridpoints per wavelength should be sufficiently large in order to result in accepted solutions. In this case, direct methods are difficult to solve, and iterative methods are the interesting alternative. However, Krylov subspace iterative methods are not competitive without a good preconditioner. In this paper, we consider an algebraic multigrid with aggregation scheme as preconditioning to improve the convergence of Krylov subspace iterative methods.

In [1], Bayliss et al. proposed a preconditioner based on the Laplace operator for solving the discrete Helmholtz equation efficiently with CGNR. In [2], Laird and Giles proposed a preconditioner where an extra term is added to the Laplace operator for solving the discrete Helmholtz equation. Subsequently, in [3], Erlangga et al. generalized the above two kinds of preconditioners and obtained a new class of preconditioners, the so-called "shifted Laplacican" preconditioner of the form $-\Delta\phi - \alpha k^2\phi$ with $\alpha = -(\beta_1 - \beta_2 i) \in \mathbb{C}$, where $i = \sqrt{-1}$ is the imaginary unit. In 2006, Erlangga et al. [4] compared multigrid with incomplete LU used to approximate the shifted Laplacian preconditioner to construct the final preconditioner for the inhomogeneous Helmholtz equation, and concluded that multigrid applied to approximate the shifted Laplacian preconditioner with Bi-CGSTAB resulted in a fast and robust method. In 2006, Erlangga et al. [5] proposed a multigrid $V(1,1)$-cycle with de Zeeuw's prolongation operator, FW (full weighted) restriction, and Jacobi smoothing with the relaxation parameter $\omega = 0.5$. The smallest size of the parameter β_2 in front of the imaginary Helmholtz term in the preconditioner, for which the multigrid method could be successfully employed, has been determined to $\beta_2 = 0.5$ and meanwhile $\beta_1 = 1$. In 2007, Van Gijzen et al. [6] analyzed the spectral of the discrete Helmholtz operator preconditioned with a shifted Laplacian, and proposed an optimal value for the shift by combining the results of the spectral analysis with an upper bound on the GMRES residual norm. In 2009, Umetani et al. [7] proposed a multigrid $V(0,1)$-cycle with AMG's prolongation operator, FW restriction, and incomplete LU postsmoothing for a fourth-order Helmholtz discretization based on [5]. The fourth-order accurate shifted Laplacian preconditioner could be easily approximated by one $V(0,1)$-cycle of multigrid and enables us to choose a somewhat smaller imaginary shift parameter ($\beta_2 = 0.4$) in the shifted Laplacian preconditioner, which improves the solvers convergence (especially for high wavenumbers on fine meshes). In 2007, Airaksinen et al. [8] proposed a preconditioner based on an algebraic multigrid approximate of the inverse of a shifted Laplacian for the Helmholtz equation. This is a generalization of the preconditioner proposed by Erlangga et al. in [5]. In 2010, Olson and Schroder [9] proposed a smoothed aggregation algebraic multigrid method for 1D and 2D scalar Helmholtz problems with exterior radiation boundary conditions discretized by 1D finite difference and 2D discontinues Galerkin. In the same year, Notay [10] proposed an aggregation-based algebraic multigrid (AGMG) method, in which double pairwise aggregation scheme firstly proposed by Notay in [11] is used.

We will consider using the double pairwise aggregation algorithm to coarsen in the setup process of the general algebraic multigrid method in this paper in order to improve the solution effects of Helmholtz problems.

This paper is organized as follows. In Section 2, we discuss the Helmholtz equation, and the discrete finite difference formulations of second order and fourth order. The iterative solution methods with the three different kinds of preconditionings are presented in Section 3. Numerical results are reported in Section 4, and conclusions are given in Section 5.

2. The Helmholtz Equation and Discretizations

2.1. Mathematical Problem Definition

We solve wave propagations in a two dimensional medium in a unit domain governed by the Helmholtz equation

$$-\Delta\phi - k^2(x,y)\phi = g, \qquad \Omega = [0,1]^2, \tag{2.1}$$

Table 1: Discretization steps related to wavenumbers.

