================== DRAFT ===================

FD3Dspher User's Manual
Version 1.0

==

Wei Zhang and Yang Shen
University of Rhode Island, Narragansett, RI 02882, USA

1 Introduction

This software package FD3Dspher simulates seismic wave propagation in a spherical coordinate. It solves the stress-velocity wave equations using a non-uniform, non-staggered finite-difference (FD) method. Unlike staggered-grid FD simulation of wave propagation in the spherical earth (Igel et al. 2002), there is no interpolation of particle velocity and stress in this scheme, so the high-order precision of FD computation is maintained. The spatial FD operator in this package is the DRP/opt MacCormack scheme (Hixon 1997) with 4th-order accuracy and optimization in numerical dispersion and dissipation. The time-marching scheme can be 2nd-order Euler or 4th-order Runge-Kutta, depending on the tradeoff between accuracy and computational efficiency. It includes a complex frequency-shifted perfectly-matched-layer implementation with auxiliary differential equations (ADE CFS-PML) to absorb waves at the boundaries surrounding the computational domain (Zhang et al. 2010) . One of the advantages of the ADE CFS-PML implementation is that it is straightforward to use with a high-order time-marching scheme to achieve a high-order accuracy.

The package FD3Dspher provides a relatively straightforward way to set up a simulation of any model dimension and at almost any location, since the effective medium properties at discontinuities in the Earth's interior are the harmonic average of the elastic moduli and the arithmetic average of densities (Moczo et al. 2002). Effects due to lateral variations in wave speed (including full 21-parameter anisotropy), density, the oceans, and attenuation are included. We caution that the treatment of attenuation in this version (1.0) is approximate as in Graves (1996). This version also has a spherical free surface. So the earth's ellipticity and topography are not accounted for. To date this software package has been applied to regional-scale problems, in which the effects of topography and ellipticity are secondary or can be corrected.

The package has been validated by comparing simulation results with synthetics calculated from normal-mode summation. See Graves (1996), Hixon (1997), Pitarka (1999), Zhang et al. (2010), and Zhang et al. (2011) for details of the finite-difference method.

If you have questions, comments and suggestions, please contact Wei Zhang (wzhang@gso.uri.edu) or Yang Shen (yshen@gso.uri.edu).

1.1 Citation

Please refer to the following reference, if you use the package FD3Dspher or components of the package for your research.
Zhang, W., Y. Shen, and L. Zhao, 2011. Three-dimensional anisotropic seismic wave modeling in spherical coordinate by a collocated-grid finite difference method, Geophys. J. Int., 188, 1359-1381, doi:10.1111/j.1365-246X.2011.05331.x, 2012.

1.2 Acknowledgements

The software package FD3Dspher (version 1.0) is based on research supported by the U.S. National Science Foundation under Grants No. EAR-0738779 and EAR-0551117 and the Air Force Research Laboratory under Contract No. FA8718-06-C-0014.

1.3 Copyright

1.4 License

Open-source license (The exact form and terms of the license will be determined). This software package is distributed for non-commercial academic and/or government research only.

1.5 Disclaimer

The software package FD3Dspher is research and experimental in nature and is being distributed as is without any warranty.

2 Compiling the FD3Dspher package

2.1 Required Libraries

The software package is written for computation on Linux/Unix clusters. Several GNU softwares are needed to compile the source codes:

- Make
- cpp (C preprocessor)
- Bash (Scripts are in Bash)
- Sed/Awk (Sed/Awk are used in scripts)

Other required libraries for compiling or using the package are: the Fortran 90 compiler, NetCDF library, MPI library and MATLAB (including snctools, http://mexcdf.sourceforge.net/). See the FD3D-all-install manual for installation of the softwares.

2.2 Subdirectories

The software package FD3Dspher is distributed as a gzipped tar file. Unpack the file in the directory where the package is to be installed,

```
tar xfz FD3Dspher.v1.0.tar.gz
```

The unpacked directory contains the following subdirectories and files:

#===

```
code/
   srcF/
       source codes
   Makefie*
       makefile for this package
   run.make.sh
       script to compile the source codes
mfiles/
   matlab scriptes to show the result
run/
   examples with different configurations
```

#===

2.3 Compiling the Source Codes

Compiling is carried out in the code/ directory. The compiling process is controlled by the Makefile. You can use "flags" in the Makefile to enable or disable "features" in the codes. To enable a feature, remove # at the beginning of the "flag" to activate the feature. The Makefile.opt.LOC file in the directory is called by the Makefile to specify the machine-and-location-dependent compiler(s) and libraries.

