

SPEDAS training session (06 Mar. 2019)
Advanced course [2 of 2]

Arase/ MGF & PWE-EFD ELF wave analysis

S. Matsuda, S. Kurita, M. Teramoto, M. Shoji, Y. Miyoshi, T.
Hori, T.-F. Chang, Y. Tsugawa, N. Umemura, T. Segawa
(ISAS/JAXA & ERG Science Center, ISEE, Nagoya Univ.)

これまでの講習会資料を下記にアップロードしています。ご参考ください。

https://ergsc.isee.nagoya-u.ac.jp/documents/others/data/SPEDAS_training_session/index.shtml

PWE/EFD, MGF Dynamic Spectrum Quick Look

https://ergsc.isee.nagoya-u.ac.jp/cef/erg_emic_2hours.cgi

ERG EMIC QL for 2017-2019

Submit -1 day Previous 2017 10 24 DOY=297 1000-1200 UT Size=150 Next +1 day Submit

CAUTION!!

This plot can be used for quick look only. Any publication is strictly prohibited. If you consider to use plots for presentation at any meetings, please contact with MGF and PWE PIs before the presentation.

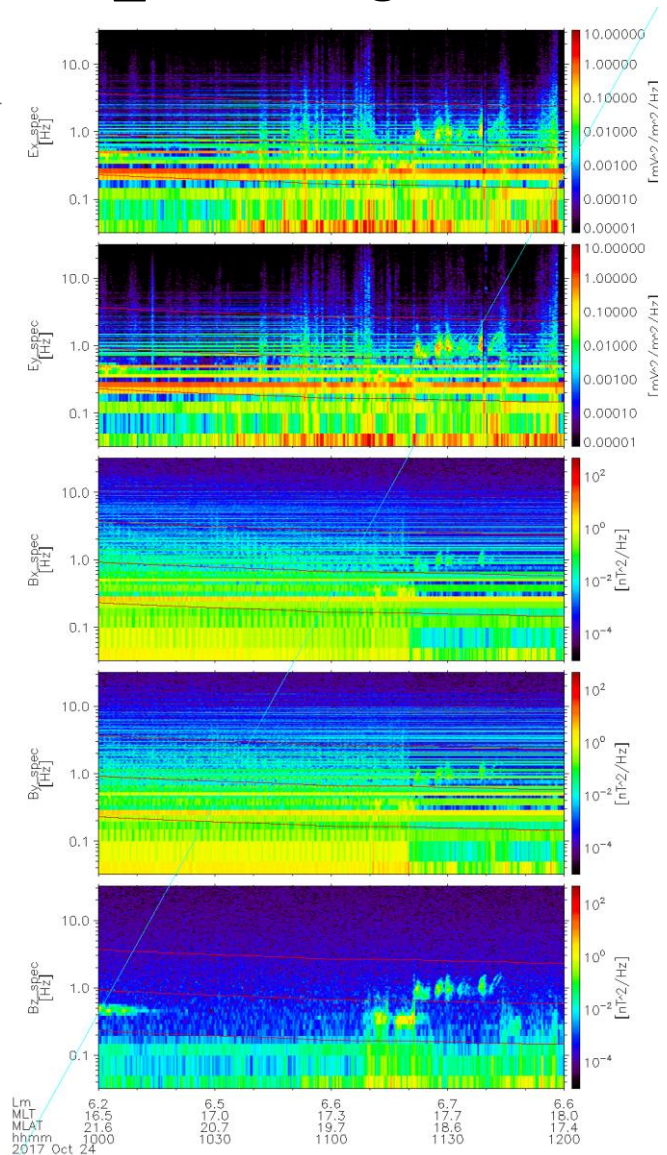
このプロットは、衛星スピン周期よりも速い磁場変動を伴う現象(EMICなど)を確認することを主な用途とするもので、あくまで Quick Lookプロットとしてご利用いただくことを想定しています。

本QLに関してのお問い合わせは各infoアドレスにお願いいたします。

コンタクト:

erg_pwe_info [at] isee.nagoya-u.ac.jp

erg_mgf_info [at] isee.nagoya-u.ac.jp



Geophysical Research Letters

RESEARCH LETTER

10.1029/2018GL079765

Instantaneous Frequency Analysis on Nonlinear EMIC Emissions: Arase Observation

Special Section:

Initial results of the ERG (Arase) project and multi-point observations in geospace

Masafumi Shoji¹, Yoshizumi Miyoshi¹, Yoshiharu Omura², Lynn M. Kistler^{1,3}, Yasumasa Kasaba⁴, Shoya Matsuda⁵, Yoshiya Kasahara⁶, Ayako Matsuoka⁵, Reiko Nomura⁷, Keigo Ishisaka⁸, Atsushi Kumamoto⁴, Fuminori Tsuchiya⁴, Satoshi Yagitani⁶, Mariko Teramoto¹, Kazushi Asamura⁵, Takeshi Takashima⁵, and Iku Shinohara⁵

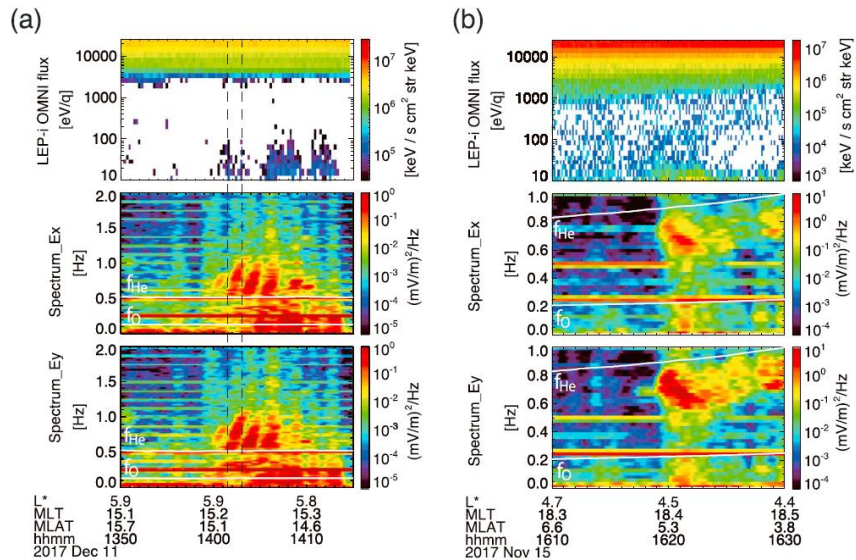


Figure 1. Energy-time diagram of ions observed by Arase LEP-i and dynamic spectrum of electromagnetic ion cyclotron waves observed by the Arase Plasma Wave Experiment-electric field detector in (a) 11 December 2017 and (b) 15 November 2017. The top panel shows the E - t diagram of ions between 10 eV to 25.5 keV. The middle and bottom panels show the dynamic spectra of the electric field E_x and E_y , respectively. The white lines in the panels show the cyclotron frequencies of He^+ (f_{He^+} , upper) and O^+ (f_{O^+} , lower). The dashed lines in panel (a) indicate the time period for the analysis in Figure 2. LEP-i = low-energy particle experiments-ion mass analyzer.

Geophysical Research Letters

RESEARCH LETTER

10.1029/2018GL080109

Spatial Distribution of Fine-Structured and Unstructured EMIC Waves Observed by the Arase Satellite

Special Section:

Initial results of the ERG (Arase) project and multi-point observations in geospace

S. Matsuda¹, Y. Kasahara², Y. Miyoshi³, R. Nomura⁴, M. Shoji⁵, A. Matsuoka⁶, Y. Kasaba⁷, S. Kurita⁸, M. Teramoto³, and K. Ishisaka⁹

