INVETA

Separation of Scattering and Intrinsic Attenuation by Envelope Inversion

Version 1.4

User's Manual

Walter Imperatori
Swiss Seismological Service, ETH Zurich
walter.imperatori@sed.ethz.ch
https://github.com/flurinursli/inveta

Contents

1 Introduction

InvEta is a numerical package to retrieve scattering and intrinsic attenuation parameters (typically abbreviated as η_s and η_i, respectively) from the inversion of recorded ground motion envelopes. It relies on the Radiative Transfer Theory (RTT) to compute one-component envelopes generated by an isotropic impulsive source embedded in an infinite three-dimensional medium characterized by homogeneous background velocity and uniformly distributed scattering elements.
Although envelopes of acoustic and elastic waves can be simulated, in the former case both isotropic and anisotropic scattering can be considered, while in the latter only isotropic scattering is allowed.
The search through the parameters space is guided by the NA multidimensional algorithm to minimize the computational time. *InvEta* can operate on each recorded ground motion separately or on all the recordings for each receiver simultaneously. Moreover, it is also capable of event-based inversions, where all absorptionrecorded waveforms referring to the same event across all receivers are considered. The code is fully parallelized, leveraging on MPI and OpenMP directives, and can be run on desktop machines and supercomputers as well.
InvEta shares some similarities with another package for envelope inversion called *Qopen* [1]. However the latter relies on a one-dimensional search algorithm and implements only the approximated solution provided by Paasschens for 3D acoustic isotropic scattering. On the other hand, *InvEta* does not provide any estimate of source and site amplification energy factors.
Finally it should be noted that the separation of scattering and intrinsic attenuation parameters has been described in a large number of papers in the past few decades (see [2] for a comprehensive review). The vast majority of these attempts relied on the multiple lapse time window analysis (MLTWA), a time-consuming technique based on the coda normalization method.

1.1 Modeling energy densities and the inversion problem

In this section we provide a synthetic description of the theory behind *InvEta*. Users are recommended to read this section carefully in order to understand how to correctly modify the input parameters.
For a given frequency band Δf, the energy density envelope due to an impulsive point source recorded at distance r can be modeled as:

$$E_{syn}(t, r) = A_0 G(t, r, \mathbf{g}) e^{-bt} \tag{1.1}$$

where G is the Green's function accounting for direct and scattered waves, $\mathbf{g}$ is the vector of scattering parameters and A_0 is a generic scalar factor. The exponential term describes the effect of the intrinsic attenuation parameter b.
The calculation of G is based on the RTT and the relevant equations are described in detail in Chapter 8 of [2]. In the acoustic approximation (S-waves only), *InvEta* can model both multiple isotropic and forward scattering. In this case the scattering parameters vector takes the following form:

$$\mathbf{g} = [\eta_{ss}, \nu, \kappa] \tag{1.2}$$

where η_{ss} is the S-to-S scattering coefficient, ν is a factor indicating isotropic ($\nu = 0$) or forward ($\nu > 0$) scattering (see also Section 1.2) and κ is called Hurst exponent. In general the larger ν the stronger the forward scattering. Note that ν, together with the statistical characteristic of the medium (e.g., Von Karman, Gaussian autocorrelation), controls the appearance of the axial symmetric scattering pattern function. The Hurst exponent, typically in the range $(0, 1)$, describes the roughness of Von Karman media and it also contributes to the shape of the scattering pattern function (although to a lesser extent). On the other hand isotropic scattering ($\nu = 0$) is not influenced by neither the autocorrelation function nor the Hurst exponent.
In the elastic approximation, where both P- and S-wave are emitted by the source, the same vector is given by:

$$\mathbf{g} = [\eta_{pp}, \eta_{ps}, \eta_{sp}, \eta_{ss}] \tag{1.3}$$

Here each η_{xy} represents the x-to-y scattering coefficient. In practice, we can shrink slightly the parameters space when considering Poisson solids: in this case $\eta_{sp} = \eta_{ps}/6$. Moreover, we take an approach similar to [3] and express $\mathbf{g}$ in terms of coefficient ratios, i.e.:

$$\mathbf{g} = [\eta_{ss}, \eta_{ss}/\eta_{pp}, \eta_{ps}/\eta_{pp}] \tag{1.4}$$

since their range can be constrained once the statistical characteristic of the medium is assumed. The scattering coefficient η is also frequently expressed in terms of a quality factor Q, for instance:

$$\eta_{ss} = \frac{2\pi f_0}{\beta_0 Q_{ss}} \tag{1.5}$$

where β_0 is the background velocity of the medium and f_0 represents the central frequency in Δf. Similar expressions hold for the other coefficients. For the intrinsic attenuation of S-waves, we can write:

$$\eta_i = \frac{2\pi f_0}{\beta_0 Q_i} = \frac{b}{\beta_0} \tag{1.6}$$

where b is the intrinsic attenuation parameter in Eq. 1.1.
Modeled and observed envelopes are compared at several locations. The latter are obtained by first applying a band-pass filter having central frequency f_0 and band-width Δf to the velocity seismograms. Then the MS envelope $\langle \dot{u}(t)^2 \rangle$, whose amplitude is linearly proportional to the observed energy density E_{obs}, is given by:

$$\langle \dot{u}(t)^2 \rangle = \frac{1}{2} \sum_{i=1}^{3} (\dot{u}(t)^2 + H(\dot{u}(t))^2) \tag{1.7}$$

where H indicates the Hilbert transform [see 2, Eq. 2.27]. Note that stacking occurs not only to enhance the signal-to-noise ratio, but also to reduce double-couple radiation pattern effects.
During the inversion we minimize the misfit ϵ^2 between the observed (E_{obs}) and predicted (E_{syn}) energy densities, expressed as:

$$\epsilon^2 = \sum_{j=1}^{m} \sum_{i=1}^{n(j)} (\ln(E_{obs,ij}) - \ln(E_{syn,ij}))^2 \tag{1.8}$$

where m is the number of observations (either recordings for a specific receiver or receivers that recorded the same event) and $n(j)$ the number of time samples for the j-th observation. Using Eq. 1.1, at each optimization step (that is, for each proposed scattering parameter vector $\mathbf{g}$), the following linear system

$$\ln(E_{obs,ij}) - \ln(G(t_{ij}, r_j, \mathbf{g})) = \ln(A_{0,j}) - b t_{ij} \tag{1.9}$$

is solved for b and $A_{0,j}$. The system, typically overdetermined since there are $\sum_{j=1}^{m} n(j)$ equations and $m + 1$ unknowns, is solved using a (weighted) least-squares approach. Once obtained, b and $A_{0,j}$ are plugged into Eq. 1.1 and Eq. 1.8 can be then evaluated. It should be noted that scalar factors $A_{0,j}$ incorporate source, site and free-surface effects.
The search through the parameters space (described by $\mathbf{g}$) is controlled by the NA multidimensional algorithm. The latter is fundamentally controlled by two parameters: the number of models generated at each iteration and the number of Voronoi cells resampled at each iteration, as these determine whether the search is more explorative or exploitative. For further details see Section 3.1.

