Computers and Geosciences 111 (2018) 223–234

Contents lists available at ScienceDirect

Computers and Geosciences

journal homepage: www.elsevier.com/locate/cageo

Research paper

Acoustic 3D modeling by the method of integral equations

M. Malovichko [a,*], N. Khokhlov [a], N. Yavich [a], M. Zhdanov [b,c,a]

[a] Moscow Institute of Physics and Technology, Dolgoprudny, Moscow Region, 141701, Russia
[b] University of Utah, Salt Lake City, UT 84112, USA
[c] TechnoImaging, Salt Lake City, UT 84107, USA

ARTICLE INFO

Keywords:
Seismology
Acoustics
Integral equation
Seismic modeling

ABSTRACT

This paper presents a parallel algorithm for frequency-domain acoustic modeling by the method of integral equations (IE). The algorithm is applied to seismic simulation. The IE method reduces the size of the problem but leads to a dense system matrix. A tolerable memory consumption and numerical complexity were achieved by applying an iterative solver, accompanied by an effective matrix-vector multiplication operation, based on the fast Fourier transform (FFT). We demonstrate that, the IE system matrix is better conditioned than that of the finite-difference (FD) method, and discuss its relation to a specially preconditioned FD matrix. We considered several methods of matrix-vector multiplication for the free-space and layered host models. The developed algorithm and computer code were benchmarked against the FD time-domain solution. It was demonstrated that, the method could accurately calculate the seismic field for the models with sharp material boundaries and a point source and receiver located close to the free surface. We used OpenMP to speed up the matrix-vector multiplication, while MPI was used to speed up the solution of the system equations, and also for parallelizing across multiple sources. The practical examples and efficiency tests are presented as well.

1. Introduction

Modeling of wave propagation within the Earth is the cornerstone of seismic full-waveform inversion (FWI). Performing inversion in the frequency domain has many advantages, the most important of which is the possibility to invert only a few frequencies in a sequential manner, avoiding trapping to local minima (Virieux and Operto, 2009).

One of the most effective approaches to large frequency-domain simulation of acoustic waves is the finite-difference (FD) method accompanied with an iterative solver. The sparse direct solvers were also considered (Operto et al., 2007), but they are quite memory consuming. Performance of FD iterative solvers depends critically on the choice of the preconditioner applied. There is a number of preconditioners designed to date (Lahaye et al., 2017), however preconditioning usually requires a complex implementation. In this paper, we considered integral equations (IE) modeling, which reduces the size of the problem but increases the complexity of the matrix-vector multiplication.

In the method of IE (Morse and Feshbach, 1953; de Hoop, 1958; Aki and Richards, 1980; Carcione et al., 2002; Zhdanov, 2002, 2015) the total field is split into a background part due to a host model and an anomalous part due to an anomalous domain. The background part of the total field is usually calculated in an analytical or semi-analytically manner, whereas the anomalous part requires a numerical solution. The solution within the anomalous model is found in form of the Green's integral by a numerical method. Therefore, if the model can be divided into an analytically-solvable host model and a compact anomalous domain, it may speed-up computations. The IE system matrix is dense, although relatively small. This fact has been the main obstacle for practical application of the method for a long time. In a straightforward implementation with the explicit matrix allocation and factorization, the computational burden becomes prohibitive for typical problems, encountered in seismology. Yet, some studies, many of them for the two-dimensional solution, have been reported (Johnson et al., 1983; Freter, 1992; Fu et al., 1997; Zhang and Ulrych, 2000; Fu, 2003). The application of the IE method in the framework of 2D inverse problems was considered in a number of publications (Wu and Toksoz, 1987; Abubakar et al., 2003; Jakobsen and Ursin, 2015). There have been also successful attempts to solve IE for 2D lossless media by means of so-called renormalized scattering series that converge for any velocity contrasts (Jakobsen and Wu, 2016; Yao et al., 2016). One of the effective applications of the IE method for large-scale 3D seismic modeling has been performed by (Abubakar and Habashy, 2013). These authors implemented the integral operator for the homogeneous space via the 3D FFT and applied the BiCGStab solver to the resulting system of linear

* Corresponding author.
E-mail address: malovichko.mikhail@gmail.com (M. Malovichko).

https://doi.org/10.1016/j.cageo.2017.11.015
Received 20 April 2017; Received in revised form 14 November 2017; Accepted 20 November 2017
Available online 24 November 2017

equations achieving the cost per iteration of $O(N\log N)$, where N is the number of model cells. That study is also interesting in that it develops a specially designed preconditioner, based on the extension for the seismic case of the contraction preconditioner originally developed for the IE method of electromagnetic modeling (Zhdanov and Fang, 1997; Hursán and Zhdanov, 2002; Zhdanov, 2002) and based on the energy inequality for electromagnetic fields (Pankratov et al., 1995). Similar approach has been recently reported in (Osnabrugge et al., 2016), who solved Helmholtz equation by means of Born series, making use of contraction preconditioner to guarantee convergence in a lossy medium, which is similar to (Zhdanov and Fang, 1997). Performance of this kind of preconditioners in real applications requires more research, since seismic models have low or no attenuation apart from absorbing boundary conditions. In contrast, the contraction-operator approach is routinely used in geophysical electromagnetic modeling because of strong attenuation of diffusive electromagnetic fields (Yavich and Zhdanov, 2016).

It is not always realized, that the system matrix, arising from the IE, is related to the FD system matrix, preconditioned with inverse of the FD system matrix for the background problem. The more closely the background model follows the original model, the better the IE system matrix is conditioned. This is our main motivation to develop the method of IE for a layered host model. Our matrix-vector multiplication algorithm for layered background media has numerical complexity of $O(N^{4/3}\log N)$ and linear memory consumption.

The IE solution may contain spurious reflections from the boundaries of the anomalous domain. Previously, research in this field focused on the accurate computation of reflections from the irregular boundaries of the anomalous domain, e.g (Fu, 2003). It has been shown that it is possible to impose absorbing boundary on the IE solution (Alles and van Dongen, 2009; Osnabrugge et al., 2016). In the present study, we partially mitigate this problem using a layered background model, since seismic models are always stratified to some degree. More work on application of the absorbing boundary conditions for the IE method should be performed in the future, however, to improve the accuracy of the modeling results.

There are many publications on numerical implementation and parallelization of the IE method for electromagnetic simulation, e.g. (Avdeev et al., 2002; Abubakar and van den Berg, 2004; Čuma et al., 2017; Kruglyakov and Bloshanskaya, 2017). Acoustic modeling is rather different in the way how the elements of the integral operator are computed. This important point is addressed in our paper as well. We discuss different strategies for shared- and distributed-memory parallelization using OpenMP and MPI, in connection with the fact, that the system matrix is dense. We propose a new method for the matrix-vector multiplication that asymptotically has linearithmic complexity estimate of $O(N\log N)$.

The paper is organized as follows. In Section 2 we discuss the problem formulation, the numerical method and its implementation. Section 3 is devoted to numerical experiments based on the developed method. We benchmark our code with an open-source time-domain finite-difference software and present several parallelization scalability tests. Concluding remarks are given in Section 4. A simple and numerically efficient solution for the Green's function in a layered medium is given in Appendix A.

Practical examples of using our IE modeling scheme for FWI are given in the paper (Malovichko et al., 2017).

2. Seismic modeling by the method of integral equations

2.1. Problem formulation

We consider a model consisting of a layered host medium and a 3D anomalous volume, $\mathscr{D}$. The host medium is a stack of L layers, each of them is characterized with density, ρ_j, and P-wave velocity, c_j, where the subscript denotes the host layer index,$j = 0..L-1$. We assume that the anomalous volume, $\mathscr{D}$, is confined to n-th layer with $\rho_b \equiv \rho_n$ and $c_b \equiv c_n$. The anomalous volume has the same background density ρ_b, and arbitrary distribution of velocity $c = c(\boldsymbol{r})$, where $\boldsymbol{r}$ is the position vector of the observation point. Outside of the anomalous volume, velocity $c(\boldsymbol{r})$ is equal to the background velocity. Let ω be the circular frequency. Assuming a lossless medium, we define the wavenumber $k = \omega/c$, background wavenumber $k_b = \omega/c_b$ and parameter $\psi = 1/c^2 - 1/c_b^2$.

The pressure field, p, everywhere within the host layer, including the anomalous domain, satisfies the following Helmholtz equation:

$$-\Delta p - \frac{\omega^2}{c^2} p = f, \tag{1}$$

where Δ is the Laplacian and f is the source term. The total pressure p can be represented as $p = p_a + p_b$, where p_b is the background part due to the host medium and p_a is the anomalous (scattered) part due to the volume with anomalous velocity.

The background field, p_b, satisfies the following Helmholtz equation everywhere within the host layer:

$$-\Delta p_b - \frac{\omega^2}{c_b^2} p_b = f. \tag{2}$$

Using the standard separation technique (Zhdanov, 2002) we arrive to the following Helmholtz equation with respect to the anomalous field, p_a,

$$-\Delta p_a - \frac{\omega^2}{c_b^2} p_a = \omega^2 (p_a + p_b)\psi. \tag{3}$$

The solution of this equation is given by the following expression:

$$p_a(\boldsymbol{r}) = \omega^2 \int_{\mathscr{D}} g(\boldsymbol{r}|\boldsymbol{r}')(p_a(\boldsymbol{r}') + p_b(\boldsymbol{r}'))\psi(\boldsymbol{r}')d^3\boldsymbol{r}', \tag{4}$$

where $g(\boldsymbol{r}|\boldsymbol{r}')$ is the background Green's function, the primed coordinates denote the source position. The problem is solved in two main steps. First, expression (4) is treated as an integral equation with respect to field p_a within domain $\mathscr{D}$ ($\boldsymbol{r} \in \mathscr{D}$). We call this equation *a domain equation*. Once the field inside the scatterer is found, equation (4) is used again to calculate field p_a in all required exterior points ($\boldsymbol{r} \notin \mathscr{D}$). In this case, expression (4) is called *a field equation*. The first step is by far the most time-consuming and storage-demanding part for realistic problems and the subsequent discussion is devoted to that part.

