SeisElastic2D v1.1
User's Manual

An open-source package for multiparameter full-waveform inversion in isotropic-, anisotropic- and visco-elastic media

30th July 2020

Contents

1 Introduction

1.1 General background

SeisElastic2D is an open-source package for multiparameter FWI in isotropic-, anisotropic- and visco-elastic media. This package consists of a set of FORTRAN 90 routines and SHELL scripts built upon several existing open-source packages of SPECFEM2D, seisDD, Seismic Unix (SU), etc. SPECFEM2D provides the forward and adjoint simulation solver based spectral-element method (Komatitsch and Tromp, 2005). In SeisElastic2D, the SPECFEM2D codes have been upgraded to calculate the sensitivity kernels and diagonal Hessian preconditioners for isotropic-, anisotropic- and visco-elastic FWI. The package seisDD was originally developed by Yuan et al., (2016) at Princeton University for double-difference adjoint tomography. SeisElastic2D inherits the modular structure, some inversion workflows, etc. from seisDD. SU provides efficient seismic data pre-processing tools.

SeisElastic2D grew out of the first author's PhD research at the University of Calgary, in multiparameter elastic FWI. It is a flexible inversion toolbox particularly focusing on evaluating model parameterizations within practical isotropic-, VTI-, TTI- and visco-elastic FWI formulations and applications. Different misfit functions, including cross-correlation traveltime and envelope-difference, are developed to reduce the nonlinearity of the inverse problem. The double-difference misfit function can provide higher resolution velocity models independent of source signatures (Yuan et al., 2016). Several special misfit functions, measuring amplitude variation and central-frequency shift, are developed for visco-elastic FWI. These misfit functions can invert for attenuation models more reliably by reducing the influences of velocity errors. The package is designed to be used on a high-performance computing (HPC) cluster supporting massively parallel interface (MPI) and satisfying the high computational requirements of large-scale inverse problem. The advanced algorithms, flexible inversion strategies, and efficient tools in SeisElastic2D make it a powerful tool for mitigating the difficulties of multiparameter elastic FWI and bridging the gap between academic studies and industrial applications.

1.2 Citation

Basics of this package including workflow, misfit functions, etc, were finished by Yanhua Q. Yuan and Frederik Simons at Princeton University for multiscale adjoint waveform tomography with surface and body waves using wavelets and double-difference adjoint tomography. Using this package for your own studies, please cite the following publications:

Yuan, O. Y. & Simons, F. J., 2014. Multiscale adjoint waveform-difference tomography using wavelets, Geophysics. 79, WA79-WA95.

Yuan, O. Y., Simons, F. J. & Bozdag, E., 2015. Multiscale adjoint waveform tomography for surface and body waves, Geophysics. 80, R281-R302.

Yuan, O. Y., Simons, F. J. & Tromp, J., 2016. Double-difference adjoint seismic tomography, Geophysical Journal International. 206, 1599-1618.

Using SeisElastic2D, we have applied isotropic- and VTI-elastic FWI with different model parameterizations to practical walk-away vertical seismic profile (W-VSP) data (Pan et al., 2018a, 2019c) and surface seismic data (Pan et al., 2019a) successfully. Visco-elastic FWI with a two-stage inversion approach and amplitude-based misfit functions is applied to the practical W-VSP data for hydrocarbon reservoir characterization (Pan and Innanen, 2019b). Please also cite the following publications for using SeisElastic2D:

Pan, W., Innanen, K. & Geng, Y., 2018. Elastic full-waveform inversion and parameterization analysis applied to walk-away vertical seismic profile data for unconventional (heavy oil) reservoir characterization, Geophysical Journal International. 213, 1934-1968.

Pan, W., Geng, Y. & Innanen, A. K., 2018. Interparameter tradeoff quantification and reduction in isotropic-elastic full-waveform inversion: synthetic experiments and Hussar land dataset application, Geophysical Journal International. 213, 1305-1333.

Pan, W., Innanen, K., Geng Y. & Li, J., 2019. Interparameter tradeoff quantification for isotropic-elastic full-waveform inversion with various model parameterizations, Geophysics. 84, R185-R206.

Pan, W. & Innanen, K., 2019. Amplitude-based misfit functions in viscoelastic full-waveform inversion applied to walk-away vertical seismic profile data, Geophysics. 84(5), B335-B351.

Pan, W., Innanen, K & Wang, Y., 2020. Parameterization analysis and field validation of VTI-elastic full-waveform inversion in a walk-away vertical seismic profile configuration, Geophysics. 85(3), B87-B107.

1.3 Support

The package was initiated by Yanhua O. Yuan and Frederik Simons at Princeton University.

Later, Wenyong Pan and Kris Innanen extended elastic FWI codes to VTI-, TTI- and visco-elastic media with different model parameterizations at University of Calgary supported by the Consortium for Research in Elastic Wave Exploration Seismology (CREWES) and National Science and Engineering Research Council of Canada (NSERC, CRDPJ 461179-13), and in part from the Canada First Research Excellence Fund. Wenyong Pan was also supported by SEG/Chevron Scholarship and Eyes High International Doctoral Scholarship.

The package is currently maintained and updated by Wenyong Pan at Institute of Geology and Geophysics, Chinese Academy of Sciences. For any concern about this package, please feel free to contact Wenyong Pan via wenyongpan@mail.iggcas.ac.cn

2 Getting started

2.1 Download

The latest version of SeisElastic2D is currently available at on Github. For downloading it, you just need to type the following command:

```
> git clone https://github.com/PanIGGCAS/SeisElastic2D_1.1.git
```

2.2 Install

SeisElastic2D can be installed on large parallel computing cluster and local workstation supporting OpenMPI. It is built based upon several existing open-source packages. For installing and running it successfully, some prerequisite packages are needed.

2.2.1 Prerequisite packages

(a) We strongly recommend the users compile the codes using icc and ifort. When compiling the codes using gcc and gfortran, some warnings and errors may appear. On a parallel computing cluster, the modules can be loaded using the following commands:

```
> module load mkl/11.3.3
# Intel MKL is a library of optimized math routines for scientific computing including fast Fourier transform, vector match, etc.

> module load intel/17.0.4
# Intel ifort compiler.

> module load openmpi/2.1.2
# OpenMPI is an open source high performance computing library.
```

On a local workstation, the users need to install intel compiler and openmpi.

a. Intel® Parallel Studio XE for Linux is available at
https://software.intel.com/content/www/us/en/develop/articles/qualify-for-free-software.html

b. openmpi_4.3.0 is available at https://www.open-mpi.org/software/ompi/v4.0/

In the package, we provide shell script (0_setup.sh) to set the paths for ifort and openmpi, as shown in the following:

```
alias ifort=’/home/wenyongp/intel/compilers_and_libraries_2020.1.217/linux/bin/intel64/ifort’
source /home/wenyongp/intel/bin/ifortvars.sh ia32
source /home/wenyongp/intel/bin/compilervars.sh intel64
export PATH=/home/wenyongp/Programs/Openmpi-4.0.3/bin:$PATH
export LD_LIBRARY_PATH=LD_LIBRARY_PATH:/home/wenyongp/Programs/Openmpi-4.0.3/lib
```

(b) SHELL scripts are provided by SeisElastic2D to call the Seismic Unix (SU) tools for preprocessing the observed and synthetic seismic data. SU is available at:

https://github.com/JohnWStockwellJr/SeisUnix.

SU user manual is available at:

https://github.com/JohnWStockwellJr/SeisUnix/wiki.

(c) Python (version higher than 2.7) is needed to plot the sensitivity kernels and models in SeisElastic2D. Python can be loaded in the cluster directly:

```
> module load python/2.7-anaconda-5.0.1
```

or, the users can install Python personally.

(d) The CREWES Matlab toolbox, which is available at https://www.crewes.org/ResearchLinks/FreeSoftware/, is needed to have some basic processing and operations.

2.2.2 Paths Setting

Shell script (0_setup.sh) is provided to set the path for running the experiments:

```
> source 0_setup.sh
```

Type the following command to check the settings:

```
> echo $SeisElastic2D
```

2.2.3 SPECFEM2D

SPECFEM2D package provides the forward and adjoint simulation solver using the spectral-element method in isotropic-, anisotropic- and visco-elastic media. In SeisElastic2D, the SPECFEM2D package has been upgraded to calculate the sensitivity kernels for isotropic-, VTI-, TTI- and visco-elastic FWI.

