Numer. Math. (1998) 79: 231–259

Numerische
Mathematik

Electronic Edition

Domain decomposition iterative procedures for solving scalar waves in the frequency domain

Seongjai Kim

Department of Mathematics, University of Kentucky, Lexington, KY 40506-0027, USA, e-mail: skim@ms.uky.edu

Received October 30, 1995 / Revised version received January 10, 1997

Summary. The propagation of dispersive waves can be modeled relevantly in the frequency domain. A wave problem in the frequency domain is difficult to solve numerically. In addition to having a complex–valued solution, the problem is neither Hermitian symmetric nor coercive in a wide range of applications in Geophysics or Quantum–Mechanics. In this paper, we consider a parallel domain decomposition iterative procedure for solving the problem by finite differences or conforming finite element methods. The analysis includes the decomposition of the domain into either the individual elements or larger subdomains ($blocks$ of finite elements). To accelerate the speed of convergence, we introduce relaxation parameters on the subdomain interfaces and an artificial damping iteration. The convergence rate of the resulting algorithm turns out to be independent on the mesh size and the wave number. Numerical results carried out on an nCUBE2 parallel computer are presented to show the effectiveness of the method.

Mathematics Subject Classification (1991): 65N55, 65F10

1. Introduction

Wave propagation in real media is affected by attenuation and dispersion. Therefore, a realistic simulation of wave propagation phenomena should be able to reproduce these two effects. Wave problems are often formulated in an unbounded domain. These problems can be solved numerically by first truncating the given unbounded domain, imposing a suitable absorbing boundary condition (ABC) on the (artificial) boundary of the truncated

bounded domain, and then solving the resulting problem using discretization methods (e.g., finite differences, finite elements, and spectral methods).

Let $\Omega \subset \mathbb{R}^d$, $d = 2$ or 3, be a logically rectangular/cubic domain with its boundary $\Gamma = \partial\Omega$. Consider the following (complex–valued) wave problem

$$\begin{aligned} -\Delta u - p(x)^2 u + \mathrm{i} q(x)^2 u &= f(x), \; x \in \Omega, \\ \frac{\partial u}{\partial \nu} + \mathrm{i}\alpha(x) u &= 0, \qquad x \in \Gamma, \end{aligned} \tag{1}$$

where i is the imaginary unit, ν denotes the unit outward normal from Ω, and the coefficients $p(x)$, $q(x)$, and $\alpha(x)$ satisfy

$$\begin{aligned} &0 < p_0 \le p(x) \le p_1 < \infty, \\ &0 \le q_0 \le q(x) \le q_1 < \infty, \\ &\alpha = \alpha_{\mathrm{r}} - \mathrm{i}\alpha_{\mathrm{i}}, \quad \alpha_{\mathrm{r}} > 0, \;\; \alpha_{\mathrm{i}} \ge 0, \end{aligned}$$

and are sufficiently regular that the existence and uniqueness of a solution of (1) lying in $H^1(\Omega)$ are assured for reasonable f. The coefficient α is properly chosen such that the second equation of (1) represents a first–order ABC that allows normally incident waves to pass out of Ω transparently. The problem (1) models the propagation of time–harmonic waves such as electromagnetic waves, seismic waves, underwater acoustics, and inverse scattering problems.

Here we consider an example of ABC. For the electromagnetic waves in homogeneous conducting media, the coefficients are given by

$$p^2 = \frac{\omega^2}{c^2}, \quad q^2 = \frac{2\pi\sigma}{c^2\varepsilon}\omega, \tag{2}$$

where ω is the angular frequency, c is the velocity of light in a vacuum, σ is the conductivity of the medium, and ε denotes the dielectric constant. Set $b = 2\pi\sigma/(c^2\varepsilon)$. Then one can derive α [9] as

$$\begin{aligned} \alpha \;&=\; \frac{\omega}{\sqrt{2}c}(1 + (1 + \omega^{-2}b^2)^{1/2})^{1/2} - \frac{\mathrm{i}b}{\sqrt{2}c}(1 + (1 + \omega^{-2}b^2)^{1/2})^{-1/2} \\ &=: \alpha_{\mathrm{r}} - \mathrm{i}\alpha_{\mathrm{i}}. \end{aligned}$$

The wave problem (1) is difficult to solve, in particular, when

$$0 \le q \ll p. \tag{3}$$

In addition to having a complex–valued solution, it is neither Hermitian symmetric nor coercive; as a consequence, most standard iterative methods either fail to converge or converge very slowly. In acoustic wave applications, for example, we often need to simulate waves up to 50 wavelengths.

(The wavelength is $2\pi/p$ by definition.) It is known that the second–order finite difference scheme requires at least 6 to 8 grid points per wavelength for a stability reason [21]. So direct algorithms may not be employed on modern computers, because of the large problem size. In this context, domain decomposition (DD) methods are attractive for such ill–conditioned large problems; DD methods can combine iterative methods at the interface level and direct algorithms at the subdomain level.

Concerning iterative numerical solvers for (1), we refer to Bayliss et al. [1] and Freund [12] for the conjugate gradient (CG)–type algorithms, and Douglas et al. [7] for an ADI algorithm. Després [5] analyzed a DD algorithm for (1) in a differential, rather than discrete, level. The author [15, 16] studied finite difference DD methods and suggested an automatic strategy for finding efficient relaxation parameters.

The main purpose of this paper is to analyze a nonoverlapping finite element (FE) DD iterative procedure that is computationally efficient. A Robin interface boundary condition is employed on the subdomain interfaces. Note that the Robin interface boundary condition imposes the continuity of both the discrete solution u^h and the normal components of its flux, while most FE methods admit discontinuity of the normal components of the flux on the element interfaces. So it should be modified: we introduce finite differences to replace the normal derivatives so that the *modified* Robin interface boundary condition imposes the continuity of the discrete solution only. The convergence analysis is inspired by Douglas et al. [8], where the authors studied a DD algorithm for solving real–valued coercive partial differential equations by mixed methods. For DD methods different from one presented in this paper, see the review articles by Dryja and Widlund [10] and Le Tallec [22].

An outline of the paper is as follows. In Sect. 2, the existence and uniqueness of the solution of (1) and the convergence properties for the FE solutions are reviewed. Some numerical difficulties arising in applying iterative algorithms to (1) are also discussed in the section. In Sect. 3, the DD algorithm is described and the iterative procedure using the Robin interface boundary condition is illustrated on a differential level. In Sect. 4, a linear FE procedure is introduced and the corresponding parallel iteration is described. The convergence of the DD method is analyzed in Sect. 5. In Sect. 6, we consider efficient acceleration techniques such as artificial damping iteration and heuristic, automatic, and line–searched relaxation parameters. The resulting algorithm turns out to converge independently on the mesh size h and the wave number p. Numerical results carried out on an nCUBE2 are given in Sect. 7 and they show the effectiveness of the algorithm.

Throughout the paper, we use standard notation for the function spaces and their norms and inner products; e.g., $L^2(K)$ is the space of all functions

f on a domain K such that

$$\int_K |f|^2 dx < \infty,$$

and $(\cdot,\cdot)_K$ and $\|\cdot\|_{0,K}$ are the corresponding inner product and norm, respectively.

2. Preliminaries

Consider the weak formulation of (1): find $u \in V = H^1(\Omega)$ such that

(4) $(\nabla u, \nabla v)_\Omega - (\,(p^2 - \mathrm{i}q^2)u, v)_\Omega + \langle \mathrm{i}\alpha u, v\rangle_\Gamma = (f, v)_\Omega, \ \forall\, v \in V,$

where

$$(f, g)_\Omega = \int_\Omega f\overline{g}dx, \quad \langle f, g\rangle_\Gamma = \int_\Gamma f\overline{g}d\sigma.$$

It is known (eg. [9]) that (4) admits a unique solution in $H^1(\Omega)$.

Let Q_m^h be the space of the m–th order splines corresponding to $\mathcal{T}_h$, the set of regular finite elements of edge length no greater than h [4]. Given a subspace V^h $(:= V \cap Q_m^h)$, the FE solution of (4) is the function $u^h \in V^h$ such that

$$\begin{aligned}&(\nabla u^h, \nabla v)_\Omega - (\,(p^2 - \mathrm{i}q^2)u^h, v)_\Omega + \langle \mathrm{i}\alpha u^h, v\rangle_\Gamma \\ &= (f, v)_\Omega, \ \forall\, v \in V^h.\end{aligned} \tag{5}$$

Throughout the paper, we will employ tensor products of the Legendre–Gauss–Lobatto polynomials for the FE basis functions and use the Gauss–Lobatto points for the numerical integration of the terms arising in variational formulations. For example, for the bilinear FE method, the quadrature points are the four corner points of each element. See [3] for details of Legendre–Gauss–Lobatto polynomials. The algebraic system of (5) can be written as

$$A\mathbf{u}^h \equiv (A_1 + \mathrm{i}A_2)\mathbf{u}^h = \mathbf{f}, \tag{6}$$

where A_1 is assembled from $(\nabla u^h, \nabla v)_\Omega - (p^2 u^h, v)_\Omega + \langle \alpha_{\mathrm{i}} u^h, v\rangle_\Gamma$ and A_2 corresponds to $(q^2 u^h, v)_\Omega + \langle \alpha_{\mathrm{r}} u^h, v\rangle_\Gamma$. The matrices A_j, $j = 1, 2$, are symmetric and real–valued and A_2 is diagonal.

Let us first show the uniqueness (and therefore, existence) of the FE solution of (5) for $0 < q_0 \le q(x)$. Take $f \equiv 0$; the choice $v = u^h$ in (5) reads

$$(\nabla u^h, \nabla u^h)_\Omega - (\,(p^2 - \mathrm{i}q^2)u^h, u^h)_\Omega + \langle \mathrm{i}\alpha u^h, u^h\rangle_\Gamma = 0.$$

So the imaginary part (and the real part) of the above equation should be zero, i.e.,

$$(q^2 u^h, u^h)_\Omega + \langle \alpha_{\rm r} u^h, u^h \rangle_\Gamma = 0.$$

Since q and $\alpha_{\rm r}$ are positive, we have $u^h \equiv 0$ on $\overline{\Omega}$. Thus the uniqueness follows for $0 < q_0 \le q(x)$. (It should be noticed that by using the arguments in [20,17], we can prove $\mathrm{Im}(\lambda) > 0$, $\lambda \in \sigma(A)$, for $q_0 \ge 0$ and for $p^4 h$ sufficiently small, where $\sigma(A)$ is the spectrum of the matrix A. So the solvability of (5) follows for the general case.)

Define the error $\mathrm{e}^h = u - u^h$, where u is the solution of (4) and u^h is the solution of (5). Then, the error can be bounded by, see [2],

$$\|\mathrm{e}^h\|_{0,\Omega} = O(p^{m+2} h^{m+1}), \quad \|\mathrm{e}^h\|_{1,\Omega} = O(p^{m+1} h^m), \tag{7}$$

for $p^4 h$ sufficiently small and for $\|f\|_{m-1,\Omega} \le C_m \|u\|_{0,\Omega}$, where C_m is independent of p. When $q^2 = O(p)$, one can obtain (7) for small $p^2 h$ [2].

