Journal of Computational Physics 345 (2017) 330–344

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

An iterative solver for the 3D Helmholtz equation

Mikhail Belonosov [a], Maxim Dmitriev [b], Victor Kostin [c,*], Dmitry Neklyudov [c], Vladimir Tcheverda [c,d]

[a] EXPEC ARC GRC Delft, Aramco Overseas Company B.V., Informaticalaan 6, 2628 ZD, Delft, The Netherlands
[b] EXPEC Advanced Research Center, Saudi Aramco, Building 137, Dhahran 31311, Saudi Arabia
[c] Trofimuk Institute of Petroleum Geology and Geophysics SB RAS, 3, Koptyug ave, 630090, Novosibirsk, Russia
[d] Kazakh–British Technical University, 59, Tole bi st, 480091, Almaty, Kazakhstan

ARTICLE INFO

Article history:
Received 12 December 2016
Received in revised form 11 May 2017
Accepted 13 May 2017
Available online 17 May 2017

Keywords:
Acoustics
Modeling
Helmholtz equation
Iterative methods
Preconditioning

ABSTRACT

We develop a frequency–domain iterative solver for numerical simulation of acoustic waves in 3D heterogeneous media. It is based on the application of a unique preconditioner to the Helmholtz equation that ensures convergence for Krylov subspace iteration methods. Effective inversion of the preconditioner involves the Fast Fourier Transform (FFT) and numerical solution of a series of boundary value problems for ordinary differential equations. Matrix-by-vector multiplication for iterative inversion of the preconditioned matrix involves inversion of the preconditioner and pointwise multiplication of grid functions. Our solver has been verified by benchmarking against exact solutions and a time-domain solver.

1. Introduction

Numerical simulation of acoustic wavefields is an important part of many algorithms developed to solve problems arising in exploration geophysics. Usually, simulation needs to be performed many times to reach the final result and consumes the main part of the computational time. In particular, it serves as an engine for acoustic frequency–domain full waveform inversion (FD FWI) ([1–3] and references cited therein). For macro velocity reconstruction such simulation is performed several times for a number of low frequencies (up to 10 Hz) at each iteration of this process. Therefore, the efficiency of the whole process strongly depends on how fast and accurately the simulation is performed.

In the frequency domain, the acoustic wave propagation in R^3 is governed by the Helmholtz equation supplied with some condition at infinity [4]. The numerical complexity of the problem rises due to the sheer size of the system of linear equations to solve and the unfavorable spectral properties of the coefficient matrix of this system. Time-domain schemes are commonly used [5,6]. Frequency–domain iterative solvers with efficient preconditioners present advantages when one deals with FD FWI and uses only a few frequencies per inversion [7,8]. Their memory and computational complexities are $O(N^3)$, where N is the number of grid points in one direction.

A common approach to solve the Helmholtz equation is using Krylov-type iterative techniques for the system of linear algebraic equations that arises in finite-difference or finite-element approximations of the respective boundary value

[*] Corresponding author.
E-mail addresses: Mikhail.Belonosov@aramcooverseas.com (M. Belonosov), maxim.dmitriev@aramco.com (M. Dmitriev), KostinVI@ipgg.sbras.ru (V. Kostin), dmitn@mail.ru (D. Neklyudov), CheverdaVA@ipgg.sbras.ru (V. Tcheverda).

http://dx.doi.org/10.1016/j.jcp.2017.05.026
0021-9991/© 2017 Elsevier Inc. All rights reserved.

problems (see, for example, [2,9,10]). The indefiniteness of the coefficient matrix leads to its poor conditionality and slow convergence of the Krylov subspace iteration method [11].

This work follows the approach proposed in [12]. We construct a right preconditioner as a complex damped Helmholtz operator (shifted Laplace operator, [13–17]) with some 1D vertically heterogeneous background velocity. This operator can be inverted effectively. We represent the actual Helmholtz operator as a perturbation of the preconditioner. As a result, the operator of the preconditioned system becomes some perturbation of the unity operator. The norm of this perturbation is proportional to the magnitude of the velocity perturbation and frequency. For cases where the perturbation is small enough, the convergence of the iterative method is guaranteed. Similar approach with 1D vertically inhomogeneous background velocity involved was presented in [18] for the Maxwell equations in the 3D half-space.

Matrix-by-vector multiplication of the preconditioned system is decomposed to a series of 2D Fast Fourier Transforms (FFTs) in horizontal directions, solving a set of pairwise independent boundary value problems for ordinary differential equations in the vertical direction followed by a series of inverse 2D FFTs and pointwise multiplication by a scalar function. The numerical complexity of such an algorithm is close to optimal. Effective parallelization techniques can be applied at all steps of the algorithm. This enables us to perform simulation of an acoustic wavefield for realistic scenarios within a reasonable computational time.

This paper is structured as follows. In Section 2, the problem is considered using a set of models that vary only in the vertical direction. We introduce Perfectly Matched Layers (PMLs) to attenuate reflections from the boundary of a computational domain and set a respective boundary value problem. We derive a method to solve the problem that consists of expansion of the unknown function in the Fourier series, solving boundary value problems for ordinary differential equations for the coefficients of the Fourier series and summing up the Fourier series. In section 3, we discuss the details of numerical implementation of the algorithm and estimate the numerical complexity. In Section 4, we develop an algorithm to solve the problem in a 3D heterogeneous model that is the main goal of this paper. The Helmholtz equation is preconditioned by the Helmholtz operator in a vertically inhomogeneous medium, and the Krylov-type iteration method is applied. Some other ingredients to improve the convergence rate are introduced. In Section 5, the feasibility of the method is verified by several numerical experiments. We end with the conclusions in Section 6.

2. Solution to the Helmholtz equation in a 3D layered medium

In the algorithm we describe in this paper, to solve the Helmholtz equation in a 3D heterogeneous bounded medium, the main step is solving a boundary value problem in a layered bounded medium. Later on, we will describe how the layered medium arises naturally in the solution process. In this section we develop an algorithm to solve the problem in a layered medium and explain step by step how we shift from the statement of the problem in a 3D infinite layered medium to a bounded medium.

2.1. 3D vertically heterogeneous infinite layer

Consider the 3D Helmholtz equation in R^3 filled with an acoustic medium having sound velocity (or compressional wave propagation velocity) $c(z)$ that depends only on depth (z coordinate)

$$\Delta u(x,y,z) + \frac{\omega^2}{c^2(z)} u(x,y,z) = f(x,y,z). \tag{1}$$

Here $u(x,y,z)$ is the pressure at point (x,y,z), $\Delta = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ is the Laplace operator, ω is the real angular frequency and $f(x,y,z)$ is the source function.

It is assumed that the unknown function satisfies the radiation condition at infinity (see [4]). Note that everywhere below, this condition is assumed implicitly when we consider equation (1) in an unbounded domain.

Suppose that function $c(z)$ is a constant function outside some interval, i.e.,

$$c(z) = c_0 \text{ for } |z| \geq Z_0 > 0,$$

and, consequently, the upgoing and the downgoing waves being described by equation (1) do not reflect for $|z| > Z_0$. If we are interested in the solution in a bounded outside interval $|z| \leq Z_0$ domain (our case), PMLs (see e.g. [19–21]) could be used to reduce reflections from the boundary.

