
bh_tomo – A Matlab borehole georadar 2D and 3D tomography package

Bernard Giroux
Yiping Han

User's guide

Preliminary version

Contents

1	Introduction	7
2	Georadar tomography	9
2.1	General principles	9
2.2	Implementations	12
2.2.1	least-squares approach	12
2.2.2	Geostatistical inversion	13
2.3	Ray tracing	16
2.3.1	Discretization of the slowness model	16
2.3.2	Implementation	18
3	Software package description	21
3.1	Main user's interface	21
3.1.1	General processing flow	21
3.2	Database management	22
3.2.1	Boreholes	23
3.2.2	Multi-offset gathers	24
3.2.3	Panels	32
3.2.4	General processing flow	32
3.3	Traveltime processing	34
3.3.1	Manual picking	34
3.3.2	Semi-automatic picking	35
3.3.3	Automatic picking	39
3.4	Amplitudes processing	39
3.5	Fitting the model covariance	40
3.6	Inversion	41
3.7	Estimating the saturation of the magnetic nanofluid	43
3.7.1	Background	43
3.7.2	User interface	46
4	Example	49
A	Exchange files formats	55
A.1	Aramco's Magnetic NanoMappers data	55
A.1.1	Motivation for the format	55

A.2 Picked arrival times	57
A.3 Model constraints	57
A.4 Inversion results	58
B LSQR inversion parameters	59

List of Figures

2.1	Illustration of domain discretization	10
2.2	Different graph templates allowing for different angular coverage	17
2.3	Illustration of elementary cells with location of primary and secondary nodes. .	18
3.1	Interactive screen of <code>bh_tomo</code>	22
3.2	Interactive screen of <code>bh_tomo_db</code>	23
3.3	Add borehole.	23
3.4	Definitions of borehole coordinates.	24
3.5	3D view of the boreholes.	25
3.6	Borehole survey modes	25
3.7	Convention for transmitter (Tx) and receiver (Rx) coordinates.	26
3.8	Illustration of the two ways to input increasing distances between Tx and Rx antennas for air shots.	27
3.9	Import MOG dialog window.	27
3.10	Window showing raw MOG data.	28
3.11	Zero-offset profile display window.	28
3.12	Example of air shot data and statistics on picked traveltimes.	29
3.13	PSD estimates window	30
3.14	Ray coverage for a MOG dataset.	31
3.15	Utility window to prune MOG datasets.	31
3.16	Illustration of reference point and feed point of dipole antennas.	32
3.17	Interactive screen of <code>bh_tomo_grille</code>	33
3.18	Interactive screen of <code>bh_tomo_tt</code>	34
3.19	Example of first-cycle isolation. The original trace, in blue, has a dominant frequency of 177 MHz. This trace is crosscorrelated with a 177 MHz synthetic wavelet, shown in red. After the application of the time window, the scattered arrivals are eliminated, as shown in green.	36
3.20	Illustration of the effect of time rescaling for three synthetic traces having different dominant frequencies. The traces are aligned without (left) and with (right) time rescaling.	37
3.21	Interactive screen of <code>bh_tomo_pick</code>	38
3.22	Interactive screen of <code>bh_tomo_amp</code>	40
3.23	Interactive screen of <code>bh_tomo_fitCovar</code>	41

3.24	Interactive screen of <code>bh_tomo_inv</code> . Two velocity models obtained with the data presented in section 4 are displayed. The leftmost model is obtained by cokriging and the rightmost model is the simulation that best fits the data.	42
3.25	Accuracy of the saturation estimates with exact input parameters.	46
3.26	Accuracy of the saturation estimates for inaccurate velocity and attenuation input data.	47
4.1	Schematic cross-section of the surveyed structure interpreted from boring logs.	50
4.2	Ray coverage for the velocity model shown in Fig. 3.24. Grayed out rays were excluded for attenuation tomography.	51
4.3	Simulation results of the geostatistical inversion of amplitude data, amplitude ratio method. On the left: simulation that best fits the data. On the right: standard deviation inferred from the 128 simulations.	52
4.4	Simulation results of the geostatistical inversion of amplitude data, centroid frequency down-shift method. On the left: simulation that best fits the data. On the right: standard deviation inferred from the 128 simulations.	53
4.5	Maps of dielectric constant and effective electric conductivity inferred from the tomographic results. Boreholes 1, 2 and 3 are drawn in green.	54

Introduction

In the last two decades, borehole georadar has received increasing attention for mineral exploration as well as for hydrogeological and geotechnical studies (Olsson et al., 1992; Hubbard et al., 1997; Giroux et al., 2004; Tronicke et al., 2004). Information about the electromagnetic velocity and attenuation structures can be obtained from the inversion of cross hole georadar travel times and amplitudes, respectively. Combined inversions of georadar travel times and amplitudes may thus allow to separate the effects of dielectric permittivity and electric conductivity on electromagnetic wave propagation (Zhou and Fullagar, 2001). Therefore, to invert for both velocity and attenuation greatly enhance the understanding of the probed media.

Besides, unless straight rays are used, attenuation tomography requires that raypaths be known to render possible the integration of the unknown parameter along the wave trajectory. Therefore, velocity tomography must be performed prior to the inversion of the amplitude data. Once the velocity field is known, ray tracing allows to infer the curved raypaths associated to each measured travel time and amplitude. In fact, inversion of the travel times is non linear and is solved iteratively by updating the raypaths at each iteration. When the data to invert was acquired in many boreholes and in different configurations (cross-hole or vertical radar profiling), it becomes somewhat cumbersome to keep the travel time, amplitude and raypath data sets consistent. A motivation for the development of `bh_tomo` was to facilitate the management of these data.

Another important motivation for the development of the `bh_tomo` package is to make an implementation of the geostatistical inversion scheme of Gloaguen et al. (2005) available. This method offers many advantages over least-squares based algorithms. The main advantages of the new method are that it is self-regularized and requires less *a priori* information, it allows the identification of stable characteristics and uncertain features through stochastic simulations, and it allows the exact fitting of any linear constraints on the sought parameter. Also, the covariance model of the unknown field is determined by the method. This implies that the approach can be advantageous in applications where stochastic simulations are used without *a priori* information about the spatial properties of the studied parameters.

The Matlab programming environment, commercialized by The MathWorks, was selected for the development of `bh_tomo` for several reasons, namely because it combines extensive numeric computation libraries, a higher-level programming language as well as visualization capabilities and easy GUI development tools. Matlab has been used widely for geophysical applications (e.g. Beaty et al., 2002; Conroy and Radzevicius, 2003; Rucker and Ferre, 2004; Witten, 2002). In addition, Matlab programs are portable across platforms supported by The

MathWorks (Mac OS X, UNIX/Linux, Windows). `bh_tomo` has been developed as free software with a GNU license, allowing public access to the source code. The package may be used for any purpose including educational, research and commercial applications. Users are encouraged to modify and add to the existing package to suit their specific needs.

The main features of the package are:

- Ray-based approach
 - Travel time tomography;
 - Attenuation tomography based on first-cycle amplitude.
- Two types of ray-tracing routines
 - Straight rays in homogeneous velocity models;
 - Curved rays in heterogeneous models.
- Two-dimensional domains:
 - Arbitrary position of transmitters & receiver in the 2D plane (arbitrary well trajectory).
- Three-dimensional domains.
- Various data input format:
 - RAMAC (Mala Geosciences);
 - PulseEKKO (Sensors & Software);
 - SEG Y.
- Handle of time units and tomography on seismic data as well
- Two core inversion options:
 - Classic least-squares solver;
 - Geostatistical tomography and simulation.
- Elliptical anisotropy (with geostatistical approach) for 2D domains.
- Treatment of time-lapse data.
- Basic database functionality with graphical user interface.
- 2D and 3D visualisations.
- Basic interpretation module
- Released under the GNU General Public License, version 3.

The software package relies on a mini database and comprises interactive modules to manage, process and interpret the data. This user's guide is organised by module, but a first chapter about tomography is presented.

Georadar tomography

2.1 General principles

Tomographic georadar measurements are accomplished by using a transmitting antenna (Tx) located in a borehole to emit a high frequency electromagnetic pulse. The radiated pulse propagates within the medium with velocity v , and is eventually recorded at a receiving antenna (Rx) located in another hole a known distance away. The recording, referred to as a *trace*, is the voltage at the receiver as a function of time. Hence, the trace contains the transmitted wavelet arriving at time t , as well as other waves produced by eventual reflections or refractions of the emitted pulse. One approach to tomographic inversion is to retrieve the arrival time of the direct wave and use it to obtain the velocity between the transmitter and receiver by considering that the wave traveled along a raypath. Another approach is to consider all the information contained in the trace, i.e. the whole waveform, to reconstruct the physical properties distributions between the transmitter and receiver (Cai et al., 1996; Ernst et al., 2007). Waveform-based algorithms require extensive computational resources. Hence, in the development of `bh_tomo` only the former has been considered so far, i.e. the ray-based approach.

Following the theory of geometrical optics, to each transmitted wavelet can be associated a raypath l , which is a curve describing the trajectory of the wave between the Tx and Rx antennas. The travel time is related to the slowness s , i.e. the reciprocal of velocity, and raypath by

$$t = \int_{\text{ray}} s(l) dl. \quad (2.1)$$

During a survey, the recording process is repeated for various Tx and Rx positions along the holes, in order to have the raypaths cover the whole area between the two holes. After completion of the survey, the arrival times of the direct waves are picked from each trace, and these data are used to perform the travel time tomography calculations. For n_o traces, this data takes the form of a $n_o \times 1$ matrix called \mathbf{t} . The aim of the tomography is to obtain the velocity field, or its inverse the slowness, between the two boreholes. To perform the calculations, the slowness field must be discretized into n_p cells, yielding a matrix \mathbf{s} containing the $n_p \times 1$ unknown slownesses. This discretization implies that the raypaths are split into segments (see Fig. 2.1). Each segment has a length equal to the distance the ray travels in that cell. The travel time associated with a raypath becomes the sum of each segment times the slowness of the

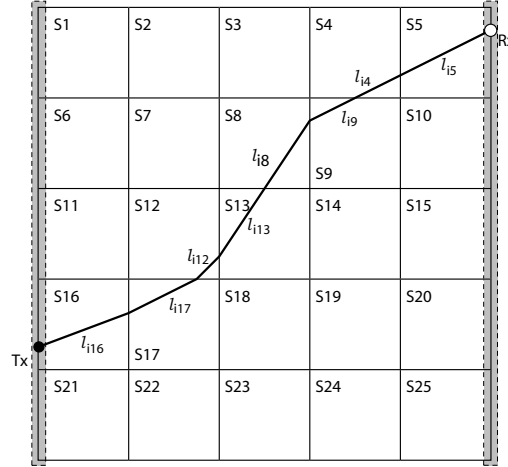


Figure 2.1: Illustration of domain discretization and consecutive raypath segmentation. Each segment has a length equal to the distance the ray travels in the cells it is traversing. The travel time is the sum of each segment times the slowness of the corresponding cell.

corresponding cell:

$$t = \sum_{i=1}^{\text{segments}} s_i l_i. \quad (2.2)$$

Gathering all measured travel times in vector \mathbf{d} and unknown slownesses in vector \mathbf{m} , a system of equations of the form

$$\mathbf{G}\mathbf{m} = \mathbf{d} \quad (2.3)$$

can be built, with \mathbf{G} being a sparse $n_o \times n_p$ matrix. The objective of the tomographic calculation is to solve this system for the slowness vector \mathbf{m} . Note that the problem is nonlinear because, by virtue of Fermat's principle, the raypaths are function of the slowness field. Fermat's principle states that the path of a ray between two points is the path that minimizes the travel time. Consequently, rays tend to "bend around" low velocity zones and "concentrate" in high velocity zones. This nonlinearity implies that the tomographic problem must be solved iteratively.

So far the focus has been on travel time tomography. Often, e.g. in environmental and hydrogeological applications, it is interesting to know the dielectric constant κ within the probed medium, because it is related to the volumetric water content (by definition $\kappa = \epsilon_e / \epsilon_0$ where ϵ_0 is the dielectric permittivity of free space)¹. Usually, it is common practice to perform travel time tomography alone to compute the dielectric constant κ . This approach is valid when electrical loss is low, i.e. when the condition $\frac{\sigma_e}{\omega \epsilon_e} \ll 1$ is satisfied. Here, ω is the angular frequency equal to $2\pi f$, f being the frequency of the radio wave, σ_e is the effective electrical conductivity and ϵ_e is the effective dielectric permittivity (for a discussion on effective parameters, see e.g. [Turner and Siggins \(1994\)](#)). Under the low loss condition, the velocity is

$$v = \frac{1}{\sqrt{\mu \epsilon_e}}. \quad (2.4)$$

Note that most earth materials are non magnetic, and in general their magnetic permeability can be assumed to be equal to the permeability of free space μ_0 . Therefore, the dielectric

¹Several empirical and theoretical models exist that relate κ to the volumetric water content. See [Huisman et al. \(2003\)](#) and [Sihvola \(2000\)](#) for reviews.

constant is obtained from

$$\kappa = \frac{1}{v^2 \mu_0 \epsilon_0}. \quad (2.5)$$

However, the above expression yields inaccurate results in media with high electric loss, i.e. if the electrical conductivity σ_e is close to, or higher than, the product of permittivity ϵ_e and ω (Giroux and Chouteau, 2010). In such circumstances, the dielectric constant must be evaluated from both the velocity and attenuation. With the attenuation α in Np/m, the dielectric constant is

$$\kappa = \frac{1}{\mu_0 \epsilon_0} \left[\frac{1}{v^2} - \left(\frac{\alpha}{\omega} \right)^2 \right]. \quad (2.6)$$

Although not rigorously exact in the context of a ray-based framework, Eq. (2.6) yields a more representative estimate of κ than Eq. (2.5). As mentioned in introduction, another reason to invert for attenuation is that its knowledge allows to infer the electrical conductivity structure and leads to an increased comprehension of the probed medium. The value of the effective conductivity is

$$\sigma_e = \frac{2\alpha}{\mu_0 v}. \quad (2.7)$$

Various methods exist to compute the attenuation within the medium. In `bh_tomo`, two methods are implemented: the amplitude ratio (Olsson et al., 1992) and the centroid frequency down shift method (Quan and Harris, 1997; Liu et al., 1998). In the former, the attenuation is obtained from the ratio of the emitted amplitude to the received amplitude. In this method, the received amplitude must be processed so that it can be linearly related to the attenuation. In short, the amplitude measured at the receiver A_m is an exponential decay of the initial amplitude at the source A_0 , times a correction term taking into account the radiation patterns of the antennas (Θ_{Tx} and Θ_{Rx}) and the geometrical spreading of the wave:

$$A_m = A_0 \exp \left(- \int_{\text{ray}} \alpha(l) dl \right) \frac{\Theta_{Tx} \Theta_{Rx}}{l}. \quad (2.8)$$

The terms A_0 , Θ_{Tx} and Θ_{Rx} cannot be easily known and in most cases are approximated. This makes amplitude tomography somewhat less robust than travel time tomography. Details and discussion on amplitude data reduction can be found in Peterson (2001) and Holliger et al. (2001). However, once A_0 , Θ_{Tx} and Θ_{Rx} are estimated, taking the natural logarithm of Eq. (2.8) leads to

$$\tau = \int_{\text{ray}} \alpha(l) dl, \quad (2.9)$$

where

$$\tau = \ln(A_0) + \ln(\Theta_{Tx} \Theta_{Rx}) - \ln(A_m l). \quad (2.10)$$

Equation (2.9) has the same form as Eq. (2.1). This means that the algorithms developed to solve Eq. (2.1) are applicable for Eq. (2.9).

Let us now consider the estimation of the attenuation from the frequency content of the data. Quan and Harris (1997) introduced the concepts and applied them for seismic tomography, and Liu et al. (1998) applied the method to GPR data. The technique is based on linear system theory which is used to describe the propagation of the EM wave in the ground. In such case, the amplitude spectrum at the receiver $R(f)$ is the product of the instrument/medium responses $G(f)H(f)$ with the amplitude spectrum of the incident wave $S(f)$, i.e. $R(f) =$

$G(f)H(f)S(f)$. The term $G(f)$ includes the instrument response, geometrical spreading, source-receiver coupling, radiation patterns, reflection/transmission coefficients and the phase accumulation caused by propagation; and $H(f)$ is the attenuation filter describing the variation of amplitude due to absorption in the probed medium. The attenuation filter has the form

$$H(f) = \exp \left(-f \int_0^r \alpha_0 ds \right), \quad (2.11)$$

where $\alpha_0 = \alpha / f$. The goal is to get an estimate for this parameter. The main assumption of the technique is that G must be independent of frequency. When this is the case, the integral of α_0 can be obtained directly from the logarithm of the ratio $S(f)/R(f)$. However, there are two difficulties to this procedure, the first being that $S(f)$ is generally unknown, and the second being that calculations based on individual frequencies are subject to instabilities due to poor signal-to-noise ratio. This is the reason why [Quan and Harris \(1997\)](#) introduced the centroid frequency, defined for a given signal $X(f)$ as

$$f_X = \frac{\int_0^\infty f X(f) df}{\int_0^\infty X(f) df}, \quad (2.12)$$

with the variance being

$$\sigma_X^2 = \frac{\int_0^\infty (f - f_X)^2 X(f) df}{\int_0^\infty X(f) df}. \quad (2.13)$$

They showed that the difference between the source and receiver centroid frequencies, f_S and f_R respectively, are related to the integral of the attenuation by

$$f_S - f_R = B^2 \int_0^r \alpha_0 ds, \quad (2.14)$$

where B^2 takes into account the type of source spectrum and variance of the centroid frequency. For Gaussian spectrum, B^2 is equal to the variance σ_S^2 . One problem with equation (2.14) is that both α_0 and f_S are unknown. In `bh_tomo`, f_S is found by linear regression and can also be manually edited. Also, by virtue of linear system theory, σ_S^2 and σ_R^2 are equal. σ_S^2 is therefore estimated from the values of σ_R^2 computed from the radargrams. Once f_S and σ_S^2 are deduced, we obtain, in discrete form for the i^{th} trace and for a Gaussian source spectrum:

$$\tau_i = \sum_j^N \alpha_j l_{ij}, \quad (2.15)$$

where in this case $\tau_i = f(f_S - f_R^i)/\sigma_S^2$. This linear system can be solved by using the same algorithm than the one developed for velocity analysis.

2.2 Implementations

2.2.1 least-squares approach

Until recently, variants of least-squares algorithms were almost the exclusive approach to solving the tomographic problem, which reduces to solving the system

$$\mathbf{G}\mathbf{m} = \mathbf{d} \quad (2.16)$$

where in our case \mathbf{G} represents a $n_o \times n_p$ matrix including the partial derivatives (ray segment lengths), \mathbf{m} the n_p slowness or attenuation values and \mathbf{d} the n_o travel time or amplitude data. Because this problem is inherently ill-posed, least-squares algorithms rely on so-called regularization techniques to stabilize the solution.

The least-squares approach implemented in `bh_tomo` is regularized using a smoothness constraint, i.e. the objective function to minimize has the form

$$\|\mathbf{G}\mathbf{m} - \mathbf{d}\|_2 + \lambda \|\mathbf{D}\mathbf{m}\|_2, \quad (2.17)$$

where \mathbf{D} is the second order spatial derivative matrix and λ is a Lagrange multiplier (Menke, 1989). The LSQR algorithm of Paige and Saunders (1982) has proved very effective to solve this type of system (Nolet, 1993). This algorithm can handle unsymmetric equations, and it is therefore possible to avoid the computation of the transpose matrices of \mathbf{G} and \mathbf{D} required in the classic least-squares solution. This property is used in `bh_tomo`, in which the system

$$\begin{bmatrix} \mathbf{G} \\ \lambda \mathbf{D} \end{bmatrix} \mathbf{m} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \end{bmatrix}, \quad (2.18)$$

is fed to a slightly modified version of the `lsqr.m` routine available at <http://www.stanford.edu/group/SOL/software/lsqr.html>.

