Tutorial for the sdhdfProc software package

George Hobbs

This document can be edited at
https://docs.google.com/document/d/1jA5lOn5MlltnOrchrggwLSnBtKRPJVUL2F984NmVRKk/edit?usp=sharing

Introduction

All the data files described here are available from:

/DATA/BRAHE_2/hob044/sdhdf_tutorial/

In each sub-directory the list of commands for the specific use-case is available in a text file called "commands".

Note that there are various ways to use the sdhdfProc software:

- Obtain the software from https://bitbucket.csiro.au/scm/cpda/sdhdf_tools.git and use the makefile to install. In this case the software will be called as described in this tutorial - e.g., to plot a spectrum use `sdhdf_plotSpectrum`.
- Use the pre-installed software on the CSIRO machines. Type:
  - `sdhdf` to get usage details.
  - Run `sdhdf plotSpectrum` (note space instead of the underscore) to run the sdhdf_plotSpectrum code.
- To use George's version of the code (which may be changed at any time), use e.g., /u/hob044/software/new_c/gitRepos/sdhdf_tools/c/sdhdfProc/src//sdhdf_plotSpectrum to plot the spectrum. Note that this code has been compiled under "buster"

In all cases the $SDHDF environment variable needs to be set an pointing to the run time directory (e.g.,

```
setenv SDHDF
/u/hob044/software/new_c/gitRepos/sdhdf_tools/c/sdhdfProc/runtime/
```

)

An easy way to inspect an sdhdf file is to use “silx view”

```
$ silx view <filename.hdf>
```

The primary sdhdfProc documentation is available from
https://docs.google.com/document/d/1PhVhzOmVZ7iqA97LxRHF0gP5OHUI4Mw0Hp3KVjU9yIA/edit?usp=sharing

Use case: Are there any bright spectral lines in my data?

We make use of observations of a region in the Orion nebula carried out as P595 by Macquarie University students (supervised by Jo Dawson). The files are in /DATA/BRAHE_2/hob044/sdhdf_tutorial/orion. We can first identify the ON and OFF source pointings using `sdhdf_describe`:

```
$ sdhdf_describe *.hdf
------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------
File                         Beam  UTC                 SDHDF PID    Sched_ID Source            Tel        Observer   RCVR
Bands  POL_TYPE   CAL   M_Time RA_s        DEC_s
------------------------------------------------------------------------------------------------------------------------------------------------------------
------------------------------
uwl_200915_194251.hdf            0     2020-09-15-19:42:51    1.9.2 P595   16673    Pink_Maser_ON       Parkes     cap056
UWL     26     AABBCRCI   ON    597.7  05:03:21.20  +02:03:04.60
uwl_200915_200436.hdf            0     2020-09-15-20:04:36    1.9.2 P595   16675    OFF_MAIN           Parkes     cap056
UWL     26     AABBCRCI   ON    597.7  05:35:05.73  -02:13:58.20
```

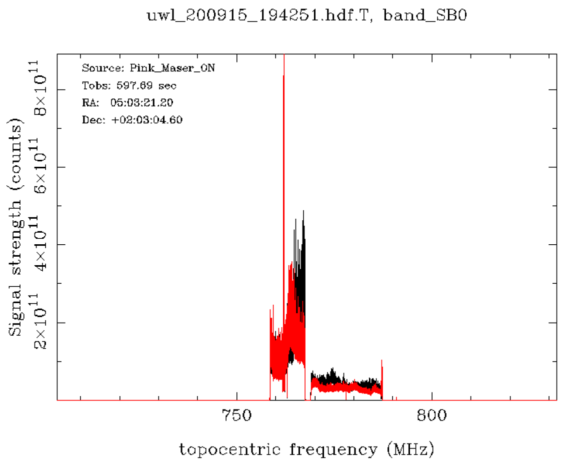
The first file is “on” source. The second file is “off” source. The files are large (>6GB in volume) and so let’s start by average the data in time:

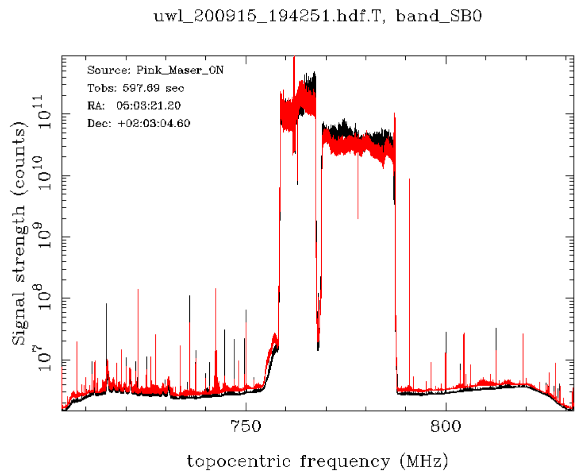
```
$ sdhdf_modify -T -e T *.hdf
```

This produces uwl_200915_194251.hdf.T and uwl_200915_212041.hdf.T. We can now plot the “on” spectrum:

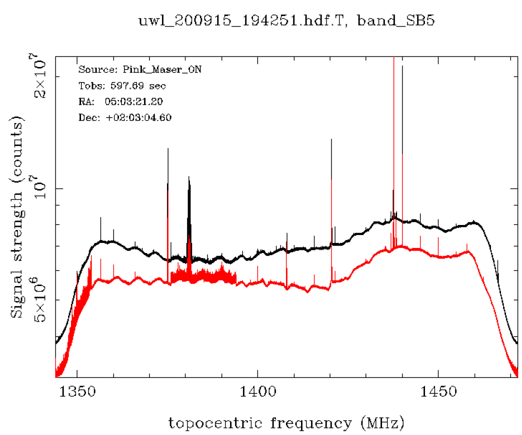
```
$ sdhdf_plotSpectrum -f uwl_200915_194251.hdf.T
```

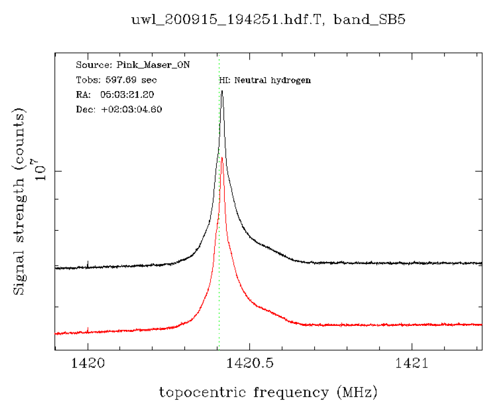
The output spectrum is as follows (left panel). Note that we’re viewing the first sub-band (which has a lot of radio interference) on a linear scale. If we press ‘l’ then we can switch to a logarithmic y-scale (see right panel). The black and red traces represent AA and BB polarisation channels respectively.



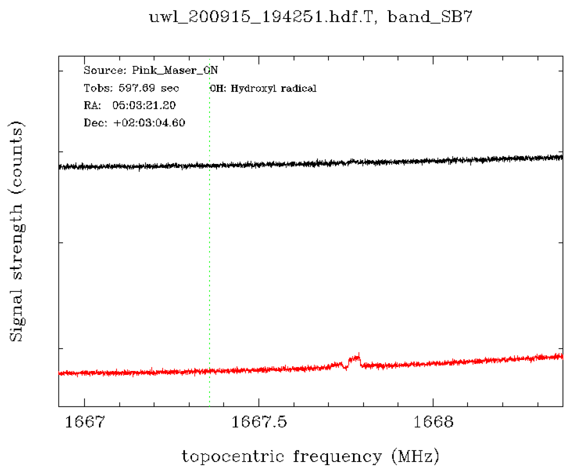


This sub-band is hard to work in, so let's use the '>' symbol to move through the sub-bands until we reach sub-band 5 (left panel below). We can zoom in around the HI line by moving the mouse cursor slightly to the left of 1420 MHz, pressing 'z', then moving the mouse to the right and clicking the mouse cursor (right panel). If we press 'm' we can plot the position of the rest frequency of the spectral line (as a vertical, green dashed line).





If we zoom-in around the OH line at 1667 MHz we see a signal that is of interest:

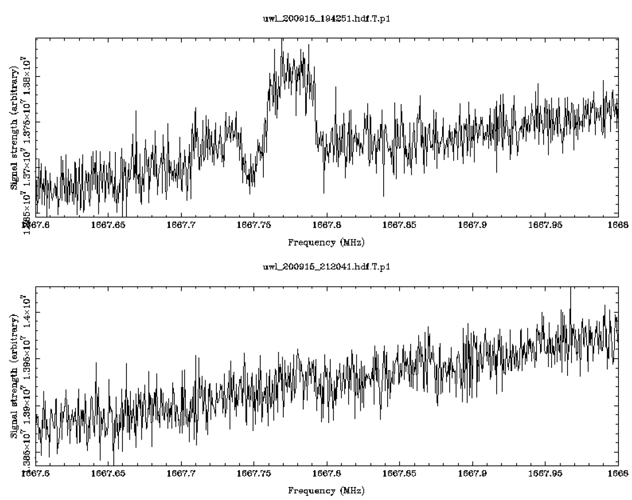


We can sum the polarisations using

$ sdhdf_modify -p1 -e p1 *.T

And then determine whether the signal is, or is not, in the off source pointing:

