GMT

Generic Mapping Tools or
Gravity, Magnetics and Topography

Lecture # 2

Gridding, Data Analysis, and Processing

Working with GMT

- Used to write "C" language programs
- Now I write UNIX scripts
- Use UNIX and GMT commands to process data
- Use GMT commands to put out intermediate (QC) and final output
- With a set directory file structure, most kinds of data processing can be automated

Data Processing

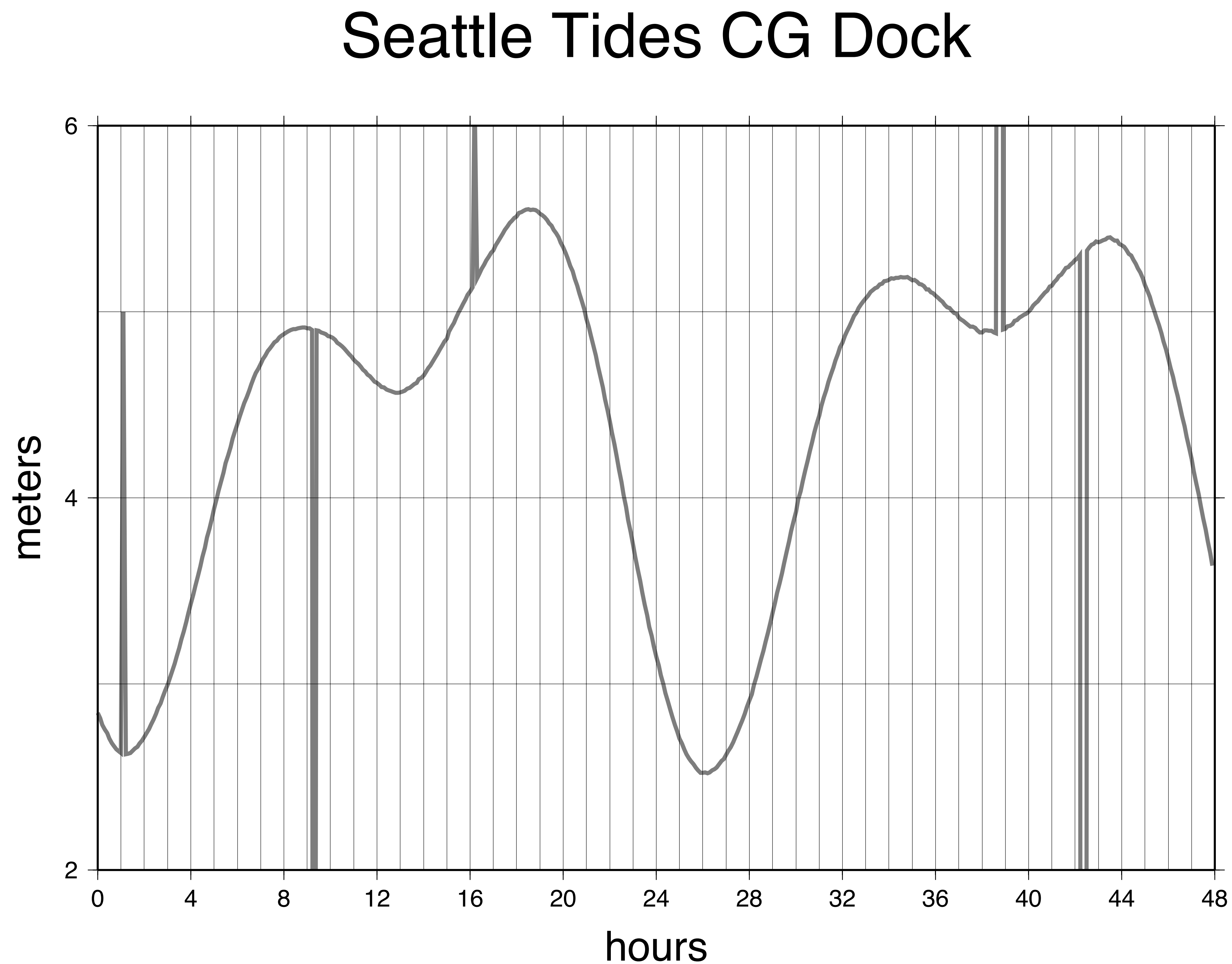
- ★ sample1d Resampling of 1-D data
- ★ filter1d Filter 1-D data (time series)

- ★ fitcircle Finds best-fitting great or small circles
- ★ grdtrend Fits polynomial trends to grdfiles (z = f(x,y))
- ★ trend1d Fits polynomial or Fourier trends to y = f(x) series
- ★ trend2d Fits polynomial trends to z = f(x,y) series

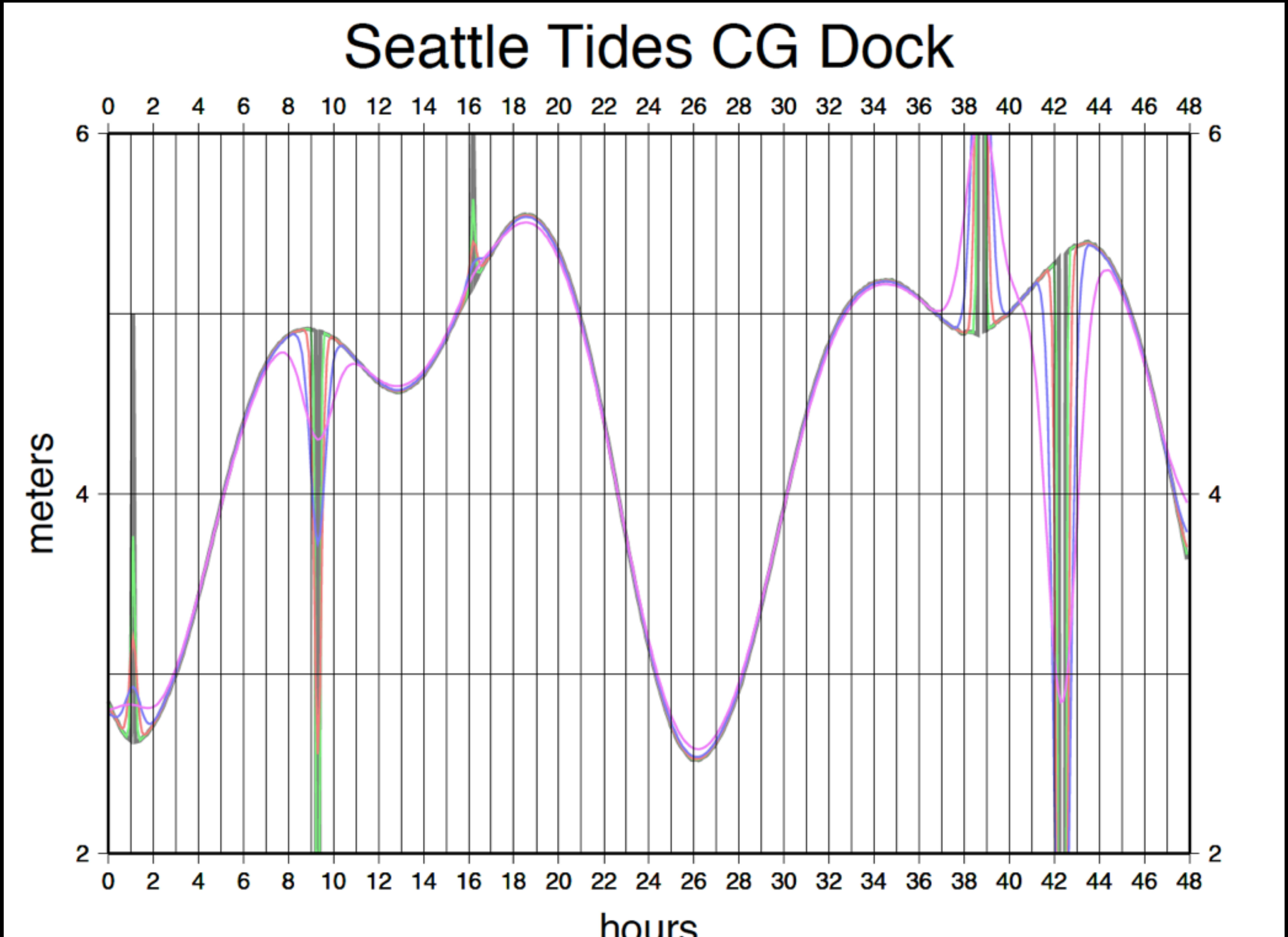
Filtering

–F Sets the filter type. Choose among convolution and non-convolution filters. Append the filter code followed by the full filter *width* in same units as time column. Available convolution filters are:

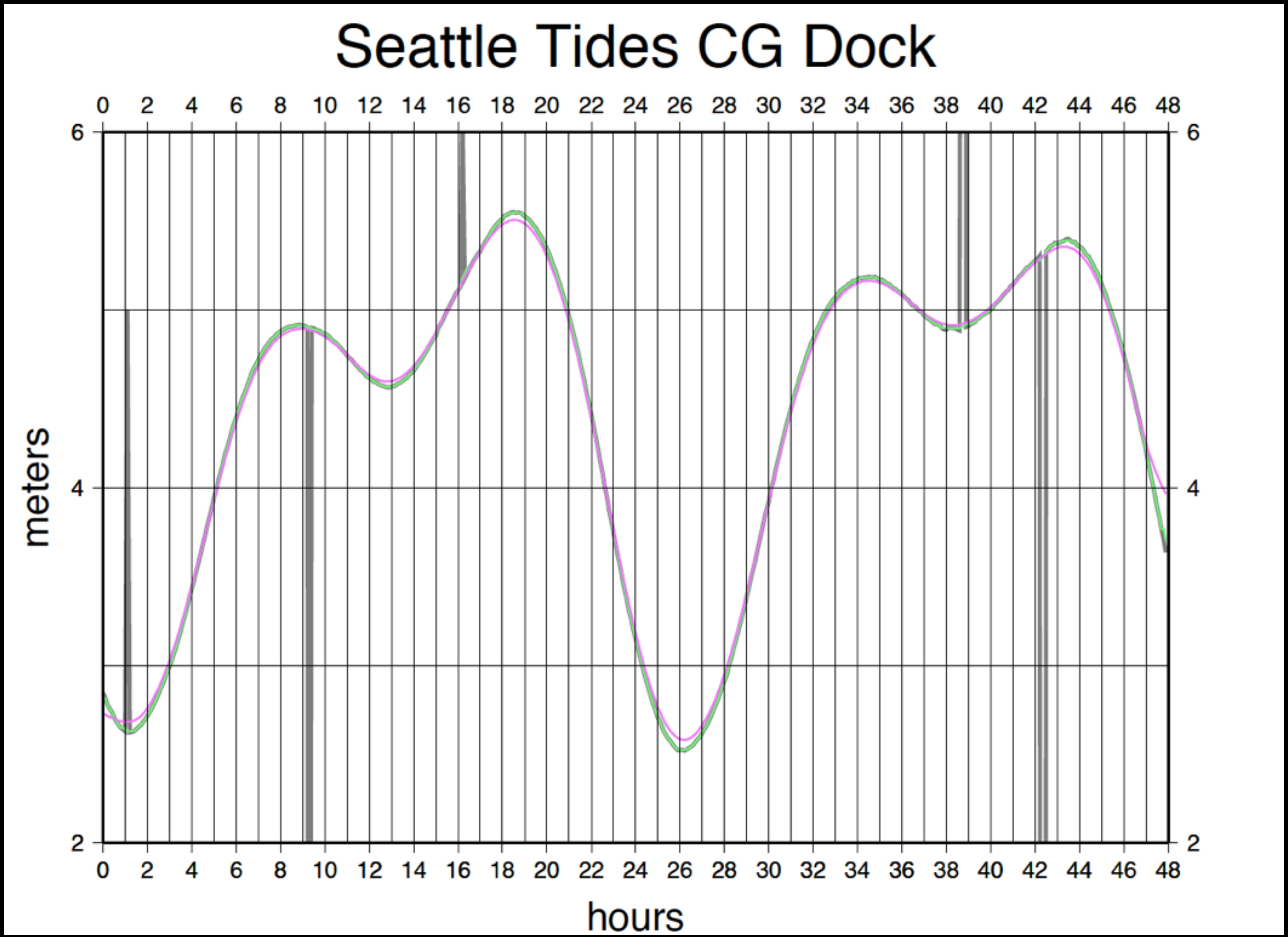
(**b**) Boxcar: All weights are equal.
(**c**) Cosine Arch: Weights follow a cosine arch curve.
(**g**) Gaussian: Weights are given by the Gaussian function.
(**f**) Custom: Instead of *width* give name of a one-column file with your own weight coefficients.
Non-convolution filters are:
(**m**) Median: Returns median value.
(**p**) Maximum likelihood probability (a mode estimator): Return modal value. If more than one mode is found we return their average value. Append - or + to the filter width if you rather want to


