

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.

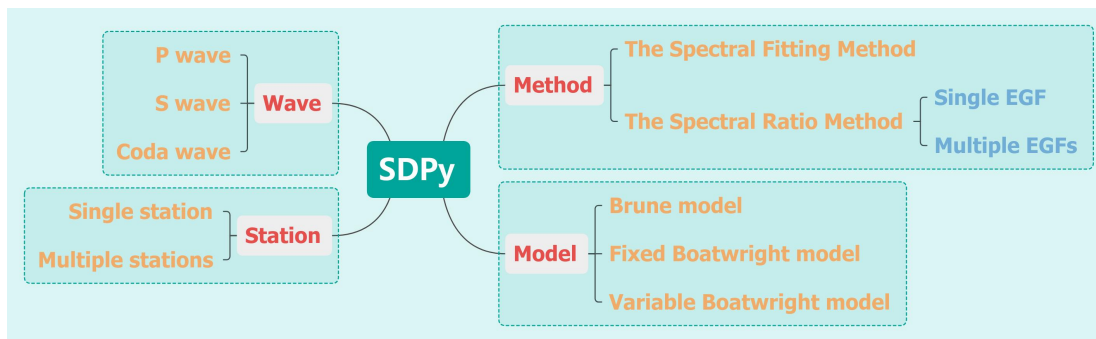


Figure 1. Overview of SDpy

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

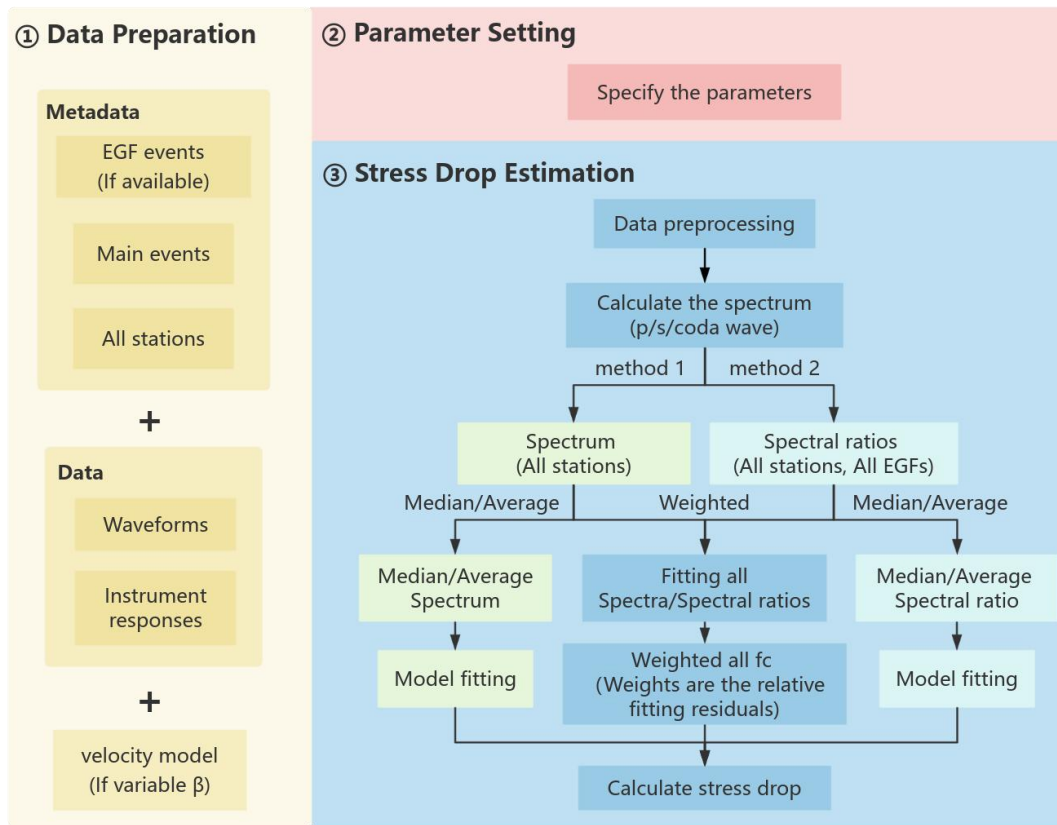


Figure 2. Workflow of SDpy

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 two columns: ‘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, Origin time, ‘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.

