UNIVERSITÀ
DEGLI STUDI DI TRIESTE

dipartimento di matematica e geoscienze

The DMG Quick Reference Manuals

1D Modal Summation Technique

Computation of modes, eigenfunctions, synthetic seismograms

QR

Document name: DMGQuick1Dmodes.pdf
Document Version: 1.3
Date Issued: 22 April 2018 at 19:56

Author: Franco Vaccari

University of Trieste
Department of Mathematic and Geosciences
Group of Seismology
via Weiss, 4
34128 Trieste, Italy

Table of Contents

Foreword

In the following, basic computations related with the modal summation technique for a 1D layered model are described. In the example, the model name is assumed to be **svalp**, so generated filenames refer to that. Change the model name in each input parameter file to whatever fit your needs, if you run your own tests. All the filenames will change accordingly.

Example input files

Required input files can be found in **/XDST/Examples/1DModesExamples**. Two example input datasets (**1Hz/Base** and **10Hz/Base** directories) are given, for 1 Hz and 10 Hz cutoff frequency computations. You should copy the required files into a new directory dedicated to your own computations.

Here is what you find in the **Base** directory for 10 Hz computations:

```
-rw-r--r--  1 vaccari  dstguest   253B Nov  4 14:25 eigl.par
-rw-r--r--  1 vaccari  dstguest   258B Nov  4 14:25 eigr.par
-rw-r--r--  1 vaccari  dstguest   1.2K Nov  4 14:25 eparatest.par
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.50
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.51
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.52
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.53
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.54
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.55
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.56
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.57
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.58
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.59
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.60
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.61
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.62
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.63
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.64
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.65
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.66
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.67
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.69
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.70
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.71
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.72
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.73
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.74
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.75
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.76
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.77
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.78
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.79
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.80
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.81
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.82
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.83
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.84
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.85
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.86
-rw-r--r--  1 vaccari  dstguest   416K Nov  4 14:25 guphas090.87
-rw-r--r--  1 vaccari  dstguest   511B Nov  4 14:25 gusev01.xy
-rw-r--r--  1 vaccari  dstguest   642B Nov  4 14:25 gusev02.xy
-rw-r--r--  1 vaccari  dstguest   581B Nov  4 14:25 gusev03.xy
-rw-r--r--  1 vaccari  dstguest   726B Nov  4 14:25 gusev04.xy
-rw-r--r--  1 vaccari  dstguest   707B Nov  4 14:25 gusev05.xy
-rw-r--r--  1 vaccari  dstguest   817B Nov  4 14:25 gusev06.xy
-rw-r--r--  1 vaccari  dstguest   890B Nov  4 14:25 gusev07.xy
-rw-r--r--  1 vaccari  dstguest   1.1K Nov  4 14:25 gusev08.xy
-rw-r--r--  1 vaccari  dstguest   1.1K Nov  4 14:25 gusev09.xy
-rw-r--r--  1 vaccari  dstguest   974B Nov  4 14:25 gusev10.xy
-rw-r--r--  1 vaccari  dstguest   844B Nov  4 14:25 modes.par
-rw-r--r--  1 vaccari  dstguest   844B Nov  4 14:25 p5r.par
-rw-r--r--  1 vaccari  dstguest   682B Nov  4 14:25 svalp.stp
```

The only addition in the set of files for 1 Hz computations is the file with the reference structure `ref1hz.str` used to complete at depth the shallow part of the structure defined in file `svalp.stp`:

```
-rw-r--r--  1 vaccari  dstguest   9.6K Nov  4 14:25 ref1hz.str
```

PostScript files

PostScript files generated at any step can be:

- visualized on screen with the *gs* or *gv* commands
- converted to pdf format with the *ps2pdf* command
- printed with the *lpr* command

To see the list of PostScript files in the working directory, sorted by modification time, you can give the command:

```
ls -t *.ps
```

To convert a single PostScript file into a PDF file give le command:

```
ps2pdf <filename>.ps
```

To convert multiple PostScript files into the corresponding PDF files give le command:

```
mps2pdf <file1>.ps <file2.ps>
```

To convert all PostScript files into the corresponding PDF files give le command:

```
mps2pdf *.ps
```

Computation of normal modes with p5r.out, ray and lov programs

The generation of the normal modes for a given structural model is the prerequisite for the computation of the synthetic seismograms.

The programs *ray* and *lov* that generate the modes for a given structural model, require a fairly complicated definition of the structural properties, stored in a file with the `.str` extension.

That file could be manually prepared by the user, but it is much more convenient to describe the structural properties in a simpler format (stored in a file with `.stp` extension) and let program p5r.out prepare the `.str` file.

Setting up a dedicated directory

Create a directory dedicated to the computations and copy there the required input files. For instance:

```
mkdir -p 1DModes/10Hz
```

copy into it the required input files:

```
cd 1DModes/10Hz
cp /XDST/Examples/1DModesExamples/10Hz/Base/* .
```

Relevant input files

To generate the modes for a given layered anelastic structural model, the following files are required:

`p5r.par`	Parameter file for program p5r.out
`ref1hz.str`	Definition of layer properties below the layers defined in svalp.stp (optional)
`svalp.stp`	Definition of layer properties

File `ref1hz.str` is not needed if the depth reached by the structure defined in file `svalp.stp` is already satisfying the depth condition for the layers definition: at least 80 km in depth must be defined for 10 Hz computations (100 km would be even better), and about 1100 km for 1 Hz computations. In the example, it is only needed for 1 Hz computations.

