ELECTRONIC SEISMOLOGIST

Large-Scale Seismic Inversion Framework

by Lion Krischer, Andreas Fichtner, Saule Zukauskaite, and Heiner Igel

INTRODUCTION

Since its development and first applications in the late 1970s (e.g., Aki and Lee, 1976; Aki *et al.*, 1977; Dziewoński *et al.*, 1977), seismic tomography has developed into one of the most powerful tools to investigate the internal structure of the Earth from local to global scales. Tomographic Earth models have become increasingly detailed, thanks to the continuous densification of the global station network (e.g., Roult *et al.*, 2010; Gee and Leith, 2011), the installation of dedicated arrays (e.g., SKIPPY, van der Hilst *et al.*, 1994; USArray, www.usarray.org, last accessed November 2014; IberArray, Díaz *et al.*, 2009), and the deployment of ocean-bottom seismometers (e.g., Shiobara *et al.*, 2009; Obayashi *et al.*, 2013). Furthermore, methodological developments have sharpened our picture of the Earth. Depending on the nature of the data, the scientific question, and the available resources, seismic tomographers can choose from a rich variety of techniques, including ray tomography (e.g., Kissling, 1988; Spakman, 1991; Grand *et al.*, 1997; Rawlinson and Sambridge, 2003), various finite-frequency methods (e.g., Yomogida, 1992; Dahlen *et al.*, 2000; Friederich, 2003; Yoshizawa and Kennett, 2004, 2005), or full-waveform inversion based on numerical solutions of the wave equation (e.g., Tarantola, 1988; Chen *et al.*, 2007; Fichtner *et al.*, 2009; Zhu *et al.*, 2012; Fichtner *et al.*, 2013; Afanasiev *et al.*, 2014).

Improvements of data coverage and inversion technology give rise to new challenges that need to be addressed to ensure continued progress. These challenges include the following: (1) Exponentially growing amounts of data and metadata must be retrieved, organized, quality controlled, and updated. (2) Data and metadata are available in many different, often purpose-tailored formats and with variable pieces of information, which makes the handling of large datasets unnecessarily cumbersome. (3) The growing complexity of increasingly sophisticated tools reduces our ability to independently assess the results of tomographic inversions and to collaborate across different research groups. The flood of provenance information needed to enable reproduction of scientific results is increasingly difficult to organize. (4) The processing of large waveform datasets and the measurement of differences between observed and synthetic seismograms becomes too computationally demanding to be performed on a single compute core. Modern high-performance computing resources should thus be harnessed for both processing and measurements to avoid bottlenecks in the seismic inversion workflow and to ensure scalability. (5) The growing complexity of hardware architectures and software developments makes it impossible for single institutions or individual researchers to maintain stable and efficient solutions for computational tasks such as seismic waveform inversion. Therefore, in almost all branches of science, the development of stable community solutions plays an increasingly important role. Eventually, such solutions may be merged with evolving science gateways (e.g., the EU-funded VERCE project for seismology) that could provide high-level access to sophisticated IT applications to the scientific community.

The goal of the LArge-scale Seismic Inversion Framework (LASIF) is to provide solutions to the above-mentioned problems, thereby reducing the time needed for research.

LASIF provides a flexible structure linking the different components of a tomographic inversion, including the download and processing of data, the computation of synthetics, and window selection and measurements, as well as visualization and data exploration. As such, it offers functionality for the retrieval, organization, parallel processing, and visualization of seismic waveform data and metadata in a variety of different formats. Furthermore, LASIF provides tools for automatic and manual window selection, the parallel measurement of differences between observed and synthetic seismograms, and the computation of adjoint sources needed in the calculation of Fréchet kernels based on adjoint techniques (e.g., Tarantola, 1988; Tromp *et al.*, 2005; Fichtner *et al.*, 2006). The strict documentation of all operations performed increases reproducibility. Through its clearly defined structure, LASIF facilitates collaborative projects. Various visualization tools allow the user to explore data and to monitor the progress of iterative inversions. LASIF is written in Python and JavaScript, under the GPLv3 open-source license, and is freely available online (http://www.lasif.net; last accessed November 2014). The code features numerous internal testing routines that reduce the probability of programming errors, and extensive documentation and a tutorial are available online. Many routines are based on *NumPy*, *SciPy* (Jones *et al.*, 2001), and *ObsPy* (Beyreuther *et al.*, 2010; Krischer *et al.*, 2015) for the seismic analysis part.

This article is organized as follows. Following a summary of LASIF's design philosophy and general structure, we describe procedures for the download of event, waveform, and station metadata. This is followed by two paragraphs on waveform processing and the link between LASIF and forward problem solvers that

doi: 10.1785/0220140248

provide synthetic waveforms. Subsequently, we provide details on the automized selection of measurement windows, the computation of various misfit measures and corresponding adjoint sources, and the actual inversion procedure. To demonstrate LASIF's ability to solve real-data problems, we show results of an ongoing full-waveform tomography for the Japanese Islands region.

PHILOSOPHY AND STRUCTURE

LASIF represents the state of a tomographic inversion in a fixed and intuitively designed directory structure on disk, summarized in Figure 1. Tools for the modification, interpretation, bookkeeping, and visualization of the inversion infer all necessary information from the data, and modifying the data in turn modifies the state of the inversion. A number of unobtrusive caches, storing basic information about the data contained in LASIF, are employed to keep LASIF fast and responsive without getting in the users' way.

These basic design principles make LASIF a data-driven framework, and they result in a number of advantages compared to approaches relying on databases or bookkeeping files: (1) simple installation and maintenance because no database needs to be set up and kept running, which is especially important on high-performance platforms; (2) increased shareability and potential for collaboration as the fixed directory structure enables others to understand what has been done and what the next steps are; (3) straightforward integration with other tools; and (4) simple backups, which, coupled with continuous snapshots of the file system on modern platforms, also enable recovery from and rolling back of errors.

```
ProjectDirectory/
├── DATA/ ..... Observed data. Raw and processed.
│   ├── event_a/
│   │   ├── raw/
│   │   └── processed_.../
│   └── event_b/
├── EVENTS/ ..... Event definitions as QuakeML files.
│   ├── event_a.xml
│   └── event_b.xml
├── ITERATIONS/ ..... Definition of the iterations.
│   ├── ITERATION_1.xml
│   └── ITERATION_2.xml
├── LOGS/
├── MODELS/
├── STATIONS/ Station information in all common formats.
│   ├── RESP/
│   ├── SEED/
│   └── StationXML/
├── SYNTHETICS/ ..... Synthetic data.
│   ├── event_a/
│   │   ├── ITERATION_1/
│   │   └── ITERATION_2/
│   └── event_b/
├── .../
└── config.xml ..... Project configuration.
```

▲ **Figure 1.** The directory structure of LArge-scale Seismic Inversion Framework (LASIF). This example omits some folders for the sake of brevity. The stateful nature of LASIF means that as soon as some data is copied or created under it, LASIF is aware of it.

The internal structure of LASIF is strictly modular, with individual components being responsible for comparatively simple tasks, such as the retrieval of station and event information, the processing of a waveform, or the calculation of a misfit. The modularity of LASIF facilitates code maintenance and the addition of new features.