By adding p_b to both sides of (4) we obtain the following IE with respect to the total field, p, written in operator form as follows:

$$p - \omega^2 \mathscr{G}[\psi p] = p_b, \tag{5}$$

where $\mathscr{G}$ is the integral operator, defined for any scalar field, ϕ,

$$\mathscr{G}[\phi] = \int_{\mathscr{D}} g(\boldsymbol{r}|\boldsymbol{r}')\phi(\boldsymbol{r}')d^3\boldsymbol{r}'. \tag{6}$$

The background Green's function can be decomposed of primary and secondary parts,

$$g = g^p + g^s, \tag{7}$$

which both satisfy the Helmholtz equation. The primary component, g^p, is the free-space Green's function. When the source and receiver are located in the same layer, g^p is nonzero,

$$g^p = \frac{e^{ik_b R}}{4\pi R}, \tag{8}$$

with $R = \sqrt{(x-x')^2 + (y-y')^2 + (z-z')^2}$

The secondary part, g^s, can be obtained for the horizontally stratified Earth by using the wavenumber integration technique:

$$g^s = \mathscr{H}_0[\tilde{g}^s], \tag{9}$$

where $\tilde{g}^s$ is the spectral Green's function, and $\mathscr{H}_0$ is the Hankel transform of zero order (Appendix A).

In two important special cases, function g^s can be expressed analytically. If the host model is homogeneous, then g^s is zero. If the host model consists of a half space with a perfectly reflecting boundary, and both the source and receiver are located within the half space, then the solution for g^s reduces to the following formula:

$$g^s = -\frac{e^{ik_b R_1}}{4\pi R_1}, \tag{10}$$

where $R_1 = \sqrt{(x-x')^2 + (y-y')^2 + (z+z')^2}$.

Let the anomalous volume, $\mathscr{D}$, be covered with $N = N_x N_y N_z$ cubical cells of size h^3. After discretization we receive the following matrix equation:

$$\boldsymbol{A}\boldsymbol{u} = \boldsymbol{b}, \tag{11}$$

where $\boldsymbol{u} = \{p_1..p_N\}$ is the vector of N unknown values approximating p in the center of each cell, $\boldsymbol{b} = \{p_{b,1}..p_{b,N}\}$ are known values of the background field in each cell, $\boldsymbol{A}$ is the scattering matrix,

$$\boldsymbol{A} = \left(\boldsymbol{I} - \omega^2 \boldsymbol{G}\boldsymbol{\Psi}\right), \tag{12}$$

where $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{\Psi} = \mathrm{diag}(\psi_1..\psi_N)$ is a diagonal matrix formed by contrasts for each cell, $\boldsymbol{G}$ contains integrals of Green's function over cells,

$$G_{ij} = \int_{\mathscr{D}_j} g(\boldsymbol{r}_i|\boldsymbol{r}')d^3\boldsymbol{r}', \quad i,j = 1..N, \quad \boldsymbol{r}_i, \boldsymbol{r}' \in \mathscr{D}, \tag{13}$$

where $\mathscr{D}_j$ is j-th cell. For the following, we will separate matrix $\boldsymbol{G}$ as $\boldsymbol{G} = \boldsymbol{G}^s + \boldsymbol{G}^p$ as we did in (7), where matrix $\boldsymbol{G}^s$ contains corresponding integrals of g^s, and $\boldsymbol{G}^p$ contains the integrals of g^p. To solve system (11) we used the (unpreconditioned) BiCGStab iterative solver (Saad, 2003).

2.2. The IE system matrix and its relation to the FD matrix

Matrix $\boldsymbol{A}$ is dense complex-valued non-hermitian and non-symmetric. Its condition number substantially improves at lower frequencies since in this case matrix spectrum has a weak dependence on the grid step size and velocity distribution. It is advisable to compare it with a FD matrix arising from equation (1). Let us consider the Helmholtz equation (1), in a volume that includes $\mathscr{D}$, completed with boundary conditions. We introduce the uniform grid with the cell size h. After the finite-difference discretization we receive a system of linear equations with system matrix, $\boldsymbol{A}^{FD}$,

$$\boldsymbol{A}^{FD} = \boldsymbol{L} - \omega^2 \boldsymbol{\Sigma}_t, \tag{14}$$

where $\boldsymbol{L}$ is the discrete version of negative Laplacian, and $\boldsymbol{\Sigma}_t$ is the diagonal matrix with discrete values of $1/c^2$ on its main diagonal.

The FD matrix will be much larger than $\boldsymbol{A}$, sparse, indefinite and possibly complex (e.g. due to boundary conditions). Its largest eigenvalue is near $O(h^{-2})$. Its least eigenvalue with respect to magnitude can be quite close to zero whenever the frequency is close to the resonance frequency. Due to these and some other reasons, the mentioned properties make the IE equation system more attractive for iterative solution than that of the FD method.

Let us consider the background Helmholtz problem (2) completed with the same boundary conditions. The corresponding FD system matrix, $\boldsymbol{B}$, is defined as follows:

$$\boldsymbol{B} = \boldsymbol{L} - \omega^2 \boldsymbol{\Sigma}_b, \tag{15}$$

where $\boldsymbol{\Sigma}_b$ is the diagonal matrix with discrete values of $1/c_b^2$ on its main diagonal. We introduce matrix $\boldsymbol{\Sigma}_a$ such that $\boldsymbol{\Sigma}_a = \boldsymbol{\Sigma}_t - \boldsymbol{\Sigma}_b$. Obviously,

$$\boldsymbol{A}^{FD} = \boldsymbol{B} - \omega^2 \boldsymbol{\Sigma}_a. \tag{16}$$

Multiplying both sides of this equation by $\boldsymbol{B}^{-1}$, we receive the following relation

$$\boldsymbol{B}^{-1}\boldsymbol{A}^{FD} = \boldsymbol{I} - \omega^2 \boldsymbol{B}^{-1}\boldsymbol{\Sigma}_a. \tag{17}$$

We should note that matrices $\boldsymbol{\Sigma}_a$ and $\boldsymbol{B}^{-1}$ are almost identical to matrices $\boldsymbol{\Psi}$ and $\boldsymbol{G}$ in equation (12), respectively, though they are not equal because of different discretizations. Thus, preconditioning of FD matrix $\boldsymbol{A}^{FD}$ with background media matrix $\boldsymbol{B}$ leads to a system with a structure and properties similar to that of the IE system. Of note, the act of $\boldsymbol{B}^{-1}$ can be computed with the method of separation of variables (Plessix and Mulder, 2003; Belonosov et al., 2017).

2.3. Matrix-vector multiplication

In this section we study several methods for multiplication $\boldsymbol{A}$ by a certain vector $\boldsymbol{v}$. Obviously, this task reduces to the computation of product $\boldsymbol{s} = \boldsymbol{G}\boldsymbol{v}$. In the following, we think about matrices/3D arrays as being split into quadrants/octants. The 2D/3D data are formed into a single vector using lexicographic numbering. We will use the following definitions: $x_m = h(m - N_x)$, $y_n = h(n - N_y)$, $z_l = h(l - N_z)$, where $m = 0..2N_x - 1$, $n = 0..2N_y - 1$, and $l = 0..2N_z - 1$.

2.3.1. Homogeneous space

In a homogeneous host medium $g^s \equiv 0$. Product $\boldsymbol{G}^p\boldsymbol{v}$ can be computed by the following formula:

$$\boldsymbol{G}^p\boldsymbol{v} = T_1\left(F^{-1}[F[\boldsymbol{S}]\, F[T_2(\boldsymbol{V})]]\right), \tag{18}$$

where $F[\cdot]$ and $F^{-1}[\cdot]$ are forward and inverse operators of 3D FFT of size $2N_x \times 2N_y \times 2N_z$, $\boldsymbol{V}$ is a three-dimensional array of size $N_x \times N_y \times N_z$ formed out of elements of $\boldsymbol{v}$. In (18) the element-wise multiplication is assumed between the two F operators. Three-dimensional $2N_x \times 2N_y \times 2N_z$ array $\boldsymbol{S}$ is defined by the following expression:

$$\boldsymbol{S}_{mnl} = \int_{x_m - h/2}^{x_m + h/2} \int_{y_n - h/2}^{y_n + h/2} \int_{z_l - h/2}^{z_l + h/2} g^p(0,0,0|x',y',z')dx'dy'dz'. \tag{19}$$

Operator T_1 takes a three-dimensional $2N_x \times 2N_y \times 2N_z$ array and rearranges its positive octant to a vector of size N. Operator T_2 takes a three-dimensional $N_x \times N_y \times N_z$ array, places it to the positive octant of a bigger $2N_x \times 2N_y \times 2N_z$ array, and fills other octants with zeros. Since g^p is symmetric, $\boldsymbol{S}$ can be constructed from the first row of matrix $\boldsymbol{G}^p$. These elements, except for the first one, is computed by quadratures. Element G^p_{11} is additionally discretized in order to approach the limit in the central subcell:

$$\lim_{V \to 0} \int_V g(\boldsymbol{r}|\boldsymbol{r}')d^3\boldsymbol{r}' = 0. \tag{20}$$

The matrix-vector multiplication, performed by (18), requires three 3D FFTs, though one of them (for $\boldsymbol{S}$) can be performed once. The numerical complexity of this method is $O(N\log N)$. The memory requirements is $O(N)$.

