SDpy Guidance

SDpy has a variety of selectivity in calculating the stress drop, methodologically we can choose to use the empirical Green's function method or the spectral fitting method. From a modeling perspective, we can choose among the Brune model, the Boatwright model, or a variable source model. And in terms of the type of wave, we can choose p wave, s wave, or coda wave. At the same time, we consider the stress drop calculation in the case of multiple stations and single station, multiple EGFs and single EGF, three channels and single channel, etc., which is characterised by wide coverage, inclusiveness and ease of use.

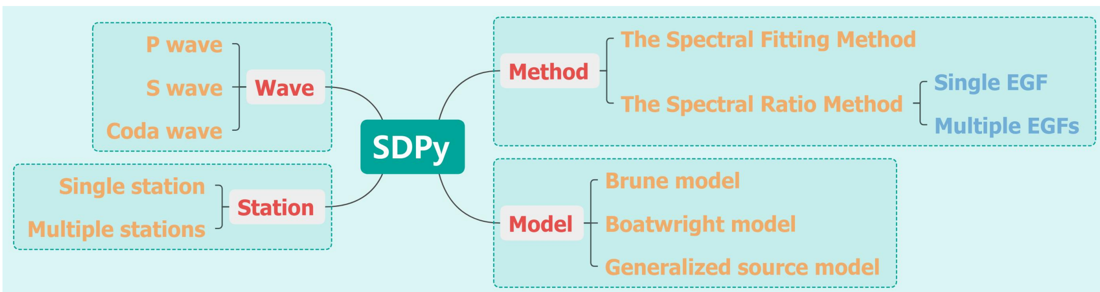

Figure 1. Overview of SDpy

1. Installation

There are two methods available for installing SDpy and it is recommended to create a virtual environment before installing it.

> conda create --name SDpy python==3.8

> conda activate SDpy

You can simply download the ZIP file and navigate to the directory and type:

> pip install .

or you can create a virtual environment from env.yml without the need for additional steps. Execute the following command in your terminal, and conda will automatically set up the environment required for SDpy:

