Electronic Seismologist

Seispy: Python Module for Batch Calculation and Postprocessing of Receiver Functions

Mijian Xu*[1,2,3] and Jing He[4]

Abstract

Seispy is a graphical interface Python module for receiver function (RF) calculation and postprocessing in seismological research. Automated workflows of RF calculations facilitate processing large volume of different types of seismic data. The graphical user interface enables an intuitive and straightforward evaluation of RF quality. All parameters about the preprocessing for RF estimation can be adjusted based on user preference. Water-level frequency-domain deconvolution and iterative time-domain deconvolution for RF estimation are available in Seispy. The current version of Seispy contains five main modules for the postprocessing of RF, such as *H*-κ stacking, crustal anisotropic estimation, harmonic decomposition, and 2D and 3D common conversion point (CCP) stacking. The CCP stacking in the different application scenarios can be handled by a rich collection of modules, such as time-to-depth conversion, 2D or 3D CCP stacking, and adaptive station or bin selection for CCP stacking profiles in a dense seismic array or a linear seismic array. As a Python module, functions in the Seispy can be called easily in Python scripts for other purposes. The modular design allows new functionality to be added in a collaborative development environment. Licensed under GPLv3, Seispy allow users and developers to freely use, change, share, and distribute copies of the package.

Cite this article as Xu, M., and J. He (2022). Seispy: Python Module for Batch Calculation and Postprocessing of Receiver Functions, *Seismol. Res. Lett.* **XX**, 1–9, doi: 10.1785/0220220288.

Introduction

Receiver function (RF) is a time-series record calculated from three components of teleseismic seismograms, which shows the response of earth structure on the receiver side. Deconvolution equalizes the source time function, and removes the instrument response and the propagation effect to extract the RF from seismic data (Langston, 1979; Ammon, 1991). Corresponding to the complex spectral ratios of the radial–transverse (R and T) components to the vertical component (for *P*-wave RF) (Langston, 1979) or the *P* component to the *SV* component (for *S*-wave RF) (Farra and Vinnik, 2000; Yuan *et al.*, 2006), a composite of *P*-to-*s* or *S*-to-*p* converted waves is isolated from the teleseismic *P* or *S* waves, respectively. RF becomes a widely used seismic method to detect the features of the Moho, the lithosphere–asthenosphere boundary (LAB), and the mantle transition zone (MTZ) due to its sensitivity to velocity discontinuities.

The conventional workflow of RF includes RF (*P*-wave and *S*-wave RFs) calculation and postprocessing. In RF calculation, we usually perform preprocessing of the teleseismic data and deconvolution to obtain RFs. The filtering and rotation parameters in the preprocessing depend on different deconvolution methods. Many deconvolution techniques are used to obtain the RF in the frequency and time domains. Because of the random noise, inaccuracies of source function estimation, and limited frequency bandwidth, deconvolution of RF is usually ill posed with the small-amplitude spectrum. The water-level deconvolution in frequency domain finds a small water-level parameter that prohibits instability resulting from division by very small numbers (Ammon, 1991). Iterative time-domain deconvolution (Ligorría and Ammon, 1999) intuitively strips the largest receiver function arrivals from the observed seismograms, and focuses on the most important and energetic features, which is inherently more stable. Other deconvolution methods have been developed for different inverse problems and data to obtain a stable RF, and improve the resolution, such as multiple-taper deconvolution (Park and Levin, 2000; Helffrich, 2006), simultaneous deconvolution (Gurrola *et al.*, 1995; Bostock, 1998), and maximum entropy

1. Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore, Singapore, https://orcid.org/0000-0001-8888-8523 (MX); 2. Earth Observatory of Singapore, Nanyang Technological University, Singapore, Singapore; 3. School of Earth Sciences and Engineering, Nanjing University, Nanjing, China; 4. National Institute of Natural Hazards, Ministry of Emergency Management of China, Beijing, China, https://orcid.org/0000-0001-6395-6585 (JH)

*Corresponding author: gomijianxu@gmail.com

deconvolution (Wu *et al.*, 2003). Postprocessing includes H-κ stacking (Zhu and Kanamori, 2000) and common conversion points (CCP) stacking (Dueker and Sheehan, 1997; Zhu, 2000) for investigating Moho or MTZ topography, and crustal anisotropic estimation using a joint method (Liu and Niu, 2012) or harmonic decomposition (Bianchi *et al.*, 2010).

In the latest decade, as the volume of seismic data increases, receiver function techniques have been applied to image structure under dense arrays. Therefore, more efficient RF calculation tools are needed for processing of large volume of seismic data with high quality and more functions of postprocessing.

Python language has become a popular programing language in seismic data computing. In the recent years, many modules for processing different seismological methods have been derived depending on Obspy—a Python module for seismic data processing, such as Noisepy (Jiang and Denolle, 2020), CC-FJpy (Li *et al.*, 2021), BayHunter (Dreiling and Tilmann, 2019), and rf (Eulenfeld, 2020). They are improving the ecology of seismic data computation in the Python environment. Meanwhile, packages for RF calculation are also developed in different platforms, such as Matlab-based SplitRFLab (Xu *et al.*, 2016), Crazyseismic (Yu *et al.*, 2017), FuncLab (Eagar and Fouch, 2012), and Python-based rf (Eulenfeld, 2020). However, these packages lack some of the following functions: automatic preprocessing with different kinds and huge volume of seismic data, automatic RF calculation, graphical user interface for RF quality control, efficient postprocessing functions (e.g., H-κ stacking, CCP stacking, and anisotropic estimation), and easily accessible application programing interfaces (API). Here, we develop a Python module of Seispy for RF calculation to realize all these functions.

Overview of Seispy

Seispy relies on the open-source Python modules like Numpy, Scipy, Obspy, PyQt, and Matplotlib, and is developed under the GPLv3 license. The module can be easily installed via conda-forge. Once installed, Seispy provides commands for fast data processing, all of which are parameterized by configure files in Python *configParser* format, without any additional Python scripts. Therefore, the RF calculation, H-κ stacking, crustal anisotropy estimation, harmonic decomposition, and CCP stacking can be performed in a single command line.

