Nordita 2015, School Data Analysis, Lecture 2, Single s/c boundary methods

Table of Contents

Change to temporary working directory

Substitute by your working directory

```
cd(tempdir)
mkdir Nordita
cd    Nordita
```

```
Warning: Directory already exists.
```

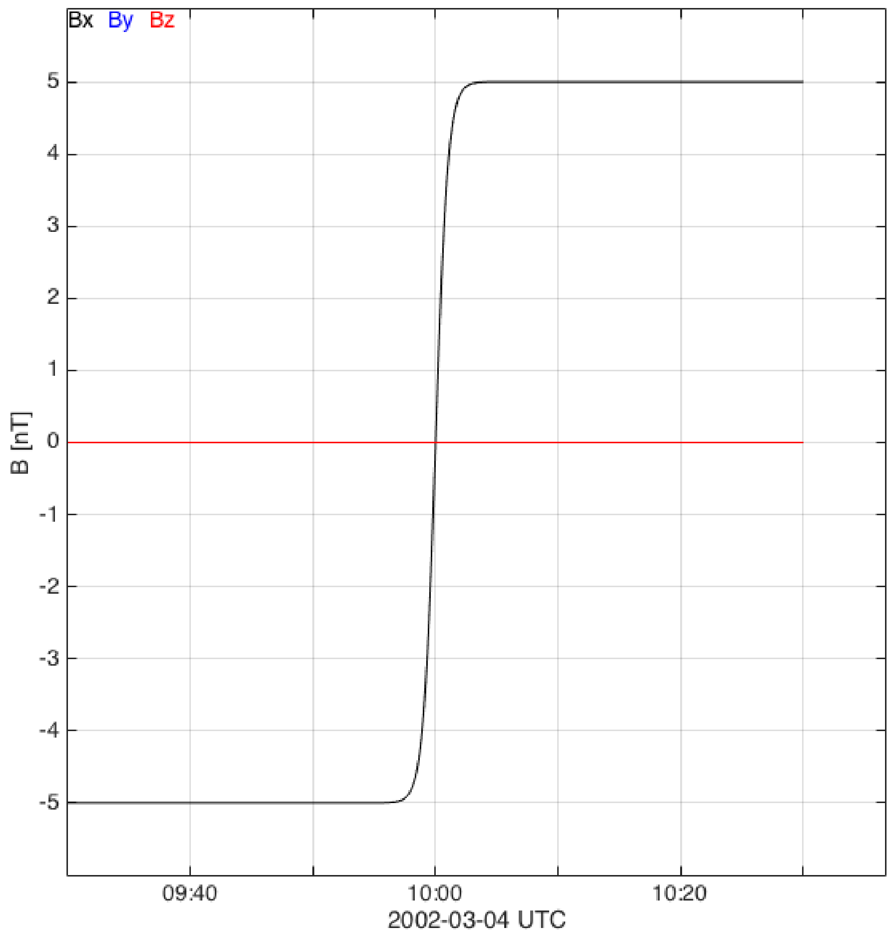
Harris like current sheet crossing

Lets generate 1sample each 5s time series during 1h after 2002-03-04 09:30 UTC, including artificial boundary in the middle of the interval.

The current sheet is such that only the Bx component changes

```
T   = EpochTT('2002-03-04T09:30:00Z'):5 ...
     :EpochTT('2002-03-04T10:30:00Z');% define time line as EpochTT
 object
t   = T - T.start;                    % define relative time in s from
 start
t   = t - mean(t);                    % time zero in the middle of
 interval
```

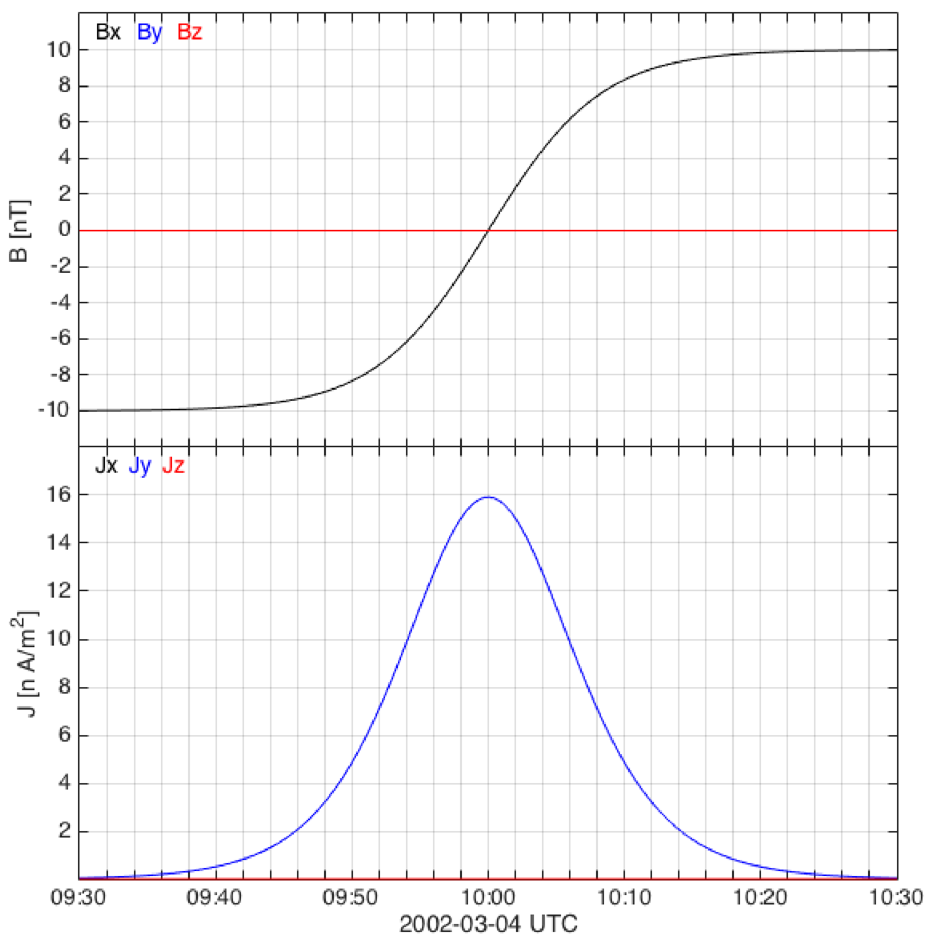
```
Bx   =  5*tanh(t/60);                    % define Bx +-5nT jump and 1min
 width
By   = t*0;                              % define Bx to be zero
Bz   = t*0;                              % define Bz to be zero

B = irf.ts_vec_xyz(T,[Bx By Bz]);        % define TSeries object (vector)
B.units = 'nT';                          % units
B.userData.LABLAXIS = 'B';               % plot label axis

h = irf_plot(1,'newfigure');          % initialize figure with one panel
irf_plot(h,B);                     % plot times series
irf_legend(h,{'Bx','By','Bz'},[0,1]);
```