Seattle Tides CG Dock
meters
hours
2
4
6
0
4
8
12
16
20
24
28
32
36
40
44
48

Gaussian Filter



Median Filter



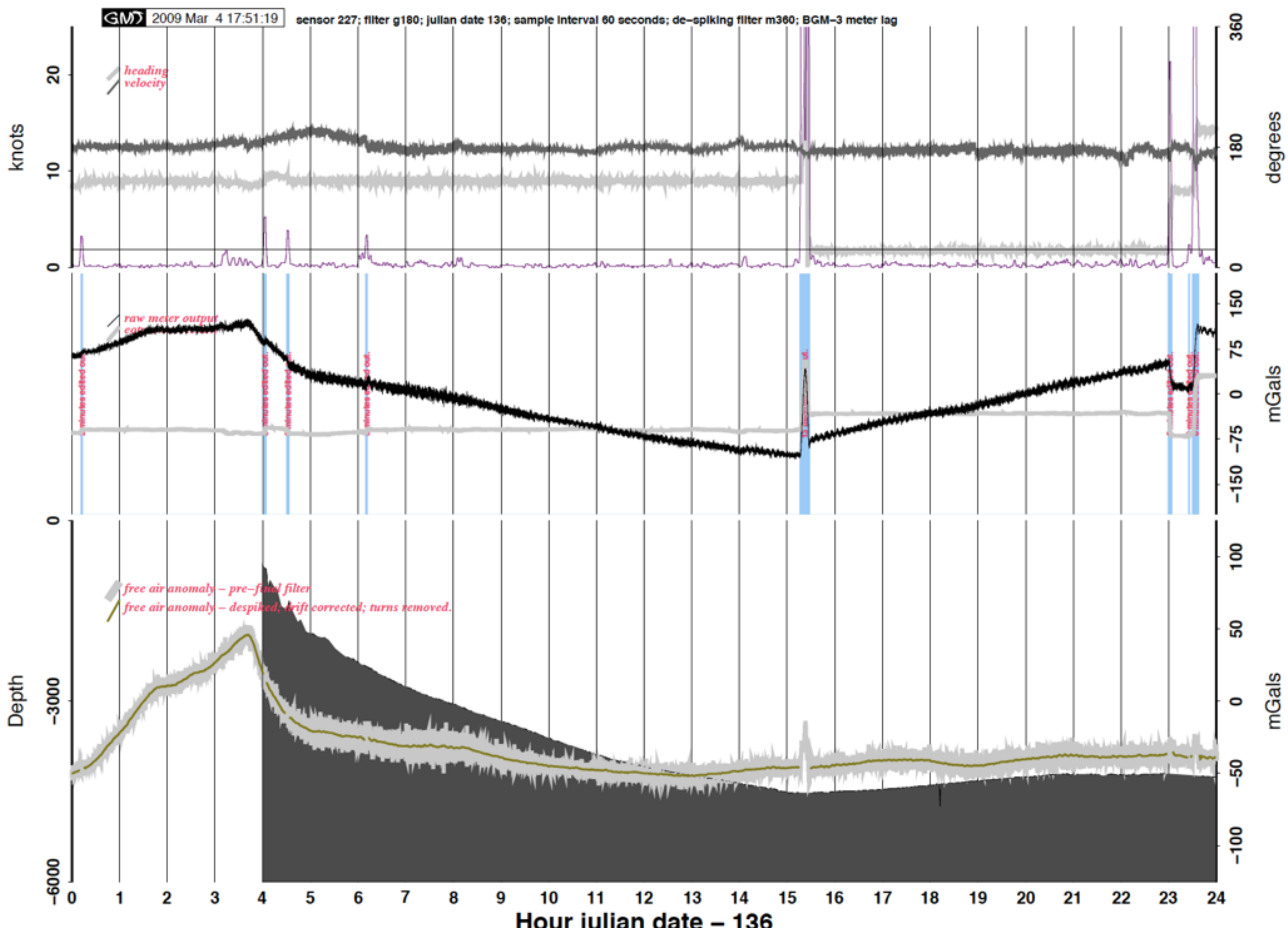
```
#!/bin/csh

# set plotting parameters

set databox = -R0/48/2/6
set scale = -JX9.0i/6.0i

# scale data from timetags to time (hours)
# input format
# 09+019:00:00:00.0000 2.844
# 09+019:00:06:00.0000 2.815
# 09+019:00:12:00.0000 2.780

cat tide.d* | awk '{hour = substr($1,8,2);  minute = substr($1,11,2); second = substr($1,13,2); time = hour+(minute + second/60)/60; if (NR > 1 && time < old_time+0)
{addhour += 24;} printf"%.2f %.3f\n",time+addhour,$2; old_time=time;}' > hold.data

# plot unfiltered data

cat hold.data | gmt psxy $databox $scale -W1,125 -Bg1a4:hours:/g1a2:meters::."Seattle Tides CG Dock":WeSn -K > filtered1.ps

# plot median (30 minute or half hour) filtered data

cat hold.data | gmt filter1d -Fm0.5 -E | gmt psxy $databox $scale -W1,125,250/125 -Bg1a2 -K -O >> filtered1.ps

# plot median then Gaussian (30 minute or half hour) filtered data

cat hold.data | gmt filter1d -Fm0.5 -E | filter1d -Fg4 -E | gmt psxy $databox $scale -W1,250/125/250 -Bg1a2 -O >> filtered1.ps

# new plot; plot raw data

cat hold.data | gmt psxy $databox $scale -W1,125 -Bg1a4:hours:/g1a2:meters::."Seattle Tides CG Dock":WeSn -K > filtered2.ps

# plot gaussian (0.5, 1, 2, 4 hour filter lengths).

cat hold.data | gmt filter1d -Fg0.5 -E | gmt psxy $databox $scale -W1,125/250/125 -Bg1a2 -K -O >> filtered2.ps

cat hold.data | gmt filter1d -Fg1 -E | gmt psxy $databox $scale -W1,250/125/125 -Bg1a2 -K -O >> filtered2.ps

cat hold.data | gmt filter1d -Fg2 -E | gmt psxy $databox $scale -W1,125/125/250 -Bg1a2 -K -O >> filtered2.ps

cat hold.data | gmt filter1d -Fg4 -E | gmt psxy $databox $scale -W1,250/125/250 -Bg1a2 -O >> filtered2.ps
```

Data Processing

Gravity Example

The basic problem is to join a number of data files, containing bathymetry, raw gravity and navigation data, to reduce the gravity values and estimate anomalies along track.

A UNIX script, using awk and other commands with GMT commands can accomplish this.



GMT 2009 Mar 4 17:51:19
sensor 227; filter g180; julian date 136; sample interval 60 seconds; de-spiking filter m360; BGM-3 meter lag
heading
velocity
knots
degrees
raw meter output
mGals
free air anomaly – pre-final filter
free air anomaly – despiked; drift corrected; turns removed.
Depth
mGals
Hour julian date – 136

Why Grid?

- Regular “data” spacing
- Easy to contour
- Amenable to quantitative analysis
- A consistent representation

How grid?

- Combine irregularly sampled data sets from different events
- Ensure same scaling
- Minimize noise
- Recognize grid resolution limits

Problems

- Poor meta-data (scaling issues)
- Crappy data
- Over and under sampling
- Best interpolant?

Requirements for Gridding

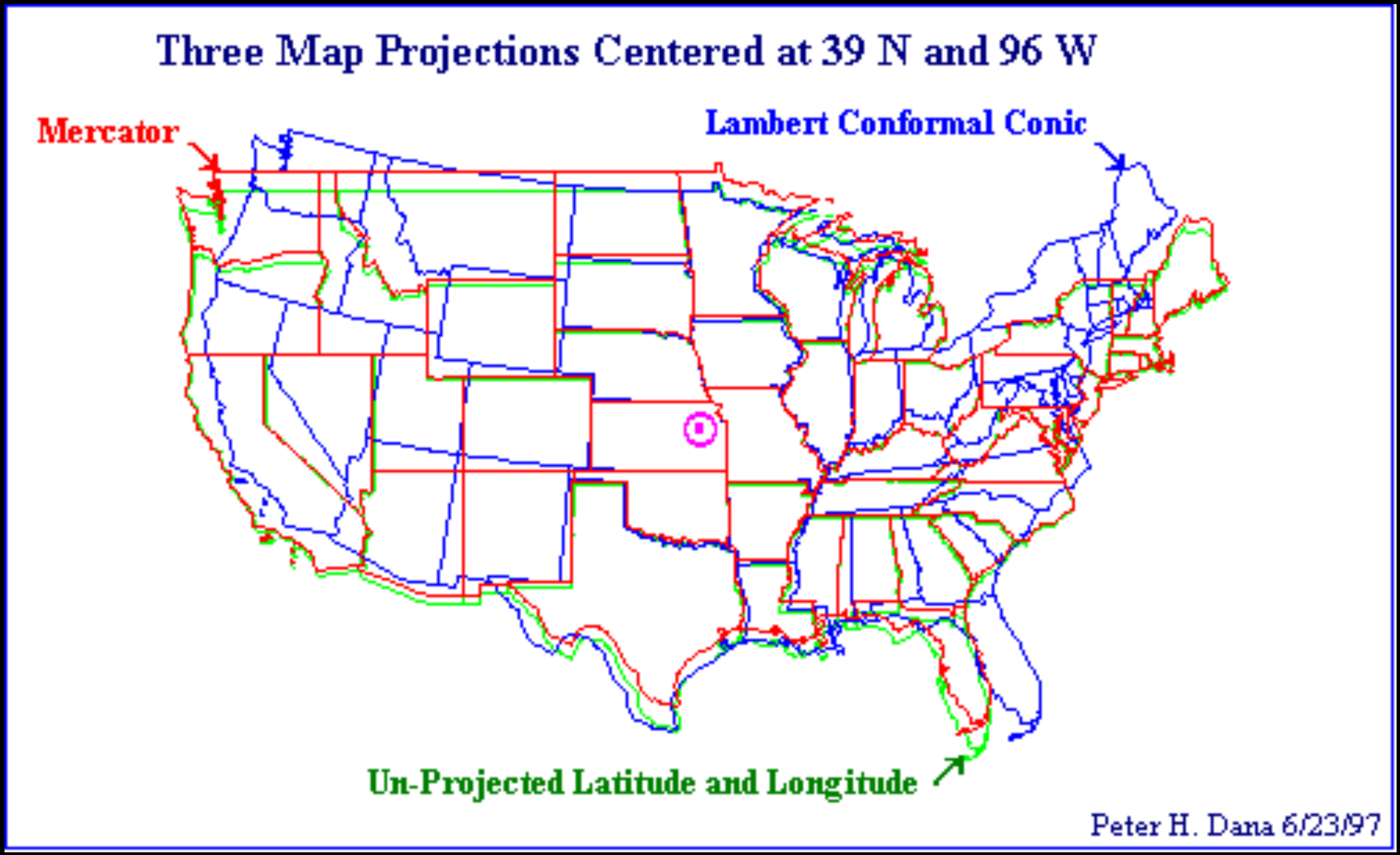
- Continuous Field
- Projected coordinates

  Map projections take the data space, where measurements are made, for example, the curved surface of the earth, and represent it as a flat plane

- Measurements on a common basis

  Analyze measurements to demonstrate they are commensurate.

