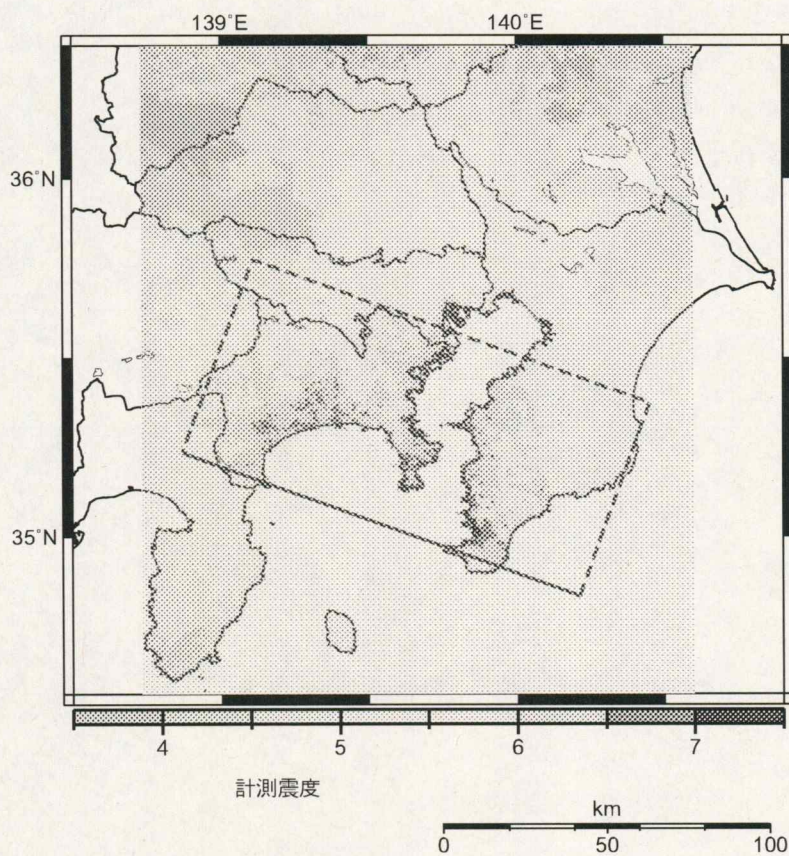

**Generic
Mapping
Tools**

GMT の使い方

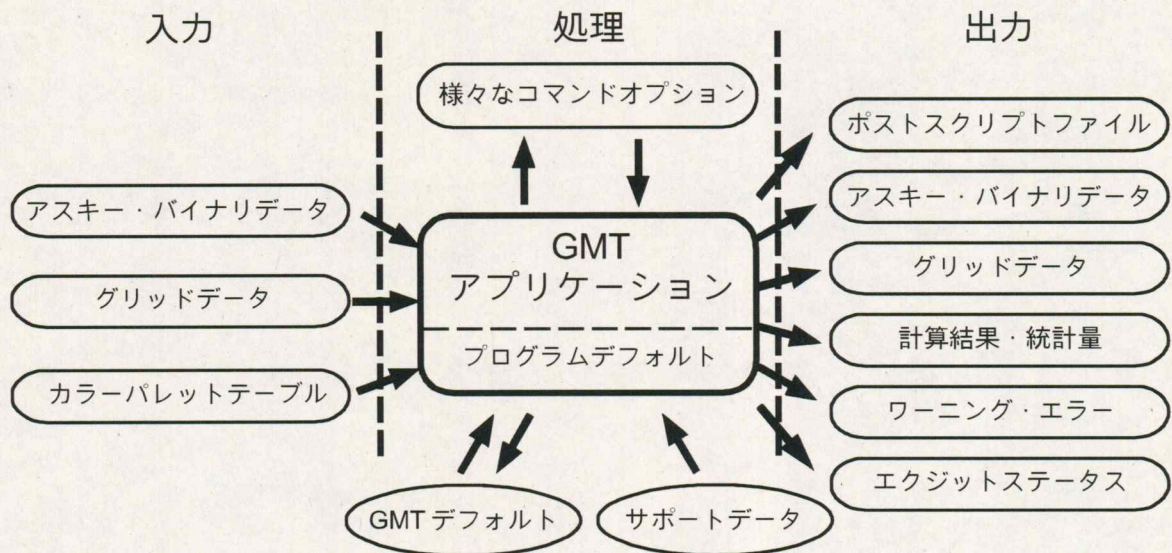


渡邊 基史
moto@ori.shimz.co.jp

GMT を使う前に

まず、GMT を使う前に GMT がどのような仕組みで動いているのか、それを理解しなければなりません。

下の図に GMT の仕組みを示す模式図を作成しました。



☆入力

入力データはおおまかに3種類に分けることができます。

1. データ (データテーブル・スプレッドシート形式のデータ)
 - ・アスキーデータ - シングルセグメント
 - ヘッダレコード付マルチセグメント (-M)
 - ・バイナリデータ - シングルセグメント
 - マルチセグメント (ヘッダはNaNデータ) (-M)
2. グリッドデータ (netCDF形式のデータマトリクス)
 - ・グリッドラインフォーマット
 - ・ピクセルフォーマット
3. カラーパレットテーブル (図を作成するときのカラーテーブル)

詳細に関してはあとで説明します。

☆処理

GMTのプログラムはその動作を様々なオプションにより操作します。

1. 様々なコマンドオプション、またはプログラムがもともと持っているデフォルトの動作オプション
2. 以前のコマンドで定義されたオプション (.gmtcommand ファイルに収納)
3. ユーザーが与えるパラメータ (.gmtdefaults ファイルに収納)
4. 地図を描画する際の海岸線データなどのサポートデータ

詳細に関してはあとで説明します。

☆出力

1. ポストスクリプトファイル (描画データ)
2. アスキー・バイナリデータ (データテーブル)
3. グリッドデータ
4. 計算結果・統計量
5. ワーニング・エラー (標準エラー出力)
6. エクジットステータス

V (^_^) おまけ

UNIX 環境の基礎知識 GMT でコマンドを書くときに便利！！

☆ リダイレクション

GMT のプログラムは標準入力からデータを受け取り、標準出力に結果をはきだします。

```
GMTprogram01 >! output-file
```

```
GMTprogram01 < input-file >! output-file
```

```
GMTprogram01 input-file >> output-file (既存のファイルに追記)
```

なんて使い方をよくします。

☆ パイプ

あるプログラムの出力を、次のプログラムの入力とするときには、

```
someprogram | GMTprogram01 | GMTprogram02 >! output-file
```

データを awk や perl で加工してから (結構便利ですよ (^_^)☆)

GMT にわたすなんてことがよく使われます。

☆ 標準エラー出力

GMT のプログラムは実行状況やエラーを標準エラー出力にはきだします。

このメッセージをとっておきたい時は

```
GMTprogram01 >! output-file >& errors.log
```

というような記述をします。

☆ 正規表現

言わずもがな f(^_^;) でしょうか * ? [] はよく使いますね。

```
GMTprogram01 data_*.a >! output-file
```

```
GMTprogram01 line_?.d >! output-file
```

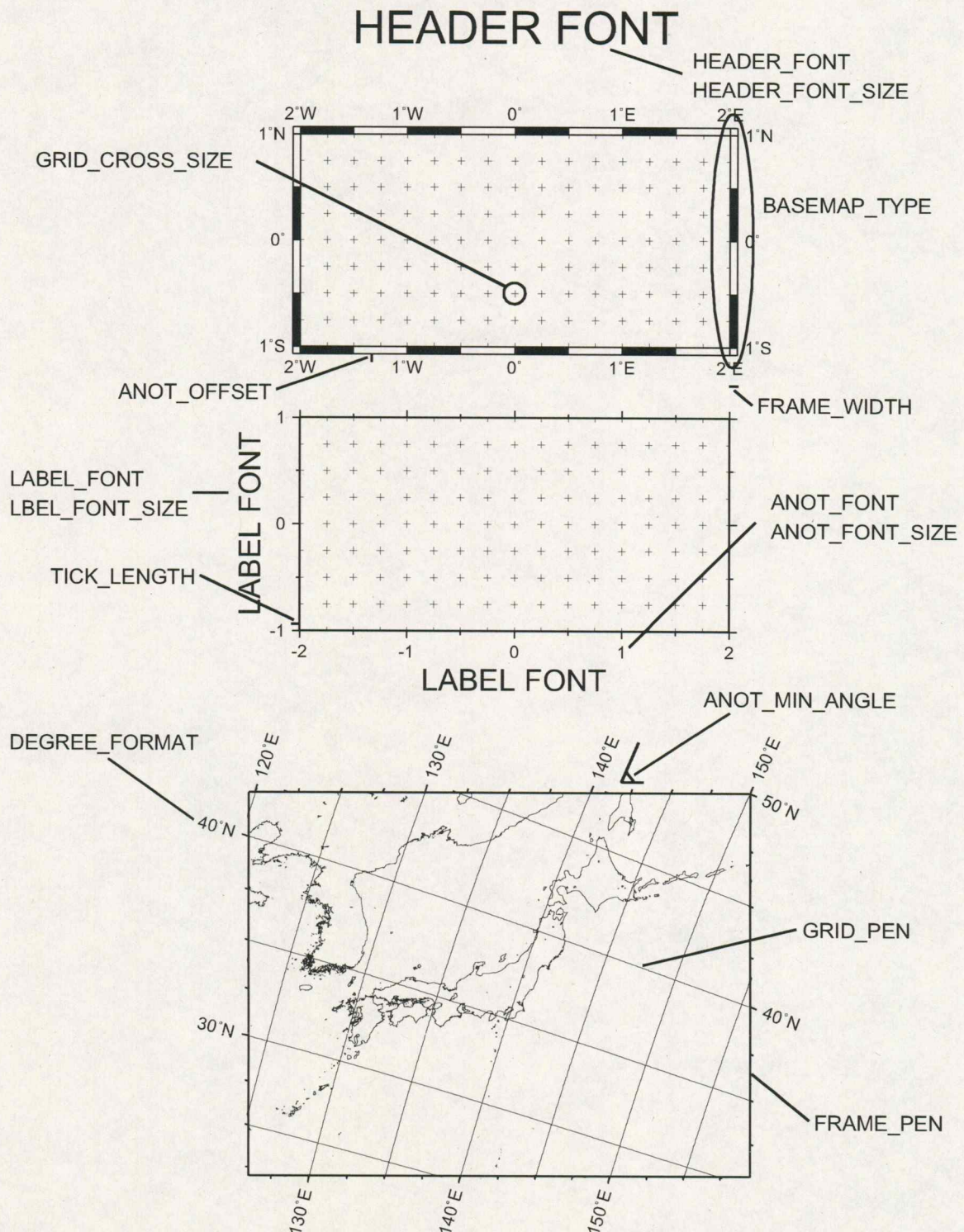
```
GMTprogram01 wave_1[0-9]0.d >! output-file
```