Harris current sheet, B and J

```
mu0 = 4*pi/1e7;

Ljy = 500e3;                             % 500km, half width of current
 sheet
B0   = 10;                               % asymptotic magnetic field [nT]
```

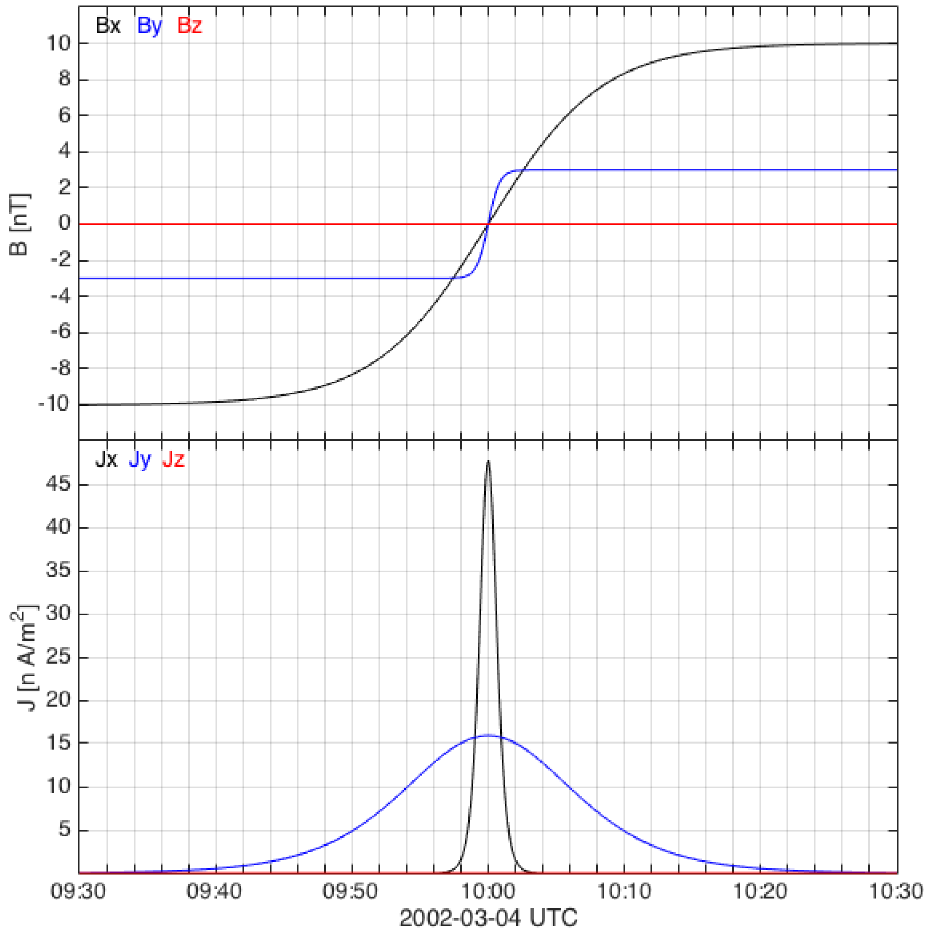
```
vz   = 1e3;                           % crossing current sheet at vz =
 1km/s

Bx   = B0*tanh(t*vz/Ljy);             % define Bx jump +- B0nT and
 width Ljy
By   = t*0;                           % define By as zero
Bz   = t*0;                           % define Bz as zero

Jx   = t*0;                           % zero current in X
Jy   = B0/Ljy*sech(t*vz/Ljy).^2/mu0;  % Harris current sheet in Y
Jz   = t*0;                           % zero current in Z directoin

B    = irf.ts_vec_xyz(T,[Bx By Bz]);  % define B as TSeries
J    = irf.ts_vec_xyz(T,[Jx Jy Jz]);  % define J as TSeries

h = irf_plot({B,J});
ylabel(h(1),'B [nT]')
ylabel(h(2),'J [n A/m^2]','interpreter','tex')
irf_legend(h(1),{'Bx','By','Bz'},[0.02,0.98]);
irf_legend(h(2),{'Jx','Jy','Jz'},[0.02,0.98]);
```