The "flags" in the Makefile are:

- WHEREAMI
  Define the machine location name, LOC. This enables the Makefile to identify Makefile.opt.LOC.
- DEBUG
  Set the flag "ON" for debug purposes.
- STATIC
  Set the flag "ON" for statically linking shared libraries.
- USEOMP
  Use OpenMP for shared-memory parallelization.
- WITHQS
  Include the effects of attenuation.
- SrcSmooth
  Enable a distributed source on grids adjacent to the source location.
- SrcSurface
  Enable a source on the free surface.
- CondFreeTIMG
  Anti-symmetrically imaging the stress components to the points above the free surface.

- CondFreeVHOC
  Use a higher-order compact FD scheme to calculate velocity difference for the free surface boundary condition.
- CondFreeVLOW
  Ues a low-order FD scheme to calculate velocity difference for the free surface boundary condition.

The variables in Makefile.opt.LOC are:

- COMPILER
  The compiler to be used (Intel, PGI).
- FC
  The full pathname of mpif90.
- NETCDF
  The full pathname of the NetCDF library root path. The NetCDF should be compiled with f90 support.

To compile the codes, execute

```
make
```

or run

```
./run.make.sh
```

If compiling runs successfully, the following compiled programs should be in the "bin" directory:

seis3d_grid, seis3d_media, seis3d_media_mpi, seis3d_metric, seis3d_source, seis3d_station, seis3d_wave_mpi

where mpi in the file names stands for distributed computing using MPI. Otherwise, check make.all.log file (if run.make.sh is used) to figure out what causes the errors.

3 Configuration Files

The programs use the following four configuration files to specify input and output parameters and variables:

- SeisFD3D.conf: The main configuration file. Do not change the name of this configuration file, as it is "hard-wired" in the source codes.
- SeisGrid.conf: The grid configuration file. The name can be changed in SeisFD3D.conf.
- SeisMedia.conf: The media configuration file. The name can be changed in SeisFD3D.conf.
- SeisSource.conf: The source configuration file. The name can be changed in SeisFD3D.conf.

3.1 SeisFD3D.conf

The variables in the main configuration file are:

- fnm_log
  Run-time log file name.
- dims
  Numbers of threads (CPU cores) in the three (colatitude,longitude and radial) dimensions.
- ni,nj,nk
  Numbers of grids in ONE thread along the colatitude,longitude, and radial dimensions. Total number of grids and the following "tags" are used by matlab scripts to draw media and snapshots.
- nt
  Total number of time steps.
- stept
  Time step in second. It should satisfy the stability criterion: $C_{max}\frac{\nabla t}{\nabla h} < 0.69$. Note: the run-time log file from executing pbs_media_mpi.sh (Section 4) provides the estimated maximum allowed time step in each thread.
- GRID_CONF
  Name of the grid configuration file (e.g., SeisGrid.conf).
- MEDIA_CONF
  Name of the media configuration file (e.g., SeisMedia.conf).
- SOURCE_CONF
  Name of the source configuration file (e.g., SeisSource.conf).
- GRID_ROOT, MEDIA_ROOT, SOURCE_ROOT, STATION_ROOT, OUTPUT_ROOT
  Directories for the input and output files.
- abs_number
  Numbers of the PML grids on the six external boundaries (zero PML for the free surface).
- abs_velocity
  Representative velocities used to calculate the PML parameters (Zhang et al. 2010).
- CFS_bmax
  Maximum b for the PML. Suggested values = Vs/(0.5*6*dh*fc), where fc is the dominant frequency (Zhang et al. 2010).
- CFS_amax
  Maximum a for the PML. Suggested value = pi*fc (Zhang et al. 2010).
- number_of_snap
  Total number of snapshot volumes and surfaces
- snap_xxx
  "xxx" should begin from "001" to "number_of_snap".
  The first three numbers are the beginning grid index along the three (colatitude, longitude, and radial) dimensions for the snapshot output.

The second three numbers are the numbers of SAVED grids in the three dimensions.
The third three numbers are the grid intervals of output in the three dimensions.
The Last two numbers are the interval of output time step and the maximum number of output time steps in each nc file. V stands for velocity output, T stress output, and TV both velocity and stress outputs.

- point_result_method
  Output seismogram is the waveform at (1) the NEAREST grid, or (2) the result of LINEAR interpolation of values at the surrounding grids. Only the first option, NEAREST, is currently tested and used.

- topo_hyper_height
  If the height of the receiver in "line_xxx" and "recv_xxx" (parameters specified below) is larger than this value, then the output point is located on the free surface.

- tinv_of_seismo
  The time step interval for seismograms.

- number_of_inline
  The total number of lines along which to output seismograms. If the number is zero, the following "line_xxx" is ineffective.

- line_xxx
  "xxx" should begin from "001" to "number_of_inline".
  The first three numbers are the coordinates of the beginning point along the line.
  The second three numbers are spatial intervals along the three dimensions.
  The last number is the total number of receivers along the line.

