Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes

Daniel Peter,1 Dimitri Komatitsch,2,3 Yang Luo,1 Roland Martin,2 Nicolas Le Goff,2 Emanuele Casarotti,4 Pieyre Le Loher,2 Federica Magnoni,4 Qinya Liu,5 Céline Blitz,2 Tarje Nissen-Meyer,6 Piero Basini6 and Jeroen Tromp1,7

1Princeton University, Department of Geosciences, 318 Guyot Hall, Princeton, NJ 08544, USA. E-mail: dpeter@princeton.edu
2Université de Pau et des Pays de l’Adour, CNRS & INRIA Magique-3D, Laboratoire de Modélisation et d’Imagerie en Géosciences UMR 5212, Avenue de l’Université, 64013 Pau Cedex, France
3Institut universitaire de France, 103 boulevard Saint-Michel, 75005 Paris, France
4Istituto Nazionale di Geofisica e Vulcanologia, Via di Vigna Murata 605, 00143, Rome, Italy
5Department of Physics, University of Toronto, Ontario, Canada
6Institute of Geophysics, ETH Zurich, Sonneggstr. 5, CH-8092 Zurich, Switzerland
7Princeton University, Program in Applied & Computational Mathematics, Princeton, NJ 08544, USA

Accepted 2011 April 12. Received 2011 April 8; in original form 2011 February 2

SUMMARY
We present forward and adjoint spectral-element simulations of coupled acoustic and (an)elastic seismic wave propagation on fully unstructured hexahedral meshes. Simulations benefit from recent advances in hexahedral meshing, load balancing and software optimization. Meshing may be accomplished using a mesh generation tool kit such as CUBIT, and load balancing is facilitated by graph partitioning based on the SCOTCH library. Coupling between fluid and solid regions is incorporated in a straightforward fashion using domain decomposition. Topography, bathymetry and Moho undulations may be readily included in the mesh, and physical dispersion and attenuation associated with anelasticity are accounted for using a series of standard linear solids. Finite-frequency Fréchet derivatives are calculated using adjoint methods in both fluid and solid domains. The software is benchmarked for a layercake model. We present various examples of fully unstructured meshes, snapshots of wavefields and finite-frequency kernels generated by Version 2.0 ‘Sesame’ of our widely used open source spectral-element package SPECFEM3D.

Key words: Tomography; Interferometry; Computational seismology; Wave propagation.

1 INTRODUCTION
We present a new software package, SPECFEM3D Version 2.0 ‘Sesame’, capable of simulating forward and adjoint seismic wave propagation on fully unstructured hexahedral meshes of arbitrary shaped model domains. In view of unrelenting growth in computational power, it has become more and more important to develop software capable of harnessing powerful computers to address a broad range of seismological forward and inverse problems. A well-established numerical technique for solving such problems in a fast and highly accurate manner is the spectral-element method (SEM). The SEM was originally developed in computational fluid dynamics (Patera 1984; Maday & Patera 1989) and has been successfully adapted to address problems in seismic wave propagation. Early seismic wave propagation applications of the SEM, utilizing Legendre basis functions and a perfectly diagonal mass matrix, include Cohen et al. (1993), Komatitsch (1997), Faccioti et al. (1997), Casadei & Gabellini (1997), Komatitsch & Vilotte (1998) and Komatitsch & Tromp (1999), whereas applications involving Chebyshev basis functions and a non-diagonal mass matrix include Seriani & Priolo (1994), Priolo et al. (1994) and Seriani et al. (1995).

The SEM is a continuous Galerkin technique, which may be made discontinuous (Bernardi et al. 1994; Chaljub 2000; Kopriva et al. 2002; Chaljub et al. 2003; Legay et al. 2005; Kopriva 2006; Wilcox et al. 2010; Acosta Minolia & Kopriva 2011); it is then close to a particular case of the discontinuous Galerkin technique (Reed & Hill 1973; Arnold 1982; Falk & Richter 1999; Hu et al. 1999; Cockburn et al. 2000; Giraldo et al. 2002; Rivière & Wheeler 2003; Monk & Richter 2005; Grote et al. 2006; Ainsworth et al. 2006; Bernacki et al. 2006; Dumbser & Käser 2006; De Basabe et al. 2008; De la Puente et al. 2009; Wilcox et al. 2010; De Basabe & Sen 2010; Etienne et al. 2010), with optimized efficiency because of its tensorized basis functions (Wilcox et al. 2010; Acosta Minolia & Kopriva 2011).

An important feature of the SEM is that it can accurately handle very distorted mesh elements (Oliveira & Seriani 2011), and
thus conforming non-structured mesh doubling bricks can efficiently accommodate mesh size variations (Komatitsch & Tromp 2002a, 2004; Lee et al. 2008, 2009a,b). The method has very good accuracy and convergence properties, such as a spectral rate of convergence (Canuto et al. 1988; Maday & Patera 1989; Seriani & Priolo 1994; Deville et al. 2002; Cohen 2002; De Basabe & Sen 2007; Seriani & Oliveira 2008). In this sense the SEM is close to the family of spectral-mesh methods (see e.g. Canuto et al. 1988; Carcione et al. 1988a, 1992; Carcione & Wang 1993; Komatitsch et al. 1996), but combined with the flexibility of finite elements, in particular in terms of mesh design. For reviews of the SEM in seismology, see for example, Komatitsch et al. (2005), Chaljub et al. (2007), Tromp et al. (2008) and Fichtner (2010).

The SEM is well suited to parallel implementations on very large supercomputers (Komatitsch & Tromp 2002a; Komatitsch et al. 2003; Tsuibo et al. 2003; Komatitsch et al. 2008; Carrington et al. 2008; Komatitsch et al. 2010b) as well as on clusters of GPU accelerating graphics cards (Komatitsch et al. 2009, 2010a, Komatitsch 2011). Tensor products inside each element may be optimized to reach very high efficiency (Deville et al. 2002), and mesh point and element numbering may be optimized to reduce processor cache misses and improve cache reuse (Komatitsch et al. 2008). The SEM can handle triangular (in 2-D) or tetrahedral (in 3-D) elements (Wingate & Boyd 1996; Taylor & Wingate 2000; Komatitsch et al. 2001; Cohen 2002; Mercaret et al. 2006), as well as mixed meshes, although with increased cost and reduced accuracy in these non-tensorized elements, as in the discontinuous Galerkin method.