Double Harris current sheet

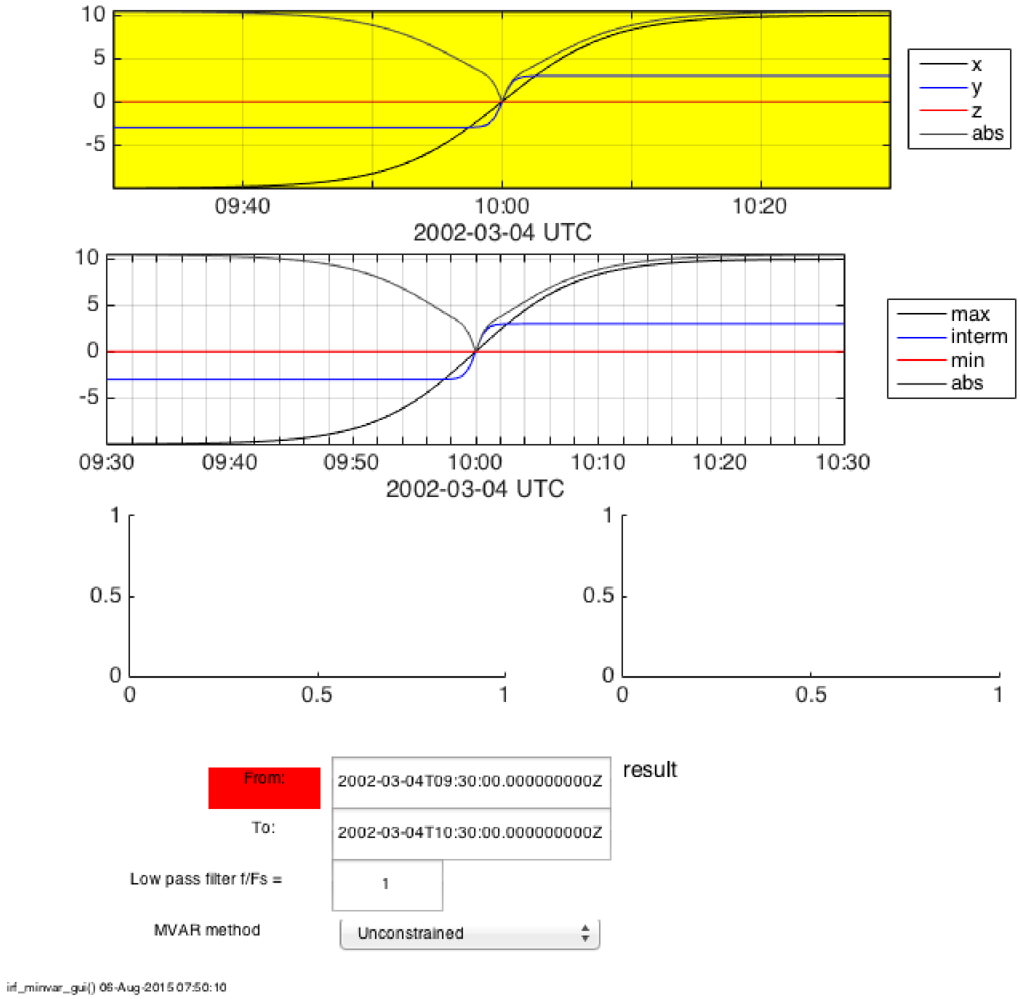
B changes in two components, where the current sheet thickness in each of the components is different. Thus we can construct one thick current sheet and on thin perpendicular to it. This can mimic a situation in space where ion current sheet is thick in one direction and electron current sheet is thin in a perpendicular direction.

```
Ljy = 500e3;                        % 500km, half width of jy
 current sheet
Ljx = 50e3;                         % 50km,  half width of jx
 current sheet
B0x = 10;                           % asymptotic Bx magnetic field
 [nT]
B0y = 3;                            % asymptotic By magnetic field
 [nT]
vz = 1e3;                           % crossing current sheet at vz =
 1km/s

Bx  = B0x*tanh(t*vz/Ljy);           % define Bx jump
By  = B0y*tanh(t*vz/Ljx);           % define By jump
Bz  = t*0;                          % define Bz as zero

Jx  = B0y/Ljx*sech(t*vz/Ljx).^2/mu0;  % Current sheet jx
Jy  = B0x/Ljy*sech(t*vz/Ljy).^2/mu0;  % Current sheet jy
Jz  = t*0;                          % zero current in Z directoin

B   = irf.ts_vec_xyz(T,[Bx By Bz]);   % define B as TSeries
J   = irf.ts_vec_xyz(T,[Jx Jy Jz]);   % define J as TSeries

h = irf_plot({B,J});
ylabel(h(1),'B [nT]')
ylabel(h(2),'J [n A/m^2]','interpreter','tex')
irf_legend(h(1),{'Bx','By','Bz'},[0.02,0.98]);
irf_legend(h(2),{'Jx','Jy','Jz'},[0.02,0.98]);
```



Minimum variance analysis (MVA)

