Resolution Analysis By Random Probing

Andreas Fichtner
ETH Zurich

Tristan van Leeuwen
Utrecht University

"Solving an inverse problem means to describe the infinite-dimensional space of data-fitting models."

George Backus & Freeman Gilbert, 1968

1. Why resolution analysis is becoming more and more difficult

A simple example

The Resolution Matrix

$$\mathbf{R}\ \mathbf{m}_{true} = \mathbf{m}_{est}$$

Resolution matrix How the true Earth is smeared into an image. Dimension N × N.

True Earth model Dimension N.

Estimated Earth model Dimension N.

THE RESOLUTION MATRIX

$$\mathbf{R} \quad \mathbf{m}_{true} = \mathbf{m}_{est}$$

Resolution matrix How the true Earth is smeared into an image. Dimension N × N.

True Earth model Dimension N.

Estimated Earth model Dimension N.

- In the days of Backus & Gilbert: N = $\mathcal{O}(10^2)$ → **R** is $\mathcal{O}(10^2)$ times larger than **m**.

The Resolution Matrix

$$\mathbf{R} \ \mathbf{m}_{true} = \mathbf{m}_{est}$$

Resolution matrix How the true Earth is smeared into an image. Dimension N × N.

True Earth model Dimension N.

Estimated Earth model Dimension N.

- In the days of Backus & Gilbert: N = $\mathcal{O}(10^2)$ → **R** is $\mathcal{O}(10^2)$ times larger than **m**.
- Today: N = $\mathcal{O}(10^7)$ → **R** is $\mathcal{O}(10^7)$ times larger than **m**.

THE RESOLUTION MATRIX

$$\mathbf{R} \ \mathbf{m}_{true} = \mathbf{m}_{est}$$

Resolution matrix How the true Earth is smeared into an image. Dimension N × N.

True Earth model Dimension N.

Estimated Earth model Dimension N.

- In the days of Backus & Gilbert: N = $\mathcal{O}(10^2)$ → **R** is $\mathcal{O}(10^2)$ times larger than **m**.
- Today: N = $\mathcal{O}(10^7)$ → **R** is $\mathcal{O}(10^7)$ times larger than **m**.

- As data volumes and computing power grow:
  - ➢ We can construct bigger and bigger models $\mathbf{m}_{est}$.

The Resolution Matrix

$$\mathbf{R} \; \mathbf{m}_{true} = \mathbf{m}_{est}$$

Resolution matrix How the true Earth is smeared into an image. Dimension N × N.

True Earth model Dimension N.

Estimated Earth model Dimension N.

The problem:

- In the days of Backus & Gilbert: N = $\mathcal{O}(10^2)$ → **R** is $\mathcal{O}(10^2)$ times larger than **m**.
- Today: N = $\mathcal{O}(10^7)$ → **R** is $\mathcal{O}(10^7)$ times larger than **m**.

- As data volumes and computing power grow:
  - We can construct bigger and bigger models $\mathbf{m}_{est}$.
  - We loose our ability to quantify the quality of $\mathbf{m}_{est}$.

We need **scalable** methods to infer useful **aspects** of resolution.

We need **scalable** methods to infer useful **aspects** of resolution.

Objectives of this Webinar:

- Describe 2 methods to quantify resolution when **R** is too expensive to compute and too big to store.
- One method for linear problems, and one for (mildly) nonlinear problems.
- Both based on random probing techniques.

2. Estimating the number of resolved parameters

tr **R**

Estimating The Number Of Resolved Parameters

m_i

- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

Estimating The Number Of Resolved Parameters

m_i

- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

R_{ij}

- A resolution matrix

Too large to computer.
Too large to store.
Too large to comprehend fully.

Estimating The Number Of Resolved Parameters

m_i

- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

R_{ij}

- A resolution matrix

$E[m_i R_{ij} m_j]$

Estimating The Number Of Resolved Parameters

m_i

- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

R_{ij}

- A resolution matrix

$E[m_i R_{ij} m_j] = R_{ij}\, E[m_i m_j]$

Estimating The Number Of Resolved Parameters

m_i
- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

R_{ij}
- A resolution matrix

$$E[m_i R_{ij} m_j] = R_{ij}\, E[m_i m_j]$$
$$= R_{ij} \left(E[m_i]E[m_j] + cov(m_i,m_j) \right)$$

ESTIMATING THE NUMBER OF RESOLVED PARAMETERS

m_i

- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

R_{ij}

- A resolution matrix

$$E[m_i R_{ij} m_j] = R_{ij}\, E[m_i m_j]$$
$$= R_{ij}\, (\, E[m_i]E[m_j] + cov(m_i,m_j)\,)$$
$$= R_{ij}\, \delta_{ij} = R_{ii} = \mathrm{tr}\, \mathbf{R}$$

Estimating The Number Of Resolved Parameters

m_i
- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

R_{ij}
- A resolution matrix

$$E[m_i R_{ij} m_j] = R_{ij}\, E[m_i m_j]$$
$$= R_{ij}\, (\, E[m_i]E[m_j] + cov(m_i,m_j)\,)$$
$$= R_{ij}\, \delta_{ij} = R_{ii} = \text{tr}\, \mathbf{R} = \text{number of resolved model parameters}$$

Hutchinson's method [Hutchinson, 1990]

Estimating The Number Of Resolved Parameters

m_i

- random test model vector
- Expectation: $E[m_i]=0$
- Covariance: $cov(m_i,m_j)=\delta_{ij}$ [uncorrelated components]

R_{ij}

- A resolution matrix

$$E[m_i R_{ij} m_j] = R_{ij}\, E[m_i m_j]$$
$$= R_{ij} \left(E[m_i]E[m_j] + cov(m_i,m_j) \right)$$
$$= R_{ij}\, \delta_{ij} = R_{ii} = \text{tr}\, \mathbf{R} = \text{number of resolved model parameters}$$

<u>Very simple recipe</u>:

- Choose a random test model **m**.
- Try to recover this model in a synthetic inversion [i.e. compute $\mathbf{m}_{est} = \mathbf{Rm}$].
- Multiply the result with **m** itself: $\mathbf{m}^T \mathbf{m}_{est} = \mathbf{m}^T\mathbf{Rm}$.
- Average over some random realisations.

