BEFORE WE START

- Download the latest version of scripts from:

  almascience.org/euarcdata/itrain06/

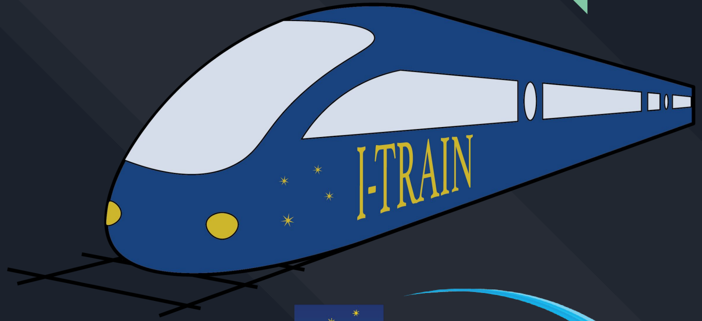
  `itrain-selfcal.py`

  `itrain-selfcal-line.py [ADVANCED]`

- If you cannot finish imaging loops on time to follow the live tutorial, you can download the corresponding gaintables, put them in your working folder and continue with the next steps:

  `caltables.tar.gz (~1MB)`

I-TRAIN #6: Improving Image Fidelity Through Self-Calibration



E. Moravec - Czech ARC node
A. M. S. Richards - UK ARC node
A. Pérez-Sánchez - Allegro ARC node
M.C. Toribio - Nordic ARC node

ALMA
EUROPEAN ARC
ALMA Regional Centre

asu
Astronomical Institute
of the Czech Academy of Sciences

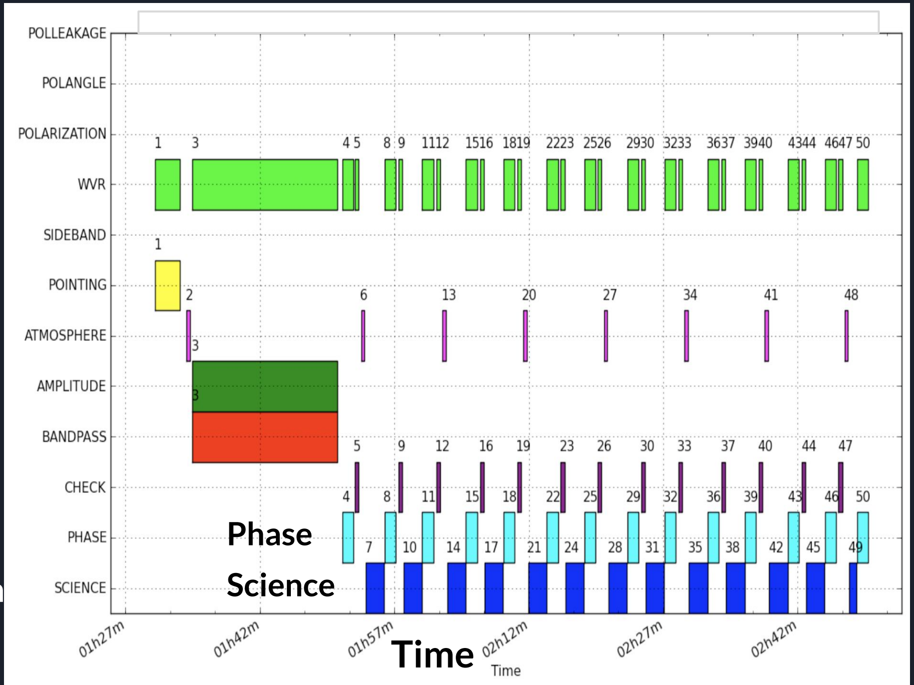
MANCHESTER 1824
The University of Manchester

NOVA
Sterrewacht Leiden
NWO

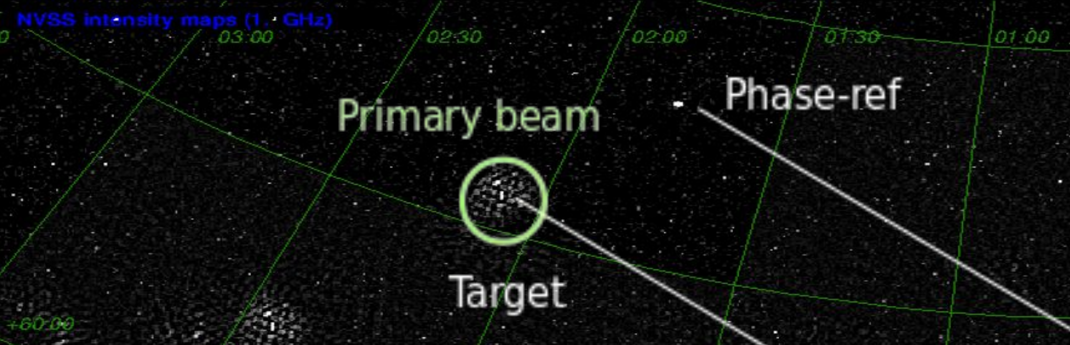
CHALMERS

Radio Interferometry.
Observing & Calibration strategy:

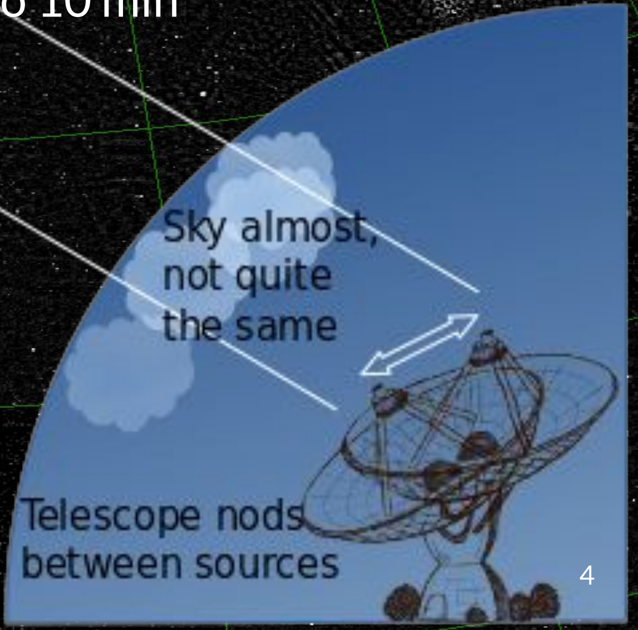
- Apply instrumental corrections
  - ALMA: WVR, Tsys etc.
    - Usually done by observatory pipeline
- Apply corrections from astrophysical sources
  - Flux scale bandpass, phase referencing etc.
    - Similar from cm to sub-mm interferometry
- Phase referencing
  - Source close to target (few degrees)
    - Point-like or well known structure ‘model’
    - Compare actual visibilities with ideal model
    - Derive corrections (per antenna)
  - Apply the solutions to the target
    - Assume sky refraction, absorption & emission is the same



Self-Calibration



- ALMA nods between phase-ref and target every 0.5 to 10 min
- Phase-ref - target separation ~1° - 10°
  - Faster, closer, at high frequencies
- Sky changes in time and direction
- Residual errors in target phase and amplitudes after applying phase-ref corrections
- Could be solved by a closer phase-reference
- The target itself!
  - If it is bright enough



What is self calibration? (self-cal)

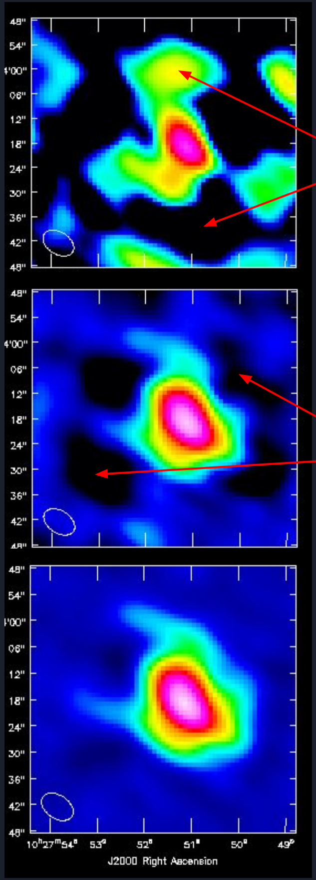
Calibration of the amplitudes and phases by using the source ‘itself’

After Calibration, Images Can Still Have Errors

- Phase errors:
  - Emission is smeared (like bad ‘seeing’)
    - Astrometry is degraded
  - Visibility amplitudes are decorrelated
    - Fluxes are reduced
  - Weak emission undetectable
  - Excess noise (but maybe not much)
  - Anti-symmetric artefacts in image

- Amplitude errors:
  - Spotty or stripy emission
    - Could also be bad data
  - Fluxes reduced
  - Noise increased
  - Symmetric artefacts in image

How can we correct for these errors? Self Calibration!



