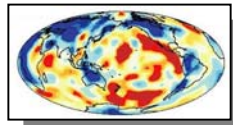
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo Methods

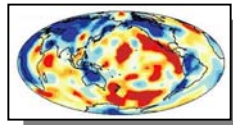


- What is a Monte Carlo Method?
- Random walks
- The Metropolis rule – importance sampling
- Near neighbor sampling
- Sampling prior and posterior probability
- Example: gravity inversion
- The ***movie*** philosophy

This lecture follows mainly:
Modesgaard and Tarantola, J. Geophys. Res., 100, B7, 12,431-12447, 1995 and
Modesgaard and Tarantola, Probabilitstic approach to inverse problems, in International Handbook of Earthquake and Engineering Seismology, 2001, Academic Press.

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

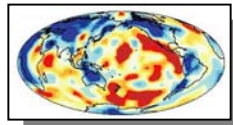
Monte Carlo Methods



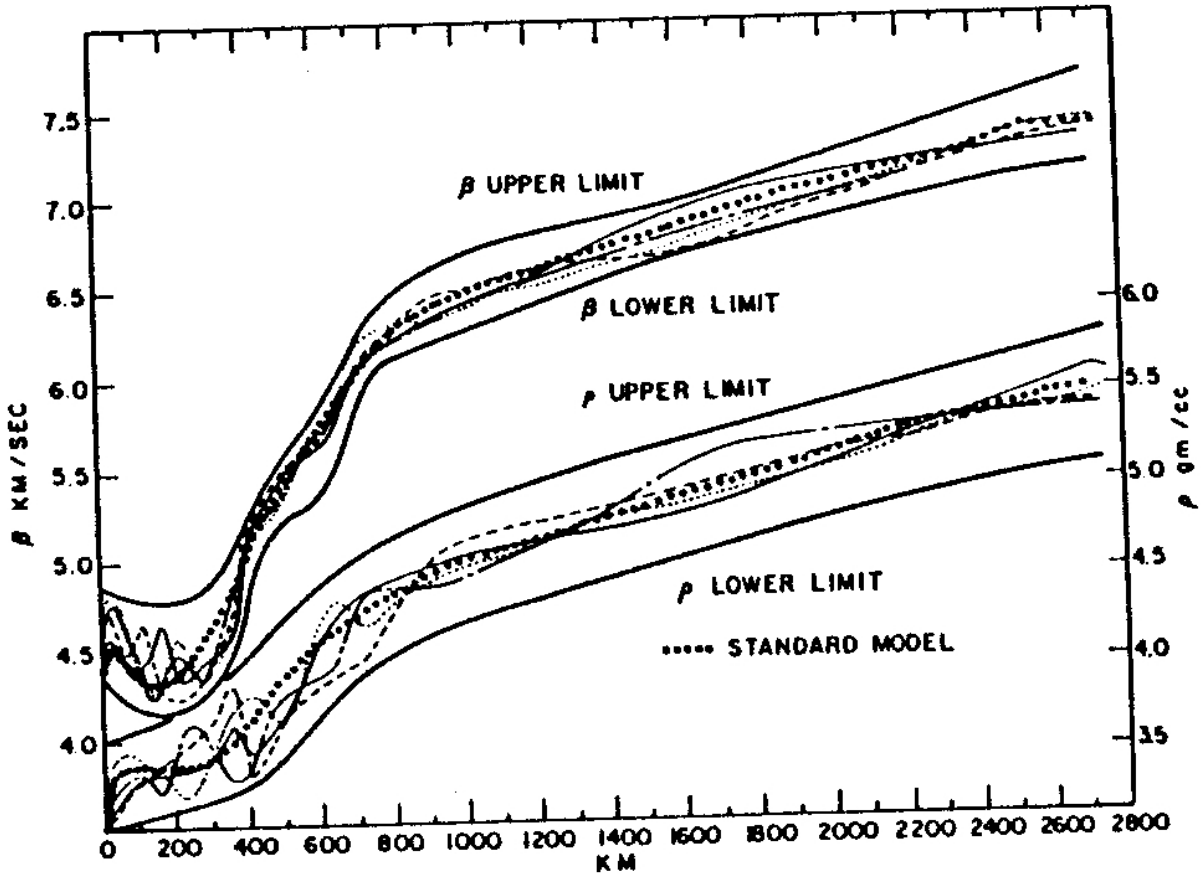
Monte Carlo methods are named after the city in the Monaco principality, because of the roulettes in the Casinos (random number generator)

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo Methods

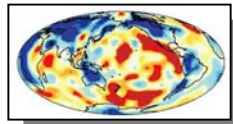


Early applications of Monte Carlo methods to the determination of Earth's structure (Press 1986).



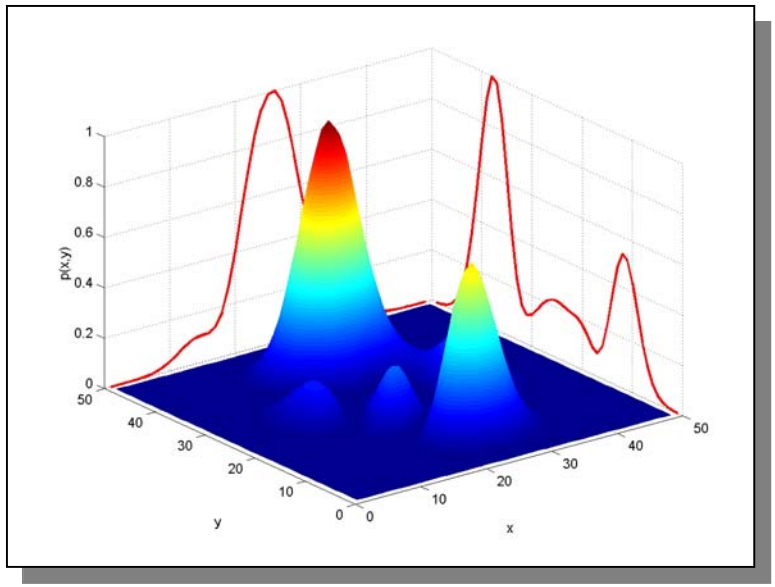
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

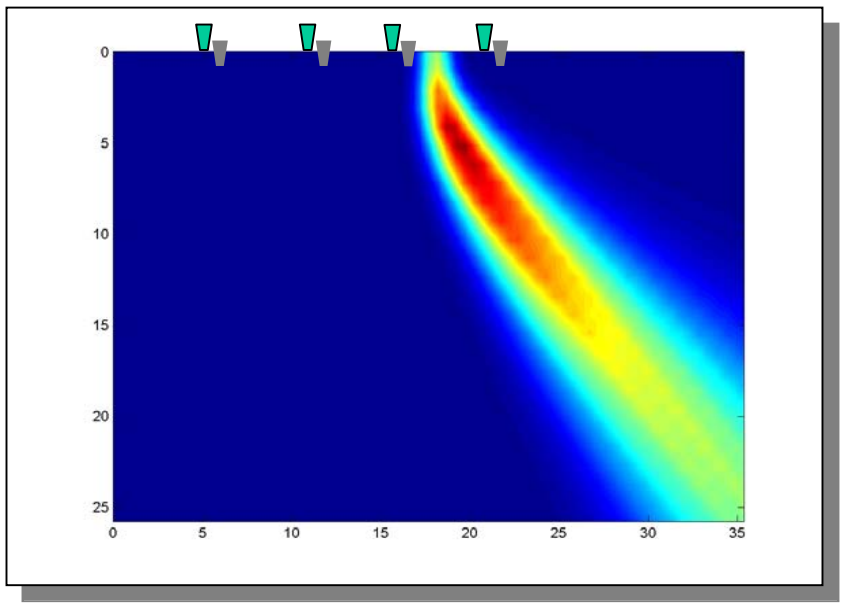
Monte Carlo Methods



The goal:

How can we efficiently **sample** the a posteriori pdf …

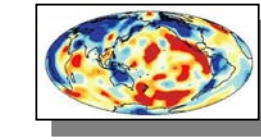


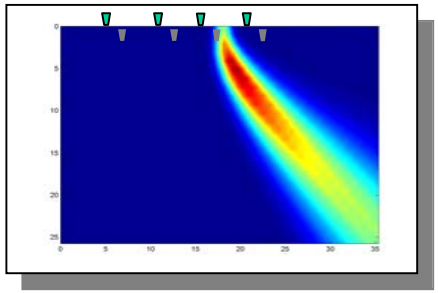


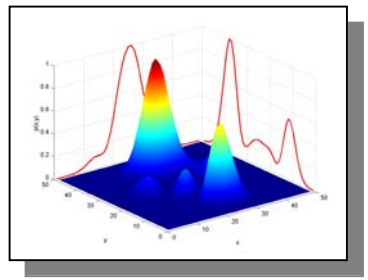
… knowing that the computation of the forward problem is expensive and computational powers is finite!

