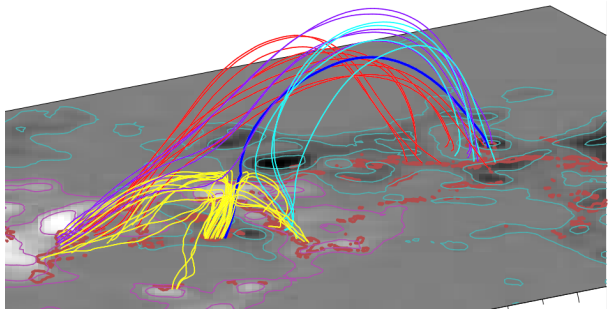
Magnetic field extrapolations: Hands-on session



Gherardo Valori

UCL MSSL

4th SOLARNET Summer School on Solar MHD and Reconnection

13-19 April 2016, MSSL

1. How to

GREEN STICKER: gotcha, go on!
RED STICKER: no luck, help needed

Let's give it a try:

1. Open an IDL-SSW session
2. Point the variable maindir to your /your_path_to/FFF_hands_on/
3. Set IDL path to the script folder
4. Check the path: got a 'yes'?

```
maindir="/your/path/to/FFF_hands_on/"
!path=maindir+"scripts:"+!path
is_path_ok
```

To follow the session copy and paste commands onto the IDL-SSW session / browser

- from extrapolation_hands_on.idl (check slide reference number for relevant section)
- from your copy of these slides

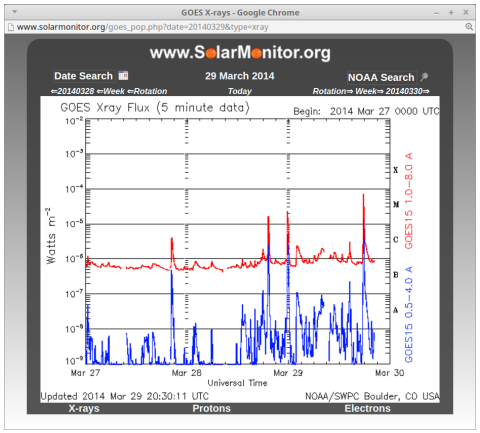
Requires: IDL, SolarSoft/HMI, Paraview, and the FFF_extrapolation tree of the school's data

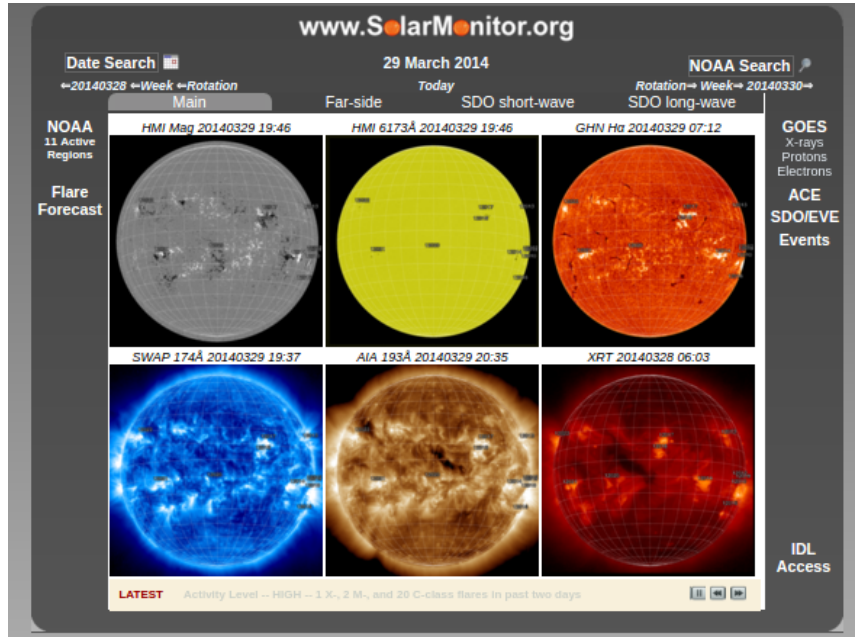
UCL MSSL

2. Solar Monitor

X1 Flare event on **SOL2014-03-29T17:48**

- Go to Solar Monitor
  http://www.solarmonitor.org/index.php?
- Browse to date
  http://www.solarmonitor.org/index.php?date=20140329
- Check list events: X1 class flare was observed from NOAA **AR12017** located at **N10W32 (503",259")**
- Have a look at the different images and info, *e.g.*, GOES SXR





Today's/Yesterday's NOAA Active Regions

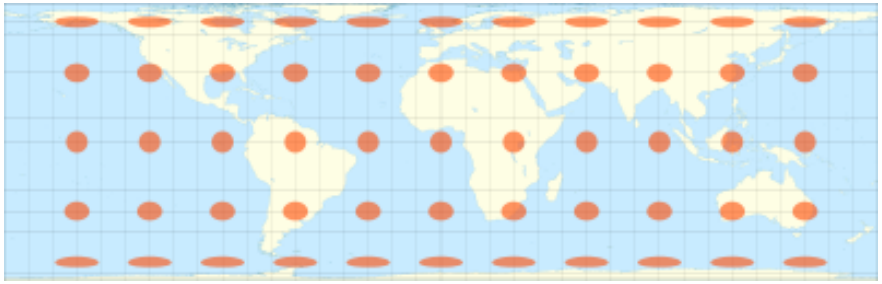
NOAA Number	Latest Position	Hale Class	McIntosh Class	Sunspot Area [millionths]	Number of Spots	Recent Flares
12010	S14W91 (931",-232")	βγ/βγ	Dao/Dao	0080/0080	10/17	C1.0(13:18) C1.2(11:31) C1.4(09:53) C1.4(05:57) C1.0(05:00) C1.0(04:06) / C1.0(20:57)
12014	S15W58 (788",-190")	βγ/βγ	Ehi/Ehi	0290/0290	22/35	C1.1(16:20) / -
12016	S28W89 (848",-446")	β/	Cro/	0030/	03/	-
12017	N10W32 (503",259")	βδ/β	Dao/Dsi	0150/0160	14/16	X1.0(17:35) C3.3(14:26) C1.5(12:41) C1.7(10:48) C2.1(07:50) / M2.6(23:44) C1.0(23:14) M2.0(19:04) C2.3(17:56) C1.2(17:30) C1.0(16:03) C1.1(12:55) C1.1(08:31)
12018	N03W28 (452",148")	β/β	Cro/Dsi	0030/0070	08/09	-
12021	S15E38 (-573",-162")	β/-	Dao/---	0040/----	07/--	-
12022	N16E52 (-729",329")	α/-	Hsx/---	0060/----	01/--	-
12012	S09W91 (948",-151")	/	/	/	/	-
12013	N15W79 (911",267")	/	/	/	/	-
12019	S10W91 (945",-167")	/β	/Bxo	/0010	/03	-
12020	S14E02	/β	/Bxo	/0010	/05	-

