



HAL
open science

SPECFEM2D-DG, an open-source software modelling mechanical waves in coupled solid-fluid systems: the linearized Navier-Stokes approach

Léo Martire, R. Martin, Q. Brissaud, R. F. Garcia

► **To cite this version:**

Léo Martire, R. Martin, Q. Brissaud, R. F. Garcia. SPECFEM2D-DG, an open-source software modelling mechanical waves in coupled solid-fluid systems: the linearized Navier-Stokes approach. *Geophysical Journal International*, 2022, 228, pp.664-697. 10.1093/gji/ggab308 . insu-03619992

HAL Id: insu-03619992

<https://insu.hal.science/insu-03619992>

Submitted on 16 Mar 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SPECFEM2D-DG, an open-source software modelling mechanical waves in coupled solid–fluid systems: the linearized Navier–Stokes approach

Léo Martire^{1,*}, R. Martin,² Q. Brissaud^{3,†} and R. F. Garcia¹

¹ISAE-SUPAERO, Toulouse 31400, France. E-mail: leo.martire@outlook.com

²OMP-GET, Toulouse, France

³Seismological Laboratory, California Institute of Technology, Pasadena, CA 91125, USA

Received 2021 May 19; in original form 2020 September 22

SUMMARY

We introduce SPECFEM2D-DG, an open-source, time-domain, hybrid Galerkin software modelling the propagation of seismic and acoustic waves in coupled solid–fluid systems. For the solid part, the visco-elastic system from the routinely used SPECFEM2D software is used to simulate linear seismic waves subject to attenuation. For the fluid part, SPECFEM2D-DG includes two extensions to the acoustic part of SPECFEM2D, both relying on the Navier–Stokes equations to model high-frequency acoustics, infrasound and gravity waves in complex atmospheres. The first fluid extension, SPECFEM2D-DG-FNS, was introduced in 2017 by Brissaud, Martin, Garcia, and Komatitsch; it features a nonlinear Full Navier–Stokes (FNS) approach discretized with a discontinuous Galerkin numerical scheme. In this contribution, we focus only on introducing a second fluid extension, SPECFEM2D-DG-LNS, based on the same numerical method but rather relying on the Linear Navier–Stokes (LNS) equations. The three main modules of SPECFEM2D-DG all use the spectral element method (SEM). For both fluid extensions (FNS and LNS), two-way mechanical coupling conditions preserve the Riemann problem solution at the fluid–solid interface. Absorbing outer boundary conditions (ABCs) derived from the perfectly matched layers’ approach is proposed for the LNS extension. The SEM approach supports complex topographies and unstructured meshes. The LNS equations allow the use of range-dependent atmospheric models, known to be crucial for the propagation of infrasound at regional scales. The LNS extension is verified using the method of manufactured solutions, and convergence is numerically characterized. The mechanical coupling conditions at the fluid–solid interface (between the LNS and elastodynamics systems of equations) are verified against theoretical reflection–transmission coefficients. The ABCs in the LNS extension are tested and prove to yield satisfactory energy dissipation. In an example case study, we model infrasonic waves caused by quakes occurring under various topographies; we characterize the acoustic scattering conditions as well as the apparent acoustic radiation pattern. Finally, we discuss the example case and conclude by describing the capabilities of this software. SPECFEM2D-DG is open-source and is freely available online on GitHub.

Key words: Infrasound; Numerical modelling; Computational seismology; Interface waves; Wave propagation.

1 INTRODUCTION

A large variety of mechanical waves propagate across the ground–atmosphere interface of terrestrial planets. Their study is an invaluable tool able to probe the structure of planetary interiors and atmospheres. Understanding the coupling between the fluid and solid parts is a critical issue for imaging sensors deployed at this interface.

*Now at: Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA 91109, USA.

†Now at: Norwegian Seismic Array (NORSAR), Oslo, Norway.

Historically, on Earth, measurement set-ups devoted to the study of the ground-atmosphere system almost always combine seismic and pressure sensors (Shaw & Dines 1905). Recently, such set-ups are now also sent and deployed on other planetary bodies, with the same goal. In November 2018, the *InSight* mission successfully landed on the surface of Mars, featuring a seismo-acoustic payload. On the one hand, the SEIS (Seismic Experiment for Interior Structure) is the most precise planetary seismometer functioning to this day (Lognonné *et al.* 2019, 2020), able to record the ground signature of the faint Martian infrasound (Martire *et al.* 2020). On the other hand, the APSS (Auxiliary Payload Sensor Suite) records almost continuously pressure data, also down to low noise levels (Banfield *et al.* 2019). Mission concepts towards Venus are also being rigorously studied, including aircraft featuring airborne barometers (Stevenson *et al.* 2015; Krishnamoorthy *et al.* 2018; Martire *et al.* 2018; Garcia *et al.* 2021) and therefore requiring a precise understanding of the ground-to-atmosphere couplings.

In order to better understand the processes involved in the propagation of mechanical waves in planetary bodies and atmospheres, numerical simulation solutions are required.

Several techniques already exist. Normal modes (Lognonné *et al.* 1998) and ray tracing (Jones *et al.* 1986; Dessa *et al.* 2005; Blom 2014) rely on approximating the physical or mathematical systems at play, and therefore are very efficient in providing quick solutions. Next, the more complete finite differences/elements/volumes (Jawahar & Kamath 2000; Kherani *et al.* 2012, 2016) or spectral methods (Priolo *et al.* 1994; Seriani & Priolo 1994; Faccioli *et al.* 1996; Komatitsch 1997; Komatitsch & Vilotte 1998; Komatitsch & Tromp 1999; Tromp *et al.* 2008; Madec 2010; Cupillard *et al.* 2012; Gokhberg & Fichtner 2016; Brissaud *et al.* 2017; Terrana *et al.* 2018; Afanasiev *et al.* 2019) are so-called full-wave methods because they do not approximate the mathematical system at play (aside from the discretization step) and allow the retrieval of full time-dependent waveforms. Furthermore, spectral methods benefit from spectral convergence, in which the error is decreasing exponentially rather than algebraically. Finally, pseudospectral methods complement the usual spectral basis by a pseudospectral basis in order to accelerate the computation of some terms (Tessmer & Kosloff 1994; Faccioli *et al.* 1997).

However, not only few numerical methods propose to jointly solve the coupled ground/atmosphere system, but even fewer of those are available as free or open-source software. Among the latter, SPECFEM (Komatitsch 1997; Komatitsch & Vilotte 1998; Komatitsch & Tromp 1999; Peter *et al.* 2011) is an open-source and routinely used code; it accurately models fluid–solid couplings, and is therefore able to correctly simulate wave propagation in bodies of water. However, SPECFEM’s potential fluid formulation is too restrictive to model atmospheres realistically, because it cannot account for winds varying vertically, viscosity, thermal conductivity, or gravity effects. Taking SPECFEM as basis, Brissaud *et al.* (2017) adapted the solid–fluid coupling to a more general solid-atmosphere coupling, and modelled the atmospheric medium more accurately. In this paper, we propose to further extend the SPECFEM software suite to introduce additional realistic features for the modelling of the atmospheric medium.

This contribution aims to present a free, open-source, full-wave, coupled ground-atmosphere, seismic and acoustic wave propagation software. Our approach relies on an hybrid (continuous/discontinuous) Galerkin discretization, and involves the systems of elastodynamics and Navier–Stokes.

The viscoelastic module of the software is exactly the elastodynamics module implemented in SPECFEM (Komatitsch & Vilotte 1998; Komatitsch & Tromp 1999; Tromp *et al.* 2008). For the fluid part, both the Full and Linearized Navier–Stokes (FNS and LNS) systems are implemented in SPECFEM2D-DG. The FNS extension was already described by Brissaud *et al.* (2017), while our present contribution focuses exclusively on the LNS extension.

The LNS system of equations is obtained by suppressing all nonlinearities from the FNS system of equations, while keeping in all the viscous aspects. The LNS equations do not profit from nonlinearities as ‘safety mechanisms’ to dissipate energy near possible critical levels (Booker & Bretherton 1967). However, the choice of static background state (ρ_0 , v_0 , E_0) is completely free and allows for instance the use of range-dependent atmospheric models, which are known to be crucial for the accurate modelling of regional acoustic propagation (Sabatini *et al.* 2019). In addition, the far-field propagation of seismic and acoustic-gravity waves caused by a large variety of geophysical events are essentially linear. Finally, a forward solver is always the first building block to build an inversion framework. This is already well-known in seismology for the computation of sensitivity kernels and general inverse modelling, thanks to a much simpler, self-adjoint operator (Fichtner *et al.* 2006; Gokhberg & Fichtner 2016). Some methodologies and even closed formulations also already exist in the context of fluid mechanics, for instance in the case of the shallow water equations (Sanders & Katopodes 2000; Lai & Monnier 2009), or in the case of the Navier–Stokes equations for subsonic cavity flows (Moret-Gabarro 2009). Such methods could readily be applied to the LNS equations.

Nonetheless, nonlinearities are often necessary for the simulation of shock waves, high-energy events, atmospheric explosions, quakes featuring large ground motions, or the breaking of internal gravity waves (Rogers & Gardner 1980; Afraimovich *et al.* 2001; Stevanović *et al.* 2017). Such applications require the numerical simulation of the FNS equations (Brissaud *et al.* 2017). However, the FNS equations can suffer from spurious oscillations, especially at high Reynolds numbers, possibly caused by the nonlinearities or strong convection, or even the very advective-diffusive character of the equations (Franca & Frey 1992).

These arguments are used in favour of our choice to implement and release both the LNS and FNS extensions in the SPECFEM2D-DG software. As introduced above, this paper shall focus only on the LNS system, its verification, and the implementation of absorbing boundary conditions (BCs). The reader is referred to the work of Brissaud *et al.* (2017) for the FNS system.

The paper is organized as follows. Section 2 presents the governing equations at play. Section 3 details how physical parameters and initial state are initialized. Section 4 presents the spatial and temporal discretization. Section 5 details the BCs used at the computational domain boundary (including the real stretching absorbing BCs), or at interfaces with other materials. Section 6 presents the verification of the

fluid LNS extension using the method of manufactured solutions. Section 7 details the verification of the fluid–solid coupling BC. Finally, an example of application studying the impact of topography on the seismically induced acoustic wavefield is presented in Section 8.

Appendix A presents closed formulas for the reflection–transmission coefficients for waves travelling across a fluid–solid interface. Appendix B proposes additional verification of the software.

2 GOVERNING EQUATIONS

In order to remain more general in all expressions, let d denote the spatial dimension (in practice, either $d = 2$ or $d = 3$). Furthermore, unless otherwise noted, all quantities depend on time ($t \geq 0$) and space ($\mathbf{x} \in \mathbb{R}^d$). However, for the sake of clarity, the notations (t, \mathbf{x}) shall be dropped unless explicitly needed. Likewise, most of the integrals shall not have the integration variable specified (unless needed). Finally, $[\mathbf{A}]_i$ shall denote the i th component of a vector \mathbf{A} , and $[M]_{ij}$ the component (i, j) of a matrix M . We shall heavily rely on the small perturbations hypothesis, which writes for a scalar quantity q :

$$\forall t > 0, \quad \forall \mathbf{x} \in \Omega, \quad |q'(t, \mathbf{x})| \ll |q_0(\mathbf{x})|, \quad (1)$$

That is, the perturbation q' is small when compared to the initial state q_0 .

In order to model the propagation of seismic waves in realistic planetary interiors, we rely on the system of visco-elastodynamics. In order to model the propagation of gravity waves, infrasound, and high-frequency acoustic waves (frequencies above 10^{-5} Hz) in realistic atmospheres, one may rely on the Navier–Stokes equations (Brissaud *et al.* 2017). Finally, mechanical coupling conditions can model the energy transmissions at the ground–atmosphere interface (Brissaud *et al.* 2017).

Following the work of Komatitsch & Vilotte (1998), Komatitsch & Tromp (1999), and Tromp *et al.* (2008), we choose to numerically resolve these systems of equations with the Spectral Finite Element Method.

First, Komatitsch & Vilotte (1998), Komatitsch & Tromp (1999), and Tromp *et al.* (2008) presented the software SPECSEM2D and SPECSEM3D, well-known to accurately model full seismic waveforms in visco-elastic media. Next, Brissaud *et al.* (2017) developed SPECSEM2D-DG-FNS, in which the FNS equations were resolved and coupled to the visco-elastic module from SPECSEM2D. Likewise, in this contribution, we propose to present SPECSEM2D-DG-LNS, which resolves the LNS equations for the atmosphere and couples them to SPECSEM2D's visco-elastic module. These three modules, resolving each one of the three aforementioned systems of equations, are brought together in the software SPECSEM2D-DG.

2.1 Solid system: visco-elastodynamics

Consider a solid domain, heterogeneous, linear, and viscoelastic. Under the small perturbations hypothesis, eq. (1), the linear wave equations known as the elastodynamics' system can be written as (Komatitsch & Tromp 1999):

$$\rho^s \partial_t^2 \mathbf{u}^s = \nabla \cdot \Sigma^s + \mathbf{F}^s, \quad (2)$$

where ρ^s is the solid density, \mathbf{u}^s is the solid displacement, Σ^s is the solid stress tensor, and $\mathbf{F}^s \in \mathbb{R}^d$ are the external forces applied on the solid.

In a general elastic and anisotropic medium, Σ^s is simply determined by Hooke's law. In attenuating media however, this law must be modified to take into account the entire strain history. The attenuation phenomena can be approximated by a series of $L \in \mathbb{N}$ standard linear solids (Liu *et al.* 1976). One can thus define (Komatitsch & Tromp 1999; Blanc *et al.* 2016, and references therein):

$$\begin{aligned} \Sigma^s &= (C^U : \epsilon) - \sum_{l=1}^L \mathbf{R}^l, \\ \epsilon &= \frac{1}{2} (\nabla \mathbf{u} + (\nabla \mathbf{u})^T), \\ \forall 1 \leq l \leq L, \quad \partial_t \mathbf{R}^l &= \frac{1}{\tau^{sl}} (-\mathbf{R}^l + \delta c^l : \epsilon), \end{aligned} \quad (3)$$

where ϵ is the strain tensor, L is the number of relaxation mechanisms, the $(\mathbf{R}^l)_{1 \leq l \leq L}$ are the relaxation auxiliary variables, and τ^{sl} are the stress relaxation time (chosen to be the same for all components of the tensor) for the mechanism l . We refer to Komatitsch & Tromp (1999) for the complete description of those parameters; as well as the definition of the modulus defect δc^l and of the unrelaxed moduli C^U .

These governing equations are the ones resolved by the seismological software SPECSEM (Komatitsch & Vilotte 1998; Komatitsch & Tromp 1999; Tromp *et al.* 2008). In SPECSEM2D-DG, either the FNS equations (Brissaud *et al.* 2017) or the LNS equations (this contribution) are coupled to the system of visco-elastodynamics.

2.2 Fluid system: LNS equations

The LNS equations are derived from the FNS equations (e.g. Landau & Lifshitz 1987; Brissaud *et al.* 2017) under the assumption of eq. (1). They write (e.g. Arina 2016):

$$\partial_t \mathbf{U} + \nabla \cdot \Sigma^c(\mathbf{U}) = \nabla \cdot \Sigma^d(\mathbf{U}) + \mathbf{G}(\mathbf{U}) + \mathbf{F}, \quad (4)$$

where:

$$U(t, \mathbf{x}) = \begin{pmatrix} \rho' \\ \rho_0 \mathbf{v}' \\ E' \end{pmatrix}, \quad (5)$$

$$\Sigma^c(U(t, \mathbf{x})) = \begin{pmatrix} \rho_0 \mathbf{v}' + \rho' \mathbf{v}_0 \\ \rho_0 \mathbf{v}_0 \otimes \mathbf{v}' + p' \mathbf{I} \\ (E_0 + p_0) \mathbf{v}' + (E' + p') \mathbf{v}_0 \end{pmatrix}, \quad (6)$$

$$\Sigma^d(U(t, \mathbf{x})) = \begin{pmatrix} 0 \\ \Sigma_v(\mathbf{v}') \\ \Sigma_v(\mathbf{v}_0) \mathbf{v}' + \Sigma_v(\mathbf{v}') \mathbf{v}_0 + \kappa \nabla T' \end{pmatrix}, \quad (7)$$

$$\mathbf{G}(U(t, \mathbf{x})) = \begin{pmatrix} 0 \\ \rho' \mathbf{g} - \rho' \partial_t(\mathbf{v}_0) - ((\rho_0 \mathbf{v}' + \rho' \mathbf{v}_0) \cdot \nabla) \mathbf{v}_0 \\ \mathbf{g} \cdot (\rho_0 \mathbf{v}' + \rho' \mathbf{v}_0) \end{pmatrix}, \quad (8)$$

$$\mathbf{F}(t, \mathbf{x}) = \begin{pmatrix} F_{\rho'} \\ F_{\rho_0 \mathbf{v}'} \\ F_{E'} \end{pmatrix}. \quad (9)$$

$U(t, \mathbf{x})$ is the state vector, encoding the perturbations on constitutive variables: $\rho'(t, \mathbf{x})$ the density perturbation, $\mathbf{v}'(t, \mathbf{x})$ the velocity perturbation, and $E'(t, \mathbf{x})$ the energy density perturbation. $(\rho_0(\mathbf{x}), \mathbf{v}_0(\mathbf{x}), E_0(\mathbf{x}))$ are their static counterparts, which do not depend on time. Similarly, p' is the pressure perturbation and p_0 is the static pressure. Σ^c gathers the convective terms of the stress tensor whereas Σ^d collects its diffusive terms. \otimes denotes the tensor product. \mathbf{I} is the identity matrix. $\Sigma_v(t, \mathbf{x})$ is the Navier–Stokes viscous stress tensor, detailed below in eq. (10). $\kappa(\mathbf{x})$ is the thermal conductivity of the fluid. $T'(t, \mathbf{x})$ is the temperature perturbation. $\mathbf{g}(\mathbf{x})$ is the full acceleration of gravity. $F_{\rho'}(t, \mathbf{x})$, $F_{\rho_0 \mathbf{v}'}(t, \mathbf{x})$, and $F_{E'}(t, \mathbf{x})$ are the source terms related to the constitutive variables.

The viscous Navier–Stokes stress tensor Σ_v writes:

$$\Sigma_v(\mathbf{v}) := \lambda (\nabla \cdot \mathbf{v}) \mathbf{I} + \mu (\nabla \mathbf{v} + (\nabla \mathbf{v})^T), \quad (10)$$

where $\mu(\mathbf{x})$ is the dynamic viscosity, and $\lambda(\mathbf{x})$ is the second viscosity (in general, $\lambda = 2\mu/3$). Note that Σ_v is linear in \mathbf{v} . Furthermore, let δ_m denote the first-order momentum perturbation:

$$\delta_m(t, \mathbf{x}) := \rho_0 \mathbf{v}' + \rho' \mathbf{v}_0. \quad (11)$$

For the sake of lightweight notations, let $(\mathfrak{N}^0, \mathfrak{N}')$ denote the components of the viscous tensor following the small perturbations' decomposition $(\Sigma_v(\mathbf{v}) = \mathfrak{N}^0 + \mathfrak{N}')$:

$$\mathfrak{N}^0 := \Sigma_v(\mathbf{v}_0), \quad \mathfrak{N}' := \Sigma_v(\mathbf{v}'). \quad (12)$$

Finally, for later use, let Σ^f denote the inviscid LNS fluid stress tensor:

$$\Sigma^f := \rho_0 \mathbf{v}_0 \otimes \mathbf{v}' + p' \mathbf{I}. \quad (13)$$

The perfect gas hypothesis (PGH) allows to write closing relations linking temperature, pressure, and energy density for the fluid side of our problem. The temperature of a perfect gas is defined as:

$$T(\rho, \mathbf{v}, E) := \frac{1}{c_V} \left(\frac{E}{\rho} - \frac{1}{2} \|\mathbf{v}\|_2^2 \right), \quad (14)$$

the total gas pressure verifies:

$$p(\rho, \mathbf{v}, E) := (\gamma - 1) \left(E - \frac{\rho}{2} \|\mathbf{v}\|_2^2 \right), \quad (15)$$

and, the gas' total energy density writes:

$$E(\rho, \mathbf{v}, p) := \frac{p}{\gamma - 1} + \frac{\rho}{2} \|\mathbf{v}\|_2^2. \quad (16)$$

The corresponding perturbed quantities naturally verify:

$$p'(\rho_0 + \rho', \mathbf{v}_0 + \mathbf{v}', E_0 + E') = p(\rho_0 + \rho', \mathbf{v}_0 + \mathbf{v}', E_0 + E') - p(\rho_0, \mathbf{v}_0, E_0) = p(\rho_0 + \rho', \mathbf{v}_0 + \mathbf{v}', E_0 + E') - p_0, \quad (17)$$

and:

$$E'(\rho_0 + \rho', \mathbf{v}_0 + \mathbf{v}', p_0 + p') = E(\rho_0 + \rho', \mathbf{v}_0 + \mathbf{v}', p_0 + p') - E(\rho_0, \mathbf{v}_0, E_0) = E(\rho_0 + \rho', \mathbf{v}_0 + \mathbf{v}', p_0 + p') - E_0, \quad (18)$$

in which p and E are the previously defined eq. (15), and eq. (16).

Kinetic and potential energies in a fluid are given by Cremer (1990) and Maess *et al.* (2006, eq. 23):

Table 1. Description of the different models for static fluid parameters. Top part of the table (\mathbf{g} , γ , μ , κ , ρ_0 , \mathbf{v}_0 , p_0 , E_0 , T_0): description of the values taken by the background parameters as function of the chosen model. Bottom part (c , N^2): other quantities of interest as function of the aforementioned parameters. Barred quantities ($\bar{\rho}$, \bar{H} , etc.) are constant parameters \mathbf{g} is gravity, $[\mathbf{g}]_d$ corresponds to its vertical component, γ is the adiabatic ratio, \bar{c}_p is the isobaric (constant pressure) specific heat capacity, \bar{c}_v is the isochoric (constant volume) specific heat capacity, μ is the dynamic viscosity, κ is the thermal conductivity, ρ_0 is the fluid's density, \mathbf{v}_0 is the velocity field, p_0 is the pressure, E_0 is the energy density, T_0 is the temperature, c is the speed of sound, and N^2 is the Brunt-Väisälä frequency.

| Model: | Analytic isobaric | Analytic isothermal | User-defined |
|----------------------------|---|---|--|
| Best used for: | Very short range | Atmospheric short range | Atmospheric long range |
| $\mathbf{g}(\mathbf{x})$ | $\mathbf{0}$ | $\bar{\mathbf{g}}$ | User-defined |
| $\gamma(\mathbf{x})$ | $\frac{\bar{c}_p}{\bar{c}_v}$ | $\frac{\bar{c}_p}{\bar{c}_v}$ | User-defined |
| $\mu(\mathbf{x})$ | $\bar{\mu}$ | $\bar{\mu}$ | User-defined |
| $\kappa(\mathbf{x})$ | $\bar{\kappa}$ | $\bar{\kappa}$ | User-defined |
| $\rho_0(\mathbf{x})$ | $\bar{\rho}_0$ | $\bar{\rho}_0 e^{-z/\bar{H}}$ | User-defined |
| $\mathbf{v}_0(\mathbf{x})$ | $\bar{\mathbf{v}}_0$ | $\bar{\mathbf{v}}_0$ | User-defined |
| $p_0(\mathbf{x})$ | $\frac{\bar{c}^2 \rho_0(\mathbf{x})}{\gamma(\mathbf{x})}$ | $\rho_0(\mathbf{x}) [\bar{\mathbf{g}}]_d \bar{H}$ | User-defined |
| $E_0(\mathbf{x})$ | Use eq. (16) | Use eq. (16) | Use eq. (16) |
| $T_0(\mathbf{x})$ | Use eq. (14) | $\frac{[\bar{\mathbf{g}}]_d \bar{H}}{\bar{c}_p - \bar{c}_v}$ | Use eq. (14) |
| $c(\mathbf{x})$ | \bar{c} | $\sqrt{\frac{\gamma p_0}{\rho_0}} = \sqrt{\gamma [\bar{\mathbf{g}}]_d \bar{H}}$ | $\sqrt{\frac{\gamma(\mathbf{x}) p_0(\mathbf{x})}{\rho_0(\mathbf{x})}}$ |
| $N^2(\mathbf{x})$ | $\frac{(\gamma-1)[\bar{\mathbf{g}}]_d^2}{\bar{c}^2}$ | $\frac{(\gamma-1)[\bar{\mathbf{g}}]_d^2}{c(\mathbf{x})^2}$ | $\frac{(\gamma(\mathbf{x})-1)[\mathbf{g}(\mathbf{x})]_d^2}{c(\mathbf{x})^2}$ |

$$E_K = \int_{\Omega} \frac{\rho \|\mathbf{v}\|_2^2}{2}, \quad (19)$$

$$E_P = \int_{\Omega} \frac{p^2}{2\rho c^2}. \quad (20)$$

The kinetic energy perturbation is thus given by:

$$E'_K = E_K - E_{K,0}. \quad (21)$$

For an ideal gas, $c^2 = \frac{\gamma p}{\rho}$, and thus $E_P = \int_{\Omega} \frac{p'}{2\gamma}$. Hence, the potential energy perturbation is given by:

$$E'_P = E_P - E_{P,0} = \int_{\Omega} \frac{p'}{2\gamma}. \quad (22)$$

3 STATIC FLUID PARAMETERS

Since we consider the LNS equations (eq. 4), we must also clearly define the static (or *background*, or *initial*) parameters. Those are quantities assumed constant with time, by opposition to constitutive or auxiliary variables. They are the physical parameters (gravity \mathbf{g} , adiabatic ratio γ , dynamic viscosity μ , thermal conductivity κ), as well as the initial state (ρ_0 , \mathbf{v}_0 , E_0).