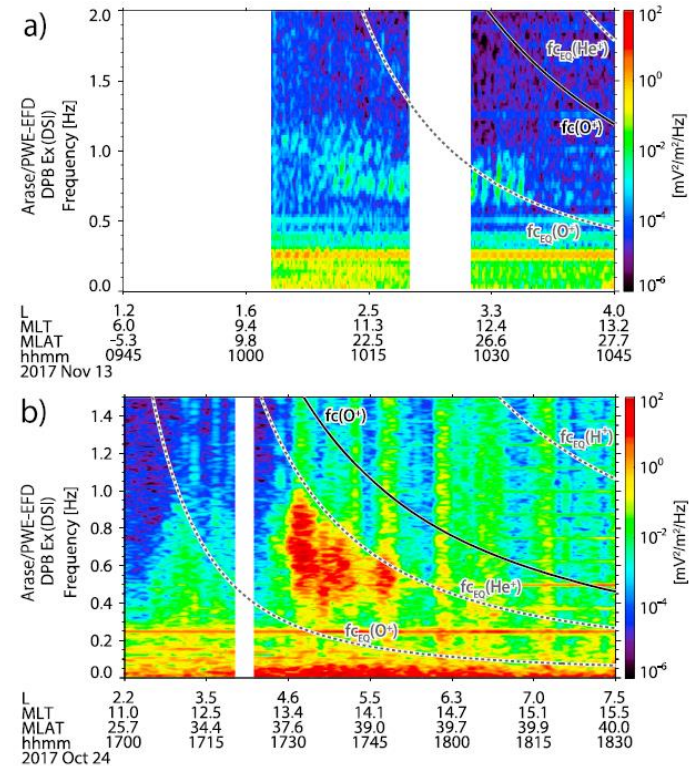


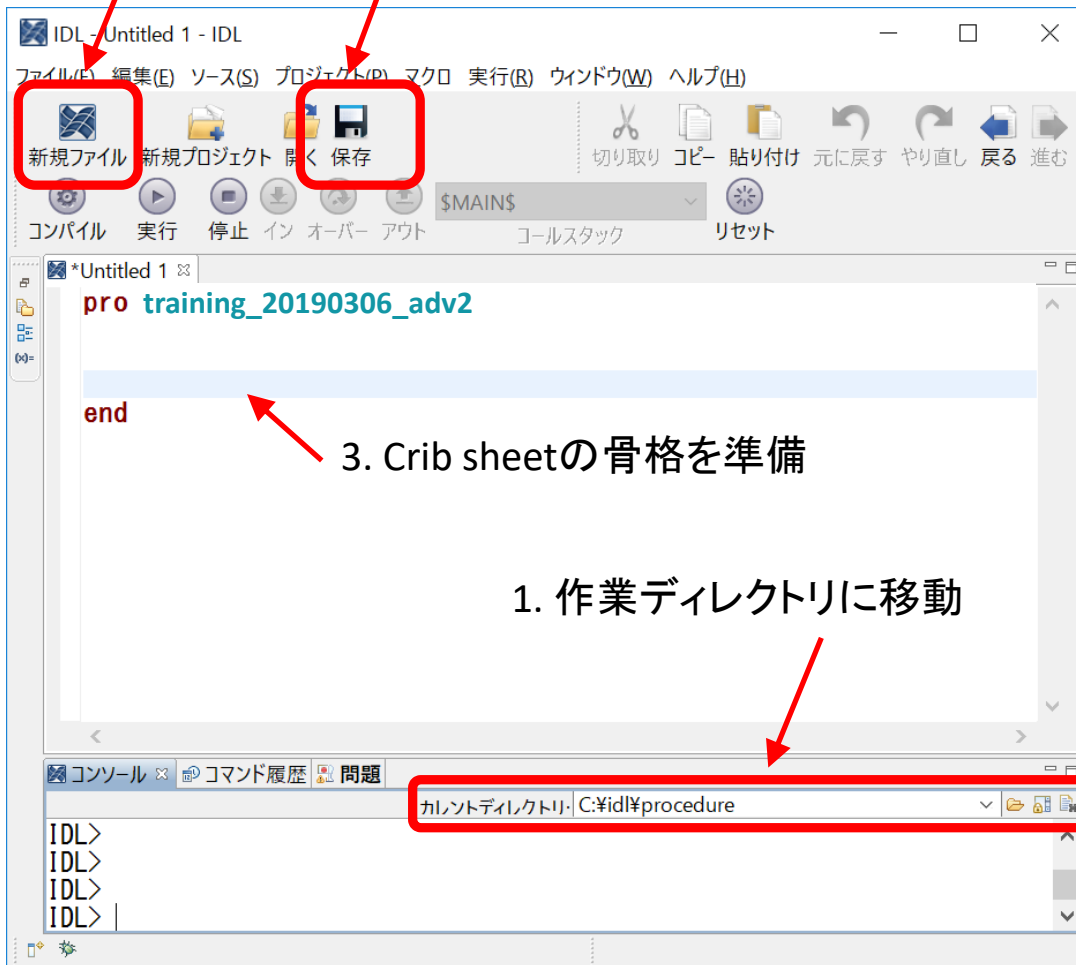
Figure 1. (a) Dynamic power spectrum of the electric field of a fine-structured EMIC wave observed by Arase PWE/EFD on 13 November 2017. (b) Dynamic power spectrum of the electric field of unstructured EMIC waves observed by Arase PWE/EFD on 24 October 2017. EMIC = electromagnetic ion cyclotron; PWE = Plasma Wave Experiment; EFD = Electric Field Detector.

0. crib sheetを作成する準備

0.1 コンソールからの入力では、複雑な手順を記述することが困難です。
Crib sheet(手順書・スクリプト)を作成しましょう。

2. 新規ファイルを作成

4. プロシージャ名と同じ名前で保存



0.2

以下のコマンドで、crib sheetの途中で「一時停止」「再開」ができます

(Crib sheet)

```
; *****
; test
; *****
print, 'test1'

;stopコマンドで一時停止
stop

print, 'test2'
```

(Console)

```
test1
% Stop encountered: *****
> .cont
test2
```

1. 必要なファイルの読み込み

1.1 プロットの初期設定と, 解析対象の日付を指定します

```

; *****
; initialize
; *****
erg_init


uname = '*****'
pass = '*****'

; *****
; set time span
; *****
timespan, '2017-12-11T13:00:00', 2, /hour

```

プロジェクトメンバー限定でファイルを公開しています。
ユーザー名/パスワードの取り扱いには十分ご注意ください!!!

1.2 軌道L2 CDFファイルを取得して, プロット下部に表示するように設定します

0.1 

Lm	4.8	4.5	6.2	1.9	6.1	3.6	5.8
MLT	5.5	1.6	4.5	19.4	3.7	6.6	2.6
MLAT	2.0	-14.8	10.2	-32.8	-9.9	13.7	-17.8
hhmm	0000	0400	0800	1200	1600	2000	0000
2017	Mar 28						Mar 29

```

; *****
; load orbit CDF & set var label
; *****
set_erg_var_label

```

1. 必要なファイルの読み込み

1.3 MGF 64 Hzデータ(DSI座標系) のCDFを取得します

```
; *****
; load MGF 64 Hz CDF
; *****
```

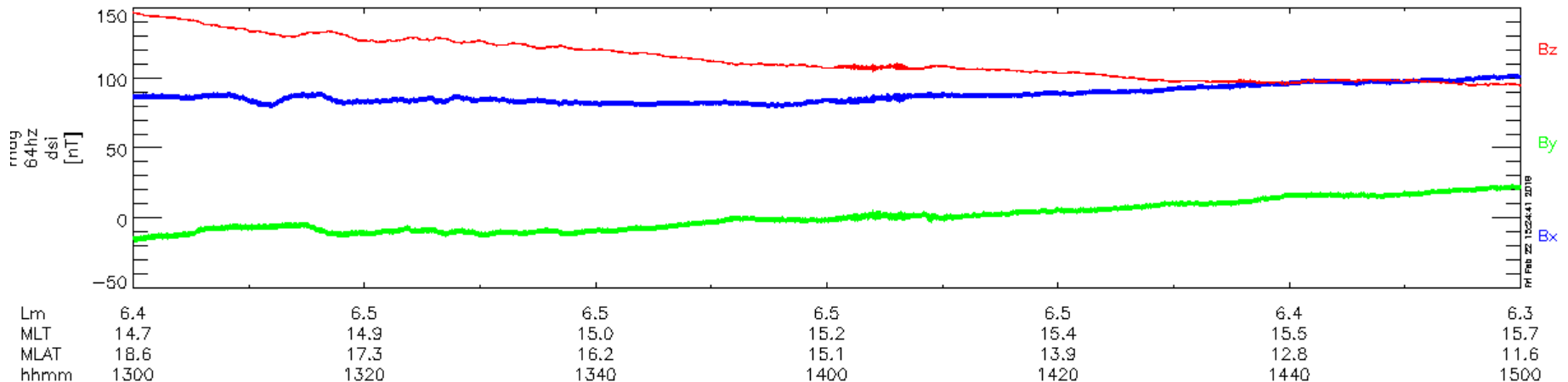
```
erg_load_mgf, datatype='64hz', coord='dsi', uname=uname, pass=pass
tplot, 'erg_mgf_l2_mag_64hz_dsi'
```

```
> tplot_names
```

```
...
27 erg_mgf_l2_date_time_64hz
28 erg_mgf_l2_mag_64hz_dsi
29 erg_mgf_l2_dyn_rng_64hz
30 erg_mgf_l2_quality_64hz
31 erg_mgf_l2_quality_64hz_gc
32 erg_mgf_l2_spin_phase_64hz
33 erg_mgf_l2_ti_64hz
```

```
'8sec'      : 8秒平均データ
'64hz'      : 64 Hz データ
'256hz'     : 256 Hzデータ
```

```
'sgi': Spinning satellite Geometry Inertia coordinate
'dsi': Despun Sun sector Inertia coordinate
```



1. パワースペクトル解析

1.4 tdpwrspcコマンドを用いて、各軸のパワースペクトルを計算します

```
; *****
; FFT analyses (MGF)
; *****
nbox = 2048
nshift = 256
```

```
split_vec, 'erg_mgf_l2_mag_64hz_dsi'
```

```
tdpwrspc, 'erg_mgf_l2_mag_64hz_dsi_x', nbox=nbox, nshift=nshift, bin=1
tdpwrspc, 'erg_mgf_l2_mag_64hz_dsi_y', nbox=nbox, nshift=nshift, bin=1
tdpwrspc, 'erg_mgf_l2_mag_64hz_dsi_z', nbox=nbox, nshift=nshift, bin=1
```

```
ERG> get_data,'erg_mgf_l2_mag_64hz_dsi_x_dpwrspc', data=data
ERG> help, data
** Structure <2479b5f0>, 3 tags, length=14710800, data
length=14710800, refs=1:
X      DOUBLE  Array[1794]      (時刻)
Y      FLOAT   Array[1794, 1024] (パワースペクトル)
V      FLOAT   Array[1794, 1024] (周波数)
```

Tips: 時系列波形のShort-time Fourier transformを計算します -> (元のtplot変数名)_dpwrspc
 tdpwrspc, 時系列波形のtplot変数名, nbox=FFT点数, nshift=overlap点数, bin=周波数方向の平均化点数

