Bayesian approach in seismic tomography

Trond Ryberg
GFZ Potsdam

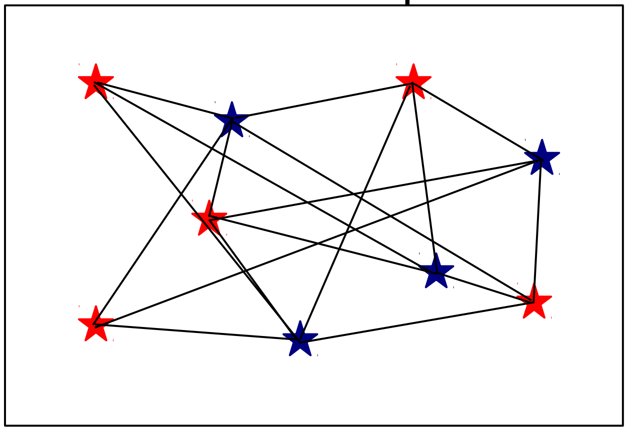
Collaborator: Ch. Haberland, GFZ

Outline

- Classic tomography
- Motivation: Why statistical methods?
- Likelihood function
- Bayesian approach
- The prior information
- Ensemble inference
- 2D Problem: Model parametrization & forward calculation
- Markov Chains
- Metropolis Hastings Algorithm
- Synthetic example
- Extension: Transdimensional rj-algorithm, example #2
- Extension: Hierarchical Bayesian framework, example #3
- Some more examples...
- Conclusions

Classic tomography

2D example

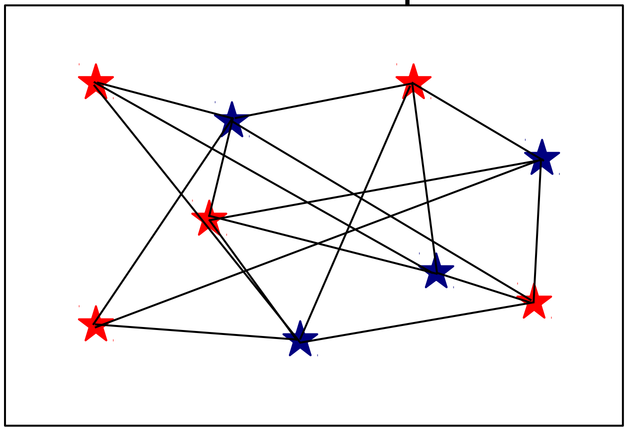


Given:
set of travel times for source-receiver pairs $\mathbf{d}_{obs}=T_i$

Wanted:
field of slowness/velocities $\mathbf{m}$ explaining the data

Classic tomography

2D example



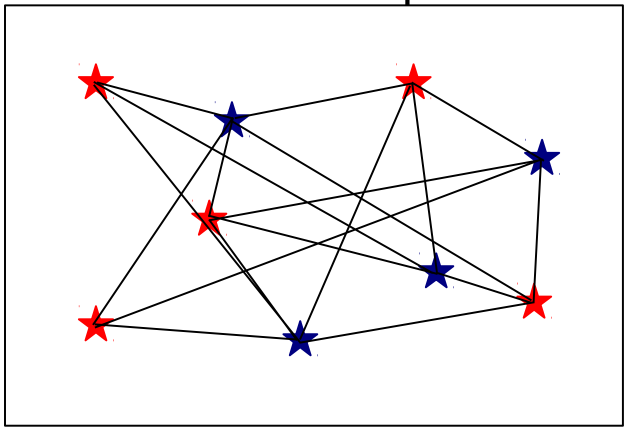
Given:
set of travel times for source-receiver pairs $\mathbf{d}_{obs}=T_i$

Wanted:
field of slowness/velocities **m** explaining the data

Assuming that we have a method to:
predict travel times for model **m** by forward modeling: $\mathbf{d}_{pred}=\mathbf{g}(\mathbf{m})$

Classic tomography

2D example



Given:
set of travel times for source-receiver pairs $\mathbf{d}_{obs}=T_i$

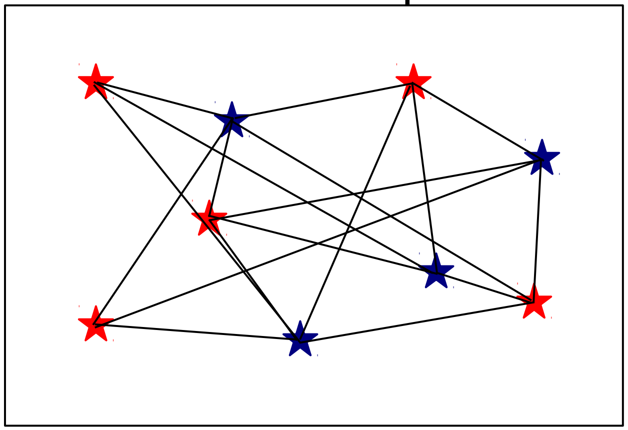
Wanted:
field of slowness/velocities **m** explaining the data

Assuming that we have a method to:
predict travel times for model **m** by forward modeling: $\mathbf{d}_{pred}=\mathbf{g}(\mathbf{m})$

Misfit calculation: $$\phi(m)=\left\|\frac{d_{pred}-d_{obs}}{\sigma_d}\right\|^2=\left\|\frac{g(m)-d_{obs}}{\sigma_d}\right\|^2$$ with σ_d variance of data noise

Classic tomography

2D example



Given:
set of travel times for source-receiver pairs $\mathbf{d}_{obs}=T_i$

Wanted:
field of slowness/velocities **m** explaining the data

Assuming that we have a method to:
predict travel times for model **m** by forward modeling: $\mathbf{d}_{pred}=\mathbf{g}(\mathbf{m})$

Misfit calculation: $$\phi(m)=\left\|\frac{d_{pred}-d_{obs}}{\sigma_d}\right\|^2=\left\|\frac{g(m)-d_{obs}}{\sigma_d}\right\|^2$$ with σ_d variance of data noise

Task:

$$\phi=\left\|\frac{g(m)-d_{obs}}{\sigma_d}\right\|^2 \rightarrow minimum$$

Classic tomography

more constraints (regularization):

$$\left\| \frac{g(m) - d_{obs}}{\sigma_d} \right\|^2 + \alpha \| m - m_0 \|^2 + \beta \| \nabla m \|^2 + \ldots \rightarrow minimum$$

data misfit

deviation from starting model

smoothness

- classic optimization problem
- (highly) non-linear problem!
- inverse problem typically ill-posed , i.e. solution is non-unique and/or ill-conditioned
- assumptions (prior information) needed:

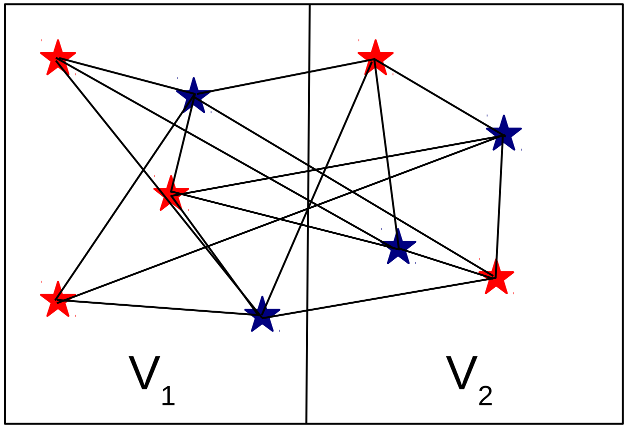
  starting model, geometric parameters of grid/mesh,

  damping, smoothing, data noise level, etc.

Motivation: Why statistical approach?

assuming a very simple inversion problem:

tomography in a 2 block model

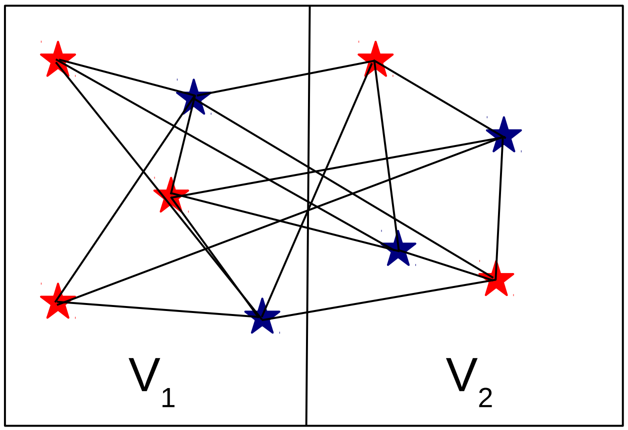


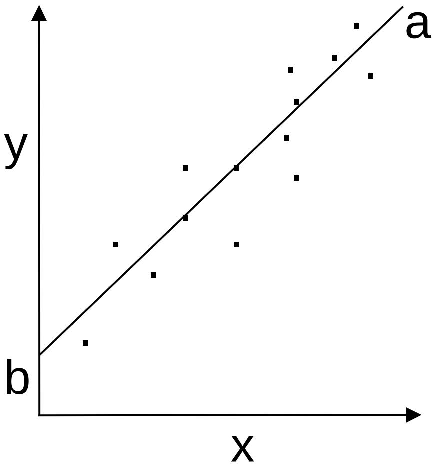
Modified from Sambridge & Mosegaard, 2002

Motivation: Why statistical approach?

assuming a very simple inversion problem:

tomography in a 2 block model or linear regression of point cloud: y=a*x+b, with slope **a** and intercept **b**





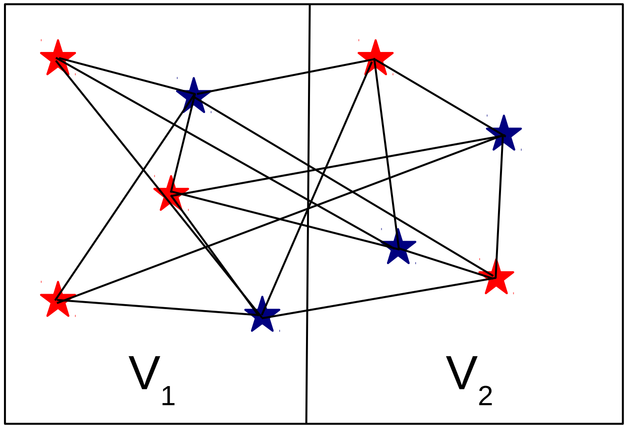
Modified from Sambridge & Mosegaard, 2002

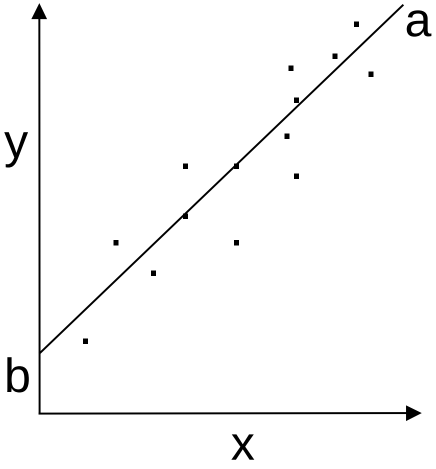
Motivation: Why statistical approach?

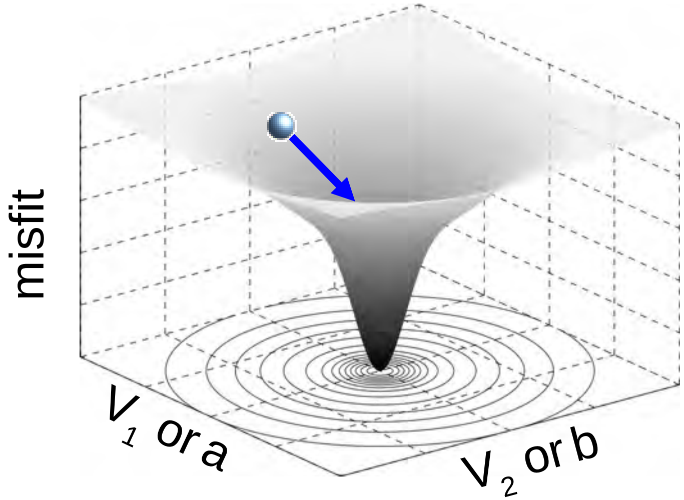
assuming a very simple inversion problem:

tomography in a 2 block model or

linear regression of point cloud:
y=a*x+b, with slope **a** and intercept **b**



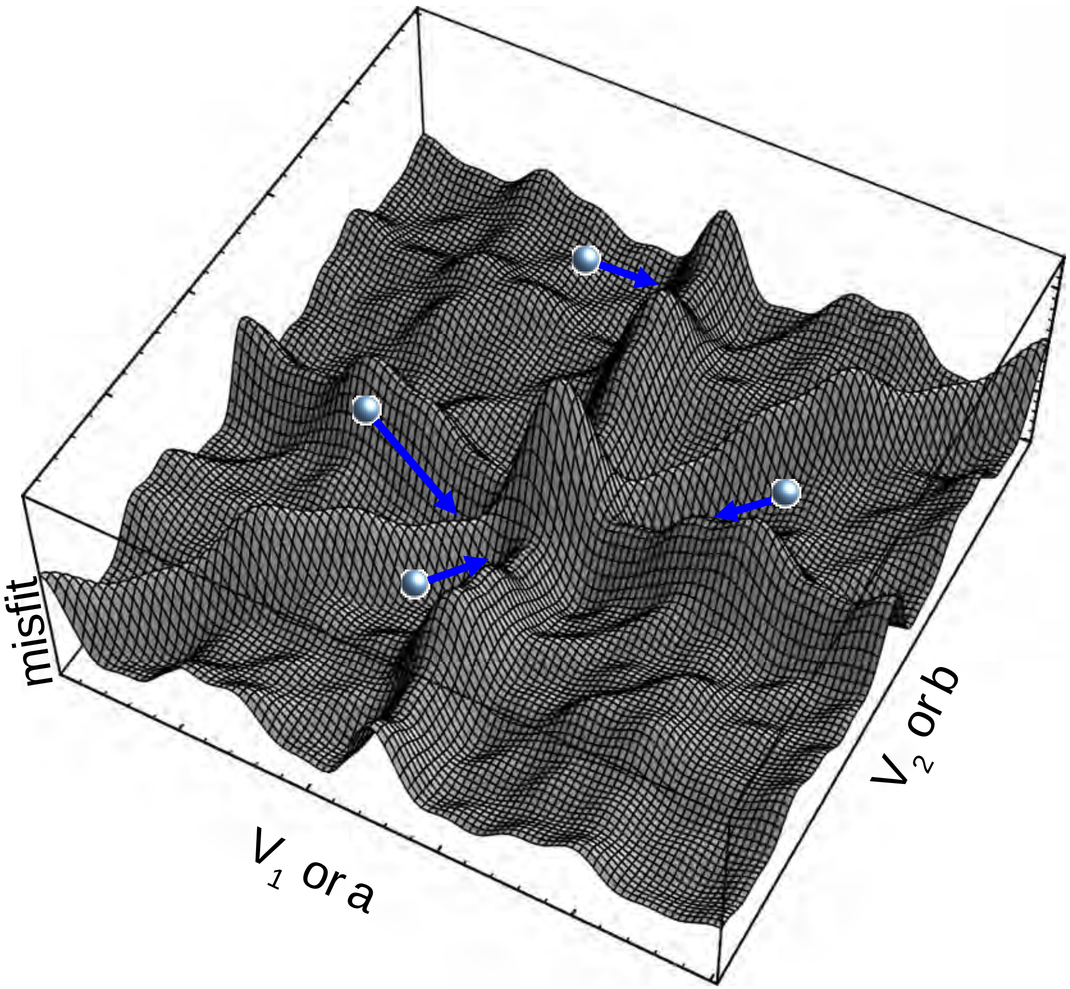




Modified from Sambridge & Mosegaard, 2002

Motivation: Why statistical approach?

but a real world's inversion problem looks rather like ...



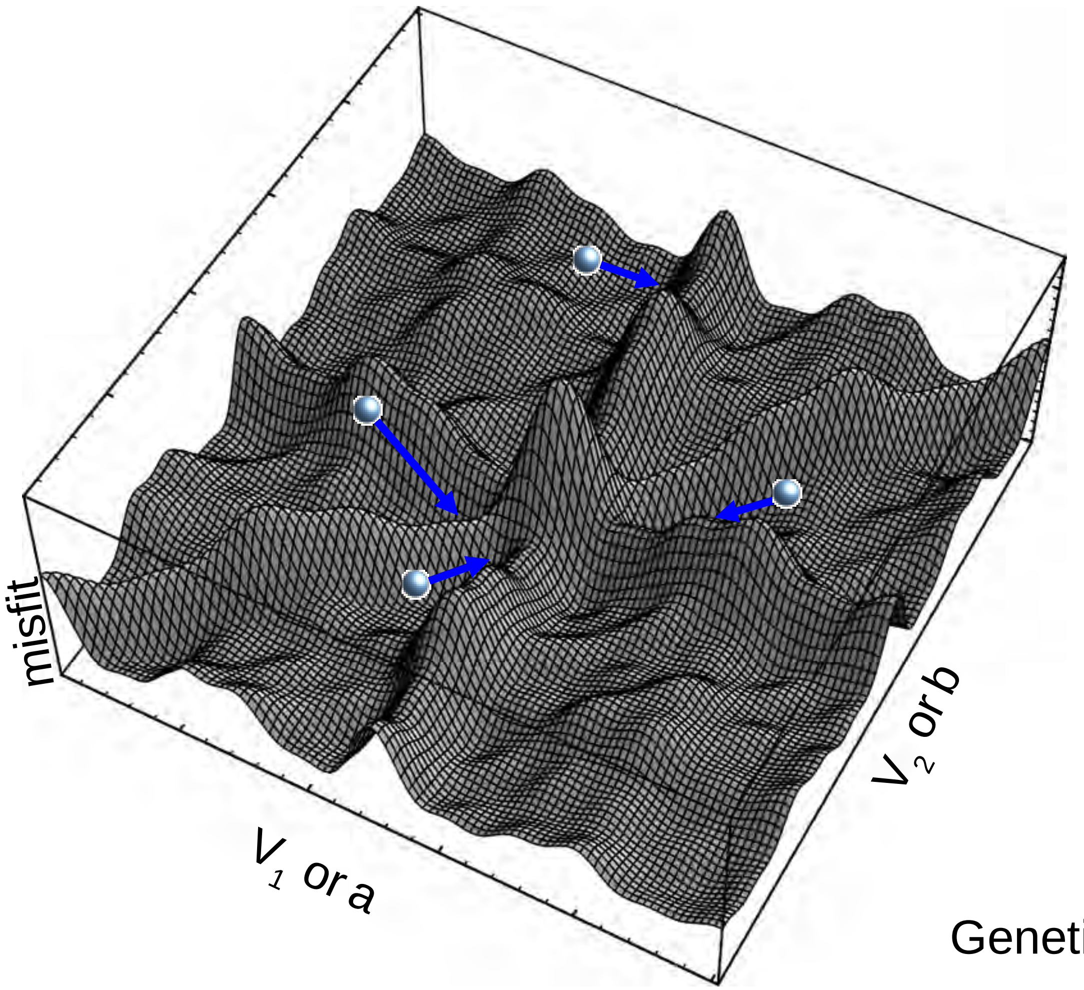
global/local minima ...

choice of starting model …

etc.

Motivation: Why statistical approach?

but a real world's inversion problem looks rather like ...



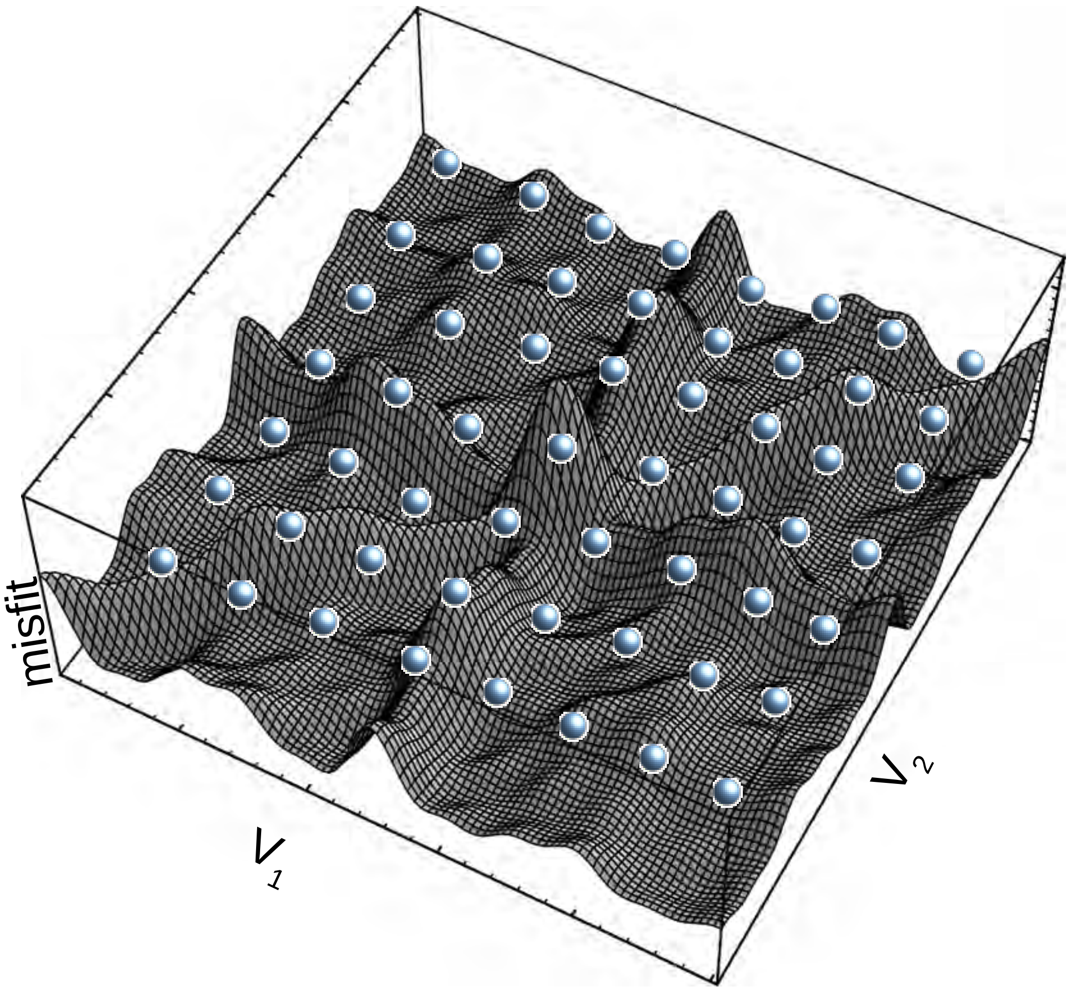
global/local minima ...

choice of starting model …

etc.

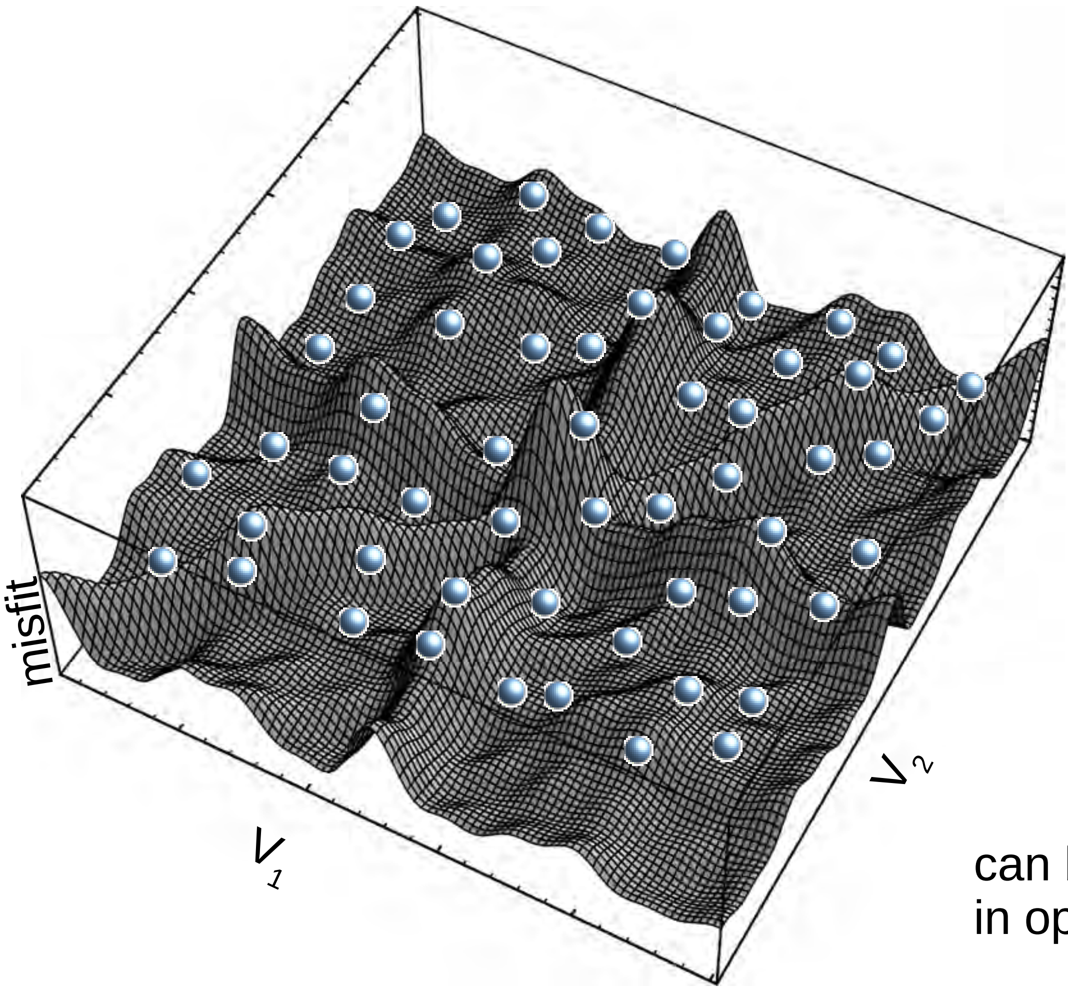
Genetic algorithms, simulated annealing, etc.

Search strategies



option: systematic grid-search

Search strategies



option: random search

→ Monte Carlo methods

can be used to search the global minimum,
in optimization framework
or
generate models for further statistical analysis

Likelihood function

Conventional inversion: search for a single, “best fitting” model

MC search: assigning a likelihood to **all** evaluated models

+ generating (large) ensembles of models

+ study of statistical properties of those ensembles

Misfit : $$\phi = \left\| \frac{g(m) - d_{obs}}{\sigma_d} \right\|^2$$

Likelihood : $$p(d_{obs} | m) \sim \exp(-\phi(m)/2) = \exp\left(-\frac{1}{2} \left\| \frac{g(m) - d_{obs}}{\sigma_d} \right\|^2 \right)$$

Note: even models with bad fits will have a (small) likelihood!

Bayesian approach

1701-1761

Source: wikipedia

Th. Bayes, 1763:
An essay towards solving a problem in the doctrine of chances

Bayes' rule relating conditional probabilities:

posterior = likelihood x prior / evidence

or

$$p(m|d_{obs}) \sim p(d_{obs}|m)\, p(m)$$

with

$p(m|d_{obs})$ posterior probability of model **m** given the data $\mathbf{d}_{obs}$

$p(d_{obs}|m)$ probability of observing data $\mathbf{d}_{obs}$ with model **m** (likelihood)

$p(m)$ prior probability density of **m** (what we know before..)

or

Rule: how to update our prior knowledge of model parameters by the data

Probabilistic framework

$$p(m|d) \sim p(d|m)\, p(m)$$

posterior PDF

likelihood
of model **m**

prior PDF: what we know about the
model before measuring

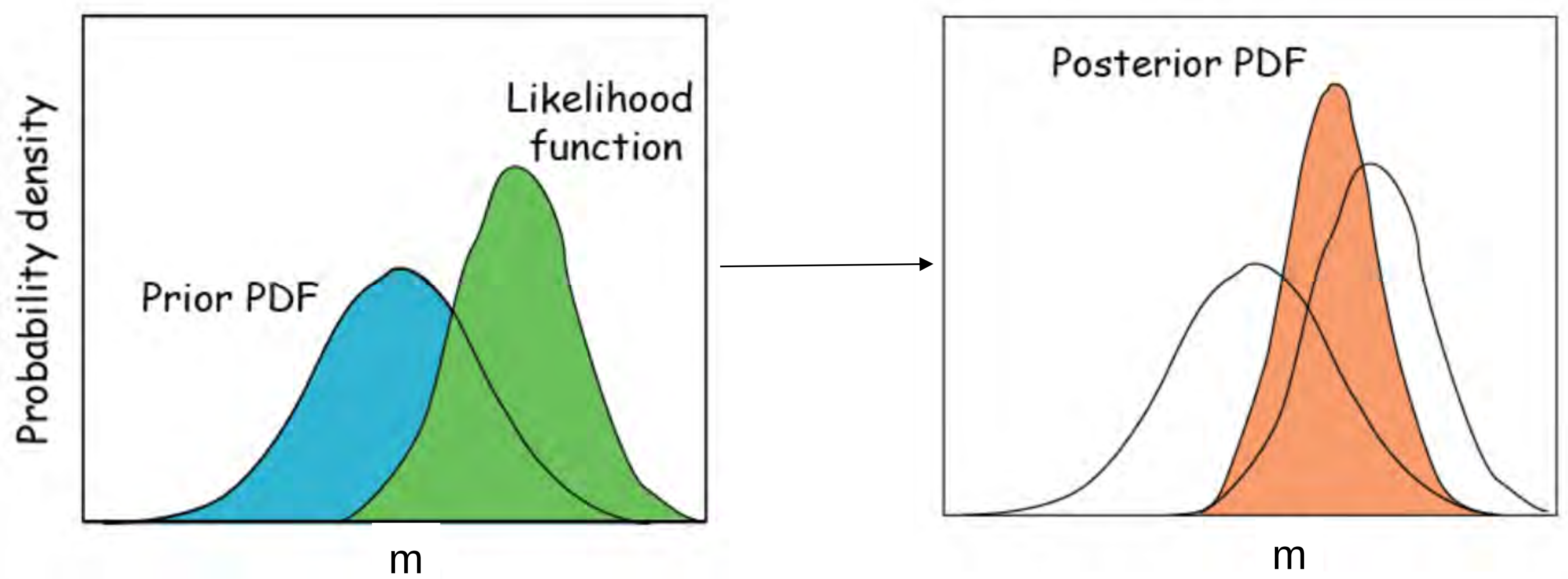
Probabilistic framework

$$p(m|d) \sim p(d|m)\,p(m)$$

posterior PDF

likelihood
of model **m**

prior PDF: what we know about the model before measuring



Modified from Sambridge & Mosegaard, 2002

The prior

What do we know about the model “before”, i.e. before having data?

i.e. How many cells to describe the model?