Another constraint can be added to the system through the imposition of model parameter values at given locations. For a vector \mathbf{m}_c containing n_c of these values, this is done by incorporating a $n_c \times n_p$ matrix \mathbf{M} to the system, such that

$$\begin{bmatrix} \mathbf{G} \\ \lambda \mathbf{D} \\ \mathbf{M} \end{bmatrix} \mathbf{m} = \begin{bmatrix} \mathbf{d} \\ \mathbf{0} \\ \mathbf{m}_c \end{bmatrix}. \quad (2.19)$$

For each line of the matrix \mathbf{M} , all values are equal to zero except for the column corresponding to the model cell where the parameter is known. For example, if the values are to be constrained only the first and last cells, \mathbf{M} and \mathbf{m}_c will be respectively

$$\begin{bmatrix} 1 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 0 & 1 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} mc_1 \\ mc_{np} \end{bmatrix}. \quad (2.20)$$

The advantage of this approach is that the LSQR algorithm can still be used to solve the system. On the other hand, the values of \mathbf{m}_c will never be exactly fitted due to the averaging effect imposed by the smoothness criteria.

2.2.2 Geostatistical inversion

Very often, the results of regularized least-squares algorithms are sensitive to the values of regularization Lagrange multipliers, which cannot easily be adjusted automatically. To circumvent this problem, an approach based on geostatistical analysis of the travel time data has been developed by Gloaguen et al. (2005). As reviewed below, it can be shown that the covariance model of the slowness field can be modeled from the travel time data. This covariance model describes the spatial characteristics of the slowness field. Using this covariance model and the fact that slowness and travel time are linearly related, it is possible to cokrig the slowness field using the travel time data. Cokriging is an interpolation technique allowing to use a more intensely sampled covariate (the travel time) in the estimation of values for a related

variate (the slowness) (Chilès and Delfiner, 1999). The procedure is said to be self-regularized because regularization is supported entirely by the data. It is also easy to go further and to impose slowness gradients or even any kind of linear constraint to the solution.

For known raypaths, slowness and travel time are linearly related. As a consequence, their covariance matrices are also linearly related. To illustrate this, define first the error between noise-free data d and the measured data d^* as

$$e = d^* - d. \quad (2.21)$$

In the following development, it is assumed that the error has zero mean and is uncorrelated with time, i.e. the expectation value $E[e] = 0$ and the covariance $\text{Cov}(d, e) = 0$. Working in matrix notation and replacing \mathbf{d} by \mathbf{Gm} , we can write $\mathbf{d}^* = \mathbf{Gm} + \mathbf{e}$. Working with contrasts, i.e. imposing $E[\mathbf{m}] = 0$ and $E[\mathbf{d}] = 0$, we get

$$E[\mathbf{d}^*] = E[\mathbf{Gm} + \mathbf{e}] = 0. \quad (2.22)$$

Since \mathbf{d}^* has zero mean, the observed data covariance matrix is simply

$$\mathbf{C}_{\mathbf{d}^*} = \mathbf{t}^* \mathbf{t}^{*\text{T}}. \quad (2.23)$$

After manipulation, we get for the $n_o \times n_o$ data covariance matrix (Gloaguen et al., 2005)

$$\mathbf{C}_{\mathbf{d}^*} = \mathbf{G} \mathbf{C}_{\mathbf{m}} \mathbf{G}^{\text{T}} + \mathbf{C}_0 \quad (2.24)$$

where $\mathbf{C}_{\mathbf{m}}$ represents the $n_p \times n_p$ model parameter covariance matrix and \mathbf{C}_0 is the $n_o \times n_o$ diagonal matrix of error variances or the nugget effect. Eq. 2.24 serves as the basis to obtain a model covariance function $\text{Cov}(m, m)$ from the experimental data. Note that the covariance $\text{Cov}(d^*, d^*)$ is not stationary. Therefore, estimators such as traditional variogram or covariogram cannot be used directly, and the model covariance function must be estimated by inversion. The program `bh_tomo_fitCovar`, part of the package, is designed to accomplish this task interactively using the concept of the V-V plot described in Asli et al. (2000), or automatically using a minimization scheme based on the Simplex method implementation of Matlab. The procedure is as following. Eq. (2.23) is first used to compute $\mathbf{C}_{\mathbf{d}^*}$. Then a theoretical covariance function $\text{Cov}(m, m)$ is chosen for the model, as well as initial values of range, sill and a nugget effect. The matrix $\mathbf{C}_{\mathbf{m}}$ is built for the tomographic inversion grid using $\text{Cov}(m, m)$, and the right term of Eq. (2.24) is finally computed. In the V-V plot approach, the so-called theoretical covariance values (right term of Eq. (2.24)) are sorted in decreasing order and binned in classes for which the mean value is calculated. The same sorting order and binning scheme is applied to the experimental data $\mathbf{C}_{\mathbf{d}^*}$. Thus, a value corresponding roughly to a single theoretical structural distance can be calculated for an experimental variogram. If the chosen covariance function and parameters are appropriate, the experimental variogram versus theoretical variogram plot (the V-V plot) should show a dispersion around a line at 45°. The advantage of our approach is that the model covariance function parameters are tuned by the observed data.

Once an acceptable model covariance function is obtained, the model parameter field is cokriged using the arrival times and any available known model data (Marcotte, 1991). This operation requires the cross covariance $\text{Cov}(d^*, m)$ to be known. It can be shown from the previous definitions that the cross covariance matrix is (Gloaguen et al., 2005)

$$\mathbf{C}_{\mathbf{dm}} = \mathbf{G} \mathbf{C}_{\mathbf{m}}. \quad (2.25)$$

In `bh_tomo`, primal cokriging is implemented to estimate the model parameter field. The cokriging weights are

$$\Lambda = (\mathbf{C}_d^*)^{-1} \mathbf{C}_{dm}, \quad (2.26)$$

and the cokriging estimator of the model parameter field is

$$\mathbf{Z}_g^* = \Lambda^T \mathbf{d}^*. \quad (2.27)$$

An advantage of cokriging is that any known value of the model parameter that is introduced in the system of equation will be found in the solution. Suppose for example that m_i model parameter values are known with variance σ_i at n_c cells along the boreholes. The cokriging weights become

$$\Lambda = \begin{bmatrix} \mathbf{C}_m^c & \mathbf{C}_{dm}^{cT} \\ \mathbf{C}_{dm}^c & \mathbf{C}_d^* \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{C}_{m0}^c \\ \mathbf{C}_{dm} \end{bmatrix}, \quad (2.28)$$

where

$$\mathbf{C}_m^c = \begin{bmatrix} \mathbf{C}_{m11} & \mathbf{C}_{m12} & \cdots & \mathbf{C}_{m1n_c} \\ \mathbf{C}_{m21} & \mathbf{C}_{m22} & \cdots & \mathbf{C}_{m2n_c} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{mn_c1} & \mathbf{C}_{mn_c2} & \cdots & \mathbf{C}_{mn_cn_c} \end{bmatrix} + \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \sigma_{n_c} \end{bmatrix}, \quad (2.29)$$

$$\mathbf{C}_{dm}^c = \begin{bmatrix} \mathbf{C}_{dm11} & \mathbf{C}_{dm12} & \cdots & \mathbf{C}_{dm1n_c} \\ \mathbf{C}_{dm21} & \mathbf{C}_{dm22} & \cdots & \mathbf{C}_{dm2n_c} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{dmn_o1} & \mathbf{C}_{dmn_o2} & \cdots & \mathbf{C}_{dmn_on_c} \end{bmatrix} \quad (2.30)$$

and

$$\mathbf{C}_{m0}^c = \begin{bmatrix} \mathbf{C}_{m11} & \mathbf{C}_{m12} & \cdots & \mathbf{C}_{m1n_p} \\ \mathbf{C}_{m21} & \mathbf{C}_{m22} & \cdots & \mathbf{C}_{m2n_p} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{C}_{mn_c1} & \mathbf{C}_{mn_c2} & \cdots & \mathbf{C}_{mn_cn_p} \end{bmatrix}. \quad (2.31)$$

At this point, it is interesting to note the similarities between our approach, which in terms of \mathbf{C}_d^* and \mathbf{C}_m is

$$\mathbf{Z}_g^* = \left[\left(\underbrace{\mathbf{G}\mathbf{C}_m\mathbf{G}^T + \mathbf{C}_0}_{\mathbf{A}} \right)^{-1} \underbrace{\mathbf{G}\mathbf{C}_m}_{\mathbf{B}} \right]^T \mathbf{d}^*, \quad (2.32)$$

and the Bayesian formulation, which writes (Tarantola and Valette, 1982)

$$\Delta \mathbf{m} = \mathbf{C}_m \mathbf{G}^T \left(\mathbf{G}\mathbf{C}_m\mathbf{G}^T + \mathbf{C}_d \right)^{-1} \Delta \mathbf{d}. \quad (2.33)$$

The latter is usually transformed in the form (Nolet, 1993; Maurer et al., 1998)

$$\begin{bmatrix} \mathbf{C}_d^{-1/2} \mathbf{G} \\ \mathbf{C}_m^{-1/2} \end{bmatrix} \mathbf{m} = \begin{bmatrix} \mathbf{C}_d^{-1/2} (\Delta \mathbf{d} + \mathbf{G}) \\ \mathbf{c} \end{bmatrix} \quad (2.34)$$

suitable in form to be solved using the LSQR algorithm. In the geostatistical framework, we use a different approach because the matrix \mathbf{A} in Eq. (2.32) is symmetric positive definite and

the system can be easily solved by Cholesky factorization. Apart from the differences in numerical solution of the above systems of equations, perhaps the most important advantage of using the geostatistical approach is that a model parameter covariance function constrained by the measured data is obtained. As shown next, this allows to study the spatial variability of the model parameter and measure the robustness of the model.

By construction, cokriging gives a smooth estimate of the model parameter field. It may be desirable and informative to obtain various reasonable solutions showing the kind of variability that can be expected from the model covariance function adopted. Within our framework, this can be done by using geostatistical simulation algorithms. Among the various simulation schemes, the FFT-MA algorithm (Le Ravalec et al., 2000) was retained for its rapidity. The FFT-MA algorithm generates non-conditional simulations \mathbf{Z}_g^s . Hence, the computed data of each simulation are not linked to the measured data. Therefore, it is necessary to post-condition the simulation by cokriging to constrain the simulations. First, the cokriging of model parameter \mathbf{Z}_g^* with measured data is performed using Eq. (2.27). Then, for each realization \mathbf{Z}_g^s , simulated data are computed using the relationship $\mathbf{d}^s = \mathbf{G}\mathbf{Z}_g^s + \mathbf{e}$, where \mathbf{e} is drawn from a normal distribution with zero mean and C_0 variance. Keeping the same cokriging weights, cokriging of the slowness \mathbf{Z}_g^{s*} with computed data \mathbf{d}^s is performed. Finally, the conditional simulated model \mathbf{Z}_g^{sc} is

$$\mathbf{Z}_g^{sc} = \mathbf{Z}_g^* + (\mathbf{Z}_g^s - \mathbf{Z}_g^{s*}). \quad (2.35)$$

\mathbf{Z}_g^{sc} has the desired covariance and fits exactly any linear constraint on the model. When the data covariances have zero nugget effect, the measured data and the computed data of each conditional simulation are exactly the same. When a nugget effect is present on observed data, the computed data depart from the measured data by an amount compatible with the level of error described by the nugget effect.

A measure of the robustness of the model can also be obtained by performing a large number of simulations and storing each realization of the slowness field. The realizations are all different and independent, but equally consistent with the measured data and the modeled covariance. The probability that the slowness exceeds a given value can then be drawn from a set of N realizations as following. At a given point x in the model parameter map, the probability of $m(x)$ being above a given threshold m_t is

$$P(m(x) > m_t) = \frac{1}{N} \sum_{i=1}^N I(m_i(x) > m_t), \quad (2.36)$$

with $m_i(x)$ being each realization i at point x and $I(m_i(x) > m_t)$ the indicator taking value 1 if $m_i(x) > m_t$ and 0 otherwise.

2.3 Ray tracing

Raytracing must be performed to build matrix \mathbf{G} . In `bh_tomo`, the shortest path method is used. The algorithm is described in the following.

2.3.1 Discretization of the slowness model

In the shortest path method, a grid of nodes is used to build a graph by connecting each node to its neighbours. The connections within the graph are assigned a length equal to the traveltimes along it. Hence, by virtue of Fermat's principle which states that a seismic ray

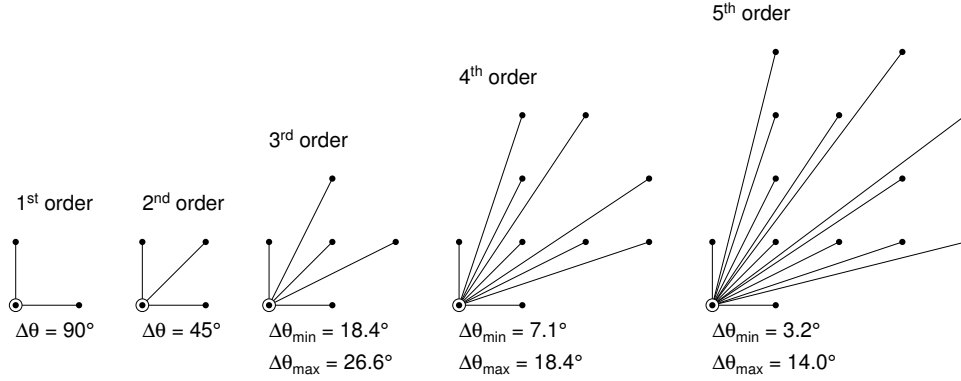


Figure 2.2: Different graph templates allowing for different angular coverage. For simplicity, only one quadrant of 2D templates are shown (after [Matarese \(1993\)](#)).

follows the minimum traveltimes curve, the shortest path between two points within the graph can be seen as an approximation of the raypath ([Nakanishi and Yamaguchi, 1986](#)).

[Moser \(1991\)](#) discusses briefly two ways of discretizing the slowness model: one attributed to [Nakanishi and Yamaguchi \(1986\)](#) in which grid nodes surround cells of constant slowness, and a second in which the slowness field is sampled at node locations. In the latter case, the sampling scheme may vary, but the nodes are generally distributed spatially in regular fashions such as on grids of cubic “cells”. Having nodes only at the corners of the cells limits drastically the allowed angular coverage, and various templates were designed to improve this aspect. For example, high order templates can reach nodes many cells away from the source point, as illustrated in Figure 2.2.

A drawback of using high order templates is that they require smooth slowness variations to maintain accuracy, or dense node sampling in otherwise high contrast regions. Hence, there is a trade-off to seek between accuracy and speed of computation. This is especially the case in inversion, where it is not always desirable to use very fine grids because this has for effect to increase the number of unknown in the system to solve, and therefore increase the overall computational effort. One could use two grids of different sizes, a coarser for the inversion step and a finer for the raytracing step, and interpolate between the two, but this adds to the computational burden and causes additional inaccuracies. Besides, using high order templates complicates the problem of building a system of equation for the inverse problem because ray segments will cross many cells and adequate bookkeeping increases the computational burden. In this regard, a discretization scheme based on nodes surrounding cells of constant slowness as proposed by [Nakanishi and Yamaguchi \(1986\)](#) is appealing.

[Gruber and Greenhalgh \(1998\)](#) have proposed a variant of the discretization approach of [Nakanishi and Yamaguchi \(1986\)](#), which is well suited for the tomography problem and which provides adequate angular coverage. In their study on the accuracy of traveltimes computation on 2D grids, they have used constant slowness cells with so-called primary nodes at the corner of the cells in addition to so-called secondary nodes on the edges of the cells. [Bai et al. \(2007\)](#) have used the idea of primary and secondary nodes in 3D and incorporate nodes on the sides of the cells. However, they interpolate the slowness at the secondary nodes and don’t use constant slowness cells. In both cases, ray segments (or connections between two nodes) are comprised within no more than one cell, which is very handy to simplify bookkeeping in tomography applications.

In `bh_tomo`, we follow [Gruber and Greenhalgh \(1998\)](#) and [Bai et al. \(2007\)](#) and use primary

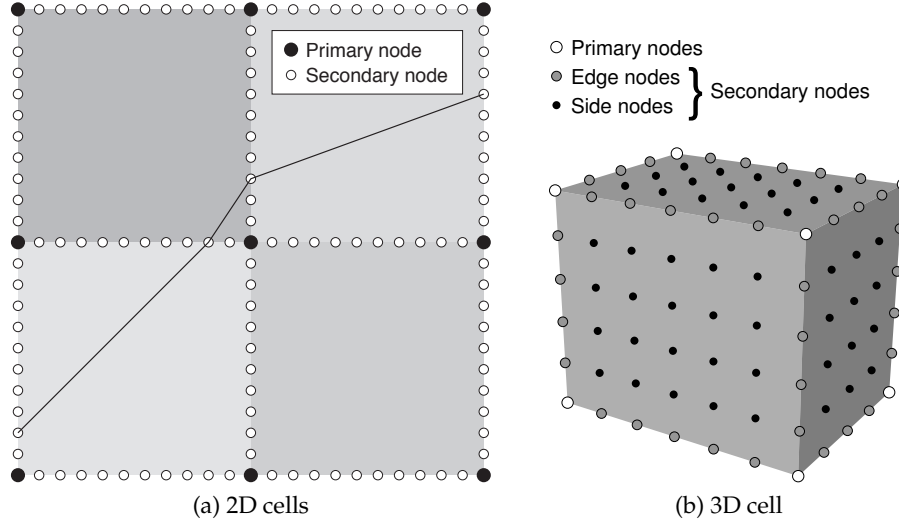


Figure 2.3: Illustration of elementary cells with location of primary and secondary nodes.

and secondary nodes surrounding rectangular cells (in 2D) or cuboids (in 3D) at their corners and on their edges and sides, as shown in Figure 2.3. The slowness value is assigned to the cell, not the nodes.

The raytracing algorithm retained in this work is synthesized in Algorithm 1, and proceeds as follows. After construction of the graph (i.e. establishing the allowed connections between the nodes), all nodes are initialized to infinite time except the source node which is assigned its “time zero” value. A priority queue is then created. Priority queues are a type of container specifically designed such that its first element is always the greatest (or smallest, depending on comparison criteria) of the elements it contains, according to some strict weak ordering condition (Cormen et al., 2009). This is similar to a heap where only the max (or min) heap element can be retrieved and elements can be inserted indefinitely. In our case, the highest priority is attributed to the node having the smallest traveltime value. The traveltime is computed for all nodes connected to the source node, the traveltime value at those nodes is updated with their new value, the parent node is set to the source node, and these nodes then are pushed into the queue. Then, the node with highest priority is popped from the queue, and the traveltime is computed at all nodes connected to it except the node parent, the traveltime and parent values are updated if the traveltime is lower than the one previously assigned, and the nodes not already in the queue are pushed in. This process is repeated until the queue is empty.

2.3.2 Implementation

The raytracing schemes are implemented in C++ following an object-oriented approach and make use of the Standard Template Library (STL) (Josuttis, 2012). There are two main classes: `Node3Dc` for the nodes and `Grid3Dc` for the grids of nodes. Member variables of the node class are given in Table 2.1. In order to compute traveltime values concurrently for multiple threads, variables `tt`, `nodeParent` and `cellParent` must be updated independently for each thread (`nodeParent` and `cellParent` are used to track raypaths and build matrix \mathbf{G} in eq. (2.3), respectively). For this reason, they are arrays of length equal to the number of threads used during the calculations. Finally, the node class is generic in that `tt`, `x`, `y`, `z` and `slowness` are C++ class template variables, and their type is either defined as `float` or

Algorithm 1 Basic algorithm for shortest path raytracing

```

1: instantiate nodes and build graph
2: instantiate priority_queue
3: for node = all nodes do
4:    $t_{node} \leftarrow \infty$ 
5: end for
6: for node = source nodes do
7:    $t_{node} \leftarrow t_0$ 
8:   push to priority_queue
9: end for
10: while priority_queue not empty do
11:    $node_1 \leftarrow \text{top of priority\_queue}$ 
12:   remove  $node_1$  from priority_queue
13:   for  $node_2$  = all nodes touching  $node_1$  except  $parent_{node_1}$  do
14:     compute  $dt$ 
15:     if  $t_{node_1} + dt < t_{node_2}$  then
16:        $t_{node_2} \leftarrow t_{node_1} + dt$ 
17:        $parent_{node_2} \leftarrow node_1$ 
18:       if  $node_2$  not in priority_queue then
19:         push  $node_2$  to priority_queue
20:       end if
21:     end if
22:   end for
23: end while

```

Table 2.1: Member variables of classes Node3Dc

Variable	Size	Type	Description
nThreads	1	int	Max number of threads used
tt	nThreads	float or double	Travelttime at the node
x, y, z	1	float or double	Coordinates
gridIndex	1	size_t	Index of the node in the grid list
nodeParent	nThreads	size_t	Index of parent node of the ray segment
cellParent	nThreads	size_t	Index of cell traversed by the ray segment
owners	up to 8	size_t	Indices of cells touching the node

double.