というようにすれば、ファイルを選択するとき手間が省けます。

GMT コマンドオプション gmtdefaults デフォルトの設定

GMTの環境設定は、作業中のディレクトリに.gmtdefaults というファイル名で自分の思う通りの設定を保存しておかなくてはなりません。

まず gmtdefaults -D >!.gmtdefaults というようにコマンドを実行し、システムのデフォルト値を保存してみてください。環境変数がいろいろと定義されていると思います。例えば次のような環境変数は、いろいろな場面で書き換えることがあると思います。参考にしてみてください。



ANOT_MIN_ANGLE

地図の枠と座標軸にふる数字とのなす角度の最小値。(20) 単位は degree。

ANOT_FONT

座標軸にふる数字のフォント。(Helvetica) どんなフォントが使えるかは、正式の man gmtdefaults を見て下さい。

ANOT_FONT_SIZE

座標軸にふる数字の大きさ。(14p) 単位は pt。

ANOT_OFFAET

座標軸にふる数字と補助目盛りの先との間の距離。(0.2c) 単位は cm。

BASEMAP_TYPE

地図の枠をどんな感じにするかを定めるパラメータです。plain か、fancy の2つのうちのどちらかしか指定できません。plain だと、普通の線枠、fancy だと、白黒交互になっている枠になります。(fancy)

fancy と指定しても、投影法によっては、普通の線枠になる場合があります。

DEGREE_FORMAT

地図を書いたときの度をどういう形で表すかを決めます。0~5の6つの値をとります。(0) それぞれの意味は、次の通りです。

0 経度：0~360、緯度：-90~90

1 経度：-180~180、緯度：-90~90

2 経度：符号なしで0~180の数字のみ、緯度：符号なしで0~90の数字のみ

3 経度：W、E、N、S、を付けて0~180(経度)、0~90(緯度)を表示

※ここまでのオプションを指定すると、1度未満の量は、分、秒を使って、表示します。

4 経度：表示の仕方(符号の付け方)は、0と同じですが、分、秒を使わずに、小数を使って1度未満の量を表示します。

5 経度：表示の仕方(符号の付け方)は、1と同じですが、分、秒を使わずに、小数を使って1度未満の量を表示します。

DOT_PR_INCH

出力デバイスの解像度です。適宜設定してください。(300) 単位は dpi

ELLIPSOID

地図を書くときの地球楕円体の選択(WGS-84)。他に、どんなものを選べるかは、man gmtdefaults を見て下さい。

FRAME_PEN

普通の線枠のときの枠線の太さ。(1.25p) 単位は pt。

FRAME_WIDTH

fancy のとき白黒交互枠の幅。(0.2c) 単位は cm。

GRID_CROSS_SIZE

緯度、経度の交点にクロスを書くときのサイズ。0を指定すると、緯線、経線をグリッド線として地図上に引きます。(0c)

GRID_PEN

グリッド線を引くときの線の太さ。(0.25p) 単位は pt。

HEADER_FONT

図のタイトルを書くときの文字のフォント。(Helvetica)

使えるフォントとその指定の仕方は、ANOT_FONT のそれと同じです。

HEADER_FONT_SIZE

図のタイトルを書くときの文字のサイズ。(36p) これでは、大きすぎると思います。適宜書き換えてください。単位は pt。

LABEL_FONT

座標軸に付けるラベルのフォントです。(Helvetica) 使えるフォントとその指定の仕方は、ANOT_FONT のそれと同じです。

LABEL_FONT_SIZE

座標軸に付けるラベルのサイズです。(24p) 単位は pt。

MEASURE_UNIT

紙の上で大きさを指定するときの単位です。(cm) 米国では inch の場合も多い。

PAGE_ORIENTATION

用紙の方向 portrait か landscape。(landscape)

PAPER_MEDIA

用紙の大きさ。(a4) EPS ファイルをつくりたいときには用紙の名前に”+”をつける。(a4+とか) BoundingBox を生成するとき、デフォルトでは紙の大きさの方を優先してしまいます。

TICK_LENGTH

座標軸に付ける補助目盛りのサイズです。マイナスで指定すると枠の内側に書くようになります。(0.2c) 単位は cm。

VECTOR_SHAPE

矢印の頭の形を指定します。値の範囲は、0~1です。(0) 0だと、矢印の頭は3角形で、1に近づくにつれてだんだんやせていきます。

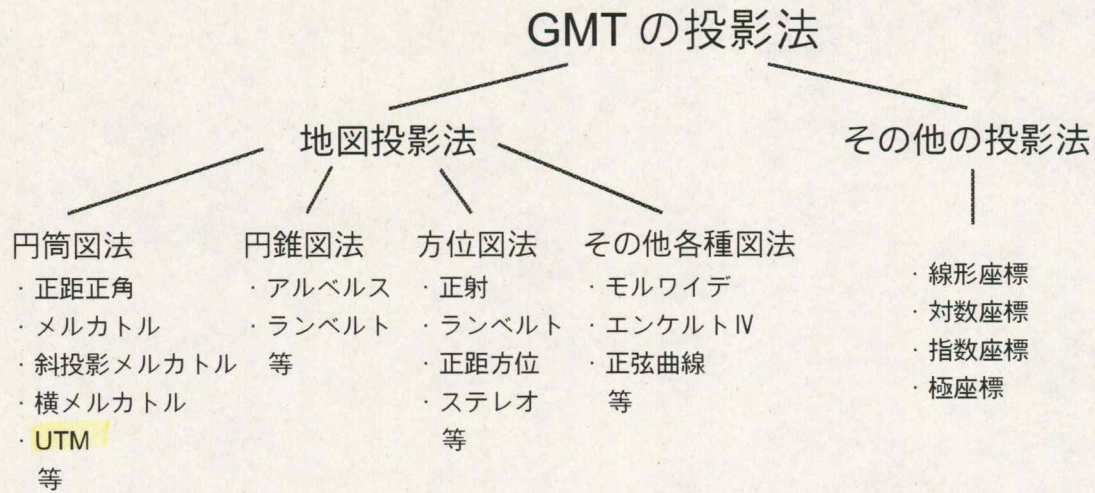
ここで示した以外のパラメータもあります。とりあえずこれくらいのパラメータを覚えておけば十分だと思いますが、詳細に関しては man gmtdefaults で確認してください。

自分の思うとおりに環境設定したファイルはどこかに保存しておいて、GMTを起動する前に作業しているディレクトリにコピーすれば、その設定どおりの図が出力されるはずです。

<参考>

GMTが参照するパラメータはまず、作業しているディレクトリの直下の.gmtdefaultsのパラメータ、それがなければ、自分のホームの下の.gmtdefaultsのパラメータ、最後にプログラムがデフォルトでもっているパラメータというような順番になっています。

GMT コマンドオプション -J 投影法の指定



-J code parameter

-J code は投影法を指定するオプションで code をいろいろ変えることで様々な投影法を選択することができます。上記に示した他にも20を越える投影法から好きなものを選ぶことができます。普通は、code を小文字で指定すると1単位(度)当たりのスケール(cm)か、縮尺(1:xx)で大きさを指定します。大文字で指定すると図全体の大きさ、幅(cm)を指定するようになっています。