In order to initialize them, three atmospheric models are implemented. The first two (*isobaric* and *isothermal*) are analytical and stratified (i.e. range-independent or 1-D), whereas the last one (*user-defined*) can be range-dependent (2-D). Table 1 presents the values taken by background parameters as function of the model used, and gives the value taken by other quantities of interest. The models are described in more details below.

The *isobaric* model is built around the assumption that pressure is homogeneous across the whole computational domain. It would be non-physical to allow gravity to act on perturbations, while the static state remains unchanged. Therefore, we require the static model to be hydrostatic, and consequently we disable gravity. It is thus advised for acoustic applications with very short range, in which gravity plays no role and density can be considered constant.

The *isothermal* model is based on the assumption that temperature is constant. Keeping non-zero gravity, the hydrostatic condition $\partial_z p(z) = -\rho(z) \mathbf{g}$ must be verified, which leads to the well-known law of atmospheres, that is the exponential decrease in density and pressure following $p(z) = p(0) \exp(-z/H)$ with H a scale height. The isothermal model can be used for accurate vertical atmospheric propagation when temperature can be considered constant (Martire *et al.* 2018). This assumption can however become too restrictive, especially in realistic set-ups. For very high altitude domains, the constant gravity assumption is no longer valid. For example, note that gravity drops by 1.2 per cent between the surface and 40 km altitude on Earth.

Finally, the user is free to define the last and most general *user-defined* model. Typically, external atmospheric models can be easily imported (such as MSISE (Hedin 1991), HWM (Drob *et al.* 2015), ECMWF (Molteni *et al.* 1996), etc.). We recall that the FNS implementation requires all models to be range-independent in order not to generate spurious perturbations (Brissaud *et al.* 2017); to the contrary, the LNS implementation allows lateral variations of parameters and therefore range-dependent atmospheric models. Nonetheless, even if the LNS

method allows to insert any model, the stability of the computation has been tested only for reasonably small perturbations of stable models - in other words, the models should not deviate too far from stable ones.

The only caveat is that it is the responsibility of the user to make sure the imported model is physically acceptable, that is such that model verifies the static FNS equations:

$$\begin{cases} \nabla \cdot (\rho_0 \mathbf{v}_0) & = 0, \\ \nabla \cdot (\rho_0 \mathbf{v}_0 \otimes \mathbf{v}_0 + p_0 \mathbf{I}) & = \nabla \cdot \Sigma_v(\mathbf{v}_0) + \rho_0 \mathbf{g}, \\ \nabla \cdot ((E_0 + p_0) \mathbf{v}_0) & = \Sigma_v(\mathbf{v}_0) \mathbf{v}_0 + \kappa \nabla T_0 + \rho_0 \mathbf{g} \cdot \mathbf{v}_0. \end{cases} \quad (23)$$

The ability to load range-dependent atmospheric models can be used to model acoustic propagation perturbed by other physical phenomena, provided their temporal variations are slow compared to the overlaying acoustic waves, for instance gravity waves (Sabatini *et al.* 2019).

Since both the isobaric and isothermal models are range-independent (1-D), they can be used as initial conditions for the FNS module. If one sets $\mu = \kappa = \mathbf{g} = 0$, using the isobaric model with the LNS system simplifies to the classical linear inviscid adiabatic case – or linear Euler equations. If, further, the static field is chosen to be incompressible ($\nabla \mathbf{v}_0 = 0$) and if the model is isobaric ($\nabla \rho_0 = 0, \nabla p_0 = 0$), one obtains the linear system of acoustics (Benzoni-Gavage *et al.* 2003). Likewise, if one chooses $\mu = \kappa = \mathbf{g} = 0$ within the isobaric model, the FNS equations reduces to the full Euler equations (Benzoni-Gavage & Serre 2007).

4 SPATIAL AND TEMPORAL DISCRETIZATION

The numerical method employed for the solid part is already presented in (e.g. Komatitsch 1997; Komatitsch & Tromp 1999), thus it will not be described in details here; the key point is that this implementation uses a continuous Galerkin approach. Just as the implementation of the FNS module (Brissaud *et al.* 2017), the spatial discretization of the fluid LNS system eq. (4) relies on a discontinuous Galerkin approach. All the modules (resolving the solid part and/or the fluid part) are marched in time using an explicit fourth-order low-storage Runge–Kutta (RK) scheme. This section aims to cover both of these numerical methods.

4.1 Spatial discretization

The choice of spatial discretization is that of the Discontinuous Galerkin Spectral Element Method (DGSEM). For the fluid part, it is applied to the weak formulation of the system of equations presented in eq. (4).

Since DG methods are inherently local to the elements, the parallelization of such methods is straightforward. Furthermore, when using weak formulations, the coupling between elements only occurs through adjacent edges (faces in 3-D). Because of that, each quadrilateral element in only has to communicate edge information to four processors at most (six in 3-D). DG parallel implementation can even present super-linear speed-up in certain cases due to cache effects (Baggag *et al.* 1999; Altmann *et al.* 2013). Some studies proved that DG methods showed better performances than CG methods in terms of error level for a given computational cost, and/or on large-scale parallel machines (see e.g. Kubatko *et al.* (2008) for the shallow-water equations). Several more precise reasons justifying the use of the DG method rather than the CG method are detailed in Brissaud (2017)’s Section 4.1.2.

The spectral element method has already proven to be powerful in the field of computational fluid dynamics (Patera 1984; Fischer 1990; Yoon & Chung 1996; Giraldo & Restelli 2008; Krank *et al.* 2017). While large-eddy simulations, Reynolds-averaged Navier–Stokes, or fully turbulent flow are of little interest to us, note that application to environmental flows are already successful (Giraldo & Restelli 2008; Brissaud *et al.* 2017). To go further, a comprehensive guide to the high-order spectral element methods is presented by Deville *et al.* (2002) or Solin *et al.* (2003).

Komatitsch (1997) applied the spectral element method to the equations of elastodynamics using Lagrange polynomials associated to Gauss–Lobatto–Legendre¹ (GLL) integration points as basis functions. A fundamental consequence of this particular choice is that the system’s mass matrix is purely diagonal by construction, which makes its inversion trivial. With N_G GLL points, the Gauss quadrature on reference elements is exact for polynomials of order up to $2N_G - 1$ (Deville *et al.* 2002; Funaro 2008), and remains accurate for deformed elements (Maday & Rønquist 1990).

Thus, we follow an approach very close to that of Komatitsch (1997) for our choice of discretization. In the following, we try to remain general in the spatial domain dimension d , but explicit expressions are provided in the 2-D ($d = 2$) case, for the sake of readability. In the remaining of this paper, we denote N_G the chosen number of GLL points. We typically chose $N_G = 5$, ensuring fourth-order accuracy, but the method can be generalized to any other order.

The whole physical domain of interest D is discretized in non-overlapping elements (quadrilaterals in 2-D and hexahedra in 3-D):

$$D = \bigcup_{k=1}^{N_{el}} D_k, \quad (24)$$

¹The GLL points on $[-1, 1]$ are defined as the roots of the polynomial $x \mapsto (1 - x^2) L'_{N_G}(x)$ (Komatitsch 1997, section 3.2.2), where L_{N_G} is the Legendre polynomial of order N_G defined on $[-1, 1]$ (see e.g. Solin *et al.* 2003, section 1.2.2).

where N_{el} is the total number of elements, and D_e is the e th individual element.

Consider now an individual element Ω . Ω is brought back to a reference element $\Lambda = [-1, 1]^d$ (the unit cube of \mathbb{R}^d) by using a differentiable and invertible mapping (Komatitsch 1997):

$$\mathbf{x} = J_{\Omega}(\boldsymbol{\xi}), \quad (25)$$

where $\mathbf{x} \in \Omega$ are the physical coordinates ($\mathbf{x} = (x, z)$ in 2-D), and $\boldsymbol{\xi} \in \Lambda$ is the reference coordinates ($\boldsymbol{\xi} = (\xi, \eta)$ in 2-D). This enables dealing uniquely with unstructured meshes, and helps define uniquely the polynomial basis. In the following, the Jacobian of J_{Ω} is denoted J_{Ω} . The generic functions $q \in \Omega$ and $q^{\Lambda} \in \Lambda$ are linked by this mapping through:

$$q = q^{\Lambda} \circ J_{\Omega}. \quad (26)$$

The approximation is taken using the polynomial basis P_{N_G} composed of the tensorized (in d dimensions) Lagrange interpolants at the GLL points on each element, that is $(h_{i,j})_{1 \leq i,j \leq N_G}$ in 2-D. This representation ensures spectral convergence (Deville *et al.* 2002; Canuto *et al.* 2007).

Consider the prototype hyperbolic equation under the form:

$$\partial_t q + \nabla \cdot \boldsymbol{\Sigma} = G + F, \quad (27)$$

where q represents the quantity of interest, $\boldsymbol{\Sigma}$ represents the stress applied to it, G represents zeroth-order terms, and F represents the sources. Every equation in the LNS fluid system (eq. 4) can readily be put under the form of the prototype equation (eq. 27). For each of those prototype equations, we assume for all our applications that q , $(\boldsymbol{\Sigma}_i)_{1 \leq i \leq d}$, G , and F are all sufficiently smooth on every element Ω . In other words, we assume that they all lie in:

$$V = \{f \in L^2(D) : f|_{D_k} = f^{\Lambda} \circ J_{D_k}, f^{\Lambda} \in P_{N_G}, 1 \leq k \leq N_{el}\}. \quad (28)$$

In the spectral element method, a subset of the finite elements methods, each prototype equation (eq. 27) is transformed into a weak formulation by multiplication with a sufficiently smooth test function ϕ and integration over an element Ω . Straightforward calculus yields:

$$\langle \partial_t q, \phi \rangle_{\Omega} + \langle \boldsymbol{\Sigma} \cdot \mathbf{n}, \phi \rangle_{\partial\Omega} - a(\boldsymbol{\Sigma}, \phi) = \langle G + F, \phi \rangle_{\Omega}, \quad (29)$$

where the integrals in eq. (29) are defined in the following general form:

$$\langle f, g \rangle_{\Omega} := \int_{\Omega} f g := \int_{\Omega} f(\mathbf{x}) g(\mathbf{x}) \, d\Omega, \quad (30)$$

$$\langle f, g \rangle_{\partial\Omega} := \int_{\partial\Omega} f g := \int_{\partial\Omega} f(\mathbf{x}) g(\mathbf{x}) \, d\Gamma, \quad (31)$$

and:

$$a(\mathbf{f}, g) := \int_{\Omega} \mathbf{f} \cdot \nabla g := \int_{\Omega} \mathbf{f}(\mathbf{x}) \cdot \nabla g(\mathbf{x}) \, d\Omega. \quad (32)$$

The next sections (Sections 4.1.1–4.1.4) present sequentially: the discretization of the volume integrals (eqs 30 and 32), the discretization of the surface integrals (eq. 31), the matrix assembling over the whole domain, and the auxiliary tensors employed to simplify the formulation.

4.1.1 Discretization of volume integrals

The volume integrals containing no derivative of the test functions, under the form eq. (30), are readily discretized using the mapping J defined in eq. (25) and quadrature at the GLL points with quadrature weights $(w_i)_{1 \leq i \leq N_G}$ (Komatitsch 1997). In 2-D:

$$\langle f, \phi \rangle_{\Omega} \simeq \langle f, \phi \rangle_{\Omega}^{N_G} := \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} f(\xi_i, \eta_j) |J_{ij}| w_i w_j, \quad (33)$$

where $J_{ij} = J(\xi_i, \eta_j)$ is the Jacobian of J evaluated at point (ξ_i, η_j) .

The bilinear form a defined in eq. (32) needs more work. First, the same change of variables J and quadrature yield:

$$a(\boldsymbol{\Sigma}, \phi) = \int_{\Lambda} (\boldsymbol{\Sigma} \cdot \nabla \phi) |J| \, d\Lambda \simeq \sum_{i,j=1}^{N_G} \boldsymbol{\Sigma}(\xi_i, \eta_j) \cdot \nabla \phi(\xi_i, \eta_j) |J_{ij}| w_i w_j. \quad (34)$$

Then, one may express $\nabla \phi$ in terms of Lagrange interpolants $(h_i)_{0 \leq i \leq N_G}$ (introduced alongside eq. 37 following Komatitsch 1997) by using the chain rule and the mapping J between Ω and Λ . For the sake of simplicity, let for the rest of this section $d = 2$. The developments trivially generalize to $d = 3$ dimensions, but are much more heavy in terms of notations. Denote the first spatial direction $x_1 = x$, and the second $x_2 = z$. The reader is advised to refer to Komatitsch (1997 e.g. appendix 3.A) for a more complete description of the methods of discretization developed here.

Remark that $\nabla\phi = (\partial_x\phi, \partial_z\phi)^T$. The chain rule yields:

$$\begin{cases} \partial_x\phi = (\partial_\xi\phi)(\partial_x\xi) + (\partial_\eta\phi)(\partial_x\eta), \\ \partial_z\phi = (\partial_\xi\phi)(\partial_z\xi) + (\partial_\eta\phi)(\partial_z\eta), \end{cases} \quad (35)$$

where $\xi(x, z)$ and $\eta(x, z)$ are the form functions involved in the mapping defined in eq. (25). Denote $\xi_x = \partial_x\xi$, $\xi_z = \partial_z\xi$, $\eta_x = \partial_x\eta$, and $\eta_z = \partial_z\eta$ to lighten notations. The gradient of ϕ rewrites:

$$\nabla\phi = \begin{pmatrix} \xi_x\partial_\xi\phi + \eta_x\partial_\eta\phi \\ \xi_z\partial_\xi\phi + \eta_z\partial_\eta\phi \end{pmatrix}. \quad (36)$$

Next, the discretization process on the GLL points leads ϕ to rewrite ϕ^h , defined as:

$$\phi^h(\xi, \eta) = \sum_{p=1}^{N_G} \sum_{q=1}^{N_G} h_p(\xi) h_q(\eta) \phi_{pq}, \quad (37)$$

by definition in the discretized space, where h_r is the Lagrange interpolant at GLL point ξ_r (or η_r). For the sake of clarity, it is easier to now drop all ‘ h ’ superscripts for the remainder of this section. By differentiation, eq. (37) yields:

$$\begin{cases} \partial_\xi\phi(\xi, \eta) = \sum_{p=1}^{N_G} \sum_{q=1}^{N_G} \phi_{pq} (\partial_\xi h_p(\xi)) h_q(\eta), \\ \partial_\eta\phi(\xi, \eta) = \sum_{p=1}^{N_G} \sum_{q=1}^{N_G} \phi_{pq} h_p(\xi) (\partial_\eta h_q(\eta)). \end{cases} \quad (38)$$

Since the Lagrange interpolant $h_r(\xi_i) = \delta_{ri}$ by definition, eq. (38) evaluated at (ξ_i, η_j) simplifies to:

$$\begin{cases} \partial_\xi\phi(\xi_i, \eta_j) = \sum_{p=1}^{N_G} \phi_{pj} \partial_\xi h_p(\xi_i), \\ \partial_\eta\phi(\xi_i, \eta_j) = \sum_{q=1}^{N_G} \phi_{iq} \partial_\eta h_q(\eta_j). \end{cases} \quad (39)$$

Now, combining eq. (36) and eq. (39), the bilinear form eq. (34) rewrites as:

$$a(\Sigma, \phi) \simeq \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} |J_{ij}| w_i w_j \Sigma(\xi_i, \eta_j) \cdot \begin{pmatrix} \xi_{x,ij} \sum_{k=1}^{N_G} \phi_{kj} \partial_\xi h_k(\xi_i) + \eta_{x,ij} \sum_{k=1}^{N_G} \phi_{ik} \partial_\eta h_k(\eta_j) \\ \xi_{z,ij} \sum_{k=1}^{N_G} \phi_{kj} \partial_\xi h_k(\xi_i) + \eta_{z,ij} \sum_{k=1}^{N_G} \phi_{ik} \partial_\eta h_k(\eta_j) \end{pmatrix}, \quad (40)$$

which in turn rewrites under the more compact form:

$$a(\Sigma, \phi) \simeq \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} \sum_{k=1}^{N_G} \left[|J_{ij}| w_i w_j (\Sigma_{x,ij} \xi_{x,ij} + \Sigma_{z,ij} \xi_{z,ij}) \phi_{kj} \partial_\xi h_k(\xi_i) + |J_{ij}| w_i w_j (\Sigma_{x,ij} \eta_{x,ij} + \Sigma_{z,ij} \eta_{z,ij}) \phi_{ik} \partial_\eta h_k(\eta_j) \right]. \quad (41)$$

Renaming indices ($i \leftrightarrow k$ in the first term, $j \leftrightarrow k$ in the second term) to make ϕ_{ij} explicitly appear (and thus put the relation under the right vector form) yields:

$$a(\Sigma, \phi) \simeq \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} \phi_{ij} \sum_{k=1}^{N_G} (|J_{kj}| w_k w_j [\Sigma \cdot \nabla \xi]_{kj} \partial_\xi h_i(\xi_k) + |J_{ik}| w_i w_k [\Sigma \cdot \nabla \eta]_{ik} \partial_\eta h_j(\eta_k)), \quad (42)$$

where the notations $[\Sigma \cdot \nabla \xi]_{pq} = \Sigma_{x,pq} \xi_{x,pq} + \Sigma_{z,pq} \xi_{z,pq}$ and $[\Sigma \cdot \nabla \eta]_{pq} = \Sigma_{x,pq} \eta_{x,pq} + \Sigma_{z,pq} \eta_{z,pq}$ (for all $1 \leq p, q \leq N_G$) were adopted in order to lighten the expressions. Finally, recalling $\phi_{ij} = 1$ yields the final expression for the bilinear form:

$$a(\Sigma, \phi) \simeq \sum_{i=1}^{N_G} \sum_{j=1}^{N_G} \sum_{k=1}^{N_G} (|J_{kj}| w_j [\Sigma \cdot \nabla \xi]_{kj} h_{ki}^p + |J_{ik}| w_i [\Sigma \cdot \nabla \eta]_{ik} h_{kj}^p), \quad (43)$$

where $h_{ki}^p = w_k \partial_\xi h_i(\xi_k)$, and $h_{kj}^p = w_k \partial_\eta h_j(\eta_k)$, in which h_q denotes the q th Lagrange interpolant. eq. (43) is only valid for a 2-D discretization; the fact that three sums are featured originates from the useful factorization of the terms stemming from the GLL discretization (see eqs 40 and 41).

4.1.2 Discretization of surface integrals and fluxes’ computation

The surface integral eq. (31) involved in the weak formulation eq. (29) is the integral over the element border $\partial\Omega$ of a flux $\Sigma \cdot \mathbf{n}$. Since the discontinuous Galerkin method is used, a Riemann problem must be solved at each given interface in order to accurately compute the surface term $\langle \Sigma \cdot \mathbf{n}, \phi \rangle_{\partial\Omega}$. This problem has the prototype form, on the interface Γ separating some elements Ω_- and Ω_+ :

$$\begin{cases} \partial_t U + \nabla \cdot \Sigma(U) = \mathcal{S}, \\ U(\mathbf{x}) = \begin{cases} U_- \quad \forall \mathbf{x} \in \Omega_- \setminus \Gamma, \\ U_+ \quad \forall \mathbf{x} \in \Omega_+ \setminus \Gamma. \end{cases} \end{cases} \quad (44)$$

For simple linear conservative hyperbolic systems, the computation of the numerical flux is straightforward (Hesthaven & Warburton 2008, section 2.4) and essentially relies on the well-known method of characteristics (Thompson 1990; Touloupoulos & Ekaterinaris 2011;

Mengaldo *et al.* 2014). However, this methods can only be applied to conservative hyperbolic forms, for which an eigenvalue decomposition can be obtained (Poinso & Lele 1992). This is not the case for our fluid equations, eq. (4), because they include second-order derivatives – viscous terms – and non-trivial zero-order terms. For simplicity and sufficient accuracy, the Rusanov approximate Riemann solver is used. That is, the flux $\Sigma(\mathbf{U}) \cdot \mathbf{n}$ is approximated as:

$$\Sigma(\mathbf{U}) \cdot \mathbf{n} \simeq \Sigma_{\text{Rus}} := \frac{1}{2} ((\Sigma(\mathbf{U}_+) + \Sigma(\mathbf{U}_-)) \cdot \mathbf{n} - \lambda_{\max}(\mathbf{U}_+ - \mathbf{U}_-)), \quad (45)$$

where \mathbf{n} is the normal vector at the considered interface (pointing from the ‘–’ side to the ‘+’ side), Σ_{Rus} is the approximated Rusanov flux, $\Sigma(\mathbf{U}_+)$ (resp. $\Sigma(\mathbf{U}_-)$) is the value of the flux taken on the ‘+’ (resp. ‘–’) side, $(\mathbf{U}_+ - \mathbf{U}_-)$ is the state’s jump across the elements’ boundary, and λ_{\max} is the largest eigenvalue of the Jacobian matrix of $\Sigma(\mathbf{U})$ taken either on the ‘–’ side or ‘+’ side. In practice, we choose λ_{\max} as the maximum linearized acoustic wave speed (Hesthaven & Warburton 2008):

$$\lambda_{\max} = \max_{s \in \{''-, ''+, ''\}} (|\mathbf{v}(\mathbf{x}_s) \cdot \mathbf{n}| + c(\mathbf{x}_s)), \quad (46)$$

with $\mathbf{v} = \mathbf{v}_0 + \mathbf{v}'$, and $c(\mathbf{x}_s)$ the local speed of sound.

In the discontinuous Galerkin spectral element method (in fact in any finite elements method), one can partition the boundary $\partial\Omega$ of each element into n_s sub-surfaces: in the case of quadrilateral elements (2-D) $n_s = 4$, and in the case of hexahedral elements (3-D) $n_s = 6$. Thus, the surface integral over $\partial\Omega$ becomes n_s surface sub-integrals. Each of those can be computed through Gaussian quadrature using the GLL points. Finally and in 2-D, the two-steps discretization of the surface integral eq. (31) involved in eq. (29) writes:

$$\langle \Sigma \cdot \mathbf{n}, \phi \rangle_{\partial\Omega} \simeq \langle \Sigma_{\text{Rus}}, \phi \rangle_{\partial\Omega} \simeq \langle \Sigma_{\text{Rus}}, \phi \rangle_{\partial\Omega}^{N_G} := \sum_{s=1}^{n_s} \sum_{i=1}^{N_G} \Sigma_{\text{Rus}}^{s,i} w_i, \quad (47)$$

where $\Sigma_{\text{Rus}}^{s,i}$ is the approximated flux eq. (45) evaluated at the i th point of the s th subsurface, and w_i is the quadrature weight at that i th point. Inviscid fluxes are implemented exactly like so. Viscous fluxes however do not represent wave propagation, therefore for them we choose the centred flux by setting $\lambda_{\max} = 0$ (Brissaud *et al.* 2017).

4.1.3 Matrix assembling

Through the classical process of finite elements assembling and using eq. (33), eq. (43), and eq. (47), the weak formulation eq. (29) rewrites under the form of the matrix–vector equality:

$$\forall t \geq 0, \quad A \dot{q}^N(t) = K q^N(t) + b(t), \quad (48)$$

where A is the mass matrix of the system, $\dot{q}^N(t)$ is the vector of components of the discretized $\partial_t q(t)$, K is the stiffness matrix of the system, $q^N(t)$ is the vector of components of the discretized $q(t)$, and $b(t)$ is a vector of zeroth-order terms. K and b are assembled point by point by running through all elements and using eq. (33), eq. (43), and eq. (47). Once the RHS is assembled and multiplied by the inverse mass matrix, time integration can ensue, as detailed in Section 4.2.

4.1.4 Auxiliary tensors

In the computation of said viscous fluxes, in order to obtain $\Sigma_v(\mathbf{v}')$ (see eqs 7 and 10), we first need to compute $\nabla \mathbf{v}'$ and $\nabla T'$. Additionally, we need to compute $\nabla \mathbf{v}_0$ in order to obtain the zeroth-order term eq. (8), though we only need to do it once at the beginning of the simulation.

