Earthquake detection and location: towards automatization
Day 3

Christian Sippl

Geofyzikální ústav Akademie věd ČR, v.v.i.

Universidad de Concepcion, January 13-17, 2020

Targeted phase pickers

Event association - coincidences, traveltime grids and more

Now: we assume we have the final set of picks and know which ones form an event –¿ we need to locate this event

- **Monday**: Intro and earthquake detection basics (triggers)
- **Tuesday**: Phase pickers and the earthquake association problem
- **Wednesday**: Basic earthquake location techniques, uncertainty estimation
- **Thursday**: Multi-event (re)location techniques, use of cross-correlations
- **Friday**: Putting it all together: how to design an automated approach

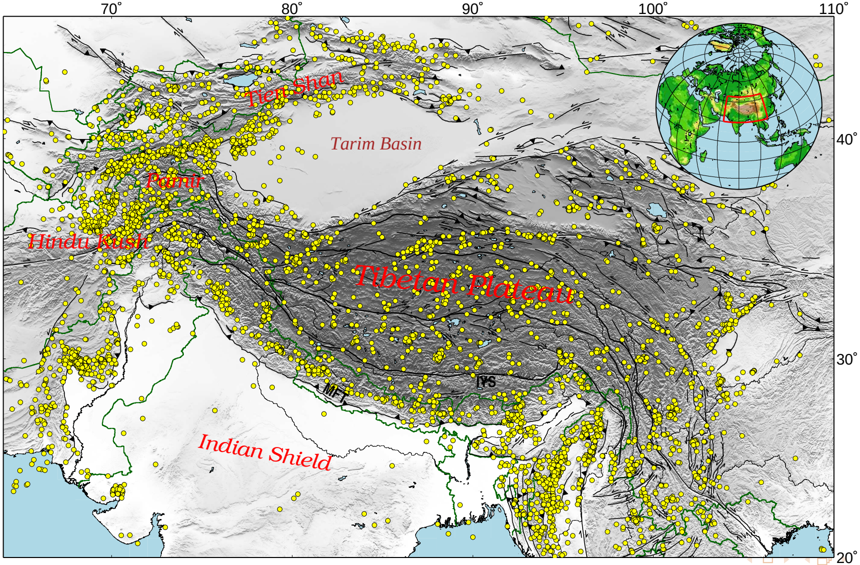
- **Monday**: Intro and earthquake detection basics (triggers)
- **Tuesday**: Phase pickers and the earthquake association problem
- **Wednesday**: Basic earthquake location techniques, uncertainty estimation
- **Thursday**: Multi-event (re)location techniques, use of cross-correlations
- **Friday**: Putting it all together: how to design an automated approach

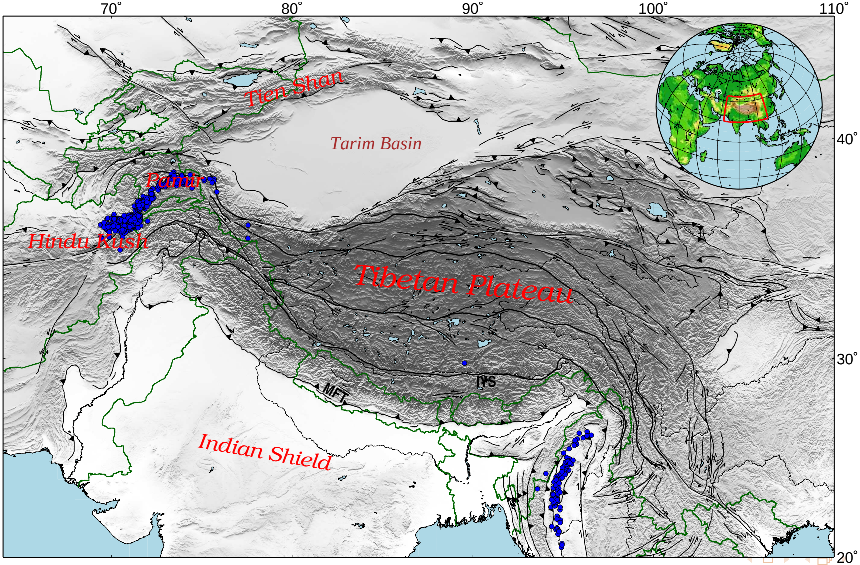
Wednesday: Basic earthquake location techniques, uncertainty estimation

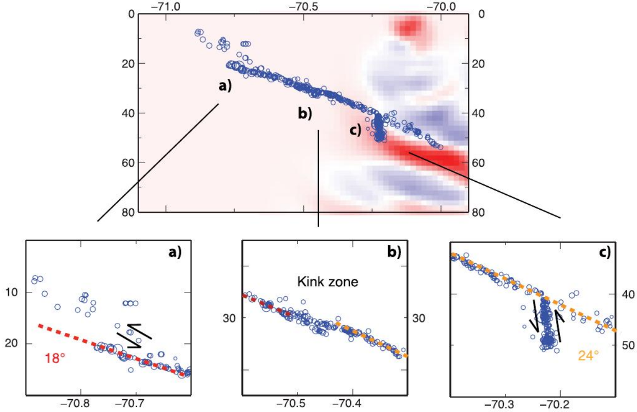
- *9:00 - 10:30*: Earthquake location basics, algorithms
- *11:00 - 12:30*: Excercises - locating earthquakes
- *13:30 - 15:15*: Location uncertainty and error estimates
- *15:30 - 17:00*: Exercises - location error estimates

