# Full waveform inversion and triangular waveform adapted meshes using Firedrake

A. Olender[1], K. Roberts[1], L. Franceschini[1], T. D. dos Santos[1], R. C. Kirby[2], B. S. Carmo[1]

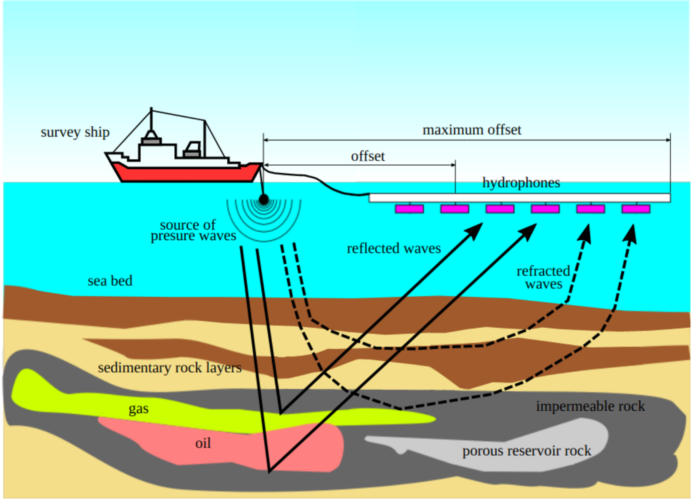
[1]Escola Politécnica, University of São Paulo
[2]Dept. of Mathematics, Baylor University

September 2021

# Table of contents

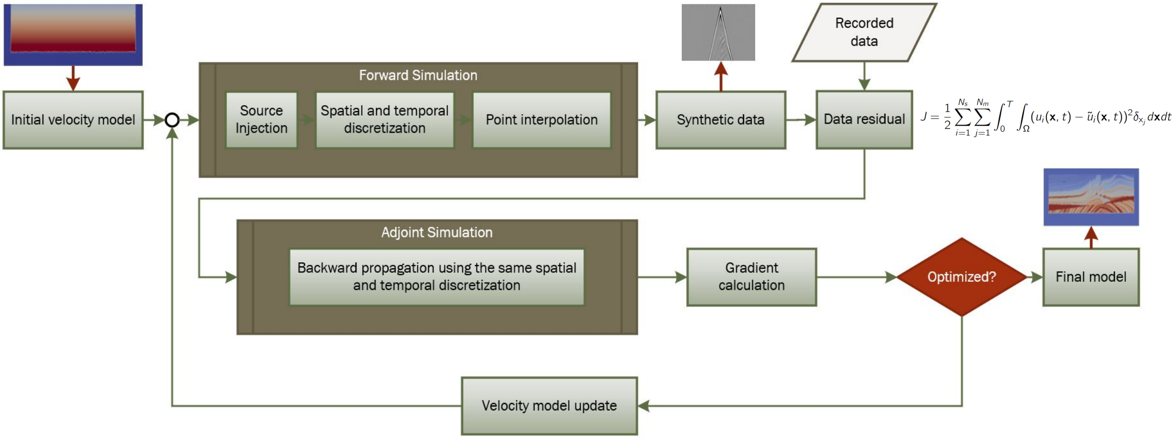
# Table of contents

# Full waveform inversion




A typical marine seismic survey

# Full waveform inversion




Iterative scheme used to run a full waveform inversion with Spyro/Firedrake

# Acoustic equation with PML

$$\frac{\partial^2 u}{\partial t^2} - \nabla \cdot (c^2 \nabla u) - \nabla \cdot \mathbf{p} + \Psi_1 \frac{\partial u}{\partial t} + \Psi_3 u + \det \Psi_1 \omega = f, \quad \Rightarrow \text{Wave eq. for pressure } u(\mathbf{x}, t)$$

$$\frac{\partial \mathbf{p}}{\partial t} + \Psi_1 \mathbf{p} + \Psi_2 (c^2 \nabla u) - \Psi_3 (c^2 \nabla \omega) = \mathbf{0}, \quad \Rightarrow \text{PML eqs. for aux. var. } \mathbf{p}(\mathbf{x}, t)$$

$$\frac{\partial \omega}{\partial t} = u, \quad \Rightarrow \text{PML eq. for aux. var. } \omega(\mathbf{x}, t)$$