よく使うのは次に示すようなものだと思います。幅 width を cm で指定しておけば希望通りの大きさの図になります。

メルカトル **-JMwidth**

アルベルス **-JBlon0/lat0/lat1/lat2/width**

投影中心(lon0/lat0)と標準緯線(lat1/lat2)を指定

ランベルト **-JLlon0/lat0/lat1/lat2/width** アルベルスと同様

線形 **-JXwidth[/height]**

幅 width を指定します。高さも指定したい場合は height を加えます。

width (height) を指定するには次に示す3書式があります。

-JXwidth[d] 通常の線形軸

-JXwidthl 対数軸

-JXwidthppower 指数軸

もし width (height) を負の数で指定すると軸の向きが反転します。つまり x 軸は左に、y 軸は下に向かって値が大きくなるようになります。

d オプションは単位を度で与えるようにするオプションです。ラベル表示が地図の時と同じようになります。

さらに詳しい解説や他の投影法に関しては man psbasemap で確認してください。

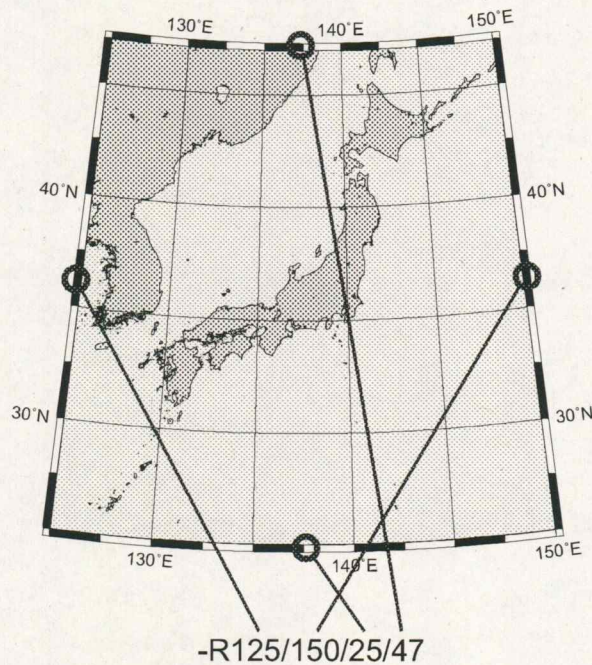
GMT コマンドオプション -R 描画領域の指定

-R xmin/xmax/ymin/ymax [r]

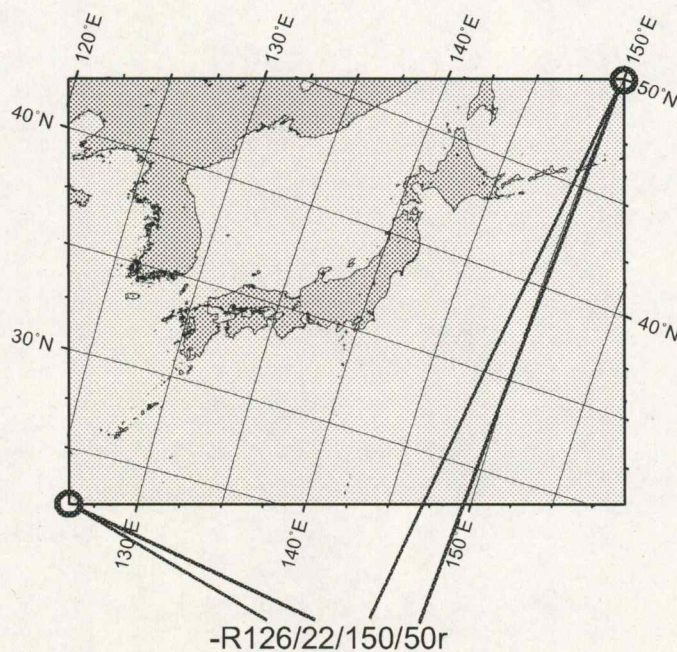
xmin (/xmax/ymin/ymax) の形式は、整数、小数、指数付きの書式、何でも指定できます。度、分 [秒] で指定するのも可能で、そのときには dd : mm [: ss] の書式で入力します。

通常は、x、y 各軸の最小値と最大値を指定することでその範囲で囲まれた矩形領域を描画領域としますが、r オプションをつけることで左下隅と右上隅の2点の座標で矩形領域を指定し、描画領域とすることの可能です。

通常の領域指定



r オプションをつけた場合の領域指定



GMT コマンドオプション -B 軸書式の指定

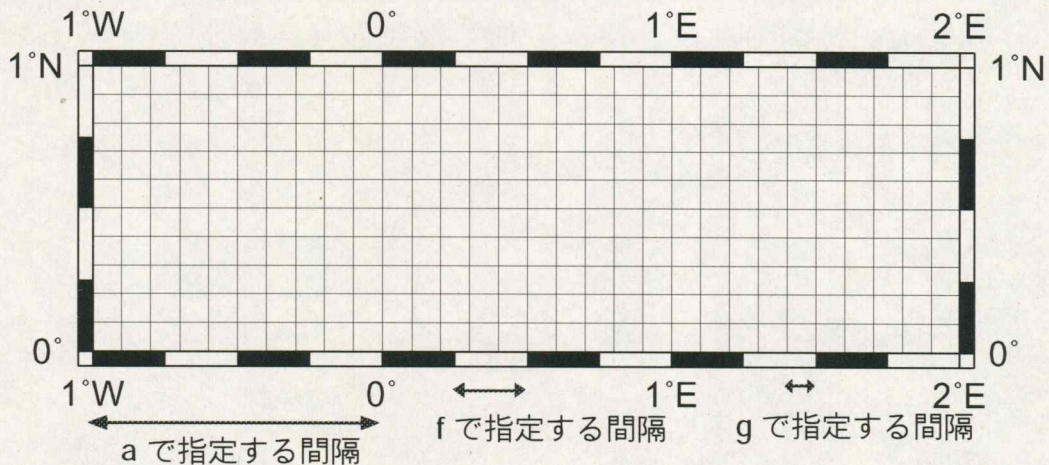
-B xinfo [/yinfo][: " title "]: [W|w] [E|e] [S|s] [N|n]

xinfo [yinfo]の形式は以下のようになります

[a] tick [m|c] [f] tick [m|c] [g] tick [m|c] [l|p][: " axis label "]:[: " unit label "]:

ここで、aはラベル、fは目盛、gはグリッドの間隔をそれぞれ指定します。m|cは単位を度、分(m)、秒(c)で指定するとき用います。デフォルトではラベルは上下左右(W E S N) 4つの軸に表示されますが、指定した軸にのみ表示させることも可能です。例えば WeSn とすれば左と下の軸にラベルが表示されます。

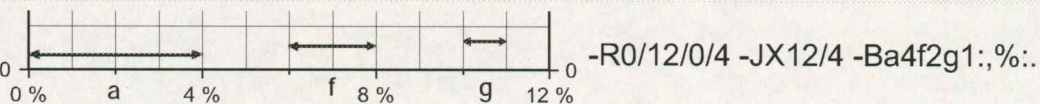
log軸の場合、tickは1,2,3で指定します。1は 1×10^n 、2は $(1,2,5) \times 10^n$ 、3はすべて、つまり $(1,2,3 \sim 9) \times 10^n$ に対応します。pを指定すると 10^n の形式で表示します。また、lを指定するとlogの値で表示します。(つまり1,10,100が0,1,2というように)



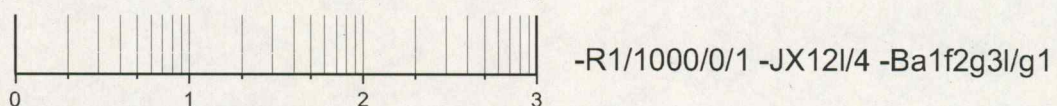
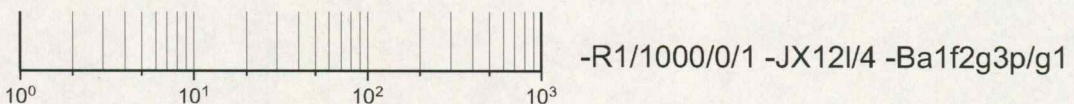
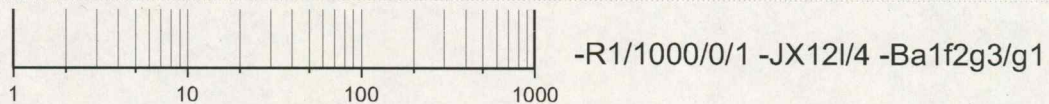
地図軸の場合の例



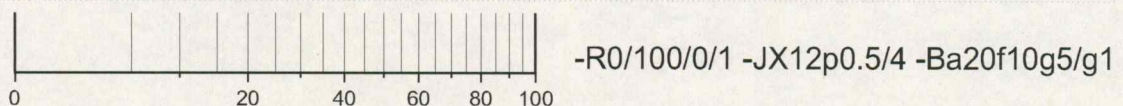
線形軸の場合の例



対数軸の場合の例



指数軸の場合の例



GMT コマンドオプション -O -K オーバーレイの指定

-O, -K はポストスクリプトコードのオーバーレイをコントロールするオプションです。ポストスクリプトファイルは大きく分けると3つの部分で構成されています。

1. ヘッダー部分 (初期化と定義)
2. 本体 (描画部分)
3. ページ印刷

これらの順番が守られるようにGMTのコマンドでオーバーレイをコントロールする必要があります。ポストスクリプトコードのオーバーレイは一般的に次のように行います。

最初のコマンド -K >! psfile

次のコマンド -O -K >> psfile

次のコマンド -O -K >> psfile

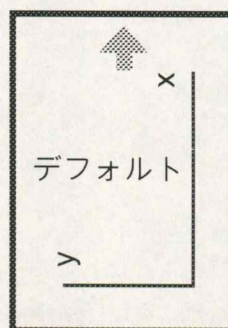
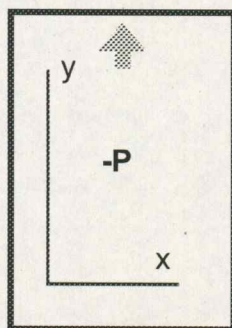
最後のコマンド -O >> psfile

-Kは追記すべき次のコマンドがあるというオプション、-Oはオーバーレイプロットモードにするというオプションです。最後のコマンドには次にコマンドがありませんから-Kを指定せず、-Oのみ指定する。 . . . というようにシェルを組むとポストスクリプトコードのオーバーレイがうまくコントロールされ、ひとつの図に様々な情報を追加していくことができます。

