PolarGUI

A MATLAB-based tool for polarization analysis of the three-component seismic data using different algorithms

User Manual

Key Laboratory of Earth Exploration and Information Technology (Chengdu University of Technology), Ministry of Education

Huailiang Li
Email: lihl@cdut.edu.cn

2021.4

Content

1. Introduction

Polarization attributes of a three-component (3C) seismic recording are crucial parameters regarding seismic wavefield analysis, because of the various propagation ways and motion trajectories of the different elastic wave types, such as P waves, S waves, Rayleigh waves, and Love waves, which all have distinct polarization patterns. Polarization parameters can be applied to polarization filter, P/S wave separation, seismic wave propagation direction estimation, geophone orientation calculation, etc. During the last few decades, various reliable approaches have been proposed to obtain the polarization properties, which includes particle motion trajectory (hodogram), eigenvalue decomposition (EVD) based on the covariance matrix (CM), which consists of two calculation procedures, named EVD-CM1 and EVD-CM2 respectively, singular value decomposition (SVD) which uses the principal component analysis (PCA), noted PCA-SVD, and eigenvalue decomposition (EVD) using the analytic signal matrix (ASM), named EVD-ASM. These methods can provide good calculation results and have a significant impact since they have been developed. Among the aforementioned approaches, the most popular method to identify the propagation direction of the single-component seismogram is the hodogram analysis. Hodograms are a visual approximation of the particle trajectories that are plotted in a 2D plane, which use the curves determined by the displacement of two components over a specific time-window. For 3C seismic recordings, more sophisticated approaches are based on the eigenvalue and eigenvector calculations of the covariance matrix, which can reliably determine the polarization azimuth and dip angle of a seismic recording.

The aim of this study is to provide a PC tool for the polarization analysis of the 3C seismic recordings using the aforementioned polarization calculation methods. This program was written in MATLAB language (created programmatically, not using GUIDE) and in open-source mode. It can be expediently modified, and other algorithms can be added. In the next section, we will briefly introduce the specific functions of the polarization analysis graphical user interface (PolarGUI), and then we describe how to perform a polarization analysis. Finally, a comparison is given to illustrate the differences between the four considered polarization approaches.

2. Overview of the PolarGUI

2.1 PolarGUI

As shown in **Figure 1**, the PolarGUI consists of 9 function modules, which include the open-file, exit-program, filter and rotate 3D on/off buttons, radio buttons for the selection of different polarization approaches, a waveform display for original seismic recording and the intercepted seismic recording, different exhibitions of the polarization calculation results, two pairs of sliding cursor buttons and dominant frequency exhibition of the two intercepted seismic recordings. The specific polarization analysis illustrations consist of two groups of rose diagrams and two groups of histograms for the azimuth distribution and the incident angle distribution. The two groups correspond to two selected seismic datasets. Moreover, two groups of statistical curve subgraphs are given to demonstrate the degree of rectilinearity and the degree of planarity for the two sections of data. In addition, this GUI also contains two 3D display windows, which are used to draw the hodograms for two the selected sections of data. The PolarGUI tool was developed on the macOS Mojave (version 10.14.6) platform using the 64-bit professional version of MATLAB (R2017b). It can also normally run on the Window 10 (64 bits) platform with the 64-bit professional version of MATLAB (R2017b). This tool may be compatible with the latest version, but it has not been tested to determine if it is backward compatible with earlier versions of MATLAB.