- number_of_recv
  The total number of individual receivers to have seismograms. If the number is zero, the following "recv_xxx" is ineffective.

- recv_xxx
  "xxx" should begin from "001" to "number_of_recv".
  The values are the coordinates of the receiver(s).

Following is an example of SeisFD3D.conf.

#===

```
########################################################################
#                  checkpoint/restart parameter                        #
########################################################################
CHECKPOINT_ROOT = ./checkpoint
checkpoint_tinv = 10000
run_from_checkpoint = 0
urgent_checkpoint = checkpoint.dat
# run-time log file
fnm_log = seis3d_wave.log
########################################################################
#                         for mpi_mod                                  #
########################################################################
########################################################################
```

```
#                        for main program                               #
#########################################################################

dims = 4 4 1
ni     = 87 # 348 total_grids_in_x
nj     = 114 # 456 total_grids_in_y
nk     = 58 # 58 total_grids_in_z
nt     = 2400 # total_time_steps
stept  = 0.50 # time_interval_in_s

#########################################################################
#                         mod_ grid,media,src                           #
#########################################################################
GRID_CONF = SeisGrid.conf
MEDIA_CONF  = SeisMedia.conf
SOURCE_CONF = SeisSource.conf

# dir configure
GRID_ROOT   = ./input
MEDIA_ROOT  = ./input
SOURCE_ROOT = ./input
STATION_ROOT = ./input
OUTPUT_ROOT = ./output
#########################################################################
#                        for abs_mod                                    #
#########################################################################
abs_number = 12 12 12 12 12 0
abs_velocity  = 5740.0 5740.0 5740.0 5740.0 10.0e3 5000.0
CFS_bmax  = 3.79 3.79 3.79 3.79 7.59 7.59
# Vs/(0.5*6*dh*fc)
CFS_amax  = 0.209 0.209 0.209 0.209 0.209 0.209
# pi*fc (fc=1/15Hz)
#########################################################################
#                         for output                                    #
#########################################################################
# final snap output
number_of_snap = 2
# id        subs        subc         subt     tinv
snap_001 =  13  13  18   81 108 40      4 4 1  4 10000 TV
snap_002 =  1   1   58   348 456 1       1 1 1  4 10000 V

# final seismogram
point_result_method = NEAREST # LINEAR
topo_hyper_height = 8.0E3  # output point on the free surface if z > this value
tinv_of_seismo = 4 # time step interval for seismogram output

# seismo-line output
number_of_inline = 0
# line_id     (x0,y0,z0)  | (dx,dy,dz)  |count
line_001 =  90.0  -15.0 9000E3 | 0.0 0.04 0.0 | 750
```

```
# seismo-point output
number_of_recv = 2
recv_001 = 59.0 116.0 9.0E3
recv_002 = 59.0 95.0  9.0E3

# vim:ft=conf:ts=4:sw=4:nu:et:ai:
```

#===

3.2 SeisGrid.conf

The variables in the grid configuration file are:

- distance2meter
  Scale to convert from km to meter.
- steph
  Grid spacings in the colatitude, longitude and radial directions. The grid locations are determined by the starting point (specified in the following line) and the numbers of grid specified in SeisFD3D.conf. If the values are zeros, then a non-uniform or uniform grid is specified individually in the following.
- theta0_phi0_rmax
  The starting point (the colatitude, longitude and radius of the northwestern most point on the free surface).
- x grid
  Colatitude of the grids. Colatitude (90-latitude) increases from north to south.
- y grid
  Longitude of the grids.
- z grid
  Radius of the grids. Radius increases outwards from the center of the Earth.

Following is an example of SeisGrid.conf (showing only the first 2 lines of the colatitude and longitude grid values and the last 3 lines of the radius grid values).

#===

```
#######################################################################
#                           for seis3d_grid                           #
#######################################################################

distance2meter = 1.0E3

#######################################################################
#                            uniform grid                             #
#######################################################################
```

```
steph = 0.00 0.00 0.0
theta0_phi0_rmax = 57.0 104.5 6371

########################################################################
#                          non-uniform grid                            #
########################################################################
# x grid
<x grid>
57.000000 57.100000 57.200000 57.300000
57.400000 57.500000 57.600000 57.700000
...
# y grid
<y grid>
104.500000 104.600000 104.700000 104.800000
104.900000 105.000000 105.100000 105.200000
...
# z grid
<z grid>
...
6334.330000 6340.930600 6346.797800 6351.931600
6356.332000 6359.999000 6363.666000 6367.333000
6371.000000
```

#===

A rule of thumb for the vertical grid size is that it should be no more than 1/3 of the horizontal grid spacing near the surface to ensure sufficient vertical parameterization for surface waves. This vertical grid size may gradually increase with depth to one to two times the horizontal grid size at depths great than one minimum wave length. It is recommended that you generate the grids with a matlab script or other programs and then copy the grids to SeisGrid.conf. See the matlab scripts creat_grid_xy.m and creat_grid_z.m in the directory ./run/test_grid/config/grid for example). If the grids in this configuration file fall outside of the volume of the media specified in SeisMedia.conf, the program seis3d_media (or seis3d_media_mpi) extends the values at the exterior boundaries of the media to fill the grids.