k	10	20	30	40	50	60	70	80	90
h	1/16	1/32	1/48	1/64	1/80	1/96	1/112	1/128	1/144

where $\Delta \equiv \partial^2/\partial x^2 + \partial^2/\partial y^2$ is the Laplace operator, $\phi(x, y)$ represents the pressure field in the frequency domain, the source term is denoted by g, and $k(x, y)$ is the wavenumber, which is a constant in the homogeneous domain. Otherwise, the wavenumber $k = \omega/c(x, y)$ is space dependent because of a spatially dependent speed of sound $c(x, y)$ in a heterogeneous medium. With angular frequency $\omega = 2\pi f$ (f is the frequency in Hertz), wavelength ℓ is defined by $\ell = c/f$. So the wavenumber $k = 2\pi/\ell$. The number of wavelengths in a domain of size L equals L/ℓ. n_ω, the number of points per wavelength, is typically chosen to be 10–12 points. A dimensionless discretization step reads $h = \ell/(n_\omega L)$, and therefore $kh = 2\pi/(n_\omega L)$. With domain size $L = 1$, an accuracy requirement for second-order discretizations is that $kh \leq \pi/5$ (≈ 0.63) for $n_\omega = 10$ points per wavelength, and $kh \leq 0.53$ with $n_\omega = 12$ points per wavelength. We select combinations of wavenumber k and the discretization step h with $kh = 0.625$, which is displayed in Table 1, in the latter numerical experiments.

The boundary condition can be the Dirichlet boundary condition or the first-order radiation boundary condition, and they are, respectively, as follows:

$$\phi = 0, \quad \text{on } \Gamma = \partial\Omega, \quad (\text{Dirichlet boundary condition}) \tag{2.2}$$

$$\frac{\partial\phi}{\partial n} - ik\phi = 0, \quad \text{on } \Gamma = \partial\Omega, \quad (\text{Sommerfeld condition}), \tag{2.3}$$

where n is an outward direction normal to the boundary.

2.2. Finite Difference Discretizations

The Helmholtz equations are discretized either by a second-order or by a fourth-order finite difference scheme, resulting in the linear system

$$A_h\phi_h = b_h, \tag{2.4}$$

where ϕ_h and b_h represent the discrete frequency-domain pressure field and the source, respectively.

The well-known 5-point stencil related to a second-order ($o(h^2)$) accurate discretization for Helmholtz equation (2.1) reads

$$A_h \triangleq \frac{1}{h^2}\begin{bmatrix} & -1 & \\ -1 & 4 - (kh)^2 & -1 \\ & -1 & \end{bmatrix}. \tag{2.5}$$

The fourth-order ($o(h^4)$) accurate discretization named by HO discretization [12] for Helmholtz equation (2.1) is given with stencil

$$A_h^{\text{HO}} \triangleq \frac{1}{h^2} \begin{bmatrix} -\dfrac{1}{6} & -\dfrac{2}{3} - \dfrac{(kh)^2}{12} & -\dfrac{1}{6} \\[2mm] -\dfrac{2}{3} - \dfrac{(kh)^2}{12} & -\dfrac{10}{3} - \dfrac{2(kh)^2}{3} & -\dfrac{2}{3} - \dfrac{(kh)^2}{12} \\[2mm] -\dfrac{1}{6} & -\dfrac{2}{3} - \dfrac{(kh)^2}{12} & -\dfrac{1}{6} \end{bmatrix}, \tag{2.6}$$

Compared with a second-order discretization method, a fourth-order discretization method could make the number of grid points per wavelength be smaller and discrete the equations to smaller matrices for the same level of accuracy. It is possible to lead to an algorithm that is more efficient in terms of the accuracy of the solution versus the computational cost. As in [7], the multigrid-based shifted Laplacian preconditioner with the fourth-order discretization is preferable.

The boundary discretization is also considered, and we could employ the first-order forward or backward scheme to approximate the first-order derivative in (2.3).

Next, we can obtain a linear system through substituting

$$A_h x = b_h, \qquad A_h \in \mathbb{C}^{N \times N}, \tag{2.7}$$

where A_h is a large, spare symmetric matrix and N is the number of grid points. The matrix A_h remains positive definite as long as k^2 is smaller than the minimum eigenvalue of the discrete Laplacian, is indefinite with both positive and negative eigenvalues for large values of k, and is complex-valued because of absorbing boundary conditions. However, it is non-Hermitian. Apparently, the size of the linear system (2.7) gets very large for the high frequency.

3. Iterative Solution Methods

3.1. Preconditioned Krylov Subspace Iterative Methods

Iterative methods for linear system (2.7) within the class of Krylov subspace method are based on the construction of iterants in the subspace

$$\mathbf{K}^j(A, r_0) = \mathbf{span}\left\{ r_0, Ar_0, A^2 r_0, \dots, A^{j-1} r_0 \right\}, \tag{3.1}$$

where $\mathbf{K}^j(A, r_0)$ is the jth Krylov subspace associated with A and r_0, $r_0 = b - Ax_0$, is the initial residual.