These computations may be done either straightforwardly using the SEM discretization (by making use of the chain rule) or by taking a mixed element formulation approach (Bassi & Rebay 1997; Brissaud *et al.* 2017). Since we choose to implement the latter, let us quickly expose the rationale, which is explained in details by Bassi & Rebay (1997). Consider the test function ϕ and the element Ω . Using Green’s identity, one has:

$$\int_{\Omega} (\nabla f) \phi \, d\Omega = - \int_{\Omega} f \nabla \phi \, d\Omega + \int_{\partial\Omega} f \phi \mathbf{n} \, d\Gamma, \quad (49)$$

where f is either T' , or the components of \mathbf{v}' or \mathbf{v}_0 . Thus, in the same fashion as in the discretization of the bilinear form (see eq. 43), one can compute $\int_{\Omega} (\nabla f) \phi \, d\Omega$. Then, assembling and multiplying by the inverse mass matrix (see Section 4.1.3) yields ∇f at every point of the element. As such, we can define and compute the auxiliary tensors:

$$\begin{cases} \mathcal{T}' := \nabla T', \\ \mathcal{V}' := \nabla \mathbf{v}', \\ \mathcal{V}_0 := \nabla \mathbf{v}_0. \end{cases} \quad (50)$$

Consequently, the perturbed viscous stress tensor (eq. 10) rewrites:

$$\Sigma_v(\mathbf{v}') = \lambda (\nabla \cdot \mathbf{v}') \mathbf{I} + \mu (\nabla \mathbf{v}' + (\nabla \mathbf{v}')^T) = \lambda \text{tr}(\mathcal{V}') \mathbf{I} + \mu (\mathcal{V}' + (\mathcal{V}')^T), \quad (51)$$

where $\text{tr}(M) = \sum_{i=1}^d M_{ii}$ is the trace of the matrix M . Next, the diffusive tensor (eq. 7) rewrites:

$$\Sigma^d(U(t, \mathbf{x})) = \begin{pmatrix} 0 \\ \Sigma_v(\mathbf{v}') \\ \Sigma_v(\mathbf{v}_0)\mathbf{v}' + \Sigma_v(\mathbf{v}')\mathbf{v}_0 + \kappa\mathcal{T}' \end{pmatrix}, \quad (52)$$

with Σ_v defined in eq. (51). Note that $\Sigma_v(\mathbf{v}_0)\mathbf{v}'$ and $\Sigma_v(\mathbf{v}')\mathbf{v}_0$ are simple matrix–vector products. Finally, the zeroth-order term (eq. 8) rewrites:

$$\mathbf{G}(U(t, \mathbf{x})) = \begin{pmatrix} 0 \\ \rho'\mathbf{g} - \rho'\partial_t(\mathbf{v}_0) - (\boldsymbol{\delta}_m \cdot \nabla)\mathbf{v}_0 \\ \mathbf{g} \cdot \boldsymbol{\delta}_m \end{pmatrix} = \begin{pmatrix} 0 \\ \rho'\mathbf{g} - \rho'\partial_t(\mathbf{v}_0) - \begin{pmatrix} \boldsymbol{\delta}_m \cdot [\mathcal{V}_0]_{1,1 \leq j \leq d} \\ \vdots \\ \boldsymbol{\delta}_m \cdot [\mathcal{V}_0]_{d,1 \leq j \leq d} \end{pmatrix} \\ \mathbf{g} \cdot \boldsymbol{\delta}_m \end{pmatrix}, \quad (53)$$

where $\boldsymbol{\delta}_m$ is defined in eq. (11), and $[\mathcal{V}_0]_{i,1 \leq j \leq d} = ([\mathcal{V}_0]_{i,1}, \dots, [\mathcal{V}_0]_{i,d})$ is the i th line of \mathcal{V}_0 .

4.2 Time discretization

Consider the prototype differential equation under the form:

$$\partial_t q(t) = L(t, q(t)), \quad (54)$$

where q represents the quantity of interest, and L is the differential operator. Every equation in the LNS fluid system eq. (4) can readily be put under the form of the prototype equation eq. (54). In the following, Δt denotes the chosen time step. It is considered here unique and constant with time, to the contrary of adaptive time step methods. Unless otherwise noted, a superscript ‘ n ’ means the value is taken at time $t^n = t^0 + n\Delta t$, where t^0 is the starting time of the simulation.

Parallelized DG methods mainly pay off when using explicit time-integration schemes, rather than implicit ones. To begin with, because the mass matrix is local and diagonal, there is no need to apply the inverse of the mass matrix globally and communication between elements and processors is limited to surface data. Furthermore, because of the structure of the chosen discretization, the amount of computations in each subdomain/partition related to a given process is far greater than the amount of communications needed between those subdomains/partitions. In short, for explicit parallelized DG schemes, the ratio of computational effort to communication effort is by definition high, and increases with polynomial degree. To go further, Lalande *et al.* (2012) show how one can achieve almost perfect scaling up to one cell per processor. Thus, we choose to implement only explicit time stepping schemes.

In particular, we use the low-storage implementation of explicit RK numerical schemes. For each constitutive variable, the corresponding register is updated at every iteration n as follows (see e.g. Williamson (1980); Carpenter & Kennedy (1994); Berland *et al.* (2006)):

$$\forall n \in \mathbb{N}, \quad \begin{cases} \tilde{U}_0 = 0, \\ Y^{n+1} = Y^n, \\ \forall 1 \leq i \leq m, \begin{cases} \tilde{U}_i = \alpha_i \tilde{U}_{i-1} + \Delta t L(t^{n,i}, Y^{n+1}), \\ Y^{n+1} = Y^{n+1} + \beta_i \tilde{U}_i, \end{cases} \end{cases} \quad (55)$$

where n is the iteration number, m is the number of sub-iterations, Y the constitutive variable’s register, \tilde{U} the auxiliary register, $(\alpha_i, \beta_i, c_i)_{1 \leq i \leq m}$ are the scheme’s coefficients, and $t^{n,i} = t^n + c_i \Delta t$ is the time of the sub-step. Let M denote the amount of memory space needed for a variable’s register. This implementation only requires $2M$ storage per variable to be iterated, which is advantageous compared to the $(m+1)M$ for the classical explicit RK implementation.

We implemented three schemes: the six-stage fourth-order scheme of Berland *et al.* (2006), the three-stage third-order scheme of Williamson (1980), and the five-stage fourth-order RK time-stepping scheme of Carpenter & Kennedy (1994). We choose the latter for all of our applications and the remaining of this paper, and list the coefficients in Table 2.

5 FLUID BOUNDARY CONDITIONS

At the outer boundary of the computational domain, periodic BCs are widely used in computational fluid dynamics numerical methods; however this technique significantly limits the possible applications of the considered simulations. Designing *absorbing boundary conditions* at the outer boundary of the fluid domain allows one to model accurately a much larger and general set of situations.

In addition, and especially for the problem of coupled ground-atmosphere simulations, the *coupling boundary conditions* with other materials have to be considered. Accurately modelling such conditions allow the transmission of energy from the ground to the air, and vice versa.

Table 2. Five-stage fourth-order ‘54LSRK’ optimal coefficients (rational form, 26-digit precision) Carpenter & Kennedy (1994).

| i | α_i | β_i | c_i |
|-----|--|--|---------------------------------------|
| 1 | 0 | $\frac{1432997174477}{9575080441755}$ | 0 |
| 2 | $-\frac{567301805773}{1357537059087}$ | $\frac{5161836677717}{13612068292357}$ | $\frac{1432997174477}{9575080441755}$ |
| 3 | $-\frac{2404267990393}{2016746695238}$ | $\frac{1720146321549}{2090206949498}$ | $\frac{2526269341429}{6820363962896}$ |
| 4 | $-\frac{3550918686646}{2091501179385}$ | $\frac{3134564353537}{4481467310338}$ | $\frac{2006345519317}{2802321613138}$ |
| 5 | $-\frac{1275806237668}{842570457699}$ | $\frac{4277821191437}{14882151754819}$ | $\frac{2924317926251}{2924317926251}$ |

We describe in this section the implementation of BCs. Under BCs, we therefore understand both the outer BCs (OBCs) and the coupling BCs (CBCs). The OBCs are the conditions applied at the outer boundary of the computational domain, and in our case aim at being fully absorbing to prevent outward-propagating waves from coming back into the inner computational domain. The CBCs play a key role at the interface between the fluid (governed by eq. 4) and the elastic (governed by eq. 2) media.

5.1 Outer boundary conditions

While exact methods exist for simpler systems and/or background models (Poinsot & Lele 1992; Benzoni-Gavage *et al.* 2003; Bécache *et al.* 2010), implementing absorbing BCs for general subsonic windy atmospheres remains difficult. In this section, Section 5.1.1 first presents the straightforward *far-field* approach. Next, Section 5.1.2 details the newly implemented real stretching absorbing boundary conditions.

5.1.1 Far-field

A first approximation is the Far-Field BCs (FFBCs). In those, some choice of relevant steady state $(\rho_\infty, \mathbf{v}_\infty, p_\infty)$ is enforced strongly at the boundary (on the outer edges). 44 We consider the Riemann system eq. (44) posed at the outer boundary of the computational domain, in which we choose the ‘+’ side to be outside the domain. Enforcing strongly the far-field condition amounts to setting the following ‘ghost values’ at the ‘+’ side:

$$\begin{cases} \rho'_+ = \rho_\infty - \rho_0 & \text{(Neumann condition),} \\ \mathbf{v}_+ = \mathbf{v}_\infty, \\ p'_+ = p_\infty - p_0 & \text{(Neumann condition),} \\ \nabla T'_+ = \nabla T'_- & \text{(Neumann condition),} \\ \mathfrak{N}'_+ = \mathfrak{N}'_- & \text{(Neumann condition).} \end{cases} \quad (56)$$

The remaining quantities are deduced classically. The FFBCs prove to be very efficient for waves propagating at orthogonal incidence relative to the boundary, but not so much for grazing incidence waves (see tests in Appendix B2).

5.1.2 Real stretching absorbing boundary conditions

A perfectly matched layer (PML) method limited to real coefficients is implemented, and referred to as Real Stretching BCs (RSBCs). In literature, this technique is sometimes also called mesh virtual stretching or sponge layers (Edgar & Visbal 2003; Arina 2016). Contrary of the FFBCs (Section 5.1.1), and similar to PMLs (Hu 2005; Hu *et al.* 2008; Lin *et al.* 2011; Xie *et al.* 2015), these conditions are buffer-based and need a number of elements to be applied onto at the computational domain exterior boundary. However, RSBCs prove more efficient than FFBCs at grazing incidences (see tests in Appendix B2).

Let D denote the computational domain, and let $(B_k)_{1 \leq k \leq N_b}$ ($B_k \subset D$ for all k) denote the buffer zones (consider a total of N_b buffer zones). The *computational domain of interest* D_i is defined as the region where the physical equations remain unmodified, that is exactly $D_i = \text{cl}\left(D \setminus \bigcup_{k=1}^{N_b} B_k\right)$, that is the closure of the computational domain with all buffers removed.

First, a change of variable is introduced following Xie *et al.* (2014). For each spatial dimension $1 \leq i \leq d$, the real coordinate component x_i is stretched by a general non-zero complex coordinate-wise function s_i such that the stretched coordinate \tilde{x}_i writes:

$$\tilde{x}_i(\mathbf{x}_i) := \int_0^{\mathbf{x}_i} s_i(\mathbf{x}_i) d\mathbf{x}_i, \quad (57)$$

that is, more explicitly and in 3-D:

$$\begin{aligned} \tilde{x}(\mathbf{x}) &:= \int_0^x s_1(\mathbf{x}) dx, \\ \tilde{y}(\mathbf{x}) &:= \int_0^y s_2(\mathbf{x}) dy, \\ \tilde{z}(\mathbf{x}) &:= \int_0^z s_3(\mathbf{x}) dz. \end{aligned} \quad (58)$$

Note how the change of variable along the direction i depends only on said direction. Moreover, we choose this change of variable constant with time. The chain rule on the gradient operator then yields:

$$\begin{aligned} \bar{\nabla} &= M_0 \nabla & \text{where } M_0 &:= \text{diag}(s_i^{-1})_{1 \leq i \leq d} \\ \Leftrightarrow \bar{\nabla} &= J^{-1} M \nabla & \text{where } J &:= \prod_{i=1}^d s_i \text{ and } M := \text{diag}(J/s_i)_{1 \leq i \leq d}. \end{aligned} \quad (59)$$

Now, consider the prototype hyperbolic system:

$$\partial_t q + \nabla \cdot \Sigma = G. \quad (60)$$

Let us denote F the temporal Fourier transform, and $\hat{f} = F(f)$ the temporal Fourier transform of f . Applying the PML stretching method on eq. (60) is done in the following few steps:

$$\begin{aligned} \partial_t q + \nabla \cdot \Sigma &= G \\ \Leftrightarrow \mathbf{i}\omega \hat{q} + \nabla \cdot \hat{\Sigma} &= \hat{G} & \text{(step 1: temporal Fourier transform)} \\ \Leftrightarrow \mathbf{i}\omega \hat{q} + \bar{\nabla} \cdot \bar{\Sigma} &= \bar{G} & \text{(step 2: change of variables)} \\ \Leftrightarrow \mathbf{i}\omega \hat{q} + J^{-1} M \nabla \cdot \bar{\Sigma} &= \bar{G} & \text{(applying chain rule)} \\ \Leftrightarrow \mathbf{i}\omega \hat{q} + J^{-1} \nabla \cdot (M \bar{\Sigma}) &= \bar{G} & \text{(each } M_{ii} \text{ is independent of } x_i) \\ \Leftrightarrow \mathbf{i}\omega J \hat{q} + \nabla \cdot (M \bar{\Sigma}) &= J \bar{G} \\ \Leftrightarrow F^{-1}(\mathbf{i}\omega J \hat{q}) + \nabla \cdot F^{-1}(M \bar{\Sigma}) &= F^{-1}(J \bar{G}) & \text{(step 3: inverse Fourier transform)} \\ \Leftrightarrow Q \star q + \nabla \cdot \bar{\Sigma} &= \bar{G}, \end{aligned} \quad (61)$$

where $Q := F^{-1}(\mathbf{i}\omega J)$, \star is the (temporal) convolution operator, and:

$$\bar{\Sigma} := F^{-1}(M \bar{\Sigma}), \quad (62)$$

$$\bar{G} := F^{-1}(J \bar{G}). \quad (63)$$

We have used the following notations: ‘ $\hat{\cdot}$ ’ indicates a quantity on which both the temporal Fourier transform and the change of variable are applied², and ‘ $\bar{\cdot}$ ’ indicates the PML-stretched counterpart of a quantity. We make use of the fact that $\tilde{f} = \hat{f}$ whenever f does not contain spatial derivatives, especially at step 2.

During step 1, the whole equation eq. (60) is Fourier transformed. During step 2, only $\nabla \cdot \Sigma$ and G changed, because they are the only terms possibly having spatial derivatives. The chain rule step results from eq. (59). During step 3, we made use of the fact that $F(\nabla q) = \nabla F(q)$ because the Fourier transform in time and the derivation in space are decoupled, because we choose the change of variable to be constant in time.

For some given $(s_i)_{1 \leq i \leq d}$, computer algebra systems performing formal calculations can give exact expressions for Q . In particular, if poles of $\mathbf{i}\omega J$ are unique, Q trivially takes the general form:

$$\forall t \geq 0, \quad Q(t) = \sum_{i=0}^M a_i \delta^{(i)}(t) + \sum_{j=0}^N b_j e^{-\beta_j t} H(t), \quad (64)$$

where $\delta^{(i)}$ is the i th derivative (to the sense of tempered distributions) of the Dirac function, H is the Heaviside function, and the $((a_i)_{1 \leq i \leq M}, (b_j)_{1 \leq j \leq N})$ are some coefficients depending on the expressions of the $(s_i)_{1 \leq i \leq d}$.

Following Xie *et al.* (2014), let simply for all $1 \leq i \leq d$:

$$\forall r \in [0, 1], \quad s_i(r) := \kappa_i(r), \quad (65)$$

with $\kappa_i : [0, 1] \rightarrow \mathbb{R}$ a real-valued stretching function in the i th direction ($1 \leq i \leq d$). We defer our choice for this parameter to Appendix B2, concerning the verification of the OBCs. Though eq. (65) appears as nothing but a variable renaming, we keep it as is because the developments below can be extended to other choices for s_i . Complex-valued s_i leads to a classical PML formulation, enables a better damping of the incoming waves, but requires implementing several auxiliary partial differential equations which are computationally prohibitive in our case.

Following eq. (65), Q defined in eq. (64) takes the form:

$$Q = F^{-1} \left(\mathbf{i}\omega \prod_{i=1}^d \kappa_i \right) = -a_1 \delta', \quad (66)$$

where $a_1 = \prod_{i=1}^d \kappa_i$, and δ' is the first derivative of the Dirac distribution. Since $\langle \delta', f \rangle = -\langle \delta, f' \rangle$ for all f :

$$Q \star q = a_1 \partial_t q. \quad (67)$$

Furthermore:

$$\forall 1 \leq i \leq d, \quad \check{s}_i := F^{-1}(s_i) = \kappa_i \delta, \quad (68)$$

²The order of application is interchangeable since we choose the change of variable is constant in time.

and:

$$\check{J} := F^{-1}(J) = a_1 \delta. \quad (69)$$

In 2-D and using eq. (68), $\check{\Sigma}$ writes:

$$\check{\Sigma} = F^{-1}(M\check{\Sigma}) = \check{M}\star F^{-1}(\check{\Sigma}) = \begin{pmatrix} \check{s}_2 \star [\Sigma]_1 \\ \check{s}_1 \star [\Sigma]_2 \end{pmatrix} = \begin{pmatrix} \kappa_2 [\Sigma]_1 \\ \kappa_1 [\Sigma]_2 \end{pmatrix}, \quad (70)$$

where \star is the element-wise (temporal) convolution, and \check{M} is the element-wise inverse Fourier transformed matrix $M^{\check{}}$. Again in 2-D, and now using eq. (69), \check{G} writes:

$$\check{G} = F^{-1}(J\check{G}) = \check{J} \star F^{-1}(\check{G}) = a_1 G. \quad (71)$$

Finally, after the RSBC transformation, the generic equation eq. (60) can be rewritten. From the last step of eq. (61), using eq. (67)–eq. (70)–eq. (71), and dividing both sides by a_1 :

$$\partial_t q + \nabla \cdot \begin{pmatrix} \kappa_1^{-1} [\Sigma]_1 \\ \kappa_2^{-1} [\Sigma]_2 \end{pmatrix} - G = 0. \quad (72)$$

Because of this very compact form, we can now make use of the SEM discretization to include the stretching without modifying the implementation. Namely, the κ_i^{-1} are directly implemented (where needed, that is in the buffers B) as a simple multiplying factor in the bilinear form eq. (43) and in the surface term eq. (45).

We note that, in eq. (4), the viscous terms and one of the zeroth-order term contain spatial derivatives. Consequently, eq. (70) and eq. (71) may not hold. However, this issue is circumvented by using the auxiliary tensors eq. (50) computed through the mixed element formulation (see Section 4.1.4). By making sure the stretching occurs also in this intermediate step, and since there are no spatial second derivatives, the stretching is fully accounted for in the aforementioned problematic terms.

At the end of the buffer, that is on the very ending edge of the whole computational domain, the far-field condition eq. (56) is applied. Since waves are then stretched, the wavefronts are effectively parallel to the edge of the computational domain⁴, and consequently this poses no issue (see Appendix B2).

5.2 Fluid–solid coupling boundary conditions

The CBCs between the fluid and solid domains (and thus systems of equations) are essential to capture realistic wave transmission, reflection, and conversion at the interface between media, including the evanescent air-to-ground motion at large incidences (Woods *et al.* 2015) and compliance effects for horizontally propagating air pressure perturbations (Sorrells 1971; Sorrells *et al.* 1971; Kenda *et al.* 2017, 2020; Martire *et al.* 2020).

For the remaining of this section, let Γ denote the considered fluid–solid boundary, between a medium and the other. Denote \mathbf{n} and \mathbf{t} the normal and tangential unit vectors (respectively) to the interface Γ . We neglect all drag effects stemming from viscous terms in eq. (4) because they are usually negligible in most geophysical applications. The CBCs presented below are verified later, in Section 7.

5.2.1 Solid-to-fluid

The *solid-to-fluid* coupling condition is applied to the LNS system by setting the following values on the fluid side of the fluid–solid interface:

$$\begin{cases} \partial_{\mathbf{n}} \rho'_f(\mathbf{x}) = 0 & \text{(Neumann condition),} \\ \mathbf{v}_f \cdot \mathbf{n} = \mathbf{v}_+ \cdot \mathbf{n}, \\ \partial_{\mathbf{n}} (\mathbf{v}_f \cdot \mathbf{t}) = 0 & \text{(Neumann slip condition),} \\ p'_f = p'_+, \\ \partial_{\mathbf{n}} (\nabla T'_f) = 0 & \text{(Neumann condition),} \\ \partial_{\mathbf{n}} \mathfrak{N}'_f = 0 & \text{(Neumann condition).} \end{cases} \quad (73)$$

Here, $\partial_{\mathbf{n}} = \sum_{i=1}^d [\mathbf{n}]_i \partial_{x_i}$ denotes the spatial derivative in the direction $[\mathbf{n}]_i$ normal to the interface. The ‘f’ subscripts denote quantities on the fluid side. Finally, \mathbf{v}_+ and p'_+ are some choice of total velocity and pressure perturbation made for the solid side, and are detailed below. Note that in eq. (73), the condition is on the total velocity \mathbf{v} , and not on the velocity perturbation \mathbf{v}' . The fluid constitutive variables of eq. (4) ρ' and $\rho_0 \mathbf{v}'$ are readily obtained through eq. (73). The energy perturbation E' can be computed from eq. (18) with $E(\rho, \mathbf{v}, p)$ defined in eq. (16).

In practice, the Neumann conditions in eq. (73) are applied by choosing relevant ghost states in the considered Riemann system (eq. 44). Then, the Rusanov flux (eq. 45) is used without any other change. The conditions on $\mathbf{v}_f \cdot \mathbf{n}$ and p'_f are chosen according to the full Riemann

³That is, $[\check{M}]_{ij} = F^{-1}([M]_{ij})$ for all $1 \leq i, j \leq d$.

⁴This supposes that a stretching effectively occurred in the axis perpendicular to the interface, that is $s_i > 1$ for each direction $1 \leq i \leq d$.

problem at the fluid–solid interface, which deals with different sets of equations (in our case, the equations of elastodynamics and the LNS equations). The solid-to-fluid CBC needs to account for the compressional waves in the fluid domain, but also for both the compressional and shear waves in the elastic domain. This means that three Rankine-Hugoniot relations must be satisfied (Terrana *et al.* 2018). Consequently, p'_+ and \mathbf{v}'_+ are chosen based on Terrana *et al.* (2018, eqs 51 and 56), though rather than relying on $p\mathbf{I}$ for the fluid stress, one can be more complete by considering the full inviscid fluid stress Σ^f (see eq. (13)). This results in choosing:

$$\begin{cases} \mathbf{v}'_+ = (\tau^s + \tau^f)^{-1} (\tau^s \mathbf{v}_s + \tau^f \mathbf{v}_f + (\Sigma^s - \Sigma^f) \mathbf{n}), \\ p'_+ = p'_f + (\tau^f (\mathbf{v}_f - \mathbf{v}'_+)) \cdot \mathbf{n}. \end{cases} \quad (74)$$

where \mathbf{v}_s is the solid's total velocity, \mathbf{v}_f is the fluid's total velocity, Σ^s is the solid's stress tensor, and p'_f is the fluid's pressure perturbation. The symmetric matrices τ^s and τ^f are based on the method of characteristics (Terrana *et al.* 2018) and are defined as follows. In the solid part:

$$\tau^s = \tau \mathbf{n} (\rho^s, v_p, v_s), \quad (75)$$

where ρ^s the solid's density and v_p, v_s are the P - and S -wave velocities. In the fluid part:

$$\tau^f = \tau \mathbf{n} (\rho, c, 0), \quad (76)$$

where ρ is the fluid's density, and c is the (compressional) speed of sound. In eqs (75) and (76), the generic symmetric matrix $\tau \mathbf{n}$ is defined following Terrana *et al.* (2018):

$$\tau \mathbf{n} (\rho, v_p, v_s) := \rho \begin{pmatrix} v_p [\mathbf{n}]_1^2 + v_s [\mathbf{n}]_2^2 + v_s [\mathbf{n}]_3^2 & (v_p - v_s) [\mathbf{n}]_1 [\mathbf{n}]_2 & (v_p - v_s) [\mathbf{n}]_1 [\mathbf{n}]_3 \\ * & v_s [\mathbf{n}]_1^2 + v_p [\mathbf{n}]_2^2 + v_s [\mathbf{n}]_3^2 & (v_p - v_s) [\mathbf{n}]_2 [\mathbf{n}]_3 \\ * & * & v_s [\mathbf{n}]_1^2 + v_s [\mathbf{n}]_2^2 + v_p [\mathbf{n}]_3^2 \end{pmatrix}, \quad (77)$$

where ρ is some density, v_p, v_s are the P - and S -wave velocities, and $[\mathbf{n}]_i$ denotes the i th component of the normal vector \mathbf{n} . The '*' indicates the term is the same as the corresponding one in the upper right triangle matrix (the matrix is symmetric). With the definition given in eq. (77), note that eq. (76) reduces exactly to $\rho c \mathbf{n} \otimes \mathbf{n}$ which depends only on the orientation of the interface.

The CBCs relying on eqs (73) and (74) are enforced weakly, through the setting of relevant ghost '+' states in the Riemann system eq. (44) posed at the fluid–solid interface Γ . This method is now implemented in SPECFEM2D-DG for both the LNS and FNS fluid systems. Notably, this condition slightly differs from the approximation first described by Brissaud *et al.* (2017).

5.2.2 Fluid-to-solid

The solid is mostly affected by the fluid through stresses normal to the interface. Consequently, the *fluid-to-solid* CBC writes:

$$\Sigma^s \mathbf{n} = \Sigma^f \mathbf{n}, \quad (78)$$

where Σ^s is the solid's stress tensor, and Σ^f is the fluid's inviscid stress tensor eq. (13). This condition is imposed strongly at the solid side of the fluid–solid boundary Γ , by adding to the acceleration ($\partial_{i_2}^2 \mathbf{u}^s$ in eq. 2) the contribution of this stress.

6 VERIFICATION OF THE FLUID EQUATIONS USING THE METHOD OF MANUFACTURED SOLUTIONS

The verification of the solid parts of the software (classical SPECFEM) has already been thoroughly covered in literature (e.g. by Komatitsch & Vilotte 1998; Komatitsch & Tromp 1999; Tromp *et al.* 2008); consequently, this section is only dedicated to the verification of the fluid equations for the LNS extension. In Section 6, we focus on verifying the fluid part itself using the method of manufactured solutions (MMS). Appendix B1 presents another verification method, using analytical solutions obtained through a frequency–wavenumber (FK) method. Finally, we verify the OBCs in Appendix B2, and especially the newly introduced real stretching BCs (which were introduced in Section 5.1.2).

