AIPS Imaging and Self-Calibration: MAPIT

91 May 7

This document describes the Astronomical Image Processing System (*AIPS*) procedure called MAPIT , which automatically maps and self-calibrates *uv*-data . The MAPIT procedures are useful to scientists who are interested in quickly producing higher dynamic range (1000:1) images from calibrated *uv*-data . For radio sources small compared to the primary beam of the VLA, MAPIT rapidly produces high quality images for the standard VLA frequencies. MAPIT is particularly suited to VLA Snap-shot observations. It has also been extensively tested with VLBI data.

The MAPIT procedures are explained with an example below. For information and hints concerning *AIPS* , see the *AIPS* cookbook. The MAPIT procedures require calibrated single-source *uv*-data in *UN*-compressed form. For an example of calibrating *uv*-data also see the *Summary of AIPS uv-data Calibration* (*AIPS* Memo 68, April 1991).

In the example below, the Gentle User has calibrated continuum *uv*-data and wishes to fourier transform the data in order to show their scientific break-through. The Busy User uses the *AIPS* uses the procedure MAPIT to fourier transform and self-calibrate the *uv*-data . The procedures are created within the *AIPS* environment using a *runfile*.

The MAPIT procedures will produce much better results if most spurious *uv*-data have been removed. Some helpful commands for first examining the *uv*-data are listed below.

UVPLT Plot your *uv*-data in a variety of ways to determine data quality and learn what to expect in the fourier transformed images. Never print more that 2000 UV points by setting XINC. In the example, the sources has 15000 visibilities so XINC=15000/2E3. TASK='UVPLT'; BPARM=6,7,0; INP; GO. When UVPLT has finished, use QMSPL, LWPLA, TVPL or TXPL to print the results. Plot source amplitude as a function of time: TGET UVPLT; BPARM=11,1,0; INP; GO. Next plot source amplitude as a function of *uv*-spacing: TGET UVPLT; BPARM=0; INP; GO.

MAPIT

Next, the Busy User will produce a quick, but often adequate, image of there scientific break-through using the *AIPS* procedure MAPIT . The procedures needed to run MAPIT are created using an *AIPS* runfile (RUN MAPIT). The user next selects the source for imaging, checks inputs and runs the procedure (i.e. GETN ?; INPUT MAPIT; MAPIT).

The inputs to MAPIT are described below. Note that ONLY the input source Name is required. (i.e.. INNAME='MY SOURCE';INCLASS='UVDATA';INDISK=?) Defaults will be found for all other parameters.

INNAME Input Name of *uv*-data set. The *uv*-data must be calibrated, and in single source format. (See HELP CALIB; HELP SPLIT)

Antenna Locations *kiloλ*

a) *b)*

Figure 1: *a)* PRTAN output showing the locations of the antennas at the time of observation. *b)* UVPLT output showing the location of the antenna pairs in the uv plane. Note there are N antennas and $N \times (N-1)/2$ pairs of antennas at each observation. The uv-data are from a 3.6 cm VLA A-array snapshot of Einstein Ring MG1654+1346 .

Use UVPLT to determine the types of errors in the data and learn the basic source structure. Choose XINC so that the total number of TGET UVPLT; BPARM = 0; INP; GO. For point sources it is easy to see the points that deviate from the average. Use the *AIPS* tasks UVFND to find the bad points and UVFLG to flag the bad points.

a) *b)*

Figure 2: *a)* Correlated Amplitude versus Antenna separation and *b)* Correlated Amplitude versus Time of observation for a 3.6 cm VLA A-array snapshot MG1654+1346 .

IN2NAME Input Name of a Point source observed with the program source. This UV data is only used to find the reference antenna for Self-Calibration.

INDISK Disk containing *uv*-data .

OUTDISK Disk to contain output images and clipped *uv*-data .

CELLSIZE X and Y size of pixels in output image. The CELLSIZE is related to the "average" *uv*-spacing of the observation. The default is found using the task UVPRM . If the maximum *uv*-spacing is U_{MAX} λ, the cellsize is set by the equation CELLSIZE=$2.06 \times 10^5/(4 * U_{MAX})$.

IMSIZE The output image size in pixels. Determined by the source size and the patience of the User. Default is 512 pixels square. IMSIZE must be a power of 2; i.e. 64, 128, 256, 512, 1024, 2048, or 4096

GAIN Clean gain for *Clark* deconvolution algorithm. The default value, GAIN=0.05, is good for most cases. The small value of GAIN is important for the latter clean component selection algorithm.

NITER Number of clean iterations. The default value is NITER=1000, but more iterations may be needed for large complex sources.

REFANT Reference antenna for Self-calibration. This antenna should be one near the center of the array. The default is chosen by the task UVPRM , by computing the RMS noise as a function of antenna.

SOLINT Solution interval (in minutes) for Self-calibration. This parameter a complex function of the signal-to-noise ratio of the observation and the phase fluctuations of the atmosphere. The default is 5 minutes.

NMAPS Number of Mapping and Self-calibration loops. One source map is produced each loop. After each map is produced, the *uv*-data are Self-Calibrated. The source model clean components are found using the task CCFND . The default is 3 passes.

DOPOL Produce polarized intensity maps if DOPOL > 0. Requires that the *uv*-data are polarization calibrated.

DOCENTER Move Source to map center if DOCENTER > 0. If requested, a large field of view map is produced and the map peak is shifted to the center for all later images.

DOALL Do all types of Self-Calibration (i.e.. both phase and amplitude self calibration). IF DOALL > 5 then it is used to determine when amplitude self-calibration is performed. Amplitude self calibration is done only when the PEAK of the input model is DOALL*RMS noise in the map. (IF DOALL < 5 then amplitude self calibration is done when the PEAK is 30 times the noise.)

FLUX Max flux level of *uv*-data clipped before de-convolving. A new clipped copy of the *uv*-data is created with visibilities limited to between 0 and FLUX Jy. If FLUX is zero, the source flux is estimated and the average visibility noise is used to estimate a safe flux upper limit (many sigma above the source flux). If FLUX is negative, no clipping is performed.

UVTAPER Width of Taper to apply to *uv*-data (kilo-λ). Should only be set if you are trying to find extended structures. Default is no tapering.

UVRANGE Allow only sub-set of *uv*-data to be include in the source image. Should only be used if some *uv*-data are thought to be bad. Default is to include all *uv*-data . (Note UVRANGE has two values, a minimum and max *uv*-spacing in units of Kilo-λ)

UVWTFN Weighting function for data. To use natural weighting set UVWTFN='NA'. Uniform weighting generally yields better resolution, while natural weighting yields better signal-to-noise ratio.

NBOX Number of clean boxes in the image. Should be zero during the first use of MAPIT. However, if you know where you source is, using boxes speeds *and* improves deconvolution.

BOXES Array of locations of the N clean boxes (in pixels). In order to use BOXES, a first map must be made and the CELLSIZE specified to MAPIT .

BMAJ Major axis of the clean restoring beam. If zero, the clean beam is measured. If BMAJ is negative, only the residual noise is left in the image. This is very useful for determining problems with your *uv*-data . The cleaned map can later be restored with the task RSTOR.

BADDISK Disks to avoid for scratch. Should be used if some of the system disks are faster than others.

Notes on Imaging

Below a few of the MAPIT fine points are described in more detail.

CELLSIZE The angular size of the pixels must be calculated for each image. The CELLSIZE is determined by the range of antenna spacings and the wavelength of the observations. The angular resolution of your observation is determined by the equation below.

$$\theta = \frac{\lambda}{D} \propto \frac{1}{kilo\ \lambda} \times 206\ arc\ seconds \tag{1}$$

The maximum separation of the antennas, D, and the wavelength of the observation, λ, are measured in meters. The factor $\frac{180}{\pi} \times 60 \times 60 = 206,264.8$ converts λ^{-1} to arc seconds. For MX to work properly, the CELLSIZE should be one third to one forth of θ, i.e. CELLSIZE $= \theta/3$. The sample observations were made at a wavelength of 3.6 cm, so $\theta_{BEAM} = \frac{0.036}{25} \times 2.06 \times 10^5 \sim 300\ arc\ seconds$.

The locations of aliased images of objects outside the field of view depends on CELLSIZE . To avoid freezing these aliases into the image, CELLSIZE is varied by a few percent for successive self calibration executions.

IMSIZE The image size is determined by the size of the Primary Beam of the telescope.

$$\theta_{BEAM} = \frac{\lambda}{25\ meters} \times 2.06 \times 10^5\ arc\ seconds. \tag{2}$$

The first map should show a large part of the primary beam. Set IMSIZE to the number of pixels with size CELLSIZE to cover a size of diameter θ_{BEAM}. Unless IMSIZE is specified, IMSIZE will be no greater than 512 pixels.

SOLINT The Self-Calibration solution interval is a complex function of the signal-to-noise ratio of the observation. Generally, SOLINT should be as short as possible to allow corrections for atmospheric fluctuations, but must be long enough to provide a good fit to the source model. Different "typical" values are used for different wavelengths and weather conditions.