The Bi-CGSTAB algorithm [13] is one of the better known Krylov subspace algorithms for non-Hermitian problems, which has been used for Helmholtz problems, for example, in [5, 8]. One of the advantages of Bi-CGSTAB, compared to full GMRES, is its limited memory requirements. Bi-CGSTAB is based on the idea of computing two mutually biorthogonal bases for the Krylov subspaces based on matrix A_h and its conjugate transpose A_h^H and is easy to implement.

Without a preconditioner, Krylov subspace methods converge very slowly, or not at all, for the problem of interest in [4]. So a preconditioner should be incorporated to improve the convergence of Krylov subspace methods. By left preconditioning, one solves a linear system premultiplied by a preconditioning matrix M_h^{-1},

$$M_h^{-1} A_h = M_h^{-1} b_h. \tag{3.2}$$

The challenge is to find a form of matrix M_h, whose inverse matrix M_h^{-1} can be efficiently approximated, such that $M_h^{-1} A_h$ has a spectrum that is favorable for Krylov subspace iterative solution methods.

In this paper, we choose Bi-CGSTAB and GMRES with preconditionings to solve discrete Helmholtz equations. In [3], GMRES is used to solve the Helmholtz equation with the shifted Laplacian preconditioner and is compared with Bi-CGSTAB. As a result, Bi-CGSTAB is preferable since the convergence for heterogeneous high wavenumber Helmholtz problems is typically faster than that of GMRES.

3.2. Shifted Laplacian (SL) Preconditioners

In [5], a shifted Laplacian operator was proposed as a preconditioner for the Helmholtz equation, with M_h defined as a discretization of

$$M_{\mathrm{SL}} = -\partial xx - \partial yy + \alpha k^2(x, y), \quad \alpha \in \mathbb{C}, \ \alpha = -(\beta_1 - \beta_2 i), \tag{3.3}$$

and boundary conditions were set identically to those for the original Helmholtz equation. The readers could refer to the related contents in [4, 7, 8, 14].

There are different choices of the parameter pair (β_1, β_2) such as $(\beta_1, \beta_2) = (0, 0)$ (Laplace preconditioner in [1]), $(\beta_1, \beta_2) = (-1, 0)$ (Laird and Giles preconditioner in [2]), $(\beta_1, \beta_2) = (0, 1)$ (Erlangga et al. preconditioner in [3]), $(\beta_1, \beta_2) = (1, 1)$ (basic parameter pair choice). In [5], Erlangga et al. evaluated the influence of parameters β_1 and β_2 to the multigrid method which was applied to approximate the shifted Laplacian operator to construct the final preconditioner, and the proposed optimal parameter pair for the solver was that $(\beta_1, \beta_2) = (1, 0.5)$. Based on [5], Umetani et al. [7] considered using the fourth-order accurate finite difference discretization with the result that the parameter pair $(\beta_1, \beta_2) = (1, 0.4)$ is the best choice. In this paper, we merely consider the shifted Laplacian preconditioners, without the multigrid or incomplete LU method to approximate, which is not the same as that in [5, 7]. So we will also choose that $(\beta_1, \beta_2) = (1, 0.5)$ and $(\beta_1, \beta_2) = (1, 0.3)$ for comparison with the previously mentioned parameter pair (β_1, β_2) in the numerical experiments.

3.3. Smoothed Aggregation (SA) Preconditioning

In this section, we will present an overview of the smoothed aggregation setup algorithm. The function of the SA setup phase is to construct coarse operators A_k through interpolation operators

$$P_k : \mathfrak{R}^{n_{k+1}} \longrightarrow \mathfrak{R}^{n_k}, \tag{3.4}$$

(1) $t_\ell = S_\ell^{\nu_1}$ (ones$(n, 1), 0$)
(2) While $\ell \leq L$
(3) [neighbour, hoods] = aggregate (n_ℓ, A_ℓ);
(4) $[P, t_{\ell+1}] = $ qr$(n_\ell, t_\ell, $hoods$)$;
(5) $S_\ell = $ smoother $(n_\ell, A_\ell, \omega, $neighbour$)$;
(6) $I_{\ell+1}^\ell = S_\ell P$;
(7) $A_{\ell+1} = (I_{\ell+1}^\ell)^T A_\ell I_{\ell+1}^\ell$;
(8) $\ell = \ell + 1$;
(9) Endwhile

Algorithm 1: SA_setup (n_ℓ, A_ℓ, ℓ).

where n_k and n_{k+1} are sizes of two successively coarser grids. The initial matrix A_0 associated with the finest level equals A. The inputs need the matrix A and one near null-space candidate t that is intended to form the interpolation basis. t is an algebraically smooth mode that is slow to relax on the fine mesh. The details are described in Algorithm 1.