- The resolution matrix itself never has to be computed!

Other Random Probing Techniques

Hutchinson, M. F. (1990), A stochastic estimator of the trace of the influence matrix for Laplacian smoothing splines, Comm. Stat. Sim., 19, 433–450.

An, M. (2012), A simple method for determining the spatial resolution of a general inverse problem, Geophys. J. Int., 191, 849–864.

Avron, H., and S. Toledo (2011), Randomized algorithms for estimating the trace of an implicit symmetric positive semi-definite matrix, J. Ass. Comp. Mach., 58, doi:10.1145/1944,345.

Drineas, P., R. Kannan, and M. W. Mahoney (2006), Fast Monte Carlo algorithms for matrices II: Computing a low-rank approximation to a matrix, SIAM J. Comput., 36, 158–183.

Frieze, A., R. Kannan, and S. Vempala (2004), Fast Monte Carlo algorithms for finding low-rank approximations, J. Assoc. Comput. Mach., 51, 1025–1041.

Halko, N., P. G. Martinsson, and J. A. Tropp (2011), Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions, SIAM Review, 53, 217–288.

MacCarthy, J. K., B. Borchers, and R. C. Aster (2011), Efficient stochastic estimation of the model resolution matrix diagonal and generalized cross-validation for large geophysical inverse problems, J. Geophys. Res., 116, doi: 1029/2011JB008,234.

Trampert, J., and A. Fichtner (2013), Resolution tests revisited: The power of random numbers, Geophys. J. Int., 192, 676–680.

3. Random probing for resolution analysis in tomography

Estimating position- and direction-dependent resolution lengths.

Point-Spread Functions

- Misfit χ in the vicinity of the optimal model **m**:

$$\chi(\mathbf{m}+\delta\mathbf{m}) = \chi(\mathbf{m}) + \frac{1}{2}\delta\mathbf{m}^T\mathbf{H}(\mathbf{m})\,\delta\mathbf{m}$$

Hessian operator
Inverse posterior covariance [assuming Gaussian errors]
Column: point-spread function

H is too expensive to compute and store.

- **But we can** infer properties of **H** from its application to random test models.

Random Probing Principle

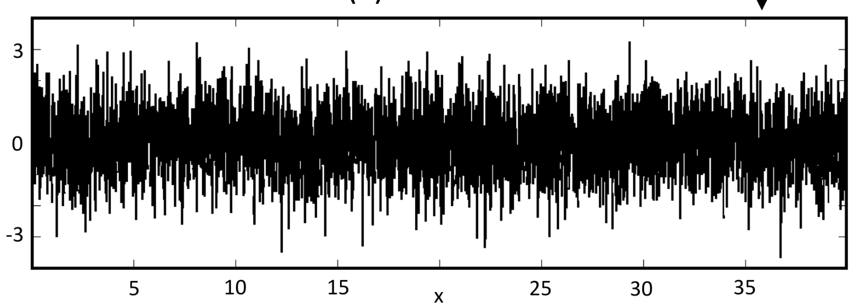
- Assume **H** is Gaussian [for simplicity and illustration]:

$$H(x;y) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2\sigma^2}(x-y)^2}$$

$$h(y) = \int H(x;y)v(x)\,dx$$

random test model

random test model v(x)



Random Probing Principle

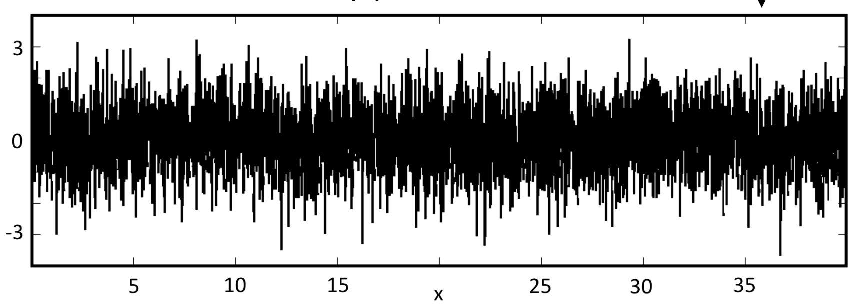
- Assume **H** is Gaussian [for simplicity and illustration]:

$$H(x;y) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-\frac{1}{2\sigma^2}(x-y)^2}$$

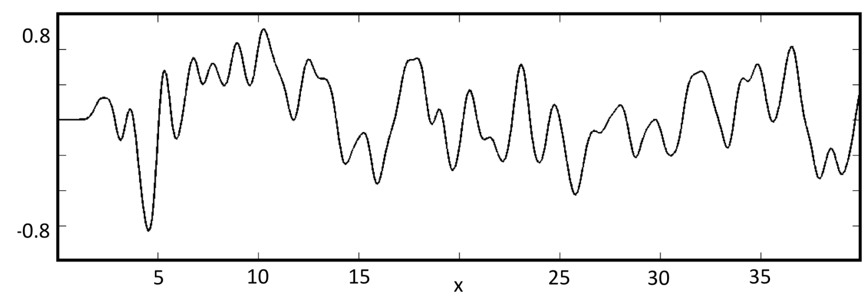
$$h(y) = \int H(x;y) v(x)\, dx$$

random test model

random test model v(x)



