AIPSLETTER

Volume XVIII, Number 1: April 15, 1998

A newsletter for users of the NRAO
Astronomical Image Processing System

Written by a cast of *AIPS*

Edited by
Eric W. Greisen
National Radio Astronomy Observatory
520 Edgemont Road, Charlottesville, VA 22903–2475
804–296–0211; Fax 804–296–0278
`aipsmail@nrao.edu`

General developments in *AIPS*

Current and next release

The April 15, 1998 release of Classic *AIPS* is now available. It may be obtained via *anonymous* ftp or by contacting Ernie Allen at the address given in the masthead. `15APR98` should be available on CDrom as well as the more traditional tape media. *AIPS* is now copyright ©1995 through 1998 by Associated Universities, Inc., NRAO's parent corporation, but may be made freely available under the terms of the Free Software Foundation's General Public License (GPL). This means that User Agreements are no longer required, that *AIPS* may be obtained via anonymous ftp without contacting NRAO, and that the software may be redistributed (and/or modified), under certain conditions. The full text of the GPL can be found in the `15JUL95` *AIPSLetter*. Details on how to obtain *AIPS* under the new licensing system appear later in this *AIPSLetter*.

The next release of *AIPS* will be `15OCT98`. It is possible to get early access to, and remain current with, this release by running a "midnight job"; see the *AIPS* home page for further details. Note that this allows your site to receive the latest improvements and bug fixes, at the cost of also receiving the latest bugs. The latter can and will be fixed as rapidly as possible when the programmers are notified of them at `daip@nrao.edu`. The `15OCT98` release already contains a new version of `XAS` which will emulate an IIS Model 70 doing full color displays on those terminals which support "24-bit TrueColor" X-Windows visuals. This enables stunning overlays of multi-spectral images, interactive hue-intensity displays, roam and other split-screen algorithms, and more. The ability to have more than one TV on a single TV-host is being developed as is the ability to adapt the screen-full function of print routines to the current size of the workstation window.

Staff changes

Computer management at the NRAO has been restructured. The Assistant Director for Computing position has been abolished and Richard Simon has undertaken challenging new duties in the Millimeter Array Project. Ruth Milner has been appointed Assistant to the Director — Computing and will be guided by a Council chaired by Alan Bridle and containing the four site computing Heads plus representatives of the Classic *AIPS* and AIPS++ projects. In November, Gareth Hunt became Division Head for Green Bank Computing (including co-ordinating all GBT-related software projects) and, this April, Pat Murphy replaced him as Division Head for Charlottesville Computing. This position should leave Pat at least half time to spend on Classic *AIPS*.

Improvements of interest to users in `15APR98`

This release began as the Charlottesville experimental version of *AIPS*, becoming the official `TST` version by November 1997. It incorporates essentially all of the changes described in previous *AIPSLetter*s plus numerous improvements over the previous releases. See the `CHANGE.DOC` file, available selectively or as a whole from the Classic *AIPS* WWW page, for more details.

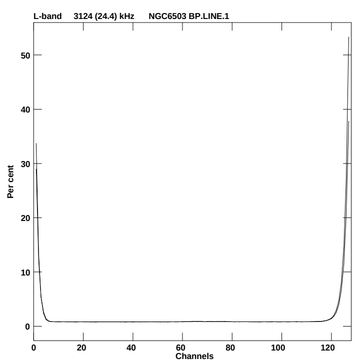
`15APR98` *introduces numerous changes which are not compatible with previous releases. Disk files written by previous versions are read transparently by* `15APR98` *(including* `SAVE/GET` *files), but users must not attempt to read disk files written by* `15APR98` *with earlier versions.* `15APR98 AIPS` *cannot start previous versions of tasks and the TV displays of the versions are incompatible.*

Interferometric data calibration

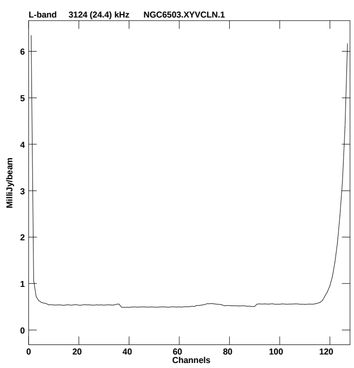
Bandpass calibration

Bandpass tables have been changed to include weights which depend on IF and polarization. `BPASS` will compute them and the calibration application routines (and `POSSM`) will use them. Older format bandpass tables will continue to function, with the missing weights taken from the "interval" column. The new application routines are able to do nearest neighbor and two-point interpolation on the bandpass table with weights (`DOBAND` 2 and 3, resp.) and without weights (`DOBAND` 4 and 5, resp.) as well as using the overall average bandpasses. Dynamic memory is used which means that all tasks will use as much or as little memory as needed. `CVEL` also works with all values of `DOBAND` to shift observations to a common velocity.

In determining bandpass calibration functions from the data, the algorithm used by `BPASS` was improved to correct data weights to take into account low fluxes in the continuum division. New adverbs and print controls were added to summarize closure failures which have been found to be important limits to the accuracy of the final images (see figure below for one VLA line observation). Adverbs `BCHAN`, `ECHAN`, and `STOKES`, which caused errors, were dropped from `BPASS`. Bugs affecting re-referencing of phases were corrected. If the solution for channel 1 failed, data would not be re-referenced despite changes to the reference antenna in the record. Dynamic memory is now used throughout to allow the program to run in the space needed, be it large or small. A new task (`BPERR`) was developed to read closure error reports from text files generated by `BPASS` and `PRTMSG` and to produce from them averaged and summed reports and plots.



Average amplitude bandpass closure failure



Non-signal rms spectrum of Cleaned image cube

`BPSMO` is a new task which makes a regularly gridded (in time) bandpass table via several different interpolation schemes. It has an option to insure amplitude normalization over a range of channels and does all interpolation and normalization in either full vector or amplitude-scalar modes. It can do weighted or un-weighted smoothing and offers options affecting the output weights. `BPSMO` may also be used just to fill in blanked (*i.e.*, failed or flagged) solutions leaving the others alone. `BPLOT` is a new task to plot bandpasses as profiles in two dimensions. Multiple times for one antenna or multiple antennas for one time appear in each plot and multiple plots can be produced. Multiple IFs and/or polarizations may appear along the horizontal axis.

