AL LAWRENCE ADDITIONS 4/25/13 IN YELLOW

21-SPONGE Data Reduction Manual: Draft 4

Claire Murray

February 24, 2013

TO KEEP LOWERCASE, USE ONLY OPENING QUOTE (')

1 Introduction

This “manual” is meant to supplement Dan Able’s wonderfully detailed recipe for AIPS tasks, written at NRAO Socorro during the summer of 2011 (hereafter referred to as “the recipe”). The descriptions listed here will follow the order of the tasks described in the recipe (for the most part), so it is essential to use both documents simultaneously. I will refer to specific pages and lines of the recipe here, mostly to give more background and explain what is important to see, check and think about in each step. Good luck!

2 Basic AIPS lingo

The following are some useful keystrokes and lingo that you can and should use while working with AIPS. The word “TASK” will be used in place of any AIPS task name (such as FITLD, LISTR, TVFLG etc.), the word “adverb” will be used in place of any adverb within a task (such as outn in FITLD, dotv in POSSM, bchan in TVFLG etc.), and “value” will denote the value of that adverb. The syntax to input into AIPS will be **bolded** and should be accompanied by a carriage return to execute.

- To view the files on your AIPS disk, type: **pcat**.
- To call a task that you have never used before, type: **default TASK**. This will load the default adverb values for the task.
- To re-load the adverbs that you previously entered for a task (i.e., instead of using **default** and having to re-set all the adverbs you want to change), type: **tget TASK**.
- To view the adverb values for a specific task, type: **inp TASK**. Note, that if you immediately previously typed **default TASK** or **tget TASK**, you can simply type **inp** to view the adverbs of that task, because AIPS already knows you want to work with it.
- To execute a task, type: **go TASK**. It is safest to always type the task name that you want to execute after “go” so as not to make unwanted mistakes.
- To change the value of an adverb, type: **adverb value**.
- To select a particular data file on your AIPS disk, use **pcat** to see what the catalog number (“#”) of the data file is, and then type: **getn #**.

- It is possible to type multiple commands on a single line (before executing with a carriage return). To do this, separate the commands with semi-colons. For example: **default TVFLG; getn 4; bchan 128; echan 128; go TVFLG**. AIPS will execute each command in the order it is typed within the line. Note: if there is an error or typo in the line, AIPS will not execute any command that appears after the error/typo.
- The AIPS MSGSRV window will display very important information, so you need to keep it open and visible on your desktop at all times. It will tell you when tasks begin and end, any important output from these tasks, and most importantly, describe the errors if anything goes wrong along the way.
- To view the header information for a data set, type: **imh**. This allows you to see which and how many flagging and calibration tables you've created, as well as some essential information about the data set, like when it was observed, by whom and how big it is. After creating flag ("FG"), bandpass ("BP"), or calibration/solution ("SN") tables, you want to make sure to keep track of how many you have and what the number of the one you want to apply to the data set in various proceeeding steps is.
- Especially based on the above piece of information, it is extremely important to take detailed notes while you're reducing data in AIPS!
- VERY IMPORTANT: before running any task, use **inp** to double and triple-check to make sure that you've input ALL adverbs correctly!!!!!!!!! This will save you a lot of time on troubleshooting later.

`*ADDED ON 01.31.13`

- The command **tvinit** will clear the TV window.
- To delete a table from a dataset, use **getn** to select the dataset, specify the type of table by typing, for example, **inext 'BP'** to delete a bandpass table, and then the version of the table by typing, for example, **invers 1** to delete version 1 (you can verify the versions that exist by using **imh**). Then type **extdest** to delete the table.

3 Initial flagging and editing

Begin by loading the A, B and C datasets for a given observation onto your AIPS disk. You can do this using FITLD if you wish to load FITS files from external disks (which have either already been reduced, or provided to you by me as a work-around to the problems we've had with Obit tasks), or you can do this using BDFLIST and BDF2AIPS from raw SDM/BDF files from the VLA archive, as explained in the recipe. BDFLIST and BDF2AIPS are Obit tasks, which can be called by AIPS if the specific paths to those files are setup correctly for your user number and AIPS machine (i.e. kloster, stripe, jelen). As of December 2012 we are still working with Aaron on getting that working, but it's okay for now as you can use FITLD as well.