A trial and error method for determining SOLINT is to vector average the amplitudes of the *uv*-data for a number of averaging times. A reasonable value for SOLINT is the averaging time for which the vector average of the data is half the scalar average.

The default value for SOLINT is half the observation time, up to a maximum of 5 minutes. After the first two self-calibration executions, SOLINT is decreased. On the final phase self calibration, SOLINT is one quarter of the original SOLINT value. Comments and suggestions are welcomed on this topic.

CLIP If the *uv*-data are not compressed, the data will be clipped to eliminate spurious high points. The upper limit to the clipping flux is estimated based on the average amplitude of the *uv*-data at the shortest spacings, and the RMS noise of the *uv*-data .

CCFND In order to self-calibrate *uv*-data , a source model is needed. The task CCFND selects "significant" clean components for the source model. The significant clean clean components are those brighter than twice the brightness of the first negative clean component. For Amplitude and Phase self calibration, all CCs brighter than the first negative clean components are used.

UVPRM After the source model has been found, the *uv*-data are self-calibrated. However, self-calibration will only succeed if the source model well represents the *uv*-data . Typically, with the MAPIT algorithm the shortest *uv*-data antenna separations are not well represented in the first models. The task UVPRM determines what *uv*-data spacings are represented by the source model.

CALIB The *AIPS* self-calibration task has many inputs and a few are described here. CALIB calculates a model for the *uv*-data based on the input image, and calculates *antenna based* corrections based on a fit of the *uv*-data to the model. MAPIT uses solution type, SOLTYP=L1; this is fit which depends linearly on the difference between the model and the data. This solution type is less sensitive to wild data points than the more common least squares fit. No data are flagged in the MAPIT executions of CALIB , by setting APARM(9)=1. The data with a signal to noise ratio less than 3 are excluded from the CALIB solution (APARM(7)=3). On all but the 75 % of the passes, only the antenna phases are calibrated. On the final passes, the antenna amplitudes are also calibrated.

MAPIT_MX The MX adverbs not modified by MAPIT are listed in by INPUT MAPIT_MX. These are also inputs to MAPIT , but are not as commonly modified by the novice user.

BATCH The MAPIT procedures have been tested in the *AIPS* batch system. To reduce un-needed printer output, the batch file should include the verb CLRMSG on the last line.

Appendices

This document includes appendices showing the MAPIT inputs, and the images for successive passes of self-calibration. Also a listing of the MAPIT procedures is included along with *AIPS* general help.

Please send comments to Glen Langston, NRAO C'ville. E-mail address: glangsto@nrao.edu

Appendix A: MAPIT inputs and results

Inputs for the MAPIT procedures used to create images of Einstein Ring MG1654+1346 (Langston *et al.* 1990, *Nature*, **344**:43) are listed below. The *uv*-data consists of a VLA A-array and B-array snapshots. The observations at 3.6 cm of this source are a good example of the importance of self-calibration on images of extended sources.