Data editing

EDITA is a new task that allows you to prepare *uv*-data flag commands interactively from displays of TY (system temperature), SN (gain solution), and CL (calibration) tables. It is similar to SNEDT (without the smoothing options), but prepares Flag Commands which can then be copied into uv-data FG (flag) tables. It applies a pre-existing FG table to the TY, SN, or CL data as they are read so that you do not need to flag those data more than once. EDITA talks in user-friendly units for delay and rate although it uses sec and sec/sec internally to match the units of the calibration files. Up to 10 antennas may be displayed on the TV for comparison purposes although only one antenna is edited at a time. The flags generated may be made to apply to all antennas and the clip operations may be made to run over all antennas. The ability to flag system temperatures that deviate by excessive amounts from the running mean has been found to be particularly effective in flagging VLA data.

EDITR is a new task that allows you to prepare *uv*-data flag commands interactively from displays of the *uv* data and, optionally, a related *uv* data set (*e.g.*, a residual data set from IMAGR, SCMAP, or UVSUB). It is a graphics-based editor like EDITA and (distantly) IBLED. It allows you to look at up to 10 baselines to the selected antenna at the same time and to view and edit upon amplitude and phase of the data and of the difference between the data and a running vector mean of the data. Since EDITR is actually an object-based class, it has also been inserted inside SCMAP, the iterative self-calibration and imaging task.

UVMLN is a new task to apply preliminary calibrations (continuum and, especially, bandpass) to a multi-source file, then fit a baseline to each record (*uv* spectrum), and generate flags whenever residuals in the baseline-fitting regions exceed specified limits. This will let you remove bad samples, and re-determine the calibration with cleaner data.

UVCOP can now apply a flag table, deleting data, while doing the copy. All TV-based editors (TVFLG, SPFLG, EDITR, EDITA) allow you to enter a REASON string for subsequent flag commands. Problems with gridding data in IBLED, TVFLG and SPFLG have been addressed. These caused data to appear at slightly wrong times and could mix data from two sources when doing the running means.

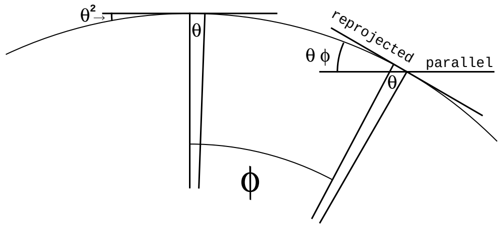
Miscellaneous calibration improvements

In CALIB, the adverbs MINAMPER and MINPHSER were added and the meanings of APARM(6) (print control), CPARM(3) and CPARM(4) were changed to allow more complete examination of closure errors without too much printing.

The old ASCAL task, which ignored IFs > 1, has been deleted. Its supporting cast (VSCAL, ASCOR, GNSMO, GNMRG, GAPLT, GNPLT, PRTGA) has also been removed.

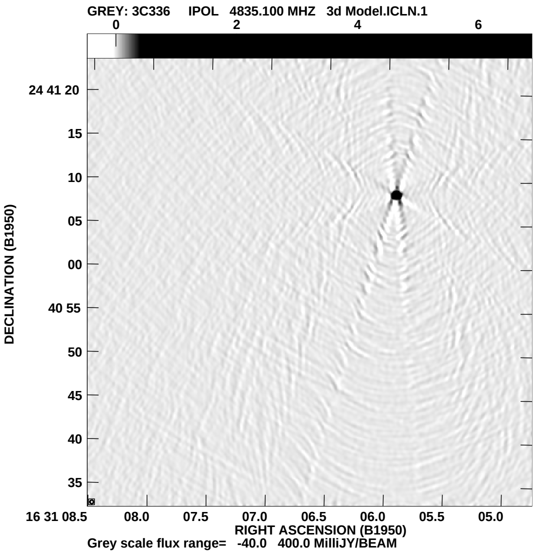
Interferometric imaging

Non-coplanar imaging

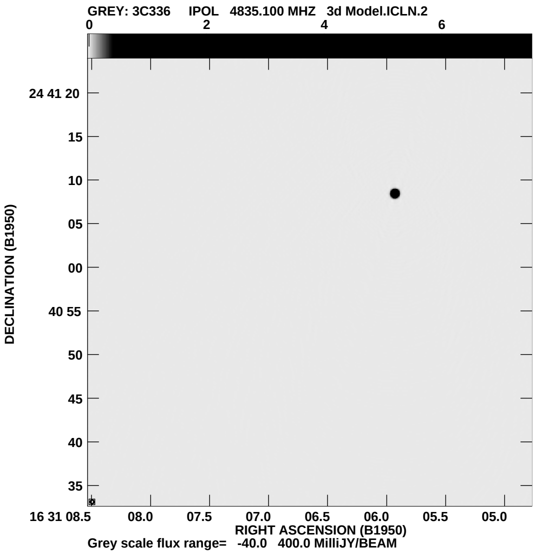


AIPS has long supported the imaging and deconvolution of ≤ 16 fields surrounding the direction toward which the telescopes were pointed. Among other things, this allows for removal of sidelobes due to sources distant from the field of primary interest. Previously, the phases of the data were rotated to the center of each field before imaging, but no other geometric corrections were made. This makes each field parallel to the tangent plane at the antenna pointing direction. Such planes separate from the celestial sphere very much more rapidly with angle from their center than would a tangent plane; see figure above.

For the `15APR98` release, `IMAGR` has been changed to offer the option (adverb `DO3DIMAG`) of re-projecting the interferometer baselines onto the center of each field. This makes each field tangent to the celestial sphere, as illustrated in the figure above. This allows even distant fields to remain "in focus" over much wider areas; previously fields often needed for VLA P-band images were in focus over an area less than the synthesized beam. Although this option requires matrix multiplications for each visibility for each Clean component, it is surprisingly inexpensive. It costs only about 1 per cent of the cpu when the option is not really needed and can greatly speed convergence in cases where it really is useful. The two Cleaned images below illustrate the effect of this option. The source is a simple Gaussian model object located 2 degrees from the pointing direction and 14.9 arcsec (36 synthesized beams) from the field center. The non-signal "rms" of the uncorrected image is 35 times that of the corrected one.