3.3 SeisMedia.conf

The code determines the medium parameters at several sampling points in each cell centered or roughly centered on the grid, then performs a harmonic or arithmetic summation to determine the volume-integrated effective media parameters for the cell. There are several ways to set up the medium parameters:

- cart1d
  A 1D model.

- interface
  A layered model with topographic interfaces and medium parameters that vary linearly in the vertical direction. The model is defined by the vertical position of each interface at horizontal sampling points and the parameter values just above and below the interface. The values of the parameters are constant along the interface.

- layered
  This is similar to the above "interface" type, except the structure is defined by the thickness of the layer at each horizontal sample point and the parameters at the top and bottom of the layer.

- composite
  This type of models has interface topography and laterally varying parameters along the interfaces. The values of the parameters at any position inside a layer are determined by a polynomial interpolation along the vertical direction using values at the top and bottom of the layer.

  See ./run/test_grid/config/conf_media_composite.m for an example of preparing composite models and Figure 1 for a FD medium constructed from the composite model. The following example shows the dimensions, variables and properties of the nc file of a composite model.

  #===

  ```
  [user dir]$ ncdump -h SeisMedia.composite.w.china.crust2.nc
  netcdf SeisMedia.composite.w.china.crust2 {
  dimensions:
          theta = 17 ;
          phi = 24 ;
          layer = 27 ;
          side = 2 ;
  variables:
          float theta(theta) ;
          float phi(phi) ;
          float Vp_poly_d(layer) ;
          float Vs_poly_d(layer) ;
          float rho_poly_d(layer) ;
          float thickness(layer, phi, theta) ;
          float Vp(layer, phi, theta, side) ;
          float Vs(layer, phi, theta, side) ;
          float rho(layer, phi, theta, side) ;
  }
  ```

  #===

- volume
  This type of models specifies parameter values at a 3D grid points, which are not necessarily the FD grids. There are no explicit interfaces in this type of models. See ./run/test_grid/config/conf_media_volume.m for an example of preparing volume models and Figure 2 for a FD medium constructed from the volume model.

Following is an example of SeisMedia.conf:

#===

```
######################################################################
#              configuration file of seis3d_media                    #
######################################################################
```

```
half_sample_point = 0 0 2

######################################################################
#                          background model                          #
######################################################################

#background_type : cart1d interface layered composite volume
#background_format : ascii  nc

#background_type   = cart1d
#background_format = ascii
#background_filename = prem_noocean.txt

#background_type   = interface
#background_format = ascii
#background_filename = model.interface.VTI.velocity.conf

#background_type   = layered
#background_format = ascii
#background_filename = earth.global.prem.iso.solid.1d.layered

#background_type   = volume
#background_format = nc
#background_filename = SeisMedia.volume.tibet.CUB2.nc

background_type   = composite
background_format = nc
background_filename = SeisMedia.composite.w.china.crust2.nc

######################################################################
#                         model perturbation                         #
######################################################################

#perturbed_type : none volume verpoly

perturbed_type =  none
perturbed_format = nc
perturbed_filename = perturbed_prem.nc

# vim:ft=conf:ts=4:sw=4:nu:et:ai:
```

#==

The parameter half_sample_point stands for the numbers of sampling points within a half cell in the 3 dimensions. If the number in one dimension is zero, then only the FD grid is used in that dimension. These medium sampling points within the cell are used for a harmonic averaging of the elastic moduli and and an arithmetic averaging of densities to determine the volume-integrated effective media parameters (Moczo et al. 2002). The cart1d, interface and layered models are ascii files, while the volume and composite models are netcdf files. If the parameter perturbed_type = none, the following lines are ineffective. The parameter perturbed_filename is the perturbation to the reference model in percentage. This is used to combine the background model with a tomographic solution. The only file format for perturbation is nc.