Remember that AIPS has a limited string length it can accept, so to specify the path to any datafile you will most likely have to define the file directory as an environment variable. If you're running tsch, you will have to execute the following command <u>before running AIPS</u> (i.e. outside the AIPS environment): "setenv dir /d/leffe/cmurray". If you're running bash, use: "export dir=/d/leffe/cmurray/", again, outside of AIPS.

To run FITLD, you can execute the following commands (for example):

- default fitld
- inp
- datain ‘dir:4C16.09-A.UVAVG
- outn ‘4C16.09-A’
- outcl ‘UVAVG’
- inp
- go fitld

Once the data is loaded, and before tackling it, it’s a good idea to print records of some essential information about the datasets themselves.[1] This is done in steps 4 and 5 of the recipe: printing scan records and the antenna pattern using LISTR and PRTAN. Please read the help files for LISTR and PRTAN and follow the recipe to input the correct adverb values and print the files to save for your records.

Although Dan’s next step is to select bad antennas with a test bandpass, I always flag data in TVFLG first (as bad antennas show up very obviously in TVFLG as well). So following LISTR and PRTAN, skip ahead to step 8 of the recipe. Use TVFLG to interactively edit each dataset:

3.1 TVFLG

As you probably have already encountered, TVFLG is a complicated task that has many options and adverbs. After correctly entering all adverbs from step 8 of the recipe (note that to select all sources, simply type **sources ”**, where those are two single-quotation marks), the visibility dataset will appear in your TV window. It is important to keep your TV window around and visible, as well as the message (or “MSGSRV”) window. I would like you to experiment with TVFLG to get a sense of the different parameters, but there are a few things that you must do immediately when you start TVFLG:

1. When you run TVFLG, you are only displaying the visibilites of one channel (selected using the adverbs “bchan” and “echan” in the recipe). However, you want to apply the flags you generate in the task to the entire dataset, not just that one channel. To change this, select “SWITCH-ALL-CH-FLAG” with your cursor, and press **A** to execute it. You will notice that, at the bottom of the TV window, where it previously said “ONE-CH” it will now say “ALL-CH” and you know you’ve done this step correctly.

2. The datasets contain dual-polarization information, i.e. both right (“RR”) and left (“LL”) circular polarization. To make sure that you are applying flags to both polarizations, select “ENTER STOKES FLAG” with your cursor and press **A** to select it. In the main AIPS terminal, you will be prompted with: “Enter Stokes flag string or mask: 4 chars must begin in col 1”. The Stokes flag string corresponding to all polarizations is ”1111”. So, in the main AIPS terminal under this prompt, type: **1111** (and then return). Then you will be ready to continue with interactive flagging.

[1] Please forgive me that this is not the order we did all of this in initially, I just wanted you guys to get a feel for the data and what it looks like in TVFLG (baselines, visibilities, etc.) before starting off with scan lists and test bandpasses.

In general, you will use the keys **A, B, C, D** to select, navigate and exit from the various TVFLG commands. To select a command, use **A** and to exit from a command, use **D**. B and C do different things depending on the command, and so based on the few essential examples I provide here, I hope you will get a feel for the various options. As a general rule, it is important to remember that *you don't need to go crazy with TVFLG–* only flag what is obviously anomalous. There will be opportunities to flag additional bad timeranges, antennas or baselines later.

Here are some parameters that I use most often in TVFLG, but again, feel free to experiment with others! To execute these, select them from the menu in the upper left with your cursor, and then press **A** to execute.

- TVFIDDLE: after selecting tvfiddle, click anywhere on the visibility map and press **A**. Then click and drag around the image to change the contrast of the visiblilities. Pressing **A** again will display the visibilities in rainbow colors. Pressing **B** and **C** will allow you to zoom in and out of a desired portion of the map. When you have zoomed into a region that you want to apply a flag to, for example, press **D** to exit tvfiddle.

