

Lecture 9, Tuesday Oct 26

Student Presentations – Victoria and Lindsay

Last Time (& next time) – ML guest lecture(s)

Today –

Genetic Algorithms

Data Assimilation – tsunami example

Homework 7 – Grid search, due Nov 2

Homework 8 – Gauss-Newton, due Nov 9

Start recording (and share screen)!

Student presentations – Victoria and Lindsay

Inverse Modeling in: A new tool for palaeoclimate reconstruction: Noble gas temperatures from fluid inclusions in speleothems

By T.Klugea, T.Marxa, D.Scholzb, S.Niggemannc, A.Manginib,
W.Aeschbach-Hertiga

(other slides in separate file)

Electromagnetic constraints on a melt region beneath the central Mariana back-arc spreading ridge

Tetsuo Matsuno, Rob L. Evans, Nobukazu Seama, Alan D. Chave

Presentation by Lindsay Harrison

Paper summary

- Electrical resistivity measurements along central Marianas region showing area of high resistivity where melt is expected to exist
 - Seismic evidence ambiguous, MT measurements didn't show melt
- Authors ran forward models and inversions to test various melt geometries that match observations and MT data from the region
- Melt not imaged directly by 2-D magnetotelluric data, but still compatible with observations and movement of melt

Data

- Inverse model used synthetic data generated from forward modeling program FS3D, which uses seafloor topography to determine electrical conductivity

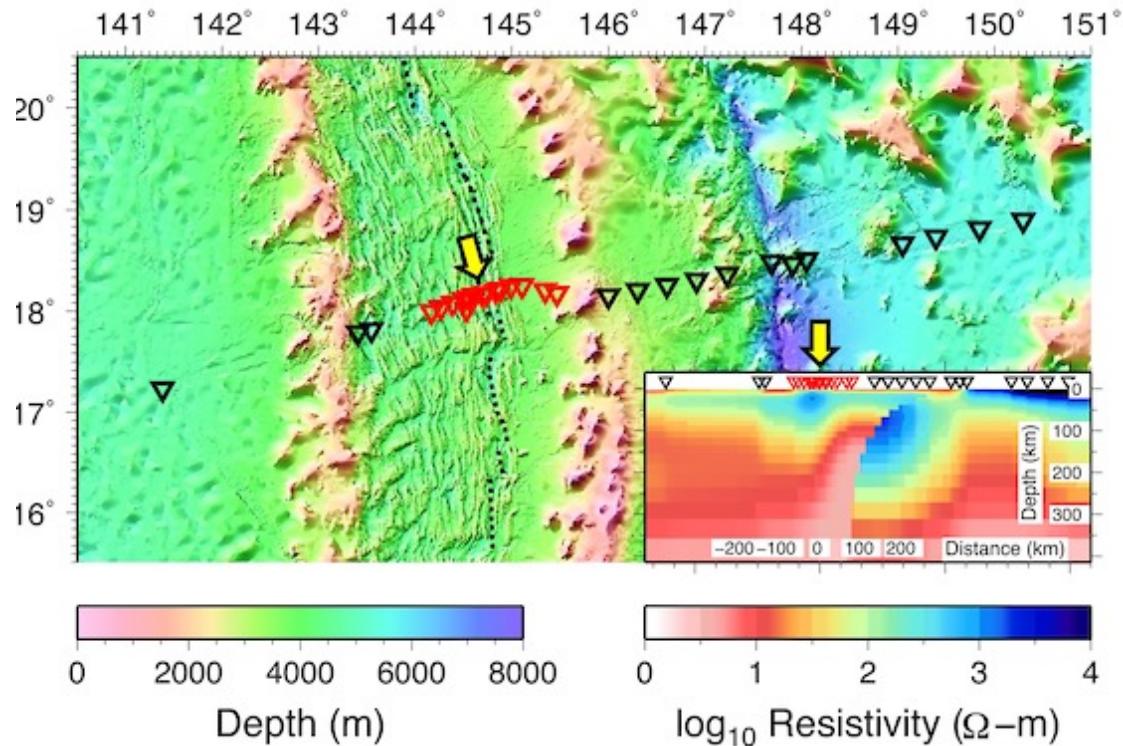


Figure 1 from Matsuno et al., 2012: A bathymetric map of the Marianas region where the MT

FS3D model (Baba and Seama, 2002)

- Seafloor topography modeled by discrete blocks
- Conductivity for both seawater and oceanic crust are multiplied by change in topography at a given coordinate, giving two different conductivity values
- Same with magnetic permeability

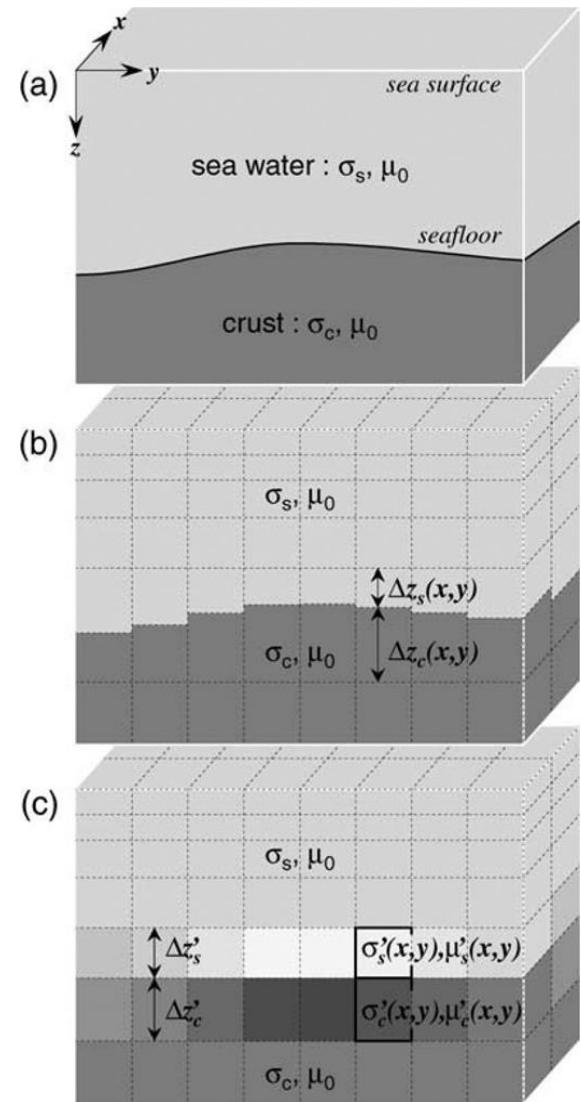
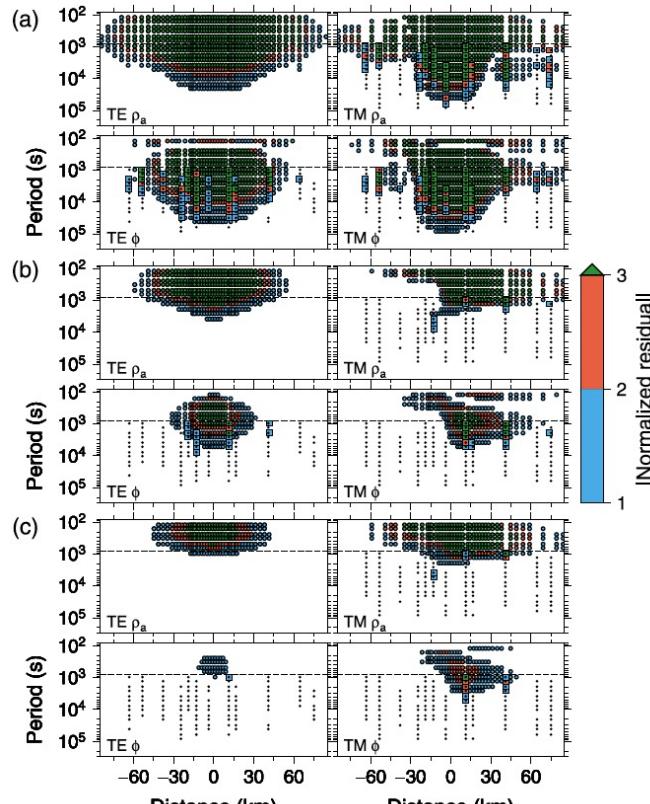


Fig. 1a from Baba and Seama, 2002, showing how the seafloor topography is discretized to develop the forward model