MAPIT produces an number of diagnostic messages and plots. A selection of these messages are also given below.

```
>run mapit
>indi=2; getn 8; in2di=2; get2n 4; outdi=1
>niter=5000; baddi=2,3,4,5,6,0
>zerosp=.15,4,4,0,1
>flux=0.25; doall=1
>input mapit
AIPS      MAPIT:  Procedure to CLEAN and Self-Calibrate UVDATA
AIPS      Adverbs         Values           Comments
AIPS      ----------------------------------------------------------------
AIPS      INNAME     '1654        '          Input UV data (name)
AIPS      INCLASS    'DBCON '                Input UV data (class)
AIPS      INSEQ         1                    Input UV data (seq. #)
AIPS      INDISK        2                    Input UV data disk drive #
AIPS      IN2NAME    '            '          Point Source UV data (name)
AIPS      IN2CLASS   '      '                Point Source UV data (class)
AIPS      IN2SEQ        0                    Point Source UV data (seq. #)
AIPS      IN2DISK       1                    Point Source UV disk drive #
AIPS      OUTDISK       1                    Output image disk drive #
AIPS      CELLSIZE      0            0       (X,Y) size of grid in asec
AIPS      IMSIZE        0            0       Image size
AIPS      NITER      5000                    Number of clean iterations
AIPS      REFANT        0                    Reference ant. for Self-cal.
AIPS      SOLINT        0                    Solution interval (min)
AIPS      NMAPS         0                    Number of Map-Selfcal loops
AIPS      DOPOL         0                    >0: Produce Polarization maps
AIPS      DOCENTER     -1                    >0: Move Source to map center
AIPS      DOALL         1                    >0: Do amplitude self-cal
AIPS      FLUX           .25                 Clip Flux Limit. <0: No Clip
AIPS      UVTAPER       0            0       Taper to apply to UV data
AIPS      UVRANGE       0            0       Range of UV data to include
AIPS      UVWTFN     '  '                    Weighting function for data
AIPS      GAIN          0                    Clean Gain
AIPS      NBOXES        3                    Number of boxes for CLEAN
AIPS      BOX         243          217       Four coordinates for each box
AIPS                  296          276        202         343
AIPS                  210          357        223         269
AIPS                  233          288       *rest 0
AIPS      BMAJ          0                    FWHM(asec) maj. axis beam
AIPS      BMIN          0                    FWHM(asec) min. axis beam
AIPS      BPA           0                    CLEAN beam position angle
AIPS      BADDISK       2            3       Disks to avoid for scratch.
AIPS                    4            5          6        *rest 0
```

```
>inp mapit_mx

AIPS      MAPIT_MX: MX Adverbs not changed by MAPIT
AIPS      Adverbs         Values            Comments
AIPS      ----------------------------------------------------------------
AIPS      BCHAN          1                   Low freq. channel 0 for cont.
AIPS      ECHAN          0                   Highest freq channel
AIPS      CHANNEL        0                   Restart channel number
AIPS      NPOINTS        1                   Number of chan. to average.
AIPS      CHINC          1                   Channel incr. between maps.
AIPS      BIF            0                   First IF in average.
AIPS      EIF            0                   Last IF in average.
AIPS      NFIELD         1                   Number of fields (max. 16)
AIPS      FLDSIZE    *all 0                  Size of each field.
AIPS      RASHIFT    *all 0                  RA shift per field (asec)
AIPS      DECSHIFT   *all 0                  DEC shift per field (asec)
AIPS      ZEROSP          .15        4       0-spacing flux and weights
AIPS                     4           0          1
AIPS      XTYPE          5                   Conv. function type in x
AIPS      YTYPE          5                   Conv. function type in y
AIPS      MINPATCH      51                   Min. BEAM halfwidth in AP.
AIPS      PHAT           0                   Prussian hat height.
AIPS      DOTV           1                   >0 => display residual field
AIPS      CMETHOD    '    '                  Modeling method:'DFT','GRID'


>mapit
AIPS      #####################################################
AIPS      #    CHECKING INPUT
AIPS      #####################################################
AIPS      #####################################################
AIPS      #    MIN AND MAX UV-RANGE
AIPS      #####################################################
UVPRM     Task UVPRM  (release of 15JUL91) begins
UVPRM     Average Flux < 2.500E-01 Jy at  1.211E+04 lambdas
UVPRM     Reference Antenna  =   4, RMS =  1.995E-02 Jy
UVPRM     Source Flux  = 8.642E-02  +/-    4.061E-04 Jy
UVPRM     FNDUV:      15293 Points Examined
AIPS      #####################################################
AIPS      # DURATION OF OBSERVATION  (MIN):        24         #
AIPS      # MAXIMUM UV RANGE (KILO-LAMBDA):       970         #
AIPS      # PIXEL SIZE       (ARC SECONDS):          .0531    #
AIPS      # VLA PRIMARY BEAM (FWHM PIXELS):      6039         #
AIPS      # IMAGE SIZE            (PIXELS):       512         #
AIPS      #####################################################
AIPS      #####################################################
AIPS      # CLIPPING POINTS BRIGHTER THAN            .25      #
AIPS      #####################################################
CLIP      Create 1654        .UVCLIP.   1 (UV)  on disk 1 cno   14
CLIP      IF  1 Ch   1 Corr 1 Pass    .0000 to    .2500 # Flagged        2
CLIP      IF  1 Ch   1 Corr 2 Pass    .0000 to    .2500 # Flagged        4
CLIP      IF  1 Ch   1 Corr 3 Pass    .0000 to    .2250 # Flagged        6
CLIP      IF  1 Ch   1 Corr 4 Pass    .0000 to    .2250 # Flagged        6
```

```
AIPS      ######################################################
AIPS      #    PLOTTING UV-DATA
AIPS      ######################################################
...
UVPLT     PLTUV:       1911 Points plotted
...
TXPL                 Amplitude vs UV dist for 1654.UVCLIP.1   Source:16528
TXPL                 Antennas  * - *     Stokes I    IF#  1
TXPL                 |++-------+--------+--------+--------+--------+
TXPL                 |||                                           |
TXPL              100|++                                          -+
TXPL            M    ||+                                           |
TXPL            i    ||+                                           |
TXPL            l  80|++                                          -+
TXPL            l    ||+-.                                         |
TXPL            i    |'++|                                         |
TXPL            J    |++O|+|                                      -+
TXPL            a  60| |O+*|                                       |
TXPL            n    | '@+++|.                                     |
TXPL            s    |  @U#|++--                                   |
TXPL            k  40|- @U#|+*.+|    |                            -+
TXPL            y    |  +@#|#++++|  +..  .     .                   |
TXPL            s    |  .++++##+#+-.+||.  .  |' .    +             |
TXPL               20|-  ++++@@#*+=.#|=+-++ |++ +: .    -  -  +'   ++
TXPL                 |   +++|@++ @#|@##++++ +#+++. #+  *. + .+. ++-|
TXPL                 |    |++|#++ +@|#++++++* ++*@#| #+  *  + |#| +++|
TXPL                0|=---+-+=+=-+++++-++++--+++++-+----+--+--+----++
TXPL                 .0       .2       .4       .6       .8
TXPL                                  Mega Wavlngth
UVPLT     PLTUV:       1911 Points plotted
TXPL                 Amplitude vs Time hrs for 1654.UVCLIP.1   Source:1658
TXPL                 Antennas  * - *     Stokes I    IF#  1
TXPL                 |------+------+------+------+-----+------+---++
TXPL                 |                                            ||
TXPL              100|-                                           ++
TXPL            M    |                                            ||
TXPL            i    |                                            ||
TXPL            l  80|-                                           ++
TXPL            l    |       +                                    ||
TXPL            i    |       +                                    ||
TXPL            J    |-      +                                    ++
TXPL            a  60|       +                                    ||
TXPL            n    |       +                                    ||
TXPL            s    |       +                                    ||
TXPL            k  40|-      +                                    ++
TXPL            y    |       +                                    ||
TXPL            s    |       +                                    ||
TXPL               20|-      +                                    ++
TXPL                 |       +                                    ||
TXPL                 |       +                                    ||
TXPL                0|=-----++-----+------+------+-----+------+---++
TXPL                00      20   116     212     308    404    500
TXPL                                   IAT (HOURS)
...
```

```
AIPS     #####################################################
AIPS     #    GET REFERENCE ANTENNA
AIPS     #####################################################
UVPRM    Average Flux < 2.500E-01 Jy at  1.211E+04 lambdas
UVPRM    Reference Antenna  =   4, RMS =  1.995E-02 Jy
UVPRM    Source Flux  = 8.642E-02  +/-    4.061E-04 Jy
UVPRM    FNDUV:      15293 Points Examined
AIPS     #####################################################
AIPS     #    DE-CONVOLVING
AIPS     #####################################################
AIPS     Image=1652+138  (UV)        Filename=1654        .UVCLIP.   1
AIPS     # visibilities     15293     Sort order  TB
AIPS     ------------------------------------------------------------
AIPS     Type    Pixels   Coord value  at Pixel    Coord incr   Rotat
AIPS     COMPLEX      3    1.000000E+00       1    1.0000E+00     .00
AIPS     STOKES       4    RR                 1       -1.0000     .00
AIPS     FREQ         1    8.414900E+09       1    5.0000E+07     .00
AIPS     RA           1    16 52 23.790       1       .000000     .00
AIPS     DEC          1    13 51 08.360       1       .000000     .00
AIPS     ------------------------------------------------------------
MX       MXIN  : Sort ='TB' not 'X*', Using Un-sorted gridding
MX       Create 1654        .IBEAM .   1 (MA)  on disk 1 cno   17
MX       Create 1654        .ICLN  .   1 (MA)  on disk 1 cno   29
MX       Create 1654        .UVWORK.   1 (UV)  on disk 1 cno   42
MX       Using all     15293 Visibilities
MX       Got data     Time= 10:25:25 CPU time=      2.85
MX       UVTBUN: Weighting grid =   512 X   512, Box = 0
MX       UVGRTB: All   132 Rows in 1280k AP (  263 Rows Max)
MX       UVGRTB: Max U Baseline      935133. lambda (=  125 cells)
MX       UVGRTB: Frequency 8.414900E+09 Hz
MX       UVZRWT: Zero spacing  1.000E-01, Weight  5.000E-01
MX       UVZRWT: Maj, Min Ax.  4.000E+00", 4.000E+00", Angle     .0 Deg
...
AIPS     #####################################################
AIPS     #    FINDING RMS NOISE IN IMAGE
AIPS     #####################################################
...
AIPS     #####################################################
AIPS     # SOURCE     1654            , RMS =      .00014    #
AIPS     #####################################################
AIPS     #####################################################
AIPS     # LOOP            1          OF          3          #
AIPS     #####################################################
AIPS     #####################################################
AIPS     #    SELECT CLEAN COMPONENTS
AIPS     #####################################################
...
CCMRG    Wrote       551 CLEAN components to CC file version  1
CCFND    Component       468 has Flux      .000022 >       .000021 Jy.
CCFND    Sum up to       468 is  Flux      .076856 Jy.
UVPRM    Average Flux < 7.686E-02 Jy at  3.234E+04 lambdas
...
```

```
AIPS     ####################################################
AIPS     #    SELF-CAL UV DATA
AIPS     ####################################################
AIPS     # NUM CLEAN COMPONENTS      468
AIPS     # REFERENCE ANTENNA           4
AIPS     # UV MINIMUM (K-LAMBDA)      48.4687
AIPS     # SOLUTION INTERVAL           5
AIPS     # SOLUTION MODE            P
AIPS     ####################################################
...
CALIB    CALIB USING 1654         . UVCLIP .   1 DISK=  1 USID=1256
CALIB    Warning: failed solutions replaced by (1,0)
CALIB    L1 Solution type
...
CALIB               Previously flagged   Flagged by gain        Kept
CALIB    Partially                 64                  0          64
CALIB    Fully                      0                  0       15229

SNPLT    SUMPLT:       106 Gain amp     points plotted
SNPLT    SUMPLT:       106 Gain phs     points plotted
...
TXPL                 Extrema vs IAT time for 1654.UVCLIP.1 SN  *pol IF *
TXPL             1.10+--+------+-------+------+------+------+-------+|
TXPL                 +-                                             -|
TXPL         G   1.05|                                               |
TXPL         a       |                                               |
TXPL         i   1.00+-  =                                      |+  -|
TXPL         n       |                                               |
TXPL              .95+-                                             -|
TXPL                 |  .      .       .      .      .      .       .|
TXPL              .90+--+=-----+-------+------+------+------+-------+|
TXPL         D     40+-  @                                          -|
TXPL         e     20+-  @                                          -|
TXPL         g       |   @                                      .-   |
TXPL         r      0+-  @                                      |+  -|
TXPL         e       |   @                                      |+   |
TXPL         e    -20+-  @                                          -|
TXPL         s    -40+-  @                                          -|
TXPL                 |   @                                           |
TXPL              -60++-+=-----+-------+------+------+------+-------+|
TXPL                      20    116    212    308    404    500    520
TXPL                                       IAT (HOURS)
...
AIPS     ####################################################
AIPS     #    DE-CONVOLVING
AIPS     ####################################################
...
AIPS     ####################################################
AIPS     #    FINDING RMS NOISE IN IMAGE
AIPS     ####################################################
...
AIPS     ####################################################
AIPS     # SOURCE     1654            , RMS =     .000099    #
AIPS     ####################################################
```

```
AIPS      ####################################################
AIPS      #    SOLUTION IMPROVED, DELETING PREVIOUS
AIPS      ####################################################
AIPS      ####################################################
AIPS      # LOOP            2          OF           3         #
AIPS      ####################################################
...
AIPS      ####################################################
AIPS      # LOOP            3          OF           3         #
AIPS      ####################################################
...
TXPL                  Extrema vs IAT time for 1654.SCAL.2 SN  *pol IF *
TXPL                  +--+------+-------+------+------+------+-------+|
TXPL                  +-  =                                          -|
TXPL              1.10|   =                                           |
TXPL          G   1.05+-  @                                   `+   -|
TXPL          a       |   I                                   `-    |
TXPL          i   1.00+-  =                                   `+   -|
TXPL          n       |   +                                   .+    |
TXPL               .95+-  +                                   .+   -|
TXPL                  |                                       |+    |
TXPL               .90+- .      .       .      .      .      .+   +|
TXPL                  +--+=-----+-------+------+------+------+-------+|
TXPL          D      2+-  +                                   `-   -|
TXPL          e      1+-  +                                   `+   -|
TXPL          g      0+-  +                                   .+   -|
TXPL          r     -1+-  @                                   |+   -|
TXPL          e       |   =                                   |*    |
TXPL          e     -2+-  +                                   |+   -|
TXPL          s     -3+-  @                                   .+   -|
TXPL                -4+-  +                                   `-   -|
TXPL                -5++-+=-----+-------+------+------+------+-------=|
TXPL                    20    116    212    308    404    500    520
TXPL                                    IAT (HOURS)
...
AIPS      ####################################################
AIPS      #    DONE WITH SOURCE ==>1654
AIPS      ####################################################
```

Eventually the final amplitude self-calibrated map is produced with signal to noise ratio much better than was possible for the un-self calibrated *uv*-data .

