C A Step-by-Step Recipe for VLBA Data Calibration in AIPS

This appendix, written by Jim Ulvestad, provides a step-by-step guide to calibrating many types of VLBA experiments. Continuum strong-source or phase-referencing observations are included, as are simple spectral-line observations. This appendix applies specifically to VLBA-only data sets, but also includes an addendum describing modifications for VLBA+VLA data sets. It may be used (with some modifications in loading amplitude data) for data sets containing other antennas. Simple VLBA utilities that go all the way up to (but not including) fringe-fitting are described.

C.1 Revision History

- Released Version 1.0 by Jim Ulvestad, 13 June 2000
- Version 1.1, VLBA utils, VLA calib, other corrections 29 June 2000
- Version 1.2, minor corrections, 20 July 2000
- Version 1.3: added more VLBA utilities, 2 January 2001
- Version 1.4: added more VLBA utilities, 1 June 2001

C.2 Table Philosophy

AIPS follows an incremental calibration process on multi-source data sets. Calibration solutions are written to `SN` ("Solution") tables, which can be inspected in various ways. `CLCAL` is used to apply an `SN` table and write a new `CL` ("Calibration") table, which stores the cumulative calibrations. The actual visibilities are not altered until the final calibration is applied using `SPLIT` (or `SPLAT`), which produces single-source data sets that can be imaged. With this philosophy, it is easy to back up a step or two if errors are made in processing. Users should keep track of which tables contain which solutions and calibrations as they go through the calibration process.

A key verb to be aware of is `EXTDEST`, which can delete any unwanted table. For example, to delete `SN` version 3 from the data set catalogued as data set 1 on disk 1, type `INDIS 1; GETNA 1; INEXT 'SN'; INVER 3; INP EXTDEST; EXTDEST`. Some people like to delete `SN` tables after they have been applied, but I prefer to keep them in case I need to back up several steps and re-use them at at some point. *Beware of the fact that once a table is deleted, there is no 'undelete' function.*

C.3 Data set assumed in this memo

This memo assumes a VLBA-only initial data set containing observations at several frequency bands (*e.g.*, 1.6, 2.3, and 5.0 GHz), with a total of k correlator jobs contained on the distribution tape. Inclusion of data from the VLA is treated in § C.9. A phase-referencing program should have been observed according to the philosophy discussed in detail in VLBA Scientific Memo No. 24. The hypothetical observation considered here contains the following sources:

- 'CAL-1' — fringe-search and bandpass calibrator
- 'CAL-2' — amplitude-check source
- 'CAL-3' — polarization position angle calibrator
- 'STRONG' — strong target source
- 'CAL-4' — phase-reference source
- 'WEAK' — weak target source, to be calibrated with CAL-4

In the text below, table versions, such as SN version 1, are referred to as SN 1.

C.4 VLBA Utilities

Note that there are simple VLBA procedures ("front ends" to standard tasks) in the 31DEC01 and later versions of AIPS that will take the user all the way from data loading up to and including fringe-fitting. These are tremendous labor-savers for those working with reasonably straightforward data sets. To access the utilities, type RUN VLBAUTIL C_R from inside AIPS. The currently available procedures that automate data reduction are

- VLBALOAD: loads VLBA data with simplified inputs
- VLBASUBS: finds subarrays in VLBA data
- VLBAMCAL: removes redundant calibration data from tables
- VLBAFQS: copies different frequency IDs to separate files
- VLBAFPOL: fixes polarization labelling for common cases
- VLBASUMM: makes summary listings of your data set
- VLBACALA: determines *a-priori* amplitude calibrations
- VLBAPANG: determines phase corrections for parallactic angles
- VLBAPCOR: determines instrumental phase corrections
- VLBACPOL: calibrates cross polarization delays
- VLBAFRNG: does global fringe fit using FRING
- VLBAKRNG: does global fringe fit using KRING
- VLBAFRGP: does global fringe fit for phase referenced experiments using FRING
- VLBAKRGP: does global fringe fit for phase referenced experiments using KRING
- VLBASNPL: plots the SN or CL tables versus time
- VLBACRPL: plots the cross-correlation spectrum

There are two additional procedures that can make life easier, called ANTNUM and SCANTIME. ANTNUM will return the antenna number of the antenna corresponding to a certain character string, for a data set containing an AN table. For example, in most data sets, typing REFANT = ANTNUM ('BR') C_R will be the equivalent of typing REFANT = 1 C_R. SCANTIME will return the time range of a given scan number, for use in various programs. Typing TIMERANG = SCANTIME(4) C_R will fill the eight-element array TIMERANG with the start and stop times of the 4$^{\text{th}}$ scan of a given data set. (There must be an NX table for this to work.)

C.5 Data Loading and Inspection

1. Load the data using FITLD. Key parameters are NCOUNT = k; DOUVCOMP = 1; DOCONCAT = 1. The typical CL table interval in minutes should range from CLINT = 0.25 to CLINT = 1.0. In order to *irrevocably* delete data that didn't record or play back well, use adverb WTTHRESH. Typical values are 0.7 or 0.8; useful selection can be made by inspecting the plot of correlator playback weights. The utility VLBALOAD has a less daunting set of input parameters for new users.