Main output files

The main output files of programs *lov* and *ray* are

`svalp.spl`	Love modes
`svalp.spr`	Rayleigh modes
`svalp.spl.pri`	debug messages for Love modes
`svalp.spr.pri`	debug messages for Rayleigh modes

Steps for the generation of the normal modes

1) Definition of the shallow part of the structure

Here is the content of the example file `svalp.stp`. with the definition of the layer properties:

```
  thk(km) rho   Vp(km/s)  Vs(km/s)    Qp       Qs     depth(km) layer
   2.0000 2.80  4.800000  2.400000   400.00   200.00    2.00000   1
   2.0000 2.83  5.800000  3.300000   400.00   200.00    4.00000   2
   2.0000 2.84  6.200000  3.500000   400.00   200.00    6.00000   3
   8.0000 2.85  5.700000  3.300100   400.00   200.00   14.00000   4
   2.0000 2.86  6.250000  3.550000   400.00   200.00   16.00000   5
  21.0000 2.87  6.500000  3.700000   400.00   200.00   37.00000   6
   3.0000 2.90  7.000000  4.000000   400.00   200.00   40.00000   7
   3.0000 3.10  7.500000  4.300000   400.00   200.00   43.00000   8
  40.0000 3.35  8.100000  4.500000   400.00   200.00   83.00000   9
```

⚠️➡️***Important notes for newly created structures!!!***⬅️⚠️

- The structure should reach at least 80 km in depth for 10 Hz computations (100 km would be even better), and about 1100 km for 1 Hz computations
- The Vs of the bottom layer should be at least 4.5 km/s for 10 Hz computations (4.7 km/s would be even better) and about 6.4 km/s for 1 Hz computations
- Wherever there is a low-velocity channel in the Vs, it must be defined for Vp as well, in the same depth interval.

2) Preparation of the parameter file for structure generation

Edit file `p5r.par` so that the file `svalp.stp` is used:

```
Parameters file for program p5r (v0001)
0                               reference structure (0 = none)
10.                             max frequency (10 Hz or 1 Hz or 0.1 Hz)
4.50                            min velocity for halfspace (1Hz=6.42,10Hz=4.50)
0.                              max depth (0=use vel; x=km,if in channel stay above)
1                               split more at physical interfaces (0=no, 1=YES)
1                               add num gradient to num layers (0=no, 1=YES)
1                               take care of low-velocity channels (0=no, 1=YES)
2.0                             largest Vs to split using wavelength (sugg. 2.0)
1.0                             wavelength correction factor (suggested 1.0)
svalp.stp                       physical layers first structure
```

Usually, for computations carried on with a cutoff frequency of 10 Hz, `.stp` file contains all the layers of the structure under consideration, while for 1 Hz computations only the uppermost part of the structure is defined in `.stp` file, and the deeper part of the structure is given in the so called “reference structure”, whose filename is given at the beginning of `p5r.par` file. Reference structure has to be prepared in `.str` file format with a preliminary run of *`p5r.out`*.

More than one structure can be prepared in a single run of *`p5r.out`*. You just have to add records at the bottom of file `p5r.par`, each with a `.stp` file definition.

⚠️➡️Be careful not to leave empty records at the bottom of `p5r.par` file!⬅️⚠️

In some cases the definition of parameters of file `p5r.par` can be difficult and may require many trials. These problems can be avoided by using a program (*`modes.out`*) that tests reasonable combinations of `p5r.par` parameters and runs *`ray`* and *`lov`* automatically. Details about *`modes.out`* are given in the next chapter.

3) Preparation of structure file in .str format

Run command

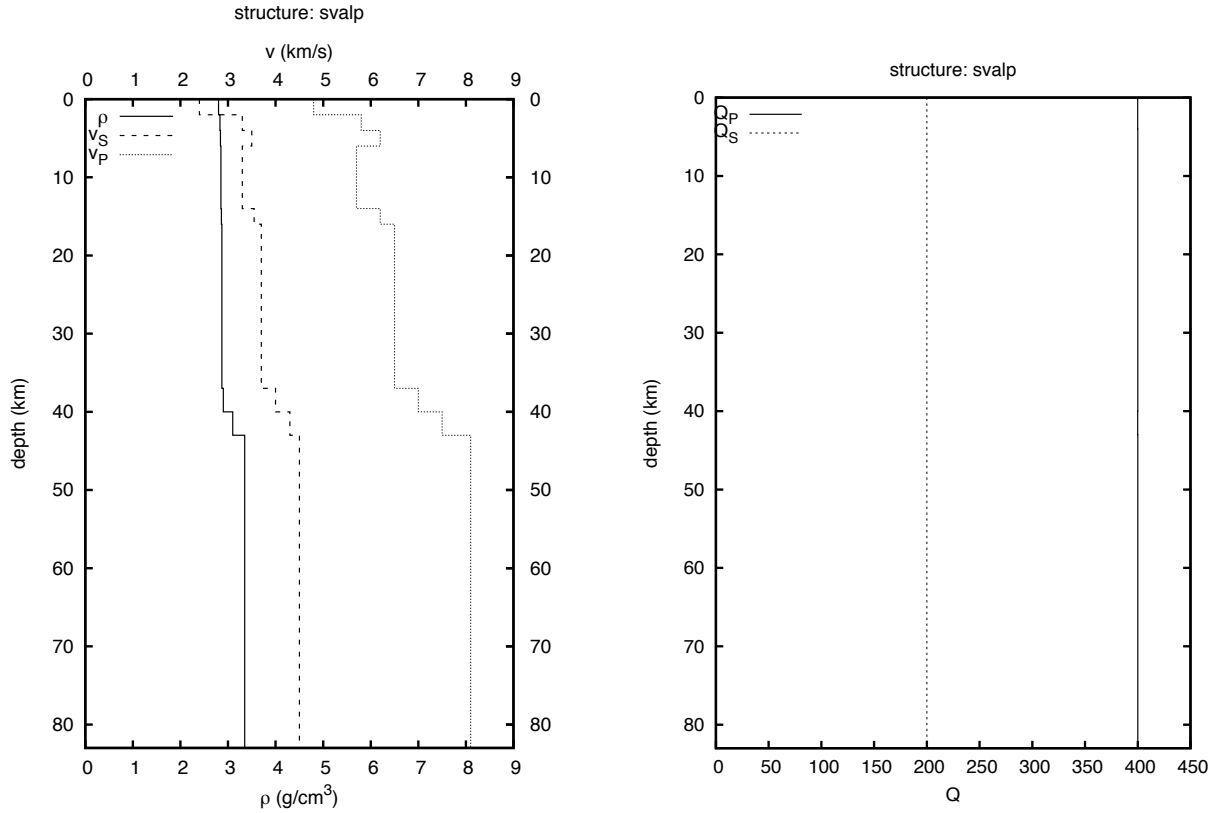
p5r.out

that will generate the file `svalp.str`, required by programs *lov* and *ray* to compute the modes.