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo Methods





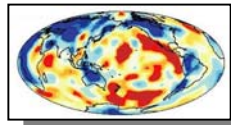


More technically:

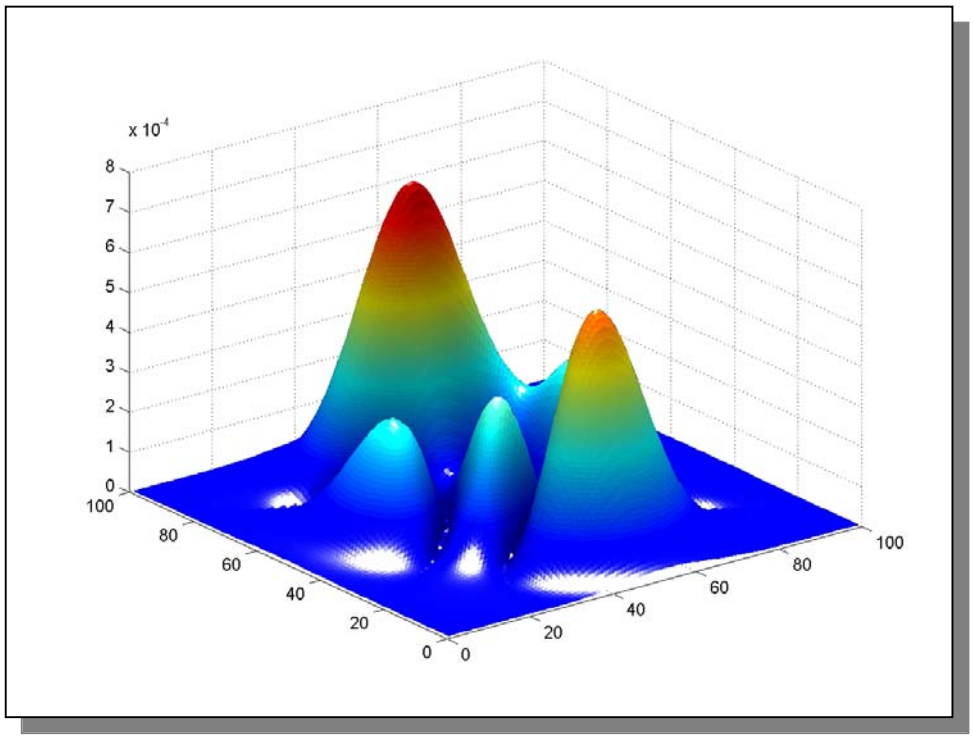
Given a set of points in space with a probability p_i attached to every point i, how can we define random rules to select points such that the probability of selecting points is $p_{i?}$

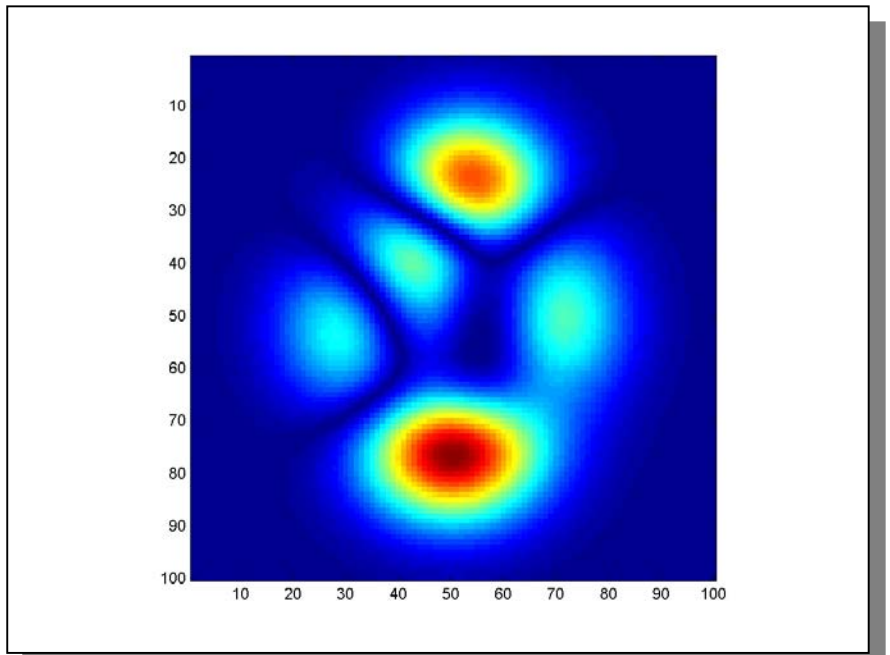
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo Methods



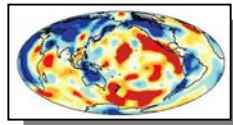
Let us use the peaks function of Matlab® to illustrate Monte Carlo techniques. The following two-dimensional function is supposed to be an a posteriori probability density function. Each point of this graph would require at least the calculation of a (possibly very expensive) forward problem and a comparison with data (misfit).





$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo Methods

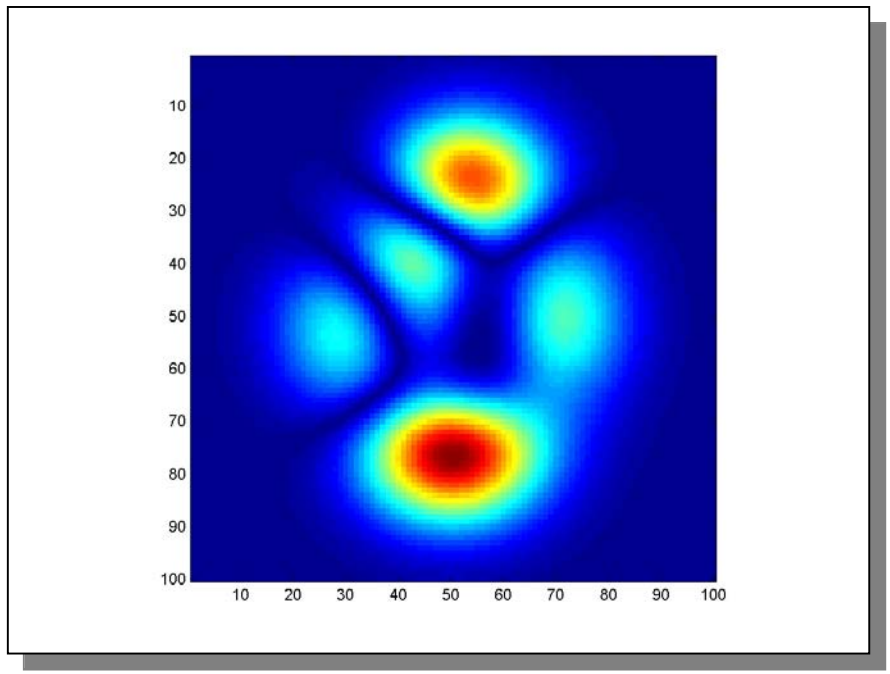


Note that for multi-dimensional problems a point in model space may represent a complex Earth model. By sampling the a posteriori pdf we collect models hopefully of good quality (small misfit between synthetics and real data). Eventually our eye may decide which of the models are realistic.

To avoid sampling the areas of low probability we introduce the concept of importance sampling.

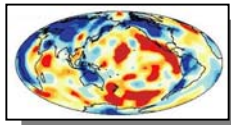
What is a random walk?

We successively visit points in the model space where the next point X_{i+1} to be visited depends on the point x_i . How can we choose points so that we sample the pdf?



$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Random walks



The most common Monte Carlo sampling methods is the Metropolis sampler:

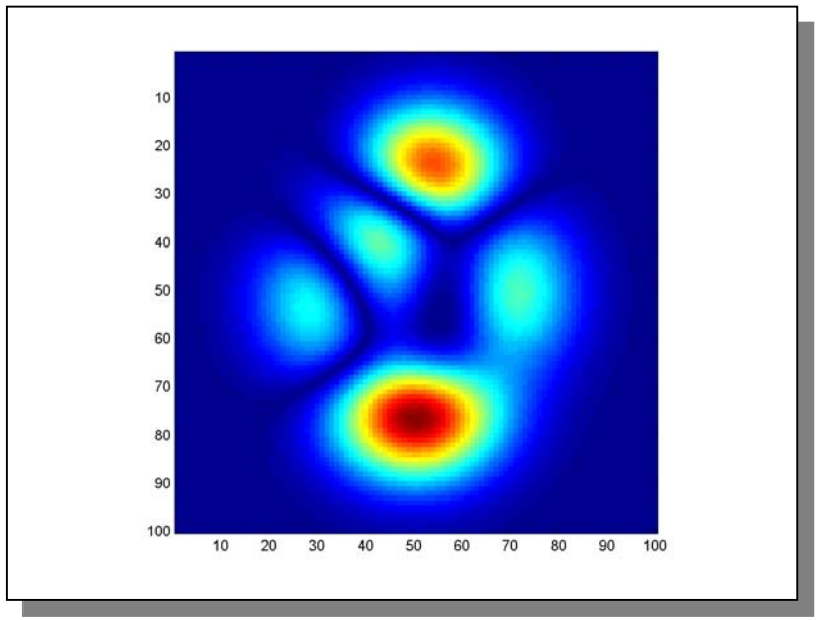
At a given point the random walker is at at point x_i and now we have to define rules how to get to another point x_j. If we accept any random move the walker would sample – no doubt – at some point the whole space. Instead of always accepting the transition, we reject the move sometimes:

Let f(x) be the probability density function:

- if $f(x_j) \geq f(x_i)$ -> accept the move
- if $f(x_j) < f(x_i)$ -> then decide randomly to move to x_j with the probability of accepting the move:

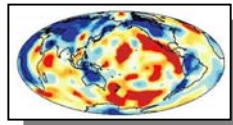
$$P = \frac{f(x_j)}{f(x_i)}$$

P is the transition probability

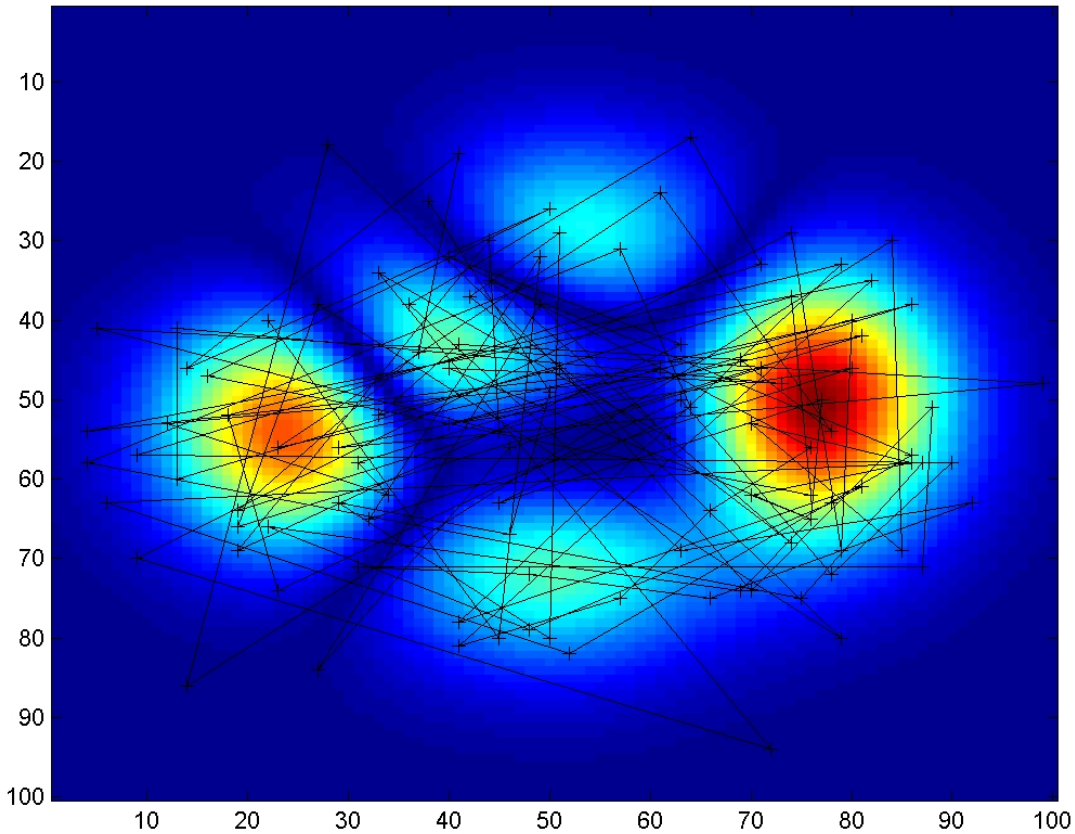


$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Random walks - Metropolis

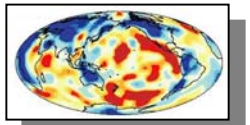


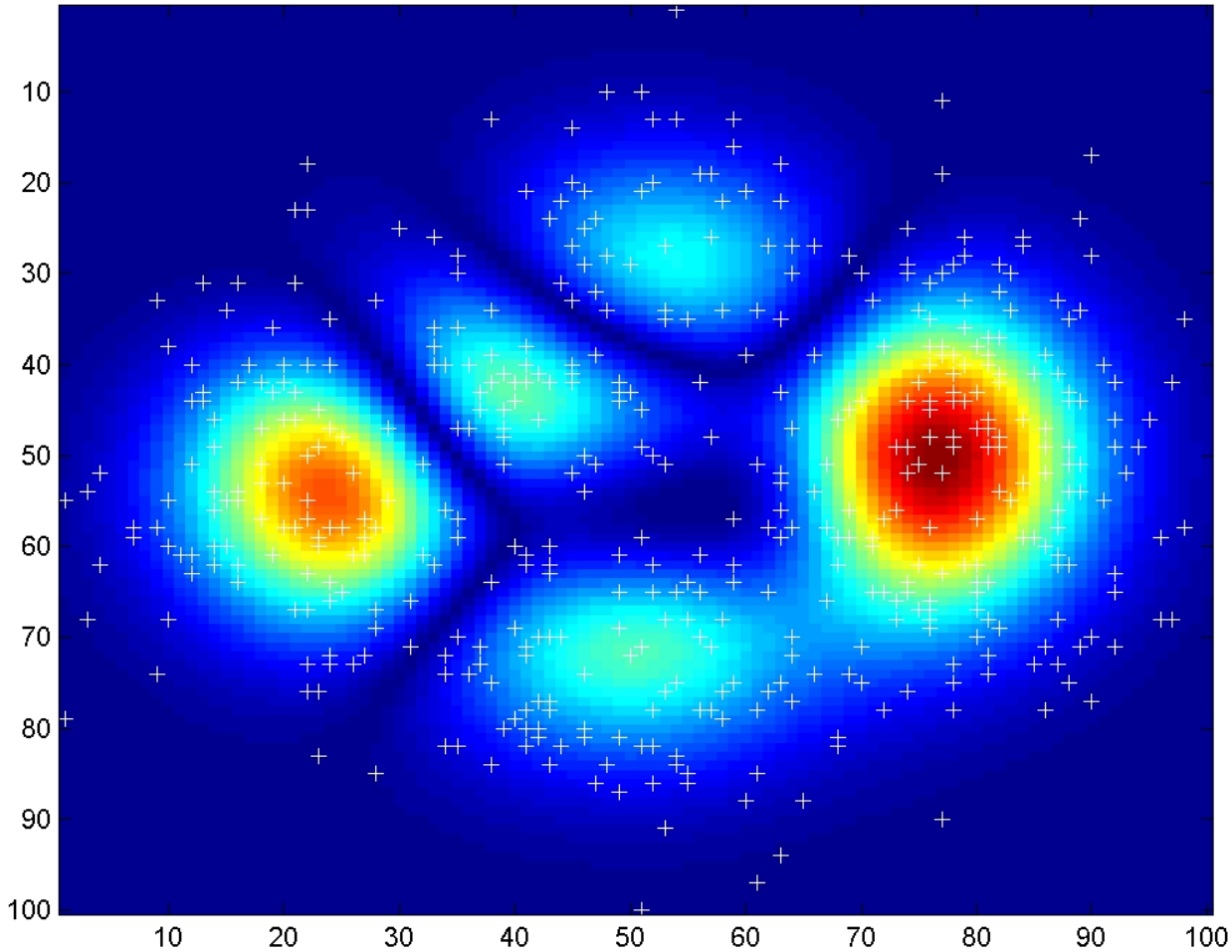
We obtain the following results for our test probability function:



$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

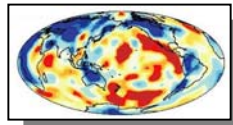
Random walks - Metropolis



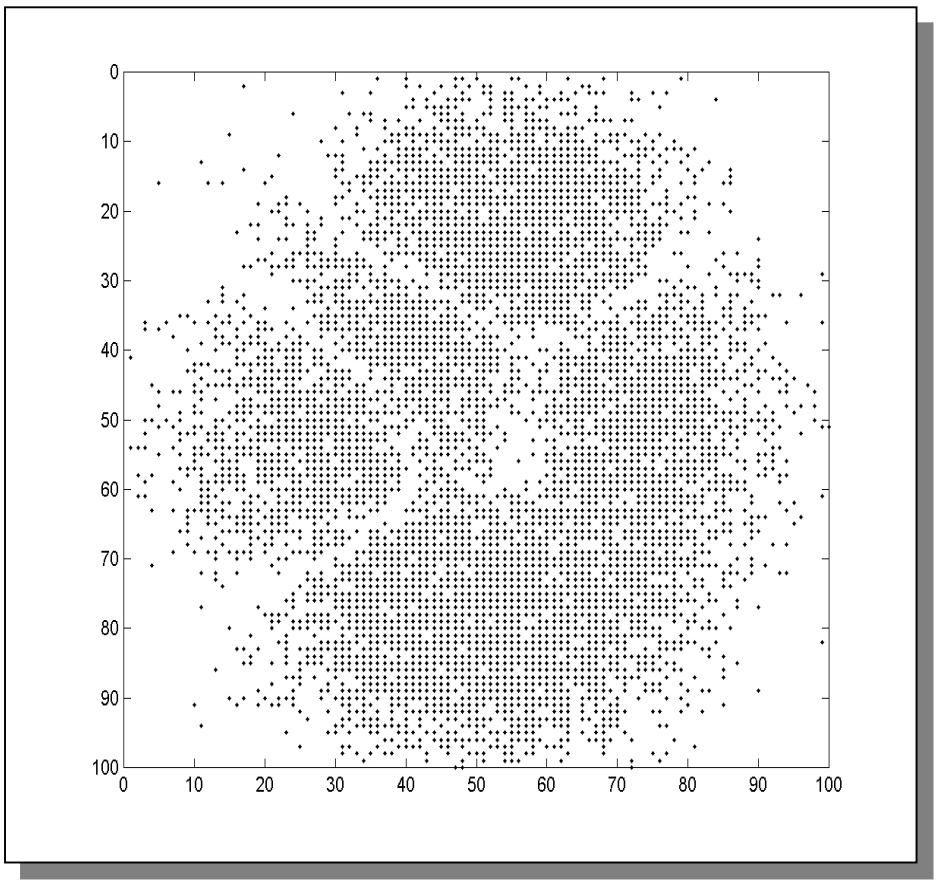


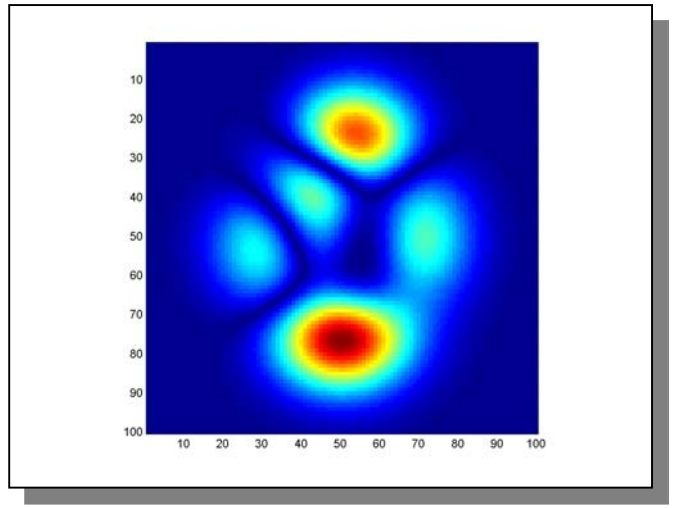
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Random walks - Metropolis



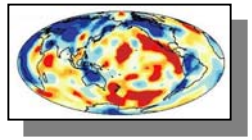
The ten thousand points visited seem to represent well the areas where something's going on