An online documentation powered by Sphinx has been deployed (see Data and Resources section). We provide tutorials on how to obtain a RF from teleseismic records, how to visually check RFs, how to perform CCP stacking, and more. It is helpful for early career scientists to understand the theory of RF. Advanced examples involving detailed usage, choice and parameters in RF batch calculation, 2D and 3D CCP stacking, crustal anisotropy estimation, and harmonic decomposition are accessible for users. In addition, the variables and returns of most submodules, methods, and functions are also available for users' reference to facilitate their invocation in other Python scripts or modules.

The framework of Seispy includes two parts: an automatic workflow of RF calculation and submodules of postprocessing, including H-κ stacking, crustal anisotropic estimation, harmonic decomposition, slant stacking, moveout correction, time-to-depth conversion, and 2D or 3D CCP stacking (Fig. 1). For advanced users or developers, the submodules in the Seispy can be invoked in the Python scripts for further purposes.

RF Calculation

Automatic workflow of RF calculation

Users can execute *prf* or *srf* in command line for RF calculation. A configure file including all parameters for RF calculation need to be set before the RF calculation. RF calculation will work as follows (Fig. 1): (1) trimmed teleseismic records involving primary phases with SAC format need to be prepared. (2) Earthquake information with customized limitation (e.g., epicentral distance, focal depth, date range, and magnitudes) are fetched from online catalogs with the International Federation of Digital Seismograph Networks (FDSN) webservice, for which providers include U.S. Geological Survey (USGS), Incorporated Research Institutions for Seismology (IRIS), the International Seismological Centre (ISC), the Global Centroid Moment Tensor (Global CMT), and so on. Seispy associates date and time information in seismic records with acquired seismic information. The associated seismograms are saved into a database with *Pandas* format. (3) Linear and mean trend are removed, and a band-pass filter with customizable cutoff frequency are applied on seismograms. (4) The theoretical P or S arrival times, and ray parameters are then calculated by TauPy submodule in Obspy (Crotwell *et al.*, 1999; Beyreuther *et al.*, 2010) with customizable velocity models following the format required by TauPy. (5) The signal-to-noise ratio (SNR) is calculated with the following formula:

$$\mathrm{SNR} = 10 \log_{10}\left(\frac{\mathrm{A}_\mathrm{S}^2}{\mathrm{A}_\mathrm{N}^2}\right), \quad (1)$$

in which A_S and A_N are time series after and before theoretical P arrival, respectively. The time length can be set in the configure file. A customized threshold is used to remove events with low SNR from the database. Optionally, the database can be saved as a binary file for recalculation. (6) The waveforms are rotated from north–east–vertical (NEZ) components to transverse–radial–vertical (TRZ) or P–SV–SH (LQT) components. (7) Waveforms are trimmed around theoretical P or S arrival calculated in step (4). (8) R components are deconvoluted by Z components with time-domain iterative deconvolution or water-level deconvolution (Langston, 1979; Ammon, 1991; Ligorría and Ammon, 1999). See online documentation for more parameters in deconvolution. (9) Some criteria can be selected for automatic quality control. For example, root mean square between original and recovered (convolution of RF and Z component) waveform in radial component should be less than a

