Journal of Applied Geophysics 161 (2019) 204–215

Contents lists available at ScienceDirect

Journal of Applied Geophysics

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

A highly efficient implicit finite difference scheme for acoustic wave propagation

Ajay Malkoti [a,*], Nimisha Vedanti [a,b], Ram Krishna Tiwari [a,b]

[a] AcSIR-NGRI, Hyderabad 500007, India
[b] CSIR-National Geophysical Research Institute, Hyderabad 500007, India

ARTICLE INFO

Article history:
Received 20 April 2018
Received in revised form 9 October 2018
Accepted 22 December 2018
Available online 29 December 2018

Keywords:
Seismic
Acoustic
Wave propagation
Finite difference
2D/3D modeling

ABSTRACT

The accuracy of a numerical derivative has a significant effect on any numerical simulation. Long stencils can provide high accuracy as well as reduce the numerical anisotropy error. However, such a long stencil demands extensive computational resources and with their growing size, such derivatives may become physically non-realistic since contributions from very far offset whereas the derivative is local in nature. Further, the application of such long stencils at boundary points may introduce errors. In this paper, we present a very efficient, accurate and compact size numerical scheme for acoustic wave propagation using implicit finite difference operator, which utilizes a lesser number of points to estimate derivatives in comparison to the conventional central difference derivative operator. The implicit derivative operator, despite its several advantages, is generally avoided due to its high computational cost. Therefore in this paper, we discuss a method which can dramatically reduce the computational cost of this scheme to almost half. This strategy is useful particularly for 2D and 3D case. Spectral characteristics of the derivative operator and the numerical scheme are compared with several other central difference schemes. We have also demonstrated an application of this scheme for seismic wave propagation in 2D and 3D acoustic media.

© 2019 Elsevier B.V. All rights reserved.

1. Introduction

Numerical seismic simulations have much importance in seismic workflows, and among its numerous applications, it includes algorithms, such as reverse time migration and full waveform inversion. It is always desired to have an accurate, stable and computationally efficient numerical scheme which can be used to solve governing (and constitutive) relations. This scheme involves both time and space derivatives, thus its accuracy chiefly depends on following three factors: (1) the step size chosen for space and time discretization (i.e. Δx,Δt), (2) number of neighborhood points used for estimating derivative, and (3) methods used for estimating spatial derivatives and to carry out time integration. Each factor affects the simulation cost in terms of memory, number of operations and computational time. For example, when a smaller size is employed for space and time discretization, the results would be more accurate but at the expense of increased computational cost due to the increased number of nodes. Increasing the derivative stencil size/length much can violate the local nature as it have a contribution from the nodes at far offset, which may make this derivative estimations non-local in nature. Also after a certain number of nodes, further increment in derivative stencil length will increase only formal accuracy (defined in terms of truncation error) whereas the spectral accuracy gets saturated (Kosloff et al., 2010). Thus a derivative stencil longer than saturation length, would only increase the computational cost but accuracy. Considering the above factors, it is desirable to carry out a numerical simulation using a scheme that can produce highly accurate numerical derivatives using only a small number of nodes for a given grid spacing without adding a significant computational cost.

As mentioned before, the accuracy of a numerical scheme is directly linked with the accuracy of the numerical derivative operator. Several types of numerical derivative operators, which have been utilized for wave propagation can be found in the literature. For example, the classical central derivative (CD) operator, which is based upon Taylor series expansion, can be used for simulating the seismic wavefield on the collocated grid (Alterman and Karal, 1968; Kelly et al., 1976, etc.). The CD operator over the staggered grid provides decoupling of even-odd modes as well as higher formal accuracy than conventional operators (Harlow and Welch, 1965). It has also been widely used in the geophysics community (Madariaga, 1976; Virieux, 1986; Levander, 1988, etc.). These derivative operators can be made to yield spectrally more accurate derivative estimates with proper choice of coefficients. These coefficients can be determined by matching the spectral characteristics of the numerical derivative to the exact in the wavenumber

* Corresponding author.
E-mail addresses: ajmalkoti@gmail.com (A. Malkoti), nimisha@ngri.res.in (N. Vedanti).

https://doi.org/10.1016/j.jappgeo.2018.12.017
0926-9851/© 2019 Elsevier B.V. All rights reserved.

domain (Holberg, 1987; Tam and Webb, 1993). Following this approach, many researchers have also proposed *N*-points Laplacian stencils (Jo et al., 1996; Shin and Sohn, 1998) as well as their optimized versions (Gosselin-Cliche and Giroux, 2014; Liu et al., 2015; Wang et al., 2016; Fan et al., 2017) for acoustic wave propagation in earth. With the increasing accuracy of FD operator, the error in phase velocity due to numerical anisotropy also reduces. Some more reduction (2–3%) in this error can be achieved with the help of the rotated staggered-grid scheme (Saenger and Bohlen, 2004; Virieux et al., 2011), however, it comes at the cost of the extra cross derivative term (Štekl and Pratt, 1998).

Another type of operator, known as implicit FD operator, is based on Padé schemes (Collatz, 1960). This operator utilizes the value of the function as well as the value of the derivative, which are provided at the given point with some of its neighborhood points for estimating derivative of a function at a given node. An improved version of the implicit FD operator was presented by Lele (1992) which provides better resolution at higher wavenumbers. This scheme was further optimized by (Kim and Lee, 1996) to achieve better performance at higher wavenumbers. Despite its advantage of high accuracy, it also shows increased computational cost due to its implicit nature which requires solving a tridiagonal or pentadiagonal matrix. This computational overburden can be reduced by prefactorization of the banded matrix (Ashcroft and Zhang, 2003; Zhou and Zhang, 2011), so that, the matrix can be solved using the simpler $(L + U)$ operation.

In this paper, we used a sixth order accurate, tridiagonal, implicit type spatial derivative operator for both, analysis and simulations in 2D & 3D domain. To reduce its computational cost, we have presented a method based upon LDM^T decomposition followed by a predecomposition technique. This scheme is very much efficient in reducing the computational cost for 2D and 3D domains. Further, we have compared this scheme with various other schemes for 2D and 3D models and discussed (i) the accuracy of derivative operators, (ii) the numerical anisotropy error, and (iii) the associated computational cost. Finally, we have demonstrated the application of implicit finite difference scheme to simulate the seismic wave propagation in 2D and 3D acoustic media to verify its feasibility.

2. Theory

2.1. The wave equation

The governing equation for the propagation of pressure field in an acoustic media with a given velocity structure is given by

$$\partial_t^2 \psi(\mathbf{x}, t) = c^2(\mathbf{x}) \nabla^2 \psi(\mathbf{x}, t) \tag{1}$$

where, $\nabla^2 = \partial_x^2 + \partial_y^2 + \partial_z^2$ is the Laplacian operator; $\psi(\mathbf{x}, t)$ describes the pressure field at a given time instance, t and location, $\mathbf{x} = (x, y, z)$ in the given medium; $c(\mathbf{x})$ is the velocity of the medium within the given computational domain. The medium is evenly discretized in all, x, y, and z direction. The subscripts indices i, j and k, denote node in x, y and z directions, respectively. The total number of nodes along respective directions are N, L, M, as shown in Fig. 1. The discretized wave equation can be written as:

$$\begin{aligned} \mathbf{2D}: \psi_{i,k}^{n+1} &= 2\psi_{i,k}^{n} - \psi_{i,k}^{n-1} + c^2(i,k)\Delta t^2 \left(D_x^2 \psi_{i,k}^{n} + D_z^2 \psi_{i,k}^{n} \right) \\ \mathbf{3D}: \psi_{i,j,k}^{n+1} &= 2\psi_{i,j,k}^{n} - \psi_{i,j,k}^{n-1} + c^2(i,j,k)\Delta t^2 \left(D_x^2 \psi_{i,j,k}^{n} + D_y^2 \psi_{i,j,k}^{n} + D_z^2 \psi_{i,j,k}^{n} \right) \end{aligned} \tag{2}$$

where Δt is the time step and n represents the time instance $t = n\Delta t$; D_x^2, D_y^2 and D_z^2 represent the approximated second order numerical derivatives in the respective directions, as indicated by their subscripts.