4) Plotting of the structure

You can plot the structure (density and velocities vs depth and quality factors for P and S wave vs depth) issuing any of the following commands

```
plotstructure svalp.stp
plotstructure svalp.str
plotstructure svalp.spl
plotstructure svalp.spr
```



Files `svalp.str.ps` and `svalp.q.ps` are created that can be printed or viewed. You can redefine the depth range and/or the density and velocity ranges, or use colors in the plots using options of *plotstructure*.

The correct syntax is:

plotstructure [-options] [<minimum depth (km) to plot>-<maximum depth (km) to plot>] <input files>

Input files can be in stp, str, spl or spr format. Plot of file .stp may differ from the others as program p5r.out may, under some circumstances, limit the depth of the structure. So it's usually preferable to plot .str file to see the maximum depth actually used for the computation of modes.

The options are:

-c*f* convert from ps format to format *f*, where *f* means:
- b bmp
- e eps
- f pdf
- j jpg
- g png
- G png (transparent)
- m ppm
- t tif

-C plot with color
-dNNN define density (dpi) of raster format, NNN is the value to use
-h print help
-n no title

Examples:

1. Plot structure svalp.stp with colors in ps and pdf files:

   ```
   plotstructure -C -cf svalp.str
   ```

   Names of output files are svalp.str.ps, svalp.q.ps and svalp.str.pdf, svalp.q.pdf.

2. Plot structure svalp.str with colors from 0 to 50 km of depth:

   ```
   plotstructure -C 0-50 svalp.str
   ```

   Names of output files are svalp.str.0.50.ps, svalp.q.0.50.ps.

5) Generation of the normal modes

Run programs *lov* and *ray* to generate the modes for Love and Rayleigh waves, respectively.

```
lov
ray
```

Modes are stored in files .spl and .spr for Love and Rayleigh waves, respectively.

6) Plotting of the modes

The modes must be plotted to check that all modes have been properly computed:

```
plotmodes svalp.spl        (plot Love modes)
plotmodes svalp.spr        (plot Rayleigh modes)
```

The first command generates the PostScript file `svalp.spl.ps` with the plot of the Love modes, the second command generates the file `svalp.spr.ps` with the Rayleigh modes.

```
gs svalp.spl.ps        (visualize on screen Love modes)
gs svalp.spr.ps        (visualize on screen Rayleigh modes)
```

You can redefine the frequency range and/or the phase velocity and/or number of modes ranges using options of plotmodes script. The correct syntax is:

```
plotmodes [-options] [name_of_var=value_of_var] [<first modes to plot>-
<last modes to plot>] <input files>
```

List of options:

-a alternate colors for consecutive modes
-cf convert from ps format to format f, where f means:
b bmp
e eps
f pdf
j jpg
g png
G png (transparent)
m ppm
t tif
-dNNN define density (dpi) of raster format, NNN is the value to use
-e plot ellipticity in additional file
-h print help
-i plot iend in additional file
-l use logscale for x axis
-q plot qx instead of c2
-t use period instead of frequency
--N define line color, default is red, 0 means black

Max and min values of y and x axis can be set by this variables:

- Fmin Fmax: frequency
- Tmin Tmax: period
- Cmin Cmax: phase velocity
- Umin Umax: group velocity
- Imin Imax: energy Integral
- C2min C2max: phase attenuation
- Qmin Qmax: Q
- Zmin Zmax: depth of structure (for iend)
- Vmin Vmax: velocity of P and S waves (for iend)

Examples:

1. Plot Rayleigh modes from file svalp.spr with alternate colors for consecutive modes with ellipticity:

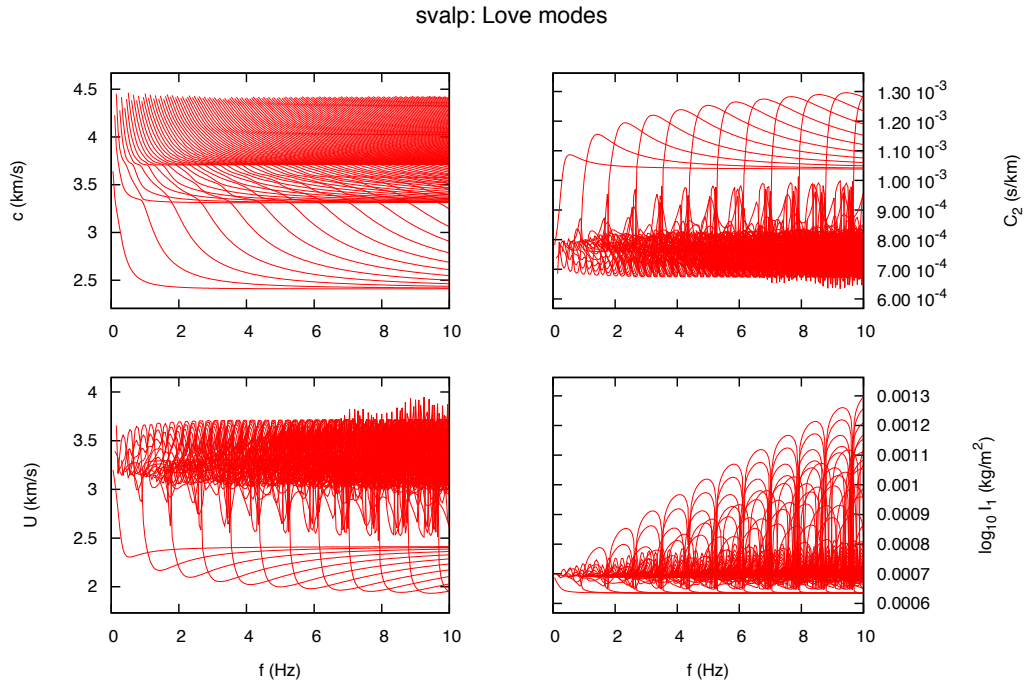
   ```
   plotmodes -a -e svalp.spr
   ```