2.3.2. Layered host model

When the host medium is layered $g^s \neq 0$ and matrix $\boldsymbol{G}$ has a block structure,

$$\boldsymbol{G} = \begin{pmatrix} \boldsymbol{B}_{11} & \cdots & \boldsymbol{B}_{1N_z} \\ \vdots & \ddots & \vdots \\ \boldsymbol{B}_{N_z 1} & \cdots & \boldsymbol{B}_{N_z N_z} \end{pmatrix}, \tag{21}$$

where $\boldsymbol{B}_{mn}$ is a $N_x N_y \times N_x N_y$ submatrix, which corresponds to m-th source cell layer and n-th receiver cell layer. By $\boldsymbol{G}_l$ we denote l-th row of

blocks of matrix $\boldsymbol{G}$. By $\boldsymbol{s}_l$ we denote the part of vector $\boldsymbol{s}$ that corresponds to the product of $\boldsymbol{G}_l$ by $\boldsymbol{v}$, $\boldsymbol{s}_l = \boldsymbol{G}_l\boldsymbol{v}$. The effective algorithm for calculating $\boldsymbol{s}_l$ is the following:

$$\boldsymbol{s}_l = R_1\left(\Phi^{-1}\left[\sum_{q=1}^{N_z}\Phi\left[\boldsymbol{Q}_{lq}\right]\Phi\left[R_2\left(\boldsymbol{V}_q\right)\right]\right]\right), \tag{22}$$

where Φ and Φ^{-1} denote the forward and inverse FFTs of size $2N_x \times 2N_y$. The element-wise multiplication is assumed between the two Φ operators. Matrix $\boldsymbol{Q}_{lq}$ has size $2N_x \times 2N_y$. Its values are determined as follows:

$$\boldsymbol{Q}_{lq} = \int_{x_m-h/2}^{x_m+h/2}\int_{y_n-h/2}^{y_n+h/2}\int_{z_q-h/2}^{z_q+h/2} g(0,0,z_l|x',y',z')dx'dy'dz'. \tag{23}$$

Matrix $\boldsymbol{Q}_{lq}$ can be constructed from the elements of block $\boldsymbol{B}_{lq}$. In total (for all l) it is sufficient to compute only $N_xN_yN_z^2$ elements of matrix $\boldsymbol{G}$. Matrix $\boldsymbol{V}_q$ has dimensions $N_x \times N_y$. It is constructed by rearranging the part of vector $\boldsymbol{v}$ that corresponds to q-th cell layer. Operator R_2 transforms an $N_x \times N_y$ matrix into a $2N_x \times 2N_y$: it puts its argument into the positive quadrant of the resulting matrix and fills the other quadrants with zeros. Operator R_1 transforms $2N_x \times 2N_y$ matrix into a N_xN_y vector by taking the positive quadrant of its argument and rearranging it into the resulting vector. The full vector $\boldsymbol{s}$ is obtained by applying formula (22) for all $l = 1..N_z$. Matrices $\Phi\left[R_2\left(\boldsymbol{V}_q\right)\right]$, $q = 1..N_z$, can be computed ones for all l. The algorithm requires $N_z + N_z^2$ double FFTs of size $2N_x \times 2N_y$ and N_z inverse double FFTs of the same size, totaling to $N_z(2 + N_z)$ double FFTs. The overall numerical complexity of the matrix-vector multiplication algorithm is $O(N_xN_yN_z^2\log(N_xN_y))$. The memory consumption is $O(N)$. It is possible to save computation time by storing $N_xN_yN_z^2$ required values of $\boldsymbol{G}$ (Kruglyakov and Bloshanskaya, 2017), but it would rise the memory consumption to $O(N_xN_yN_z^2)$. The matrix-vector multiplication algorithm described above is exact up to round-off errors.

2.3.3. Reduction of numerical complexity of the matrix-vector multiplication for the layered background model

In this section we investigate the reduction of numerical complexity of the matrix-vector multiplication from $O(N_xN_yN_z^2\log(N_xN_y))$ to $O(N_xN_yN_z\log(N_xN_y))$. A similar problem was considered in the work (Avdeev and Knizhnik, 2009) as applied to electromagnetic diffusion modeling. It can be shown, that our approach is closely related to the well-known angular spectrum method (Leeman and Healey, 1997).

Term $\Phi[\boldsymbol{Q}_{lq}]$ in (22) is a rough approximation of the spectral Green's function. Its wavenumber resolution in, say, the x direction, is $\Delta k_x = \pi/(N_xh)$. This is inadequate, except for the near-field regime. The approximation error can be made arbitrary small by extending the double FFT beyond the body extent. We replace operator Φ, whose dimension is dictated by the body discretization, with a larger operator, $\tilde{\Phi}$. Dimensions of $\tilde{\Phi}$ is $2M \times 2M$, where M is a multiple of $\max(N_x, N_y)$. Letting M to be larger, we improve wavenumber resolution, $\pi/(Mh)$, through increase of memory usage. Operators R_1 and R_2 are replaced with wider operators $\tilde{R}_1$ and $\tilde{R}_2$. They transform a $N_x \times N_y$ submatrix of a $2M \times 2M$ matrix into a N_xN_y vector and vise versa.

By $\tilde{g}$, $\tilde{g}^p$, and $\tilde{g}^s$ we denote the spectral Green's function, its primary part, and its secondary part, respectively. Thus we have $\tilde{g} = \tilde{g}^p + \tilde{g}^s$. The expression for $\tilde{g}^s$ can be written as follows (Appendix A):

$$\tilde{g}^s = c_n e^{ik_z(z-z_{n-1})} + d_n e^{ik_z(z_n-z)}, \tag{24}$$

where $k_z = \sqrt{k^2 - k_x^2 - k_y^2}$, z_{n-1} and z_n are depths of the top and bottom boundaries of the host layer, c_n and d_n are some coefficients, which need to be determined for each value of k_z. Coefficients c_n and d_n do not depend on the depth of the receiver, and the exponentials do not depend on the depth of the source. Let us rewrite (24) for $z = z_q$, $z' = z_l$ as

$$\tilde{\boldsymbol{g}}^s\left(k_x, k_y, z_l, z_q\right) = \tilde{\boldsymbol{C}}_{lq} + \tilde{\boldsymbol{D}}_{lq}, \tag{25}$$

with

$$\begin{aligned}\tilde{\boldsymbol{C}}_{lq} &= \boldsymbol{c}_n\left(k_x, k_y, z_l\right)e^{i\left(z_q - z_{n-1}\right)\sqrt{k^2-k_x^2-k_y^2}},\\ \tilde{\boldsymbol{D}}_{lq} &= \boldsymbol{d}_n\left(k_x, k_y, z_l\right)e^{i\left(z_n - z_{q-1}\right)\sqrt{k^2-k_x^2-k_y^2}},\end{aligned} \tag{26}$$

$$k_x = \frac{n\pi}{Mh},\ m = -M..M-1,$$
$$k_y = \frac{m\pi}{Mh},\ n = -M..M-1.$$

Obviously,

$$\tilde{\boldsymbol{C}}_{lq} = \tilde{\boldsymbol{C}}_{l1}(\boldsymbol{P})^q, \quad \tilde{\boldsymbol{D}}_{lq} = \tilde{\boldsymbol{D}}_{l,N_z-1}(\boldsymbol{P})^{N_z-q-1}, \tag{27}$$

where

$$\boldsymbol{P} = e^{ih\sqrt{k^2-k_x^2-k_y^2}}, \tag{28}$$

with k_x and k_y take on the same discrete values, as in (26). We arrive at the following decomposition of matrix $\boldsymbol{G}$:

$$\boldsymbol{G} = \boldsymbol{G}^p + \boldsymbol{G}^c + \boldsymbol{G}^d, \tag{29}$$

where $\boldsymbol{G}^p$ corresponds to the primary part of the Green's function, $\boldsymbol{G}^c$ and $\boldsymbol{G}^d$ correspond to the two terms of (25), respectively. Let us consider matrix $\boldsymbol{G}^c$. It has a block structure:

$$\boldsymbol{G}^c = \begin{pmatrix}\boldsymbol{C}_{11} & \cdots & \boldsymbol{C}_{1N_z}\\ \vdots & \ddots & \vdots\\ \boldsymbol{C}_{N_z1} & \cdots & \boldsymbol{C}_{N_zN_z}\end{pmatrix}. \tag{30}$$

Using the above results, the product of l-th block row of matrix $\boldsymbol{G}^c$ by a vector $\boldsymbol{v}$, which we denote by $\boldsymbol{w}_l$, can be expressed as follows:

$$\boldsymbol{w}_l \approx h^3\tilde{R}_1\left(\tilde{\Phi}^{-1}\left[\sum_{q=1}^{N_z}\tilde{\boldsymbol{C}}_{lq}\tilde{\Phi}\left[\tilde{R}_2\left(\boldsymbol{V}_q\right)\right]\right]\right), \tag{31}$$

where h^3 accounts for volumetric integration. This equality holds approximately because $\boldsymbol{Q}_{lq} \approx \tilde{\Phi}^{-1}\left[\tilde{\boldsymbol{C}}_{lq}\right]h^3$. Using properties (27) we rewrite this expression as

$$\boldsymbol{w}_l \approx h^3\tilde{R}_1\left(\tilde{\Phi}^{-1}\left[\tilde{\boldsymbol{C}}_{l1}\boldsymbol{W}\right]\right), \tag{32}$$

with

$$\boldsymbol{W} = \tilde{\boldsymbol{V}}_1 + \boldsymbol{P}\left(\tilde{\boldsymbol{V}}_2 + \boldsymbol{P}\left(\tilde{\boldsymbol{V}}_3 + ..\right)\right), \tag{33}$$

where

$$\tilde{\boldsymbol{V}}_q = \tilde{\Phi}\left[\tilde{R}_2\left(\boldsymbol{V}_q\right)\right]. \tag{34}$$

Matrix $\boldsymbol{W}$ can be computed once for all l. Formula (32) requires one FFT per cell layer. Similar expressions can be derived for product $\boldsymbol{G}^d\boldsymbol{v}$. Product $\boldsymbol{G}^p\boldsymbol{v}$ is calculated as shown in Section 2.3.1. The overall asymptotic complexity of the described algorithm is $O(N_xN_yN_z\log(N_xN_y))$.