Modules interact with the help of three different user interfaces to perform more sophisticated operations. LASIF's web interface, a screenshot of which is shown in Figure 2, allows the user to visually explore event and waveform data and to monitor the evolution of synthetic waveforms in the course of an iterative inversion. The command line interface is used to steer the tomographic inversion. Executing, for instance, the UNIX shell command

```
$ lasif init_project Example
```

creates a new LASIF project entitled *Example* by setting up the directory structure from Figure 1, as well as initial configuration files. Furthermore, the command line interface can be used to retrieve waveform and metadata from online data centers, to preprocess data, and to automatically select measurement windows. Additional examples involving the command line interface are provided in the following sections. Finally, a measurement interface can be used to select windows manually and to inspect observed and synthetic waveforms.

EVENT, WAVEFORM, AND METADATA

LASIF offers various tools for the retrieval of event, waveform, and metadata from online data centers. Executing, for instance, the built-in command

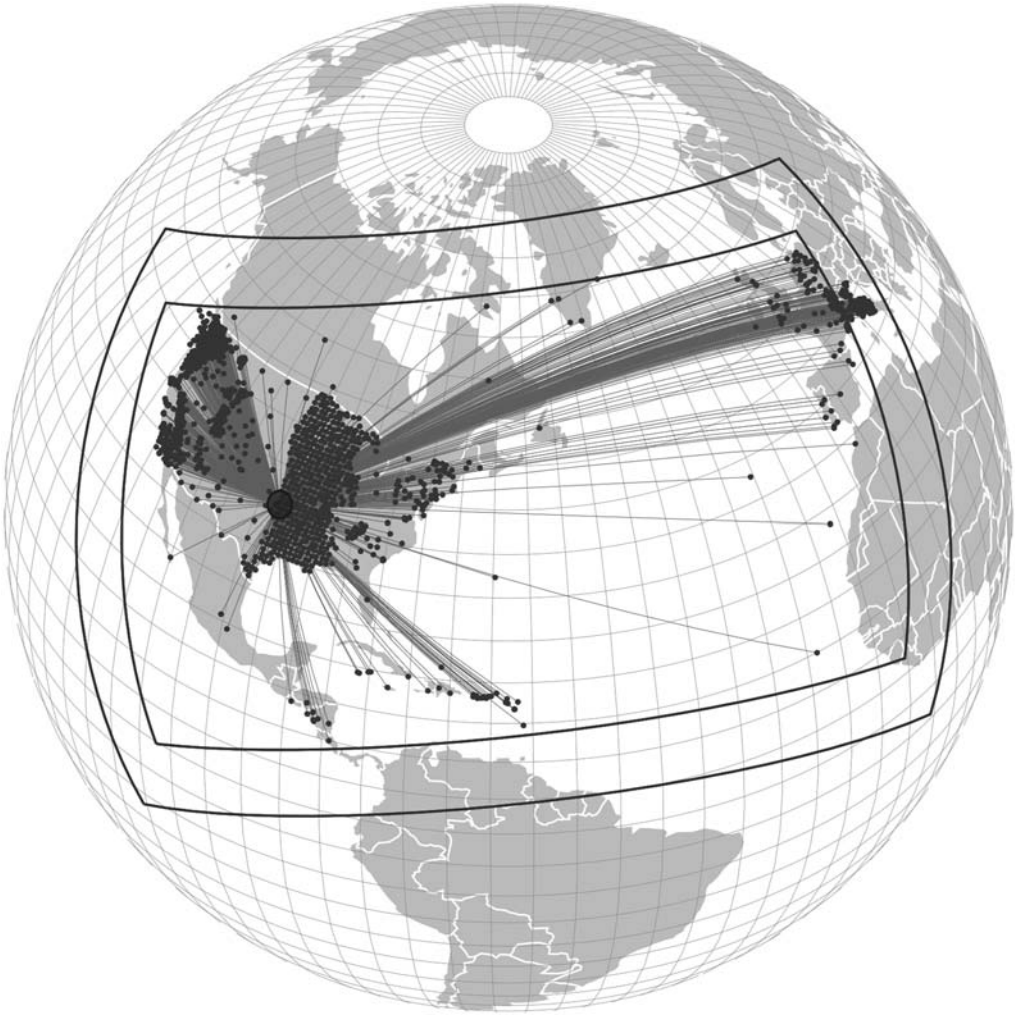
```
$ lasif add_gcmt_events –min_year 2005 10 5 7 250
```

will query the Global Centroid Moment Tensor project catalog (Ekström *et al.*, 2012) to add up to 10 earthquakes, from 2005 or later, with magnitudes between 5 and 7, and a minimum interevent distance of 250 km to the current project. The event distribution is optimal in the sense that it approximates a Poisson disk distribution. This is intended to generate a set of events with good data coverage and few redundancies. Each new event is chosen from all available events by having the largest possible minimum distance to the next closest earthquake already part of the project, while still satisfying the geographic, time, and magnitude constraints. An example of automatically selected events is presented in Figure 3. Alternatively, individual events can be added to the project via the Incorporated Research Institutions for Seismology (IRIS) SPUD service (www.iris.edu/spud/momenttensor; last accessed November 2014), in which the command

```
$ lasif add_spud_event http://www.iris.edu/spud \
momenttensor/example
```

adds the event with ID example to the EVENTS/ folder. All event information is written in the form of QuakeML files.

Following the retrieval of event information, waveform data can be obtained by invoking LASIF's download_data



▲ **Figure 2.** A screenshot of LASIF's web interface which can be launched with the lasif serve command at any point. The example shows the interactive map currently set to display the ray paths and recording stations for a single event. The main purpose of the web interface is to interactively explore the dataset and state of the inversion.

command. Assuming the user has defined a QuakeML file GCMT_event_ROMANIA.xml describing an event, then the command

```
$ lasif download_data GCMT_event_ROMANIA
```

queries a collection of FDSN webservice providers and automatically downloads all waveform and station data it can find for the time frame of that event. In addition to LASIF, any other tool may be used by simply copying data into the correct folders, in this case DATA/ and STATION/, respectively.

