2016 CIG WORKSHOP
SPECFEM3D

Dmitry Borisov, Princeton University

June 24, UC DAVIS

Contents

1 Getting started 3

2 Example #1: Internal mesher 4

3 Example #2: External mesher 7

1 Getting started

Connect to cluster:

ssh -X specfem3d_user@ubuntu.cse.ucdavis.edu

Download the SPECFEM3D_Cartesian software package:

git clone –recursive –branch devel https://github.com/geodynamics/specfem3d.git

Load appropriate fortran (GNU/intel) and MPI compilers:

module load ???
module load ???

Load python distribution:

module load ???

Configure SPECFEM3D_Cartesian for your system from the SPECFEM3D root directory:

cd specfem3d
./configure –with-openmp=no –with-mpi=yes –with-cuda=no –with-adios=no make -j 4 all

Compile all the source code:

make all

Check generated executable files:

ls ./bin

2 Example #1: Internal mesher

Copy an existing working directory to your location:
cp -r /home/specfem3d_user/Desktop/ex1_internal ./

Go to the **working directory** you have just created:
cd ./ex1_internal

Create **output folders**
mkdir OUTPUT_FILES; mkdir OUTPUT_FILES/DATABASES_MPI

Check the content of the "DATA" folder and the **parameter file**:
vim ./DATA/Par_file

Input model is a text-file with a header (first four lines) and the rest 6-columns: 1) x, 2) y, 3) z, 4) Vp, 5) Vs, 6) Density.
vim ./DATA/tomo_files/tomography_model.xyz

Source (active source):
vim ./DATA/FORCESOLUTION

Sensors information:
vim ./DATA/STATIONS

(1) Create mesh

Check the "Mesh_Par_file":
vim DATA/meshfem3D_files/Mesh_Par_file

Make sure that you create the mesh in a ".vtk" format for visualization in paraview:
CREATE_VTK_FILES = .true.

Submit job for mesh creation:
sbatch submit16_mesh3d

Check the status of the submitted job:
squeue

When the mesher is finished, check the output information:
vim ./OUTPUT_FILES/output_meshfem3D.txt

Copy generated *mesh.vtk files to your machine:
scp specfem3d_user@ubuntu.cse.ucdavis:~/ex1_internal/OUTPUT_FILES/DATABASES_MPI/*mesh.vtk ./

Open *mesh.vtk files in paraview (fig. 1):