Phase-reference solutions only Opposite -ive & +ive artefacts **- phase-**errors dominate

Self-calibration phase only Symmetric artefacts remain - amp. errors

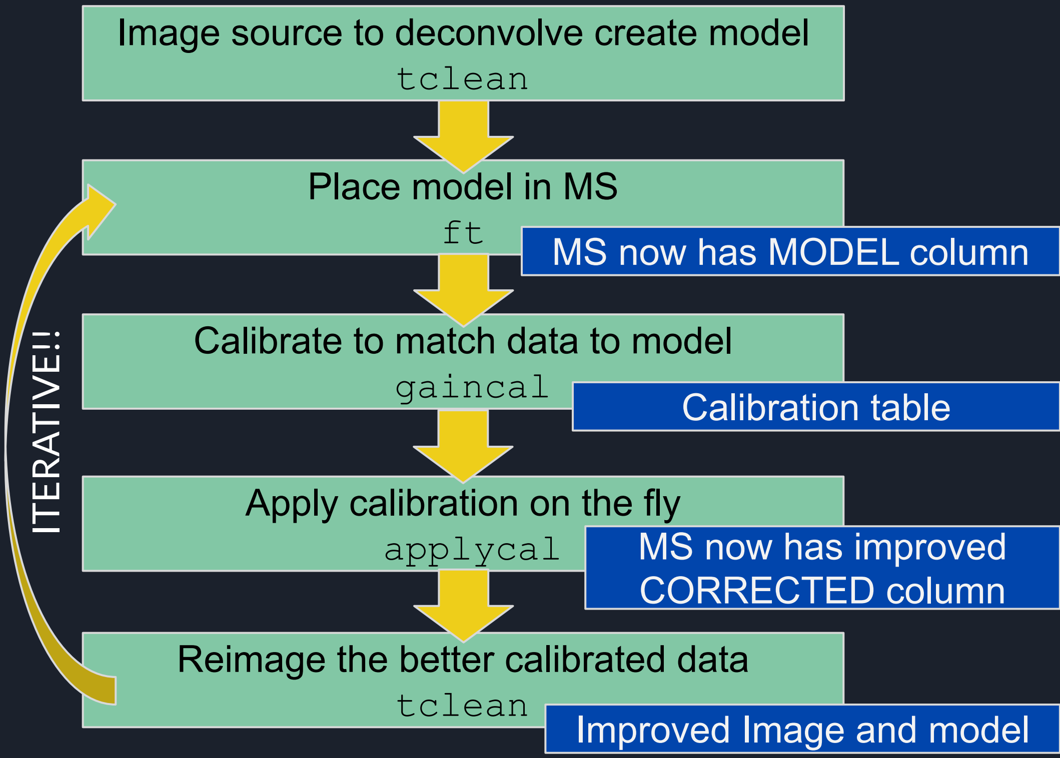
Self-calibration phase followed by amplitude

Data from CASA Guide NGC 3256

How to Implement Self-Calibrate: CASA

1st round phase calibration

1. Conservative clean to produce model (tclean)
2. Save model (ft) **safest way**
3. Investigate solution intervals with plotms to get a sense for the data (gaincal)
4. Choose solution interval (gaincal)
5. Apply calibration table (applycal)
6. Iterate



Adapted from: https://science.nrao.edu/facilities/alma/naasc-workshops/alma_dr/ALMA_ImagingSelfcal_AKL.pdf

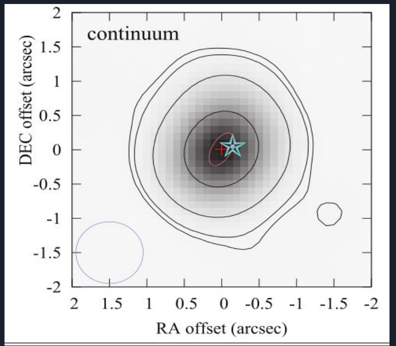
When can you self-calibrate your data?

Initial S/N: from delivered sample image (or make your own target image with phase cal solutions applied).
Predicted S/N: estimate target flux from e.g. publications, and ideal sensitivity from listobs time on source etc.

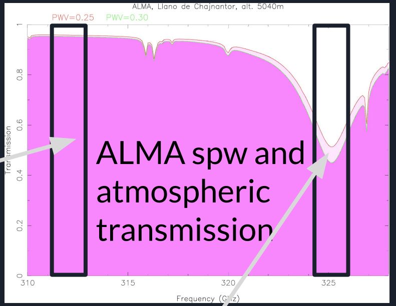
1. Target image has rms noise > predicted
2. Target has ‘bright enough’ peak (see Advanced)
    - Need image signal/noise (S/N) few 100 for ALMA with ~40 antennas in typical continuum
        - Fewer antennas - lower S/N needed
            - 20 antennas for VY CMa
    - Predicted S/N is ‘ideal’, may be lower at first in initial image
    - Measure peak in Jy/beam
        - Does not have to be in the centre of the field
        - It’s OK if there is additional, weaker flux

Dataset: VY CMa

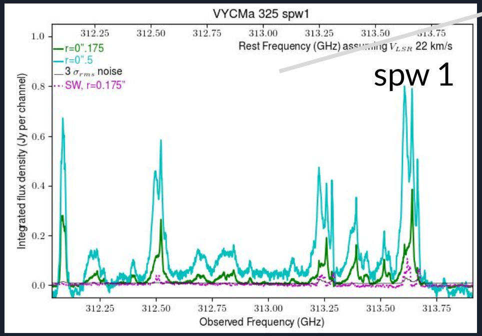
- Science verification data taken in 2013 to test baselines ⩾2.7 km at Band 7
  - Data available in ALMA archive
  - Richards et al. 2014, O Gorman et al 2015 etc.
    - NB For speed, this tutorial used just one EB, one tuning out of many
  - 2 x 1.875 GHz spectral windows, each with 1920 channels
  - Spectral resolution 0.9 km/s
- VY CMa is a Red Supergiant star at ~1.2 kpc
  - Compact stellar continuum expected to be ~100 mJy at 325 GHz
  - Even more extended flux from irregularly distributed dust
    - Previous barely-resolved SMA image
- Thick wind, bright molecules and masers up to hundreds Jy
  - Total extent of circumstellar envelope ~12”

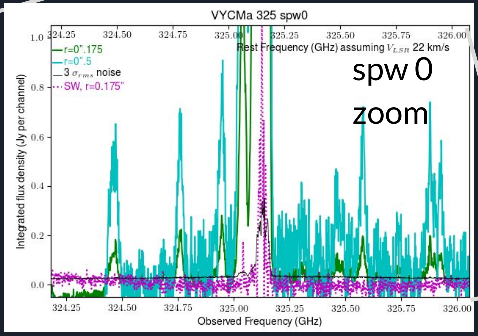


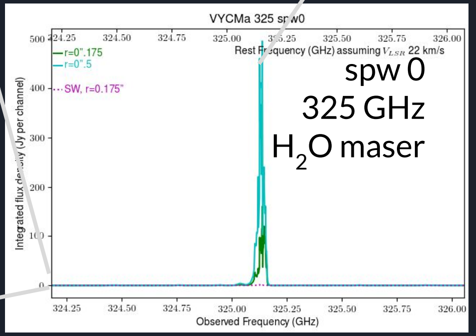
Kaminski et al. 2013
~1” resolution SMA



VY CMa ALMA spectra







Why does this dataset need self-cal? Why is it a good candidate for self-cal?

- Adding up scan times in listobs, 31.5 min on source, 20 antennas
- Guess 1 GHz line free (out of 3.75 GHz total bandwidth)
- Lower frequency spw centre ~313 GHz
- Use ALMA Sensitivity Calculator - predict 0.2 mJy continuum rms(ideal)
  - Try for yourself: more accurate continuum b/w, what difference does it make close to 325 GHz?
- Continuum flux density predicted stellar peak ~100 mJy/beam so S/N(ideal) ~500
  - Should be plenty to self-calibrate
- Does it need it?
  - Check S/N when you made made first image
  - Inspect target phase as a function of time
    - You expect the phase of a slightly-extended target to vary slowly with time
    - Rapid changes, different for different antennas, are probably atmospheric errors
- Maser line also very bright, could use that (but see later for why we start with continuum)

START OF LIVE DEMO

Live Demo: Open CASA

Launch CASA from your Applications icon or from the terminal with :

`casapy` Or `casa`

depends on your alias setup

```
IPython 7.11.1 -- An enhanced Interactive Python.

Using matplotlib backend: MacOSX
Telemetry initialized. Telemetry will send anonymized usage statistics to
 NRAO.
You can disable telemetry by adding the following line to the config.py f
ile in your rcdir (e.g. ~/.casa/config.py):
telemetry_enabled = False
--> CrashReporter initialized.
CASA 6.1.1.15 -- Common Astronomy Software Applications [6.1.1.15]
```

Script for this tutorial

The tutorial script `itrain-selfcal.py` is organized in STEPS.