2. An antenna gain (GC) entry is supplied for each correlator job, and passed on by FITLD. It's necessary to merge duplicate entries in GC, PC (pulse-cal) and TY (system temperature) tables. To do this, RUN MERGECAL loads a procedure running the *AIPS* task TAMRG, then MERGECAL actually invokes the procedure. Alternatively, use the VLBA utility VLBAMCAL, which is as simple as we can make things!

3. FITLD may give an error message about subarraying, especially in phase-referencing observations where antennas switch sources in a few seconds. This is sometimes caused by data being out of proper time order. If an error message of this type appears, then run MSORT with SORT = 'TB'. You may write to a new file (safer in case you need to ABORT the task for some reason) or overwrite the input file. If the subarraying message was noted, then run USUBA with OPCODE = 'AUTO', or run the simpler VLBA utility, VLBASUBS. Finally, if MSORT and/or USUBA have been invoked, run INDXR with CPARM(4) = 1 and CPARM(3) equal to the desired interval for the CL table.

4. Data sets containing multiple frequency bands should be divided into individual data files and calibrated separately. First, run LISTR with OPTYP = 'SCAN', and determine which frequency ID is which band. Then, run UVCOP once for each band to split the data into individual bands, using the appropriate values for FREQID. After UVCOP, run INDXR on each output data set to create the NX (Index) table. Alternatively, the UVCOP and INDXR steps can be done with the utility VLBAFQS, which has simple and straightforward inputs.

5. From now on, assume a data set containing only a single frequency band. Run LISTR with OPTYP = 'SCAN' and DOCRT = −1 to print the scan summary. Run PRTAN with DOCRT = −1 to get a copy of the antenna table.

6. Polarization data may be incorrectly labeled in some instances, due to a lack of adequate room in the VLBI FITS specification to label the polarization of each band and IF channel. There are several common situations in which this occurs, two of which are dealt with below. A simpler approach may be to run the VLBA utility VLBAFPOL to fix the polarization labels or, in more complicated cases, to recommend a course of action.

   (a) If dual polarization (RR and LL, no cross-hands) was used, run FXPOL with BANDPOL = '*RL' for normal VLBA setups. (For Mark 4 setups, which were probably not used for a VLBA-only data set, you may need to run FXPOL with BANDPOL = '*LR'.)

   (b) If multiple bands were used, standard setup files probably caused 2.3 and 8.4 GHz to be observed in RCP, while others were observed in LCP. Therefore, when RCP and LCP observations occur in the same program, the polarizations are almost certainly mislabeled. Identify the polarization that you know was used for a given frequency band (*e.g.*, from the schedule file). Then, run FXPOL with BANDPOL = '*L' to change to LCP, or BANDPOL = '*R' to change to RCP. The result can be checked using IMHEAD to show the data-set header, which will contain STOKES = -1 for RCP and STOKES = -2 for LCP.

7. For a simple spectral-line data set, or any data set with high spectral resolution (*i.e.*, more than 16 or 32 channels per IF), it is a very good idea to average the data set to 16 or 32 channels before deriving the calibration parameters. Otherwise, the calibration tasks may take forever to run on the large data set. To reduce the data-set size, run the task AVSPC with AVOPTION = 'SUBS'. For example, to average IFs with $N_{\rm chan}$ down to 16 channels, set the adverb CHANNEL = $N_{\rm chan}/16$ (*e.g.*, to average from 1024 to 16 channels, use CHANNEL = 64).

8. The VLBA utility VLBASUMM should now be run to generate listings showing the scans, sources, frequencies, and antennas in your data set. These will be valuable references throughout.

C.6 Amplitude Calibration

Amplitude calibration uses measured antenna gains and system temperatures, as well as finding a correction for voltage offsets in the samplers. It is assumed that the gain and T_{sys} data are supplied as tables attached to the correlator output. If this is not the case (VLBA data before April 1999, or data from other antennas), consult the *AIPS CookBook* § 9.4.2.5.

1. This is where you have to incorporate the VLA, if a single antenna or the phased VLA was used. See § C.9 for details on setting up an input file, then running ANTAB and APCAL. That section also gives more details about things to watch out for when running APCAL. A similar procedure can be followed for other non-VLBA antennas.

2. All steps below (except data inspection), including reading and applying the gain and T_{sys} tables, and calculating, clipping and applying the sampler offsets, can be carried out using the VLBA utility procedure VLBACALA. This will produce a CL table containing both the *a priori* amplitude calibration and the sampler offsets. Inputs are self-explanatory.

3. Apply the GC and TY tables with APCAL to make a gain solution table, SN 1. Use GCVER = 0; TYVER = 0; SNVER = 0. If APCAL fails with some bizarre-sounding error message, it is possible that you have an antenna with no calibration data, or calibration data for an antenna that is not in your subarray.

