Characterization of crustal models for quantitative ground motion estimation

B – Model Design
(large-scale velocity structure)

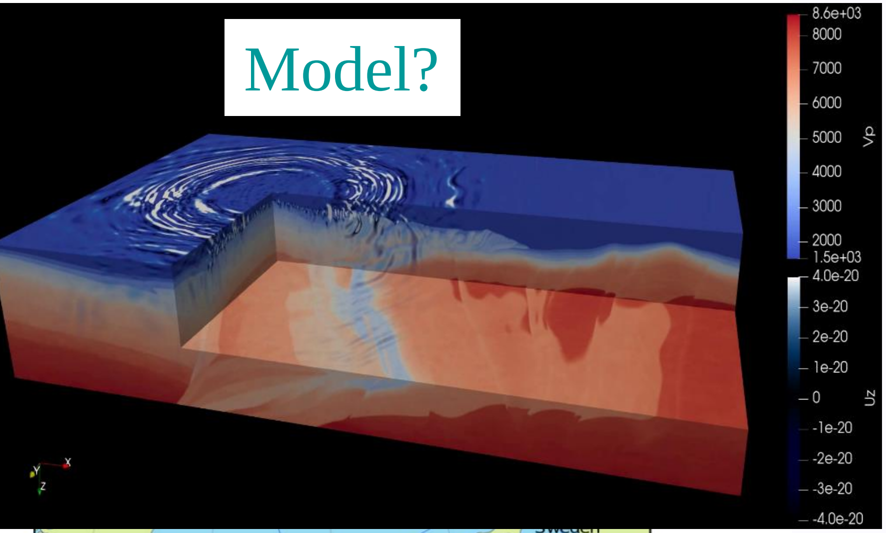
Jean Virieux

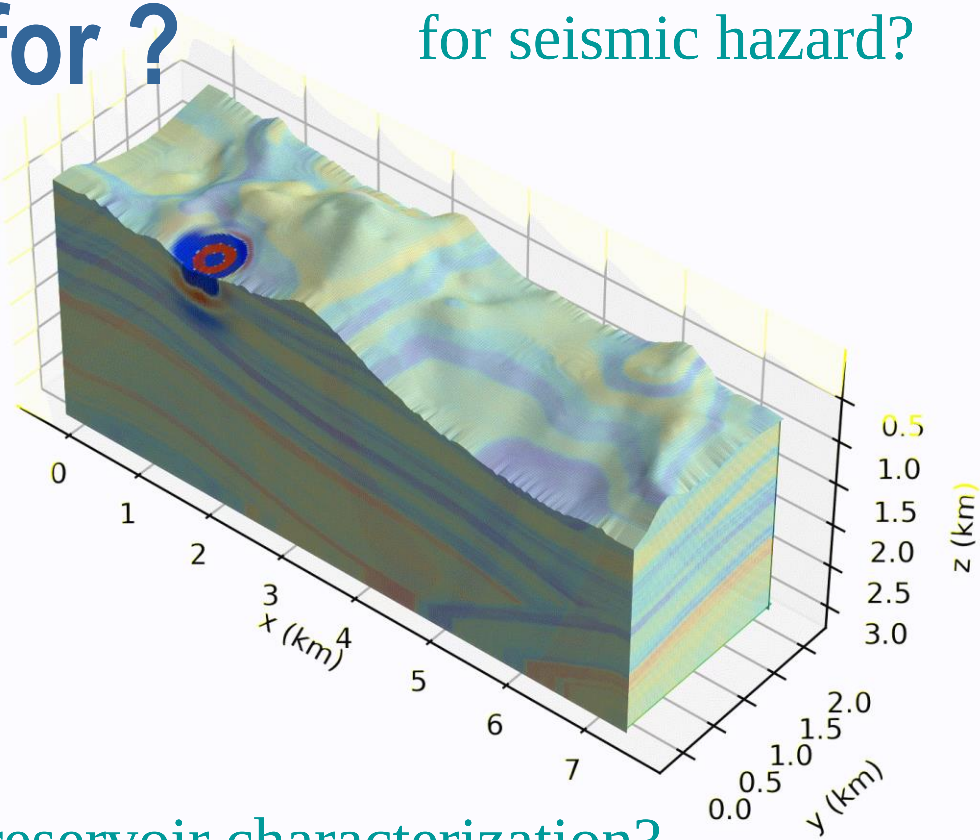
Emeritus Professor at UGA

Some slides are inspired from Seiscope Group
(Present PIs Romain Brossier & Ludovic Métivier)

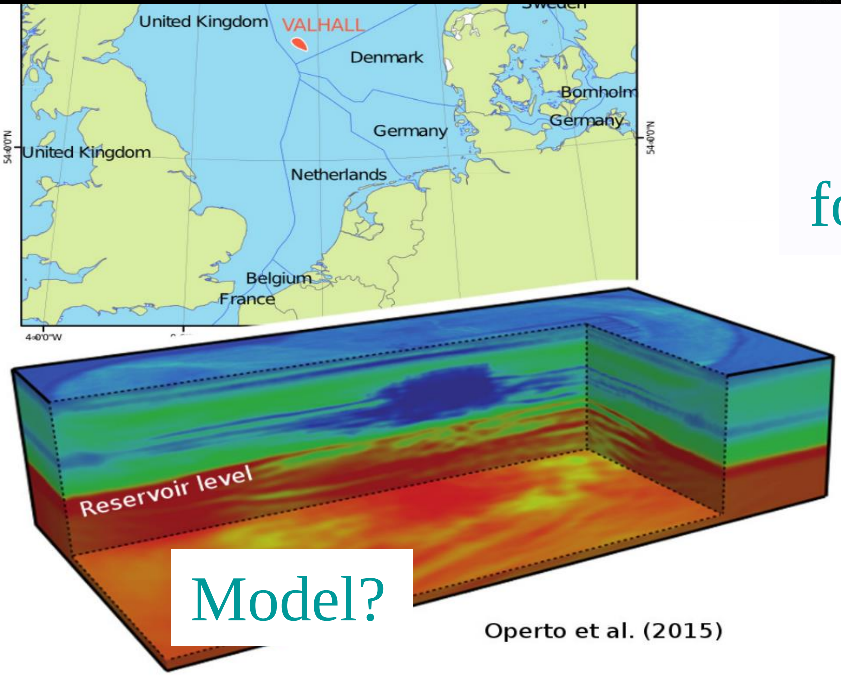
Model Design: what for ?

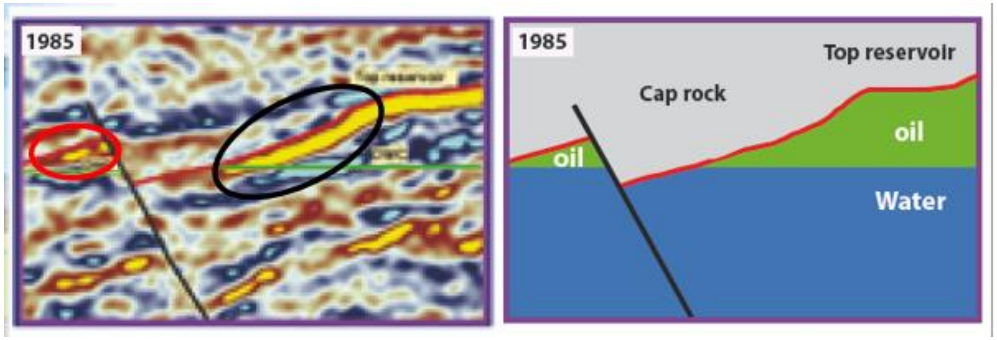
for seismic hazard?





for reservoir characterization?





Reservoir characterization: the goal!

Tracking properties of the subsurface!

Medium upscaling and downscaling

Real medium

Seismic imaging – upscaling

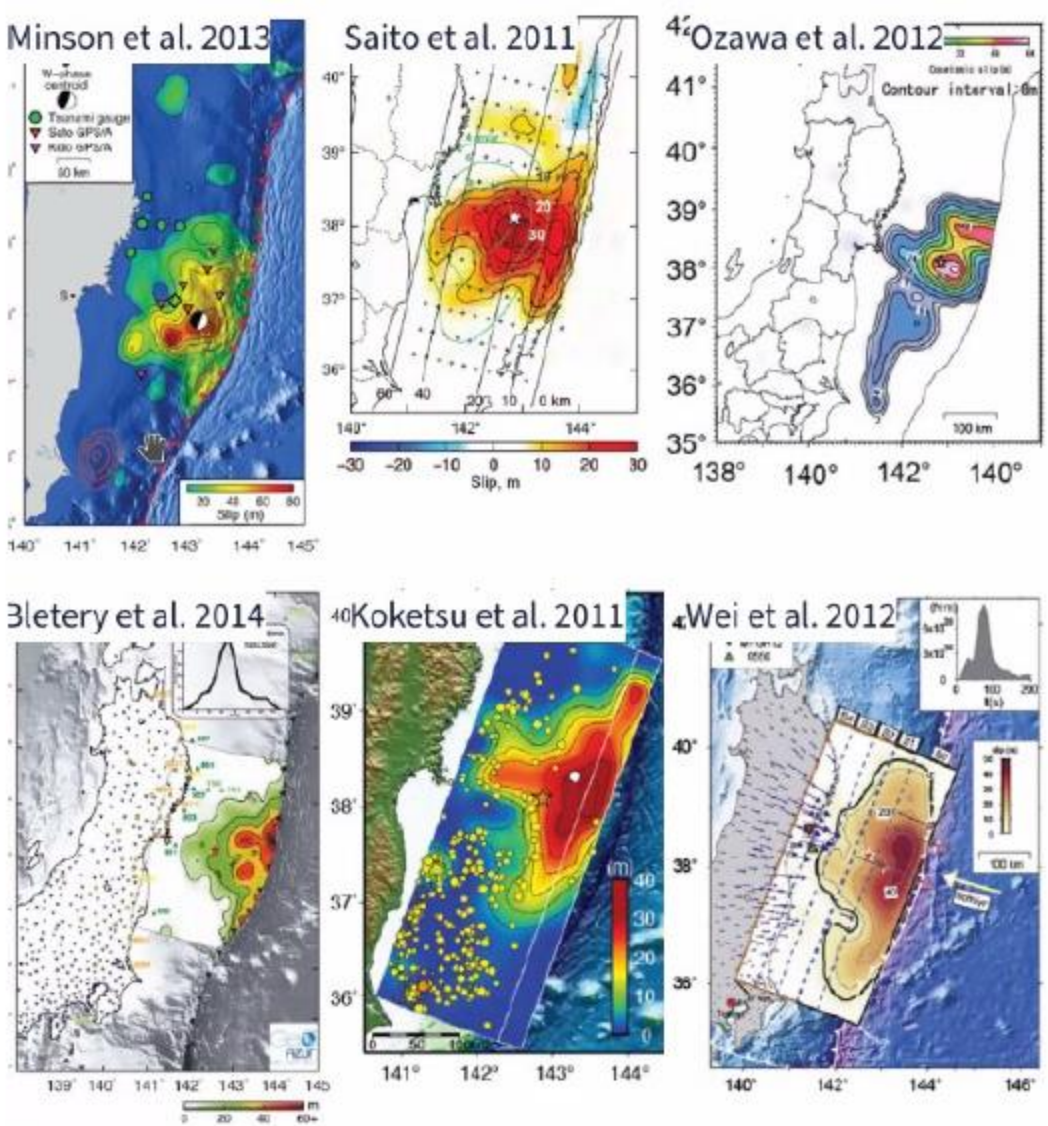
- Seismic finite frequency (sources/receivers)
- Observer effect (acquisition design)
- Attenuation (Q)
- Imaging methods (e.g. travel-time tomography, FWI)
- ...

Traveltime information ($\sqrt{\lambda L}$)

Waveform information ($\frac{\lambda}{2}$)

Interpretation (downscaling)

Seismic rupture investigation: the goal!



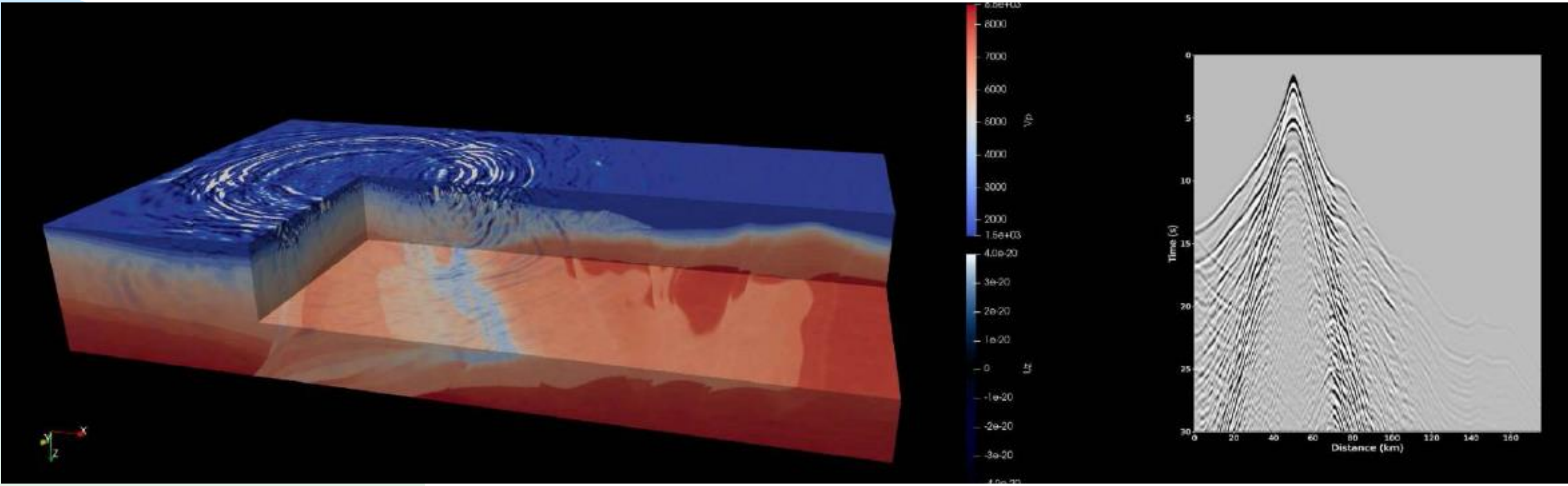
(Ragon, 2023)

2011 Mw 9.1 Tohoku-Oki earthquake, Japan

Rupture physics?

Not the purpose of Engineer Seismology

Ground motion simulation: the challenge!



Model design does not meet realistic description of true medium

Model design should predict somehow a valid (not exactly accurate) ground motion (3 components; free surface influence; subsurface complexity)

Wave propagation tools exist!

https://speed.mox.polimi.it/project/

SPEED: a high performance numerical code for seismic wave propagation

MOX Laboratory for Modeling and Scientific Computing
Department of Mathematics

DICA Department of Civil and Environmental Engineering

POLITECNICO di MILANO
Italy

Many available codes!