When the current coarse level $\ell \leq L(L$ is the coarsest level), we gain two sets respectively named by "neighbour" and "hoods." The "neighbor $[i]$" includes the nodes which are strongly connective to the node i, and the "hoods $[i]$" contains the nodes which are in the ith aggregation set. Next, we reset t_ℓ according to "hoods," and QR factorization is applied to it in order to get the initial interpolation operator P and the next $t_{\ell+1}$. Third, we obtain the smooth operator S_ℓ according to the relative algorithm in [15]. Fourth, we get the final interpolation operator $I_{\ell+1}^\ell$, through the smooth operator S_ℓ is acted on the initial interpolation operator P. Finally, we can easily obtain the coarse operator $A_{\ell+1}$. The interested readers could consult the related contents in [9]. We will employ the SA setup algorithm in AMG to approximate the original system in order to obtain the preconditioner and we name this process by SA preconditioning.

3.4. Double Pairwise Aggregation (DPA) Preconditioning

An algebraic coarsening algorithm sets up an interpolation operator P only making use of the information available in the fine grid matrix A. The interpolation operator P is a $n \times n_c$ matrix, where n_c $(n_c \leq n)$ is the number of coarse variables. With acted on by the interpolation operator, a vector defined on the coarse variable set $[1, n_c]$ could be transferred on the fine grid. The coarsen grid matrix A_c depends not only on the interpolation matrix P, but also on the restriction matrix R and the fine grid matrix A. It is usual to take the restriction operator R equals to the transpose of the interpolation operator P. Therefore, the coarsen grid matrix A_c is computed from the Galerkin formula

$$A_c = RAP = P^T AP. \tag{3.5}$$

Coarsening by aggregation needs to define aggregates G_i, which are disjoint subsets of the variable set. The number of these aggregations is the number of coarse variables n_c, and the interpolation operator P is given by

$$P_{i,j} = \begin{cases} 1, & \text{if } i \in G_j, \\ 0, & \text{otherwise} \end{cases} \quad (1 \leq i \leq n, \ 1 \leq j \leq n_c). \tag{3.6}$$

Note that there is no need to explicitly form interpolation operator P, and the coarse grid matrix is practically computed by

$$(A_c)_{i,j} = \sum_{k \in G_i} \sum_{\ell \in G_j} a_{k\ell} \quad (1 \le i, j \le n_c). \tag{3.7}$$

The double pairwise aggregation algorithm was first proposed by Notay in [11], which employed two passes of the pairwise aggregation algorithm, with the result of decreasing the number of variables by a factor slightly less than four in most cases. Subsequently, we will take the double pairwise aggregation scheme in the setup process of AMG, in order to construct a better preconditioner with Bi-CGSTAB for discrete Helmholtz equations. We also name this process by DPA preconditioning. The details of aggregation algorithms could be consulted in [10, 11].

4. Experiments

We will consider the following problem and use a uniform mesh with constant mesh size h in all directions. In the experiments, we will compare the three preconditionings combined with Bi-CGSTAB or GMRES for solving discrete Helmholtz equations. The shifted Laplacian preconditioner (SLP), the smoothed aggregation preconditioning (SAP), and the double pairwise aggregation preconditioning (DPAP) are, respectively, introduced in Section 3. We let that the iterative method is terminated at the kth step if $\|b - Ax_k\|_2 / \|b\|_2 \le 10^{-6}$. All the numerical results are from running relative Matlab programs in the computer with the CPU of AMD Pentium Dual Core 4000+.

4.1. The Close-off Problem and Numerical Results

We consider a problem in a rectangular homogeneous medium governed by

$$-\Delta\phi - k^2\phi = \sin(\pi x)\sin(\pi y)\sin\left(\sqrt{2}\pi x\right)\sin\left(\sqrt{3}\pi y\right), \quad x = [0,1], \ y = [0,1], \tag{4.1}$$

$\phi = 0$, at the boundaries.

Different grid resolutions are used to solve this problem with various wavenumbers in Table 1.