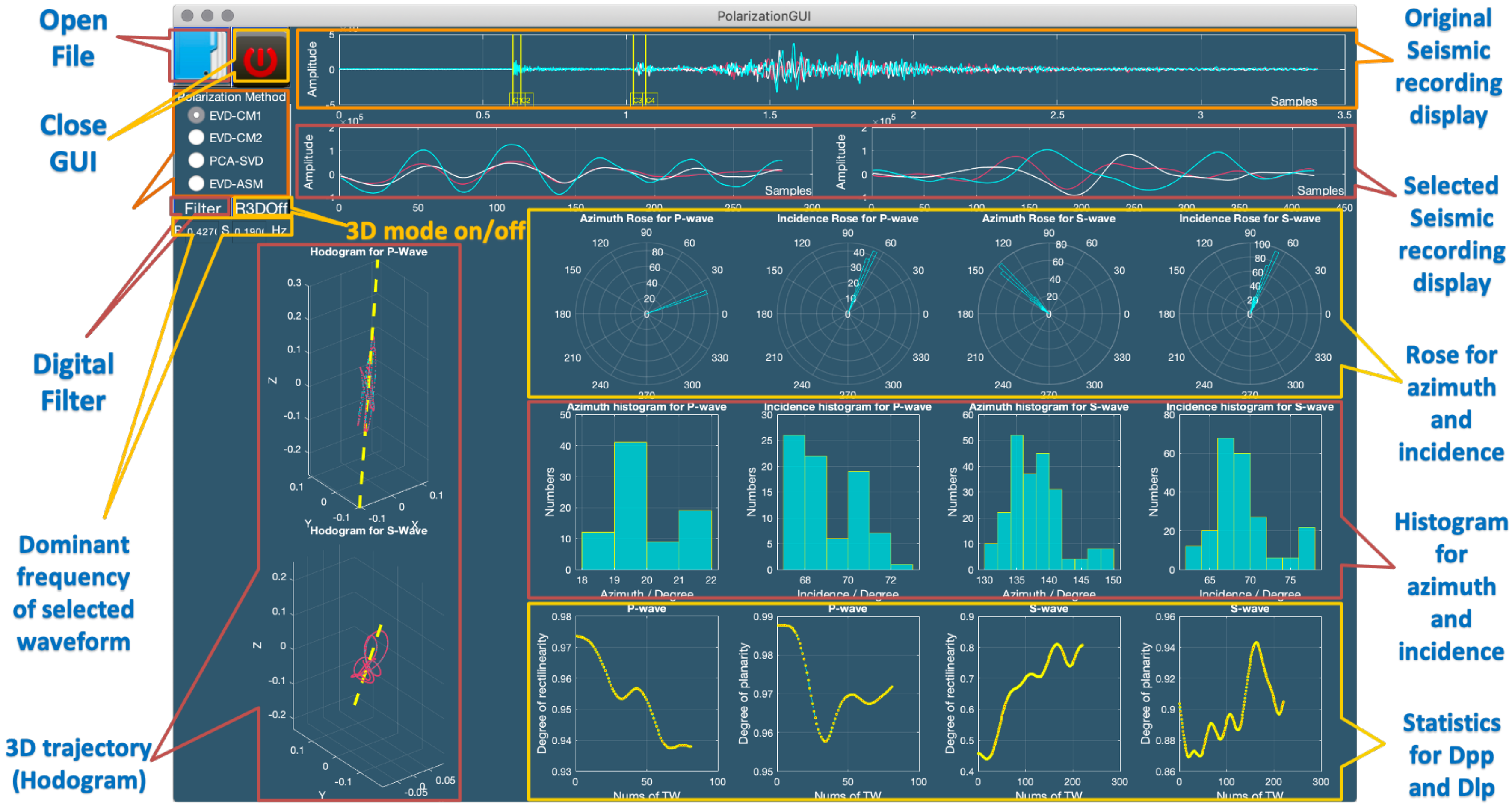

Figure 1. The function modules of the PolarGUI.

The PolarGUI can be applied to open various data formats, including the SEG-2 and SEG-Y formats (corresponding to the engineering seismic data), SAC and MiniSEED formats (corresponding to the natural earthquake data), and the general xls, xlsx, or txt format for pure seismic data. Moreover, for different file storage modes, PolarGUI supports simultaneous opening of a single file or multi files, which is convenient when handling different data resources. To accurately display of the analyzed data section, two pairs of cursors are available in the original seismic wave display window, and all of the cursors can be arbitrarily slid along with the horizontal axis. The intercepted waveforms will be exhibited in two independent windows. The general workflow of PolarGUI is summarized in **Figure 2**.

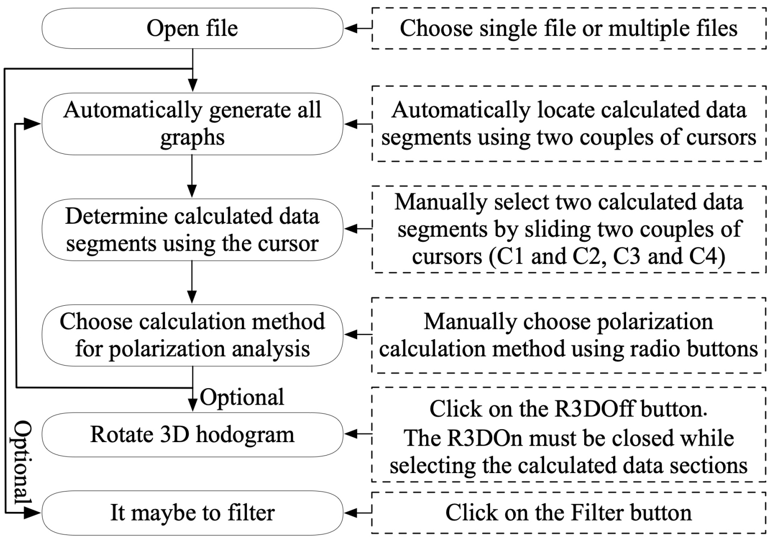

Figure 2. General workflow of polarization analysis using the PolarGUI program.

2.2 Digital filter GUI (DFGUI)

In addition, we designed an auxiliary digital filter GUI (DFGUI) to perform the data preprocessing operation (Figure 3). It includes a low-pass filter and a band-pass filter, and the corresponding cut-off frequency can be modified according to the actual requirements. In order to avoid the phase distortion of the original seismic dataset, the DFGUI was developed using a finite impulse response (FIR) filter, which possesses rigorous linear phase features. We used different window functions to achieve the FIR filter design. This window function consists of a Hamming window, a Gauss window, a Kasier window, a Hanning window and a Blackman window. This procedure involves the 'filtfilt' function, which is provided by the MATLAB software. It can perform a zero-phase digital filtering, which ensure that it does not affect the phase of the original seismic signal. The operation instructions for the DFGUI will be given using a concise user manual in the software package in the supplemental material..

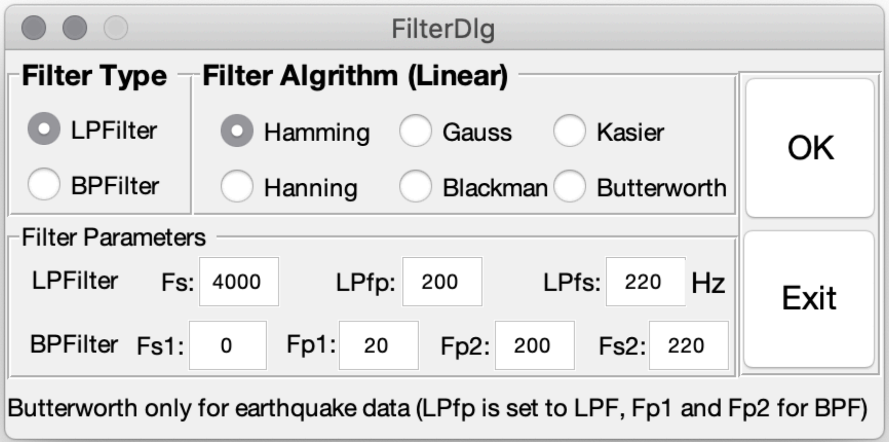

Figure 3. The digital filter GUI.