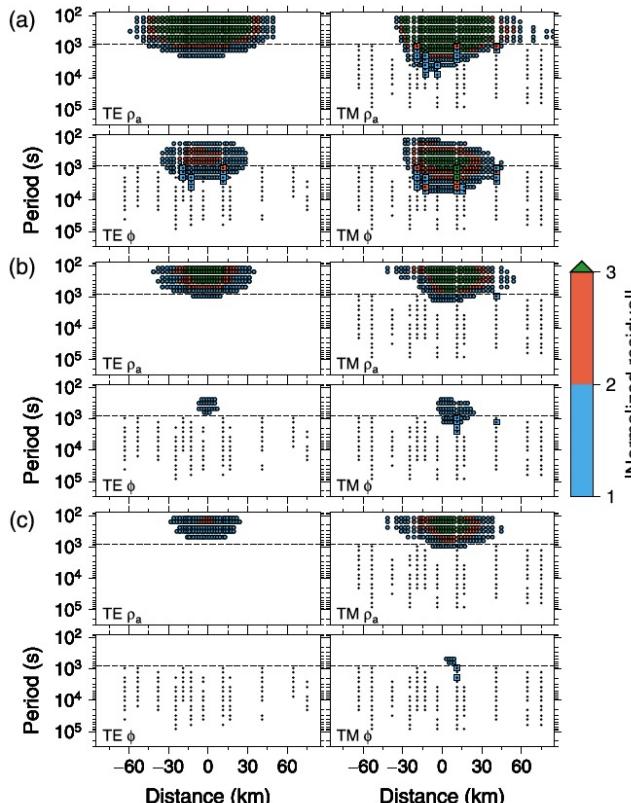
Inversion model

- 2-D synthetic data where 3-D melt body properties are superimposed, then inverted to evaluate perturbations in results from forward model
- Inversion program modeled after Rodi and Mackie (2001), which is an advanced nonlinear conjugate gradient algorithm, incorporating transverse anisotropy
- 2-D model slightly changed to remove small-scale artifacts in upper 6km
 - Prevents numerical instability during forward modeling

Pseudosections of absolute normalized residual for three melt regions



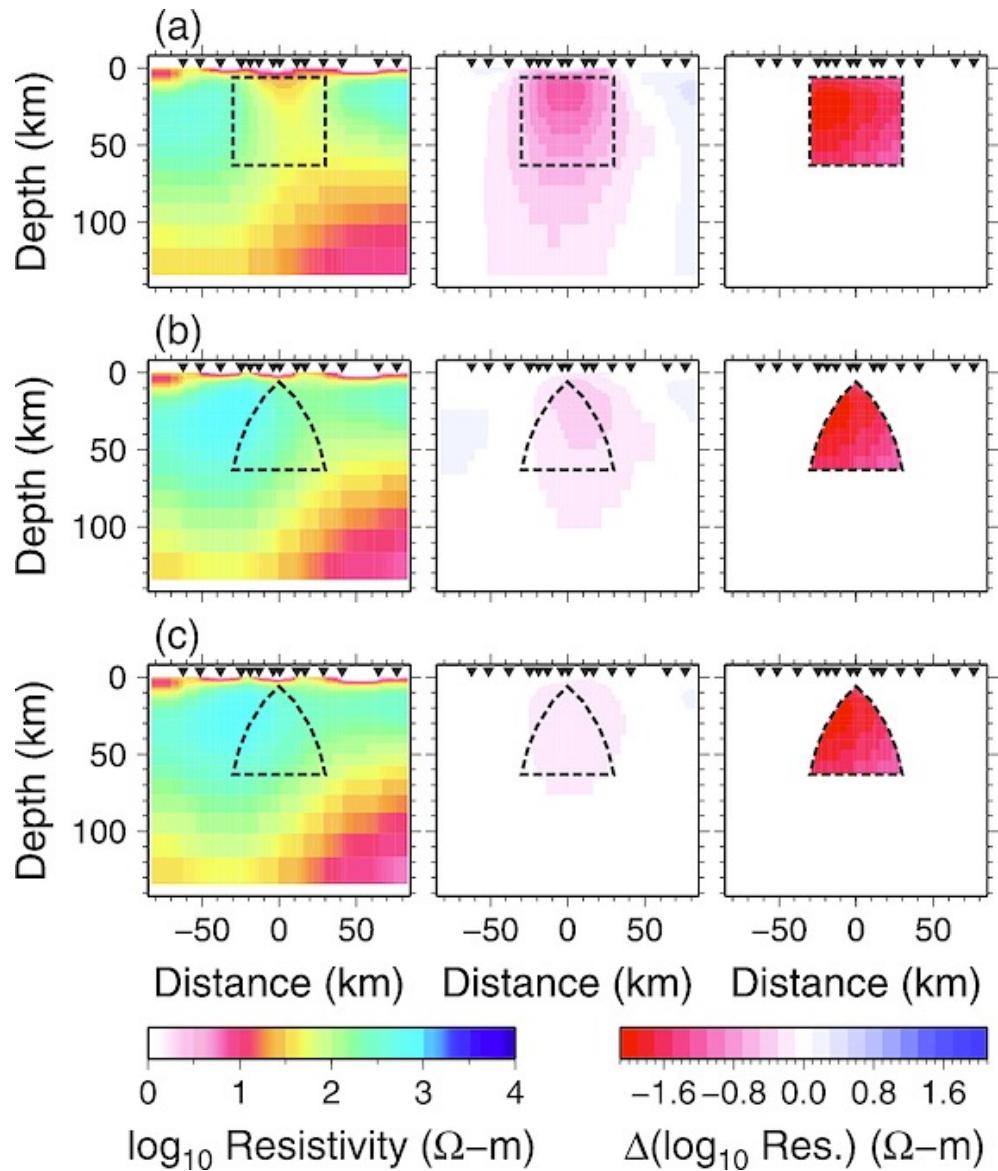
Resistivity of melt regions is 10 $\Omega\text{-m}$



Resistivity of melt regions is 100 $\Omega\text{-m}$

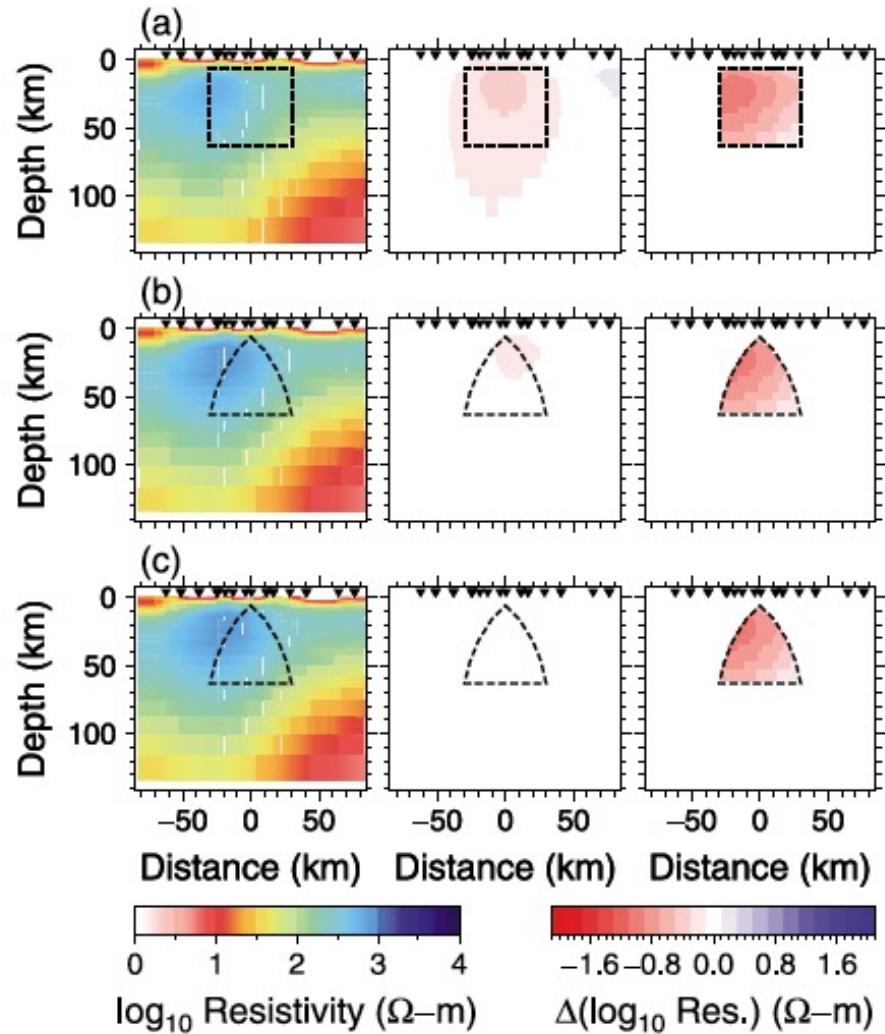
Results

- Cross-sections of the inversion models using (a) square, (b) triangular, and (c) pyramidal geometries. Variations from background resistivity are shown on the right of the figures as $\Delta(\log_{10}\text{Res.})$
- Resistivity of melts is $10 \Omega\text{-m}$