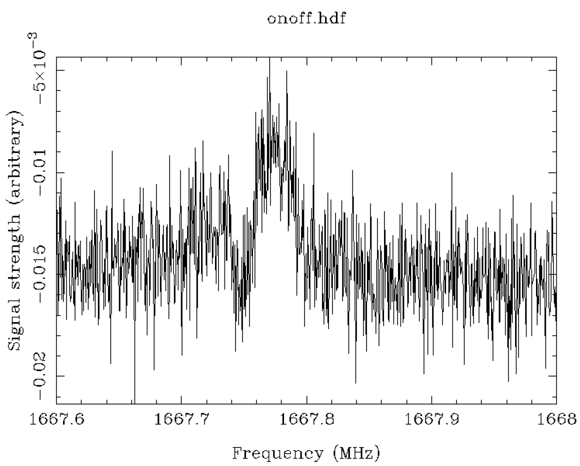
$ sdhdf_plotMultiSpec -sb 7 -f0 1667.6 -f1 1668 -ny 2 uwl_200915_194251.hdf.T.p1 uwl_200915_212041.hdf.T.p1

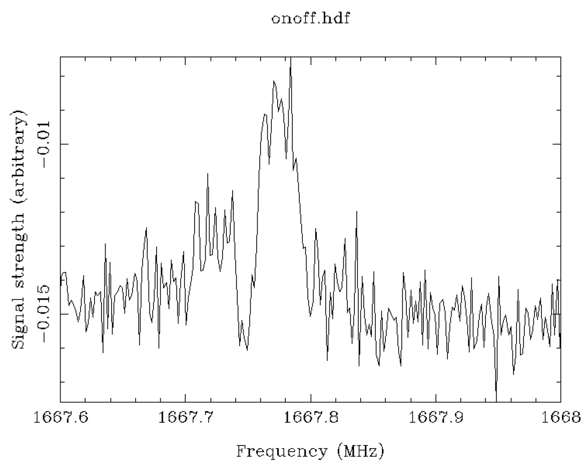


This seems convincing. We can form the quotient [(ON-OFF)/OFF] using:

sdhdf_onoff -on uwl_200915_194251.hdf.T.p1 -off uwl_200915_212041.hdf.T.p1 -o onoff.hdf
sdhdf_plotMultiSpec -sb 7 -f0 1667.6 -f1 1668 onoff.hdf

The resulting profile is shown in the left-hand plot below. We can include the command line argument -av 4 when using sdhdf_plotMultiSpec to average 4 adjacent frequency channels together (right-hand panel):





Note that in this simple example we have not carried out many standard processing tasks including calibration, converting to the local standard of rest (and velocity), RFI flagging etc.

Use case: Having a play with a scan across a source

Request: The user wishes to explore the SDHDF format and software using a scan across a bright continuum source. We will use a P737 scan across PKS 0407-658.

Note: This scan was a great circle scan. Dhagu allows different types of scans.

Let us first check the basic properties of this file:

```
$ sdhdf_describe  uwl_200720_024617.hdf
```

This gives basic information:

```
----------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------
File                                  Beam  UTC                    SDHDF PID     Sched_ID Source                Tel
Observer    RCVR     Bands   POL_TYPE   CAL   M_Time RA_s          DEC_s
----------------------------------------------------------------------------------------------------------------------
--------------------------------------------------------------------------------
uwl_200720_024617.hdf                 0     2020-07-20-02:46:17    1.9.1 P737    14710    0407scan_ROW5         Parkes
hob044      UWL      26      AABBCRCI   ON    116.9  04:03:50.81 -65:40:11.50
```

We can see that the source name is 0407scan_ROW5, the project code (P737), that 4 polarisation products were recorded (AABBCRCI) and that the calibrator was ON. The integration time was 116.9 seconds.

