FrosPy: A Modular Python Toolbox for Normal Mode Seismology

Simon Schneider*[1], Sujania Talavera-Soza[1], Lisanne Jagt[1], and Arwen Deuss[1]

Abstract

We present free oscillations Python (FrosPy), a modular Python toolbox for normal mode seismology, incorporating several Python core classes that can easily be used and be included in larger Python programs. FrosPy is freely available and open source online. It provides tools to facilitate pre- and postprocessing of seismic normal mode spectra, including editing large time series and plotting spectra in the frequency domain. It also contains a comprehensive database of center frequencies and quality factor (*Q*) values based on 1D reference model preliminary reference Earth model for all normal modes up to 10 mHz and a collection of published measurements of center frequencies, *Q* values, and splitting function (or structure) coefficients. FrosPy provides the tools to visualize and convert different formats of splitting function coefficients and plot these as maps. By giving the means of using and comparing normal mode spectra and splitting function measurements, FrosPy also aims to encourage seismologists and geophysicists to learn about normal mode seismology and the study of the Earth's free oscillation spectra and to incorporate them into their own research or use them for educational purposes.

Cite this article as Schneider, S., S. Talavera-Soza, L. Jagt, and A. Deuss (2022). FrosPy: A Modular Python Toolbox for Normal Mode Seismology, *Seismol. Res. Lett.* **93**, 967–974, doi: 10.1785/0220210208.

Supplemental Material

Introduction

Studying the Earth's free oscillations, or normal modes, requires processing raw time-series data recorded by seismometers and converting these to spectra in the frequency domain. Currently, there is a limited number of programs and scripts available for the pre- and postprocessing of seismic normal mode spectra. These programs are often not public and are usually just shared within research groups, making it necessary to write the same routines repeatedly.

Here, we present free oscillations Python (FrosPy), an easy-to-use Python toolbox for handling normal mode spectra and splitting function coefficients. FrosPy is built using the Python library ObsPy (Beyreuther *et al.*, 2010), which handles all input and output and which nowadays is widely used among seismologists. The core functionality of FrosPy enables the user to build complex applications on top of FrosPy. Our new FrosPy routines are readily combined and implemented in larger Python programs, making use of powerful libraries such as NumPy (Harris *et al.*, 2020) and SciPy (Virtanen *et al.*, 2020) or visualization libraries such as Matplotlib (Hunter, 2007) and Cartopy (Met Office, 2010–2015). Because Python and all other libraries used here are open source, there is a significant advantage over proprietary software, such as the widely used MATLAB (2010), because there are no restrictions due to licensing. This makes FrosPy an ideal tool for students and teachers who are interested in learning about making normal mode spectra observations as well as experienced researchers who would like to have a central and expandable modular library for normal mode research.

FrosPy provides all tools for the typical workflow of a seismologist who studies normal modes. First, we need to obtain time-series records of strong earthquakes ($M_w > 7$) that last several days or sometimes even weeks or months to observe normal modes. In Python, these long time-series records can be obtained using, for example, ObsPy's International Federation of Digital Seismograph Networks webservice. Next, because normal modes are observed at the lowest frequency range of the frequency spectrum (usually $f < 10$ mHz), the data need to be processed and cleaned. Therefore delta-like spikes in the time domain, which cause the spectra in the frequency domain to become "noisy," need to be removed. FrosPy provides functions in its preprocessing module to edit the data (e.g., cutting all records to the same start and end timestamps) and apply corrections (e.g., remove_delta_pulses, geocentric, and rotate). Then the time series needs to be tapered and converted to the frequency domain to plot the actual frequency spectra. This is done by Spectrum, one of the core classes of FrosPy that also takes care of the plotting of the spectra. After this, the modes are readily

1. Department Earth Sciences, Utrecht University, Utrecht, The Netherlands, https://orcid.org/0000-0003-0947-917X (ST-S)

*Corresponding author: s.a.schneider@uu.nl

© Seismological Society of America

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user

visible in the frequency domain and can be identified by their center frequencies. The center frequencies of each mode are in turn available through the Modes class and can be plotted on the frequency spectra. The resulting frequency spectra are now sufficient for educational purposes and making seismic observations.

In normal mode seismology research, it is common to use the normal mode spectra to measure splitting functions, which are depth-weighted averages of how a mode "sees" the Earth that provide local frequency perturbations of a specific mode (e.g., Woodhouse and Giardini, 1985; Deuss *et al.*, 2011; Schneider and Deuss, 2021). To make such measurements, we need to identify (1) frequency windows around specific mode peaks and (2) time windows to set the length of the tapered time series for which the spectrum is calculated. With Pick, FrosPy provides a class that stores the information related to the frequency and time windows. To plot splitting function measurements and compare them with already published measurements, the SplittingFunc class provides functionality to plot and compare splitting function coefficients, both as individual coefficients or as a splitting function map. All together, these classes provide all the basic functionality needed to use seismic time-series records for the purpose of education or for processing and plotting frequency spectra and making normal mode observations, such as splitting function measurements. Still, measuring splitting function coefficients is not straightforward because it is a nonlinear inverse problem, which requires calculating synthetic normal mode spectra, and calculating partial derivatives, which requires a different suite of software and can be computationally expensive (e.g., Resovsky and Ritzwoller, 1998; Masters *et al.*, 2000; Pachhai *et al.*, 2015; Schneider and Deuss, 2021). For this reason, it is out of the scope of this article, and it will be the goal of future research and development to tackle the implementation within the scope of FrosPy.