Class Grid3Dc hold its nodes in a vector STL container. This class also has a neighbors variable which holds, for each cell, a table listing the touching nodes. Finally, Grid3Dc objects keep slowness values themselves in a slowness member array. The class has a raytrace method which implements Algorithm 1. The core of this method is the STL priority_queue class. In the STL implementation, priority queues are implemented as container adaptors, which are classes that use an encapsulated object of a specific container class as its underlying container. The underlying container may be any of the standard container class templates or some other specifically designed container class, provided that it supports a minimal set of operations, specifically `front()`, `push_back()`, `pop_back()` (Josuttis, 2012). Therefore, the STL containers `vector` and `deque` (double ended queue) can be used. Both were tested to see if one offers consistent performance advantage over the other, and the tests showed that `vector` performed slightly better. Note that for efficiency reasons, the priority queue holds pointers to node objects rather than copies. As such, memory allocation calls are kept to a minimum, and `tt` as well as `nodeParent` and `cellParent` node member data can be readily updated.

Finally, parallelism is implemented by using the OpenMP (Open multiprocessing) Appli-

cation Program Interface. OpenMP supports shared memory multiprocessing programming in C, C++, and Fortran, and is portable to various processor architectures and operating systems (OpenMP Architecture Review Board, 2012)). An important advantage of OpenMP is its extreme ease of use; for example, one line of code is very often enough to parallelize a loop (Chapman et al., 2007). Note however that this comes at the expense of less performance tuning possibilities. In the proposed implementations, grid objects are instantiated with the maximum number of threads (`nThreads`) as an input argument, and this value is passed to the instantiation of node objects. Hence, storage is allocated to hold traveltime and raypath information for each thread, at each node. Then, the loop in which the raytracing method is called for each source is parallelized with an `omp parallel for` directive, and the number of the currently executing thread is passed as an argument to the `raytrace` method. This value is used to update the right elements of the `tt`, `nodeParent` and `cellParent` arrays. It should be stressed that the number of threads available on a given system is usually less than the number of sources. In such case, the task of splitting the jobs to the threads is handled automatically by OpenMP, which is achieved by a single directive, illustrated by line 5 in Algorithm 2. In this pseudocode, the OpenMP routine `omp_set_num_threads` is first called to set the number of threads that will be available in the thread pool, and `omp_get_thread_num` is called in the `for` loop to retrieve the ID of the thread that will handle a particular source.

Algorithm 2 Pseudocode for invocation of OpenMP directive

```

1: Choose nThreads value
2: Assign to the system: omp_set_num_threads(nThreads)
3: Instantiate Grid object: g = Grid(..., nThreads)
4: Instantiate source array: Tx = Source(nSources)
5: #pragma omp parallel for
6: for n = nSources do
7:   g.raytrace(Tx[n], ..., omp_get_thread_num())
8: end for

```

Software package description

3.1 Main user's interface

The `bh_tomo` package is divided into many interactive modules that allow one to create and manage a mini database; pick the traveltimes semi-automatically, automatically or manually; process the traveltime and amplitude data; adjust a covariance model to the data; perform the inversions, visualize time-lapse data, and ultimately estimate the saturation of magnetic nano-fluid. These programs are accessible from the `bh_tomo` window (Fig. 3.1), or from the Matlab command line.

3.1.1 General processing flow

Database creation

To start, the user has to execute the module database in which boreholes, Multi-offset gathers (MOGs) data, panels and grids are defined, imported and created. A database created and saved in the module `bh_tomo.db` can be defined as default for all modules by selecting it from **Choose DB** in the main menu **File**. The name of the database will be displayed in the white text box on the top of the window and it will be available to all the modules of the package. It is important to note that the database is reloaded only when the modules are called.

When an existing database is reloaded, the program will check if the database fields are consistent. If any of the necessary parameters is not defined during the creation of the database, a warning dialog will appear and all the missing fields will be displayed in a list.

The database can also be reloaded from **File** menu of each module. Database that are modified at any step of processing can be saved or saved under a different name. The database loaded in a module is only local to this module and will not be available to the other modules until it is saved. So the user may work with different databases in different modules.

Traveltime and amplitude data

After a database is created, the next step is to pick traveltime data. Traveltime picking can be done semi-automatically, automatically or manually by executing corresponding module. Traveltime picking is done on a per MOG basis, each module processing MOG datasets independently.

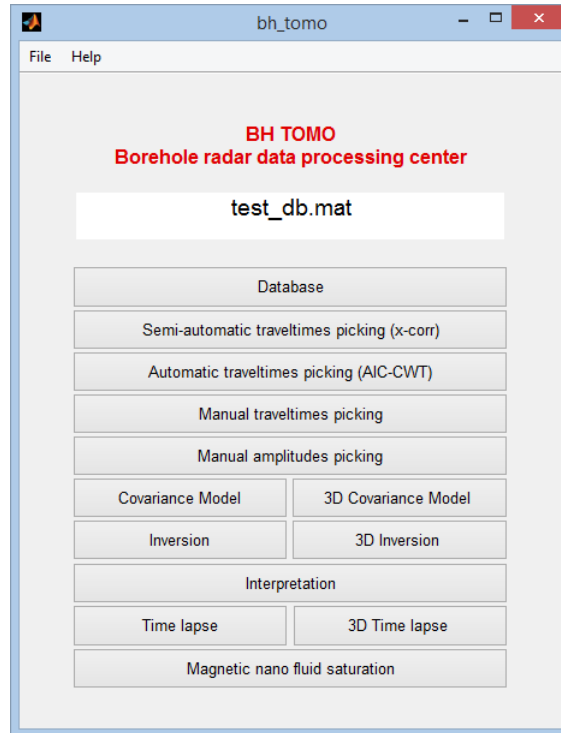


Figure 3.1: Interactive screen of `bh_tomo`.

Processing amplitude data is usually done after inversion of traveltimes data, but this is not a requirement.

Inversion

Covariance model function should be set before carry out geostatistical inversions. For the first inversion run, the covariance model functions are computed using straight rays. After an inversion is done, the results can be passed to the `bh_tomo_fitCovar` or `bh_tomo_fitCovar3D` modules and can be used in computing new covariance model functions with curved rays.

Inversion of amplitude data must be performed after inversion of traveltimes data because the ray path must be known to solve equation (2.15).

Interpretation

In the module `bh_tomo_timelapse`, the user can visualise all the computed inversion tomograms and difference tomograms.

3.2 Database management

The program `bh_tomo_db` must be used first to create the database, define the borehole locations, import the data and create inversion grids. Fig. 3.2 shows the interface of `bh_tomo_db`.

If a database is already imported from the main interface, the data will be uploaded when the database module is executed and all the fields of the user's interface that are defined will

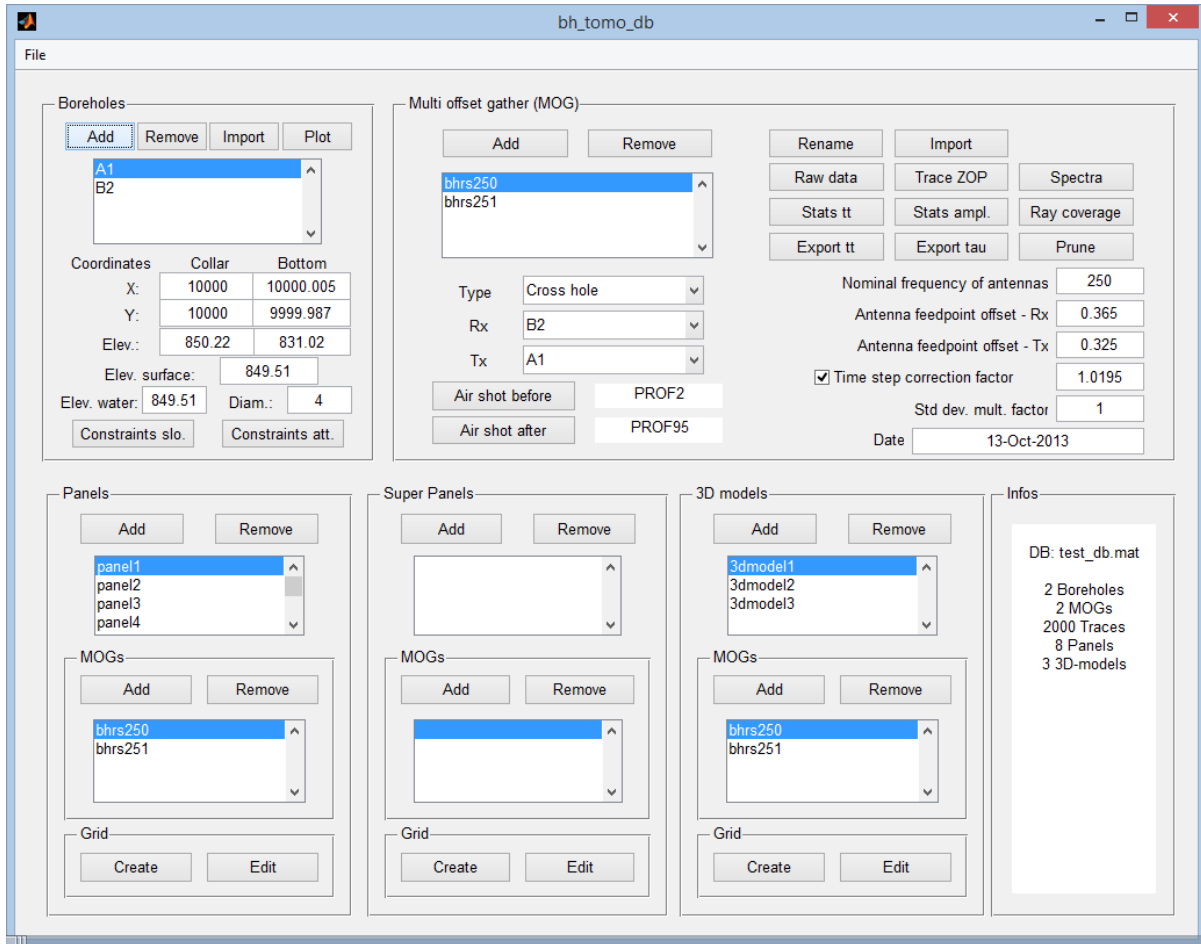


Figure 3.2: Interactive screen of bh_tomo_db.

be filled. The user can also import another database from this module from the **File** menu and the new one will replace the previously opened database.

3.2.1 Boreholes

Boreholes can be defined manually or imported from a file.

Add and **Remove** When the button **Add** in the **Boreholes** pane is clicked, an input dialog (Fig. 3.3) is popped and a name is required to identify the borehole. To remove a borehole, click the button **Remove**.

The borehole coordinates and other parameters can be entered in the corresponding edit

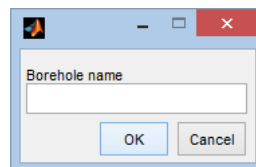


Figure 3.3: Add borehole.

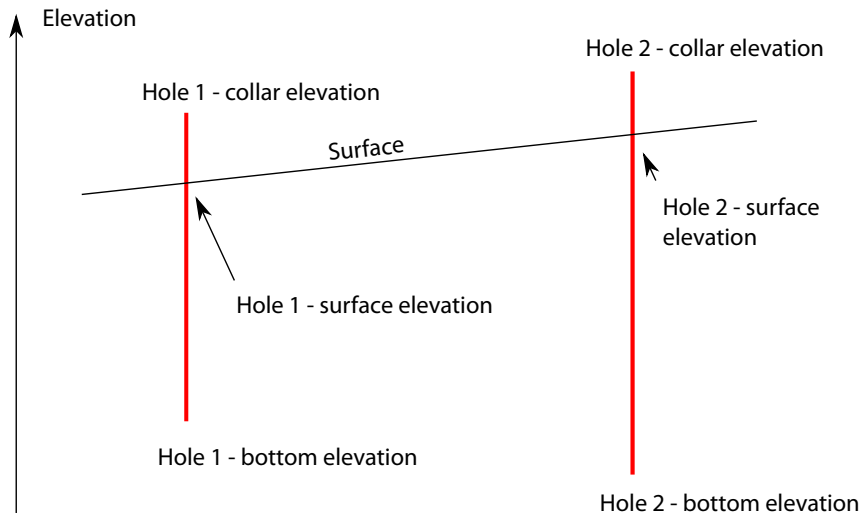


Figure 3.4: Definitions of borehole coordinates.

boxes in the pane. In such case, straight boreholes are defined. Constraints on slowness and attenuation can be imported by clicking corresponding buttons. Note that Z coordinates are elevations. Figure 3.4 illustrates the definitions of borehole coordinates.

Import Boreholes trajectory can also be predefined in an ASCII file and imported by clicking the button **Import**. This is currently the only way to have deviated boreholes. The file must have a `.xyz` extension. These files must contain three columns, for X, Y and Z coordinates respectively.

Plot The boreholes can be visualised by clicking the button plot and a 3D graphic will be displayed (Fig. 3.5).

Constraints slo. This button is for importing constraints from a file of slowness data. *This feature will work correctly for vertical boreholes only.* The file must have a `.con` extension, and contain two or three columns. The first column is the distance along the borehole starting from the collar, the second column is the slowness value at the corresponding elevation, and an optional third column holds the variance of the slowness.

Constraints att. This button is for importing constraints from a file on attenuation. This feature will also work correctly for vertical boreholes only. The file format is the same as for slowness constraints.

3.2.2 Multi-offset gathers

The **Multi-offset gathers** pane regroups routines to manage and visualize the raw data. The first step is to import a dataset. After importing a multi-offset gather datafile, the user must specify which of vertical radar profiling (VRP, i.e. surface to hole measurements) or cross hole acquisition mode was used for the survey file, and specify in which boreholes were the transmitting and receiving antennas (see Fig. 3.6). It is also possible to specify the names of two data files corresponding to “air shots”, i.e. shots done with the antennas lifted above the ground and separated by a measured distance. The air shots are used to determine the true emission time (noted t_0 in `bh.tomo`) and also to control for instrument drift. Air shots should be performed immediately before and immediately after data collection. In this manner, it is possible to compensate approximately for an instrumental drift (a linear correction is applied).

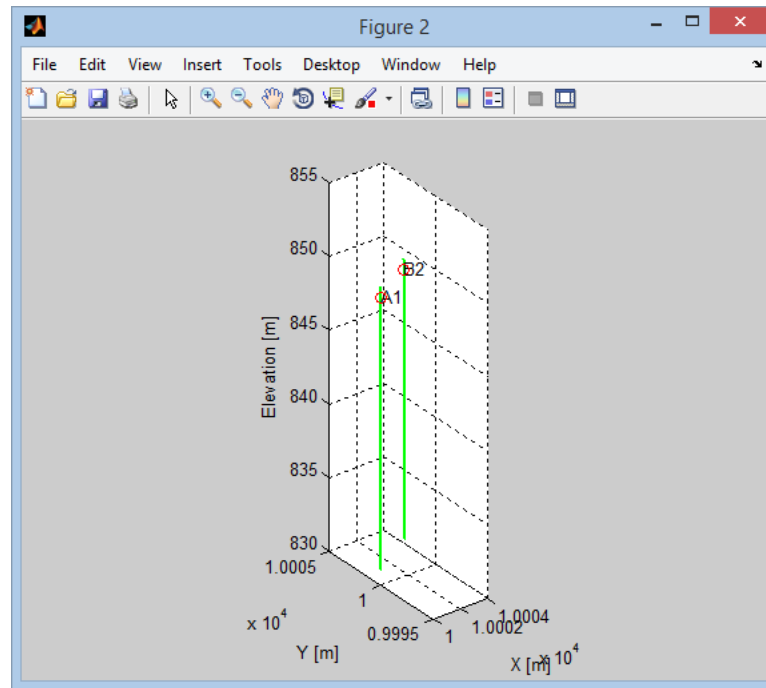


Figure 3.5: 3D view of the boreholes.

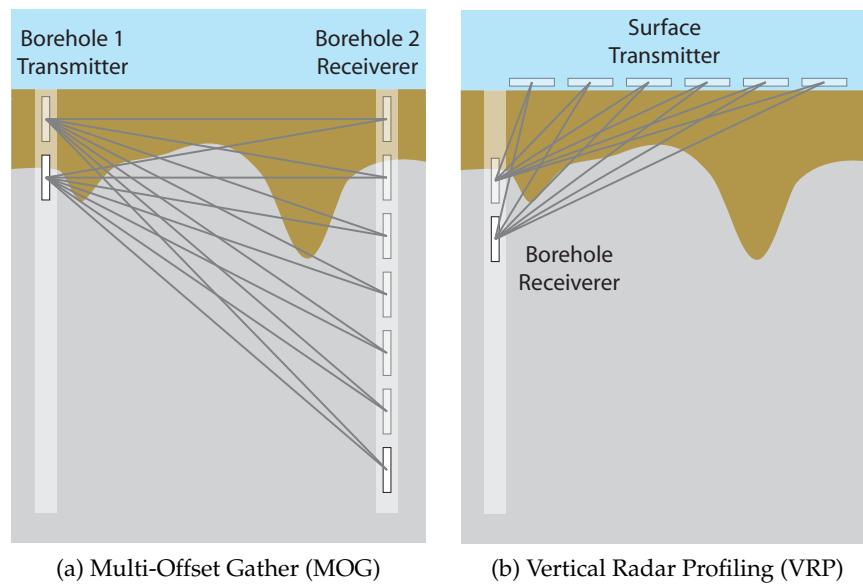


Figure 3.6: Borehole survey modes

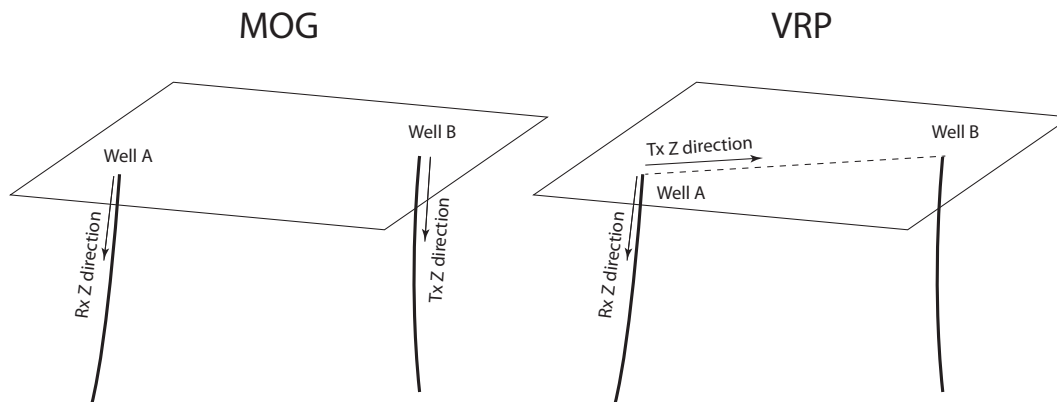


Figure 3.7: Convention for transmitter (Tx) and receiver (Rx) coordinates.