2.4. Parallelization

In this section we describe parallelization of our modeling code, based on the matrix-vector multiplication algorithm from Section 2.3.2. We used MPI for inter-node communication, whereas OpenMP was used for multi-core parallelization.

(I) The highest level of parallelism is the data decomposition across different compute nodes. The forward problem typically has to be

solved for many sources with the same model. Naturally, the available pool of MPI processes is divided into several groups. All processes in a group have their copies of model parameters and simultaneously work on a subset of sources, processing them one by one.

(II) The middle level of parallelism is the solution of a single system of linear equations on several compute nodes by BiCGStab. The BiCGStab solver requires four inner products and two matrix-vector products per iteration. Most of computation time is spent on computing products $\boldsymbol{Gv}$. In our implementation matrix $\boldsymbol{G}$ is divided into P strips,

$$\boldsymbol{G} = \begin{pmatrix} \boldsymbol{G}_1 \\ \vdots \\ \boldsymbol{G}_P \end{pmatrix}, \tag{35}$$

where P is the number of MPI processes in a group. Each MPI process computes product $\boldsymbol{G}_p\boldsymbol{v}$. The resulting vector is assembled on one MPI process and then sent to all other MPI processes to update their copies. This is performed by a call to *Allgather*. Some work in our code is not shared between MPI processes. For example, calculation of $\Phi[R_2(\boldsymbol{V}_q)]$ in (22) is performed on each node independently. The last issue can be improved by a fairly complex implementation (Kruglyakov and Bloshanskaya, 2017).

(III) The fine-grained parallelism with OpenMP includes parallel execution of the partial matrix-vector product $\boldsymbol{G}_p\boldsymbol{v}$. The matrix-vector multiplication algorithm, described in section 2.3.2, is parallelized in the following way. Expression $\widehat{\boldsymbol{V}}_q = \Phi[R_2(\boldsymbol{V}_q)], \quad q = 1..N_z$, is computed in parallel on OpenMP threads. This part of the code is not shared among several different MPI processes. Then the loop over $l \in$ *zlist* follows, where *zlist* is the list of cell layers assigned to a particular MPI process. Inside that loop, several OpenMP threads compute partial sums in form $\sum_{q=q_1}^{q_2} \Phi[\boldsymbol{Q}_{lq}]\ \widehat{\boldsymbol{V}}_q$ with subsequent summation of the partial sums. Finally, inverse FFTs (see (22)) are computed in parallel on several OpenMP threads.

3. Numerical study

3.1. Comparison and verification

We have verified our solution with SFAWEFD3D, which is a part of MADAGASCAR software package (http://www.ahay.org/). SFAWEFD3D solves the three-dimensional acoustic wave equation with the constant density in the time domain. We will refer this solution as the finite-difference time-domain (FDTD) solution.

We used the model consisting of a cubical body immersed into the half space. The body had dimensions of $1 \times 1 \times 1$ km^3. The center of the body was located at (0,0,1500) m. The body had constant velocity $c =$ 3000 m/s. The half space had velocity $c_b = 4000$ m/s and density $\rho_b =$ 2500 kg/m^3. There were 40 receivers spanning x from 100 m to 4000 m at a depth of 50 m with $y = 0$. The point source was located at (0,0,50) m.

For the IE solution, the anomalous domain was discretized into $64 \times 64 \times 64$ cubical cells with size $h = 15.525$ m. The problem was solved for 64 frequencies equally distributed from 0.25 Hz to 16 Hz. We use a twelve-core work station with Intel Xeon CPU E5-2620 v2 running at 2.1 GHz, equipped with 256 Gb RAM. The run times for different frequencies varied from 8 s at 0.25 Hz to 173 s at 16 Hz with the total time 62 min. These 64 solutions were Fourier-transformed into the time-domain.

For FDTD simulation, we created a $15 \times 11 \times 5$ km^3 domain (Fig. 1). We set the reflecting boundary condition on the top of the domain, and the hybrid one-way absorbing boundary conditions on the other sides. Such a large computational domain was selected to prevent strong remnant reflections from the bottom and sides before 2.5 s.

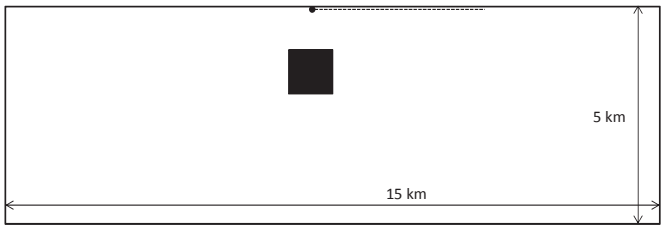

Fig. 1. Computational domain $15 \times 11 \times 5$ km^3 used for FDTD, the x-z plane is shown. The filled square is the $1 \times 1 \times 1$ km^3 body. The dotted line depicts location of receivers. The circle depicts the source position.

We employed the $O(\tau^2, h^4)$ scheme. The scheme was running with the time step $\tau = 0.4$ ms up to 4 s. The grid step size, h, was set to 10 m. The source wavelet was specified as the Ricker wavelet with the central frequency of 5 Hz, delayed by 240 ms. The total run time was 16.5 h. The agreement between the IE and FDTD two methods is perfect (Fig. 2).

We computed the relative misfit between traces from the two seismograms by the formula $\varepsilon = ||p_{IE} - p_{FDTD}||/||p_{IE}||$, where the subscript denotes the computational method, and $||\cdot||$ is the maximum norm. We added one another FDTD simulation with $h = 20$ m to check convergence of the FD solution. We observe that the relative error has doubled as the grid step size increased from 10 to 20 m (Fig. 3). The maximum relative misfit between the IE and FDTD traces computed with $h =$10 m is 8% on trace #9 at 900 m offset.

Trace # 9 is plotted in Fig. 4. Since the part of that trace before 0.6 s contains only the direct wave, we added the analytical solution for the half space as a reference. The IE solution is in excellent agreement with the exact solution at early times. The small amplitude mismatch in the FDTD solution at the first arrival, which is present at all receivers, may be caused either by the free surface or source term implementation. The visible phase shift in the FDTD solution after 0.65 s, which is present too at all receivers, is, likely, due to the fact that, the accuracy of a regular FD scheme drops to first order at a coefficient discontinuity. In general, the two methods show a decent match.

We applied our modeling algorithm to SEG/EAGE Overthrust model (Aminzadeh et al., 1997). The grid points of the original model within the ranges of 271–526, 271–526, and 1–128 cells in x,y, and z directions, respectively, were assigned to cells of the model shown in Fig. 5. The anomalous domain was immersed into the half space with $c_b = 2345$ m/s, and $\rho_b = 2500$ kg/m^3. Its top was placed at 500 m below the surface. The velocity in the model varied from 2345 m/s to 5500 m/s. The dimension of each model cell was $25 \times 25 \times 25$ m^3.

The source was located at (0,0,50) m. The pressure response was computed at frequencies of 2,5, and 8 Hz. We used 16 twelve-core nodes with Intel Xeon (Westmere X5660) processors running at 2.8 GHz and equipped with 24 Gb RAM and QDR Infiniband interconnect. During this simulation the code consumed 2.6 Gb memory per node. The BiCGStab solver performed 189, 1082, and 2639 double iterations, respectively (Fig. 6). The total run time was 47 min, 4.2 h, and 10.4 h, respectively. The anomalous pressure response at 8 Hz inside the anomalous domain is shown in Fig. 7.

3.2. Parallelization

In order to empirically assess the efficiency of parallelization, we used the same Overthrust model. The grid points of the original model within the ranges of 336–463, 336–463, and 1–128 in x,y, and z directions, respectively, were assigned to cells of the model, shown in Fig. 8. The other parameters were the same as in the previous example.

