



SuperDARN可視化チュートリアル

IUGONETデータ解析講習会@STEL (Feb. 24, 2012)

Tomoaki Hori (STEL)

チュートリアルの目的

- ▶ SDデータ可視化の方法(の1つ)を習得する。
 - ▶ **UDAS by IUGONET** (SD plug-in by ERG-SC)を使う方法
- ▶ 典型的な可視化形態であるRTIプロット、2Dマッププロットを作成できるようにする。

謝辞

- ▶ データはERG-SC(<http://gemsissc.stelab.nagoya-u.ac.jp/erg/>)作成・配布のfitacfデータ(CDFフォーマット)
- ▶ この講習で使うIDLプログラムもERG-SCとIUGONETとの共同開発
- ▶ UDASは、THEMISチームによって開発されたTHEMIS Data Analysis Software suite (TDAS, <http://themis.ssl.berkeley.edu/software.shtml>) をベースにしている

SuperDARNレーダーとは?



Hokkaido HF radar

▶ Super Dual Auroral Radar Network (SuperDARN)

▶ HF帯電波を用いたcoherentレーダーの世界的ネットワーク

▶ 観測対象

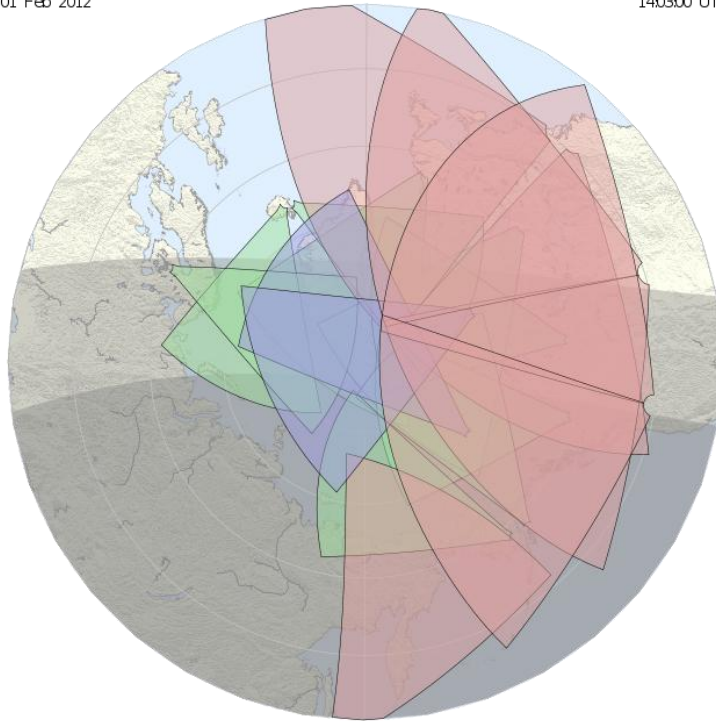
▶ 電離圏F層、E層のプラズマの動き(電場ドリフト)

