GMT Workflow

AIM: This is an an introduction to a workflow for GMT which is

- Flexible for rapid updates (as in when the reviewer or co-author says a line is hard to see, or if you suddenly decide to change all your colormaps for salinity in a paper to something with better contrast/colorblind friendly).

- Consistent in formatting a set of figures for a paper (same aspect ratios of figures, same font sizes and types, etc)

- Conducive to updating figures without getting back into analysis code (as in if you want to change a figure for a talk or proposal, but don't want to re-run the Matlab scripts you used to analyse the data and create the figure).

GMT also naturally creates a separation between your analysis software (i.e. Matlab) and presentation of the results. This separation helps ensure a trackable workflow from the start to final analysed datasets.

DATA: Grab the working directory from Dropbox: `http://bit.ly/10gNNrF`.

GETTING GMT: The GMT website is here: `http://www.soest.hawaii.edu/gmt5/`. Note: if you google "GMT soest", the default page is not the latest version, which at time of writing is GMT5! All scripts below will not perform as expected on GMT4. Instructions for how to install GMT on a Mac using Macports are here: `http://frajka-williams.com/resources/installing-gmt5-on-a-mac/`.

HOW TO USE THIS DOCUMENT: Download and install GMT, then grab the working directory from the Data link above. Once you have the directory downloaded and unzipped, try the commands and edits suggested below to get a feel for how GMT works.

---

1   The file structure

I use a single folder per project (e.g. per paper) which contains all the data used, figures created, scripts for each figure, and preferences. When a paper goes through a major update or revision, I'll duplicate the entire folder and start in a new version, effectively archiving the old bits.

1. Typical subfolders:

    - `antfiles`: which contains annotation files for custom annotations, as well as any colormaps or contour-interval files used
    - `data`: which contains the GMT-format data files used for the figures
    - `epsfiles`: which is an output folder for `*.eps` figure files (for LaTeX)
    - `pngfiles`: which is an output folder for `*.png` figure files (for e-mailing, web or ppt)
    - `shfiles`: which contains one script per figure (multiple panels are generated by the same script)

2. Main files

    - `paper_figs.sh`, the "wrapper" file which calls all the individual figure scripts, and sets preferences (like colours to use for particular lines, or symbol sizes, or latitude and longitude limits)
    - `gmt.conf` which sets the default GMT preferences (like frame style, line widths, background colours, tick lengths, etc)

2   Quickstart

To start a new project, I'll copy an old project directory and retain the `paper_figs.sh`, `gmt.conf`. I then clear the subfolders (though sometimes leave the shfiles). If you're just starting with GMT, then you can download the directory (above, in Data) as your starter project.

3   Running a figure

When I'm editing and creating figures, I'll comment out all but one of the scripts called at the end of `paper_figs.sh`. This way, only one figure-script is executed when I run `paper_figs.sh`. To run it, open a terminal window in the working directory (same directory as where `paper_figs.sh` is located):

```
$ ./paper_figs.sh
```

3.1   Make a map

The first example is `map_fig.sh` which creates a coastline map with bathymetry shaded, and a further inset map to show where the smaller region came from. To execute this one, uncomment the line in the `paper_figs.sh` file beginning with `shfiles/./map_fig.sh` and then run `paper_figs.sh` from the working directory.

  Once it works properly, check the output file in the `pngfiles` subfolder. Try a few edits to customise the plot.

1. Make a copy of the script - give it a new name, and add a line in `paper_figs.sh` to call the new figure. Comment out the old line.

2. Try changing a symbol `-S` to something new. See the man file on psxy for options `http://www.soest.hawaii.edu/gmt5/gmt/html/gmt_man.html`

3. Instead of shading bathymetry, contour it with 1000 m spacing.

4. Change the region to one where you work by updating the `latlim` and `lonlim` in the `paper_figs.sh` file. Rerun the figure. (Note, if you don't have a bathymetry file, you can download one from ETOPO (`http://frajka-williams.com/resources/data-sources/bottom-bathymetry-from-etopo/`), conveniently in GMT format already.

5. Try changing the projection to something more exotic (or appropriate). See chapter 6 in the Tech Ref: GMT_Docs.pdf in the Docs/ folder.

6. Change the location and size of the inset map (e.g. larger, higher up).

3.2   Make a time series

Create your own version of the RAPID MOC time series using `moc_tseries.sh`. Once it works properly, try a few edits to customise the plot.

1. Change the colours and line styles for the plot in `paper_figs.sh`.

2. Change the x-annotations to be something custom (like April to April)

3.3   Make a property section

Shade a temperature and salinity transect from a Seaglider in the Labrador Sea. Once it works properly, try a few edits to customise the plot.

1. Change the colourmap to one of GMTs other options.

2. Change the axes labels on the colorbar to include "psu" for salinity.

4   Troubleshooting

1. If you try to execute a script and get a command about a file (e.g. `map_fig.sh`) not being executable or permissions incorrect,

   ```
   ./paper_figs.sh: line 200: shfiles/./map_fig.sh: Permission denied
   ```

   this is probably because you need to make the script executable. You can do this using the `chmod` command and adding the executable toggle,

   ```
   $ chmod +x shfiles/map_fig.sh
   ```

2. If you're executing a script and a lot of gobbledygook comes up, and you've used the `ps2raster` command at the end to create a `*.eps` file, it may be because you've told GMT to expect another overlay. These are denoted by the `-K` and `-O` options to commands. The `-K` tells GMT to keep the file open for later overlays; the `-O` says this is an overlay, dump it on top. The very last command in a script file can have only the `-O` option and not the `-K` option. If you shuffle your commands around, it's easy to forget to change this.

5   Further resources

The folks at SOEST have created several useful documents which are useful to have to hand at all times. I've put a copy in the downloadable directory, in the `Docs` folder.

1. To work through a few more basic examples when you're just getting started, try the tutorial: **GMT_tutorial.pdf**. Beyond the basics, however, I never refer to this.

2. For everyday reference to individual commands (like `pscoast` or `ps2raster`) try the manual pages, **GMT_Manpages.pdf**. These are available as html (`http://www.soest.hawaii.edu/gmt5/gmt/html/gmt_man.html`) as well, but the same information can also be accessed at the command line by typing in the function name followed by no arguments.

   ```
   $ pscoast
   ```

3. For more in-depth fancy examples of what can be done, the GMT technical reference is great: **GMT_Docs.pdf**. It was so useful (and hard to navigate), that you may want to print it out. The parts that I use most frequently are the annotations (pg. 21–27), projections (pg. 55–84), the color tables (pg. 186), hachures (pg. 160), the symbol charts (pg. 161–162), fonts (pg. 163), and occasionally, annotating contours (pg. 193–201). Otherwise, I'll browse the figures in the document in case there's something I'd like to use, then I'll scan the example scripts to see the part that is of interest.