**Special Pleading in GMT\_Encode\_Options**

Because of historical syntax decisions that predate the API concept, some modules pose a challenge to proper API usage since they are chameleons, i.e., they may return different types of output depending on options. Also, some complicated options may imply an input file and to pass this via the API as a virtual file may require some gymnastics. This is not ideal when calling from external interfaces since we usually need to know what to expect. Here is a list of issues.

pscoast: While normally producing PostScript, this module can instead return a dataset (with -M or -E).

psxy[z]: Always plots PostScript, but for quoted or decorated lines there may be a request to read a file with fixed or crossing lines. Thus, the -S option key is S?(=2.

gmtmath or grdmath: Has used the syntax “= outfile” for a long time but this is not true for any other module where we simply write to stdout or redirect via >. Basically, the = is an operator that says “time to write output” and then accepts an optional file name. We therefore combine the two options = and outfile into a single option under the hood.

grdconvert: It has the unusual syntax of taking both an input and output file, without things like -Goutgrid syntax used everywhere else. We fix this under the hood.

greenspline: Has an unusual -R option that can do 1-D, 2-D or 3-D. The dimensionality determines what the output format is (dataset vs grid).

triangulate: For some combination of options the output default format is grid, not dataset.

grdcontour or pscontour: For some version of -G that involves external dataset files we need to modify the key to accept a secondary data set or not.

talwani3d: Need to check option -N to determine if default output is grid or dataset

blockm\*: Need to check if gridded output and how many grids and add the required -G? options.

Perhaps we should consider making some radical changes to these situation so that the life viewed from the API is simpler? As is, this is only dealt with via GMT\_Encode\_Options when called from gmtmex.c, but not in Python, for instance.