**Time scheme**: Explicit FD $\Rightarrow$ only Mass-Matrices to invert:

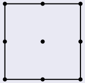
$$\frac{\partial^2 u}{\partial t^2} \Rightarrow M_u, \quad \Psi_1 \frac{\partial u}{\partial t} \Rightarrow M_{u_1}, \quad \frac{\partial \mathbf{p}}{\partial t} \Rightarrow M_p \quad \frac{\partial \omega}{\partial t} \Rightarrow M_\omega$$

see Kaltenbacher et al, JCP (2013) or Grote and Sim, preprint (2010) for PML

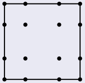
# Table of contents

# 2D Elements

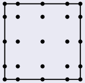
## Spectral quadrilateral finite elements with GLL nodes



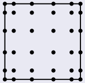
(a) $P = 2$



(b) $P = 3$



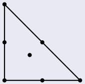
(c) $P = 4$



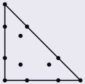
(c) $P = 5$

(Karniadakis and Sherwin, 2013; Patera, 1984)

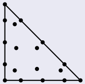
## Mass-lumped simplicial finite elements (KMV)



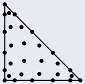
(a) $KMV2tri$



(b) $KMV3tri$



(c) $KMV4tri$

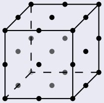


(c) $KMV5tri$

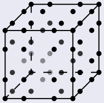
(Kong, Mulder, and Veldhuizen, 1999)

# 3D Elements

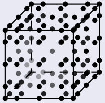
## Spectral hexahedral finite elements with GLL nodes



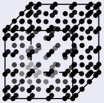
(a) $P = 2$



(b) $P = 3$



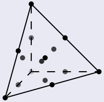
(c) $P = 4$



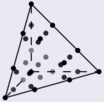
(c) $P = 5$

(Karniadakis and Sherwin, 2013; Patera, 1984)

## Mass-lumped simplicial finite elements (KMV)



(a) *KMV2tet*



(b) *KMV3tet*

(Kong, Mulder, and Veldhuizen, 1999; Geevers, Mulder, and Vegt, 2018)

# Element usage in Firedrake

## We need to get the quadrature rule from FInAT

```
from firedrake import *
import finat
kmv_rule = finat.quadrature.make_quadrature(
        V.finat_element.cell, V.ufl_element().degree(), "KMV"
)
```

## Then, we use it for the mass matrix assembly

```
V = FunctionSpace(mesh, "KMV", 3)
u = TrialFunction(V)
v = TestFunction(V)

m1 = ((u - 2.0 * u_n + u_nm1) / Constant(dt ** 2)) * v * dx(kmv_rule)
```

# Table of contents

# spyro: Acoustic wave modeling

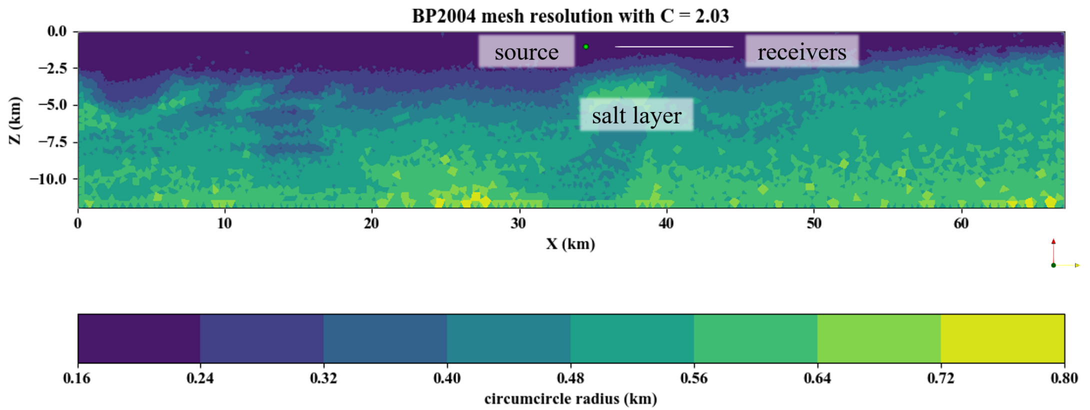
## Features

- Based on Firedrake (Rathberger et al. 2017)
  - Mass-lumped triangular and tetrahedral elements (KMV)
  - Spectral quadrilateral and hexahedral elements
- Perfectly Matched Layer (PML) to absorb reflected waves
- Capable of doing the complete FWI loop
  - Mesh-independent functional gradient using the optimize-then-discretize approach
  - Sparse interpolation and injection with point and force sources
- Automatic mesh generation with SeismicMesh (Roberts, et al. 2021a)

## Link

https://github.com/krober10nd/Spyro

# spyro works with heterogeneous wave velocity




Example of mesh generated with SeismicMesh

# Mesh design: cells per wavelength

Experiment setup based on:

- Ricker source with peak frequency of 5 Hz;
- Homogeneous velocity model of constant $c = 1,429$ km/s;
- Square grid of 36 point receivers that don't coincide with mesh nodes;
- Results compared with a reference solution in a fine high- order mesh;
- Based on experiments by Lyu et at. (2020).