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', EGF_events = None,*

data_center = data_center, 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_information.csv'.
<i>data_path</i>	str, path object or file-like object Path of Your data.
<i>All_stations</i>	str, path object or file-like object The path of 'all_stations.csv'.
<i>resp_path</i>	str, path object or file-like object Path of the instrument response files.
<i>chan</i>	str The channel of the data that needs to be downloaded, such as 'EH*'. The path of 'EGF_events_information.csv' (If available).
<i>EGF_events</i>	str, path object or file-like object, default None
<i>data_center</i>	str, default 'IRIS' The data center name, such as 'NCEDC'.
<i>vel_model</i>	str, default 'ak135' Velocity model used for computing theoretical travel times.
<i>pre_event_time</i>	float, default 0.5 Time duration from the start of the waveform data to the origin time of the event (in minutes).
<i>past_event_time</i>	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.stressdrop('./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

<i>method</i>	int, default 1 The method you wish to use to calculate the stress drops. <i>method</i> = 1: The spectral fitting method; <i>method</i> = 2: The spectral ratio method.
<i>wv</i>	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.
<i>wave_align</i>	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'].
<i>chan</i>	str, default '*' The channel used, by default, uses all available channels.
<i>num_windows</i>	int, default 1 Number of time windows used to calculate spectra.
<i>overlap</i>	float, default 0.0 The proportion of overlap for constructing the spectra ($0 \leq \text{overlap} \leq 1$).
<i>fixed_window</i>	float, default 1.0 The window length used for calculating the seismic event spectra; 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 <i>fixed_window</i> (the whole length of wave = <i>fixed_window</i> * (<i>num_window</i> * (1 - <i>overlap</i>) +

overlap).

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/m³).

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 ($U_{\phi\theta}$).

snrthres float, default 3.0

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 3, but you can customize it based on your specific requirements.

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 Qimin Wu (2016).

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 to analyze the spectrum of event.

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.

sumtype str, default 'weighted'

For each target event, the summation method for each calculated spectral ratio curve or frequency spectrum curve. Choose from ['average', 'median', 'weighted']. When choose 'weighted', the relative fitting error are used as weights.

showfc2 str, default 'yes'

Whether to show the corner frequency of EGF event when using Single EGF method. Choose 'yes' or 'no'.

<i>time_add</i>	float, default 1.0	The noise ending time before P wave.
<i>min_mag_diff</i>	float, default 0.3	Minimum magnitude difference to be guaranteed between event pairs
<i>assume_drop</i>	float, default 3.0	Assumed stress drop used to calculate rupture radius. Generally we default to $\text{assume_drop} = 3$ (Unit: Mpa).
<i>k</i>	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.
<i>beta</i>	float, default 3500.0	Shear wave velocity (Unit: m/s)
<i>Q</i>	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.
<i>fit_freq_range</i>	list, default None.	Specify the frequency range of the fit if needed.
<i>mode</i>	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.

Target_events str, path object or file-like object
The path of 'Target_events_information.csv'.
./example/Input/Target_events_information.csv

EGF_events str, path object or file-like object
The path of 'EGF_events_information.csv'.
./examples/example1/Input/EGF_events_information.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 instrument 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/

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	sumtype
1	'p'	'sm'	'weighted'