Firstly, we will observe the distribution of eigenvalues of the systems preconditioned by the three different preconditionings in order to estimate which preconditioning is best combined with Krylov subspace iterative methods. Since the spectral analysis of shifted Laplacian preconditioners $(M_{SL})_h$ with different parameter pairs (β_1, β_2) was considered in many papers [5–7, 14] and $(\beta_1, \beta_2) = (1, 0.5)$ is an almost better choice, we will only present the distribution of eigenvalues of the system preconditioned by the shifted Laplacian preconditioner $(M_{SL})_h$ with $(\beta_1, \beta_2) = (1, 0.5)$ with second-order and fourth-order accurate discretizations. We employ two-gird process to approximate V-cycle of AMG, and so $M_{AMG}^{-1}A = I - S(I - PA_C^{-1}P^TA)S$, where I is unit matrix and S is smoothing matrix. Second-order and fourth-order accurate discretizations are taken into account. Next, we note preconditioners produced by the smoothed aggregation preconditioning and by the double pairwise aggregation preconditioning separately as $(M_{SA})_h$ and $(M_{DPA})_h$. In Figures 1(a),
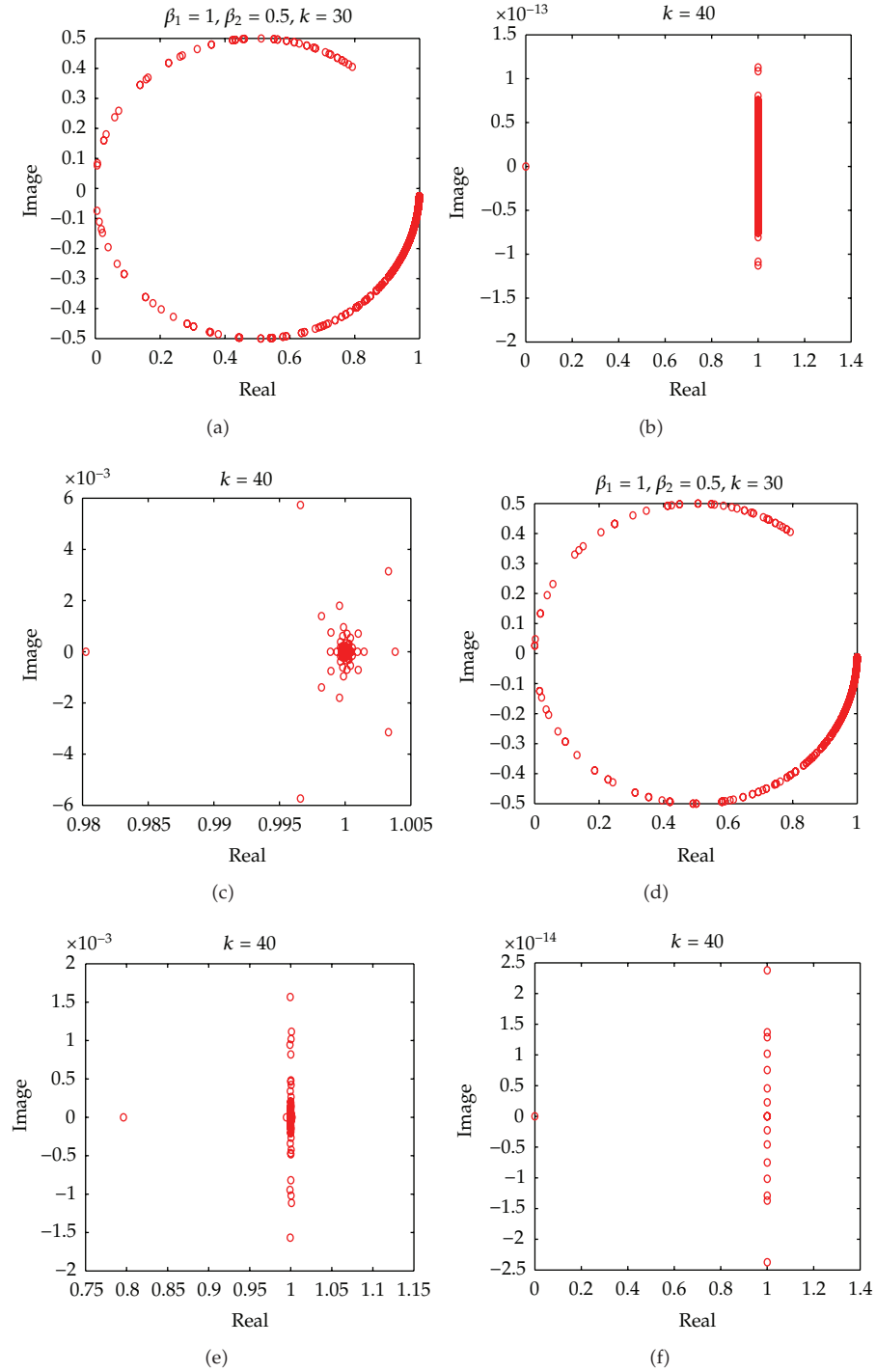
(a)

(b)

(c)

(d)

(e)

(f)

Figure 1: Spectral pictures.