In order to calculate:

- $C$: cells per wavelength necessary to inhibit numerical error
- $G$: degrees-of-freedom per wavelength necessary to inhibit numerical error.

$$E = \sqrt{\frac{\sum_{r=1}^{N_r} \int_0^t {}_f (p_r - p_{r_ref})^2 dt}{\sum_{r=1}^{N_r} \int_0^t {}_f p_{r_ref}^2 dt}} \quad (1)$$

# Mesh design: cells per wavelength

| Element | Homogeneous minimum $G$ | Homogeneous minimum $C$ |
|---|---|---|
| $KMV1tri$ | DNF | DNF |
| $KMV2tri$ | 10.1 | 5.85 |
| $KMV3tri$ | 7.86 | 3.08 |
| $KMV4tri$ | 7.36 | 2.22 |
| $KMV5tri$ | 7.88 | 1.69 |

Table: Results from Roberts et al. (2021b), detailing the necessary cells per wavelength to use in waveform adapted meshes. For heterogeneous velocity models we propose an increase of 20%.

# Table of contents

# Strong scaling on the forward problem

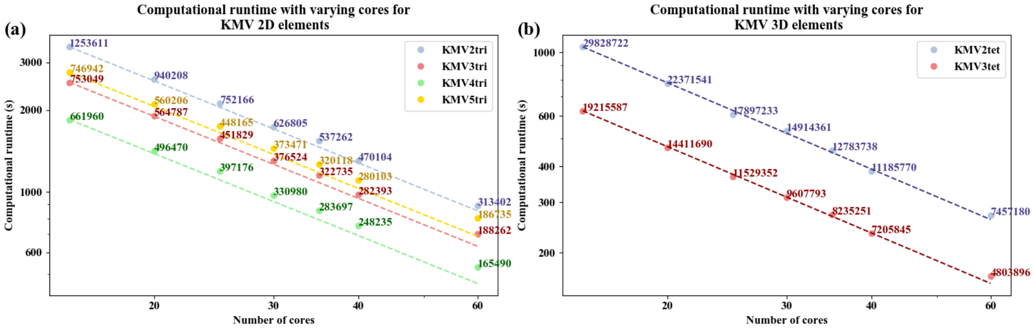
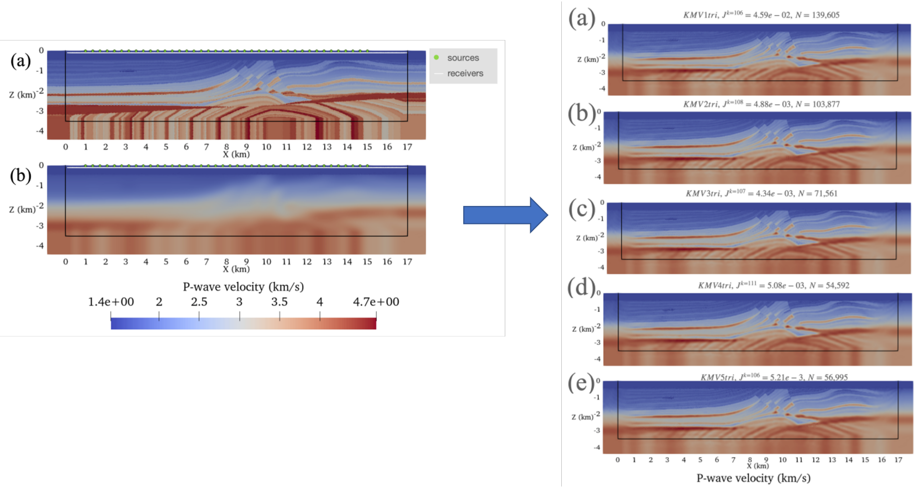



Figure: The performance of the forward.py wave propagator was assessed in the following benchmark 2D (a) and 3D meshes (b), where the ideal strong scaling line for each KMV element is represented as dashed and the number of degrees of freedom per core is annotated. For the 2D benchmark, the domain spans a physical space of 110 km by 85 km. A domain of 8 km by 8 km by 8 km was used in the 3D case. Both had a 0.287 km wide PML included on all sides of the domain except the free surface and a uniform velocity of 1.43 km/s.