Parallelism level (I) should scale almost linearly in K, where K is the number of equally-sized MPI groups. In our tests, which involved a moderate number of compute nodes (hundreds), the scaling was nearly perfect. One of the typical results is presented in Fig. 9. The code had to

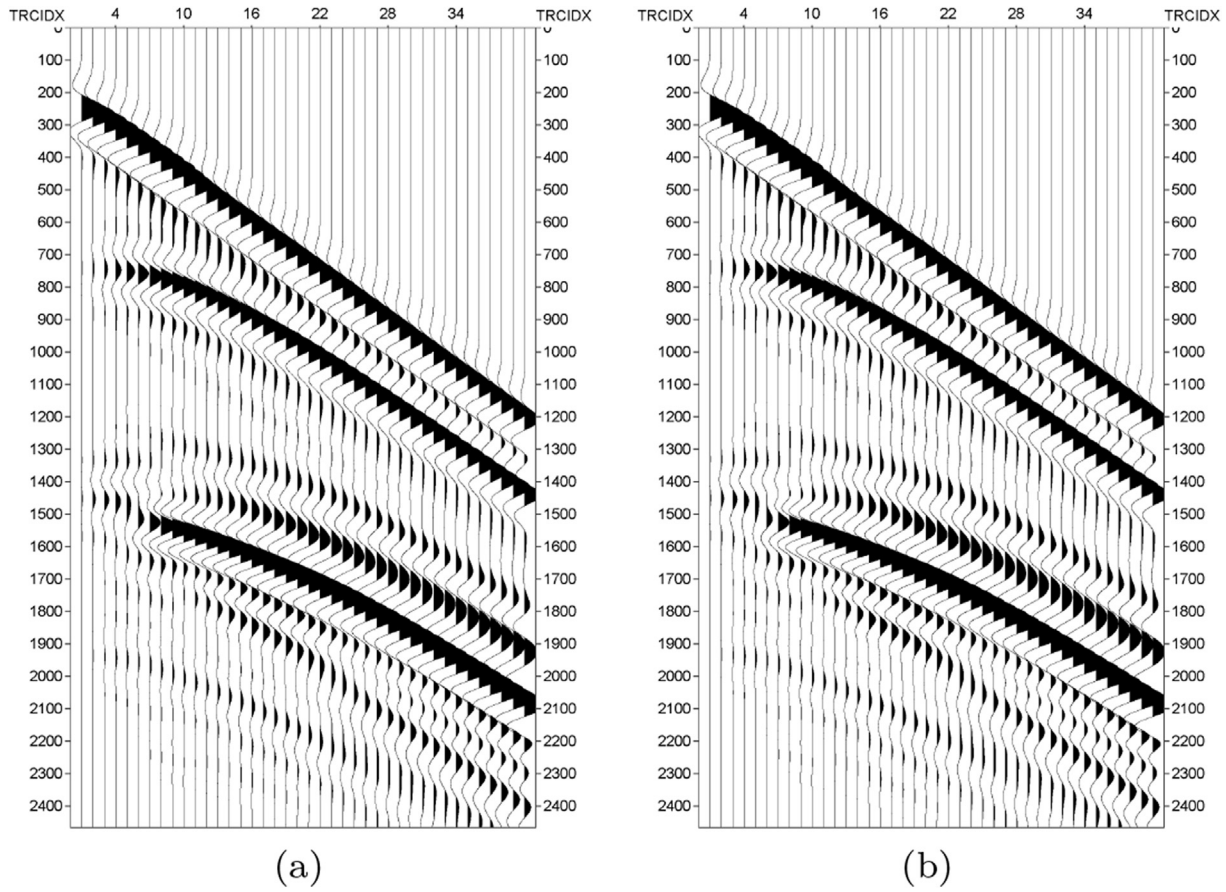

Fig. 2. Benchmark of IE against FDTD: (a) - traces converted from the spectrum computed by IE, (b) - FDTD seismogram, $h = 10$ m. Each trace was normalized by its maximum value independently in both panels. The time axis shows the arrival time in milliseconds.

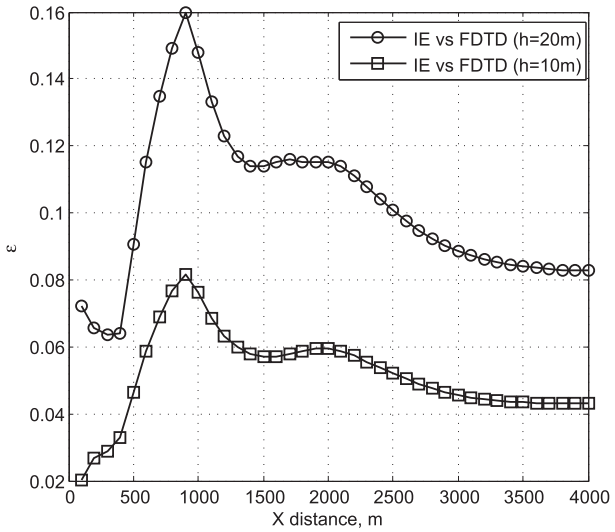

Fig. 3. Comparison of relative misfit between the IE and FDTD solutions in the maximum norm.

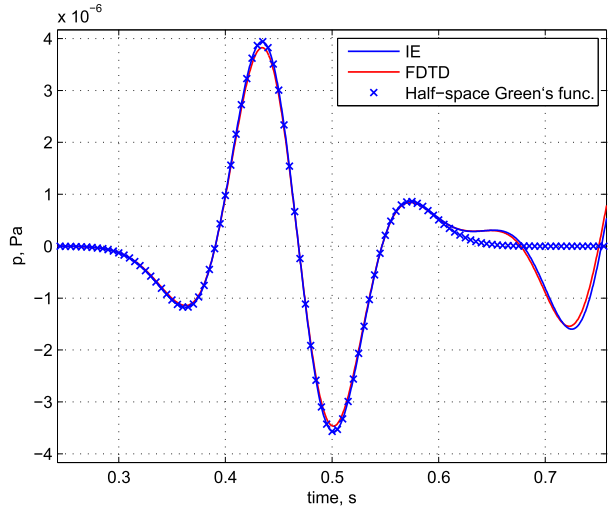

Fig. 4. Comparison of trace #9 (offset 900 m) from the shot records in Fig. 2. The curves are not normalized.

solve 1 to 128 identical forward problems on 1 to 128 compute nodes, one forward problem per node. Each node consisted of two four-core Intel Xeon X5570 processors running at 2.93 GHz and was equipped with 12 Gb RAM and QDR Infiniband interconnect. The resulting efficiency curves deviate from the maximum value 1.0 to less than 3% indicating a good speedup.

Parallelism level (II) represents iterative solution of the system of linear equations by P nodes simultaneously. In our experiments this level demonstrated limited scalability. One of the numerical experiments is presented in Fig. 10. In this test we used the same setup, as before. This test was run on a cluster consisted of twelve-core nodes with Intel Xeon (Westmere X5660) processors running at 2.8 GHz and equipped with 24 Gb RAM and QDR Infiniband interconnect. Only one core per node was working. In each individual runs all nodes were running the parallel iterative solver. The iterative solver was forced to terminate at iteration

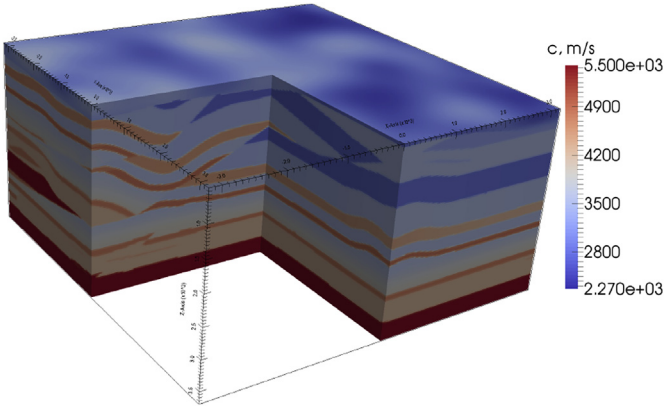

Fig. 5. Velocity model.

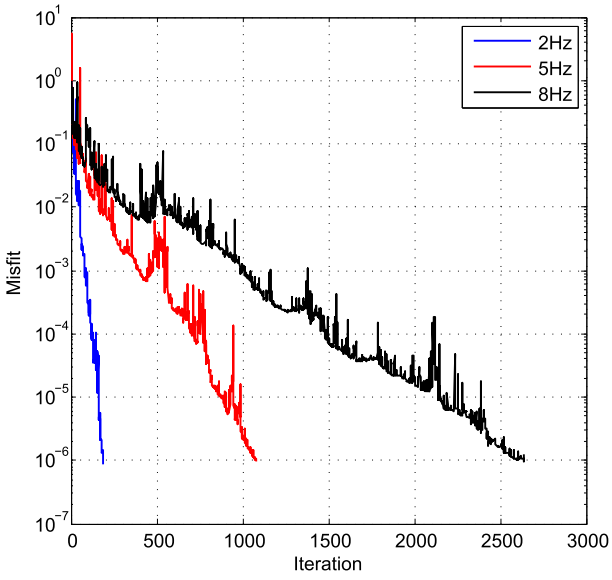

Fig. 6. Convergence of BiCGStab solver for different frequencies.

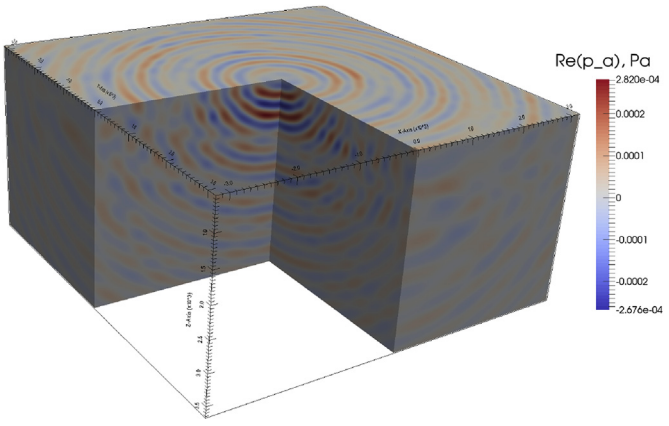

Fig. 7. Real part of the anomalous pressure field, p_a, at 8 Hz.