For installing the upgraded SPECFEM2D package, please following the steps below:

```
> cd $SeisElastic2D/specfem2d
> make clean
> ./configure FC=ifort --with-mpi
> make all
```

SPECFEM2D relies on the SCOTCH library to partition meshes. The SCOTCH library is an open-source package, which provides efficient static mapping, graph and mesh partitioning routines. You may need to install SCOTCH first following the instruction in "./scotch_6.0.4/INSTALL.txt":

```
> cd $SeisElastic2D/specfem2d/src/meshfem2D
> tar -zxvf scotch_6.0.4.tar.gz
> cd ./scotch_6.0.4/src
```

In the file of “Makefile.inc”, set your paths of “icc” and “mpicc” for the parameters of “CCS”, “CCP” and “CCD”. Then,

```
> make scotch
> make ptscotch
```

If the scotch is installed successfully, then configure SPECFEM2D indicating the SCOTCH location:

```
> cd $SeisElastic2D/specfem2d
> ./configure FC=ifort --with-mpi --with-scotch-dir=./src/meshfem2D/scotch_6.0.4
> make all
```

Detailed introduction for installing SPECFEM2D can be found in its user manual, which is available at:

https://geodynamics.org/cig/software/SPECFEM2D/SPECFEM2D_manual.pdf

2.2.4 SeisElastic2D

A set of SHELL scripts are provided under the directories of “`./SeisElastic2D/workflow`” and “`./SeisElastic2D/scripts`” to the flexible inversion workflows. For examples:

```
.......
Modeling.sh
# forward modelling job
Kernel_attenuation.sh
# calculating sensitivity kernels in visco-elastic media
AdjointInversion_aniso.sh
# inversion workflow in anisotropic media
Forward_specfem2D.sh
# call SPECFEM2D forward modeling solver
adjoint_source_attenuation.sh
# calculating the adjoint source for attenuation sensitivity kernels
.......
```

The Fortran 90 routines are provide under the directories of “`./SeisElastic2D/SRC`” and “`./SeisElastic2D/lib/src`
For examples:

```
.......
optimization.f90
# different optimization methods for model update
data_misfit.f90
# line search and produce data misfit
myMath_module.f90
# basic math operations
adjoint_lib.f90
# calculate the adjoint sources for different misfit functions
.......
```

For compiling the Fortran 90 routines in SeisElastic2D:

```
> cd $SeisElastic2D/SeisElastic2D/lib
> make -f make_lib clean
> make -f make_lib
```

In the package, we have provided one single shell script of "1_compile.sh" to compile scotch, SPECFEM2D and SeisElastic2D.

3 Forward modeling

3.1 Creating model and acquisition files

The model files (e.g., P-wave velocity model) need to be created as inputs for forward modeling and inversion. Here, we introduce the methods for generating the model“`*.bin`” files. MATLAB scripts are provided in the directory of to create the files of “`*.xyz`”, “`source_wavelet.dat`”, “`sources.dat`” and “`STATIONS`”.

The “`*.xyz`” file contains the model size, grid size, model parameters, etc. It is needed as input for forward modeling using SPECFEM2D to create the model “`*.bin`” files. Its structure is introduced in the following:

```
0.0 0.0 625.0 1250.0
# $x_0$, $z_0$, $x_{\max}$ and $z_{\max}$
5.0 5.0
# grid size $\Delta x$ and $\Delta z$
126.0 251.0
# number of regular grids nx and nz


2709.7 3782.4 1622.6 2264.9 2241.6 2429.3 9998.0 9998.0 9998.0 9998.0 20490678567.4
44145713320.5 6282048933.9 13486849827.9 -5038657295.5 4424296466.3 16719295970.8 36115821421.2
-1391304569.7 1122544724.4 6045315465.3 13087660351.8 0.0 0.0 0.0 0.0 0.0 0.0
# min vp, max vp, min vs, max vs, min rho, max rho, min Qp, max Qp, min Qs, max Qs,
min c11, max c11, min c13, max c13, min c15, max c15, min c33, max c33, min c35, max
c35, min c55, max c55, min c12, max c12, min c23, max c23, min c25, max c25


0.0 0.0 2737.2 1639.1 2244.6 9998.0 9998.0 21006604729.0 6436022154.5 -1252765856.1
17171074633.6 -295641739.7 6223245604.6 0.0 0.0 0.0
# grid location $x$, $z$ and values of the models vp, vs, rho, Qp, Qs, c11, c13, c15,
c33, c35, c55, c12, c23, c25
...............
...............
```

The file of “`source_wavelet.dat`” contains the source time function. You need to generate the

```
0.000000 0.000803
0.000100 0.000594
0.000200 0.000377
# value and time
.........
.........
```

The file of “`sources.dat`” contains the locations of sources:

```
75.000000 10.000000
195.000000 10.000000
315.000000 10.000000
# depth and horizontal location of the sources
```

```
.........
.........
```

The “STATIONS” file contains the locations of receivers:

```
S0100 AA 15.00 10.00 0.00 0.00
S0101 AA 30.00 10.00 0.00 0.00
# receiver number, depth and horizontal position
.........
.........
```

3.2 Forward modeling

The created files of “*.xyz”, “source_wavelet.dat”, “sources.dat” and “STATIONS” are used as inputs for the forward modelling simulation. The files of “./DATA/Par_file”, “./DATA/interfaces.dat” and “./DATA/SOURCE” need to be edited for doing forward modeling simulation in isotropic-, anisotropic- and visco-elastic media. Here, we introduce some key parameters in “./DATA/Par_file”:

```
.........
SIMULATION_TYPE = 1
# 1:  forward modeling; 3:  adjoint simulation
SAVE_FORWARD = .true.
# for adjoint simulation, set as false
MODEL = default
# default:  read model from *.xyz file; binary:  read model from *.bin files
ANISO = .true.
# true:  forward modeling in anisotropic media; false:  forward modeling in isotropic
media.
M_PAR = ttiec
# Isotropic-elastic media
isodv:  velocity-density parameterization
isodl:  lame-density parameterization
isodm:  modulus-density parameterization
isodip:  impedance-density parameterization
isovipi:  velocity-impedance-I parameterization
isovipii:  velocity-impedance-II parameterization
isosigma:  velocity-vp/vs ratio parameterization
VTI-elastic media
vtiec:  elastic-constant parameterization
vtithom:  velocity-Thomsen's parameter parameterization
vtithom2:  velocity-anelliptic parameter parameterization
vtivel:  velocity-only parameterization
TTI-elastic media
ttiec:  rotated elastic-constant parameterization
ttiecu:  unrotated elastic-constant parameterization
ttithom:  velocity-Thomsen's parameter parameterization
The physical parameters in these model parameterizations are illustrated in Appendix.
For different model parameterizations, different model ``*.bin" files will be produced.
SAVE_MODEL = binary
..........
ATTENUATION_VISCOELASTIC_SOLID = .false.
```

```
# true:  visco-elastic media forward modeling; false:  non-attenuative elastic media
forward modeling.
nt = 5000
# maximum time steps
deltat = 2.0d-4
# time interval
..........
TOMOGRAPHY_FILE = ./DATA/model_file_bp_tti_true_vti.xyz
# model file name and location
xmin = 0.d0
# minimum horizontal distance in m
xmax = 625
# maximum horizontal distance in m
nx = 30
# number of meshes in horizontal direction
1 30 1 60 1
# number of meshes in horizontal and vertical directions
```

In "./DATA/SOURCE", the source parameter can be set:

```
..........
source_type = 2
# 1:  P-SV souce; 2:  moment tensor source.
time_function_type = 1
# 1:  Ricker source wavelet; 8:  read source function from data file
name_of_source_file = "DATA/source_wavelet.dat"
# location the source wavelet file
f0 = 20
# dominant frequency of the source wavelet
..........
```

Run the script "run_this_example.sh" for forward modeling with parallel computing:

```
> sbatch ./run_this_example.sh
```

In "run_this_example.sh", the meshes are created first by:

```
> ./xmeshfem2D > mesh_output.txt
```

Then SPECFEM2D forward modelling solver is run by:

```
> mpirun -np $NPROC ./xspecfem2D > specfem_output.txt
```

The generated seismic data files are stored under the directory of "./OUTPUT_FILES":

```
.......
../OUTPUT_FILES/image0001000.jpg
../OUTPUT_FILES/image0001500.jpg
# wavefield snapshots
```

```
.......
../model_true_bin/proc000003_c11.bin
../model_true_bin/proc000003_c13.bin
../model_true_bin/proc000003_c15.bin
../model_true_bin/proc000003_c33.bin
../model_true_bin/proc000003_c35.bin
../model_true_bin/proc000003_c55.bin
# produced model "*.bin" files
.......
```

The model "`*.bin`" files will be created under the folder of "`./model_true_bin`":

```
.......
../OUTPUT_FILES/Ux_file_single.su
../OUTPUT_FILES/Uz_file_single.su
# recorded seismic data in SU format
.......
```

The initial models can be created following the same process.

3.3 Plotting models

The python script is provided to plot the models:

```
> python $VISUALIZE/plot_bin_kernel ./model_true_bin c11 8
```

One Shell script of "2_forward_modeling_example.sh" is provided to run the forward modelling example and plot the models.