思ったような図が出力されなかったり、プリントができなかったりする原因の多くはこのルールを守らないことによるものですので注意してください。

GMT コマンドオプション -P 用紙方向の指定

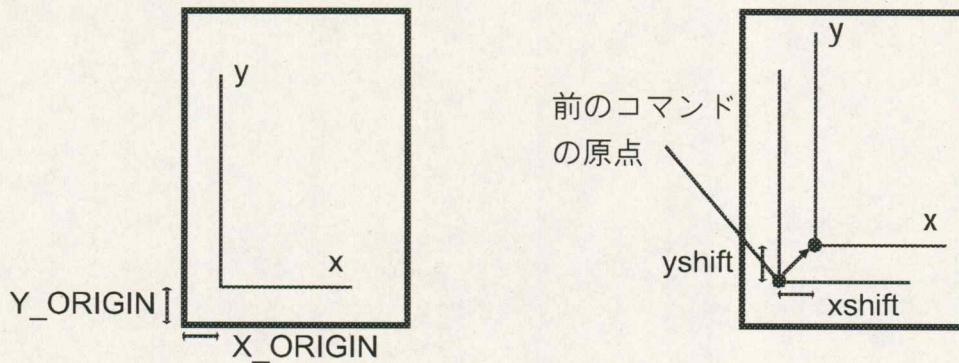
-P は用紙方向を縦にするオプションです。デフォルトの用紙方向は横ですから -P を指定しない限り用紙方向は縦になりません。(GMTはもともと地図描画ソフトで、世界地図は横長が多いからでしょうか?) もちろん .gmtdefaults ファイルを書き換えれば、デフォルトの変更も可能です。デフォルトの用紙ですが、.gmtdefaults ファイルでは、PAPER_MEDIA= a4 となっており、これはA4 縦の用紙の値で設定されています。(デフォルトの用紙方向は横ですが、大きさは縦方向に設定されています。) 他の用紙を用いる場合はMEDIAの値を変更すればよいことになります。ただし、ポストスクリプトファイルをEPSファイルに変換するとページ概念がなくなりますので、用紙の大きさが常に保存されるわけではありません。



GMT コマンドオプション `-X -Y` 原点位置の指定

`-X, -Y` は原点位置のシフトを指定するオプションです。デフォルトは $(2.5c, 2.5c)$ になっています。この値は最初のコマンドの原点位置の値です。次に追記するコマンドからは前のコマンドにおける原点からの相対位置で指定します。(GMTのコマンドは用紙上の絶対位置を知ることはできません。)

デフォルトの値では軸ラベルが用紙からはみ出す場合があります。そのような場合は適切な値を `-X, -Y` オプションで指定して図の位置を調整してください。



用紙方向のオプションの項でも書きましたが、ポストスクリプトファイルをEPSファイルに変換するとページ概念がなくなりますので、最初のコマンドの原点位置の値は意味を持たなくなります。次に追記するコマンドから指定する相対位置のみ有効となります。

GMT コマンドオプション `-U` タイムスタンプ

`-U` を指定するとタイムスタンプと簡単なテキストを用紙の左隅に出力できます。

GMT コマンドオプション `-V` レポートモードの指定

`-V` を指定するとエラー出力に各コマンドのレポートが出力されます。

GMT コマンドオプション `-H` ヘッダーレコードの指定

`-H` を使って入力データのヘッダーレコードの行数を指定すると、その行数分読み飛ばされます。デフォルトは1です。

```
Data from AAA
File name -aaa_01.dat-
x-pos y-pos data(deg)
0.02 0.03 0.012483
0.04 0.85 0.008542
0.32 0.21 0.032581
```

例えば左のようなデータの場合は
`-H3`
となります。

GMT コマンドオプション `-:` 入力順序の指定

通常の入力データスタイルは(経度、緯度)もしくは (x, y) の順序となっていますが、`-:` を指定すると、順序が逆に、つまり(緯度、経度)もしくは (y, x) の順序にすることができます。

GMT コマンド psxy 2次元データプロット

psxy は2次元でライン、ポリゴン、シンボルをプロットするコマンドです。コマンドオプションは以下のようになっています。

```
psxy files -Jparameters -Rwest/east/south/north[r] [ -A ] [ -Btickinfo ]
[ -Ccptfile ] [ -E[x|y][cap][pen] ] [ -Gfill ] [ -H[nrec] ] [ -K ] [ -L ] [ -N ]
[ -M[flag] ] [ -O ] [ -P ] [ -S[symbol][size] ] [ -U[dx/dy/][label] ]
[ -V ] [ -Wpen ] [ -Xx-shift ] [ -Yy-shift ] [ -: ] [ -ccopies ] [ -bi[s][n] ]
```

コマンドオプションに関して、詳しくはオンラインヘルプを参照してください。ここでは代表的なコマンドオプションに関して解説します。

GMT コマンドオプション -W ペンの設定

ライン、ポリゴンをプロットするときに用いるコマンドオプションは [-Wpen] オプションです。このオプションで線の太さ、線種、色を指定することができます。書式は、-Wwidth color texute で、デフォルトは太さ1、黒の実線 (-W1/0/0) です。太さの単位 width は dpi 単位で指定します (ということは整数で設定することになります)。.gmtdefaults ファイルで指定した DOT_PR_INCH の出力デバイスの解像度の値に依存する量ですので、それほど精密に定義されてはいないことを理解しておいてください。ただし、5p というように指定すると単位が pt となります。

color は r/g/b で指定します。0/0/0 が黒、255/255/255 が白になります。texute は以下に示す例を参考にしてください。

```
-----
-W3
-----
-W3t20_10:0

-----
-W3t10_10:0
-----
-W3t30_10_10_10:0

-----
-W3t50_10_15_10_15_10:0
```

このように各セグメント (プロットするセグメントとしないセグメント) の長さを単位で指定することで、実線、破線、点線、一点鎖線、二点鎖線などが自由なプロポーショナルで指定することができます。各セグメントの長さは太さ指定のときと同様に出力デバイスの解像度の値に依存する量となっています。

GMT コマンドオプション -G 塗りつぶしの設定

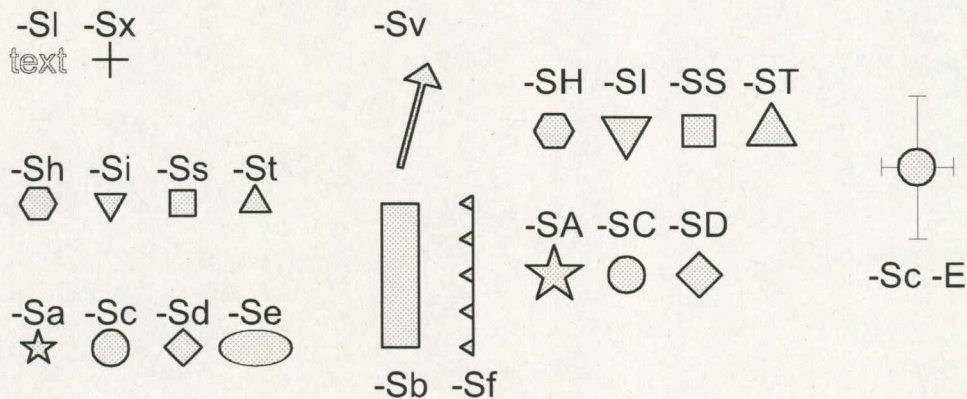
シンボルやポリゴンを塗りつぶす場合、外枠の線の太さ、色は、-Wwidth color で、なかの色は [-Gfill] オプションで指定します。色は r/g/b で指定します。-G0/0/0 が黒、-G255/255/255 が白になります。-GP|pdpi/pattern のように指定すると、領域を斜線や模様で塗りつぶすことも可能です。

GMTコマンドオプション -S シンボルの設定

シンボルをプロットするときに用いるコマンドオプションは `[-S[symbol][size]]` オプションです。このオプションでシンボルの種類、大きさを指定することができます。

シンボルの種類とその表示例を以下に示します。シンボルの大きさは、`-Ssymbol/size` で、sizeはcm単位で指定しますが、下に示す例のように同じ大きさで指定してもシンボルの種類によって見た目の大きさが(種類によって直径や、辺の長さで定義するので)変わります。

<code>-Sasize</code>	星形
<code>-Sbsize[/base][u]</code>	バー(棒)・sizeは幅、uの場合はx軸の単位で
<code>-Scsize</code>	円形
<code>-Sdsize</code>	ひし形
<code>-Se</code>	楕円・入力は方向(水平軸から反時計回り)、長軸、短軸の長さ(cm)の3つ
<code>-SE</code>	楕円・入力は方向(北から時計回り)、長軸、短軸の長さ(km) 地図上で用いる
<code>-Sfgap/tick[l L r R]</code>	断層・gapとtickはtickマークの長さと間隔を指定
<code>-Shsize</code>	六角形
<code>-Sisize</code>	下向き三角形
<code>-Sisize/string[%font]</code>	文字・sizeは文字の大きさ
<code>-Sp</code>	点・大きさはなく1ピクセル
<code>-Sssize</code>	四角形
<code>-Sstsize</code>	三角形
<code>-Sv[thick/length/width][norm]</code>	矢印・入力は方向(水平軸から反時計回り)と長さ オプションで矢印の形状をthick/length/widthで指定できる
<code>-SV[thick/length/width][norm]</code>	矢印・入力は方向(北から時計回り)と長さ 地図上で用いる
<code>-Sxsize</code>	十字形



