

Local time stepping with the discontinuous Galerkin method for wave propagation in 3D heterogeneous media

Sara Minisini¹, Elena Zhebel¹, Alexey Kononov², and Wim A. Mulder³

ABSTRACT

Modeling and imaging techniques for geophysics are extremely demanding in terms of computational resources. Seismic data attempt to resolve smaller scales and deeper targets in increasingly more complex geologic settings. Finite elements enable accurate simulation of time-dependent wave propagation in heterogeneous media. They are more costly than finite-difference methods, but this is compensated by their superior accuracy if the finite-element mesh follows the sharp impedance contrasts and by their improved efficiency if the element size scales with wavelength, hence with the local wave velocity. However, 3D complex geologic settings often contain details on a very small scale compared to the dominant wavelength, requiring the mesh to contain elements that are smaller than dictated by the wavelength. Also, limitations of the mesh generation software may produce regions where the elements are much smaller than desired. In both cases, this leads to a reduction of the time step required to solve the wave propagation and significantly increases the computational cost. Local time stepping (LTS) can improve the computational efficiency and speed up the simulation. We evaluated a local formulation of an LTS scheme with second-order accuracy for the discontinuous Galerkin finite-element discretization of the wave equation. We tested the benefits of the scheme by considering a geologic model for a North-Sea-type example.

INTRODUCTION

Efficient numerical methods for the simulation of acoustic, electromagnetic, or seismic wave phenomena are of fundamental

importance for modeling and inversion. Finite-difference methods are widely used because they are relatively easy to implement and computationally efficient. Moreover, the use of explicit time stepping is natural. However, their reliance on Cartesian structured grids makes them less suitable for complex heterogeneous geologic settings. When abrupt changes in the properties of the material arise, the scheme loses its accuracy. This can be amended by using very fine grids at the expense of computational time. A similar limitation is encountered when considering a heterogeneous medium that includes topography such as hills and mountains (Hestholm and Ruud, 1994; Bartel et al., 2000; Bohlen and Saenger, 2006). In contrast, finite-element methods (FEMs) easily handle unstructured meshes and spatial local refinement. Moreover, their extension to high-order spatial discretization schemes is straightforward, an important feature if we want to keep numerical dispersion minimal. The finite-element Galerkin approximation of hyperbolic problems typically leads to a system of ordinary differential equations. However, if explicit time stepping is subsequently employed, the mass matrix arising from the spatial discretization with standard conforming finite elements must be inverted at each time step, significantly reducing the efficiency of the scheme.

There are several classes of finite elements that do not require the inversion of a large sparse mass matrix, for instance, spectral, continuous mass-lumped, or discontinuous Galerkin (DG) elements. The spectral finite-element method (SEM) employs Gauss-Lobatto nodes for the discretization on hexahedra, leading to a diagonal mass matrix (Orszag, 1980; Patera, 1984; Rønquist and Patera, 1987; Maday and Rønquist, 1990; Komatitsch and Vilotte, 1998; Komatitsch and Tromp, 1999; Komatitsch et al., 1999; Mulder, 1999). However, hexahedral grids are less flexible than tetrahedral meshes, for instance, near pinch-outs. To overcome this problem, each tetrahedron can be divided into four hexahedra, increasing the number of nodes per tetrahedron (Taniguchi et al., 1996; Charara et al., 2011). The resulting hexahedral meshes, however, often have poor quality (Owen, 1998). Continuous finite

Manuscript received by the Editor 3 July 2012; revised manuscript received 21 November 2012; published online 29 April 2013.

¹Shell Global Solutions International BV, Rijswijk, The Netherlands.

²Source Contracting, Culemborg, The Netherlands.

³Shell Global Solutions International BV, Rijswijk, The Netherlands and Delft University of Technology, Department of Geoscience & Engineering, Faculty of Civil Engineering and Geosciences, Delft, The Netherlands.

© 2013 Society of Exploration Geophysicists. All rights reserved.

elements with mass lumping belong to the second class. In this case, the mass matrix is replaced by a diagonal one that is trivial to invert. This technique is straightforward for piecewise linear elements but requires particular quadrature rules and additional discretization nodes for higher-order schemes to preserve accuracy (Fried and Malkus, 1975; Cohen et al., 1993; Mulder, 1996; Chin-Joe-Kong et al., 1999). Currently, however, the set of additional nodes needed for mass lumping for tetrahedral elements beyond polynomial degree three (cubic shape function) is unknown for 3D problems. Discontinuous Galerkin methods (DGMs) offer an attractive alternative for the spatial discretization of time-dependent hyperbolic problems (Arnold et al., 2002) and geophysical applications (Käser and Dumbser, 2006; de la Puente et al., 2007; Etienne et al., 2010; De Basabe and Sen, 2010; Baldassari et al., 2011).

The DGM is a generalization of the FEM that allows the approximating functions to be discontinuous at the element's faces. The continuity is weakly imposed with some extra terms, so-called fluxes, in the weak formulation. Moreover, the discontinuous representation of the solution results in a block-diagonal mass matrix that is easily inverted. There are many different ways to weakly impose the continuity, leading to different formulations of the DGM. Apart from the advantages of the FEM mentioned above, the DGM can accommodate discontinuities, not only in the medium parameters but also in the wavefield; can be energy conservative; and can handle more general meshes. The method can accommodate elements of various types and shapes, irregular nonmatching grids, and even a locally varying polynomial degree; hence they offer greater flexibility in the mesh design.

Table 1 summarizes the main advantages and disadvantages of the three spatial discretizations. Formulations of the DGM can differ in the choice of fluxes and interior penalty. Some flux formulations use the velocity-stress form of the wave equation. Among them is the arbitrary high-order derivative discontinuous Galerkin (ADER-DG) by Käser and Dumbser (2006), which has the important characteristic that it yields the same order of accuracy in space and time. The time-stepping scheme is based on the Cauchy-Kovalevski or Lax-Wendroff (1960) procedure, replacing higher time derivatives by spatial derivatives using the partial differential equation. This trick is also known as the modified equation approach (Shubin and Bell, 1987) or Dablain's (1986) scheme. An interesting feature of the ADER-DG implementation is that it incorporates local time stepping (LTS), which can improve efficiency. Another class of schemes are the interior penalty formulations, based on the displacement formulation of the wave equation in the elastic case or on the pressure formulation in the acoustic case. These include the symmetric interior penalty Galerkin (SIPG), non-symmetric interior penalty Galerkin (NIPG), and incomplete interior penalty Galerkin (IIPG). Appendix A describes the details.

When a spatial DG discretization is combined with explicit time integration, the resulting time-marching scheme will be truly explicit and easily parallelized. This is straightforward in case of an element-by-element formulation. To have uniform accuracy, the size of the elements should scale with the dominant wavelength of the solution, hence with the local velocity. The mesh can be finer than required by the dominant wavelength for the following reasons: (1) singularities in the wavefield, for instance, around the source, requiring local mesh refinement and (2) small-scale geologic structures such as variations in the landscape, irregular interfaces, faults, and thin beds. Other difficulties arise when trying to mesh a region with large contrasts or complex geometries, such as pinch-outs. Also, near a sharp velocity contrast, mesh generation usually creates a transition zone to maintain conformity across the interface while avoiding poorly shaped tetrahedra. In that case, the desired element size as a function of the local velocity cannot be honored. Some of the elements in the transition zone will be too small. Unfortunately, meshes that are locally refined or finer than required for accurate representation of the wavefield impose severe stability constraints on explicit time-marching schemes, in which the maximum time step allowed by the Courant-Friedrichs-Lewy (CFL) condition (Courant et al., 1928) is dictated by the smallest elements in the mesh, where "small" is defined relative to the wavelength of the wavefield, hence the local wave speed. When mesh refinement is restricted to a small region, the use of a very small time step in the entire computational domain is a high price to pay.

To overcome the overly restrictive stability constraint, various LTS schemes (Collino et al., 2003; Piperno, 2006) have been developed, which use either implicit time-stepping or explicit but smaller time steps, but only where the smallest elements in the mesh are located. Because DGMs allow for a local formulation, they are particularly well suited for the development of explicit LTS schemes. Among several LTS schemes, we focus on the approach of Diaz and Grote (2009). Baldassari et al. (2011) use a similar approach to speed up reverse-time migration for 2D problems. Instead of assembling a global stiffness and mass matrix, as Diaz and Grote (2009) do, we consider a local formulation in which mass and stiffness matrices per element are assembled on the fly. This avoids the storage of the global mass and stiffness matrix, which is more important in 3D than in 2D. The local formulation requires a reformulation of the LTS method.

To have uniform accuracy, we would like to have a computational mesh in which the size of the elements is proportional to the local velocity. This strategy produces meshes with an optimal number of elements and ideally gives rise to a uniform discretization time step in the domain. However, small-scale geologic structures or shortcomings of the mesh generator as mentioned above will lead to a usually small number of elements that require LTS.

Table 1. Advantages and disadvantages of various discretization methods.

	FD	SEM	DG
Main features	Rely on Cartesian structure grid	High-order shape functions	Discontinuous shape functions
Advantages	Easy to implement, computationally efficient	High accuracy	Geometric flexibility
Disadvantages	Loss of accuracy	Limited geometric flexibility	Large number of unknowns

In the next two sections, we provide a description of the DG formulation and the LTS. Then, we briefly review the mesh generation procedure. Finally, we describe an heuristic algorithm for the selection of those elements that require LTS and present results for a 3D salt diapir model.