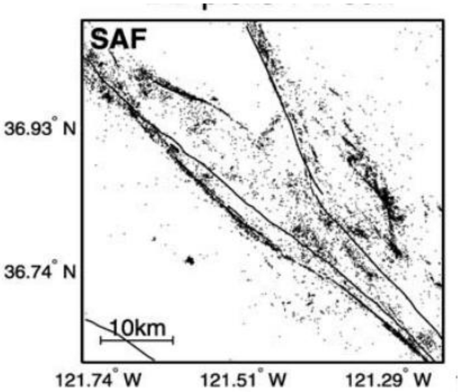
Day 3: Goals

1. You have acquired basic knowledge about how earthquake location programs work
2. You have a good understanding about the sources of location uncertainty, and how it can be estimated
3. You have gained experience in using earthquake location codes, and judging how reliable their output may be


70°
80°
90°
100°
110°
40°
30°
20°
Tien Shan
Tarim Basin
Pamir
Hindu Kush
Tibetan Plateau
MFT
IYS
Indian Shield


70°
80°
90°
100°
110°
40°
30°
20°
Tien Shan
Tarim Basin
Pamir
Hindu Kush
Tibetan Plateau
MFT
IYS
Indian Shield


−71.0
−70.5
−70.0
0
20
40
60
80
a)
b)
c)
Kink zone
18°
24°
10
20
30
40
50
−70.8
−70.7
−70.5
−70.4
−70.3
−70.2


SAF
36.93° N
36.74° N
10km
121.74° W
121.51° W
121.29° W

Question

How many stations are necessary to locate an earthquake?

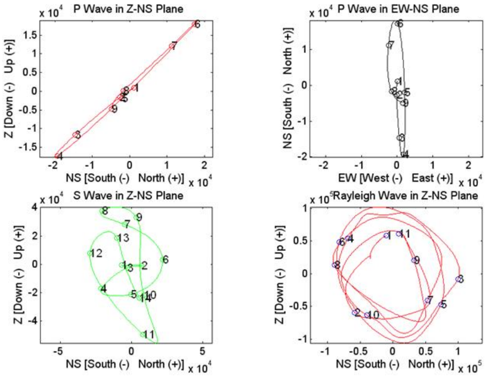
Question

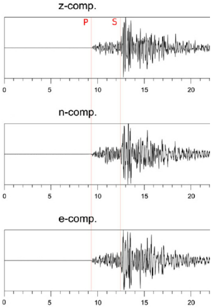
How many stations are necessary to locate an earthquake?

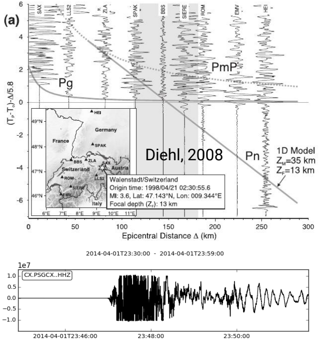
Actually, one can estimate a location with a **single station**, provided that it records three components and both P and S are clearly visible

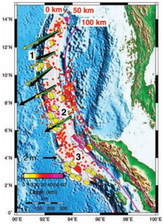
1) (P-S) time difference multiplied by 8 gives estimate of distance (in km)
2) Polarization of P wave gives estimate of direction (first motion?)

But: with conventional approaches (and only P waves), **four stations** are necessary









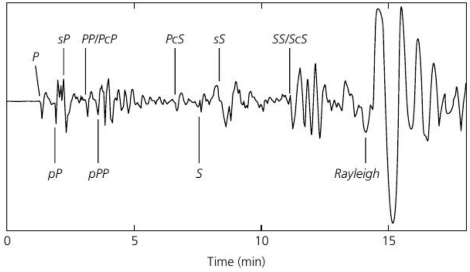
Banerjee et al., 2005

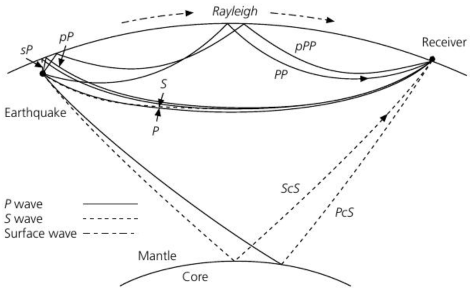
Ingredients 1

- At least four stations that recorded the earthquake
- Ideally P and S arrivals, plus uncertainty estimates
- tricky: P_n vs. P_g...takeover distance, which one comes first?
- if earthquake is large: source time function + instrument clipping

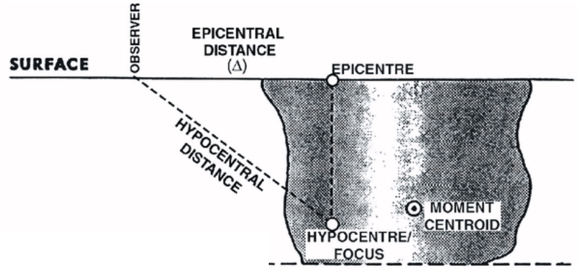
Teleseismic (large) earthquakes

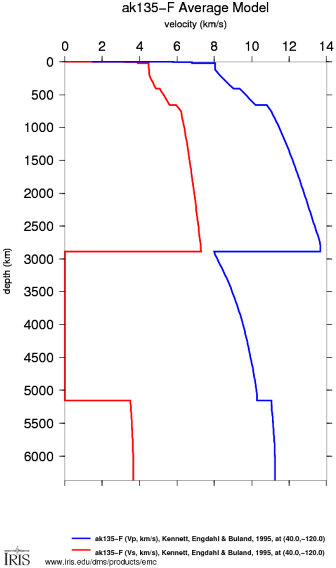
- more phases (can be) visible, which can give additional information
- especially useful: depth phases pP, sP etc.
- these may, however, be hard to identify at times
- lower-frequent arrivals → higher picking uncertainties



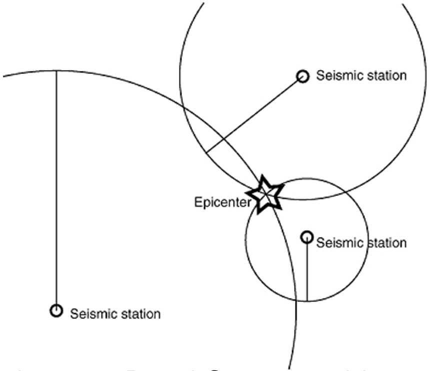


Hypocentral location uses first arrivals
→ systematic difference to centroid location that is e.g. supplied from moment tensor inversion (globalCMT and others).
Difference between hypocenter and centroid:





- Seismic wavespeed models are key for translating traveltimes into distances
- earth is - to first order - horizontally layered; global 1-D models (ak135, iasp91)
- Second order: heterogeneity exists in all directions $\rightarrow$ using 2-D or 3-D models (if available) is more appropriate
- such models are usually produced with seismic tomography, but most location routines only allow 1D models



- rough rule of thumb: time between P and S multiplied by 8 $\sim$ distance to earthquake (in km)
- allows a simple graphical location if more than 3 stations exist
- overly simple approach, which neglects a lot of things (like what?)

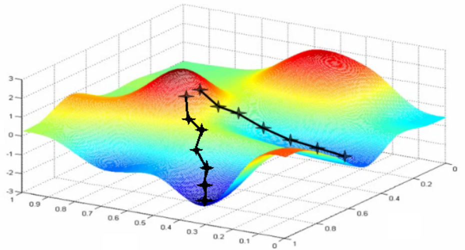
- circle excercise was state-of-the-art before about 1910
- since then: location by minimizing traveltime residuals (r, see below)
- four unknowns (x,y,z and t) $\rightarrow$ at least four observations needed

Equations

$$t_i^c = T(x_i, y_i, z_i, x_h, y_h, z_h) + t_h \quad (1)$$

$$r_i = t_i^o - t_i^c \quad (2)$$

i - station, o - observed
c - calculated, h - hypocenter



- Simplest solution is grid search over x,y,z,t (slow)
- better: iterative techniques (e.g. gradient descent)

$$t_{obs} = t_0 + \int_{x_{src}}^{x_{stat}} u(\mathbf{x}) ds \tag{3}$$

$\mathbf{x}$: vector sampling ray path; $u(\mathbf{x})$: slowness model (1/velocity); t_0: origin time

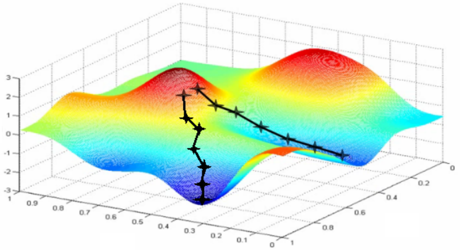
Non-linear problem: change in source location changes ray path and hence integral

$\rightarrow$Coupled velocity-hypocenter problem

Solution(s): linearization via Taylor expansion, or explicit treatment of non-linearity

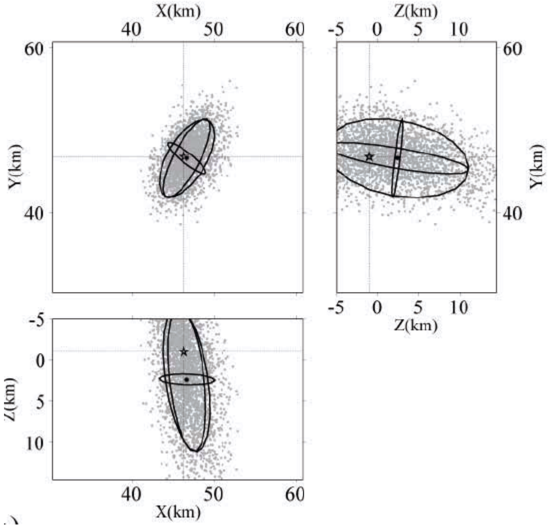
In both cases: iterative process is necessary

1. Take current (starting) location, calculate theoretical traveltimes to all stations (raytracing)
2. Compute traveltime residuals, total (weighted) RMS misfit, and **its spatial derivatives**
3. Take a step (length is derived from additional parameters) into direction of steepest RMS descent
4. Repeat steps 1 to 3 until no further misfit reduction is possible



Problem: local minima (see image) $\rightarrow$ good estimate of starting point is necessary

- Not iteration towards an "optimal" location, but statistical analysis of a large number of forward locations
- Variation of input parameters (e.g. perturbing arrival time picks) leads to "cloud" of hypocenters
- Location defined as mean and standard deviation of this probability density function



- **ISC** (**I**nternational **S**eismological **C**entre) collects earthquake locations and traveltimes from agencies and researchers worldwide, publishes and curates a giant catalog
- http://www.isc.ac.uk/iscbulletin/
- Raw bulletin vs. reviewed
- How many earthquakes are contained? Ever increasing rate, raw bulletin now close to 500k/year, reviewed more like 60k/year
- Reviewed bulletin is usually a couple of years “behind”
- alternative NEIC PDE - somewhat smaller but more up-to-date: https://earthquake.usgs.gov/earthquakes/search/

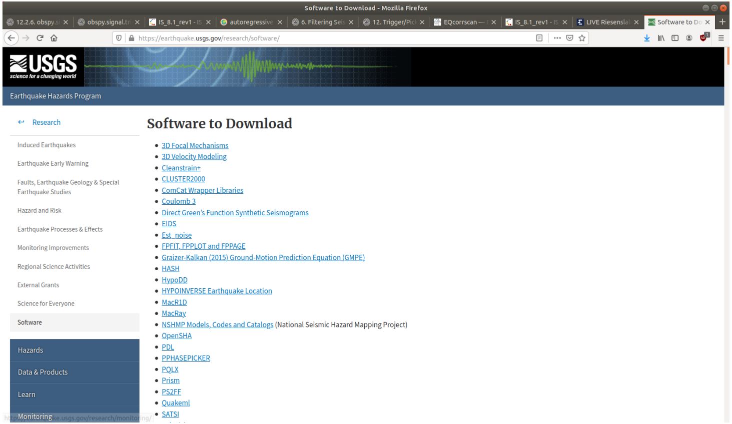
Últimos sismos

con magnitud igual o superior a 3.0 (1)