Here, we demonstrate the usage of FrosPy's core classes and how they tie into existing Python programs. To this end, we group these classes by their type of functionality in the typical workflow of a seismologist. We also give three examples of the usability of the FrosPy package: (1) the interactive app spectrum, included in the package, which allows the user to process and manipulate real and synthetic seismograms into frequency spectra; (2) the plotting of spectra and splitting function coefficients as maps; and (3) the plotting of splitting function coefficients along a normal mode branch.

Functionality

FrosPy contains the following classes and tools, which are essential for normal mode seismology research:

1. Preprocessing: applying seismic data corrections such as removal of delta pulses and preparation of the time-series data
2. Spectrum: applying the Fourier transformation to convert the time-series data to the frequency domain and plotting the frequency spectra
3. Modes: a comprehensive database consisting of center frequency and quality factor (Q) values based on preliminary reference Earth model (PREM; Dziewonski and Anderson, 1981) for all modes up to 10 mHz
4. Pick and Segment: reading and writing of frequency and time windows required for splitting function measurement inversion of specific modes
5. SplittingFunc and Set: reading, writing, and plotting of splitting function coefficients c_{st} (e.g., Deuss *et al.*, 2013) in various previously used formats, including sqlite3 (Hipp, 2020)
6. Load: a database of c_{st} coefficients, center , and Q values of various published measurements (e.g., Masters *et al.*, 2000; Deuss *et al.*, 2013; Koelemeijer *et al.*, 2013; Laske *et al.*, 2013; Talavera-Soza and Deuss, 2020, 2021; Schneider and Deuss, 2021)

Preprocessing

FrosPy provides functions to edit the data (e.g., cutting all records to the same start and end times) and apply corrections (e.g., to remove delta pulses and any other points that are bigger than a given threshold compared with the nearby points in the time series and interpolate the missing values using remove_delta_pulses). Further functions in its preprocessing module include geocentric, which applies geocentric coordinate corrections, and rotate, which rotates the time-series components from north and east to radial and transversal (or vice versa). To analyze the spectral data quality, additional functions are provided, such as signal2noise, which calculates the signal-to-noise ratio in a frequency spectrum, and signal2fwhm, which calculates the signal to full-width-half-maximum ratio.

Plotting frequency spectra: spectrum

The Spectrum class provides all basic functionality for users who want to convert a seismic time-series record to the frequency domain and plot its spectrum to study individual normal mode peaks (Fig. 1). The time series is first tapered using typically a Hann or cosine taper. For N samples at sampling times $t = n\Delta t$ with $n = 0, 1, \ldots, N-1$, the Hann taper weights are given by

$$\omega_n = \omega(n\Delta t) = \frac{1}{2}\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right). \quad (1)$$

Other taper functions are available, too. (Additional tapers include Bartlett with $\omega_n = \frac{2}{N-1}(\frac{N-1}{2} - |n - \frac{N-1}{2}|)$, Hamming with $\omega_n = 0.54 - 0.46\cos(\frac{2\pi n}{N-1})$, Blackman with $\omega_n = 0.42 - 0.5\cos(\frac{2\pi n}{N-1}) + 0.08\cos(\frac{4\pi n}{N-1})$, Kaiser with $\omega_n = I_0(\beta\sqrt{1 - \frac{4n^2}{(N-1)^2}})/I_0(\beta)$, and I_0 as the modified zeroth-order Bessel function and also Boxcar.) Before performing the Fourier transform, the tapered series is padded with zeros

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user

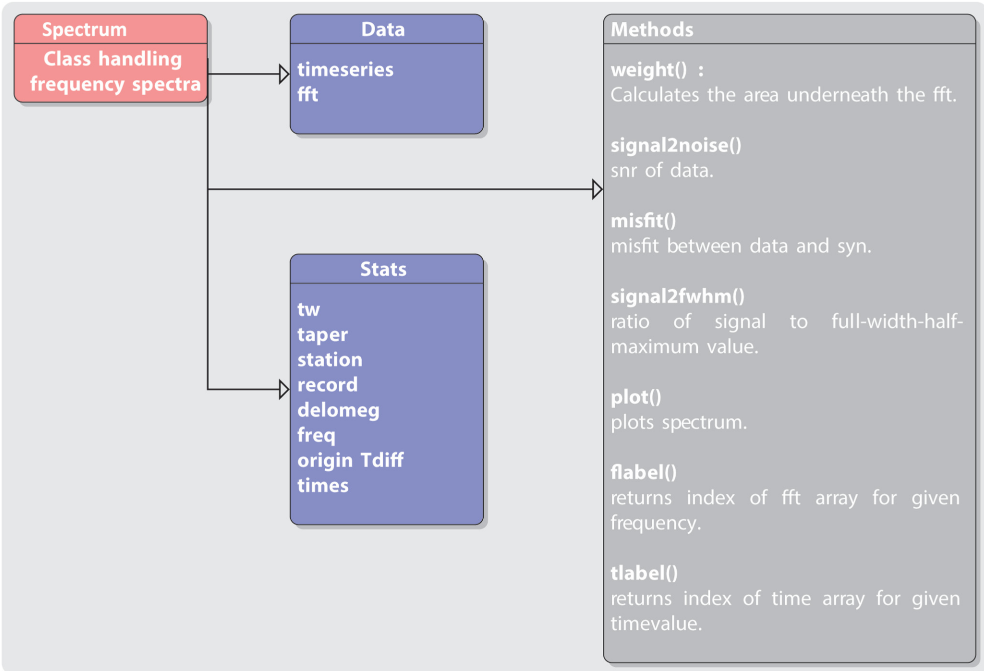


Figure 1. Spectrum class containing Data and Stats subclasses and methods.