THE DISCONTINUOUS GALERKIN METHOD

There are many different DG formulations suitable for solving acoustic-wave propagation. Arnold et al. (2002) provide an overview of these variants and a unified formulation that summarizes most of them. They show that the interior penalty DGM guarantees stability and an optimal order of convergence. We refer to the papers by Grote et al. (2006) and Grote and Schötzau (2009) for the analysis of the convergence and conservation properties of the scheme when applied to the acoustic-wave equation. Baldassari et al. (2011) compare the method with spectral finite elements in terms of accuracy and performance. They conclude that the two methods perform well in terms of accuracy and that they involve the same computational burden. They consider the interior penalty DG formulation that is suitable for the widest range of applications because it can be applied on any mesh while keeping the same level of accuracy.

Here, we consider the second-order scalar wave equation:

$$\begin{cases} c^{-2}u_{tt} - \Delta u = f, & \text{in } (0, T) \times \Omega, \\ u|_{t=0} = u_0, \quad u_t|_{t=0} = v_0, & \text{in } \Omega, \end{cases} \quad (1)$$

on a bounded domain $\Omega \subset \mathbb{R}^3$ and with a given source term $f(t, \mathbf{x})$. Here, $u = u(t, \mathbf{x})$ represents the wavefield and $c = c(\mathbf{x})$ is the velocity of propagation, which is assumed to be piecewise constant per element and strictly positive. Moreover, we assume a constant density. Two initial conditions are needed, one for the pressure u and one for its first derivative in time u_t . To complete equation 1, suitable boundary conditions are imposed.

We briefly review the formulation of acoustic wave propagation when using the SIPG formulation. A detailed derivation can be found in a paper by Rivière (2008). Let τ_h denote a finite-element partition of the domain Ω . We collect all the faces of the elements in the set \mathcal{F}^h and the internal faces in the subset \mathcal{F}_i^h . Let $F \in \mathcal{F}_i^h$ be an interior face shared by two neighboring elements K^+ and K^- as shown in Figure 1. The outward unit normal on F is \mathbf{n} . In particular, we denote with \mathbf{n}^+ the normal on F^+ and with \mathbf{n}^- the normal on F^- . For a scalar function, the jump and average are defined as

$$\begin{aligned} [q] &:= q^+ \mathbf{n}^+ + q^- \mathbf{n}^-, \\ \{q\} &:= \frac{1}{2}(q^+ + q^-). \end{aligned} \quad (2)$$

We look for an approximate solution of the wave equation in the space, $W_p^h := \{w^h \in L^2(\Omega) : w^h|_K \in P_p(K), \forall K \in \tau_h\}$, with P_p polynomials up to degree p . A function that belongs to W_p^h can be discontinuous on the internal boundaries F^+ and F^- . The first step consists of multiplying the equation by a test function and integrating over space to obtain the so-called weak formulation. If, for the sake of simplicity, we assume

homogeneous Neumann boundary conditions, meaning that the normal derivative of the solution equals zero (other types of boundary conditions can be used such as Dirichlet at the free surface or absorbing boundary conditions that will lead to minor changes in the formulation), then the problem becomes: Find $u : (0, T) \times \mathbb{R}^3 \rightarrow W_p^h$ such that

$$\begin{cases} (c^{-2}u_{tt}, v) + a(u, v) = (f, v), & \forall v \in W_p^h, t \in (0, T), \\ u|_{t=0} = u_0, \quad u_t|_{t=0} = v_0, & \text{in } \Omega, \end{cases} \quad (3)$$

where v is the test function. The bilinear form $a(u, v)$ is defined by

$$\begin{aligned} a(u, v) &= \sum_{K \in \tau_h} \int_K \nabla u \cdot \nabla v dK - \sum_{F \in \mathcal{F}_i^h} \int_F [u] \cdot \{\nabla v\} dF \\ &\quad - \sum_{F \in \mathcal{F}_i^h} \int_F [v] \cdot \{\nabla u\} dF + \sum_{F \in \mathcal{F}_i^h} \int_F \gamma [u] \cdot [v] dF. \end{aligned} \quad (4)$$

The three integrals on the faces define the fluxes across the element boundaries (see Appendix A for details). The last integral is the penalty term with a parameter γ depending on the size of the elements of the partition τ_h . This term is necessary to guarantee the optimal convergence of the spatial discretization. It vanishes in case of a continuous solution because then the jump is zero. We define h as the diameter of the inscribed sphere for the element K . The mesh size can then be taken as $h_{\min} = \min_{K \in \tau_h} h$. We generalize the coefficient γ proposed by Ainsworth et al. (2006) to 3D by choosing $\gamma = N_{DG}/h_{\min}$, with $N_{DG} = (p+1)(p+2)(p+3)/6$ being the number of nodes per element for polynomial degree p . Considering that we will choose an explicit scheme to discretize the second-order time derivative, the semidiscrete problem in space is formulated on a single element K^+ as

$$\mathbf{M}^+ \mathbf{u}_{tt} + \mathbf{S}^+ \mathbf{u} + \sum_{F \in \mathcal{F}_i^+} \mathbf{F}_F^+ \mathbf{u} + \sum_{F \in \mathcal{F}_i^-} \mathbf{F}_F^- \mathbf{u} = \mathbf{f}, \quad (5)$$

where \mathbf{M}^+ and \mathbf{S}^+ are the local mass and stiffness matrix, respectively, both of dimension $N_{DG} \times N_{DG}$; and \mathbf{u}^- is the solution on one of the neighboring elements K^- . We also have the contribution of

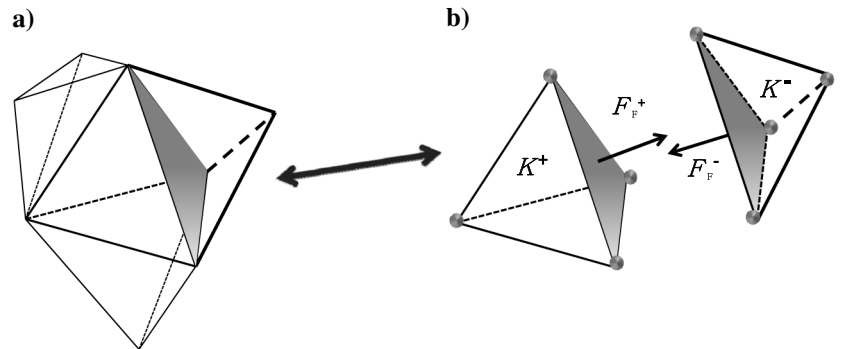


Figure 1. Schematic representation of the fluxes for tetrahedral elements. (a) One element with 4 neighbors. (b) The element K^+ discretized by polynomial shape functions of degree one. The element has four degrees of freedom corresponding to the vertices of the tetrahedron. On each face, we have an outgoing flux F_F^+ and an incoming flux F_F^- from the neighboring element.

the fluxes. The first term denotes the sum of outgoing fluxes over faces F in the element K^+ . The second term contains incoming fluxes from the neighboring elements K^- . In this way, we distinguish the computation of fluxes for the element K^+ from the contribution coming from the neighboring elements K^- .

LOCAL TIME STEPPING

To the best of our knowledge, high-order LTS methods are considered for the wave equation by [Dumbser et al. \(2007\)](#) and [Diaz and Grote \(2009\)](#). The approach proposed in [Käser and Dumbser \(2006\)](#), [Dumbser and Käser \(2006\)](#), and [Dumbser et al. \(2007\)](#) is based on the use of arbitrary high-order derivatives methods. The method of [Diaz and Grote \(2009\)](#) conserves a discrete energy, which guarantees the long-time stability of the scheme. Moreover, it can be coupled with any type of spatial discretization method, whereas the arbitrary high-order derivatives method requires the use of DGMs. The approach has been extended by [Baldassari et al. \(2011\)](#) to allow for p -adaptivity in time as well as in space when using an interior penalty scheme.

In this section, we summarize the main ideas behind LTS as [Diaz and Grote \(2009\)](#) propose and discuss our modification of the algorithm to solve the problem element by element. A detailed derivation of the scheme and its energy conservation properties can be found in the cited paper. To ensure stability, an explicit time-integration scheme must satisfy the usual CFL condition that restricts the time step Δt by $\Delta t \leq \text{CFL}(h/c)$ for each element. Here, c is the velocity inside the element, h is the diameter of the inscribed sphere, and the CFL number CFL is dictated by the numerical scheme. The element in the domain with the smallest ratio $r = h/c$ defines the actual largest time step Δt_{\max} that can be used for computing the solution of the problem. Polynomials of higher degree require a more restrictive CFL number. In [Zhebel et al. \(2012\)](#), we estimated the value of CFL for polynomial degrees one to four on tetrahedra. We also analyzed its dependence on the distortion of the element. [Epshteyn and Rivière \(2007\)](#) present a theoretical analysis to estimate the lower bounds of the penalty parameters for stable and convergent symmetric interior penalty Galerkin methods. In particular, the authors derive the explicit dependence of the penalty parameter on the polynomial degree, angles, and diameter of the mesh elements. However, the study was carried out for a stationary reaction-diffusion equation.