What model to start from?

How to sample different models in model space?

Which constraints on model velocities, i.e. range of values?

The prior

What do we know about the model “before”, i.e. before having data?

i.e. How many cells to describe the model?

What model to start from?

How to sample different models in model space?

Which constraints on model velocities, i.e. range of values?

avoid any dependence of the inversion on any/most (prior) assumption made

Concept of non-informative prior or minimum prior knowledge (Jeffrey’s prior):

chose completely random models with a wide range of velocities, varying number of cells, different noise levels, etc.

→ representative sampling of model space

The prior

What do we know about the model "before", i.e. before having data?

i.e. How many cells to describe the model?

What model to start from?

How to sample different models in model space?

Which constraints on model velocities, i.e. range of values?

avoid any dependence of the inversion on any/most (prior) assumption made

Concept of non-informative prior or minimum prior knowledge (Jeffrey's prior):

chose completely random models with a wide range of velocities, varying number of cells, different noise levels, etc.

→ representative sampling of model space

minimize the number of assumptions (i.e. V>0, model described by at least one cell, etc.)

wide search range for V from V_{min} to V_{max}, including "unrealistic" values

instead, let the data decide...

Ensemble inference

conventional inversion: search for the best fitting model (optimization)

with MC methods we generate a large set of models by thoroughly sampling the model space

What is the “final” model?

What about the best fitting (most likely) of all models?

Ensemble inference

conventional inversion: search for the best fitting model (optimization)

with MC methods we generate a large set of models by thoroughly sampling the model space

What is the “final” model?

What about the best fitting (most likely) of all models?

NOT! We can do more...

Lets take advantage of the Bayesian approach - ensemble inference rather than optimization:

if the posterior distributions of model parameters (velocities) is +- Gaussian

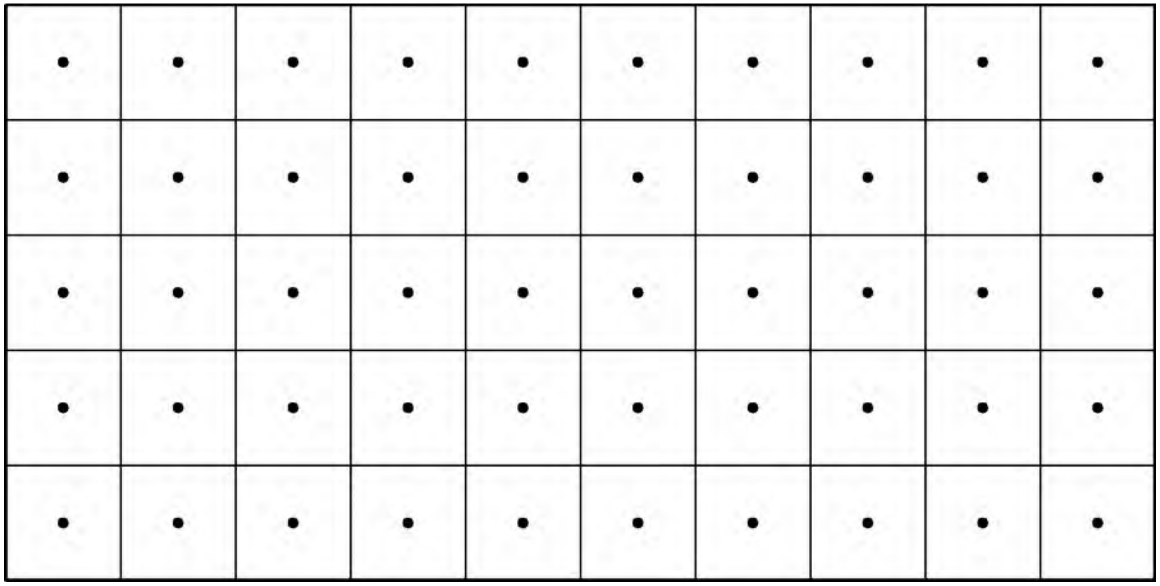
take averages of the velocity at point (x,y)
→ **reference model**

calculate standard deviations of the velocity at point (x,y)
→ **uncertainty of velocity** determination (“error bar”, resolution)

other approaches: median, confidence intervals, “bounds”, etc.

Model parametrization

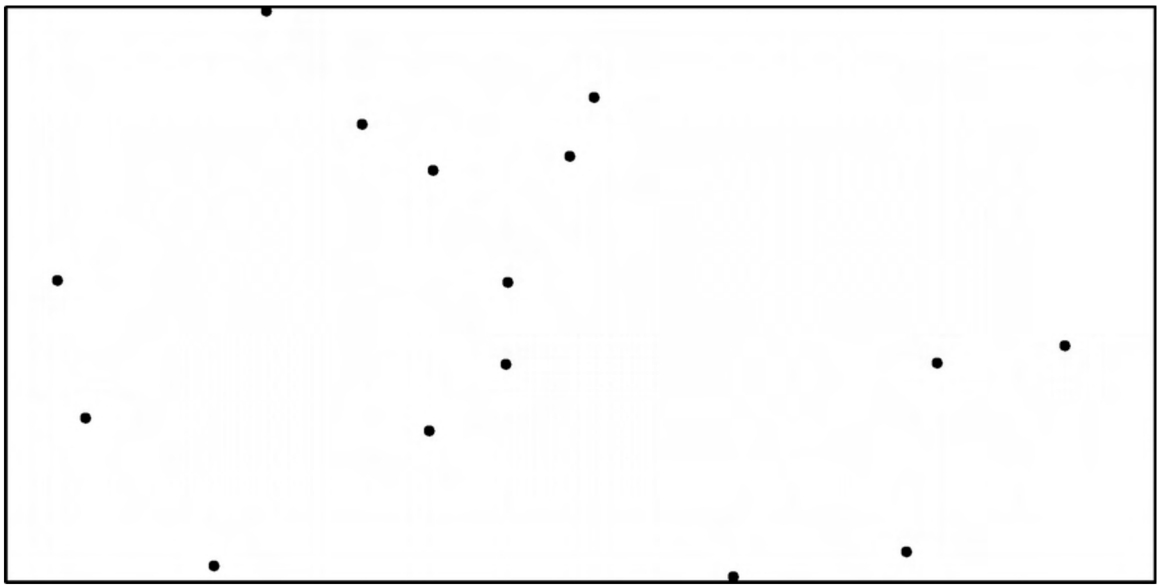
- model description by regular grid



Model parametrization

- model description by regular grid

- model description by irregular mesh

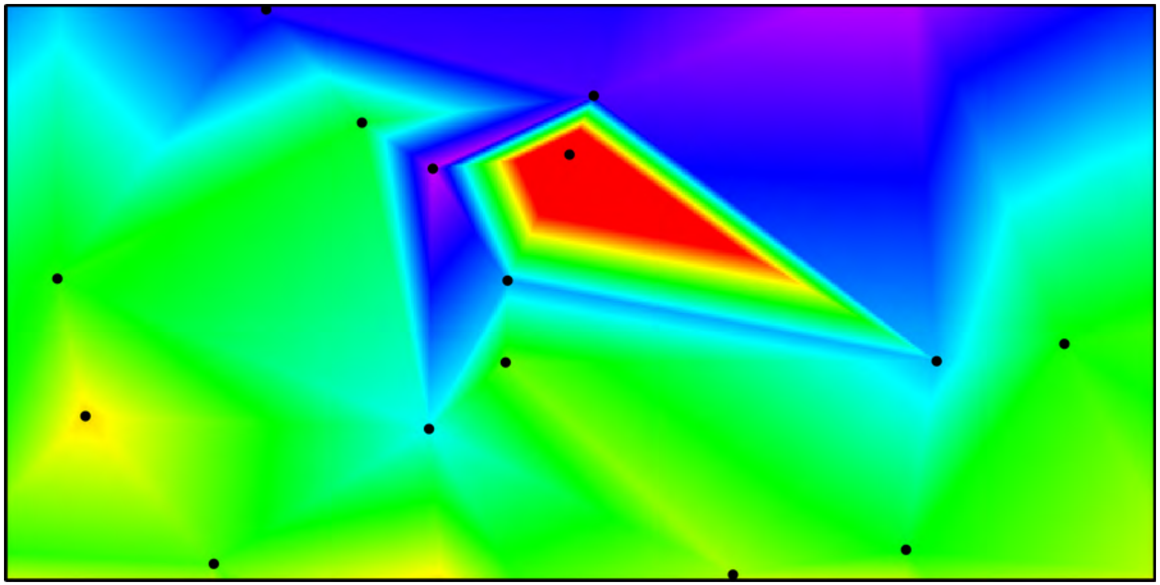


Model parametrization

- model description by regular grid

- model description by irregular mesh

interpolation between cell nuclei (linear, b-splines, etc.)



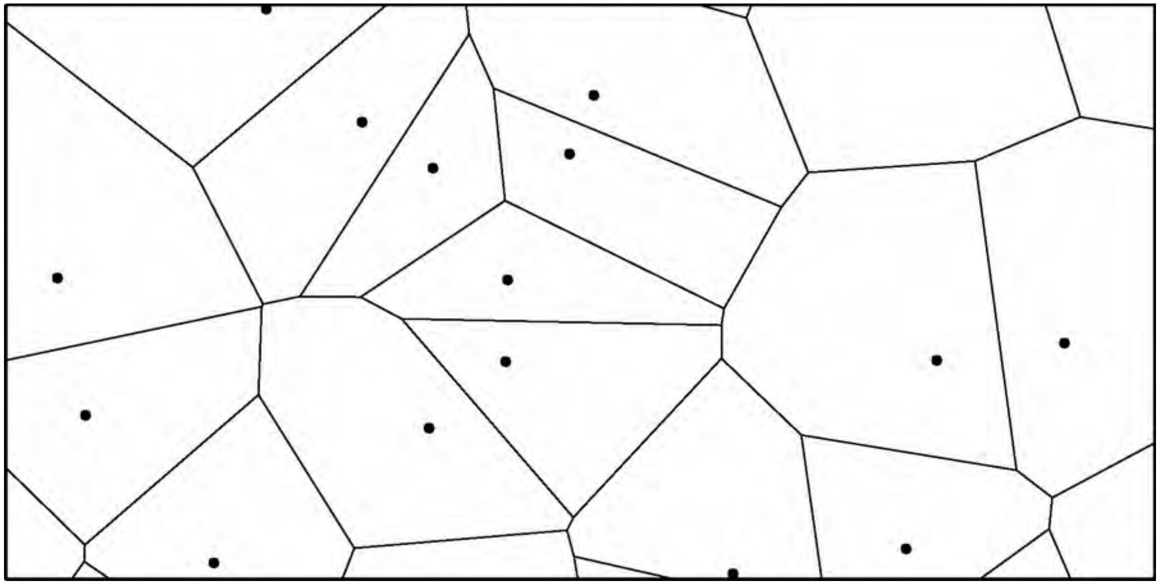
Model parametrization

- model description by regular grid

- model description by irregular mesh

Voronoi cells (tesselation)

G. Voronoi, 1868-1908
Source: wikipedia



Model parametrization

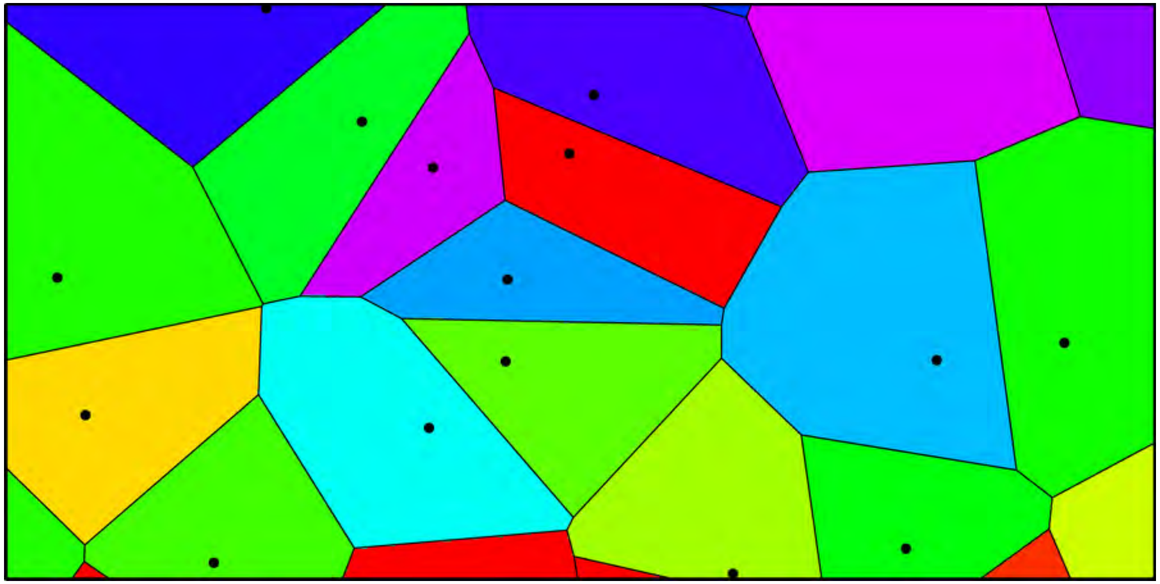
- model description by regular grid

- model description by irregular mesh

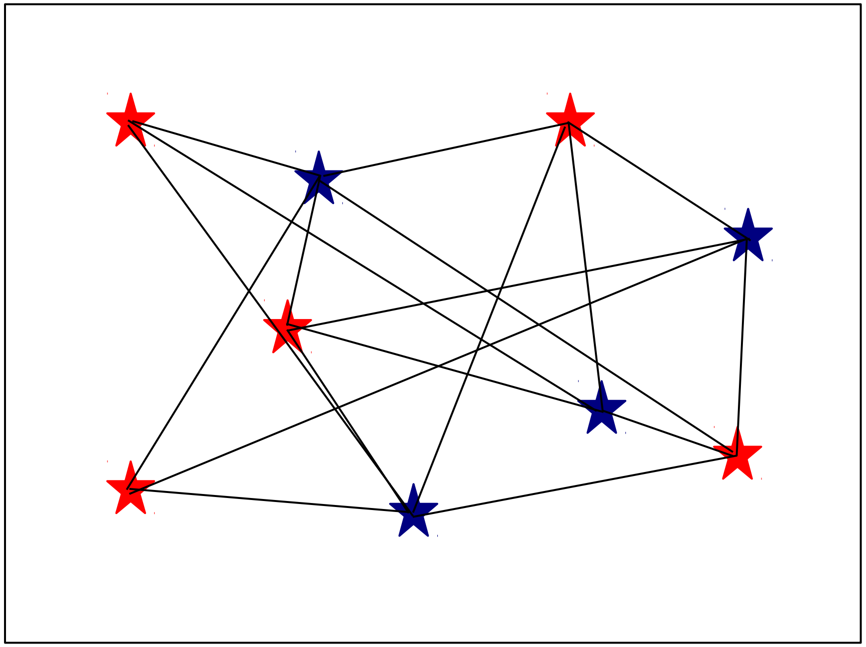
constant values in cells: Voronoi cells

model **m** (x_1, y_1, s_1; x_2, y_2, s_2; …. x_N, y_N, s_N) with $s_i = 1/V_i$

G. Voronoi, 1868-1908
Source: wikipedia



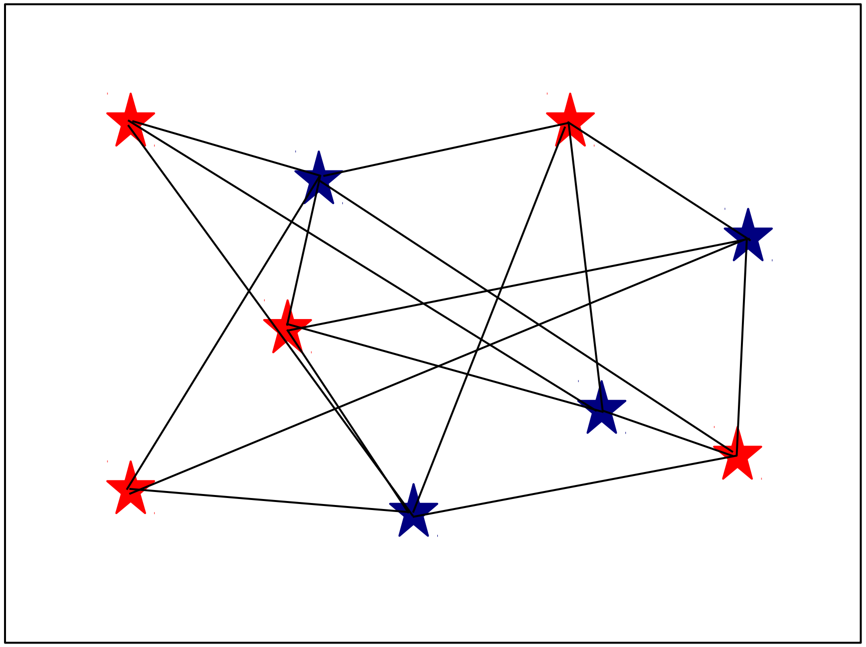
Solving the forward problem



Task: calculate travel times between all sources and receivers for a given velocity model

conventionally done by ray tracing, but might be slow...

Solving the forward problem



Task: calculate travel times between all sources and receivers for a given velocity model

conventionally done by ray tracing, but might be slow...

Instead we can solve the eikonal equation (Podvin & Lecomte, 1991)

$$(\nabla t(x,y))^2 = s^2(x,y)$$ with t – travel time field, s – slowness (1/v) field

this can be done very efficiently by finite-difference methods

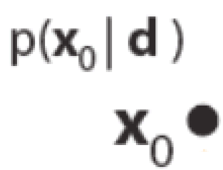
→ very fast!

Markov chain Monte Carlo

Markov chain = sequence of models following specific rules

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

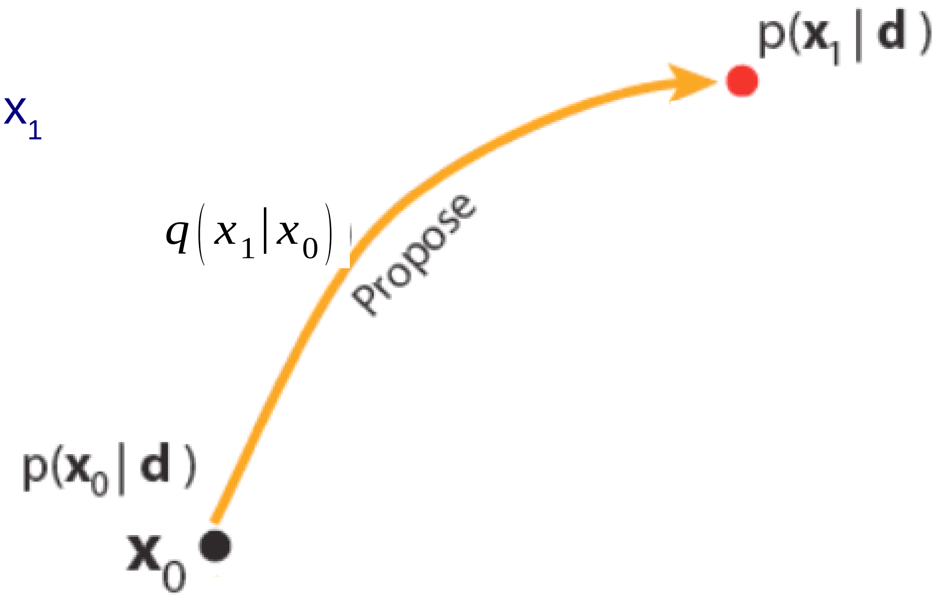
start with x_0



Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

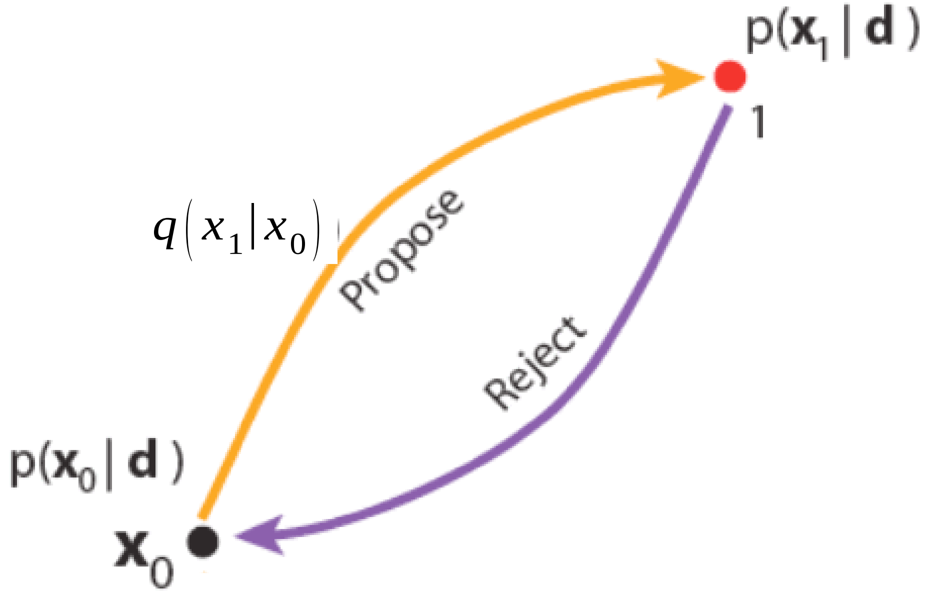
propose x_1



Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

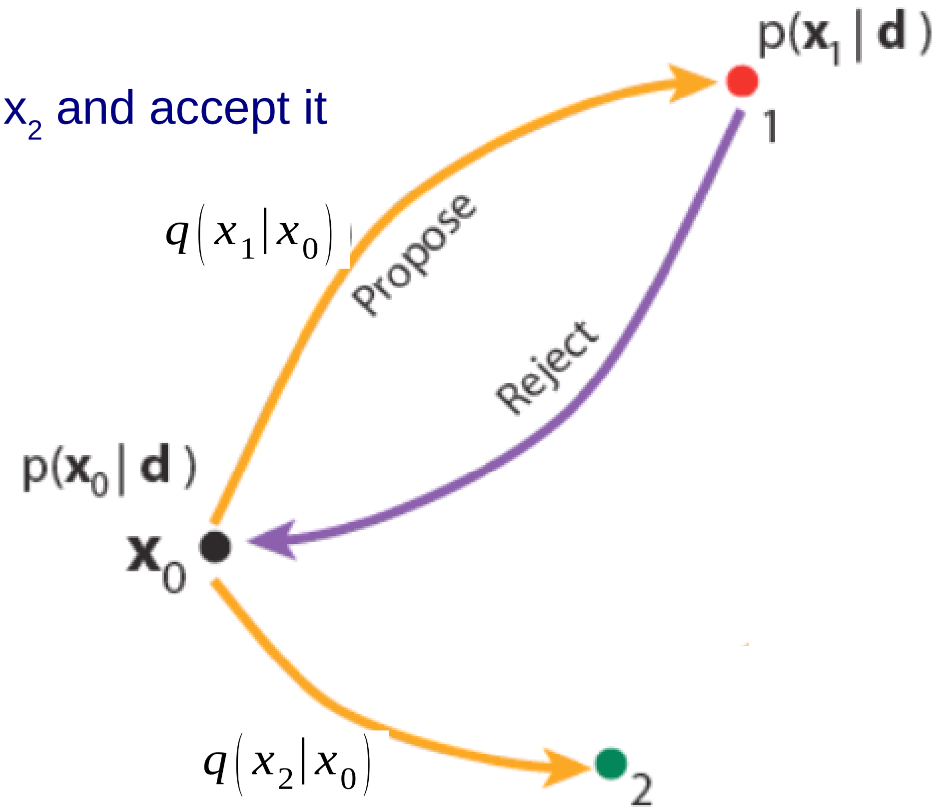
reject x_1



Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

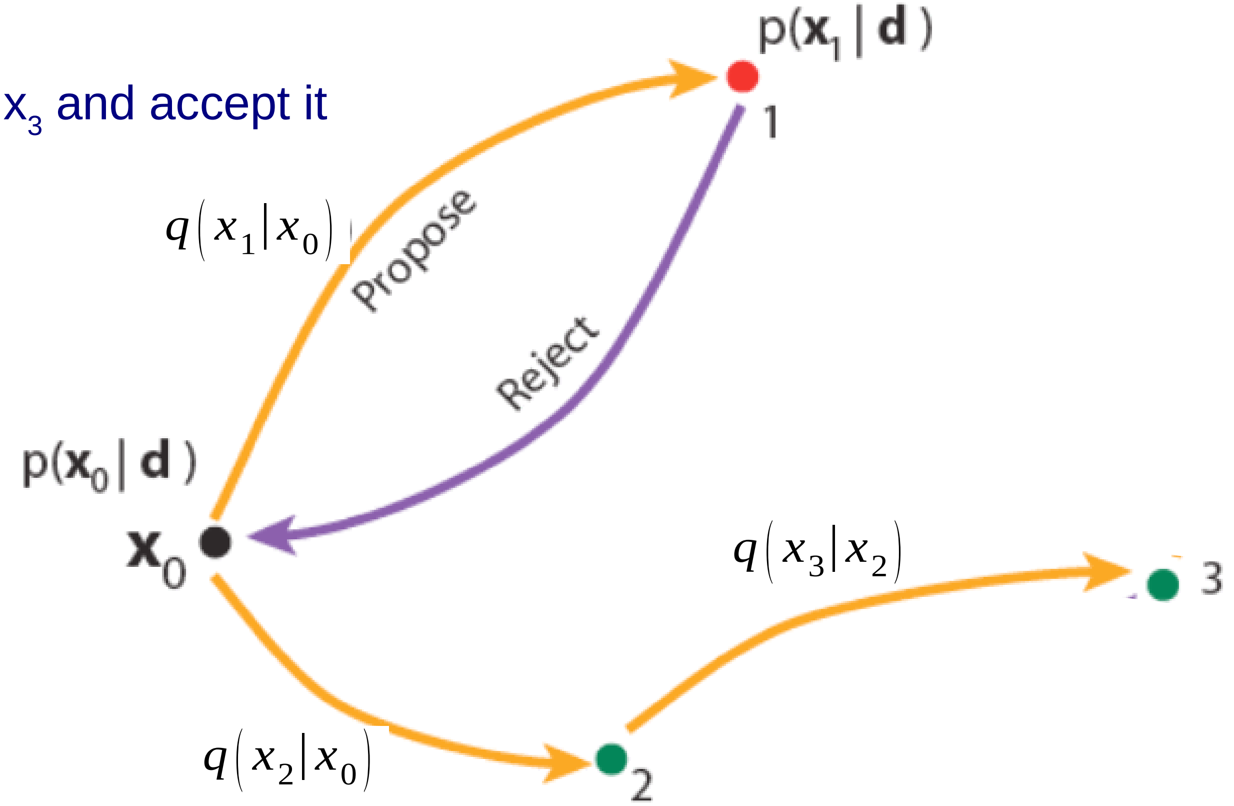
propose x_2 and accept it



Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

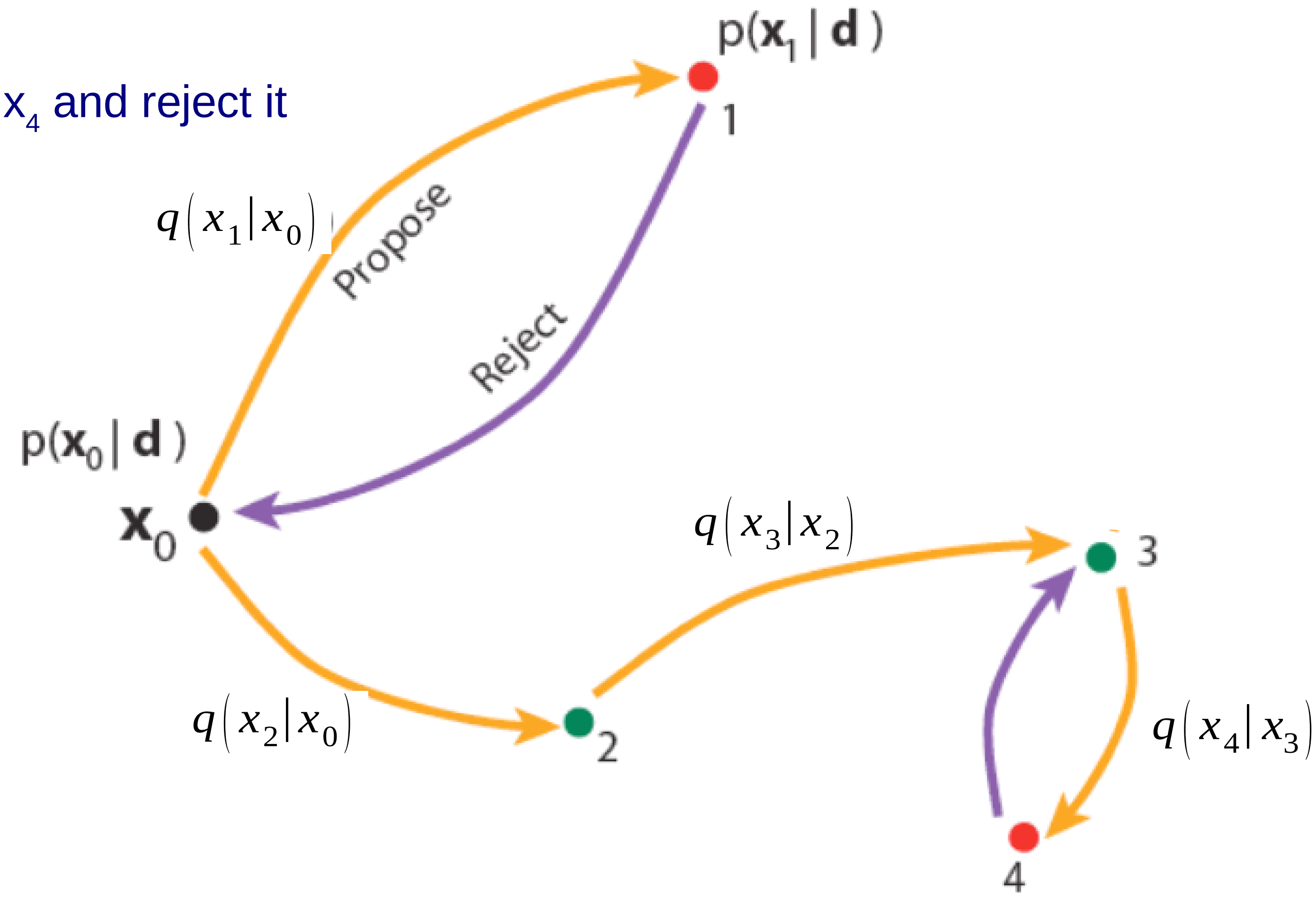
propose x_3 and accept it



Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

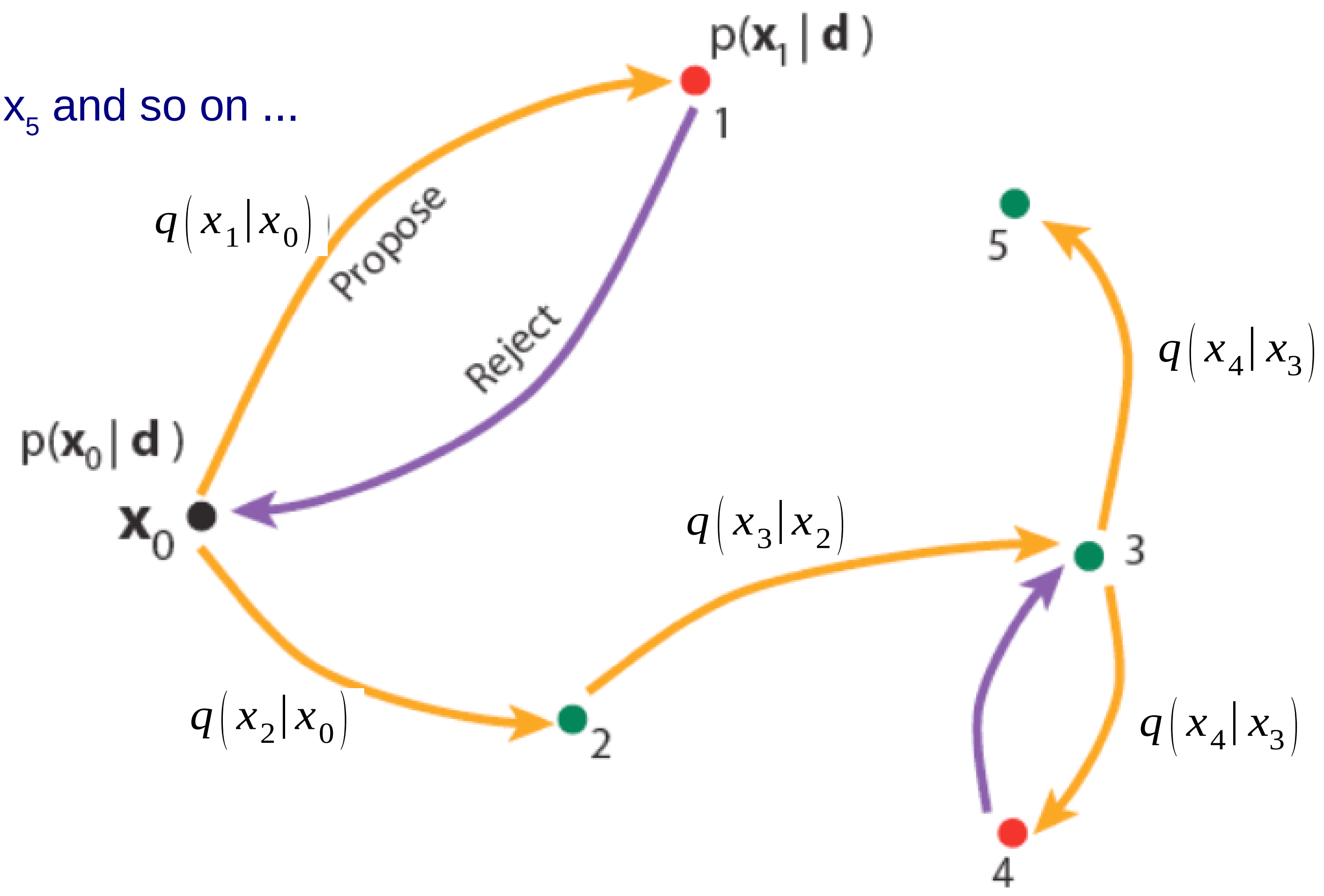
propose x_4 and reject it



Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

propose x_5 and so on ...