Although the Green's function in Eq. 1.1 assumes an impulsive source, in the actual calculations it is replaced by a Gaussian pulse as in [3], whose duration is set to 0.25s. This value was chosen to minimize instability rising during numerical integration without affecting the inversion results.
Because of peak broadening, finite bandwidth and other effects, the observed envelope around the direct (P- and) S-wave has generally a different shape than the modeled one, especially when the isotropic scattering model is adopted. To solve this problem we follow [1] and define two time-windows: one for the direct wave(s) and one for the coda. When comparing envelopes, only the mean value inside each direct wave window is considered. During the linear regression (Eq. 1.9), these values can be weighted based on the number of samples in the direct-wave window(s). Experience has shown that such weighting should be used for isotropic scattering only, as forward scattering can reproduce much better the observed energy density around the direct S-wave.
It is worth closing this section by mentioning the main assumptions of the scattering model implemented in *InvEta*. From a practical perspective, these are the presence of an unbounded medium with constant background velocity, an homogeneous distribution of point-like scattering elements and an impulsive, isotropic point source. Another important assumption is related to the Born approximation, requiring conditions $\epsilon^2 \ll 1$ and $\epsilon^2 a^2 k_0^2 \ll 1$ to hold (here ϵ is the root-mean-square of the fractional fluctuation of S-wave velocity, a the correlation length characterizing the medium and k_0 the wavenumber at f_0). Nonetheless, it has been suggested [e.g., 4] that the RTT may be successfully applied also outside these ranges.
Finally it should be noted that body waves only are modeled. Scattered surface waves and body wave to surface wave scattering conversions are completely neglected. Usually this does not represent a problem, as the latter are expected to be relevant only at low frequency and at long lapse times. However, as noted by [5], scattered surface waves may dominate the seismic coda even at early arrival times for shallow sources if efficient body wave to surface wave conversion mechanisms exist in the shallow crust.

1.2 Isotropic and forward scattering

Here we briefly clarify a few key terms concerning the isotropic and forward scattering models. Following [2], in the framework of the RTT with the Born approximation, the scattering coefficient η (sometimes indicated also as g) at wavenumber k_0 and scattering angle ψ is defined as:

$$\eta(k_0, \psi) = \frac{k_0^4}{\pi} P(2k_0 \sin \frac{\psi}{2}) \tag{1.10}$$

where P is the power spectral density characterizing the medium. ψ is the angle between the propagation direction of a scalar wave impinging on a scattering element and the propagation direction of the resulting scattered wave (see also Figure 4.1 in [2]). When the random medium is statistically homogeneous and isotropic, the scattering coefficient

is axially symmetric with respect to the incident direction. Given the length scale of the random inhomogeneity (correlation length) a, scattering occurs mostly in the forward direction when $ak_0 \gg 1$ ($\lambda_0 \ll a$). By introducing the scattering pattern function $\zeta(\psi)$, we can write

$$\eta(k_0, \psi) = \eta_0(k_0)\zeta(\psi) \tag{1.11}$$

where

$$\eta_0(k_0) = \frac{1}{4\pi} \oint \eta(k_0, \psi) d\Omega \tag{1.12}$$

(integration is over the solid angle). $\eta_0(k_0)$ is called total scattering coefficient. When $ak_0 \ll 1$ ($\lambda_0 \gg a$) energy is scattered more uniformly around the scattering element, becoming wide-angle scattering: in the limit of isotropic scattering (i.e. $\zeta(\psi) = 1$), $\eta_0(k_0)$ coincides with $\eta(k_0, \psi)$. See Figure 1.1 for some sample scattering pattern functions. **Note that η_0 is the quantity appearing in the scattering parameters vector g introduced in Section 1.1**. The other parameter, ν, is equal to $a^2k_0^2$ for both Von Karman and Gaussian media and it controls the scattering pattern function ζ (as mentioned, the latter is a constant for $\nu = 0$).

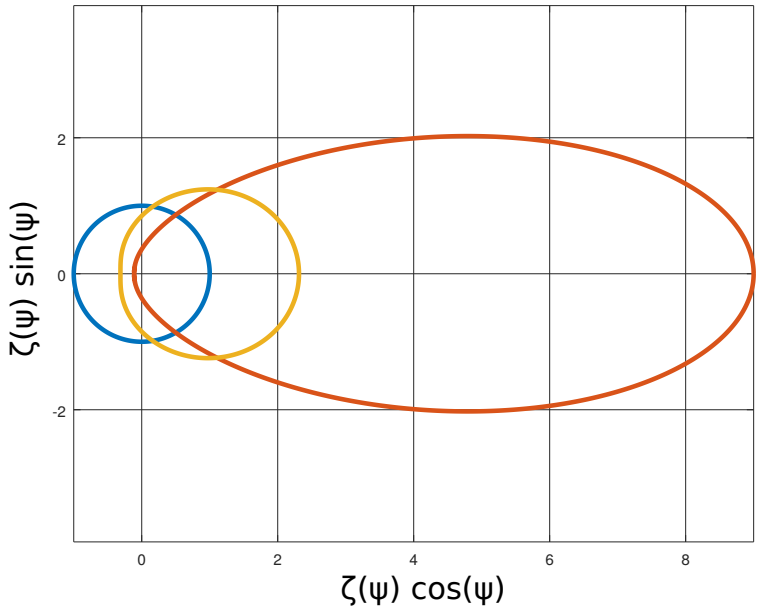

Figure 1.1: Sample scattering pattern functions for $\nu = 0$ (blue) and $\nu = 2$ (orange: gaussian media; red: exponential (Hurst = 0.5) media). While 12 spherical harmonics are required to reproduce the scattering pattern function for exponential media, only 4 are necessary for the gaussian media (despite identical ν values).

Another quantity of interest is the transport (momentum transfer) scattering coefficient

η_m (sometimes denoted as g_m or g^*):

$$\eta_m(k_0) = \frac{1}{4\pi} \oint \eta(k_0, \psi)(1 - \cos\psi) d\Omega = \frac{\eta_0(k_0)}{4\pi} \oint \zeta(\psi)(1 - \cos\psi) d\Omega \tag{1.13}$$

and it can be interpreted as the propagation distance required for a wave to lose memory of its initial direction. In other words, it describes the behavior of energy envelopes after many scattering events where envelopes are identical to solutions of the diffusion equation. From Eq. 1.13 it is easy to verify that η_m and η_0 coincide for isotropic scattering. The transport scattering coefficient is frequently replaced by the transport mean free path l_m that, in the acoustic case, is simply its inverse.
To conclude this section, we recall that *InvEta* can model both isotropic and forward scattering for S-waves, while isotropic scattering only for both P- and S-wave. Backscattering ($\nu < 0$), although formally acceptable, can quickly lead to numerical instability and is therefore not allowed in the program.