In order to find out the sub-bands stored in the file we use the -band option:

```
$ sdhdf_describe -band uwl_200720_024617.hdf
```

This gives:

```
-----------------------------------------------------------------------------
        #   BandID           F0        F1       NCHAN     TDUMP     NPOL NDUMP TOBS
-----------------------------------------------------------------------------
 [Band] 000 band_SB0          704.00    832.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 001 band_SB1          832.00    960.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 002 band_SB2          960.00   1088.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 003 band_SB3         1088.00   1216.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 004 band_SB4         1216.00   1344.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 005 band_SB5         1344.00   1472.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 006 band_SB6         1472.00   1600.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 007 band_SB7         1600.00   1728.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 008 band_SB8         1728.00   1856.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 009 band_SB9         1856.00   1984.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 010 band_SB10        1984.00   2112.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 011 band_SB11        2112.00   2240.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 012 band_SB12        2240.00   2368.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 013 band_SB13        2368.00   2496.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 014 band_SB14        2496.00   2624.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 015 band_SB15        2624.00   2752.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 016 band_SB16        2752.00   2880.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 017 band_SB17        2880.00   3008.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 018 band_SB18        3008.00   3136.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 019 band_SB19        3136.00   3264.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 020 band_SB20        3264.00   3392.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 021 band_SB21        3392.00   3520.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 022 band_SB22        3520.00   3648.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
 [Band] 023 band_SB23        3648.00   3776.00 32768     0.999     4     117    116.933  uwl_200720_024617.hdf
```

```
 [Band] 024 band_SB24        3776.00  3904.00 32768    0.999    4    117   116.933  uwl_200720_024617.hdf
 [Band] 025 band_SB25        3904.00  4032.00 32768    0.999    4    117   116.933  uwl_200720_024617.hdf
```

Showing that 26 sub-bands were recorded with the spectral dump time of 1 second.

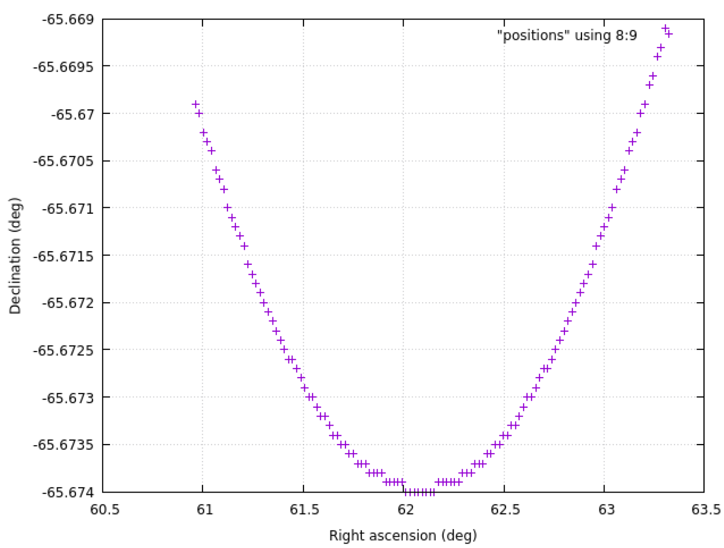
In particular for a scan we would like to know exactly where we were pointing at the centre of each spectral dump. We can obtain this with the -dump command line argument:

```
$ sdhdf_describe -dump uwl_200720_024617.hdf
```

```
-----------------------------------------------------------------------------------------------------------------------------------
         ElapsedTime MJD           UTC         AEST        RA          DEC          RADEG   DECDEG AZ       EL     GL      GB
-----------------------------------------------------------------------------------------------------------------------------------
 [SDUMP]        0.500 59050.1154800 02:46:17.50 12:46:18.40 04:03:50.81 -65:40:11.5  60.962 -65.670 209.390  39.053 278.829 -41.327
 [SDUMP]        1.499 59050.1154920 02:46:18.50 12:46:19.40 04:03:55.66 -65:40:12.0  60.982 -65.670 209.390  39.059 278.825 -41.319
 [SDUMP]        2.499 59050.1155030 02:46:19.50 12:46:20.40 04:04:00.51 -65:40:12.5  61.002 -65.670 209.390  39.067 278.820 -41.312
 [SDUMP]        3.498 59050.1155150 02:46:20.50 12:46:21.39 04:04:05.35 -65:40:13.0  61.022 -65.670 209.390  39.075 278.815 -41.304
 [SDUMP]        4.497 59050.1155270 02:46:21.50 12:46:22.39 04:04:10.20 -65:40:13.5  61.042 -65.670 209.390  39.080 278.810 -41.297
 [SDUMP]        5.497 59050.1155380 02:46:22.50 12:46:23.39 04:04:15.05 -65:40:14.0  61.063 -65.671 209.390  39.087 278.805 -41.289
…
```

We can store the positions in a text file using:

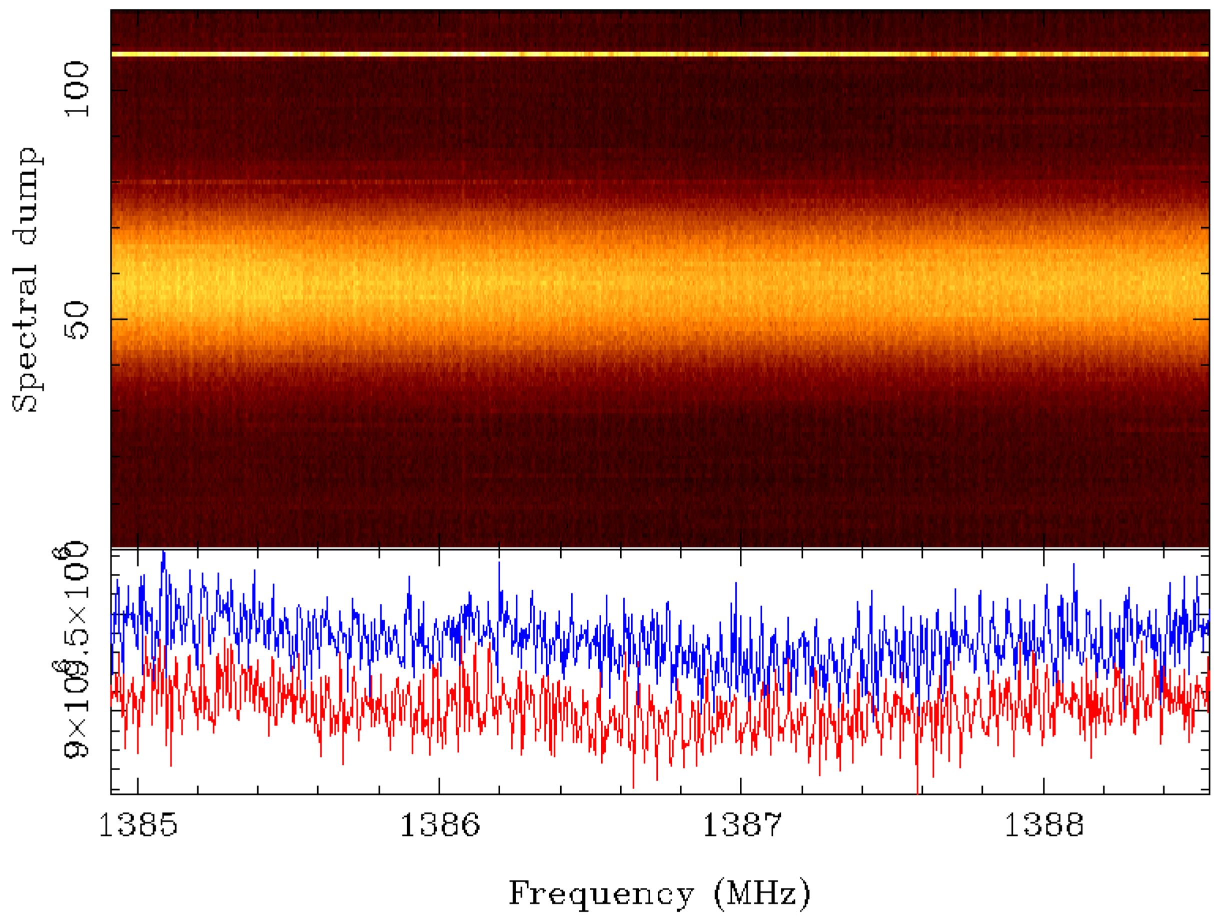
```
$ sdhdf_describe -dump uwl_200720_024617.hdf | grep SDUMP > positions
```



We can make a simple waterfall plot using sdhdf_waterfall:

```
$ sdhdf_waterfall -band 5 uwl_200720_024617.hdf
```

Here we selected a relatively clean band (band 5) and then used ‘x’ to zoom-in on the x-axis to a clean part of that sub-band. The result is:

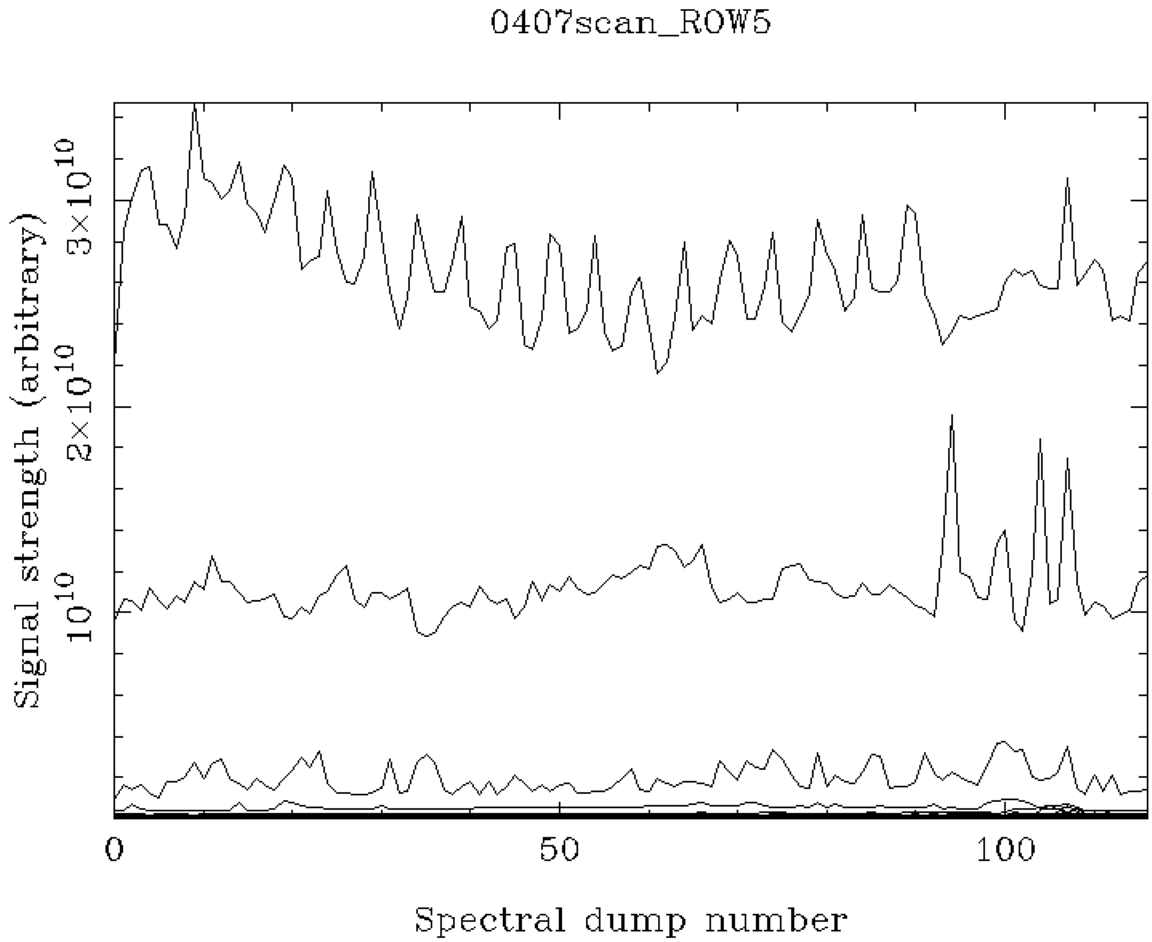


The top panel shows the uncalibrated signal strength as a function of frequency and time (colour scale). The bottom panel shows the min-hold and averaged spectrum averaged over these spectral dumps.

