Manual of Auto wave gradiometry method (AWGM)

V3.0

CAO Feihuang and LIANG Chuntao

Chengdu University of Technology

2023

DIRECTORY

1. Introduction and dependent software

This package aims to calculate the phase velocity and azimuth anisotropy of surface waves using the wave field spatial gradient ($du/dx, du/dy$) of seismic surface wave waves, and then to invert the three-dimensional velocity and anisotropic structure. The package uses seismic waveforms recorded in 2007 by the ChinArray (2006) as an example.

To use this package, you need to install MATLAB, CPS330 and OpenMPI, firstly.
Matlab: https://www.mathworks.com/products/matlab.html
CPS330: https://www.eas.slu.edu/eqc/eqccps.html （Linux）
OpenMPI: https://www.open-mpi.org/ （Linux）

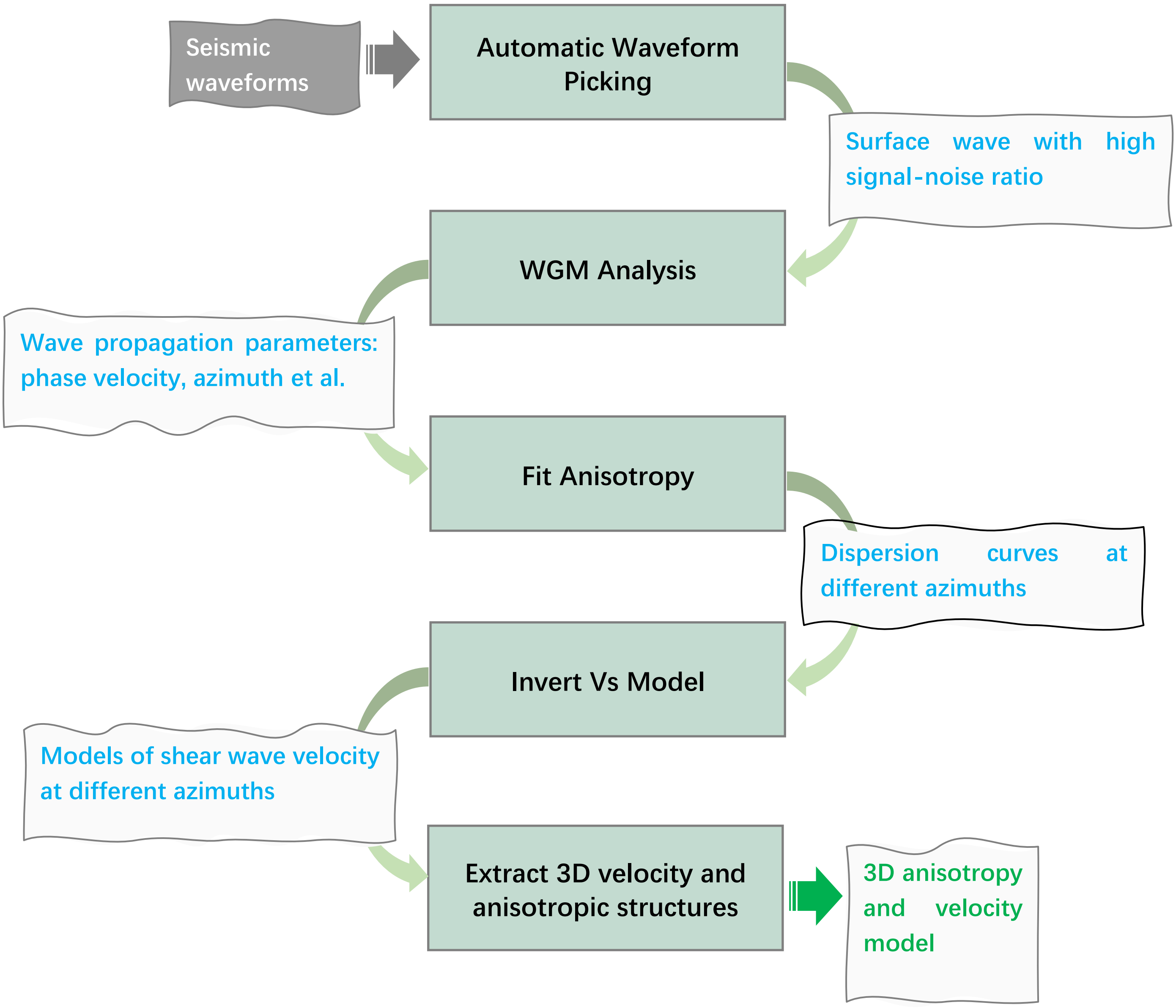

Figure 1. Workflow

2. AWGM for Surface Wave Imaging

2.1 Introduction of WGM

Wave Gradient Method (WGM) is a seismic data processing technique based on dense arrays. We can use a dense-seismic array to calculate the wave gradient accurately, and then fetch the propagation parameters of seismic wave in array (e.g., phase velocity, azimuth, geometric diffusion and radiation pattern). The basic theory of WGM has been detailedly described in previous articles (Liang and Langston, 2009; Cao et al. 2020; Liang et al. 2023; Cao and Liang, 2023). This package provided an automatic wave gradiometry method (AWGM) to compute the wave propagation parameters of surface wave.

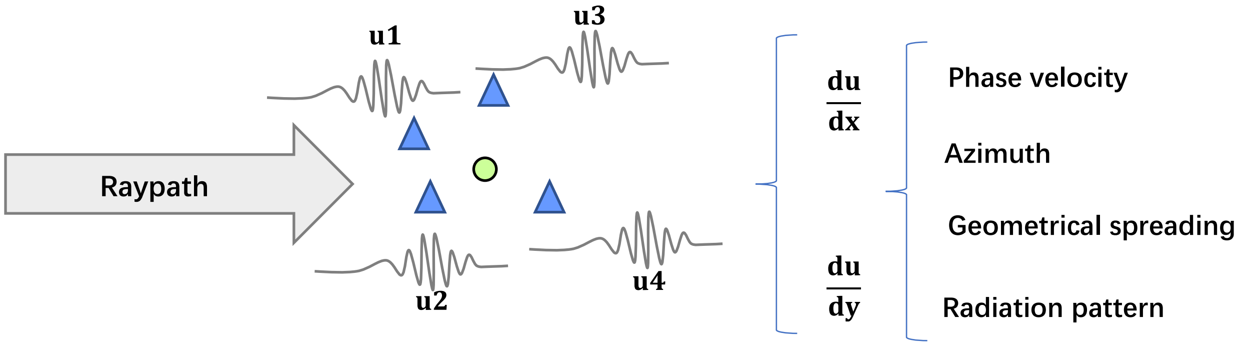

Figure 2. Schematic diagram of WGM on a subarray, where blue triangles represent auxiliary stations, and green dot is reference location. The waveforms (**u**) recorded by auxiliary stations are used to calculate the wave gradient, which directly related to the wave propagation parameters (Langston, 2007; Liang and Langston 2009）。