> conda env create -f env.yml

2. Overview of SDpy

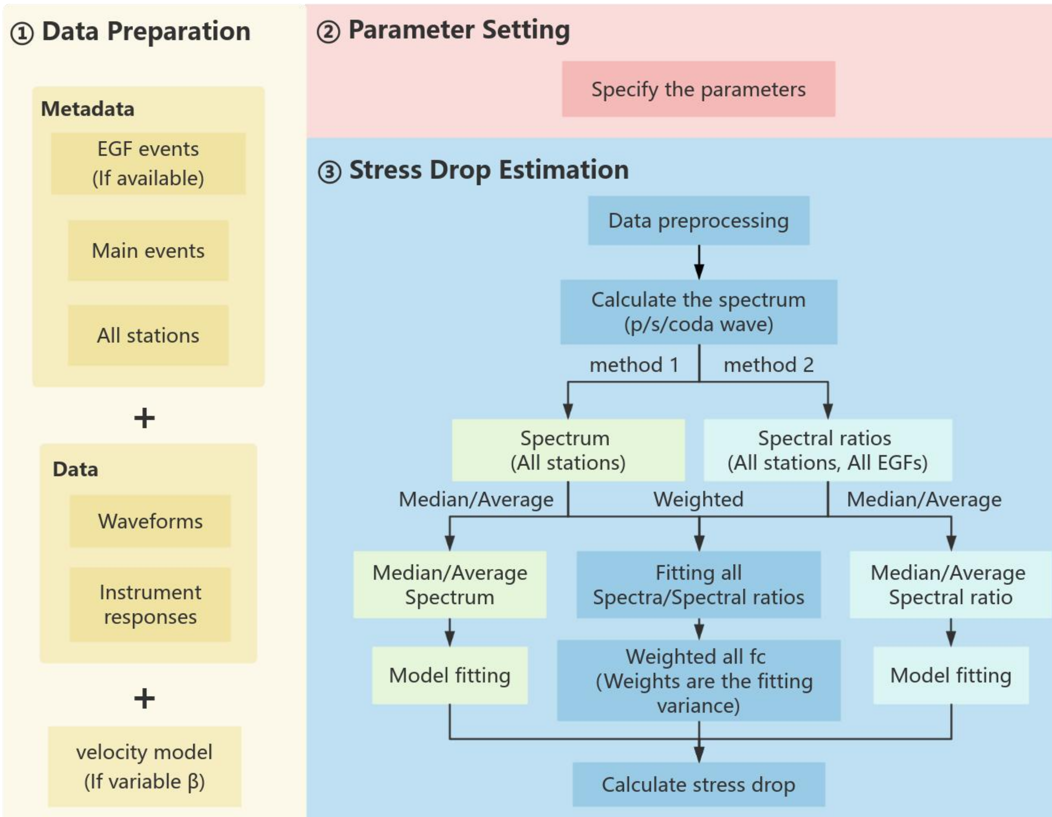

Figure 2. Workflow of SDpy

Input

All_stations A CSV file containing all the stations involved in the stress drop calculation.The path of the file will be read as a parameter by SDpy.

> **Note:** The file "***All_stations.csv***" contains three columns: 'Network' lists the names of all networks, 'Stations' lists the names of all stations, and when plotting, the curves corresponding to the stations specified in 'Plot stations' will be plotted. If 'Plot stations' is empty, the spectral ratio curves or spectral curves of all stations will be plotted by default.The header of this file is as follows:

Networks	Stations	Plot stations

Target_events_information A CSV file containing all the information of the target events.The path of the file will be read as a parameter by SDpy.

Note: The "***Target_events_information.csv***" file must include the Event ID, Original time, Seismic moment, Local magnitude, Latitude, Longitude, and Depth of the seismic events.The header of the file is as follows:

Event ID	Origin time	Moment	Mag	Lat	Lon	Dep

EGF_events_information A CSV file containing all the information of the EGFs .The path of the file will be read as a parameter by SDpy.

Note: The file header is consistent with that of the target events.And if the spectral fitting method is used to calculate the stress drop, the file or the path to this file in the parameter control file can be left blank.

Velocity_model A CSV file containing the velocity model of the study area.The path of the file will be read as a parameter by SDpy.

Note: If the stress drop calculation is to be performed according to the velocity model, the path to the "***Velocity_model.csv***" file needs to be provided as a parameter for SDpy to read, and the header of this file is as follows:

Dep(km)	V_s(m/s)

control_file Parameter control file, all parameters are stored in a JSON file as a dictionary of key-values. The path to this file is read as input and SDpy updates all parameters and performs calculations based on them.

./examples/example1/control_file.json

Data The data needs to be in a format readable by ObsPy, such as SAC or MSEED.The path of the data will be read as a parameter by SDpy.

Note: All data needs to be saved in different directories by station. The data file names need to begin with "Event ID" , and use an underscore '_' to separate it from other information, for example, '21402158_BP.CCRB..DP1.SAC'.
You can also use the DownloadData module of SDpy to download data. SDpy will retrieve waveform data and instrument responses for all stations in ***All_stations.csv*** based on the ***Target_events_information.csv*** and, if available, the ***EGF_events_information.csv***.

Response The instrument response files should also be in a format readable by ObsPy, such as XML or RESP. The path of the instrument response files will be read as a parameter by SDpy.

Note: The corresponding file name of the instrument needs to contain the station name., for example, 'BP.CCRB.xml'.

Output

1. The fitting result

The fitting result of each target event spectrum or spectral ratio will be exported in ***PDF*** format.

2. The Stress drop

The estimated stress drop result will also be saved as an ***XLSX*** file in the output directory.

Columns description of the ***XLSX*** file:

Columns name	**Description**
Target	Target event ID or name.
Origtime	Origin time of the target event (date-time format).
LON	Longitude of the target event.
LAT	Latitude of the target event.
DEP	Depth of the target event.
EGF	EGF event ID or name.
Origtime_egf	Origin time of the EGF event.
wavetype	Wave type used for spectral analysis (e.g., P, S, or coda).
fc(target)	Estimated corner frequency of the target event.
fc_err	Uncertainty of the estimated corner frequency.
fc(egf)	Estimated corner frequency of the EGF event.
Moment (Nm)	Seismic moment of the target event.
Stressdrop (MPa)	Estimated stress drop.
SD_err (MPa)	Uncertainty of the stress-drop estimate.
Radius (m)	Estimated source radius.

* The columns shown in blue are related to the Single EGF method and are displayed only when the EGF approach is applied.

Steps to use SDpy

1. Data Preparation

(1) Metadata

Before using SDpy, please ensure that you have prepared the metadata files for the target earthquakes, the metadata files for the EGF events (if available), and the stations information file, all of which are listed in the Input section.