シンボル設定が a, c, d, h, i, s, t, x のとき、これを大文字で設定した場合には、サイズを設定した円形と面積が等しくなるように大きさを調整します。また `-E` オプションを指定してシンボルにエラーバーを表示することもできます。

このようなシンボルに対して、大きさや色の指定を変数として与えて、各データごとに大きさ、色を変化させてプロットしたい様な場合は、次のようにデータを与えます。

x y [z] [size] [dx] [dy] [symbol]

通常のプロットの場合は最初の2項、(x, y)の座標値が与えられていればよいこととなります。

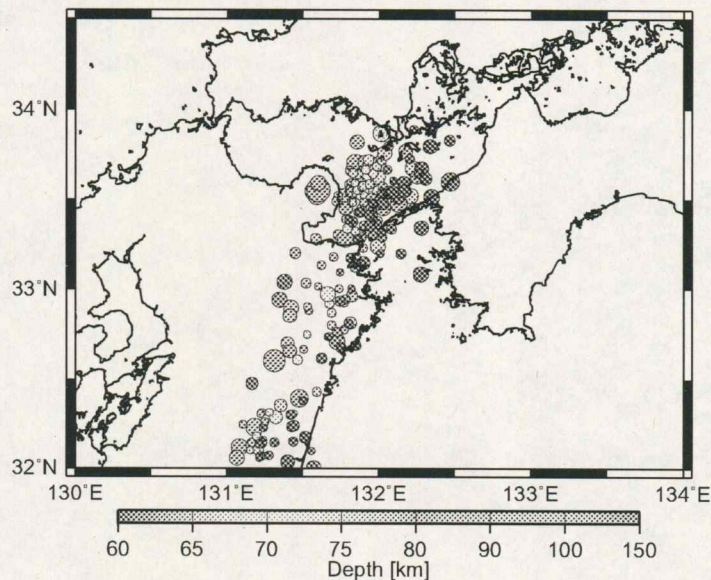
シンボルの色を変化させたい場合には、-C オプションでカラーパレットファイルを指定し、そのカラーパレットファイルで色とそれに対応する変数の値との関係を定義をします。このときには色を変化させる変数は、入力データの値[z]の位置となります。

シンボルの大きさを変化させる場合には、[-Ssymbol/size] オプションで、size を指定せず、入力データをシンボルの大きさを変化させる変数とします。大きさを変化させる変数は入力データの値[size]の位置となります。

エラーバーを表示する場合は、-E オプションを指定して、[dx] [dy]の値を与えます。一方向のみにエラーバーを表示する場合は、-Ex か -Ey、xy 両方向に表示したい場合は、-Exy というように -E オプションを変化させます。

矢印(ベクトル)を表示する -Sv オプションのとき、入力データ値[size]に対応するのは、方向と長さの2つのデータとなります。

楕円を表示する -Se オプションの場合、入力データ値[size]に対応するのは、方向と長軸、短軸長さの3つのデータとなります。



具体例として、上の図は震源データをプロットしたのですが、入力データの形式は
経度 緯度 深さ マグニチュード

となっています。よって、このデータ入力形式によれば、先述のルールに従って、深さによってシンボルの色が変化し、マグニチュードによってシンボルの大きさが変化するようにプロットされることとなります。

このようなシンボルに対して、大きさや色の指定を変数として与えて、各データごとに大きさ、色を変化させてプロットしたい様な場合は、次のようにデータを与えます。

x y [z] [size] [dx] [dy] [symbol]

通常のプロットの場合は最初の2項、(x, y)の座標値が与えられていればよいことになります。

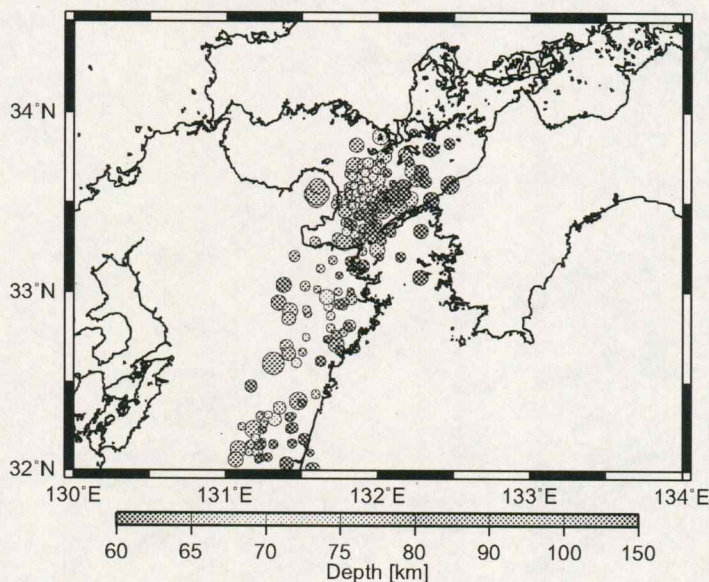
シンボルの色を変化させたい場合には、-C オプションでカラーパレットファイルを指定し、そのカラーパレットファイルで色とそれに対応する変数の値との関係を定義をします。このときには色を変化させる変数は、入力データの値[z]の位置となります。

シンボルの大きさを変化させる場合には、[-Ssymbol/size] オプションで、size を指定せず、入力データをシンボルの大きさを変化させる変数とします。大きさを変化させる変数は入力データの値[size]の位置となります。

エラーバーを表示する場合は、-E オプションを指定して、[dx] [dy]の値を与えます。一方向のみにエラーバーを表示する場合は、-Exか-Ey、xy両方向に表示したい場合は、-Exy というように-E オプションを変化させます。

矢印(ベクトル)を表示する-Sv オプションのとき、入力データ値[size]に対応するのは、方向と長さの2つのデータとなります。

楕円を表示する-Se オプションの場合、入力データ値[size]に対応するのは、方向と長軸、短軸長さの3つのデータとなります。



具体例として、上の図は震源データをプロットしたのですが、入力データの形式は
経度 緯度 深さ マグニチュード

となっています。よって、このデータ入力形式によれば、前述のルールに従って、深さによってシンボルの色が変わり、マグニチュードによってシンボルの大きさが変化するようプロットされることになります。

GMT コマンドオプション -C カラーパレット

シンボルプロットやイメージの色、コンターラインの値などを設定するのが-C オプションです。このオプションで指定したカラーパレットファイルによって色分けされます。カラーパレットファイルの書式は、以下のようになっています。

```

z0 Rmin Gmin Bmin z1 Rmax Gmax Bmax [A]
. . . . .
zn-2 Rmin Gmin Bmin zn-1 Rmax Gmax Bmax [A]

```

まず、z の値、続いて色を r g b で指定という書式で1行に2セット分、右左に分けて書きます。この指定方法で、目的の色が出せるように数行書けばよいのです。ただし、左(小さい方)のzの値、r g bの値は、前の行の右(大きい方)のzの値、r g bの値との間にギャップが生じないように設定してください。最後の[A]はカラースケールやコンターラインで文字を表示する時に指定するオプションです。[A]はLUBの3つ文字で指定するもので、Lは1行の小さい(Lower)方のzの値を文字で表示する場合、Uは1行の大きい(Upper)方のzの値を文字で表示する場合、Bは1行の両方(Both)のzの値を文字で表示する場合に対応します。

以下にカラーパレットファイルの実例を示します。

<pre> # cpt file created by: makecpt -T0/200/20 -Z #COLOR_MODEL = RGB # 0 255 0 255 20 127 0 255 20 127 0 255 40 0 0 255 40 0 0 255 60 0 127 255 60 0 127 255 80 0 255 255 80 0 255 255 100 0 255 127 100 0 255 127 120 0 255 0 120 0 255 0 140 127 255 0 140 127 255 0 160 255 255 0 160 255 255 0 180 255 127 0 180 255 127 0 200 255 0 0 B 0 0 0 F 255 255 255 N 128 128 128 </pre>	<pre> # cpt file created by: makecpt -T0/220/20 #COLOR_MODEL = RGB # 0 255 0 255 20 255 0 255 20 127 0 255 40 127 0 255 40 0 0 255 60 0 0 255 60 0 127 255 80 0 127 255 80 0 255 255 100 0 255 255 100 0 255 127 120 0 255 127 120 0 255 0 140 0 255 0 140 127 255 0 160 127 255 0 160 255 255 0 180 255 255 0 180 255 127 0 200 255 127 0 200 255 0 0 220 255 0 0 B 0 0 0 F 255 255 255 N 128 128 128 </pre>
--	---

(1)

(2)

上記の例では、(1)が色が徐々に変化するように設定する場合のファイルです。1行の中で小さい方の値のr g bと大きい方の値のr g bとが異なっています。zの値が左から右に変化するに伴って色も変化します。一方、(2)では左右のr g bが同じですから、zの値がその範囲にあるあいだは同じ色で色分けされます。カラーコンターを描く場合、(2)のようなカラーパレットファイルを使った方が、色に対応するzの値がわかりやすく、きれいに見える場合があります。

最後の3行の色指定は、B (background) がz0より小さなzに対する色、F (foreground) がzn-1より大きなzに対する色、N (NaN)がzの値がない場合に対する色の指定となっています。この項目は.gmtdefaultsファイルでも指定される値です。