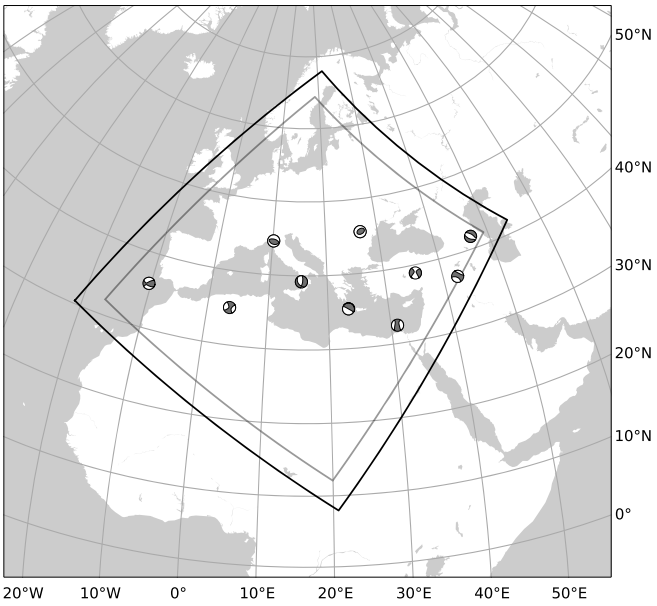
To honor the real world situation of multiple data providers with different standards, LASIF has been designed to be as format-agnostic as possible. Although we recommend using MiniSEED for waveform data and StationXML for station data, LASIF can also deal with Seismic Analysis Code (SAC), Group of Scientific Experts Format Version 2 (GSE2), Standard for Exchange of Earthquake Data (SEED), RESP, and a variety of other file formats and any combination of them. This is achieved by utilizing *ObsPy* (Beyreuther *et al.*, 2010; Megies *et al.*, 2011) wherever possible. As a fallback for some combinations of waveform and station data that do not contain station coordinates, LASIF can query webservices to complement the dataset with the missing information.

DATA PROCESSING

Data processing in LASIF is intended to correct and filter waveform data and to ensure the compatibility of observed and synthetic waveforms. Taking information on the time stepping and frequency band of the forward problem's solution, the command

```
$ mpirun -n 16 lasif preprocess_data 1
```

processes all data used in iteration 1 on 16 CPUs. It can also be invoked without the message-passing interface (MPI), resulting in execution on only one core. The processing of observed waveforms includes the following operations: (1) removal of the mean and linear trends, (2) tapering, (3) band-pass filtering to the frequency band used in the computation of synthetic seismograms, (4) removal of the instrument response, and



▲ **Figure 3.** A small set of automatically selected events. The map shows the unedited output of $ lasif plot_events, which is one of several visualization commands available in LASIF. The black lines mark the boundaries of the simulation domain, the gray inner lines show an optional buffer zone used to safeguard against boundary effects from numerical waveform solvers.

(5) downsampling or interpolation to a sampling interval that equals the time step of the forward problem solution.

LASIF's nature enables it to make good choices for many of the parameters required for these operations. Further required information is stored in iteration XML files, which are explained in the later inversion section. To minimize the time required for these tasks, the processing in LASIF is fully parallelized, using MPI. This parallelism allows users to process data on a large number of compute cores.

The data processing is fully configurable on a per-project and iteration basis. Furthermore LASIF can optionally process synthetic data, which might be necessary depending on the specifics of the chosen inversion workflow. This processing will be applied on-the-fly anytime synthetics are required for an operation.

SYNTHETIC DATA

LASIF provides functionality to generate input files for seismic wave propagation solvers. Taking the previously compiled information about events and stations, LASIF can currently produce input files for the global spectral-element solver SPECFEM3D GLOBE (e.g., Komatitsch and Tromp, 2002a,b; Peter *et al.*, 2011), and the regional-scale spectral-element solver SES3D (Fichtner and Igel, 2008; Fichtner *et al.*, 2009). Thanks to the modular structure of LASIF, input file generators for other wave equation solvers can be added easily. LASIF's responsibility stops here, and the users are expected to copy the input files to an available high-performance computer, run the simulations, and move the resulting synthetics to the project directory managed by LASIF.

WINDOW SELECTION

The selection of time windows for the comparison of observed and synthetic data is a critical aspect of seismic tomography. It strongly affects resolution, convergence, and the impact of noise on the final Earth model. In addition to the manual window selection in the measurement interface, LASIF offers an automatic window selection. Similar to FLEXWIN, developed by Maggi *et al.* (2009), LASIF's window selection algorithm was originally developed for full-waveform inversion applications where complete seismograms, in principle, can be assimilated into the inversion. However, the algorithm can be tuned to select, for instance, specific body or surface-wave phases. It has been tested and successfully applied in inversions ranging from regional and continental scales (Fichtner *et al.*, 2013) to the full globe.

The window selection operates on pairs of observed and synthetic waveforms, assuming both have been appropriately processed. In addition to the waveforms, the algorithm takes the following inputs: locations of source and receiver, the minimum and maximum period, and a set of adjustable parameters summarized in Table 1.

The algorithm proceeds in four steps that are detailed in the paragraphs below: (1) determination of window bounds based on travel times, (2) global trace rejection based on the noise level and the overall similarity between observations and synthetics, (3) preselection of windows based on a sliding cross correlation, and (4) a number of successive elimination stages involving amplitude ratios, the minimum window length, and various other criteria.

Window Bounds Based on Travel Times

The first stage of the automatic window selection determines the bounds of all possible windows based on the theoretical travel times of seismic phases. The first body-wave arrival computed for the 1D Earth model ak135 (Kennett *et al.*, 1995) marks the lower bound, and the minimum surface-wave velocity min_velocity (see Table 1) marks the upper bound. At both ends, a buffer of half the minimum period of the data is added to account for the effects of (a)causal filters.

Global Rejection Criteria

Prior to the detailed selection of time windows, the algorithm rejects data based on their noise level and overall similarity to the synthetics.

The relative noise level is defined as the ratio between the maximum amplitude prior to the first arrival and the maximum amplitude in the complete seismogram. Data are rejected when the relative noise level is above max_noise. The definition of noise is to some extent subjective. It could be improved in future versions of LASIF using, for instance, the upcoming IRIS MUSTANG service (IRIS, 2015) that is currently in the testing phase.

Table 1
Parameters for the Window Selection Algorithm

Global Rejection Parameters	
min_cc	Minimum normalized correlation coefficient between observed and synthetic traces
max_noise	Maximum relative noise level of the data trace
Window Acceptance/Rejection Parameters	
min_velocity	Minimum apparent velocity; later arrivals are rejected
threshold_shift	Maximum cross-correlation time shift within a sliding window
threshold_corr	Minimum normalized correlation coefficient within a sliding window.
min_length_period	Minimum length of a time window relative to the minimum period
min_peaks_troughs	Minimum number of extrema in an individual window
max_energy_ratio	Maximum energy ratio between observed and synthetic data within a window
max_noise_window	Maximum relative noise level for individual windows

Correlation coefficients are normalized to range between −1.0 and 1.0, time durations are expressed as fractions of the minimum period of the input data.

To ensure a basic comparability of observed and synthetic seismograms, the normalized zero-lag correlation coefficient

$$cc = \frac{\mathbf{d}^T\mathbf{s}}{\sqrt{(\mathbf{d}^T\mathbf{d})(\mathbf{s}^T\mathbf{s})}} \tag{1}$$

must not be lower than min_cc. In equation (1), $\mathbf{d}$ and $\mathbf{s}$ denote the arrays of observed and synthetic waveforms, respectively. A strongly negative correlation coefficient can indicate problems with the polarity and may be used as a criterion for flipping data.