It is known for (1) and (3) that relaxation methods (e.g. Jacobi and SOR iterations) do not converge and that CG–type algorithms either converge slowly or have possible breakdowns [1, 11, 14]. The existence of a convergent nonsymmetric CG–type algorithm requires the positive definiteness of A_2, i.e. $q_0 > 0$ [11, 14].

For an iterative algorithm, a coarse grid solution can be used to obtain a better initial guess or to accelerate the convergence. Such an idea has been widely studied in terms of multi–grid (MG) methods where one uses a collection of successive coarser finite element meshes [13, 6]. There are two steps in MG methods: coarse grid correction and smoothing. The MG method is known to be efficient for solving certain coercive elliptic problems. In the coarse grid correction the low and medium frequency components of the error (corresponding to small and medium eigenvalues) are significantly reduced, while the smoothing step reduces the high frequency components of the error. This, roughly speaking, is why the MG method is efficient.

There are theoretical and computational difficulties in applying MG methods to problem (1) that is complex–valued and non–coercive; convergence analyses for MG methods are often based on coercivity inequalities. One can find a computational difficulty in the smoothing step, due to the lack of convergent relaxation algorithms and the *global property* of (1). By the global property we mean the phenomenon of an oscillatory solution over the whole domain from a single point source. One can encounter another difficulty in the coarse grid correction. Since we need to choose at least 6 to 8 grid points per wavelength [21], the coarsest grid problem is yet large for high frequency applications and it should be solved without introducing another coarser mesh. For large p, the grid size h can be chosen such that ph is 1/3 to 1/4. This choice of h is the same as choosing 18 to 25 grid points per wavelength and it is often required for accuracy reasons [7, 21].

3. Domain decomposition method

Let $\{\Omega_j,\ j=1,\cdots,M\}$ be a partition of Ω:

$$\overline{\Omega}=\bigcup_{j=1}^{M}\overline{\Omega}_j;\quad \Omega_j\cap\Omega_k=\emptyset,\quad j\neq k.$$

Assume that Ω_j are also logically rectangular/cubic regions. Let

$$\Gamma_j=\Gamma\cap\partial\Omega_j,\quad \Gamma_{jk}=\Gamma_{kj}=\partial\Omega_j\cap\partial\Omega_k.$$

Let us consider the decomposition of (1) over $\{\Omega_j\}$. In addition to requiring $u_j,\ j=1,\cdots,M$, to satisfy

$$\begin{aligned}-\Delta u_j-p(x)^2u_j+\mathrm{i}q(x)^2u_j&=f(x),\ x\in\Omega_j,\\ \frac{\partial u_j}{\partial\nu_j}+\mathrm{i}\alpha(x)u_j&=0,\qquad x\in\Gamma_j,\end{aligned}\tag{8}$$

it is necessary to impose the following consistency conditions

$$u_j=u_k,\quad \frac{\partial u_j}{\partial\nu_j}+\frac{\partial u_k}{\partial\nu_k}=0,\quad x\in\Gamma_{jk},\tag{9}$$

where ν_j is the unit outward normal from Ω_j. It is more convenient to replace (9) by the following Robin–type interface operator (see [19,5,8,15]):

$$\frac{\partial u_j}{\partial\nu_j}+\mathrm{i}\beta\,u_j=-\frac{\partial u_k}{\partial\nu_k}+\mathrm{i}\beta\,u_k,\quad x\in\Gamma_{jk},\tag{10}$$

where $\beta=\beta_{\mathrm{r}}-\mathrm{i}\beta_{\mathrm{i}}$, $\beta_{\mathrm{r}}>0$, $\beta_{\mathrm{i}}\geq 0$, is a complex–valued function on the interfaces. Test (8) against $v\in V_j\ (:=H^1(\Omega_j))$ to obtain

$$\begin{aligned}&(\nabla u_j,\nabla v)_{\Omega_j}-((p^2-\mathrm{i}q^2)u_j,v)_{\Omega_j}-\sum_k\langle\frac{\partial u_j}{\partial\nu_j},v\rangle_{\Gamma_{jk}}+\langle\mathrm{i}\alpha u_j,v\rangle_{\Gamma_j}\\ &\quad=(f,v)_{\Omega_j},\quad\forall\,v\in V_j.\end{aligned}\tag{11}$$

Using (10), we obtain the weak formulation of (8) over the partition $\{\Omega_j\}$: find $u_j\in V_j,\ j=1,\cdots,M$, such that

$$\begin{aligned}&(\nabla u_j,\nabla v)_{\Omega_j}-((p^2-\mathrm{i}q^2)u_j,v)_{\Omega_j}+\sum_k\langle\mathrm{i}\beta\,u_j,v\rangle_{\Gamma_{jk}}+\langle\mathrm{i}\alpha u_j,v\rangle_{\Gamma_j}\\ &\quad=\sum_k\langle-\frac{\partial u_k}{\partial\nu_k}+\mathrm{i}\beta u_k,v\rangle_{\Gamma_{jk}}+(f,v)_{\Omega_j},\quad\forall\,v\in V_j.\end{aligned}\tag{12}$$

The basic idea of a DD iterative method is to localize the computations to smaller subdomain problems. It is feasible to localize to each Ω_j by

evaluating the quantities in (12) related to each Ω_j at the new iterate level and those in (12) related to neighboring subdomains Ω_k at the old level. The iterative algorithm in the differential case can be defined as follows: Choose $u_j^0 \in V_j$, $j = 1, \cdots, M$, then recursively build the sequences $u_j^n \in V_j$, $n \geq 1$, by solving

$$
\begin{aligned}
&(\nabla u_j^n, \nabla v)_{\Omega_j} - ((p^2 - \mathrm{i}q^2)u_j^n, v)_{\Omega_j} + \sum_k \langle \mathrm{i}\beta\, u_j^n, v\rangle_{\Gamma_{jk}} + \langle \mathrm{i}\alpha u_j^n, v\rangle_{\Gamma_j} \\
&\qquad = \sum_k \langle -\frac{\partial u_k^{n-1}}{\partial \nu_k} + \mathrm{i}\beta u_k^{n-1}, v\rangle_{\Gamma_{jk}} + (f, v)_{\Omega_j}, \quad \forall\, v \in V_j.
\end{aligned} \tag{13}
$$

Després [5], indebted to Lions [19], analyzed the above algorithm on a differential, rather than discrete, level. Our objectives in this paper are to analyze a parallel iterative algorithm for (12) on a discrete level and to give remedies that can help improve the performance of the algorithm.

4. Bilinear finite element method

Consider the bilinear (conforming) FE methods for the 2–dimensional problem. The analysis to be presented can be applied to higher–order FE methods in 2– and 3–dimensional spaces [18]. Let V^h be a FE space of V with quadrilateral partitions $\mathcal{T}_h$ not crossing the subdomain interfaces and of the maximum edge length h. The local FE subspaces are defined by $V_j^h = \{v|_{\Omega_j} : v \in V^h\}$. Our analysis below includes the cases in which $\{\Omega_j\}$ is a partition of Ω into either individual elements or larger subdomains ($blocks$ of elements).

From the view point of (11) and (10), a finite element DD method might be defined as follows: find $u_j^h \in V_j^h$, $j = 1, \cdots, M$, satisfying

$$
\begin{aligned}
&(\nabla u_j^h, \nabla v)_{\Omega_j} - ((p^2 - \mathrm{i}q^2)u_j^h, v)_{\Omega_j} - \sum_k \langle \frac{\partial u_j^h}{\partial \nu_j}, v\rangle_{\Gamma_{jk}} + \langle \mathrm{i}\alpha u_j^h, v\rangle_{\Gamma_j} \\
&\qquad = (f, v)_{\Omega_j}, \quad \forall\, v \in V_j^h, \\
&\frac{\partial u_j^h}{\partial \nu_j} + \mathrm{i}\beta u_j^h = -\frac{\partial u_k^h}{\partial \nu_k} + \mathrm{i}\beta u_k^h, \quad x \in \Gamma_{jk}.
\end{aligned} \tag{14}
$$

There is a technical difficulty in the use of the Robin interface boundary condition for nonoverlapping DD methods that incorporate conforming FE methods. Since the Robin boundary condition imposes, on the subdomain interfaces, the continuity of both the discrete solution u^h and the normal components of its flux, there will be a flux conservation error; i.e., (9) would not be satisfied unless the discrete solution $u^h \in V^h$ is a linear function over the domain Ω, a totally uninteresting case. So, we should replace $\partial u_j^h/\partial \nu_j$

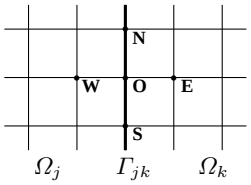

Fig. 1. The five point stencil

in (14) by another variable that may be, for example, Lagrange multipliers or some other approximations. (Lagrange multipliers on element interfaces, in general, cannot be incorporate in conforming FE methods; a nonconforming FE methods should be introduced. We will not attempt to give the details in this direction.)

We will replace the normal derivatives by finite differences. Assume that the domain Ω is a union of nonoverlapping rectangular regions Ω_j whose interfaces are parallel to one of the coordinate axes and that the finite elements are uniform of size h. Let $\partial_{\mathbf{b},jk}u_j^h$ and $\partial_{\mathbf{f},jk}u_j^h$ be the backward and forward differences for $\partial u_j^h/\partial\nu_j$ on Γ_{jk}, respectively. At the nodal point $\mathbf{o} \in \Gamma_{jk}$ (see Fig. 1), they are defined as follows:

$$\partial_{\mathbf{b},jk}u_j^h(\mathbf{o}) = (u_j^h(\mathbf{o}) - u_j^h(\mathbf{w}))/h,\ \ \partial_{\mathbf{f},jk}u_j^h(\mathbf{o}) = (u_j^h(\mathbf{E}) - u_j^h(\mathbf{o}))/h,$$
$$\partial_{\mathbf{b},kj}u_k^h(\mathbf{o}) = (u_k^h(\mathbf{o}) - u_k^h(\mathbf{E}))/h,\ \ \partial_{\mathbf{f},kj}u_k^h(\mathbf{o}) = (u_k^h(\mathbf{w}) - u_k^h(\mathbf{o}))/h.$$

Here an exterior bordering of the nonoverlapping subdomains is assumed. Note that $\partial_{\mathbf{f},jk}u_j^h(\mathbf{o})$ may not be the same as $\partial_{\mathbf{b},jk}u_j^h(\mathbf{o})$. However, a DD method which solves the original discrete problem should satisfy

$$u_j^h = u_k^h \quad \text{and} \quad \partial_{\mathbf{f},jk}u_j^h = -\partial_{\mathbf{b},kj}u_k^h.$$

Define the centered difference $\partial_{\mathbf{c},jk}u_j^h$ on Γ_{jk} as

$$\partial_{\mathbf{c},jk}u_j^h(\mathbf{o}) = (\partial_{\mathbf{b},jk}u_j^h(\mathbf{o}) + \partial_{\mathbf{f},jk}u_j^h(\mathbf{o}))/2.$$

A nonoverlapping DD method with an exterior bordering of the subdomains can be considered as a generalized Schwarz alternating method with minimum overlap [23].