2.2. Derivative operator

Consider a 1*D* domain with N number of nodes (Fig. 1). The position of the node is given by $x_i = i\Delta x = ih$, and the function/pressure value at i^{th} node is $f_i = f(x_i)$. The function and its derivative value at all the nodes, can be represented by a vector as $\mathbf{f} = [f_1, f_2, f_3, \ldots f_n]^T$ and $\mathbf{f}'' = [f_1'', f_2'', f_3'', \ldots f_n'']^T$, respectively. A generalized relation between $\mathbf{f}$, and $\mathbf{f}''$ can be established as following.

$$\mathbf{P}\mathbf{f}'' = \frac{1}{h^2}\mathbf{Q}\mathbf{f} \tag{3}$$

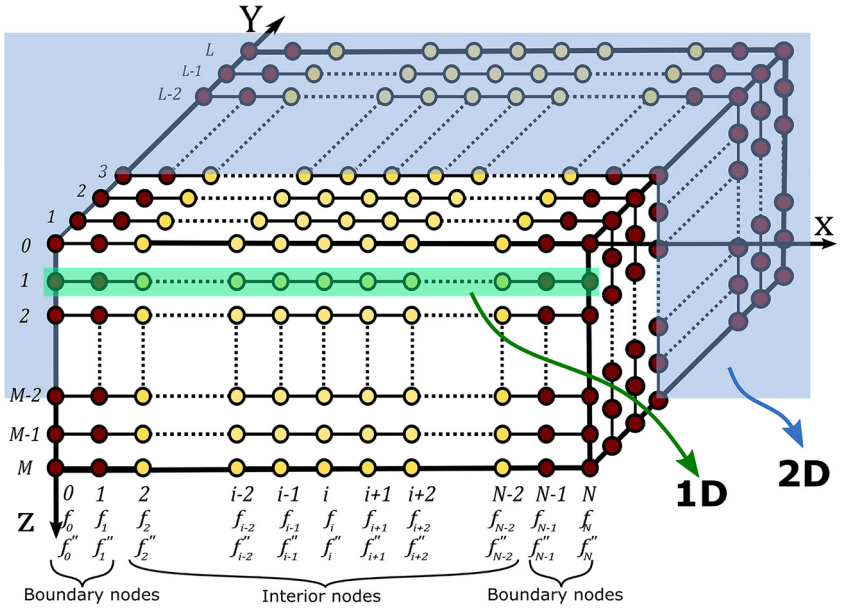

Fig. 1. The figure shows a 3D Grid used for the calculation of derivatives $\partial_x^2 f$. The shaded area and line shows the 2D Grid and 1D Grid as more basic unit of this grid. Number of boundary nodes varies according to the scheme.

where, **P** and **Q** are matrices. To calculate explicit derivative (see Appendix A) at the given node (say i) it is required to have values of the function at all the nodes in the neighborhood of point 'i' (including node 'i'). In this case, **P** is an identity matrix whereas **Q** is a diagonally banded matrix. In contrast to that, the implicit finite difference operator requires values of the function as well as its derivative, at all the nodes in neighborhood of point 'i'. It leads to an implicit relation between the values of the function and its derivative. The structure of both **P** and **Q** matrices is banded and their bandwidth depends on the type of scheme. An example of an implicit scheme is shown below

$$\begin{aligned} &f_i'' + \alpha\left(f_{i+1}'' + f_{i-1}''\right) + \beta\left(f_{i+2}'' + f_{i-2}''\right) \\ &= a\frac{f_{i+1} - 2f_i + f_{i-1}}{h^2} + b\frac{f_{i+2} - 2f_i + f_{i-2}}{(2h)^2} + c\frac{f_{i+3} - 2f_i + f_{i-3}}{(3h)^2} \end{aligned} \tag{4}$$

The coefficients appearing on the left-hand side (α, β) and the right-hand side (a,b,c) should be determined as per the required accuracy. We can obtain higher accuracy by adding more nodes on either side. To determine the coefficients in Eq. (4) (i.e., α, β, a, b, and c), first, we expand the each side using the Taylor series and then match the coefficients of the respective derivative. The comparison provides us the following relation between coefficients.

$$\begin{aligned} h^0, f^2 &: \quad (1 + 2\alpha + 2\beta) = 2\frac{1}{2!}(a + b + c) \\ h^2, f^4 &: \quad 2\frac{1}{2!}\left(\alpha + 2^2\beta\right) = 2\frac{1}{4!}\left(a + 2^2 b + 3^2 c\right) \\ h^4, f^6 &: \quad 2\frac{1}{4!}\left(\alpha + 2^4\beta\right) = 2\frac{1}{6!}\left(a + 2^4 b + 3^4 c\right) \\ h^6, f^8 &: \quad 2\frac{1}{6!}\left(\alpha + 2^6\beta\right) = 2\frac{1}{8!}\left(a + 2^6 b + 3^6 c\right) \\ h^8, f^{10} &: \quad 2\frac{1}{8!}\left(\alpha + 2^8\beta\right) = 2\frac{1}{10!}\left(a + 2^8 b + 3^8 c\right) \end{aligned} \tag{5}$$

The coefficients of the numerical derivative operator with an accuracy of order $O(h^n)$ can be obtained by solving respective equations provided in Eq. (5). For example, a $O(h^6)$ accurate implicit derivative operator (IPT6) can be obtained using the following coefficients: $[a, b, c, \alpha, \beta] = [\frac{12}{11}, \frac{3}{11}, 0, \frac{2}{11}, 0]$ Substituting these values back into the Eq. (4) give rise to a implicit system. Hence the final solution can be written as:

$$\mathbf{f}'' = \frac{1}{h^2}\mathbf{P}^{-1}\mathbf{Q}\mathbf{f} \tag{6}$$

2.3. Boundary conditions

The formulation described above works only for the interior part where the FD stencil remains symmetric. In the boundary region, more nodes are available to one side of the stencil, and thus stencil turns out to be non-symmetric. The interior and boundary regions/nodes are marked in Fig. 1. The unavailability of the nodes at one side in the boundary region demands a reformulation of the implicit FD scheme. It can be written as the following:

$$i = 0: \qquad f_1'' + \alpha f_2'' = \frac{1}{h^2}\left[c_1 f_1 + c_2 f_2 + \ldots + c_n f_n\right] \tag{7}$$

$$i = 1: \qquad \alpha f_1'' + f_2'' + \alpha f_3'' = \frac{1}{h^2}\left[c_1 f_1 + c_2 f_2 + \ldots + c_n f_n\right] \tag{8}$$

It should be noted that coefficients (c_i and α) above are only for the respective boundary node indicated by i. Now following the same procedure as for interior scheme, the unknown coefficients for respective boundary node can be determined. The coefficients for the Eq. (7) and Eq. (8) are given in the Table 1. It can be readily observed that the coefficient for the boundary points i^{th} node and $(N - i)^{th}$ have the same weight but their signs are opposite. We can also say that the boundary values given in Table 1 for right and left edges are flipped in order and opposite in sign to each other.

The first/last boundary node is sometimes constrained by physical constraints, e.g., free surface condition. In such a case, it may be desired to explicitly define the values at the boundary node. For this reason, the end node should be decoupled from the finite difference boundary stencil as given in the Eq. (9). Coefficients for this case can also be determined as for the previous case.

$$i = 0: \qquad f_2'' + \alpha f_3'' = \frac{1}{h^2}\left[c_1 f_1 + c_2 f_2 + \ldots + c_n f_n\right] \tag{9}$$

The interior scheme (Eq. 4) and boundary scheme (Eq. (7), (8) and (9)) completes the numerical scheme for 1D case. It can be extended to find derivative in 2D/3D models in two ways. The first way is to decompose the whole 2D/3D domain 1D vectors along the direction of derivative (say x). Now we can apply the derivative operation recursively over these vectors (Fig. 1). In the second method, a global matrix of second derivative ($\partial^2 f$) for the entire 2D/3D domain can be formulated using the Kronecker product as follow:

$$\begin{aligned} &\mathbf{2D}: \quad [I_M \otimes P]f'' = [I_M \otimes Q]f \\ &\mathbf{3D}: [I_L \otimes I_M \otimes P]f'' = [I_L \otimes I_M \otimes Q]f \end{aligned} \tag{10}$$

where **P** and **Q** are the same matrices as for the 1D case, described above. I_M and I_L are identity matrices of size M and L respectively and $\otimes$ represent the Kronecker product. The resulting matrices for 2D and 3D are of size $N\mathbf{x}M$ and $N\mathbf{x}L\mathbf{x}M$ respectively. The operation preserves the banded nature of the matrices and hence the algorithm to solve the system of equations can be applied in the same way as for the 1D case.

2.4. Spectral characteristics of derivative operators

A Fourier analysis of a derivative gives more insight about the accuracy of derivative in terms of its spectral characteristics. In the wavenumber domain, it is possible to make a comparison of exact and approximated (numerical) derivative's capability to handle the higher order wavenumbers. For this purpose, first we write the Fourier transform pair as follows:

$$\begin{aligned} \hat{f}(\mathbf{k}) &= \mathscr{F}\{f(\mathbf{x})\} = \int_{-\infty}^{\infty} f(\mathbf{x}) e^{-i\mathbf{k}.\mathbf{x}} d\mathbf{x} \\ f(\mathbf{x}) &= \mathscr{F}^{-1}\left\{\hat{f}(\mathbf{k})\right\} = \frac{1}{2\pi}\int_{-\infty}^{\infty} \hat{f}(\mathbf{k}) e^{i\mathbf{k}.\mathbf{x}} d\mathbf{k} \end{aligned} \tag{11}$$

Table 1
The coefficients used to determine derivative at the boundary for the $O(h^6)$ scheme.