To consider this case, let D be an acoustic layer

$$D = \{(x,y,z) : (x,y) \in R^2, -Z < z < Z\},$$

where $Z > Z_0$. This domain includes sublayers $\{(x,y,z) : (x,y) \in R^2, Z_0 < z < Z\}$ and $\{(x,y,z) : (x,y) \in R^2, -Z < z < -Z_0\}$, where the sound velocity is constant. Layer D is extended to

$$D_{PML} = \{(x,y,z) : (x,y) \in R^2, -Z_1 < z < Z_1\} \supset D,$$

where $Z_1 > Z$. Unknown function $u(x, y, z)$ is defined in D_{PML} and satisfies differential equation (1) in D, and in extra layers $Z < z < Z_1$ and $-Z_1 < z < -Z$ it satisfies modified equations. In layer $Z < z < Z_1$, the modified equation is

$$\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial \tilde{z}^2}\right) u(x, y, z) + \frac{\omega^2}{c^2(z)} u(x, y, z) = f(x, y, z), \tag{2}$$

where source function $f(x, y, z)$ is extended into D_{PML} with zeros. In this expression, a new variable

$$\tilde{z}(z) = z - \frac{i}{\omega} \int\limits_0^z d(s) ds$$

is introduced using some smooth damping function $d(s)$ [21], which is zero for $0 \le z \le Z$ and positive for $Z < s < Z_1$. So, $\tilde{z}$ coincides with z for $0 < z \le Z$ and becomes complex valued with a negative imaginary part for $Z < z < Z_1$. On boundary $z = Z_1$, the Dirichlet homogeneous condition $u = 0$ is imposed. The damping function assures attenuation of waves within layer $Z < z < Z_1$, in particular the reflection from boundary $z = Z_1$ due to the Dirichlet condition. We do not consider here the questions of how to choose the damping function and the width of the PML. These and other important practical questions are considered in the numerous papers on PMLs. Construction of the PML for sublayer $-Z_1 < z < -Z$ is similar.

Equation (2) can be used in the whole domain D_{PML} with a note that it coincides with equation (1) within D. Solutions to equation (2) are complex valued functions, though only real parts of these solutions have physical meaning.

Along with equation (1), in domain D, we consider the modified Helmholtz equation

$$\Delta u(x, y, z) + (1 - i\beta) \frac{\omega^2}{c^2(z)} u(x, y, z) = f(x, y, z) \tag{3}$$

with some constant parameter $\beta > 0$. This parameter ensures existence and uniqueness of the solution to equation (3), which tends to zero for $\sqrt{x^2 + y^2} \to \infty$. The operator of equation (3), called the shifted Laplace operator, has been proved to be a good preconditioner for the Helmholtz equation (see [14,15]).

The PML for equation (3) is constructed in a similar way as that for the original Helmholtz equation (1) by considering equation

$$\tilde{\Delta} u(x, y, z) + (1 - i\beta) \frac{\omega^2}{c^2(z)} u(x, y, z) = f(x, y, z) \tag{4}$$

in domain D_{PML} with $\tilde{\Delta} = \left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial \tilde{z}^2}\right)$ and homogeneous Dirichlet boundary conditions posed on outer boundaries.

So far, we have dealt with R^3, but similarly we could consider equation (1) in a half-space $\{(x, y, z) : (x, y) \in R^2, z > 0\}$ with the free-surface boundary condition $u = 0$ on plane $z = 0$. In this case, only one PML is needed and domains D_{PML} and D are

$$\begin{aligned} D_{PML} &= \{(x, y, z) : (x, y) \in R^2, 0 < z < Z_1\}, \\ D &= \{(x, y, z) : (x, y) \in R^2, 0 < z < Z\}. \end{aligned} \tag{5}$$

Below, we consider only this case but it is worth mentioning that all the results can be easily rewritten for the case of R^3.

Equation (4) is considered in D_{PML} with boundary conditions

$$\begin{aligned} u(x, y, 0) &= 0, \\ u(x, y, Z_1) &= 0. \end{aligned} \tag{6}$$

Applying a 2D Fourier transform over x and y in equation (4) results in a set of linear ordinary differential equations of second order

$$\frac{d^2}{d\tilde{z}^2} \hat{u}(k_x, k_y; z) + \left(k^2(z) - k_x^2 - k_y^2\right) \hat{u}(k_x, k_y; z) = \hat{f}(k_x, k_y; z), \tag{7}$$

where k_x, k_y are spatial frequencies (Fourier variables dual to x and y respectively) and $k^2(z) = (1 - i\beta)\omega^2/c^2(z)$. The caret symbol $\hat{}$ above functions in equation (7) and below denotes respective Fourier images:

$$\begin{aligned} \hat{u}(k_x, k_y; z) &= \int\limits_{R^2} u(x, y, z) e^{-i(k_x x + k_y y)} dx dy, \\ \hat{f}(k_x, k_y; z) &= \int\limits_{R^2} f(x, y, z) e^{-i(k_x x + k_y y)} dx dy. \end{aligned} \tag{8}$$

Equation (7) can be written in terms of differentiation with respect to variable z

$$\left(\mu(z)\frac{d}{dz}\mu(z)\frac{d}{dz}+k^2(z)-k_x^2-k_y^2\right)\hat{u}(k_x,k_y;z)=\hat{f}(k_x,k_y;z), \tag{9}$$

where

$$\mu(z)=\frac{i\omega}{d(z)+i\omega}.$$

Differential equations (9) (or equivalently, (7)) are fully independent of each other for different values of k_x, k_y. Boundary conditions (6), posed for solutions of equation (4), imply the following boundary conditions for solutions of equation (9)

$$\begin{aligned}\hat{u}(k_x,k_y;0) &= 0,\\ \hat{u}(k_x,k_y;Z_1) &= 0.\end{aligned} \tag{10}$$

As a result, we get a set of independent boundary value problems on interval $[0, Z_1]$ for functions $\hat{u}(k_x,k_y;z)$.

The correctness (i.e. existence and uniqueness of solutions) of the above boundary value problems for equations (9) is guaranteed by parameter β being positive. The solutions can be represented in the form

$$\hat{u}(k_x,k_y;z)=\int\limits_0^{Z_1} G(z,\zeta;k_x,k_y)\hat{f}(k_x,k_y,\zeta)d\zeta, \tag{11}$$

where $G(z,\zeta;k_x,k_y)$ are the respective Green's functions.

Having found $\hat{u}(k_x,k_y;z)$, the solution $u(x,y,z)$ of equation (4) is computed using the inverse Fourier transform

$$u(x,y,z)=\frac{1}{4\pi^2}\int\limits_{R^2}\hat{u}(k_x,k_y;z)e^{i(k_x x+k_y y)}dk_x dk_y. \tag{12}$$

Let L_0 be a differential operator defined by the left-hand side of equation (4) for functions subject to boundary conditions posed on boundaries $z=0$ and $z=Z_1$, as described above. The domain of operator L_0 is defined as a linear subspace in Sobolev functional space $H^1(D_{PML})$ (quadratically integrable over D_{PML} functions with quadratically integrable weak partial derivatives of first order) subject to its elements satisfying the boundary conditions.

The problem we solve can be written in the operator form as

$$L_0 u=\tilde{\Delta}u+(1-i\beta)\frac{\omega^2}{c^2(z)}u=f, \tag{13}$$

where f is the right-hand side of the differential equation (4) and is supposed to belong to the functional subspace $H^{-1}(D_{PML})$. The solution to this problem is formally represented as

$$u=L_0^{-1}f.$$

Above, we described an algorithm to compute $L_0^{-1}f$. The algorithm comprises three steps:

1. Computing the forward Fourier transform of source function f using formula (8).
2. Solving boundary value problems (9) and (10) for the ordinary differential equations (see formula (11), which formally represents the solutions).
3. Computing the inverse Fourier transform (see (12)).

The simplicity of the algorithm for inverting L_0 makes this operator suitable to be used as a preconditioner. In the next section we modify the operator and the inversion algorithm to be applicable for bounded parallelepipedal domains. Note, that this modification does not complicate the algorithm. Keep in mind, that existence of a fast inversion method is a necessary requirement for an operator to be used as a preconditioner.

2.2. 3D layered bounded medium

In practical applications, for instance when we deal with seismic waves, bounded domains are considered. In this subsection, we modify the aforementioned algorithm to be able to work in a cuboid domain

$$D_{bound}=\{(x,y,z):0\le x\le X, 0\le y\le Y, 0\le z\le Z_1\} \tag{14}$$

of size $X\times Y\times Z_1$. Equation (4) is considered in D_{bound} with PML included.