2.2 Data Preparation

(1) SAC data

The input data is event-based SAC files, and should be put in the folder name as YYYYMMDD.JDAY.hhmmss.ms. And the name of each SAC file should follow the rule of **net.station.YYYYMMDD. JDAY.hhmmss.ms.cmp.sac.** In order to isolate the surface wave, users can use the time window of **T0**(P wave arrival)-**T1**(P wave arrival+4000) to cut the waveforms from continuous recorded waveforms.

```
>>SACdat=readSAC('./20080415.106.073245.860/T1. T100X.20080415.106.073245.860.BHZ.sac')
```

sacdat=

DELTA:	1	**B:**	455.8929
E:	4. 456e+03	**T0:**	455.8929
STLA:	30	**STLO:**	104
STEL:	589	**EVLA:**	16.2260
EVLO:	144.9950	**EVDP:**	10
DIST:	4.4679e+03	**AZ:**	298.9282
BAZ:	101.8255	**NZYEAR:**	2008
NZJDAY:	106	**NZHOUR:**	7
NZMIN:	32	**NZSEC:**	45
NZMSEC:	860	**NPTS:**	4097
IZTYPE:	'IO'	**KSTNM:**	'T100X'
KCMPNM:	'BHZ'	**KNETWK:**	'T1'
DATA1:	[4097x1] ···		

(2) Station information file.

Station information save in the text file of **st.info**, and each column of the file from left to right corresponds to the station network name, station name, latitude, longitude, and altitude:

```
T1    T100X    30.00    104.00    589
```

2.3 Global parameter settings.

Most of the parameters of AWGM are placed in the Matlab scripts of **aa_WGM_parameters.m**.

Seismic network and reference location information

```
%% ----- information of seismic array and------------
par.stinfo = './st.info';                % Station Information File: Net Station stla stlo stevl
par.ev_sac_rootpath = '.. /ev_sac/';     %sac file path
par.lalim = [26 32.4];                   % Network Dimension Range [minimum latitude, maximum latitude].
par.lolim = [99.6 105.4];                % Grid longitude range [minimum latitude, maximum latitude].
par.component = 'BHZ';                   % component
par.Net='T1';                            % network name,
par.gridsize = 0.2;                      % degree, refer to the location grid scale
ddst=0. 7;                               % deg, Parameters that control the generation of grid points
```

Users can adjust parameters of **par.lalim, par.lolim, par.gridsize,** and **ddst** to modify the grid points (reference location). The reference location information is stored in the matlab file of **gstall.mat**. Figure 2 shows the effect of different **ddst** value on the generation of grid points.

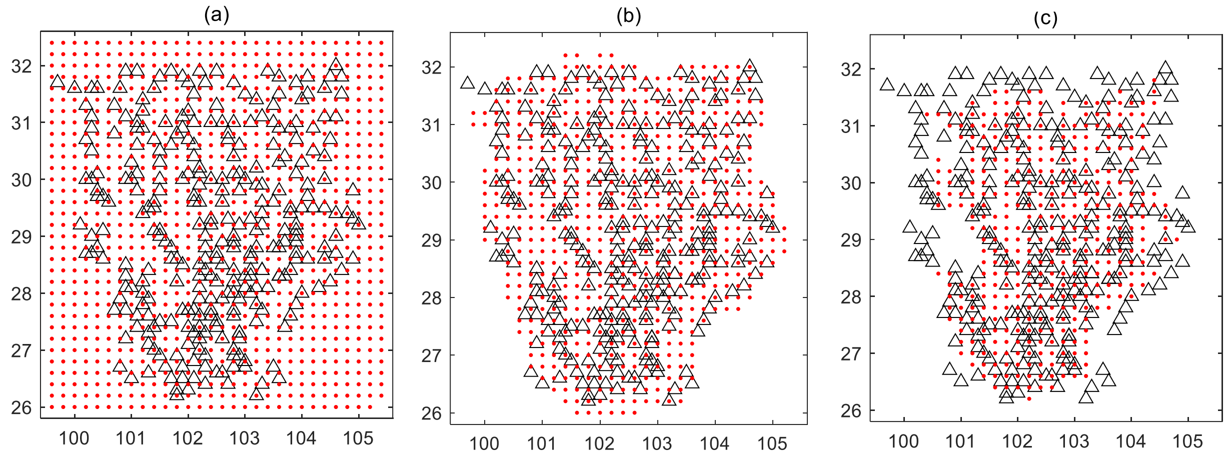

Figure 3: Distribution of ChinArray (2006; black triangle) and reference positions (red dots). (a) **ddst** = 1.9, with too large values resulting in extra grid points; （b） **ddst** =0.9. Grid points consistent with the distribution of the array; (iii) **ddst** = 0.45, a value that is too small to cause grid points to be missing in some areas.

Settings parameter for automatic waveform picking

```
%% -------- Auto waveforms pick --------
par.wvPick_output = './wvPick/';
par.periods = 10:2:80; % s, central periods s
par.ev_minDist = deg2km(2); % km, minimum epicentral distance
par.ev_maxDist = deg2km(160);    % km, maximum epicentral distance
par.ev_maxDepth = 100;    % km, maximum depth of earthquakes
par.vglen = [2,5];     % km/s, minimum and maximum Vg (km/s) for isolate Surface wave
par.dvg = 0. 2;            % km/, outlier of Surface wave group arrival
par. SNR=5;                % minimum SNR of one waveform
par.mSNR=8;                  % minimum mean(SNR>5) of one earthquake
```

par.wvPick_outpu: Output path

par.periods: Central period

par.ev_minDist: Stations with an epicenter distance of less than **par.ev_minDist** are automatically discarded

par.ev_maxDis: Stations with an epicenter distance greater than **par.ev_maxDis** are automatically discarded

par.ev_maxDepth: earthquakes with a focal depth greater than **par.ev_maxDepth** are automatically discarded

par. SNR: stations with low $SNR = AS/AN$ (SNR<**par. SNR**) are automatically discarded.

par.mSNR: For a period, earthquakes with low $mSNR = mean(SNR > 5)$ (mSNR<**par.mSNR)** are automatically discarded.

par.vglen: minimum and maximum group velocity, which are used to isolate surface wave (Figure 4b yellow shade).

par.dvg: upper and lower limits of group velocity vg, $[mean(vg) - \mathbf{par.dvg},\ mean(vg) + \mathbf{par.dvg}]$.