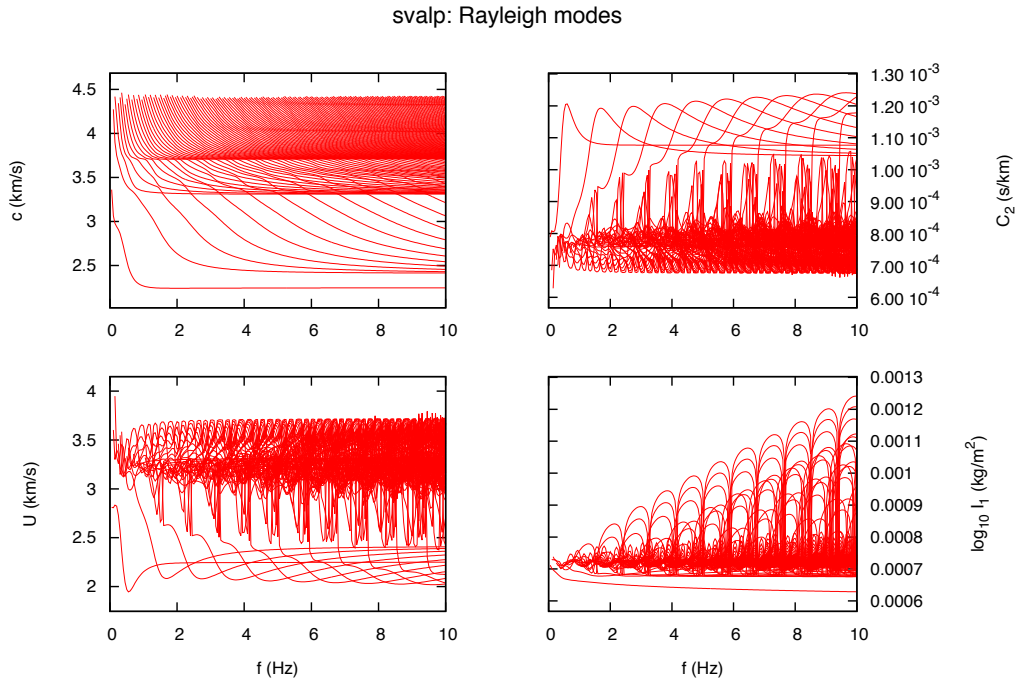
   Names of output files are *svalp.spr.ps*, *svalp.spr.ell.ps*.

2. Plot modes from 13° to 20° up to 4 Hz:

   ```
   plotmodes Fmax=4 13-20 svalp.spr
   ```

   Name of output file is svalp.spr.013.020.ps.





6) Comparing modes

You can compare modes of different structures (named for example svalp and new) using command:

```
overplotmodes svalp.spl new.spl
```

Syntax of this command is similar to the syntax of plotmodes:

```
overplotmodes [-options] [name_of_var=value_of_var] (<first modes to
plot>-<last modes to plot>) <input files>
```

List of options:

-a alternate colors for consecutive modes
-cf convert from ps format to format f, where f means:
- b bmp
- e eps
- f pdf
- j jpg
- g png
- G png (transparent)
- m ppm
- t tif

-dNNN define density (dpi) of raster format, NNN is the value to use
-e plot ellipticity in additional file
-h print help
-i plot iend in additional file
-l use logscale for x axis
-q plot qx instead of c2
-t use period instead of frequency
--N define line color, default is red, 0 means black

Max and min values of y and x axis can be set by this variables:

- Fmin Fmax: frequency
- Tmin Tmax: period
- Cmin Cmax: phase velocity
- Umin Umax: group velocity
- Imin Imax: energy Integral
- C2min C2max: phase attenuation
- Qmin Qmax: Q
- Zmin Zmax: depth of structure (for iend)
- Vmin Vmax: velocity of P and S waves (for iend)

Name of output file can be choose with "plotname" variable

Examples:

1. Plot Rayleigh modes from file svalp.spr and new.spl with alternate colors for consecutive modes with ellipticity:

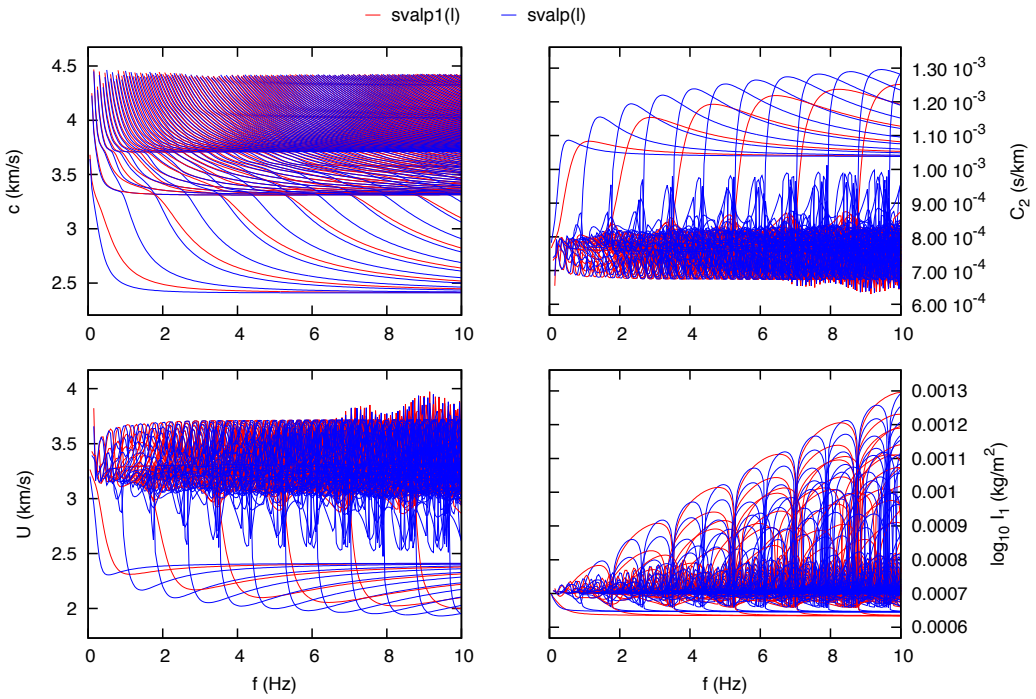
   ```
   overplotmodes -a -e svalp.spr svalp1.spl
   ```