▶ 中間圏エコー、中性風(ある仮定の元で)、…

全SuperDARNレーダーの視野

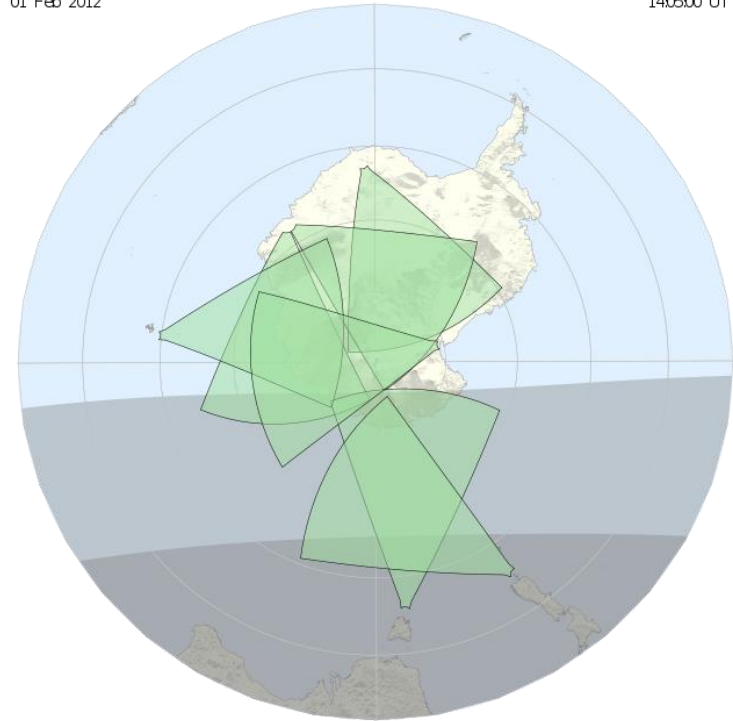
01 Feb 2012

140300 UT



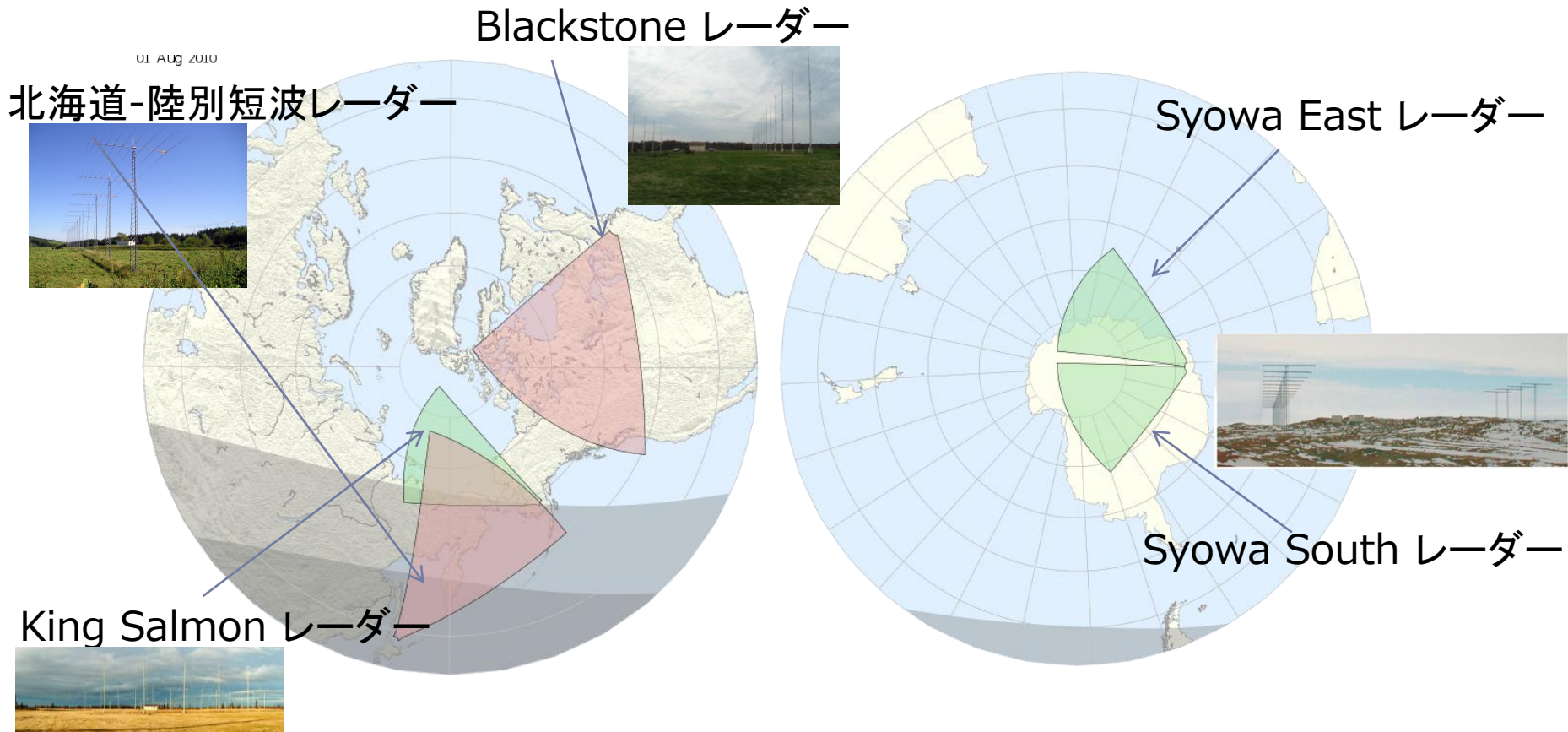
01 Feb 2012

140500 UT



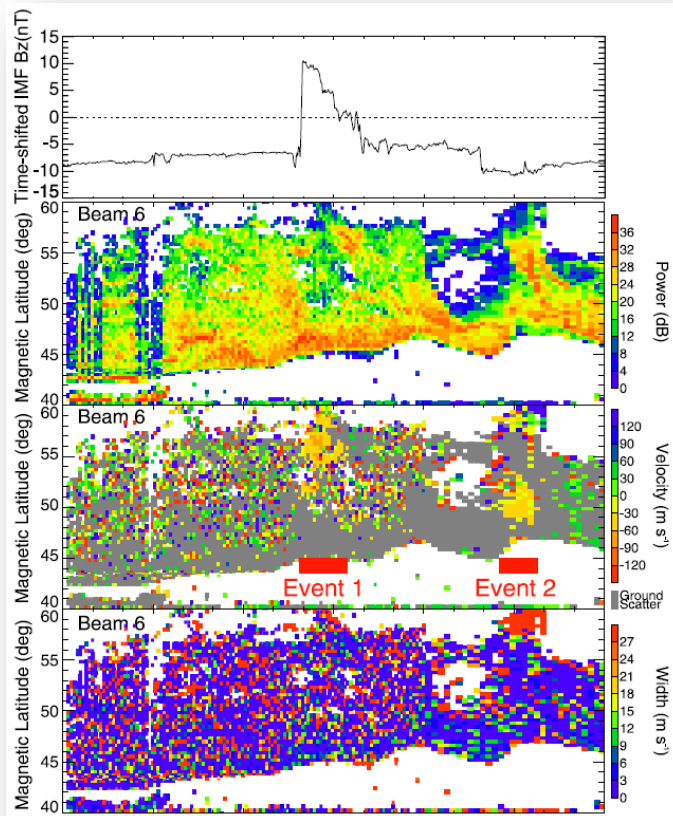
▶ 2012年2月現在

UDASで使える SuperDARNレーダーデータの視野



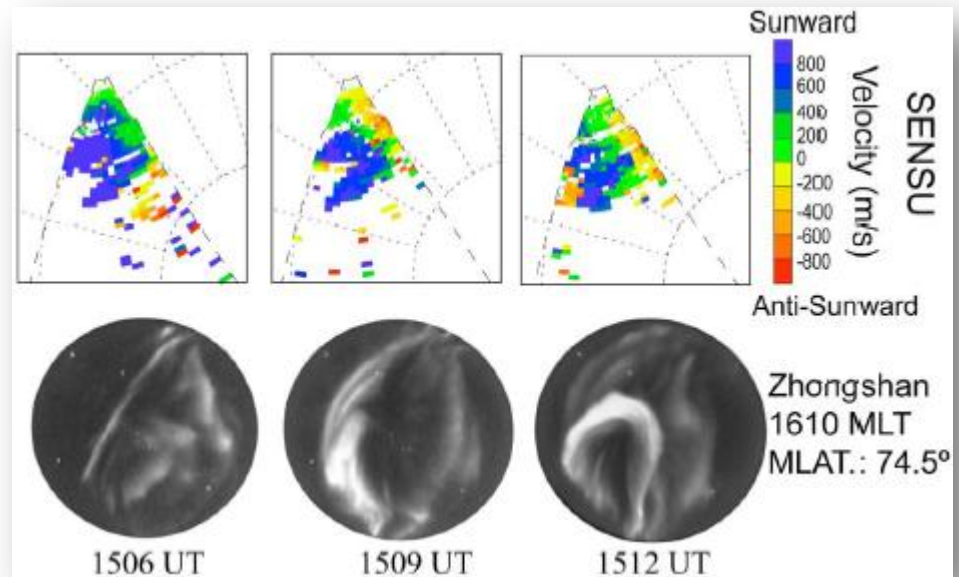
- ▶ この後の講習で使用するソフトウェアは、これら5つのレーダーのデータを扱うことができる（今後扱えるレーダーどんどん増やす予定）

SuperDARNデータのプロット



[Ebihara et al., 2008]

Range-Time-Intensity (RTI) プロットの例
変動のタイミング・周期を詳細に見るのに使われる



[Liu et al., 2011]

緯度-経度グリッド上での2Dプロットの例
ある時刻の2D空間プロファイルを見るのに使われる

データの読み込み

レーダーのデータを読み込む

▶ iug_load_sdfit を使う

- ▶ get_support_dataキーワードをセットすることで、座標変換に必要なパラメータなども一緒に読み込む

siteキーワードでレーダー名を指定する

hok: 北海道-陸別短波レーダー
ksr: King Salmon レーダー (アラスカ)
sye: Syowa East レーダー (南極昭和基地)
sys: Syowa South レーダー (")
Bks: Blackstone レーダー (北米)

iug_load_sdfit コマンド1つで

(実際はerg_load_sdfit のエイリアス)

fitacfデータをCDFファイルにしたものを自動ダウンロード(ローカルPC上に保存される)



各種パラメータをtplot変数としてIDL上に読み込む

データの保存先は
Windows: c:\%data
Unix系: ~/data の下

をやってくれる

```
THEMIS> timespan, '2012-01-22'  
THEMIS> iug_load_sdfit, site='hok', /get_support_data  
...  
(ロードの過程がいっぱい表示される)  
...  
##### RULES OF THE ROAD #####  
Data distributed with this CDF file can be used ...  
##### RULES OF THE ROAD #####  
THEMIS>  
THEMIS> tplot_names  
1 sd_hok_azim_no_1  
2 sd_hok_pwr_1  
3 sd_hok_pwr_err_1  
4 sd_hok_spec_width_1  
5 sd_hok_spec_width_err_1  
6 sd_hok_vlos_1  
7 sd_hok_vlos_err_1  
8 sd_hok_echo_flag_1  
9 sd_hok_quality_1  
10 sd_hok_quality_flag_1  
11 sd_hok_vnorth_1  
12 sd_hok_veast_1  
13 sd_hok_vlos_iscat_1  
14 sd_hok_vlos_gscat_1  
15 sd_hok_vlos_bothscat_1 sd_hok_vlos_iscat_1  
sd_hok_vlos_gscat_1  
16 sd_hok_vnorth_iscat_1  
17 sd_hok_vnorth_gscat_1  
18 sd_hok_vnorth_bothscat_1 sd_hok_vnorth_iscat_1  
sd_hok_vnorth_gscat_1  
19 sd_hok_veast_iscat_1  
20 sd_hok_veast_gscat_1  
21 sd_hok_veast_bothscat_1 sd_hok_veast_iscat_1  
sd_hok_veast_gscat_1  
22 sd_hok_position_tbl_1  
23 sd_hok_positioncnt_tbl_1
```

Rules of the roadはデータ使用上の注意ですので、必ず内容を熟読してからデータをお使い下さい

Range-Time-Intensity (RTI) プロット の作成

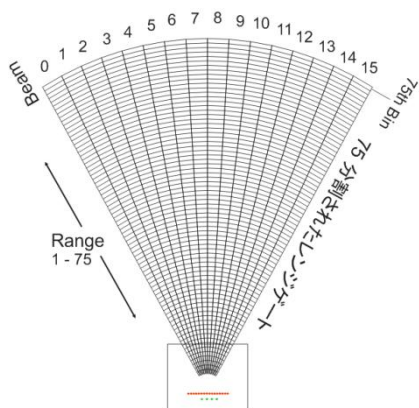
Range-Time-Intensity (RTI)プロット

- ▶ Backscatter power, Line-of-sight Doppler velocity, Spectral width の時間変動をプロット

```
THEMIS> tplot, 'sd_hok_pwr_1'
...
TPLOT(398): 2 sd_hok_pwr_1
...

THEMIS>

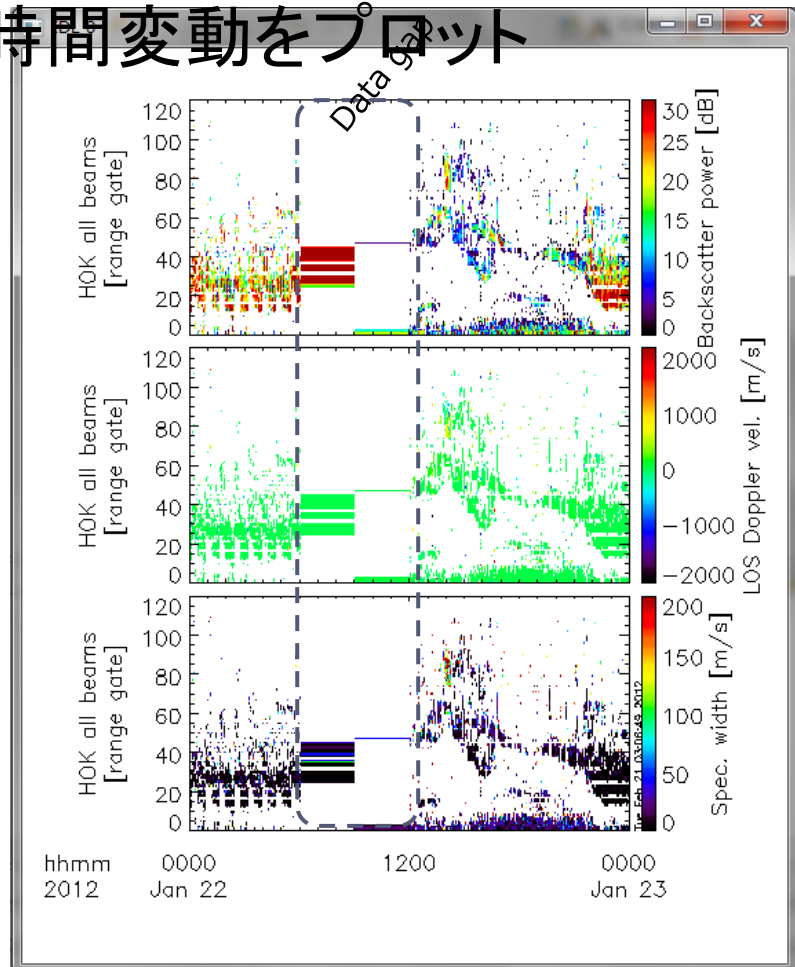
THEMIS> tplot, ['sd_hok_pwr_1',
'sd_hok_vlos_1', 'sd_hok_spec_width_1']
TPLOT(398): 2 sd_hok_pwr_1
TPLOT(398): 6 sd_hok_vlos_1
TPLOT(398): 4 sd_hok_spec_width_1
```