(2) Waveforms and Responses

If the waveform data and instrument response files have not yet been prepared, they can be obtained by calling the ***download_data*** function from the ***DownloadData*** module.

```
#If you need to download data through SDpy
from SDpy import DownloadData as dd
Target_events = './examples/example1/Input/Target_events_information.csv'
data_path = './examples/example1/Data(test)'
All_stations = './examples/example1/Input/All_stations.csv'
resp_path = './examples/example1/Response'
data_center = 'NCEDC'
dd.download_data(Target_events, data_path, All_stations, resp_path, chan = 'DP*', data_center =
data_center, EGF_events = None, vel_model = 'ak135', pre_event_time = 0.5, past_event_time = 1.5)
```

Parameters

Target_events	str, path object or file-like object
	The path of 'Target_events_infomation.csv'.
data_path	str, path object or file-like object
	Path of Your data.
All_stations	str, path object or file-like object
	The path of 'All_stations.csv'.
resp_path	str, path object or file-like object

	Path of the instument response files.
chan	str
	The channel of the data that needs to be downloaded, such as 'EH*'.
EGF_events	str, path object or file-like object, default None
	The path of 'EGF_events_infomation.csv' (If available).
data_center	str, default 'IRIS'
	The data center name, such as 'NCEDC'.
vel_model	str, default 'ak135'
	Velocity model used for computing theoretical travel times.
pre_event_time	float, default 0.5
	Time duration from the start of the waveform data to the origin time of the event (in minutes).
past_event_time	float, default 1.5
	Time duration from the end of the waveform data to the origin time of the event (in minutes).

(3) Velocity model

If stress drop estimation requires a velocity model, the corresponding velocity model file must be prepared in advance.

2. Parameter Setting

SDpy reads the ***control_file.json*** file to configure the parameters for stress drop estimation. Users are required to prepare this file in advance and modify the parameters according to their specific needs.

3. Stress Drop Estimation

Once all input files are prepared, SDpy can be used to estimate the stress drop of the target earthquakes. This is achieved by calling the ***Stressdrop*** function within the ***StressDrop*** module. The specific usage is as follows:

- **In Python Editor**

You can import the ***StressDrop*** module of SDPy within a Python editor to accomplish the estimation of stress drop.

```
#If you have prepared seismic waveform data and instrument response files
from SDpy import StressDrop as sd
results=sd.stressdop('./examples/example1/control_file.json')
```

- **In the Terminal**

You can also specify the path of the '***control_file.json***' in the terminal and run ***SD.py*** to achieve the estimation of the stress drop.

```
> python SD.py ./examples/example1/control_file.json
```

> **Note:** For the parameter types and their meanings in ***control_file.json***, please refer to the Parameters section.

3. Parameters

Target_events str, path object or file-like object

The path of 'Target_events_infomation.csv'.

./example/Input/Target_events_infomation.csv

EGF_events str, path object or file-like object

The path of 'EGF_events_infomation.csv'.

./examples/example1/Input/EGF_events_infomation.csv

All_stations str, path object or file-like object

The path of 'all_stations.csv'.

./examples/example1/Input/All_stations.csv

data_path str, path object or file-like object

Path of Your data.

./examples/example1/Data/

resp_path str, path object or file-like object

Path of the instument response files.

./examples/example1/Response/

out_path str, path object or file-like object

Save path of the calculation results and the relevant files.

./examples/example1/Output/

method int, default 1

The method you wish to use to calculate the stress drops.

method = 1: The spectral fitting method;

method = 2: The spectral ratio method.

wv str, default 'p'

The wave type used to compute spectrum. Choose from [‘p’, ‘s’, ‘coda’], which represent p wave, s wave, and coda wave, respectively.

wave_align str, default ‘cc’

Target event and EGF event waveform alignment method. ‘cc’ represents alignment using the cross-correlation method, and ‘mark’ represents alignment using marked arrival times. Choose from [‘cc’, ‘mark’].

fixed_window float, default None

By default (*fix_window* = *None*), SDpy determines the time-window length for spectral calculations based on the seismic moment of each event following *Abercrombie* (2017). If the seismic moment is not provided, the window length is estimated from the local magnitude following *Ruhl et al.* (2017). Once specified, the fixed window used for calculating the seismic event spectra for a given dataset; when using multiple windows to compute the spectra, it represents the length of the sub-window. The total length of the window equals three times fixed_window (the whole lenth of wave = *fixed_window* * (*num_window* * (1 - *overlap*) + *overlap*).

time_add float, default 1.0

The noise ending time before P wave.

num_windows int, default 1

Number of time windows used to calculate spectra.

overlap float, default 0.0

The proportion of overlap for constructing the spectra ($0 \leq overlap \leq 1$).