UCL MSSL

3. SDO/HMI/SHARP

For this session we use the so-called SHARP data series from

Solar Dynamic Observatory / Helioseismic and Magnetic Imager
http://hmi.stanford.edu/

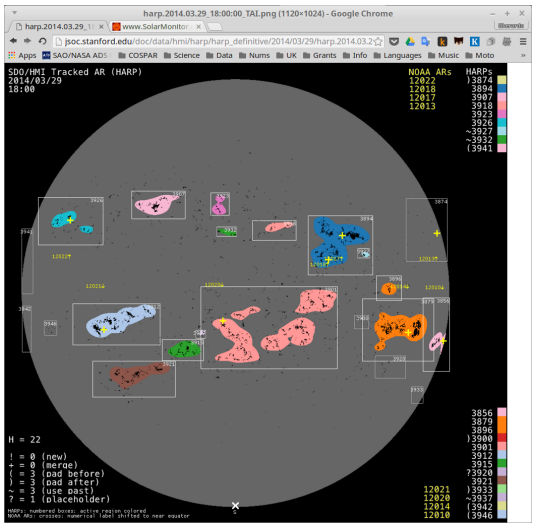


Wikipedia example of CEA

HMI Vector Magnetic Field Data Products

Principal HMI Vector Field Pipeline Data Products			
Vector Magnetic Field Product	HMI Data Series	**Type	Availability/Status
Stokes Parameters	hmi.S_720s	Full Disk / Definitive	Since 1 May, 2010
VFISV Milne-Eddington Inversions	hmi.ME_720s_fd10	Full Disk / Definitive	Since 1 May, 2010
NRT VFISV Milne-Eddington Inversions	hmi.ME_720s_fd10_HARP_nrt	HARP / NRT	Since 1 May, 2010
NRT HARP Vector Field & Space-Weather	hmi.sharp_720s_nrt	HARP / NRT	September, 2012 onward
NRT Projected HARP Vector Field & Space-Weather	hmi.sharp_cea_720s_nrt	HARP / NRT	September, 2012 onward
HARP Vector Field & Space Weather Quantities	hmi.sharp_720s	HARP / Definitive	Since 1 May, 2010 • coverage
Remapped HARP Vector Field & Space Weather	hmi.sharp_cea_720s	HARP / Definitive	Since 1 May, 2010 • coverage
Full Disk Vector Magnetic Field	hmi.B_720s	Full Disk / Definitive	Since 15 Jan, 2014 • coverage
Type			
** **Full Disk** = Entire 4096**2 image		** **Definitive** = Available ~30 days after observation	
** **HARP** = HMI Active Region Patch		** **NRT** = Available in near real time	

http://jsoc.stanford.edu/jsocwiki/VectorMagneticField



SHARP is the simplest one (Bobra et al., 2014)

- 720 sec averaged SP measurements
- Inversion using VFISV with Milne-Eddington model of the solar atmosphere (Borrero et al., 2011)
- Cylindrical-Equal-Area projection (pixel size = 0.03 deg $\simeq$ 0.5'' at disk center) (Sun,X. arXiv:1309.2392v1)
- 180deg ambiguity resolved by Minimum Energy Code (Leka et al., 2009).

We need the HARP number corresponding to AR12017 on 2014 March 29. Use the catalogue

http://jsoc.stanford.edu/doc/data/hmi/harp/harp_definitive/2014/03/29/harp.2014.03.29_18:00:00_TAI.png

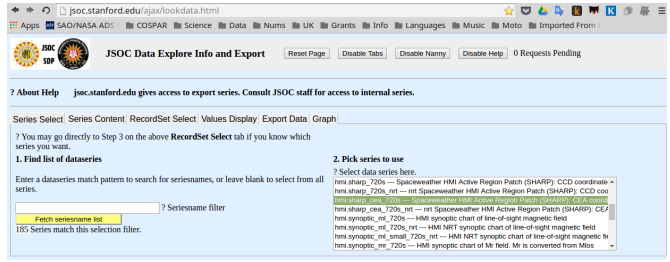
And the HARP number is ....
Note that is an AR complex rather than a single one!

UCL MSSL

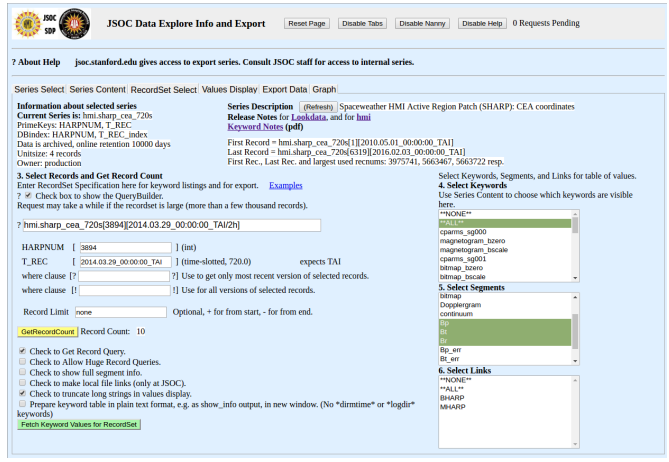
4. JSOC download

http://jsoc.stanford.edu/ajax/lookdata.html

- On Tab: Series select
- Click on 'Fetch seriesname list'
- Select 'hmi.sharp_cea_720s', the next tab (Series Content) opens automatically



- On Tab: Series Content
- Spun 'Check box to show the 'QueryBuilder' to learn how the request is built, *e.g.*, fill in the HARP number
- Fill in 'hmi.sharp_cea_720s[3894][2014.03.29_00:00:00_TAI/2h]' for two hours data at full-time cadence of 12 minutes
- Click on 'GetRecordCount' to know how many records you have selected
- Select 'All' in Keywords: information about data and processing
- Select 'Bp, 'Bt', 'Br' in Segments: which data are actually required for each record
- Click on 'Fetch Keywords Values for RecordSet', the next tab (Values display) opens automatically