4 Inversion

4.1 Setting parameters

The parameters in “`parameter`” and “`./DATA/Par_file`” need to be set or upgraded for running the experiments. The “`parameter`” file contains the inversion parameters:

```
#!/bin/bash

Job_title='CW_TTI_syn_tti'
# parallel computing job title
system='slurm'
# parallel computing system: slurm, lsf or pbs
queue='tiger'
ntasks=32
# number of tasks
NPROC_SPECFEM=8
# number of processors with SPECFEM2D
compiler='ifort'
# which compiler
GPU_MODE=false
# GPU computing or not
job='kernel'
# numerical experiments: modeling, kernel or inversion
solver='SPECFEM2D'
# forward modeling solver: SPECFEM2D or specfem3D
ReStart=true
# true: relaunch; false: start from current status

package_path="$SeisElastic2D/SeisElastic2D"
# source codes location
specfem_path="$SeisElastic2D/specfem2d"
# SPECFEM2D location
working_path="$WORK_PATH"
# working path

ExistDATA=true
# using existed observed data or not
DATA_DIR="$ALL_DATA/CW_TTI_obs_data_30Hz"
# If ExistDATA=true, using the observed data provided under DATA_DIR. If ExistDATA=false, doing forward modelling and save the data under DATA_DIR

target_velocity_dir="$FWI_EXAMPLES/CW_TTI_syn_tti/model_true_bin"
# true velocity models location
initial_velocity_dir="$FWI_EXAMPLES/CW_TTI_syn_tti/model_true_bin"
# initial velocity models location

NSTEP=4000
# maximum time steps when forward modelling for kernel calculation
deltat=2.0e-4
```

```
# sampling interval
f0=30
# dominant frequency of the source wavelety
NREC=162
# number of sources
NSRC=16
# number of receivers
data_list='x,z'
# data type:  x, z, y or p

measurement_list=WD
# misfit functions
WD: waveform-difference
ED: envelope-difference
ER: envelope-ratio
CC: cross-correlation traveltime
RD: RMS amplitude-difference
RR: RMS amplitude-ratio
IP: Instantaneous phase
SR: spectral amplitude-ratio
SM: spectral amplitude-matching
MT: multi-taper traveltime
MA: multi-taper amplitude
Expressions of these misfit functions and the corresponding adjoint sources are illustrated
in Appendix
misfit_type_list=AD
# misfit function type
AD: absolute difference
DD: double difference
cc_threshold=0.9
# cross-correlation threshold parameter
initial_step_length=0.04
# initial step length in line search method
min_step_length=0.00001
# the minimum step length tolerance
misfit_ratio_initial=0.00001
misfit_ratio_previous=0.00001

kernel_list='tti_ec_c11_kernel,tti_ec_c13_kernel,
tti_ec_c15_kernel,tti_ec_c33_kernel,tti_ec_c35_kernel,
tti_ec_c55_kernel,tti_ec_rho_kernel'
# list of kernels for TTI-elastic FWI within TTIEC model parameterization
precond=false
# diagonal Hessian preconditioning or not
precond_list='pdh_tti_ec_c11,pdh_tti_ec_c13,
pdh_tti_ec_c15,pdh_tti_ec_c33,pdh_tti_ec_c35, pdh_tti_ec_c55,pdh_tti_ec_rho'
# the diagonal Hessian preconditioners corresponding to the kernels.  The numbers
of preconditioners must be in consistent with the kernels.
model_list='c11,c13,c15,c33,c35,c55,rho'
```

```
# list of model parameters
opt_scheme=SD
# optimization methods
SD: steepest descent
CG: nonlinear conjugate gradient
QN: l-BFGS
iter_start=1
# starting iteration number
iter_end=60
# the maximum iteration number

SU_process=false
# pre-processing the seismic data or not
SU_process_path="$FWI_EXAMPLES/CW_TTI_syn_tti/SU_process"
# location of the pre-processing scripts
Wscale=0
wavelet_path="$SeisElastic2D/SeisElastic2D/lib/WT_basis"
# location of the wavelet transform basis

smooth=false
# smoothing the kernels or not
sigma_x=10
# smoothing radius in horizontal direction
sigma_z=20
# smoothing radius in vertical direction

MASK_SOURCE=false
# mask source or not
source_radius=20
# mask radius at source location
MASK_STATION=true
# mask station or not
station_radius=20
# mask radius at receiver location
MASK_MODEL=false
# mask model or not
DISPLAY_DETAILS=false
# display details or not
VISCOELASTIC=false
# viscoelastic forward modelling and inversion or not
measurement_attenuation=AMP
# measurement type for attenuation sensitivity kernel calculation
```

Several parameters need to be upgraded in "`./DATA/Par_file`":

```
.........
MODEL = binary
# model file should be changed to binary
SAVE_MODEL = default
```

```
# this parameter should be set as default
..........
nt = 4000
deltat = 2.0d-4
# the maximum time steps and sampling interval should be consistent with those in
‘‘parameter" file.
..........
```

4.2 Pre-processing data

Under directory of "./SU_process", SHELL scripts "process_obs.sh", "process_syn.sh" and "process_adj.sh" are provided to pre-process the observed data, synthetic data and adjoint sources. The file of "process_par" contains the processing parameters. For example, in "process_par", the parameters are set as:

```
..........
filt_freq=12,14,34,36
amps=0,1,1,0
# band pass filtering parameters
tmute1=0.25,0.8
xmute1=1,222
key1='tracl'
mode1=1
# muting parameters
..........
```

In the script of "process_obs.sh", SU tools are called to pre-process the observed data:

```
..........
sufilter f=$filt_freq amps=$amps <in_file >out_file
cp out_file in_file
..........
sumute <in_file tmute=$tmute1 xmute=$xmute1 key=$key1 mode=$mode1>out_file
cp out_file in_file
..........
```

One Shell script of "3_inversion_example.sh" is provided to generate the observed data and run the inversion experiment.

One Shell script of "4_plot_results.sh" is provided to plot the inversion results.

4.3 Numerical experiments

4.3.1 Generating observed data

For synthetic inversion experiments, the observed data can be generated first. The following parameters in "parameter" file need to be modified:

```
..........
job='modeling'
# job should be changed to modeling
ExistDATA=false
# set ExistDATA as false.  the generated data will be stored under DATA_DIR
DATA_DIR="$ALL_DATA/CW_TTI_data/CW_TTI_obs_data"
```

```
# observed data directory
..........
```

Submit the job:

```
> bash submit.sh
```

If the job runs successfully, the observed data will be produced under DATA_DIR.

4.3.2 Kernel calculation

Several parameters can be set to calculate the sensitivity kernels:

```
..........
job='kernel'
# job should be changed to kernel
ExistDATA=true
# turn ExistDATA to true
..........
measurement_list=WD
# using the standard WD misfit function
..........
smooth=true
# smoothing the kernels
sigma_x=20
sigma_z=40
..........
```

Submit the job:

```
> bash submit.sh
```

If the job runs successfully, the sensitivity kernels will be generated under "./submit_job/RESULTS/kernel/Scale0_WD_AD/msifit_kernel". For plotting the kernels:

```
> cd ./RESULTS/kernel/Scale0_WD_AD
> python $VISUALIZE/plot_bin_kernel ./misfit_kernel tti_ec_c11_kernel_smooth 8
```

4.3.3 Inversion

If the forward modeling and kernel jobs run successfully, inversion experiments can be carried out. Several inversion parameters need to be set:

```
..........
job='inversion'
# change job to inversion
..........
initial_step_length=0.04
```

```
# initial step length in line search method
min_step_length=0.00001
# the minimum step length tolerance
misfit_ratio_initial=0.00001
misfit_ratio_previous=0.00001

kernel_list='tti_ec_c11_kernel,tti_ec_c13_kernel,
tti_ec_c15_kernel,tti_ec_c33_kernel,tti_ec_c35_kernel,
tti_ec_c55_kernel,tti_ec_rho_kernel'
# list of kernels for TTI-elastic FWI within TTIEC model parameterization
precond=false
# diagonal Hessian preconditioning or not
precond_list='pdh_tti_ec_c11,pdh_tti_ec_c13,
pdh_tti_ec_c15,pdh_tti_ec_c33,pdh_tti_ec_c35, pdh_tti_ec_c55,pdh_tti_ec_rho'
# the diagonal Hessian preconditioners corresponding to the kernels.  The numbers
of preconditioners must be in consistent with the kernels.
model_list='c11,c13,c15,c33,c35,c55,rho'
# list of model parameters
opt_scheme=SD
# optimization methods
SD: steepest descent
CG: nonlinear conjugate gradient
QN: l-BFGS
iter_start=1
# starting iteration number
iter_end=60
# the maximum iteration number

SU_process=false
# pre-processing the seismic data or not
..........
```

5 Example

5.1 VTI-elastic FWI with VTIEC parameterization

In this section, we give synthetic example of VTI-elastic FWI with the VTIEC model parameterization. Figures 1a-1d shows the true c_{11}, c_{13}, c_{33} and c_{55} models. Figures 1e-1h are the corresponding initial models. The true and initial ρ models are homogeneous with constant value of 2000 g/cm^3. The sources and receivers are arranged regularly on the left and right sides of the model. We generate the observed data using a Ricker source wavelet with dominant frequency of 30 Hz. A multi-scale inversion strategy is adopted by expanding the frequency band from [3 Hz, 10 Hz] to [3 Hz, 20 Hz] and then [3 Hz, 30 Hz].

Figures 2a-2d are the sensitivity kernels $K_{c_{11}}^{\mathrm{vtiec}}$, $K_{c_{13}}^{\mathrm{vtiec}}$, $K_{c_{33}}^{\mathrm{vtiec}}$ and $K_{c_{55}}^{\mathrm{vtiec}}$ for the elastic constants of c_{11}, c_{13}, c_{33} and c_{55} within the frequency band of [3 Hz, 10 Hz]. Figures 2e-2h are the corresponding diagonal Hessian preconditioners. The sensitivity kernel $K_{c_{13}}^{\mathrm{vtiec}}$ for c_{13} suffers from strong trade-offs. Thus, at each frequency band, the c_{11}, c_{33} and c_{55} models are simultaneously inverted first. Then, the c_{13} model is inverted. Figure 3 illustrates the final inverted models.