3. Software code file directory structure

(1) Polarization_GUI.m -----> Main program;

(2) polarize_estimation.m -----> The polarization calculation method for eigenvalue decomposition based on the covariance matrix (EVD-CM1);

(3) polarize_estimation2.m -----> The polarization calculation method for eigenvalue decomposition based on the covariance matrix (EVD-CM2);

(4) polarization_PCA.m -----> The polarization calculation method for singular value decomposition (PCA-SVD);

(5) polar_analysticSig.m -----> The polarization calculation method for eigenvalue decomposition based on the constructed analytic signal matrix (EVD-ASM);

(6) FilterDlg.fig and FilterDlg.m -----> The digital filter GUI and filter code;

(7) buttern_low.m and buttern_filter.m --> low-pass or band-pass butterworth filter

(8) seg2read.m -----> Read seismic recording for Seg2 format;

(9) rdsac.m -----> Read seismic recording for SAC format;

(10) rdmseed.m -----> Read seismic recording for Miniseed format;

(11) RdWrSgy folder -----> Read seismic recording for SegY format;

(12) Cursors folder -----> Cursors operation code for selecting calculation window;

(13) pca_ica folder -----> Principal component analysis (PCA) code;

(14) plot_dir3N.m and plot_dir3.m -----> Plot 3d curve;

(15) order.m -----> Order eigenvalues, d, into descending order;

(16) Fig2Dlg.m and Fig2Dlg.fig -----> Produce figure, just ignoring;

(17) dmean.m -----> Remove the mean from a row vector;

(18) csigm.m -----> Truncate and reconstruct a signal matrix;

(19) seismicdata folder -----> seismicdata repository.

4. Startup

The PolarGUI may be run on any platform supported by Matlab. We recommend that Matlab's version should be much higher than R2017b, due to that the PolarGUI was developed on the macOS Mojave (version 10.14.6) platform using the 64-bit professional version of MATLAB (R2017b). It can also normally run on the Window 10 (64 bits) platform with the 64-bit professional version of MATLAB (R2017b). This tool may be compatible with the latest version, but it has not been tested to determine if it is backward compatible with earlier versions of MATLAB.

On a Mac, just to copy all the files of the software package to a new single folder, and add the path of the created folder, then, open the Polarization_GUI.m, click on the run button in Matlab platform, or use shortcut shift+command+R to run. The following **Figure 1** will be showed.

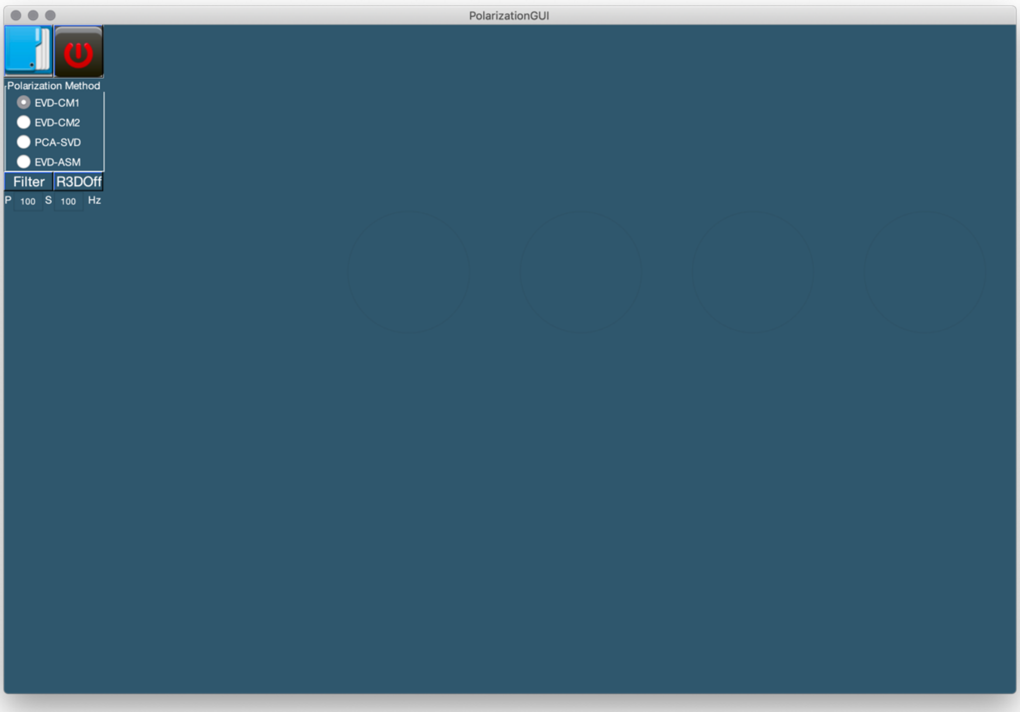

Figure 4. The startup views of the PolarGUI.

5. Instructions for use

5.1 Starting polarization analysis

As shown in **Figure 1**, single click on the Openfile icon on the GUI. From here you can go to 2 choices for seismic recording below: a single three-component (3C) seismic recording (**Figure 6**) or three single files for (3C) seismic recording (**Figure 7**) which is in the order E (X), N (Y) and Z (Z).