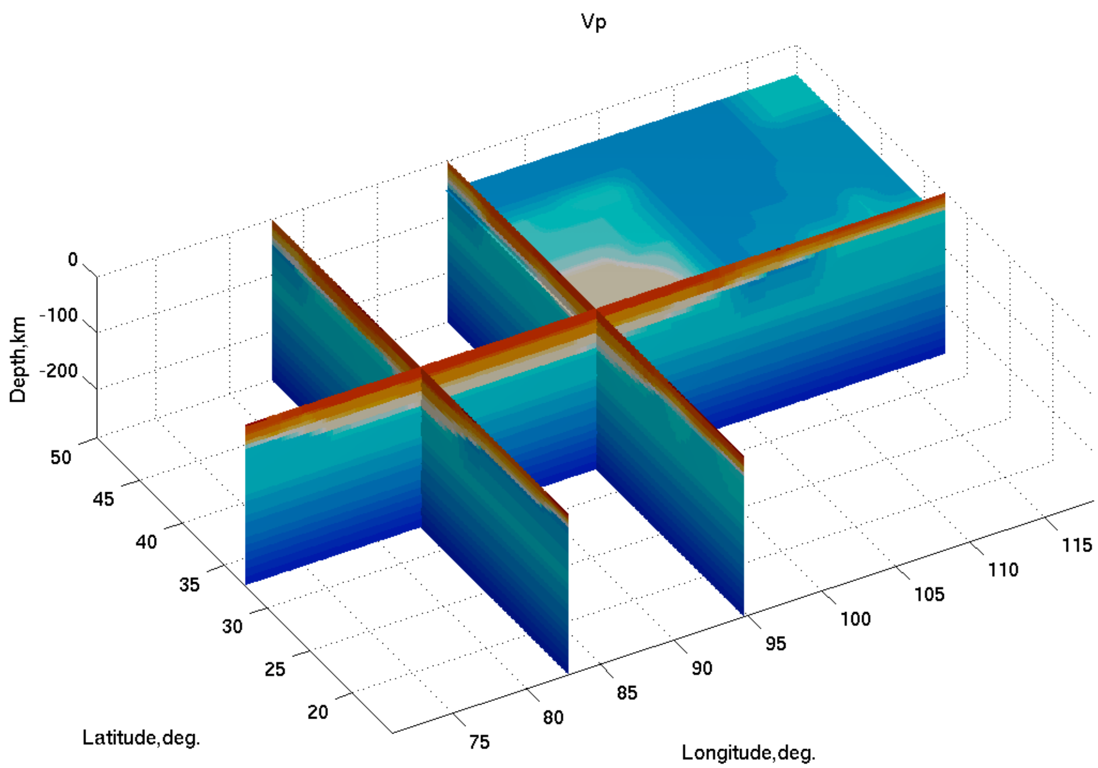

Figure 1: Vertical and horizontal cross-sections of a FD medium constructed from a composite of the CRUST 2.0 and AK135 models. The P-wave speed shows the thick crust beneath the Tibetan plateau. The color scale is 5.5-9 km/s from red to blue.

3.4 SeisSource.conf

The variables in the source configuration file are:

- distance2meter
  Scale to convert from km to meter.
- src_hyper_height
  If the source height is great than this value, then the source is located on the free surface.
- number_of_force_source
  Self explanatory. If the number is zero, then the force parameters are ineffective.
- force_stf_window
  Number of source time window.