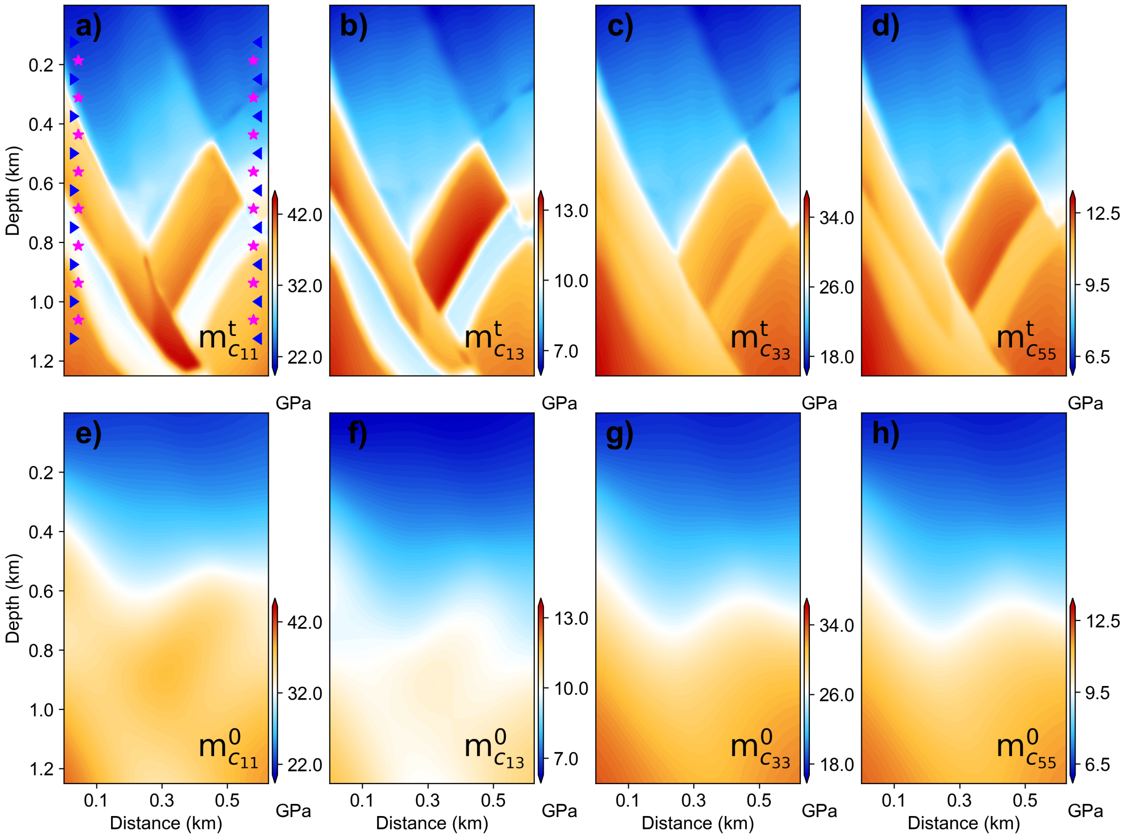

Figure 1: (a-d) are the true c_{11}, c_{13}, c_{33} and c_{55} models within VTIEC model parameterization; (e-h) are the corresponding initial models. The cyan stars and blue triangles in (a) indicate the locations of the sources and receivers.

A Model parameterizations in isotropic-, VTI- and TTI-elastic FWI

Different model parameterizations are provided for isotropic-, VTI- and TTI-elastic FWI in SeisElastic2D, as illustrated in Table 1. Here, we illustrate the sensitivity kernels in these model parameterizations, the names of kernels, diagonal Hessian approximations and model names when running the inversion experiments.

Within the DM model parameterization for isotropic-elastic FWI, the expressions of the sensitivity kernels are given by:

$$K_\kappa^{\mathrm{dm}} = -\kappa \partial_i u_i^\dagger \partial_k u_k, \tag{1a}$$

$$K_\mu^{\mathrm{dm}} = -\mu \left[\partial_j u_i^\dagger \left(\partial_i u_j + \partial_j u_i \right) - \frac{2}{3} \partial_i u_i^\dagger \partial_k u_k \right], \tag{1b}$$

$$K_\rho^{\mathrm{dm}} = -\rho u_i^\dagger \partial_t^2 u_i. \tag{1c}$$

Table 1: Model parameterizations for isotropic-, VTI- and TTI-elastic FWI.

Media	Model parameterization	Acronym	Physical parameters
Isotropic-elastic	Modulus-density	DM	κ, μ and ρ
	Velocity-density	DV	α, β and ρ
	Lamé-density	DL	λ, μ and ρ
	Impedance-density	DIP	I_P, I_S and ρ
	Velocity-impedance-I	VIP-I	α, β and I_P
	Velocity-impedance-II	VIP-II	α, β and I_S
	Velocity-α/β ratio	V-ratio	α, σ and ρ
VTI-elastic	Elastic-constants	VTIEC	c_{11}, c_{13}, c_{33}, c_{55} and ρ
	Velocity-Thomsen's parameter	VTITH	α_v, β_v, ε, δ and ρ
	Velocity-Anelliptic parameter	VTIAN	α_h, β_v, ε, η and ρ
	Velocity	VTIVEL	α_v, β_v, α_h, α_n and ρ
TTI-elastic	Rotated elastic-constants	TTIEC	c_{11}, c_{13}, c_{15}, c_{33} c_{35}, c_{55} and ρ
	Unrotated elastic-constants	TTIECU	c_{11}^{u}, c_{13}^{u}, c_{33}^{u}, c_{55}^{u}, θ and ρ
	Velocity-Thomsen's parameter	TTITH	α_v, β_v, ε, δ, θ and ρ

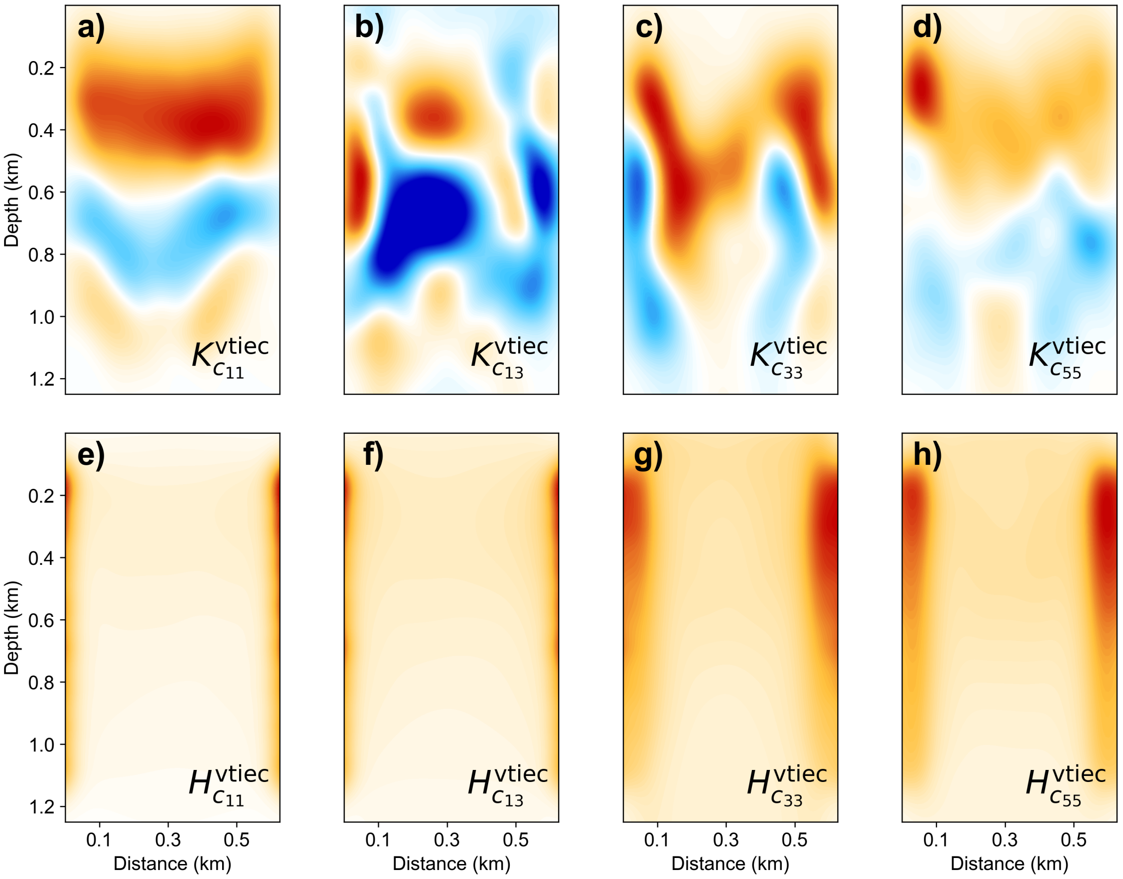

Figure 2: (a-d) are the sensitivity kernels for model parameters of c_{11}, c_{13}, c_{33} and c_{55}; (e-h) are the corresponding diagonal Hessian preconditioners.