the time domain trace to the frequency domain. The Spectrum object contains the frequency spectrum and all additional trace information (e.g., corresponding time series, station name and event location, date, earthquake parameters). The calculation of the time differences between data record and source origin time is done automatically based on the timestamp of the trace. The Spectrum object is created for a specific time window of the time series, enabling us to perform the fast Fourier transform once and browse through the available frequency spectrum without recomputing it multiple times. All auxiliary information, such as the chosen time window, the kind of taper, station code, and so forth, is stored as well.

to interpolate the spectrum between the Fourier frequencies $\omega_n = \frac{2\pi n}{N\Delta t}$. Then the data can be plotted as a frequency spectrum to investigate the normal mode amplitude and phase spectrum of the seismic record (see Fig. 2 for a simple example of a frequency spectrum plot) using our basic plotting routine for a user-defined frequency range.

The Spectrum class takes a Trace object from ObsPy as input and creates a Spectrum object by Fourier transforming

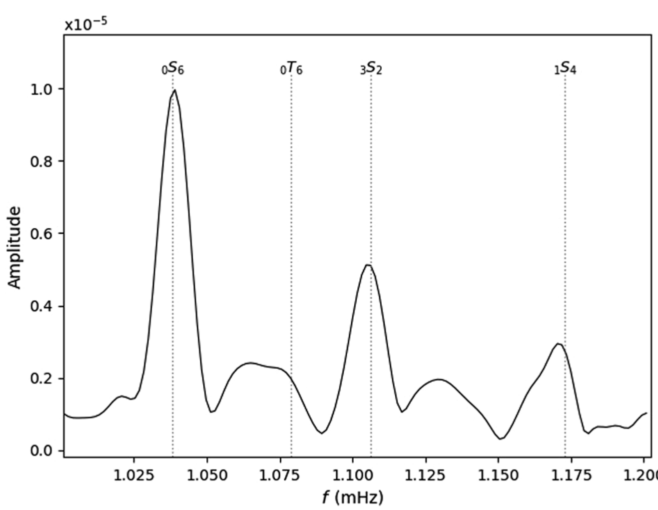


Figure 2. Example of amplitude spectrum plot made using the Spectrum class. Also shown are mode names and corresponding center frequencies (dashed lines) for the 1D model preliminary reference Earth model (PREM) in the given frequency window.

Catalog of center frequencies and *Q* values: modes

In addition, we provide a database of normal mode data information, such as center frequencies, *Q* values, and other user-defined information, that can be added, such as sensitivity to shear- or compressional-wave velocity, or to a certain region like the inner core. The Modes class is the container for Mode objects and makes it easy to gather multiple modes for further processing (Fig. 3). Each object of the Modes class contains all basic information about the specific spheroidal mode nS_l or toroidal mode nT_l, such as overtone number n, angular order l, Q value, center frequency, and special sensitivity (e.g., v_p, Stoneley modes, and inner core, and so on) The database can be loaded using FrosPy's read function. Our database

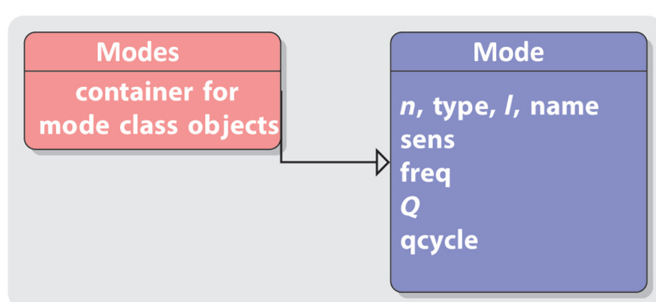


Figure 3. Modes container and mode class.

