A Novel Immersed Boundary Approach for Irregular Topography with Acoustic Wave Equations

Edward Caunt (edward.caunt15@imperial.ac.uk)*, Rhodri Nelson (rhodri.nelson@imperial.ac.uk)*, Tim MacArthur (tim@tbi.ca)† Fabio Luporini (fabio@devitocodes.com)‡, Gerard Gorman (g.gorman@imperial.ac.uk)*

**Imperial College London, Department of Earth Science and Engineering, London, UK* †*Thrust Belt Imaging, Calgary, Canada* ‡*Devito Codes, London, UK*

(March 13, 2024)

Running head: **Immersed boundary topography approach**

ABSTRACT

Irregular terrain has a pronounced effect on the propagation of seismic and acoustic wavefields but is not straightforwardly reconciled with structured finite-difference (FD) methods used to model such phenomena. Accurate wavefield simulation is paramount in subsurface imaging applications such as reverse-time migration (RTM) and full-waveform inversion (FWI), requiring suitable topography handling. Methods currently detailed in the literature are generally limited in scope application-wise or non-trivial to apply to real-world geometries. A general immersed boundary treatment capable of imposing a range of boundary conditions in a relatively

equation-agnostic manner has been developed, alongside a framework implementing this approach to complement emerging code-generation paradigms. The approach is distinguished by the use of N-dimensional Taylor-series extrapolants constrained by boundary conditions imposed at some suitably distributed set of surface points. The extrapolation process is encapsulated in modified derivative stencils applied in the vicinity of the boundary, utilizing hyperspherical support regions. This method ensures boundary representation is consistent with the FD discretization. Furthermore, high-dimensional and vector boundary conditions can be applied without approximation prior to discretization. A consistent methodology can thus be applied across free and rigid surfaces with first and second-order acoustic wave equation formulations. Application to both equations is demonstrated, and numerical examples based on analytic and real-world topography implementing free and rigid surfaces in 2D and 3D are presented. Numerical examples and convergence tests demonstrate the accuracy of boundary treatments devised by the prescribed approach, their suitability to practical applications, and the feasibility of automatically generating treatments to suit each case.

INTRODUCTION

Irregular topography introduces considerable complexity to geophysical wavefield models, creating complex path effects which turn clean, defined reflections into cascades of overlapping arrivals. Interactions with topography cause waves to diffract and scatter, focus and defocus (Takemura et al., 2015; Reinoso et al., 1996; Boore, 1972; Griffiths and Bollinger, 1979): effects that must be encapsulated by wavefield simulations (Borisov et al., 2018). Whilst early numerical experiments demonstrated topography's marked effect upon recorded data (Boore, 1973), the proliferation of wave-equation-based workflows, most notably full-waveform inversion (FWI) and reverse-time migration (RTM), has placed a sharpened focus on understanding topographic effects.

Topographic effects have been explored in contexts including FWI (Bleibinhaus and Rondenay, 2009), understanding seismic wave scattering (Takemura et al., 2015), infrasound propagation problems (Kim and Lees, 2014; Fee et al., 2021), and is crucial for the emerging field of teleseismic FWI (Monteiller et al., 2013, 2015). It is recognised that poor topography implementation adversely affects model accuracy (Monteiller et al., 2013; Li and Yao, 2020; Borisov et al., 2018), often severely (Nuber et al., 2016) although this may be acceptable in some applications (Bleibinhaus and Rondenay, 2009). When applied to imaging problems, unrealistic propagation paths result in artefacts in the processed image (Bleibinhaus and Rondenay, 2009; Nuber et al., 2016; Borisov et al., 2018) which risk being mistaken for geological features, whilst models of ground motion have been found to significantly underestimate local

amplification factors if topography is omitted (Reinoso et al., 1996).

Unstructured meshes suitably conformed to the topography offer an immediate solution. Finite-element methods (FEMs) have been demonstrated as effective for modelling wave propagation in the presence of variable topography (Zhebel et al., 2014; Dupros et al., 2010; Borisov et al., 2018; Liu et al., 2014; Mulder and Shamasundar, 2016), and have been applied to include complex terrain in models of earthquake peak ground acceleration (Galis et al., 2008), in shallow structure characterization applications (Romdhane et al., 2011), and to imaging problems such as full waveform inversion (FWI) (Roberts et al., 2021b; Shin et al., 2013; Monteiller et al., 2015). Unstructured finite-difference methods (FDMs) have also been developed (Takekawa et al., 2015; Martin et al., 2015), and successfully used to perform FWI on synthetic datasets (Wu et al., 2021).

However, structured FD methods dominate in geophysical wavefield modelling and processing applications, including seismic reverse-time migration (RTM) (Fletcher et al., 2009), full-waveform inversion (FWI) (Warner et al., 2013), and source localisation (Mckee et al., 2014). They eschew potentially cumbersome grid-generation algorithms (Brehm and Fasel, 2013), which become increasingly problematic at large scales (Slotnick et al., 2014) and require *a priori* knowledge of seismic velocities (Roberts et al., 2021a,b). This information is potentially unavailable in practice or is iteratively updated in outer-loop problems, necessitating repeated mesh adaptation. Furthermore, the geometry of geological structures is prone to generating ill-conditioned sliver elements as units get pinched out resulting in an unreasonably small

timestep (Roberts et al., 2021b). Automated mesh generation for seismic wave propagation in models containing arbitrary horizons remains an open problem; unstructured approaches are rarely used in production. Structured FD methods are simple to implement, with relatively-low computational footprints (Liu and Sen, 2009), even before considering the suite of known optimizations (Luporini et al., 2020; Louboutin et al., 2019). However, in the presence of irregular topography, accurately representing the sharp, uneven material discontinuity on regular grids can be problematic (Zeng et al., 2012; Mulder, 2017; Gao et al., 2015). Ideally, one would want to accurately represent complex, curvilinear topography as a sharp interface whilst retaining the advantages of structured grids.

Air- or vacuum-layer approaches, achieved through a low-impedence layer at the surface (Schultz, 1997; Zeng et al., 2012) trade accuracy for simplicity in mimicking the free surface. Satisfactory results can be obtained with careful implementation (Zeng et al., 2012), although error analyses and numerical experiments indicate sub-second-order convergence in space, regardless of interior discretization (Zhebel et al., 2014; Symes and Vdovina, 2009) and potentially egregious (Graves, 1996; Zahradnik et al., 1993) error when applied to complex geometries. Suppression of spurious scattering requires heavy oversampling (Bohlen and Saenger, 2006), and smoothing or stabilisation routines are often required to stabilise the surface treatment (Bartel et al., 2000; Zeng et al., 2012; Vieira et al., 2018). Staircased image methods marginally improve on this (Robertsson, 1996): numerous variations have been explored (Robertsson, 1996; Boore, 1972; Hayashi et al., 2001; Ohminato and Chouet,

1997; Ripperger et al., 2003), with practical applications including macro-scale topographic scattering effects (Takemura et al., 2015; Nakamura et al., 2012) and volcanic source location (Kim and Lees, 2014; Fee et al., 2021). The stepped boundary still generates diffractions (Muir et al., 1992; Hu, 2016), and a time shift proportional to the difference between the interface and computational grids (Symes and Vdovina, 2009). Whilst staircased representations can be improved with locally-refined subgrids (Oprsal and Zahradnik, 1999; Tavelli et al., 2019), such refinement is non-trivial (Lai and Peskin, 2000; Goldstein et al., 1993), often requiring careful filtering and interpolation routines (Zhang et al., 2013b). A more accurate image method can be achieved via coordinate transform (Petersson and Sjögreen, 2015) such that the surface is a horizontal plane in the iteration space (Zhang and Chen, 2006; Hestholm and Ruud, 2002). This approach is widely adaptable, and various wave equations have been solved with such schemes (Zhang and Chen, 2006; Zhang et al., 2012; Hestholm and Ruud, 1994; Hestholm et al., 1999; Hestholm and Street, 2000; Sun et al., 2017; de la Puente et al., 2014), demonstrating a high degree of accuracy, on par with FEMs (Zhang et al., 2012). However, a smooth conformal mapping may be challenging to obtain with grid skewness and roughness potentially introducing errors and instability (Shragge and Konuk, 2020). This can be addressed with the use of mimetic FDMs (de la Puente et al., 2014; Shragge and Tapley, 2017; Shragge and Konuk, 2020), which have the additional advantage of enabling accurate and stable imposition of the free surface condition (de la Puente et al., 2014; Shragge and Tapley, 2017; Shragge and Konuk, 2020). However, such approaches do introduce considerable computational overhead and the presence of locally small cells remains an issue, potentially limiting

the maximum timestep (de la Puente et al., 2014; Shragge, 2014; Shragge and Konuk, 2020), similar to the aforementioned sliver elements in FEM.

An alternative approach, popular in computational fluid dynamics (CFD) contexts (Mittal et al., 2008; Seo and Mittal, 2011; Dong et al., 2010; Vargas et al., 2008), is the immersed boundary method. This approach embeds a curvilinear interface within a Cartesian grid by locally modifying the FD operators in the vicinity of the boundary (Brehm and Fasel, 2013; Brehm et al., 2015; Mulder, 2017). Several immersed boundary variations for second-order acoustic wave equation formulations (Zhang et al., 2013a; Mulder, 2017; Li and Yao, 2020) have been proposed, with some extending to first-order formulations (Mulder and Huiskes, 2017; Hu, 2016), and even the isotropic elastic wave equation with some success (Lombard et al., 2008; Almuhaidib and Toksoz, 2015; Gao et al., 2015). The standard Cartesian grid and equations are retained and complex geometries including sharp edges and concavities can be represented.

To date, most applications of immersed boundary schemes in geophysical wave modelling scenarios have made use of problem-specific approximations prior to discretization to impose boundary conditions, typically considering a 1D version of the condition normal to the boundary. Such approximations yield satisfactory results (Hu, 2016; Mulder, 2017; Almuhaidib and Toksoz, 2015; Li et al., 2022) and their usage is widespread. Whilst these approximations offer intuitive parallels with conventional image methods, the condition imposed may not equal the true boundary condition as resolution goes to zero. This is particularly apparent for vector boundary conditions

such as those applied to the velocity fields at the free surface. Hu (2016) addressed this by decomposing velocity field components into tangential and radial components in a local cylindrical coordinate system, whilst Mulder and Huiskes (2017) assumed locally vertical or horizontal boundaries, incurring numerical error.

Such approximations are reliant upon domain knowledge and do not necessarily generalise straightforwardly; it is not immediately clear as to their applicability across multiple wave equations. Indeed, these approximations may be inherently limiting, precluding adaption to more complex physics (wave equations featuring transverse isotropy or elastic wave equations for example). Given the trajectory towards increasingly complex physics and geometries, this paper sets out to develop a generalised immersed boundary approach, such that a consistent methodology can be applied in a relatively problem-agnostic manner.

This approach aligns with current trends in geophysical modelling: domain-specific languages (DSLs) and automatic code generation are increasingly prevalent, with projects such as Devito (Luporini et al., 2020; Louboutin et al., 2019) and Firedrake (Rathgeber et al., 2016) leveraging high-level abstractions to generate low-level FDM and FEM solver kernels from symbolic partial differential equations (PDEs). Abstracting low-level aspects of implementation achieves a separation of concerns between the overarching problem and its underlying implementation (Rathgeber et al., 2016), enabling domain specialists to focus on the application level. High-level interfaces substantially reduce development time (Louboutin et al., 2019), whilst sophisticated optimization routines in the lowering process yield high-performance, portable

code (Luporini et al., 2020; Louboutin et al., 2019). The ability to generate a low-level implementation from a high-level specification hinges on a general method for solving a given problem so that suitable abstractions can be developed. This cannot be easily achieved with the application-specific immersed boundary implementations developed to date, as they are generally not designed with generalisation across multiple problems in mind.

Developing a generalised mathematical approach enables abstraction, which in turn lends itself to automation and thus code generation. Furthermore, direct generalisation to other wave equations widens potential application, spanning a range of geophysical problems. The generalised nature of the method presented greatly simplifies inclusion of immersed boundaries in geophysical models, as a suitable treatment can be devised according to the prescribed formula to suit the physical problem at hand.

An immersed boundary approach supporting the imposition of multidimensional and vector boundary conditions is outlined, followed by its application to a selection of equations and boundary conditions of interest. This mathematical approach forms the backbone of Schism: a plugin for Devito used to implement the examples shown in this paper (Caunt, 2024). The numerical accuracy of the approach proposed is explored through convergence testing of the resultant treatment, and several geophysically-relevant test cases based on the first and second-order formulations of the acoustic wave equation are explored. Note that whilst the demonstrations within this paper focus solely on variants of the acoustic wave equation, the method

is nominally equally suited to pseudoacoustic wave equations featuring vertical and tilted transverse isotropy (VTI and TTI), alongside elastic wave equations. Further exploration of these areas is planned in subsequent publications.

THEORY

Constructing finite-difference approximations

FD methods approximate the evolution of PDE solutions by calculating derivatives of a basis function fitted to discrete function values, using these to estimate the updated values at these discrete points at the next timestep. These derivative approximations contain some error relative to their true value, diminishing as resolution increases. To outline how these approximations are made in the typical case, consider some function f in a 1D space; approximating with an Mth-order Taylor series expansion at some point x_0 yields

$$f(x) = \sum_{m=0}^{M} \frac{(x - x_0)^m}{m!} \frac{\partial^m f(x_0)}{\partial x^m} + O((x - x_0)^{M+1}), \tag{1}$$

concisely represented as

$$\mathbf{a} \cdot \boldsymbol{\delta} = f(\boldsymbol{x}), \tag{2}$$

where

$$\mathbf{a}^T = \left(1, (x - x_0), ..., \frac{(x - x_0)^M}{M!}\right), \tag{3}$$

and

$$\boldsymbol{\delta}^T = \left(f(x_0), \frac{\partial f}{\partial x}(x_0), ..., \frac{\partial^M f}{\partial x^M}(x_0)\right). \tag{4}$$

For some even M, expansions at $M+1$ discrete points, labelled $x_{-M/2}$ through $x_{M/2}$, enable the formation of the linear system

$$\mathbf{A}\boldsymbol{\delta} = \begin{pmatrix} f(x_{-M/2}) \\ \vdots \\ f(x_{M/2}) \end{pmatrix}. \tag{5}$$

To illustrate, for $M = 2$, the system is as follows

$$\begin{pmatrix} 1 & x_{-1} - x_0 & \frac{(x_{-1}-x_0)^2}{2} \\ 1 & 0 & 0 \\ 1 & x_1 - x_0 & \frac{(x_1-x_0)^2}{2} \end{pmatrix} \begin{pmatrix} f(x_0) \\ \frac{\partial f}{\partial x}(x_0) \\ \frac{\partial^2 f}{\partial x^2}(x_0) \end{pmatrix} = \begin{pmatrix} f(x_{-1}) \\ f(x_0) \\ f(x_1) \end{pmatrix}, \tag{6}$$

which can be rearranged in the form

$$\mathbf{A}^{-1} \begin{pmatrix} f(x_{-M/2}) \\ \vdots \\ f(x_{M/2}) \end{pmatrix} = \boldsymbol{\delta}, \tag{7}$$

thereby obtaining derivatives as some weighted sum of discrete function values: a stencil. With the assumption of a regular grid, relative positions of points become fixed regardless of where the derivative is being taken, yielding

$$\begin{pmatrix} 1 & -1 & \frac{1}{2} \\ 1 & 0 & 0 \\ 1 & 1 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} f(x_0) \\ \Delta x \frac{\partial f}{\partial x}(x_0) \\ \Delta x^2 \frac{\partial^2 f}{\partial x^2}(x_0) \end{pmatrix} = \begin{pmatrix} f(x_{-1}) \\ f(x_0) \\ f(x_1) \end{pmatrix}, \tag{8}$$

for the above case, the inverse being

$$\begin{pmatrix} 0 & 1 & 0 \\ -\frac{1}{2} & 0 & \frac{1}{2} \\ 1 & -2 & 1 \end{pmatrix} \begin{pmatrix} f(x_{-1}) \\ f(x_0) \\ f(x_1) \end{pmatrix} = \begin{pmatrix} f(x_0) \\ \Delta x \frac{\partial f}{\partial x}(x_0) \\ \Delta x^2 \frac{\partial^2 f}{\partial x^2}(x_0) \end{pmatrix}. \tag{9}$$

The leftmost matrix contains weights for FD stencils of every derivative order up to that of the basis.

For higher dimensions, a suitably higher-dimensional polynomial is required, formed as a product of per-dimension 1D series of the form

$$f(x) = \sum_{m=0} a_m x^m, \tag{10}$$

in 3D yielding

$$f(\boldsymbol{x}) = \sum_{m=0}\sum_{n=0}\sum_{l=0} a_{mnl} x^m y^n z^l. \tag{11}$$

Given this, the process outlined above is extensible to higher dimensions as in Takekawa et al. (2015). In the pursuit of consistency and accuracy, it is crucial to maintain consistent error throughout any numerical scheme; polynomials used in such schemes must be of an order matching the spatial discretization employed. For an order M spatial discretization, the 3D expansion is truncated as follows:

$$f(\boldsymbol{x}) = \sum_{m=0}^{M_x}\sum_{n=0}^{M_y}\sum_{l=0}^{M_z} a_{mnl} x^m y^n z^l + O(|\Delta \boldsymbol{x}|^{(M+1)}), \tag{12}$$

subject to

$$M_x + M_y + M_z = M, \tag{13}$$

removing any terms with order greater than M.

An N-dimensional Taylor series is a robust candidate for such applications, retaining similarities to the aforementioned 1D case and being suitably problem-agnostic. In 3D, this is given by

$$f(\boldsymbol{x}) = \sum_{m=0}\sum_{n=0}\sum_{l=0} \frac{(x-x_0)^m (y-y_0)^n (z-z_0)^l}{m!n!l!} \delta_{m,n,l}, \tag{14}$$

where

$$\delta_{m,n,l} = \frac{\partial^{m+n+l}}{\partial x^m \partial y^n \partial z^l} f(\boldsymbol{x}_0). \tag{15}$$

Truncating this expansion as in Equation 13, the polynomial expansion at points within the support region can be represented as a matrix-vector multiplication, as in Equation 2. Reflecting the higher dimensionality and corresponding increased number of terms, $\mathbf{a}$ is given by

$$\begin{aligned}\mathbf{a}^T = \Bigg(1, (x-x_0), ..., (z-z_0), \frac{(x-x_0)^2}{2}, (x-x_0)(y-y_0), ..., \\ \frac{(z-z_0)^2}{2}, ..., \frac{(x-x_0)^M}{M!}, ..., \frac{(z-z_0)^M}{M!}\Bigg),\end{aligned} \tag{16}$$

whilst

$$\boldsymbol{\delta}^T = \left(1, \frac{\partial}{\partial x}, ..., \frac{\partial}{\partial z}, \frac{\partial^2}{\partial x^2}, \frac{\partial^2}{\partial x \partial y}, ..., \frac{\partial^2}{\partial z^2}, ..., \frac{\partial^M}{\partial x^M}, ..., \frac{\partial^M}{\partial z^M}\right) f(\boldsymbol{x}_0). \tag{17}$$

As in the 1D case, expansions taken at some set of points are used to form a linear system. However, there is increased flexibility regarding the distribution of these points. Considering some arbitrarily-distributed set of function values, in 1D it is straightforward to construct a polynomial expansion of any desired order to fit this data, provided the number of linearly-independent data points equals or exceeds the coefficients in the truncated expansion. However, when constraining the higher-dimensional basis, this linear independence requires suitable point distribution.

A hyperspherical support region footprint, similar to that used by Takekawa et al. (2015), is proposed as a suitable choice for a large range of boundary conditions. Extensible to any number of dimensions, this choice of support region has a defined centre and can be straightforwardly expanded and contracted without introducing

drastic behavioural changes. Furthermore, any given number of points within the support region will be located as close to the stencil centre as possible. Whilst this work limits itself to Cartesian structured grids, such support regions are nominally equally suited to structured and unstructured data as no assumption is required regarding the underlying data structure. In the structured case, points can be located by index, removing the need for nearest-neighbour searches to to construct the support region. Support regions of various radii are shown in Figure 1: a radius of $(M+1)/2\Delta x$ ensures sufficient information to constrain a polynomial basis on a regular, structured grid.