Map Projections Matter



How Map Projections work.

- Conserve different aspects of the map in representation
- Area - May map projections are designed to be "equal-area", so that an object of any size, on one part of the map covers exactly the same area of the actual Earth as the same object on any other part of the map.
- Shape - In these projections the relative local angles about every point on the map are shown correctly.
- Scale - Though no map projection shows scale correctly throughout the map, there are usually one or more lines on the map where the scale is accurate.
- Direction - Conformal maps give the relative local direction correctly at any given point.
- Special Characteristics - Several map projections provide some special characteristics that no other map projections provides.

Mercator projection has all the lines of constant direction shown as straight lines.

The Gnomonic projection has all the great circle paths which are shown as straight lines.

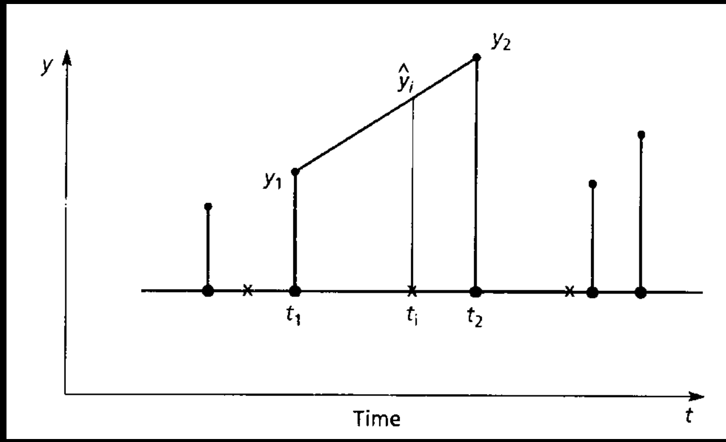
Interpolation

Estimating a map value at a arbitrary point

Interpolation – Estimating value at a new point from points in the neighborhood:

This is useful in one–dimension (transects or time series) or

2 dimensions (maps) or even in 3 dimensions (like in a mine).



There are a variety of interpolation methods available:

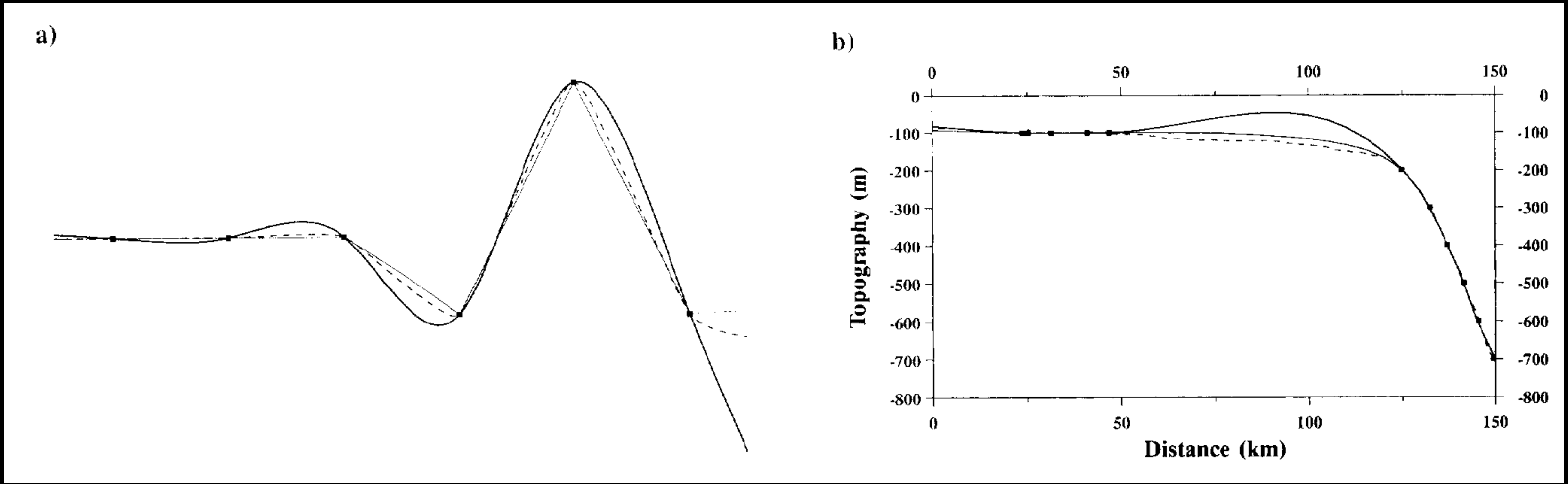
Assign the value at a point to that of the nearest neighbor.

Assign the value to that of the average of some number of close neighbors.

Weight the average by some measure of distance between each point and the unknown.

Somehow take into account all measured points.