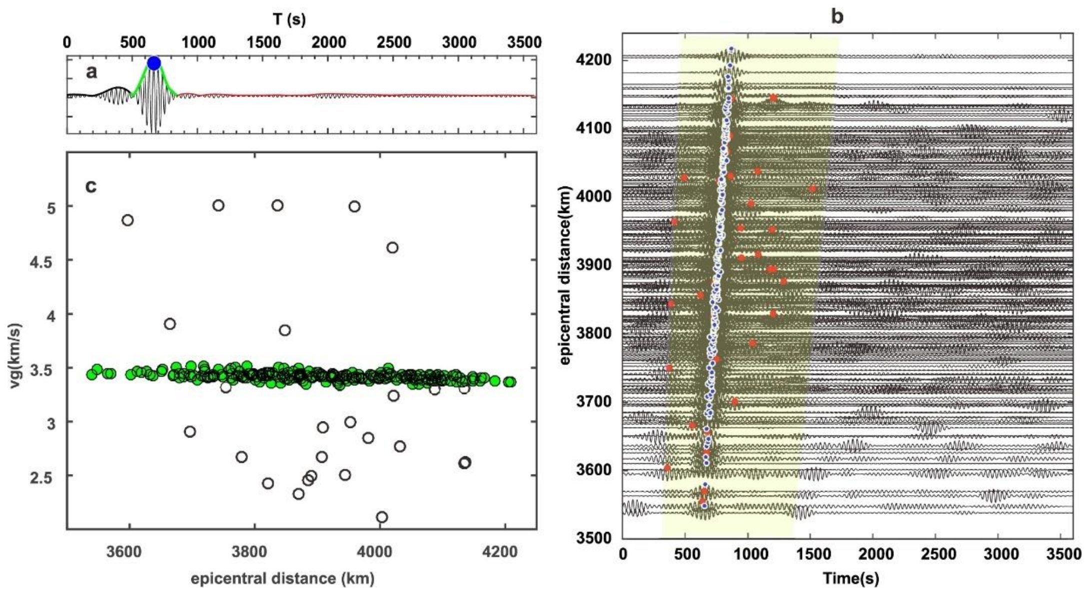

Figure 4. Automatically selecting for waveforms of Nevel'sk M5.5 earthquake (UTC: 2007-08-02 10:37:31). (a) The teleseismic waveform of one station with center period of 35 s. The blue dot marks the arrival time of the maximum amplitude of Rayleigh wave; (b) waveforms of Nevel'sk M5.5 earthquake (UTC: 2007-08-02 10:37:31), the blue and the red dots mark the arrival time of maximum amplitude of Rayleigh wave with good and bad quality, respectively; (c) vg distribution with epicentral distance, the green points and the black circles are the vg inside and outside the range of $[mean(vg) - \mathbf{par.dvg}, mean(vg) + \mathbf{par.dvg}]$.

Setting parameter for WGM

```
%% ------- 3. parameters of WGM --------
par. WG_output = './WG/';
par.minNST=6;              % minimum of stations number in a subarray (NSS)
par.maxstadist = 30;       % km, subarray maxR
par.maxdsmax = 150;        % km, maximum maxR when NSS<miNST
par.cutWin = 500;          % s, time window around the surface wave
par.dvmax = 0.01;          % km/s, min dv in the reducing velocity method
par. Rcof=0.7;             % minimum correlation coefficient between the predicted wavefield and
                           observed wavefield

% ----------- smoothing parameters -----------
par.vmodtype='ak135';              % ='ak135',reference phase velocity from AK135;
                                   %=[],user difine reference phase velocity,
par.v_modf='./VphaseMod.mat';      % when par.vmodtype=[], user needs to give reference-velocity file here
par.vlim=0.8;                      % outlier <median(v)-vlim or outlier > median(v)+vlim
prdb=[18 50 70];                   %period boundary for smoothing radius
smrv=[0.35 0.3 0.4 0.6];           %deg, smoothing radius of velocity
smra=[0.45 0.4 0.6 0.7];           %deg, smoothing radius of anisotropy
```

par. WG_output: Output path

par.minNST: minimum number of auxiliary stations in the subarray.

par.maxstadist: radius of subarray. The radius of the subarray is closely related to the stability and resolution of the gradient analysis. Theoretically, the resolution is higher in subarray with smaller radius, and more stations in subarray may generate more stable results. Generally, it is guaranteed that more than 5 auxiliary stations in a subarray, that is, **par.minNST** are set between 5~8.

par.maxdsmax: Maximum radius of subarray. When the number of auxiliary stations is not enough, the program will automatically increase the radius of subarray to include more station into subarray, but not greater than **par.maxdsmax**.

par.cutWin: parameter used to define a time window $(min\_t_{pick} - \mathbf{par.cutWin}, max\_t_{pick} + \mathbf{par.cutWin})$ for cutting surface wave. min_t_{pick} and max_t_{pick} indicate the earliest arrival and latest arrival of the surface wave recorded by the auxiliary stations in the subarray, respectively, and the t_{pick} are the time at maximum amplitude of the surface wave (blue dot in Figure 4). The larger one $\mathbf{par.cutWin}$ will cost more time in wave gradiometry analysis, and it is recommended to set it between 200 and 1000. Figure 5 shows the gradient analysis results of a subarray, with $\mathbf{par.cutWin} = 500$.

par.dvmax: parameters for iteration in reducing velocity method. The initial phase velocity of one reference location (grid point) is given by $C_0 = vg + 0.2$. Update the phase velocity C_i after i times iteration, when $|C_i - C_{i-1}| < \mathbf{par.dvmax}$, the script will stop iterating and save the results of one grid point.

par. Rcof: The minimum correlation coefficient between the measured waveforms and the predicted waveforms of auxiliary stations. Auxiliary station with correlation coefficient less than **par. Rcof** are discarded.

par.vmodtype: Reference phase velocity model for removing outlier. = [ ], user defined; ='AK135', using the reference phase velocity generated by the AK135 model

par.v_modf: when **par.vmodtype**=[ ], user needs to provide a file of reference phase velocity here.

par.vlim: grid points with phase velocity C < vmod-**par.vlim** or C > vmod-**par.vlim** are discarded.

prdb, **smrv, smra:** parameters for defining smoothing radius of velocity and anisotropy in different periods. **prdb** is the boundary of periods, and **smrv** and **smra** are the smoothing radius of phase velocities that used to fitting isotropy phase velocity and anisotropy, respectively (Figure 6).

