



**HAL**  
open science

# 3D Frequency-Domain Elastic Wave Modeling Using Spectral Element Method with a Parallel Direct Linear Solver

Yang Li, Romain Brossier, Ludovic Métivier

► **To cite this version:**

Yang Li, Romain Brossier, Ludovic Métivier. 3D Frequency-Domain Elastic Wave Modeling Using Spectral Element Method with a Parallel Direct Linear Solver. 81st EAGE Conference and Exhibition 2019, Jun 2019, London, United Kingdom. 10.3997/2214-4609.201901534 . hal-02325645

**HAL Id: hal-02325645**

**<https://hal.science/hal-02325645>**

Submitted on 24 Nov 2020

**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.

# 3D frequency-domain elastic wave modeling using spectral element method with a parallel direct linear solver

Y. Li<sup>1\*</sup>, R. Brossier<sup>1</sup>, L. Métivier<sup>1,2</sup>

<sup>1</sup> Univ. Grenoble Alpes, ISTerre, F-38058 Grenoble, France

<sup>2</sup> CNRS, Univ. Grenoble Alpes, LJK, F-38058 Grenoble, France

March 18, 2019

Main objectives Apply the spectral element method to solve the 3D frequency-domain anisotropic elastic wave modeling and solve the linear system with a parallel direct solver. Analyze the performance of the direct solver, the factorization time and memory cost. Present the current limit of the modeling scale and illustrate the interest on future adoption of low-rank compression based techniques.

New aspects covered High order spectral element method for 3D frequency-domain elastic wave modeling with complex topography and free surface boundary condition. A feasibility study of a parallel direct solver on the linear system generated from discretization.

## Summary

Complex topography, free surface boundary condition and inelastic properties of media should be well considered for onshore geophysical prospecting. Thus an appropriate and accurate forward modeling engine is very important. Unlike the time-domain implementation of many seismic imaging techniques, the counterpart in the frequency domain is rarely studied, despite of having many advantages, for example, only limited number of frequencies is needed for the inversion process, and solving the multiple-source problem is quite cheap if a direct solver is used. In this study, the spectral element method is applied to discretize the 3D frequency-domain anisotropic elastic wave modeling and the parallel direct solver MUMPS is used to solve the linear system. The structure and building process of the impedance matrix is thoroughly explained. We validate the numerical results by comparing with analytical solutions. A hybrid implementation of MPI and OpenMP for MUMPS is shown to be more efficient in flops and memory cost during the factorization. The influence of complex topography on MUMPS performance is negligible. With the available resources, the largest scale modeling, 30 wavelength in each dimension, is achieved. Other direct solvers and different low-rank techniques will also be investigated to reduce the flops and memory cost.

## Introduction

Onshore seismic imaging has drawn greater attention recently in the framework of geophysical prospecting techniques. For such challenging problem, acoustic wave modeling can no longer be used because of the elastic, visco-elastic or anisotropic effects in the onshore environment. It is thus necessary to perform the anisotropic visco-elastic wave modeling. The majority of current seismic imaging applications relies on the time-domain forward modeling. It is straightforward to decompose the physical domain and parallelize over the sources. The memory cost is acceptable in terms of the scale of model size. In contrast, study on frequency-domain modeling in the framework of onshore seismic imaging is relatively rare. A successful application in offshore environment is given in Operto and Miniussi (2018), which prompts us to investigate the 3D frequency-domain anisotropic elastic wave modeling to prepare for further imaging. One popular method in 3D elastic case is the finite-difference method (FDM) due to its simplicity and high efficiency. However, it is required to refine the grid or use specific coordinate transform techniques when dealing with the complex topography and free surface boundary condition (FSBC). This would largely counteract its efficiency. The finite element method (FEM) seems a more appropriate choice because FSBC is naturally satisfied in the weak form of wave equation and an adaptive design of the mesh could further simplify the realization of complex topography and acknowledge the variations in the media. The spectral element method (SEM), as a specific FEM, has been investigated particularly in seismic imaging and seismology (Komatitsch et al., 2000; Trinh et al., 2019). The specific character of SEM consists in using a mesh of hexahedra in 3D and choosing the Gauss-Lobato-Legendre (GLL) points for the integration and Lagrange interpolation. Using high-order Lagrange polynomials and Gauss quadrature on GLL points enables spectral convergence when solving smooth problems. Although SEM, as a continuous FEM, could not handle the discontinuities in the media and solutions, it fits well the applications for seismic exploration where we can presume the continuity of the solution and media. Note that anisotropy can be considered without making an extra effort like in the FDM case. The viscosity can also be easily incorporated by using complex-valued elastic modulus in the frequency domain.