71, when the solution misfit was slightly above 10^{-6}, to ensure the same number of iterations for all runs.

The speed-up curves start to deteriorate at 8 nodes, where effectiveness is 84% and 85% for the two curves shown. Effectiveness decreases to

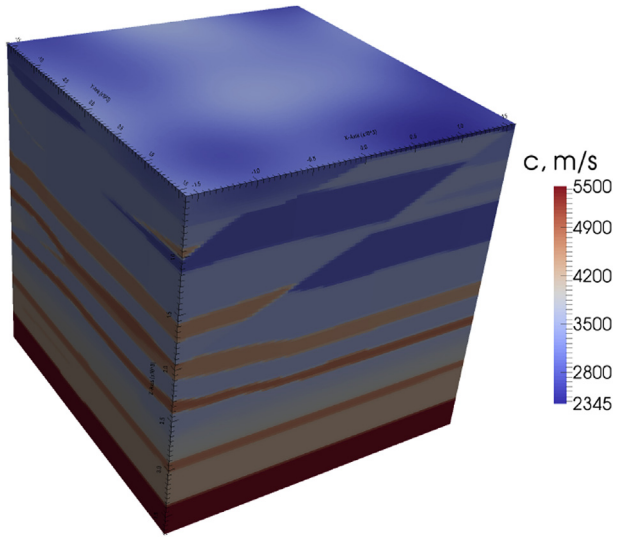

Fig. 8. The model used to test the efficiency of parallelization.

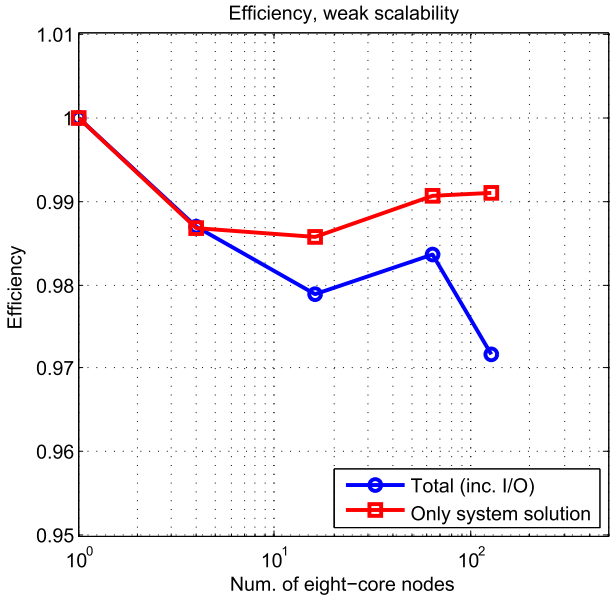

Fig. 9. Scalability of the source distribution across different compute nodes (weak scalability).

55% and 61%, respectively, at 64 nodes. We attribute such limited scaling to duplicated computations and a large number of collectives. This level of parallelism should be used with care and other levels should be preferred whenever possible.

Parallelism level (III) includes the parallel execution of matrix-vector multiplication on several cores within the same node. The algorithm, outlined in section 2.4, has a sequential segment within the critical section. This segment is responsible for decreasing the effectiveness as the number cores grows. A typical result of the parallelization test is shown in Fig. 11. In this test we used a 20-core shared-memory system with two Intel Xeon Ivybridge E5-2670 processors running at 2.5 GHz and equipped with 64 GB memory. The forward problem was solved with 1–20 cores in parallel. The iterative solver was forced to terminate at iteration 69, when the solution misfit was slightly above 10^{-6}. The CPU affinity was set in such a way to rigidly bind OpenMP threads to physical cores. In this test we used the same model, described above, without 8 most lower cell layers (thus, the model was of 128×128×120 cells). Such a modification of the model ensures that the number of cell layers per

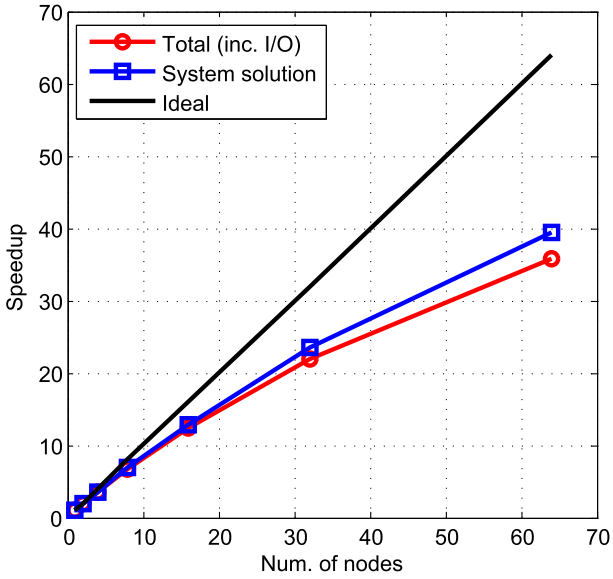

Fig. 10. Scalability of the parallel system solution.

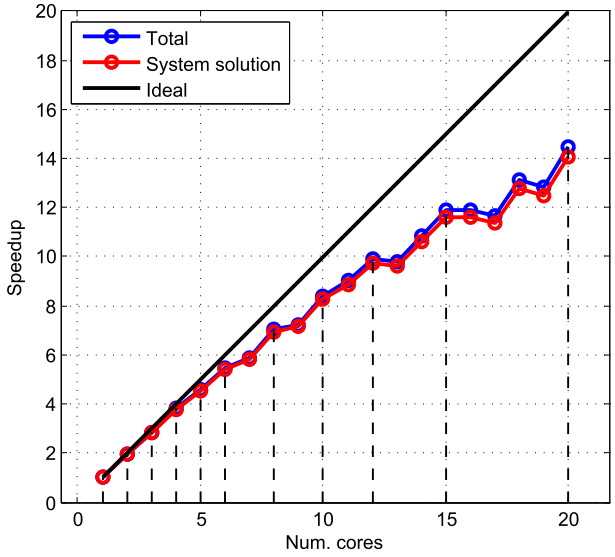

Fig. 11. Scalability test of OpenMP parallelization. Vertical dashed lines depict numbers of cores that evenly divide the number of cell layers (i.e. 120).

core is integer for 1,2,3,4,5,6,8,10,12,15, and 20 cores.

The resulting speedup curves revealed fairly good scaling versus the number of cores. Efficiency is above 70% at 20 nodes. Efficiency rises when the number of cores is evenly divisible by number of cell layers, which is indicated by local peaks. Note, that vector-vector products inside BiCGStab algorithm was running on the master thread without parallelization.

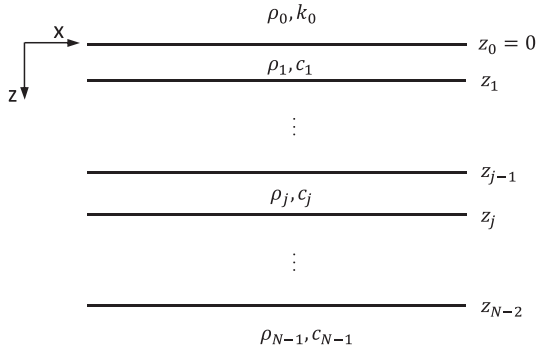

Fig. 12. The layered model.

4. Conclusions

We have developed an effective numerical method and a parallel algorithm for 3D frequency-domain modeling of the acoustic field. The system of integral equations is solved iteratively; the matrix-vector product is computed via FFT. This makes the computational complexity and memory requirements tolerable for application to realistically large problems. We extended this approach by parallelizing the code at three levels: distribution of seismic sources across different groups of MPI processes, solution of system of linear equation with several MPI processes, and parallelization of the matrix-vector multiplication over OpenMP threads.

We studied the efficiency of all three levels of parallelization. The parallelization of the system solution has been found the most difficult part, because the system matrix is dense. This limits the scaling of this level. Presumably, any algorithms, based on volume integral equations, would have similar problems. On the other hand, the coarse-grained parallelism (the distribution of sources) and fine-grained parallelism (matrix-vector multiplication) are effectively parallelized.

The results of our paper demonstrate that the IE method may be an invaluable tool for accurate acoustic modeling of seismic field in the medium with sharp material property discontinuities, and with the infinitely extended domains. The parallelization of the developed algorithm, as presented above, allows the user to simulate realistically large seismic models. If the anomalous domain is relatively small, then the IE method can outperform the standard finite-difference schemes in terms of computation time. Future research will be aimed at application of the developed method to a solution of the full elastic wave problem.

Acknowledgments

This research was supported by the Russian Science Foundation, project No. 16-11-10188. The authors also acknowledge support from the University of Utah's Consortium for Electromagnetic Modeling and Inversion (CEMI) and TechnoImaging. We are thankful to the Supercomputing Center of Lomonosov Moscow State University for providing computational facility. We have also used computing resources of the federal collective usage center Complex for Simulation and Data Processing for Mega-science Facilities at NRC “Kurchatov Institute”.