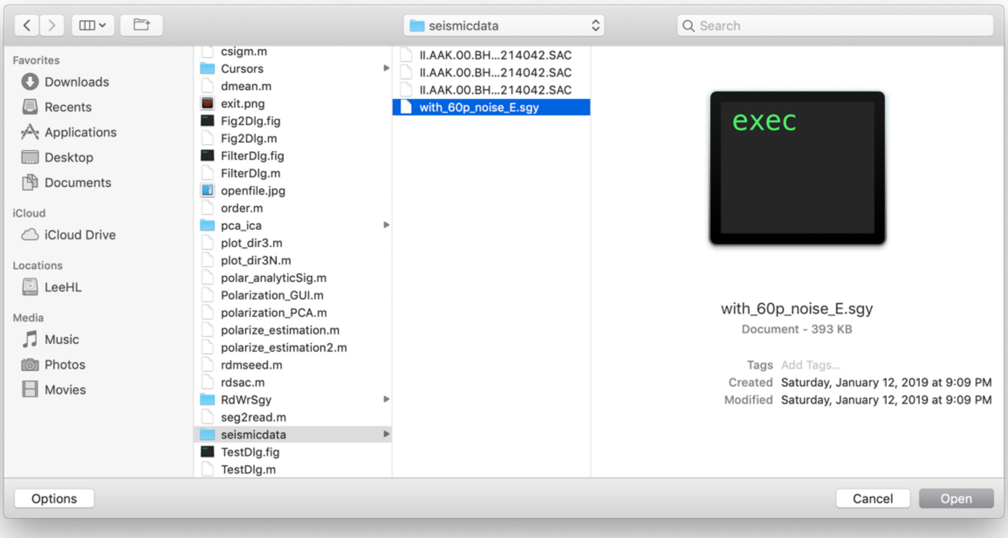

Figure 6. Open a single 3C file.

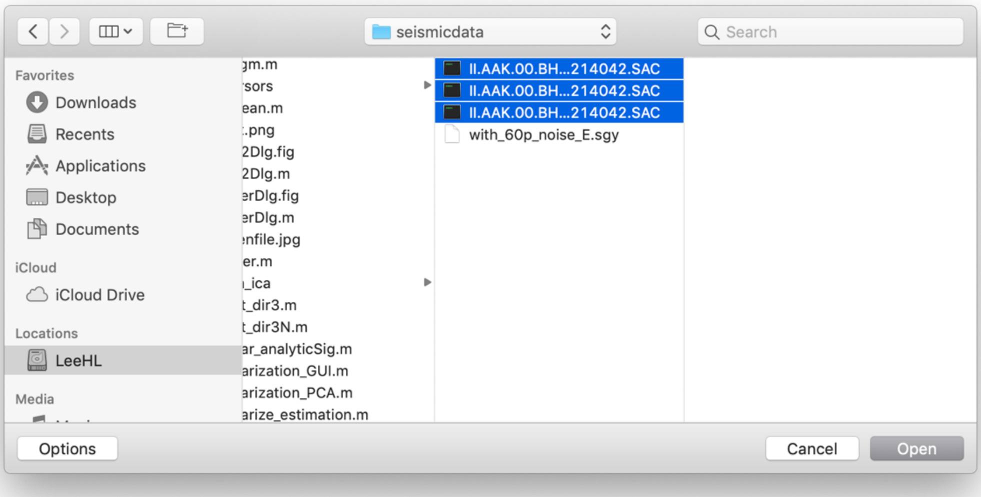

Figure 7. Open simultaneously multi files.

The PolarGUI can be applied to open various data formats, including the SEG-2 and SEG-Y formats (corresponding to the engineering seismic data), SAC and MiniSEED formats (corresponding to the natural earthquake data), and the general xls, xlsx, or txt format for pure seismic data. Moreover, for different file storage modes, PolarGUI supports simultaneous opening of a single file or multi files, which is convenient when handling different data resources .

Now, press 'Enter' key or click on 'open' button, a default calculation result will be shown in **Figure 8**, which uses a default EVD-CM1 polarization calculation method. In this sense, the location of two couples of cursors (yellow lines) are also default in the original seismic wave display window. The default cursor location is given as g_nslen/4 (C1), g_nslen/3 (C2), g_nslen/2 (C3), g_nslen (C4), which the g_nslen is the length of the original seismic recording. The C1 and C2 are a pair of selection region, C3 and C4 are another pair.

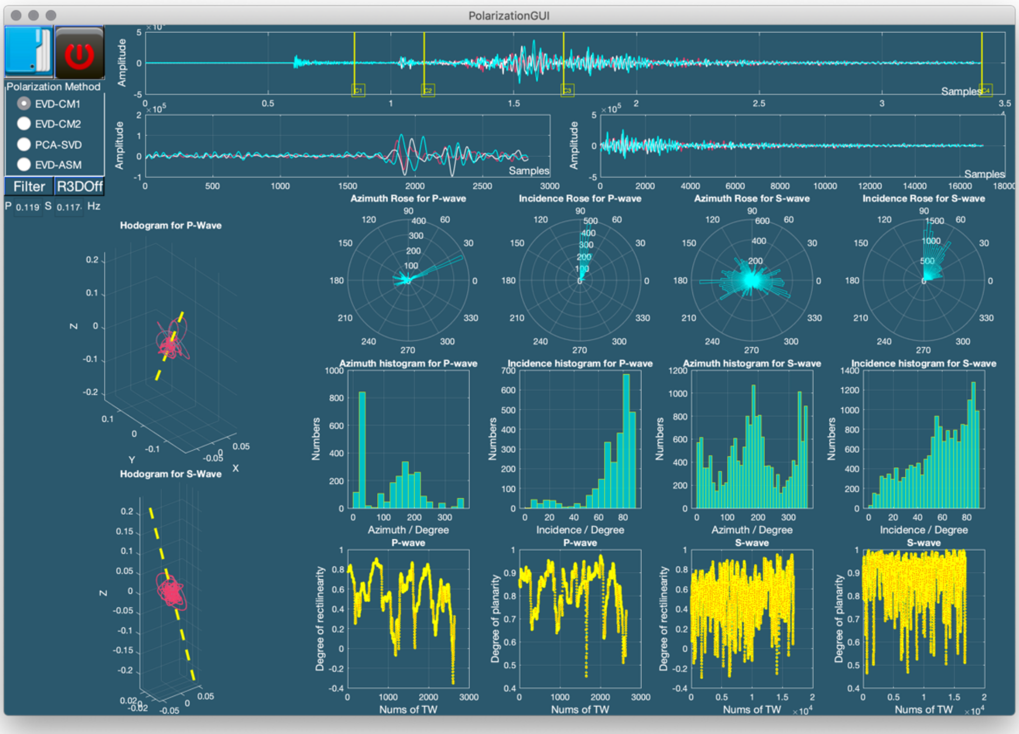

Figure 8. Default view of the calculated result.