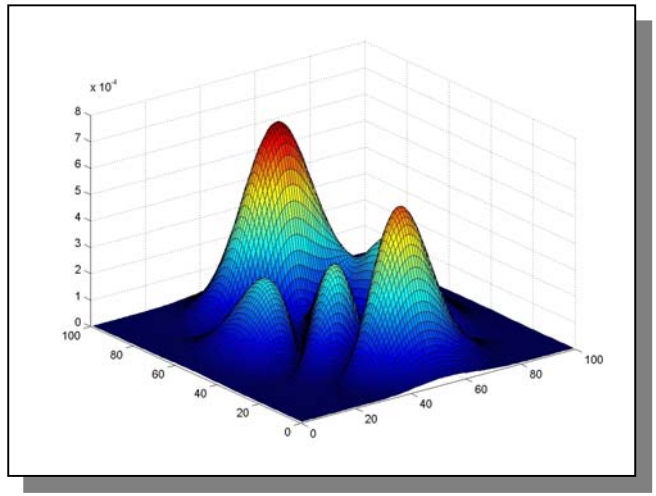


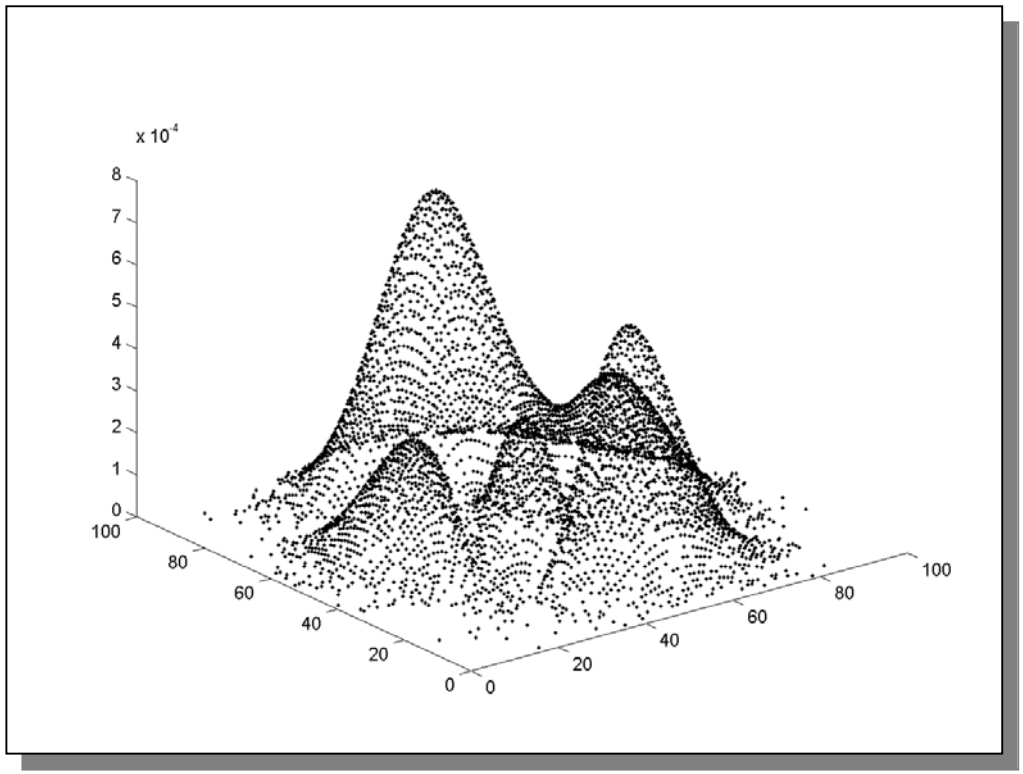
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Random walks - Metropolis



… and the function is now well represented ..
but still at a high cost … only slightly less expensive than the crude Monte Carlo approach …

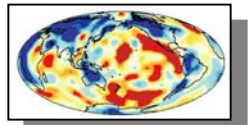




These results were obtained with the Matlab program **mc_metro.m**

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Random walks - Metropolis

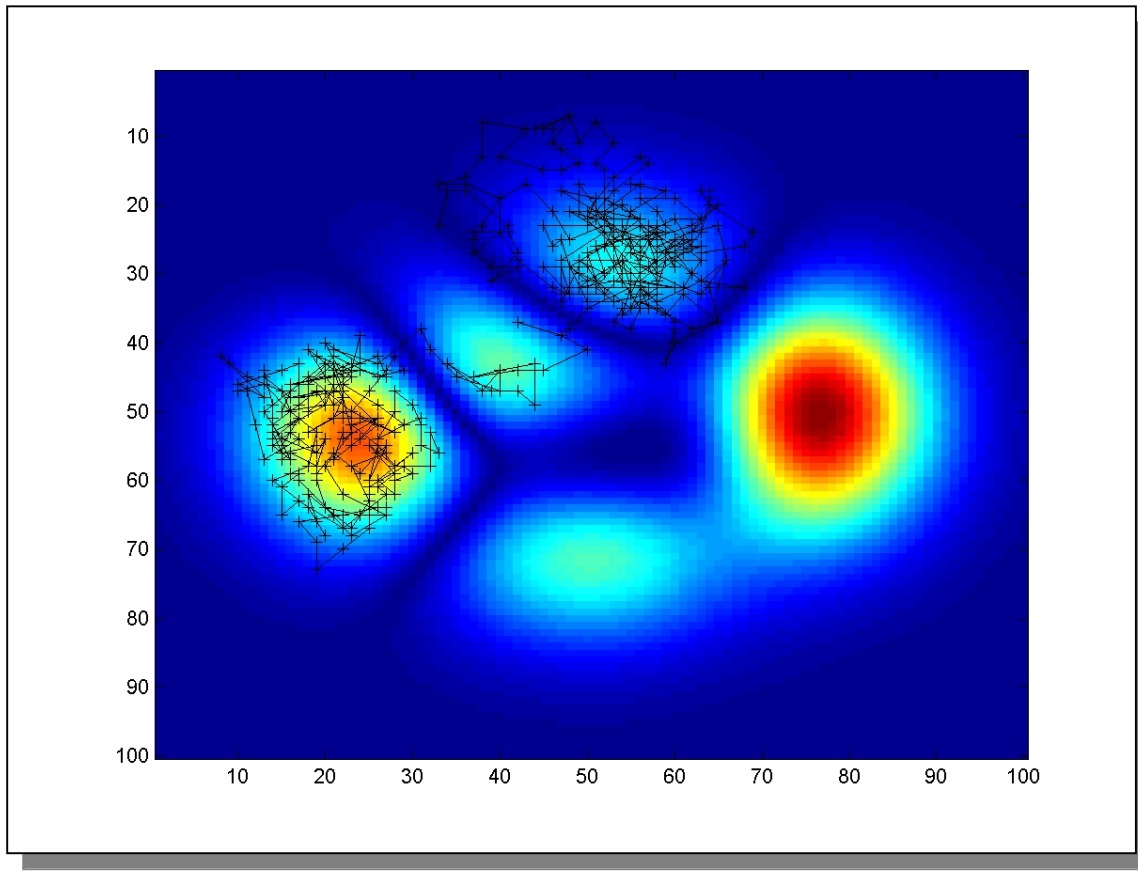


First modifications:

Limit the algorithm to look in the neighborhood of the present point. This is called **near neighbor sampling**.

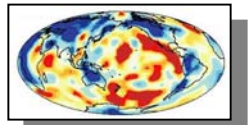
Here we allowed the walker to move only within 10% of the total size of the model Space.

The program used was mc_neigh.m and the relevant Parameter neigh=0.1.



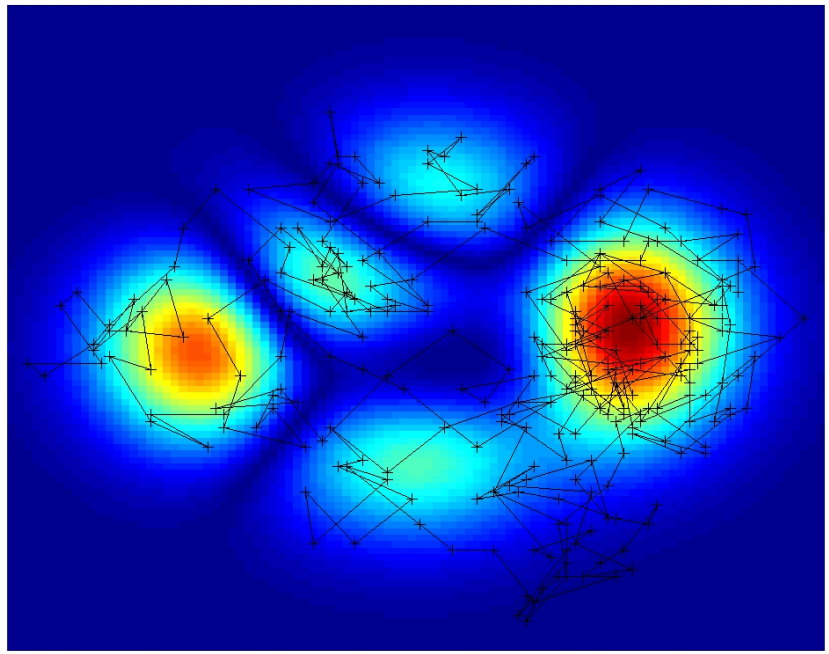
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Random walks - Metropolis



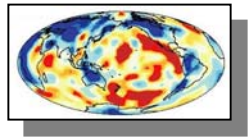
Near neighbor sampling.

The program used was mc_neigh.m and the relevant Parameter neigh=0.2.