1.3 Practical considerations

The assumption of isotropic scattering is generally not valid for real media. Therefore, given the definitions introduced in Section 1.1 and 1.2, it may look tempting to search the parameters space for η_0, ν and possibly κ based on the acoustic approximation and then estimate the correlation length from ν. A similar approach, for instance, was followed in [6] and [7].
However, as pointed out by [8] and [9], there is an important parameters trade-off between fluctuation strength (ϵ) and correlation length (a). This is clearly visible in Figure 1.2, where misfit values for different $[\epsilon, a]$ pairs are shown. Note that horizontal lines parallel to the ϵ-axis would correspond to different ν values and that, along every line, η_0 increases from left to right. This figure evidences that several $[\epsilon, a]$ pairs (or, equivalently, $[\nu, \eta_0]$ pairs) fit the data equally well. The only parameter that can be constrained without too much uncertainty is the transport mean free path (represented by colored lines in Figure 1.2). It is also worth noting that wide-angle scattering ($\nu < 1$) is associated to higher misfit.
The same studies have also shown that the Hurst exponent κ does not influence significantly the appearance of the envelope, particularly in the coda. Moreover, according to [9], in the acoustic approximation the following equation holds for Von Karman media:

$$l_m \propto f^{2\kappa - 1} \tag{1.14}$$

where l_m is the transport scattering mean free path and $0 < \kappa < 0.5$.
Although forward scattering best explains the observed data, [7] and [8] have demonstrated that the η_0 retrieved from inversions based on the (acoustic) isotropic scattering model can be interpreted as η_m, although with larger uncertainty. These observations suggest that the inversion approach proposed by [6] and [7] is highly questionable and

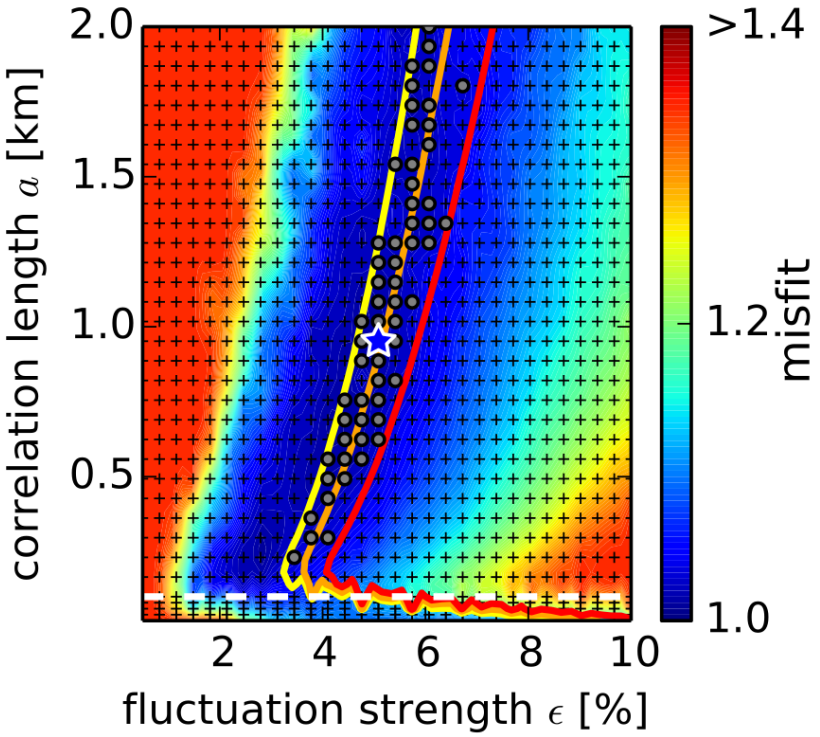

Figure 1.2: Misfit value between recorded and modeled envelopes for different pairs of correlation length and fluctuation strength referred to a single event at a given central frequency. Star indicates minimum misfit, dots are pairs with very similar misfits. Colored lines are isolines of three different transport mean free paths. Dashed line corresponds to $\nu = 1$ and separates wide-angle scattering from forward scattering. Modified from [8].

should be avoided.

To summarize, for the acoustic approximation we can conclude that:

- It is not possible to resolve ν and η_0, since different pairs lead to very similar misfit levels
- For the same reason, it is not possible to retrieve the correlation length
- The transport scattering coefficient η_m is the only parameter that can be reliably constrained
- If computational time is a problem, it may be advantageous to fix the Hurst exponent (to, say, 0.25) during the inversion and then estimate its value from Eq. 1.14
- Estimates of η_m in the forward scattering regime ($\nu > 1$) are more reliable. Setting ν to 6 is likely a good compromise between accuracy and computational cost.

- Estimates of η_m based on (acoustic) isotropic scattering ($\nu = 0$) are less accurate but not too far from those where $\nu > 1$

In practice, this means that for the acoustic RTT the scattering parameters vector in Eq. 1.2 can be reduced to

$$\mathbf{g} = [\eta_{ss}, \kappa] \tag{1.15}$$

or

$$\mathbf{g} = [\eta_{ss}] \tag{1.16}$$

after setting ν and/or κ to some constant value based on the above recommendations. In *InvEta* all three forms are accepted. For the elastic RTT, the scattering parameters vector remains unchanged.

2 Installation

2.1 Introduction

In this chapter we describe how to install *InvEta* on Unix/Linux systems and the necessary system requirements. The code can run on desktop machines and supercomputers as well.

2.2 Download

InvEta is freely available at `https://github.com/flurinursli/inveta`. The downloaded zip file contains the following items:

- `lib/`: source files to build an internal set of libraries needed by *InvEta*
- `main/`: source files for the main program
- `postprocessing/`: Matlab/Octave scripts to visualize parameters space search and best fitting envelopes
- `input.par`: sample input file
- `debug.par`: sample input file for debugging mode
- `License.txt`: the GNU GPL license
- `Makefile`: the makefile itself
- `Makefile.inc`: file with user-defined input parameters for `make`
- `Manual.pdf`: this manual

2.3 Hardware and Memory Requirements

InvEta has very modest memory requirements. In general, inversions based on the isotropic scattering model (either acoustic or elastic) do not require relevant computational power, unless users can exploit a large number of observations. In any case, most multi-core workstations and small clusters should suffice to obtain solutions in acceptable time.