For additional details refer to

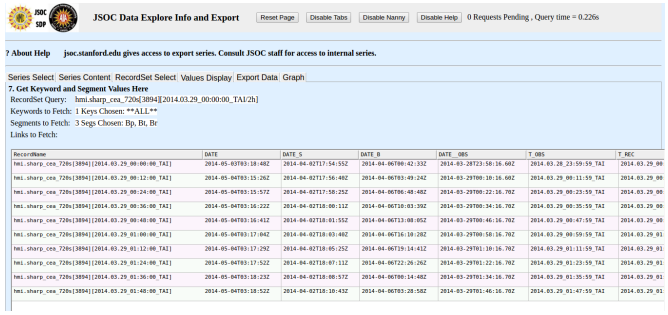
https://www.lmsal.com/sdodocs/doc/dcur/SDOD0060.zip/zip/entry/

UCL MSSL

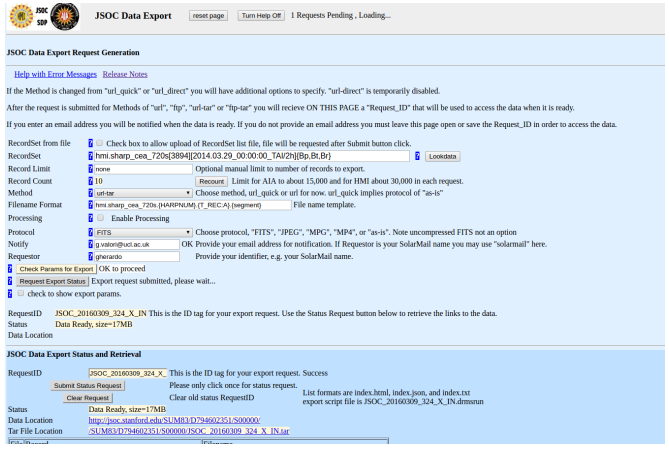
4. JSOC download

http://jsoc.stanford.edu/ajax/lookdata.html

- On Tab: 'Values display'
- A summary of the chosen content is shown
- click 'Export' to open the 'Export window'



- On 'Export window'
- Select 'url-tar' for fetching a single compressed file
- Fill in a Requestor name and email
- Click 'Check param for export'
- if all fine, click the -now green- 'Submit Export Request'
- The 'Submit statue request' and an email tell you when the file is ready to be downloaded



For additional details refer to

https://www.lmsal.com/sdodocs/doc/dcur/SDOD0060.zip/zip/entry/

UCL MSSL

4. JSOC download

http://www.lmsal.com/solarsoft/jsoc/ssw_jsoc_routines.html

One can use directly SSW, but instructions are a bit cryptic

```
;   ser=ssw_jsoc(/SHOW_SERIES,filter='hmi.sharp_cea_720s')
;   serstr=ssw_jsoc(/SERIES_STRUCT,DS=ser.names(1).name)
;   exp_request=ssw_jsoc(/EXPORT,serstr=serstr,ds=serstr.
      interval.lastrecord,method='url_quick')
```

Here, we use a local copy

```
datadir =maindir+ "data/fits/"                                  ;where fits are
hmifile ="hmi.sharp_cea_720s.3894.20140329_013600_TAI."         ;CHANGE TO PREFLARE
filenm  =datadir+hmifile+["Br","Bp","Bt"]+".fits"               ;Add suffix array
read_sdo, filenm ,index,data                                    ;use SSW reading routines
```

- index contains information about the observation
- data contains the field components in the loaded order

```
help,index,data,/struct
```

5. Build vmgm

Extract essential infos from index

```
nx=(index.naxis1)[0]      ;The grid is uniform in the
ny=(index.naxis2)[0]      ; CEA reference system
dx=(index.cdelt1)[0]      ;Pixel size in CEA-deg
dy=(index.cdelt2)[0]      ;Pixel size in CEA-deg
ixcen=(index.crpix1)[0]   ;index x central pixel in CEA-deg
iycen=(index.crpix2)[0]   ;index y central pixel in CEA-deg
xcen=(index.crval1)[0]    ;coord x central pixel in CEA-deg
ycen=(index.crval2)[0]    ;coord y central pixel in CEA-deg
units="CEA-deg"           ;CEA degrees
```

```
;Build axes
x_mgm=fltarr(nx) & for i=0,nx-1 do x_mgm[i]=xcen+ (-ixcen)*dx + i*dx
y_mgm=fltarr(ny) & for i=0,ny-1 do y_mgm[i]=ycen+ (-iycen)*dy + i*dy
```

Build axes ...

... and field.
Note the sign of B_y

```
b_mgm=fltarr(nx,ny,3)
b_mgm[*,*,2]= data[*,*,0]   ;Br= Bz   As of Eq.14 of X.Sun 2013
b_mgm[*,*,1]=-data[*,*,2]   ;Bt=-By   arXiv:1309.2392v1
b_mgm[*,*,0]= data[*,*,1]   ;Bp= Bx
```