In many cases of practical seismological interest, using a conforming mesh and a continuous formulation are sufficient, because in most geological models material property contrasts are not too dramatic. When this ceases to be true, requiring a discontinuous formulation, one can either turn to a discontinuous version of the SEM (Bernardi et al. 1994; Chaljub 2000; Kopriva et al. 2002; Chaljub et al. 2003; Legay et al. 2005; Kopriva 2006; Wilcox et al. 2010; Acosta Minolia & Kopriva 2011) or to a discontinuous Galerkin technique. A discontinuous formulation is particularly suitable for dynamic rupture simulations, because high frequencies or super-shear rupture need to be accommodated near the fault, where a significantly denser mesh and a more sophisticated (upwind) time scheme are required, thereby suppressing the amplification of unstable modes (see e.g. Benjema et al. 2007, 2009; de la Puente et al. 2009; Tago et al. 2010). Another example that may require a discontinuous formulation involves the resolution of a shallow geotectonic layer, in which seismic shear wave speeds may be reduced by an order of magnitude.

For seismological applications, the SEM has been successfully implemented for 3-D global- and regional-scale simulations (Komatitsch & Vilotte 1998; Paolucci et al. 1999; Chaljub 2000; Komatitsch & Tromp 2002a,b; Capdeville et al. 2003; Chaljub & Valette 2004; Fichtner et al. 2009a), as well as local-scale simulations in complex and/or densely populated regions, for example in southern California, USA (Komatitsch et al. 2004; Tape et al. 2009, 2010), Taipei, Taiwan (Lee et al. 2008, 2009a,b), Caracas, Venezuela (Delavaud et al. 2006) and Grenoble, France (Chaljub et al. 2005; Stupazzini et al. 2009; Chaljub et al. 2010). The SEM may also be used to study elastic wave propagation on smaller scales, for instance the propagation of ultrasonic waves in crystals (van Wijk et al. 2004).

Two complementary SEM software packages—namely, SPECFEM3D_GLOBE for global and regional simulations, and SPECFEM3D for local simulations—are feature-rich, well benchmarked and documented implementations. Data parallelism in the SEM is efficiently exploited using the Message-Passing Interface (MPI) standard, crucial for modern high-performance computing. These open source packages are freely available via the Computational Infrastructure for Geodynamics (CIG) and widely used by the seismological community.

To extend the range of local-scale applications, easing the task of mesh generation is paramount. The two community software packages separate a simulation into two distinct steps: first, creation of a hexahedral mesh, and second, solution of the seismic wave equation. This separation avoids the overhead of remeshing when running multiple simulations for the same region, for example, repeated simulations at the same resolution. Focussing on local-scale simulations, previous versions of SPECFEM3D used an internal mesh which was explicitly tied to the specific purposes of the package: all geological models were based on a layercake model. Consequently, the solver was restricted by its internal mesh. It was impossible to run spectral-element simulations on more complex 3-D models without significant recoding, nor was it possible to run such simulations in regions of interest for on- and off-shore exploration seismology, because acoustic wave propagation in fluids was not supported by the package.

The purpose of this paper is to present forward and adjoint simulations in various 3-D models using the new software package, SPECFEM3D Version 2.0 ‘Sesame’, thereby illustrating its current capabilities. The original SPECFEM3D package for local simulations was extended, improved and optimized in various ways. The Version 2.0 ‘Sesame’ release includes a more flexible internal mesher and accommodates more powerful external meshers, such as CUBIT (Blacker et al. 1994; White et al. 1995; Mitchell 1996; Casarotti et al. 2008). Adding such external meshers into the workflow greatly increases flexibility for high-performance applications, as illustrated by the GeoELSE software package (Casadei & Gabellini 1997; Stupazzini et al. 2009; Chaljub et al. 2010). Advantages of GeoELSE include the accommodation of viscoplastic and non-linear rheologies, whereas benefits of SPECFEM3D include coupled fluid-solid domains and adjoint capabilities; the latter enable one to address seismological inverse problems. Load balancing parallel simulations in SPECFEM3D is accomplished based on the graph partitioning software package SCOTCH (Pellegrini & Roman 1996; Chevalier & Pellegrini 2008). The new package facilitates coupled forward and adjoint acoustic/(an)elastic simulations, which are especially interesting for problems in exploration seismology, ocean acoustics and medical tomography. The new software is freely available under the GNU GPL Version 2 license via CIG.

2 GOVERNING EQUATIONS

Let us briefly summarize the equations governing seismic wave propagation implemented in SPECFEM3D. For more technical details, the reader is referred to Komatitsch & Tromp (1999). SPECFEM3D Version 2.0 ‘Sesame’ implements wave propagation in coupled (an)elastic and acoustic materials on local scales. We may thus safely neglect additional effects that would arise from self-gravitation and rotation (Komatitsch & Tromp 2002b, 2005; Chaljub et al. 2007), which are important at longer periods. In the following, we first discuss (an)elastic wave propagation and subsequently consider acoustic waves.

© 2011 The Authors, GJI, 186, 721–739
Geophysical Journal International © 2011 RAS
2.1 Elastic domain

For elastic materials, the displacement wavefield \( s(x, t) \) is governed by

\[
\rho \frac{\partial^2 s}{\partial t^2} = \nabla \cdot \mathbf{T} + \mathbf{f},
\]

(1)

where \( \rho \) denotes mass density, \( \mathbf{T} \) the stress tensor and \( \mathbf{f} \) the seismic source. On free surfaces, the traction vector must vanish, that is,

\[
\mathbf{n} \cdot \mathbf{T} = 0,
\]

(2)

where \( \mathbf{n} \) denotes the unit outward normal on the surface. On boundaries between different elastic materials, both traction \( \mathbf{n} \cdot \mathbf{T} \) and displacement \( \mathbf{s} \) need to be continuous. On boundaries between elastic and acoustic domains, traction \( \mathbf{n} \cdot \mathbf{T} \) and the normal component of displacement \( \mathbf{n} \cdot \mathbf{s} \) need to be continuous. The initial conditions are

\[
\mathbf{s}(x, 0) = 0, \quad \frac{\partial \mathbf{s}}{\partial t}(x, 0) = 0.
\]

(3)

We thus initiate the simulation in a medium at rest. To accommodate simulations, which may be captured by prescribing a moment-density function. The software also accommodates kinematic rupture simulations, for example, for risk mitigation (Xu et al. 2010). In an anelastic medium, we approximate an absorption-band solid: 2.1 Elastic domain

\[
\frac{\partial^2 \phi}{\partial t^2} = \nabla \cdot \mathbf{H}(t) \mathbf{s} + f.
\]

(4)

We define a scalar potential \( \phi \) such that the displacement \( \mathbf{s} \) may be written as

\[
\mathbf{s} = \rho^{-1} \nabla \phi.
\]

(5)

The equation of motion in terms of the potential \( \phi \) becomes

\[
\kappa^{-1} \frac{\partial^2 \phi}{\partial t^2} = \nabla \cdot (\rho^{-1} \nabla \phi) + f,
\]