$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Random walks - Convergence



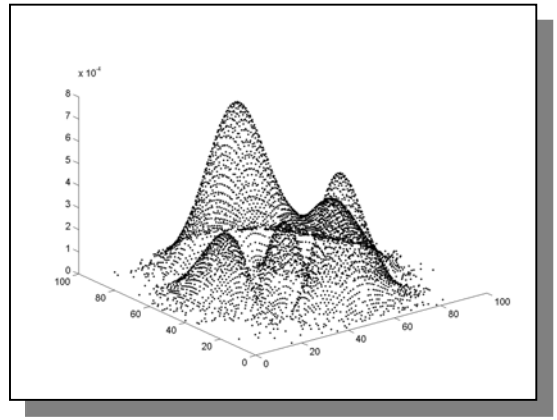
There is a crucial question:

How many points do we have to visit until we have a good idea about the solution to our problem?

This is an extremely difficult question to answer because:

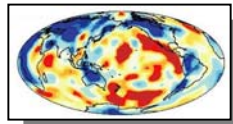
- the behavior of a random walk algorithm may seem straight forward in 2D or 3D but may behave very differently in systems with many dimensions
- the reason why you use Monte Carlo Methods is because you don't know what the function looks like. So how can you be sure you are sampling fine enough to get the good areas of the model space?

…. some of these problems lead us to improved techniques such as simulated annealing …



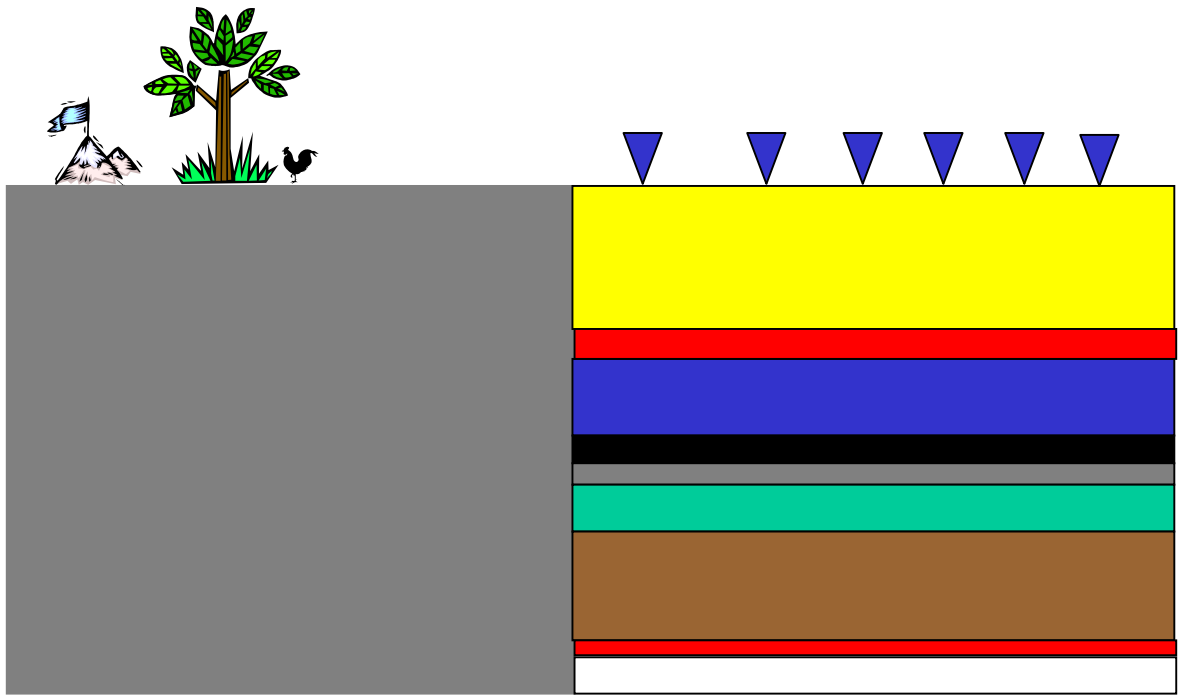
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo method: gravity

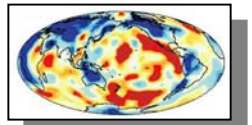


Inversion of gravity data: a classical test for all theories of inversion!

The problem: Find the depth-dependent density structure to the right of the vertical fault. The observations are horizontal gradients of the gravity to the right of the fault.

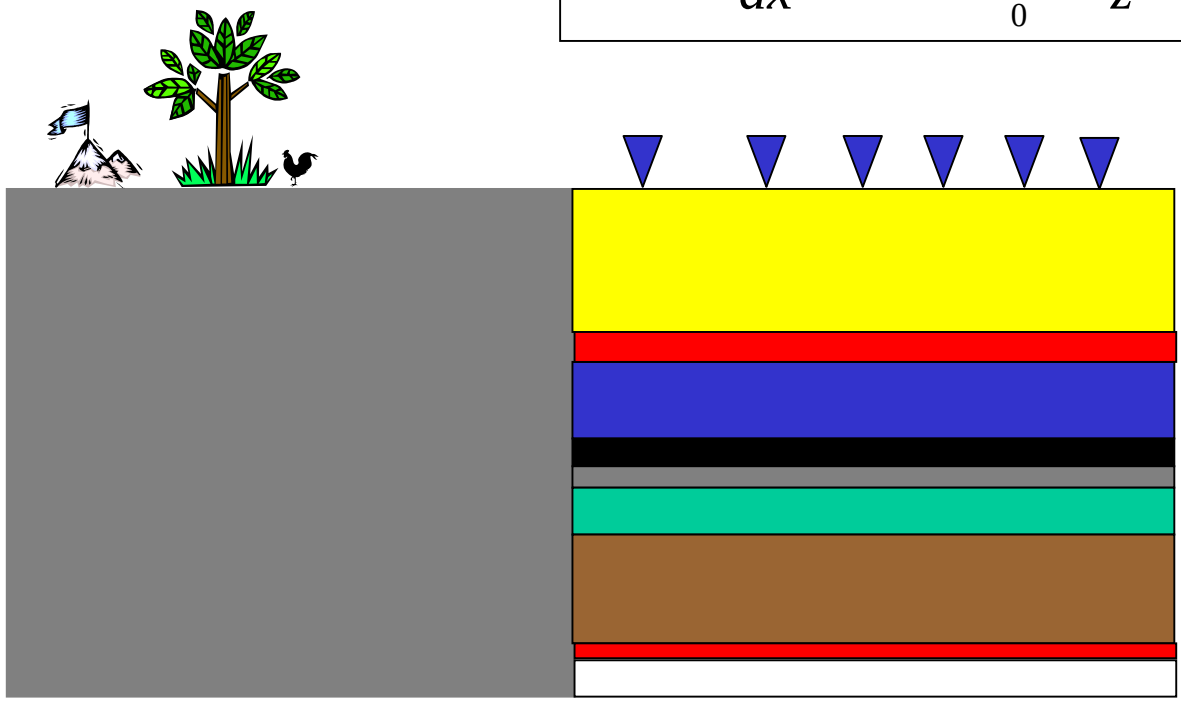


Monte Carlo method: gravity



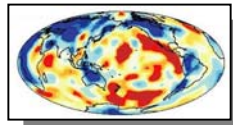
The forward problem: The gravity gradient at the surface is given by:

$$d(x) = \frac{dg}{dx}(x) = 2G\int_0^\infty dz\,\frac{z\Delta\rho(z)}{z^2 + x^2}$$



$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo sampling of prior information



Let us now walk through this inverse problem and make use of the Monte Carlo ideas:

For any particular forward problem **the first step** would be

1. Sampling the a priori probability

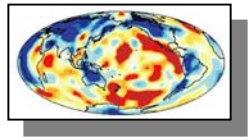
   - we need a pseudorandom process (our Monte Carlo approach) to find samples of the prior information
   - We know the a priori pdf analytically

Example: From observations in a well (or in many many wells) we have found that in stratified media the distribution of layer thickness is approximately an exponential distribution and the densities have log normal distributions. We sample this prior information by:

- Select a layer uniformly at random
- Choose a new value for the layer thickness according to the exponential distribution
- Choose a value for the mass density according to the log-normal distribution

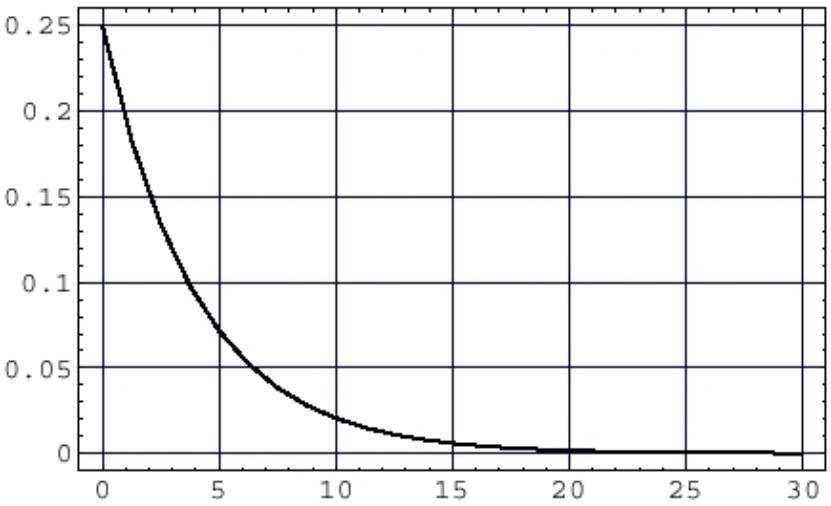
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo sampling of prior information