   (a) Inspect SN 1 using SNPLT, to plot voltage gains versus time, or use the VLBA utility VLBASNPL. This can reveal data that need flagging, and show that initial calibration worked correctly. To inspect IF m, use BIF = m; EIF = m; OPTYP = 'AMP'; INVER = 1; INEXT = 'SN'; OPCODE = ' '; NPLOT = 10; DOTV = 1; GO SNPLT. For hard copy, use DOTV = -1; GO SNPLT; GO LWPLA. Plotted amplitudes are the square-roots of the system-equivalent flux densities (SEFDs), in Jansky, where the SEFD is the flux density of a source that would double the system temperature. (Low numbers are good!) At centimeter wavelengths, VLBA antennas have SEFDs near 300 Jy, so gains above 30° elevation should be near 17–18. If two polarizations are included, run SNPLT once for STOKES = 'L' and once for STOKES = 'R'. To look at the input system temperatures, run SNPLT with OPTYP = 'TSYS'; INEXT = 'TY'; INVER = 0. On rare occasions, you might find clearly discrepant points that have leaked in from a different frequency band. In that case, you can use task SNEDT, or the 'CLIP' option of SNSMO (see below), to get rid of the bad points.

   (b) At this point, you may wish to use your favorite method of inspecting data for flagging (*e.g.*, TVFLG, EDITR, IBLED). On-line flags are already included in FG 1; it is a good idea to use TACOP to copy FG 1 to FG 2, then work with FG 2, so the data do not have to be loaded again if mistakes are made. (But be careful to use the proper FLAGVER in various programs.) Some people do no additional flagging at this stage, but later use the results of fringe-fitting and visibility plots of calibrated data to point the way to bad data, or they do their flagging in the Caltech program DIFMAP. All data below a given elevation also can be flagged in a separate run of UVFLG; *e.g.*, to flag all data below 10°, run UVFLG with APARM(5) = 10. Elevation *vs.* time can be listed with LISTR, using OPTYP = 'GAIN'; INEXT = 'SN'; INVER = 1; DOCRT = 1; DPARM(1) = 11. Note that FG tables are not applied to other tables, so flagged data may still have points plotted by SNPLT.

4. Pick a reference antenna now. A good choice is an antenna in the southwestern U.S. (PT, LA, FD, or KP) that performed well according to the log, the PI letter, and the initial amplitude calibration. Hereafter, this is denoted as antenna n.

5. Run CLCAL, to make CL 2. Critical inputs are SOURCE = ' '; CALSOUR = ' '; INTERPOL = 'SELF'; GAINVER = 1; GAINUSE = 2; SNVER = 1; REFANT = n; INP CLCAL; GO CLCAL. (Using

`INTERPOL='SELF'` avoids system temperature interpolation between different sources at different elevations.)

(a) In case of paranoia, run `SNPLT` again, with `INEXT = 'CL'; OPTYP = 'AMP'; INVER = 2`. This should look similar to `SN` 1, but with more entries (due to interpolation).

(b) Run `POSSM` on a calibrator to check that `CL` 2 has appropriate values; look at calibrator flux and phase coherence within each IF. Plot cross-correlation amplitude and phase for a short time period. The VLBA utility `VLBACRPL` is helpful here. The phase will show a slope *vs.* frequency, indicating an uncalibrated (so far) residual delay. Sample inputs are `SOURCE = 'CAL-1'; TIMER = 0 8 10 0 0 8 12 0; BIF = 0; EIF = 0; NPLOT = 9; APARM(9) = 1; ANTENNA =` n`; BASELINE = 0; DOCALIB = 1; GAINUSE = 2`. Use `STOKES = 'RR'` or `'LL'` as appropriate. For a weak phase-referencing calibrator, the flux density may look too high due to scalar averaging of the amplitudes, which are dominated by noise. If the data are coherent over the desired time range, a vector average, using `APARM(1) = 1`, will provide a more realistic estimate of the source flux density.

6. Run `ACCOR` to use autocorrelations to correct for sampler voltage offsets. This is probably significant only for 2-bit sampled data, but should be run for 1-bit data too, just in case. Use `SOLINT = 2`. `ACCOR` will write `SN` 2. In `SNPLT`, the solutions should be near 1000 in milligain. Some IFs may be $\sim$ 5% lower than other IFs due to the VLBA system design; application of the `ACCOR` solutions will (among other things) give proper relative calibration among the IFs.

7. Run `SNSMO` to clip bad points, which occasionally occur due to samples that leaked in from another band, or to antennas looking at the Earth. Use `INTERPOL = 'MWF'; SMOTYPE = 'AMPL'; BPARM = 0; CPARM = 0; CPARM(1) = 0.5; CPARM(6) = 0.02; INVER = 2; OUTVER = 3`. This clips all `SN` entries differing by more than 0.02 from the value given by a 0.5-hr median-weight filter, and writes to `SN` 3. (`SNSMO` does *not* flag visibility data; it just clips the `SN` table.) Use `SNPLT` to inspect `SN` 3.

8. Run `CLCAL` to make `CL` 3. Use the same inputs as before, except `GAINVER = 2; GAINUSE = 3; SNVER = 3`. `CL` 3 will look a lot like `CL` 2, with modifications at the 5%–10% level. Use `POSSM` or `VLBACRPL` as before, changing to `GAINUSE = 3`. Most variations among different IFs on a given baseline should now be leveled out.

