RotateGrid

An open source IDL script to transform Geographical longitude and latitude to coordinates of rotated grids for visualization and data extraction of Climatic Model outputs using GRADS or FERRET

Javier G. Corripio
IfU - ETH Z urich
corripio@ifu.baug.ethz.ch
Online version at http://www.arolla.ethz.ch/softwmodels.html

1 Introduction

Many numerical weather prediction and long term climatic models use rotated grids with an arbitrary north pole that differs from the geographical pole. This is a common solution to avoid the problem of diminishing linear distances along the longitudes near the pole. As the horizontal grid spacing get smaller at higher latitudes, the Courant-Friedrichs-Lewy (Courant et al. 1928) stability criteria is not fulfilled and the computation suffer from numerical instabilities. A common solution is to use rotated grids, such as the Arakawa grid (Arakawa 1997).

Most of the models involved in the PRUDENCE Project (Prediction of Regional scenarios and Uncertainties for Defining EuropeaN Climate change risks and Effects) use these types of grids. Their visualization in GRADS is problematic, as GRADS does not handle curvilinear grids. FERRET can visualize these type of grids, e.g. `SHADE variable, lon, lat`, but cannot extract data from a specific region. The command `SAMPLEXY_CURV` will allow this in a future version. At present data could be extracted for a specified region if the coordinates are given in the reference system of the rotated grid. The scripts provided were specifically developed for the visualization of the PRUDENCE project outputs in the free software GRADS (http://www.iges.org/grads/), or FERRET (http://ferret.pmel.noaa.gov/Ferret/) though it should be possible to apply them to different grids and other software.

The climatic model outputs are given as netCDF files. The netCDF (network Common Data Format) is a self-describing scientific data access interface and library developed at the Unidata Program Center in Boulder, Colorado. The array-oriented netCDF interface and library use XDR (eXternal Data Representation) to make the data format machine-independent. The Unidata web page http://www.unidata.ucar.edu/packages/netcdf/ provides a full description of the format, specification, libraries and tools for creation and manipulation of this data format.

The scripts consist of several IDL routines and a graphical user interface (GUI) to facilitate their use. It is designed to be a user friendly interface. A pre-compiled executable is also provided. This executable requires the installation of the free software IDL Virtual machine, available from ITT visual technologies, formerly RSI: http://www.ittvis.com/idlvm/index.asp

2 Running the scripts

The scripts are designed to produce a ascii .txt file with the rotated coordinates, plus the grads or ferret command to select the region circumscribed by those coordinates. In GRADS it would also be necessary to use the eXternally-Described Files (XDF) interface. It would require a .ddf file describing the x and y coordinates, and in some cases the variable definitions e.g.

```
DSET ^oro.DMI.50km.nc
XDEF rlon
YDEF rlat
VARS 1
oro 1 84,94  orography
ENDVARS
```

The IDL code is provided in 4 files:

```
Rotateaxis_y.pro
Rotateaxis_z.pro
rotategrid.pro
rotategridw.pro
```

plus an executable called `rotategridw.sav`.

From a fully licensed IDL, open, compile and run `rotategridw.pro`. A GUI will open as shown in figure 1. The inputs are either two comma-separated arrays of longitudes and latitudes, or a text file containing 1 header line and two fields of latitudes and longitudes as in `lonlat.txt`. There is no record number limit

Additional common inputs are the longitude and latitude of the rotated pole. The coordinates of the rotated pole can be obtained from the original model description, from the summary page at the prudence project (http://prudence.dmi.dk/public/DDC/areas.html) or using ncdump or similar. Table 1 provides a summary of coordinate for PRUDENCE models.