The task SNPLT plots the extrema of the antenna phase and amplitude corrections. Notice that by the third self-calibration the phase corrections are small. The amplitude corrections will only be successful if the phase errors have been significantly reduced.

RIGHT ASCENSION

a) *b)* *c)*

Figure 3: *a)* Fourier Transformed image (Dirty Map), *b)* Cleaned image of a 3.6 cm VLA snap shot of MG1654+1346 . The *uv*-data are not self-calibrated. In *c)* the *uv*-data were self-calibrated based on a source model from image *b)*. The image quality is measured by the RMS signal in a box away from the source. For *a)* the RMS noise is 0.45 mJy/beam. For *b)* the RMS noise is 0.13 mJy/beam. For *c)* the RMS noise is 0.10 mJy/beam.

RIGHT ASCENSION

d) *e)*

Figure 3 cont'd: *d)* Cleaned image after self-calibration of original *uv*-data based on source model *c)*. The final map is shown in figure *3)*, produced from *uv*-data which was both amplitude and phase self-calibrated. For *d)* the RMS noise is 0.09 mJy/beam. For *e)* the RMS noise is 0.09 mJy/beam. There were apparently few amplitude errors in the *uv*-data, and further iterations of self-calibration did not improve the RMS noise.

AIPS has online help which is available at three levels of detail.

INPUTS Lists the a very short summary of the function of a task, verb or adverb.

HELP Gives an expanded description of the adverbs for a task.

EXPLAIN Gives detailed explaination of a task functions, what the adverbs are used for and references to the mathmatical methodes used by the task.