You need to slide the cursors to determine the corresponding data section of the polarization calculation, the selected data sections are individually indicated in the two follow-up graphs. After determining the calculation segment, please click on another radio button to active a new calculation procedure for the designated data section. The calculation results are exhibited in **Figure 9.**

Certainly, to a proper data segment, different polarization calculation approaches can be arbitrarily adjusted.

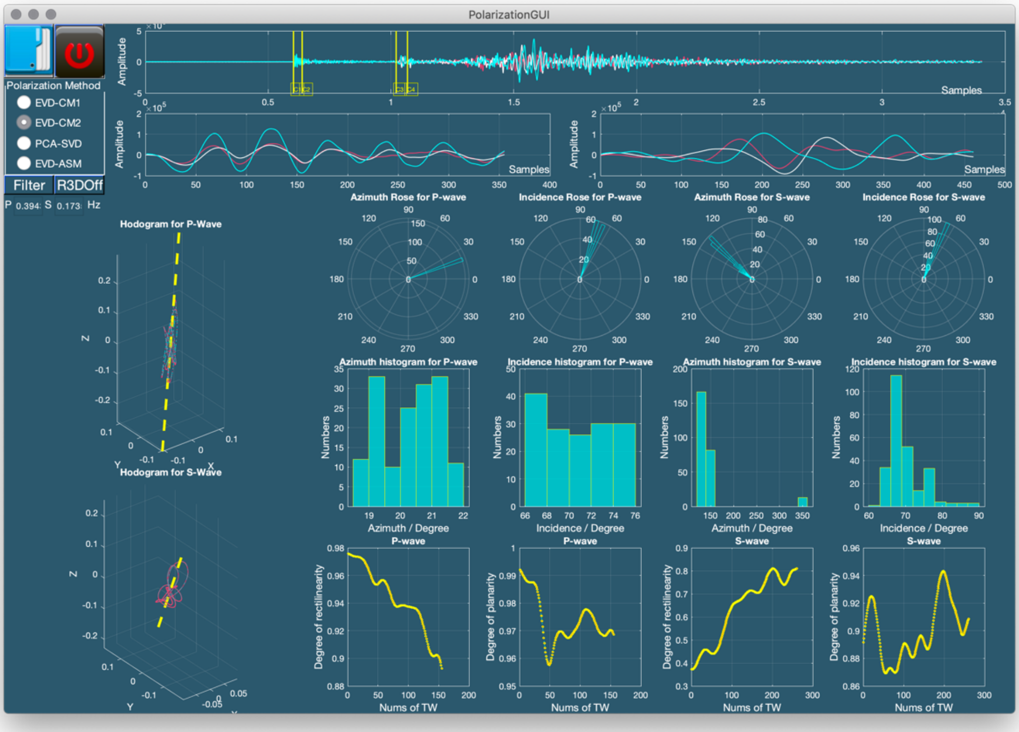

Figure 9. The new calculated result for the selected data section.

Besides, for two 3D hodograms, you can press 'R3DOff' button to open the 3D mode, and to perform a rotation operation by using the mouse, the 'R3DOff' button will be changed as 'R3DOn', Meanwhile, you can click on the right mouse button to obtain some rotation options, as shown in **Figure 10**. However, you cannot slide the cursor while the 'R3DOn' is opened, hence, you need to close the 'R3DOn' by clicking on the

‘R3DOn’ button again, which ensures that you can slide two pairs of cursors again. The polarization methods choice is unaffected when the ‘R3DOn’ is opened, and the 3D mode is closed by default.

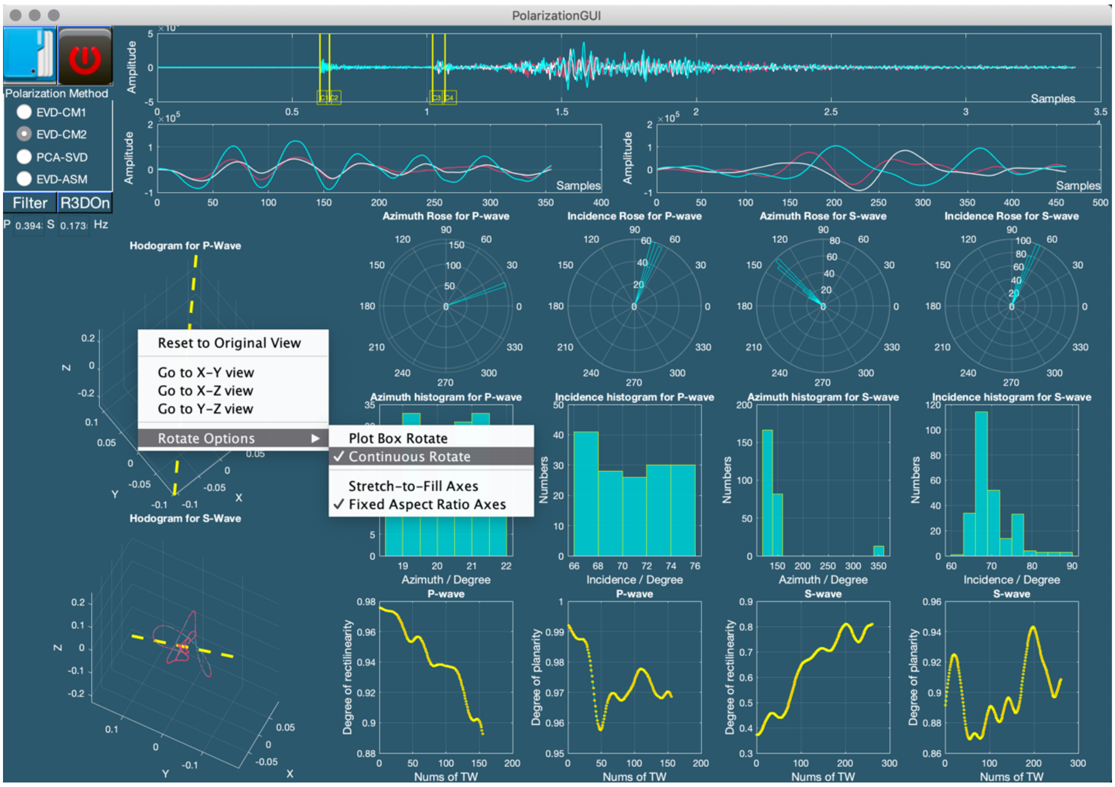

Figure 10. The view of that the ‘R3DOn’ is opened.