- Search for the variable `step_title` that holds all the steps executed by the script.

```
#==========================================================================
# STEPS
#==========================================================================

# List of steps executed by this script
thesteps=[]
step_title = {0:  'List the data set and plot antennas and visibility spectrum',
              1:  'Make dirty image of continuum',
              ### INITIAL MODEL
              2:  'Make an initial, conservative cleaning',
              3:  'Check and save model',
              ### FIRST ROUND OF SELF-CALIBRATION - PHASE
              4:  'Calculate gain solution table - phase-only, solution interval = scan-length',
              5:  'Explore different solution intervals',
              6:  '[ADVANCED] Calculate SNR of the different solution intervals',
              7:  'Apply calibration table',
              8:  'Make second, conservative cleaning and save model',
              ### SECOND ROUND OF SELF-CALIBRATION - PHASE
              9:  'Explore different solution intervals',
```

Script for this tutorial

The variable `mysteps` controls the steps that will be executed

- If not defined, all steps will be executed
- To run steps of your choice, define mysteps before executing the script, e.g.:

```
CASA <1>: mysteps=[0,1]

CASA <2>: execfile('itrain-selfcal.py')
List of steps to be executed ... [0, 1]
Step  0 List the data set and plot antennas and visibility spectrum
```

Set of Variables

Some variables are defined in the script that are tailored to this dataset, among others:

```
visname = 'X1de2_VYCMa_325'

field = 'Vy Cma'

refantenna='DV15'

contchans='0:5~224;365~379;425~479;615~739;1185~1249;1460~1569;1615~1699;1820~1919,1:0~214;440~475;487~494;580~609;750~789;830~1049;1305~1364;1490~1529;1610~1659;1775~1829;1815~1834;1895~1919'
```

(1) Beforehand you should have executed Steps 0 and 1.
mysteps = [0,1]

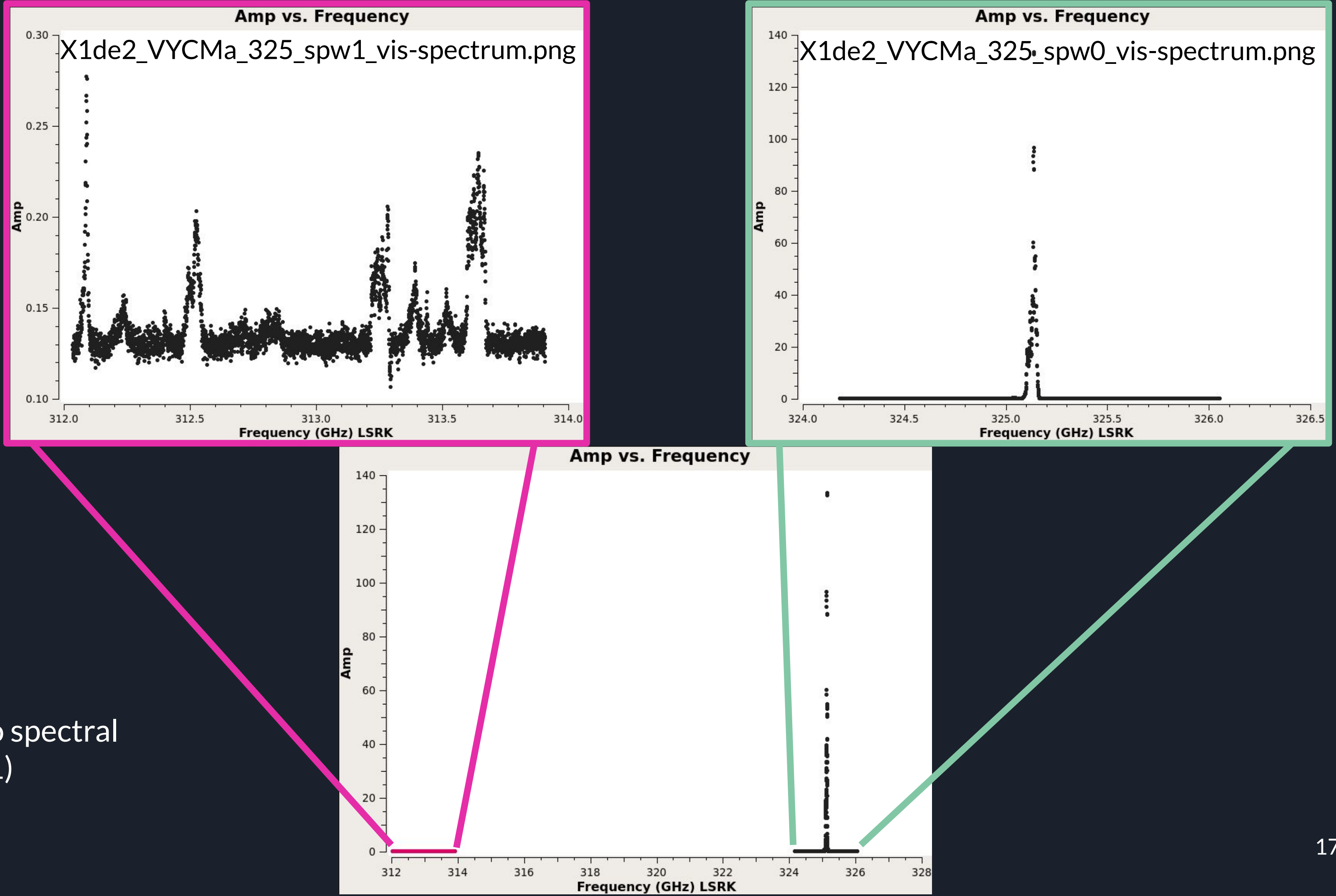
Step 0 = listobs, visibilities for spws, plotants

Step 1 = dirty image

```
CASA <1>: mysteps=[0,1]


CASA <2>: execfile('itrain-selfcal-casa6.py')

List of steps to be executed ... [0, 1]
Step  0 List the data set and plot antennas and
visibility spectrum
```



Dataset: Two spectral windows (0,1)

Dataset: listobs()

listobs(vis=obj, listfile='X1de2_VYCMa_325_listobs.txt', verbose=True)

```
================================================================================
   Observer: violette     Project: uid://A002/X6ac013/X13
Observation: ALMA
Data records: 61560       Total elapsed time = 2834.88 seconds
   Observed from   17-Aug-2013/11:15:15.0   to   17-Aug-2013/12:02:29.9 (UTC)

   ObservationID = 0         ArrayID = 0
  Date        Timerange (UTC)          Scan  FldId FieldName             nRows     SpwIds   Average Interval(s)    ScanIntent
  17-Aug-2013/11:15:15.0 - 11:19:59.5     8      0 Vy Cma                    9234  [0,1]  [10.1, 10.1] [CALIBRATE_WVR#ON_SOURCE,OBSERVE_TARGET#ON_SOURCE]
              11:22:24.6 - 11:27:40.8    11      0 Vy Cma                   10260  [0,1]  [10.1, 10.1] [CALIBRATE_WVR#ON_SOURCE,OBSERVE_TARGET#ON_SOURCE]
              11:30:27.3 - 11:35:43.5    15      0 Vy Cma                   10260  [0,1]  [10.1, 10.1] [CALIBRATE_WVR#ON_SOURCE,OBSERVE_TARGET#ON_SOURCE]
              11:38:09.6 - 11:43:25.8    18      0 Vy Cma                   10260  [0,1]  [10.1, 10.1] [CALIBRATE_WVR#ON_SOURCE,OBSERVE_TARGET#ON_SOURCE]
              11:46:13.0 - 11:51:29.2    22      0 Vy Cma                   10260  [0,1]  [10.1, 10.1] [CALIBRATE_WVR#ON_SOURCE,OBSERVE_TARGET#ON_SOURCE]
              11:53:55.3 - 11:59:11.5    25      0 Vy Cma                   10260  [0,1]  [10.1, 10.1] [CALIBRATE_WVR#ON_SOURCE,OBSERVE_TARGET#ON_SOURCE]
              12:01:59.7 - 12:02:29.9    29      0 Vy Cma                    1026  [0,1]  [10.1, 10.1] [CALIBRATE_WVR#ON_SOURCE,OBSERVE_TARGET#ON_SOURCE]
           (nRows = Total number of rows per scan)
Fields: 1
  ID   Code Name                RA               Decl           Epoch        nRows
  0    none Vy Cma              07:22:58.334544 -25.46.03.32752 J2000        61560
Spectral Windows:  (2 unique spectral windows and 1 unique polarization setups)
  SpwID  Name                          #Chans   Frame   Ch0(MHz)  ChanWid(kHz)  TotBW(kHz)  CtrFreq(MHz) BBC Num  Corrs
  0      ALMA_RB_07#BB_1#SW-01#FULL_RES  1920   LSRK  324180.573       976.594   1875060.2 325117.6144        1  XX  YY
  1      ALMA_RB_07#BB_3#SW-01#FULL_RES  1920   LSRK  313908.766      -976.594   1875060.2 312971.7242        3  XX  YY
The SOURCE table is empty: see the FIELD table
Antennas: 19:
  ID   Name  Station   Diam.    Long.         Lat.                Offset from array center (m)              ITRF Geocentric coordinates (m)
                                                                     East         North     Elevation               x               y               z
  0    DA41  A079      12.0 m   -067.45.13.6  -22.53.35.0        116.8295     -920.3183       22.7141  2225122.719081 -5439951.198760 -2481886.540940
  1    DA46  A067      12.0 m   -067.45.12.7  -22.53.27.2        142.4098     -678.7313       20.1277  2225181.070626 -5440026.290738 -2481662.974509
  2    DA50  A045      12.0 m   -067.45.17.9  -22.53.30.1         -5.4173     -767.4396       22.6030  2225032.052439 -5440052.425350 -2481745.659670
  4    DV03  A137      12.0 m   -067.45.15.2  -22.53.22.7         71.1262     -540.4365       20.6157  2225135.630597 -5440103.481224 -2481535.759629
  5    DV04  A004      12.0 m   -067.45.15.9  -22.53.28.0         52.6623     -704.4157       21.2709  2225094.623137 -5440051.993958 -2481687.080943
  6    DV05  A082      12.0 m   -067.45.08.3  -22.53.29.2        269.0432     -740.9511       15.7824  2225287.593514 -5439952.243342 -2481718.604105
  7    DV07  A096      12.0 m   -067.45.29.9  -22.53.15.7       -347.1447     -322.7951       22.7962  2224781.300787 -5440342.022417 -2481336.102204
  8    DV11  A031      12.0 m   -067.45.19.1  -22.53.27.1        -37.8146     -675.5186       21.7276  2225015.298360 -5440097.037593 -2481660.637147
  9    DV13  A072      12.0 m   -067.45.12.6  -22.53.24.0        147.1745     -580.5880       18.1820  2225199.254542 -5440058.161226 -2481571.802858
  10   DV14  A025      12.0 m   -067.45.18.7  -22.53.27.4        -26.4283     -685.5216       21.7069  2225024.356544 -5440089.108343 -2481669.844283
  11   DV15  A074      12.0 m   -067.45.12.1  -22.53.32.0        161.8158     -828.6186       18.7685  2225176.483458 -5439963.820617 -2481800.528766
  12   DV17  A138      12.0 m   -067.45.17.1  -22.53.34.4         19.1461     -901.2602       26.0137  2225036.268961 -5439997.853078 -2481870.267488
  13   DV18  A080      12.0 m   -067.45.14.7  -22.53.20.2         87.4842     -461.2347       20.6299  2225162.438024 -5440125.813526 -2481462.799597
  14   DV19  A117      12.0 m   -067.45.52.4  -22.53.24.2       -988.6155     -585.4402        8.8592  2224144.040254 -5440478.348582 -2481572.646718
  15   DV20  A020      12.0 m   -067.45.17.8  -22.53.28.0         -2.9646     -703.4387       21.6624  2225043.419137 -5440073.737486 -2481686.333091
  16   DV21  A015      12.0 m   -067.45.15.3  -22.53.26.0         68.8266     -640.1811       21.0213  2225118.955833 -5440068.788075 -2481627.807707
  17   DV22  A011      12.0 m   -067.45.14.4  -22.53.28.4         95.9130     -716.4998       21.0897  2225132.810214 -5440031.115688 -2481698.142940
  18   DV24  A131      12.0 m   -067.44.27.8  -22.54.05.7       1423.9491    -1870.2797       47.6262  2226201.105359 -5439135.401842 -2482771.346444
  19   DV25  A106      12.0 m   -067.45.14.0  -22.53.02.5        105.8174       86.4720       24.7033  2225261.468394 -5440319.498942 -2480959.792268
```