9. This is a useful time to run `TASAV` to save all your ancillary tables to another file. If you foul up the calibration, the relevant tables can be copied back using `TACOP`.

C.7 Delay, Rate, and Phase Calibration

Now that the data are amplitude-calibrated, the next step is to do the calibration of the antenna delays, rates, and phases. This section describes that process.

1. It may be essential (for polarization) or very important (for phase-referencing) to correct for the antenna parallactic angles. *Be sure you have run* `TASAV` *first!* since `CLCOR` modifies an existing `CL` table rather than creating a new one. You can skip this step for strong-source, single-polarization observations, but it won't hurt if you do it. To make the correction, run `CLCOR` with `OPCODE = 'PANG'; CLCORPRM = 1,0; GAINVER = 3`. Alternatively, you can run the VLBA utility `VLBAPANG`, which has a much simpler (and self-explanatory!) set of inputs.

2. Next, the large delay residuals due to clock and clock-like uncertainties are taken out. Two different methods for this are described below. **Use only one; do not accidentally apply both!** Also, make sure you run `CLCOR` to correct for parallactic angles before taking out the delay residuals.

(a) One method is to apply the measured pulse-cal tones from the PC table, which provide a measure of delay and phase in each IF. Run PCCOR on a calibrator scan, with typical inputs of CALSOUR = 'CAL-1'; INVER = 0; SNVER = 0; REFANT = n; DELCORR = 0. Make sure that TIMERANG is large enough to include at least one PC entry for each antenna, for deciding on ambiguities; usually 5 minutes or so will suffice. This will make SN 4. The VLBA utility VLBAPCOR simplifies this process and will even use FRING (see below) on those antennas which lack pulse-cal tones.

(b) The alternate method is to use the fringes on a strong source to compute the delay and phase residuals for each antenna and IF. This is useful if some antenna does not have pulse-cal tones (*e.g.*, the VLA) or if you don't trust the tones for some reason. Run FRING on about two minutes of data including all the antennas. Typical inputs are CALSOUR = 'CAL-1'; DOCALIB = 2; SNVER = 0; TIMER = 0 8 10 0 0 8 12 0; GAINUSE = 3; REFANT = n; APARM = 2, 0; APARM(6) =1; DPARM = 1, 0; DPARM(8) = 1; SOLINT = 0. This will make SN 4. To prevent the rate solutions from being applied, it is important to set DPARM(8) = 1; you only want delay calibration. If there appears to be a problem with the FRING solutions, a list of possible steps to take is given below, after the description of the final FRING run.

(c) Warning: Sometimes, you will not find a time range for which all the antennas could see the source, or for which all the antennas performed well. Or FRING may not find a solution for some antenna, for a reason that escapes you. In FRING, you might try reducing the SNR threshold by setting APARM(7) to 4, or changing to a different scan or TIMERANG, or narrowing the search window with DPARM(2) and DPARM(3). But if things still don't work right, you may have to resort to running FRING or PCCOR twice (or more), on two different scans. If you do this, write one solution to SN 4, and the other to SN 5.

3. Run CLCAL as before, except with GAINVER = 3; GAINUSE = 4; SNVER = 4. If you had to run FRING or PCCOR twice (see above), then you will need to run CLCAL twice. For the first run, use GAINVER = 3; GAINUSE = 4; SNVER = 4. In addition, provide a list of antennas for which the solution should be applied, using the ANTENNAS adverb. Be sure to include the reference antenna in the list. Then, run CLCAL again, with GAINVER = 3; GAINUSE = 4; SNVER = 5. Use ANTENNAS to provide a list of antennas for which SN 5 is to be applied; this list should *not* have any antennas in common with the list for the run using SN 4. The same REFANT must be used, but the reference antenna should not be specified in the ANTENNAS list for the second run.

(a) CL 4 can be checked using SNPLT or POSSM or their VLBA utility equivalents VLBASNPL and VLBACRPL. In SNPLT, choose OPTYP = 'DELA' or OPTYP = 'PHAS'. For a given antenna and IF, SNPLT should show a single value of delay repeated for the entire length of the data set, while phase will vary slightly due to the parallactic angle correction and/or the pulse-cal. application. If only a portion of the observation appears, there may have been a problem, such as specifying only a certain time range or source in CLCAL. If some antenna was not operating at the beginning (typically MK or BR) or end (typically SC or HN) of the observation, some CL entries will be missing; this is okay. If there appears to be a problem, use SNPLT to look at the previous CL tables to see where things went wrong, delete any erroneous tables, back up to the appropriate stage of the calibration, and move forward from there. For POSSM, use the same inputs as before, except with GAINUSE = 4. The plotted cross-correlations should show the phase slope removed from each IF, although the phase will not usually be at 0°. (Non-zero phase slopes may still be seen at low elevations, where the atmosphere causes additional delay residuals, or for low-frequency observations where the ionospheric delay varies.) If POSSM is run on a scan of the phase-reference source, CAL-4, there may be more phase noise than for CAL-1, because the source is likely to be much weaker. Try APARM(1) = 1 (vector averaging) to get a true measure of the source flux density.