Sliding Cross Correlation

Provided that data pass the global rejection criteria, LASIF makes a selection of candidate windows using a sliding cross-correlation technique that is intended to avoid cycle skips. With the discrete cross correlation between two arrays $\mathbf{f}$ and $\mathbf{g}$ defined as

$$(\mathbf{f} * \mathbf{g})[n] = \sum_m f[m]g[n-m], \tag{2}$$

the sliding normalized cross correlation of observed data $\mathbf{d}_i$ and synthetic data $\mathbf{s}_i$ windowed around index i is given by

$$cc_i = \frac{\mathbf{d}_i * \mathbf{s}_i}{\sqrt{(\mathbf{d}_i^T\mathbf{d}_i)(\mathbf{s}_i^T\mathbf{s}_i)}}. \tag{3}$$

The current implementation of LASIF uses a Hanning window with a length equal to twice the minimum period. Different sliding windows can be implemented with ease when needed.

At each index i, the maximum is extracted, yielding the maximum correlation at each point in time. Furthermore, the time shift is computed as the lag time where the maximum correlation occurs. A time index i is kept as a candidate index when the maximum correlation is above threshold_corr and when the time shift is below threshold_shift.

Elimination Phases

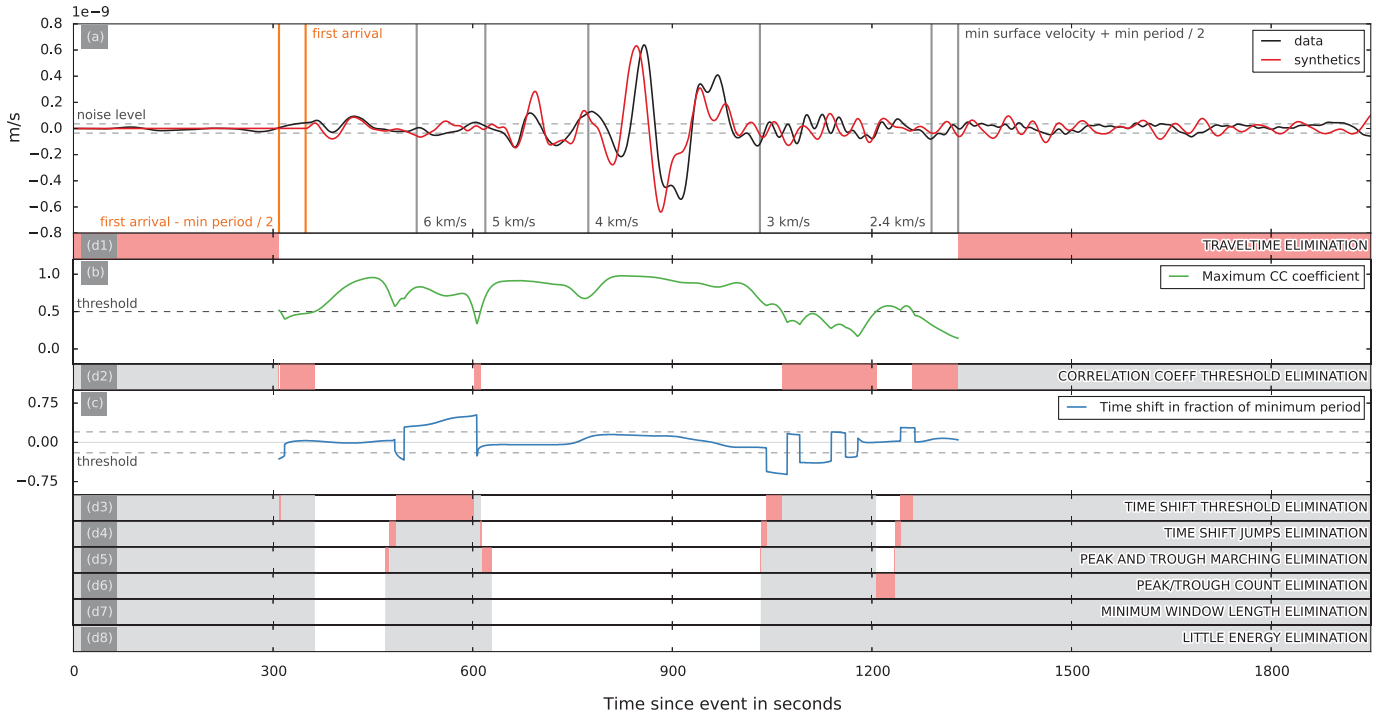
The algorithm proceeds with the following elimination phases, which are intended to exclude time intervals where observed and synthetic waveforms differ too much.

(1) A buffer around each jump in the cross-correlation time shift is marked as invalid. The occurrence of such jumps, illustrated by the blue curve in Figure 4, is indicative of cycle skips that the algorithm attempts to avoid. (2) The peaks and troughs of observed and synthetic waveforms are detected by finding local extrema. Intervals where the timing of matching peaks and troughs differs by more than half the minimum period are marked as invalid. This criterion is primarily intended to detect high-frequency oscillations on top of lower-frequency data. (3) Windows with less then min_peaks_troughs local extrema are discarded. (4) Windows shorter than min_length_period are excluded. (5) Windows where the maximum amplitude divided by the absolute noise level prior to the first arrival is smaller than max_noise_window are eliminated as well. (6) Candidate windows are kept only when the amplitudes in the ratio between observed and synthetic amplitudes is below max_energy_ratio.

Automatic window selection algorithms should generally not be used blindly because the (to some extent subjective) goodness of the adjustable parameters is strongly data and application dependent. Considering the immense quantity of waveform data that are available today, we recommend manually tuning the window selection parameters with a small subset of the data. The selection parameters can then be used to compute time windows for the remaining data. A conservative choice is generally advisable because the damage caused by inappropriate windows typically outweighs the benefit of having slightly more windows.

MISFIT MEASUREMENTS AND ADJOINT SOURCES

Once appropriate time windows have been selected, LASIF can compute various types of misfit measures between observations



▲ **Figure 4.** Graphical illustration of the window selection algorithm. (a) Observed and synthetic seismograms, including the theoretical arrival times of the first body-wave phase for model ak135 (Kennett *et al.*, 1995) in orange. The arrival times for a range of apparent surface-wave velocities are plotted in gray. The noise level estimated from the amplitudes prior to the first arrival is indicated by the gray dashed lines. (b) and (c) Maximum windowed cross-correlation coefficient and the corresponding time shift, respectively. (d1)–(d8) Successive elimination stages of the window selection algorithm. In each stage, gray corresponds to the time intervals that have been eliminated in the previous stages. Red time intervals are eliminated in the current stage, and white corresponds to the time intervals that are still being considered. Thus, the white intervals in the bottom bar represent the final time windows.

and synthetics, as well as the corresponding adjoint sources needed for the calculation of Fréchet kernels via adjoint techniques (e.g., Tarantola, 1988; Tromp *et al.*, 2005; Fichtner *et al.*, 2006). Executing the command

```
$ lasif finalize_adjoint_sources 1 GCMT_event_ROMANIA
```

performs this task for iteration 1 and the chosen event. For each chosen window, it will calculate the misfit and derive the associated adjoint source; it will then combine all measurements for a single component, weight them, and produce the final adjoint source for that component. Weighting can be done per event, per station, and also per window. The adjoint sources will be stored in whatever format the chosen numerical waveform solver requires.