- Use the various "FLAG ..." commands to isolate different regions of the visibility map.

  FLAG BASELINE: drag the purple line around the map and observe the baseline numbers in the top left corner of the window change to figure out which antennas to isolate. Do not press B or C while using this command unless you actually want to flag a baseline: **B** will flag the baseline you select and allow you to continue flagging other baselines, **C** will flag the baseline you select and exit back to the main menu. **D** will exit back to the main menu without flagging any baselines.

  FLAG TIME RANGE: drag the purple line to the beginning of the desired time range and press **A** to hold it in place. Then, drag the second purple line that appears to the end of the desired time range and press either **B** to flag the time range and continue flagging, **C** to flag the time range and exit the command, **A** to edit the placement of the first purple line or **D** to exit without flagging anything (for example, if you would like to zoom in or out using tvfiddle first).

  Others, like FLAG PIXEL, FLAG AREA and FLAG ANTENNA are useful and operate in essentially the same way. Please try them out.

- OFFZOOM: to completely zoom-out back to the initial zoom level.

- The various "DISPLAY..." commands allow you to see different aspects of the data set. Please read about these in the TVFLG help file (by typing **help tvflg** in AIPS outside of the TVFLG task, or for more information: http://www.aips.nrao.edu/cgi-bin/ZXHLP2.PL?TVFLG) so that I don't have to re-explain them!

- To exit from TVFLG, select "EXIT" and press **A** to execute it and return to the normal AIPS environment.

Once you've exited TVFLG, you will be asked whether or not you'd like to apply the flags you just created to the "flag table". Press "Y" or "N" for yes or no. If you didn't make any mistakes and would like to appy the flags, obviously press "Y". This will either generate a new FG (which means "flag") table or append the flags to the most recent FG table, depending on what you set the adverg "outfgver" to when you initially ran TVFLG.

SOURCE WITH LARGEST VISIBILITY = BANDPASS CALIBATOR

EXAMPLE: CALCOUR '3c433 (USE ONLY OPENING QUOTE MARK ')

4 Bandpass calibration

4.1 BPASS

Once you're happy with the job you've done in TVFLG for all three datasets, we continue by working with bandpass calibration using the B and C datasets. Now go back to setp 6 in the recipe to find the essential adverbs to input in the task BPASS. You will now need to use the output files you created in LISTR and PRTAN to define some of these adverbs.

In the LISTR output for the B and C datasets, you can find the name of the bandpass calibrator, which is what you input as a string in calsour: i.e. **calsour 'J0318+1628'**. To define "refant", you need to select the reference antenna for calibration. In your output from PRTAN (a previous step), you should have a map of the antennas as they were during the observation. Select one reference antenna from the three center-most antennas, and preferrably pick one which had few-to-no obvious problems when you were examining the data in TVFLG. If the antenna you select has problems, they will become apparent later on in the process and we can easily re-run all of the intervening steps with a new reference antenna.

Once you've set all the necessary adverbs, run BPASS. This creates a bandpass table ("BP" table), which you will select later with the adverbs "doband" (to apply the BP table) and "bpver" (to select the BP table version you want, which can be verified with **imh**).

4.2 POSSM

Then use the powerful task POSSM to view the bandpass tables just created, by inputting the adverbs listed in the recipe step following the BPASS step (which is listed as 5, but should be 7, and shows me that all the numbering in this document is wacky...but fear not, we'll be fine!). As we discussed, you want to see a well-behaved step function, normalized to 1, which a phase solution that oscillates around 0 with as small a scatter as possible.