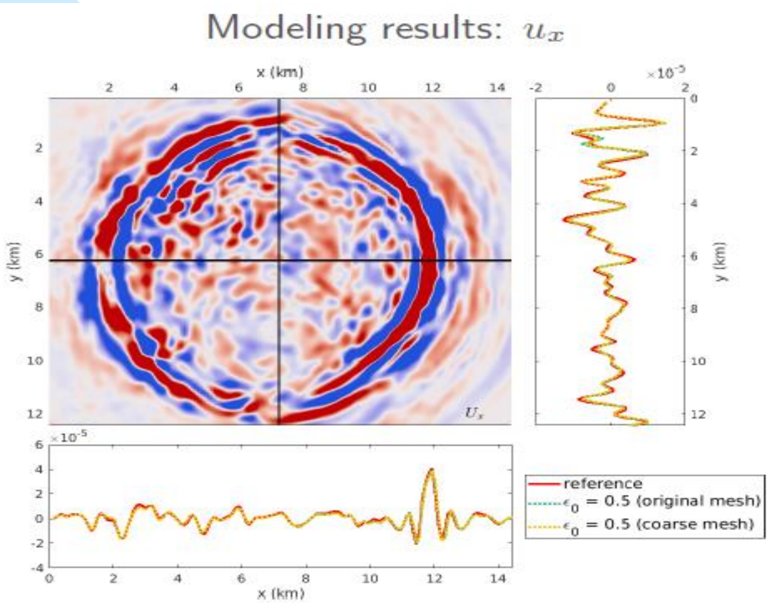
Do not discover the wheel again!

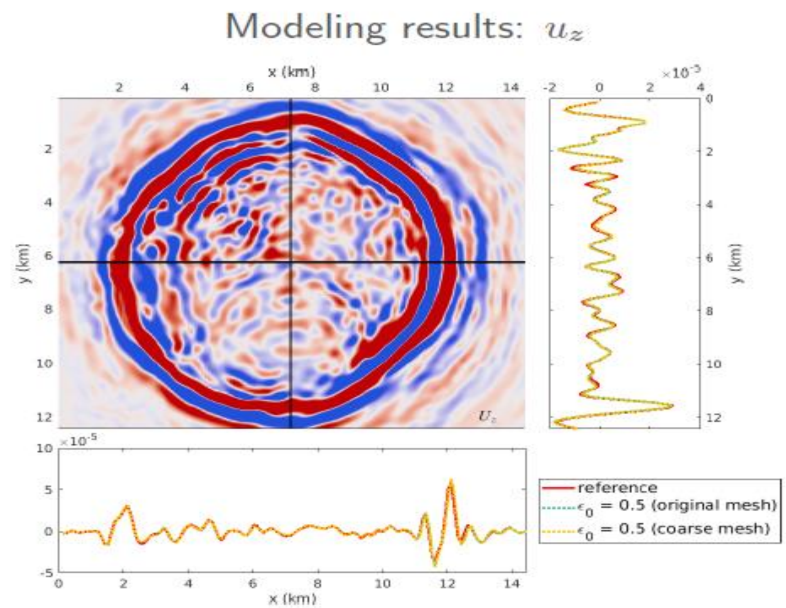
On what focus your energy!

Scenarios design:

- Source knowledge (strong variability)
- Velocity model (small variability)
  - Large-scale model
  - Short-scale model
- Site variability (strong variability)

Wave propagation modeling

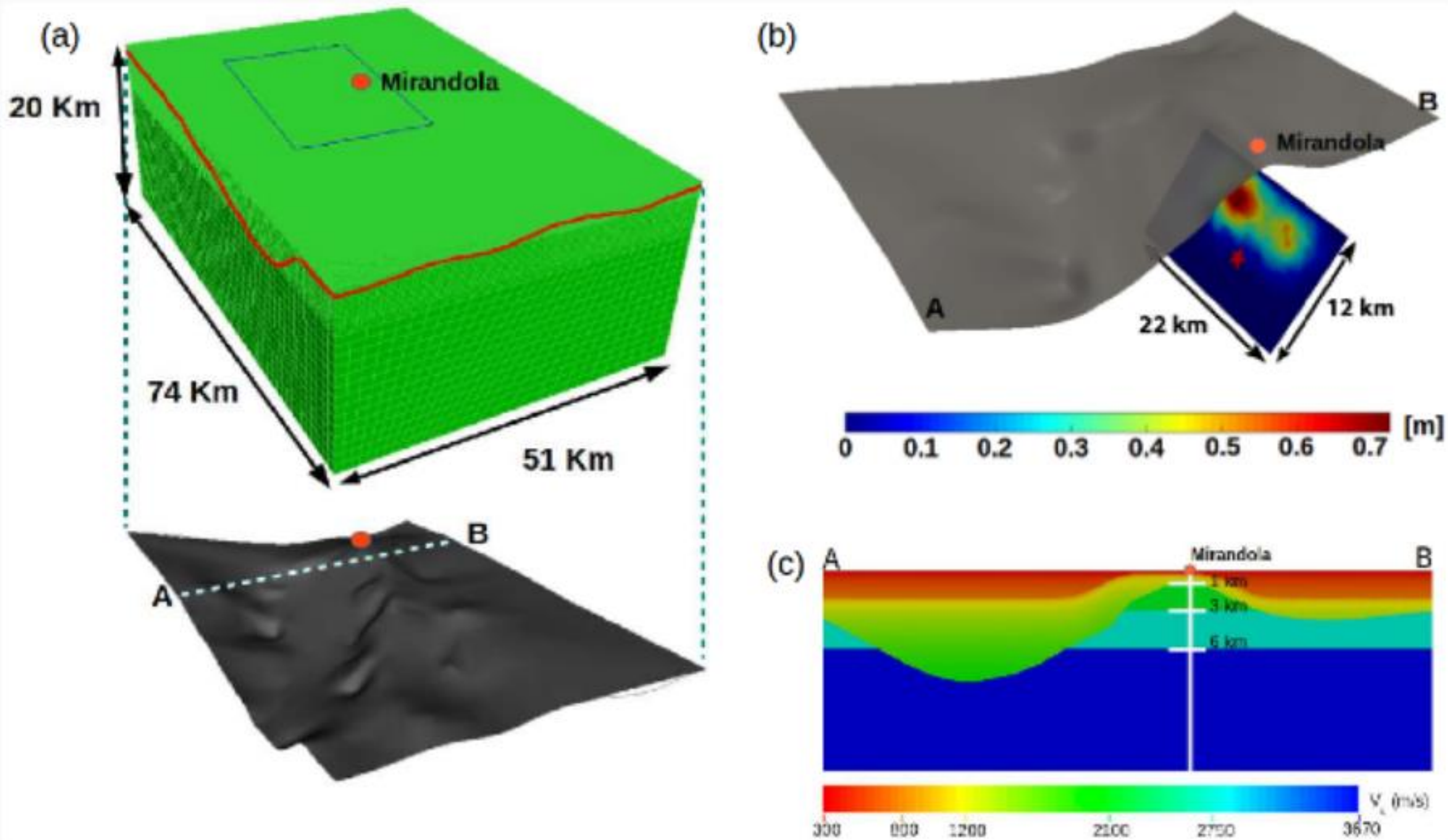




Partial differential equations: visco-elastic wave propagation!
Numerical tools: significant computer resources

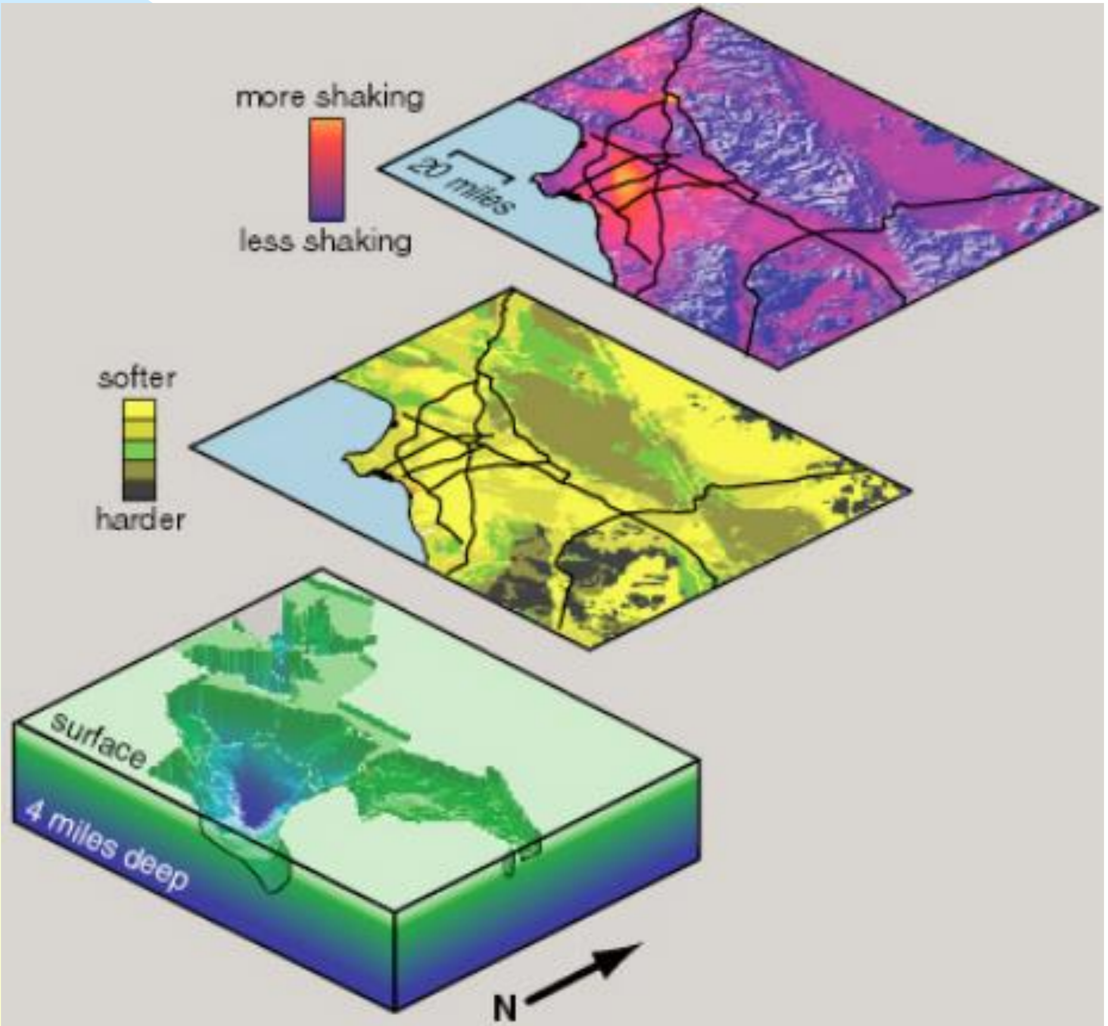
Model-reduction strategy: homogenization

Model design: large-scale variation



(a) 3D numerical model including the seismic fault responsible of the Mw 6.0 May 29 earthquake and the buried topography, corresponding to top of the Miocene formations. (b) Assumed slip distribution to model the earthquake fault rupture. (c) Representative NS cross-section of the numerical model passing through Mirandola, showing the Vs model adopted in the 3D numerical simulations for both Quaternary-Pliocene deposits and bedrock older formations.

Local/Site: short-scale variation



Local ground shaking depends on

softness of the surface rocks

thickness of surface sediments.

(from SCEC website)

Large-scale modeling building

Wave-medium interaction: two basic interactions

Transmission regime

most of tomographical models based on this regime

Reflection (converted) regime

seismic imaging based on this regime (migration, FWI)

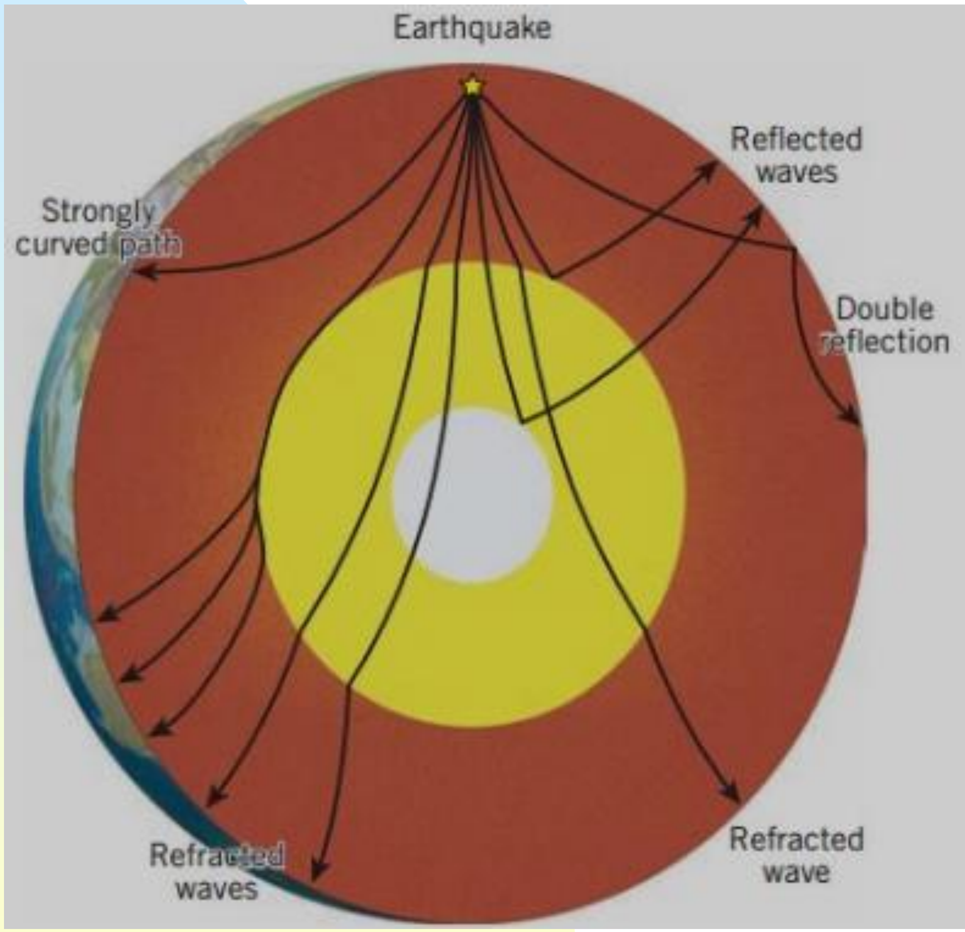
wave equation tomography (alias FWI in seismology)

Transmission Regime!