GMT コマンドオプション pstext テキストの表示

pstext は任意の位置にテキストをプロットするコマンドです。
コマンドオプションは以下のようになっています。

```
pstext textfile -Jparameters -Rwest/east/south/north[r] [ -Btickinfo ] [ -Cdx/dy ]  
-Ddx/dy[ v[red/green/blue] ] [ -Eazimuth/elevation ] [ -Gred/green/blue ]  
[ -H ] [ -K ] [ -L ] [ -N ] [ -O ] [ -P ] [ -Spen ] [ -U[dx/dy/][label] ]  
[ -V ] [ -W[red/green/blue][o|O|c|C] pen ]  
[ -Xx-shift ] [ -Yy-shift ] [ -Zzlevel ] [ -ccopies ] [ -: ]
```

pstext は以下に示す 7 データを入力として読みとり、テキストをプロットします。

```
x y size angle fontno justify text
```

入力順は上記の通りです。x, y はテキストをプロットするときの基準点の座標値、size は文字の大きさをポイントで指定し、angle は回転角です。fontno はフォントを指定するパラメータでフォント番号で指定することになります。justify は基準点とテキストストリングとの位置関係を規定するパラメータです。text の部分にはプロットしたいテキストを書き込みます。

入力パラメータの angle と justify の指定例を示します。下の図を見てください。

angle20

angle-20

justify-LB justify-CB justify-RB+

justify-LM justify-CM justify-RM+

justify-LT justify-CT justify-RT+

angle は回転角を反時計回りに指定します。justify は水平方向の位置関係を示す L(left) C(center) R(right) と上下方向の位置関係を示す T(top) M(middle) B(bottom) の文字を組み合わせた 9 種類で、基準点とテキストストリングとの位置関係を規定します。上記の例では+の位置が x, y で指定する基準点の座標値になっています。この justify をうまく利用してテキストのプロット位置を決めて下さい。

-W オプションはテキストボックスの色を指定するオプションです。デフォルトは白 (255/255/255) になっています。また -Wo[pen] のように o をつけて指定するとテキストボックスを囲む線種を pen で指定することもできます。デフォルトは黒の実線 (1/0/0/0) です。これに関して、テキストとテキストボックスとの距離を指定するオプションが -C オプションとなっています。例を示すと次のようになります。

-W オプションで指定する
テキストボックスのスタイル



Text string

-C オプションで指定する
テキストボックスとの距離

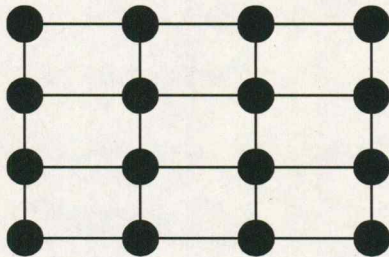
GMTコマンドオプション xyz2grd データコンバート

xyz2grd は ASCII または binary 形式の (x, y, z) 等間隔グリッドデータを GMT で用いられている 2次元バイナリの netCDF 形式のファイル (グリッドファイル) に変換するコマンドです。

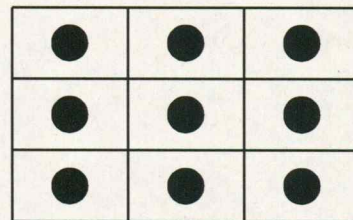
コマンドオプションは以下のようになっています。

```
xyz2grd xyzfile -Ggrdfile -lx_inc[m|c][/y_inc[m|c]] -Rwest/east/south/north  
[-Dxunit/yunit/zunit/scale/offset/title/remark] [-F] [-H]  
[-L] [-Nnodata] [-V] [-Z] [-:] [-b[d]]
```

-R オプションで範囲を指定し、-I オプションで x, y 各方向のグリッド間隔を指定します。基本的にはこれだけで十分なのですが、注意しなければならないオプションがあります。それが -F オプションで、これはグリッドファイルのフォーマットを指定するオプションです。グリッドファイルには 2種類のフォーマットが存在します。下の図を見てください。



グリッドラインフォーマット



ピクセルフォーマット

グリッドラインフォーマットではデータはグリッド交点上に与えられ、ピクセルフォーマットではデータはグリッドの間にあるセルの中心に与えられます。図を見るとわかるように同じ範囲、同じグリッド間隔でもグリッドラインフォーマットとピクセルフォーマットとでは、中に含まれるデータ数が異なっていることがわかります。

グリッドラインフォーマット

$$nx = (xmax - xmin) / x_inc + 1 = 4, \quad ny = (ymax - ymin) / y_inc + 1 = 4$$

ピクセルフォーマット

$$nx = (xmax - xmin) / x_inc = 3, \quad ny = (ymax - ymin) / y_inc = 3$$

-F オプションはデータフォーマットをピクセルフォーマットにするためのオプションです。グリッドデータの書式はどちらを用いても構わないのですが、grdimageなどに用いるにはピクセルフォーマットの方がよいと思います。xyz2grd では -F オプションとデータ個数の関係をよく理解して使用してください。指定を間違えてデータ個数が足りなかったり、多すぎたりといったことはないように気をつけてください。

-N オプションはデータが存在しないグリッド (またはピクセル) に対して、ある特定の値を与えるためのオプションです。(デフォルトは NaN)

GMT コマンドオプション `grdimage` 2次元カラーイメージ

`grdimage` は2次元バイナリの netCDF 形式のファイル (グリッドファイル) から2次元のカラーイメージ (塗りつぶしコンター) を出力するコマンドです。

コマンドオプションは以下のようになっています。

```
grdimage grdfile -Ccptfile -Jparameters [ -Btickinfo ] [ -Edpi ]  
[ -Iintensfile ] [ -K ] [ -M ] [ -O ] [ -P ]  
[ -Rwest/east/south/north[r] ] [ -Sserach_radius ] [ -T[s] ]  
[ -U[dx/dy][label] ] [ -V ] [ -Xx-shift ] [ -Yy-shift ] [ -ccopies ]
```

グリッドファイルを `-C` オプションで指定したカラーパレットファイルに基づいて色分けします。まずは地形データを色分けした例題を示します。



この図は、グリッドファイル、カラーパレットファイル (`-C`)、投影法 (`-J`) のみを指定し、他のコマンドオプションを用いずに描いたいちばん単純な例題です。計算結果を塗りつぶしコンターで表示するような場合には、このようにすれば良いでしょう。しかし、おおまかな変化を表示するには単なる色分けで十分でも、細かな変化を色の変化だけで表示するのは難しくなります。

特に地形を表示するような場合には、どの辺りが高くて、どの辺りが低くなっているのかは色だけで表現できるかもしれませんが、どこが尾根で、どこが谷になっているかというような地形の細かい起伏は色の違いだけでは表現しきれません。地図の世界ではそれを陰影で表現するという方法が良く用いられます。`grdimage` の `-I` オプションがそれに対応するオプションで、イメージに人工的に陰影の効果を追加することができます。`-I` オプションはグリッドデータの傾斜角計算などの結果を用いて、カラーイメージに人工的な陰影や色の濃淡を与えるオプションです。オプションで指定するインテンシ

GMT コマンドオプション `grdimage` 2次元カラーイメージ

`grdimage` は2次元バイナリの `netCDF` 形式のファイル (グリッドファイル) から2次元のカラーイメージ (塗りつぶしコンター) を出力するコマンドです。

コマンドオプションは以下のようになっています。

```
grdimage grdfile -Ccptfile -Jparameters [ -Btickinfo ] [ -Edpi ]  
[ -Iintensfile ] [ -K ] [ -M ] [ -O ] [ -P ]  
[ -Rwest/east/south/north[r] ] [ -Sserach_radius ] [ -T[s] ]  
[ -U[dx/dy][label] ] [ -V ] [ -Xx-shift ] [ -Yy-shift ] [ -ccopies ]
```

グリッドファイルを `-C` オプションで指定したカラーパレットファイルに基づいて色分けします。まずは地形データを色分けした例題を示します。

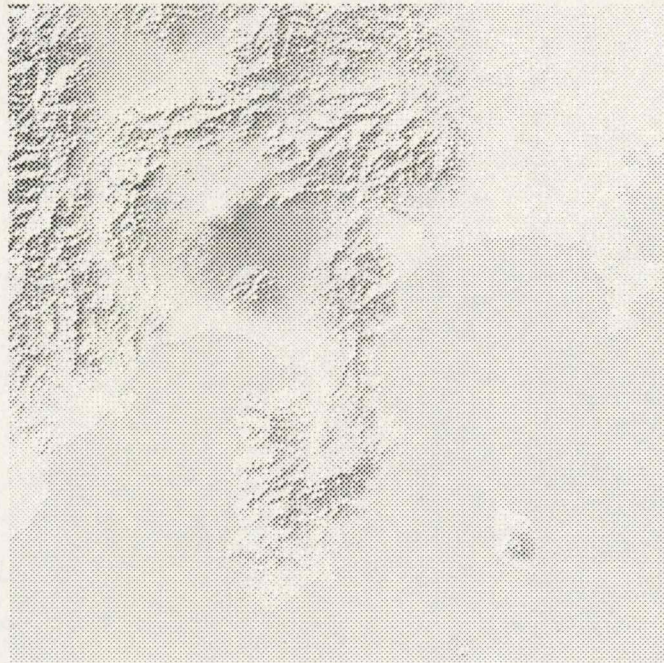


この図は、グリッドファイル、カラーパレットファイル (`-C`)、投影法 (`-J`) のみを指定し、他のコマンドオプションを用いずに描きたいちばん単純な例題です。計算結果を塗りつぶしコンターで表示するような場合には、このようにすれば良いでしょう。しかし、おおまかな変化を表示するには単なる色分けで十分でも、細かな変化を色の変化だけで表示するのは難しくなります。