(6)

where \( \kappa \) denotes the bulk modulus. It follows that velocity \( \mathbf{v} \) and pressure \( p \) may be expressed as

\[
\mathbf{v} = \rho^{-1} \nabla \phi \quad \text{and} \quad p = -\kappa (\nabla \cdot \mathbf{s}) = -\nabla^2 \phi.
\]

(7)

The resulting formulation for pressure \( p \) is the reason why we choose to define the potential \( \phi \) as in eq. (8). Since pressure is continuous across first-order discontinuities, it follows that \( \nabla^2 \phi \) and thus \( \phi \) must be continuous, a requirement which is honoured automatically by the basis functions of the SEM. The source \( f \) may be expressed in terms of pressure \( P \) acting at location \( x_s \).

\[
f = -\kappa^{-1} P(t) \delta(x - x_s).
\]

(8)

Note that the source is multiplied by a factor \( \kappa^{-1} \) due to the formulation used in eq. (9). Using Gauss’ theorem and a scalar test function \( w \), the weak form becomes

\[
\int_{\Omega} \kappa^{-1} w \frac{\partial^2 \phi}{\partial t^2} \, dx = \int_{\Omega} \rho^{-1} w \cdot \nabla \phi \, dx + \int_{\Omega} \nabla w : \mathbf{H}(t) \mathbf{s} \, dx - \int_{\Omega} \rho^{-1} \nabla w \cdot \nabla \phi \, dx - \kappa^{-1} P(t) w(x_s).
\]

(9)
At the free surface $\partial \Omega$ we set the pressure $p = -\partial_t^2 \phi = 0$, thereby enforcing $\phi = 0$, $\partial_t \phi = 0$ and $\partial_t^2 \phi = 0$, that is, we implement a Dirichlet boundary condition along the surface. At a fluid–solid boundary, the interface coupling integral may be used to exchange the normal component of displacement between fluid and solid: 

$$\rho^{-1} \mathbf{n} \cdot \nabla \phi = \mathbf{n} \cdot \mathbf{s}_{\text{solid}}.$$ 

3 MESHING, MESH PARTITIONING AND LOAD BALANCING

The first step in a SEM consists of constructing a high-quality mesh for the region of interest. In this section, we outline the key issues based on various 3-D examples. Fig. 1 draws the schematic workflow from meshing and partitioning to finally running spectral-element simulations. We discuss each phase separately, focussing on the use of an external mesher, in our case CUBIT (Blacker et al. 1994).

3.1 Hexahedral meshing

We subdivide the model volume $\Omega$ into a set of non-overlapping, hexahedral elements. We impose that the discretization creates a conforming mesh, that is, elements match on a full face or edge, and the mesh cannot be discontinuous. Using the SEM with hexahedral elements leads to computational benefits over tetrahedral finite elements (Komatitsch et al. 2001; Mercier et al. 2006; Vos et al. 2010). Especially for parallel implementations, taking advantage of the diagonal mass matrix and optimized tensor products is critical in terms of computational speed (Komatitsch et al. 2003; Carrington et al. 2008; Vos et al. 2010). Hexahedral meshing is also attractive for the SEM because it benefits from reduced errors and generally smaller element counts compared to tetrahedral meshing (Hesthaven & Teng 2000; Komatitsch et al. 2001; Vos et al. 2010).

Unfortunately, automatic 3-D hexahedral mesh generation is more demanding than unstructured tetrahedral meshing (Shepherd & Johnson 2008; Staten et al. 2010). To construct hexahedral meshes, our examples make use of an external hexahedral mesher, such as CUBIT (Blacker et al. 1994). We focus on this particular mesh generation tool kit because it is a well-documented and feature-rich package, on which most of our own experience is based. One may readily use other meshing tools, such as Abaqus (SIMULIA 2008), ANSYS (ANSYS 2011), GOCAD (Mallet 1992; Caumon et al. 2009), GID (Gardia-Donoro et al. 2010; Ribi et al. 2011), Gmsh (Geuzaine & Remacle 2009), TrueGrid (Noble & Nuss 2004; Rainsberger 2006) or Salome (Ribes & Caremoli 2007; Bergeaud et al. 2010).

Fig. 2 shows several examples of fully unstructured hexahedral meshes. In the Mount St Helens region, the mesh employs a mesh tripling layer to increase resolution at the topographic surface. Tripling is the default refinement in CUBIT for subdividing hexahedral elements in a conforming fashion. Surface topography is imported using Shuttle Radar Topographic Mission (SRTM) data, converted to Universal Transverse Mercator (UTM) coordinates with an original resolution of 90 m (Jarvis et al. 2008). Meshing is performed automatically by CUBIT using a sweep algorithm. The resolution of the mesh enables seismic wave simulations with frequencies up to $\sim 1.5$ Hz. The mesh for the L’Aquila region, Italy, consists of $\sim 7$ M hexahedra with an element size of $\sim 90$ m at the top surface. This mesh facilitates simulations of seismic wave propagation up to $\sim 5$ Hz. For the exploration geophysics model, the hexahedral mesh honours a salt dome body inside a 3-D model capped by a water layer. The mesh for asteroid 433-Eros with a close-bound surface has a resolution of roughly 300 m. Finally, the filled coffee cup model discretized into hexahedra couples an elastic domain for the cup with an acoustic domain for the coffee inside the cup.

To ensure compatibility with previous versions of SPECFEM3D (see e.g. Komatitsch et al. 2004; Liu et al. 2004), the in-house mesher based on analytical linear interpolation from the top to the bottom of the mesh has been adapted to the new code structure. It facilitates the design of simpler, alternative meshes for layercake models.

3.2 Partitioning and load balancing

Balancing the computational load and distributing the mesh on a large number of cores is crucial for optimized high-performance simulations (Martin et al. 2008a). To do so, we make use of an external partitioner, namely SCOTCH (Pellegrini & Roman 1996; Chevalier & Pellegrini 2008), which we use to balance spectral-element computations on an arbitrary number of cores. An alternative partitioner able to fulfill these tasks is METIS (Karypis & Kumar 1998), but SCOTCH is more actively maintained (Chevalier & Pellegrini 2008) and performs better in many cases that we have tested.