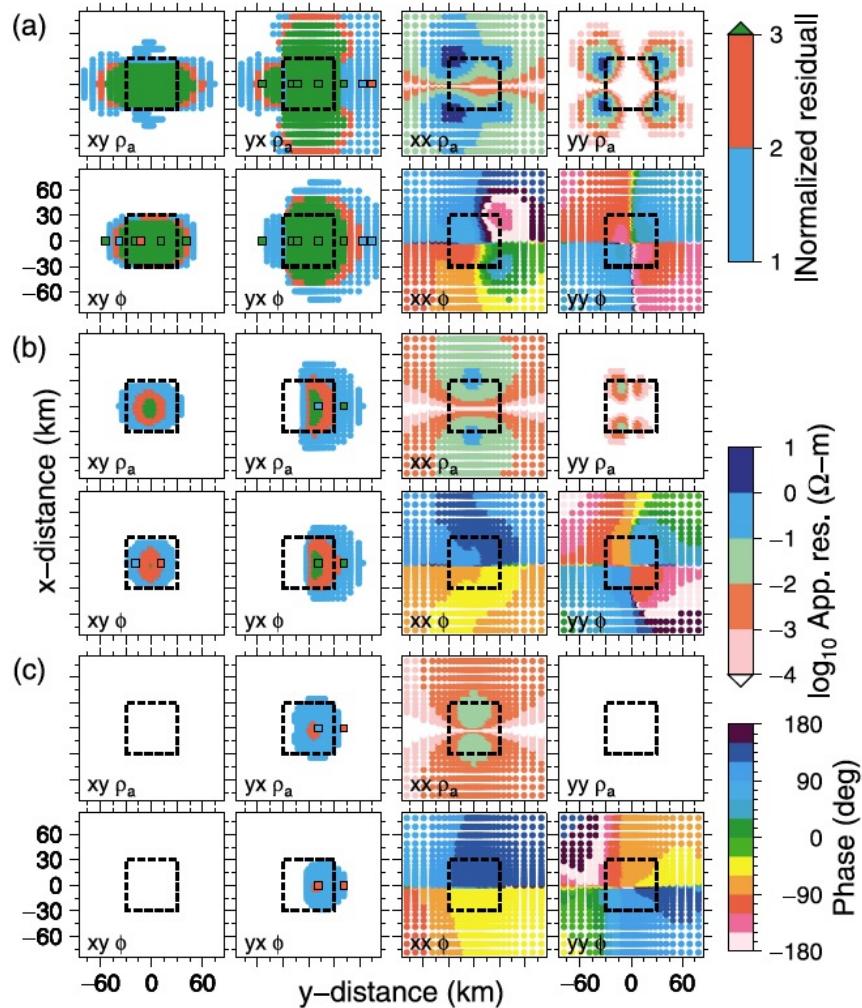
Results,
continued

- Figure same as before, but resistivity is 100 $\Omega\text{-m}$



Discussion

- 2-D MT data sufficient to image 3D melt bodies
- Resistivity model does not require presence of melt region but is compatible with MT observations
- Plan view maps of apparent resistivity



GEOL/PHYS 6670 Geophysical Inverse Theory
Homework #7, Due Tuesday, Nov. 2, 2021

You are given P-wave arrival times for an earthquake recorded by a 13-station seismic array and wish to solve for the earthquake epicenter. The station locations and times are listed below and also in the file HW7-data.txt on Canvas.

Write a computer program that performs a grid search to find the best location for the earthquake. Try every 1 km point in a 100 km by 100 km array ($x = 0$ to 100 km, $y = 0$ to 100 km). At each point, compute the distance to each of the 13 stations. Convert these distances to travel time by assuming the velocity is 6 km/s (this is a 2-D problem, don't worry about depth). Compute the average sum of the squares of the travel time residuals (travel time residual is the predicted time from the grid point to each station that you calculated minus the observed time (P-wave arrival time) at each station listed above) for each grid point (after finding the best-fitting origin time at the grid point; see below). List the best-fitting location and origin time for the earthquake.

GEOL/PHYS 6670 Geophysical Inverse Theory

Homework #8, Due Tuesday, Nov. 9, 2021

This uses the same earthquake location problem as last time, but this time determine the location by linearizing the problem and iterating to solve.

Write a computer program to find the earthquake location by linearizing the earthquake location problem via a Taylor series expansion, and solve via iterative least squares (also called Gauss-Newton method). List the location and origin time for the earthquake that you find by the iterative least squares method.

Upload your code as well as a document describing what you did, and your solution.

The station locations and times are listed below and also in the file HW7-data.txt on Canvas.

P-wave arrival times for an earthquake (each line represents a different seismic station)

x(km) y(km) P-wave arrival time(seconds)

9.000 24.000 14.189

24.000 13.200 13.679

33.000 4.800 13.491

45.000 10.800 14.406

39.000 27.000 13.075

54.000 30.000 15.234

15.000 39.000 13.270

36.000 42.000 12.239

27.000 48.000 12.835

48.000 48.000 14.574

15.000 42.000 12.624

18.000 15.000 13.496

30.000 36.000 10.578

Global Optimization Toolbox

Search MathWorks.com



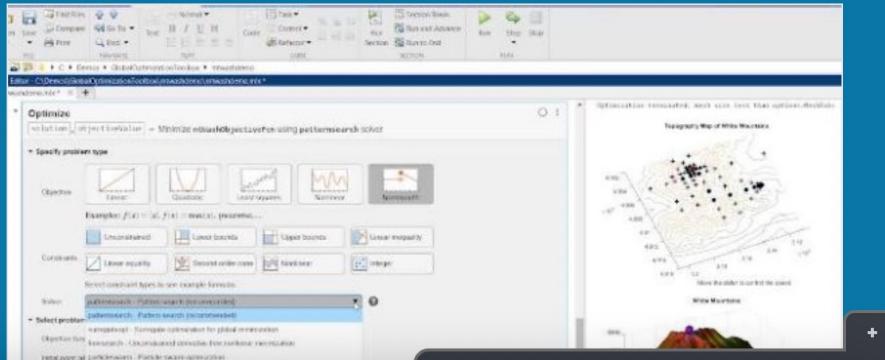
Global Optimization Toolbox

Solve multiple maxima, multiple minima, and nonsmooth optimization problems

 Download a free trial

[View Pricing](#)

Global Optimization Toolbox provides functions that search for global solutions to problems that contain multiple maxima or minima. Toolbox solvers include surrogate, pattern search, genetic algorithm, particle swarm, simulated annealing, multistart, and global search. You can use these solvers for optimization problems where the objective or constraint function is continuous, discontinuous, stochastic, does not possess derivatives, or includes simulations or black-box functions. For problems with multiple objectives, you can identify a Pareto front using genetic algorithm or pattern search solvers.



Global Optimization

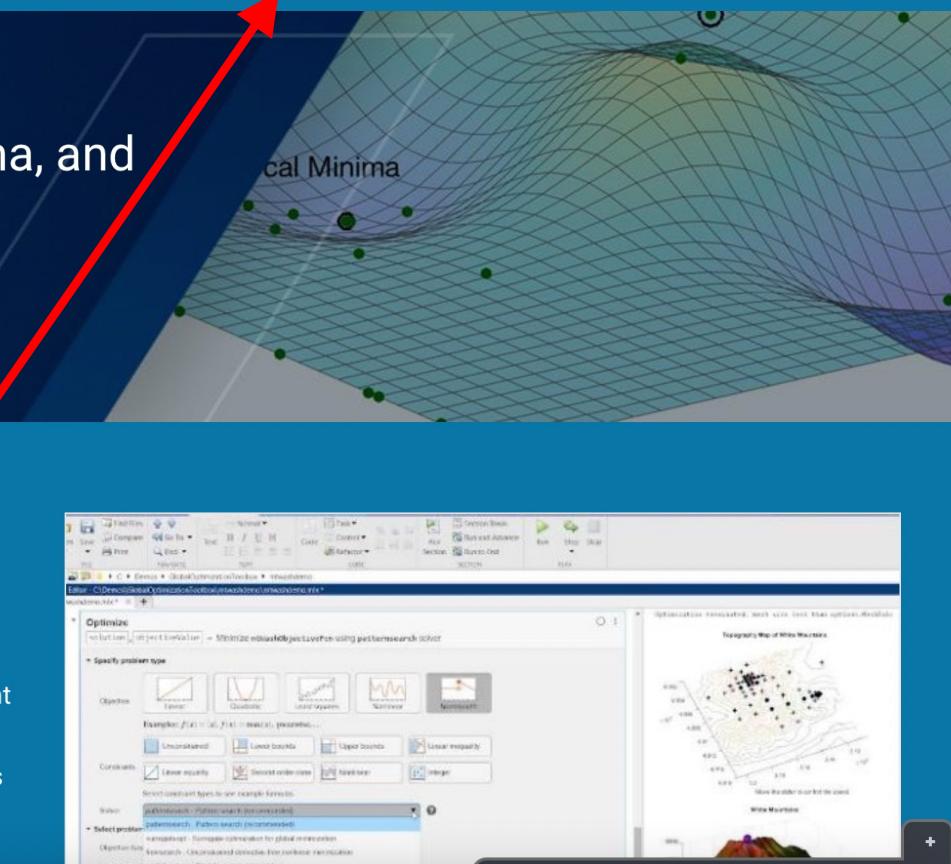
Global Optimization Toolbox provides functions that search for global solutions to problems that contain multiple maxima or minima. Toolbox solvers include surrogate, pattern search, genetic algorithm, particle swarm, simulated annealing, multistart, and global search. You can use