Whilst extensible to any number of dimensions, the proceeding example considers the 2D case for clarity and brevity. Correspondence between a 2D support region for a 4th-order discretization and $\mathbf{A}$ is shown in Figure 2. The linear system

$$\mathbf{A}\boldsymbol{\delta} = f(\mathbf{X}), \tag{18}$$

where

$$\begin{aligned}\boldsymbol{\delta}^T = \Bigg(&1, \Delta x\frac{\partial}{\partial x}, \Delta y\frac{\partial}{\partial y}, \Delta x^2\frac{\partial^2}{\partial x^2}, \Delta x\Delta y\frac{\partial^2}{\partial x\partial y}, \Delta y^2\frac{\partial^2}{\partial y^2},\\ &\Delta x^3\frac{\partial^3}{\partial x^3}, \Delta x^2\Delta y\frac{\partial^3}{\partial x^2\partial y}, \Delta x\Delta y^2\frac{\partial^3}{\partial x\partial y^2}, \Delta y^3\frac{\partial^3}{\partial y^3},\\ &\Delta x^4\frac{\partial^4}{\partial x^4}, \Delta x^3\Delta y\frac{\partial^4}{\partial x^3\partial y}, \Delta x^2\Delta y^2\frac{\partial^4}{\partial x^2\partial y^2},\\ &\Delta x\Delta y^3\frac{\partial^4}{\partial x\partial y^3}, \Delta y^4\frac{\partial^4}{\partial y^4}\Bigg) f(\boldsymbol{x}_0).\end{aligned} \tag{19}$$

is overdetermined. Inverting this system using a Moore-Penrose pseudoinverse yields expressions for FD stencils using the aforementioned 2D basis.

Introducing boundaries

The evolution of a continuous scalar function, potentially pressure, a component of the particle velocity vector, or even some stress tensor component, is dictated not only by the governing PDE, but boundary conditions imposed. In seismology, these typically demark a half-space, bounded at the top by the ground-air interface. This boundary may truncate derivative operators, leaving requisite stencil points on the exterior, necessitating incorporation of additional constraints to form FD approximations. By imposing suitable conditions on this interface, these constraints can be introduced and the wavefield made to interact with topography in a realistic manner.

Note that the points removed from the stencil correspond to rows removed from $\mathbf{A}$.

Figure 3 shows the effect of a boundary cutting through the support region shown in Figure 2 on the linear system: $\mathbf{A}$ is underdetermined without additional constraints. Applying boundary constraints to the basis can be used to introduce additional rows to $\mathbf{A}$, reducing the number of interior points required and imposing the appropriate behaviour. The process of forming these rows given boundary conditions and the points at which they are imposed follows that of forming polynomial approximations of interior function values.

Approximating boundary conditions

In 3D, linear boundary conditions imposed upon a single field which do not yield the trivial expression $0 = g(\boldsymbol{x_b})$ when approximated with the basis have the form

$$\sum_{m=0}^{M_x}\sum_{n=0}^{M_y}\sum_{l=0}^{M_z}\alpha_{mnl}(\boldsymbol{x_b})\frac{\partial^{m+n+l}f}{\partial x^m\partial y^n\partial z^l}(\boldsymbol{x_b}) = g(\boldsymbol{x_b}) \tag{20}$$

where M_x, M_y, and M_z are as in Equation 13. For applications considered in this paper, coefficients $\alpha_{mnl}(\boldsymbol{x_b})$ are constant.

Substituting f for its Taylor-series approximation yields

$$\sum_{m=0}^{M_x}\sum_{n=0}^{M_y}\sum_{l=0}^{M_z}\sum_{i=0}^{M_x}\sum_{j=0}^{M_y}\sum_{k=0}^{M_z}\beta_{mnlijk}\frac{\partial^{i+j+k}f}{\partial x^i\partial y^j\partial z^k}(\boldsymbol{x}_0) = g(\boldsymbol{x_b}), \tag{21}$$

where

$$\beta_{mnlijk} = \alpha_{mnl}(\boldsymbol{x_b})\frac{\partial^{m+n+l}}{\partial x^m\partial y^n\partial z^l}\frac{(x-x_0)^i(y-y_0)^j(z-z_0)^k}{i!j!k!}. \tag{22}$$

As is the case with Taylor-series approximations at interior points, this is expressible as a dot product:

$$\left(\sum_{m=0}^{M_x}\sum_{n=0}^{M_y}\sum_{l=0}^{M_z}\alpha_{mnl}(\boldsymbol{x_b})\frac{\partial^{m+n+l}}{\partial x^m\partial y^n\partial z^l}\mathbf{a}\right)\cdot\boldsymbol{\delta} = g(\boldsymbol{x_b}). \tag{23}$$

Note that the derivative vector is identical to that used for Taylor-series expansion at interior points, and thus some set of interior and boundary constraints can be encapsulated as the multiplication of the derivative vector and a matrix of coefficients. There is no distinction between interior and boundary constraints: it is apparent that interior constraints can be formed from Equation 20, and both can be represented as rows of the linear system.

This linear system is used to fit the basis, as in Equation 5. Figure 4 shows the effect of adding free-surface boundary conditions to the truncated stencil shown in Figure 3; the matrix is once again overdetermined, and thus nominally has sufficient information to constrain the derivatives. Note however, that it may be the case that such a matrix is still not full-rank. For example, particular boundary constraints may contain redundant information, seen in the lowermost rows of the matrix shown in Figure 4, or lack information regarding particular derivatives.

With interior function values and some J boundary conditions imposed at the boundary points $\mathbf{X_b}$, the linear system

$$\mathbf{A}\boldsymbol{\delta} = \begin{pmatrix} f(\mathbf{X}) \\ g_1(\mathbf{X_b}) \\ \vdots \\ g_J(\mathbf{X_b}) \end{pmatrix}, \tag{24}$$

can be formed, with rows of the form given in Equations 2 and 23, as shown in Figure 4. $\mathbf{X}$ is the set of interior points used to construct the extrapolant and vectors $g_1(\mathbf{X_b})$ through $g_J(\mathbf{X_b})$ contain forcing values corresponding to each boundary condition and forcing point.

As a tangible example, consider constructing a second-order extrapolant in 1D from two interior points (labelled x_1 and x_2) and a boundary point upon which the boundary conditions $f(x_b) = 0$ and $\frac{\partial^2 f}{\partial x^2}(x_b) = 0$ are imposed. This particular case

yields the linear system

$$\begin{pmatrix} 1 & (x_1 - x_0) & \frac{(x_1-x_0)^2}{2} \\ 1 & (x_2 - x_0) & \frac{(x_2-x_0)^2}{2} \\ 1 & (x_b - x_0) & \frac{(x_b-x_0)^2}{2} \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} f(x_0) \\ \frac{\partial f}{\partial x}(x_0) \\ \frac{\partial f^2}{\partial x^2}(x_0) \end{pmatrix} = \begin{pmatrix} f(x_1) \\ f(x_2) \\ 0 \\ 0 \end{pmatrix}. \tag{25}$$

Whilst higher-order accuracy and higher dimensionality increase the size of this system, the process of constructing this system remains the same. Note that removing a single interior point from this example gives the approach detailed in Mulder (2017) and Mulder and Huiskes (2017) in which all boundary constraints are selected then supplemented with interior constraints to obtain a square system for which an inverse is found. The process of selecting a suitable set of interior and boundary conditions is worth considering. Whilst the method used by Mulder (2017); Mulder and Huiskes (2017) yields easily inverted systems and is intuitive in 1D, more choices are available in higher dimensions.

To preserve the Cartesian topology of the grid, allowing points to be directly indexed, boundary points are taken as the intersection of the boundary normal at a grid point and the boundary, assuming this lies within a point-centred hyperrectangle whose sides correspond to a single grid increment in each dimension. Such a support region is shown in Figure 5. Interior points where a boundary point lies within a gridpoint-centred hyperrectangle with sides of length $2\eta\Delta x_n$ where Δx_n is the grid increment in the n^{th} dimension and η is a cutoff parameter are excluded for the purposes of constructing the extrapolant to ensure stability. Suitable values of η are

empirically determined in this work. Rigorous analysis would be preferable, although Mulder and Huiskes (2017) notes the challenge of carrying this out in practice.

An initial support region radius of $(M+1)/2$ is selected, expanded incrementally if insufficient information to constrain the extrapolation is contained therein. If the basis is sufficiently constrained, $rank(A)$ will equal the number of expansion terms. The required rank is $M+1$ in 1D, $(M+1)(M+2)/2$ in 2D, and $(M+1)(M+2)(M+3)/6$ in 3D respectively, the latter two corresponding to the $(M+1)^{\text{th}}$ triangular and tetrahedral numbers respectively. If one does not intend to be so strict about maintaining formal order, the order of the polynomial basis can be reduced instead, as in Mulder (2017); Mulder and Huiskes (2017). It is anticipated that this may be advisable with some boundary conditions and geometries to avoid stencil footprints becoming excessively large in edge cases.

Reconciliation with interior numerical scheme

Whilst the method described above can be used to directly obtain FD stencils, sudden switches in derivative approximation may cause instability. It is thus desirable to retain the interior discretization throughout the computational domain. Function values at stencil points lying outside the boundary are approximated with the boundary-constrained N-dimensional polynomial basis. The process of projecting this basis onto some set of points $\mathbf{X_e}$ can be represented by the matrix-vector multiplication

$$\mathbf{B}\boldsymbol{\delta} = \tilde{f}(\mathbf{X_e}), \tag{26}$$

where $\mathbf{B}$ contains the terms associated with each derivative in the Taylor series evaluated at the respective points in $\mathbf{X_e}$. Derivatives are approximated as

$$\boldsymbol{\delta} = \mathbf{A}^{+} \begin{pmatrix} f(\mathbf{X}) \\ g_1(\mathbf{X_b}) \\ \vdots \\ g_J(\mathbf{X_b}) \end{pmatrix}, \tag{27}$$

where $\mathbf{A}^{+}$ is the Moore-Penrose pseudoinverse of $\mathbf{A}$. Note that this is the inverse of Equation 24.

Using the 1D example in Equation 25, assume two exterior points, designated x_3 and x_4 are required to complete the stencil operator applied at the specified position. $\mathbf{B}$ is of the form

$$\mathbf{B} = \begin{pmatrix} 1 & (x_3 - x_0) & \frac{(x_3 - x_0)^2}{2} \\ 1 & (x_4 - x_0) & \frac{(x_4 - x_0)^2}{2} \end{pmatrix}, \tag{28}$$

and thus Equation 26 will yield approximations of $f(x_3)$ and $f(x_4)$ as functions of $f(x_1)$ and $f(x_2)$ (remaining entries in the right-hand side vector of Equation 25 are zero). More generally, this process yields exterior function values as functions of interior values and boundary conditions.

Substituting these into the interior stencil, a modified operator is obtained. This operator has a support region consisting of the unity of interior points in the original stencil ($\mathbf{X_i}$), those in the extrapolation stencil, and boundary points where conditions are enforced.

A stencil expression approximating some derivative of f can be expressed as

$$\mathbf{w} \cdot f(\mathbf{X}) = d, \tag{29}$$

where d is some arbitrary derivative and $\mathbf{w}$ contains the stencil weights corresponding to each function value in the vector $f(\mathbf{X})$. Separating the corresponding vector of stencil weights $\mathbf{w}$ into two subvectors $\mathbf{w_i}$ containing weights for points on the interior and $\mathbf{w_e}$ for points on the exterior, the expression for the modified stencil d can be obtained via

$$\begin{pmatrix} \mathbf{w_i} \\ \\ \mathbf{w_e} \end{pmatrix} \cdot \begin{pmatrix} f(\mathbf{X_i}) \\ \\ \tilde{f}(\mathbf{X_e}) \end{pmatrix} = d. \tag{30}$$

By constructing modified operators in this manner, extrapolated values are used without being explicitly calculated, with the extrapolation process encapsulated by the stencil coefficients. Furthermore, derivative stencils centred at different gridpoints will have independent extrapolations. This independence and the local nature of these extrapolation operators ensures that the linear systems constructed remain relatively small and simplifies the process of obtaining a solution to the system: both criterion prioritised by Hu (2016).

APPLICATION (2ND-ORDER ACOUSTIC WAVE EQUATION)

The 2nd-order acoustic wave equation is the first target for which boundary treatments are derived. Containing only a single time-variant field and equation, it has reduced computational cost and implementation complexity compared to other formu-

lations and wave equations; simple equations also yield simple boundary conditions. Whilst only P-wave components can be propagated and more complex physics such as anisotropy and viscoelasticity commonly desired for seismic imaging are omitted, applications remain in medical imaging (Guasch et al., 2020) and infrasound studies (Kim and Lees, 2014). Furthermore, it offers a platform to test the applicability of the method to various boundary conditions, paving the way for more complex physics.

The equation itself is given as

$$\frac{\partial^2 p}{\partial t^2} = c^2 \nabla^2 p + f, \tag{31}$$

containing pressure p and wavespeed c. An additional forcing term f is commonly included, typically some point source or set thereof in seismological contexts.

In seismic applications, this equation is typically discretized with an explicit timestepping scheme, replacing the time derivative with a second-order centred-difference approximation:

$$\frac{p^{t+1} - 2p^t + p^{t-1}}{\Delta t^2} = c^2 \nabla^2 p + f. \tag{32}$$

Rearranging for pressure at the forward timestep,

$$p^{t+1} = 2p^t - p^{t-1} + \Delta t^2 c^2 \nabla^2 p + \tilde{f}, \tag{33}$$

is obtained, where $\tilde{f} = f\Delta t^2$. This update equation approximates the field at the next iteration.

Spatial derivatives are all contained within the Laplacian; for the purposes of an immersed boundary, modified 2nd-derivative operators with respect to each dimension will need to be generated.

The reflection coefficient is near 1 for waves approaching topography from above and -1 when approaching from below. Considering the latter case, the surface can be approximated as an irregular free surface, upon which the condition

$$p(t, \boldsymbol{x_b}) = 0, \tag{34}$$

holds. Using Equation 31, incrementally higher-order boundary conditions are derived, these being $\nabla^2 p(t, \boldsymbol{x_b}) = 0$, $\nabla^4 p(t, \boldsymbol{x_b}) = 0$, and so forth.

Free-surface boundary constraints

These boundary conditions are approximated by substituting the polynomial basis into these boundary conditions in the place of pressure. Conditions of higher order than the spatial discretization will yield the trivial expression $0 = 0$, and are thus discarded.

Considering a fourth-order discretization in 2D, substituting the corresponding basis into the zeroth-order boundary condition yields the equation

$$\begin{aligned}
&p(x_0, y_0) + (x_b - x_0)\frac{\partial p}{\partial x}(x_0, y_0) + (y_b - y_0)\frac{\partial p}{\partial y}(x_0, y_0) + \frac{(x_b - x_0)^2}{2}\frac{\partial^2 p}{\partial x^2}(x_0, y_0) \\
&+ (x_b - x_0)(y_b - y_0)\frac{\partial^2 p}{\partial x \partial y}(x_0, y_0) + \frac{(y_b - y_0)^2}{2}\frac{\partial^2 p}{\partial y^2}(x_0, y_0) + \frac{(x_b - x_0)^3}{6}\frac{\partial^3 p}{\partial x^3}(x_0, y_0) \\
&+ \frac{(x_b - x_0)^2(y_b - y_0)}{2}\frac{\partial^3 p}{\partial x^2 \partial y}(x_0, y_0) + \frac{(x_b - x_0)(y_b - y_0)^2}{2}\frac{\partial^3 p}{\partial x \partial y^2}(x_0, y_0) \\
&+ \frac{(y_b - y_0)^3}{6}\frac{\partial^3 p}{\partial y^3}(x_0, y_0) + \frac{(x_b - x_0)^4}{24}\frac{\partial^4 p}{\partial x^4}(x_0, y_0) + \frac{(x_b - x_0)^3(y_b - y_0)}{6}\frac{\partial^4 p}{\partial x^3 \partial y}(x_0, y_0) \\
&+ \frac{(x_b - x_0)^2(y_b - y_0)^2}{4}\frac{\partial^4 p}{\partial x^2 \partial y^2}(x_0, y_0) + \frac{(x_b - x_0)(y_b - y_0)^3}{6}\frac{\partial^4 p}{\partial x \partial y^3}(x_0, y_0) \\
&+ \frac{(y_b - y_0)^4}{24}\frac{\partial^4 p}{\partial y^4}(x_0, y_0) = 0,
\end{aligned} \tag{35}$$

and similarly, the zero Laplacian and Biharmonic boundary conditions yield

$$\begin{aligned}
&\frac{\partial^2 p}{\partial x^2}(x_0,y_0)+\frac{\partial^2 p}{\partial y^2}(x_0,y_0)+(x_b-x_0)\frac{\partial^3 p}{\partial x^3}(x_0,y_0)+(y_b-y_0)\frac{\partial^3 p}{\partial x^2\partial y}(x_0,y_0)\\
&+(x_b-x_0)\frac{\partial^3 p}{\partial x\partial y^2}(x_0,y_0)+(y_b-y_0)\frac{\partial^3 p}{\partial y^3}(x_0,y_0)+\frac{(x_b-x_0)^2}{2}\frac{\partial^4 p}{\partial x^4}(x_0,y_0)\\
&+(x_b-x_0)(y_b-y_0)\frac{\partial^4 p}{\partial x^3\partial y}(x_0,y_0)+((x_b-x_0)^2+(y_b-y_0)^2)\frac{\partial^4 p}{\partial x^2\partial y^2}(x_0,y_0)\\
&+(x_b-x_0)(y_b-y_0)\frac{\partial^4 p}{\partial x\partial y^3}(x_0,y_0)+\frac{(y_b-y_0)^2}{2}\frac{\partial^4 p}{\partial y^4}(x_0,y_0)=0,
\end{aligned} \tag{36}$$

and

$$\frac{\partial^4 p}{\partial x^4}(x_0,y_0)+2\frac{\partial^4 p}{\partial x^2\partial y^2}(x_0,y_0)+\frac{\partial^4 p}{\partial y^4}(x_0,y_0)=0, \tag{37}$$

respectively. Each of these equations can be expressed as in Equation 2, with $\boldsymbol{\delta}$ given by

$$\begin{aligned}
\boldsymbol{\delta}^T=\Bigg(&p(x_0,y_0),\frac{\partial p}{\partial x}(x_0,y_0),\frac{\partial p}{\partial y}(x_0,y_0),\frac{\partial^2 p}{\partial x\partial y}(x_0,y_0),\frac{\partial^2 p}{\partial y^2}(x_0,y_0),\frac{\partial^3 p}{\partial x^3}(x_0,y_0),\\
&\frac{\partial^3 p}{\partial x^2\partial y}(x_0,y_0),\frac{\partial^3 p}{\partial x\partial y^2}(x_0,y_0),\frac{\partial^3 p}{\partial y^3}(x_0,y_0),\frac{\partial^4 p}{\partial x^4}(x_0,y_0),\frac{\partial^4 p}{\partial x^3\partial y}(x_0,y_0),\\
&\frac{\partial^4 p}{\partial x^2\partial y^2}(x_0,y_0),\frac{\partial^4 p}{\partial x\partial y^3}(x_0,y_0),\frac{\partial^4 p}{\partial y^4}(x_0,y_0)\Bigg).
\end{aligned} \tag{38}$$

For the zeroth-order condition,

$$\begin{aligned}
\mathbf{a}^T=\Bigg(&1,(x_b-x_0),(y_b-y_0),\frac{(x_b-x_0)^2}{2},(x_b-x_0)(y_b-y_0),\frac{(y_b-y_0)^2}{2},\frac{(x_b-x_0)^3}{6},\\
&\frac{(x_b-x_0)^2(y_b-y_0)}{2},\frac{(x_b-x_0)(y_b-y_0)^2}{2},\frac{(y_b-y_0)^3}{6},\frac{(x_b-x_0)^4}{24},\\
&\frac{(x_b-x_0)^3(y_b-y_0)}{6},\frac{(x_b-x_0)^2(y_b-y_0)^2}{4},\frac{(x_b-x_0)(y_b-y_0)^3}{6},\frac{(y_b-y_0)^4}{24},
\end{aligned} \tag{39}$$

whilst the Laplacian and Biharmonic conditions similarly correspond to

$$\begin{aligned}
\mathbf{a}^T=\Bigg(&0,0,0,1,0,1,(x_b-x_0),(y_b-y_0),(x_b-x_0),(y_b-y_0),\frac{(x_b-x_0)^2}{2},(x_b-x_0)(y_b-y_0),\\
&(x_b-x_0)^2+(y_b-y_0)^2),(x_b-x_0)(y_b-y_0),\frac{(y_b-y_0)^2}{2}\Bigg)
\end{aligned} \tag{40}$$

and

$$\mathbf{a}^T = (0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 2, 0, 1) \tag{41}$$

respectively. The right-hand side in each case is zero.

With Equations 39-41, three matrix rows can be constructed for each boundary point used to constrain the basis. Note that the fourth-order condition is invariant in space, and the corresponding row contains identical information irrespective of boundary point location.

Application to example geometry

To illustrate the process of constructing modified stencils for these conditions, consider the case of a fourth-order stencil approximating $\frac{\partial^2 p}{\partial y^2}$, truncated by an arc-shaped boundary, as shown in Figure 6. It is clear that it will not be possible to form this stencil as one would in free space since two of the requisite points lie outside the physical domain.