特に地形を表示するような場合には、どの辺りが高くて、どの辺りが低くなっているのかは色だけで表現できるかもしれませんが、どこが尾根で、どこが谷になっているかというような地形の細かい起伏は色の違いだけでは表現しきれません。地図の世界ではそれを陰影で表現するという方法が良く用いられます。`grdimage` の `-I` オプションがそれに対応するオプションで、イメージに人工的に陰影の効果を追加することができます。`-I` オプションはグリッドデータの傾斜角計算などの結果を用いて、カラーイメージに人工的な陰影や色の濃淡を与えるオプションです。オプションで指定するインテンシ

ティファイルは、カラーイメージを作成するグリッドファイルとディメンジョンがそろっている2次元バイナリの netCDF 形式のファイルでなければいけません。具体的にはx方向の最大値、最小値、グリッド間隔、y方向の最大値、最小値、グリッド間隔がそれぞれそろっていなければいけません。

-I オプションを用いて、地形図に陰影を加えた例を以下に示します。左上 45° から光が当たるようにみせるための傾斜角計算を行いました。



どうですか？ 先ほど示した単純なカラーイメージよりも情報量がかなり増えたと思います。たとえ人工的に作られた色の濃淡でも、人間の目にはきちんとした立体の情報として理解されます。これを利用すれば、わざわざ3次元表示を行わなくても、2次元の平面上に3次元情報を落とし込むことができます。地形の表示のみならず、例えば膜の運動などにも適用できるのではないのでしょうか。

grdimage を用いる際の注意としてもう一つ座標系の問題があります。グリッドファイルはxy座標系のデータですから、図の表示をメルカトルや円錐図法へ変換した場合に問題が生じることがあります。grdimage のデフォルトではポストスクリプトファイルのサイズを小さくしようとして、xy座標系から図の表示に用いる別の座標系へデータ変換を行い、イメージをプロットしようとしています。その際にデータの補間が行われるのですが、緩やかに変化するデータを変換するときには気にならないのですが、値が急変するようは場所では、思いもよらない補間を行い、座標変換する前の図とかけ離れたものになる可能性があります。これを回避するオプションが、-T オプションです。出力しようとするデータ形式に応じて使い分けてください。-T オプションを用いるとポストスクリプトファイルのサイズは間違いなく大きくなります。

-E オプションは画像の解像度を dpi で指定するオプションです。

-M オプションはカラーのイメージをモノクロに変換するときのオプションです。ちょうど白黒テレビでカラー放送を表示するような色変換を行います。

GMT コマンドオプション nearneighbor データの補間

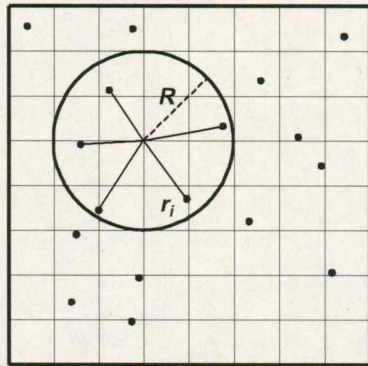
nearneighbor は任意の位置に設定された (x, y, z) のデータから等間隔グリッドのデータ (グリッドファイル) にデータを変換するコマンドです。

コマンドオプションは以下のようになっています。

```
nearneighbor xyzfile -Ggrdfile -lx_inc[m|c][y_inc[m|c]] -Nsectors  
-Rwest/east/south/north -Ssearch_radius[m|c|k|K]  
[-Empty] [-F] [-H] [-L] [-V] [-W] [-:][ -bi[s][n]]
```

nearneighbor は nearest neighbor 法と呼ばれる補間方法を用いて、任意の位置に与えられたデータからグリッド上のデータを求めるコマンドです。この方法は対象とする領域に比較的密にデータが存在している場合に有効なデータ補間方式です。

nearest neighbor 法によるデータ補間の様子を説明します。下の図を見てください。



$$\bar{z} = \frac{\sum_{i=1}^s z_i w_i}{\sum_{i=1}^s w_i}$$
$$w_i = \left(1 + \frac{9r_i^2}{R^2}\right)^{-1}$$

データの補間方法は、計算ポイント (グリッド上) から距離の近いデータポイントの値を用いた平均化作業と考えればよいと思います。探索半径 (R) 内のデータからある個数 (s) を取り出し、そのデータを重みづけ平均したものを、そのグリッドの値とするという方法です。重みづけ係数 (w_i) はグリッドからの距離を用いて上に示した式により計算します。

コマンドオプションの -S が探索半径を指定するオプションです。m をつけると分単位 k をつけると km 単位で指定できます。-N オプションで探索半径内のデータのうち平均化作業に用いるデータの個数を指定します。

重みづけ係数は、先に述べたような式で計算するのがデフォルトですが、(x, y, z) のデータを与える際に、4 番目のデータとして (x, y, z, w) というように外から与えることも可能です。そのときのオプションが -W オプションです。

nearest neighbor 法によるデータ補間は、あくまでも領域に密にデータが存在している場合に有効な補間方法です。定義した全グリッドに対してデータが計算されない場合は探索半径が小さいということですし、探索半径を大きくしすぎると、適切な補間がなされていないということも考えられます。

GMT コマンドオプション surface データの補間

surface は任意の位置に設定された (x, y, z) のデータから等間隔グリッドのデータ (グリッドファイル) にデータを変換するコマンドです。

コマンドオプションは以下のようになっています。

```
surface xyzfile -Ggrdfile -lx_inc[m|c][y_inc[m|c]] -Rwest/east/south/north  
[ -Aaspect_ratio ] [ -Cconvert_limit ] [ -H ] [ -Lllower ] [ -Luupper ] [ -Nmax_iteration ]  
[ -Q ] [ -Ssearch_radius[m] ] [ -Ttension_factor[i|b] ] [ -V ]  
[ -Zover-relaxation_factor ] [ -: ] [ -bi[s][n] ]
```

surface は曲率最小化アルゴリズムをもとにした補間方法を用いて、任意の位置に与えられたデータからグリッド上のデータを求めるコマンドです。あるテンションが与えられた薄板に対して、板上の任意の点をそれぞれに押し下たり、引っ張ったりしたときの板の形状から、グリッドポイント上の値を求める作業と考えればよいと思います。式で書くと、次のようになります。

$$\begin{aligned} z(x_k, y_k) &= z_k && \text{for all data}(x_k, y_k, z_k) \quad k = 1, n \\ (1-t)\nabla^4 z - t\nabla^2 z &= 0 && \text{elsewhere} \end{aligned}$$

上の式の t がテンションレベルを規定する定数です。基本的に $t=0$ の場合が曲率最小化の解となります。この t の値を与えるのが $-T$ オプションです (デフォルトは 0)。通常データの補間を行う場合には 0.2 ~ 0.3 程度の値を選択すると良いと思います。

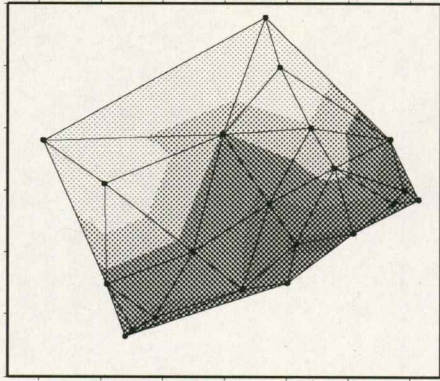
その他のオプションとしては、グリッド間隔が x, y 方向で異なっていた場合、そのアスペクト比を定義するのが $-A$ オプションです。

surface を用いる際には blockmean, blockmedian, blockmode といったコマンドを用いて領域内で変なエイリアシングが起こるのを事前に防いでおいたほうが良いでしょう。blockmean は比較的スムーズなデータの場合に、blockmedian, blockmode は地形データのように起伏に富んだデータの場合に用います。

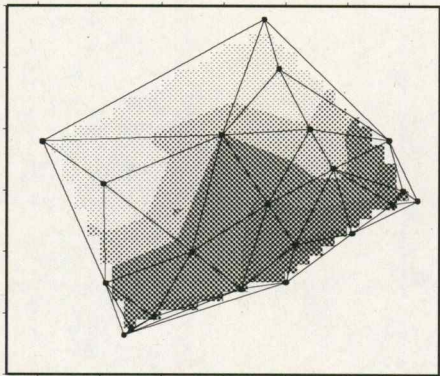
surface が実際にどのような補間を行うのか例題を示します。

例題では blockmean などのプリプロセスは行わずにデータ補間を行いました。例題中「元のデータの pscontour」、「triangulate」としたのが任意のデータに対して三角形要素を自動生成して線形補間した場合の例です。これらを基本形として、surface のテンションレベルを $-T$ オプションを用いて、いろいろと変化させて補間した結果と比較しています。コンターの形状で比較すると、先にも述べたとおり、テンションレベルを 0.25 とした場合と三角形要素を自動生成して線形補間した場合とが比較的よく対応していることがわかるとおもいます。

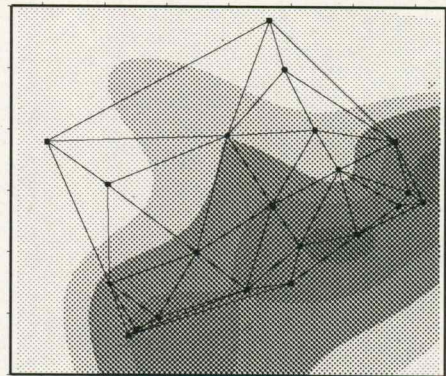
3次元表示した結果もあわせて示します。テンションレベルの違いにより、補間の様子がかなり異なっているのがわかると思います。