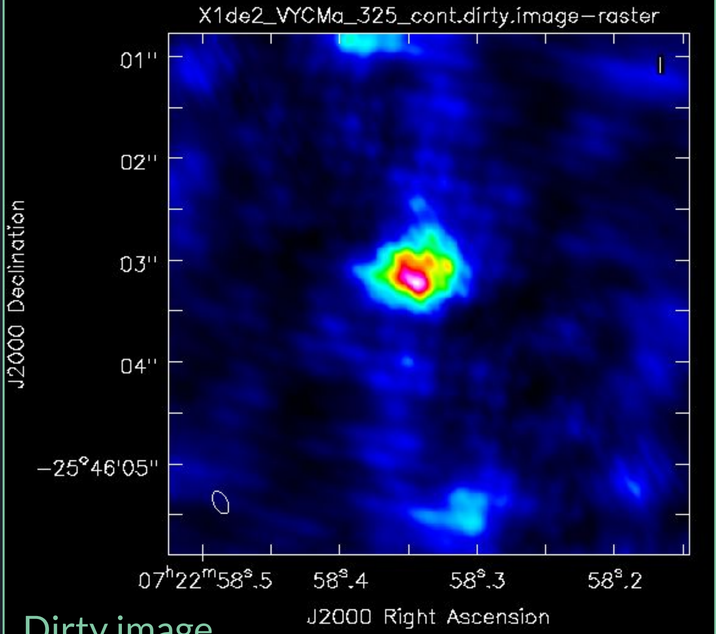
This tutorial: continuum self-cal

In a dataset with spectral lines, you need to find all the channels within the spectral window(s) of the Science target that do not contain spectral features (lines in emission or absorption).

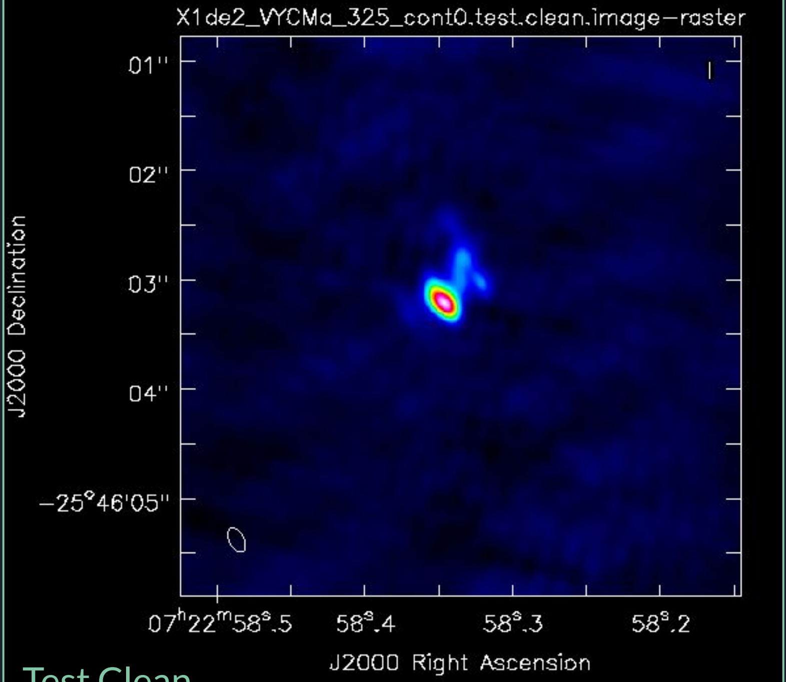
In this tutorial this has been done for you and contained in `‘contchans’`

From now on, we will be doing all analysis with only the continuum channels!

What does an image look like? (Step 1 Dirty image)



Dirty image



Test Clean

Live Demo: 1st Round of Self-Cal

A. Conservative clean to produce model (`tclean`) (Step 2)
B. Save model (`ft`) (Step 3)
C. Choose solution interval and create solution table (`gaincal`) (Step 4)
D. Investigate solution intervals with plotms to get a sense for the data (`gaincal`) (Step 5)
E. Apply calibration table (`applycal`) (Step 7)

**this tutorial does C then D but when you do this for your datasets should do D then C.

(2) Interactive conservative clean (tclean)

Create image products of the continuum channels of the selected SPW(s) and create/build model which will be the basis of self-cal.

What is a conservative clean?

- Masking what you are CERTAIN is real emission. **DO NOT mask noise blobs**, otherwise the model created could include fake features in the self-calibrated products.

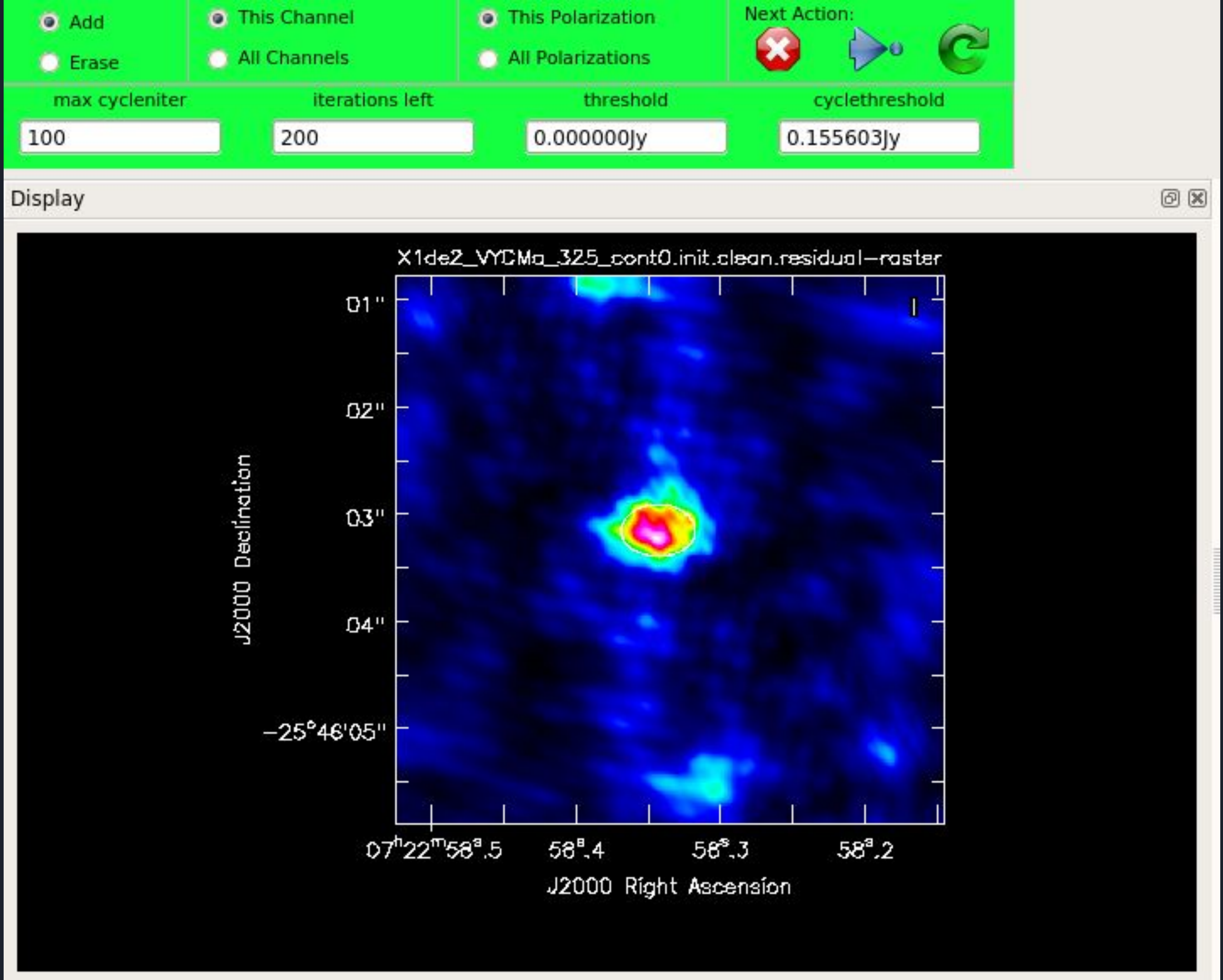
Basic set of parameters for first **TCLEAN**: Building the Model.
vis, spw,imsize,cellsize, specmode, savemodel