The restriction on the time step Δt can severely impact the efficiency of the simulation when most of the elements in the domain have a ratio exceeding $r_{\min} = \min_{K \in \tau_h} h/c$. The LTS approach aims to divide the elements into two or more subsets. For simplicity, we only consider two sets. The set of coarse elements S_c consists of the coarse elements that do not share any face with a fine element. The set of the fine elements S_f contains all the other elements. The solution for the fine elements is computed with a time step $\Delta \tau = \text{CFL} r_{\min}$, whereas on the coarse elements $\Delta t = q \Delta \tau$ for an integer $q > 1$. Then, for each global time step Δt , each local time step of size $\Delta t/q$ requires the solution of a modified problem, which we denote by \mathcal{P} . The second-order ‘‘leapfrog’’ time-integration scheme is used to compute the solution for all the elements in S_c . The fully algebraic system formulated on each element reads

$$\mathbf{u}_{n+1} = 2\mathbf{u}_n - \mathbf{u}_{n-1} - \Delta t^2 (\mathbf{M}^+)^{-1} (\mathbf{S}^+ \mathbf{u}_n + \mathbf{F}^+ \mathbf{u}_n + \mathbf{F}^- \mathbf{u}_n^-), \quad (6)$$

where we assumed the source term f to be zero. Equation 6 is used to compute the local solution \mathbf{u}_{n+1} at time t_{n+1} , knowing the solution at two previous time steps, \mathbf{u}_{n-1} and \mathbf{u}_n .

To describe the modified problem \mathcal{P} on the elements in S_f , we briefly summarize the derivation of the algorithm for a globally assembled problem. We express the global semidiscrete problem on the domain as

$$\mathbf{y}_{tt} + \mathbf{A}\mathbf{y} = 0, \quad (7)$$

where \mathbf{y} is the vector of the unknowns and \mathbf{A} is the elliptic operator containing the stiffness matrix as well as the contribution of the fluxes. First, we rewrite the unknown vector as the sum of two components, $\mathbf{y} = \mathbf{y}_c + \mathbf{y}_f$, where $\mathbf{y}_c = (\mathbf{I} - \mathbf{P})\mathbf{y}$ and $\mathbf{y}_f = \mathbf{P}\mathbf{y}$, where \mathbf{P} is a diagonal matrix with unit entries corresponding to the unknowns belonging to the fine region and \mathbf{I} is the identity matrix. The solution of equation 7 satisfies

$$\begin{aligned} \mathbf{y}(t + \Delta t) - 2\mathbf{y}(t) + \mathbf{y}(t - \Delta t) \\ = -\Delta t^2 \int_{-1}^1 (1 - |\theta|) \mathbf{A}\mathbf{y}(t + \theta \Delta t) d\theta. \end{aligned} \quad (8)$$

Rewritten in terms of \mathbf{y}_c and \mathbf{y}_f , this becomes

$$\begin{aligned} \mathbf{y}(t + \Delta t) - 2\mathbf{y}(t) + \mathbf{y}(t - \Delta t) = \\ -\Delta t^2 \int_{-1}^1 (1 - |\theta|) [\mathbf{A}\mathbf{y}_c(t + \theta \Delta t) + \mathbf{A}\mathbf{y}_f(t + \theta \Delta t)] d\theta. \end{aligned} \quad (9)$$

To circumvent the severe stability constraints, we have to treat \mathbf{y}_f in a different way. The following approximation is introduced by [Diaz and Grote \(2009\)](#):

$$\begin{aligned} \mathbf{y}(t + \Delta t) - 2\mathbf{y}(t) + \mathbf{y}(t - \Delta t) \approx -\Delta t^2 \mathbf{A}\mathbf{y}_c(t) \\ - \Delta t^2 \int_{-1}^1 (1 - |\theta|) \mathbf{A}\mathbf{y}_f(t + \theta \Delta t) d\theta. \end{aligned} \quad (10)$$

This approximation allows us to solve the evolution of the solution \mathbf{y} during the time step Δt while fixing the contribution of the coarse elements at the time step t_n . With some additional technical steps and using the symmetry properties of the solution ([Diaz and Grote, 2009](#); [Grote and Mitkova, 2010](#)), the term that involves \mathbf{y}_f can be approximated as

$$\int_{-1}^1 (1 - |\theta|) \mathbf{A}\mathbf{y}_f(t + \theta \Delta t) d\theta = \int_0^1 \mathbf{A}\mathbf{P}\mathbf{q}(\tau) d\tau, \quad (11)$$

where $\mathbf{q}(\tau)$ solves the following modified differential problem, \mathcal{P} , for $\tau \in [0, \Delta t]$:

$$\mathbf{q}(0) = \mathbf{y}^n, \quad \mathbf{q}_t(0) = 0, \quad \mathbf{q}(\tau)_{tt} = -\mathbf{A}\mathbf{y}_c(t) - \mathbf{A}\mathbf{P}\mathbf{q}(\tau). \quad (12)$$

To advance the solution \mathbf{y}_n from t_n to t_{n+1} , we have to evaluate $\mathbf{q}(\Delta t)$ by solving equation 12 on $[0, \Delta t]$. We solve equation 12 with time step $\Delta \tau$ until $\tau = \Delta t$, using q steps of the leapfrog scheme with a smaller time step $\Delta \tau = \Delta t/q$, where q is the rate of local refinement. The solution of the global time step can then be reconstructed as $\mathbf{y}_{n+1} = -\mathbf{y}_{n-1} + 2\mathbf{q}(\Delta t)$. In case of $\mathbf{P} = 0$, there is no LTS

($q = 1$) and we recover the standard leapfrog scheme. If the fraction of nonzero entries in \mathbf{P} is small, the overall cost is dominated by the computation of the solution on the coarse elements.

So far, the modified problem \mathcal{P} has been formulated for the global problem. We will now outline the algorithm to do the same element by element. For each element that belongs to S_f , we solve the system of equations 12 with the leapfrog scheme and time step $\Delta\tau$ for q iterations. We need two initial conditions $\hat{\mathbf{u}}_0$ and $\hat{\mathbf{u}}_1$ to start the iterations, as shown in Figure 2. These conditions are

$$\hat{\mathbf{u}}_0 = \mathbf{u}_n, \quad \hat{\mathbf{u}}_1 = \hat{\mathbf{u}}_0 - \frac{1}{2} \left(\frac{\Delta t}{q} \right)^2 (\mathbf{M}^+)^{-1} (\mathbf{w} + \mathbf{J}\hat{\mathbf{u}}_0). \quad (13)$$

For $m = 2, \dots, q - 1$, we use the leapfrog scheme to advance the solution, which in this case reads,

$$\begin{aligned} \hat{\mathbf{u}}_{(m+1)/q} &= 2\hat{\mathbf{u}}_{m/q} - \hat{\mathbf{u}}_{(m-1)/q} \\ &+ \left(\frac{\Delta t}{q} \right)^2 (\mathbf{M}^+)^{-1} (\mathbf{w} + \mathbf{J}\hat{\mathbf{u}}_{m/q}). \end{aligned} \quad (14)$$

The vector \mathbf{w} is kept fixed during the q iterations and is constructed from solution \mathbf{u}_n on the element at time t_n . For the computation of \mathbf{w} and \mathbf{J} , we have to consider three cases that we summarize as follows:

- 1) Fine element that shares one or more faces j with coarse neighbors. Let us define \mathcal{F}_c as the set that contains the faces shared with a coarse neighbor. The flux coming from the coarse neighbors is computed at time t_n . Therefore,

$$\mathbf{w} = \sum_{j \in \mathcal{F}_c} \mathbf{F}_F^- \mathbf{u}_n^- \quad (15)$$

and

$$\mathbf{J} = \mathbf{S}^+ \hat{\mathbf{u}}_{m/q} + \sum_{F \in \mathcal{F}_1 \setminus \mathcal{F}_c} \mathbf{F}_F^+ \hat{\mathbf{u}}_{m/q} + \sum_{F \in \mathcal{F}_1 \setminus \mathcal{F}_c} \mathbf{F}_F^- \hat{\mathbf{u}}_{m/q}. \quad (16)$$

The term \mathbf{J} contains the stiffness matrix and all the fluxes that are updated at each iteration q .

- 2) Coarse element that shares one or more faces j with a fine neighbor. Let us define \mathcal{F}_f as the set that contains the faces shared with fine neighbors. Thus, we have

$$\mathbf{w} = \mathbf{S}^+ \mathbf{u}_n + \sum_{F \in \mathcal{F}_1 \setminus \mathcal{F}_c} \mathbf{F}_F^+ \mathbf{u}_n + \sum_{F \in \mathcal{F}_1 \setminus \mathcal{F}_c} \mathbf{F}_F^- \mathbf{u}_n \quad (17)$$

and

$$\mathbf{J} = \sum_{j \in \mathcal{F}_f} \mathbf{F}_F^- \hat{\mathbf{u}}_{m/q}. \quad (18)$$

Here, \mathbf{w} contains the stiffness matrix and the fluxes that do not need to be updated at each iteration q because they are coming from coarse neighbors.