To rectify this, a circular support region of radius $2.5\Delta x$ is extended from the stencil centre, as in Figure 7. This support radius encircles 5 boundary points, determined via the previously-discussed criteria. A cutoff of $\eta = 0.5$ for the pressure field was found to be necessary to prevent instability related to the boundary forcing when imposing free surface boundary conditions, as in Mulder (2017). Consequently, 11 interior points are available to construct the 2D extrapolant. Each boundary point has three matrix rows associated with it, one for each boundary condition imposed,

whilst each interior point corresponds to a single row of $\mathbf{A}$.

The set of interior points used to fit the basis is shown in Figure 8a alongside the resultant submatrix containing the constraints applied at this set of points. Boundary points and corresponding matrix rows are shown in Figures 9a and 9b respectively.

For this case, the rank is equal to the number of columns and the Moore-Penrose pseudoinverse was used to invert this overdetermined system. A weighted least-squares approach was explored to prioritise particular boundary conditions or points but was prone to generating ill-conditioned linear systems whilst having minimal discernible benefit.

Continuing the pressure field onto the pair of required exterior points requires the construction of $\mathbf{B}$ as in Equation 26, constructing the expansion at exterior points and constructing matrix rows accordingly. Note that as all boundary forcing values are zero, columns of $\mathbf{BA}^+$ corresponding to boundary points can be ignored for the purpose of constructing the stencil. Applying weights for the interior stencil, the modified boundary operator for this particular point is obtained, shown in Figure 10

Reduction to 1D

Through particular choices made when applying the method, other, equally feasible stencils can be obtained, including the 1D approximation detailed in Mulder (2017). Considering the 1D case, the aforementioned free-surface boundary conditions reduce

to

$$p(x_b) = 0, \quad \frac{\partial^2 p}{\partial x^2}(x_b) = 0, \quad \frac{\partial^4 p}{\partial x^4}(x_b) = 0, \quad ... \tag{42}$$

and so forth. Suppose some case is encountered where a boundary truncates a stencil as in Figure 6. Selecting the $M/2$ closest points to the boundary (at distance greater than $\eta\Delta x$), $\mathbf{A}$ can be constructed such that it is square, enabling derivatives to be obtained by inverting this matrix. Thus the method described by Mulder (2017) can be considered a special case of the overarching method described in this paper, albeit distinct from the specific approach taken for the examples shown.

Figure 11 compares the proposed approach based on an N-dimensional basis with circular support to to an immersed boundary implementation based on per-dimension 1D extrapolations and a vacuum layer approach for a 2D second-order acoustic example featuring a sinusoidal free surface. The vacuum layer is far noisier than either immersed boundary approach with clear spurious diffactions and artefacts trailing the reflected wavefront. The immersed boundary based on axially-aligned 1D extrapolants performs better, although some artefacts remain as a result of this approximation. These are most visible immediately following the reflected wavefront. By contrast, the immersed boundary approach based on N-dimensional extrapolations yields a very smooth reflected wave with no visible unevenness. It is apparent that the use of N-dimensional extrapolations enables improved representation of the surface geometry and boundary conditions, yielding visibly superior results to approaches which rely on 1D approximations.

Convergence testing

To examine the convergence behaviour of the proposed approach, a setup initially presented by Mulder (2017) and subsequently used in Mulder and Huiskes (2017) (the second example in both) was replicated, as shown in Figure 12a. An exact solution exists for this example, allowing error in the numerical solution to be evaluated. Details of this solution are provided in Appendix B. Figure 12b shows the convergence behaviour of a scheme based on N-dimensional extrapolations, compared against a scheme based on axially-aligned 1D extrapolations.

For a fourth-order spatial discretization, the reduction in observed error with respect to grid increment was found to be initially just short of fourth-order, flattening around a grid increment of 0.02 as the spatial error is eclipsed by second-order timestepping error for finer grids. The timestep was set at 10% of the critical value, implying error introduced by the immersed boundary was minimal in all cases and that in many cases, topography implementation will cease to be the accuracy bottleneck when this immersed boundary treatment is applied. Reducing the timestep enabled the continuation of the fourth-order trend to finer grids, with accumulation of floating-point error resulting in a similar flattening at a smaller grid increment. The immersed boundary approach based on N-dimensional extrapolations was found to yield reduced error versus that based on 1D extrapolations for all grid increments tested, particularly at finer resolutions, albeit with similar convergence behaviour.

APPLICATION (1ST-ORDER ACOUSTIC WAVE EQUATION)

The method outlined in this paper yields accurate results for equations concerning a single field, whilst enabling higher-dimensional conditions to be imposed on the boundary. Another appeal of this approach is the readiness with which it is extended to cases where multiple fields are present. The acoustic wave equation can be formulated as a coupled system of pressure and particle velocity, introducing a spatially-variant density parameter. These equations take the form

$$\frac{\partial p}{\partial t} = \rho c^2 \boldsymbol{\nabla} \cdot \mathbf{v} + f, \quad \frac{\partial \mathbf{v}}{\partial t} = \frac{1}{\rho} \boldsymbol{\nabla} p, \tag{43}$$

where $\mathbf{v}$ is particle velocity and ρ is density. This formulation introduces additional fields in the form of components of the particle velocity vector. Given the aforementioned free-surface condition imposed on the pressure field, it is apparent from Equation 43 that the condition

$$\boldsymbol{\nabla} \cdot \mathbf{v}(t, \boldsymbol{x_b}) = 0 \tag{44}$$

must also be imposed. Whilst boundary conditions considered prior to now have concerned some property of a single field at the boundary, when multiple fields are present in a model, boundary conditions specifying some relationship between these may be present. Each vector component can be approximated by an independent polynomial basis in free space, but at the edge of the domain, they will require construction such that this relationship is respected. As will be highlighted - this extension can be naturally handled by the method described.

The proceeding section considers boundary conditions of this type more generally, before honing in on the application of this approach to the particle velocity free surface.

Boundary conditions containing multiple fields

Where multiple fields are present, individual Taylor Series are used to approximate each. Supposing K separate fields, labelled f_1 through f_K are present: ignoring boundary conditions for now, the polynomial fitting process can be expressed as

$$\begin{pmatrix} \mathbf{A_1} & & \\ & \ddots & \\ & & \mathbf{A_K} \end{pmatrix} \begin{pmatrix} \boldsymbol{\delta_1} \\ \vdots \\ \boldsymbol{\delta_K} \end{pmatrix} = \begin{pmatrix} f_1(\mathbf{X_1}) \\ \vdots \\ f_K(\mathbf{X_K}) \end{pmatrix}, \tag{45}$$

where $\mathbf{A_k}$ is the matrix containing coefficients associated with each derivative in the polynomial expansion approximating field f_k at the set of points $\mathbf{X_k}$. The vector $\boldsymbol{\delta_k}$ contains the various derivatives of f_k, analogous to the single-field case. The left-hand matrix is block-diagonal, and thus the linear system can be split into K smaller systems, each to be solved individually. In the case that boundary conditions concern only a single field apiece, this remains true, and one can still separate the system in this manner. The intuitive implication of this is the independence of polynomial expansions approximating each field in the absence of any constraints which would otherwise link them.

However, if boundary conditions impose some particular relationship between fields, then it is apparent that the resultant polynomial approximations of one of

these fields will require information regarding the other fields present in the boundary condition for the condition to be respected. As in the single-field case, each boundary condition will be approximated with Taylor-series expansions. Each field present within the constraint will have its own expansion. Each of these series contains derivatives of its respective function, contained within the corresponding $\boldsymbol{\delta_k}$ as in Equation 45. As such, the Taylor-series approximation of a constraint specifying some relationship between multiple fields will contain derivatives of multiple fields, and thus when represented in the previously-detailed dot-product form, the vector $\boldsymbol{\delta}$ will consist of multiple $\boldsymbol{\delta_k}$.

In the context of the linear system shown in Equation 45, rows corresponding to such boundary conditions will span multiple previously-separate blocks, linking them. As the extrapolant-fitting process relies on the inversion of the left-hand-side matrix, it follows that these linked blocks will need to be inverted in tandem as they can no longer be separated. As a boundary condition row in such cases maps multiple $\boldsymbol{\delta_k}$ onto a single boundary forcing value, and each of the previously-separate blocks maps its respective $\boldsymbol{\delta_k}$ onto the corresponding $f_k(\mathbf{X_k})$, the inverse of the resultant block will consequently map all $f_k(\mathbf{X_k})$ present onto any given derivative.

As a simple example, consider some pair of 1D functions f and h upon which the condition

$$f(x_b) + h(x_b) = 0, \tag{46}$$

is to be applied at the boundary. Approximating this constraint with 2nd-order Taylor

series expanded around some x_0 yields

$$\begin{aligned} &f(x_b) + (x_b - x_0)\frac{\partial f}{\partial x}(x_b) + \frac{(x_b - x0)^2}{2}\frac{\partial^2 f}{\partial x^2}(x_b) \\ &+ h(x_b) + (x_b - x_0)\frac{\partial h}{\partial x}(x_b) + \frac{(x_b - x0)^2}{2}\frac{\partial^2 h}{\partial x^2}(x_b) = 0. \end{aligned} \tag{47}$$

Representing this as a dot product of two vectors

$$\begin{pmatrix} \mathbf{a_f} \\ \\ \mathbf{a_h} \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\delta_f} \\ \\ \boldsymbol{\delta_h} \end{pmatrix} = 0 \tag{48}$$

where

$$\mathbf{a_f}^T = \mathbf{a_h}^T = \left(1, (x_b - x_0), \frac{(x_b - x0)^2}{2}\right), \tag{49}$$

$$\boldsymbol{\delta_f}^T = \left(f(x_b), \frac{\partial f}{\partial x}(x_b), \frac{\partial^2 f}{\partial x^2}\right), \tag{50}$$

and

$$\boldsymbol{\delta_h}^T = \left(h(x_b), \frac{\partial h}{\partial x}(x_b), \frac{\partial^2 h}{\partial x^2}\right). \tag{51}$$

To avoid confusion, note that $\mathbf{a_f} = \mathbf{a_h}$ results from the boundary condition applied and will generally not be the case, depending upon on constraints applied. A single point of f in this form is given as

$$\begin{pmatrix} \mathbf{a_f} \\ \\ 0 \end{pmatrix} \cdot \begin{pmatrix} \boldsymbol{\delta_f} \\ \\ \boldsymbol{\delta_h} \end{pmatrix} = 0. \tag{52}$$

Approximations of h at interior points can be formed in a similar manner. Assembling values of f and h at some set of points into a linear system using the above methodology, it is apparent that the row corresponding to the boundary condition links two otherwise separate blocks pertaining to interior points of f and h. A more thorough discussion on the effect of such boundary conditions on the structure of the linear system can be found in Appendix A.

Solving the resultant linear system gives derivatives of the basis (now multiple bases) at the expansion point as in the single-field case. Given these derivatives, the function can be continued beyond the boundary to obtain values at exterior points required by boundary-adjacent stencils. This is achieved in the same manner as the single-field case. Extrapolated function values are given as a weighted sum of boundary forcing values and interior values of all fields linked to the function of interest via boundary constraints. Consequently, any stencil formed using these approximated values will span all of these fields as well.

The particle velocity free surface

Returning to the zero-divergence boundary condition to be imposed on the particle velocity vector, approximating this with a second-order basis gives

$$\begin{aligned}&\frac{\partial v_x}{\partial x}(x_0,y_0)+(x_b-x_0)\frac{\partial^2 v_x}{\partial x^2}(x_0,y_0)+(y_b-y_0)\frac{\partial^2 v_x}{\partial x\partial y}+\frac{\partial v_y}{\partial y}(x_0,y_0)\\&+(x_b-x_0)\frac{\partial^2 v_y}{\partial x\partial y}+(y_b-y_0)\frac{\partial^2 v_y}{\partial y^2}(x_0,y_0)=0,\end{aligned} \tag{53}$$

yielding the characteristic row of $\mathbf{A}$:

$$\mathbf{a}^T=\Big(0,1,0,(x_b-x_0),(y_b-y_0),0,0,0,1,0,(x_b-x_0),(y_b-y_0)\Big). \tag{54}$$

There will be one such row for every boundary point. Returning to the geometry shown in Figure 7, boundary conditions are imposed at a single set of points for all fields to ensure consistency. For this particular geometry, there is insufficient information within the initially-selected radius of $1.5\Delta x$, and the support region must be expanded to that shown in Figure 7 to constrain the basis.

Setting $\eta = 0.5$ for the velocity field extrapolations was found to produce errors and nonphysical effects (although these did not result in instability). It is postulated that these resulted from overextension of the velocity field extrapolations. Setting $\eta = 0$ for both velocity fields was found to rectify this. The support region for each particle velocity component is as shown in Figure 13 alongside the corresponding $\mathbf{A}$. The structure of $\mathbf{A}^+$ is shown in Figure 14: it is apparent that this matrix is dense. Given $\mathbf{A}^+$ maps values of velocity components and boundary forcing values onto derivatives of both basis functions, each of these derivatives will be approximated as a weighted sum of values of both components. Consequently, any extrapolations using these bases will also be in terms of both v_x and v_y and thus stencils completed using these extrapolations will span both these fields too.

Note that a staggered scheme is used to prevent the emergence of checkerboard instability when solving this particular formulation. As such the base stencil used to construct the modified FD operators applied to the velocity fields will be backward-staggered.

The necessity of expanding the support region in this case is worth revisiting. The particle velocity free-surface and accompanying higher-order conditions $\nabla^2(\boldsymbol{\nabla} \cdot \mathbf{v}(\boldsymbol{x_b})) = 0$, $\nabla^4(\boldsymbol{\nabla} \cdot \mathbf{v}(\boldsymbol{x_b})) = 0$, and so forth concern multiple fields. Consequently, the number of boundary constraints within a support region of some given radius is likely to be low relative to the number of fields, requiring larger support regions in some cases.

Convergence testing

To explore the convergence behaviour of the boundary treatment devised, the previous setup was replicated for testing with the 1st-order acoustic wave equation. As before, a 4th-order spatial discretization was used and the timestep was set to 10% of the critical timestep. Results are shown in Figure 15.

As in the previous test, initial convergence is just short of 4th-order, before gradually flattening around a grid increment of 0.01, at which point convergence is around 2nd-order as timestepping error saturates the solution. A handful of blackspots where error is anomalously high compared to the prevailing trend are present at fine resolutions. Investigation of these anomalies was inconclusive, although they appear associated with specific resolutions; adding or removing a single node from the grid was sufficient to avoid these increases in error. Maximum error broadly continues to fall as the discretization is refined. Error introduced by the boundary is rapidly eclipsed by other sources as the discretization is refined, implying that the boundary treatment is unlikely to be a significant source of error in practical applications.

IMPLEMENTATION

Given a set of symbolic equations that hold on the boundary surface and a discretized signed-distance function (SDF) encapsulating the boundary position, a corresponding immersed boundary scheme and modified stencils implementing this boundary can be automatically devised via the method outlined. A framework to do so, Schism, was

developed as a plugin for Devito. This was done not only to expediate and simplify the implementation of the following test cases but to explore synergies between this generalised immersed boundary method and code generation. Due to the high-level nature of the abstractions created, the introduction of an immersed boundary to a numerical model written in Devito can be achieved with only a handful of additional lines of code and a qualitative understanding of the low-level mathematics.

This enabled all the examples shown in this paper to be implemented with a common codebase - only the top-level model specification is changed between examples. The unprecedented flexibility of this approach enables a wide range of geophysical scenarios to be condensed into an understandable, repurposable form, enabling maximum code reuse. Whilst a comprehensive overview of the mechanisms by which this was achieved is beyond the scope of this paper, the proceeding examples all leverage this functionality.

NUMERICAL EXAMPLES

Reflecting the wide range of relevant geophysical applications, a suite of examples showcasing our approach is presented. These examples are designed to resemble problems of interest and are based on real-world topography.

2D 2nd-order acoustic free-surface

The first example is based on an East-West profile across the summit of Mount St Helens, Washington, USA. With near-vertical crater walls, a mixture of concavity and convexity on the crater floor, and steep, uneven flanks rising over a kilometre from the surrounding landscape, it represents an ideal test for the method proposed.

The topographic surface was represented as an SDF discretised onto the FD grid. This representation has several advantages; its mathematical properties lend themselves to straightforward geometry handling, and it ensures that the resolution of the surface and resultant accuracy of the surface representation within the numerical scheme are consistent with the interior discretization. The surface can be located with much finer precision than the FD discretization, despite its discretization on the same grids, and the SDF can be constructed from high-resolution digital elevation models (DEMs) without the requirement to downsample or alias the raw data to match the FD grid. This is beneficial for real-world applications where such data may be structured or unstructured, or more finely resolved than the discretization required for numerical-dispersion-free wavefield propagation.

Material properties are homogeneous throughout the model domain, such that all topographic interactions observed are a product of the boundary treatment rather than any material discontinuity. Inclusion of a material contrast at the surface is unnecessary. Furthermore, care must be taken to avoid inconsistency between implicit interfaces represented by material parameters and the explicit interface encapsulated

by the immersed boundary. Variable material parameters are a separable concern and are trivially included.

The free-surface boundary condition is imposed on the pressure field on the topographic surface. The remaining edges of the computational domain have zero Dirichlet boundary conditions imposed for simplicity. Damping boundary conditions would be applied along these edges in practice. These are straightforwardly combined with the method presented in this paper.

A Ricker wavelet with a peak frequency of 8Hz is injected below the lava dome, at an elevation of 1250m. Placing the source close to the surface maximises the observed interaction between wavefield and topography. $\Delta x = \Delta y = 30m$ and the Courant number is set to 0.5. The spatial discretization used is fourth-order accurate: a precedent continued to all other examples shown. The temporal discretization is second-order accurate, as used in all examples presented in this work.

The uneven topography results in several distinct reflections, with further minor reverberations trailing the main wavefront, shown in Figure 16. Also apparent is the focusing and defocusing effect of concave and convex topography respectively, and the diffraction of the wavefront around obstructions. This yields a much more complex series of arrivals than would be observed for a flat surface, although first arrivals are only mildly distorted, in agreement with previously published findings.

2D 2nd-order acoustic rigid-surface

The free surface is not the only boundary condition of geophysical interest in the context of wavefield propagation. For acoustic waves propagating in the air, the Earth represents an extremely dense and essentially immobile surface: zero particle velocity at the interface corresponds to a rigid surface. The Mount St Helens profile used in the previous example once again features, although the surface now forms the lower bound of the domain.

A 1Hz Ricker source is placed at an elevation of 2600m above the lava dome. Located only slightly above the crater rim, this location was chosen to capture reverberation within the crater without trapping the majority of the wave.

Figure 17 shows the propagating wavefield, including the multiple distinct reflected wavefronts. Note the reversed polarity of these wavefronts versus those found in the previous example. Considerable reverberation within the crater is observed, alongside the wavefront diffracting over obstacles (most notably the crater rim).

3D 2nd-order acoustic free-surface

Whilst in many cases, wave propagation within a 3D physical domain can be adequately approximated along a suitable transect, this requires homogeneity in the omitted dimension. Where strong topographic variation in all directions is present, full 3D modelling is necessary to achieve realistic wavefield propagation.

This example was parameterised as in the prior 2D free surface example. Figure

18 shows the results of wavefield interaction with the volcanic topography. The first arrivals radiate outwards with minimal interruption, diffracting around obstacles in their path, with complex trains of reflected wavefronts from larger-scale surfaces forming a layered series of distinct arrivals. With only first arrivals remaining relatively unaffected, it is apparent that a flat surface will not adequately reproduce wavefield behaviour observed in rough terrain.

Wavefield slices shown in Figure 19 are superficially similar to the 2D approximations shown in Figure 16: the position, shape, and relative amplitude of major wavefronts are the same. On closer inspection however, greater complexity emerges: the main wavefronts contain additional reflections and smaller-scale differences are visible, particularly around the crater rim where some out-of-plane reverberations are visible.

3D 2nd-order acoustic rigid-surface

The previous rigid surface example is extended to 3D, the results of which are shown in Figure 20. Pronounced out-of-plane reflections are observed, particularly within the confines of the caldera where complex and protracted reverberations can be seen. Reflected wavefronts become more confused in 3D, exhibiting less coherency and continuity due to the wide range of paths taken by its constituent reflections.

These effects are particularly apparent when comparing Figure 21 to Figure 17, demonstrating the strongly 3D nature of wavefield propagation in this test case.

2D 1st-order acoustic free-surface

The immersed boundary approach presented in this paper is equally applicable to the 1st-order formulation of the acoustic wave equation, and to this end, the parameterisation and geometry from the prior 2D 2nd-order acoustic free-surface example are revisited, the results of which are shown in Figure 22.

Reflection geometries are as in the 2nd-order example, with particle velocity fields highlighting the partitioning of energy between horizontal and vertical particle motion, dependent on the orientation of the reflector. Clean, artefact-free reflections imply the accuracy of the vector boundary condition implementation.

3D 1st-order acoustic free-surface

Results of 3D test case featuring the 1st-order acoustic wave equation are shown in Figure 23. The pattern of radiation is identical to that in the equivalent 2nd-order test case (note that the wavefields differ between the two as the same time series was injected). Figure 24 shows wavefield snapshots. The y particle velocity field exemplifies the strongly 3D nature of interaction with topography, with reflected energy into the page strongly apparent. As with the 2nd-order formulation, reflections become more complex and confused in 3D due to the nature of the topography.