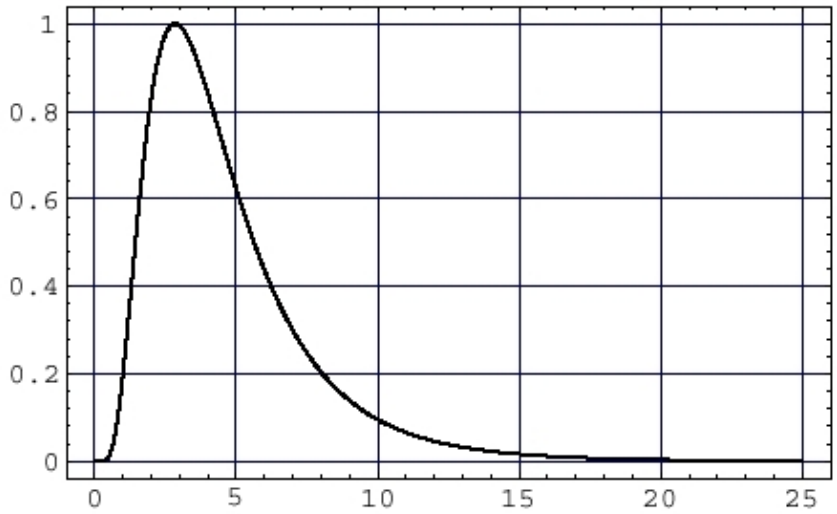


Let us look at the outcome of this process: **the prior probabilites**

Layer tickness (km). l_0=4km



Density (g/cm^3). ρ_0=3.98g/cm^3, σ=.58

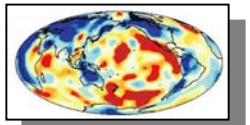


$$f(l) = \frac{1}{l_0}\exp\left(\frac{l}{l_0}\right)$$

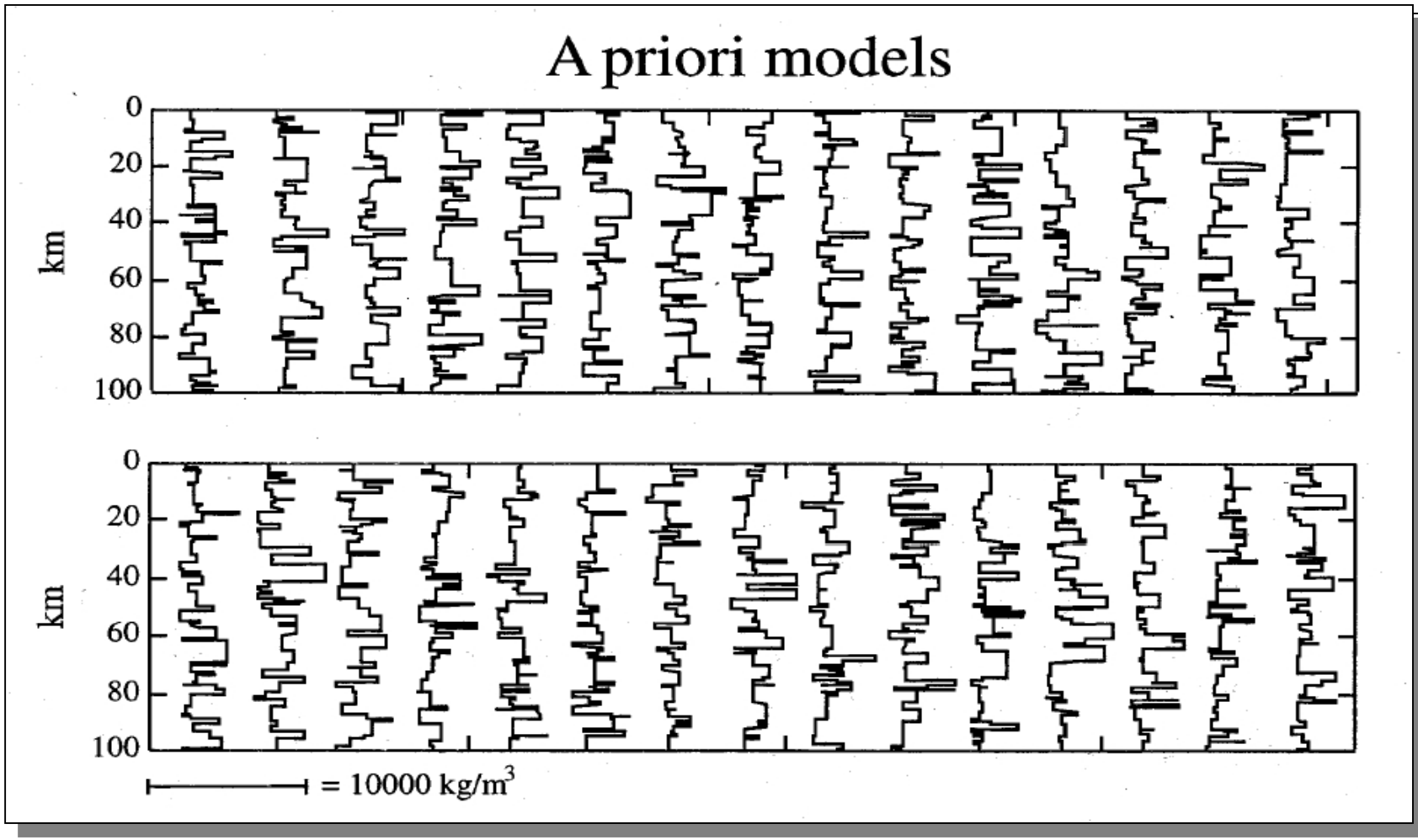
$$g(\rho) \propto \frac{1}{\rho}\exp\left(-\frac{1}{2\sigma^2}\left(\log\frac{\rho}{\rho_0}\right)^2\right)$$

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo sampling of prior information

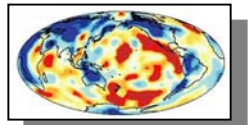


… a **random walk** through the prior probabilities produce models that look like this:

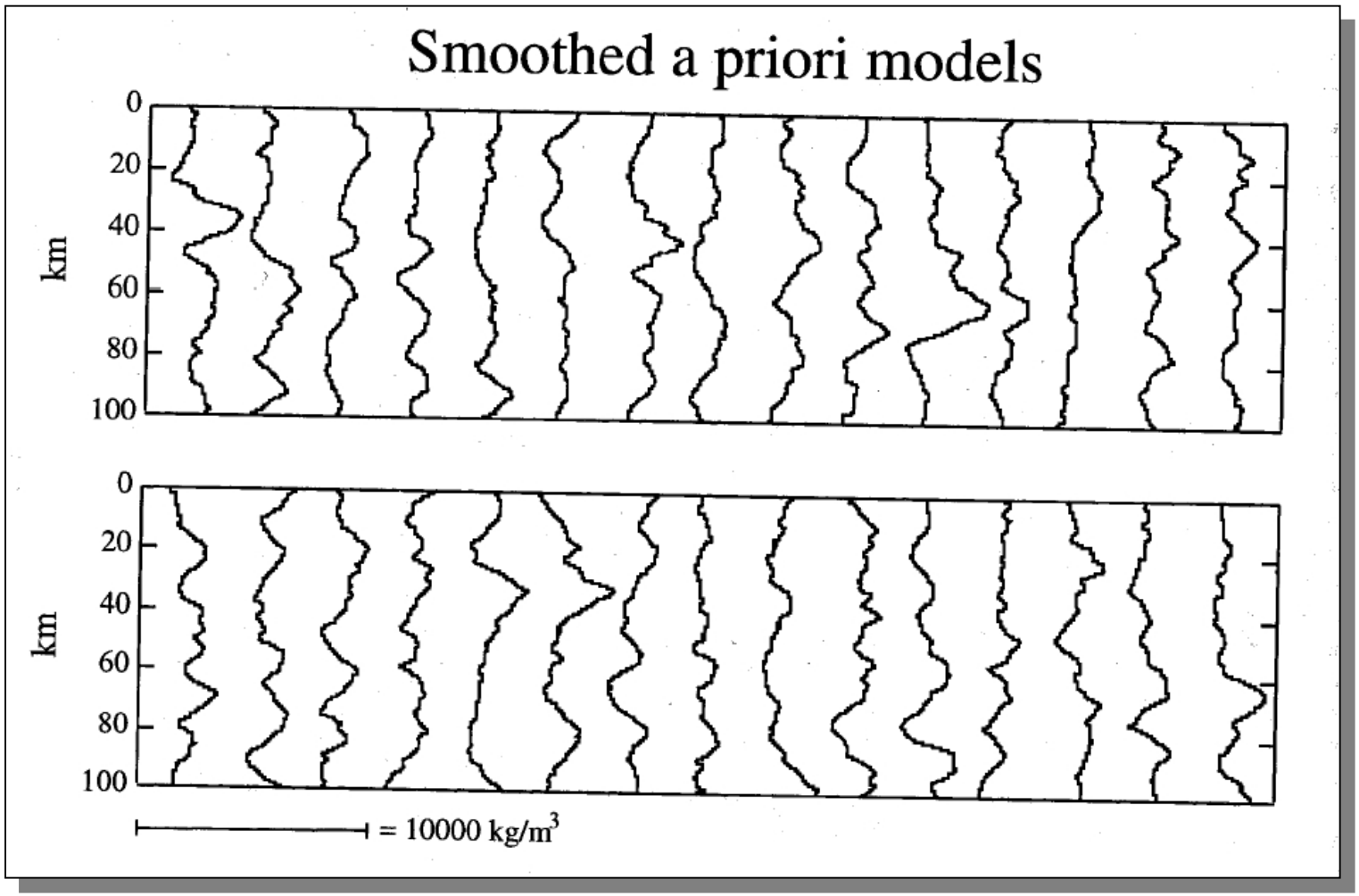


$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Monte Carlo sampling of prior information

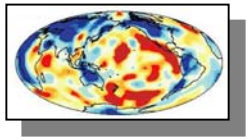


... we do not expect that the fine layering is well resolved, which is why it makes sense to look a smoothed models …