Boundary Node	α	c_1	c_2	c_3	c_4	c_5	c_6	c_7
$i = 1$	$\frac{126}{11}$	$\frac{2077}{157}$	$-\frac{2943}{110}$	$\frac{573}{44}$	$\frac{167}{99}$	$-\frac{18}{11}$	$\frac{57}{110}$	$-\frac{9}{136}$
$i = 2$	$\frac{11}{128}$	$\frac{585}{512}$	$-\frac{141}{64}$	$\frac{459}{512}$	$\frac{9}{32}$	$-\frac{81}{512}$	$\frac{3}{64}$	$-\frac{3}{512}$
$i = N - 1$	$\frac{11}{128}$	$\frac{3}{512}$	$-\frac{3}{64}$	$\frac{81}{512}$	$-\frac{9}{32}$	$-\frac{459}{512}$	$\frac{141}{64}$	$-\frac{585}{512}$
$i = N$	$\frac{126}{11}$	$\frac{9}{136}$	$-\frac{57}{110}$	$\frac{18}{11}$	$-\frac{167}{99}$	$-\frac{573}{44}$	$\frac{2943}{110}$	$-\frac{2077}{157}$

where, $\mathscr{F}$ and $\mathscr{F}^{-1}$ are forward and inverse Fourier transforms; f and $\hat{f}$ are the Fourier pairs; and x and k represent the space and wavenumber. The exact derivate of the function holds the following identity:

$$\mathscr{F}\left\{\partial_x^2 f(\mathbf{x},t)\right\} = (i\mathbf{k})^2 \hat{f}(\mathbf{k},t) \quad \text{or} \quad \mathbf{k}^2_{exact} = -\mathbf{k}^2 \tag{12}$$

To analyze the nature of numerical derivatives, we will use the discrete Fourier transform (DFT). For a discretized domain we can write $x_j = \frac{L}{N} j$, where L is the length of the domain and $j = \{0,1,2, \ldots, N\}$. Similarly, the discretized wave number can be written as $k = \frac{2\pi}{L} p$, where $p = \{0,1,2,3, \ldots, N\}$. Thus the DFT can be written as:

$$\hat{f}_j = \mathscr{F}\{f\} = \sum_{p=0}^{N} f_p e^{-i\frac{2\pi jp}{N}}$$
$$f_p = \mathscr{F}^{-1}\left\{\hat{f}\right\} = \frac{1}{N}\sum_{j=0}^{N} \hat{f}_j e^{i\frac{2\pi jp}{N}} \tag{13}$$

Here the function is represented as the superposition of many Fourier modes or plane waves, out of which a single mode or wave can be chosen for analysis. We substitute a single mode in the Eq. (4) to evaluate the spectral characteristics of the FD scheme. Thus the approximated wavenumber for the central difference derivative on the collocated grid can be expressed as:

$$k^2_{approx}(\omega) = \frac{1}{h^2}\left(c_0 + 2\sum_{l=1}^{L} c_l \cos(l\omega)\right) \tag{14}$$

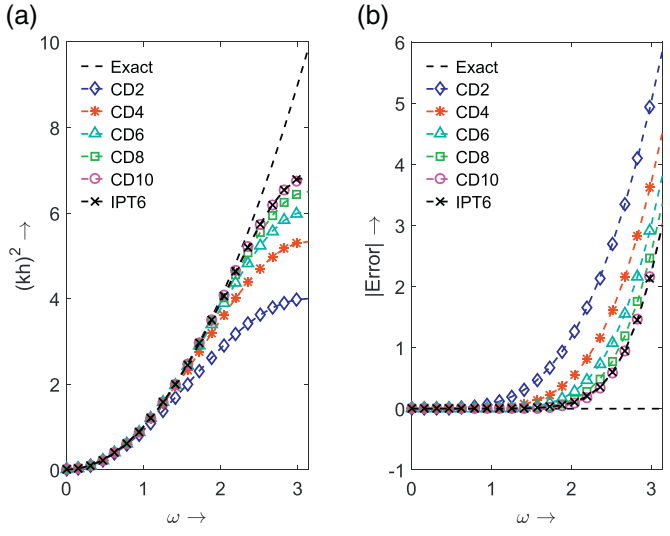

Fig. 2. A comparison of various central difference and implicit derivative operators with exact derivative for (a) Numerically approximated wave number and (b) Error in numerically approximated wave number. Abbrevations used are: CD2, CD4, CD6, CD8, CD10 refers central derivative of order 2, 4, 6, 8 and 10; IPT6 refers to implicit scheme of sixth order; and Exact means the exact derivative.

Similarly, the approximated wavenumber for the implicit finite difference scheme can be written as follows

$$k^2_{approx}(\omega) = -\frac{2a(1-\cos(\omega)) + \frac{2b}{4}(1-\cos(2\omega)) + \frac{2c}{9}(1-\cos(3\omega))}{h^2(1+2\alpha\cos(\omega)+2\beta\cos(2\omega))} \tag{15}$$

where, $\omega = \frac{2\pi}{N} p$.

A comparison of the various CD operator and the implicit derivative operator is presented in Fig. 2a Also, the error for respective numerical operators (in comparison to the exact derivative) is shown in Fig. 2b.

The exact second derivative in spectral domain is parabolic in nature (k^2) and independent of ω (Fig. 2a). On the other hand, numerical derivatives have a dispersive nature (i.e. dependent on ω) which deviates from the exact dispersion curves (Fig. 2a). The deviation at higher wave numbers is very large due to which a numerical simulation may not model the higher modes accurately. To mitigate this effect higher order derivatives are used which have higher formal accuracy, thus leading to an increase in spectral accuracy. The accuracy is also dependent on the type of operator which may give usual formal accuracy but higher spectral accuracy for example the accuracy of the 6th order implicit derivative operator is equivalent to that of 10th order CD operator. The error between dispersion curves of the exact and numerically approximated derivative is shown for each scheme in Fig. 2b.

2.5. Spectral characteristics of the numerical scheme

The spectral analysis of a numerical scheme gives the insight about the numerical anisotropy that an FD scheme can incorporate during simulation. The numerical anisotropy is the variation of the wave phase velocity (v_{ph}) along the different directions. The phase velocity is given by $v_{ph} = \omega/k$, where, ω is the circular frequency and k is the wave number. The phase velocity of the numerical schemes (classical and implicit) can be obtained by taking discretized Fourier transform of the Eq. (2). After rearranging the terms, we can find the relative phase velocity (v_r) as following:

$$\mathbf{2D}: v_r = \frac{\tilde{v}_{ph}}{c} = \frac{2}{\xi p} sin^{-1}\left[\frac{h\xi}{2}\sqrt{K_x^2(p,\theta) + K_z^2(p,\theta)}\right]$$
$$\mathbf{3D}: v_r = \frac{\tilde{v}_{ph}}{c} = \frac{2}{\xi p} sin^{-1}\left[\frac{h\xi}{2}\sqrt{K_x^2(p,\theta,\phi) + K_y^2(p,\theta,\phi) + K_z^2(p,\theta,\phi)}\right] \tag{16}$$

where, $\xi = c\frac{\Delta t}{\Delta x}$, also known as the Courant number; K_x, K_y and K_z are the approximated wavenumber along x, y and z directions respectively; θ and ϕ are the azimuthal and polar angle respectively which determine the direction of propagating wave; $p = kh$ defines the number of grids in a wavelength. For the CD operators, we can express K_x, K_y and K_z as follow:

$$K_x^2(p,\theta,\phi) = \frac{1}{h^2}\left(c_0 + \sum_{l=1}^{L} 2c_l \cos(lp\cos\theta\sin\phi)\right)$$
$$K_z^2(p,\theta,\phi) = \frac{1}{h^2}\left(c_0 + \sum_{l=1}^{L} 2c_l \cos(lp\sin\theta\sin\phi)\right) \tag{17}$$
$$K_z^2(p,\theta,\phi) = \frac{1}{h^2}\left(c_0 + \sum_{l=1}^{L} 2c_l \cos(lp\cos\phi)\right)$$