Formal replacement of integration over R^2 in (8) to integrals over rectangle $\Pi = [0, X] \times [0, Y]$ leads to

$$\begin{aligned} f_{k_x,k_y}(z) &= \frac{1}{XY}\int\limits_{\Pi} f(x,y,z)e^{-2\pi i\left(\frac{k_x x}{X}+\frac{k_y y}{Y}\right)}dxdy, \\ u_{k_x,k_y}(z) &= \frac{1}{XY}\int\limits_{\Pi} u(x,y,z)e^{-2\pi i\left(\frac{k_x x}{X}+\frac{k_y y}{Y}\right)}dxdy, \end{aligned} \tag{15}$$

where k_x, k_y are integer indices. These are Fourier series coefficients (with respect to x and y) for functions $u(x, y, z)$ and $f(x, y, z)$.

Application of transform (15) to equation (4) in D_{bound} results in ordinary differential equations for Fourier series coefficients $u_{k_x,k_y}(z)$

$$\left(\mu(z)\frac{d}{dz}\mu(z)\frac{d}{dz} + k^2(z) - k_x^2 - k_y^2\right)u_{k_x,k_y}(z) = f_{k_x,k_y}(z) \tag{16}$$

with boundary conditions

$$\begin{aligned} u_{k_x,k_y}(0) &= 0, \\ u_{k_x,k_y}(Z_1) &= 0. \end{aligned} \tag{17}$$

These boundary value problems differ from (9), (10) only by notations.

Having found solutions $u_{k_x,k_y}(z)$ to these boundary value problems, function $u(x, y, z)$ is restored using the summation formula

$$u(x,y,z) = \sum_{k_y=-\infty}^{\infty}\sum_{k_x=-\infty}^{\infty} u_{k_x,k_y}(z)e^{2\pi i\left(\frac{k_x x}{X}+\frac{k_y y}{Y}\right)}. \tag{18}$$

This formula replaces the inverse Fourier transform (12) in the third step above. Note that $u(x, y, z)$, obtained by formula (18), is a periodic function with respect to x and y with periods X and Y respectively.

Let us remember that these modifications are performed to develop an algorithm to invert operator L_0 that acts on functions defined in D_{bound}. Given source function $f \in H^{-1}(D_{bound})$ Function $u = L_0^{-1} f$ belongs to the functional space $H^1(D_{bound})$ and satisfies boundary conditions (6). It also satisfies periodic boundary conditions with respect to x and y:

$$\begin{aligned} u(0,y,z) &= u(X,y,z), \\ u(x,0,z) &= u(x,Y,z). \end{aligned} \tag{19}$$

In other words, application of operator L_0^{-1} to any function $f \in H^{-1}(D_{bound})$ results in a function, which is periodic with respect to x and y.

To sum up, the modified algorithm of inverting L_0 is as follows:

1. Computing Fourier series coefficients $f_{k_x,k_y}(z)$ of source function f using formula (15).
2. Solving boundary value problems (16) and (17).
3. Restoring the solution by summation formula (18).

3. Computational complexity to invert L_0

In the algorithm we describe, solving boundary value problems $L_0 v = g$ is to be performed many times for different right-hand sides $g(x, y, z)$ defined in D_{bound}. Assume this domain is covered by a uniform grid of $N_x \times N_y \times N_z$ points that we use to approximate the operator equation $L_0 v = g$. For the respective discrete problem, we use the same notation

$$L_0 v = g,$$

where g and v are grid functions for the right-hand side and the unknown function respectively (we hope that using the same notations for functions and their grid counterparts will not confuse the reader).

Computing Fourier series coefficients $g_{k_x,k_y}(z)$ of function $g(x, y, z)$ reduces to a numerical approximation of the integral over rectangle $\{0 \le x \le X, 0 \le y \le Y\}$ and can be carried out by use of the FFT algorithm [22]. On the equidistant grid of $N_x \times N_y$ points the FFT needs $O(N_x N_y \log(N_x N_y))$ floating point operations resulting in $N_x \times N_y$ values of the Fourier coefficients. The FFT should be applied to every (x, y)-plane. In total, the FFT applied to the right-hand side g requires $O(N_x N_y N_z \log(N_x N_y))$ floating point operations and results in $N_x N_y N_z$ Fourier coefficients.

The second step of the algorithm is solving boundary value problems (16) and (17) on interval $[0, Z_1]$. Each differential equation should be approximated on the grid of N_z points. Along with the boundary conditions, this results in a system of N_z linear equations with a banded coefficient matrix. The bandwidth depends on the method chosen for approximation of the differential equation. In our implementation, we use a 6th order finite difference scheme resulting in a seven-diagonal coefficient matrix.[1] The computational complexity to solve a system of linear equations with banded coefficient matrix of size N_z with fixed bandwidth is $O(N_z)$. The systems have to be solved for every pair of spatial frequencies k_x, k_y. As a result, the total complexity of the second step is $O(N_x N_y N_z)$.

The last step is the summation formula (18), which is carried out by the inverse FFT with the same computational complexity as the first step. Summing up all estimates, we conclude that numerical complexity of the described algorithm is $O(N_x N_y N_z \log(N_x N_y))$.

It is possible to reduce the number of floating point operations while inverting L_0. Solutions to equations (16) vanish rapidly for $k_x^2 + k_y^2 > \frac{\omega^2}{c_{\min}^2}$, where $c_{\min} = \min_{0 \le z \le Z} c(z)$. So, in practice, for a given accuracy level there is no need to consider equations for the spatial frequencies lying outside the circle $k_x^2 + k_y^2 \le \gamma \frac{\omega^2}{c_{\min}^2}$ (for example, to achieve 1% accuracy one can take $\gamma = 2.5$).

4. Preconditioning the Helmholtz equation in a 3D inhomogeneous medium

So far, we have dealt with vertically inhomogeneous media. Now we proceed to a general case of the Helmholtz equation

$$Lu = \Delta u(x, y, z) + \frac{\omega^2}{c^2(x, y, z)} u(x, y, z) = f(x, y, z) \tag{20}$$

in cuboid

$$D' = \{(x, y, z) : X_l < x < X_r, Y_l < y < Y_r, 0 < z < Z_b\}.$$

We embed D' in some bigger cuboid D_{bound} defined by (14). As before, boundary $z = 0$ is treated as a free surface.

We expand equation (20) from D' to domain D_{bound}. Outside of D' function f is set to zero. In subdomain $\{(x, y, z) : 0 < x < X, 0 < y < Y, Z_b < z < Z_1\}$ we build a PML assuming the sound velocity function in this subdomain is extended by a constant as described in section 2. The lateral velocity extension outside of D' will be explained later in this section. As a result, we come to equation[2]

$$Lu = \tilde{\Delta} u(x, y, z) + \frac{\omega^2}{c^2(x, y, z)} u(x, y, z) = f(x, y, z) \tag{21}$$

in domain D_{bound} that includes the PML and where the unknown function satisfies boundary conditions

$$\begin{aligned} u(x, y, 0) &= 0, \\ u(x, y, Z_1) &= 0. \end{aligned} \tag{22}$$

On vertical boundaries of D_{bound} we set periodic boundary conditions (cf. (19)) with respect to x and y

$$\begin{aligned} u(0, y, z) &= u(X, y, z), \\ u(x, 0, z) &= u(x, Y, z). \end{aligned} \tag{23}$$

This way, the boundary value problem is completely defined and the operator L, which is introduced in (21), is defined as well. Its domain is the functional space $H^1(D_{bound})$ of functions u that satisfy boundary conditions (22) and (23). Note that by using periodic boundary conditions we ensure the domains of operators L and L_0 (subsection 2.2) are the same. This is crucial for applicability of the algorithm.

Let operator L_0 be defined by a modified Helmholtz equation in a layered medium (cf. equation (13))

$$L_0 u = \tilde{\Delta} u(x, y, z) + (1 - i\beta) \frac{\omega^2}{c_0^2(z)} u(x, y, z) \tag{24}$$

with the same boundary conditions as for operator L. Given $c_0(z)$ this operator acts on functions defined in D_{bound} and has the same domain as operator L.