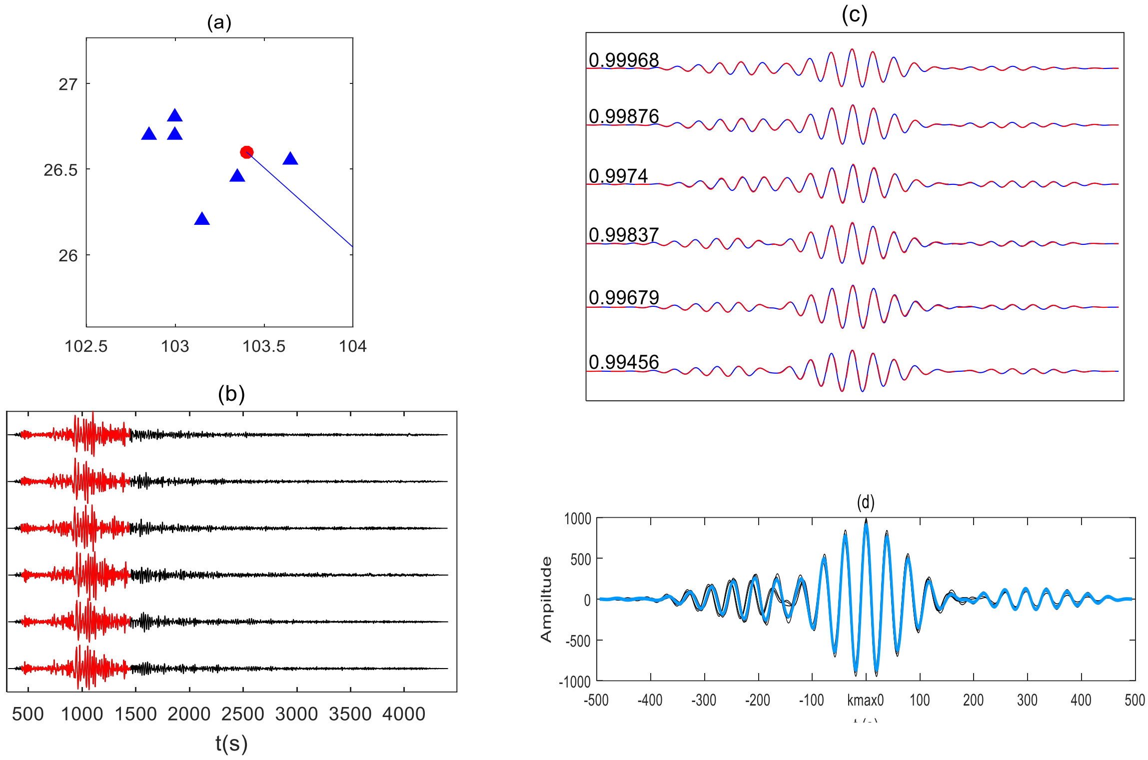

Figure 5: Reconstructed waveforms on a subarray. (a) distribution of subarray, blue triangle represents auxiliary stations, red dots represent reference location, and blue lines represent raypath. (b) Seismic waveforms recorded by auxiliary stations. Red waveforms represent **the surface wave** cut by time window of [$min\_t_{pick} - \mathbf{500s}, max\_t_{pick} + \mathbf{500s}$]. (c) Comparison of the difference between measured waveforms (blue) and predicted waveforms (red) of auxiliary station. (d) Comparison between reconstructed waveform in reference location (light blue) and the measured waveforms of the auxiliary station (black).

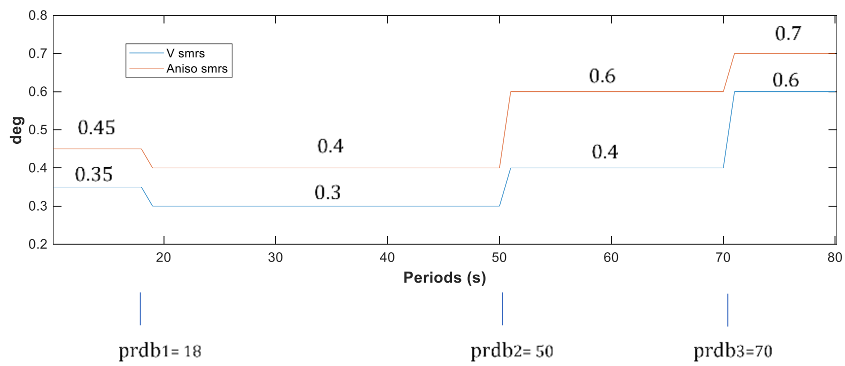

Figure 6. smoothing radius of different periods.

2.4 Run the script

Add the folder of 'src' and its subfolder to MATLAB path

(1)	**ab_step1_AutoPickWaveform.m**	(MATLAB: Automatic waveform selection).
(2)	**ac_step2_WGM_grid.m**	(MATLAB: Wave Field Gradient Calculation).
(3)	**ad_step3_event2sta. m**	(matlab: extracts different azimuth phase velocities).
(4)	**ae_step4_AnisoFit. m**	(matlab: azimuth anisotropy fitting of surface waves).

Details:

(1) **ab_step1_AutoPickWaveform. m**

Automatic waveform picking

>> Enter:

Seismic event waveform data in SAC format (refer to Section 1.2).

>> running parameters:

is_parfor:	=1, parallel; =0, single-core calculation
par.sacstr:	The universal naming format for sac data
par.is_overwrite:	=1, overwrite; =0, not overwrite
dataF:	the folder of dataset

> > output:

(1) Matlab files of earthquake waveform: /wvPick/**$YYYMMDD. $JDAY. $hhmmss. $ms.mat.** Each file contains a data structure with the variable **rsta**,

rsta =

stn:	(station name)	st:	(latitude and longitude, and elevation in station)
dis:	(epicenter distance)	azm:	(azimuth).
bazm:	(backazimuth)	ev:	(epicenter latitude, epicenter longitude, and focal depth)
dt :	(Sample Rate)	dat:	(waveform)
npt :	(number of sampling points)	cmp:	(component)
net:	(name of network)		
tphs :	(user-defined reference time for cutting event waveforms, e.g. P-wave arrival).		
tbeg :	(start time of wavefrom)		
t0	(YYY,JDAY,hh,mm,tbeg)		
t1:	(YYY,JDAY,hh,mm,tbeg+npt*dt)		

(2) Automatic waveform picking result: /wvPick/wvpick/**wvpick. $YYYMMDD. $JDAY. $hhmmss. $ms.mat.** Each Matlab file contains a data structure with variable of **wvpick**

wvpick =

Evflag:	(picking flag of earthquakes)
datFlag:	(quality flag for different periods)
period:	(center period)
vg:	(automatically recognized polygon velocity).
dist:	(epicenter distance)
goodstID:	(waveform quality flag of different stations).
kmax0:	(the sample point of the maximum amplitude of the surface wave envelope)
SNR :	(signal-to-noise ratio).