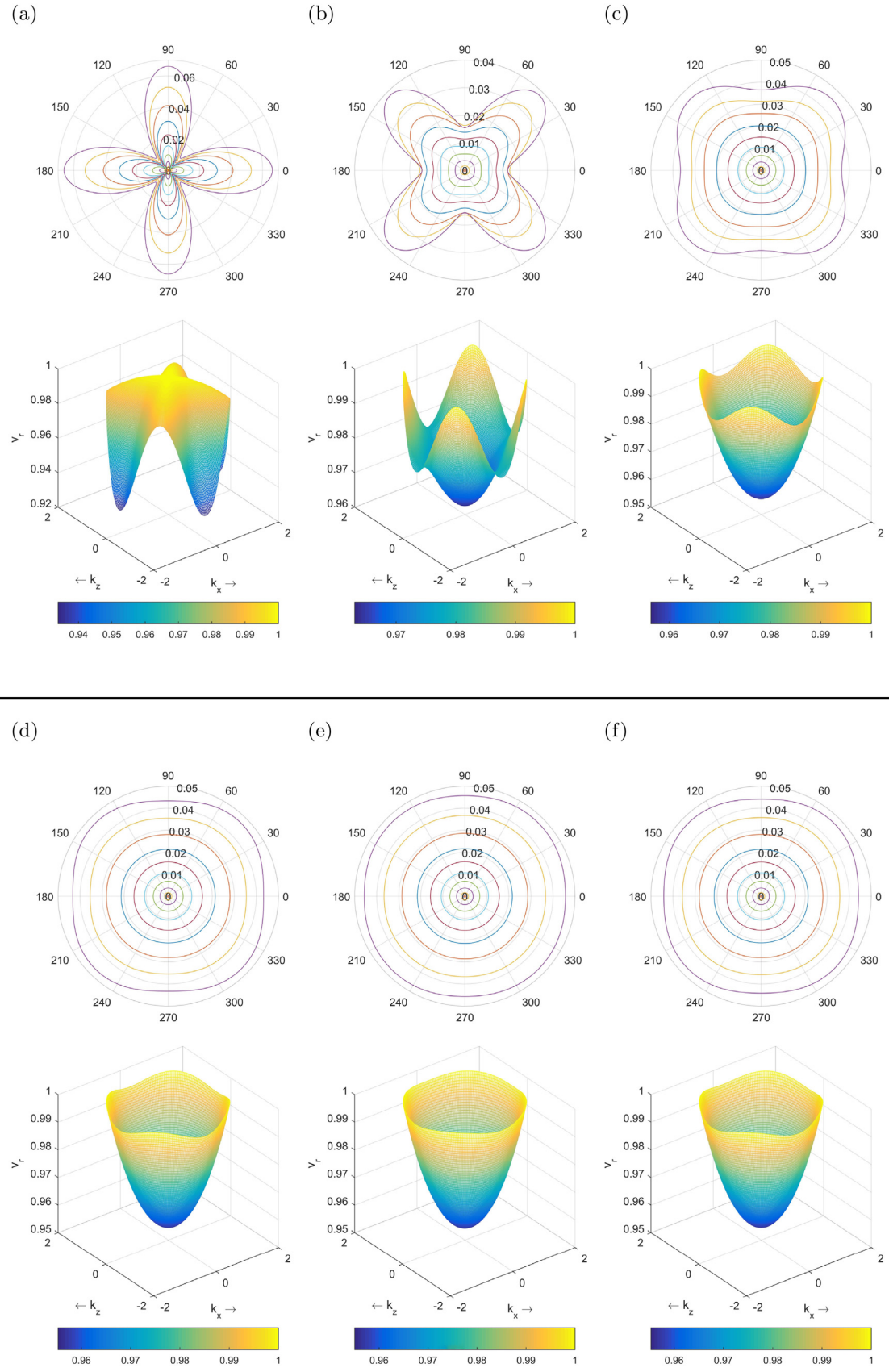

Fig. 3. The plots above depict "the error in phase velocity" (as 2D polar plot, top of each subplot) and "the relative phase velocity" (as 3D surface plot, bottom of each subplot) for various 2D schemes. Subplots (a)-(f) refers to scheme CD2 ($\mathcal{O}(\Delta t^2, \Delta x^2)$), CD4 ($\mathcal{O}(\Delta t^2, \Delta x^4)$), CD6 ($\mathcal{O}(\Delta t^2, \Delta x^6)$), CD8 ($\mathcal{O}(\Delta t^2, \Delta x^8)$) CD10 ($\mathcal{O}(\Delta t^2, \Delta x^{10})$) and IPT6 ($\mathcal{O}(\Delta t^2, \Delta x^6)$), respectively.

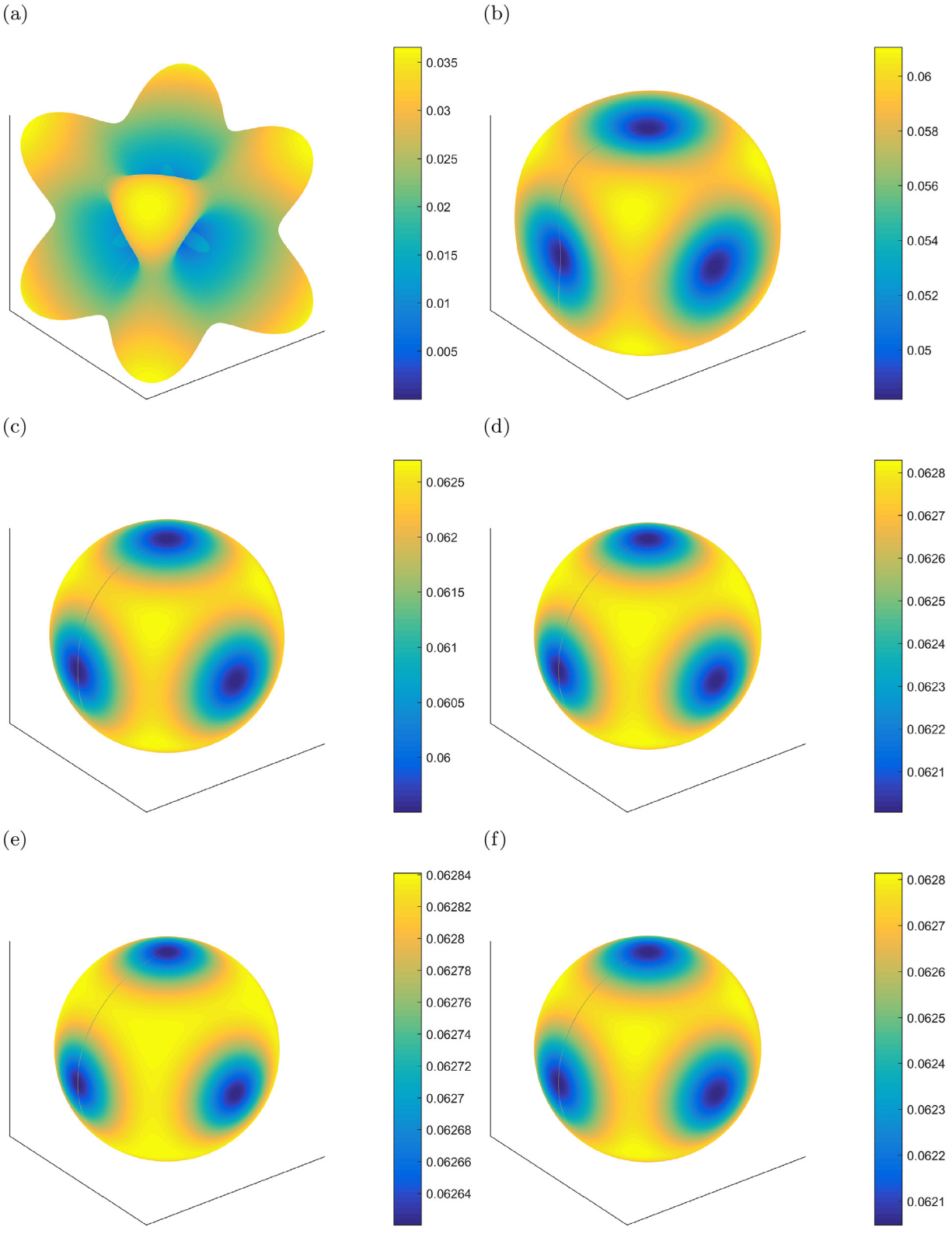

Fig. 4. 3D surface plots above shows the error in phase velocity with respect to direction in 3D space. Plots (a)-(f) refers to 3D scheme corresponding to CD2 ($\mathcal{O}(\Delta t^2, \Delta x^2)$), CD4 ($\mathcal{O}(\Delta t^2, \Delta x^4)$), CD6 ($\mathcal{O}(\Delta t^2, \Delta x^6)$), CD8 ($\mathcal{O}(\Delta t^2, \Delta x^8)$) CD10 ($\mathcal{O}(\Delta t^2, \Delta x^{10})$) and IPT6 ($\mathcal{O}(\Delta t^2, \Delta x^6)$), respectively.