Fecha Local	Fecha UTC	Latitud	Longitud	Profundidad [Km]	Magnitud	Agencia	Referencia Geográfica
2019/12/17 07:26:26	2019/12/17 10:26:26	-24.002	-66.814	215.0	3.2 Ml	GUC	119 km al E de Socaire
2019/12/16 18:04:22	2019/12/16 21:04:22	-22.854	-69.247	94.6	3.1 Ml	GUC	9 km al NE de Sierra Gorda.
2019/12/16 17:30:36	2019/12/16 20:30:36	-28.352	-70.230	105.4	3.2 Ml	GUC	52 km al NE de Alto del Carmen
2019/12/16 09:43:13	2019/12/16 12:43:13	-23.849	-69.772	68.6	3.7 Ml	GUC	58 km al S de Baquedano
2019/12/16 06:35:13	2019/12/16 09:35:13	-30.669	-71.692	36.9	3.4 Ml	GUC	45 km al O de Punitaqui
2019/12/16 05:13:38	2019/12/16 08:13:38	-23.727	-66.839	205.8	4.1 Ml	GUC	108 km al E de Socaire
2019/12/16 03:18:44	2019/12/16 06:18:44	-22.935	-68.846	99.0	3.0 Ml	GUC	48 km al E de Sierra Gorda.
2019/12/16 02:55:06	2019/12/16 05:55:06	-23.874	-67.146	239.9	3.7 Ml	GUC	82 km al E de Socaire
2019/12/16 02:28:46	2019/12/16 05:28:46	-32.661	-70.195	127.7	3.1 Ml	GUC	42 km al NE de Los Andes.
2019/12/16 01:06:18	2019/12/16 04:06:18	-21.207	-68.531	135.2	3.3 Ml	GUC	29 km al O de Ollagüe
2019/12/15 23:55:22	**2019/12/16 02:55:22**	**-34.818**	**-71.347**	**65.2**	**3.2 Ml**	**GUC**	**20 km al S de Santa Cruz**
2019/12/15 23:16:16	2019/12/16 02:16:16	-23.786	-66.866	190.5	4.0 Ml	GUC	107 km al E de Socaire
2019/12/15 18:11:18	2019/12/15 21:11:18	-18.379	-69.343	134.9	3.1 Ml	GUC	31 km al SE de Putre
2019/12/15 15:42:09	2019/12/15 18:42:09	-24.166	-67.477	203.2	5.0 Mw	GUC	77 km al SE de Socaire
2019/12/15 14:39:10	2019/12/15 17:39:10	-18.087	-69.935	130.4	3.1 Ml	GUC	34 km al E de Tacna

GUC (Geofísica Universidad de Chile): Sigla con la cual es conocido el Centro Sismológico Nacional de la Universidad de Chile en la Red Sismológica Mundial.
(1) En caso de Sismos Sensibles serán publicados cualquiera sea su magnitud.
(2) Sólo los sismos sensibles (reportados), tienen informe de intensidades.
(3) Sólo los sismos sensibles se muestran en negrita.

http://sismologia.cl/links/ultimos_sismos.html
Catalog (∼100,000 events since 2000) is available upon request



USGS software page is a good starting point:
https://earthquake.usgs.gov/research/software/

Hypo71: Basic iterative linearized location routine; only suitable for local problems (uses Cartesian coordinate system)
Old but still extensively used (USGS code)
Necessary inputs:

- velocity model (P and S)
- Starting depth
- Distance weighting parameters
- Station locations and elevations
- Picks with quality classes

Returns: location, standard error, residuals, gap, dmin, etc.

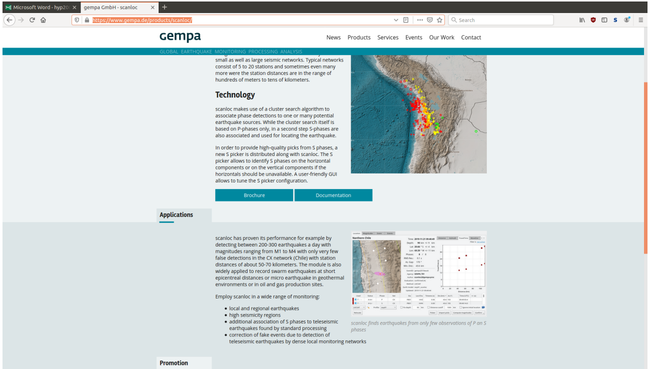
Hypoellipse, Hypoinverse: Successors of hypo71 (all from USGS)
Same principle as HYPO71, but some additional features:

- Gradient velocity models and usage of different 1D models by region is possible
- Magnitude estimation is included
- Elimination of earthquakes after certain criteria

Can be retrieved from USGS webpage (see a few slides previous).

Scanloc: Commercial software from gempa (same company as seiscomp3); mostly intended for monitoring purposes using live data

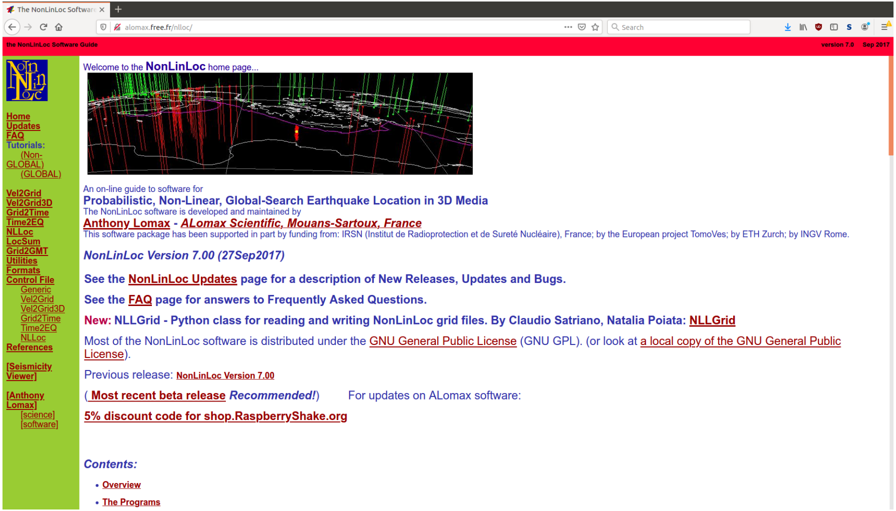
Includes pickers, association and location, which can be tuned