More consistent resources are recommended when the forward scattering model is employed instead. In this case the program performs many evaluations of the Gauss hypergeometric function ${}_2F_1$, whose cost is proportional to parameter ν (see Section 1.2): the related scattering pattern function is expanded by means of l spherical harmonics and the computing time scales roughly as l^2. Inversions requiring the forward scattering model should run on hundreds of CPU cores.

2.4 Compilers

InvEta can be compiled using any compiler suite supporting FORTRAN (standard 2008), provided the required libraries are already installed (see Section 2.5). To date, we have successfully built *InvEta* using the following compilers:

- **GCC**, version 7.4, 9.2, 10.1, `https://gcc.gnu.org/`
- **Intel**, version 19.1, `https://software.intel.com/content/www/us/en/develop/tools/compilers.html`

GCC can be obtained for free, while Intel compilers require a commercial license. Other suites may be used after the `Makefile.inc` file is modified accordingly (see Section 2.6).

2.5 External Libraries

In order to build *InvEta*, the following external libraries must be present in your system:

- **MPI** (the Message Passing Interface), standard 3 or higher, `https://www.mpi-forum.org/`
- **FFTW** (the Fastest Fourier Transform in the West), version 3.0 or higher (version 3.3.8 recommended), `http://www.fftw.org/`
- **GSL** (GNU Scientific Library), version 2.6 or higher, `https://www.gnu.org/software/gsl/`

MPI provides support for message passing on parallel machines and it is required even for building *InvEta* on single core systems. A popular open-source implementation is OpenMPI (`https://www.open-mpi.org/`). FFTW is a widely distributed collection of fast routines to compute the discrete Fourier transform (DFT). GSL is a multi-purpose, free C library: in our program it is used to evaluate Wigner-3j coefficients and associated Legendre polynomials.
Instructions on how to install these libraries can be found on the respective websites, although they are typically found pre-installed in many systems.

Variable	Meaning	Accepted values
`FC`	MPI Fortran wrapper.	(any)
`OPT`	Compiler optimization flags.	(any)
`OMP`	Compiler-dependent flag to enable OpenMP parallelism.	(any)
`ERROR_TRAP`	Enable error checking at run-time.	`y/n`
`PERF`	Enable detailed performance measurement.	`y/n`
`MKL`	Enable Intel MKL version of FFT and LAPACK routines.	`y/n`
`INCL_FFTW`	Path to FFTW' *include* folder.	(any)
`LINK_FLAGS`	Flags to link against BLAS, LAPACK and FFTW libraries.	(any)

Table 2.1: Environmental variables in `Makefile.inc` and their meaning.

2.6 How to compile

InvEta can be built using **make**. The process is controlled by environmental variables defined in the provided `Makefile.inc` and summarized in Table 2.1.
The `Makefile.inc` already provides good optimization flags for GCC and Intel compilers. These can be further fine-tuned or changed to allow the use of other compiler suites. Enabling the OpenMP parallelism is recommended on most systems, especially in case of forward scattering modeling. Note that `LINK_FLAGS` is compiler- and system-dependent. In general, we advice to disable the `PERF` flag when building *InvEta* as it introduces internal communication overhead. On the other hand, `ERROR_TRAP` should be always enabled unless users are fully confident on their choice of the input parameters (see Section 3.4).

Once all flags are duly set, to build *InvEta* type:

```
make
```

The resulting executable, named `inveta.exe`, will be located in the current folder. To build *InvEta* in debugging mode (see Section 3.4 and 3.6 to understand what this means), type:

```
make debug
```

This will create an executable called `debug.exe`, whose use is clarified in Section 3.6.

3 Using InvEta

In this chapter we explain how to use *InvEta* and illustrate the output files it produces. The program is distributed with an heavily commented sample input file, named `input.par`. Here below we describe more in detail the meaning of each entry. A few scripts to read the output files can be found in Section 3.5.

3.1 The input file

Users control the behavior of *InvEta* by setting a series of parameters in the input file, which can have any name. Tables 3.1 through 3.4 list all the parameters, including a brief description and allowed values. **Note that the parsing routine is case-sensitive**. Parameters can be given in any order, also in the same line (use commas for spacing), but those listed in Table 3.2, 3.3 and 3.4 must be introduced by specific upper-case keywords. In any case, **lines in the input file must be terminated by a comma**. Comments are introduced by the `'#'` symbol.
As explained in Section 1.1, before an inversion *InvEta* filters data with a zero-phase 4 poles Butterworth band-pass filter. Filtering can occur at different frequency bands Δf, typically in half-octave bands to best preserve the signal pulse width. For instance `Bands = {4, 8; 8, 16}` instructs the code to operate two distinct inversions on time-series filtered in the range 4-8 and 8-16 Hz, respectively. Users can specify as many frequency bands as desired.
`Mode` controls how data are inverted. When set to 0, recordings satisfying the coda duration requirements of Table 3.3 will be inverted for all receivers one by one ($m = 1$ in Eq. 1.8). This approach is in general not recommended, unless a careful statistical analysis of the results is performed. The main reason is related to the isotropic source radiation assumption in the scattering model: as shown by [2], envelopes generated by isotropic and double-couple sources can be quite different between the S-wave arrival time and twice this value. If `Mode` is set to 1, *InvEta* inverts all recordings available for each receiver, one receiver after the other (i.e. m in Eq. 1.8 will be a receiver-dependent number). This approach in principle compensates for the non-isotropic source radiation but hinder any statistical analysis of the results. Finally, when `Mode = 2` all recordings referred to the same earthquake are inverted jointly among all receivers (i.e. m in Eq. 1.8 will be an earthquake-dependent number). *InvEta* will scan all folders to group recordings by event automatically. In this case we can have several estimates of the scattering parameters for a given location, depending on the number of earthquakes recorded, thus enabling a statistical analysis. Note that for `Mode` 1 and 2 an inversion

Variable	Description	Value
`Bands`	Lower/upper corner frequencies for bandpass filtering	> 0
`EtaSS`	S-to-S scattering coefficient	> 0
`Nu`[a]	Factor controlling the scattering pattern function	≥ 0
`Hurst`[a,c]	Hurst exponent for autocorrelation function	$(0, 1)$
`EtaSS/PP`[b]	Ratio between S-to-S and P-to-P scattering coefficients	> 0
`EtaPS/PP`[b]	Ratio between P-to-S and P-to-P scattering coefficients	> 0
`acf`[a]	Autocorrelation function	'vk'/'gs'
`Beta`	Assumed S-wave velocity	> 0
`Mode`	Set inversion mode	$[0, 2]$
`Threshold`[d]	Minimum number of observations required	> 0
`Weight`	Enable/disable weighted linear regression	'y'/'n'
`InitialModels`	Number of models sampled during first NA iteration	> 0
`Models`	Number of models sample during subsequent iterations	> 0
`Resampled`	Number of Voronoi cells resampled at each iteration	> 0
`Iterations`	Total number of iterations	> 0
`Seed`	Seed used to initialize the random numbers generator	> 0

[a] Acoustic RTT only
[b] Elastic RTT only
[c] Required for Von Karman media
[d] Not necessary when `Mode`$= 0$

Table 3.1: Input parameters not introduced by any keyword.

will take place only if there are at least `Threshold` recordings/receivers available. In any case, these must also fulfill the coda duration requirements of Table 3.3.
Parameters `InitialModels`, `Models`, `Resampled` and `Iterations` control the NA search. Typically `Models` is of the order twice the number of elements in the scattering parameters vector `g` (see Section 1.1 and 1.3) and `Resampled` can be anywhere between 2 and half the value of `Models`. Large values of both parameters result in a more explorative search. In any case, the total numbers of models explored is given by `InitialModels` + `Models` · `Iterations`.