5.2 Use of digital filter

The designed digital filter GUI (DFGUI) is only an assistant tool in the PolarGUI. The Screenshot views of the DFGUI is as **Figure 11**. The DFGUI consists of a low-pass filters (LPFilter) and a band-pass filters (BPFilter), and the corresponding cut-off frequency can be modified according to the actual requirements. In the **Figure 11**, different window functions are used to achieve the FIR filter design, these window function consists of a Hamming window, a Gauss window, a Kasier window, a Hanning window and a Blackman window. Besides, we also employ a Butterworth filter which was designed by M. Reiss and G. Rümpker (2016), the Butterworth filter is only used to the natural earthquake recording, since it will produce a not very satisfactory result for microseismic recording.

To the LPFilter part, LPfp is the edge frequency of passband, LPfs represents the edge frequency of stopband, the corresponding cutoff frequency of the LPFilter can be defined as $F_c = (LPfp + LPfs)/2$. In the BPFilter part, Fs1 is the lower edge frequency of stopband, Fp1 is the lower edge frequency of passband, Fs2 is the upper edge frequency of stopband, Fp2 is the upper edge frequency of passband, the corresponding cutoff frequency of the BPFilter can be defined as $F_c = [Fs1 + Fp1)/2, Fs2 + Fp2)/2]$.

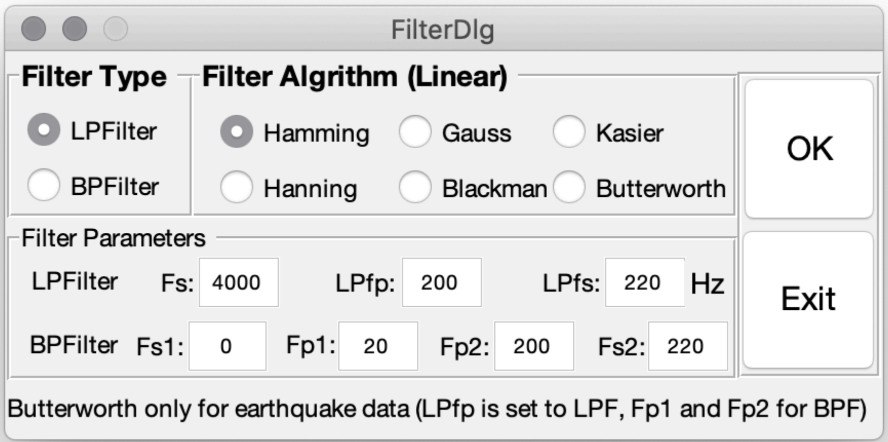

Figure 11. Screenshot views of the digital filter.

Moverover, the designed butterworth filter is only for seismic recording with very low frequency (VLF, e.g., natural earthquake recording), and for its LPFilter, the LPfp is set as the cutoff frequency, as for the BPFilter, Fp1 and Fp2 are set as the lower cutoff frequency and upper cutoff frequency, respectively.

To use the DFGUI, first, you need to choose a filter type (LPFilter or BPFilter). Then, choose a window function (the default is Hamming window). Finally, set the filter parameters according to the regulation of the aforementioned cutoff frequency calculation for different filter types.

For example, a noisy synthetic microseismic dataset is given in the **Figure 12**, the sample rate of the microseismic recording is 4000 Hz, resulting in that the Fs is filled automatically with 4000 while reading from the original microseismic data.

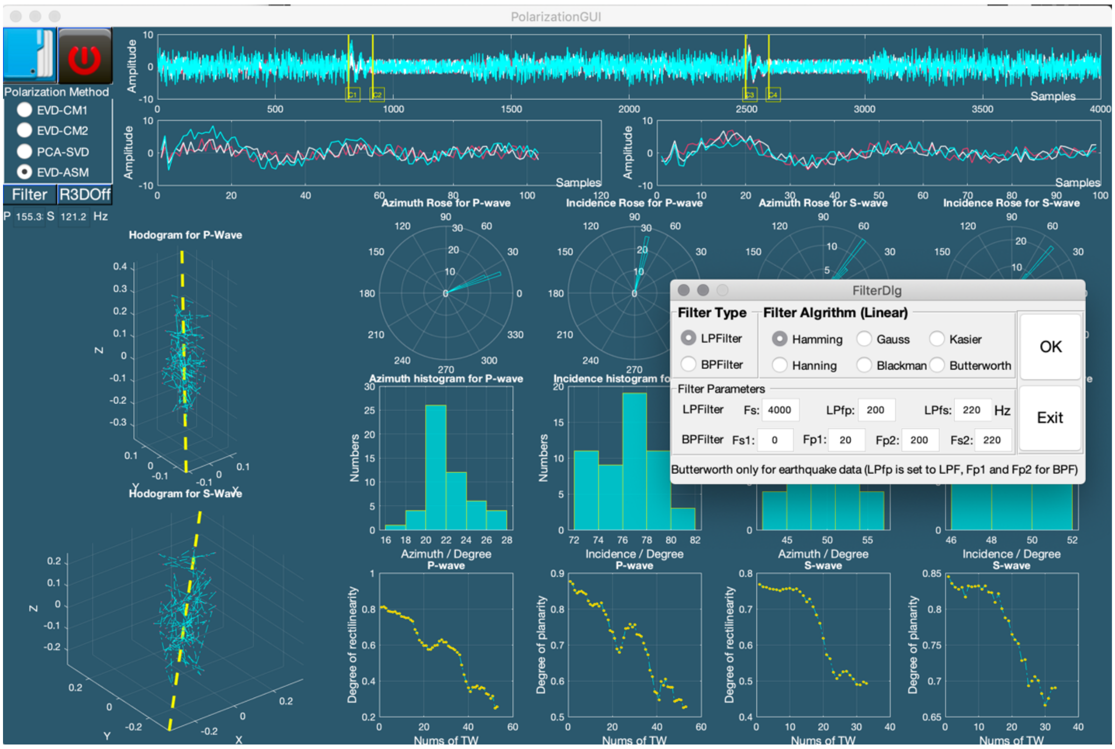

Figure 12. Polarization calculation results for noisy microseismic data.