http://www.gempa.de/products/scanloc/creenshot

NonLinLoc: Probabilistic approach that takes non-linearity into account

- Uncertainty estimates for picks, velocity model etc. are input, program uses them as σ in a normal distribution
- Location problem is solved with an iterative method, but many many times (for different input perturbations)
- Evaluation of the resulting PDF (probability density function) yields error estimates and maximum-likelihood location



http://alomax.free.fr/nlloc/

Exercises

7. Location with hypo71

Familiarize yourself with the different input files and output files of hypo71. Run the code as is to locate an example event. Then try to manipulate some of the input values (e.g. starting depth) and see how the resulting location changes.

In hypo71.py, you will find some functions that can read picks from a file, write the appropriate hypo71 input and read its output. With the help of these functions, try to write a routine that sequentially locates all the events provided in picks.txt.
If there is time left, we can try to apply the same routine to the short catalog of events we created yesterday.

- Question: what can contribute to errors in hypocenter location?

- Question: what can contribute to errors in hypocenter location?
- Uncertainty in arrival time picks

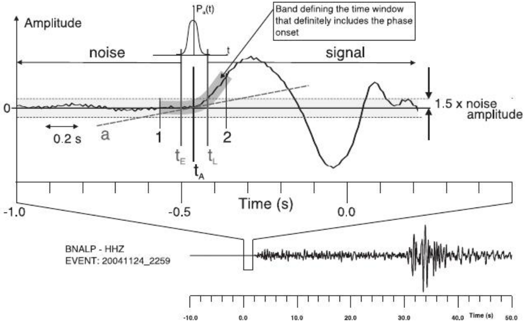
- Question: what can contribute to errors in hypocenter location?
- Uncertainty in arrival time picks
- Phase misidentification

- Question: what can contribute to errors in hypocenter location?
- Uncertainty in arrival time picks
- Phase misidentification
- “Resolution” (event-station geometry, number and type of picks etc.)

- Question: what can contribute to errors in hypocenter location?
- Uncertainty in arrival time picks
- Phase misidentification
- "Resolution" (event-station geometry, number and type of picks etc.)
- Velocity difference between assumed velocity model and reality

- Question: what can contribute to errors in hypocenter location?
- Uncertainty in arrival time picks
- Phase misidentification
- "Resolution" (event-station geometry, number and type of picks etc.)
- Velocity difference between assumed velocity model and reality
- Errors in algorithm (inversion or determining theoretical traveltimes)

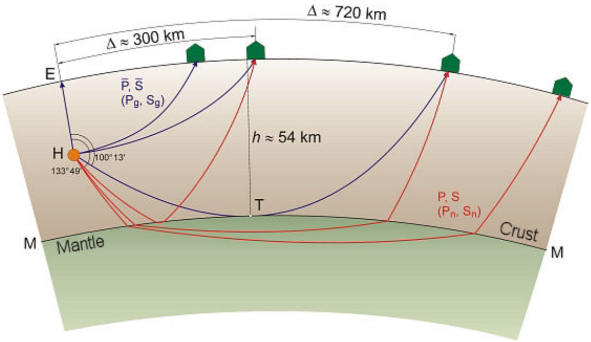
Arrival time picks are usually given together with a "quality class" that scales with uncertainty