Especially for simulations involving coupled elastic and acoustic domains, balancing the mesh becomes paramount. Most of the computation time is spent resolving the divergence of the stress tensor in each element. The computational cost for an elastic element is approximately four times larger than for an acoustic element, which may be established by running simulations for one domain at a time. During partitioning, we therefore weight each element according to its associated domain type and computational cost to balance the overall numerical cost rather than simply the number of elements between partitions. The major improvement in SPECFEM3D code performance focuses on these tensor products, using highly efficient algorithms developed by Deville et al. (2002) and optimizing cache usage. Another key aspect of mesh partitioning is minimization of the number of edge cuts, because this reduces the amount of MPI communications between processor cores (an edge cut occurs when two contiguous elements are assigned to distinct cores). On machines comprising a very large number of cores, it is crucial to resort to non-blocking communications between compute nodes, for instance using non-blocking MPI message passing, to obtain good performance scaling (Danielson & Namburu 1998; Komatitsch et al. 2008; Martin et al. 2008a).
Figure 2. Mesh examples: (a) Mount St Helens meshed by hexahedral elements. The mesh honours surface topography and includes a mesh tripling layer in the middle of the model. The smallest element size is approximately 280 m. (b) L’Aquila, Italy, region discretized for high-frequency simulations. The mesh honours surface and Moho topography and includes two mesh tripling layers. The yellow and blue volumes denote slower and faster than average wave speeds, respectively. (c) Salt dome body meshed inside an exploration model for a SEG/EAGE benchmark test. (d) 3-D hexahedral mesh of the asteroid 433 Eros. (e) Arbitrarily shaped mesh for coupled solid-fluid simulations involving a coffee cup.

Fig. 3 presents a simple example of partitioning and load balancing the mesh around Mount St Helens, as shown in Fig. 2. For illustrative purposes, we decompose the mesh onto four cores using the SCOTCH library. The total number of spectral-elements is ~24,000, such that each partition contains ~6,000 elements after decomposition. Partitioning and load balancing equally distributes the elements over the different cores, since the whole domain is purely elastic. A partitioner such as SCOTCH can also load balance computationally more complex meshes, for example, containing PML elements along absorbing boundaries of the model; this is the subject of future research.

In a final, separate step we generate mesh databases for each partition needed for the spectral-element solver. These databases contain Gauss–Lobatto–Legendre (GLL) points for all spectral elements. Material properties are assigned to these GLL points, and thus sampling resolution of a geological model not only depends on element size but also on polynomial degree. Furthermore, the generation of mesh databases automatically detects interfaces between elastic and acoustic domains, needed for coupling seismic waves from one domain to another. Load-balancing of the simulation persists, because we keep the polynomial degree fixed for all spectral elements. Note that this final step of generating mesh...
Figure 3. Mount St Helens mesh partitioned and load balanced to run in parallel on four cores. The four partitions are indicated by different colours.

Figure 4. Outer (highlighted) and inner (transparent colours) elements for the mesh shown in Fig. 3. Outer elements have at least one point in common with an element from another slice and must therefore be computed first, before initiating non-blocking MPI communications. While MPI messages are travelling across the computer network, simultaneous computations are performed on inner elements. Non-blocking MPI communication is crucial to obtain good scaling results for simulations running on a large number of parallel cores.

databases provides additional freedom in assigning and changing wave speeds, which is important for seismic inversion procedures.

3.3 Overlapping computation and communication

The elements that compose the mesh slices shown in Figs 2 and 3 are in contact through a common face, edge or point. To allow for overlap of communication between compute nodes with computations within each mesh slice—thereby speeding up the simulation—a list of all elements in contact with any other mesh slice through a common face, edge or point is created. Members of this list are termed ‘outer’ elements, and all other elements are termed ‘inner’ elements, as illustrated in Fig. 4.

Once the outer elements have been identified following a standard procedure (see e.g. Danielson & Namburu 1998; Martin et al. 2008a; Micikevicius 2009; Michéa & Komatitsch 2010; Komatitsch 2010a, 2011), MPI buffers are filled and a non-blocking MPI call is issued, which initiates communication and returns immediately. While MPI messages are travelling across the network, computations are performed on inner elements. Achieving effective overlap requires that the ratio of the number of inner to outer elements is sufficiently large, a condition which is satisfied for suitably large mesh slices. Under these circumstances, MPI data transfer will generally finish before the completion of computations on inner elements.

4 SAMPLE SIMULATIONS

In this section, we present various simulations with increasing complexity to highlight the flexibility of our new spectral-element package. We start with a layercake model and finish with an example of an arbitrarily shaped model.

4.1 Validation example: two-layer model

The SEM has been well benchmarked against discrete wavenumber methods for layercake models by Komatitsch & Tromp (1999). Here we compare their two-layer model solution (Fig. 8, left-hand side) against the solution obtained by the new code. The model has a horizontal size of 134 km × 134 km, with a depth of 60 km. We discretize the model into 70 200 elements, using an approximate element size of 1000 m at the top and 4500 m at the bottom. A mesh tripling layer is placed below the upper layer, between 3 km and 10 km, with the wave speed properties of the lower layer. We use SCOTCH to partition the model onto six cores, each with 11 700 elements. The final mesh is generated using GLL points for a polynomial degree $N = 5$, which results in 9 025 941 global mesh points.

A vertical force is placed at a depth of 25.05 km in the middle of the model. The source–time function is a Ricker wavelet with a dominant frequency of 0.4 Hz. The simulation uses a time step of 6.5 ms and propagates for 6000 steps. We compare our solution with seismograms obtained by Komatitsch & Tromp (1999, Fig. 9). The mesh and seismograms are shown in Fig. 5. The seismograms match very closely with the reference solutions, exhibiting almost identical displacements. Maximum waveform differences reach ∼0.3 per cent, arising from differences in mesh geometry and source implementation.

The performance of the code is summarized in Fig. 6, using simulations with the optimized routines by Deville et al. (2002)
Figure 5. Validation for a two-layer mesh (left, top panel), using six partitions (left, bottom panel), and seismograms recorded at the surface at horizontal distances of 2.39 km (right, top panel) and 31.11 km (right, bottom panel). Plotted are radial displacements (SEM, red) against reference solutions (REF, black) from Komatitsch & Tromp (1999), as well as their exaggerated differences (blue).