Add and **Remove** MOG data must be saved in one of the following format: Malå's RAMAC, Aramco's Magnetic NanoMapper (defined in A.1), Sensors & Software's EKKO, MSIS, Seismic Unix, SEG Y. It is added to the database by clicking the **Add** button. All the added MOGs will be displayed in the list box below the **Add** and **Remove** buttons. To remove a MOG, select a MOG in the list box and then click the **Remove** button.

Data in RAMAC format It should be emphasized that `bh_tomo` works with elevation (positive axis upward) rather than depth for the vertical axis. For practical reasons, the data acquisition procedure developed with RAMAC systems considers the positive vertical direction downward, with the zero position at the collar of the borehole, and this is the information that appears in the datafile. To obtain the antenna true positions, the `bh_tomo_db` routines subtract the vertical antennas position given in the RAMAC data file from the elevation of the borehole collars. In VRP acquisition mode, the "vertical" direction corresponds to the direction from the Rx borehole toward any other borehole which is then labeled the Tx borehole. The zero is at the Rx borehole. The (x, y, z) coordinates of each position of the transmitter are the projection of the "vertical" positions on the line defined by the x, y and surface elevations of both boreholes (see Fig. 3.7).

Air shot before and **Air shot after** Use these buttons to import air shot recordings. The user will be prompted to enter the distance between Tx and Rx antennas, which is needed to establish the true emission time. It is also possible to check the instrument sampling rate by doing measurements in air at increasing distance between Tx & Rx. The slope of distance vs traveltime data should be equal to the velocity of EM waves in air, i.e. 0.3 m/ns, and deviation from this value is used to compute a correction factor to apply to the data (see below the **Time step correction factor** text field). To use this feature, the user should enter a number of distance values equal to the number of traces in the data file. Figure 3.8 shows an example of the two ways to input increasing distances. In the first case, the matlab notation involving the colon is used for giving a vector of values regularly increasing. In the second case, values in brackets are used to give arbitrary values.

Rename Select a MOG from the list box and then click the **Rename** button. A dialog with the selected MOG name will appear and user can enter a new name for the MOG.

Import MOGs can also be imported from another database. When the button **Import** is clicked, a dialog window **Choose MOG** appears, which promotes the user to select a database

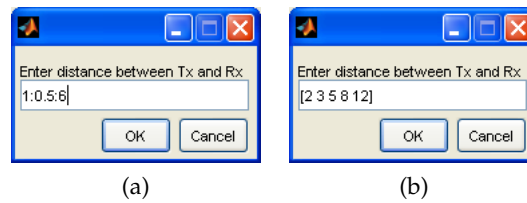


Figure 3.8: Illustration of the two ways to input increasing distances between Tx and Rx antennas for air shots.

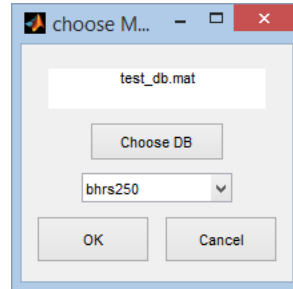


Figure 3.9: Import MOG dialog window.

(Fig. 3.9). If a database is already imported from the principal interface, the database name, as well as a list of available MOGs will be displayed in the white text box and popup menu respectively.

Raw data This button is used to display the MOG's raw data. Figure 3.10 is an example.

Trace ZOP is for displaying zero-offset profile (ZOP) plot. When the button is clicked, a new window appears (Fig. 3.11). The ZOP traces are displayed in the leftmost figure, and the apparent velocity is shown in the middle, provided that traveltimes have been picked for the displayed traces. The edit boxes `t_min` and `t_max` are used to define time scales of the elevation vs time graphic, which is at the left of the window; `z_min` and `z_max` for elevation scales in both graphics. The next edit box is used to modify the value of maximum vertical Tx-Rx offset. If this value is set to zero, only traces that are perfectly horizontal will be displayed. Increase this value if only a few traces are displayed. The colorscale sensitivity of the leftmost figure can be changed either by selecting the level in the popup menu or by entering a different value in the edit box, which are located in the middle of the Control pane at the right of the window. The first three checkboxes are used to show Apparent velocity, borehole velocity constraints and Peak-to-peak amplitude graphics. The last checkbox is to display the value that is used to correct for geometrical spreading. The graphics can be printed by clicking the button **Print**. When the button **Show rays** is clicked, the ray coverage will be displayed in a new figure window.

Stats tt Displays the air shot data, traveltimes statistics and ray coverage plot (Fig. 3.12). Figure 3.12a will appear only if air shots were done with increasing distance between Tx and Rx antennas.

Stats ampl. Displays amplitude statistics.

Spectra Displays frequency spectra of the data and S/N ratio of the corresponding traces (Fig. 3.13). Traces are binned by common transmitter. The user has the option to select between

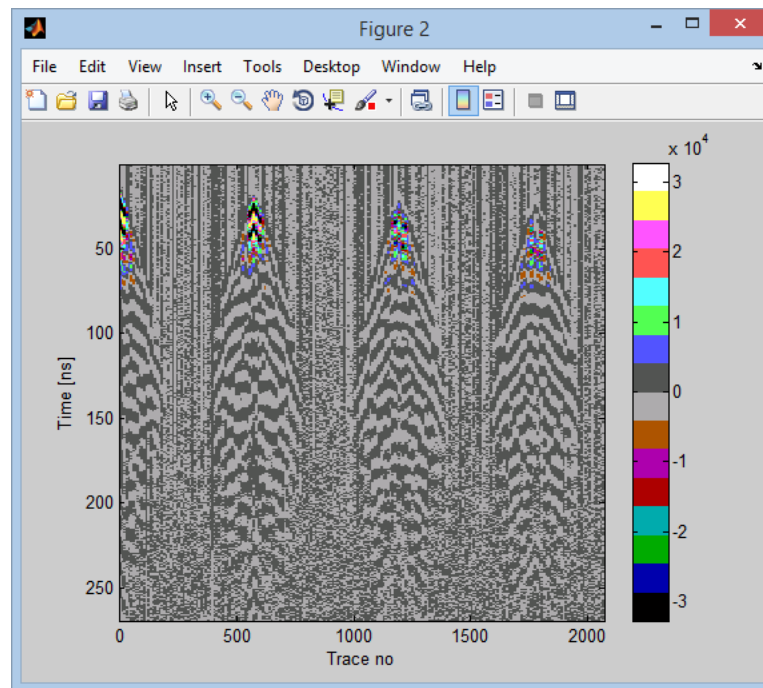


Figure 3.10: Window showing raw MOG data.

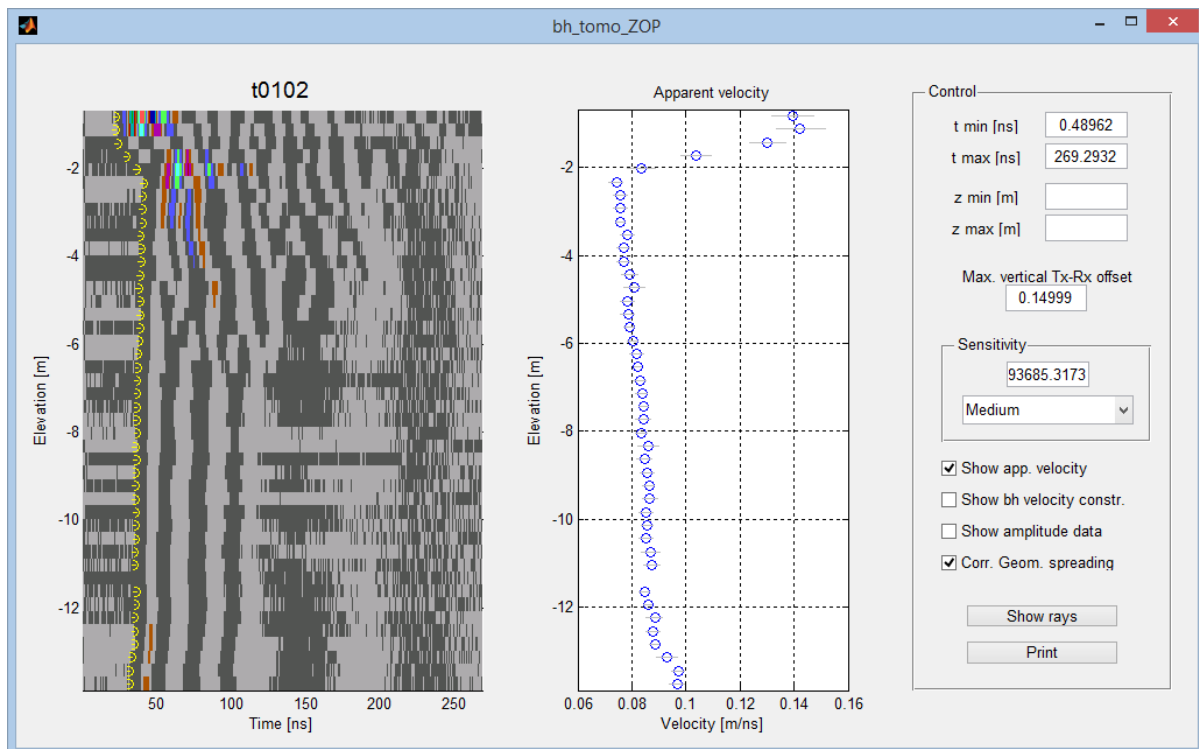
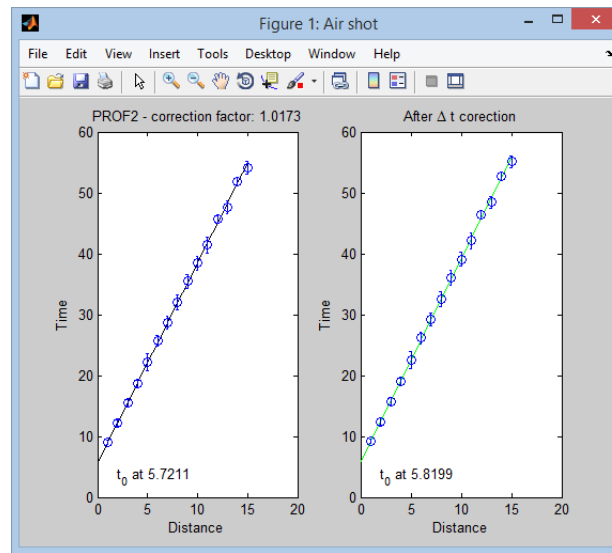
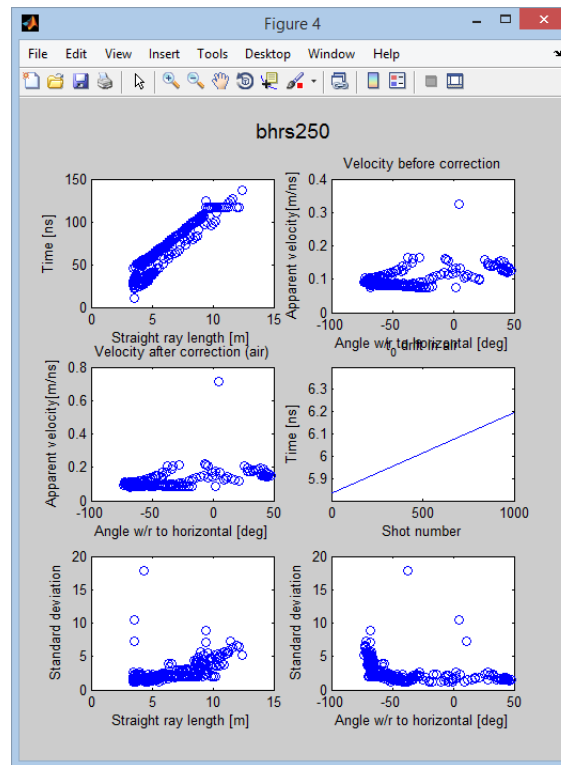


Figure 3.11: Zero-offset profile display window.



(a) Air shot data



(b) Traveltime statistics

Figure 3.12: Example of air shot data and statistics on picked traveltimes.

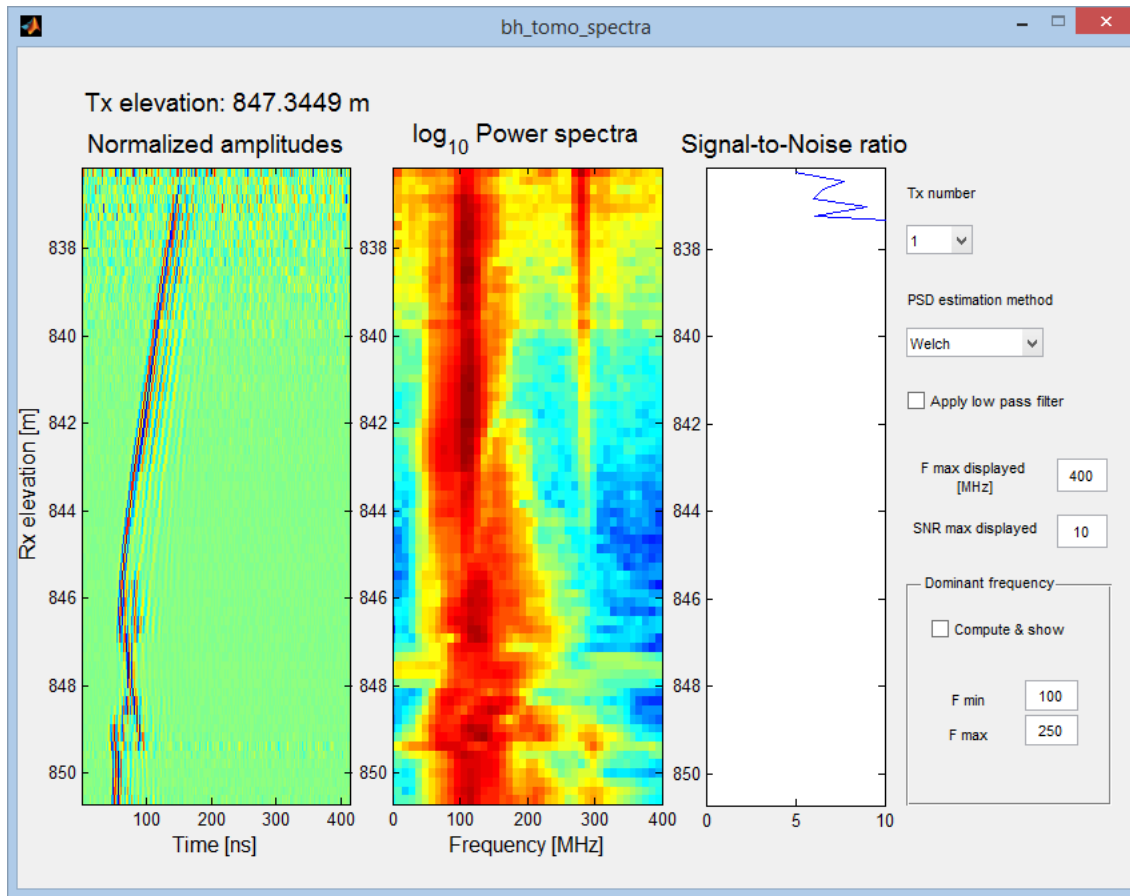


Figure 3.13: Window showing power spectral density estimates (center) of the selected traces (left) along with S/N ratio (right).

the methods of Welch and Burg to compute the power spectral density estimates.

Ray coverage Displays ray coverage for a homogeneous model (straight rays). Figure 3.14 shows an example. Rays that were pruned are drawn in red, whereas rays that are kept are in green.

Export tt Export picked traveltimes data to an external file.

Export tau Export reduced amplitudes (equation (2.10)) to an external file.

Prune This button brings an interactive window (Fig. 3.15) in which it is possible to define parameters that will tag rays as “excluded”, i.e. these rays will not be used in inversion but will remain in the database. It is possible to change these parameters afterwards, but the user should be aware that rays that were obtained from traveltimes inversion of previously pruned dataset cannot be used to invert amplitude data due to inconsistencies in the size of the ray matrix.

There are also a number of text fields in the lower right part of the pane. **Nominal frequency of antennas** can be used to specify the value. **Antenna feedpoint offset – Rx** and **Antenna feedpoint offset – Tx** are used to specify the distance between the feed point of dipole antennas and the point used as reference when positioning the antenna at the well head (see Fig. 3.16). **Time step correction factor** can be used to correct for inaccurate sampling rate at the digitizer. If selected, the time step is multiplied by the factor. **Std. dev. mult. factor**

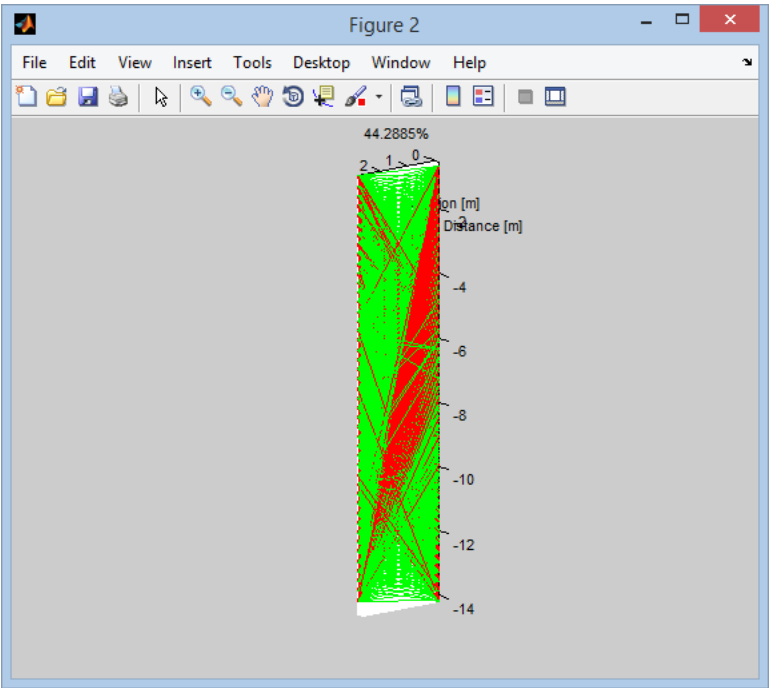


Figure 3.14: Ray coverage for a MOG dataset.

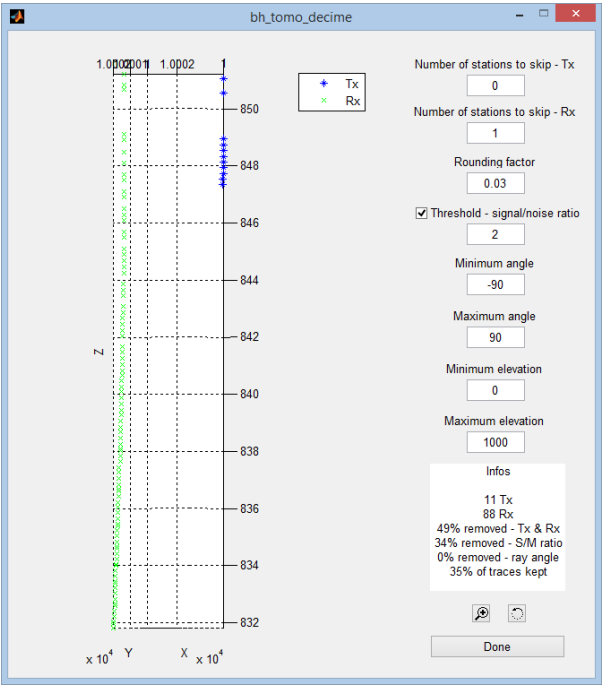


Figure 3.15: Utility window to prune MOG datasets.



Figure 3.16: Illustration of reference point and feed point of dipole antennas.

is a multiplier that is applied to the standard deviation of traveltime data. Finally, **Date** is the date of data acquisition, which is used when processing time-lapse data.

3.2.3 Panels

Panels and super-panels are entities that are used to represent 2D cross-sections that are imaged by the tomographic process. Super-panels are adjacent panels that can be juxtaposed. Currently, *super-panels can be built only from panels made with vertical boreholes*.