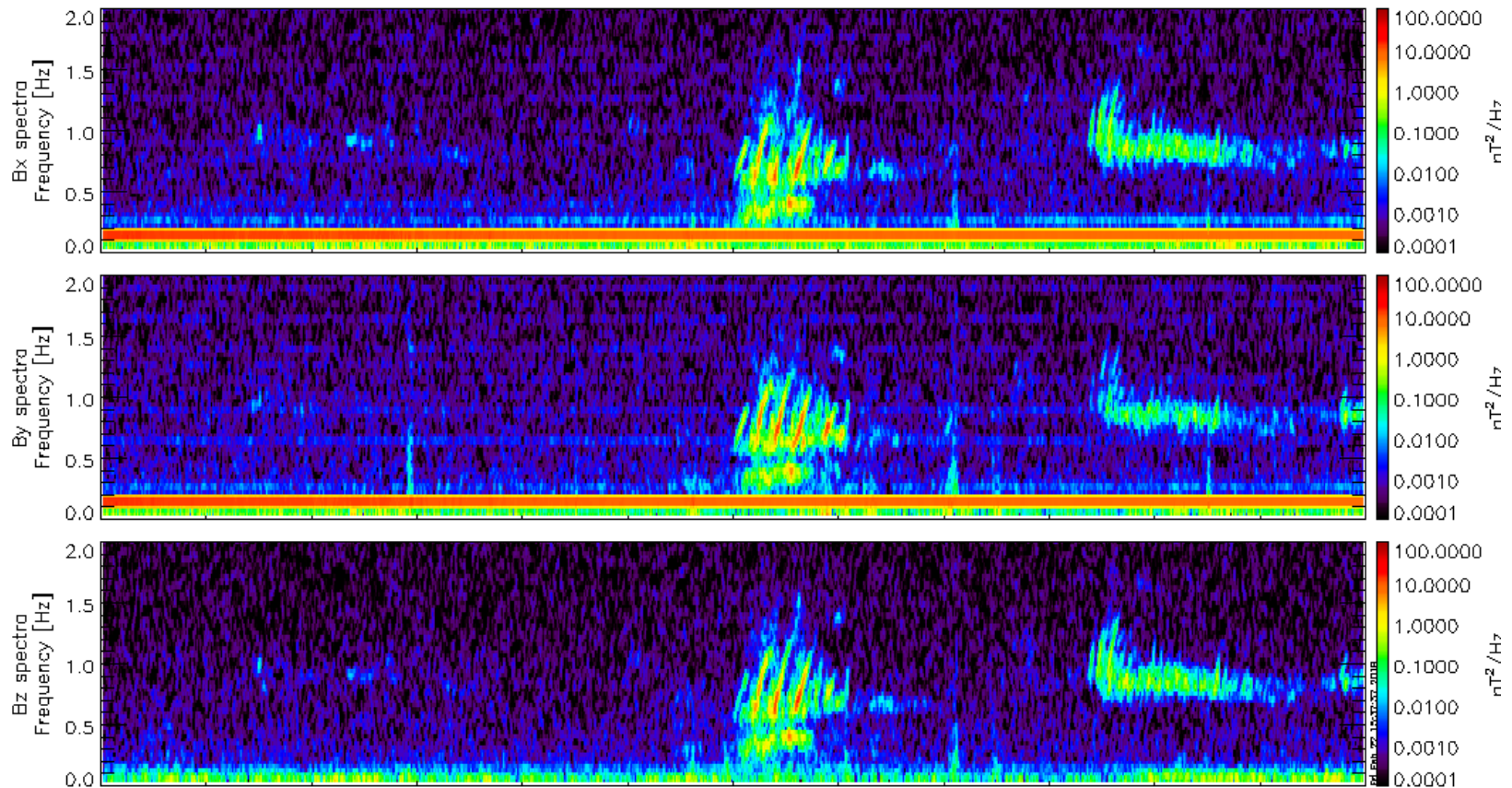
1.5 軸やラベルの設定を行います

```
;y-title
options, 'erg_mgf_l2_mag_64hz_dsi_x_dpwrspc', ytitle='Bx spectra'
options, 'erg_mgf_l2_mag_64hz_dsi_y_dpwrspc', ytitle='By spectra'
options, 'erg_mgf_l2_mag_64hz_dsi_z_dpwrspc', ytitle='Bz spectra'
;y-subtitle
options, 'erg_mgf_l2_mag_64hz_dsi_?_dpwrspc', ysubtitle='Frequency [Hz]'
;z-title
options, 'erg_mgf_l2_mag_64hz_dsi_?_dpwrspc', ztitle='nT!U2!N/Hz'
;
ylim, 'erg_mgf_l2_mag_64hz_dsi_?_dpwrspc', 0, 2, 0
zlim, 'erg_mgf_l2_mag_64hz_dsi_?_dpwrspc', 1E-4, 1E+2, 1
;
tplot, 'erg_mgf_l2_mag_64hz_dsi_' + ['x','y','z'] + '_dpwrspc'
```

1. パワースペクトル解析

1.6 計算したパワースペクトルを描画します

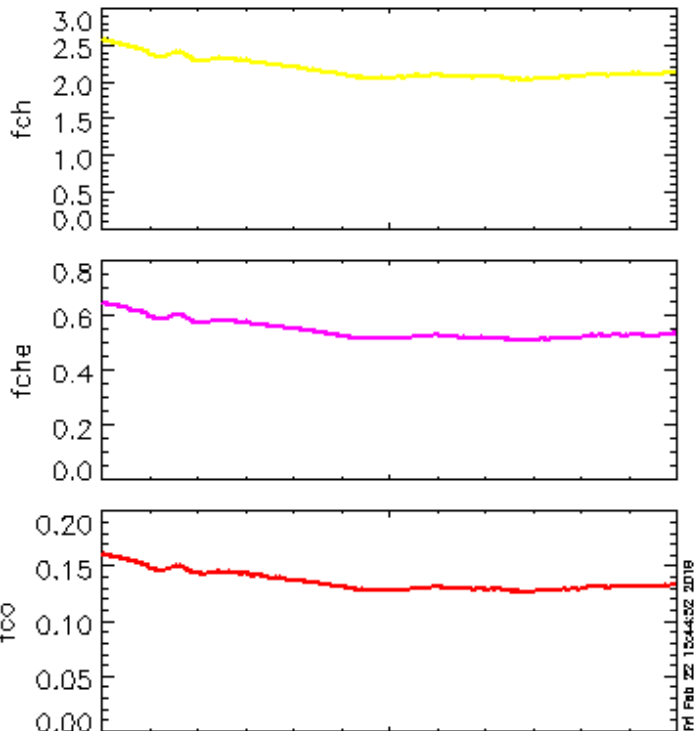
```
tplot, 'erg_mgf_l2_mag_64hz_dsi_' + ['x','y','z'] + '_dpwrspc'
```



Lm	6.4	6.5	6.5	6.4	6.3
MLT	14.7	14.9	15.2	15.4	15.7
MLAT	18.6	16.7	15.1	13.5	11.6
hhmm	1300	1330	1400	1430	1500

1. パワースペクトル解析

1.7 MGF 8secデータからイオンサイクロトロン周波数を計算します



Lm	6.4	6.5	6.3
MLT	14.7	15.2	15.7
MLAT	18.6	15.1	11.6
hhmm	1300	1400	1500
2017 Dec 11			

'8sec' : 8秒平均データ
'64hz' : 64 Hz データ
'256hz' : 256 Hz データ

```

; *****
; load MGF L2 CDF
; *****
erg_load_mgf, datatype='8sec', uname=uname, pass=pass

; *****
; overplot fce (EFD & MGF)
; *****

get_data, 'erg_mgf_l2_magt_8sec', data=data
fce = data.y / 10^(9.) * 1.6 * 10^(-19.) $
      / (9.1093D * 10^(-31.)) / 2. / !pi ; Hz
fch = fce / 1837.

;
store_data, 'fch', data={x:data.x, y:fch}
store_data, 'fche', data={x:data.x, y:fch/4.}
store_data, 'fco', data={x:data.x, y:fch/16.}

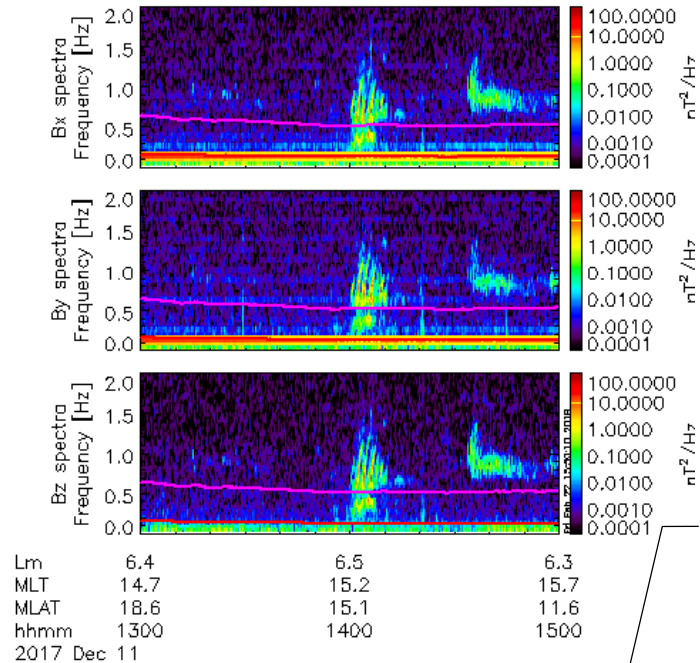
;
options, 'fch', colors=fsc_color('yellow'), thick=2
options, 'fche', colors=fsc_color('magenta'), thick=2
options, 'fco', colors=fsc_color('red'), thick=2

;
tplot, ['fch', 'fche', 'fco']

