CAP
Calibration and Analysis Package
Version 0.9

Fredrik T Rantakyrö
fredrik@astbo1.bo.cnr.it

June 5, 1996

JIVE
IRA - CNR

Contents

1 What is CAP

CAP is a **UNIX** script based program that will produce an brief analysis of the output from the **AQUIR** package. The program will make plots of pertinent data and can be used to find out the changes to the $\mathbf{T_{cal}}$ and the **GAINCURVE**, determine the aforementioned factors if they are unknown. A planned and soon to be implemented feature is to determine $\mathbf{T_{cal}}$ as a function of observing frequency and LO.

2 How to use CAP

2.1 Installation

As stated in the the introduction **CAP** is script based and thus doesn't need any compilation to run. The program heavily uses the **sed** and the **awk** programs to create the correct output, thus both programs are needed in the path. For making graphical output (postscript/x-windows) the program uses the **GNUPLOT** program, version 3.5. This program is included in the FS9 distribution, although a patch that includes the **GNUFIT** is needed. A patched **GNUPLOT** can be found with source code in the same directory where the **CAP** programs are located. You can also make a **fsadapt** then this patch will be put in place.

2.1.1 How to get CAP

CAP can be found on **ASTBO1.BO.CNR.IT**.

1. Log in as **EVN** using the password you can obtain from your local Friend of VLBI.
2. The tar archive can be found in [**EVN.CAP**] under the name **CAP**. To get there just issue the command **cd CAP** after you login.
3. In the same directory you can find the **gnufit.tar** archive which contains executables and all files needed if you want to recompile the program yourself.
4. Select binary transfer by **bin**.
5. **get CAP.tar** to get the archive.
6. If you want the **gnufit.tar** download this one too.
7. After successful transfer **quit**.

In the same directory as the tar archive you can find this document in both postscript and in **LaTeX** form.

2.1.2 First Installation

This describes the first installation for **CAP**

1. Put the **tar**file in the directory where your binaries are stored, typically in **/usr2/oper/bin** in a standard fieldsystem setup.

2. Unpack the **tar**file with the commando **tar xvf cap.tar**.

3. If you put the files in a different directory than specified above you will need to edit the path specified inside the script program. Edit the lines in the **cap** scripts file that say:
   /usr2/oper/bin/
   to say:
   your-new-path-goes-here

4. You are now done with the installation.

2.1.3 Updating the antenna information

All the specific antenna information can be found in the database for each antenna. The source database **sources-freq-code.dat** is the same for all stations, while **station-frequency-code.dat** has to be changed for each station to reflect the local **POLY, DPFU, $\mathbf{T_{cal}}$**.To edit the station file file just open it with your favourite editor and do the necessary changes.

2.2 Usage

The usage is really simple:

1. Change your working directory to that where you have the output from the **AQUIR** package.

2. Run the programs in the package one by one. Below follows a example run with explanation of program parameters and the usage.

   (a) **cap_format frequency-code log-file**.
      - **frequency-code** is the standard **VLBI** code (l/c/q/k/s/x).
      - **log-file** is the **AQUIR** log-file that you want to process.

      This program will take the log-file and make a formatted output that can be used by the other programs in the package. The output will be called station.acp and contains all relevant information in the following order:
      Source_Name Assumed_source_flux Date_in_days Azimuth Elevation Src/Cal Derived_Tcal Derived_Source_flux

No_gaincurve_Derived_Tcal No_gaincurve_Derived_Source_flux Flag(for flagging bad data).
All values with negative fluxes will automatically be flagged from being sent to the output file **station.acp**.