Cleaned model without 3d-reprojection



Cleaned model with 3d-reprojection

All of `15APR98` *AIPS* now understands these images, handling Clean-component models made with or without the re-projection. Thus `CALIB`, `UVSUB`, `FRING`, `BPASS`, `PCAL`, `VPLOT`, *et al.* can handle such models. `FLATN` is a new task to create a single (large) output image by re-projecting all fields onto a single coordinate grid.

Other improvements in imaging

`IMAGR` was also changed to deal with overlapping fields more sensibly. Previously, Clean components found in one field were restored only to that field even if the object was present in more than one of the fields. In `15APR98`, you may request that all Clean components be restored to all fields in which they occur by setting adverb `OVERLAP` greater than zero. An even more insidious problem can arise if the same source is included in the Clean boxes in two separate fields. Previously, the object would then be Cleaned twice, appear in the next cycle as a deep negative, and so on, slowing the convergence enormously. By setting `OVERLAP = 2`, and `DO3DIMAG = TRUE`, you may now request a new mode of Cleaning that follows a somewhat different path. At the beginning, the end, and whenever instructed by the user, `IMAGR` makes an image of all fields using the current residual *uv* data. Otherwise, it makes an image of one field, does one major cycle on that field, subtracts the new Clean components from the *uv* residual data and then repeats the process with another field. The TV display makes it clear that multiple fields are accessible during that TV display session and it is important to set Clean windows when they are accessible since the selection of fields to image depends on the extrema in the Clean windows only. When only one field is accessible, the field number is displayed in the menu and the option to re-image all fields is offered. That option will cause the next TV display to have all fields accessible. It turns out that this new schema aids not only with overlapped objects but can be important in preventing distant sidelobes of strong objects from being picked up as sources in the weaker fields during the early stages of Cleaning.

The maximum number of fields for imaging was changed from 16 to 64. This changed the dimension of adverbs **FLDSIZE**, **RASHIFT**, **DECSHIFT**, **BCOMP**, and **NCOMP**. It also forced changes in system I/O structures and in the Task-Data intercommunication file (**IMAGR** now requires 782 words of adverb values!). **IMAGR** will read in field parameters from the adverbs and also from the optional **BOXFILE**. This file can save problems typing in large numbers of adverbs in complex situations. Field centers can even be given in standard Celestial coordinates. The maximum number of Clean boxes was reduced from 500 to 256 per field. **IMAGR**'s work file may be re-used, although the data in it are never re-used. An **ABORT-TASK** option was added to the interactive TV menu.

SCMAP is the task used to make images using iterative deconvolution and self-calibration steps. It was revised to allow the full capability of **EDITR** to be selected from the self-calibration TV menu. **SCMAP** now compares with **DIFMAP** for VLB and even VLA imaging. It has acquired interactive options to abort the task, to switch to amplitude and phase solutions from phase-only solutions, and to reset a wide range of parameters. When doing amplitude and phase solutions, it can apply a time smoothing to the solution amplitudes before applying them to the data. This smoothing is also used to interpolate over failed amplitude and phase solutions in the hope that they won't fail in the next iteration.

UVSUB now has an option to write out the model evaluated at the input sample points instead of the difference or ratio of the observations and model. **VPLOT**, **CLPLT**, and **IBLED** compute multi-field models correctly in **15APR98** and support the 3D imaging option. **VPLOT** offers the option of computing the model only at the data points which is more accurate than computing it at predicted values of u, v, w versus time. **CLPLT** now reads the data correctly.

Single-dish imaging

The circular convolving functions used in single-dish imaging were found to have an addressing error. The effect was to move the object one-half pixel toward decreasing X and toward increasing Y. This bug was corrected December 19, 1996 in the CVX version of *AIPS*, but remained in the official versions until **15APR98**. Another bug causing bad values of circular convolving functions to be used was found and fixed 6 January 1998. This bug should either have used 0 (nearly the right value anyway) or really bad values producing obvious effects in the output images.

Spectral-line on-the-fly images

OTFUV translates NRAO 12m "on-the-fly" spectral-line data into *AIPS* format. An option to interpolate the OFF and Gain measurements in time before applying them to the ON data was added. In addition, the ability to read up to 8 "IF"s at a time (from **BIF** through **EIF** but excluding those that do not match **BIF**'s frequency center and increment) was added. This option avoids the later need for a sort. Lower-case file names are now allowed. A new task, **OTFIN**, was written to print an index of the contents of a 12m spectral-line or continuum "on-the-fly" data file.

SDLSF is a new task to average all spectra observed at the same time, fit a linear baseline to the average, and then subtract that baseline from each spectrum. This is an attempt to remove the "weather" before imaging, but its main effect may be to avoid the **TRANS-IMLIN-TRANS** process used after imaging to remove the spectral baseline. For multi-feed instruments dominated by instrumental effects, the option to remove a baseline from each spectrum individually is included. Options to flag data with excessive discrepancies in the baseline regions are also available.

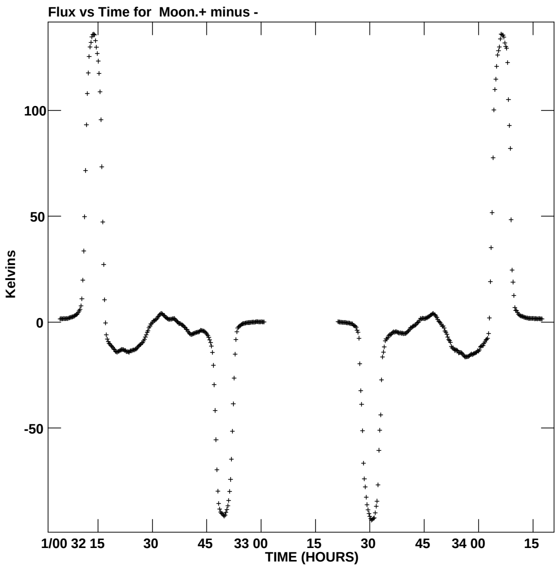
Using the 12m to observe narrow lines, users stumbled across a fairly obvious (in hindsight) effect. The definition of "velocity" in a spectrum is a function of position and observing time. For reasons of calibration stability, one observes at a fixed frequency for some time and then shifts to a new frequency. For interferometer data, in which all times and positions are mingled together in the synthesis process, there is very little one can do about this problem. But, for single-dish spectra, the new task **SDVEL** may be used to shift each spectrum to the correct velocity for the actual time and pointing position. The velocity error in a 2x2 degree field can be as much as 1.2 km/sec which is significant at mm wavelengths. The new task **VTEST** is used to help determine how big this effect can be.