Panic button:

```
restore, maindir+ "data/one_vmgm.sav",/v
```

UCL MSSL

5. Plot vmgm

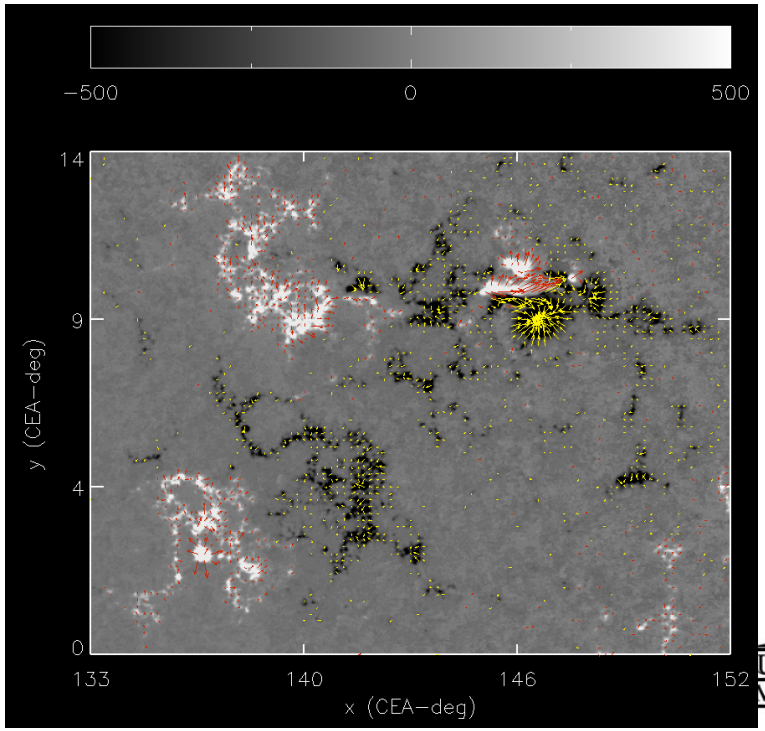
```
sizesym=2.5 & sizechar=2.5 & linethick=3. & thickchar=1.5   ;Set line/character size
ixmin=0 & ixmax=nx-1 &  iymin=0 & iymax=ny-1                ;Field of view to plot
ywind=1000                                                  ;Vertical window extension
ar=(0.75-0.15)/(0.95-0.15)*float(ixmax-ixmin-1)/float(iymax-iymin-1) ;Set plot aspect ratio
xwind=fix(ywind*ar)
window,1,xsize=xwind,ysize=ywind, /free                     ;create window then plot
cont_vecmgm,b_mgm[ixmin:ixmax,iymin:iymax,0], b_mgm[ixmin:ixmax,iymin:iymax,1],$
    b_mgm[ixmin:ixmax,iymin:iymax,2], xaxis=x_mgm[ixmin:ixmax],yaxis=y_mgm[iymin:iymax],$
    shrink=0.15,drawbar=1, sizechar=sizechar,thickchar=thickchar,ctini=1,ctend=254,$
    nlevcont=40, neutral=0, skiptrbelow=50, skiplosbelow=50, units=units,colortb=0,$
    arrlen=3,barsim=500
```

Use the cont_vecmgm procedure to draw isocontours of B_z and arrows for the horizontal field

Try changing

- shrink / arrlen : change number /length of arrows
- neutral =0/1 do not / do draw PIL of B_z
- skiptrbelow do not draw arrow if the field is below a given threshold

Refer to routines' headers in ./scripts for additional explanations



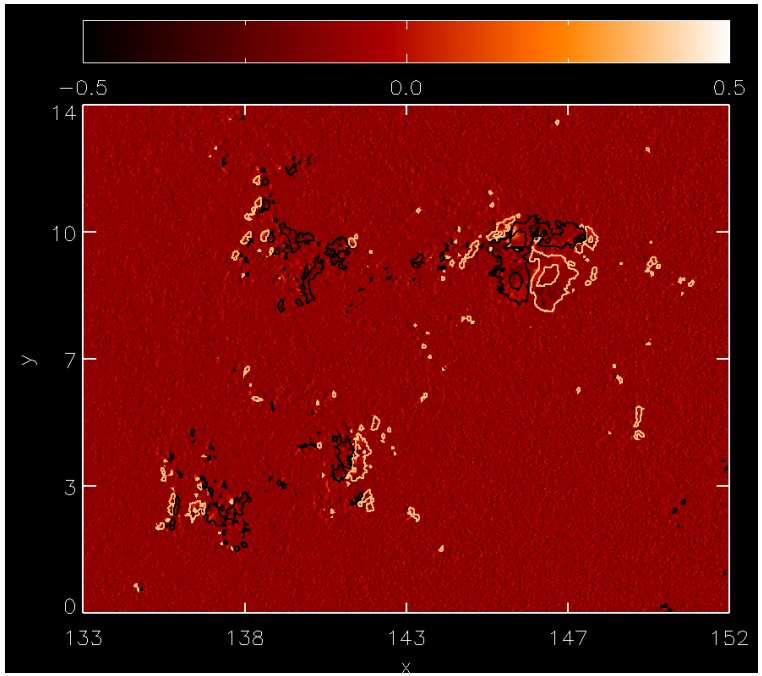
UCL MSSL

5. Vertical current density

```
jz_mgm=0.5*((shift(b_mgm[*,*,1],-1, 0)-shift(b_mgm[*,*,1],1, 0))/dx+$
            (shift(b_mgm[*,*,0], 0,-1)-shift(b_mgm[*,*,0],0,-1))/dy)
window,1, xsize=xwind,ysize=ywind,/free
cont_field_isoline,jz_mgm[ixmin:ixmax,iymin:iymax]/max(abs(jz_mgm)),$
                   b_mgm[ixmin:ixmax,iymin:iymax],$
   xaxis=x_mgm[ixmin:ixmax],yaxis=y_mgm[iymin:iymax],drawbar=1,$
   sizechar=sizechar,thickchar=thickchar,drawnl=0,$
   isonlevcont=5,leviso=[-800.,-200.,200.,800.],conthick=2,colortb=3,barmax=0.5,barmin=-0.5
```

- Define $J_z = \partial_x B_y - \partial_y B_x$
- Draw isocontours of J_z with $\pm$200G and $\pm$800G reference isolines of B_z
- cont_field_isoline has similar keywords as cont_vecmgm