Beam0, Beam1, Beam2, ..., Beam15 の順に 1ビーム約3秒ずつ観測していく

右図ではビーム順に時間方向に並べてプロットしている

この時北海道レーダーは110レンジゲートモードだった

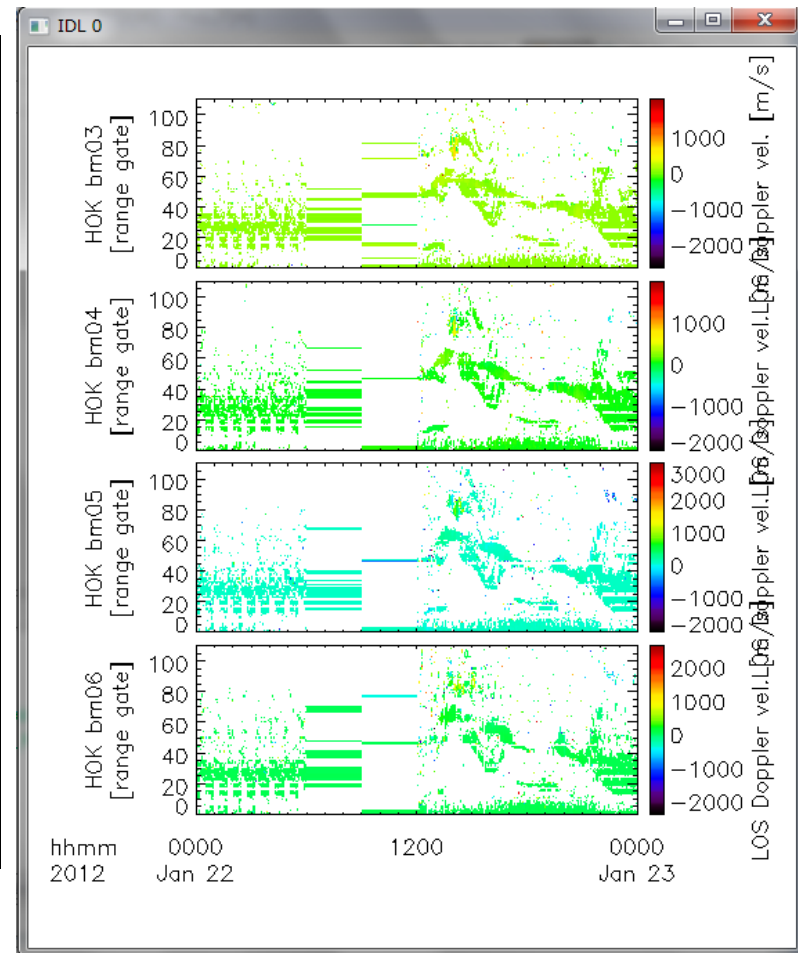


Range-Time-Intensity (RTI)プロット

▶ Beamに分割して複数beamを並べる

```
THEMIS> splitbeam, 'sd_hok_vlos_1'  
STORE_DATA(155): Altering tplot variable: 24 sd_hok_vlos_1_azim00  
STORE_DATA(155): Altering tplot variable: 25 sd_hok_vlos_1_azim01  
STORE_DATA(155): Altering tplot variable: 26 sd_hok_vlos_1_azim02  
STORE_DATA(155): Altering tplot variable: 27 sd_hok_vlos_1_azim03  
STORE_DATA(155): Altering tplot variable: 28 sd_hok_vlos_1_azim04  
STORE_DATA(155): Altering tplot variable: 29 sd_hok_vlos_1_azim05  
STORE_DATA(155): Altering tplot variable: 30 sd_hok_vlos_1_azim06  
STORE_DATA(155): Altering tplot variable: 31 sd_hok_vlos_1_azim07  
STORE_DATA(155): Altering tplot variable: 32 sd_hok_vlos_1_azim08  
STORE_DATA(155): Altering tplot variable: 33 sd_hok_vlos_1_azim09  
STORE_DATA(155): Altering tplot variable: 34 sd_hok_vlos_1_azim10  
STORE_DATA(155): Altering tplot variable: 35 sd_hok_vlos_1_azim11  
STORE_DATA(155): Altering tplot variable: 36 sd_hok_vlos_1_azim12  
STORE_DATA(155): Altering tplot variable: 37 sd_hok_vlos_1_azim13  
STORE_DATA(155): Altering tplot variable: 38 sd_hok_vlos_1_azim14  
STORE_DATA(155): Altering tplot variable: 39 sd_hok_vlos_1_azim15  
THEMIS>  
THEMIS> tplot,  
['sd_hok_vlos_1_azim03','sd_hok_vlos_1_azim04','sd_hok_vlos_1_azim05','sd  
_hok_vlos_1_azim06'] ;長いけど tplot から ..azim06] まで1行で書く  
TAPLOT(398): 25 sd_hok_vlos_1_azim03  
TAPLOT(398): 27 sd_hok_vlos_1_azim04  
TAPLOT(398): 29 sd_hok_vlos_1_azim05  
TAPLOT(398): 31 sd_hok_vlos_1_azim06  
THEMIS>
```

Beam03, 04, 05, 06 を並べてみた。しかしカラーバーのスケールがばらばら…



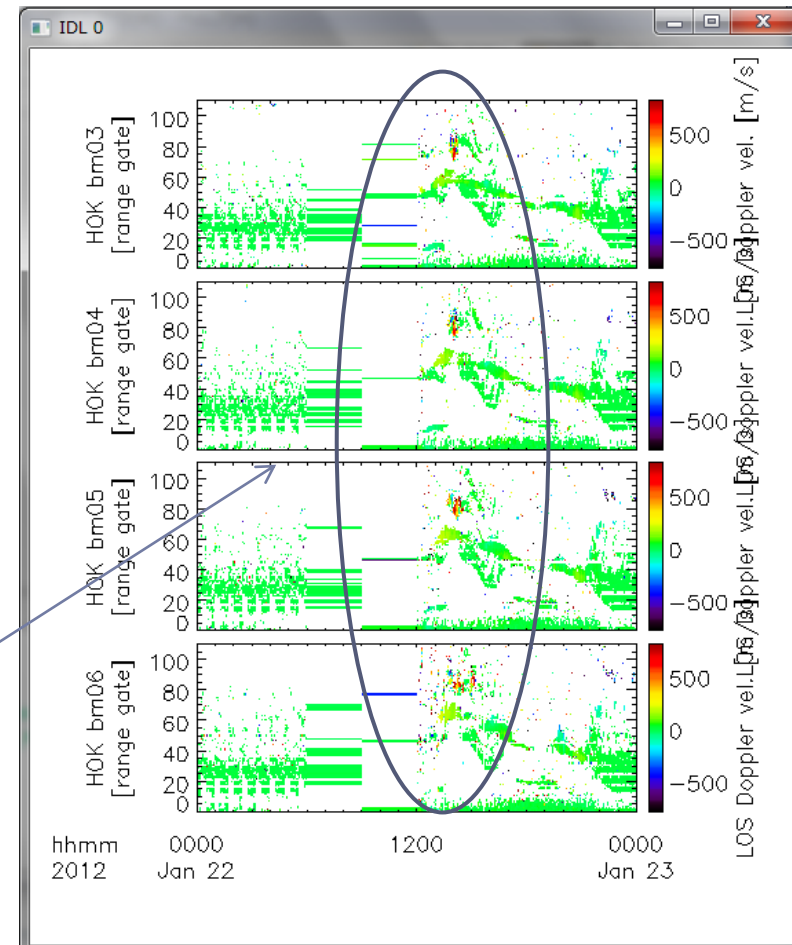
Range-Time-Intensity (RTI)プロット

▶ カラーバーのスケールを変える

```
THEMIS> zlim, 'sd_hok_vlos_1_azim??', -800, 800,  
  
THEMIS>  
  
THEMIS> tplot ;tplotのみだと前回のプロットしたtplot変数を再プロットする  
TPLOT(398): 25 sd_hok_vlos_1_azim03  
TPLOT(398): 27 sd_hok_vlos_1_azim04  
TPLOT(398): 29 sd_hok_vlos_1_azim05  
TPLOT(398): 31 sd_hok_vlos_1_azim06  
THEMIS>
```

zlim, 'sd_hok_vlos_1_azim??' のようにワイルドカードを使うことで、…azim00 ~ …azim15 全てについて、カラーバーの上下限を設定できる。