In this study, SEM is applied to 3D frequency-domain elastic wave modeling, taking into account the heterogeneous anisotropic media, complex topography and FSBC. A general mesh is considered at the very beginning. Simpler mesh, such as a vertically-deformed mesh or an undeformed Cartesian mesh, could be realized simply by modifying the Jacobian matrix. MUMPS 5.1.2 is used to solve the large sparse linear system obtained from the discretization. Several numerical experiments are conducted to validate the accuracy of the numerical results and to assess the performance of MUMPS in terms of time and memory complexity.

### Spectral element method for 3D frequency-domain elastic wave modeling

The 3D frequency-domain elastic wave equation reads

$$\rho \omega^2 u_j + \frac{\partial}{\partial x_i} \left( c_{ijkl} \frac{\partial u_k}{\partial x_l} \right) + f_j(\omega, \mathbf{r}_s) = 0, \quad i, j, k, l = 1, 2, 3, \quad (1)$$

where  $\rho$  is the density,  $\omega$  is the angular frequency,  $u_j$  is the displacement vector,  $c_{ijkl}$  is the elastic modulus tensor and  $f_j(\omega, \mathbf{r}_s)$  is the point source force vector located at  $\mathbf{r}_s$ . Einstein convention is used here. The seismic attenuation in viscoelastic media could be incorporated easily by using complex-valued elastic moduli. The weak form of equation (1) is obtained by multiplying a test function  $\phi$  and integrating over the physical volume  $\Omega$ . Using the integration by parts and incorporating FSBC and absorbing boundary conditions, the weak form rewrites as

$$\omega^2 \int_{\Omega} \rho u_j \phi \, d\mathbf{x} + \int_{\Omega} c_{ijkl} \frac{\partial u_k}{\partial x_l} \frac{\partial \phi}{\partial x_i} \, d\mathbf{x} + \int_{\Omega} f_j(\omega, \mathbf{r}_s) \phi \, d\mathbf{x} = 0. \quad (2)$$

The volume  $\Omega$  is then divided into a set of non-overlapping hexahedral elements. A mapping is defined to transform a unitary cube  $[-1, 1] \otimes [-1, 1] \otimes [-1, 1]$  into a single element  $\Omega^e$ . The unitary cube is discretized by GLL points. With respect to these GLL points, a scalar function could be approximated by corresponding Lagrange polynomials and an integral could be discretized by Gauss quadrature. Taking the basis functions as the test function and incorporating the interpolation and GLL quadrature, we obtain the discretized linear system of equation (2) as follows

$$\mathbf{A}\mathbf{u} = \mathbf{f}, \quad (3)$$

where  $\mathbf{A} = \omega^2 \mathbf{M} + \mathbf{K}$  is the impedance matrix,  $\mathbf{M}$  is the mass matrix and  $\mathbf{K}$  is the stiffness matrix. Vector  $\mathbf{u}$  is the discretized displacement and  $\mathbf{f}$  represents the discretized source vector. Note that the combination of using the Gauss quadrature and the Lagrange interpolation at GLL points leads to a diagonal mass matrix  $\mathbf{M}$  for the discretization of the wave equation. This is a huge advantage for the time-domain wave modeling if explicit time marching scheme is used, no matrix inversion is required.

As for the boundary conditions, we adopt the anisotropic PML method (Shi et al., 2016) on lateral and bottom sides of the 3D model. With appropriate arrangement, the complex coordinate stretching tech-

nique leads to complex-valued elastic parameters and density, while keeping the original wave equation unchanged. The new parameters are defined as follows

$$\tilde{\rho} = \rho s_{x_1} s_{x_2} s_{x_3}, \quad \tilde{c}_{ijkl} = c_{ijkl} \frac{s_{x_1} s_{x_2} s_{x_3}}{s_{x_i} s_{x_k}}, \quad (4)$$

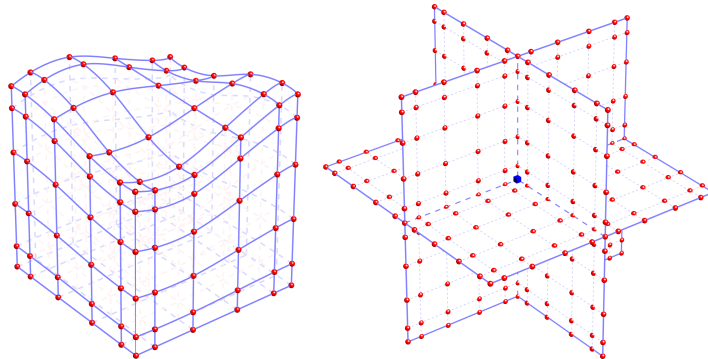
where  $s_*$  corresponds to the complex coordinate stretching in each dimension.