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user

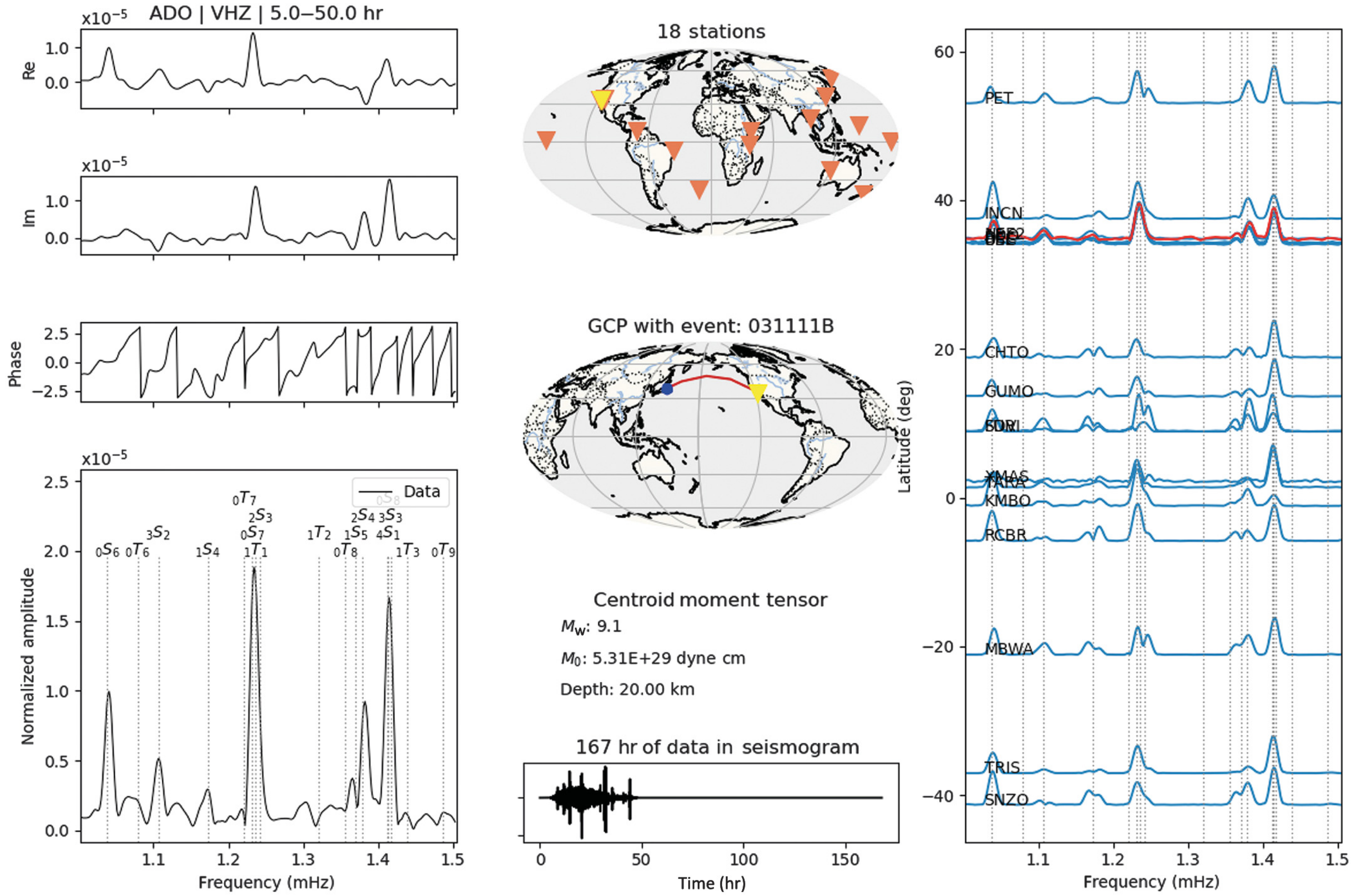


Figure 4. Example plot of the application spectrum (lower case "s") showing all information stored in the Spectrum and Set objects, including the amplitude and phase of the frequency spectrum, the station and event location, the earthquake centroid moment tensor, and the tapered time series. In addition, a plot of all amplitude spectra, loaded for this event, is shown, and sorted by latitude and with all the modes present within the frequency window indicated using vertical dashed lines at their PREM center frequency values.

includes the aforementioned information for all normal modes up to 10 mHz based on 1D reference model PREM (Dziewonski and Anderson, 1981), which can be expanded or created from scratch by the user for other reference models.

The interactive app spectrum (note the lower case "s") can be used to plot and inspect normal mode frequency spectra and additional information. It accepts seismic data in any file format that is accepted by ObsPy (e.g., Ad Hoc [AH], Seismic Analysis Code [SAC], Group of Scientific Experts 2 [GSE2], or miniSEED) and event information (e.g., from a QuakeML file). We used the Spectrum class to convert the given time series into Spectrum objects and used it to produce Figure 4 as an example of the processing and visualization capabilities of FrosPy. The app uses the input to plot information of the frequency domain, such as the real and imaginary parts; the phase and the amplitude spectrum; and station locations, centroid moment tensor, and the time series. The user can specify settings, such as frequency and time windows, for the spectra and can browse through all stations found in the input data while the plot is updated automatically.

Splitting function inversion preparation: Pick and Segment

In addition, we present a simple tool to help prepare auxiliary input data for splitting function measurements, which require frequency and time windows for the targeted modes as a basic input. To store each individual time and frequency window, we use the Pick class; multiple Pick objects are gathered in a Segment (Fig. 5).

Each Pick is created using a seismic record as a reference and used to automatically calculate a normalization factor for the given frequency window. This factor is calculated as the area underneath the amplitude spectrum that is the integral. Each amplitude spectrum is then normalized using this factor and weighed independently of the source. In combination with the Spectrum class, this automates the picking process of frequency windows, which is useful to prepare the inversion of normal mode spectra to measure splitting functions.