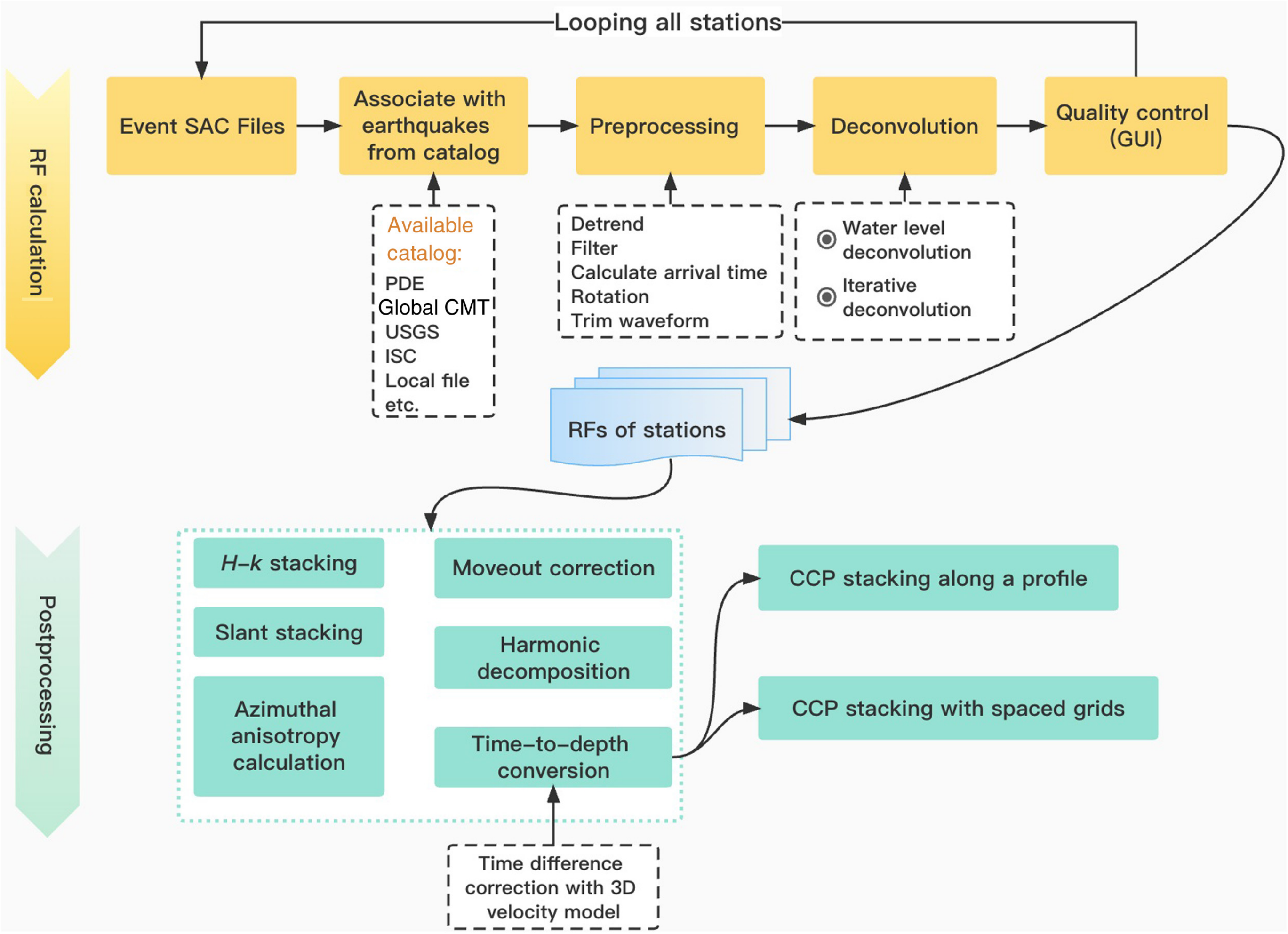

Figure 1. General workflow for Seispy. CCP, common conversion point; Global CMT, Global Centroid Moment Tensor catalog; ISC, International Seismological Centre; PDE, Preliminary Determination of Epicenters Bulletin; RF, receiver function; and USGS, U.S. Geological Survey National Earthquake Information Center. The color version of this figure is available only in the electronic edition.

customized value. The average amplitude after 30 s should be less than a specified proportion of direct *P* amplitude. Finally, RFs are saved to SAC files after the automatic workflow. Users can choose different parameters in the preprocessing for RF calculation. Meanwhile, we can use a script for RF calculation in batch.

Manual quality control with a GUI

PyQt is a very popular Graphical User Interface (GUI) development tool, and Matplotlib is an excellent scientific plotting toolkit. In the Seispy, we plot waveforms with the Matplotlib and embed them into the PyQt interface. We have developed two interfaces to manually pick good seismograms for RF calculation and good RFs, respectively. One GUI is developed to pick arrival time and reject seismograms with low quality (Fig. 2). After the step (5) of the RF calculation workflow, this GUI may be open with setting a command-line option. Users can pick the *P*- or *S*-wave arrival time with a simple mouse click, and manually remove seismograms from the database via pushbuttons or shortcut keys. Because the SNR of *S* wave is much less than that of *P* wave, this manual quality control is essential for *S*-wave RF calculation.

After the RF calculation workflow, we designed a GUI to manually check qualities of RFs. This GUI can be opened with the *pickrf* command with RFs arranging by backazimuth or by epicentral distance (Figs. 3 and 4). Users can mark a waveform with low quality by direct mouse click. Top buttons and shortcut keys are used to page down or up. Users can press the spacebar to preview RFs without marked waveforms in a new window. Finally, pressing “Finish” button will reject the RFs marked as low quality and save a list of RFs with good quality in a text file named “finallist.dat”. The “finallist.dat” includes the information of the RFs such as epicentral location, backazimuth, ray parameter, and Gaussian factor. The good RFs in the text file of “finallist.dat” will be used for the postprocessing. Figures 3 and 4 show the main windows for visually picking *P* and *S* RFs, respectively. The *plotrt* command can plot the RFs arranged by backazimuth or by epicentral distance like the Preview windows in Figures 3 and 4.

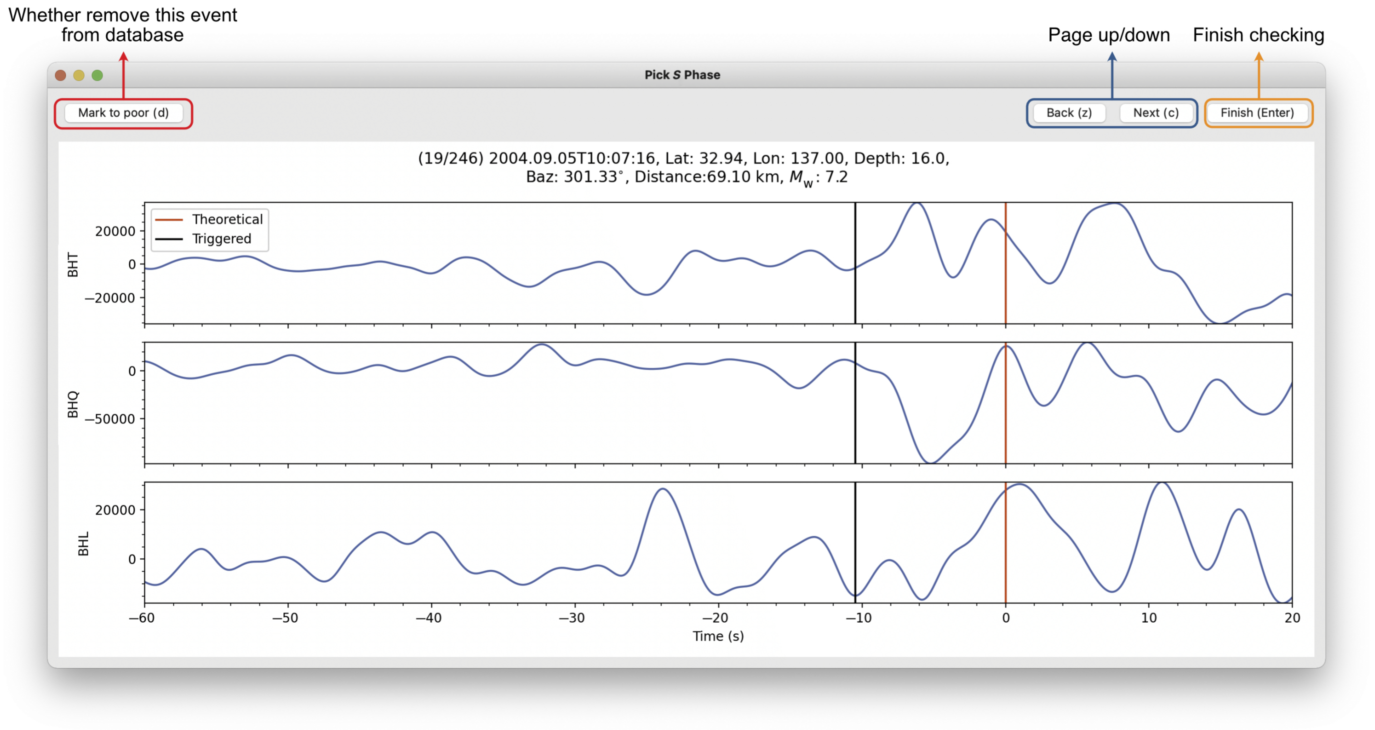

Figure 2. Arrival time-picking interface. This is an example for picking *S* arrival by station YKW3 of network CN in Canada. The red line represents the theoretical arrival time, and the black line is new picked arrival time. The color version of this figure is available only in the electronic edition.

