Commun. Comput. Phys.
doi: 10.4208/cicp.041109.120210a

Vol. **8**, No. 5, pp. 1074-1110
November 2010

Stable Grid Refinement and Singular Source Discretization for Seismic Wave Simulations

N. Anders Petersson* and Björn Sjögreen

Center for Applied Scientific Computing, L-422, Lawrence Livermore National Laboratory, PO Box 808, Livermore, CA 94551.

Received 4 November 2009; Accepted (in revised version) 12 February 2010

Available online 31 May 2010

Abstract. An energy conserving discretization of the elastic wave equation in second order formulation is developed for a composite grid, consisting of a set of structured rectangular component grids with hanging nodes on the grid refinement interface. Previously developed summation-by-parts properties are generalized to devise a stable second order accurate coupling of the solution across mesh refinement interfaces. The discretization of singular source terms of point force and point moment tensor type are also studied. Based on enforcing discrete moment conditions that mimic properties of the Dirac distribution and its gradient, previous single grid formulas are generalized to work in the vicinity of grid refinement interfaces. These source discretization formulas are shown to give second order accuracy in the solution, with the error being essentially independent of the distance between the source and the grid refinement boundary. Several numerical examples are given to illustrate the properties of the proposed method.

AMS subject classifications: 65M06, 65M12, 74B05, 86A15

Key words: Elastic wave equation, mesh refinement, stability, summation by parts, singular source term.

1 Introduction

The finite difference method on a uniform Cartesian grid is a highly efficient and easy to implement technique for solving the elastic wave equation in seismic applications [12, 20, 32]. However, the spacing in a uniform Cartesian grid is fixed throughout the computational domain, whereas the resolution requirements in realistic seismic simulations

*Corresponding author. *Email addresses:* andersp@llnl.gov (N. A. Petersson), sjogreen2@llnl.gov (B. Sjögreen)

usually are higher near the surface than at depth. This can be seen from the well-known formula

$$h \leq L/P,$$

which relates the grid spacing h to the wave length L, and the required number of grid points per wavelength P for obtaining an accurate solution [15]. The compressional and shear wave lengths in the earth generally increase with depth and are often a factor of ten larger below the Moho discontinuity (at about 30 km depth), than in sedimentary basins near the surface. A uniform grid must have a grid spacing based on the small wave lengths near the surface, which results in over-resolving the solution at depth. As a result, the number of points in a uniform grid is unnecessarily large.

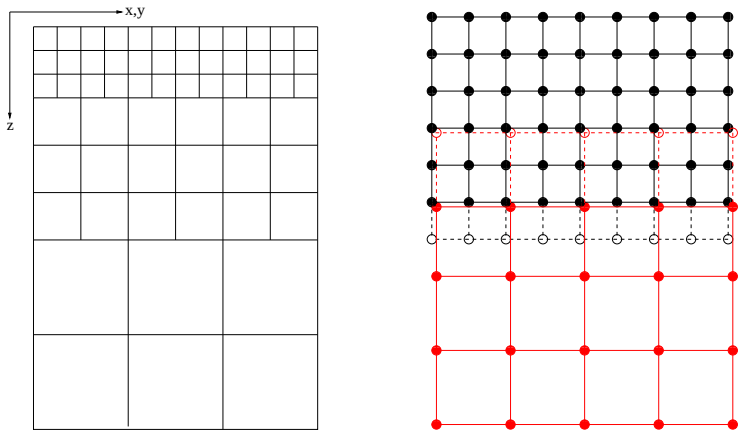


Figure 1: (a) Outline of a 2-D cross-section of the 3-D computational domain with grid refinements. (b) Close up of a grid refinement interface, where interior points are drawn with filled circles and ghost points with open circles. The grids have been plotted with a small offset to clarify the grid layout.

In this paper, we address the over-resolution-at-depth issue by generalizing the single grid finite difference scheme described in [23] to work on a composite grid consisting of a set of structured rectangular grids of different spacings, as outlined in Fig. 1(a). The computational domain in a regional seismic simulation often extends to depth 40-50 km. Hence, using a refinement ratio of two, we need about three grid refinements from the bottom of the computational domain to the surface, to keep the local grid size in approximate parity with the local wave lengths. Generating the composite grid is trivial once the locations of the component grids have been determined, and the resulting composite grid has ideal wave propagation properties due to its perfect regularity.

The composite grid discretization developed here, together with the generalization of the method to curvilinear grids [2], which enables accurate modeling of free surfaces on realistic (non-planar) topography, makes the finite difference method a very attractive alternative to the recently developed finite element [7], spectral element [16], discontinuous Galerkin [10, 14], and finite volume [11] discretizations on unstructured grids. An

unstructured grid can be made to follow realistic topography and can in principle also be made to follow arbitrarily complex shapes in the underlying 3-D geological structure, with a grid size that gets larger as the material wave speed increases. However, generating a high quality unstructured 3-D grid can be a difficult task in itself. Compared to a finite difference method on a structured grid, the unstructured nature of the grid also requires extra book keeping and additional memory to keep track of the connectivity in the grid, making an efficient implementation of these methods more challenging, in particular on massively parallel machines.

The challenge of the composite grid approach is to find a stable and accurate method for coupling the solution across the grid refinement interface. Of particular importance is the treatment of the solution at the hanging nodes, i.e., the fine grid points which are located in between coarse grid points. A natural way of advancing the solution in time on a composite grid is to first use the difference scheme to update the solution at all interior points, followed by assigning solution values to the ghost points through interpolation from the neighboring grid (see Fig. 1(b)). For example, this approach works well for many other problems in conjunction with finite difference methods on overlapping grids [6]. We call this the non-conservative interpolation technique. Because it is based on interpolation, the non-conservative technique is only accurate if the solution is smooth across the refinement interface. Furthermore, when applied to wave propagation problems, the non-conservative interpolation sometimes induces numerical instabilities, originating from the grid interface. To stabilize the non-conservative method, one can try adding artificial dissipation to the finite difference scheme. However, in our experience, it is very difficult to automatically choose a strength of the dissipation term and a time step that guarantees a stable numerical solution for realistic heterogeneous materials.

To overcome the problems with non-conservative interpolation, this article presents a new, energy conserving, coupling procedure for the elastic wave equation at grid refinement interfaces. When used together with the difference scheme of [23], it results in a method that is provably stable, without artificial dissipation, for arbitrary heterogeneous isotropic elastic materials. We call this the conservative interpolation technique. The new coupling procedure is based on satisfying the summation-by-parts principle across refinement interfaces. From a practical standpoint, an important advantage of the proposed method is the absence of tunable numerical parameters, which are seldom appreciated by application experts. The non-dissipative character of the method also makes it well suited for wave propagation over large distances, where dissipation otherwise can lead to significant amplitude errors.

The performance of the different interpolation techniques can be illustrated by solving a layer over half-space problem where the grid refinement interface is aligned with the material discontinuity. The details of this experiment are described in Section 6.4. Here we outline some of the results as a motivating example. Fig. 2 displays the error at a point on the surface as function of time. The upper half of the figure illustrates that the error with non-conservative interpolation (shown in red) is larger than the error when the energy conservative interface treatment is used (shown in blue). This is due to the so-

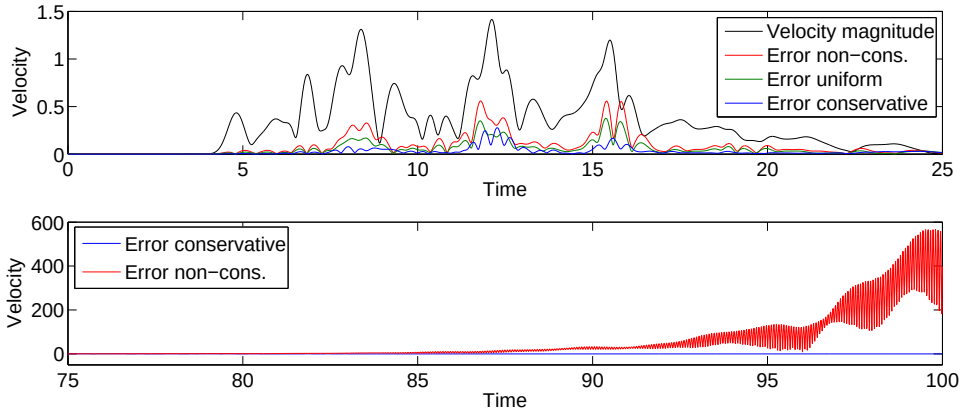


Figure 2: (Top): The velocity magnitude as function of time at $\mathbf{x}_r = (3.1\times 10^4, 3.3\times 10^4, 0)^T$ in the solution of a layer over half-space problem due to a moment tensor source located at $\mathbf{x}_* = (2.5\times 10^4, 2.5\times 10^4, 2.55\times 10^3)^T$ (black). The errors in the numerical approximations correspond to conservative interpolation (blue), non-conservative interpolation (red), and a uniform fine grid (green). (Bottom): Errors at later times. See Section 6.4 for details.

lution having a discontinuous gradient across the material discontinuity. The lower half of Fig. 2 shows that for later times, the method based on non-conservative interpolation (shown in red) becomes unstable, while the energy conserving approach remains stable (shown in blue).

Conservative grid interface conditions for finite volume approximations of first order hyperbolic systems, in particular the Euler equations of gas dynamics, have been extensively studied for a long time [4, 5, 26, 27]. In these articles the objective is to conserve integrals of the dependent variables, such as the mass, across boundaries between grids of different refinement, or between grids from different curvilinear mappings. The conserved integrals are not norms of the solution, and consequently the conservation property in this context does not automatically lead to stability estimates. Instead conservation is needed for obtaining correct shock propagation speeds. Stability at mesh interfaces is usually studied separately for linearized versions of these equations by the so called GKS analysis, see for example [25, 30].

For linear hyperbolic equations, which for example govern acoustic or elastic wave propagation, the energy is a norm of the discrete solution (usually under a CFL-condition), and therefore, energy conservation leads to stability. Energy estimation with grid interfaces for finite difference methods can be done by summation by parts difference operators. The summation by parts technique has been used extensively for analyzing interfaces for first order hyperbolic problems, both for sudden changes in grid spacing and for discontinuous coefficients in the equation, see for example [17, 24] and the references therein. However these articles are limited to constant coefficient problems in one space dimension, thereby not addressing either the difficulties associated with variable coefficients, or the difficulties associated with hanging nodes in several space di-

mensions. An exception is the recent two-dimensional summation by parts discretization for grid refinement interfaces developed in [18]. That paper considers mesh refinement interfaces with hanging nodes, including effects of corners where two refinement interfaces meet. Results from solving a scalar constant coefficient advection problem in two space dimensions shows some degeneracy of the grid convergence rate, due to reduced order of accuracy of the discretization at the corners of the refinement patches.

Material jumps for the acoustic wave equation in second order formulation was handled by the summation by parts technique in [21], but also here as a purely one dimensional procedure, which however was applied to an unstructured grid discretization in three space dimensions.

An interface condition for the acoustic wave equation in two space dimensions was presented in [3], where stability was proven for a finite element discretization. In this work, discontinuous material properties was treated together with a grid refinement interface with hanging nodes. Although the finite element formulation makes the analysis very different, the method in [3] is close in spirit to our proposed approach.

Seismology researchers have also constructed algorithms for coupling finite difference schemes for the elastic wave equation across grid refinement interfaces. These approaches are given without a proof of their stability, but have been demonstrated to work reasonably well in realistic seismic applications, see for example [35].

In the second part of this paper, we study the discretization of singular source terms in the vicinity of grid refinement interfaces. These source terms are essential in seismic applications where they are used to model time-dependent slip on an earthquake fault. In larger earthquakes the slip occurs over an extended area of the fault surface, and source terms can be distributed over this area to model variable amounts of slip and variations in the source time function, according to the earthquake source model. Because of a potentially significant variation in material wave speed over a fault surface, it is desirable to allow fault surfaces to extend through grid refinement interfaces. This means that some point sources may be located near a refinement interface, in which case our previous single grid source discretization technique based on Waldén's formulas [33] does not apply. In the present paper, we generalize the source discretization formula by enforcing certain moment conditions, which mimic properties of the Dirac distribution and its gradient, leading to a second order accurate discretizations of point forces and point moment tensor sources which are located anywhere on the computational grid, in particular near grid refinement interfaces. The discretization is first derived in one space dimension, both on a uniform and a composite grid near a grid refinement interface. In three dimensions we use a Cartesian product of one-dimensional singular source discretizations, which guarantees that the corresponding 3-D moment conditions are satisfied. We find by numerical experiments that the solution with a singular source term becomes second order accurate independently of its location relative to the grid, both away from a grid refinement interface and in its vicinity.

The remainder of the paper is organized as follows. In Section 2, we derive stability properties of a finite difference approximation of the 1-D scalar wave equation with

discontinuous wave speed on a composite grid. Relevant stability properties of our previous single-grid discretization of the elastic wave equation are presented in Section 3, followed by a derivation of the energy conserving grid refinement coupling for the elastic wave equation in Section 4. An accurate way of discretizing point force and moment tensor sources located anywhere on the grid, in particular close to a grid refinement interface, is presented in Section 5. Numerical experiments are given in Section 6, followed by concluding remarks in Section 7.

2 The scalar wave equation in 1-D

We start by illustrating the energy conserving interpolation technique for the scalar wave equation with variable wave speed $c(x)=\sqrt{\mu(x)}$ in one space dimension

$$u_{tt}=\big(\mu(x)u_x\big)_x, \quad \mu(x)>0, \quad -\infty<x<\infty, \quad t\geq 0. \tag{2.1}$$

The solution $u(x,t)$ is subject to initial conditions

$$u(x,0)=\phi_1(x), \qquad u_t(x,0)=\phi_2(x),$$

and is assumed to tend to zero as $x\to\pm\infty$ such that the integrals in the following estimates become bounded.

