INSTRUCTIONS FOR RUNNING FD\_seis\_sim

# Calling the function

The FD\_seis\_sim algorithm can be called as a simple MATLAB function, using as input the model parameters and optionally the topography and the attenuation models. To use the algorithm use the following command:

FD\_seis\_sim(Vp,Vs,rho,X,Y,Z);

* Vp: 3d matrix with the P-wave velocity values
* Vs: 3d matrix with the S-wave velocity values
* rho: 3d matrix with the density values
* X: The X node locations
* Y: The Y node locations
* Z: The Z node locations (negative depth)

Care must be taken with the input models to ensure that the units are uniform: e.g. Vp in km/sec and X, Y, Z in kms. To use the algorithm with a topographic surface, use the following command:

FD\_seis\_sim(Vp,Vs,rho,X,Y,Z,topo\_data,X\_topo,Y\_topo);

* topo\_data: 2d matrix with the elevation values
* X\_topo: X node locations of the topographic grid
* Y\_topo: X node locations of the topographic grid

Once again units must be uniform. It is also possible to use the program with a 3d attenuation model. In this case the following commands can be used:

FD\_seis\_sim(Vp,Vs,rho,X,Y,Z,topo\_data,X\_topo,Y\_topo,Q0); FD\_seis\_sim(Vp,Vs,rho,X,Y,Z,topo\_data,X\_topo,Y\_topo,Q0,f0);

* Q0: The Q quality factor (see Graves, 1996)
* f0 (optional): The central attenation frequency (see Graves, 1996)

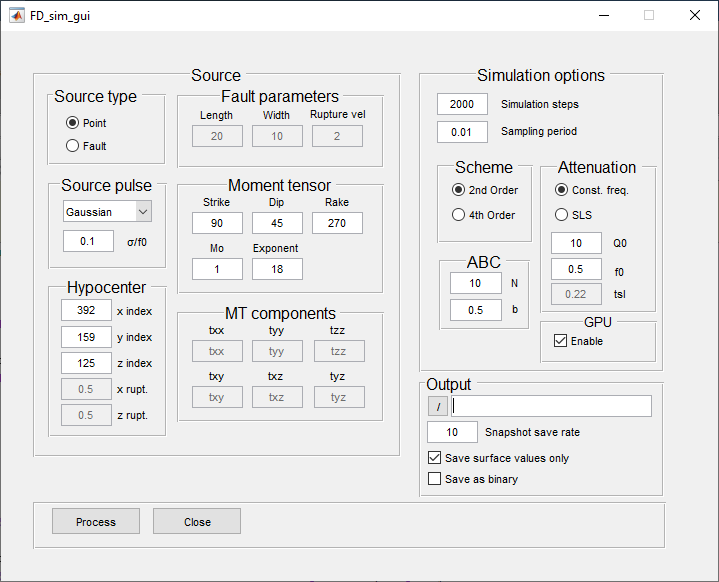
FD\_seis\_sim(Vp,Vs,rho,X,Y,Z,topo\_data,X\_topo,Y\_topo,TP,TS,tsl);

* TP (optional): P relaxation times (see Bohlen, 2002)
* TS (optional): S relaxation times (see Bohlen, 2002)
* tsl (optional): stress relaxation times (see Bohlen, 2002)

The first case represents frequency independent attenuation and does not separate P-wave from S-wave attenuation. A 3d f0 matrix is optional. In the second case, attenuation is frequency dependent. If not all 3 parameters (TP, TS or tsl) are 3D matrices, the rest are required as scalar values.

# Main GUI with simulation parameters

Once the FD\_seis\_sim function is called using any of the above input parameter combination, A GUI is then opened, where the rest of the simulation parameters can be selected. The following figure shows an example of the GUI:



The following options are available:

### Source main panel

* *Source type panel*: selection of source type (Point/Fault).
* *Fault Parameters pane*l (if fault source type is selected): set the fault length, width and rupture velocity.
* *Source pulse panel*: selection of source pulse type (Gaussian/Gaussian 1st derivative/ Ricker) and setting of σ value of the Gaussian pulses or the central period of the Ricker wavelet.
* *Moment tensor panel*: Set the strike, dip, rake, scalar moment and scaling factor of the source.
* *Hypocenter panel*: Set the x, y and z node **indices** of the source. If a fault source is selected, then set the x, y, z node indices of the upper left corner of the fault surface, as well as the rupture initiation point (z rupt., z\_rupt.) within the fault, as a ratio of its total length and width (e.g. 0.5 is the middle of the fault).

If a fault source is selected, the **fin\_fault** and **fault\_shift** functions are automatically called before the main simulation part in order to obtain a series of nodes within the 3D grid that better represent the desired fault surface according to the fault length, width, strike and dip. Each node is a sub-source that is activated with a time delay relative to the initial sub-source (set by the x\_rupt. And z\_rupt values) according to the rupture velocity.