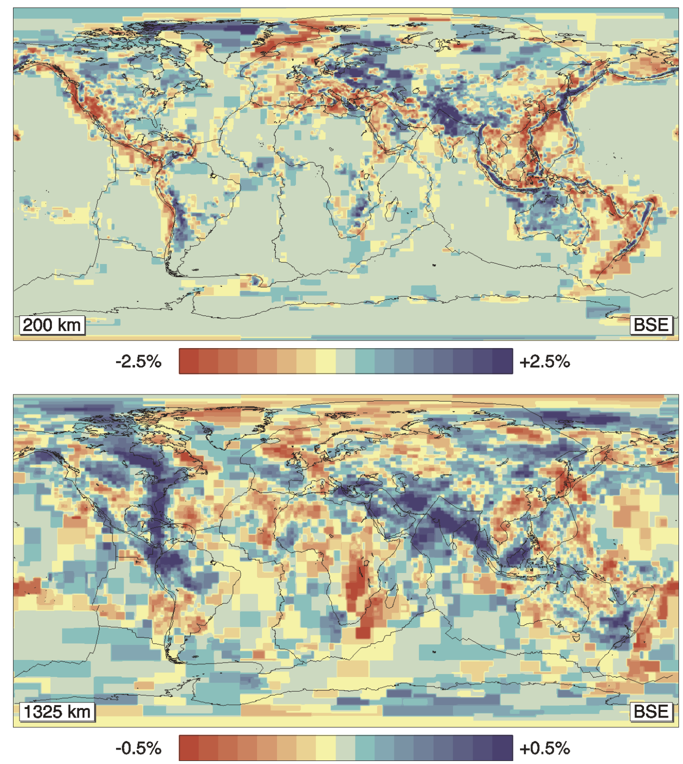
Most of tomographical models are based on transmission interaction

- Global scale
- Lithospheric/continental scale
- Upper crustal scale
- Near-surface scale
- Laboratory scale

Transmission Regime (global scale)!

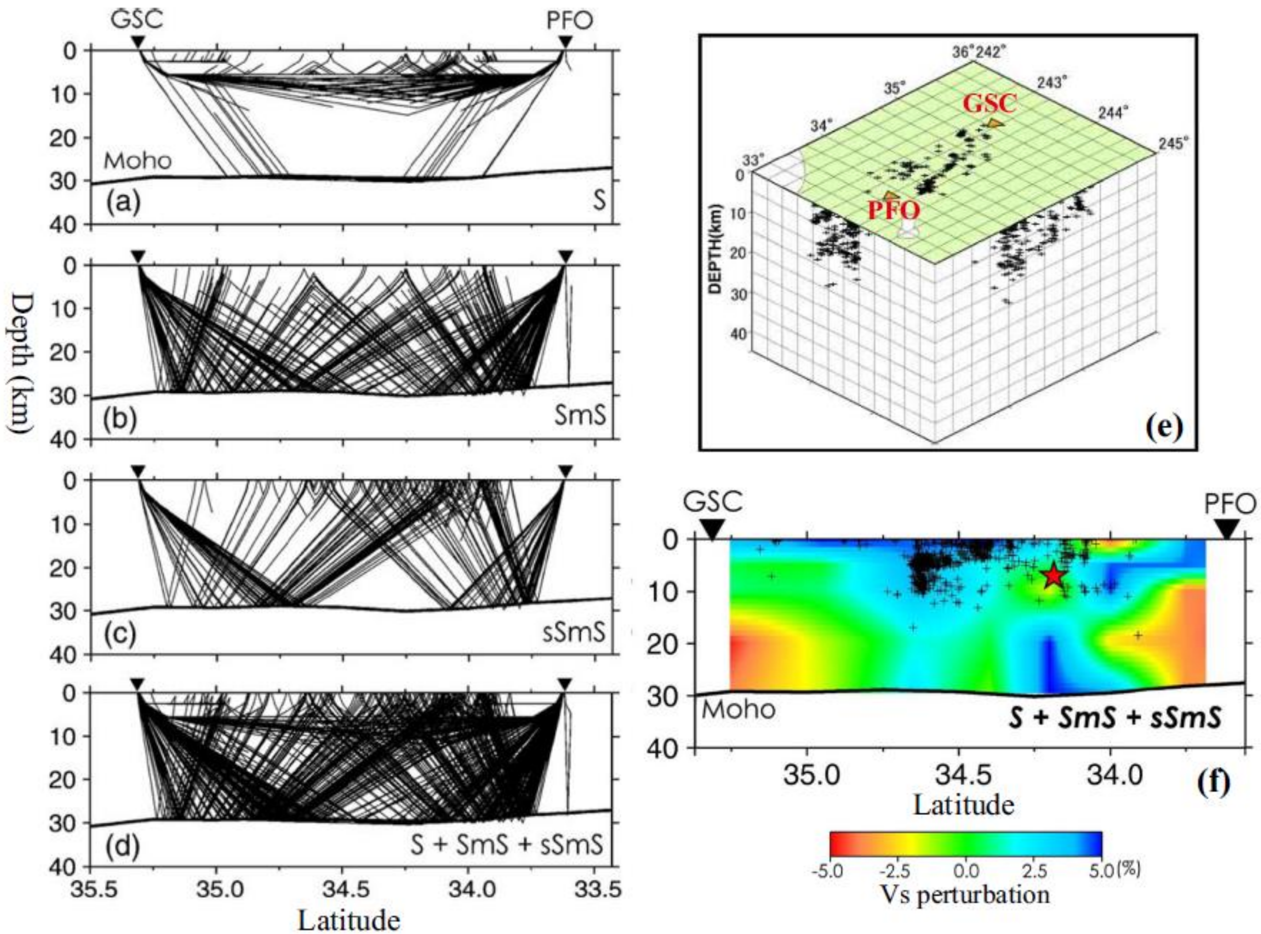


(Courtesy of W. Spakman)



Transmission Regime (crustal scale)!

Velocity analysis using reflection data

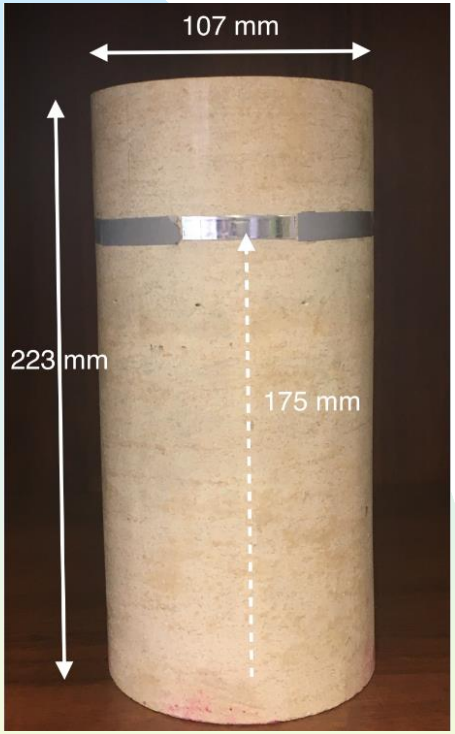


Reflection phases are used during their transmission travel down and up!

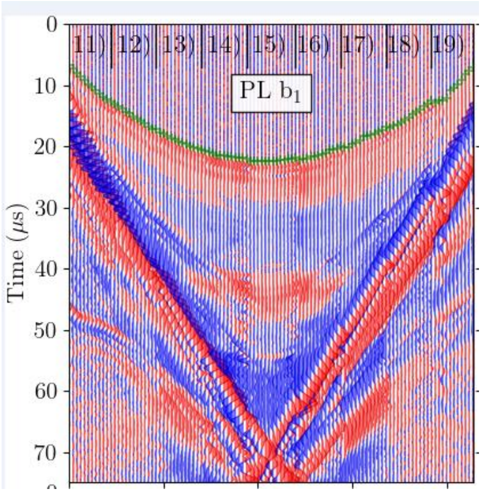
Not an interface imaging!

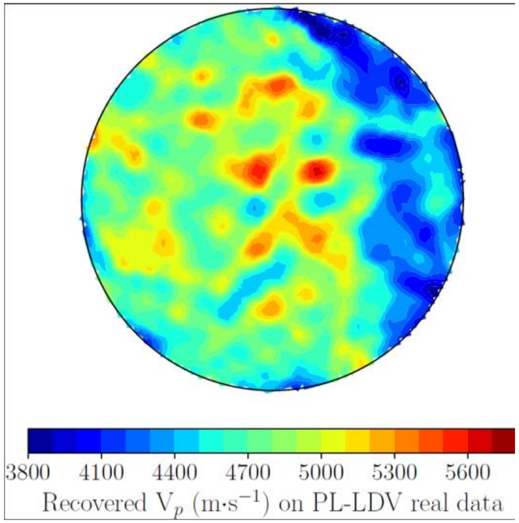
Transmission Regime: (lab. scale)!

Carbonate sample



Shen et al., (2022)





Only first-arrival picks are used for building the velocity model

Large-scale

velocity model building

(macro-velocity analysis)

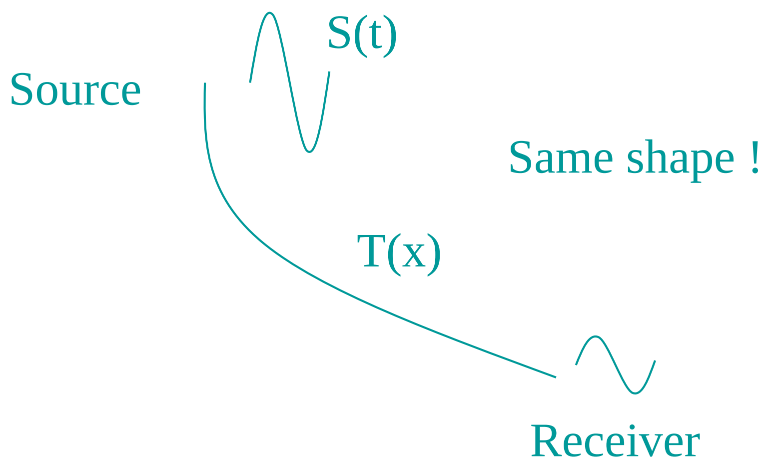
the ray concept for transmission interaction

This approach is widely used, even if wave equation steps over

(even if one still need ray approach)

Velocity model building: ray concept

$$u(x,t) = A(x)\, S\big(t - T(x)\big)$$
$$u(x,\omega) = A(x)\, S(\omega)\, e^{i\omega T(x)}$$



Travel-time $T(x)$ (phase $\omega T(x)$)
and Amplitude $A(x)$

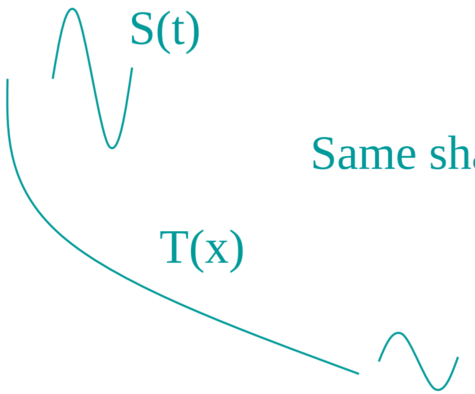
Highly diffracting medium:
Loosing wavefront coherence!

Preserved wavefront:
spatial continuity

« Nearly » same waveform at receiver position as the one emitted by the source

Velocity model building: ray concept

$$u(x,t) = A(x)\, S\big(t - T(x)\big)$$
$$u(x,\omega) = A(x)\, S(\omega)\, e^{i\omega T(x)}$$



Travel-time $T(x)$ (phase $\omega T(x)$)
and Amplitude $A(x)$

Asymptotic approach with growing frequencies

Diffraction still present!

Eikonal solution – GTD (J. Keller, 1962)

No diffraction at all!

Ray solution ⇔ Geometrical Optics ⇔ ∞ frequency

⇔ Singularities topology (shadow zone)

Link to the Catastrophe Theory (F. Math. René Thom)

Complex phase analysis (discontinuity)…

Low complexity of ray equations

- Ray tracing is a fast 1D integration in 2D/3D

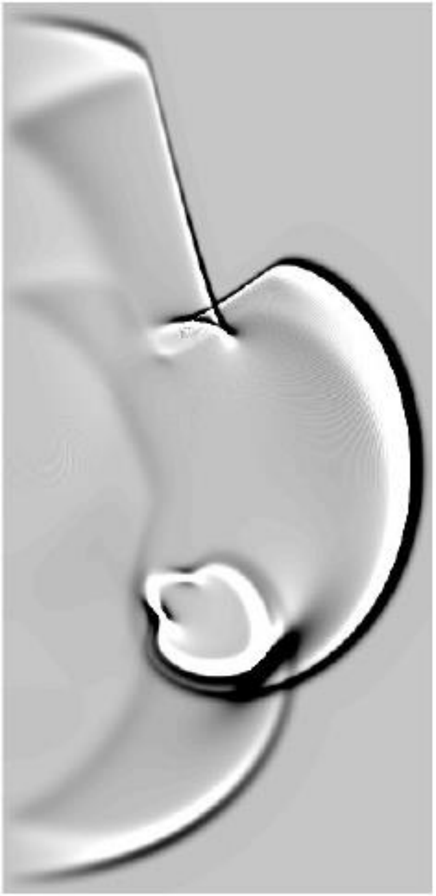
- Ray tracing equations as ODEs may sample the model quite evenly

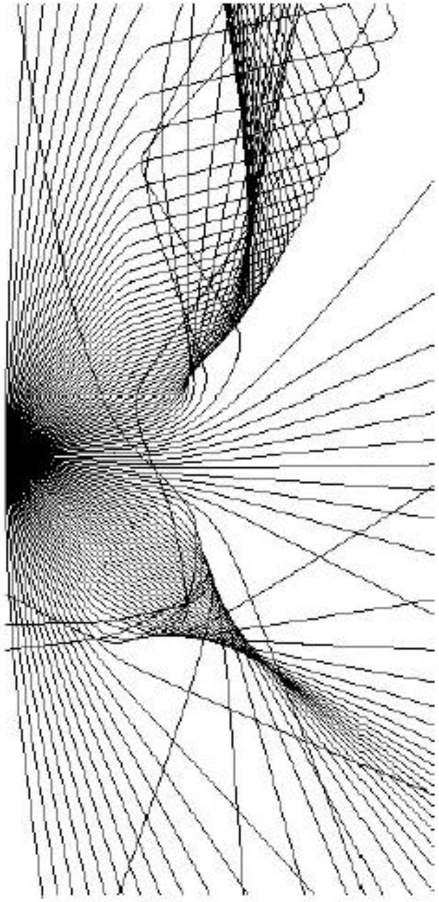
- Lagrangian formulation: we follow a point while tracing rays without regarding the density of rays inside the model

Keeping computer complexity low!

Inter-/extra-polation challenges

Wave solution





Ray solution

Inter-/extra-polation challenges

How to control the ray sampling of the model?