By multiplying (2.1) by u_t and integrating over the spatial interval $a_1<x<a_2$, we get the identity

$$\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\int_{a_1}^{a_2} u_t^2(x,t)+\mu(x)u_x^2(x,t)\,\mathrm{d}x=u_t(a_2,t)\mu(a_2)u_x(a_2,t)-u_t(a_1,t)\mu(a_1)u_x(a_1,t). \tag{2.2}$$

Assume that (2.1) is solved over the subdomains $-\infty<x<0$ and $0<x<\infty$, and denote the corresponding solutions by u and U, respectively. Relation (2.2) shows that

$$\begin{aligned}&\frac{1}{2}\frac{\mathrm{d}}{\mathrm{d}t}\Big(\int_{-\infty}^{0} u_t^2(x,t)+\mu(x)u_x^2(x,t)\,\mathrm{d}x+\int_0^{\infty} U_t^2(x,t)+\mu(x)U_x^2(x,t)\,\mathrm{d}x\Big)\\ &=u_t(0,t)\mu(0)u_x(0,t)-U_t(0,t)\mu(0)U_x(0,t),\end{aligned} \tag{2.3}$$

where $U_t(0,t)$ and $\mu(0)U_x(0,t)$ should be interpreted as limits of $U_t(x,t)$ and $\mu(x)U_x(x,t)$ when $x\to 0$ from the right $(x>0)$. Similarly, $u_t(0,t)$ and $\mu(0)u_x(0,t)$ are the corresponding limits as $x\to 0$ from the left $(x<0)$. It follows from (2.3) that the energy

$$\int_{-\infty}^{0} u_t^2(x,t)+\mu(x)u_x^2(x,t)\,\mathrm{d}x+\int_0^{\infty} U_t^2(x,t)+\mu(x)U_x^2(x,t)\,\mathrm{d}x, \tag{2.4}$$

is conserved in time if we impose the interface conditions

$$u(0,t)=U(0,t), \tag{2.5a}$$

$$\lim_{\epsilon\to 0+}\mu(-\epsilon)u_x(-\epsilon,t)=\lim_{\epsilon\to 0+}\mu(\epsilon)U_x(\epsilon,t). \tag{2.5b}$$

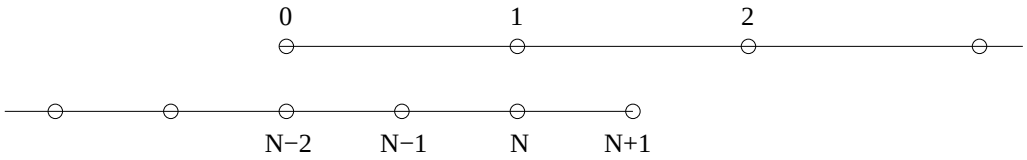


Figure 3: The composite grid in one space dimension.

Note that if $\mu(x)$ is discontinuous at $x=0$, the solution is continuous but its gradient is discontinuous.

We proceed by deriving corresponding interface condition for a finite difference discretization of the wave equation. We discretize (2.1) on the grid configuration shown in Fig. 3, where the grid refinement interface is located at $x=0$. The fine grid has grid spacing $h>0$, the coarse grid has grid spacing $2h$, and the grid points are located at

$$\begin{aligned} x_j &= (j-N)h, \quad j \le N+1, \\ X_j &= (j-1)2h, \quad j \ge 0. \end{aligned}$$

Note that the points X_0 and x_{N+1} are ghost points, which are used to simplify the enforcement of the interface conditions. Time is discretized on a uniform grid $t_n = n\delta_t$ with time step $\delta_t > 0$.

We denote the variables on the coarse grid by upper case letters, and the variables on the fine grid by lower case letters, i.e.,

$$U_j^n = U(X_j,t_n), \quad \text{and} \quad u_j^n = u(x_j,t_n).$$

The composite grid function over the entire domain is denoted

$$\bar{u} = (u,U)^T, \quad u = (\cdots, u_{N-1}, u_N)^T, \quad U = (U_1, U_2, \cdots,)^T. \tag{2.6}$$

The discrete 1-D scalar product and norm for grid functions u, v on a grid with size h are defined by

$$\begin{aligned} (u,v)_{1,h,p,q} &= \frac{h}{2}u_p v_p + h\sum_{i=p+1}^{q-1} u_i v_i + \frac{h}{2}u_q v_q, \\ \|u\|_{1,h,p,q}^2 &= (u,u)_{1,h,p,q}. \end{aligned}$$

At interior grid points, the discretization of the wave equation (2.1) is

$$\frac{u_j^{n+1}-2u_j^n+u_j^{n-1}}{\delta_t^2} = D_-(h)\big(E_{1/2}\mu(x_j)D_+(h)u_j^n\big), \qquad j \le N, \tag{2.7}$$

$$\frac{U_j^{n+1}-2U_j^n+U_j^{n-1}}{\delta_t^2} = D_-(2h)\big(E_{1/2}\mu(X_j)D_+(2h)U_j^n\big), \quad j \ge 1, \tag{2.8}$$

where the usual divided difference and averaging operators are defined by

$$D_+(h)u_j=\frac{u_{j+1}-u_j}{h},\quad D_-(h)u_j=\frac{u_j-u_{j-1}}{h},\quad E_{1/2}\mu(x_j)=\frac{\mu(x_{j+1})+\mu(x_j)}{2}.$$

The dependence on the grid and its size will be suppressed when the meaning is obvious. On a uniform grid, all grid functions u_j, v_j, and μ_j satisfy (see the appendix of [28])

$$\begin{aligned}&\big(u,D_-((E_{1/2}\mu)D_+v)\big)_{1,h,p,q}\\ =&-h\sum_{j=p}^{q-1}(E_{1/2}\mu_j)(D_+u_j)(D_+v_j)-\frac{u_p}{2}\big[(E_{1/2}\mu_{p-1})D_+v_{p-1}+(E_{1/2}\mu_p)D_+v_p\big]\\ &+\frac{u_q}{2}\big[(E_{1/2}\mu_{q-1})D_+v_{q-1}+(E_{1/2}\mu_q)D_+v_q\big].\end{aligned}\tag{2.9}$$

This identity can be applied separately on each grid in the composite grid to show

$$\begin{aligned}&\big(u,D_-(mD_+v)\big)_{1,h,-\infty,N}+\big(U,D_-(MD_+V)\big)_{1,2h,1,\infty}\\ =&-h\sum_{j=-\infty}^{N-1}m_{j+1/2}(D_+u_j)(D_+v_j)-2h\sum_{j=1}^{\infty}M_{j+1/2}(D_+U_j)(D_+V_j)\\ &+\frac{u_N}{2}\Big(m_{N+1/2}\frac{v_{N+1}-v_N}{h}+m_{N-1/2}\frac{v_N-v_{N-1}}{h}\Big)\\ &-\frac{U_1}{2}\Big(M_{3/2}\frac{V_2-V_1}{2h}+M_{1/2}\frac{V_1-V_0}{2h}\Big),\end{aligned}\tag{2.10}$$

where

$$m_{j+1/2}=E_{1/2}\mu(x_j),\qquad M_{j+1/2}=E_{1/2}\mu(X_j).$$

Hence, the symmetry property

$$\begin{aligned}&\big(u,D_-(mD_+v)\big)_{1,h,-\infty,N}+\big(U,D_-(MD_+V)\big)_{1,2h,1,\infty}\\ =&\big(v,D_-(mD_+u)\big)_{1,h,-\infty,N}+\big(V,D_-(MD_+U)\big)_{1,2h,1,\infty},\end{aligned}\tag{2.11}$$

is satisfied if both composite grid functions $\bar{u}=(u,U)^T$ and $\bar{v}=(v,V)^T$ satisfy the interface conditions

$$U_1=u_N,\tag{2.12}$$

$$M_{3/2}\frac{U_2-U_1}{2h}+M_{1/2}\frac{U_1-U_0}{2h}=m_{N-1/2}\frac{u_N-u_{N-1}}{h}+m_{N+1/2}\frac{u_{N+1}-u_N}{h}.\tag{2.13}$$

Note that (2.12) and (2.13) are second order accurate approximations of the continuous jump conditions (2.5), also when $\mu(x)$ is discontinuous at $x=0$.

If we define the energy on the composite grid (cg) by

$$e_{1,cg}^{n+1} = \left\| \frac{u^{n+1}-u^n}{\delta_t} \right\|_{1,h,-\infty,N}^2 - \left(u^{n+1}, D_-(mD_+u^n)\right)_{1,h,-\infty,N} + \left\| \frac{U^{n+1}-U^n}{\delta_t} \right\|_{1,2h,1,\infty}^2 - \left(U^{n+1}, D_-(MD_+U^n)\right)_{1,2h,1,\infty}, \tag{2.14}$$

we can use (2.7) to show

$$\left\| \frac{u^{n+1}-u^n}{\delta_t} \right\|_{1,h,-\infty,N}^2 = \left\| \frac{u^n-u^{n-1}}{\delta_t} \right\|_{1,h,-\infty,N}^2 + \left(u^{n+1}-u^{n-1}, D_-(mD_+u^n)\right)_{1,h,-\infty,N}.$$

The corresponding relation for $(U^{n+1}-U^n)/\delta_t$ follows from (2.8), and we have

$$e_{1,cg}^{n+1} = \left\| \frac{u^n-u^{n-1}}{\delta_t} \right\|_{1,h,-\infty,N}^2 - \left(u^{n-1}, D_-(mD_+u^n)\right)_{1,h,-\infty,N} + \left\| \frac{U^n-U^{n-1}}{\delta_t} \right\|_{1,2h,1,\infty}^2 - \left(U^{n-1}, D_-(MD_+U^n)\right)_{1,2h,1,\infty}.$$

When both (u^n, U^n) and (u^{n-1}, U^{n-1}) satisfy the jump conditions (2.12)-(2.13), symmetry property (2.11) implies that the energy is conserved by the time-stepping, i.e.,

$$e_{1,cg}^{n+1} = e_{1,cg}^n.$$

Furthermore, using Lemmas 2 and 3 in [28] shows that $e_{1,cg}^n \geq 0$ under a CFL-condition on the time step. Hence the discretization is stable.

By using (2.10), the discrete energy (2.14) can be written

$$e_{1,cg}^{n+1} = \left\| \frac{u^{n+1}-u^n}{\delta_t} \right\|_{1,h,-\infty,N}^2 + h\sum_{j=-\infty}^{N-1} m_{j+1/2}\left(D_+(h)u_j^n\right)\left(D_+(h)u_j^{n+1}\right) + \left\| \frac{U^{n+1}-U^n}{\delta_t} \right\|_{1,2h,1,\infty}^2 + 2h\sum_{j=1}^{\infty} M_{j+1/2}\left(D_+(2h)U_j^n\right)\left(D_+(2h)U_j^{n+1}\right),$$

which shows that $e_{1,cg}^{n+1}$ is a second order accurate approximation of the continuous energy (2.4), evaluated at time $t_n+\delta_t/2$.

While (2.13) is a linear relation involving the solution at the ghost points u_{N+1}^n and U_0^n, the condition (2.12) does not explicitly depend on those values. However, u_N^{n+1} and U_1^{n+1} depend on the ghost point values u_{N+1}^n and U_0^n through the difference scheme (2.7)-(2.8) applied on the grid interface. By enforcing (2.12) at time level $n+1$, we therefore get

$$\begin{aligned} &2U_1^n - U_1^{n-1} + \frac{\delta_t^2}{(2h)^2}\left(M_{3/2}(U_2^n-U_1^n) - M_{1/2}(U_1^n-U_0^n)\right) \\ =&2u_N^n - u_N^{n-1} + \frac{\delta_t^2}{h^2}\left(m_{N+1/2}(u_{N+1}^n-u_N^n) - m_{N-1/2}(u_N^n-u_{N-1}^n)\right). \end{aligned} \tag{2.15}$$

Conditions (2.13) and (2.15) lead to a linear system of equations for the two unknowns u^n_{N+1} and U^n_0, with matrix

$$A=\begin{pmatrix} \frac{\delta_t^2}{(2h)^2}M_{1/2} & -\frac{\delta_t^2}{h^2}m_{N+1/2} \\ \frac{1}{2}M_{1/2} & m_{N+1/2} \end{pmatrix}, \qquad \det A = m_{N+1/2}M_{1/2}\frac{3\delta_t^2}{4h^2}.$$

By assumption, $\mu(x)>0$, so the determinant of A is non-zero. Hence, (2.13) and (2.15) are two linearly independent relations which can be solved for the two ghost point values.

3 Symmetric discretization of the elastic wave equation

The 3-D elastic wave equation for the displacement vector $\mathbf{u}(\mathbf{x},t)$, where $\mathbf{x}$ is the spatial location and t is time, is given by

$$\rho\mathbf{u}_{tt}=\operatorname{div}\mathcal{T}(\mathbf{u})+\mathbf{f}(\mathbf{x},t), \qquad \mathbf{x}\in\Omega, \quad t\geq 0. \tag{3.1}$$

Here $\rho=\rho(\mathbf{x})>0$ is the density, $\mathbf{f}(\mathbf{x},t)$ is the external forcing, and $\mathcal{T}(\mathbf{u})$ is the stress tensor,

$$\mathcal{T}(\mathbf{u})=\lambda\operatorname{div}(\mathbf{u})I+2\mu\mathcal{D}(\mathbf{u}).$$

The Lamé parameters $\mu(\mathbf{x})>0$ and $\lambda(\mathbf{x})>0$ characterize the elastic properties of the material. The identity matrix is denoted by I, and $\mathcal{D}$ is the symmetric part of the displacement gradient. In terms of the Cartesian components $\mathbf{u}=(u(\mathbf{x},t),v(\mathbf{x},t),w(\mathbf{x},t))^T$ and $\mathbf{x}=(x,y,z)^T$,

$$\mathcal{D}(\mathbf{u})=\frac{1}{2}\begin{pmatrix} 2u_x & u_y+v_x & u_z+w_x \\ u_y+v_x & 2v_y & v_z+w_y \\ u_z+w_x & v_z+w_y & 2w_z \end{pmatrix}.$$

We consider the finite difference approximation of (3.1)

$$\rho_{i,j,k}\frac{\mathbf{u}^{n+1}_{i,j,k}-2\mathbf{u}^n_{i,j,k}+\mathbf{u}^{n-1}_{i,j,k}}{\delta_t^2}=\mathbf{L}_h(\mathbf{u}^n)_{i,j,k}+\mathbf{f}_{i,j,k}(t_n), \tag{3.2}$$

on a domain $(x,y,z)\in[0,a]\times[0,b]\times[0,c]$. Here, $\mathbf{L}(\mathbf{u})_{i,j,k}$ represents the discretization of the spatial operator in (3.1) which is described below. The external forcing is discretized according to

$$\mathbf{f}_{i,j,k}(t)=\mathbf{f}(\mathbf{x}_{i,j,k},t)=\left(f^{(u)}_{i,j,k}(t),f^{(v)}_{i,j,k}(t),f^{(w)}_{i,j,k}(t)\right)^T.$$

Let the grid function $u^n_{i,j,k}$ denote the approximation of the x-component of the displacement at grid point

$$x_i=(i-1)h, \qquad y_j=(j-1)h, \tag{3.3}$$

and $z_k=(k-1)h$, at time level $t_n=n\delta_t$. In the same way, $v^n_{i,j,k}$ and $w^n_{i,j,k}$ denote the approximations of the y- and z-components of the displacement, respectively. The domain sizes and the grid spacing are defined such that $x_{N_x}=a$, $y_{N_y}=b$ and $z_{N_z}=c$.