### Simulation options main panel

* *Simulation steps:* Total number of simulation time steps.
* *Sampling period:* Simulation sampling period in seconds. The value is directly related to the grid velocity values and the node spacing. Too large values cause unstable results (inf amplitude values).
* *Scheme Panel:* Selection of the spatial derivative scheme (2nd or 4th order).
* *Attenuation Panel:* If no 3d grids are given for any of the parameters that are related to the anelastic attenuation (Q0/f0 or TP/TS/tsl), then the attenuation scheme and the related values can be given here.
* *ABC (Absorbing Boundary Condition) panel:* Set the number of nodes that act as a transition zone and the b value (typically between 0.3-0.5). These nodes should be excluded from the results!
* *GPU panel:* Enable or disable GPU processing.

### Output main panel

* /: Set the output path where the resulting wavefield snapshots for the 3 components will be saved.
* *Snapshot save rate:* Set after how many simulation steps a wavefield snapshot will be saved.
* *Surface values only:* If selected, only the values of the simulated wavefield in the surface of the 3D volume will be saved (greatly reduces the size of the saved data).
* *Save as binary:* If selected, the results will be saved in binary format.

# Examples

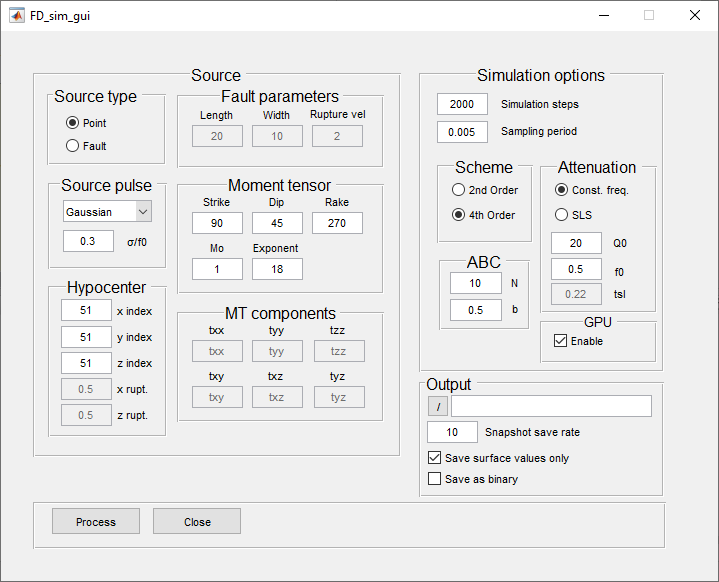
## Example 1: Double couple point source with layered 3D model and flat surface.

For the first example, we used a layered 3D model with a flat top surface. The input data for this case can be loaded into MATLAB workspace from the file example1\_input.mat. The input Vp model used has a velocity of 4 km/sec up to the depth of 2 km, 5.2 km/s up to 6 kms of depth and 8 km/s up to 10 kms. The Vs model was derived from the Vp model using the formula: Vs=Vp./1.78. The density model (rho) was also derived from the Vp model using Gardner’s formula (Gardner et al., 1974).

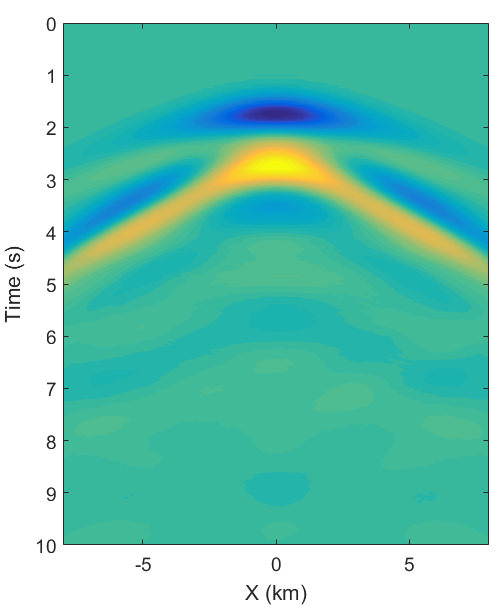
To run the first example we call the GUI with the following command:

FD\_seis\_sim(Vp,Vs,rho,X,Y,Z);