Refer to routines' headers in ./scripts for additional explanations

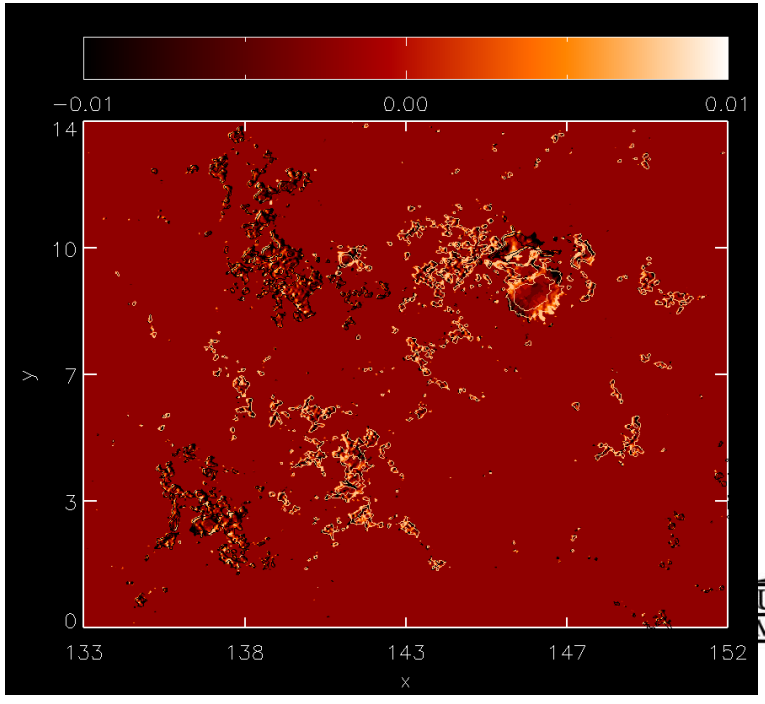


5. Force-free parameter

```
alpha=fltarr(nx,ny)
thresh=0.05                    ;Set a threshold on Bz to avoid zeroes
bz_mgm=b_mgm[*,*,2]
ind=where(abs(bz_mgm) ge thresh*max(abs(bz_mgm)),cnt)
alpha[ind]=jz_mgm[ind]/bz_mgm[ind]*(0.03/0.73)^2          ;Convert to approx 1/Mm
skip=4                                                    ;Remove edge values
alpha[0:skip-1,*]=0. & alpha[nx-skip-1:nx-1,*]=0
alpha[*,0:skip-1]=0. & alpha[*,ny-skip-1:ny-1]=0.
window,1,xsize=xwind,ysize=ywind, /free                      ;create window then plot
cont_field_isoline,alpha[ixmin:ixmax,iymin:iymax],bz_mgm[ixmin:ixmax,iymin:iymax],$
  xaxis=x_mgm[ixmin:ixmax],yaxis=y_mgm[iymin:iymax],drawbar=1,$
  sizechar=sizechar,thickchar=thickchar,nlevcont=10,drawnl=0,$
  isonlevcont=5,leviso=[-800.,-200.,200.,800.],conthick=1,colortb=3,barmax=0.01,barmin=-0.01
```

- Define $\alpha = J_z / B_z$ with a threshold on B_z to avoid small values
- Draw isocontours of α with $\pm$200G and $\pm$800G reference isolines of B_z
- Again cont_field_isoline

Refer to routines' headers in ./scripts for additional explanations



UCL MSSL

5. Forces on the mgm

Looking at Lorentz forces on the mgm, *e.g.*,

$$\mathcal{F}_x = -\frac{\int_{mgm} B_x B_z \; dxdy}{\frac{1}{2}\int_{mgm}(B_x^2 + B_x^2 + B_z^2)\; dxdy}$$

- epsfor takes mgm and axes in input
- fnorm[3] are normalized force components
- flux = half unsigned flux = $0.5 \int |B_z| \; dxdy$

```
epsfor, b_mgm, x_mgm, y_mgm, $
        ef,fnorm, flux=flux,/verbose
```

```
IDL>  epsfor, b_mgm, x_mgm, y_mgm, ef,fnorm,flux=flux,/verbose
------------------------------
   Magnetic flux                          1642.9957
   Relative flux imbalance               -0.18605007
   Force components:
           x:          -0.14295208
           y:          -0.16926409
           z:          -0.12644268
   Total force and  torque balances       0.28255076      0.20907142
   Laplacian  smoothness                 0.093366346
------------------------------
```

UCL MSSL

6 Time evolution

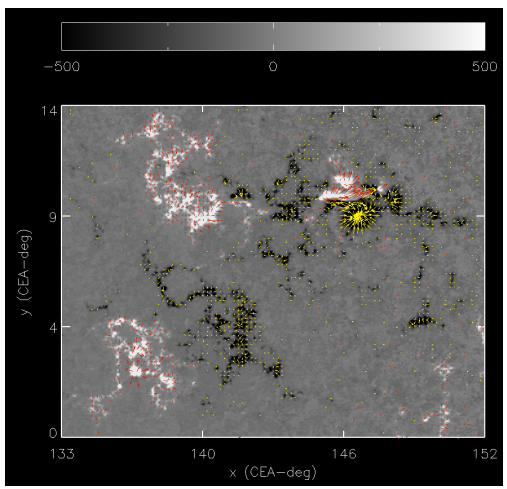
Let's repeat for the whole time series by combining previous routines

- load_fits load all fits in datadir and return a 4D array of [*nx*, *ny*, 3, *nframes*]
- frametime is some string label based on filenames for time stamp
- 1.5 days at 12 min cadence

```
datadir =maindir+ "data/fits/"
nn=long([ixmax-ixmin+1,iymax-iymin+1])
load_fits, datadir,ixmin,iymin,nn,x_mgm,y_mgm,$
     units,b_mgm,frametime
```

```
mvname="movie_vmgm"                ;Name your movie
outdir=maindir+"data/"             ;Where to save it
tmpdir=maindir+"data/tmp/"         ;tmp directory
vmgm2mp4, datadir,outdir,mvname,x_mgm,y_mgm,b_mgm,$
  tmpdir=tmpdir, keep_png=0,units=units,srk=0.1,$
  arrlen=2.5,pil=0,sizechar=3,ywind=900,$
  label=frametime
```

- Use vmgm2mp4 to produce the png of the frames
- If you are on linux -wise guy- and you have ffmpeg installed, you can try to build the movie (keep_png=0 and overwrite)
- If not, set keep_png=1 to save png images and merge them later



A copy is saved in ./data/movie_vmgm.mp4

UCL MSSL

6 Time evolution

Plot flux and forces in time

- Cycle over all snapshots
- Store forces in force_t[4,nframes] and flux in flux_t[nframes]

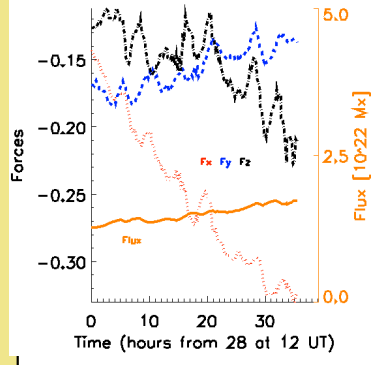
```
nframes=(size(b_mgm))[4]
time= make_array(nframes,/index)*0.20   ;Time [h] from start series
ef_t= fltarr(4,nframes)
force_t= fltarr(3,nframes)
flux_t = fltarr(nframes)
for iframe=0,nframes-1 do begin &$
 epsfor,b_mgm[ixmin:ixmax,iymin:iymax,*,iframe],$
    x_mgm[ixmin:ixmax],y_mgm[iymin:iymax],ef,force, flux=flux,verbose=0  &$
 ef_t[0:3,iframe]=ef[*] & force_t[*,iframe]=force[*] &flux_t[iframe]=flux &$
endfor
```