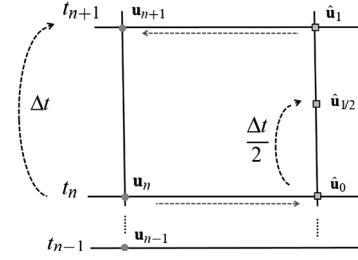


Figure 2. Schematic representation of the LTS algorithm in the case of $q = 2$. The solution u_{n+1} for a coarse element is computed in a single step. For the other elements, the initial step \hat{u}_0 is computed based on the solution u_n . Then, two time steps are performed to build the solution \hat{u}_1 used to update u_{n+1} .

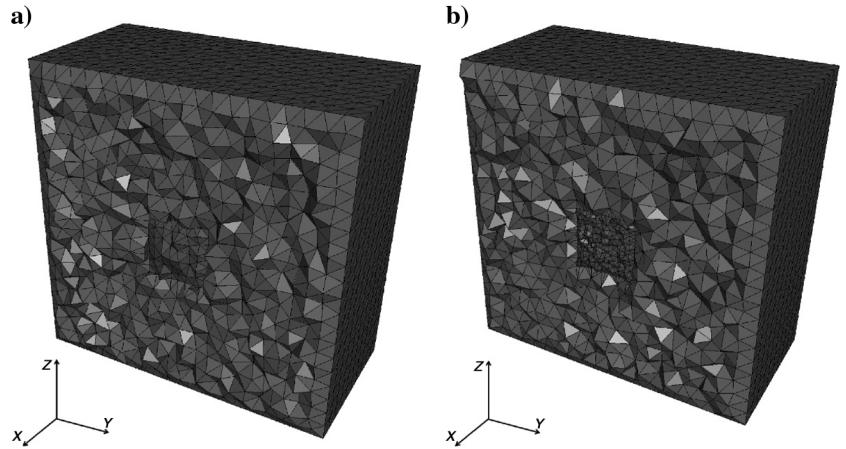


Figure 3. Irregular tetrahedral meshes used in the convergence analysis. In both cases, $h = h_{\text{coarse}} = 0.0531$. In (a), the inner region has $h_{\text{fine}} = h/2$. In (b), the inner region has $h_{\text{fine}} = h/4$.

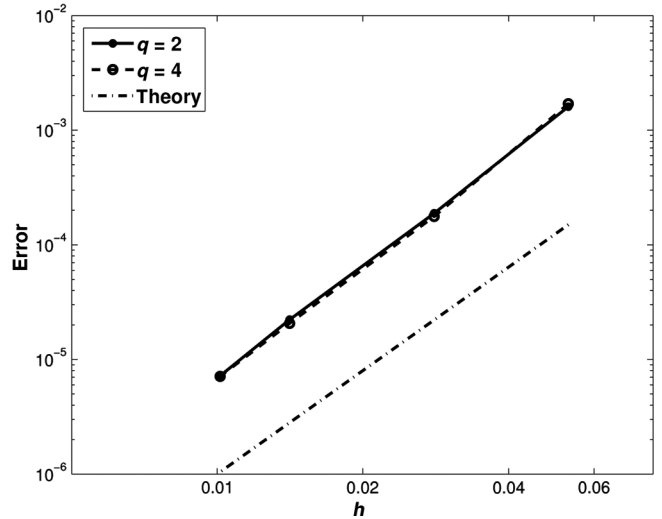


Figure 4. Convergence analysis on a series of four meshes with $h = h_{\text{coarse}} = 0.0531, 0.0281, 0.0141, \text{ and } 0.01015$, for $q = 2$ and $q = 4$.

3) Finally, for a fine element with no coarse neighbor, $\mathbf{w} = 0$ and

$$\mathbf{J} = \mathbf{S}^+ \hat{\mathbf{u}}_{m/q} + \sum_{F \in \mathcal{F}_I \subset K^+} \mathbf{F}_F^+ \hat{\mathbf{u}}_{m/q} + \sum_{F \in \mathcal{F}_I \subset K^-} \mathbf{F}_F^- \hat{\mathbf{u}}_{m/q}. \quad (19)$$

The proposed LTS scheme is second-order accurate in time. Diaz and Grote (2009) (Lemma 3.2) prove the accuracy of the scheme by rewriting the formulation in a more general form. They also prove that the scheme conserves the discrete energy (section 3.2 in Diaz and Grote, 2009) and is stable under a CFL condition. They perform a systematic numerical study and show that the overall rate of convergence is independent of the number of local time steps q .

Nevertheless, for completeness, we test the convergence on a simple example to confirm the expected numerical accuracy. We consider 3D wave propagation with a constant wave speed c of 1 km/s on the cube $\Omega = [0, 1] \times [0, 1] \times [0, 1]$ km³ for the time interval $[0, 1]$ s. The initial conditions are chosen to yield the exact solution $u_{\text{ex}}(\mathbf{x}, t) = -\cos(k\mathbf{x}) \cos(kct)$, where k is the spatial frequency, taken equal to five half-periods over the length of the domain. We insert inside the domain a smaller cube-shaped region R_f with dimensions of $[0.1] \times [0.1] \times [0.1]$ km³, discretized with tetrahedra of size $h_{\text{fine}} = h_{\text{coarse}}/q$, h_{coarse} being the size of the elements outside R_f . We consider a spatial discretization with quadratic polynomial shape functions and a second-order leapfrog time-stepping scheme. We perform the convergence analysis on a series of meshes with decreasing element sizes. Figure 3 depicts two of them. We reduce the global mesh size h_{coarse} as well as h_{fine} and the time step. The CFL is kept constant and equal to 0.01.

We measure the difference between the numerical solution and the exact one in terms of the L^2 error norm, defined as

$$\left(\int_{\Omega} (u - u_{\text{ex}})^2 \Big|_{t=T} d\Omega \right)^{1/2}, \quad (20)$$

where $T = 1$ s is the final time. Figure 4 shows the numerical error against the size of the mesh $h = h_{\text{coarse}}$. We found that, for $q = 2$ and 4, the convergence rate is of order three, in agreement with the theoretical estimates (Zampieri and Pavarino, 2006; Grote and Schötzau, 2009). Figure 5 displays the recorded signal at one of the receiver's position.

MESH GENERATION

The complexity in mesh generation for seismic applications is due to sharp and abruptly varying interfaces and a large modeling volume. Mesh generation becomes even more complex if there is a large number of interfaces and if they have multiple intersections. Moreover, to have the equivalent of a fixed number of points per wavelength as is common with finite differences and to have all elements running at not too small a time step, the element size should be proportional to the local velocity. Our approach is based on five consecutive steps, as summarized in

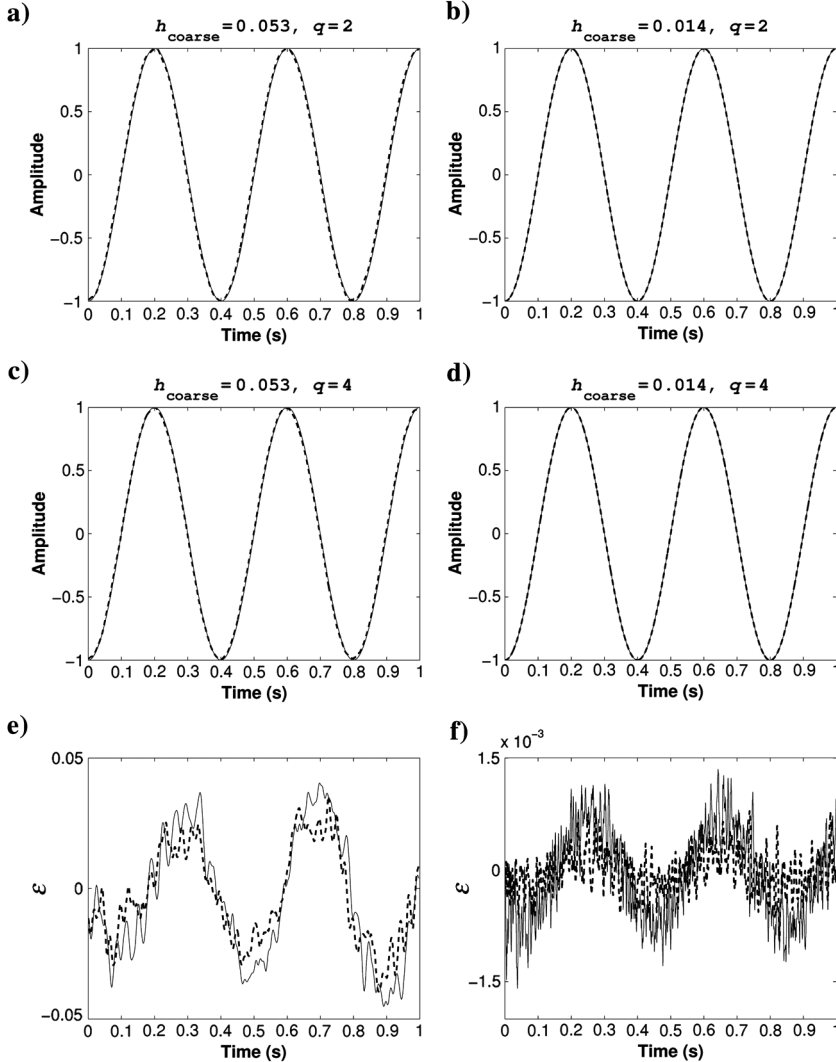


Figure 5. (a, b, c, d) Comparison between the numerical solution (dashed line) and the exact solution (solid line) at one receiver position. We use two meshes with $h_{\text{coarse}} = 0.053$ and 0.014. (a, b) Refer to the experiments on the two meshes with $q = 2$, while (c, d) have $q = 4$. (e, f) Difference between the exact and the numerical solution, $\epsilon = u - u_{\text{ex}}$, for the cases $q = 2$ (solid line) and $q = 4$ (dashed line).