Appendix A. Acoustic Green's function in a layered medium

The task of computing wave fields in a horizontally stratified elastic medium has been analyzed in detail in (Schmidt and Jensen, 1985). We give a less general, but more simple and fast solution for the acoustic case. Our exposition is somewhat similar to (Zhdanov, 2009) and (Brekhovskikh, 1980), though the first reference deals with the electromagnetic Green's tensor, and the second one considers a source located above a stack of elastic layers.

The earth is assumed to be a horizontally stratified stack of N layers. The layers are numbered from 0, to $N-1$ (Fig. 12).

Each j-th layer is characterized by its velocity of P-waves, c_j, density, ρ_j, and the depth of its upper and lower boundary, z_{j-1} and z_j, respectively. The layers $j = 0, N-1$ are the half spaces. In each layer, the Green's function satisfies to the following Helmholtz equation:

$$-\Delta u - k_j^2 u = \delta(r - r'), \tag{A.1}$$

where $k_j = \omega/c_j$ is the wavenumber, δ is the Dirac delta-function, r is the position vector of the receiver, and r' is the position vector of the source. This equation is completed with the boundary conditions (or the radiation conditions at infinity). We will seek the solution in form

$$u = \mathscr{F}^{-1}[U], \tag{A.2}$$

where $\mathscr{F}$ and $\mathscr{F}^{-1}$ are the pair of double Fourier transform over the XY plane, and $U = \mathscr{F}[u]$. We call U the spectral Green's function. By k_x, k_y, and k_z we will denote the components of the wave vector $\boldsymbol{k}$, such as $\boldsymbol{k} = (k_x, k_y, k_z)$. We will denote by μ z-component of $\boldsymbol{k}$:

$$\mu \equiv k_z = \sqrt{k^2 - k_x^2 - k_y^2}. \tag{A.3}$$

Let the point source be located in layer m, and the point receiver be located at the layer n. The spectral Green's function, U, satisfies to the following set of N equations:

$$U_j'' + \mu_j^2 U_j = \begin{cases} \delta(z - z'), & j = m \\ 0, & otherwise. \end{cases} \tag{A.4}$$

Equation (A.4) are supplemented with $2(N-1)$ conditions on layers interfaces plus the radiation conditions as follows:

$$[U]_j = 0, \quad j = 0..N-2,$$

$$\left[\frac{\rho}{\mu}\frac{U}{U'}\right]_j = 0, \quad j = 0..N-2, \tag{A.5}$$

$$U_\infty = 0, \quad U_{-\infty} = 0.$$

Solutions of equation (A.4) are given in the following form:

$$U_j(z) = c_j f_j(z) + d_j g_j(z), \tag{A.6}$$

$$f_j(z) = e^{i\mu_j\left(z - z_{j-1}\right)}, \quad j = 0..N-1, \tag{A.7}$$

$$g_j(z) = e^{i\mu_j\left(z_j - z\right)}, \quad j = 0..N-1.$$

From (A.5) it follows that,

$$c_0 \equiv 0, \quad f_0 \equiv 0,$$

$$d_{N-1} \equiv 0, \quad g_{N-1} \equiv 0. \tag{A.8}$$

Let us consider an auxiliary variable R_j,

$$R_j(z) = \frac{U_j}{U'_j} = \frac{c_j f_j(z) + d_j g_n(z)}{c_j f_j(z) - d_j g_j(z)}, \quad z_{j-1} \le z \le z_j. \tag{A.9}$$

It follows from (A.8) that:

$$R_0(z) \equiv -1, \qquad z \ge z_0,$$

$$R_{N-1}(z) \equiv 1, \quad z \le z_{N-2}. \tag{A.10}$$

Starting from these two values, R_j at the top and bottom of the source layer m are computed recursively. We rewrite the second set of conditions (A.5) as follows:

$$\xi_j R_j\left(z_j\right) = \xi_{j+1} R_{j+1}\left(z_j\right), \tag{A.11}$$

where $\xi_j = \rho_j/\mu_j$. Using identity

$$\frac{Ce^{ix} + De^{-ix}}{Ce^{ix} - De^{-ix}} = coth\left(ix + ln(C/D)^{1/2}\right), \tag{A.12}$$

the following recursion relations can be obtained

$$R_j(z_j) = coth\left(i\mu_j(z_j - z_{j-1}) + acoth\left(\frac{\xi_{j-1}}{\xi_j} R_{j-1}(z_{j-1})\right)\right), \quad j = 1..m-1,$$
$$R_j(z_{j-1}) = coth\left(-i\mu_j(z_j - z_{j-1}) + acoth\left(\frac{\xi_{j+1}}{\xi_j} R_{j+1}(z_j)\right)\right), \quad j = N-2..m+1, \tag{A.13}$$

where *coth* and *acoth* are the hyperbolic cotangent and inverse hyperbolic cotangent, respectively. Note that, the free-space and double half-space cases are threated separately.

Since values $R_{m-1}(z_{m-1})$ and $R_{m+1}(z_m)$ are known, we can obtain the coefficients, c_m and d_m, in the source layer m. Within the source layer, the solution is the sum of the general solution of the homogeneous differential equation, U_m, and a particular solution of its nonhomogeneous version. The particular solution, u^0, is the free-space Green's function, which has the following spectral representation:

$$U^0(z) = \frac{i}{2\pi}\frac{1}{\mu_m}\begin{cases} e^{i\mu_m(z'-z)}, & z < z' \\ e^{i\mu_m(z-z')}, & z > z'. \end{cases} \tag{A.14}$$

Its z-derivative, $Q(z) = dU^0/dz$, is expressed as follows:

$$Q(z) = \frac{1}{2\pi}\begin{cases} e^{i\mu_m(z'-z)}, & z < z' \\ e^{i\mu_m(z-z')}, & z > z'. \end{cases} \tag{A.15}$$

Continuity of the solution and discontinuity of its z-derivative must be satisfied at the source layer interfaces:

$$\xi_{m-1}R_{m-1}(z_{m-1}) = \xi_j\frac{c_m f_m(z_{m-1}) + d_m g_m(z_{m-1}) + U^0(z_{m-1})}{c_m f_m(z_{m-1}) - d_m g_m(z_{m-1}) + Q(z_{m-1})}, \tag{A.16}$$

and

$$\xi_{m+1}R_{m+1}(z_{m+1}) = \xi_m\frac{c_m f_m(z_m) + d_m g_m(z_m) + U^0(z_m)}{c_m f_m(z_m) - d_m g_m(z_m) + Q(z_m)}. \tag{A.17}$$

These equations can be combined into the following 2×2 system of linear equations with respect to c_m and d_m:

$$c_m f_m(z_{m-1})(\xi_m - T\xi_{m-1}) + d_m g_m(z_{m-1})(\xi + T\xi_{m-1}) = T\xi_{m-1}Q(z_{m-1}) - \xi_m U^0(z_{m-1}) \tag{A.18}$$

$$c_m f_m(z_m)(\xi_n - B\xi_{m+1}) + d_m g_m(z_m)(\xi_m + B\xi_{m+1}) = B\xi_{m+1}Q(z_m) - \xi_m U^0(z_m) \tag{A.19}$$

where $T = R_{m-1}(z_{m-1})$, $B = R_{m+1}(z_m)$. Note that, $f_m(z_{m-1}) = 1$, $g_m(z_m) = 1$. The cases $m = 0, 1, N-2$, and $N-1$ are treated separately.

Coefficients c_j,d_j in all other layers are calculated recursively. For $j < m$ we impose continuity of the solution at j interface:

$$c_j f_j(z_j) + d_j g_j(z_j) = U_{j+1}(z_j). \tag{A.20}$$

We use the following identity at interface j:

$$\xi_j R_j(z_j) = \xi_j\frac{c_j f_j(z_j) + d_j g_j(z_j)}{c_j f_j(z_j) - d_j g_j(z_j)}. \tag{A.21}$$

Values of $R_j(z_j)$ are computed at the previous steps. We have the following relation:

$$d_j = \frac{U_{j+1}(z_j)\left(R_j(z_j) - 1\right)}{2R_j(z_j)}. \tag{A.22}$$

In order to determine c_j we use the discontinuity condition on R at $j-1$ interface:

$$\xi_{j-1}R_{j-1}(z_{j-1}) = \xi_j\frac{c_j f_j(z_{j-1}) + d_j g_j(z_{j-1})}{c_j f_j(z_{j-1}) - d_j g_j(z_{j-1})}. \tag{A.23}$$

We have the following relation:

$$c_j = \frac{R_{j-1}(z_{j-1})\xi_{j-1} + \xi_j}{R_{j-1}(z_{j-1})\xi_{j-1} - \xi_j} d_j g_j(z_{j-1}). \tag{A.24}$$

We use different equations for c_j and d_j ((A.23) and (A.21), respectively) to avoid growing exponentials in the solution even for attenuating media (that is $Im(k) > 0$, by convention).