Since the mass matrix  $M$  is diagonal and easy to construct, the main part of building  $A$  lies in the stiffness matrix  $K$ . We take as an example the discretization of only one term in the expansion of equation (2) to illustrate the connection between the grid points, i.e., the structure of  $K$ . Denoting  $u_j$  as  $u$  and  $c_{ijkl}$  as  $c$  and taking test function as a product of Lagrange polynomials in each dimension  $\phi(\mathbf{x}) = \ell_p(x)\ell_q(y)\ell_r(z)$ , we have

$$\begin{aligned} \int_{\Omega^e} c \partial_1 u \partial_1 \phi \, dx \approx & \sum_{\bar{i}} w_{\bar{i}} w_q w_r c_{\bar{i}qr} \ell'_p(\xi_{\bar{i}}) \partial_1 \xi \left[ \sum_i u_{iqr} \ell'_i(\xi_{\bar{i}}) \partial_1 \xi + \sum_j u_{\bar{i}jr} \ell'_j(\eta_q) \partial_1 \eta + \sum_k u_{\bar{i}qk} \ell'_k(\zeta_r) \partial_1 \zeta \right] \mathcal{J}_{\bar{i}qr} \\ & + \sum_{\bar{j}} w_p w_{\bar{j}} w_r c_{p\bar{j}r} \ell'_q(\eta_{\bar{j}}) \partial_1 \eta \left[ \sum_i u_{i\bar{j}r} \ell'_i(\xi_p) \partial_1 \xi + \sum_j u_{pjr} \ell'_j(\eta_{\bar{j}}) \partial_1 \eta + \sum_k u_{p\bar{j}k} \ell'_k(\zeta_r) \partial_1 \zeta \right] \mathcal{J}_{p\bar{j}r} \\ & + \sum_{\bar{k}} w_p w_q w_{\bar{k}} c_{pq\bar{k}} \ell'_r(\zeta_{\bar{k}}) \partial_1 \zeta \left[ \sum_i u_{i\bar{k}} \ell'_i(\xi_p) \partial_1 \xi + \sum_j u_{p\bar{j}k} \ell'_j(\eta_q) \partial_1 \eta + \sum_k u_{pqk} \ell'_k(\zeta_{\bar{k}}) \partial_1 \zeta \right] \mathcal{J}_{p\bar{k}}. \end{aligned} \quad (5)$$

where  $w_*$  are the GLL quadrature weights and  $\mathcal{J}$  is the Jacobian matrix. The red terms in the sum indicate that one point in an element is connected to those points on three ‘planes’, which intersect at this point, i.e.,  $u_{pqr}$  is connected to  $u_{ijr}$  ( $i, j = 1, \dots, n$ ),  $u_{iqk}$  ( $i, k = 1, \dots, n$ ) and  $u_{pj k}$  ( $j, k = 1, \dots, n$ ), where  $n$  is the number of GLL points in each dimension in one element. The discretization of other terms in equation (2) has similar formula. Thus from a global point of view, the point with the largest number of connected points is a vertex of elements and is shared by 8 surrounding elements. In this case, the point is connected to 12 ‘elemental planes’. Figure 1 (a) shows the distribution of points in a 5th-order SEM element and in (b) a detailed illustration of the connection mode. The blue cube at the center is an example of the mentioned points. The points on these three planes, i.e., the 12 ‘elemental planes’, are connected to this point. For a SEM of polynomial order  $n$ , the number of points connected (number of nonzeros in each matrix row) is  $36n^2 + 18n + 3$ , i.e., 993 for a 5th-order SEM. The matrix will thus be denser than that of standard FDM. However, the higher order accuracy of SEM will decrease the total number of grid-points per wavelength compared with conventional FDM. Moreover, implementing FSBC would also require a FDM grid fine enough to simulate the surface waves, while for SEM no extra treatment needs to be taken. Both of these factors largely reduce the size of the SEM mesh and consequently generate an impedance matrix much smaller than that from FDM.