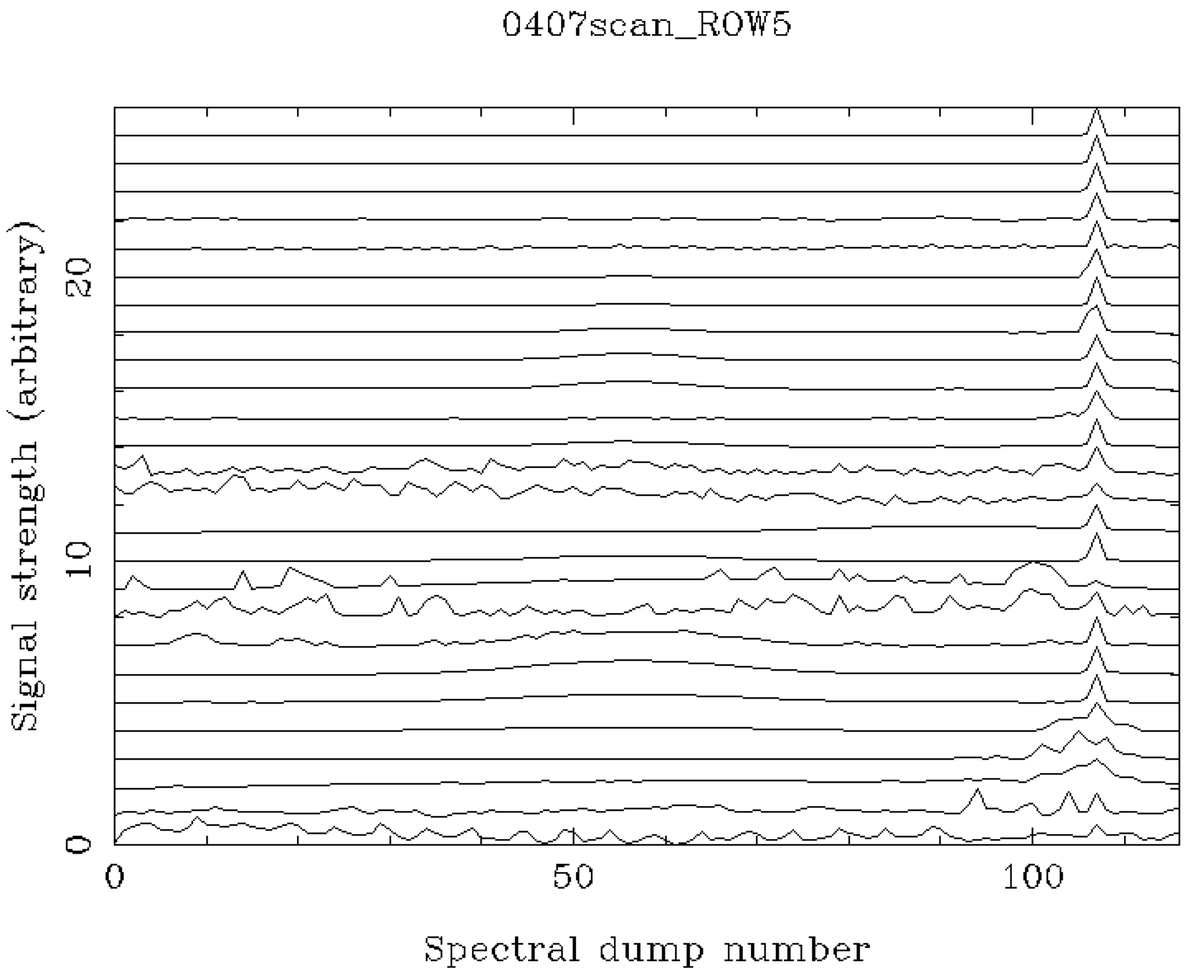
For a more detailed look at the scan we can use sdhdf_plotScan:

```
$ sdhdf_plotScan -f uwl_200720_024617.hdf
```

The output is shown below. The x-axis is the spectral dump number and the y-axis is the signal strength. The data in each sub-band is averaged in frequency and then plotted. Clearly some sub-bands are at a much higher level than other bands.



Pressing ‘n’ allows us to normalise each plot and then pressing ‘o’ offsets each band from each other:



The “triangular” feature on the right hand side is broad-band RFI that lasted for one spectral dump. The slight “mound” in the middle is the continuum source. Pressing ‘z’ allows you to use the mouse to zoom in on a region and press ‘u’ to unzoom back to the original plot. Pressing ‘x’ toggles the x-axis from spectral dump number to various other parameters (e.g., right ascension, azimuth etc.)

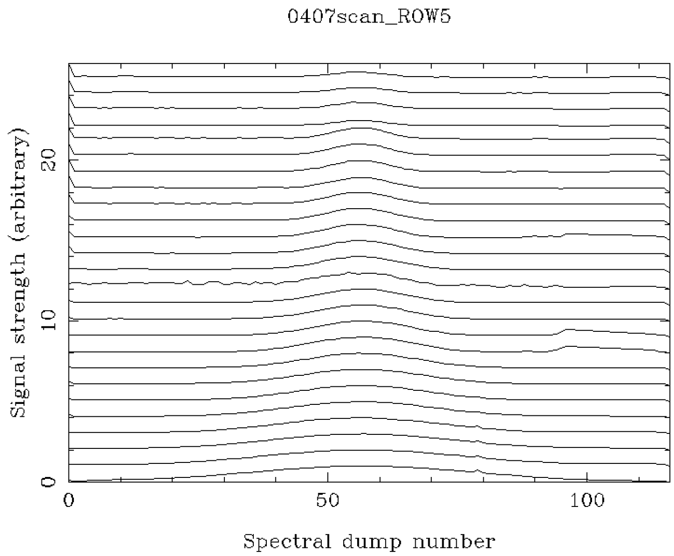
Clearly we need to carry out some RFI flagging as we are averaging in frequency across wide bands that contain RFI and we have at least one spectral dump affected by strong broadband interference. Note that this observation was unfortunately affected by: mobile handsets, aircraft DME, satellites and WiFi/Bluetooth!