We apply the discrete equation (3.2) at all interior grid points $1\le i\le N_x$, $1\le j\le N_y$, and $1\le k\le N_z$. The ghost points with $i=0$, $i=N_x+1$; $j=0$, $j=N_y+1$; $k=0$, or $k=N_z+1$ are used to impose the boundary conditions.

In [28], we generalized the summation-by-parts discretization introduced in [23] to include boundary modified mixed derivatives on all six sides of the domain. In component form, the spatial discretization is given by

$$\mathbf{L}_h(\mathbf{u})=\left(L_h^{(u)}(u,v,w),L_h^{(v)}(u,v,w),L_h^{(w)}(u,v,w)\right)^T,$$

where

$$\begin{aligned}L_h^{(u)}(u,v,w)=&D_-^x\left(E_{1/2}^x(2\mu+\lambda)D_+^x u\right)+D_-^y\left(E_{1/2}^y(\mu)D_+^y u\right)+D_-^z\left(E_{1/2}^z(\mu)D_+^z u\right)\\&+\widetilde{D_0^x}\left(\lambda\widetilde{D_0^y}v+\lambda\widetilde{D_0^z}w\right)+\widetilde{D_0^y}\left(\mu\widetilde{D_0^x}v\right)+\widetilde{D_0^z}\left(\mu\widetilde{D_0^x}w\right),\end{aligned}\tag{3.4}$$

$$\begin{aligned}L_h^{(v)}(u,v,w)=&D_-^x\left(E_{1/2}^x(\mu)D_+^x v\right)+D_-^y\left(E_{1/2}^y(2\mu+\lambda)D_+^y v\right)+D_-^z\left(E_{1/2}^z(\mu)D_+^z v\right)\\&+\widetilde{D_0^x}\left(\mu\widetilde{D_0^y}u\right)+\widetilde{D_0^y}\left(\lambda\widetilde{D_0^x}u+\lambda\widetilde{D_0^z}w\right)+\widetilde{D_0^z}\left(\mu\widetilde{D_0^y}w\right),\end{aligned}\tag{3.5}$$

$$\begin{aligned}L_h^{(w)}(u,v,w)=&D_-^x\left(E_{1/2}^x(\mu)D_+^x w\right)+D_-^y\left(E_{1/2}^y(\mu)D_+^y w\right)+D_-^z\left(E_{1/2}^z(2\mu+\lambda)D_+^z w\right)\\&+\widetilde{D_0^x}\left(\mu\widetilde{D_0^z}u\right)+\widetilde{D_0^y}\left(\mu\widetilde{D_0^z}v\right)+\widetilde{D_0^z}\left(\lambda\widetilde{D_0^x}u+\lambda\widetilde{D_0^y}v\right).\end{aligned}\tag{3.6}$$

Here we use a multi-dimensional notation of the standard divided difference operators, i.e.,

$$D_+^x u_{i,j,k}=\frac{u_{i+1,j,k}-u_{i,j,k}}{h},\quad D_-^x u_{i,j,k}=D_+^x u_{i-1,j,k},\quad D_0^x=\frac{1}{2}(D_+^x+D_-^x).$$

The boundary modified operator for differences in the x-direction is defined by

$$\widetilde{D_0^x}u_{i,j,k}=\begin{cases}D_+^x u_{i,j,k}, & i=1,\\ D_0^x u_{i,j,k}, & 2\le i\le N_x-1,\\ D_-^x u_{i,j,k}, & i=N_x,\end{cases}$$

and the multi-dimensional averaging operator is defined by

$$E_{1/2}^x(\mu)_{i,j,k}=\frac{\mu_{i+1,j,k}+\mu_{i,j,k}}{2}.$$

The superscripts on the difference and averaging operators denote the direction in which the operator is applied and we use corresponding definitions for the difference operators in the y- and z-directions.

To analyze the stability of the discrete equations, we define the 3-D weighted scalar product and norm

$$(u,v)_{3,h,N_x,N_y,N_z} = h^3 \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} a_i^{(x)} a_j^{(y)} a_k^{(z)} u_{i,j,k} v_{i,j,k}, \tag{3.7a}$$

$$\|u\|^2_{3,h,N_x,N_y,N_z} = (u,u)_{3,h,N_x,N_y,N_z}. \tag{3.7b}$$

We will use the more compact syntax $(u,v)_{3,h}$ when the meaning is obvious. The weights in the scalar product satisfy

$$a_i^{(x)} = \begin{cases} 1/2, & i=1 \text{ or } i=N_x, \\ 1, & 1<i<N_x, \end{cases}$$

with corresponding definitions of $a^{(y)}$ and $a^{(z)}$. We define the vector scalar product and norm by

$$(\mathbf{u}^0,\mathbf{u}^1)_{3,h} = (u^0,u^1)_{3,h} + (v^0,v^1)_{3,h} + (w^0,w^1)_{3,h}, \quad \|\mathbf{u}\|^2_{3,h} = (\mathbf{u},\mathbf{u})_{3,h}.$$

In [28], we proved that the energy, defined by

$$e^{n+1}_{3,h} = \left\| \rho^{1/2} \frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\delta_t} \right\|^2_{3,h} - \left(\mathbf{u}^{n+1}, \mathbf{L}_h(\mathbf{u}^n)\right)_{3,h} + T_h(\mathbf{u}^{n+1},\mathbf{u}^n),$$

is positive under a CFL-condition. Furthermore, it satisfies

$$e^{n+1}_{3,h} = e^n_{3,h} + T_h(\mathbf{u}^{n+1}-\mathbf{u}^{n-1},\mathbf{u}^n),$$

where the boundary term is

$$\begin{aligned} T_h(\mathbf{u},\mathbf{v}) = & h^2 \sum_{j=1}^{N_y}\sum_{k=1}^{N_z} a_j^{(y)} a_k^{(z)} \left(-\mathbf{u}_{1,j,k}\cdot \mathbf{B}(\mathbf{v})^{(x)}_{1,j,k} + \mathbf{u}_{N_x,j,k}\cdot \mathbf{B}(\mathbf{v})^{(x)}_{N_x,j,k} \right) \\ & + h^2 \sum_{i=1}^{N_x}\sum_{k=1}^{N_z} a_i^{(x)} a_k^{(z)} \left(-\mathbf{u}_{i,1,k}\cdot \mathbf{B}(\mathbf{v})^{(y)}_{i,1,k} + \mathbf{u}_{i,N_y,k}\cdot \mathbf{B}(\mathbf{v})^{(y)}_{i,N_y,k} \right) \\ & + h^2 \sum_{i=1}^{N_x}\sum_{j=1}^{N_y} a_i^{(x)} a_j^{(y)} \left(-\mathbf{u}_{i,j,1}\cdot \mathbf{B}(\mathbf{v})^{(z)}_{i,j,1} + \mathbf{u}_{i,j,N_z}\cdot \mathbf{B}(\mathbf{v})^{(z)}_{i,j,N_z} \right). \end{aligned} \tag{3.8}$$

Here, $\mathbf{B}^{(x)}$, $\mathbf{B}^{(y)}$, and $\mathbf{B}^{(z)}$ denote the discretized boundary stresses normal to the x, y, and z-directions, respectively. The normal stresses are discretized as described in [23] and [28]. For example, the discretized boundary stress normal to the z-direction is

$$\mathbf{B}(\mathbf{u})^{(z)}_{i,j,k} = \begin{pmatrix} \frac{1}{2}\mu_{i,j,k-1/2} D^z_- u_{i,j,k} + \frac{1}{2}\mu_{i,j,k+1/2} D^z_+ u_{i,j,k} + \mu_{i,j,k} \widetilde{D^x_0} w_{i,j,k} \\ \frac{1}{2}\mu_{i,j,k-1/2} D^z_- v_{i,j,k} + \frac{1}{2}\mu_{i,j,k+1/2} D^z_+ v_{i,j,k} + \mu_{i,j,k} \widetilde{D^y_0} w_{i,j,k} \\ \frac{1}{2}(2\mu+\lambda)_{i,j,k-1/2} D^z_- w_{i,j,k} + \frac{1}{2}(2\mu+\lambda)_{i,j,k+1/2} D^z_+ w_{i,j,k} \\ \qquad + \lambda_{i,j,k}(\widetilde{D^x_0} u_{i,j,k} + \widetilde{D^y_0} v_{i,j,k}) \end{pmatrix}. \tag{3.9}$$

Note that homogeneous Dirichlet ($\mathbf{u}=\mathbf{0}$) or free surface boundary conditions ($\mathbf{B}=\mathbf{0}$), make the boundary term $T_h(\mathbf{u},\mathbf{v})$ vanish, which results in a stable, energy conserving, numerical scheme.

4 Energy conserving interpolation

Consider a computational domain $0\le x\le a$, $0\le y\le b$, and $0\le z\le c$, which is divided into two sub-domains in the z-direction, such that the upper subdomain $0\le z\le c_0$ is covered by a grid of spacing h, and the lower subdomain $c_0\le z\le c$ is covered by a grid of spacing $2h$, see Fig. 1(b). As was discussed in the introduction, this grid configuration is desirable in seismology where the material properties vary along the vertical (z-) direction in such a way that a fine grid is needed to resolve waves near the surface ($z=0$), while a coarse grid gives adequate resolution deeper into the ground, where the waves are longer.

We align the grid points in a node centered fashion on the grid refinement interface and add one ghost point on each side of the grid refinement boundary $z=c_0$. In the z-direction, grid points are denoted by z_k and Z_k on the fine and coarse grids, respectively. They are located at

$$\begin{aligned} z_{k_1} &= c_0+(k_1-n_z)h, \quad 0\le k_1\le n_z+1,\\ Z_{k_2} &= c_0+(k_2-1)2h, \quad 0\le k_2\le N_z+1. \end{aligned}$$

Here, N_z, n_z, and h are chosen such that $z_1=0$ and $Z_{N_z}=c$. By construction, $z_{n_z}=Z_1$, so the grid arrangement in the z-direction is similar to the one-dimensional case, see Fig. 3.

In the x- and y-directions, the grid points are given by (3.3) with grid spacings h and $2h$ on the fine and coarse grids, respectively, see Fig. 4. Excluding the ghost points, the fine grid has n_x and n_y grid points in the x- and y-directions, respectively. The coarse grid has N_x and N_y grid points in the corresponding directions. Since the refinement ratio is two, we have

$$n_x=2N_x-1, \qquad n_y=2N_y-1.$$

Furthermore, grid point $(i,j,1)$ on the coarse grid is co-located with grid point $(2i-1,2j-1,n_z)$ on the fine grid. The grid points on the fine grid with indices $(2i,2j-1,n_z)$, $(2i-1,2j,n_z)$, and $(2i,2j,n_z)$ are located in between coarse grid points and are called hanging nodes.

We denote the solution on the coarse grid by $\mathbf{U}_{i,j,k}$ and the solution on the fine grid by $\mathbf{u}_{i,j,k}$. Similar to the 1-D wave equation in Section 2, the discrete energy is defined as the sum of the energies from the two subdomains

$$\begin{aligned} e_{3,cg}^{n+1} = &\left\| \sqrt{\rho}\,\frac{\mathbf{u}^{n+1}-\mathbf{u}^n}{\delta_t} \right\|_{3,h}^2 - \left(\mathbf{u}^{n+1},\mathbf{L}_h(\mathbf{u}^n)\right)_{3,h} + T_h(\mathbf{u}^{n+1},\mathbf{u}^n)\\ &+ \left\| \sqrt{\rho}\,\frac{\mathbf{U}^{n+1}-\mathbf{U}^n}{\delta_t} \right\|_{3,2h}^2 - \left(\mathbf{U}^{n+1},\mathbf{L}_{2h}(\mathbf{U}^n)\right)_{3,2h} + T_{2h}(\mathbf{U}^{n+1},\mathbf{U}^n), \end{aligned} \tag{4.1}$$

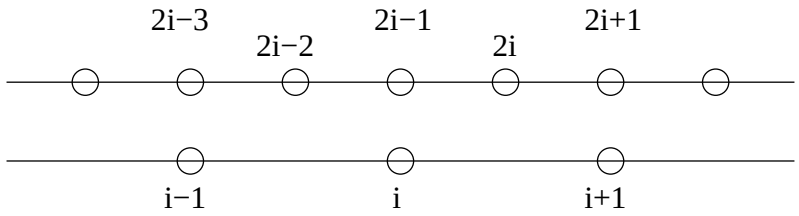


Figure 4: The alignment of grid points along a horizontal grid refinement boundary.

where subscripts $2h$ and h denote quantities evaluated on the coarse and fine grids, respectively.

We enforce a free surface boundary condition at $z=0$ and homogeneous Dirichlet conditions on all other sides, which for example are natural to use in conjunction with a damping sponge layer near far-field boundaries. Because of the outer boundary conditions, the only non-zero contributions to the boundary terms T_{2h} and T_h arise from the interface between the two grids, i.e., the terms involving $\mathbf{B}(\mathbf{v})^{(z)}$ in (3.8). To simplify the notation, we set

$$\mathbf{B}_{i,j}=\mathbf{B}(\mathbf{U}^{(n)})^{(z)}_{i,j,1}, \qquad \mathbf{b}_{i,j}=\mathbf{B}(\mathbf{u}^{(n)})^{(z)}_{i,j,n_z},$$

in the remainder of this section. To avoid confusion between the scaling factors in the scalar products, we denote the weights in the fine grid scalar product (3.7) by

$$\alpha_i^{(x)}, \quad \alpha_j^{(y)}, \quad \alpha_k^{(z)}.$$

By re-organizing the terms in (3.8) we arrive at the following result.