# 2D Experiments




From Roberts et al. (2021b).

# Comparison of memory storage 2D

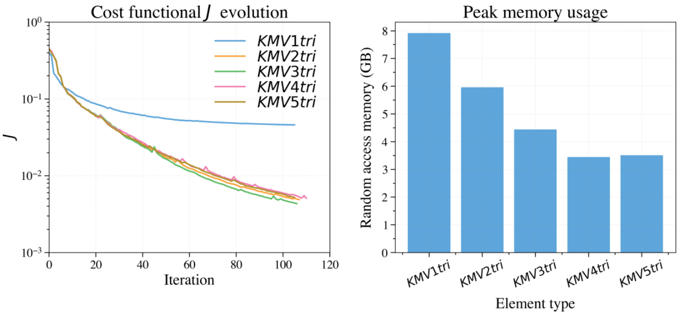



Figure: From Roberts et al. (2021b).

# Table of contents

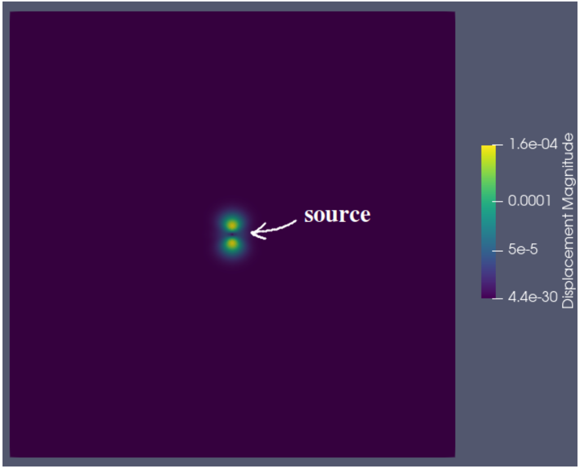
# Elastic FWI - forward model so far...

$$\rho \frac{\partial^2 \mathbf{u}}{\partial t^2} = \mathrm{div}\left(\lambda\, \mathrm{tr}\left(\boldsymbol{\epsilon}\right) \mathbf{I} + 2\mu\, \boldsymbol{\epsilon}\right) + \mathbf{f},$$
$$\boldsymbol{\epsilon} = \frac{1}{2}\left(\nabla \mathbf{u} + \nabla \mathbf{u}^{\intercal}\right). \tag{2}$$

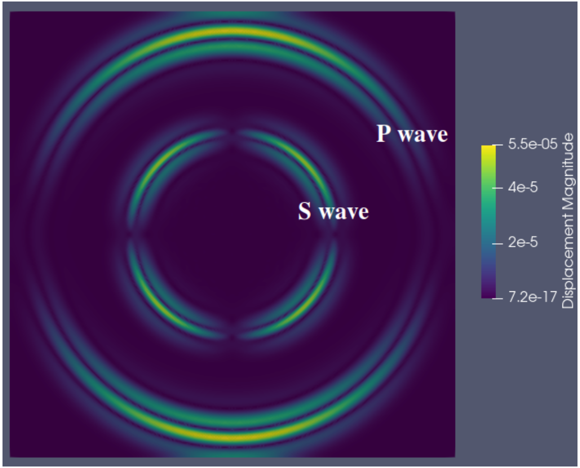
- $\mathbf{u}$: displacements
- $\rho$: density
- $\lambda$: 1st Lamé parameter (to be inverted by FWI)
- $\mu$: 2nd Lamé parameter (to be inverted by FWI)

$$\text{wave speeds} \begin{cases} c_p = \sqrt{\dfrac{\lambda + 2\mu}{\rho}} & \text{(P wave)} \\ \\ c_s = \sqrt{\dfrac{\mu}{\rho}} & \text{(S wave)} \end{cases} \tag{3}$$