Consider the following problem: find $\{u_j^h \in V_j^h :\ j = 1, \cdots, M\}$ such that

$$\begin{aligned} &(\nabla u_j^h, \nabla v)_{\Omega_j} - ((p^2 - \mathrm{i}q^2)u_j^h, v)_{\Omega_j} + \sum_k \langle -\partial_{\mathbf{c},jk}u_j^h, v\rangle_{\Gamma_{jk}} \\ &\qquad + \langle \mathrm{i}\alpha u_j^h, v\rangle_{\Gamma_j} = (f, v)_{\Omega_j}, \quad \forall\, v \in V_j^h, \\ &u_j^h = u_k^h, \quad \text{on } \Gamma_{jk}, \\ &\partial_{\mathbf{f},jk}u_j^h = -\partial_{\mathbf{b},kj}u_k^h, \quad \text{on } \Gamma_{jk}. \end{aligned} \tag{15}$$

From the similarity between the bilinear FE method and the 5–point finite difference scheme, it can be easily verified that problem (15) solves the original discrete problem (5): for a basis function v whose support is in the closure of a subdomain, the algebraic equation of (15) is identical to that of the 5–point finite difference scheme. Let $v \in V^h$ be the basis function such that $v(\mathbf{o}) = 1$, where $\mathbf{o}$ is a nodal point on the interface Γ_{jk} (see Fig. 1). Then (15) reduces

$$\frac{1}{2}\left\{4u_j^h(\mathbf{o}) - 2u_j^h(\mathbf{w}) - u_j^h(\mathbf{s}) - u_j^h(\mathbf{N})\right\} - \frac{1}{2}(p^2 - \mathrm{i}q^2)h^2\, u_j^h(\mathbf{o}) \\ -\frac{1}{2}\left\{u_j^h(\mathbf{E}) - u_j^h(\mathbf{w})\right\} \;=\; \frac{1}{2}fh^2,$$

where p, q, and f are evaluated at the point $\mathbf{o}$; the above equation again is the 5–point finite difference scheme.

Now, we present the DD iterative procedure which solves (15): for given $u_j^{h,0} \in V_j^h$, $j = 1, \cdots, M$, build $u_j^{h,n} \in V_j^h$, $n \geq 1$, recursively by solving

$$\begin{aligned} &(\nabla u_j^{h,n}, \nabla v)_{\Omega_j} - ((p^2 - \mathrm{i}q^2)u_j^{h,n}, v)_{\Omega_j} + \sum_k \langle -\partial_{\mathbf{c},jk} u_j^{h,n}, v\rangle_{\Gamma_{jk}} \\ &\qquad + \langle \mathrm{i}\alpha u_j^{h,n}, v\rangle_{\Gamma_j} = (f, v)_{\Omega_j}, \quad \forall\, v \in V_j^h, \\ &\partial_{\mathbf{f},jk} u_j^{h,n} + \mathrm{i}\beta u_j^{h,n} = -\partial_{\mathbf{b},kj} u_k^{h,n-1} + \mathrm{i}\beta u_k^{h,n-1}, \quad \text{on } \Gamma_{jk}. \end{aligned} \tag{16}$$

Note that $\partial_{\mathbf{c},jk} u_j^{h,n} = (\partial_{\mathbf{b},jk} u_j^{h,n} + \partial_{\mathbf{f},jk} u_j^{h,n})/2$ and that the second equation of (16) is used as the boundary condition on the subdomain interfaces. When $\beta = -\mathrm{i}h^{-1}$, algorithm (16) can be viewed as the standard Schwarz alternating method with a minimum overlap, communicating Dirichlet data.

Algorithm (16) can be rewritten as follows: choose initial guess $u_j^{h,0} \in V_j^h$, $j = 1, \cdots, M$, then find $u_j^{h,n} \in V_j^h$, $n \geq 1$, satisfying

$$\begin{aligned} &(\nabla u_j^{h,n}, \nabla v)_{\Omega_j} - ((p^2 - \mathrm{i}q^2)u_j^{h,n}, v)_{\Omega_j} + \langle \mathrm{i}\alpha u_j^{h,n}, v\rangle_{\Gamma_j} \\ &\qquad + \sum_k \langle \frac{1}{2}(-\partial_{\mathbf{b},jk} u_j^{h,n} + \mathrm{i}\beta u_j^{h,n}), v\rangle_{\Gamma_{jk}} \\ &= \sum_k \langle \frac{1}{2}(-\partial_{\mathbf{b},kj} u_k^{h,n-1} + \mathrm{i}\beta u_k^{h,n-1}), v\rangle_{\Gamma_{jk}} + (f, v)_{\Omega_j}, \quad \forall\, v \in V_j^h. \end{aligned} \tag{17}$$

Using finite difference methods, one can define an iterative procedure equivalent to the above algorithm. Let $\Delta_h u_j$ be the centered 5–point difference approximation of Δu_j, and $\partial_{\mathbf{c},j} u_j$ the centered difference for $\partial u_j/\partial \nu_j$ on the boundary Γ_j. Then, it is easy to check that the iteration matrix of (16)

is exactly the same as that of the following system:

$$
\begin{aligned}
-\Delta_h u_j^n - p^2 u_j^n + \mathrm{i}q^2 u_j^n &= f, && x \in \Omega_j,\\
\partial_{\mathbf{c},j} u_j^n + \mathrm{i}\alpha u_j^n &= 0, && x \in \Gamma_j,\\
\partial_{\mathbf{f},jk} u_j^n + \mathrm{i}\beta\, u_j^n &= -\partial_{\mathbf{b},kj} u_k^{n-1} + \mathrm{i}\beta\, u_k^{n-1}, && x \in \Gamma_{jk}.
\end{aligned} \tag{18}
$$

The finite difference $\partial_{\mathbf{c},jk} u_j^h$ in (15) is in fact an approximation of the average of normal derivatives on the subdomain interfaces:

$$
\partial_{\mathbf{c},jk} u_j^h \;=\; \frac{1}{2}\left(\partial_{\mathbf{b},jk} u_j^h + \partial_{\mathbf{f},jk} u_j^h\right) \;\cong\; \frac{1}{2}\left(\frac{\partial u_j^h}{\partial \nu_j}\bigg|_{\overline{\Omega_j}} + \frac{\partial u_j^h}{\partial \nu_j}\bigg|_{\overline{\Omega_k}} \right). \tag{19}
$$

When one uses the average of normal derivatives instead of its approximation in (15), the assumption that the interfaces are parallel to one of the coordinate axes can be eliminated. But still the basis functions in V^h should be bilinear splines to guarantee the equivalence between (5) and (15).

An application to higher–order FE methods requires the introduction of overlapping subdomains $\{\widetilde{\Omega}_j\}$ that are expansions of $\{\Omega_j\}$ by a strip of finite elements along the interfaces $\Gamma_{jk}\,(= \partial\Omega_j \cap \partial\Omega_j)$. The overlapping DD algorithm easily satisfies the consistency conditions, which is one of the advantages over the nonoverlapping ones. We can accelerate its convergence by introducing relaxation parameters on the interfaces of the original nonoverlapping subdomains and the computations of subdomain problems can be localized to each Ω_j by evaluating the quantities related to $\widetilde{\Omega}_j \setminus \Omega_j$ (the outside of Ω_j) at the old level [18].

5. Convergence analysis

In this section, we show the convergence of algorithm (16) for $q \geq q_0 > 0$. The first half of the analysis in this section is in principle the same as the arguments in [19], [5], and [8]. Let

$$
\mathrm{e}_j^n = u_j^h - u_j^{h,n}, \tag{20}
$$

where $\{u_j^h\}$ is the global solution of (15) over the partition $\{\Omega_j\}$ and $\{u_j^{h,n}\}$ are the iterates of (16). Then, from (15) and (16), one has

$$
\begin{aligned}
&(\nabla \mathrm{e}_j^n, \nabla v)_{\Omega_j} - (\,(p^2 - \mathrm{i}q^2)\mathrm{e}_j^n, v)_{\Omega_j}\\
&\qquad + \sum_k \langle -\partial_{\mathbf{c},jk} e_j^n, v\rangle_{\Gamma_{jk}} + \mathrm{i}\langle \alpha \mathrm{e}_j^n, v\rangle_{\Gamma_j} = 0, \quad \forall\, v \in V_j^h.
\end{aligned} \tag{21}
$$

$$
\partial_{\mathbf{f},jk} e_j^n + \mathrm{i}\beta e_j^n = -\partial_{\mathbf{b},kj} e_k^{n-1} + \mathrm{i}\beta e_k^{n-1}, \quad \text{on } \Gamma_{jk}. \tag{22}
$$

For a simple presentation, we assume that $\beta = \beta_{\rm r} - {\rm i}\beta_{\rm i}$ is a (complex–valued) constant over $\{\Omega_j\}$. Choose $v = {\rm e}_j^n$ in (21). Then,

$$
\begin{aligned}
&(\nabla {\rm e}_j^n, \nabla {\rm e}_j^n)_{\Omega_j} - (\,(p^2 - {\rm i}q^2){\rm e}_j^n, {\rm e}_j^n)_{\Omega_j} \\
&\qquad + \sum_k \langle -\partial_{\mathbf{c},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}} + {\rm i}\langle \alpha {\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_j} = 0,
\end{aligned} \tag{23}
$$

so both the real and imaginary parts of (23) should be zero:

$$
\begin{aligned}
&(\nabla {\rm e}_j^n, \nabla {\rm e}_j^n)_{\Omega_j} - (p^2 {\rm e}_j^n, {\rm e}_j^n)_{\Omega_j} \\
&+{\rm Re}(\textstyle\sum_k \langle -\partial_{\mathbf{c},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}}) + \langle \alpha_{\rm i}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_j} = \ 0, \\
&(q^2 {\rm e}_j^n, {\rm e}_j^n)_{\Omega_j} + {\rm Im}(\sum_k \langle -\partial_{\mathbf{c},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}}) + \langle \alpha_{\rm r}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_j} = \ 0.
\end{aligned} \tag{24}
$$