Currently, implemented misfit measures include the L_2 waveform difference typically used in exploration applications (e.g., Igel *et al.*, 1996; Pratt *et al.*, 1998; Afanasiev *et al.*, 2014), the cross-correlation travel-time shift used in waveform travel-time inversion (Luo and Schuster, 1991), and the time-frequency phase misfit (Fichtner *et al.*, 2008, 2013).

The modularity of LASIF allows for the straightforward implementation of additional misfit measures, such as, for instance, multitaper measurements (e.g., Laske and Masters, 1996; Zhou *et al.*, 2004; Tape *et al.*, 2010) or generalized seismological data functionals (Gee and Jordan, 1992).

INVERSION

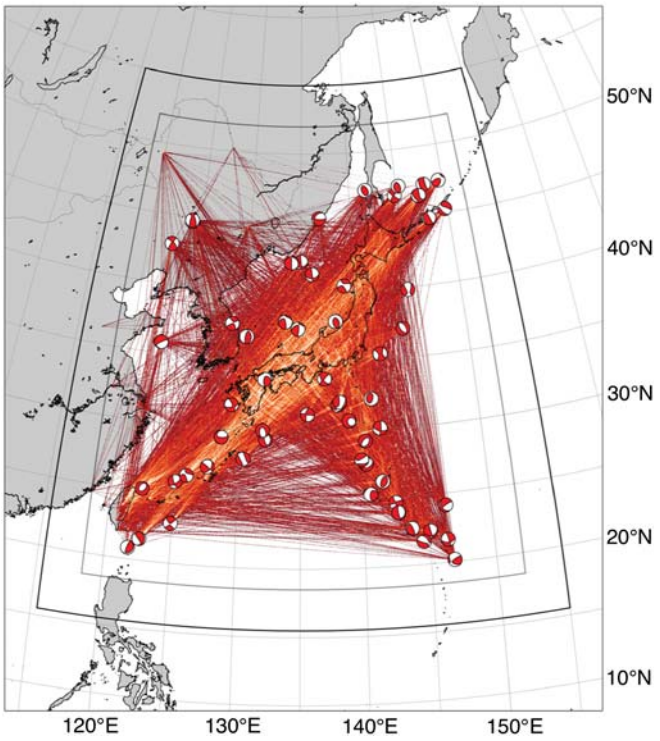
A key functionality of LASIF consists of the tracking of the inversion process through a series of iterations. When event and station information and waveform data are available, a new iteration can be defined via the command line interface

```
$ lasif create_new_iteration iteration_name passband \
  forward_solver
```

All relevant information about an iteration is stored in a custom XML file that can be read and modified by any modern programming language. The iteration XML file contains (1) information on the frequency passband, (2) a list of all stations for each event with optional weighting factors and time corrections, and (3) the name of the forward problem solver, plus all setup parameters needed to run forward simulations.

The iteration XML files for a sequence of iterations keep a large part of the provenance information in a compact form, thereby facilitating reproducibility and collaborative inversion projects. Furthermore, the iteration XML files serve as input for the data preprocessing, the automatic window selection algorithm, the computation of misfits and adjoint sources, and numerous other functionalities of LASIF.

Progressing from the current to the next iteration, requires the generation of a successor to the current iteration XML file,



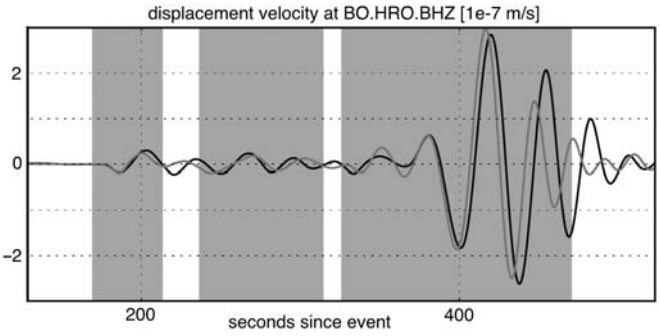
▲ **Figure 5.** Ray density map for the study region. Produced with the lasif plot_raydensity command, which extracts the required information from the project file structure. It will only plot ray paths for data that are actually part of the project.

as well as the translation of the current time windows to the next iteration. These tasks can be performed also via LASIF's command line interface:

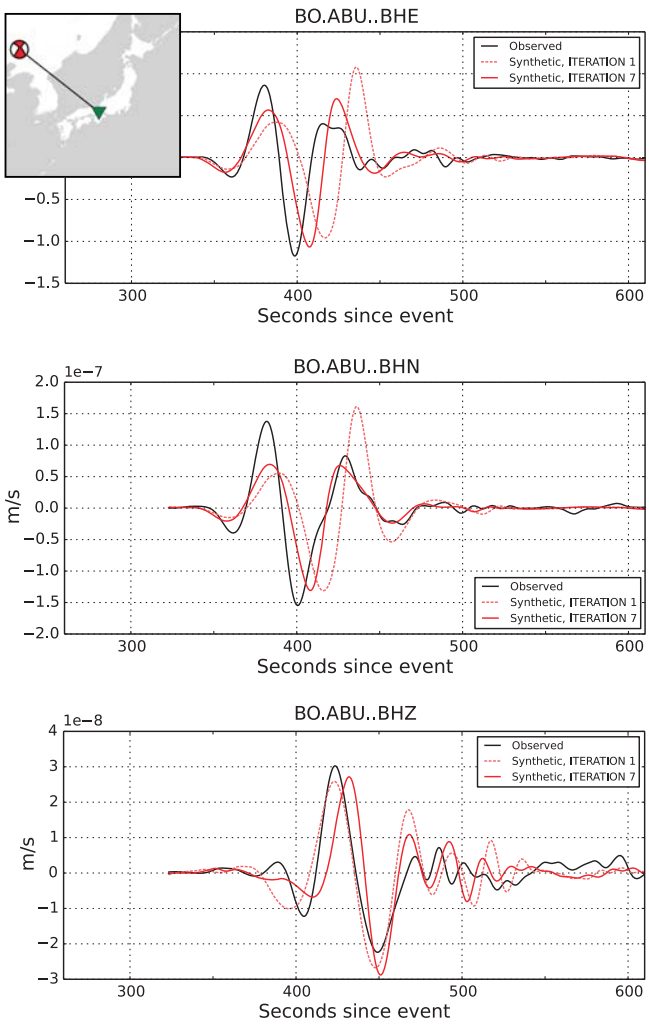
```
$ lasif create_successive_iteration current_iteration \
  name next_iteration_name

$ lasif migrate_windows current_iteration_name next \
  iteration_name
```

WHAT LASIF DOES NOT DO (BY DESIGN)