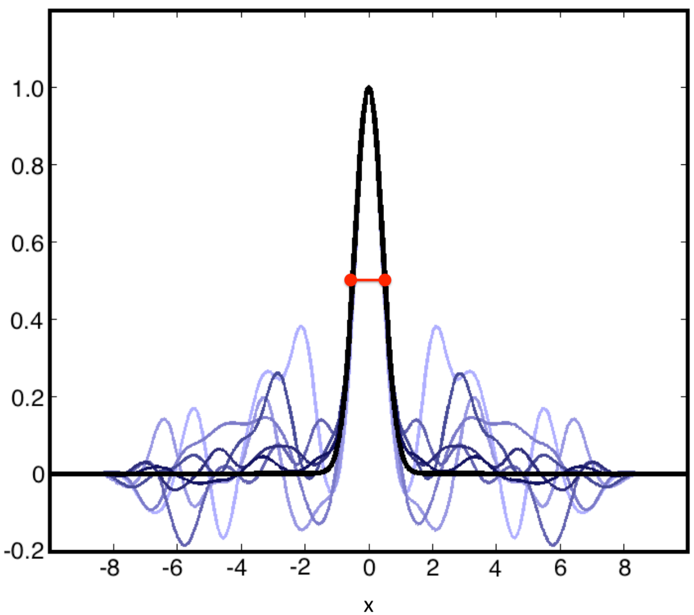
smoothed version of v(x) [h=H•v]



- Length scales of **h** contain information on length scales of **H**.

- **Auto-correlation** of the output **h** [averaged over many realisations]:

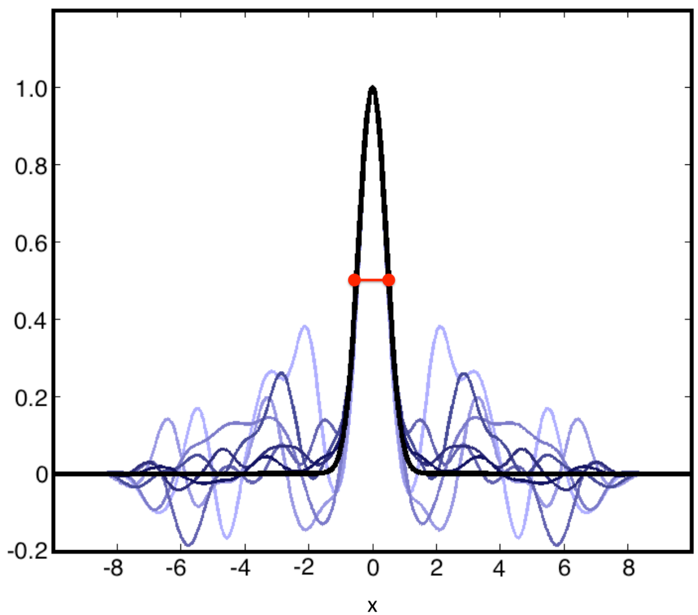
average auto-correlations of h
[for 1, 2, 3, 5, 10, 20, 50 samples]



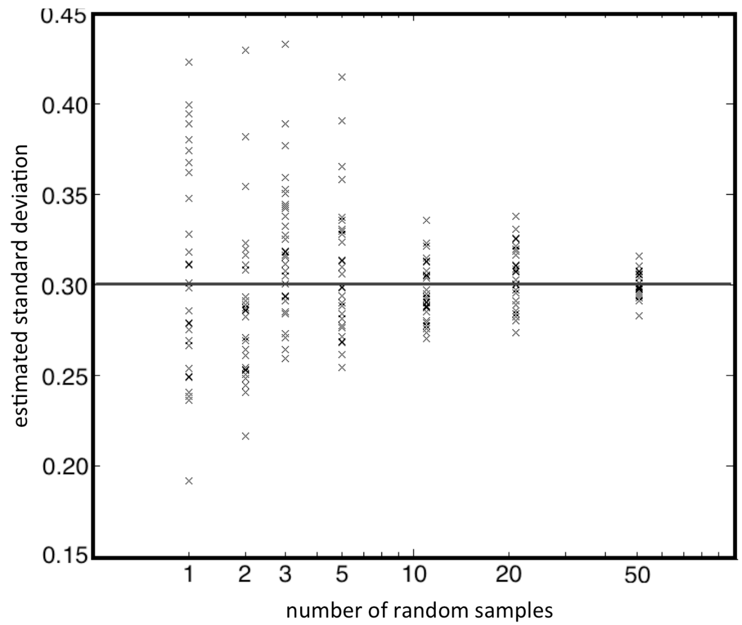
RANDOM PROBING PRINCIPLE

- **Auto-correlation** of the output **h** [averaged over many realisations]:
- **Asymptotically**: width of auto-correlation = √2 • width of **H**

average auto-correlations of h
[for 1, 2, 3, 5, 10, 20, 50 samples]



estimated with of **H**
[for 1, 2, 3, 5, 10, 20, 50 samples]



PRELIMINARY CONCLUSIONS

1. Resolution and correlations

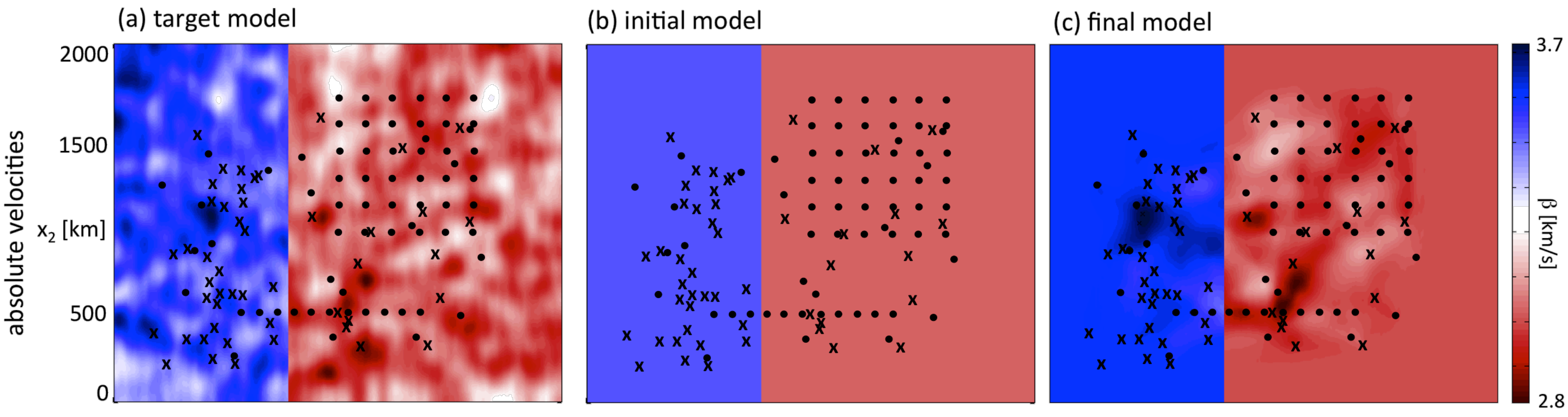
   - The Hessian acts as a smoother of random functions.
   - The smoothed functions carry information on **resolution.**
   - Can be **extracted with correlations**.

