

2-D seismic wave propagation using the distributional finite-difference method: further developments and potential for global seismology

Yder Masson¹, Chao Lyu², Peter Moczo³, Yann Capdeville⁴,
Barbara Romanowicz^{2,5,6} and Jean Virieux^{7,8}

¹LFCR/e2s, Université de Pau et des Pays de l'Adour, avenue de l'Université 64000 Pau, France. E-mail: yder.masson@gmail.com

²Berkeley Seismological Laboratory, University of California, Earth & Planetary Science, Berkeley, CA 94720-4767, USA

³Comenius University Bratislava, Faculty of Mathematics, Physics and Informatics, Mlynska dolina F1, 84248 Bratislava, Slovakia

⁴Université de Nantes/CNRS, LPG, UMR6112, 2 Chem. de la Houssinière Bâtiment 4, 44300 Nantes, France

⁵Institut de Physique du Globe, 1 Rue Jussieu, 75005 Paris, France

⁶Collège de France, 11 Pl. Marcelin Berthelot, 75231 Paris, France

⁷ISTerre - Institut des Sciences de la Terre, 1381 Rue de la Piscine, 38610 Gières, Grenoble, France

⁸UGA - Université Grenoble Alpes, 621 Av. Centrale, 38400 Saint-Martin-d'Hères, Grenoble, France

Accepted 2024 January 16. Received 2023 October 24; in original form 2023 May 30

SUMMARY

We present a time-domain distributional finite-difference scheme based on the Lebedev staggered grid for the numerical simulation of wave propagation in acoustic and elastic media. The central aspect of the proposed method is the representation of the stresses and displacements with different sets of B-splines functions organized according to the staggered grid. The distributional finite-difference approach allows domain-decomposition, heterogeneity of the medium, curvilinear mesh, anisotropy, non-conformal interfaces, discontinuous grid and fluid–solid interfaces. Numerical examples show that the proposed scheme is suitable to model wave propagation through the Earth, where sharp interfaces separate large, relatively homogeneous layers. A few domains or elements are sufficient to represent the Earth's internal structure without relying on advanced meshing techniques. We compare seismograms obtained with the proposed scheme and the spectral element method, and we show that our approach offers superior accuracy, reduced memory usage, and comparable efficiency.

Key words: Numerical modelling; Numerical solutions; Computational seismology; Seismic anisotropy; Wave propagation; Structure of the Earth.

1 INTRODUCTION

Over time, seismology has become increasingly dependent on high-performance computing resources to compute the seismic wavefield, and efficient algorithms for solving the wave equation have been developed. Different methods have been adopted within the earthquake seismology community and in exploration geophysics.

Many earthquake seismologists has adopted the spectral element method (SEM), beginning with pioneering modelling of the crust (Tape *et al.* 2009) and uppermost mantle (Fichtner 2010) at local and regional scales, since then extended to many regions of the world (e.g. Magnoni *et al.* (2022)). Notably, the latest global tomographic models that unveil the structure of our planet's deep interior typically required millions of CPU hours for their construction (French & Romanowicz 2014; Bozdağ *et al.* 2016; Lei *et al.* 2020). Indeed, until the advent of the SEM, seismic waveform modelling of the Earth's mantle at the global scale relied on the computation of synthetics using normal mode perturbation theory (Woodhouse & Dziewonski 1984; Li & Romanowicz 1996; Mégnin & Romanowicz 2000), the validity of which required strong assumptions on the strength and smoothness of the 3-D heterogeneity. The introduction of the SEM to global seismology (Komatitsch & Vilotte 1998; Komatitsch & Tromp 1999) opened up a new era, providing a comparatively efficient method for the accurate numerical computation of the entire seismic wavefield, in the presence of topography and potentially arbitrary lateral variations of 3-D structure.

Given the computational challenges associated with full 3-D simulations, where the computational cost increases with the fourth power of frequency, several approaches have been introduced to approximate the wavefield under specific conditions. For example, when assuming an axisymmetric or partly axisymmetric model is a good enough approximation to compute the wavefield generated by localized structures.

This led to the development of C-SEM, a ‘coupled mode-SEM’ code (Capdeville *et al.* 2003a) in which SEM computations in the mantle were coupled through a DtN operator to fast 1D normal mode computations in the core, which was used in the development of the first SEM-based global radially anisotropic shear velocity models of the upper mantle (Lekić & Romanowicz 2011; French *et al.* 2013) and of the whole mantle (French & Romanowicz 2014). Capdeville *et al.* (2003b) extended the C-SEM method to an annulus of SEM coupled above and below with 1-D modes, enabling the modelling of ultra-low velocity zones at the base of the mantle in 3-D, using diffracted *S* waves at periods down to 8 s (Cottaar & Romanowicz 2012) or even shorter (To *et al.* 2005). Hypothesizing that it is more important to accurately represent the heterogeneity in the vertical plane containing the source and the receiver, Nissen-Meyer *et al.* (2014) proposed an efficient global 2.5-D solver, AxiSEM, in which cylindrical symmetry is assumed around each source. The requirement of cylindrical symmetry in AxiSEM was later released in AxiSEM3D (Leng *et al.* 2019), which allows for a smooth variation of the structure in the direction perpendicular to the great circle plane. AxiSEM3D has already received much attention in the global seismological community, in particular for the modelling of complex structures in D” (Li *et al.* 2022). Meanwhile, SPECFEM3D-GLOBE (Tromp *et al.* 2008), which makes no approximations on the structure, is open-source and has been significantly optimized over time, has now become the standard for full seismic wavefield computations at regional and global scales and has been used in particular for the development of global radially anisotropic shear velocity models GLAD-M15 (Bozdağ *et al.* 2016) and GLAD-M25 (Lei *et al.* 2020). Other groups have developed their own SEM codes, notably in the framework of the collaborative seismic earth model (CSEM, e.g. Fichtner *et al.* 2018) or in the framework of regional and crustal seismic imaging (Lu *et al.* 2018; Trinh *et al.* 2019). These codes are however not open source. Although very higher-order spectral element methods, such as the 12th-order method, are more computationally efficient than the classical 4th-order SEM (Lyu *et al.* 2020), the overall computational efficiency of SEM decreases significantly when its order exceeds 20 due to the extremely small available time step, which limits the usefulness of super-high-order (≥ 20 th-order) SEM for global-scale simulations.

Still, there is a need for more efficient methods to reach the higher frequencies required for the study of complex structures in the deep Earth, as the computational burden of the SEM is prohibitive for global waveform modelling at periods shorter than 10 s. In this study, we investigate an alternative to SEM for modelling seismic wave propagation at the global scale called the distributional finite-difference method (DFDM) introduced in Masson (2022). The proposed algorithm shall allow for domain decomposition, medium heterogeneity, curvilinear mesh, anisotropy, non-conformal interfaces, discontinuous grids, free surfaces and fluid–solid interfaces.

SEM is the favoured approach to model wave propagation at the planet’s scale; on the other hand, the finite-difference method (FDM) is predominant in exploration geophysics and earthquake ground motion simulations in local surface sedimentary structures (e.g. Virieux *et al.* 2011; Moczo *et al.* 2014, 2021). This apparent discrepancy may be due to communities of practice, software availability, and, most likely, the nature of the seismic data and the model geometry. Active seismic imaging relies primarily on body wave analysis with a focus on reflections, while surface waves represent an essential part of the data processed in seismology and global tomography. Thus, SEM, especially suited for surface wave modelling, is preferred in seismology. FDM needs to be adapted to model surface waves accurately. Otherwise, it is remarkably simple and efficient: an attractive feature in exploration geophysics. Fluid–solid interfaces also need to be modelled accurately and influence the method to be chosen (De Basabe & Sen 2015). There are, of course, numerous exceptions to the general statements above. FDM has been successfully used to model wave propagation at the global and regional scale (Igel & Weber 1995; Igel *et al.* 1995; Chaljub & Tarantola 1997; Jahnke *et al.* 2005; Kawai *et al.* 2006; Li *et al.* 2014). Summation-by-part finite-difference schemes, named SBP-SAT approaches, which naturally account for the free surface, have reached recently an improved accuracy (Sjögreen & Petersson 2012; Petersson & Sjögreen 2015). Such an advanced SBP-SAT software is openly available to the seismology community (SW4, Petersson & Sjögreen 2017). SEM has also been used for exploration and geotechnical applications (Zheng *et al.* 2014). The literature in FDM is abundant, and we refer the reader to review papers (Virieux *et al.* 2011; Moczo *et al.* 2014, 2021), and the references therein for a comprehensive review.

We focus on the staggered-grid finite-difference approach, which is relevant in this study. We highlight the pioneering work of Yee (1966) in electromagnetism, Madariaga (1976) in seismic source modelling, and Virieux (1986) in seismic wave propagation. In staggered-grid finite-difference approaches, the field variables are discretized on different grids, leading to centred finite-difference operators with improved accuracy for spatial derivatives of fields. Further, the staggered-grid approach effectively annihilates the spurious (non-physical) parasitic modes that are problematic in FDM (see, e.g. New *et al.* 1998; Dovgilevich & Sofronov 2015). Thus, this staggered-grid FDM is praised for its simplicity and computational efficiency and is widely used for the 3-D modelling of seismic wave propagation (Matsushima 1998; Graves 1993; Olsen 1995; Pitarka *et al.* 1998; Wald & Graves 1998; Kristek *et al.* 1999). The former Virieux algorithm (Virieux 1986) permits limited anisotropy (i.e. orthotropic) and can accommodate orthogonal curvilinear coordinates. Several staggered-grid FDM schemes have been introduced to account for general anisotropy and non-orthogonal curvilinear grids, notably, partly staggered grid scheme (e.g. Saenger & Bohlen 2004) and Lebedev scheme (e.g. Lisitsa & Vishnevskiy 2010; de la Puente *et al.* 2014). Even though they do not strictly fall in the staggered-grid FDM category, one should also mention the collocated approaches relying on the MacCormack scheme where left and right FD operators are used to prevent the parasitic modes from appearing (Zhang & Chen 2006; Zhu *et al.* 2009; Lan & Zhang 2011; Sun *et al.* 2016). Bernth & Chapman (2011) emphasizes that the Lebedev scheme is more efficient and accurate than the partly staggered grid scheme. The stability and grid dispersion of the staggered-grid FDM is analysed carefully in Moczo *et al.* (2000). Kristek *et al.* (2002) and Bohlen & Saenger (2006) addressed the difficulty of accurately modelling the free surface using staggered-grid FDM. Nonetheless, several approaches have been introduced to address this issue (Lombard *et al.* 2008; Zhang *et al.* 2012a, b; de la Puente *et al.* 2014; Gao *et al.* 2015; Sun *et al.* 2016; Shragge & Konuk 2020; Zang *et al.* 2021). An aspect of special importance in the finite-difference modelling in both seismic exploration and ground motion simulations is the implementation of material interfaces. It is not trivial in the case of a general shape and

position in the uniform grid. The problem has been recently addressed by Mittet (2017), Jiang & Zhang (2021), Koene *et al.* (2022) and Moczo *et al.* (2022). In this work, we build up the Lebedev approach to construct our DFDM algorithm.

Besides SEM, numerous methods that decompose the computational domain in multiple elements exist. In particular, the classical finite-element method (FEM; Lysmer & Drake 1972; Smith 1975; Marfurt 1984), Boundary-element methods (BEM; Bouchon & Coutant 1994; Papageorgiou & Pei 1998; Ba & Gao 2017), finite-volume methods (FVM; Dormy & Tarantola 1995; Dumbser *et al.* 2007) and Discontinuous Galerkin methods (DGM; Käser *et al.* 2008; De Basabe *et al.* 2008; de la Puente *et al.* 2008; Étienne *et al.* 2010; Warburton 2013; Bonnasse-Gahot *et al.* 2018). Mixed FEM approaches following partially the staggered-grid strategy have been developed (Nédélec 1980; Bécache *et al.* 2002). Over the last decade, Isogeometric Analysis (IGA; Hughes *et al.* 2005; Auricchio *et al.* 2010) has garnered significant attention within the field of structural engineering. This approach discretize partial differential equations (PDEs) using the same type of functions used in Computer Aided Design (CAD): Non-Uniform Rational B-Splines (NURBS). These NURBS not only provide a precise representation of the geometric aspects of the computational domain but also serve as the basis for defining the solution space for PDEs. Element-based approaches allow to mesh complex geological features and are well suited to model free surfaces accurately and discontinuities thanks to their weak formalism. However, they lead to several numerical challenges. They are more difficult to implement than FDM, often more expensive in computational time, and their accuracy depends on the quality of the meshing (Virieux *et al.* 2011). One should also mention pseudo-spectral methods (PSM) that are highly accurate but less flexible for representing media with complex geometries (Fornberg 1987; Carcione 1996).

Masson (2022) introduced a distributional finite-difference method (DFDM) that combines some essential features of FDM, FEM, SEM, IGA and DGM. DFDM decomposes the space into multiple elements with arbitrary sizes. Thus, a volumetric mesh may be used to describe the model better whenever needed. Within the elements, DFDM has an algorithmic structure similar to FDM and relies on compactly supported operators, which ensures efficiency. Our algorithm relies on B-spline bases similarly to IGA with two important differences. First, DFDM does not require designing optimal quadrature rules (e.g. Zou *et al.* 2022) for achieving efficiency. The mass and stiffness matrices are assembled independently prior to the computations, while the element shapes and heterogeneity are accounted for approximately using weights associated with virtual gridpoints. This approach is well-suited for geophysical applications where the exact structure of the medium in which waves propagate is rarely known precisely. Secondly, in DFDM, field variables are represented using different basis functions (similarly to staggered grid FDM). This has the dual benefit of reducing the number of degrees of freedom in the problem and eliminating parasitic modes that are notoriously problematic in wave propagation modelling (Virieux 1986; Lisitsa & Vishnevskiy 2010). Finally, in DFDM, the numerical wavefield is discontinuous between neighbouring elements, both the displacement and the stress can be defined explicitly at the boundaries, similarly to DGM.

The algorithm proposed by Masson (2022) relies on a MacCormack type of strategy to remove the parasitic modes; all the displacements are represented in a first basis (with the basis function skewed to the left), and all the stresses are represented in a second basis (with basis functions skewed to the right). This algorithm also shares the structure of the partly staggered-grid schemes with all the displacement components located at one grid position and all the stress components located at another grid position (e.g. Saenger & Bohlen 2004). Masson & Virieux (2023) proposed a DFDM scheme analogous to the classic algorithm from Virieux (1986) where field variables are represented in different bases organized according to the staggered grid: specific differential operators switching consistently from one basis to the other one when estimating a field and its spatial derivative are designed as we shall also see in this work (this operation is not performed in mixed FEM approaches). They showed that the DFDM algorithm is more accurate and faster than its FDM counterpart. Interestingly, the staggered approach permits the computation of the partial derivatives by applying 1-D operators in a single spatial direction. In contrast, the method of Masson (2022) requires using the 1-D operators in the two spatial directions. The switch between bases is still performed in both strategies. The two latter approaches also differ in the time scheme. Masson (2022) uses a centred second-order time integration scheme with a displacement formulation and Masson & Virieux (2023) use a first-order leap-frog strategy. Last, the two studies account for material heterogeneity in different manners. Masson (2022), includes material heterogeneity using a mass lumping technique and Masson & Virieux (2023) discretizes the material properties on a virtual grid associated with the basis functions. For clarity and to outline the similarities with FDM, Masson & Virieux (2023) only considered a single isotropic element. Nonetheless, their algorithm applies to limited anisotropy (i.e. orthotropic material) and adapts to orthogonal curvilinear grid (e.g. Xie *et al.* 2002).

In this study, we expand the Masson & Virieux (2023) algorithm to the Lebedev grid, which consists of two Virieux grids. It is needed in seismology to model wave propagation in arbitrary anisotropic media and to work with non-orthogonal curvilinear coordinates. Further, it allows the meshing of complex geological features and accurately model interfaces, particularly free surfaces and fluid–solid interfaces with topography. A key aspect of this paper is the evaluation of the proposed algorithm for modelling seismic wave propagation through the Earth at the global scale.

This manuscript is organized as follows. The first section starts with the second-order elastodynamic system of equations. We apply the non-orthogonal curvilinear coordinate transformation. Then, we discretize the field variables using the basis functions organized according to the Lebedev grid. Thereafter, we briefly recall how a dual basis can be used to approximate functions and their derivatives. We proceed with the derivation of the 1-D DFDM operators used for approximating the partial derivatives. The 1-D operators are then extended to 2-D using a tensor product. The treatment of boundary conditions and interfaces is discussed together with the introduction of the 2-D operators. Finally, we detail our 2-D time-domain algorithm that models wave propagation. In the second section, we present numerical examples demonstrating that the proposed numerical algorithm can accurately and efficiently model wave propagation through the Earth at the global scale. We

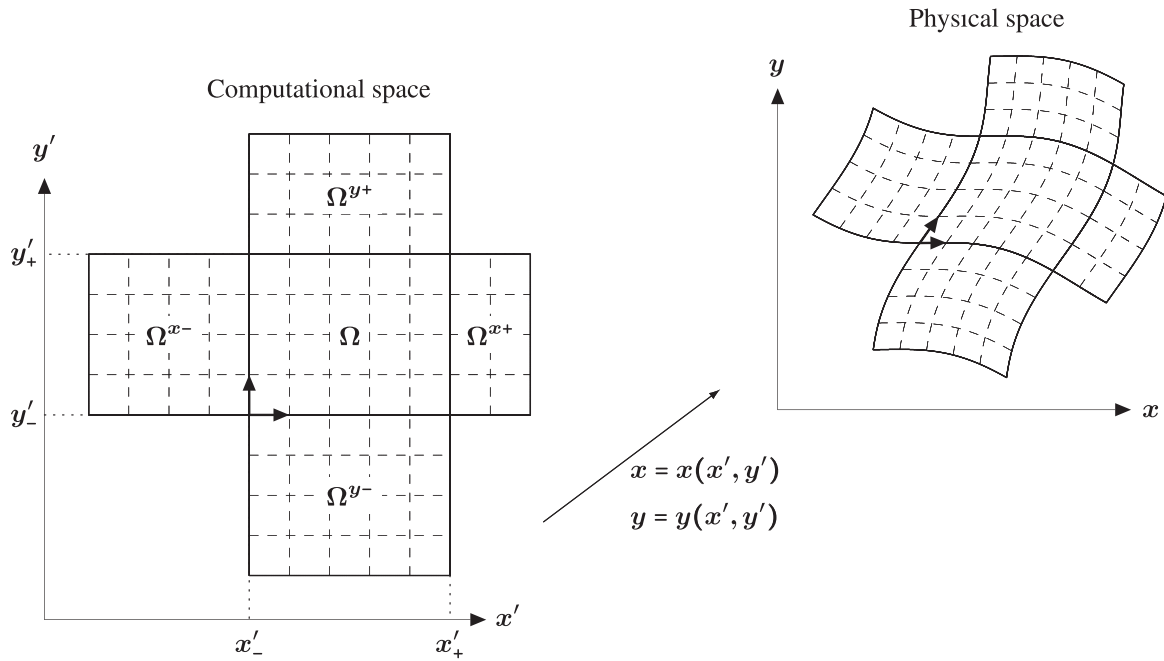


Figure 1. Illustration of the geometrical mapping between the physical space coordinates (x, y) and the computational space coordinates (x', y') .

compare the results obtained with DFDM to those obtained using SEM. DFDM is significantly more accurate than SEM when using a similar setup. We finish with concluding remarks and future perspectives.