The way we build  $K$  is the same as calculating  $K\mathbf{u}$  in the time-domain modeling, apart from altering  $\mathbf{u}$  into a canonical basis vector  $\mathbf{e}^i = (0, \dots, 0, 1, 0, \dots, 0)$  for the  $i$ -th row. Hence the matrix is built row by row. Note that the calculation involves only the points inside one element. Therefore, building one row of  $K$  is an elemental behavior, which is very efficient. We adopt the highly efficient algorithm proposed by Deville et al. (2002) and implemented in Trinh et al. (2019), which recasts the matrix-vector product as a tensor-product evaluation, benefits from the tensorial properties of hexahedral elements, the optimization of cache usage, and the combination of efficient loop vectorization and manual unrolling. Numerical experiments indicate that performing the tensor-product evaluation is at least ten times faster than the direct matrix-vector product.



**Figure 1** Point distribution in one element (left) and connection mode of points (right) for a 5th-order SEM.

### Validation of numerical solutions

We validate the accuracy of the numerical results by comparison with analytical solutions. The parameter settings for modeling are summarized in Table 1. A 5th-order SEM is used in the following experiments. Thus setting one element, i.e., 5 points per wavelength should guarantee the accuracy. The real parts of 3D elastic wavefields  $u_y$  are shown in Figure 2. The difference between the analytical and numerical solutions are negligible. Same conclusion applies to  $u_x$  and  $u_z$ .

#e/dim	#e in PML	e  (m)	$V_p$ (m/s)	$V_s$ (m/s)	$\rho$ (g/cm <sup>3</sup> )	f (Hz)	DOF/wavelength $\lambda$
20	2	100	5000	2500	1	25	5

Table 1 Parameter settings

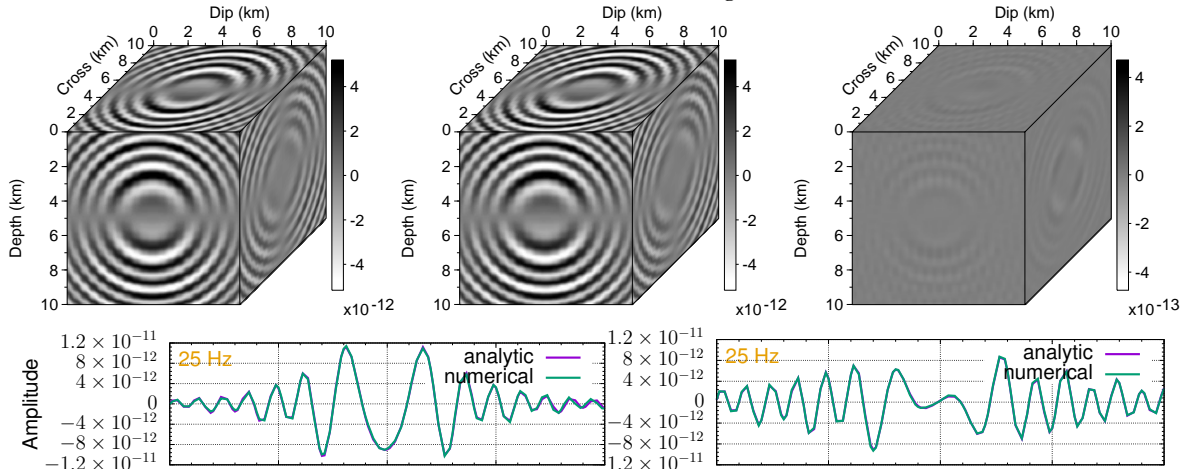


Figure 2 First row: Analytical (left), numerical (middle) solutions and the difference (right) for wavefields  $u_y$ . Second row: slices from the analytical and numerical wavefields