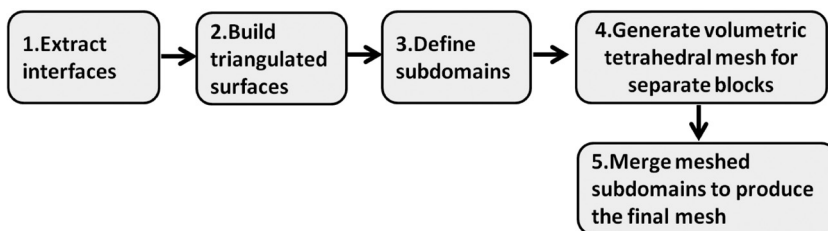


Figure 6. The five main steps needed to build the mesh.

Figure 6. We refer to the paper by Kononov et al. (2012) for a full description of the mesh generation procedure. The first step consists of the extraction of the interfaces, the surfaces that describe the discontinuities in the velocity model. Often, the velocity field is described on a finite-difference grid. That is because most of the realistic models were constructed with the aim of producing a finite-difference wave-propagation solution. Then, the interfaces need to be constructed by finding and extracting the locations of the discontinuities. Thus, the input consists of a set of interface points together with the velocities on the original finite-difference grid. For large problems, the domain can be divided into a few blocks to accelerate the whole meshing procedure, allowing for parallel mesh generation for each block. In the second step, the extracted interfaces are transformed into triangulated surface meshes bounding the subdomains. These should be identical over the shared areas and edges of neighboring subdomains. If the initial model is already specified in a surface mesh form, steps 1 and 2 can be skipped. However, most of the time, the surfaces coming from CAD models need to be remeshed and regularized. The third step is to disassemble the model into several interconnected volumes whose connectivity is accomplished via the compatible triangulated surface meshes. As a fourth step, one has to generate volumetric tetrahedral meshes using the surface meshes as input to a 3D mesh generator. The size of the tetrahedra is controlled by either the velocity fields enclosed in the subdomains or by the surface mesh sizes of nearby interfaces. The fifth step is to merge the meshed subdomains together, producing the final mesh. This approach has many advantages: it can be directly parallelized in a straightforward manner because each volume and each surface can be meshed independently, thereby reducing the total meshing time; further mesh adjustments and optimization can be applied locally for each subdomain; and, if the quality of surfaces meshes is good enough, then the overall mesh can be easily refined or coarsened by applying the corresponding operations to the surface and volumetric meshes.

RESULTS

To illustrate how LTS can speed up the computation, we consider wave propagation in a 3D geometry with variable velocity for the model in Figure 11. The geometry is typical for a part of the North Sea. The domain Ω has a size of $4.4 \times 4.4 \times 5.1 \text{ km}^3$. The model contains eight different layers, and the velocity increases with depth from 1.5 km/s to 4.0 km/s. In each region, the velocity is assumed to be constant. The orange inclusion in Figure 11 is a salt diapir with a high velocity of 4.5 km/s. For simplicity, we impose a reflecting Neumann boundary condition on the boundary of Ω in this test. The source is located below the surface, as shown in Figure 14. We describe the steps for the generation of the mesh of Figure 6 for this specific example. From the initial velocity model, we extract the interfaces or contrasts among regions with a different velocity as shown in Figure 7. Then we make a 2D triangulation of the extracted layers and the salt, Figure 8. We then merge all the surfaces in such a way that the nodes in the intersections are matching, Figure 9. Finally, we grow the tetrahedra in the volumes. An example of a mesh volume is shown in Figure 10. In this step, we impose the constraint that the diameter of the elements has to be proportional to the velocity. For instance, the water layer, the dark-blue region in Figure 11, has a low velocity and small tetrahedra, whereas the salt diapir, the orange volume, has a high velocity and large elements. Ideally, the adaptation of the mesh to the

velocity field would lead to a uniform Δt in the domain. However, due to the complex geometry of the 3D layers and the thinness of some layers close to the salt, as marked in black in Figure 11, some elements are smaller than necessary for resolving the dominant wavelength of the wavefield. Geometrical constraints and the limitations of mesh generation lead to elements with a ratio $r = h/c$ smaller than that of most of the other elements, which in turn reduces the maximum allowable time step.

To apply LTS, we first have to identify the two sets of elements, coarse and fine. In 3D, this is not as simple as in 1D or 2D. In our example, there is no unique region where the elements have a clearly smaller ratio of h/c . Instead, these elements are scattered throughout the domain. To use LTS to its full potential, we should have a set of fine elements that contains far less elements than the set of coarse ones. To find the optimal balance between the fine and coarse sets, we determine them for 10 different values of $q = 1, \dots, 10$. We compute Δt_{\max} in the domain and then select coarse elements whose ratio h/c is larger than $q\Delta t_{\max}/\text{CFL}$. The rest of the elements are fine. In the computation, we use finite

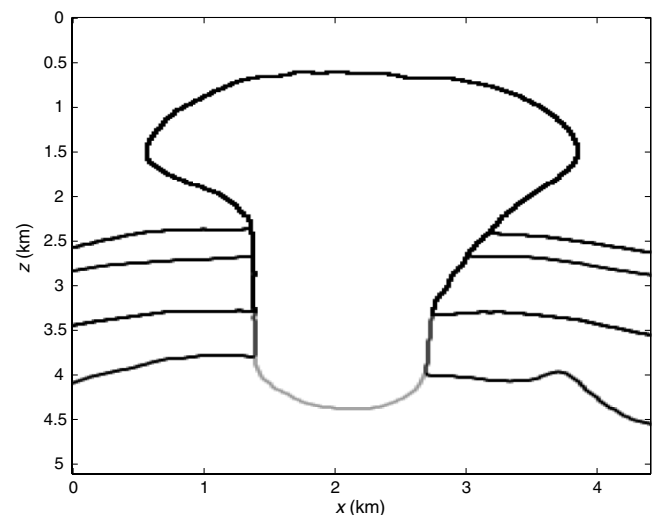


Figure 7. Identification and extraction of the interfaces that corresponds to velocity discontinuities.

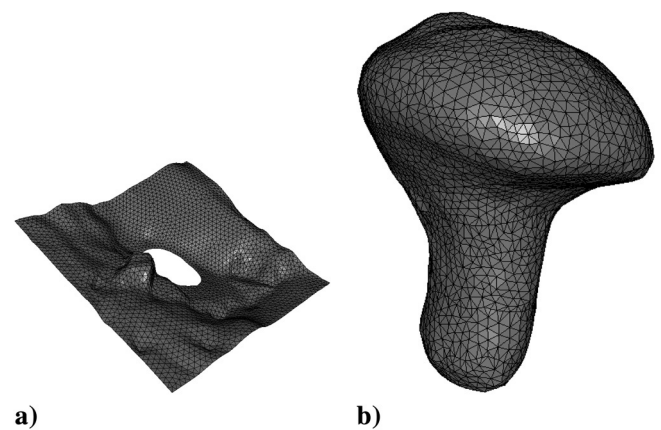


Figure 8. Triangulated surfaces. (a) A layer intersecting with the salt. (b) The salt body.

elements of degree two (P_2) and $CFL = 0.07$ equal to 70% of the maximum value of CFL for P_2 as estimated by Zhebel et al. (2012). The histogram in Figure 12 shows the distribution of fine and coarse elements, with black for the percentage of coarse elements and gray for the fine elements. For $q = 1, \dots, 5$, the number of fine elements is less than the 10%. The total computational cost can be estimated as the number of iterations $N_{it} = T/\Delta t_{max}$, T being the simulation time, multiplied by the number of elements in the domain. In the case of LTS, the computational cost can be estimated as $N_f N_{it} + N_c(N_{it}/q)$, where N_f is the number of fine elements and N_c is the number of coarse elements. When no LTS is applied, the cost becomes $(N_c + N_f)N_{it}$. The speed-up is then defined as $(N_f + N_c)/(N_f + N_c/q)$. Figure 13 shows the estimated speed-up as a function of q for the current example. Choosing $q = 2$

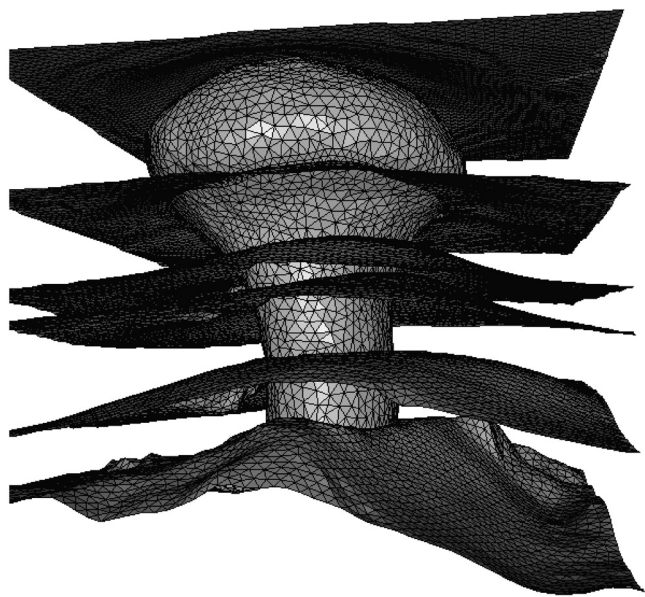


Figure 9. Triangulated surfaces merged together. The grid is conforming.

