Producing ANTAB Files from FS Logs

Cormac Reynolds, JIVE
email: `reynolds@jive.nl`

October 29, 2004

Abstract

This document briefly describes the use of the antabfs suite of perl scripts to produce ANTAB formatted files for EVN stations from the FS logs. For further information read the usage instructions printed when the scripts are run.

1 Introduction

The following shorthand terms are used in the description:

tpical = BBC power levels when the cal diode is fired.

tpi' = BBC power levels without the cal diode, but measured close in time to the firing of the cal diode.

tpi = BBC power levels monitored during the scan.

tpdiff = $(\mathrm{tpical} - \mathrm{tpi}')$

tpgain = BBC AGC level. Used as a proxy for the continuous tpi monitoring on a VLBA rack.

tpzero = BBC power level when maximum attenuation is on.

tpical and tpi' are typically measured at the beginning/end of scans. The difference $(\mathrm{tpical} - \mathrm{tpi}' = \mathrm{tpdiff})$ is used to derive the conversion factor from tpi level to T_{sys}. The T_{sys} is given by the expression:

$$T_{\mathrm{sys}} = T_{\mathrm{cal}} \times \frac{\mathrm{tpi} - \mathrm{tpzero}}{(\mathrm{tpical} - \mathrm{tpi}')}$$

where T_{cal} is the temperature of the calibration diode (read from the RXG file).

The term BBC is used for both MkIV and VLBA racks.

2 Download and Install

The antabfs suite can be downloaded from:
`ftp://ftp.jive.nl/pub/reynolds/antabfs.tar.gz`

Place the tarball in a convenient directory and

```
> gunzip antabfs.tar.gz
> tar -xvf antabfs.tar
```

The directory structure of the tar archive should be maintained so that the scripts can find the necessary subroutine modules. Apart from that you should be able to untar them on any convenient disk.

The first line of all the scripts is:

```
#!/usr/bin/perl -w
```

where `/usr/bin/perl` is assumed to be the location of the perl interpreter on your system. If this is not correct, you can find the location of the perl interpreter with:

```
> where perl
```

Replace `/usr/bin/perl` with the path returned by the `where` command.

The scripts make use of the PGPLOT perl module. The path to the PGPLOT module should be set in the shell environment variable 'PERL5LIB' if it is not installed in the standard place. If you do not already have the PGPLOT perl module on your system it can be downloaded from:

`http://www.aao.gov.au/local/www/kgb/pgperl/`

2.1 Location of the RXG files

The RXG files contain the antenna gain curves and the $T_{\rm cal}$ values as a function of frequency. Usually there is one RXG file for each receiver. Obviously the scripts also need to know where the RXG files are stored. They will look in the directory './rxg/'. I suggest you use a soft link if that is not appropriate, e.g.

```
> ln -s /usr2/control/ ./rxg
```

Alternatively you can set the variable '$gaindir' in the script `antabfs.pl` to the appropriate directory. The RXG files should have the name `<text><tel><band>.rxg` where `<text>` is any random string, `<tel>` is the 2-letter station code and `<band>` is the one letter waveband identifier. `<band>` can be followed by an optional digit if you have more than one RXG file in a given waveband (e.g. 18/21 cm).

If your station has more than one telescope and you need to use different RXG files for each, then you should make sure that the `<text>` part of the name of the RXG file for each telescope begins with the name of that telescope (i.e. the full telescope name as given in the log header – *not* the two-letter station code). If you then run `antabfs.pl` with the `-g` switch it will only search for RXG files whose name also contains the name of the telescope given in the log header. E.g. the Lovell telescope at C-band could have the name `jodrell1_jbc.rxg` to avoid confusing it with the MkII RXG file with the name `jodrell2_jbc.rxg`. The log header is the comment printed near the beginning of the log file with the format:

```
2004.142.20:51:49.32:" EB028A    2004 JODRELL2  J   Jb
2004.142.20:51:49.32:" J  JODRELL2 AZEL 0.4580 20.0    20 -999.9 999.9 20.0    2 0 -999.9 999.9 0.0 Jb
2004.142.20:51:49.32:" Jb JODRELL2  3822846.76000  -153802.28000  5086285.90000
2004.142.20:51:49.32:"      JODRELL2 1417600
2004.142.20:51:49.32:"  drudg version 030915 compiled under FS 9.6.9
2004.142.20:51:49.32:" Rack=Mark4     Recorder 1=Mark5A    Recorder 2=none
```