[1] In a particular case of piecewise constant function $c(z)$, it is possible to get the solution in a closed form of linear combinations of exponential functions (the up-going and the down-going waves) on each interval where $c(z)$ is constant [12]. For this approach, the coefficient matrix of the system of linear equations is tridiagonal.

[2] Note we use the same notation L for the operator.

Operator L can be represented as a perturbation of operator L_0:

$$\begin{aligned} Lu &= \Delta u + \frac{\omega^2}{c^2(x,y,z)}u \\ &= \Delta u + (1-i\beta)\frac{\omega^2}{c_0^2(z)}u + \left(\frac{\omega^2}{c^2(x,y,z)} - (1-i\beta)\frac{\omega^2}{c_0^2(z)}\right)u \\ &= L_0 u - \delta L u, \end{aligned} \tag{25}$$

where the action of operator δL on function u is multiplication of u by function

$$\gamma(x,y,z;\omega,\beta) = -\omega^2\left(\frac{1}{c^2(x,y,z)} - \frac{(1-i\beta)}{c_0^2(z)}\right).$$

Let us explain how the velocity function $c(x,y,z)$ is defined outside of horizontal cross-sections of D'. Assume $c_0(z)$ is already defined. To avoid reflections from the interfaces between D' and D_{bound}, we require $c(x,y,z)$ to be continuous across these interfaces. Additionally, function $\gamma(x,y,z;\omega,\beta)$ that defines operator δL is required to be zero on outer boundaries of D_{bound}. One may notice that, due to these conditions, $c(x,y,z)$ becomes complex valued in the relative complement of D' with respect to D_{bound}. Using a complex valued sound velocity is equivalent to introducing absorbing layers (see [23]) and helps to reduce the influence of periodic boundary conditions on the solution.

Taking into account the invertibility of L_0 (section 2.2) we can factor L as a product of two operators

$$L = L_0 - \delta L = \left(I - \delta L \cdot L_0^{-1}\right)L_0. \tag{26}$$

Factorization (26) can be used to split the solution of equation $Lu = f$ in two steps. First, we can solve equation

$$\left(I - \delta L \cdot L_0^{-1}\right)v = f, \tag{27}$$

and after that find u from equation $L_0 u = v$.

Factorization (26) can be treated as right preconditioning of operator L by operator L_0^{-1}. The preconditioned operator (the operator of the equation (27)) is the unity operator I perturbed by bounded operator $\delta L \cdot L_0^{-1}$. To reduce the perturbation, function $c_0(z)$ should be close to $c(x,y,z)$ and can be defined as some average of velocity $c(x,y,z)$ over horizontal cross-sections of D_{bound}. For example, we can use the following formula:

$$\frac{1}{c_0^2(z)} = \frac{1}{XY}\int_0^X\int_0^Y \frac{1}{c^2(x,y,z)}dxdy.$$

Another way to define $c_0(z)$ is

$$c_0(z) = \frac{1}{2}\left(\max_{(x,y)} c(x,y,z) + \min_{(x,y)} c(x,y,z)\right).$$

For the last choice, the C-norm of the difference $\left(\frac{1}{c^2(x,y,z)} - \frac{1}{c_0^2(z)}\right)$ would be minimized, which might help to minimize the norm of operator δL (for the case $\beta = 0$ only). Let us note that it is not the norm, but the spectrum of perturbation $\delta L \cdot L_0^{-1}$ that influences the convergence of the Krylov subspace iteration method used to solve equation $\left(I - \delta L \cdot L_0^{-1}\right)v = g$. Therefore, we would suggest experimenting with different ways of averaging $c(x,y,z)$.

In order to solve equation $\left(I - \delta L \cdot L_0^{-1}\right)v = g$ we apply the Induced Dimension Reduction algorithm (IDR [24–26]), which has moderate memory requirements. As with any Krylov subspace iteration type algorithm, IDR requires an effective implementation of the matrix-by-vector multiplication, which in our case is evaluating expression $\left(I - \delta L \cdot L_0^{-1}\right)w$ for a given vector w. For w being a 3D grid function, this operation consists of applying the inverse of operator L_0 described in section 2, pointwise multiplication of the result by scalar function $\gamma(x,y,z;\omega,\beta)$ (once again, a 3D grid function) and subtracting the last result from w. The IDR itself uses a few BLAS (Basic Linear Algebra Subprograms) level 1 [27] operations with vectors. All needed functionality (including the FFT) is available in the highly optimized software library Intel® MKL [28] we use in our solver.

Our numerical experiments show that preconditioner L_0 is effective for models of moderate complexity. In situations of strong lateral variations, high contrasts or high frequency ω, convergence rate of the iterative method may be slow or even may fail to converge. To handle this issue we suggest a second-level "$\delta L \cdot L_0^{-1}$-based" preconditioner [29]. The reason relies on the fact that the cost of evaluation of $\delta L \cdot L_0^{-1}$ is small enough, especially if it is allowed to sacrifice accuracy for the sake of faster computations.

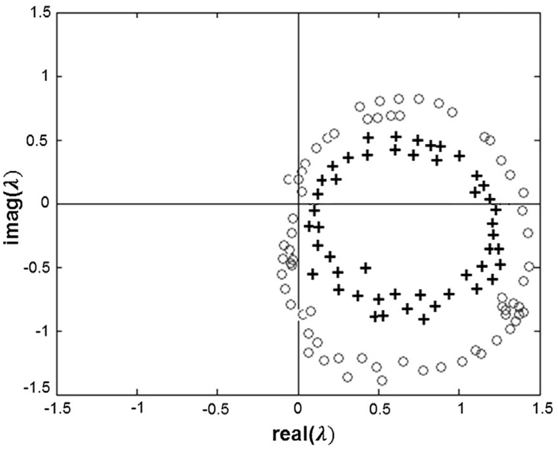

Fig. 1. Eigenvalues of two preconditioned operators: circles correspond to a single-stage preconditioner; pluses correspond to a two-stage preconditioner.

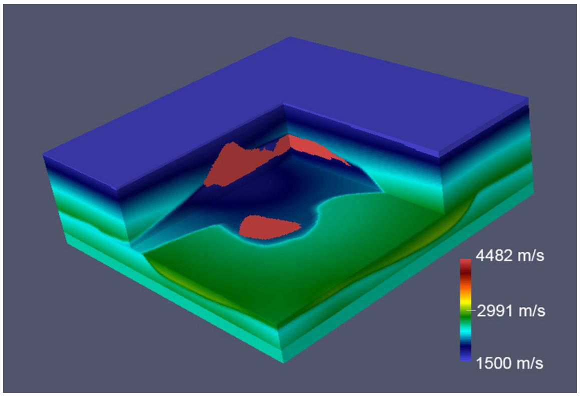

Fig. 2. A 3D view of the SEG/EAGE salt P-wave velocity model (13.5 km × 13.5 km × 4.2 km).

We introduce a second-level preconditioner for equation (27)

$$\left(I - \delta L \cdot L_0^{-1}\right)\left(I - \alpha \cdot \delta L \cdot L_0^{-1}\right)^{-1} \tilde{v} = g, \tag{28}$$

where $0 < \alpha < 1$ is some parameter, $\tilde{v} = \left(I - \alpha \cdot \delta L \cdot L_0^{-1}\right) L_0 u$. Using a truncated Neumann series decomposition of the second preconditioner in equation (28) one obtains an expression for the numerical implementation:

$$\left(I - \delta L \cdot L_0^{-1}\right)\left(I + \alpha \delta L \cdot L_0^{-1} + \cdots + \left(\alpha \delta L \cdot L_0^{-1}\right)^N\right) \tilde{v} = g, \tag{29}$$

where N is some predefined parameter.