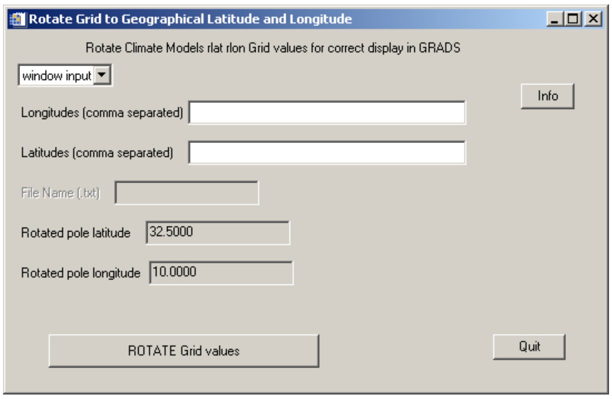

Figure 1. IDL GUI to enter the input data for the grid rotation

If *Window Input* is selected, the inputs are:

```
LONGITUDES: comma-separated array of geographical longitudes
LATITUDES: comma-separated array of geographical latitudes
```

If *File Input* is selected enter the name of an ascii file containing 1 line header and two fields of longitudes and latitudes. There is no record limit.

Sample file *lonlat.txt*:

```
lon    lat
-10.0 35.0
-11.0 44.0
4.0   44.0
4.0   35.0
```

Whether file or window input is selected for the longitudes and latitudes, the common inputs are:

```
ROTATED POLE LONGITUDE:
   Geographical longitude of the rotated north pole
ROTATED POLE LATITUDE:
   Geographical latitude of the rotated north pole
```

The coordinates of the rotated pole in the model of the prudence project are given in table 1 or on the area description page: http://prudence.dmi.dk/public/DDC/areas.html

2.1 Examples

2.1.1 Example 1

Select a region including the Iberian Peninsula from the ETH orography data `lsmoro.ETH.nc`:

The geographical coordinates of the vertices of a rectangle containing the region are (-10., 35.); (-11., 44.); (4., 44.)and (4., 35). The ETH rotated pole is at (10.,32.5). Thus the data input to the rotetegrid GUI should as shown in figure 2

The results are written to file `projected_coordinates.txt` and shown graphically in figure 3

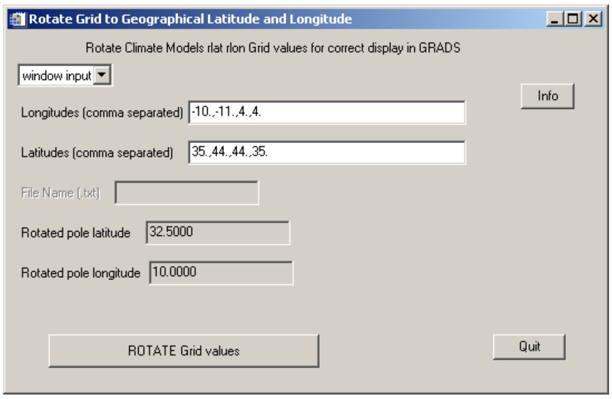

Figure 2. GUI with input data for example 1

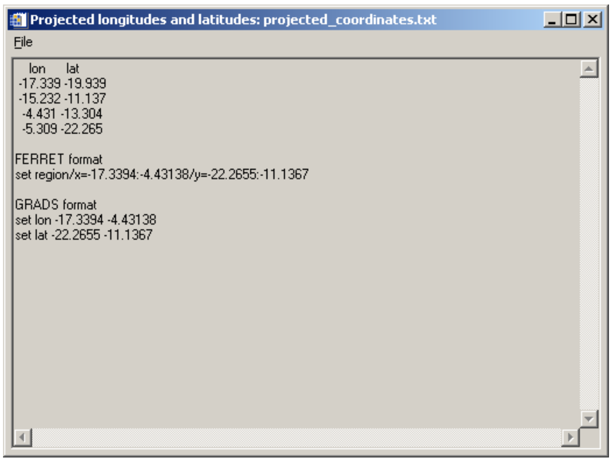

Figure 3. IDL script results of example 1

Now, we can display the results with the commands:

In GRADS:

file lsmoro.ETH.nc.ddf:

```
DSET ^lsmoro.ETH.nc
XDEF rlon
YDEF rlat
VARS 2
lsm 1 1,1,75,65 land surface mask
oro 1 1,1,75,65 orography
ENDVARS
```

```
   $gradsnc
ga->xdfopen lsmoroETH.nc.ddf
```

```
   ga->set lon -17.3394 -4.43138 set lat -22.2655 -11.1367
```

Or in FERRET:

```
use lsmoro.ETH.nc
set region/x=-17.3394:-4.43138/y=-22.2655:-11.1367
shade lsm
```

Will produce the inset of figure 4

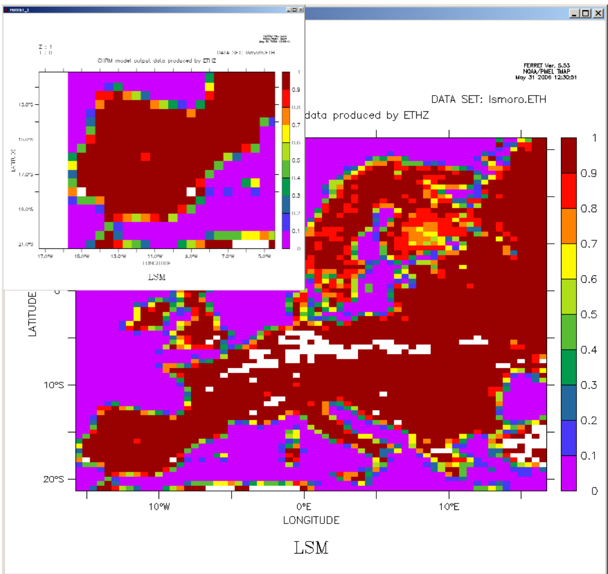

Figure 4. Graphic output of example 1

2.1.2 Example 2

Plot the time-series of precipitation for one single location. We do not consider whether it should be inetrpolated/weighted/averaged with surrounding cells. For a location in the Alps du Valais at 45.9N, 7.5E on the DMI model (Pole at 27W, 37N) we have the inputs in figure 5, the results of the script in figure 6 and the plot of the selected data in figure 7