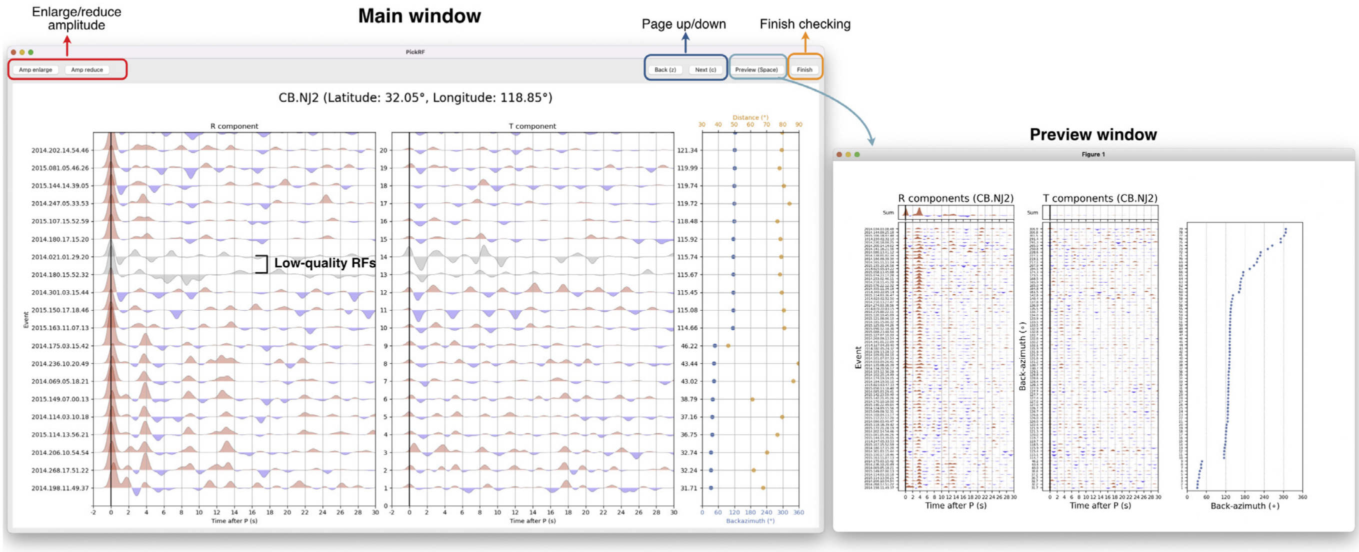

Figure 3. Main view for visually checking *P*-wave RFs in radial and transverse components. Main window is used for picking good *P*-wave RFs. The Preview window shows the RFs arranged by backazimuth. The example data are the station NJ2 of network CB in China. The color version of this figure is available only in the electronic edition.

Postprocessing

Because the RF is sensitive to velocity discontinuities, some mature techniques based on RF are used to investigate different properties of the earth's interior, like the topographies of velocity discontinuities in the earth's interior and the crustal anisotropy. We have integrated the *H*-κ stacking, crustal