4.3 BPLOT

To examine the BP table in much more detail and possibly identify bad antennas, we use the task BPLOT. This is not in Dan's recipe, so bear with me because this is the first description of how to best use it in this context. I would use the following commands with BPLOT to start:

- default bplot
- getn #
- codetyp 'difa'
- sort 'ta'
- bchan 20
- echan 110
- inext 'bp'
- invers 1
- dotv 1

freqid 0

USE ZAP TO DELETE MAIN FILE AS FOLLOWS:

getn # (FILE TO DELETE
zap (DELETE FILE)
recat (RENUMBER FILES)

IF STAT OF FILE = READ/WRITE THEN DO
FOLLOWING BEFORE DELETING:

getn # (SELECT FILE)
clrstat (CLEAR STAT)

- go bplot

You should see a plot on the TV window that shows the difference in amplitude between the bandpass solution for each individual antenna (plotted on the y-axis) and the solution for the reference antenna (selected in BPASS). Any outliers will be obvious and dominate the plot. Click on the TV window and press **B** and **C** to scroll through subsequent plots, as BPASS generates a solution for every antenna for every scan (equivalently, for every time range). Take notes about what you observe here.

The "codetyp 'difa' " means that we have been looking at the difference in *amplitude* between the antennas and a reference– to look at the difference in phase, change the code type to phase by inputting **codetyp 'difp'**. Take notes about what you observe, so that you can flag bad antennas/time ranges/polarizations.

If there are bad antennas, they will dominate relative to the rest of the antennas for a given scan or polarization ("POL1=RR" and "POL2=LL"). Note which antennas are bad, and then you can flag them manually using UVFLG.

4.3.1 UVFLG

The task UVFLG can be used to flag specific time ranges, polarizations, antennas or combinations of those, based on what you observe in various calibration steps. Please read the help file for the task to become familiar with its capabilities. To find the particular time range that you would like to flag, use the task LISTR with "optype 'scan' " to print the scan lists. You should have already done this in a previous step of this document, so you can just read the scan times from that output. Specify the time range by entering each number followed by a space, making sure to include the "day number".

As an example, if you would like to flag "POL0" of antenna 16 in the first scan of the bandpass calibrator, you observe from LISTR that the time range is "0/09:24:25–0/09:32:50"

- getn #
- source 'J0137+3309"
- antenna 16
- stokes 'LL'
- timerang 0 9 24 24 0 9 32 51 1 SEC BEFORE AND 1 SEC AFTER
- opcode 'flag'
- inp
- go uvflg

After excuting UVFLG, you should see the following message in your MSGRV window:

```
localh> UVFLG1: Task UVFLG (release of 31DEC11) begins
localh> UVFLG1: Wrote  1 flags to flag table version 1
localh> UVFLG1: Appears to have ended successfully
localh> UVFLG1: kloster 31DEC11 TST: Cpu= 0.0 Real=  0
```

SOURCE WITH SMALLEST VISIBILITY = PHASE CALIBRATOR
" " LARGEST " = BANDPASS CALIBRATOR

The number of flags written to the flag table may be higher than 1, depending on what you told it to flag (i.e. if you wanted to flag multiple antennas at once, which would specify by typing **antenna 16,23**, for example).

To verify what you've done (and this is always a good idea to do often), type **imh**. Delete the bandpass ('BP') table you created in the previous steps and re-run BPASS and BPLOT to check. (**see the revised notes at the beginning of this document to see how to delete tables**)

Once you've created satisfactory BP tables for both the B and C datasets, we will move on to amplitude and phase calibration.

5 CALIB

CHECK RMS < 0.01 FOR ACCEPTABLE

Go to step 11 of Dan's reduction recipe.

Use the task **CALRD** to create a model of the bandpass calibrator. Use the following list of common calibrators to determine which object to select (i.e. if the bandpass calibrator is J0137+3309 then input **object '3C48'**). The band is 'L'.

```
3C48:  RA 01:37:41.3, Dec 33:09:35.1
3c286: RA 13:31:08.3, Dec 30:30:33.0
3C147: RA 05:42:36.1, Dec 49:51:07.2
```

ONLY USE IF MODEL IS SAME AS BANDPASS CALIBRATOR

Follow the directions for **CALIB** in the reduction recipe, making sure to input all highlighted abverbs correctly. The 'calsour' is the bandpass calibrator, and you can use 'get2n' to select the model file generated by CALRD. After running CALIB, you will generate a "solution table", or 'SN' table if you check the image header of the dataset (again, using **imh**). Next, run calib again, but this time the calsour will be the phase calibrator and you will need to clear 'in2name' and 'in2class' so that you don't use the bandpass calibrator model again. This will create SN 2. Check the header to make sure you have two SN tables (i.e. the highest number listed in the header for SN tables is 2).