Modified from Sambridge 2012

Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

model PDF $$p(m_0|d) \sim p(d|m_0)\, p(m_0)$$

transition PDF from model $\mathbf{m_0}$ to $\mathbf{m_1}$ $$q(m_1|m_0)$$

MH acceptance probability $$\alpha(m_1, m_0) = min\left[1, \left(\frac{p(m_1|d)}{p(m_0|d)}\right)\right]$$

Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

model PDF $$p(m_0|d) \sim p(d|m_0)\, p(m_0)$$

transition PDF from model $\mathbf{m_0}$ to $\mathbf{m_1}$ $$q(m_1|m_0)$$

MH acceptance probability $$\alpha(m_1, m_0) = min\left[1, \left(\frac{p(m_1|d)}{p(m_0|d)}\right)\right]$$

Note: even proposed models with poorer data fits might be accepted!

Markov chain Monte Carlo

Evolution of a Markov chain by
Metropolis-Hastings Algorithm (Metropolis et al., 1953)

model PDF $$p(m_0|d) \sim p(d|m_0)\,p(m_0)$$

transition PDF from model $\mathbf{m_0}$ to $\mathbf{m_1}$ $$q(m_1|m_0)$$

MH acceptance probability $$\alpha(m_1, m_0) = min\left[1, \left(\frac{p(m_1|d)}{p(m_0|d)}\right)\right]$$

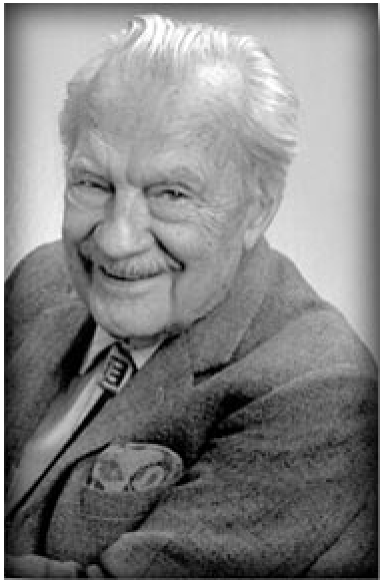
Note: even proposed models with poorer data fits might be accepted!

for exact expressions see i.e. Bodin & Sambridge, 2009

Metropolis-Hastings algorithm

N. Metropolis developed the Monte Carlo methods in 1950s (+ S. Ulam, J. von Neumann, E. Teller, etc.)

Hastings generalized Metropolis' approach in 1970



N. Metropolis, 1915-1999

Source: wikipedia

Metropolis-Hastings algorithm

Implementation of Metropolis-Hastings Algorithm

1. randomly pick one cell in model $\mathbf{m}_0$

Metropolis-Hastings algorithm

Implementation of Metropolis-Hastings Algorithm

1. randomly pick one cell in model $\mathbf{m}_0$

   1a. randomly change slowness in this cell to create model $\mathbf{m}_1$

Metropolis-Hastings algorithm

Implementation of Metropolis-Hastings Algorithm

1. randomly pick one cell in model $\mathbf{m}_0$

1a. randomly change slowness in this cell to create model $\mathbf{m}_1$

or

1b. randomly change position of this cell to create model $\mathbf{m}_1$

Metropolis-Hastings algorithm

Implementation of Metropolis-Hastings Algorithm

1. randomly pick one cell in model $\mathbf{m}_0$

   1a. randomly change slowness in this cell to create model $\mathbf{m}_1$

   or

   1b. randomly change position of this cell to create model $\mathbf{m}_1$

2. solve forward problem, i.e. calculate misfit and likelihood of model $\mathbf{m}_1$

Metropolis-Hastings algorithm

Implementation of Metropolis-Hastings Algorithm

1. randomly pick one cell in model $\mathbf{m}_0$

   1a. randomly change slowness in this cell to create model $\mathbf{m}_1$

   or

   1b. randomly change position of this cell to create model $\mathbf{m}_1$

2. solve forward problem, i.e. calculate misfit and likelihood of model $\mathbf{m}_1$

3. calculate acceptance probability of model $\mathbf{m}_1$

Metropolis-Hastings algorithm

Implementation of Metropolis-Hastings Algorithm

1. randomly pick one cell in model $\mathbf{m}_0$

   1a. randomly change slowness in this cell to create model $\mathbf{m}_1$

   or

   1b. randomly change position of this cell to create model $\mathbf{m}_1$

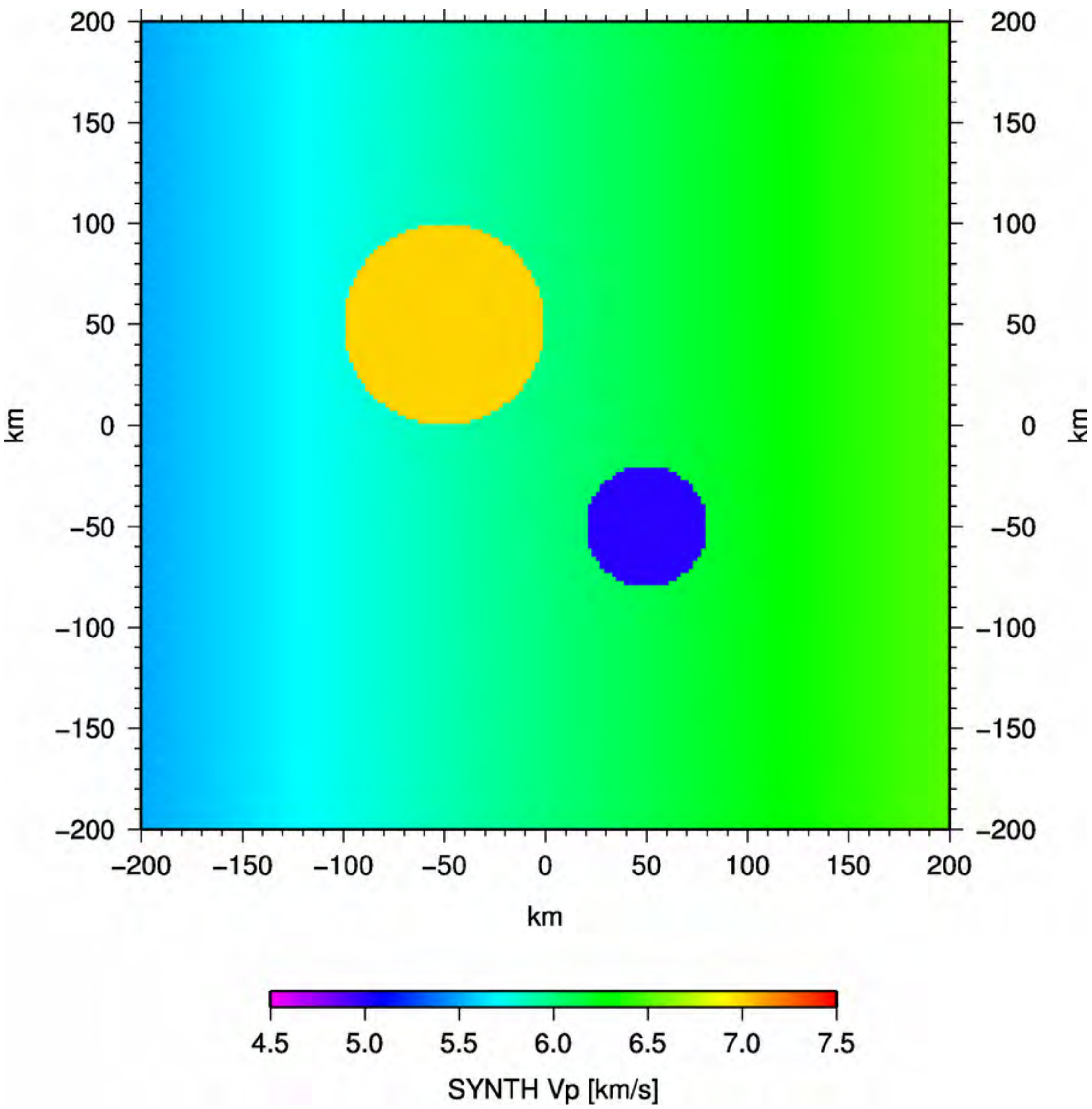
2. solve forward problem, i.e. calculate misfit and likelihood of model $\mathbf{m}_1$

3. calculate acceptance probability of model $\mathbf{m}_1$

4. accept or reject model $\mathbf{m}_{1,}$ then goto 1

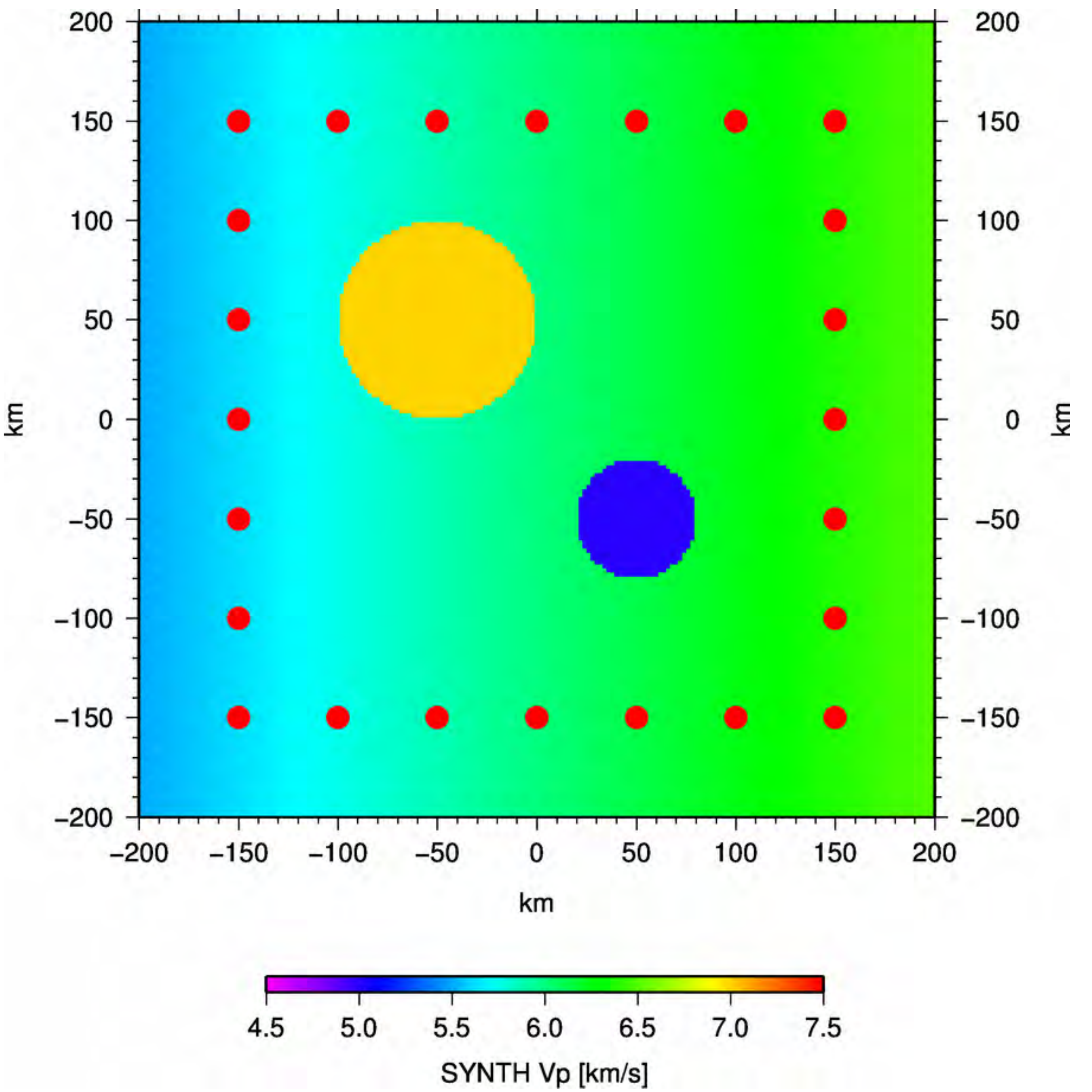
Synthetic example

2D velocity model



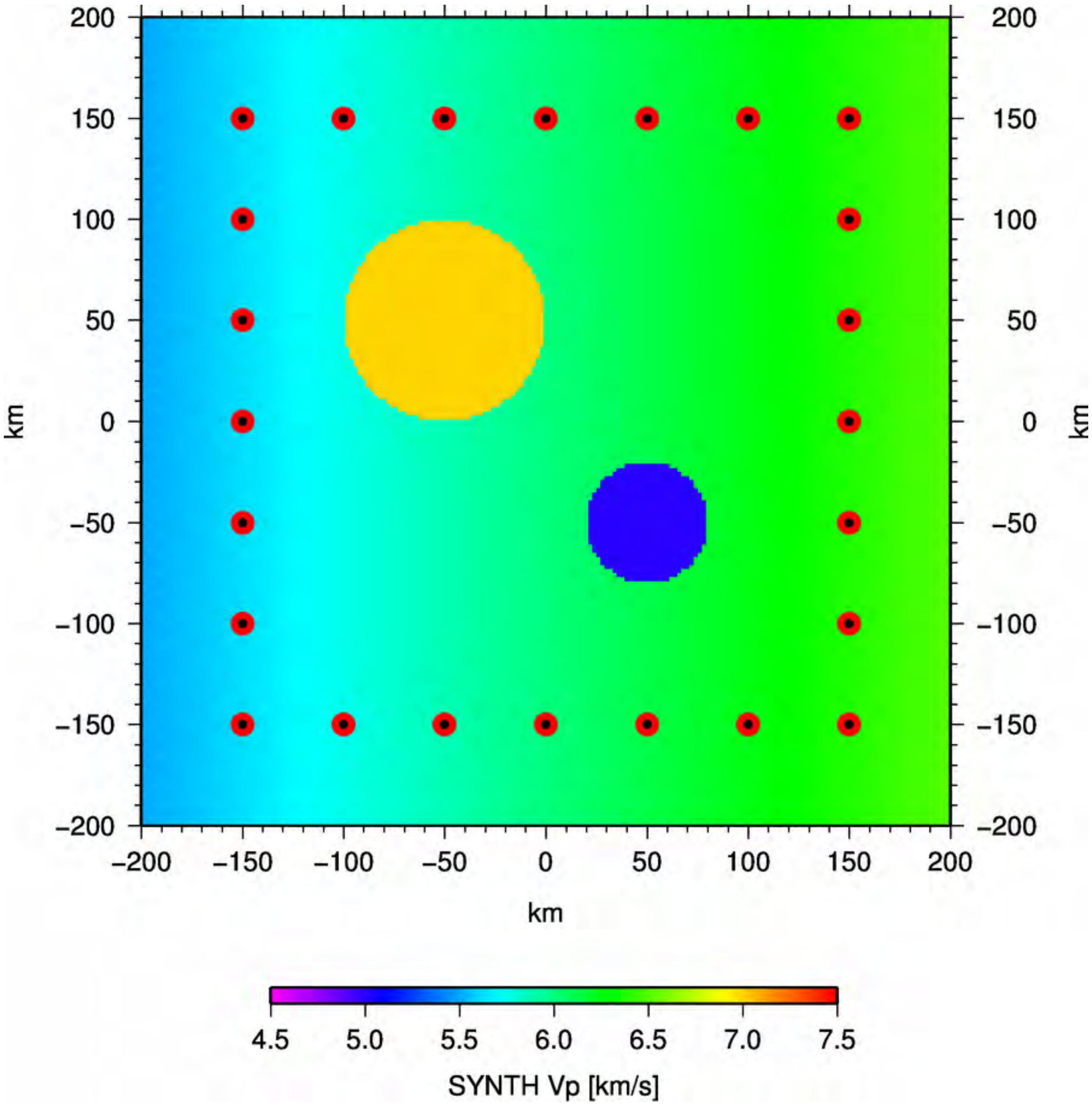
Synthetic example

2D velocity model, 24 sources



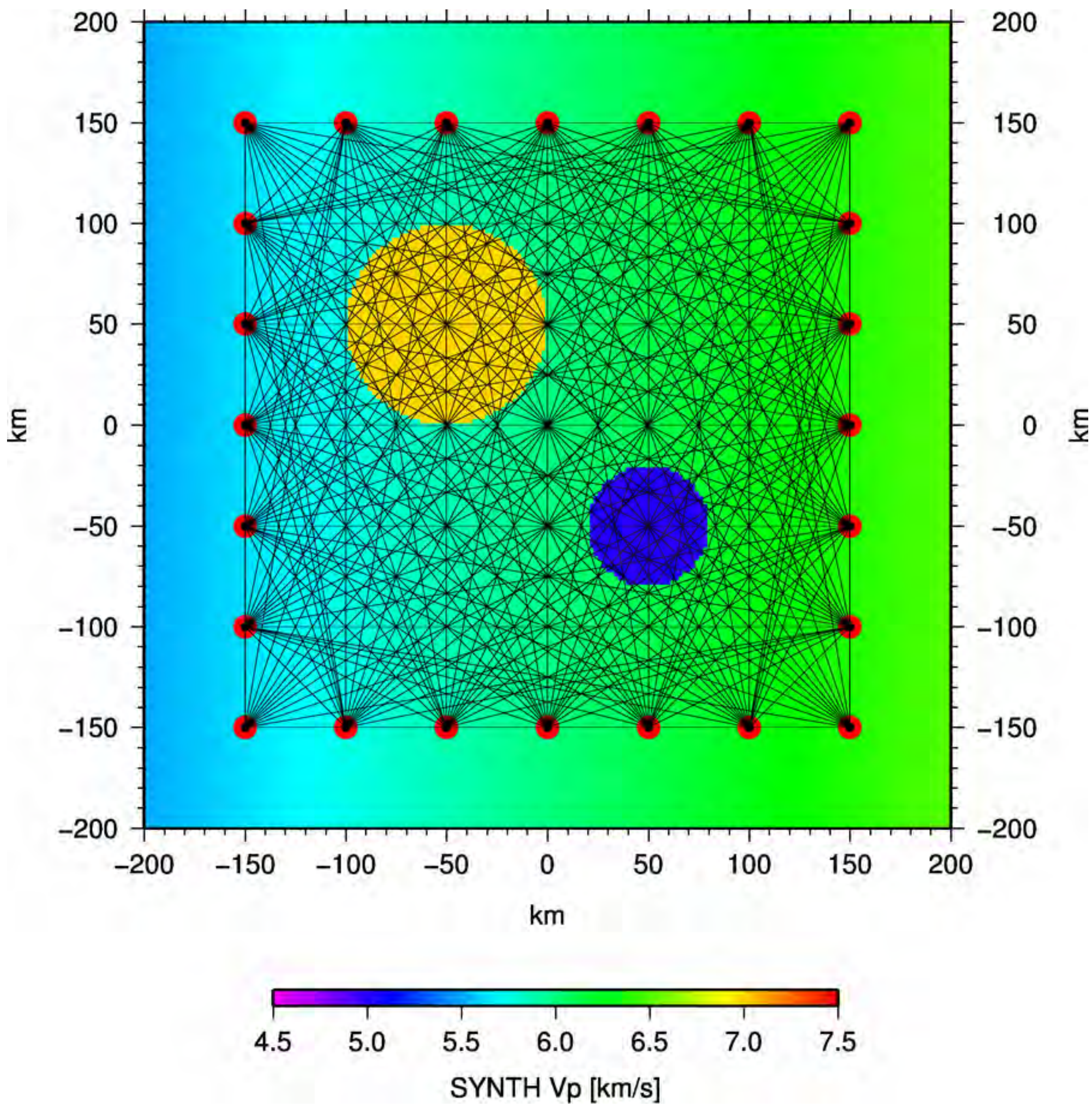
Synthetic example

2D velocity model, 24 sources, 24 receivers



Synthetic example

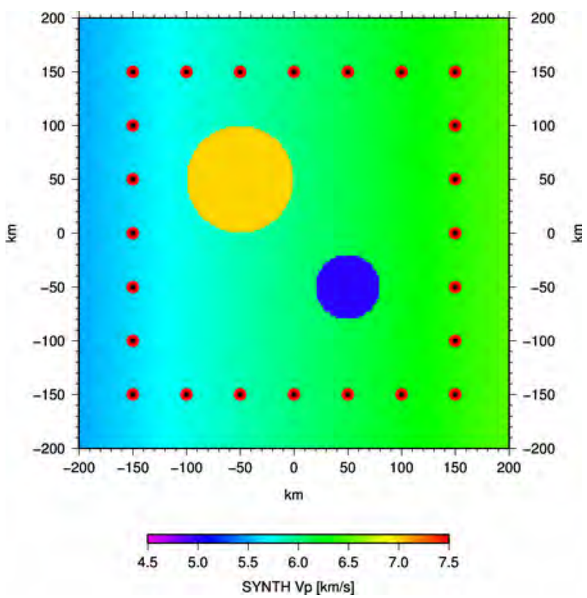
2D velocity model, 24 sources, 24 receivers, 552 travel time picks, 50ms noise added



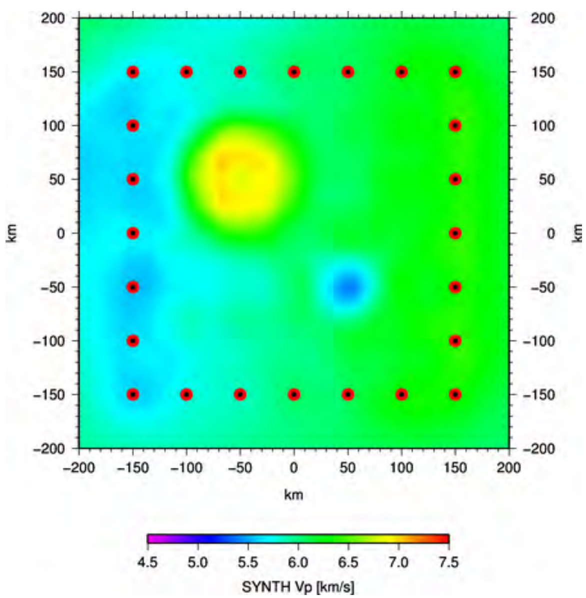
Synthetic example

Result of a conventional inversion,
with regular grid, (FAST, Zelt & Barton, 1998):

Synthetic model



Recovered model

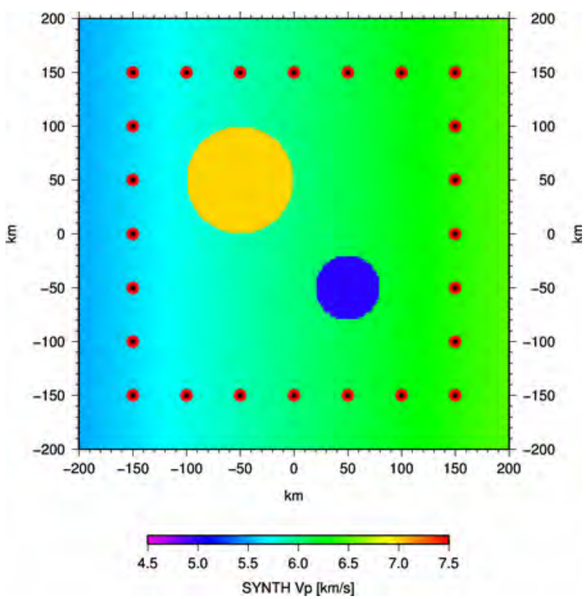


but...