The impact of the second-level preconditioner is briefly illustrated in Fig. 1 where we compare the eigenvalues of operator $\left(I - \delta L \cdot L_0^{-1}\right)$ and a two-stage preconditioner given by the left-hand side operator in equation (29). This experiment has been performed using the SEG/EAGE salt model [30] (Fig. 2) at a frequency of 8 Hz. In Fig. 1 we present so-called extremal eigenvalues, i.e., eigenvalues with a maximum magnitude or a maximum/minimum real/imaginary part. These eigenvalues have been estimated by the Arnoldi [31] method. Eigenvalues corresponding to the two-stage preconditioner are better clustered around unity on the complex plane and better separated from the origin. This leads to a better convergence for Krylov-type iterative methods (Fig. 3). For more information regarding the influence of the two-level preconditioner refer to Appendix A.

As one might notice, in particular from Fig. 3, the usage of the second-level preconditioner is not free, namely, it increases the computational time of each iteration. However, cost can be decreased if inversion of L_0 in (29) is replaced with a less costly operation. For instance, we can decrease the approximation order of the finite difference schemes with respect to variable z used to solve boundary value problems (16), (17) – instead of the 6th order finite difference scheme use the

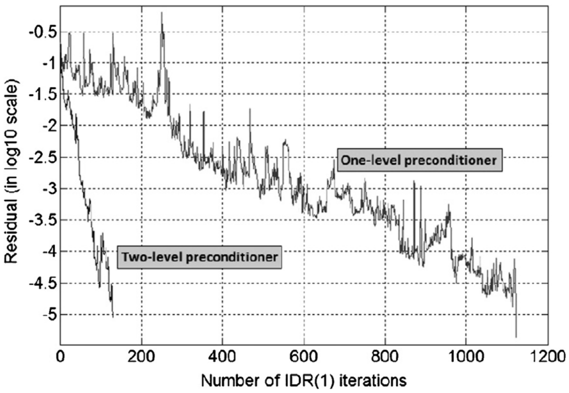

Fig. 3. Convergence curves of the iterative solver using a one-level and a two-level preconditioners. Note that computation time is 5.5 times longer for the two-level preconditioner. The two-level preconditioner (29) was used with parameters $\alpha = 0.75$ and $N = 12$.

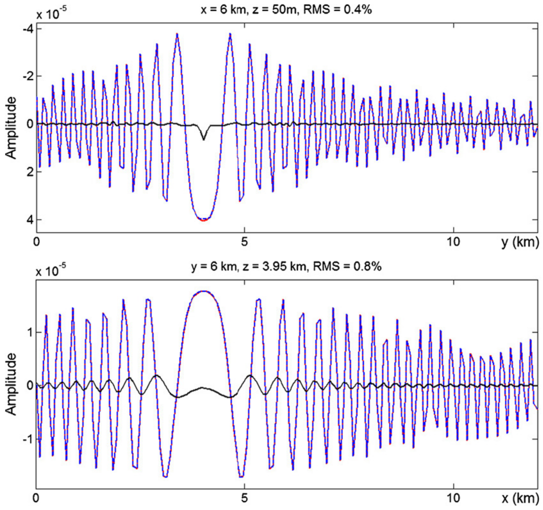

Fig. 4. 1D horizontal profiles (real part) of solutions for a homogeneous model at locations $x = 6000$ m, $z = 50$ m (top) and $y = 6000$ m, $z = 3950$ m (bottom). Exact solutions are dashed blue lines, our solver solutions – red lines and residuals multiplied by 10 – black lines.

2nd order scheme. One more optimization opportunity is decreasing the number of the spatial frequencies to be taken into account (see the last paragraph of Section 3) while inverting L_0 to build the second-level preconditioner.

5. Numerical experiments

In this section we show the results of several numerical experiments that have been performed using the solver. Computations were run on one cluster node with an Intel® Xeon® CPU E5-2670 v2 @ 2.50 GHz. As a stopping criterion for an IDR we use a 10^{-5} threshold for the relative residual of the L_2-norm.

5.1. Homogeneous medium

The first example comprises a domain of size 12 km × 12 km × 4 km filled with an acoustic medium with homogeneous velocity 1000 m/s discretized over a 151 × 151 × 161 uniform grid with steps $h_x = h_y = 80$ m, $h_z = 25$ m. The wave is excited by a point source located at the point $x = 4000$ m, $y = 4000$ m, $z = 25$ m. For this experiment we used PMLs on both top and bottom boundaries of width 20 grid points. The width of the absorbing zones in lateral directions are 30 grid points.

The solution at frequency $\nu = 5$ Hz obtained by our solver is compared to the exact solution $\frac{-e^{ikr}}{4\pi r}$, where r is the distance from the observation point to the source position, $k = \frac{2\pi\nu}{c} = 0.01\pi\ \text{m}^{-1}$. Note that horizontal grid steps of 80 m correspond to 2.5 grid steps per wavelength at this frequency and velocity. In Fig. 4, 1D profiles of these solutions along a few lines are given which show the good agreement with the exact solution.

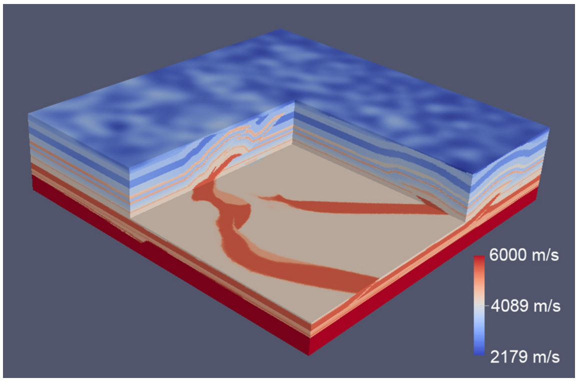

Fig. 5. A 3D view of the SEG/EAGE overthrust P-wave velocity model (19.8 km × 19.8 km × 4.65 km).

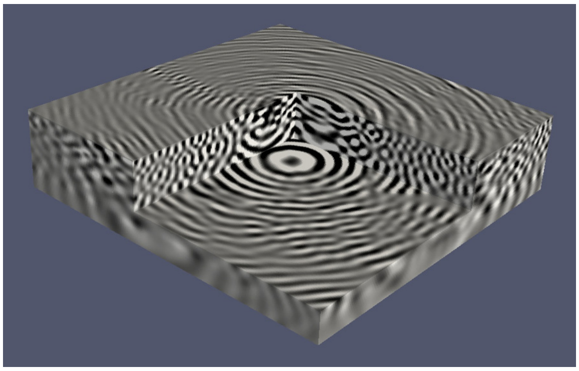

Fig. 6. A 3D view of the real part of the frequency–domain wavefield for the SEG/EAGE overthrust model.

5.2. 3D SEG/EAGE overthrust model

In this test the 3D SEG/EAGE overthrust model [30] is used. P-wave velocity is represented over a 660 × 660 × 155 grid (Fig. 5) with cell sizes of 30 m in all dimensions resulting in 10 grid points per minimal wavelength. At the top boundary a free surface boundary condition was used. The width of the PML at the bottom boundary is 30 grid points and the width of the absorbing zones in the lateral directions is 60 grid points. The solution was computed for the frequency of 7 Hz with source coordinates $x = 9900$ m, $y = 9900$ m and $z = 30$ m.

The simulated wavefield is depicted in Fig. 6. To verify the solution obtained with our solver we compare it to the solution obtained with a time-domain finite-difference (TDFD) scheme. Snapshots and 1D profiles along the x-axis of both solutions are presented in Figs. 7 and 8, respectively. Their close agreement is clearly seen (RMS difference is 2.8–3.2%). The total computation time for the iterative solver is 21.6 min.

5.3. Convergence rate of the iterative solver in a complex model

To estimate the convergence rate and the computational time in the media with strong lateral contrasts and the impact of the second-level preconditioner consider the model presented in Fig. 2. We discretize this model with a uniform grid of 540 × 540 × 168 points and cell sizes of 25 m. At the top boundary a free-surface boundary condition was used. The width of the PML at the bottom boundary is 30 grid points and the width of the absorbing zones in the lateral directions is 60 grid points. The source is placed in the middle of the target area at a depth of 25 m. We have run two types of experiments: with preconditioner L_0 only and with the second-level preconditioner (29) with parameters $\alpha = 0.75$ and $N = 12$. As one can see from the Table 1 if we use a basic preconditioner L_0 poor convergence is an issue, particularly at higher frequencies. Use of the second-level preconditioner provides a reasonable convergence rate and improves the total performance.