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user

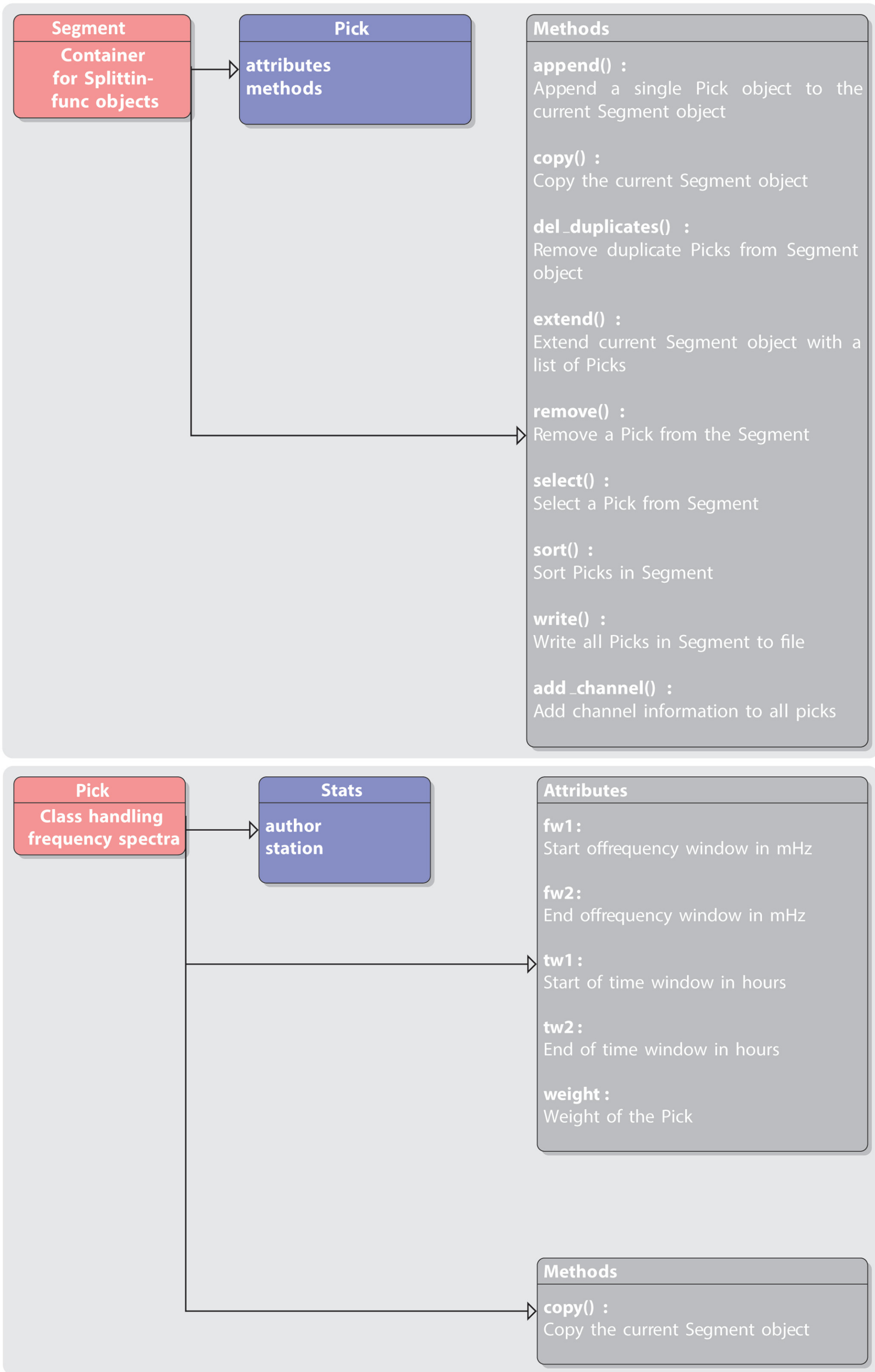


Figure 5. Segment container with Pick class and methods.