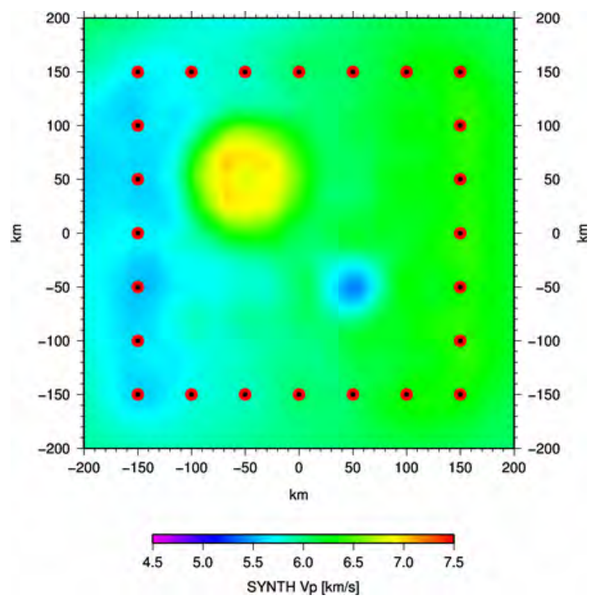
Synthetic example

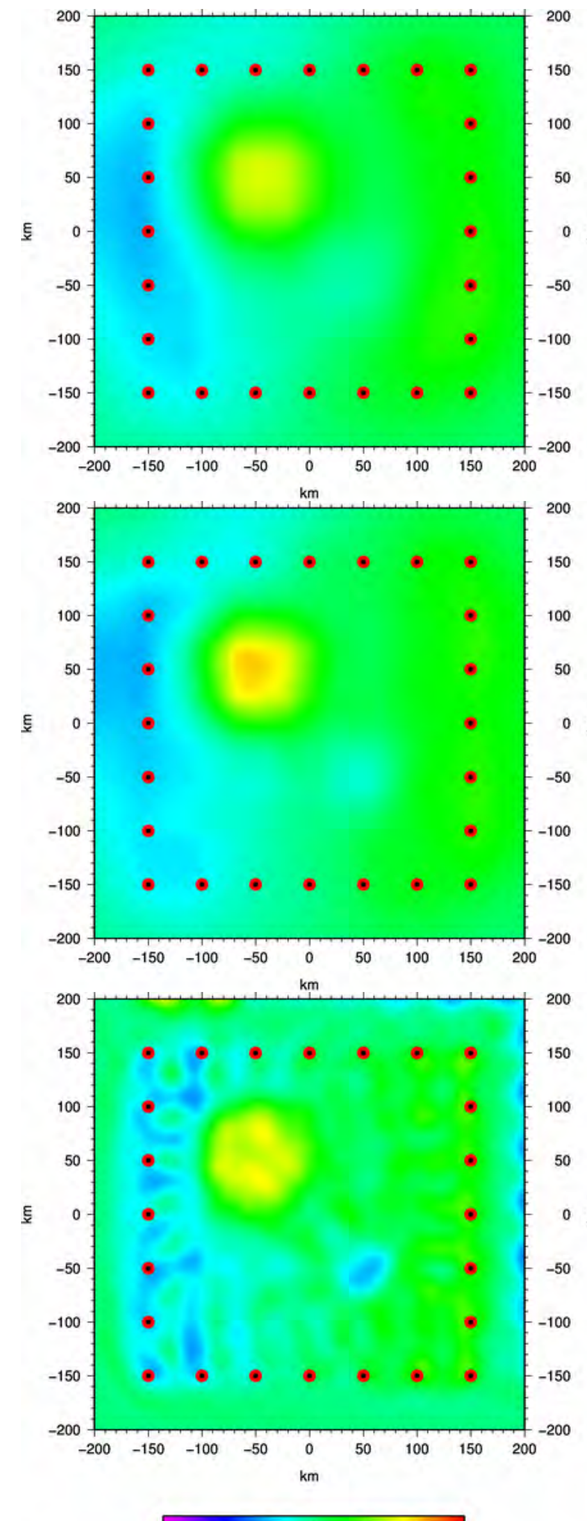
Result of a conventional inversion,
with regular grid, (FAST, Zelt & Barton, 1998):

Synthetic model



Recovered model



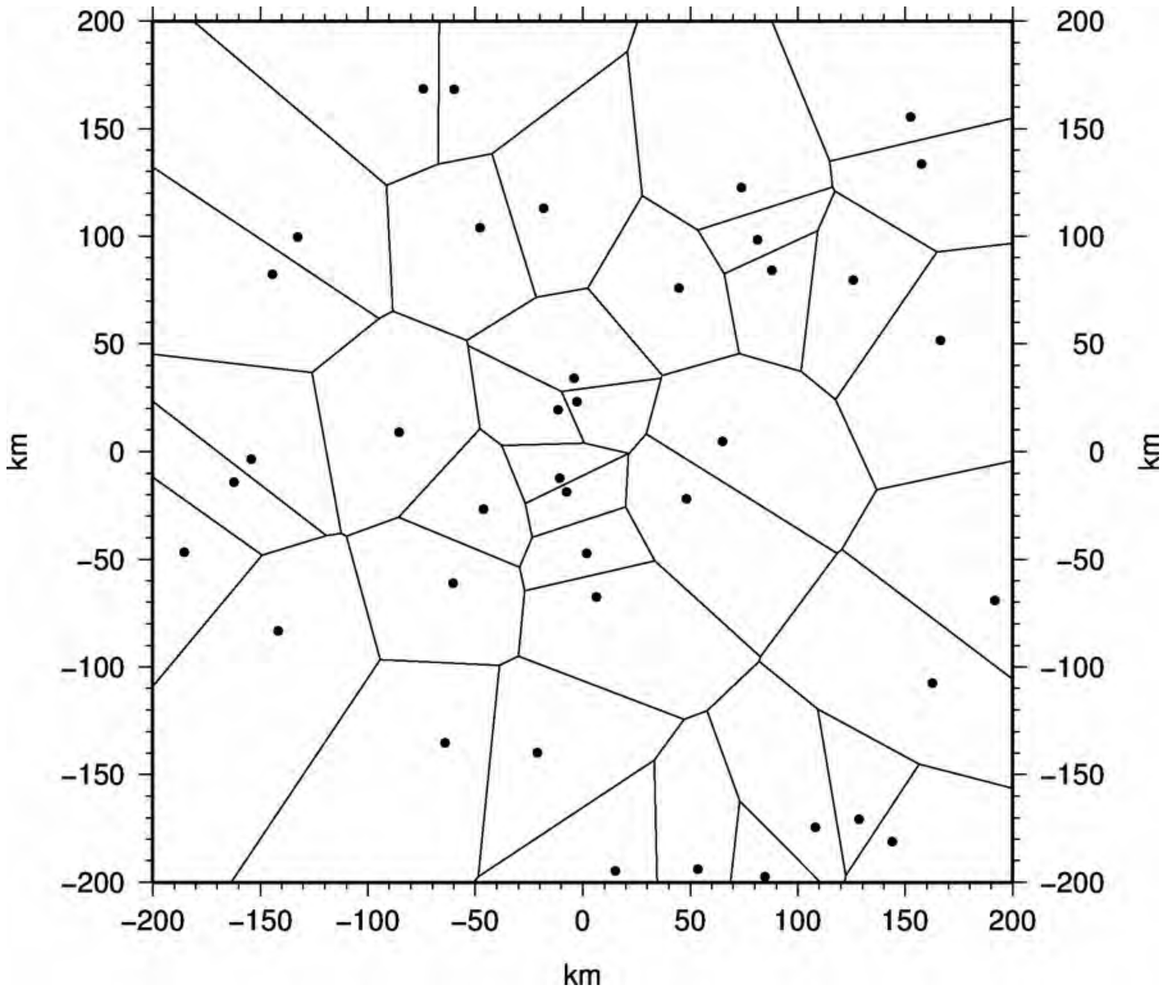


changing (prior) inversion parameters (smoothness, damping, grid size, etc.) changes the inversion result! All of them have the same misfit!

What is “the final” result?

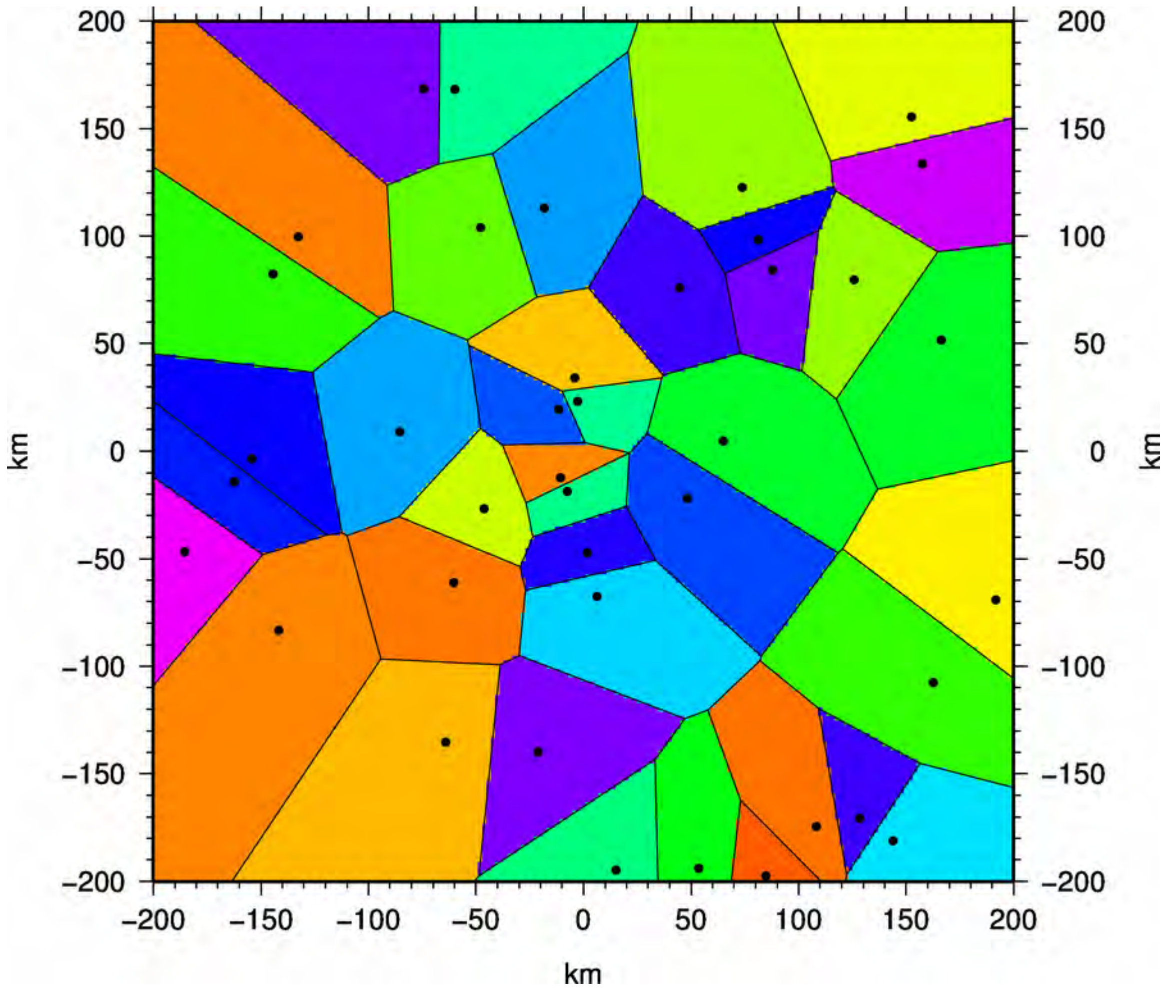
Synthetic example MCMC

40 Voronoi cells, randomly distributed



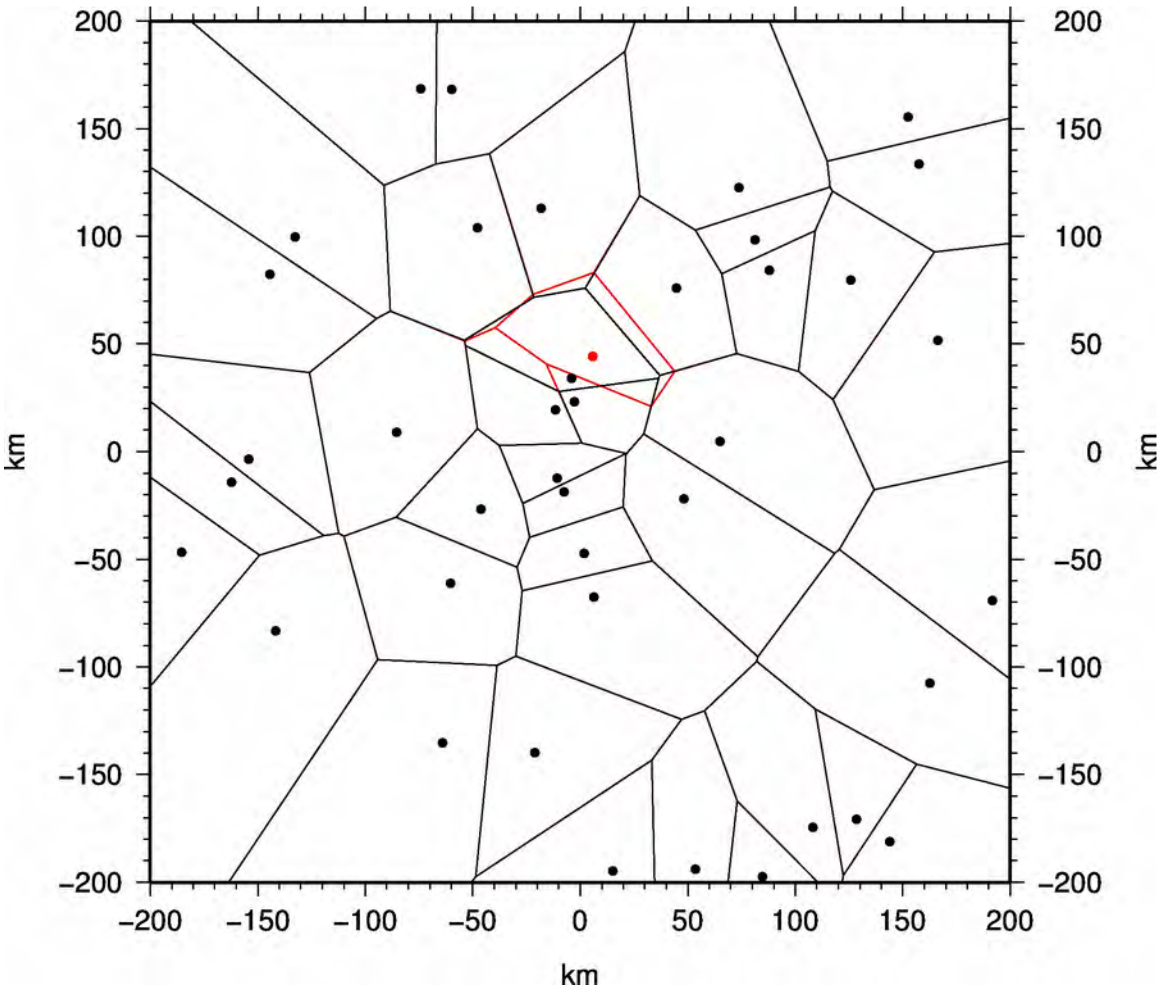
Synthetic example MCMC

40 Voronoi cells, random velocity assigned



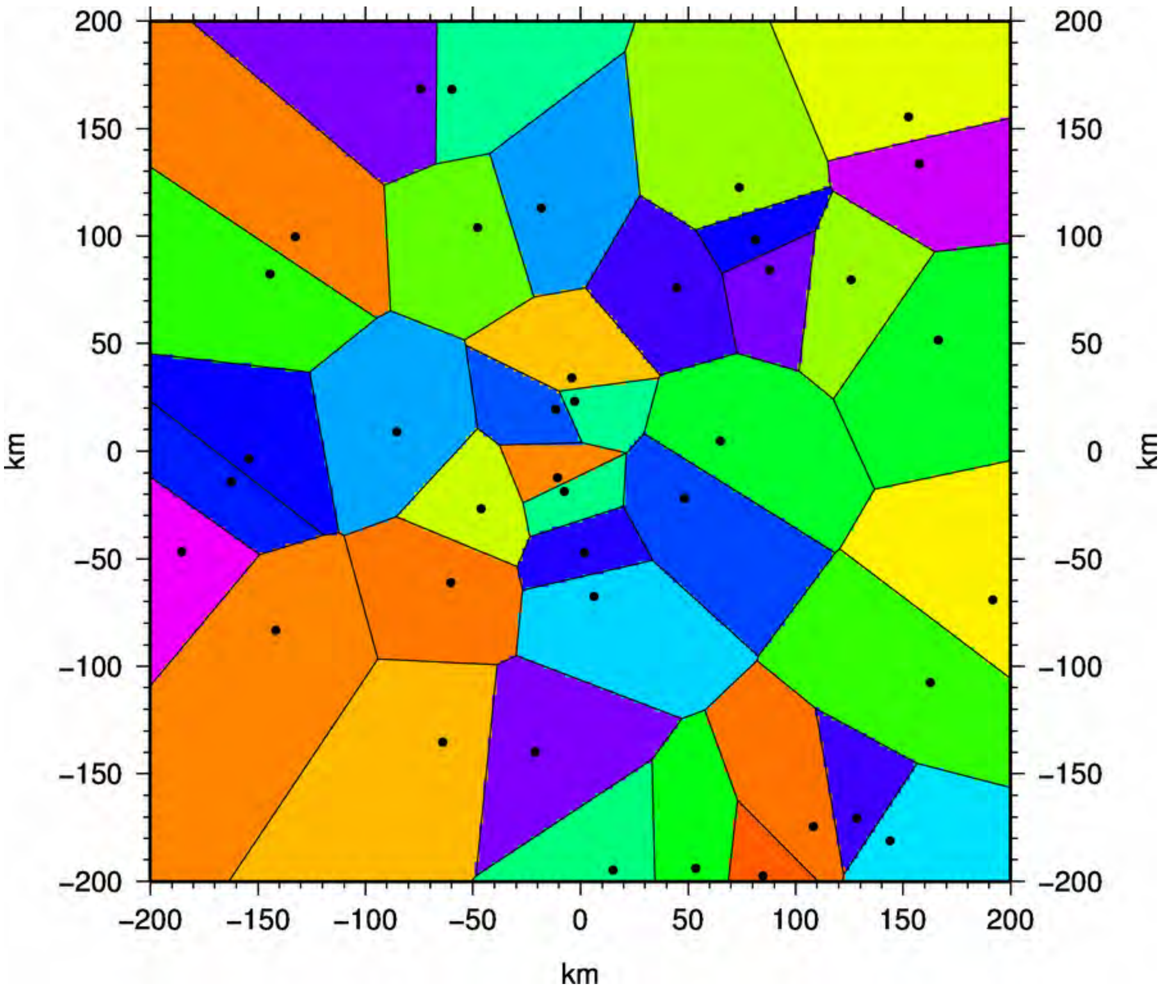
Synthetic example MCMC

Markov chains by perturbing cell nuclei position



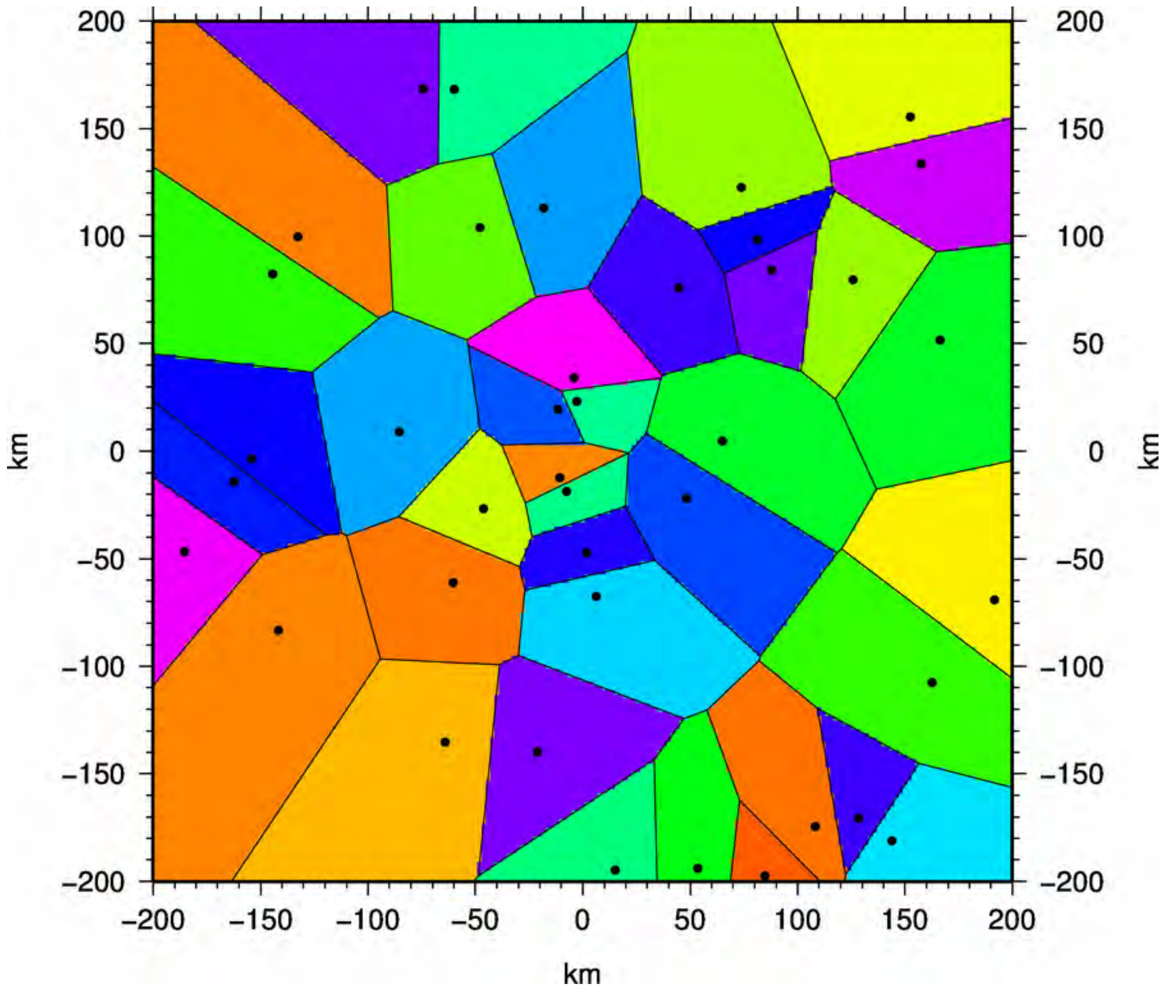
Synthetic example MCMC

Markov chains by perturbing cell velocities

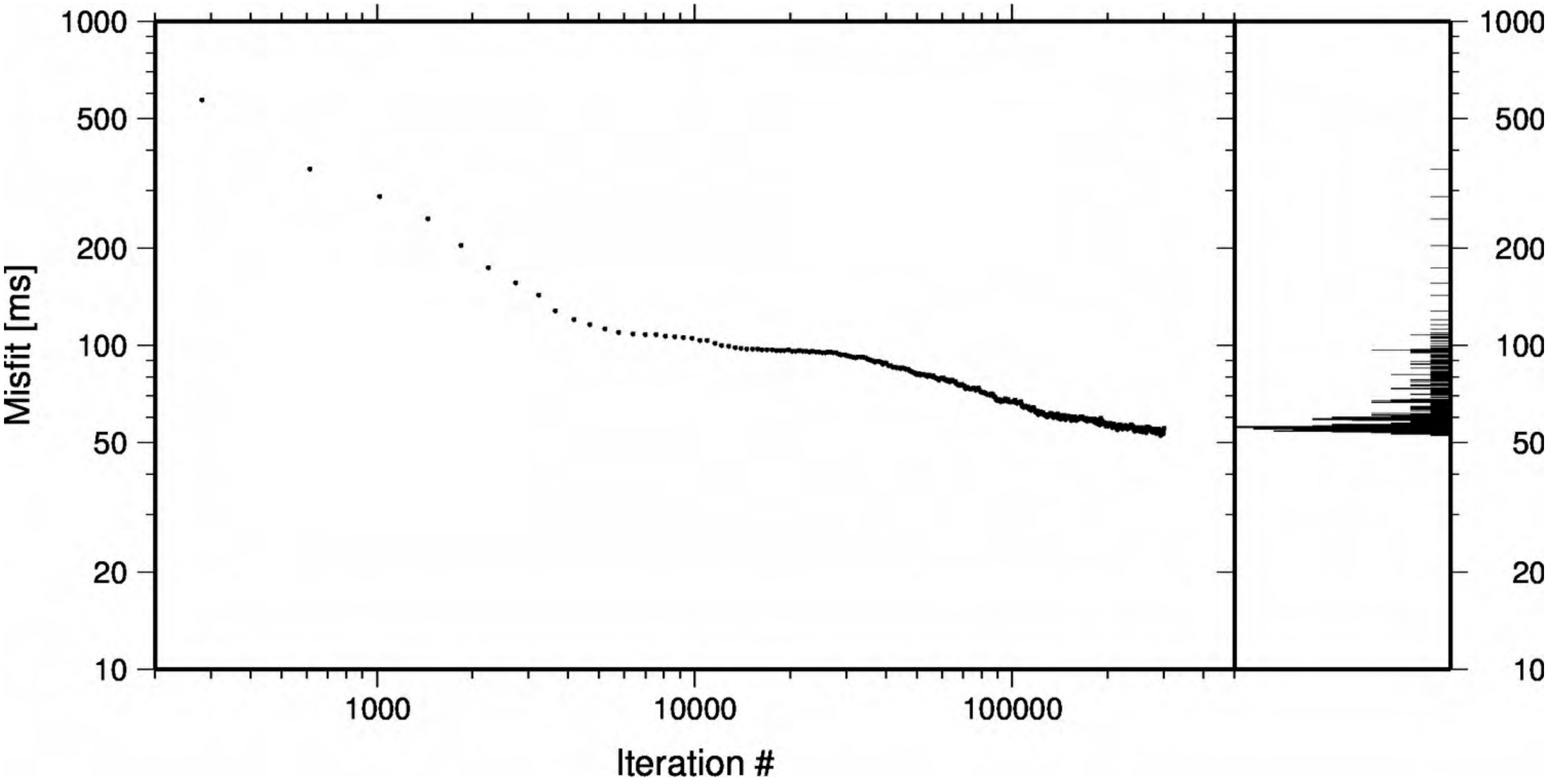


Synthetic example MCMC

Markov chains by perturbing cell velocities

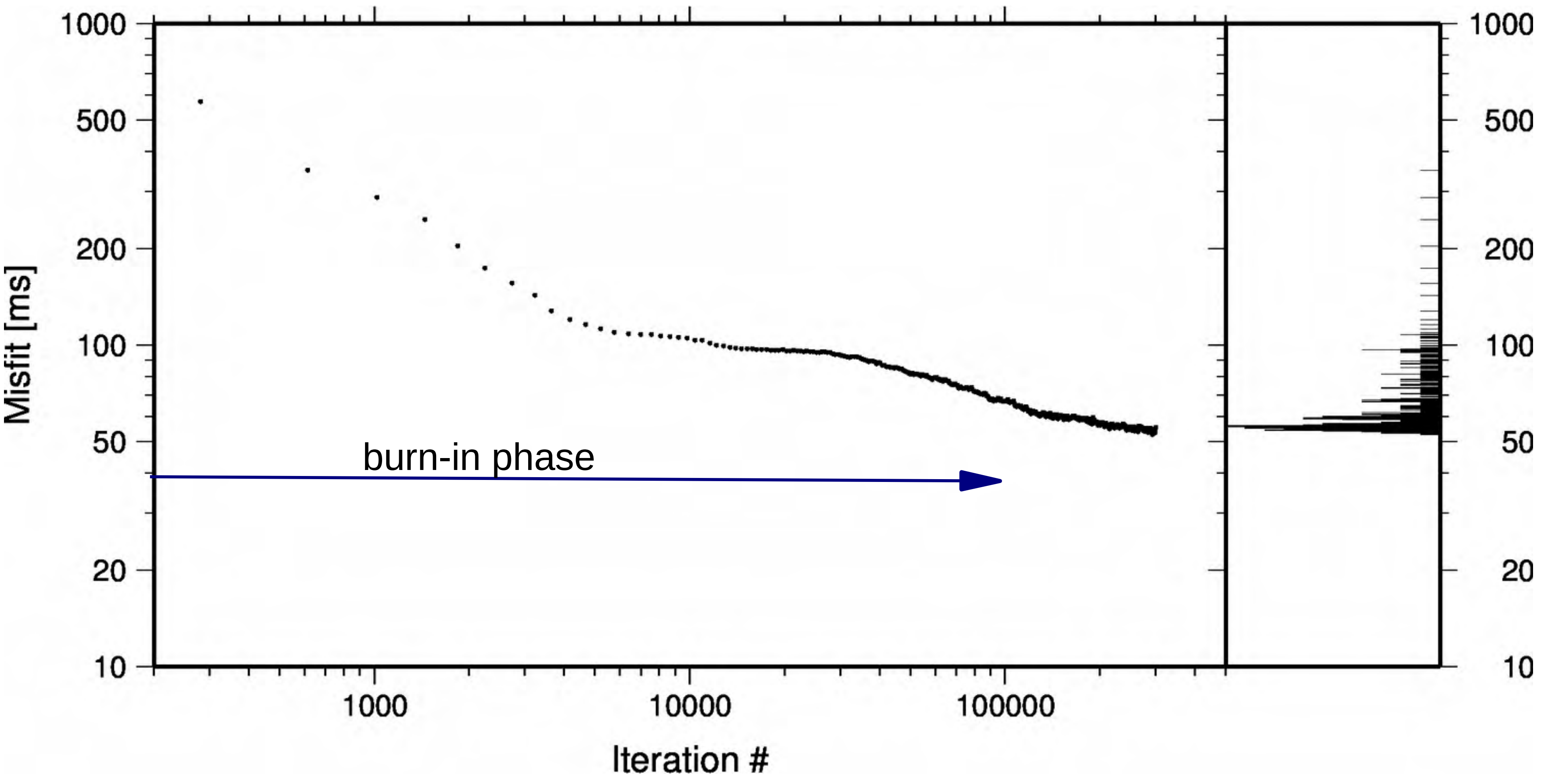


Evolution of Markov Chains



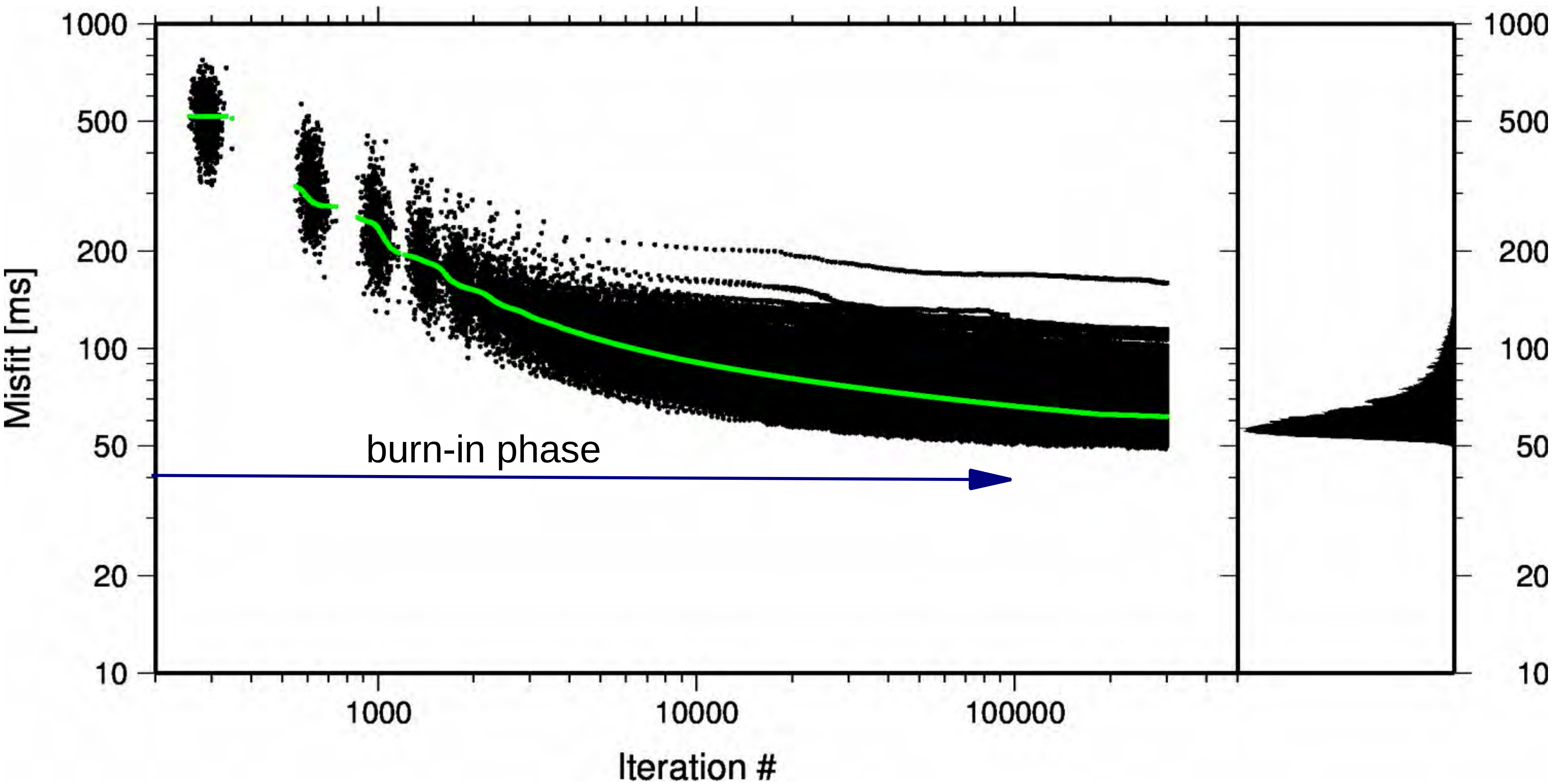
Evolution of RMS misfit of a single Markov Chain