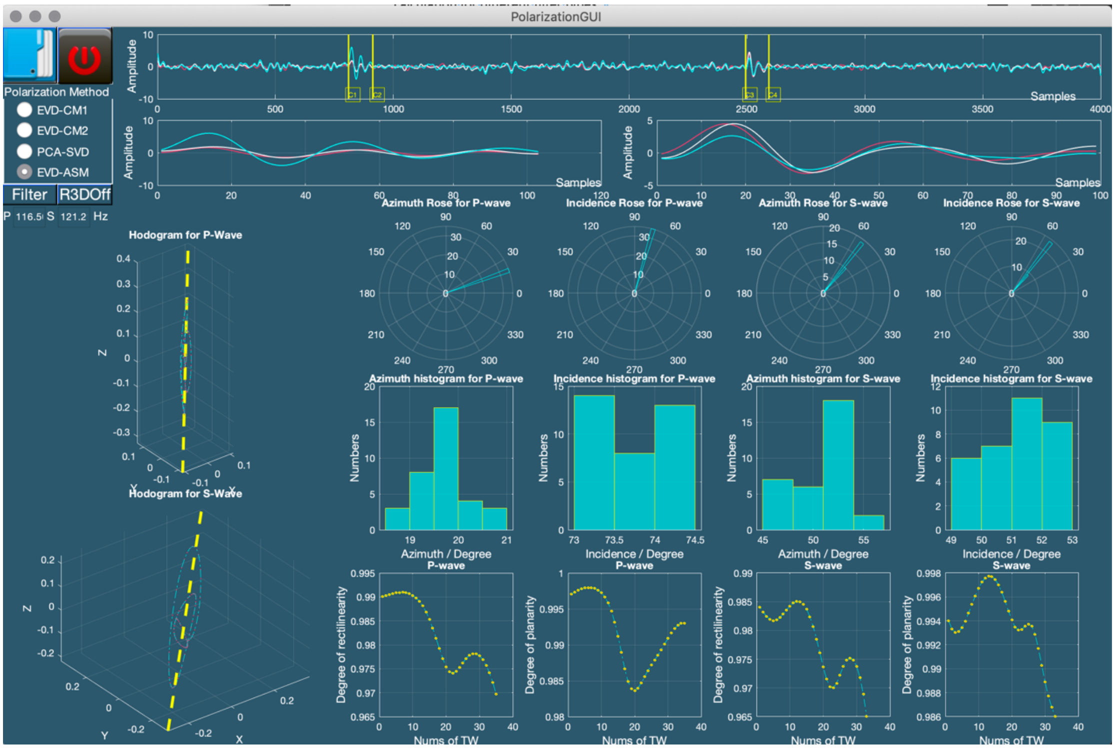

Figure 13. Polarization calculation results for noisy microseismic data filtering noise.

To perform a reasonable filter operation, we choose the LPFilter, and LPfp is 200 Hz, LPfs is 220 Hz, as exhibited in the **Figure 12**. Clicking on the 'OK' button, a polarization calculation results after being filtered is shown in **Figure 13**.

Note that the cut off frequency of the filter should meet with Nyquist criterion, it means that the $2\pi * F_c$ should be less than Fs:

$$2\pi * F_c < Fs,$$

in which F_c is the cut off frequency, Fs is the sample rate.

We recommend that F_c should be far less than Fs. In other words, LPfp and LPfs for LPFilter, Fs1, Fp1, Fs2 and Fp2 for BPFilter should be far less than Fs.

6. How to modify this code by yourself

6.1 Important functions or parameters instructions in main program

Functions	Instructions
pb_openfile_Callback	Open file, which you can add or modify different data format, and add or modify the open way of single or multiple files.
pb_exit_Callback()	Close the PolarGUI
pb_filter_Callback()	Filter callback function (call the filter GUI)
btp_SelFcn(source,eventdata)	Polarization method choose
DrawAllGraph()	Draw all polarization graphs
Drawwave(ha,alldata)	Draw original seismic 2d waveform; ha is axes, alldata is the specific data
DrawCutwave(ha,alldata)	Draw selected seismic 2d waveform; ha is axes, alldata is the specific data
Draw3DPolarizTraj(ha,X,Y,Z,titlename)	Draw 3D hodogram; ha is axes, X,Y,Z, are 3C seismic data. titlename is titlename of the hodogram
DrawAzimuthHist(ha,azi,titlename)	Draw azimuth histogram
DrawIncidenceHist(ha,inci,titlename)	Draw incidence histogram
DrawDlp(ha,Dlp,titlename)	Draw rectilinearity
DrawDlp(ha,Dpp,titlename)	Draw planarity
pb_r3d_Callback()	Open or close 3D mode
polar_plot_own.m	Plot azimuth rose graph in the GUI
polar_plot_own2.m	Plot azimuth rose for sub-graph
polar_plot_In_own.m	Plot incidence rose graph in the GUI
polar_plot_In_own2.m	Plot incidence rose for sub-graph