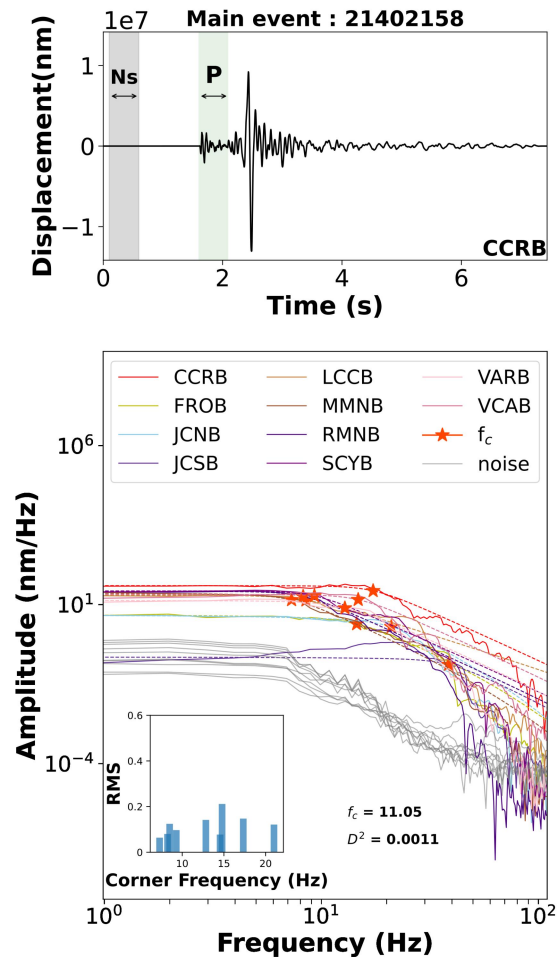


Figure 3. The spectral ratio fitting plot for example1.

Example2

Table 2. The key parameters for example2.

method	wv	source_model	mode	sumtype
2	'p'	'b'	1	'weighted'