Table 2: Computational performance of GMRES with SLPs.

| k | $(\beta_1, \beta_2) = (0,0)$ | | $(\beta_1, \beta_2) = (-1,0)$ | | $(\beta_1, \beta_2) = (0,1)$ | | $(\beta_1, \beta_2) = (1,0.5)$ | | $(\beta_1, \beta_2) = (1,0.3)$ | |
	Iter	$T\_$solve (s)	Iter	$T\_$solve (s)	Iter	$T\_$solve (s)	Iter	$T\_$solve (s)	Iter	$T\_$solve (s)
10	15	0.593	19	0.282	18	0.625	13	0.39	11	3.422
20	50	1.218	64	1.406	56	98.032	38	2.046	29	6.328
30	99	4.547	125	5.203	113	185.468	69	170.484	—	—

Table 3: Computational performance of Bi-CGSTAB with SAP.

| k | SAP (T_O) | | | SAP (F_O) | | |
	Iter	$T\_$setup (s)	$T\_$solve (s)	Iter	$T\_$setup (s)	$T\_$solve (s)
10	5	4.484	0.328	5	4.391	0.265
20	7	72.718	0.500	5	72.516	0.391
30	4	417.172	0.672	4	419.953	0.703
40	6	3114.88	14.578	6	1586.28	6.375

Note: presmooth = 0, postsmooth = 1, smoother = ILU, setup_Alg = sagg, $V\_$cycle.

1(b), and 1(c) are, respectively, spectral pictures of $(M_{\text{SL}})_h^{-1} A_h$, $(M_{\text{SA}})_h^{-1} A_h$ and $(M_{\text{DPA}})_h^{-1} A_h$ with second-order accurate discretization, and Figures 1(d), 1(e), and 1(f) are respectively spectral pictures of $(M_{\text{SL}})_h^{-1} A_h$, $(M_{\text{SA}})_h^{-1} A_h$, and $(M_{\text{DPA}})_h^{-1} A_h$ with fourth-order accurate discretization. Meanwhile, we compute the condition number of the preconditioned systems, and $\text{cond}_2((M_{\text{SL}})_h^{-1} A_h) = 211.1400$, $\text{cond}_2((M_{\text{SA}})_h^{-1} A_h) = 1.3062$, and $\text{cond}_2((M_{\text{DPA}})_h^{-1} A_h) = 1.3078$ separately correspond to Figures 1(a), 1(b), and 1(c).

Next, we will present numerical results in Tables 2, 3, and 4 and explain the meaning of signs in them. Tables 2, 3, and 4, respectively, show the computational performance of GMRES with SLPs, Bi-CGSTAB with SAP, and Bi-CGSTAB with DPAP for solving the closed-off problem with finite difference discretizations, in terms of the number of iterations and computational time, to reach the specified convergence. We also test Bi-CGSTAB with SLPs, but it leads to more iterations and solve time. So we do not present the corresponding results. Meanwhile, we find that Bi-CGSTAB with SLPs when $\beta_1 = 1$ and β_2 in the interval $(0,1)$ is better than GMRES, and the fourth-order accurate discretization could make the iterative solution rate more quick. Nevertheless, we make the focus only on the second-order accurate discretization for shifted Laplacian preconditioners, so the second-order accurate discretization is used in Table 2. "Iter" denotes the number of iterations, and "—" means that the corresponding experimental result is not given. "$T\_$setup (s)," and "$T\_$solve (s)" respectively, denote the setup time of preconditioner and the time of iterative solution by the second, and "T_O" and "F_O," respectively, mean the second-order and fourth-order finite difference discretizations shown in Section 2. The values of "presmooth" and "postsmooth" respectively, denote the number of pre- and postsmoothing iterations in the solve process of AMG, "smoother" denotes the chosen smoothing method, and "setup_Alg" denotes the coarsening algorithm used in the setup process of AMG.

4.2. Analysis of Numerical Results

We will analyze the numerical results from the following aspects.

Firstly, from Figure 1, by comparing pictures (a), (b), (c), we find that the eigenvalues of (b) and (c) are more farther from zero point and more close with one, but the eigenvalues of

Table 4: Computational performance of Bi-CGSTAB with DPAP.

k	DPAP(T_O)			DPAP(F_O)		
	Iter	$T\_$setup(s)	$T\_$solve(s)	Iter	$T\_$setup(s)	$T\_$solve(s)
10	3	0.968	0.547	3	0.922	0.453
20	3	2.938	0.531	2	3.313	0.421
30	2	9.031	0.578	3	10.844	0.812
40	3	22.828	1.265	4	29.969	1.703
50	4	57.391	2.859	3	70.594	2.125
60	3	113.17	3.344	11	144.844	12.032
70	4	202.578	7.000	6	273.75	14.375
80	3	339.531	7.547	5	453.469	12.281
90	4	561.031	13.766	8	723.343	40.765

Note: presmooth = 1, postsmooth = 1, smoother = ILU, setup_Alg = dpagg, $V\_$cycle.