T_coda float, default 3.0

Coda wave start time, at *T_coda* times the S wave travel time starting from the origin time of the earthquake. We default *T_coda* to 3 according to *Wu et al.* (2017).

source_model str, default ‘sm’

Source model type. Choose from [‘b’ , ‘fb’, ‘sm’], which represent Brune model (for $n = 2$, $\gamma = 1$), fixed Boatwright model (for $n = 2$, $\gamma = 2$), generalized source model (for $1.5 \leq n \leq 3$; $1 \leq \gamma \leq 2$), respectively. We have opted for the gernalized source model by default.

snrthres float, default 2.0

Signal-to-noise ratio threshold, only spectra with a signal-to-noise ratio greater than *snrthres* will be used to calculate the stress drop. By default, we set *snrthres* to 2, but you can customize it based on your specific requirements.

stack_method str, default ‘median’

For each target event, the stack method for all calculated spectral ratio curves or spectrum curves. Choose from [‘average’, ‘median’, ‘weighted’]. When choose ‘weighted’,the fitting variance are used as weights.

fit_method str, default ‘L-BFGS-B’

The optimization algorithms for fitting observed amplitude spectra or spectral ratios. All commonly used methods in scipy.optimize.minimize can be employed.

fit_freq_range list, default None.

Specify the frequency range of the fit if needed.

fit_amp_ratio float, default 2.

The difference in amplitude between low- and high-frequency spectral components in the fit The parameter is only used when the EGF method is applied.

k float, default 0.21 or 0.32

The constant depending on assumed source model. According to *Madariaga* (1976), we default to $k = 0.32$ when calculating stress drop using P-waves, and $k = 0.21$ when using S-waves/coda waves.

beta float, default 3500.0

Shear wave velocity (Unit: m/s)

Q int or float or list, default [200, 2000]

The quality factor. When Q is specified as a single value (either an integer or a float), it is directly used in the fitting process. When Q is defined as a range (a list), the optimal value is searched within the given range.

mode int, default 1

The mode of calculating stress drops when there are multiple EGFs for the target event.

mode = 0: Calculate the stress drop for seismic events under each EGF separately;

mode = 1: Calculate the mean of the stressdrop for seismic events under all EGFs.

chan str, default '*'

The channel used, by default, uses all available channels.

remove_resp str, default 'no'

Whether to remove the instrument response or not. Choose ‘yes’ or ‘no’.

fs_cor str, default ‘no’

Whether to apply free surface correction. Choose ‘yes’ or ‘no’.

fs_vp float, default 6000

P-wave velocity at the free surface (Unit: m/s).

fs_vs float, default 3500

S-wave velocity at the free surface (Unit: m/s).

fs_factor float, default 2.0

The free surface factor.

rho float, default 2700.0

Source region rock density (Unit: kg/m^3).

c float, default $c_S = \beta$ and $c_P = 1.73 * \beta$

The propagation velocity of seismic waves (Unit: m/s). Where β is the S-wave velocity utilized for stress drop estimation.

U float, default 0.52 for P-wave and 0.63 for S-wave

The average value of the radiation pattern term ($U_{\phi\theta}$).

assume_drop float, default 3.0

Assumed stress drop used to calculate rupture radius. Generally we default to assume_drop = 3 (Unit: Mpa).

num_workers int, default 6

The number of CPU used to parallel. You can configure this parameter according to your actual circumstances.

num_tapers int, default 5

The number of tapers when using multitapers (*Prieto*, 2022) to analyze the spectrum of event.

showfc2 str, default ‘yes’

Whether to show the corner freqency of EGF event when using Single EGF method. Choose ‘yes’ or ‘no’.

4. Examples

The following four examples represent the primary calculation methods of SDpy. By examining the corresponding PDF images output for each example, you can observe the specific application effects of SDpy.