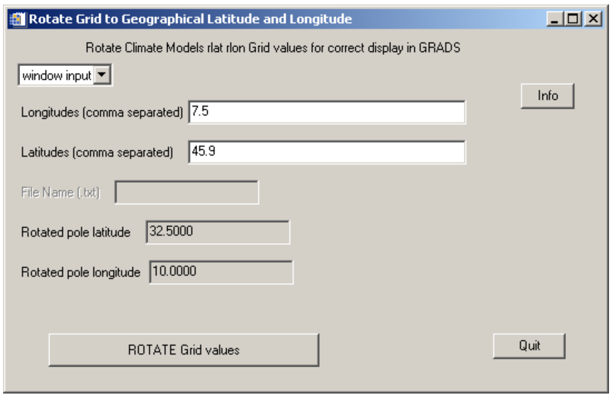

Figure 5. GUI with input data for example 2

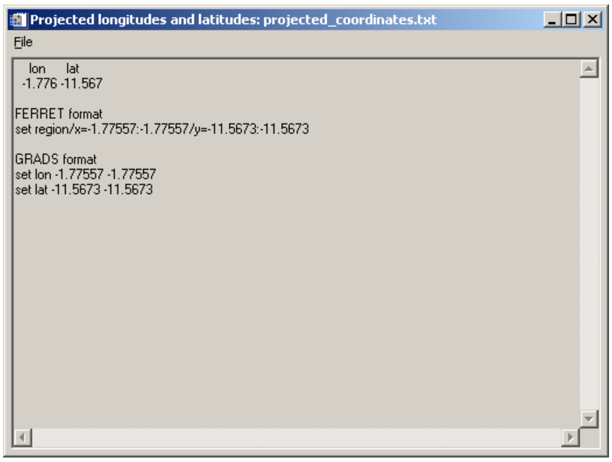

Figure 6. IDL script results of example 2

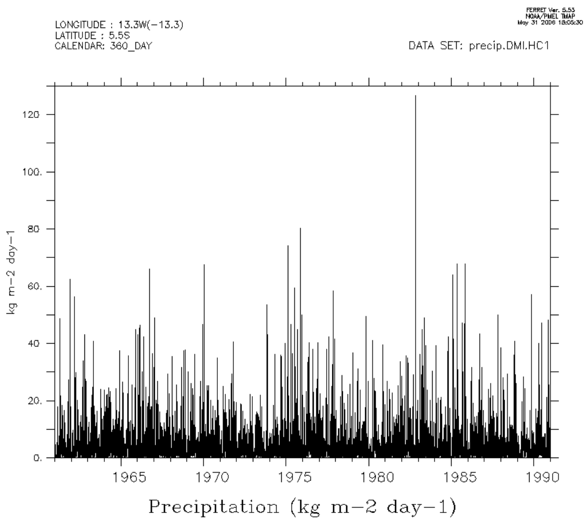

Figure 7. Graphic output of example 2

3 Extracting netCDF information

There are several ways of extracting metadata from the netCDF files. A straightforward way is using the UNIDATA utility `ncdump`, available from
http://www.unidata.ucar.edu/software/netcdf/guide_12.html, e.g.
`ncdump -c filename.nc >dumptextfile.txt`.

Other possibilities using IDL is S. Rupert's routine cdf2idl.pro, availble at the National Science Digital Library:
http://comm.nsdl.org/viewcvs/viewcvs.cgi/avc/visualization_tools/cdftoidl.pro?rev=HEAD

Or enquiring directly with the IDL commands, e.g. `IDL>n=ncdf_open(filename)`

```
IDL>idvar1= NCDF_VARID(n, 'rlat')
IDL>ncdf_varget,n,idvar1,rlat
IDL>idvar2= NCDF_VARID(n, 'rlon')
IDL>ncdf_varget,n,idvar2,rlon
IDL>ncdf_close,n
IDL>print,'rlat',rlat
IDL>print,'rlon',rlon
```

4 Coordinate rotation

The rotation of the coordinates can be described as two successive rotations of the original, geographic coordinate system e.g. (Goldstein 1980). Firstly a rotation around the axis Z an angle equal to the latitude of the rotated pole (ϕ), and then a rotation around the axis Y an angle equal to the inverse of the colatitude ($\theta = \pi - \lambda$, where λ is the latitude). The process is explained graphically in figure 8

The rotation can be expressed as $P' = R_{y(\beta)} R_{z(\gamma)} P$:

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{pmatrix} \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ z \end{pmatrix} \tag{1}$$

Where P are the x, y, z coordinates with respect to the geographical pole and P' are the x', y', z' coordinates with respect to the rotated pole, β is equal to the latitude and γ is equal to the inverse of the colatitude.