Global Optimization Toolbox

Solve multiple maxima, multiple minima, and nonsmooth optimization problems

[Download a free trial](#)[View Pricing](#)

Global Optimization Toolbox provides functions that search for global solutions to problems that contain multiple maxima or minima. Toolbox solvers include surrogate, pattern search, genetic algorithm, particle swarm, simulated annealing, multistart, and global search. You can use these solvers for optimization problems where the objective or constraint function is continuous, discontinuous, stochastic, does not possess derivatives, or includes simulations or black-box functions. For problems with multiple objectives, you can identify a Pareto front using genetic algorithm or pattern search solvers.



Genetic algorithms – a smart parameter search method (global optimization technique). Using analogy of genetics and natural selection from biology. Good models allowed to ‘reproduce’. Mutations are allowed (to explore other parts of model space).

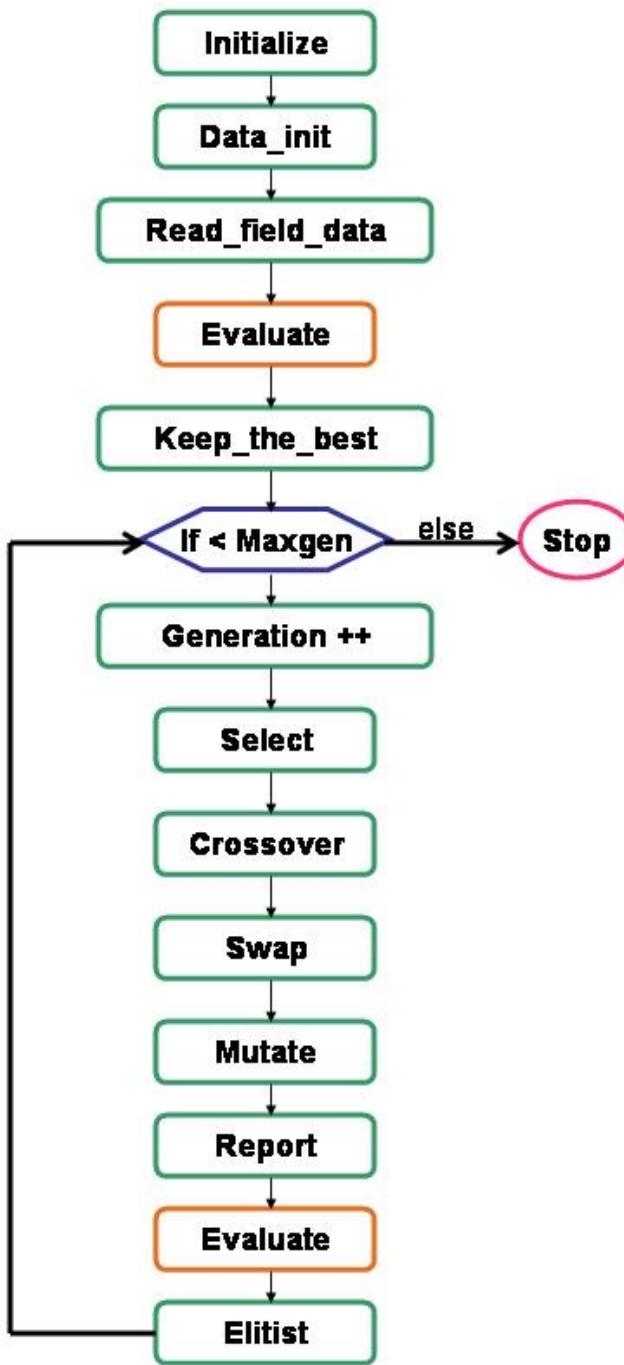
Genetic Algorithms

- For optimization (inversion) problems, GAs use the principle of genetics to search for new and better solutions to a problem, based on some fitness criteria.
- *Remember, you get to choose the fitness criteria! For an optimization problem, that becomes the source function you want to solve.*
- During *reproduction* and *crossover*, new genetic combinations occur and, finally, a subject can inherit a good gene from both parents to produce the best answer.
- The GA is especially effective because recombination, the crossover of partial solutions, greatly improves the capability of the algorithm to approach the correct or best solution.
- Natural selection is coded via the *fitness function* and a selection subroutine, that keeps more of the better members, driving the population to better solutions.

An example GA

- Each genome in the population represents the parameters of interest in our inversion problem.
- An original 100-member population is created with a standard random number generator.
- Real-valued GA (for some GAs, the parameters are coded as binary); this saves their having to be converted back and forth.
- The best member of the population is always kept, as it would defeat the purpose of optimization to lose the fittest solution.
- The fitness function is coded separately, and can be called in any language as an executable. It computes the forward model for each member of the population, and then computes how close that member is to the measured values.
- *The measured values are actual data, what the GA is trying to find is that set of parameters that produces the closest solution to the data.*

Flow Chart



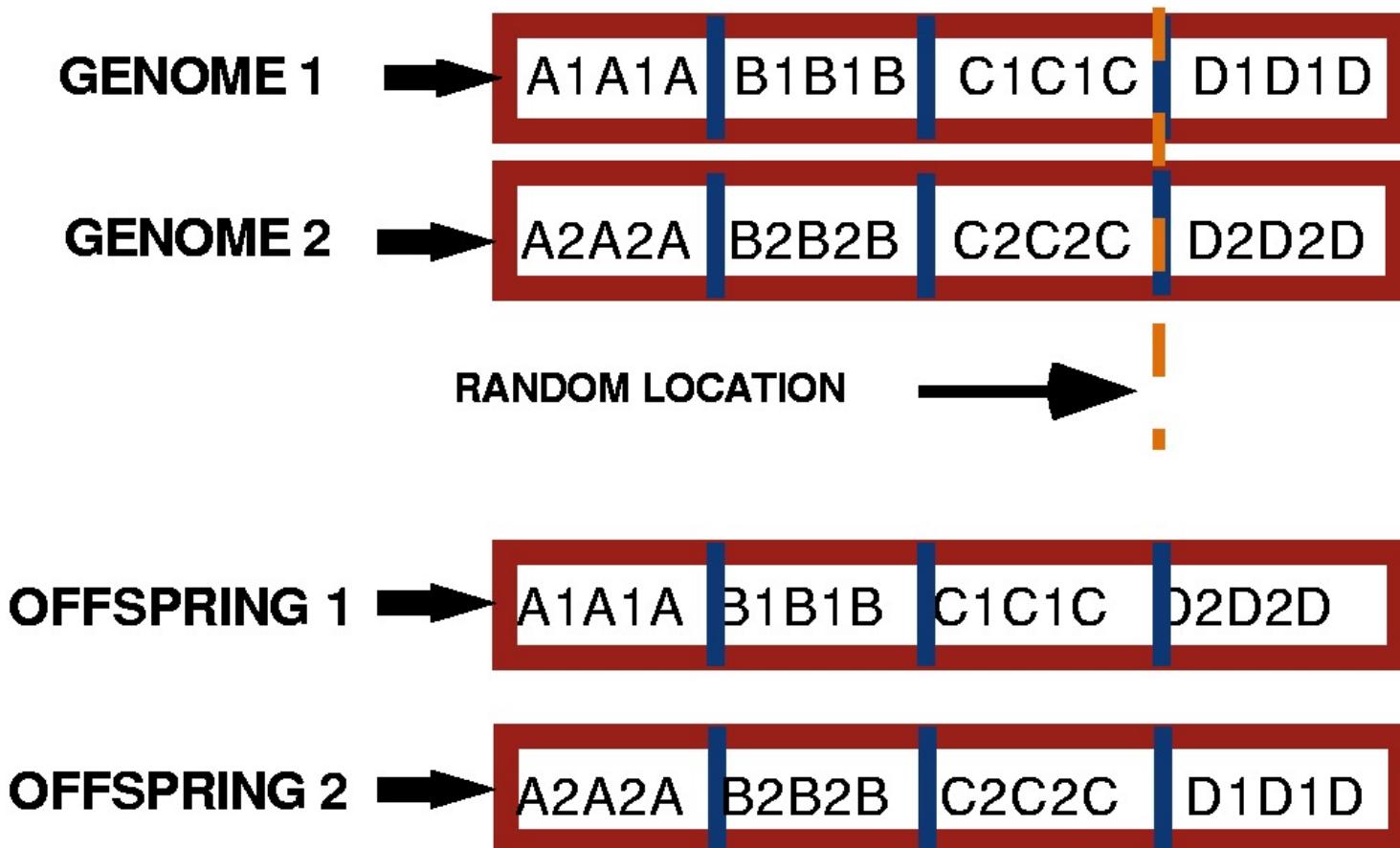
Selection

- This is natural selection, or survival of the fittest.
- Rank each new generation by their relative fitness.
- Set up some scheme for keeping more of the fitter members of the population, still preserving some variation. The actual scheme can vary.
- You don't want to keep only the very best members of the population – you lose too much good information in the smaller pieces of the genome.