(b) contain zero point. So we could boldly deduce that DPAP and SAP are more beneficial for Krylov subspace iterative method. Perhaps DPAP is better than SAP since the eigenvalues of (b) contain zero point. Next, we contrast (a), (b), (c), respectively, with (d), (e), (f). We notice that the higher accurate discretization is possible to improve the solution efficiency of SLP, since the fourth-order discretization makes eigenvalues more close to one in (d), is probably good for SAP for avoiding the zero point eigenvalue, and is likely bad for DPAP because of appearing the zero point eigenvalue. As we all know, the larger the condition number of matrix is, the worse the numerical stability of solution of the corresponding matrix equation is. So the system preconditioned by the shifted Laplacian operator has worse numerical stability, which means that the small change of the right hand will bring large change of solution, since $\text{cond}_2((M_{\text{SL}})_h^{-1} A_h) = 211.1400$ is much larger. It is perhaps the reason for using multigrid or LU to approximate the shifted Laplacian operator to obtain final preconditioner, such as [4, 5, 7, 8].

Secondly, from Table 2, we could easily see that the number and time of iterative solution are both going up largely with the increasing wavenumber. The shifted Laplacian preconditioner degenerates into the Laplace preconditioner when $(\beta_1, \beta_2) = (0, 0)$. Next, we find out that the number of iterations is smallest when $(\beta_1, \beta_2) = (1, 0.3)$, but it is not best in terms of the time of iterative solution. In general, the best choice is that $(\beta_1, \beta_2) = (1, 0.5)$ by considering both aspects of the number and time of iterative solution.

Thirdly, from Table 3, we find that the fourth-order accurate discretization only affects the iterative solution a little, and it is little better than the second-order accurate discretization when the wavenumber $k = 10, 20, 40$. However, only the low wavenumbers k are computed in Table 3 because of the too long time of iterative solution or the inadequate memory for higher wavenumbers. From Table 4, we discover that the fourth-order accurate discretization does not evidently improve the efficiency of iterative solution and that the second-order accurate discretization looks more stable in the aspect of the number of iterations. The fourth-order accurate discretization improves the efficiency of iterative solution both on the sides of setup-time and solve time when the wavenumber $k = 10, 20$ but costs more time and iterations for the higher wavenumber $k = 30, \ldots, 90$. So we could conclude that the second-order accurate discretization combined with DPAP is preferable.

Finally, to compare results of Tables 2, 3, and 4, according to the number of iterations, we can easily find that Bi-CGSTAB with DPAP and second-order accurate discretization is preferable. However, presmooth = 0 is chosen in Table 3, and, when we take presmooth

= 1 for Bi-CGSTAB with SAP, the iterative steps will be smaller but still not better than Bi-CGSTAB with DPAP, with leading to the longer solve time for the added smoothing process. Following, we contrast solve time and setup time with second-order accurate discretization in Tables 2, 3, and 4. Then, we see that GMRES with SLP with $(\beta_1, \beta_2) = (-1, 0)$ need the least solve-time for wavenumber $k = 10$, Bi-CGSTAB with SAP has the least time of iterative solution for wavenumber $k = 20$, and Bi-CGSTAB with DPAP has the quickest iterative rate for wavenumber $k = 30, 40$. SAP has much more setup time than DPAP for the same wavenumber, and setup time of SAP increases more rapidly than that of DPAP. Though setup time will lead to the whole solution time longer, if there are some matrix equations with the same left matrix and the different right hands, DPAP will excel for higher wavenumber because of needing the setup process once. The results of preconditioning for wavenumber $k = 50, \ldots, 90$ are only shown in Table 4, and the reason is that the larger the wavenumber k is, the longer the solution time is for higher wavenumber. So we do not present the results for the higher wavenumber in Tables 2 and 3 due to the limited experiment condition. Next, we observe that, with the increasing wavenumber k, the number of iterative solution changes between linearly and squarely in Table 2, but it fluctuates between 4 and 7 in Table 3 and between 2 and 4 in Table 4. In other words, the number of iterative solution is not related to the wavenumber k when Bi-CGSTAB with DPAP and SAP is employed. In contrast with Bi-CGSTAB with SAP in Table 3, Bi-CGSTAB with DPAP in Table 4 makes the time of iterative solution change more slowly while the wavenumber k is increasing.