2 METHODOLOGY

2.1 Governing equations

We consider wave propagation in a 2-D linear anisotropic elastic medium. The proposed algorithm relies on the second-order elastodynamic system of equations.

$$\rho \ddot{u}_x = \partial_x \sigma_{xx} + \partial_y \sigma_{xy} + f_x \quad (1a)$$

$$\rho \ddot{u}_y = \partial_x \sigma_{xy} + \partial_y \sigma_{yy} + f_y \quad (1b)$$

$$\sigma_{xx} = \mathcal{C}_{11} \partial_x u_x + \mathcal{C}_{12} \partial_y u_y + \mathcal{C}_{13} (\partial_x u_y + \partial_y u_x) \quad (1c)$$

$$\sigma_{yy} = \mathcal{C}_{12} \partial_x u_x + \mathcal{C}_{22} \partial_y u_y + \mathcal{C}_{23} (\partial_x u_y + \partial_y u_x) \quad (1d)$$

$$\sigma_{xy} = \mathcal{C}_{13} \partial_x u_x + \mathcal{C}_{23} \partial_y u_y + \mathcal{C}_{33} (\partial_x u_y + \partial_y u_x), \quad (1e)$$

where u_i is the displacement component in direction i , f_i is the component of the source in direction i , σ_{ij} are the component of the stress tensor, ρ is the density, $\mathcal{C}_{ij}$ can be inferred from elastic parameters of the linear Hooke law, ∂_t is the partial derivative with respect to time and ∂_i is the partial derivative with respect to spatial direction i .

2.2 Curvilinear transformation

To allow for curvilinear meshes that follow the topography of the interfaces, we apply a smooth, locally invertible coordinate transformation

$$x = x(x', y') \quad y = y(x', y'), \quad (2)$$

where x and y are the Cartesian coordinates in the physical space and x' and y' are the coordinates in the computational space. We illustrate such a transformation in Fig. 1. Using the transformation in eq. (2), we can write eq. (1) in the conservative form (Thompson *et al.* 1982; Appelö & Petersson 2009; Dovgilevich & Sofronov 2015)

$$|\mathbf{J}| \begin{bmatrix} \rho & 0 \\ 0 & \rho \end{bmatrix} \begin{bmatrix} \ddot{u}_x \\ \ddot{u}_y \end{bmatrix} = \begin{bmatrix} \frac{\partial}{\partial x'} & \frac{\partial}{\partial y'} & 0 & 0 \\ 0 & 0 & \frac{\partial}{\partial x'} & \frac{\partial}{\partial y'} \end{bmatrix} |\mathbf{J}| \begin{bmatrix} \frac{\partial x'}{\partial x} & 0 & \frac{\partial x'}{\partial y} \\ \frac{\partial y'}{\partial x} & 0 & \frac{\partial y'}{\partial y} \\ 0 & \frac{\partial x'}{\partial y} & \frac{\partial x'}{\partial x} \\ 0 & \frac{\partial y'}{\partial y} & \frac{\partial y'}{\partial x} \end{bmatrix} \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{bmatrix} + |\mathbf{J}| \begin{bmatrix} f_x \\ f_y \end{bmatrix} \quad (3a)$$

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{12} & C_{22} & C_{23} \\ C_{13} & C_{23} & C_{33} \end{bmatrix} \begin{bmatrix} \frac{\partial x'}{\partial x} & \frac{\partial y'}{\partial x} & 0 & 0 \\ 0 & 0 & \frac{\partial x'}{\partial y} & \frac{\partial y'}{\partial y} \\ \frac{\partial x'}{\partial y} & \frac{\partial y'}{\partial y} & \frac{\partial x'}{\partial x} & \frac{\partial y'}{\partial x} \end{bmatrix} \begin{bmatrix} \frac{\partial}{\partial x} & 0 \\ \frac{\partial}{\partial y} & 0 \\ 0 & \frac{\partial}{\partial x'} \\ 0 & \frac{\partial}{\partial y'} \end{bmatrix} \begin{bmatrix} u_x \\ u_y \end{bmatrix}, \quad (3b)$$

where

$$\mathbf{J}(x', y') = \begin{bmatrix} \frac{\partial x}{\partial x'} & \frac{\partial x}{\partial y'} \\ \frac{\partial y}{\partial x'} & \frac{\partial y}{\partial y'} \end{bmatrix} \mathbf{J}^{-1}(x', y') = \begin{bmatrix} \frac{\partial x'}{\partial x} & \frac{\partial x'}{\partial y} \\ \frac{\partial y'}{\partial x} & \frac{\partial y'}{\partial y} \end{bmatrix} |\mathbf{J}| = \frac{\partial x}{\partial x'} \frac{\partial y}{\partial y'} - \frac{\partial y}{\partial x'} \frac{\partial x}{\partial y'} \quad (3c)$$

and $|\mathbf{J}|$ is the determinant of the Jacobian matrix $\mathbf{J}$. We need to transform the linear system above into a discrete linear system to obtain a numerical solution.

2.3 Discrete representation of the field variables

We discretize the field variables in eq. (3) following the approach introduced in Masson (2022) and Masson & Virieux (2023). We represent the wavefield using orthogonal basis functions that are staggered according to the so-called Lebedev grid for the following reasons:

- (i) Orthonormal bases maintain the **self-adjoint antisymmetric** nature of the wave equation after discretization (Masson 2022). This is essential to achieve numerical reciprocity and ensure the stability of the numerical scheme (subject to a CFL condition).
- (ii) Staggered basis functions ensure that the spurious(non-physical) **parasitic modes** associated with the centred finite-difference operators are effectively removed (see, e.g. New *et al.* 1998).

In two dimensions, the Lebedev grid consists of two Virieux grids (four in three dimensions) as pictured in Fig. 2. The field variables ($u_x, u_y, \sigma_{xx}, \sigma_{yy}, \sigma_{xy}$) are represented accordingly using two sets of variables ($u_x^{21}, u_y^{12}, \sigma_{xx}^{11}, \sigma_{yy}^{11}, \sigma_{xy}^{22}$) and ($u_x^{12}, u_y^{21}, \sigma_{xx}^{22}, \sigma_{yy}^{22}, \sigma_{xy}^{11}$) with discrete representations ($\mathbf{U}_x^{21}, \mathbf{U}_y^{12}, \mathbf{S}_{xx}^{11}, \mathbf{S}_{yy}^{11}, \mathbf{S}_{xy}^{22}$) and ($\mathbf{U}_x^{12}, \mathbf{U}_y^{21}, \mathbf{S}_{xx}^{22}, \mathbf{S}_{yy}^{22}, \mathbf{S}_{xy}^{11}$). Here, the upper indices indicate the basis in which the field variable are represented in the two spatial directions. For example, u^{12} is represented using the basis B^1 in direction x and the basis B^2 in direction y . Each set of variables is organized following the staggered grid (Virieux 1986). For the first Virieux grid, we have

$$u_x^{21}(x', y', t) = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^1} \mathbf{U}_{xij}^{21}(t) \hat{B}_{ij}^{21}(x', y') = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^1} \mathbf{U}_{xij}^{21}(t) \hat{B}_{xi}^2(x') \hat{B}_{yj}^1(y') \quad (4a)$$

$$u_y^{12}(x', y', t) = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^2} \mathbf{U}_{yij}^{12}(t) \hat{B}_{ij}^{12}(x', y') = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^2} \mathbf{U}_{yij}^{12}(t) \hat{B}_{xi}^1(x') \hat{B}_{yj}^2(y') \quad (4b)$$

$$\sigma_{xx}^{11}(x', y', t) = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^1} \mathbf{S}_{xij}^{11}(t) \hat{B}_{ij}^{11}(x', y') = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^1} \mathbf{S}_{xij}^{11}(t) \hat{B}_{xi}^1(x') \hat{B}_{yj}^1(y') \quad (4c)$$

$$\sigma_{yy}^{11}(x', y', t) = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^1} \mathbf{S}_{yij}^{11}(t) \hat{B}_{ij}^{11}(x', y') = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^1} \mathbf{S}_{yij}^{11}(t) \hat{B}_{xi}^1(x') \hat{B}_{yj}^1(y') \quad (4d)$$

$$\sigma_{xy}^{22}(x', y', t) = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^2} \mathbf{S}_{xyij}^{22}(t) \hat{B}_{ij}^{22}(x', y') = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^2} \mathbf{S}_{xyij}^{22}(t) \hat{B}_{xi}^2(x') \hat{B}_{yj}^2(y') \quad (4e)$$

and for the second Virieux grid we have

$$u_x^{12}(x', y', t) = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^2} \mathbf{U}_{xij}^{12}(t) \hat{B}_{ij}^{12}(x', y') = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^2} \mathbf{U}_{xij}^{12}(t) \hat{B}_{xi}^1(x') \hat{B}_{yj}^2(y') \quad (5a)$$

$$u_y^{21}(x', y', t) = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^1} \mathbf{U}_{yij}^{21}(t) \hat{B}_{ij}^{21}(x', y') = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^1} \mathbf{U}_{yij}^{21}(t) \hat{B}_{xi}^2(x') \hat{B}_{yj}^1(y') \quad (5b)$$

$$\sigma_{xx}^{22}(x', y', t) = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^2} \mathbf{S}_{xij}^{22}(t) \hat{B}_{ij}^{22}(x', y') = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^2} \mathbf{S}_{xij}^{22}(t) \hat{B}_{xi}^2(x') \hat{B}_{yj}^2(y') \quad (5c)$$

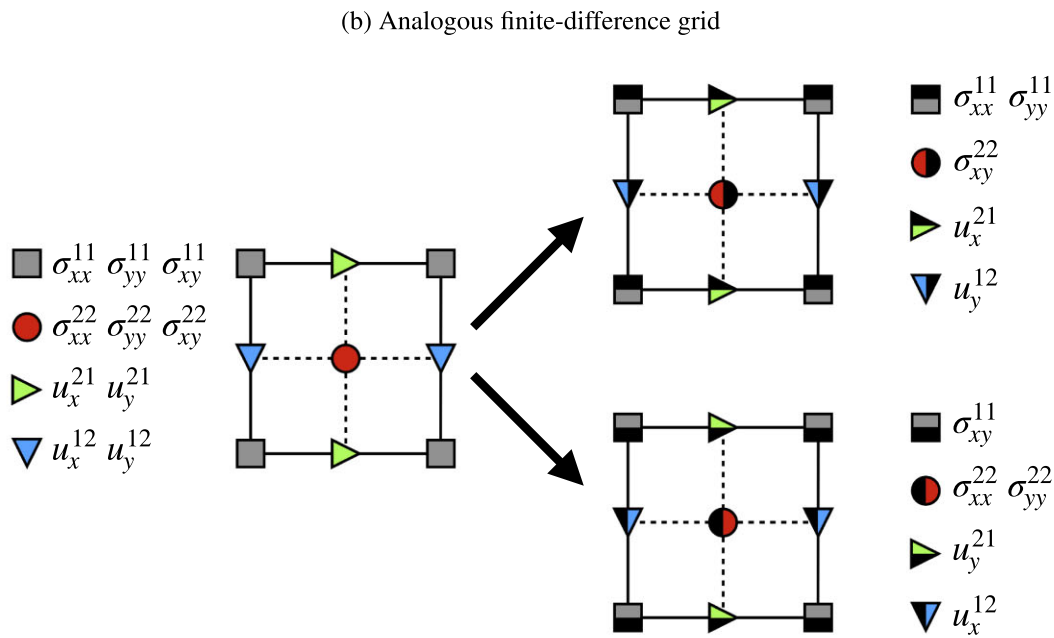
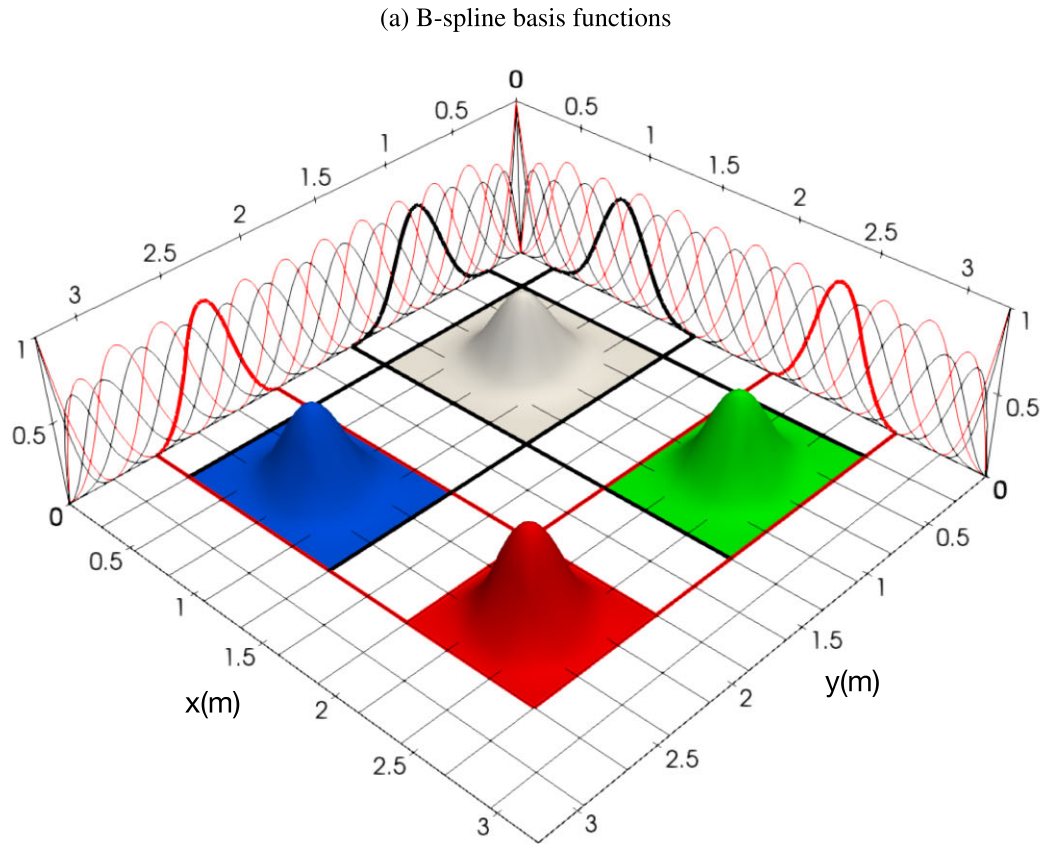


Figure 2. (a) Representation of the canonical B-spline bases used to construct the orthonormal bases that represent the field variables in the proposed DFD algorithm. The stresses σ_{xx}^{11} , σ_{yy}^{11} and σ_{xy}^{11} are represented using the basis pictured in grey. The stresses σ_{xx}^{22} , σ_{yy}^{22} and σ_{xy}^{22} are represented using the basis pictured in red. The displacements u_x^{12} and u_y^{12} are represented using the basis pictured in blue. The displacements u_x^{21} and u_y^{21} are represented using the basis pictured in green. (b) Representation of the analogous finite-difference staggered grid. The complete grid to the left is called the Lebedev grid. It decomposes into the two staggered grids to the right [i.e. as in Virieux (1986)].

$$\sigma_{yy}^{22}(x', y', t) = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^2} \mathbf{S}_{yyij}^{22}(t) \hat{B}_{ij}^{22}(x', y') = \sum_{i=1}^{N_x^2} \sum_{j=1}^{N_y^2} \mathbf{S}_{yyij}^{22}(t) \hat{B}_{xi}^2(x') \hat{B}_{yj}^2(y') \quad (5d)$$

$$\sigma_{xy}^{11}(x', y', t) = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^1} \mathbf{S}_{xyij}^{11}(t) \hat{B}_{ij}^{11}(x', y') = \sum_{i=1}^{N_x^1} \sum_{j=1}^{N_y^1} \mathbf{S}_{xyij}^{11}(t) \hat{B}_{xi}^1(x') \hat{B}_{yj}^1(y'), \quad (5e)$$

where the $\hat{B}_{ij}^{kl}$ ($k = 1, 2$) are the 2-D orthonormal B-spline basis functions. They are constructed from the 1-D basis functions $\hat{B}_{xi}^k$ ($k = 1, 2$) and $\hat{B}_{yj}^l$ ($l = 1, 2$) using a tensorial product. The orthonormal bases $\hat{B}_{ij}^{kl}$ ($k = 1, 2$) may be constructed from arbitrary basis functions B_{ij}^{kl} ($k = 1, 2$) that we will refer to as the canonical bases. In this study, the canonical bases are taken as the 2-D B-spline bases detailed in Appendix A and pictured in Fig. 2.

2.4 Approximation using dual bases

Before constructing the DFD operators, we show that the orthonormal bases used for representing the field variables can be used to approximate functions and their derivatives. The orthonormal bases $\hat{B}^k$ ($k = 1, 2$) defined in Appendix C are self-dual. Thus, they provide us with a direct way to find the approximation $f^k(\theta)$ of an arbitrary function $f(\theta)$, where θ is a spatial coordinate (see, e.g. Woźny 2013). We have

$$f^k(\theta) = \sum_{i=1}^{N^k} \mathbf{f}_i^k \hat{B}_i^k(\theta) \approx f(\theta), \quad (6)$$

where the approximation $f^k(\theta)$ is represented in the basis $\hat{B}^k$ and the expansion coefficients $\mathbf{f}_i^k$ are obtained directly using the integrals

$$\mathbf{f}_i^k = \int_{\theta_-}^{\theta_+} \hat{B}_i^k(\theta) f(\theta) d\theta. \quad (7)$$

In eq. (6), the function $f^k(\theta)$ is an optimal approximation of $f(\theta)$ in the least square sense; it minimizes the norm $\|f - f^k\|$ in the interval $[\theta_-, \theta_+]$. Notably, the approximation error vanishes and we obtain an exact result (i.e. $\|f - f^k\| = 0$) when the function $f(\theta) = f^k(\theta)$ can be represented in the basis $\hat{B}^k$. We can approximate the derivative $f'(\theta)$ of the function $f(\theta)$ with a similar approach. We have

$$f'^k(\theta) = \sum_{i=1}^{N^k} \mathbf{f}_i'^k \hat{B}_i^k(\theta) \approx f'(\theta), \quad (8)$$

where $\mathbf{f}_i'^k$ are the expansion coefficients representing the approximation $f'^k(\theta)$ of $f'(\theta)$ in the basis $\hat{B}^k$. Let us note that this basis for the derivative might not be the one used for approximating the function itself. By substituting $f(\theta)$ with $f'(\theta)$ in eq. (7) and using integration by parts, we obtain the expansion coefficients

$$\mathbf{f}_i'^k = \int_{\theta_-}^{\theta_+} \hat{B}_{\theta i}^k(\theta) f'(\theta) d\theta \quad (9a)$$

$$= - \int_{\theta_-}^{\theta_+} (\hat{B}_{\theta i}^k)'(\theta) f(\theta) d\theta - \hat{B}_{\theta i}^k(\theta_-) f(\theta_-) + \hat{B}_{\theta i}^k(\theta_+) f(\theta_+), \quad (9b)$$

where $(\hat{B})'$ are the derivatives of $\hat{B}$. eqs (7) and (9) are used to approximate the derivatives of the field variables and thus construct the DFD operators. Eqs (7) and (9) are central in DFDM; they allow one to compute the value/derivative of a function in a basis different from the one in which they are represented. It is similar to what is done in staggered-grid FDM, where we take wavefield values on one grid and compute their derivative on the corresponding shifted grid. This is the key point regarding the staggered-grid strategy we promote.