For the layers located below the source layer ($j > m$) we impose continuity of the solution at z_{j-1} interface:

$$c_j f_j(z_{j-1}) + d_j(z_{j-1}) = U_{j-1}(z_{j-1}). \tag{A.25}$$

To determine c_j, we use the following identity at $j-1$ interface:

$$\xi_j \frac{c_j f_j(z_{j-1}) + d_j g_j(z_{j-1})}{c_j f_j(z_{j-1}) - d_j g_j(z_{j-1})} = U_{j-1}(z_{j-1}). \tag{A.26}$$

Expression for c_j is as follows:

$$c_j = \frac{U_{j-1}(z_{j-1})\left(R_j(z_{j-1}) + 1\right)}{2R_j(z_{j-1})}. \tag{A.27}$$

To determine d_j, we use the discontinuity of R at j-th interface:

$$\xi_{j+1} R_{j+1}(z_j) = \xi_j \frac{c_j f_j(z_j) + d_j g_j(z_j)}{c_j f_j(z_j) - d_j g_j(z_j)}. \tag{A.28}$$

Expression for d_j is as follows:

$$d_j = \frac{\xi_{j+1} R_{j+1}(z_j) - \xi_j}{\xi_{j+1} R_{j+1}(z_j) + \xi_j} c_j f_j(z_j). \tag{A.29}$$

The algorithm can be summarize as follows:

1. Recursively compute $R_j(z_j)$ for $j = 0..m-1$ and $R_j(z_{j-1})$ for $j = N-1..m+1$ using (A.13).
2. Compute U^0 and Q at the top and bottom boundaries of the source layer using (A.14) and (A.15).
3. Solve system (A.18)-(A.19) for c_m and d_m.
4. Recursively compute c_n and d_n using (A.22), (A.24) ($n<m$) or (A.27), (A.29) ($n>m$).

Care must be taken in evaluating *acoth* close to its brunch points, and also for *coth* for large arguments.
In order to solve the original problem (A.1), we must perform the inverse Fourier transform (A.2), or, equivalently, the Hankel transform as follows:

$$u(r, z, z') = \mathscr{H}_0[U_n(z, z')] = \frac{1}{2\pi} \int_0^\infty U_n(\alpha, z, z') J_0(\alpha r) \alpha d\alpha \tag{A.30}$$

where J_0 is the Bessel function of the first kind of zero order, $\alpha = \sqrt{k_x^2 + k_y^2}$, $r = \sqrt{(x-x')^2 + (y-y')^2}$. The digital filtering approach, which is routinely used in geophysical electromagnetic simulation, does not work for (A.30). The difficulties arise from the fact that the integrand oscillates, has the branch point at $\alpha = k_n$, has poles that correspond the horizontal waves in the layers, and, generally, decays very slowly with r. Despite a vast body of literature on the matter integration of the spectral Green's function remains difficult. From our experience, the most reliable result is achieved by the adaptive integration along a contour in the lower half plane with the additional adaptive integration of the integrand tail along the real axis.

References

Abubakar, A., Habashy, T., 2013. Three-dimensional visco-acoustic modeling using a renormalized integral equation iterative solver. J. Comput. Phys. 249, 1–12.

Abubakar, A., van den Berg, P.M., 2004. Iterative forward and inverse algorithms based on domain integral equations for three-dimensional electric and magnetic objects. J. Comput. Phys. 195 (1), 236–262.

Abubakar, A., Van Den Berg, P., Fokkema, J., 2003. Towards non-linear inversion for characterization of time-lapse phenomena through numerical modelling. Geophys. Prospect. 51 (4), 285–293.

Aki, K., Richards, P.G., 1980. Quantitative Seismology. W. R. Freeman and Co.

Alles, E., van Dongen, K., 2009. Frequency domain perfectly matched layers for acoustic scattering integral equation problems. In: Ultrasonics Symposium (IUS), 2009 IEEE International.

Aminzadeh, F., Brac, J., Kunz, T., 1997. 3-D Salt and Overthrust Models. Volume 1 of SEG/EAGE 3-D Modeling Series. SEG, Tulsa, pp. 247–256.

Avdeev, D., Knizhnik, S., 2009. 3D integral equation modeling with a linear dependence on dimensions. Geophysics 74 (5), F89–F94.

Avdeev, D.B., Kuvshinov, A.V., Pankratov, O.V., Newman, G.A., 2002. Three-dimensional induction logging problems, Part I: an integral equation solution and model comparisons. Geophysics 67 (2), 413–426.

Belonosov, M., Dmitriev, M., Kostin, V., Neklyudov, D., Tcheverda, V., 2017. An iterative solver for the 3d helmholtz equation. J. Comput. Phys. 345, 330–344.

Brekhovskikh, L.M., 1980. Waves in Layered Media, second ed. Academic Press.

Carcione, J.M., Herman, G.C., ten Kroode, A.P.E., 2002. Seismic modeling. Geophysics 67 (4), 1304–1325.

Čuma, M., Gribenko, A., Zhdanov, M.S., 2017. Inversion of magnetotelluric data using integral equation approach with variable sensitivity domain: application to earthscope mt data. Phys. Earth Planet. Interiors 270, 113–127.

de Hoop, A.T., 1958. Representation Theorems for the Displacement in an Elastic Solid and Their Application to Elastodynamic Diffraction Theory. Technische Hogeschool, Delft. PhD thesis.

Freter, H., 1992. An integral equation method for seismic modelling and inversion. In: Vogel, A., Sarwar, A., Gorenflo, R., Kounchev, O. (Eds.), Theory and Practice of Geophysical Data Inversion, Volume 5 of Theory and Practice of Applied Geophysics. Vieweg+Teubner Verlag, pp. 243–264.

Fu, L.-Y., 2003. Numerical study of generalized Lippmann-Schwinger integral equation including surface topography. Geophysics 68 (2), 665–671.

Fu, L.-Y., Mu, Y.-G., Yang, H.-J., 1997. Forward problem of nonlinear Fredholm integral equation in reference medium via velocity-weighted wavefield function. Geophysics 62 (2), 650–656.

Hursán, G., Zhdanov, M.S., 2002. Contraction integral equation method in three-dimensional electromagnetic modeling. Radio Sci. 37 (6), 1–1–1–13. 1089.

Jakobsen, M., Ursin, B., 2015. Full waveform inversion in the frequency domain using direct iterative T-matrix methods. J. Geophys. Eng. 12 (3), 400–418.

Jakobsen, M., Wu, R., 2016. Renormalized scattering series for frequency-domain waveform modelling of strong velocity contrasts. Geophys. J. Int. 206, 880–899.

Johnson, S., Zhou, Y., Berggren, M., Tracy, M., 1983. Acoustic inverse scattering solutions by moment methods and back propagation. In: Theory and Applications. SIAM, pp. 144–155.

Kruglyakov, M., Bloshanskaya, L., 2017. High-performance parallel solver for integral equations of electromagnetics based on Galerkin method. Math. Geosci. 49 (6), 751–776.

Lahaye, D., Tang, J., Vuik, K., 2017. Modern Solvers for Helmholtz Problems. Geosystems Mathematics. Springer International Publishing.

Leeman, S., Healey, a. J., 1997. Field Propagation via the Angular Spectrum Method. Springer US, Boston, MA, pp. 363–368.

Malovichko, M., Khokhlov, N., Yavich, N., Zhdanov, M., 2017. Approximate solutions of acoustic 3D integral equation and their application to seismic modeling and full-waveform inversion. J. Comput. Phys. 346, 318–339.

Morse, P.M., Feshbach, H., 1953. Methods in Theoretical Physics. McGraw-Hill Book Company, inc.

Operto, S., Virieux, J., Amestoy, P., LExcellent, J.-Y., Giraud, L., Ali, H.B.H., 2007. 3d finite-difference frequency-domain modeling of visco-acoustic wave propagation using a massively parallel direct solver: a feasibility study. Geophysics 72 (5), SM195–SM211.

Osnabrugge, G., Leedumrongwatthanakun, S., Vellekoop, I.M., 2016. A convergent Born series for solving the inhomogeneous helmholtz equation in arbitrarily large media. J. Comput. Phys. 322, 113–124.

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

Plessix, R.E., Mulder, W.A., 2003. Separation-of-variables as a preconditioner for an iterative Helmholtz solver. Appl. Numer. Math. 44, 385–400.
Saad, Y., 2003. Iterative Methods for Sparse Linear Systems. Society for Industrial and Applied Mathematics, second ed. Philadelphia, PA, USA.
Schmidt, H., Jensen, F.B., 1985. A full wave solution for propagation in multilayered viscoelastic media with application to Gaussian beam reflection at fluidsolid interfaces. J. Acoust. Soc. Am. 77 (3), 813–825.
Virieux, J., Operto, S., 2009. An overview of full-waveform inversion in exploration geophysics. Geophysics 74 (6).
Wu, R.-S., Toksoz, M.N., 1987. Diffraction tomography and multisource holography applied to seismic imaging. Geophysics 52 (1), 11–25.
Yao, J., Lesage, A.-C., Hussain, F., Kouri, D.J., 2016. Forward scattering and Volterra renormalization for acoustic wavefield propagation in vertically varying media. Commun. Comput. Phys. 20 (2), 353–373.
Yavich, N., Zhdanov, M.S., 2016. Contraction pre-conditioner in finite-difference electromagnetic modelling. Geophys. J. Int. 206 (3), 1718–1729.
Zhang, R., Ulrych, T.J., 2000. Seismic Forward Modeling by Integral Equation and Some Practical Considerations (chapter 593). SEG, pp. 2329–2332.
Zhdanov, M.S., 2002. Geophysical Inverse Theory and Regularization Problems. Elsevier.
Zhdanov, M.S., 2009. Geophysical Electromagnetic Theory and Methods. Elsevier.
Zhdanov, M.S., 2015. Inverse Theory and Applications in Geophysics, second ed. Elsevier.
Zhdanov, M.S., Fang, S., 1997. Quasi-linear series in three-dimensional electromagnetic modeling. Radio Sci. 32 (6), 2167–2188.