Example1

Table 1. The key parameters for example1.

method	wv	source_model	stack_method
1	‘p’	‘sm’	‘median’

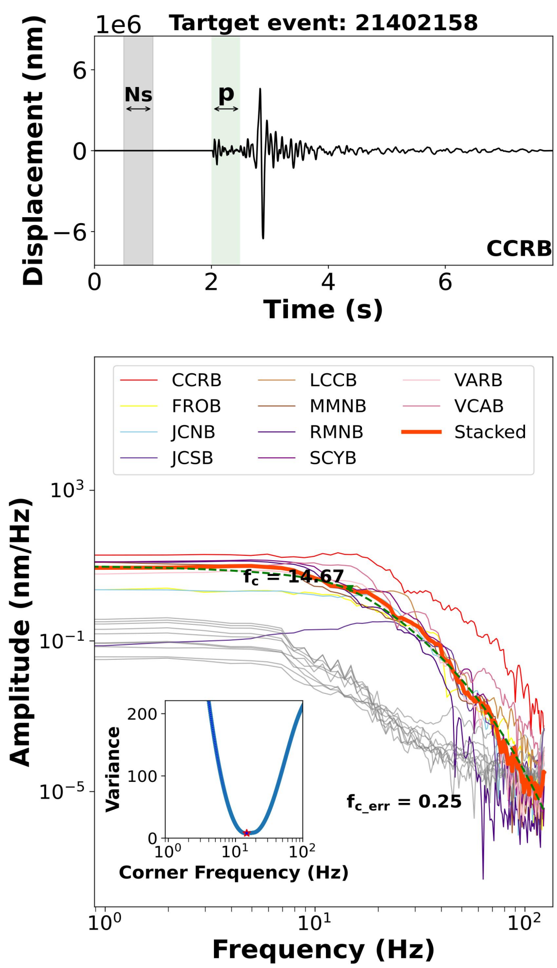

Figure 3. The spectral ratio fitting plot for example1.

Example2

Table 2. The key parameters for example2.

method	wv	source_model	mode	stack_method
2	‘p’	‘b’	1	‘weighted’