(b) **cap_plot fit yaxis xaxis form source frequency**

- **fit** can be either **0, 1, 2, 3, 4 or 5** for fitting a polynomial of the aforementioned order to the plot, i.e **0** means no fitting just a plot, **1** makes a flat fit and **5** makes a fourth order polynomial fit to the plot(x^4). Start with **0** to get a handle on how the data looks like, then try with **1** to find a good start and then continue with **2, 3,** and finally a fourth order polynomial plot. If the program gives the error **single matrix** edit the **dummy.dat** to try with a different value (if you're fitting to the **nogain_curve** change the **station-freqcode.dat** file and start from scratch with **cap_format**). The best way if the gain-curve is unknown is to start with **station-freqcode.dat** with **caltemp=** used caltemp value and p1-p4 set to 0.0 and do the procedure with **dummy.dat** described above.
- **xaxis** can be either **el** (elevation), or **t** (time in days).
- **yaxis** which is **tcal** (T_{cal}) or **flux**.
- **form** is either **ps** (postscript) or **x** (xwindow output).
- **source** is either a named source or **all** to include all sources in the plot.
- **frequency** is the one-letter frequency code.

This program makes a plot using the **GNUPLOT** package that is either shown to the screen (**x** option) or saved as a postscript file (**ps** option). The ps-file is called **station.ps**. Typically the first time one plots the $\mathbf{T_{cal}}$ vs **time** to find out if there are some bad data so this can be flagged with **cap_clip**. There will always be made a LSQ-fit to the data (a linear fit for $\mathbf{T_{cal}}$, a fourth order fit for $\mathbf{T_{cal}}$ as a function of frequency and for a **GAINCURVE** polynomial fit.

(c) **cap_clip Minimum_Tcal Maximum_Tcal**. This program passes through the **station.acp** file and flags all observations that have $\mathbf{T_{cal}}$ values outside the specified range. All the data are kept and can be unflagged by giving a larger allowed range for $\mathbf{T_{cal}}$.

3. The steps **b - c** are repeated until a satisfactory solution has been found for $\mathbf{T_{cal}}$ and for the **GAINCURVE**. I'll describe the two extreme cases of either a bad value for $\mathbf{T_{cal}}$ or an unknown or poorly known **GAINCURVE**.

   - If the $\mathbf{T_{cal}}$ is unknown just run step **b** plotting $\mathbf{T_{cal}}$ vs **t**. Then clipping unwanted values in step **c**. Incorrect source fluxes will show up as offsets from the ideal flat line in the plot, to complicate it further weather and **gaincurve** effects will distort the plot. The **gaincurve** effects can be avoided by taking a small timesegment and weather by doing the calibration in good clear weather. The wanted $\mathbf{T_{cal}}$ is the the average $\mathbf{T_{cal}}$ in the plot.

- Assuming that $\mathbf{T_{cal}}$ is well known and calibrated then finding out/testing the gaincurve polynomial is fairly trivial. Plot $\mathbf{T_{cal}}$ vs **el** then if the **POLY**nomial factors for the **gaincurve** are incorrect the plot will deviate from a straight line. Any up/down shift of the whole curve curve with respect to a previously known curve can be traced to a incorrect $\mathbf{T_{cal}}$ value. Redo the step described above to determine the correct $\mathbf{T_{cal}}$ value. The deviation from a flat line is due to poorly known parameters in the **POLY**nomial and the new values can be found by making a **least squares fit** to the data.

2.3 Error messages

Occasionally within the fitting rutine there might be the error messages **singular matrix** and **syntax error in data file**. The **singular matrix** is due to bad initial values for the fit, either try a simpler fit or clip away bad data. The **syntax error in data file** is a internal bug for **GNUFIT**, the program dislikes any column which starts with a character, thus any source that has a name that starts with a character will stop the program. The temporary solution is to edit the **station.acp** file with an editor and rename f.ex cygnusa to 3cygnusa.

2.4 Known bugs

There is the problem with alphabetic characters as the first character for a value in a column (see above for a temporary fix). A second known bug is also related to the names of the sources, the sourcenames cannot contain any mathematical operator (+-*/) or any non numerical or nonalphabetical character with the exception of _ sign. The **AQUIR** files sent out should contain names that are setup correctly.

3 How to contact the author

I can be contacted by email at **fredrik@astbo1.bo.cnr.it**. Please send any bugreports, comments and/or suggestions that you have to this address.