Figure 6. CPU scaling results for the model shown in Fig. 5, (a) using a fixed total problem size (strong scaling) and (b) a fixed problem size per processor (weak scaling) for up to 256 cores. Perfect weak scaling deviates slightly from a straight line, because a larger number of processors involves more MPI buffers and a polynomial degree $N = 4$. We are interested in how the code behaves when the number of calculations is decreased linearly with the number of CPU cores (strong scaling), and how performance varies when the number of calculations on each core is kept constant while increasing the total number of CPU cores (weak scaling). To assess strong scaling, we fix the total mesh size but vary the number of CPU cores used for the simulation. We run the simulation for a duration of 4000 time steps and show the corresponding average elapsed time per time step in Fig. 6(a). More interesting for high-performance applications, we assess weak scaling by fixing the problem size per processor and varying the number of CPU cores. This leads to higher mesh resolutions for an increasing number of CPU cores but should keep the average elapsed time per time step constant. We summarize the simulation times in Fig. 6(b). The computations were performed on a high-performance cluster with compute nodes consisting of two Intel Nehalem quad-core processors; each core has 3 GB of RAM. The code scales linearly within $\sim 90$ per cent up to 256 CPU cores for both strong and weak scaling, achieving excellent performance on this parallel system. Note that for the strong scaling examples shown here, simulations using more than 64 CPUs see a performance decrease since communications no longer overlap, thus they no longer profit from the default non-blocking MPI scheme (Martin et al. 2008a).
4.2 Mount St Helens example: layercake model with surface topography

To include surface topography, we import SRTM data with an original resolution of 90 m (Jarvis et al. 2008) and convert it to UTM coordinates for the corresponding UTM zone. We read in this data set using CUBIT and create a surface honouring these data points. A 3-D volume is built manually with topography on top.

The simulation uses an explosive source at a depth of 5 km. In Fig. 7, we show the vertical displacement field at the free surface at consecutive times. Note that once the wavefield hits the model boundary, it gets absorbed by the Clayton–Engquist–Stacey absorbing boundary conditions.

4.3 L’Aquila example: layercake model honouring surface and Moho topography

The purpose of this example is to show that additional surfaces may be honoured by the mesh, for example the Moho. We import not only surface topography, but also create a Moho surface that is honoured by the boundaries of the spectral elements. The mesh for the L’Aquila region was built using an additional ‘Python’ library that semi-automates the mesh creation process with CUBIT (Casarotti et al. 2008). Once these mesh files are constructed, the default partitioning and database generation process may be used to create fully load-balanced spectral-element simulations on an arbitrary number of parallel processors.

Fig. 8 shows several snapshots of the seismic wavefield at consecutive times for an anelastic material, using a kinematic source description for the 2009 April 6, L’Aquila earthquake. Simulations are accurate up to 5 Hz and may be used to discriminate between different wave speed models and/or kinematic source solutions. These high-frequency simulations may be used to assess the response of engineered structures and may guide the development of better seismic building codes for the L’Aquila region.

4.4 SEG/EAGE salt dome example: exploration model

Our new spectral-element package can combine acoustic and (an)elastic simulations by coupling these distinct domains. In this
example, we generate acoustic waves in the top water layer and propagate them down through a salt dome body included in the lower, anelastic domain. The mesh honours the surface of the salt dome and the fluid–solid boundary, that is, the bathymetry.

Fig. 9 shows the acoustic wavefield at the free surface of the water layer at different times. The source is a pressure source, located slightly below the free surface in the water layer, with a Ricker source–time function. The wavefield is reflected and refracted by the salt dome in the anelastic domain below the water layer. Note how these reflected/refracted waves, which include $P$-to-$S$ converted waves, are recorded in the water layer.

4.5 Asteroid example: arbitrarily shaped model

This final example shows that our new software package may be used for simulating wave propagation in arbitrarily shaped models, such as asteroid Eros, which was imaged by the NEAR spacecraft in 2000–2001. This silicated asteroid is 34 km long with a peanut-like shape and is thought to be covered with a regolith layer, corresponding to a blanket of loose material crushed by impacts (Richardson et al. 2005). We meshed the asteroid with 5,797,440 hexahedral elements having an approximate resolution of 70 m. To simulate a thin, 70 m regolith layer superimposed on strong bedrock, as suggested by Robinson et al. (2002), we assigned a low wave speed material to the elements touching the free surface and a high wave speed material to elements inside the asteroid, representing solid bedrock.

We simulated the propagation of seismic waves from a source represented by a point force normal to the surface. The source–time function corresponds to a Dirac pulse low-pass filtered up to a cut-off frequency of 5 Hz. Fig. 10 displays wavefield snapshots for the first $\sim 10$ s of the simulation. It shows the propagation of $P$, $S$ and surface waves with a refocusing effect on the opposite side. The regolith layer strongly increases physical dispersion of surface waves. Peak ground accelerations are plotted in Fig. 11 for a simulation without a regolith layer, showing that refocusing occurs on the asteroid.

5 ADJOINT SENSITIVITY KERNELS

An important goal in seismology is to use differences between observed and simulated seismograms to improve Earth and source models, that is, we are interested in the inverse problem. An elegant way to address this issue is to take advantage of adjoint methods (Tarantola 1984; Tromp et al. 2005) to calculate Fréchet derivatives for a pre-defined objective function. These derivatives may then be used in a conjugate-gradient approach to minimize differences between data and synthetics. The key ingredients of such an adjoint approach are sensitivity kernels. Following Tromp et al. (2005), Liu & Tromp (2006, 2008) and Tromp et al. (2010), we show examples of sensitivity kernels for various models using our new software package.

5.1 Elastic sensitivity kernels

Following Tromp et al. (2005), we may write the variation of a misfit function $\chi$ as

$$\delta \chi = \int_V \left( K_\rho \delta \ln \rho + K_{ijklm} \delta c_{ijklm} \right) \, d^3 x,$$

(14)

where $\delta \ln \rho = \delta \rho / \rho$ denotes relative perturbations in density and $\delta c_{ijklm}$ denotes perturbations in the elastic tensor. The misfit kernels are given by

$$K_\rho = -\rho \int_0^T s^\top(T - t) \cdot \hat{\varepsilon}(t) \, dt.$$
Figure 10. Wavefield snapshots for an asteroid simulation taken after 3 s, 4.5 s, 6.5 s and 10.5 s. Plotted is the norm of the velocity at the free surface of the asteroid.

Figure 11. ShakeMap views for an asteroid simulation. Plotted are different views of the peak ground accelerations at the free surface of the asteroid.

\[ K_{\text{c}_{jklm}} = -\int_0^T \epsilon_{\text{l}_m}^j(T-t) \epsilon_{\text{l}_m}(t) \, dt, \quad (16) \]

where \( \epsilon_{\text{l}_m} \) and \( \epsilon_{\text{l}_m}^j \) denote elements of the strain and adjoint strain tensors, and where we have suppressed the spatial dependence to avoid clutter.