# Elastic FWI - forward model, simple test

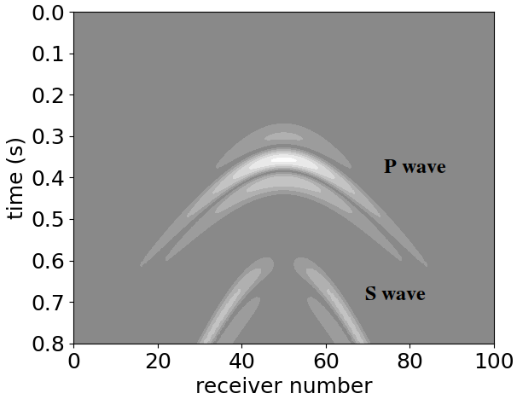


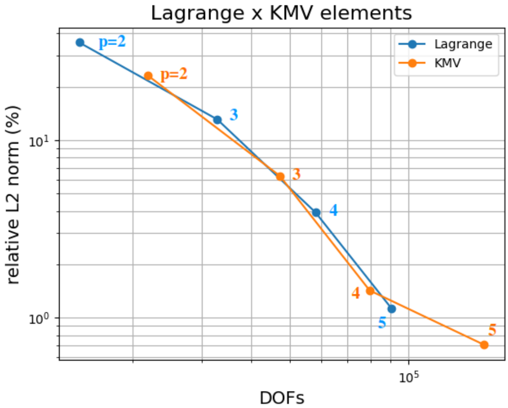

front wave, t=0.1 s




front wave, t=0.8 s

# Elastic FWI - forward model, simple test



# Ongoing work

- Weight-Adjusted Discontinuous Galerkin
- Multiple time integration schemes using irksome

# References I


Billette, F. and Brandsberg-Dahl, S. (2005).
The 2004 BP velocity benchmark.
In *67th EAGE Conference & Exhibition*, pages cp–1. European Association of Geoscientists & Engineers, European Association of Geoscientists & Engineers.

Chin-Joe-Kong, M., Mulder, W. A., and Van Veldhuizen, M. (1999).
Higher-order triangular and tetrahedral finite elements with mass lumping for solving the wave equation.
*J. Eng. Math.*, 35(4):405–426.

Geevers, S., Mulder, W., and van der Vegt, J. (2018).
New higher-order mass-lumped tetrahedral elements for wave propagation modelling.
*SIAM J. Sci. Comput.*, 40(5):A2830–A2857.

Grote, M. and Sim, I. (2010).
Efficient PML for the wave equation.
*Global Science Preprint*, arXiv:1001.0319v1 [math.NA]:1–15.

Kaltenbacher, B., Kaltenbacher, M., and Sim, I. (2013).
A modified and stable version of a perfectly matched layer technique for the 3-d second order wave equation in time domain with an application to aeroacoustics.
*J. Comput. Phys.*, 235:407–422.

Karniadakis, G. and Sherwin, S. (2005).
*Spectral/hp Element Methods for Computational Fluid Dynamics*.
Oxford University Press.

# References II


Lyu, C., Capdeville, Y., and Zhao, L. (2020).
Efficiency of the spectral element method with very high polynomial degree to solve the elastic wave equation.
*Geophysics*, 85(1):T33–T43.

Patera, A. T. (1984).
A spectral element method for fluid dynamics: Laminar flow in a channel expansion.
*Journal of Computational Physics*, 54(3):468–488.

Rathgeber, F., Ham, D. A., Mitchell, L., Lange, M., Luporini, F., Mcrae, A. T., Bercea, G.-T., Markall, G. R., and Kelly, P. H. (2017).
Firedrake.
*ACM Trans. Math. Softw.*, 43(3):1–27.
Article 24.

Roberts, K. J., dos Santos Gioria, R., and Pringle, W. J. (2021a).
Seismicmesh: Triangular meshing for seismology.
*Journal of Open Source Software*, 6(57):2687.

Roberts, K. J., Olender, A., Franceschini, L., Kirby, R. C., Gioria, R. S., and Carmo, B. S. (2021b).
Full waveform inversion using triangular waveform adapted meshes.
*arXiv preprint arXiv:2108.10946*.

Thank you for listening!

## Acknowledgements

This research was carried out in association with the ongoing R&D project registered as ANP 20714-2, "Software technologies for modelling and inversion, with applications in seismic imaging" (University of Sao Paulo / Shell Brasil / ANP).

# spyro: Acoustic wave modeling

## Features

- Based on the Firedrake automated system for solution of PDEs using FEM.
  - Mass-lumped 2D and 3D using triangular and tetrahedral elements.
  - Spectral quadrilateral and hexahedral elements.
- Perfectly Matched Layer (PML) to absorb reflected waves in both 2D and 3D.
- Capable of doing the complete FWI loop:
  - Mesh-independent functional gradient using the optimize-then-discretize approach.
  - Sparse interpolation and injection with point sources or force sources.
- Integrated with automatic mesh generation software SeismicMesh (Roberts, et al. 2020) for triangles and tetrahedrals.

## Link

https://github.com/krober10nd/Spyro