2.5 Distributional finite-difference operators in 1-D

We now construct the 1-D DFD operators that are the basic bricks of our algorithm. Following (Masson & Virieux 2023), they should act on the field variables expressed in the first basis and return their derivatives' representations in the second basis. We assimilate the function $f(\theta)$ introduced in the previous section to a discrete field variable $f^l(\theta)$ represented in the orthogonal basis $\hat{B}^l$. We assume that the discrete representation $\mathbf{f}^l$ of $f^l(\theta)$ is known. We can express $f^l(\theta)$ and its derivative $f'^l(\theta)$ as

$$f^l(\theta) = \sum_{j=1}^{N^l} \mathbf{f}_j^l \hat{B}_j^l(\theta) \quad (10a)$$

$$f'^l(\theta) = \sum_{j=1}^{N^l} \mathbf{f}_j'^l (\hat{B}_j^l)'(\theta), \quad (10b)$$

where $\mathbf{f}_j^l$ are the expansion coefficients stored in the vector $\mathbf{f}^l$, $\hat{B}_j^l$ are the basis functions, and $(\hat{B}_j^l)'$ are their derivatives. Keeping in mind that the upper indices k and l indicate the basis in which the variables are represented; our objective is to obtain the approximations $f^k(\theta) \approx f^l(\theta)$ and $f'^k(\theta) \approx f'^l(\theta)$ expressed in $\hat{B}^k$ of the function and its derivative in eq. (10). By substituting $\hat{B}_i^k(\theta)$ and $f^l(\theta)$ in eq. (7) with their expressions in eqs (C2) and (10a) we obtain the linear system

$$\mathbf{f}^k = \mathbf{T}^{kl} \cdot \mathbf{f}^l, \quad (11)$$

where

$$\mathbf{T}^{\mathbf{kl}} = (\mathbf{L}^{\mathbf{k}})^{-1} \cdot \mathbf{M}^{\mathbf{kl}} \cdot (\mathbf{L}^{\mathbf{l}^T})^{-1}. \quad (12)$$

The mass matrix $\mathbf{M}^{\mathbf{kl}}$ and its Cholesky factorization $\mathbf{L}^{\mathbf{n}}$ are defined in Appendices B and C. The operator $\mathbf{T}^{\mathbf{kl}}$ can be thought of as a basis transform; it acts on the vector $\mathbf{f}^{\mathbf{l}}$ representing the function $f^{\mathbf{l}}(\theta)$ in the basis $\hat{B}^{\mathbf{l}}$ and returns the vector $\mathbf{f}^{\mathbf{k}}$ representing the approximation $f^{\mathbf{k}}(\theta) \approx f^{\mathbf{l}}(\theta)$ in the basis $\hat{B}^{\mathbf{k}}$.

By substituting $\hat{B}_i^{\mathbf{k}}(\theta)$ and $f^{\mathbf{l}}(\theta)$ in eq. (9) with their expressions in eqs (C2) and (10b) we obtain the linear system associated with the derivative

$$\mathbf{f}^{\mathbf{k}} = \mathbf{D}^{\mathbf{kl}} \cdot \mathbf{f}^{\mathbf{l}} - \mathbf{b}^{\mathbf{k}} f^{\mathbf{l}}(\theta_-) + \mathbf{d}^{\mathbf{k}} f^{\mathbf{l}}(\theta_+), \quad (13)$$

where

$$\mathbf{D}^{\mathbf{kl}} = -(\mathbf{L}^{\mathbf{k}})^{-1} \cdot \mathbf{K}^{\mathbf{lk}^T} \cdot (\mathbf{L}^{\mathbf{l}^T})^{-1}. \quad (14)$$

and

$$\mathbf{b}^{\mathbf{k}} = (\mathbf{L}^{\mathbf{k}})^{-1} \mathbf{p}^{\mathbf{k}} \quad (15a)$$

$$\mathbf{d}^{\mathbf{k}} = (\mathbf{L}^{\mathbf{k}})^{-1} \mathbf{q}^{\mathbf{k}}. \quad (15b)$$

The stiffness matrix $\mathbf{K}^{\mathbf{lk}}$ and the vectors $\mathbf{p}^{\mathbf{k}}$ and $\mathbf{q}^{\mathbf{k}}$ are defined in Appendix B. When the boundary values $f^{\mathbf{l}}(\theta_-)$ and $f^{\mathbf{l}}(\theta_+)$ are equal to the first and last elements of the vector $\mathbf{f}^{\mathbf{l}}$, the linear system in eq. (13) becomes

$$\mathbf{f}^{\mathbf{k}} = \mathbb{D}^{\mathbf{kl}} \cdot \mathbf{f}^{\mathbf{l}}, \quad (16)$$

and by definition, the first-order distributional finite-difference operator is

$$\mathbb{D}^{\mathbf{kl}} = (\mathbf{L}^{\mathbf{k}})^{-1} \cdot \left[-\mathbf{K}^{\mathbf{lk}^T} - \mathbf{p}^{\mathbf{k}} \cdot \mathbf{p}^{\mathbf{l}^T} + \mathbf{q}^{\mathbf{k}} \cdot \mathbf{q}^{\mathbf{l}^T} \right] \cdot (\mathbf{L}^{\mathbf{l}^T})^{-1}. \quad (17)$$

The operator $\mathbb{D}^{\mathbf{kl}}$ acts on the vector $\mathbf{f}^{\mathbf{l}}$ representing the function $f^{\mathbf{l}}(\theta)$ in the basis $\hat{B}^{\mathbf{l}}$. It returns the vector $\mathbf{f}^{\mathbf{k}}$ representing the approximation $f^{\mathbf{k}}(\theta) \approx f^{\mathbf{l}}(\theta)$ in the basis $\hat{B}^{\mathbf{k}}$.

Note that the factorization used here is different (but equivalent) to that used in Masson (2022) where the DFD operators have an adjoint form and partly encompass the boundary values. In this paper, the self-adjoint form of the global or assembled linear system is recovered by using an appropriate averaging of the boundary values between neighbouring elements, as detailed in the next section. Note that all the matrices involved in calculating the DFD operators in eqs (12) and (14) are **band-diagonal matrices** because the B-spline canonical basis functions have compact support. It ensures the efficiency of the proposed algorithm.

2.6 Distributional finite-difference operators in 2-D

We construct the 2-D-DFD operators needed to compute the partial derivative of a function using a tensor product, that is by applying the 1-D-DFD operators in the spatial dimensions x' and y' as detailed in Masson (2022).

Let $f^{kl}(x', y')$ be one of the field variables in eqs (4) and (5). It is represented in the basis $\hat{B}^{kl}$ through its expansion coefficients $\mathbf{F}_{mn}^{\mathbf{kl}}$ stored in the matrix $\mathbf{F}^{\mathbf{kl}}$. Throughout this paper, we assume that the indices (i, j) of the arrays correspond to (row, column) and are associated with directions (x, y) . We have

$$f^{kl}(x', y') = \sum_{m=1}^{N_x^k} \sum_{n=1}^{N_y^l} \mathbf{F}_{mn}^{\mathbf{kl}} \hat{B}_{mn}^{kl}(x', y') = \sum_{m=1}^{N_x^k} \sum_{n=1}^{N_y^l} \mathbf{F}_{mn}^{\mathbf{kl}} \hat{B}_{xm}^k(x') \hat{B}_{yn}^l(y'), \quad (18)$$

where the 2-D basis functions $\hat{B}_{mn}^{kl}(x', y') = \hat{B}_{xm}^k(x') \hat{B}_{yn}^l(y')$ are constructed from the 1-D basis functions $\hat{B}_{xm}^k(x')$ and $\hat{B}_{yn}^l(y')$. According to the staggered grid, one should represent the partial derivative $\frac{\partial}{\partial x} f^{kl}(x', y')$ in the basis $\hat{B}^{il}$ (with $l \in [1, 2]$ and $i = 3 - l$) and the partial derivative $\frac{\partial}{\partial y} f^{kl}(x', y')$ in the basis $\hat{B}^{kj}$ (with $k \in [1, 2]$ and $j = 3 - k$). We have

$$\frac{\partial}{\partial x} f^{kl}(x', y') \approx \sum_{m=1}^{N_x^i} \sum_{n=1}^{N_y^l} \mathbb{D}_{xmn}^{il} \hat{B}_{mn}^{il}(x', y') = \sum_{m=1}^{N_x^i} \sum_{n=1}^{N_y^l} \mathbb{D}_{xmn}^{il} \hat{B}_{xm}^i(x') \hat{B}_{yn}^l(y') \quad (19)$$

$$\frac{\partial}{\partial y} f^{kl}(x', y') \approx \sum_{m=1}^{N_x^k} \sum_{n=1}^{N_y^j} \mathbb{D}_{ymn}^{kj} \hat{B}_{mn}^{kj}(x', y') = \sum_{m=1}^{N_x^k} \sum_{n=1}^{N_y^j} \mathbb{D}_{ymn}^{kj} \hat{B}_{xm}^k(x') \hat{B}_{yn}^j(y'), \quad (20)$$

where the expansion coefficients $\mathbb{D}_{xmn}^{il}$ and $\mathbb{D}_{ymn}^{kj}$ stored in the matrices $\mathbb{D}_x^{il}$ and $\mathbb{D}_y^{kj}$ represent the approximations $\frac{\partial}{\partial x} f^{kl}(x', y')$ and $\frac{\partial}{\partial y} f^{kl}(x', y')$ in the bases $\hat{B}^{il}$ and $\hat{B}^{kj}$. Note that the function $f^{kl}(x', y')$ and its partial derivatives are represented in the same 1-D basis perpendicular to the partial derivative's direction. Following the approach in Masson (2022), the expansion coefficients representing the partial derivatives can be computed using

$$\mathbb{D}_{\mathbf{x}}^{ij}(\mathbf{F}^{kj}) = \mathbf{D}_{\mathbf{x}}^{ik} \cdot \mathbf{F}^{kj} - \mathbf{b}_{\mathbf{x}}^i \cdot \mathbf{f}^T(x'_-) + \mathbf{d}_{\mathbf{x}}^i \cdot \mathbf{f}^T(x'_+) \quad (21a)$$

$$\mathbb{D}_{\mathbf{y}}^{ij}(\mathbf{F}^{il}) = \mathbf{F}^{il} \cdot \mathbf{D}_{\mathbf{y}}^{jT} - \mathbf{f}^i(y'_-) \cdot \mathbf{b}_{\mathbf{y}}^T + \mathbf{f}^i(y'_+) \cdot \mathbf{d}_{\mathbf{y}}^T, \quad (21b)$$

where

(i) $\mathbf{F}^{kl}$ contains the internal expansion coefficients representing the function $f^{kl}(x', y')$ inside the domain Ω . The indices $\mathbf{kl}$ indicates the basis B^{kl} in which $f^{kl}(x', y')$ is represented inside Ω .

(ii) $\mathbf{f}^i(x'_{\pm})$ contains the expansion coefficients representing the boundary value $f^{kl}(x'_{\pm}, y')$ along the faces separating the domains $\Omega^{\pm}$ and Ω in Fig. 1. The index $\mathbf{i}$ indicates the basis $B_{\mathbf{y}}^i$ in which $f^{kl}(x'_{\pm}, y')$ is represented inside Ω .

(iii) $\mathbf{f}^i(y'_{\pm})$ contains the expansion coefficients representing the boundary value $f^{kl}(x', y'_{\pm})$ along the faces separating the domains $\Omega^{\pm}$ and Ω in Fig. 1. The index $\mathbf{j}$ indicates the basis $B_{\mathbf{x}}^j$ in which $f^{kl}(x', y'_{\pm})$ is represented inside Ω .

(iv) $\mathbf{D}_{\theta}^{ij}$, $\mathbf{b}_{\theta}^i$ are the 1D-DFD operators introduced in Section 2.5.

In the domain Ω , the internal boundary values in eq. (21) shall be set to impose the boundary conditions or computed from the internal expansion coefficients $\mathbf{F}^{ij}$ using

$$\mathbf{f}^j(x'_-) = [\mathbf{f}^j(x'_-)]_{\Omega} = [\mathbf{F}^{kjT} \cdot \mathbf{b}_{\mathbf{x}}^k]_{\Omega} \quad (22a)$$

$$\mathbf{f}^j(x'_+) = [\mathbf{f}^j(x'_+)]_{\Omega} = [\mathbf{F}^{kjT} \cdot \mathbf{d}_{\mathbf{x}}^k]_{\Omega} \quad (22b)$$

$$\mathbf{f}^i(y'_-) = [\mathbf{f}^i(y'_-)]_{\Omega} = [\mathbf{b}_{\mathbf{x}}^{iT} \cdot \mathbf{F}^{il}]_{\Omega} \quad (22c)$$

$$\mathbf{f}^i(y'_+) = [\mathbf{f}^i(y'_+)]_{\Omega} = [\mathbf{d}_{\mathbf{x}}^{iT} \cdot \mathbf{F}^{il}]_{\Omega}. \quad (22d)$$

Here, the brackets $[\]_{\Omega}$ indicate that the expansion coefficients are taken inside the current domain Ω . To model a free surface, for example, one can take boundary values of the traction equal to zero when evaluating the partial derivatives of the stresses and use the internal values when evaluating the partial derivatives of the displacements. Similarly, for a domain with fixed boundaries, one can take boundary values of displacement equal to zero when evaluating the partial derivatives of displacements and use the internal values in eq. (22) when evaluating the partial derivatives of the stresses.

We average the boundary values between neighbouring domains to connect multiple domains or elements. This continuity condition is needed to ensure numerical reciprocity and the stability of the numerical scheme (Masson 2022). Note however, that the continuity of the wavefield is not satisfied numerically and the values to the left and right of an interface may be different. The partial derivatives in eq. (21) are evaluated in the domain Ω using

$$\mathbf{f}^j(x'_-) = (\alpha_{x'_-}) [\mathbf{f}^j(x'_-)]_{\Omega} + (1 - \alpha_{x'_-}) \mathbf{T}_{\mathbf{y}}^{ij} \cdot [\mathbf{f}^j(x'_+)]_{\Omega^{x'_-}} \quad (23a)$$

$$\mathbf{f}^j(x'_+) = (\alpha_{x'_+}) [\mathbf{f}^j(x'_+)]_{\Omega} + (1 - \alpha_{x'_+}) \mathbf{T}_{\mathbf{y}}^{ij} \cdot [\mathbf{f}^j(x'_-)]_{\Omega^{x'_+}} \quad (23b)$$

$$\mathbf{f}^i(y'_-) = (\alpha_{y'_-}) [\mathbf{f}^i(y'_-)]_{\Omega} + (1 - \alpha_{y'_-}) \mathbf{T}_{\mathbf{x}}^{ji} \cdot [\mathbf{f}^i(y'_+)]_{\Omega^{y'_-}} \quad (23c)$$

$$\mathbf{f}^i(y'_+) = (\alpha_{y'_+}) [\mathbf{f}^i(y'_+)]_{\Omega} + (1 - \alpha_{y'_+}) \mathbf{T}_{\mathbf{x}}^{ji} \cdot [\mathbf{f}^i(y'_-)]_{\Omega^{y'_+}} \quad (23d)$$

Here, the brackets' subscripts indicate the domain from which the boundary values are taken. $[\]_{\Omega}$ corresponds to the current domain, and $[\]_{\Omega^{x'_-}}$, $[\]_{\Omega^{x'_+}}$, $[\]_{\Omega^{y'_-}}$ and $[\]_{\Omega^{y'_+}}$ correspond to the neighbouring domains to the left, right, bottom and top, respectively, as pictured in Fig. 1. The weighted average coefficients $\alpha_{x'_-}$, $\alpha_{x'_+}$, $\alpha_{y'_-}$ and $\alpha_{y'_+}$ are associated with the left, right, bottom and top faces of the domain Ω , respectively. The matrices $\mathbf{T}_{\theta}^{nn}$ in eq. (23) are associated with the faces. They transform the boundary values taken from the neighbouring domains Ω^n and return their representation in the basis of the current domain Ω . This operation can be thought of as an approximate basis transform. We have

$$\mathbf{T}_{\theta}^{nn} = [\mathbf{L}_{\theta}^n]_{\Omega}^{-1} \cdot \mathbf{M}_{\theta}^{nn} \cdot [\mathbf{L}_{\theta}^n]_{\Omega^n}^{-1} \quad (\theta = x, y; \mathbf{n} = \mathbf{i}, \mathbf{j}) \quad (24a)$$

$$\mathbf{M}_{\theta}^{nn} \stackrel{\text{def}}{=} (\mathbf{M}_{\theta}^{nn})_{ij} = \int_{\theta_-}^{\theta_+} [B_{\theta i}^n(\theta)]_{\Omega} [B_{\theta j}^n(\theta)]_{\Omega^n} d\theta \quad (\theta = x, y; \mathbf{n} = \mathbf{i}, \mathbf{j}) \quad (24b)$$

where the quantities within the square brackets $[\]_{\Omega}$ and $[\]_{\Omega^n}$ are those associated with the current Ω and the neighbouring Ω^n domains. The matrices $\mathbf{L}_{\theta}^n$ are defined in eq. (C1). The matrix $\mathbf{T}_{\theta}^{nn}$ in eq. (24a) is similar to the zeroth-order DFD operator in eq. (12) but acts on the boundary values taken from the neighbouring domains $[\]_{\Omega^n}$. Accordingly, the matrix $\mathbf{M}_{\theta}^{nn}$ in eq. (24b) is similar to the mass matrix in eq. (B1) but the second basis functions $B_{\theta j}^n(\theta)$ are taken from the neighbouring domain Ω^n . When the wavefield is represented in the same basis along the interface between two elements (i.e. for conformal interfaces), the matrices $\mathbf{T}_{\theta}^{nn}$ in eq. (23) may be omitted because $\mathbf{T}_{\theta}^{nn} = \mathbf{I}$.

To obtain a stable algorithm that satisfies numerical reciprocity, the boundary values must be uniquely defined between neighbouring domains (Masson 2022). We must apply the same weighting scheme in neighbouring elements. For example, one enforces the following relations between the five domains pictured in Fig. 1:

$$[\alpha_{x'_-}]_{\Omega} = 1 - [\alpha_{x'_+}]_{\Omega_{x'_-}} \quad (25)$$

$$[\alpha_{x'_+}]_{\Omega} = 1 - [\alpha_{x'_-}]_{\Omega_{x'_+}} \quad (26)$$

$$[\alpha_{y'_-}]_{\Omega} = 1 - [\alpha_{y'_+}]_{\Omega_{y'_-}} \quad (27)$$

$$[\alpha_{y'_+}]_{\Omega} = 1 - [\alpha_{y'_-}]_{\Omega_{y'_+}}. \quad (28)$$

Here $[\alpha_a]_{\Omega}$ denotes the weight associated with the face a of the domain Ω^a . The continuity conditions in eq. (25) ensure that the global differential operators (that approximate the partial derivatives in all domains taken together) will be adjoint after applying the boundary conditions (Masson 2022). Thus far, we derived the DFD operators approximating the partial derivatives in eq. (21) independently of the wave equation. We can use these operators as such to solve the elastodynamic system in eqs (3).