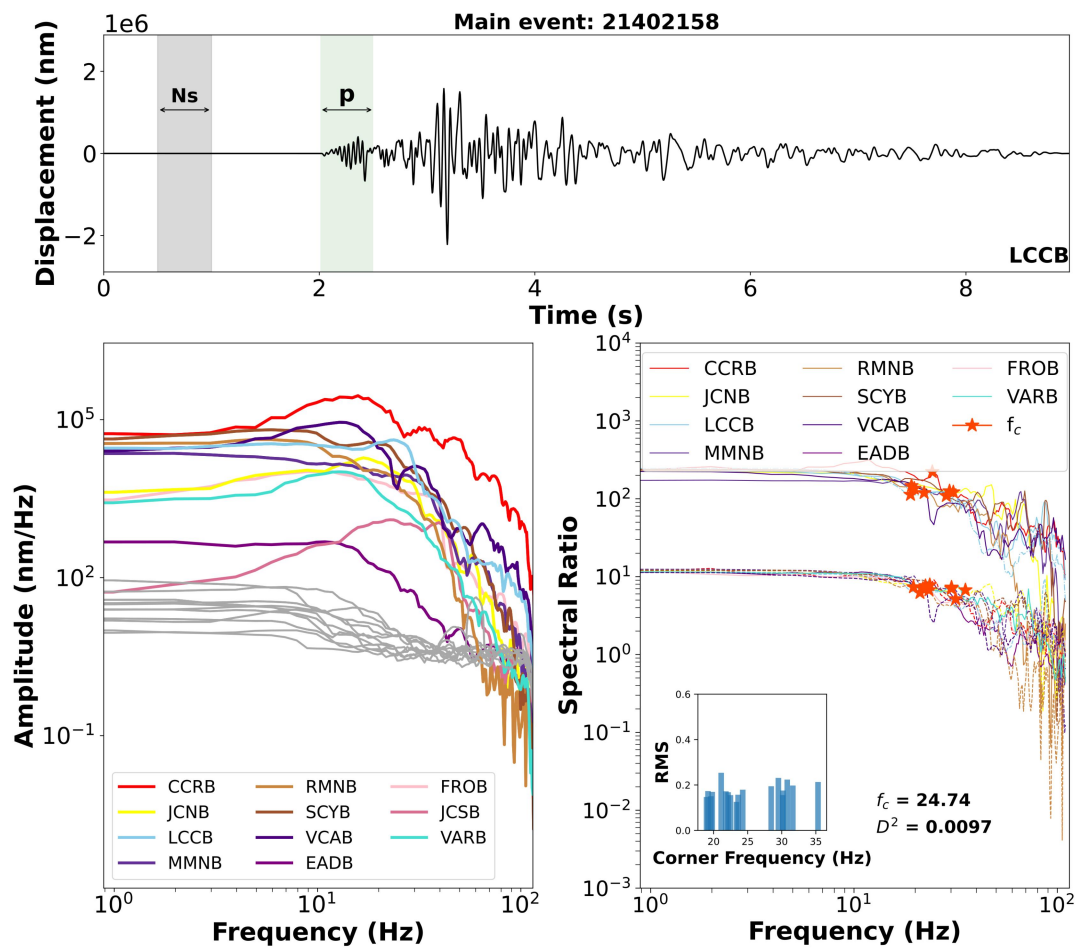


Figure 4. The spectral ratio fitting plot for example2.

Example3

Table 3. The key parameters for example3.

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

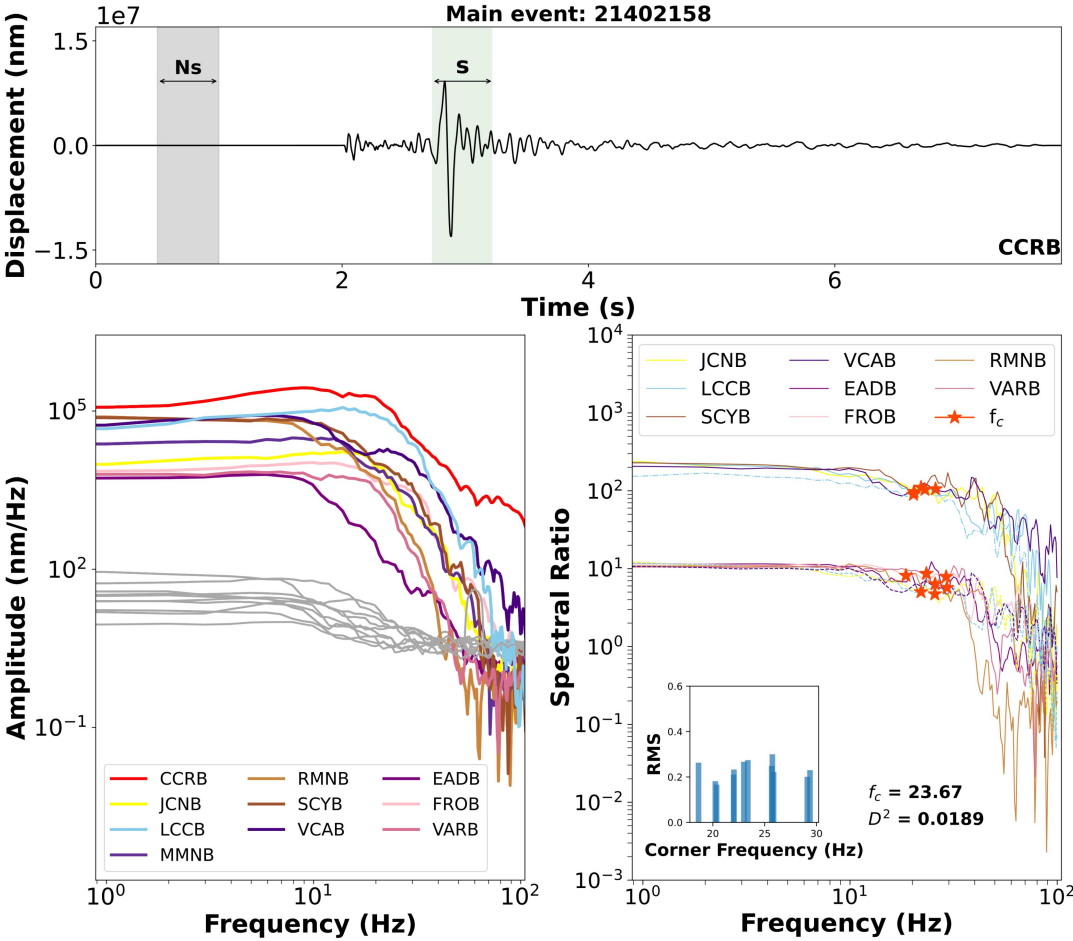


Figure 5. The spectral ratio fitting plot for example3.