In this test case, it was found that undesirably large particle velocity stencils were required at a small number of of points, highlighting limitations of the Taylor series as a basis. It is anticipated that this could be avoided by improvements to the choice

of basis, linear system construction, and solver used. Alternatively, the construction of a more targeted support region may aid in alleviating this issue. For this test case, a basis-order-reduction strategy (as used by Mulder, 2017) was used at points where insufficient information was present to construct the extrapolant to limit stencil size.

1994 Canadian Foothills Model

To demonstrate the capacity of the immersed boundary approach described in this paper to handle the complexities of imaging applications, combining sharply varying topography with inhomogeneous material parameters, sources and receivers, and absorbing boundary conditions, a test case based on the 1994 BP Migration from Topography synthetic is presented. The 2nd-order acoustic wave equation is solved, with free surface conditions imposed on the topography. Sochacki sponge layers (Sochacki et al., 1987) are applied on all remaining edges to damp outgoing waves. A 20Hz Ricker source was injected and the model ran for 3000ms, the results of which are shown in Figure 25, with the corresponding shot gather shown in Figure 26.

The immersed boundary is demonstrably effective in this capacity, producing accurate results, evidenced by the clean shot gathers and lack of artefacts in the propagating wavefield. The separability of the boundary treatment from other aspects of the model including the velocity model, absorbing boundary conditions, sources, and receivers made it straightforward to integrate into the model, avoiding the need for any notable reworking.

CONCLUSIONS

A general immersed boundary treatment is presented, enabling a consistent methodology to be applied across multiple wave equations and boundary conditions. The boundary is encapsulated by modified FD stencils with a circular support region and spatially-variant coefficients, using an N-dimensional Taylor-series extrapolation scheme to continue the field beyond the edge of the domain. As the approach proposed naturally accommodates the implementation of higher-dimensional and vector boundary conditions, and it is not necessary to make any application-specific approximations to the boundary. The efficacy of this approach was demonstrated via convergence tests and a range of numerical examples featuring real-world topography, implementing both free and rigid surfaces with the first and second-order acoustic wave equations in 2D and 3D. The generality of the method presented enabled the development of a high-level framework, Schism, allowing all of these examples to be implemented via a broadly common codebase. This approach to immersed boundary implementation synergises with emerging code-generation approaches to FD kernel implementation, which was leveraged throughout this paper.

CODE AVAILABILITY AND REPRODUCIBILITY

Schism is an open-source codebase and can be found at `https://github.com/devitocodes/schism`. This repository contains all examples shown in this paper (excluding the 1994 BP Migration from Topography example), and a suite of unit tests. Schism can be installed from this repository as a Python module using Pip, or alternatively, a Dock-

erfile to run the code is also included. The codebase at the time of publication is archived on Zenodo at `https://zenodo.org/records/10726479`.

ACKNOWLEDGEMENTS

We wish to thank Wim Mulder for providing assistance with replicating his exact solutions. We also extend thanks to the rest of the Devito team and wider community for their feedback and support, without which this work would not have been possible. This work was funded as part of an EPSRC Standard Research Studentship (DTP).

APPENDIX A

BOUNDARY EFFECT ON MATRIX STRUCTURE

The boundary serves both to truncate the support region (as no gridpoints lie beyond it) and to introduce additional constraints to the polynomial fitting. This profoundly alters the linear system that must be solved to fit the polynomial, particularly where boundary conditions linking fields are present: the case of the particle velocity boundary condition for the free surface is one such example.

In free space, the stencil has an uninterrupted circular footprint comprising entirely of interior points. The corresponding matrix has a block-diagonal structure, as shown in Figure A-1b, with each block corresponding to one of the fields. Each block can be inverted separately, meaning that the resultant polynomial extrapolations have no impact on one another. Note that splitting the matrix up in this manner consid-

erably reduces the computational cost of finding the matrix inverse (or pseudoinverse as required), and it is thus desirable to do so where possible.

Inserting a 45°free-surface boundary truncating the support region, both the internal structure of the blocks and the overarching structure of the matrix are altered as shown in Figure A-2b. Inclusion of particle velocity boundary conditions merges the corresponding blocks, requiring them to be inverted in tandem. Resulting polynomial extrapolations will be dependent on both fields. As the pressure boundary conditions only concern the pressure field, this block retains its independence.

Whilst the matrices corresponding to these hyperspherical support regions will have consistently overdetermined blocks in free space, this may not be the case when boundaries are introduced. If a block becomes underdetermined, the radius of the support regions for functions contained therein can be expanded, introducing additional rows to the block until sufficient information is present.

APPENDIX B

EXACT SOLUTION USED FOR CONVERGENCE TESTS

The convergence tests detailed in this paper are based on an exact solution introduced by Mulder (2017). The pressure field is given as

$$p(t, x, y) = cos(mx_1 - \alpha t) cos\left(\frac{\pi y_1}{2}\right), \tag{B-1}$$

where

$$x_1 = x - Asin(x)cosh(y), \tag{B-2}$$

and

$$y_1 = y - Acos(x)sinh(y). \tag{B-3}$$

The parameter α is given by

$$\alpha = c_0\sqrt{m^2 + \left(\frac{\pi}{2}\right)^2}, \tag{B-4}$$

whilst the celerity field is given as

$$c(x,y) = c_0\left((1 - Acos(c)cosh(y))^2 + (Asin(x)sinh(y))^2\right)^{\frac{1}{2}}. \tag{B-5}$$

In the convergence tests shown, this solution was parameterised with $c_0 = 1$, $m = 8$, and $A = \frac{1}{4}$. For the first-order convergence test, density was set to 1. The particle velocity solutions are

$$\begin{aligned} v_x = &-\frac{m}{\alpha}\left(1 - Acos(x)cosh(y)\right)cos(mx_1 - \alpha t)cos\left(\frac{\pi y_1}{2}\right) \\ &+ \frac{\pi}{2\alpha}Asin(x)sinh(y)sin(mx_1 - \alpha t)sin\left(\frac{\pi y_1}{2}\right), \end{aligned} \tag{B-6}$$

and

$$\begin{aligned} v_y = &\frac{m}{\alpha}Asin(x)sinh(y)cos(mx_1 - \alpha t)cos\left(\frac{\pi y_1}{2}\right) \\ &+ \frac{\pi}{2\alpha}\left(1 - Acos(x)cosh(y)\right)sin(mx_1 - \alpha t)sin\left(\frac{\pi y_1}{2}\right). \end{aligned} \tag{B-7}$$

The free surface is defined by $y_1 = -1$ and the solution is periodic in $x \in [0, 2\pi)$. Zero pressure and zero particle velocity flux are imposed at $y = 0$.

REFERENCES

Almuhaidib, A. M., and M. N. Toksoz, 2015, Finite difference elastic wave modeling with an irregular free surface using ADER scheme: Journal of Geophysics and Engineering, **12**, 435–447.

Bartel, L. C., N. P. Symons, and D. F. Aldridge, 2000, Graded boundary simulation of air/earth interfaces in finite-difference elastic wave modeling: Presented at the SEG Technical Program Expanded Abstracts.

Bleibinhaus, F., and S. Rondenay, 2009, Effects of surface scattering in full-waveform inversion: Geophysics, **74**, 69–77.

Bohlen, T., and E. H. Saenger, 2006, Accuracy of heterogeneous staggered-grid finite-difference modeling of Rayleigh waves: Geophysics, **71**, 109–115.

Boore, D. M., 1972, A note on the effect of simple topography on seismic SH waves: Bulletin of the Seismological Society of America, **62**, 275–284.

———, 1973, The effect of simple topography on seismic waves: implications for the accelerations recorded at Pacoima Dam, San Fernando Valley, California: Bulletin of the Seismological Society of America, **63**, 1603–1609.

Borisov, D., R. Modrak, F. Gao, and J. Tromp, 2018, 3D elastic full-waveform inversion of surface waves in the presence of irregular topography using an envelope-based misfit function: Geophysics, **83**, R1–R11.

Brehm, C., and H. F. Fasel, 2013, A novel concept for the design of immersed interface methods: Journal of Computational Physics, **242**, 234–267.

Brehm, C., C. Hader, and H. F. Fasel, 2015, A locally stabilized immersed boundary

method for the compressible Navier – Stokes equations: Journal of Computational Physics, **295**, 475–504.

Caunt, E., 2024, devitocodes/schism: Schism v1.0.2.

de la Puente, J., M. Ferrer, M. Hanzich, J. E. Castillo, and J. M. Cela, 2014, Mimetic seismic wave modeling including topography on deformed staggered grids: Geophysics, **79**, 125–143.

Dong, H., M. Bozkurttas, R. Mittal, P. Madden, and G. Lauder, 2010, Computational modelling and analysis of the hydrodynamics of a highly deformable fish pectoral fin: Journal of Fluid Mechanics, **645**, 345–373.

Dupros, F., F. D. Martin, E. Foerster, D. Komatitsch, and J. Roman, 2010, High-performance finite-element simulations of seismic wave propagation in three-dimensional nonlinear inelastic geological media: Parallel Computing, **36**, 308–325.

Fee, D., L. Toney, K. Kim, R. W. Sanderson, A. M. Iezzi, R. S. Matoza, S. D. Angelis, A. D. Jolly, J. J. Lyons, M. M. Haney, and D. Keir, 2021, Local explosion detection and infrasound localization by reverse time migration using 3-D finite-difference wave propagation: Frontiers in Earth Science, **9**, 1–14.

Fletcher, R. P., X. Du, and P. J. Fowler, 2009, Reverse time migration in tilted transversely isotropic (TTI) media: Geophysics, **74**, 179–187.

Galis, M., P. Moczo, and J. Kristek, 2008, A 3-D hybrid finite-difference — finite-element viscoelastic modelling of seismic wave motion: Geophysical Journal International, **175**, 153–184.

Gao, L., R. Brossier, B. Pajot, J. Tago, and J. Virieux, 2015, An immersed free-surface boundary treatment for seismic wave simulation: Geophysics, **80**, 193–209.

Goldstein, D., R. Handler, and L. Sirovich, 1993, Modelling a no-slip flow boundary with an external force field: Journal of Computational Physics, **105**, 354–366.

Graves, R. W., 1996, Simulating seismic wave propagation in 3D elastic media using staggered-grid finite differences: Bulletin of the Seismological Society of America, **86**, 1091–1106.

Griffiths, D. W., and G. A. Bollinger, 1979, The effect of Appalachian mountain topography on seismic waves: Bulletin of the Seismological Society of America, **69**, 1081–1105.

Guasch, L., O. Calderón Agudo, M. X. Tang, P. Nachev, and M. Warner, 2020, Full-waveform inversion imaging of the human brain: npj Digital Medicine, **28**, 1–12.

Hayashi, K., D. R. Burns, and M. N. Tokso, 2001, Discontinuous-grid finite-difference seismic modeling including surface topography: Bulletin of the Seismological Society of America, **91**, 1750–1764.

Hestholm, S., and B. Ruud, 1994, 2D finite-difference elastic wave modelling including surface topography: Geophysical Prospecting, **42**, 371–390.

———, 2002, 3D free-boundary conditions for coordinate-transform finite-difference seismic modelling: Geophysical Prospecting, **50**, 463–474.

Hestholm, S., and C. Street, 2000, 3-D surface topography boundary conditions in seismic wave modelling: SEG Technical Program Expanded Abstracts, 2432–2435.

Hestholm, S. O., B. O. Ruud, and E. S. Husebye, 1999, 3-D versus 2-D finite-difference seismic synthetics including real surface topography: Physics of the Earth and Planetary Interiors, **113**, 339–354.

Hu, W., 2016, An improved immersed boundary finite-difference method for seismic wave propagation modeling with arbitrary surface topography: Geophysics, **81**, 311–322.

Kim, K., and J. M. Lees, 2014, Local Volcano Infrasound and Source Localization Investigated by 3D Simulation: Seismological Research Letters, **85**, 1177–1186.

Lai, M.-C., and C. S. Peskin, 2000, An immersed boundary method with formal second-order accuracy and reduced numerical viscosity: Journal of Computational Physics, **719**, 705–719.

Li, X., and G. Yao, 2020, An immersed boundary method with iterative symmetric interpolation for irregular surface topography in seismic wavefield modelling: Journal of Geophysics and Engineering, **0**, 1–18.

Li, X., G. Yao, F. Niu, D. Wu, and N. Liu, 2022, Waveform inversion of seismic first arrivals acquired on irregular surface: Geophysics, **87**, 291–304.

Liu, S., X. Li, W. Wang, and Y. Liu, 2014, A mixed-grid finite element method with PML absorbing boundary conditions for seismic wave modelling: Journal of Geophysics and Engineering, **11**, 055009.

Liu, Y., and M. K. Sen, 2009, A new time – space domain high-order finite-difference method for the acoustic wave equation: Journal of Computational Physics, **228**, 8779–8806.

Lombard, B., J. Piraux, C. Gelis, and J. Virieux, 2008, Free and smooth boundaries in 2-D finite-difference schemes for transient elastic waves: Geophysical Journal International, **172**, 252–261.

Louboutin, M., M. Lange, F. Luporini, N. Kukreja, P. A. Witte, F. J. Herrmann, P.

Velesko, and G. J. Gorman, 2019, Devito (v3.1.0): an embedded domain-specific language for finite differences and geophysical exploration: Geoscientific Model Development, **12**, 1165–1187.

Luporini, F., M. Louboutin, M. Lange, N. Kukreja, P. Witte, J. Hückelheim, C. Yount, P. H. J. Kelly, F. J. Herrmann, and G. J. Gorman, 2020, Architecture and performance of Devito, a system for automated stencil computation: ACM Transactions on Mathematical Software, **46**.

Martin, B., B. Fornberg, and A. St-cyr, 2015, Seismic modeling with radial-basis-function-generated finite differences: Geophysics, **80**, 137–146.

Mckee, K., D. Fee, C. Rowell, and A. Yokoo, 2014, Network-based evaluation of the infrasonic source location at Sakurajima Volcano, Japan: Seismological Research Letters, **85**, 1200–1211.

Mittal, R., H. Dong, M. Bozkurttas, and F. M. Najjar, 2008, A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries: Journal of Computational Physics, **227**, 4825–4852.

Monteiller, V., D. Komatitsch, and N. Fuji, 2013, A hybrid method to compute short-period synthetic seismograms of teleseismic body waves in a 3-D regional model: Geophysical Journal International, **192**, 230–247.

Monteiller, V., D. Komatitsch, and Y. Wang, 2015, Three-dimensional full waveform inversion of short-period teleseismic wavefields based upon the SEM-DSM hybrid method: Geophysical Journal International, **202**, 811–827.

Muir, F., J. Dellinger, J. Etgen, and D. Nichols, 1992, Modeling elastic fields across irregular boundaries: Geophysics, **57**, 1189–1193.

Mulder, W. A., 2017, A simple finite-difference scheme for handling topography with the second-order wave equation: Geophysics, **82**, 111–120.

Mulder, W. A., and M. J. Huiskes, 2017, A simple finite-difference scheme for handling topography with the first-order wave equation: Geophysical Journal International, **210**, 482–499.

Mulder, W. A., and R. Shamasundar, 2016, Performance of continuous mass-lumped tetrahedral elements for elastic wave propagation with and without global assembly: Geophysical Journal International, **207**, 414–421.

Nakamura, T., H. Takenaka, T. Okamoto, and Y. Kaneda, 2012, FDM simulation of seismic-wave propagation for an aftershock of the 2009 Suruga Bay earthquake : effects of ocean-bottom topography and seawater layer: Bulletin of the Seismological Society of America, **102**, 2420–2435.

Nuber, A., E. Manukyan, and H. Maurer, 2016, Ground topography effects on near-surface elastic full waveform inversion: Geophysical Journal International, **207**, 67–71.

Ohminato, T., and B. A. Chouet, 1997, A free-surface boundary condition for including 3D topography in the finite-difference method: Bulletin of the Seismological Society of America, **87**, 494–515.

Oprsal, I., and J. Zahradnik, 1999, Elastic finite-difference method for irregular grids: Geophysics, **64**, 240–250.

Petersson, N. A., and B. Sjögreen, 2015, Wave propagation in anisotropic elastic materials and curvilinear coordinates using a summation-by-parts finite difference method: Journal of Computational Physics, **299**, 820–841.

Rathgeber, F., D. A. Ham, L. Mitchell, M. Lange, F. Luporini, A. T. T. Mcrae, G.-t. Bercea, G. R. Markall, and P. H. J. Kelly, 2016, Firedrake : automating the finite element method by composing abstractions: ACM Transactions on Mathematical Software, **43**, 24.

Reinoso, E., L. C. Wrobel, and H. Power, 1996, Three-dimensional scattering of seismic waves from topographical structures: Soil Dynamics and Earthquake Engineering, **16**, 41–61.

Ripperger, J., H. Igel, and J. Wasserman, 2003, Seismic wave simulation in the presence of real volcano topography: Journal of Volcanology and Geothermal Research, **128**, 31–44.

Roberts, K. J., S. Gioria, and W. J. Pringle, 2021a, SeismicMesh : triangular meshing for seismology: The Journal of Open Source Software, **6**, 2687.

Roberts, K. J., A. Olender, L. Franceschini, R. C. Kirby, R. S. Gioria, and B. S. Carmo, 2021b, Spyro : a Firedrake-based wave propagation and full waveform inversion finite element solver: Geoscientific Model Development, [**preprint**].

Robertsson, J. O. A., 1996, A numerical free-surface condition for elastic/viscoelastic finite-difference modeling in the presence of topography: Geophysics, **61**, 1921–1934.

Romdhane, A., G. Grandjean, R. Brossier, F. Réjiba, S. Operto, and J. Virieux, 2011, Shallow-structure characterization by 2D elastic full-waveform inversion: Geophysics, **76**, 1–13.

Schultz, C. A., 1997, A density-tapering approach for modeling the seismic response of free-surface topography: Geophysical Research Letters, **24**, 2809–2812.

Seo, J. H., and R. Mittal, 2011, A high-order immersed boundary method for acoustic wave scattering and low-Mach number flow-induced sound in complex geometries: Journal of Computational Physics, **230**, 1000–1019.

Shin, J., Y. Sim, C. Shin, and H. Calandra, 2013, Laplace-domain full waveform inversion using irregular finite elements for complex foothill environments: Journal of Applied Geophysics, **96**, 67–76.

Shragge, J., 2014, Solving the 3D acoustic wave equation on generalized structured meshes : A finite-difference time-domain approach: Geophysics, **79**, 363–378.

Shragge, J., and T. Konuk, 2020, Tensorial elastodynamics for isotropic media: Geophysics, **85**, T359–T373.

Shragge, J., and B. Tapley, 2017, Solving the tensorial 3d acoustic wave equation: A mimetic finite-difference time-domain approach: Geophysics, **82**, T183–T196.

Slotnick, J., A. Khodadoust, J. Alonso, D. Darmofal, W. Gropp, E. Lurie, and D. Mavriplis, 2014, CFD vision 2030 study : A path to revolutionary computational aerosciences: Technical Report March 2014.

Sochacki, J., R. Kubichek, J. George, W. Fletcher, and S. Smithson, 1987, Absorbing boundary conditions and surface waves: Geophysics, **52**, 60–71.

Sun, Y.-C., W. Zhang, J.-K. Xu, and X. Chen, 2017, Numerical simulation of 2D seismic wave propagation in th presence of a topographic fluid – solid interface at the sea bottom by the curvilinear grid finite-difference method: Geophysical Journal International, **210**, 1721–1738.

Symes, W. W., and T. Vdovina, 2009, Interface error analysis for numerical wave propagation: Computational Geosciences, **13**, 363–371.

Takekawa, J., H. Mikada, and N. Imamura, 2015, A mesh-free method with arbitrary-order accuracy for acoustic wave propagation: Computers and Geosciences, **78**, 15–25.

Takemura, S., T. Furumura, and T. Maeda, 2015, Scattering of high-frequency seismic waves caused by irregular surface topography and small-scale velocity inhomogeneity: Geophysical Journal International, **201**, 459–474.

Tavelli, M., M. Dumbser, D. Etienne, L. Rannabauer, T. Weinzierl, and M. Bader, 2019, A simple diffuse interface approach on adaptive Cartesian grids for the linear elastic wave equations with complex topography: Journal of Computational Physics, **386**, 158–189.

Vargas, A., R. Mittal, and H. Dong, 2008, A computational study of the aerodynamic performance of a dragonfly wing section in gliding flight: Bioinspiration and Biomimetics, **3**, 1–13.

Vieira, N., H. Alexander, and D. Di, 2018, Accurate seabed modeling using finite difference methods: Computational Geoscience, **22**, 469–484.