2.7 2-D algorithm

Our algorithm is obtained by substituting the partial derivatives in eq. (3) with the DFD operators in eq. (21) and by substituting the physical quantities by their discrete counterparts. In this work, we adopt the virtual point approach introduced in Masson & Virieux (2023). The material properties and quantities associated with the curvilinear mapping are simply evaluated at the virtual gridpoints, as detailed in Appendix D. We approximate the second-time derivative using a centred second-order finite-difference scheme. Thus, eq. (3) are solved using a recursive procedure. For clarity, we give the update expressions for the variables associated with the first Virieux grid in eq. (4). The update expressions for the second Virieux grid in eq. (5) are similar and are obtained by performing the index substitutions ($1 \rightarrow 2$; $2 \rightarrow 1$) in eqs (29)–(31) below. At each step, one first evaluates the stresses in each domain at time t using:

$$\Sigma_{xx}^{11} = \left[(C_{11}^{11} \circ J_{xx}^{-111} + C_{13}^{11} \circ J_{xy}^{-111}) \circ \mathbb{D}_x^{11}(\mathbf{U}_x^{21}) \right](t) \quad (29a)$$

$$+ \left[(C_{11}^{11} \circ J_{yx}^{-111} + C_{13}^{11} \circ J_{yy}^{-111}) \circ \mathbb{D}_y^{11}(\mathbf{U}_x^{12}) \right](t) \quad (29b)$$

$$+ \left[(C_{12}^{11} \circ J_{xy}^{-111} + C_{13}^{11} \circ J_{xx}^{-111}) \circ \mathbb{D}_x^{11}(\mathbf{U}_y^{21}) \right](t) \quad (29c)$$

$$+ \left[(C_{12}^{11} \circ J_{yy}^{-111} + C_{13}^{11} \circ J_{yx}^{-111}) \circ \mathbb{D}_y^{11}(\mathbf{U}_y^{12}) \right](t) \quad (29d)$$

$$+ \mathcal{M}_{xx}^{11}(t) \quad (29e)$$

$$\Sigma_{yy}^{11} = \left[(C_{12}^{11} \circ J_{xx}^{-111} + C_{23}^{11} \circ J_{xy}^{-111}) \circ \mathbb{D}_x^{11}(\mathbf{U}_x^{21}) \right](t) \quad (29f)$$

$$+ \left[(C_{12}^{11} \circ J_{yx}^{-111} + C_{23}^{11} \circ J_{yy}^{-111}) \circ \mathbb{D}_y^{11}(\mathbf{U}_x^{12}) \right](t) \quad (29g)$$

$$+ \left[(C_{22}^{11} \circ J_{xy}^{-111} + C_{23}^{11} \circ J_{xx}^{-111}) \circ \mathbb{D}_x^{11}(\mathbf{U}_y^{21}) \right](t) \quad (29h)$$

$$+ \left[(C_{22}^{11} \circ J_{yy}^{-111} + C_{23}^{11} \circ J_{yx}^{-111}) \circ \mathbb{D}_y^{11}(\mathbf{U}_y^{12}) \right](t) \quad (29i)$$

$$+ \mathcal{M}_{yy}^{11}(t) \quad (29j)$$

$$\Sigma_{xy}^{22} = \left[(C_{13}^{22} \circ J_{xx}^{-122} + C_{33}^{22} \circ J_{xy}^{-122}) \circ \mathbb{D}_x^{22}(\mathbf{U}_x^{21}) \right](t) \quad (29k)$$

$$+ \left[(C_{13}^{22} \circ J_{yx}^{-122} + C_{33}^{22} \circ J_{yy}^{-122}) \circ \mathbb{D}_y^{22}(\mathbf{U}_x^{21}) \right](t) \quad (29l)$$

$$+ \left[(C_{23}^{22} \circ J_{xy}^{-122} + C_{33}^{22} \circ J_{xx}^{-122}) \circ \mathbb{D}_x^{22}(\mathbf{U}_y^{12}) \right](t) \quad (29m)$$

$$+ \left[(C_{23}^{22} \circ J_{yy}^{-122} + C_{33}^{22} \circ J_{yx}^{-122}) \circ \mathbb{D}_y^{22}(\mathbf{U}_y^{21}) \right](t) \quad (29n)$$

$$+ \mathcal{M}_{xy}^{22}(t). \quad (29o)$$

The stresses are then transformed according to the curvilinear mapping using:

$$\mathbf{S}_{xx}^{11} = |\mathbf{J}|^{11} \circ \left[J_{xx}^{-111} \circ \Sigma_{xx}^{11} + J_{xy}^{-111} \circ \Sigma_{xy}^{11} \right](t) \quad (30a)$$

$$\mathbf{S}_{xy}^{22} = |\mathbf{J}|^{22} \circ \left[\mathbf{J}_{yx}^{-122} \circ \Sigma_{xx}^{22} + \mathbf{J}_{yy}^{-122} \circ \Sigma_{xy}^{22} \right](t) \quad (30b)$$

$$\mathbf{S}_{yx}^{22} = |\mathbf{J}|^{22} \circ \left[\mathbf{J}_{xy}^{-122} \circ \Sigma_{yy}^{22} + \mathbf{J}_{xx}^{-122} \circ \Sigma_{xy}^{22} \right](t) \quad (30c)$$

$$\mathbf{S}_{yy}^{11} = |\mathbf{J}|^{11} \circ \left[\mathbf{J}_{yy}^{-111} \circ \Sigma_{yy}^{11} + \mathbf{J}_{yx}^{-111} \circ \Sigma_{xy}^{11} \right](t). \quad (30d)$$

Finally, the displacements are updated to the next time step, at time $t + \Delta_t$, using:

$$\frac{1}{\Delta_t^2} \left[\mathbf{U}_x^{21}(t + \Delta_t) - 2\mathbf{U}_x^{21}(t) + \mathbf{U}_x^{21}(t - \Delta_t) \right] = \left[\left(\mathbb{D}_x^{21}(\mathbf{S}_{xx}^{11}) + \mathbb{D}_y^{21}(\mathbf{S}_{xy}^{22}) \right) \oslash |\mathbf{J}|^{21} \oslash \rho^{21} \right](t) \quad (31a)$$

$$\frac{1}{\Delta_t^2} \left[\mathbf{U}_y^{12}(t + \Delta_t) - 2\mathbf{U}_y^{12}(t) + \mathbf{U}_y^{12}(t - \Delta_t) \right] = \left[\left(\mathbb{D}_x^{12}(\mathbf{S}_{yx}^{22}) + \mathbb{D}_y^{12}(\mathbf{S}_{yy}^{11}) \right) \oslash |\mathbf{J}|^{12} \oslash \rho^{12} \right](t). \quad (31b)$$

Note that the displacements at the two previous time step needs to be stored in memory while the stresses are only used as a intermediate step to update the displacements. The arrays above are as follow:

- (i) Σ_{ij}^{kl} are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values or expansion coefficients representing the stresses $\sigma_{ij}(t)$ in the basis $\hat{B}^{kl}$.
- (ii) $\mathbf{U}_i^{kl}$ are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values or expansion coefficients representing the displacements $u_i(t)$ in the basis $\hat{B}^{kl}$.
- (iii) $\mathbb{D}_x^{kl}(\mathbf{F})$ are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values or expansion coefficients representing the partial derivative $\frac{\partial}{\partial x'}(f^{kl}(x', y'))$ in the basis $\hat{B}^{kl}$.
- (iv) $\mathbb{D}_y^{kl}(\mathbf{F})$ are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values or expansion coefficients representing the partial derivative $\frac{\partial}{\partial y'}(f^{kl}(x', y'))$ in the basis $\hat{B}^{kl}$.
- (v) $\mathcal{M}_{ij}^{kl}(t)$ are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values or expansion coefficients representing the components of the moment tensor point source M_{ij} in the basis $\hat{B}^{kl}$.
- (vi) $\mathcal{C}_{ij}^{kl}$ are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values of the parameters $C_{ij}(\hat{x}_i^k, \hat{y}_j^l)$ evaluated at the virtual gridpoints $(\hat{x}_i^k, \hat{y}_j^l)$ associated with the basis $\hat{B}^{kl}$.
- (vii) $\mathbf{J}_{ij}^{-1kl}$ are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values of the inverse Jacobian matrix's component $(J^{-1})_{ij}(\hat{x}_i^k, \hat{y}_j^l)$ evaluated at the virtual gridpoints $(\hat{x}_i^k, \hat{y}_j^l)$ associated with the basis $\hat{B}^{kl}$.
- (viii) $|\mathbf{J}|^{kl}$ are 2-D arrays with dimensions $(N_x^k \times N_y^l)$ containing the values of the determinant of the Jacobian matrix $|\mathbf{J}|(\hat{x}_i^k, \hat{y}_j^l)$ evaluated at the virtual gridpoints $(\hat{x}_i^k, \hat{y}_j^l)$ associated with the basis $\hat{B}^{kl}$.
- (ix) Δ_t denotes the time step.
- (x) $^\circ$ denotes the element-wise or the Hadamard product.
- (xi) $\oslash$ denotes the element-wise or the Hadamard division.

In eqs (29)–(31), most of the computational burden lies in the evaluation of the partial derivatives $\mathbb{D}_x^{kl}()$ and $\mathbb{D}_y^{kl}()$ using eq. (21). We give detailed expressions for the partial derivatives needed to evaluate eqs (29) and (31) in Appendix E.

In order to obtain a stable scheme, the coefficients α in eq. (23) must be complementary when evaluating the partial derivatives $\mathbb{D}_x^{kl}()$ and $\mathbb{D}_y^{kl}()$ in eqs (29) and (31). If the coefficients $[\alpha_{x'_-}, \alpha_{x'_+}, \alpha_{y'_-}, \alpha_{y'_+}]$ are used to compute the partial derivatives $\mathbb{D}_x^{kl}(\mathbf{U}_i)$ and $\mathbb{D}_y^{kl}(\mathbf{U}_i)$ in eq. (29), then, the complementary coefficients $[1 - \alpha_{x'_-}, 1 - \alpha_{x'_+}, 1 - \alpha_{y'_-}, 1 - \alpha_{y'_+}]$ must be used to compute the partial derivatives $\mathbb{D}_x^{kl}(\mathbf{S}_{ij})$ and $\mathbb{D}_y^{kl}(\mathbf{S}_{ij})$ in eq. (31). This rather simple strategy simply enforces numerical reciprocity and ensures that the discrete wave equation remains self-adjoint (Masson 2022). Note that more elaborated yet related approaches have been introduced in DGM (e.g., Grote *et al.* 2007).

3 VERIFICATION

To evaluate the accuracy of the proposed algorithm, we model 2-D seismic wave propagation in the isotropic 1-D radial ak135 earth model (Kennett *et al.* 1995) using the DFDM algorithm. While our model is isotropic, the numerical tests demonstrate our algorithm's capability to effectively handle anisotropic media. This is because the coefficient representing the curvilinear mapping and the anisotropic moduli are the same, leading to comparable apparent numerical anisotropy (i.e. when assuming undeformed elements). We do not consider a cylindrical symmetry as in Nissen-Meyer *et al.* (2014); nonetheless, such a symmetry can be envisioned and may be relevant for future applications. We compare the results to those obtained using the SEM method. We present the setups for our DFDM and the SEM benchmark simulations and then discuss our results.

3.1 Example setup

We use a simplified isotropic version of the ak135 earth model with no attenuation and a single-layer crust. Inside the crustal layer, we use a linear velocity profile. It interpolates the values of the elastic properties of the ak135 model from the surface to the base of the lower crust.

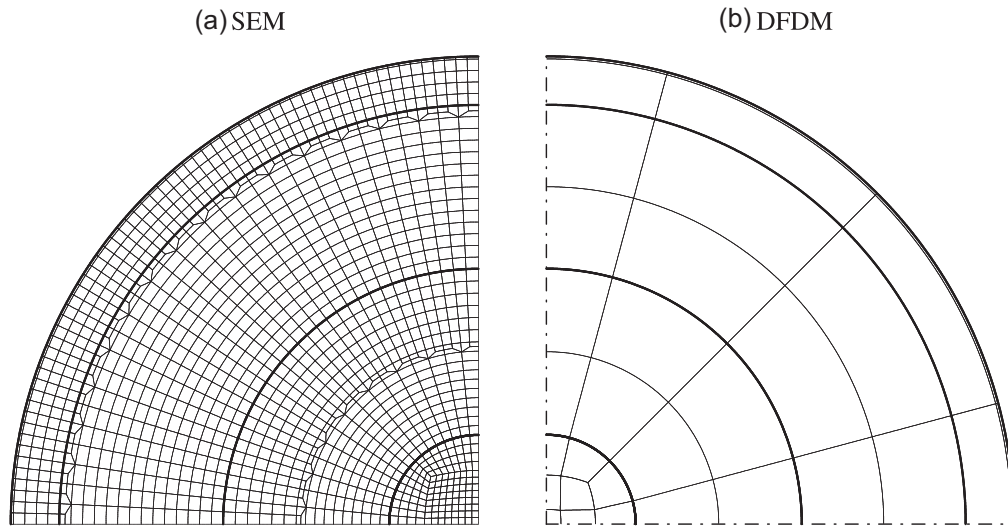


Figure 3. Numerical mesh used for the SEM simulations (left) and the DFDM simulation (right). In both cases, the mesh matches the principal discontinuities, that is the MOHO, the 670 discontinuity, the CMB, and the ICB. In the SEM simulations, the mesh has two doubling layers, and the number of elements is adjusted as a function of the desired accuracy. The first doubling layer is just under the 670 km discontinuity, and the second one is close to the middle of the outer core, between the CMB and the ICB. For the DFDM simulation, the number of elements is independent of the desired accuracy. We adjust the accuracy by varying the number of gridpoints within the elements. Because DFDM allows for non-conformal interfaces, the number of gridpoints is adjusted independently inside individual elements. The numbers of elements used for the SEM-Ref, SEM-1, SEM-2 and DFDM are indicated in Table 1.

To perform the DFDM simulation, we decompose the model into multiple elements, following a standard cube sphere approach (Komatitsch & Tromp 1999; Nissen-Meyer *et al.* 2008), as shown in Fig. 3(b). There are eight layers of elements in the numerical mesh. They match the principal interfaces of the ak135 model, that is, the MOHO at a depth of 35 km, the transition zone between the upper and the lower mantle at a depth of 660 km, the core–mantle boundary (CMB) at a depth of 2891.5 km and the inner core boundary (ICB) at a depth of 5153.5 km. To achieve a relatively homogeneous number of gridpoints per minimum wavelength and improve efficiency, we use two sublayers in the lower mantle, the outer core, and the inner core. In the horizontal direction, the mesh consists of 12 slices (i.e. three slices connected to each face of the central cube). Thus, the numerical mesh contains 93 elements in total. We set the number of gridpoints inside the elements so that there are at least three gridpoints per wavelength (using the grid spacing associated with the virtual gridpoints in Appendix D). We shall see that this gives an accuracy comparable to a sufficiently accurate SEM simulation. The enhanced accuracy of DFDM is attributed to different factors. First, as opposed to SEM, no approximation is used in the mass matrix, which gives the differential operators a spectral nature. Secondly, the numerical solution is highly continuous, thanks to the B-spline bases. Because fewer elements are needed, this reduces the number of interfaces at which the numerical wavefield is discontinuous. Last, mixed conditions are used to share the boundary values between neighbouring elements. Because DFDM allows for non-conformal interfaces, the number of gridpoints in different directions does not need to be the same in neighbouring elements. Finally, we assign the model parameters by evaluating the ak135 model at the virtual gridpoints in eq. (D1).

For the simulations, we consider a seismic event located at the north pole, at an intermediate depth of 50 km, to mitigate the amplitudes of the surface and body waves. We represent the seismic source using the moment tensor point source in eq. (D3) with

$$\mathbf{M} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \quad (32)$$

We take the source time function as the Ricker wavelet

$$s(t) = (1 - 2\pi^2 f_0^2 t^2) e^{-\pi^2 f_0^2 t^2} \quad (33)$$

with peak frequency $f_0 = 2.5 \times 10^{-2}$ Hz and an estimated maximum frequency $f_{\max} = 6.25 \times 10^{-2}$ Hz.

We recorded the wavefield at 20 receivers at the surface of the Earth. The receivers represented by triangles in Fig. 4 are equally spaced and buried at a depth of 10 m. Their epicentral distances vary from 9° to 180° .

3.2 Comparison with SEM

To assess the accuracy of the DFDM algorithm, we compare the DFDM solution to SEM solutions. The SEM method is well established and known to produce sufficiently accurate results for the setup considered. It is very close to the analytical when a sufficient number of gridpoints per wavelength is used. We ran three SEM simulations, we used the SPECFEM2D current source (Komatitsch *et al.* 2012) published under the GPL3 license. The first SEM simulation we will refer to as SEM-Ref is our reference simulation. It is highly accurate and relies on a fine mesh with at least 11 gridpoints per wavelength. The second SEM simulation, which we will refer to as SEM-1, uses a relatively standard

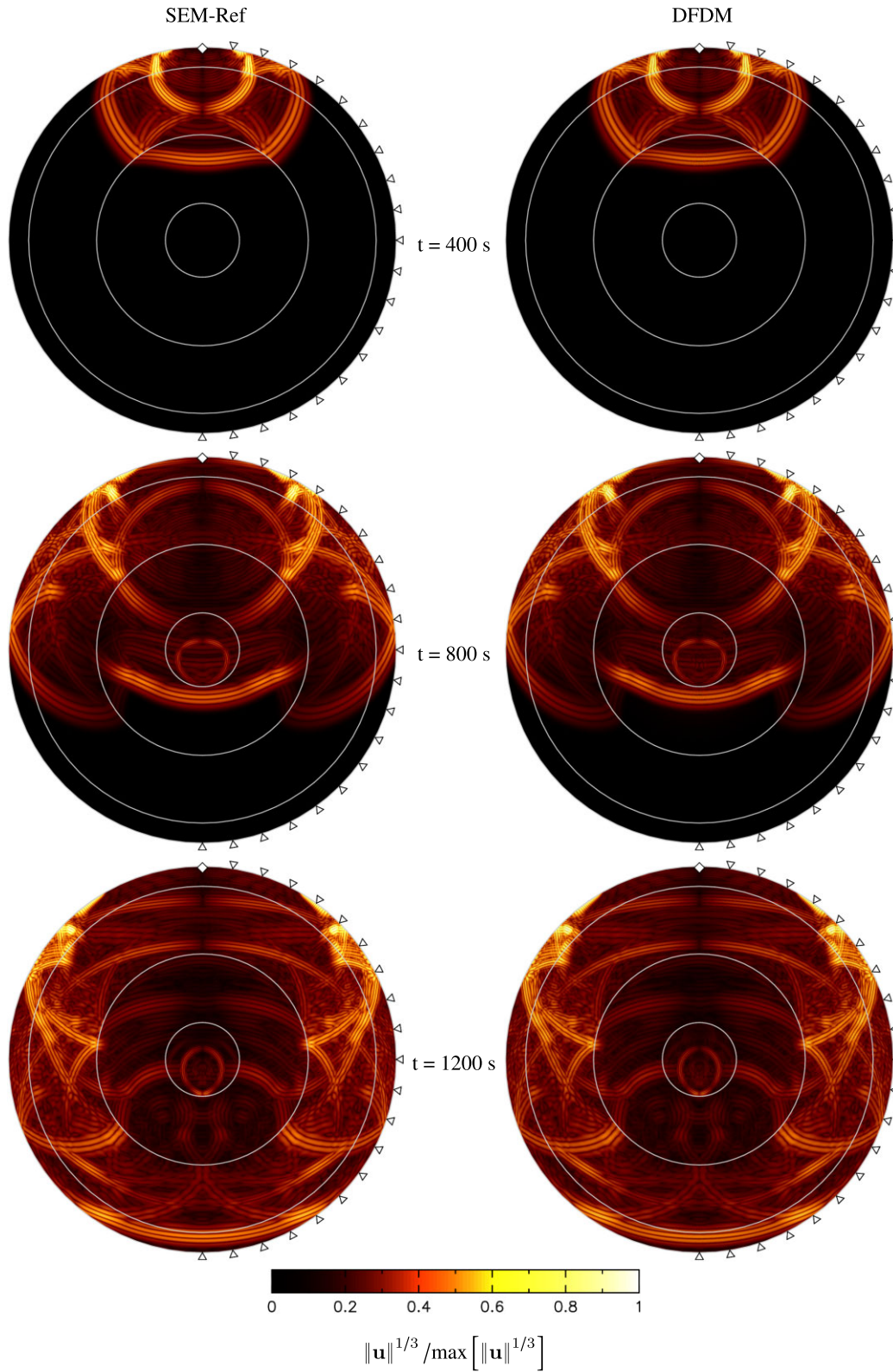


Figure 4. Snapshots showing the magnitude of the displacement modelled in the SEM-Ref (left) and DFDM (right) simulations, at times 400, 800 and 1200 s. The two wavefields look perfectly identical even though the mesh of the DFDM simulation is much coarser than in the SEM-Ref simulation.

Table 1. Summary of the parameters used in the different simulations, the measured errors, and CPU times. The SEM-Ref simulation is the reference simulation with the highest accuracy (i.e. at least 11 gridpoints per wavelength). The SEM-1 simulation has a standard setup (with at least 5.5 gridpoints per wavelength). The SEM-2 simulation uses a finer mesh than SEM-1 and is more accurate (with at least 6.9 gridpoints per wavelength). The DFDM simulation uses a coarse mesh (with at least 3 gridpoints per wavelength); it is more accurate and faster than SEM-1 and SEM-2.