where $u_i^\dagger$ indicates the adjoint displacement field. In the file of "`parameter`", the "`kernel_list`", "`precond_list`" and "`model_list`" are:

```
kernel_list='kappa_kernel, mu_kenrel, rho_kernel'
precond_list='pdh_kappa, pdh_mu, pdh_rho'
model_list='kappa, mu, rho'
```

DV parameterization:

$$K_\alpha^{\mathrm{dv}} = 2\left(1 + \frac{4}{3}\frac{\mu}{\kappa}\right) K_\kappa^{\mathrm{dm}}, \tag{2a}$$

$$K_\beta^{\mathrm{dv}} = 2\left(K_\mu^{\mathrm{dm}} - \frac{4}{3}\frac{\mu}{\kappa} K_\kappa^{\mathrm{dm}}\right), \tag{2b}$$

$$K_\rho^{\mathrm{dv}} = K_\rho^{\mathrm{dm}} + K_\kappa^{\mathrm{dm}} + K_\mu^{\mathrm{dm}}. \tag{2c}$$

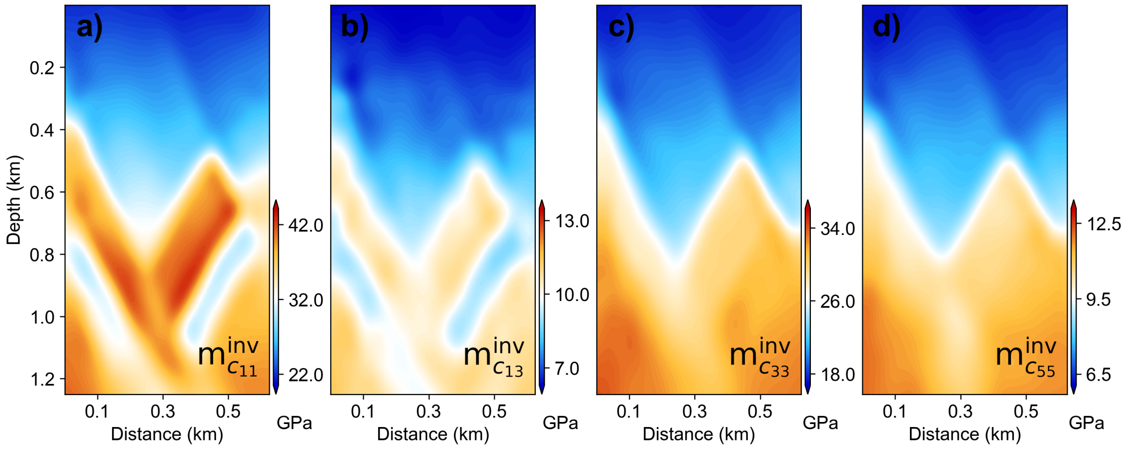

Figure 3: (a-d) are the inverted c_{11}, c_{13}, c_{33} and c_{55} models in VTI-elastic FWI with the VTIEC model parameterization.

```
kernel_list=’alpha_kernel, beta_kenrel, rhop_kernel’
precond_list=’pdh_vp, pdh_vs, pdh_rhop’
model_list=’vp, vs, rho’
```

DL parameterization:

$$K_{\lambda}^{\mathrm{dl}} = \left(1 - \frac{2}{3}\frac{\mu}{\kappa}\right) K_{\kappa}^{\mathrm{dm}} = \frac{\lambda}{2\,(\lambda + 2\mu)} K_{\alpha}^{\mathrm{dv}}, \tag{3a}$$

$$K_{\mu}^{\mathrm{dl}} = K_{\mu}^{\mathrm{dm}} + \frac{2}{3}\frac{\mu}{\kappa} K_{\kappa}^{\mathrm{dm}} = \frac{\mu}{\lambda + 2\mu} K_{\alpha}^{\mathrm{dv}} + \frac{K_{\beta}^{\mathrm{dv}}}{2}, \tag{3b}$$

$$K_{\rho}^{\mathrm{dl}} = K_{\rho}^{\mathrm{dm}} = -\frac{K_{\alpha}^{\mathrm{dv}}}{2} - \frac{K_{\beta}^{\mathrm{dv}}}{2} + K_{\rho}^{\mathrm{dv}}. \tag{3c}$$

```
kernel_list=’dl_lambda_kernel, dl_mu_kenrel, dl_rho_kernel’
precond_list=’pdh_dl_lambda, pdh_dl_mu, pdh_dl_rho’
model_list=’lambda, mu, rho’
```

DIP parameterization:

$$K_{I_P}^{\mathrm{dip}} = 2\left(1 + \frac{4}{3}\frac{\mu}{\kappa}\right) K_{\kappa}^{\mathrm{dm}} = K_{\alpha}^{\mathrm{dv}}, \tag{4a}$$

$$K_{I_S}^{\mathrm{dip}} = 2\left(K_{\mu}^{\mathrm{dm}} - \frac{4}{3}\frac{\mu}{\kappa} K_{\kappa}^{\mathrm{dm}}\right) = K_{\beta}^{\mathrm{dv}}, \tag{4b}$$

$$K_{\rho}^{\mathrm{dip}} = -K_{\kappa}^{\mathrm{dm}} - K_{\mu}^{\mathrm{dm}} + K_{\rho}^{\mathrm{dm}} = -K_{\alpha}^{\mathrm{dv}} - K_{\beta}^{\mathrm{dv}} + K_{\rho}^{\mathrm{dv}}. \tag{4c}$$

```
kernel_list=’dip_ip_kernel, dip_is_kenrel, dip_rho_kernel’
precond_list=’pdh_dip_ip, pdh_dip_is, pdh_dip_rho’
```

```
model_list='ip, is, rho'
```

V-IP-I parameterization:

$$\begin{aligned}
K_{\alpha}^{\text{vipi}} &= 2\left(1+\frac{8}{3}\frac{\mu}{\kappa}\right)K_{\kappa}^{\text{dm}} - K_{\mu}^{\text{dm}} - K_{\rho}^{\text{dm}} = K_{\alpha}^{\text{dv}} - K_{\rho}^{\text{dv}}, && \text{(5a)}\\
K_{\beta}^{\text{vipi}} &= 2\left(K_{\mu}^{\text{dm}} - \frac{4}{3}\frac{\mu}{\kappa}K_{\kappa}^{\text{dm}}\right) = K_{\beta}^{\text{dv}}, && \text{(5b)}\\
K_{I_P}^{\text{vipi}} &= K_{\rho}^{\text{dm}} + K_{\kappa}^{\text{dm}} + K_{\mu}^{\text{dm}} = K_{\rho}^{\text{dv}}. && \text{(5c)}
\end{aligned}$$

```
kernel_list='dvipi_vp_kernel, dvipi_vs_kenrel, dvipi_ip_kernel'
precond_list='pdh_dvipi_vp, pdh_dvipi_vs, pdh_dvipi_ip'
model_list='vp, vs, ip'
```

V-IP-II parameterization:

$$\begin{aligned}
K_{\alpha}^{\text{vipii}} &= 2\left(1+\frac{4}{3}\frac{\mu}{\kappa}\right)K_{\kappa}^{\text{dm}} = K_{\alpha}^{\text{dv}}, && \text{(6a)}\\
K_{\beta}^{\text{vipii}} &= K_{\mu}^{\text{dm}} - \frac{11}{3}\frac{\mu}{\kappa}K_{\kappa}^{\text{dm}} - K_{\rho}^{\text{dm}} = K_{\beta}^{\text{dv}} - K_{\rho}^{\text{dv}}, && \text{(6b)}\\
K_{I_S}^{\text{vipii}} &= K_{\rho}^{\text{dm}} + K_{\kappa}^{\text{dm}} + K_{\mu}^{\text{dm}} = K_{\rho}^{\text{dv}}. && \text{(6c)}
\end{aligned}$$

```
kernel_list='dvipii_vp_kernel, dvipii_vs_kenrel, dvipii_is_kernel'
precond_list='pdh_dvipii_vp, pdh_dvipii_vs, pdh_dvipii_is'
model_list='vp, vs, is'
```

$$\begin{aligned}
K_{\alpha}^{\text{sigma}} &= K_{\alpha}^{\text{dv}} + K_{\beta}^{\text{dv}}, && \text{(7a)}\\
K_{\sigma}^{\text{sigma}} &= -K_{\beta}^{\text{dv}}, && \text{(7b)}\\
K_{\rho}^{\text{sigma}} &= K_{\rho}^{\text{dv}}. && \text{(7c)}
\end{aligned}$$

```
kernel_list='sigma_vp_kernel, sigma_sigma_kenrel, sigma_rho_kernel'
precond_list='pdh_sigma_vp, pdh_sigma_sigma, pdh_sigma_rho'
model_list='vp, sigma, rho'
```