2. Convergence

   - Correlations themselves may require large sample sizes to converge.
   - The **width** of the correlation **converges extremely quickly**.
   - ➢ Useful **resolution proxies** may already be obtained with **very few samples.**

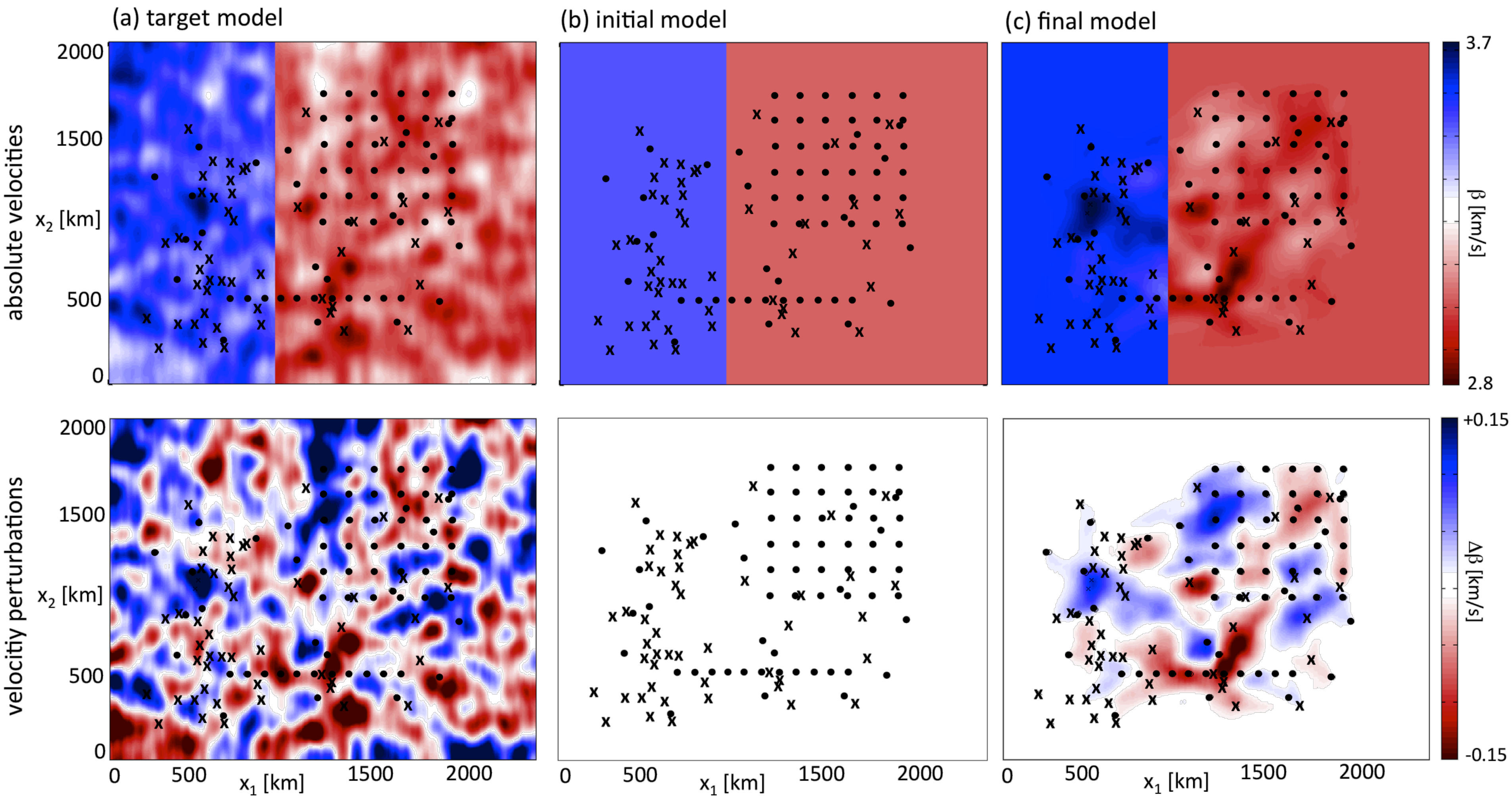
Synthetic full-waveform inversion in 2D

- Synthetic inversion setup



SYNTHETIC EXAMPLE IN 2D

- Synthetic inversion setup

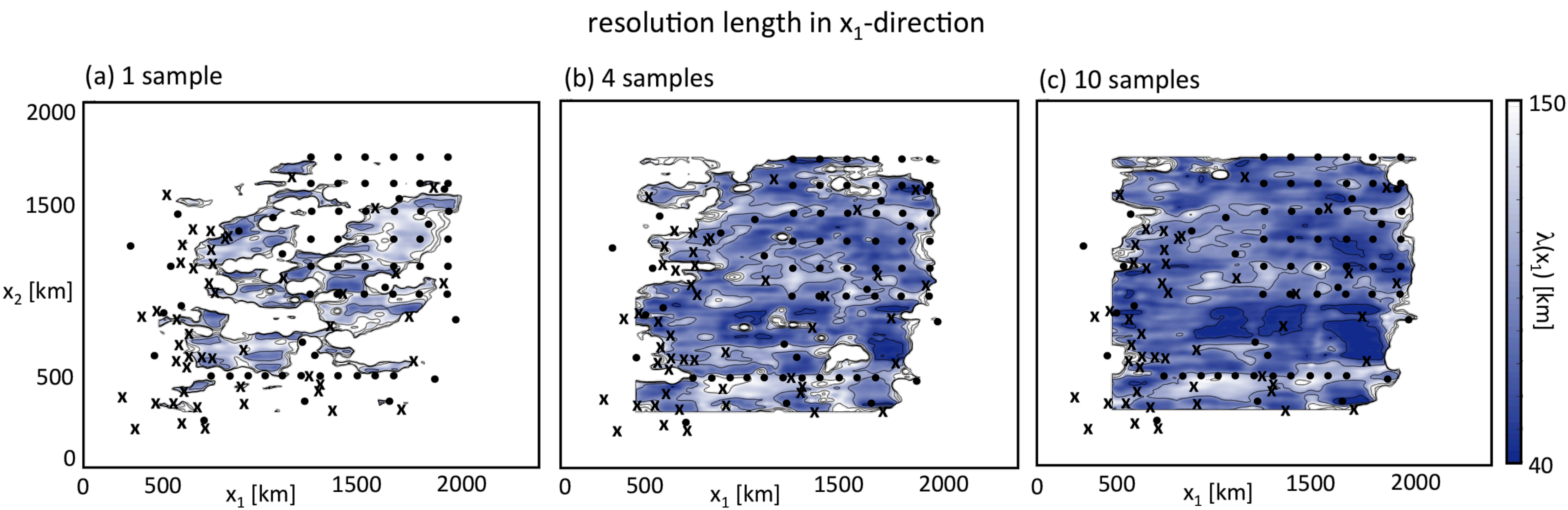


SYNTHETIC EXAMPLE IN 2D

- Application of random test models to the Hessian via second-order adjoints