カラーバーの上限下限を-800, +800 [m/s] にしたら
Doppler速度の変動が浮き出てきた！



Range-Time-Intensity (RTI)プロット

▶ プロットの時間幅を変える

```
THEMIS> tlimit
```

```
;ウィンドウに十字スケールが出てくるので、プロットしたい時間帯の  
;最初の辺に合わせて左クリック1回  
;最後の辺りに合わせて左クリック1回、 で時間幅を選択できる  
;右の例では 13:00, 16:00 辺りをクリックして得られたプロット
```

```
THEMIS> tlimit, /last ;時間幅を前回プロットした時の時間幅に戻す
```

```
THEMIS> tlimit, /full ;時間幅をtimespan で指定してある幅に戻す
```

```
THEMIS> tlimit, '2012-01-22/13:00', '2012-01-22/16:00'  
;時間幅を直書きして指定することもできる
```

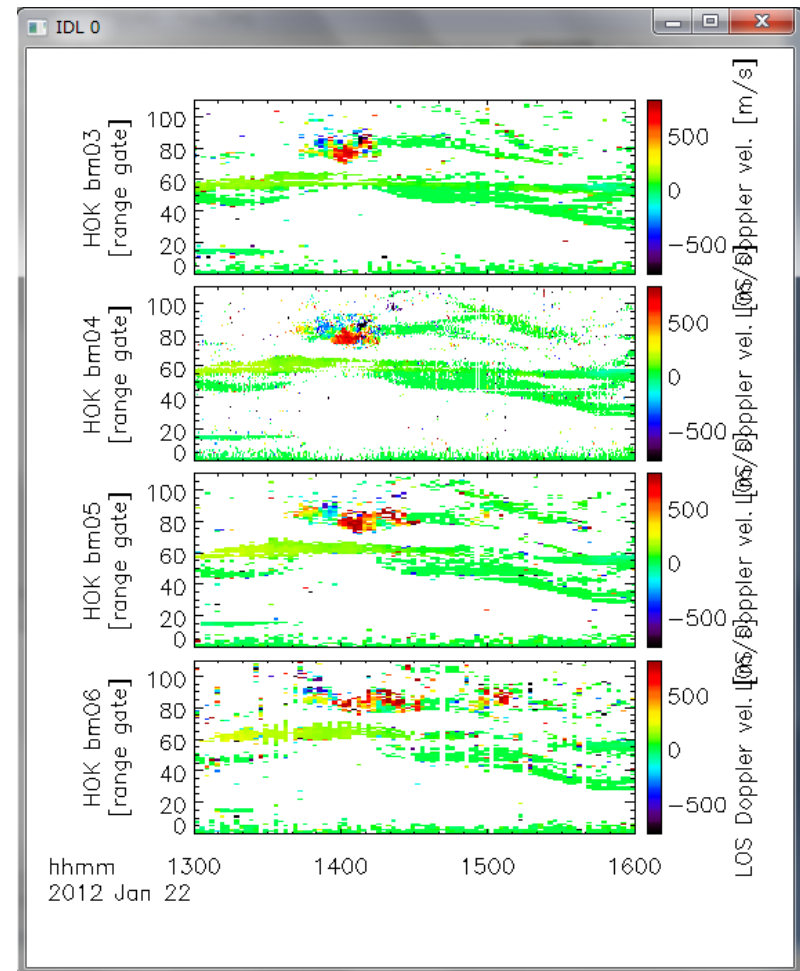
tlimit + マウスクリック

tlimit, /last

tlimit, /full

tlimit, [開始時刻, 終了時刻]

を駆使することで、プロットの時間幅を自由に設定できる



Range-Time-Intensity (RTI)プロット

▶ 電離圏エコー、地上エコーを区別してプロット

sd_hok_vlos_bothscat_1 は sd_hok_vlos_iscat_1 と sd_hok_vlos_gscat_1 の2つを指すマルチtplot変数なので、以下のコマンドで iscat, gscat 両方をビーム毎に分割することができる

```
THEMIS> splitbeam, 'sd_hok_vlos_bothscat_1'
```

; 電離圏エコーのみをプロット

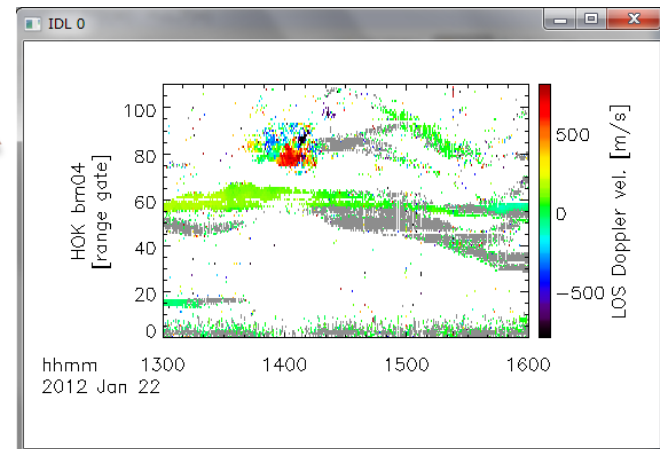
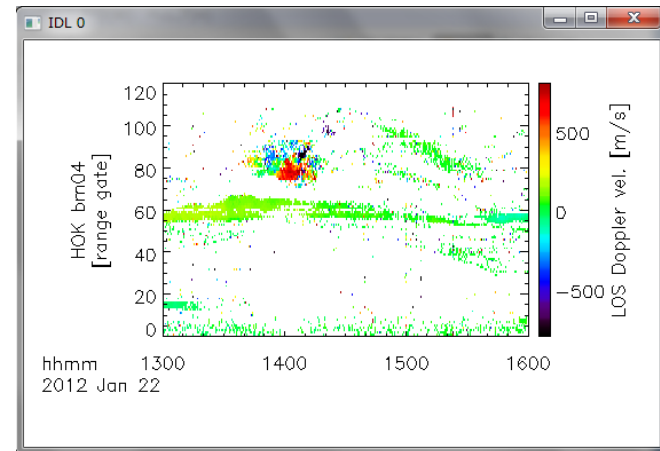
```
THEMIS> zlim, 'sd_hok_vlos_*scat*', -800,800
```

```
THEMIS> tplot, 'sd_hok_vlos_iscat_1_azim04'
```

; 電離圏エコーと地上エコー(灰色でマスク)をプロット

```
THEMIS> loadct_sd, 43 ; 灰色入りカラーテーブルをセット
```

```
THEMIS> tplot, 'sd_hok_vlos_bothscat_1_azim04'
```



Range-Time-Intensity (RTI)プロット

▶ 縦軸をRange gateから地理緯度、地磁気緯度にする

```
; 縦軸を地理緯度に変換
```

```
THEMIS> set_coords, ['sd_hok_vlos_bothscat_1_azim04'], 'glat'
```

```
STORE_DATA(155): Altering tplot variable: 43 sd_hok_vlos_iscat_1_azim04
```

```
sd_hok_vlos_iscat_1_azim04: vertical axis --> Geographical lat.
```

```
STORE_DATA(155): Altering tplot variable: 59 sd_hok_vlos_gscat_1_azim04
```

```
sd_hok_vlos_gscat_1_azim04: vertical axis --> Geographical lat.
```

```
; 再プロット (縦軸がrange gateから地理緯度になる)
```

```
THEMIS> tplot, ['sd_hok_vlos_bothscat_1_azim04']
```

```
; 縦軸をAACGM緯度に変換
```

```
THEMIS> set_coords, ['sd_hok_vlos_bothscat_1_azim04'], 'mlat'
```

```
STORE_DATA(155): Altering tplot variable: 43 sd_hok_vlos_iscat_1_azim04
```

```
sd_hok_vlos_iscat_1_azim04: vertical axis --> AACGM lat.
```

```
STORE_DATA(155): Altering tplot variable: 59 sd_hok_vlos_gscat_1_azim04
```

```
sd_hok_vlos_gscat_1_azim04: vertical axis --> AACGM lat.
```