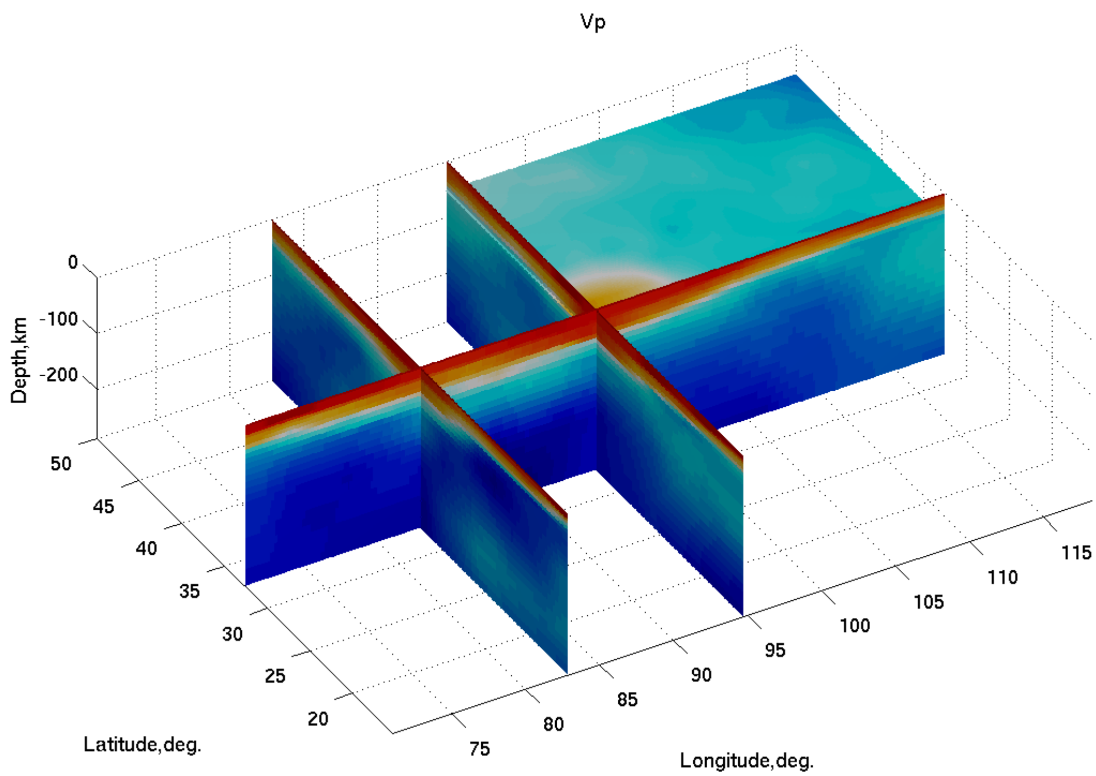

Figure 2: Vertical and horizontal cross-sections of a FD medium constructed from the CUB 2.0 model (Shapiro and Ritzwoller 2002). The P-wave speed is converted from the S-wave speed. The geometries of the cross-section and the color scale are the same as in Figure 1.

- force_stf_type
  Type of the source time function (gauss, ricker, bell).

- force_stf_timefactor
  The meaning depends on the type of the source time function.
  gaussian: the time shift of the center from the start time (t0)
  ricker: t0
  bell: starting time

- force_stf_freqfactor
  The meaning depends on the type of the source time function.
  gaussian: the half width of the Gaussian (a)
  ricker: fc
  bell: width

- anchor_force
  The eight force parameter values are: colatitude longitude height start-time f0 fx fy fz
  The force is f0*(fx,fy,fz). A positive fz is a vertical force point upwards.

- number_of_moment_source
  Self explanatory. If the number is zero, then the moment parameters are ineffective.

- moment_stf_*
  Similar to the parameters of the force source time function

- moment_mech_input
  Options: moment and angle

- anchor_moment
  The eleven moment parameters are: Colatitude, longitude, height, start-time, m0, Mtt, Mpp, Mrr Mtp Mrt Mrp
  where m0 is in N-m. The mement is m0*(Mtt, Mpp, Mrr, Mtp, Mrt, Mrp). The subscript r stands for up, t is south, and p is east. See Aki and Richards for conversions to/from other coordinate systems.

Following is an example of SeisSource.conf.

```
#===========================================================

######################################################################
#                        for seis3d_source                           #
######################################################################

distance2meter = 1.0E3
src_hyper_height = 9e3

######################################################################
#                        single force source                        #
######################################################################
number_of_force_source = 1
force_stf_window = 1
force_stf_type = gauss
force_stf_timefactor = 12  # gauss t0; ricker t0; bell starting
force_stf_freqfactor = 4  # gauss a;  ricker fc; bell width
# x,y,z           | start  | f0    | fx   fy   fz
<anchor_force>
59.00  106.50  2e10   0.0    1.0e+16   0.0 0.0 1.0

######################################################################
#                        moment tensor source                        #
######################################################################
number_of_moment_source = 0
moment_stf_window = 1
moment_stf_type = bell_int
moment_stf_timefactor = 0.0  # gauss t0; ricker t0; bell starting
moment_stf_freqfactor = 4.0 # gauss a;  ricker fc; bell width
moment_mech_input = moment  # moment, angle
# x,y,z    start(s) | m0(N.M) |        Mtt    Mpp      Mrr    Mtp     Mrt    Mrp
```

```
<anchor_moment>
60.28 95.490 6359 0.0 1.0e23 -6.69e+00 6.20e+00 4.87e-01 1.02e+01 -4.56e+00 8.35e-01

#===================================================================
```

4 Running Wave Simulation Programs

After compiling the codes and setting up the configuration files, you can now run the programs. There are six steps: (1) generating the FD grids for the individual threads, (2) setting up media parameters, (3) calculating the metrics, (4) assigning source points, (5) assigning receivers, and (6) running the wave equation solver.

1. Generating grids and related parameters. Run
   ```
   ./bin/seis3d_grid
   ```

2. Setting up media parameters. Run
   ```
   ./bin/seis3d_media
   ```
   or the mpi version
   ```
   qsub pbs_media_mpi.sh
   ```
   You need to edit the mpi shell script to distribute cpu threads that are consistent with the thread dimensions in SeisFD3D.conf. After seis3d_media or its mpi version is finished, check the run-time log (e.g., spher.media.*) to see if the maximum allowed time step is greater than the time step (*stept*) in SeisFD3D.conf. Adjust *stept* if necessary. You can inspect the FD medium using the MATLAB code draw_media_surf_all.m (Figures 1 and 2).

