

rf: Receiver function calculation in seismology

Tom Eulenfeld¹

¹ Institute of Geosciences, Friedrich Schiller University Jena, Germany

DOI: [10.21105/joss.01808](https://doi.org/10.21105/joss.01808)

Software

- [Review](#) ↗
- [Repository](#) ↗
- [Archive](#) ↗

Editor: [Leonardo Uieda](#) ↗

Reviewers:

- [@ThomasLecocq](#)
- [@seisman](#)

Submitted: 09 October 2019

Published: 06 April 2020

License

Authors of papers retain copyright and release the work under a Creative Commons Attribution 4.0 International License ([CC-BY](#)).

Summary

The receiver function method is a popular seismological technique to investigate velocity discontinuities in the crust and upper mantle, like the crust-mantle boundary (the Moho). The basic concept behind the method is that a part of the incident P-waves from a teleseismic event (e.g., a distant earthquake) gets converted to S-waves at significant discontinuities underneath a seismometer (receiver). The travel time difference between the first arriving P-wave and the converted S-wave constrains the depth of the discontinuity and the velocity ratio of P- and S-waves of the overlaying structure. To remove source side and propagation effects, receiver function calculation often involves a deconvolution of different components of the rotated seismogram.

rf is a Python package for calculating receiver functions. It provides the `RFStream` class that extends Obspy's `Stream` class (Krischer et al., 2015) with functionality needed for receiver function calculation. Among those are waveform input/output with metadata preservation, calculation of incidence angles for rotation of waves into ZRT or LQT coordinate systems, frequency domain or time domain deconvolution techniques, move-out correction, and calculation of and stacking by piercing points. Receiver functions of S to P converted waves can also be calculated.

rf is designed to be used by both researchers and students of seismology. Most often it will be used as a library in other researchers' code. Additionally, rf could act as glue for other codes with specialized functionality relevant to receiver function calculation. These codes are often written in Fortran and it is easy to integrate Fortran code into the project with NumPy's `f2py` (Peterson, 2009). As an example, rf already makes use of the Fortran Toeplitz package (Arushanian et al., 1983) for time domain deconvolution. The package can be used together with the recently open-sourced `telewavesim` package (Audet, Thomson, Bostock, & Eulenfeld, 2019) to invert for crustal velocity models. rf can be installed from PyPI. Online documentation and tutorials are available on the project site.

References

- Arushanian, O. B., Samarin, M. K., Voevodin, V. V., Tyrtshnikov, E. E., Garbow, B. S., Boyle, J. M., Cowell, W. R., et al. (1983). *TOEPLITZ package users' guide* (No. ANL-83-16). Moskovskij Gosudarstvennyj University, Argonne National Laboratory. Retrieved from <http://cds.cern.ch/record/151420>
- Audet, P., Thomson, C. J., Bostock, M. G., & Eulenfeld, T. (2019). Telewavesim: Python software for teleseismic body wave modeling. *Journal of Open Source Software*, 4(44), 1818. doi:[10.21105/joss.01818](https://doi.org/10.21105/joss.01818)
- Krischer, L., Megies, T., Barsch, R., Beyreuther, M., Lecocq, T., Caudron, C., & Wassermann, J. (2015). ObsPy: a bridge for seismology into the scientific Python ecosystem. *Computational Science & Discovery*, 8(1), 014003. doi:[10.1088/1749-4699/8/1/014003](https://doi.org/10.1088/1749-4699/8/1/014003)

Peterson, P. (2009). F2PY: A tool for connecting Fortran and Python programs. *International Journal of Computational Science and Engineering*, 4(4), 296–305. doi:[10.1504/IJCSE.2009.029165](https://doi.org/10.1504/IJCSE.2009.029165)