Using (24), and with $|\cdot|_{0,\Gamma_{jk}}$ denoting the L^2–norm on Γ_{jk}, one has

$$
\begin{aligned}
&\sum_k |\partial_{\mathbf{f},jk}{\rm e}_j^n + {\rm i}\beta {\rm e}_j^n|^2_{0,\Gamma_{jk}} \\
&= \sum_k \{|\partial_{\mathbf{f},jk}{\rm e}_j^n|^2_{0,\Gamma_{jk}} + |\beta|^2 |{\rm e}_j^n|^2_{0,\Gamma_{jk}}\} \\
&\quad +2\beta_{\rm r}\,{\rm Im}(\sum_k \langle \partial_{\mathbf{f},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}}) + 2\beta_{\rm i}\,{\rm Re}(\sum_k \langle \partial_{\mathbf{f},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}}) \\
&= \sum_k \{|\partial_{\mathbf{f},jk}{\rm e}_j^n|^2_{0,\Gamma_{jk}} + |\beta|^2 |{\rm e}_j^n|^2_{0,\Gamma_{jk}}\} \\
&\quad +4\beta_{\rm r}\{(q^2 {\rm e}_j^n, {\rm e}_j^n)_{\Omega_j} + \langle \alpha_{\rm r}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_j}\} \\
&\quad +4\beta_{\rm i}\{(\nabla {\rm e}_j^n, \nabla {\rm e}_j^n)_{\Omega_j} - (p^2 {\rm e}_j^n, {\rm e}_j^n)_{\Omega_j} + \langle \alpha_{\rm i}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_j}\} \\
&\quad -\{2\beta_{\rm r}{\rm Im}(\sum_k \langle \partial_{\mathbf{b},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}}) \\
&\quad +2\beta_{\rm i}{\rm Re}(\textstyle\sum_k \langle \partial_{\mathbf{b},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}})\},
\end{aligned} \tag{25}
$$

$$
\begin{aligned}
&\sum_k |-\partial_{\mathbf{b},jk}{\rm e}_j^n + {\rm i}\beta {\rm e}_j^n|^2_{0,\Gamma_{jk}} \\
&= \sum_k \{|\partial_{\mathbf{b},jk}{\rm e}_j^n|^2_{0,\Gamma_{jk}} + |\beta|^2 |{\rm e}_j^n|^2_{0,\Gamma_{jk}}\} \\
&\quad -\{2\beta_{\rm r}\,{\rm Im}(\sum_k \langle \partial_{\mathbf{b},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}}) + 2\beta_{\rm i}\,{\rm Re}(\sum_k \langle \partial_{\mathbf{b},jk}{\rm e}_j^n, {\rm e}_j^n\rangle_{\Gamma_{jk}})\}.
\end{aligned} \tag{26}
$$

Set

$$
E(e) = \sum_j \sum_k |\partial_{\mathbf{f},jk} e_j + {\rm i}\beta e_j|^2_{0,\Gamma_{jk}}, \tag{27}
$$

and let $E^n = E(\mathrm{e}^n)$. Then, from (22) and (25)–(26),

$$
\begin{aligned}
E^n &= \sum_j \sum_k |\partial_{\mathbf{f},jk} \mathrm{e}_j^n + \mathrm{i}\beta \mathrm{e}_j^n|^2_{0,\Gamma_{jk}} \\
&= \sum_j \sum_k |-\partial_{\mathbf{b},kj} \mathrm{e}_k^{n-1} + \mathrm{i}\beta \mathrm{e}_k^{n-1}|^2_{0,\Gamma_{jk}} \\
&= E^{n-1} - R^{n-1},
\end{aligned} \tag{28}
$$

where

$$
\begin{aligned}
R^n &\equiv R(\mathrm{e}^n) \\
&= \sum_j \sum_k |\partial_{\mathbf{f},jk} e_j^n|^2_{0,\Gamma_{jk}} - \sum_j \sum_k |\partial_{\mathbf{b},jk} e_j^n|^2_{0,\Gamma_{jk}} \\
&\quad + 4\beta_{\mathrm{r}} \sum_j \{(q^2 \mathrm{e}_j^n, \mathrm{e}_j^n)_{\Omega_j} + \langle \alpha_{\mathrm{r}} \mathrm{e}_j^n, \mathrm{e}_j^n \rangle_{\Gamma_j}\} \\
&\quad + 4\beta_{\mathrm{i}} \sum_j \{(\nabla \mathrm{e}_j^n, \nabla \mathrm{e}_j^n)_{\Omega_j} - (p^2 \mathrm{e}_j^n, \mathrm{e}_j^n)_{\Omega_j} + \langle \alpha_{\mathrm{i}} \mathrm{e}_j^n, \mathrm{e}_j^n \rangle_{\Gamma_j}\}.
\end{aligned}
$$

We will choose $\beta = \beta_{\mathrm{r}} - \mathrm{i}\beta_{\mathrm{i}}$ such that R^n, $n \geq 0$, are nonnegative. There is a positive constant C_1 independent on h such that

$$
\sum_j \sum_k |\partial_{\mathbf{b},jk} e_j^n|^2_{0,\Gamma_{jk}} \;\leq\; C_1 h^{-1} \sum_j (\nabla \mathrm{e}_j^n, \nabla \mathrm{e}_j^n)_{\Omega_j},
$$

and

$$
(p^2 \mathrm{e}_j^n, \mathrm{e}_j^n)_{\Omega_j} \;\leq\; \frac{p_1^2}{q_0^2} (q^2 \mathrm{e}_j^n, \mathrm{e}_j^n)_{\Omega_j}.
$$

It is not difficult to estimate C_1 explicitly for a given FE method and subdomain geometry. (Let H be the subdomain size. For $H = O(h)$, we can see $C_1 = 2$.) Let

$$
\beta_{\mathrm{i}} \geq \frac{C_1 h^{-1}}{4} \quad \text{and} \quad \beta_{\mathrm{r}} \geq \beta_{\mathrm{i}} \frac{p_1^2}{q_0^2}. \tag{29}
$$

Then

$$
\begin{aligned}
\sum_j \sum_k |\partial_{\mathbf{b},jk} e_j^n|^2_{0,\Gamma_{jk}} &\leq 4\beta_{\mathrm{i}} \sum_j (\nabla \mathrm{e}_j^n, \nabla \mathrm{e}_j^n)_{\Omega_j}, \\
4\beta_{\mathrm{i}} \sum_j (p^2 \mathrm{e}_j^n, \mathrm{e}_j^n)_{\Omega_j} &\leq 4\beta_{\mathrm{r}} \sum_j (q^2 \mathrm{e}_j^n, \mathrm{e}_j^n)_{\Omega_j}.
\end{aligned} \tag{30}
$$

The inequalities in (30) imply that R^n, $n \geq 0$, are nonnegative, i.e., $\{E^n\}$ is a decreasing sequence of nonnegative numbers. So, from (28), we have

$$
\sum_{n=0}^{\infty} R^n = \lim_{k\to\infty} \sum_{n=0}^{k} (E^n - E^{n+1}) = E^0 - \lim_{k\to\infty} E^{k+1} \;\leq\; E^0 \;<\; \infty,
$$

which implies $\lim_{n\to\infty} R^n \to 0$. Since R^n has two negative terms, the choice of β in (29) may not guarantee the convergence of e^n to zero, unless at least one of the inequalities in (29) is strict: consider $\beta = \beta_\mathrm{r} - \mathrm{i}\beta_\mathrm{i}$ such that

$$\frac{C_1 h^{-1}}{4} \le \beta_\mathrm{i} < \beta_\mathrm{r} \frac{q_0^2}{p_1^2}. \tag{31}$$

Then we have

$$\sum_{n=0}^{\infty} \sum_j (q^2 \mathrm{e}_j^n, \mathrm{e}_j^n)_{\Omega_j} = \sum_{n=0}^{\infty} \|q\mathrm{e}^n\|_{0,\Omega}^2 < \infty,$$

and therefore

$$\mathrm{e}^n \to 0, \quad \text{in } L^2(\Omega) \text{ as } n \to \infty. \tag{32}$$

We have completed the convergence proof of algorithm (16), as stated in the following theorem.

Theorem 1. *Let $q \ge q_0 > 0$ and β satisfy* (31)*. Then the iterates $\{u_j^{h,n}\} \in V_j^h$ of algorithm* (16) *converge to the solution $\{u_j^h\}$ of* (15) *in the following sense:*

$$u_j^{h,n} \ \to \ u_j^h = u^*|_{\Omega_j} \ \text{ in } \ L^2(\Omega_j), \quad j = 1, \cdots, M,$$

where $u^* \in V^h$ is the original discrete solution of (5)*.*

Now, let $\mathcal{A}$ be the affine mapping from V^h to itself such that, for any $\theta \in V^h$, $e \equiv \mathcal{A}(\theta)$ is the solution for the following equation:

$$\begin{aligned} &(\nabla e_j, \nabla v)_{\Omega_j} - ((p^2 - \mathrm{i}q^2) e_j, v)_{\Omega_j} + \sum_k \langle \frac{1}{2}(-\partial_{\mathbf{b},jk} e_j + \mathrm{i}\beta\, e_j), v\rangle_{\Gamma_{jk}} \\ &+ \langle \mathrm{i}\alpha\, e_j, v\rangle_{\Gamma_j} = \sum_k \langle \frac{1}{2}(-\partial_{\mathbf{b},kj}\theta_k + \mathrm{i}\beta\theta_k), v\rangle_{\Gamma_{jk}}, \quad \forall\, v \in V_j^h. \end{aligned} \tag{33}$$

We denote the spectral radius of $\mathcal{A}$ by $\rho(\mathcal{A})$.

Theorem 2. *Let $q \ge q_0 > 0$ and β satisfy* (31)*. Then*

$$\rho(\mathcal{A}) < 1. \tag{34}$$

Thus, the iterative procedure (16) *is convergent.*

Proof. Let $\{\gamma, u\}$ be an eigenvalue–eigenvector pair of $\mathcal{A}$; i.e., $\mathcal{A}(u) = \gamma\, u$. Our objective is to show that $|\gamma| < 1$. It follows from (27)–(28) that

$$\begin{aligned} E(\mathcal{A}(u)) &= |\gamma|^2 E(u), \\ E(\mathcal{A}(u)) &= E(u) - R(u). \end{aligned}$$

Combining the two equations above yields

$$|\gamma|^2 = (\, E(u) - R(u)\,) \,/\, E(u), \tag{35}$$

which implies that $|\gamma| \le 1$. Equality holds if and only if $R(u) = 0$, so that $u \equiv 0$. This contradicts the fact that the eigenvectors are non–trivial, which completes the proof. □

It seems that the positivity of q ($q \ge q_0 > 0$) is essential for algorithm (16) to converge. For problems in which q is small or zero, one can employ the artificial damping iteration presented in Sect. 6.1.