For VTIEC model parameterization in VTI-elastic media, the sensitivity kernels are given by

$$\begin{aligned}
K_{c_{11}}^{\text{vtiec}} &= -\ll c_{11}\partial_x u_x^{\dagger}\partial_x u_x \gg, && \text{(8a)}\\
K_{c_{13}}^{\text{vtiec}} &= -\ll c_{13}\left[\left(\partial_x u_x^{\dagger}\partial_z u_z + \partial_z u_z^{\dagger}\partial_x u_x\right) + \partial_z u_z^{\dagger}\partial_x u_x + \partial_x u_x^{\dagger}\partial_z u_z\right] \gg, && \text{(8b)}\\
K_{c_{33}}^{\text{vtiec}} &= -\ll c_{33}\partial_z u_z^{\dagger}\partial_z u_z \gg, && \text{(8c)}\\
K_{c_{55}}^{\text{vtiec}} &= -\ll c_{55}\left(\partial_z u_x^{\dagger} + \partial_x u_z^{\dagger}\right)\left(\partial_z u_x + \partial_x u_z\right) \gg, && \text{(8d)}\\
K_{\rho}^{\text{vtiec}} &= K_{\rho}^{\text{dm}}. && \text{(8e)}
\end{aligned}$$

```
kernel_list='hti_ec_c11_kernel, hti_ec_c13_kernel, hti_ec_c33_kernel, hti_ec_c55_kernel,
hti_ec_rho_kernel'
precond_list='pdh_hti_ec_c11, pdh_hti_ec_c13, pdh_hti_ec_c33, pdh_hti_ec_c55, pdh_hti_ec_rho'
model_list='c11, c13, c33, c55, rho'
```

$$\begin{aligned}
K_{\alpha_v}^{\text{vtith}} &= 2K_{c_{33}}^{\text{vtiec}} + 2K_{c_{11}}^{\text{vtiec}} + \frac{2\alpha_v^4\left(2\delta+1\right) - 2\alpha_v^2\beta_v^2\left(\delta+1\right)}{M\left(M-\beta_v^2\right)}K_{c_{13}}^{\text{vtiec}}, && \text{(9a)}\\
K_{\beta_v}^{\text{vtith}} &= 2K_{c_{44}}^{\text{vtiec}} - \frac{2\beta_v^2\left(\alpha_v^2\delta - \beta_v^2 + \alpha_v^2\right)}{M\left(M-\beta_v^2\right)}K_{c_{13}}^{\text{vtiec}} - \frac{2\beta_v^2}{M-\beta_v^2}K_{c_{13}}^{\text{vtiec}}, && \text{(9b)}\\
K_{\varepsilon}^{\text{vtith}} &= \frac{2\varepsilon}{2\varepsilon+1}K_{c_{11}}^{\text{vtiec}}, && \text{(9c)}\\
K_{\delta}^{\text{vtith}} &= \frac{\delta\alpha_v^2\left(\alpha_v^2-\beta_v^2\right)}{M\left(M-\beta_v^2\right)}K_{c_{13}}^{\text{vtiec}}, && \text{(9d)}\\
K_{\rho}^{\text{vtith}} &= K_{\rho}^{\text{vtiec}} + \alpha_v^2K_{c_{33}}^{\text{vtiec}} + \beta_v^2K_{c_{55}}^{\text{vtiec}} + \left(M-\beta_v^2\right)K_{c_{13}}^{\text{vtiec}}, && \text{(9e)}
\end{aligned}$$

where M in the denominator is given by

$$M = \sqrt{2\delta\alpha_v^2\left(\alpha_v^2-\beta_v^2\right) + \left(\alpha_v^2-\beta_v^2\right)^2}. \qquad (10)$$

```
kernel_list='hti_thom_alpha_kernel, hti_thom_beta_kernel, hti_thom_epsilon_kernel,
hti_thom_delta_kernel, hti_thom_rhop_kernel'
precond_list='pdh_hti_thom_alpha, pdh_hti_thom_beta, pdh_hti_thom_epsilon,
pdh_hti_thom_delta, pdh_hti_thom_rhop'
model_list='hti_thom_vp, hti_thom_vs, hti_thom_epsilon,
hti_thom_delta, hti_thom_rhop'
```

The sensitivity kernels in VTIAN and VTIVEL parameterizations for VTI-elastic FWI can be expressed using the sensitivity kernels in VTITH parameterization, as listed in the following equations:

$$\begin{aligned}
K_{\alpha_h}^{\text{vtian}} &= K_{\alpha_v}^{\text{vtith}}, && \text{(11a)}\\
K_{\beta_v}^{\text{vtian}} &= K_{\beta_v}^{\text{vtith}}, && \text{(11b)}\\
K_{\varepsilon}^{\text{vtian}} &= K_{\varepsilon}^{\text{vtith}} + \frac{\varepsilon}{\varepsilon-\eta}K_{\delta}^{\text{vtith}} - \frac{\varepsilon}{1+2\varepsilon}K_{\alpha_v}^{\text{vtith}}, && \text{(11c)}\\
K_{\eta}^{\text{vtian}} &= -\left(\frac{2\eta}{1+2\eta} + \frac{\eta}{\varepsilon-\eta}\right)K_{\delta}^{\text{vtith}}, && \text{(11d)}\\
K_{\rho}^{\text{vtian}} &= K_{\rho}^{\text{vtith}}, && \text{(11e)}
\end{aligned}$$

```
kernel_list='hti_thom2_alpha_kernel, hti_thom2_beta_kernel, hti_thom2_epsilon_kernel,
hti_thom2_eta_kernel, hti_thom2_rhop_kernel'
precond_list='pdh_hti_thom2_alpha, pdh_hti_thom2_beta, pdh_hti_thom2_epsilon,
pdh_hti_thom2_eta, pdh_hti_thom2_rhop'
model_list='hti_thom2_vp, hti_thom2_vs, hti_thom2_epsilon,
hti_thom2_eta, hti_thom2_rhop'
```

and

$$K_{\alpha_v}^{\text{vtivel}} = K_{\alpha_v}^{\text{vtith}} - \frac{2\alpha_h^2}{\alpha_h^2 - \alpha_v^2} K_{\varepsilon}^{\text{vtith}} - \frac{2\alpha_n^2}{\alpha_n^2 - \alpha_v^2} K_{\delta}^{\text{vtith}}, \tag{12a}$$

$$K_{\beta_v}^{\text{vtivel}} = K_{\beta_v}^{\text{vtith}}, \tag{12b}$$

$$K_{\alpha_h}^{\text{vtivel}} = \frac{2\alpha_h^2}{\alpha_h^2 - \alpha_v^2} K_{\varepsilon}^{\text{vtith}}, \tag{12c}$$

$$K_{\alpha_n}^{\text{vtivel}} = \frac{2\alpha_n^2}{\alpha_n^2 - \alpha_v^2} K_{\delta}^{\text{vtith}}, \tag{12d}$$

$$K_{\rho}^{\text{vtivel}} = K_{\rho}^{\text{vtith}}. \tag{12e}$$

```
kernel_list='hti_vel_alpha_kernel, hti_vel_beta_kernel, hti_vel_alphah_kernel,
hti_vel_alphan_kernel, hti_vel_rhop_kernel'
precond_list='pdh_hti_vel_alpha, pdh_hti_vel_beta, pdh_hti_vel_alphah,
pdh_hti_vel_alphan, pdh_hti_vel_rhop'
model_list='hti_vel_vp, hti_vel_vs, hti_vel_vph,
hti_vel_vpn, hti_vel_rhop'
```

The sensitivity kernels in TTIEC model parameterization of TTI-elastic FWI are given by

$$K_{c_{11}}^{\text{ttiec}} = -c_{11} \partial_x u_x^{\dagger} \partial_x u_x, \tag{13a}$$

$$K_{c_{13}}^{\text{ttiec}} = -c_{13} \left(\partial_x u_x^{\dagger} \partial_z u_z + \partial_z u_z^{\dagger} \partial_x u_x \right), \tag{13b}$$

$$K_{c_{15}}^{\text{ttiec}} = -c_{15} \left(2\partial_x u_x^{\dagger} \partial_x u_z + 2\partial_x u_z^{\dagger} \partial_x u_x \right), \tag{13c}$$

$$K_{c_{33}}^{\text{ttiec}} = -c_{33} \partial_z u_z^{\dagger} \partial_z u_z, \tag{13d}$$

$$K_{c_{35}}^{\text{ttiec}} = -c_{15} \left(2\partial_z u_z^{\dagger} \partial_x u_z + 2\partial_x u_z^{\dagger} \partial_z u_z \right), \tag{13e}$$

$$K_{c_{55}}^{\text{ttiec}} = -4c_{55} \partial_x u_z^{\dagger} \partial_x u_z, \tag{13f}$$

$$K_{\rho}^{\text{ttiec}} = K_{\rho}^{\text{dm}}. \tag{13g}$$

```
kernel_list='tti_ec_c11_kernel, tti_ec_c13_kernel, tti_ec_c15_kernel,
tti_ec_c33_kernel, tti_ec_c35_kernel, tti_ec_c55_kernel, tti_ec_rho_kernel'
precond_list='pdh_tti_ec_c11, pdh_tti_ec_c13, pdh_tti_ec_c15,
pdh_tti_ec_c33, pdh_tti_ec_c35, pdh_tti_ec_c55, pdh_tti_ec_rho'
model_list='c11, c13, c15, c33, c35, c55, rho'
```