Add panel When the button Add is clicked in the Panels pane, a dialog is popped (Fig. 3-14) that allows the user to enter a name for the panel to be created. By default, the panel name is automatically numbered incrementally.

Add MOG After a panel is created, MOG data must be tied to it. This is done by clicking the **Add** button in the MOGs pane. A dialog window appears and the user must then select one or more MOGs from the list.

Grid utility

Once the panels are defined, their inversion grids must be created. The program `bh_tomo_grille`, called from `bh_tomo_db` with the buttons in the **Grid** boxes, allows designing the grids. Using `bh_tomo_grille`, it is possible to select the relative positioning of the boreholes, as well as the grid step size and extents. It is also from `bh_tomo_grille` that the model constraints utility is called. Adding constraints is possible only for regular panels, not super-panels.

3.2.4 General processing flow

Operations in `bh_tomo_db` should follow the following order:

1. Define boreholes;

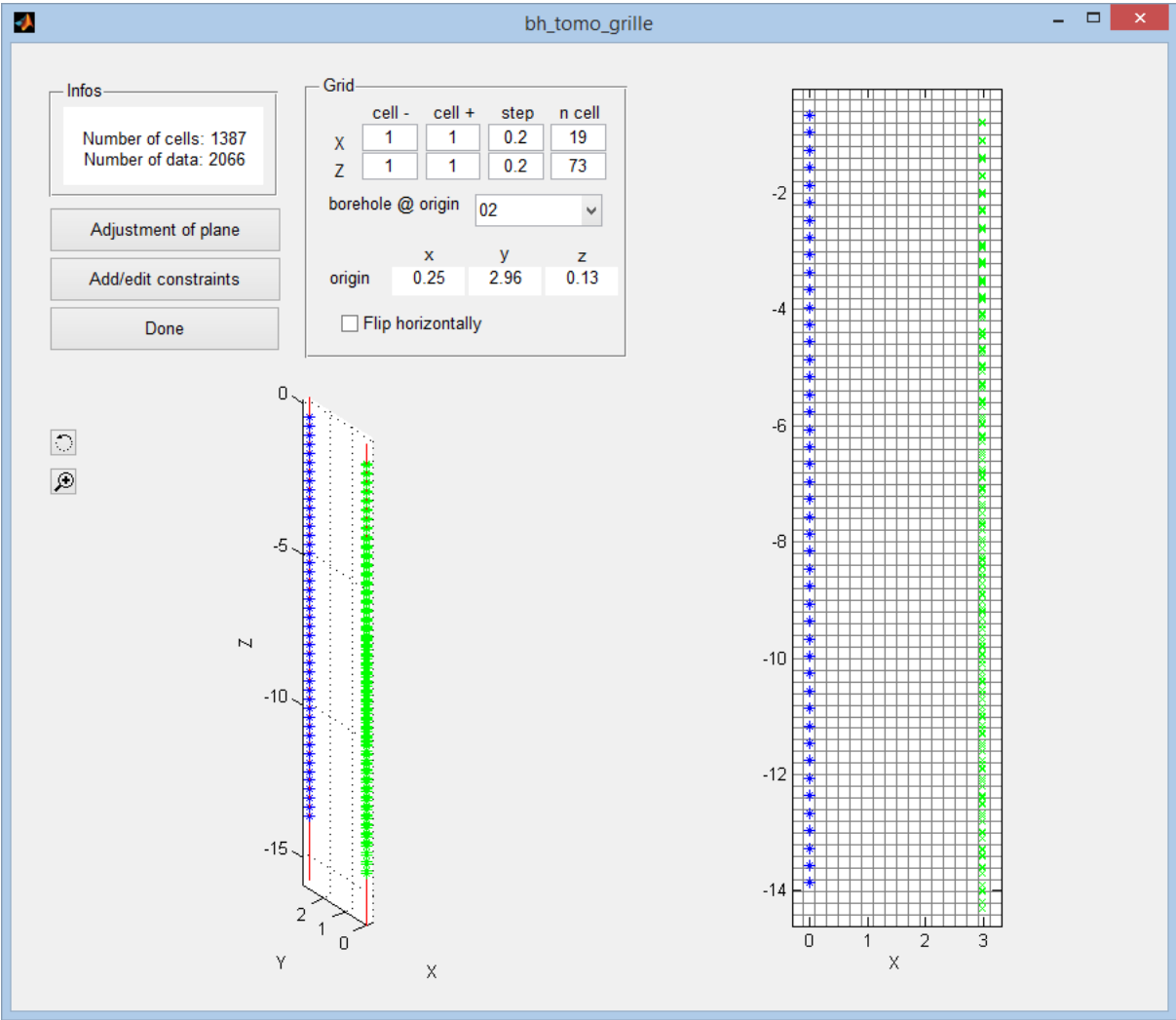


Figure 3.17: Interactive screen of `bh.tomo_grille`.

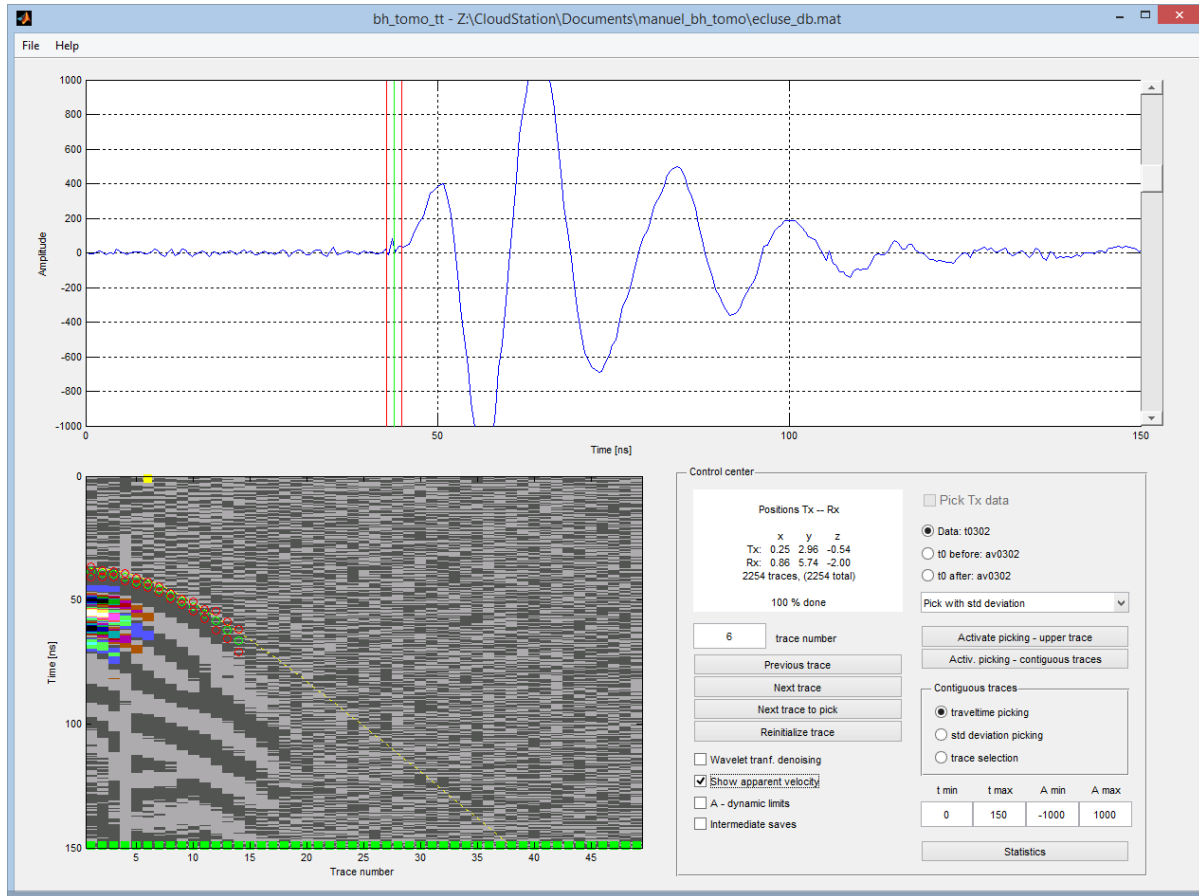


Figure 3.18: Interactive screen of `bh_tomo_tt`.

2. Import MOG data. Optional steps are:

- Define air shots;
- Prune dataset.

3. Create panels, super-panels or 3D models.

3.3 Traveltime processing

3.3.1 Manual picking

Once data are imported in the database, the travel times can be picked with `bh_tomo_tt`. In this program, multi-offset gather (MOG) data are processed independently rather than on a per panel basis. As shown in Figure 3.18, the data are displayed in two windows: single traces are shown on top, and all traces acquired for a given Tx position are drawn contiguously in the lower left box. This lower box allows a qualitative comparison of the picked times. The travel times for the air shots can also be picked in `bh_tomo_tt`, by selecting the appropriate dataset in the upper right part of the Control center box.

In `bh_tomo_tt`, it is possible to pick an interval of confidence (standard deviation) for each trace. If standard deviations are picked, they will be used in the geostatistical inversion

to build the C_0 matrix. The interval of confidence is a *subjective* parameter, and it is used to give more or less weight to the traveltime data during the inversion. It is thus important that it is picked consistently, i.e. preferably by the same user.

Picking is normally done in the upper window, but it can also be done in the lower left box, albeit with less accuracy. When picking in the upper window, the user can choose between “picking with standard deviation” and “simple picking” in the popup menu on the right. Once picking is activated, each mouse button has a different function:

- left button to pick traveltime;
- center button to accept pick and go to next trace;
- right button to pick standard deviation.

Picking is deactivated by clicking outside the upper window.

It is possible to view the arrival times and standard deviations as a function of the ray length or ray angle by pressing the **Statistics** button. It is also possible to enhance the signal quality using a wavelet transform (WT) de-noising function. Note that this function requires the wavelet toolbox sold by the MathWorks. The filtering parameters are hard-coded in the program. Because of this, the original trace is shown in the background in light grey to appraise the quality of the filter when WT de-noising is selected. Note also that the picked times are saved in the database and can be accessed at any time. Hence, picking sessions can be interrupted and restarted afterwards. Finally, travel times picked in a third party software can be imported in `bh_tomo.tt` (see appendix A.2 for file format).

3.3.2 Semi-automatic picking

The semi-automatic picking method is based on the work of [Irving et al. \(2007\)](#). Irving’s algorithm is based on the crosscorrelation of traces with a reference waveform for which the arrival time is known. The originality of the approach resides in the reference waveform being obtained iteratively from the data set, through a specific sorting of the traces. First, the traces are aligned using crosscorrelation with the reference trace being initially the one that has the highest S/N ratio. The aligned traces are then stacked to produce a new reference waveform, which is used to realign the traces. This process is repeated until all traces are properly aligned. Finally, the arrival time is picked manually on the reference trace with the arrival times of the other traces determined from the crosscorrelation lags. The method is said to be semi-automatic because the user must pick the reference trace and also intervene during the alignment of the traces in order to reject those that cannot be aligned properly.

To mitigate the effect of changes in the GPR phases on crosscorrelation, [Irving et al. \(2007\)](#) determine a number of different reference waveforms that represent ranges in transmitter/receiver angle where the arriving pulses have similar shape. This is achieved by gathering the traces by ray angle intervals. As illustrated in their paper, this step is essential in order to obtain high quality results. The main assumption behind this step is that waveforms recorded at similar angles have similar shapes. [Irving et al. \(2007\)](#) clearly point out that significant problems may arise in situations where this assumption is violated, such as for data collected above the water table or near the surface, where sharp velocity contrasts, refracted arrivals, and changes in antenna coupling impact greatly on the waveforms and frequency content.

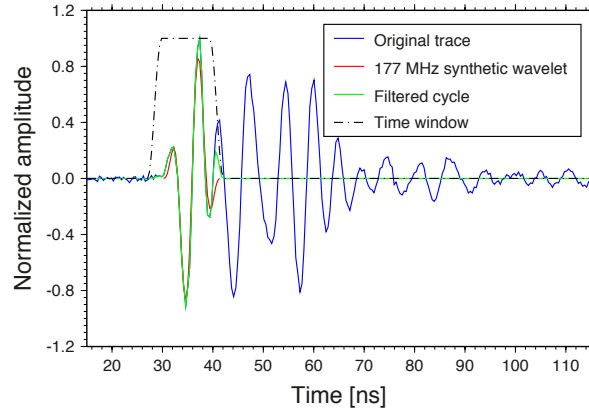


Figure 3.19: Example of first-cycle isolation. The original trace, in blue, has a dominant frequency of 177 MHz. This trace is crosscorrelated with a 177 MHz synthetic wavelet, shown in red. After the application of the time window, the scattered arrivals are eliminated, as shown in green.

First-cycle isolation

To overcome the limitation of required waveform similarity, it is possible to preprocess the data in a manner that isolates the first cycle of the GPR waveform (referred to as “first-cycle isolation”). By doing so, waveform similarity is increased and interference with scattered arrivals during crosscorrelation can be avoided partly, if not completely. This step is typically applied to traces with moderate to high S/N ratios because the scattered arrivals are usually not visible on traces with low S/N ratio. The preprocessing step proceeds as follows. Each trace is first normalized to its maximum absolute value. The traces are then crosscorrelated with a wavelet of the form (Arcone, 1991)

$$w(t) = \sin(2\pi ft) \sin^2\left(\frac{2\pi ft}{4}\right) \quad \text{with} \quad 0 \leq t \leq \frac{2}{f}, \quad (3.1)$$

where f is chosen to be the dominant frequency of the signal. To automate the procedure, the beginning of the first cycle is picked at the first time lag exceeding a given crosscorrelation amplitude threshold, typically 0.25. A time window having a width slightly larger than the wavelet of equation 3.1 and beginning slightly before the above time lag datum is then applied to nullify the undesired signals (see Figure 3.19). In this manner, we ensure that the sought onset is not altered. The time window is extended with cosine tapers at the leading and trailing edges, to smooth out transitions. Note that the wavelet of equation 3.1 could be substituted by any other that would be more adapted to the data at hand. However, since we only need to use a time window to suppress later arrivals and do not need to determine the onset time itself, equation 3.1 works well enough.

Dominant frequency rescaling

Another problem that arises when using crosscorrelation for onset picking is that when the dominant frequencies of the wavelets vary, the widths of the wavelets vary and, therefore, they differ from the reference wavelet (left of Figure 3.20). An immediate consequence is that the crosscorrelation lags will not fall exactly at the onset, leading to an error in the picked time.

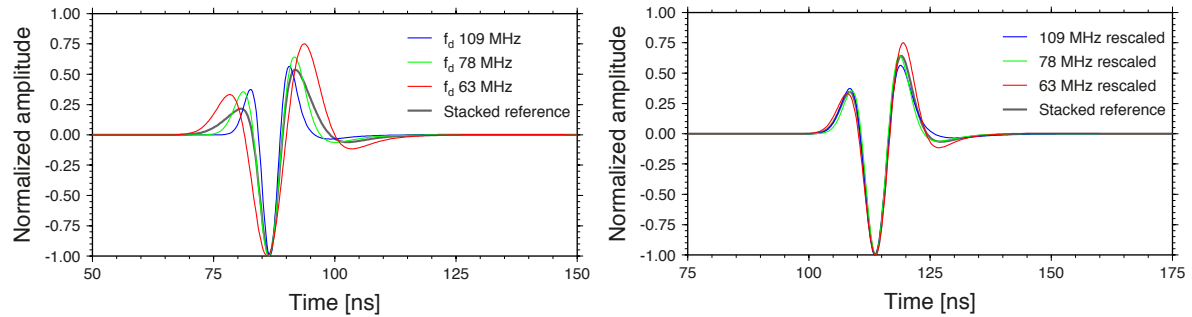


Figure 3.20: Illustration of the effect of time rescaling for three synthetic traces having different dominant frequencies. The traces are aligned without (left) and with (right) time rescaling.

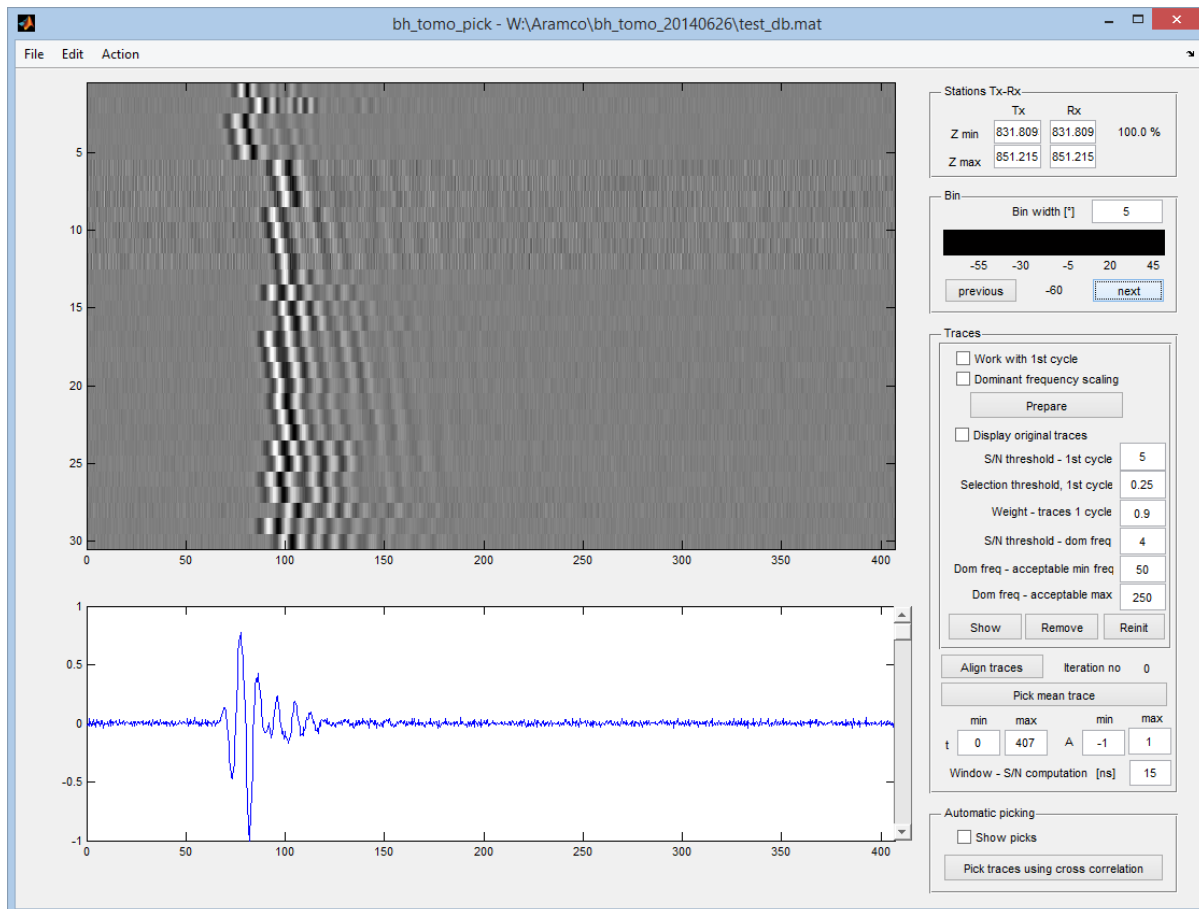
To help minimize this problem, the ability to stretch the first cycles to a common length has been added to the procedure. This is done by rescaling the time scale of each trace by the ratio f_d^i / \bar{f}_d , where f_d^i is the dominant frequency of the i^{th} trace and \bar{f}_d is the average of the dominant frequency for all traces. Because the crosscorrelation picking procedure requires that the time scales be consistent, all the stretched traces must be interpolated to a common time scale. An example of the results obtained with this procedure is shown in Figure 3.20, illustrating the improved similarity between the rescaled traces and the reference obtained after stacking. Finally, note that after completion of the picking procedure, the obtained times are scaled back to their original time frames.