Similarly for the implicit finite difference operators, K_x, K_y and K_z are expressed as

$$K_x^2(p,\theta,\phi) = -\frac{2a[1-\cos(p\cos\theta\sin\phi)]+\frac{2b}{4}[1-\cos(2p\cos\theta\sin\phi)]+\frac{2c}{9}[1-\cos(3p\cos\theta\sin\phi)]}{h^2[1+2\alpha\cos(p\cos\theta\sin\phi)+2\beta\cos(2p\cos\theta\sin\phi)]}$$
$$K_z^2(p,\theta,\phi) = -\frac{2a[1-\cos(p\sin\theta\sin\phi)]+\frac{2b}{4}[1-\cos(2p\sin\theta\sin\phi)]+\frac{2c}{9}[1-\cos(3p\sin\theta\sin\phi)]}{h^2[1+2\alpha\cos(p\sin\theta\sin\phi)+2\beta\cos(2p\sin\theta\sin\phi)]} \quad (18)$$
$$K_z^2(p,\theta,\phi) = -\frac{2a[1-\cos(p\cos\phi)]+\frac{2b}{4}[1-\cos(2p\cos\phi)]+\frac{2c}{9}[1-\cos(3p\cos\phi)]}{h^2[1+2\alpha\cos(p\cos\phi)+2\beta\cos(2p\cos\phi)]}$$

The above relations for 3D can be transformed to 2D by restricting the **k** to azimuthal plane i.e., fixing ϕ equal to $\pi/2$. The error in relative phase velocity, i.e., $\xi = (1 - v_r)$, at different wavenumber is plotted as the polar plot for each scheme (top row in Fig. 3). Relative phase velocities for each scheme are also plotted as 3D polar diagrams for a better understanding of velocity variation with azimuth (bottom row in Fig. 3). For each 3D schemes, error in phase velocities are plotted as 3D surface plots using the highest wavenumber (Fig. 4).

The accuracy of a derivative operator affects the $\tilde{v}_{ph}$ of the discretized wave equation. Ideally, $\tilde{v}_{ph}$ is independent of the direction of wave propagation but it is not true for the disretized wave equation. To analyze this, we plot the 3D surface plots for all the schemes and we can observe that the plots resemble more to paraboloidal shape (ideal case) with an increase in accuracy (bottom figures in Fig. 3). The bounding curve at the top of the paraboloid, or the brim of the cup, is not circular due to the numerical anisotropy; which can be better understood by plotting numerical anisotropy error (top row of Fig. 3). It can be easily observed in Fig. 3 that in all the FD schemes shows higher error as well as distortion for the higher value of p (related to wave numbers). The schemes IPT6 and CD10 demonstrate the least error in v_r with azimuth (nearly circular close to the ideal case) and thus have least numerical anisotropy. For 3D, Fig. 4 shows the variation in relative phase velocity for the highest value of p of respective numerical schemes. Ideally, it should be spherical however due to numerical anisotropy it is not. Similar to 2D, the plots (Fig. 4) shows that the 3D implicit scheme has the least numerical anisotropy as its polar 3D plot is nearly spherical. The 2D/3D plots show that all schemes have the least error in phase velocity along the coordinate axis (i.e., $\theta = n\frac{\pi}{2}, \forall n = \{0,1,2,3\}; \phi = \{0,\pi\}$) and largest error is along the diagonal direction in each quadrant, e.g. $(\theta,\phi) = (\pi/4,\pi/4)$ in the first quadrant.

3. Computational cost

3.1. Theoretical cost (FLOPS)

The computational cost of a numerical operator/scheme can be estimated by computing the total floating point arithmetic operations (FLOPs) which includes basic operations viz. addition, subtraction, multiplication or division of two floating point numbers.

First, we obtain the computational cost in FLOPS for 1D CD operator to estimate $h^2\partial_x^2 f(x_i)$ using the Eq. (A.1). It requires a stencil of length $2L + 1$ to achieve an accuracy of order $2L$. For this, CPU has to perform $2L$ additions and $2L + 1$ multiplications at each node, leading to a total of $5L + 1$ FLOPs for each nodes. However, the number of multiplications can be reduced by considering the symmetrical structure of the operator and thus it now requires a total of $(3L + 1)N$ operations for N nodes. The computational cost for the CD scheme of order 2, 4, 6, 8 and 10 can be estimated as ~$4N$, $7N$, $10N$, $13N$, and $16N$, respectively.

For finding the computational cost of 1D implicit FD operator (of sixth order) we refer to the Eq. (6) and its matrix structure is given in Appendix B. The computational cost for the right-hand side operation (i.e., **Qf**) can be estimated directly as ~$5N$ (two multiplication and three addition operations). It should be noted that the diagonal is zero (except for boundaries) and hence excluded from the calculation. After this step, the right-hand side is reduced to a column and the system can be solved with a tridiagonal matrix solver such as the Thomas algorithm. To compute the cost for the tridiagonal schemes we have to break it into three steps, namely: factorization, forward substitution and backward substitution. The costs associated with these steps are 3 $(N-1)$, $2(N-1)$, and $3(N-1)$ respectively. So the cost for the $\mathbf{P}^{-1}$ operation is about $8(N-1)$ leading to a total cost for complete derivative operation, i.e. $\mathbf{P}^{-1}\mathbf{Qf}$, is equal to $13(N-1)$. It should be noted that the above estimates are obtained after ignoring a few extra multiplications at boundaries since those are negligible in comparison to the total count.

The computational cost ($\mathfrak{C}$) associated with each time step for the given numerical scheme (in 3D) can be calculated as:

$$\mathfrak{C} = 5LMN + MN\mathbf{x}\mathfrak{C}\left(D_x^2 f\right) + LN\mathbf{x}\mathfrak{C}\left(D_y^2 f\right) + LM\mathbf{x}\mathfrak{C}\left(D_z^2 f\right) \quad (19)$$

where, $x_i \in \{x,y,z\}$ and $N_i \in \{L,M,N\}$ for $i \in \{1,2,3\}$ respectively. It should be noted that we assume the quantity $c^2\frac{\Delta t^2}{h^2}$ is pre-calculated.

3.2. Real cost (computation time)

Theoretically, all operations cost is equal to 1 FLOP, however, in reality only addition and subtraction cost equal to 1 FLOP. Single multiplication costs little greater than 1FLOP while single division operation can cost from 2 FLOP to 4 FLOPS (standard in the industry). The flop count also depends upon the machine architecture, compiler, precision, etc. Further, it depends on memory access pattern and temporal dependency in iterations. The computation time for different schemes are shown in Table 2. It shows that the simple **LU** factorization based Implicit scheme costs the most. Our analysis has shown that, there are two operations causing large computation time: (i) division operation in solving tridiagonal matrix, and (ii) recursion process to solve the tridiagonal matrix. The Thomas algorithm requires matrix to decompose into **L** and **U** matrices which causes computation of two new vectors. It involves division operations and recursion process which involves temporal dependency, i.e. i^{th} step dependent on $(i-1)^{th}$. To reduce this time we minimize the division operations using the $\mathbf{LDM^T}$ factorization (Quarteroni et al., 2010) which can be given as follows.

$$\mathbf{P} = \mathbf{LDM^T} \quad (20)$$

Table 2
The computation time for different numerical derivative schemes for 2D models of given sizes (NX^2) and number of time steps NT.

NX, NT	CD2	CD4	CD6	CD8	CD10	IPT6 (**LU**)	IPT6 (**LDM^T**)
Computation time (sec)							
1000^2, 2000	10.23	14.17	18.28	22.50	26.73	62.34	34.20
1000^2, 4000	20.38	28.38	36.53	45.44	53.55	124.67	68.33
2000^2, 2000	40.48	56.62	73.83	90.08	106.20	249.80	140.59
2000^2, 4000	82.25	116.11	148.09	182.30	215.34	500.98	281.09
Normalized time							
1000^2, 2000	1.00	1.38	1.79	2.20	2.61	6.09	3.34
1000^2, 4000	1.00	1.39	1.79	2.23	2.63	6.12	3.35
2000^2, 2000	1.00	1.40	1.82	2.23	2.62	6.17	3.47
2000^2, 4000	1.00	1.41	1.80	2.22	2.62	6.09	3.42

$$\begin{bmatrix} b_1 & c_1 & & 0 \\ a_2 & b_2 & \ddots & \\ & \ddots & & c_{n-1} \\ 0 & & a_n & b_n \end{bmatrix} = \begin{bmatrix} \gamma_1^{-1} & & & 0 \\ a_2 & \gamma_2^{-1} & \ddots & \\ & \ddots & & \\ 0 & & a_n & \gamma_n^{-1} \end{bmatrix} \begin{bmatrix} \gamma_1 & & & 0 \\ 0 & \gamma_2 & \ddots & \\ & \ddots & & \\ 0 & & 0 & \gamma_n \end{bmatrix} \begin{bmatrix} \gamma_1^{-1} & c_1 & & 0 \\ 0 & \gamma_2^{-1} & \ddots & \\ & & \ddots & c_{n-1} \\ 0 & & 0 & \gamma_n^{-1} \end{bmatrix} \tag{21}$$

where,

$$\gamma_i = \begin{cases} 1/a_1 & \text{if } i = 1 \\ (b_i - a_i\gamma_{i-1}c_{i-1})^{-1} & \text{if } i = 2, \ldots, n \end{cases}$$

The solution for the above system is given by

$$y_i = \begin{cases} \gamma_i f_i & \text{for } i = 1 \\ \gamma_i(f_i - b_i\gamma_{i-1}) & \text{for } i = 2, \ldots, n \end{cases} \tag{22}$$

$$f_i'' = \begin{cases} y_n & \text{for } i = 1 \\ y_i - \gamma_i c_i x_{i+1} & \text{for } i = n-1, \ldots, 1 \end{cases} \tag{23}$$

This operation is over one dimension, however, it can save a reasonable amount of time by following the steps given below:

Step 1: Determine the matrix system $\mathbf{Pf}'' = \mathbf{Qf}$ for 1D.
Step 2: Decompose the matrix $\mathbf{P} = \mathbf{LDM^T}$.
Step 3: Calculate and save the vector γ.
Step 4: Apply this 1D operator over all other dimensions iteratively.