Panic button:

```
restore, maindir+ "data/flux_long.sav",/verbose
```

Plot the time evolution of the force components, and the of the total flux (use approximate conversion factor CEA-deg to cm)

```
cea2deg=(0.5*0.72*10^8./0.03)     ;Approx. conversion CEA-deg to cm
loadct,39 & black=0 & white=255 & orange=212 & blue=64 & red=228
lineplot=[1,2,3,0] & color=[red,blue,black,orange] & labels=["Fx","Fy","Fz","Flux"]
window,3, xsize=600,ysize=600,/free
pxmin=0 & pxmax=1.1*max(time) & pymin=min(force_t) & pymax=max(force_t)
plot,time,ef_t[0,*],/nodata, xtitle="Time (hours from 28 at 12 UT)",$
     ytitle="Forces",charsize=sizechar,charthick=linethick,$
     xrange=[pxmin,pxmax],yrange=[pymin,pymax],xstyle=9,ystyle=9,$
     color=black ,background=white,position=[0.22,0.15,0.80,0.90]
for i=0,2 do oplot, time, force_t[i,*], thick=4, linestyle=lineplot[i],color=color[i]
for i=0,2 do xyouts,0.50+i*0.05,0.50,/normal,labels[i],color=color[i],charthick=linethick
axis,yaxis=1,yrange=[0,5],charsize=sizechar,yticks=2,charthick=linethick,$
     color=color[3],ystyle=1,ytitle="Flux [10^22 Mx]",/save
oplot,time, flux_t[*]*cea2deg^2/10^22.,thick=4,linestyle=lineplot[3],color=color[3]
xyouts,0.3,0.3,/normal,labels[3],color=color[3],charthick=linethick
```



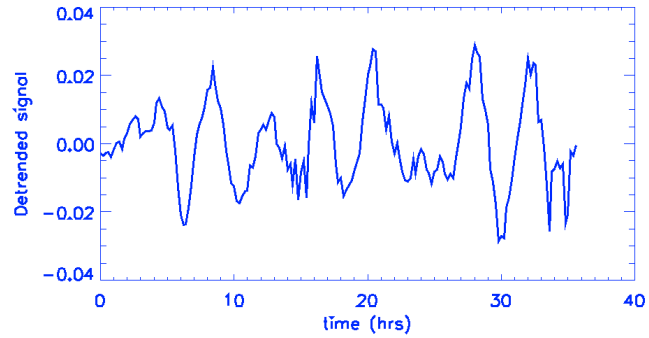
Force oscillations? Longitude-dependece in F_x?

6 Time evolution

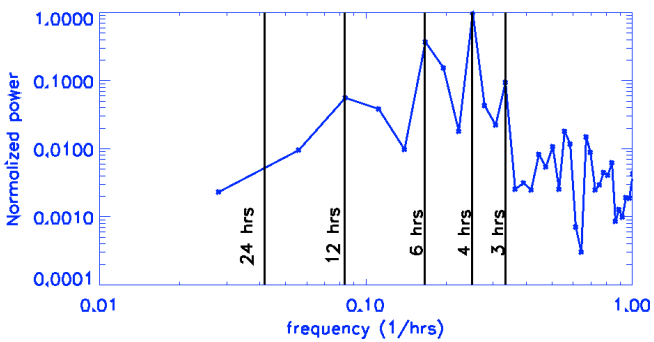
```
signal=reform(force_t[2,*])                                          ;Consider Fz
signal=signal-smooth(signal,30,/edge_truncate)                       ;Detrend removing long average
plot,time,signal,xtitle="time (hrs)",ytitle="Detrended signal",$
     xrange=[pxmin,pxmax],yrange=[-0.03,0.03], charsize=sizechar,$
     charthick=linethick, thick=linethick,color=blue ,background=white
```

Analysis of force signal.

- Remove long wavelengths using smooth with 30 hrs carbox




```
.r fourier_power_spectrum_1D
fourier_power_spectrum_1D, time,signal,halffreq,fpower
fday=1./24
plot, halffreq,fpower/max(fpower), /ylog,/xlog, xtitle="frequency (1/hrs)",$
      ytitle="Normalized power",xrange=[0.03,1],psym=-2,charsize=sizechar,$
     charthick=linethick, thick=linethick,color=blue ,background=white
pos=[1.,2.,4.,6.,8.]                                                 ;Plot power spectrum
lab=[24,12,6,4,3]
for i=0,4 do plots, pos[i]*[fday,fday],[0.0001,1],color=black,thick=linethick
for i=0,4 do xyouts,(pos[i]-0.1)*fday,0.00002,string(lab[i])+" hrs",orient=90,$
                    color=black,charthick=linethick,charsize=sizechar
```



Fourier analysis of force signal.

- fourier_power_spectrum_1D is a simple implementation of FFT
- Power on harmonics of the satellite's orbital period
- Try signal=reform(force_t[1,*]) or signal=flux_t

UCL MSSL

7. Potential field extrapolation

Compute the current-free magnetic field above the HMI vmgm

- Interpolate to reduce computational time
- New grid $80 \times 63 \times 50$
- New resolution 0.24 CEA-deg
- Warning: Flux is sensitive to interpolation

```
nn_lr=nn/8                                          ;1/8th resolution
nz_lr=50
x_lr=fltarr(nn_lr[0]) & y_lr=fltarr(nn_lr[1])  ;Interpolate axes
x_lr[*]=congrid(x_mgm,nn_lr[0])
y_lr[*]=congrid(y_mgm,nn_lr[1])
z_lr=make_array(nz_lr+1,/float,/index)*(x_lr[1]-x_lr[0])
b_lr=fltarr(nn_lr[0],nn_lr[1],3)                    ;and field
b_lr[*,*,0]=congrid(reform(b_mgm[*,*,0,150]),nn_lr[0],nn_lr[1],cubic=-0.5)
b_lr[*,*,1]=congrid(reform(b_mgm[*,*,1,150]),nn_lr[0],nn_lr[1],cubic=-0.5)
b_lr[*,*,2]=congrid(reform(b_mgm[*,*,2,150]),nn_lr[0],nn_lr[1],cubic=-0.5)
```

