

Spatial representation with the Python Basemap and PyNGL libraries

Nicolas Barrier

UMR MARBEC

nicolas.barrier@ird.fr

May 9, 2019



Description

The [Basemap](#) library is the mapping extension of the standard graphic library

Pros

- It is now quite easy to install by using [Anaconda](#)
- Very simple to use
- Multiple map backgrounds can be used (from TIFF image for instance)
- Comes with usefull tool (distance calculations, etc.)

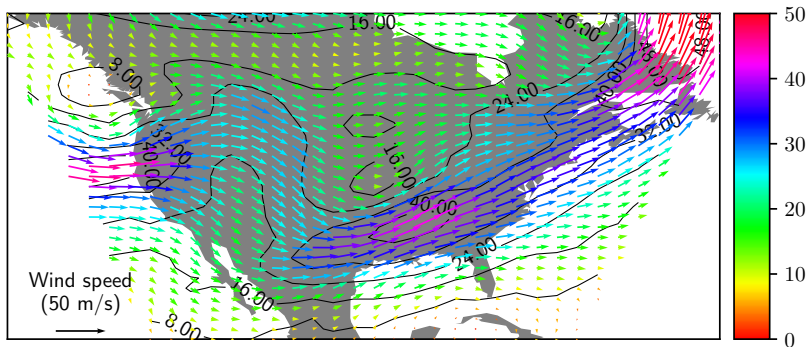
Cons

- Rendering is not always perfect
- No Masked Lambert Conformal projection
- Need to switch from geographical to map coordinates
- Need to navigate between Matplotlib and Basemap methods/functions

Example

```
1 import matplotlib.pyplot as plt
2 from mpl_toolkits.basemap import Basemap
3
4 # map initialisation
5 m = Basemap(llcrnrlon=np.min(lon), llcrnrlat=np.min(lat),
6             urcrnrlon=np.max(lon), urcrnrlat=np.max(lat),
7             projection='cyl', resolution='c')
8
9 # conversion from geo. to map coordinate
10 lon2d, lat2d = np.meshgrid(lon, lat) # needs 2D coord. arrays
11 x, y = m(lon2d, lat2d)
12
13 # map background using basemap function
14 m.fillcontinents(color='gray', lake_color='gray')
15
16 # drawing contour lines using Basemap function
17 cs = m.contour(x, y, z, colors='k', linewidths=0.5)
18 plt.clabel(cs, fmt='%.2f')
19
20 # drawing quiver plot
21 # use of pyplot function !!!
22 q = plt.quiver(x, y, u, v, z, cmap=plt.cm.get_cmap('hsv'), scale=1000, zorder
23               =1000)
24 cb = m.colorbar(q)
25 keys = plt.quiverkey(q, -131, 21, 70,
26                      'Wind speed\n(50 m/s)', coordinates='data')
```

Example



Description

The NCAR Graphic library is a very powerfull library for graphics, especially mapping. It is available in Python through the [PyNGL](#) library.

Pros

- It is now quite easy to install by using [Anaconda](#)
- Very nice rendering
- The drawing of velocity fields is very easy (curlyvectors, etc.)
- Many examples are provided in the [gallery](#)

Cons

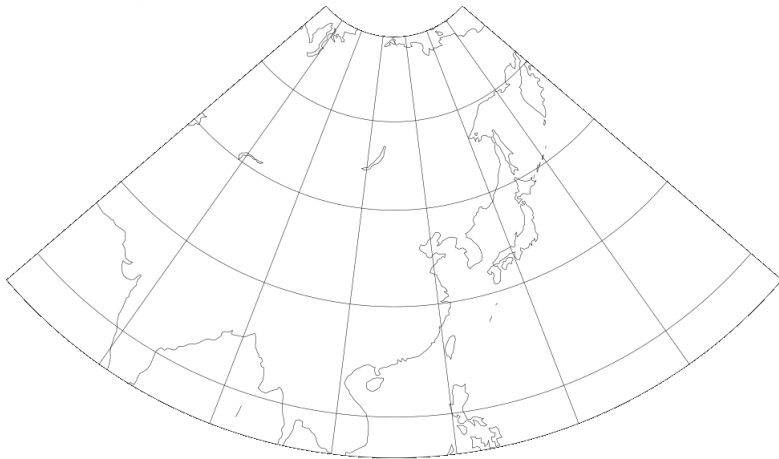
- Not compatible with the default Python graphic library [Matplotlib](#)
- Heavy code! To use only for paper plots (not for working material)
- Only 256 colors can be used

Simple map

```
1 import Ngl
2
3 wks_type = "png"
4 wks = Ngl.open_wks(wks_type, "conmask1c")
5
6 res = Ngl.Resources()
7
8 res.nglDraw = False
9 res.nglFrame = False
10
11 res.mpProjection = "LambertConformal"
12
13 res.mpLimitMode = "LatLon" # limit map via lat/lon
14 res.mpMinLatF = 10. # map area
15 res.mpMaxLatF = 75. # latitudes
16 res.mpMinLonF = 60. # and
17 res.mpMaxLonF = 165. # longitudes
18
19 res.nglMaskLambertConformal = True
20 res.nglMaskLambertConformalOutlineOn = True
21
22 res.tiMainString = "Map example"
23 res.tiMainFontHeightF = 0.010
24
25 map = Ngl.map(wks,res)
26
27 Ngl.draw(map)
28 Ngl.frame(wks)
```

Simple map

Map example



More complicated example

366 lines of code later (cf. `plot_currents_pyngl.py`)...