5. Conclusions

As is known to all, Helmholtz equations are difficult to solve, especially with high wavenumbers. The reason is that the high wavenumber is able to make the corresponding matrices of Helmholtz equations highly indefinite. The original Krylov subspace iterative method is not efficient to solve these problems. Preconditioning is needed to apply into Krylov subspace iterative methods in order to improve the efficiency of solution. We consider using the algebraic multigrid method to construct the preconditioner and look for a concrete AMG method which is fit for Helmholtz problems. According to numerical results and analysis in Section 4, the double pairwise aggregation is applied in the setup process of AMG and finally better results are gained. Meanwhile, the number of iterative solution is independent on the wavenumber, with the result that Helmholtz equations with high wavenumbers can be solved more efficiently in terms of time and memory. The methods that are Bi-CGSTAB with DPAP and with SAP get fewer number of iterative solution however, SAP prolongs the setup time in contrast with DPAP. The two kinds of preconditionings are obviously more efficient than the shifted Laplacian preconditioners. In view of the above, we can draw the conclusion that the double pair aggregation algorithm in the setup process of AMG is a good choice to solve Helmholtz equations especially with high wavenumbers. Though only the 2 D close-off problem with Dirichlet boundary condition is experimented in Section 4, this conclusion is also possible to generalize to Helmholtz equations with other boundary conditions.

Acknowledgments

References

[1] A. Bayliss, C. I. Goldstein, and E. Turkel, "An iterative method for the Helmholtz equation," *Journal of Computational Physics*, vol. 49, no. 3, pp. 443–457, 1983.

[2] A. L. Laird and M. B. Giles, *Preconditioned Iterative solution of the 2D Helmholtz equation, report 02/12*, Oxford Computer Laboratory, Oxford, UK, 2002.

[3] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee, "On a class of preconditioners for solving the Helmholtz equation," *Applied Numerical Mathematics*, vol. 50, no. 3-4, pp. 409–425, 2004.

[4] Y. A. Erlangga, C. Vuik, and C. W. Oosterlee, "Comparison of multigrid and incomplete LU shifted-Laplace preconditioners for the inhomogeneous Helmholtz equation," *Applied Numerical Mathematics*, vol. 56, no. 5, pp. 648–666, 2006.

[5] Y. A. Erlangga, C. W. Oosterlee, and C. Vuik, "A novel multigrid based preconditioner for heterogeneous Helmholtz problems," *SIAM Journal on Scientific Computing*, vol. 27, no. 4, pp. 1471–1492, 2006.

[6] M. B. van Gijzen, Y. A. Erlangga, and C. Vuik, "Spectral analysis of the discrete Helmholtz operator preconditioned with a shifted Laplacian," *SIAM Journal on Scientific Computing*, vol. 29, no. 5, pp. 1942–1958, 2007.

[7] N. Umetani, S. P. MacLachlan, and C. W. Oosterlee, "A multigrid-based shifted Laplacian preconditioner for a fourth-order Helmholtz discretization," *Numerical Linear Algebra with Applications*, vol. 16, no. 8, pp. 603–626, 2009.

[8] T. Airaksinen, E. Heikkola, A. Pennanen, and J. Toivanen, "An algebraic multigrid based shifted-Laplacian preconditioner for the Helmholtz equation," *Journal of Computational Physics*, vol. 226, no. 1, pp. 1196–1210, 2007.

[9] L. N. Olson and J. B. Schroder, "Smoothed aggregation for Helmholtz problems," *Numerical Linear Algebra with Applications*, vol. 17, no. 2-3, pp. 361–386, 2010.

[10] Y. Notay, "An aggregation-based algebraic multigrid method," *Electronic Transactions on Numerical Analysis*, vol. 37, pp. 123–146, 2010.

[11] Y. Notay, "Aggregation-based algebraic multilevel preconditioning," *SIAM Journal on Matrix Analysis and Applications*, vol. 27, no. 4, pp. 998–1018, 2006.

[12] I. Singer and E. Turkel, "High-order finite difference methods for the Helmholtz equation," *Computer Methods in Applied Mechanics and Engineering*, vol. 163, no. 1–4, pp. 343–358, 1998.

[13] H. A. van der Vorst, "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*, vol. 13, no. 2, pp. 631–644, 1992.

[14] C. W. Oosterlee, C. Vuik, W. A. Mulder, and R.-E. Plessix, "Shifted-laplacian preconditioners for heterogeneous helmholtz problems," *Lecture Notes in Electrical Engineering*, vol. 71, pp. 21–46, 2010.

[15] C. Wagner, "Introduction to algebraic multigrid," Course Note of an Algebraic Multigrid Course at the university of Heidelberg in the Wintersemester, 1998.