3. Calculating the metrics. Run
   ```
   ./bin/seis3d_metric
   ```

4. Assigning source parameters. Run
   ```
   ./bin/seis3d_source
   ```

5. Distributing receivers. Run
   ```
   ./bin/seis3d_station
   ```

6. Submiting the wave simulation job under a job control system
   ```
   qsub pbs_wave_mpi.sh
   ```
   Again, you need to edit the shell script to distribute cpu threads that are consistent with the thread dimensions in SeisFD3D.conf.

5 Post Processing and Graphics

When the program finishes, each CPU thread puts its output files into "./output" directory. You can inspect the results using the following matlab scripts:

- draw_snap_surf_all.m
  This matlab script reads the outputs and draws snapshots of wave propagation. It calls several functions and scripts in the directory mfiles/.

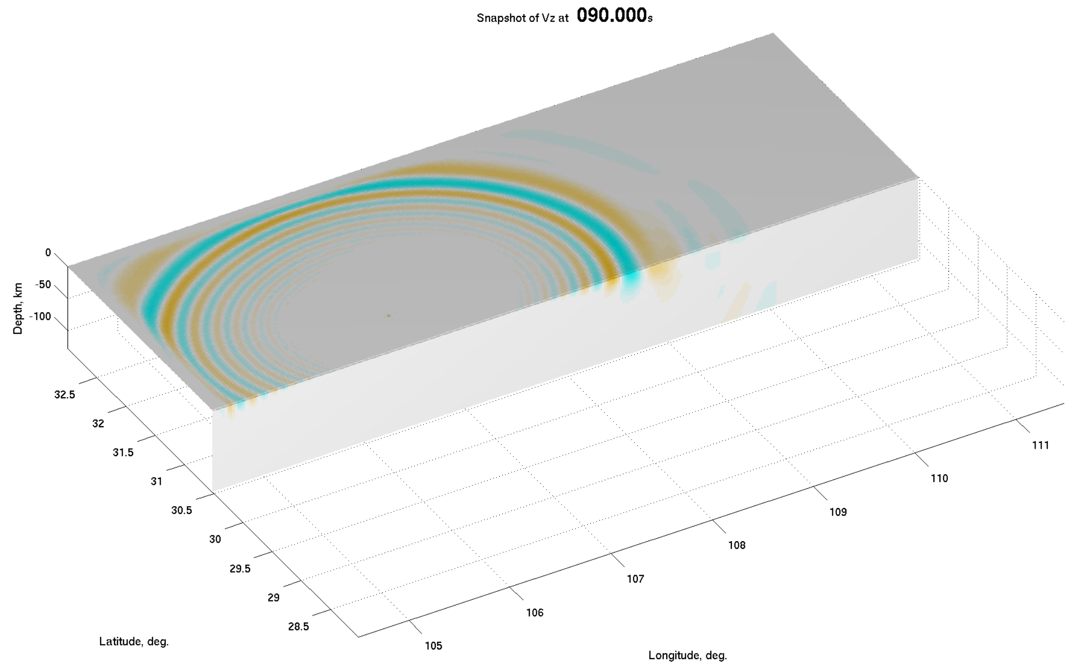

Figure 3: A snapshot of the 3D wave field (particle velocity in the radial direction) generated by a point source on the free surface. The snapshot is shown on the free surface and a vertical cross-section at 30.5 degrees north. Only the top one-third of the simulation box is shown. Notice the clean absorption of the wave field at the north and west boundaries of the model on the free surface.

- draw_seismo_single.m
  If you have seismograms at individual receivers, you can view the waveforms using this script. It calls several functions and scripts in the directory mfiles/.