In the MMS, the mathematical form of the solution is chosen *a priori* (Roache 2002; Roy *et al.* 2002, 2004). Then, the differential operator of interest is analytically applied to this solution in order to produce analytical source terms. Finally, these source terms are input into the software, and a numerical solution is performed. The initial conditions are chosen as the default isobaric background model (see Section 3), as described below. The OBCs are chosen as Neumann conditions matching the analytical solution, so as not to perturb the convergence to the steady-state solution. If the implementation perfectly reflects the governing equations, then the numerical solution fields must agree with the analytical fields chosen *a priori*.

In the remaining of this section, let $d = 2$. Let the fluid model (and initial conditions) be isobaric (see Table 1) with $\bar{\rho}_0 = 1.4 \text{ kg m}^{-3}$, $\gamma \simeq 1.401$, and $\bar{c} = 340 \text{ m s}^{-1}$. Let the wind $\mathbf{v}_0 = w(1, 0)^T$ with $w = 10 \text{ m s}^{-1}$. The viscosity coefficients (μ, κ) are chosen differently for each test case.

Table 3. Presentation of the method of manufactured solutions (MMS) verification test cases. They are described in details in Section 6. These parameters are for the analytical fields described in eq. (79). A ‘–’ denotes a parameter which needs not be specified (due to another being zero).

| Test | inviscid | κ | μ |
|------------|----------|----------|-------|
| ρ'_c | 0.001 | 0 | 0 |
| ρ'_x | 1 | – | – |
| ρ'_z | 2 | – | – |
| $v'_{x,c}$ | 0 | 0 | 0.001 |
| $v'_{x,x}$ | – | – | 2 |
| $v'_{x,z}$ | – | – | 0 |
| $v'_{z,c}$ | 0 | 0 | 0 |
| $v'_{z,x}$ | – | – | – |
| $v'_{z,z}$ | – | – | – |
| E'_c | 0.05 | 0.05 | 0 |
| E'_x | 3 | 3 | – |
| E'_z | 4 | 4 | – |

The analytical solutions (and OBCs) are chosen under the following parametric form:

$$\begin{cases} \rho'(\mathbf{x}) &= \rho'_c (\sin(\rho'_x \pi x) + \sin(\rho'_z \pi z)), \\ (\rho_0 \mathbf{v}')(\mathbf{x}) &= \rho_0 \begin{pmatrix} v'_{x,c} (\sin(v'_{x,x} \pi x) + \sin(v'_{x,z} \pi z)) \\ v'_{z,c} (\sin(v'_{z,x} \pi x) + \sin(v'_{z,z} \pi z)) \end{pmatrix}, \\ E'(\mathbf{x}) &= E'_c (\sin(E'_x \pi x) + \sin(E'_z \pi z)), \end{cases} \quad (79)$$

where $(\rho'_c, \rho'_x, \rho'_z, v'_{x,c}, v'_{x,x}, v'_{x,z}, v'_{z,c}, v'_{z,x}, v'_{z,z}, E'_c, E'_x, E'_z)$ are non-negative parameters. Note that the numerical solution is time-dependent, however it should converge to a steady-state time-independent solution, which should correspond to the chosen analytical solution. Remarking that the pressure perturbation p' verifies eq. (17) with $p(\rho, \mathbf{v}, E)$ as defined in eq. (15), p' reads:

$$p'(\mathbf{x}) = (\gamma - 1) \left(\frac{\rho_0 w^2}{2} + E'(\mathbf{x}) - \frac{1}{2} (\rho_0 + \rho'(\mathbf{x})) \left(([\mathbf{v}'(\mathbf{x})]_1 + w)^2 + [\mathbf{v}'(\mathbf{x})]_2^2 \right) \right). \quad (80)$$

where w is the ambient wind, and (ρ', \mathbf{v}', E') are defined in eq. (79).

The choice eq. (79) is based on two arguments. First, it is simple to apply the differential LNS operator eq. (4) to trigonometric functions; more complicated functions can be dealt with using computer algebra software. Second, these forms involve both spatial dimensions in all constitutive variables, which is desirable in order to ‘probe’ the implementation of all possible cross-terms.

Three tests are now conducted. The parameters involved in those are listed in Table 3. The choice of those parameters for each variable q is done such that $|q'(t, \mathbf{x})| \ll |q_0|$, that is such that the small perturbations’ hypothesis eq. (1) remains true.

Next, the analytical solutions eq. (79) (with the relevant choice of parameters) are injected into eq. (4), in order to generate source terms. These source terms are then introduced within the code as classical (though non-physical) source terms. eq. (79) is forced analytically on the outer boundary of the computational domain, and thus serve as OBCs. Then, the numerical resolution can be performed. All constitutive variables are recorded at every single point on the whole domain, in order to be compared with the exact solution eq. (79).

The computational domain is chosen as $D = [0, 1] \times [0, 1]$ m, with $N \in \mathbb{N}$ elements along each direction ($\Delta x = \Delta z = 1/N$).

$$\epsilon_q(N) := \left| \frac{q(\mathbf{x}) - q_N(\mathbf{x})}{q(\mathbf{x})} \right|, \quad (81)$$

where q is some analytical scalar field (typically obtained from eq. (79) or derived from those), q_N its numerical counterpart (numerically solved with N elements in each direction), and \mathbf{x} is the position of the selected receiver chosen arbitrarily as $\mathbf{x} = (0.33, 0.33)^T$. This error is only computed after the simulation has reached steady-state, that is after some numerical transition during which the fields take place. This is why the error does not account for any variation with respect to time.

6.1 Test 1: inviscid (linearized Euler)

Consider eq. (79), with parameters taken from the ‘inviscid’ case column in Table 3. Note that E'_c is chosen based on ρ'_c , in order for the internal and kinetic energies’ contributions to have the same order of magnitude in pressure perturbation (see eq. 80). Injecting such analytical solution into eq. (4) and setting $\mu = \lambda = \kappa = 0$ yields the following source terms:

$$\begin{cases} F_\rho &= \rho'_c \rho'_x \pi \cos(\rho'_x \pi x) w, \\ \mathbf{F}_{\rho_0 \mathbf{v}'} &= (\gamma - 1) \begin{pmatrix} E'_c E'_x \pi \cos(E'_x \pi x) - \frac{\rho'_c \rho'_x \pi}{2} \cos(\rho'_x \pi x) w^2 \\ E'_c E'_z \pi \cos(E'_z \pi z) - \frac{\rho'_c \rho'_z \pi}{2} \cos(\rho'_z \pi z) w^2 \end{pmatrix}, \\ F_E &= w \pi \left(\gamma E'_c E'_x \cos(E'_x \pi x) - \frac{(\gamma-1)\rho'_c \rho'_x}{2} \cos(\rho'_x \pi x) w^2 \right). \end{cases} \quad (82)$$

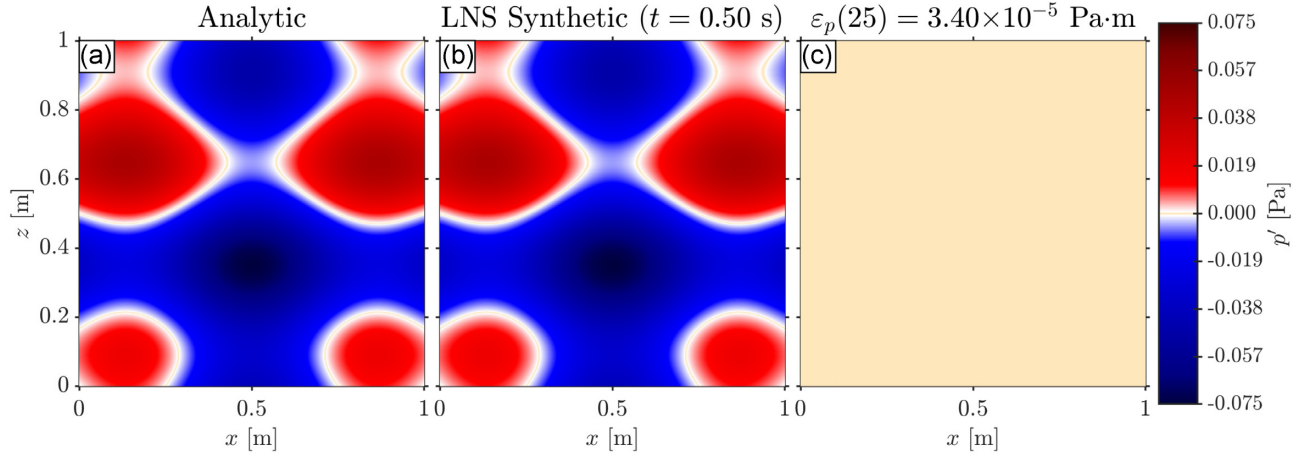


Figure 1. Analytical versus numerical pressure field comparison, using the method of manufactured solutions (MMS), for the inviscid test (see Table 3). (a) Analytical pressure perturbation field. (b) Numerical pressure perturbation field. (c) Signed difference between the two fields; the title shows the value of the L^2 error $\|q - q_N\|_{L^2(D)}$ although it is dominated by an order 2 linear subsampling error. The colour scale is identical between each panel.

An example of field comparison for this test is presented in Fig. 1. It compares the analytical pressure perturbation field (computed from eqs 79 and 17) and the one obtained through numerical solution.

6.2 Test 2: thermal viscosity (κ)

Consider eq. (79), with parameters taken from the ‘ κ ’ case column in Table 3, in order to test the thermal conductivity κ . The source terms are again obtained by injecting the analytical solutions into the LNS equations eq. (4), but this time with non-zero thermal conductivity ($\kappa > 0$). We use the closure formula eq. (14) for temperature. The source terms write:

$$\begin{cases} F_\rho &= 0, \\ \mathbf{F}_{\rho_0 \mathbf{v}'} &= (\gamma - 1) E'_c \begin{pmatrix} E'_x \pi \cos(E'_x \pi x) \\ E'_z \pi \cos(E'_z \pi z) \end{pmatrix}, \\ F_E &= w \pi \gamma E'_c E'_x \cos(E'_x \pi x) + \frac{E'_c \kappa \pi^2}{\rho_0 c_V} \left(\sin(E'_x \pi x) E_x'^2 + \sin(E'_z \pi z) E_z'^2 \right). \end{cases} \quad (83)$$

The thermal conductivity is chosen as $\kappa = 300 \text{ m kg (s}^3 \text{ K)}^{-1}$ in order for the inviscid and viscous parts to have roughly the same magnitude in the source term F_E . This is a quite high and unrealistic value for a gas/fluid (only metals such as aluminum have such a high thermal conductivity), but again this is for purely mathematical verification.

6.3 Test 3: dynamic viscosity (μ)

Consider eq. (79), with parameters taken from the ‘ μ ’ case column in Table 3, in order to test the dynamic viscosity μ . The source terms are obtained as in the ‘ κ ’ test, and write:

$$\begin{cases} F_\rho &= \rho_0 v'_{x,c} v'_{x,x} \pi \cos(v'_{x,x} \pi x), \\ \mathbf{F}_{\rho_0 \mathbf{v}'} &= \begin{pmatrix} \mathbf{F}_{\rho_0 \mathbf{v}'}^i \\ 0 \end{pmatrix} + \begin{pmatrix} \mathbf{F}_{\rho_0 \mathbf{v}'}^v \\ 0 \end{pmatrix}, \\ F_E &= F_E^i + F_E^v, \\ \mathbf{F}_{\rho_0 \mathbf{v}'}^i &= \rho_0 v'_{x,c} v'_{x,x} \pi \cos(v'_{x,x} \pi x) \\ &\quad \times (w - (\gamma - 1)(w + v'_{x,c} \sin(v'_{x,x} \pi x))), \\ \mathbf{F}_{\rho_0 \mathbf{v}'}^v &= \pi^2 v'_{x,c} v'_{x,x} \frac{2\mu}{3} \sin(v'_{x,x} \pi x), \\ F_E^i &= \frac{\rho_0 v'_{x,c} v'_{x,x} \pi \cos(v'_{x,x} \pi x)}{\gamma - 1} \times (0.5(\gamma - 1)w^2 + c^2 - (\gamma - 1)w(w + v'_{x,c} \sin(v'_{x,x} \pi x))), \\ F_E^v &= \pi^2 v'_{x,c} v'_{x,x} \frac{2}{3} \times \left(\frac{8\mu}{3} w \sin(v'_{x,x} \pi x) + \frac{\kappa}{c_V} (v'_{x,c} \sin(2v'_{x,x} \pi x) - w \sin(v'_{x,x} \pi x)) \right). \end{cases} \quad (84)$$

In order to balance inviscid and viscous contributions in the source terms, μ should be chosen quite high. However, high dynamic viscosity yields high Knudsen numbers, and it is well known that the validity of the Navier–Stokes breaks down for high Knudsen numbers. The von Kármán relation links Kn, the Mach number Ma, and the Reynolds number Re as follows:

$$\text{Kn} = \frac{\text{Ma}}{\text{Re}} \sqrt{\frac{\gamma \pi}{2}} = \frac{\mu}{c \rho_0 l_0} \sqrt{\frac{\gamma \pi}{2}}. \quad (85)$$

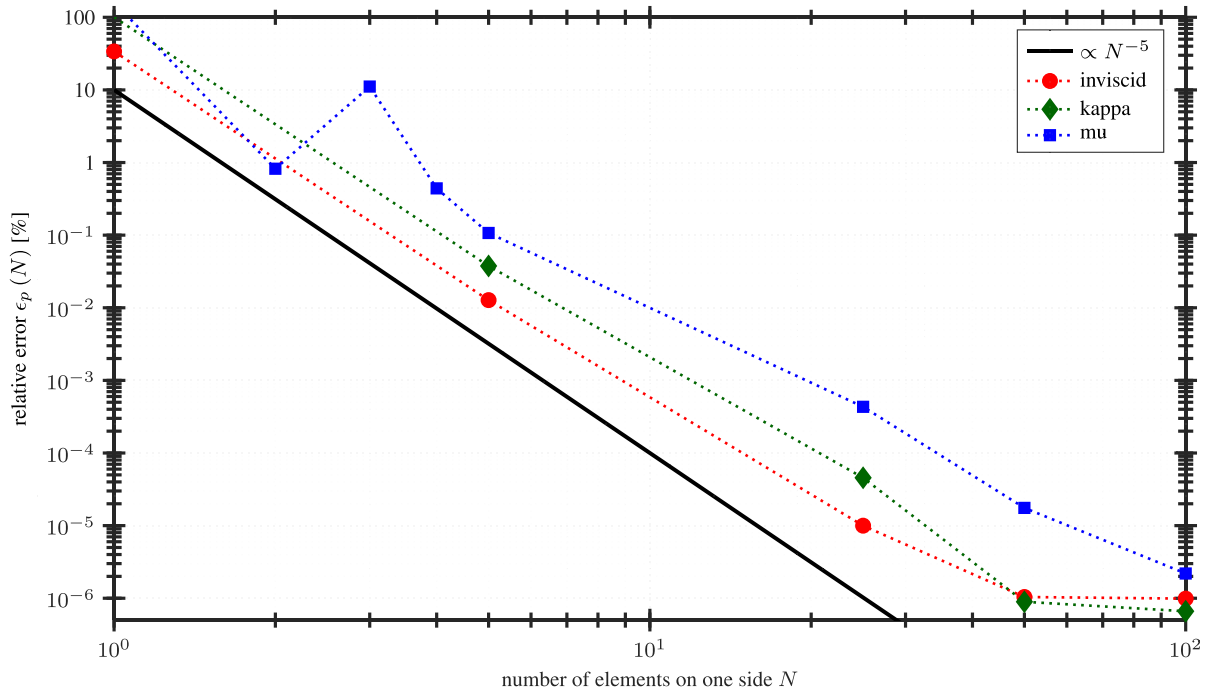


Figure 2. Verification of the software using the method of manufactured solutions. Relative error on a selected receiver eq. (81), as function of the number of elements N on each side, for each test (see Table 3). The theoretical spectral error order (black) is added for comparison. Remark how $\epsilon(N) \propto N^{-N_G}$. Machine error is reached for the kappa (green) and mu (blue) test cases at $N \geq 50$.

where μ is the dynamic viscosity, c the local speed of sound, ρ_0 the background density, l_0 a considered characteristic length, and γ the adiabatic index. The Navier–Stokes equations being only valid for continuous media, that is $\text{Kn} < (\text{Kn})_{\max} := 10^{-3}$, the previous definition yields a maximum μ that can be taken, given the model:

$$\mu_{\max} \simeq (\text{Kn})_{\max} c \rho_0 l_0 \sqrt{\frac{2}{\gamma \pi}}. \quad (86)$$

We choose $\mu = 0.35 \text{ kg (m s)}^{-1} \simeq \mu_{\max}$ ($l_0 = 1 \text{ m}$ the domain size). This viscosity is quite high for a gas/fluid with such density, but again this assumption is done only for mathematical verification purposes. This can also be seen as a worst-case approach.

6.4 Results and discussion

Fig. 2 presents a spectral error convergence study for all test cases, with varying number of elements N , for the relative error eq. (81). We now discuss the results.

Fig. 2 presents the relative error on a selected receiver (eq. 81). We recall that we chose $N_G = 5$. We remark all errors are following the expected convergence, $\epsilon(N) \propto N^{-N_G}$ (Deville *et al.* 2002). By increasing N_G in addition to N , one would achieve spectral convergence. We therefore consider our numerical implementation as verified.

Comparisons using other fields (e.g. ρ' , \mathbf{v}' , or E') always yield similar results, and thus shall not be detailed here. Likewise, variation of the parameters (Table 3) does not change the conclusions (within obvious limitations, for example, of the elements-per-wavelength ratio). More tests can be done by choosing more general cases: other analytical solutions (choices in Table 3 are only simple sub-cases), other winds \mathbf{v}_0 , varying viscosity coefficients (μ , λ , κ), models other than isobaric, etc. Indeed, plugging those in eq. (4) remains trivial. However, the expression of the source terms quickly become extremely tedious, even with computer algebra software. Thus, we choose not to present such cases. In conclusion, the verification of the fluid equations' implementation using the MMS shows good agreement and convergence.

7 VERIFICATION OF THE FLUID–SOLID COUPLING BOUNDARY CONDITIONS

In this section, we verify the wave–wave coupling conditions at the fluid–solid boundary (eqs 73 and 78), that is between the solid system and the LNS fluid system. The method is classical (Brissaud *et al.* 2017) and consists in sending plane waves at various incidence angles against an interface, both from the solid to the fluid and from the fluid to the solid, and checking the reflection and transmission coefficients.

We do not assess the validity of the energy transmission modes occurring over the wave–wave critical angle (see eq. 5 by Martire *et al.* 2020), that is the evanescent ground motion or the compliance mode (Martire *et al.* 2020, section 2.2). Such verification will have to be

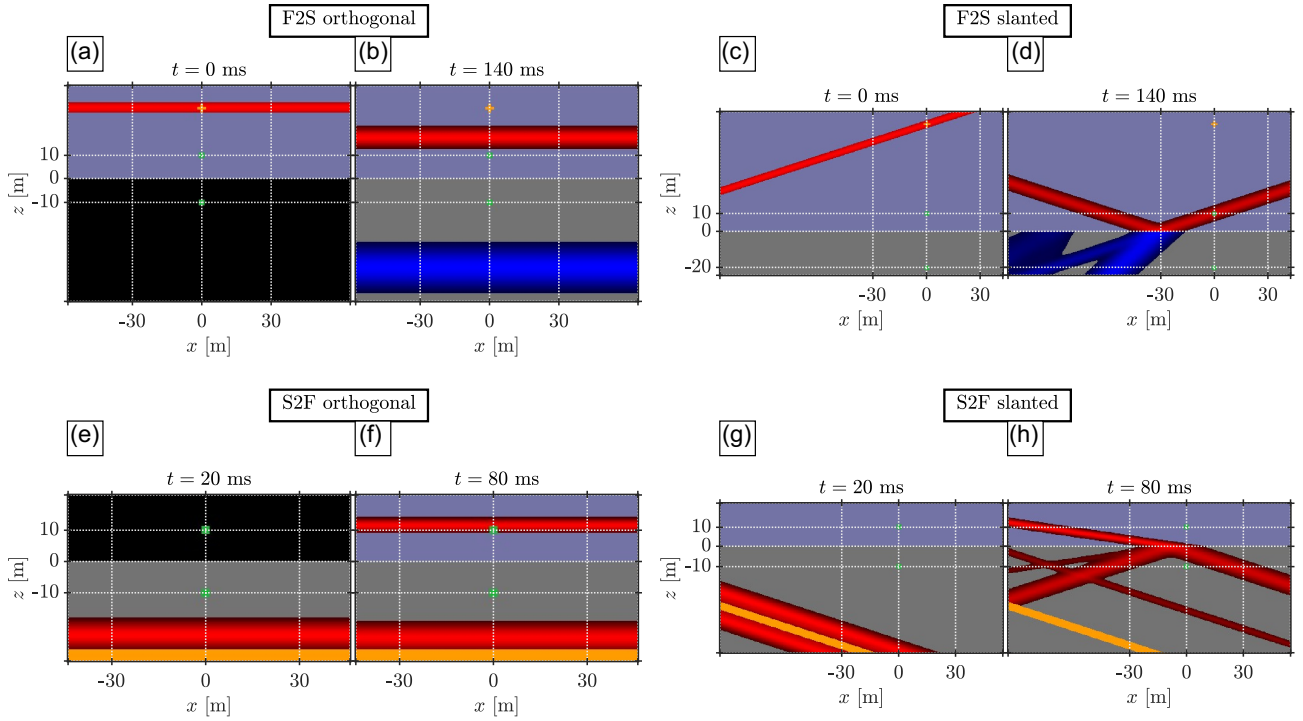


Figure 3. Snapshots of the simulation presented in Section 7, taken at various times (indicated on each panel). (a–d) Fluid-to-solid (F2S) verification, for orthogonal incidence (panels a and b) and slanted incidence (panels c and d). (e–h) Solid-to-fluid (S2F) verification, for orthogonal incidence (panels e and f) and slanted incidence (panels g and h). Snapshots represent pressure perturbations in the atmosphere and vertical velocity in the ground. Red indicates values $> +1$ per cent of maximum amplitude, while blue indicates values < -1 per cent. The green dots represent receivers. The yellow cross pinpoints the source.

performed in the future. Nonetheless, since SPECFEM2D-DG-FNS verifies the compliance mode (Martire *et al.* 2020, appendix A) and since this effect is linear, SPECFEM2D-DG-LNS is expected to also include this mode.

Let the fluid model be isobaric (Table 1), with $\gamma = 1.401$, $\bar{\mu} = \bar{\kappa} = 0$, $\bar{\rho}_0 = 1.00 \text{ kg m}^{-3}$, $\bar{\mathbf{v}}_0 = 0$, $\bar{c} = 370 \text{ m s}^{-1}$. Let the solid media be homogeneous, with: $\rho^s = 5000 \text{ kg m}^{-3}$, $v_p = 800 \text{ m s}^{-1}$, $v_s = 380 \text{ m s}^{-1}$, and no attenuation processes. Let us consider plane waves (coming either from the fluid or the solid side) and a flat topography.

We perform two types of simulations: one *fluid-to-solid* test (F2S) and one *solid-to-fluid* test (S2F). In F2S, a $f_0 = 100 \text{ Hz}$ plane (compressional) wave is created in the fluid domain and launched downwards towards the interface. In S2F, an $f_0 = 100 \text{ Hz}$ plane P wave is created in the solid domain and launched upwards towards the interface. In both types, we record synthetics on both sides (ground velocity and air pressure) and compute the transmission and reflection ratios, which are then compared to the theoretical ones. We perform those tests with varying element sizes. The largest elements' size is adjusted to match the smallest P -wave wavelength in the simulation ($\lambda_0 = \bar{c}/f_0 = 3.7 \text{ m}$). The time step is adjusted for each test in order for the Courant–Friedrichs–Lewy condition number to stay below $1/2$. To begin with, Fig. 3 presents illustrative snapshots of the two simulations, at various times.

The theoretical reflection-transmission coefficients of interest are obtained by following the classical developments introduced by (Aki & Richards 2002, section 5.2.4), which rely on the continuity of displacement and traction at the interface and require extensive algebraic manipulations. We give the full closed forms for these coefficients in Appendix A for both the F2S and S2F cases.

For each case, we compute the aforementioned coefficients using the simulated receivers (see Fig. 3). We summarize our results and therefore the F2S and S2F verification in Fig. 4. In addition, Supporting Information Fig. S1 showcases the time-series involved in each test case, and compare the expected angles at play against the effective simulated signals. All test cases yield excellent agreement with theory, both qualitatively (in terms of effective angles) and quantitatively (in terms of effective amplitudes). The convergence rate does not reach the expected spectral convergence, because of the hybrid implementation at the interface. However, the convergence rate is no less than N^{-3} , and choosing at least 3 elements per wavelength close to the interface ensures a sub-per cent error in the transmission of waves.

