Seismic tomography

We will study travel time tomography (body waves), which can be assimilated to imaging. But one has to be aware that imaging can be interpreted as reflection seismic, velocity reconstruction by the industry (SEG). One can also look at surface wave tomography but these two types of tomography are not part of the same theory. There is no easy way to combine them; the assumptions made are different in each case.

The steps:
- We have observations, represented by dots on the figure below (picking of the different waves: P, S, PP…)

Please see:

- People generate linear regression to give the best fit of these observations
- We obtain models (ex: IASP91, PREM…)
  [Wiechert and Herglotz (GER) found a 1D classical model from travel time curves.]
- We use an inversion technique to get the P & S wave velocity as a function of the depth (see the figure below).
- From these models, we can produce travel time curves again.
- Go back to the observations again and look at the deviation from the model → the scattering around it.

\[ \delta t = T_{\text{obs}} - T_{\text{ref}} \]

We look at the difference between the data and the model and NOT at the travel time itself!

**OBJECTIVE:** find a model (3D) that minimizes \( \delta t \).

\[ T_{\text{obs}} = 3D \text{ structure} + \text{error in source location} + \text{noise (instrument, measurement)} \]

Hypocenter: to, x, y, z (or \( \theta, \varphi, r \))

\[ \Rightarrow \delta t = \delta t_{3D} + \delta t_{\text{mislocation}} + \delta t_{\text{noise}} \]

But can we isolate \( \delta t_{3D} \)?
What people do is combine the model (\( \delta t_{3D} \)) and the source location (\( \delta t_{\text{mislocation}} \)).
The noise is assumed to be white, with a Gaussian distribution.
If we know the 1D model, we can apply Snell’s law to estimate the geometry:

\[ T_{\text{obs}} = \int_{\text{3D ray path}} \frac{1}{v(r)} \, dl \]

But we know that:

\[ \nabla T = \frac{1}{c} \hat{k} \]

If \( c \) changes, the ray path changes. We end up with a non-linear problem. The ray path depends on the answer of the problem.

What do we need to do?

\[ \Rightarrow \text{linearize the inversion} \]

To do so, we use the Fermat’s principle:

We have two kinds of “deviations” from the reference ray:

1. First contribution: effect of changes in velocity \( \Delta v \)
2. Second contribution: effect of change in the ray path

Fermat’s principle says that the 2\textsuperscript{nd} effect is of 2\textsuperscript{nd} order: we can ignore it!
**Linearization of the travel time**

\[
\delta t = T_{\text{obs}} - T_{\text{ref}} = T - T_0 = \int \frac{1}{v(r)} \, dl - \int \frac{1}{v_0(r)} \, dl'
\]

1D ray (unknown)  
Reference ray (known), model

\[
= \int s.\, dl - \int s_0.\, dl'
\]

Because the deviation of the ray is ignored, we only look at the effect of the heterogeneities. It means that in linearizing the travel time, we obtain:

\[
\int s.\, dl \cong \int s_0.\, dl' + 2\text{nd order effect}
\]

Because the deviation of the ray is ignored, we only look at the effect of the heterogeneities. It means that in linearizing the travel time, we obtain:

\[
\delta t = \int (s - s_0).\, dl' = \int \Delta s.\, dl'
\]

Reference ray (known)

In linearizing the problem, we get rid of the unknown ray. We can do our calculation in a reference earth model.

[Rayleigh’s principle is similar]

The travel time tomography is an *iterative process*:
- create 1D model
- ray tracing and get new rays in the model
- update ray geometry
- get the reference ray related to the 3D

\[
\delta t = T_{\text{obs}} - T_{\text{ref}}(3D)
\]

(The reference model does not have to be a 1D model.)

**Linearization of the hypocenter mislocation**

\[
\delta t_{\text{misloc}} = \frac{\partial T}{\partial t_0} \delta t_0 + \frac{\partial T}{\partial \hat{x}} \delta \hat{x} + \frac{\partial T}{\partial \hat{y}} \delta \hat{y} + \frac{\partial T}{\partial \hat{z}} \delta \hat{z}
\]

\[
\delta \hat{\odot} = \frac{\partial \hat{\odot}}{\partial \odot} \delta \odot
\]

with \( t_0 \) the origin time, \((x,y,z)\) the location of the earthquake.
What we want to know is: \[ \Delta s \]
\[ \delta t_o, \delta x, \delta y, \delta z \]

From the travel time tomography, we get the wavespeed model and the estimation of the mislocation of the source.

- If we want to work with the velocity, not the slowness:
  Fermat’s principle \[ -\int \frac{\Delta v}{v_o} dl' \] for small \[ \Delta v = v - v_o \]
  \[ \Delta s \approx -\frac{\Delta v}{v_o^2} \] conversion

- **Inversion problem**: linear relationship between \( \delta t \) and the model.