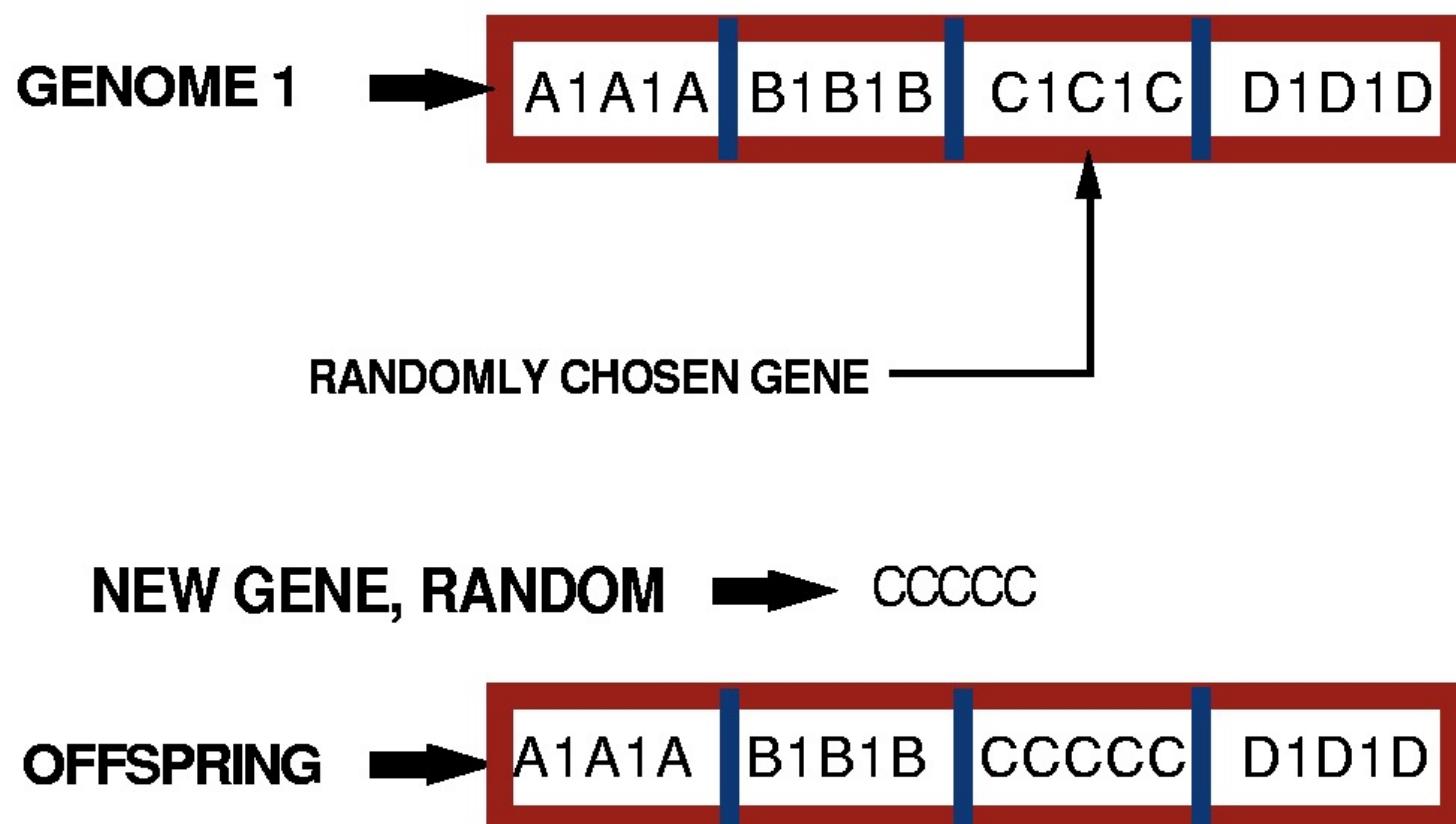
Mutation

- Mutation prevents stagnation of the GA inversion – it prevents the GA from getting stuck in a local minima by introducing new genetic material.
- Again, mutation adds variation to the inversion process.

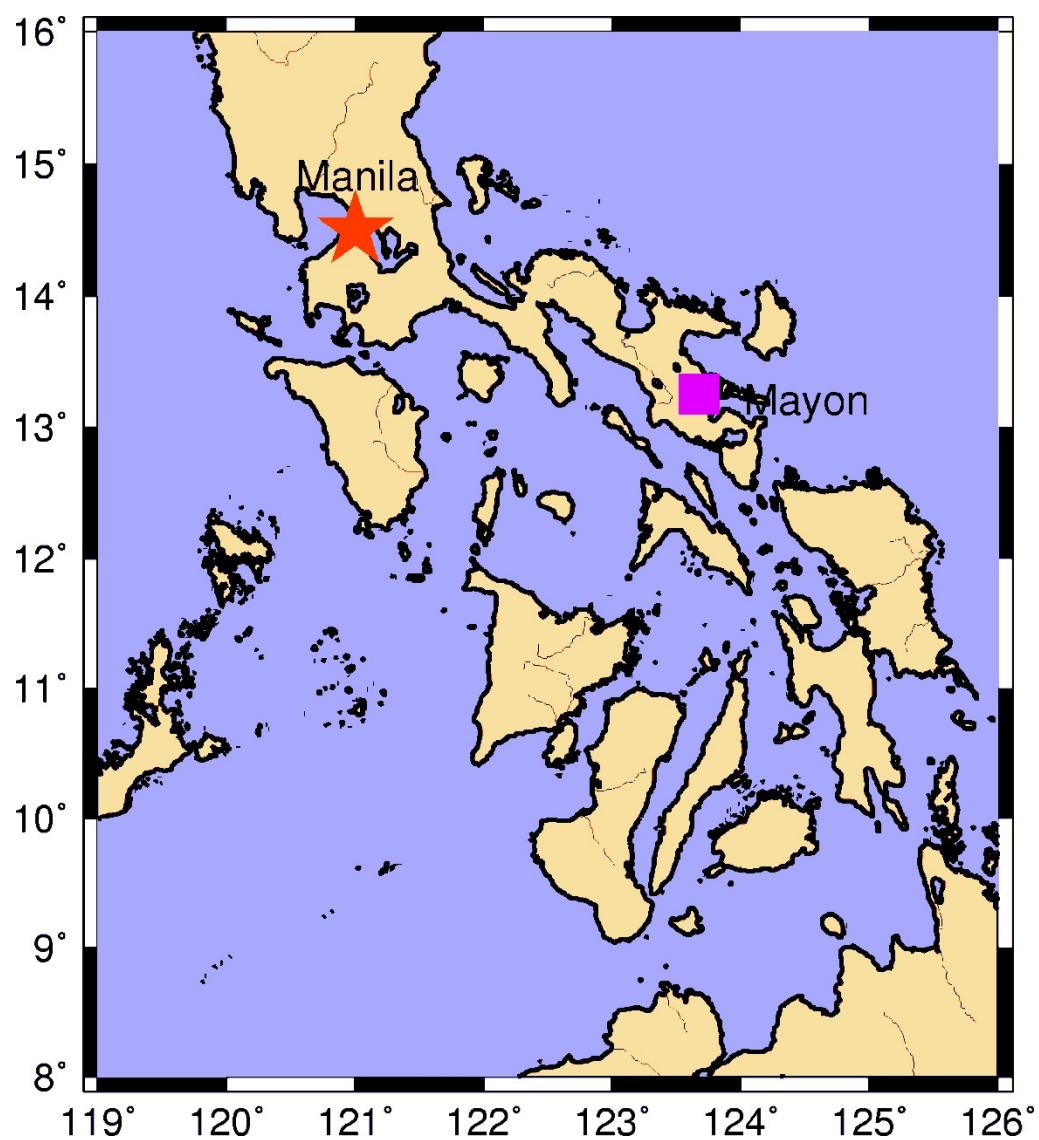
Crossover



Mutation



An example – Mayon volcano, Philippines



Tiampo et al., 2004

Mayon Volcano

- Located 330 km southeast of Manila in the Bicol volcanic chain, nearly 1 million people live within 15 km of Mayon.
- The most active volcano in the Philippines, it has erupted 49 times since 1616. These eruptions include pyroclastic flows, lahars, mudflows, heavy ash plumes, ejecta, and lava flows.
- 12 activity periods distributed regularly throughout this century.
- Eruptions in 2000 and 2001 closed the airport and caused the evacuation of 83,000 and 46,000 people, respectively.
- The 1993 eruption killed 68 people and prompted the evacuation of 60,000 people.
- Gravity measurements, in conjunction with GPS measurements, began in 1992.