This is possible because, for the 2D/3D case, the left-hand matrix remains tridiagonal and acquires the blocked matrix structure. This block can be computed once and then can be carried out by iterating over other dimensions. These operations reduce the computational cost drastically which is reflected in Table 2.

4. Numerical test

To demonstrate the application of implicit finite difference scheme, we carried out the numerical simulation on two types of models, namely, (1) homogeneous and (2) basalt. Both models are also extended to 3D and used for simulation.

1. Homogeneous model:
   The model has a uniform medium velocity of $2000ms^{-1}$. The model is discretized with the same grid spacing of $dh = 4m$ along all direction. The length of each side of the model is $1600m$ and thus have 401 nodes in each horizontal (x or y) and vertical (z) direction. A snippet of wave propagation is shown in Fig. 5a and 5b for 2D and 3D respectively.
2. Basalt model:
   The model is shown in Fig. 6a which was obtained from Singh (2005). The P-wave velocity for different layers ranges between $1.5km/s$ to $4.5km/s$. The 2D basalt model was discretized with same grid spacing, $dh = 5$ along both x and z directions. The dimensions of the model along x and z are $6000m$ and $3550m$ respectively, thus having a total number of nodes as 1201**x**711. A snippet of the wave propagating in this medium and the synthetic seismogram generated for this are shown in Fig. 6b and 6c respectively.
   3D/2.5D basalt model (Fig. 7a) was obtained after extending the previous 2D model laterally in the y-direction. Due to memory

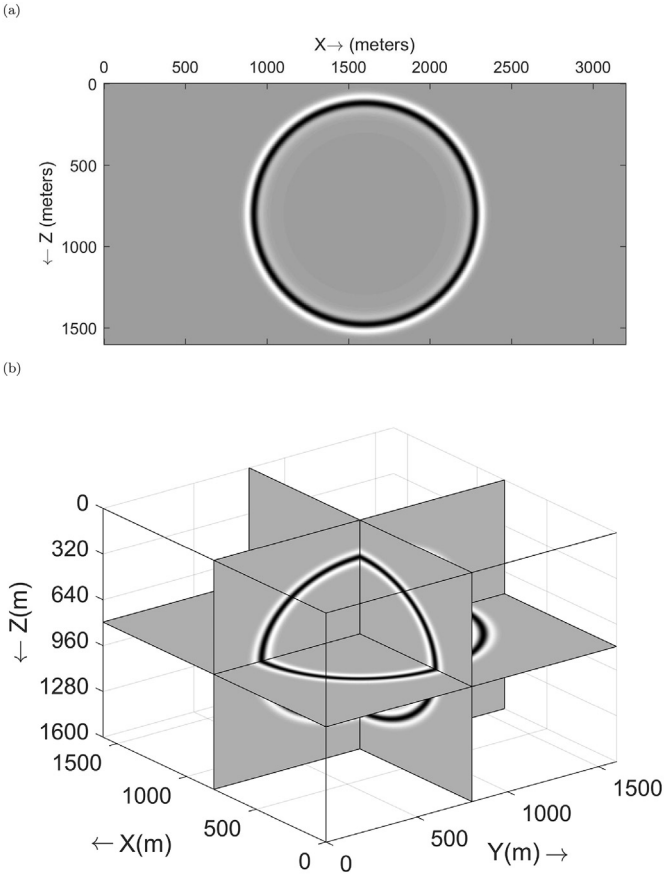

Fig. 5. (a) Snapshot of the seismic wave propagation in a 2D homogeneous acoustic media using implicit scheme. (b) Snapshot of the seismic wave propagation in the 3D homogeneous acoustic media using implicit scheme.

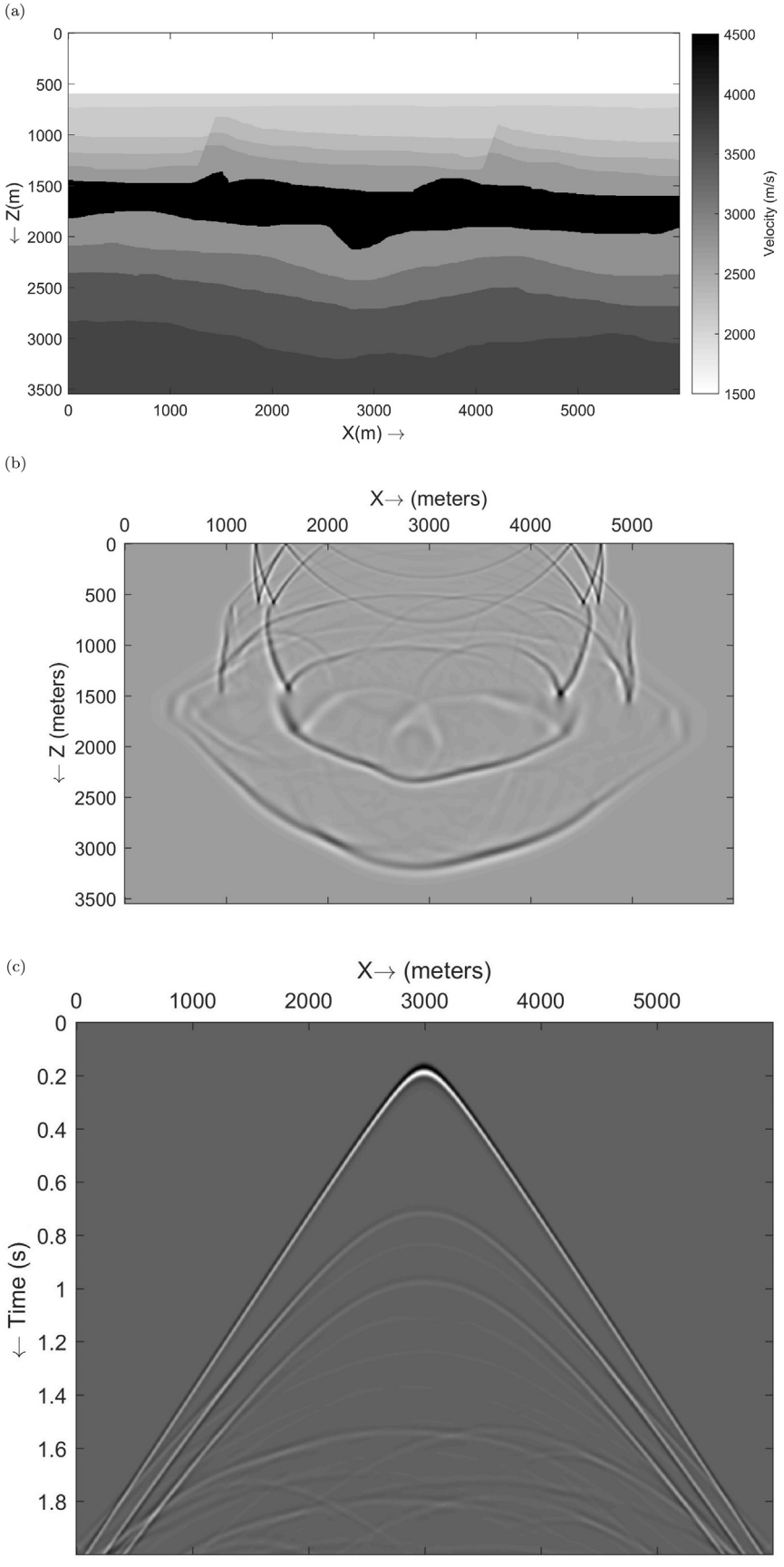

Fig. 6. (a) Velocity model used for seismic wave simulation using the IPT6 scheme. (b) A snapshot of the forward propagating wave in this medium. (c) Synthetic Seismogram generated for above model.