User interface

The module that allows performing semi-automatic picking is `bh_tomo_pick`. The user's interface is shown in Figure 3.21. The steps to follow to process the data are:

1. Load MOG data, choose bin width;
2. Optionally select whether first-cycle isolation and/or dominant frequency rescaling should be applied. If so, press the **Prepare** button to process the data accordingly. These processing steps are restricted to traces with signal to noise ratio above the values specified in the "Traces" box. Finally, in first cycle isolation, it is possible to combine the isolated cycle with the original trace by giving a weight less than 1 in the field named *Weight - traces 1 cycle*.
3. For each bin:
 - (a) align traces by pressing **Align traces**; repeat as necessary;
 - (b) Pick the traveltime and standard deviation of the reference trace. Picking is activated by pressing **Pick mean trace**. The mouse buttons have the same effect as described in section 3.3.1.
4. After all bins have been processed and reference traces been picked, perform automatic picking of the remaining traces by pushing **Pick traces using cross correlation**.
5. Save your work.

Quality control can be done afterwards in `bh_tomo_tt`.

Figure 3.21: Interactive screen of `bh_tomo_pick`.

3.3.3 Automatic picking

Automatic picking method is based on the Akaike information criterion (AIC) and continuous wavelet transform (CWT). The window of Automatic picking module is shown in fig XXX

Different wave types can be present in a radar trace. Hence, similarity between traces recorded at different locations is never guaranteed. It is therefore appealing to have an automatic picking method that does not rely on waveform similarity. For this reason, the AIC picker is chosen as a starting basis because of its simplicity and efficiency. CWT is used to correct the ACI to obtain a higher level of accuracy of picking.

The approaches can be summarised as following:

- 1). After pre-processing the data, the traces are truncated in time at their maximum amplitude, the AIC is calculated on all truncated traces; the onset picking is determined by finding the AIC minimum.
- 2). For each trace, the dominant frequency of the radar wavelet is calculated and then used to scale the complex wavelet transform. If the dominant frequency cannot be found with the procedure, the frequency at the maximum of the amplitude spectrum is taken.
- 3). The position of the first break of radar wavelet is extracted from the phase part of the complex wavelet transform. Then, ensure the CWT correction falls near the ACI peak.
- 4). After all traces are processed, common-receiver-gather plots with associated first-arrival picks and uncertainties are displayed.

3.4 Amplitudes processing

Similarly, `bh_tomo_amp` can be used to process the amplitude data. Like in `bh_tomo_tt`, multi-offset gather data are processed independently rather than on a per panel basis. In `bh_tomo_amp`, the user selects a time window into which the analysis is carried out. Also, upon user selection, peak-to-peak amplitude or centroid frequency is computed. In the later case, the frequency spectrum can be computed using a simple FFT after application of a Hanning window to the selected portion of the trace, or using the S-transform (Stockwell et al., 1996). The S-transform gives a joint time-frequency representation of the signal, similar to the continuous wavelet transform, but which retains a direct relationship with the Fourier spectrum. To compute the centroid frequency with the S-transform, the spectrum with maximum energy in the selected time window is used. An example is shown in Fig. 3.22. A trace is shown in the upper part and its corresponding time-frequency spectrum is shown in the lower left box. In this case, the selected time window is [45-63] ns. The frequency spectrum with maximum energy in this interval is used to compute f_r and σ_R^2 .

When saving the data, the amplitudes or centroid frequencies are reduced to pseudo travel times (τ in Eq. 2.9 or Eq. 2.15). To perform this transformation, the length of the rays must be known. Straight rays are computed by default. However, it is possible to load rays that were traced after the inversion of travel times by pressing the appropriate button (see Fig. 3.22). During the saving process, a small utility is launched to view and manually adjust A_0 or f_S . Prior to saving, it is possible to discard the traces for which the ray are departing or arriving with small angles at the antennas. This is done by adjusting the minimum angle value in the specified box. If curved rays are loaded, the traces for which the first arrival is refracted on surface are also rejected. If this is not desirable, the user must specify a value of "Z surface" higher than the highest elevation attained by the rays.

Note that in `bh_tomo_amp`, each trace datum can be weighted by the inverse of the signal-

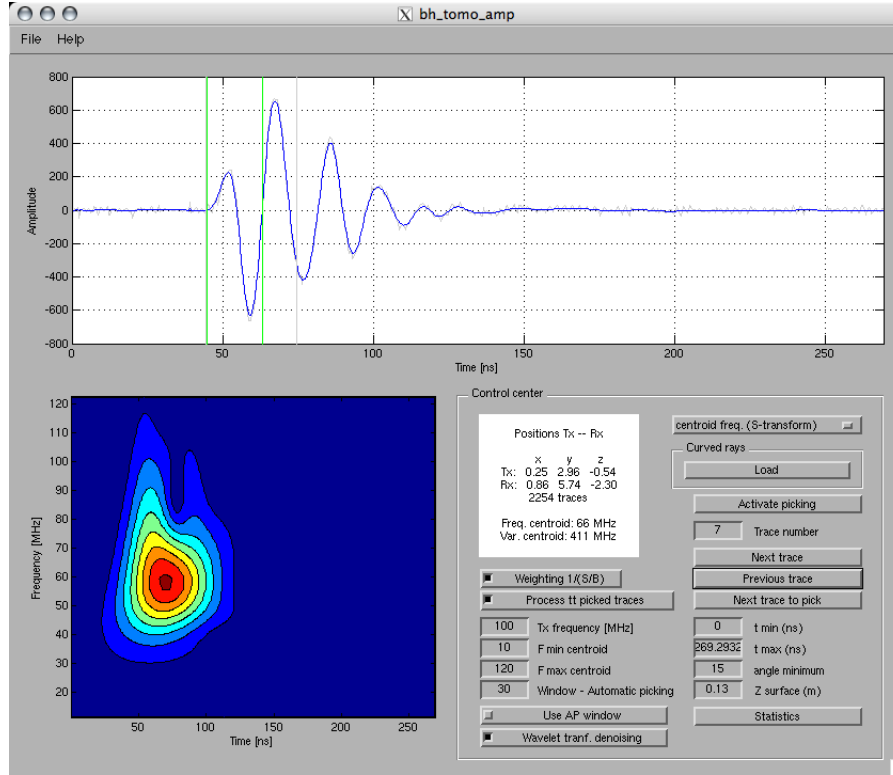


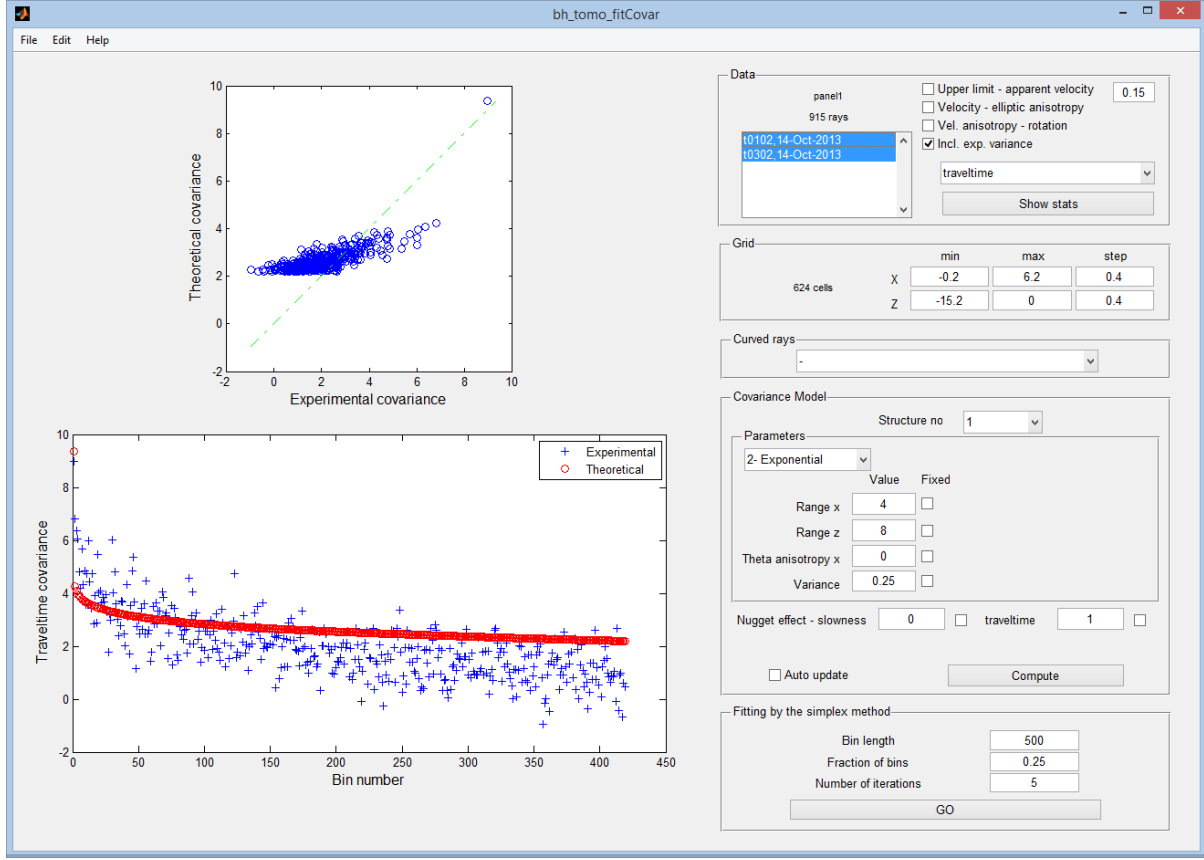
Figure 3.22: Interactive screen of `bh_tomo_amp`.

to-noise ratio (S/N) by selecting the **Weighting 1/(S/N)** button. In such case, the inverse of S/N is taken as the standard deviation in the covariance model. However, there is no physical meaning to this so-called standard deviation, and our experience showed that this feature does not produce good results. At this point, it is therefore not recommended to select this option.

3.5 Fitting the model covariance

Once the travel times are picked, a slowness covariance model ($\text{Cov}(s, s)$) must be chosen to perform the geostatistical inversion. The covariance $\text{Cov}(s, s)$ is obtained through Eq. (2.24). However, the covariance $\text{Cov}(t^*, t^*)$ is not stationary. Therefore, estimators such as traditional variogram or covariogram cannot be used directly, and the model parameters for slowness covariance must be estimated by inversion. The program `bh_tomo_fitCovar` is designed to accomplish this task interactively using the concept of the V-V plot described in [Asli et al. \(2000\)](#), or automatically using a minimization scheme based on the Simplex method implementation of Matlab. In the V-V plot approach, the theoretical covariance values are sorted in decreasing order and binned in classes for which the mean value is calculated. The same sorting order and binning scheme is applied to the experimental data. Thus, a value corresponding roughly to a single theoretical structural distance can be calculated for an experimental variogram. If the chosen slowness covariance model is appropriate, the experimental variogram versus theoretical variogram plot (the V-V plot) should show a dispersion around a line at 45° .

Contrary to `bh_tomo_tt` and `bh_tomo_amp` in which the MOG data are processed independently, the calculations made in `bh_tomo_fitCovar` are done with the entire data of a

Figure 3.23: Interactive screen of `bh_tomo_fitCovar`.

panel or super-panel. If the panel contains more than one MOG dataset, they are merged together. The interface of `bh_tomo_fitCovar`, shown in Fig. 3.23, comprises two boxes where the experimental covariance, $\text{Cov}(t^*, t^*)$, can be visually compared with the theoretical model $\mathbf{L} \text{Cov}(s, s) \mathbf{L}^T + \mathbf{C}_0$. The upper box is the V-V plot, whereas the sorted binned covariances are shown in the lower box. The covariance model parameters can be set manually, or, as stated previously, adjusted automatically using a minimization scheme based on a Simplex method implementation. The covariance model can include up to two elementary structures and a nugget effect on slowness. A nugget effect on traveltime can also be specified. Note that in order to speed up the calculations, bins with low covariance values can be discarded by giving a value below 1 for the fraction of bins (typically 0.25 to 0.5 is sufficient). Note also that the ray matrix \mathbf{L} can be given in input, for example in the case where amplitude data are processed. When processing time data, matrix \mathbf{L} is not known unless a prior inversion has been carried out, for example using only straight rays. However, our experience indicates that generally, using straight rays yields adequate estimates of the covariance model.

3.6 Inversion

The program covered in this section is `bh_tomo_inv`, in which the tomographic inversion is performed. The interface is shown in Fig. 3.24. Both travel time and amplitude data can be inverted, provided that proper processing has been carried out. As a mean of control, the

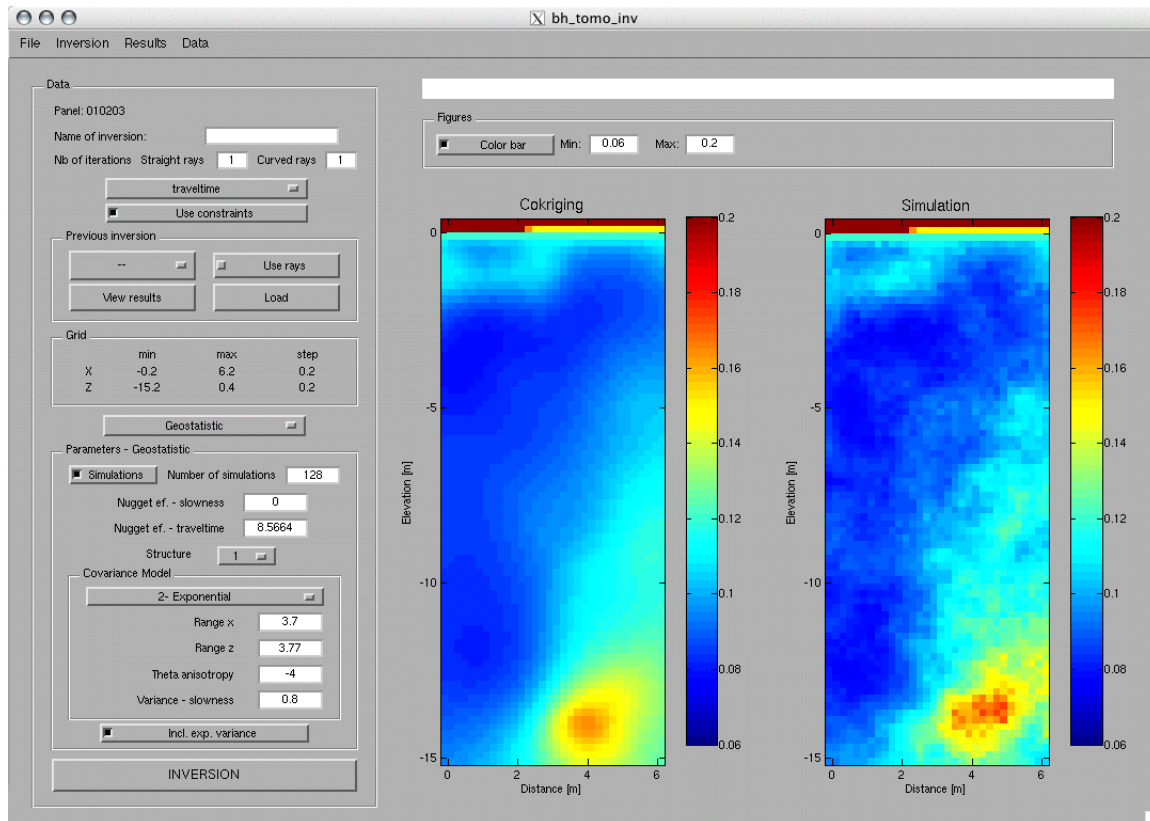


Figure 3.24: Interactive screen of `bh_tomo_inv`. Two velocity models obtained with the data presented in section 4 are displayed. The leftmost model is obtained by cokriging and the rightmost model is the simulation that best fits the data.

travel time and amplitude data can be viewed using the Data menu items. Various parameters affecting the inversion can be set, notably the use of the model constraints and the use of previously calculated rays. The choice between geostatistical and LSQR algorithm is also made in `bh_tomo_inv`. Depending on the type of algorithm, the appropriate parameters that can be adjusted are displayed in the bottom left frame. The parameters for LSQR are described in appendix B. In the case of geostatistical inversion, only the covariance model needs to be given (it is loaded automatically if `bh_tomo_fitCovar` has been used). The covariance model parameters can be modified by the user. When performing LSQR inversion, a starting model can be built interactively using a GUI accessible from the Inversion-Initial model menu item. In the case of geostatistical inversion, the starting model is always homogeneous.

The number of straight and curved rays iterations is given in the upper part of the Data pane. When performing geostatistical inversion of traveltime data, one iteration of each usually suffice. When geostatistical inversion of amplitude data is performed, curved rays previously obtained from traveltime inversion can be used. The list of previous inversions saved in the database appears in the upper left popup menu of the “Previous inversion” pane. The user can select which inversion should be used by choosing the appropriate item in the list and activate the “Use rays” checkbox. Note that when performing geostatistical inversion of amplitude data, the ray matrix should not be updated, i.e. the number of curved ray iterations must always be equal to 0.

The inversion is started by pressing the **Inversion** button. After completion, the results can be saved in the database for later retrieval, using the File-Save menu item, or exported in a separate mat-file. The format of this file is given in appendix A.4. The inversion parameters are saved together with the results. The inversion results and ray coverage can be viewed in separate windows by calling the appropriate Results menu item. When geostatistical simulations are carried out, all simulated fields are saved after completion of the inversion. If no constraints are imposed, the “best” simulated field is the one obtained after gradual deformation of all simulated fields. In the gradual deformation method, the realizations are combined sequentially, two at a time, with weights chosen as to maximize a goodness of fit criterion (Hu, 2002). This method has proven its efficiency in petroleum applications. In the case where constraints are applied, the gradual deformation scheme cannot be used directly because one working hypothesis of the method is that the simulated fields have a Gaussian distribution with zero mean. When the properties of air (above the surface) are fixed, the distribution of the properties is no longer Gaussian. In this case, the “best” simulated field is the one that minimizes the L_1 norm of the residuals.

3.7 Estimating the saturation of the magnetic nanofluid

3.7.1 Background

Tomographic measurements allow mapping the velocity and attenuation of radio frequency EM waves in the ground. The attenuation and velocity are expressed in terms of physical properties by

$$\alpha = \omega \sqrt{\mu \epsilon} \sqrt{\frac{1}{2} \left[\sqrt{1 + \left(\frac{\sigma}{\omega \epsilon} \right)^2} - 1 \right]}, \quad (3.2)$$

$$v = \left\{ \sqrt{\mu \epsilon} \sqrt{\frac{1}{2} \left[\sqrt{1 + \left(\frac{\sigma}{\omega \epsilon} \right)^2} + 1 \right]} \right\}^{-1}, \quad (3.3)$$

where ω is the angular frequency, μ is the magnetic permeability, ϵ is the dielectric permittivity and σ is the electric conductivity. The injection of the magnetic nano fluid will change the physical properties of the reservoir rock and thus affect the velocity and attenuation of the EM wave propagating in that reservoir. In this section we show how can EM measurements allow mapping and retrieving the quantity of injected nano fluid.

Using a time-lapse approach, the available data are v_1 , α_1 , v_2 and α_2 , where subscript 1 indicates baseline data and subscript 2 repeat data. The idea is to combine the time-lapse data and petrophysical relationships to infer the saturation of the magnetic nano fluid at the time of the repeat survey. The saturation, noted S , is defined as the volumetric proportion of the magnetic fluid.

A number of assumptions must be used to infer S . We first assume that there is no magnetic material in the earth at baseline, i.e. μ_1 is equal to the magnetic permeability of vacuum (μ_0). We also assume that the physical properties of the rock are known: the permittivity of the rock matrix ϵ_m is known as well as the Archie parameters a and m are known. Finally, the properties of the nano fluid (σ_{nf} , ϵ_{nf} and μ_{nf}) are also assumed to be known. It is also possible to assume that the properties of the reservoir fluid are known, e.g. from well sampling; if this is not the case, the porosity of the reservoir must be known.