Evolution of Markov Chains



Evolution of RMS misfit of a single Markov Chain

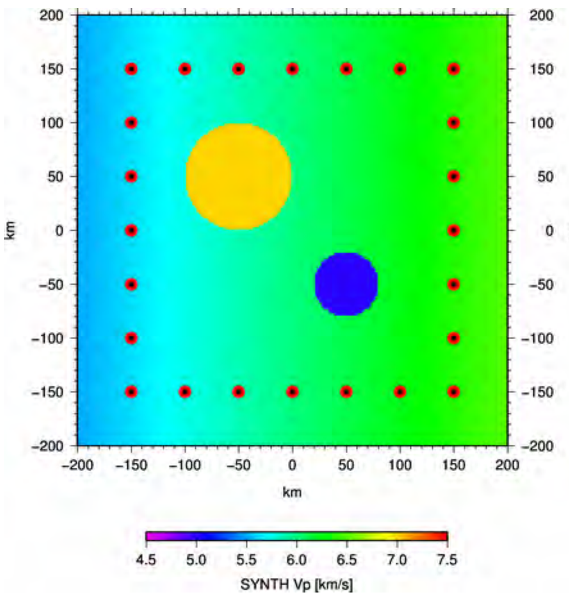
Evolution of Markov Chains



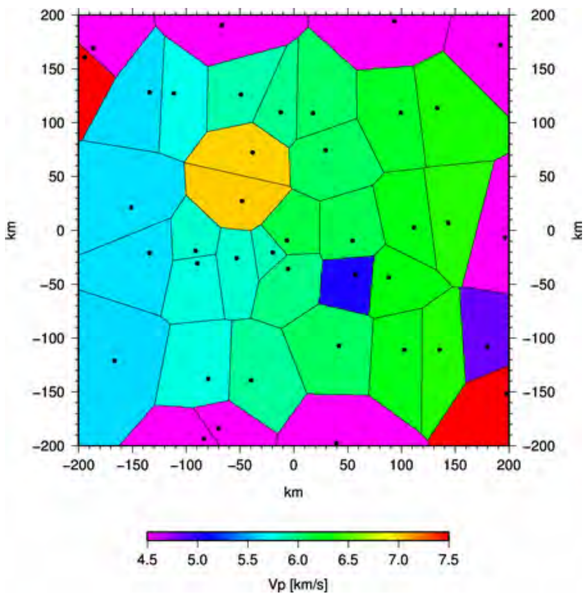
Evolution of RMS misfit of 1000 Markov Chains

Synthetic example MCMC

Input model

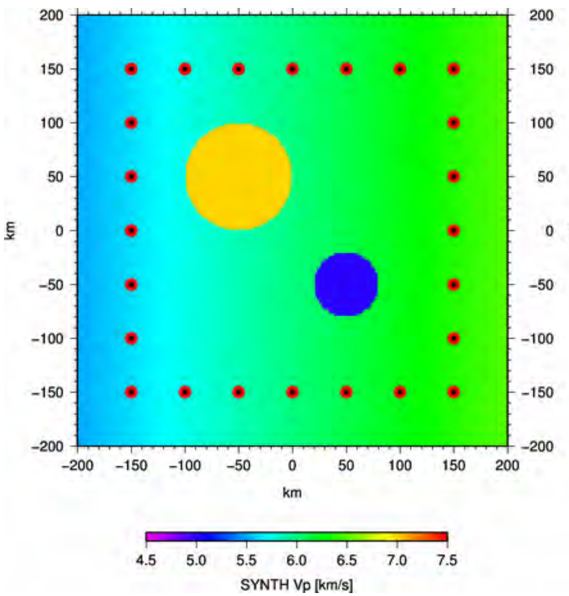


Single, best-fitting model of all chains

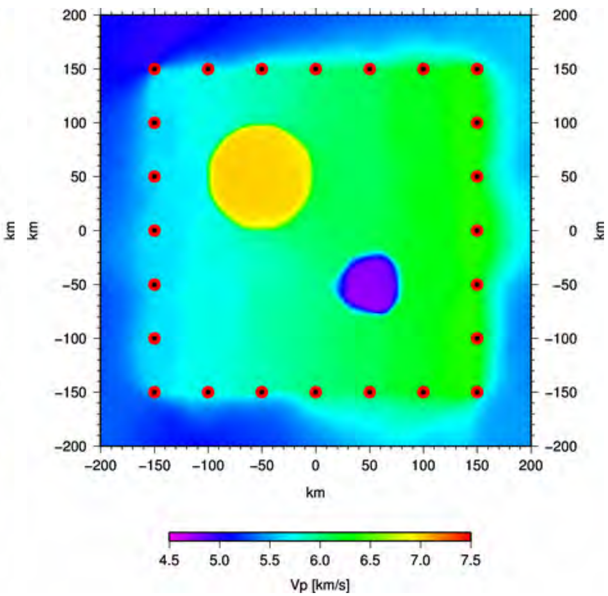


Synthetic example MCMC

Input model



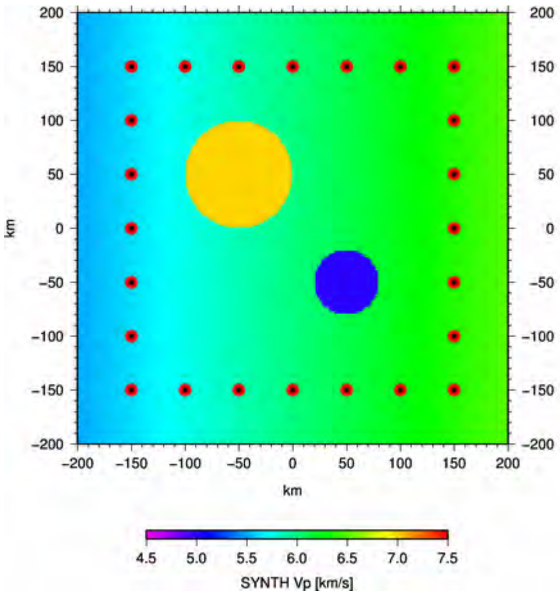
Ensemble average of all chains

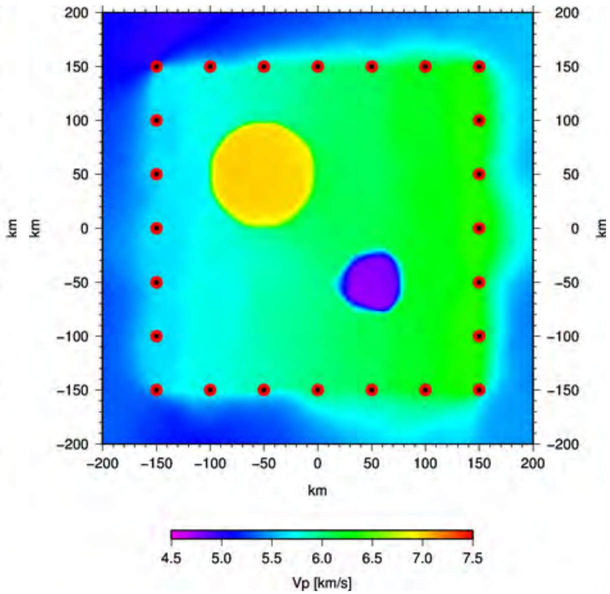


Synthetic example MCMC

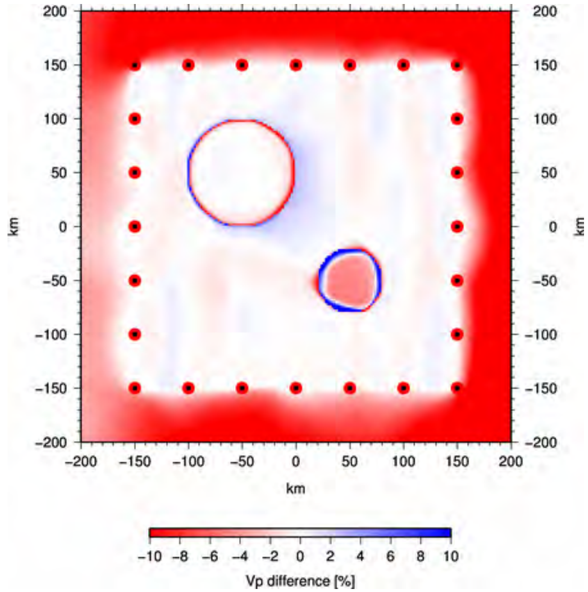
Input model

Ensemble average of all chains



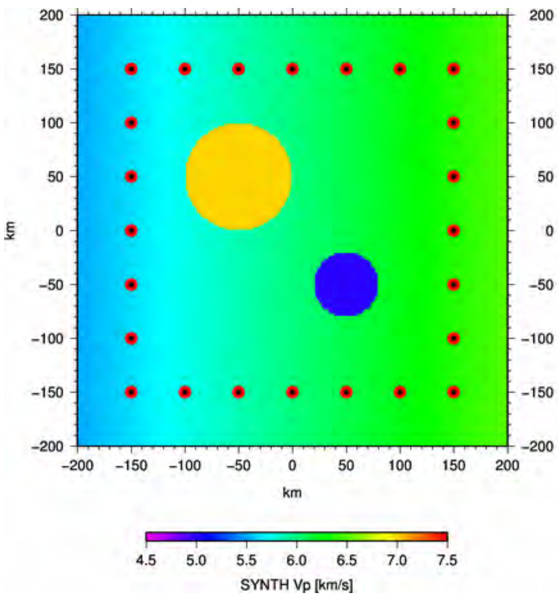


difference

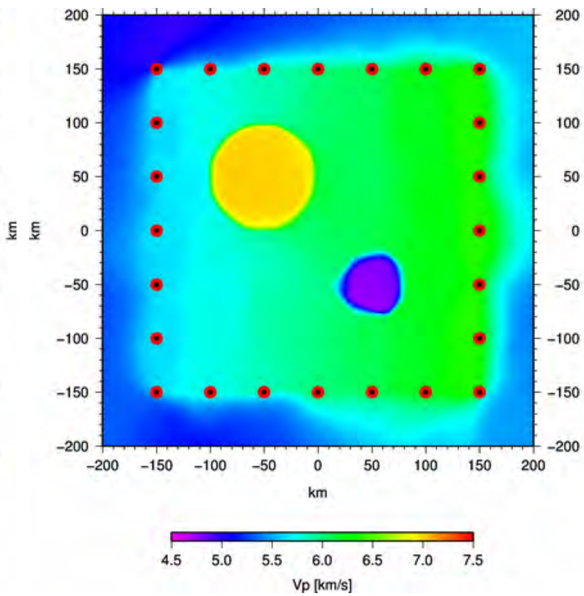


Synthetic example MCMC

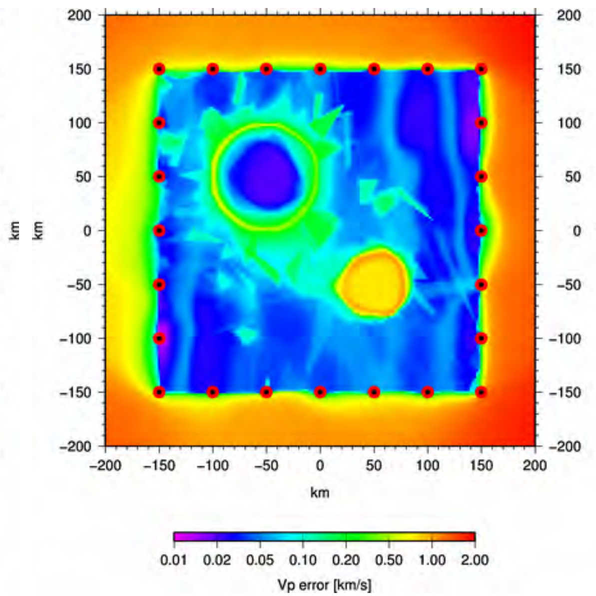
Input model



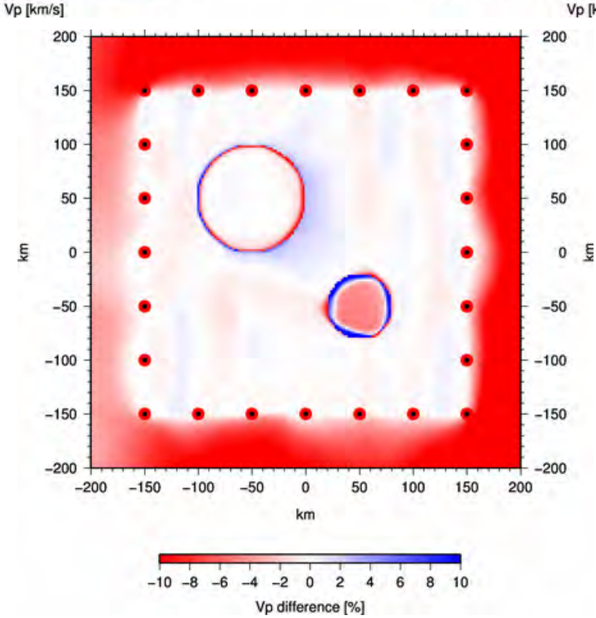
Ensemble average of all chains



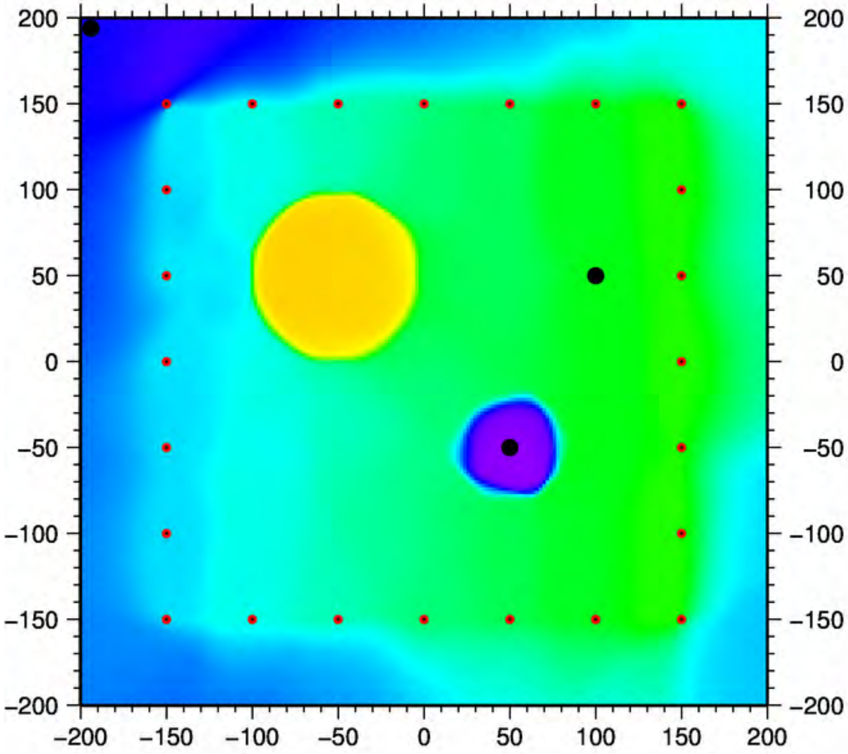
deviation = error = resolution



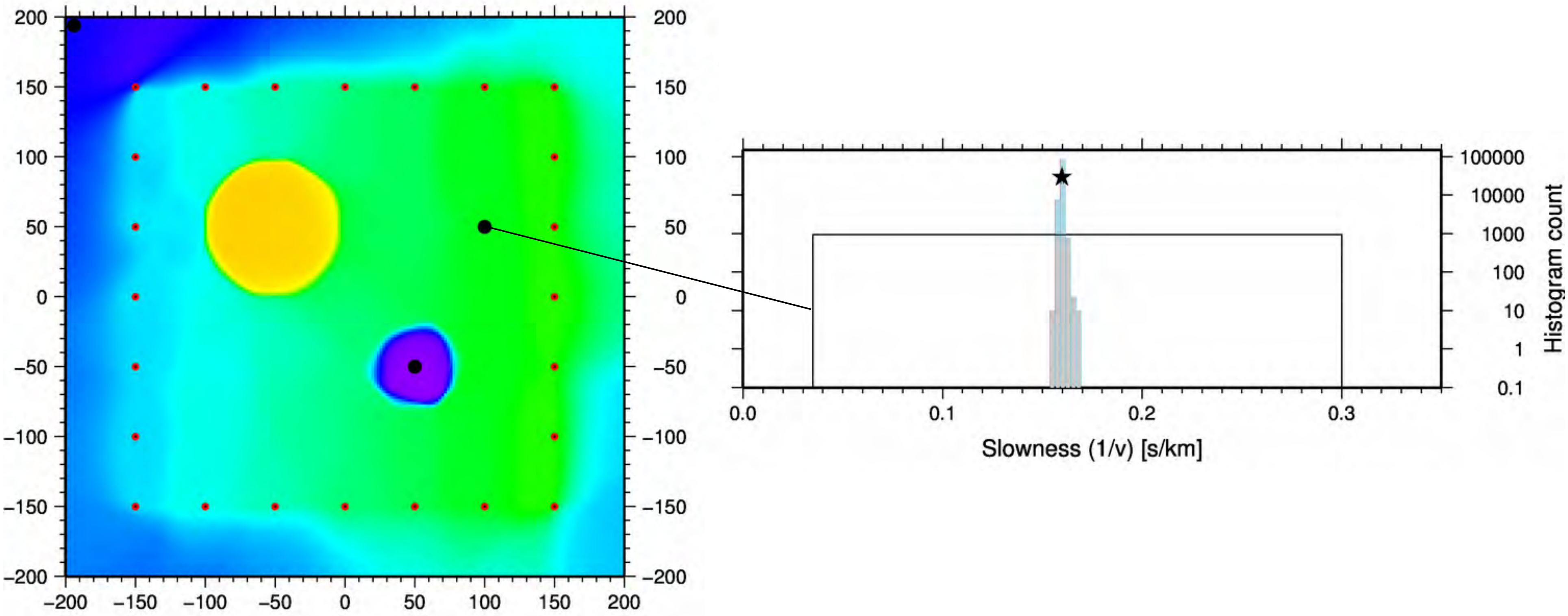
difference



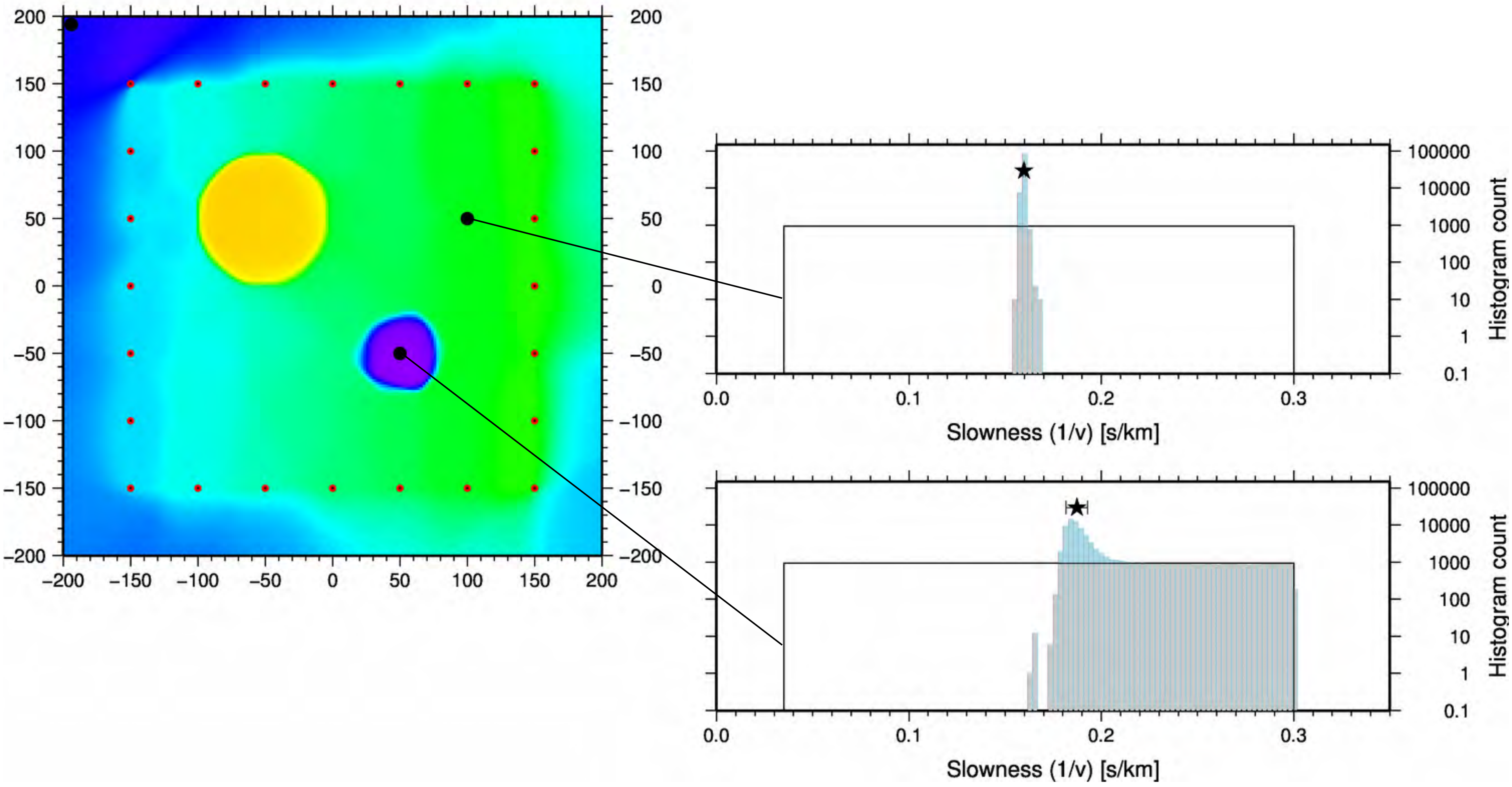
Posterior distribution



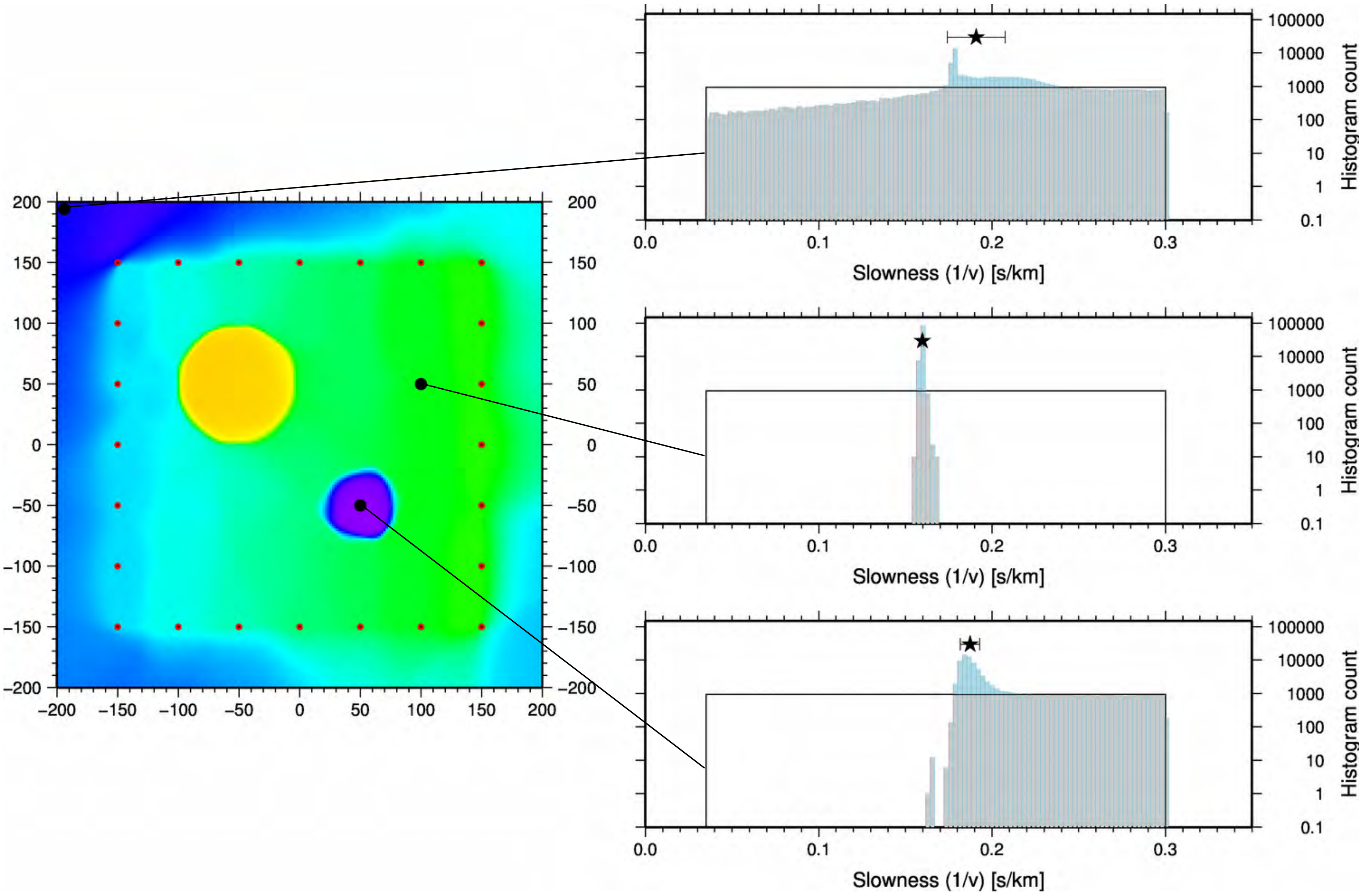
Posterior distribution



Posterior distribution

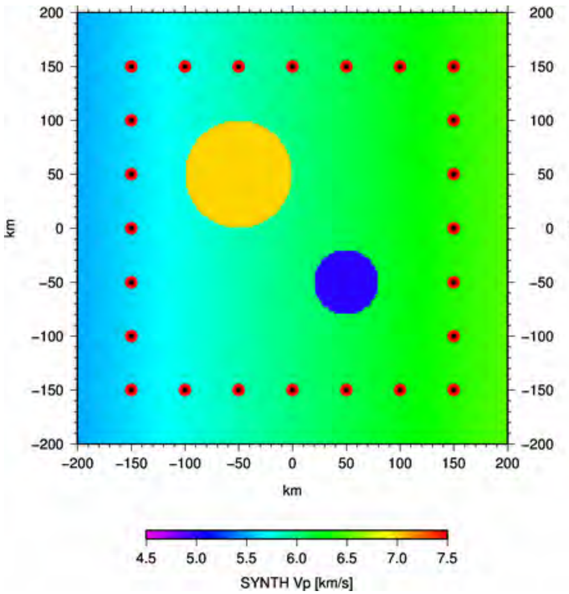


Posterior distribution

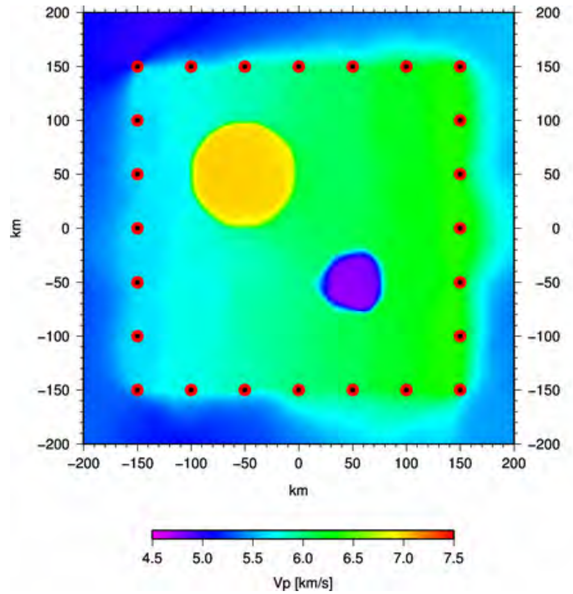


Super resolution

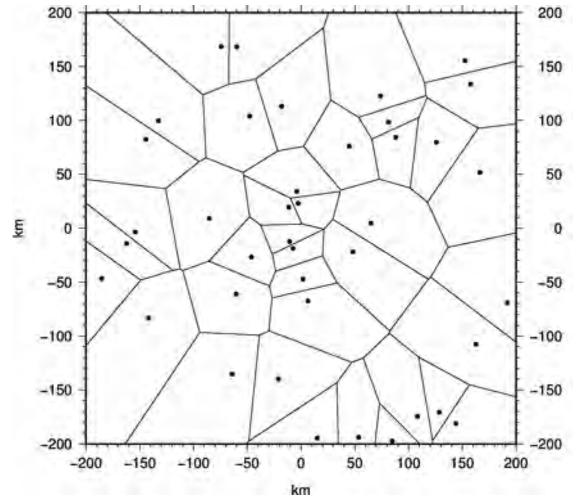
true model



average of >50000 models

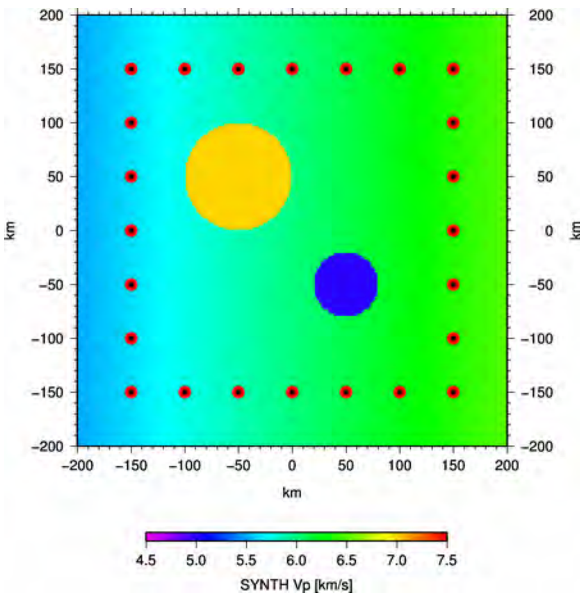


one model

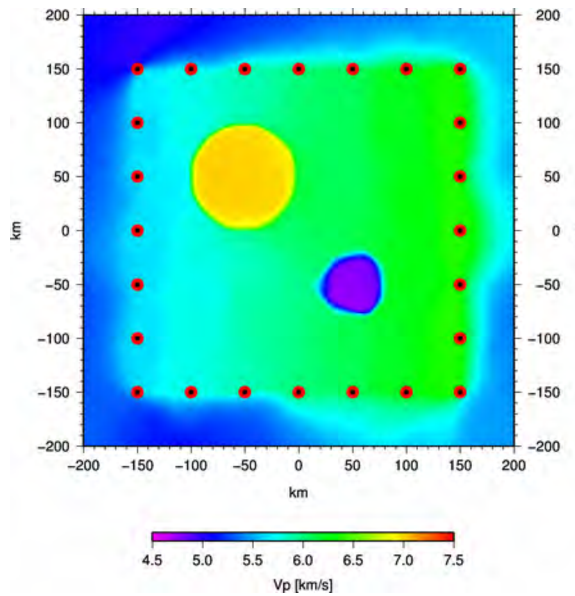


Super resolution

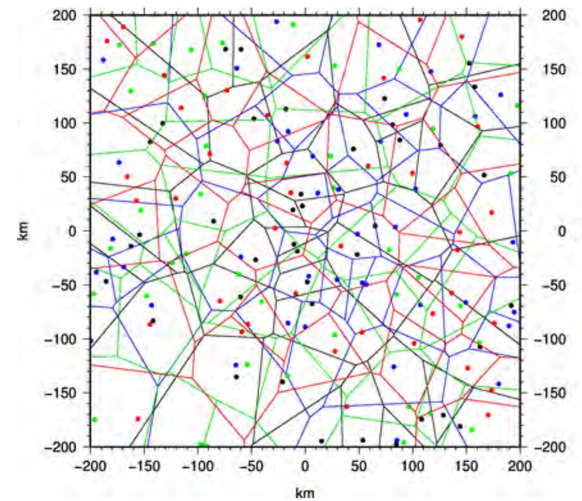
true model



average of >50000 models



4 models



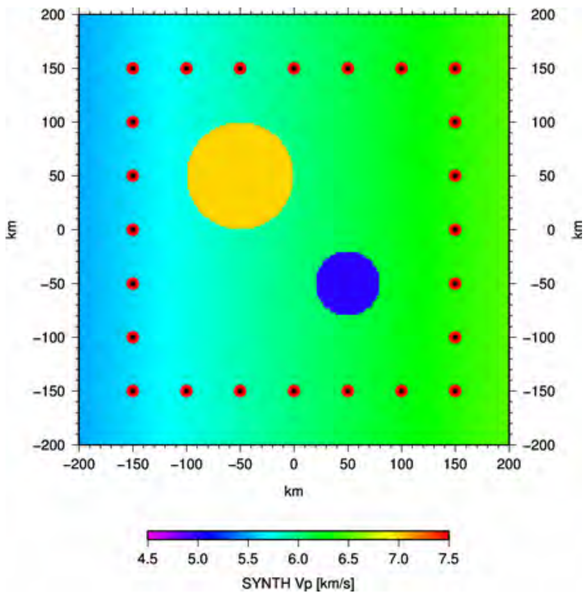
averaging over sparse models results a higher resolved reference model

effective spatial resolution can be much higher than average Voronoi cell size!