（2） ac_step2_WGM_grid. m

WGM analysis

> > input:

./wvPick/**$YYYMMDD. $JDAY. $hhmmss. $ms.mat**

./wvPick/wvpick/**wvpick. $YYYMMDD. $JDAY. $hhmmss. $ms.mat**

>> Operating parameters:

par.sacstr:	The common naming format for input data
is_smooth:	keep is_smooth = =1
dataF:	Matlab file of Event waveform

> > output:

Matlab files of **wga.* $YYYMMDD. $JDAY. $hhmmss. $ms.mat,** which are saved in the folder of **/WG/ $YYYMMDD. $JDAY. $hhmmss. $ms/**. The output files with the string of "smr" save the smoothing phase velocity,which are used to fitting the anisotropy of Rayleigh wave.

wga =

stn:	(ID of reference locations)
st:	(latitude and longitude, and elevation in station)
azmo:	(azimuths estimate by WGM)
nst:	(number of auxiliary stations ultimately used in each reference location)
envu0:	(envelope peak of surface wave)
Time:	(Time corresponding to surface wave envelope peak)
amp:	(waveform amplitude corresponding to the peak of the surface wave envelope)
evla:	(epicenter latitude)
evlo:	(epicenter longitude)
evdpth:	(focal depth)
v:	(phase velocity smoothed with **smrv**)
va:	(phase velocity smoothed with **smra**)
dv:	(standard deviation of phase velocity)
az:	(azimuth difference).
da:	(standard deviation of azimuth difference)
rd:	(radiation pattern).
dr:	(standard deviation of radiation pattern)
gs:	(geometric spreading).
dg:	(standard deviation of geometric spreading)
Ax:	Wave Gradient Coefficient Ax).
Ay:	(Wave Gradient Coefficient Ay)
Bx:	(Wave Gradient Coefficient Bx).
By:	(Wave Gradient Coefficient By)
dBdx:	(spatial derivative of the coefficient Bx).
dBdy:	(spatial derivative of the coefficient By)
dAdx:	(spatial derivative of the coefficient Ax).
dAdy:	(spatial derivative of the coefficient Ay)
vcr:	(phase velocity corrected by amplitude term and smoothed by **smrv**).
vcra:	(phase velocity corrected by amplitude term and smoothed by **smra**).
dvcr:	(v-vcr).
dsmax:	(ultimate radius of each subarray)

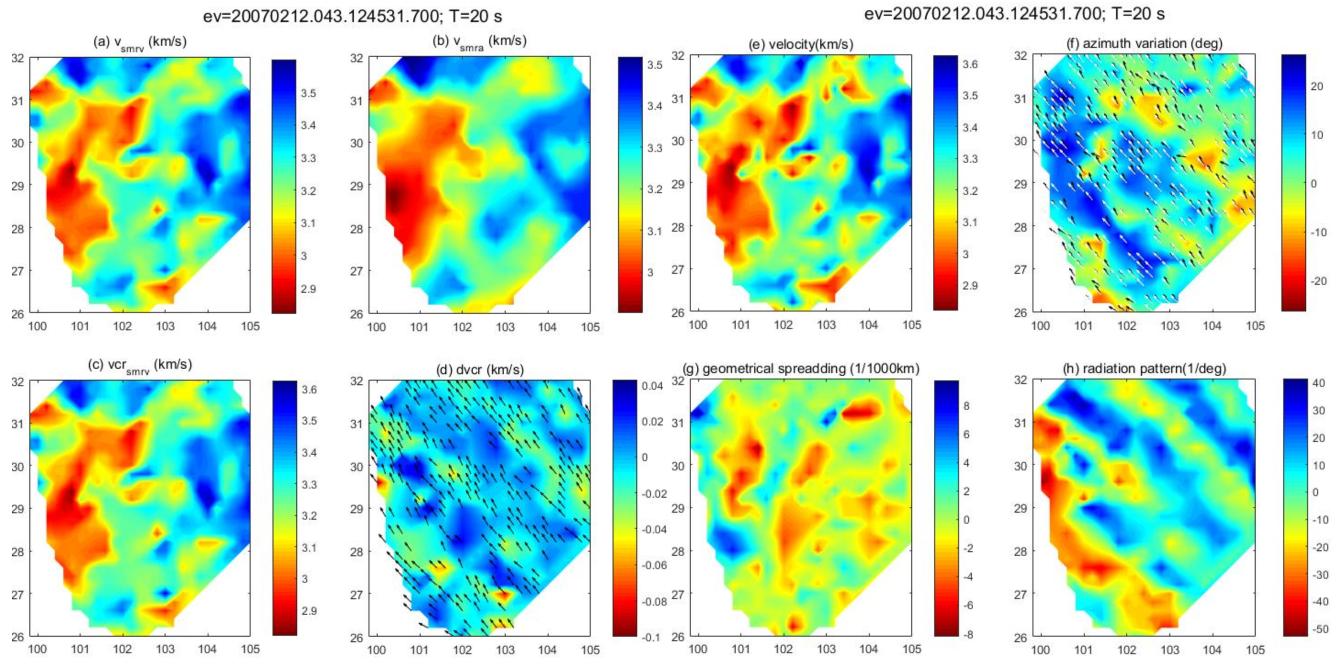

Figure 7 Rayleigh wave imaging results of earthquake 20070212.043.124537.700 at T=2 0 s. (a) phase velocity smoothed with **smrv**; (b) phase velocity smoothed with **smra**; (c) vcr; (d) dvcr; (e) phase velocity without smoothing and amplitude correction; (f) Azimuth variation, the white arrow is the great circle azimuth, the black arrow indicates the azimuth estimated by WGM; (g) geometrical spreading; (h) (radiation pattern).