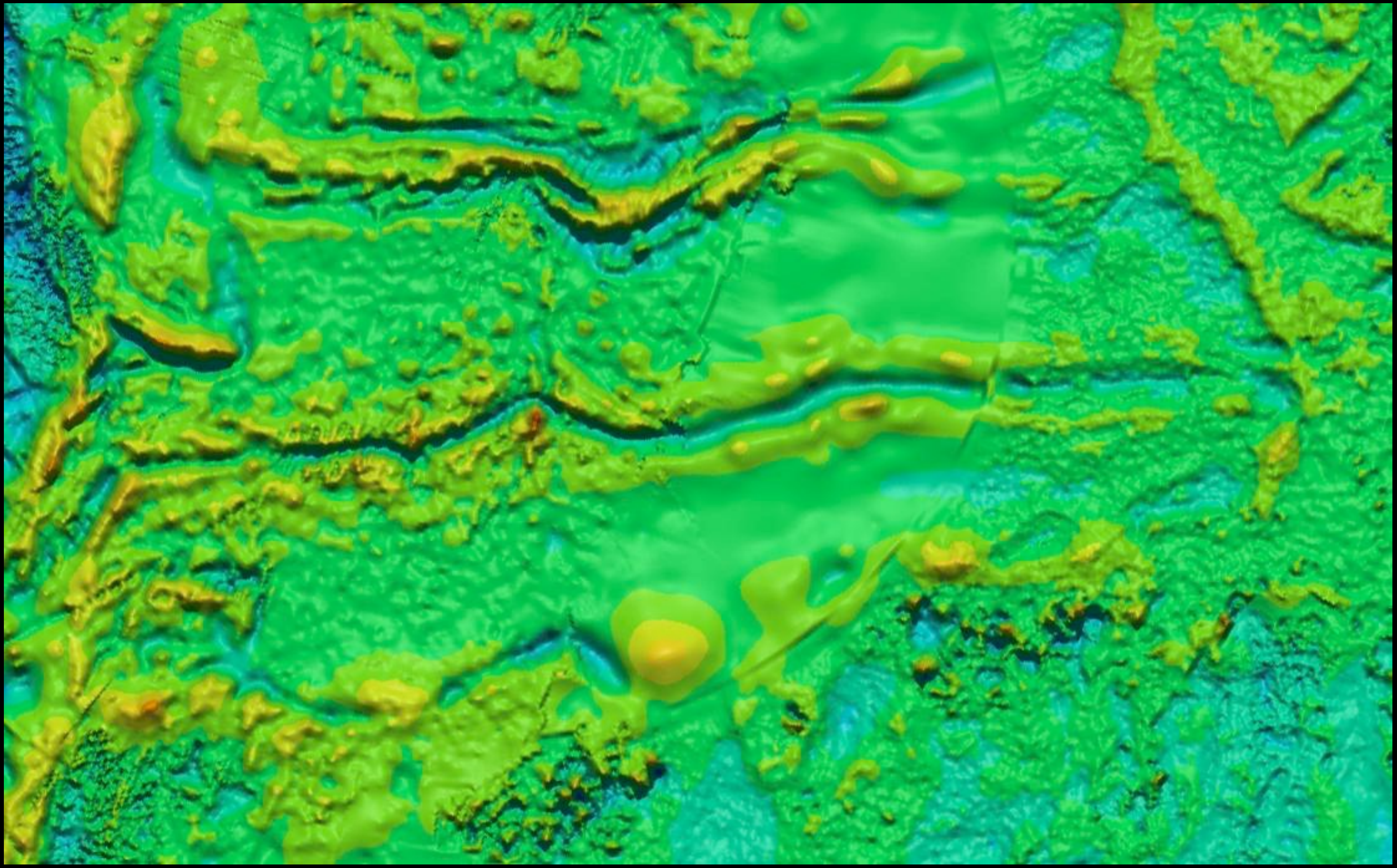
1-D Interpolation



Splines in Tension

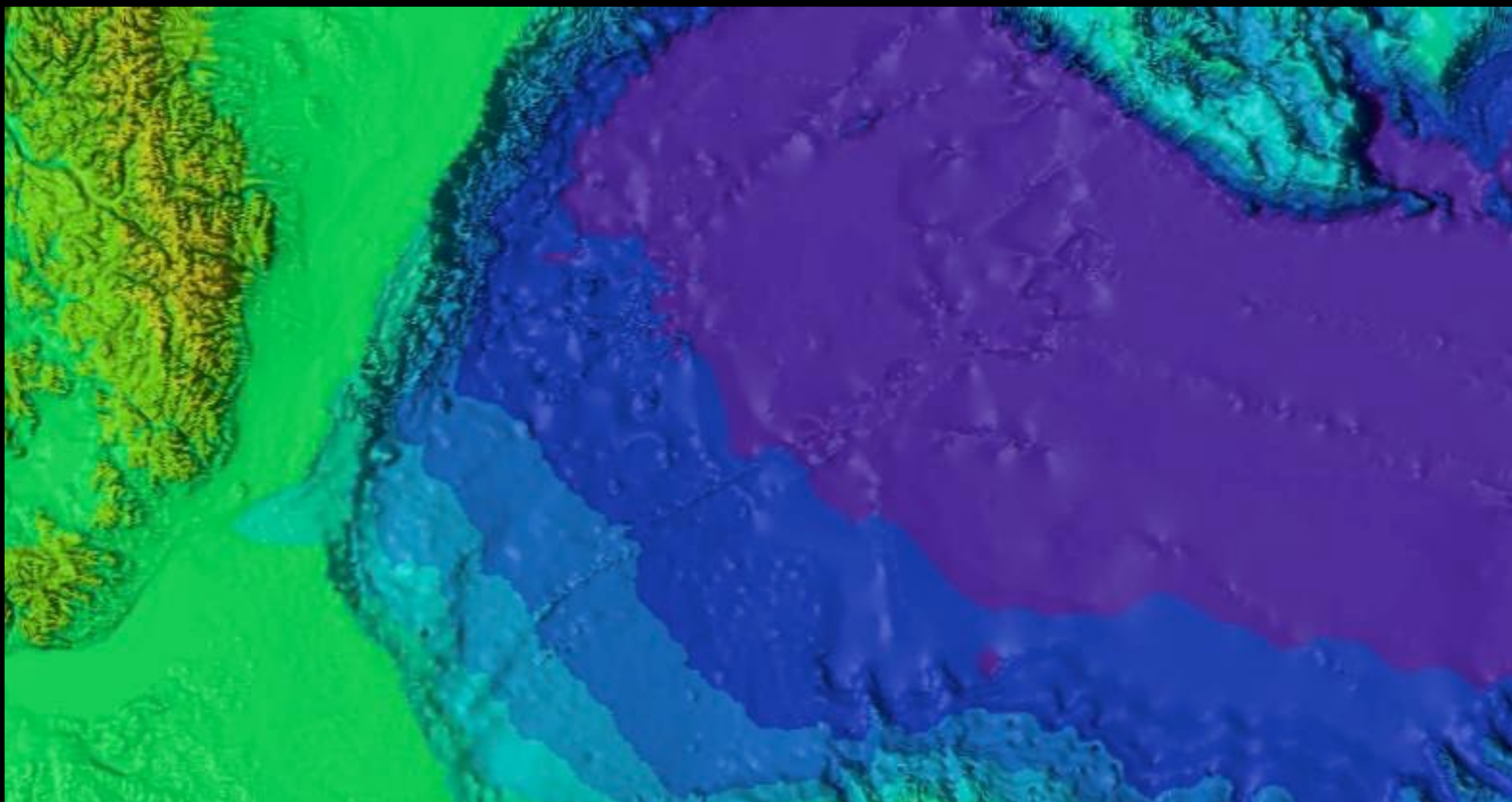
Eurasia Basin

Sampling issues