TTIECU model parameterization:

$$
\begin{aligned}
K_{c_{11}^{\mathrm{u}}}^{\mathrm{ttiecu}} =& c_{11}^{\mathrm{u}} \left(\frac{m_1 K_{c_{11}}^{\mathrm{ttiec}}}{8c_{11}} + \frac{m_2 K_{c_{13}}^{\mathrm{ttiec}}}{8c_{13}} \right. \\
& \left. + \frac{m_5 K_{c_{15}}^{\mathrm{ttiec}}}{4c_{15}} + \frac{m_3 K_{c_{33}}^{\mathrm{ttiec}}}{8c_{33}} + \frac{(m_5 - 2m_4) K_{c_{35}}^{\mathrm{ttiec}}}{4c_{35}} + \frac{m_2 K_{c_{55}}^{\mathrm{ttiec}}}{8c_{55}} \right), \qquad \text{(14a)}
\end{aligned}
$$

$$
\begin{aligned}
K_{c_{13}^{\mathrm{u}}}^{\mathrm{ttiecu}} =& c_{13}^{\mathrm{u}} \left(\frac{m_2 K_{c_{11}}^{\mathrm{ttiec}}}{4c_{11}} + \frac{m_6 K_{c_{13}}^{\mathrm{ttiec}}}{8c_{13}} - \frac{m_4 K_{c_{15}}^{\mathrm{ttiec}}}{2c_{15}} \right. \\
& \left. + \frac{m_2 K_{c_{33}}^{\mathrm{ttiec}}}{4c_{33}} + \frac{m_4 K_{c_{35}}^{\mathrm{ttiec}}}{2c_{35}} - \frac{m_2 K_{c_{55}}^{\mathrm{ttiec}}}{4c_{55}} \right), \qquad \text{(14b)}
\end{aligned}
$$

$$
\begin{aligned}
K_{c_{33}^{\mathrm{u}}}^{\mathrm{ttiecu}} =& c_{33}^{\mathrm{u}} \left(\frac{m_3 K_{c_{11}}^{\mathrm{ttiec}}}{8c_{11}} + \frac{m_2 K_{c_{13}}^{\mathrm{ttiec}}}{8c_{13}} - \frac{m_5 K_{c_{15}}^{\mathrm{ttiec}}}{4c_{15}} \right. \\
& \left. + \frac{m_1 K_{c_{33}}^{\mathrm{ttiec}}}{8c_{33}} + \frac{m_4 K_{c_{35}}^{\mathrm{ttiec}}}{2c_{35}} - \frac{m_5 K_{c_{35}}^{\mathrm{ttiec}}}{4c_{35}} + \frac{m_2 K_{c_{55}}^{\mathrm{ttiec}}}{8c_{55}} \right), \qquad \text{(14c)}
\end{aligned}
$$

$$
\begin{aligned}
K_{c_{55}^{\mathrm{u}}}^{\mathrm{ttiecu}} =& c_{55}^{\mathrm{u}} \left(\frac{m_2 K_{c_{11}}^{\mathrm{ttiec}}}{2c_{11}} - \frac{m_2 K_{c_{13}}^{\mathrm{ttiec}}}{2c_{13}} - \frac{m_4 K_{c_{15}}^{\mathrm{ttiec}}}{c_{15}} \right. \\
& \left. + \frac{m_2 K_{c_{33}}^{\mathrm{ttiec}}}{2c_{33}} + \frac{m_4 K_{c_{35}}^{\mathrm{ttiec}}}{c_{35}} + \frac{K_{c_{55}}^{\mathrm{ttiec}}}{c_{55}} - \frac{m_2 K_{c_{55}}^{\mathrm{ttiec}}}{2c_{55}} \right), \qquad \text{(14d)}
\end{aligned}
$$

$$
\begin{aligned}
K_{\theta}^{\mathrm{ttiecu}} =& \theta \left[- \frac{(2p_2 \sin 2\theta + p_1 \sin 4\theta) K_{c_{11}}^{\mathrm{ttiec}}}{2c_{11}} + \frac{p_1 \sin 4\theta K_{c_{13}}^{\mathrm{ttiec}}}{2c_{13}} \right. \\
& + \frac{(2p_2 \cos 2\theta + p_1 \cos 4\theta) K_{c_{15}^{\mathrm{ttiec}}}}{2c_{15}} + \frac{(2p_2 \sin 2\theta - p_1 \sin 4\theta) K_{c_{33}^{\mathrm{ttiec}}}}{2c_{33}} \qquad \text{(14e)} \\
& \left. + \frac{(2p_2 \cos 2\theta - p_1 \cos 4\theta) K_{c_{35}}^{\mathrm{ttiec}}}{2c_{35}} + \frac{p_1 \sin 4\theta K_{c_{55}}^{\mathrm{ttiec}}}{2c_{55}} \right],
\end{aligned}
$$

$$
K_{\rho}^{\mathrm{ttiecu}} = K_{\rho}^{\mathrm{ttiec}}, \qquad \text{(14f)}
$$

where the coefficients m_1, m_2, m_3, m_4, m_5, m_6, p_1 and p_2 are:

$$m_1 = 3 + 4\cos 2\theta + \cos 4\theta, \qquad \text{(15a)}$$

$$m_2 = 1 - \cos 4\theta, \qquad \text{(15b)}$$

$$m_3 = 3 - 4\cos 2\theta + \cos 4\theta, \qquad \text{(15c)}$$

$$m_4 = \sin 4\theta, \qquad \text{(15d)}$$

$$m_5 = \sin 2\theta + \sin 4\theta, \qquad \text{(15e)}$$

$$m_6 = 6 + 2\cos 4\theta \qquad \text{(15f)}$$

$$p_1 = c_{11}^{\mathrm{u}} - 2c_{13}^{\mathrm{u}} + c_{33}^{\mathrm{u}} - 4c_{55}^{\mathrm{u}}, \qquad \text{(15g)}$$

$$p_2 = c_{11}^{\mathrm{u}} - c_{13}^{\mathrm{u}}, \qquad \text{(15h)}$$

```
kernel_list='tti_ecu_c11_kernel, tti_ecu_c13_kernel, tti_ecu_c33_kernel,
tti_ecu_c55_kernel, tti_ecu_theta_kernel, tti_ecu_rho_kernel'
precond_list='pdh_tti_ecu_c11, pdh_tti_ecu_c13, pdh_tti_ecu_c33,
pdh_tti_ecu_c55, pdh_tti_ecu_theta, pdh_tti_ecu_rho'
model_list='c11, c13, c33, c55, theta, rho'
```

TTITHOM model parameterization:

$$K_{\alpha_v}^{\text{ttith}} = 2K_{c_{33}}^{\text{ttiecu}} + 2K_{c_{11}}^{\text{ttiecu}} + \frac{2\alpha_v^4\left(2\delta+1\right) - 2\alpha_v^2\beta_v^2\left(\delta+1\right)}{M\left(M-\beta_v^2\right)}K_{c_{13}}^{\text{ttiecu}}, \tag{16a}$$

$$K_{\beta_v}^{\text{ttith}} = 2K_{c_{44}}^{\text{ttiecu}} - \frac{2\beta_v^2\left(\alpha_v^2\delta - \beta_v^2 + \alpha_v^2\right)}{M\left(M-\beta_v^2\right)}K_{c_{13}}^{\text{ttiecu}} - \frac{2\beta_v^2}{M-\beta_v^2}K_{c_{13}}^{\text{ttiecu}}, \tag{16b}$$

$$K_{\varepsilon}^{\text{ttith}} = \frac{2\varepsilon}{2\varepsilon+1}K_{c_{11}}^{\text{ttiecu}}, \tag{16c}$$

$$K_{\delta}^{\text{ttith}} = \frac{\delta\alpha_v^2\left(\alpha_v^2-\beta_v^2\right)}{M\left(M-\beta_v^2\right)}K_{c_{13}}^{\text{ttiecu}}, \tag{16d}$$

$$K_{\theta}^{\text{ttith}} = K_{\theta}^{\text{ttiecu}}, \tag{16e}$$

$$K_{\rho}^{\text{ttith}} = K_{\rho}^{\text{ttiecu}} + \alpha_v^2K_{c_{33}}^{\text{ttiecu}} + \beta_v^2K_{c_{55}}^{\text{ttiecu}} + \left(M-\beta_v^2\right)K_{c_{13}}^{\text{ttiecu}}. \tag{16f}$$

```
kernel_list=’tti_thom_alpha_kernel, tti_thom_beta_kernel, tti_thom_epsilon_kernel,
tti_thom_delta_kernel, tti_thom_theta_kernel, tti_thom_rho_kernel’
precond_list=’pdh_tti_thom_alpha, pdh_tti_thom_beta, pdh_tti_thom_epsilon,
pdh_tti_thom_delta, pdh_tti_thom_theta, pdh_tti_thom_rhop’
model_list=’tti_thom_vp, tti_thom_vp, tti_thom_vp, tti_thom_vp, tti_thom_theta, tti_thom_rhop’
```

B Misfit functions and adjoint sources

In this Appendix, we summarize the misfit functions and the corresponding adjoint sources provided by SeisElastic2D.