The single-dish spectral-line imaging task `SDGRD` was renamed `SDIMG` and a new `SDGRD` was written to be an OOP task. It is now limited to images small enough to fit one plane into the pseudo AP, but it does not require the, potentially, very large scratch file still used by `SDIMG`. The cpu time may be longer than `SDIMG` since the coordinate projection has to be repeated for each set of spectral channels, but the real time is usually less.

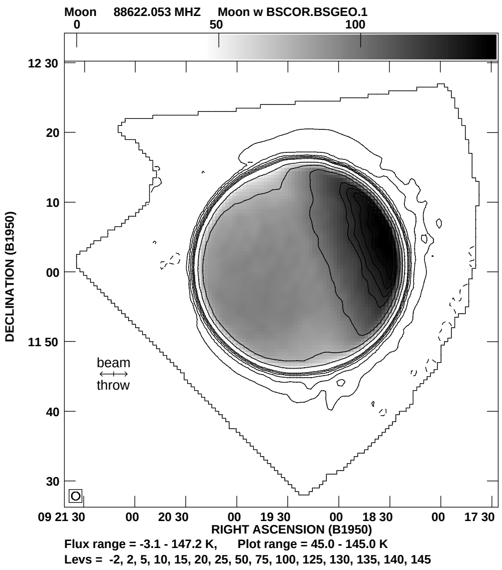
`WTSUM`, the task used to do weighted averages or sums of images using images of the weights, has been given the option to average a large number of input images in a single execution, looping over the image "sequence" number.

Continuum beam-switched imaging

A package of tasks to handle beam-switched on-the-fly continuum imaging from the NRAO 12m has been developed. Below are figures illustrating observations of a large object with a modest beam throw.



Two scans on the Moon of plus - minus data `DIFUV`



Final image with reprojection `BSGRD`

The package contains

OTFBS A new task to read beam-switched continuum data and make two *AIPS* *uv* data sets, one for "plus" throws and one for "minus."

BSGRD A new task to make a beam-switched image in standard coordinates by imaging the plus and minus throws, rotating the two images, correcting them by the standard convolutional algorithm, and then regridding into standard coordinates. It combines `SDGRD`, `OGEOM`, `BSCOR`, and `BSGEO`.

BSCOR A new task to combine images made with plus and minus beam switching using the "standard" (Emerson *et al.*) convolutional algorithm. It is now coded as a class method in the *AIPS* OOP system.

BSGEO A new task to regrid the output of `BSCOR` into a standard right ascension by declination image. It is coded in OOP.

BSAVG A new task to average multiple BS images using a weighting in the Fourier transform space.

BSTST A new task to test and plot one-dimensional beam-switched restoration (really frequency-switched in 1-D). Options to include various instrumental errors are available.

BSFIX A new task to compute and correct the RA and declination offsets in beam-switched data.

DIFUV A new task to difference similar *uv* data sets. It is used here to examine the difference of the plus and minus data sets for purposes of display and editing.

SDMOD This task subtracts a model from single-dish data or replaces the data with a model. Models are up to 4 Gaussians or, now, an image. Beam-switched data may be modeled including instrumental errors such as throw length.

VLBI data processing

KRING

KRING is a new, experimental fringe-fitting task intended to be substantially more convenient than FRING. It creates scratch files one scan at a time so that, if the data set is divided into a reasonable number of scans, the scratch space required on disk is significantly reduced. KRING is also more parsimonious with memory than FRING; this allows longer solution intervals and/or larger delay and rate search windows.. KRING should almost never run out of memory, although it may happily bring your machine to a halt before it gives up for lack of memory. Currently, KRING should yield the same results as FRING, but features will be added as time and user demand dictates. KRING is almost a complete rewrite of FRING and, as such, may contain bugs; users are encouraged to check their results carefully. Please send comments and suggestions to kdesai@nrao.edu.

OMFIT

The *uv* model-fitting task OMFIT has had some significant improvements: the help file has been improved, the code has been tested on a variety of data sets, new models have been added, data excision is now possible, and the output options have been improved.

The input adverbs have been reorganized and simplified considerably. Data flagging based upon RMS-residuals and low-SNR threshold is now available. User estimates of the *a priori* visibility noise can now be entered via the input adverbs, allowing proper Chi-Squared and error-bar calculations. A simple perl script for converting between OMFIT-style and DIFMAP-style model files is available from the web [http://www.nrao.edu/ kdesai/files/d2a.pl]. This script currently only understands point sources and Gaussians between the two formats but could easily be extended to all available DIFMAP model types if requested. This script will be available as part of the 15OCT98 *AIPS* distribution.

The convergence criteria have been improved and, under the best of conditions, OMFIT converges within 4-6 iterations. The error bar analysis has been reworked and tested for consistency using fake VLBI data as well as real VLBI and VLA data.

The adverb OUTPRINT can now be used to write out model residuals to a text file. Both self-calibrated and un-self-calibrated model residuals are available in the output text file. Note that OMFIT can self-calibrate the data while model-fitting, a capability not available elsewhere in *AIPS* or in DIFMAP. The OUTFILE now contains most of the input adverbs for accounting purposes. Optionally, model component information can be written to the OUTFILE at each iteration for later examination; this can be useful for checking model convergence in individual parameters. Like DIFMAP, the output file written using OUTFILE can be used as an INFILE to restart OMFIT.

Three new models are available for modeling Zeeman data, maser data, and polarized point sources. The Zeeman model type allows for both circular and linear polarization and accommodates Gaussian and Lorentzian spectral profile models. The maser model type incorporates maser features with simple Gaussian spectral profiles. The polarized point source model type allows the introduction of polarized point sources.

OMFIT has no built-in limit on the number of visibilities or model components, but extremely large data sets ($> 100,000$ vis) and/or extremely complicated models (> 10–15 components) do increase the program execution time considerably. OMFIT now uses the double-precision LAPACK/BLAS code which was not found to increase execution time appreciably. Planned future changes to OMFIT are listed in the source code ($QPGOOP/OMFIT.FOR just past the LOCAL INCLUDEs section). Please direct comments and questions to kdesai@nrao.edu.