As explained in Section 1.1, fitting envelopes near direct-waves can be particularly diffi-

Variable	Description	Value
`Pwin`[a]	Window width (s) for direct P-wave averaging	≥ 0
`Swin`	Window width (s) for direct S-wave averaging	≥ 0
`Factor`[b]	Controls how much of a window extends after the direct wave	$[0, 100]$

[a] Elastic RTT only
[b] Optional parameter (default value is 100%)

Table 3.2: Input parameters introduced by keyword `DIRECT`.

cult given the assumptions of the scattering model implemented in *InvEta* and therefore two different windows can be specified. The averaging of direct waves is controlled by the parameters listed in Table 3.2 and these must be introduced by the keyword `DIRECT`. Setting `Pwin` and/or `Swin` to 0 implies no averaging (i.e. the algorithm will try to fit all samples around the direct wave). In our experience a window width of about 3 s seems to work well in most cases, at least for epicentral distances in the range 10 to 100 km. Parameters controlling the coda waves window are listed in Table 3.3 and must be introduced by the keyword `CODA`. In this case `Pwin` and `Swin` provide the minimum acceptable window width for coda waves. Note that the actual envelope segments entering the inversion procedure are determined by the combined use of the `DIRECT` and `CODA` parameters. As an example, let us consider the following set of commands for the acoustic RTT:

```
DIRECT Swin = 4
CODA Swin = 6, Tlim = 3
```

In this case *InvEta* will fit the average envelope amplitude computed in the range $[t_s, t_s + 4]$ and all the other time samples in the range $(t_s + 4, 3 \cdot t_s]$, where t_s is the receiver-dependent S-wave arrival time (referred to origin time). Recordings whose duration is less than $t_s + 4 + 6$ and samples beyond $3 \cdot t_s$ will be ignored.

Variable	Description	Value
`Pwin`[b]	Minimum coda window width (s) after direct P-wave	> 0
`Swin`	Minimum coda window width (s) after direct S-wave	> 0
`Tlim`	Fit envelopes up to Tlim times the direct S-wave arrival time	> 0

[b] Elastic RTT only

Table 3.3: Input parameters introduced by keyword `CODA`.

Similarly, let us consider the following for the elastic case:

```
DIRECT Pwin = 2, Swin = 4
CODA Pwin = 5, Swin = 6, Tlim = 3
```

Here the program will consider an average value computed in the range $[t_p, t_p+2]$ and all samples between $(t_p + 2, t_s)$, beside those mentioned for the acoustic case. Once again, recordings where $t_s - (t_p + 2)$ is less than 5 or whose duration is less than $t_s + 4 + 6$ will be ignored. `Tlim` can be used to exclude parts of the late coda from the inversion.

For every station to be considered in the inversion, users must provide a file containing basic information (including P- and S-wave arrival times) and the path to the folder containing time-series. This can be achieved by means of the parameters shown in Table 3.4 and introduced by keyword `REC`. Users can refer to a single receiver with the `REC` command, but this can be repeated as many times as necessary. The station-specific file provides information for each earthquake recorded, it can have any name but must follow the format below:

```
# HEADER
EVENT_1 STATION NETWORK CHANNEL P_1 S_1
EVENT_2 STATION NETWORK CHANNEL P_2 S_2
⋮
EVENT_n STATION NETWORK CHANNEL P_n S_n
```

where `EVENT, STATION, NETWORK` and `CHANNEL` are the earthquake, station, network and channel (e.g., SEED convention) identification strings. `P` and `S` are the direct P- and S-wave arrival times, respectively. All fields must be tab-separated. For instance:

```
# EVENT STATION NETWORK CHANNEL P S
2013-01-02 19:55:33.140000 BERGE CH HH 13.1500 22.1350
2013-06-08 03:06:04.910000 BERGE CH HH 6.0300 9.7650
⋮
2013-11-03 21:42:43.955000 BERGE CH HH 10.9150 18.6800
```

Variable	Description	Value
`Folder`	Path to folder with recordings	any
`File`	Path to file with station information	any

Table 3.4: Input parameters introduced by keyword `REC`.

These information are also used to build the name of the files containing the time-series. In the case above, *InvEta* will expect to find the following files inside `Folder`:

```
20130102195533_CH_BERGE_HHE.mseed.ascii
20130102195533_CH_BERGE_HHN.mseed.ascii
20130102195533_CH_BERGE_HHZ.mseed.ascii
20130608030604_CH_BERGE_HHE.mseed.ascii
⋮
```

Notice that the program automatically adds the three traditional orientations `E, N, Z`. Time-series must be provided in velocity (arbitrary units) and each component of motion stored in a multi-column ASCII format (SLIST) supported by, e.g., ObsPy (`https://docs.obspy.org/tutorial/code_snippets/export_seismograms_to_ascii.html`).

3.2 Execution

InvEta is fully parallelized, leveraging on MPI and OpenMP directives. Parallelization is particularly useful for the forward scattering model. Typically users must indicate the total amount of processes (MPI tasks) and the number of threads (OpenMP execution units) available to each process when submitting their job. It is important to note that there is an upper bound n_t to the number of exploitable MPI tasks for a given set-up:

$$n_t = n_s \cdot n_o \cdot 512/n_p \tag{3.1}$$

where n_s is the number of models to be explored at each iteration (i.e. `Models`), n_o the available observations (either recordings-per-receiver or receivers-per-earthquake) and n_p the OpenMP threads. Exceeding tasks will result in idle CPUs until n_o changes. Our recommendation is to set a relatively high value for n_p (such as 12, 16 or 24, depending on the underlying hardware characteristics) in order to minimize inter-node communications and load unbalancing for inversions relying on the forward scattering model.
To run *InvEta*, a typical command looks like:

```
mpirun -np 32 -x OMP_NUM_THREADS=12 -bind-to socket ./inveta.exe input.par
```

For instance, on a system featuring computing nodes equipped with 2 12-cores CPUs, the program would allocate 16 nodes.