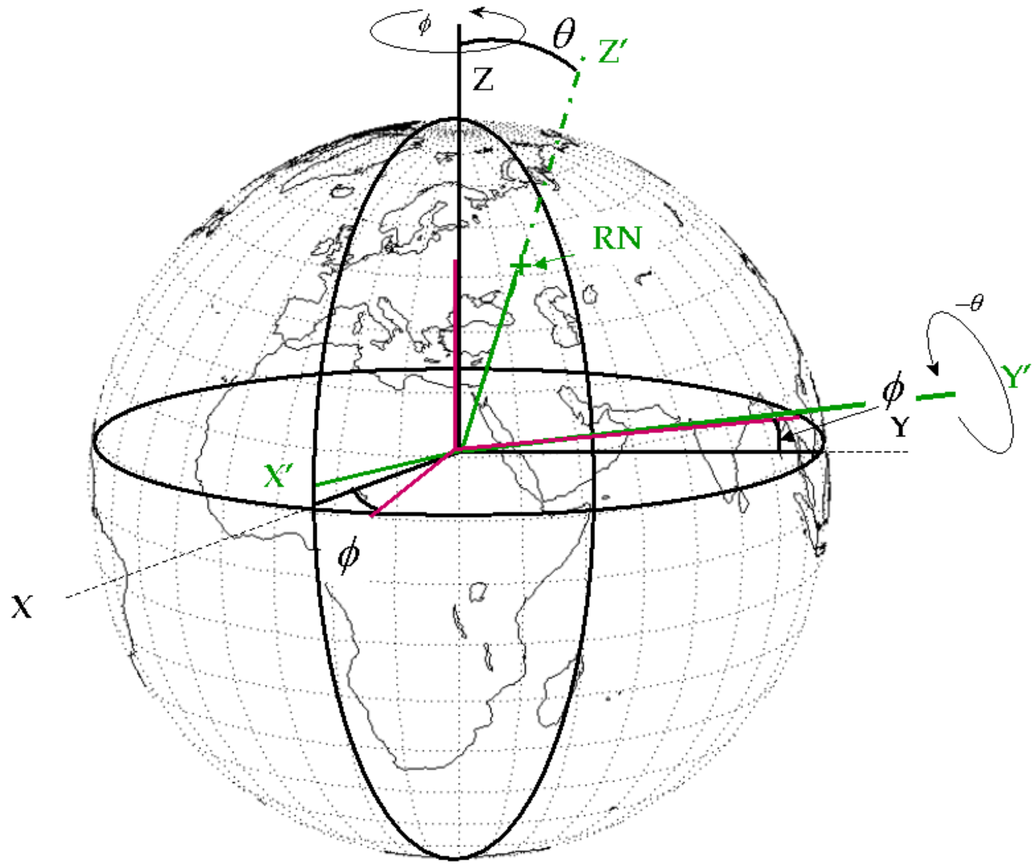

Figure 8. Schematic representation of the rotation of the coordinate reference axis from Geographic longitude and latitude to rotated grid longitude and latitude. The rotation follows two steps. First step is a rotation of the original reference system (XYZ, inb lack) around axis Z an angle ϕ, the longitude of the rotated pole. The original geographical pole is at the axis Z. The second step is a rotation of the previously rotated axes (red) around axis Y an angle θ, equal to the inverse of the colatitude of the rotated pole. Thus we obtain the rotated reference system X'Y'Z' (in green), with rotated north pole at RN.

Appendix

Table 1. Rotated North Pole for the Prudence project models

model	**longitude**	**latitude**
CRU domain	0	90
HadAM	0	90
HadAM winds	0	90
HadRM	10	38
HadRM winds	10	38
DMI 50km	27	37
DMI 25km	27	37
SMHI 50km	25	32
SMHI 25km	25	32
ETH	10	32.5
MPI	10	32.5
MPI 3002	10	32.5
GKSS	10	32.5
KNMI	23	28
UCM	N/A	N/A
ICTP	N/A	N/A
METNO	0	25

References

Arakawa, A.: 1997, Computational design for long-term numerical integration of the equations of fluid motion: Two-dimensional incompressible flow. part I, *Journal of Computational Physics* **135**(2), 103–114.

Courant, R., Friedrichs, K. and Lewy, H.: 1928, uber die partiellen differenzengleichungen der mathematischen physik, *Mathematische Annalen* **100**(1), 32–74. Translated as: ”On the partial difference equations of mathematical physics,” IBM Journal of Resarch and Development 11 (1967), 215-234.

Goldstein, H.: 1980, *Classical mechanics*, 2nd edn, Addison-Wesley, Reading, Mass. 672 p.