5.1 GETJY

After calibrating the bandpass and phase calibrators in both the B and C datasets (**important, make sure you've been doing all of the above steps for both the B and C datasets), use **GETJY** to extract the flux of the phase calibrator. Follow step 13 in the reduction recipe. The output in the message window will be something like this:

```
localh> GETJY1: Task GETJY (release of 31DEC11) begins
localh> GETJY1: Calibrator robust averaging used 276 of  276 gain samples
localh> GETJY1:    Source:Qual      CALCODE IF Flux (Jy)
localh> GETJY1: J2250+1419 :  0   D      1   2.1456 +/-0.0078 0.00117
localh> GETJY1:    Source:Qual      CALCODE  used   total bad used tot bad
localh> GETJY1: J2250+1419 :  0   D      46 46 0     45  46   1
localh> GETJY1: Appears to have ended successfully
localh> GETJY1: kloster 31DEC11 TST: Cpu= 0.0 Real=  0
```

Record the extracted flux value (in the above example, this would be 2.1456 Jy). Then average the results from the B and C datasets and record the result (i.e. write it down).

6 DBCON

GAINUSE = 2

After performing bandpass and amplitude/phase calibration on the B and C datasets you are ready to combine them. Follow step 14 in the reduction manual. Then use POSSM (same inputs as before) to look at the combined bandpass solution. Make a plot file of the bandpass solutions from POSSM by setting **dotv -1**. Follow step 16 of the reduction manual to print the plot file using the task **LWPLA**. You will need to know the 'plver' or plot version of the file, which you can check in the image header just like you've done with the BP and SN tables. Keep a copy of the final combined bandpass solution plots so that we can always go back and look at them.

If everything looks good, copy the BP table from the combined BC dataset to the A dataset using the task **TACOP**:

- default tacop
- getn # *the BC dataset catalog number, to select the file from here*
- geton # *the A dataset catalog number, to send the file to here*
- inext 'BP'
- inver 1
- outver 1
- inp
- go tacop

If successful, you should get a message in the MSGSRV window that looks like:

```
localh> TACOP1: Task TACOP (release of 31DEC11) begins
localh> TACOP1: Copied BP file from vol/cno/vers 1 10    1 to   1  7   1
localh> TACOP1: TACOP   / Copied     1 extension files
localh> TACOP1: Appears to have ended successfully
localh> TACOP1: kloster 31DEC11 TST: Cpu= 0.0 Real=  0
```

otherwise, you'll get an angrier message that involves the line:

```
localh> TACOP1: Purports to die of UNNATURAL causes
```

so make sure your inputs make sense! Or fix them and try again.

At this point, we're done with the B and C datasets and we can move on to the A dataset.

7 Target Calibration

After combining the B and C datasets and copying the master BP table to the A dataset, follow the instructions in the reduction recipe to input the flux of the phase calibrator via the task **SETJY**. Also, if you haven't already, flag data in the A dataset in the same way as before using **TVFLG**.

Calibrate the phase calibrator in the A dataset using the task **CALIB** as described in the reduction recipe. All of the inputs should be the same, and as always make sure that you have input the correct source names. If you didn't do so already, it will be helpful to print out a scan list for the A dataset using **LISTR**.

***If the phase calibrator and target source are the same (as it is for your test case, 4C16.09:* don't worry! Instead of specifying the source name for the phase calibrator, you can specify the calibrator code (which in this case, should be "D"). Actually, for 4C16.09 I notice that the name and code are the same...this problem only exists for older observations like this one, before we realized a more convenient strategy for combining targets and calibrators. Just ignore this issue and continue with the calibration (I previously had to fix the name of the scans manually, and that is an onerous process you don't have to deal with).