Lemma 4.1. *Assume that homogeneous Dirichlet or free surface boundary conditions are imposed on the outer boundaries of the computational domain. The discrete energy is conserved* ($e^{n+1}_{3,cg}=e^n_{3,cg}$) *if the following grid interface condition is satisfied:*

$$\begin{aligned}
&\sum_{i=1}^{N_x}\sum_{j=1}^{N_y} a_i^{(x)}a_j^{(y)}\mathbf{U}^{n+1}_{i,j,1}\cdot\mathbf{B}_{i,j}\\
=&\frac{1}{4}\sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\alpha^{(x)}_{2i-1}\alpha^{(y)}_{2j-1}\mathbf{u}^{n+1}_{2i-1,2j-1,n_z}\cdot\mathbf{b}_{2i-1,2j-1}+\frac{1}{4}\sum_{i=1}^{N_x-1}\sum_{j=1}^{N_y}\alpha^{(x)}_{2i}\alpha^{(y)}_{2j-1}\mathbf{u}^{n+1}_{2i,2j-1,n_z}\cdot\mathbf{b}_{2i,2j-1}\\
&+\frac{1}{4}\sum_{i=1}^{N_x}\sum_{j=1}^{N_y-1}\alpha^{(x)}_{2i-1}\alpha^{(y)}_{2j}\mathbf{u}^{n+1}_{2i-1,2j,n_z}\cdot\mathbf{b}_{2i-1,2j}+\frac{1}{4}\sum_{i=1}^{N_x-1}\sum_{j=1}^{N_y-1}\alpha^{(x)}_{2i}\alpha^{(y)}_{2j}\mathbf{u}^{n+1}_{2i,2j,n_z}\cdot\mathbf{b}_{2i,2j}. \qquad (4.2)
\end{aligned}$$

When homogeneous Dirichlet boundary conditions are enforced on the outer boundaries, all edge and corner terms in (4.2) vanish, i.e.,

$$\begin{aligned}
&\mathbf{U}^{n+1}_{1,j,1}=0, \quad \mathbf{U}^{n+1}_{i,1,1}=0, \quad \mathbf{U}^{n+1}_{N_x,j,1}=0, \quad \mathbf{U}^{n+1}_{i,N_y,1}=0,\\
&\mathbf{u}^{n+1}_{1,j,n_z}=0, \quad \mathbf{u}^{n+1}_{i,1,n_z}=0, \quad \mathbf{u}^{n+1}_{n_x,j,n_z}=0, \quad \mathbf{u}^{n+1}_{i,n_y,n_z}=0.
\end{aligned}$$

Therefore, all remaining terms in (4.2) have unit weights in the scalar products.

In order to satisfy (4.2), it is natural to enforce

$$\mathbf{U}^n_{i,j,1} = \mathbf{u}^n_{2i-1,2j-1,n_z}, \qquad 2 \le i \le N_x - 1,\ 2 \le j \le N_y - 1, \tag{4.3}$$

together with second order accurate averaging conditions for the solution at the hanging nodes

$$\mathbf{u}^n_{2i-1,2j,n_z} = \frac{1}{2}(\mathbf{u}^n_{2i-1,2j-1,n_z} + \mathbf{u}^n_{2i-1,2j+1,n_z}), \quad 2 \le i \le N_x - 1,\ 1 \le j \le N_y - 1, \tag{4.4}$$

$$\mathbf{u}^n_{2i,2j-1,n_z} = \frac{1}{2}(\mathbf{u}^n_{2i-1,2j-1,n_z} + \mathbf{u}^n_{2i+1,2j-1,n_z}), \quad 1 \le i \le N_x - 1,\ 2 \le j \le N_y - 1, \tag{4.5}$$

$$\mathbf{u}^n_{2i,2j,n_z} = \frac{1}{4}(\mathbf{u}^n_{2i-1,2j-1,n_z} + \mathbf{u}^n_{2i+1,2j-1,n_z} + \mathbf{u}^n_{2i-1,2j+1,n_z} + \mathbf{u}^n_{2i+1,2j+1,n_z}), \tag{4.6}$$

where (4.6) is enforced for $1 \le i \le N_x - 1$, $1 \le j \le N_y - 1$. Finally, expression (4.2) determines the interpolation formula that $\mathbf{B}$ must satisfy:

$$\begin{aligned}\mathbf{B}_{i,j} = {} & \frac{1}{4}\mathbf{b}_{2i-1,2j-1} + \frac{1}{8}\left(\mathbf{b}_{2i-1,2j-1} + \mathbf{b}_{2i+1,2j-1} + \mathbf{b}_{2i-1,2j+1} + \mathbf{b}_{2i+1,2j+1}\right) \\ & + \frac{1}{16}\left(\mathbf{b}_{2i-2,2j-2} + \mathbf{b}_{2i-2,2j} + \mathbf{b}_{2i,2j-2} + \mathbf{b}_{2i,2j}\right),\end{aligned} \tag{4.7}$$

where $2 \le i \le N_x - 1$, $2 \le j \le N_y - 1$.

Note that (4.2) is symmetric in $(\mathbf{u},\mathbf{U})$ and $(\mathbf{b},\mathbf{B})$, which implies that an alternative, dual, energy conserving interface condition is obtained by swapping $(\mathbf{u},\mathbf{U})$ and $(\mathbf{b},\mathbf{B})$ in formulas (4.3)-(4.7). The conditions (4.3)-(4.7) are natural to use when Dirichlet conditions are imposed at $x = 0$, $x = a$, $y = 0$, and $y = b$, because (4.7) do not depend on any i- or j-boundary points, while some of (4.3)-(4.6) do. If free surface conditions were used on these boundaries, the dual of (4.3)-(4.7) would be a more natural interface condition.

There are $(N_x - 2)(N_y - 2)$ unknown ghost points in the coarse grid, and $(2N_x - 3)(2N_y - 3)$ unknown points in the fine grid. The total number of unknown points is therefore

$$5N_xN_y - 8(N_x + N_y) + 13.$$

The number of equations is (note the index bounds in (4.3)-(4.7))

$$\begin{aligned}& 2(N_x - 2)(N_y - 2) + (N_x - 2)(N_y - 1) + (N_x - 1)(N_y - 2) + (N_x - 1)(N_y - 1) \\ = {} & 5N_xN_y - 8(N_x + N_y) + 13.\end{aligned} \tag{4.8}$$

Thus the number of equations equal the number of unknowns.

Similar to (2.12) in the 1-D example problem, (4.3)-(4.6) do not depend explicitly on the solution values at the ghost points. To obtain an equation for the ghost point values at time level n, we impose (4.3)-(4.6) on time level $n+1$ and substitute the finite difference scheme (3.2) in the same way as done for the 1-D example problem in (2.15).

One difficulty with the above approach is that the hanging node treatment leads to a coupling of the conditions along the interface. Hence, the solution values at the ghost points must be obtained by solving a linear system of equations involving all ghost points on the grid interface. The linear system has of the order $N_x N_y$ unknowns and has a band structure with band width of the order $\min(N_x, N_y)$. Since the coefficients in the linear system are constant in time, we could in principle LU-decompose the system once and then back-substitute to obtain ghost point values at each time step. However, LU-decomposition is not straight forward to perform over many processors on a parallel machine, and it is therefore desirable to solve the interface conditions iteratively.

After some experimentation, we found an iterative block Jacobi relaxation method that works very well in practice. Each equation in (4.3)-(4.7) is a vector equation for the ghost point values of $\mathbf{u}$ on the coarse and fine grids. The equations decouple into three separate linear systems for the three components $(u\ v\ w)$, which can be solved independently of each other. We now outline the block Jacobi method for the u-component. Away from the boundaries, each blocks consist of the five unknowns

$$\mathbf{w}_{i,j} = (U^n_{i,j,0},\ u^n_{2i-1,2j-1,n_z+1},\ u^n_{2i,2j-1,n_z+1},\ u^n_{2i-1,2j,n_z+1},\ u^n_{2i,2j,n_z+1})^T. \tag{4.9}$$

In each block we solve the u-component of the linear equations (4.3)-(4.7) evaluated at the same index (i,j). We write each block system as

$$A_{i,j}\mathbf{w}_{i,j} = \mathbf{d}_{i,j},$$

where $A_{i,j}$ is a 5×5 matrix. The right hand side $\mathbf{d}_{i,j}$ holds contributions from interior grid points and forcing functions (which are constant during the Jacobi iteration) as well as contributions from the solution values in the neighboring ghost points

$$u^n_{2i-1,2j+1,n_z+1},\ u^n_{2i+1,2j-1,n_z+1},\ u^n_{2i+1,2j+1,n_z+1},\ u^n_{2i-2,2j-2,n_z+1},\ u^n_{2i-2,2j,n_z+1},\ u^n_{2i,2j-2,n_z+1},$$

which are outside the block. The contribution from these ghost points follows from the approximate values from the previous Jacobi iteration. The number of equations and unknowns in the blocks are reduced near boundaries and corners since some equations are not defined on the boundary and some unknowns are already determined by the outer boundary conditions.

The block-Jacobi approach has turned out to be very efficient in practical computations where the method converges to acceptable precision in less than five iterations. As with all Jacobi iterations, the convergence properties do not depend on the decomposition over a parallel machine. However, our choice of unknowns in each block is slightly asymmetric with respect to the center point $(2i-1,2j-1)$ in the fine grid, leading to an asymmetric communication requirement in our parallel implementation.

In the case of more than two refinements, for example with spacing h in $0\le z\le c_0$, spacing $2h$ in $c_0\le z\le c_1$, and spacing $4h$ in $c_1\le z\le c$, the interpolation technique applies separately at each of the interfaces.

5 Source term discretization near grid interfaces

In seismic applications of the 3-D elastic wave equation, we need to discretize point moment tensor sources

$$g(t)\mathcal{M}\nabla\delta(\mathbf{x}-\mathbf{x}_*), \tag{5.1}$$

where $\mathcal{M}$ is a symmetric 3×3 matrix, and point forces

$$g(t)\mathcal{F}\delta(\mathbf{x}-\mathbf{x}_*), \tag{5.2}$$

where $\mathcal{F}$ is a vector with 3 components. In these expressions, $\delta(\mathbf{x}-\mathbf{x}_*)$ is the Dirac distribution, $\nabla\delta$ is its gradient, and the time function $g(t)$ is assumed to be sufficiently smooth in time.

The n-dimensional Dirac distribution has the property

$$\int \phi(\mathbf{x})\delta(\mathbf{x}-\mathbf{x}_*)\,\mathrm{d}\mathbf{x}=\phi(\mathbf{x}_*),$$

for any smooth function $\phi(\mathbf{x})$. Similarly, if x is a Cartesian component of $\mathbf{x}$, the s'th derivative with respect to the x-direction satisfies

$$\int \phi(\mathbf{x})\frac{\partial^s\delta}{\partial x^s}(\mathbf{x}-\mathbf{x}_*)\,\mathrm{d}\mathbf{x}=(-1)^s\frac{\partial^s\phi}{\partial x^s}(\mathbf{x}_*).$$

Our discretization principle for the Dirac distribution and its derivatives is to satisfy the integral condition for polynomials up to a given degree, where the integrals are replaced by a discrete scalar product.

5.1 One-dimensional discretization of source terms

On a uniform grid with grid size h, we approximate $\delta(x-x_*)$ by a grid function d_h, by requiring that all polynomial functions

$$P_q(x)=\sum_{\nu=0}^{q}\pi_\nu x^\nu,$$

satisfy

$$(P_q,d_h)_{1,h}=P_q(x_*),\qquad q=0,\cdots,Q. \tag{5.3}$$

Similarly, a grid function d_h' that approximates $\delta'(x-x_*)$ should satisfy

$$(P_q,d_h')_{1,h}=-P_q'(x_*),\qquad q=0,\cdots,Q. \tag{5.4}$$

Because (5.3) or (5.4) give $Q+1$ conditions, d_h or d_h' need to be non-zero at $Q+1$ grid points, i.e., their stencils are $Q+1$ points wide.

For the Helmholtz equation discretized on a uniform grid, Waldén [33] showed that if the difference approximation of the partial differential equation is p'th order accurate, $\partial^s\delta/\partial x^s(x-x_*)$ must be discretized with

$$Q\geq p-1+s,$$

to obtain overall p'th order convergence of the solution (away from the solution singularity at x_*). Our experience is that this requirement is sharp also for the elastic wave equation discretized on a uniform grid, at least for $s=0$ and $s=1$.

We begin by deriving a third order discretization of $\delta'(x-x_*)$ on a uniform grid using the scalar product $(u,v)_{1,h,-\infty,\infty}$. Let j be the grid point index such that $x_j\leq x_*<x_{j+1}$. Because $p=3$ and $s=1$, we take $Q=3$, i.e., there are four conditions in (5.4). Hence, we use a four point stencil

$$d_h'=(\cdots,0,\delta_{j-1}',\delta_j',\delta_{j+1}',\delta_{j+2}',0,\cdots)^T.$$

To make the coefficients in the linear system $\mathcal{O}(1)$, we enforce (5.4) for the polynomial functions

$$P_q(x)=(x-x_j)^q,\quad 0\leq q\leq 3,$$

leading to the conditions

$$(x_{j-1}-x_j)^q\delta_{j-1}'+(x_j-x_j)^q\delta_j'+(x_{j+1}-x_j)^q\delta_{j+1}'+(x_{j+2}-x_j)^2\delta_{j+2}'=-\frac{q}{h}(x_*-x_j)^{q-1},\quad (5.5)$$

where $q=0,1,2,3$.

Note that, by definition,

$$(x_j-x_j)^q=1,\quad \text{for } q=0;\qquad (x_j-x_j)^q=0,\quad \text{for } q\geq 1.$$

Introducing the relative source location

$$\alpha=(x_*-x_j)/h,$$

gives the system

$$\begin{pmatrix}1&1&1&1\\-1&0&1&2\\1&0&1&4\\-1&0&1&8\end{pmatrix}\begin{pmatrix}\delta_{j-1}'\\\delta_j'\\\delta_{j+1}'\\\delta_{j+2}'\end{pmatrix}=\begin{pmatrix}0\\-1/h^2\\-2\alpha/h^2\\-3\alpha^2/h^2\end{pmatrix},$$

which has the solution

$$\delta_{j-1}'=\frac{1}{h^2}\Big(\frac{1}{3}-\alpha+\frac{\alpha^2}{2}\Big),\qquad \delta_j'=\frac{1}{h^2}\Big(\frac{1}{2}+2\alpha-\frac{3\alpha^2}{2}\Big),\quad (5.6a)$$

$$\delta_{j+1}'=\frac{1}{h^2}\Big(-1-\alpha+\frac{3\alpha^2}{2}\Big),\qquad \delta_{j+2}'=\frac{1}{h^2}\Big(\frac{1}{6}-\frac{\alpha^2}{2}\Big).\quad (5.6b)$$

We remark that the source discretization formulas are valid regardless of the center point x_j. Thus it is easy to bias the discretization stencil away from the outer boundary of the domain. For example, if x_{j+2} is outside the domain, the stencil can be centered around x_{j-1}. The resulting formulas (5.6a) and (5.6b) will be the same, but with j replaced by $j-1$ and with $\alpha=(x_*-x_{j-1})/h$.