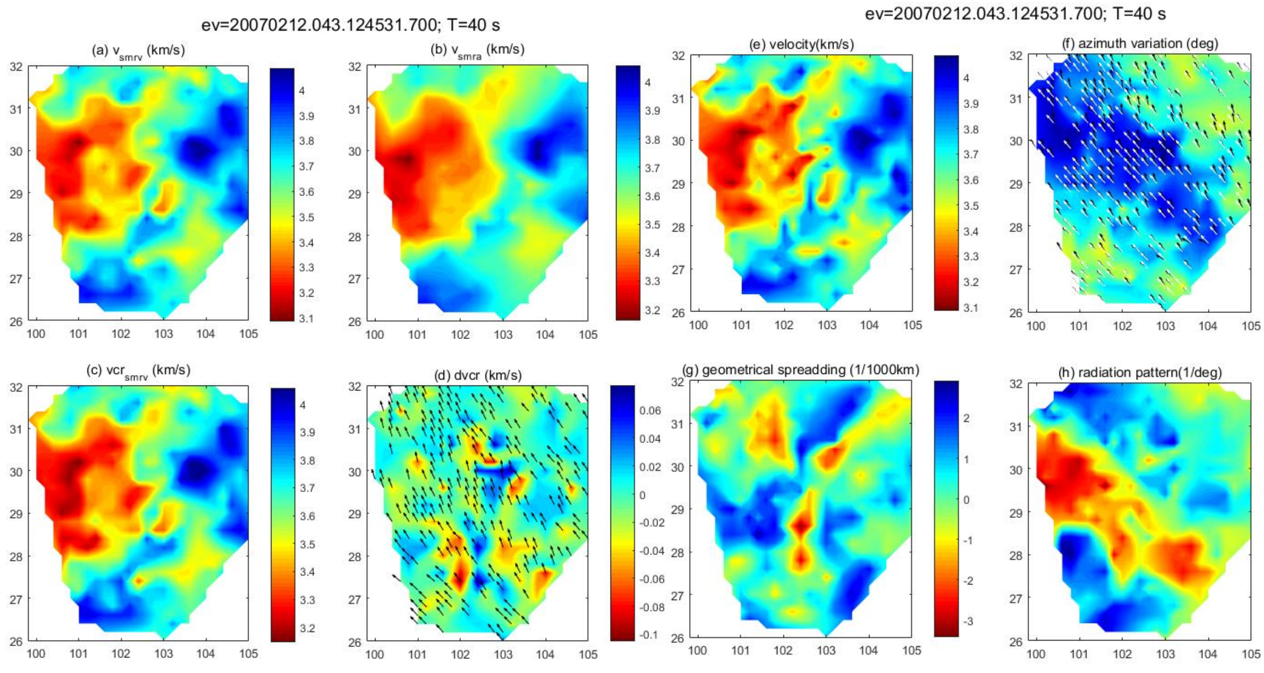

Figure 8. Same as Figure 7, but with period of T = 40s.

（3） **ad_step3_event2sta.m**

Convert the results into a station-based mat file

> > input:

./WG/ $YYYMMDD. $JDAY. $hhmmss. $ms/ **wga.* $YYYMMDD. $JDAY. $hhmmss. $ms.mat**

>> running parameters:

EVS: Input path
fstr: common naming format of the input file
outpath: cache path for parallel computation
outpath2: output path for saving the final results

> > output:

Matlab files of **bk_$longitude_$latitude.mat,** which are saved in "**outpath2". These** files save the WGM results of different grid points:

wgst =

periods:

st: (latitude and longitude, and elevation in station)

evla: (epicenter latitude)

evlo: (epicenter longitude)

v: (phase velocity smoothed with **smrv)**

va: (phase velocity smoothed with **smra**)

vcr: (phase velocity corrected by amplitude term and smoothed by **smrv**).

vcra: (phase velocity corrected by amplitude term and smoothed by **smra**).

dvcr: (v-vcr).

azm: (azimuth).

（4） ae_step4_AnisoFit.m

Fit azimuth anisotropy

> > input:

```
./aniso/station/wga.* $YYYMMDD. $JDAY. $hhmmss. $ms.mat,
```

>> parameter settings

inroot	input path
ststr	input data universal naming format
outpath0	cache path for parallel computation
outpath1	output path for saving the final results
azmbin	azimuth interval
MaxAzm	=180, azm(azm>180) = azm(azm>180)-180; =360, azm=azm
is_onephase	=0, fitting model: v=viso+A*cos(2*(theta-fai2)); =1, fitting model: v=viso+a*cos(2*(theta-fai2)))+b*cos(theta-fai1).
vlim:	km/s, phase velocity C satisfying abs(v-median(v))> **vlim** are deemed as outlier
is_CIR	=1, phase velocity C satisfying v-viso-a*cos(2*theta)-b*sin(2*theta)) > 0.5***vlim** are deemed as outlier
velocityType	= 'dym', using the v to fitting anisotropy; ='strc', using the vcr to fitting anisotropy
is_bootstrap	is_bootstrap (1)=1, calculate the standard deviation, costs more time is_bootstrap (2) is number of resampling times; is_bootstrap (3) Resampling ratio for each time.
prdb	periods boundary for making smoothing parameter
smrv	smoothing radii for velocity in different period intervals
smra	smoothing radii for velocity in different period intervals
outmatf	output file without smoothed
outmatf2	**o**utput file with smoothed

> > output:

"outmatf2", saved in "outpath1". This Matlab file save the results of isotropic phase velocity and azimuth anisotropy, and contain the data structure with variable **Aniso**

Aniso =

st:	(locations).	viso:	(isotropic phase velocity).
stdv:	(standard deviation of phase velocity).	Fai:	(fast propagation).
M:	(anisotropic magnitude).	a:	(anisotropic model parameter).
b:	(anisotropic model parameter).		
stdFai:	(standard deviation in the direction of fast waves).		
stdM:	(standard deviation of anisotropic magnitude).		
stdA:	(standard deviation of **a**).	stdB:	(standard deviation of **b**)

>> GUI or app Interface:

ae_step4_AnisoFitGUI.m or AnisoFitApp.mlapp is the interface script corresponding to this step. The interface is as follows:

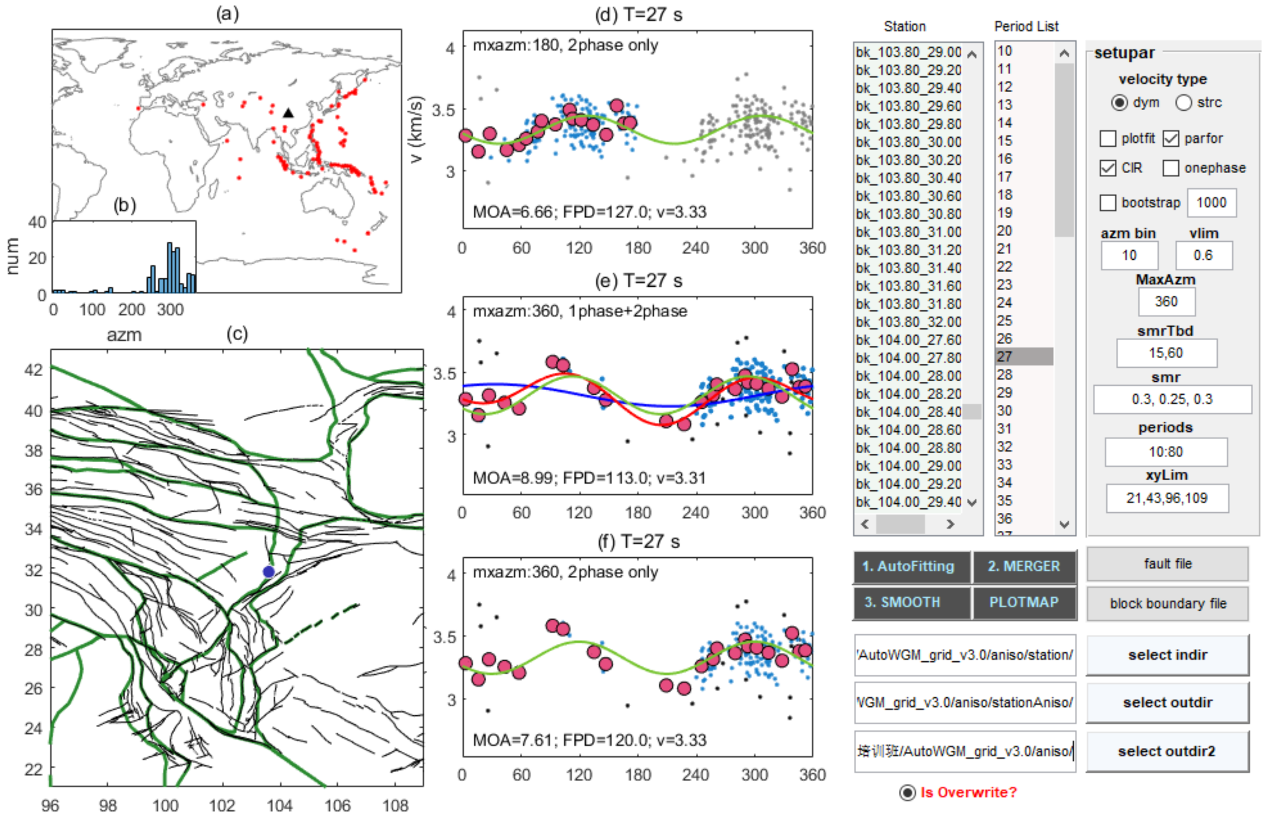

Figure 9. The GUI interface for anisotropic fitting. The left two columns are the plot panel, and the right two columns are the data list, parameters setting, and button panels. (a) earthquakes distribution map; (b) Statistical chart of the azimuth of the earthquake; (c) display window for reference locations; (d) Anisotropic fitting results, azm(azm>180) = azm(azm>180)-180, fitting only 2-phase anisotropic model of Rayleigh waves (Cao et al. 2020; Cao et al. 2023). The gray dots are the input phase velocity, the blue dots are the phase velocity after removing the outlier, the red point is the median phase velocity at each azimuth bin, and the green curve is the final fitting result (text above the X-axis). (e) Same as figure (d), but **MaxAzm** = 360°, and fitting both the 1-phase and 2-phase anisotropic models. The green curve is the fitting result of 2-phase model, and the blue curve is the fitting result of 1-phase model. The red curve is the sum of the two. (f) Same as figure (e), but only fits the 2-phase anisotropy model.

"**Station**": list of grid points;

"**periodList**": list of cycles;

"**setupar**": parameter setting panel;

"**1.AutoFitting**": automatic calculation;

“**2.MERGER**": Combine the results of parallel computations;

" **3. SMOOTH**": smoothing results;

" **PLOTMAP**": Displays isotropic and anisotropic results;

' **fault file**': Select the fault file;

"block boundary file": Select a block boundary file;

" **select indir**": select the input path;

" select outdir": Select the cache path for parallel computation;

" Select outdir2": Select the output path

3. Extracting the 3D velocity and anisotropic structure based on the ADDCI method

3.1 Introduction of ADDCI

This part follows the Azimuth dependent dispersion curve inversion (ADDCI) method developed by Liang et al. (2020). A major feature of this method is that anisotropic perturbation is added to the dispersion curves in different azimuths, and velocity models in different azimuth are obtained by inverting these azimuth-dependent dispersion curves. Figure 10 shows a flowchart of the ADDCI method, which is based on the linear inversion package of the CPS330 to invert the dispersion curve of a single azimuth (Users can also take advantage of other nonlinear inversion methods or add receiver functions for joint inversion (e.g., Cao et al. 2023)). One azimuth corresponds to a three-dimensional shear wave velocity model that incorporates anisotropic perturbations. Three-dimensional anisotropic structures can be extracted directly from these models.

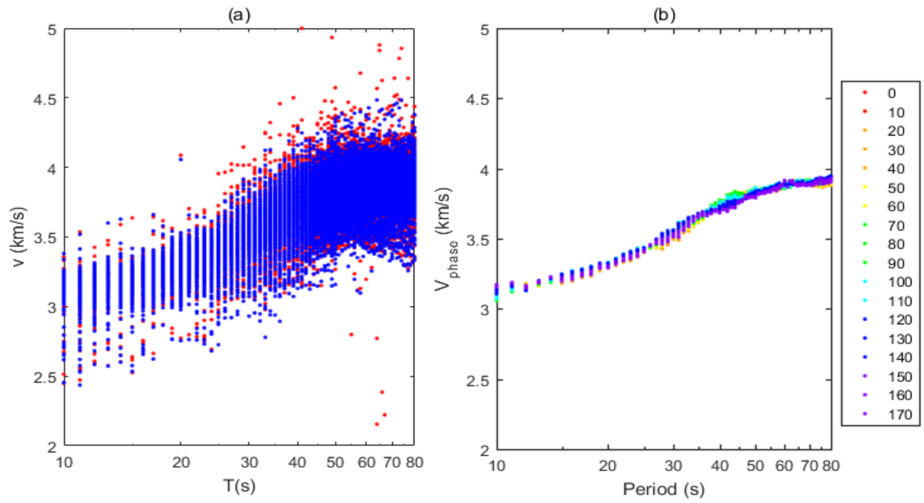

图 10. Azimuth-dependent dispersion of grid point. (a) dispersion of earthquakes in different azimuth. where the red and blue dots are amplitude-corrected phase velocities (vcra) and unamplitude-corrected phase velocity (va). (b) Azimuth-dependent dispersion curves in different azimuth bin extracted by Matlab script of **af_step5_WG2surface96.m.**

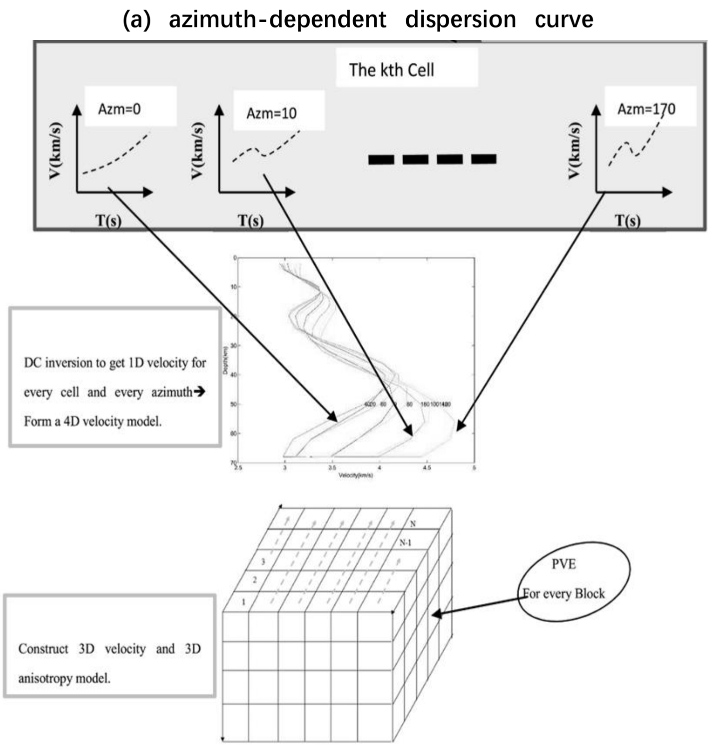

Figure 11. flowchart of the ADDCI method (Liang et al. 2020).