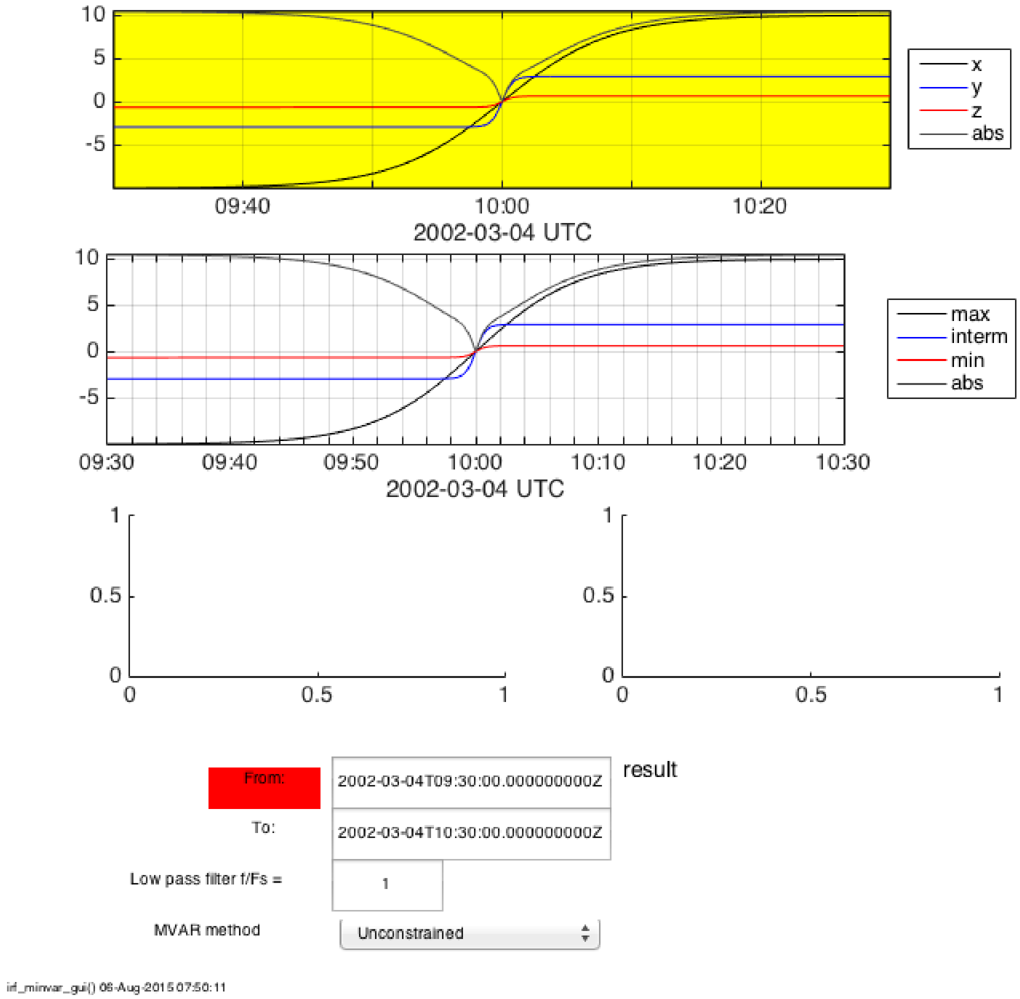
```
irf_minvar_gui(B);      % run MVA on B time series
```



MVA on B in a different reference frame

This is to illustrate that in MVA reference frame time series look the same independent of the original reference frame of data.

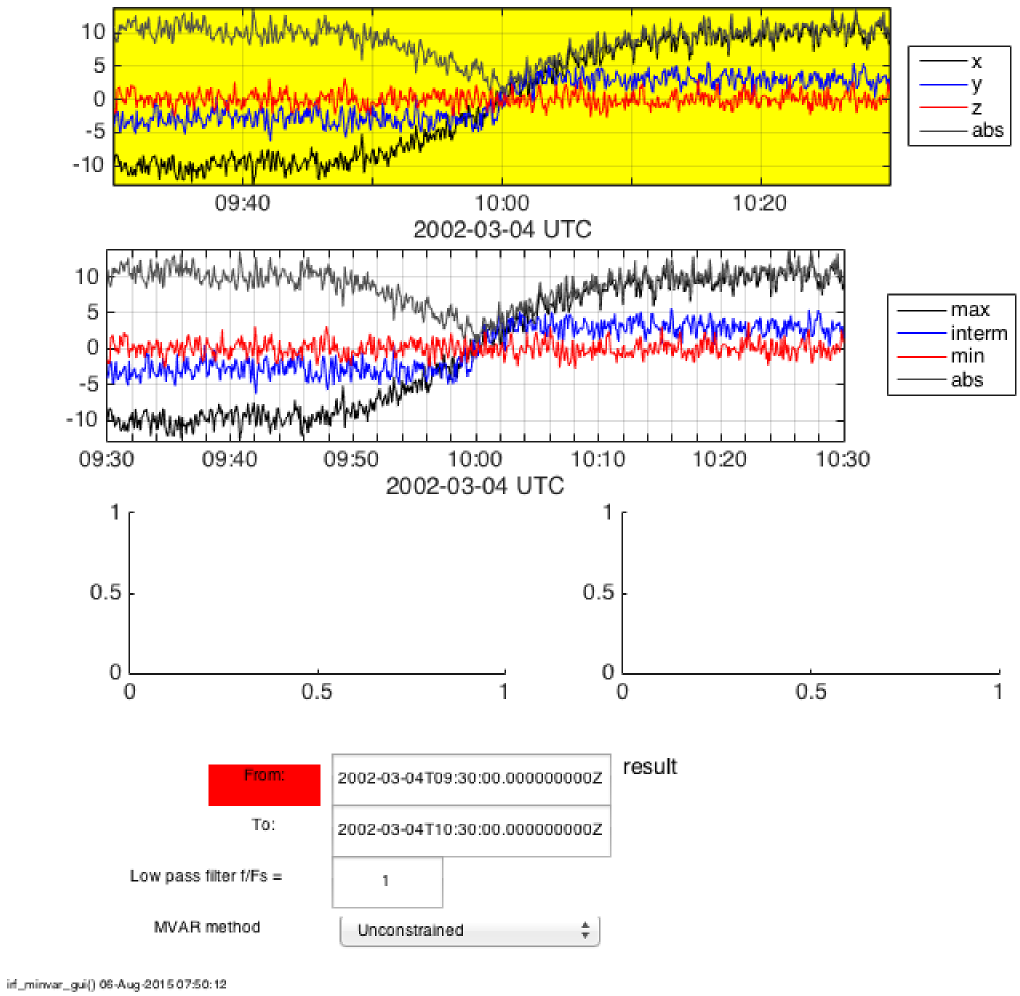
```
Bgsm = irf_gse2gsm(B);
irf_minvar_gui(Bgsm);   % run MVA on B time series in a different
 reference frame
```