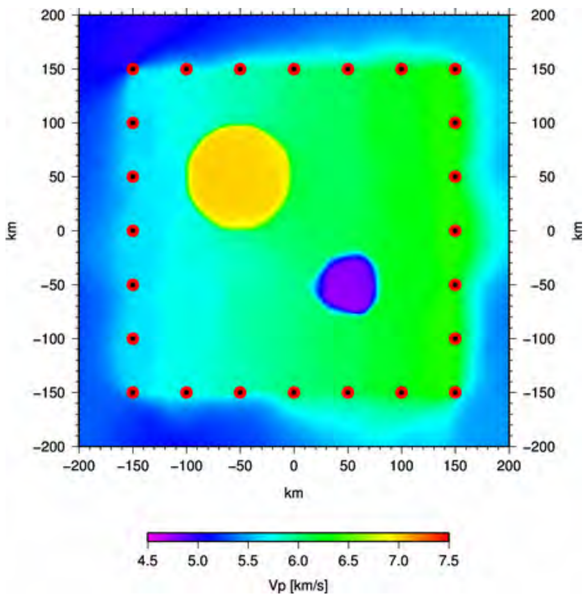
“Wisdom of the cloud”→ super resolution!

Synthetic example MCMC

true model



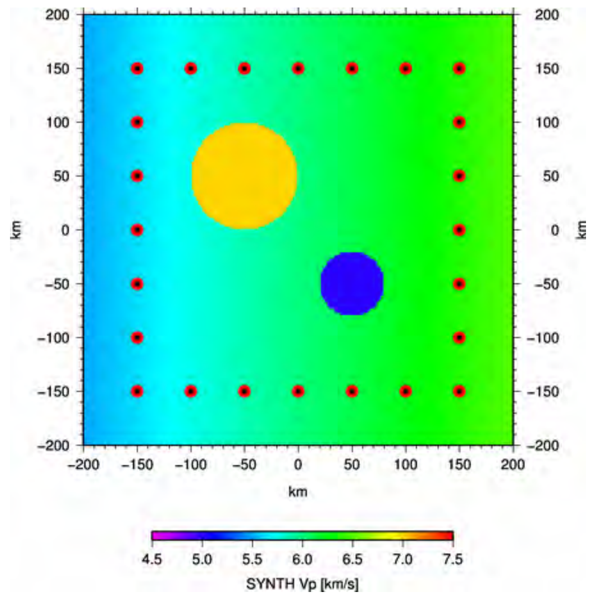
40 cells



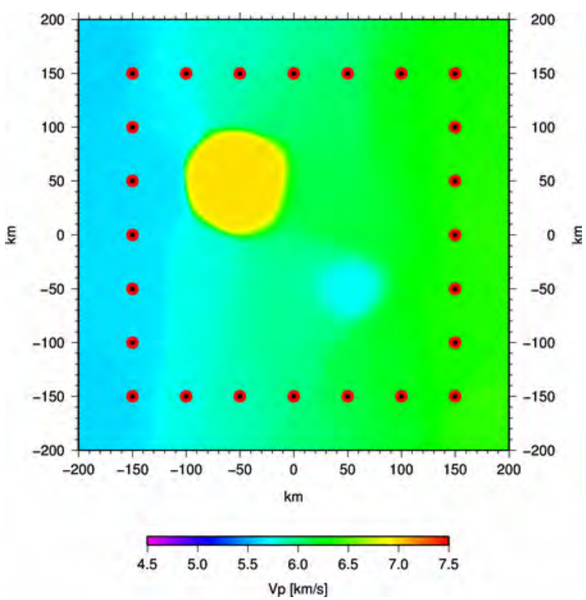
Synthetic example MCMC

different number of cells

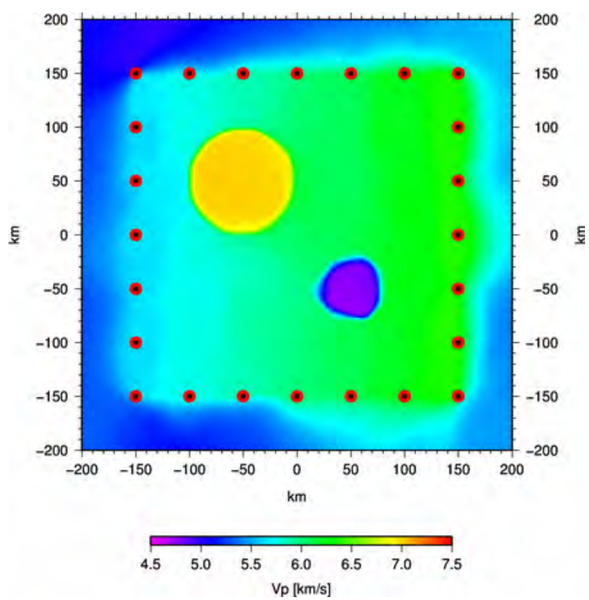
true model



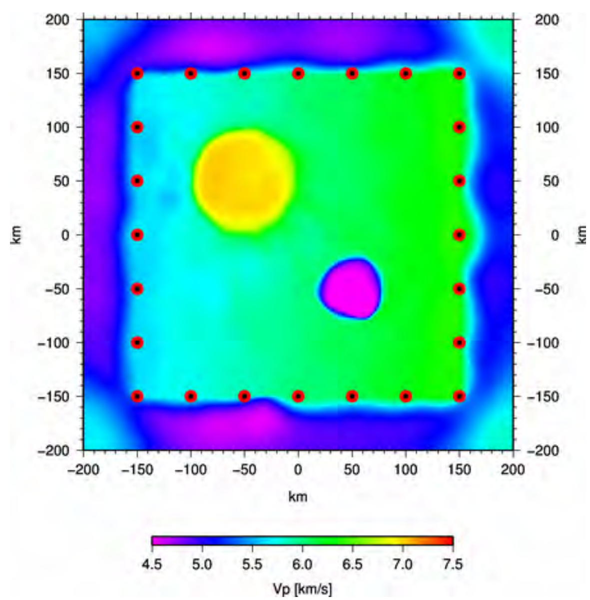
15 cells



40 cells

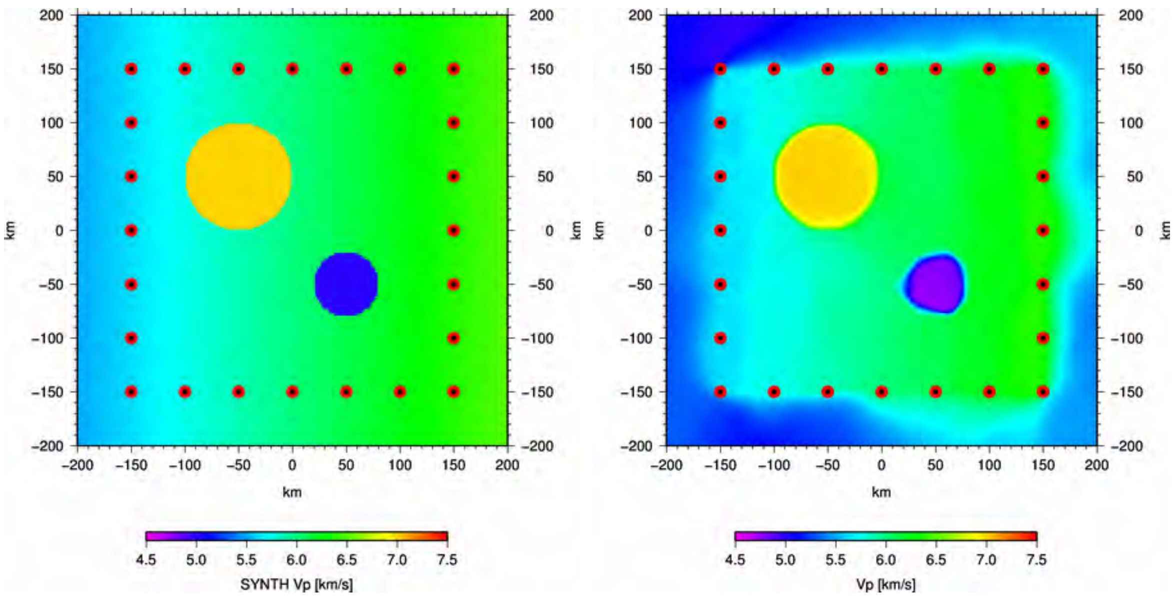


100 cells



Synthetic example MCMC

Ensemble average of all chains

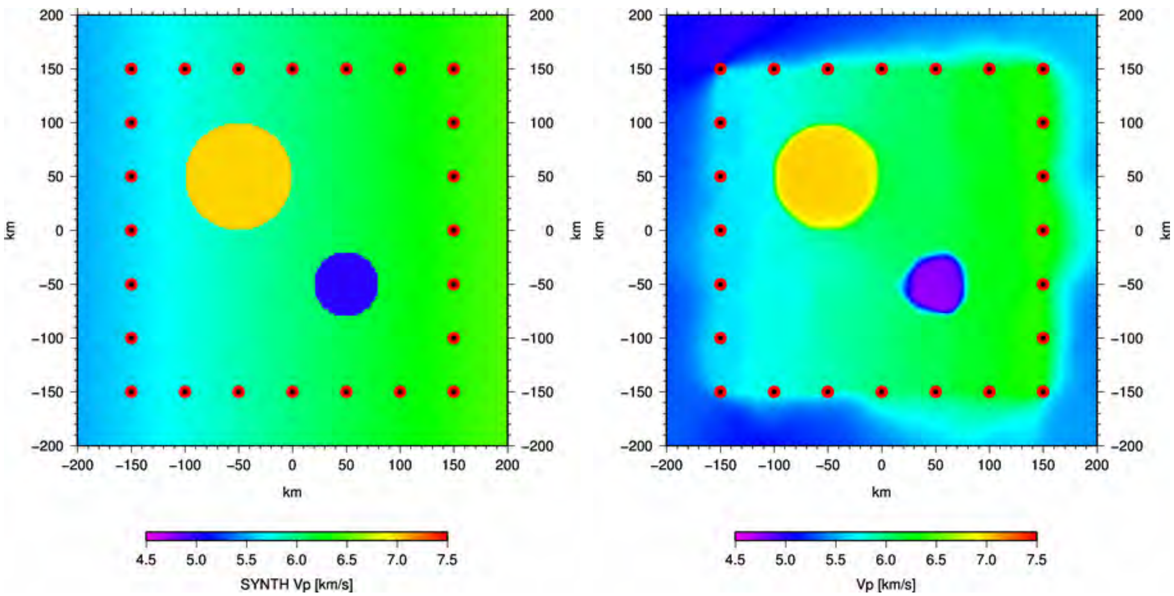


good recovery

but number of cells given prior the inversion

Synthetic example MCMC

Ensemble average of all chains



good recovery

but number of cells given prior the inversion ← bad!

solution: lets treat the number of cells (complexity of model) as unknown and invert for it! Let the data decide...

extending the Metropolis-Hastings algorithm to be transdimensional

Markov chain Monte Carlo

Implementation of transdimensional Metropolis-Hastings Algorithm (reversible jump)

1. randomly pick one cell in model $\mathbf{m}_0$

   1a. randomly change slowness in this cell to create model $\mathbf{m}_1$

   or

   1b. randomly change position of this cell to create model $\mathbf{m}_1$

   or

   1c. delete this cell to create model $\mathbf{m}_1$

or

2. add a new cell at random position/slowness to create model $\mathbf{m}_1$

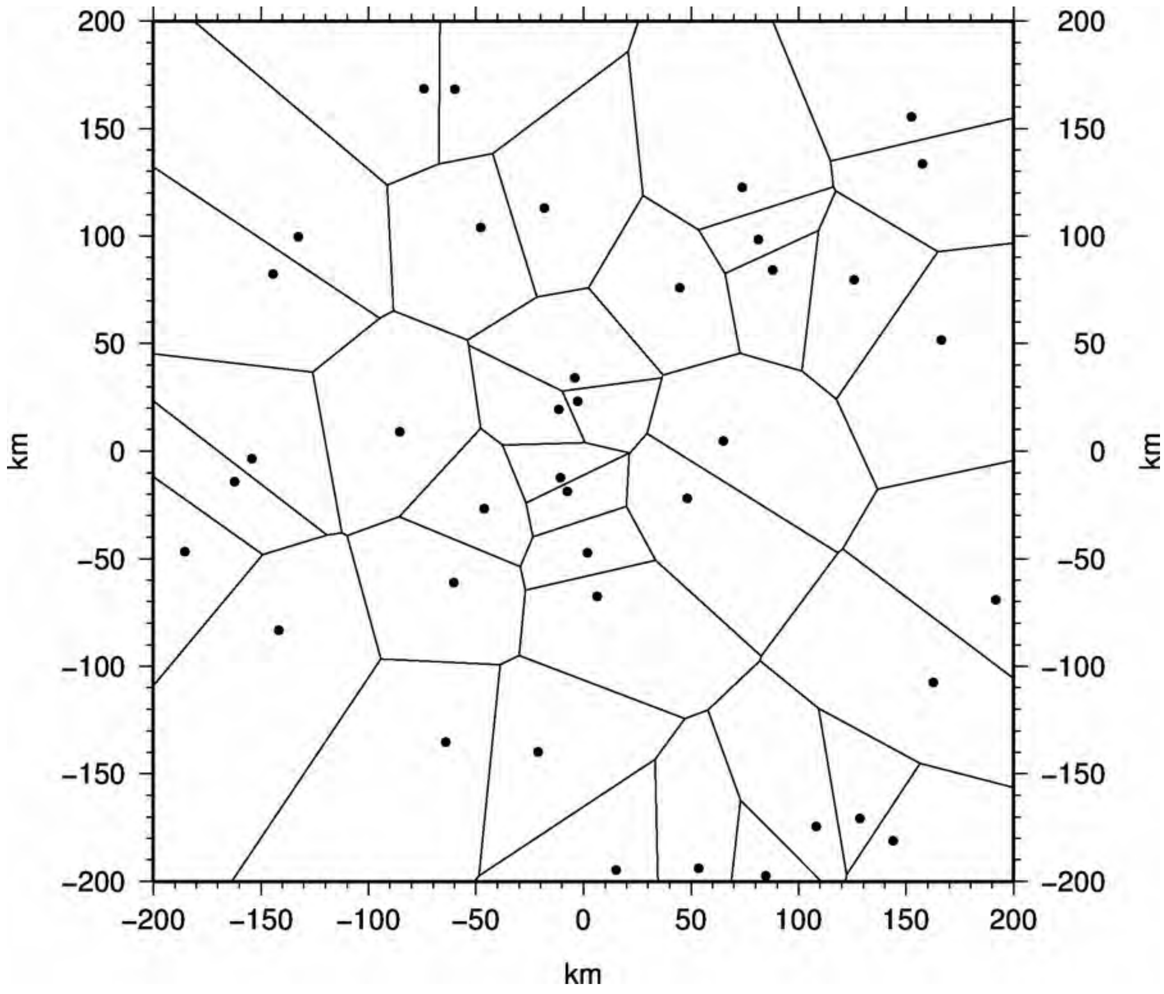
3. solve forward problem, i.e. calculate misfit and likelihood of model $\mathbf{m}_1$

4. calculate acceptance probability of model $\mathbf{m}_1$

5. accept or reject model $\mathbf{m}_{1,}$ then goto 1

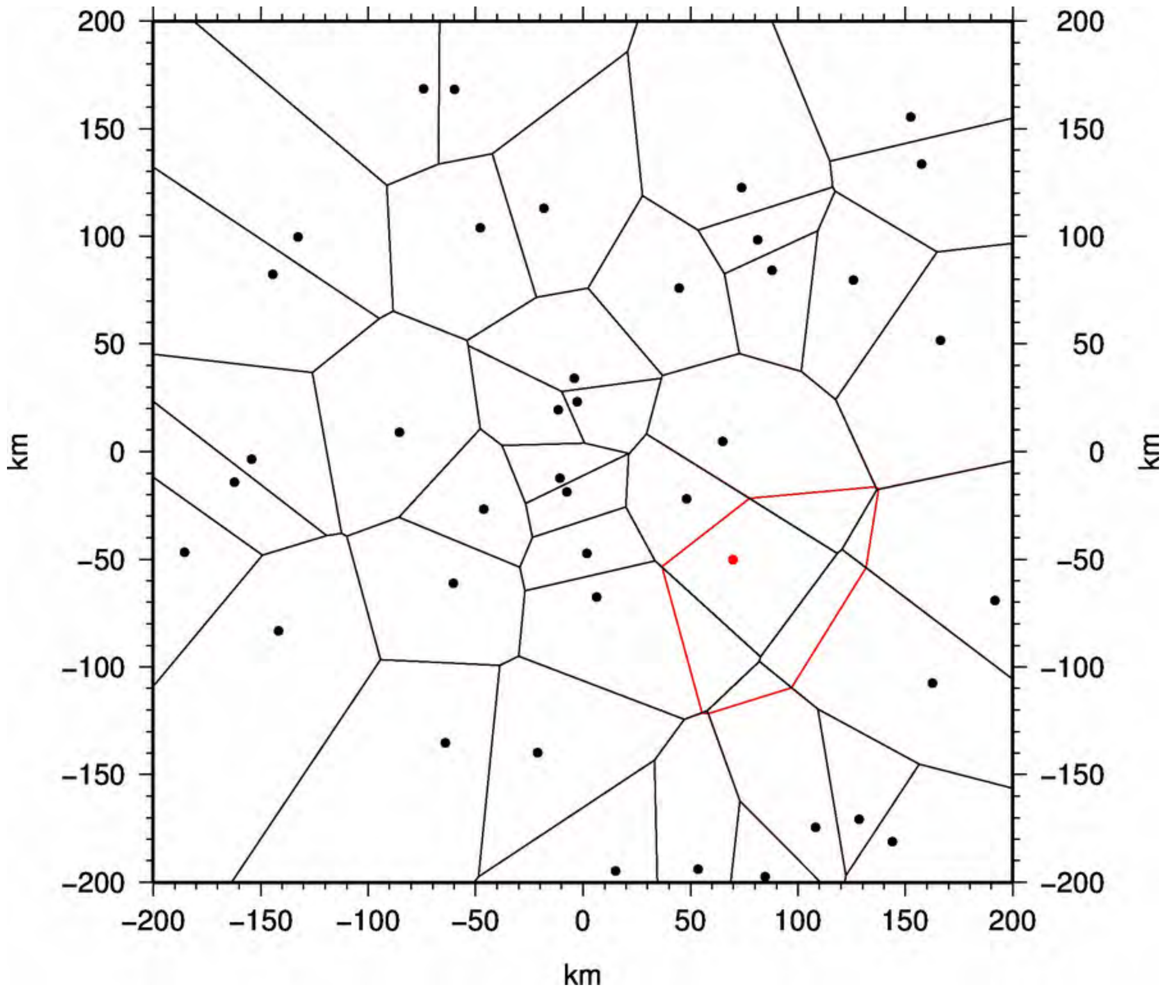
Synthetic example MCMC

40 Voronoi cells, randomly distributed



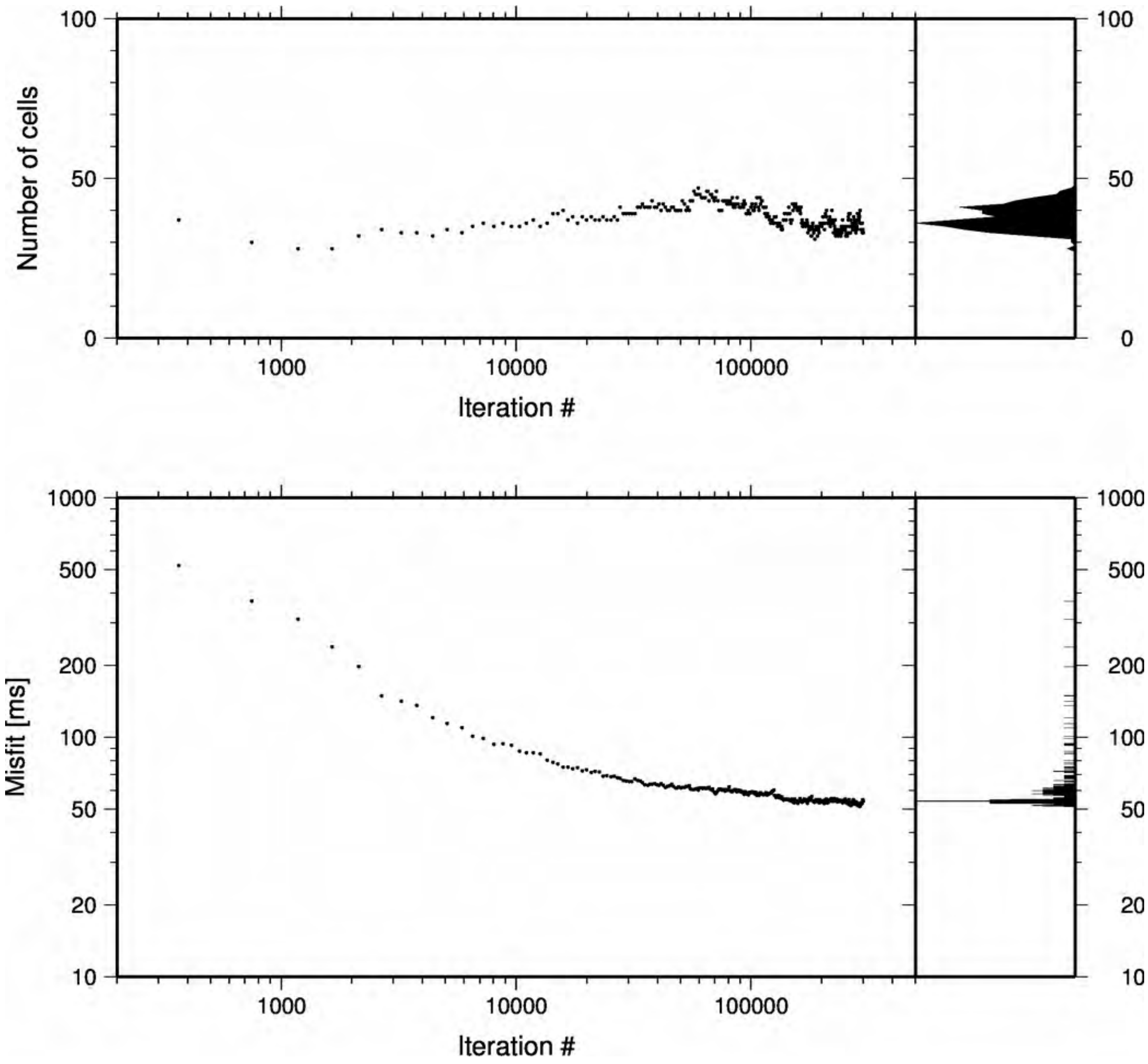
Synthetic example MCMC

40 Voronoi cells, one cell added (birth)



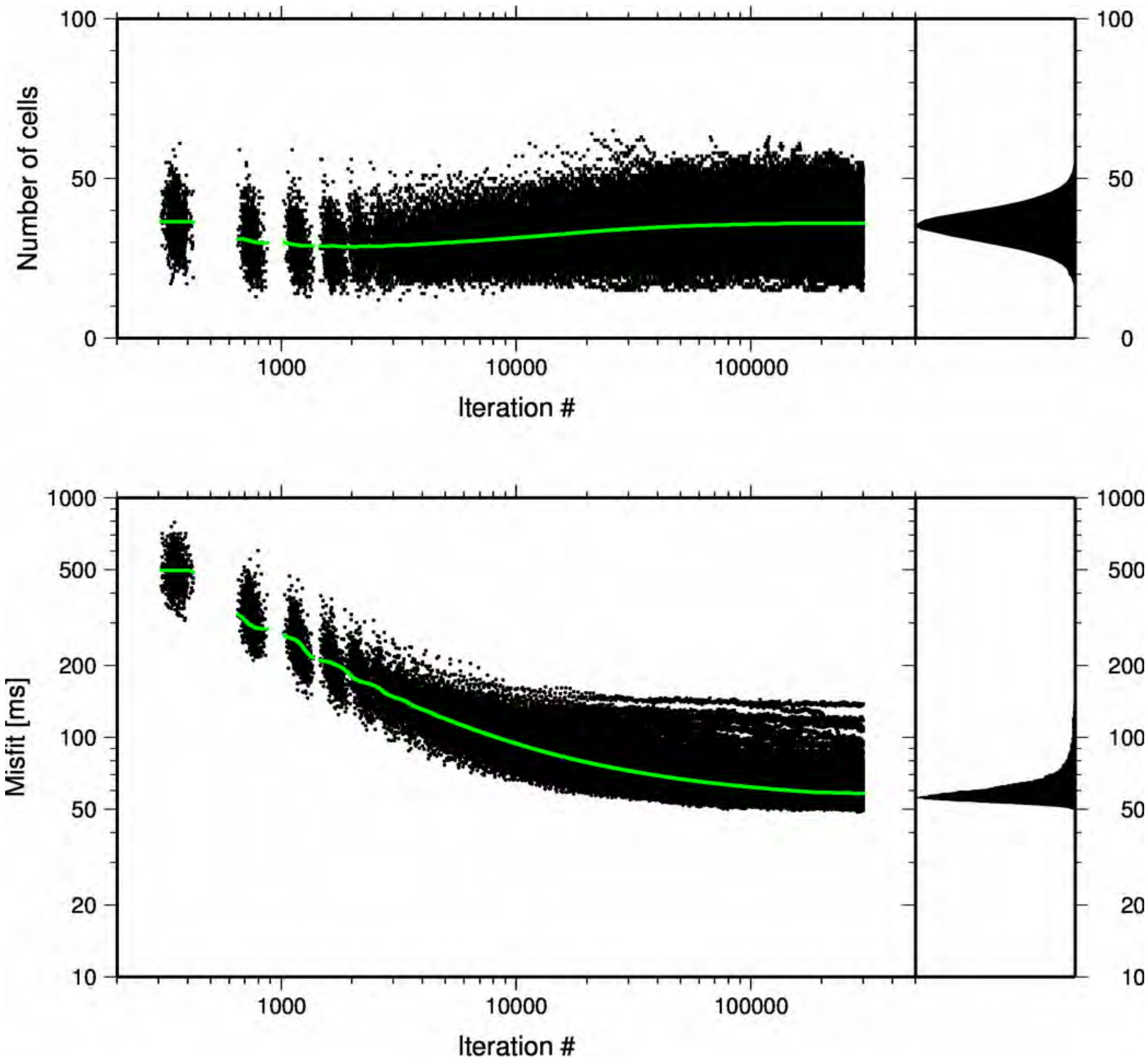
acceptance of new model: only if velocity differs significantly from original cell!

