CGFD2D-wave
User Manual

October 26, 2021

Contents

Chapter 1

How to give media parameters and run wave simulations for different media types

par.json

```
"medium" : {
    "type"   : "elastic_aniso",
    "#code_generate" : 1,
    "infile_layer" : "/home/usr1/testCGFD2D/prep_medium/basin.mdlay",
    "#infile_grid" : "/home/usr1/testCGFD2D/prep_medium/basin.mdgrd",
    "#import" : "/home/usr1/testCGFD2D/output/media",
    "equivalent_medium_method" : "tti"
},
"is_export_media" : 1,
"media_export_dir" : "/home/usr1/testCGFD2D/output/media",

"#visco_config" : {
    "type" : "graves_Qs",
    "Qs_freq" : 1.0
}
```

- `type` is the flag of the solver, that is, what media do you want to simulate, and four options are supported in this code:
  - `acoustic_iso`: choose this to call the solver of acoustic wave equation of isotropic media.
  - `elastic_iso`: choose this to call the solver of the elastic wave equation of isotropic media.
  - `elastic_vti`: choose this to call the solver of the elastic wave equation of vertical transversely isotropic media.
  - `elastic_aniso`: choose this to call the solver of the elastic wave equation of anisotropic media.

  The media type is given in `import`, `code_generate`, `infile_layer` or `infile_grid`.

- The media can be given in four ways:
  - `code_generate`: Choose this flag if you want to give the media by code. You should edit **forward/md_t.c** file and recompile the code.
  - `infile_layer`: If you want to give media by the intereface line, this option should be helpful. The media parameters are given by the **mdlay** file, and the file format is shown in Section 1.1
  - `infile_grid`: If you want to give media by a given grid media, this option should be helpful. The media parameters are given by the **mdgrd** file, the file format is shown in Section 1.2.

2

- **import**: If you already generated media from the above three options, and do not want to re-generate media, select this option and give the folder of the exported media.

- `equivalent_medium_method`: If you give media by `infile_layer` or `infile_grid`, equivalent medium parameterization methods can be applied. For different media, we provide different equivalent medium parameterization methods, please see Section 1.3 for detail.

- `is_export_media` should be **1** if you want to export the media, and a **media_export_dir** should be given.

- The `visco_config` is selected if the media is viscoelastic, the visco_type and Qs frequency are needed.

1.1 Format of .mdlay

If you want to get the model from the interface line, you can select the option `infile_layer`. And the media are given by a **mdlay** file. The file format is:

- The first line is the media type, it can be
  `one_component`,
  `acoustic_isotropic`,
  `elastic_isotropic`,
  `elastic_vti_prem`,
  `elastic_vti_thomsen`,
  `elastic_vti_cij`,
  `elastic_tti_thomsen`,
  `elastic_tti_bond`,
  `elastic_aniso_cij`.

- The second line is the number of interface (`NI`).

- After then, for each interface below, give the number of points (`npoints`) and then x, z, and media parameters for each point. Take the `elastic_isotropic` media for example, the media parameters are read as:

```
for (ni=0; ni<NI; ni++)
  for (ip=0; ip<npoints; ip++)
     fscanf(layer_file, "%f %f %f %f %f %f %f %f %f %f %f",
            &x_interface, &z_interface,
            &rho, &rho_grad, &rho_pow,
            &vp, &vp_grad, &vp_pow,
            &vs, &vs_grad, &vs_pow);
```

Note that the x here need to be increasing, the max range of interfaces need to be lager than the calculation area in x direction. `*_grad` and `*_pow` is the gradient and power of the media parameters in z-direction. The media parameters below interface are calculated by

$$var_{grid\ point} = var + \left(z\_\text{interface} - z_{grid\ point}\right)^{var\_pow} * var\_grad.$$

For the points higher than the interface, assigned by the uppermost medium. The media parameters in x-direction are calculated by linear interpolation.

For different media type, different media parameters are given,

- `one_component`: var, var_grad, var_pow.

- `acoustic_isotropic`: ρ, ρ_grad, ρ_pow, vp, vp_grad, vp_pow.

- `elastic_isotropic`: ρ, ρ_grad, ρ_pow, vp, vp_grad, vp_pow, vs, vs_grad, vs_pow.