Simulation	SEM-Ref	SEM-1	SEM-2	DFDM
Numerical method	SEM	SEM	SEM	DFDM
accuracy/polynomial order: p	4	4	4	4
Number of elements: N_e	322560	81408	126720	93
Number of gridpoints: N_G	5165057	1304577	2030081	$2 \times 188660 = 377320$
Number of element layers in the crust	2	1	2	1
Minimum number of points per P -wavelength: $\text{Min}[G_p(f_{\max})]$	19.0	9.5	11.9	3.1
Maximum number of points per P -wavelength: $\text{Max}[G_p(f_{\max})]$	68.5	34.2	42.8	14.1
Minimum number of points per S -wavelength: $\text{Min}[G_s(f_{\max})]$	11.0	5.5	6.9	3.1
Maximum number of points per S -wavelength: $\text{Max}[G_s(f_{\max})]$	22.0	11.0	13.7	4.6
Stable time step (estimated): Δt_{CFL}	7.23×10^{-2} s	1.4×10^{-1} s	1.2×10^{-1} s	1.5×10^{-1} s
Time step used: Δt	7×10^{-2} s			
Dominant frequency: f_0	2.5×10^{-2} Hz			
Maximum frequency: $f_{\max}$	6.25×10^{-2} Hz			
Dominant period: T_0	40 s			
Minimum period: $T_{\max}$	16 s			
Number of time steps: N_t	72000			
Simulation duration: T	5040 s			
CPU time (indicational): N_t	6 h 34 m	1 h 50 m	2 h 33 m	1 h 48 m
Normalized standard error: $\sigma[\mathbf{u} - \mathbf{u}_{\text{ref}}] / \sigma[\mathbf{u}_{\text{ref}}]$	0.00	0.19	0.12	0.10
Normalized maximum error: $\text{Max}[\mathbf{u} - \mathbf{u}_{\text{ref}}] / \text{Max}[\mathbf{u}_{\text{ref}}]$	0.00	0.33	0.19	0.18
Pearson correlation coefficient: $\rho = \frac{\text{Cov}[\mathbf{u}, \mathbf{u}_{\text{ref}}]}{\sigma[\mathbf{u}]\sigma[\mathbf{u}_{\text{ref}}]}$	1.000	0.946	0.982	0.995

setup with at least 5.5 gridpoints per wavelength. The third simulation, which we will refer to as SEM-2, is more accurate than SEM-1 and uses a finer mesh with at least 6.9 gridpoints per wavelength.

Similarly to the DFDM simulation, the SEM simulations mesh pictured in Fig. 3(a) matches the principal interfaces of the ak135 model. The three SEM meshes used for the simulations have two doubling layers to obtain a relatively homogeneous number of gridpoints per minimum wavelength. The doubling layer is located just under the 670 km discontinuity. The second one is close to the middle of the outer core, between the CMB and the ICB. We give the total number of elements for SEM-Ref, SEM-1 and SEM-2 in Table 1. They are adjusted to achieve the desired accuracy with a minimum number of gridpoints per wavelength. Notably, the SEM-1 mesh has a single layer of elements inside the crust, while SEM-Ref and SEM-2 use two layers of elements. Thus, the crust is modelled much more accurately in the SEM-Ref and SEM-2 simulations. In the SEM simulations, we assign the model parameters by evaluating the ak135 model at the gridpoints. We use the same time step in all simulations. It is small enough to minimize the numerical dispersion associated with the time integration scheme. It is close to the stable time step of reference simulation SEM-Ref, which has the most restrictive stability condition. The stable time step in DFDM depends on the basis functions used and may be adjusted by varying the spacing between the basis functions toward the sides of the elements. With the setup we use here, the stability condition is slightly more restrictive for DFDM than for SEM. However, because DFDM requires less gridpoints per wavelength to achieve a certain accuracy, the stable time step is larger in the DFDM simulations.

3.3 Results and discussion

We first compare the results obtained using DFDM to those obtained for the reference solution SEM-Ref, which is assumed to be nearly perfectly accurate. In Fig. 4, we present snapshots showing the wavefield obtained for DFDM and SEM-Ref. Even though DFDM uses as few as 3 points per wavelength, we observe no visible difference between the two wavefields, which highlights the high accuracy of DFDM when using coarse meshes (Masson 2022; Masson & Virieux 2023). Note the P to S and S to P wave conversions at the solid–fluid interfaces are accurately modelled by DFDM, which uses the same scheme in the entire domain, while SEM uses a specific solid–fluid coupling approach. This is due to the specific choice of the boundary coefficients α in eq. (23), as detailed in Appendix E. Generally, when dealing with interconnected purely solid or fluid elements, the choice of alpha values is relatively arbitrary. However, in the case of solid–fluid coupling interfaces, the value of alpha is fixed. On the one hand, we use the internal value of the displacement in the solid elements (i.e. $\alpha = 1$). On the other hand, we take the internal value of the pressure in the fluid elements (i.e. $\alpha = 0$). This specific choice enforces the continuity of the pressure and gives the most accurate results.

In Figs 5(a) and 6(a), we superimpose the seismograms simulated by DFDM and SEM-Ref at the 20 receivers. There is no notable difference between the seismograms obtained using DFDM and SEM-Ref, both for the body waves and for the strong dispersive surface waves. This is valid for all epicentral distances, which is remarkable and confirms the high accuracy of DFDM.

To quantify the error, we measured the following quantities for our DFDM simulation:

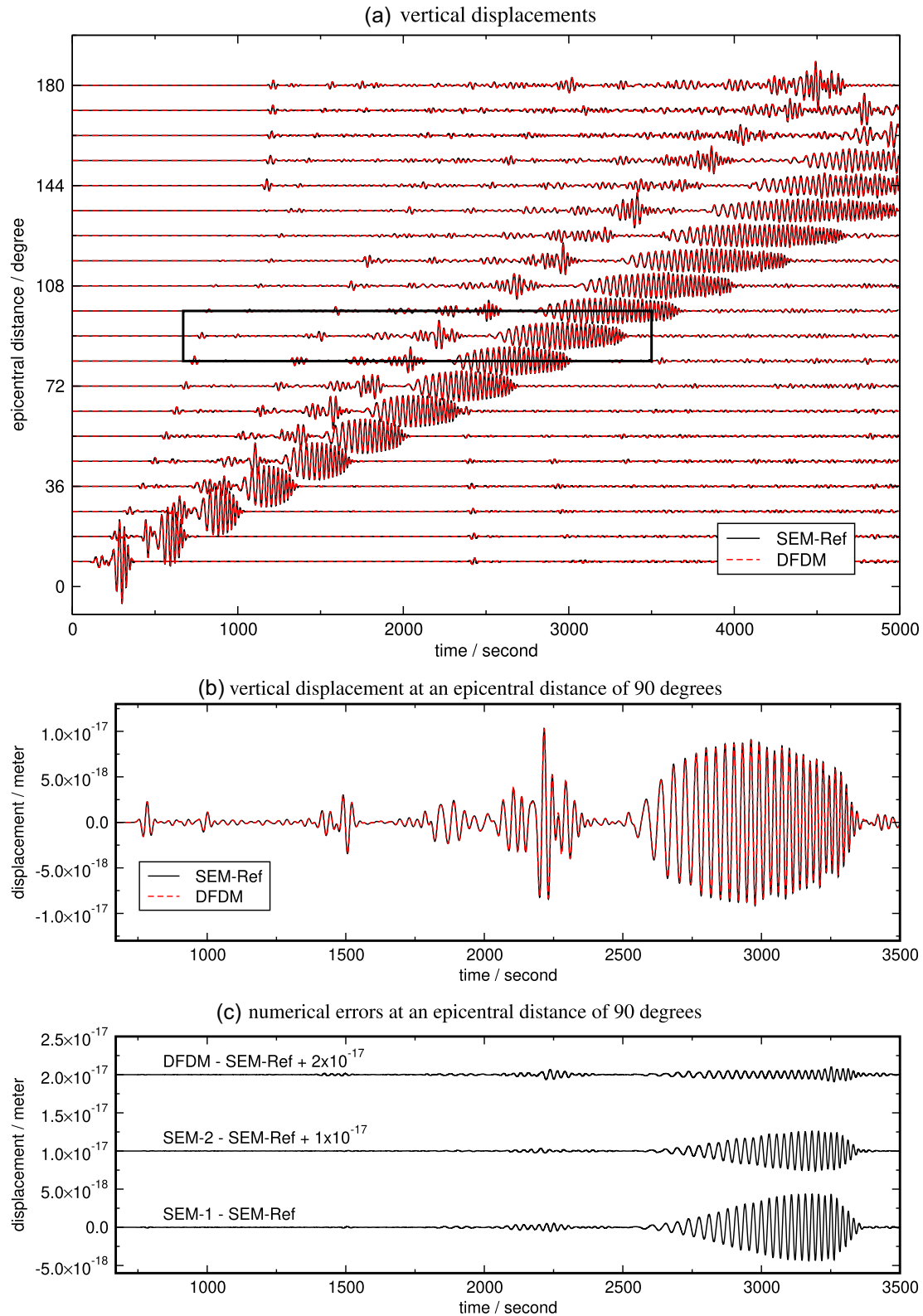


Figure 5. (a) Seismograms (vertical component of the displacement vector) simulated at the 20 receivers represented by triangles in Fig. 4. The solid black and the dashed red lines show the seismogram modelled in the reference SEM-Ref and the DFDM simulations. (b) Magnified view of the seismogram recorded at an epicentral distance of 90° inside the black box in (a). (c) The errors of the seismograms obtained by the SEM-1, SEM-2 and DFDM simulations with respect to seismograms obtained by the reference SEM-Ref simulation.

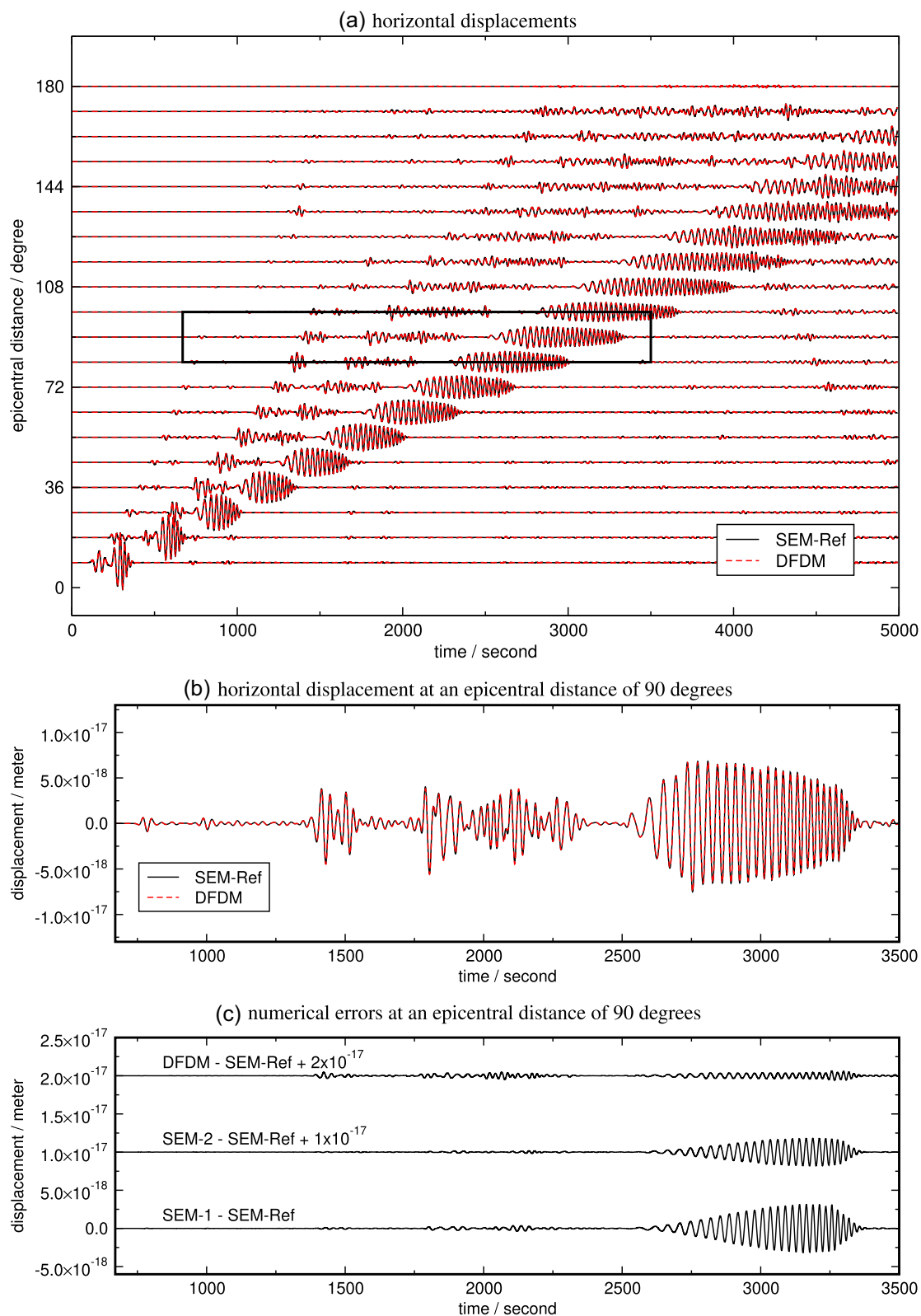


Figure 6. Same as in Fig. 5 but for the horizontal components.

(i) The normalized maximum error

$$\tilde{\epsilon} = \frac{\text{Max}[|\mathbf{u} - \mathbf{u}_{\text{ref}}|]}{\text{Max}[|\mathbf{u}_{\text{ref}}|]}, \quad (34)$$

where $\mathbf{u}$ and $\mathbf{u}_{\text{ref}}$ are the vectors containing the displacement components at all times at all receivers for the considered and the reference solutions.

(ii) The standard deviation of the error normalized by that of the reference solution

$$\tilde{\sigma} = \frac{\sigma[\mathbf{u} - \mathbf{u}_{\text{ref}}]}{\sigma[\mathbf{u}_{\text{ref}}]}, \quad (35)$$

where σ denotes the standard deviation.

(iii) The Pearson correlation coefficient

$$\rho = \frac{\text{Cov}[\mathbf{u} - \mathbf{u}_{\text{ref}}]}{\sigma[\mathbf{u}]\sigma[\mathbf{u}_{\text{ref}}]}, \quad (36)$$

where Cov denotes the covariance.

We show values of the errors in Table 1. These errors may appear relatively large, but the measured Pearson correlation coefficient $\rho = 0.995$ indicates an excellent correlation between the DFDM and the SEM-Ref simulations ($\rho = 1$ corresponds to a perfect correlation). In the DFDM simulation, the maximum difference and its standard deviation are 18 and 10 per cent.

To investigate the relative accuracy of DFDM with respect to SEM when a standard setup is used, we compare the differences between the seismograms obtained using DFDM, SEM-1, SEM-2 and SEM-Ref in Figs 5(c) and 6(c). We observe that SEM-1 has the most prominent error, while DFDM has the smallest error. Note that the error is principally associated with the surface waves that are notoriously difficult to model accurately. The surface waves are primarily sensitive to the velocity in the crust, and we know that DFDM is more accurate than SEM when using a similar number of points per wavelength (Masson 2022). Thus, we anticipate that DFDM is more accurate than SEM-1 because DFDM and SEM-1 use one layer of elements in the crust with 5 gridpoints in the vertical direction. What is more remarkable is the superior accuracy of DFDM with respect to SEM-2, because SEM-2 uses two layers of elements inside the crust (which corresponds to 9 gridpoints in the vertical direction). In contrast, DFDM uses only one layer of elements (with 5 gridpoints in the vertical direction). Furthermore, DFDM uses only 3.5 points per wavelength in the crust in the horizontal direction, while SEM-2 uses almost 7. These observations show that DFDM is remarkably accurate and adapted to model surface waves. Such waves are notoriously challenging in global and regional seismological modelling.

To better quantify the numerical error, we also measured the error, its standard deviation, and the Pearson correlation coefficient for SEM-1 and SEM-2. We show values of the errors in Table 1. In SEM-1, we use the recommended setup for SEM, that is at least 5 points per wavelength (Komatitsch & Vilotte 1998), and we expect no significant numerical dispersion. We find a maximum error of 33 per cent, and its standard deviation is 19 per cent. Thus the typical error is twice as large in SEM-1 as in DFDM. The measured Pearson correlation coefficient $\rho = 0.946$ also indicates a degraded correlation with the reference solution. It is remarkable given that the grid size (i.e. the number of gridpoints) in SEM-1 is about 3.5 times larger than in DFDM. For SEM-2, where there are at least 6.9 points per wavelength, we find a maximum error of 19 per cent, a standard deviation is 12 per cent, and the Pearson correlation coefficient is $\rho = 0.982$. These values show that the error in SEM-2 is slightly larger than in DFDM. We consider, however, that these two simulations have comparable accuracy. Looking at the grid dimensions in DFDM and SEM-2 suggests that SEM requires roughly five times more gridpoints to achieve an accuracy similar to DFDM in 2-D. Because DFDM requires more computational effort than SEM to update the wavefield, comparing the CPU time measured for the different simulations is interesting. The values given in Table 1 are only indicative because the specfem2D code used for the SEM simulation is a well-established, and optimized software while our DFDM code is still at an early development stage. Nonetheless, we observe that the CPU time for DFDM is slightly shorter than for SEM-1 and significantly smaller than for SEM-2. Thus DFDM appears substantially faster than SEM to achieve comparable accuracy.

4 DISCUSSION AND OUTLOOK

We showed that DFDM is significantly more accurate than SEM in numerical simulations of the global wave propagation. Through measurements of the CPU time, we observed that the gain in accuracy achieved with DFDM also translates into better computational efficiency. It is especially striking that we can model the surface waves with fewer points per wavelength than in SEM.

For further applications and to better estimate the net gain in computational efficiency with respect to SEM, our algorithm needs to be expanded from two to three dimensions. This is the subject of a forthcoming paper. Lateral variations in the velocity structure and topography will also need to be accounted for in real applications. In this study, we limited our numerical tests to an axisymmetric model without topography for an easier comparison with SEM. This is because material heterogeneity is accounted for using slightly different approximations in SEM and DFDM, and obtaining a trustworthy reference solution in heterogeneous media is delicate. Nonetheless, Masson & Virieux (2023) demonstrated that DFDM can accurately account for heterogeneity within the elements, including sharp interfaces. Furthermore, Masson (2022) showed that DFDM is in agreement with SEM in the case of strong topographic variations. Accounting for the 3-D structure of the

Earth's crust is especially challenging in global seismology. Our numerical examples demonstrate that DFDM allows us to use non-conformal meshes to efficiently handle the strong contrast between the crust and the upper mantle. This is more delicate in SEM, where doubling layers or similar approaches are needed to refine the numerical grid with respect to depth or as a function of velocity fluctuations. Furthermore, DFDM also allows us to use small elements as SEM, and similar strategies may be adopted. Thus, DFDM brings additional flexibility to accurately account for structures with complicated geometries. In practice, a 3-D crust may be implemented by varying the upper element thickness to accommodate Moho topography; additional elements may be added to account for sharp lateral variations.

Accounting for attenuation should pose no particular problems and may be modelled using readily available time-domain algorithms relying on the generalized Maxwell body (GMB-EK) or the generalized Zener body. For regional simulations, the computational domain needs to be truncated using absorbing boundaries. This will be discussed in a separate manuscript where we show that the perfectly matched layer method can be easily adapted to DFDM and is stable under certain conditions that we outline.

In this study, we used a second-order displacement time integration scheme which makes it possible to a parsimonious strategy that avoids the storage of stress components within the elements (Luo & Schuster 1990). While the proposed algorithm is not strictly parsimonious, It can be made parsimonious by changing the synchronization, that is adjusting the moments when the boundary values are exchanged and the boundary conditions are applied.