Evolution of Markov Chains



Evolution of RMS misfit of a single Markov Chain

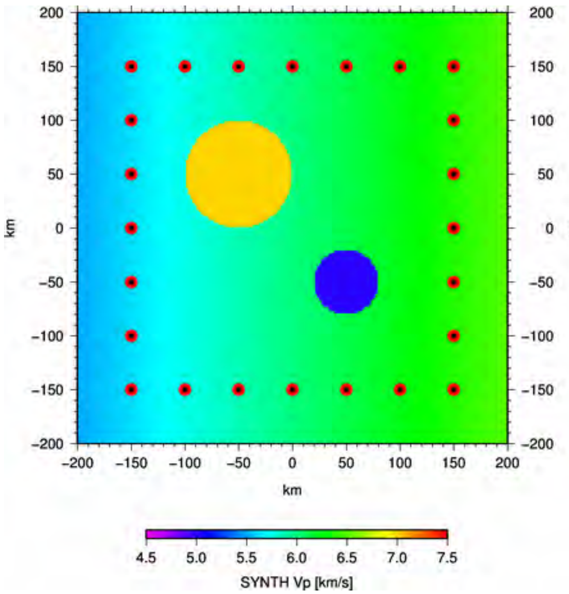
Evolution of Markov Chains

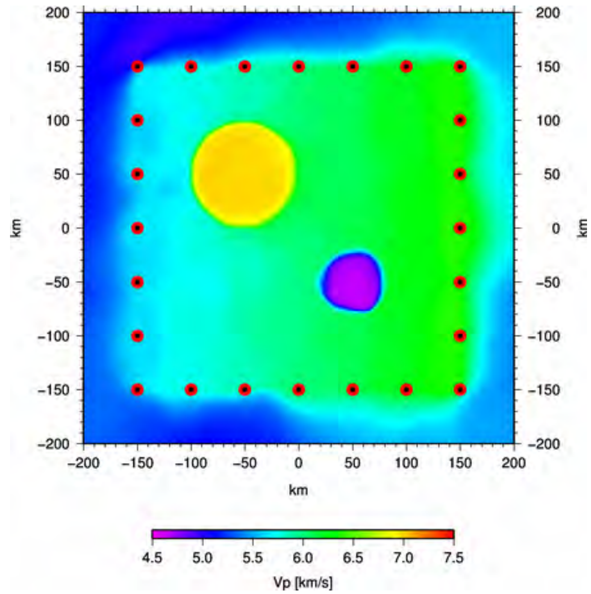


Evolution of RMS misfit of 1000 Markov Chains

Final trans-dimensional model

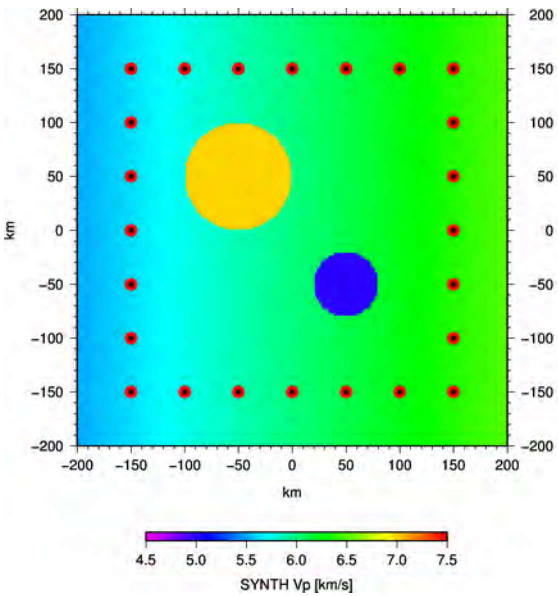
Ensemble average of all chains with fixed noise=0.05

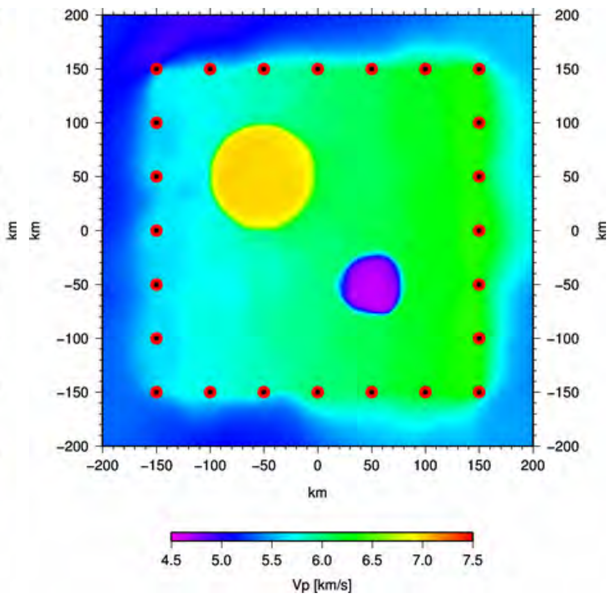


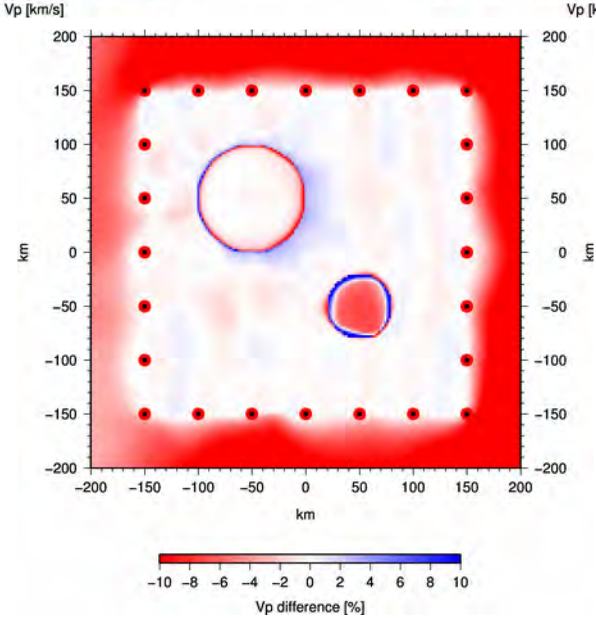


Final trans-dimensional model

Ensemble average of all chains with fixed noise=0.05

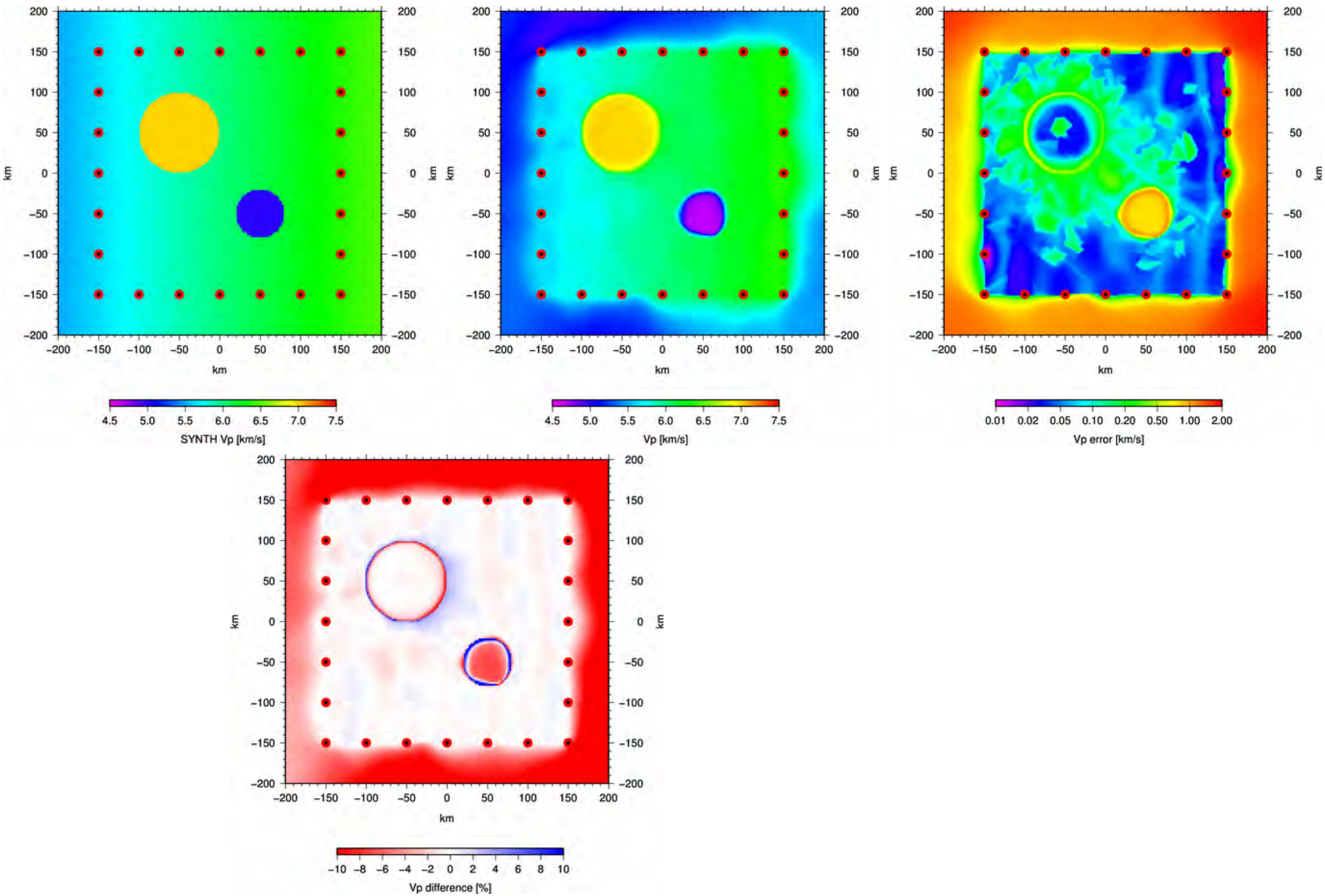






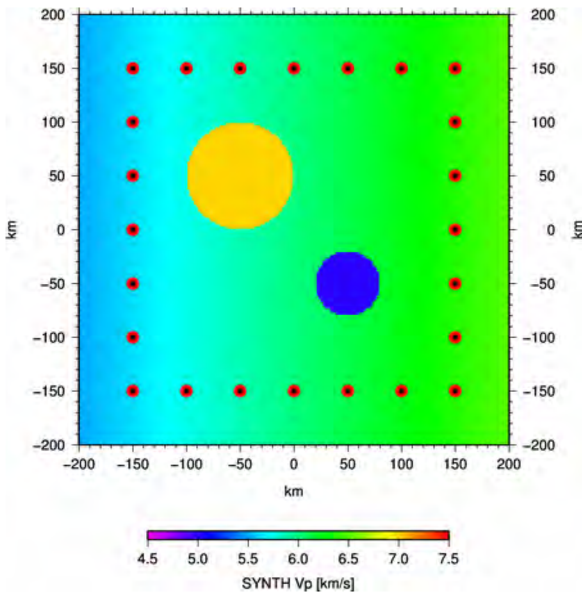
Final trans-dimensional model

Ensemble average of all chains with fixed noise=0.05

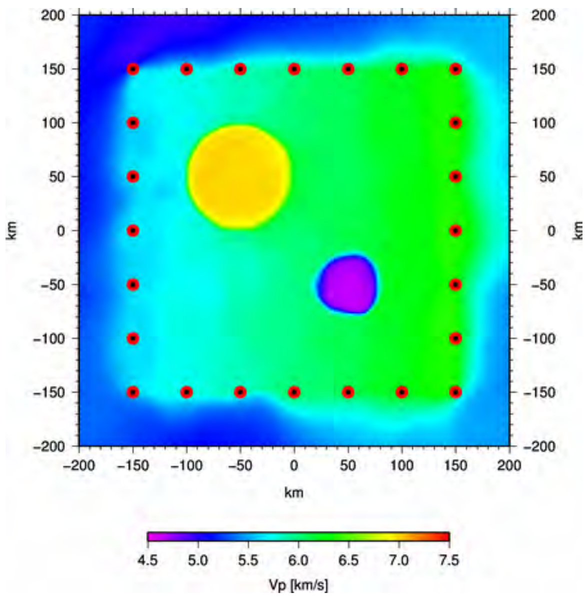


Synthetic example MCMC

true model

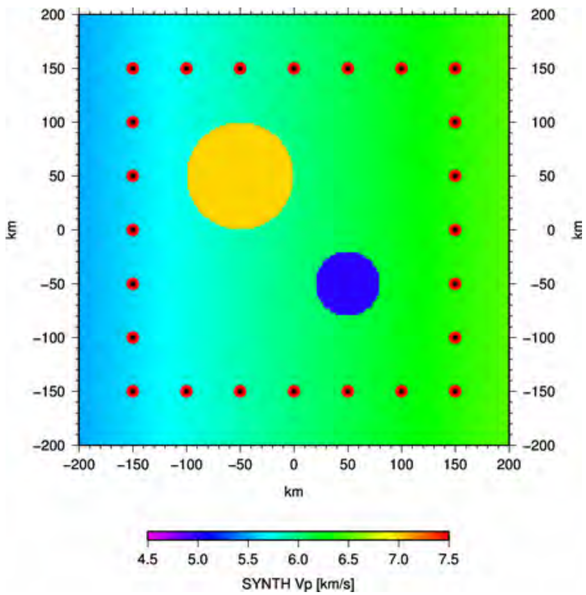


noise=0.05



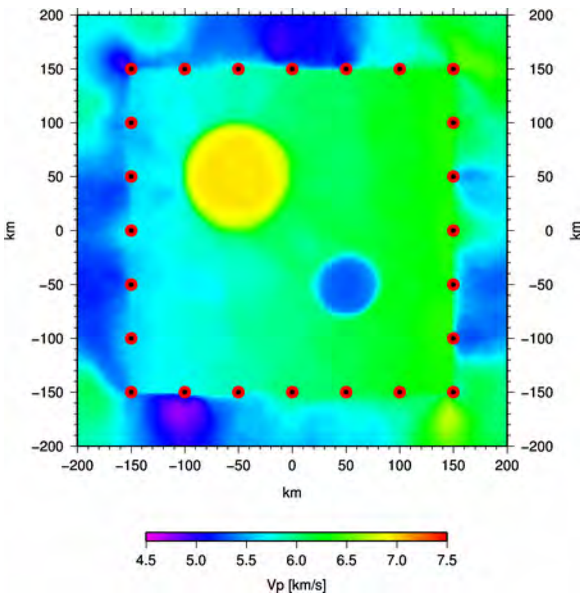
Synthetic example MCMC

true model

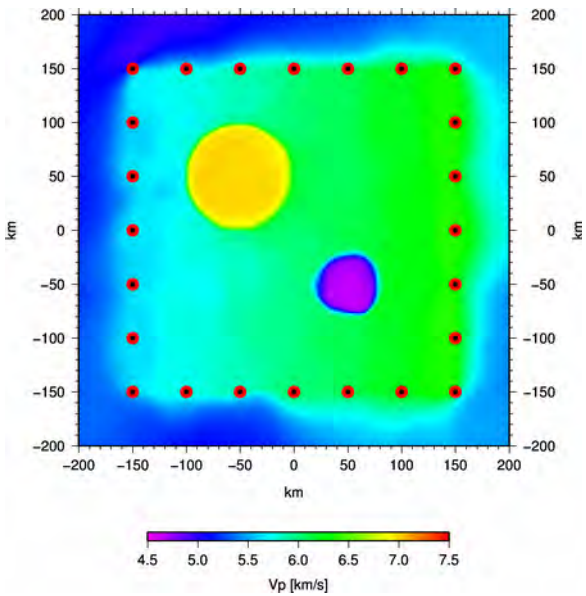


different assumed noise
(data variance)

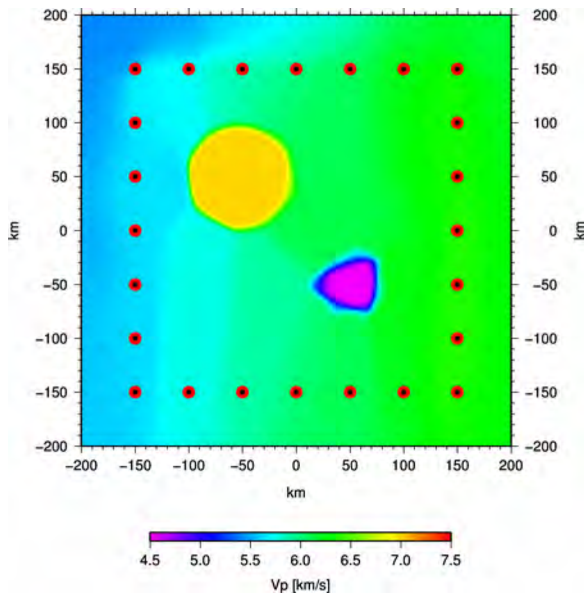
noise=0.001



noise=0.05

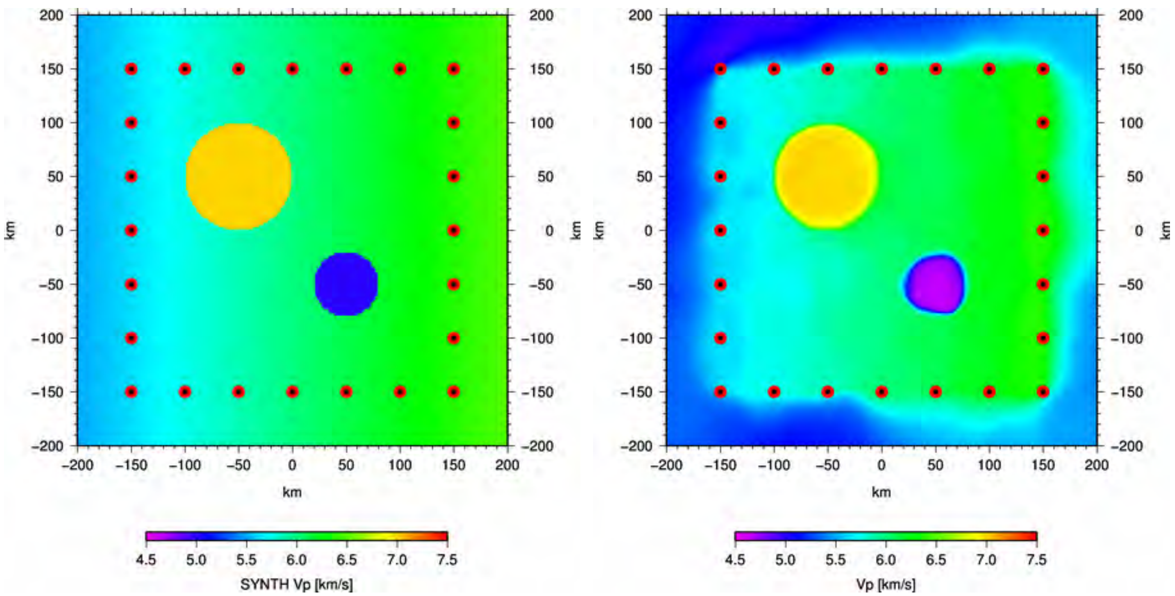


noise=0.5



Synthetic example MCMC

Ensemble average of all chains

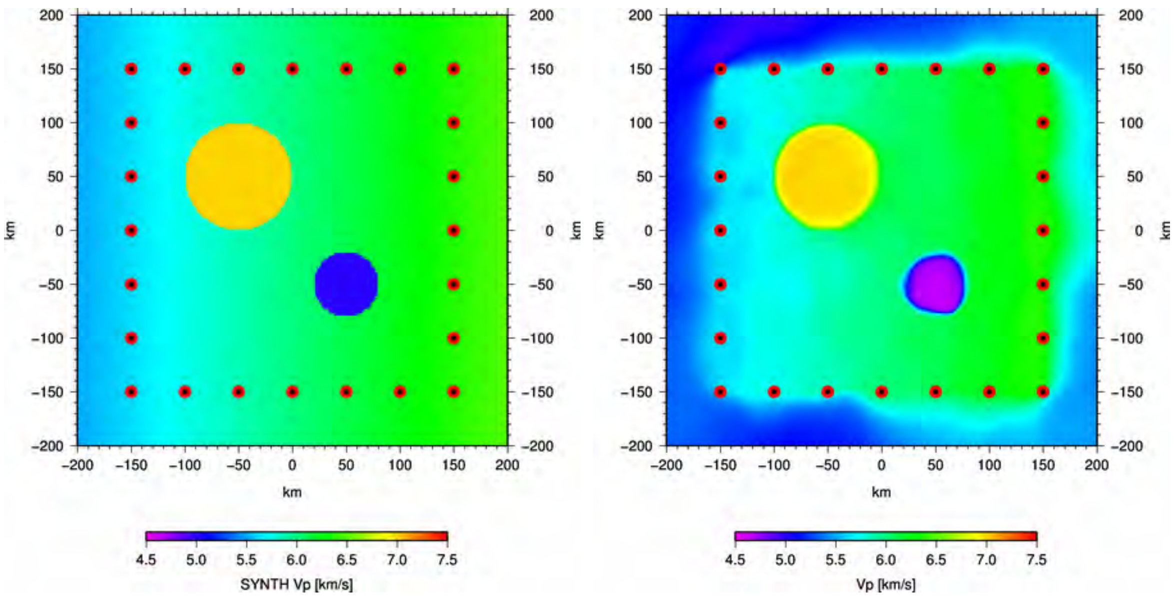


again, good recovery

assumed noise level given prior the inversion

Synthetic example MCMC

Ensemble average of all chains



again, good recovery

assumed noise level given prior the inversion ← bad!

solution: lets treat the noise level as unknown and invert for it! Let the data decide...

extending the Metropolis-Hastings algorithm to be hierarchical

Markov chain Monte Carlo

Implementation of transdimensional, hierarchical Metropolis-Hastings Algorithm

1. randomly pick one cell in model $\mathbf{m}_0$

   1a. randomly change slowness in this cell to create model $\mathbf{m}_1$

   or

   1b. randomly change position of this cell to create model $\mathbf{m}_1$

   or

   1c. delete this cell to create model $\mathbf{m}_1$

or

2. add a new cell at random position/slowness to create model $\mathbf{m}_1$

or

4. randomly change data variance (noise) to create model $\mathbf{m}_1$

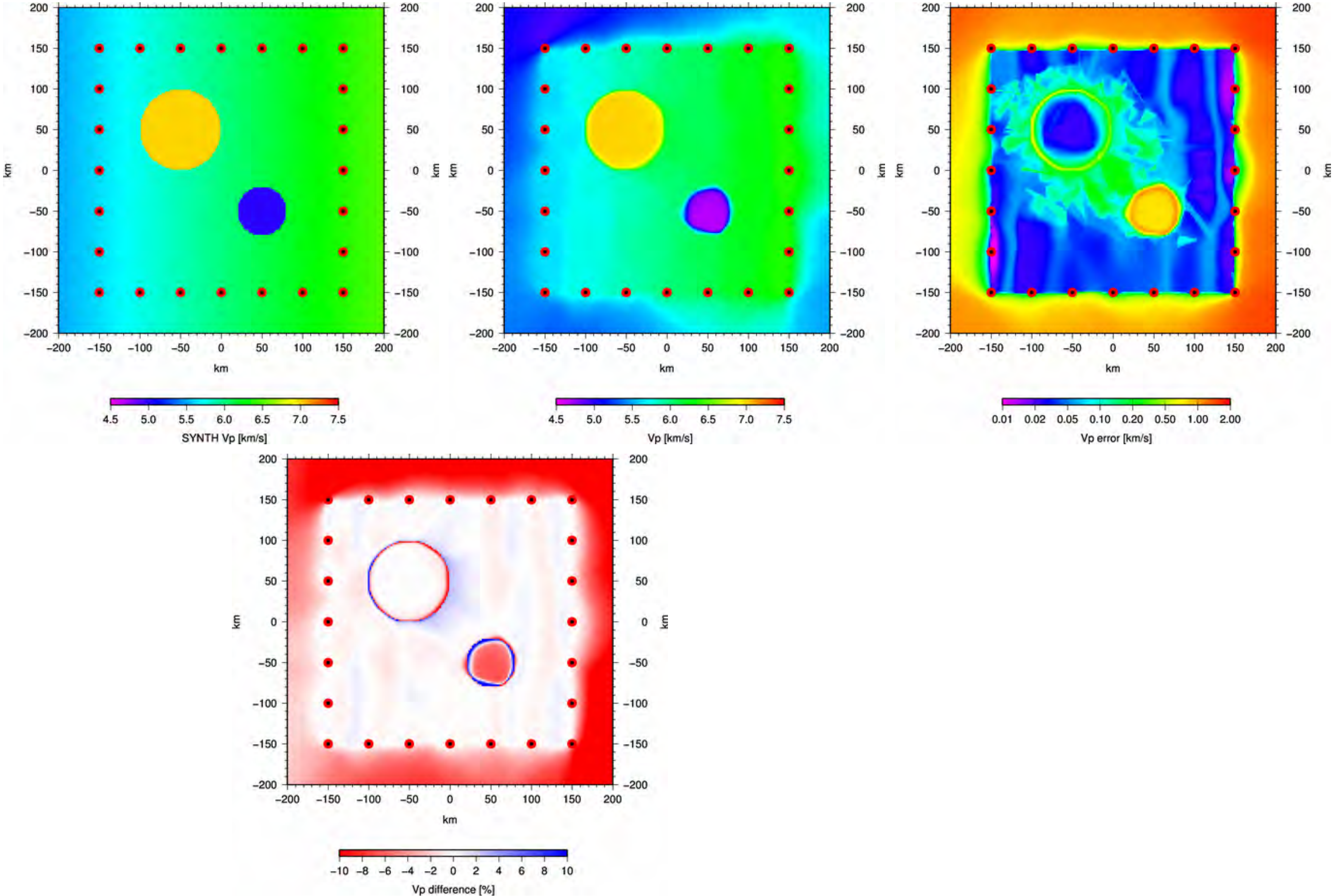
5. calculate likelihood of model $\mathbf{m}_1$

6. calculate acceptance probability of model $\mathbf{m}_1$

7. accept or reject model $\mathbf{m}_1$, then goto 1

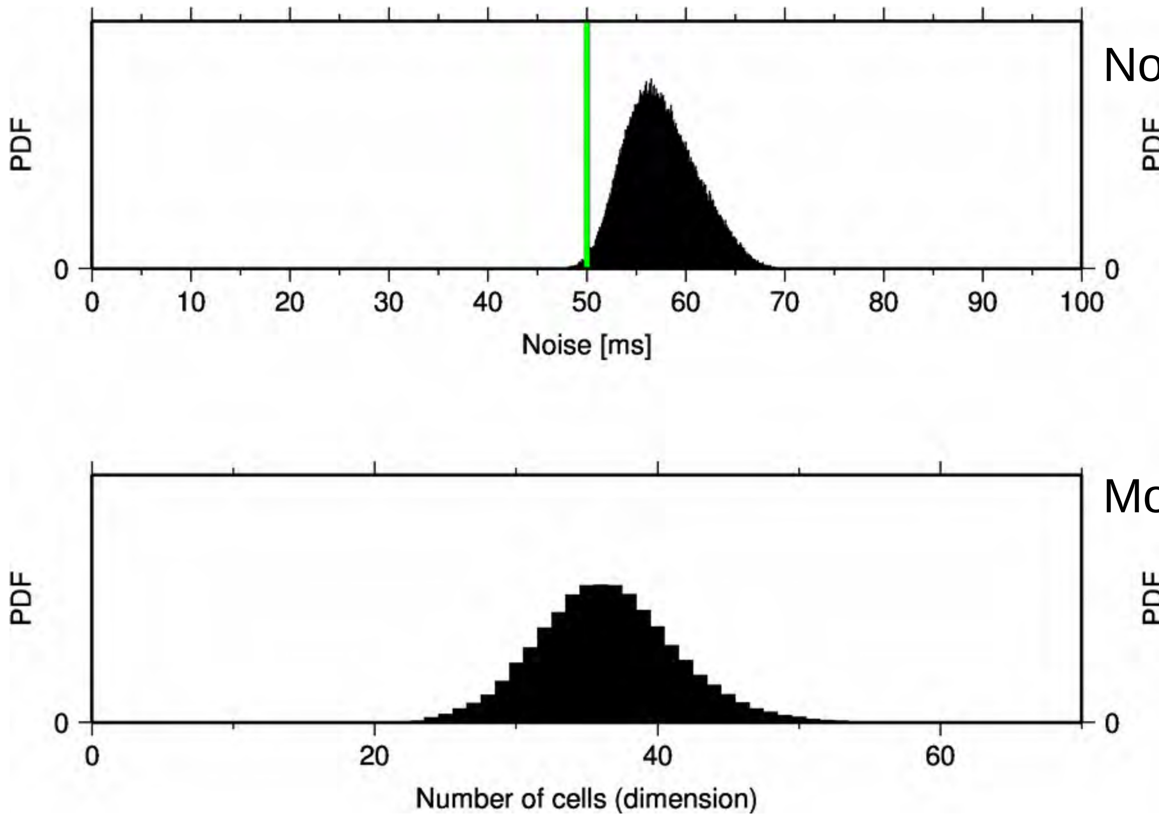
Final trans-dimensional hierarchical model