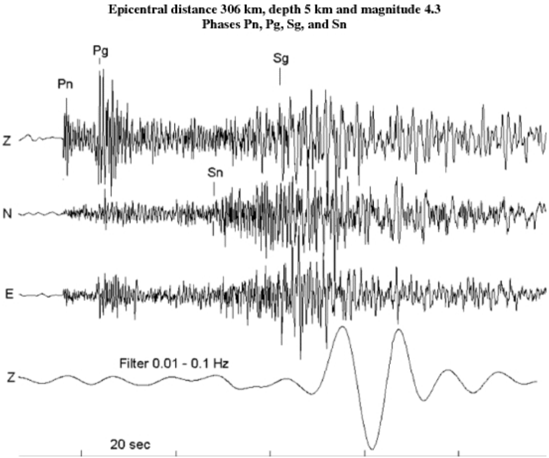


Most location algorithms translate these quality classes into weighting factors

Usual assumption: picking errors are random

Problem: this does not take into account phase misidentifications, e.g. Pn vs Pg

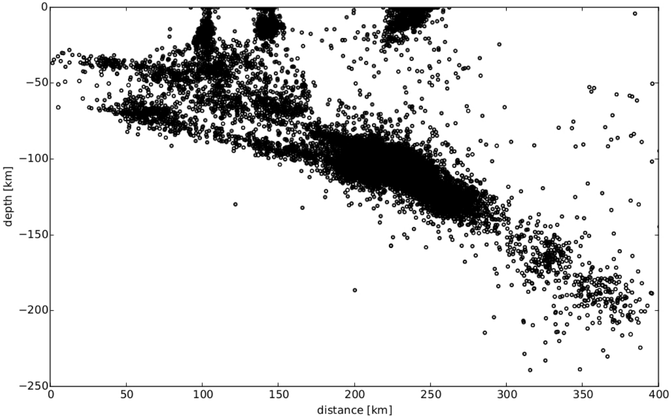




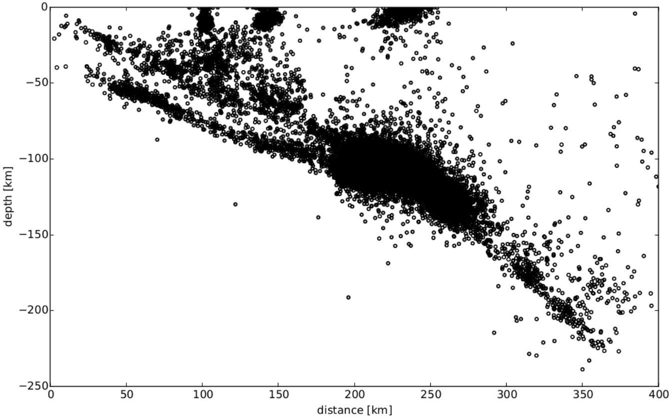
Dangerous when systematic because it means uncertainty is way outside of provided estimates

Most codes expect first arrivals, i.e. P_n/S_n beyond overtaking distance

"True" velocity model is unknowable, but velocity models influence resulting geometries



"True" velocity model is unknowable, but velocity models influence resulting geometries



Same picks/stations, different velocity model (1D vs. 2D)

Location accuracy should be a function of...

- **Geometry between hypocenter and station network**: Hypocenter in center/periphery/outside of network $\rightarrow$ best/OK/bad locations
- **Station density**: Dense network of close stations is better than sparse network of distant stations
- **Number and type of observations**: More and variable is better
- **Velocity model**: 3D or 2D is better than 1D, local is better than global

Location accuracy should be a function of...

- **Geometry between hypocenter and station network**: Hypocenter in center/periphery/outside of network → best/OK/bad locations
- **Station density**: Dense network of close stations is better than sparse network of distant stations
- **Number and type of observations**: More and variable is better
- **Velocity model**: 3D or 2D is better than 1D, local is better than global

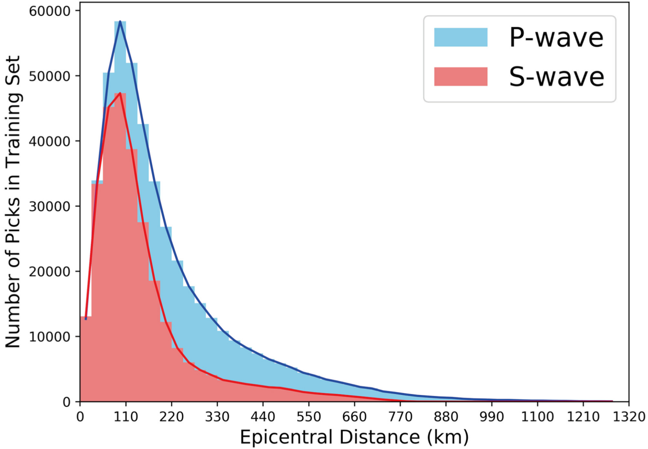
Location accuracy should be a function of...

- **Geometry between hypocenter and station network**: Hypocenter in center/periphery/outside of network → best/OK/bad locations
- **Station density**: Dense network of close stations is better than sparse network of distant stations
- **Number and type of observations**: More and variable is better
- **Velocity model**: 3D or 2D is better than 1D, local is better than global

Location accuracy should be a function of...

- **Geometry between hypocenter and station network**: Hypocenter in center/periphery/outside of network $\rightarrow$ best/OK/bad locations
- **Station density**: Dense network of close stations is better than sparse network of distant stations
- **Number and type of observations**: More and variable is better
- **Velocity model**: 3D or 2D is better than 1D, local is better than global

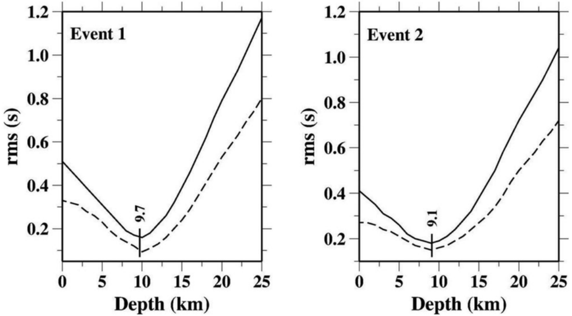
Number of observations:



Mixed P and S is better than only P (tradeoff origin time-distance)

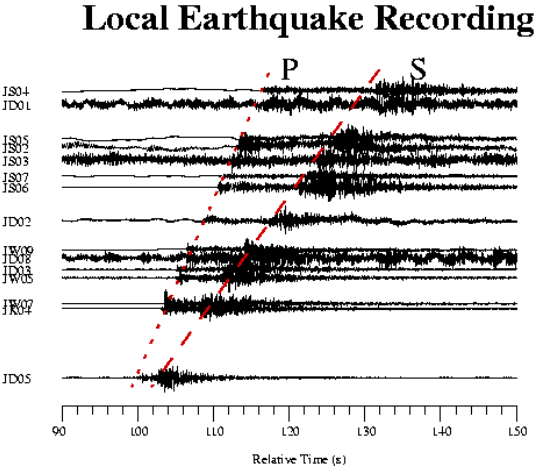
S-P difference is much better suited for hypocentral depth determination

RMS Residual:
sum of squared station (pick) residuals in s (sometimes weighted according to distances etc)
rule of thumb: RMS higher than 1-2 s is suspicious (but this depends hugely on network size)



Often: events above a certain RMS are thrown away; but: it is often better to evaulate station residuals!
plus: low RMS is easier to achieve with few constraining picks (careful about internal weightings)