3.3 Output

InvEta produces three kinds of output files: the best-fitting model parameters for each receiver and all frequency bands are written to files labeled

`bestpar_<STATION>.txt`

where `<STATION>` is a station-identifier (see also Section 3.1). Note that this file includes the central frequency f_0, the sought scattering coefficient vector $\mathbf{g}$ (as in Eq. 1.2 and 1.3), the intrinsic attenuation parameter b (Eq. 1.6) and, for the acoustic case, the transport scattering coefficient η_m (Eq. 1.13).
The explored parameters space at each frequency band Δf specified in the input file is stored to

`nasearch_<LF-HF>_<CODE>.txt`

where `<LF-HF>` indicates the lower- and upper-frequency delimiting the current frequency band, and `<CODE>` is either a station (`Mode = 1`) or earthquake (`Mode = 2`) identifier or both (`Mode = 0`). The file consists of four columns, three for the scattering parameters vector $\mathbf{g}$ and one for the misfit averaged over the number of samples used in the inversion. It can be used to verify, for instance, the misfit reduction at every iteration.
The last kind of output file contains both observed and best-fitting envelopes and is labeled

`bestfit_<LF-HF>_<STATION>_<EVENT>.txt`

Here `<STATION>` and `<EVENT>` are the receiver and earthquake identifier, respectively. The file consists of three columns, the first one representing the time vector and the other two the observed and synthetic envelope, respectively.

3.4 Error messages and Troubleshooting

During execution, *InvEta* echoes to standard output a series of messages (see also Section 3.6), including descriptions of any numerical error detected if instructed to do so (Section 2.6). Error detection degrades performance a bit but we believe that obtaining a correct solution is much more important than a small performance penalty.
The most common error is due to inaccurate numerical integration during envelope calculation: if the integrand presents important singularities, the resulting envelope may contain negative values. In this case *InvEta* will stop the execution and inform users about the error, including the parameters (e.g. η_0, t_{max}, etc.) that triggered it. This error typically occurs when the NA algorithm picks "extreme" parameters combinations (e.g. very small g_0 and ν at very short distance from the source) during the search. In our experience, a small reduction of the search range often solves the problem. However, a better approach to avoid this issue is to build *InvEta* in debugging mode (Section 2.6) and proceed as described in Section 3.6 to test arbitrary parameter combinations and quickly find a numerically stable search range. Note that *InvEta* requires a different

input file when built in debugging mode: the provided sample file `debug.par` needs few parameters and can be easily interpreted.
In some cases *InvEta* may report a warning message concerning possible loss of accuracy during integration: this is not a fatal error and it simply suggests that some envelopes may look noisy at large time-lapse (envelopes may oscillates around the true values). We believe that users should not worry too much in this case: oscillations are usually small and occurs mostly at small absolute amplitude levels, thus they will unlikely affect the inversion results.
The table below summarizes errors and warning users could face when using *InvEta* and possible solutions.

Error/Warning description	Solution
`Possible loss of accuracy detected`	No action required (see text)
`Could not expand scattering pattern function`	Raise parameter `ndegree` in m_rtt.f90
`Linear regression failed`	Check returned error code in LAPACK/MKL library documentation
`Detected negative values in synthetic envelope`	Check parameters range for NA search (see text)

Table 3.5: Errors and warning issued by *InvEta*.

3.5 Visualization

Folder `postprocessing/` contains two Matlab/Octave scripts, `space_search.m` and `fdisplay.m`, that can be used to visualize the explored parameters space and the best-fitting envelopes. The list of arguments is described in detail in the header of each script.

3.6 Sample application

In this section we report a complete sample inversion, showing the entire workflow of *InvEta*. This can help users to correctly setup the input file and avoid possible mistakes. Let us suppose we want to find the best fitting scattering parameters at a single receiver for which we have about 50 recordings for epicentral distances up to 100 km. Time-series have different length, depending on the adopted signal-to-noise ratio selection criteria. We also assume we have entered all the station-related information into a text file called `Picks.txt` as explained at the end of Section 3.1 and that this file is located under:

```
/home/user/picking/SEIS
```

All waveforms, stored in a series of SLIST files, can be found under:

```
/home/user/recordings/SEIS
```

We want to apply the forward scattering model in order to have more reliable estimates. Following the discussion in Section 1.3, we set ν to 6 and let η_0 vary. We also assume a Von Karman-type random medium characterized by an average background velocity of 3.5 km/s and let κ vary between 0.05 and 0.5 (larger values are usually considered unlikely).
To set the parameters range for the NA search as wide as possible, we use *InvEta* in debugging mode (Section 2.6). We notice that picked S-wave arrival times are in the range 5 s to 30 s and decide to limit the inversion of each observed envelope to a maximum of 4 times the arrival time (`Tlim = 4`, resulting in a maximum time between 20 s and 120 s). We compute synthetic envelopes for a few (η_0, κ) pairs at these minimum and maximum arrival times and quickly determine that stable calculations and plausible envelopes can be obtained for η_0 in the interval 0.005 to 0.2.
We choose to retrieve the scattering parameters at nine frequency bands. The resulting input file, which we label `sample.par`, looks as follows:

```
# set frequency bands
Bands = {2,4; 3,6; 4,8; 6,12; 8,16; 10,20; 12,24; 14,28; 16,32},
EtaSS = {0.005, 0.2},
Hurst = {0.01, 0.5},
# nu is set to constant!
Nu = {6, 6},
# other parameters related to medium
acf = 'vk', Beta = 3.5,
# invert all envelopes at once, choose ordinary regression
Mode = 1, Threshold = 1, Weight = 'n',
# average direct wave over a 3s-long time-window
DIRECT Swin = 3.,
# do not set minimum length criteria for coda
```

```
CODA Swin = 0., Tlim = 4.,
# set NA parameters
InitialModels=8, Models=4, Resample=2, Iterations=10, Seed=1754,
# now list receivers
REC Folder='/home/recordings/SEIS', File= '/home/picking/SEIS/Picks.txt',
```

The first frequency band is 2-4 Hz: lower bands should be avoided as the filter becomes too narrow. The first thing to check before running any inversion is whether the Born approximation holds ($\epsilon^2 a^2 k_0^2 \ll 1$, see Section 1.1). This is the case, since $a^2 k_0^2 = \nu = 6$ and ϵ is generally assumed to be 0.1 or less.
The inversion completes in about 2 hours on 24 nodes equipped with two 16 cores CPUs (we requested 48 MPI tasks, each spawning 16 threads). We double-check the log file to exclude any error in the input file or during execution.
The NA algorithm sampled 48 models for each frequency band, as shown in Figure 3.1.