In an isotropic model, we have \( c_{jklm} = (\kappa - 2\mu/3) \delta_{ij} \delta_{lm} + \mu (\delta_{im} \delta_{jl} + \delta_{jm} \delta_{il}) \), and thus eq. (14) may be rewritten as

\[ \delta \chi = \int_V \left( K_\rho \delta \ln \rho + K_\mu \delta \ln \mu + K_\kappa \delta \ln \kappa \right) d^3 x. \quad (17) \]

The isotropic misfit kernels \( K_\rho, K_\mu \) and \( K_\kappa \) are defined by

\[ K_\rho = -2\mu \int_0^T \mathbf{D}^i(T-t) : \mathbf{D}(t) \, dt, \quad (18) \]

\[ K_\mu = -\kappa \int_0^T \left[ \nabla \cdot s^i(T-t) \right] \left[ \nabla \cdot s(t) \right] \, dt, \quad (19) \]

where \( \mathbf{D} = \frac{1}{2} \left[ \nabla s + (\nabla s)^T \right] - \frac{1}{2} \left( \nabla \cdot s \right) \mathbf{I} \) and \( \mathbf{D}^i = \frac{1}{2} \left[ \nabla s^i + (\nabla s^i)^T \right] - \frac{1}{2} \left( \nabla \cdot s^i \right) \mathbf{I} \) are the traceless strain deviator and its adjoint, respectively. In terms of a parametrization involving compressional wave speed \( \alpha \), shear wave speed \( \beta \) and density \( \rho \), the corresponding kernels are given by a linear combination of these primary kernels (Tromp et al. 2005, eq. 20):

\[ K_\alpha = 2 \left( \frac{\kappa + 4\mu}{3\kappa} \right) K_\kappa, \]

\[ K_\beta = 2 \left( \frac{\kappa}{\beta} - \frac{4\mu}{3\kappa} K_\kappa \right), \quad K'_\rho = K_\alpha + K_\mu + K_\rho. \quad (20) \]

Note that a suitable parametrization for isotropic inversions is to use bulk sound wave speed \( \Phi_1 = \sqrt{\kappa/\rho} \), shear wave speed \( \beta \) and density \( \rho \) (Tarantola 1987). Bulk sound and shear wave speeds are independent combinations of the bulk and shear moduli \( \kappa \) and \( \mu \). The corresponding kernels are given by

\[ K_0 = 2K_\kappa, \quad K'_\beta = 2K_\mu, \quad K'_\rho = K_\alpha + K_\mu + K_\rho. \quad (21) \]

We place an explosive source at a depth of 7 km and a horizontal distance of 16 km from the receiver in a homogeneous model with topography around Mount St Helens. The \( P \) wave at the receiver is used to construct a traveltime adjoint source for the kernel simulation. Fig. 12(a) shows the isotropic kernels \( K_\kappa, K_\mu \) and \( K_\rho \), and Fig. 12(b) the isotropic kernels \( K_\alpha, K_\beta \) and \( K'_\rho \) for the same model and source–receiver geometry.

Note that although we construct the adjoint source using the \( P \) wave, significant non-zero \( S \)-wave sensitivity is visible in the \( K_\beta \) and \( K_\mu \) kernels. We interpret these areas of high sensitivity as...
Traveltime sensitivity to elastic structure. Fréchet derivatives for isotropic parametrizations (a) $K_\kappa$, $K_\mu$ & $K_\rho$ and (b) $K_\alpha$, $K_\beta$ & $K_\rho'$ are compared in a model of Mount St Helens using traveltime adjoint sources for the $P$ wave. Shown are vertical cross-sections through the source–receiver line and perpendicular to this line.

$P$-to-$S$ scattering locations, which affect the signal within the chosen time window. As may be observed in Fig. 12, such scattering sensitivity is especially strong at the free surface close to the receiver.

5.2 Acoustic sensitivity kernels

Liu & Tromp (2008) calculated global sensitivity kernels, which include sensitivity to the liquid outer core. In this section, we present acoustic sensitivity kernels for general local- or regional-scale models. Such kernels may be used, for example, in ocean acoustics, non-destructive testing and medical tomography.

For acoustic simulations, the kernels are given by

$$K_\rho = \rho^{-1} \int_0^T \left[ \nabla \partial_\tau \phi^\dagger(T - t) \right] \cdot \left[ \nabla \partial_\tau \phi(t) \right] \, dt,$$  \hspace{1cm} (22)

$$K_\kappa = -\kappa^{-1} \int_0^T \partial_\tau^2 \phi^\dagger(T - t) \partial_\tau^2 \phi(t) \, dt,$$  \hspace{1cm} (23)

where $\phi$ and $\phi^\dagger$ denote the acoustic scalar potential and adjoint potential, respectively. To illustrate these kernels, we use a model with acoustic and elastic regions. The model combines a water layer on top of a homogeneous elastic layer, separated by a bathymetric surface. The dimensions of the model volume are approximately $2\,km \times 2\,km$ horizontally and $1\,km$ in depth. Bathymetry is taken from a location in front of Pearl Harbor (Hawaii, USA), with a resolution of $\sim 11\,m$. For the forward simulation, we use a pressure source in the form of an explosion with a Gaussian source–time function, and record pressure variations at the receiver. Both source and receiver are in the acoustic domain at a depth of $10\,m$ and $\sim 1.1\,km$ apart from each other. We use the simulated pressure...
variation within the measurement window as the pressure misfit for
the adjoint source, as explained in Section A1 of the Appendix.

Fig. 13 shows the corresponding combined acoustic and elastic
kernels. The kernels highlight how the pressure waveform in the
chosen measurement window is affected by a head wave (a Scholte
wave) travelling along the seafloor. Since the acoustic region does
not support shear waves, the kernels $K_\mu$ and $K_\beta$ are zero in this
upper domain. However, they do exhibit non-zero sensitivity in the
elastic domain, due to $P$-to-$S$ coupling along the seafloor.

5.3 Noise sensitivity kernels

As demonstrated by Tromp et al. (2010), noise cross-correlation
sensitivity kernels may also be calculated based on an adjoint
method, and the new package has the necessary capabilities to
perform such calculations. Consider two receivers located at $x^a$
and $x^\beta$. In seismic interferometry, ensemble sensitivity kernels are
calculated in terms of interactions between an ensemble forward
wavefield $\Phi^a$, generated at location $x^a$, and an ensemble adjoint
wavefield $\Phi^a$†, generated at $x^\beta$ and triggered by the differences
between simulated and observed ensemble-averaged cross correla-
tions at $x^a$ and $x^\beta$. The isotropic ensemble sensitivity kernels are
given by

$$\langle K_\rho \rangle = -\int \rho \left[ \Phi^a_\mu(-t) \cdot \Phi^a(t) + \Phi^a_\beta(-t) \cdot \Phi^\beta(t) \right] dt,$$

(24)

$$\langle K_\mu \rangle = -\int 2\mu \left[ D^a_\mu(-t) \cdot D^a(t) + D^a_\beta(-t) \cdot D^\beta(t) \right] dt,$$

(25)

$$\langle K_\kappa \rangle = -\int \kappa \left[ \nabla \cdot \Phi^a_\mu(-t) \cdot \Phi^a(t) \right. \nabla \cdot \Phi^\beta(t) \right] dt,$$