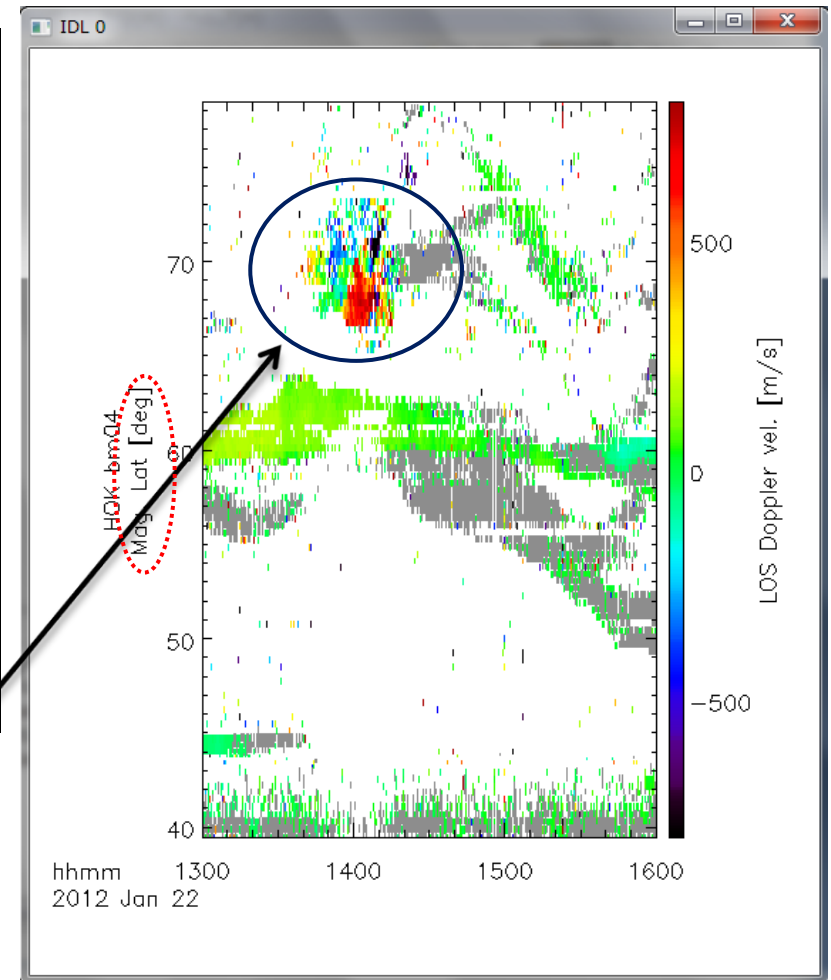
```
THEMIS> tplot, ['sd_hok_vlos_bothscat_1_azim04']
```

Doppler速度を変動している部分は地理緯度70-75° 前後
だがAACGM緯度だと66-73° くらいになる

AACGM: Altitude-Adjusted Corrected GeoMagnetic
coordinates

磁力線マッピングを考慮した地磁気座標の1つ

(http://superdarn.jhuapl.edu/tutorial/Baker_AACGM.pdf)



Range-Time-Intensity (RTI)プロット

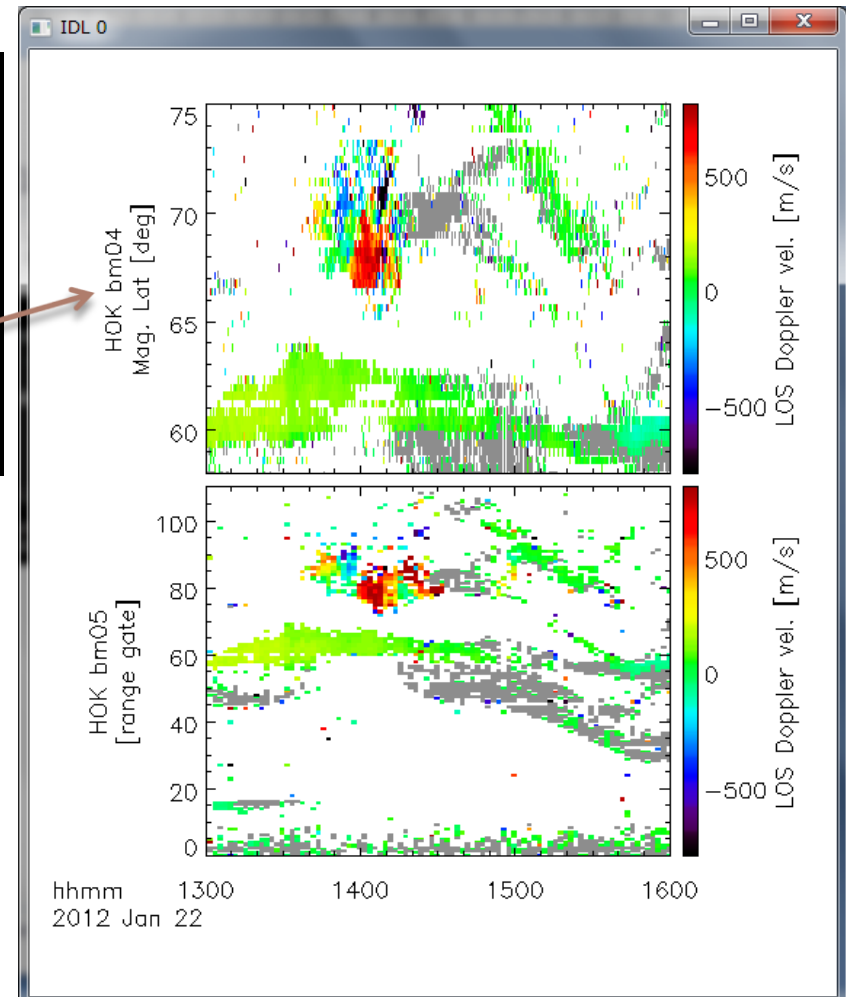
▶ 縦軸の範囲を変える

```
THEMIS> ylim, 'sd_hok_vlos_bothscat_1_azim04', 58.0, 75.0  
;縦軸の範囲を 58° -- 75° にする
```

```
THEMIS> tplot, 'sd_hok_vlos_bothscat_1_azim04' ;再プロット
```

```
THEMIS> tplot,  
['sd_hok_vlos_bothscat_1_azim04','sd_hok_vlos_bothscat_1_azim05']
```

; azim04は縦軸のパラメータ・範囲が変わっているが、azim05はそのまま。
; set_coords, zlim, ylim はそれぞれのtplot変数に対して設定する必要がある。



Range-Time-Intensity (RTI)プロット

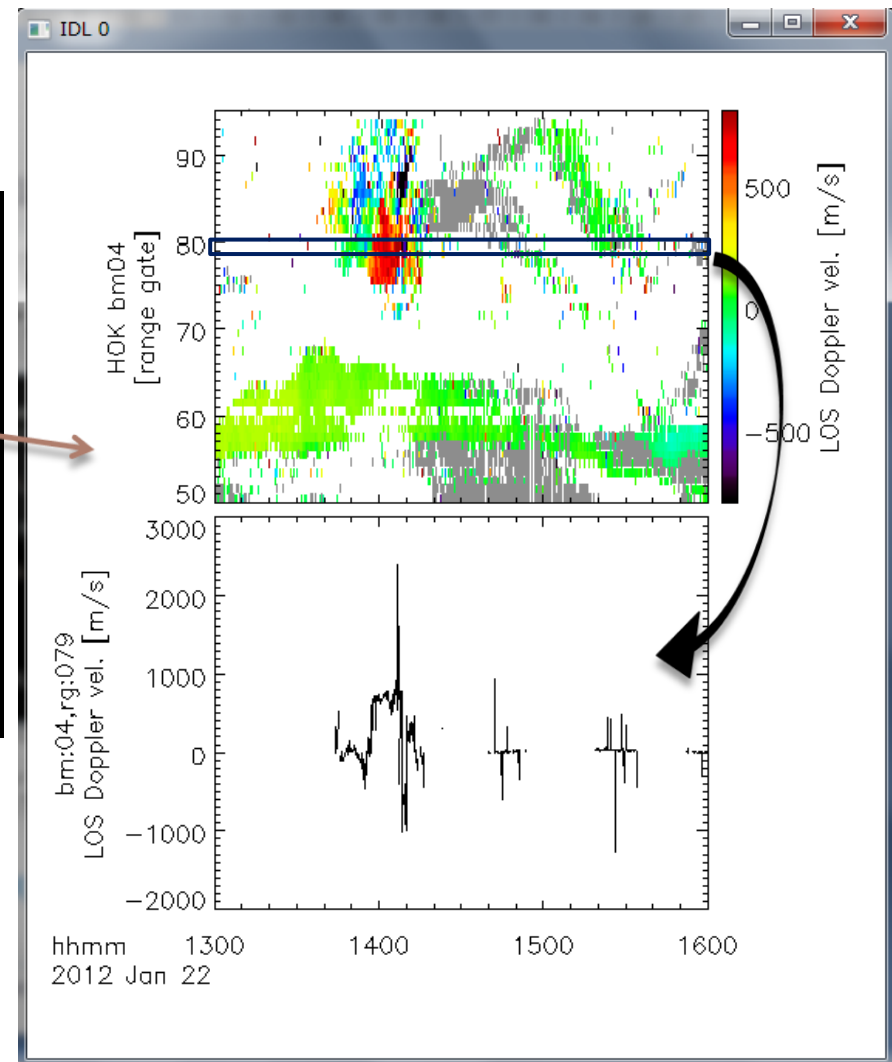
- ▶ 1つのpixelの時間変化を線プロットで描画する

```
THEMIS> set_coords, 'sd_hok_vlos_bothscat_1_azim04', 'gate'  
THEMIS> ylim, 'sd_hok_vlos_bothscat_1_azim04', 50, 95  
  
THEMIS> get_fixed_pixel_graph, 'sd_hok_vlos_1', $  
    beam=4, range_gate=79, /newvn  
                                ; beam4, range gate 79のpixel  
を選択  
  
THEMIS> tplot, ['sd_hok_vlos_bothscat_1_azim04', $  
    'sd_hok_vlos_1_bm04rg079' ]  
  
; get_fixed_pixel_graph  
; beam4, range gate 79 のpixel の値を取り出して、新しいtplot変数  
; に格納する
```