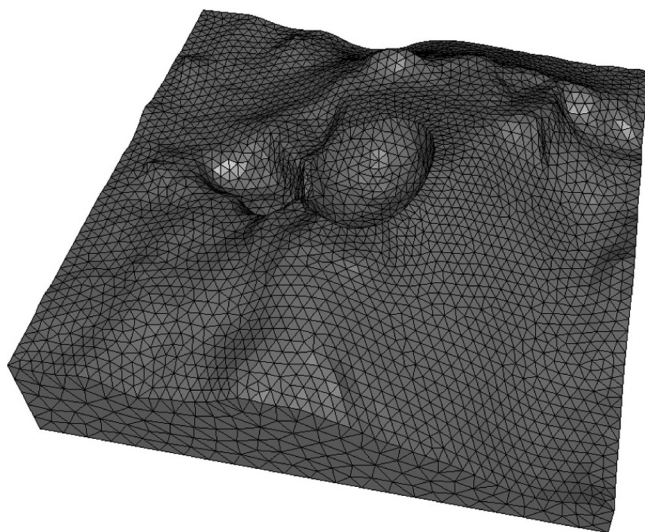


Figure 10. Subdomain of the model filled with tetrahedra.

speeds up the computation by a factor of about two, whereas the optimal choice is $q = 4$. Figure 14 displays the modeled wavefield along a section of the 3D domain. The source is located in a region with a constant velocity of 1.7 km/s. We ran the computation with $q = 4$. The total number of time steps for the coarse elements is 2104, whereas without LTS, we would need 8417 steps for each element. The fine set contains 20,648 elements, and the coarse set contains 1,268,144. In terms of computational time, LTS was observed to be four times faster. Figure 15 shows a section of the domain with the fine elements highlighted in red. The other elements are coarse. The fine elements are concentrated along the connection between regions with different velocity. It is clear that the LTS does not require a contiguous volume of fine elements but allows them to be isolated.

The computations were performed in parallel using OpenMP in a shared-memory environment, treating each element in parallel but having all elements share the same memory. The execution of the

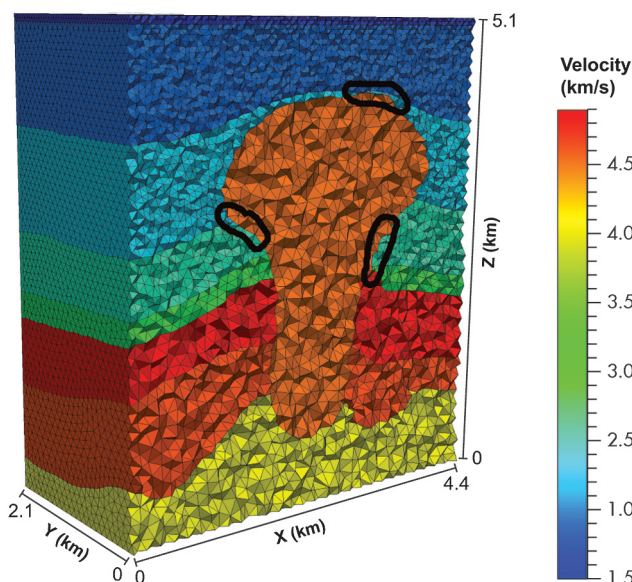


Figure 11. Mesh used for the computation. Black lines identify regions where the size of the elements is not properly scaled with the velocity.

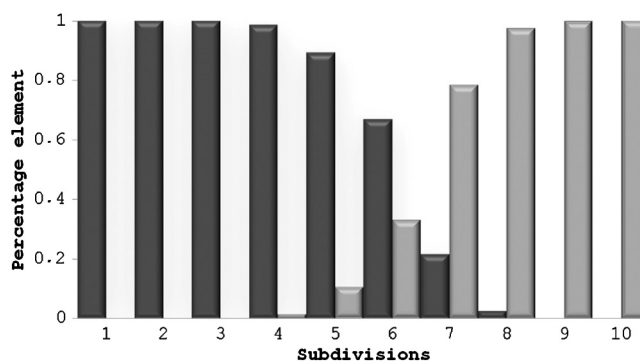


Figure 12. Distribution of the elements. For each value of q , we can determine the number of the elements in S_f represented with gray bars and those in S_c with black bars. A fine element has $h/c(\mathbf{x}) \leq q\Delta t_{max}/CFL$.

LTS in an MPI environment is more challenging (Dumbser et al., 2007). In that case, the partitioning of mesh should account for the presence of fine or coarse elements because they require a different number of updates, which will impact the overall communication. Therefore, it is not easy to find a good load balance. The LTS can be used for those subsets in which, due to the complexity of the geometry, the size of the elements is too small compared to the dominant wavelength.

Another interesting application is the use of the LTS formulation together with p -adaptivity. Etienne et al. (2010) propose using p -adaptivity to mitigate the presence of undersized elements in the domain. As lower-degree polynomials have a less restrictive CFL, they can be used for those elements. Baldassari et al. (2011) apply a hybrid LTS strategy, with a varying polynomial degree in the domain, to simulate reverse-time migration in a 2D setting.

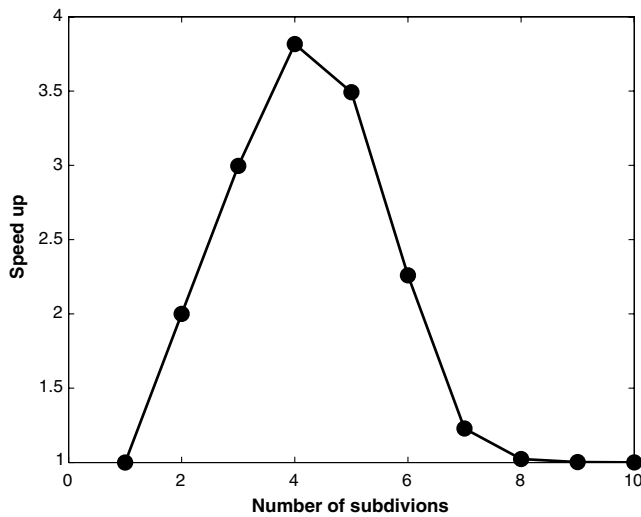


Figure 13. Computational speed-up for different values of q . The graph shows that $q = 4$ is the optimal choice to divide the elements between fine and coarse because it results in a speed-up close to 4.

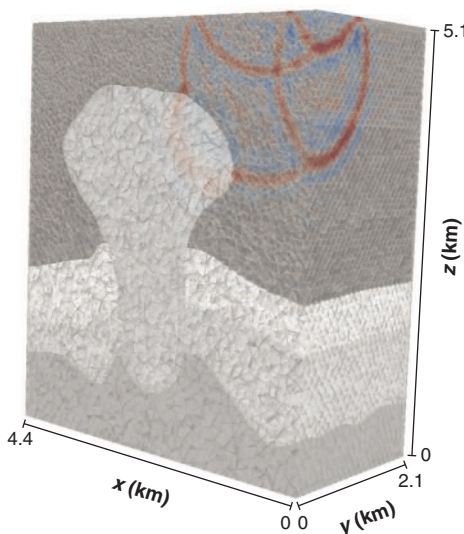


Figure 14. Snapshot of the wavefield after 0.48 s. The total simulation time is 1 s.

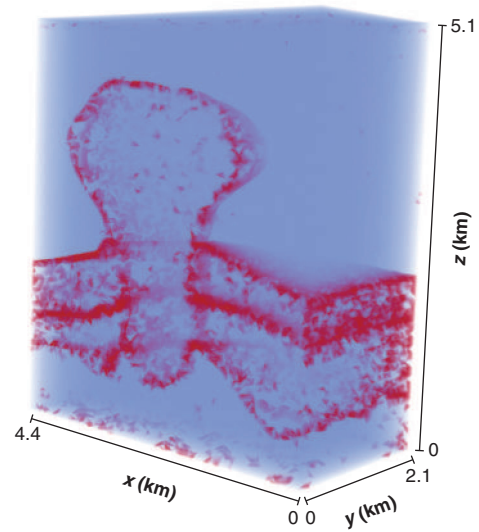


Figure 15. Distribution of the elements for $q = 4$. The red elements in this cut-away view are classified as fine, the others are classified as coarse.