  ![Diagram](image)

  - Observations
  - Model perturbation (m-\( m_o \))
  - Sensitivity kernel / Frechet derivatives

  \[ d_{i} = \int_{\text{volume}} G_{i}(r) \mu(r) dr \]

  \[ d \otimes = \frac{\partial \otimes}{\partial \otimes} \Delta \otimes \]
  Partial derivatives / Frechet derivatives

  - The first thing to do next is to discretize the problem: do a **parametrization** (not work with a continuous integral).

    \[ \mu(r) = \sum_{k=1}^{M} \gamma_k h_k(r) \]
    \( \gamma_k \): weight
    \( h_k \): basis function

  *For example,*
  1. **Fourier theory, harmonic functions:**
    \[ h_k(r) = \cos(k\theta) \]
    *Just add a weight and we obtain a signal.*
2. Plane wave summation:
\[ \phi = \iiint \Phi(x,y,z,w) e^{i(kx - wt)} \, dk \, dw \]

We inject \( \mu \) into the equation:
\[ d_i = \sum_{k=1}^{M} \left\{ G_i(r) h_i(r) \, dk \right\} \gamma_k \]

Green functions: solution of a point source
\( G_i \) is a Green function for a point perturbation. We need to do a convolution with a point perturbation in order to get the observations.

M: number of model parameters
N: number of observations (data)
in general, \( N \neq M \).

As a consequence, the matrix \( A \) is not square in general.

\[ A = \begin{pmatrix} \vdots \end{pmatrix} \]
\[ m = A^{-1} d \quad \rightarrow \text{we cannot do that!} \]
We have to multiply by $A^T$:

$$A^T d = (A^T A)m$$

- $A^T A$ is a square matrix ($M \times M$)

$$\hat{m} = (A^T A)^{-1} A^T d$$ **Generalized Least Squares inversion**

$$\| \hat{m} - (A^T A)^{-1} A^T d \| = \varepsilon \rightarrow \text{Goal is to minimize } \varepsilon.$$ 

- Usually, we work with 10 million x 10 million matrix. So, we need to use numerical techniques to solve the problem: LSQR, SIRT…

- $d = Am$

$$\delta t = \int \Delta s dl \rightarrow \text{parametrize } \Delta s$$

$$\Delta s = \sum_{k=1}^{M} \gamma_k h_k$$

One way is to take $h_k$ as a series of **cells/blocks**. $h_k$ will have a value for $r$ inside the cell $k$ and will be zero for $r$ outside the cell $k$.

Another way:

- $h_k$ as splines, wavelets
- $h_k$: **spherical harmonics** (in global seismology)

$$h_{klm} = f_k(r) Y_{lm}(\theta, \phi)$$

surface spherical harmonic radial function

**model perturbation:**

$$\mu(r) = \mu(r, \theta, \phi) \sum_k \sum_l \sum_m \left( \alpha_{lm}^m \cos(m\phi) + \beta_{lm}^m \sin(m\phi) \right) f_k(r) P^m(\cos \theta)$$

for gravity Legendre polynomial

$$\sum \gamma_k \times h_k = \mu(r)$$

$$\sum \text{weight} \times \text{basis function} = \text{model}$$
\[ \delta t_i = \int \Delta s \, dl = \sum_{k=1}^{M_{\text{ref}}} \Delta s_k (dl)_k \]

- \( i \): event-station pair
- in a block \( k \)

\[ dl.s = \frac{dl}{\nu} \]

\[ -d = Am \]

with \( A \) the sensitivity matrix containing the path length

\[
\begin{bmatrix}
\Delta s_1 \\
\Delta s_2 \\
\vdots \\
\Delta s_m
\end{bmatrix} =
\begin{bmatrix}
\delta t_1 \\
\delta t_2 \\
\vdots \\
\delta t_m
\end{bmatrix}
\]

each ray gives a row

We have an average wavespeed along the ray. In order to construct a model vector, we need to get data from different rays crossing each other.

\( A \) is a sparse matrix. If we look at one ray:

- \( M \sim 300,000 \)
- \( \sim 20 \) layers
- the ray samples \( \sim 100 \) cells

\( A \) will have only \( \sim 100 \) elements non-zero.

The \textbf{good} thing about sparse matrix is that \( A^T A \) is \( \sim \) diagonal.

The \textbf{problem} is that there are many singularities, which makes the inversion unstable (in that case, we need to add a damping factor or regularize the problem).
One possibility is to not use cells of the same size. Consequently, it reduces the number of cells; the inverse matrix is less singular. Nevertheless, the computation time increases.