Varied matters

VLBI

INDXR was given the capacity for merging atmospheric delay and clock offset information from VLBA model-components (MC) tables into newly created CL tables when possible. Removed assumptions that all access to IM and MC tables was in strict time order, which it cannot be when there are subarrays. A text file may be supplied that specifies times at which scan boundaries must occur.

UVFIX was enhanced to handle orbiting antennas using polynomials fit through orbit parameters in the OB table. The correction for differential aberration, required for precise wide-field astrometry, had been done outside of AIPS using a routine written by Ed Fomalont. The correction was done for each component found in the image after the image had been obtained. The new version of UVFIX directly corrects the *uv* data, simplifying the correction dramatically.

M3TAR was submitted by Walter Alef to read Mk3 VLBI data in Unix tar archives. He also submitted task TFILE to sort and edit Mk3 AFILEs for M3TAR.

COHER is a new task to estimate coherence time by baseline for VLBI data. It is able to work even on non-zero fringe-rate data and on data of low signal-to-noise.

VLOG has been improved so that, for VLBA-only experiments, it will pick up the correct gain curves if the gains.key file is prepended to the cal.vlba. The user should check the L-band gain curves carefully as there are differences in the wavelength designations in the cal.vlba and gains.key files.

uv data

FILLM was changed to allow TY tables to have a smaller, user-controlled time increment than CL tables, to test for data out of time order, to avoid excessive creation of FQ numbers, and to allow weighting by system temperatures (which are probably still too inaccurate to use for this purpose).

DBCON was changed to apply all corrections to compressed as well as uncompressed data. The two data sets do not have to have the same compression state. Fixed the DOPOS phase correction to use frequency at all times and to do it correctly.

UJOIN is a new task to average or difference overlapped parts of multiple IFs, producing a new *uv* data set with one IF and, usually, more spectral channels.

FIXWT was changed to avoid deleting too much data and to handle short scans and breaks in the data more appropriately.

DTSUM was changed to handle single-dish data, single-source data, and missing tables usefully.

SPECR was changed to handle autocorrelation data and to be able to increase the number of channels on output correctly. It was changed to work correctly on compressed data.

FXTIM is a simple new task to correct times in *uv* data set if a wrong reference data was selected (leading to negative times).

UVSUB has a new option to replace the data with the model.

Imaging

OGEOM is a new task to rotate and re-scale the geometry of an image, writing a new image. It handles blanked pixels properly, allowing small blanked regions to be filled in rather than to increase as in LGEOM. OHGEO, which does a more complicated re-gridding, was also enhanced to give the user more controls.

SERCH uses algorithms developed by Juan Uson to convolve the spectra in a data cube with Gaussians of various widths and then report those pixels/channels having signal/noise exceeding a user selected cutoff. This is a good way to find weak signals in large cubes. The pixels found may be written out in a hyper-cube of signal-to-noise ratio.

COMB was changed to allow noise images to be input rather than simple constants. A noise image may be written at the same time as the usual product image. PANG has a new clip level on the total polarization; SPIX and OPTD also have new clip level parameters. POLC now uses Jim Condon's method covering the full range of signal-to-noise. Most prohibitions were dropped and several defaults changed.

POLCO was changed to do a low S/N solution for the correction as well as the previous, higher S/N correction. This allows noise in the corrected image to be estimated properly.

PBCOR was changed to offer the option of applying rather than removing the primary beam.

UVMOD and IMMOD were changed to offer additional source models and to correct the units of some parameters.

SAD and JMFIT and IMFIT were changed in an attempt to make them more reliable. In particular, the first two were sensitive to differences in the magnitude of the flux and coordinate scales. The search, retry, and rejection criteria were changed in SAD and the initial guess for the width is now almost always a Clean beam. All three can now write MF files.

Display

COSTAR is a new verb to plot a symbol at a user-specified coordinate on the TV display.

COWINDOW is a new verb to set BLC and TRC to be centered on a coordinate.

TVCPS was changed to honor the zoom and scroll so that "what you see is what you get" and was given an additional control, primarily for transparency paper, over the color of blanked and edge pixels.

UVPRT and PRTSD now use BCOUNT as an offset into the file rather than reading the first BCOUNT records. This makes them more predictable and much faster.

PRTUV and friends were changed to determine the range of data weights and to select an appropriate format for them.

PRTAB was changed to give much more control of what and how much is printed.

KNTR was given the option to have the step wedge over the full range of image intensity rather than just PIXRANGE.

IMRMS is a new task to plot spectra of the rms found by IMEAN and printed to a text file by PRTMSG.

The *CookBook*

The *AIPS CookBook* was modified for 15APR98. You can access the individual chapters (as PostScript files) using a version of the Table of Contents found through WWW starting at the Classic *AIPS* home page. The introductory chapters (1 and 2) had minor revisions made to update the code sizes and various 15APR98 related matters. The basic utilities chapter (3) was revised to correct the WWW address for 15APR98 and to update the description of HELP. The calibration chapter (4) was revised to describe the new and significant task EDITA, to update FILLM's use of system temperatures, to describe the closure reporting in CALIB, and to describe additions and changes to bandpass calibration techniques. The imaging chapter (5) was updated to describe added BOXFILE capabilities and the new "3D" and "overlapped" imaging and to describe EDITR rather than the old IBLED baseline-editor task. The VLBI chapter (9) was extensively modified to mention new tasks and to describe recent developments including Space VLBI. The chapter on single-dish data (10) has been modified to describe beam-switched continuum analysis and numerous other improvements to on-the-fly imaging in 15APR98. The list of *AIPS* symbols by type (13 = the ABOUT files) was brought up to date. The Index and Table of Contents have been modified to match these changes.

Improvements in system matters in April 15, 1998

System parameters

Handling large problems

The parameter `MAXCIF` has been raised to 16384. This is the maximum number of polarizations times IFs times spectral channels which can appear in a single visibility data record. All *AIPS* tasks should be able to handle such data. The number of extension file types allowed with a single catalog entry has been raised to 50. This should be enough to handle even the most extreme VLBI experiments. It is this change that causes the main problems in accessing `15APR98` data files with earlier releases of *AIPS*.