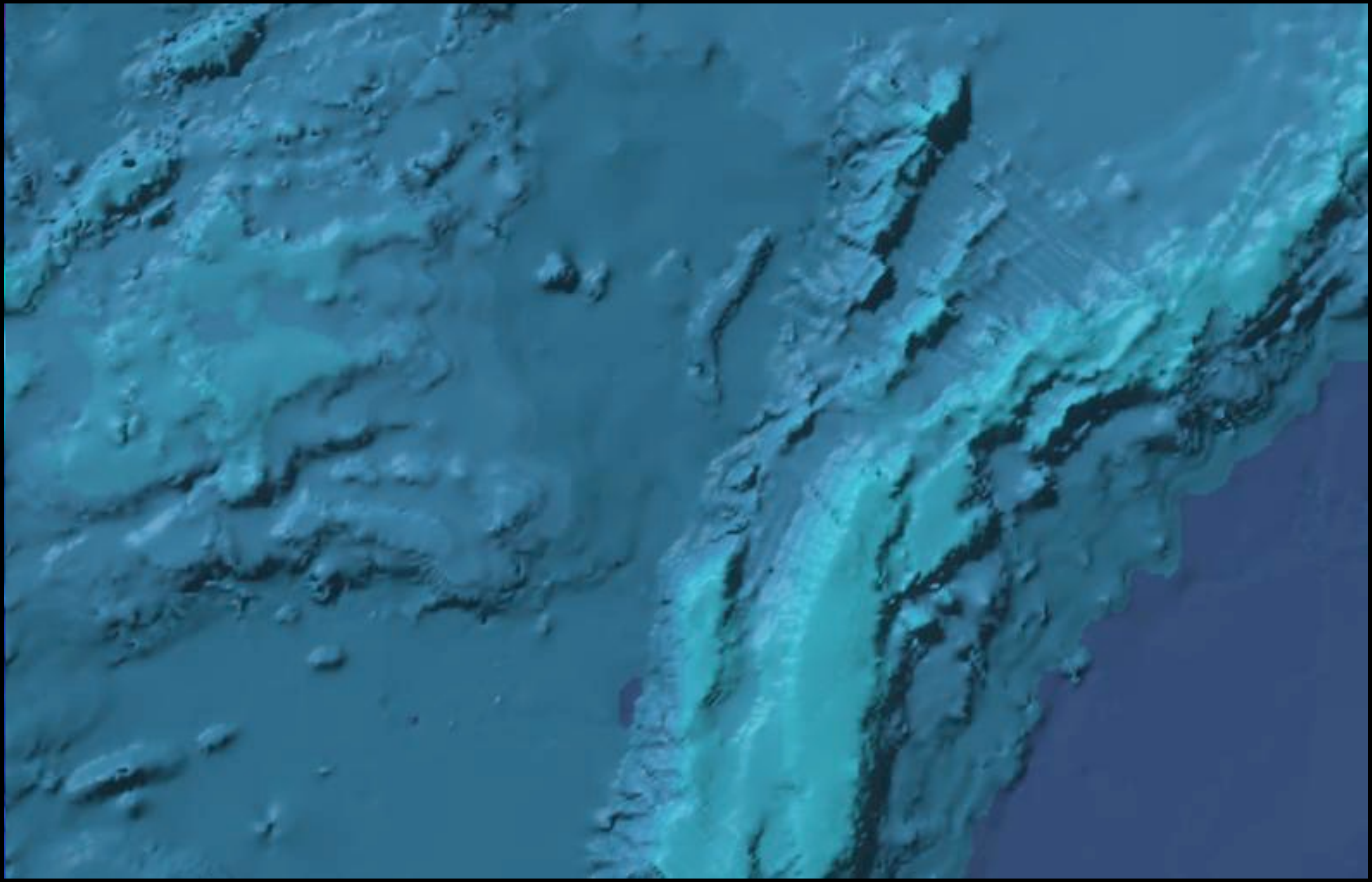


Canada Basin

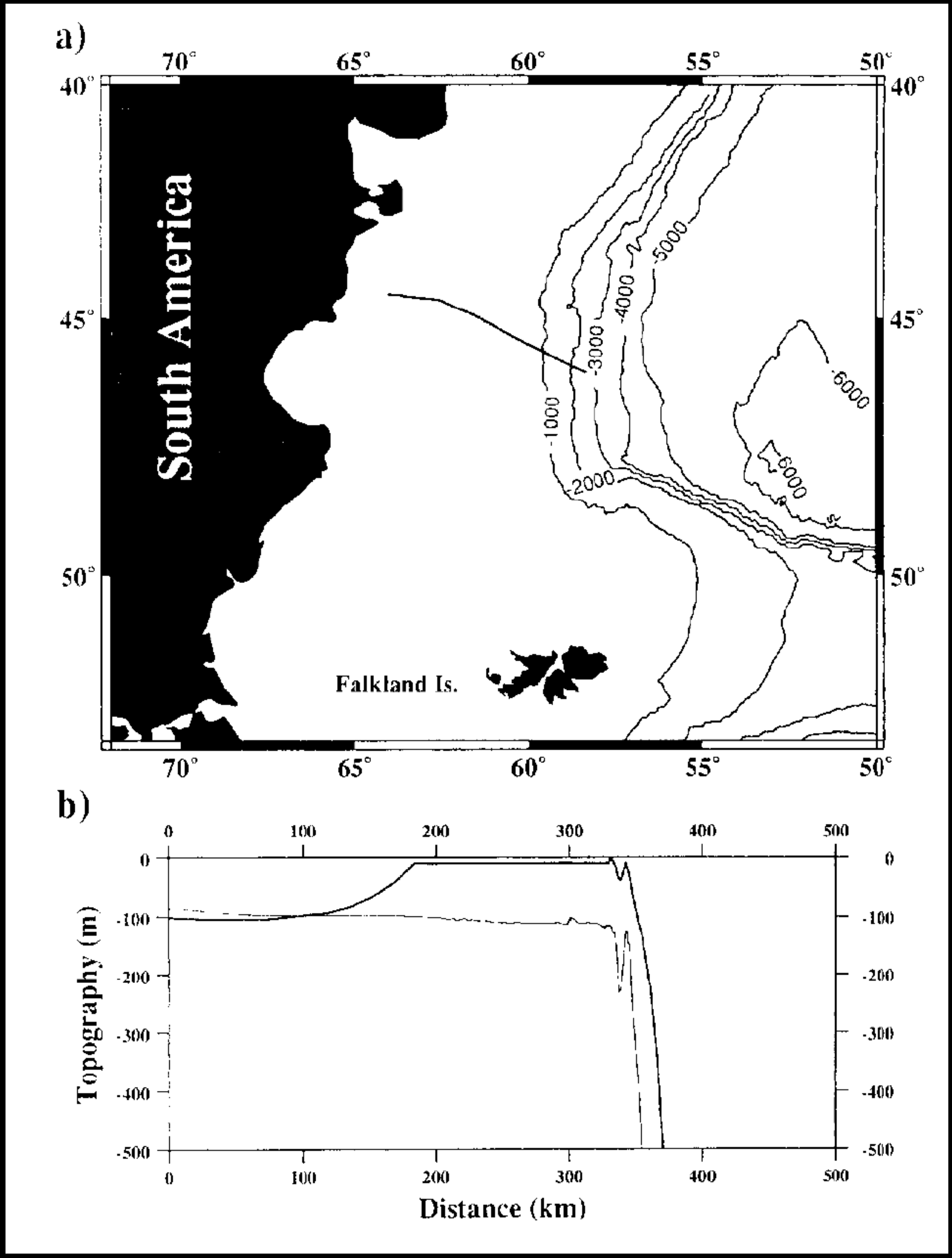
Trackline artifacts

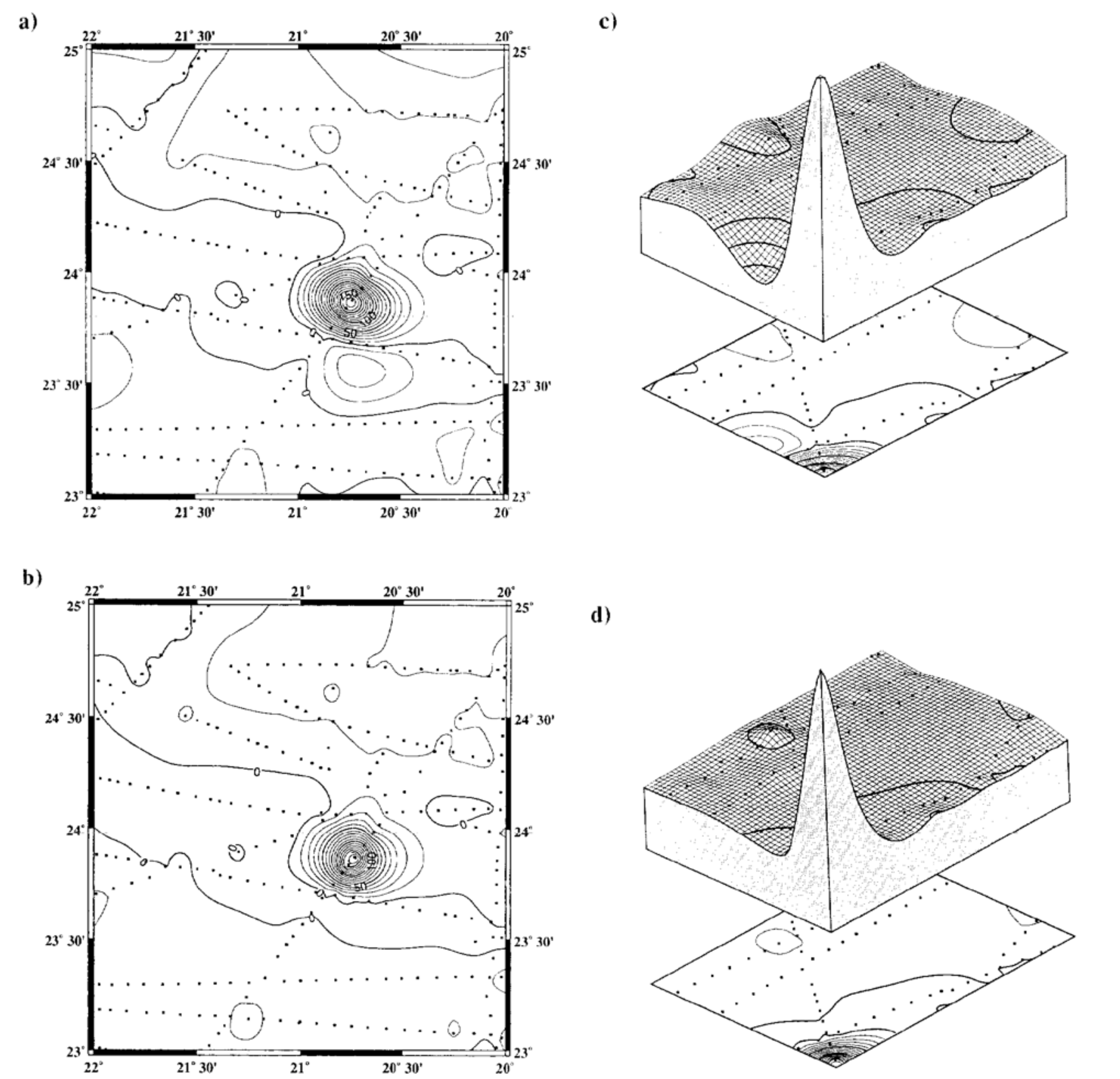


More data better?