- `elastic_vti_prem`: ρ, ρ_grad, ρ_pow, vph, vph_grad, vph_pow, vpv, vpv_grad, vpv_pow, vsv, vsv_grad, vsv_pow, η, η_grad, η_pow. The meaning of the media parameters can be found from [Dziewonski and Anderson, 1981].

- `elastic_vti_thomsen`: ρ, ρ_grad, ρ_pow, α_0, α_0_grad, α_0_pow, β_0, β_0_grad, β_0_pow, ε, ε_grad, ε_pow, δ, δ_grad, δ_pow. The meaning of the media parameters can be found from [Thomsen, 1986].

- elastic_vti_cij: ρ, ρ_grad, ρ_pow, c_{11}, c_{11}_grad, c_{11}_pow, c_{33}, c_{33}_grad, c_{33}_pow, c_{55}, c_{55}_grad, c_{55}_pow, c_{13}, c_{13}_grad, c_{13}_pow.

- elastic_tti_thomsen: ρ, ρ_grad, ρ_pow, α_0, α_0_grad, α_0_pow, β_0, β_0_grad, β_0_pow, ε, ε_grad, ε_pow, δ, δ_grad, δ_pow, θ, θ_grad, θ_pow. The meaning of the media parameters can be found from [Thomsen, 1986].

- elastic_tti_bond: ρ, ρ_grad, ρ_pow, c_{11}, c_{11}_grad, c_{11}_pow, c_{33}, c_{33}_grad, c_{33}_pow, c_{55}, c_{55}_grad, c_{55}_pow, c_{13}, c_{13}_grad, c_{13}_pow, θ, θ_grad, θ_pow. θ is the dip angle.

- elastic_aniso_cij: ρ, ρ_grad, ρ_pow, c_{11}, c_{11}_grad, c_{11}_pow, c_{13}, c_{13}_grad, c_{13}_pow, c_{15}, c_{15}_grad, c_{15}_pow, c_{33}, c_{33}_grad, c_{33}_pow, c_{35}, c_{35}_grad, c_{35}_pow, c_{55}, c_{55}_grad, c_{55}_pow.

You can also find examples in the **example/prep_medium** directory.

1.2   Format of .mdgrd

The option `infile_grid` is useful if you want to give the media by a grid model. If `infile_grid` is selected, the media should be given by a **.mdgrd** file. The file format is:

- The first line is the media type, it can be
  ```
  one_component,
  acoustic_isotropic,
  elastic_isotropic,
  elastic_vti_prem,
  elastic_vti_thomsen,
  elastic_vti_cij,
  elastic_tti_thomsen,
  elastic_tti_bond,
  elastic_aniso_cij.
  ```

- The second line is the number of layer (`NL`), if `NL` > 1, there is a designated interface, and the equivalent medium parameterization methods can be applied across this interfaces.

- The next `NL` lines are the number of grids in the z-direction of each layer

- Next, the information of the interface mesh is given:
  `NX   MIN_X   SPACING_X`
  `NX` is the number of points along *x* direction;
  `MIN_X` is the minimal *x* coordinates;
  `SPACING_X` is spacing between points along *x*.

- After then, the elevation, media parameters are given in every grid points. Take the `elastic_isotropic` media for example, the media parameters are read as:

  ```
  for (nl=0; nl< NL; nl++)
    for (ip=0; ip<np[nl]; np++)
      for (ix=0; ix<NX; ix++)
        fscanf(in_file_grid, "%f %f %f %f",
               &elevation, &rho, &vp, &vs);
  ```

  The media parameters are calculated by interpolation of the values at the given grid points.

  For the points higher than the interface, assigned by the uppermost medium.

  For different media type, different media parameters are given,

  - one_component: var.
  - acoustic_isotropic: ρ, vp.
  - elastic_isotropic: ρ, vp, vs.
  - elastic_vti_prem: ρ, vph, vpv, vsv, η. Please see [Dziewonski and Anderson, 1981] for detail.
  - elastic_vti_thomsen: ρ, α_0, β_0, ε, δ. Please see [Thomsen, 1986] for detail.

    – `elastic_vti_cij`: ρ, c_{11}, c_{33}, c_{55}, c_{13}.

    – `elastic_tti_thomsen`: ρ, α_0, β_0, ε, δ, θ. Please see [Thomsen, 1986] for detail.

    – `elastic_tti_bond`: ρ, c_{11}, c_{33}, c_{55}, c_{13}, θ. θ is the dip angle.