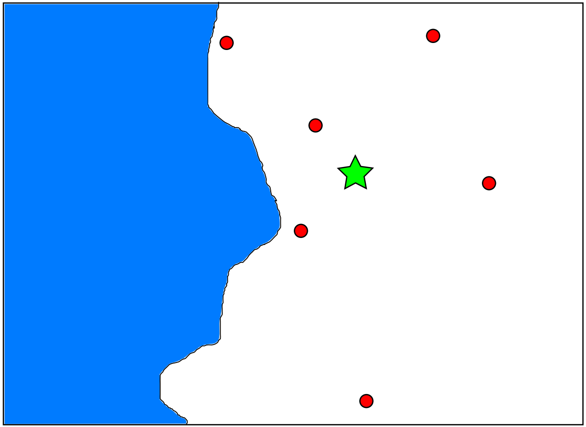
Closest station: For really well-constrained hypocentral depth, having a station with P and S pick within 1 hypocentral depth is a “gold standard”



Gap criterion: GAP $=$ largest angle between station azimuths (only stations with picks)

GAP $< 180^\circ$ means event is within network, GAP $> 180^\circ$ means outside network

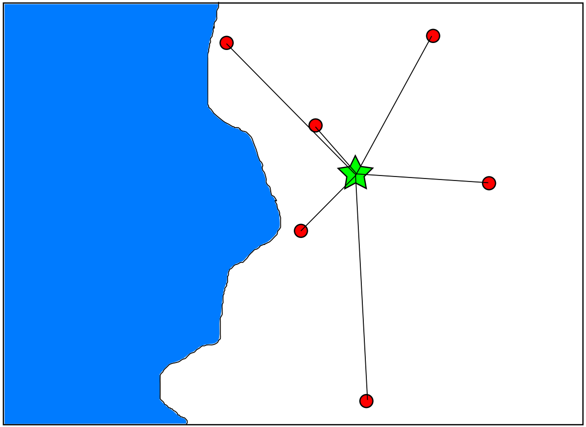
Important measure, but neglects 3D "gap" (different conditions for shallow and deep EQ)



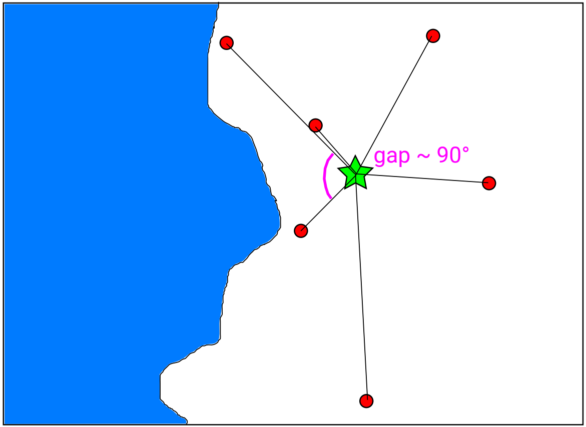
Gap criterion: GAP = largest angle between station azimuths (only stations with picks)

GAP $< 180^\circ$ means event is within network, GAP $> 180^\circ$ means outside network

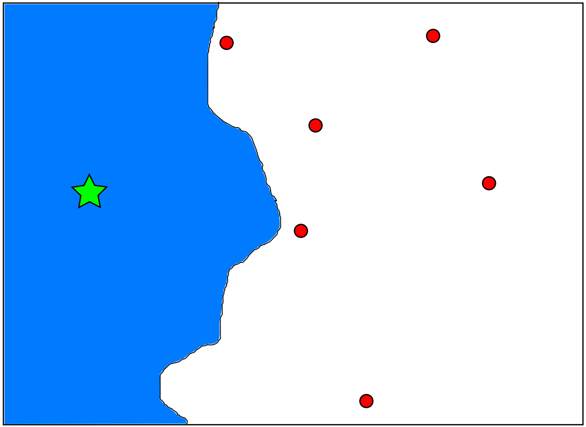
Important measure, but neglects 3D "gap" (different conditions for shallow and deep EQ)



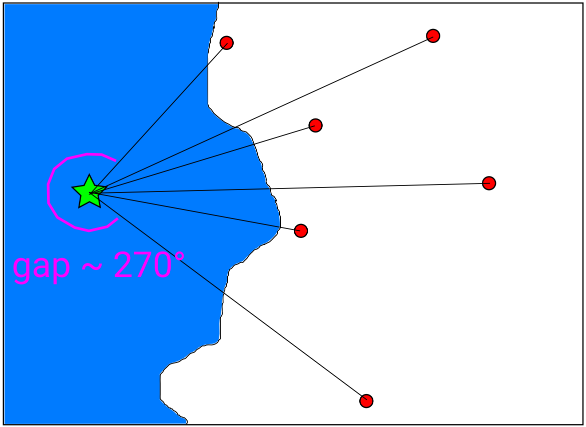
Gap criterion: GAP = largest angle between station azimuths (only stations with picks)
GAP $< 180^\circ$ means event is within network, GAP $> 180^\circ$ means outside network
Important measure, but neglects 3D "gap" (different conditions for shallow and deep EQ)



Gap criterion: GAP = largest angle between station azimuths (only stations with picks)
GAP $< 180^\circ$ means event is within network, GAP $> 180^\circ$ means outside network
Important measure, but neglects 3D "gap" (different conditions for shallow and deep EQ)



Gap criterion: GAP = largest angle between station azimuths (only stations with picks)
GAP $< 180^\circ$ means event is within network, GAP $> 180^\circ$ means outside network
Important measure, but neglects 3D “gap” (different conditions for shallow and deep EQ)



Ideal case: ground truth exists and can be used to check location errors

Ground truth

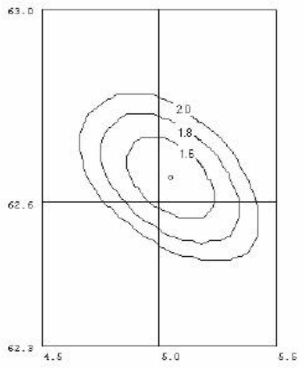
ground truth = controlled explosion, mining blast, etc., where correct origin time and location is known

This allows detemining actual location error; but: these sources are only available at or close to the surface!!