   Names of output files are overplotmodes.ps, overplotmodes.ell.ps.

2. Plot modes from 13th to 20th up to 4 Hz:

   ```
   overplotmodes plotname=compare 13-20 svalp.spr new.spr
   ```

   Names of output files is compare.013.020.ps.


svalp1(l)
svalp(l)
c (km/s)
U (km/s)
f (Hz)
f (Hz)
C_2 (s/km)
$\log_{10} I_1$ (kg/m^2)

Computation of normal modes with modes.out

Problems in definition of `p5r.par` can be avoided by using *`modes.out`*, that tests reasonable combinations of `p5r.par` parameters and runs *`ray`* and *`lov`* automatically.

Required input files

To generate the modes for a given layered anelastic structural model, the following files are required:

`modes.par`	Parameter file for program modes.out
`ref1hz.str`	Definition of layer properties below the layers defined in svalp.stp (optional)
`svalp.stp`	Definition of layer properties

File `ref1hz.str` is not needed if the depth reached by the structure defined in file `svalp.stp` is already satisfying the depth condition previously mentioned (i.e. the structure should reach at least 80 km in depth for 10 Hz computations (100 km would be even better), and about 1100 km for 1 Hz computations). In the example, it is only needed for 1 Hz computations.

Output files

The final output files of program *`modes.out`* will be in the folder `10.0Hz`, if modes are computed at 10 Hz, or in folder `1.0Hz`, if modes are computed at 1 Hz. Contents of result folder are:

`modes.par`	Parameter file of modes.out
`svalp.stp`	Definition of layer properties
`svalp.spl`	Love modes
`svalp.spr`	Rayleigh modes
`svalp.stp.p5r.par`	Final parameter file used for program p5r.out

Generation of the normal modes with modes.out

Edit file `modes.par` so that the file `svalp.stp` is used:

```
Parameter file for program modes (v0002)
----------------------------------------
0                   Reference Structure (0=no)
10                  Max Hz
0                   Max depth (km, 0=use Vs)
svalp.stp
```

Usually, for computations carried on with a cutoff frequency of 10 Hz, `.stp` file contains all the layers of the structure under consideration, while for 1 Hz computations only the uppermost part of the structure is defined in `.stp` file, and the deeper part of the structure is given in the so called “reference structure”, whose filename is given at the beginning of `modes.par` file. Reference structure has to be prepared in `.str` file format with a preliminary run of *`p5r.out`*.

More than one structure can be prepared in a single run of *`modes.out`*. You just have to add records at the bottom of file `modes.par`, each with a `.stp` file definition.

Plotting of eigenfunctions

After the files with the modes have been generated, it is easy to generate and plot the eigenfunctions associated with selected eigenvalues. You can specify the modes and the frequencies for which the eigenfunctions are computed and plotted.

This is an optional step, not required for the generation of synthetic seismograms. It can teach you about the range of modes that will be excited by a given source place at a given depth in the structure.

Required input files

`svalp.spl`	Modes for Love waves
`svalp.spr`	Modes for Rayleigh waves

Plotting the eigenfunctions

Script checkEig computes and plots the eigenfunctions starting from spectrum (spl or spr) file. The correct syntax is:

checkEig [-options] [name_of_var=value_of_var] m=<first modes to plot>-<last modes to plot> f=<first frequency to plot>-<last frequency to plot> <input files>

Input files must be a Love or Rayleigh spectrum. Name of the output file has this format:

`<root name of input file>.<first mode to plot>-<last mode to plot>.eig[rl].ps`

List of options:

-B grey-scale plot
-b plot Ray and Love eigenfunctions
-cf convert from ps format to format f, where f means:
- b bmp
- e eps
- f pdf
- j jpg
- g png
- G png (transparent)
- m ppm
- t tif

-dNNN define density (dpi) of raster format, NNN is the value to use
-i plot iend
-l use logscale for x axis in iend plot
-t use period instead of frequency in iend plot
-h print help
-n no title

Useful variables that can be set:

iend	defines iend for eigenfunction computation
depth	defines maximum depth for eigenfunction computation
Zmax, Zmin	maximum and minimum depths to plot
Fmax, Fmin	maximum and minimum frequency to plot (for iend)