Single Source Elastic-Gravitational Model

- This is a six parameter fit for the GA: depth, pressure, mass, radius, and x & y location.
- These six parameters are coded as genes, and fed into the fitness function to calculate the modeled gravity and deformation changes.
- How close they come to the actual measurements is a measure of each members fitness.

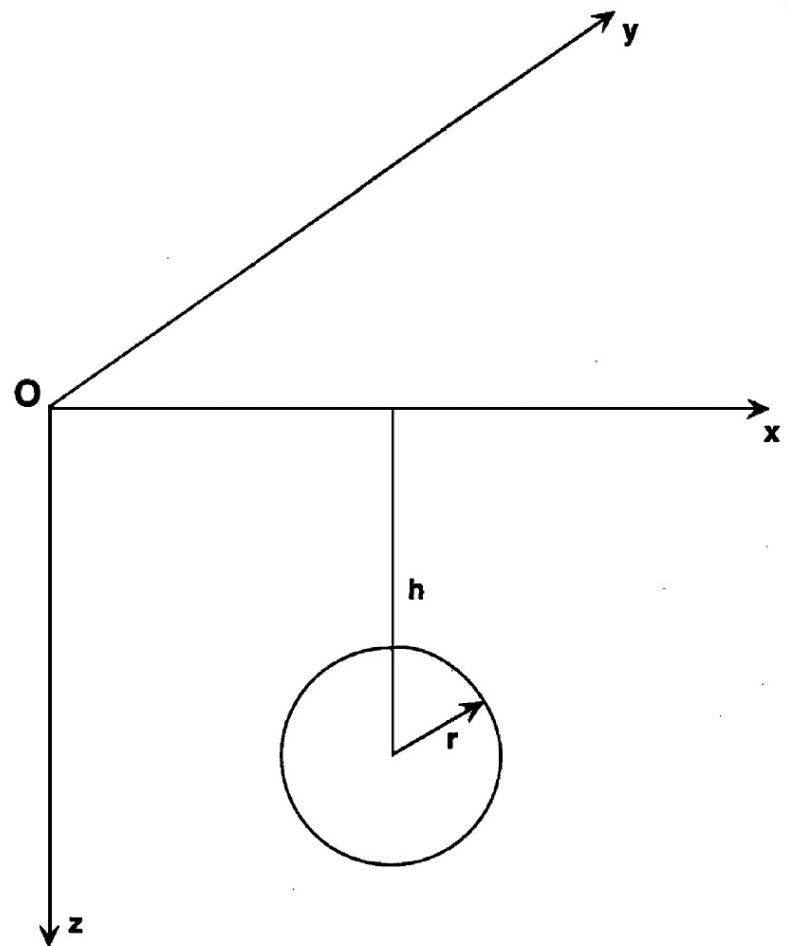


Fig. 3. Six parameter elastic-gravitational source. With O as origin, parameters are x and y location, intrusion depth h , radius r , pressure increment P , and mass increment, m .

actual data. Specifically, the gravity or deformation pattern for the expected volcanic source is calculated in this subroutine and then the value of chi-square, χ^2 , is calculated for that deformation pattern as it relates to the measurement pattern.

$$\chi^2 = \sum_{all \ k} \frac{(C_k - E_k)^2}{\sigma_k^2}, \quad (1)$$

where E_k are the expected, or measured deformations or gravity changes, C_k are the calculated deformations or gravity changes, σ_k is the standard deviation for each measurement, and $k=1, 2, \dots, N$ is the total number of measurements (Taylor, 1982; Bevington and Robinson, 1992). The GA picks the fittest members of the

An example – Seismic waveform inversion

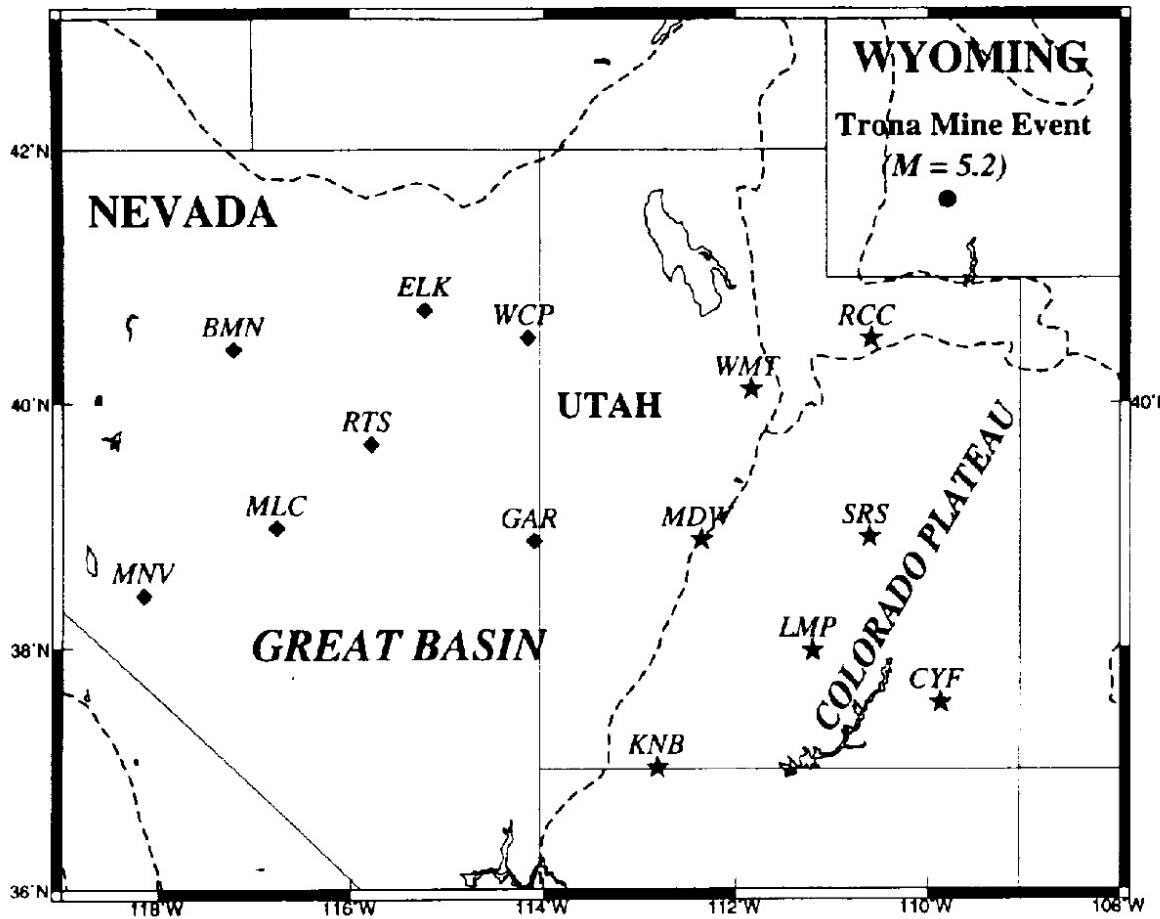


Figure 1. Location of the 3 February 1995 mine-collapse event and the seismic stations at regional distances used in this study. Stations BMN, DUG, ELK, KNB, and MNV belong to the U.S. National Seismic Network (USNSN), and the rest of the stations are from the Colorado plateau to Great Basin (CPGB) PASSCAL experiment. The diamonds indicate stations used for estimating Great Basin crustal structure, and stars indicate stations used for modeling the Colorado plateau crust. Solid lines denote state boundaries, and dashed lines show boundaries between geologic provinces.