```

1. パワースペクトル解析

1.8 イオンサイクロトロン周波数を重ねてプロットします



Tips: 複数のtplot変数をマージしたtplot変数を作る (store_data)

store_data, tplot変数1, data=[tplot変数2, tplot変数3]

tplot変数2とtplot変数3をマージして, tplot変数1に格納する。

といっても, tplot変数1の実体は 'tplot変数2, tplot変数3' という単なる文字列

```
store_data, 'erg_mgf_l2_mag_64hz_dsi_x_dpwrspc_gyro',
data=['erg_mgf_l2_mag_64hz_dsi_x_dpwrspc', 'fch', 'fche', 'fco']
;
store_data, 'erg_mgf_l2_mag_64hz_dsi_y_dpwrspc_gyro',
data=['erg_mgf_l2_mag_64hz_dsi_y_dpwrspc', 'fch', 'fche', 'fco']
;
store_data, 'erg_mgf_l2_mag_64hz_dsi_z_dpwrspc_gyro',
data=['erg_mgf_l2_mag_64hz_dsi_z_dpwrspc', 'fch', 'fche', 'fco']
;
ylim, 'erg_mgf_l2_mag_64hz_dsi_?_dpwrspc_gyro', 0, 2, 0
;
tplot, 'erg_mgf_l2_mag_64hz_dsi_' + ['x','y','z'] + '_dpwrspc_gyro'
```

1. パワースペクトル解析

Tips:

スピン周波数近傍の波の解析は、
非スピン座標系 (あらせ衛星では DSI座標系)で行うこと

$$V_s \cos \omega_{\text{signal}} t \cos \omega_{\text{spin}} t$$

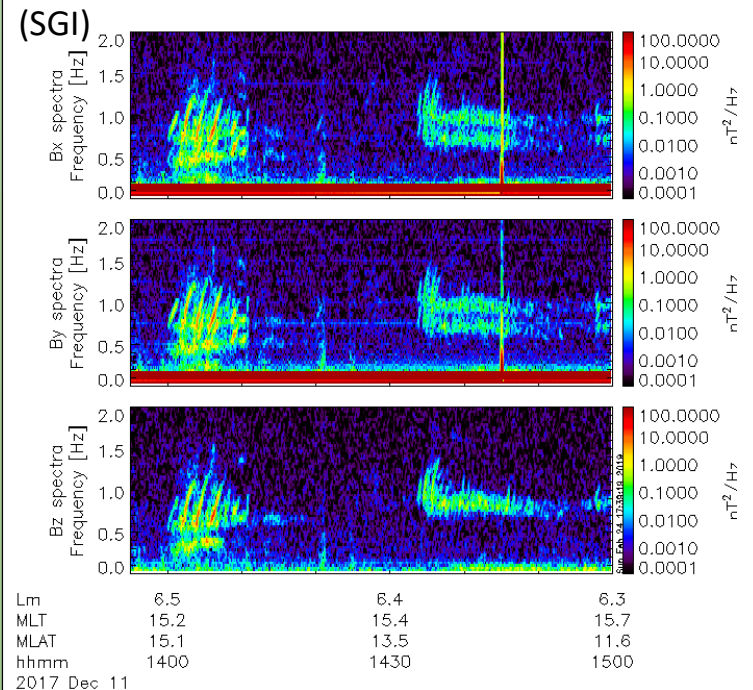
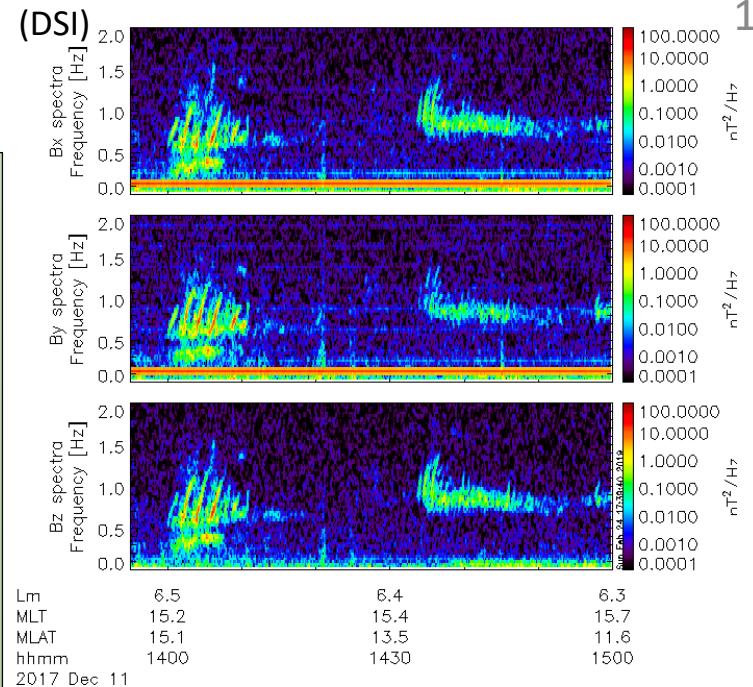
$$= \frac{V_s}{2} [\cos(\omega_{\text{spin}} t + \omega_{\text{signal}} t) + \cos(\omega_{\text{spin}} t - \omega_{\text{signal}} t)]$$

```

, *****
; load MGF 64 Hz CDF (SGI)
, *****
erg_load_mgf, datatype='64hz', coord='sgi', uname=uname, pass=pass
tplot, 'erg_mgf_l2_mag_64hz_sgi'

, *****
; FFT analyses (MGF, SGI)
, *****
nbox = 2048
nshift = 256
split_vec, 'erg_mgf_l2_mag_64hz_sgi'
tdpwrspc, 'erg_mgf_l2_mag_64hz_sgi_x', nbox=nbox, nshift=nshift, bin=1
tdpwrspc, 'erg_mgf_l2_mag_64hz_sgi_y', nbox=nbox, nshift=nshift, bin=1
tdpwrspc, 'erg_mgf_l2_mag_64hz_sgi_z', nbox=nbox, nshift=nshift, bin=1
;
options, 'erg_mgf_l2_mag_64hz_sgi_x_dpwrspc', ytitle='Bx spectra'
options, 'erg_mgf_l2_mag_64hz_sgi_y_dpwrspc', ytitle='By spectra'
options, 'erg_mgf_l2_mag_64hz_sgi_z_dpwrspc', ytitle='Bz spectra'
options, 'erg_mgf_l2_mag_64hz_sgi_?_dpwrspc', ysubtitle='Frequency [Hz]'
options, 'erg_mgf_l2_mag_64hz_sgi_?_dpwrspc', ztitle='nT!U2!N/Hz'
;
ylim, 'erg_mgf_l2_mag_64hz_sgi_?_dpwrspc', 0, 2, 0
zlim, 'erg_mgf_l2_mag_64hz_sgi_?_dpwrspc', 1E-4, 1E+2, 1
tplot, 'erg_mgf_l2_mag_64hz_sgi_' + ['x','y','z'] + '_dpwrspc'

```



2. 時間波形からスペクトルマトリクスを作る

2.1 MGF 64Hzデータ(DSI座標系)から、スペクトルマトリクスの要素を計算します

```

; *****
; calculate spectral matrix elements
; *****
get_data, 'erg_mgf_l2_mag_64hz_dsi', data=mgf_64hz

ndata = n_elements(mgf_64hz.x)

```

ERG> help, 'mgf_64hz_fft'

MGF_64HZ_FFT DCOMPLEX = Array[1794, 2048, 3]

```

mgf_64hz_fft = dcomplexarr((ndata-nbox)/nshift+1, nbox, 3)
mgf_64hz_t = dblarr((ndata-nbox)/nshift+1)
w = hanning(nbox, /double)
fs = 1.0 / (mgf_64hz.x[1] - mgf_64hz.x[0])

```

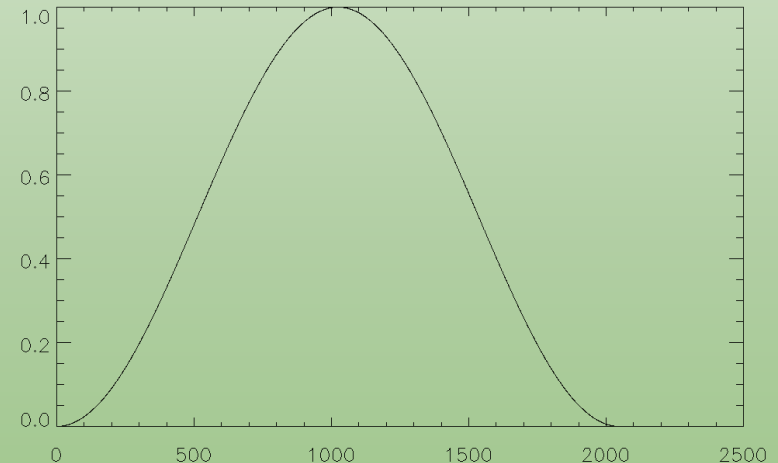
ERG> help,w

W DOUBLE = Array[2048]