Our next objective is to analyze the asymptotic convergence rate of (16). Choose $\beta = \beta_{\rm r} - {\rm i}\beta_{\rm i}$ as

$$\beta_{\rm i} = \frac{C_1 h^{-1}}{4}, \quad \beta_{\rm r} = \kappa\, \beta_{\rm i} \frac{p_1^2}{q_0^2}, \tag{36}$$

where $\kappa > 1$. Then

$$|\beta|^2 \;\le\; S\beta_{\rm r}^2, \quad \text{where } \; S = 1 + \frac{q_0^4}{p_1^4}. \tag{37}$$

(One can assume that $S \le 2$ in practice.) Let

$$\begin{aligned}
E_j(u) &= \sum_k |\partial_{\mathbf{f},jk} u_j + {\rm i}\beta u_j|^2_{0,\Gamma_{jk}}, \\
R_j(u) &= \sum_k |\partial_{\mathbf{f},jk} u_j|^2_{0,\Gamma_{jk}} - \sum_k |\partial_{\mathbf{b},jk} u_j|^2_{0,\Gamma_{jk}} \\
&\quad + 4\beta_{\rm r}\{(q^2 u_j, u_j)_{\Omega_j} + \langle \alpha_{\rm r} u_j, u_j\rangle_{\Gamma_j}\} \\
&\quad + 4\beta_{\rm i}\{(\nabla u_j, \nabla u_j)_{\Omega_j} - (p^2 u_j, u_j)_{\Omega_j} + \langle \alpha_{\rm i} u_j, u_j\rangle_{\Gamma_j}\}, \\
R_j'(u) &= \sum_k |\partial_{\mathbf{f},jk} u_j|^2_{0,\Gamma_{jk}} + 4\beta_{\rm r}\{(q^2 u_j, u_j)_{\Omega_j} + \langle \alpha_{\rm r} u_j, u_j\rangle_{\Gamma_j}\}.
\end{aligned}$$

Lemma 3. *Let β is chosen as in* (36). *Then*

$$R_j'(u) \;\le\; \frac{\kappa}{\kappa - 1}\, R_j(u). \tag{38}$$

Proof. Due to the first inequality of (30), it suffices to show that

$$4\beta_{\rm r}(q^2 u_j, u_j)_{\Omega_j} \le \frac{\kappa}{\kappa - 1} \left\{ 4\beta_{\rm r}(q^2 u_j, u_j)_{\Omega_j} - 4\beta_{\rm i}(p^2 u_j, u_j)_{\Omega_j} \right\},$$

or, equivalently,

$$(\kappa - 1)\frac{p_1^2}{q_0^2}\,(q^2 u_j, u_j)_{\Omega_j} \;\le\; \kappa \frac{p_1^2}{q_0^2}\,(q^2 u_j, u_j)_{\Omega_j} - (p^2 u_j, u_j)_{\Omega_j}.$$

However, the above inequality holds clearly for $\kappa > 1$. □

It is known that there is a positive constant C_2, independent on h, such that

$$\sum_k |u_j|^2_{0,\Gamma_{jk}} \le C_2 h^{-1} \| u_j \|^2_{0,\Omega_j}, \quad \forall\, u_j \in V_j^h, \quad j = 1, 2, \cdots, M. \tag{39}$$

(For $H = O(h)$, $C_2 = 4$.) From (37)–(39), we have

$$\begin{aligned} E_j(u) &\le 2 \sum_k |\partial_{\mathbf{f},jk} u_j|^2_{0,\Gamma_{jk}} + 2|\beta|^2 \sum_k |u_j|^2_{0,\Gamma_{jk}} \\ &\le \left(2 + SC_2 h^{-1} \frac{\beta_{\mathrm{r}}}{2q_0^2}\right) R_j'(u) \\ &\le g(\kappa)\, R_j(u), \end{aligned} \tag{40}$$

where

$$g(\kappa) = \frac{\kappa}{\kappa - 1} \left(2 + SC_1 C_2 p_1^2 h^{-2} \frac{\kappa}{8 q_0^4}\right).$$

So, from (35) and (40),

$$\max_{\gamma \in \sigma(\mathcal{A})} |\gamma|^2 \;\le\; 1 - \frac{1}{g(\kappa)},$$

and therefore

$$\max_{\gamma \in \sigma(\mathcal{A})} |\gamma| \;\le\; 1 - \frac{1}{2g(\kappa)}. \tag{41}$$

It is not difficult to check that $g(\kappa)$ can be minimized when

$$\kappa = 1 + \left(1 + \frac{16 h^2 q_0^4}{SC_1 C_2 p_1^2}\right)^{1/2}. \tag{42}$$

After plugging (42) into (41), one can get the following result.

Theorem 4. *Let $H = O(h)$ and β be chosen as in (36) and (42). Then the spectral radius of the iteration matrix of (16) is minimized and bounded as*

$$\rho(\mathcal{A}) \;\le\; 1 - \frac{C_3 h^2 q_0^4}{p_1^2 + C_4 h^2 q_0^4}, \tag{43}$$

for some $C_3 > 0$ and $C_4 > 0$ independent of h, p, and q.

For more interesting applications, q is in the order of $p/10$ and the grid size h is chosen such that $ph \approx 1/2 \sim 1/4$. So, in practice, (43) reads

$$\rho(\mathcal{A}) \;\le\; 1 - C_5 \frac{q_0^4}{p_1^2}\, h^2. \tag{44}$$

Now, let us consider real–valued parameters. Let

$$\beta \;=\; \beta_{\rm r} \;>\; \frac{C_6}{q_0^2}\, h^{-3}, \tag{45}$$

for some $C_6 > 0$. (For $H = O(h)$, $C_6 = 4$.) Then, using the same argument as in this section, one can establish an analogue of Theorem 4: for

$$\beta \;=\; \beta_{\rm r} \;=\; \kappa \frac{C_6}{q_0^2}\, h^{-3}, \quad \kappa = 1 + \left(1 + \frac{4 q_0^4 h^4}{C_6 C_2}\right)^{1/2}, \tag{46}$$

the spectral radius of the iteration matrix of (16) is minimized and bounded as

$$\rho(\mathcal{A}) \;\leq\; 1 - C_7\, q_0^4 h^4, \tag{47}$$

for some $C_7 > 0$ independent of h, p, and q.

6. Acceleration techniques

This section deals with the case: q is zero or $Q\ (:= p^2/q^2)$ is large, where Q is the *quality factor* by definition. The convergence analysis in the previous section indicates that algorithm (16) may not be convergent in this case. In fact, there has been no efficient iterative algorithm for the Helmholtz problem of a large real–valued wave number. As remedies, we consider an artificial damping iteration and a strategy for finding efficient relaxation parameters β.

6.1. Artificial damping iteration

First consider the following:

Lemma 5. *Let* $\lambda \in \sigma(A)$. *Then* $\mathrm{Im}(\lambda) \geq 0$ *for* $q \geq q_0 \geq 0$.

Proof. Let $\{\lambda, \mathbf{y}\}$ be an eigenvalue–eigenvector pair of A, i.e.,

$$(A - \lambda I)\, \mathbf{y} = 0.$$

Suppose $\lambda_{\rm i} < 0$, where $\lambda = \lambda_{\rm r} + {\rm i}\lambda_{\rm i}$. We would like to derive a contradiction by showing that the eigenvector is trivial. Consider the following problem

$$\begin{aligned} &(\nabla u^h, \nabla v)_\Omega - ((p^2 - {\rm i}q^2) u^h, v)_\Omega + \langle {\rm i}\alpha u^h, v\rangle_\Gamma - (\,\xi\, u^h, v)_\Omega = 0, \\ &\qquad v \in V^h, \end{aligned} \tag{48}$$

where $\xi = \xi_{\rm r} + {\rm i}\xi_{\rm i}$. Then, clearly the eigenvector $\mathbf{y}$ is the solution of (48) for a properly chosen ξ with $\xi_{\rm i} < 0$. Let $v = u^h$ in (48). Then, the imaginary part of (48) reads

$$(q^2 u^h, u^h)_\Omega + \langle \alpha_{\rm r} u^h, u^h\rangle_\Gamma + (\mathrm{Im}\,(-\xi)\; u^h, u^h)_\Omega = 0. \tag{49}$$

Since each term of the left side of the above equation is nonnegative and the coefficients $\alpha_{\rm r}$ and Im $(-\xi)$ are positive, it follows $u^h \equiv 0$ on $\overline{\Omega}$. This implies $\mathbf{y} = 0$ which contradicts the fact that the eigenvector is nonzero. Thus, we have shown that Im $(\lambda) \geq 0$. $\square$

Motivated by the sensitivity of (44) to q and Lemma 5, we introduce the *artificial damping iteration* (ARTDI) as follows: for given initial guess $u^{h,0} \in V^h$, find recursively $u^{h,\ell} \in V^h$, $\ell \geq 1$, such that

$$\begin{aligned} &\mathcal{L}(u^{h,\ell}, v; \Omega) + (\mathrm{i}\eta^2\, u^{h,\ell}, v)_\Omega = (f, v)_\Omega + (\mathrm{i}\eta^2\, u^{h,\ell-1}, v)_\Omega, \\ &\forall\, v \in V^h, \end{aligned} \tag{50}$$

where $\eta > 0$ and, for $K \subset \Omega$,

$$\mathcal{L}(u, v; K) = (\nabla u, \nabla v)_K - (\,(p^2 - \mathrm{i}q^2)u, v\,)_K + \langle \mathrm{i}\alpha u, v\rangle_{\partial K \cap \Gamma}\,.$$

Using notations as in (6), we can rewrite (50) as

$$(A + \mathrm{i}D)\mathbf{u}^{h,\ell} = \mathrm{i}D\mathbf{u}^{h,\ell-1} + \mathbf{f}, \tag{51}$$

where D is a positive definite diagonal matrix. We may choose $\eta(x)$ such that $D = \mathrm{diag}\{d\}$, for some constant $d > 0$. Let $\mathbf{u}^h$ be the solution vector in (6) and

$$\mathbf{e}^\ell = \mathbf{u}^h - \mathbf{u}^{h,\ell}.$$

Then, from (6) and (51), one has

$$(A + \mathrm{i}dI)\,\mathbf{e}^\ell = \mathrm{i}d\mathbf{e}^{\ell-1}. \tag{52}$$

So,

$$\|\mathbf{e}^\ell\| \;\leq\; \gamma \cdot \|\mathbf{e}^{\ell-1}\|, \quad \gamma = \max_{\lambda_k \in \sigma(A)} \left| \frac{id}{\lambda_k + \mathrm{i}d} \right|. \tag{53}$$

Since A is implicitly assumed to be nonsingular, we can see from Lemma 5 that $\gamma < 1$ for all $d > 0$ and therefore (50) converges.

In each step, algorithm (50) solves a damped problem of the same size as the original problem, but it is better–conditioned. We will solve the damped problem iteratively and incompletely (by using (16)) as in the following

algorithm:

$$
\begin{array}{l}
\text{select } \eta \text{ and } n_* ; \\
\text{guess } \{u_j^{h,0}\}, \quad j = 1, \cdots, M ; \\
\text{for } \ell = 1, 2, \cdots \\
\quad u_j^{h,\ell,0} = u_j^{h,\ell-1} ; \\
\quad \text{do } n = 1, \cdots, n_* \\
\quad \left\{ \begin{array}{l}
\mathcal{L}(u^{h,\ell,n}, v; \Omega_j) + (\mathrm{i}\eta^2\, u^{h,\ell,n}, v)_{\Omega_j} \\
\qquad + \sum_k \langle \frac{1}{2}(-\partial_{\mathbf{b},jk} u_j^{h,\ell,n} + \mathrm{i}\beta u_j^{h,\ell,n}), v \rangle_{\Gamma_{jk}} \\
\quad = (f, v)_{\Omega_j} + (\mathrm{i}\eta^2\, u^{h,\ell,0}, v)_{\Omega_j} \\
\qquad + \sum_k \langle \frac{1}{2}(-\partial_{\mathbf{b},kj} u_k^{h,\ell,n-1} + \mathrm{i}\beta u_k^{h,\ell,n-1}), v \rangle_{\Gamma_{jk}}, \\
\qquad \forall\, v \in V_j^h ;
\end{array} \right. \\
\quad \text{end do} \\
\quad u_j^{h,\ell} = u_j^{h,\ell,n_*} ; \\
\quad \text{if } \sum_j \|u_j^{h,\ell} - u_j^{h,\ell-1}\|_{0,\Omega_j} \Big/ \sum_j \|u_j^{h,\ell}\|_{0,\Omega_j} \le \varepsilon, \ \text{stop} ; \\
\text{end for}
\end{array} \tag{54}
$$

In (54), the DD method (16) is applied as the inner loop, while the ARTDI is the outer loop. For n_* sufficiently large, one can show the convergence of algorithm (54) even for the nonattenuative case ($q \equiv 0$). We are more interested in the problem of choosing computationally efficient algorithm parameters: β, η, and n_*.