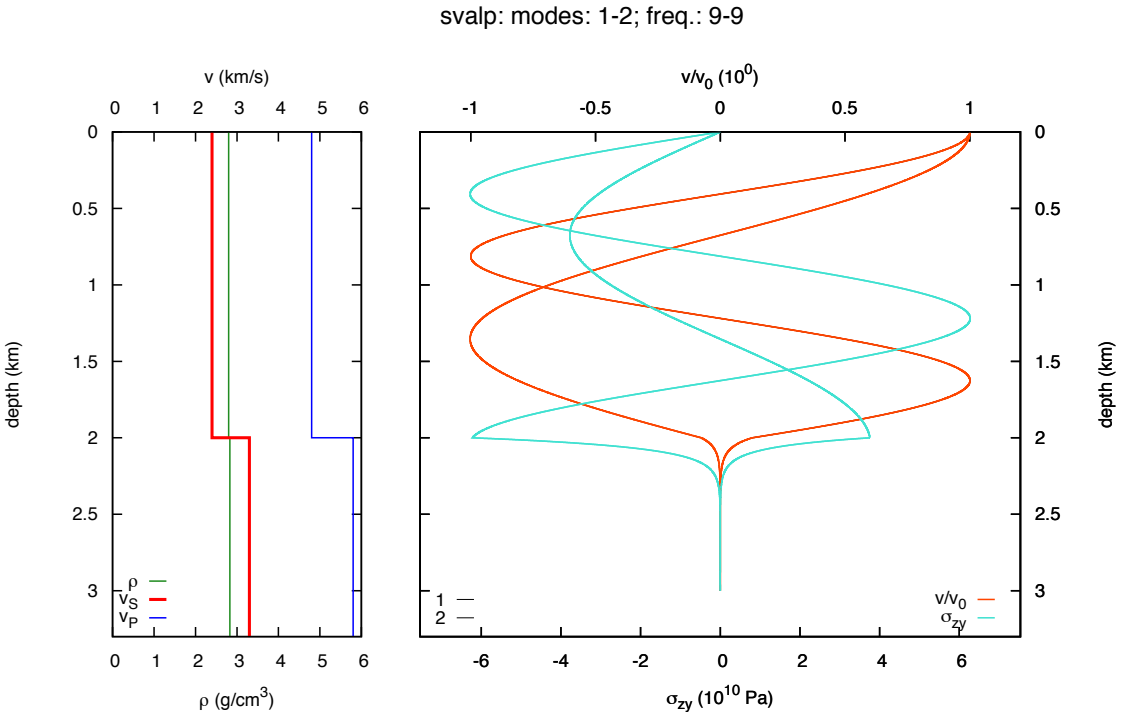
Examples:

1. Plot eigenfunctions of fundamental Rayleigh mode from 1.0 to 1.2 Hz from file svalp.spr:

```
checkEig m=1-2 f=9-9 svalp.spl
```

2. Plot eigenfunctions of fundamental Rayleigh and Love modes from 1.0 to 1.2 Hz from file svalp.spr and svalp.spl:

```
checkEig -b m=0-0 f=1-1.2 svalp.spr
```



Computation of synthetic seismograms in 1D layered structural models, with parametric tests

Once the files with the modes have been generated for a given layered anelastic model, it is easy to generate synthetic seismograms for user-defined source configurations. It is generally wise to perform parametric tests in order to check the influence of each parameter on the waveform.

Required input files

`eparatest.par`	Parameters defining the test
`svalp.spl`	Modes for Love waves
`svalp.spr`	Modes for Rayleigh waves
`gusev*.xy`	10 files with Gusev source functions for scaling
`guphas090.*`	38 files with Gusev source functions for scaling (phase included)

Steps to perform the parametric tests

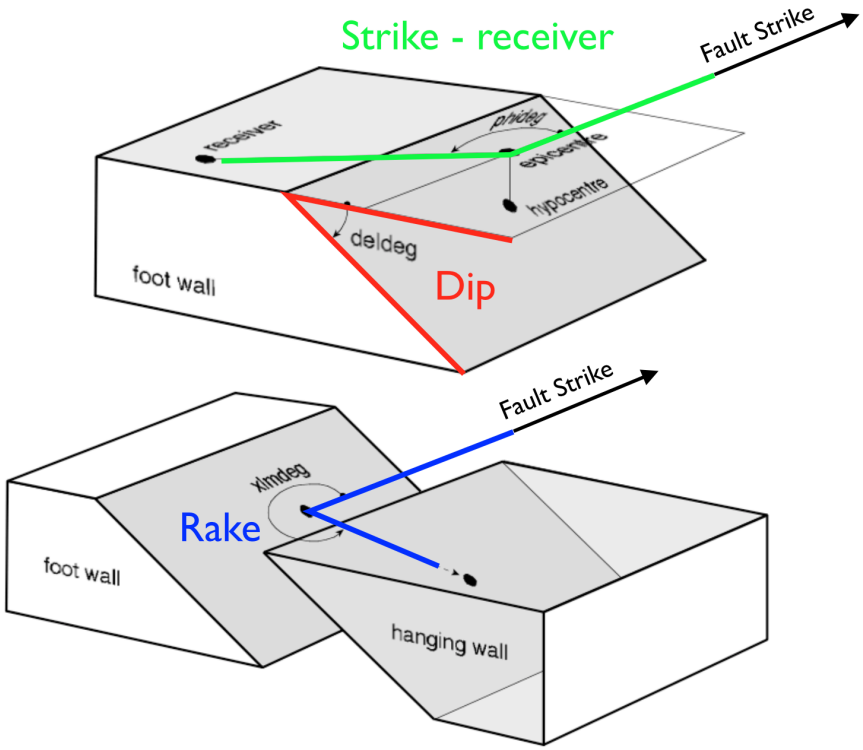
1) Preparation of the parameter file

File `eparatest.par`, shown below, contains all the parameters that can be changed by the user to configure the experiment.

```
Parameter file for program eparatest
--------------------------------------------------------------------------------
svalp                 Test label (root for output filenames - 13 chars max)
0                     Ref. box for values not listed below (0=no, 13 chars max)
svalp.spl             Love spectrum file
svalp.spr             Rayleigh spectrum file
2                     Motion (1=displ, 2=vel, 3=acc)
50                    Time length for plot seismograms (s)
1  13.0 45.0 80       Source (1=point, 2=extended), lon, lat, strike (cw from North)
SRE 1   0 360 15      Strike (loop 0=no,1=yes, start, stop, step) (Degrees)
DIP 0  30  90 10      Dip (loop 0=no,1=yes, start, stop, step) (Degrees)
RAK 0  10  40 10      Rake (loop 0=no,1=yes, start, stop, step) (Degrees)
SDE 0   7   9  1      Source Depth (loop 0=no/1=yes, start, stop, step) (km)
EDI 0  15 200 15      Epic. Distance (loop 0=no/1=yes, start, stop, step) (km)
RDE 0   0   3  1      Receiver Depth (loop 0=no/1=yes, start, stop, step) (km)
MOD 0   0   0  1      Modes (loop 0=no/1=yes, start, stop (step must be 1) )
INT 0   1  30  1      Interpolation (0-9) (flag 0=no,1=yes, start, stop, step)
MAG 0 6.5 7.0 .1      Magnitude (flag 0=no,1=yes, start, stop, step)
```