7.1 SN Table Details

Check the header to ensure that you've created one SN table. At this point, you should have one SN table ("solution" from CALIB), one BP table ("bandpass" from TACOP) and one FG ("flag", from TVFLG) table in the A dataset. You now need to examine the contents of the SN table to check for any remaining bad data. To do this, use **LISTR**.

- tget listr
- getn # *–the A dataset*
- optype 'GAIN'
- inver 1
- docrt 132
- inp
- go listr

The output will be a table of solution amplitudes for each antenna at each time range, in two batches (one for each polarization "LL" or "RR"). You should examine this table for outlying numbers, i.e. for a given antenna, the solutions amplitudes should not fluctuate greatly. If you see outliers, make note of the antenna, time range and polarization. Use **UVFLG** as you did for flagging bandpass data to flag these bad data. Then delete the SN table using extdes (explained in the first section) and re-run CALIB (using the same inputs as before). Check the SN table again using LISTR to make sure you correctly flagged the outlying data.

7.2 CLCAL

Once you're happy with your SN table, interpolate the target source using **CLCAL** as outlined in the reduction recipe. This will create CL table #2 in the header (check this).

7.3 Reviewing Results

After CLCAL, verify the results of the calibration using POSSM. The following inputs should be changed:

- tget possm

- getn # – *the A dataset*
- doband 1– *to apply bandpass calibration*
- bpver 1 – *to select the BP table version*
- docal 1 – *to apply amp/phase calibration*
- gainuse 2 – *to select the SN table version*
- aparm 0 – *to clear the BP table viewing parameters*
- inp
- go possm

The TV window will display the non-continuum subtracted spectrum! Change the value of dotv to save a file version of the plots, and print them using the task LWPLA for your records. If everything looks good, you'll be ready to move on to self-calibration and continuum imaging.

8 Self Calibration

Following the reduction recipe, use the task **SPLIT** to apply the calibration table (CL 2) and bandpass table (BP 1) to the target dataset. Then follow the steps to use **UVLSF** to subtract the continuum from the target dataset. The important parameters in UVLSF are "dooutput 1", which ensures that you produce a continuum-subtracted dataset (which will show up in your catalog list as .UVLSF) and a continuum-only dataset (which will show up in your catalog list as .BASFIT). Another important adverb is **ichansel**, which you change to select line-free channels from which to estimate the continuum level. Use your plot from POSSM in the previous section to select a range of line-free channels on either side of the HI absorption lines, and input them as "ichansel 20 50 1 1 200 220 1 1" for the ranges 20-50 and 200-220, for example. You should delete the .UVLSF file and re-number the catalog files using **recat**.

9 Continuum Imaging

Then, follow the recipe to create an image of the .BASFIT (i.e. the continuum) file. Make sure to input "dotv 1" to ensure that the image process shows up on the TV window. Use your cursor and the A, B, C and D keys to draw a box around the central continuum source (don't worry about making a perfect box, just make sure the entire point source lies within the box). Click "continue clean" and press A to "clean" the image (i.e. remove phase and side lobe artifacts). When you've done this many times, reducing the levels in the box to what looks like random noise, click "stop cleaning". This will produce two new catalog files, .IBM001 and .ICL001. Delete the .IBM001 file (it is just an image of the "dirty beam") and use **recat** to re-order the catalog numbers.

Display the image on the TV window by using getn to select the .ICL001 file, and then the command **tvlod**. Then, use **twin** to draw a large box on the image, somewhere *not* on the central point source (or, fully in the noise area). Then type **imstat** to print out some important image parameters. The output will look like the following:

```
AIPS 4: Mean=-3.138E-03 rms= 5.889E-03 JY/BEAM  over         441. pixels
AIPS 4: Maximum= 1.1757E-02 at    39    37    1    1    1    1    1
AIPS 4: Skypos: RA 09 42 21.19712  DEC 13 44 21.3155
AIPS 4: Skypos: IPOL  1420.070 MHZ
AIPS 4: Minimum=-1.3194E-02 at   31   22    1    1    1    1    1
AIPS 4: Skypos: RA 09 42 21.41669  DEC 13 44 15.3152
AIPS 4: Skypos: IPOL  1420.070 MHZ
AIPS 4: Flux density = -2.9889E-02 Jy.   Beam area =  46.30 pixels
```