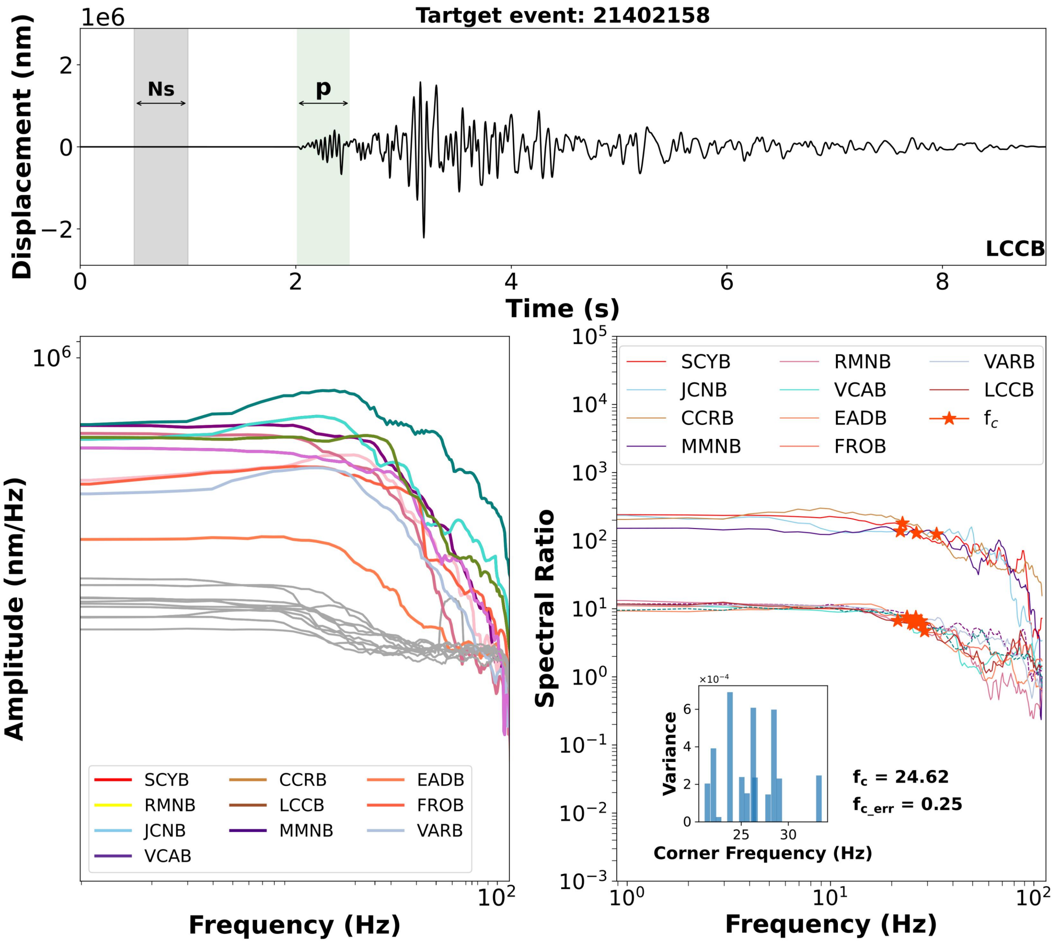

Figure 3. The spectral ratio fitting plot for example2.

Example3

Table 3. The key parameters for example3.

method	wv	source_model	mode	stack_method
2	's'	'b'	1	'weighted'

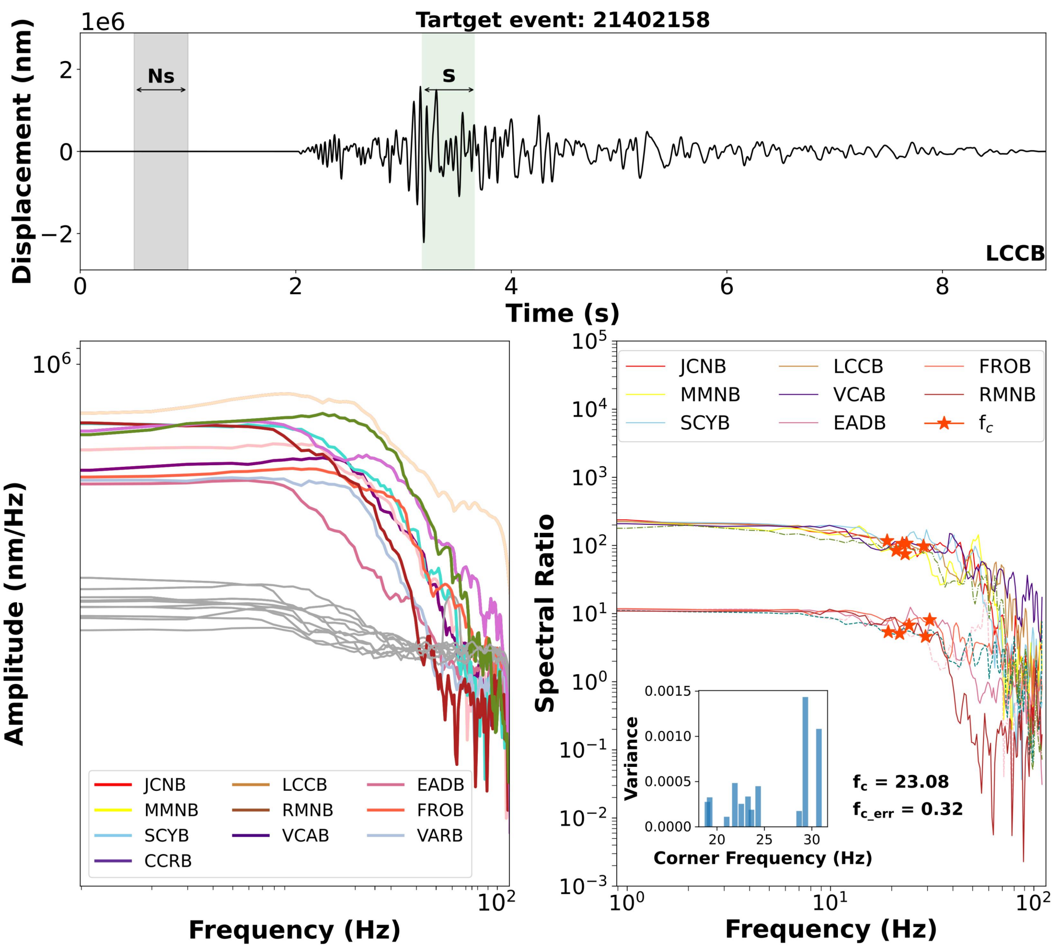

Figure 4. The spectral ratio fitting plot for example3.