Parameters that can be varied:

SRE	strike/receiver angle (°)
DIP	fault dip angle (°)
RAK	fault rake angle (°)
SDE	source depth (km)
EDI	epicentral distance (km)
RDE	receiver depth (km)
MOD	modes to use (0 0 means all)
INT	interpolation in frequency domain (n. of samples to add between two samples)
MAG	magnitude



To mark a single parameter as variable in an experiment, a value of 1 has to be placed in the first numeric field associated with it. In the above example, the parametric test is carried on the strike/ receiver angle (SRE) and the parameter will be varied between 0° and 360° with a step of 15°. All the other parameters (DIP, RAK, etc) will assume a fixed value, that is the value defined in the second numeric field (ie: DIP 30°, RAK 10° etc).

⚠️➡️***Important notes***⬅️⚠️

- For a reasonably well looking plot of the waveforms, please configure your experiment so that 10 to 30 steps are used to explore the parameter's space
- At most three parameters can be activated for looping in a single run of program *`eparatest.out`*. This is done by placing a 1 2 or 3 in the first numeric field associated with them. 1 will be the outer loop, 3 the inner loop for parameter variation. All other parameters must have a 0 in the first numeric field. It is anyway suggested to explore one parameter at a time
- Under most circumstances, there is no need to define a reference box.

2) Generation of the script

Preliminary to the computation of the synthetic seismograms is the preparation of the shell script (named `eparajob`) that will call the sequence of programs performing the actual computations.

To create the shell script simply run the command:

```
eparatest.out
```

The script generated will look like:

```
#!/bin/sh
date>svalpparajob.pri
echo "Start of parajob job">>svalpparajob.pri
cp syr.cntl.r syr.cntl
echo "Computing Radial Component..."
syr0048.out
date>>svalpparajob.pri
echo "Radial Computed">>svalpparajob.pri
cp syr.cntl.z syr.cntl
echo "Computing Vertical Component..."
syr0048.out
date>>svalpparajob.pri
echo "Vertical Computed">>svalpparajob.pri
cp syl.cntl.t syl.cntl
echo "Computing Transverse Component..."
syl0048.out
date>>svalpparajob.pri
echo "Transverse Computed">>svalpparajob.pri
date>>svalpparajob.pri
echo "Scaling seismograms..."
efft.out
date>>svalpparajob.pri
cat tmploop | awk '{ print $1,$2 }'  > tmploopsel
grep amaxa svalpf1.syr | awk '{ print $8 }'  > tmpr
grep amaxa svalpf1.syz | awk '{ print $8 }'  > tmpz
grep amaxa svalpf1.syl | awk '{ print $8 }'  > tmpl
echo '#num sre  PGV rad   PGV ver   PGV tra   dip=30.0 rak= 10.0 sde=  7.000 edi=
15.000 rde=  0.000\n\ mod=  0-  0 int= 1 mag=6.5' > svalp.sta
paste tmploopsel tmpr tmpz tmpl >> svalp.sta
minmax -C tmpr | awk '{ print $2 }' > tmpsort
minmax -C tmpz | awk '{ print $2 }' >> tmpsort
minmax -C tmpl | awk '{ print $2 }' >> tmpsort
MAX=`(cat tmpsort | sort -rn | head -1 )`
rm -f tmploop tmploopsel tmpr tmpz tmpl tmpsort *.plot
gnuplot svalp.sre.gplot
echo " "
sisplots.pl
echo "*** To plot the seismograms: gs svalp.sre.sis.*.ps"
echo "*** To plot the peak values: gs svalp.sre.ps"
echo " "
```

The script is configured so to generate the synthetic seismograms for the three components of motion (transverse, vertical and radial) and to produce (rough) plots of the waveforms and of the variation of the peak values as a function of the parameter value.

***Output files created by* `eparatest.out`**

A run of *`eparatest.out`* will generate the files listed below:

`eparajob`	shell script that will generate the seismograms and the plots
`fft.par`	parameter file for scaling the seismograms
`sisplots.par`	parameter file for plotting the waveforms
`svalp.boxr`	parameter file for program syr0048.out (for radial component seismograms)
`svalp.boxt`	input file for program syl0048.out (for transverse component seismograms)
`svalp.boxz`	input file for program syr0048.out (for vertical component seismograms)
`svalp.pri`	debug writings
`svalp.sre.gplot`	gnuplot script for plotting seismograms peak values
`syr.cntl.r`	control file for program syr0048.out (radial component seismograms)
`syl.cntl.t`	control file for program syl0048.out (transverse component seismograms)
`syr.cntl.z`	control file for program syr0048.out (vertical component seismograms)

Power users can look inside the above files for a deeper knowledge of what's going on.

3) Computation and plotting of the synthetic seismograms

The command that will execute the script is:

eparajob

The script calls the programs *syl0048.out* (for transverse component) *syr0048.out* (for radial and vertical components), *efft.out* for scaling the seismograms according to the magnitude.