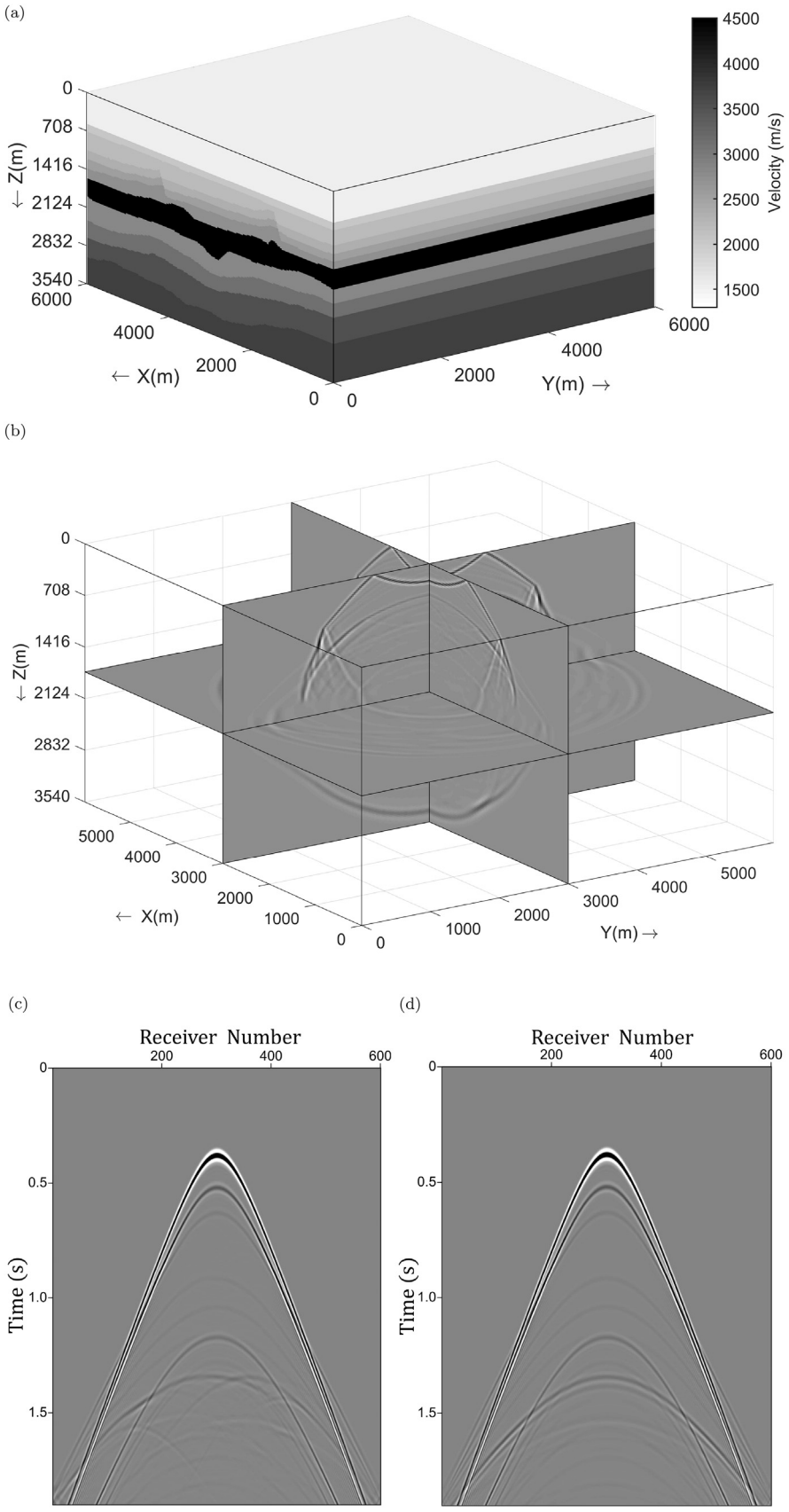

Fig. 7. (a) A 3D/2.5D acoustic velocity model (an extended version of 2D model shown in Fig. 6a) used for seismic wave simulation using the Implicit finite difference scheme. (b) A snapshot of the forward propagating wave in this medium. (c) and (d) shows synthetic seismogram recorded by receiver line along X direction and Y direction, respectively. These mutually orthogonal recording line intersects at source location.

limitation, the model is discretized with $dh = 10m$ which is coarser grid spacing than the 2D case. The dimension of the model along x, y and z are 6000m,6000m and 3550m respectively resulting total number of nodes as N_x**x**N_y**x**N_z = 601**x**601**x**355. A snippet of the wave propagating in the 3D basalt model and the synthetic seismogram generated for this are shown in Fig. 7b and 7c,d respectively.

The source, Ricker wavelet, was inserted as a point source whose signature, $R(t)$, at a given time, t for a given dominant frequency, f_0, is given by

$$R(t) = \left(1-2\pi^2 f_0^2 t^2\right) e^{-\pi^2 f_0^2 t^2} \tag{24}$$

For all the simulation we have used a source with the central frequency of $f_0 = 20Hz$ with appropriate zero time shift for proper initial conditions. Time steps were taken according to stability conditions defined by (Lines et al., 1999) as following:

$$\frac{\Delta t}{h} c < \frac{2}{\sqrt{\sum_{i=-N}^{i=N} | C_i |}} \tag{25}$$

where, C_i are the coefficients of the FD operator. Above criteria is utililzed for the explicit schemes but can also be used as a proxy criteria for the implicit scheme. To determine the stability for the 6^{th} order implicit scheme, coefficients of the 10^{th} order explicit scheme must be used as these two are equivalent. The numerical test demonstrates that the modified 10^{th} order implicit FD scheme is computationally efficient and can be used for accurate simulation.

5. Discussion

As we know that, the high accuracy and reduced numerical anisotropy can be achieved at the expense of computational resources (in terms of memory utilized and the number of CPU cycles). In explicit schemes, it happens due to the higher number of nodes calculation, whereas, implicit scheme requires node summation and a matrix inversion. A theoretical FLOPS calculation shows that both the schemes are equivalent but the simulation time shows otherwise. The compactness of the stencil in the implicit scheme is compelling but its simulation cost restricts its use. The memory requirement for saving matrices P and Q of size N**x**N could be huge. These requirements are reduced by considering the fact that Q can be represented with a few elements and P has banded structure. A theoretical estimate of the FLOP count required by different schemes with respective accuracy shows that the 6th order accurate implicit derivative requires a nearly equal number of FLOPs as that of conventional 8th order CD operator. It makes the 6^{th} Order implicit scheme cheaper than its equivalent 10^{th} order explicit scheme. The computational cost in time (simulation time) reveals contrary to that due to the high cost operations viz. division operator and recursion. One may think of using the parallel algorithms (Sun and Zhang, 2001; Chang et al., 2012; László et al., 2016) to reduce the computational time of implicit scheme however, the explicit parallel scheme still remains more efficient. The implicit scheme described in this work takes advantage of tridiagonal repetitive nature of P matrix for 2D and 3D models and further utilize the LDM^T decomposition to improve the performance. The final cost of the implicit scheme is a little higher than the explicit scheme (1.27 times).

At this point, it is worth to emphasize on the boundaries, which also could be a source of, though small, error. At the interior part of a domain, the stencil has an equal number of nodes on both sides making it symmetric and thus zero phase error. The boundary stencil has an unequal number of nodes on its sides so it becomes non-symmetric and thus can introduce some phase errors. In a numerical scheme, this effect is unavoidable due to the one-sided nature of stencil. However, this effect is reduced in the implicit scheme as it utilizes much lesser boundary points than the CD scheme. Further, the effect can be neglected as it is restricted to a very small region, consisting of a few nodes only. We also found that the boundary conditions are stable and can be readily used for the seismic wave simulation in acoustic media.

6. Conclusions

In this paper, we have presented an efficient method for acoustic wave simulation using implicit FD operator. We have compared various aspects of the implicit scheme with various CD schemes of different orders, viz. accuracy of the numerical derivative operator, error in phase velocity due to discretization, and computational cost. The spectral characteristics of 6^{th} order implicit scheme are equivalent to 10^{th} order conventional CD schemes. The error characteristics (i.e., minima and maxima locations) for all the schemes, including the implicit scheme, are similar. This scheme has the advantage of smaller stencil which are more local in nature than the lengthy explicit FD operator. Even for boundary implementation, it requires a lesser number of points, as compared to CD scheme and therefore introduces less phase error. The 6^{th} order implicit scheme can be easily extended from 1D to higher dimensions (2D/3D) while having computational cost almost equivalent to 10^{th} order explicit scheme. The applicability and stability of the scheme were demonstrated by successfully simulating the wave propagation in 2D and 3D acoustic media using a sixth order accurate implicit FD scheme.

Data and resources

All data used in this paper came from published sources listed in the references.

Acknowledgment

AM is grateful to CSIR-UGC for awarding him the fellowship to carry out his Ph.D. work. RKT is grateful to Department of Atomic Energy (DAE) for awarding Raja Ramanna Fellowship. Authors are also thankful to AcSIR-NGRI Coordinator & Director NGRI for support and permission to publish the results. This work was carried out under the project MLP-6402-28.