Ensemble average of all chains



Final trans-dimensional hierarchical model

Ensemble average of all chains: recovered noise level and model dimension

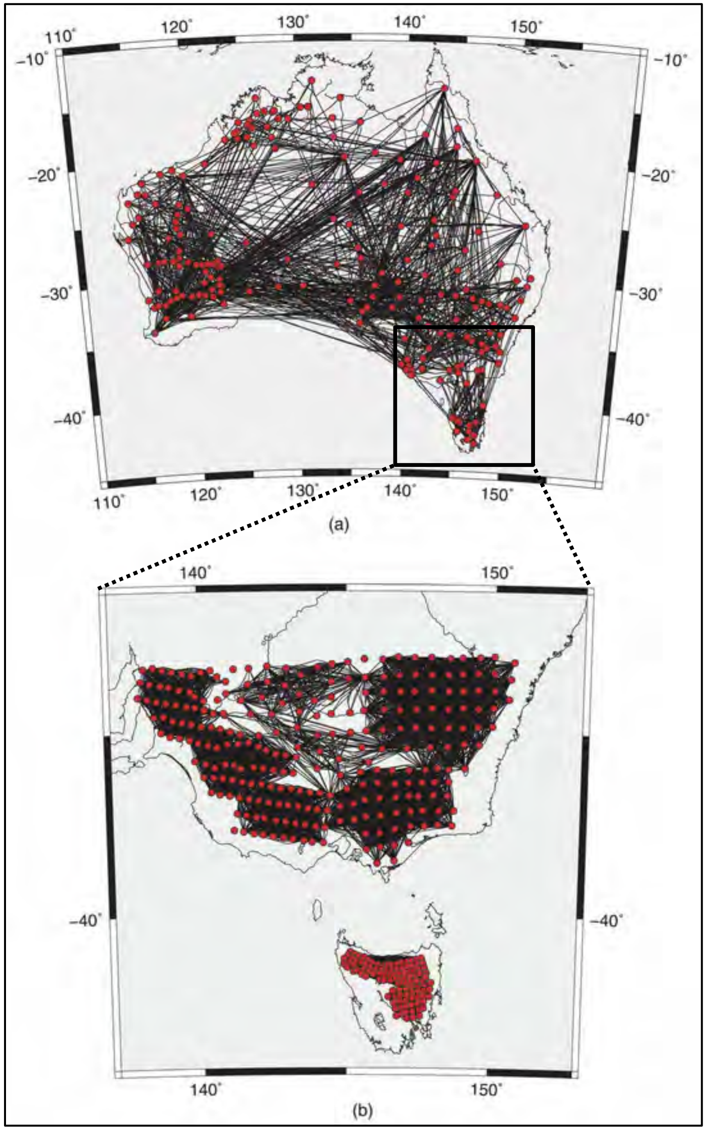


Noise: what we don’t fit with the data

Model complexity: natural parsimony

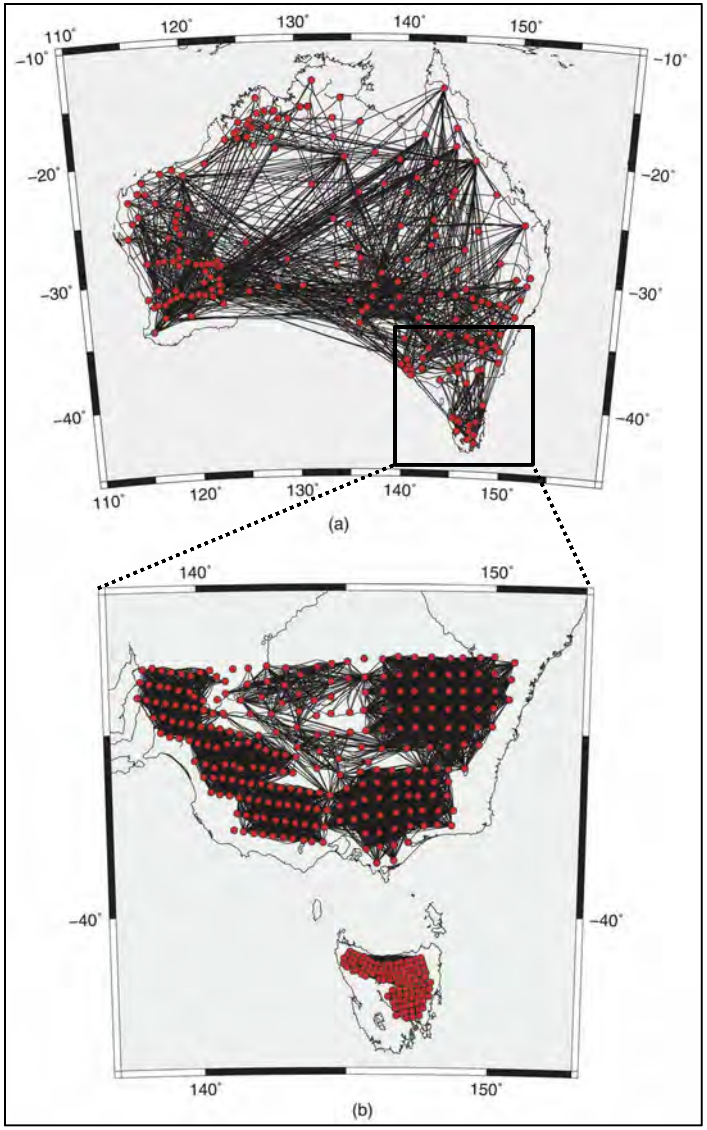
Real world example

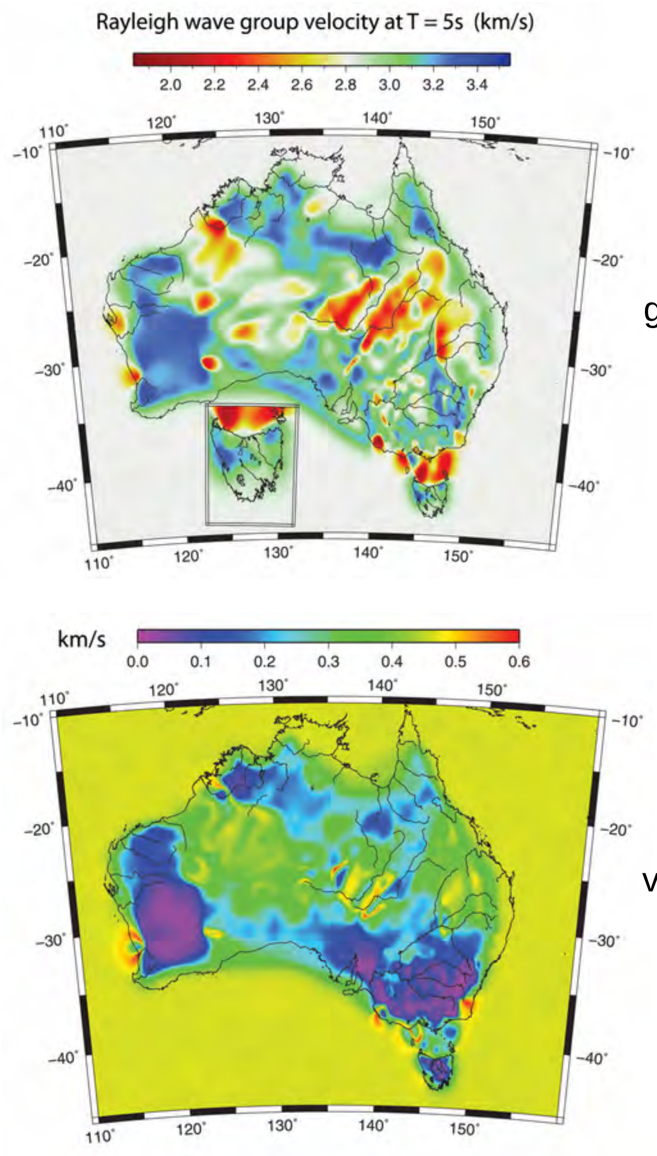
Ambient noise tomography in Australia from Bodin et al., 2012



Real world example I

Ambient noise tomography in Australia from Bodin et al., 2012



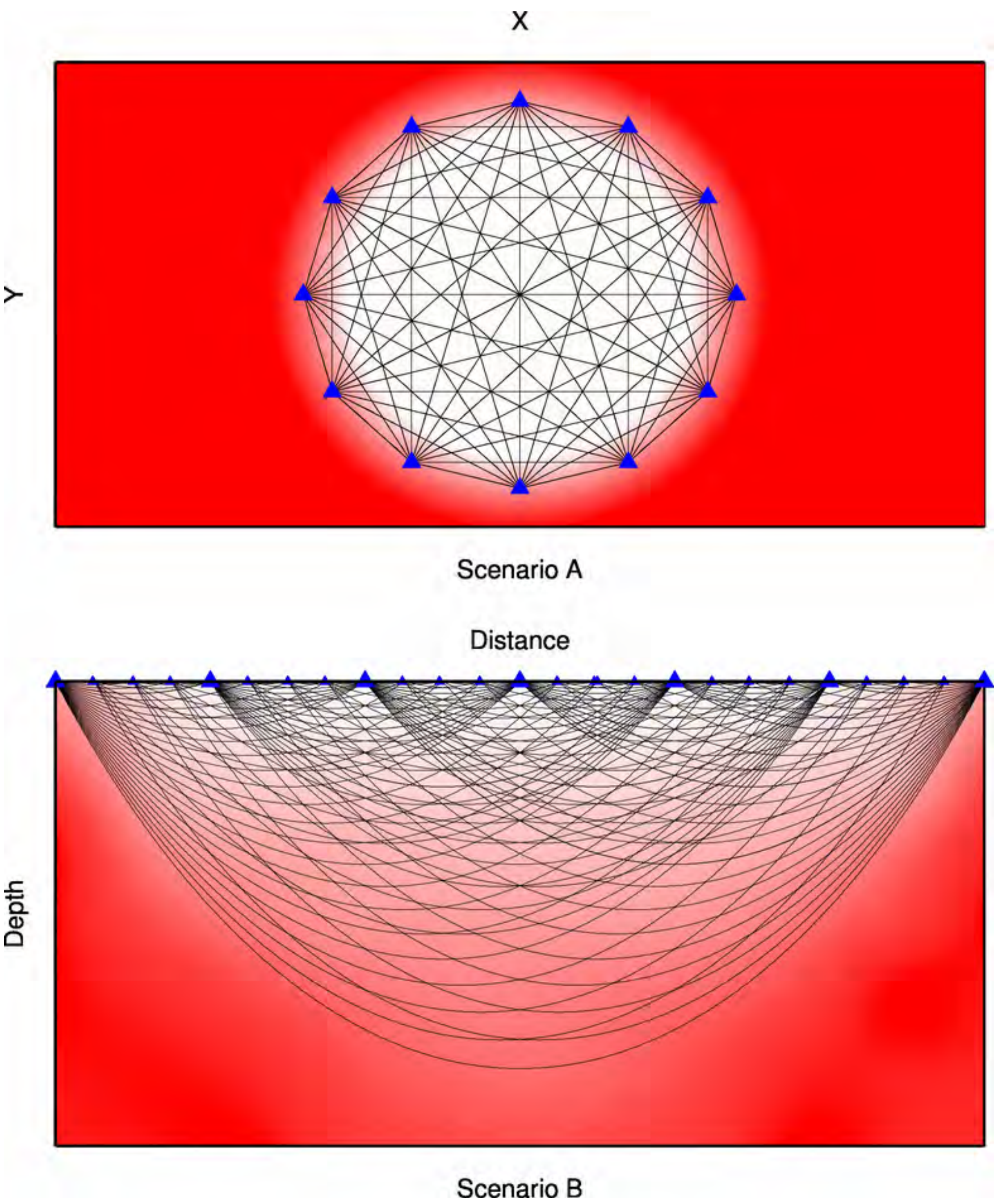


group velocity [km/s]

velocity error [km/s]

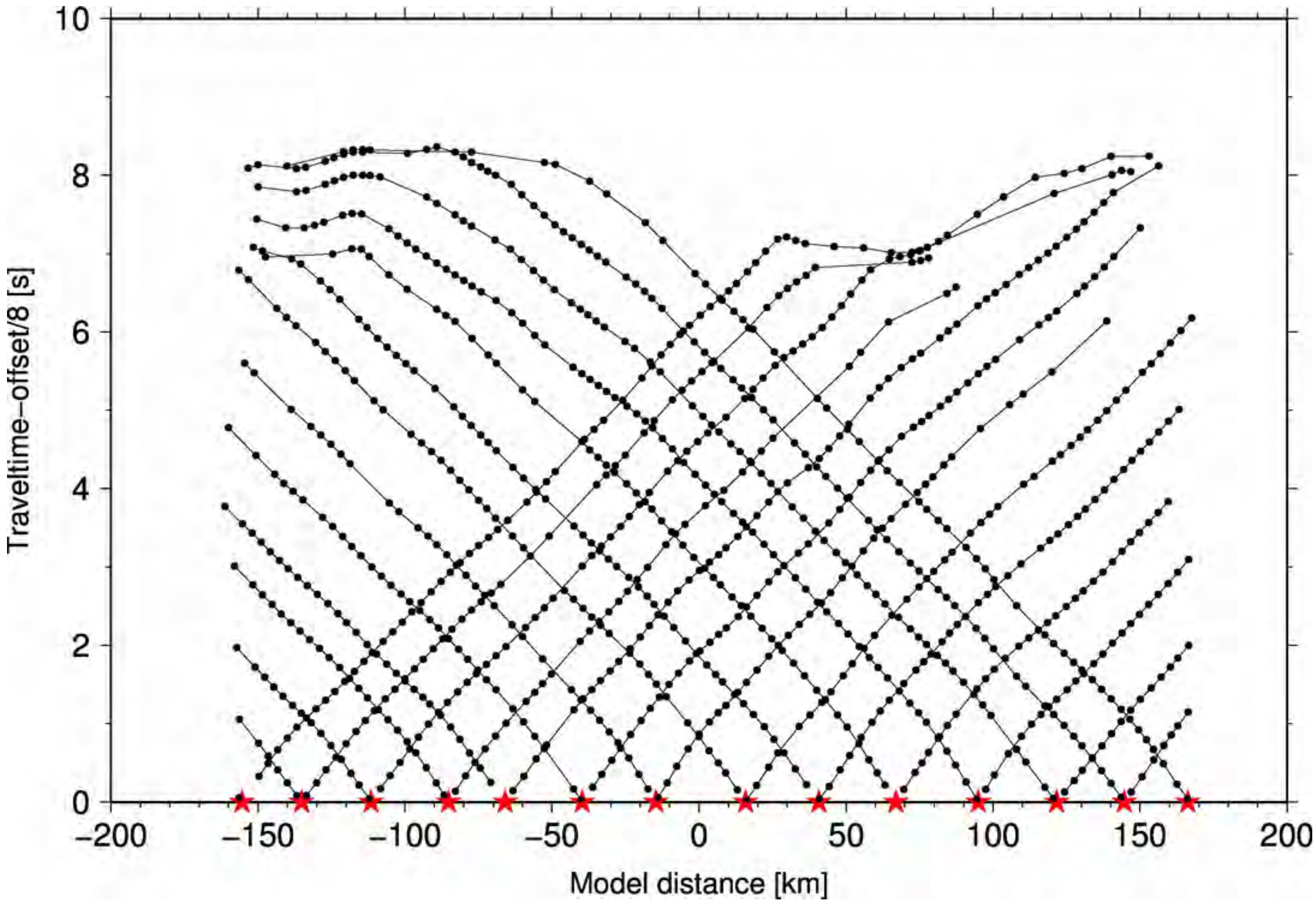
Real world example II

Controlled source seismic tomography in Namibia



Real world example II

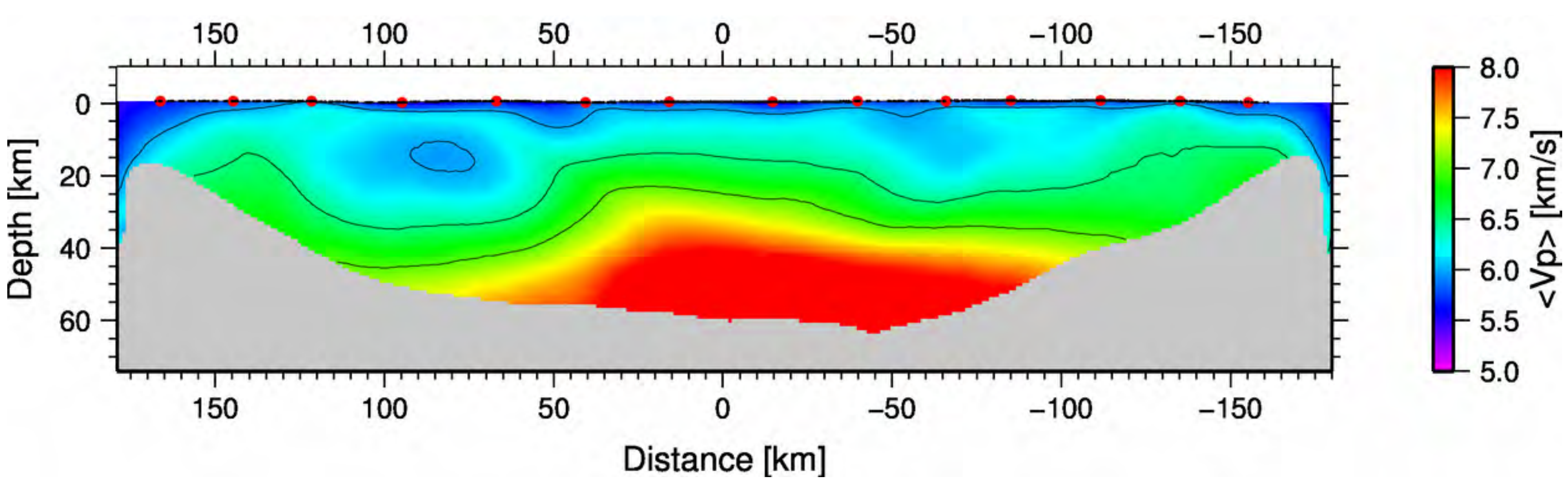
Controlled source seismic tomography in Namibia



travel time data

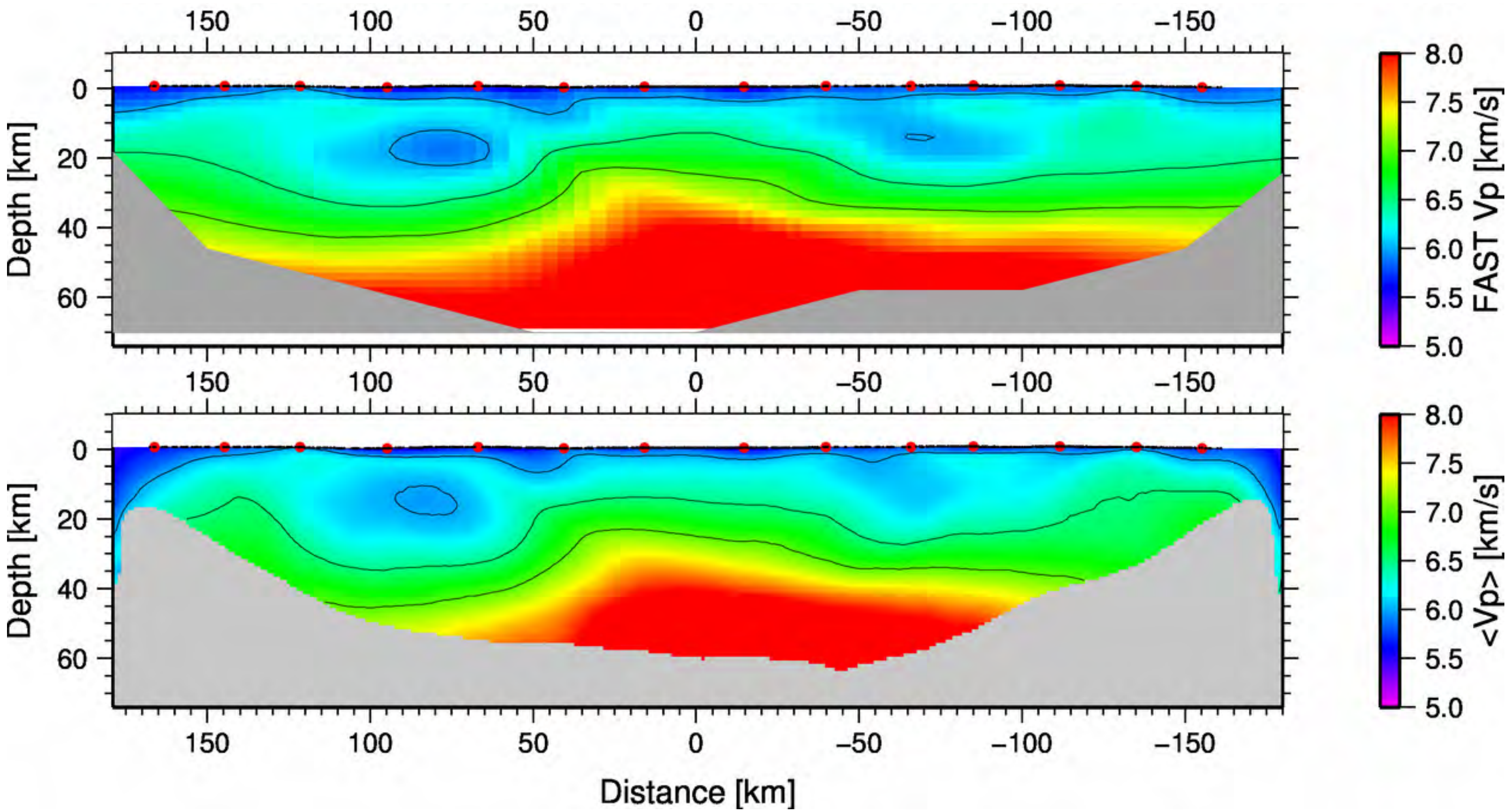
Real world example II

Controlled source seismic tomography in Namibia



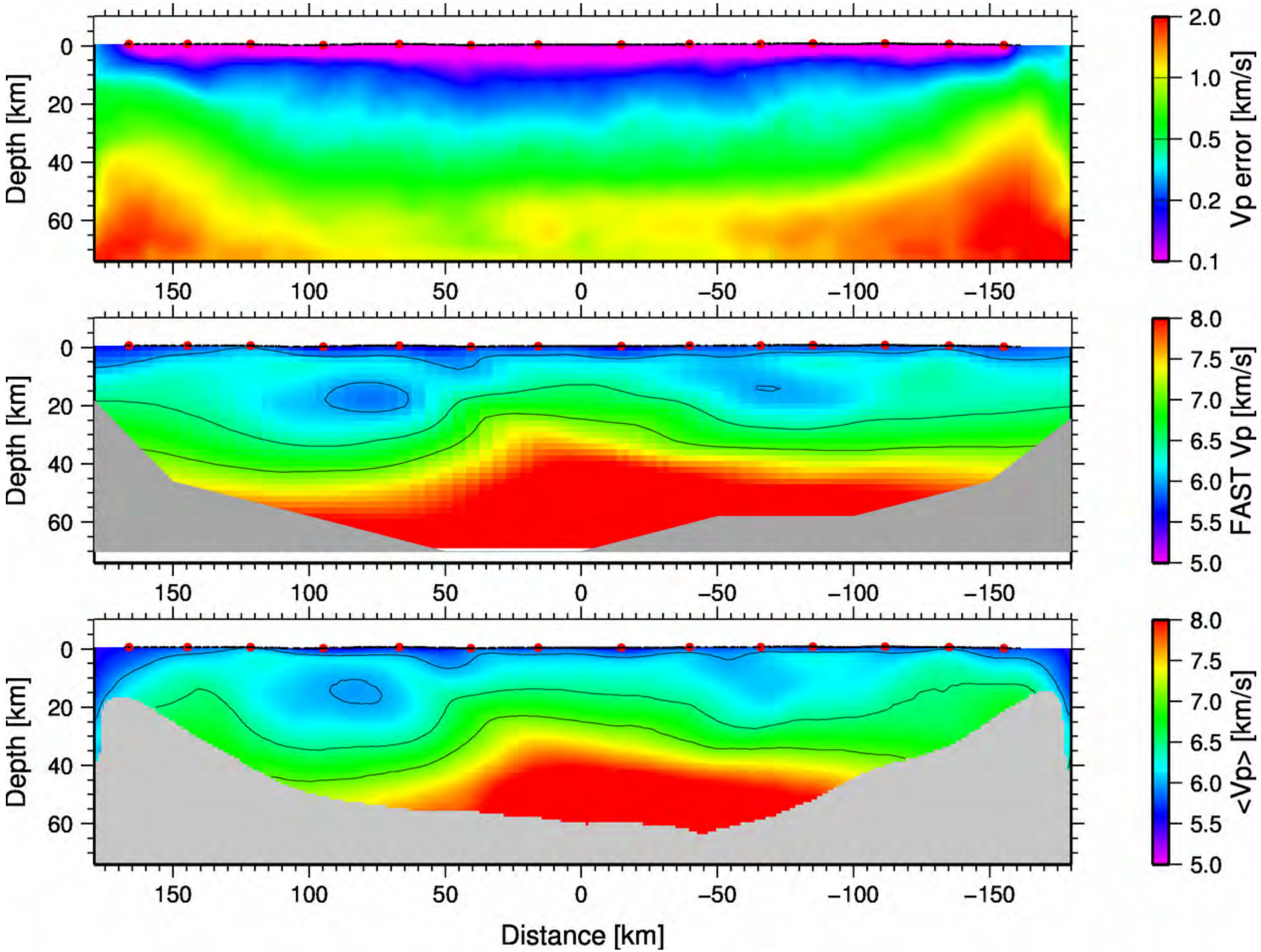
Real world example II

Controlled source seismic tomography in Namibia



Real world example II

Controlled source seismic tomography in Namibia



Summary

- Monte Carlo based inversions using Markov Chains can be used to thoroughly explore the model space
- Reference model: construction from large numbers of “good” fitting models, ensemble properties (wisdom of the cloud)
- Model uncertainties: error maps for model parameters can be derived by the posterior distribution (estimation of parameter uncertainties)
- Self-parameterizing: data itself is used to constrain the model parametrization
- Adaptive model parameterization: trans-dimensional based approaches, super-resolution, adaptation to actual model complexity driven by data
- Estimation of data noise (as part of the data not be explained by the model)

Summary

- Monte Carlo based inversions using Markov Chains can be used to thoroughly explore the model space

- Reference model: construction from large numbers of “good” fitting models, ensemble properties (wisdom of the cloud)

- Model uncertainties: error maps for model parameters can be derived by the posterior distribution (estimation of parameter uncertainties)

- Self-parameterizing: data itself is used to constrain the model parametrization

- Adaptive model parameterization: trans-dimensional based approaches, super-resolution, adaptation to actual model complexity driven by data

- Estimation of data noise (as part of the data not be explained by the model)

Summary

- Monte Carlo based inversions using Markov Chains can be used to thoroughly explore the model space
- Reference model: construction from large numbers of “good” fitting models, ensemble properties (wisdom of the cloud)
- Model uncertainties: error maps for model parameters can be derived by the posterior distribution (estimation of parameter uncertainties)
- Self-parameterizing: data itself is used to constrain the model parametrization
- Adaptive model parameterization: trans-dimensional based approaches, super-resolution, adaptation to actual model complexity driven by data
- Estimation of data noise (as part of the data not be explained by the model)

Summary

- Monte Carlo based inversions using Markov Chains can be used to thoroughly explore the model space
- Reference model: construction from large numbers of “good” fitting models, ensemble properties (wisdom of the cloud)
- Model uncertainties: error maps for model parameters can be derived by the posterior distribution (estimation of parameter uncertainties)
- Self-parameterizing: data itself is used to constrain the model parametrization
- Adaptive model parameterization: trans-dimensional based approaches, super-resolution, adaptation to actual model complexity driven by data
- Estimation of data noise (as part of the data not be explained by the model)

Summary

- Monte Carlo based inversions using Markov Chains can be used to thoroughly explore the model space
- Reference model: construction from large numbers of “good” fitting models, ensemble properties (wisdom of the cloud)
- Model uncertainties: error maps for model parameters can be derived by the posterior distribution (estimation of parameter uncertainties)
- Self-parameterizing: data itself is used to constrain the model parametrization
- Adaptive model parameterization: trans-dimensional based approaches, super-resolution, adaptation to actual model complexity driven by data
- Estimation of data noise (as part of the data not be explained by the model)

Summary

- Monte Carlo based inversions using Markov Chains can be used to thoroughly explore the model space
- Reference model: construction from large numbers of “good” fitting models, ensemble properties (wisdom of the cloud)
- Model uncertainties: error maps for model parameters can be derived by the posterior distribution (estimation of parameter uncertainties)
- Self-parameterizing: data itself is used to constrain the model parametrization
- Adaptive model parameterization: trans-dimensional based approaches, super-resolution, adaptation to actual model complexity driven by data
- Estimation of data noise (as part of the data not be explained by the model)

Summary ...

- Suitable for inversion of multi-scale data sets
- Suitable for joint inversion of different data sets (gravity-velocities, resistivity, etc.)
- Works in 1D, 2D, 3D, …
- Almost no prior knowledge needed: grid-size, starting model, smoothing, damping, …
- Disadvantage: computationally VERY expensive (orders of magnitudes higher compared to conventional methods)

Summary ...

- Suitable for inversion of multi-scale data sets
- Suitable for joint inversion of different data sets (gravity-velocities, resistivity, etc.)
- Works in 1D, 2D, 3D, …
- Almost no prior knowledge needed: grid-size, starting model, smoothing, damping, …
- Disadvantage: computationally VERY expensive (orders of magnitudes higher compared to conventional methods)

Summary ...

- Suitable for inversion of multi-scale data sets
- Suitable for joint inversion of different data sets (gravity-velocities, resistivity, etc.)
- Works in 1D, 2D, 3D, …
- Almost no prior knowledge needed: grid-size, starting model, smoothing, damping, …
- Disadvantage: computationally VERY expensive (orders of magnitudes higher compared to conventional methods)

Summary ...

- Suitable for inversion of multi-scale data sets
- Suitable for joint inversion of different data sets (gravity-velocities, resistivity, etc.)
- Works in 1D, 2D, 3D, …
- Almost no prior knowledge needed: grid-size, starting model, smoothing, damping, …
- Disadvantage: computationally VERY expensive (orders of magnitudes higher compared to conventional methods)

Summary ...

- Suitable for inversion of multi-scale data sets
- Suitable for joint inversion of different data sets (gravity-velocities, resistivity, etc.)
- Works in 1D, 2D, 3D, …
- Almost no prior knowledge needed: grid-size, starting model, smoothing, damping, …
- Disadvantage: computationally VERY expensive (orders of magnitudes higher compared to conventional methods)

References

Introduction into Monte Carlo methods:

Sambridge & Mosegaard: Monte Carlo methods in geophysical inverse problems, Reviews in Geophysics, 2002

Introduction into Bayesian inversion methods (general, not limited to tomography):

PhD thesis Th. Bodin http://perso.ens-lyon.fr/thomas.bodin/publi.html

Transdimensional tomography:

Bodin & Sambridge: Seismic tomography with reversible jump algorithm, GJI, 2009

Hierarchical transdimensional tomography

Bodin et al., Transdimensional tomography with unknown data noise, GJI, 2012