(26)
Figure 14. Noise cross-correlation sensitivity to elastic structure. Shown are (a) first, (b) second and (c) summed contributions to the \( \langle K_\beta \rangle \) Fréchet derivative in a homogeneous isotropic model. Plotted are vertical and horizontal cross-sections through the line connecting the two receivers (white dots) and perpendicular to this line.

\[
D^{\alpha} = \frac{1}{2} \left[ \nabla \Phi^{\alpha} + (\nabla \Phi^{\alpha})^T \right] - \frac{1}{3} (\nabla \cdot \Phi^{\alpha}) I, \quad (27)
\]

\[
D^{i \alpha \beta} = \frac{1}{2} \left[ \nabla \Phi^{i \alpha \beta} + (\nabla \Phi^{i \alpha \beta})^T \right] - \frac{1}{3} (\nabla \cdot \Phi^{i \alpha \beta}) I, \quad (28)
\]

denote the traceless ensemble strain deviator and corresponding adjoint.

Fig. 14 shows the isotropic kernel \( \langle K_\beta \rangle \) calculated according to eq. (20) using the primary isotropic ensemble sensitivity kernels given. Plotted are the two contributions from ensemble wavefields generated at the first receiver location, \( x^\alpha \), and the second receiver location, \( x^\beta \), as well as the combined ensemble sensitivity kernel \( \langle K_\beta \rangle \), which is the sum of the two contributions. The two receivers are placed at a distance of 65 km from each other on top of a homogeneous block model with dimensions of 134 km \( \times \) 134 km horizontally and 60 km in depth. We smooth the kernel contributions using a 3-D Gaussian filter with a standard deviation of 2 km in the horizontal and vertical directions. Note that these noise sensitivity kernels exhibit strong 3-D variability. Depth sensitivity is controlled by the period range (5–100 s in this example).

6 CONCLUSIONS AND FUTURE WORK

We have taken advantage of recent advances in high-performance computing, fully unstructured hexahedral meshing, load balancing and mesh partitioning to facilitate forward and adjoint simulations of seismic wave propagation in coupled fluid and solid domains. Our new open source software package, SPECFEM3D Version 2.0 ‘Sesame’, performs acoustic and (an)elastic simulations of seismic wave propagation in complex geological models. Hexahedral meshes may be generated based on packages such as CUBIT, Abaqus, ANSYS, GOCAD, GiD, Gmsh, TrueGrid or Salome, but the simple in-house mesher used in previous versions of SPECFEM3D remains available for backcompatibility.

Partitioning and load balancing meshes may be accomplished based on graph partitioning software, such as SCOTCH. By coupling acoustic and (an)elastic wave propagation, we are able to calculate related sensitivity kernels, which are useful for waveform inversions in off-shore exploration seismology, ocean acoustics, non-destructive testing and medical tomography. Attenuation is important on all scales of seismic wave propagation and is accommodated based on a series of standard linear solids. In particular for simulations in medical tomography, strong attenuation and related dispersion play a dominant role. Finally, the new package can be used to calculate finite-frequency noise cross-correlation sensitivity kernels, which may be used for seismic interferometry. In future work, we will add C-PML and GPU support to the package. Viscoplastic and non-linear elastic rheologies (e.g. Xu et al. 2003; di Prisco et al. 2007) are accommodated by the GeoELSE software package (Stupazzini et al. 2009; Chaljub et al. 2010), and we will consider such non-linear constitutive relationships in future releases of SPECFEM3D.

The next grand challenge involves the development of 3-D seismic imaging and inversion tools for the characterization of earthquakes, Earth ‘noise’ and mapping of the Earth’s interior on all scales, that is, to address the seismological inverse problem. The goal is to harness the power of forward and adjoint modelling tools, such as SPECFEM3D, together with modern computers to enhance the quality of images of the Earth’s interior and the earthquake rupture process. Most traditional tomographic methods utilize traveltime or phase information measured by comparing data with simulations, and interpret such measurements based on ray theory or other approximate methods. Because of the limitations of these approximate theories, only parts of seismograms can be used, and initial models are generally restricted to be spherically symmetric. With the new generation of modelling tools we can go beyond
classical tomography by using fully 3-D initial models (e.g. Akçelik et al. 2002, 2003; Askan et al. 2007; Chen et al. 2007; Fichtner et al. 2009b; Fichtner 2010), and utilizing as much information contained in seismograms as possible (e.g. Maggi et al. 2009; Valentine & Woodhouse 2010). Our approach will be to minimize frequency-dependent phase and amplitude differences between simulated and observed seismograms based on adjoint techniques in combination with conjugate gradient methods, an approach we refer to as ‘adjoint tomography’ (Tape et al. 2009, 2010). The development of such capabilities will affect the fields of exploration geophysics, regional & global seismology, ocean acoustics, non-destructive testing, medical tomography and helioseismology.

ACKNOWLEDGMENTS
We thank two anonymous reviewers for comments which improved the manuscript. All simulations were performed on a Dell cluster built and maintained by the Princeton Institute for Computational Science & Engineering (USA). The open source spectral-element software package SPECFEM3D Version 2.0 ‘Sesame’ used for this science & engineering (USA). The open source spectral-element built and maintained by the Princeton Institute for Computational all simulations were performed on a Dell cluster and the manuscript. All simulations were performed on a Dell cluster built and maintained by the Princeton Institute for Computational Science & Engineering (USA). The open source spectral-element software package SPECFEM3D Version 2.0 ‘Sesame’ used for this science & engineering (USA). The open source spectral-element built and maintained by the Princeton Institute for Computational

REFERENCES

© 2011 The Authors, GJI, 186, 721–739
Geophysical Journal International © 2011 RAS


Mickevicius, P. 2009. 3D finite-difference computation on GPUs using CUDA, in GPGPU-2: Proceedings of the 2nd Workshop on General Purpose Processing on Graphics Processing Units, pp. 79–84, Washington, DC.