We can do this with sdhdf_flag (see later tutorial, or the documentation, on the use of this package)

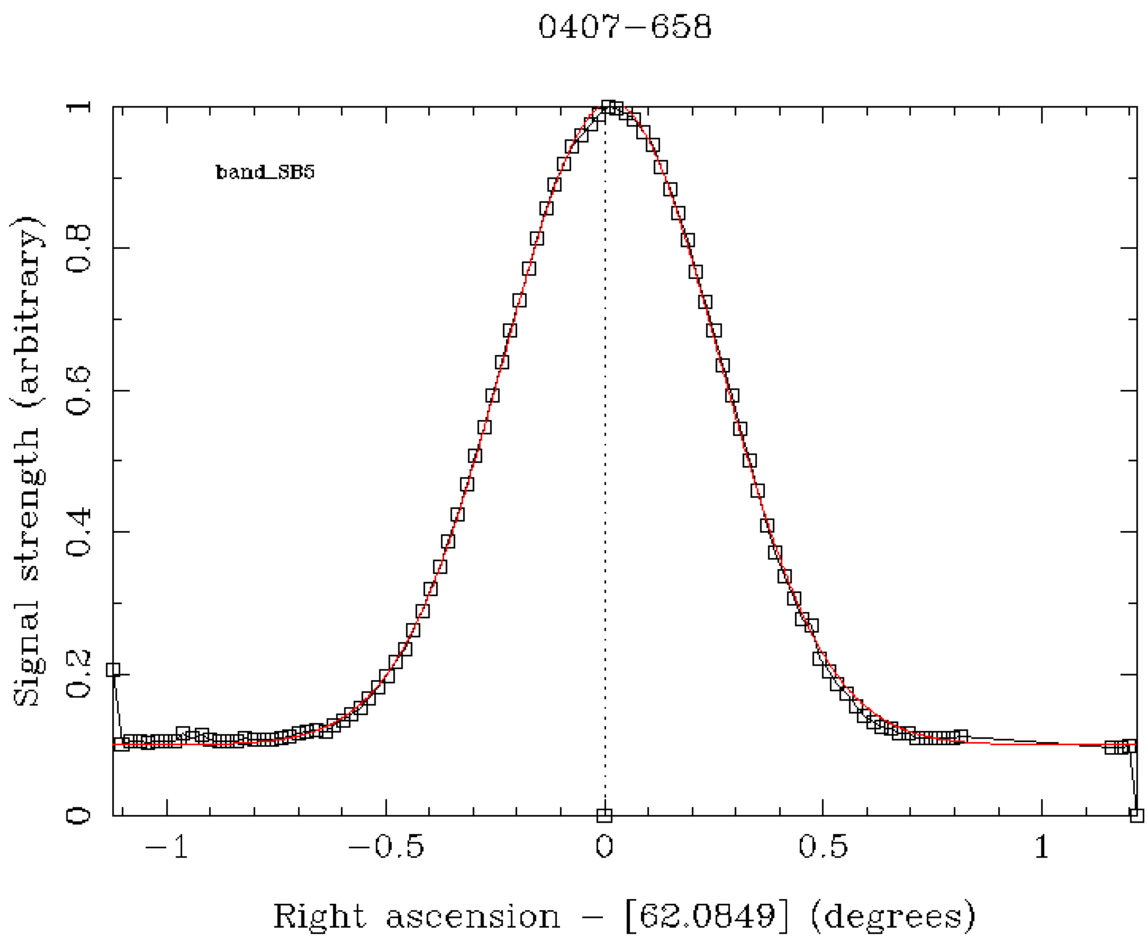
```
$ sdhdf_flag -f uwl_200720_024617.hdf
```

After flagging we can plot the scan again (note that we’re now using the -src option to specify the continuum source name):

```
$ sdhdf_plotScan -src 0407-658 -f uwl_200720_024617.hdf.flag
```



We can use the ‘+’ button to select a specific sub-band and then press ‘s’ to select that sub-band:



Note that we've used 'x' to select a right ascension offset for the plot and 'f' to fit a Gaussian. The parameters for the fitted Gaussian (for each sub-band) are listed on the terminal.