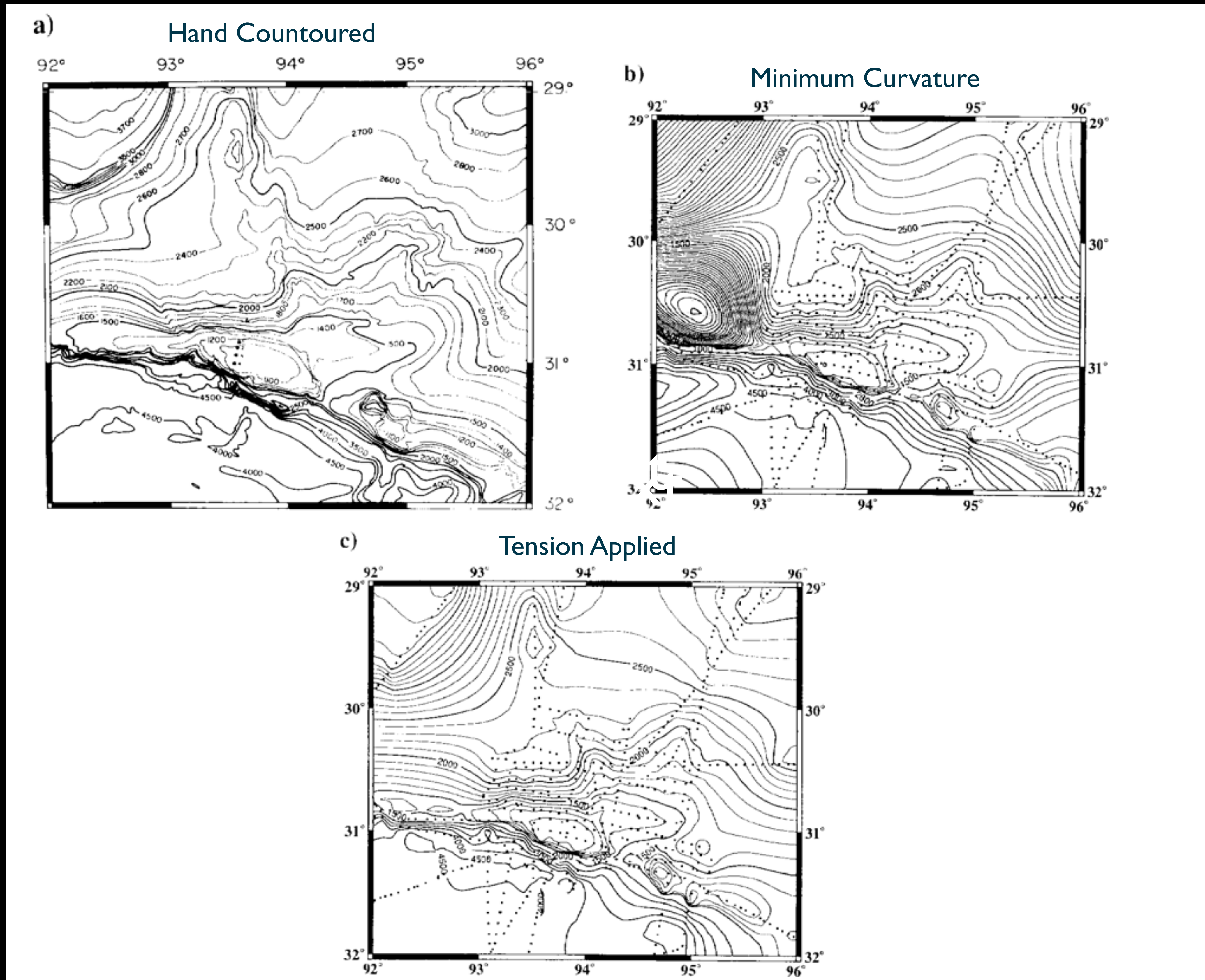
Minimum Curvature vs New Data





a)
22°
21° 30'
21°
20° 30'
20°
25°
24° 30'
24°
23° 30'
23°
150
100
50
0
b)
100
50
0
c)
d)

Broken Ridge



Gridding Programs

- blockmean L2 (x,y,z) data filter/decimator
- blockmedian L1 (x,y,z) data filter/decimator
- blockmode Mode-estimating (x,y,z) data filter/decimator

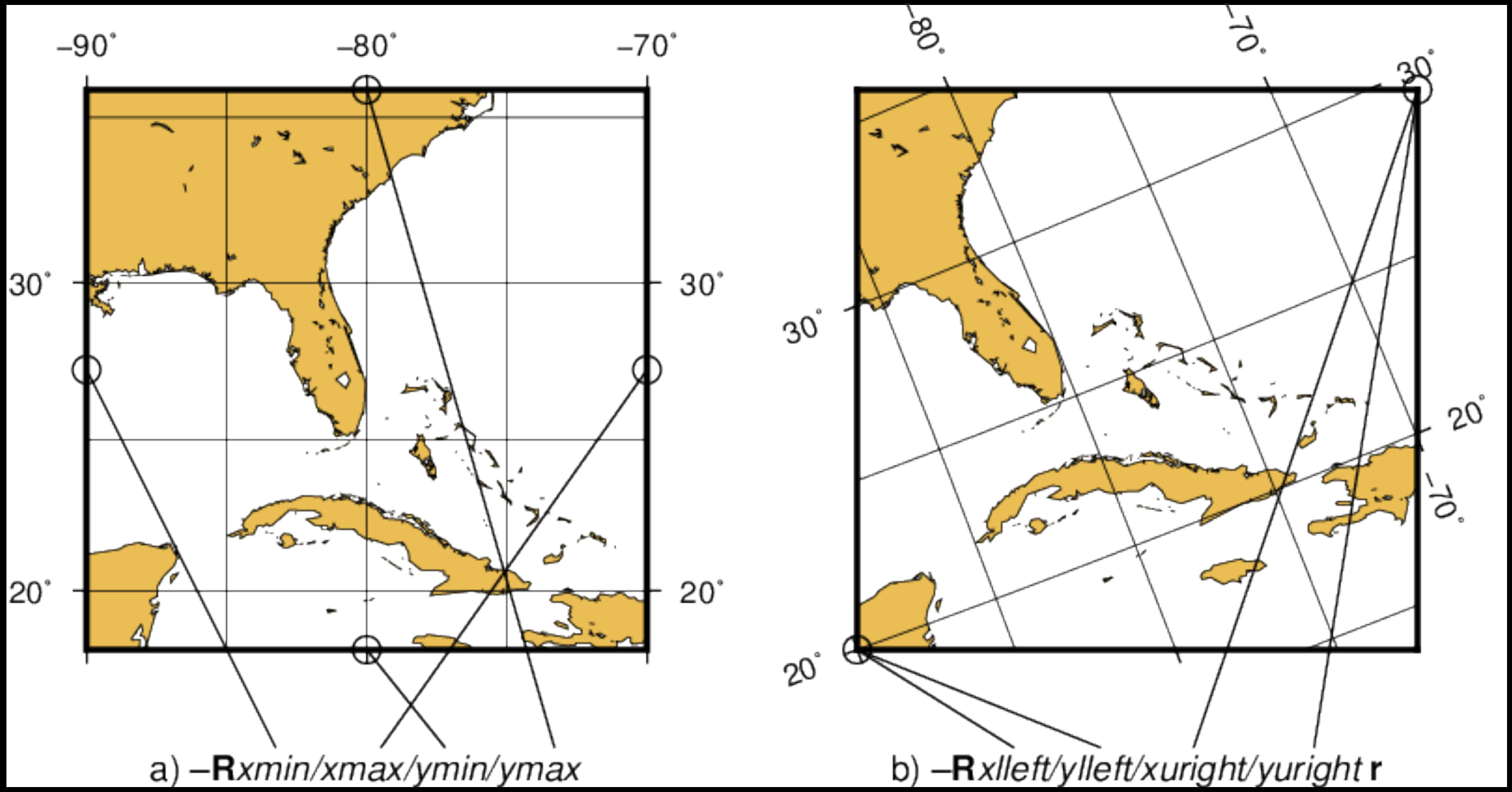
- greenspline Gridding using Green's function splines
- nearneighbor Nearest-neighbor gridding scheme
- surface Continuous curvature gridding algorithm
- triangulate Perform optimal Delauney triangulation on xyz data