APPENDIX A: ACOUSTIC ADJOINT EQUATIONS

A1 Pressure waveform misfit kernels

For acoustic tomographic studies, it is convenient to define a pressure misfit function

\[ \chi = \frac{1}{2} \sum_i \int |p_i^{\text{syn}}(m) - p_i^{\text{obs}}|^2 \, dx, \]  

(A1)

where \( p_i^{\text{syn}} \) is the numerically computed pressure and \( p_i^{\text{obs}} \) the observed pressure at location \( x_i \). The variation in pressure may be written in terms of the variation in the potential \( \phi \) as

\[ \delta p = -\hat{\phi} \delta \phi, \]  

(A2)

which follows from the definition of the scalar potential \( \phi \) in eq. (11). The corresponding action in the acoustic case is given by

\[ \chi = \frac{1}{2} \sum_i \int |p_i^{\text{syn}} - p_i^{\text{obs}}|^2 \, dx - \int \int_{\Omega} \lambda \left[ k^{-1} \hat{\phi} \nabla \cdot \rho^{-1} \nabla \phi - f \right] \, d^3x, \]  

(A3)

where \( \lambda \) denotes a scalar Lagrange multiplier. Setting \( \Delta p_i = p_i^{\text{syn}} - p_i^{\text{obs}} \) and taking the variation of the action, we obtain

\[ \delta \chi = \sum_i \int \Delta p_i \, \delta p_i \, dx - \int \int_{\Omega} \left[ k^{-1} \hat{\phi} \nabla \cdot \rho^{-1} \nabla \phi - \nabla \cdot \left( \rho^{-1} \nabla \phi \right) - \lambda \delta f \right] \, d^3x \]  

\[ - \int \int_{\Omega} \left[ k^{-1} \hat{\phi} \nabla \cdot \rho^{-1} \nabla \phi \right] \delta \phi \, d^3x \, dx - \int \int_{\Omega} \hat{n} \cdot \left( \rho^{-1} \nabla \phi \right) \delta \phi \, d^3x \, dx. \]  

(A4)

© 2011 The Authors, GJI, 186, 721–739
Geophysical Journal International © 2011 RAS
Since eq. (A4) must be stationary when no model perturbations are present, that is, \( \delta \rho = 0, \delta \kappa = 0 \) and \( \delta f = 0 \), we obtain

\[
\int \int_{\Omega} \left[ \kappa^{-1} \partial_t^2 \lambda - \nabla \cdot (\rho^{-1} \nabla \lambda) \right] \delta \phi \, d^3x \, dt
\]

\[
= \int \int_{\Omega} \sum_i \Delta p_i \, \delta(x - x_i) \, \delta p \, d^3x \, dt
= \int \int_{\Omega} \nabla \cdot (\rho^{-1} \nabla \lambda) \, \delta \phi \, d^3x \, dt
\]

\[
= -\int \int_{\Omega} \sum_i \partial_i^2 \Delta p_i \, \delta(x - x_i) \, \delta \phi \, d^3x \, dt
= -\int \int_{\Omega} \partial_i^2 \nabla \cdot \nabla \lambda \, \delta \phi \, d^3x \, dt.
\] (A5)

where \( x_i \) is the station location of the corresponding \( i \)th measurement. Note that the last integration by parts of the first term is valid under the assumption that \( \Delta p_i \) and \( \partial_i \Delta p_i \) vanish at the limits of the time integration, that is, for a given measurement window \([0, T]\), \( \Delta p_i(x, 0) = \Delta p_i(x, T) = 0 \) and \( \partial_i \Delta p_i(x, 0) = \partial_i \Delta p_i(x, T) = 0 \). This is generally true because we taper the ends of the misfit window.

Let us define the adjoint scalar potential as

\[
\phi^\dagger(x, t) = \lambda(x, T - t).
\] (A6)

It follows from (A5) that \( \phi^\dagger \) must satisfy the adjoint wave equation

\[
\kappa^{-1} \partial_t^2 \phi^\dagger - \nabla \cdot \left( \rho^{-1} \nabla \phi^\dagger \right) = f^\dagger,
\] (A7)

where the adjoint source is given by

\[
f^\dagger(x, t) = -\sum_i \partial_i^2 \Delta p_i (T - t) \, \delta(x - x_i).
\] (A8)

The initial conditions for the adjoint potential must satisfy \( \phi^\dagger(T) = 0 \) and \( \partial_t \phi^\dagger(T) = 0 \). The corresponding fluid–solid boundary conditions involve terms with \( \rho^{-1} \nabla \phi^\dagger \).

For acoustic simulations, there is no shear contribution and we may set \( K_\mu = 0 \). Using

\[
\nabla \cdot s = -\kappa^{-1} p = \kappa^{-1} \partial_t \phi,
\] (A9)

\[
\nabla \cdot s^\dagger = \kappa^{-1} \partial_t^2 \phi^\dagger,
\] (A10)

the kernel \( K_\kappa \) given in eq. (18) becomes

\[
K_\kappa = -\int_0^T \kappa^{-1} \partial_t^2 \phi^\dagger(T - t) \, \partial_t \phi(t) \, dt.
\] (A11)

It is this last kernel expression that is actually implemented, since the values for \( \partial_t^2 \phi \) and \( \partial_t^2 \phi^\dagger \) are obtained at each time step in the Newark time scheme used to propagate acoustic waves.

### A2 Pressure traveltime adjoint sources

Instead of measuring waveform misfits, one may also define a traveltime misfit for pressure signals, that is,

\[
\chi = \frac{1}{2} \sum_i \left| T_{i}^{\text{syn}}(m) - T_{i}^{\text{obs}} \right|^2,
\] (A12)

where \( T_{i}^{\text{syn}}(m) \) denotes the arrival time in the synthetics pressure records computed for model \( m \), and \( T_{i}^{\text{obs}} \) the arrival time of the observed pressure wave. The variation in traveltime \( \delta T \) may be written to first order in terms of perturbations in pressure as \( \delta p \) (Hung et al. 2000)

\[
\delta T = \frac{1}{N} \int \partial_t p \, \delta p \, dt.
\] (A13)

where \( N = \int \partial_t p \, \delta p \, dt \). Using \( \delta p = -\partial_t \delta \phi \), this leads to

\[
\delta T = -\frac{1}{N} \int \partial_t p \, \partial_t^2 \delta \phi \, dt.
\] (A14)

Defining \( \Delta T_i = T_{i}^{\text{syn}}(m) - T_{i}^{\text{obs}} \), the variation of the action becomes

\[
\sum_i \Delta T_i \, \delta T_i = -\frac{1}{N} \Delta T_i \int \partial_t p \, \partial_t^2 \delta \phi \, dt
\] (A15)

\[
= -\int \int_{\Omega} \sum_i \frac{1}{N} \Delta T_i \, \delta(x - x_i) \, \partial_t p \, \partial_t^2 \delta \phi \, d^3x \, dt.
\] (A16)
Under the assumption that $\partial_t p(0) = \partial_t p(T) = 0$ and $\partial^2_t p(0) = \partial^2_t p(T) = 0$ (which can be achieved by carefully selecting and tapering the measurement time windows), we find after some further manipulation that the adjoint source for a traveltime misfit between observed and simulated pressure signals is given by

$$f^\dagger(x, t) = -\sum_i \frac{1}{N} \Delta T_i \partial^3_t p(x, T - t) \delta(x - x_i).$$

(A17)