We proceed by generalizing Waldén's source discretization formulas to the situation (which surprisingly often occurs in practice) where the grid spacing is larger than the distance between the source and the grid refinement interface. Obviously, one can avoid the problem altogether by moving the source by $\mathcal{O}(h)$ away from the grid interface, but that approach inevitably results in an $\mathcal{O}(h)$ solution error. Here we aim for a procedure which gives overall second order convergence. From a theoretical standpoint, as long as the source is not located exactly on the grid interface, the grid size can always be made fine enough to put a fixed number of grid points between the source and the interface. Waldén's formulas would therefore give second order convergence rate in the limit when the grid spacing goes to zero, but the required grid size might be much smaller than what is practically possible.

When the source is near a grid refinement interface, we enforce the moment conditions (5.3) or (5.4) in terms of the 1-D composite scalar product

$$(\bar{u},\bar{v})_{1,cg,p_1,q_1,p_2,q_2}=(u,v)_{1,h,p_1,q_1}+(U,V)_{1,2h,p_2,q_2}. \tag{5.7}$$

As before, we will abbreviate the notation to $(\bar{u},\bar{v})_{1,cg}$ when the meaning is obvious. Consider the situation displayed in Fig. 3 and let us derive a third order discretization of $\delta(x-x_*)$, when the grid point closest to x_* is the interface point $x_N=X_1$. $Q=2$ implies a three point stencil. However, the interface point is represented on both grids, so we use the points with indices $N-1$, N on the fine grid and indices 1, 2 on the coarse grid. (Note that the source term is only defined on interior points so the ghost points x_{N+1} and X_0 are not needed to discretize the source.) Similar to the composite grid notation (2.6) for the solution, we denote the weights on the composite grid by

$$\bar{d}=(d_h,d_{2h}),\quad d_h=(\cdots,0,\delta_{N-1},\delta_N)^T,\quad d_{2h}=(\Delta_1,\Delta_2,0,\cdots)^T.$$

The order conditions (5.3) together with the scalar product (5.7), applied to the polynomial functions

$$P_q(x)=(x-x_N)^q,\quad q=0,1,2,$$

give

$$(x_{N-1}-x_N)^q\delta_{N-1}+(x_N-x_N)^q\Big(\frac{\delta_N}{2}+\Delta_1\Big)+2(X_2-X_1)^q\Delta_2=\frac{(x_*-x_N)^q}{h}, \tag{5.8}$$

where $q=0,1,2$.

It is natural to introduce the combined interface source weight $\mathcal{I}=\delta_N/2+\Delta_1$. In terms of $\alpha=(x_*-x_N)/h$, the solution of (5.8) is

$$\delta_{N-1}=\frac{1}{3h}(-2\alpha+\alpha^2),\quad \mathcal{I}=\frac{1}{h}\Big(1+\frac{\alpha}{2}-\frac{\alpha^2}{2}\Big),\quad \Delta_2=\frac{1}{12h}(\alpha+\alpha^2). \tag{5.9}$$

Note that the moment conditions only prescribe $\mathcal{I}$, and not the individual weights δ_N and Δ_1. From numerical experiments, we have found that is very important how the combined interface weight is distributed between the grids, and that the best choice is given by

$$\delta_N = \mathcal{I}, \qquad \Delta_1 = \mathcal{I}/2. \tag{5.10}$$

We are currently unaware of a theoretical justification for this formula.

5.2 Numerical experiments in one space dimension

As a model for the elastic wave equation, we begin by studying the one-dimensional wave equation where the source term is the s'th derivative of a Dirac distribution

$$u_{tt} = \big(\mu(x)u_x\big)_x + g(t)\frac{\partial^s \delta}{\partial x^s}(x-x_*), \quad -\infty < x < \infty, \ t \geq 0, \tag{5.11}$$

subject to homogeneous initial conditions

$$u(x,0) = 0, \quad u_t(x,0) = 0, \quad -\infty < x < \infty.$$

We start by constructing an analytical solution. Let $\hat{u}(\xi,t)$ denote the spatial Fourier transform of $u(x,t)$. When

$$\mu = c^2 = \text{const},$$

the Fourier transform of (5.11) is an ordinary differential equation in time, whose solution is

$$\hat{u}(\xi,t) = \frac{1}{2i\xi c}\Big(F_1(\xi,t)e^{i\xi ct} - F_2(\xi,t)e^{-i\xi ct}\Big), \quad -\infty < \xi < \infty, \ t \geq 0. \tag{5.12}$$

The dependence on s is reflected in the functions F_1 and F_2, according to

$$F_1(\xi,t) = (i\xi)^s e^{-i\xi x_*}\int_0^t e^{-i\xi c\tau} g(\tau)\,\mathrm{d}\tau, \tag{5.13}$$

and $F_2(\xi,t)$ follows by replacing c by $-c$ in (5.13).

Let the time function be a general polynomial of order p

$$g(t) = \sum_{\nu=0}^{p} \gamma_\nu t^\nu.$$

Repeated integration by parts gives

$$\int_0^t e^{-i\xi c\tau} g(\tau)\,\mathrm{d}\tau = \sum_{\nu=0}^{p}\gamma_\nu\Big(\frac{b_\nu}{(\xi c)^{\nu+1}} + e^{-i\xi ct}\sum_{m=0}^{\nu+1}\frac{a_{m,\nu}}{(\xi c)^m}\Big), \tag{5.14}$$

for some constants $a_{m,\nu}$ and b_ν. Insertion of (5.14) into (5.13) and (5.12) gives

$$\hat{u}=\frac{(i\xi)^s}{2i\xi c}\sum_{\nu=0}^{p}\gamma_\nu\left(e^{i\xi(ct-x_*)}\frac{b_\nu}{(\xi c)^{\nu+1}}-e^{-i\xi(ct+x_*)}\frac{b_\nu}{(-\xi c)^{\nu+1}}\right.$$
$$\left.+e^{-i\xi x_*}\sum_{m=0}^{\nu+1}\left(\frac{a_{m,\nu}}{(\xi c)^m}-\frac{a_{m,\nu}}{(-\xi c)^m}\right)\right). \tag{5.15}$$

The solution of (5.11) follows from the inverse Fourier transform formula

$$u_e(x,t)=\frac{1}{2\pi}\int_{-\infty}^{\infty}e^{i\xi x}\hat{u}(\xi,t)\,\mathrm{d}\xi. \tag{5.16}$$

Each term in the sum (5.15) leads to an integral over a function of the form

$$e^{i\xi(x-x_*)}/\xi^m,\quad e^{i\xi(x-x_*+ct)}/\xi^m,\quad \text{or}\quad e^{i\xi(x-x_*-ct)}/\xi^m,$$

where m is an integer. The formula

$$\int_{-\infty}^{\infty}\frac{e^{i\xi\alpha}}{\xi^n}\mathrm{d}\xi=\mathrm{sgn}(\alpha)\pi i^n\frac{\alpha^{n-1}}{(n-1)!}, \tag{5.17}$$

can be used to write down an explicit expression for the analytical solution, which we omit in the interest of space.

Numerical experiments are made on a composite grid configuration, covering the domain $0\le x\le 1$, with grid size h on $0\le x\le 1/2$ and grid size $2h$ on $1/2\le x\le 1$. The grid points are laid out such that the grid interface $x=1/2$ always coincides with a grid point in both grids, similar to Fig. 3. The wave equation (5.11) with $\mu=1$ is discretized by the second order approximations (2.7) and (2.8). We compute the solution errors by comparison with the analytical solution (5.16) for the case $s=1$, i.e., when the source is a first derivative of a Dirac distribution. Homogeneous Dirichlet boundary conditions are imposed on the numerical solution at $x=0$ and $x=1$. Hence, the solution error can only be evaluated in the time interval $0\le t\le t_*$, during which the analytical solution (defined on $-\infty<x<\infty$) satisfies the homogeneous Dirichlet conditions $u_e(0,t)=u_e(1,t)=0$.

We consider the polynomial time function

$$g(t)=\begin{cases}-20t^7+70t^6-84t^5+35t^4, & 0\le t<1,\\ 1, & t>1,\end{cases}$$

which represents a smooth transition from 0 to 1 with three vanishing derivatives at $t=0$ and $t=1$, see Fig. 5. Due to the unit wave speed and the hyperbolic nature of (5.11), we only need to evaluate the analytical solution on $0\le x\le 1$ for $t<1$. Hence, the case $g(t)=1$ for $t>1$ does not need to be considered when evaluating the inverse Fourier transform formula.

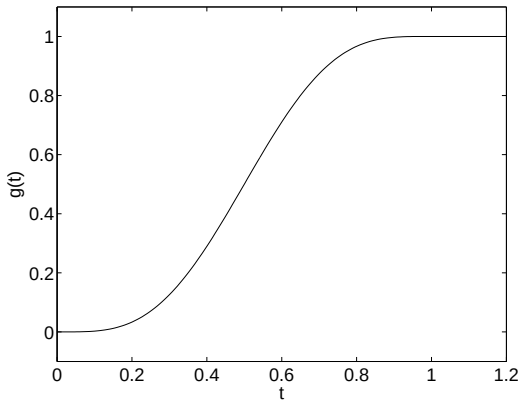


Figure 5: Source time function $g(t)$.

In the following numerical experiments, we have used (5.10) to distribute the combined source weight at the grid interface. Furthermore, the moment conditions are satisfied to one order higher than the minimal requirement for second order accuracy on a uniform grid, i.e., we use $Q=2$ in (5.3) and $Q=3$ in (5.4).

We start by comparing the energy conserving interpolation (2.12), (2.13) to a straight forward interpolation formula

$$U_0 = u_{N-2}, \tag{5.18a}$$

$$u_{N+1} = -\frac{1}{8}U_0 + \frac{3}{4}U_1 + \frac{3}{8}U_2 = -\frac{1}{8}u_{N-2} + \frac{3}{4}U_1 + \frac{3}{8}U_2, \tag{5.18b}$$

which is third order accurate for smooth solutions. The results with non-conservative interpolation are shown in Fig. 6. On the left we show the solution when the source is located away from the grid interface, at $x_* = 0.33$. Here the solution is smooth over the grid interface and the interpolation formula (5.18) works well. The situation is very different on the right side of Fig. 6, in which case the source is located at $x_* = 0.5077$, i.e., very close to the grid interface. This case uses the composite grid discretization of the source, because x_* is less than one coarse grid spacing ($2h = 0.0125$) away from the grid refinement interface. Since the solution is discontinuous at x_*, the interpolation formula (5.18) leads to large errors in the ghost points which pollute the numerical solution away from the discontinuity.

The distance between the source and the grid interface has a much smaller influence on the solution error when the energy conserving interpolation (2.12), (2.13) is used. The left side of Fig. 7 shows the solution when the source is placed away from the grid refinement interface, at $x_* = 0.33$. The right side of the same figure displays the result when the source is located very close to the grid refinement boundary, at $x_* = 0.5077$. For both source locations, the numerical solution is accurate in all but one or two points, right next to the source.

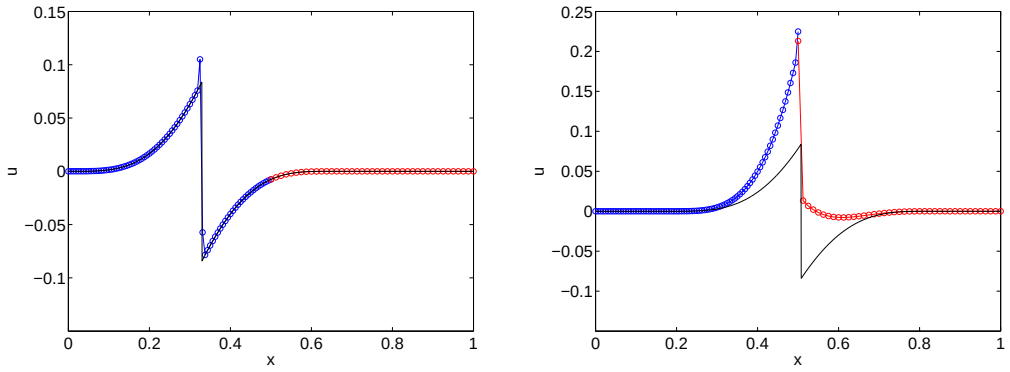


Figure 6: Solution on the composite grid at time $t=0.33$ when the source is located at $x_*=0.33$ (left) and at $x_*=0.5077$ (right) with non-conservative interpolation. Here blue and red color mark the numerical solution on the fine and coarse grids, respectively. The black curve is the exact solution. In both figures, the grid size on the fine grid is $h=0.00625$.

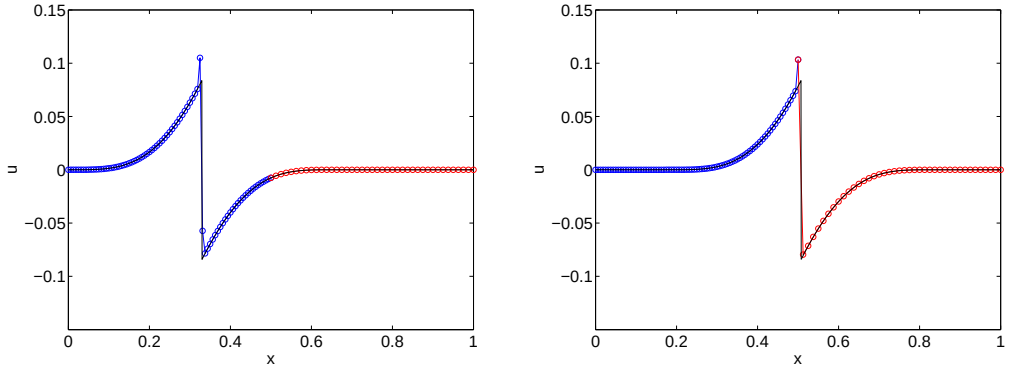


Figure 7: Solution on the composite grid at time $t=0.33$ when the source is located at $x_*=0.33$ (left) and at $x_*=0.5077$ (right) with energy conserving interpolation. Here blue and red color mark the numerical solution on the fine and coarse grids, respectively. The black curve is the exact solution. In both figures, the grid size on the fine grid is $h=0.00625$.

Table 1 quantifies the maximum norm of the error when $x_*=0.33$ and $x_*=0.5077$. To eliminate effects of the homogeneous Dirichlet boundary conditions in the numerical solution, we compare the solution at time $t=0.33$. The error, $\tilde{e}$, is defined as the difference between the numerical and exact solutions at all grid points, except at the two grid points closest to x_*, where it is set to zero. In these two points, the error remains $\mathcal{O}(1)$ as the grid is refined. However, as shown by Table 1, these errors do not affect the accuracy away from x_*. As the grid size h goes to zero, the norm of the error should behave as

$$\|\tilde{e}(h)\| = Ch^p,$$

where C is a constant. When the grid size is reduced by a factor of two, we can therefore estimate the convergence rate as

$$p = \log_2\left(\|\tilde{e}(2h)\| / \|\tilde{e}(h)\|\right).$$

Table 1: Errors and convergence rates with energy conserving interpolation at time $t=0.33$ on the composite grid when the source is located at $x_*=0.33$ and $x_*=0.5077$, respectively. Here, h is the grid size on the fine grid. The two grid points closest to x_* are excluded from the norm computation.