- Folding zones!  Available information
- Shadow zones!  Missing information

Ray equations: position & slowness ODE

Curvilinear stepping

$$\frac{d\vec{q}(s)}{ds} = c(\vec{q})\vec{p}$$
$$\frac{d\vec{p}(s)}{ds} = \nabla_{\vec{q}} \frac{1}{c(\vec{q})}$$
$$\frac{dT(s)}{ds} = \frac{1}{c(\vec{q})}$$

Time stepping

$$\frac{d\vec{q}(t)}{dt} = c^2(\vec{q})\vec{p}$$
$$\frac{d\vec{p}(t)}{dt} = c(\vec{q})\nabla \frac{1}{c(\vec{q})}$$

$$dt = \frac{1}{c(\vec{q})} ds = \frac{1}{c(\vec{q})^2} d\xi$$

Particule stepping

$$\frac{d\vec{q}(\xi)}{d\xi} = \vec{p}$$
$$\frac{d\vec{p}(\xi)}{d\xi} = \frac{1}{c(\vec{q})} \nabla \frac{1}{c(\vec{q})}$$
$$\frac{dT(\xi)}{d\xi} = \frac{1}{c^2(\vec{q})}$$

Any numerical integration tool: Runge-Kutta or Predictor-Corrector schemes.
However, Eikonal quantity $p^2 = 1/c^2(\vec{q})$ may be used for quality control. No need of automatic control of schemes.
Many analytical solutions (gradient of velocity; gradient of slowness square ...)

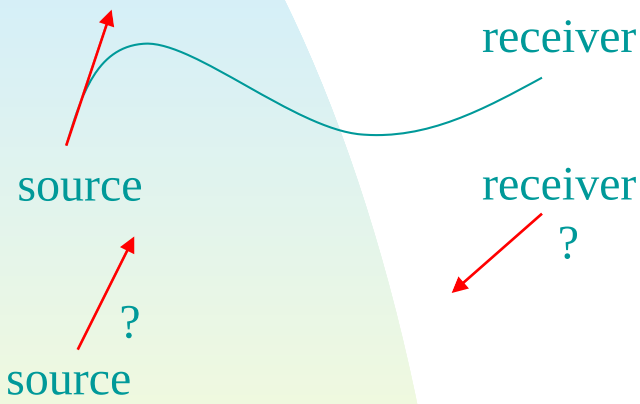
Integration of ray equations

1D sampling of 2D/3D medium : FAST

Runge-Kutta second-order integration

Predictor-Corrector integration

A very good QC: the eikonal must be equal to zero !



Initial conditions EASY

Boundary conditions VERY DIFFICULT

Shooting δp ?

Bending δx ?

Continuing δc ?

Save slowness conditions if possible !

AND FROM TIME TO TIME IT FAILS ! (inherent to geometrical optics)

But we need 2-points ray tracing because we have a source and a receiver to connect ! We even need more: branch identification (triplication for example)

Integration of ray equations

- Runge-Kutta of second order
- Write a computer program for an analytical law for the velocity: take a gradient with a component along x and a component along z

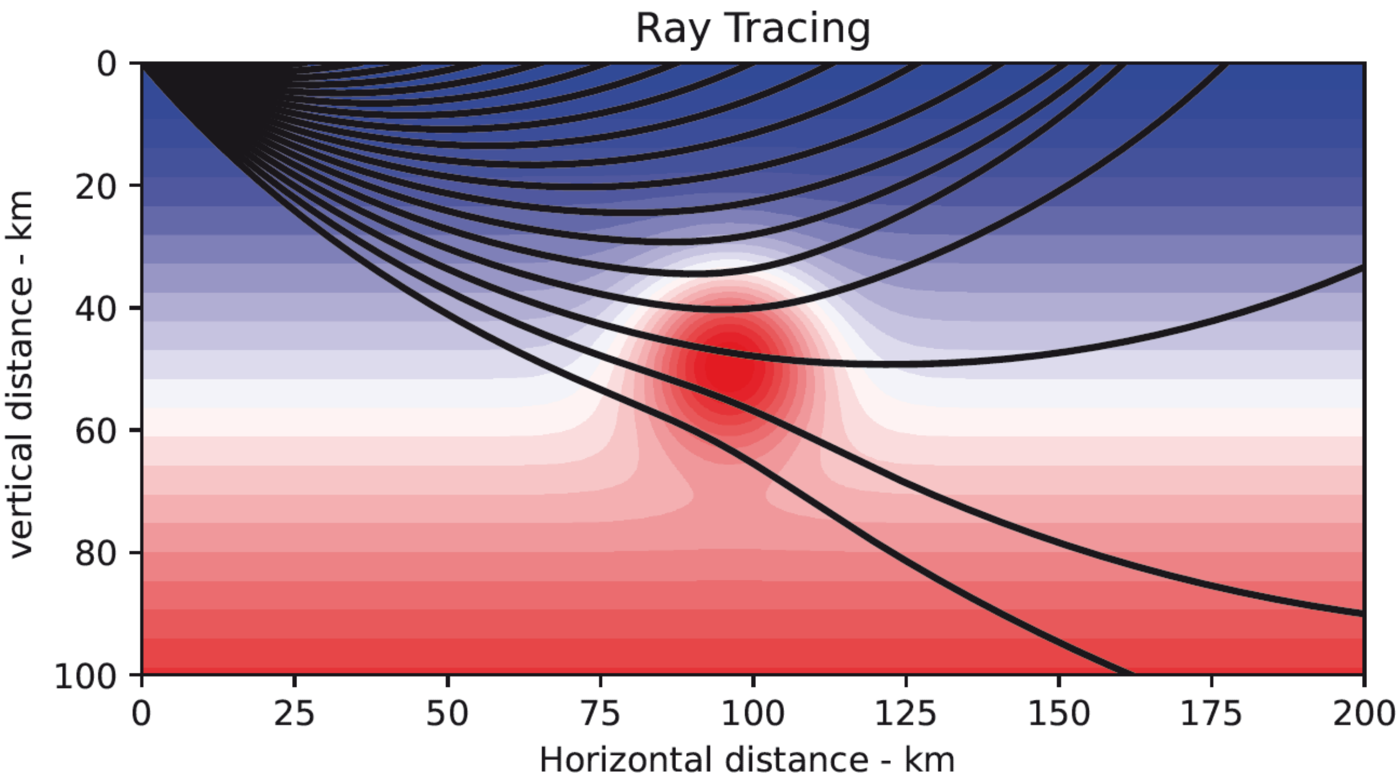
Home work : redo the same thing with a Runge-Kutta of fourth order (look after its definition) and predictor-corrector scheme if you are brave

Consider a gradient of the square of slowness and/or a vertical gradient of velocity

Consider a model defined by a grid with spline interpolation for computing spatial derivatives

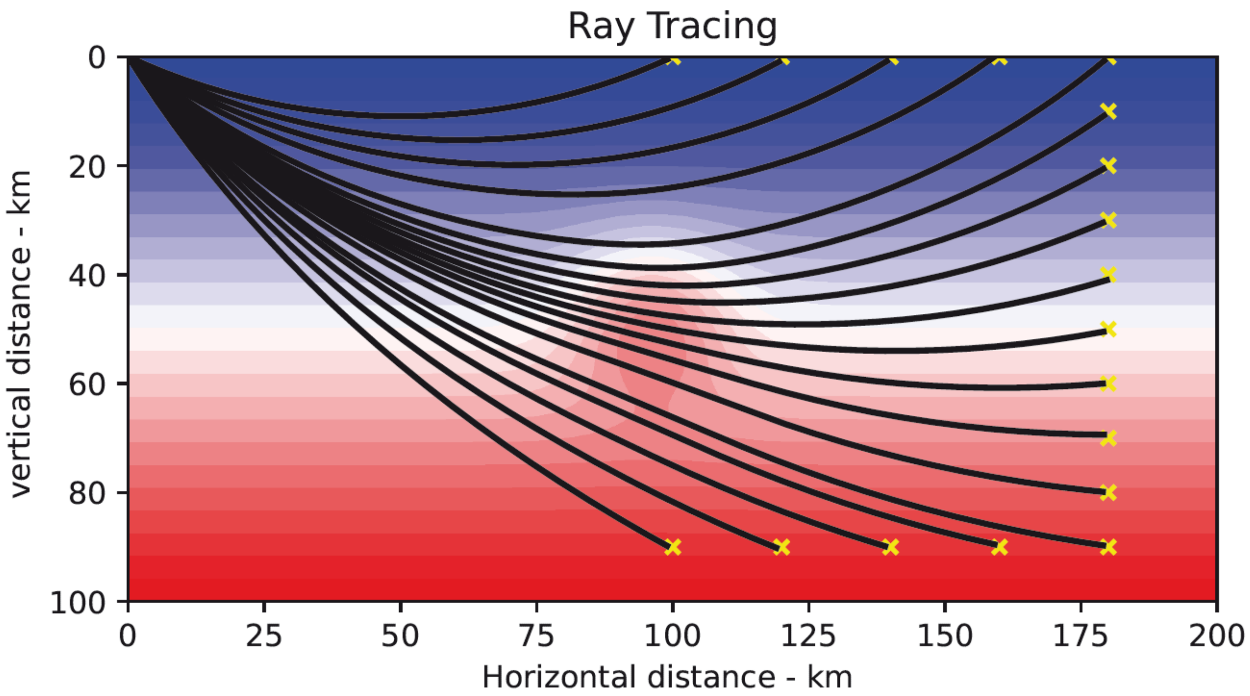
Ray tracing by rays

Ray tracing with initial conditions



Ray tracing by rays

Two-points ray tracing: how ?



Boundary conditions

How to sample the model around a given ray?

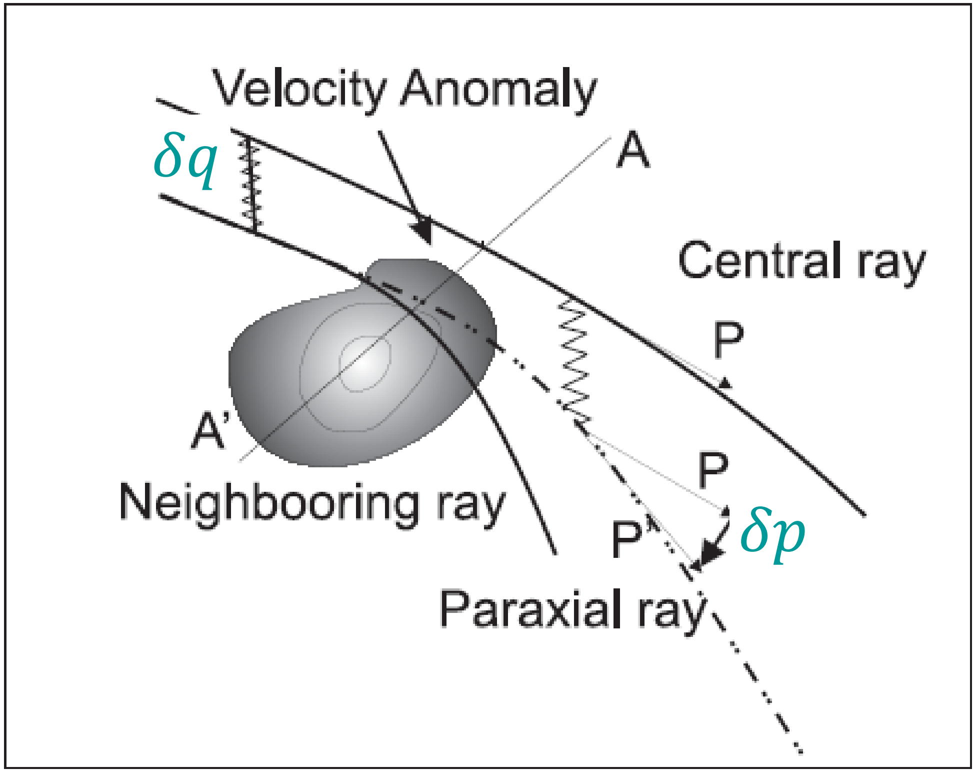
How to consider interfaces?

Paraxial theory around one ray

Paraxial ray theory

similar to

Gauss optics



2D simple linear system: isotropic case

The perturbation machinery

$$\frac{d\vec{q}(\xi)}{d\xi} = \mathcal{H}(\vec{q}(\xi), \vec{p}(\xi)) \Rightarrow \frac{d(\overrightarrow{q_0} + \overrightarrow{\delta q})}{d\xi} = \nabla_{\overrightarrow{p_0}+\overrightarrow{\delta p}} \mathcal{H}\left(\overrightarrow{q_0} + \overrightarrow{\delta q}, \overrightarrow{p_0} + \overrightarrow{\delta p}\right)$$

$$\frac{d\overrightarrow{\delta q}}{d\xi} = \nabla_{\overrightarrow{p_0}} \nabla_{\overrightarrow{p_0}} \mathcal{H}(\overrightarrow{q_0}, \overrightarrow{p_0})\overrightarrow{\delta p} + \nabla_{\overrightarrow{p_0}} \nabla_{\overrightarrow{q_0}} \mathcal{H}(\overrightarrow{q_0}, \overrightarrow{p_0})\overrightarrow{\delta q}$$

$$\frac{d}{d\xi}\begin{bmatrix}\delta q_x \\ \delta q_z \\ \delta p_x \\ \delta p_z\end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0.5\,\frac{\partial^2\, 1/c^2(\vec{q}_0)}{\partial x^2} & 0.5\,\frac{\partial^2\, 1/c^2(\vec{q}_0)}{\partial x \partial z} & 0 & 0 \\ 0.5\,\frac{\partial^2\, 1/c^2(\vec{q}_0)}{\partial z \partial x} & 0.5\,\frac{\partial^2\, 1/c^2(\vec{q}_0)}{\partial z^2} & 0 & 0 \end{bmatrix} \begin{bmatrix}\delta q_x \\ \delta q_z \\ \delta p_x \\ \delta p_z\end{bmatrix}$$

Linear system!

More complex for anisotropic structure but still workable

Linear system: perturbation!

In a 2D model, four elementary paraxial trajectories

Two coordinates $\delta q_x, \delta q_z$ and two slowness components $\delta p_x, \delta p_z$

δy1^t(0)=(1,0,0,0)

δy2^t(0)=(0,1,0,0)

δy3^t(0)=(0,0,1,0) NOT paraxial RAY ! Linear system

δy4^t(0)=(0,0,0,1)

$$\delta y^t = (\delta q_x, \delta q_z, \delta p_x, \delta p_z)^t = \alpha^1 \delta \mathrm{y}^{1t} + \alpha^2 \delta \mathrm{y}^{2t} + \alpha^3 \delta \mathrm{y}^{3t} + \alpha^4 \delta \mathrm{y}^{4t}$$

Any paraxial ray is a linear combination of these four elementary trajectories

Point source condition

Point source: no shift in the position when doing perturbation:

$$\delta q_x(O) = \delta q_z(O) = 0 \;\Rightarrow\; p_x(0)\delta p_x(0) + p_z(0)\delta p_z(0)=0$$

$\delta p_x(0) = \alpha p_z(0)$

This is enough to verify initially such a condition

$\delta p_z(0) = -\alpha p_x(0)$

α *arbitrary constant (linear system)*

Point source paraxial solution $\delta y^{a}(\xi) = \alpha p_z(0)\,\delta y3(\xi) - \alpha p_x(0)\,\delta y4(\xi)$

elementary trajectories

From paraxial trajectories, one can combine them for paraxial rays as long as the linearized eikonal equation is verified.