Warner, M., A. Ratcliffe, T. Nangoo, J. Morgan, A. Umpleby, N. Shah, V. Vinje, Š. Ivan, L. Guasch, C. Win, G. Conroy, and A. Bertrand, 2013, Anisotropic 3D full-waveform inversion: Geophysics, **78**, 59–80.

Wu, H., C. Sun, S. Li, J. Tang, and N. Xu, 2021, Mesh-free radial-basis-function-generated finite differences and their application in reverse time migration: Geophysics, **86**, S91–S101.

Zahradnik, J., P. Moczo, and F. Hron, 1993, Testing four elastic finite-difference schemes for behaviour at discontinuities: Bulletin of the Seismological Society of

America, **83**, 107–129.

Zeng, C., J. Xia, R. D. Miller, and G. P. Tsoflias, 2012, An improved vacuum formulation for 2D finite-difference modeling of Rayleigh waves including surface topography and internal discontinuities: Geophysics, **77**, 1–9.

Zhang, D., G. Zhan, G. T. Schuster, and K. Abdullah, 2013a, Multi-source least-squares reverse time migration with topography: SEG Technical Program Extended Abstracts, 3736–3740.

Zhang, W., and X. Chen, 2006, Traction image method for irregular free surface boundaries in finite difference seismic wave simulation: Geophysical Journal International, **167**, 337–353.

Zhang, W., Z. Zhang, and X. Chen, 2012, Three-dimensional elastic wave numerical modelling in the presence of surface topography by a collocated-grid finite-difference method on curvilinear grids: Geophysical Journal International, **190**, 358–378.

Zhang, Z., W. Zhang, H. Li, and X. Chen, 2013b, Stable discontinuous finite-difference seismic wave modelling: Geophysical Journal International, 1179–1188.

Zhebel, E., S. Minisini, A. Kononov, and W. A. Mulder, 2014, A comparison of continuous mass-lumped finite elements with finite differences for 3-D wave propagation: Geophysical Prospecting, **63**, 1111–1125.

LIST OF FIGURES

shown as a green line. Boundary conditions increase in order towards the bottom of the matrix. Note that the highest-order boundary conditions here are invariant with position and thus redundant. Where multiple boundary conditions are imposed, a single boundary point will correspond to multiple constraints.

5 Construction of the support region in the vicinity of the boundary. Symbols are as in Figures 1 and 4. The solid black line represents the boundary surface. Hollow black crosses are considered to be outside or too close to the boundary for the purposes of constructing the extrapolant (as defined in the proceeding section).

6 The stencil footprint of a 4th-order-accurate $\frac{\partial^2 p}{\partial y^2}$ stencil truncated by an arc-shaped boundary. The bold black cross is the stencil position, whilst pale green crosses are other interior points within the stencil. Values at both hollow black crosses are required by the stencil but are located outside the computational domain.

7 Support region for a 2D polynomial fitted with a combination of available interior points and boundary points.

8 Interior points within the support region and structure of the corresponding submatrix. Rows in the matrix from top to bottom correspond to points within the support region working downwards from left to right.

9 Boundary points within the support region and structure of the corresponding submatrix. Grey arrows indicate the vector $\boldsymbol{x_b} - \boldsymbol{x}_0$ for these points. Rows in the matrix from top to bottom correspond to points from left to right.

10 Comparison of the standard stencil footprint and weights to that of the modified boundary operator for the case illustrated in Figure 6. Colours indicate the values of stencil weights at each point.

11 Snapshots at 700ms of a wavefront reflecting off a sinusoidal hill. The leftmost example implements the free surface as a vacuum layer, whilst the remaining subfigures impose this condition as an immersed boundary. The centre subfigure shows the results of an immersed boundary treatment based on axially-aligned 1D extrapolations as in Mulder (2017), whilst the rightmost subfigure shows the interaction of the wavefield with an immersed free surface based on N-dimensional extrapolants with circular support regions. Resolution and discretization is identical for all runs; only the boundary treatment is altered. The location of the Ricker source is marked with a red star. Wavefield amplitudes are normalised against the maximum absolute value in each subfigure. This convention is continued henceforth.

12 The exact solution at $t = 0$ is shown in the left subfigure. The solid black line is the free surface, the left and right sides of the domain have periodic boundary conditions applied, and the wavefield is mirrored across the lower boundary. A comparison of the convergence behaviour of the method proposed to one using 1D approximations is shown on the right.

13 Assembling $\mathbf{A}$ for the particle velocity fields. Horizontal arrows correspond with v_x points used to construct the extrapolant, whilst vertical arrows are the respective v_y points. The black cross designates the expansion point. Note that only the subgrids of concern are shown for clarity. Note how the interior points of each field correspond with an independent block, whilst the boundary condition rows span both.

14 The structure of the Moore-Penrose pseudoinverse of $\mathbf{A}$ shown in Figure 13.

15 Convergence of the numerical scheme for the 1st-order formulation of the acoustic wave equation using the same setup shown in Figure 12a. Note that scales are slightly different in this figure.

16 Snapshots of the 2D 2nd-order acoustic wavefield interacting with a free surface at 375ms, 750ms, 1125ms, and 1500ms. Celerity is 2.5km/s throughout the model. The black line designates the isosurface $s(x, y) = 0$ on the SDF, coinciding with the surface. The wavefield shown in each snapshot is normalised for clarity.

17 Snapshots of the 2D 2nd-order acoustic wavefield interacting with a rigid surface at 2.5s, 5s, 7.5s, and 10s. Celerity is 350m/s throughout the model. Parameters are altered in this example to better reflect infrasound propagation problems to which this boundary condition is applicable. The wavefield shown in each snapshot is normalised for clarity.

18 Render of the 3D 2nd-order acoustic wavefield at 1125ms, showing the free surface topography. Slices of the wavefield are shown aligned and diagonal to each compass direction for clarity. Wavefield transparency is scaled with amplitude to emphasise the wavefronts.

19 Slices through the 3D 2nd-order acoustic wavefield interacting with the free surface on the profile used for the 2D examples. Snapshots are at 375ms, 750ms, 1125ms, and 1500ms respectively. Celerity is 2.5km/s throughout the model. The black line designates the isosurface $s(x, y, z) = 0$ on the SDF, coinciding with the surface. The wavefield shown in each snapshot is normalised for clarity.

20 Render of the 3D 2nd-order acoustic wavefield at 7.5s, interacting with the topographic rigid surface. Slices of the wavefield are shown aligned and diagonal to

each compass direction for clarity. Wavefield transparency is scaled with amplitude as before.

21 Snapshots of the 3D 2nd-order acoustic wavefield interacting with the rigid surface at 2.5s, 5s, 7.5s, and 10s. The transect is chosen to match that used for the 2D examples. Celerity is 350m/s throughout the model. The wavefield shown in each snapshot is normalised for clarity.

22 An irregular free surface implemented for the 1st-order acoustic wave equation in 2D. Snapshots of the pressure (left), x particle velocity (middle), and y particle velocity (right) wavefields at 375ms, 750ms, 1125ms, and 1500ms. Celerity is 2.5km/s throughout the model, density is homogeneous throughout. The black line designates the isosurface $s(x, y) = 0$ on the SDF, coinciding with the surface. The wavefields shown in each snapshot are normalised for clarity.

23 Render of the pressure wavefield at 1125ms when solving the 3D 1st-order acoustic wave equation in the presence of free surface topography. Slices of the wavefield are shown aligned and diagonal to each compass direction for clarity. Wavefield transparency is scaled with amplitude to emphasise the wavefronts.

24 Snapshots of the pressure (left), x particle velocity (middle left), y particle velocity (middle right), and z particle velocity (right) wavefields at 375ms, 750ms, 1125ms, and 1500ms. The ty-axis is oriented into the page. Celerity is 2.5km/s throughout the model, density is homogeneous throughout. The black line designates the isosurface $s(x, y, z) = 0$ on the SDF, coinciding with the surface. The wavefields shown in each snapshot are normalised for clarity.

25 A snapshot of the wavefield propagating through the 1994 BP Migration

from Topography synthetic at 3000ms. The free surface, shown as a black line, is implemented using the same immersed boundary treatment as demonstrated in Figure 16. The source location is shown as a red star, whilst decimated receivers used to generate Figure 26 are shown in orange.

26 The shot gather for the geometry shown in Figure 25. The minimal noise, smoothness of the reflectors, and clearly defined diffractions demonstrate the accuracy achieved by the immersed boundary approach.

A-1 A circular support region in free space and its corresponding matrix structure for the first-order acoustic wave equation. Nonzero elements are black, zero elements are white. Staggered particle velocity subgrids are omitted to prevent excessive cluttering. Individual blocks are highlighted in green, and from top-left to bottom-right correspond to pressure, horizontal particle velocity, and vertical particle velocity. It is clear that this matrix can be split into three smaller systems which can be solved individually; each polynomial can be fitted independently of the others.

A-2 A circular support region cut by a diagonal boundary and its corresponding matrix structure for the first-order acoustic wave equation. Nonzero elements are black, zero elements are white. Staggered particle velocity subgrids are omitted to prevent excessive cluttering. Blocks are highlighted with solid green boxes, with the dashed green lines separating rows corresponding to interior points and those corresponding to boundary conditions. Note that the previously separate particle velocity blocks have become merged, as the zero-divergence boundary condition and corresponding higher-order conditions span both particle velocity fields.

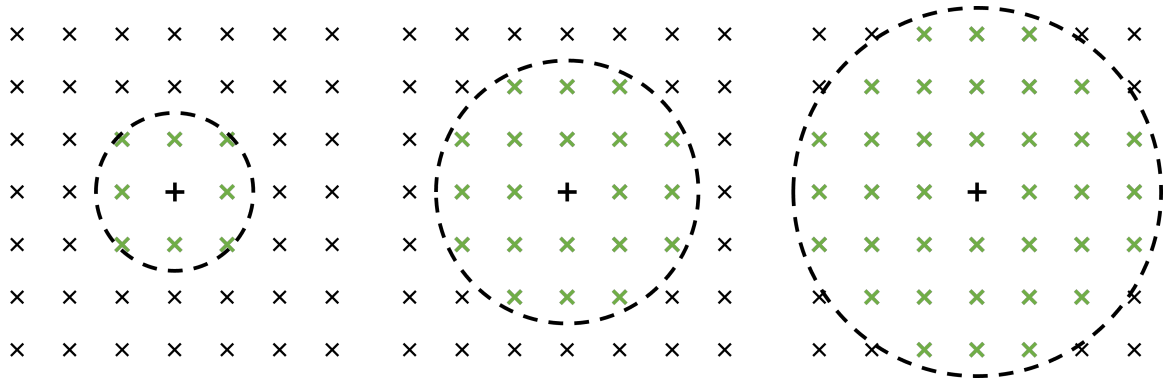


Figure 1: A selection of 2D support region footprints with radii of 1.5, 2.5, and 3.5 respectively. The vertical black cross designates the stencil centre (the position at which the stencil is applied), green crosses are interior points, and the dotted black line shows the extent of the support region.

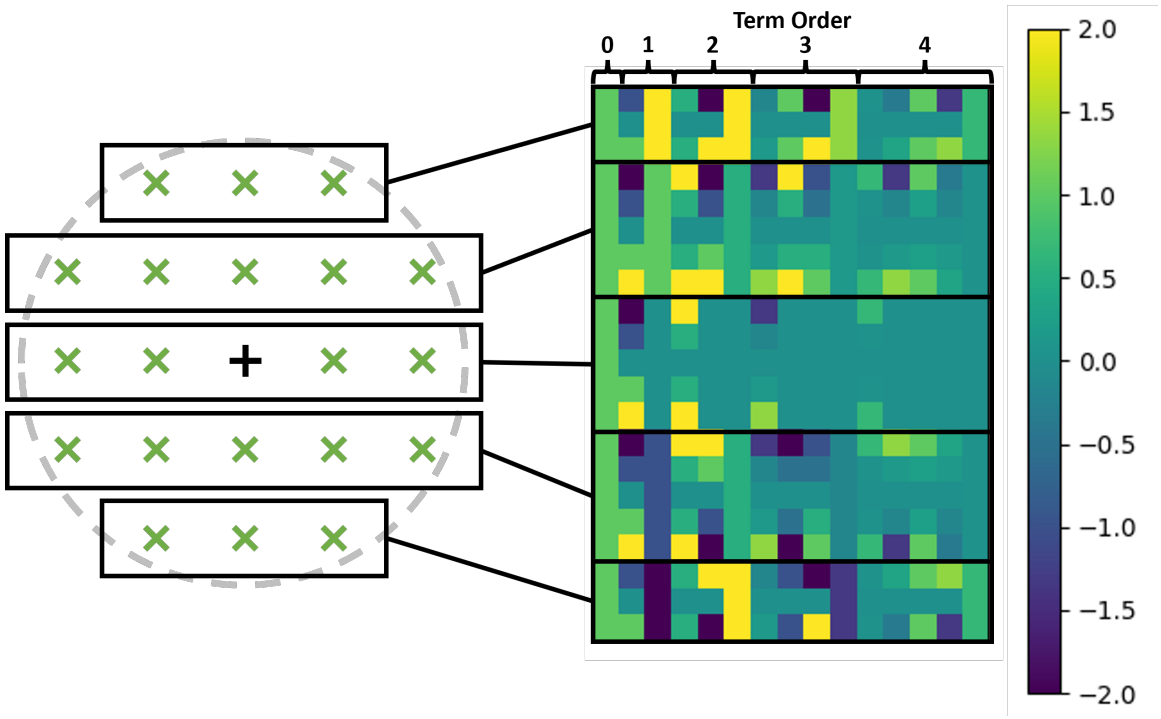


Figure 2: A 2D support region footprint for a 4th-order basis consisting of points **X** and its correspondence to the matrix **A** (the structure of which is shown on the right). The support region radius is marked by a dashed grey line. The stencil points in each box correspond to the matrix rows in the connected box, containing coefficients of the polynomial expansion at that point. The order of derivatives to which coefficients correspond increases from left to right. Colours within the matrix plot correspond to the values of those elements, with yellows and blues representing positive and negative values respectively. The vertical black cross designates the stencil centre: the position at which the stencil is applied and expansion point for the polynomial.

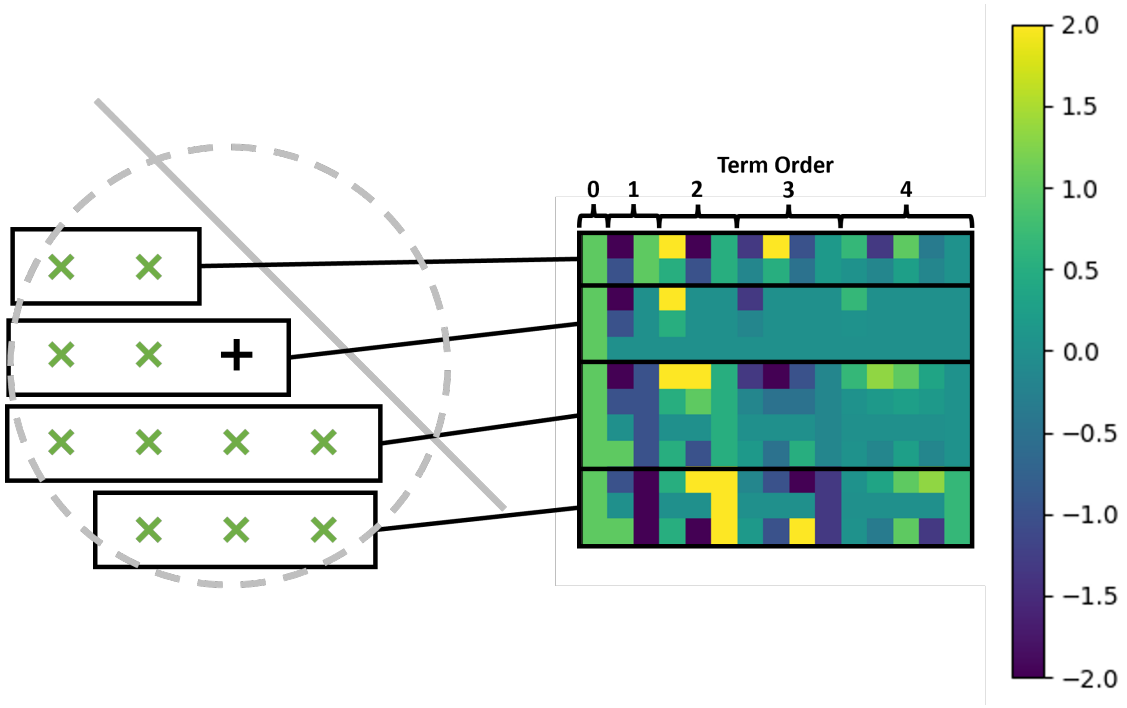


Figure 3: The effect of truncation of the stencil footprint shown in Figure 2 by a 45°boundary. The support region radius is marked as a dashed grey line, whilst the boundary is denoted with a solid grey line.

Caunt et al. –

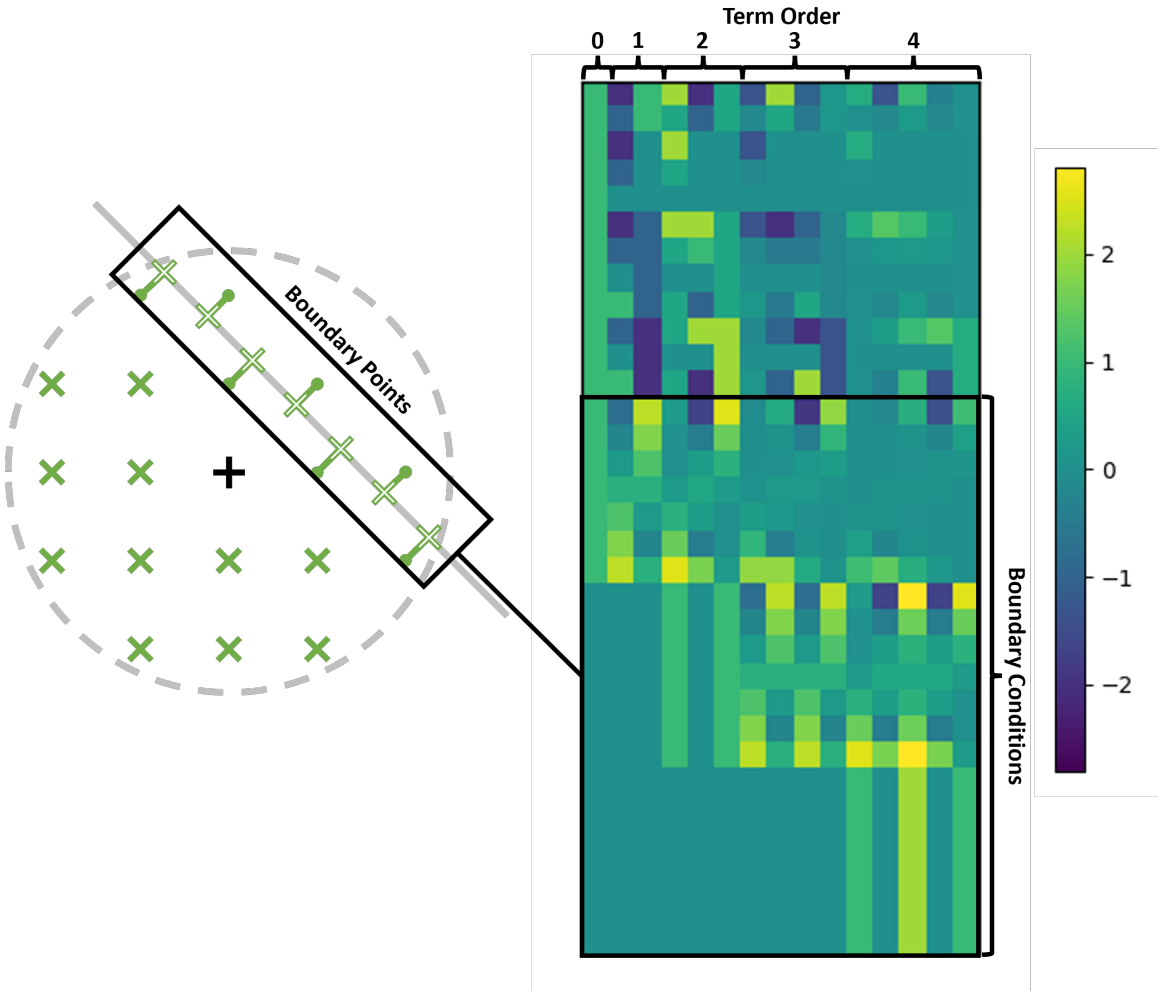


Figure 4: The effect of adding boundary constraints to the truncated support region shown in Figure 3. The support region radius is marked as a dashed grey line, whilst the boundary is denoted with a solid grey line. Boundary points where conditions are imposed are shown as hollow green crosses. Solid green dots show the centre of FD cells containing a boundary point; the normal from this point to the boundary is shown as a green line. Boundary conditions increase in order towards the bottom of the matrix. Note that the highest-order boundary conditions here are invariant with position and thus redundant. Where multiple boundary conditions are imposed, a single boundary point will correspond to multiple constraints.