	$x_*=0.33$		$x_*=0.5077$	
h	$\|\tilde{e}\|_\infty$	p	$\|\tilde{e}\|_\infty$	p
1.25×10^{-2}	1.37×10^{-4}	–	4.69×10^{-4}	–
6.25×10^{-3}	3.36×10^{-5}	2.02	1.37×10^{-4}	1.78
3.125×10^{-3}	8.45×10^{-6}	1.99	3.14×10^{-5}	2.12
1.5625×10^{-3}	2.12×10^{-6}	2.00	7.84×10^{-6}	2.00

We conclude that the numerical solution is second order accurate for both source locations. Note that $x_*=0.5077$ is located on the coarse grid, which explains why those errors are approximately four times larger than for $x_*=0.33$.

Our numerical experiments show that the composite grid discretization of a singular source works well with the energy conserving interpolation, and produces second order accurate solutions under two conditions. First, the combined source weight should be distributed according to (5.10) on the grid interface, and secondly, the moment conditions should be satisfied to one order higher than what is necessary for a uniform grid. Violation of either of these conditions leads to solution errors that are orders of magnitude larger, where the error is substantial over a significant portion of the computational domain, similar to the right side of Fig. 6.

5.3 Three space dimensions

To generalize the one-dimensional formulas to the three-dimensional case we discretize the singular source as a Cartesian product of one-dimensional grid functions. We focus our presentation on the grid configuration described in Section 4, where a fine grid of grid size h meets a coarse grid of grid size $2h$ along $z=c_0$. At the grid interface, the grid points in the x- and y-directions are aligned as in Fig. 4, while they are laid out as in Fig. 3 in the z-direction. To approximate $\delta(\mathbf{x}-\mathbf{x}_*)$, where $\mathbf{x}_*=(x_*,y_*,z_*)$, we use one-dimensional grid functions $d_h^{(x)}$, $d_{2h}^{(x)}$, $d_h^{(y)}$, $d_{2h}^{(y)}$, and $\bar{d}^{(z)}$. For example, with $Q=2$ the grid functions in the x-direction have three non-zero components on either the fine or the coarse grid

$$\begin{cases} d_h^{(x)}=(\cdots,0,\delta_{i-1}^{(x)},\delta_i^{(x)},\delta_{i+1}^{(x)},0,\cdots)^T, & \text{on the fine grid,} \\ d_{2h}^{(x)}=(\cdots,0,\Delta_{I-1}^{(x)},\Delta_I^{(x)},\Delta_{I+1}^{(x)},0,\cdots)^T, & \text{on the coarse grid.} \end{cases}$$

Here, grid points x_i and X_I are the closest to x_*. The grid functions $d_h^{(y)}$ and $d_{2h}^{(y)}$ have corresponding components centered around the grid points y_j and Y_J, which are the closest to y_*. In the z-direction, we use the composite grid notation from Section 5.1. For example, if the closest grid point to z_* is $z_{n_z}=Z_1$ and $Q=2$, we get the components

$$\bar{d}^{(z)}=(d_h^{(z)},d_{2h}^{(z)}),\quad d_h^{(z)}=\left(\cdots,0,\delta_{n_z-1}^{(z)},\delta_{n_z}^{(z)}\right)^T,\quad d_{2h}^{(z)}=\left(\Delta_1^{(z)},\Delta_2^{(z)},0,\cdots\right)^T.$$

Denote the discretization of the three-dimensional Dirac distribution by the three-dimensional composite grid function $\bar{d}^{(3)}$. Its components are defined by the Cartesian product

$$\bar{d}^{(3)}_{i,j,k} = \begin{cases} \Delta^{(x)}_i \Delta^{(y)}_j \Delta^{(z)}_k, & (i,j,k) \text{ belongs to the coarse grid,} \\ \delta^{(x)}_i \delta^{(y)}_j \delta^{(z)}_k, & (i,j,k) \text{ belongs to the fine grid.} \end{cases}$$

In the above example, $\Delta^{(x)}_i = 0$, for $i < I-1$ and $i > I+1$; $\Delta^{(y)}_j = 0$, for $j < J-1$ and $j > J+1$; and $\Delta^{(z)}_k = 0$, for $k \geq 3$. Therefore, $\bar{d}^{(3)}$ is only non-zero on the $3\times 3\times 2$ stencil $I-1 \leq i \leq I+1$, $J-1 \leq j \leq J+1$, $1 \leq k \leq 2$ on the coarse grid, with a corresponding $3\times 3\times 2$ stencil on the fine grid.

The sources should satisfy moment conditions in the composite scalar product over both grids:

$$\begin{aligned}(\bar{u},\bar{w})_{3,cg} &= (u,w)_{3,h,n_x,n_y,n_z} + (U,W)_{3,2h,N_x,N_y,N_z} \\ &= h^3 \sum_{i=1}^{n_x}\sum_{j=1}^{n_y}\sum_{k=1}^{n_z} \alpha^{(x)}_i \alpha^{(y)}_j \alpha^{(z)}_k u_{i,j,k} w_{i,j,k} + (2h)^3 \sum_{i=1}^{N_x}\sum_{i=1}^{N_y}\sum_{k=1}^{N_z} a^{(x)}_i a^{(y)}_j a^{(z)}_k U_{i,j,k} W_{i,j,k}.\end{aligned}$$

The three-dimensional moment conditions follow directly from the one-dimensional moment conditions. For example, the discrete moment of the polynomial functions

$$P(x,y,z) = x^q, \quad 0 \leq q \leq Q,$$

satisfy

$$\begin{aligned}&(\bar{d}^{(3)}, x^q)_{3,cg} \\ &= h^3 \sum_{i=1}^{n_x}\sum_{j=1}^{n_y}\sum_{k=1}^{n_z} \alpha^{(x)}_i \alpha^{(y)}_j \alpha^{(z)}_k \delta^{(x)}_i \delta^{(y)}_j \delta^{(z)}_k x^q_i + (2h)^3 \sum_{i=1}^{N_x}\sum_{j=1}^{N_y}\sum_{k=1}^{N_z} a^{(x)}_i a^{(y)}_j a^{(z)}_k \Delta^{(x)}_i \Delta^{(y)}_j \Delta^{(z)}_k X^q_i \\ &= \Big(h\sum_{i=1}^{n_x} \alpha^{(x)}_i \delta^{(x)}_i x^q_i\Big)\Big(h\sum_{j=1}^{n_y} \alpha^{(y)}_j \delta^{(y)}_j\Big)\Big(h\sum_{k=1}^{n_z} \alpha^{(z)}_k \delta^{(z)}_k\Big) \\ &\quad + \Big(2h\sum_{i=1}^{N_x} a^{(x)}_i \Delta^{(x)}_i X^q_i\Big)\Big(2h\sum_{j=1}^{N_y} a^{(y)}_j \Delta^{(y)}_j\Big)\Big(2h\sum_{k=1}^{N_z} a^{(z)}_k \Delta^{(z)}_k\Big). \end{aligned} \tag{5.19}$$

Each sum on the right hand side of (5.19) can be identified as a one-dimensional scalar product

$$h\sum_{i=1}^{n_x} \alpha^{(x)}_i \delta^{(x)}_i x^q_i = (d^{(x)}_h, x^q)_{1,h,1,n_x} = x^q_*, \qquad 2h\sum_{i=1}^{N_x} a^{(x)}_i \Delta^{(x)}_i X^q_i = (d^{(x)}_{2h}, x^q)_{1,2h,1,N_x} = x^q_*,$$

$$h\sum_{j=1}^{n_y} \alpha^{(y)}_j \delta^{(y)}_j = (d^{(y)}_h, 1)_{1,h,1,n_y} = 1, \qquad 2h\sum_{j=1}^{N_y} a^{(y)}_j \Delta^{(y)}_j = (d^{(y)}_{2h}, 1)_{1,2h,1,N_y} = 1,$$

$$h\sum_{k=1}^{n_z} \alpha^{(z)}_k \delta^{(z)}_k = (d^{(z)}_h, 1)_{1,h,1,n_z}, \qquad 2h\sum_{k=1}^{N_z} a^{(z)}_k \Delta^{(z)}_k = (d^{(z)}_{2h}, 1)_{1,2h,1,N_z}.$$

Since

$$\left(d_h^{(z)},1\right)_{1,h,1,n_z}+\left(d_{2h}^{(z)},1\right)_{1,2h,1,N_z}=\left(\bar{d}^{(z)},1\right)_{1,cg}=1,$$

it follows that

$$\left(\bar{d}^{(3)},x^q\right)_{3,cg}=x_*^q,\quad q=0,1,\cdots,Q.$$

Because we have restricted the grid refinement boundary to only occur in the z-direction, the sums over i and j will always be over uniform grids. This leads to the general expression

$$\left(\bar{d}^{(3)},x^q y^l z^m\right)_{3,cg}=x_*^q y_*^l\left(\bar{d}^{(z)},z^m\right)_{1,cg}=x_*^q y_*^l z_*^m, \tag{5.20}$$

for all $0\le q,l,m\le Q$.

The gradient of the Diract distribution, $\nabla\delta(\mathbf{x}-\mathbf{x}_*)$, is discretized by a composite grid function $\bar{\mathbf{d}}'$. Similar to the Dirac distribution, $\bar{\mathbf{d}}'$ is defined by a Cartesian product of one-dimensional grid functions

$$\bar{\mathbf{d}}'_{i,j,k}=\begin{pmatrix}\Delta_i'^{(x)}\Delta_j^{(y)}\Delta_k^{(z)}\\ \Delta_i^{(x)}\Delta_j'^{(y)}\Delta_k^{(z)}\\ \Delta_i^{(x)}\Delta_j^{(y)}\Delta_k'^{(z)}\end{pmatrix},\quad \text{on the coarse grid,} \tag{5.21a}$$

$$\bar{\mathbf{d}}'_{i,j,k}=\begin{pmatrix}\delta_i'^{(x)}\delta_j^{(y)}\delta_k^{(z)}\\ \delta_i^{(x)}\delta_j'^{(y)}\delta_k^{(z)}\\ \delta_i^{(x)}\delta_j^{(y)}\delta_k'^{(z)}\end{pmatrix},\quad \text{on the fine grid.} \tag{5.21b}$$

The three-dimensional moment conditions for $\bar{\mathbf{d}}'$ follows from the one-dimensional properties in the same way as for $\bar{d}^{(3)}$. However, note that the rule of using $Q=3$ for d' while $Q=2$ for d means that not all of the moments conditions for $\bar{\mathbf{d}}'$ are satisfied to the same order. As a topic for future investigation, it might be advantageous to use $Q=3$ also for d when they are used as part of a moment source.

6 Numerical experiments

6.1 Method of manufactured solutions

We start by evaluating the error in the numerical solution on a composite grid, when both the material and the solution are smooth. Let the computational domain be the cube $(x,y,z)\in[0,5]^3$ and impose a free surface boundary condition on the $z=0$ boundary and Dirichlet conditions on all other boundaries. We take the material properties to be

$$\begin{aligned}\rho(x,y,z)&=A_\rho\left(2+\sin(\omega_m x+\theta_m)\cos(\omega_m y+\theta_m)\sin(\omega_m z+\theta_m)\right),\\ \mu(x,y,z)&=A_\mu\left(3+\cos(\omega_m x+\theta_m)\sin(\omega_m y+\theta_m)\sin(\omega_m z+\theta_m)\right),\\ \lambda(x,y,z)&=A_\lambda\left(2+\sin(\omega_m x+\theta_m)\sin(\omega_m y+\theta_m)\cos(\omega_m z+\theta_m)\right),\end{aligned}$$

where ω_m=3.2, θ_m=0.8, A_ρ=2, A_μ=3, and A_λ=1. The internal forcing, boundary forcing and initial conditions are chosen such that the exact (manufactured) solution becomes

$$u_e(x,y,z,t) = \sin\big(\omega(x-c_e t)\big)\sin(\omega y+\theta)\sin(\omega z+\theta), \tag{6.1}$$

$$v_e(x,y,z,t) = \sin(\omega x+\theta)\sin\big(\omega(y-c_e t)\big)\sin(\omega z+\theta), \tag{6.2}$$

$$w_e(x,y,z,t) = \sin(\omega x+\theta)\sin(\omega y+\theta)\sin\big(\omega(z-c_e t)\big), \tag{6.3}$$

with $\omega=3$, $\theta=0.2$, and $c_e=1.3$.

The computational grid consists of a base grid with size $2h$ in $2\le z\le 5$ and a refined grid with size h in $0\le z\le 2$. In terms of the number of grid points in the x-direction on the base grid, the grid sizes are

$$2h = 5/(N_x-1), \qquad h = 2.5/(N_x-1),$$

respectively. For sufficiently small h, the norm of the error should behave as $e(h)=Ch^p$, where C is a constant. As before, we estimate the convergence rate as $p=\log_2[e(2h)/e(h)]$. The errors in the numerical solutions and the convergence rates, evaluated in max and L_2-norm at time $t=4.8$ are given in Table 2. As we can see the error in both norms is of the order $\mathcal{O}(h^2)$.

Table 2: Errors in the numerical solution and convergence rates at time $t=4.8$, on a composite grid, when the exact solution is (6.1)-(6.3).