3 The Antabfs Suite of Scripts

This section describes how to use each of the scripts in the suite, with details of the possible command line options which can be invoked.

3.1 `antabfs.pl`

This is the main script for producing the antab files. Basic usage is:

```
> antabfs.pl <logfile>
```

The script begins by parsing the log file, determining the frequency setup(s) and extracting the tpi , tpi′ and tpical values as it goes. Once this is complete, you will be asked to enter a plotting device so that various plots can be displayed. I recommend using `/xs` as the plotting device. For each BBC in the experiment, the tpdiff ($\mathrm{tpdiff} = \mathrm{tpical} - \mathrm{tpi}'$) values are plotted as a function of time. The values are fitted in a piecewise linear fashion (default is to use a minimum of 30 points per fitting bin, but this can be changed interactively). Also shown on the plot are the ± 5 median deviations from the best fit straight line. By default, i.e. hitting return at the prompt, tpdiff values outside the indicated range are deleted. The remaining data are re-fitted with straight lines and you will be offered the chance to do further editing based on deviations from the new lines. If the default range is too small or too large, you can change the number of median deviations beyond which to clip by entering the required number at the prompt. The plot will be redrawn with the new clipping range indicated.

If clipping based on deviations from straight lines is not sufficient, you may edit interactively. Pressing 'e' when the tpdiff plot is presented will start interactive plotting. When in this mode you may use the left-mouse button to draw boxes around undesired tpdiff in order to delete them. You may also zoom in by pressing 'z' instead of the left-mouse button. The right-mouse button exits from interactive editing. Read the on-screen instructions for more details on interactive editing options.

When you have finished editing, enter 'Y' at the prompt and you will be offered the next BBC.

After the tpdiff values for all BBCs have been edited, the T_{sys} are calculated. The T_{sys} on each BBC are then plotted and may be edited interactively in a similar manner to the tpdiff values, by pressing 'e' at the prompt. When all BBCs have been checked, a final plot showing the T_{sys} from all the BBCs plotted simultaneously is made so you can check for global consistency. At this point it is possible to go back and re-examine the T_{sys} one BBC at a time if desired.

Optional command line switches to `antabfs.pl`

There are several command line switches. The most useful ones are described here.

```
> antabfs.pl -r <logfile>
```

When the script runs for the first time it produces two files in addition to the ANTAB file. The first is `<exper><TEL>_tpipairs.dat` which contains a list of all the $\mathrm{tpical}/\mathrm{tpi}'$ pairs in the experiment. The second is `<exper><TEL>_tpi.dat` which contains a list of all the tpi values in the experiment. (See Appendix A for a description of the file formats.) If you wish to rerun the script, but keep the editing of the tpdiff that was done during the previous run, then use the `-r` switch. This causes the tpi values and $\mathrm{tpical}/\mathrm{tpi}'$ pairs to be read from the text files instead of from the log.

```
> antabfs.pl -f <logfile>
```

By default, the script uses the 'flagr' output in the log to delete $\mathrm{tpical}/\mathrm{tpi}'$ pairs taken when the antenna was off-source. In cases where the 'flagr' output is in some way corrupted, the -f switch may be used to ignore the flagr output.

```
> antabfs.pl -a <logfile>
```