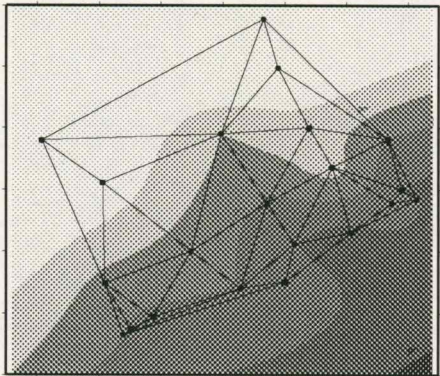
元のデータの pscontour



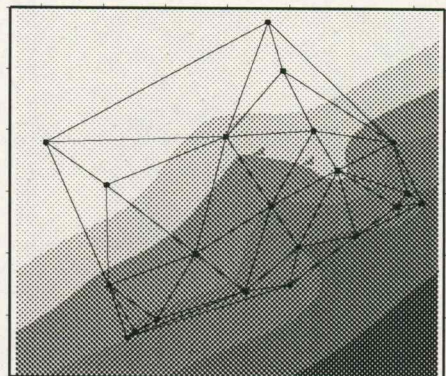
triangulate



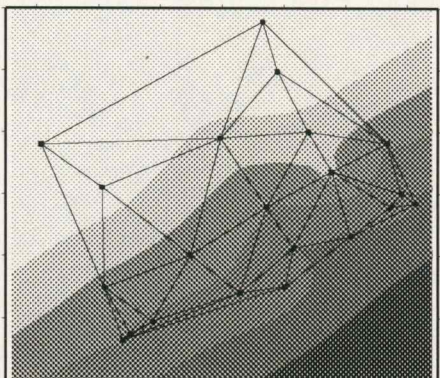
surface (-T0.00)



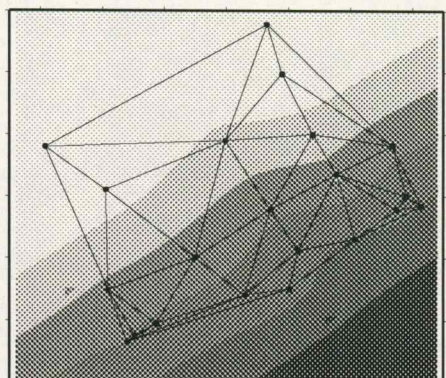
surface (-T0.25)



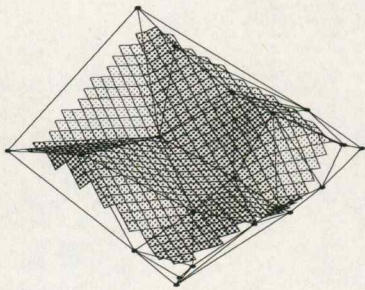
surface (-T0.50)



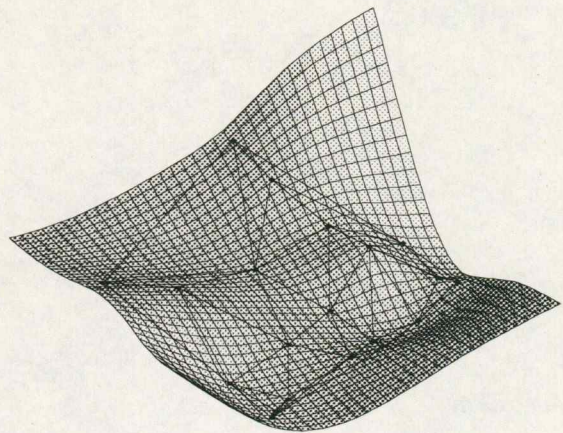
surface (-T0.75)



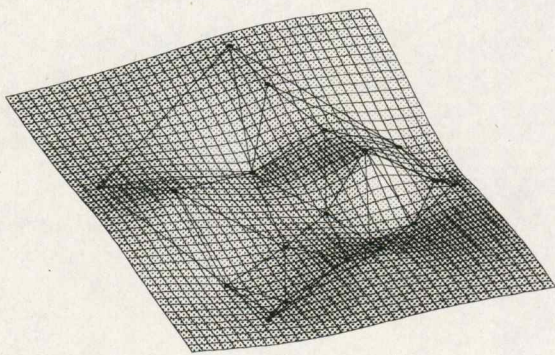
surface (-T1.00)



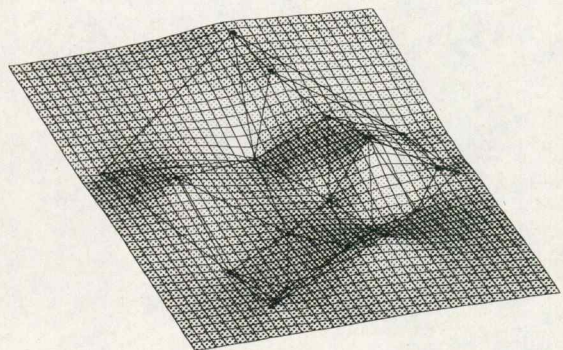
triangulate



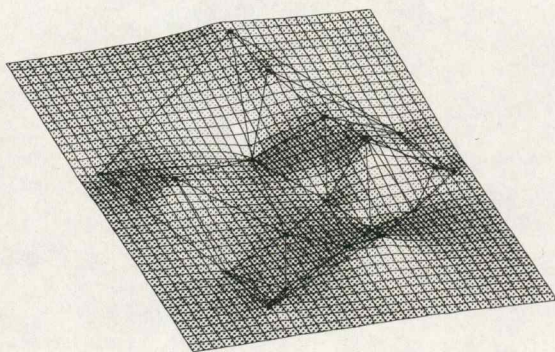
surface (-T0.00)



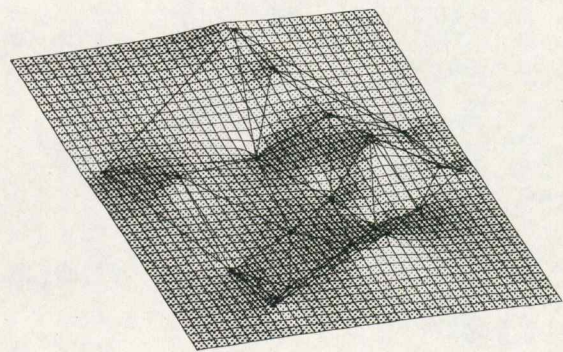
surface (-T0.25)



surface (-T0.50)



surface (-T0.75)



surface (-T1.00)

GMT コマンドオプション `grdview` グリッドファイルの3次元表示

`grdview` は2次元バイナリの `netCDF` 形式のファイル (グリッドファイル) から3次元表示のカラーイメージ、コンター、メッシュ図を出力するコマンドです。

コマンドオプションは以下のようになっています。

```
grdview relief_file -Jparameters [ -Btickinfo ] [ -Ccptfile ]  
  [ -Eview_az/view_el ] [ -Gdrapefile ] [ -Iintensfile ] [ -K ] [ -L ]  
  [ -Nlevel[r/g/b] ] [ -O ] [ -P ] [ -Qtype ] [ -Rwest/east/south/north/zmin/zmax[r] ]  
  [ -Ssmooth ] [ -T[s] ] [ -U[dx/dy/][label] ]  
  [ -V ] [ -Wtype/pen ] [ -Xx-shift ] [ -Yy-shift ] [ -Zzlevel ] [ -ccopies ]
```

`grdview` の出力形式を指定するオプションは `-Q` オプションです。 `m` オプションをつけると (`-Qm`)、 `relief_file` で指定したグリッドファイルに対して `-W` オプション (`-Wm pen`) による線種で3次元のメッシュ図を作成します。 `s` オプションをつけると (`-Qs`)、同様に3次元表面に対する `-C` オプション (カラーパレットファイル) によるカラーイメージや、 `-W` オプション (`-Wc pen`) による線種でコンターを出力します。 `m` オプションと `s` オプションは合せて (`-Qsm`) 使用することもできます。また、 `i` オプションをつけると (`-Qi`)、3次元表面の図を希望の解像度によるラスタライズ (デフォルトは100dpi) を行ってから表示します。

3次元表面のカラーイメージを作成するとき、 `-G` オプションを用いると `relief_file` で指定したグリッドファイルに対して3次元表面を生成し、その表面上に `drapefile` で指定したグリッドファイルによる `relief_file` とは別のデータを用いた色分け表示をすることも可能です。

`-G` オプションに関する例題を次のページに示します。上の図が `-G` オプションを用いた例です。色分けは、3次元表面を生成しているデータとは別のデータで行っています。また、この図の例では、 `-I` オプションを用いて人工的な陰影 (色の濃淡) を与えて立体感をもたせる効果も用いています。下の図は、上の図の3次元表面をメッシュ図にした例です。

3次元表示を行う際の視点を設定するオプションが `-E` オプションです。2つの角度で視点位置を規定します。角度は、北 (`y` 軸プラス方向) からの回転角 (反時計回り) と水平面 (`xy` 平面) から鉛直方向への回転角で指定します。デフォルトは、 `-E180/90` で、視点は水平面内では南 (`y` 軸マイナス方向) に、鉛直面内では `xy` 平面から 90° の位置 (つまり真上) にあることになっていますから、2次元の `xy` プロットと同じことになっています。

`grdimage` の場合と同様に、グリッドファイルは `xy` 座標系のデータですから、図の表示をメルカトルや円錐図法にした場合、座標変換の問題が生じることがあります。 `xy` 座標系から図の表示に用いる別の座標系へデータ変換を行う際にデータの補間が行われず、値が急変するような場所での思いもよらない補間を回避するオプションが、 `-T` オプションです。出力しようとするデータ形式に応じて使い分けてください。

grdview による 3次元プロット