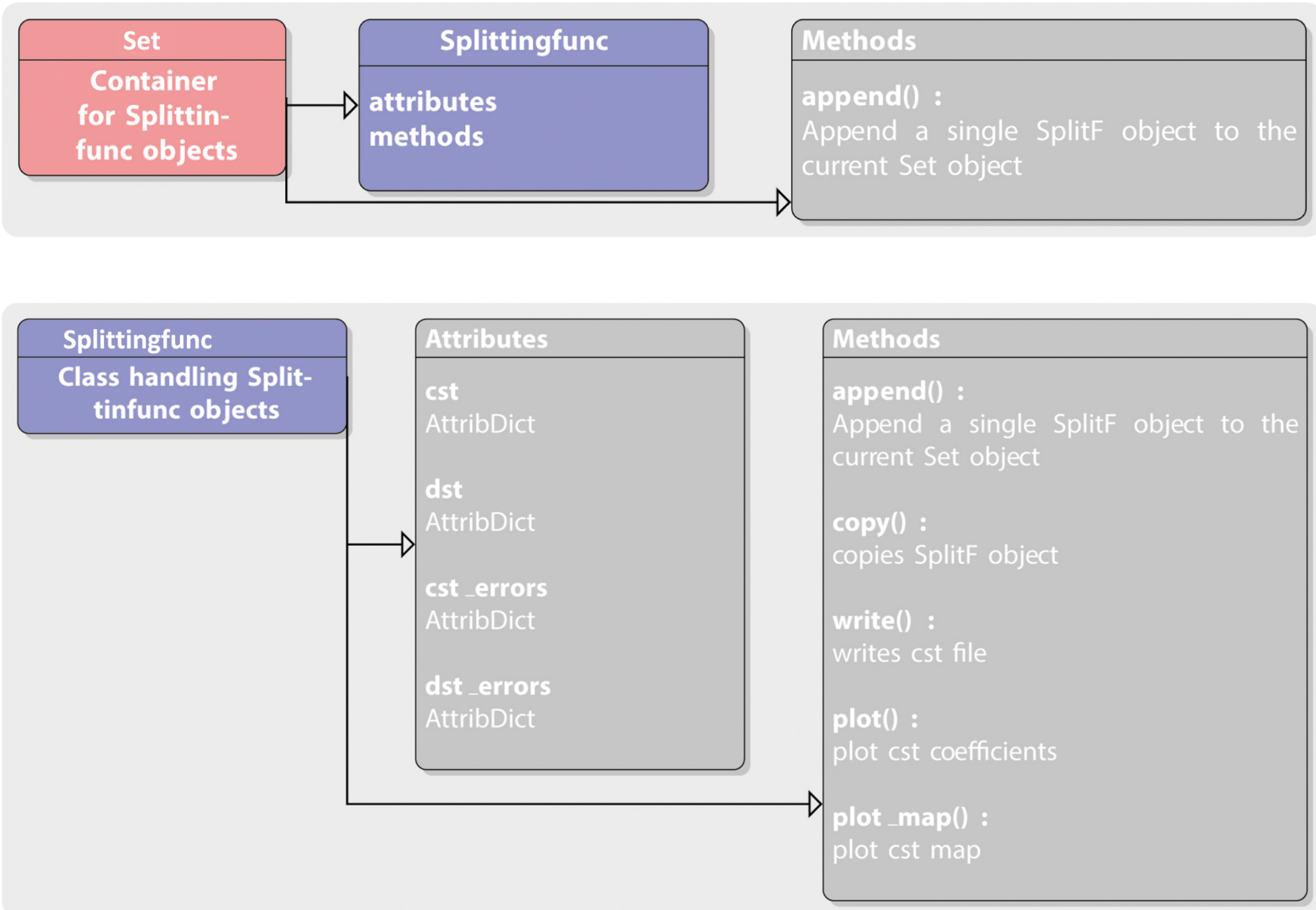
Splitting function plotting: SplittingFunc and Set

Splitting function coefficients are handled by the SplittingFunc class. We define splitting function coefficients c_{st} as complex spherical harmonic coefficients of angular order s and azimuthal order t. The Set class is a collection of multiple SplittingFunc objects (Fig. 6). For each mode and each spherical harmonic degree s, SplittingFunc contains an array with the corresponding coefficients, providing straightforward access to the individual coefficients and to the plotting routines. We provide a load and a write module from user-defined input files (including all commonly used c_{st} file formats) or from our database provided with FrosPy. In addition, we provide a plotting function to plot c_{st} measurements for a specific mode branch that is plotting mode center frequencies or splitting functions coefficients with constant n as a function of increasing angular order l (see Fig. 7) using Set as an input. The function automatically sorts the splitting function coefficients in Set, here by angular order l of the toroidal modes $_nT_l$, and calculates the center frequency based on the coefficients. In addition, we present an example of plotting splitting function maps, whose corresponding coefficients can directly be obtained from the catalog provided with FrosPy. The load function creates one or more SplittingFunc objects, which can then be used to visualize the splitting function as a map (see Fig. 8). For further information on splitting function maps, see Deuss *et al.* (2013).

Databases

In addition, we provide a database of several published datasets of splitting function coefficient measurements, center frequencies, and Q values from different groups, either partially or in its entirety (Roult *et al.*, 1990; Widmer-Schnidrig

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user



et al., 1991; Widmer *et al.*, 1992; Durek and Ekström, 1995; Masters and Widmer, 1995; Tromp and Zanzerkia, 1995; He and Tromp, 1996; Resovsky and Ritzwoller, 1998; Ritsema *et al.*, 1999; Beghein *et al.*, 2008; Ritsema *et al.*, 2011; Deuss *et al.*, 2011, 2013; Koelemeijer *et al.*, 2013; Talavera-Soza and Deuss, 2020, 2021; Schneider and Deuss, 2021). To our knowledge, this is the first comprehensive recent collection of all normal mode measurements.

Conclusion

We developed an accessible and versatile Python library for seismologists focusing on processing and plotting normal mode spectra. It is a platform-independent package with a modular structure based on ObsPy and other well-known open-source third-party Python tools and is open for further development. In addition, we provide a normal mode database of center frequency and Q values up to 10 mHz based on 1D reference model PREM and a database of published measurements of center frequencies, Q values, and splitting function coefficients from the past 30 yr with tools to convert and plot these in various ways. We believe that this provides a useful toolbox for many seismologists. The code is accompanied by documentation and a Wiki and is freely available online.

Figure 6. Set container with SplittingFunc class and methods.