速度の値をグラフで確認することができる

```
THEMIS> get_data, 'sd_hok_vlos_1_bm04rg079',  
data=data
```

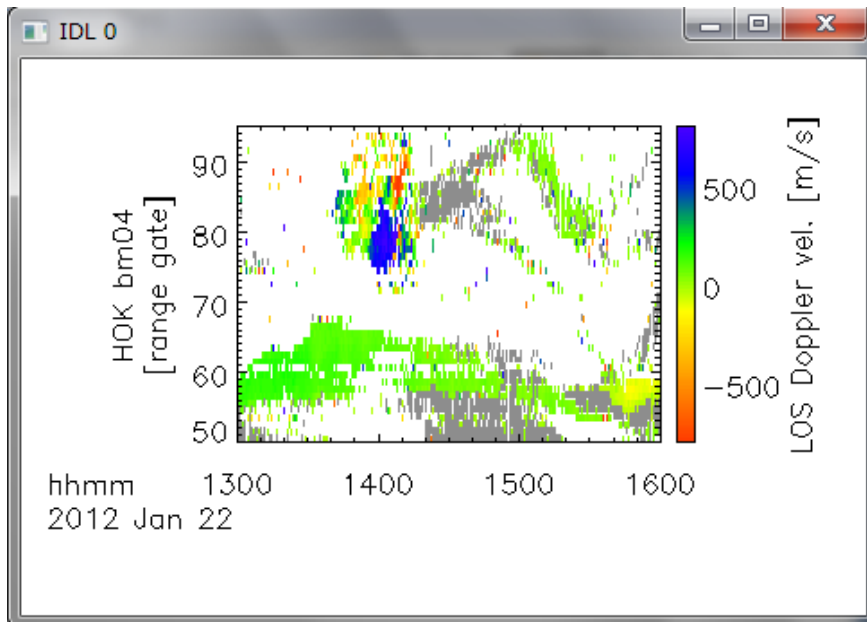
で、値を通常のIDL変数(実際には構造体)に代入することも可能



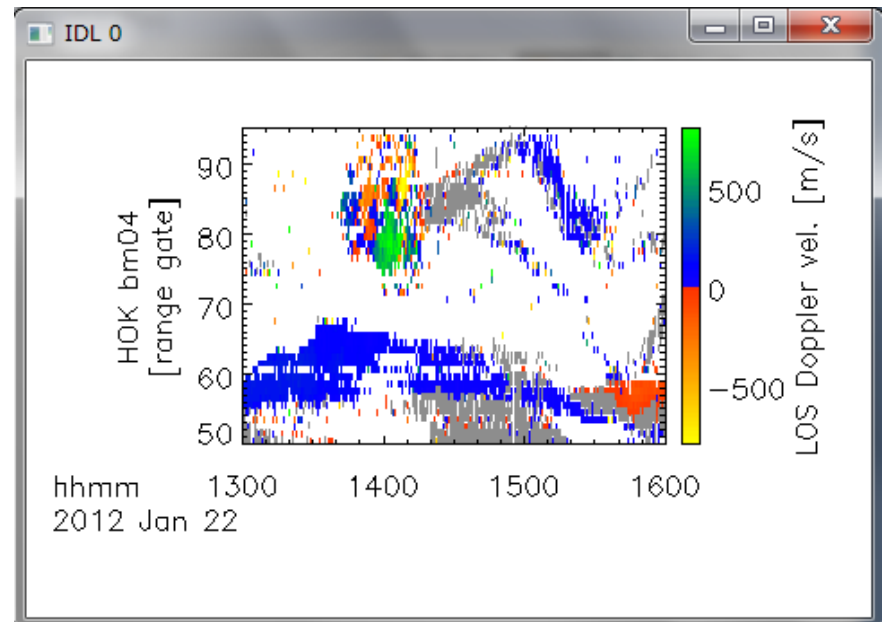
Range-Time-Intensity (RTI)プロット

▶ カラーテーブルを変えてプロット

```
THEMIS> loadct_sd, 44  
THEMIS>  
tplot,'sd_hok_vlos_bothscat_1_azim04'
```



```
THEMIS> loadct_sd, 45  
THEMIS>  
tplot,'sd_hok_vlos_bothscat_1_azim04'
```



正の値を寒色系、負の値を暖色系で描画。
SDプロットでは慣習的にこっちを使う。

ゼロのところを寒色系-暖色系を切り替える
→ 値の正負をわかりやすくする

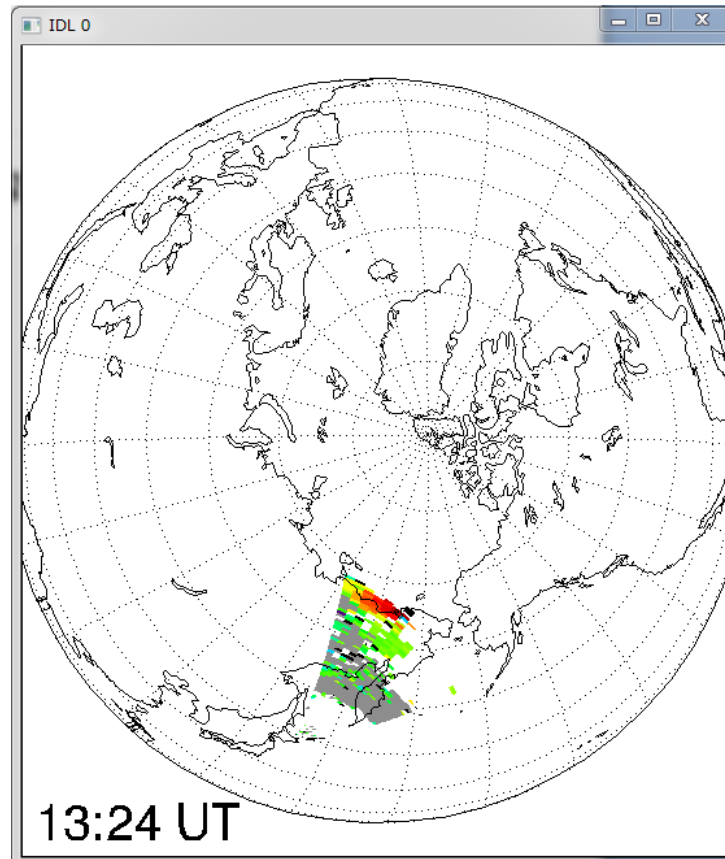
2次元プロットの作成

2次元プロット

- ▶ ある時刻の2次元スキャンのデータを、緯度・経度グリッド(+世界地図)上に描画する

観測値の2次元空間
分布がわかる

他の観測データを重ね
描きすれば位置関係を
調べることができる



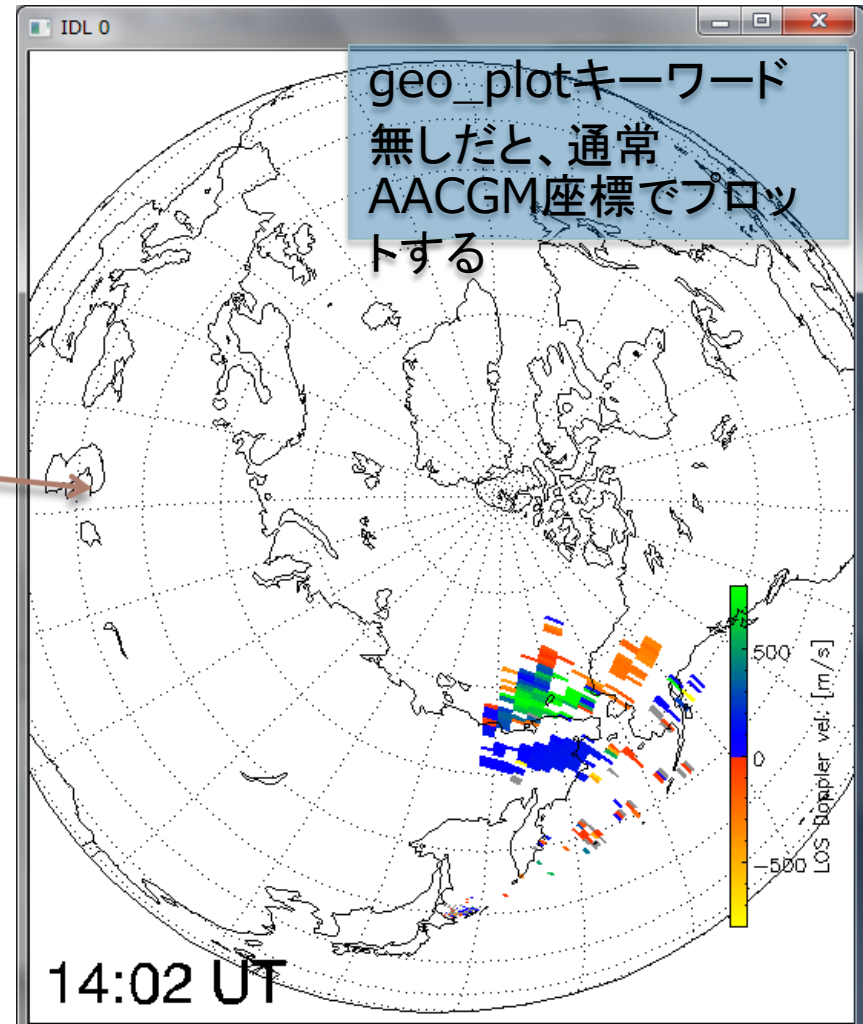
ある1つの時刻のデータ
しかプロットすることが
できない
(異なる時刻の複数のプ
ロットを作る必要)