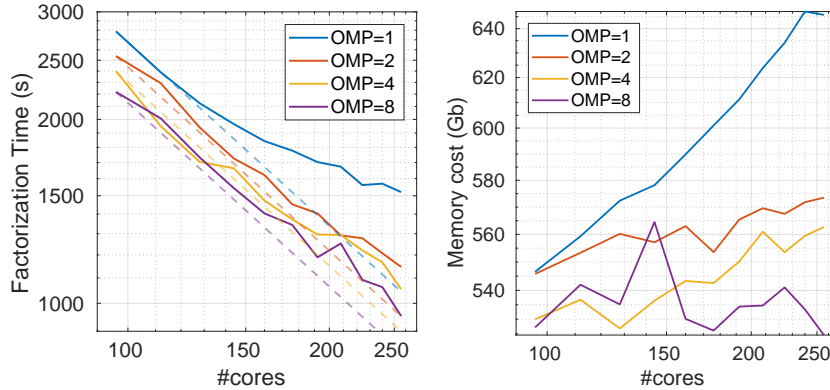
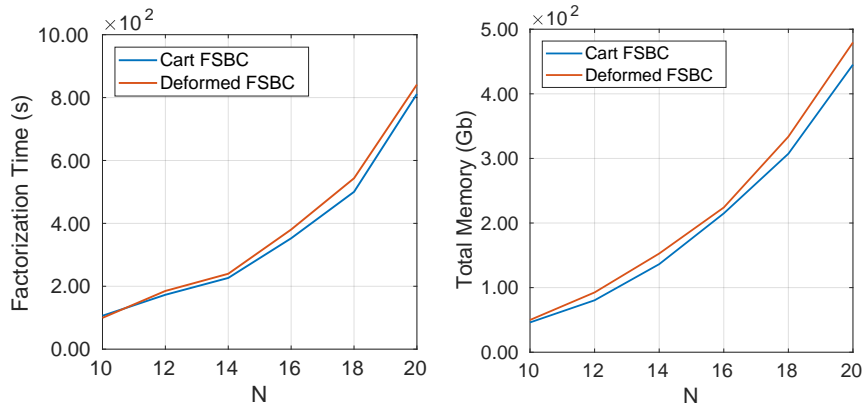


Figure 3 Factorization time and memory cost of MUMPS with different number of OpenMP threads for fixed number of total cores (MPI + OpenMP)

### Analysis of MUMPS performance

MUMPS 5.1.2 (full rank version) is used to solve the linear system (3). It is based on a multifrontal method (MUMPS-team, 2017), which recasts the original matrix into multiple frontal matrices and computes the LU decomposition of these smaller matrices to save memory and computational cost. We use MUMPS with a hybrid implementation of MPI and OpenMP, which fully benefits from the high-level BLAS (Basic Linear Algebra Subprograms) and allows a better usage of memory hierarchy and better performance thanks to the blocking techniques used in BLAS routines. The numerical settings are the same as in Table 1. The total number of MPI and OpenMP is increased from 96 to 256 and the number of OpenMP threads varies from 1 to 8 in order to fit our hardware settings (2 Intel E5-2670 processors per node, 8 cores per processor). Figure 3 presents the corresponding factorization time and memory cost of MUMPS with different number of OpenMP threads. The dashed lines indicate the ideal scalability and the solid curves are real computing time. With a fixed total number of MPI and OpenMP, the more OpenMP threads we use, the better MUMPS scales. Fewer MPI leads to larger blocks of matrix for BLAS to deal with where BLAS could fully adopt the OpenMP parallelization. As for the memory cost, using multithreaded BLAS allows a better usage of memory, which is also illustrated in Figure 3. Although the memory usage of 8 threads is larger for smaller number of total MPI + OpenMP, the trend agrees well with the expectation when the number of total MPI + OpenMP increases.

To investigate the growth trend of MUMPS memory cost and flops, we increase the model size from  $10 \times 10 \times 10$  to  $20 \times 20 \times 20$  elements. The parameters settings are the same as in Table 1 except for #e/dim. The number of cores is 64, 96, 160, 192, 224, 256 respectively with #OMP = 8. Free surface boundary condition is taken into account. Two sets of experiments, one with Cartesian non-deformed mesh and the other with vertically deformed mesh, are conducted to test the influence of complex topography. The factorization time and memory cost are presented in Figure 4. It is promising to see that the deformed mesh does not introduce great increment of the factorization time and memory cost because the matrix structure in each case is very similar. A direct solver is thus not affected by this modification. Conversely, the surface waves generated from the complex topography and FSBC may lead to drastic convergence delay for iterative solvers (Li et al., 2015).



**Figure 4** Factorization time (left) and total memory (right) of wave modeling for different model size, with Cartesian and deformed mesh.