For small eigenvalues λ of A (in modulus), $|\mathrm{Re}\,(\lambda)\,|$ can be arbitrarily small and $\mathrm{Im}\,(\lambda) = O(h^2)$. So

$$
\gamma = \max_{\lambda_k \in \sigma(A)} \left| \frac{\mathrm{i}d}{\lambda_k + \mathrm{i}d} \right| \le \frac{d}{\mathrm{Im}\,(\lambda_{k^*}) + d}
$$

for some k^*, where $d \approx h^2\eta^2$ and $\mathrm{Im}\,(\lambda_{k^*}) = O(h^2)$. Let

$$
\eta = O(h^r), \quad r = -1/2 \text{ or } -1, \tag{55}
$$

and ℓ_{AD} be the iteration number for ARTDI (50) to achieve a given error level, i.e.

$$
\left(\frac{d}{\mathrm{Im}\,(\lambda_{k^*}) + d} \right)^{\ell_{\mathrm{AD}}} \le \varepsilon.
$$

Then, it is not difficult to see

$$\ell_{\mathrm{AD}} = \begin{cases} O(h^{-1}), \, r = -1/2, \\ O(h^{-2}), \, r = -1. \end{cases} \tag{56}$$

Similarly, from (44), the number of DD iterations for a given error level can be derived as

$$n_{\mathrm{DD}} = \begin{cases} O(p_1^2), \quad r = -1/2, \\ O(h^2 p_1^2), \, r = -1. \end{cases} \tag{57}$$

So when $\eta = O(h^{-1})$, the total number of DD iterations for (54) to satisfy the given error level is $O(p_1^2)$, independent on h; the algorithm is *optimal*. The h–independence can be explained by the strong sensitivity of the DD method to q (and η); see (44).

6.2. An automatic, efficient relaxation parameter

In this subsection, we consider an automatic, heuristic, and nonexpensive strategy for finding efficient relaxation parameters β for (16) (and (54)). Its basic idea was first presented in [23] for strip–type DD methods for solving constant coefficient coercive elliptic problems. In [15], it was applied to the treatment of more general box–type DD methods for solving variable coefficient Helmholtz problems. Here we present the idea of the extension for a completeness of the paper.

For simplicity, we assume the domain $\Omega \ (:= \ (0,1)^2)$ is decomposed into M_x and M_y subdomains in the x– and y–directions, respectively, and each subdomain is equally partitioned into $(m_x - 1) \times (m_y - 1)$ elements. Let $(0, y_0)$ be a grid point in the domain. Consider the following one–dimensional problem:

$$\begin{aligned} -\frac{\partial^2}{\partial x^2} u - (p(x)^2 - \mathrm{i}q(x)^2)u &= f(x), \, x \in \Omega_{y_0}, \\ \frac{\partial u}{\partial \nu} + \mathrm{i}\alpha(x) u &= 0, \qquad x \in \Gamma_{y_0}, \end{aligned} \tag{58}$$

where $\Omega_{y_0} = \{(x,y) : y = y_0\}$, the horizontal grid line in $\overline{\Omega}$ passing $(0, y_0)$, and $\Gamma_{y_0} = \Gamma \cap \partial\Omega_{y_0}$. We apply a DD method to (58) in exactly the same way as in the original problem: decompose Ω_{y_0} into nonoverlapping M_x subintervals $\{\Omega_{y_0,j}\}$ ordered from left to right and partition the subintervals equally into $(m_x - 1)$ elements each. Let $\delta_h^2 u$ be the centered (second–order) finite difference approximation of $\partial^2 u / \partial x^2$. Then, consider the following

DD iteration: given an initial guess $u^0_{y_0,j}$, $j = 1, \cdots, M_x$, find $u^n_{y_0,j}$, $n \geq 1$, such that

$$
\begin{aligned}
-\delta_h^2 u^n_{y_0,j} - (p^2 - \mathrm{i}q^2)\, u^n_{y_0,j} &= f, && x \in \Omega_{y_0,j},\\
\partial_{\mathbf{c},j} u^n_{y_0,j} + \mathrm{i}\alpha u^n_{y_0,j} &= 0, && x \in \Gamma_{y_0,j},\\
\partial_{\mathbf{f},jk} u^n_{y_0,j} + \mathrm{i}\beta\, u^n_{y_0,j} &= -\partial_{\mathbf{b},kj} u^{n-1}_{y_0,k} + \mathrm{i}\beta\, u^{n-1}_{y_0,k}, && x \in \Gamma_{y_0,jk},
\end{aligned}
\tag{59}
$$

where $\Gamma_{y_0,j} = \partial\Omega_{y_0} \cap \partial\Omega_{y_0,j}$ and $\Gamma_{y_0,jk} = \partial\Omega_{y_0,j} \cap \partial\Omega_{y_0,k}$, $k = j+1$ or $j-1$. We will call problem (59) a *one–dimensional reduction* (ODR) of (18) in x–direction. (One can consider ODRs in y–direction similarly.)

Now, we will find β on the interface grid points $\{\Gamma_{y_0,j,j+1}\}, j = 1, \cdots, M_x - 1$, such that *the spectral radius of the iteration matrix of ODR* (59) *is zero.* This does not imply that (59) converges after one iteration (since its iteration matrix is not Hermitian), but implies that it converges arbitrarily fast. For a simple presentation, we let $M = M_x$ and $m = m_x$ and define a tridiagonal $m \times m$ matrix by

$$
T_m(\theta, a, b, c, d) = \begin{bmatrix}
\theta_1 - a & -c & & & \\
-1 & \theta_2 & -1 & & \\
& \ddots & \ddots & \ddots & \\
& & -1 & \theta_{m-1} & -1 \\
& & & -d & \theta_m - b
\end{bmatrix},
$$

where $\theta = (\theta_1, \cdots, \theta_m)^{\mathrm{T}}$. When the nodal points on each subinterval $\Omega_{y_0,j}$ are ordered from left to right, the matrix formulation for (59) reads

$$
\mathbf{u}^n_{y_0} = B^{-1} R\, \mathbf{u}^{n-1}_{y_0} + B^{-1}\mathbf{f}, \quad n = 1, 2, \cdots,
\tag{60}
$$

where

$$
B = \mathrm{diag}(D_1, \cdots, D_M), \quad \text{and}
$$

$$
R = \begin{bmatrix}
0 & F_1 & & & \\
E_2 & 0 & F_2 & & \\
& \ddots & \ddots & \ddots & \\
& & E_{M-1} & 0 & F_{M-1} \\
& & & E_M & 0
\end{bmatrix}.
$$

Here the matrices are defined by

$$
\begin{aligned}
D_1 &= T_m(\theta_1,\, \mathrm{i}2\alpha h,\, -\varphi_{12}, 2, 1),\\
D_j &= T_m(\theta_j,\, -\varphi_{j,j-1},\, -\varphi_{j,j+1}, 1, 1),\ \ j = 2, \cdots, M-1,\\
D_M &= T_m(\theta_M,\, -\varphi_{M,M-1},\, i2\alpha h, 1, 2),
\end{aligned}
$$

where, for $j = 1, \cdots, M$ and $i = 1, \cdots, m$,

$$
\begin{array}{l}
\theta_j = (\theta_{j,1}, \cdots, \theta_{j,m})^{\mathrm{T}},\ \ \theta_{j,i} = 2 - (\, p^2(x_{j,i}) - \mathrm{i}q^2(x_{j,i})\,)h^2, \\
x_{j,i} = \ \text{the } i\text{-th nodal point in } \Omega_{y_0,j}, \\
\varphi_{j,k} = 1 - \mathrm{i}\beta_{j,k}h, \\
\beta_{j,k} = \beta|_{\Gamma_{y_0,jk}},\ \ k = j \pm 1,
\end{array}
\tag{61}
$$

and

$$
E_j = \begin{bmatrix} 0 \cdots 0\ 1 & -\varphi_{j,j-1} \\ 0 \cdots 0\ 0 & 0 \\ \vdots \ddots \vdots\ \vdots & \vdots \\ 0 \cdots 0\ 0 & 0 \end{bmatrix}, \quad F_j = \begin{bmatrix} 0 & 0\ 0 \cdots 0 \\ \vdots & \vdots\ \vdots \ddots \vdots \\ 0 & 0\ 0 \cdots 0 \\ -\varphi_{j,j+1} & 1\ 0 \cdots 0 \end{bmatrix}.
$$

Here our main goal is to find $\beta_{j,k}$ such that the spectral radius of $G(:= B^{-1}R)$ is zero. It should be noticed that R has only $4(M-1)$ nonzero elements, and therefore the matrix G has only $4(M-1)$ nonzero columns that are related to the first or last columns of the matrices $D_j^{-1}, j = 1, \cdots, M$. Let $t_{\ell,k}^{(j)} = (D_j^{-1})_{\ell,k}$, the (ℓ, k)-element of D_j^{-1}. For a simple expression, we first consider the two subdomain case: $M = 2$. Then, removing zero columns and the corresponding rows from matrix G, we have

$$
\sigma(G) = \{0\} \cup \sigma(G'),
$$

where

$$
G' = \begin{bmatrix} 0 & 0 & -\varphi_{12}t_{m-1,m}^{(1)} & t_{m-1,m}^{(1)} \\ 0 & 0 & -\varphi_{12}t_{m,m}^{(1)} & t_{m,m}^{(1)} \\ t_{1,1}^{(2)} & -\varphi_{21}t_{1,1}^{(2)} & 0 & 0 \\ t_{2,1}^{(2)} & -\varphi_{21}t_{2,1}^{(2)} & 0 & 0 \end{bmatrix}.
$$

Let

$$
P = \begin{bmatrix} 1 & \varphi_{21} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \varphi_{12} & 1 \end{bmatrix}.
$$

Then a simple calculation yields

$$
P^{-1}\, G'\, P = \begin{bmatrix} 0 & 0 & 0 & t_{m-1,m}^{(1)} - \varphi_{21}t_{m,m}^{(1)} \\ 0 & 0 & 0 & t_{m,m}^{(1)} \\ t_{1,1}^{(2)} & 0 & 0 & 0 \\ t_{2,1}^{(2)} - \varphi_{12}t_{1,1}^{(2)} & 0 & 0 & 0 \end{bmatrix}.
$$

Again, removing zero columns and the corresponding rows from $P^{-1}\,G'\,P$ produces

$$\sigma(G) = \{0\} \cup \sigma(G') = \{0\} \cup \sigma(G''),$$

where

$$G'' = \begin{bmatrix} 0 & g_{12} \\ g_{21} & 0 \end{bmatrix},$$

with

$$g_{12} = t^{(1)}_{m-1,m} - \varphi_{21} t^{(1)}_{m,m}, \quad g_{21} = t^{(2)}_{2,1} - \varphi_{12} t^{(2)}_{1,1}.$$

Set $g_{12} = 0$ or $g_{21} = 0$; we can see that $\rho(G) = 0$.