In the case where the porosity is known, we can infer the properties of the fluid in place before injection (σ_{f_1} and ϵ_{f_1}) from v_1 and α_1 . First get the values for the rock-fluid mixture (σ_1 and ϵ_1):

$$\epsilon_1 = \frac{1}{\mu_0} \left(\frac{1}{v_1^2} - \frac{\alpha_1^2}{\omega^2} \right); \quad (3.4)$$

$$\sigma_1 = \frac{2\alpha_1}{\mu_0 v_1}. \quad (3.5)$$

Using Archie's equation (Schön, 2004), the conductivity of the fluid is

$$\sigma_{f_1} = a\sigma_1\phi^{-m}. \quad (3.6)$$

The permittivity of the rock-fluid mixture is generally described by the self-similar model of Sen et al. (1981)

$$\frac{\epsilon_1 - \epsilon_m}{\epsilon_{f_1} - \epsilon_m} \left(\frac{\epsilon_{f_1}}{\epsilon_1} \right)^\Psi = \phi, \quad (3.7)$$

where Ψ is the depolarization factor, assumed equal to 1/3. This equation must be solved numerically. The self-similar model is also used to compute the conductivity and permeability of the rock-fluid mixture (Mavko et al., 2009). When the properties of the fluid are known, the porosity can be calculated using (3.6) and (3.7) and averaged.

After injection, the magnetic permeability of the medium is no longer equal to μ_0 , and we have

$$\epsilon_2 = \frac{1}{\mu_2} \left(\frac{1}{v_2^2} - \frac{\alpha_2^2}{\omega^2} \right); \quad (3.8)$$

$$\sigma_2 = \frac{2\alpha_2}{\mu_2 v_2}. \quad (3.9)$$

There are thus three unknowns (σ_2 , ϵ_2 and μ_2), but only two equations. Also, the fluid is a mixture of the initial fluid and the magnetic fluid. The aim is to isolate S from these equations and known petrophysical relationships.

The electric conductivity of the fluid mixture after injection can be described by the self-similar model

$$\frac{\sigma_{f_2} - \sigma_{f_1}}{\sigma_{nf} - \sigma_{f_1}} \left(\frac{\sigma_{nf}}{\sigma_{f_2}} \right)^\Psi = S. \quad (3.10)$$

The conductivity of the saturated rock is then

$$\sigma_2 = \frac{1}{a} \sigma_{f_2} \phi^m. \quad (3.11)$$

Because the nano fluid is the only magnetic material, the permeability of the medium after injection is

$$\frac{\mu_2 - \mu_0}{\mu_{nf} - \mu_0} \left(\frac{\mu_{nf}}{\mu_2} \right)^\Psi = S\phi. \quad (3.12)$$

In the case of permittivity, the expression for the fluid mixture is

$$\frac{\epsilon_{f_2} - \epsilon_{f_1}}{\epsilon_{nf} - \epsilon_{f_1}} \left(\frac{\epsilon_{nf}}{\epsilon_{f_2}} \right)^\Psi = S, \quad (3.13)$$

Fluid	σ (S/m)	ϵ (ϵ_o)	μ (μ_o)
nano fluid	0.01	75	3
oil	0.001	2.25	1
brine	1	76	1

Table 3.1: Physical properties of fluids

and the expression for the saturated rock is

$$\frac{\epsilon_2 - \epsilon_m}{\epsilon_{f_2} - \epsilon_m} \left(\frac{\epsilon_{f_1}}{\epsilon_2} \right)^\Psi = \phi. \quad (3.14)$$

Unfortunately, all the expressions above preclude finding a closed-form expression for S . The proposed approach for solving the problem is to use a global optimization algorithm. The constrained simulated annealing is adopted here (Kirkpatrick et al., 1983; Press et al., 1992), where the problem is constrained such that $0 \leq S \leq 1$. The input parameters are the measured data v_1 , α_1 , v_2 and α_2 , as well as the known properties ϵ_m , ϕ , a , m , σ_{nf} , ϵ_{nf} and μ_{nf} . The steps are

1. compute ϵ_{f_1} and σ_{f_1} , or evaluate ϕ .
2. start with a guess value S_0 .
3. iterations:
 - compute ϵ_{f_2} and σ_{f_2} ;
 - compute μ_2 , ϵ_2 and σ_2 ;
 - compute estimates of velocity and attenuation \hat{v}_2 and $\hat{\alpha}_2$;
 - update S with the simulated annealing algorithm.

Implementation and validation

The proposed algorithm has been implemented in matlab and incorporated in `bh_tomo`. The self-similar model is solved with the matlab function `fsolve` which by default uses the trust-region-dogleg algorithm. This function is used by the function `findS.m` that allows estimating the saturation. The function `simulannealbnd` of the global optimization toolbox is at the core of the function.

The code has been validated by testing four scenarios. Two types of rock are considered: carbonates and sandstone, which are distinguished by their relative dielectric permittivity values (respectively 8 and 4.65). For both types of rock, the scenarios where oil and brine are substituted by the magnetic nanofluid are evaluated. In the four cases, the porosity of the reservoir is 20%, the Archie parameters a and m are equal to 0.62 and 1.72 respectively, the depolarization factor Ψ is 1/3, and the calculations are done for a frequency of 50 MHz. The properties of the fluids are given in Table 3.1.

The first test were carried out by feeding the program with the true values of fluids and rock matrix properties, and also assuming that the EM wave velocity and attenuation are exactly known. In all cases, the starting value of S was 0.5. Figure 3.25 show the fit between the estimated saturations and the true saturations. For each case, the norm of relative error ($\delta S = \Delta S/S$, for $S > 0$) is indicated in the sub-figures. The results are excellent, with $\|\delta S\|$

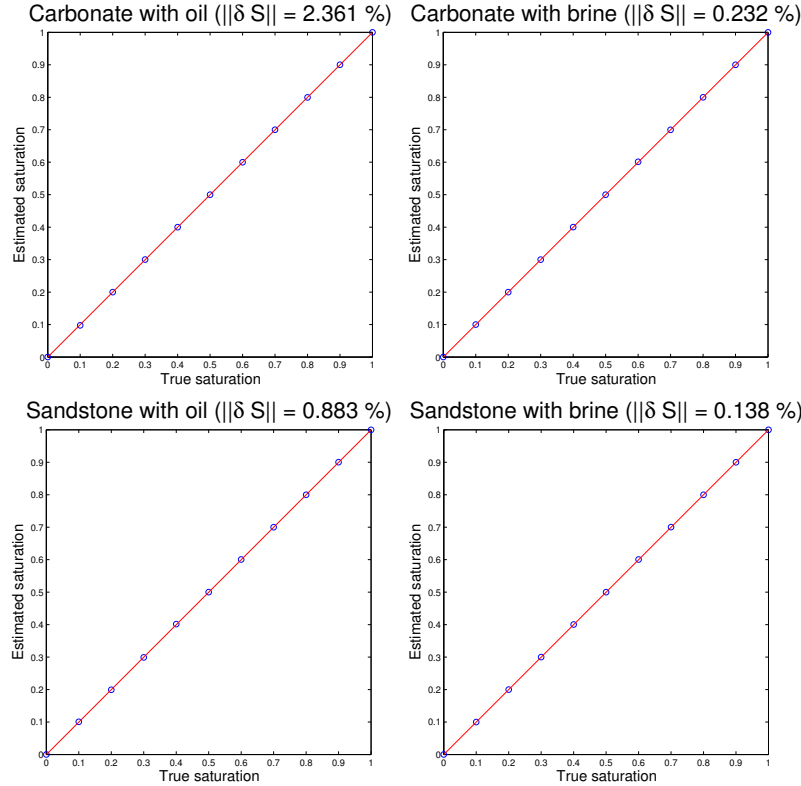


Figure 3.25: Accuracy of the saturation estimates with exact input parameters.

varying between 0.05 and 2.4%. Obviously, in real-life situations the fluid and rock properties will be known only approximately, as will be the case for velocity and attenuation inferred by tomographic measurements. To assess the effect of inaccuracies in the estimation of the latter two parameters, the carbonate/oil scenario above was repeated with v_1 and v_2 underestimated and overestimated by 5%. Feeding the estimation routine with these values results in errors growing dramatically as function of S , as shown in Figure 3.26. The same exercise was repeated with α_1 and α_2 underestimated and overestimated by 20%. Such a large value was chosen because amplitude tomography is much less accurate than traveltime tomography (Giroux and Chouteau, 2006). We can see however in Figure 3.26 that α has less impact on estimating the saturation, which is positive given that we have less grasp on estimating α . Nevertheless, the relative error becomes important for $S > 0.5$ and being in position to rely on accurate maps of α and v appears crucial.

3.7.2 User interface

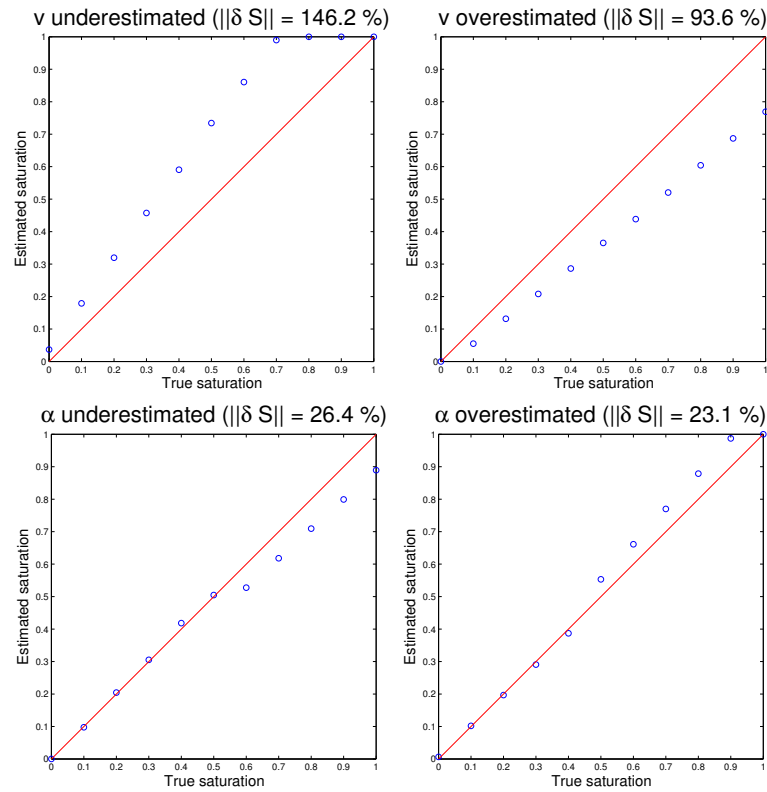


Figure 3.26: Accuracy of the saturation estimates for inaccurate velocity and attenuation input data.

Example

The following example succinctly illustrates the capabilities of the package. The data presented here were collected on a hydraulic structure made of concrete affected by the alkali-aggregate reaction. This reaction, which causes the expansion of concrete and the damaging of the structure, is governed partly by the water content in the concrete. Hence, the purpose of the survey was to estimate the water content in the structure. Various models exist that relate the volumetric water content to the dielectric constant κ . The specific objective was therefore to draw the map of κ between six boreholes drilled along the structure.

Data were collected in the six boreholes, but only a subset acquired in three holes is presented. A schematic cross-section of the surveyed structure and the borehole configuration is shown in Fig. 4.1. The survey was done with a RAMAC system from Malå, with antennas having a central frequency of 100 MHz. VRP and cross hole measurements were performed. The data is spreaded in six files (4 VRPs and 2 cross hole). Examination of the data revealed a high attenuation within the probed structure. This implies that in order to obtain an accurate estimate of κ , both velocity and attenuation need to be performed (see Eq. (2.6)).

VRP and cross hole data were used for velocity tomography. Both arrival times and standard deviations were picked and used for the inversion. The velocity model was constrained on surface, where the velocity was fixed to 0.3 m/ns. The velocity models obtained from geostatistical inversion is shown in Fig. 3.24: the leftmost model is obtained by cokriging and the rightmost model is the simulation that best fits the data. The ray coverage is shown in Fig. 4.2. Only one curved ray iteration was done. The overall structure seen in Fig. 4.1 is recognized in the velocity model.

To perform attenuation tomography, only cross hole data were used, and rays refracted on surface were discarded due to the difficulty in correcting the amplitude (Tronicke and Knoll, 2005). Rays that were kept for attenuation tomography are drawn in black in Fig. 4.2. The amplitude data were processed with both the amplitude ratio and centroid frequency methods. In the case of the amplitude ratio method, the radiation patterns of the antennas (Θ_{Tx} and Θ_{Rx}) are approximated by a sine function $\sin(\theta)$ where θ is the angle between the antenna and the ray. In the case of the centroid frequency down-shift method, the spectrum were calculated using the procedure relying on the S-transform described above. Also, with this technique, α_0 must be multiplied by the central frequency of the antenna to obtain α (see Eq. (2.11)). However, it is known that coupling of the antenna with earth materials decreases the central frequency of the received signal (Zhou et al., 2001). Based on the measured data, a value of 80 MHz was chosen instead of 100 MHz.

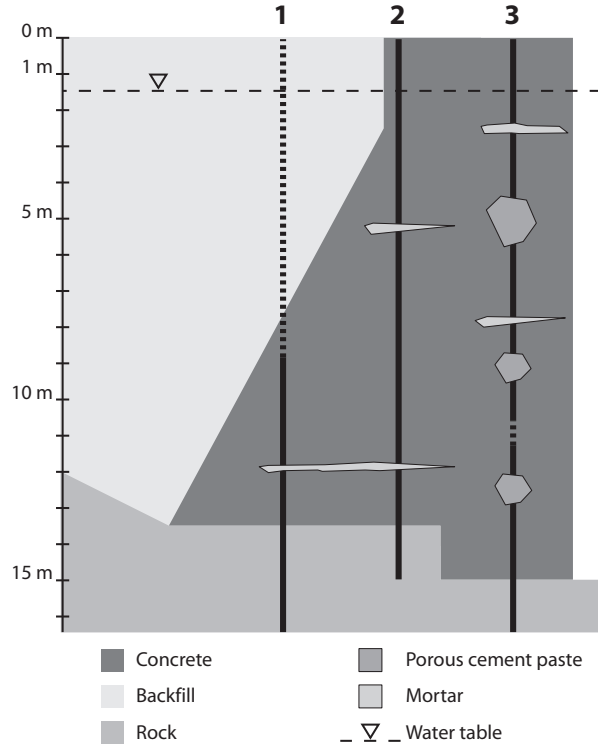


Figure 4.1: Schematic cross-section of the surveyed structure interpreted from boring logs.

For both datasets, the inversion parameters were the same: identical covariance model, model constrained on surface ($\alpha = 0$ above the ground), and 128 simulations to perform. The results of the simulations are shown in Figs. 4.3 and 4.4. These figures show the simulation that best fits the reduced data (L_1 norm), as well as the standard deviation of the model parameter computed from the 128 simulated attenuation fields. Both datasets yield very similar results, although the average attenuation is slightly lower in the case of the centroid frequency dataset.

Dielectric constant and electric conductivity maps computed with equations (2.6) and (2.7) are shown in Fig. 4.5. The best-fit simulation results were used, and attenuation obtained with the centroid frequency down-shift method was selected. These maps were constructed outside `bh_tomo`, after the data have been exported from `bh_tomo`. They show different anomalies, especially in the backfill, at the backfill/concrete interface, and between Holes 2 and 3.

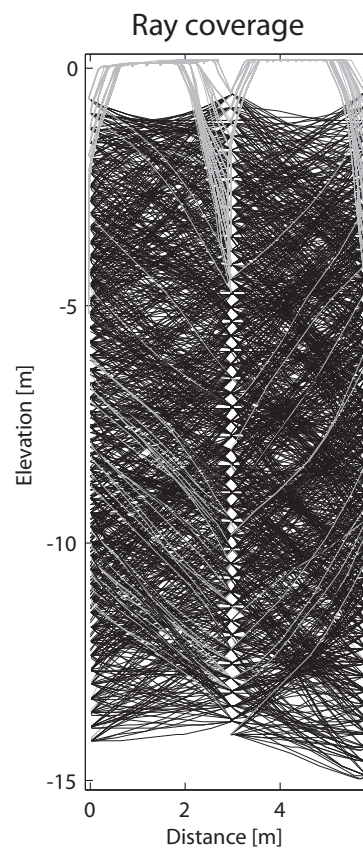


Figure 4.2: Ray coverage for the velocity model shown in Fig. 3.24. Grayed out rays were excluded for attenuation tomography.

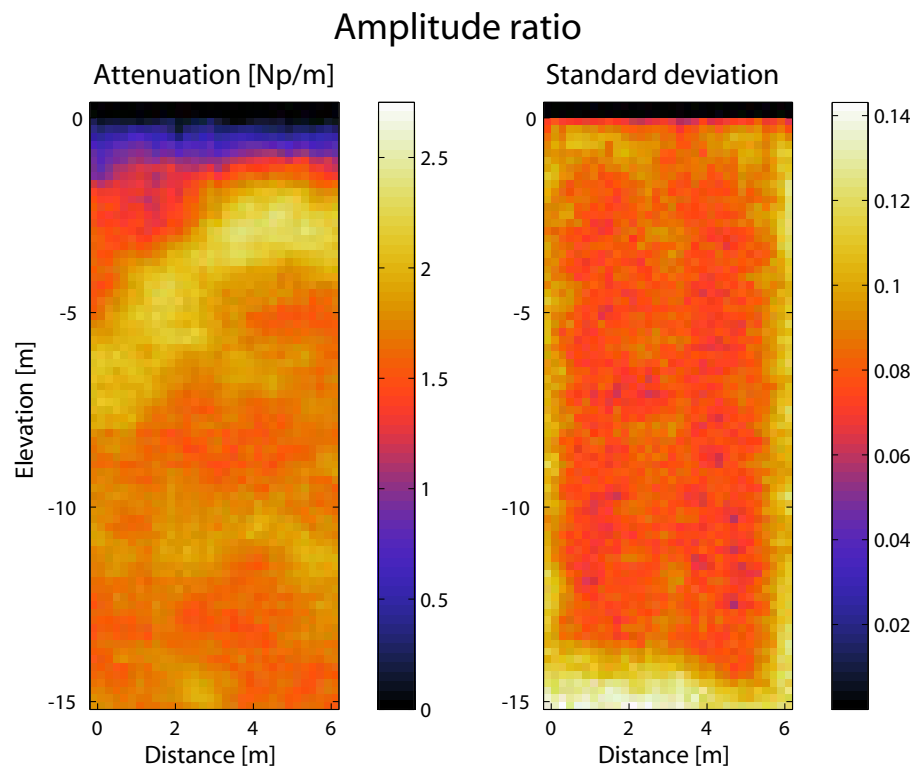


Figure 4.3: Simulation results of the geostatistical inversion of amplitude data, amplitude ratio method. On the left: simulation that best fits the data. On the right: standard deviation inferred from the 128 simulations.

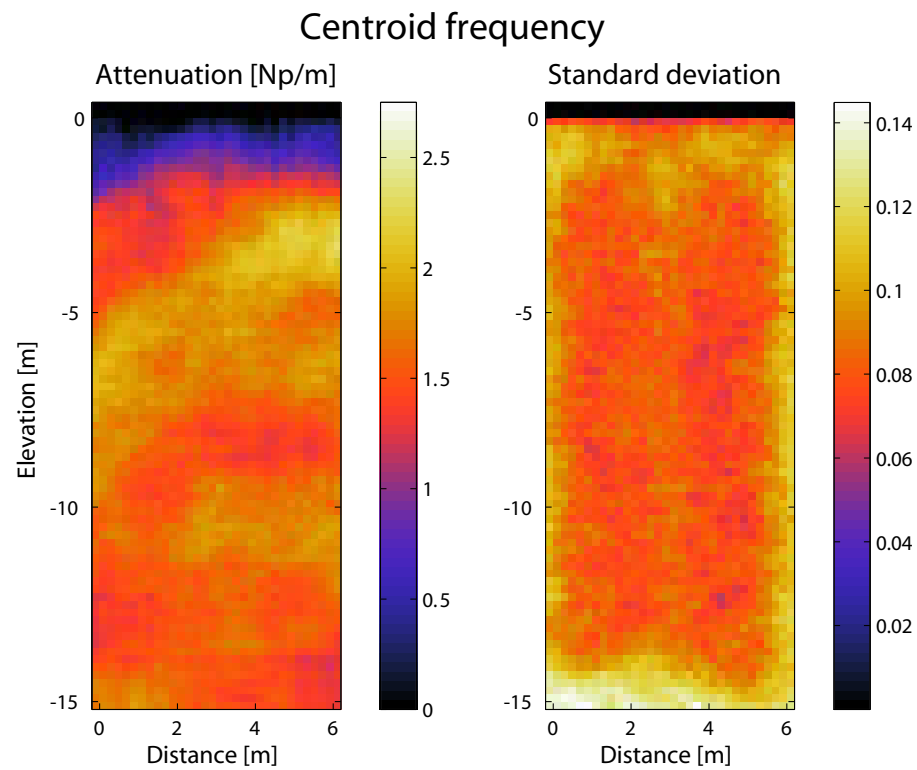


Figure 4.4: Simulation results of the geostatistical inversion of amplitude data, centroid frequency down-shift method. On the left: simulation that best fits the data. On the right: standard deviation inferred from the 128 simulations.