Record the rms noise level in the image, which in the example case is 5.889e-3 Jy/beam. Then type **imh** to view the image header. The bottom of the output should look like the following:

```
AIPS 4: Keyword = 'WTNOISE '  value =  1.185144E+00
AIPS 4: Keyword = 'SUMWTIN '  value =  3.244519E+08
AIPS 4: Keyword = 'CCFLUX  '  value =  3.252069E+00
AIPS 4: Keyword = 'CCTOTAL '  value =  3.252069E+0
```

Record the flux value, which in this case is 3.252 Jy. Calculate the signal-to-noise ratio from these values and record this number.

Continue following the reduction recipe by using **CCMRG** to organize the clean components of the image (the .ICL001 file). Then use **PRTCC** to print them out, pressing enter until you reach the bottom of the list. At the end of the list, it will tell you the "First negative component is number : ". Record this number.

Use CALIB to create a solution table based on this image (or "clean map"). Follow the recipe, most of the inputs are the same as before, but in this case the clean map is the .ICL001 file and the base file is the .BASFIT file. Input the first negative component number ("#") as "ncomp #, 0". Make sure all inputs are correct. The noise level in the solution table in this step may be higher than from the previous calibration steps, don't worry about this.

Once you've created a new SN table in the .BASFIT file, inspect it using the task **SNPLT**:

- default snplt
- getn # – *the .BASFIT file*
- inext 'sn'; inver 1; nplots 4
- inp
- go snplt

This will plot phase amplitude vs. time for each antenna and each polarization. Use "B" to scroll through the antennas, and take notes about what you see in these plots.

Now, copy the SN table from the .BASFIT file to the .SPSFCL (i.e., the file you made with the task SPLIT) file using the task TACOP. Re-run UVLSF, but now make sure to change "docalib 1" and "gainuse 1" to apply the SN table from the continuum map. Repeat the above process in full and calculate the signal-to-noise ratio in the new image. Continue repeating this whole process until the signal to noise stops improving significantly, which usually occurs after two total iterations. The SN table from the final iteration of this process will be your "self-calibration" SN table.

10 Final Steps

***The following steps are in different order than they appear in the reduction recipe! This is very important, you must apply UVLSF before CVEL (see below) in order to avoid introducing Gibbs ringing into the final spectra. Some of the task inputs are different as well, so be careful!*

Return to the target dataset and use **UVLSF** to subtract the continuum and apply the bandpass (BP 1) and calibration tables (CL 2). Make sure to specify the target source name in this step, and your other inputs should be the same as in the self-calibration process, except now set "dooutput -1" to avoid creating a .BASFIT file.

10.1 CVEL

Use the task **CVEL** to correct for the source velocities. Input the following parameters (this is different from the reduction recipe):

- default CVEL
- getn # – *the .UVLSF file just created*
- outn inn
- outclass 'CVEL'
- source 'J0318+1628" – *the target source name*
- doband -1
- aparm -5000 129 0 0 1.420E9 5752 0 1 0 0 – *see the descriptions on the side to understand each of these inputs*
- inp
- go cvel

If everything works correctly, the "average shift" for each antenna should be on the order of 1-2 channels. If the shift is huge (for example, -128 channels), delete the output .CVEL file, re-check your inputs and run CVEL again.

10.2 Apply self-cal SN table

Copy the self-cal SN table from the .SPSFCL (or most recent .BASFIT) file to the .CVEL file. Use SPLIT to apply the SN table to the .CVEL file. Your inputs should be the same, but change the following: "docalib 1; gainuse 1; outclass 'SCVEL'". After you've done this, use POSSM again to create plot files of the target source (inputs should be exactly the same, just select the .SCVEL file with getn). Compare these to the POSSM output from before.

If everything looks good, you're done with calibration and editing! The next step will be imaging.