Appendix A.

Taylor expansion based second order derivative operators are defined as follows

$$\frac{\partial^2 f(x_i)}{\partial^2 x} = \frac{1}{h^2} \sum_{l=-L}^{L} c_l f\left(x_{(i+l)}\right) \tag{A.1}$$

For desired order of accuracy, the coefficients can be chosen from the Table A1.

Appendix B.

Following is the example of the sixth order accurate implicit finite difference operator matrix. It is obtained by incorporating the coefficients for interior (using Eq. (4)) as well as boundary (using Eq. (7) and (8))

$$\begin{pmatrix} 1 & \frac{126}{11} & 0 & 0 & \cdots \\ \frac{11}{128} & 1 & \frac{11}{128} & 0 & \cdots \\ \frac{2}{11} & 1 & \frac{2}{11} & 0 & \cdots \\ 0 & \frac{2}{11} & 1 & \frac{2}{11} & \ddots \\ 0 & 0 & \frac{2}{11} & 1 & \ddots \\ \vdots & \vdots & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} f_1'' \\ f_2'' \\ f_3'' \\ f_4'' \\ f_5'' \\ \vdots \end{pmatrix} = \begin{pmatrix} \frac{2077}{157} & -\frac{2943}{110} & \frac{573}{44} & \frac{167}{99} & -\frac{18}{11} & \frac{57}{110} & -\frac{9}{136} & 0 & \cdots \\ \frac{585}{512} & -\frac{141}{64} & \frac{459}{512} & \frac{9}{32} & -\frac{81}{512} & \frac{3}{64} & -\frac{3}{512} & 0 & \cdots \\ \frac{3}{11} & \frac{12}{11} & 0 & \frac{12}{11} & \frac{3}{11} & 0 & 0 & 0 & \cdots \\ 0 & \frac{3}{11} & \frac{12}{11} & 0 & \frac{12}{11} & \frac{3}{11} & 0 & 0 & \cdots \\ 0 & 0 & \frac{3}{11} & \frac{12}{11} & 0 & \frac{12}{11} & \frac{3}{11} & 0 & \cdots \\ 0 & 0 & 0 & \frac{3}{11} & \frac{12}{11} & 0 & \frac{12}{11} & \frac{3}{11} & \ddots \\ \vdots & \vdots & \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \\ f_5 \\ \vdots \end{pmatrix} \tag{B.1}$$

Table A1
Numerically approximated 2^{nd} order derivative's (∂_x^2) coefficients for various order accurate central difference numerical derivative on the collocated grid.

Order	x_{i-5}	x_{i-4}	x_{i-3}	x_{i-2}	x_{i-1}	x_i	x_{i+1}	x_{i+2}	x_{i+3}	x_{i+4}	x_{i+5}
2					1	−2	1				
4				$\frac{-1}{12}$	$\frac{4}{3}$	$\frac{-5}{2}$	$\frac{4}{3}$	$\frac{-1}{12}$			
6			$\frac{1}{90}$	$\frac{-3}{20}$	$\frac{3}{2}$	$\frac{-49}{18}$	$\frac{3}{2}$	$\frac{-3}{20}$	$\frac{1}{90}$		
8		$-\frac{1}{560}$	$\frac{8}{315}$	$-\frac{1}{5}$	$\frac{8}{5}$	$-\frac{205}{72}$	$\frac{8}{5}$	$\frac{-1}{5}$	$\frac{8}{315}$	$\frac{-1}{560}$	
10	$\frac{1}{3150}$	$-\frac{5}{1008}$	$\frac{5}{126}$	$-\frac{5}{21}$	$\frac{5}{3}$	$-\frac{5269}{1800}$	$\frac{5}{3}$	$-\frac{5}{21}$	$\frac{5}{126}$	$-\frac{5}{1008}$	$\frac{1}{3150}$

References

Alterman, Z., Karal, F., 1968. Propagation of elastic waves in layered media by finite difference methods. Bull. Seismol. Soc. Am. 58, 367–398.

Ashcroft, G., Zhang, X., 2003. Optimized prefactored compact schemes. J. Comput. Phys. 190, 459–477.

Chang, L.W., Stratton, J.A., Kim, H.S., Hwu, W.M.W., 2012. A scalable, numerically stable, high-performance tridiagonal solver using gpus. Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis. IEEE Computer Society Press, p. 27.

Collatz, L., 1960. The Numerical Treatment of Differential Equations. vol. 60. Springer Science & Business Media.

Fan, N., Zhao, L.F., Xie, X.B., Tang, X.G., Yao, Z.X., 2017. A general optimal method for a 2d frequency-domain finite-difference solution of scalar wave equation. Geophysics 82, T121–T132.

Gosselin-Cliche, B., Giroux, B., 2014. 3d frequency-domain finite-difference viscoelastic-wave modeling using weighted average 27-point operators with optimal coefficients. Geophysics 79, T169–T188. https://doi.org/10.1190/geo2013-0368.1.

Harlow, F.H., Welch, J.E., 1965. Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface. Phys. Fluids 8, 2182–2189.

Holberg, O., 1987. Computational aspects of the choice of operator and sampling interval for numerical differentiation in large-scale simulation of wave phenomena. Geophys. Prospect. 35, 629–655.

Jo, C.H., Shin, C., Suh, J.H., 1996. An optimal 9-point, finite-difference, frequency-space, 2-d scalar wave extrapolator. Geophysics 61, 529–537.

Kelly, K., Ward, R., Treitel, S., Alford, R., 1976. Synthetic seismograms: a finite-difference approach. Geophysics 41, 2–27.

Kim, J.W., Lee, D.J., 1996. Optimized compact finite difference schemes with maximum resolution. AIAA J. 34, 887–893.

Kosloff, D., Pestana, R.C., Tal-Ezer, H., 2010. Acoustic and elastic numerical wave simulations by recursive spatial derivative operators. Geophysics 75, T167–T174.

László, E., Giles, M., Appleyard, J., 2016. Manycore algorithms for batch scalar and block tridiagonal solvers. ACM Trans. Math. Softw. (TOMS) 42, 31.

Lele, S.K., 1992. Compact finite difference schemes with spectral-like resolution. J. Comput. Phys. 103, 16–42.

Levander, A.R., 1988. Fourth-orderfinite-differencep-sv seismograms. Geophysics 53, 1425–1436.

Lines, L.R., Slawinski, R., Bording, R.P., 1999. A recipe for stability of finite-difference wave-equation computations. Geophysics 64, 967–969.

Liu, Z.l., Song, P., Li, J.s., Li, J., Zhang, X.b., 2015. An optimized implicit finite-difference scheme for the two-dimensional helmholtz equation. Geophys. J. Int. 202, 1805–1826.

Madariaga, R., 1976. Dynamics of an expanding circular fault. Bull. Seismol. Soc. Am. 66, 639–666.

Quarteroni, A., Sacco, R., Saleri, F., 2010. Numerical Mathematics. vol. 37. Springer Science & Business Media.

Saenger, E.H., Bohlen, T., 2004. Finite-difference modeling of viscoelastic and anisotropic wave propagation using the rotated staggered grid. Geophysics 69, 583–591.

Shin, C., Sohn, H., 1998. A frequency-space 2-d scalar wave extrapolator using extended 25-point finite-difference operator. Geophysics 63, 289–296.

Singh, S.K., 2005. Sub-Basalt Imaging Modeling and Demultiple. Master's Thesis. Texas A&M University.

Štekl, I., Pratt, R.G., 1998. Accurate viscoelastic modeling by frequency-domain finite differences using rotated operators. Geophysics 63, 1779–1794.

Sun, X.H., Zhang, W., 2001. A parallel two-level hybrid method for diagonal dominant tridiagonal systems. Parallel and Distributed Processing Symposium., Proceedings International, IPDPS 2002, Abstracts and CD-ROM, IEEE, p. 6.

Tam, C.K., Webb, J.C., 1993. Dispersion-relation-preserving finite difference schemes for computational acoustics. J. Comput. Phys. 107, 262–281.

Virieux, J., 1986. P-sv wave propagation in heterogeneous media: velocity-stress finite-difference method. Geophysics 51, 889–901.

Virieux, J., Calandra, H., Plessix, R.É., 2011. A review of the spectral, pseudo-spectral, finite-difference and finite-element modelling techniques for geophysical imaging. Geophys. Prospect. 59, 794–813.

Wang, E., Liu, Y., Sen, M.K., 2016. Effective finite-di fference modelling methods with 2-d acoustic wave equation using a combination of cross and rhombus stencils. Geophys. J. Int. 206, 1933–1958.

Zhou, H., Zhang, G., 2011. Prefactored optimized compact finite-difference schemes for second spatial derivatives. Geophysics 76, WB87–WB95.