The `15APR98` release of *AIPS* fully supports data files larger than the traditional Unix limit of 2 Gbytes. It does this by a method that is much simpler than that used in `15OCT97` but which has a limit of 2048 Gbytes (on 32-bit computers). The code has been tested on DEC alphas (Digital Unix 4.0B), SGI (Irix 6), Sun (Solaris 2.6), and HP (HP-UX 10) computers, which are all of the computers we have available which provide operating system support for these large files. You should note, however, that file systems may need to be set up in a particular manner before they can be used to store these large files: consult your system manager to see what needs to be done.

Miscellaneous parameters

A translation of the environment variable `$HOST` was added to the internals of *AIPS*. This string is then used as the host name on printouts, accounting, and the like rather than the `SYSNAM` which is `UNREGISTERED!!!!!` on all machines that are not registered. This fix keeps two hosts from destroying each other's scratch files on shared data areas. (It helps, of course, if you do register your computers with us.) Similarly, the number of tape drives is now determined from an environment variable (`$NAIPST`) set by the *AIPS* startup procedures rather than depending on a hand-maintained parameter in the `SP` file.

Throughout *AIPS* there are various calls to `ZDELAY` to cause the program to wait for a while before attempting something, *e.g.*, to wait before checking to see if a task is still running. A host speed parameter has been added to the `SP` file to be used to scale these time delays. A three-second delay may be reasonable on an old IPX, but is an eternity an a new Ultra.

Two new computer architectures were added. They are called `SUL` for Sun Ultras and `HP2` for the latest in the Hewlett Packard line. These "new" architectures allow sites to have binary areas built with special options for the latest/fastest machines co-existing with binary areas for their older machines.

AIPS startup scripts

Two of the scripts used to start an *AIPS* session were changed in ways that may require *AIPS* Managers to modify their local files. The script that selects data disks for the user was changed so that the local host data areas are selected first, required areas second, and those selected via `DA=` on the command line last. Data area names must contain the host name terminated by an underscore or end-of-line. This allows automatic selection to take place while avoiding ambiguities between, for example, `HP10` and `HP107`. We recommend that you name all data areas with the form *something/host_n*, *e.g.*, `/AIPS/DATA/HP10_1`, `/AIPS/DATA/HP10_2`, etc. Please note that these do not have to be the actual names of the data areas; they can be symbolic links. They do have to be known to the automounters of any system which may need to use them.

When the user is sitting at one workstation, but starting `AIPS` on another, the startup scripts attempt to start the *AIPS* display servers (`XAS`, `MSGSRV`, `TEKSRV`, and `TVSERV`) on the remote workstation. This attempt is fraught with pitfalls. The main problem is that the remote shell command must have privilege to run and must somehow find the display servers on the remote machine. The privilege problem can be fixed with additions to the user's `.rhosts` file, but the latter has no easy solution. We have attempted to use the environment variable `$AIPS_ROOT` to find the needed procedures, but some Posix-compliant shells refuse to execute anything for a remote shell and, hence, have no environment variables set. Because of this, the startup script on the compute host must specify the absolute pathname for the startup scripts on the remote host. This may be set locally with the environment variable `$REMOTE_ROOT`, but if you wish to have your site be able to start *AIPS* display servers automatically from compute servers at other sites, you must then create an area known to all of your local computers as `/AIPS` which is, or points to, your local actual `$AIPS_ROOT`. Please note that this convention is required only for automatic startup of the servers. The user may start the servers by running `AIPS` on his desktop machine and these servers (if they use Internet sockets) are then available to *AIPS* sessions running on other computers, literally around the World.

Varied matters

AIPS tables may now be created with any version number (1 — 46655) rather than be required to increase sequentially by one.

OOP *uv* data may now be read without interference by the calibration software in a new "read-raw" mode.

***uv* data reads** may now begin at a specified visibility number (`INITVS`) even for multi-source data sets.

Year 2000

The "Year-2000" problem has been addressed fully in `15APR98`. The internal date string is now of the form `YYYYMMDD`, although headers containing the old form `DD/MM/YY` are read and corrected transparently. The FITS reading tasks are all capable of reading both the old and the new FITS data form `YYYY-MM-DD` and `FITTP` is poised to write it beginning on January 1, 1999. Quite a number of places in *AIPS* were changed for aesthetic reasons to handle a full 4-digit year and a few were changed to consider years less than some cutoff (*e.g.*, 40) as year 2000 based rather than year 1900 based. Only one routine handling Julian dates (`JD2DAT`) was found to be in error for dates after 2000.

TV display

The TV display server `XAS` and the routines that talk to it were changed substantially for the `15APR98` release. The I/O interface was made much simpler, so that knowledge about the information content and format of the data records would be found only in `XAS` and in the corresponding `Y` routines. Previously, the I/O `Z` routines were required to know formats and do conversions of data words to/from network forms. It is mostly this change which makes the `15APR98` TV incompatible with previous releases. Having simplified the interface, it became easy to add capabilities. Initialization, character generation, vector drawing, and area fill are now done inside `XAS` with a minimum of I/O to the invoking program. The TV was enlarged to have 4 grey-scale memories to allow for better movies and later developments such as true color and split screen (roam) which will appear in the `15OCT98` release. There is also a new icon to distinguish this version of `XAS` from previous ones.

AIPS' displays are stateful — they remember information about the images displayed on them and about what portions of the display are currently visible. In the past, we have depended on disk files to maintain this information. Unfortunately, these files had to be mounted to all computers that might want to know about the display on a particular computer. This mounting and locking posed a significant maintenance problem at large sites and led to erratic behavior at best when the displays were accessed between sites (or byte-swapping orders within one location). `15APR98` institutes a new system in which `XAS` (and the Tektronix emulator as well) maintain their own device characteristics data array and their own image catalog. They do this in a device-independent fashion so that any computer may determine the state of any `XAS` display to which it can talk. The disk files are no longer needed. Also, the image catalog information reflects the current TV, not one that was running yesterday; `TVINIT` is not needed to re-initialize the image catalog.