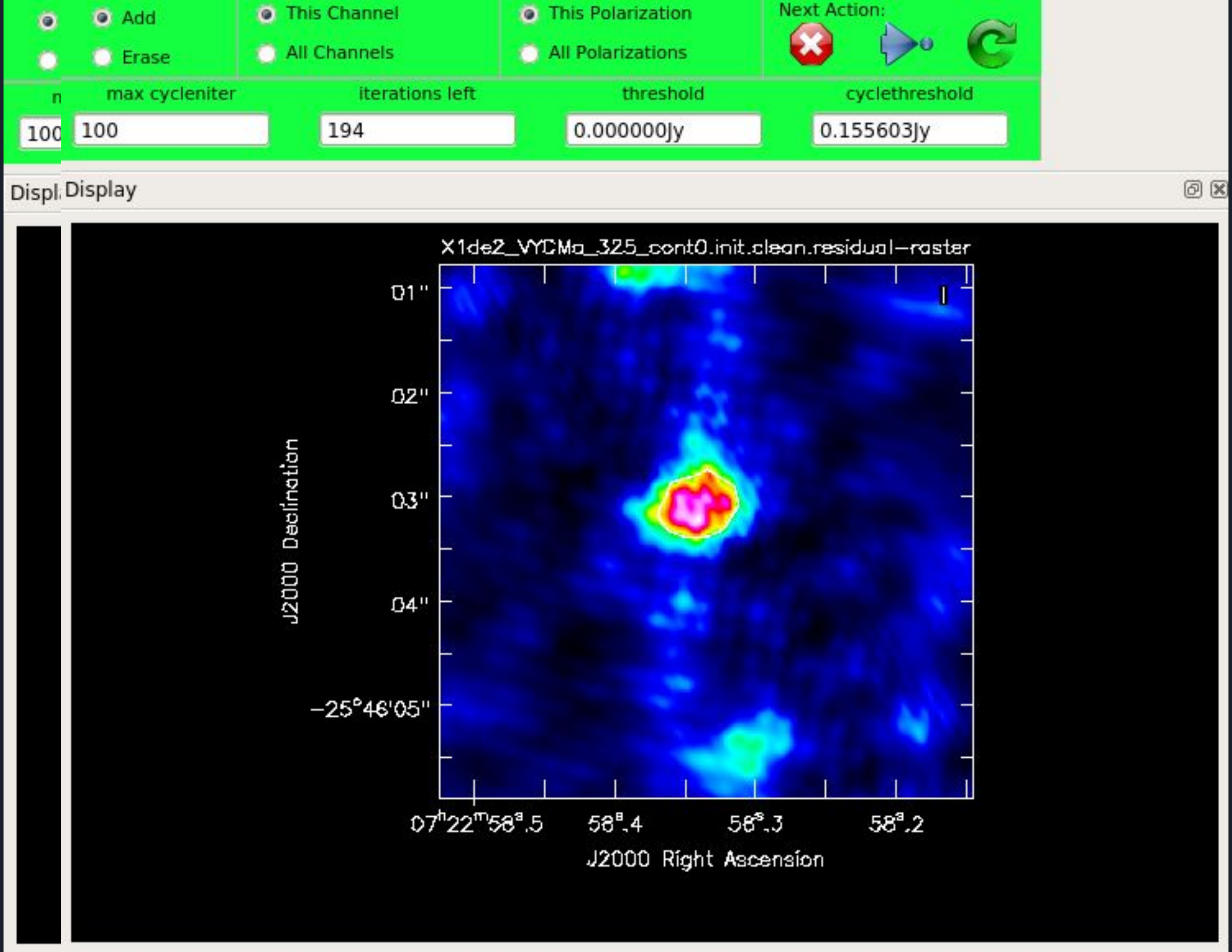
Output: image products of the continuum channels of the selected SPW(s).

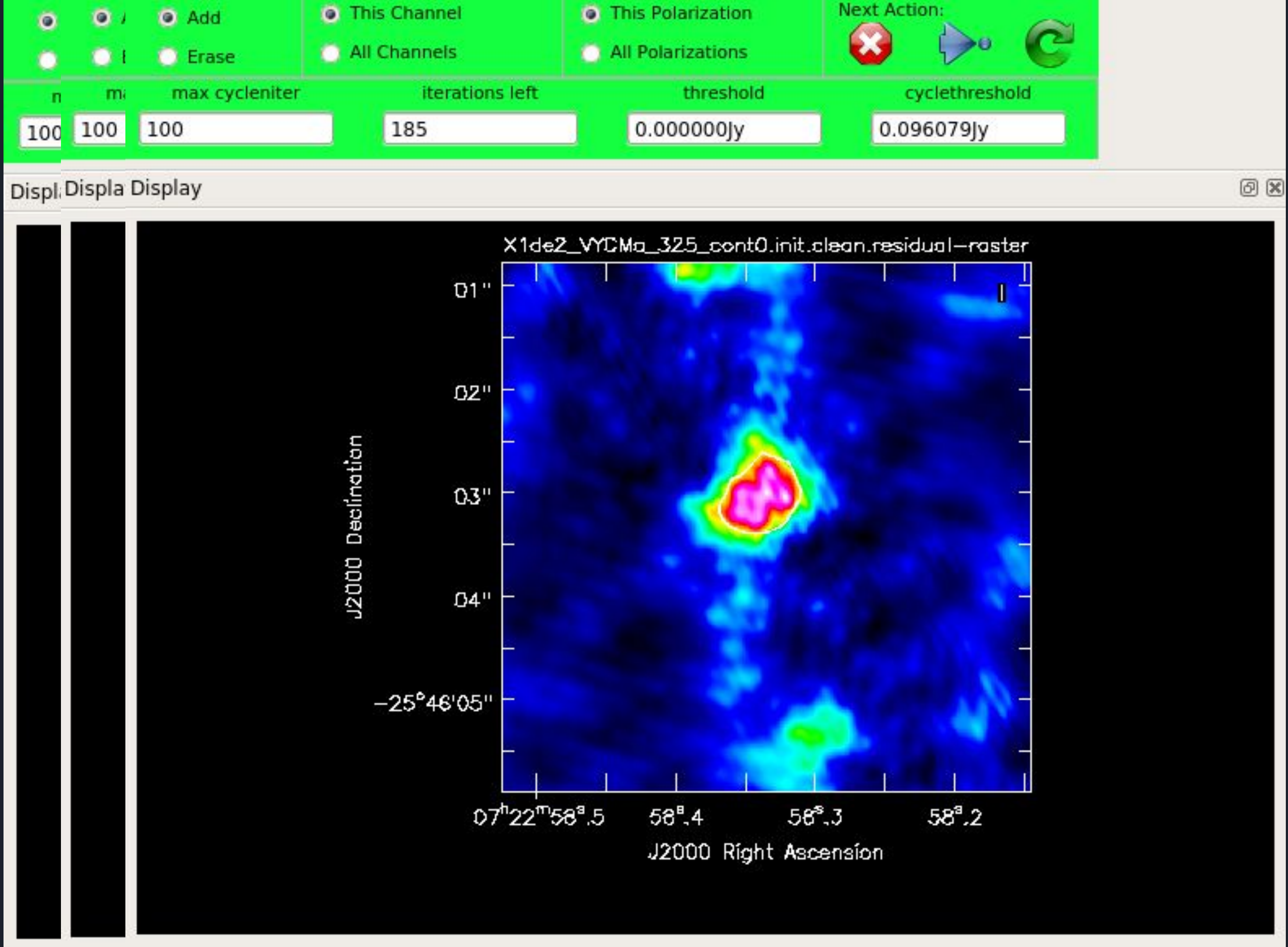
IMPORTANT to check imagename.model. If such file is not created, set number of iterations to zero; i.e. niter=0, and re-run the TCLEAN command.