Double Harris current sheet, B change in two components + noise

Let's add some random noise to see how MVA behaves on noisy data.

```
B0noise = 3;                                      % noise amplitude
noisyTimeSeries = model.synthetic_time_series(...
 'fs',1,'f',0,'peakHalfWidth',0.3,...
 'timeInterval',1e4,'components',3);
Bnoisy = B + B0noise*noisyTimeSeries(1:B.length,2:4);% add random
 noise of amplitude 1nT
Bnoisy.units = 'nT';                              % specify units
Bnoisy.userData.LABLAXIS = 'B';

irf_minvar_gui(Bnoisy);                           % run minimum variance analysis
 on time series
```



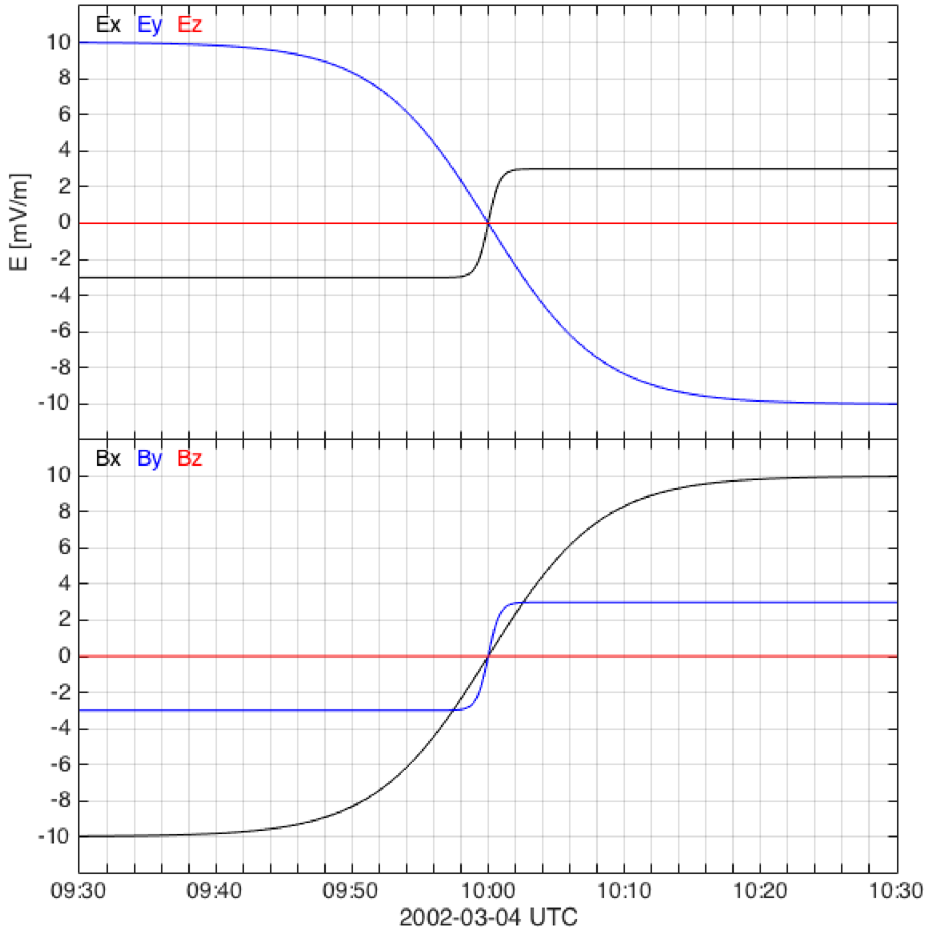
De Hoffmann - Teller frame

De Hoffmann - Teller velocity VHT defines a frame in which electric field E is minimized. In the case of 1D boundary VHT component along the boundary normal gives the boundary speed.

```
vSpacecraft = [0 0 vz];                % s/c moves in z with vz [m/s]
E           = irf_e_vxb(vSpacecraft,B); % E=-vxB
VHT         = irf_vht(E,B);            % returns value of VHT

h = irf_plot({E,B});
irf_legend(h(1),{'Ex','Ey','Ez'},[0.02,0.98]);
irf_legend(h(2),{'Bx','By','Bz'},[0.02,0.98]);
```

```
De Hoffmann-Teller frame is calculated using all 3 components of
 E=(Ex,Ey,Ez)
V_{HT}=1e+03 [ 0.00  0.00  1.00] =[0.00    0.00 1000.00] km/s
slope=1  offs=1.1e-15
cc=1
De Hoffmann-Teller frame is calculated using all 3 components of
 E=(Ex,Ey,Ez)
```

```
\delta V_{HT}=0 [ NaN    NaN    NaN] =[0.00   0.00   0.00] km/s
```



Harris current sheet based on vector potential (extra material)

When describing reconnection in 2D it is very convenient to use vector potential to show the magnetic structure of field lines. Magnetic field lines in (X,Z) plane are defined by A_Y component. Contour lines of A_Y show the topology of the field. The distance between such contour lines is inversely proportional to the magnetic field strength.