Parameters	Instructions
g_ntrace	Seismic recording traces
g_nslen	Seismic data length of single trace
g_nsfre	Sample period in second
g_seisdata	All original seismic recordings
g_filterdata	All seismic recordings after filtering
g_PrinFreP	Dominant frequency of the front selected seismic data
g_PrinFreP	Dominant frequency of the latter

	selected seismic data
g_Pcursorb/g_Scursorb	Cursor handle for the begin of p-wave/s-wave selected window
g_Pcursore/g_Scursore	Cursor handle for the end of p-wave/s-wave selected window
g_ptb/g_stb	Begin point of p/s-wave selected window
g_pte/g_ste	End point of p/s-wave selected window

6.2 Add or Modify different polarization calculation methods by yourself

Step 1: Create a new polarization calculation code in Matlab, and ensure that the input and output are as follows:

[azi,inci,maxeig,Dlp,Dpp] = Function_Name (X,Y,Z,delt,ttot,twin),

in the left outputs: azi is azimuth, inci means incidence, maxeig represents max eigenvalue;

in the right inputs: X, Y, Z represent three components of the seismic recording, which correspond to the R (Radial), T (Transverse) and Z (Vertical) component, respectively; delt means the sample period in second, ttot is the TW length of the entire selected window, e.g., ttotp = (g_pte-g_ptb+1)*delt; twin is the TW length of the calculated window for polarization calculation per cycle, e.g., twinp = (wndop)*delt, in which wndop is the TW samples per cycle.

Step 2: Add the created ex.m file to the folder, and call it using 'switch g_polarimod' in the DrawAllgraph() function.

6.3 Polarization calculation time window (TW)

To develop a universal software tool, we provide an automatic TW determination method combing the principal frequency of the signal, the TW calculation is as follows:

cycs = 2; % number of cycles (2 to 3 is usually sufficient)

```
    if(g_PrinFreP < 0.5) % may be natural earthquake (g_PrinFreP is the principal frequency)
      wndop = 200; % TW for P-wave in samples
      wndos = 200; % TW for S-wave in samples
  end
  if (0.5<=g_PrinFreP)%1Hz,May be natural earthquake or low-frequency microseismic
      wndop = floor( (1/g_PrinFreP) * (1/delt) )*cycs; % samples per cycle times
      wndos = floor( (1/g_PrinFreS) * (1/delt) )*cycs; % samples per cycle times
  end
  %-----------------------------------------------------------------------
      twinp = (wndop)*delt;
      twins = (wndos)*delt;
```

where the twinp and twins are applied to the final TW selection in different polarization calculation subprogram function:

```
    nsall = fix(ttot/delt);      % total number of samples to analyze
    nswin = fix(twin/delt);    % number of samples in a time window
    npshift = 1;    % number of samples to shift over (recommend)
    kfin = fix((nsall-nswin)/npshift); % number of time windows considered
```

For example, the users can modify the nwin = 1 (original nwin = fix(twin/delt) to make the polarization information obtained at any point of the seismogram.

6.4 Digital filter design using window function

In order to avoid the phase distortion of the original seismic dataset, the DFGUI was developed employing finite impulse response (FIR) filter, which possesses rigorous linear phase feature. We use different window functions to achieve the FIR filter design, these window function consists of Hamming window, Gauss window, Kasier window, Hanning window and Blackman window. This procedure involves a 'filtfilt' function which is provided by the Matlab, it can perform a zero-phase digital filtering, which ensure that it does not affect the phase of the original seismic signal.

To modify the filter, please open the FilterDlg.m file, locating the ‘switch g_ftype’. First, to determine the filters design parameters according the aforementioned section 5.2, the specific codes of the low-pass filter (LPFilter) are as follows:

```
fs = str2num(get(handles.ed_fs,'string'));% transfer the sample rate in Hz
fcl = str2num(get(handles.ed_lpfp,'string')); %the edge frequency of passband in Hz
fch = str2num(get(handles.ed_lpfs,'string')); %the edge frequency of stopband in Hz
wp = 2*pi*fcl/fs; %normalization angular frequency for the passband
ws = 2*pi*fch/fs; %normalization angular frequency for the stopband
wn = (wp+ws)/(2*pi); % cut off frequency;
Bt = ws-wp; % transition Bandwidth
N0 = ceil(6.2*pi/Bt)+1; %filter orders
N = N0+mod(N0+1,2);
if N>101 N=101; end % Keep order <= 101
winHan = hanning(N); %returns the N-point symmetric Hanning window
winHam = hamming(N); %.....
winBlkman = blackman(N); %.....
winGauss = gausswin(N); %.....
[n,Wn,beta,ftype] = kaiserord([20 25],[1 0],[0.01 0.01],100);
            winKs = kaiser(n+1,beta); %.........
%------------------------------------------------------------------------
```

After obtaining different window functions, use the ‘fir1’ function to obtain Filter coefficients for different window functions

```
[b,a] = fir1(N-1,wn,'low',winHam);

[b,a] = fir1(N-1,wn,'low',winGauss);

………….;
```

Then, using the ‘filtfilt’ to achieve the filtering operation:

```
filtfilt(b, a, g_filterdata);
```

This procedure involves a ‘filtfilt’ function which is provided by the Matlab, it can perform a zero-phase digital filtering, which ensure that it does not affect the phase of the original seismic signal.

As for the Butterworth filter, it is only designed to natural earthquake recording which it has an extremely low frequency (such as 0.001~1Hz). The designed filter by M. Reiss and G. Rümpker (2016) was employed to achieve the filter operation for natural earthquake recording, which we think the developed Butterworth filter can give an efficient filter result. Certainly, it also can be applied to the microseismic recordings, but it will produce a not very satisfactory result. Hence, we recommend that you should use the window-function FIR filter for the microseismic recordings. Furthermore, the corresponding parameters of the Butterworth filter are same to the window-function FIR filter window-function FIR filter.

Generally, the design procedure of the band-pass filter (BPFilter) is similar to that of the LPFilter, we have provided users with a specific code in detail.