ERG> plot, indgen(n_elements(w)), w

ERG> help,fs

FS DOUBLE = 64.103683



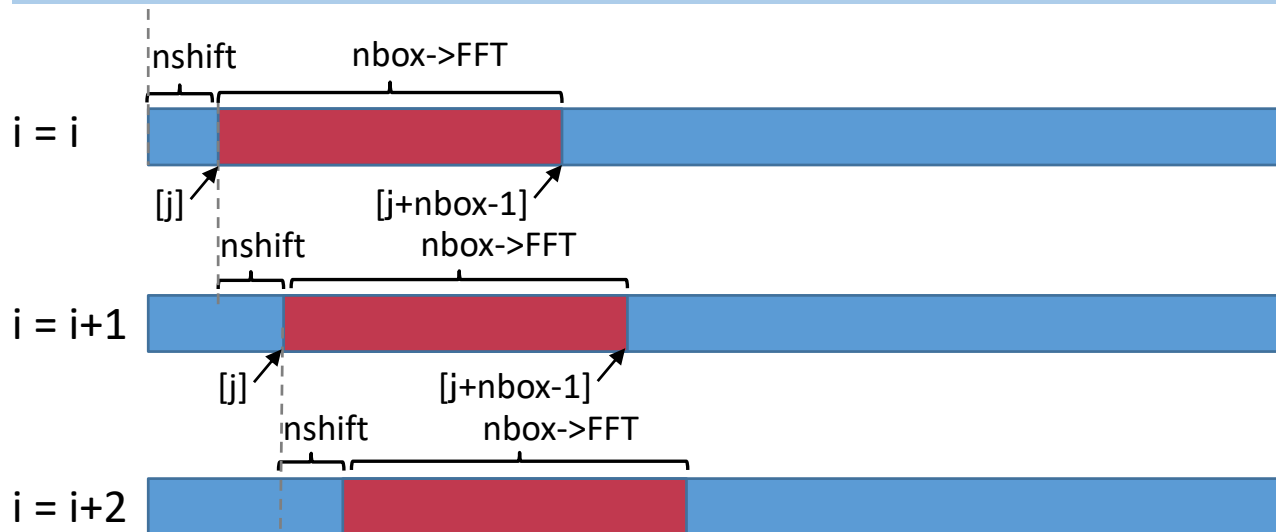
2. 時間波形からスペクトルマトリクスを作る

2.1 MGF 64Hzデータ(DSI座標系)から, スペクトルマトリクスの要素を計算します(続き)

```
ERG> help,mgf_64hz_fft[0, 0, 0]
<Expression> DCOMPLEX =( 43.351752, 0.00000000)
```

```
ERG> help,mgf_64hz_fft[0, 1, 0]
<Expression> DCOMPLEX =( -21.687504, 0.015383164)
```

```
i = 0L
for j=0L, ndata-nbox-1, nshift do begin
  for k=0, 2 do begin
    mgf_64hz_fft[i,*,k] = fft(mgf_64hz.y[j:j+nbox-1,k] * w)
  endfor
  mgf_64hz_t[i++] = mgf_64hz.x[j]
endfor
```



2. 時間波形からスペクトルマトリクスを作る

2.1 MGF 64Hzデータ(DSI座標系)から、スペクトルマトリクスの要素を計算します(続き)

$$S_B(f) = \frac{1}{N} \sum_{n=0}^N \begin{pmatrix} |B_u^{(n)}|^2 & B_u^{(n)} B_v^{(n)*} & B_u^{(n)} B_w^{(n)*} \\ B_v^{(n)} B_u^{(n)*} & |B_v^{(n)}|^2 & B_v^{(n)} B_w^{(n)*} \\ B_w^{(n)} B_u^{(n)*} & B_w^{(n)} B_v^{(n)*} & |B_w^{(n)}|^2 \end{pmatrix}$$

ERG> help, t_s

T_S
DOUBLE = Array[1794]

ERG> help, freq

FREQ
DOUBLE = Array[1024]

ERG> help, w_ave

<Expression>
DOUBLE = 0.37500000

```
npt = (ndata-nbox)/nshift + 1
```

```
t_s = mgf_64hz_t[0:n_elements(mgf_64hz_t)-1]
```

```
freq = findgen(nbox/2) * fs / nbox
```

```
w_ave = average(w^2)
```

ERG> help, bubu1

BUBU1
DCOMPLEX = Array[1794, 2048]

```
bubu1 = mgf_64hz_fft[*,*,0] * conj(mgf_64hz_fft[*,*,0]) / w_ave
```

```
bubv1 = mgf_64hz_fft[*,*,0] * conj(mgf_64hz_fft[*,*,1]) / w_ave
```

```
bubw1 = mgf_64hz_fft[*,*,0] * conj(mgf_64hz_fft[*,*,2]) / w_ave
```

```
bvbv1 = mgf_64hz_fft[*,*,1] * conj(mgf_64hz_fft[*,*,0]) / w_ave
```

```
bvbv1 = mgf_64hz_fft[*,*,1] * conj(mgf_64hz_fft[*,*,1]) / w_ave
```

```
bvbw1 = mgf_64hz_fft[*,*,1] * conj(mgf_64hz_fft[*,*,2]) / w_ave
```

```
bwbu1 = mgf_64hz_fft[*,*,2] * conj(mgf_64hz_fft[*,*,0]) / w_ave
```

```
bwbv1 = mgf_64hz_fft[*,*,2] * conj(mgf_64hz_fft[*,*,1]) / w_ave
```

```
bwbw1 = mgf_64hz_fft[*,*,2] * conj(mgf_64hz_fft[*,*,2]) / w_ave
```

2. 時間波形からスペクトルマトリクスを作る

2.2 時間平均されたスペクトルマトリクスの要素を計算します

```

; *****
; time average of spectral matrix elements
; *****
bubu2 = dcomplexarr(npt, n_elements(freq))
bubv2 = dcomplexarr(npt, n_elements(freq))
bubw2 = dcomplexarr(npt, n_elements(freq))
bvbu2 = dcomplexarr(npt, n_elements(freq))
bvbv2 = dcomplexarr(npt, n_elements(freq))
bvbw2 = dcomplexarr(npt, n_elements(freq))
bwbu2 = dcomplexarr(npt, n_elements(freq))
bwbv2 = dcomplexarr(npt, n_elements(freq))
bwbw2 = dcomplexarr(npt, n_elements(freq))

for i=0, npt-1 do begin
  indx = i + indgen(3)
  if( (where(indx LT 0))[0] NE -1 ) then break
  if( (where(indx GT npt-1))[0] NE -1 ) then break

  for j=0, n_elements(freq)-1 do begin
    bubu2[i, j] = average(bubu1[indx, j])
    bubv2[i, j] = average(bubv1[indx, j])
    bubw2[i, j] = average(bubw1[indx, j])
    bvbu2[i, j] = average(bvbu1[indx, j])
    vbv2[i, j] = average(bvbv1[indx, j])
    vbw2[i, j] = average(bvbw1[indx, j])
    bwbu2[i, j] = average(bwbu1[indx, j])
    bwbv2[i, j] = average(bwbv1[indx, j])
    bwbw2[i, j] = average(bwbw1[indx, j])
  endfor
endfor
endfor

```

$$S_B(f) = \frac{1}{N} \sum_{n=0}^N \begin{pmatrix} |B_u^{(n)}|^2 & B_u^{(n)} B_v^{(n)*} & B_u^{(n)} B_w^{(n)*} \\ B_v^{(n)} B_u^{(n)*} & |B_v^{(n)}|^2 & B_v^{(n)} B_w^{(n)*} \\ B_w^{(n)} B_u^{(n)*} & B_w^{(n)} B_v^{(n)*} & |B_w^{(n)}|^2 \end{pmatrix}$$

ERG> help, bubu2
 BUBU2 DCOMPLEX = Array[1794, 1024]

2. 時間波形からスペクトルマトリクスを作る

2.3 各要素から、スペクトルマトリクスを構成します

$$S_B(f) = \frac{1}{N} \sum_{n=0}^N \begin{pmatrix} |B_u^{(n)}|^2 & B_u^{(n)} B_v^{(n)*} & B_u^{(n)} B_w^{(n)*} \\ B_v^{(n)} B_u^{(n)*} & |B_v^{(n)}|^2 & B_v^{(n)} B_w^{(n)*} \\ B_w^{(n)} B_u^{(n)*} & B_w^{(n)} B_v^{(n)*} & |B_w^{(n)}|^2 \end{pmatrix}$$

ERG> help, rr

RR DOUBLE = Array[3, 3, 1794, 1024, 2]

3: 列方向のindex

3: 行方向のindex

1794: 時刻のindex

1024: 周波数のindex

2: 実部/虚部のindex

```
; *****
; make spectral matrix
; *****
rr=dblarr(3, 3, npt, n_elements(freq), 2)

rr[0,0,*,*,0]=real_part(bubu2)
rr[1,0,*,*,0]=real_part(bubv2)
rr[2,0,*,*,0]=real_part(bubw2)
rr[0,1,*,*,0]=real_part(bvbv2)
rr[1,1,*,*,0]=real_part(bvbv2)
rr[2,1,*,*,0]=real_part(bvbv2)
rr[0,2,*,*,0]=real_part(bwbv2)
rr[1,2,*,*,0]=real_part(bwbv2)
rr[2,2,*,*,0]=real_part(bwbv2)

rr[0,0,*,*,1]=imaginary(bubu2)
rr[1,0,*,*,1]=imaginary(bubv2)
rr[2,0,*,*,1]=imaginary(bubw2)
rr[0,1,*,*,1]=imaginary(bvbv2)
rr[1,1,*,*,1]=imaginary(bvbv2)
rr[2,1,*,*,1]=imaginary(bvbv2)
rr[0,2,*,*,1]=imaginary(bwbv2)
rr[1,2,*,*,1]=imaginary(bwbv2)
rr[2,2,*,*,1]=imaginary(bwbv2)
```