5.4. Benchmarking against a known iterative solver

For purposes of performance comparison, we considered a solver developed by Plessix (see [9]). For the sake of brevity, everywhere below that solver is referred to as PS. Following the paper, the same simulations were performed in the parts of the SEG/EAGE overthrust and salt models. In Table 2, the computation times (row t_{os}) and the number of iterations (row n_{iter}) are presented as well as the data mentioned in [9] for PS (see rows t_{PS} and $n_{\text{iter,PS}}$). By the number of iterations we

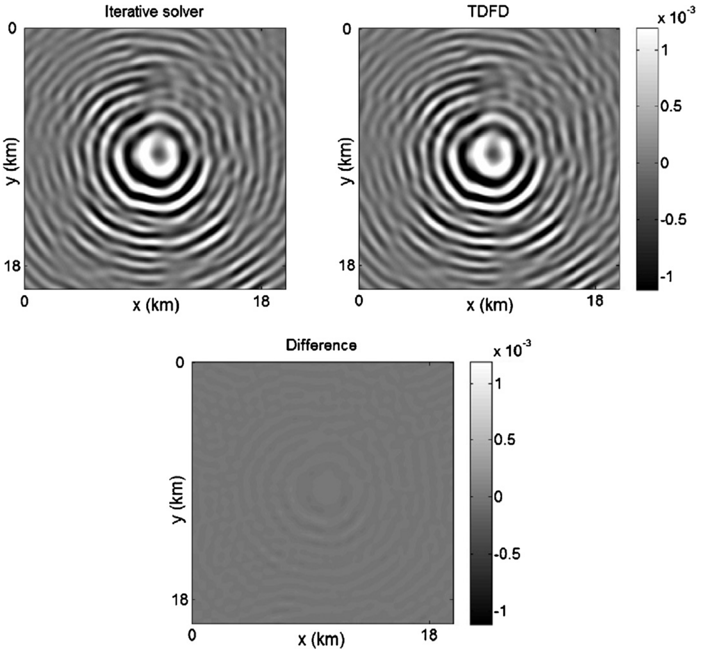

Fig. 7. Snapshots of the solution for the SEG/EAGE overthrust model at $z = 1500$ m; the top left picture computed by our solver, the top right picture computed with TDFD and the bottom one is their difference.

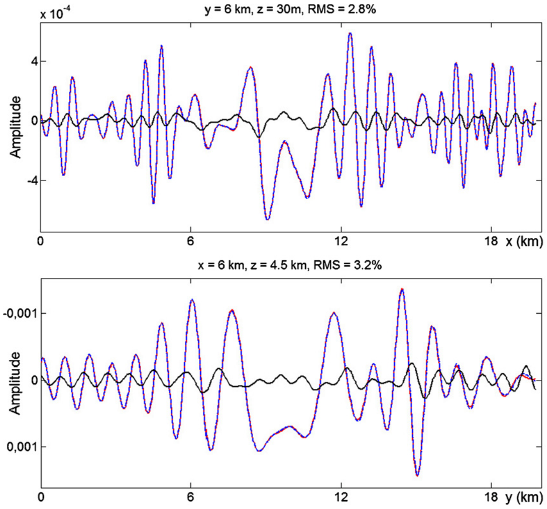

Fig. 8. 1D horizontal profiles (real part) of solutions for the SEG/EAGE overthrust model at locations $y = 6000$ m, $z = 30$ m (top) and $x = 6000$ m, $z = 4500$ m (bottom). Exact solutions are dashed blue lines, our solver solutions – red lines and residuals multiplied by 5 – black lines.

Table 1
Convergence of the algorithm under different conditions of the source frequency and preconditioner.

		8 Hz	15 Hz
L_0 preconditioner	Number of iterations	1715	No convergence
	Computation time	106 min	No convergence
Second-level preconditioner	Number of iterations	162	624
	Computation time	54 min	207 min

Table 2
Our solver vs PS: model sizes are in grid points; h (m) stands for grid steps (the same in all directions) used to approximate the models; t_{os} (min) – computation time (in minutes) for our solver; t_{PS} (min) – computation time for PS. See more details in the text.

	SEG overthrust model		SEG salt model	
ν (Hz)	5	10	5	10
model size	261 × 261 × 110	471 × 471 × 160	295 × 295 × 140	520 × 520 × 210
h (m)	100	50	60	30
t_{os} (min)	3	127	7	293
n_{iter}	24	131	47	225
t_{PS} (min)	126	1380	154	1111
$n_{\text{iter,PS}}$	125	252	125	210

mean the number of iterations of IDR(1) only and do not take into account the second-level preconditioner iterations. Note that PS is based on BiCGSTAB algorithm [32].

We tested our method on Intel® Core™ i7-3770K @3.5 GHz with all parallelization options switched off, whereas PS was run on AMD Opteron™ @2.2 GHz.

Definitely, data in Table 2 are not applicable for ‘apple-to-apple’ comparison due to the difference in hardware and software, code optimizations applied and, possibly, some other factors. Having no access to the actual PS code, it is problematic to estimate its computation time on our hardware and with our settings. However, one might infer that PS is less sensitive to the frequency increase then our solver: two times increase of the frequency results in a factor of 5 increase of iterations for our solver and a factor of 2 increase for PS. One of the reasons for that is different preconditioners applied in these solvers. Both of them are shifted Laplace operators but built in a different way. In our case a shifted Laplace operator is constructed as some 1D approximation of an original 3D sound velocity. This greatly benefits us while inverting the preconditioner and this way we satisfy the requirement for a preconditioner to be ’easily’ inverted. For PS a preconditioner is a shifted Laplace operator for the original sound velocity which is approximately inverted by applying of one step of the multigrid V-cycle. Different preconditioners lead to different spectral properties of the preconditioned operator involved in computations.

Another fact that Table 2 reflects is that for higher frequencies the price per iteration increases. But this is in a good agreement with respective increase of vectors’ size.

6. Conclusions

We developed a frequency–domain iterative solver for 3D acoustic wave propagation problems. The boundary value problem for the Helmholtz equation in an unbounded domain is approximated by posing the problem in a bounded domain surrounded with PMLs and absorbing zones. The operator of the Helmholtz equation is considered as a perturbation of a shifted Laplace operator for a layered model. After preconditioning by the shifted Laplace operator, the operator of the Helmholtz equation becomes a perturbation of the unity operator and the respective equation can be effectively solved by modern Krylov-type subspace iteration methods. The shifted Laplace operator for a layered model can be inverted using FFTs and solving the resulting systems of linear equations with banded coefficient matrices. Matrix-by-vector multiplication in iterative inversion of the preconditioned operator involves inversion of the preconditioner and pointwise multiplication of grid functions.

The solver has been verified by simulation in several models, and the results have been benchmarked against known solutions for a homogeneous velocity model with the fundamental solution of the Helmholtz equation in R^3, and for the SEG/EAGE overthrust model against a solution obtained by using TDFD. The accuracy of the proposed method with relatively reasonable computational time justifies using the solver in FWI problems.

In the future, we plan to extend the approach to the elastic wave modeling.

Acknowledgements

We greatly appreciate the hard work of Vincent Etienne, Michael Jervis, and Robert Smith who reviewed our manuscript and suggested numerous ideas to improve it. Special thanks to Jean Virieux and an anonymous reviewer for their insightful notes and interesting references. We thank Saudi Aramco for permission to publish this paper. Three of the authors (Victor Kostin, Dmitry Neklyudov, and Vladimir Tcheverda) have been supported by RFBR (Grant Nos. 15-35-20022, 15-05-01310, 15-55-20004, 16-05-00800 and 17-01-00416). Vladimir Tcheverda was partially sponsored by the Ministry of Education and Science of the Republic of Kazakhstan Grant No. 1771/GF4. We are grateful to the Siberian Supercomputing Center and Moscow State University Supercomputing Center for providing us with computational resources.