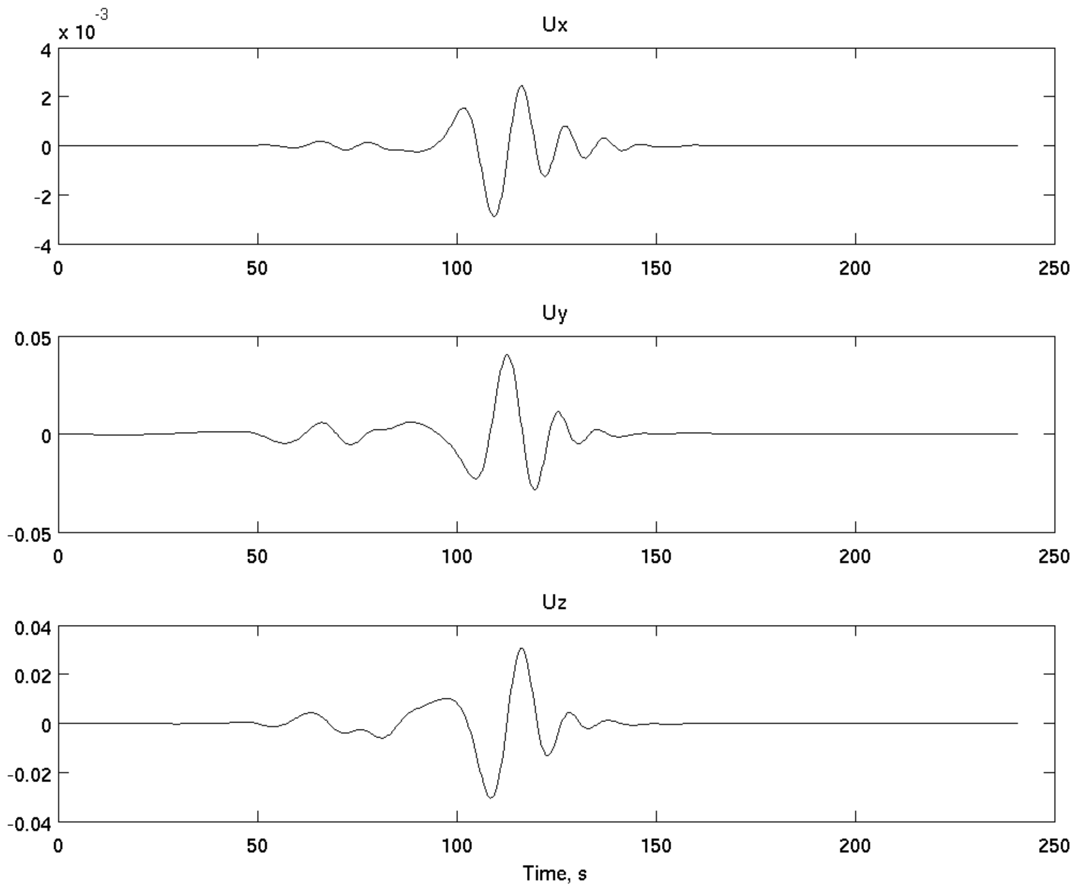

Figure 4: An example of the waveforms (displacements) at a surface receiver. The waveforms have been filtered between 0.02-0.1 Hz

References

Graves, R.W., 1996. Simulating seismic wave propagation in 3D elastic media using staggered-grid finite differences, *Bull. Seism. Soc. Am.*, **86**, 1091-1106.

Hixon, R., 1997. On increasing the accuracy of MacCormack schemes for aeroacoustic applications, *AIAA Paper*, **97**, 1586.

Igel, H., Nissen-Meyer T., & Jahnke G., 2002. Wave propagation in 3D spherical sections: effects of subduction zones, *Phys. Earth Planet. Inter.*, **132**, 219-234.

Moczo, P., Kristek, J., Vavrycuk, V., Archuleta, R.J. & Ladislav, H., 2002. 3D heterogeneous staggered-grid finite-difference modeling of seismic motion with volume harmonic and arithmetic averaging of elastic moduli and densities, *Bull. Seism. Soc. Am.*, **92**, 3042-3066.

Pitarka, A. & Irikura, K., 1996. Modeling 3D surface topography by finite-difference method: Kobe-JMA station site, Japan, case study, *Geophys. Res. Lett.*, **23**, 2729-2732.

Pitarka A., 1999. 3D elastic finite-difference modeling of seismic motion using staggered grids with nonuniform spacing. *BSSA*, **89(1)**:54-68

Shapiro, N.M. & Ritzwoller, M.H., 2002. Monte-Carlo inversion for a global shear velocity model of the crust and upper mantle, *Geophys. J. Int.*, **151**, 88-105.

Zhang, W. & Chen, X.F., 2006. Traction image method for irregular free surface boundaries in finite difference seismic wave simulation, *Geophys. J. Int.*, **167**, 337-353

Zhang, W., 2006. Finite Difference Seismic Wave Modelling in 3D Heterogeneous Media with Surface Topography and its Implementation in Strong Ground Motion Study, Phd Thesis (in Chinese), Peking University.

Zhang, W. & Shen, Y., 2010. Unsplit Complex Frequency Shifted PML Implementation Using Auxiliary Differential Equation for Seismic Wave Modeling. *Geophysics*, **75**, 141-154

Zhang, W., Shen, Y., & Zhao, L., 2011. Three-dimensional anisotropic seismic wave modeling in spherical coordinate by a collocated-grid finite difference method, *Geophys. J. Int.*, in revision.