Data and Resources

The facilities of Incorporated Institutions for Seismology (IRIS) Data Services, specifically the IRIS Data Management Center, were used for access to waveforms and related metadata used in this study. IRIS Data Services are funded through the Seismological Facilities for the Advancement of Geoscience and EarthScope (SAGE) Proposal of the National Science Foundation under Cooperative Agreement

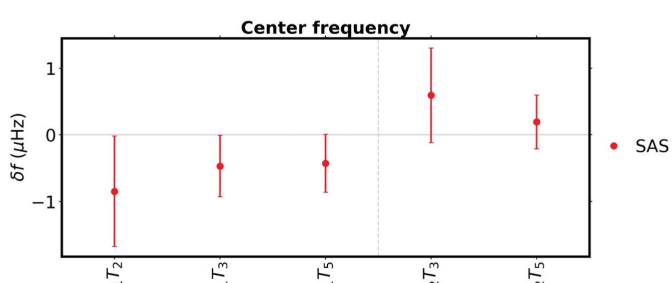


Figure 7. Example branch plot for toroidal mode overtones center frequency values measured by Schneider and Deuss (2021). The center frequency and uncertainty values are loaded into set using the load function and then plotted using the branch function.

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user

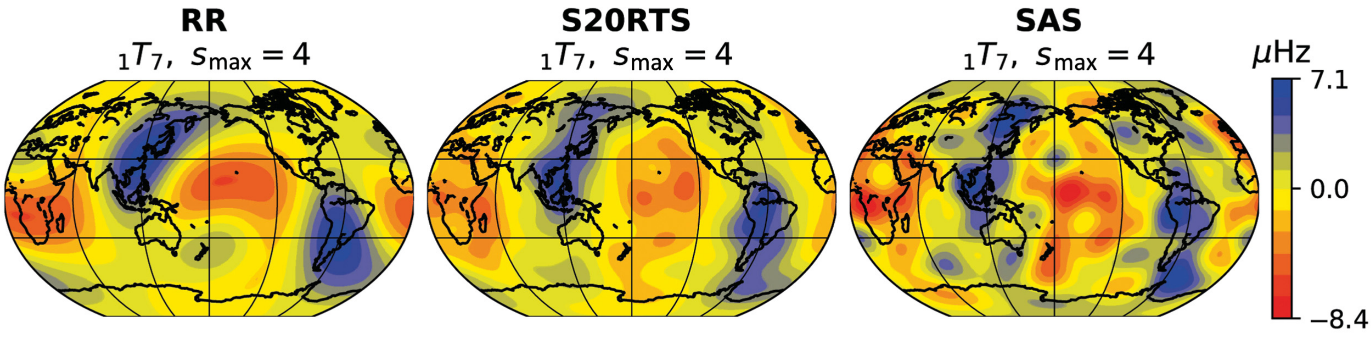


Figure 8. Example plot of splitting function maps, using measured coefficients of Resovsky and Ritzwoller (1998, referred as RR), predictions for 3D tomographic shear-wave velocity model S20RTS (Ritsema *et al.*, 2011) and measured coefficients of Schneider and Deuss (2021, referred as SAS), which have all been loaded from our database.

EAR-1261681. In addition, the authors acknowledge the use of the "Global Centroid Moment Tensor (CMT) project" webpage for the earthquake source parameters used in this study (Dziewonski *et al.*, 1981; Ekström *et al.*, 2012). The data analysis and figures were generated using Python (Van Rossum and Drake, 2009), the Python library ObsPy (Beyreuther *et al.*, 2010), and the Python package free oscillations Python (FrosPy; https://www.frospy.org). The supplemental material for this article includes the script to make the simple frequency spectrum plot (Fig. 2), as well as the code for the interactive app spectrum and corresponding script to make the plot of the spectrum and data (Fig. 4), the script to make the branch plot (Fig. 7), and the script to create splitting function maps (Fig. 8), which are all available online at https://github.com/s-schneider/frospy. All websites were last accessed in December 2021.

Declaration of Competing Interests

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

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (Grant Agreement Number 681535—Attenuation Tomography Using Novel observations of Earth's free oscillations [ATUNE]) and a Vici Award Number 016.160.310/526 from the Nederlandse Organisatie voor Wetenschappelijk Onderzoek (NWO).

References

Beghein, C., J. Resovsky, and R. D. Van Der Hilst (2008). The signal of mantle anisotropy in the coupling of normal modes, *Geophys. J. Int.* **175,** no. 3, 1209–1234.

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.

Deuss, A., J. Ritsema, and H. van Heijst (2011). Splitting function measurements for Earth's longest period normal modes using recent large earthquakes, *Geophys. Res. Lett.* **38,** no. 4, doi: 10.1029/2010GL046115.

Deuss, A., J. Ritsema, and H. van Heijst (2013). A new catalogue of normal-mode splitting function measurements up to 10 mhz, *Geophys. J. Int.* **193,** no. 2, 920–937.

Durek, J. J., and G. Ekström (1995). Evidence of bulk attenuation in the asthenosphere from recordings of the Bolivia earthquake, *Geophys. Res. Lett.* **22,** no. 16, 2309–2312.

Dziewonski, A. M., and D. L. Anderson (1981). Preliminary reference Earth model, *Phys. Earth Planet. Int.* **25,** no. 4, 297–356.