- Alissandrakis method with $\alpha = 0 \Longrightarrow$ same routine for potential and linear
- Routine by M. Georgoulis: lff_extrap.pro
- If not otherwise specified
  - Domain is padded with zeros to avoid aliasing
  - enlarge x3 $\Longrightarrow$ reduce $\alpha_{\max}$
- Takes in input
  - the vertical field at the photosphere (B_lr[*,*,2])
  - the value of the force-free parameter (α)
  - the required height in gp (nz_lr)
- Output: B_pot_lr[3,nn_lr[0],nn_lr[1],nn_lr], note the order of dimensions

```
.r lff_extrap
LFF_EXTRAP,B_lr[*,*,2],B_pot_lr,alpha=0.,z=nz_lr,/normal
```

UCL MSSL

7. Visualize field lines

- Specify starting points (seeds)
- Follow the the fl starting at the seed

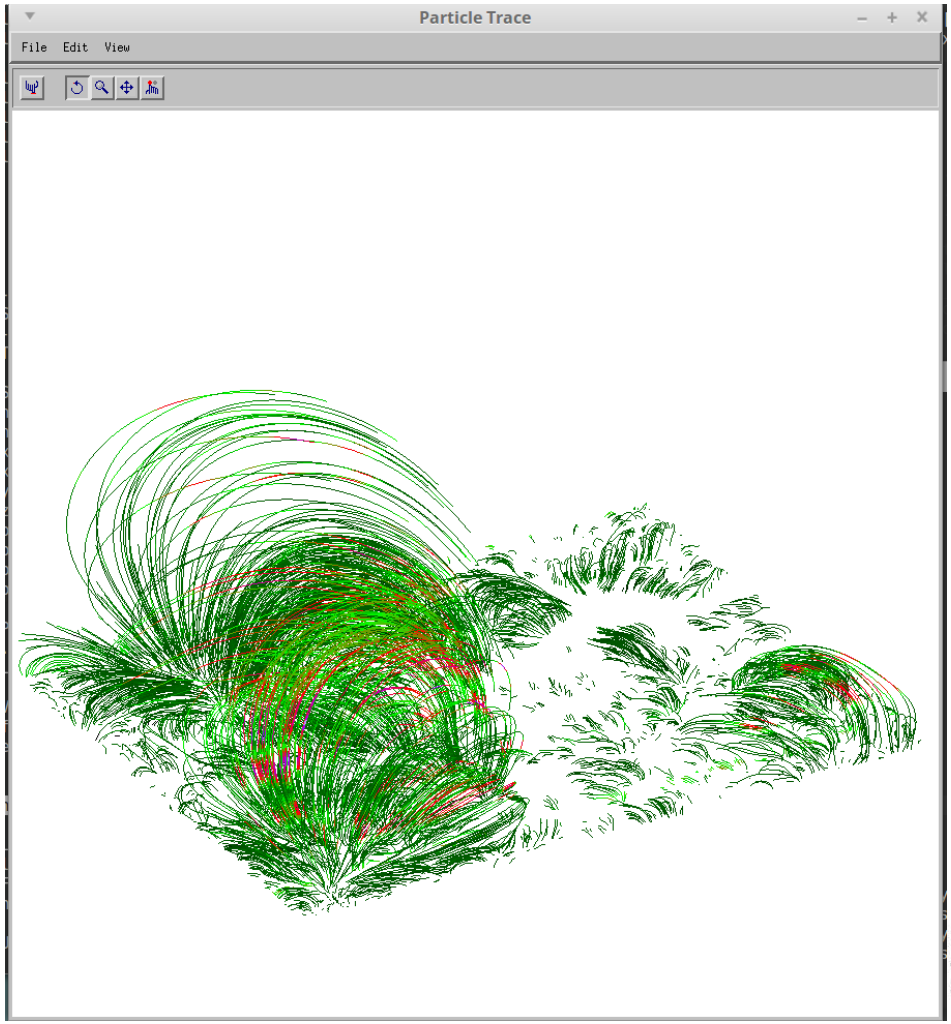
```
nfl=10000                              ;number of seeds
seeds=transpose([[(RANDOMU(S,nfl))*nn_lr[0]],$
                 [(RANDOMU(S,nfl))*nn_lr[1]],$
                  [replicate(0.,nfl)]])
show_stream,b_pot_lr,seeds=seeds,/lines
```

IDL has no dedicated field-line tracing tool

SHOW_STREAM is largely inadequate, but it can be used as a starting point to develop such a tool

Panic button:

```
restore, maindir+"data/extrapolations.sav",/v
```



UCL MSSL

7. Linear field extrapolation

Similarly, compute the linear field using the same routine by

- change α
- change output array name

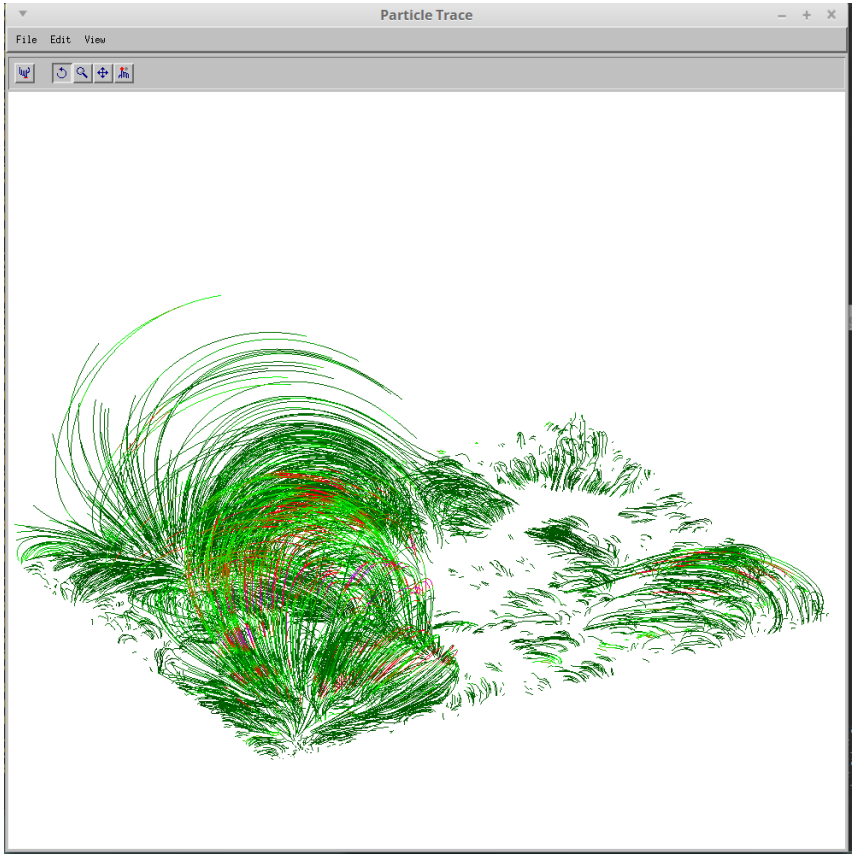
```
alpha=0.016
LFF_EXTRAP,B_lr[*,*,2],B_lin_lr,alpha=alpha,z=nz_lr,/normal
```

```
show_stream,b_lin_lr,seeds=seeds,/lines
```