For a point source, the parameter α could be set to an arbitrary small value: this is a derivative or plan tangent computation (Gauss optics)

Plane wave condition

Plane source: shift in the shooting position when doing perturbation:

$$\delta p_x(O) = \delta p_z(O) = 0 \;\Rightarrow\; \frac{1}{2}\frac{\partial\, 1/c^2(x,z)}{\partial x}(0)\,\delta q_x(0) + \frac{1}{2}\frac{\partial\, 1/c^2(x,z)}{\partial z}(0)\,\delta q_z(0) = 0$$

$$\delta q_x(0) = \alpha \frac{1}{2}\frac{\partial\, 1/c^2(x,z)}{\partial z}(0)$$

$$\delta q_z(0) = -\alpha \frac{1}{2}\frac{\partial\, 1/c^2(x,z)}{\partial x}(0)$$

This is enough to verify initially such a condition but gradient of velocity at the source could be quite arbitrariry

Cerveny's condition (both x and z variation)

$\alpha\ arbitrary\ constant\ (linear\ system)$

Paraxial solution $\delta y^b(\xi) = \alpha \frac{1}{2}\frac{\partial 1/c^2(x,z)}{\partial z}(0)\delta y1(\xi) - \alpha \frac{1}{2}\frac{\partial 1/c^2(x,z)}{\partial x}(0)\,\delta y2(\xi)$

We combine the first two paraxial ray trajectories.

elementary trajectories

Two independent paraxial rays in 2D ($\delta y^a\ and\ \delta y^b$): point (seismograms) and plane (beams) paraxial rays

Paraxial source conditions

Two independent paraxial rays in 2D (δy^a *and* δy^b):
point (seismograms) and plane (beams) paraxial rays

Four independent paraxial rays in 3D ($\delta y^a, \delta y^b, \delta y^c,$ *and* δy^d):
2 point (seismograms) and 2 plane (beams) paraxial rays

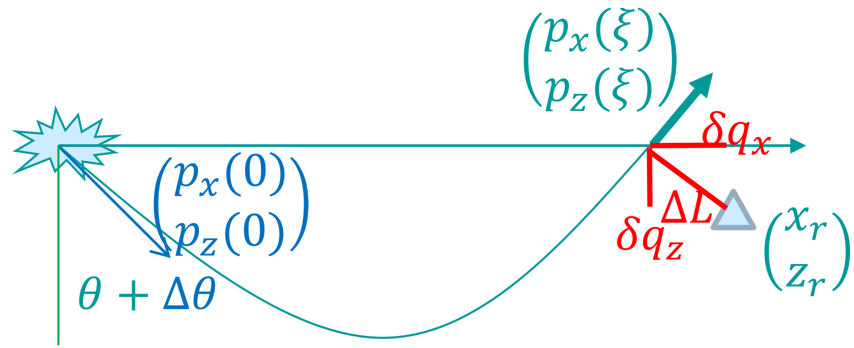
Remark: working with trajectories implies that paraxial conditions could be defined on the fly for having local conditions at different points of the model

Remark: 2 point and 2 plane paraxial trajectories in 3D!

Two-points ray tracing with paraxial values

Consider the orthogonal distance between ray and receiver

$$\Delta L = (x_r - \delta q_x)p_z(\xi) - (z_r - \delta q_z)p_x(\xi)$$



Point paraxial condition

Solve iteratively $\Delta L = \frac{dq_L}{d\theta}\ \Delta\theta$

or $\Delta L = \left(\frac{dq_x}{d\theta} p_z(\xi) - \frac{dq_z}{d\theta} p_x(\xi)\right)\Delta\theta$

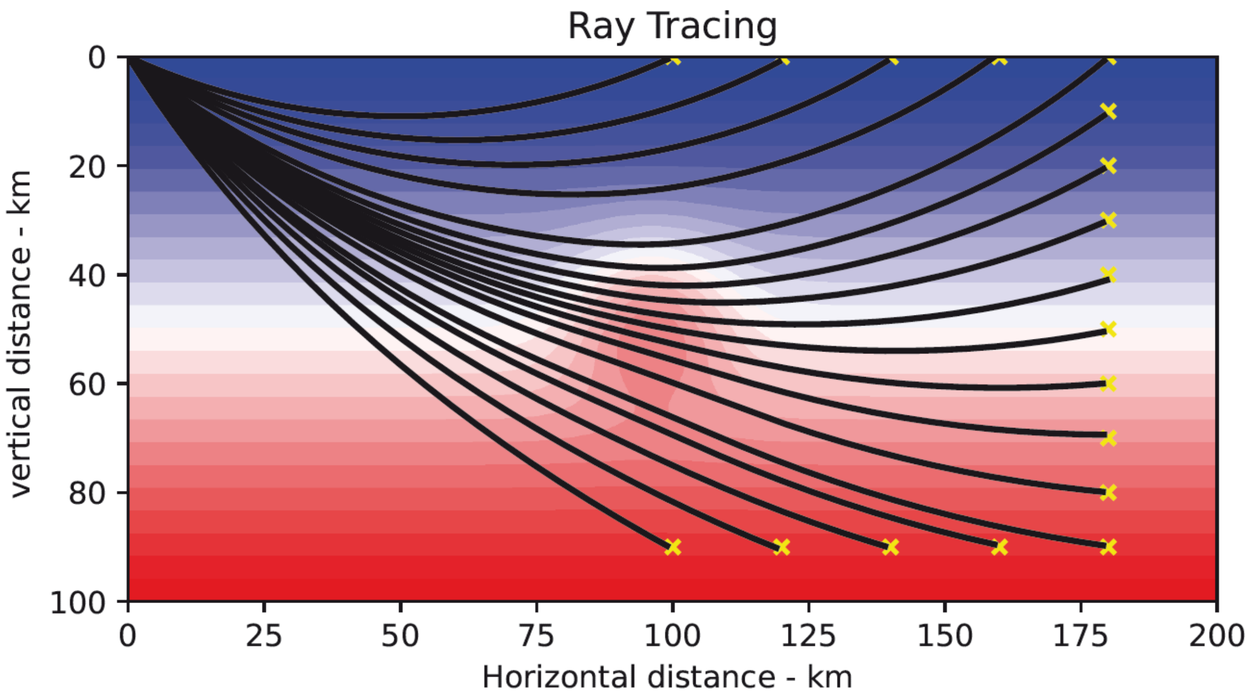
Derivative wrt shooting angle $\frac{dq_x}{d\theta} = \frac{\partial q_x}{\partial p_x}\frac{dp_x}{d\theta}(0) + \frac{\partial q_x}{\partial p_z}\frac{dp_z}{d\theta}(0)$ or $\frac{dq_x}{d\theta} = \frac{\partial q_x}{\partial p_x}\ p_z(0) - \frac{\partial q_x}{\partial p_z} p_x(0)$

$$\frac{dq_x}{d\theta} = \delta q_x 3\ \ p_z(0) - \ \delta q_x 4\ \ p_x(0) = \delta q_x$$
$$\frac{dq_z}{d\theta} = \delta q_z 3\ \ p_z(0) - \ \delta q_z 4\ \ p_x(0) = \delta q_z$$

$$\Delta\theta = \frac{(x_r - \delta q_x)p_z(\xi) - (z_r - \delta q_z)p_x(\xi)}{\delta q_x p_z(\xi) - \delta q_z p_x(\xi)}$$

Ray tracing by rays

Two-points ray tracing



Amplitude, polarization & beam summation

Similar cooking receipts

for these quantities in order to compute asymptotic seismograms.

However, these receipts are not often used: most velocity model building are based on traveltime information (and sometimes on propagation direction/slowness vector, known as slope tomography or double-difference tomography)

Ray tracing: partial lessons to take away

Rays: a quite useful tool for interpretation and understanding

- Geometrical optics: ODE versus PDE
  - Choose PDE when possible !
- ODE: tracing one (paraxial) ray is fast
  - Please always trace paraxial rays as incremental cost
- Keep complexity low (seismic waves are finite frequency waves)
  - Do not drown yourself into the no-scale « optical » infinite-frequency singularities
- Rays help the identification of phases: key interpretation
  - PDE does not allow easy interpretation! (maybe work in progress …)

Toy computer codes for ray tracing

https://github.com/jeanvirieux/Tomography_training

Sub-directory: ray_tracing_analytic(.template)

comparison between analytic solutions and runge-kutta solutions

Sub-directory: ray-tracing_grid(.template)

ray tracing over a velocity grid using bspline interpolation

Sub-directory: two_points_ray_tracing(.template)

ray tracing including paraxial ray tracing when hitting receiver

Simple codes based on python3 for practical understanding of the different equations of this presentation

To be done during the training

Scrutinize python codes
and
run the shooting examples

Numerical strategies are simpler behind the ray theory semantics ☺

Ray-based tomography: Fréchet derivative

Delayed ray-based tomography based

on sensitivity matrix building
on Jacobian matrix building
on Fréchet derivative building

Delayed traveltime tomography

$$t(source, receiver) = \int_s^r s(x,y,z)dl$$

Finding slowness $s(x,y,z)$ from $t(s,r)$ difficult problem: only solution for one variable $s(z)$ (Abel) !

Consider small perturbations $\delta s(x,y,z)$ of the slowness field $s_0(x,y,z)$

$$t(s,r) = \int_s^r s(x,y,z)dl = \int_s^r s_0(x,y,z)dl + \int_s^r \delta s(x,y,z)dl$$

$$t(s,r) \approx \int_{s_0}^{r_0} s_0(x,y,z)dl + \int_{s_0}^{r_0} \delta s(x,y,z)dl$$

« frozen » ray approximation (ray connecting source/receiver for the known slowness s_0)

do not ask rescue from *Fermat!*

$$t(s,r) - t_0(s,r) \approx \int_{s_0}^{r_0} \delta s(x,y,z)dl$$

$$\delta t(s,r) \approx \int_{s_0}^{r_0} \delta s(x,y,z)dl$$

LINEARIZED PROBLEM $\delta t(d) = J(d,m)\, \delta s(m)$ from model domain to data domain

Discretization of the slowness perturbation

Velocity perturbation field or slowness field $\delta s(x, y, z)$ can be described into a meshed cube regularly spaced in the three directions.

For each node, we specify a value $\delta s_{i,j,k}$. The interpolation will be performed with functions as step functions. For each grid point (i,j,k), shape functions $h_{i,j,k} = 1$ for i,j,k, and zero for other indices.

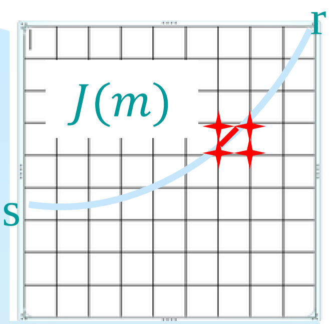
$$\delta s(x, y, z) = \sum_{cube} \delta s_{i,j,k}\, h_{i,j,k}$$

Nodal approach

Other shape functions are possible with two-end members (nodal versus modal): fourier functions (cos,sin), chebychev, spline, wavelet … and so on

Sampling the model space is the mandatory stabilization strategy (smoothing or damping ones)
Model discretization provide an implicit limit to the wavenumber range to be filled in

Discrete linearized inversion problem



$$\delta t(\mathrm{s},\mathrm{r}) = \int_{ray_0} \sum_{cube} \delta s_{i,j,k}\, h_{i,j,k}\, dl = \sum_{cube} \delta s_{i,j,k} \boxed{\int_{ray_0} h_{i,j,k} dl}$$

Weighted ray segment

$$\delta t(s,r) = \sum_{i,j,k} \delta s_{i,j,k} \Delta l_{i,j,k} = \sum_{i,j,k} \frac{\partial t}{\partial s_{i,j,k}} \delta s_{i,j,k}$$

Discretization of the model fats the ray $\delta t(s,r) = \sum_{i,j,k} J_{i,j,k}\ \delta s_{i,j,k}$

$$\delta t(n) = J(n,m)\, \delta s(m)$$

to be solved in least-squares sense

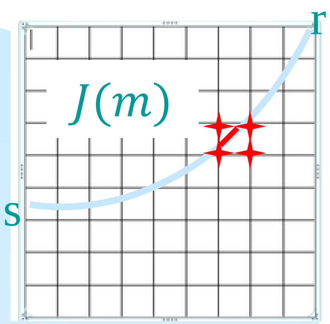
$$\begin{pmatrix} \delta t_1 \\ \delta t_2 \\ \vdots \\ \delta t_{n-1} \\ \delta t_n \end{pmatrix} = \begin{pmatrix} \frac{\partial t_1}{\partial s_1} & \cdots & \frac{\partial t_1}{\partial s_m} \\ \frac{\partial t_2}{\partial s_1} & \cdots & \frac{\partial t_2}{\partial s_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial t_{n-1}}{\partial s_1} & \cdots & \frac{\partial t_{n-1}}{\partial s_m} \\ \frac{\partial t_n}{\partial s_1} & \cdots & \frac{\partial t_n}{\partial s_m} \end{pmatrix} \begin{pmatrix} \delta s_1 \\ \delta s_2 \\ \vdots \\ \delta s_{m-1} \\ \delta s_m \end{pmatrix}$$

Sensitivity matrix J is a sparse matrix

also named Fréchet derivative or Jacobian matrix

…

Discrete linearized inversion problem



$$\delta t(s,r) = \sum_{i,j,k} \delta s_{i,j,k} \Delta l_{i,j,k}$$

$$\delta t(s,r) = \sum_{i,j,k} J_{i,j,k} \; \delta s_{i,j,k}$$

Discretization of the model fats the ray

(billion, million)

$$\delta t(n) = J(n,m)\, \delta s(m)$$

- n dimension of the data space
- m dimension of the model space

Sparse system

$$\begin{pmatrix} \delta t_1 \\ \delta t_2 \\ \vdots \\ \delta t_{n-1} \\ \delta t_n \end{pmatrix} = \begin{pmatrix} \frac{\partial t_1}{\partial s_1} & \cdots & \frac{\partial t_1}{\partial s_m} \\ \frac{\partial t_2}{\partial s_1} & \cdots & \frac{\partial t_2}{\partial s_m} \\ \vdots & \ddots & \vdots \\ \frac{\partial t_{n-1}}{\partial s_1} & \cdots & \frac{\partial t_{n-1}}{\partial s_m} \\ \frac{\partial t_n}{\partial s_1} & \cdots & \frac{\partial t_n}{\partial s_m} \end{pmatrix} \begin{pmatrix} \delta s_1 \\ \delta s_2 \\ \vdots \\ \delta s_{m-1} \\ \delta s_m \end{pmatrix}$$

Least-squares solution

The rectangular system can be recast into a square system (sometimes called normal equations).