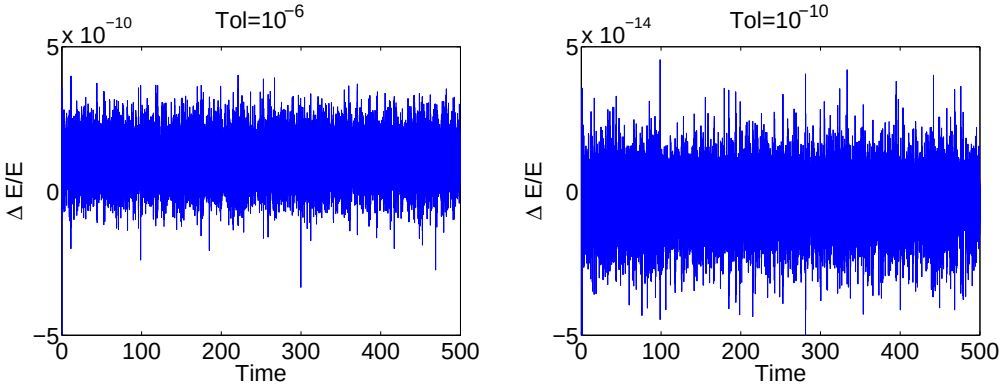
N_x	$2h$	$\|\mathbf{u}(\cdot,t)-\mathbf{u}_e(\cdot,t)\|_2$	p_2	$\|\mathbf{u}(\cdot,t)-\mathbf{u}_e(\cdot,t)\|_\infty$	p_∞
31	1.67×10^{-1}	2.36×10^{-1}	–	6.16×10^{-2}	–
61	8.33×10^{-2}	5.74×10^{-2}	2.04	1.59×10^{-2}	1.95
121	4.17×10^{-2}	1.42×10^{-2}	2.02	4.17×10^{-3}	1.92
241	2.08×10^{-2}	3.52×10^{-3}	2.01	1.03×10^{-3}	2.02

6.2 Energy conservation test

To test the energy conserving property of the new interface condition, we solve the elastic wave equation on the domain $(x,y,z)\in[0,5]^3$ and impose a free surface boundary condition on the z=0 boundary and homogeneous Dirichlet conditions on all other boundaries. The material is

$$\rho = 2+\theta, \quad \mu = 3+\theta \quad \text{and} \quad \lambda = (r-2)^2\mu+\theta,$$

where the velocity ratio $r=c_p/c_s=3$ in the computations below. Here θ is a stochastic variable, uniformly distributed between zero and one. Furthermore, the initial data are also assigned by stochastic variables with uniform distribution. Note that every evaluation of θ gives a different value between 0 and 1, resulting in extremely noisy material properties and initial data. The computational grid is the same two-grid configuration as in Section 6.1, here with $N_x=31$. The equations were integrated to time 500, which corresponds to 30,746 time steps.



(a) (b)

Figure 8: Relative energy change vs. time with random material and initial data. The convergence tolerance in the Jacobi iteration is 10^{-6} in Fig. (a) (to the left) and 10^{-10} in Fig. (b) (to the right).

Fig. 8(a) shows the relative change of energy, defined as

$$(e^{n+1}_{3,cg} - e^{n}_{3,cg})/e^{0}_{3,cg},$$

as function of time, where $e^{n}_{3,cg}$ is the energy defined in (4.1). The average relative change per time step is 1.1×10^{-10}, hence there is a small increase in energy. This increase is due to the iterative Jacobi iteration that solves the interface equations (4.3)-(4.7) only to a given residual tolerance. For the computation in Fig. 8(a), the residual tolerance was set to 10^{-6}. We show in Fig. 8(b) results from exactly the same computation, but with the residual tolerance set to 10^{-10}. The average relative energy increase per time step is now -3.9×10^{-15}, i.e., the energy changes are smaller and decreasing. This is consistent with the analysis presented in Section 4, which states that the energy will be conserved perfectly if the interface equations are solved exactly.

6.3 Source discretization

To test the discretization of a moment tensor source term for the 3-D elastic wave equation, we consider the half-space problem with homogeneous material properties: $\rho = 2{,}650$, $c_s = 2{,}000$, $c_p = 4{,}000$, corresponding to

$$\mu = \rho c_s^2, \qquad \lambda = \rho(c_p^2 - 2c_s^2).$$

In the simulation, the half-space $z \geq 0$ is truncated to the computational domain $(x,y,z) \in [0,4\times 10^4]^2 \times [0,5\times 10^4]$. The moment source (5.1) is located at the center of the computational domain in the (x,y)-plane, at depth z_*

$$\mathbf{x}_* = \left(2\times 10^4, 2\times 10^4, z_*\right)^T,$$

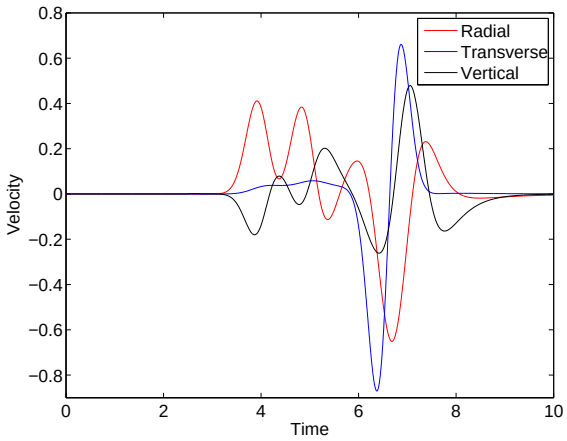


Figure 9: Semi-analytical solution as function of time at receiver location $x_r=2.6\times10^4$, $y_r=2.8\times10^4$, $z_r=0$ for a source at depth $z_*=2.1\times10^3$ in a uniform material.

and is characterized by the matrix

$$\mathcal{M}=10^{18}\begin{pmatrix}0&1&0\\1&0&0\\0&0&0\end{pmatrix},$$

which in seismic applications often is described by the seismic moment $m_0=10^{18}$ and the angles dip$=90^\circ$, rake$=0^\circ$, and strike$=0^\circ$ (when the x-axis is parallel to North), see [1]. The time function is the Gaussian

$$g(t)=\frac{1}{\sigma\sqrt{2\pi}}e^{-(t-t_0)^2/2\sigma^2}, \tag{6.4}$$

where $\sigma=0.25$, $t_0=6\sigma$. The solution is recorded in time at $x_r=2.6\times10^3$, $y_r=2.8\times10^4$, $z_r=0$ and compared to a semi-analytical frequency-wavenumber (FK) solution [13,34,37] using the FK code [36]. An example is shown in Fig. 9, where we present the radial, transverse, and vertical components of the solution, as is customary in seismology. The radial, transverse, and vertical components are defined in a polar coordinate system centered at the (x,y)-location of the source, with the vertical component in the negative z-direction, i.e., $u_{\text{vert}}=-w$. Since $x_r-x_*=0.6\times10^4$, $y_r-y_*=0.8\times10^4$, the radial component is $u_{\text{rad}}=0.6u+0.8v$, and the transverse component is $u_{\text{tran}}=-0.8u+0.6v$.

We measure the error in the time interval $0\le t\le T$ using the norms

$$\|\mathbf{u}(\mathbf{x}_r,\cdot)\|_2^2=\frac{1}{T}\int_0^T\left(u(\mathbf{x}_r,t)^2+v(\mathbf{x}_r,t)^2+w(\mathbf{x}_r,t)^2\right)\mathrm{d}t, \tag{6.5}$$

$$\|\mathbf{u}(\mathbf{x}_r,\cdot)\|_\infty=\max_{0\le t\le T}\sqrt{u(\mathbf{x}_r,t)^2+v(\mathbf{x}_r,t)^2+w(\mathbf{x}_r,t)^2}. \tag{6.6}$$

As before, we estimate the convergence rate using the formula $p=\log_2[e(2h)/e(h)]$, where $e(h)$ is the norm of the error in the solution on a composite grid with characteristic grid size h. The computational domain is discretized on a composite grid where the fine grid with size h covers $0\le z\le 2\times 10^3$ and the base grid with size $2h$ covers $2\times 10^3\le z\le 5\times 10^4$. A free surface boundary condition is imposed on the $z=0$ boundary and homogeneous Dirichlet conditions are used on all other boundaries together with a damping sponge layer. As was indicated in Section 5.3, the stencil for discretizing the gradient of the Dirac distribution (5.21) should to be 4 points wide ($Q=3$) to obtain second order accuracy. Since we center the source discretization stencil around the source location $\mathbf{x}_*$, the composite grid formula is only used if the source is within 2 grid sizes of the refinement boundary. In the results given in Table 3, the source location $z_*=2{,}100$ therefore uses the composite grid discretization for all presented grid sizes, while the case $z_*=4{,}000$ always uses the uniform grid discretization formula. Our results indicate that the solutions on the two finest composite grids are in the asymptotic regime and give $\mathcal{O}(h^2)$ accuracy for both source locations.

Table 3: Errors and convergence rates in the numerical solution on a composite grid, with a moment tensor source at depth z_*, measured at the receiver location $x_r=2.6\times 10^4$, $y_r=2.8\times 10^4$, $z_r=0$. The norms are taken over the period $0\le t\le 10$ after which the solution is essentially zero.

z_*	$2h$	$\|\mathbf{u}_s(\mathbf{x}_r,\cdot)\|_2$	$\|\mathbf{u}(\mathbf{x}_r,\cdot)-\mathbf{u}_s(\mathbf{x}_r,\cdot)\|_2$	p_2	$\|\mathbf{u}(\mathbf{x}_r,\cdot)-\mathbf{u}_s(\mathbf{x}_r,\cdot)\|_\infty$	p_∞
4000	400	2.52×10^{-1}	1.53×10^{-1}	–	6.16×10^{-1}	–
4000	200		4.85×10^{-2}	1.66	2.26×10^{-1}	1.45
4000	100		1.25×10^{-2}	1.96	5.88×10^{-2}	1.94
2100	400	2.87×10^{-1}	1.99×10^{-1}	–	7.41×10^{-1}	–
2100	200		5.96×10^{-2}	1.74	2.64×10^{-1}	1.49
2100	100		1.34×10^{-2}	2.15	6.37×10^{-2}	2.05

In a second test, we calculate the error as function of the source depth z_*, for grid sizes $h=100$ and $2h=200$ in the refined and base grids, respectively. The results shown in Fig. 10 demonstrate that the error is essentially independent of the distance between the source and the grid refinement boundary.

6.4 The layer over half-space problem

Here we consider a variation of the layer over half-space problem which was used by Day et al. [9] to evaluate the accuracy of various seismic wave propagation codes. We consider the half-space $z\ge 0$, where the material properties in the top layer ($0\le z\le 2{,}000$) are $\rho=2{,}500$, $c_p=2{,}000$, and $c_s=1{,}000$. Below the top layer ($z>2{,}000$), the material properties are $\rho=2{,}650$, $c_p=4{,}000$, and $c_s=2{,}000$. Because of the slow material in the top layer, the solution is more complex compared to a homogeneous material, and we increase the size of the computational domain to $(x,y,z)\in[0,5\times 10^4]^3$ to further reduce artificial reflections from the outer boundaries. As before, we impose a free surface boundary condition on

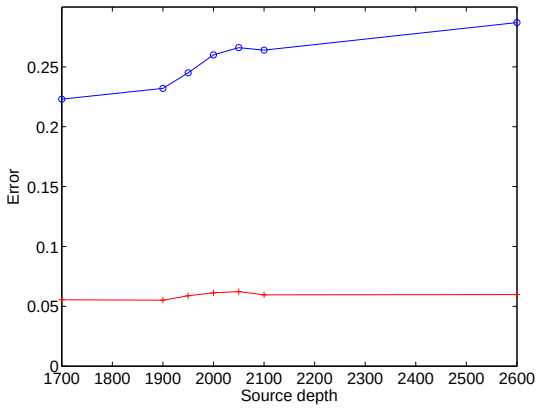


Figure 10: Errors in the numerical solution on a composite grid, with a moment tensor source at depth z_*, measured at the receiver location $x_r = 2.6\times 10^4$, $y_r = 2.8\times 10^4$, $z_r = 0$. The L_2-norm is shown in red '+' and the max norm in blue 'o'. The norms are taken over the period $0 \le t \le 10$ after which the solution is essentially zero. The grid size is $2h = 200$ in the base grid and $h = 100$ in the refined grid.

the $z = 0$ surface and homogeneous Dirichlet conditions together with a sponge layer on all other boundaries. Similar to Section 6.3, we place a point moment tensor source with $m_0 = 10^{18}$, dip$= 90^\circ$, rake$= 0^\circ$, and strike$= 0^\circ$, at the center of the (x,y)-plane at depth z_*

$$\mathbf{x}_* = (2.5\times 10^4, 2.5\times 10^4, z_*)^T,$$

and use the Gaussian time-function (6.4).

The velocity structure of this problem makes it an ideal candidate for grid refinement and we use a refined grid with size $h = 50$ in the top layer ($0 \le z \le 2{,}000$), and a base grid with size $2h = 100$ in $z \ge 2{,}000$. The example from the introduction shown in Fig. 2 reports the L_2-norms of the solution and the error as function of time at receiver location $\mathbf{x}_r = (3.1\times 10^4, 3.3\times 10^4, 0)^T$ when the source is located at $z_* = 2{,}550$. As in the previous section, the error is evaluated by comparing the numerical simulation with a solution from the semi-analytical FK code [36]. Since the source is separated by more than two grid sizes from the grid interface, we use the uniform source discretization formula. To further demonstrate the importance of the conservative interpolation, we compare it with a 3-D generalization of the straight forward non-conservative interpolation formula (5.18). Due to the material discontinuity, the gradient of the solution is discontinuous at $z = 2{,}000$, which also coincides with the grid refinement interface. The accuracy of the non-conservative interpolation is therefore reduced, even though the source is separated from the grid refinement interface. As a result the solution error is larger for the non-conservative approach compared to the conservative interpolation, see Table 4. A more serious problem occurs at later times, when the non-conservative interpolation formula makes the simulation go unstable, while the conservative approach remains stable, see Fig. 2. For this simple material model, the instability can be controlled by adding artificial dissipation to the non-conservative scheme, but for more complex heterogeneous

Table 4: Errors in the numerical solutions of the layer over half-space problem.

Case	Grid size	$\|\mathbf{u}(\mathbf{x}_r,\cdot)-\mathbf{u}_s(\mathbf{x}_r,\cdot)\|_2$	$\|\mathbf{u}(\mathbf{x}_r,\cdot)-\mathbf{u}_s(\mathbf{x}_r,\cdot)\|_\infty$
Conservative	100/50	4.88×10^{-2}	2.77×10^{-1}
Non-conservative	100/50	1.41×10^{-1}	5.58×10^{-1}
Uniform grid	50	8.61×10^{-2}	3.76×10^{-1}

Table 5: Computational requirements for different simulation approaches for solving the layer over half-space problem to time $t=25$.

Grid type	Grid points	Interpolation	Cores	Wall clock [s]	Total CPU [s]
Composite	1.62×10^8	Conservative	80	1,525	1.22×10^5
Composite	1.62×10^8	Non-conservative	80	2,660	2.13×10^5
Uniform	1.00×10^9	N/A	1024	1,534	1.57×10^6

materials it is difficult to automatically choose both the strength of the dissipation and the time step. Furthermore, artificial dissipation requires additional computational resources, often slowing down the simulation by 50 percent or more.