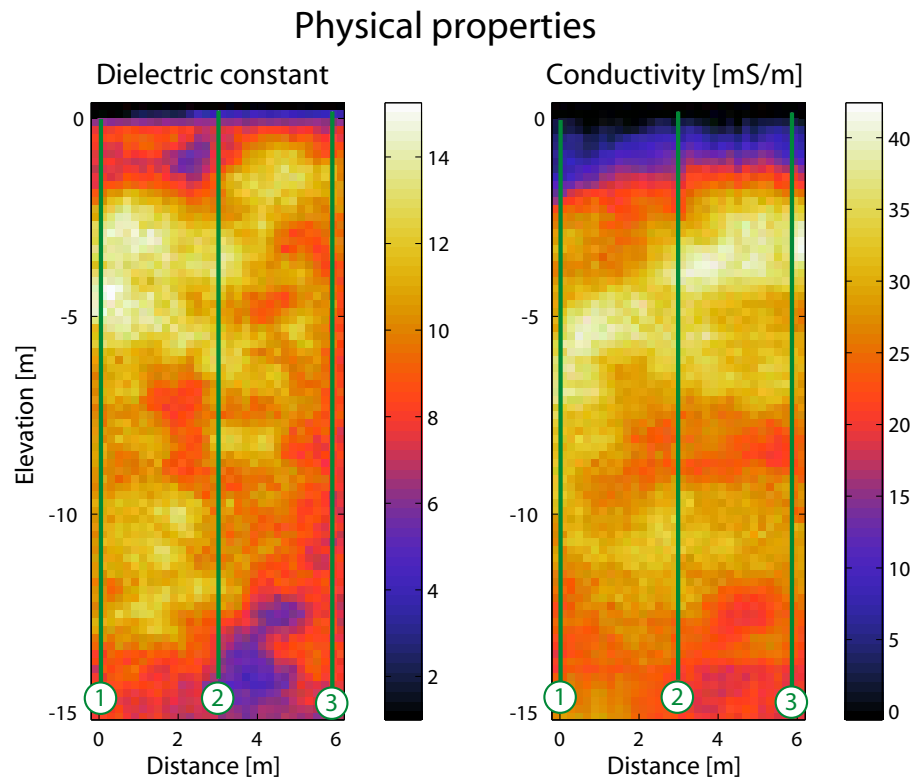


Figure 4.5: Maps of dielectric constant and effective electric conductivity inferred from the tomographic results. Boreholes 1, 2 and 3 are drawn in green.

Exchange files formats

`bh_tomo` modules can import or export various data in different formats. Data that can be imported are picked travel times (in `bh_tomo_tt`), and slowness or attenuation constraints (in `bh_tomo_constraints`). Data that can be exported are the inversion results (from `bh_tomo_inv`). Note that all data are stored in the mini database which is one large mat-file that can be loaded at any time in Matlab.

A.1 Aramco's Magnetic NanoMappers data

A.1.1 Motivation for the format

The purpose of the format is to have all information needed to perform traveltime and amplitude tomography easily accessible. A single `struct` variable is used, with appropriate fields for each parameter. The data can then be imported in matlab with a single `load` command. The name of the `struct` variable is chosen to be `data`.

Definitions

`bh_tomo` was first developed for processing borehole ground-penetrating-radar data. Many parameters are related to borehole GPR antennas and data acquisition. Figure 3.6 illustrates two common survey modes, namely the multi-offset gather and vertical radar profiling.

Description of the fields

Table A.1 lists the fields of the `struct` variable and gives a short description.

Note that the date can be edited in the database GUI (`bh_tomo_db`).

Important For data acquired in a well (MOG survey), the Z coordinate of transmitter and receiver are the *depth* (positive downward) from the collar of the well, or kelly bushing or any other reference elevation on surface. For VRP survey configuration, the Tx is on surface and moved away from the well head (where receivers are) along a line pointing toward a second well. The Z coordinates of the Tx correspond to the distance from well, as shown in Figure 3.7.

Name	Size	Description
ntrace	1	Number of traces in the file
nptsptrc	1	Number of points per trace
timec	1	Sampling period
timestp	1×nptsptrc	Time axis data
rdata	nptsptrc×ntrace	Trace data
Tx_x	1×ntrace	X coordinate of transmitter
Tx_y	1×ntrace	Y coordinate of transmitter
Tx_z	1×ntrace	Z coordinate of transmitter
Rx_x	1×ntrace	X coordinate of receiver
Rx_y	1×ntrace	Y coordinate of receiver
Rx_z	1×ntrace	Z coordinate of receiver
rstepsz	1	Distance along well between two measurements
TxOffset	1	Offset between antenna head and feed point
RxOffset	1	Offset between antenna head and feed point
antennas	string	Antenna type
rnomfreq	1	Nominal frequency of the antennas (MHz)
synthetique	1	0 for real data, 1 for modelled traces
cunits	string	Distance units
tunits	string	Time units
csurvmod	string	Survey Mode, either 'MOG' or 'VRP'
date	string	Date of data acquisition in the field

Table A.1: Fields of the `struct` variable

In the current 2D version, X and Y coordinates are not used. The coordinates are projected along the well trajectory. If one *does not want* the coordinates to be interpolated, a string variable comment equal to 'true positions' should be added to data.

An example of the content of the `struct` variable is given below:


```

>> data

data =

    ntrace: 945
   nptsptrc: 512
    timec: 0.8012
   timestp: [1x512 double]
    rdata: [512x945 double]
    Tx_x: [1x945 double]
    Tx_y: [1x945 double]
    Tx_z: [1x945 double]
    Rx_x: [1x945 double]
    Rx_y: [1x945 double]
    Rx_z: [1x945 double]
   rstepsz: 0.2493
  TxOffset: 0.6650
  RxOffset: 0.6650
  antennas: '100 MHz Borehole - Ramac'
   rnomfreq: 100
synthetique: 0
    cunits: 'm'
    tunits: 'ns'
   csurvmod: 'SURVEY MODE' = Trans. - MOG'
    date: '15-Oct-2012'

```

The first trace can be displayed by issuing the command at the matlab prompt:
`plot(data.timestp,data.rdata(:,1)).`

A.2 Picked arrival times

Arrival times picked in a third party software can be imported in `bh_tomo_tt`. The picked times must correspond to the multi-offset gather data file being processed. They must be stored in an ASCII file, formatted in two or three columns. The first column is the trace number in the current MOG data file and the second column is the picked time. An optional third column contains the standard deviation of the corresponding arrival time.

A.3 Model constraints

Similarly to the arrival times, constraints in the velocity or attenuation models can be imported in `bh_tomo_constraints` (this program is accessible from `bh_tomo_db` via the grid utilities). The constraints must also be stored in an ASCII file, in three columns: the first column contains the elevations and the second column contains the horizontal distances relative to the grid origin where to insert the constraint values, which are included in the third column.

A.4 Inversion results

The results of inversion runs can be exported in a Matlab mat-file from `bh_tomo_inv`. This file contains two structured variables: `param` and `tomo`. The fields of the variable `param` correspond to the inversion parameters, whereas the fields of the variable `tomo` are the inversion results. The fields of `tomo` are detailed in Table A.2. Note that some fields might not be present, depending on the type of inversion carried out, i.e. LSQR vs geostatistical, with or without simulations.

Table A.2: Description of the fields of the structured variable `tomo`

Name	Type	Size	Description
<code>rais</code>	cell array	$1 \times n_o$	Each element of the cell array contains a $n \times 3$ matrix of the $n - 1$ ray segments coordinates.
<code>L</code>	matrix	$n_o \times n_p$	Matrix <code>L</code> of Eq. (2.3)
<code>no_trace</code>	matrix	$n_o \times 1$	Trace number in the database for each of the n_o observations
<code>s</code>	matrix	$n_p \times 1$	Slowness or attenuation field
<code>simu</code>	matrix	$n_p \times n_{simu}$	Simulated slowness or attenuation field
<code>sgr</code>	matrix	$n_p \times 1$	Simulated slowness or attenuation field after gradual deformation
<code>x</code>	matrix	$n_x \times 1$	Horizontal coordinates of the center of the grid cells
<code>z</code>	matrix	$n_z \times 1$	Elevation coordinates of the center of the grid cells
<code>lmoy</code>	scalar	–	Mean slowness or attenuation
<code>diff1</code>	matrix	$1 \times n_{simu}$	Fit of the simulated slowness or attenuation fields to the data
<code>diff1_min</code>	scalar	–	Number of the simulated slowness or attenuation field that fits best the data
<code>dt</code>	matrix	$n_o \times 1$	Data (centered)

LSQR inversion parameters

The parameters that can be adjusted by the user when using the LSQR algorithm are defined in Table B.1.

Table B.1: Description of the LSQR inversion parameters

Name	Description
Tolerance	Iterations stopped if L_2 norm of the residuals lower than Tolerance.
alpha	Lagrange multiplier of the smoothing operator (second derivative).
gradmin	Iterations stopped if gradient of the L_2 norm of the residuals lower than gradmin.
correlmin	Iterations stopped if correlation of models between iterations is higher than correlmin.
Nb of iterations	Maximum number of iterations.
Max vel. variation per it.	Maximum variation of velocity in percent that is allowed at each iteration. If exceeded, the velocity perturbation is rescaled to this maximum.

Bibliography

- Arcone, S. A. (1991). Dielectric constant and layer-thickness interpretation of helicopter-borne short-pulse radar waveforms reflected from wet and dry river-ice sheets. *IEEE Transactions on Geosciences and Remote Sensing*, [29\(5\):768–777](#). 36
- Asli, M., Marcotte, D., and Chouteau, M. (2000). Direct inversion of gravity data by cokriging. In Kleingeld, W., editor, *Geostat2000 - Proceedings of the International Geostatistics Congress*, Cape Town. 14, 40
- Bai, C.-Y., Greenhalgh, S., and Zhou, B. (2007). 3D ray tracing using a modified shortest-path method. *Geophysics*, [72\(4\):T27–T36](#). 17
- Beatty, K. S., Perron, G., Kay, I., and Adam, E. (2002). DSISoft—a MATLAB VSP data processing package,. *Computers and Geosciences*, [28\(4\):501–511](#). 7
- Cai, W., Qin, F., and Schuster, G. T. (1996). Electromagnetic velocity inversion using 2-D Maxwell’s equations. *Geophysics*, [61\(4\):1007–1021](#). 9
- Chapman, B., Jost, G., and van der Pas, R. (2007). *Using OpenMP: Portable Shared Memory Parallel Programming*. MIT Press. 20
- Chilès, J.-P. and Delfiner, P. (1999). *Geostatistics: Modeling Spatial Uncertainty*. John Wiley & Sons. 14
- Conroy, J. P. and Radzevicius, S. J. (2003). Compact MATLAB code for displaying 3D GPR data with translucence. *Computers and Geosciences*, [29\(5\):679–681](#). 7
- Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (2009). *Introduction to Algorithms*. MIT Press, 3rd edition. 18
- Ernst, J., Maurer, H., Green, A., and Holliger, K. (2007). Full-waveform inversion of crosshole radar data based on 2-d finite-difference time-domain solutions of maxwell’s equations. *Geoscience and Remote Sensing, IEEE Transactions on*, [45\(9\):2807–2828](#). 9
- Giroux, B. and Chouteau, M. (2006). On the processing of borehole georadar data for ray-based attenuation tomography. In Daniels, J. J. and Chen, C.-C., editors, *Proceedings of the Eleventh International Conference on Ground-Penetrating Radar*, number BOR.5, Columbus, Ohio, USA. GPR2006. 46

- Giroux, B. and Chouteau, M. (2010). Quantitative analysis of water-content estimation errors using ground-penetrating radar data and a low-loss approximation. *Geophysics*, [75\(4\):WA241–WA249](#). 11
- Giroux, B., Gloaguen, E., and Chouteau, M. (2004). Geotechnical application of borehole GPR – a case history. In Slob, E. and Yarovoy, A., editors, *Tenth International Conference on Ground-Penetrating Radar*, pages 249–252, Delft, The Netherlands. TUDelft. 7
- Gloaguen, E., Marcotte, D., Chouteau, M., and Perroud, H. (2005). Borehole radar velocity inversion using cokriging and cosimulation. *Journal of Applied Geophysics*, [57\(4\):242–259](#). 7, 13, 14
- Gruber, T. and Greenhalgh, S. A. (1998). Precision analysis of first-break times in grid models. *Geophysics*, [63\(3\):1062–1065](#). 17
- Holliger, K., Musil, M., and Maurer, H. R. (2001). Ray-based amplitude tomography for cross-hole georadar data: a numerical assessment. *Journal of Applied Geophysics*, [47:285–298](#). 11
- Hu, L. Y. (2002). Combination of dependent realizations within the gradual deformation method. *Mathematical Geology*, [34\(8\):953–963](#). 43
- Hubbard, S. S., Peterson, J. E., Majer, E. L., Zawislanski, P. T., Williams, K. H., Roberts, J., and Wobber, F. (1997). Estimation of permeable pathways and water content using tomographic radar data. *The Leading Edge*, [16\(11\):1623–1628](#). 7
- Huisman, J. A., Hubbard, S. S., Redman, J. D., and Annan, A. P. (2003). Measuring soil water content with ground penetrating radar: A review. *Vadose Zone Journal*, 2:476–491. 10
- Irving, J. D., Knoll, M. D., and Knight, R. J. (2007). Improving crosshole radar velocity tomograms: A new approach to incorporating high-angle traveltime data. *Geophysics*, [72\(4\):J31–J41](#). 35
- Josuttis, N. M. (2012). *The C++ Standard Library: A Tutorial and Reference*. Addison Wesley, 2nd edition. 18, 19
- Kirkpatrick, S., Gelatt Jr., C. D., and Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, [220:671–680](#). 45
- Le Ravalec, M., Noetinger, B., and Hu, L. Y. (2000). The FFT moving average (FFT-MA) generator: An efficient numerical method for generating and conditioning gaussian simulations. *Mathematical Geology*, [32\(6\):701–723](#). 16
- Liu, L., Lane, J. W., and Quan, Y. (1998). Radar attenuation tomography using the centroid frequency downshift method. *Journal of Applied Geophysics*, [40:105–116](#). 11
- Marcotte, D. (1991). Cokriging with matlab. *Computers and Geosciences*, [17\(9\):1265–1280](#). 14
- Matarese, J. R. (1993). *Nonlinear Traveltime Tomography*. PhD thesis, Massachusetts Institute of Technology. 17
- Maurer, H., Holliger, K., and Boerner, D. E. (1998). Stochastic regularization: Smoothness or similarity? *Geophysical Research Letters*, [25\(15\):2889–2892](#). 15

- Mavko, G., Mukerji, T., and Dvorkin, J. (2009). *The Rock Physics Handbook*. Cambridge University Press, 2 edition. 44
- Menke, W. (1989). *Geophysical Data Analysis: Discrete Inverse Theory*, volume 45 of *International Geophysics Series*. Academic Press. 13
- Moser, T. J. (1991). Shortest path calculation of seismic rays. *Geophysics*, 56(1):59–67. 17
- Nakanishi, I. and Yamaguchi, K. (1986). A numerical experiment on nonlinear image reconstruction from first-arrival times for two-dimensional island arc structure. *Journal of Physics of the Earth*, 34:195–201. 17
- Nolet, G. (1993). Solving large linearized tomographic problems. In Iyer, H. M. and Hirahara, K., editors, *Seismic tomography: theory and practice*, chapter 20, pages 227–247. Chapman & Hall. 13, 15
- Olsson, O., Falk, L., Forslund, O., and Sandberg, E. (1992). Borehole radar applied to the characterization of hydraulically conductive fracture zones in crystalline rock. *Geophysical Prospecting*, 40:109–142. 7, 11
- OpenMP Architecture Review Board (2012). The OpenMP API specification for parallel programming. URL: <http://openmp.org>, last accessed 2012-08-28. 20
- Paige, C. C. and Saunders, M. A. (1982). LSQR: An algorithm for sparse linear equations and sparse least squares. *ACM Transactions on Mathematical Software*, 8:43–71. 13
- Peterson, Jr., J. E. (2001). Pre-inversion corrections and analysis of radar tomographic data. *Journal of Environmental and Engineering Geophysics*, 6:1–18. 11
- Press, W. H., Teukolsky, S. A., Vetterling, W. T., and Flannery, B. P. (1992). *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 2nd edition. 45
- Quan, Y. and Harris, J. M. (1997). Seismic attenuation tomography using the frequency shift method. *Geophysics*, 62:895–905. 11, 12
- Rucker, D. F. and Ferre, T. P. A. (2004). BGPR.Reconstruct: A MATLAB(R) ray-tracing program for nonlinear inversion of first arrival travel time data from zero-offset borehole radar. *Computers and Geosciences*, 30(7):767–776. 7
- Schön, J. H. (2004). *Physical properties of rocks: Fundamentals and principles of petrophysics*, volume 18 of *Handbook of Geophysical Exploration - Seismic Exploration*. Elsevier, 1 edition. 44
- Sen, P. N., Scala, C., and Cohen, M. H. (1981). A self-similar model for sedimentary rocks with application to the dielectric constant of fused glass beads. *Geophysics*, 46(5):781–795. 44
- Sihvola, A. (2000). Mixing rules with complex dielectric coefficients. *Subsurface Sensing Technologies and Applications*, 1(4):393–415. 10
- Stockwell, R. G., Mansinha, L., and Lowe, R. P. (1996). Localization of the complex spectrum: The S transform. *IEEE Transactions on Signal Processing*, 44(4):998–1001. 39
- Tarantola, A. and Valette, B. (1982). Generalized nonlinear inverse problems solved using the least squares criterion. *Reviews of geophysics and space physics*, 20(2):219–232. 15

- Tronicke, J., Holliger, K., Barrash, W., and Knoll, M. D. (2004). Multivariate analysis of cross-hole georadar velocity and attenuation tomograms for aquifer zonation. *Water Resources Research*, [40:W01519](#). 7
- Tronicke, J. and Knoll, M. D. (2005). Vertical radar profiling: influence of survey geometry on first-arrival traveltimes and amplitudes. *Journal of Applied Geophysics*, [57\(3\):179–191](#). 49
- Turner, G. and Siggins, A. F. (1994). Constant Q attenuation of subsurface radar pulses. *Geophysics*, [59\(8\):1192–1200](#). 10
- Witten, A. (2002). Geophysica: MATLAB-based software for the simulation, display and processing of near-surface geophysical data. *Computers and Geosciences*, [28:751–762](#). 7
- Zhou, B. and Fullagar, P. K. (2001). Delineation of sulphide ore-zones by borehole radar tomography at Hellyer Mine, Australia. *Journal of Applied Geophysics*, [47:261–269](#). 7
- Zhou, C., Liu, L., and Lane, J. W. (2001). Nonlinear inversion of borehole-radar tomography data to reconstruct velocity and attenuation distribution in earth materials. *Journal of Applied Geophysics*, [47:271–284](#). 49