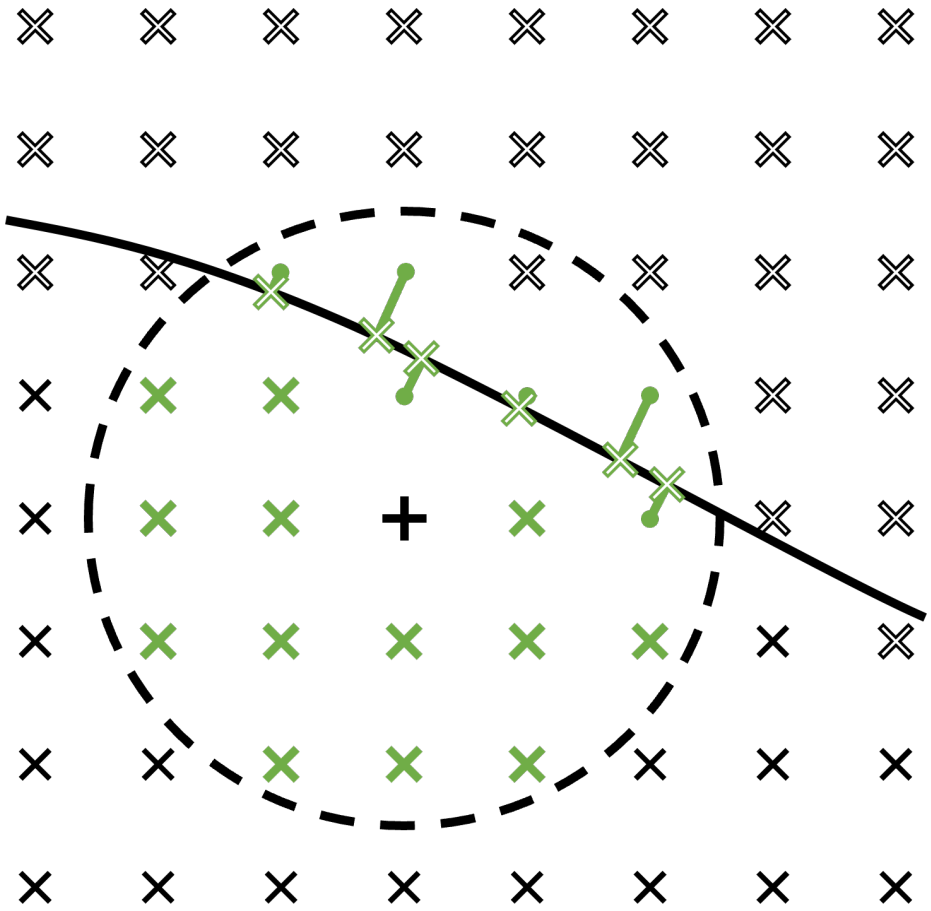


Figure 5: Construction of the support region in the vicinity of the boundary. Symbols are as in Figures 1 and 4. The solid black line represents the boundary surface. Hollow black crosses are considered to be outside or too close to the boundary for the purposes of constructing the extrapolant (as defined in the proceeding section).

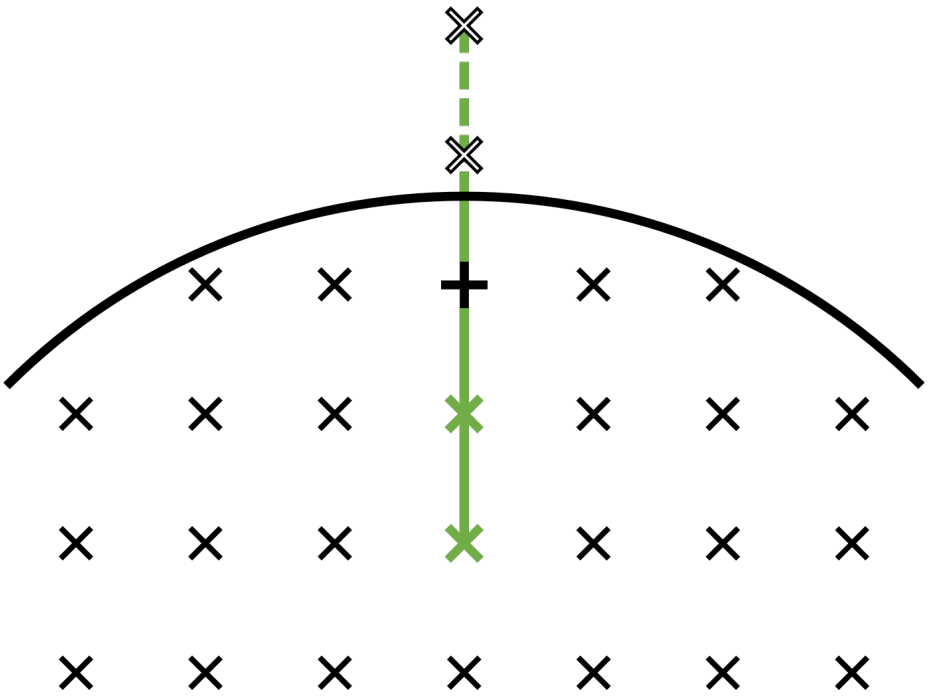


Figure 6: The stencil footprint of a 4th-order-accurate $\frac{\partial^2 p}{\partial y^2}$ stencil truncated by an arc-shaped boundary. The bold black cross is the stencil position, whilst pale green crosses are other interior points within the stencil. Values at both hollow black crosses are required by the stencil but are located outside the computational domain.

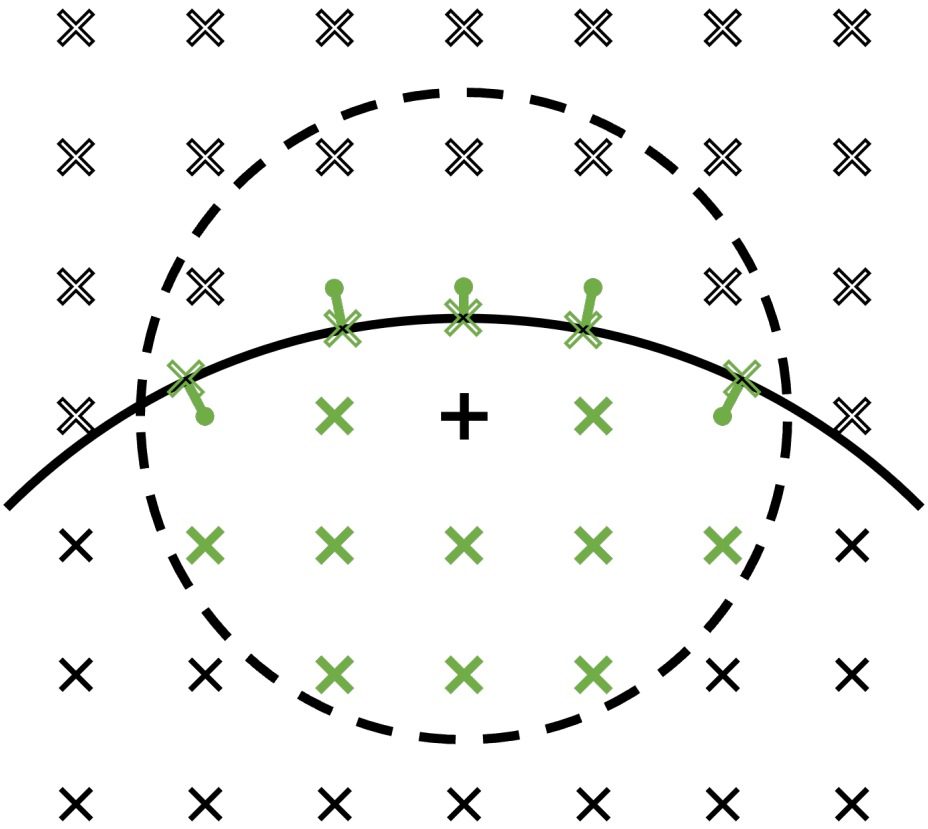
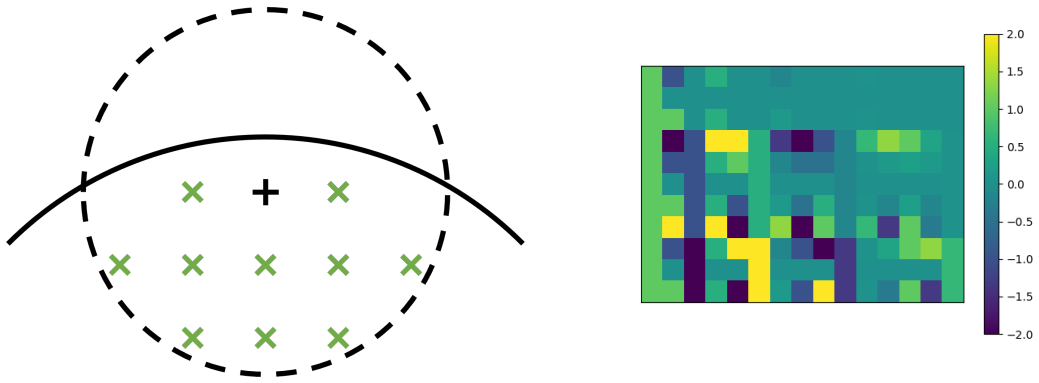


Figure 7: Support region for a 2D polynomial fitted with a combination of available interior points and boundary points.

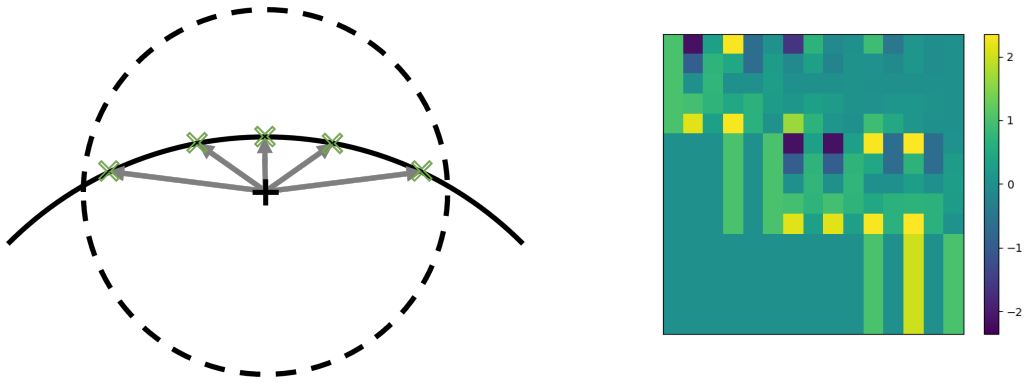
Caunt et al. –



(a) (b)

Figure 8: Interior points within the support region and structure of the corresponding submatrix. Rows in the matrix from top to bottom correspond to points within the support region working downwards from left to right.

Caunt et al. –



(a) (b)

Figure 9: Boundary points within the support region and structure of the corresponding submatrix. Grey arrows indicate the vector $\boldsymbol{x_b} - \boldsymbol{x}_0$ for these points. Rows in the matrix from top to bottom correspond to points from left to right.

Caunt et al. –

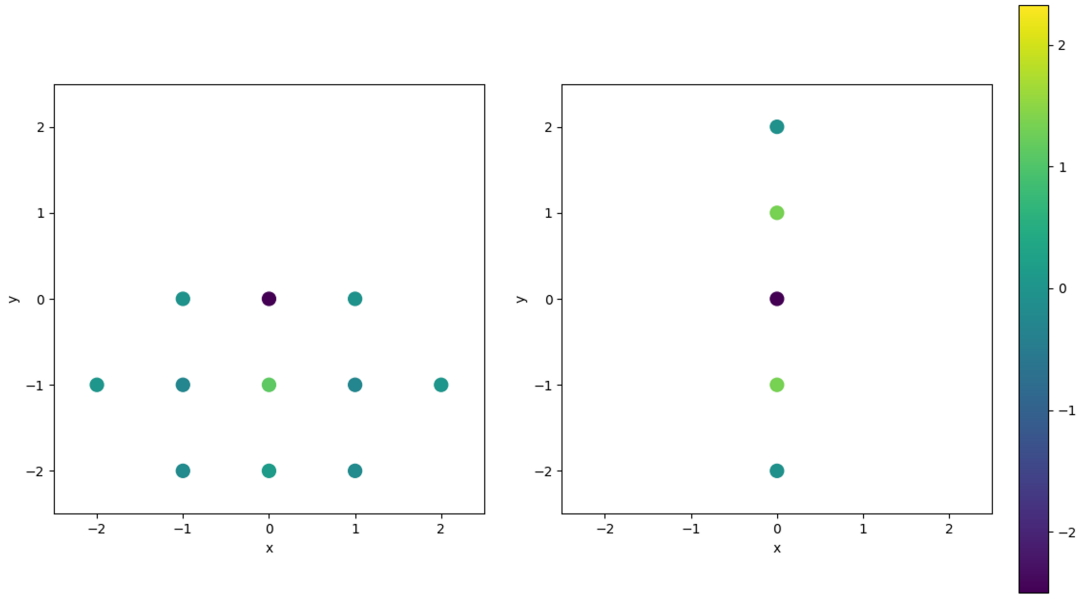


Figure 10: Comparison of the standard stencil footprint and weights to that of the modified boundary operator for the case illustrated in Figure 6. Colours indicate the values of stencil weights at each point.

Caunt et al. –

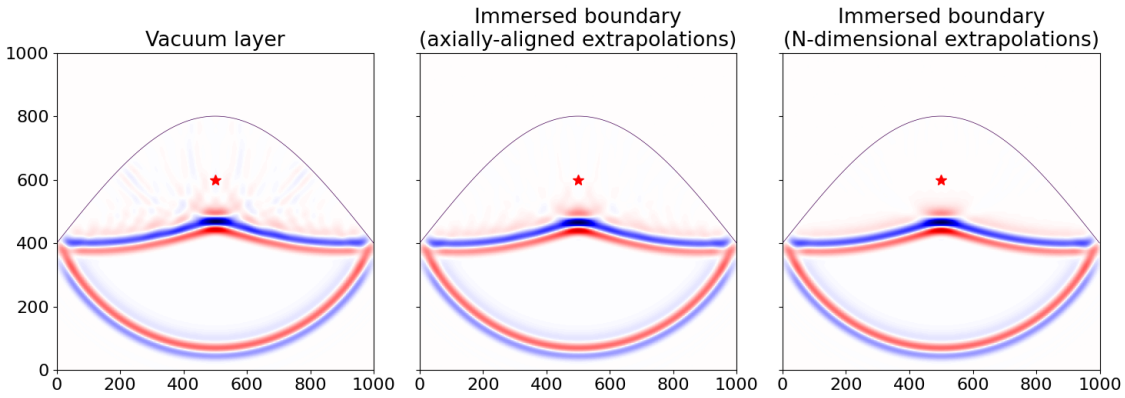
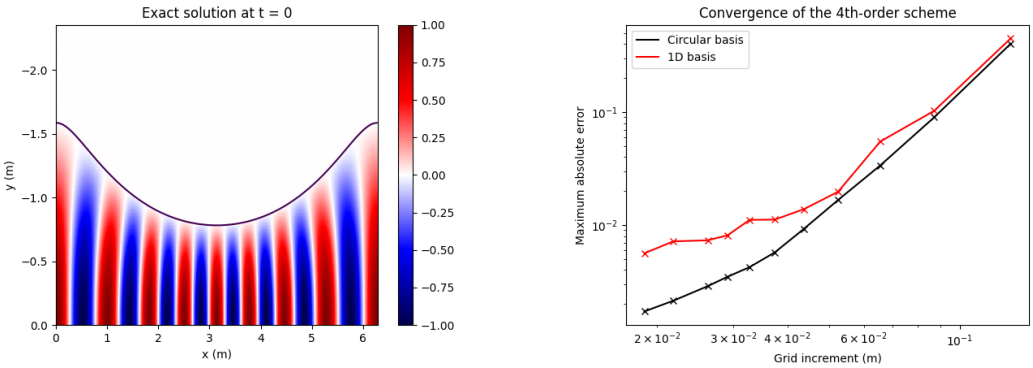


Figure 11: Snapshots at 700ms of a wavefront reflecting off a sinusoidal hill. The leftmost example implements the free surface as a vacuum layer, whilst the remaining subfigures impose this condition as an immersed boundary. The centre subfigure shows the results of an immersed boundary treatment based on axially-aligned 1D extrapolations as in Mulder (2017), whilst the rightmost subfigure shows the interaction of the wavefield with an immersed free surface based on N-dimensional extrapolants with circular support regions. Resolution and discretization is identical for all runs; only the boundary treatment is altered. The location of the Ricker source is marked with a red star. Wavefield amplitudes are normalised against the maximum absolute value in each subfigure. This convention is continued henceforth.



(a) (b)

Figure 12: The exact solution at $t = 0$ is shown in the left subfigure. The solid black line is the free surface, the left and right sides of the domain have periodic boundary conditions applied, and the wavefield is mirrored across the lower boundary. A comparison of the convergence behaviour of the method proposed to one using 1D approximations is shown on the right.

Caunt et al. –

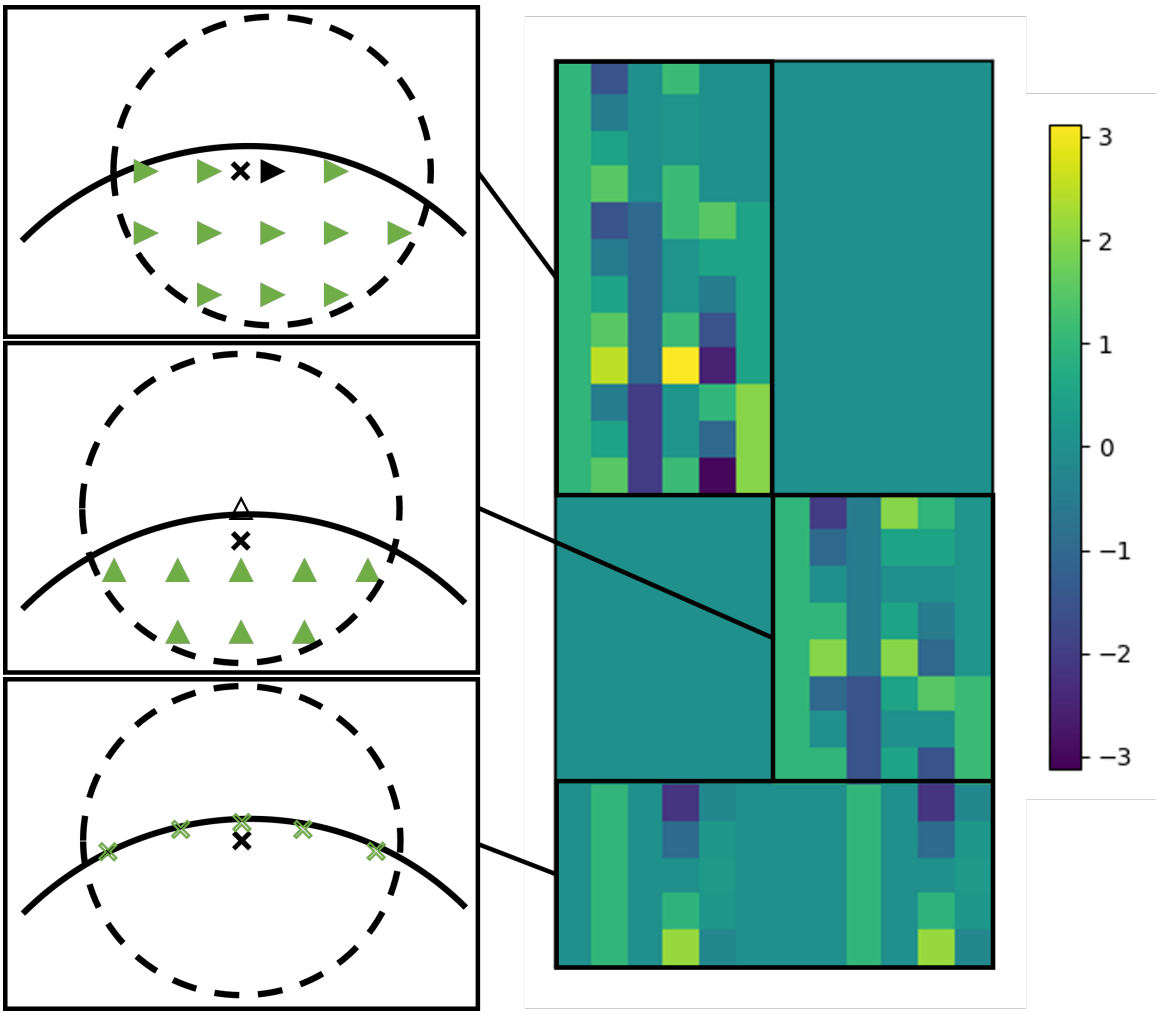


Figure 13: Assembling **A** for the particle velocity fields. Horizontal arrows correspond with v_x points used to construct the extrapolant, whilst vertical arrows are the respective v_y points. The black cross designates the expansion point. Note that only the subgrids of concern are shown for clarity. Note how the interior points of each field correspond with an independent block, whilst the boundary condition rows span both.