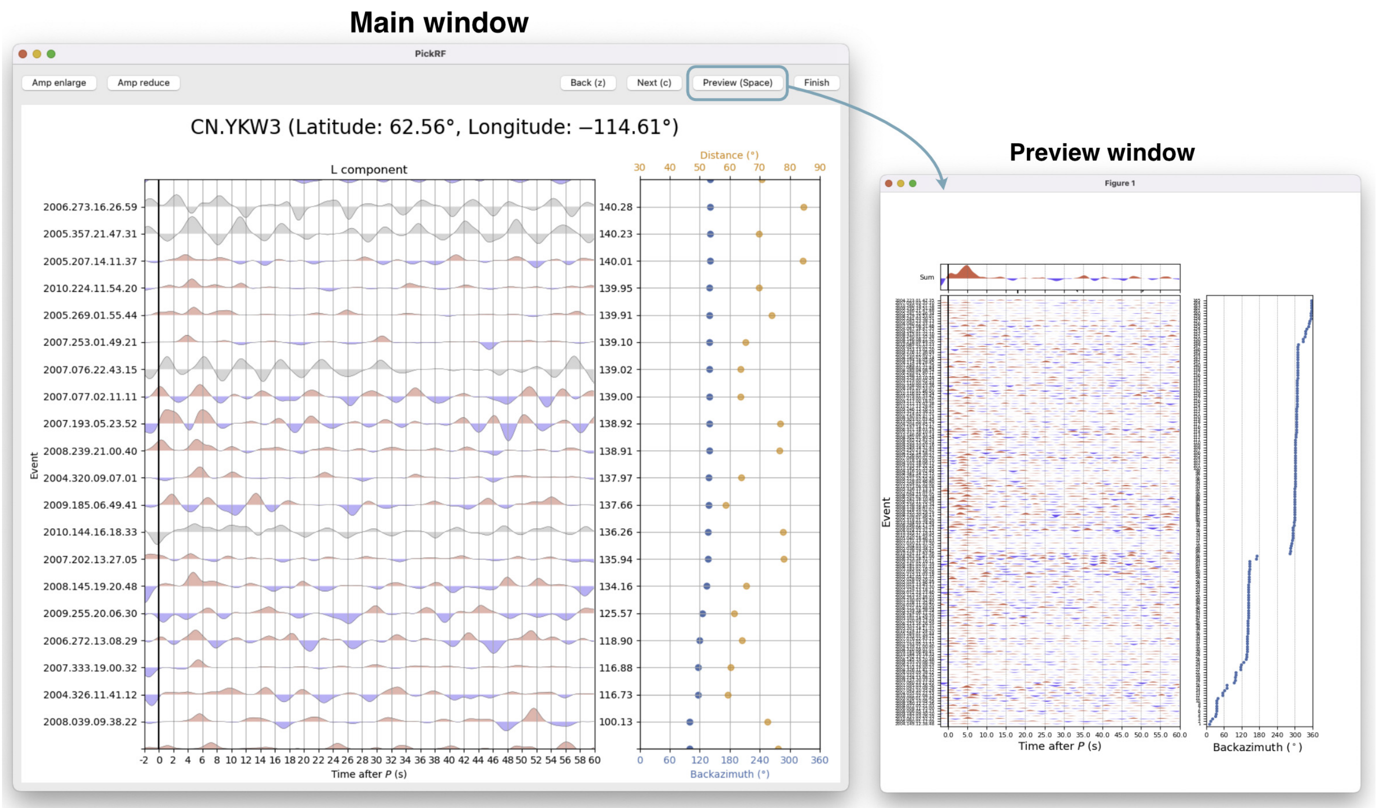

Figure 4. Main view for visually checking *S*-wave RFs. Main window is used for picking good SRFs in radial component. The Preview window shows the RFs arranged by backazimuth. The example data is the station YKW3 of network CN in Canada (Yuan *et al.*, 2006). The color version of this figure is available only in the electronic edition.

anisotropic estimation, harmonic decomposition, and CCP stacking into Seispy with friendly usages in the command line.

H-*κ* stacking

H-*κ* stacking (Zhu and Kanamori, 2000) is used to estimate the Moho depth and crustal V_P/V_S beneath the seismic station. To do *H*-*κ* stacking, users need to set up parameters in a configure file first. Then just execute "*hk*" in the command line. The optimal estimations of Moho depth and V_P/V_S will be written into a text file. The figure of *H*-*κ* stacking will be also saved with a high dots-per-inch (DPI) value (Fig. 5a). Figure 5a is an example showing the *H*-*κ* stacking result for the station NJ2 of network CB in China.

Crustal anisotropic estimation

Seispy involves crustal anisotropic estimation with a joint method (Liu and Niu, 2012) of radial energy maximization with cosine moveout correction, radial energy maximization with cosine moveout correction, and transverse energy minimization. The command *rfani* is available for anisotropic estimation with time window, including *Ps* phases and weights of three methods specified in the command line arguments. Figure 5b shows the results of different methods, which is generated automatically after executing the anisotropic estimation. This example is for the station LTA of network SC in China.

Harmonic decomposition

Harmonic decomposition is an effective technique to separate anisotropic and dipping components from radius and transverse RFs (Bianchi *et al.*, 2010). This technique is not only used to estimate azimuthal crustal anisotropy and crustal layer dipping (Liu *et al.*, 2015; Han *et al.*, 2020), but also to extract the isotropic components of RF for 1D RF inversion (Shen *et al.*, 2013). In the Seispy, execute the command *rfharmo* to calculate the harmonic decomposition, plot the different components after decomposition, and save the isotropic components to a local SAC file. Figure 5c is an example of the harmonic decomposition for the station LTA of network SC in southwest China.

CCP stacking

CCP stacking is a routine method to image topography of discontinuities, such as the Moho, lithosphere asthenosphere boundary (LAB), and mantle transition zone interfaces (i.e., the 410 and 660 km discontinuities). The Seispy provides rich functions to handle CCP stacking for *Ps* or *Sp* conversions in different application scenarios. CCP stacking contains two main steps: the time-to-depth conversion (*rf2depth*) and CCP stacking (*ccp_profile* for 2D CCP stacking along a profile or *ccp3d* for 3D CCP stacking).