Typing HELP HELP gives the summary of *AIPS* commands.

```
HELP ADVERB      The nature and use of adverbs
HELP ADVERBS     does HELP REALS, HELP ARRAYS, and HELP STRINGS
HELP ARRAY       The nature and use of arrays
HELP ARRAYS      lists all arrays in the symbol table
HELP BATCHJOB    Operations to prepare, submit and monitor batch jobs
HELP DOTASK      Information used in executing tasks
HELP NEWTASK     Writing and incorporating a new task into AIPS
HELP PANIC       Solution to common problems
HELP PROC        The nature and use of procedures
HELP PROCEDUR    The nature and use of procedures
HELP PROCS       lists all procedures in the symbol table
HELP PSEUDO      The nature and use of pseudo verbs
HELP REALS       lists all real, non-array adverbs in the symbol table
HELP ROAM        Using the ROAM capability on the TV
HELP RUN         The nature and use of run files
HELP STRINGS     lists all characters string adverbs in the symbol table
HELP TASKS       lists all tasks which are started by GO
HELP VERB        The nature and use of verbs
HELP VERBS       lists all verbs in the symbol table
                 followed by all pseudoverb
HELP WHATSNEW    Major changes in AIPS since the last update
```

```
The following HELPers give lists of verbs & tasks grouped in general classes.
```

```
HELP INDEX       Lists all tasks, verbs, pseudoverbs and
                 built-in procedures in alphabetical order
HELP ANALYSIS    Image processing, analysis and combination
HELP APTASKS     Tasks which use the Array Processor
HELP CATINFO     Dealing with the image catalog
HELP CUBE        Data cube (spectral line) processing
HELP CURSOR      Interactive use of cursor on TV and TEK
HELP DELETE      Deleting files
HELP GENERAL     General AIPS utilities
HELP MAPETC      Mapping, Cleaning and Self-Calibration
HELP PL2D        Two-dimensional displays and plot files
HELP POPSYM      Symbols used in POPS interpretive language
HELP SL1D        One-dimensional displays and analysis
HELP TAPU        Reading from, writing on and moving tapes
HELP TVCOLOR     Using color on the TV
HELP TVGEN       General TV function for loading and display
HELP TVINTER     Interactive use of the TV display
HELP UVPR        UV processing
HELP VLBI        Software dealing specifically with VLB data
```

```
$     ---- MAPIT.001 ----
$Automatic Mapping batch map making procedures
$Procedures search for a maximum in the field and shift
$the center of a high resolution map to the maximum.
$Procedures next self-calibrate the UV data and produce
$high dynamic range I Q and U images.
$BECAUSE of limited AIPS procedure space, comments are
$mostly outside the procedures.
$
$Glen Langston, MIT, MPIfR, NRL, and NRAO
$Joseph Lehar and Sam Conner, MIT
$last editted by glen langston on 1992 June 25
RESTORE 0

$turn of messages for long procedures
MSGKILL TRUE
$Define variables
PROC DECLARE;SCALAR OKSCAL
SCALAR RACELL,DECELL,PIXSI,XAXIS,ISTOKE,INBX,II,IJ,IJK,IIM
SCALAR PEAK,NUMCC,NSCAL,SNVER,UVFREQ,CRP1,CRP2,UV0SEQ,BMSIZ
SCALAR UVDIS,MSCDIS,MADIS,UV0DIS,IN0DI,IN0SE,MSCSEQ,UVSEQ
SCALAR VIEW,OTIME,LOOP,CCFACT,UVLOW,UVMAX,X1,X2,Y1,Y2,XY,DXY
SCALAR BOXSI,PEEK1,PEEK2,RMS,LASTRMS,RMSFACT,RMSLIM,NOLDUV
SCALAR MNSIZ,MSOLIN,CSOLIN,NSOLIN,ZRMS,ZFLUX,PBIF,PEIF,MINRMS
SCALAR MNMAP,CLSAV,CLFLUX,CLIMIT,NUMVIS,NUMIF,NUMSTOK,NUMFREQ
SCALAR MGAIN,MNITER,MFLUX,MBMAJ,MBMIN,MBPA,MFACT,MPIXSI,MBOX
SCALAR DOAMP,MDOPOL,MDOTV,DOSTOP,DOZAP,DOINT,MDOINT,MDOALL,MNBOX
ARRAY  MUVRANG(2),MFLDSI(2,16),RMSBOX(4),SOLFAC(20),XYSIZE(2)
ARRAY  MRASHI(16),MDECSHI(16),MIMSIZ(2)
STRING*1 ASTOKE,BSTOKE(7);STRING*2 MUVWTFN;STRING*4 SCTYPE
STRING*6 MAPCLASS,MSCLASS,UVCLASS,UV0CLAS,IN0CL,STCLASS(7)
STRING*8 MAPKEY(3,7),INTASK;STRING*16 MSCNAM,UVNAM,MAPNAM,IN0NAM
STRING*50 OUTLIN,WEAKPEA,ENTBOX,GOODBX,NOBOX,TWOPEA,NOCLIP
STRING*53 POUNDS,BIGMAPA,BIGMAPB,ADJUSTV,MDLPNT,MDLIMG,MDLFST
FIN

$gwt: sets the task name, goes and waits
PROC GWT(TASK);GO;WAIT;RET;FIN

$clrzap: clears the file status and zaps it
PROC CLRZAP;CLRST;ZAP;RET;FIN

$thecat: utility for listing all files with same name
PROC THECAT; INTASK=TASK;TPUT ZAP
 INCL='';INS=0;INTY='';CATA;TGET ZAP;TASK=INTASK;RET;FIN

$gotxpl: Puts TXPL output in history and on CRT
$        plot on tty with no waiting. plot file is deleted
PROC GOTXPL;INS=0;INV=0;DOCRT=3;GWT('TXPL')
 INEXT='PL';CLRST;IF(DOZAP>0)THEN EXTDES;END
RET;FIN

$typound: TYPE POUNDS to mark STEPS
POUNDS='#####################################################'
PROC PRPND;TYP POUND;RET;FIN

$head: Types next MAPIT Step.
$      Pounds signs surround output
PROC HEAD(OUTLIN);PRPND;TYP '#',OUTLIN;PRPND;RET;FIN

$setmap: Set Name adverbs for Latest MAP
PROC SETMAP
 INN=MAPNA;INDI=MADI;INCL=MAPCL;INS=0;INTY='MA';RET;FIN

$setmod: Set Name of self cal input model
PROC SETMOD;IN2N=MSCNA;IN2D=MSCDI;IN2C=MSCLA;IN2S=MSCS;RET;FIN

$resetmap: ReSet Map Name adverbs to Inname
PROC RESETMAP;MAPNAM=INN;MADIS=INDI;MAPCL=INCL;RET;FIN

$setuvd: Set Name adverbs for input UV-Data
PROC SETUVD
 INN=UVNAM;INDI=UVDI;INCL=UVCL;INS=UVSE;INTY='UV';CLRST;RET;FIN

$getsize: return ra and dec size in pixels of current map
PROC GETSIZE;KEYW='NAXIS1';GETH;XYSIZ(1)=KEYV(1)
 KEYW='NAXIS2';GETH;XYSIZ(2)=KEYV(1);RET;FIN

$peakfnd: find peak in map
$ PEAK  - OUTPUT brightness of highest peak in the images
$ PIXXY - OUTPUT pixel containing highest peak
PROC PEAKFND;HEAD('FINDING PEAK')
 SETMAP;DOINV=0;BLC=0;TRC=0;IMST;PEAK=PIXV;RET;FIN

$tvpix: procedure to load the central part of an image to
$       to the tv.
$ PEAK - Peak of image to be loaded
$ RMS  - Rms noise level of image to be loaded

PROC TVPIX(PEAK,RMS);PIXRA=-2*RMS,MIN(20*RMS,PEAK)
 GETSIZ;TXINC=XYSIZ(1)/512;TYINC=XYSIZ(2)/512;TVLOD;RET;FIN

$gettvb: Get TV box for MX; Asks the user for input clean boxes
$        The previous clean image is loaded on TV.
$        TV pixel range is set so that faint features are seen
$        User Fiddles the TV colors until satisfied.
$        Next user may quit or enter a number of clean boxes
$        Boxes are placed in the center of the screen
$DOINT   - IF > 0 then DO interactive setting of clean boxes.
$DOSTOP  - IF > 0 then DO stop map+self-cal loop
$RMS     - RMS Noise level of image.
$LASTRMS - Increase Last RMS to allow new boxes to be used.
ADJUSTV = 'ADJUST TV and INPUT CLEAN BOXES:'
NOBOX ='Enter  0 to Continue without stopping again:'
ENTBOX='Enter Number of Clean Boxes (<0 Means stop):'
PROC GETTVB
 IF (DOINT>0&DOSTOP<0) THEN HEAD(ADJUSTV);TVIN;TVPIX(PEAK,RMS)
  TVFID;BOXSI=MNSIZ/5;NBOX=0;INBX=11
  WHILE (INBX<-10!INBX>10) TYP ENTBOX;TYP NOBOX;READ INBX;END
  IF (INBX>0) THEN INBX=MIN(MAX(INBX,1),10);NBOX=INBX
   FOR II=1:INBX
    IF (BOX(1,II)=0) THEN XY=II*3
     BOX(1,II)=XY+(XYSIZ(1)*.4); BOX(2,II)=XY+(XYSIZ(2)*.4)
     FOR IJ=1:2;BOX(IJ+2,II)=BOX(IJ,II)+BOXSI;END
    END;END; LASTRMS=2*LASTRMS
   REBOX;TYP GOODBX;TYP NOBOX;READ INBX
   IF (INBX<0) THEN REBOX;INBX=1;END
  END;END