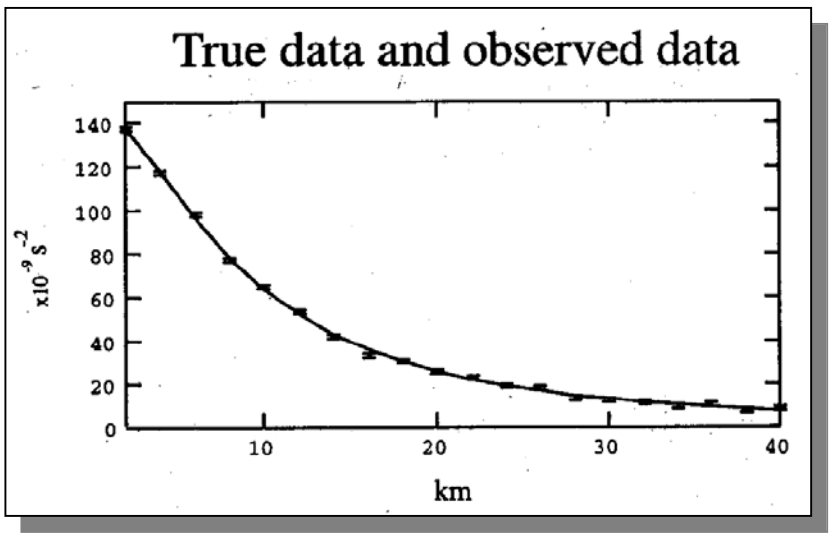
$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

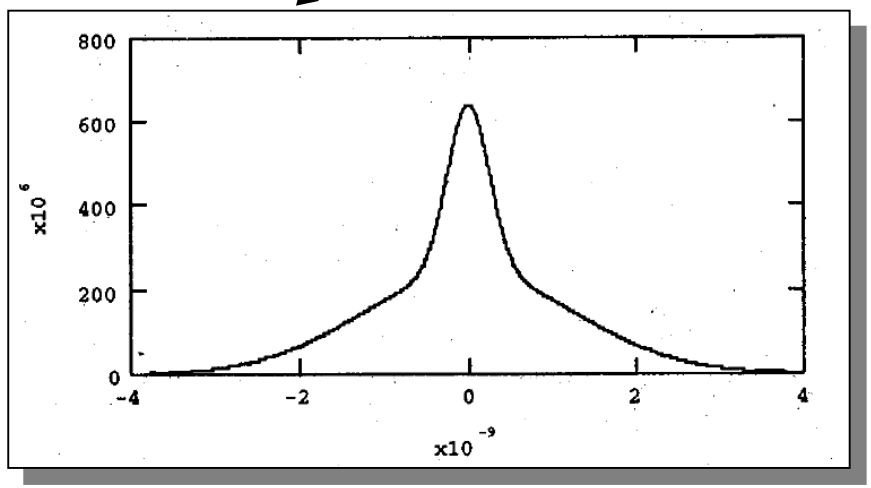
Gravity: experimental uncertainties



The measued data are assumed to be contaminated by random, uncorrelated noise. To make it a little more complicated, the errors are assumed to come from two processes with difference variances σ_i and relative probabilities (expressed through a):

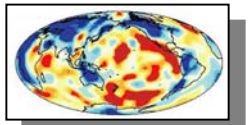
$$f(\varepsilon) = \frac{a}{\sqrt{2\pi}\sigma_1}\exp\left(-\frac{\varepsilon^2}{2\sigma_1^2}\right) + \frac{1-a}{\sqrt{2\pi}\sigma_2}\exp\left(-\frac{\varepsilon^2}{2\sigma_2^2}\right)$$





$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Gravity: posterior random walk



To calculate the forward problem the we need to sum over all layers:

$$d(x_j) = \sum_i^{nl} \Delta\rho_i \log\left(\frac{D_i^2 + x_j^2}{d_i^2 + x_j^2}\right)$$

… this is the horizontal gravity gradient as a function of the layers with D being the bottom depth of layer i, Δd the density cotnrast across the fault and x the horizontal distanc e of gravimeter j. The likelihood function L(m) can now be calculated according to:

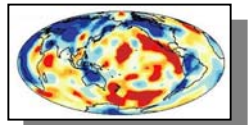
$$L(m) = k\prod_i \frac{a}{\sqrt{2\pi}\sigma_1}\exp\left(-\frac{(g_i(m) - d_i^{obs})^2}{2\sigma_1^2}\right) + \frac{1-a}{\sqrt{2\pi}\sigma_2}\exp\left(-\frac{(g_i(m) - d_i^{obs})^2}{2\sigma_2^2}\right)$$

With out double errors the solution would have been

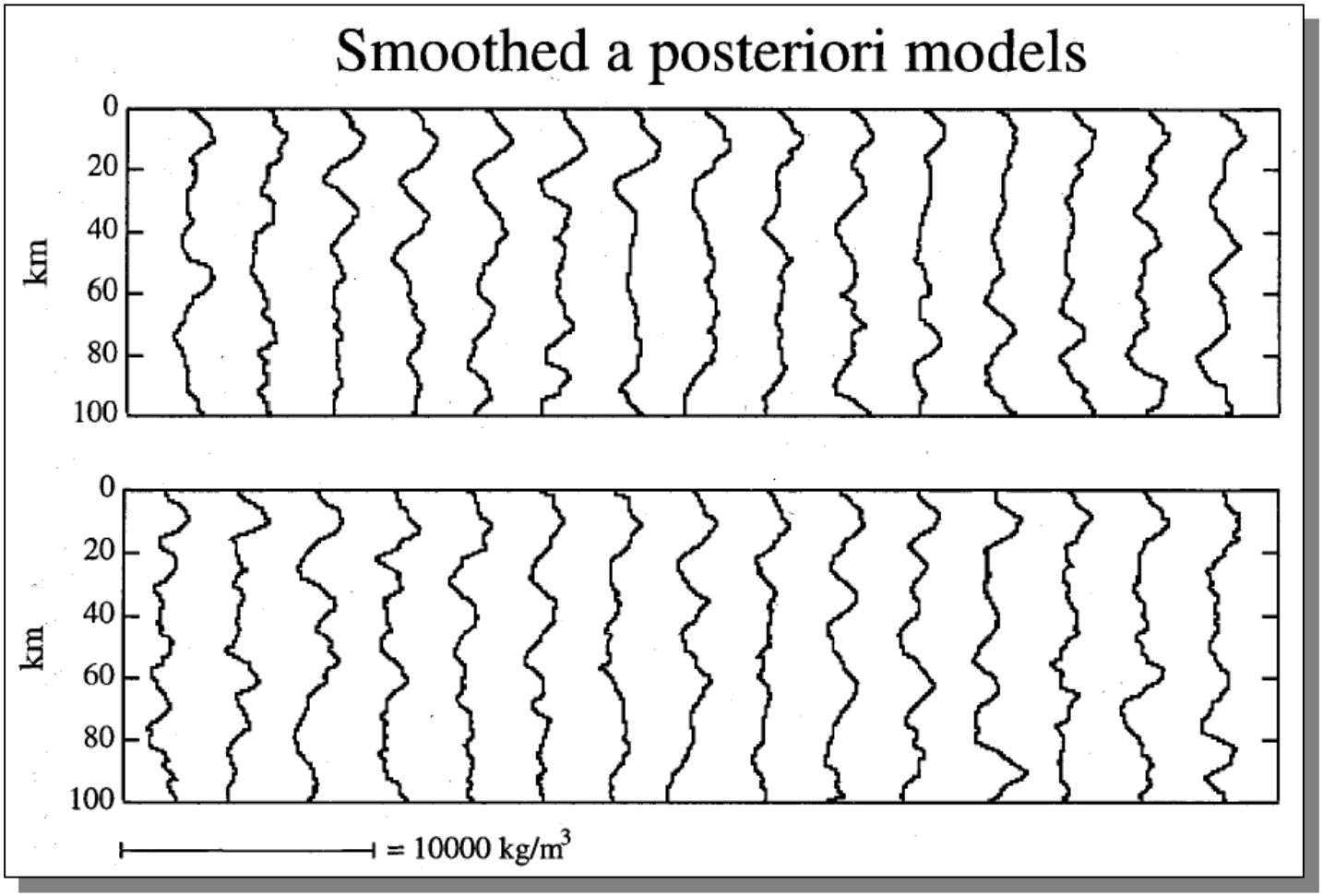
$$L(m) = k\exp\left(-\sum_i \frac{(g_i(m) - d_i^{obs})^2}{\sigma_i}\right)$$

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Gravity: posterior random walk