It turns out that the Internet is too smart for our own good; it does not let one program know that another is currently using a particular `XAS` display. To avoid semi-infinite hangups, a TV-lock server has been added to the `15APR98` release. This requires another *AIPS* system service to be created, but does an efficient job of device locking without all the problems associated with NFS file locking of `ID` files in the past.

Parallel processing

The 15APR98 release of *AIPS* contains some support for parallel processing on multiprocessor machines. The initial efforts at parallel processing are directed towards the SGI environment. In future releases of *AIPS* we will expand both the range of platforms that can be used for parallel processing and the number of tasks that can benefit from it. The effort spent in this area will also benefit users with single-processor systems since the modified versions of the Q-routines are often better suited to modern workstation architectures than the original vector code; note that the run-time performance of the parallel code is almost twice that of the original vector code. The techniques used to write parallel code in *AIPS* will be described in a forthcoming *AIPS* Memo. Several of the pseudo-AP Q-routines can be rewritten to favour parallel execution with significant gains in performance as shown in the following table which shows the time taken by CALIB on the large DDT problem with a parallel version of QPTDIV running on an SGI Origin 200 with four processors. CALIB will rarely show this level of improvement in normal use since the parallel version of QPTDIV is still much slower than the gridded model division that would normally be used in this case.

Version	Processors	CPU Time seconds	Real Time seconds	Speedup
Original	1	613.8	618	0.51
Parallel	1	311.6	313	1.00
Parallel	2	160.9	163	1.93
Parallel	3	110.3	112	2.80
Parallel	4	85.5	86	3.65

The routine that performs the coarse search in FRING has also been adapted for parallel processing in *AIPS*. As this routine is responsible for most of the time spent in FRING this can be a significant gain for VLBI users. Some typical timings from a VLBA data set are given here. The gains from using multiple processors are greater for larger numbers of spectral channels and for longer solution intervals (4-processor speedups of up to 2.44 have been seen for space VLBI data).

Version	Processors	CPU Time seconds	Real Time seconds	Speedup
Original	1	122.1	124	0.54
Parallel	1	66.2	67	1.00
Parallel	2	44.5	45	1.49
Parallel	3	38.3	39	1.72
Parallel	4	34.3	36	1.86

The benefits of using multiple processors depend on the size of the problem with smaller problems gaining the least and saturating with a smaller number of processors (note that using 4 processors is not much better than using 3 for the FRING example above). In extreme cases, using more processors may result in a program running more slowly. For this reason, *AIPS* allows the user to choose how many processors will be used by a parallel task. The PARALLEL pseudoverb takes an integer argument that is taken to be the number of processors used by any tasks that are started subsequently. A user might, for example, type PARALLEL 3 before running the FRING problem shown above, leaving one processor free to run other programs.

AIPS on CDrom

Starting with this version, we expect to make *AIPS* available on CDrom on a limited basis. The initial tests using recordable CD's were very successful, and resulted in a CD with source code, two binary versions (Linux and Solaris), and a GNU-zipped version of the documentation (TEXT) area. It was possible to either perform a full installation on disk (*i.e.*, copying the binaries from CD to local disk), or to run from the CD. In the latter case, the "footprint" on the local disk was under 10 Megabytes! (This figure does not include user data, obviously.) Furthermore, the setup script was given the ability to switch between a "run from CD" installation and a "full" installation. It is hoped that this functionality, and the availability of *AIPS* on this new medium, will be of considerable use to the Astronomical Community. Distributing *AIPS* on CDrom instead of tapes should end up as a per copy cost savings to NRAO as well, given the low cost (under $2) of each CD blank.

AIPS Distribution

A total of 107 copies of the `15OCT97` release were distributed, of which 53 were in source code form and 54 were distributed as binary executables. The table below shows the breakdown of how these copies were distributed. This includes both source code distributions and binary distributions.

ftp	8mm	4mm	ZIP	Floppy
101	3	2	1	0

User feedback suggests that the distribution over operating systems for installed versions of `15OCT97` was:

Operating System	No.	`15OCT97` %	`15APR97` %	`15OCT96` %
Solaris/SunOS 5	101	50	66	46
PC Linux	47	23	16	19
HP-UX	6	3	6	4
Dec Alpha	19	9	6	10
SunOS 4	28	14	5	13
SGI	1	1	1	5
IBM /AIX	0	0	0	4
Total	202			

The distribution of `15OCT97` has been rather lower than that for `15OCT96` (222 copies) and even `15APR97` (148 copies), perhaps reflecting the lower rate of developments in these releases. The figures on computers using *AIPS* are affected by the percentage of *AIPS* users that register with NRAO. Of 95 non-NRAO sites receiving `15OCT97` only 36 have registered. We remind serious *AIPS* users that registration is required in order to receive user support.

DDT or How to speed up your *AIPS*

The performance of *AIPS* on a number of platforms has been measured over the past few months. The results are summarized in a table below. However, it is important to point out a number of things we learned about improving performance while we did the tests. For users of personal computers under Linux, we have been using `f2c` followed by `gcc` to compile *AIPS*. Previous attempts to use the GNU `g77` Fortran compiler had failed, but a concerted new attempt led to the discovery of the compiler directives needed to make *AIPS* work, and work remarkably well, on Intel boxes. The main cause of error was the inadequate use of the `SAVE` statement in *AIPS* code. Almost all compilers pre-allocate all variables rather than making them dynamic as is allowed by the Fortran standard. In the former case, variables retain their values between calls to a routine, but in the latter they do not. GNU chose to make the default in their compiler dynamic allocation, but they fortunately offer a directive to make the variable allocation static. With the addition of some optimization directives as well, the `g77` compiler produced code nearly twice as fast as the `f2c/gcc` code! We tried `g77` on a Sun Ultra computer, but the resulting code ran 50–80 percent slower than when compiled with Sun's Fortran compiler. A "trick" that does work on Solaris boxes was suggested by Sun engineers. If one assigns an *AIPS* disk area to `/tmp`, the memory/swap file system, and tells *AIPS* to use that disk only for scratch files, then performance is remarkably enhanced. This same trick had little effect under Linux. This suggests that Solaris is not using disk caching as effectively as it could or as is used by Linux.