3. 磁場座標系への回転行列

3.1 変動成分を取り除くために、MGF 64Hzデータ(DSI座標系)の移動平均を計算します

Tips: データの移動平均を計算する (tsmooth_in_time)

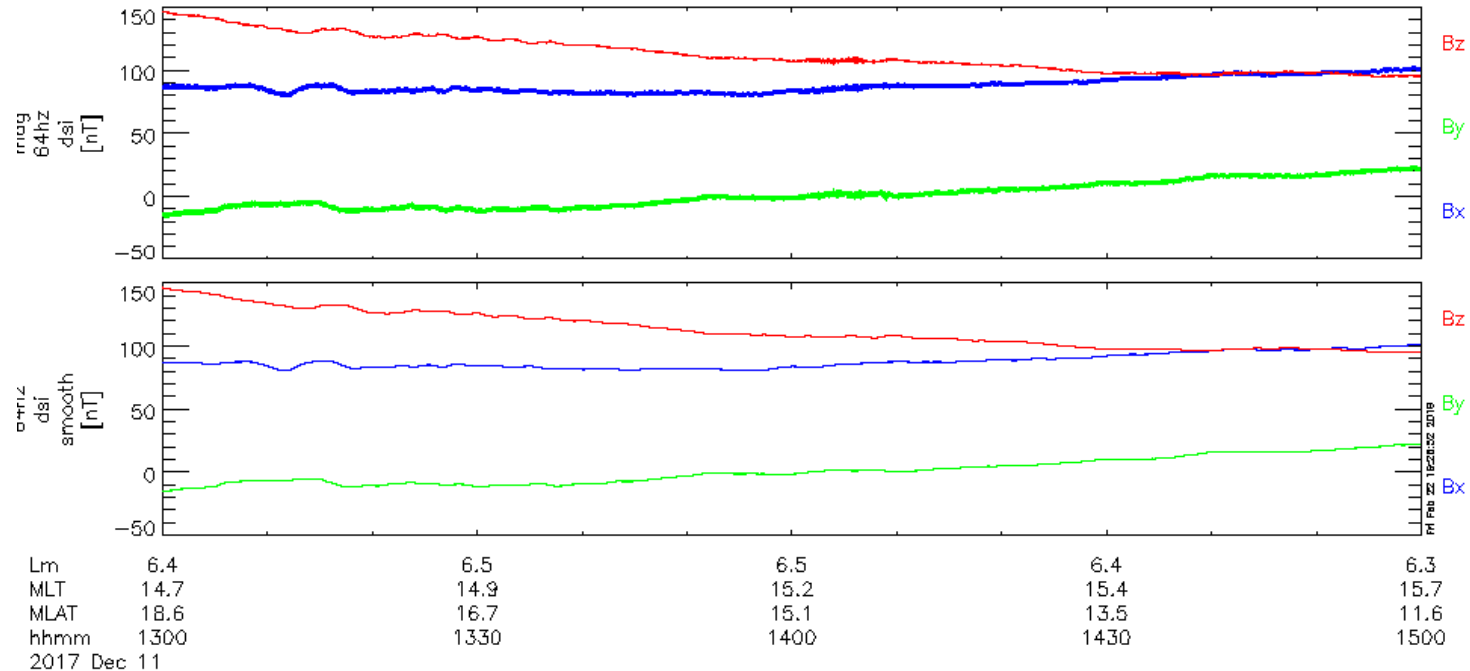
tsmooth_in_time, tplot変数1, 平均幅, newname=tplot変数2

tplot変数1の時系列データに対してrunning averageを計算し、結果をtplot変数2に格納する。
Running averageの結果は、平均する前のtplot変数と同じ時刻タグを付けて返される

```

; *****
; MGF matrix rotation
; *****
tsmooth_in_time, 'erg_mgf_12_mag_64hz_dsi', 60., newname='erg_mgf_12_mag_64hz_dsi_smooth'
tplot, ['erg_mgf_12_mag_64hz_dsi', 'erg_mgf_12_mag_64hz_dsi_smooth']

```



3. 磁場座標系への回転行列

3.2 スムージングされたMGF 64Hzデータ(DSI座標系)を, スペクトルマトリクスの時刻タグに合わせて内挿します

```
split_vec, 'erg_mgf_l2_mag_64hz_dsi_smooth'  
  
tinterpol_mxn, 'erg_mgf_l2_mag_64hz_dsi_smooth_x', t_s, $  
  newname='erg_mgf_l2_mag_64hz_dsi_smooth_x_interp'  
tinterpol_mxn, 'erg_mgf_l2_mag_64hz_dsi_smooth_y', t_s, $  
  newname='erg_mgf_l2_mag_64hz_dsi_smooth_y_interp'  
tinterpol_mxn, 'erg_mgf_l2_mag_64hz_dsi_smooth_z', t_s, $  
  newname='erg_mgf_l2_mag_64hz_dsi_smooth_z_interp'
```

Tips: データを内挿する (tinterpol_mxn)

tinterpol_mxn, **tplot変数1**, **tplot変数2**, newname=**tplot変数3**

tplot変数2の時刻タグに合わせて, **tplot変数1**のデータを内挿し, **tplot変数3**に格納する
newnameを省略した場合は, '元のtplot変数名_interp'という名前の変数に格納される

```
get_data, 'erg_mgf_l2_mag_64hz_dsi_smooth_x_interp', data = data_x  
get_data, 'erg_mgf_l2_mag_64hz_dsi_smooth_y_interp', data = data_y  
get_data, 'erg_mgf_l2_mag_64hz_dsi_smooth_z_interp', data = data_z
```

3. 磁場座標系への回転行列

3.3 内挿したスムージング済みMGF 64Hzデータ(DSI座標系)から, DSI座標系 → 背景磁場座標系($B_0 // z$)の変換行列を求めます

```
rotmat=dblarr(3,3,n_elements(data_x.x))
rotmat_t=dblarr(3,3,n_elements(data_x.x))

for i=0, n_elements(data_x.x)-1 do begin

  bvec=[data_x.y[i],data_y.y[i],data_z.y[i]]
  zz=[0.,0.,1.]

  yhat=crossp(zz,bvec)
  xhat=crossp(yhat,bvec)
  zhat=bvec

  xhat=xhat/sqrt(xhat[0]^2+xhat[1]^2+xhat[2]^2)
  yhat=yhat/sqrt(yhat[0]^2+yhat[1]^2+yhat[2]^2)
  zhat=zhat/sqrt(zhat[0]^2+zhat[1]^2+zhat[2]^2)

  rotmat[*,* ,i]=([[xhat],[yhat],[zhat]])
  rotmat_t[*,* ,i]=transpose([[xhat],[yhat],[zhat]])
endfor
```

3. 磁場座標系への回転

3.4 求めた回転行列を用いて,
スペクトルマトリクス (DSI座標系)を背景磁場座標系($B_0 // z$)に変換します

```

; matrix rotation
for i=0, n_elements(t_s)-1 do begin
  for j=0, n_elements(freq)-1 do begin
    for k=0, 1 do begin

      rr[*,*,i,j,k] = $
        (reform(rotmat[i,*,*]) ## rr[*,*,i,j,k]) ## reform(rotmat_t[i,*,*])

    endfor
  endfor
endfor
endfor

```

ERG> help, rotmat[0,*,*]
<Expression> DOUBLE = Array[1, 3, 3]

ERG> help, reform(rotmat[0,*,*])
<Expression> DOUBLE = Array[3, 3]

分散共分散行列の定義
 $C = E(XX^T)$

回転行列を R
 $X' = RX$ とする

$$\begin{aligned}
 C' &= E(X'X'^T) \\
 &= E(RXX^TR^T) \\
 &= RE(XX^T)R^T \\
 &= RCR^T
 \end{aligned}$$

Tips: 行列の積 (A # B, A ## B)

$$C = A \# B \rightarrow C = BA$$

$$C = A \## B \rightarrow C = AB$$

4. パワースペクトルをプロット

- 4.1 計算したスペクトルマトリクスから、パワースペクトルをプロットします
 スペクトルマトリクスの対角成分は、各軸のパワースペクトルを表します
 (対角成分のimaginary partは常にゼロ)

$$S = \begin{bmatrix} \langle B_x B_x^* \rangle & \langle B_x B_y^* \rangle & \langle B_x B_z^* \rangle \\ \langle B_y B_x^* \rangle & \langle B_y B_y^* \rangle & \langle B_y B_z^* \rangle \\ \langle B_z B_x^* \rangle & \langle B_z B_y^* \rangle & \langle B_z B_z^* \rangle \end{bmatrix}$$

$$= \begin{bmatrix} |B_x|^2 & |B_x||B_y|e^{i(\phi_x-\phi_y)} & |B_x||B_z|e^{i(\phi_x-\phi_z)} \\ |B_y||B_x|e^{-i(\phi_x-\phi_y)} & |B_y|^2 & |B_y||B_z|e^{i(\phi_y-\phi_z)} \\ |B_z||B_x|e^{-i(\phi_x-\phi_z)} & |B_z||B_y|e^{-i(\phi_y-\phi_z)} & |B_z|^2 \end{bmatrix}$$

```
ERG> help, rr[0, 0, *, *, 0]
```

```
<Expression> DOUBLE = Array[1, 1, 1794, 1024]
```

```
ERG> help, reform( rr[0, 0, *, *, 0] )
```

```
<Expression> DOUBLE = Array[1794, 1024]
```