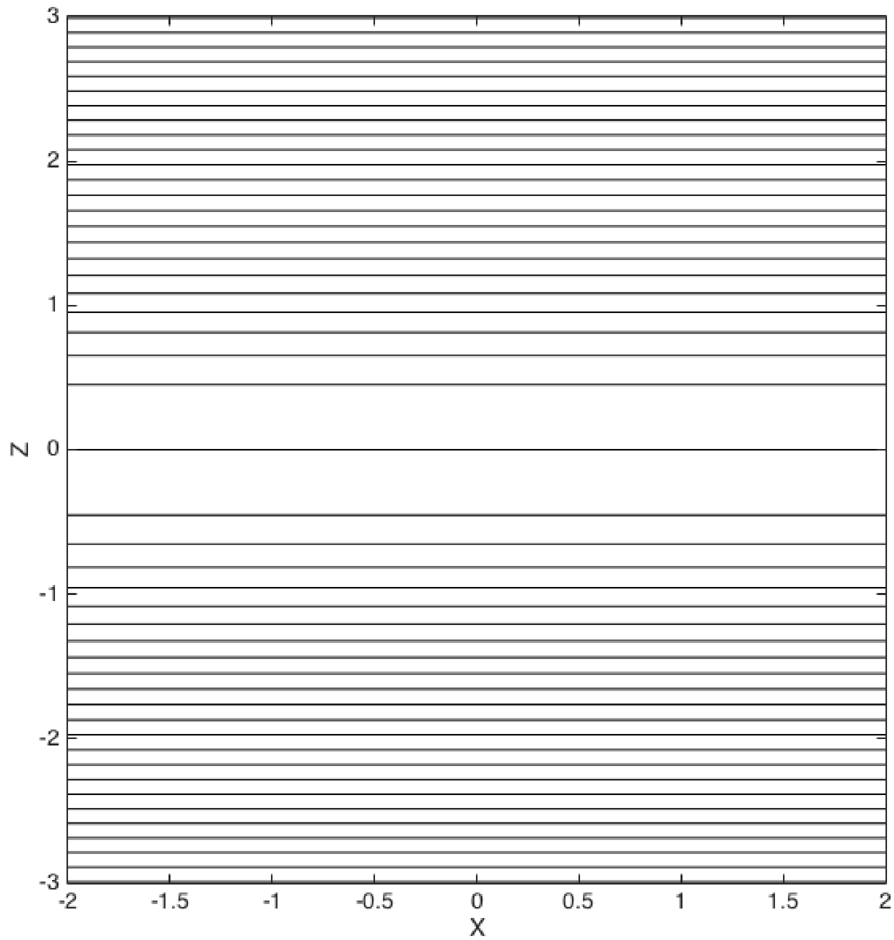
Let's define functions describing for Harris sheet the strength of B_X as a function of Z (the distance from the current sheet) and corresponding A_Y.

```
Bo = @(z,l) tanh(z./l);           % Bx, l - thickness
Ao = @(z,l) -l.*log(cosh(z./l)); % Ay
Acontours = [0:-.1:-3];
```

2D Harris current sheet (extra material)

Plotting undisturbed 2D Harris current sheet.

```
[X,Z] = meshgrid(-2:.1:2,-3:.1:3);
irf_plot(1,'newfigure');
contour(X,Z,Ao(Z,1),Acontours,'k')
ylabel('Z'); xlabel('X');
```