- Solving this square linear system gives the so-called least-squares solution.

Least-squares solution

$$J^t J \delta s = J^t \delta t$$

$$\delta s = (J^t J)^{-1} J^t \delta t$$

- Another interesting solution with minimum norm

Remark

Least-norm solution

$$J J^t \delta u = \delta t$$

$$\delta s = J^t (J J^t)^{-1} \delta t$$

The system is both under-determined and over-determined depending on the considered zone (and the number of rays going through).

Damped least-squares solution

$$\delta t = J \delta s$$
$$d = Gm$$
$$b = Ax$$

Damping parameter ε

A is a rectangular matrix (either over- or under-determined)

$$\min_x \|Ax - b\|^2 + \varepsilon \, \|x\|^2$$

LSQR solves it using only products Ax or $A^T b$ by considering the system

$$(A^T A + \varepsilon \, I)x = A^T b$$

widely used subroutine in traveltime tomography

LSMR solves it using only products Ax or $A^T b$ by considering the system

$$(A^T A + \varepsilon \, I)x = A^T b$$

http://www.numerical.rl.ac.uk/spral/doc/latest/Fortran/
http://web.stanford.edu/group/SOL/software/lsmr

Occam's razor

- Do not use more complicated maths than the data deserves

- Approximate the least constrained quantity

Given: data (observed and modeled)
Assumed: wavefront propagation
Unknown: Earth structure

- Occam's Razor: parcimonious principle

When you have many explanations for predicting exactly the same quantities and that there is no way to distinguish them, select the simplest one… until you end up with a contradiction.

Ockham (~1295-~1349)

Constable et al (1987)

Constrained damped least-squares solution

Constrained damped least-squares solution

$$\min_x(\|Ax-b\|^2+\lambda\|Dx\|^2+\varepsilon\,\|x\|^2)$$

Two hyper-parameters λ and ε to be selected?

Penalty approach is often selected

$$\begin{bmatrix}\mathcal{G}_k\\ \varepsilon I\end{bmatrix}[\delta s_k]=\begin{bmatrix}\dfrac{\partial t}{\partial s_k}\\ \lambda D\\ \varepsilon I\end{bmatrix}[\delta s_k]=\begin{bmatrix}\delta t_k\\ 0\\ 0\end{bmatrix}$$

D operator is a smoothing operator, such as a Laplacian operator which limit variations of the spatial second derivative of the slowness model.

Smoothing could vary with coordinates $\lambda_x D_x+\lambda_y D_y+\lambda_z D_z$
with seven-points finite-difference stencil along each direction for the laplacian

Discrete Fréchet-Ray algorithm

Two-loops procedure

$k = 0$ Initial velocity model s_0

Outer loop

$k = k + 1$

Ray tracing (src/rec)

Sensitivity matrix $J(s_k)$

(accurate synthetic delayed times dt)

Constrained matrix $\mathcal{g}(s_k)$

(70% of CPU)

LSQR or LSRM includes the damping operator

LSQR or LSRM iterative method

$\mathcal{g}\,(s_k)ds_l = dt_k$

Inner loop

l= $l + 1$

$\text{Err} = |\mathcal{g}(s_k)ds_l - dt_k|$

Err

$ds_k small$

End

The slowness field is denoted by s and it is often the one we reconstruct

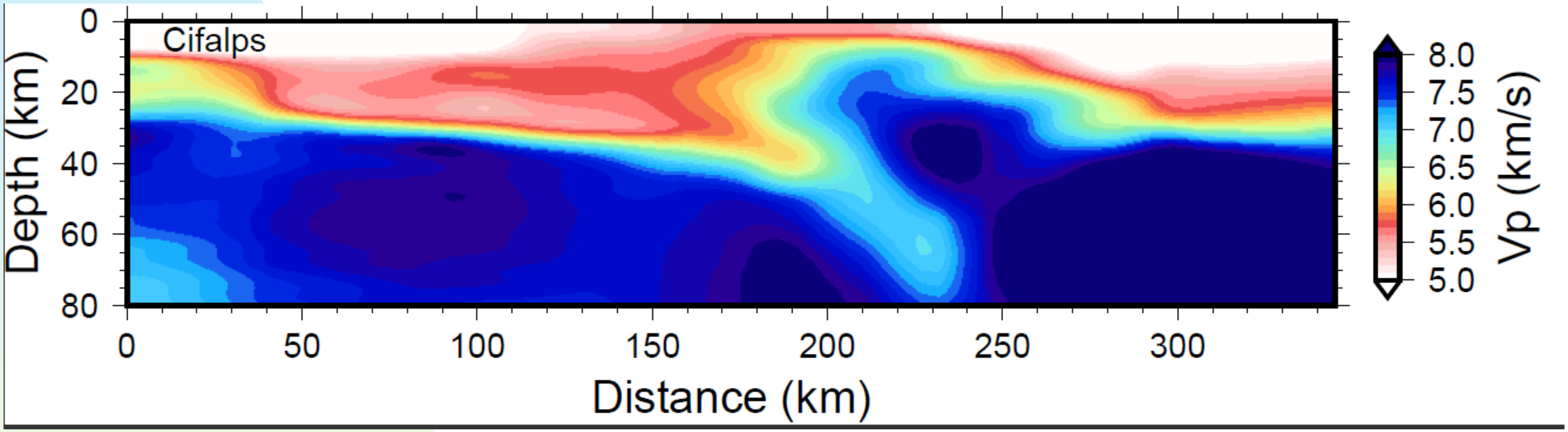
Surface wave tomography
Ambient Noise tomography

Dispersion curve analysis of the fundamental model of Rayleigh waves (and also Love waves) is also based on transmission regime with ray approximation

The particular feature is the vertical eigenfunction of the fundamental model connected to the frequency

Still a transmission regime!

Ambient Noise tomography



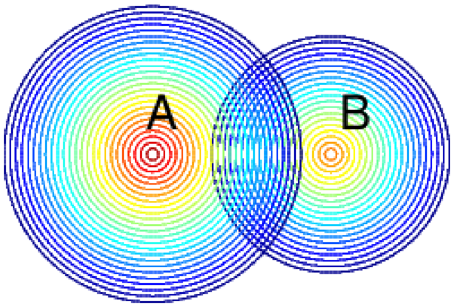
Still a transmission regime!

Ray ODE vs Eikonal PDE

(after Runborg, 1998)

❑Scalar wave equation PDE

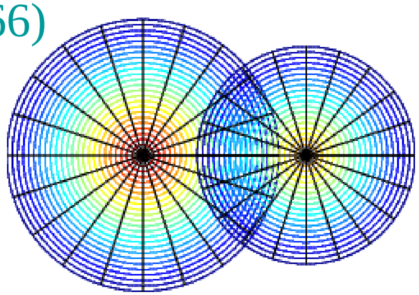
❖ Linear partial differential equation
❖ Eulerian formulation



Wave solution

❑Ray ODE: Methods of characteristics (Courant & Hilbert, 1966)

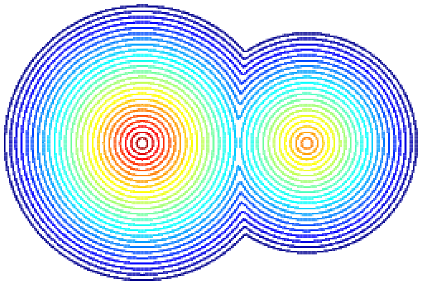
❖Non-linear ordinary differential equations
❖Lagrangian formulation as we integrate along rays



Ray solution

❑Eikonal PDE: challenging equation (complete solution – the Graal)!

❖ Non-linear partial differential equations
❖ Eulerian formulation as we compute quantities at fixed positions
❖ Fastest solution through fast marching or fast sweeping methods



Eikonal solution

Asymptotic solution; beyond ray solution!

Geometrical theory of diffraction (Keller, 1962)

- Ray solution is **one** asymptotic solution among many other expansions.

Ray Ansatz is limited to integer power of frequency
Zero-out any diffraction effect (fractional frequency power)

- Airy, Bessel, Mathieu expansions (complicated formulations)!

alternative expansions
with or without diffraction

- Fastest solution known as viscous solution (Crandall & Lions, 1983)

Eikonal equation

Ray solution vs first-arrival solution

(Cerveny, 2001)

3.8.1 Ray Theory Travel Times and First-Arrival Travel Times

In this section, we shall define the ray-theory travel times and first-arrival travel times and explain the main differences between them.

One extracted property of first-arrival solution mentioned by Cerveny (2001)

c. The first-arrival travel time is a *unique* function of position. It is defined at any point of the model. There are no shadow zones. Moreover, the first-arrival travel time is a continuous function of coordinates. The first spatial derivatives of the first-arrival travel time, however, may be discontinuous. They may be discontinuous even at points where the velocities are continuous. (Example of such discontinuity include the intersection of the wavefronts of direct and head waves.)

Viscous solution!

Understanding this asymptotic solution!

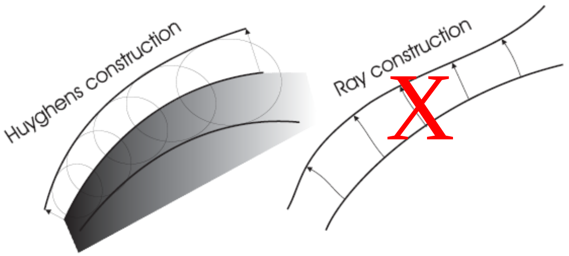
Defining its validity domain!

Crandall & Lions (1983)

Fermat/Huygens principles

These principles are backbones to Eikonal PDE

First-arrival traveltimes follow Fermat principle of minimum time along any trajectory connecting the starting point and the end point. This principle is highly connected to the Huygens principle related to the wavefront construction



The related variational problem can be written

slowness $u = {}^{1}/_{c}$

$$\delta \int u(s)ds = 0$$

curvilinear coordinalte s

The Eikonal equation is thought of the related PDE of this variational problem, leading to the Hamilton-Jacobi(-Bellman) equation.

(Kalaba, 1961 (isotropic case); Brandstatter, 1974 (anisotropic case)).

Fermat principle to Eikonal equation

In a 2D medium defined by coordinates (x, z), the path of the ray (perpendicular to the wavefront) is such that

$$\mathrm{T}(x, z) = \min_{l} \int u\big(x(l), z(l)\big) dl \qquad \textbf{(1)}$$

We consider an infinitesimal path Δl from $(x - sin\theta\ \Delta l$, $\mathrm{z} - cos\theta\ \Delta l)$ where the angle θ is the tangent angle to the current trajectory.

We get

From Lakshminarayanan and Varadharajan (1997)

$$\mathrm{T}(x, z) = \min_{\theta}[T(x - sin\theta\ \Delta l\ , \mathrm{z} - cos\theta\ \Delta l) + u(x - sin\theta\ \Delta l\ , \mathrm{z} - cos\theta\ \Delta l)\Delta l + \sigma(\Delta l)]$$

Expanding in Taylor series, we get

$$T(x, z) = \min_{\theta}\left[T(x, \mathrm{z}) - sin\theta\Delta l \frac{\partial T}{\partial x} - cos\theta \frac{\partial T}{\partial z} + u(x, z)\Delta l + \sigma(\Delta l)\right]$$

Or

$$\min_{\theta}[-sin\theta\ \Delta l\ T_x - cos\theta\ \Delta l\ T_z + u(x, z)\Delta l + \sigma(\Delta l)] = 0$$

When Δl is small, we get

$$sin\theta\ T_x + cos\theta\ T_z = u(x, z) \qquad \textbf{(2)}$$

with compact notation $T_x = \frac{\partial T}{\partial x}$ and $T_z = \frac{\partial T}{\partial z}$

Fermat principle to Eikonal equation

Minimizing with respect to θ gives

$$\sin\theta = T_x/\sqrt{T_x^2 + T_z^2} \text{ and } \cos\theta = T_z/\sqrt{T_x^2 + T_z^2}$$

Putting these expressions into equation (2) gives the eikonal equation (3)

$$T_x^2 + T_z^2 = u^2(x,z) \qquad \textbf{(3)}$$

This can be extended to 3D geometry as well

Fermat principle

$$\mathrm{T}(x,z) = \min_l \int u\big(x(l), z(l)\big)dl$$

Non linear Eikonal equation

$$T_x^2 + T_z^2 = u^2(x,z)$$

First-arrival time matches the zero-order ray time when it exists. However, such time could be evaluated when there is no ray time.

No ray ansatz and frequency power expansion

From Lakshminarayanan and Varadharajan (1997)

Viscous solution versus ray solution

Fermat principle

$$T(x,z) = \min_{s} \int u\big(x(s), z(s)\big) ds \quad \Longrightarrow \quad T_x^2 + T_z^2 = u^2(x,z)$$

Non-linear Eikonal equation

also in 3D

First-arrival time matches the zero-order ray time when it exists.

However, such time could be obtained when there is no ray time.

No ray ansatz and no high-frequency asymptotic solutions.

Such Eikonal solution is sometimes called viscous solution.

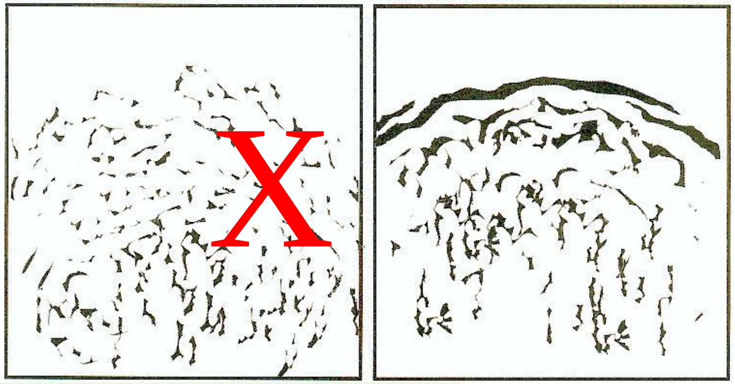
No shadow zone!

Still coherent wavefront!

Only one-value solution!

Multi-values « viscous » solution: the Graal!

Viscous solution with wavefront continuity



Viscous solution: first-arrival solution when wavefront continuity is preserved (maybe not differentiable!).

Diffraction is included: no shadow zone!

Asymptotic solution; beyond ray solution!

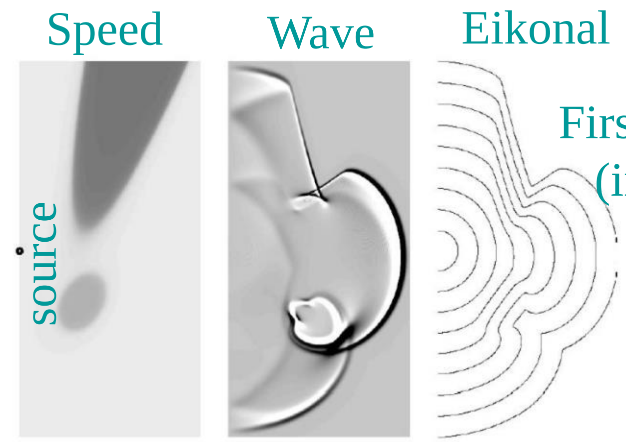
Viscous solution: efficient tools exist for computing it!