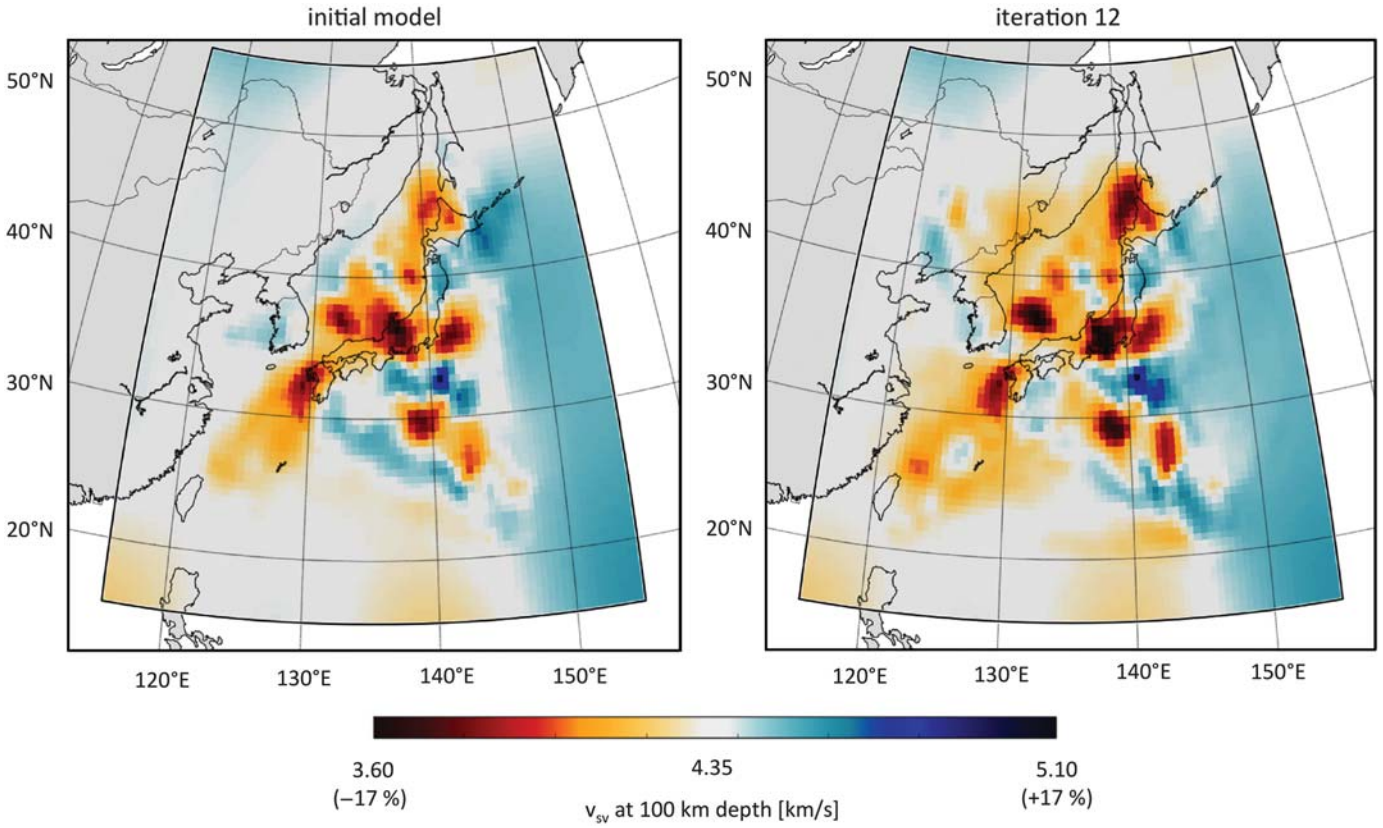
LASIF provides a basic functionality for the computation of iterative model updates in the form of a Python script that computes steepest-descent and conjugate-gradient updates. Given the enormous amount of different optimization and regularization schemes, this script is deliberately simplistic, merely outlining the general procedure involved in the computation of a model update in a gradient-based inversion scheme. Furthermore, LASIF contains no means to manage and deal with the potentially massive volumes of kernels and model updates. We made these decisions for simplicity in order to keep LASIF maintainable and efficient. Thus LASIF offers no push-button solution to full waveform inversions but significantly facilitates and stabilizes them.



▲ **Figure 6.** Measurement time windows on a vertical-component velocity seismogram recorded at station BO.HRO. Windows are selected in time intervals where observed and synthetic seismograms are sufficiently close to allow for their meaningful comparison.



▲ **Figure 7.** Waveform comparison between iteration 1 (dashed light red line) and iteration 7 (solid red line) for an M_w 5.0 event in northeastern China and station BO.ABU. Observed data are plotted in black. Although the waveform fit for horizontal components improves substantially, the fit in the vertical component slightly declines.



▲ **Figure 8.** Comparison of the *SV* velocity at 100 km depth in the initial model (left; Diaz-Steptoe, 2013) and the model after 12 iterations (right).

APPLICATION

We illustrate some of LASIF's functionality and visualization tools with an example waveform inversion in East Asia. The study area, shown in Figure 5, covers the Japanese islands, Taiwan, the Korean peninsula, the easternmost parts of China and Russia, Sakhalin, and the majority of the Kuril Island chain. Because of the presence of numerous plate boundaries between the Pacific, Philippine Sea, Okinawa, Sunda, Yangtze, and Amur plates (Bird, 2003), the Earth's structure in the region is exceptionally complex.

Within the model domain, we selected 58 earthquakes, distributed spatially as uniformly as possible and with magnitudes ranging between M_w 5.0 and 6.9. We obtained waveform data from all freely available seismic networks in the area, namely the Full Range Seismograph Network of Japan, the Broadband Array in Taiwan for Seismology, the Korea National Seismographic Network and several stations from the China National Seismic Network, the New China Digital Seismograph Network, the Global Seismograph Network, and the Korea National Seismographic Network, made available by IRIS Data Management Center. With 165 available seismic stations and 58 events, our dataset contains more than 5500 three-component waveforms. A ray density plot that provides a first rough estimate of the achievable tomographic resolution can be produced via LASIF's command line interface (Fig. 5). For the forward simulations we use the spectral-element wave propagation code SES3D (Fichtner and Igel, 2008; Fichtner *et al.*, 2009), run on the high-performance computers of the Swiss National Supercomputing Centre. LASIF produces all relevant SES3D input, including the geometric setup, parallelization, viscoelastic relaxation parameters, source–time function, earthquake source parameters, and receiver positions. The automatic generation of input files for the forward solver reduces the risk of errors and facilitates reproducibility.

To ensure meaningful measurements of waveform differences, LASIF applies the same processing to observed and synthetic waveforms. Using the tunable automatic window selection described above, we determine an initial set of measurement windows that we adjust manually when needed. An example window selection as it appears in LASIF's measurement interface is shown in Figure 6. To first constrain the long wavelength structure, we started with longer-period data filtered between 50 and 80 s. In total, we selected around 4000 measurement windows where the time–frequency phase differences between observed and synthetic seismograms, as well as the corresponding adjoint sources, were calculated (Fichtner *et al.*, 2008). Taking the 3D model of Diaz-Steptoe (2013) as the initial model, we achieved a misfit reduction of 27% after six iterations. Figure 7 visualizes the improving match between observations and synthetics that can be monitored through LASIF's web interface.