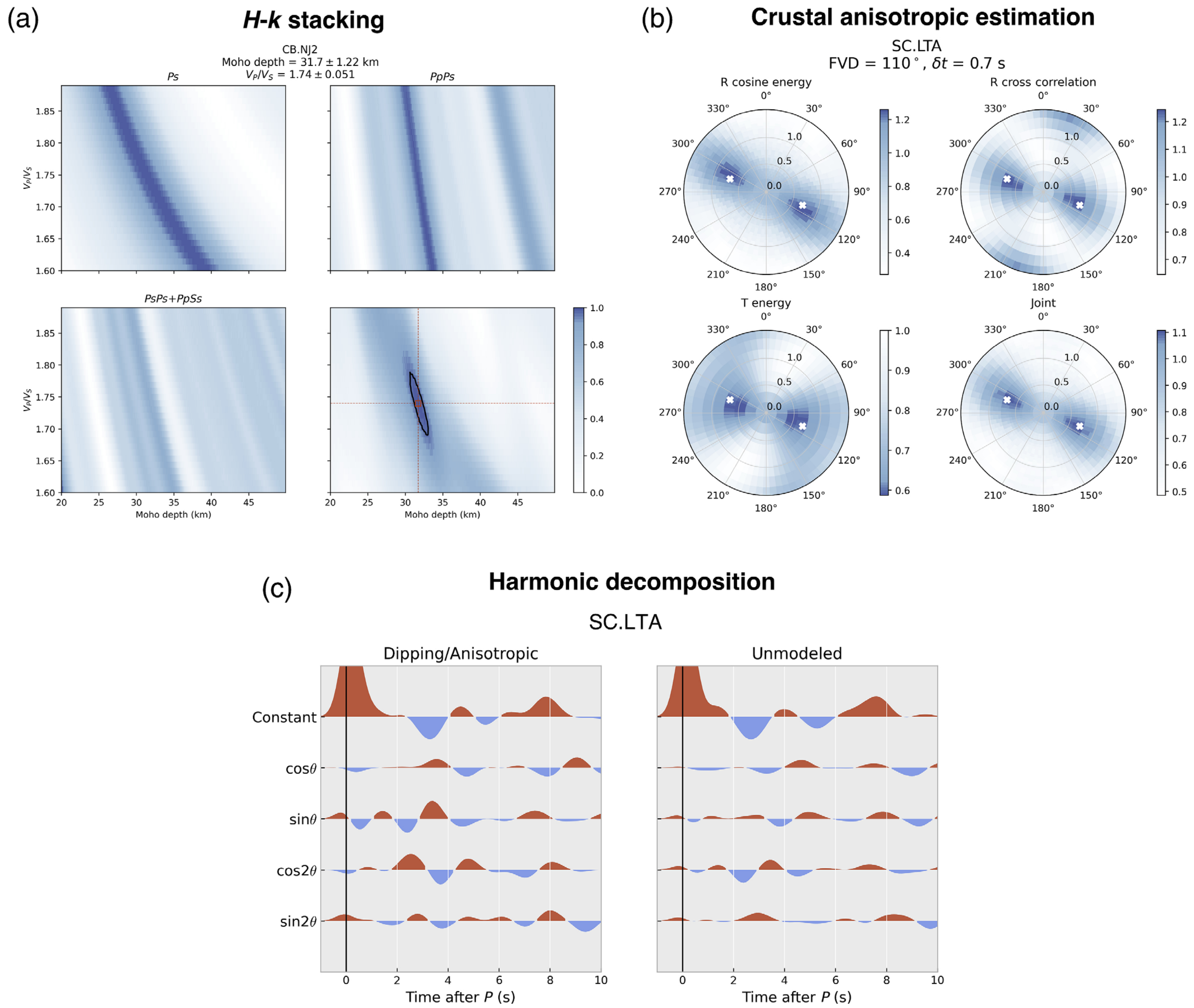

Figure 5. (a) Results of *H*-*κ* stacking for the station NJ2 of network CB in China. (b) Crustal anisotropic estimation. (c) Harmonic decomposition. The data in panels (b) and (c) are from the station LTA of network SC in China. The color version of this figure is available only in the electronic edition.

Time-to-depth conversion

The relationship between the time difference and the depths of converted phases (*Ps* or *Sp*) depends on the velocity structure and ray parameters. Traditionally, the ray parameters of the converted phases are usually approximated as those of the direct phases. However, for deep discontinuities, such as the 410 and 660 km discontinuities, this approximation results in a nonnegligible depth error (Shi *et al.*, 2020). To avoid this error, we provide an option to calculate the ray parameters of the converted phases at different depths through the TauPy submodule in Obspy (Crotwell *et al.*, 1999; Beyreuther *et al.*, 2010).

The velocity structure under the station has a great influence on the time differences between the converted and direct phases, and the locations of the conversion points in different depth layers. Therefore, we use various methods to correct the time differences and the locations of conversion points with a priori 1D or 3D velocity model:

1. Time difference correction with 1D velocity models of different stations. 1D RF inversion (or joint inversion with surface dispersion) can obtain the average velocity model under each station. We call the 1D velocity models at different stations in the fixed format (same as the format in the TauPy Crotwell *et al.*, 1999; Beyreuther *et al.*, 2010) for time difference correction.
2. Time difference correction with 3D velocity model. For the 410 and 660 km discontinuities, a 3D velocity model is a better choice for the time difference correction due to the larger radius of ray distribution. In the 3D time difference

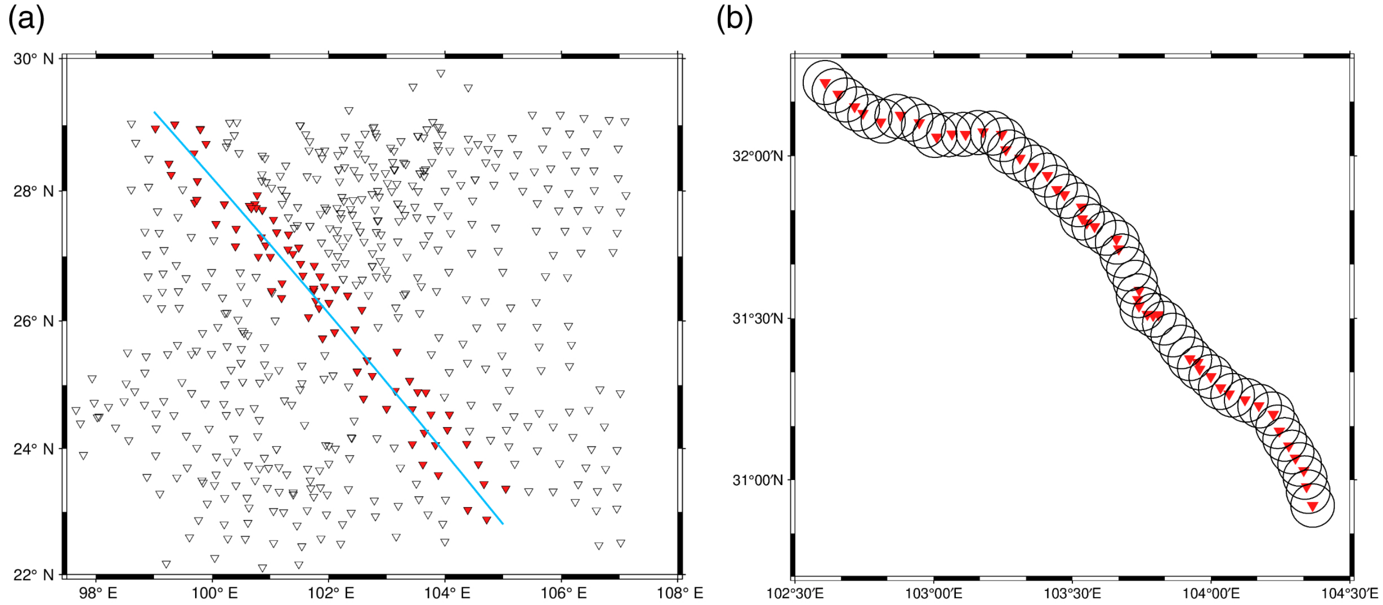

Figure 6. (a) Stations used for CCP stacking when setting two endpoint locations. The triangles are the seismic stations. The blue line shows the location of 2D CCP stacking profile. The red triangles are the stations used in the CCP stacking profile. (b) Adaptive bins for the curved CCP stacking profile. The red triangles are the seismic stations. The circles are the CCP stacking bins. The color version of this figure is available only in the electronic edition.

correction, we first use the 1D velocity model to calculate the locations of conversion points in depth for different RFs and then interpolate the velocity model by these conversion points from the 3D velocity model. The interpolated velocity models are used to calculate the time difference of different RFs.