- Fast marching method O(N) for travel-times and for amplitudes
- Fast sweeping method O(N) for travel-times and for amplitudes
  - Amplitude equations have to be designed
- Finite element methods put into the scene
  - Stencils are moving to higher orders and h-adaptivity
- Discontinuous Galerkin methods
  - This is the prime road for interface investigation in the frame of PDE.

Viscous solution – first-arrival solution

A non-familiar interpretation of first phases

(often associated to HF approximation)



First phase
(initial wavefront)

Runborg, 2007)

Wave disturbance (field discontinuity)

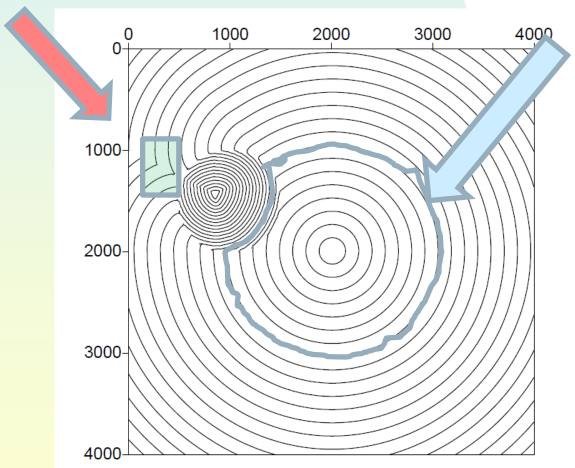
valid for any media

single value and always an answer

observable: continuous wavefronts

but possible discontinuous derivative

Line of same dansers

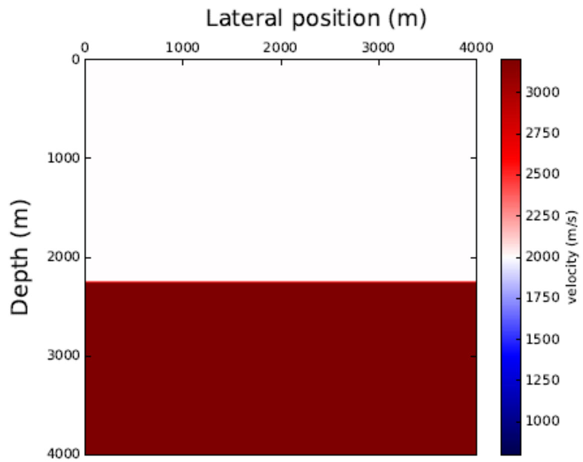


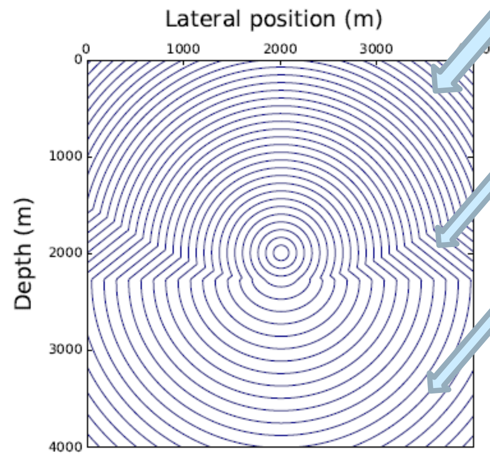
(Le Bouteiller, 2018)

Wavefront: particles moving a synchronized way. They are in phase.

Diffraction effects included ($u \propto \omega^{1/2}$)

Example of continuous wavefront





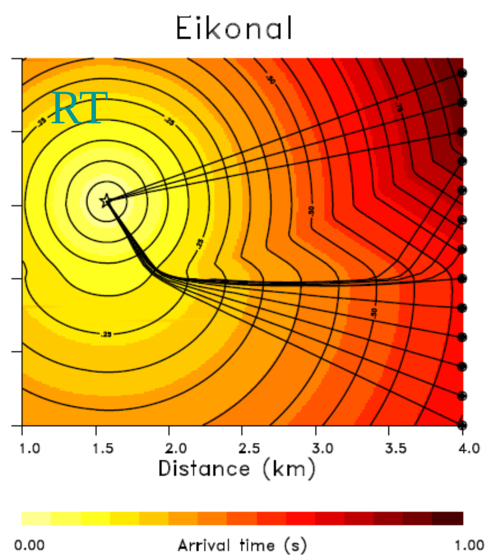
Direct phase

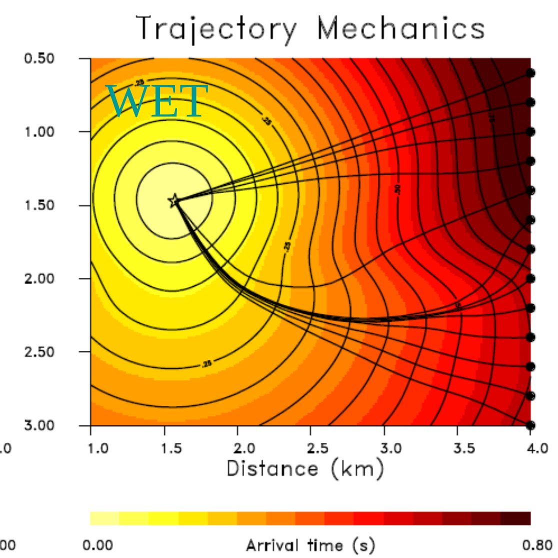
Head phase

Transmission phase

(Le Bouteiller, 2019)

Meaning of paths when considering Eikonal solutions with sharp interfaces!
ray or trajectory!

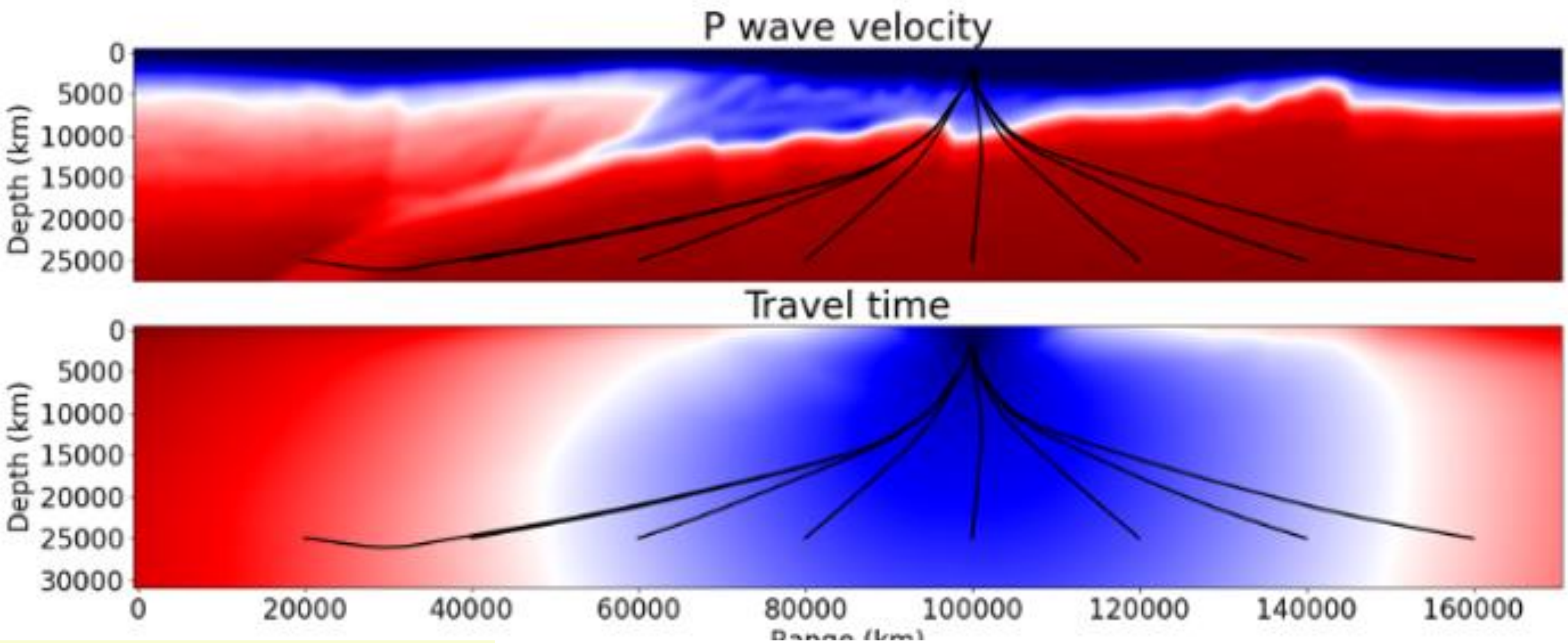




(Vasco & Nihei, 2019)

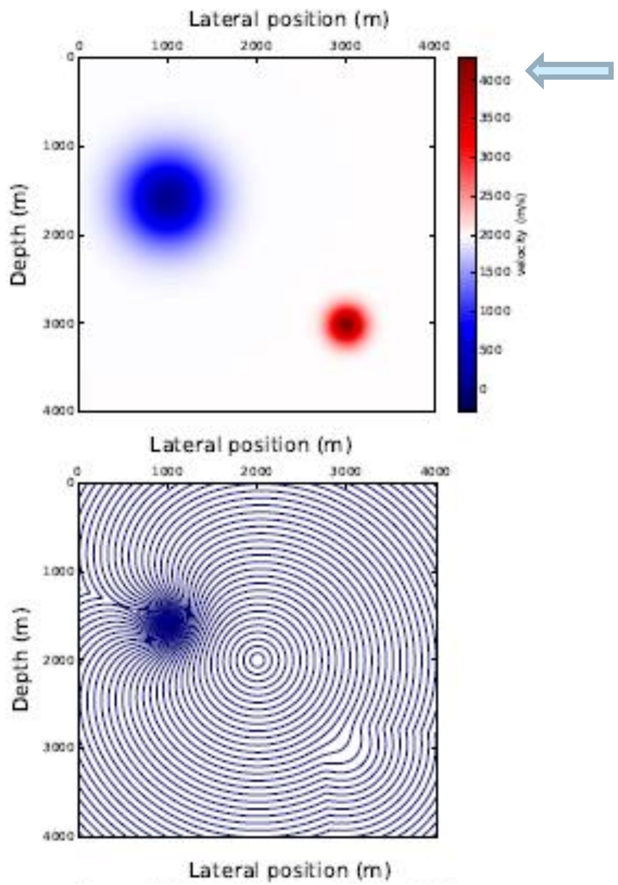
Viscous solution: minimum path…

Minimal path between two points: Fermat principle

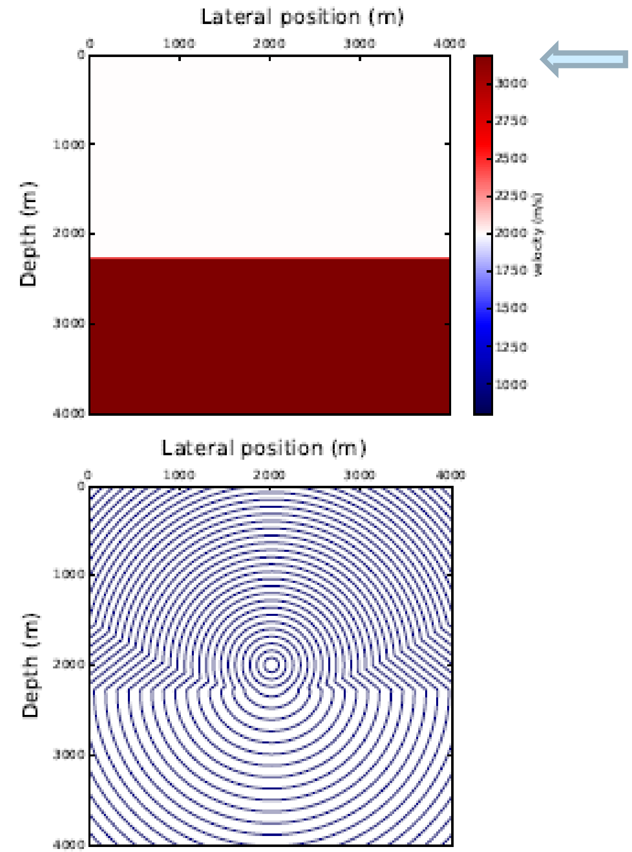


Viscous solution: examples (# ray solution)

Gaussian model



2-layers model



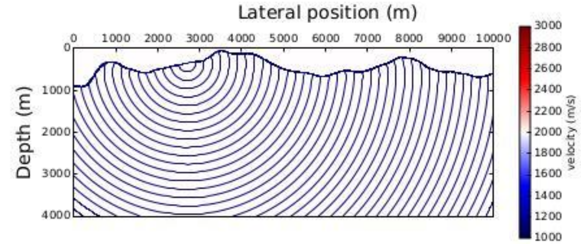
Viscous solution depends on the mesh discretization

(Le Bouteiller, 2018)

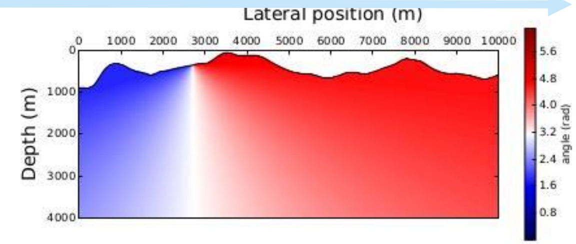
Transport equation and related PDEs

Isotropic case

Non-linear PDE: Eikonal equation $$(\nabla T)^2 - \frac{1}{c^2} = 0$$

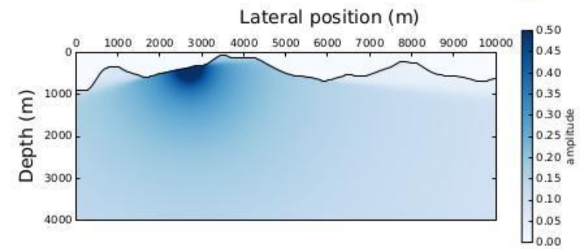


Linear PDE: take-off angle equation $$\vec{\nabla}\varphi . \overrightarrow{\nabla T} = 0$$

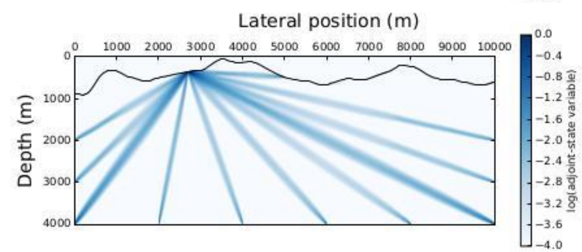


Linear PDE: amplitude equation $$\vec{\nabla} . \left(A^2 \overrightarrow{\nabla T}\right) = 0$$

« transport without dissipation »



Linear PDE: adjoint equation $$\vec{\nabla} . \left(\lambda \overrightarrow{\nabla T}\right) = \mathcal{F}$$



Le Bouteiller (2018)

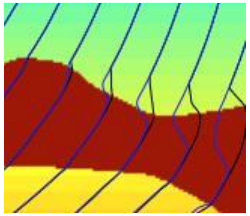
Useful family of equations

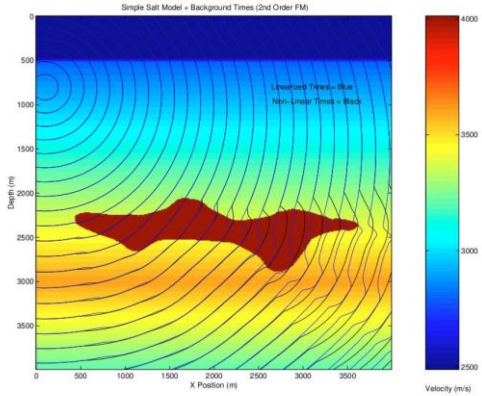
Linear Eikonal equations: time delay, angle, arclength estimation

Franklin & Harris (2001)

$$\overrightarrow{\nabla T}.\overrightarrow{\nabla \tau} = \mathcal{R}$$

Linear perturbation





Linear Transport equations: amplitude $\vec{\nabla}.\left(u\overrightarrow{\nabla T}\right) = 0$ singularities!