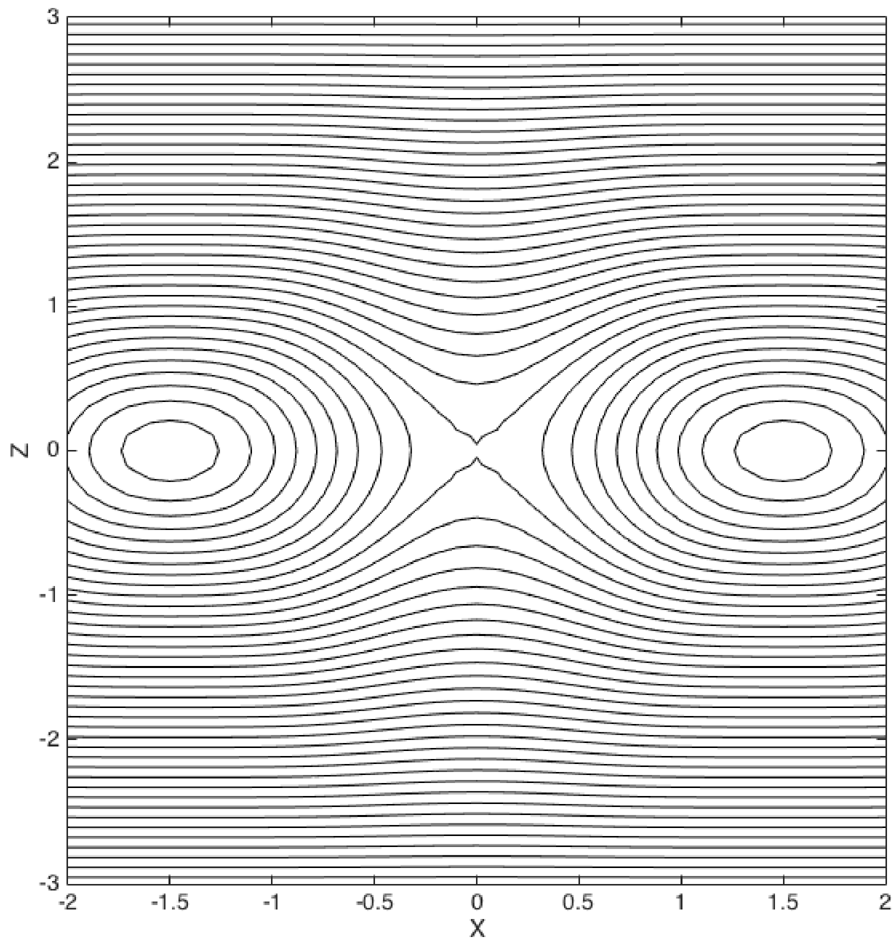


2D Harris current sheet with magnetic islands (extra material)

Construct A_Y such that one obtains magnetic islands inside the current sheet. In practice it is achieved by varying the current thickness as a function of X, and putting A_Y values to be constant for all X values at some large distance from the current sheet (large Z).

```
Zref                = 3;                          % the distance
 at which A_Y is put constant for all X
```

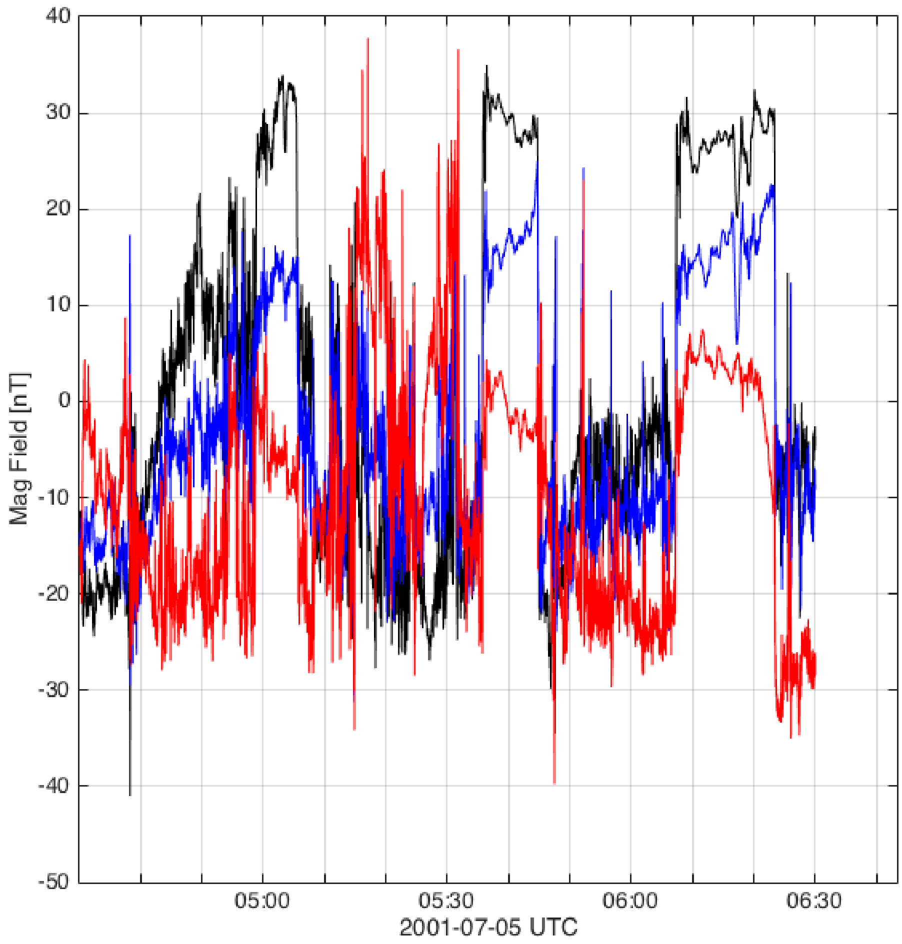
```
Acontours            = (-1:.03:1)*Ao(Zref,1);         % defined the
 levels of Acontours
[X,Z]                = meshgrid(-2:.1:2,-Zref:.1:Zref);
thicknessVariation  = .5;                             % the amplitude
 of thickness variation
variationWavelength = 3;                              % the wave
 length of thickness variation
thick                = @(x) 1 + thicknessVariation ... % thickness as
 function of X
                       *cos(x*2*pi/variationWavelength);
refAddition          = Ao(Zref,1)-Ao(Zref,thick(X));   % addition
 required at each X to make A(Zref,X) constant
A                    = Ao(Z,thick(X))+refAddition;

irf_plot(1,'newfigure');
contour(X,Z,A,Acontours,'k')
ylabel('Z'); xlabel('X');
```