3. 3D time difference correction migrated by Snell's law. Assumed a layered model under the station, the ray parameters are approximated as those of the direct primary waves. Based on the 3D velocity model, Snell's law is used to calculate the ray incidence angles and locations of conversion points for different RFs in each layer. Then, the corrected time differences of converted phases will be obtained by interpolating the migrated ray paths in the 3D velocity model.

The RFs in the depth domain are obtained by interpolating the RF amplitude with the corrected time difference after removing the station elevation.

2D and 3D CCP stacking

Seispy can handle both 2D and 3D CCP stacking. The 2D CCP stacking is used to image the cross section along a profile. Thus, the location of each bin centroid and the width of the profile are crucial in the 2D CCP stacking. According to different distribution of seismic arrays, we provide the following ways to set the stacking profiles:

1. Set two endpoint locations and the width of the profile (Fig. 6a). This setting is suitable for seismic arrays with spatial distribution (e.g., the USArray or ChinArray). Stations are selected with the distance of stations perpendicular to the profile less than the width of the profile.
2. Set the endpoint locations of the profile, and give the station locations. This setting is suitable for linear arrays that are relatively straight.
3. Adaptive bins along the curved profile (Fig. 6b). The user just needs to set the stations along the profile in order. The bins are automatically set at equal intervals along the curved profile. This technique is applicable to a dense array that cannot be deployed along a straight line due to topographic constraints.

In Seispy, the shape of the bin can be set as a circle or a rectangle. What is more, the radius of the bin can be set to a constant value or the first Fresnel radius that varies with depth. With bins set at equal intervals along the profile, amplitudes of RFs in each bin are stacked at discrete depths.

For spatial distributed arrays, we also provide 3D CCP stacking for extracting the discontinuity depths. The circle bins, whose radius can be set to a constant or the first Fresnel radius, are meshed with uniform central angles and depth intervals. This method has been widely used to extract the topography of the MTZ (i.e., the 410 and 660 km discontinuities). The data and example for imaging the MTZ structure in central Tibet can be found in Xu, Huang, Wang, Xu, Mi, *et al.*, (2020) and the documentation pages. It can also be used to extract the Moho depths in dense seismic arrays. Compared with the H-κ stacking, this method takes into account the effect of the difference of RFs with backazimuth and the effect caused by the ambiguous multiples when a reliable velocity model is applied for time difference correction. In our previous study, a sharp lateral Moho variation in the southeastern Tibet was revealed using this method (Xu, Huang, Wang, Xu, Zhang, *et al.*, 2020).

Conclusions

Seispy is a Python module, GUI-equipped software designed to provide automated processing and a graphical interface for RF calculation in seismology. Its automatic and batch processing for RF estimation is ideal for processing large volume of data with different types. Seispy includes basic RF-based methods of H-κ stacking, crustal anisotropic estimation, harmonic decomposition, and 2D and 3D CCP stacking. Rich modules are provided for processing different applications of CCP stacking. The modular design of Seispy allows its functionality to be easily extended by creating and merging new modules.

Data and Resources

Seispy is freely available at https://github.com/xumi1993/seispy. A complete documentation and examples for Seispy can be accessed at https://seispy.xumijian.me. It also can be easily installed via conda-forge (https://anaconda.org/conda-forge/seispy). The data used here as examples for RF calculation can be downloaded from the Incorporated Research Institution for Seismology (IRIS). The data used for crustal anisotropy estimation are downloaded from the Data Management Centre of China National Seismic Network. The graphical user interfaces are built upon the PyQt5 (https://www.riverbankcomputing.com/). The graphics are plotted using Matplotlib (Hunter, 2007). All websites were last accessed in December 2022.

Declaration of Competing Interests

The authors acknowledge that there are no conflicts of interest recorded.

Acknowledgments

This work was funded by the National Natural Science Foundation of China (Grant Number 42074102) and Research grants from National Institute of Natural Hazards, Ministry of Emergency Management of China (Grant Number ZDJ2020-11). The authors thank the Editor-in-chief, Allison Bent, and two anonymous reviewers for their constructive advice. The authors thank the Python, PyQt5, and ObsPy communities for sharing open-source codes.

References

Ammon, C. (1991). The isolation of receiver effects from teleseismic P waveforms, *Bull. Seismol. Soc. Am.* **81,** no. 6, 2504–2510, doi: 10.1785/BSSA0810062504.

Beyreuther, M., R. Barsch, L. Krischer, T. Megies, Y. Behr, and J. Wassermann (2010). ObsPy: A Python toolbox for seismology, *Seismol. Res. Lett.* **81,** no. 3, 530–533, doi: 10.1785/gssrl.81.3.530.

Bianchi, I., J. Park, N. Piana Agostinetti, and V. Levin (2010). Mapping seismic anisotropy using harmonic decomposition of receiver functions: An application to northern Apennines, Italy, *J. Geophys. Res.* **115,** no. B12, B12,317, doi: 10.1029/2009JB007061.

Bostock, M. G. (1998). Mantle stratigraphy and evolution of the Slave province, *J. Geophys. Res.* **103,** no. B9, 21183–21200, doi: 10.1029/98JB01069.

Crotwell, H. P., T. J. Owens, and J. Ritsema (1999). The TauP Toolkit: Flexible seismic travel-time and ray-path utilities, *Seismol. Res. Lett.* **70,** 154–160, doi: 10.1785/gssrl.70.2.154.

Dreiling, J., and F. Tilmann (2019). BayHunter—McMC transdimensional Bayesian inversion of receiver functions and surface wave dispersion, *GFZ Data Services*, 5 pp., doi: 10.5880/GFZ.2.4.2019.001.

Dueker, K. G., and A. F. Sheehan (1997). Mantle discontinuity structure from midpoint stacks of converted P to S waves across the Yellowstone hotspot track, *J. Geophys. Res.* **102,** no. B4, 8313–8327, doi: 10.1029/96JB03857.

Eagar, K. C., and M. J. Fouch (2012). FuncLab: A MATLAB interactive toolbox for handling receiver function datasets, *Seismol. Res. Lett.* **83,** no. 3, 596–603, doi: 10.1785/gssrl.83.3.596.