3.2 Run the script

(1) af_step5_WG2surface96.m

(MATLAB: Extract dispersion curves of different azimuth bins, Figure 1 1a).

(2) mpi_job_submit.sh or **job_submit.sh**

(shell: dispersion curve inversion, Figure 1 1a-11b)

(3) Fit_3Dmod.m

(MATLAB: Extract 3D velocity and anisotropic structures, Figure 11b-11c)

detail

(1) af_step5_WG2surface96.m

Extract dispersion curves of different azimuth bins (adding anisotropic perturbations to dispersion curves).

> > input:

ae_step4_AnisoFit.m output of isotropic phase velocity and anisotropy (**Aniso*azmbin*_allst*.mat**)

>> Operating parameters:

indir: (input path).
infile: (input file).
outpath: (output path or dispersion curve inversion path).
prdLim: (period range you want to output).
azms: (azimuth bin).

> > output:

(1) Dispersion curve file: **bk$*bkID*.deg.$azm.dsp**, saved in the folder of **$outpath/$bkID/**

SURF96 R C X 0 Period Phase Velocity 0.000 6.000

...

(2) location file: **xyloc.txt**, save in the folder of $**outpath/$bkID/.**

$bkID latitude longitude

1 96.400 39.400

(3) 3D inversion script

Dispersion curve inversion script:

job_submit.sh :
mpi_job_submit.sh:
DispInv.sh:
jointinversion.sh:
mod.d2:
run_shell_mpi.f90:
stacksac2.0AJWA:

Extract the 3D velocity model and azimuth anisotropic structure:

Extract_3Daniso folder containing scripts for extracting 3D velocity models and azimuth anisotropic structures

Fit_3Dmod.m:
plot_aniso.m:
read_dsp.m:
smoothmod.m:

(2) Dispersion curve inversion

cd the inversion path (Default is inv)

> > input:

Initial model:
Fixed Model (default): Modify the initial model file of **mod.d2**
Spatially dependent initial model: user can modify the ***"intMod"*** parameter in the script of **DispInv.sh** to call different initial model
Dispersion curve: The output of the previous step

>> parameter settings

DispInv.sh :

intMod:	(initial model file).
rfpath:	(accepts function paths).
rfile:	(receiver function file name, when the parameter **"joint96 43"** in the **jointinversion.sh** is less than 1, the user needs to provide additional receiver function dataset for joint inversion).
invScriptF:	(inverting subroutine, corresponding to the inversion of a single dispersion curve).
jobi:	(input parameter file).

jointinversion.sh : (for more detailed parameter settings, please refer to the parameter settings of the "joint96" command in CPS330).

joint96 32 0.5	(damping factor setting)
joint96 36 1	(smoothing setting)
joint96 43 1	(1, surface wave dispersion; 0, receiver function; 0<w<1, joint inversion)
joint96 1 2 6 6 1 2 6 1 2 1 2	(Set the number of iterations)

> > submit job

Running with single core: **./job_submit.sh**

Parallel running: **./mpi_job_submit.sh**

mpirun -np $num run_shell_mpi #$num: number of core

> > output:

velocity model in different azimuth bin: **bk.$bkID.deg.$azm.dsp.modo3**

(3) Fit_3Dmod.m (in the Extract_3Daniso folder).

Extract 3D velocity model in different azimuth bin.

> > input:

velocity model in different azimuth bin: **bk.$bkID.deg.$azm.dsp.modo3**

>> parameter settings

pthi0	(input and output path).
bk	(grid point ID).
par.outyp	(= 'aniso', output anisotropy and Vs model; = 'iso', input data not contain anisotropy).
smr	(Model smoothing radius, = [] (default)).
par. MaxAzm	(maximum azimuth).
par.isparfor	(=1, parallel operation, =0 single-core operation).
par.isfigure	(default=0).
par.isoverwrite	(=1, overwrite the original result, =0, skip the original result).
par.onephase	(=0 (default); =1, used to test the effect of the dip interface on the anisotropy from different azimuth receiver function during the joint inversion)
par.depthrange	(the depth range of the model).
dp	(different depth ranges for outputting average results).

> > output:

(1) 3D vs model and azimuth anisotropic structure **mod3d_1phase*_maxazm*.mat**. Vs and azimuth anisotropy of different location and depths are saved in the data structure of **mod3d**:

mod3d =

1x720 struct array with fields:

ibk	(grid pointID).
xbk	(longitude).
ybk	(latitude).
dp	(depth).
AZMS	(azimuth).
viso	(speed).
FPD	(Fast Wave Direction).
MOA	(Anisotropic Strength).
a	(anisotropy model coefficient a).
b	(anisotropic model coefficient b).
vs	(velocity model with different azimuth).

(2) Horizontal slice: vsdep_*_km_*.dat

Latitude,	Longitude,	Velocity,	MOA,	FPD
29.200	99.800	3.150	7.235	20.0

(3) Average results of different depth ranges: vslay_dep*_*km_*.dat

Latitude,	Longitude,	Velocity,	MOA,	FPD,	Delay time,	Grid ID
29.2	99.8	3.250	2.75	51.0	0.470	1

References

Langston CA (2007c). Wave gradiometry in the time domain. Bull Seismol Soc Am 97(3): 926 – 933 https://doi.org/10.1785/0120060152

Liang CT and Langston CA (2009). Wave gradiometry for USArray: rayleigh waves. J Geophys Res: Solid Earth 114(B2): B02308 https://doi.org/10.1029/2008JB005918

Liang, C., Liu, Z., Hua, Q., Wang, L., Jiang, N., & Wu, J. (2020). The 3D seismic azimuthal anisotropies and velocities in the eastern Tibetan Plateau extracted by an azimuth-dependent dispersion curve inversion method. Tectonics, 39, e2019TC005747. https://doi.org/10.1029/2019TC005747

Cao FH, Liang CT, Zhou L and Zhu JS (2020). Seismic azimuthal anisotropy for the southeastern Tibetan Plateau extracted by wave gradiometry analysis. J Geophys Res:Solid Earth 125(5): e2019JB018395 https://doi.org/10.1029/2019JB018395

Cao FH, Liang CT, Yang YH, Zhou L, Liu ZQ and Liu Z (2023). 3D velocity and anisotropy of the southeastern Tibetan plateau extracted by joint inversion of wave gradiometry, ambient noise, and receiver function. Tectonophysics 848: 229690 https://doi.org/10.1016/j.tecto.2022.229690