Subsequently, we broadened the period band to 30–80 s and selected around 5500 new measurement windows. Another six iterations reduced the misfits by 19%, leading to the model displayed in Figure 8.

Using LASIF's command line interface, the inversion procedure outlined above can be fully automized. This, however, does not mean that LASIF should be used as a black box. Human intuition remains essential for the meaningful solution of any ill-posed inverse problem, including seismic tomography. Nonetheless, LASIF enabled significant improvements in speed resulting in this inversion being carried out by a student in the course of a master's thesis.

CONCLUSIONS AND PERSPECTIVES

We present a data management and inversion framework for potentially large-scale seismic tomography problems. LASIF is intended to increase the quality of and reduce the time to research. It does so by providing solutions to current challenges, the rapidly growing amount of seismic data, the existence of different data formats, and the decreasing reproducibility of increasingly complex inversions.

Written mostly in Python, LASIF has a modular structure that facilitates maintenance and the addition of new features. LASIF is well documented, open source, and freely available online (http://www.lasif.net; last accessed November 2014), and its source code is managed via GitHub. LASIF includes (1) tools for the download of event, waveform, and station data, (2) a command line and a web interface to explore data and monitor the progress of an inversion, (3) tools for data processing, (4) tools for the generation of input files needed in forward problem solvers, (5) a tunable automatic window selection algorithm, (6) routines for the calculation of various waveform misfit measures and corresponding adjoint sources, and (7) a wide range of visualization tools.

Although LASIF is a production-stage code, several future developments could still be envisioned. The incorporation of noise correlation data, for instance, currently requires a deliberate misuse of data formats that were originally designed for earthquake or active-source data. The design of a generic format for noise correlations with their complex processing history (e.g., Bensen *et al.*, 2007), and the incorporation of this format into LASIF, has the potential to greatly improve the efficiency and reproducibility of noise tomography.

Other types of datasets with unique features, like scattered body waves used in the receiver function community, could be utilized within LASIF with only slight modifications. LASIF is independent of the numerical waveform solver, so it is straightforward to integrate, for example, hybrid methods (e.g., Tong *et al.*, 2014) and define additional misfit functionals.

Furthermore, the interfacing of LASIF with a nonlinear optimization toolbox, as well as tools for the exchange of data with high-performance computers are currently being considered. The incorporation of such new features has to be weighted against the increasing complexity of the code.

Eventually it is conceivable that entire work flows such as LASIF can be offered to the community through gateways as envisaged in the VERCE project (http://www.verce.eu; last accessed November 2014). In the future, it is important that such software products are treated as (real) infrastructure by the communities and funding agencies with sustained support. Although this might require a paradigm shift, without it we will not be able to make efficient use of the continuously expanding cyberinfrastructures for our sciences.

ACKNOWLEDGMENTS

We would like to thank Editor Zhigang Peng, as well as Qinya Liu and Carl Tape for their thoughtful and constructive reviews, which helped improve the manuscript. The development of LASIF, as well as a series of pilot applications were supported by the EU-FP7 VERCE project (number 283543), the Swiss National Supercomputing Centre (CSCS) through the CHRONOS Project ch1, and by the Platform for Advanced Scientific Computing (PASC). The authors are grateful to the first users of LASIF, Michael Afanasiev, Yesim Cubuk, Erdinc Saygin, Katrin Peters, and Korbinian Sager.

REFERENCES

Afanasiev, M. V., R. G. Pratt, R. Kamei, and G. McDowell (2014). Waveform-based simulated annealing of crosshole transmission data: A semi-global method for estimating seismic anisotropy, *Geophys. J. Int.* **199,** 1586–1607.

Aki, K., and W. H. K. Lee (1976). Determination of three-dimensional velocity anomalies under a seismic array using first *P* arrival times from local earthquakes: 1. A homogeneous initial model, *J. Geophys. Res.* **81,** 4381–4399.

Aki, K., A. Christoffersson, and E. S. Husebye (1977). Determination of the three-dimensional seismic structure of the lithosphere, *J. Geophys. Res.* **82,** 227–296.

Bensen, G. D., M. H. Ritzwoller, M. P. Barmin, A. L. Levshin, F. Lin, M. P. Moschetti, N. M. Shapiro, and Y. Yang (2007). Processing seismic ambient noise data to obtain reliable broad-band surface wave dispersion measurements, *Geophys. J. Int.* **169,** 1239–1260.

Beyreuther, M., R. Barsch, L. Krischer, and J. Wassermann (2010). ObsPy: A Python toolbox for seismology, *Seismol. Res. Lett.* **81,** 47–58.

Bird, P. (2003). An updated digital model of plate boundaries, *Geochem. Geophys. Geosyst.* **4,** 1027–1079.

Chen, P., L. Zhao, and T. H. Jordan (2007). Full 3D tomography for the crustal structure of the Los Angeles region, *Bull. Seismol. Soc. Am.* **97,** 1094–1120.

Dahlen, F., S.-H. Hung, and G. Nolet (2000). Fréchet kernels for finite-frequency traveltimes: I. Theory, *Geophys. J. Int.* **141,** 157–174.

Díaz, J., A. Villaseñor, J. Gallart, J. Morales, A. Pazos, D. Córdoba, J. Pulgar, J. L. García-Lobón, M. Harnafi, and TopoIberia Seismic Working Group (2009). The IBERARRAY broadband seismic network: A new tool to investigate the deep structure beneath Iberia, *ORFEUS Newsletter* **8,** 1–6.

Diaz-Steptoe, H. (2013). Full seismic waveform tomography of the Japan region using adjoint methods, *Master's Thesis*, Utrecht University, The Netherlands.

Dziewoński, A. M., B. H. Hager, and R. J. O'Connell (1977). Large-scale heterogeneities in the lower mantle, *J. Geophys. Res.* **82,** 239–255.

Ekström, G., M. Nettles, and A. M. Dziewoński (2012). The Global CMT project 2004–2010: Centroid-moment tensors for 13,017 earthquakes, *Phys. Earth Planet. In.* **200/201,** 1–9.

Fichtner, A., and H. Igel (2008). Efficient numerical surface wave propagation through the optimization of discrete crustal models: -A technique based on non-linear dispersion curve matching (DCM), *Geophys. J. Int.* **173,** 519–533.

Fichtner, A., H.-P. Bunge, and H. Igel (2006). The adjoint method in seismology: I. Theory, *Phys. Earth Planet. In.* **157,** 86–104.