Eulenfeld, T. (2020). rf: Receiver function calculation in seismology, *J. Open Source Software* **5,** no. 48, 1808, doi: 10.21105/joss.01808.

Farra, V., and L. Vinnik (2000). Upper mantle stratification by P and S receiver functions, *Geophys. J. Int.* **141,** no. 3, 699–712, doi: 10.1046/j.1365-246x.2000.00118.x.

Gurrola, H., G. E. Baker, and J. B. Minster (1995). Simultaneous time-domain deconvolution with application to the computation of receiver functions, *Geophys. J. Int.* **120,** no. 3, 537–543, doi: 10.1111/j.1365-246X.1995.tb01837.x.

Han, C., M. Xu, Z. Huang, L. Wang, M. Xu, N. Mi, D. Yu, T. Gou, H. Wang, S. Hao, *et al.* (2020). Layered crustal anisotropy and deformation in the SE Tibetan plateau revealed by Markov-Chain-Monte-Carlo inversion of receiver functions, *Phys. Earth Planet. In.* 106522, doi: 10.1016/j.pepi.2020.106522.

Helffrich, G. (2006). Extended-time multitaper frequency domain cross-correlation receiver-function estimation, *Bull. Seismol. Soc. Am.* **96,** no. 1, 344–347, doi: 10.1785/0120050098.

Hunter, J. D. (2007). Matplotlib: A 2D graphics environment, *Comput. Sci. Eng.* **9,** no. 3, 90–95, doi: 10.1109/MCSE.2007.55.

Jiang, C., and M. A. Denolle (2020). NoisePy: A new high-performance Python tool for ambient-noise seismology, *Seismol. Res. Lett.* **91,** no. 3, 1853–1866, doi: 10.1785/0220190364.

Langston, C. A. (1979). Structure under Mount Rainier, Washington, inferred from teleseismic body waves, *J. Geophys. Res.* **84,** no. B9, 4749–4762, doi: 10.1029/JB084iB09p04749.

Li, Z., J. Zhou, G. Wu, J. Wang, G. Zhang, S. Dong, L. Pan, Z. Yang, L. Gao, Q. Ma, *et al.* (2021). CC-FJpy: A Python package for extracting overtone surface-wave dispersion from seismic ambient-noise cross correlation, *Seismol. Res. Lett.* **92,** no. 5, 3179–3186, doi: 10.1785/0220210042.

Ligorría, J. P., and C. J. Ammon (1999). Iterative deconvolution and receiver-function estimation, *Bull. Seismol. Soc. Am.* **89,** no. 5, 1395–1400, doi: 10.1785/BSSA0890051395.

Liu, H., and F. Niu (2012). Estimating crustal seismic anisotropy with a joint analysis of radial and transverse receiver function data, *Geophys. J. Int.* **188,** no. 1, 144–164, doi: 10.1111/j.1365-246X.2011.05249.x.

Liu, Z., J. Park, and D. M. Rye (2015). Crustal anisotropy in northeastern Tibetan plateau inferred from receiver functions: Rock textures caused by metamorphic fluids and lower crust flow? *Tectonophysics* **661,** 66–80, doi: 10.1016/j.tecto.2015.08.006.

Park, J., and V. Levin (2000). Receiver functions from multiple-taper spectral correlation estimates, *Bull. Seismol. Soc. Am.* **90,** no. 6, 1507–1520, doi: 10.1785/0119990122.

Shen, W., M. H. Ritzwoller, V. Schulte-Pelkum, and F.-C. Lin (2013). Joint inversion of surface wave dispersion and receiver functions: A Bayesian Monte-Carlo approach, *Geophys. J. Int.* **192,** no. 2, 807–836, doi: 10.1093/gji/ggs050.

Shi, J., T. Wang, and L. Chen (2020). Receiver function velocity analysis technique and its application to remove multiples, *J. Geophys. Res.* **125,** no. 8, doi: 10.1029/2020JB019420.

Wu, Q., X. Tian, N. Zhang, W. Li, and R. Zeng (2003). Receiver function estimated by maximum entropy deconvolution, *Acta Seismol. Sin.* **16,** no. 4, 404–412, doi: 10.1007/s11589-003-0073-y.

Xu, M., H. Huang, Z. Huang, and L. Wang (2016). SplitRFLab: A MATLAB GUI toolbox for receiver function analysis based on SplitLab, *Earthq. Sci.* doi: 10.1007/s11589-016-0141-8.

Xu, M., Z. Huang, L. Wang, M. Xu, N. Mi, and D. Yu (2020). Lateral variation of the mantle transition zone beneath the Tibetan plateau: Insight into thermal processes during Indian–Asian collision, *Phys. Earth Planet. In.* **301,** 106,452, doi: 10.1016/j.pepi.2020.106452.

Xu, M., Z. Huang, L. Wang, M. Xu, Y. Zhang, N. Mi, D. Yu, and X. Yuan (2020). Sharp lateral Moho variations across the SE Tibetan margin and their implications for plateau growth, *J. Geophys. Res.* **125,** no. 5, doi: 10.1029/2019JB018117.

Yu, C., Y. Zheng, and X. Shang (2017). Crazyseismic: A MATLAB GUI-based software package for passive seismic data preprocessing, *Seismol. Res. Lett.* **88,** no. 2A, 410–415, doi: 10.1785/0220160207.

Yuan, X., R. Kind, X. Li, and R. Wang (2006). The S receiver functions: Synthetics and data example, *Geophys. J. Int.* **165,** no. 2, 555–564, doi: 10.1111/j.1365-246X.2006.02885.x.

Zhu, L. (2000). Crustal structure across the San Andreas fault, southern California from teleseismic converted waves, *Earth Planet. Sci. Lett.* **179,** no. 1, 183–190, doi: 10.1016/S0012-821X(00)00101-1.

Zhu, L., and H. Kanamori (2000). Moho depth variation in southern California from teleseismic receiver functions, *J. Geophys. Res.* **105,** no. B2, 2969, doi: 10.1029/1999jb900322.

Manuscript received 10 September 2022
Published online 13 December 2022