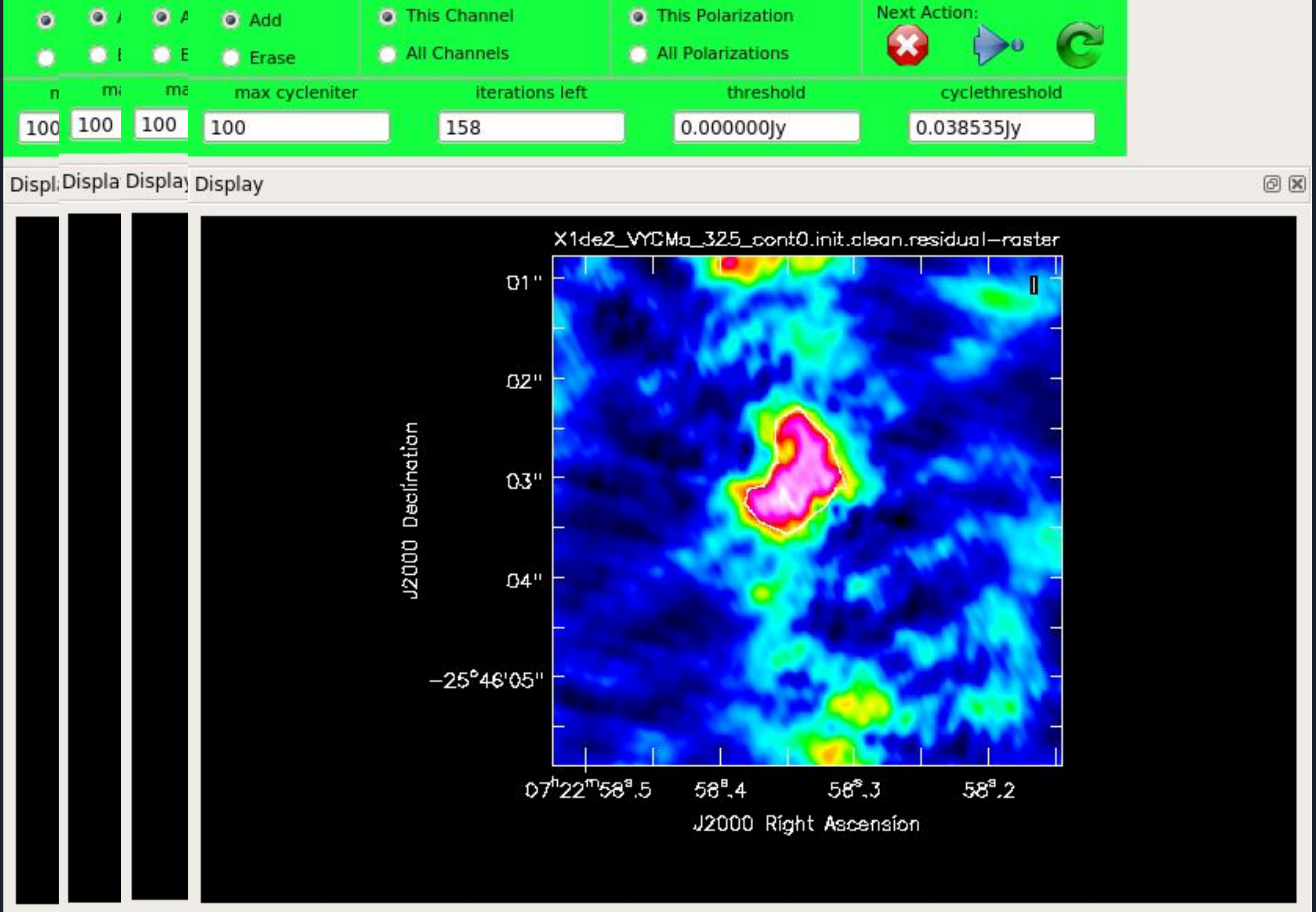
(2) Interactive conservative clean (tclean)
mysteps = [2]

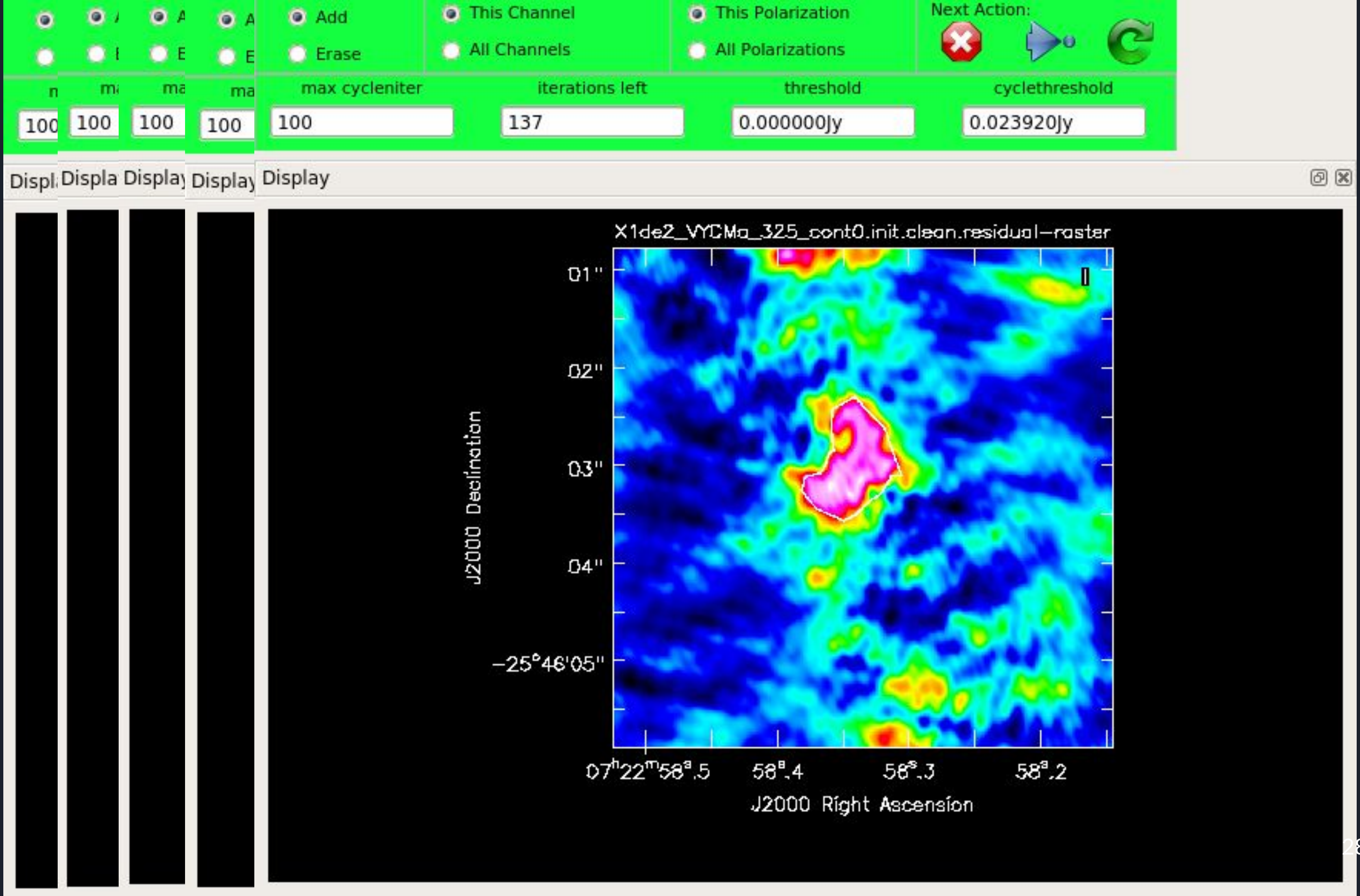
```
tclean(vis = vis,
    imagename='X1de2_VYCMa_325_cont0.init.clean',
    field=field,
    spw=contchans,
    specmode='mfs',
    cell='0.01arcsec',
    imsize=512,
    niter=200,
    interactive=True)
```

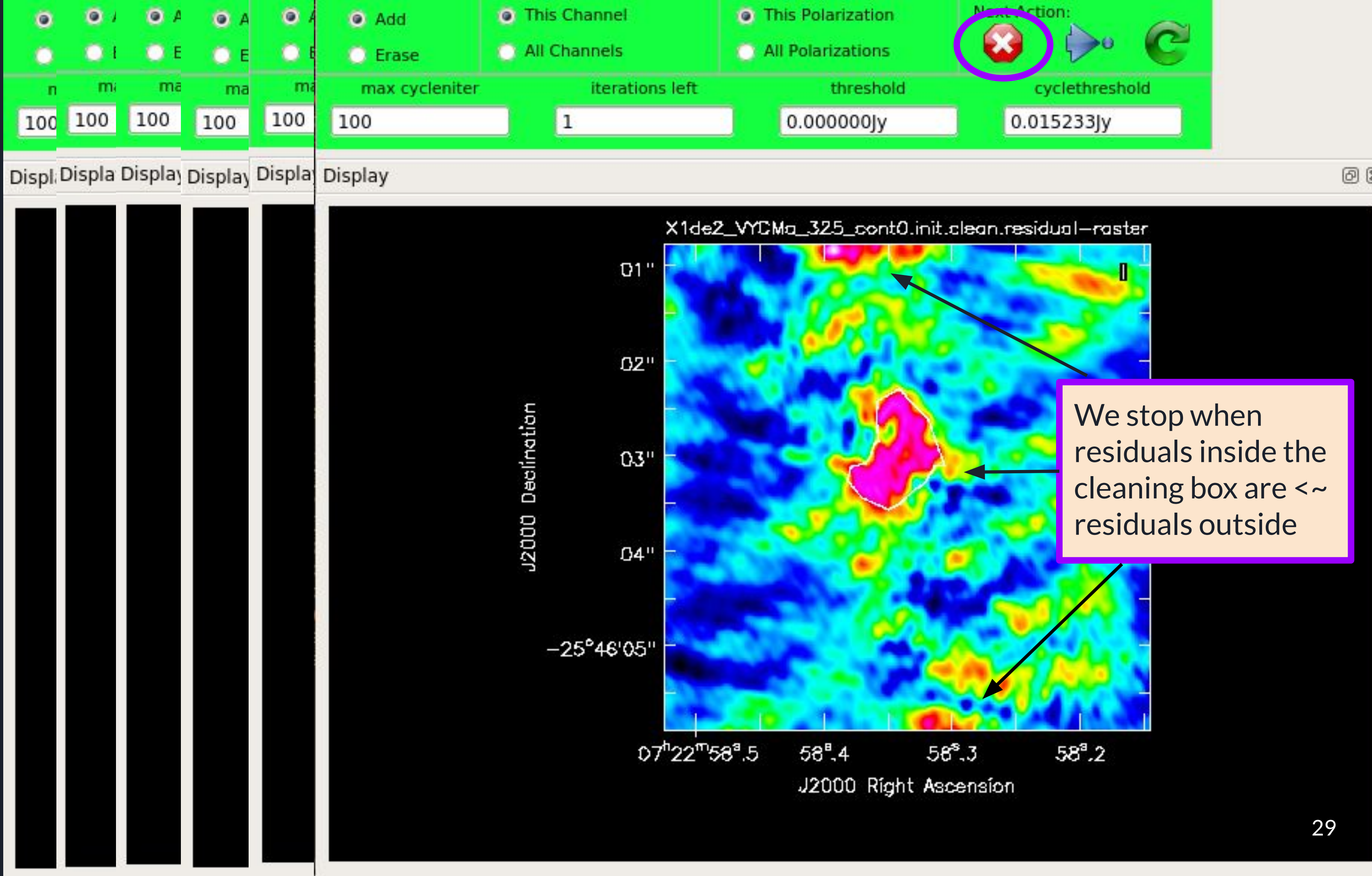

Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
200
threshold
0.000000Jy
cyclethreshold
0.155603Jy
Display
X1de2_VYCMa_325_cont0.init.clean.residual-raster
01"
02"
03"
04"
−25°46'05"
J2000 Declination
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension


Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
194
threshold
0.000000Jy
cyclethreshold
0.155603Jy
Display
X1de2_VYCMa_325_cont0.init.clean.residual-raster
J2000 Declination
J2000 Right Ascension


Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
iterations left
threshold
cyclethreshold
100
100
100
185
0.000000Jy
0.096079Jy
Display
X1de2_VYCMa_325_cont0.init.clean.residual-raster
J2000 Declination
J2000 Right Ascension
-25°46'05"
58s.4
58s.3
58s.2

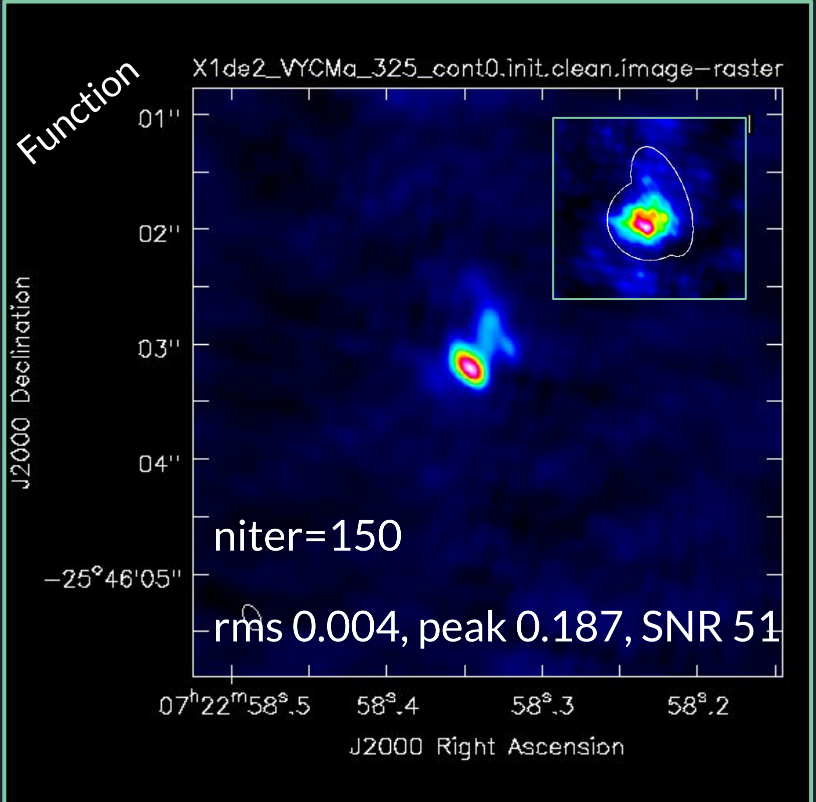

Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
158
threshold
0.000000Jy
cyclethreshold
0.038535Jy
Display
X1de2_VYCMa_325_cont0.init.clean.residual-raster
J2000 Declination
01"
02"
03"
04"
-25°46'05"
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension

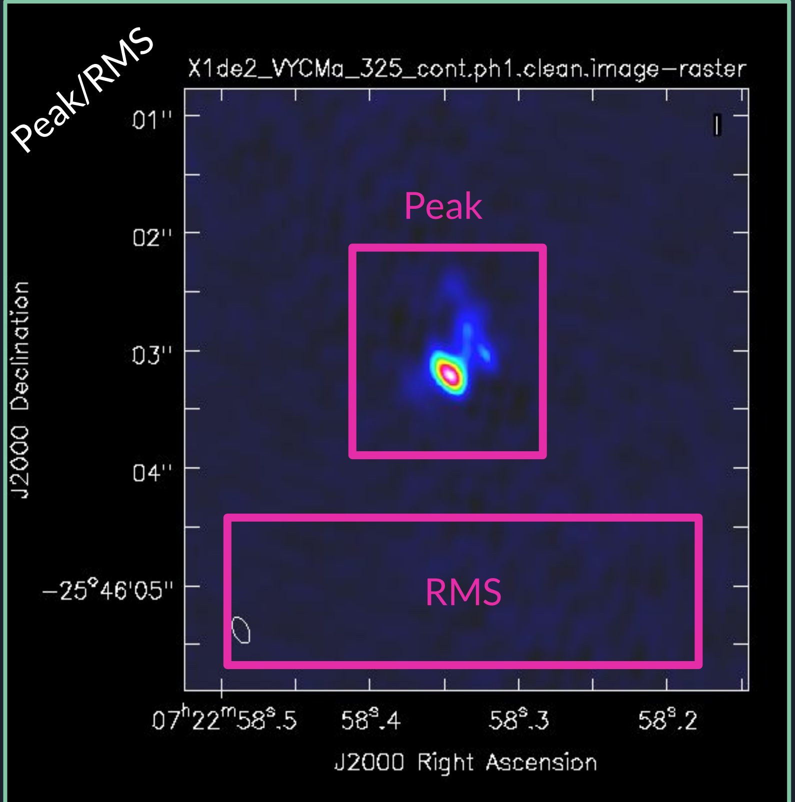

Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
137
threshold
0.000000Jy
cyclethreshold
0.023920Jy
Display
X1de2_VYCMa_325_cont0.init.clean.residual-raster
01"
02"
03"
04"
-25°46'05"
J2000 Declination
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension


Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
max cycleniter
iterations left
threshold
cyclethreshold
100
1
0.000000Jy
0.015233Jy
Display
X1de2_VYCMa_325_cont0.init.clean.residual-raster
J2000 Declination
J2000 Right Ascension
We stop when residuals inside the cleaning box are <~ residuals outside

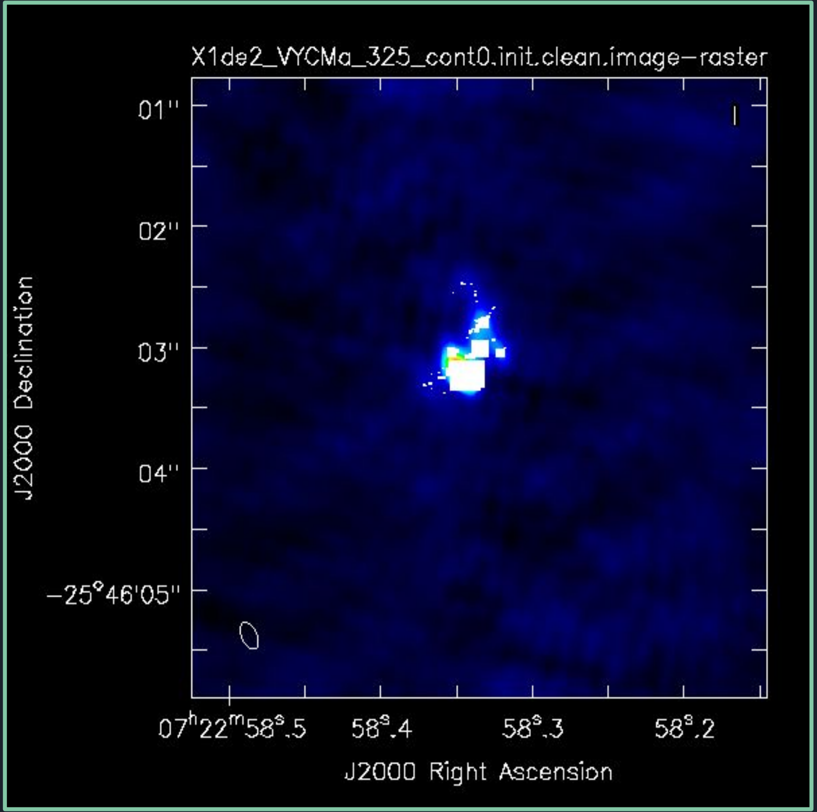
(2) Initial Image Statistics

* Can use the function defined in script (left image). But a fast way is to compare RMS box (*ideally a relatively large box with no emission that is representative of the noise in the image*) and peak of the emission (right image)