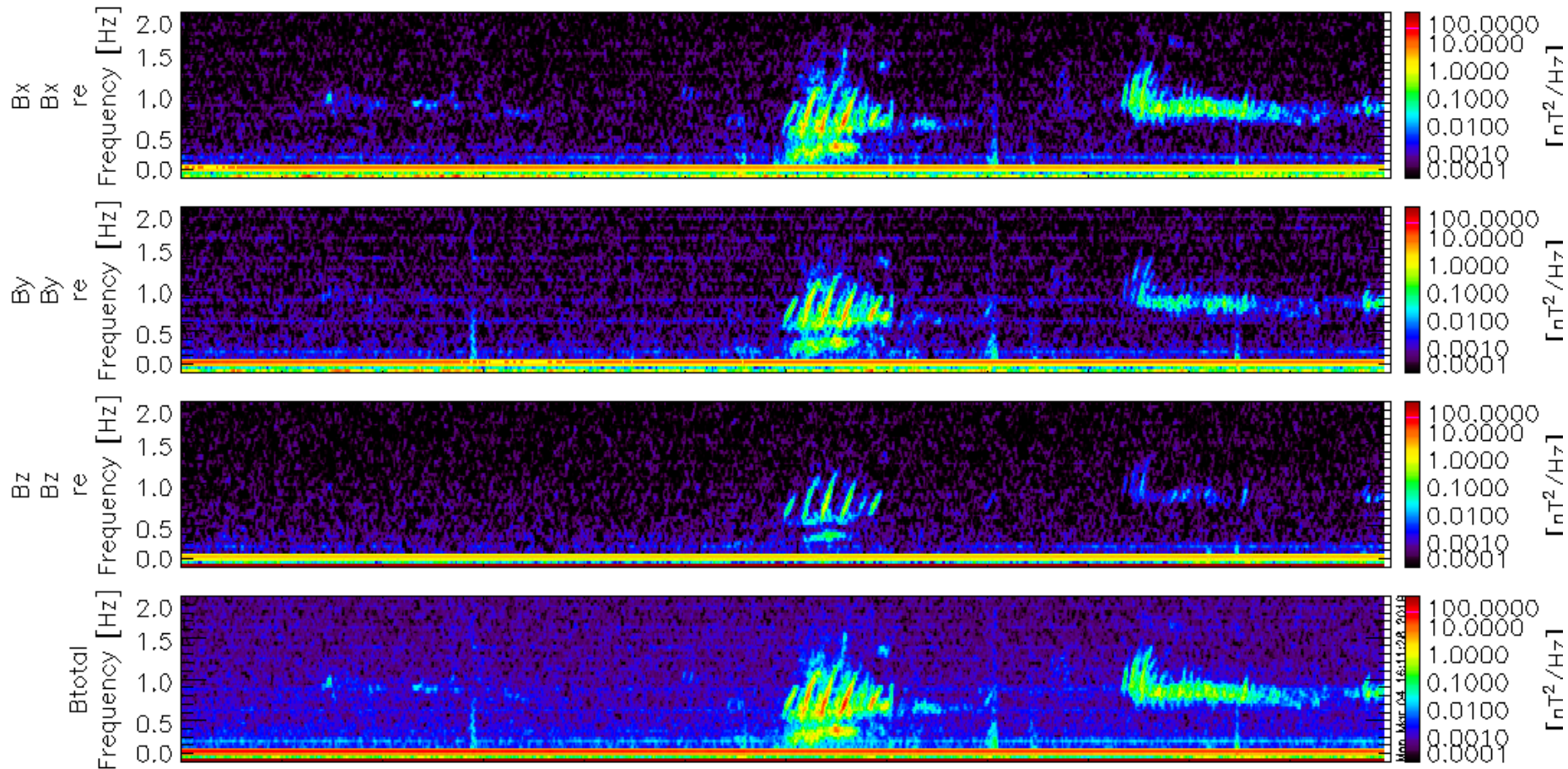
```
; *****
; auto-spectra
; *****

store_data, 'Bx_Bx_re', data={x:t_s, y:reform(rr[0,0,*,*,0])/(fs/nbox), v:freq}
store_data, 'By_By_re', data={x:t_s, y:reform(rr[1,1,*,*,0])/(fs/nbox), v:freq}
store_data, 'Bz_Bz_re', data={x:t_s, y:reform(rr[2,2,*,*,0])/(fs/nbox), v:freq}
calc, '"Btotal"="Bx_Bx_re"+"By_By_re"+"Bz_Bz_re"'

options, ['Bx_Bx_re','By_By_re','Bz_Bz_re','Btotal'], ysubtitle = 'Frequency [Hz]'
options, ['Bx_Bx_re','By_By_re','Bz_Bz_re','Btotal'], ztitle = '[nT!U2!N/Hz]'
options, ['Bx_Bx_re','By_By_re','Bz_Bz_re','Btotal'], spec=1
ylim, ['Bx_Bx_re','By_By_re','Bz_Bz_re','Btotal'], 0, 2, 0 ; Hz
zlim, ['Bx_Bx_re','By_By_re','Bz_Bz_re','Btotal'], 1E-4, 1E2, 1 ; nT^2/Hz
;
tplot, ['Bx_Bx_re','By_By_re','Bz_Bz_re','Btotal']
```

4. パワースペクトルをプロット

4.1 計算したスペクトルマトリクスから、パワースペクトルをプロットします



Lm	6.4	6.5	6.5	6.4	6.3
MLT	14.7	14.9	15.2	15.4	15.7
MLAT	18.6	16.7	15.1	13.5	11.6
hhmm	1300	1330	1400	1430	1500

5. Means' methodによるwave normal angle推定

5.1 スペクトルマトリクス (背景磁場座標系)にMeans' methodを適用します

```

; *****
; wave normal angle
; *****
wna = dblarr(n_elements(t_s), n_elements(freq))

for i=0, n_elements(t_s)-1 do begin
  for j=0, n_elements(freq)-1 do begin
    wna_x = rr[2,1,i,j,1] / sqrt(rr[2,1,i,j,1]^2 + rr[0,2,i,j,1]^2 + rr[1,0,i,j,1]^2)
    wna_y = rr[0,2,i,j,1] / sqrt(rr[2,1,i,j,1]^2 + rr[0,2,i,j,1]^2 + rr[1,0,i,j,1]^2)
    wna_z = rr[1,0,i,j,1] / sqrt(rr[2,1,i,j,1]^2 + rr[0,2,i,j,1]^2 + rr[1,0,i,j,1]^2)

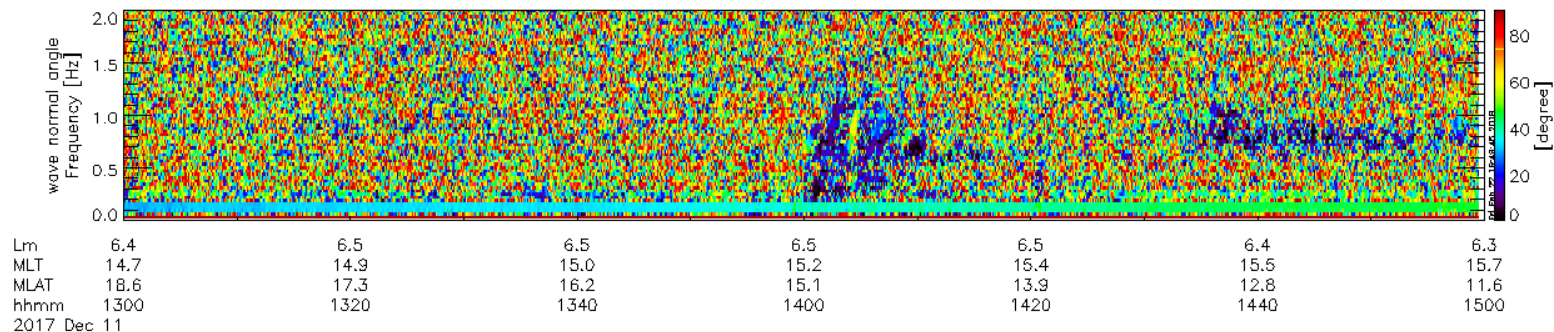
    wna[i,j] = abs(atan(sqrt(wna_x^2+wna_y^2)/wna_z)/!dior)
  endfor
endfor

store_data, 'kvec', data={x:t_s, y:wna, v:freq}
options, 'kvec', ytitle='wave normal angle', ztitle='[degree]', ysubtitle='Frequency
[Hz]', spec = 1
ylim, 'kvec', 0, 2, 0 ; Hz
zlim, 'kvec', 0, 90, 0 ; degree

```

Means et al. (1972)

$$k \parallel \begin{pmatrix} \text{Im}(S_{YZ}) \\ \text{Im}(S_{ZX}) \\ \text{Im}(S_{XY}) \end{pmatrix}$$