5 CONCLUSION

We introduced a staggered-grid DFDM for modelling seismic wave propagation through the 2-D Earth. We compared the performance of DFDM and the popular SEM when modelling wave propagation through the same reference earth model. We emphasize the following characteristics of the proposed approach (i) the simplicity of the algorithm, locally similar to the finite-difference method, globally similar to the discontinuous finite-element method; (ii) the versatility of the implementation allowing for the domain decomposition, heterogeneity of the medium, curvilinear mesh, anisotropy, non-conformal interfaces, or discontinuous grid and fluid–solid interfaces and (iii) the efficiency with superior accuracy, reduced memory usage ($G \approx 3$), and potentially higher efficiency than SEM. Thus, we conclude that DFDM is a potentially interesting alternative to SEM for global tomography, which heavily relies on sizeable global scale simulations.

ACKNOWLEDGMENTS

This research used resources of the National Energy Research Scientific Computing Center (NERSC), a U.S. Department of Energy Office of Science User Facility located at Lawrence Berkeley National Laboratory, operated under Contract No. DE-AC02-05CH11231, using NERSC ERCAP awards m554". Yder Masson's work was supported by the I-SITE E2S/UPPA/THERMapp grant. Yder Masson, Barbara Romanowicz and Chao Lyu are grateful to the France-Berkeley Fund (Project #10-2019). The authors would like to express their gratitude to Editor Ebru Bozdag and Reviewer Hom Nath Gharti for their constructive comments and remarks, which helped improve our paper.

DATA AVAILABILITY

The data underlying this paper will be shared on reasonable request to the corresponding author (yder_masson@berkeley.edu). The SPECfem software can be obtained at the following address <https://geodynamics.org/resources/specfem2d> and is described in Komatitsch & Vilotte (1998).

REFERENCES

- Appelö, D. & Petersson, N.A., 2009. A stable finite difference method for the elastic wave equation on complex geometries with free surfaces, *Commun. Comput. Phys.*, **5**(1), 84–107.
- Auricchio, F., Da Veiga, L.B., Hughes, T., Reali, A. & Sangalli, G., 2010. Iso-geometric collocation methods, *Math. Models Methods Appl. Sci.*, **20**(11), 2075–2107.
- Ba, Z. & Gao, X., 2017. Soil-structure interaction in transversely isotropic layered media subjected to incident plane SH waves, *Shock Vib.*, **2017**, doi:10.1155/2017/2834274.
- Bécache, E., Joly, P. & Tsogka, C., 2002. A new family of mixed finite elements for the linear elastodynamic problem, *SIAM J. Numer. Anal.*, **39**(6), 2109–2132.
- Bernth, H. & Chapman, C., 2011. A comparison of the dispersion relations for anisotropic elastodynamic finite-difference grids, *Geophysics*, **76**(3), WA43–WA50.
- Bohlen, T. & Saenger, E.H., 2006. Accuracy of heterogeneous staggered-grid finite-difference modeling of Rayleigh waves, *Geophysics*, **71**(4), T109–T115.
- Bonnasse-Gahot, M., Calandra, H., Diaz, J. & Lanteri, S., 2018. Hybridizable discontinuous Galerkin method for the 2-D frequency-domain elastic wave equations, *Geophys. J. Int.*, **213**(1), 637–659.
- Bouchon, M. & Coutant, O., 1994. Calculation of synthetic seismograms in a laterally varying medium by the boundary element-discrete wavenumber method, *Bull. seism. Soc. Am.*, **84**(6), 1869–1881.
- Bozdağ, E., Peter, D., Lefebvre, M., Komatitsch, D., Tromp, J., Hill, J., Podhorszki, N. & Pugmire, D., 2016. Global adjoint tomography: first-generation model, *Geophys. J. Int.*, **207**(3), 1739–1766.
- Capdeville, Y., Chaljub, E. & Montagner, J.P., 2003a. Coupling the spectral element method with a modal solution for elastic wave propagation in global earth models, *Geophys. J. Int.*, **152**(1), 34–67.
- Capdeville, Y., To, A. & Romanowicz, B., 2003b. Coupling spectral elements and modes in a spherical earth: an extension to the 'sandwich' case, *Geophys. J. Int.*, **154**(1), 44–57.
- Carcione, J., 1996. A 2D Chebyshev differential operator for the elastic wave equation, *Comput. Methods Appl. Mech. Eng.*, **130**(1–2), 33–45.
- Chaljub, E. & Tarantola, A., 1997. Sensitivity of ss precursors to topography on the upper-mantle 660-km discontinuity, *Geophys. Res. Lett.*, **24**(21), 2613–2616.

- Cottaar, S. & Romanowicz, B., 2012. An unusually large ULVZ at the base of the mantle near Hawaii, *Earth planet. Sci. Lett.*, **355**, 213–222.
- De Basabe, J.D. & Sen, M.K., 2015. A comparison of finite-difference and spectral-element methods for elastic wave propagation in media with a fluid-solid interface, *Geophys. J. Int.*, **200**(1), 278–298.
- De Basabe, J.D., Sen, M.K. & Wheeler, M.F., 2008. The interior penalty discontinuous Galerkin method for elastic wave propagation: grid dispersion, *Geophys. J. Int.*, **175**(1), 83–93.
- De Boor, C., 1978. *A Practical Guide to Splines*, Vol. 27, Springer-Verlag New York.
- de la Puente, J., Dumbser, M., Käser, M. & Igel, H., 2008. Discontinuous Galerkin methods for wave propagation in poroelastic media, *Geophysics*, **73**(5), T77–T97.
- de la Puente, J., Ferrer, M., Hanzich, M., Castillo, J.E. & Cela, J.M., 2014. Mimetic seismic wave modeling including topography on deformed staggered grids, *Geophysics*, **79**(3), T125–T141.
- Dormy, E. & Tarantola, A., 1995. Numerical simulation of elastic wave propagation using a finite volume method, *J. geophys. Res.*, **100**(B2), 2123–2133.
- Dovgilevich, L. & Sofronov, I., 2015. High-accuracy finite-difference schemes for solving elastodynamic problems in curvilinear coordinates within multiblock approach, *Appl. Numer. Math.*, **93**, 176–194.
- Dumbser, M., Käser, M. & De La Puente, J., 2007. Arbitrary high-order finite volume schemes for seismic wave propagation on unstructured meshes in 2D and 3D, *Geophys. J. Int.*, **171**(2), 665–694.
- Étienne, V., Chaljub, E., Virieux, J. & Glinsky, N., 2010. An hp-adaptive discontinuous Galerkin finite-element method for 3-D elastic wave modelling, *Geophys. J. Int.*, **183**(2), 941–962.
- Fichtner, A., 2010. *Full Seismic Waveform Modelling and Inversion*, Springer Science & Business Media.
- Fichtner, A. et al., 2018. The collaborative seismic earth model: generation 1, *Geophys. Res. Lett.*, **45**(9), 4007–4016.
- Fornberg, B., 1987. The pseudospectral method: comparisons with finite differences for the elastic wave equation, *Geophysics*, **52**(4), 483–501.
- French, S. & Romanowicz, B.A., 2014. Whole-mantle radially anisotropic shear velocity structure from spectral-element waveform tomography, *Geophys. J. Int.*, **199**(3), 1303–1327.
- French, S., Lekic, V. & Romanowicz, B., 2013. Waveform tomography reveals channelled flow at the base of the oceanic asthenosphere, *Science*, **342**(6155), 227–230.
- Gao, L., Brossier, R., Pajot, B., Tago, J. & Virieux, J., 2015. An immersed free-surface boundary treatment for seismic wave simulation, *Geophysics*, **80**(5), T193–T209.
- Graves, R.W., 1993. Modeling three-dimensional site response effects in the marina district basin, San Francisco, California, *Bull. seism. Soc. Am.*, **83**(4), 1042–1063.
- Grote, M.J., Schneebeli, A. & Schötzau, D., 2007. Interior penalty discontinuous Galerkin method for Maxwell's equations: energy norm error estimates, *J. Comput. Appl. Math.*, **204**(2), 375–386.
- Hughes, T.J., Cottrell, J.A. & Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement, *Comp. Methods Appl. Mech. Eng.*, **194**(39–41), 4135–4195.
- Igel, H. & Weber, M., 1995. SH-wave propagation in the whole mantle using high-order finite differences, *Geophys. Res. Lett.*, **22**(6), 731–734.
- Igel, H., Mora, P. & Rioulet, B., 1995. Anisotropic wave propagation through finite-difference grids, *Geophysics*, **60**(4), 1203–1216.
- Jahnke, G., Thorne, M.S., Cochard, A. & Igel, H., 2005. Global SH-wave propagation using a parallel axis-symmetric finite-difference scheme, *Geophys. J. Int.*, **173**(3), 815–826.
- Jiang, L. & Zhang, W., 2021. TTI equivalent medium parametrization method for the seismic waveform modelling of heterogeneous media with coarse grids, *Geophys. J. Int.*, **227**, 2016–2043.
- Käser, M., Hermann, V. & Puente, J.D.L., 2008. Quantitative accuracy analysis of the discontinuous Galerkin method for seismic wave propagation, *Geophys. J. Int.*, **173**(3), 990–999.
- Kawai, K., Takeuchi, N. & Geller, R.J., 2006. Complete synthetic seismograms up to 2 Hz for transversely isotropic spherically symmetric media, *Geophys. J. Int.*, **164**, 411–424.
- Kennett, B.L., Engdahl, E. & Buland, R., 1995. Constraints on seismic velocities in the earth from traveltimes, *Geophys. J. Int.*, **122**(1), 108–124.
- Koene, E.F.M., Wittsten, J. & Robertsson, J.O.A., 2022. Finite-difference modelling of 2-D wave propagation in the vicinity of dipping interfaces: a comparison of anti-aliasing and equivalent medium approaches, *Geophys. J. Int.*, **229**, 70–96.
- Komatitsch, D. & Tromp, J., 1999. Introduction to the spectral element method for three-dimensional seismic wave propagation, *Geophys. J. Int.*, **139**(3), 806–822.
- Komatitsch, D. & Vilotte, J.-P., 1998. The spectral element method: an efficient tool to simulate the seismic response of 2D and 3D geological structures, *Bull. seism. Soc. Am.*, **88**(2), 368–392.
- Komatitsch, D. et al., 2012. *SPECFEM2D v7.0.0 [software]*, Computational Infrastructure for Geodynamics.
- Kristek, J., Moczo, P., Irikura, K., Iwata, T., Sekiguchi, H., Kudo, K., Okada, H. & Sasatani, T., 1999. The 1995 Kobe mainshock simulated by the 3D finite differences, in *The Effects of Surface Geology on Seismic Motion*, Vol. 3, pp. 1361–1368.
- Kristek, J., Moczo, P. & Archuleta, R.J., 2002. Efficient methods to simulate planar free surface in the 3D 4th-order staggered-grid finite-difference schemes, *Stud. Geophys. Geod.*, **46**, 355–381.
- Lan, H. & Zhang, Z., 2011. Three-dimensional wave-field simulation in heterogeneous transversely isotropic medium with irregular free surface, *Bull. seism. Soc. Am.*, **101**(3), 1354–1370.
- Lei, W. et al., 2020. Global adjoint tomography—model GLAD-M25, *Geophys. J. Int.*, **223**(1), 1–21.
- Lekić, V. & Romanowicz, B., 2011. Inferring upper-mantle structure by full waveform tomography with the spectral element method, *Geophys. J. Int.*, **185**(2), 799–831.
- Leng, K., Nissen-Meyer, T., Van Driel, M., Hosseini, K. & Al-Attar, D., 2019. AxisSEM3D: broad-band seismic wavefields in 3-D global earth models with undulating discontinuities, *Geophys. J. Int.*, **217**(3), 2125–2146.
- Li, D., Helmberger, D., Clayton, R.W. & Sun, D., 2014. Global synthetic seismograms using a 2-D finite-difference method, *Geophys. J. Int.*, **197**(2), 1166–1183.
- Li, X.-D. & Romanowicz, B., 1996. Global mantle shear velocity model developed using nonlinear asymptotic coupling theory, *J. geophys. Res.*, **101**(B10), 22 245–22 272.
- Li, Z., Leng, K., Jenkins, J. & Cottaar, S., 2022. Kilometer-scale structure on the core–mantle boundary near Hawaii, *Nat. Commun.*, **13**(1), doi:10.1038/s41467-022-30502-5.
- Lisitsa, V. & Vishnevskiy, D., 2010. Lebedev scheme for the numerical simulation of wave propagation in 3D anisotropic elasticity, *Geophys. Prospect.*, **58**(4), 619–635.
- Lombard, B., Piraux, J., Gélis, C. & Virieux, J., 2008. Free and smooth boundaries in 2-D finite-difference schemes for transient elastic waves, *Geophys. J. Int.*, **172**(1), 252–261.
- Lu, Y., Stehly, L., Paul, A. & AlpArray Working, G., 2018. High-resolution surface wave tomography of the European crust and uppermost mantle from ambient seismic noise, *Geophys. J. Int.*, **214**(2), 1136–1150.
- Luo, Y. & Schuster, G., 1990. Parsimonious staggered grid finite-differencing of the wave equation, *Geophys. Res. Lett.*, **17**(2), 155–158.
- Lysmer, J. & Drake, L.A., 1972. A finite element method for seismology, *Methods Comput. Phys.*, **11**, 181–216.
- Lyu, C., Capdeville, Y. & Zhao, L., 2020. Efficiency of the spectral element method with very high polynomial degree to solve the elastic wave equation, *Geophysics*, **85**(1), T33–T43.
- Madariaga, R., 1976. Dynamics of an expanding circular fault, *Bull. seism. Soc. Am.*, **66**(3), 639–666.
- Magnoni, F., Casarotti, E., Komatitsch, D., Di Stefano, R., Ciaccio, M.G., Tape, C., Melini, D., Michelini, A., Piersanti, A. & Tromp, J., 2022. Adjoint tomography of the Italian lithosphere, *Commun. Earth Environ.*, **3**(1), 1–12.
- Marfurt, K.J., 1984. Accuracy of finite-difference and finite-element modelling of the scalar and elastic wave equations, *Geophysics*, **49**(5), 533–549.
- Masson, Y., 2022. Distributional finite-difference modelling of seismic waves, *Geophys. J. Int.*, **233**(1), 264–296.

- Masson, Y. & Virieux, J., 2023. P-SV wave propagation in heterogeneous media: velocity-stress distributional finite-difference method, *Geophysics*, **88**(3), 1–92.
- Matsushima, S., 1998. 3D simulation of aftershocks of the Hyogoken Nanbu earthquake of 1995, in *The Effects of Surface Geology on Seismic Motion*, Vol. 2, pp. 1129–1136, Springer.
- Mégnin, C. & Romanowicz, B., 2000. The three-dimensional shear velocity structure of the mantle from the inversion of body, surface and higher-mode waveforms, *Geophys. J. Int.*, **143**(3), 709–728.
- Mittet, R., 2017. On the internal interfaces in finite-difference schemes, *Geophysics*, **82**, T159–T182.
- Moczo, P., Kristek, J. & Halada, L., 2000. 3D fourth-order staggered-grid finite-difference schemes: stability and grid dispersion, *Bull. seism. Soc. Am.*, **90**(3), 587–603.
- Moczo, P., Kristek, J. & Galis, M., 2014. *The Finite-Difference Modelling of Earthquake Motions: Waves and Ruptures*, Cambridge Univ. Press.
- Moczo, P. et al., 2021. Numerical wave propagation simulation, in *ESG 2021, The 6th IASPEI/IAEE International Symposium: Effects of Surface Geology on Seismic Motion*, Kyoto, Japan, USB, paper ID: GS3-103.
- Moczo, P., Kristek, J., Kristekova, M., Valovcan, J., Galis, M. & Gregor, D., 2022. Material interface in the finite-difference modeling: a fundamental view, *Bull. seism. Soc. Am.*, **113**, 281–296.
- Nédélec, J.-C., 1980. Mixed finite elements in R^3 , *Numer. Math.*, **35**, 315–341.
- New, K.C., Watt, K., Misner, C.W. & Centrella, J.M., 1998. Stable 3-level leapfrog integration in numerical relativity, *Phys. Rev. D*, **58**(6), doi:10.1103/PhysRevD.58.064022.
- Nissen-Meyer, T., Fournier, A. & Dahlen, F., 2008. A 2-D spectral-element method for computing spherical-earth seismograms—II. Waves in solid-fluid media, *Geophys. J. Int.*, **174**(3), 873–888.
- Nissen-Meyer, T., van Driel, M., Stähler, S.C., Hosseini, K., Hempel, S., Auer, L., Colombi, A. & Fournier, A., 2014. AxisSEM: broadband 3-D seismic wavefields in axisymmetric media, *Solid Earth*, **5**(1), 425–445.
- Olsen, K.B., 1995. Magnitude 7.75 earthquake on the San Andreas fault: three-dimensional ground motion in Los Angeles, *Science*, **270**, 1628–1632.
- Papageorgiou, A.S. & Pei, D., 1998. A discrete wavenumber boundary element method for study of the 3-D response 2-D scatterers, *Earthq. Eng. Struct. Dyn.*, **27**(6), 619–638.
- Petersson, N. & Sjogreen, B., 2017. *SW4, version 2.01 [software]*, Davis, CA.
- Petersson, N.A. & Sjogreen, B., 2015. Wave propagation in anisotropic elastic materials and curvilinear coordinates using a summation-by-parts finite difference method, *J. Comput. Phys.*, **299**, 820–841.
- Pitarka, A., Irikura, K., Iwata, T. & Sekiguchi, H., 1998. Three-dimensional simulation of the near-fault ground motion for the 1995 Hyogo-Ken Nanbu (Kobe), Japan, earthquake, *Bull. seism. Soc. Am.*, **88**(2), 428–440.
- Saenger, E.H. & Bohlen, T., 2004. Finite-difference modeling of viscoelastic and anisotropic wave propagation using the rotated staggered grid, *Geophysics*, **69**(2), 583–591.
- Shragge, J. & Konuk, T., 2020. Tensorial elastodynamics for isotropic media, *Geophysics*, **85**(6), T359–T373.
- Sjogreen, B. & Petersson, N.A., 2012. A fourth order accurate finite difference scheme for the elastic wave equation in second order formulation, *J. Sci. Computing*, **52**, 17–48.
- Smith, W.D., 1975. The application of finite element analysis to body wave propagation problems, *Geophys. J. Int.*, **42**(2), 747–768.
- Sun, Y.-C., Zhang, W. & Chen, X., 2016. Seismic-wave modeling in the presence of surface topography in 2D general anisotropic media by a curvilinear grid finite-difference method, *Bull. seism. Soc. Am.*, **106**(3), 1036–1054.
- Tape, C., Liu, Q., Maggi, A. & Tromp, J., 2009. Adjoint tomography of the southern California crust, *Science*, **325**(5943), 988–992.
- Thompson, J.F., Warsi, Z. & Mastin, C.W., 1982. *Numerical Grid Generation*, Elsevier Science Ltd.
- To, A., Romanowicz, B., Capdeville, Y. & Takeuchi, N., 2005. 3D effects of sharp boundaries at the borders of the African and Pacific superplumes: observation and modeling, *Earth planet. Sci. Lett.*, **233**(1–2), 137–153.
- Trinh, P.T., Brossier, R., Métivier, L., Tavad, L. & Virieux, J., 2019. Efficient 3D time-domain elastic and viscoelastic full waveform inversion using a spectral-element method on flexible Cartesian-based mesh, *Geophysics*, **84**(1), R75–R97.
- Tromp, J., Komatitsch, D. & Liu, Q., 2008. Spectral-element and adjoint methods in seismology, *Commun. Comput. Phys.*, **3**(1), 1–32.
- Virieux, J., 1986. P-SV wave propagation in heterogeneous media: velocity-stress finite-difference method, *Geophysics*, **51**(4), 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.
- Wald, D.J. & Graves, R.W., 1998. The seismic response of the Los Angeles Basin, California, *Bull. seism. Soc. Am.*, **88**(2), 337–356.
- Warburton, T., 2013. A low-storage curvilinear discontinuous Galerkin method for wave problems, *SIAM J. Sci. Comput.*, **35**(4), A1987–A2012.
- Woodhouse, J.H. & Dziewonski, A.M., 1984. Mapping the upper mantle: three-dimensional modeling of earth structure by inversion of seismic waveforms, *J. geophys. Res.*, **89**(B7), 5953–5986.
- Woźny, P., 2013. Construction of dual bases, *J. Comput. Appl. Math.*, **245**, 75–85.
- Xie, Z., Chan, C.-H. & Zhang, B., 2002. An explicit fourth-order orthogonal curvilinear staggered-grid FDTD method for Maxwell's equations, *J. Comput. Phys.*, **175**(2), 739–763.
- Yee, K., 1966. Numerical solution of initial boundary value problems involving Maxwell's equations in isotropic media, *IEEE Trans. Antenn. Propagat.*, **14**(3), 302–307.
- Zang, N., Zhang, W. & Chen, X., 2021. An overset-grid finite-difference algorithm for simulating elastic wave propagation in media with complex free-surface topography, *Geophysics*, **86**(4), T277–T292.
- Zhang, W. & Chen, X., 2006. Traction image method for irregular free surface boundaries in finite difference seismic wave simulation, *Geophys. J. Int.*, **167**(1), 337–353.
- Zhang, W., Shen, Y. & Zhao, L., 2012a. Three-dimensional anisotropic seismic wave modelling in spherical coordinates by a collocated-grid finite-difference method, *Geophys. J. Int.*, **188**(3), 1359–1381.
- Zhang, W., Zhang, Z. & Chen, X., 2012b. Three-dimensional elastic wave numerical modelling in the presence of surface topography by a collocated-grid finite-difference method on curvilinear grids, *Geophys. J. Int.*, **190**(1), 358–378.
- Zheng, L., Zhao, Q., Milkereit, B., Grasselli, G. & Liu, Q., 2014. Spectral-element simulations of elastic wave propagation in exploration and geotechnical applications, *Earthq. Sci.*, **27**, 179–187.
- Zhu, H.-J., Zhang, W. & Chen, X.-F., 2009. Two-dimensional seismic wave simulation in anisotropic media by non-staggered finite difference method, *Chinese J. Geophys.*, **52**(6), 1536–1546.
- Zou, Z., Hughes, T., Scott, M., Miao, D. & Sauer, R., 2022. Efficient and robust quadratures for isogeometric analysis: reduced Gauss and Gauss–Greville rules, *Comput. Methods Appl. Mech. Eng.*, **392**, doi:10.1016/j.cma.2022.114722.