Example4

Table 4. The key parameters for example4.

method	wv	source_model	mode	sumtype
2	'coda'	'fb'	0	'demean'

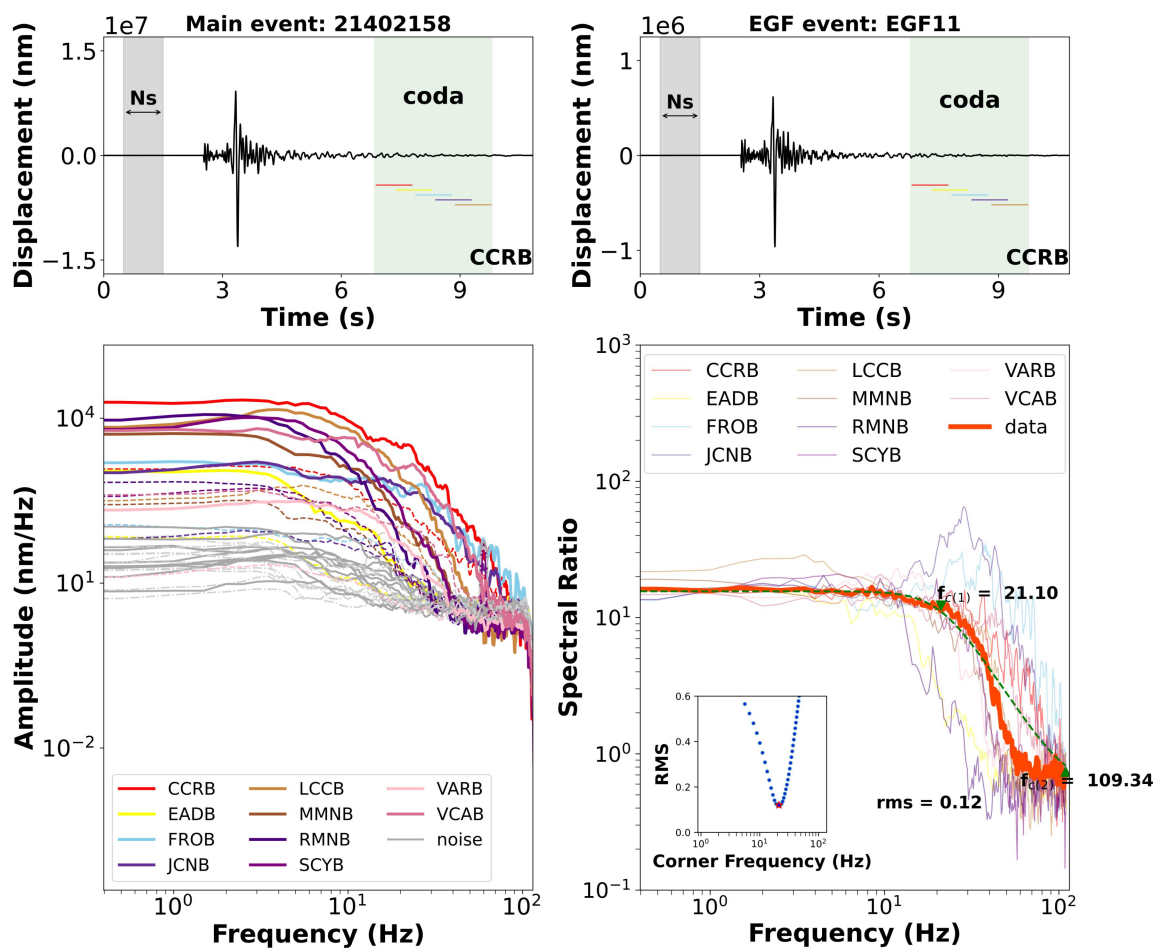


Figure 6. The spectral ratio fitting plot for example4.