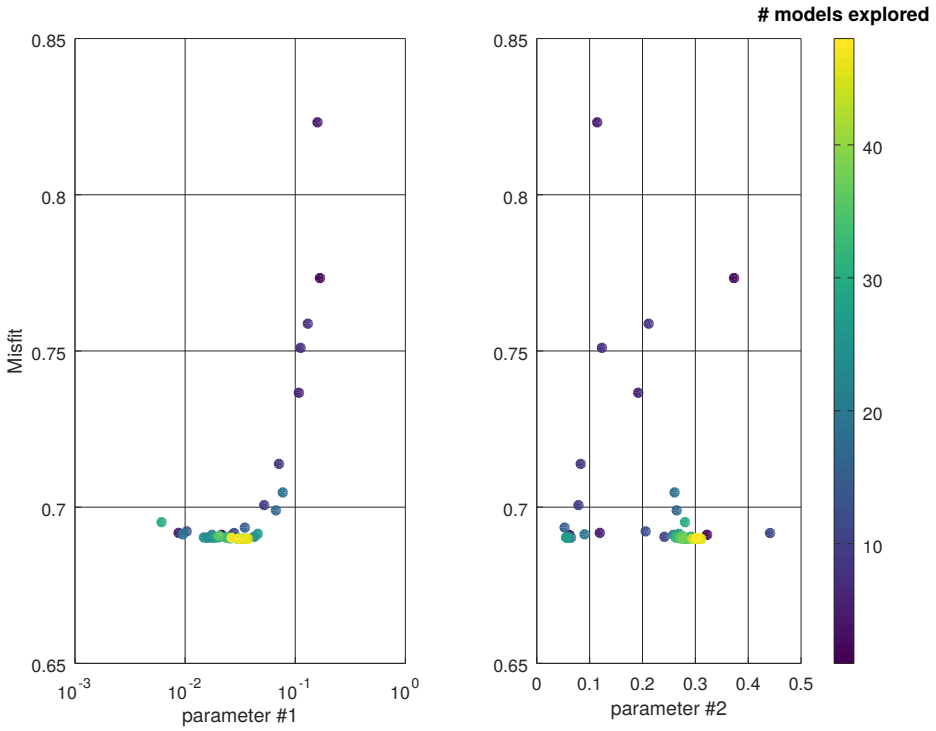

Figure 3.1: NA parameters space search for η_0 (left) and κ (right) in frequency range 8-16 Hz.

This figure indicates that there are many models (i.e. many (η_0, κ) pairs) able to provide similar misfit: in particular, κ seems to play a marginal role, as discussed in Section 1.3, suggesting that we could exclude it from the search. Also, we could probably decrease the number of iterations since the inversion converges fast: about 6-7 iterations would have been likely sufficient.

Final results are stored by *InvEta* in a file called `bestpar_SEIS.txt` and summarized in Table 3.6.

f_0	η_0	ν	κ	b	η_m
3.0	2.138-E02	6.0	0.382	5.089-E02	4.753-E03
4.5	1.391-E02	6.0	0.432	4.468-E02	2.934-E03
6.0	1.597-E02	6.0	0.391	4.482-E02	3.517-E03
9.0	2.502-E02	6.0	0.369	5.288-E02	5.637-E03
12.0	3.468-E02	6.0	0.311	6.223-E02	8.321-E03
15.0	2.169-E02	6.0	0.085	6.668-E02	6.619-E03
18.0	2.290-E02	6.0	0.068	7.034-E02	7.112-E03
21.0	2.392-E02	6.0	0.068	7.213-E02	7.430-E03
24.0	2.392-E02	6.0	0.067	7.284-E02	7.440-E03

Table 3.6: Final results from sample inversion.

In Figure 3.2 we compare some observed and best-fitting envelopes at two frequency ranges. Overall there is a very good correspondence except around the direct S-wave, as we should expect.

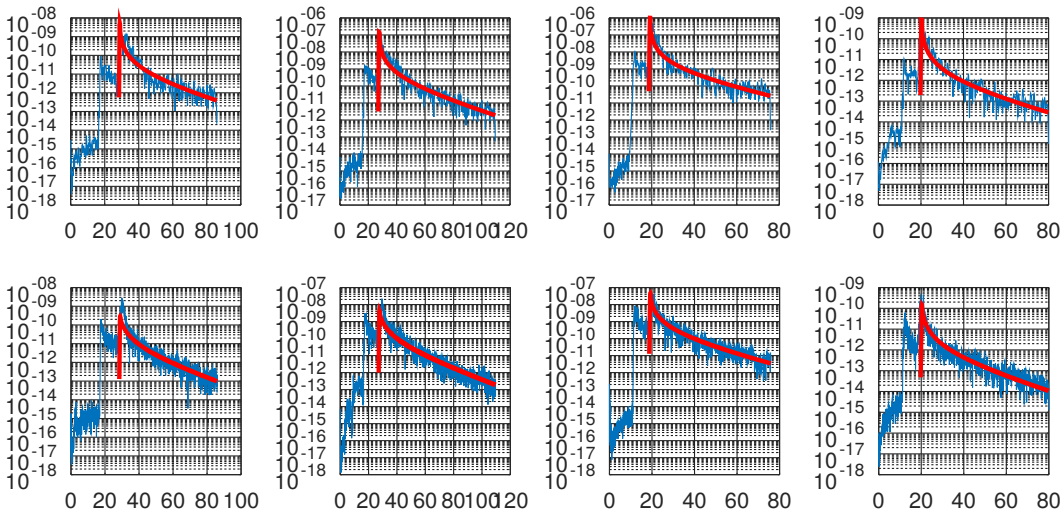

Figure 3.2: Comparison between observed (blue) and best-fitting envelopes (red) at 2-4 Hz (top) and 8-16 Hz (bottom).

Running the same inversion after fixing `Hurst` to 0.25 and reducing `Iterations` to 5 leads to similar transport coefficient scattering values and an estimated Hurst exponent of 0.15 based on Eq. 1.14. Note that this roughly corresponds to the average κ value of Table 3.6. Assuming isotropic scattering ($\nu = 0$), we obtain somewhat different estimates but not too distant from the previous ones; however, calculations completed in just 2 minutes! Note that in this case we enabled a weighted least-square regression (see discussion in Section 1.1). Figure 3.3 summarizes results from these three different inversions in terms of transport mean free path l_m, Q_s^{-1} and Q_i^{-1}, where for the latter two we used :

$$Q_s^{-1} = \frac{\beta_0 \eta_m}{2\pi f_0}, \qquad Q_i^{-1} = \frac{b}{2\pi f_0} \tag{3.2}$$

For our simple case study we can conclude that intrinsic attenuation (absorption) strongly dominates over scattering attenuation at all frequency bands and that the heterogeneity distribution can be reasonably represented by a Von Karman autocorrelation function with Hurst exponent around 0.15. Obviously these conclusions should not be over-interpreted since our analysis is based on observations at a single location.