APPENDIX A: CANONICAL BASES: STAGGERED B-SPLINE BASES

In order to derive the distributional finite-difference algorithm, one first needs to choose the canonical bases. In this study, we use B-spline bases (see, e.g. De Boor 1978) that are highly continuous and offer good accuracy. Some alternative bases are given in Masson (2022).

In one dimension, the B-spline basis β^p with polynomial order p and basis functions $\beta_i^p(x')$ ($i = 1, N$) compactly supported in the interval $[x'_-, x'_+]$ is constructed as follows. We first define the equally spaced knot vector $\mathbf{k}$ with $N + p + 1$ elements that starts and ends with

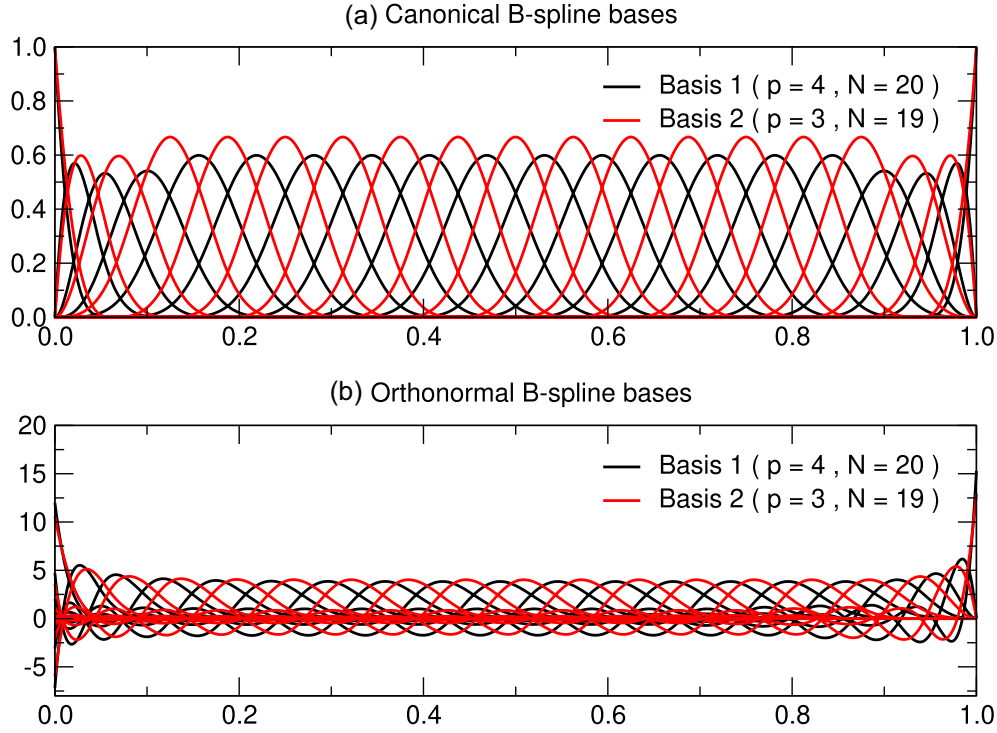


Figure A1. Illustration of the basis functions B^1 and B^2 and their orthogonal counterparts $\hat{B}^1$ and $\hat{B}^2$ for $p = 5$.

$p + 1$ repeated knots

$$\mathbf{k} \stackrel{\text{def}}{=} [k_1, k_2, \dots, k_{N+p+1}] \stackrel{\text{def}}{=} \begin{cases} k_i = x'_- & \text{for } i = 1, p + 1 \\ k_i = x'_- + \frac{i-p-1}{N-p}(x'_+ - x'_-) & \text{for } i = p + 2, N \\ k_i = x'_+ & \text{for } i = N + 1, N + p + 1 \end{cases}. \quad (\text{A1})$$

Then, starting with the zero-order B-spline basis β^0 with basis functions

$$\beta_i^0(x') = \begin{cases} 1 & \text{if } k_i < x' < k_{i+1} \\ 0 & \text{otherwise} \end{cases}, \quad (\text{A2})$$

we obtain the basis functions of the B-spline basis β^p with polynomial order p recursively using the Cox-de Boor formula

$$\beta_i^p(x') = \frac{x' - k_i}{k_{i+p} - k_i} \beta_i^{p-1}(x') + \frac{k_{i+p+1} - x'}{k_{i+p+1} - k_{i+1}} \beta_{i+1}^{p-1}(x'). \quad (\text{A3})$$

In our examples, we use pairs B^1/B^2 of B-spline bases where $B^1 \stackrel{\text{def}}{=} \beta^{p_1}(x')$ with $(p_1 = p = 4, N_1 = N)$ and $B^2 \stackrel{\text{def}}{=} \beta^{p_2}(x')$ with $(p_2 = p - 1 = 3, N_2 = N - 1)$. In this case, the basis functions are shifted similarly to the staggered finite-difference grid, as illustrated in Fig. A1(a).

The 2-D canonical B-spline bases are constructed from the 1-D bases $B_x^k(x')$ ($k = 1, 2$) and $B_y^l(y')$ ($l = 1, 2$) defined in direction x' and y' using a tensor product. We have

$$B_{ij}^{kl}(x', y') = B_{x_i}^k(x') B_{y_j}^l(y'). \quad (\text{A4})$$

Such 2-D canonical basis functions are illustrated in Fig. 2(a).

APPENDIX B: DEFINITIONS: MASS, STIFFNESS AND BOUNDARY MATRICES

We define the matrices and vectors needed to construct the orthonormal basis functions and the DFD operators. We build those matrices and vectors from an arbitrary pair of canonical bases B^1/B^2 . In this paper, we use staggered B-spline bases, as detailed in Appendix A and illustrated in Fig. A1.

The mass matrix is:

$$\mathbf{M}_\theta^{\mathbf{kl}} \stackrel{\text{def}}{=} (\mathbf{M}_\theta^{\mathbf{kl}})_{ij} = \int_{\theta_-}^{\theta_+} B_{\theta_i}^k(\theta) B_{\theta_j}^l(\theta) d\theta \quad \begin{cases} \theta \stackrel{\text{def}}{=} x' \text{ or } y' \\ (\mathbf{k}, \mathbf{l}) \stackrel{\text{def}}{=} (1, 2) \text{ or } (2, 1) \end{cases}. \quad (\text{B1})$$

Here, $B_{\theta i}^k(\theta)$ and $B_{\theta j}^l(\theta)$ are the 1-D canonical basis functions presented in Appendix A. The subscript $\theta \stackrel{\text{def}}{=} x'$ or y' denotes the direction of the computational domain x' or y' along which $\mathbf{M}_\theta$ is computed. $\theta \stackrel{\text{def}}{=} x'$ or y' denotes the coordinates in the computational domain with bounds $[\theta_-, \theta_+] \stackrel{\text{def}}{=} [x'_-, x'_+]$ and $[y'_-, y'_+]$. The superscripts **kl** indicate the bases from which $\mathbf{M}_\theta^{\text{kl}}$ is computed (B^1 or B^2 in Appendix A). When the basis functions have compact support, the mass matrix is band diagonal, ensuring the computations' efficiency. Furthermore, if the two bases are identical (i.e. $\mathbf{k} = \mathbf{l}$), then the mass matrix $\mathbf{M}_\theta^{\text{kk}}$ is symmetric. It is used to construct the orthonormal bases introduced in the next section.

The stiffness matrix is

$$\mathbf{K}_\theta^{\text{kl}} \stackrel{\text{def}}{=} (\mathbf{K}_\theta^{\text{kl}})_{ij} = \int_{\theta_-}^{\theta_+} B_{\theta i}^k(\theta) B_{\theta j}^{l(1)}(\theta) d\theta \quad (\text{B2a})$$

$$= - \int_{\theta_-}^{\theta_+} B_{\theta i}^{k(1)}(\theta) B_{\theta j}^l(\theta) d\theta + [B_{\theta i}^k(\theta) B_{\theta j}^l(\theta)]_{\theta_-}^{\theta_+} \quad (\text{B2b})$$

$$\stackrel{\text{def}}{=} -(\mathbf{K}_\theta^{\text{lk}})^T + \mathbf{Q}_\theta \quad (\text{B2c})$$

where $\theta \stackrel{\text{def}}{=} x$ or y and $(\mathbf{k}, \mathbf{l}) \stackrel{\text{def}}{=} (1, 2)$ or $(2, 1)$. In the expression above, $B_\theta^{(1)}$ denotes the first derivative of the basis functions B_θ .

The boundary matrix $\mathbf{Q}_\theta$ in eq. (B2) is:

$$\mathbf{Q}_\theta = -\mathbf{p}_\theta^{\mathbf{k}} \cdot \mathbf{p}_\theta^{\mathbf{l}^T} + \mathbf{q}_\theta^{\mathbf{k}} \cdot \mathbf{q}_\theta^{\mathbf{l}^T} = \begin{bmatrix} -1 & & \\ & 0 & \\ & & +1 \end{bmatrix} \quad \mathbf{p}_\theta^{\mathbf{k}/\mathbf{l}} = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \quad \mathbf{q}_\theta^{\mathbf{k}/\mathbf{l}} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}. \quad (\text{B3})$$

Note that eq. (B3) is specific to the B-spline bases considered in this paper because all the basis functions are equal to zero at the extremities of the interval considered, except the first one and the last one. More general expressions are available in (Masson 2022).

APPENDIX C: ORTHONORMAL BASES: ORTHONORMAL B-SPLINE BASES

To construct the orthonormal bases $\hat{B}_\theta^k$ used to represent the field variables, we start by computing the mass matrices $\mathbf{M}_\theta^{\text{kk}}$ in eq. (B1) associated with the canonical bases B^k (for $k = 1, 2$). Then, we compute the Cholesky factorizations $\mathbf{L}_\theta^k$ of the mass matrices. we have

$$\mathbf{M}_\theta^{\text{kk}} = \mathbf{L}_\theta^k \cdot \mathbf{L}_\theta^{k^T} \quad \begin{cases} \theta \stackrel{\text{def}}{=} x' \text{ or } y' \\ \mathbf{k} \stackrel{\text{def}}{=} 1 \text{ or } 2 \end{cases} \quad (\text{C1})$$

and we define the basis functions $\hat{B}_{x_i}^k(x')$ of the orthonormal bases $\hat{B}_x^k(x')$ as the linear combinations

$$\hat{B}_{\theta i}^k(\theta) = \sum_{j=1}^{N_\theta} (\mathbf{L}_\theta^{\mathbf{k}})^{-1}_{ij} B_{\theta j}^k(\theta) \quad \text{for } \theta = x', y' \text{ and } \mathbf{k} = 1, 2, \quad (\text{C2})$$

where $(\mathbf{L}_\theta^{\mathbf{k}})^{-1}_{ij}$ denotes the elements of the inverse of the matrix $\mathbf{L}_\theta^{\mathbf{k}}$. The basis function in eq. (C2) are orthonormal in the sense that

$$\hat{\mathbf{M}}_\theta^{\text{kk}} \stackrel{\text{def}}{=} (\hat{\mathbf{M}}_\theta^{\text{kk}})_{ij} = \int_{\theta_-}^{\theta_+} \hat{B}_{\theta i}^k(\theta) \hat{B}_{\theta j}^k(\theta) d\theta = \delta_{ij} \stackrel{\text{def}}{=} \mathbf{I}, \quad (\text{C3})$$

where δ_{ij} is the Kronecker Delta and $\mathbf{I}$ is the identity matrix. This result is easily verified by inserting eq. (C2) in eq. (C3) and using eq. (B1). Note that we chose the Cholesky factorization in eq. (C1) for its simplicity and efficiency. However, any factorization of the form $\mathbf{M}_\theta^{\mathbf{k}} = \mathbf{A}_\theta^{\mathbf{k}} \cdot \mathbf{A}_\theta^{k^T}$ may be used to construct the proposed algorithm. Eq. (C3) shows that the orthonormal bases $\hat{B}_\theta^k(\theta)$ are their own dual bases, by definition.

Finally, we construct the 2-D orthonormal B-spline bases used to represent the discrete field variables $(v_x^{21}, v_y^{12}, \sigma_{xx}^{11}, \sigma_{yy}^{11}, \sigma_{xy}^{22})$ and $(v_x^{12}, v_y^{21}, \sigma_{xx}^{22}, \sigma_{yy}^{22}, \sigma_{xy}^{11})$ introduced in Section 2.3 using a tensor product. We have

$$\hat{B}_{ij}^{kl}(x', y') = \hat{B}_{x_i}^k(x') \hat{B}_{y_j}^l(y'), \quad (\text{C4})$$

where $\hat{B}_x^k(x')$ ($k = 1, 2$) and $\hat{B}_y^l(y')$ ($l = 1, 2$) are the 1-D bases defined in directions x' and y' .

APPENDIX D: DISCRETIZATION OF THE MATERIAL PARAMETERS USING VIRTUAL GRIDS

To discretize the material parameters, we adopt the approach introduced in Masson & Virieux (2023). We define virtual grids where each basis function is associated with a unique gridpoint. The grid coordinates are computed from the 1-D basis in the eq. (C2); we have

$$\hat{x}_i^k = \frac{\int_{x'_-}^{x'_+} \hat{B}_{x_i}^k(x') dx'}{\int_{x'_-}^{x'_+} \hat{B}_{x_i}^k(x') dx'} \quad \hat{y}_j^l = \frac{\int_{y'_-}^{y'_+} \hat{B}_{y_j}^l(y') dy'}{\int_{y'_-}^{y'_+} \hat{B}_{y_j}^l(y') dy'}. \quad (\text{D1})$$

The coordinates $(\hat{x}_i^k, \hat{y}_j^l)$ define the virtual grids G^{kl} associated with the basis $\hat{B}^{kl}$ in each domain. They can be used to assign the values to the material parameter arrays. In the proposed algorithm, we use the following:

$$|\mathbf{J}|^{\text{kl}} \stackrel{\text{def}}{=} [|\mathbf{J}|^{\text{kl}}]_{i,j} = |J|(\hat{x}_i^k, \hat{y}_j^l) \quad (\text{D2a})$$

$$\mathbf{J}_{ab}^{-1\text{kl}} \stackrel{\text{def}}{=} [\mathbf{J}_{ab}^{-1\text{kl}}]_{i,j} = J_{ab}(\hat{x}_i^k, \hat{y}_j^l) = \frac{\partial a'}{\partial b}(\hat{x}_i^k, \hat{y}_j^l) \quad \text{where} \quad ab = xx, xy, yx, yy \quad (\text{D2b})$$

$$\rho^{\text{kl}} \stackrel{\text{def}}{=} [\rho^{\text{kl}}]_{i,j} = \rho(\hat{x}_i^k, \hat{y}_j^l) \quad (\text{D2c})$$

$$C_{mn}^{\text{kl}} \stackrel{\text{def}}{=} [C_{mn}^{\text{kl}}]_{i,j} = C_{mn}(\hat{x}_i^k, \hat{y}_j^l). \quad (\text{D2d})$$

Moment tensor point source representation

We represent the seismic source using the moment tensor point source

$$\mathbf{M}(\mathbf{x}, t) = [\mathbf{M} \cdot \delta(\mathbf{x} - \mathbf{x}_s)] \cdot s(t) \quad \mathbf{M} = \begin{bmatrix} M_{xx} & M_{xy} \\ M_{xy} & M_{yy} \end{bmatrix}, \quad (\text{D3})$$

where $s(t)$ is the source time function, $\mathbf{M}$ is the moment tensor, δ is the delta function and $\mathbf{x}_s$ is the source position vector. Practically, we implement the seismic source by adding the components of the moment tensor to the stresses in eq. (29). We have

$$\mathcal{M}_{xx}^{kl}(t) = M_{xx} \cdot \delta_s^{kl} \cdot s(t) \quad (\text{D4})$$

$$\mathcal{M}_{yy}^{kl}(t) = M_{yy} \cdot \delta_s^{kl} \cdot s(t) \quad (\text{D5})$$

$$\mathcal{M}_{xy}^{kl}(t) = M_{xy} \cdot \delta_s^{kl} \cdot s(t) \quad (\text{D6})$$

where we take the delta function as

$$\delta_s^{kl} \stackrel{\text{def}}{=} (\delta_s^{kl})_{ij} = \iint \phi_{ij}^{kl}(\mathbf{x}) \delta(\mathbf{x} - \mathbf{x}_s) d\mathbf{x} d\mathbf{y} = \phi_{ij}^{kl}(\mathbf{x}_s). \quad (\text{D7})$$

In the examples, the source time function used is the Ricker wavelet in eq. (33).