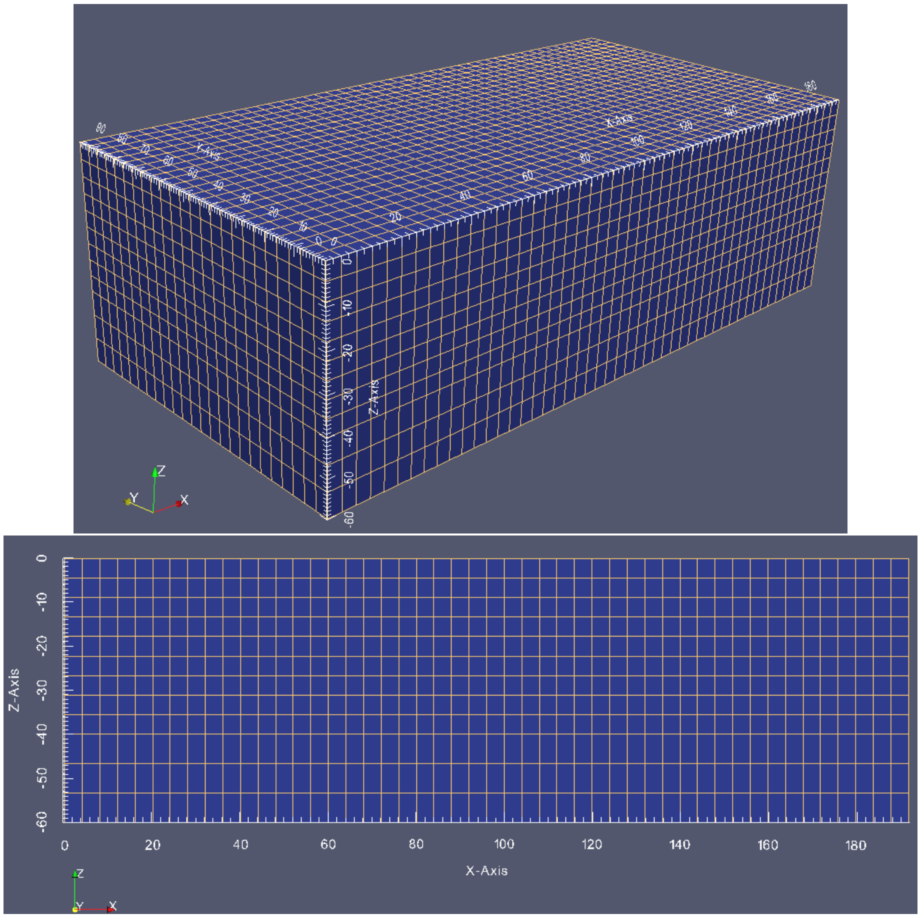

Figure 1: 3D mesh (top) and its vertical slice (bottom) from Example 1.

.

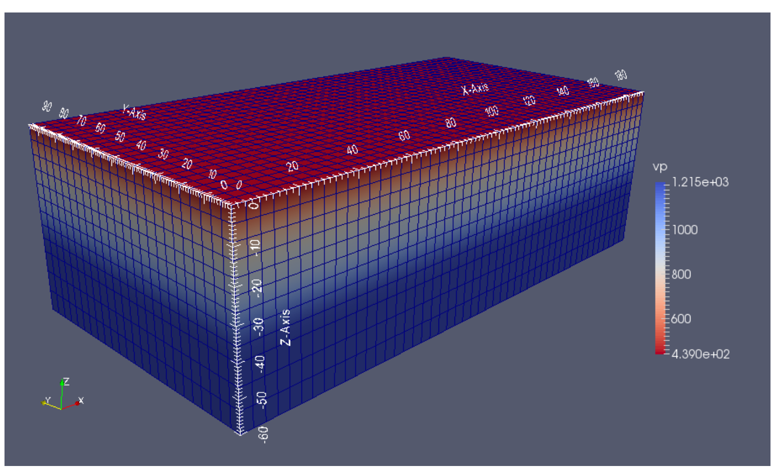

Figure 2: 3D P-wave velocity model overlaid with the mesh from Example 1

.

(2) Generate the databases:

sbatch submit24_generate

When the databases generation is finished, check the output information:
vim ./OUTPUT_FILES/output_mesher.txt

Combine the model from domain-decomposed pieces:
./bin/xcombine_vol_data 0 16 vp ./OUTPUT_FILES/DATABASES_MPI/ OUTPUT_FILES/ 0

Copy created vtk-files to your machine and then use PARAVIEW for model visualization (fig. 2).

(3) Run the solver

sbatch submit16_specfem

Find why there is a following problem with the solver:
forrtl: error (72): floating overflow

When the solver is finished, check the output information:
vim ./OUTPUT_FILES/output_solver.txt

Check seismograms by running python script from "py" folder (fig. 3):
python plot_3traces.py
Create a movie showing wavefield propagation at the surface:

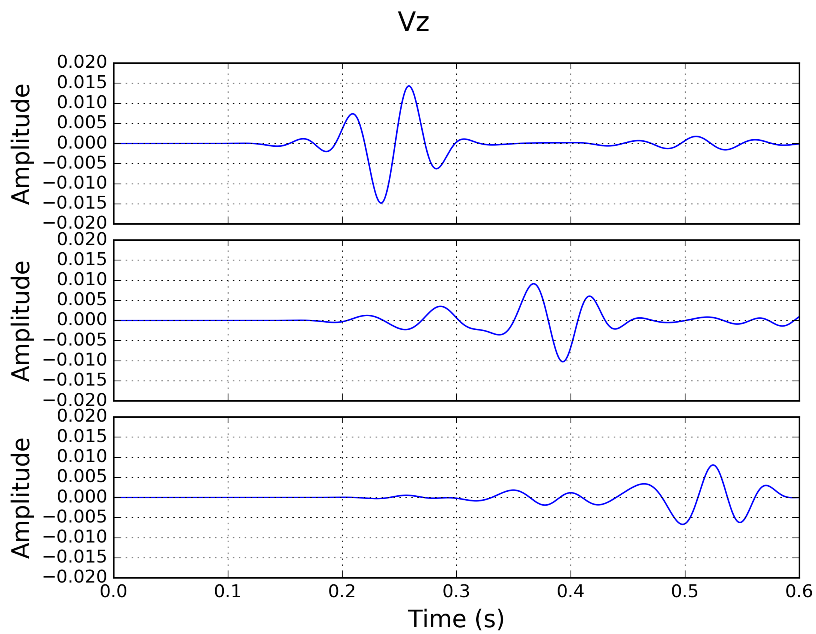

Figure 3: Seismic records (vertical component of particle velocity) from Example 1. Three traces from near (top), middle (mid) and far (bottom) offset are presented.

./bin/xcreate_movie_shakemap_AVS_DX_GMT

Use the following option values during that process:
2,1,2001,1,1

Copy AVS-files to your machine
scp specfem3d_user@ubuntu.cse.ucdavis:∼/ex1_internal/OUTPUT_FILES/AVS* ./

Open AVS-files in PARAVIEW and play (enjoy) the movie.

Attenuation:
vim ./DATA/Par_file
ATTENUATION = .true. USE_OLSEN_ATTENUATION = .true.

3 Example #2: External mesher

Copy an existing working directory to your location:
cp -r /scratch/fast/dborisov/ex2_external ./

Go to the **working directory** you have just created:
cd ./ex2_external

Create **output folders**
mkdir OUTPUT_FILES; mkdir OUTPUT_FILES/DATABASES_MPI

Go to the MESH-folder in your working directory:
cd ./MESH

Load required software:
module load netcdf/gcc/hdf5-1.8.16/4.4.0
module load hdf5/gcc/1.8.16

(1) Mesh format conversion

Compile and run c-code, required to convert the mesh from Trelis (Cubit) to SPECFEM3D format
gcc trelis2specfem3d.c -o trelis2specfem3d
./trelis2specfem3d mountain_mesh.e -bin=1

Rename files:
mv mountain_mesh_absorbing_surface_file_xmax absorbing_surface_file_xmax
mv mountain_mesh_absorbing_surface_file_ymax absorbing_surface_file_ymax
mv mountain_mesh_absorbing_surface_file_ymin absorbing_surface_file_ymin
mv mountain_mesh_absorbing_surface_file_xmin absorbing_surface_file_xmin
mv mountain_mesh_absorbing_surface_file_zmin absorbing_surface_file_zmin

```
mv mountain_mesh_free_or_absorbing_surface_file_zmax    free_or_absorbing_surface_file_zmax
mv mountain_mesh_materials_file    materials_file
mv mountain_mesh_mesh_file    mesh_file
mv mountain_mesh_nodes_coords_file    nodes_coords_file
```

Check the elastic properties of the model:

```
vim nummaterial_velocity_file
```

Based on the Example1, define your own sensors and sources in the DATA folder.

(2) Decompose mesh

From the working directory:

```
./bin/xdecompose_mesh 24 ./MESH/ ./OUTPUT_FILES/DATABASES_MPI/
```

Check the output:

```
ls ./OUTPUT_FILES/DATABASES_MPI/
```

(3) Generate the databases

```
sbatch submit24_generate
```

(4) Run the solver

```
sbatch submit24_specfem
```