8 SCATTERING AND RADIATION PATTERN OF SEISMICALLY INDUCED INFRASOUND OVER TOPOGRAPHY

Recently, the study of seismo-acoustic couplings has gained in popularity. Planetary scientists are currently actively developing various remote sensing strategies for Venus quakes (Stevenson *et al.* 2015; Cutts *et al.* 2019), including balloon-borne pressure measurements, satellite-based electron density measurements, and radio occultation (Cutts *et al.* 2019). Brissaud *et al.* (2019) characterized the dependence of seismically induced infrasound on seismic source mechanisms. Godin (2006, 2007) and Averbuch *et al.* (2020) argue that for shallow sources (i.e. sources

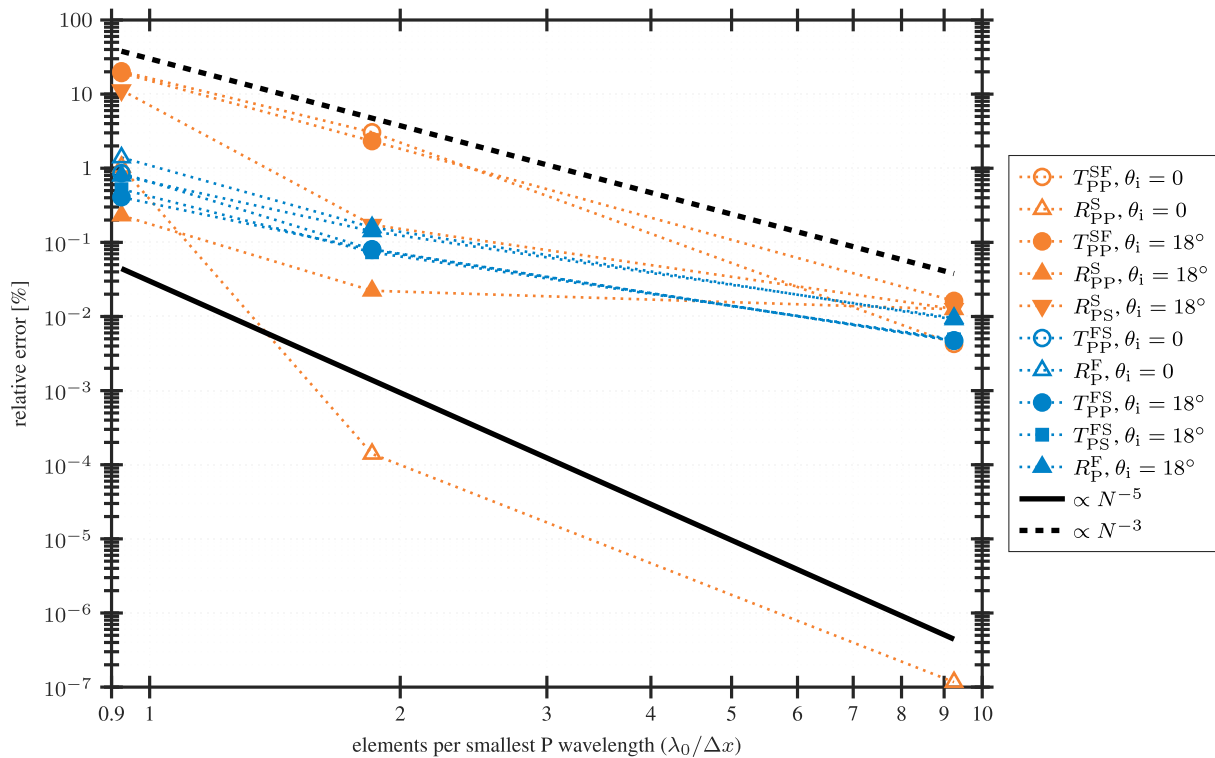


Figure 4. Verification of the wave–wave reflection–transmission coefficients in SPECSEM2D-DG-LNS for two cases, S2F (solid-to-fluid, orange) and F2S (fluid-to-solid, blue) as described in Section 7. For each case, two incident angles θ_i are tested; $\theta_i = i_2$ the P -wave incident angle on the solid side for the S2F case, while $\theta_i = i_1$ the P -wave incident angle on the fluid side for the F2S case. For each case and incident angle, the transmission and reflection coefficients (see Appendix A) are computed and compared to the theoretical values. We plot the results in terms of relative error as function of the number of elements per smallest acoustic P wavelength in the simulation ($\lambda_0 = \bar{c}/f_0 = 3.7$ m).

located at depths shallower than one seismic wavelength), the seismo-acoustic energy radiation patterns is expected to be mostly isotropic; however this is only valid for isotropic underground sources and when the surface roughness is negligible compared to the seismic wavelength (Godin 2007). Variations of the source depth, as well as of the seismic source mechanism, can change the radiation pattern (Martire *et al.* 2018). Long-range ground infrasonic observations are mostly associated with topographic features (Le Pichon *et al.* 2006; Green *et al.* 2009), though classical models always assume the Earth’s surface acts as an isotropic speaker (Walker *et al.* 2013).

On Earth, mountains are expected to be efficient radiators of infrasound (Le Pichon *et al.* 2006; Walker *et al.* 2013). For planetary scientists, seismo-atmospheric couplings for sources at various depths (not only shallow) are of crucial interest. We therefore propose to apply our numerical software to investigate infrasound induced by deep quakes under significant underlying topographies, and characterize the eventual scattering and/or radiation pattern anisotropy.

This section also illustrates the capabilities of SPECSEM2D-DG-LNS, showcasing the use of unstructured meshes to model complex topographies and the ground-atmosphere coupling conditions (detailed in Section 5.2, and verified in Section 7). In the following, the topography is represented by an unstructured mesh made of quadrilateral elements. The input topography (either synthetic or from elevation maps) is first mapped to a continuous piecewise linear interface, with a resolution fine enough to capture all necessary details. Then, the mesh is computed along this interface. For each quadrilateral element, the mapping J (eq. 25) allows to use any number of control nodes; the software implementation supports 4 or 9-nodes elements, but we only used 4-nodes elements here for simplicity.

8.1 Set-up

The computational domain is chosen as $D = [-x_{\max}, x_{\max}] \times [-15, 15]$ km with $x_{\max} = 23$ km. A parametric interface (detailed below) splits the computational domain in two: a solid domain below the interface, and an atmospheric one above it.

The ground model is based on the one-degree lithospheric model LITHO1.0 (Pasyanos *et al.* 2014). We strip away the relatively thin (<1 km) sedimentary layers for simplicity, and consider a single crustal layer between the reference altitude $z = 0$ (sea level) and 15 km depth. The elastic parameters are chosen as: $v_p = 6078.06$ m s⁻¹ for the P -wave velocity, $v_s = 3537.23$ m s⁻¹ for the S -wave velocity, $\rho^s = 2730.14$ kg m⁻³ for the density, and $Q_p = 530$ and $Q_s = 600$ for the quality factors.

The atmospheric model is chosen isothermal (Table 1), with $\gamma = 1.401$, $\bar{\rho}_0 = 1.4$ kg m⁻³, $[\bar{\mathbf{g}}]_d = g_z = 9.81$ m s⁻², $\bar{H} = 8.5$ km, and no viscosity ($\bar{\mu} = \bar{\kappa} = 0$). Note that under these conditions $c \simeq 342$ m s⁻¹. No wind is considered in this application ($\mathbf{v}_0 = 0$).

Table 4. Parameters of the simulations with synthetic topographies. The first simulation has a flat interface and used as a control run; the characteristics of peaks are Not Applicable (N. A.) here. The others feature a sinusoidal interface, parametrized by eq. (87). The columns of this Table present the parameters used for each run. The last column is informative and gives the maximum slope encountered in each case, defined by eq. (88).

| Simulation number | Number of peaks N_{peaks} | Main period λ_{topo} (m) | Height H_{topo} (m) | Maximum slope θ_{max} ($^{\circ}$) |
|-------------------|------------------------------------|---|------------------------------|--|
| S1 | N. A. | N. A. | 0 | 0 |
| S2 | 36 | 1003 | 1500 | 77.98 |
| S3 | 36 | 1003 | 167 | 27.61 |
| S4 | 4 | 9117 | 1500 | 27.34 |

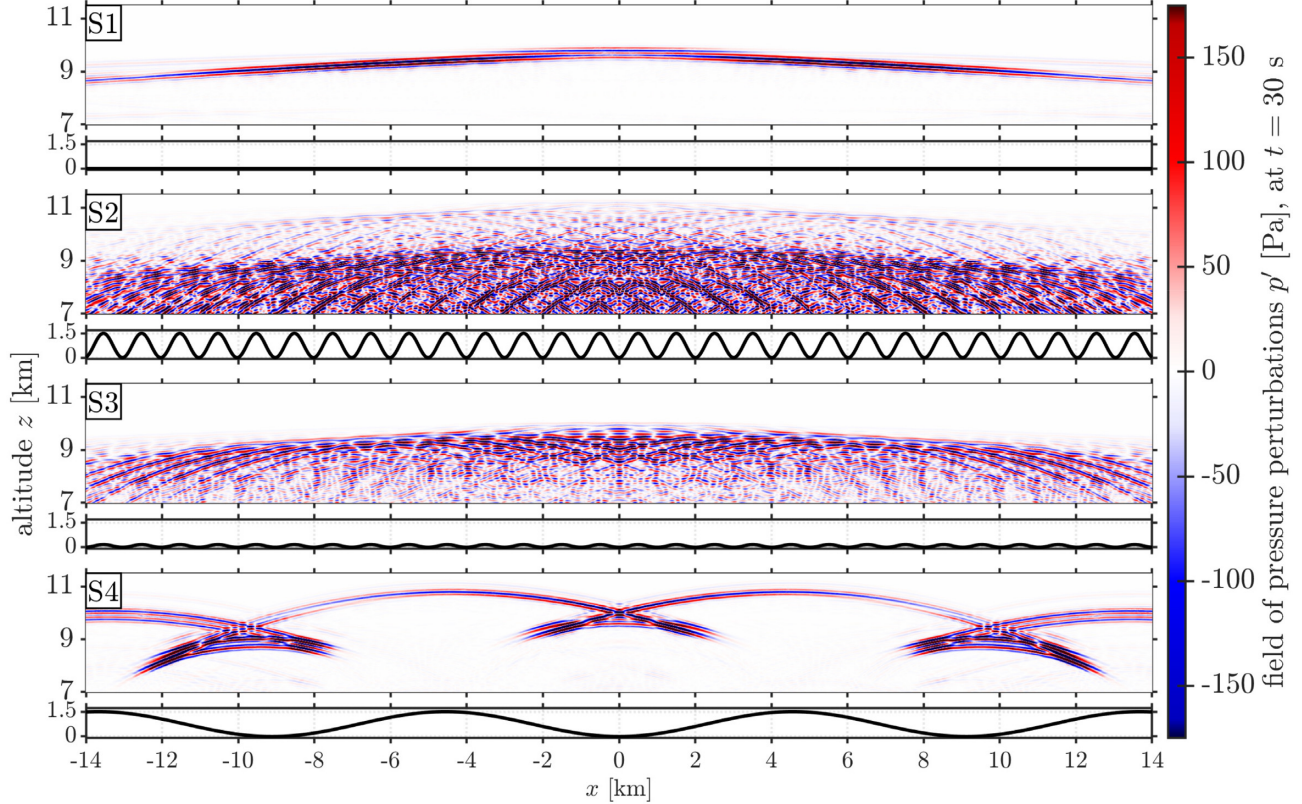


Figure 5. SPEC-FEM2D-DG-LNS synthetic pressure fields at $t = 30$ s for the four simulations with synthetic topographies presented in Section 8.1. For each simulation (S1 to S4, see Table 4), we plot both the pressure field in the 7–15 km altitude range (coloured field, top part of the panels), and the underlying topography (solid black line, bottom part of the panels). Scale is preserved across cases for illustrative purposes. The interfaces are parametrized by eq. (87).

Four simulations with idealized topographies are conducted. In the first one (S1), no topography is considered, the interface is flat: $z_{\text{int}}(x) = 0$ for all x . In simulations S2 to S4, the interface consists in N_{peaks} sinusoidal peaks with dominant wavelength λ_{topo} and height H_{topo} ; z_{int} writes:

$$z_{\text{int}}(x) = \begin{cases} 0.5H_{\text{topo}}(1 - \cos(2\pi x/\lambda_{\text{topo}})) & \text{if } |x| < 0.5N_{\text{peaks}}\lambda_{\text{topo}}, \\ 0 & \text{if } 0.5N_{\text{peaks}}\lambda_{\text{topo}} < |x| < x_{\text{max}}. \end{cases} \quad (87)$$

The maximum slope in such topographies is readily obtained as:

$$\theta_{\text{max}} = \max_x \left(\arctan \left(\frac{\partial z_{\text{int}}(x)}{\partial x} \right) \right) = \arctan \left(\frac{\pi H_{\text{topo}}}{\lambda_{\text{topo}}} \right). \quad (88)$$

Table 4 summarizes the characteristics of the four set-ups; the interfaces z_{int} for each set-up are presented in Fig. 5 below.

The source is chosen as a moment tensor modelling a 45° dip earthquake:

$$M(t, \vec{x}) = -2M_0(\pi f_0)^2 t e^{-(\pi f_0 t)^2} \delta_{\mathbf{x}_0}(\mathbf{x}) \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad (89)$$

where M_0 is the seismic moment, $f_0 = 2$ Hz is the dominant frequency, δ is the Dirac function (the source is applied at a single point in space), and $\mathbf{x}_0 = [0, -5]$ km is the source localization in $[x, z]$ coordinates. The seismic moment M_0 represents a magnitude $M_w = 2.5$

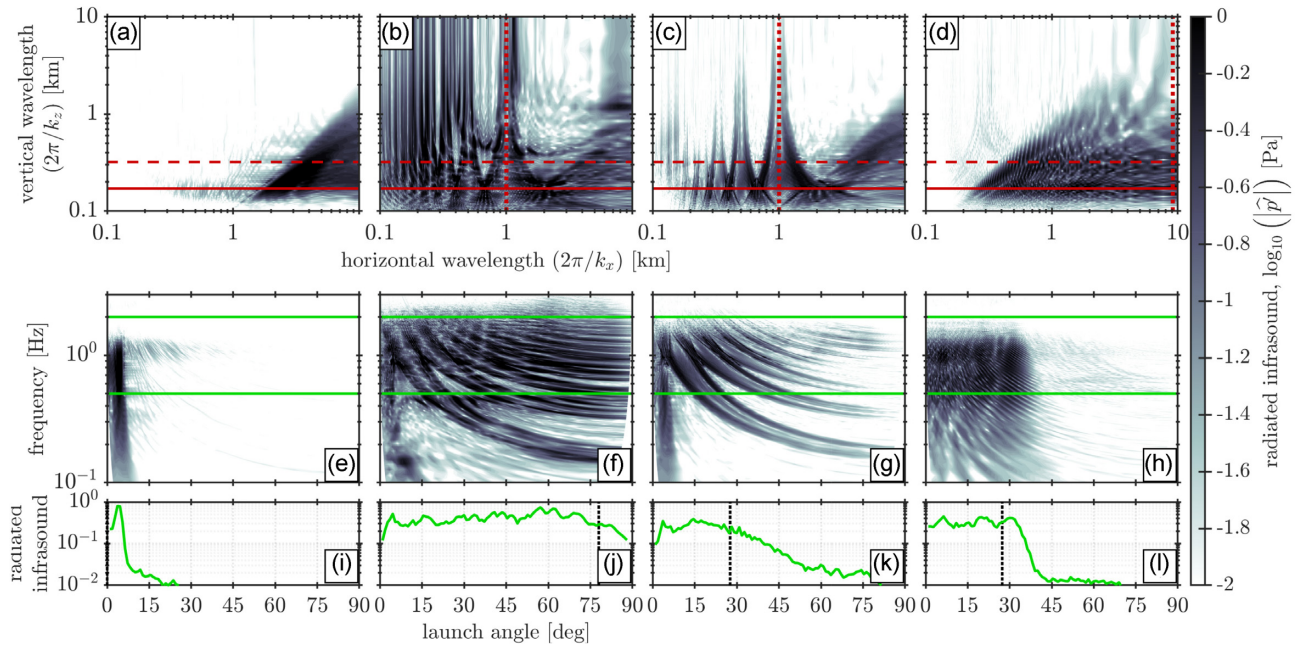


Figure 6. Pressure spectra, computed as the 2-D Fourier transform of each pressure field between 1.5 and 15 km altitude, taken at $t = 30$ s (see Fig. 5). Left to right: simulations S1 (a, e and i), S2 (b, f and j), S3 (c, g and k) and S4 (d, h and l); see Table 4. Top (a–d): spectrum presented in the wavelength space, theoretical vertical wavelengths for infrasound caused by P waves at orthogonal incidence (red solid line) and for infrasound caused by Rayleigh waves (red dashed line). In panels (b)–(d), a vertical dotted line marks the horizontal topography period L_{topo} (see Table 4). Middle (e–h): spectrum presented in the equivalent frequency–incidence space (see description in Section 8.2); in this context the infrasound incidence corresponds to the launch angle from the ground (0° being vertical, 90° being horizontal towards positive x). Bottom (i–l): radiated infrasound energy as function of launch angle, in Pascals (see colour scale), that is the spectrum averaged in the infrasound band (0.5–2 Hz, highlighted in green in the middle panel); a black dashed line marks the maximum slopes present in each topography (see Table 4).

earthquake, that is $M_0 = 10^{1.5M_w+9.05} \simeq 6.31 \times 10^{12}$ J (Kanamori 1977). The depth, magnitude, and mechanism of the source are chosen according to the Pyrenean seismicity presented by Rigo *et al.* (2015). We choose the dominant frequency based on the magnitude and using empirical frequency–magnitude relations (Allen 2003). Based on the ground model and the chosen dominant frequency f_0 , the dominant seismic wavelength is $\lambda_0 = v_p/f_0 = 3039$ m. We note that the source depth z_0 is greater than one seismic wavelength λ_0 , that $\lambda_{\text{topo}}/\lambda_0 < 1$ for simulations S2 and S3, and that $\lambda_{\text{topo}}/\lambda_0 > 1$ for simulation S4.

8.2 Results with synthetic topographies

We perform coupled ground–atmosphere simulations for the four set-ups (see Table 4) using SPEC-FEM2D-DG-LNS. Fig. 5 presents the synthetic pressure perturbation field caused by the earthquake under the various topographies. Significant scattering occurs in simulations S2 and S3. In simulation S4, we can clearly see the focusing of acoustic rays owing to the curved interface, leading to constructive interference above valleys (at $x \simeq -10, 0$, or 10 km). Fig. 6 presents the computed 2-D spatial spectrum of those fields. The spectra are simply defined as 2-D spatial discrete Fourier transforms. The horizontal and vertical wavelengths are defined as $(2\pi/k_x, 2\pi/k_z)$, with $k_{x,z}$ the horizontal and vertical wavenumbers.

The control simulation S1 is dominated by a single wavenumber set, with horizontal wavelengths $\in [1.8, 5]$ km and vertical wavelengths $\in [160, 350]$ m. These vertical wavelengths are caused by the direct ground-to-air transmission of P waves in the near field (close to the epicentre) and Rayleigh waves in the far-field. Orthogonal incidence transmission at the epicentre yields an acoustic vertical wavelength of $c/f_0 \simeq 171$ m. Away from the epicentre, the difference in propagation speeds between the air and the ground results in acoustic vertical wavelengths of $c v_p / (f_0 v_r) \simeq 320$ m (where the Rayleigh wave speed $v_r \simeq 3242$ m s^{-1}). Large horizontal wavelengths exist at the epicentre due to the nearly instantaneous arrival of the P wave, while the seismic surface waves cause shorter ones slanted at an angle $\arctan(c/v_r) \simeq 6^\circ$.

Next, for simulations S2–S4, we perform a comparative analysis with simulation S1. Simulations S2 and S3 both exhibit significant acoustic scattering and verify $\lambda_{\text{topo}}/\lambda_0 < 1$. In contrast, S4 features a wavelength pattern very similar to S1, only shifted towards lower horizontal wavelengths; here $\lambda_{\text{topo}}/\lambda_0 > 1$ and next to no scattering can be observed. This confirms that a low $\lambda_{\text{topo}}/\lambda_0$ ratio causes the scattering of seismo-acoustic waves.

Furthermore, in simulations S2 and S3, the scattered infrasonic field exhibits clear interference patterns. Each horizontal seismic wavelength causes several acoustic vertical wavelengths, owing to the small topographic periods at play. For both S2 and S3 (see panels b) and c) of Fig. 6), the main mode is located around the main topographic wavelength (H_{topo}), and is repeated at integer multiples ($H_{\text{topo}}/2, H_{\text{topo}}/3,$

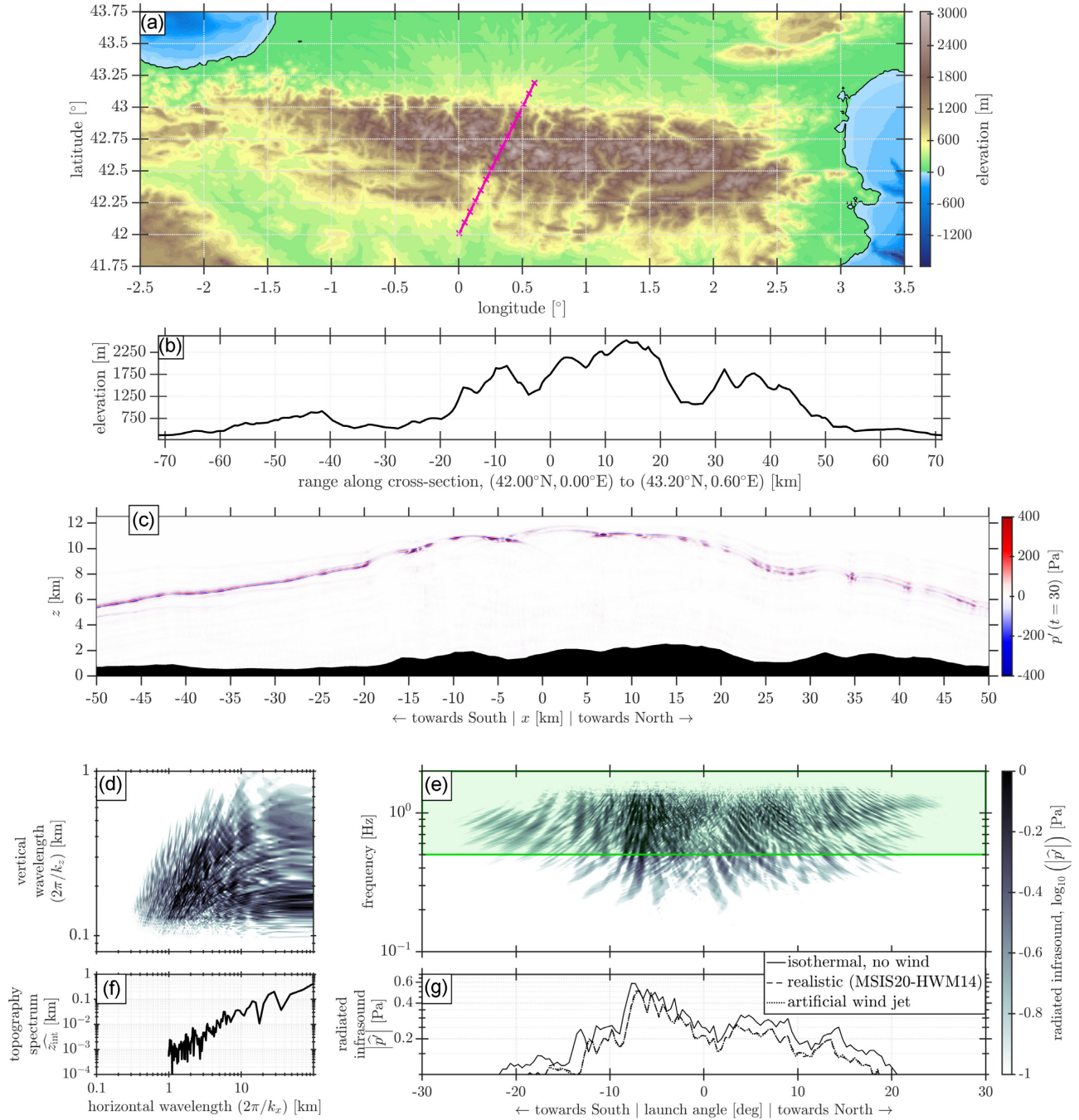


Figure 7. Summary of the scattering and radiation pattern study for the realistic topography case. (a and b) Chosen cross-section. (a) ETOPO1 (Amante & Eakins 2009) elevation map of the Pyrenees region; the chosen cross-section is highlighted by a purple solid line (with panel b)’s ticks superimposed). (b) Elevation profile along the chosen cross-section (see panel a), linearly interpolated from the ETOPO1 map. (c–f) Results from the SPEC-FEM2D-DG-LNS simulation under the isothermal atmosphere. Panels (c)–(f) for the realistic windy atmospheres do not differ significantly and are presented in Supporting Information Fig. S4 or S6. (g) Results under both the isothermal and realistic atmospheres. (c) SPEC-FEM2D-DG-LNS synthetic pressure field above the chosen topography, at $t = 30$ s. (d and e) Pressure perturbation spectrum computed as the 2-D Fourier transform of the pressure field. (d) One-sided pressure spectrum in the wavelength space $|\hat{p}'(-k_x, k_z)| + |\hat{p}'(k_x, k_z)|$. (e) Full two-sided pressure spectrum presented in the frequency-incidence space (see description in Section 8.2). (f) Topography spectrum computed as the Fourier transform of the fluid–solid interface z_{int} . (g) Amount of radiated energy, as function of launch angle, in the infrasound band (0.5–2 Hz, highlighted in green in panel e), for both the isothermal and the realistic atmospheric models.

$H_{\text{topo}}/4$, etc.) with decreasing energy. Since S2 features higher peaks (or deeper troughs), the same overall pattern can be seen, spanning up to larger vertical wavelengths.

Finally, we investigate the impact of slope angles. Fig. 6 also presents all pressure spectra in an equivalent ‘frequency-incidence’ space. Assuming each point in the wavelength space represents a local plane wave, its local incidence is:

$$\alpha = \arctan(k_x/k_z). \quad (90)$$

The corresponding wavelength, in the direction fully described by α , can be retrieved as:

$$(2\pi \cos(\alpha)/k_z + 2\pi \sin(\alpha)/k_x). \quad (91)$$

Denoting c the local speed of sound, this wavelength maps to a frequency:

$$c / (2\pi \cos(\alpha)/k_z + 2\pi \sin(\alpha)/k_x). \quad (92)$$

In our case, c is constant because of the isothermal atmosphere, making this process and the following analysis straightforward. However, note that SPECSEM2D-DG can account for realistic atmospheric models featuring varying speeds of sound and winds (Brissaud *et al.* 2017; Martire *et al.* 2018, 2020). We focus in particular on the infrasound band (0.5–2 Hz), and therefore consider the average amount of energy in this band as a function of incidence, which we also plot in Fig. 6.