CONCLUSIONS

We have described and tested an explicit LTS method for wave-propagation modeling, allowing for arbitrarily small time steps precisely where the smallest elements in the mesh are located. Here, smallness is defined in terms of the element diameter compared to the local dominant wavelength. The latter is determined by the local velocity. The speed-up of the computation depends on the mesh and on the ratio between the minimum and maximum values of the local diameter divided by the velocity in the mesh. In the example, we have considered only two types of elements, coarse and fine, but the method can be applied in a hierarchical way. Each local time step in the fine region can itself include further local time steps inside a smaller subregion with an even higher degree of local time refinement. This would be necessary to model structures that are an order of magnitude smaller than the dominant wavelength, for instance, a fracture.

APPENDIX A

DISCONTINUOUS GALERKIN FORMULATION

In this appendix, we provide a brief description of the interior penalty scheme. Starting from the weak formulation, we show the steps needed to obtain the semidiscrete formulation of the problem. At the end, we introduce the basis functions considered in this paper.

To discretize the wave equation 1 with finite elements, we multiply by a test function and we integrate over the domain. After integrating by parts, we obtain the so-called weak formulation,

$$\begin{aligned} \int_{\Omega} \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} \phi d\Omega + \int_{\Omega} \nabla u \cdot \nabla \phi d\Omega - \int_{\delta\Omega} (\mathbf{n} \cdot \nabla u) \phi d\Omega \\ = \int_{\Omega} f \phi d\Omega, \end{aligned} \quad (\text{A-1})$$

for all test functions ϕ that are chosen as piecewise polynomials up to degree p . Here, \mathbf{n} denotes the outward normal to the boundary and $\delta\Omega$ consists of internal and external boundaries of the domain Ω . In case of DG finite elements, the solution and the shape functions ϕ are discontinuous across the internal boundaries. The third integral term in equation A-1, called a flux term, is given by

$$\begin{aligned} & +\epsilon \int_{\delta\Omega} [u] \cdot \{\nabla\phi\} d\Omega - \int_{\delta\Omega} [\phi] \cdot \{\nabla u\} d\Omega \\ & + \gamma \int_{\delta\Omega} [u] \cdot [\phi] d\Omega. \end{aligned} \quad (\text{A-2})$$

If u^+ is the solution inside the element K^+ and u^- lives on one of the neighboring elements K^- , then $[u] := u^+ \mathbf{n}^+ + u^- \mathbf{n}^-$ denotes the jump across the element boundary and $\{u\} := \frac{1}{2}(u^+ + u^-)$ is the average, whereas γ is a penalty parameter. Different values of the parameters ϵ and γ define different variants of the interior penalty finite-element discretization:

- the SIPG with $\epsilon = -1$ and $\gamma > 0$;
- the IIPG with $\epsilon = 0$ and $\gamma > 0$;
- the NIPG with $\epsilon = 1$ and $\gamma > 0$.

If we express the solution in the basis of the test functions $u(x, y, z, t) = \sum u_j(t) \phi_j(x, y, z)$, then the flux term for a face F of element K^+ is given by

$$\begin{aligned} & -\frac{1}{2} \int_F (\nabla\phi_j \cdot \mathbf{n})^+ \phi_j^+ u_j^+ ds + \frac{1}{2} \int_F (\nabla\phi_j \cdot \mathbf{n})^- \phi_j^- u_j^- ds \\ & + \frac{\epsilon}{2} \int_F (\nabla\phi_i \cdot \mathbf{n})^+ \phi_j^+ u_j^+ ds - \frac{\epsilon}{2} \int_F (\nabla\phi_i \cdot \mathbf{n})^- \phi_j^- u_j^- ds \\ & + \gamma \int_F u_j^+ \phi_j^+ \phi_i^+ ds - \gamma \int_F u_j^- \phi_j^- \phi_i^- ds \\ & = : (A_{ij} + \epsilon B_{ij} + \gamma C_{ij}) u_j^+ + (D_{ij} + \epsilon E_{ij} + \gamma F_{ij}) u_j^-. \end{aligned} \quad (\text{A-3})$$

We then define

$$\mathbf{F}_F^+ = (A_{ij} + \epsilon B_{ij} + \gamma C_{ij}), \quad \mathbf{F}_F^- = (D_{ij} + \epsilon E_{ij} + \gamma F_{ij}). \quad (\text{A-4})$$

The semidiscrete problem in space for each element becomes

$$\mathbf{M}^+ \mathbf{u}_t + \mathbf{S}^+ \mathbf{u} + \mathbb{F}^+ + \mathbb{F}^- = \mathbf{f}, \quad (\text{A-5})$$

where the mass matrix is

$$\mathbf{M}_{ij}^+ = \int_{K^+} \phi_j \phi_i dK \quad (\text{A-6})$$

and the stiffness matrix is

$$\mathbf{S}_{ij}^+ = \int_{K^+} \nabla\phi_j \cdot \nabla\phi_i dK, \quad (\text{A-7})$$

whereas for the fluxes we have to sum over the four faces of the element

$$\mathbb{F}^+ = \sum_{F \in \mathcal{F}_j \subset K^+} \mathbf{F}_F^+ \mathbf{u}_n, \quad \mathbb{F}^- = \sum_{F \in \mathcal{F}_j \subset K^-} \mathbf{F}_F^- \mathbf{u}_n. \quad (\text{A-8})$$

With the DGM, the basis functions are not required to be continuous over the entire domain but only inside the elements. This important feature of the DGM implies that the mass matrix is always block-diagonal, which translates into an efficient time-marching algorithm. Furthermore, the basis functions can be chosen such that the mass matrix is exactly diagonal. Because, in our implementation, we use Lagrangian shape functions, the local mass matrix \mathbf{M}^+ is not diagonal. However, the matrix is local and has size 4×4 for linear polynomial shape functions, or in general, $N_{DG} \times N_{DG}$, N_{DG} being the local number of degrees of freedom in the tetrahedron. A general description of the basis functions can be found in the book by Hughes (2000) and a description of the basis functions used in DGM in that of Li (2006).

REFERENCES

- Ainsworth, M., P. Monk, and W. Muniz, 2006, Dispersive and dissipative properties of discontinuous Galerkin finite element methods for the second-order wave equation: *Journal of Scientific Computing*, **27**, 5–40, doi: [10.1007/s10915-005-9044-x](https://doi.org/10.1007/s10915-005-9044-x).
- Arnold, D. N., F. Brezzi, B. Cockburn, and L. D. Marini, 2002, Unified analysis of discontinuous Galerkin methods for elliptic problems: *SIAM Journal on Numerical Analysis*, **39**, 1749–1779, doi: [10.1137/S0036142901384162](https://doi.org/10.1137/S0036142901384162).
- Baldassari, C., H. Barucq, H. Calandra, and J. Diaz, 2011, Numerical performances of a hybrid local-time stepping strategy applied to the reverse time migration: *Geophysical Prospecting*, **59**, 907–919, doi: [10.1111/j.1365-2478.2011.00975.x](https://doi.org/10.1111/j.1365-2478.2011.00975.x).
- Bartel, L. C., N. P. Symons, and D. F. Aldridge, 2000, Graded boundary simulation of air/earth interfaces in finite-difference elastic wave modeling: 70th Annual International Meeting, SEG, Expanded Abstracts, 2444–2447.
- Bohlen, T., and E. Saenger, 2006, Accuracy of heterogeneous staggered-grid finite-difference modeling of Rayleigh waves: *Geophysics*, **71**, no. 4, T109–T115, doi: [10.1190/1.2213051](https://doi.org/10.1190/1.2213051).
- Charara, M., A. Vershinin, D. Sabitov, and G. Pekar, 2011, SEM wave propagation in complex media with tetrahedral to hexahedral mesh: 73rd Annual International Conference and Exhibition, EAGE, Extended Abstracts, 41–45.
- Chin-Joe-Kong, M. J. S., W. A. Mulder, and M. van Veldhuizen, 1999, Higher-order triangular and tetrahedral finite elements with mass lumping for solving the wave equation: *Journal of Engineering Mathematics*, **35**, 405–426, doi: [10.1023/A:1004420829610](https://doi.org/10.1023/A:1004420829610).
- Cohen, G., P. Joly, and N. Tordjman, 1993, Construction and analysis of higher order finite elements with mass lumping for the wave equation: in R. Kleinman, T. Angell, D. Colton, F. Santosa, and I. Stakgold, eds., *Second International Conference on Mathematical and Numerical Aspects of Wave Propagation*, SIAM, 152–160.
- Collino, F., T. Fouquet, and P. Joly, 2003, A conservative space-time mesh refinement method for the 1-D wave equation. I. Construction: *Numerische Mathematik*, **95**, 197–221, doi: [10.1007/s00211-002-0446-5](https://doi.org/10.1007/s00211-002-0446-5).
- Courant, R., K. Friedrichs, and H. Lewy, 1928, Über die partiellen Differenzgleichungen der mathematischen Physik: *Mathematische Annalen*, **100**, 32–74, doi: [10.1007/BF01448839](https://doi.org/10.1007/BF01448839).
- Dablain, M. A., 1986, The application of high-order differencing to the scalar wave equation: *Geophysics*, **51**, 54–66, doi: [10.1190/1.1442040](https://doi.org/10.1190/1.1442040).
- De Basabe, J. D., and M. K. Sen, 2010, Stability of the high-order finite elements for acoustic or elastic wave propagation with high-order time stepping: *Geophysical Journal International*, **181**, 577–590, doi: [10.1111/j.1365-246X.2010.04536.x](https://doi.org/10.1111/j.1365-246X.2010.04536.x).
- de la Puente, J., M. Käser, M. Dumbser, and H. Igel, 2007, An arbitrary high order discontinuous Galerkin method for elastic waves on unstructured meshes — IV. Anisotropy: *Geophysical Journal International*, **169**, 1210–1228, doi: [10.1111/j.1365-246X.2007.03381.x](https://doi.org/10.1111/j.1365-246X.2007.03381.x).
- Diaz, J., and M. J. Grote, 2009, Energy conserving explicit local time stepping for second order wave equations: *SIAM Journal on Scientific Computing*, **31**, 1985–2014, doi: [10.1137/070709414](https://doi.org/10.1137/070709414).
- Dumbser, M., and M. Käser, 2006, An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes — II. The three-dimensional isotropic case: *Geophysical Journal International*, **167**, 319–336, doi: [10.1111/j.1365-246X.2006.03120.x](https://doi.org/10.1111/j.1365-246X.2006.03120.x).