 IF (INBX=0) THEN DOINT=-1;END;PIXRA=0
 IF (INBX<0) THEN HEAD('STOPPING SELF-CAL');DOSTOP=1;END
RET;FIN

$startmx: runs mx and cleans up files after mx has finished
$  All files (UVWORK, CLEAN and BEAM) are created on MADIS
$  UVWORK and BEAM are deleted.
$  Inname is reset to output map name
PROC STARTMX;SETUVD;CLR2N;CLRON;IN2DI=MADI;OUTD=MADI
 FACTOR=MFACT;GWT('MX');INS=0
 INCL='UVWORK';INTY='UV';INDI=OUTD;CLRZA
 INCL='?BEAM'; INTY='MA';INDI=OUTD;CLRZA;SETMAP
RET;FIN

$setnear( BOXSI) : sets a box around PIXXY(1,2) location
$                  Limits box to within the image
$ PIXXY  - X and Y pixel number for box center
$ BOXSI  - radius of region to set blc and trc
$ XYSIZ  - size of image in pixels
PROC SETNEAR(BOXSI)
 BLC MAX(1,PIXX(1)-BOXSI),MAX(1,PIXX(2)-BOXSI);GETSIZE
 TRC MIN(XYSIZ(1),PIXX(1)+BOXSI),MIN(XYSIZ(2),PIXX(2)+BOXSI)
```

```
RET;FIN

$GETBOX  - Returns the rms inside a box at X,Y
$ BOXSI  - Radius of the box for statistics measurement
$ RMS    - OUTPUT RMS noise of box centered on X,Y
PROC GETBOX(X,Y,BOXSI,RMS)
 PIXXY=X,Y,0;SETNEAR(BOXSI);IMST;RMS=PIXSTD;RET;FIN

$GETRMS: Search four corners for second lowest RMS
$ More than one corner may have the source in it.
$ RMS   - OUTPUT average of middle 2 RMS values
$         RMSNOISE is put in map header.
$ X2, Y2- Scratch variables used nowhere else
PROC GETRMS(RMS); HEAD('FINDING RMS NOISE IN IMAGE')
 DOINV=-1;GETSIZ;BOXSI=XYSIZ(1)/10;XY=1.5*BOXSI
 GETBOX(XY,XY,BOXSI,RMSB(1));X2=XYSIZ(1)-XY
 GETBOX(X2,XY,BOXSI,RMSB(2));Y2=XYSIZ(2)-XY
 GETBOX(X2,Y2,BOXSI,RMSB(3));GETBOX(XY,Y2,BOXSI,RMSB(4))
 RMS=MAX(MIN(RMSB(1),RMSB(2)),MIN(RMSB(3),RMSB(4)))/2
 RMS=MIN(MAX(RMSB(1),RMSB(2)),MAX(RMSB(3),RMSB(4)))/2+RMS
 KEYWORD='RMSNOISE';KEYV=RMS,0;KEYTYP='R';PUTHEAD
 PRPND; TYP '# SOURCE ',INN,', RMS =',RMS,'#'; PRPND
RET;FIN

$MAPEND;  MAP cleanup info
PROC MAPEND;PEAKFND;GETRMS(RMS);RET;FIN

$calcshift: Set MX parameters to put map center on peak.
$           If map was shifted, add previous shift.
$ PIXXY(1) - INPUT Ra  Pixel location
$ PIXXY(2) - INPUT Dec Pixel location
$ X        - OUTPUT Ra  shift in arc seconds
$ Y        - OUTPUT Dec shift in arc seconds
 PROC CALSHIFT
 KEYW='CDELT1';GETH;RACELL=KEYV(1)
 KEYW='CDELT2';GETH;DECELL=KEYV(1)
 KEYW='CRPIX1';GETH;CRP1=KEYV(1)
 KEYW='CRPIX2';GETH;CRP2=KEYV(1)
 X=((PIXX(1)-CRP1)*RACELL)*3600;RASHI=X,0
 Y=((PIXX(2)-CRP2)*DECELL)*3600;DECSH=Y,0
RET;FIN

$MIDSHIFT: locates the best shift location;
$          If first peak is 6 times noise, get position
$          if second peak > than 50 % of first and 7 RMS
$          then shifts between two peaks, if both peaks in view
$ VIEW      - Field of view of final image, shift between two
$             peaks must be smaller, or they both fall off map
$ RASHI(1)  - output shift in arc seconds.
$ DECSH(1)  - output shift in arc seconds.
TWOPEA ='# SHIFT: Shifting between two peaks'
PROC MIDSHIFT;DOINV=-1;PEAKFND;PEEK1=PEAK;X1=0;Y1=0;PRPND
 IF (PEEK1>6*RMS) THEN
  MAXFIT;CALSHI;X1=X;Y1=Y;DOINV=1;SETNEAR(12);IMST
  DOINV=0;PEEK2=PIXV;CALSHI;DXY=MAX(ABS(X1-X),ABS(Y1-Y))
  IF (PEEK2>MAX(PEEK1/2,MINRMS*RMS) & DXY<VIEW) THEN
   X1=(X1+X)/2;Y1=(Y1+Y)/2;FLDSI=0;MFLDSI=0; TYP TWOPEA
  END; END
 TYP '# SHIFT: DELTA RA  =',X1,' DEC =',Y1
 RASHI=X1,0;DECSH=Y1,0;PRPND
RET;FIN

$GETIT:  GET Number of clean ITerations
$ NITER   - Output Number of clean iterations in map
PROC GETIT;SETMAP;KEYW='NITER';GETH;NITER=KEYV(1);RET;FIN

$GETSEQ:  GET sequence number of file
PROC GETSEQ;INS=0;KEYW='IMSEQ';GETH;INS=KEYV(1);RET;FIN

$mamx: clean map, using dirty map to shift the phase center
$  PIXSI  - size of one pixel in arcseconds
$  FLUX   - cut off flux for clean
$  MUVRANG- Limit of UV range to use for deconvolution
$  RMS    - OUTPUT rms noise in map
PROC MAMX;HEAD('DE-CONVOLVING');SETUVD;UVRANG=MUVRA
 OUTD=MADIS;OUTN='';OUTS=0;CELLS=PIXSI;DOTV=MAX(MDOT-1,-1)
 QH;STARTMX;UVRANG=0;DOTV=-1;MAPEND
 IF (MDOT>0) THEN TVIN;TVPIX(PEAK,RMS);END
RET;FIN

$scalit: self calibrate UV file, using the clean image
$  NUMCC - number of clean components for model
$  UVLOW - lower limit of uvdata to self-calibrate
$  REFANT- reference antenna number
$  DOAMP - Do amplitude selfcalibration
$  SOLINT- Solution interval (min). Calculated from input
$          interval (CSOLIN) times a facter for each LOOP
$ CALIB APARM parameters are important for solution
$ quality: APARM(1) == Number of antennas for solution
$          APARM(3) >0 Average RR, LL data
$          APARM(5) >0 Average IFs
$          APARM(7) == SNR cutoff (I like 3)
$          APARM(9) >0 Pass Failed Solutions (Passing is
$                      a good idea for first few solutions)
$ avoid aliases on side lobes by varying cell size
$ do amp and phase self-cal only if requested
PROC SCALIT
 SOLMOD='P';SETUVD;NMAP=NFIELD;UVRANGE=UVLOW,1E12;SOLCON=0
 SNVER=0;XINC=0;CALSOUR='';SUBAR=0;SMOOTH=0;INVER=0
 ANTWT=0;GAINERR=0;APARM=3,0,0,0,0, 0,3,0,1,0;CPARM=0,1,0
 SOLIN=SOLFAC(MAX(1,MIN(NSOLIN,LOOP)))*CSOLIN; SOLTYP=SCTYPE
 IF (PEAK<MDOALL*RMS) THEN
  APARM(5)=MIN(NUMIF-1,1);APARM(3)=MIN(NUMSTO-1,1)
 ELSE
  IF (LOOP>0.75*NSCAL&DOALL>0) THEN DOAMP=1; END
 END
 IF (DOAMP>0&(PEAK>MDOALL*RMS)&DOALL>0) THEN
  SOLMOD='A&P';SOLINT=4*CSOLIN; APARM(1)=4;NUMCC=0;END
 NCOMP=NUMCC;HEAD('SELF-CAL UV DATA')
 IF (NUMCC>0) THEN TYP '# Num Clean Components ',NUMCC
 ELSE TYP '# Using All Clean Components ';END
 TYP '# Reference Antenna    ',REFANT
 TYP '# UV Minimum (K-lambda)',UVLOW
 TYP '# Averaging Time (min) ',SOLINT
 TYP '# Solution Mode        ',SOLMOD; PRPND
 QH;OUTCL='SCALO';OUTD=UVDI;OUTS=0;GWT('CALIB')
 INS=0;INVER=0;XAXIS=0;NCOUNT=0;PIXRA=0;INEXT='SN'
 OPTYP='SUM';GWT('SNPLT');GOTXPL;OPTYP=''
 INCLA='SCALO';OUTCL='SCAL';CLRST;RENAM
 UVCLAS=OUTCL;UVSEQ=0;SETUVD;INEXT=''
RET;FIN

$CLPIT: Remove wild points many sigma above mean
$ set clip flux limit based on number of vis and RMS
$ assume source is less than 90% polarized
$ Source Flux and RMS noise are read from UVDATA header
$ CLFLUX - Flux level to clip data
$ NUMVIS - Number of visiblities
$ ZFLUX  - Flux of source measured by UVPRM
$ ZRMS   - RMS noise of UVdata measured by UVPRM
PROC CLPIT; SETUVD;UVRA=MUVRA;ANTE=0;BASELI=0
 KEYW='UVPFLUX0';GETH;ZFLUX=KEYV(1)
 KEYW='UVPRMSF0';GETH;ZRMS =KEYV(1)
 IF (CLFLUX<=0) THEN CLFLUX=ZFLUX+(SQRT(NUMVI/2)*ZRMS);END
 APARM=CLFLUX,(CLFLUX*.9),(ZRMS/SQRT(NUMVI/2)),0
 IF (CLFLUX>CLIMIT) THEN PRPND
  TYP '# Clipping points brighter than ',CLFLUX,'#';PRPND
  CLRON; OUTD=MADIS; OUTCL='UVCLIP'; OUTS=0; GWT('CLIP')
  UV0CLAS=OUTCL;UVCL=OUTCL;UVDI=MADIS;UVSE=0;UV0SE=0;SETUVD
  END
```

```
RET;FIN

$PLOTIT: Plot UV data vs Antenna Spacings
$        set XINC to plot only 2000 points
$ NUMVIS - Input Number of visibilities in UV data set
$ Two plots are made for each IF,
$  1) amplitude vs baseline length and 2) amplitude vs time
PROC PLOTIT;SETUVD;PBIF=MIN(BIF,NUMIF);PEIF=EIF
 IF (PEIF<BIF!PEIF<=0) THEN PEIF=NUMIF;END
 FOR BIF=MAX(1,PBIF):MAX(1,PEIF);BPARM=0;BPARM(8)=250