In simulation S1, most infrasound is radiated upwards at incidences up to 6° , the launch angle of acoustic waves due to seismic surface waves (see previous paragraph). For both scattered and un-scattered cases, respectively in simulations S3 and S4, infrasound is radiated to larger incidence angles. Finally, in simulation S2, the significant scattering causes an almost isotropic infrasound radiation pattern.

The maximum infrasound launch angle is related to the maximum topographic slope. Most of the infrasonic energy is radiated at incidence angles lower than the maximum slope. Scattering spreads the radiation pattern past this limit, as seen in simulations S3 and S4.

Overall, for deep events (in which depth is greater than one seismic wavelength), plains (simulation S1) produce seismo-acoustic energy at vertical launch angles only, while mountains (simulations S2 to S4) radiate energy at larger launch angles. The atmospheric refraction of acoustic waves is known to be more efficient for large launch angles, therefore seismo-acoustic waves coming from mountainous regions are more prone to be recorded by ground stations. This property of seismo-acoustic couplings was already observed on data for earthquakes occurring under mountains (Le Pichon *et al.* 2006) or near cliffs (Green *et al.* 2009).

While the source depth is a crucial parameter for the acoustic radiation pattern of underground sources (Godin 2006, 2007; Averbuch *et al.* 2020), our simulations show that the impact of topography needs to be assessed. For instance, although the topography in simulation S3 is negligible compared to the seismic wavelength ($H_{\text{topo}}/\lambda_0 \ll 1$), the acoustic radiation pattern is clearly anisotropic (Fig. 6 k) due to scattering.

As Fig. 6 showcases, our simulations allow to retrieve the radiation pattern of seismically induced infrasound for a variety of topographies. This type of simulations can readily be extended to other source mechanisms, ground models, atmospheric models, etc. In Section 8.3, we thus decided to study a realistic topography.

8.3 Results with a realistic topography

We choose a cross-section of the Pyrenees, running from (42.00°N, 0°E) to (43.20°N, 0.60°E). We use the ETOPO1 (Amante & Eakins 2009) elevation map of Europe, which is a topography-bathymetry model with a 1 arc min resolution ($\simeq 1.8$ km). Panels (a) and (b) of Fig. 7 present a map and the interpolated altitude of the cross-section. The set-up is exactly the same as the one presented in Section 8.1 for the synthetic topographies. However, the interface z_{int} is now chosen according to the cross-section depicted in Fig. 7b, and the horizontal span of the simulated domain is extended to account for the whole profile (i.e. $x_{\text{max}} = -x_{\text{min}} \simeq 72$ km).

The isothermal atmospheric model does not feature any wind or intrinsic attenuation (viscous or thermal). Therefore, in addition to the baseline isothermal run, we run the same simulation with a realistic atmospheric model, computed from the MSIS20 and HWM14 models (Drob *et al.* 2015; Emmert *et al.* 2021) at the mid-point of the considered cross-section (42.6°N, 0.3°E), and on 2000/07/15 at 23:00 UTC. We smoothly set the winds to zero at altitudes where topography is present (altitudes $z < 2500$ m) since the topography modifies the characteristics of the boundary layer (Blom 2020) which is not accounted for in HWM14. The time of day for this model was chosen in order to have the strongest possible winds at this location. This model is depicted in Supporting Information Fig. S3.

As another addition, we also artificially introduce a strong jet at low altitudes to see its potential impact on the radiation pattern. To do so, we take the atmosphere previously extracted from MSIS20, and choose the wind field as a Gaussian jet of amplitude 50 m s^{-1} at altitude 5 km and with half-width 800 m:

$$w(z) = 50 \exp\left(-\left(\frac{z - 5000}{800}\right)^2\right). \quad (93)$$

This model is plotted in Supporting Information Fig. S5.

We start by analysing the results obtained with the isothermal atmosphere. In Fig. 7, panels c) to g) present the pressure field obtained with SPECSEM2D-DG-LNS, alongside the corresponding pressure spectrum. Panel (f) shows that the topography exhibits mostly horizontal wavelengths above 5 km, resulting in a ratio $\lambda_{\text{topo}}/\lambda_0 > 1$ and therefore little to no scattering (as expected, see synthetic case S4). However, we note that short horizontal wavelengths can be underestimated due to the 1 arc min ($\simeq 1.8$ km) resolution of this topography.

Fig. 7d shows that most of the infrasonic energy is contained in vertical wavelengths between 100 and 400 m, while horizontal wavelengths exist in the 0.5–5 km range. The spectrum pattern for this case is very similar to the spectrum of synthetic case S4 (Fig. 7d), showing that the distribution of horizontal acoustic wavelengths is not fully governed by the periodic contents of the topography. Finally, the relative lack of topographic wavelengths around $\simeq 17.8$ km (Fig. 7f) could be the cause of the relative decrease in acoustic energy at this horizontal wavelength (Fig. 7d).

Panels (e) and (g) of Fig. 7 highlight the infrasonic incidence angles, or launch angles. Seismo-acoustic waves are created at incidences up to 30° , showing a clearly asymmetric pattern. At low incidences (launch angles in $[-10, 10]^\circ$), the infrasonic energy is mostly directed towards the south (with launch angles in $[-10, 0]^\circ$). At higher incidences (launch angles in $[-20, -10]^\circ$ or $[10, 20]^\circ$), slightly more infrasonic energy is directed towards the north (with launch angles in $[10, 20]^\circ$). Finally, we remark that the launch angles are closely linked to the topography (Fig. 7b).

We now compare the isothermal atmosphere run to the realistic atmosphere run, and to the artificial wind jet run. Panels (c)–(e) of Fig. 7 can be compared to the corresponding ones in Supporting Information Figs S4 and S6. Fig. 7(g) presents the radiation patterns for all three cases. As expected for infrasound propagating mostly upward, the effects of horizontal winds are minimal. The dominant wind is mostly Southward in the realistic atmospheric model (Supporting Information Fig. S3) and northward in the strong jet atmospheric model (Supporting Information Fig. S5); however, we do not see any additional asymmetry introduced by windy conditions. The aspect of the pressure perturbation wavefields as well as their spectra do not change significantly between the different cases.

However, we can observe differences in the spectra's amplitudes (Fig. 7d/e, to be compared with Supporting Information Fig. S4d/e and Supporting Information Fig. S6d/e) and in the resulting radiation pattern (Fig. 7g). We ascribe these differences to the differences in atmospheric attenuation: the baseline model has no attenuation ($\alpha = 0$), whereas the two other models include realistic viscosities and thermal conductivities. At these altitudes, the realistic models amount to $\alpha < 10^{-9} \text{ m}^{-1} \simeq 10^{-5} \text{ dB km}^{-1}$ (see Supporting Information Figs S3 and S5). As expected from this value, the effect of this low attenuation remains relatively weak.

Finally, we note the span of launch angles is slightly reduced in the presence of wind (Fig. 7g). We attribute this effect to the fact that the waveforms are less prone to geometrical spreading because of their overall advection by wind, and this regardless of the wind strength.

9 CONCLUSION

In this contribution, we have presented the *LNS extension* and its implementation in the SPECFEM2D-DG software. With this addition, SPECFEM2D-DG features three main modules:

- (i) SPECFEM2D's visco-elastic module (Komatitsch 1997; Komatitsch & Vilotte 1998; Komatitsch & Tromp 1999; Peter *et al.* 2011), resolving elastic materials and seismic waves subject to attenuation.
- (ii) SPECFEM2D-DG-FNS (Brissaud *et al.* 2017), resolving fluid materials and nonlinear wave propagation in windy, attenuating, and stratified atmospheres.
- (iii) SPECFEM2D-DG-LNS (described in this paper), resolving fluid materials and linear atmospheric wave propagation.

Both fluid extensions (FNS and LNS) allow the modelling of complex viscous atmospheres, as well as gravity waves, infrasound, and high-frequency waves. Both fluid extensions are coupled to the visco-elastic module through two-way mechanical coupling conditions, allowing the resolution of waves travelling across the fluid–solid interface as well as general energy transmission through leaky modes. Finally, SPECFEM2D-DG is parallelized and benefits greatly from high-performance computing architectures.

The FNS extension (Brissaud *et al.* 2017) is essential for considering nonlinear propagation, involved in cases such as shock waves, high-amplitude events, or gravity wave breaking. Configurations in which the nonlinear phenomena take precedence, especially in terms of viscous attenuation effects, should be modelled with the FNS extension. The FNS extension is however limited to range-independent atmospheric models for stability concerns, and only profits from limited absorbing OBCs.

The LNS extension, modelling linear phenomena, has two main advantages. First, range-dependent mean states (i.e. atmospheric models) may be used, allowing the accurate resolution of regional acoustic propagation. Second, these linear decoupled equations enable the straightforward implementation of absorbing OBCs (presented in Section 5.1.2 and verified in Appendix B2). In addition, the LNS module was verified using the Method of Manufactured Solutions (Section 6) and analytical plane wave solutions (Appendix B1). Finally, fluid–solid CBCs were introduced in Section 5.2 and verified in Section 7.

The software SPECFEM2D-DG is fully implemented, functional, and readily usable in 2-D. The software (including its source code, the examples presented in this paper, and post-processing scripts) is freely available to the community on the GitHub platform at <https://github.com/samosa-project/specfem2d-dg>.

Using SPECFEM2D-DG-LNS, we simulated earthquakes under realistic topographies with varying wavelengths, and studied the atmospheric signature of seismically induced infrasound (Section 8). This illustrated the capability of the LNS extension to deal with unstructured meshes, complex topographies, and the solid-to-fluid coupling condition. This extends previous studies, in which our modelling of infrasound induced by seismic surface waves was shown to agree with field data on a flat topography (Martire *et al.* 2018) and a low-elevation topography (Garcia *et al.* 2021).

We have confirmed that a low ratio of seismic to topographic wavelengths ($\lambda_{\text{topo}}/\lambda_0$) causes a strong scattering of seismo-acoustic signals. By investigating the acoustic radiation pattern, we have numerically demonstrated why and how seismo-acoustic waves recorded by ground networks generally come from areas showcasing prominent topographic features (Le Pichon *et al.* 2006; Green *et al.* 2009). Furthermore, using a numerical case study, we concluded that the effect of horizontal wind and intrinsic attenuation on the infrasonic radiation pattern of seismic events is minimal, and that the prominent features are essentially due to the topography. These types of simulations are readily applicable to other planetary topographies, quake mechanisms and depths, atmospheric models, or even other geophysical configurations.

Including its visco-elastic module and its two atmospheric extensions, SPECFEM2D-DG can resolve a wide variety of geophysical phenomena, including but not limited to linear seismic waves subject to attenuation, gravity waves, infrasound, high-frequency acoustics, atmospheric shock waves, sound due to leaky seismic surface waves, or ground compliance effects.

Numerous geophysical events were already accurately modelled, such as the generation of seismically induced infrasound (Krishnamoorthy *et al.* 2018; Martire *et al.* 2018; Krishnamoorthy *et al.* 2019; Garcia *et al.* 2021), or the conversion of pressure perturbations to ground motion (Martire *et al.* 2020). In general, both large-scale and small-scale phenomena can be modelled, as well as atmospheric sources or surface-induced atmospheric perturbations.

Future investigations include studying the sensitivity of seismo-acoustic infrasound to other source mechanisms, source depths, topographies, or frequency contents. The seismo-acoustic amplification in and around sedimentary basins (Bordoni *et al.* 2012; Marchetti *et al.* 2016; Hernandez *et al.* 2018) can be modelled; the coupling and resonance of seismic waves within the basin can be fully accounted for, as well as the subsequent acoustic waves. Earthquakes are known to cause seismically induced sound to resonate in valleys (Sylvander *et al.* 2007); aside from reproducing the seismo-acoustic records, our simulations can pave the way towards an understanding of this peculiar phenomenon. Finally, Godin (2006, 2007) and Averbuch *et al.* (2020) argue that shallow sources under relatively flat topographies produce an isotropic seismo-acoustic radiation pattern; however, acoustic anisotropy is expected at least for deeper events (Section 8) and anisotropic seismic focal mechanisms (Martire *et al.* 2018).

As a next step concerning the development of accurate computational tools, efforts will be put towards a generalization of SPECFEM2D-DG to 3-D, also with the goal of releasing it open source to the community.

ACKNOWLEDGEMENTS

The authors acknowledge two reviewers for their constructive remarks. The authors thank the TGCC (Paris, France, project GENCI gen10476) and CALMIP (Toulouse, France, project #p1404) computing centres for HPC resources. The authors acknowledge both the ‘Direction Générale de l’Armement’ (French Directorate General of Armaments) and the ‘Région Occitanie Pyrénées-Méditerranée’ for funding the PhD grant of LM. *Author contribution statement:* LM developed the Linear Navier–Stokes extension, implemented it in the SPECFEM2D-DG software, designed and tested the verification cases, performed and analysed the application cases, and wrote the paper; RM provided an expertise in numerical modelling and helped design the absorbing boundary conditions; QB first introduced the discontinuous Galerkin method in SPECFEM2D-DG, implemented the Full Navier–Stokes extension SPECFEM2D-DG-FNS (Brissaud *et al.* 2017) and provided key help in developing SPECFEM2D-DG-LNS; RFG contributed to the conceptualization and analysis of the application cases; all co-authors thoroughly helped proof-reading the manuscript. The authors declare no conflict of interest.

DATA AVAILABILITY

The open-source software SPECFEM2D-DG is available open source on GitHub at: <https://github.com/samosa-project/specfem2d-dg>. The verification cases, the application cases and the scripts for post-processing all results are included in the repository.

REFERENCES

- Afanasyev, M., Boehm, C., van Driel, M., Krischer, L., Rietmann, M., May, D. A., Knepley, M. G. & Fichtner, A., 2019. Modular and flexible spectral-element waveform modelling in two and three dimensions, *Geophys. J. Int.*, **216**(3), 1675–1692.
- Afraimovich, E. L., Perevalova, N. P., Plotnikov, A. V. & Uralov, A. M., 2001. The shock-acoustic waves generated by earthquakes, *Ann. Geophys.*, **19**(4), 395–409.
- Aki, K. & Richards, P. G., 2002. *Quantitative Seismology*, 2nd edn, University Science Books.
- Allen, R. M., 2003. The Potential for Earthquake Early Warning in Southern California, *Science*, **300**(5620), 786–789.
- Altmann, C., Beck, A. D., Hindenlang, F., Staudenmaier, M., Gassner, G. J. & Munz, C. D., 2013. An efficient high performance parallelization of a discontinuous Galerkin spectral element method, in *Facing the Multicore-Challenge III*, chap. 4, pp. 37–47, eds Keller, R., Kramer, D. & Weiss, J.-P., Springer.
- Amante, C. & Eakins, B. W., 2009. ETOPO1: 1 Arc-Minute Global Relief Model: Procedures, Data Sources and Analysis (NOAA Technical Memorandum NESDIS NGDC-24), Tech. rep., National Geophysical Data Center, NOAA.
- Arina, R., 2016. Validation of a discontinuous Galerkin implementation of the time-domain linearized Navier–Stokes equations for aeroacoustics, *Aerospace*, **3**(1), 7, doi:10.3390/aerospace3010007.
- Averbuch, G., Waxler, R. M., Smets, P. S. M. & Evers, L. G., 2020. Probabilistic inversion for submerged source depth and strength from infrasound observations, *J. acoust. Soc. Am.*, **147**(2), 1066–1077.
- Baggag, A., Atkins, H. & Keyes, D., 1999. Parallel implementation of the discontinuous Galerkin method, Tech. Rep., Old Dominion University, Norfolk, VA 23529, USA.
- Banfield, D. J. *et al.*, 2019. InSight Auxiliary Payload Sensor Suite (APSS), *Space Sci. Rev.*, **215**(1), 4, doi:10.1007/s11214-018-0570-x.
- Bass, H.E. & Chambers, J.P., 2001. Absorption of sound in the Martian atmosphere, *The Journal of the Acoustical Society of America*, **109**, 3069–3071.
- Bassi, F. & Rebay, S., 1997. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations, *J. Comput. Phys.*, **131**, 267–279.
- Bécache, E., Givoli, D. & Hagstrom, T., 2010. High-order absorbing boundary conditions for anisotropic and convective wave equations, *J. Comput. Phys.*, **229**(4), 1099–1129.
- Benzoni-Gavage, S., Coulombel, J.-F. & Aubert, S., 2003. Boundary Conditions for Euler Equations, *AIAA J.*, **41**(1), doi:10.2514/2.1913.
- Benzoni-Gavage, S. & Serre, D., 2007. *Multi-dimensional Hyperbolic Partial Differential Equations: First-order Systems and Applications (Oxford Mathematical Monographs)*, Oxford University Press.
- Berland, J., Bogey, C. & Bailly, C., 2006. Low-dissipation and low-dispersion fourth-order Runge-Kutta algorithm, *Comput. Fluids*, **35**(10), 1459–1463.

- Blanc, E., Komatitsch, D., Chaljub, E., Lombard, B. & Xie, Z., 2016. Highly accurate stability-preserving optimization of the Zener viscoelastic model, with application to wave propagation in the presence of strong attenuation, *Geophys. J. Int.*, **205**(1), 414–426.
- Blom, P. S., 2014. GeoAc.
- Blom, P. S., 2020. The influence of irregular terrain on infrasonic propagation in the troposphere, *J. acoust. Soc. Am.*, **148**(4), 1984–1997.
- Booker, J. R. & Bretherton, F. P., 1967. The critical layer for internal gravity waves in a shear flow, *J. Fluid Mech.*, **27**(3), 513–539.
- Bordoni, P. *et al.*, 2012. Preliminary results from EMERSITO, a rapid response network for site-effect studies, *Ann. Geophys.*, **55**(4), 599–607.
- Brissaud, Q., 2017. Modélisation numérique des ondes atmosphériques issues des couplages solide/océan/atmosphère et applications, *PhD thesis*, Université Toulouse 3 Paul Sabatier.
- Brissaud, Q., Martin, R., Garcia, R. F. & Komatitsch, D., 2016. Finite-difference numerical modelling of gravitoacoustic wave propagation in a windy and attenuating atmosphere, *Geophys. J. Int.*, **206**(1), 308–327.
- Brissaud, Q., Martin, R., Garcia, R. F. & Komatitsch, D., 2017. Hybrid Galerkin numerical modelling of elastodynamics and compressible Navier–Stokes couplings: applications to seismo-gravito acoustic waves, *Geophys. J. Int.*, **210**(2), 1047–1069.
- Brissaud, Q., Preston, R. A., Ao, C. O., Bocanegra-Bahamon, T. M., Vergados, P., Komjathy, A. & Krishnamoorthy, S., 2019. Modeling seismically-induced acoustic waves on Venus, in *American Geophysical Union, Fall Meeting Abstract*, A21S–2810.
- Canuto, C. G., Hussaini, M. Y., Quarteroni, A. & Zang, T. A., 2007. *Spectral Methods: Evolution to Complex Geometries and Applications to Fluid Dynamics*, Springer.
- Carpenter, M. H. & Kennedy, C. A., 1994. Fourth-Order 2N-Storage Runge-Kutta Schemes, *Nasa Technical Memorandum*, **109112**, 1–26.
- Cremer, L., 1990. *Vorlesungen über Technische Akustik*, Springer Berlin Heidelberg.
- Cupillard, P., Delavaud, E., Burgos, G., Festa, G., Vilotte, J.-P., Capdeville, Y. & Montagner, J.-P., 2012. RegSEM: a versatile code based on the spectral element method to compute seismic wave propagation at the regional scale, *Geophys. J. Int.*, **188**(3), 1203–1220.
- Cutts, J. A. *et al.*, 2019. Roadmap for Venus, Tech. rep., LPI USRA.
- Dessa, J. X., Virieux, J. & Lambotte, S., 2005. Infrasonic modeling in a spherical heterogeneous atmosphere, *Geophys. Res. Lett.*, **32**(12), 1–5.
- Deville, M. O., Fischer, P. F. & Mund, E. H., 2002. *High Order Methods for Incompressible Fluid Flow*, Cambridge Univ. Press.
- Dmitrienko, I. S. & Rudenko, G. V., 2016. Oscillations of a vertically stratified dissipative atmosphere. I. Solution above source, *J. Atmos. Sol.-Terr. Phys.*, **142**, 120–136.
- Drob, D. P. *et al.*, 2015. An update to the Horizontal Wind Model (HWM): the quiet time thermosphere, *Earth Space Sci.*, **2**(7), 301–319.
- Edgar, N. B. & Visbal, M. R., 2003. A general buffer zone-type non-reflecting boundary condition for computational aeroacoustics, *American Institute of Aeronautics and Astronautics*, **3300**(May), 1–9.
- Emmert, J. T. *et al.*, 2021. NRLMSIS 2.0: a whole-atmosphere empirical model of temperature and neutral species densities, *Earth Space Sci.*, **8**(3), doi:10.1029/2020EA001321.
- Faccioli, E., Maggio, F., Paolucci, R. & Quarteroni, A., 1997. 2D and 3D elastic wave propagation by a pseudo-spectral domain decomposition method, *J. Seismol.*, **1**(3), 237–251.
- Faccioli, E., Maggio, F., Quarteroni, A. & Taghan, A., 1996. Spectral-domain decomposition methods for the solution of acoustic and elastic wave equations, *Geophysics*, **61**(4), 1160–1174.
- Fichtner, A., Bunge, H.-P. & Igel, H., 2006. The adjoint method in seismology, *Phys. Earth planet. Inter.*, **157**(1–2), 86–104.
- Fischer, P. F., 1990. Analysis and application of a parallel spectral element method for the solution of the Navier–Stokes equations, *Comput. Methods Appl. Mech. Eng.*, **80**(1–3), 483–491.
- Franca, L. P. & Frey, S. L., 1992. Stabilized finite element methods: II. The incompressible Navier–Stokes equations, *Comput. Methods Appl. Mech. Eng.*, **99**(2–3), 209–233.
- Francis, S. H., 1973. Acoustic-gravity modes and large-scale traveling ionospheric disturbances of a realistic, dissipative atmosphere, *J. geophys. Res.*, **78**(13), 2278–2301.
- Funaro, D., 2008. *Polynomial Approximation of Differential Equations*, Springer Science & Business Media.
- Garcia, R. F. *et al.*, 2021. An active source seismo-acoustic experiment using tethered balloons to validate instrument concepts and modelling tools for atmospheric seismology, *Geophys. J. Int.*, **225**, 186–199.
- Garcia, R.F., Brissaud, Q., Rolland, L.M., Martin, R., Komatitsch, D., Spiga, A., Lognonné, P.H. & Banerdt, W.B., 2017. Finite-Difference Modeling of Acoustic and Gravity Wave Propagation in Mars Atmosphere: Application to Infrasonics Emitted by Meteor Impacts, *Space Science Reviews*, **211**, 547–570.
- Giraldo, F. X. & Restelli, M., 2008. A study of spectral element and discontinuous Galerkin methods for the Navier–Stokes equations in nonhydrostatic mesoscale atmospheric modeling: Equation sets and test cases, *J. Comput. Phys.*, **227**(8), 3849–3877.
- Godin, O. A., 2006. Anomalous transparency of water-air interface for low-frequency sound, *Phys. Rev. Lett.*, **97**(16), 1–29.
- Godin, O. A., 2007. Transmission of low-frequency sound through the water-to-air interface, *Acoust. Phys.*, **53**(3), 305–312.
- Godin, O. A., 2014. Dissipation of acoustic-gravity waves: an asymptotic approach, *J. acoust. Soc. Am.*, **136**(6), EL411–EL417.
- Gokhberg, A. & Fichtner, A., 2016. Full-waveform inversion on heterogeneous HPC systems, *Comput. Geosci.*, **89**, 260–268.
- Green, D. N., Guilbert, J., le Pichon, A., Sebe, O. & Bowers, D., 2009. Modelling ground-to-air coupling for the shallow ML 4.3 Folkestone, United Kingdom, earthquake of 28 April 2007, *Bull. seism. Soc. Am.*, **99**(4), 2541–2551.
- Hedin, A. E., 1991. Extension of the MSIS thermosphere model into the middle and lower atmosphere, *J. geophys. Res.*, **96**(A2), 1159–1172.
- Hernandez, B. *et al.*, 2018. Estimating the ground-motion distribution of the 2016 Mw 6.2 Amatrice, Italy, earthquake using remote infrasonic observations, *Seismol. Res. Lett.*, **89**(6), 2227–2236.
- Hesthaven, J. S. & Warburton, T., 2008. *Nodal Discontinuous Galerkin Methods—Algorithms, Analysis, and Applications*, Vol. 54, Springer.
- Hu, F. Q., 2005. A perfectly matched layer absorbing boundary condition for linearized Euler equations with a non-uniform mean flow, *J. Comput. Phys.*, **208**(2), 469–492.
- Hu, F. Q., Li, X. D. & Lin, D. K., 2008. Absorbing boundary conditions for nonlinear Euler and Navier–Stokes equations based on the perfectly matched layer technique, *J. Comput. Phys.*, **227**(9), 4398–4424.
- Jawahar, P. & Kamath, H., 2000. A high-resolution procedure for euler and Navier–Stokes computations on unstructured grids, *J. Comput. Phys.*, **164**(1), 165–203.
- Jones, R. M., Riley, J. P. & Georges, T. M., 1986. *HARPA: A Versatile Three-Dimensional Hamiltonian Ray-Tracing Program for Acoustic Waves in the Atmosphere above Irregular Terrain*, Wave Propagation Laboratory.
- Kanamori, H., 1977. The energy release in great earthquakes, *J. geophys. Res.*, **82**(20), 2981–2987.
- Kenda, B. *et al.*, 2020. Subsurface structure at the InSight landing site from compliance measurements by SEIS and APSS, *J. geophys. Res.*, **125**, 1–30.
- Kenda, B., Lognonné, P. H., Spiga, A., Kawamura, T., Kedar, S., Banerdt, W. B., Lorenz, R. D., Banfield, D. J. & Golombek, M. P., 2017. Modeling of ground deformation and shallow surface waves generated by martian dust devils and perspectives for near-surface structure inversion, *Space Sci. Rev.*, **211**(1–4), 501–524.
- Kherani, E. A. *et al.*, 2012. Modelling of the total electronic content and magnetic field anomalies generated by the 2011 Tohoku-Oki tsunami and associated acoustic-gravity waves, *Geophys. J. Int.*, **191**(3), 1049–1066.
- Kherani, E. A., Rolland, L. M., Lognonné, P. H., Sladen, A., Klausner, V. & de Paula, E. R., 2016. Traveling ionospheric disturbances propagating ahead of the Tohoku-Oki tsunami: a case study, *Geophys. J. Int.*, **204**(2), 1148–1158.
- Knight, H. K., Broutman, D. & Eckermann, S. D., 2019. A causality-preserving Fourier method for gravity waves in a viscous, thermally diffusive, and vertically varying atmosphere, *Wave Motion*, **88**, 226–256.