4. For phase-referencing observations at 1.6 or 2.3 GHz, it may be useful to correct for the total electron content along the line of sight to each source, with TECOR. Such a correction is not needed if the program source itself is strong enough to give good fringe fits. The correction uses global ionosphere models

available from NASA Goddard Space Flight Center. Details are in VLBA Scientific Memo No. 22, and in § C.10 of this document.

5. Now, calibrate the delays, rates, and phases for the entire data set using FRING. The VLBA utilities VLBAFRNG and VLBAKRNG simplify access to FRING and KRING, respectively, while VLBAFRGP and VLBAKRGP are variants appropriate to phase-refencing observations. Inputs are similar to those given for the FRING run described above, except for TIMERANG = 0; DPARM(8) = 0; GAINUSE = 4 (or GAINUSE = 5 if you ran TECOR). SOLINT must be no longer than the expected coherence time at the observing frequency, typically a few minutes at 5 or 8 GHz, possibly one minute or less at 22 GHz and higher. Generally speaking, it is a good idea to fringe-fit to all calibrators and strong target sources, so CALSOUR = 'CAL-1','CAL-2','CAL-3','CAL-4','STRONG'. (You might want to restrict the channel range slightly using BCHAN and ECHAN, since the channels at the high end of each IF will have lower SNR, due to the cutoffs in the bandpass filters. For a data set with 16 channels per IF, numbered from 1 to 16, setting ECHAN to 14 or 15 may be worth trying.) The solutions will be written in SN 5. (If you had to run FRING or PCCOR more than once previously to get solutions for all the antennas, the SN table number should be incremented appropriately.) Note that some people like to run CALIB rather than FRING for this stage of phase-referencing observations, but I prefer to stick to FRING, because it solves for rates, which CALIB doesn't do.

   (a) The above fringe fit will take a bit of time, depending on the computer and the spectral resolution. Then, use SNPLT, if desired, to inspect the solutions. It's not totally out of the question that some data will be found that need flagging, which can be done with UVFLG. In that case, it's a good idea to delete the last SN table and re-run FRING.

   (b) This fringe-fitting stage is the most likely place where things can go wrong, for reasons that are not immediately apparent to the observer. Below, a few common examples are listed.

      - **Many solutions failed.** The source may be too weak, or the coherence time too short. Try increasing or decreasing SOLINT. Or reduce the SNR threshold with APARM(7), and narrow the search window. For most VLBA data, DPARM(2) = 400 and DPARM(3) = 60 should be a good first step, though the rate window specified in DPARM(3) is proportional to the observing frequency, and may need to be larger at 22 GHz and above. Try setting ECHAN so that the top one or two spectral channels in each IF are not used. If the phase-reference source was too weak, you might try restricting solutions to the shorter baselines with UVRANGE, but it also might be that you're out of luck!
      - **Some antenna has low SNR, and may cause an entire set of solutions to go bad.** This typically happens because an antenna should have been flagged. A common cause is when OV is looking at the White Mountains, and neither the on-line system nor the astronomer has flagged the data. Then, you need to run UVFLG and re-run FRING.
      - **The task fails with some message related to memory allocation.** This may happen if there are lots of spectral channels, or a long SOLINT. Possible solutions are to run AVSPC, to reduce the size of the search window with DPARM(2) and DPARM(3), or to reduce SOLINT. This is much less likely to occur in 31DEC00 or later versions downloaded after July 20, 2000.
      - **There are discrepant delay/rate solutions.** Look at the solutions you believe, and try FRING again with DPARM(2) and DPARM(3) specified appropriately. Full widths are specified, so if the good solutions fall between +15 mHz and −15 mHz, use DPARM(3) = 30. (Actually, you should use a value somewhat larger to allow some margin.) It may be that an antenna is suffering from radio-frequency interference, so some channels and/or IFs will need to be flagged.
      - **Some solutions are outside the specified delay/rate range.** This can happen because the initial coarse fringe search uses the range specified by DPARM(2) and DPARM(3), but the least-squares solution can take off from there and go elsewhere.
      - **Delays and rates for some station change rapidly near the beginning or end of the observation.** This may be caused by low elevation at the relevant station. Depending

on how desperate you are to include low-SNR data, you may wish to flag some time range, or flag all data at elevations below 5° or 10° (particularly at high frequencies).

- **Phases wrap rapidly, particularly on the phase-reference source, CAL-4.** There may not be a lot you can do about this initially, because it's possible that the tropospheric delay just changed too fast for the cycle time used in the observation, especially at low elevation. However, you may wish to note the times and antennas when the phase connection is best (typically the southwestern antennas near transit). Later, when imaging the program source, it can be helpful to image with a subset of antennas and time ranges, then use that initial image to self-calibrate the rest of the data.