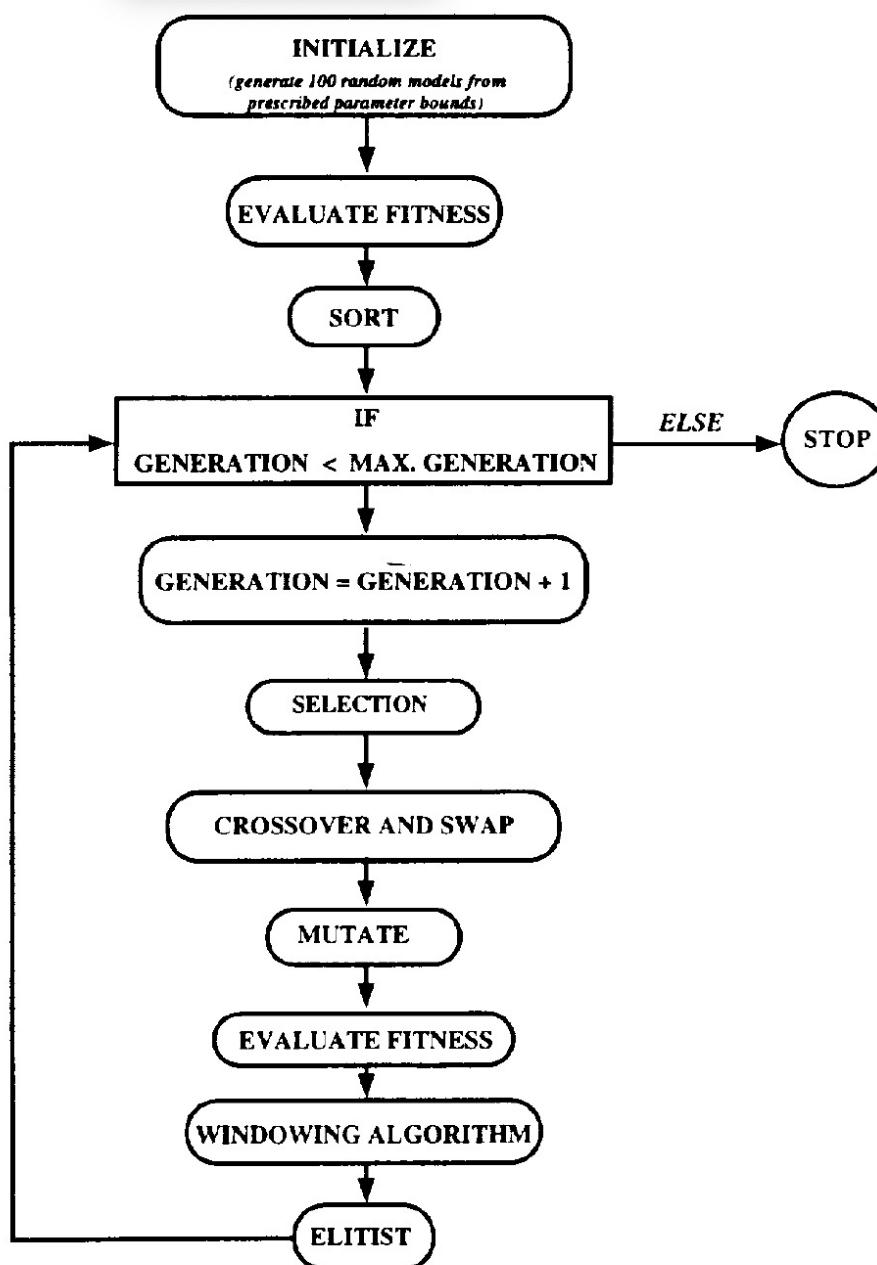


Figure 3. The flowchart of the genetic algorithm adopted in this study. The individual steps are discussed in the text.

Input Parameters. The following parameters are set *a priori* in the modeling algorithm and have the same value in each generation:

1. Probability of crossover. We tested the convergence for both one- and two-point crossover schemes and found their rates to be similar, producing nearly the same model solutions. We thus use both of these variations of GA in our modeling. The crossover probability was set based on tests run on synthetic data to determine the best convergence for different crossover rates.
2. Probability of mutation. This probability value controls the number of times mutation occurs in a given population, that is, for models in a particular generation. This value can be optimally set to equal $1/n$, the number of model parameters (Bäck, 1996). In this study, n is equal to 8.
3. Range of model parameters. The algorithm chosen for this study randomly searches for model parameters within a prescribed range of values. Unrealistic model parameters can lead to inordinately large model spaces, thus leading to slower convergence.

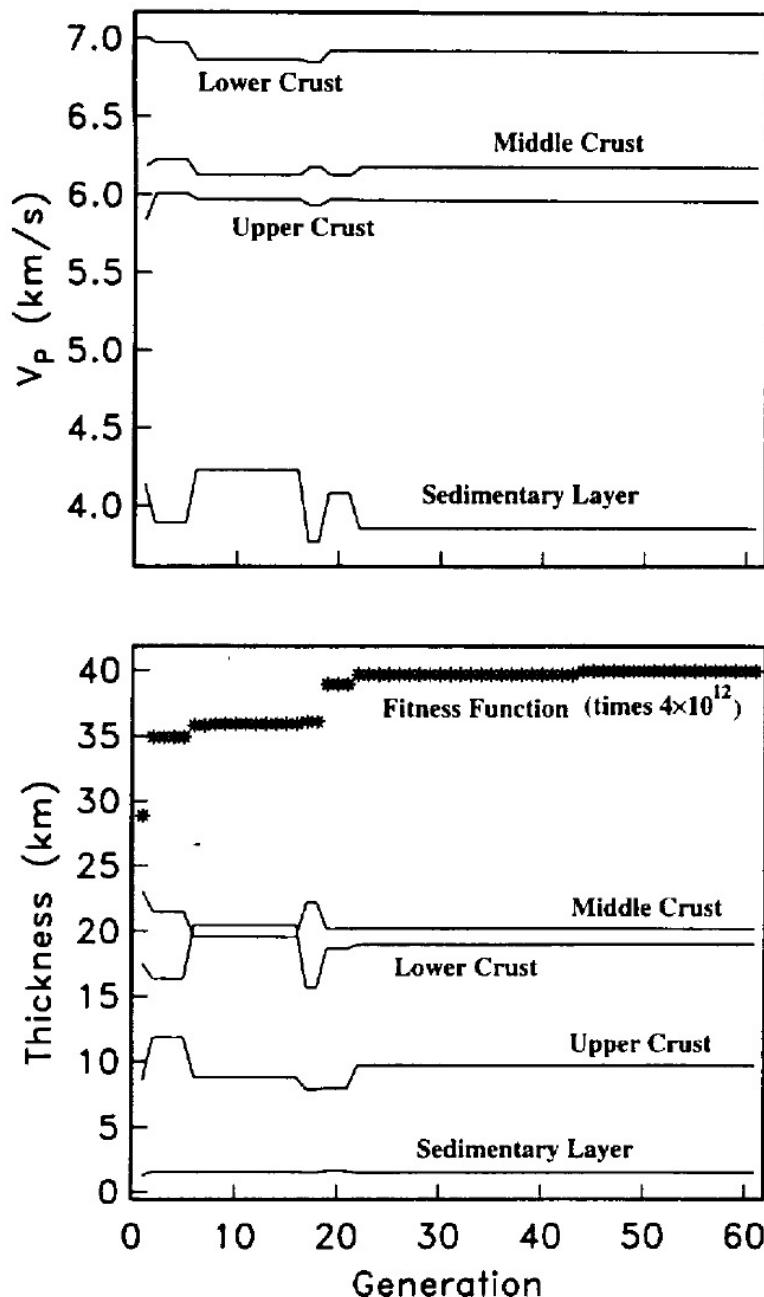


Figure 5. P -wave velocity, thickness, and waveform fitness as a function of generation number for a genetic algorithm run with noise-free data (described in text). The crustal model that best fits the synthetic seismograms for an input model (Fig. 7) is plotted for each generation. We note that adequate convergence is reached by the 25th iteration. A one-point crossover scheme is adopted for this run with the crossover probability equal to 0.85 and the mutation probability equal to 0.125 (described in text).