Dziewonski, A., T.-A. Chou, and J. H. Woodhouse (1981). Determination of earthquake source parameters from waveform data for studies of global and regional seismicity, *J. Geophys. Res.* **86,** no. B4, 2825–2852.

Ekström, G., M. Nettles, and A. Dziewoński (2012). The global CMT project 2004–2010: Centroid-moment tensors for 13,017 earthquakes, *Phys. Earth Planet. Int.* **200,** 1–9.

Harris, C. R., K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, *et al.* (2020). Array programming with NumPy, *Nature* **585,** no. 7825, 357–362, doi: 10.1038/s41586-020-2649-2.

He, X., and J. Tromp (1996). Normal-mode constraints on the structure of the earth, *J. Geophys. Res.* **101,** no. B9, 20,053–20,082.

Hipp, R. D. (2020). SQLite, https://www.sqlite.org/index.html (last accessed December 2021).

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

Koelemeijer, P., A. Deuss, and J. Ritsema (2013). Observations of core-mantle boundary Stoneley modes, *Geophys. Res. Lett.* **40,** no. 11, 2557–2561.

Laske, G., A. Dziewonski, and G. Masters (2013). The reference Earth model website, https://igppweb.ucsd.edu/~gabi/rem.html (last accessed December 2021).

Masters, T., and R. Widmer (1995). Free oscillations: Frequencies and attenuations, in *Global Earth Physics: A Handbook of Physical Constants*, Vol. 1, American Geophysical Union, Washington, D.C., 104 pp.

Masters, G., G. Laske, and F. Gilbert (2000). Autoregressive estimation of the splitting matrix of free-oscillation multiplets, *Geophys. J. Int.* **141,** no. 1, 25–42.

MATLAB (2010). Version 7.10.0 (v.7.10.0) (R2010a), The MathWorks Inc., Natick, Massachusetts, available at https://www.mathworks.com/products/matlab (last accessed December 2021).

Met Office (2010–2015). *Cartopy: A Cartographic Python Library with a Matplotlib Interface*, Exeter, Devon, available at http://scitools.org.uk/cartopy (last accessed December 2021).

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user

Pachhai, S., H. Tkalčić, and G. Masters (2015). Estimation of splitting functions from Earth's normal mode spectra using the neighbourhood algorithm, *Geophys. J. Int.* **204,** no. 1, 111–126.

Resovsky, J. S., and M. H. Ritzwoller (1998). New and refined constraints on three-dimensional earth structure from normal modes below 3 mhz, *J. Geophys. Res.* **103,** no. B1, 783–810.

Ritsema, J., A. A. Deuss, H. van Heijst, and J. Woodhouse (2011). S40RTS: A degree-40 shear-velocity model for the mantle from new Rayleigh wave dispersion, teleseismic traveltime and normal-mode splitting function measurements, *Geophys. J. Int.* 184**,** no. 3, 1223–1236.

Ritsema, J., H. J. van Heijst, and J. H. Woodhouse (1999). Complex shear wave velocity structure imaged beneath Africa and Iceland, *Science*, **286,** no. 5446, 1925–1928.

Roult, G., B. Romanowicz, and J. Montagner (1990). 3-D upper mantle shear velocity and attenuation from fundamental mode free oscillation data, *Geophys. J. Int.* **101,** no. 1, 61–80.

Schneider, S., and A. Deuss (2021). A new catalogue of toroidal-mode overtone splitting function measurements, *Geophys. J. Int.* **225,** no. 1, 329–341.

Talavera-Soza, S., and A. Deuss (2020). Constraining 1-D inner core attenuation through measurements of strongly coupled normal mode pairs, *Geophys. J. Int.* **223,** no. 1, 612–621.

Talavera-Soza, S., and A. Deuss (2021). New measurements of long period radial modes using large earthquakes, *Geophys. J. Int.* **224,** no. 2, 1211–1224.

Tromp, J., and E. Zanzerkia (1995). Toroidal splitting observations from the great 1994 Bolivia and Kuril islands earthquakes, *Geophys. Res. Lett.* **22,** no. 16, 2297–2300.

Van Rossum, G., and F. L. Drake (2009). *Python 3 Reference Manual*, CreateSpace, Scotts Valley, California, ISBN 1441412697.

Virtanen, P., R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, *et al.* (2020). SciPy 1.0: Fundamental algorithms for scientific computing in python, *Nat. Methods* **17,** 261–272, doi: 10.1038/s41592-019-0686-2.

Widmer, R., W. Zürn, and G. Masters (1992). Observation of low-order toroidal modes from the 1989 Macquarie Rise event, *Geophys. J. Int.* **111,** no. 2, 226–236.

Widmer-Schnidrig, R., G. Masters, and F. Gilbert (1991). Spherically symmetric attenuation within the Earth from normal mode data, *Geophys. J. Int.* **104,** no. 3, 541–553.

Woodhouse, J., and D. Giardini (1985). Inversion for the splitting functions of isolated normal mode multiplets, *EoS* **66,** 301–301.

Manuscript received 3 August 2021
Published online 5 January 2022

Downloaded from http://pubs.geoscienceworld.org/ssa/srl/article-pdf/93/2A/967/5552868/srl-2021208.1.pdf
by Utrecht University Library user