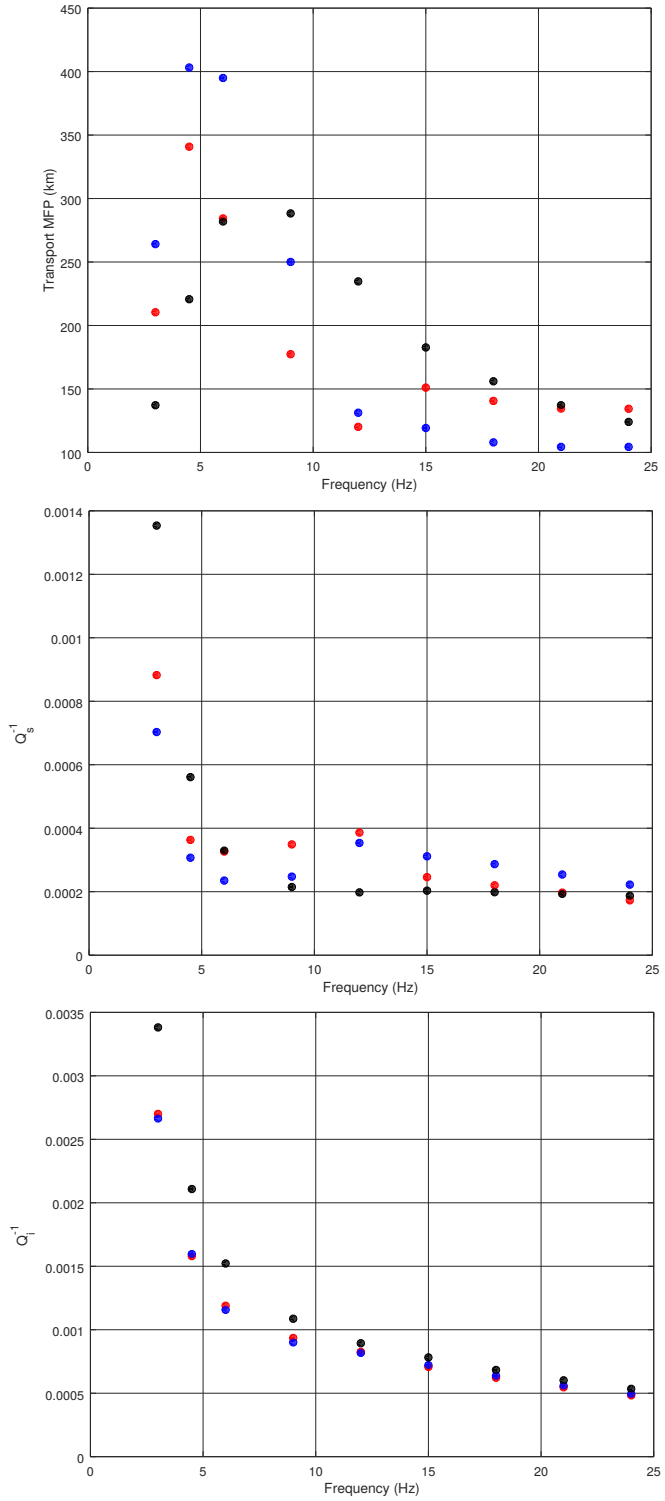

Figure 3.3: Transport mean free path (top), scattering attenuation (middle) and intrinsic attenuation (bottom) estimated from three different inversion approaches on the same dataset: forward scattering with free η_0 and κ (red), forward scattering with free η_0 and κ constrained to 0.25 (blue), isotropic scattering (black).

License

InvEta comes with ABSOLUTELY NO WARRANTY; released under GPL v3. This is free software, and you are welcome to redistribute it under certain conditions; see LICENSE.txt for more details.

References

[1] Tom Eulenfeld and Ulrich Wegler. "Measurement of intrinsic and scattering attenuation of shear waves in two sedimentary basins and comparison to crystalline sites in Germany". In: *Geophys. J. Int.* 205.2 (May 2016), pp. 744–757. ISSN: 0956-540X. DOI: 10.1093/gji/ggw035.

[2] Haruo Sato, Michael C. Fehler, and Takuto Maeda. *Seismic wave propagation and scattering in the heterogeneous earth: Second edition*. Vol. 9783642230. 2012, pp. 1–494. ISBN: 9783642230295. DOI: 10.1007/978-3-642-23029-5.

[3] Mare Yamamoto and Haruo Sato. "Multiple scattering and mode conversion revealed by an active seismic experiment at Asama volcano, Japan". In: *J. Geophys. Res.* 115.B7 (July 2010), B07304. ISSN: 0148-0227. DOI: 10.1029/2009JB007109.

[4] Haruo Sato. "Power spectra of random heterogeneities in the solid earth". In: *Solid Earth* 10.1 (Feb. 2019), pp. 275–292. ISSN: 18699529. DOI: 10.5194/se-10-275-2019.

[5] Yuehua Zeng. "Scattered Surface Wave Energy in the Seismic Coda". In: *Pure Appl. Geophys.* 163.2-3 (Mar. 2006), pp. 533–548. ISSN: 0033-4553. DOI: 10.1007/s00024-005-0025-7.

[6] Yueling Jing, Yuehua Zeng, and Gao Lin. "High-frequency seismogram envelope inversion using a multiple nonisotropic scattering model: Application to aftershocks of the 2008 wells earthquake". In: *Bull. Seismol. Soc. Am.* 104.2 (2014), pp. 823–839. ISSN: 19433573. DOI: 10.1785/0120120334.

[7] Yuehua Zeng. "Modeling of High-Frequency Seismic-Wave scattering and propagation using radiative transfer theory". In: *Bull. Seismol. Soc. Am.* 107.6 (2017), pp. 2917–2926. ISSN: 19433573. DOI: 10.1785/0120160241.

[8] Peter J. Gaebler, Tom Eulenfeld, and Ulrich Wegler. "Seismic scattering and absorption parameters in the W-Bohemia/Vogtland region from elastic and acoustic radiative transfer theory". In: *Geophys. J. Int.* 203.3 (Dec. 2015), pp. 1471–1481. ISSN: 0956-540X. DOI: 10.1093/gji/ggv393.

[9] Jens Przybilla, Ulrich Wegler, and Michael Korn. "Estimation of crustal scattering parameters with elastic radiative transfer theory". In: *Geophys. J. Int.* 178.2 (Jan. 2009), pp. 1105–1111. ISSN: 0956540X. DOI: 10.1111/j.1365-246X.2009.04204.x.

Index