Now, let us generalize the above argument. For example, consider the case: $M = 4$. Then, it is not difficult to derive the following:

$$\sigma(G) = \{0\} \cup \sigma(G''), \tag{62}$$

where

$$G'' = \begin{bmatrix} 0 & g_{12} & 0 & 0 & 0 & 0 \\ g_{21} & 0 & 0 & g_{24} & 0 & 0 \\ g_{31} & 0 & 0 & g_{34} & 0 & 0 \\ 0 & 0 & g_{43} & 0 & 0 & g_{46} \\ 0 & 0 & g_{53} & 0 & 0 & g_{56} \\ 0 & 0 & 0 & 0 & g_{65} & 0 \end{bmatrix}, \tag{63}$$

where

$$\begin{aligned} g_{2j-1,2j} &= t^{(j)}_{m-1,m} - \varphi_{j+1,j} t^{(j)}_{m,m}, \\ g_{2j,2j-1} &= t^{(j+1)}_{2,1} - \varphi_{j,j+1} t^{(j+1)}_{1,1}, \end{aligned} \quad j = 1, \cdots, M-1. \tag{64}$$

We wish to find $\beta_{j,j+1}$ such that, for $j = 1, \cdots, M-1$,

$$g_{2j-1,2j} = 0, \quad \text{or} \quad g_{2j,2j-1} = 0;$$

in this case, from (62) and (63), we see $\rho(G) = 0$. Set $g_{2j-1,2j} = 0$, $j = 1, \cdots, M-1$. Then, from (61) and (64),

$$1 - \mathrm{i}\beta_{j+1,j} h = \frac{t^{(j)}_{m-1,m}}{t^{(j)}_{m,m}}, \quad j = 1, \cdots, M-1. \tag{65}$$

The problem is thus reduced to the problem of finding the terms in the right–hand side of (65). When the problem coefficients p and q are constants, it is not difficult to find the terms analytically [23, 15]. Here our problem has variable coefficients; we probably should consider the following computer–aided method. Assume $\beta_{j,j-1}$ is known and let X be the last column of D_j^{-1}, i.e., $X = (t^{(j)}_{1,m}, t^{(j)}_{2,m}, \cdots, t^{(j)}_{m,m})^{\mathrm{T}}$. Then

$$D_j\, X = I_m, \tag{66}$$

where $I_m = (0, \cdots, 0, 1)^{\mathrm{T}}$. So, we can find X explicitly by inverting the tridiagonal $m \times m$ matrix D_j. In fact, D_j need not be inverted completely: let D_j be factorized into $L_j U_j$, where the diagonal elements of L_j are all one. It can be easily verified that

$$L_j U_j \, X = U_j \, X = I_m$$

and the $(m-1)$–th equation of the linear system $U_j \, X = I_m$ is

$$U_{j,m-1,m-1} \, t^{(j)}_{m-1,m} + U_{j,m-1,m} \, t^{(j)}_{m,m} = 0, \tag{67}$$

where $U_{j,\ell,k}$ is the (ℓ, k)–th element of U_j. It follows from (65) and (67) that

$$1 - \mathrm{i}\beta_{j+1,j} h = -\frac{U_{j,m-1,m}}{U_{j,m-1,m-1}}, \quad j = 1, \cdots, M-1. \tag{68}$$

Here it is assumed that $\beta_{j,j-1}$ is known, which can be satisfied when we start the procedure with D_1 to find β_{21}. Then, for known β_{21}, one can deal with D_2 to find β_{32}, and so on. So we have shown that $\rho(G) = 0$ when $\beta_{j+1,j}$ are obtained recursively by using (68).

Let us return to (16) defined on a 2-D domain. For each (horizontal or vertical) grid line passing interfaces Γ_{jk} perpendicularly, we can apply the above procedure to find the parameter β at the grid points on Γ_{jk}.

I called this whole procedure ADOP (Alternating Direction Optimal Procedure) in [15]. ADOP seeks the parameter β in such a way that the spectral radii of the iteration matrices of the ODRs are all zero. It should be noticed that ADOP is automatic and cheap. For 2 (resp. 3)–dimensional problems, its cost is strictly less than that of one pass (resp. one and a half passes) of the line Jacobi iterative algorithm. In Sect. 7, the efficiency of ADOP will be verified numerically.

ADOP is applicable to nonuniform meshes. Let

$$\{x_{j,1}, \, x_{j,2}, \, \cdots, \, x_{j,m}\}, \quad x_{j,1} < x_{j,2} < \cdots < x_{j,m},$$

be the grid points in $\Omega_{y_0,j}$ and $h_{j,m} = x_{j,m} - x_{j,m-1}$. Then, one can choose $\beta_{j+\mathrm{i},j}$ (on $\Gamma_{y_0,j,j+1}$) satisfying

$$1 - \mathrm{i}\beta_{j+1,j} h_{j,m} = -\frac{U_{j,m-1,m}}{U_{j,m-1,m-1}}$$

to impose zero spectral radii for the iteration matrices of ODRs.

7. Numerical results

This section presents experimental results to illustrate the effectiveness of iterative algorithms (16) and (54). The program was implemented in FORTRAN with complex double precision, and the experiments were carried out on an nCUBE2 parallel computer with 64 processors (in a hypercube configuration).

Let $\Omega = (0,1)^2$ and $\mathbf{x} = (x,y)$, the independent variables. We set the problem coefficients as follows:

$$p(x,y) \;=\; \frac{\omega}{c(x,y)}, \quad \alpha(x,y) \;=\; \omega, \quad q \;=\; \text{a nonnegative constant;}$$

the wave speed $c(x,y)$ is chosen as

$$\begin{aligned} &c_1(x,y) = 2 + 2x^3 + y^4, \\ &c_2(x,y) = \mathrm{e}^{xy}(2+\sin(\pi x))(2-\sin(3\pi y)), \\ &c_3(x,y) = \begin{cases} 4, & x \le 0.5, \\ 1 + \mathrm{e}^{x} + \sin(2\pi xy), & \text{elsewhere.} \end{cases} \end{aligned} \tag{69}$$

The source f is selected such that

$$u(x,y) = \frac{\phi(x)\cdot\phi(y)}{\omega^2}, \quad \phi(x) = \mathrm{e}^{\mathrm{i}\omega(x-1)} + \mathrm{e}^{-\mathrm{i}\omega x} - 2, \tag{70}$$

is the true solution of (1).

We consider the uniform bilinear finite elements of size h and the domain is decomposed into box–type subdomains. We denote the subdomain numbers in x and y–directions by M_x and M_y, respectively. Each subproblem is solved directly on a processor using a Gauss–elimination. The error is estimated by the relative maximum norm r_∞^n, and the iteration is stopped when $s_\infty^n \le 10^{-4}$:

$$r_\infty^n = \frac{\|U^n - u\|_{L^\infty(\Omega)}}{\|u\|_{L^\infty(\Omega)}}, \quad s_\infty^n = \frac{\|U^n - U^{n-1}\|_{L^\infty(\Omega)}}{\|U^n\|_{L^\infty(\Omega)}},$$

where U^n is the approximate solution of the n–th iteration. Zero initial values are assumed: $U^0 \equiv 0$. The integer n denotes the number of DD iterations and T_P (seconds) is the wall clock time for P processors to finish a given job.

In Table 1, we present the numerical convergence rate of algorithm (16) for the case: $\omega = 25$, $c = c_2$, $q = 3$, and $M_x \times M_y = 16 \times 4$. β_c denotes the relaxation parameter determined using (36) and (42) and β_{ADOP} is that of ADOP presented in Sect. 6.2. For β_c, the iteration number n increases as the mesh size h decreases, as shown in Theorem 4. It has been observed that the complex–valued parameter β_c behaves better than the real–valued

Table 1. Numerical results obtained using (4.3), when $\omega = 25$, $c = c_2$, $q = 3$, and $M_x \times M_y = 16 \times 4$

	$\beta = \beta_c$			$\beta = \beta_{\text{ADOP}}$		
$1/h$	n	r_∞^n	T_{64}	n	r_∞^n	T_{64}
32	108	0.059	3.5	51	0.059	2.2
64	189	0.015	9.7	54	0.014	3.6
128	329	0.0036	58.6	54	0.0036	11.2

Table 2. The performance of (4.3), when $\omega = 100$, $q = 20$, and $1/h = 256$. The DD iteration numbers are approximately $3(M_x + M_y)/2$

	$c = c_1$			$c = c_2$			$c = c_3$		
$M_x \times M_y$	n	r_∞^n	T_{32}	n	r_∞^n	T_{32}	n	r_∞^n	T_{32}
8×4	23	0.015	131.3	20	0.013	121.9	21	0.014	124.9
16×2	29	0.014	73.3	20	0.013	57.8	23	0.014	62.6
32×1	54	0.014	73.0	42	0.013	62.2	43	0.014	61.3

Table 3. The performance of (4.3) for $\omega = 200$, $q = 7$, and $1/h = 384$. The quality factor has values between 100 and 900 in most of the domain.

	$c = c_1$			$c = c_2$			$c = c_3$		
$M_x \times M_y$	n	r_∞^n	T_{64}	n	r_∞^n	T_{64}	n	r_∞^n	T_{64}
16×4	96	0.040	342.2	86	0.038	312.6	80	0.039	292.0
32×2	169	0.040	329.6	137	0.038	268.5	149	0.038	288.2

parameter (46), but both are unsatisfactory for problems of large wave numbers or variable coefficients. For the parameter β_{ADOP}, the algorithm seems to converge independently on the mesh size h. (See Table 4 below.)

In Tables 2–4, we will investigate the effectiveness of ADOP applied to (16).

In Table 2, we present numerical results obtained using (16) for various wave speeds and domain decompositions. We choose 32 processors and set $\omega = 100$, $q = 20$, and $1/h = 256$. In this table, we considered highly attenuated problems, i.e. the quality number $Q\,(= p^2/q^2)$ is small. The algorithm converges rapidly; the DD iteration numbers are approximately $3(M_x + M_y)/2$.

Table 3 contains the numerical results of (16) for 64 processors and $\omega = 200$, $q = 7$, and $1/h = 384$. The quality factor Q for the problems considered in the table has values between 100 and 900 in most of the domain. (It is known that $50 \le Q \le 1000$ for rocks.) It is easy to see that for the DD methods of the kind treated in this paper, at least $N\,(:= M_x + M_y - 1)$ iterations are required for complete convergence. We can see from the results in Tables 2 and 3 that the iteration number for (16) incorporating ADOP to achieve a given error level is $O(N)$, i.e. $O(H^{-1})$.