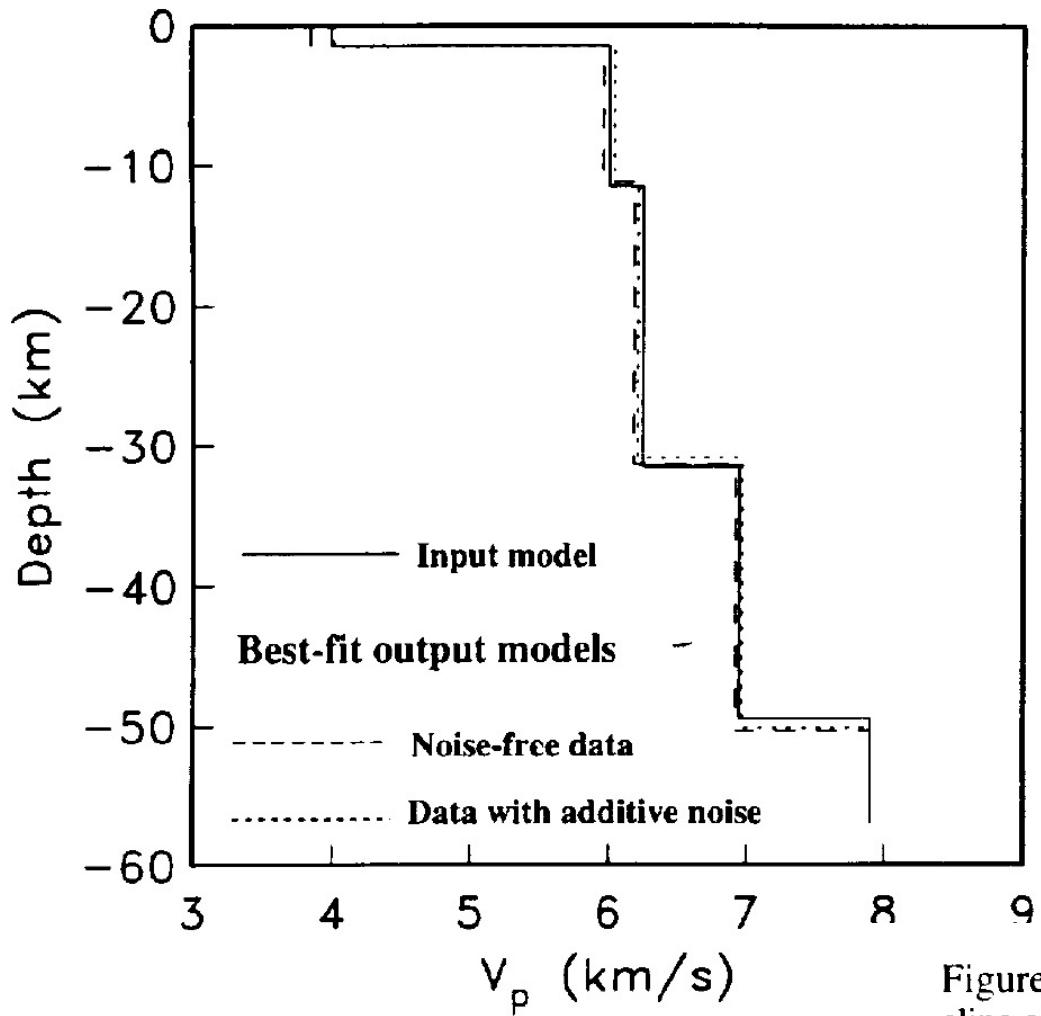


Figure 4. Results from synthetic test of the modeling algorithm developed in this study. We show the best-fit inverted models computed using seismograms constructed from the given input model. Two sets of seismograms, that is, with and without additive noise, are inverted several times each using different randomizing seed values and crossover schemes. We show the average models in this figure. Note that most of the model parameters are retrieved to within $\pm 5\%$ of their input values.

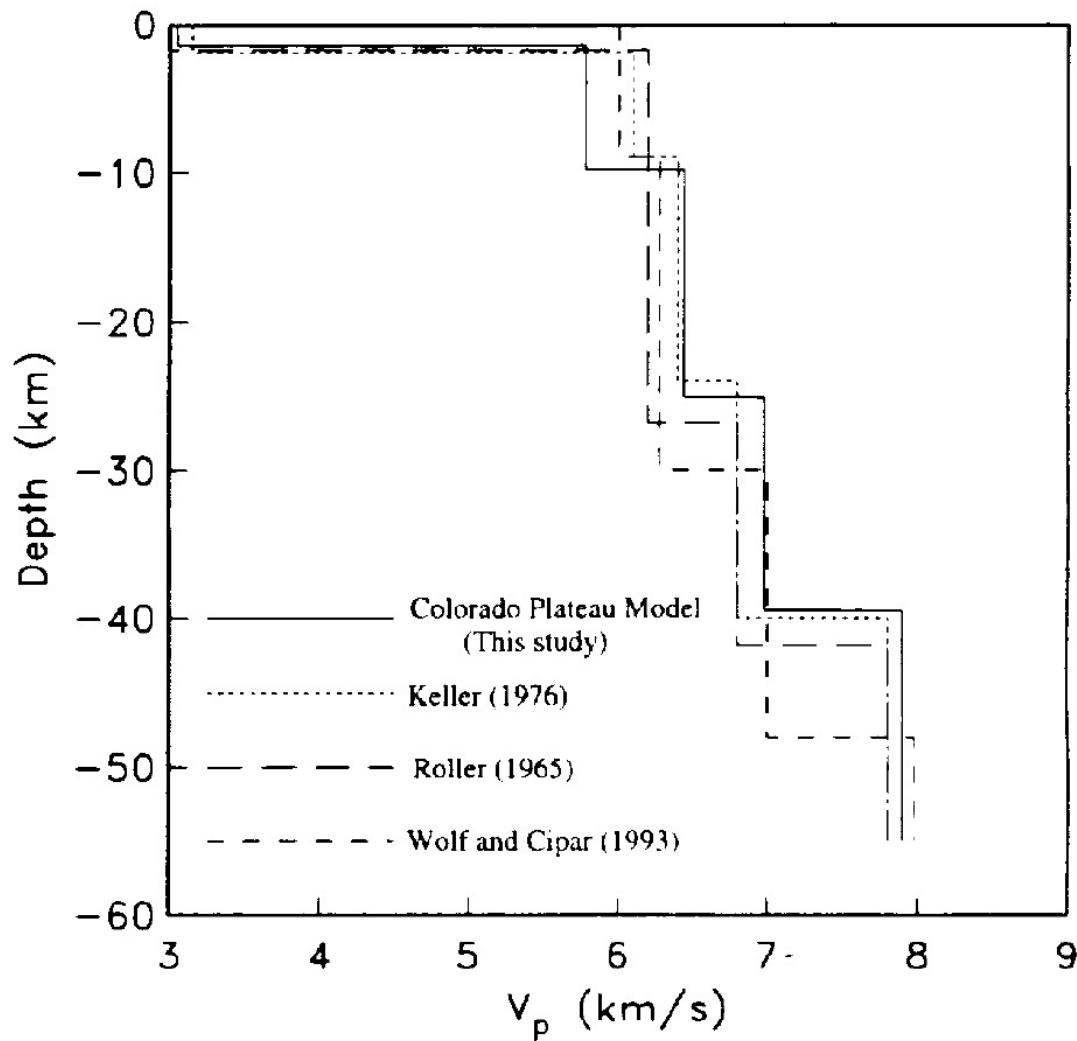


Figure 7. Comparison of our final Colorado plateau model with published models of one-dimensional crustal structure of the same region. The error bounds of our preferred model are given in Table 1.

Genetic Algorithm

Search MathWorks.com

[Trial software](#) [Contact sales](#)

Find global minima for highly nonlinear problems

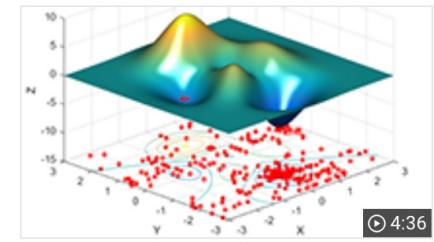
A genetic algorithm (GA) is a method for solving both constrained and unconstrained optimization problems based on a natural selection process that mimics biological evolution. The algorithm repeatedly modifies a population of individual solutions. At each step, the genetic algorithm randomly selects individuals from the current population and uses them as parents to produce the children for the next generation. Over successive generations, the population "evolves" toward an optimal solution.

You can apply the genetic algorithm to solve problems that are not well suited for standard optimization algorithms, including problems in which the objective function is discontinuous, nondifferentiable, stochastic, or highly nonlinear.

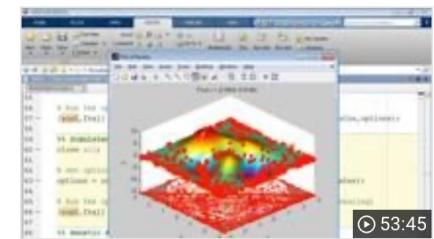
The genetic algorithm differs from a classical, derivative-based, optimization algorithm in two main ways, as summarized in the following table.

Classical Algorithm	Genetic Algorithm
Generates a single point at each iteration. The sequence of points approaches an optimal solution.	Generates a population of points at each iteration. The best point in the population approaches an optimal solution.
Selects the next point in the sequence by a deterministic computation.	Selects the next population by computation which uses random number generators.

For more information about applying genetic algorithms, see [Global Optimization Toolbox](#).



[What Is a Genetic Algorithm?](#)



[Tips and Tricks: Getting Started Using Optimization with MATLAB](#)

Data assimilation

(in separate powerpoint)