  PRPND;TYP '# Plotting UV-data for IF Number',BIF,'#';PRPND
  XINC=0;TIMER=0;INV=0;DOPOL=0;DOCAL=-1;SUBAR=0;SOURC=''
  GWT('UVPLT');GOTXPL;XINC=MAX(1,NUMVIS/2E3);BPARM=11,0
  GWT('UVPLT');GOTXPL;XINC=0;BPARM=0
 END; BIF=PBIF
RET;FIN

$GETPIX: Read Min and max UV range and observing duration
$ PIXSI - Pixel size in arc seconds, calculated if not set
$ UVMAX - OUTPUT maximum uvrange of input data (in lambda)
$ OTIME - OUTPUT duration of observation in minutes.
$         Time is converted from days to minutes
$ Task UVPRM finds the Min and Max Uvrange, within input
$      UVRANGE; UVPRM results are passed in Keywords
PROC GETPIX; HEAD('Finding Min and Max UV-Range')
 SETUVD;UVRA=MUVRA;BPARM=0;DOPOL=0;GWT('UVPRM')
 IF (MUVRA(2)>0) THEN UVMAX=MUVRA(2)*1000
 ELSE KEYW='UVPRAMAX'; GETH; UVMAX=KEYV(1); END
 PRPND; KEYW='UVPOTIME'; GETH; OTIME=KEYV(1)*1440
 TYP '# DURATION OF OBSERVATION  (MIN): ',OTIME,'#'
 TYP '# MAXIMUM UV RANGE (KILO-LAMBDA): ',CEIL(UVMAX/1E3),'#'
 IF (PIXSI<=0) THEN PIXSI=2.06E5/(3.5*UVMAX); END
 TYP '# PIXEL SIZE       (arc seconds): ',PIXSI,'#'
RET;FIN

$getaxis: determine whether data are compressed and
$         find number of stokes parameters and IFs
$         if COMPLEX axis size is not 3,
$         then then data are compressed.
PROC GETAXIS; KEYW='GCOUNT';GETH;NUMVIS=KEYV(1)
 KEYW='NAXIS';GETH;NAXIS=KEYV(1);NUMSTO=0;NUMIF=0;NUMFRE=0
 FOR I = 1 TO NAXIS
  KEYW=MAPKEY(1,I); GETH; KEYW=MAPKEY(2,I); GETH
  IF (KEYSTR='STOKES') THEN NUMSTO=KEYV(1);END
  IF (KEYSTR='IF') THEN NUMIF=KEYV(1);END
  IF (KEYSTR='FREQ') THEN
   NUMFRE=KEYV(1);KEYW=MAPKEY(3,I);GETH;UVFREQ=KEYV(1)
  END; END
RET;FIN

$CHECKIN: Check User input via MAPIT.HLP
$ parameters are stored for latter re-use
$  UVNAM =INNAM - Input UVdata
$  MSCNAM=IN2NA - Model Self Cal Image (for restarting)
$  MAPNAM       - Output Image name
$  NSCAL =NMAPS - Number of self calibration loops
$  The "M" ADVERBS are used to recored the MAPIT inputs
PROC CHECKIN; CLRMSG; HEAD('CHECKING INPUT'); MSOLI=SOLIN
 NUMVIS=1;KEYV=1;UVMAX=1;LOOP=-1;NSCAL=1;UVFREQ=22E9
 PRTAS=''; PRNU=0; INP MAPIT_NA; INP MAPIT_UV; INP MAPIT_MX
 TPUT MAPIT; TPUT MAPIT_MX; PIXSI=CELLS(1);CSOLIN=SOLINT
 IN0CL=INCL;IN0S=0;IN0DI=INDI;IN0N=INN;SUBAR=0;BCOM=0;DOAMP=0
 MDOT=DOTV;MFACT=FACTOR;IF(MFACT>.5) THEN MFACT=-.1;END
 MUVWTFN=UVWTFN;MNMAP=NMAP;GAINU=0;OKSCAL=-1
 MFLUX=FLUX;MNIT=NITER;MGAIN=GAIN;MDOPOL=DOPOL;MBOX=NBOX
 MBMAJ=BMAJ;MBMIN=BMIN;MBPA=BPA;MUVRA=UVRANG;MFLDSI=FLDSI
 MRASHI=RASHI;MDECSHI=DECSHI;DOTV=-1;DOCAL=-1;MIMSIZ=IMSIZ
 CLSAV=CUTOF;CLFLU=CUTOF;CUTOF=0;DOBAN=-1;DOPOL=-1;DOSTOP=-1
 ANTENN=0;BASELI=0;NSCAL=NMAP; IF (NMAP=0) THEN NSCAL=3; END
 MDOALL=ABS(DOALL); IF (MDOALL<5) THEN MDOALL=30; END
 RMS=1E-9; PEAK=10*RMS; PIXV=RMS; LASTRMS=RMS/2; PIXSTD=RMS
 UVNAM=INN;UVCLAS=INCL;UVSEQ=INS;UVDI=MAX(INDI,1)
 UV0CL=INCL;UV0SE=INS;MAPNAM=INN;MAPCL='ICLN';MADIS=OUTD
 MSCNA=IN2N;MSCLAS=IN2CLA;MSCSE=IN2S;MSCD=IN2DI
 CLRON; OUTDI=MADIS; ASTOK=SUBSTR(STOKES,1,1)
 FOR I=1 TO 7; IF (ASTOK=BSTOKE(I)) THEN ISTOK=I; END; END
 IF (ISTOK=2!ISTOK=3) THEN ISTOK=1;END; MAPCL=STCLA(ISTOK)
 STOKES=BSTOK(ISTOK); IF (MADIS<=0) THEN MADIS=INDI;END
 IF (MNIT<=0) THEN MNIT=1000;END
 IF (MGAI<=0) THEN MGAI=0.05;END
$ get UV par
 GETAXIS; GETPIX; BMSIZ=FLOOR((2.7E12/UVFREQ)/PIXSI)
 TYP '# VLA Primary beam (FWHM pixels): ',BMSIZ,'#'
 IF (IMSIZ(1)<=7) THEN
  IMSIZ=MIN(2**CEIL(LN(BMSIZ)/LN(2)),512); END
 MNSIZ=MIN(IMSIZ(1),IMSIZ(2))
 TYP '# Image size           (pixels): ',MNSIZ,'#'
 PRPND; VIEW=IMSIZ(1)*PIXSI*.95; MPIXS=PIXSI
 IF (CSOLI<=0) THEN CSOLI = MIN((OTIME+.5)/2,5); END
 TYP '# Self-Calibrate if Signal/Noise >',MINRMS,'#'
 IF (DOALL>0) THEN
  TYP '# Ampl. Self-cal if Signal/Noise >',MDOALL,'#';END
RET;FIN

$RECHECK: Reset a few adverbs to the input values
PROC RECHECK; RASH=MRASH;DECSH=MDECSH;NMAP=MNMAP
 FLUX=MFLUX;NITER=MNIT;GAIN=MGAIN;DOPOL=MDOPOL;IMSIZ=MIMSIZ
 BMAJ=MBMAJ;BMIN=MBMIN;BPA=MBPA;SOLINT=MSOLI;NBOX=MBOX
 UVRANG=MUVRA;FLDSI=MFLDS;DOTV=MDOT;FACTO=MFACT;SETMOD
 UVWTFN=MUVWT;CUTOF=CLSAV;DOINT=MDOIN
 INN=IN0N;INDI=IN0DI;INCL=IN0CL;INS=IN0S
RET;FIN

$GETANT: Get Reference antenna if not already set
$ The reference antenna is put in header by UVPRM.
PROC GETANT(REFANT)
 IF (REFANT<1) THEN HEAD('GET REFERENCE ANTENNA')
  SETUVD; BPARM=0; DOPOL=0; GWT('UVPRM')
  KEYW='UVPREFAN';GETH;REFANT=KEYV(1)
 END
RET;FIN

$GETSHIFT: Get ra and dec shift if requested.
$          Modifies MX inputs for a large field of view
$          First makes a map of large region around
$          the center of the observation. ONLY SHIFTS IF peak
$          is brighter than MINRMS times noise.
$ DOCENTER - If true find RASHIFT and DECSHIFT
BIGMAPA='# Making an Image with a large pixel size.  Peak in'
BIGMAPB='# next Image will be the center of de-convolutions'
PROC GETSHIFT
 IF (DOCENTER>0) THEN HEAD('CENTER MAP ON SOURCE')
  TYP BIGMAPA;TYP BIGMAPB;NITER=15;UVWTFN='NA';NBOX=0
  GAIN=.333;BMAJ=0;BMIN=0;FLDSI=0;PIXSI=2*PIXSI;MAMX;NBOX=MBOX
  PRPND; TYP '# SOURCE ',INN,', PEAK=',PEAK,'#';PRPND
  IF (PEAK>RMS*MINRMS) THEN MIDSHIFT;CLRZAP;END
  GAIN=MGAIN;UVWTFN=MUVWTFN;PIXSI=MPIXS
  BMAJ=MBMAJ; BMIN=MBMIN;FLDSI=MFLDSI
 END
RET;FIN

$GETCC: find positive clean components for self-cal
$         Mapname is assumed to be correctly set.
$ NUMCC - Output Number of clean components brighter than
$         Twice the first negative
$ FLUX  - Output Sum of flux of first NUMCC clean components
$         CCFND passes the number of CCs found via MAP
$         header NTHCC.
PROC GETCC(NUMCC,FLUX); HEAD('SELECT CLEAN COMPONENTS')
 FLUX=0;FACTOR=CCFACT;ECOUNT=0;INV=1;OUTV=1;GWT('CCMRG')
```

```
 INV=0;OUTV=0;GWT('CCFND');KEYW='NTHCC';GETH;NUMCC=KEYV(1)
 KEYW='SUMFLUX';GETH;FLUX=KEYV(1)
RET;FIN

$GETUVLOW: Find lower limit of UV data represented
$           by Clean components
$ FLUX - Input to UVPRM, is the brightness represented by
$           the image clean components.
$ UVLOW- Output Lower Limit of UV Range for self-cal (k-lambda)
$           limit uvlow to less than 25% of the UV range
PROC GETUVLOW(FLUX,UVLOW)
 SETUVD;UVRA=MUVRA;BPARM=0;BLVER=-1;SUBAR=0;GWT('UVPRM')
 KEYW='UVPFLMIN';GETH;UVLOW=KEYV(1)
 KEYW='UVPRAMAX';GETH;UVMAX=KEYV(1)
 UVLOW=MAX(MIN(UVMAX/4,UVLOW),UVMAX/20)/1000
RET;FIN

$DELMAPS: Delete multi-field MX CLEAN maps
$ MAP NAME, CLASS and SEQUENCE are assumed correct
$ First MAP is deleted and other fields if they exist
PROC DELMAPS; CLRZAP
 IF (NFIELD > 1 & INS>0) THEN
  FOR I = 2 TO NFIELD;INCL='?CLN*';CLRZAP;END;  END
RET;FIN

$DELAST:  Delete last map and self calibration
PROC DELAST; HEAD('SOLUTION NOT IMPROVING, DELETING')
 SETMAP;GETSEQ;IF (INS>1) THEN DELMAPS;END
 SETUVD;INCL='SCAL';GETSEQ;IF (INS>1) THEN CLRZAP;END;INS=0
RET;FIN

$DELPRE: Delete Previous map and self calibration
$ DOZAP: if true, delete, else save for diagnostics
$ LOOP : Only Delete self-cal after first loop
PROC DELPRE(LOOP); HEAD('SOLUTION IMPROVED')
 IF (DOZAP>0) THEN HEAD('DELETING PREVIOUS');SETMAP