Visualize using the same set of seeds

SHOW_STREAM is largely inadequate, but it can be used as a starting point to develop such a tool

... or ...

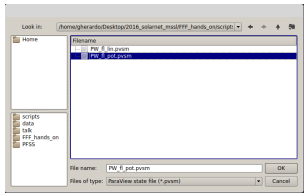


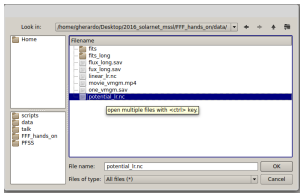
UCL MSSL

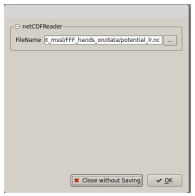
Dedicated visualization tools

- Open, free, server-client structure for use on cluster, in VTK (Paraview/Visit) or Python (Mayave) (here Paraview)
- Before leaving IDL, save potential andl linear fields in a format recognizable by Paraview (here netCDF4)

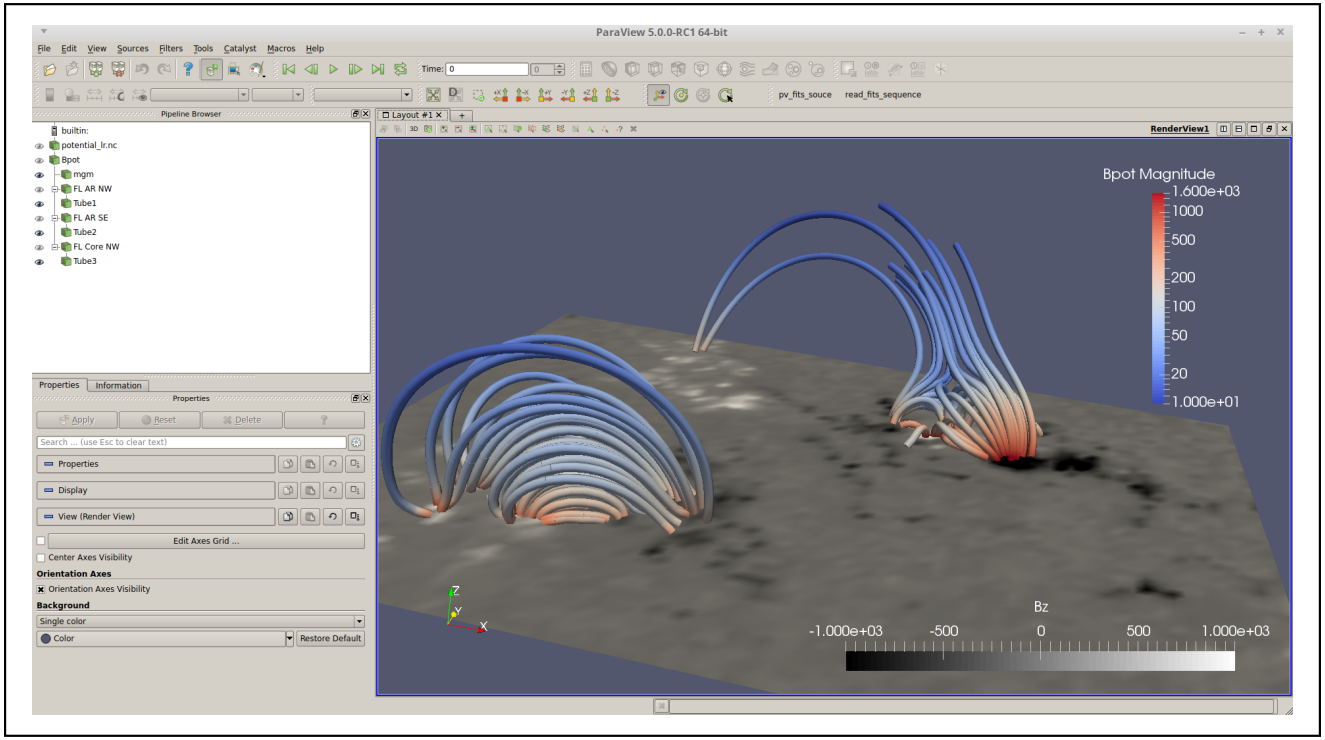
```
flag=IDL_netCDF4_4D(B_pot_lr,xax_in=x_lr,yax_in=y_lr,zax_in=z_lr,outfile=maindir+"data/potential_lr.nc")
flag=IDL_netCDF4_4D(B_lin_lr,xax_in=x_lr,yax_in=y_lr,zax_in=z_lr,outfile=maindir+"data/linear_lr.nc")
```







1. Launch Paraview
2. Open File → Load State
3. Select FFF_hands_on/script/PW_fl_pot.pvsm
4. Select the associated datafile FFF_hands_on/data/potential_lr.nc



UCL MSSL

Dedicated visualization tools

Both potential and linear can be loaded at the same time

1. Open File → Load State
2. Select FFF_hands_on/script/PW_fl_lin.pvsm
3. Select the associated datafile FFF_hands_on/data/linear_lr.nc

Paraview concept

- Left top panel lists filters applied to data
- Left bottom panel set properties of selected filter
- Use  to select visible items