- Local auto-correlation of the output in different directions.

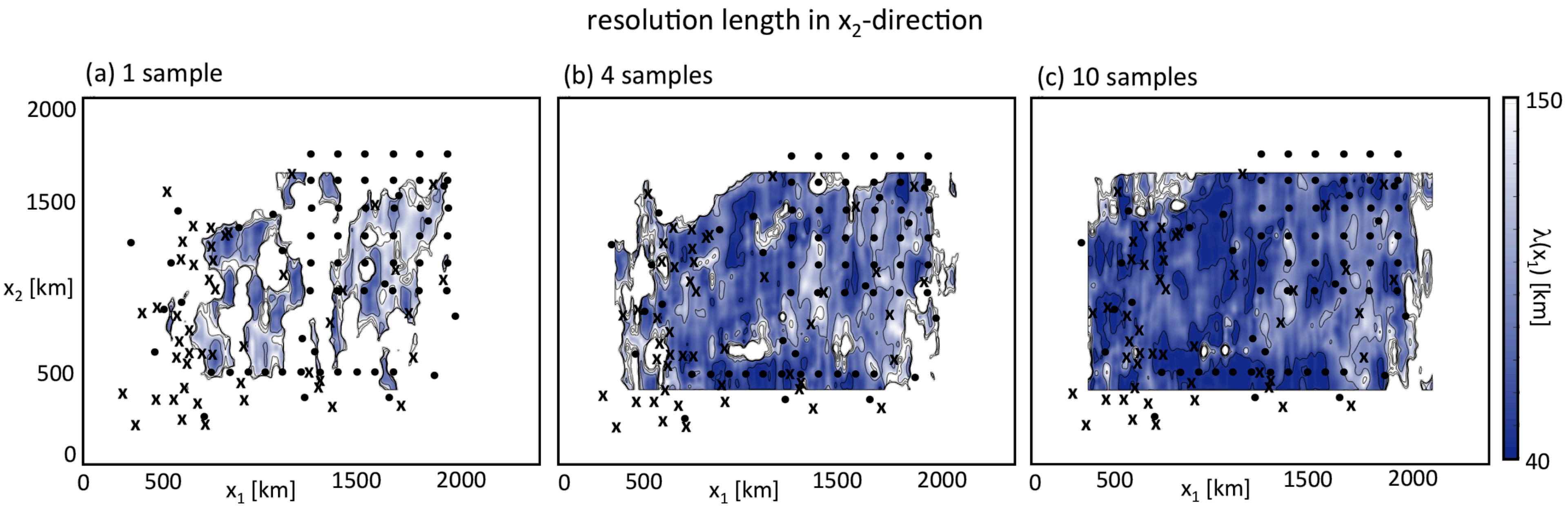
SYNTHETIC EXAMPLE IN 2D

- Application of random test models to the Hessian via second-order adjoints.
- Local auto-correlation of the output in different directions.
- Estimated width of the point-spread functions in **x_1-direction** [resolution length].



- Around 5-10 samples to converge.
- Resolution is strongly heterogeneous.

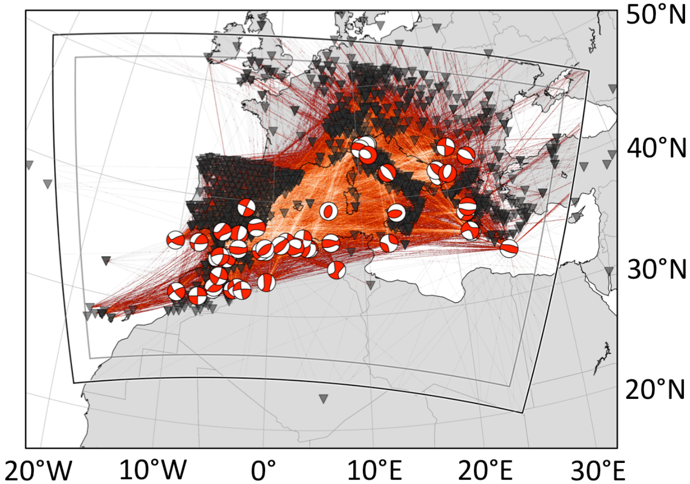
- Estimated width of the point-spread functions in **x_2-direction** [resolution length].