Chris Flatters has found that the following relation gives a fair, ball-park estimate of the AIPSMark$^{(93)}$ that can be expected on a machine with known SPEC(95) benchmarks:

$$\text{AIPSMark}^{(93)} = 0.898 \times \text{SPECfp}(95) + 0.110 \times \text{SPECint}(95) - 1.665$$

There is a considerable discussion of the benchmarking results on pages found from the Classic *AIPS* home WWW pages. The table summarizing recent measurements is given below. See the WWW pages for more details.

Hardware	Operating System	Compiler	AIPS Version	$A_m^{(93)}$	Flatters' prediction	Notes
DEC AlphaStation 500/400	OSF/1 4.0	DEC Fortran	15APR97	**12.0**	12.3	512 Mbytes memory, 24 Gbytes disk, Binaries on NFS-mounted partition.
DEC Alpha 21164/433 MHz	Linux 2.1.57	DEC Fortran	15APR97	**9.0**	16.1	Binaries built statically under OSF/1 4.0, run in emulation mode.
HP 9000/J282	HP-UX 10.01	HP supplied	15APR98	**13.3**	17.0	1 Processor; 1 Gigabyte main memory
IBM RS/6000 SP2	AIX 4	xlf/cc	15APR98	**4.7**	7.1	machine not empty, compiler options non-optimal
IBM RS/6000 3BT	AIX 3.2.5	xlf/cc	15JUL94	**4.6**	5.6	No preprocessor (-Pv, -Pk) used
Intel Pentium Pro 200	Linux 2.0.32	g77/egcs 1.0.1	15APR98	**6.7**	5.3 / cpu	2 CPUs, SCSI disks (similar IDE based system got **6.4**) was **3.6** with f2c/gcc
Intel Pentium II 300	Linux 2.0.32	g77/egcs 1.0.1	15APR98	**9.6**	6.3 (est.)	IDE disks, was **4.6** with f2c/gcc
Intel Pentium 166	Linux 2.0.32	g77/egcs 1.0.1	15APR98	**3.5**	2.4	narrow SCSI disks, was **2.2** with f2c/gcc
SGI Origin 200	Irix 6.4	SGI	15APR98	**18.2**	13.3 / cpu	4 CPUs, speed parameter set to 14 based on older AIPSMark
Sun Sparc Ultra 1/170	SunOS 5	Sunsoft	15APR98	**9.0**	7.2	**/tmp** / **no /tmp** / **comment**: **9.0** / **6.9** / Ultra Specific (SUL) **8.4** / **6.3** / non Ultra-specific (SOL) **7.8** / **5.9** / SOL, bin/help over LAN
Sun Sparc Ultra 1/170	SunOS 5	g77/egcs 1.0.1	15APR98	**5.9**	7.2	**4.9** without /tmp trick, cpu times 75% greater than with Sunsoft
Sun Sparc Ultra 2	SunOS 5	Sunsoft	15APR97	**9.0**	9.6 / cpu	167 MHz, dual processor
Sun Sparc Ultra 5/270	SunOS 5	Sunsoft	15APR98	**7.7**	8.4	**4.3** without /tmp trick, OS on data disk
Sun Sparc Ultra 10/300	SunOS 5	Sunsoft	15APR98	**10.9**	11.2	**8.8** without /tmp trick, OS on data disk
Sun Sparc Ultra 30/300	SunOS 5	Sunsoft	15APR98	**15.0**	16.1	**10.0** without /tmp trick

AIPS Order Form for **15APR98** *(Unix, "tar" format)*

Now licensed under the Free Software Foundation GPL

AIPS is available via anonymous ftp to aips.nrao.edu (192.33.115.103)

1. Name and address of Contact Person: ______________________

☐ Check here if address on back is ok. ______________________

(Include street address for UPS delivery) ______________________

I ☐ want ☐ do not want to receive paper copies of the Aipsletter

2. ☐ New order ☐ ReOrder Do you have an old User Agreement? ☐ Yes ☐ No

Have you registered previously? ☐ Yes ☐ No

(Registration is free to educational/research sites and is required *on every release* if you want support from NRAO)

3. Tape "media" desired: .... ☐ Exabyte (5GB) ☐ Exabyte (2.2G) ☐ 4mm DDS DAT ☐ QIC 150 (150MB) ☐ 9-track 6250 bpi ☐ CDrom ☐ 100 MB ZIP "Floppy"

We reserve the right to institute a media charge at any time

☐ Check here for BINARY TAPE (load and go), exabyte or DAT only, and indicate OS/hardware

☐ Sparc/Solaris 2.5 (SC4.2) ☐ Sparc/SunOS 4.1.2 (SC2.0.1) ☐ IBM RS6000/AIX 3.2

☐ Linux 2.0 (ELF format) ☐ Silicon Graphics IRIX 6.4.1

☐ Dec Alpha/OSF-1 4.0 ☐ HP 9000/7xx HP/UX B.10.01

4. Computer(s) make, model and OS version: ______________________

Include hardware/software/quantity info, e.g. ______________________

Sun 4/SunOS 5.4/3, Intel 486DX2/66 Linux 1.2.8/5, etc. ______________________

5. DDT test package (15JAN94 version): .... ☐ (includes small, medium, and large tests;

(FITS tape) NOTE: Also available via ftp on aips.nrao.edu in **/pub/aips/DDT/15JAN94/***

7. Printed Documents Requested: .... ☐ AIPS Cookbook (http://www.cv.nrao.edu/aips/cook.html)

☐ AIPS Memos: ______________ Going AIPS (15APR90) ☐ Vol 1 ☐ Vol 2

(See separate list at ftp://aips.nrao.edu/pub/aips/TEXT/PUBL/AIPSMEMO.LIST)

8. Custom 3-ring binders .... Cookbook ☐

Registration required if you order binders Going AIPS ☐ Vol 1 ☐ Vol 2

Send this order form (or an electronic equivalent) to:

AIPS Group
National Radio Astronomy Observatory
520 Edgemont Road
Charlottesville, VA 22903-2475
USA

Network Addresses
Internet: aipsmail@nrao.edu
UUCP: ...uunet!nrao.edu!aipsmail
Web: http://www.cv.nrao.edu/aips/
Anonymous ftp: aips.nrao.edu/aips/

AIPSLETTER
National Radio Astronomy Observatory
520 Edgemont Road
Charlottesville, VA 22903-2475

ADDRESS CORRECTION REQUESTED

To:

DATED MATERIAL-DO NOT DELAY