Appendix A. Spectrum of a double-preconditioned operator

In this section we describe the impact of second-level preconditioner (28) while solving equation (27). In particular, for different values of parameter α we derive the area of possible locations of eigenvalues of matrix $\delta L \cdot L_0^{-1}$ for the second-level preconditioner to be reasonable to apply.

For the sake of convenience we denote matrix $\delta L \cdot L_0^{-1}$ by B. Then equation (27) takes the form

$$(I-B)v=f. \tag{A.1}$$

The Neumann series expansion of the inverse to the left-hand side operator in (A.1) gives the representation for the unknown function

$$v=f+Bf+B^2f+\cdots. \tag{A.2}$$

This series converges if all eigenvalues λ of matrix B belong to unit disc $|\lambda|<1$. If this condition is not satisfied one might precondition equation (A.1) a second time by operator $(I-\alpha B)^{-1}$ and come to equation

$$(I-B)(I-\alpha B)^{-1}w=f \tag{A.3}$$

with left-hand side operator $C=(I-B)\,(I-\alpha B)^{-1}$ being a function of matrix B. Its eigenvalues μ are related to eigenvalues λ of matrix B in the following way

$$\mu=\frac{1-\lambda}{1-\alpha\lambda}. \tag{A.4}$$

Similarly to formula (A.2), we might represent the unknown function w in (A.3) as

$$w=f+(I-C)f+(I-C)^2f+\cdots.$$

The convergence criterion for the series in the right-hand side is that all eigenvalues μ of matrix C have to be inside disc $|\mu-1|<1$. Applying the second-level preconditioner we aim to get a new operator, i.e. operator C, that satisfies this criterion. In other words, the second-level preconditioner is beneficial if operator B is such that transform (A.4) maps its eigenvalues λ onto disc $|\mu-1|<1$. Below we present the domain that is mapped onto this disc depending on the different values of parameter α.

- For $\alpha\in(0.5,1)$ this is an exterior of disc

  $$|\lambda-\frac{\alpha}{2\alpha-1}|<\left|\frac{1-\alpha}{2\alpha-1}\right| \tag{A.5}$$

  (for $\alpha=0.75$ this disc is a dark gray {3} area on the left image in Fig. A.1). Inversion of operator $(I-\alpha B)$ in (A.3) by the Neumann series expansion

  $$(I-\alpha B)^{-1}=I+\alpha B+(\alpha B)^2+\cdots \tag{A.6}$$

  implies the necessary condition

  $$|\alpha\lambda|<1 \tag{A.7}$$

  to be satisfied. Hence, the area of interest is an intersection of exterior of disc (A.5) with disc $|\lambda|<1/\alpha$ (for $\alpha=0.75$ this is a union of bright grey {2} and grey {1} domains on the left image in Fig. A.1). Note, that if all eigenvalues λ are in disc $|\lambda|<1$ (a bright grey {2} domain on the left image in Fig. A.1) then the second-level preconditioner is not needed at all, since the convergence criterion for the series in (A.2) is already satisfied.
- For $\alpha\in[1/3,1/2)$ this is an interior of disc (A.5) (for $\alpha=0.35$ this disc is a bright grey {2} plus grey {1} domain on the right image in Fig. A.1). Taking into account the necessary condition (A.7) the area of interest is an intersection of disc (A.5) with disc (A.7) (for $\alpha=0.35$ this area is depicted by bright grey {2} color on the right image in Fig. A.1).
- For $\alpha\in(0,1/3)$ the domain we search for is defined in the same way as in the previous case, but disc (A.5) is completely located inside disc (A.7) and that is why it is the whole disc (A.5).
- For $\alpha=1/2$ formula (A.5) is not defined. In this case the half-plane $\mathrm{Re}(\lambda-1)<0$ stands instead of disc (A.5). Intersection of this half-plane with disc (A.7) is the area we search for.

A union of the aforementioned domains gives an area of possible locations for eigenvalues λ such that the presented second-level preconditioning technique (see formula (A.3)) is beneficial. Its boundary is a cardioid-like line $\lambda(\alpha)=x(\alpha)+iy(\alpha)$ parametrized in the following way

$$x(\alpha)=\frac{1}{2\alpha}+\frac{1}{\alpha^2}-\frac{1}{2\alpha^3},\quad y(\alpha)=\pm\sqrt{\frac{1}{\alpha^2}-x^2(\alpha)},\qquad \alpha\in[\frac{1}{3},1]. \tag{A.8}$$

It represents the intersection points of boundaries of discs (A.5) and (A.7) (see the bold black line in Fig. A.2).

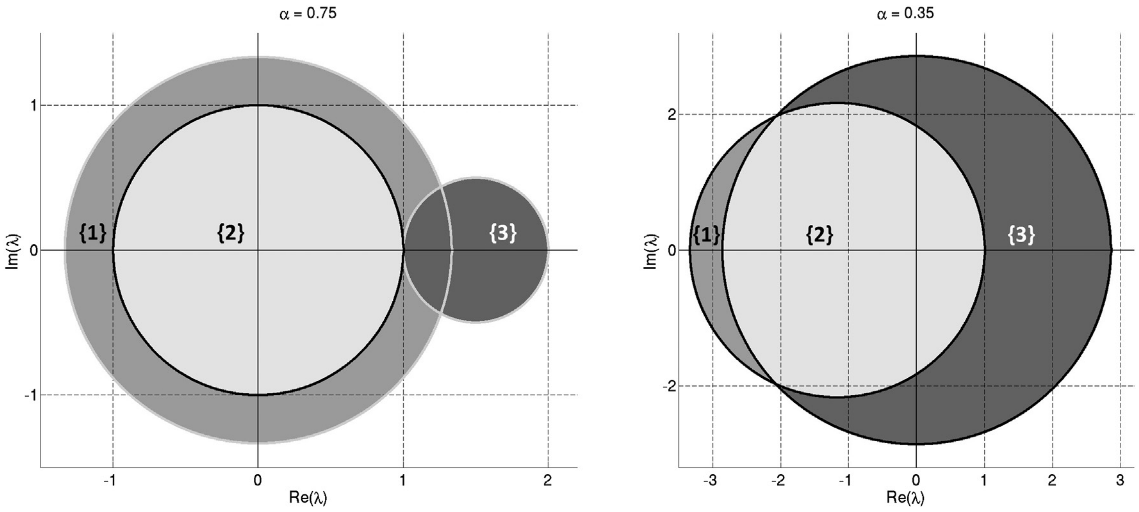

Fig. A.1. The left image: the domain depicted in bright grey {2} and grey {1} reflects the possible locations for eigenvalues of operator $\delta L \cdot L_0^{-1}$ in order to the second-level preconditioner (see formula (A.3)) be beneficial with $\alpha = 0.75$. The right image: the domain depicted in bright grey {2} reflects the possible locations for eigenvalues of operator $\delta L \cdot L_0^{-1}$ in order to the second-level preconditioner be beneficial with $\alpha = 0.35$.

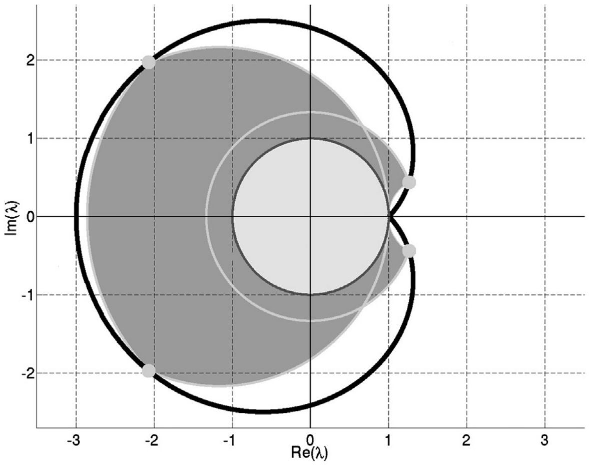

Fig. A.2. Cardioid-like bold black line encompasses the area of all possible eigenvalues of operator $\delta L \cdot L_0^{-1}$ for which the second-level preconditioner defined by formula (A.3) is reasonable to be applied. This line passes through intersections of the respective circles (see Fig. A.1), in particular ones depicted by bold bright grey dots.