Real-data application

INVERSION SETUP

surface wave ray coverage



Technical summary:

Data

- 52 earthquakes, >1000 stations
- body waves, surface waves, ...
- periods: **10 – 150 s**

Forward modelling

- spectral elements
- **3D visco-elastic, anisotropic**

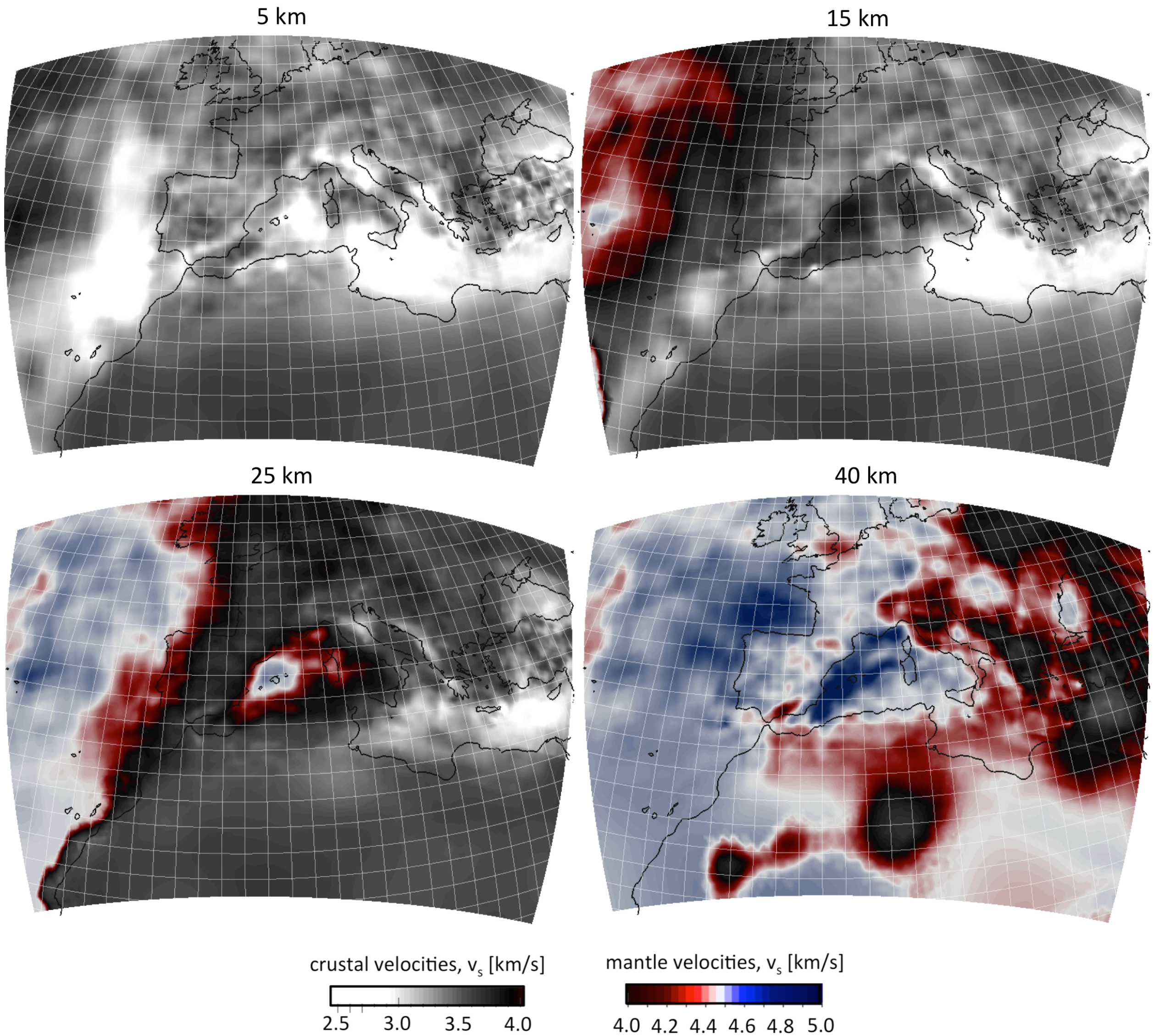
Inversion

- initial model from previous European FWI
- adjoint-based CG
- invert for v_{sh}, v_{sv}, v_p, ρ **and** source location/mechanism

Fichtner & Villasenor, EPSL 2015.