2次元プロット

1発コマンド, plot_map_sdffitを使う

```
;環境をセットアップする  
THEMIS> sd_init  
  
;プロットする時刻を指定する  
THEMIS> sd_time, 1402  
  
;指定時刻のLOS velocityデータを描画する  
THEMIS> plot_map_sdffit,  
'sd_hok_vlos_bothscat_1'  
  
; coast キーワードをセットすると世界地図を重ねて描く  
THEMIS> plot_map_sdffit,  
'sd_hok_vlos_bothscat_1', /coast  
  
; geo_plotキーワードで"地理"座標上にプロット  
THEMIS> plot_map_sdffit,  
'sd_hok_vlos_bothscat_1', /coast, $  
/geo_plot
```

sd_init は1回実行すればよい。IDLを再起動して同じことをする場合は、もう1度 sd_init を実行する。



2次元プロット

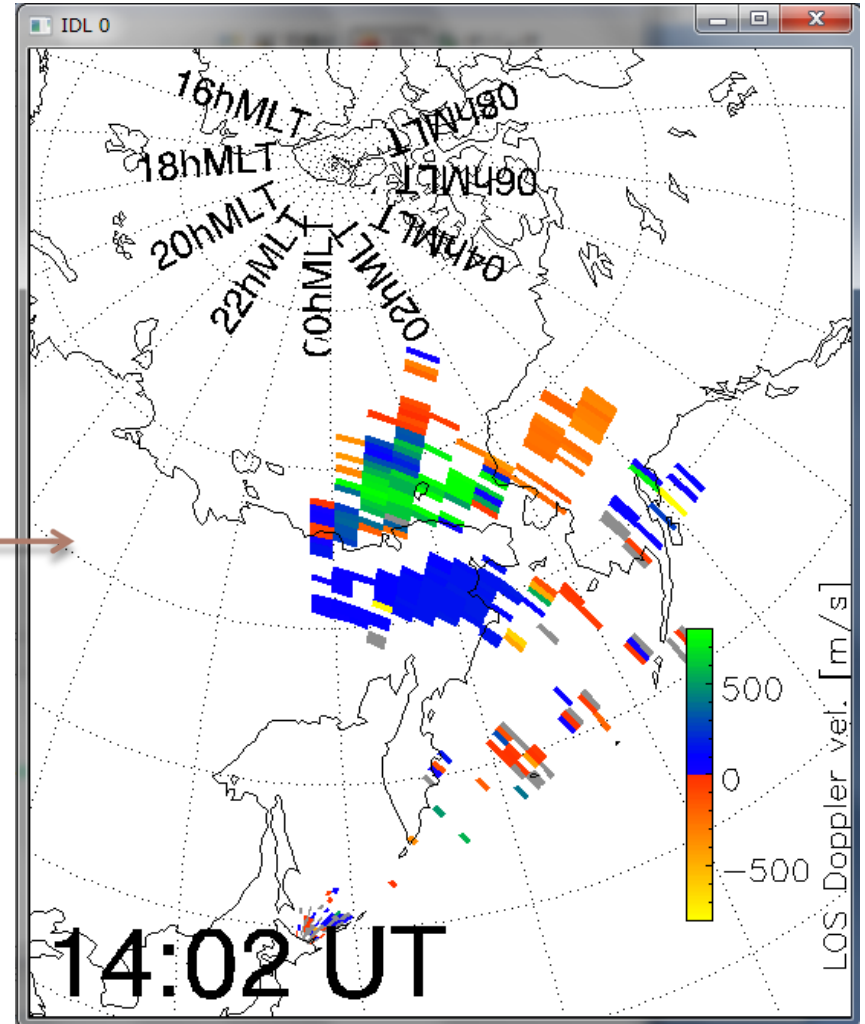
1発コマンド, plot_map_sdfitを使う

```
;clip キーワードをセットするとズームイン。  
;レーダー視野を外れることがあるので center_glat,  
;center_glon キーワードで描画中心の地理緯度経度を指定  
;する。
```

```
THEMIS> plot_map_sdfit, 'sd_hok_vlos_bothscat_1',  
$  
    /coast, /clip, $  
    center_glat=70, center_glon=180  
    ; $(ダラー)を付けると1行を分割できる
```

;拡大するとスケールがはみ出ることが多いので、手動で位置を調整するとよい。colorscalepos キーワードで指定する。

```
THEMIS> plot_map_sdfit, 'sd_hok_vlos_bothscat_1',  
$  
    /coast, /clip, $  
    center_glat=70, center_glon=180, $  
    colorscalepos=[0.80, 0.1, 0.83, 0.40]
```



2次元プロット

マニュアル描画

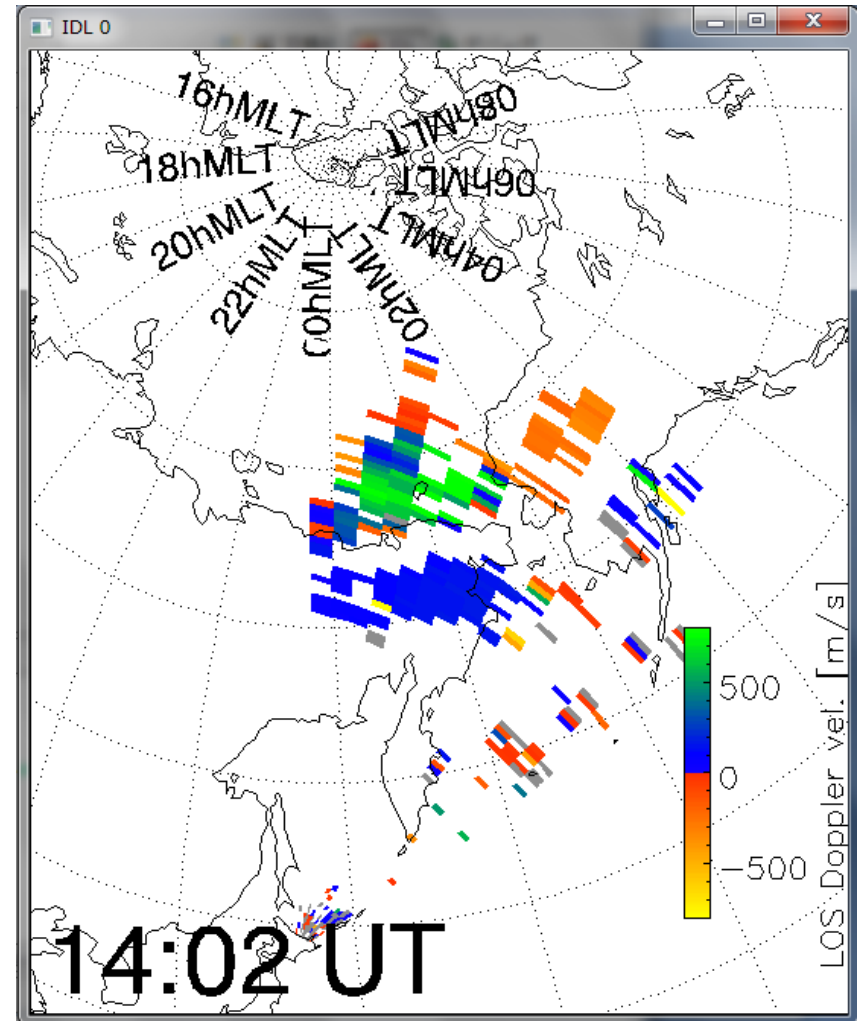
```
THEMIS> sd_init
THEMIS> sd_time, '2012-01-22/14:02'

;緯度経度描画モードをオンにして、グリッドを描画する。
; center_glat/glon で描画する際の中心位置を指定。
; clip でズーム
; mltlabel キーワードをセットするとMLTのラベルを描く
; erase キーワードをセットすると一度ウィンドウ内を消去
THEMIS> sd_map_set,
center_glat=70,center_glon=180, $
        /clip, /mltlabel, /erase

;指定時刻のLOS velocityデータを重ね描きする
THEMIS> overlay_map_sdfit,
'sd_hok_vlos_bothscat_1', $
        colorscalepos=[0.80, 0.1, 0.83, 0.40]

;世界地図を重ね描きする
THEMIS> overlay_map_coast
```