Caunt et al. –

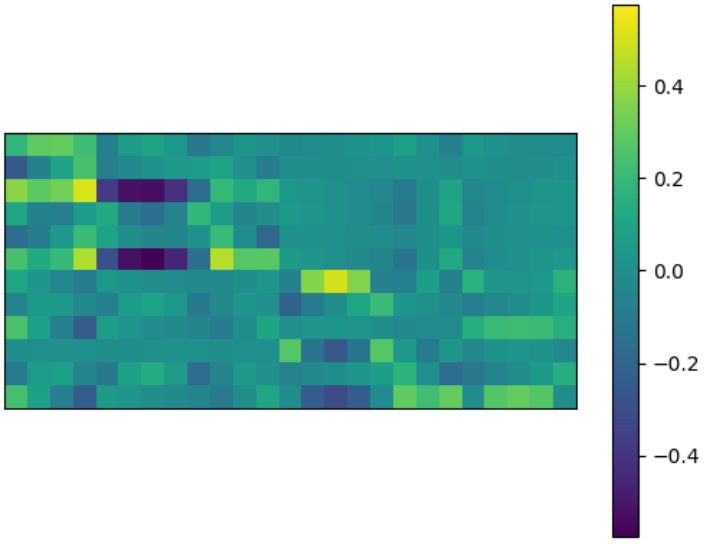


Figure 14: The structure of the Moore-Penrose pseudoinverse of **A** shown in Figure 13.

Caunt et al. –

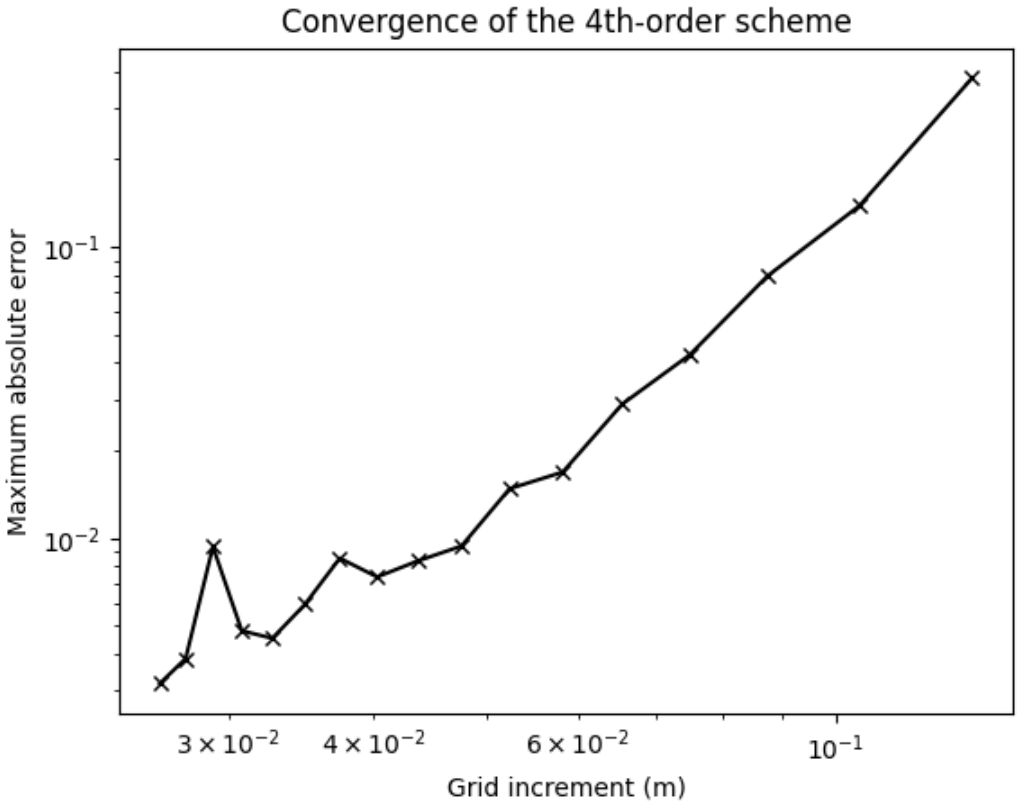


Figure 15: Convergence of the numerical scheme for the 1st-order formulation of the acoustic wave equation using the same setup shown in Figure 12a. Note that scales are slightly different in this figure.

Caunt et al. –

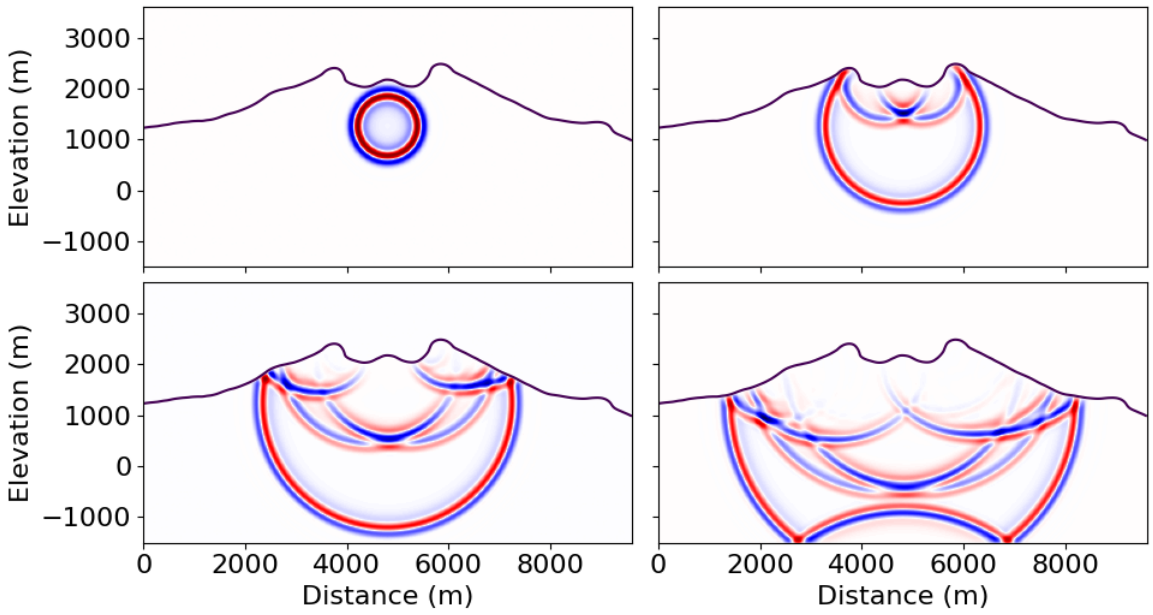


Figure 16: Snapshots of the 2D 2nd-order acoustic wavefield interacting with a free surface at 375ms, 750ms, 1125ms, and 1500ms. Celerity is 2.5km/s throughout the model. The black line designates the isosurface $s(x, y) = 0$ on the SDF, coinciding with the surface. The wavefield shown in each snapshot is normalised for clarity.

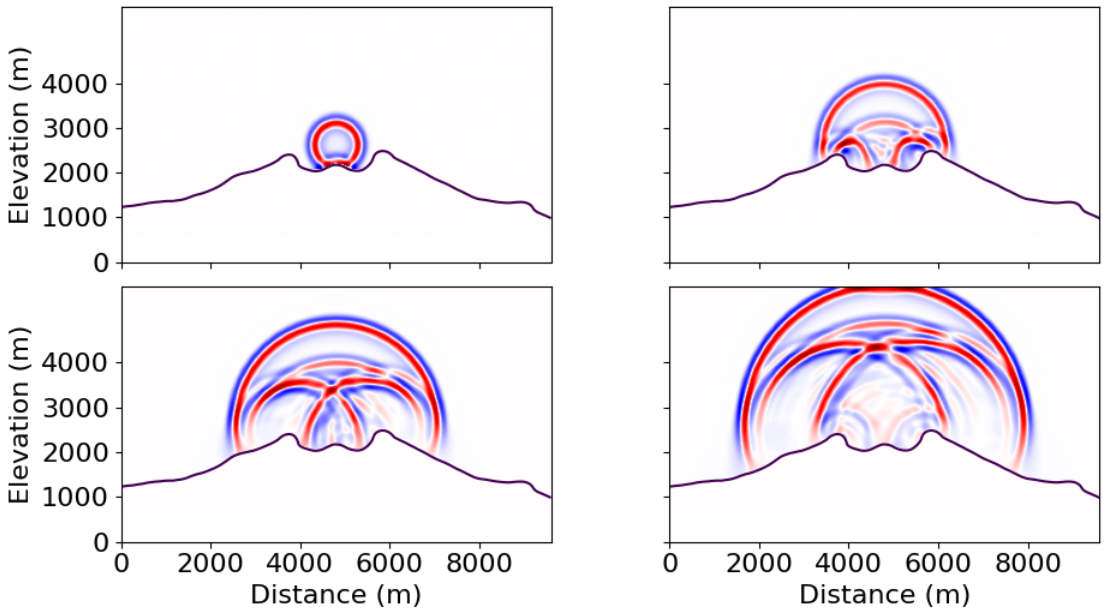


Figure 17: Snapshots of the 2D 2nd-order acoustic wavefield interacting with a rigid surface at 2.5s, 5s, 7.5s, and 10s. Celerity is 350m/s throughout the model. Parameters are altered in this example to better reflect infrasound propagation problems to which this boundary condition is applicable. The wavefield shown in each snapshot is normalised for clarity.

Caunt et al. –

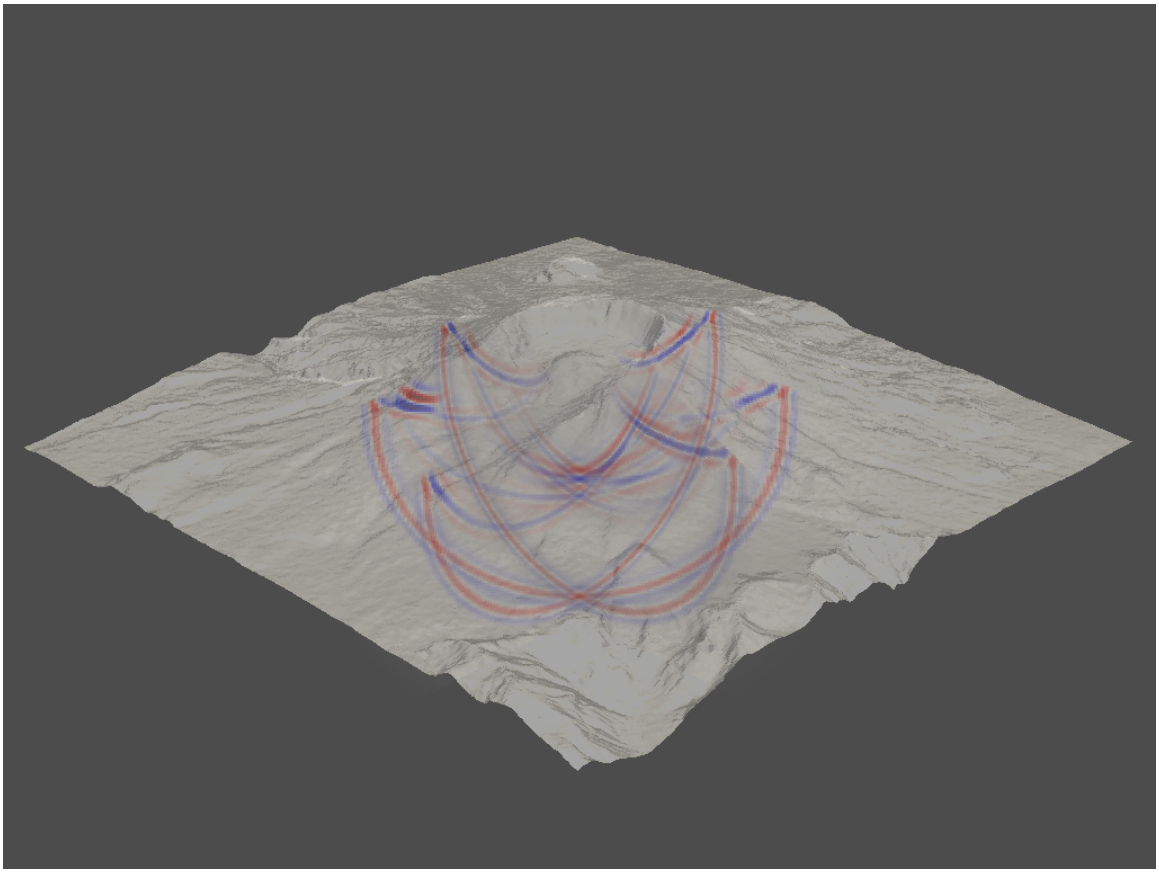


Figure 18: Render of the 3D 2nd-order acoustic wavefield at 1125ms, showing the free surface topography. Slices of the wavefield are shown aligned and diagonal to each compass direction for clarity. Wavefield transparency is scaled with amplitude to emphasise the wavefronts.

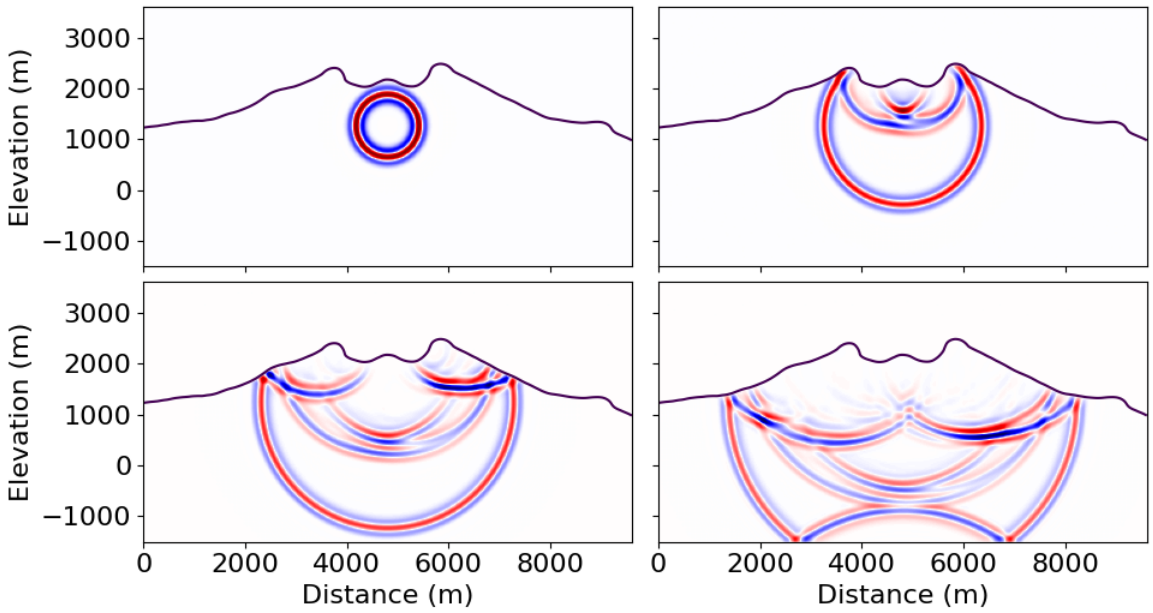


Figure 19: Slices through the 3D 2nd-order acoustic wavefield interacting with the free surface on the profile used for the 2D examples. Snapshots are at 375ms, 750ms, 1125ms, and 1500ms respectively. Celerity is 2.5km/s throughout the model. The black line designates the isosurface $s(x, y, z) = 0$ on the SDF, coinciding with the surface. The wavefield shown in each snapshot is normalised for clarity.

Caunt et al. –

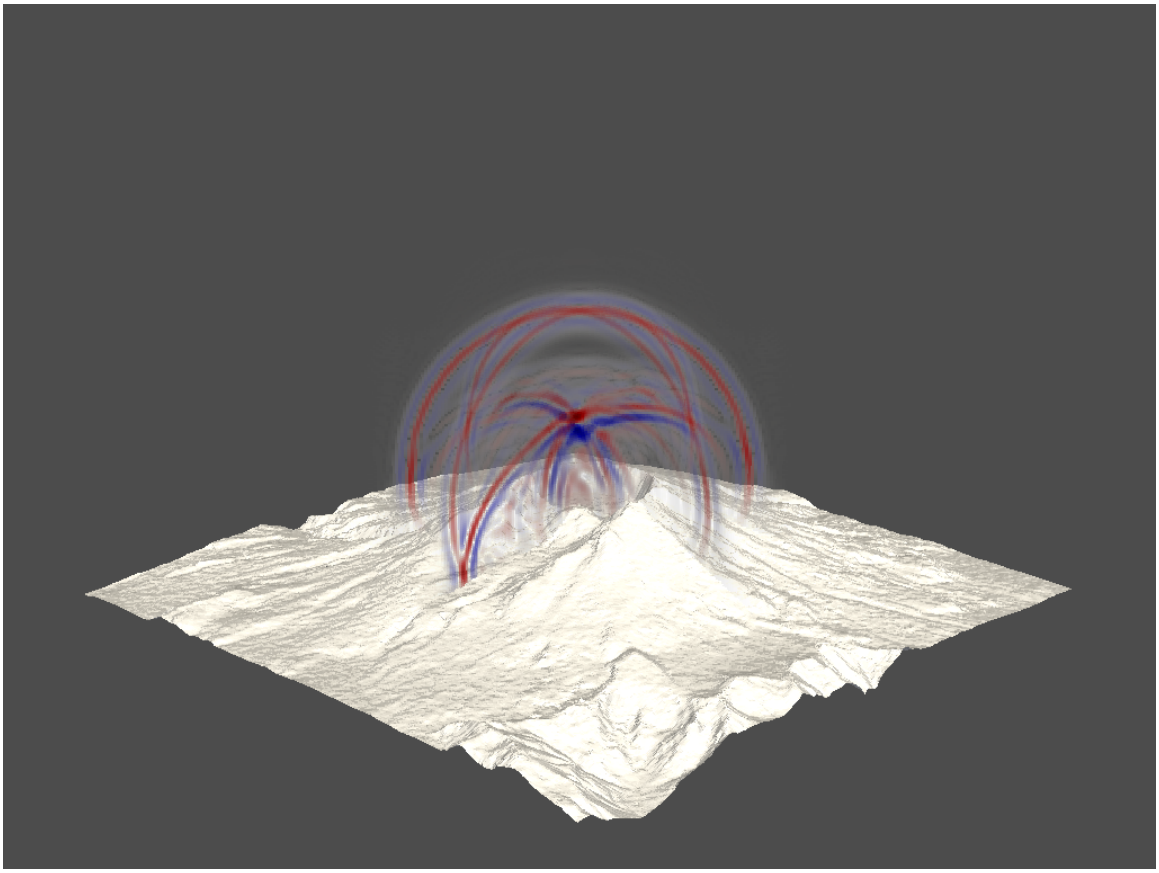


Figure 20: Render of the 3D 2nd-order acoustic wavefield at 7.5s, interacting with the topographic rigid surface. Slices of the wavefield are shown aligned and diagonal to each compass direction for clarity. Wavefield transparency is scaled with amplitude as before.

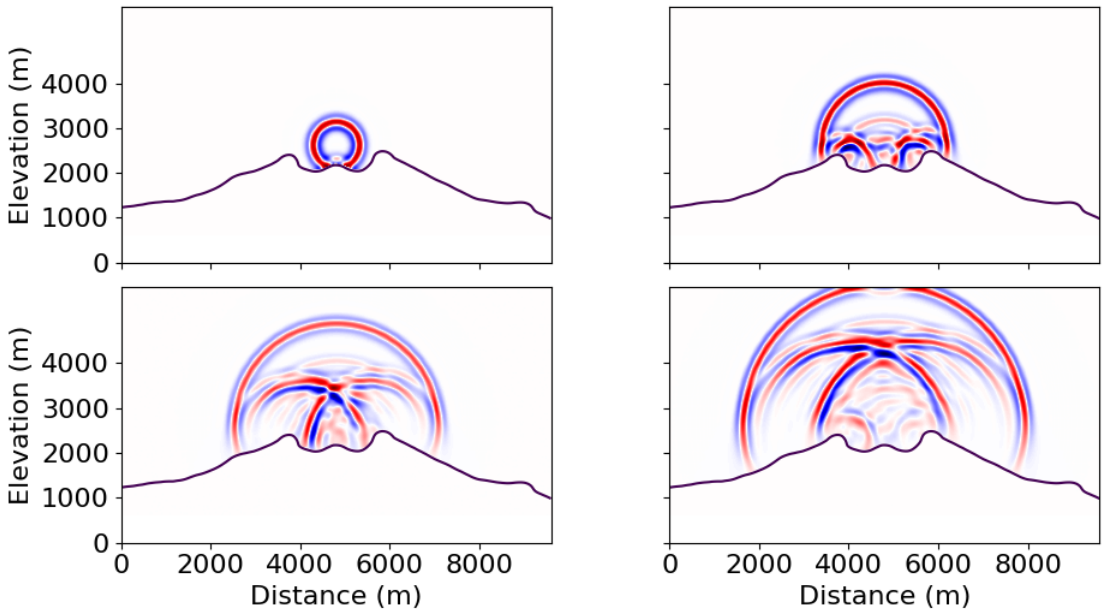


Figure 21: Snapshots of the 3D 2nd-order acoustic wavefield interacting with the rigid surface at 2.5s, 5s, 7.5s, and 10s. The transect is chosen to match that used for the 2D examples. Celerity is 350m/s throughout the model. The wavefield shown in each snapshot is normalised for clarity.