Table 2 summarizes the largest modeling scale we have achieved so far (30 and 28 wavelengths respectively in each dimension) both with Cartesian and vertically deformed mesh. The size of the linear system reaches tens of millions and the number of nonzeros in the matrix surpasses  $10^9$ . As seen, these results could be obtained using moderate scale of computing resources. Time for solving each RHS is trivial, which is very appealing for seismic imaging applications with thousands of sources. The current bottleneck comes from the matrix ordering before the factorization. We use the sequential METIS (Karypis, 2013) which is quite efficient for the subsequent factorization in terms of both time and memory. However, the memory cost of METIS reaches its limit as the model size increases to about  $30 \times 30 \times 30$  elements. The parallel version of METIS (ParMETIS) and other ordering algorithms have been tested as well. But the flops and memory cost of factorization with these ordering algorithms exceeds the capacity of our current computing environment.

	# $\lambda$	DOF	$N_{\text{NNZ}}$	#core	Memory <sub>total</sub> (GB)	Time <sub>Factorization</sub> (s)	Time <sub>Solve</sub> (s)
Cartesian	30	1.4E7	2.1E9	384	1913.6	2085.9	5.1
Deformed	28	1.2E7	5.4E9	320	1434.6	1828.2	4.6

**Table 2** Current largest modeling with MUMPS (#OMP=16)

## Conclusions and perspectives

We have applied SEM to perform the 3D frequency-domain anisotropic elastic wave modeling and used MUMPS to solve the generated linear system. Complex topography and FSBC are well handled by deformed mesh. The accuracy of numerical results are validated by comparing with the analytical solutions. MUMPS with a hybrid implementation of MPI and OpenMP presents a satisfactory performance in terms of scalability, flops and memory cost. Using deformed mesh only introduce slight extra flops and memory cost compared with using Cartesian mesh. It is very promising for future onshore applications where complex topography has to be considered. With limited computing resources, a moderate scale modeling (30 wavelengths in each dimension) is achieved. Note that the experiments are conducted with a full rank MUMPS. Using the block low-rank version of MUMPS as well as other direct solvers and other low-rank techniques will be investigated in the future. This could reduce the flops and memory cost during the factorization. Other parallel ordering methods should also be tested to avoid current bottleneck. Ultimately, incorporating the forward modeling into the framework of FWI might be considered.

## Acknowledgements

This study was partially funded by the SEISCOPE consortium (<http://seiscope2.osug.fr>), sponsored by AKERBP, CGG, CHEVRON, EQUINOR, EXXON-MOBIL, JGI, PETROBRAS, SCHLUMBERGER, SHELL, SINOPEC and TOTAL. This study was granted access to the HPC resources of CIMENT infrastructure (<https://ciment.ujf-grenoble.fr>) and CINES / IDRIS / TGCC under the allocation 046091 made by GENCI.

## References

- Deville, M., Fischer, P. and Mund, E. [2002] *High Order Methods for Incompressible Fluid Flow*. Cambridge University Press.
- Karypis, G. [2013] *METIS - A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-Reducing Orderings of Sparse Matrices - Version 5.1.0*. University of Minnesota.
- Komatitsch, D., Barnes, C. and Tromp, J. [2000] Simulation of anisotropic wave propagation based upon a spectral element method. *Geophysics*, **65**(4), 1251–1260.
- Li, Y., Métivier, L., Brossier, R., Han, B. and Virieux, J. [2015] 2D and 3D frequency-domain elastic wave modeling in complex media with a parallel iterative solver. *Geophysics*, **80**(3), T101–T118.
- MUMPS-team [2017] *MUMPS - MULTifrontal Massively Parallel Solver users' guide - version 5.1.2 (October, 2017)*. ENSEEIHT-ENS Lyon, <http://www.enseeiht.fr/apo/MUMPS/> or <http://graal.ens-lyon.fr/MUMPS>.
- Operto, S. and Miniussi, A. [2018] On the role of density and attenuation in 3D multi-parameter visco-acoustic VTI frequency-domain FWI: an OBC case study from the North Sea. *Geophysical Journal International*, **213**, 2037–2059.
- Shi, L., Zhou, Y., Wang, J.M., Zhuang, M., Liu, N. and Liu, Q.H. [2016] Spectral element method for elastic and acoustic waves in frequency domain. *Journal of Computational Physics*, **327**(Supplement C), 19 – 38.
- Trinh, P.T., Brossier, R., Métivier, L., Tvard, L. and Virieux, J. [2019] Efficient 3D time-domain elastic and viscoelastic Full Waveform Inversion using a spectral-element method on flexible Cartesian-based mesh. *Geophysics*, **84**(1), R75–R97.