S Velocity Model

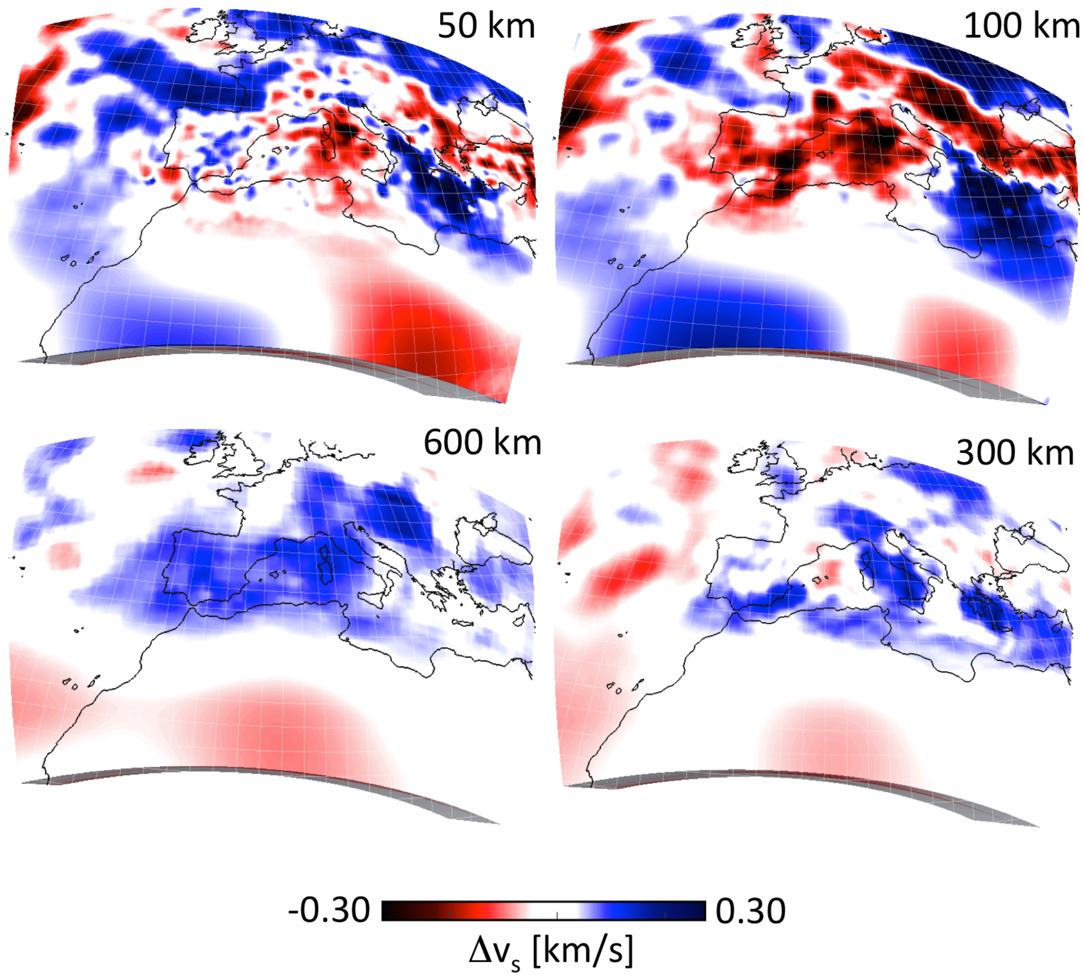
isotropic S velocity



Fichtner & Villasenor, EPSL 2015.