6. Apply your last SN table (probably 5 or 6) using CLCAL. Here, be careful to calibrate only the proper sources with each calibrator, which requires running CLCAL several times. This is done by the VLBA utility procedures. Inputs are TIMERANG = 0; REFANT = n; GAINVER = 4; GAINUSE = 5; INTERPOL = 'AMBG'; OPCODE = 'CALI'. SNVER is probably 5 or 6, depending on your initial delay calibration with PCCOR or FRING. Note that if TECOR was run, use GAINVER = 5; GAINUSE = 6. Then run CLCAL once for each calibrator/target source or pair, as follows:
   (a) CALSOUR = 'CAL-1'; SOURC = 'CAL-1'; GO
   (b) CALSOUR = 'CAL-2'; SOURC = 'CAL-2'; GO (ignore the message that the SN table has already been applied)
   (c) CALSOUR = 'CAL-3'; SOURC = 'CAL-3'; GO
   (d) CALSOUR = 'STRONG'; SOURC = 'STRONG'; GO
   (e) CALSOUR = 'CAL-4'; SOURC = 'CAL-4','WEAK'; GO
7. If you used AVSPC to reduce the size of the data set used in determining calibration, you must copy your final calibration tables back to the full-size data set. This can be done with task TACOP. For bookkeeping purposes, it may be best to copy over all the CL tables with the same table numbers in both the averaged and un-averaged data sets.
8. Now look at the last CL table with SNPLT, and the overall calibration of particular scans with POSSM, to make sure nothing got messed up. Pay particular attention to the solutions for your phase-reference source, CAL-4, and use APARM(1) = 1 to get a good value for the flux density of that source.

C.8 Final Calibration Steps

1. In some cases (spectral-line observations and continuum experiments seeking dynamic ranges of a few thousand or more, or large fields of view), it is important to calibrate the bandpass shapes. To do this, run BPASS on the bandpass calibrator, CAL-1. Inputs are CALSOUR = 'CAL-1'; BCHAN = 0; ECHAN = 0; DOCALIB = 2; GAINUSE = 5 (or 6 if TECOR was run); SOLINT = 0; SMOOTH = 1,0; BPASSPRM = 0; REFANT = n; INP; GO. This will make BP 1.
2. Polarization calibration still remains, if desired, and if all the appropriate calibration sources were observed. This is currently relevant for a relatively small fraction of VLBA observations, and can be done in a variety of ways. See § 9.4.8 of the *AIPS CookBook* for details.
3. Finally, apply the calibration to the visibility data and make single-source data sets using SPLIT. (Some people might wish to use SPLAT to average over time as well as spectral channel.) Inputs for a continuum observation are SOURCES = ' '; BIF = 0; EIF = 0; DOPOL = -1 (or 1 if polarization calibration was attempted); DOBAND = -1 (or 1 if bandpass calibration was done); DOUVCOMP = 1; NPOINT = 0; APARM = 1,0; DOCALIB = 2; GAINUSE = 5 (or 6 if TECOR was run). For a spectral-line observation, set APARM = 0, because you don't want to average over frequency. Use OUTDISK and OUTCLASS as appropriate for your computer and record-keeping purposes.

The single-source data sets are now ready for imaging and possible self-calibration. At this point, it is a good idea to look at the amplitude check source 'CAL-2' using tasks such as UVPLT or VPLOT in order to see if there are any antenna gain calibrations that must be adjusted. Doing a UVPLT for each target source also is a good idea, because there may be discrepant amplitude points due to interference or poor fringe fits (among other things). UVFLG and CLIP are useful tasks to deal with these bad points.

C.9 Incorporating the VLA

The observation being calibrated may have incorporated either a single VLA antenna or the phased VLA, but the amplitude calibration parameters for the VLA are not transferred automatically. (See VLBA Operations Memo No. 34 for some details.) You will need to create an input text file for the VLA, then run ANTAB before APCAL. The gains and system temperatures for this file, in an appropriate format, are supplied in a file called *xxxxx*cal.y.gz', where '*xxxxx*' is the observation code (*e.g.*, 'bm120'). That file contains instructions on editing the file to get correct inputs. For a phased array or a 1.3-cm observation in which 3 antennas are used, follow the instructions in § C.9.2; for a single antenna, use § C.9.1.

Note that for another "foreign" (non-VLBA or non-VLA) antenna, a procedure similar to that in § C.9.1 can be followed.

C.9.1 Single VLA Antenna

In the input text file, add an INDEX entry within the TSYS card (**Do not separate the INDEX entry from the TSYS entry by a "/" !!!**), uncomment the GAIN line for your particular observing frequency, and uncomment the TSYS line. Then, place this text file in an appropriate directory to read it in with ANTAB. The most straightforward step is to place it in directory $FITS with filename 'VLA.ANTAB'. At the AOC in Socorro, the $FITS directory for a given computer, *e.g.*, 'laguna', is located at /DATA/LAGUNA_1/FITS.

1. After merging the calibration tables for the VLBA antennas, prevent confusion and any chance of having to re-run FITLD by first copying the VLBA TY and GC tables. If you used VLBAMCAL, the VLBA parameters will be in TY 1 and GC 1, and they should be copied to TY 2 and GC 2. If you used MERGECAL, the VLBA parameters will be in TY 2 and GC 2, which should be copied to TY 3 and GC 3. For the example used here, then run ANTAB with INFILE = 'FITS:VLA.ANTAB', setting TYVER and GCVER to their highest numbered values (either 2 or 3). If ANTAB fails, it is most likely caused by having an incorrect format for the input file. Perhaps you forgot to add the INDEX entry within the TSYS card, or gave it the wrong format, or failed to uncomment the GAIN or TSYS lines.