The common waveform-difference (WD) misfit function and its adjoint source are given by

$$\chi^{\text{WD}} = \frac{1}{2}\left[u_i - u_i^{\text{obs}}\right]^2, \tag{17}$$

and

$$\tilde{f}_{i,\text{WD}}^{\dagger} = u_i - u_i^{\text{obs}}, \tag{18}$$

where u_i and u_i^{obs} are the ith component synthetic and observed displacement fields.

For cross-correlation (CC) based wave-equation traveltime inversion, the misfit function minimizes the traveltime difference:

$$\chi^{\text{CC}} = \frac{1}{2}\sum_i\left[\Delta t_i\right]^2, \tag{19}$$

where the traveltime difference Δt_i is obtained by cross-correlation:

$$\Delta t_i = \arg\max_{\tau}\int_0^{t'} u_i\left(t+\tau\right)u_i^{\text{obs}}\left(t\right)dt. \tag{20}$$

and the adjoint source is

$$\tilde{f}_{i,\text{CC}}^{\dagger} = \frac{\Delta t_i\partial_t u_i\left(-t\right)}{M_i}, \tag{21}$$

where M_i in the denominator is

$$M_i = \int_0^{t'} \partial_t^2 u_i\left(t\right) u_i\left(t\right) dt. \tag{22}$$

In double-difference (DD) adjoint tomography, the misfit function measures the differential traveltime difference:

$$\chi^{\mathrm{DD}} = \frac{1}{2} \sum_i \sum_{j>i} \left[\Delta t_{ij} - \Delta t_{ij}^{\mathrm{obs}}\right]^2, \tag{23}$$

where the Δt_{ij} is obtained by

$$\Delta t_{ij} = \arg\max_{\tau} \int_0^{t'} u_i\left(t+\tau\right) u_j^{\mathrm{obs}}\left(t\right) dt. \tag{24}$$

and the adjoint source becomes:

$$\begin{aligned} f_{i,\mathrm{DD}}^{\dagger} &= +\sum_{j>i} \frac{\Delta\Delta t_{ij} \partial_t u_j(-[t-\Delta t_{ij}])}{N_{ij}}, \\ f_{j,\mathrm{DD}}^{\dagger} &= -\sum_{i<j} \frac{\Delta\Delta t_{ij} \partial_t u_i(-[t+\Delta t_{ij}])}{N_{ij}}, \end{aligned} \tag{25}$$

where N_{ij} is the denominator is

$$N_{ij} = \int_0^{t'} \partial_t^2 u_i(t+\Delta t_{ij}) u_j(t) dt, \tag{26}$$

and $\Delta\Delta t_{ij}$ is the differential traveltime difference.

Envelope measures the instantaneous amplitude of seismic signal. We design envelope-difference (ED) and -ratio (ER) misfit functions:

$$\chi^{\mathrm{ED}} = \frac{1}{2} \left[E_i - E_i^{\mathrm{obs}}\right]^2, \tag{27}$$

$$\chi^{\mathrm{ER}} = \frac{1}{2} \left[\ln \frac{E_i^{\mathrm{obs}}}{E_i}\right]^2, \tag{28}$$

where E_i is the envelope of the seismic data

$$E_i = \sqrt{u_i^2 + \mathcal{H}^2\left[u_i\right]}, \tag{29}$$

where ln is the natural logarithm. $\mathcal{H}$ indicates Hilbert transform. The corresponding adjoint sources are

$$\tilde{f}_{i,\mathrm{ED}}^{\dagger} = \frac{E_i - E_i^{\mathrm{obs}}}{E_i} u_i - \mathcal{H}\left[\frac{E_i - E_i^{\mathrm{obs}}}{E_i} \mathcal{H}\left[u_i\right]\right], \tag{30}$$

$$\tilde{f}_{i,\mathrm{ER}}^{\dagger} = -\ln\left[\frac{E_i^{\mathrm{obs}}}{E_i}\right]\frac{u_i}{E_i^2} - \mathcal{H}\left\{\ln\left[\frac{E_i^{\mathrm{obs}}}{E_i}\right]\frac{\mathcal{H}\left[u_i\right]}{E_i^2}\right\}. \tag{31}$$

Instantaneous phase (IP) misfit functions is give by

$$\chi^{\mathrm{IP}} = \frac{1}{2}\left[\phi_i^{\mathrm{obs}} - \phi_i\right]^2, \tag{32}$$

where ϕ_i indicates the instantaneous phase of the seismic data:

$$\phi_i = \arctan\frac{u_i}{\mathcal{H}\left[u_i\right]}, \tag{33}$$

and the adjoint source is

$$\tilde{f}_{i,\mathrm{IP}}^{\dagger} = -\Delta\phi_i\frac{\mathcal{H}\left[u_i\right]}{E_i^2} - \mathcal{H}\left[\Delta\phi_i\frac{u_i}{E_i^2}\right], \tag{34}$$

Root mean square (RMS) amplitude measures the energy variation of the seismic data within a time period. We developed RMS amplitude-difference (RD) and -ratio (RR) misfit functions for attenuation estimation in visco-elastic FWI. The RD and RR misfit functions are given by

$$\chi^{\mathrm{RD}} = \frac{1}{2}\left[A_i^{\mathrm{obs}} - A_i\right]^2, \tag{35}$$

$$\chi^{\mathrm{RR}} = \frac{1}{2}\left[\ln\frac{A_i^{\mathrm{obs}}}{A_i}\right]^2, \tag{36}$$

where A_i indicates the RMS amplitude of the seismic data within time window $\mathrm{w}(t)$:

$$A_i = \sqrt{\int_0^{t'} \mathrm{w}\left(t\right) u_i^2\left(t\right) dt}. \tag{37}$$

The adjoint sources for RD and RR misfit functions are given by

$$\tilde{f}_{i,\mathrm{RD}}^{\dagger} = -\left[A_i^{\mathrm{obs}} - A_i\right]\frac{\mathrm{w}u_i}{A_i}, \tag{38}$$

$$\tilde{f}_{i,\mathrm{RR}}^{\dagger} = -\ln\left[\frac{A_i^{\mathrm{obs}}}{A_i}\right]\frac{\mathrm{w}u_i}{A_i^2}. \tag{39}$$

In frequency domain, phase information is naturally separated from the amplitude information. We develop spectral amplitude-difference (SD) and -ratio (SR) misfit functions for attenuation estimation in visco-elastic FWI. The SD and SR misfit functions are given by

$$\chi^{\mathrm{SD}} = \frac{1}{2}\left[\mathcal{A}_i^{\mathrm{obs}} - \mathcal{A}_i\right]^2, \tag{40}$$

$$\chi^{\mathrm{SR}} = \frac{1}{2}\left[\ln \frac{\mathcal{A}_i^{\mathrm{obs}}}{\mathcal{A}_i}\right]^2, \tag{41}$$

where $\mathcal{A}_i$ is amplitude spectrum of the data:

$$\mathcal{A}_i = |\tilde{u}_i(\omega)|. \tag{42}$$

The adjoint sources in SD and SR misfit functions are

$$\tilde{f}_{i,\mathrm{SD}}^{\dagger} = -\left[\mathcal{A}_i^{\mathrm{obs}} - \mathcal{A}_i\right] \frac{\tilde{\mathrm{w}}\mathcal{R}\left[\tilde{u}_i\right]}{\mathcal{A}_i}, \tag{43}$$

$$\tilde{f}_{i,\mathrm{SR}}^{\dagger} = -\ln\left[\frac{\mathcal{A}_i^{\mathrm{obs}}}{\mathcal{A}_i}\right] \frac{\tilde{\mathrm{w}}\mathcal{R}\left[\tilde{u}_i\right]}{\mathcal{A}_i^2}. \tag{44}$$

where $\mathcal{R}$ means real part.

Attenuation results in central-frequency shift of the seismic data. Thus, we also develop an central-frequency shift (CS) misfit function for attenuation estimation in visco-elastic FWI. The CS misfit function is given by

$$\chi^{\mathrm{CS}} = \frac{1}{2}\left[\Delta\tilde{\omega}_i\right]^2, \tag{45}$$

where $\Delta\tilde{w}_i$ indicates the central-frequency difference obtained by cross-correlation in frequency domain

$$\Delta\tilde{w}_i = \int_0^{\omega'} \mathcal{A}_i^{\mathrm{obs}}(\omega + \Delta\omega)\,\mathcal{A}_i(\omega)\,d\omega. \tag{46}$$

The adjoint source is

$$\tilde{f}_{i,\mathrm{CS}}^{\dagger} = \Delta\tilde{\omega}_i \frac{\partial_{\Delta\omega}\mathcal{A}_i^{\mathrm{obs}}\mathcal{R}\left[\tilde{u}_i\right]}{M_\omega \mathcal{A}_i}, \tag{47}$$

where M_ω in the denominator is

$$M_\omega = \int_0^{\omega'} \partial_{\Delta\omega}^2 \mathcal{A}_i^{\mathrm{obs}}(\omega)\,\mathcal{A}_i(\omega)\,d\omega. \tag{48}$$