For comparison, we also simulated the layer over half-space problem on a uniform grid with size $h=50$, see Fig. 2. In this case, no jump conditions are enforced at the material discontinuity and the finite difference stencil is applied across the discontinuous material properties without any special treatment. This leads to significant numerical errors in the solution, and the uniform grid simulation produces larger errors compared to the conservative composite grid case, see Table 4.

The computational requirements of the different computational approaches are summarized in Table 5. Not surprisingly, the uniform grid calculation requires more than six times the number of grid points, and has to be run on a much larger partition of the parallel machine. Since the conservative composite grid case has a grid size which is in perfect parity with the material velocities, it only requires 1,530 time steps to get to time $t=25$, while the uniform grid approach has to use 3,061 time steps. The total CPU time for the uniform grid case is therefore about 13 times longer than in the conservative composite grid case. The computation with non-conservative interpolation also requires 3,061 time steps because the ghost points and the grid points on the material interface in the refined grid must use the faster velocities of the half-space to produce a solution with reasonable accuracy. As a result, that calculation took about 75 percent longer than the conservative case, even though the conservative method requires additional computations to satisfy the jump conditions. This example illustrates the significant savings that can be realized by using a composite grid in seismic applications. In more realistic material models, where the material velocities often vary by a factor of ten, several levels of grid refinements can be used to further improve on the efficiency.

To make it straight forward to display the solution in a vertical plane extending through both the source and the receiver, we rotate the computational grid clock-wise

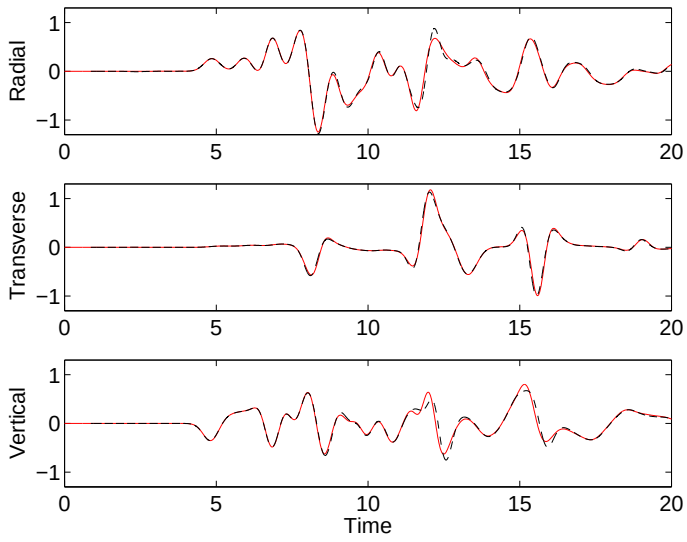


Figure 11: Radial, transverse and vertical velocity components as function of time at $x_r=3.5\times10^4$, $y_r=2.5\times10^4$, $z_r=0$. The dashed black line is the semi-analytical FK solution and the red solid line is the numerical solution on the composite grid.

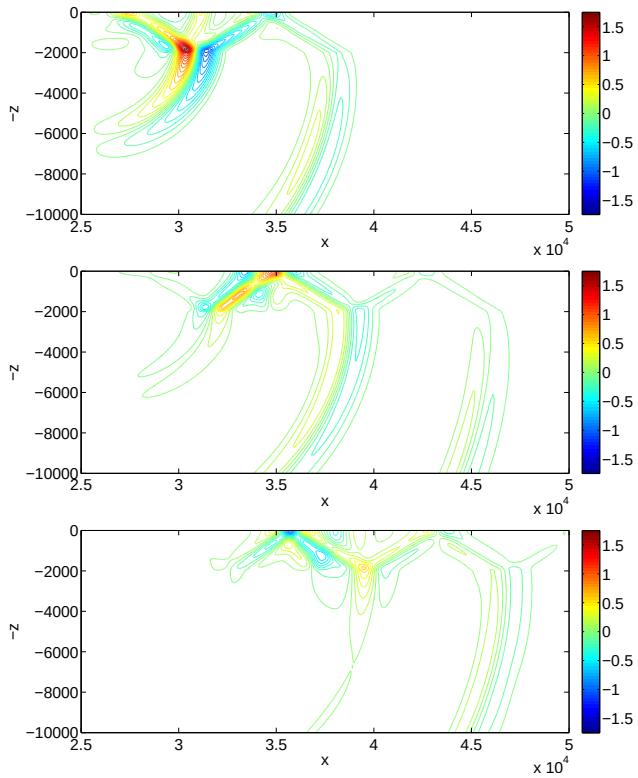


Figure 12: Transverse velocity component in the source to receiver plane at times 8 (top), 12 (center), and 16 (bottom).

in the horizontal plane by the angle $\cos^{-1}(0.6)\approx 53.13^\circ$. In the rotated coordinate system, the receiver is located at $x_r=35{,}000$, $y_r=25{,}000$, $z_r=0$, and the strike angle in the moment tensor source is now -51.13° relative to the rotated x-direction. In Fig. 11, we plot the three velocity components as function of time at the receiver location, providing a visual comparison of the solutions from the numerical simulation and the semi-analytical FK code. Overall, the wave forms agree very well, in particular for the transverse component. There are however some small amplitude errors in the radial and vertical components around time 12, and in the vertical component around time 16. To estimate the resolution in terms of grid points per shortest wave length, we approximate the upper power frequency in the Gaussian time function (6.4) by $f_{\max}\approx 2.5 f_0$, where the fundamental frequency is $f_0=1/2\pi\sigma\approx 0.6366$. This results in the shortest wave length $L=c_s/f_{\max}\approx 628.3$, corresponding to approximately $P=L/h\approx 12.6$ grid points per wave length.

Note the three main wave pulses in the transverse velocity arriving around times 8, 12, and 15.5. Snap-shots at times 8, 12, and 16, of the same velocity component along a vertical cross-section through the source-receiver plane are shown in Fig. 12. Note that the top layer acts as a wave guide, and the motion is dominated by waves trapped between the material interface and the free surface. As expected, the wave lengths in the fast material below $z=2{,}000$ are significantly longer than in the top layer, further illustrating the sensibility of using a refined grid where the velocities are low.

7 Conclusions

We have described a second order accurate and energy conserving discretization of the elastic wave equation in second order formulation on a composite grid. The composite grid consists of a set of structured rectangular component grids with hanging nodes on the grid refinement interface, allowing the grid size to approximately follow the material velocity structure in seismic applications. We have also developed second order accurate discretizations of singular source terms of point force and point moment tensor type, that work in the vicinity of grid refinement interfaces. This allows complex ruptures to be modeled on a composite grid, where the fault surface extends through grid refinement interfaces. The composite grid method and the source discretization have been implemented as part of version 2.0 of the open source software WPP [29], which also handles free surfaces on a realistic topography.

Plans for the near future include generalizations to visco-elastic materials. As a longer term goal, it would be desirable to raise the spatial and temporal accuracy to fourth order. This would improve the efficiency of the method in terms of the number of grid points per wave length that is required to obtain a given accuracy. The main challenge is clearly to develop a stable fourth order spatial discretization, because the temporal accuracy can easily be matched by using a modified equation approach, see for example [8].

Much work is needed to develop a higher order accurate summation by parts operator for the elastic wave equation with heterogeneous material properties. A good starting

point would be to first develop a higher order accurate approach for a uniform grid, using the framework developed for higher order approximations of second derivatives in 1-D [22]. Cross-derivatives could be discretized as first derivatives in each spatial direction using the higher order approach in [31]. The combined scheme would need to satisfy the free surface boundary condition to higher order accuracy, and handle heterogeneous material properties in a stable fashion. Once a higher order accurate uniform grid discretization has been established, the remaining challenge would be a stable and higher order accurate treatment of grid refinement interfaces with hanging nodes.

To accurately handle elastic wave propagation through complex 3-D geological structures in the earth, jump conditions should be enforced across the material interfaces. For this purpose it would be desirable to generalize the embedded boundary method with jump conditions for the scalar wave equation [19].

Acknowledgments

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. This is contribution LLNL-JRNL-419382.

References

[1] K. Aki and P. G. Richards, Quantitative Seismology, University Science Books, Sausalito, CA, USA, 2nd edition, 2002.
[2] D. Appelö and N. A. Petersson, A stable finite difference method for the elastic wave equation on complex geometries with free surfaces, Comm. Comput. Phys., 5 (2009), 84–107.
[3] A. Bamberger, R. Glowinski and Q. H. Tran, A domain decomposition method for the acoustic wave equation with discontinuous coefficients and grid change, SIAM J. Numer. Anal., 34 (1997), 603–639.
[4] M. J. Berger, Stability of interfaces with mesh refinement, Math. Comp., 45 (1985), 301–318.
[5] M. J. Berger, On conservation at grid interfaces, SIAM J. Numer. Anal., 24 (1987), 967–984.
[6] G. Chesshire and W. D. Henshaw, Composite overlapping meshes for the solution of partial differential equations, J. Comput. Phys., 90(1) (1990), 1–64.
[7] G. Cohen and S. Fauqueux, Mixed spectral finite elements for the linear elasticity system in unbounded domains, SIAM J. Sci. Comput., 26(3) (2005), 864–884.
[8] G. C. Cohen, Higher-Order Numerical Methods for Transient Wave Equations, Springer, 2002.
[9] S. M. Day, J. Bielak, D. Dreger, S. Larsen, R. Graves, A. Pitarka and K. B. Olsen, Tests of 3D elastodynamic codes: lifelines program task 1A01, Technical report, Pacific Earthquake Engineering Center, 2001.
[10] M. Dumbser and M. Käser, An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes - II: the three-dimensional isotropic case, Geophys. J. Int., 167(1) (2006), 319–336.

[11] M. Dumbser, M. Käser and J. de la Puente, Arbitrary high-order finite volume schemes for seismic wave propagation on unstructured meshes in 2D and 3D, Geophys. J. Int., 171(2) (2007), 665–694.
[12] R. W. Graves, Simulating seismic-wave propagation in 3-D elastic media using staggered-grid finite differences, Bull. Seismo. Soc. Amer., 86(4) (1996), 1091–1106.
[13] N. A. Haskell, Radiation pattern of surface waves from point sources in a multi-layered medium, Bull. Seismo. Soc. Amer., 54(1) (1964), 377–393.
[14] M. Käser and M. Dumbser, An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes - I: the two-dimensional isotropic case with external source terms, Geophys. J. Int., 166(2) (2006), 855–877.
[15] H.-O. Kreiss and J. Oliger, Comparison of accurate methods for the integration of hyperbolic equations, Tellus., 24 (1972), 199–215.
[16] D. Komatitsch and J. Tromp, Introduction to the spectral element method for three-dimensional seismic wave propagation, Geophys. J. Int., 139 (1999), 806–822.
[17] R. M. J. Kramer, C. Pantano and D. I. Pullin, A class of energy stable, high-order finite-difference interface schemes suitable for adaptive mesh refinement of hyperbolic problems, J. Comput. Phys., 226 (2007), 1458–1484.
[18] R. M. J. Kramer, C. Pantano and D. I. Pullin, Nondissipative and energy-stable high-order finite-difference interface schemes for 2-D patch-refined grids, J. Comput. Phys., 228 (2009), 5280–5297.
[19] H.-O. Kreiss and N. A. Petersson, An embedded boundary method for the wave equation with discontinuous coefficients, SIAM J. Sci. Comput., 28 (2006), 2054–2074.
[20] A. R. Levander, Fourth-order finitie-difference P-SV seismograms, Geophysics, 53(11) (1988), 1425–1436.
[21] K. Mattsson, F. Ham and G. Iaccarino, Stable and accurate wave-propagation in discontinuous media, J. Comput. Phys., 227 (2008), 8753–8767.
[22] K. Mattsson and J. Norström, Summation by parts operators for finite difference approximations of second derivatives, J. Comput. Phys., 199 (2004), 503–540.
[23] S. Nilsson, N. A. Petersson, B. Sjögreen and H.-O. Kreiss, Stable difference approximations for the elastic wave equation in second order formulation, SIAM J. Numer. Anal., 45 (2007), 1902–1936.
[24] J. Nordström, J. Gong, E. van der Weide and M. Svärd, A stable and conservative high order multi-block method for the compressible Navier-Stokes equations, J. Comput. Phys., 228 (2009), 9020–9035.
[25] F. Olsson and N. A. Petersson, Stability of interpolation on overlapping grids, Comput. Fluids., 25(6) (1996), 583–605.
[26] E. Pärt-Enander and B. Sjögreen, Conservative and non-conservative interpolation between overlapping grids for finite volume solutions of hyperbolic problems, Comput. Fluids., 23 (1994), 551–574.
[27] R. B. Pember, J. B. Bell, P. Colella, W. Y. Crutchfield and M. L. Welcome, An adaptive Cartesian grid method for unsteady compressible flow in irregular regions, J. Comput. Phys., 120 (1995), 278–304.
[28] N. A. Petersson and B. Sjögreen, An energy absorbing far-field boundary condition for the elastic wave equation, Comm. Comput. Phys., 6 (2009), 483–508.
[29] N. A. Petersson and B. Sjögreen, Reference guide to WPP version 2.0, Technical Report LLNL-TR-422928, Lawrence Livermore National Laboratory, 2010. (Source code available from `https://computation.llnl.gov/casc/serpentine`.)

[30] G. Starius, On composite mesh difference methods for hyperbolic differential equations, Num. Math., 35 (1980), 241–255.
[31] B. Strand, Summation by parts for finite difference approximations for $\mathrm{d}/\mathrm{d}x$, J. Comput. Phys., 110 (1994), 47–67.
[32] J. Virieux, P-SV wave propagation in heterogeneous media: velocity-stress finite-difference method, Geophysics, 51(4) (1986), 889–901.
[33] J. Waldén, On the approximation of singular source terms in differential equations, Numer. Meth. Part. Diff. Eq., 15 (1999), 503–520.
[34] C. Y. Wang and R. B. Hermann, A numerical study of P-, SV- and SH-wave generation in a plane layered medium, Bull. Seismo. Soc. Amer., 70(4) (1980), 1015–1036.
[35] Y. Wang, J. Xu and G. T. Schuster, Viscoelastic wave simulation in basins by a variable-grid finite-difference method, Bull. Seism. Soc. Am., 91 (2001), 1741–1749.
[36] L. Zhu, The Frequency-Wavenumber (FK) Synthetic Seismogram Package, 1999, available at `http://www.eas.slu.edu/People/LZhu/home.html`.
[37] L. Zhu and L. A. Rivera, A note on the dynamic and static displacements from a point source in multilayered media, Geophys. J. Int., 148 (2002), 619–627.