Workflow

- Decide!
- Select data (probably not everything)
- Pre-process data (scaling, get rid of the garbage, NaN)
- Block process it to reduce aliasing
- Grid it (set parameters)
- Plot it

Parameters

- Surface [*xyzfile*] **–G***outputfile.grd*
- **–I***xinc/yinc*
- **–R***west/east/south/north*[**r**]
- **–C***convergence_limit*
- **–Ll***lower* **–Lu***upper*
- **–N***max_iterations*
- **–Q**
- **–S***search_radius*
- **–T***tension_factor*
- **–Z***over-relaxation_factor*



Grid Manipulations

grd2cpt Make color palette table from grdfile
grdfilter Filter 2-D data in space domain
grdsample Resample a 2-D gridded data onto new grid
grdtrack Sampling of 2-D data along 1-D track

grdblend Blend several gridded data sets into one
grdclip Limit the z-range in gridded data sets
grdedit Modify grd header information
grdfft Operate on grdfiles in frequency domain
grdgradient Compute directional gradient from grdfiles
grdhisteq Histogram equalization for grdfiles
grdlandmask Creates mask grdfile from coastline database
grdmask Set nodes outside a clip path to a constant
grdmath Reverse Polish calculator for grdfiles

NETCDF Grids

(**Network Common Data Form**)

a set of software libraries and self-describing, machine-independent data formats

supports the creation, access, and sharing of array-oriented scientific data.

Characteristics

- Data should be self-describing, without external tables needed for interpretation.
- Conventions should be developed only as needed, rather than anticipating possible needs.
- Conventions should not be onerous to use for either data-writers or data-readers.
- Metadata should be readable by humans as well as interpretable by programs.
- Redundancy should be avoided to prevent inconsistencies when writing data
- Data provenance: title, institution, contact, source (e.g. model), history (audit trail of operations), references, comment
- Description of associated activity: project, experiment
- Description of data: units, standard_name, long_name, auxiliary_variables, missing_value, valid_range, flag_values, flag_meanings
- Description of coordinates: coordinates, bounds, grid_mapping (with formula_terms); time specified with reference_time ("time since T0") and calendar attributes.
- Meaning of grid cells: cell_methods, cell_measures, and climatological statistics.

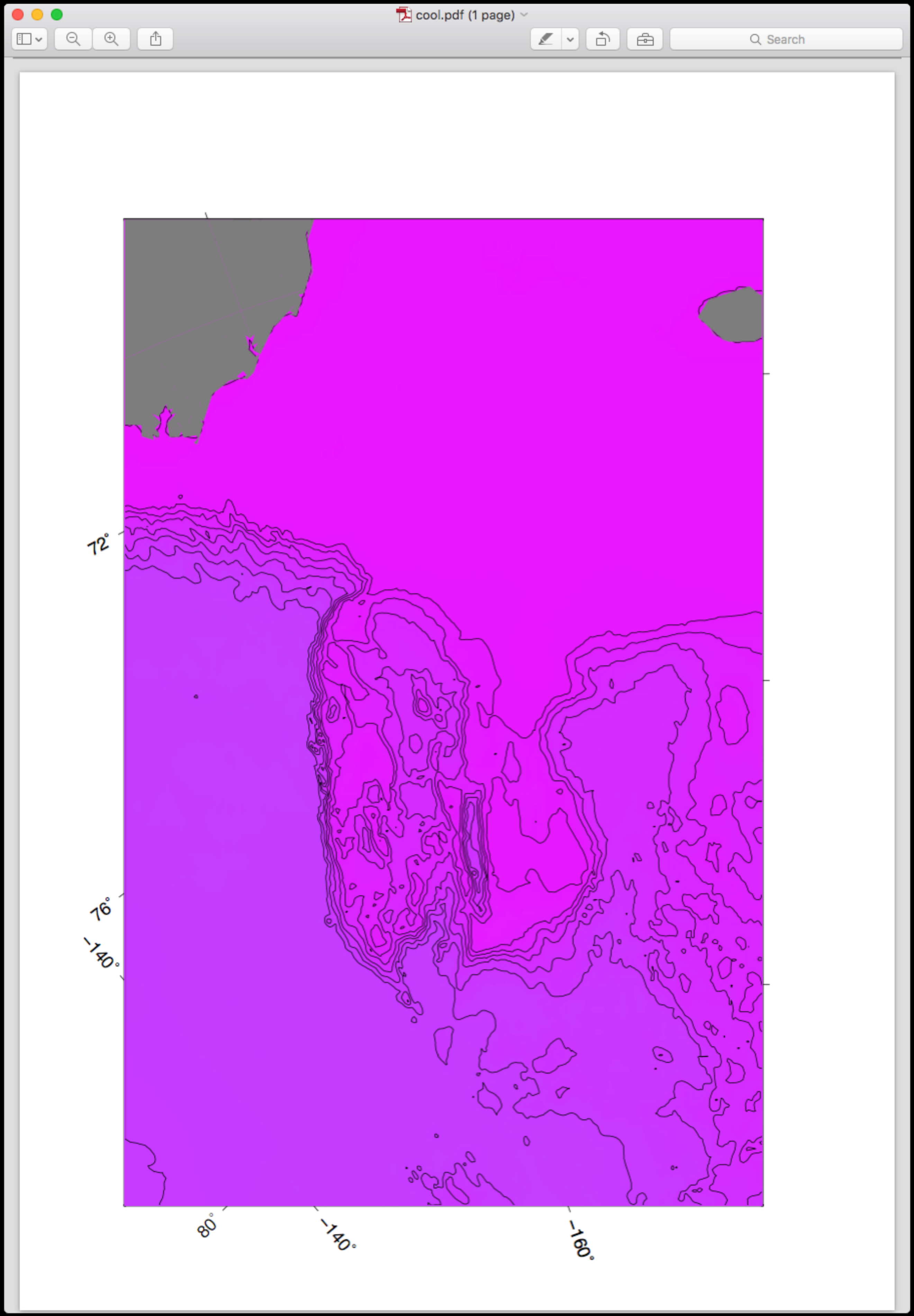
Other grid functions

- gmt2rgb Convert Sun raster or grdfile to red, green, blue component grids
- gmtconvert Convert table data from one format to another
- gmtmath Reverse Polish calculator for table data
- gmtselect Select table subsets based on multiple spatial criteria
- grd2xyz Convert 2-D gridded data to table
- grdcut Cut a sub-region from a grd file
- grdpaste Paste together grdfiles along common edge
- grdreformat Convert from one grdformat to another
- splitxyz Split xyz files into several segments
- xyz2grd Convert table to 2-D grd file

Issues of Representation

- How do you show data?
- Honestly.
- Does display distort?
- What about color?
- Why not a rainbow?

Note that each color table highlights difference elements in the grid.



There is no "natural" scaling between most data sets and color. So choosing any color table is arbitrary.