If not available: use of uncertainty estimation

Problem: Influence of input uncertainty (picks) and event-station geometry can be checked; **but**:systematic errors (e.g. from wrong velocity model) can not be assessed!!

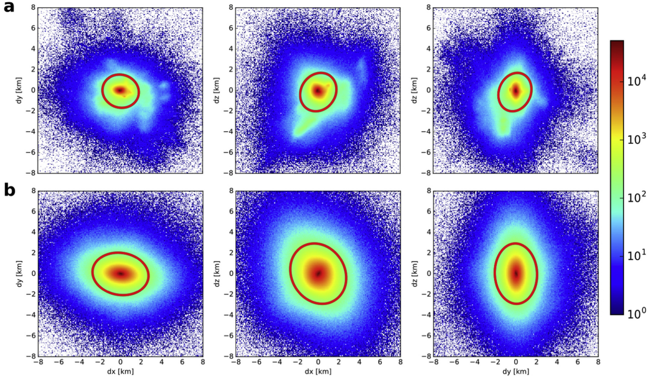
- Best case: ground truth is available (e.g. explosions)
- "Standard errors" are often given in earthquake catalogs; these usually derive from sharpness of RMS minimum; can severely underestimate true location uncertainties
- A better way: probabilistic approach



Bootstrapping: Statistical resampling of input traveltimes (e.g. random perturbation of all traveltimes with draw from normal distribution, std = quality estimate)

Jackknifing: evaluating dependence of solution on single station(s) by randomly leaving stations out

Many locations are performed, resulting in 3D point cloud (σ or 2σ taken as error ellipsoid)

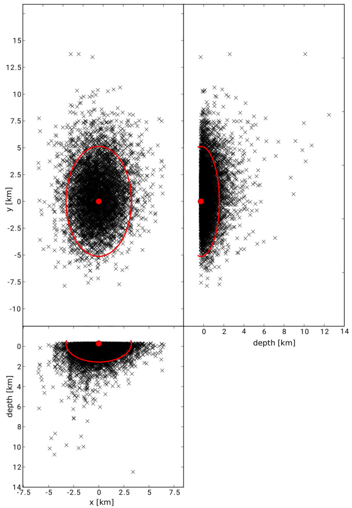


- Standard error only depend on width of the minimum, not on input uncertainties
- Comparison between standard errors and bootstrapping/jackknifing shows that standard errors often too small, especially in cases with high uncertainties
- Thus: one should not rely on standard errors alone!

Picture oae?

NonLinLoc (probabilistic approach)

- Picks and velocity model have uncertainty; Gaussian distribution
- Computation of a large amount of locations for different combinations of errors
- Output: point cloud or "probability density function" of single locations
- Definition of error ellipsoid as standard deviation of this point cloud (68%)
- More realistic errors, although systematically wrong velocity model can still be underestimated



Absolute vs. relative errors

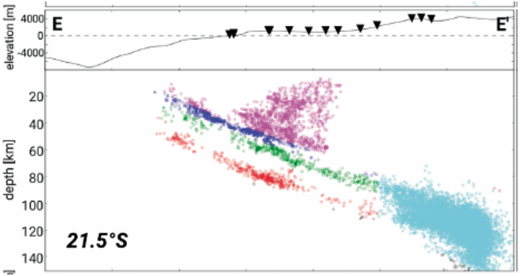
Absolute error = difference between determined location and real location

Relative error = difference between determined and real distance between event pairs

→ Relative error does not contain systematic bias (e.g. from velocity model usage)!

Depending on the goal of the study, absolute or relative error can be more important

Relative error can be further reduced with multi-event and relative location techniques (tomorrow)



1. Depth nearly always has highest uncertainty (except events far outside network); epicenters are more robust
2. Comparison of different locations of same event can be very instructive (e.g. teleseismic vs. local locations)
3. Never trust one number (e.g. standard error) alone; combining multiple methods to estimate location uncertainty is best
4. Keep in mind whether absolute or relative errors are relevant for the problem/interpretation at hand

1. Depth nearly always has highest uncertainty (except events far outside network); epicenters are more robust
2. Comparison of different locations of same event can be very instructive (e.g. teleseismic vs. local locations)
3. Never trust one number (e.g. standard error) alone; combining multiple methods to estimate location uncertainty is best
4. Keep in mind whether absolute or relative errors are relevant for the problem/interpretation at hand

1. Depth nearly always has highest uncertainty (except events far outside network); epicenters are more robust
2. Comparison of different locations of same event can be very instructive (e.g. teleseismic vs. local locations)
3. Never trust one number (e.g. standard error) alone; combining multiple methods to estimate location uncertainty is best
4. Keep in mind whether absolute or relative errors are relevant for the problem/interpretation at hand

1. Depth nearly always has highest uncertainty (except events far outside network); epicenters are more robust
2. Comparison of different locations of same event can be very instructive (e.g. teleseismic vs. local locations)
3. Never trust one number (e.g. standard error) alone; combining multiple methods to estimate location uncertainty is best
4. Keep in mind whether absolute or relative errors are relevant for the problem/interpretation at hand

Exercises

8. Hypo71 Error estimates

Try to apply bootstrapping and jackknifing to the hypo71 locations performed in the morning. This means that we should

a) randomly remove single stations and relocate

b) randomly perturb travel times and relocate

Do this for a single earthquake first, and locate 500 different realizations.Take the original hypocenter as 0 and plot the resulting point clouds in km relative to this origin. Determine a formal error estimate by computing the standard deviations and compare this to the standard errors hypo71 outputs.

Day 3: Goals

1. You have acquired basic knowledge about how earthquake location programs work
2. You have a good understanding about the sources of location uncertainty, and how it can be estimated
3. You have gained experience in using earthquake location codes, and judging how reliable the output may be

Tomorrow: multi-event location and use of cross-correlations