- Komatitsch, D., 1997. Méthodes spectrales et éléments spectraux pour l'équation de l'élastodynamique 2D et 3D en milieu hétérogène, *PhD thesis*, Institut de Physique du Globe de Paris.
- Komatitsch, D. & Tromp, J., 1999. Introduction to the spectral element method for three-dimensional seismic wave propagation, *Geophys. J. Int.*, **139**, 806–822.
- Komatitsch, D. & Vilotte, J.-P., 1998. The spectral element method : an efficient tool to simulate the seismic response of 2D and 3D geological structures, *Bull. seism. Soc. Am.*, **88**(2), 368–392.
- Krank, B., Fehn, N., Wall, W. A. & Kronbichler, M., 2017. A high-order semi-explicit discontinuous Galerkin solver for 3D incompressible flow with application to DNS and LES of turbulent channel flow, *J. Comput. Phys.*, **348**, 634–659.
- Krishnamoorthy, S. et al., 2018. Detection of artificially generated seismic signals using balloon-borne infrasound sensors, *Geophys. Res. Lett.*, **45**(8), 3393–3403.
- Krishnamoorthy, S. et al., 2019. Aerial seismology using balloon-based barometers, *IEEE Trans. Geosci. Remote Sens.*, **57**(12), 10191–10201.
- Kubatko, E. J., Bunya, S., Dawson, C., Westerink, J. J. & Mirabito, C., 2008. A performance comparison of continuous and discontinuous finite element shallow water models, Tech. rep., The University of Texas at Austin.
- Lai, X. & Monnier, J., 2009. Assimilation of spatially distributed water levels into a shallow-water flood model. Part I: mathematical method and test case, *J. Hydrol.*, **377**(1–2), 1–11.
- Lalande, J.-M., Sèbe, O., Landès, M., Blanc-Benon, P. H., Matoza, R. S., Le Pichon, A. & Blanc, E., 2012. Infrasound data inversion for atmospheric sounding, *Geophys. J. Int.*, **190**(1), 687–701.
- Landau, L. D. & Lifshitz, E. M., 1987. *Fluid Mechanics—Volume 6 of Course of Theoretical Physics*, 2nd edn, Pergamon Press.
- Le Pichon, A., Mialle, P., Guilbert, J. & Vergoz, J., 2006. Multistation infrasonic observations of the Chilean earthquake of 2005 June 13, *Geophys. J. Int.*, **167**(2), 838–844.
- Lin, D. K., Li, X. D. & Hu, F. Q., 2011. Absorbing boundary condition for nonlinear Euler equations in primitive variables based on the Perfectly Matched Layer technique, *Comput. Fluids*, **40**(1), 333–337.
- Liu, H.-P., Anderson, D. L. & Kanamori, H., 1976. Velocity dispersion due to anelasticity; implications for seismology and mantle composition, *Geophys. J. R. astr. Soc.*, **47**(1), 41–58.
- Lognonné, P. H. et al., 2019. SEIS: InSight's seismic experiment for internal structure of Mars, *Space Sci. Rev.*, **215**(1), 12, doi:10.1007/s11214-018-0574-6.
- Lognonné, P. H. et al., 2020. Constraints on the shallow elastic and anelastic structure of Mars from InSight seismic data, *Nat. Geosci.*, **13**(3), 213–220.
- Lognonné, P. H., Clévédy, E. & Kanamori, H., 1998. Computation of seismograms and atmospheric oscillations by normal-mode summation for a spherical earth model with realistic atmosphere, *Geophys. J. Int.*, **135**(2), 388–406.
- Maday, Y. & Rønquist, E. M., 1990. Optimal error analysis of spectral methods with emphasis on non-constant coefficients and deformed geometries, *Comput. Methods Appl. Mech. Eng.*, **80**(1–3), 91–115.
- Madec, R., 2010. Méthodes des éléments spectraux pour la propagation des ondes sismiques en milieu géologique fluide-solide avec pas de temps locaux et couches absorbantes parfaitement adaptées C-PML, *PhD thesis*, Université de Pau et des Pays de l'Adour.
- Maess, M., Wagner, N. & Gaul, L., 2006. Dispersion curves of fluid filled elastic pipes by standard FE models and eigenpath analysis, *J. Sound Vib.*, **296**(1–2), 264–276.
- Marchetti, E., Lacanna, G., Le pichon, A., Piccinini, D. & Ripepe, M., 2016. Evidence of large infrasonic radiation induced by earthquake interaction with alluvial sediments, *Seismol. Res. Lett.*, **87**(3), 678–684.
- Martire, L. et al., 2018. Numerical simulation of the atmospheric signature of artificial and natural seismic events, *Geophys. Res. Lett.*, **45**(21), 12 085–12 093.
- Martire, L., Garcia, R. F., Rolland, L. M., Spiga, A., Lognonné, P. H., Banfield, D. J., Banerdt, W. B. & Martin, R., 2020. Martian Infrasound: numerical modeling and analysis of InSight's data, *J. geophys. Res.*, doi:10.1029/2020JE006376.
- Mengaldo, G., De Grazia, D., Peiro, J., Farrington, A., Witherden, F., Vincent, P. E. & Sherwin, S. J., 2014. A guide to the implementation of boundary conditions in compact high-order methods for compressible aerodynamics, in *7th AIAA Theoretical Fluid Mechanics Conference*, American Institute of Aeronautics and Astronautics, Atlanta, GS, p. 33.
- Midgley, J. E. & Liemohn, H. B., 1966. Gravity waves in a realistic atmosphere, *J. geophys. Res.*, **71**(15), 3729–3748.
- Molteni, F., Buizza, R., Palmer, T. N. & Petroliagis, T., 1996. The ECMWF ensemble prediction system: methodology and validation, *Q. J. R. Meteorol. Soc.*, **122**(529), 73–119.
- Moret-Gabarro, L., 2009. Aeroacoustic investigation and adjoint analysis of subsonic cavity flows—Part III, *PhD thesis*, Université de Toulouse.
- Pasyanos, M. E., Masters, T. G., Laske, G. & Ma, Z., 2014. LITHO1.0: an updated crust and lithospheric model of the Earth, *J. geophys. Res.*, **119**(3), 2153–2173.
- Patera, A. T., 1984. A spectral element method for fluid dynamics: laminar flow in a channel expansion, *J. Comput. Phys.*, **54**(3), 468–488.
- Peter, D. et al., 2011. Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes, *Geophys. J. Int.*, **186**(2), 721–739.
- Poinsot, T. J. & Lele, S. K., 1992. Boundary conditions for direct simulations of compressible viscous flows, *J. Comput. Phys.*, **101**(1), 104–129.
- Priolo, E., Carcione, J. M. & Seriani, G., 1994. Numerical simulation of interface waves by high-order spectral modeling techniques, *J. acoust. Soc. Am.*, **95**(2), 681–693.
- Rigo, A. et al., 2015. Present-day deformation of the Pyrenees revealed by GPS surveying and earthquake focal mechanisms until 2011, *Geophys. J. Int.*, **201**(2), 947–964.
- Roache, P. J., 2002. Code verification by the method of manufactured solutions, *J. Fluids Eng.*, **124**(1), 4–10.
- Rogers, P. H. & Gardner, J. H., 1980. Propagation of sonic booms in the thermosphere, *J. acoust. Soc. Am.*, **67**(1), 78–91.
- Roy, C. J., Nelson, C. C., Smith, T. M. & Ober, C. C., 2004. Verification of Euler/Navier–Stokes codes using the method of manufactured solutions, *Int. J. Numer. Methods Fluids*, **44**(6), 599–620.
- Roy, C. J., Smith, T. M. & Ober, C. C., 2002. Verification of a compressible CFD code using the method of manufactured solutions, in *32nd AIAA Fluid Dynamics Conference and Exhibit*, American Institute of Aeronautics and Astronautics, Reston, Virginia.
- Sabatini, R., Snively, J. B., Bailly, C., Hickey, M. P. & Garrison, J. L., 2019. Numerical modeling of the propagation of infrasonic acoustic waves through the turbulent field generated by the breaking of mountain gravity waves, *Geophys. Res. Lett.*, **46**(10), 5526–5534.
- Sanders, B. F. & Katopodes, N. D., 2000. Adjoint sensitivity analysis for shallow-water wave control, *J. Eng. Mech.*, **126**(9), 909–919.
- Seriani, G. & Priolo, E., 1994. Spectral element method for acoustic wave simulation in heterogeneous media, *Finite Elem. Anal. Des.*, **16**(3–4), 337–348.
- Shaw, W. N. & Dines, W. H., 1905. The study of the minor fluctuations of atmospheric pressure, *Q. J. R. Meteorol. Soc.*, **31**(133), 39–52.
- Solin, P., Segeth, K. & Dolezel, I., 2003. *Higher-Order Finite Element Methods*, Chapman & Hall.
- Sorrells, G. G., 1971. A preliminary investigation into the relationship between long-period seismic noise and local fluctuations in the atmospheric pressure field, *Geophys. J. R. astr. Soc.*, **26**(1–4), 71–82.
- Sorrells, G. G., McDonald, J. A., Der, Z. A. & Herrin, E. T., 1971. Earth motion caused by local atmospheric pressure changes, *Geophys. J. R. astr. Soc.*, **26**(1–4), 83–98.
- Stevanović, J., Teanby, N. A., Wookey, J., Selby, N., Daubar, I. J., Vaubaillon, J. & Garcia, R. F., 2017. Bolide airbursts as a seismic source for the 2018 Mars InSight Mission, *Space Sci. Rev.*, **211**(1–4), 525–545.
- Stevenson, D., Cutts, J. A. & Mimoun, D., 2015. Probing the Interior Structure of Venus, Tech. rep., Keck Institute for Space Studies, Pasadena, CA 91106, USA.
- Sylvander, M., Ponsolles, C., Benahmed, S. & Fels, J.-F., 2007. Seismoacoustic recordings of small earthquakes in the Pyrenees: experimental results, *Bull. seism. Soc. Am.*, **97**(1B), 294–304.

- Terrana, S., Vilotte, J.-P. & Guillot, L., 2018. A spectral hybridizable discontinuous Galerkin method for elastic-acoustic wave propagation, *Geophys. J. Int.*, **213**(1), 574–602.
- Tessmer, E. & Kosloff, D., 1994. 3-D elastic modeling with surface topography by a Chebychev spectral method, *Geophysics*, **59**(3), 464–473.
- Thompson, K. W., 1990. Time-dependent boundary conditions for hyperbolic systems, II, *J. Comput. Phys.*, **89**, 439–461.
- Touloupoulos, I. & Ekaterinaris, J. A., 2011. Artificial boundary conditions for the numerical solution of the Euler equations by the discontinuous galerkin method, *J. Comput. Phys.*, **230**(15), 5974–5995.
- Tromp, J., Komatitsch, D. & Liu, Q., 2008. Spectral-element and adjoint methods in seismology, *Commun. Comput. Phys.*, **3**, 1–32.
- Walker, K. T., Pichon, A. L., Kim, T. S., de Groot-Hedlin, C. D., Che, I.-Y. & Garcés, M. M., 2013. An analysis of ground shaking and transmission loss from infrasound generated by the 2011 Tohoku earthquake, *J. geophys. Res.*, **118**(23), 831–12.
- Williamson, J. H., 1980. Low-storage Runge-Kutta schemes, *J. Comput. Phys.*, **35**(1), 48–56.
- Woods, D. C., Bolton, J. S. & Rhoads, J. F., 2015. On the use of evanescent plane waves for low-frequency energy transmission across material interfaces, *J. acoust. Soc. Am.*, **138**(4), 2062–2078.
- Xie, Z., Komatitsch, D., Martin, R. & Matzen, R., 2014. Improved forward wave propagation and adjoint-based sensitivity kernel calculations using a numerically stable finite-element PML, *Geophys. J. Int.*, **198**(3), 1714–1747.
- Xie, Z., Matzen, R., Cristini, P., Komatitsch, D. & Martin, R., 2015. A perfectly matched layer for fluid-solid problems: application to ocean-acoustics simulations with solid ocean bottoms, *J. acoust. Soc. Am.*, **140**(1), 165–175.
- Yoon, K. T. & Chung, T. J., 1996. Three-dimensional mixed explicit-implicit generalized Galerkin spectral element methods for high-speed turbulent compressible flows, *Comput. Methods Appl. Mech. Eng.*, **135**(3–4), 343–367.

SUPPORTING INFORMATION

Supplementary data are available at [GJI](https://doi.org/10.1017/gji.2021.174) online.

Table S1: Detailed scalar expressions of the LNS’ strong formulation terms, as defined in eq. (2). These play a role in the volume terms of the weak formulation detailed in Appendix B1. The notations are that of eq. (2), as well as some additional ones provided in Section 3. Here, only the case $d = 2$ is presented. Generalization to $d = 3$ is straightforward, though more tedious to write down.

Figure S1: Time-series comparison for the validation of the fluid–solid coupling. (a and b) Fluid-to-solid cases. (c and d) Solid-to-fluid cases. (a and c) Orthogonal incidence ($i_1 = i_2 = \theta_i = 0^\circ$). (b and d) Slanted incidence ($i_1 = i_2 = \theta_i \neq 0^\circ$). On all plots, time progression is coloured for illustrative purposes. P_2^u denotes the velocity amplitude of the upgoing elastic P wave in the S2F case, while P_1^d denotes the velocity amplitude of the downgoing acoustic P wave in the F2S case. For each panel, the subpanel (i) displays pressure perturbation time-series, the subpanel (ii) shows horizontal/vertical ground motion, and the subpanel (iii) presents the ground motion polarization. On sub-panels (i), the red dotted lines represent either the expected reflected amplitude $R_F^d P_1^d$ for the F2S case (panels a and b) or the expected transmitted amplitude $Z_1 T_{PP}^{SF} P_2^u$ for the S2F case (panels c and d); see Table 2. On sub-panels (iii), the dotted lines highlight the expected incident (black, if applicable), and transmitted/reflected P (red) or S (green) angles; the respective expected amplitudes (see Table 2) are displayed with circle segments.

Figure S2: Arina (2016)’s stretching functions. The stretching function is defined in eq. (D.12). The dependency of the function on its parameters (p , q and ϵ) is illustrated here. The choice used for the validation presented in Appendix D2 is the blue thick solid line.

Figure S3: Realistic atmospheric model associated to the realistic topography case. It is computed from the MSIS20 and HWM14 models (Drob et al. 2015; Emmert et al. 2021), at the mid-point of the considered cross-section (42.6°N; 0.3°E), and on 2000/07/15 at 23:00 UTC (chosen due to get the strongest winds). The ‘leftward’ and ‘rightward’ keywords are given in the context of this simulation: ‘rightward’ means towards the nrth, and ‘leftward’ means towards the south. The leftward propagation undergoes the strongest winds. The attenuation coefficients (in dB km⁻¹) are computed using the relations presented, for example, in the works of Bass & Chambers (2001) or Garcia et al. (2017). Vibrational attenuation is negligible here $\alpha_{\text{vib}} \simeq 0$.

Figure S4: Summary of the scattering and radiation pattern study for the realistic topography case, with the atmosphere presented in Fig. S3. See also Fig. 7 in the main text for comparison. For the chosen cross-section, see panels (a) and (b) of Fig. 7 in the main text. (c–g) Results from the SPECFEM2D-DG-LNS simulation under the realistic atmosphere. (c–f) The isothermal atmosphere do not differ significantly and are presented in Fig. 7 in the main text. (c) SPECFEM2D-DG-LNS synthetic pressure field above the chosen topography, at $t = 30$ s. (d and e) Pressure perturbation spectrum computed as the 2-D Fourier transform of the pressure field. (d) One-sided pressure spectrum in the wavelength space $|\hat{p}'(-k_x, k_z)| + |\hat{p}'(k_x, k_z)|$. (e) Full two-sided pressure spectrum presented in the frequency-incidence space (see description in Section 8.2). (f) Topography spectrum computed as the Fourier transform of the fluid–solid interface z_{int} . (g) Amount of radiated energy, as function of launch angle, in the infrasound band [0.5–2 Hz, highlighted in green in panel (e)], for the realistic atmospheric model.

Figure S5: Realistic atmospheric model associated to the realistic topography case with wind jet. It is exactly the same model as the one presented in Fig. S3 (see it for a description of the panels), apart from the wind. The wind here is chosen as a Gaussian jet of amplitude 50 m s⁻¹ at altitude 5 km and with half-width 800 m: $50 \exp\left(-\left(\frac{z-5000}{800}\right)^2\right)$.

Figure S6: Summary of the scattering and radiation pattern study for the realistic topography case, with the atmosphere presented in Fig. S5. See also Fig. 7 in the main text for comparison. See caption of Fig. S4 for a description of the panels.

Please note: Oxford University Press is not responsible for the content or functionality of any supporting materials supplied by the authors. Any queries (other than missing material) should be directed to the corresponding author for the paper.

APPENDIX A: REFLECTION-TRANSMISSION COEFFICIENTS AT A FLUID–SOLID INTERFACE

We consider the reflection and transmission of waves across a fluid–solid interface. We follow the classical developments introduced by (Aki & Richards 2002, section 5.2.4). Let s denote the horizontal slowness vector, assumed constant at the interface, and defined using Snell's relation:

$$s = \frac{\sin i_1}{c} = \frac{\sin i_2}{v_p} = \frac{\sin j_2}{v_s}, \quad (\text{A1})$$

where c is the fluid P -wave velocity, v_p is the solid P -wave velocity, v_s is the solid S -wave velocity, i_1 is the propagation angle of the fluid P wave (either incident, reflected, or transmitted), i_2 is the propagation angle of the solid P wave (either incident, reflected, or transmitted), and j_2 is the propagation angle of the solid S wave (either reflected or transmitted). For the following formulas, it is useful to define an intermediate variable, homogeneous to a speed:

$$W = v_s^3 s^2 \cos(i_2) \cos(j_2) + v_p (v_s^2 s^2 - 1/2)^2. \quad (\text{A2})$$

For the fluid-to-solid (F2S) case, the P to P transmission coefficient T_{PP}^{FS} , the P to S transmission coefficient T_{PS}^{FS} , and the reflection coefficient R_P^{F} respectively write:

$$T_{PP}^{\text{FS}} = \frac{-4\bar{\rho}_0 \cos(i_1) (v_s^2 s^2 - 1/2) c}{4W \rho^s \cos(i_1) + \cos(i_2) \bar{\rho}_0 c}, \quad (\text{A3})$$

$$T_{PS}^{\text{FS}} = \frac{-4v_s s \cos(i_2) \bar{\rho}_0 c \cos(i_1)}{4W \rho^s \cos(i_1) + \cos(i_2) \bar{\rho}_0 c}, \quad (\text{A4})$$

$$R_P^{\text{F}} = \frac{4W \rho^s \cos(i_1) - \cos(i_2) \bar{\rho}_0 c}{4W \rho^s \cos(i_1) + \cos(i_2) \bar{\rho}_0 c}. \quad (\text{A5})$$

Likewise, for the solid-to-fluid (S2F) case, the P to P transmission coefficient T_{PP}^{SF} , the P to P reflection coefficient R_{PP}^{S} , and the P to S reflection coefficient R_{PS}^{S} respectively write:

$$T_{PP}^{\text{SF}} = \frac{-4 \cos(i_2) v_p (v_s^2 s^2 - 1/2) \rho^s}{4W \rho^s \cos(i_1) + \cos(i_2) \bar{\rho}_0 c}, \quad (\text{A6})$$

$$R_{PP}^{\text{S}} = \frac{-4 \left(-v_s^3 s^2 \cos(i_2) \cos(j_2) + v_p (v_s^2 s^2 - 1/2)^2 \right) \rho^s \cos(i_1) + \cos(i_2) \bar{\rho}_0 c}{4W \rho^s \cos(i_1) + \cos(i_2) \bar{\rho}_0 c}, \quad (\text{A7})$$

$$R_{PS}^{\text{S}} = \frac{-8 \cos(i_2) v_p v_s \cos(i_1) s (v_s^2 s^2 - 1/2) \rho^s}{4W \rho^s \cos(i_1) + \cos(i_2) \bar{\rho}_0 c}. \quad (\text{A8})$$

Setting $i_1 = i_2 = 0$ (j_2 has no meaning since no S wave can exist at orthogonal incidences), $s = 0$, $W = v_p/4$, and one recovers the classical Fresnel coefficients applicable to a reflection/transmission line model.

APPENDIX B: ADDITIONAL VERIFICATION

B.1 Verification of the fluid equations using the FK method (plane wave analytical solutions)

Consider eq. (4) with the third equation in pressure instead of energy (Arina 2016):

$$\partial_t p' + \nabla \cdot (\gamma p_0 \mathbf{v}' + p' \mathbf{v}_0) = \nabla \cdot ((\gamma - 1) \kappa \nabla T') + (\gamma - 1) (\mathbf{v}' \cdot \nabla p_0 - p' \nabla \cdot \mathbf{v}_0 + (\Sigma_v(\mathbf{v}_0) \cdot \nabla) \cdot \mathbf{v}' + (\Sigma_v(\mathbf{v}') \cdot \nabla) \cdot \mathbf{v}_0), \quad (\text{B1})$$

The two versions of the third equation are exactly equivalent if we use either the last equation of eq. (4) with the closure relation eq. (15), or eq. (B1) with the closure relation eq. (16).

Let us assume plane wave solutions under the form:

$$(\rho', \mathbf{v}', p')(\mathbf{x}, t) = (\tilde{\rho}, \tilde{\mathbf{v}}, \tilde{p}) \exp(\mathbf{i}(\mathbf{k} \cdot \mathbf{x} - \omega t)), \quad (\text{B2})$$

with $\mathbf{k} := [k_1, \dots, k_d]$ the wavenumber vector. By successive Fourier transforms from the (t, \mathbf{x}) space to the (ω, \mathbf{k}) space, eq. (4) can trivially be rewritten under the following prototype linear system in the (ω, \mathbf{k}) space:

$$A(\omega, \mathbf{k}) \begin{pmatrix} \tilde{\rho} \\ \tilde{\mathbf{v}} \\ \tilde{p} \end{pmatrix} = \mathbf{b}. \quad (\text{B3})$$

In some cases, analytical solutions to eq. (B3) in the (ω, \mathbf{k}) domain may be obtained. We call these solutions *FK solutions*, for frequency–wavenumber solutions. From the FK solutions, a solution in the time domain is obtained by discrete direct inverse Fourier transforms [see e.g. the work of Brissaud *et al.* (2016, section 5.1) for a detailed rationale].

If viscosity is kept ($\min(\mu, \lambda, \zeta, \kappa) > 0$), the linear system eq. (B3) can still be written, but the coefficients of the matrix A become nonlinear in (ω, \mathbf{k}) . This is because the viscous terms in eq. (4) are second-order derivatives in space. The determination of the FK solutions becomes a more difficult problem, which shall not be detailed here. Rather, we refer the reader to the work of Godin (2014) or Dmitrienko & Rudenko (2016). The work of Knight *et al.* (2019) can also be useful for those more specifically interested in gravity waves. In the following, viscosity shall be neglected by setting $\mu = \lambda = \zeta = \kappa = 0$, for the sake of simplicity. We consider viscosity verified thanks to the method of manufactured solutions (Section 6).

In the remaining of this Section, the computational domain is chosen as:

$$D = [-375, 375] \text{ m} \times [0, 80] \text{ km}, \quad (\text{B4})$$

and discretized with a step $\Delta x = 250$ m. The source is chosen as an homogeneous bottom vertical velocity forcing defined by:

$$\forall t, \quad \forall x, \quad [\mathbf{v}]_d(t, x, z) = \partial_t G_{1/P_0}(t - t_0) \delta_0(z), \quad (\text{B5})$$

with G_f the Gaussian function defined by:

$$\begin{aligned} G_{f_0} : t &\mapsto \exp(-(\pi f_0 t)^2), \\ \partial_t G_{f_0} : t &\mapsto -2(\pi f_0)^2 t \exp(-(\pi f_0 t)^2). \end{aligned} \quad (\text{B6})$$

Typically, eq. (B5) is a plane wave forcing the bottom boundary. The fundamental period is chosen as $P_0 = 5$ s (dominant frequency $f_0 = 0.2$ Hz). The time delay is chosen as $t_0 = 10$ s.