In order to prevent stray entries in the log which really belong to a different experiment from being used, the script matches the experiment name derived from the name of the log file to the lines in the log header and ignores log entries which come before a match is found. If the experiment name is not present in the log header or if the name of the log file does not follow the standard naming convention (`<`*`expname`*`><`*`station`*`>.log`), use the `-a` switch to prevent this check.

```
> antabfs.pl -g <logfile>
```

Only use an RXG file if its name begins with the full telescope name extracted from the log header (see Section 2.1).

```
> antabfs.pl -S <logfile>
```

turns off the tpdiff plotting/editing. Useful when you know that no editing is required.

```
> antabfs.pl -e <logfile>
```

turns off the T_{sys} editing.

More than one switch may be specified, but they must be specified separately, e.g.

```
> antabfs.pl -a -r -S -e <logfile>
```

Information on other less frequently used switches is printed when you run the script.

3.2 plot_tsys.pl

```
> plot_tsys.pl <antabfile>
```

plots the contents of the final ANTAB file. Will print a warning if the number of valid data points on any channel is less than 10% of that on any other channel (usually indicates a severely corrupted channel).

Optional command line switches:

`-n` plots each channel on a separate panel, instead of overplotting all channels on a single plot.

`-s` marks the scan boundaries (if indicated in the ANTAB file).

3.3 plot_badtsys.pl

```
> plot_badtsys.pl <antabfile>
```

indicates whether each $T_{\rm sys}$ value was valid or invalid (invalid means negative or 999.9). Valid points are plotted with value 1, invalid points are plotted with value 0 and appear red on a colour display. Use this to make sure there are a reasonable number of $T_{\rm sys}$ values on each channel.

3.4 plot.pl

Generic plotting program. For details, see the usage information printed when you run the program.

3.5 antab_plottpi.pl

Allows you to plot the *tpi.dat and *tpipairs.dat files which are created by `antabfs.pl` and contain the tpi and tpdiff values respectively (see Appendix A for a description of the file format). This is rarely necessary, but can be a useful diagnostic if problems occur.

```
> antab_plottpi.pl -d <tpifile>
```

will plot the difference between the 3rd and 5th columns of the data file. For the *tpipairs.dat file, this is (tpical − tpi′) as a function of time.

```
> antab_plottpi.pl -d -bbc=bbc01 <tpifile>
```

will plot only data points for, in this case, bbc01. Otherwise all BBCs are plotted. Please refer to the relevant BBC using the name bbc* even if you really have a MkIV rack. The BBC number should be specified as two digits, with a leading zero where the number is less than 10.

```
> antab_plottpi.pl <tpifile>
```

will plot tpical or tpi as a function of time, depending on whether you are plotting the *tpipairs.dat or *tpi.dat file.

```
> antab_plottpi.pl -y 5 <tpifile>
```

will plot the fifth column of the *tpipairs.dat file along the y axis. This corresponds to the tpi′ values (instead of the tpical values plotted by default).

```
> antab_plottpi.pl -y 7 <tpifile>
```

will plot the seventh column of the *tpipairs.dat file along the y axis. This corresponds to the tpgain values on a VLBA rack.

4 Multi-frequency Experiments

`antabfs.pl` deals with experiments using frequency switching by creating a separate ANTAB table for each frequency setup found in the log file. tpdiff and $T_{\rm sys}$ values for each frequency setup are plotted and edited separately.

A Appendix: Description of File Formats

The *tpipairs.dat file has 5 columns:

- column 1 = BBC name and unique frequency setup identifier
- column 2 = time of tpical measurement
- column 3 = tpical value
- column 4 = time of tpi' measurement (should be close to column2)
- column 5 = tpi' value

If the rack is a VLBA rack, then the *tpipairs.dat file contains two additional columns:

- column 6 = time of tpgain measurement
- column 7 = value of tpgain

The *tpi.dat file has 3 columns:

- column 1 = BBC name and unique frequency setup identifier
- column 2 = time of tpi measurement
- column 3 = tpi value