6. Polarizationを計算

6.1 スペクトルマトリクス(背景磁場座標系)から、偏波(polarization)を計算します

```

; *****
; polarization
; *****
polarization = $
  dblarr(n_elements(t_s), n_elements(freq))

for i=0, n_elements(t_s)-1 do begin
  for j=0, n_elements(freq)-1 do begin
    polarization[i,j] = (2 * rr[1,0,i,j,1]) / (rr[0,0,i,j,0] + rr[1,1,i,j,0])
  endfor
endfor

store_data, 'polarization', data={x:t_s, y:polarization, v:freq}
options, 'polarization', ytitle='polarization', $
  ysubtitle='Frequency [Hz]', spec = 1
ylim, 'polarization', 0, 2, 0 ; Hz
zlim, 'polarization', -1, 1, 0 ;

```

磁力線に対して垂直な面における信号を、次のようにおく

$$H_x(t) = a_x(t)e^{i(\omega t + \phi_x(t))}$$

$$H_y(t) = a_y(t)e^{i(\omega t + \phi_y(t))}$$

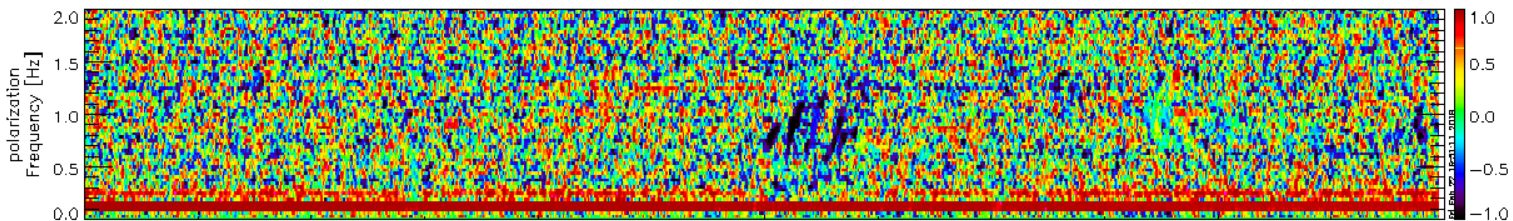
スペクトルマトリクスは

$$S = \begin{bmatrix} \langle H_x H_x^* \rangle & \langle H_x H_y^* \rangle \\ \langle H_y H_x^* \rangle & \langle H_y H_y^* \rangle \end{bmatrix} \\ = \begin{bmatrix} \langle a_x^2 \rangle & \langle a_x a_y e^{i(\phi_x - \phi_y)} \rangle \\ \langle a_x a_y e^{-i(\phi_x - \phi_y)} \rangle & \langle a_y^2 \rangle \end{bmatrix}$$

$$a_x a_y e^{i(\phi_x - \phi_y)} = a_x a_y \cos(\phi_x - \phi_y) + i a_x a_y \sin(\phi_x - \phi_y)$$

偏波率 η は

$$\eta = \frac{\langle 2 a_x a_y \sin(\phi_x - \phi_y) \rangle}{\langle a_x^2 \rangle + \langle a_y^2 \rangle} = \frac{2 \text{Im}(s_{10})}{s_{00} + s_{11}}$$



Lm	6.4	6.5	6.5	6.5	6.5	6.4	6.3
MLT	14.7	14.9	15.0	15.2	15.4	15.5	15.7
MLAT	18.6	17.3	16.2	15.1	13.9	12.8	11.6
hhmm	1300	1320	1340	1400	1420	1440	1500

2017 Dec 11

7. 解析結果をマスク

7.1 パワースペクトル(Btotal)を使って、解析結果をマスクします

```

; *****
; mask
; *****
get_data, 'Btotal', data=data_ref

; kvec
get_data, 'kvec', data=data, dlim=dlim, lim=lim
data.y[where(data_ref.y LT 5E-2)] = 'NaN'
store_data, 'kvec_mask', data={x:data.x, y:data.y, v:data.v}, dlim=dlim, lim=lim

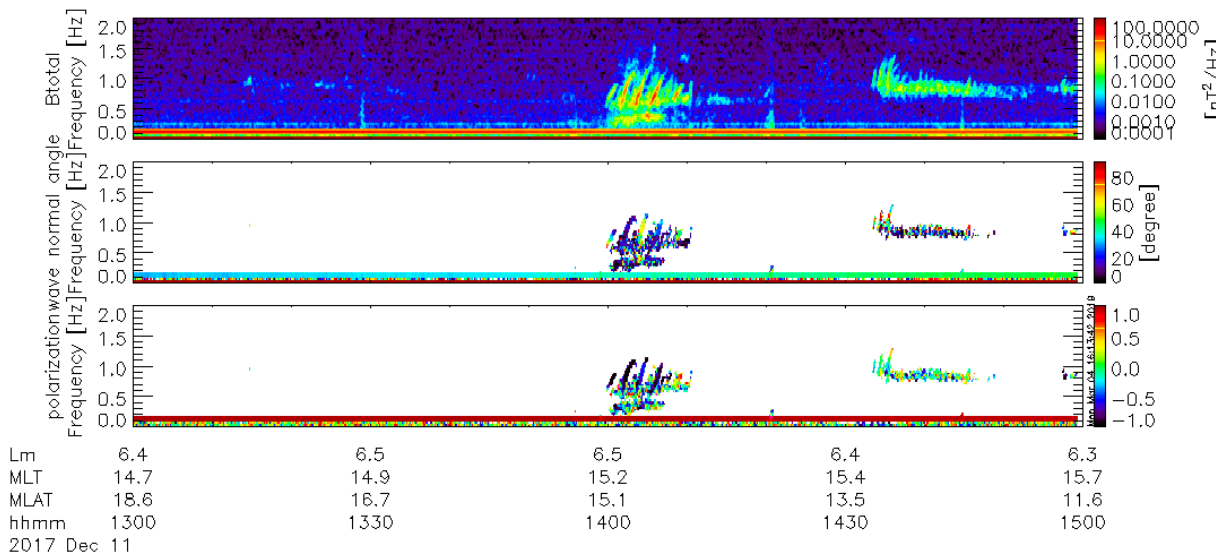
; polarization
get_data, 'polarization', data=data, dlim=dlim, lim=lim
data.y[where(data_ref.y LT 5E-2)] = 'NaN'
store_data, 'polarization_mask', data={x:data.x, y:data.y, v:data.v}, dlim=dlim, lim=lim

tplot, ['Btotal', 'kvec_mask', 'polarization_mask']

```

Tips: 条件を満たす配列要素の番号を返す (where)

ret = where(配列 <演算子> 条件)
 配列の中から、引数として与えられる条件式を満たす要素の番号を、配列として返す



8. まとめてプロット

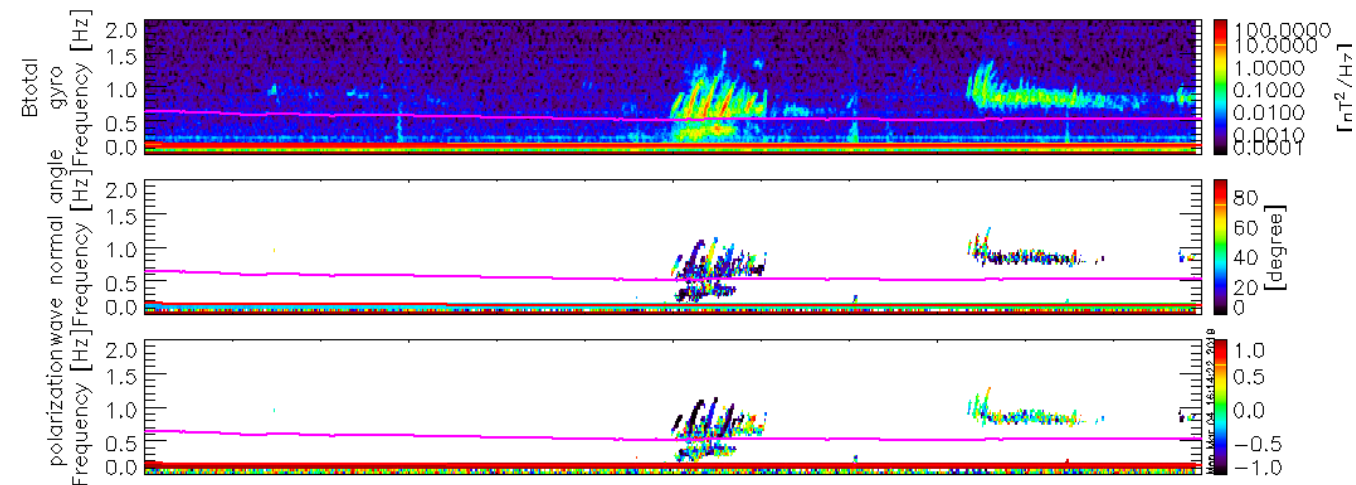
8.1 MGF 8秒平均値から電子サイクロトロン周波数を計算して、重ねてプロットします

```

; *****
; overplot fce
; *****
store_data, 'Btotal_gyro', data=['Btotal', 'fch', 'fche', 'fco']
store_data, 'kvec_mask_gyro', data=['kvec_mask', 'fch', 'fche', 'fco']
store_data, 'polarization_mask_gyro', $
  data=['polarization_mask', 'fch', 'fche', 'fco']
;
ylim, 'Btotal_gyro', 0, 2, 0
ylim, 'kvec_mask_gyro', 0, 2, 0
ylim, 'polarization_mask_gyro', 0, 2, 0

tplot, ['Btotal', 'kvec_mask', 'polarization_mask'] + '_gyro'

```



Lm	6.4	6.5	6.5	6.4	6.3
MLT	14.7	14.9	15.2	15.4	15.7
MLAT	18.6	16.7	15.1	13.5	11.6
hhmm	1300	1330	1400	1430	1500

Memo