Two background atmospheric models are tested, and constitute two test cases. First, in Appendix B1.1, the isobaric model (Table 1) is considered. Then, an isothermal model (Table 1) is presented in Appendix B1.2. In both cases, we let the wind be defined as $\mathbf{v}_0 = w(1, 0)^T$ with $w \in \mathbb{R}$; this choice has no impact since the LNS equations eq. (4) are isotropic. Finally, results are detailed and discussed in Appendix B1.3.

B1.1 Isobaric model

Let the model be isobaric (Table 1), with $\gamma = 1.401$, $\bar{\rho}_0 = 1.4 \text{ kg m}^{-3}$, $\bar{c} = 371 \text{ m s}^{-1}$. The matrix A and right-hand side (RHS) \mathbf{b} involved in the linear system eq. (B3) are:

$$A(\omega, \mathbf{k}) = \begin{pmatrix} wk_1 - \omega & k_1 \rho_0 & \rho_0 k_2 & 0 \\ 0 & (wk_1 - \omega) \rho_0 & \rho_0 wk_2 & k_1 \\ 0 & 0 & -\omega \rho_0 & k_2 \\ 0 & c^2 k_1 \rho_0 & c^2 \rho_0 k_2 & wk_1 - \omega \end{pmatrix}, \quad \mathbf{b} = 0. \quad (\text{B7})$$

The RHS being zero, this system only has non-trivial solutions if $|A| = 0$ (Midgley & Liemohn 1966; Francis 1973; Godin 2014), that is if

$$((c^2 - w^2)(k_1)^2 + 2k_1 \omega w + c^2 k_2^2 - \omega^2)(k_1 w - \omega) = 0. \quad (\text{B8})$$

The only four (two unique, and one double) solutions to eq. (B8) in terms of (ω, \mathbf{k}) are:

$$\begin{cases} S_1 := (0, k_1, k_2), \\ S_2 := (wk_1, k_1, k_2), \\ S_3 := \left(\omega, k_1, \pm \sqrt{\left(\frac{\omega - wk_1}{c}\right)^2 - k_1^2} \right), \end{cases} \quad (\text{B9})$$

where c is the speed of sound, and w is the horizontal wind. In eq. (B9), mode S_1 represents a standing wave field (with wavenumber \mathbf{k}) constant with time. S_2 represents a wave field (with wavenumber \mathbf{k}) horizontally transported by the constant wind. S_3 is a wave (with pulsation ω and horizontal wavenumber k_1) whose vertical propagation is modified by wind. Because of the trivial nature of the FK solutions $S_{1,2}$, only the mode S_3 is tested.

B1.2 Isothermal model

Let the model be isothermal (Table 1), with $\gamma = 1.401$, $\bar{\rho}_0 = 1.4 \text{ kg m}^{-3}$, $[\bar{\mathbf{g}}]_d = g_z = 9.81 \text{ m s}^{-2}$, $\bar{H} = 10 \text{ km}$, and no viscosity ($\bar{\mu} = \bar{\kappa} = 0$). Note that under these conditions $c \simeq 371 \text{ m s}^{-1}$. The linear system eq. (B3) has for matrix and RHS:

$$A(\omega, \mathbf{k}) = \begin{pmatrix} wk_1 - \omega & \bar{\rho}_0 k_1 e^{-\frac{z}{\bar{H}}} & e^{-\frac{z}{\bar{H}}} \bar{\rho}_0 k_2 & 0 \\ 0 & (wk_1 - \omega) \bar{\rho}_0 e^{-\frac{z}{\bar{H}}} & e^{-\frac{z}{\bar{H}}} \bar{\rho}_0 w k_2 & k_1 \\ \mathbf{i}g_z & 0 & -e^{-\frac{z}{\bar{H}}} \omega \bar{\rho}_0 & k_2 \\ 0 & g_z H \gamma \bar{\rho}_0 k_1 e^{-\frac{z}{\bar{H}}} & H e^{-\frac{z}{\bar{H}}} g_z \gamma \bar{\rho}_0 k_2 & wk_1 - \omega \end{pmatrix}, \quad \mathbf{b} = 0. \quad (\text{B10})$$

Again, with the RHS being zero, this system only has non-trivial solutions if $|A| = 0$, that is:

$$(wk_1 - \omega) e^{-2\frac{z}{\bar{H}}} \left((g_z H \gamma - w^2) k_1^2 + 2w\omega k_1 - \omega^2 + (H\gamma k_2 + i) g_z k_2 \right) \omega = 0 \quad (\text{B11})$$

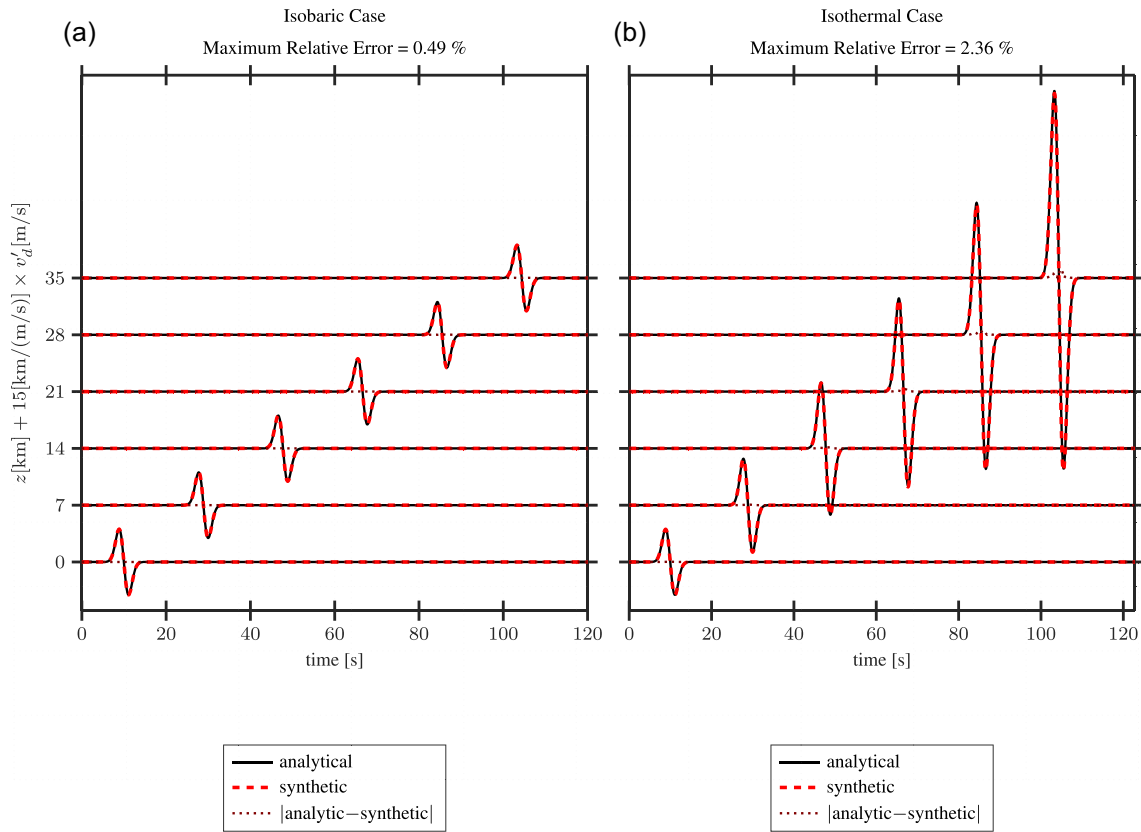


Figure B1. verification of the software using the FK method, that is using plane wave analytical solutions, as detailed in Appendix B1. Comparison between the synthetic times series of vertical velocity $[v']_d$, under the form of a time-distance plot. We test only the analytical FK mode S_3 (solid black lines) expressed either in eq. (B9) for the isobaric case or in eq. (B12) for the isothermal case. The corresponding SPECSEM2D-DG-LNS numerical solution is added as the red dashed line. The red dotted line is the absolute error between both solutions. (a) Isobaric test case. (b) Isothermal test case.

This problem has three solutions. The first two are the two solutions $S_{1,2}$ from the isobaric case (see eq. B9). The third mode is defined as:

$$S_3 := \left(\omega, k_1, \frac{-ig_z + \sqrt{-g_z (4H^2 g_z \gamma^2 k_1^2 - 4H\gamma w^2 k_1^2 + 8H\gamma \omega w k_1 - 4H\gamma \omega^2 + g_z)}}{2g_z H\gamma} \right), \quad (\text{B12})$$

where γ is the adiabatic index, H is the scale height, w is the horizontal wind, and g_z is gravity. Remark that, for an isothermal model, $g_z H\gamma = c^2$ (see Table 1). Again, because of the trivial nature of the FK solutions $S_{1,2}$, only the FK solution eq. (B12) (new S_3 mode) is tested.

B1.3 Results and discussion

Fig. B1 presents the comparisons between the analytical vertical velocity perturbation $[v']_d$ (obtained through inverse Fourier transforms), and $[v'_N]_d$ its numerical counterpart. In this Figure, no wind is considered ($w = 0$) for simplicity. The isobaric case showcasing an error lower than 1 per cent, we consider the verification excellent. The error for the isothermal case rises to 2.36 per cent at 35 km altitude. This is mainly a consequence of the exponential growth rate, which can only lead to exponentially higher errors than the isobaric case. A possible mitigation strategy would be to render the equations adimensional, which is planned as a future evolution of code. That being said, this case can reasonably be considered also verified.

Numerous other FK verification cases could be considered, for instance the ones presented by Brissaud *et al.* (2016). However, for the sake of conciseness, we limit ourselves to the two presented here, and rely on the other verification cases performed and presented alongside these.

B.2 Verification of the absorbing boundary conditions

In this Section, we test the outer boundary conditions (OBCs) as described in Section 5.1. More specifically, the far-field boundary conditions (FFBC, defined in eq. 56) are compared to the newly introduced real stretching boundary conditions (RSBC, defined in eq. 72). Following

Table B1. Description of the simulations (test cases and runs) used for the verification of the absorbing outer boundary conditions (ABCs). The test cases (line 1) and various runs (line 3) are described in full details in Appendix B2; the acronyms are: PW = Plane Wave, PS = Point Source, and WPS = Windy Point Source. The sources (line 2) are described in text (Appendix B2). The computational domain is defined as $D = D_x \times D_z$, with D_x presented in line 4 and D_z presented in line 5. The OBCs (line 6) are the choice of Outer Boundary Conditions. For the large case, since the signals never encounter the outer boundary, no OBCs are specified (‘-’). For the other cases, various Boundary Conditions (BCs) are chosen, as follows. BC1 consists in choosing the far-field BC eq. (56) (FFBC) on all four sides. For BC2, Periodic Boundary Conditions (PBCs) are chosen on the left/right sides, the bottom boundary is used for launching the plane wave, and a 10 m buffer of Real Stretching Boundary Conditions eq. (72) (RSBCs) is applied to the top boundary. For BC3, a 10 m RSBC is applied at the top boundary, and FFBCs are applied to the remaining sides. For BC4, all boundaries consist in 10 m RSBCs. The computational domain of interest (i.e. the computational domain with all buffers removed) is defined in the seventh and eighth lines as $D_i = D_{i,x} \times D_{i,z}$. Snapshots of the buffer runs for each case are presented in Fig. B2, for illustrative purposes.

| Case | PW | | PS | | | WPS | | | |
|-----------|-----------|------------|------------|-------------|------------|------------|-------------|------------|------------|
| Source | Eq. (B14) | | Eq. (B15) | | | Eq. (B15) | | | |
| Run | Large | far-field | buffer | Large | far-field | buffer | Large | far-field | buffer |
| D_x | | [-3, 3] | | [-200, 200] | | | [-120, 200] | | [-54, 100] |
| D_z | [0, 200] | [0, 60] | | [-200, 200] | [-200, 60] | | [-150, 150] | | [-80, 80] |
| OBCs | - | BC1 | BC2 | - | BC1 | BC3 | - | BC1 | BC4 |
| $D_{i,x}$ | | [-3, 3] | | [-200, 200] | | | [-44, 90] | | |
| $D_{i,z}$ | | [0, 50] | | [-200, 50] | | | [-70, 70] | | |

Arina (2016), let the stretching functions involved in eq. (72) be chosen as:

$$\forall r \in [0, 1], \quad s(r) = \kappa_1(r) = \kappa_2(r) = \frac{1}{1 - (1 - \varepsilon r)(1 - (1 - r)^p)^q}, \quad (\text{B13})$$

where $p = 3.25$, $q = 6$, and $\varepsilon = 0.25$. Supporting Information Fig. S2 illustrates eq. (B13). At the end of the buffer, elements are stretched by a factor $1/\varepsilon$. At the beginning of the buffer, elements are not stretched ($s(r) = 1$). The factors p and q govern the growth of the stretching: a high p forces the stretching to be flat at the end of the buffer, while a high q forces it to be flat at the beginning.

Three test cases are considered.

- (i) A *Plane Wave* (PW) test features an homogeneous plane wave encountering the top boundary of the computational domain at orthogonal incidence.
- (ii) A *Point Source* (PS) test features an isotropic point source, encountering the boundary at various incidence angles.
- (iii) A *Windy Point Source* (WPS) also features a point source, but in a windy medium, in order to test the OBCs for windy conditions.

For each test case, three runs are conducted.

- (i) A *large* run, which is taken as truth. This run has a computational domain large enough for the signal never to encounter the boundaries, thus mimicking an infinite domain.
- (ii) A *far-field* run, in which the computational domain is drastically reduced, and whose OBCs are only the far-field boundary conditions eq. (56).
- (iii) A *buffer* run, having real stretching boundary conditions eq. (72) on one or more sides.

All cases and corresponding runs are described in Table B1.

The physical model is chosen isobaric (Table 1) for all of them, with $\bar{\rho}_0 = 1.4 \text{ kg m}^{-3}$, $\bar{c} = 340 \text{ m s}^{-1}$, $\gamma = 1.401$, $\bar{\mu} = 1.25 \times 10^{-5} \text{ kg (m s)}^{-1}$, and $\bar{\kappa} = 0.025 \text{ m kg (s}^3 \text{ K)}^{-1}$. The static wind \mathbf{v}_0 is set to zero for all cases except WPS, in which it is set to $(10, 0) \text{ m s}^{-1}$. The sources used are as follows. In order to generate a plane wave in the PW case, the vertical velocity is strongly forced at the bottom boundary by a Gaussian derivative pulse:

$$\forall t, \quad \forall \mathbf{x}, \quad [\mathbf{v}']_d(t, \mathbf{x}, z = 0) = \partial_t G_{\frac{1}{1.4 \times 10^{-2}}}(t - 3 \times 10^{-2}). \quad (\text{B14})$$

A point source in the PS and WPS cases simply choosing $F_{\rho'} = F_{\rho_0} \mathbf{v}' = 0$ and:

$$\forall t, \quad \forall \mathbf{x}, \quad F_{E'}(t, \mathbf{x}) = G_{75}(t) \exp\left(-\frac{\|\mathbf{x}\|_2^2}{0.6^2}\right) \quad (\text{B15})$$

in eq. (4). In eq. (B14) and eq. (B15), G_f is the Gaussian function with fundamental frequency f (in Hz) as defined in eq. (B6).

The metric used for comparisons is based on 1) total energy and 2) selected time-series. On the one hand, the potential and kinetic energies are integrated only over the domain of interest (smallest subdomain for each test case, as described in Table B1). From the definition of kinetic and potential energies in a fluid, the respective energy perturbations write as eq. (21) and eq. (22).

On the other hand, simulated receivers record pressure perturbation time-series at selected points of interest. The time-series are then compared between runs. An absolute L^2 error is computed between the large run and the absorbing boundary condition test runs (either FFBC

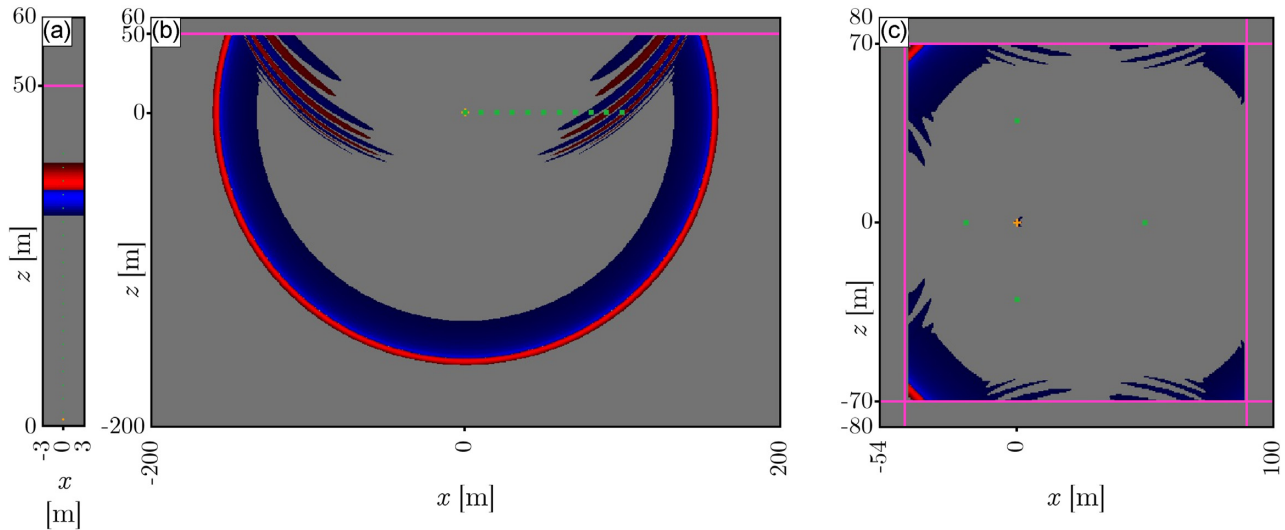


Figure B2. Illustrative snapshots of the simulations used for the verification of the Absorbing Boundary Conditions (ABCs). The buffer run is depicted for each test cases: PW (panel a), PS (panel b) and WPS (panel c). Each run and test case is described in Table B1. Snapshots represent pressure perturbations in the atmosphere. Red indicates values $>+1$ per cent of maximum amplitude, while blue indicates values <-1 per cent. The green dots represent receivers. The yellow cross pinpoints the source. Pink lines demarcate the beginning of the buffers where the Real Stretching Boundary Conditions (RSBCs, see eq. 72) are applied.

or RSBC), as

$$\|p'_{\text{large}} - p'_{\text{test}}\|_{L^2([0, t_{\text{max}}])}^2 = \int_0^{t_{\text{max}}} |p'_{\text{large}} - p'_{\text{test}}|^2 dt, \quad (\text{B16})$$

where p'_{large} are the pressure perturbation time-series recorded in the large case, p'_{test} are the pressure perturbation time-series recorded either in the FFBC test or the RSBC test, and t_{max} is simply some maximum simulation time.

Illustrative snapshots of the buffer runs for each test case are presented in Fig. B2. The full results are summarized in Fig. B3.

In the PW test case (Table B1), the far-field condition is excellent as no significant spurious reflected wave can be seen coming back into the domain of interest (panels a and d of Fig. B3). The real stretching condition shows good performance, where spurious reflected waves have an amplitude less than 5×10^{-4} per cent of the incident wave. The energy levels during those reflections ($t > 0.2$ s) remain negligible compared to during the active time interval ($t < 0.2$ s). Both boundary conditions fully absorb head-on plane waves (PW case, that is a 90° incident waves case). However, such waves remain a very specific case, quite improbable when considering real-life configurations.

When considering non-zero incidences (PS and WPS cases, see Table B1), the real stretching boundary conditions (RSBCs, eq. 72) outperforms the far-field conditions (FFBCs, eq. 56) both in terms of energy and time-series. In the PS case, spurious reflections coming from the far-field boundaries have systematically a higher amplitude than those coming from the real stretching ones (panels b and e of Fig. B3). In the WPS case, real stretching reflections' amplitudes remain under 1 per cent of the incident wave amplitude, while far-field reflections rise to a few percent (panels c, f and h of Fig. B3). The L^2 error between time-series in the large case and each OBC method (defined in eq. (B16) and plotted in panel g) of Fig. B3) shows the RSBCs eq. (72) always perform better than the FFBCs.

Using the RSBC inevitably implies additional computations, since the buffers elements have to be resolved in addition to the inner domain of interest. However, the size of buffers is usually much smaller than the computational domain of interest. Over the three cases, all ran over 16 CPUs, the RSBCs did not induce any additional cost when compared with the FFBCs. In the PW case, the *buffer* run is only 6 per cent (2 seconds) longer than the *far-field* case. In the PS and WPS cases, the *buffer* run lasted as long as the *far-field* case. For all cases, the use of either the FFBC or RSBC allowed to speed up simulations by a factor ranging between 1.5 and 3.7, compared to the *large* run.

We therefore consider the newly introduced real stretching method eq. (72) as a better alternative for outer boundary conditions for the fluid equations. Better (lower) reflection coefficients at the outer boundary of the computational domain require thoroughly exploring the parameters of the stretching functions. Such a parametric study needs to be performed for completeness.

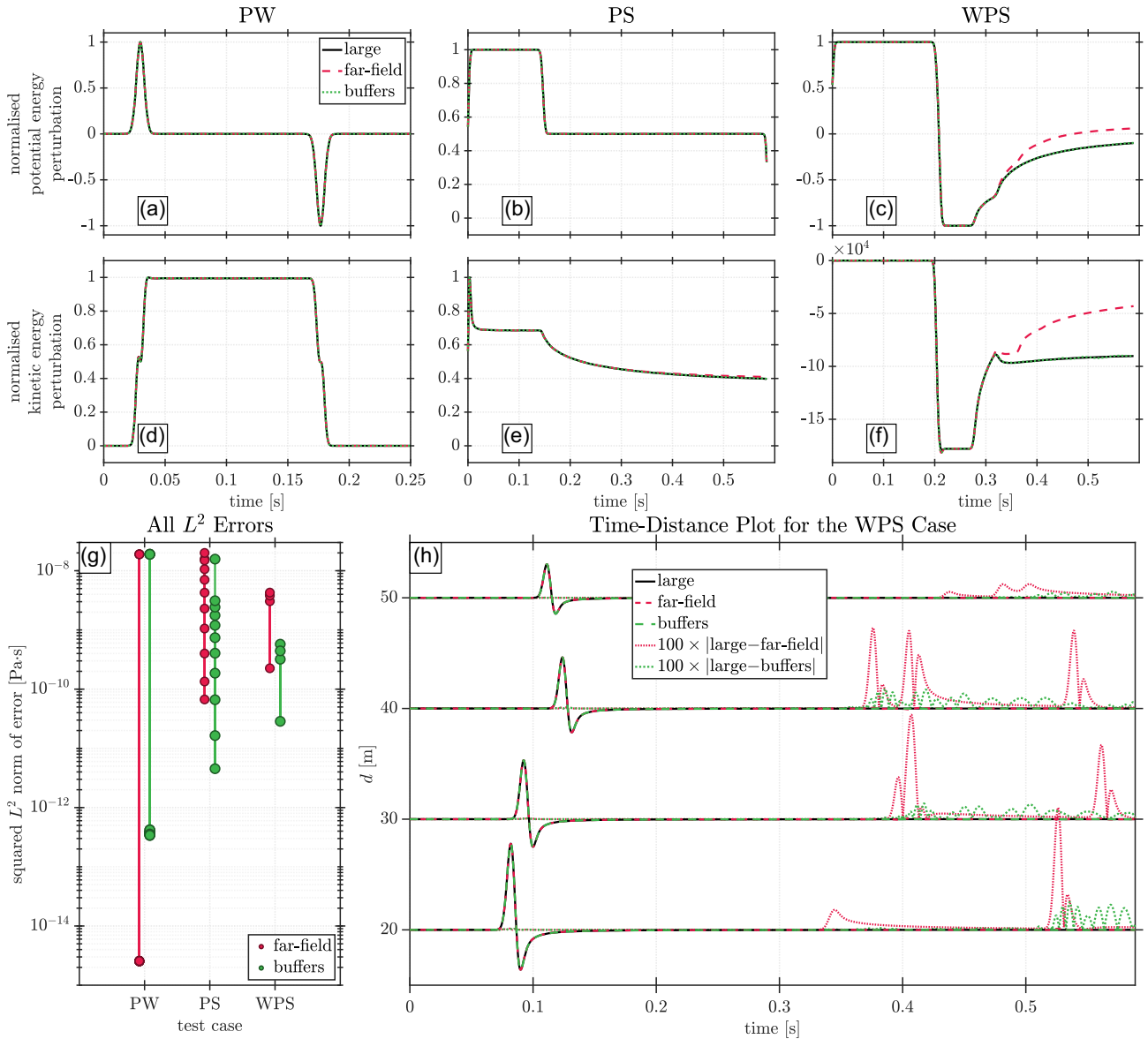


Figure B3. Comparison between outer boundary conditions (OBCs) methods. (a–f) Comparison in terms of potential and kinetic energies, integrated over the domain of interest (see Table B1), for each test case. (a–c) Normalized potential energy perturbation (see eq. 22). (d–f) Normalized kinetic energy perturbation (see eq. 21). The energy perturbation is computed by removing from the total energies the initial energies. The normalization is relative to the maximum energy perturbation. (g and h) Comparison in terms of pressure time-series. (g) L^2 norm (eq. B16) of the difference between the large run and each BC method, for each test case. The circles are the L^2 norms computed at various simulated receivers, distributed in the domain of interest (see Fig. B2). For each test case, the location of the receivers is identical between the three runs. The colours correspond to either the far-field case (red) or the buffer case (green). (h) Time-series comparison for the windy point source (WPS) case, for all four simulated receivers (see Fig. B2), in terms of distance to the source d . The large test case (mimicking an infinite domain) is represented by the solid black lines. The far-field test is the red dashed lines, and the buffer test is the green dashed lines. An absolute error between each test case and the large run is computed, and added as the dotted lines; we display it in red for the far-field test case and in green for the buffer test case, and scaled them up for readability.