Table 4. Numerical results from (4.3), when $\omega = 40$, $q = 2$, and $M_x \times M_y = 8 \times 8$. The convergence rate does not seem to deteriorate as the mesh size h decreases

	$c = c_1$			$c = c_3$		
$1/h$	n	r_∞^n	T_{64}	n	r_∞^n	T_{64}
64	100	0.076	7.0	125	0.078	8.5
128	98	0.019	30.1	122	0.020	36.9
256	97	0.005	206.3	122	0.005	253.0

Table 5. The total number of DD iterations of (6.7), when $c = c_3$, $\omega = 55$, $q = 0$, $1/h = 128$, and $M_x \times M_y = 8 \times 8$. The results marked by asterisks are obtained from (4.3), i.e. (6.7) with $n_* = 1$ and $\eta = 0$

	$n_* = 2$		$n_* = 4$		$n_* = 6$		$n_* = 8$	
η	$f_{\rm t}$	f_δ	$f_{\rm t}$	f_δ	$f_{\rm t}$	f_δ	$f_{\rm t}$	f_δ
0	876*	92*	–	–	–	–	–	–
2	566	89	341	79	336	77	418	83
4	305	85	130	56	116	52	171	67
5	244	85	117	52	101	52	126	61
6	211	86	121	56	111	58	118	62
8	235	90	127	67	125	71	111	78

Table 4 is designed to consider relationship between the mesh size h and the convergence rate of (16). In this table, we set $\omega = 40$, $q = 2$, and $M_x \times M_y = 8 \times 8$. The convergence rate does not seem to deteriorate as the mesh size h decreases. So, algorithm (16) incorporating ADOP will lead to great advantages for large–scale problems. The second order convergence of approximation error in $L^\infty(\Omega)$–norm is observed.

We will use $\beta_{\rm ADOP}$ for the results in Tables 5 and 6.

Table 5 reports the total number of DD iterations of (54) for various n_*'s and η's. We set $c = c_3$, $\omega = 55$, $q = 0$, $1/h = 128$, and $M_x \times M_y = 8 \times 8$. Here we treat two different source functions: $f_{\rm t}$ denotes the source corresponding to the solution in (70) and

$$f_\delta(\mathbf{x}) = \sum_j \delta(\mathbf{x} - \mathbf{c}_j),$$

where $\mathbf{c}_j$ is the center of Ω_j. (For $f = f_\delta$, we set $\alpha(x, y) = \omega / c(x, y)$; f_δ is considered only in this table.) The asterisks indicate the results obtained from (16), i.e. (54) with $n_* = 1$ and $\eta = 0$. From the table one can see that algorithm (54) converges fastest when $n_* = 6$ and $\eta = 5$. For $f = f_{\rm t}$, it converges 7.8 times faster than (16), measured in wall clock time. In most numerical tests, algorithm (54) converges 2–10 times faster than (16), provided that n_* and η are well–chosen. It seems impossible to find the optimal parameters n_* and η analytically. It is numerically verified that

Table 6. The relative L^∞ error and the total number of DD iterations for (6.7), when $c \equiv 1$, $q \equiv 0$, $n_* = (M_x + M_y)/5$, and $\eta = \omega/4$. For $1/h = 512$, $M_x \times M_y = 64 \times 1$; $M_x \times M_y = 16 \times 4$ for others. For the case $1/h = 512$ and $\omega = 25$, the stopping criterion was set as $s_\infty^n \leq 10^{-5}$ and the corresponding total number of DD iterations was marked by an asterisk

$1/h$	$\omega = 25$	$\omega = 50$	$\omega = 100$	$\omega = 200$
64	0.04834 (68)	–	–	–
128	0.01208 (60)	0.08987 (60)	–	–
256	0.00301 (68)	0.02233 (52)	0.18100 (52)	–
512	0.00077 (234*)	0.00568 (195)	0.04571 (182)	0.35704 (182)

algorithm (54) converges fast when the parameters are chosen as follows:

$$n_* = \frac{M_x + M_y}{3} \sim \frac{M_x + M_y}{5}, \quad \eta = \frac{p}{4} \sim \frac{p}{5}, \tag{71}$$
$$\text{where } 0 \leq q \ll p.$$

Table 6 addresses the convergence rate of (54) as a function of mesh size and wave number. The floating point numbers denote the relative maximum errors r_∞^n and the total number of DD iterations are presented in parentheses. We choose $c \equiv 1$, $q \equiv 0$, $n_* = (M_x + M_y)/5$, and $\eta = \omega/4$. Set $M_x \times M_y = 64 \times 1$ for $1/h = 512$, due to the limit of the computer memory; $M_x \times M_y = 16 \times 4$ for others. For the case $1/h = 512$ and $\omega = 25$, a stronger stopping criterion ($s_\infty^n \leq 10^{-5}$) was set and the corresponding total number of DD iterations was marked by an asterisk in the table. We did not try to solve the problems corresponding to the right upper part of the table. From the table we can see that the relative maximum error grows even though the number of grid points per wavelength remains fixed. It seems that the maximum relative error

$$r_\infty^n \approx 0.0118\,\omega^3 h^2. \tag{72}$$

(See (7).) So, the number of points per wavelength will have to increase with the wave number to maintain accuracy. The use of a higher order discretization scheme can lead to greater advantages both in accuracy and in the convergence rate of DD iterations [2, 18]. It should be noticed that the convergence rate does not deteriorate as the mesh size h decreases and/or the wave number increases, which has been observed in various numerical experiments.

In summary, we claim that algorithm (54) converges independently on the mesh size and the wave number, for the parameters β, n_*, and η chosen by ADOP and (71). Also the total number of DD iterations for the algorithm to reach a given error level has been observed to be $O(H^{-1})$. Note that the algorithm does not involve any of global corrections. It is known [10] that

the condition number of any DD preconditioner built only from solvers on local subregions is necessarily at least $O(H^{-2})$.

One can consider mesh refinement near the sources and the discontinuities of the coefficients. In view to save computer storage and avoiding complicated data structures it is interesting (and reasonable) to incorporate local grid refinement over subdomains where the strong variations are expected and retain coarser grids elsewhere. Domain decomposition methods (or their basic concepts) provide a systematic and elegant way to implement the above ideas.

Acknowledgements. I wish to thank Professors Jim Douglas, Jr. and William W. Symes for their valuable comments and encouragement. All numerical results were performed on the nCUBE2 at the Department of Computer Sciences, Purdue University. I would like to thank Professor Elias N. Houstis for making that parallel computer available.

References

1. Bayliss, A., Goldstein, C., Turkel, E. (1983): An iterative method for the Helmholtz equation. J. Comput. Phys. **49**, 443–457
2. Bayliss, A., Goldstein, C., Turkel, E. (1985): On accuracy conditions for the numerical computation of waves. J. Comput. Phys. **59**, 396–404
3. Canuto, C., Hussaini, M.Y., Quarteroni, A., Zang, T.A. (1988): Spectral Methods in Fluid Dynamics. Springer-Verlag, Berlin, Heidelberg, New York, London, Paris, Tokyo, Hong Kong, Barcelona
4. Ciarlet, P.G. (1978): Finite Element Methods for Elliptic Problems. North–Holland, Amsterdam
5. Després, B. (1991): Domain decomposition method and the Helmholtz problem. In. Cohen, G., Halpern, L., Joly, P. (eds.) Mathematical and Numerical Aspects of Wave Propagation Phenomena, pp. 44–51. SIAM, Philadelphia
6. Douglas, C.C., Douglas, J.Jr. (1993): A unified convergence theory for abstract multigrid or multilevel algorithms, serial and parallel. SIAM J. Numer. Anal. **30**, 136–158
7. Douglas, J.Jr., Hensley, J.L., Roberts, J.E. (1993): An alternating–direction iteration method for Helmholtz problems. Technical Report #214, Mathematics Department, Purdue University, W. Lafayette, IN 47907
8. Douglas, J.Jr., Paes Leme, P.S., Roberts, J.E., Wang, J. (1993): A parallel iterative procedure applicable to the approximate solution of second order partial differential equations by mixed finite element methods. Numer. Math. **65**, 95–108
9. Douglas, J.Jr., Santos, J.E., Sheen, D. (1994): Approximation of scalar waves in the space–frequency domain. Math. Models Methods Appl. Sci. **4**, 509–531
10. Dryja, M., Widlund, O.B. (1994): Some recent results on Schwarz type domain decomposition algorithms. In: Quarteroni, A., Periaux, J., Kuznetsov, Y.A., Widlund, O.B. (eds.) Domain Decomposition Methods in Science and Engineering, pp. 53–61. Contemporary Mathematics 157, SIAM, Philadelphia
11. Faber, V., Manteuffel, T. (1984): Necessary and sufficient conditions for the existence of a conjugate gradient method. SIAM J. Numer. Anal. **21**, 352–362
12. Freund, R.W. (1992): Conjugate gradient–type methods for linear systems with complex symmetric coefficient matrices. SIAM J. Sci. Stat. Comput. **13**, 425–448

13. Hackbusch, W. (1985): Multigrid Methods and Applications. Springer, Berlin
14. Joubert, W.D., Young, D.M. (1987): Necessary and sufficient conditions for the simplification of the generalized conjugate–gradient algorithms. Linear Algebra Appl. **88/89**, 449–485
15. S. Kim, A parallelizable iterative procedure for the Helmholtz problem, Appl. Numer. Math. 14 (1994), pp. 435–449.
16. Kim, S. (1995): Parallel multidomain iterative algorithms for the Helmholtz wave equation. Appl. Numer. Math. **17**, 411–429
17. Kim, S. (1994): On the use of rational iterations and domain decomposition methods for solving the Helmholtz problem. Technical Report # 251, Center for Applied Mathematics, Purdue University, W. Lafayette, IN 47907
18. Kim, S.: Domain decomposition methods for the Helmholtz problem. In preparation
19. Lions, P.L. (1990): On the Schwarz alternating method III: a variant for nonoverlapping subdomains. In: Chan, T.F., Glowinski, R., Periaux, J., Widlund, O.B. (eds.) Third International Symposium on Domain Decomposition Method for Partial Differential Equations, pp. 202–223. SIAM, Philadelphia
20. Schatz, A.H. (1974): An observation concerning Ritz–Galerkin methods with indefinite bilinear forms. Math. Comp. **28**, 959–962
21. Shaidurov, V.V., Ogorodnikov, E.I. (1991): Some numerical methods of solving Helmholtz wave equation. In: Cohen, G., Halpern, L., Joly, P. (eds.) Mathematical and Numerical Aspects of Wave Propagation Phenomena, pp. 73–79. SIAM, Philadelphia
22. Le Tallec, P. (1994): Domain decomposition methods in computational mechanics. Comput. Mech. Advances **1**, 121–220
23. Tang, W.P. (1992): Generalized Schwarz splittings. SIAM J. Sci. Stat. Comput. **13**, 573–595