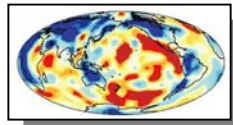


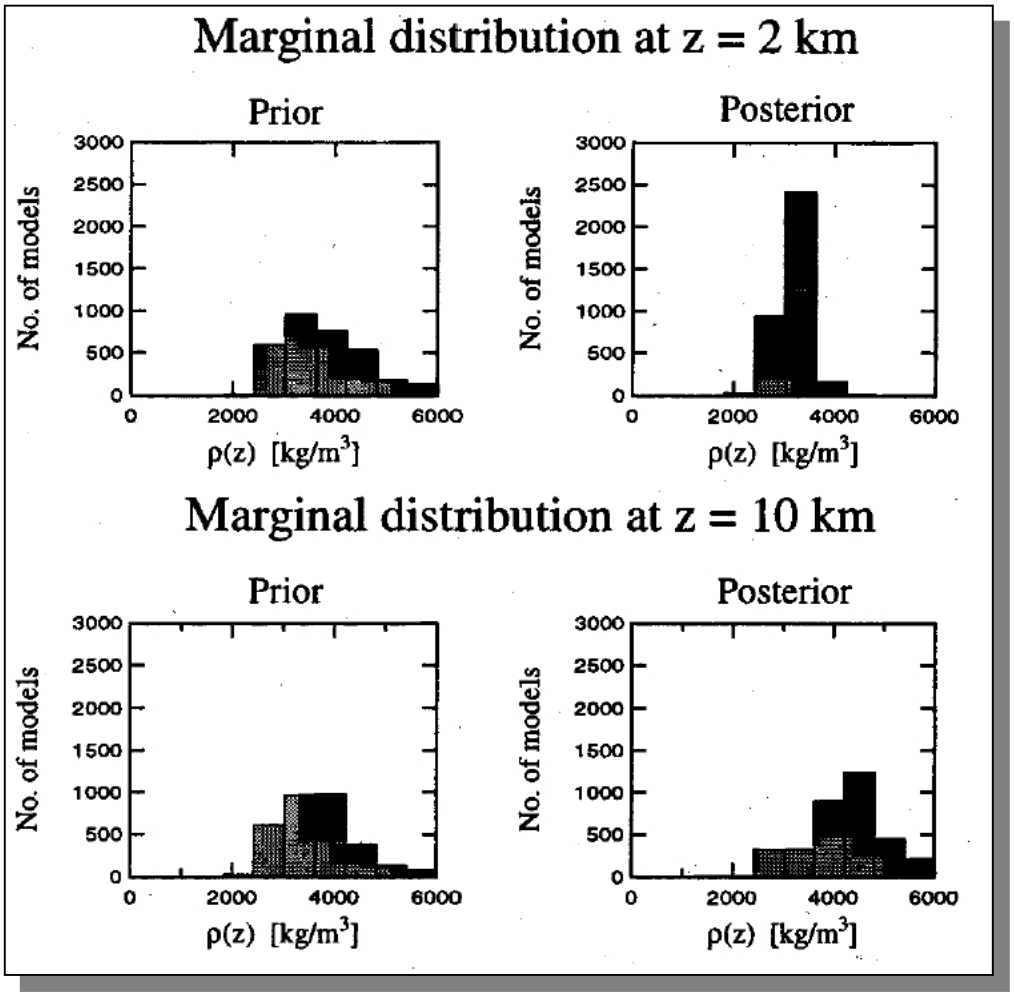
The random walk through the *a posteriori probability* leads to the models:



$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Gravity: posterior probability





Note:

These models are samples of the a posteriori probability density function. They represent the state of information we have on our Earth model. With these samples we can now ask questions like:

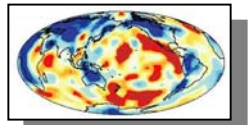
What is the value for the mass density at depth z=2km or z=20km and how well is it constrained? We only have to calculate the marginal probabilities to answer this questions.

Note:

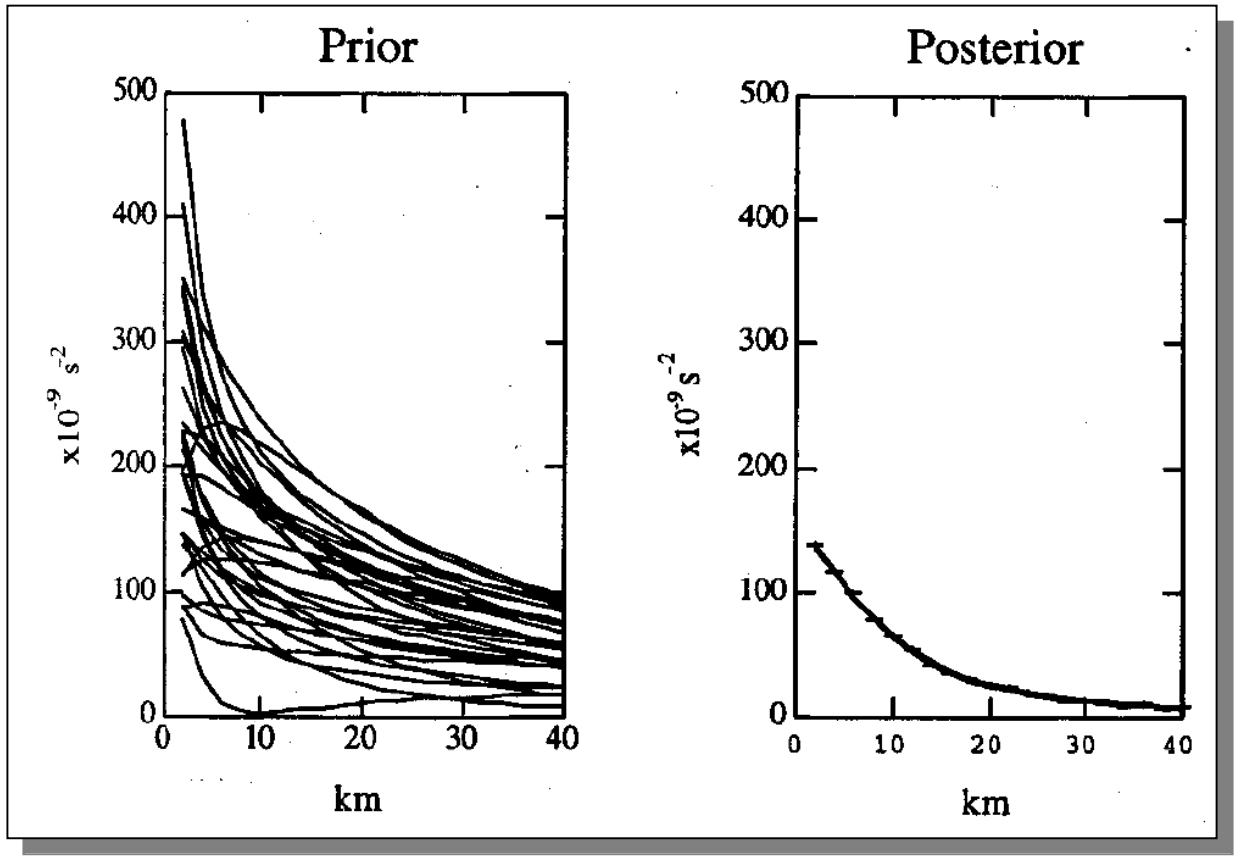
At depth 2km we seem to have clearly gained information.

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Gravity: posterior probability



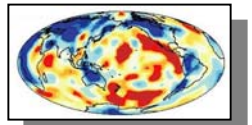
How has the misfit of our models improved compared to the a priori models?



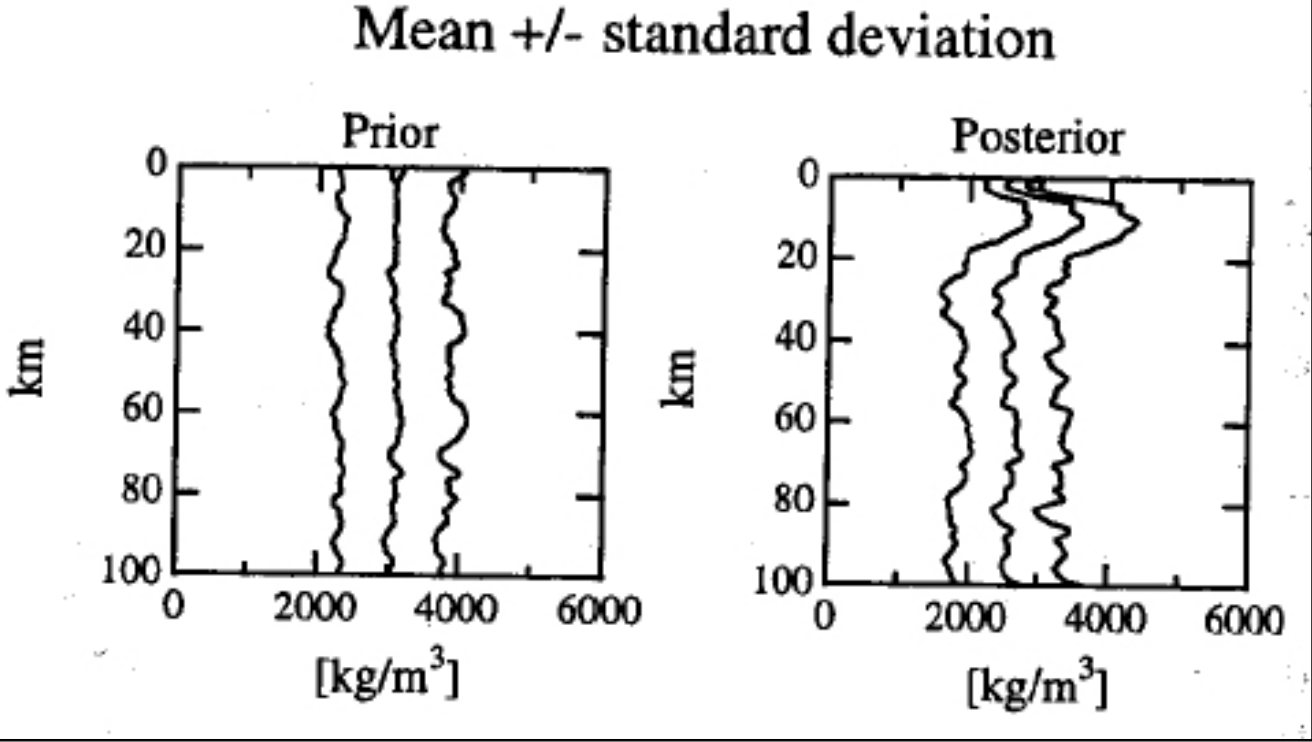
The misfit is almost perfect for all our a posteriori models but again we hit on the particular gravity problem that many **very different** models explain the data!

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Gravity: posterior probability



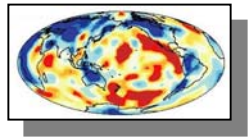
What are the mean values and standard deviations of the density as a function of depth?



Here we clearly see that we gave gained information in the top 20 km !

$$\sigma(d,m) = k\frac{\rho(d,m)\theta(d,m)}{\mu(d,m)}$$

Summary



Monte Carlo methods can be applied to sample a possibly high-dimensional model space defining **prior and posterior probability density functions** of a physical inverse problem.

The sampling of the a posteriori probability seems to be the optimal way of describing the **state of information** in a particular (physical) system.

The key to a successful Monte Carlo algorithm is to **efficiently walk** through the model space and calculate the least possible number of models while providing a representative sample of the a posteriori probability function.