  GETSEQ;IF (INS>1) THEN INS=INS-1;DELMAPS;END
  SETUVD;IF (OKSCAL>0) THEN INCL='SCAL';GETSEQ;END
  IF (INS>1&LOOP>1) THEN INS=INS-1;CLRZ;END;INS=0
  END
RET;FIN

$getpeak: return PEAK value of the map
PROC GETPEAK; KEYW='DATAMAX';GETH;PEAK=KEYV(1);RET;FIN

$POLARIT - creats I, Q and U maps and combines to make a plot
$  DOPOL - >0: make polarization plots.
$      Polarization maps do not require much cleaning,
$      so the number of iterations is one quarter of ICLN.
PROC POLARIT; GETPEA
 IF ((DOPOL>0)&(PEAK>MDOALL*RMS)) THEN FACTOR=MFACT
  HEAD('Q and U IMAGES');ZEROSP=0;FLUX=MAX(0,MFLUX)
  NITER=MNITER/4;STOKES='Q';MAMX;STOKES='U';MAMX;STOKE=''
  END
RET;FIN

$GETSCAL: Determine parameters for self-calibration
$         After first third of self-calibration passes,
$         use previously self-calibrated data for next
$         self-calibration.  (Input Image name must be set)
$ NUMCC - Number of clean components in model
$ UVLOW - Lower limit of UV range to self-calibrate
$ NSCAL - Number of self-calibration loops
$ FLUX  - Brightness containined in clean component model
$ NOLDUV- Number of loops to use original UV data for self cal
PROC GETSCAL; GETCC(NUMCC,FLUX)
  IF (LOOP<MAX(NSCAL/3,NOLDUV+1)) THEN
   UVCLAS=UV0CLAS; UVSEQ=UV0SEQ; END
  SMODEL=0;IN2N=INN;IN2CL=INCL;IN2S=INS;IN2DI=INDI
  GETUVLOW(FLUX,UVLOW);SCALIT
RET;FIN

$PSCAL: Point Source Model Self CALibration
$ZFLUX - Estimated point source model Flux
$       Source is assumed to be at the phase center
PROC PSCAL
 CLR2N;IN2DI=INDI;SMODEL=ZFLUX,0;NUMCC=0
 GETUVLOW(ZFLUX,UVLOW); SCALIT
RET;FIN

$GETMOD: Get model image for first self calibration
$ MSCNAM - Model Self Cal image name (IN2Name set by SETMOD)
$           If the model name is set, restart with model
$         - The next map set will use UVCLAS for imaging
MDLPNT='Starting Self-Cal with POINT MODEL'
MDLIMG='Starting Self-Cal with Input Image'
MDLFST='Creating Image for First Self-Calibration Loop'
PROC GETMOD; LOOP=0
 IF (MSCNAM<>'') THEN
  IF (MSCNAM='POINT') THEN HEAD(MDLPNT);PSCAL
  ELSE HEAD(MDLIMG);INN=MSCNA;INCL=MSCLA;INS=MSCS;INDI=MSCD
   GETRMS(RMS); GETSCAL;GETPEA;END;END
 FLU=MFLU;IF(NSCAL>0)THEN HEAD(MDLFST);FLU=-1;END
 NITE=MNIT;MAMX
 IF (DOINT>0) THEN GETTVB;SETMAP;DELMAPS;MAMX;END
 GETPEA;LASTRMS=RMS
RET;FIN

$MAPLOOP: Inner Deconvolution and Self-calibration loop.
$         Amplitude self-cal is done on 80 % of cycles.
$         Loop executes only if noise is getting smaller.
$         Shifts phase center again after first pass.
$ LOOP  - Execution number
$ NSCAL - Maximum number of self calibration executions
$ MFLUX - for all but last loop, clean only to first negative
$         source model component
$         Loop stops executing when RMS is significantly worse
$         Than previous execution. setting RMS=LASTRMS allows
$         another try with different Self-Cal parameters
$ PIXSI - Cell Size of image
$ DOSTOP- STOP executing loop
WEAKPEA='PEAK TOO FAINT TO SELF-CALIBRATE'
PROC MAPLOOP(LOOP); RMSLIM=RMSFAC*LASTRMS; PIXSI=MPIXS
 IF ((DOSTOP<=0)&(RMS<=RMSLIM)&(LOOP<=NSCAL)) THEN
  PRPND; PRINT '# LOOP ',LOOP,' OF ',NSCAL,'#';PRPND;SETMAP
  IF (PEAK<MINRMS*RMS) THEN DOSTOP=1;HEAD(WEAKPEA)
  ELSE GETSCAL;SETMAP;GETTVB
   IF (LOOP<NSCAL&DOSTOP<=0) THEN FLU=-1;MAMX
$   if rms worse, delete
    IF (RMS>RMSLIM) THEN DELAST
     IF (DOAMP<=0&DOALL>0) THEN DOAMP=1;RMS=LASTRMS;END
    ELSE OKSCAL=1
     IF (DOCENTER>0) THEN MIDSHIFT;END;DELPRE(LOOP)
   END; END; END; END
 LASTRMS=MIN(LASTRMS,RMS)
RET;FIN

$MAPIT: Main procedure for mapping and self-calibration
$ Inputs to MAPIT are set with the MAPIT Help file
$ i.e. INPUT MAPIT
$    The self calibration loop continues only if peak is
$    seven times the noise level.
$ MAPIT first attemps self-calibration based on the Fourier
$       Transform (FT) of the UV data.  If the FT fails, a
$       point source model is used for self-calibration.
$ Self Cal only if noise greater than MINRMS time RMS
MINRMS=8
PROC MAPIT
  CHECKIN; CLPIT; PLOTIT
$ ref ant, shift center, 1st s-cal
```

```
  GETANT(REFANT); GETSHIFT; GETMOD
$ If Self-calibrating
  IF (NSCAL>0) THEN FOR LOOP=1:NSCAL;MAPLOOP(LOOP);END;END
  DOPOL=MDOPOL;POLARIT;OUTS=0
$ final clean, clean-up
  FLUX=MFLUX;NITER=MNIT;MAMX
  IF (RMS<LASTRMS*RMSFAC) THEN DELPRE(NSCAL);END
  IF (CLFLUX>CLIM&DOZAP>0) THEN
   SETUVD;INCLA='UVCLIP';CLRZ;END
  IF (DOZAP>2&OKSCAL>0) THEN SETUVD;INCL='SCAL';CLRZ;END
  HEAD('DONE WITH SOURCE ==> '!!INNA); RECHECK
RET;FIN

$map parameters
XYSIZ=512; MGAIN=0.05; GAIN=MGAIN;ZEROSP=0; UVBOX=0
UVWTFN=''
DOCENTER = -1; DOPOL = -1; DOALL = -1
NUMVIS=1; LASTRMS=0.5; RMS=1; KEYV=1,1; ZRMS=1; PEAK=RMS
UVMAX=1; LOOP=1; NSCAL=1; PIXSI=1; UVFREQ=22E9; PIXV=PEAK

$DOINT: Do interactive set of boxes before MX
DOINT=-1
$DOSTOP:  Flag to stop self-cal + image loop
DOSTOP=-1

$FACTOR determines MX minor cycle clean stopping point.
$       a small value stops clean just before reaching
$       side-lobe level (FACTOR=0 => stop at level)
$       FACTOR=-1 => one component per minor cycle. (slow)
FACTOR=-.1

$RMSFAC is the rms noise factor for stopping the MAPLOOP
$RMSFAC if RMS is less than RMSFAC*LASTRMS the loop continues
$RMSFAC > 1 implies solution is allowed to get a little worse.
$       before the solution improves
RMSFAC=1.2
$CLIMIT is minimum source flux for CLIP
CLIMIT=0.04

$CCFACT determines the CC components to include in SCAL
$       model. Default is twice negative CC is minimum.
CCFACT=2

$Self-Calibration solution interval Factor Defaults
FOR I=1 TO 20; SOLFA(I) = 1/I; END

$"Tuned" Self-Calibration Solution Interval Factors
NSOLIN=10
SOLFA(1)=1  ;SOLFA(2)=3/4;SOLFA(3)=2/3;SOLFA(4)=1/2;
SOLFA(5)=1/3;SOLFA(6)=1/4;SOLFA(7)=1/5;SOLFA(8)=1/6;
SOLFA(9)=1/8;SOLFA(10)=1/10

$sctype is the self cal solution type used in mapit.
SCTYPE='L1'

$nolduv is number of loops to use orginal UVdata for self-cal
$after NOLDUV self-cals, the previous self-cal output is used as
$input to the next self-cal input.
NOLDUV=2

$general parameters
$ set keywords used to read map and uvdata headers
FOR I = 1 TO 7; MAPKEY(1,I)='CTYPE'!!CHAR(I);END
FOR I = 1 TO 7; MAPKEY(2,I)='NAXIS'!!CHAR(I);END
FOR I = 1 TO 7; MAPKEY(3,I)='CRVAL'!!CHAR(I);END

$ set stokes clean classes
BSTOKE= 'I','Q','U','V','L','R',' '
FOR I = 1 TO 6; STCLASS(I)=BSTOKE(I)!!'CLN';END
$default stokes is I
STCLASS(7) = 'ICLN'

GOODBX ='Enter >0 to Continue (<0 = re-select boxs) :'

DOCRT -1; PRNU -1; INEXT 'PL'; DOTV=-1; DOZAP=1

$clear messages
CLRMSG

$save for quick recovery
SAVE MAPIT
$turn on the message file
MSGKILL FALSE

$To execute MAPIT:
$ set INDISK, and the NAME of the INPUT UV data set.
$ set OUTDISK and BADDISK to speed execution.
$ TYPE:
$       MAPIT
$ to start execution

$restore previous parameters
$(FOLLOWING STEP WILL NOT WORK ON FIRST EXECUTION !!)
$(BUT DO NOT WORRY ABOUT IT !?!)
TGET MAPIT

$save again for quick recovery
SAVE MAPIT
```