***Output files created by* eparajob**

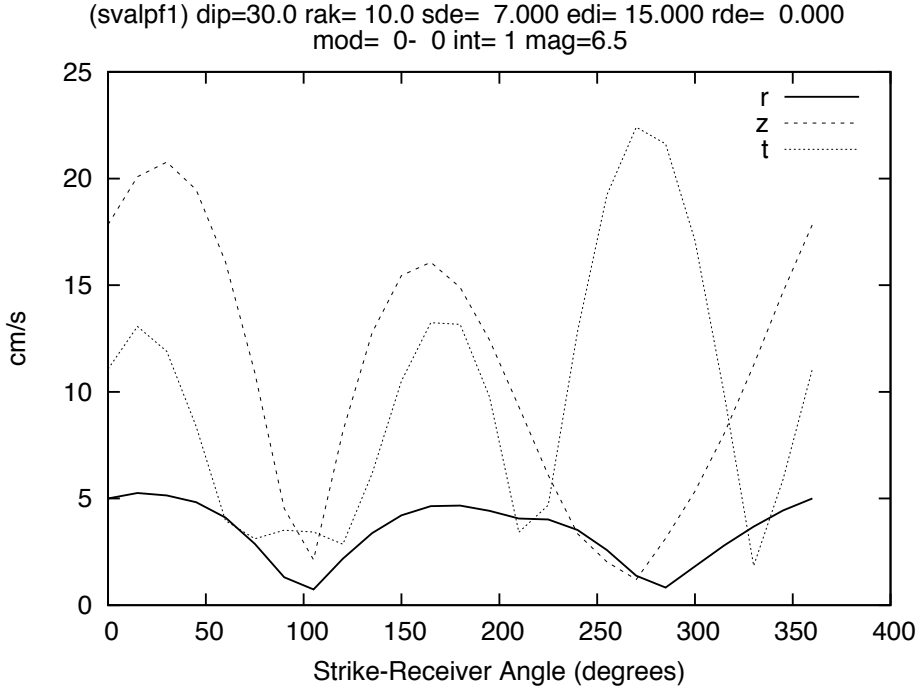
The execution of *eparajob* will generate the files listed below:

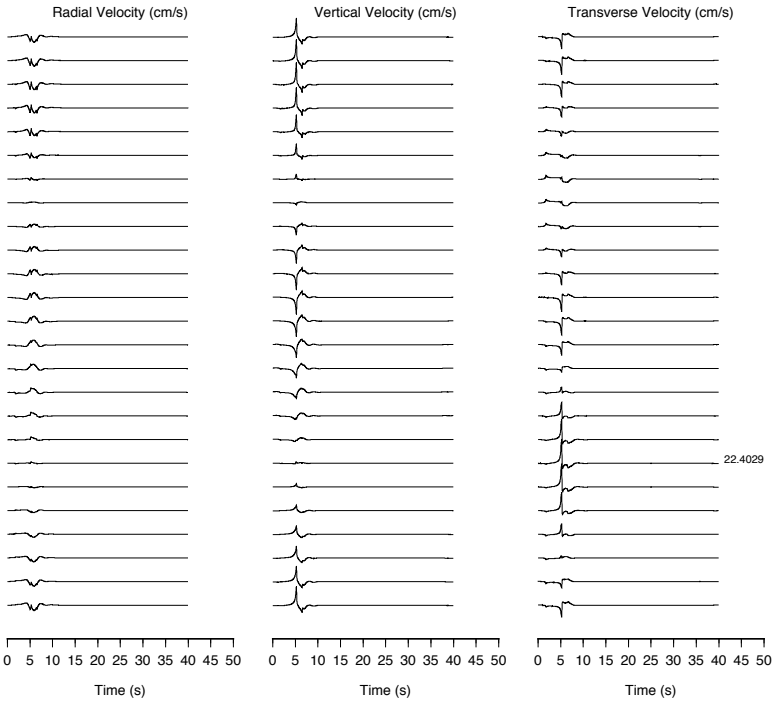
fft.pri	seismogram scaling debug writings
svalp.eil	eigenfunctions computed at the source depth (Love)
svalp.eir	eigenfunctions computed at the source depth (Rayleigh)
svalp.sre.ps	peak values plot
svalp.sre.sis.1.ps	waveform plot
svalp.sta	tabulated values for plotting peaks
svalp.syl	unscaled synthetic seismograms (transverse component)
svalp.syl.pri	debug writing for synth. seism. (transverse component)
svalp.syr	unscaled synthetic seismograms (radial component)
svalp.syr.pri	debug writing for synth. seism. (radial component)
svalp.syz	unscaled synthetic seismograms (vertical component)
svalp.syz.pri	debug writing for synth. seism. (vertical component)
svalpf1.syl	scaled synthetic seismograms (transverse component)
svalpf1.syr	scaled synthetic seismograms (radial component)
svalpf1.syz	scaled synthetic seismograms (vertical component)
svalpf1syl.frq	spectral peak values (transverse component)
svalpf1syl.sta	-- not relevant --
svalpf1syr.frq	spectral peak values (radial component)
svalpf1syr.sta	-- not relevant --
svalpf1syz.frq	spectral peak values (vertical component)
svalpf1syz.sta	-- not relevant --
svalpparajob.pri	job debug writings
syl.cntl	-- not relevant --
syr.cntl	-- not relevant --

4) Plotting of the results

PostScript files generated by gnuplot can be visualized issuing the *gs* commands suggested on the screen when *eparajob* is done. In the example

gs svalp.sre.ps	(plot of the peak value variation)
gs svalp.sre.sis..ps*	(plot of the waveforms)


(svalpf1) dip=30.0 rak= 10.0 sde= 7.000 edi= 15.000 rde= 0.000
mod= 0- 0 int= 1 mag=6.5
r
z
t
cm/s
0
5
10
15
20
25
0
50
100
150
200
250
300
350
400
Strike-Receiver Angle (degrees)


Radial Velocity (cm/s)
Vertical Velocity (cm/s)
Transverse Velocity (cm/s)
22.4029
0 5 10 15 20 25 30 35 40 45 50
Time (s)
0 5 10 15 20 25 30 35 40 45 50
Time (s)
0 5 10 15 20 25 30 35 40 45 50
Time (s)