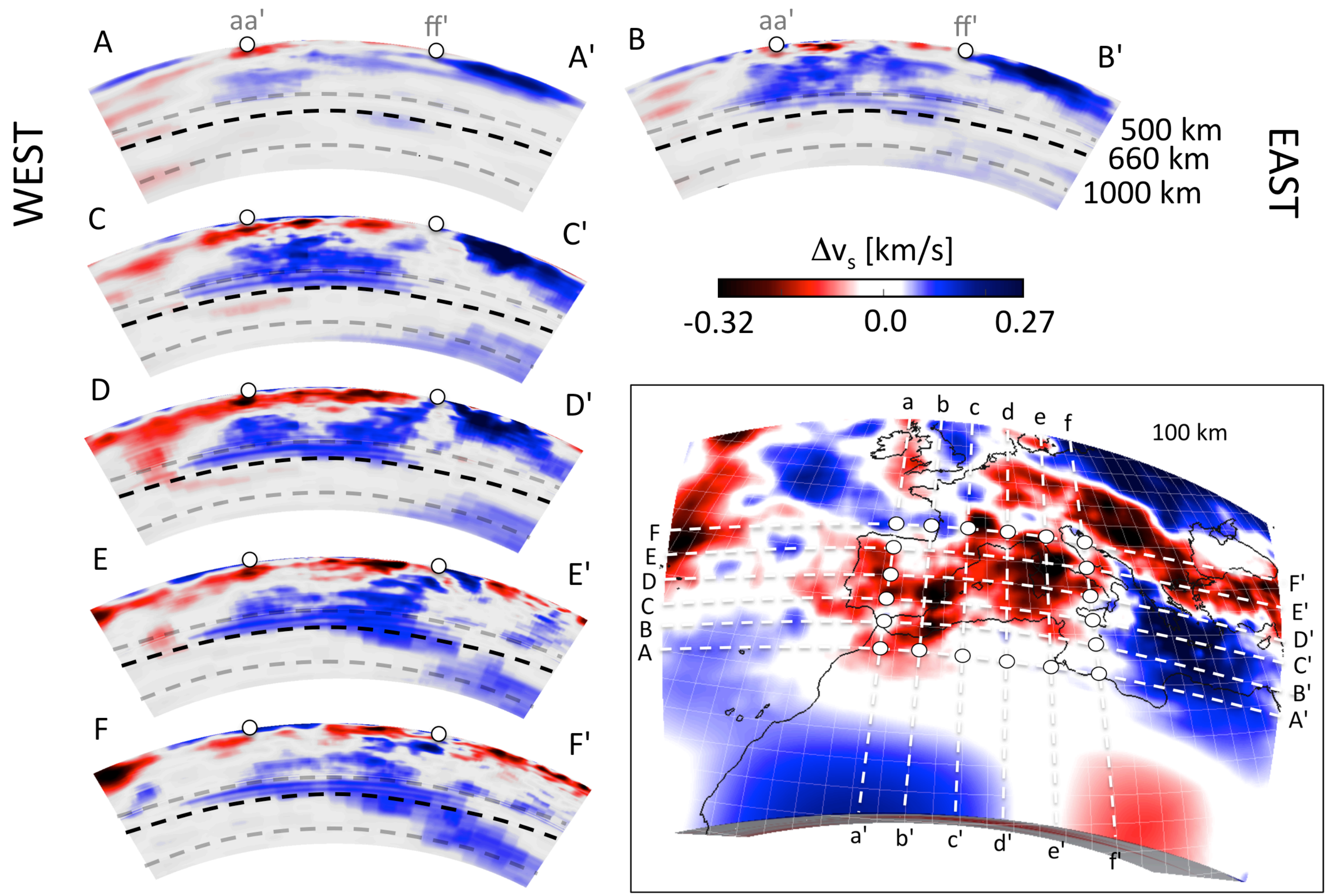
S Velocity Model

isotropic S velocity variations



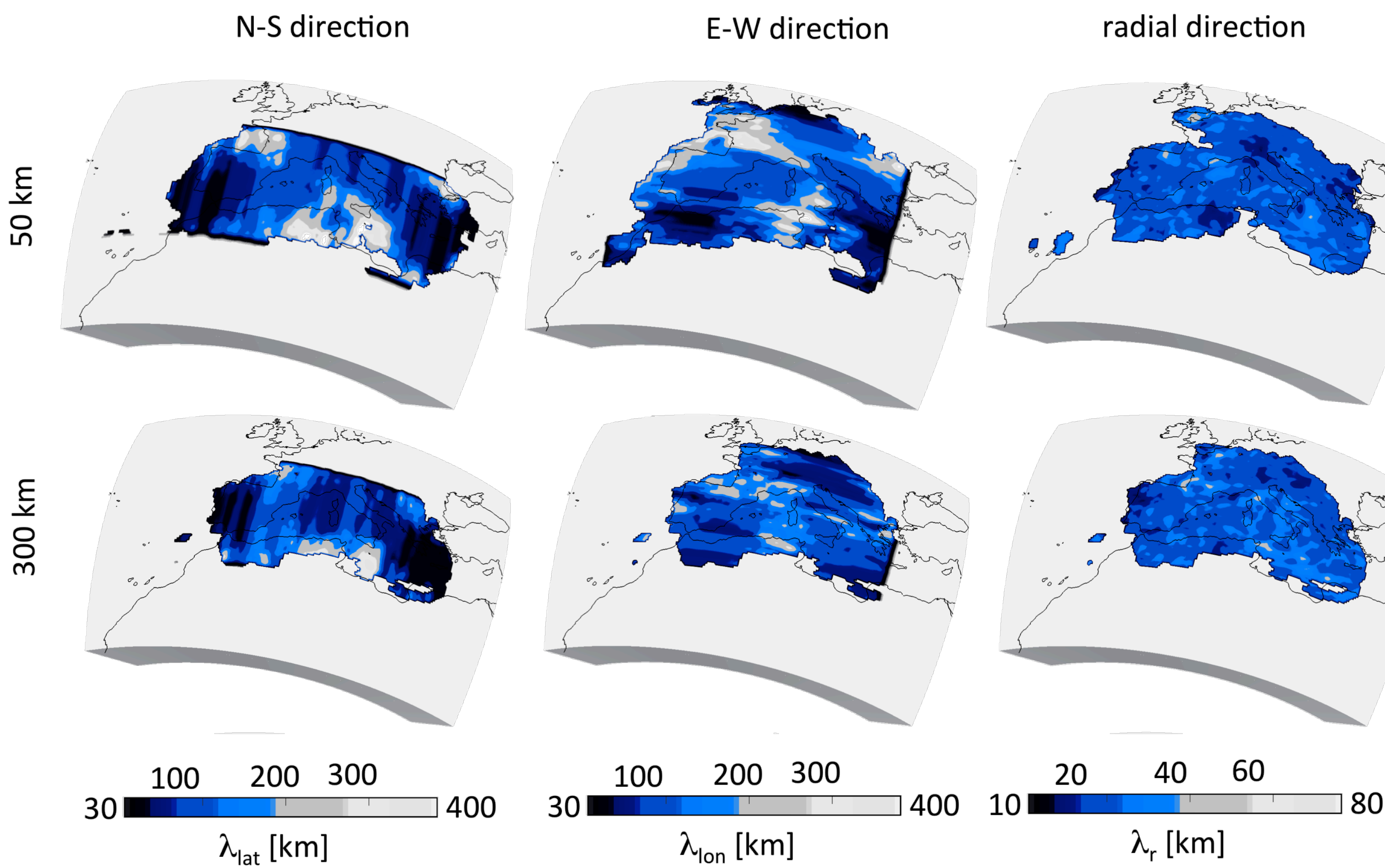
Fichtner & Villasenor, EPSL 2015.

S Velocity Model



Fichtner & Villasenor, EPSL 2015.

Position- and Direction-Dependent Resolution Lengths



Conclusions

Limitations:

1. Local analysis

Limitations:

1. Local analysis

Benefits:

1. Quantify spatial resolution and inter-parameter trade-offs.

Limitations:

1. Local analysis

Benefits:

1. Quantify spatial resolution and inter-parameter trade-offs.

2. Low computational costs

    - around 5 Hessian-model applications
    - equivalent to around 5 CG iterations
    - much less than a synthetic inversion

Limitations:

1. Local analysis

Benefits:

1. Quantify spatial resolution and inter-parameter trade-offs.

2. Low computational costs

    - around 5 Hessian-model applications
    - equivalent to around 5 CG iterations
    - much less than a synthetic inversion

3. Low algorithmic complexity

    - easy to implement without modifications of existing codes

Limitations:

1. Local analysis

Benefits:

1. Quantify spatial resolution and inter-parameter trade-offs.

2. Low computational costs

   - around 5 Hessian-model applications
   - equivalent to around 5 CG iterations
   - much less than a synthetic inversion

3. Low algorithmic complexity

   - easy to implement without modifications of existing codes

4. Scalability

   - 5 random models sufficient in 1, 2 and 3 dimensions [empirical]

Limitations:

1. Local analysis

Benefits:

1. Quantify spatial resolution and inter-parameter trade-offs.

2. Low computational costs

   - around 5 Hessian-model applications
   - equivalent to around 5 CG iterations
   - much less than a synthetic inversion

3. Low algorithmic complexity

   - easy to implement without modifications of existing codes

4. Scalability

   - 5 random models sufficient in 1, 2 and 3 dimensions [empirical]

5. Applicability to any tomographic technique

Thanks for your attention!