実は前項の plot_map_sdfit は 内部で
•sd_map_set
•overlay_map_sdfit
•overlay_map_coast (/coastの場合)
を順に実行している。



2次元プロット

応用編 – 複数時刻プロット

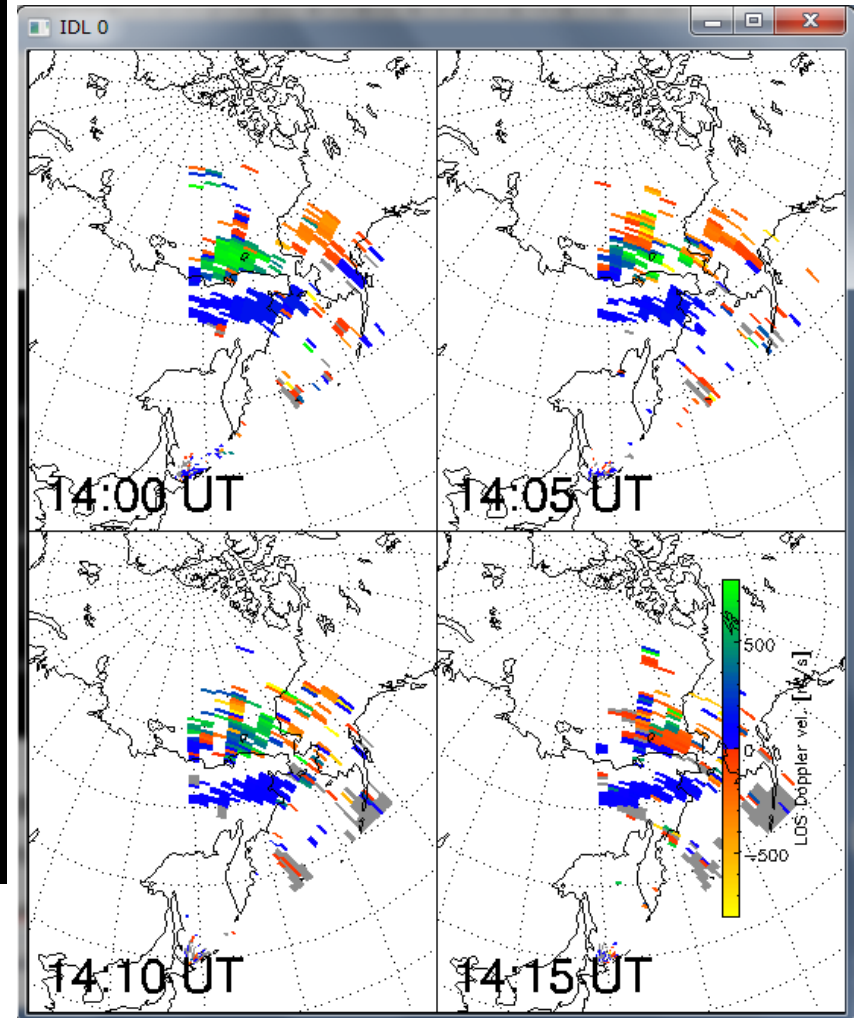
```
THEMIS> sd_time,1400
THEMIS> plot_map_sdfit,
'sd_hok_vlos_bothscat_1',/clip, /coast,
center_glat=70, center_glon=170,
position=[0.0,0.5,0.5,1.0] , /nocolorscale
```

```
THEMIS> sd_time, 1405
THEMIS> plot_map_sdfit, 'sd_hok_vlos_bothscat_1',
/clip, /coast, center_glat=70, center_glon=170,
position=[0.5,0.5,1.0,1.0], /noerase , /nocolorscale
```

```
THEMIS> sd_time, 1410
THEMIS> plot_map_sdfit,
'sd_hok_vlos_bothscat_1',/clip, /coast,
center_glat=70, center_glon=170,
position=[0.0,0.0,0.5,0.5], /noerase, /nocolorscale
```

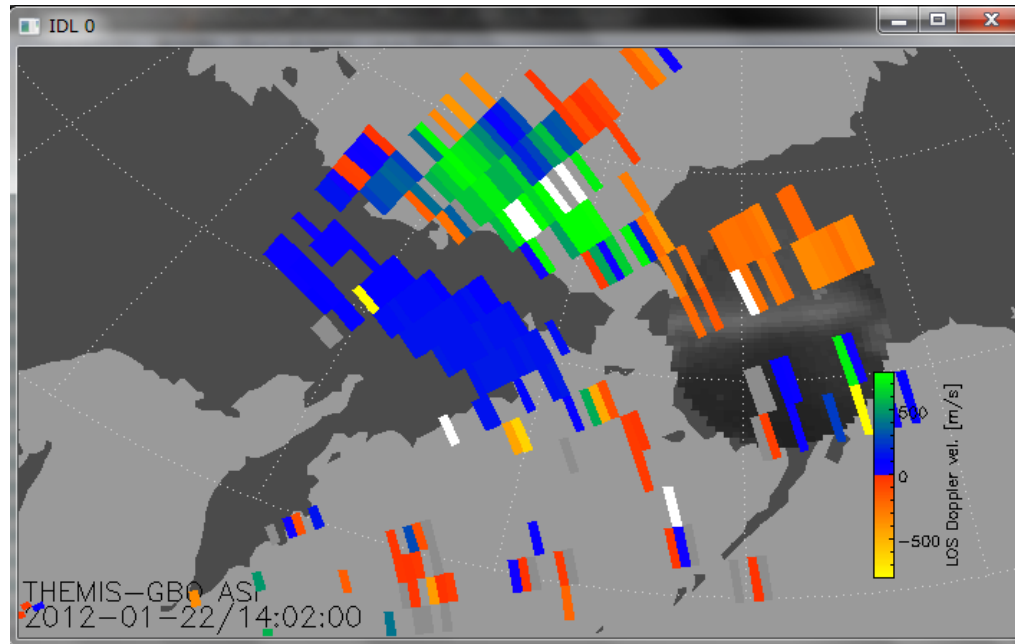
```
THEMIS> sd_time, 1415
THEMIS> plot_map_sdfit,
'sd_hok_vlos_bothscat_1',/clip, /coast,
center_glat=70, center_glon=170,
position=[0.5,0.0,1.0,0.5], /noerase
```

- positionキーワードにnormal座標でのプロットの位置を与える ([x0, y0, x1, y1])
- plot_map_sdfit はデフォルトで描画毎にウィンドウをクリアしてしまうので、2つ目以降は /noerase を付ける
- 4番目のみカラースケールを描画する (/nocolorscale無し)



2次元プロット

応用編 – THM/GBOと重ねる



```
THEMIS> timespan, '2012-01-22/14:00:00',1,/hour
```

```
THEMIS> thm_asi_stations,site,loc
```

```
THEMIS> ex_site = site[ where( site ne 'MCGR' ) ]
```

MCGR以外を描画から排除するためのリストを作成
(データロードに長時間かかってしまうため)

```
THEMIS> thm_asi_create_mosaic,'2012-01-22/14:02:00',/thumb,  
central_lat=65.,central_lon=180., exclude=ex_site,scal=2.e7
```

```
THEMIS> loadct_sd, 45
```

```
THEMIS> sd_time, 1402 & overlay_map_sdfit,  
'sd_hok_vlos_bothscat_1',/geo_plot,/notimelabel
```

補足資料

tplot変数名のnotation

sd_hok_vlos_1_azim03

レーダー名

変数名

Range gate (RG) suffix

0: 75 RGデータ
1: 110 RGデータ
2: 70 RGデータ
...

Beam番号
ほとんどのレーダーで beam0
が一番西端のbeam (例外有
り)

▶ 主な変数名と中身

pwr: エコー強度

vlos: Line-of-sight(LOS) ドップラー速度

spec_width: スペクトル幅

vnorth: LOSドップラー速度の地理緯度成分(北向き)

veast: LOSドップラー速度の地理経度成分(東向き)

(vlos|vnorth|veast)_iscat: 電離圏エコーのみのデータ

(vlos|vnorth|veast)_gscat: 地上エコーのみのデータ

(vlos|vnorth|veast)_bothscat: 電離圏・地上エコー両方のデータ

elev_angle: elevation angle値

echo_flag: 電離圏エコーか地上エコーかの判定フラグ

quality: データのqualityについての情報(0: good, 1以上: poor)

quality_flag: quality判定の内訳 (詳細は担当者へ)

position_tbl: 各pixelの四隅の緯度、経度値テーブル

positioncnt_tbl: 各pixelの中心の緯度、経度値テーブル

cpid: beam毎の観測モード

tfreq: beam毎の周波数

noise: beam毎のノイズレベル