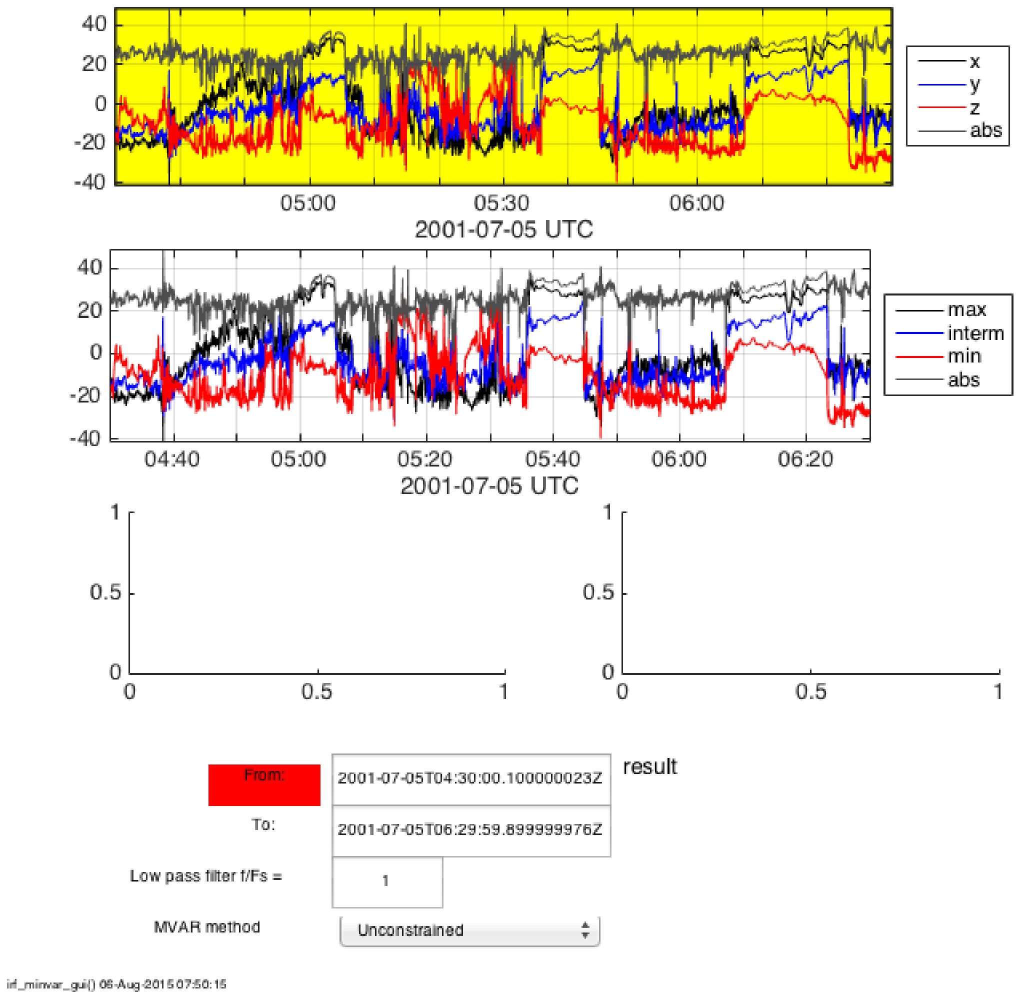


Magnetopause crossings in data

As real data we use event from (Paschmann et al., 2005 AnGeo) http://www.cluster.rl.ac.uk/csdsweb-cgi/csdsweb_pick?P_TYPE=P1&YEAR=2001&MONTH=Jan&DAY=26&SUB_PLOT=S02

```
cd /Users/andris/Dropbox/Projects/Nordita2015/Data/
CAA_20010705_0430_20010705_0630

% Tint = irf.tint('2001-01-26T10:30:00Z/2001-01-26T11:00:00Z');
% Tint = irf.tint('2001-07-05T04:30:00Z/2001-07-05T06:30:00Z');
% Tint = irf.tint('2001-09-15T05:00:00Z/2001-09-15T05:15:00Z');

if 0,
 caa_download(Tint,'C1_CP_FGM_SPIN');
 caa_download(Tint,'C?_CP_FGM_5VPS');
 caa_download(Tint,'C?_CP_FGM_FULL');
 caa_download(Tint,'C1_CP_EFW_L2_E3D_GSE');
 caa_download(Tint,'C1_CP_EFW_L3_E3D_GSE');
 caa_download(Tint,'C?_CP_CIS_HIA_ONBOARD_MOMENTS');
 caa_download(Tint,'C1_CP_CIS_HIA_HS_1D_PEF');
 caa_download(Tint,'C1_CP_RAP_ESPCT6');
 caa_download(Tint,'C1_CP_PEA_PITCH_SPIN_DEFlux');
end

caa_load C1_CP_FGM_SPIN
B1 = irf_get_data('B_vec_xyz_gse__C1_CP_FGM_5VPS','caa','ts');
irf_plot(1,'newfigure');
irf_plot(B1);
irf_minvar_gui(B1)
```


Mag Field [nT]
40
30
20
10
0
-10
-20
-30
-40
-50
05:00
05:30
06:00
06:30
2001-07-05 UTC



Find De Hofmann - Teller frame for data

```
B1 = irf_get_data('B_vec_xyz_gse__C1_CP_FGM_SPIN','caa','ts');
V1 =
 c_caa_var_get('velocity_gse__C1_CP_CIS_HIA_ONBOARD_MOMENTS','caa','ts');
E1 = c_caa_var_get('E_Vec_xyz_GSE__C1_CP_EFW_L3_E3D_GSE','caa','ts');
E1vxb = irf_e_vxb(V1,B1);
V1exb = irf_e_vxb(E1,B1,-1);

VHT = irf_vht(E1,B1);

De Hoffmann-Teller frame is calculated using all 3 components of
 E=(Ex,Ey,Ez)
V_{HT}=245 [ -0.88 -0.47  0.05] =[-216.63 -114.34   12.51] km/s
slope=1.04  offs=0.0098
cc=0.972
De Hoffmann-Teller frame is calculated using all 3 components of
 E=(Ex,Ey,Ez)
\delta V_{HT}=2.3 [ 0.65  0.54  0.54] =[1.49  1.24  1.23] km/s
```

Published with MATLAB® R2015a