Belayouni (2013)

$$\vec{\nabla}.\left(\bar{u}u_0T_0\overrightarrow{\nabla \tau}\right) + u_0\overrightarrow{\nabla T_0}.\overrightarrow{\nabla(\bar{u}\tau)} = 0$$

Factorization for removing these singularities at the source (and at receivers …)

$$u = \bar{u}u_0; T = \tau T_0$$

with $known\ (analytical)\ solution\ \vec{\nabla}.\left(u_0\overrightarrow{\nabla T_0}\right) = 0$

Fast Marching Method: 3D geometry…

Computer codes available

Podvin & Lecomte (1996)

- Solving Eikonal equation for isotropic models: many codes

Cartesian and Spherical coordinates

- Solving Eikonal equation for anisotropic models: few codes
- Solving Eikonal equation and Sensitivity kernel: very few codes

Hamilton Fast Marching (HFM) and Adaptive Grid Discretizations (AGD) from Dr. Jean-Marie Mirebeau

HFM is written in C++17: Github repository (type Mirebeau and HFM on your browser)
Follow content of the file « Readme.md »
AGD is written in python & CUDA: Github repository (same location as the HFM software)
Recommendation of the installation from conda environment. See content of the file « Readme.md »

Fast Marching method: HFM illustration

Two interfaces for HFM library

Use of the FileHFM for any langage (C, Fortran …) at the expense of written files (not dramatic)

An interface with Python is included …

Attractive features in 2D geometry, 3D geometry, and on curved surface (Riemann metrics)

General anisotropy solver
Efficient TTI Eikonal solver,
Including topography through masks (or deformed Cartesian grid),
Computation of sensitivity kernels
Computation of adjoint field
Computation of rays

DO NOT WRITE YOUR OWN CODE! (**Pykonal, pyekfmm, scikit-fmm** from github … among many other codes)

Ray and Eikonal toy examples

https://github.com/jeanvirieux/Tomography_training

See the README.md

To be done (maybe … at home ?)

Analyze python codes
and
Compile HFM
Run simple examples

Take-away message

- Ray solution (multi-valued)
  When available, fruitfull for interpretation

- Viscous solution (single-valued)
  Efficient computer codes, even for anisotropy (TTI)

  No approximate solution for anisotropy …

- Viscous solution (multi-valued)?
  Still open problem for efficient numerical strategy
  Such solution is single-valued in the phase space!!!

Discrete Gradient-Eikonal algorithm

Initial velocity model s_0

$k = 0$

One loop

$k = k + 1$

Eikonal solver $time(x)$ (src)

Adjoint solver $adjoint(x)$ (src)

Gradient estimation $\gamma_k = adjoint(x) * ds(x)$

$ds_k small$

End

Linear system to be solved

$$J\,\delta s = \delta t$$
$$J^t J\,\delta s = J^t \delta t$$

$$\mathcal{H}\,\delta s = -\gamma$$

No need to store the sensitivity matrix

Only the gradient γ need to be computed and the contribution of the Hessian $\mathcal{H}$

Forward problem: frugal approach

Ray approach or Eikonal approach?

New search direction:

Neural Eikonal Solver: physics-informed Neural Network

PINNeikonal from github … (U. Waheed)

peikonal from github … (J. Calder)

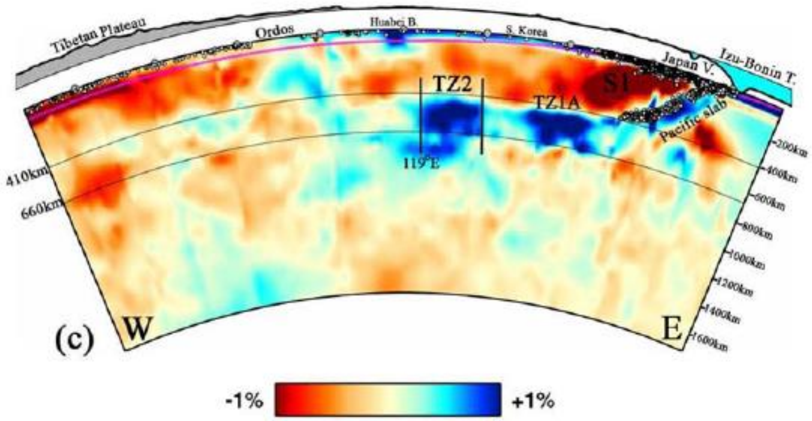
Cool feature: open road for efficient tomography strategies …

(work in progress … see relation with the third part of this presentation)

DRT versus DET: spatial difference?

Delayed ray-based Tomography

$$\delta \mathrm{T}(\mathrm{s},\mathrm{r}) = \int u(x(l))dl = \iiint u(x)\delta\big(x - x(l)\big)K(s,r,x)dv$$



Still DRT provides impressive images while we do believe that DET would provide better images in the future, thanks to the densification of the available data.

(Li & van der Hilst, 2010)

Delayed Eikonal-based Tomography?

$$\delta \mathrm{T}(s,r) = \iiint u(x)\, K(s,r,x)dv$$

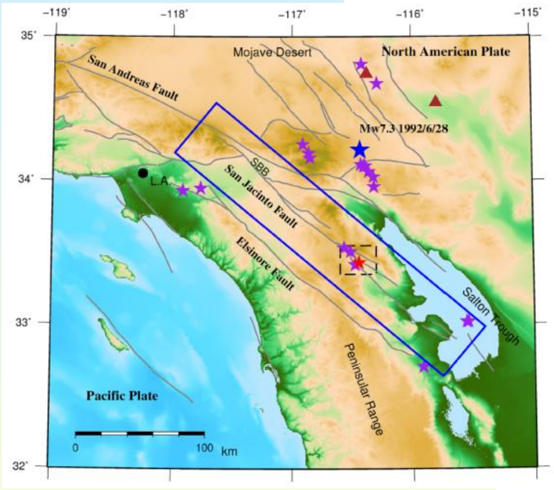
Volumic $K(s,r,x)$ still frequency-independent

DET: application to sparse dataset

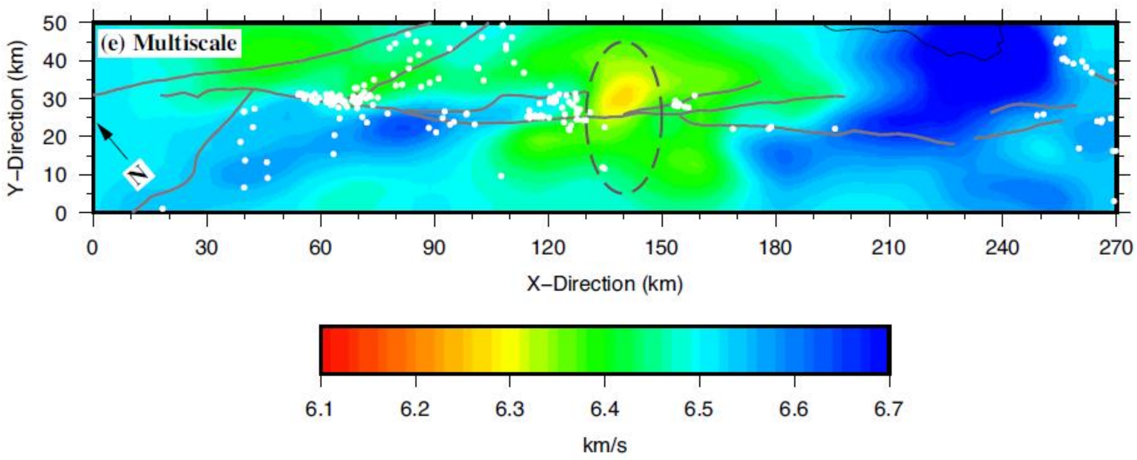
Delayed **Eikonal-based** Tomography?

$$\delta \mathrm{T}(s,r) = \iiint u(x)\, K(s,r,x)\, dv$$

Volumic $K(s,r,x)$ still frequency-independent



(Tong et al, 2019)



182 stations; 4010 quakes; 82105 P picks

DAS perspective: dense kinematic observables?

DRT versus DET: winner?

Delayed ray-based Tomography

$$\delta \mathrm{T}(\mathrm{s}, \mathrm{r}) = \int u(x(l))dl = \iiint u(x)\delta\big(x - x(l)\big)K(s,r,x)dv$$

Delayed Eikonal-based Tomography?

$$\delta \mathrm{T}(s,r) = \iiint u(x)\, K(s,r,x)dv$$

Volumic $K(s,r,x)$ still frequency-independent

Delayed Eikonal-based tomography has the same computational complexity than Delayed Ray-based tomography (high-frequency versus frequency-immune).

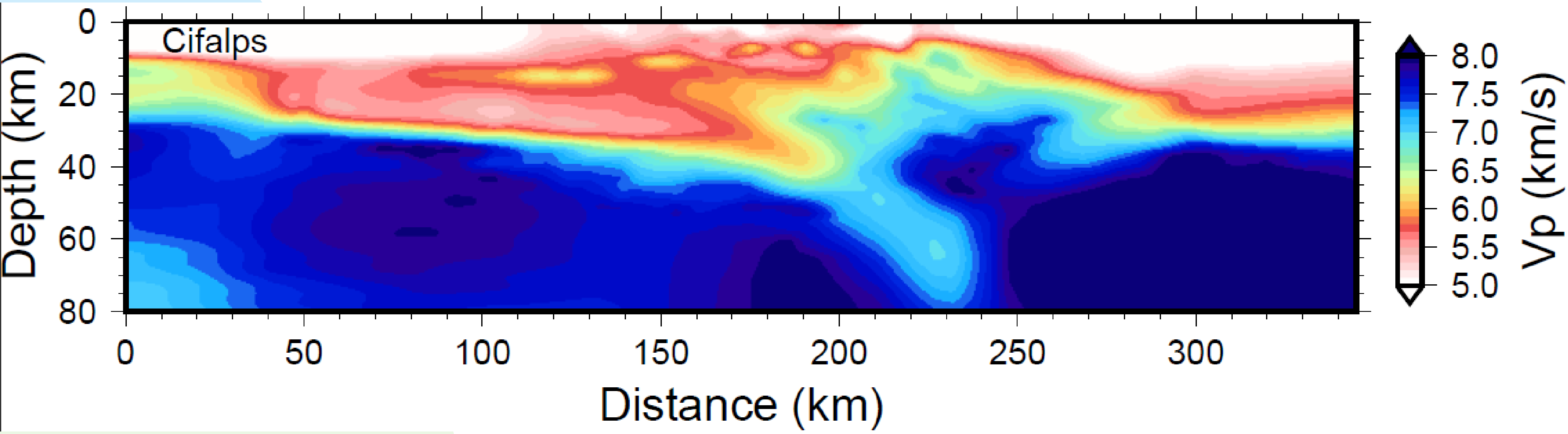
Both are agnostic to frequency content of seismic waves … blue-sky information.

First-arrival traveltime tomography – Transmission regime

Large-scale velocity model

Large-scale model building

Regional/Local tomographical model



P-wave and S-wave velocities

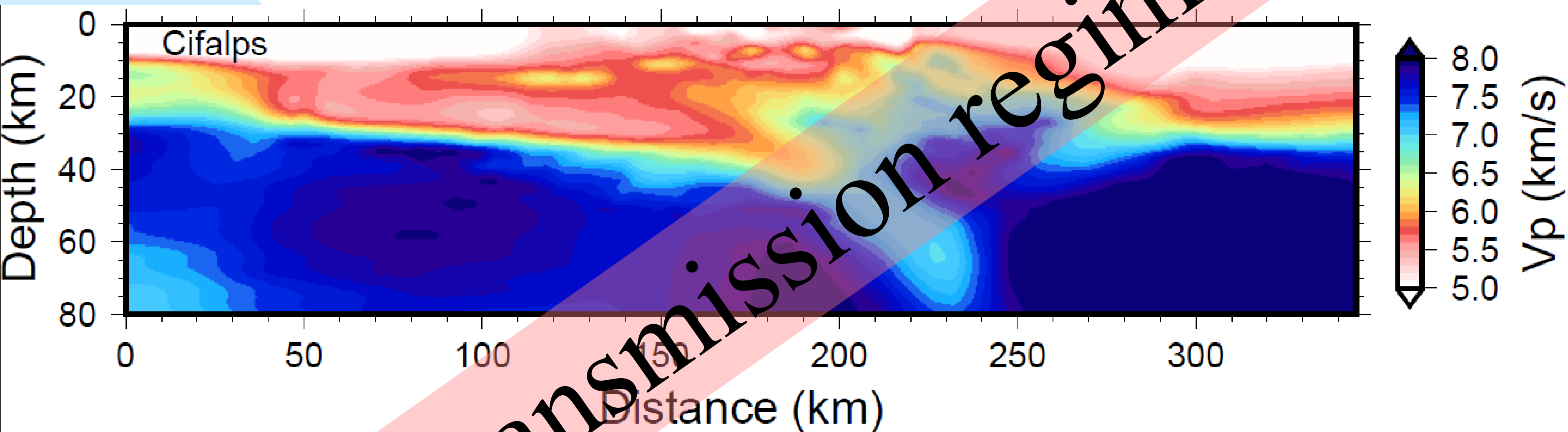
Density?

Computing Green functions!!!

Ingredients for wave propagation

Large-scale model building

Regional/Local tomographical model



Still transmission regime

Widely used tool for model building!!!

Characterization of crustal models for quantitative ground motion estimation

C – Model Design
(short-scale velocity structure)

Jean Virieux

Emeritus Professor at UGA

Some slides are inspired from Seiscope Group (PIs Romain Brossier & Ludovic Métivier