2. Now, run APCAL as described in § C.6 to combine the gain and system temperature information for all antennas into the appropriate SN table. Presumably, this will be SN 1. Therefore, the APCAL inputs should include TYVER = 0; GCVER = 0; SNVER = 1; ANTENNAS = 0. If APCAL fails, it is possible that your input file was not properly set up for ANTAB. Perhaps the INDEX line gives polarizations that are inconsistent with those specified in the headers of the VLBI data set, which could mean you forgot to run FXPOL. Then, use SNPLT as described in § C.6 to make sure that SN 1 now contains amplitude calibration for all VLBA antennas and the VLA. At most frequencies, the VLBA antennas should perform slightly better than a VLA antenna, so the amplitude gains plotted for the VLA antenna will be slightly higher. **Make sure that all antennas and IFs are included!**

C.9.2 Phased VLA

The VLA may be phased on a program source ('STRONG'), or may be phased on a phase-reference source ('CAL-4'), with the resulting solutions applied to the program source ('WEAK'). Rather than recording a system temperature, the VLA system will record a ratio of antenna temperature to system temperature, which will vary as the array phases up. In order to convert the ratio of antenna and system temperatures to a usable gain, the flux density of some source will be needed.

1. Load and calibrate the VLA data by standard means (see Chapter 4 of the *AIPS CookBook*). Determine the flux density of a relevant strong source, usually either 'STRONG' or 'CAL-4'. Then, on the VLBI data set, insert the flux density of this source into the `SU` table using `SETJY`. For example, if the source is 'CAL-4' and its flux density is 0.432 Jy, run `SETJY` with `SOURCES = 'CAL-4'; BIF = 0; EIF = 0; ZEROSP = 0.432,0; OPTYPE = ' '`.

2. Edit the input file as indicated above for a single VLA antenna. Again, an `INDEX` line, a `GAIN` line, and a `TSYS` line must be created or uncommented. The `GAIN` line is independent of observing band (the source flux is used to determine the gain), and the `TSYS` line should include the parameter `'SRC/SYS'`, indicating that the ratio of antenna temperature to system temperature is being supplied.

3. Run `ANTAB` to read in the input file of amplitude calibration parameters. Then run `APCAL` to put this in `SN` 1. Both steps are essentially the same as for a single VLA antenna (see § C.9.1). The most likely problem is that `APCAL` will fail because you forgot to enter a source flux density using `SETJY`, although the error message may not always make this obvious.

4. Run `SNPLT` to inspect `SN` 1, as for the single VLA antenna. In this instance, you should see that the phased VLA is very sensitive. If the phasing worked well at centimeter wavelengths, the amplitude should be near 4 or 5 instead of the value of 17 or 18 seen for a single VLBA antenna. At the start of scans where the VLA is being phased, you may see a rapid change in the amplitude gain (toward smaller numbers) as the antenna phases are brought into alignment. The `SN` table should be inspected very carefully, because there may be data that should be flagged when the VLA phasing did not work well. Three possible reasons for poor phasing are (1) the source is too weak; (2) the troposphere is misbehaving; or (3) there was radio-frequency interference at the VLA.

C.9.3 Summary

Following the insertion of the amplitude solutions for the VLA, one can return to follow the standard path for calibration of VLBA data. Although the procedures from here on are identical to the VLBA-only case, the observer may wish to pay attention to several issues.

1. The phased VLA is far more sensitive than a single VLBA antenna, so it is often a good idea to use the phased VLA as the reference antenna for fringe-fitting.

2. The phased VLA has a large delay offset which should have been taken into account by use of a GPS file during correlation. Still, the user should pay close attention to the fringe fits, and be aware of the possibility that the VLA may have larger residual delays and rates than a VLBA antenna.

3. The VLA does not slew as rapidly as the VLBA. The `FG` table supplied by calibration transfer includes back-end flags only, and does not incorporate information about the pointing of the VLA antennas, and when they arrive on source. Therefore, some judicious flagging by the user may be necessary.

4. The VLA elevation limit is 8°, while the VLBA antennas can go much lower. This means that a source may set at the VLA well before it sets at Pie Town or Los Alamos, for example.

5. The VLA observing time is allocated in Local Sidereal Time rather than UTC. Therefore, it may start or finish observing as much as 15 or 20 minutes before/after the VLBA, even if the same amount of time is allocated.

C.10 Applying Global Ionosphere Models

This addendum describes (briefly) a method for using global models to correct for delays imposed by the Earth's ionosphere.

1. Outside of *AIPS*, change directory in some window to your `$FITS` area, *e.g.*, `/DATA/LAGUNA_1/FITS` for data reduction on workstation 'laguna'.
2. Invoke anonymous ftp to `cddisa.gsfc.nasa.gov`.
3. `cd /gps/products/ionex/2000/055` (for observation on 2000 February 24).
4. `get jplg0550.00i.Z JPLG0550.00I.Z` (get the JPL global model).
5. After quitting from ftp, `uncompress JPLG0550.00I.Z`.
6. Run `TECOR` with inputs `INFILE = 'FITS:JPLG0550.00I'; APARM = 0; GAINVER = 4; GAINUSE = 5`.