It is worth mentioning that when solving the problem numerically we consider only the finite number of items in Neumann series (A.6) and the domain surrounded by the cardioid-like line (A.8) will be replaced by its subdomain containing the unit disc (see the bright grey domain in Fig. A.2). It will increase with the increasing number of items in the Neumann series remaining its subdomain.

References

[1] W. Mulder, R.-E. Plessix, How to choose a subset of frequencies in frequency–domain finite-difference migration, Geophys. J. Int. 158 (2004) 801–812, http://dx.doi.org/10.1111/j.1365-246X.2004.02336.x.

[2] J. Virieux, S. Operto, H. Ben-Hadj-Ali, R. Brosssier, V. Etienne, F. Sourber, L. Giraud, A. Haidar, Seismic wave modeling for seismic imaging, Lead. Edge 28 (2009) 538–544, http://dx.doi.org/10.1190/1.3124928.

[3] C. Shin, Y.H. Cha, Waveform inversion in the Laplace domain, Geophys. J. Int. 173 (2008) 922–931, http://dx.doi.org/10.1111/j.1365-246X.2008.03768.x.

[4] B.R. Vainberg, Principles of radiation, vanishing attenuation and limit amplitude in the general theory of partial differential equations, Usp. Mat. Nauk 21 (1966) 115–194 (in Russian).

[5] K.T. Nihei, X. Li, Frequency response modelling of seismic waves using finite difference time domain with phase sensitive detection (TD-PSD), Geophys. J. Int. 169 (2007) 1069–1078, http://dx.doi.org/10.1111/j.1365-246X.2006.03262.x.

[6] V. Etienne, T. Tonellot, P. Thierry, V. Berthoumieux, C. Andreolli, Optimization of the seismic modeling with the time-domain finite-difference method, in: 84th SEG Annual Meeting Expanded Abstracts, 2014, pp. 3536–3540.

[7] R.E. Plessix, Three-dimensional frequency–domain full-waveform inversion with an iterative solver, Geophysics 74 (2009) WCC149–WCC157, http://dx.doi.org/10.1190/1.3211198.

[8] Y.A. Erlangga, F.J. Herrmann, Full-waveform inversion with Gauss–Newton–Krylov method, in: 79th SEG Annual Meeting Expanded Abstracts, SEG, 2009, pp. 2357–2361.

[9] R.E. Plessix, A Helmholtz iterative solver for 3D seismic-imaging problems, Geophysics 72 (2007) SM185–SM194, http://dx.doi.org/10.1190/1.2738849.

[10] Y.A. Erlangga, F.J. Herrmann, An iterative multilevel method for computing wavefields in frequency–domain seismic inversion, in: 78th SEG Annual Meeting Expanded Abstracts, SEG, 2008, pp. 1957–1960.

[11] L. Trefethen, D. Bau, Numerical Linear Algebra, SIAM, Philadelphia, PA, 1997.

[12] D. Neklyudov, V. Tcheverda, A Helmholtz iterative solver without of finite-difference approximations, in: 72nd EAGE Conference and Exhibition Expanded Abstracts, 2010, p. G006.

[13] Y.A. Erlangga, C. Vuik, C.W. Oosterlee, On a class of preconditioners for solving the Helmholtz equation, Appl. Numer. Math. 50 (2004) 409–425, http://dx.doi.org/10.1016/j.apnum.2004.01.009.

[14] C.W. Oosterlee, C. Vuik, W.A. Mulder, R.-E. Plessix, Shifted-Laplacian preconditioners for heterogeneous Helmholtz problems, in: Advanced Computational Methods in Science and Engineering, Springer, 2010, pp. 21–46.

[15] T. Airaksinen, S. Mönkölä, Comparison between the shifted-Laplacian preconditioning and the controllability methods for computational acoustics, J. Comput. Appl. Math. 234 (2010) 1796–1802, http://dx.doi.org/10.1016/j.cam.2009.08.030.

[16] Y.A. Erlangga, R. Nabben, On a multilevel Krylov method for the Helmholtz equation preconditioned by shifted Laplacian, Electron. Trans. Numer. Anal. 31 (2008) 403–424.

[17] A. Sheikh, D. Lahaye, L.G. Ramos, R. Nabben, C. Vuik, Accelerating the shifted Laplace preconditioner for the Helmholtz equation by multilevel deflation, J. Comput. Phys. 322 (2016) 473–490, http://dx.doi.org/10.1016/j.jcp.2016.06.025.

[18] O. Pankratov, D. Avdeyev, A. Kuvshinov, Electromagnetic field scattering in a heterogeneous earth: a solution to the forward problem, Phys. Solid Earth 31 (3) (1995) 201–209.

[19] J.-P. Berenger, Three-dimensional perfectly matched layer for the absorption of electromagnetic waves, J. Comput. Phys. 127 (1996) 363–379, http://dx.doi.org/10.1006/jcph.1996.0181.

[20] I. Singer, E. Turkel, A perfectly matched layer for the Helmholtz equation in a semi-infinite strip, J. Comput. Phys. 201 (2004) 439–465, http://dx.doi.org/10.1016/j.jcp.2004.06.010.

[21] F. Collino, C. Tsogka, Application of the PML absorbing layer model to the linear elastodynamic problem in anisotropic heterogeneous media, Tech. rep. RR-3471, INRIA, 1998.

[22] K.R. Rao, D.N. Kim, J.J. Hwang, Fast Fourier Transform: Algorithms and Applications, Springer, Netherlands, 2010.

[23] P.G. Petropoulos, Reflectionless sponge layers as absorbing boundary conditions for the numerical solution of Maxwell equations in rectangular, cylindrical, and spherical coordinates, SIAM J. Appl. Math. 60 (2000) 1037–1058, http://dx.doi.org/10.1137/S0036139998334688.

[24] P. Sonneveld, M.B. van Gijzen, IDR(s): a family of simple and fast algorithms for solving large nonsymmetric systems of linear equations, SIAM J. Sci. Comput. 31 (2008) 1035–1062, http://dx.doi.org/10.1137/070685804.

[25] M.B. van Gijzen, P. Sonneveld, Algorithm 913: an elegant IDR (s) variant that efficiently exploits biorthogonality properties, ACM Trans. Math. Softw. 38 (1) (2011) 5, http://dx.doi.org/10.1145/2049662.2049667.

[26] M.B. van Gijzen, G.L. Sleijpen, J.P.M. Zemke, Flexible and multi-shift induced dimension reduction algorithms for solving large sparse linear systems, Numer. Linear Algebra Appl. 22 (1) (2015) 1–25, http://dx.doi.org/10.1002/nla.1935.

[27] BLAS (Basic Linear Algebra Subprograms), http://www.netlib.org/blas/.

[28] Intel® Math Kernel Library (Intel® MKL), https://software.intel.com/en-us/intel-mkl.

[29] D. Neklyudov, M. Dmitriev, M. Belonosov, V. Tcheverda, Frequency–domain iterative solver for 3D acoustic wave equation with two-stage semi-analytical preconditioner, in: 76th EAGE Conference and Exhibition Expanded Abstracts, 2014, Tu G105 09.

[30] F. Aminzadeh, J. Brac, T. Kuntz, 3-D salt and overthrust models, in: SEG/EAGE Modelling Series, vol. 1, SEG, Tulsa, OK, 1997.

[31] Y. Saad, Iterative Methods for Sparse Linear Systems, 2nd ed., SIAM, Philadelphia, PA, 2003.

[32] H.A. van der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM J. Sci. Stat. Comp. 13 (2) (1992) 631–644, http://dx.doi.org/10.1137/0913035.