- Dumbser, M., M. Käser, and E. F. Toro, 2007, An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes — V. Local time stepping and p -adaptivity: *Geophysical Journal International*, **171**, 695–717, doi: [10.1111/j.1365-246X.2007.03427.x](https://doi.org/10.1111/j.1365-246X.2007.03427.x).
- Epshteyn, Y., and B. Rivière, 2007, Estimation of penalty parameters for symmetric interior penalty Galerkin methods: *Journal of Computational and Applied Mathematics*, **206**, 843–872, doi: [10.1016/j.cam.2006.08.029](https://doi.org/10.1016/j.cam.2006.08.029).
- Etienne, V., J. Chaljub, J. Virieux, and N. Glinsky, 2010, An hp-adaptive discontinuous Galerkin finite elements method for 3-D elastic wave modelling: *Geophysical Journal International*, **183**, 941–962, doi: [10.1111/j.1365-246X.2010.04764.x](https://doi.org/10.1111/j.1365-246X.2010.04764.x).
- Fried, I., and D. S. Malkus, 1975, Finite element mass matrix lumping by numerical integration with no convergence rate loss: *International Journal of Solids and Structures*, **11**, 461–466, doi: [10.1016/0020-7683\(75\)90081-5](https://doi.org/10.1016/0020-7683(75)90081-5).
- Grote, M., A. Schneebeli, and D. Schotzau, 2006, Discontinuous Galerkin finite element method for the wave equation: *SIAM Journal on Numerical Analysis*, **44**, 2408–2431, doi: [10.1137/05063194X](https://doi.org/10.1137/05063194X).
- Grote, M. J., and T. Mitkova, 2010, Explicit local time-stepping for Maxwell's equations: *Journal of Computational and Applied Mathematics*, **234**, 3283–3302, doi: [10.1016/j.cam.2010.04.028](https://doi.org/10.1016/j.cam.2010.04.028).
- Grote, M. J., and D. Schötzau, 2009, Optimal error estimates for the fully discrete interior penalty DG method for the wave equation: *Journal of Scientific Computing*, **40**, 257–272, doi: [10.1007/s10915-008-9247-z](https://doi.org/10.1007/s10915-008-9247-z).
- Hestholm, S., and B. Ruud, 1994, 2D finite-difference elastic wave modelling including surface topography: *Geophysical Prospecting*, **42**, 371–390, doi: [10.1111/j.1365-2478.1994.tb00216.x](https://doi.org/10.1111/j.1365-2478.1994.tb00216.x).
- Hughes, T., 2000, *The finite element method*: Dover Publications.
- Käser, M., and M. Dumbser, 2006, An arbitrary high-order discontinuous Galerkin method for elastic waves on unstructured meshes — I. The two-dimensional isotropic case with external source terms: *Geophysical Journal International*, **166**, 855–877, doi: [10.1111/j.1365-246X.2006.03051.x](https://doi.org/10.1111/j.1365-246X.2006.03051.x).
- Komatitsch, D., and J. Tromp, 1999, Introduction to the spectral-element method for 3-D seismic wave propagation: *Geophysical Journal International*, **139**, 806–822, doi: [10.1046/j.1365-246x.1999.00967.x](https://doi.org/10.1046/j.1365-246x.1999.00967.x).
- Komatitsch, D., and J. P. Vilotte, 1998, The spectral-element method: An efficient tool to simulate the seismic response of 2-D and 3-D geological structures: *Bulletin of the Seismological Society of America*, **88**, 368–392.
- Komatitsch, D., J. P. Vilotte, R. Vai, J. M. Castillo-Covarrubias, and F. J. Sánchez-Sesma, 1999, The spectral element method for elastic wave equations: Application to 2-D and 3-D seismic problems: *International Journal for Numerical Methods in Engineering*, **45**, 1139–1164, doi: [10.1002/\(ISSN\)1097-0207](https://doi.org/10.1002/(ISSN)1097-0207).
- Kononov, A., S. Minisini, E. Zhebel, and W. A. Mulder, 2012, A 3-D tetrahedral mesh generator for seismic problems: 74th Annual International Conference and Exhibition, EAGE, Extended Abstracts, B006.
- Lax, P., and B. Wendroff, 1960, Systems of conservation laws: *Communications on Pure and Applied Mathematics*, **13**, 217–237, doi: [10.1002/\(ISSN\)1097-0312](https://doi.org/10.1002/(ISSN)1097-0312).
- Li, B. Q., 2006, *Discontinuous finite elements in fluid dynamics and heat transfer*: Computational Fluid and Solid Mechanics, Springer.
- Maday, Y., and E. M. Rønquist, 1990, Optimal error analysis of spectral methods with emphasis on non-constant coefficients and deformed geometries: *Computer Methods in Applied Mechanics and Engineering*, **80**, 91–115, doi: [10.1016/0045-7825\(90\)90016-F](https://doi.org/10.1016/0045-7825(90)90016-F).
- Mulder, W. A., 1996, A comparison between higher-order finite elements and finite differences for solving the wave equation: *in* J.-A. Désidéri, P. LeTallec, E. Oñate, J. Périaux, and E. Stein, eds., *Proceedings of the Second ECCOMAS Conference on Numerical Methods in Engineering*, John Wiley & Sons, 344–350.
- Mulder, W. A., 1999, Spurious modes in finite-element discretizations of the wave equation may not be all that bad: *Applied Numerical Mathematics*, **30**, 425–445, doi: [10.1016/S0168-9274\(98\)00078-6](https://doi.org/10.1016/S0168-9274(98)00078-6).
- Orszag, S. A., 1980, Spectral methods for problems in complex geometries: *Journal of Computational Physics*, **37**, 70–92, doi: [10.1016/0021-9991\(80\)90005-4](https://doi.org/10.1016/0021-9991(80)90005-4).
- Owen, S. J., 1998, A survey of unstructured mesh generation technology: *in* *Proceedings of the 7th International Meshing Roundtable*, Argonne National Laboratory, 26–28.
- Patera, A. T., 1984, A spectral element method for fluid dynamics: Laminar flow in a channel expansion: *Journal of Computational Physics*, **54**, 468–488, doi: [10.1016/0021-9991\(84\)90128-1](https://doi.org/10.1016/0021-9991(84)90128-1).
- Piperno, S., 2006, Symplectic local time-stepping in non-dissipative DGTD methods applied to wave propagation problems: *ESAIM: Mathematical Modelling and Numerical Analysis*, **40**, 815–841, doi: [10.1051/m2an:2006035](https://doi.org/10.1051/m2an:2006035).
- Rivière, B., 2008, *Discontinuous Galerkin methods for solving elliptic and parabolic equations: Theory and implementation*: Society for Industrial and Applied Mathematics, *Frontiers in Applied Mathematics*, 35.
- Rønquist, E. M., and A. T. Patera, 1987, A Legendre spectral element method for the Stefan problem: *International Journal for Numerical Methods in Engineering*, **24**, 2273–2299, doi: [10.1002/\(ISSN\)1097-0207](https://doi.org/10.1002/(ISSN)1097-0207).
- Shubin, G. R., and J. B. Bell, 1987, A modified equation approach to constructing fourth order methods for acoustic wave propagation: *SIAM Journal on Scientific and Statistical Computing*, **8**, 135–151, doi: [10.1137/0908026](https://doi.org/10.1137/0908026).
- Taniguchi, T., T. Goda, H. Kasper, and W. Zielke, 1996, Hexahedral mesh generation of complex composite domain, *in* B. K. Soni, J. F. Thompson, J. Häuser, and P. R. Eiseman, eds., *5th International Conference on Numerical Grid Generation in Computational Field Simulations*: Mississippi State University, 699–707.
- Zampieri, E., and L. Pavarino, 2006, An explicit second order spectral element methods for acoustic waves: *Advances in Computational Mathematics*, **25**, 381–401, doi: [10.1007/s10444-004-7626-z](https://doi.org/10.1007/s10444-004-7626-z).
- Zhebel, E., S. Minisini, A. Kononov, and W. A. Mulder, 2012, On the time-stepping stability of continuous mass-lumped and discontinuous Galerkin finite elements for the 3D acoustic wave equation: *ECCOMAS 2012 — European Congress on Computational Methods in Applied Sciences and Engineering*, ebook full papers, 1041–1051.