Currently, `TECOR` requires a data set that covers the entire length of your `CL` table, so it will fail if the observation goes across UT midnight. Suppose the observation goes across midnight on February 24. In this case, you must also `get jplg0560.00i.Z` and uncompress it. Then two model files must be concatenated in time order and edited by hand. This is painful, with a sample described below:

1. Edit the header of `JPLG0550.00I` to change the date of the last map from 24 February to 25 February.
2. Change the number of maps given in that header from 12 to 24.
3. Delete the header from `JPLG0560.00I`.
4. In `JPLG0560.00I`, add 12 to the numbers of all entries of type "END OF TEC MAP," "START OF TEC MAP," "END OF RMS MAP," and "START OF RMS MAP." For example, "5 END OF TEC MAP" becomes "17 END OF TEC MAP".
5. `cat JPLG0550.00I JPLG0560.00I > IONO5556.00I`.
6. Edit `IONO5556.00I` to move TEC MAPs 13 through 24 ahead of RMS MAPs 1 through 12.
7. Then, run `TECOR` as above with `INFILE = 'FITS:IONO5556.00I'`.

This will all make more sense when you actually look at the contents of the file! A sample concatenated file is available from the author, on request.

C.11 Some Useful References

1. Chatterjee, S., "Recipes for Low Frequency VLBI Phase-referencing and GPS Ionospheric Correction," VLBA Scientific Memo No. 22, May 1999. http://www.aoc.nrao.edu/vlba/html/MEMOS/scimemos.html
2. NRAO, *AIPS CookBook*, 31-Dec-2001 and earlier. http://www.cv.nrao.edu/aips/aipsdoc.html
3. Ulvestad, Jim, "VLBA Calibration Transfer with External Telescopes, Version 1.1," VLBA Operations Memo No. 34, July 30, 1999. http://www.aoc.nrao.edu/vlba/html/OBSERVING
4. Ulvestad, Jim, "A Step-By-Step Recipe for VLBA Data Calibration in AIPS, Version 1.3" VLBA Scientific Memo No. 25 (the basis of this appendix), January 2, 2001. http://www.aoc.nrao.edu/vlba/html/MEMOS/scimemos.html

5. Wrobel, J. M., Walker, R. C., Benson, J. M., & Beasley, A. J., "Strategies for Phase Referencing with the VLBA," VLBA Scientific Memo No. 24, June 2000.
http://www.aoc.nrao.edu/vlba/html/MEMOS/scimemos.html

C.12 Additional recipes

C.12.1 Cream of banana soup

1. Cook 1 quart green **banana** pulp, 1 1/2 quarts **chicken stock**, 1 small **celery stalk**, 1/2 **onion**, 1 **carrot**, 1 small **bay leaf**, 5 **peppercorns**, and **salt** to taste together for about 30 minutes until the mixture thickens.
2. Strain over 1/4 cup **flour** and 1/4 cup **butter** which have been combined as for a white sauce. Cook until thickened.
3. Just before serving, add 2 cups **cream** or **milk** and heat.

C.12.2 Banana curried chicken

1. Fry 2 chopped **onions** in 50 ml **cooking oil** until light brown.
2. Add 1/4 cup **cake flour** and mix well. Add 1 (cup?) **chicken stock** gradually while stirring.
3. Add 1 cup **raisins**, 1 teaspoon **salt**, 2 pounds cooked, boned **chicken**, 5 sliced **bananas**, 2 grated **apples**, 2 tablespoons grated **lemon rind**, 1 tablespoon **sugar**, 1 1/2 tablespoons **curry powder**, 1 **bay leaf**, 4 **peppercorns**.
4. Cover saucepan and simmer for 20 minutes.
5. Remove bay leaf. Add 1 cup **cream** and heat just before serving.
6. Serve on a bed of rice. Decorate with pineapples if preferred.

Thanks to Turbana Corporation (`www.turbana.com`).

C.12.3 Banana July cocktail

1. Sprinkle 3 sliced **bananas** with 1 tablespoon **lemon juice**.
2. Mix with 1 1/4 cans drained and flaked **tuna**, 1/2 **onion** chopped, and 2 tablespoons chopped **gherkins** or **olives**. Spoon into 7 **cocktail shells**.
3. Melt 2 tablespoons **butter** in a saucepan. Add 2 tablespoon **cake flour** and salt and pepper to taste.
4. Add 1/4 cup **chicken stock** and 1/4 cup dry **white wine**. Simmer for one minute stirring constantly.
5. Add 1/3 cup grated **cheddar cheese** and allow to cool.
6. Add 1/4 cup fresh cream to sauce and pour over banana-tuna mixture.
7. Sprinkle with 1 tablespoon grated **cheese** and **paprika**. Decorate with a slice of **gherkin** pr **olive**.
8. Bake 15–20 minutes at 350° F; serve warm.

Thanks to Turbana Corporation (`www.turbana.com`).