APPENDIX E: COMPUTATION OF THE PARTIAL DERIVATIVES AND COMMUNICATION OF THE BOUNDARY VALUES

The partial derivatives of the displacement components needed to evaluate the right-hand side of eq. (29) are:

$$\mathbb{D}_x^{11}(\mathbf{U}_x^{21}) = \mathbf{D}_x^{12} \cdot \mathbf{U}_x^{21} - \mathbf{b}_x^1 \cdot \mathbf{u}_x^{1T}(x'_-) + \mathbf{d}_x^1 \cdot \mathbf{u}_x^{1T}(x'_+) \quad (\text{E1a})$$

$$\mathbb{D}_y^{11}(\mathbf{U}_x^{12}) = \mathbf{U}_x^{12} \cdot \mathbf{D}_y^{12T} - \mathbf{u}_x^1(y'_-) \cdot \mathbf{b}_y^{1T} + \mathbf{u}_x^1(y'_+) \cdot \mathbf{d}_y^{1T} \quad (\text{E1b})$$

$$\mathbb{D}_x^{11}(\mathbf{U}_y^{21}) = \mathbf{D}_x^{12} \cdot \mathbf{U}_y^{21} - \mathbf{b}_x^1 \cdot \mathbf{u}_y^{1T}(x'_-) + \mathbf{d}_x^1 \cdot \mathbf{u}_y^{1T}(x'_+) \quad (\text{E1c})$$

$$\mathbb{D}_y^{11}(\mathbf{U}_y^{12}) = \mathbf{U}_y^{12} \cdot \mathbf{D}_y^{12T} - \mathbf{u}_y^1(y'_-) \cdot \mathbf{b}_y^{1T} + \mathbf{u}_y^1(y'_+) \cdot \mathbf{d}_y^{1T} \quad (\text{E1d})$$

$$\mathbb{D}_x^{22}(\mathbf{U}_x^{12}) = \mathbf{D}_x^{21} \cdot \mathbf{U}_x^{12} - \mathbf{b}_x^2 \cdot \mathbf{u}_x^{2T}(x'_-) + \mathbf{d}_x^2 \cdot \mathbf{u}_x^{2T}(x'_+) \quad (\text{E1e})$$

$$\mathbb{D}_y^{22}(\mathbf{U}_x^{21}) = \mathbf{U}_x^{21} \cdot \mathbf{D}_y^{21T} - \mathbf{u}_x^2(y'_-) \cdot \mathbf{b}_y^{2T} + \mathbf{u}_x^2(y'_+) \cdot \mathbf{d}_y^{2T} \quad (\text{E1f})$$

$$\mathbb{D}_x^{22}(\mathbf{U}_y^{12}) = \mathbf{D}_x^{21} \cdot \mathbf{U}_y^{12} - \mathbf{b}_x^2 \cdot \mathbf{u}_y^{2T}(x'_-) + \mathbf{d}_x^2 \cdot \mathbf{u}_y^{2T}(x'_+) \quad (\text{E1g})$$

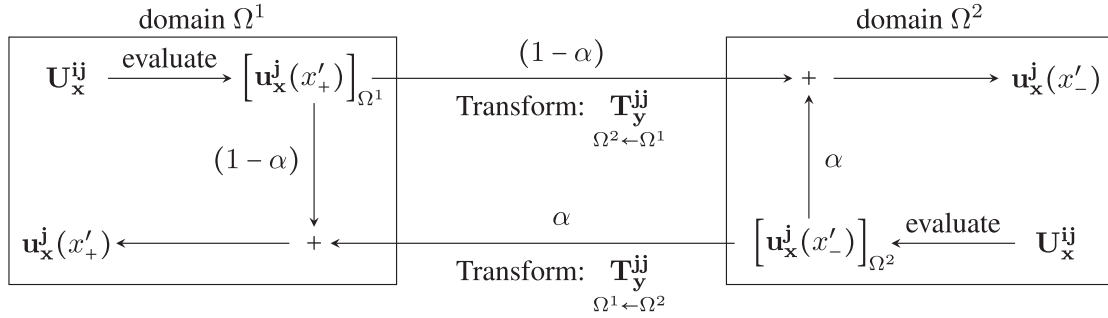
$$\mathbb{D}_y^{22}(\mathbf{U}_y^{21}) = \mathbf{U}_y^{21} \cdot \mathbf{D}_y^{21T} - \mathbf{u}_y^2(y'_-) \cdot \mathbf{b}_y^{2T} + \mathbf{u}_y^2(y'_+) \cdot \mathbf{d}_y^{2T}, \quad (\text{E1h})$$

where the 2-D arrays $\mathbf{U}_\theta^{ij}$ contain the expansion coefficients representing the displacements inside the current domain Ω and the 1-D arrays $\mathbf{u}_\theta^j()$ contain the expansion coefficients representing the displacements at the boundaries of the current domain Ω . We compute the boundary values in eq. (E1) using

$$\mathbf{u}_\theta^j(x'_-) = (\alpha_{x'_-}) \left[\mathbf{u}_\theta^j(x'_-) \right]_\Omega + (1 - \alpha_{x'_-}) \left[\mathbf{T}_y^{ij} \cdot \left[\mathbf{u}_\theta^j(x'_+) \right]_{\Omega^{x'_-}} \right]_{\Omega \leftarrow \Omega^{x'_-}} \quad (\theta = x, y; j = 1, 2) \quad (\text{E2a})$$

$$\mathbf{u}_\theta^j(x'_+) = (\alpha_{x'_+}) \left[\mathbf{u}_\theta^j(x'_+) \right]_\Omega + (1 - \alpha_{x'_+}) \left[\mathbf{T}_y^{ij} \cdot \left[\mathbf{u}_\theta^j(x'_-) \right]_{\Omega^{x'_+}} \right]_{\Omega \leftarrow \Omega^{x'_+}} \quad (\theta = x, y; j = 1, 2) \quad (\text{E2b})$$

(a) Exchange procedure for the displacements' boundary values



(b) Exchange procedure for the stresses' boundary values

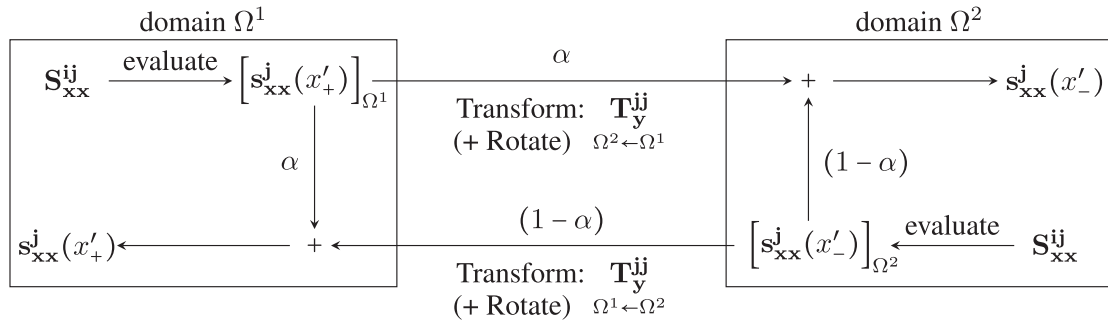


Figure A2. (a) Workflow for computing and exchanging the boundary values between two neighbouring domains Ω^1 and Ω^2 . Here, we consider the case where the right face of domain Ω^1 coincides with the left face of domain Ω^2 . In both domains, we first evaluate the displacement $\mathbf{U}_x^{ij}$ at the boundary using eq. (E3). We then send the boundary values $[\mathbf{u}_x^j(x'_+)]_{\Omega^1}$ to the neighbouring domain. During the exchange, the transformation matrix $\mathbf{T}_y^{ij}$ is applied to obtain the boundary values in the basis of the receiving domain. Finally, the internal and external boundary values are weighted (according to the weights on the arrows) and summed to obtain the averaged boundary value in eq. (E2). Panel (b) is similar to (a) but for the stresses. The steps are the same except the following: (1) the weights are complementary to those used for the displacement (this is needed to obtain a stable scheme). (2) If the axes of the computational domain are not oriented similarly in the two domains, the components of the transformed stress need to be rotated.

$$\mathbf{u}_\theta^i(y'_-) = (\alpha_{y'_-}) [\mathbf{u}_\theta^i(y'_-)]_\Omega + (1 - \alpha_{y'_-}) \mathbf{T}_x^{ii} \cdot [\mathbf{u}_\theta^i(y'_+)]_{\Omega^{y'_-}} \quad (\theta = x, y; i = 1, 2) \quad (\text{E2c})$$

$$\mathbf{u}_\theta^i(y'_+) = (\alpha_{y'_+}) [\mathbf{u}_\theta^i(y'_+)]_\Omega + (1 - \alpha_{y'_+}) \mathbf{T}_x^{ii} \cdot [\mathbf{u}_\theta^i(y'_-)]_{\Omega^{y'_+}} \quad (\theta = x, y; i = 1, 2), \quad (\text{E2d})$$

where the internal boundary values $[\mathbf{u}_\theta^i(0)]_\Omega$ are taken from the current domain Ω and the external boundary values $[\mathbf{u}_\theta^i(0)]_{\Omega^{\theta'_\pm}}$ are taken from the neighbouring domains. The matrices $\mathbf{T}_\theta^{ij}$ perform basis transformations to represent the external boundary values in the same basis as the displacement inside the current domain Ω . The weights $\alpha_{\theta'_\pm}$ control whether the boundary values $\mathbf{u}_\theta^i(0)$ are taken inside ($\alpha_{\theta'_\pm} = 1$), outside ($\alpha_{\theta'_\pm} = 0$) or in between ($0 < \alpha_{\theta'_\pm} < 1$) the current and the neighbouring domains. In eq. (E2), the boundary values are computed in all domains Ω using:

$$[\mathbf{u}_\theta^j(x'_-)]_\Omega = [\mathbf{U}_\theta^{kjT} \cdot \mathbf{b}_x^k]_\Omega \quad (\theta = x, y; j = 1, 2; k = 3 - j) \quad (\text{E3a})$$

$$[\mathbf{u}_\theta^j(x'_+)]_\Omega = [\mathbf{U}_\theta^{kjT} \cdot \mathbf{d}_x^k]_\Omega \quad (\theta = x, y; j = 1, 2; k = 3 - j) \quad (\text{E3b})$$

$$[\mathbf{u}_\theta^i(y'_-)]_\Omega = [\mathbf{b}_x^i \cdot \mathbf{U}_\theta^{il}]_\Omega \quad (\theta = x, y; i = 1, 2; l = 3 - i) \quad (\text{E3c})$$

$$[\mathbf{u}_\theta^i(y'_+)]_\Omega = [\mathbf{d}_x^i \cdot \mathbf{U}_\theta^{il}]_\Omega \quad (\theta = x, y; i = 1, 2; l = 3 - i). \quad (\text{E3d})$$

The workflow for computing and exchanging the boundary values according to eq. (E2) is sketched in Fig. A2.

The partial derivatives of the stress components need to evaluate the right-hand side of eq. (31) are:

$$\mathbb{D}_{xx}^{21}(\mathbf{S}_{xx}^{11}) = \mathbf{D}_x^{21} \cdot \mathbf{S}_{xx}^{11} - \mathbf{b}_x^2 \cdot \mathbf{s}_{xx}^{1T}(x'_-) + \mathbf{d}_x^2 \cdot \mathbf{s}_{xx}^{1T}(x'_+) \quad (\text{E4a})$$

$$\mathbb{D}_Y^{21}(\mathbf{S}_{xy}^{22}) = \mathbf{S}_{xy}^{22} \cdot \mathbf{D}_y^{12T} - \mathbf{s}_{xy}^2(y'_-) \cdot \mathbf{b}_y^{1T} + \mathbf{s}_{xy}^2(y'_+) \cdot \mathbf{d}_y^{1T} \quad (\text{E4b})$$

$$\mathbb{D}_X^{12}(\mathbf{S}_{yx}^{22}) = \mathbf{D}_x^{12} \cdot \mathbf{S}_{yx}^{22} - \mathbf{b}_x^1 \cdot \mathbf{s}_{yx}^{2T}(x'_-) + \mathbf{d}_x^1 \cdot \mathbf{s}_{yx}^{2T}(x'_+) \quad (\text{E4c})$$

$$\mathbb{D}_Y^{12}(\mathbf{S}_{yy}^{11}) = \mathbf{S}_{yy}^{11} \cdot \mathbf{D}_y^{21T} - \mathbf{s}_{yy}^1(y'_-) \cdot \mathbf{b}_y^{2T} + \mathbf{s}_{yy}^1(y'_+) \cdot \mathbf{d}_y^{2T} \quad (\text{E4d})$$

$$\mathbb{D}_X^{12}(\mathbf{S}_{xx}^{22}) = \mathbf{D}_x^{12} \cdot \mathbf{S}_{xx}^{22} - \mathbf{b}_x^1 \cdot \mathbf{s}_{xx}^{2T}(x'_-) + \mathbf{d}_x^1 \cdot \mathbf{s}_{xx}^{2T}(x'_+) \quad (\text{E4e})$$

$$\mathbb{D}_Y^{12}(\mathbf{S}_{xy}^{11}) = \mathbf{S}_{xy}^{11} \cdot \mathbf{D}_y^{21T} - \mathbf{s}_{xy}^1(y'_-) \cdot \mathbf{b}_y^{2T} + \mathbf{s}_{xy}^1(y'_+) \cdot \mathbf{d}_y^{2T} \quad (\text{E4f})$$

$$\mathbb{D}_X^{21}(\mathbf{S}_{yx}^{11}) = \mathbf{D}_x^{21} \cdot \mathbf{S}_{yx}^{11} - \mathbf{b}_x^2 \cdot \mathbf{s}_{yx}^{1T}(x'_-) + \mathbf{d}_x^2 \cdot \mathbf{s}_{yx}^{1T}(x'_+) \quad (\text{E4g})$$

$$\mathbb{D}_Y^{21}(\mathbf{S}_{yy}^{22}) = \mathbf{S}_{yy}^{22} \cdot \mathbf{D}_y^{12T} - \mathbf{s}_{yy}^2(y'_-) \cdot \mathbf{b}_y^{1T} + \mathbf{s}_{yy}^2(y'_+) \cdot \mathbf{d}_y^{1T}, \quad (\text{E4h})$$

where the 2-D arrays $\mathbf{s}_{\theta\theta}^i$ contain the expansion coefficients representing the stresses inside the current domain Ω and the 1-D arrays $\mathbf{s}_{\theta\theta}^i()$ contain the expansion coefficients representing the stresses at the boundaries of the current domain Ω . We compute the boundary values in eq. (E4) using:

$$\mathbf{s}_{\theta\theta}^j(x'_-) = (1 - \alpha_{x'_-}) \left[\mathbf{s}_{\theta\theta}^j(x'_-) \right]_{\Omega} + (\alpha_{x'_-}) \mathbf{T}_{\theta}^{ij} \cdot \left[\mathbf{s}_{\theta\theta}^j(x'_+) \right]_{\Omega^{x'_-}} \quad \begin{cases} \theta\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy} \\ \mathbf{j} = 1, 2 \end{cases} \quad (\text{E5a})$$

$$\mathbf{s}_{\theta\theta}^j(x'_+) = (1 - \alpha_{x'_+}) \left[\mathbf{s}_{\theta\theta}^j(x'_+) \right]_{\Omega} + (\alpha_{x'_+}) \mathbf{T}_{\theta}^{ij} \cdot \left[\mathbf{s}_{\theta\theta}^j(x'_-) \right]_{\Omega^{x'_+}} \quad \begin{cases} \theta\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy} \\ \mathbf{j} = 1, 2 \end{cases} \quad (\text{E5b})$$

$$\mathbf{s}_{\theta\theta}^i(y'_-) = (1 - \alpha_{y'_-}) \left[\mathbf{s}_{\theta\theta}^i(y'_-) \right]_{\Omega} + (\alpha_{y'_-}) \mathbf{T}_{\theta}^{ii} \cdot \left[\mathbf{s}_{\theta\theta}^i(y'_+) \right]_{\Omega^{y'_-}} \quad \begin{cases} \theta\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy} \\ \mathbf{i} = 1, 2 \end{cases} \quad (\text{E5c})$$

$$\mathbf{s}_{\theta\theta}^i(y'_+) = (1 - \alpha_{y'_+}) \left[\mathbf{s}_{\theta\theta}^i(y'_+) \right]_{\Omega} + (\alpha_{y'_+}) \mathbf{T}_{\theta}^{ii} \cdot \left[\mathbf{s}_{\theta\theta}^i(y'_-) \right]_{\Omega^{y'_+}} \quad \begin{cases} \theta\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy} \\ \mathbf{i} = 1, 2 \end{cases} \quad (\text{E5d})$$

where the internal boundary values $\left[\mathbf{s}_{\theta\theta}^i() \right]_{\Omega}$ are taken from the current domain Ω and the external boundary values $\left[\mathbf{s}_{\theta\theta}^i() \right]_{\Omega^{\theta'_{\pm}}}$ are taken from the neighbouring domains. The matrices $\mathbf{T}_{\theta}^{ij}$ perform basis transformations to represent the external boundary values in the same basis as the displacement inside the current domain Ω . The weight $\alpha_{\theta'_{\pm}}$ controls whether the boundary values $\mathbf{s}_{\theta\theta}^i()$ are taken inside ($\alpha_{\theta'_{\pm}} = 0$), outside ($\alpha_{\theta'_{\pm}} = 1$) or in between ($0 < \alpha_{\theta'_{\pm}} < 1$) the current and the neighbouring domains. In eq. (E5), the boundary values are computed in all domains Ω using:

$$\left[\mathbf{s}_{\theta\theta}^j(x'_-) \right]_{\Omega} = \left[\mathbf{s}_{\theta\theta}^{kjT} \cdot \mathbf{b}_x^k \right]_{\Omega} \quad (\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy}; \mathbf{j} = 1, 2; \mathbf{k} = 3 - \mathbf{j}) \quad (\text{E6a})$$

$$\left[\mathbf{s}_{\theta\theta}^j(x'_+) \right]_{\Omega} = \left[\mathbf{s}_{\theta\theta}^{kjT} \cdot \mathbf{d}_x^k \right]_{\Omega} \quad (\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy}; \mathbf{j} = 1, 2; \mathbf{k} = 3 - \mathbf{j}) \quad (\text{E6b})$$

$$\left[\mathbf{s}_{\theta\theta}^i(y'_-) \right]_{\Omega} = \left[\mathbf{b}_x^{iT} \cdot \mathbf{S}_{\theta\theta}^{il} \right]_{\Omega} \quad (\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy}; \mathbf{i} = 1, 2; \mathbf{l} = 3 - \mathbf{i}) \quad (\text{E6c})$$

$$\left[\mathbf{s}_{\theta\theta}^i(y'_+) \right]_{\Omega} = \left[\mathbf{d}_x^{iT} \cdot \mathbf{S}_{\theta\theta}^{il} \right]_{\Omega} \quad (\theta = \mathbf{xx}, \mathbf{yy}, \mathbf{xy}; \mathbf{i} = 1, 2; \mathbf{l} = 3 - \mathbf{i}). \quad (\text{E6d})$$

Finally, when the parametric coordinates do not have the same orientation in the two domains, one needs to rotate the boundary values of the transformed stresses. For example, using

$$\begin{bmatrix} [\mathbf{s}_{xx}^{kl}]_{i,j} & [\mathbf{s}_{xy}^{kl}]_{i,j} \\ [\mathbf{s}_{xy}^{kl}]_{i,j} & [\mathbf{s}_{yy}^{kl}]_{i,j} \end{bmatrix} \leftarrow \begin{bmatrix} [\mathbf{s}_{xx}^{kl}]_{i,j} & [\mathbf{s}_{xy}^{kl}]_{i,j} \\ [\mathbf{s}_{xy}^{kl}]_{i,j} & [\mathbf{s}_{yy}^{kl}]_{i,j} \end{bmatrix} \cdot \frac{1}{|\mathbf{J}|} \cdot \mathbf{J}, \quad (\text{E7})$$

where $\mathbf{J}$ is the Jacobian matrix that aligns the parametric coordinates in the two domains.

Following Masson (2022), we set the coefficients $\alpha_{\theta'_{\pm}}$ in eqs (E2) and (E5) depending on the nature of the interfaces. We have:

- (i) $\alpha_{\theta'_{\pm}} = 1$ at all interfaces surrounding the computational domain to model a free surface.
- (ii) $\alpha_{\theta'_{\pm}} = 1/2$ at all interfaces between domains that are both solid/elastic or fluid/acoustic.
- (iii) $\alpha_{\theta'_{\pm}} = 1$ inside the solid/elastic domain and $\alpha_{\theta'_{\pm}} = 0$ inside the fluid/acoustic domain, at all solid–fluid interfaces.