    – `elastic_aniso_cij`: ρ, c_{11}, c_{13}, c_{15}, c_{33}, c_{35}, c_{55}.

If `NL` > 1, there is a designated interface; and the elevation of ng[il]+1 needs to be the same as ng[il]. The equivalent medium parameterization method can be applied on the points cutting by this interface.

You can also find the example in the **example/prep_medium** directory.

1.3   Note about equivalent medium parameterization methods

When there are strong interfaces in the model, using grid points to discrete models directly may cause interface error and the generation of artificial diffraction from stair step interfaces. In this code, we provide some equivalent medium parametrization methods to reduce this interface error.

If you give the model by `infile_layer` or `infile_grid`, you can apply different equivalent medium parametrization methods by give different `equivalent_medium_method`.
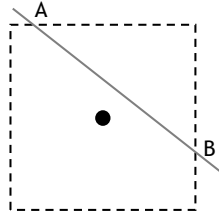

Figure 1.1: Illustration of an interface passing through an FD cell. The black circle represents the grid point, the dashed-line square represents a finite0-difference mesh centred on the black circle. The equivalent medium parameterization methods are act on such grid points passed through by the interfaces.

If the media type in the **mdgrd** or **mdlay** file is `one_component`, `equivalent_medium_method` can be

• `loc`: using local value to discrete model.

• `ari`: volume integral arithmetic average,

$$\langle var \rangle = \frac{1}{\Delta V} \int_{iz-1/2}^{iz+1/2} \int_{ix-1/2}^{ix+1/2} var\, dxdz \tag{1.1}$$

• `har`: volume integral harmonic average.

$$\langle var \rangle^H = \frac{\Delta V}{\int_{iz-1/2}^{iz+1/2} \int_{ix-1/2}^{ix+1/2} \frac{1}{var} dxdz} \tag{1.2}$$

If the media type is `acoustic_isotropic`, `equivalent_medium_method` can be

• `loc`: using local media parameters to discrete model.

• `har`: applying harmonic average to κ, and applying arithmetic average to density ρ.

• `ari`: applying arithmetic average to κ and ρ.

If the media type is `elastic_isotropic`, `equivalent_medium_method` can be

• `loc`: using local media parameters to discrete model.

- `har`: applying harmonic average to elastic modulus, and applying arithmetic average to density ρ. See Moczo et al. [2002] and [Moczo et al., 2014] for detail.

- `ari`: applying arithmetic average to elastic modulus and density.

- `tti`: applying TTI representation to elastic modulus and arithmetic average to density. See [Jiang and Zhang, 2021] for detail.

    If the media type is `elastic_vti_*`, `equivalent_medium_method` can be

- `loc`: using local media parameters to discrete model.

- `har`: applying harmonic average to elastic modulus c_{ij}, and applying arithmetic average to density ρ.

- `ari`: applying arithmetic average to elastic modulus c_{ij} and density.

- `tti`: applying TTI representation to elastic modulus and arithmetic average to density.

    If the media type is `elastic_aniso_cij` or `elastic_tti_*`, `equivalent_medium_method` can be

- `loc`: using local media parameters to discrete model.

- `har`: applying harmonic average to elastic modulus c_{ij}, and applying arithmetic average to density ρ.

- `ari`: applying arithmetic average to elastic modulus c_{ij} and density.

Copyright

Main historical authors:

© October 2021

<u>Evolution of the code:</u>

Bibliography

A. M. Dziewonski and D. L. Anderson. Preliminary reference earth model. *Physics of the earth and planetary interiors*, 25(4):297–356, 1981.

L. Jiang and W. Zhang. Tti equivalent medium parametrization method for the seismic waveform modelling of heterogeneous media with coarse grids. *Geophysical Journal International*, 227(3):2016–2043, 2021.

P. Moczo, J. Kristek, V. Vavrycuk, R. J. Archuleta, and L. Halada. 3D heterogeneous staggered-grid finite-difference modeling of seismic motion with volume harmonic and arithmetic averaging of elastic moduli and densities. *Bulletin of the Seismological Society of America*, 92(8):3042–3066, 2002.

P. Moczo, J. Kristek, and M. Gãąlis. *The finite-difference modelling of earthquake motions: waves and ruptures*. Cambridge University Press, New York, 2014. ISBN 978-1-107-02881-4.

L. Thomsen. Weak elastic anisotropy. *Geophysics*, 51(10):1954–1966, 1986.