Fichtner, A., B. L. N. Kennett, H. Igel, and H.-P. Bunge (2008). Theoretical background for continental- and global-scale full-waveform inversion in the time-frequency domain, *Geophys. J. Int.* **175,** 665–685.
Fichtner, A., B. L. N. Kennett, H. Igel, and H.-P. Bunge (2009). Full seismic waveform tomography for upper-mantle structure in the Australasian region using adjoint methods, *Geophys. J. Int.* **179,** 1703–1725.
Fichtner, A., J. Trampert, P. Cupillard, E. Saygin, T. Taymaz, Y. Capdeville, and A. Villasenor (2013a). Multiscale full waveform inversion, *Geophys. J. Int.* **194,** no. 1, 534–556, doi: 10.1093/gji/ggt118.
Friederich, W. (2003). The S-velocity structure of the East Asian mantle from inversion of shear and surface waveforms, *Geophys. J. Int.* **153,** 88–102.
Gee, L. S., and T. H. Jordan (1992). Generalized seismological data functionals, *Geophys. J. Int.* **111,** 363–390.
Gee, L. S., and W. S. Leith (2011). The Global Seismographic Network, *U.S. Geological Fact Sheet, 20113021*.
Grand, S., R. van der Hilst, and S. Widiyantoro (1997). Global seismic tomography: A snapshot of convection in the earth, *Geol. Soc. Am. Today* **7,** no. 4, 1–7.
Igel, H., H. Djikpesse, and A. Tarantola (1996). Waveform inversion of marine reflection seismograms for *P* impedance and Poisson's ratio, *Geophys. J. Int.* **124,** 363–371.
Incorporated Research Institutions for Seismology (IRIS) (2015). IRIS Mustang, available online at http://service.iris.edu/mustang/ (last accessed April 2015).
Jones, E., T. Oliphant, and P. Peterson, and others (2001). SciPy: Open source scientific tools for Python, available online at http://www.scipy.org/ (last accessed October 2014).
Kennett, B. L. N., E. R. Engdahl, and R. Buland (1995). Constraints on seismic velocities in the Earth from traveltimes, *Geophys. J. Int.* **122,** 108–124.
Kissling, E. (1988). Geotomography with local earthquake data, *Rev. Geophys.* **26,** 659–698.
Komatitsch, D., and J. Tromp (2002a). Spectral-element simulations of global seismic wave propagation, Part I: Validation, *Geophys. J. Int.* **149,** 390–412.
Komatitsch, D., and J. Tromp (2002b). Spectral-element simulations of global seismic wave propagation, Part II: 3-D models, oceans, rotation, and gravity, *Geophys. J. Int.* **150,** 303–318.
Krischer, L., T. Megies, R. Barsch, M. Beyreuther, T. Lecocq, C. Caudron, and J. Wassermann (2015). ObsPy: A bridge for seismology into the scientific Python ecosystem, *Comput. Sci. Discov.* **8,** no. 1, 014003, doi: 10.1088/1749-4699/8/1/014003.
Laske, G., and G. Masters (1996). Constraints on global phase velocity maps from long-period polarization data, *J. Geophys. Res.* **101,** 16,059–16,075.
Luo, Y., and G. T. Schuster (1991). Wave-equation traveltime inversion, *Geophysics* **56,** 645–653.
Maggi, A., C. Tape, M. Chen, D. Chao, and J. Tromp (2009). An automated time-window selection algorithm for seismic tomography, *Geophys. J. Int.* **178,** no. 1, 257–281.
Megies, T., M. Beyreuther, R. Barsch, L. Krischer, and J. Wassermann (2011). ObsPy: What can it do for datacenters and observatories? *Ann. Geophys.* **54,** 47–58.
Obayashi, M., J. Yoshimitsu, G. Nolet, Y. Fukao, H. Shiobara, H. Sugioka, H. Miyamachi, and Y. Gao (2013). Finite frequency whole mantle *P* wave tomography: Improvement of subducted slab images, *Geophys. Res. Lett.* **40,** 1–6.
Peter, D., D. Komatitsch, Y. Luo, R. Martin, N. Le Goff, E. Casarotti, P. Le Loher, F. Magnoni, Q. Liu, C. Blitz, *et al.* (2011). Forward and adjoint simulations of seismic wave propagation on fully unstructured hexahedral meshes, *Geophys. J. Int.* **186,** 721–739.
Pratt, R., C. Shin, and G. Hicks (1998). Gauss–Newton and full Newton methods in frequency domain seismic waveform inversion, *Geophys. J. Int.* **133,** 341–362.
Rawlinson, N., and M. Sambridge (2003). Seismic traveltime tomography of the crust and lithosphere, *Adv. Geophys.* **46,** 81–199.
Roult, G., J.-P. Montagner, B. Romanowicz, M. Cara, D. Rouland, R. Pillet, J.-F. Karczewski, L. Rivera, E. Stutzmann, and A. Maggi (2010). The GEOSCOPE program: Progress and challenges during the past 30 years, *Seismol. Res. Lett.* **81,** 427–452.
Shiobara, H., K. Baba, H. Utada, and Y. Fukao (2009). Ocean bottom array probes stagnant slab beneath the Philippine Sea, *Eos Trans. AGU* **90,** 70–71.
Spakman, W. (1991). Delay-time tomography of the upper mantle below Europe, the Mediterranean and Asia Minor, *Geophys. J. Int.* **107,** 309–332.
Tape, C., Q. Liu, A. Maggi, and J. Tromp (2010). Seismic tomography of the southern California crust based upon spectral-element and adjoint methods, *Geophys. J. Int.* **180,** 433–462.
Tarantola, A. (1988). Theoretical background for the inversion of seismic waveforms, including elasticity and attenuation, *Pure Appl. Geophys.* **128,** 365–399.
Tong, P., D. Komatitsch, T.-L. Tseng, S.-H. Hung, C.-W. Chen, P. Basini, and Q. Liu (2014). A 3D spectral-element and frequency–wave number hybrid method for high-resolution seismic array imaging, *Geophys. Res. Lett.* **41,** 7025–7034.
Tromp, J., C. Tape, and Q. Liu (2005). Seismic tomography, adjoint methods, time reversal and banana-doughnut kernels, *Geophys. J. Int.* **160,** 195–216.
van der Hilst, R. D., B. L. N. Kennett, D. Christie, and J. Grant (1994). Project SKIPPY explores the lithosphere and mantle beneath Australia, *Eos Trans. AGU* **75,** 180–181.
Yomogida, K. (1992). Fresnel zone inversion for lateral heterogeneities in the earth, *Pure Appl. Geophys.* **138,** 391–406.
Yoshizawa, K., and B. L. N. Kennett (2004). Multimode surface wave tomography for the Australian region using a three-stage approach incorporating finite frequency effects, *J. Geophys. Res.* **109,** B02310, doi: 10.1029/2002JB002254.
Yoshizawa, K., and B. L. N. Kennett (2005). Sensitivity kernels for finite-frequency surface waves, *Geophys. J. Int.* **162,** 910–926.
Zhou, Y., F. A. Dahlen, and G. Nolet (2004). Three-dimensional sensitivity kernels for surface wave observables, *Geophys. J. Int.* **158,** 142–168.
Zhu, H., E. Bozdağ, D. Peter, and J. Tromp (2012). Structure of the European upper mantle revealed by adjoint tomography, *Nat. Geosci.* **5,** 493–498.

Lion Krischer
Heiner Igel
Department of Earth and Environmental Sciences
Ludwig-Maximilians University
Theresienstrasse 41
Munich 80333
Germany
krischer@geophysik.uni-muenchen.de

Andreas Fichtner
Saule Zukauskaite
Department of Earth Sciences
ETH Zürich
Sonneggstrasse 5
8092 Zürich, Switzerland

Published Online 3 June 2015