Caunt et al. –

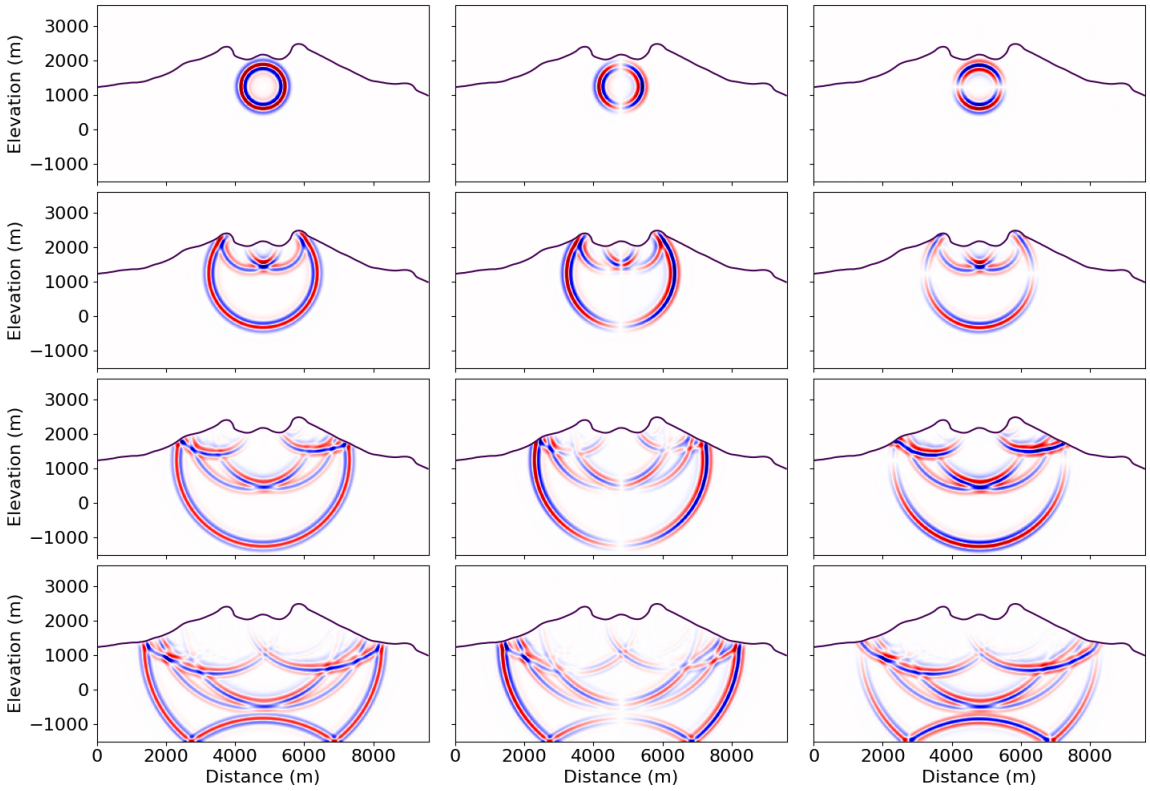


Figure 22: An irregular free surface implemented for the 1st-order acoustic wave equation in 2D. Snapshots of the pressure (left), x particle velocity (middle), and y particle velocity (right) wavefields at 375ms, 750ms, 1125ms, and 1500ms. Celerity is 2.5km/s throughout the model, density is homogeneous throughout. The black line designates the isosurface $s(x, y) = 0$ on the SDF, coinciding with the surface. The wavefields shown in each snapshot are normalised for clarity.

Caunt et al. –

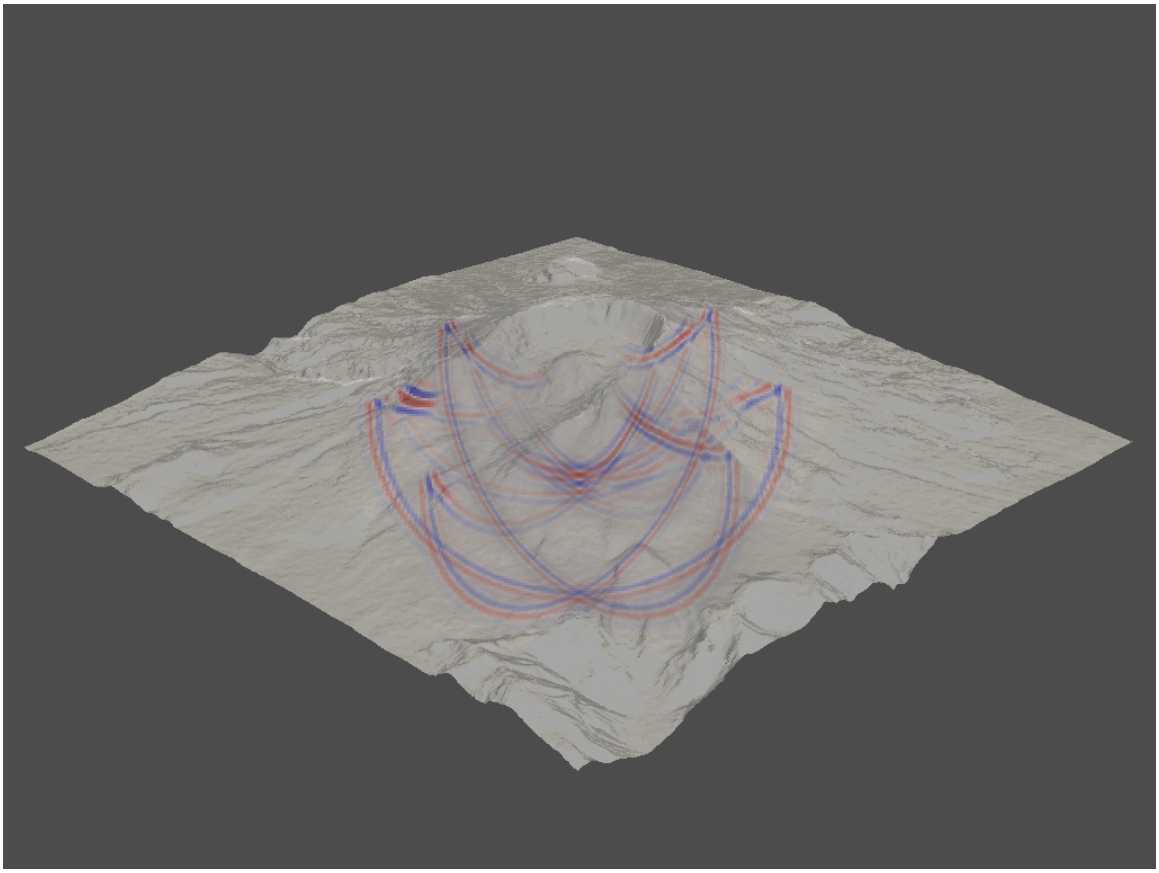


Figure 23: Render of the pressure wavefield at 1125ms when solving the 3D 1st-order acoustic wave equation in the presence of free surface topography. Slices of the wavefield are shown aligned and diagonal to each compass direction for clarity. Wavefield transparency is scaled with amplitude to emphasise the wavefronts.

Caunt et al. –

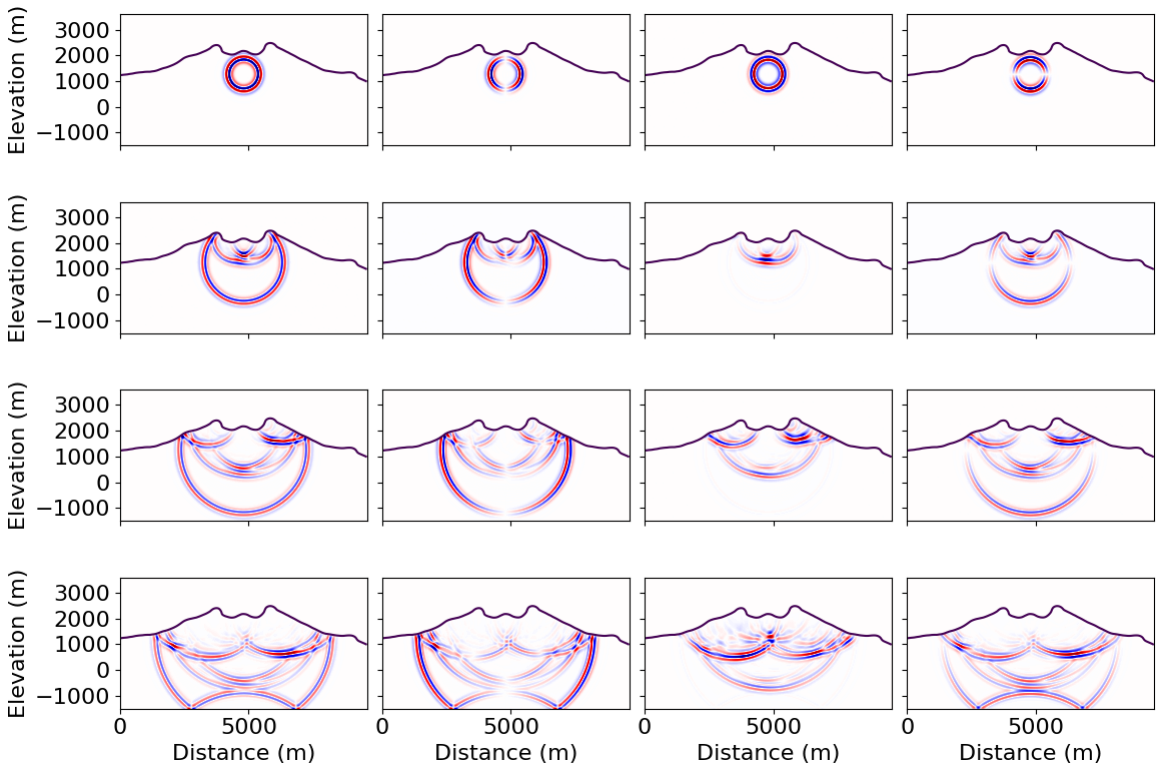


Figure 24: Snapshots of the pressure (left), x particle velocity (middle left), y particle velocity (middle right), and z particle velocity (right) wavefields at 375ms, 750ms, 1125ms, and 1500ms. The ty-axis is oriented into the page. Celerity is 2.5km/s throughout the model, density is homogeneous throughout. The black line designates the isosurface $s(x, y, z) = 0$ on the SDF, coinciding with the surface. The wavefields shown in each snapshot are normalised for clarity.

Caunt et al. –

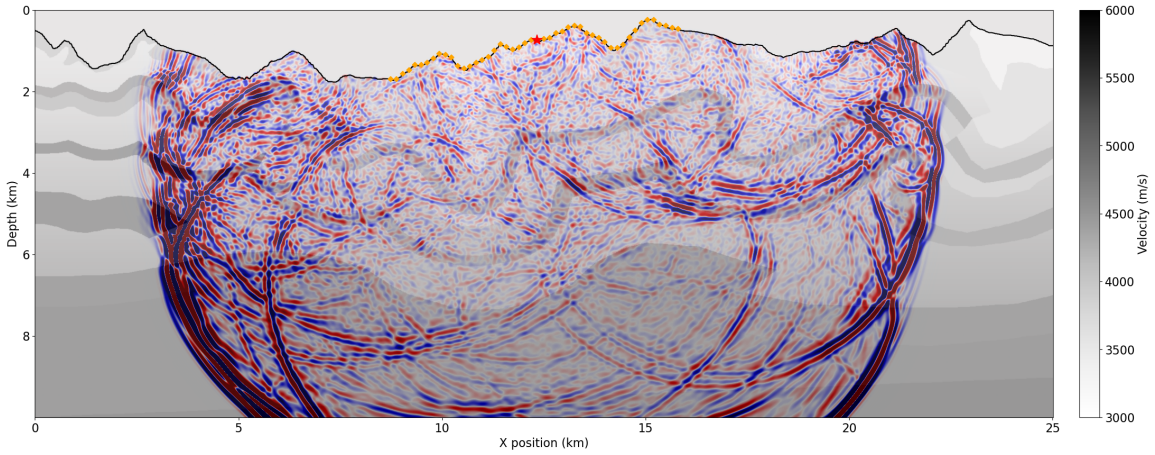


Figure 25: A snapshot of the wavefield propagating through the 1994 BP Migration from Topography synthetic at 3000ms. The free surface, shown as a black line, is implemented using the same immersed boundary treatment as demonstrated in Figure 16. The source location is shown as a red star, whilst decimated receivers used to generate Figure 26 are shown in orange.

Caunt et al. –

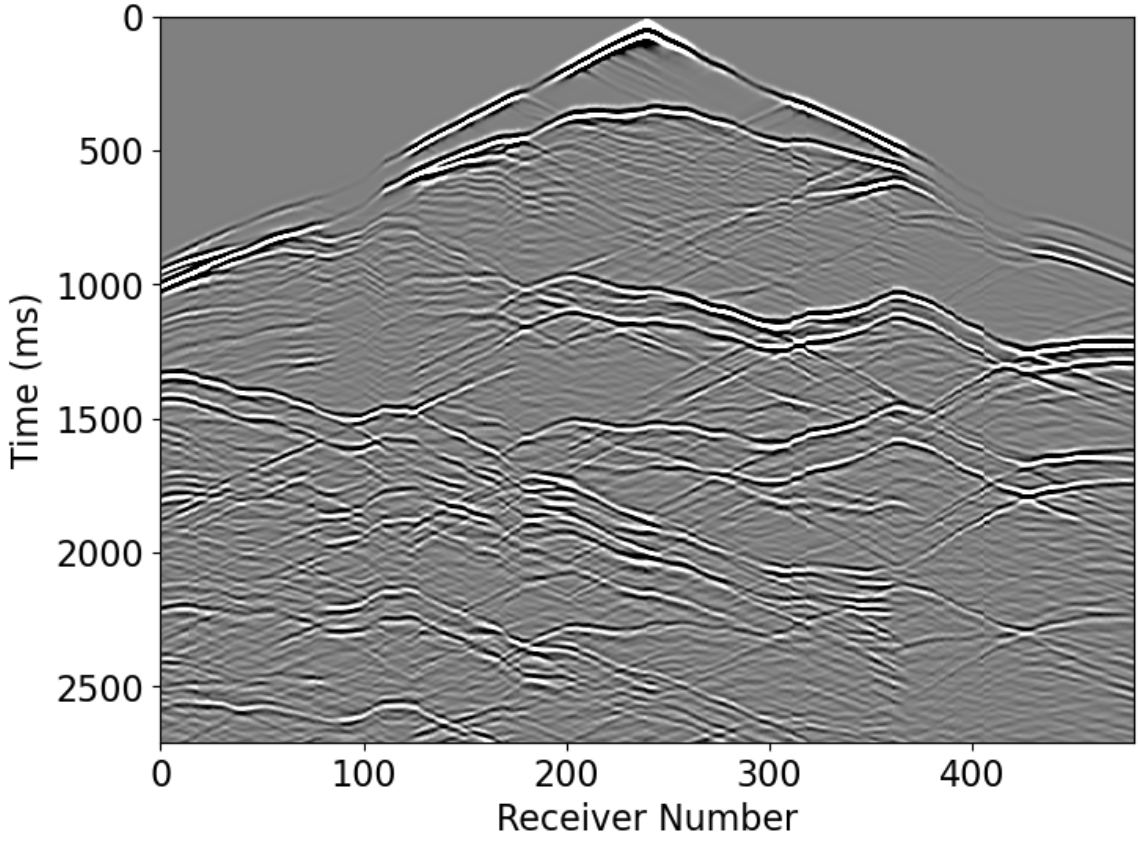
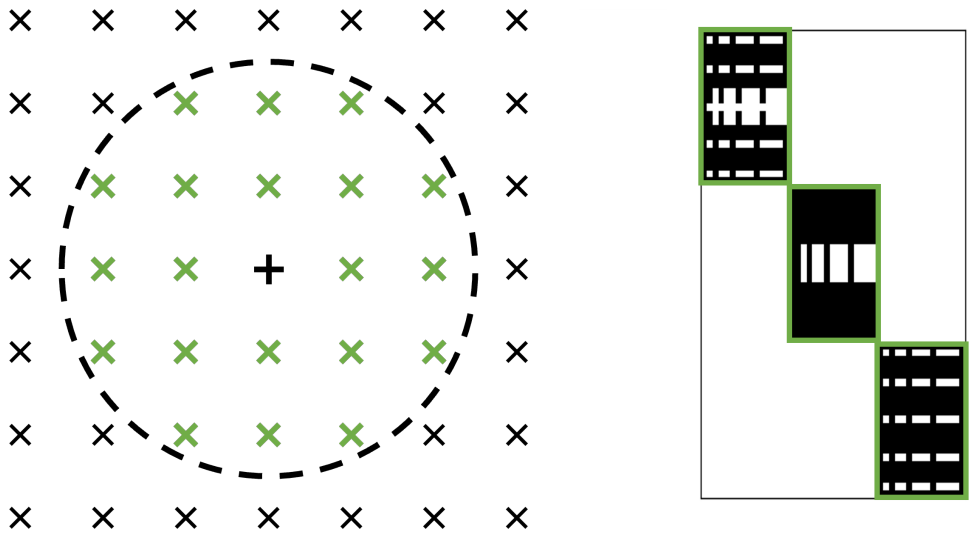


Figure 26: The shot gather for the geometry shown in Figure 25. The minimal noise, smoothness of the reflectors, and clearly defined diffractions demonstrate the accuracy achieved by the immersed boundary approach.

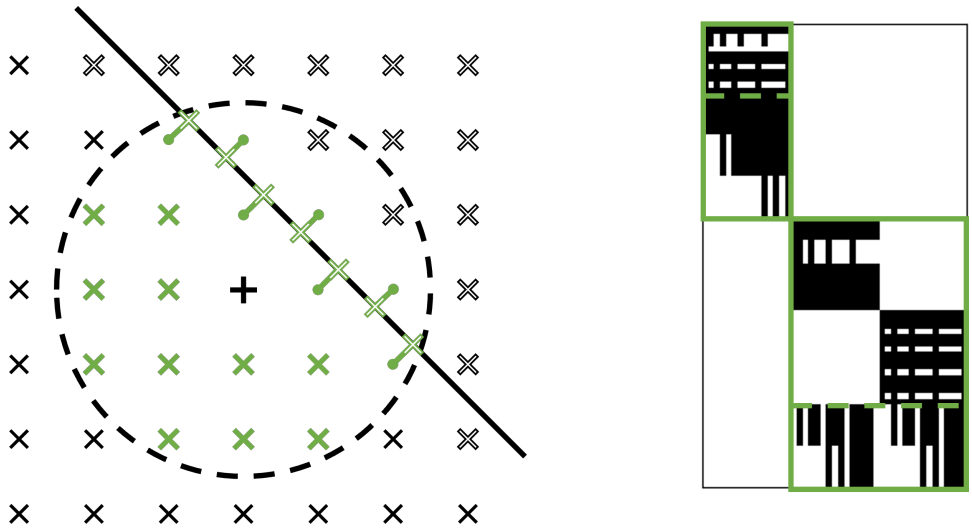
Caunt et al. –



(a) (b)

Figure A-1: A circular support region in free space and its corresponding matrix structure for the first-order acoustic wave equation. Nonzero elements are black, zero elements are white. Staggered particle velocity subgrids are omitted to prevent excessive cluttering. Individual blocks are highlighted in green, and from top-left to bottom-right correspond to pressure, horizontal particle velocity, and vertical particle velocity. It is clear that this matrix can be split into three smaller systems which can be solved individually; each polynomial can be fitted independently of the others.

Caunt et al. –



(a) (b)

Figure A-2: A circular support region cut by a diagonal boundary and its corresponding matrix structure for the first-order acoustic wave equation. Nonzero elements are black, zero elements are white. Staggered particle velocity subgrids are omitted to prevent excessive cluttering. Blocks are highlighted with solid green boxes, with the dashed green lines separating rows corresponding to interior points and those corresponding to boundary conditions. Note that the previously separate particle velocity blocks have become merged, as the zero-divergence boundary condition and corresponding higher-order conditions span both particle velocity fields.

Caunt et al. –