Example4

Table 4. The key parameters for example4.

method	wv	source_model	mode	stack_method
2	'coda'	'fb'	0	'median'

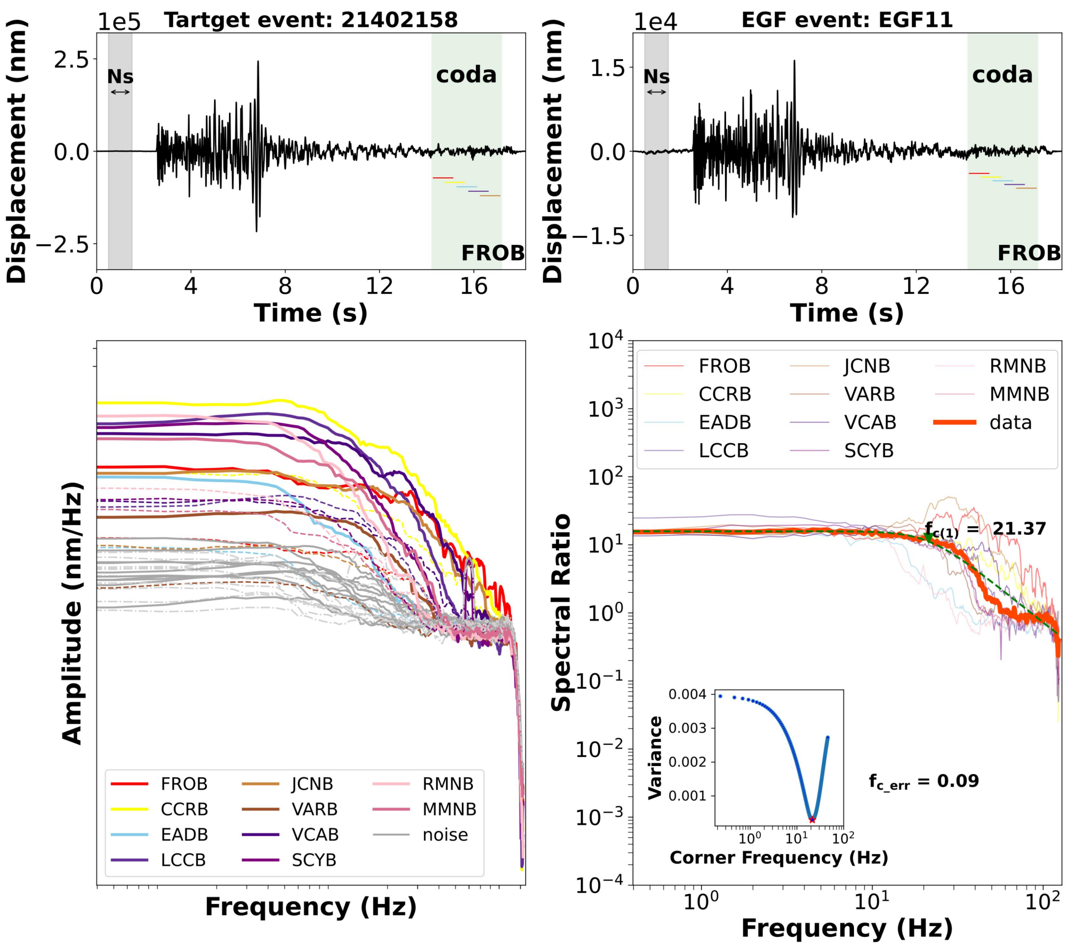

Figure 6. The spectral ratio fitting plot for example4.

Refernces

Abercrombie, R. E., Poli, P., and Bannister, S. (2017). Earthquake directivity, orientation, and stress drop within the subducting plate at the Hikurangi margin, New Zealand. *Journal of Geophysical Research: Solid Earth*, 122(12), 10-176.

Madariaga, R. (1976), Dynamics of an expanding circular crack, *Bull. Seismol. Soc. Am*., 66, 639-666.

Prieto, G.A. (2022). multitaper: A multitaper spectrum analysis package in Python. *Seis. Res. Lett.* 93(3), 1922-1929.

Ruhl, C. J., Abercrombie, R. E., and Smith, K. D. (2017). Spatiotemporal variation of stress drop during the 2008 Mogul, Nevada, earthquake swarm. *Journal of Geophysical Research: Solid Earth*, 122, 8163-8180.

Wu, Q., and Chapman, M. (2017). Stress-drop estimates and source scaling of the 2011 Mineral, Virginia, mainshock and aftershocks. *Bulletin of the Seismological Society of America*, 107(6), 2703-2720.