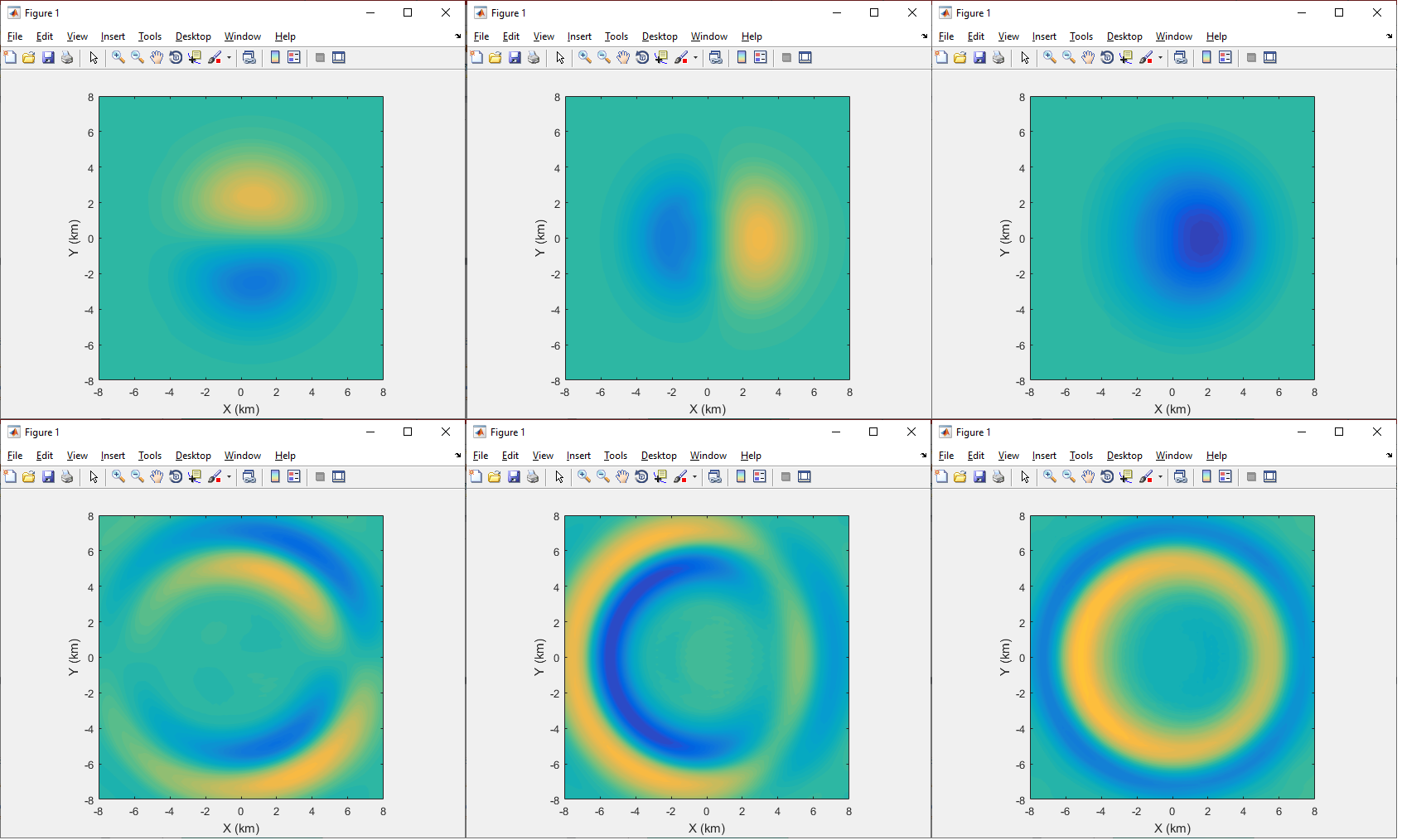
The following parameters where used for this example:



The resulting shot records on a linear receiver array aligned on the X axis in the middle of the model and on the surface are the following:



Resulting time snapshot at 2 and 4 secs of the three wavefield components (Vx, Vy, Vz) on the surface are shown below:

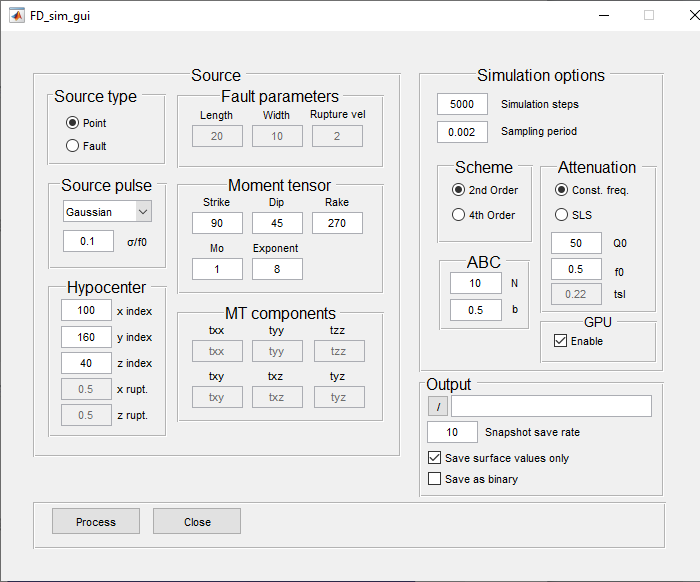


## Example 2: Double couple point source with layered 3D model and arbitrary topographic surface

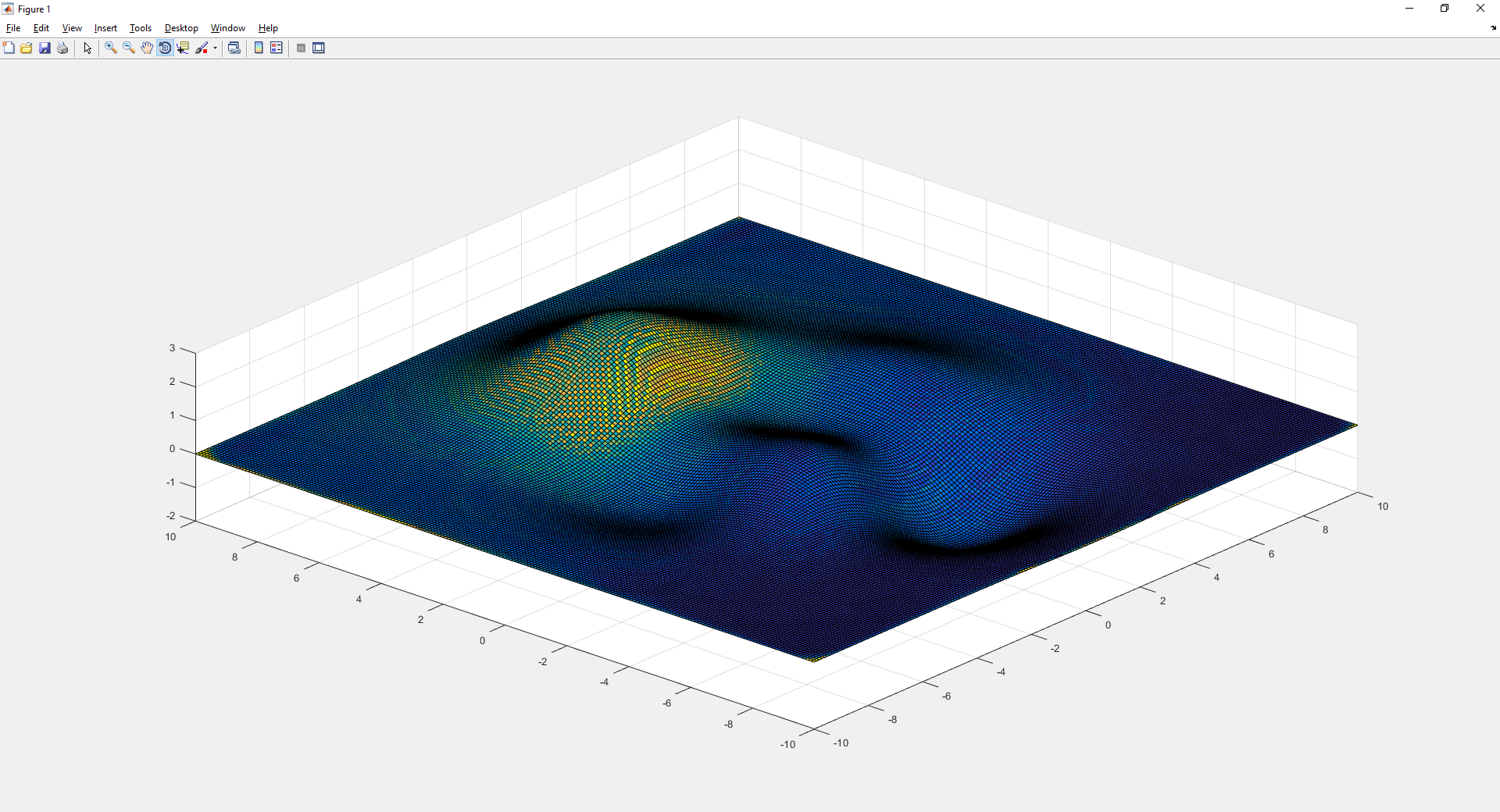
For the second example, the input velocity model was a simple 2D layered model with a Vp velocity of 3.2 km/sec up to the depth of 4 kms and 7 km/sec up to the depth of 10 kms. The Vs model is again related to the Vp model by a Vp/Vs ratio = 1.73, whereas the density distribution was again derived from Gardner’s formula. In this example case, a topographical relief was used instead of a flat top surface. To run the first example we call the GUI with the following command:

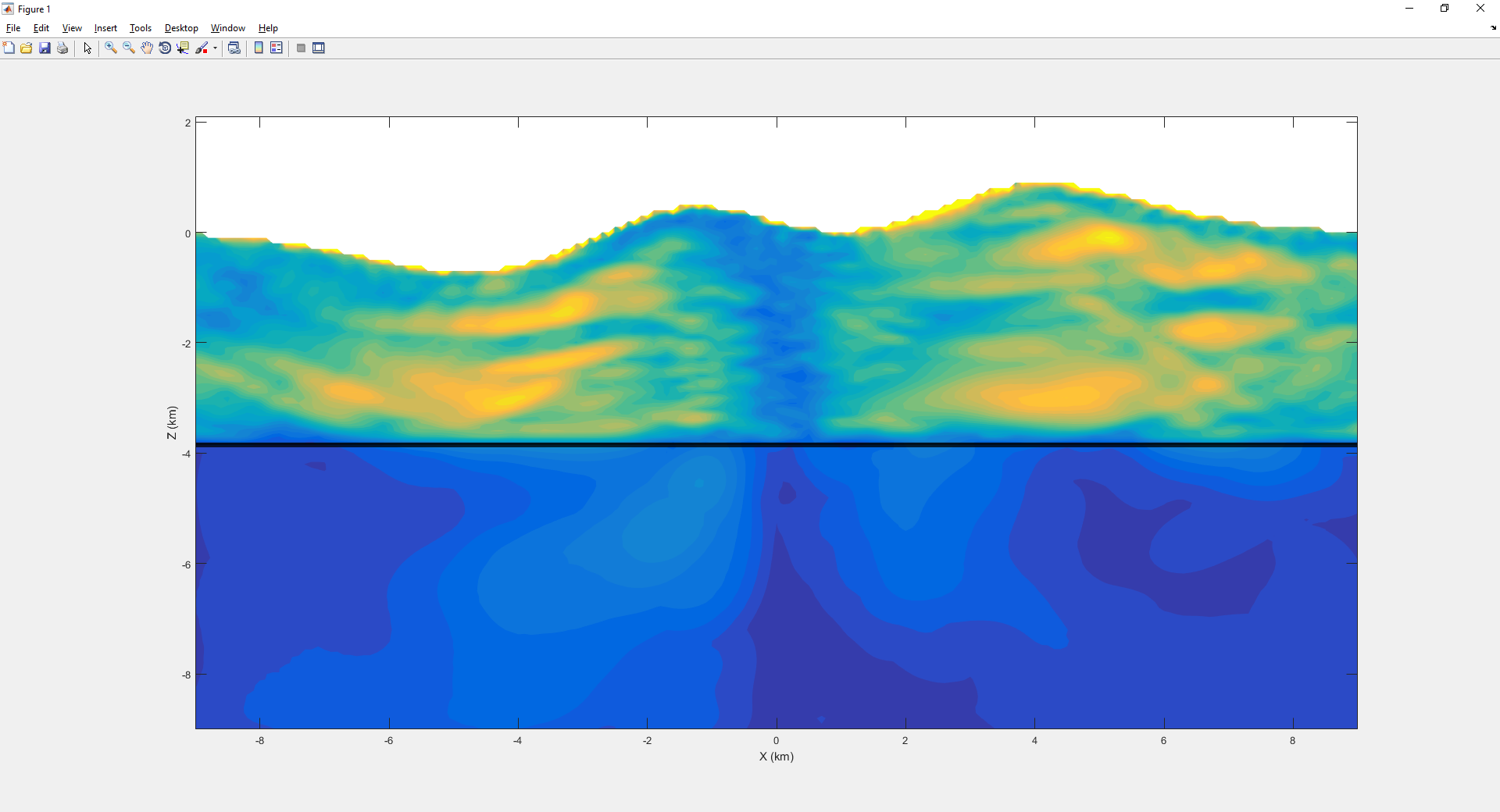
FD\_seis\_sim(Vp,Vs,rho,X,Y,Z,elev\_data,X,Y);

The following parameters where used for this example:



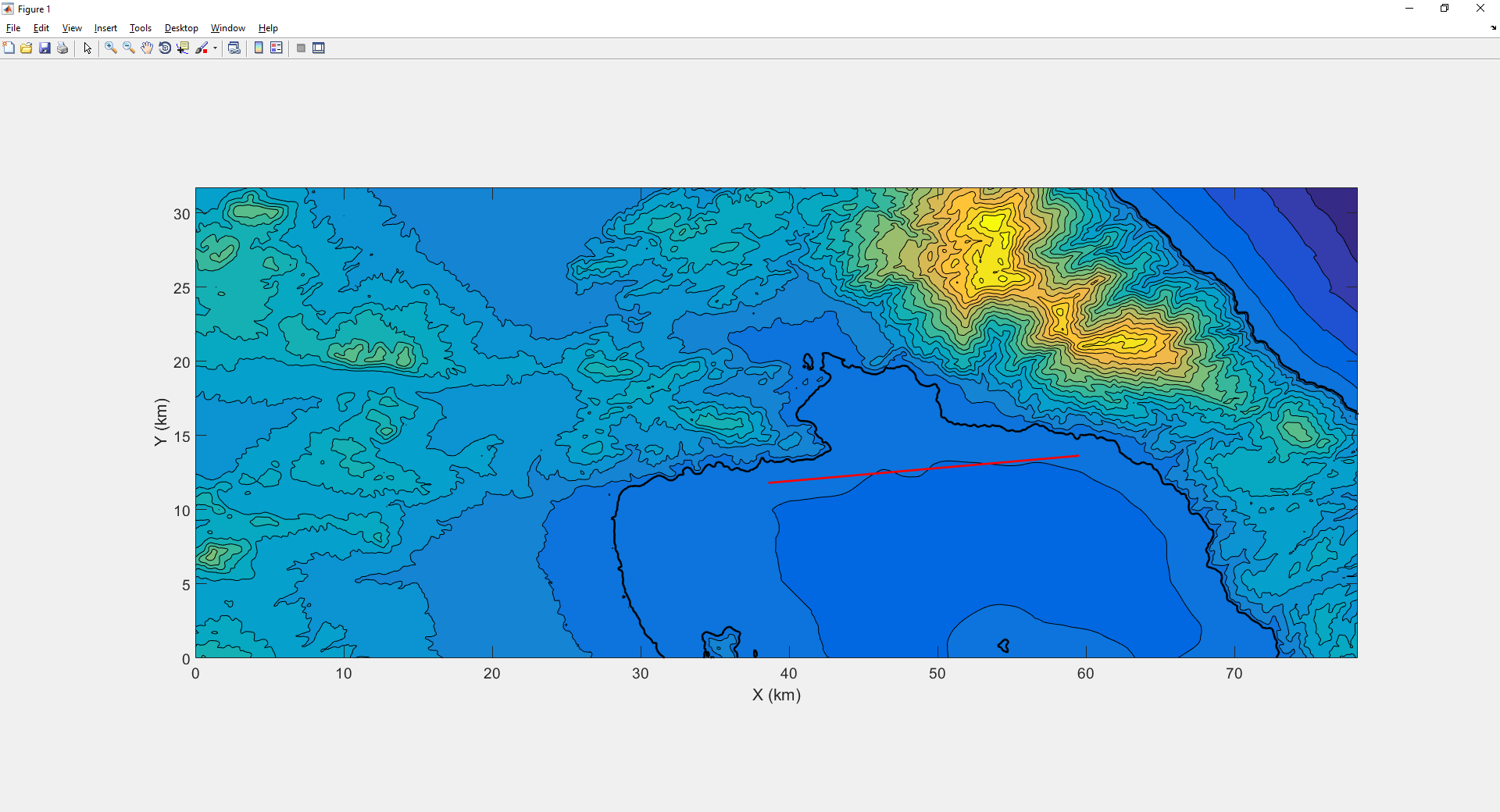
The following figure shows the maximum squared simulated velocity at the surface and a vertical slice at Y=5 km respectively. The figures were produced by the script: make\_figs2.





## Example 3: Finite source with layered 3D model and topographic surface (Volos fault in Volos region, Greece)

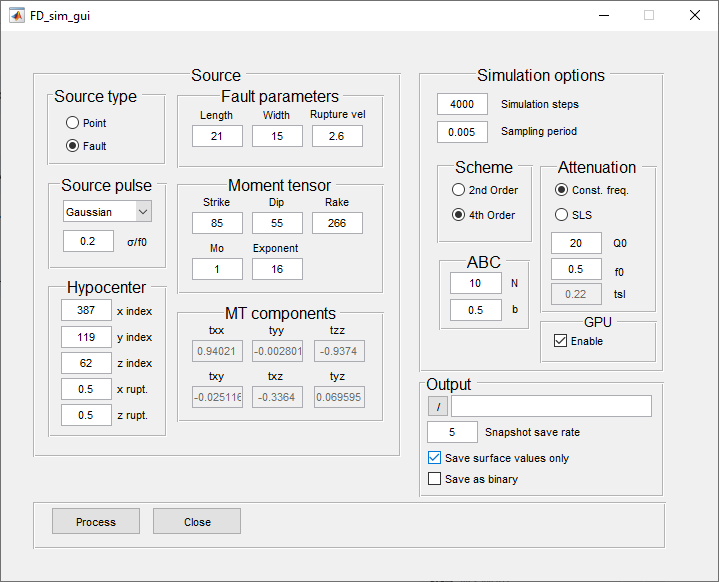
For the third example, we used a layered 3D model, related to the central Greek region (Volos region) with a Vp velocity of 4 km/sec for the first 1.2 kms of depth and 6 km/s from 1.2 to 15 kms. Similarly the Vs velocities are Vs=Vp./1.73 and rho=0.31\*((Vp\*1000)^0.25) Kg/m3. A topographical grid was also used as a top surface for the model, which had a resolution of 100m. The model includes 784 nodes in the X and 318 nodes in the Y direction. Variable node spacing between 65m (top) and 100m (bottom) was used in the Z direction for a total of 250 nodes. The total number of nodes in this case is 62.328x106. For this simulation example a finite fault source was also used, representing the fault of Volos (e.g. Caputo, 1996). The topography and the projection of the used fault on the surface are shown below:



To run the first example we call the GUI with the following command:

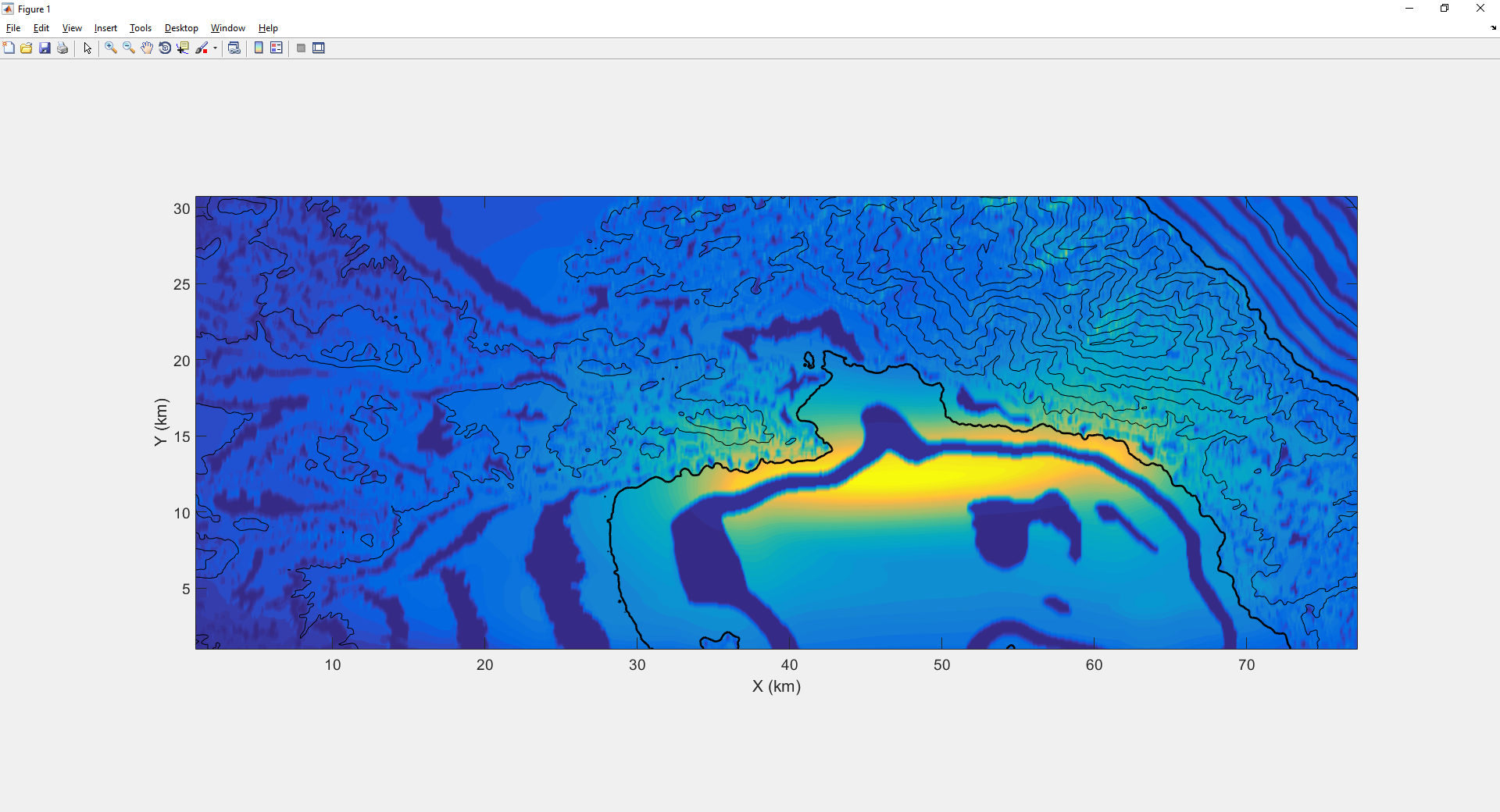
FD\_seis\_sim(Vp,Vs,rho,X\_out,Y\_out,Z\_out,topo\_data,X\_out,Y\_out)

The input data is loaded from the file: input\_data\_Volos.mat. The parameters used in this example are the following:



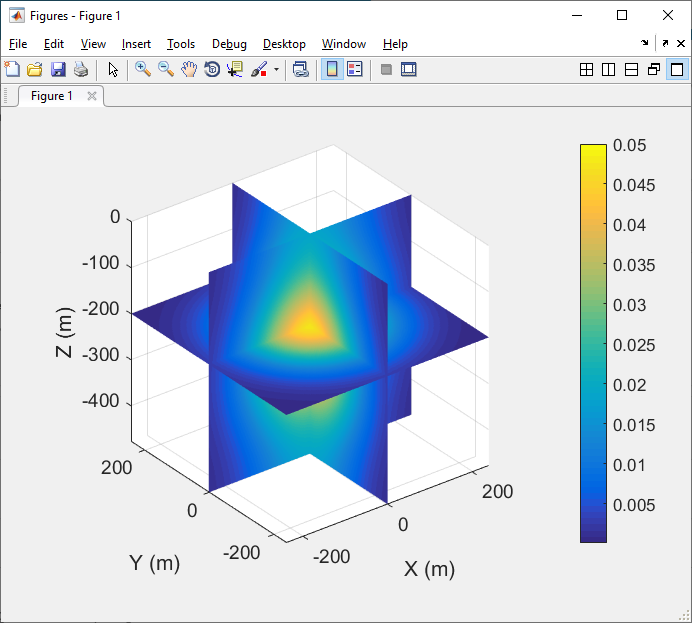
To run this example with this input dataset, with GPU processing enabled requires ~7.1 GB of available GPU RAM. Total simulation time on a Nvidia RTX2070 GPU for this example is: ~115 mins. In contrast, without GPU processing, the simulation time on an AMD Ryzen 7 3700x CPU is ~1320 mins.

The resulting maximum squared simulated velocity at the surface can be seen in the following figure. The image was obtained using the script make\_figs3.m within the example folder. Furthermore, a video showing the propagation of the Y component of the wavefield can also be created using the same script.



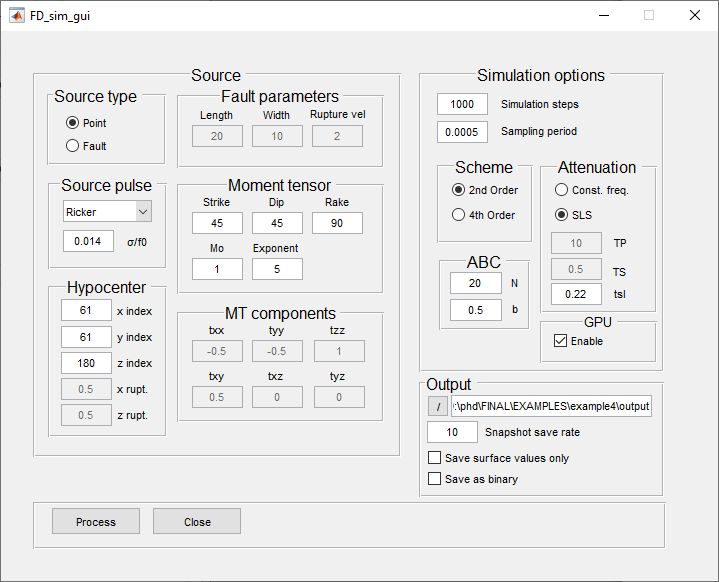
## Example 4: Homogeneous material with 3D attenuation values

The final example involves the use of a homogeneous velocity model (Vp=3000 m/s, Vs=Vp/1.78, rho=2158 gr/m3) with a 3D attenuation model represented by the Standard Linear Solid method with a TP and TS (see Bohlen, 2002) distribution as shown in the following figure (for TP):



To load the GUI for this example use the following command (Assuming Vs=Vp/1.73, rho=0.71.\*Vp/ TP=TS, tsl=2TP):

FD\_seis\_sim(Vp,Vp./1.73,0.71.\*Vp,X,Y,Z,1,1,1,TP,TP,2.\*TP);

The remaining parameters are given below: 

A video that shows the propagation of the resulting Z component of the simulated wavefield, as well as the image of the TP model can be created with the make\_figs4.m script.

# References

Bohlen, T. (2002). Parallel 3-D viscoelastic finite difference seismic modelling. *Computers & Geosciences*, *28*(8), 887-899.

Caputo, R. (1996). The active Nea Anchialos Fault System (Central Greece): comparison of geological, morphotectonic, archaeological and seismological data.

Gardner, G. H. F., Gardner, L. W., & Gregory, A. R. (1974). Formation velocity and density; the diagnostic basics for stratigraphic traps. *Geophysics*, *39*(6), 770-780.

Graves, R. W. (1996). Simulating seismic wave propagation in 3D elastic media using staggered-grid finite differences. *Bulletin of the Seismological Society of America*, *86*(4), 1091-1106.