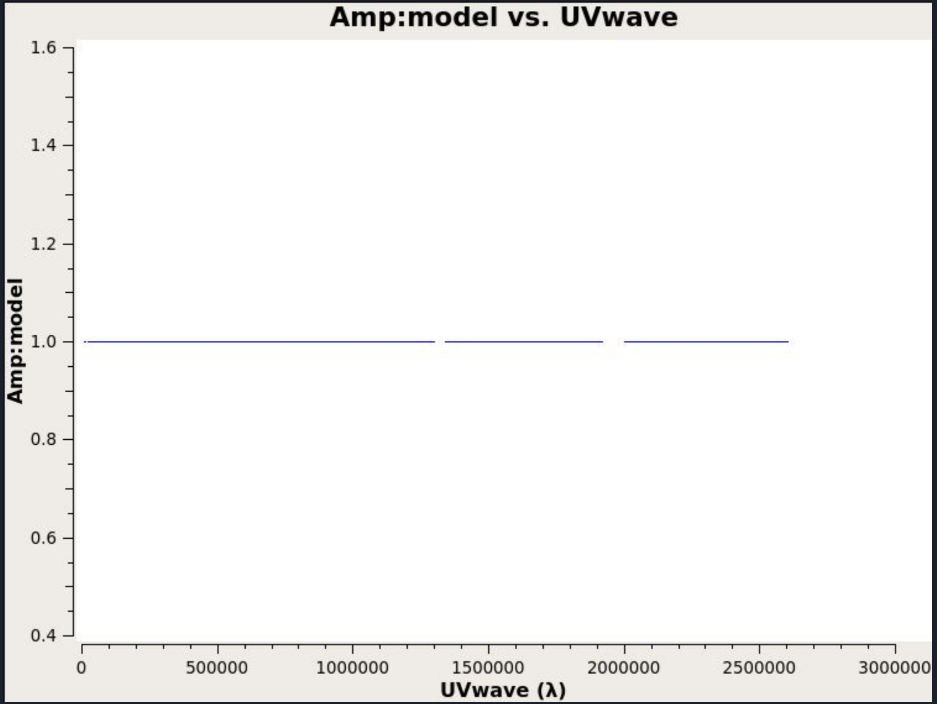
The Model



X1de2_VYCMa_325_cont0.init.clean.image−raster
01''
02''
03''
04''
−25°46'05''
J2000 Declination
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension

(3) Check that the model has saved (plotms)
mysteps = [3]

```
plotms(vis=vis,
xaxis='UVwave',
yaxis='amp',
ydatacolumn='model',
showgui=False,
plotfile='X1de2_VYCMa_325_
cont0.init.clean.model.png')
```



Forcing a model to save

Several ways of saving a model in CASA:

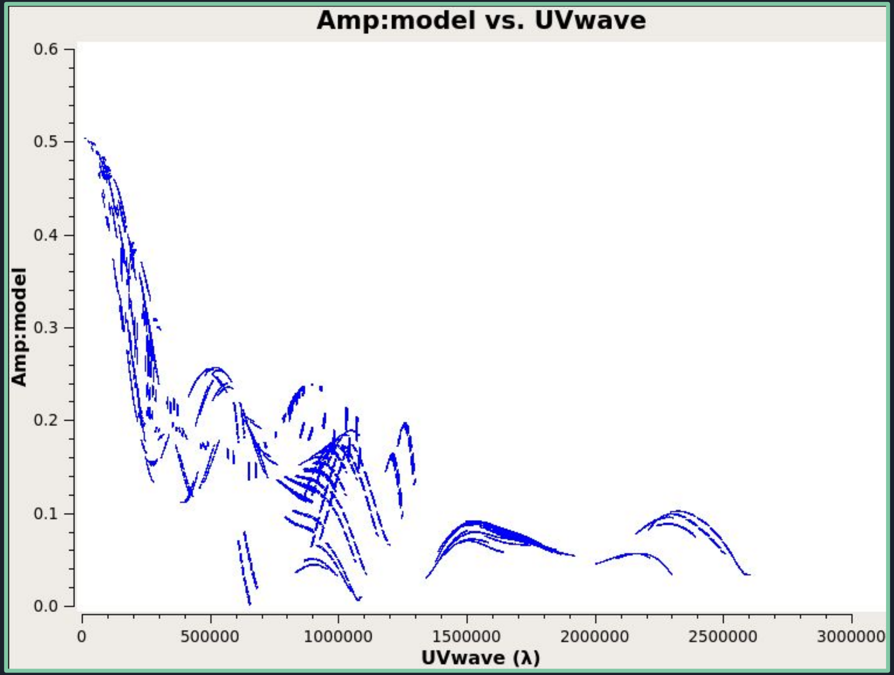
1. tclean (savemodel=True)
2. ft()
3. Interactive -- set niter = 1 and press 
4. Run tclean with niter=0,1

We have found that `ft` is the “safest” and quickest way. ALWAYS check that the model has saved with plotms!

(3) Force model to save (ft)
mysteps = [3]

ft(vis=vis,
model='X1de2_VYCMa_325_cont0.init.clean.model', usescratch=True)

plotms(vis=vis
...
plotfile='X1de2_VYCMa_325_cont0.init.clean.model_ft.png')



(4) Create a solution table (gaincal) mysteps = [4]

Calculates the gain correction factors for each antenna/spw by comparing (contrasting, calculating the ratio of) the visibilities in the DATA column .vs. the MODEL column for the solution interval requested.

Command:gaincal(vis=vis,caltable='X1de2_VYCMa_325_cont.ph1.solint_inf.tb',solint='inf',refant=refantenna,spw=contchans,calmode='p',gaintype='G',minsnr=3)

Gaincal -- basic set of parameters:

vis, **caltable**, **gaintype(G,T)**, spw, **solint**, **calmode(a,p,ap)**, **minsnr**,spwmap,combine,...

Output: caltable.

What is a solution interval?

Time range over which to calculate the gain.

‘int’ = integration time

‘inf’ = scan time

Can range from ‘int’ - ‘inf’

What you want in a solution interval

A balance between:

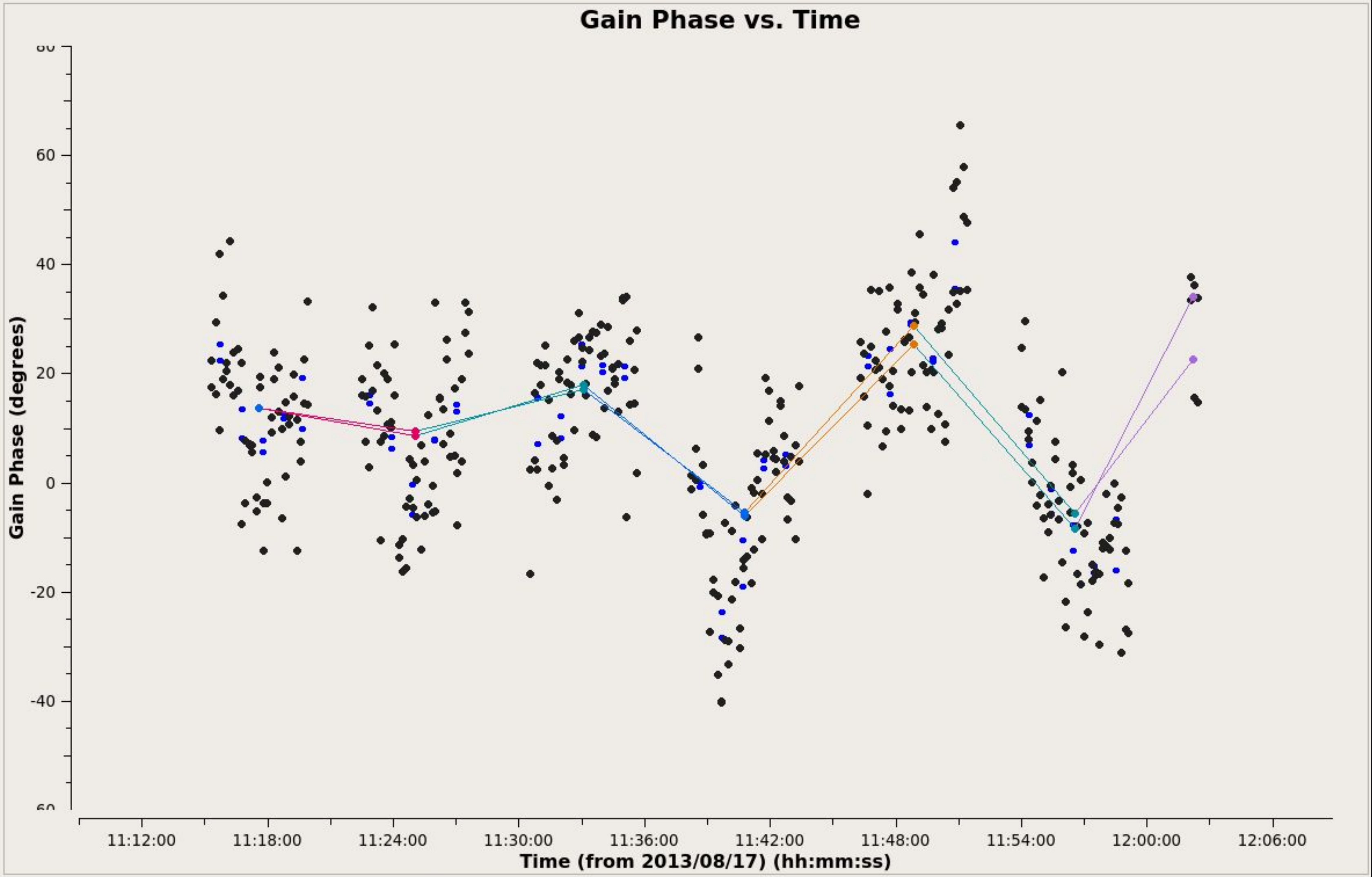
(1) sampling the variations in the solutions

(2) flagging the least amount of solutions

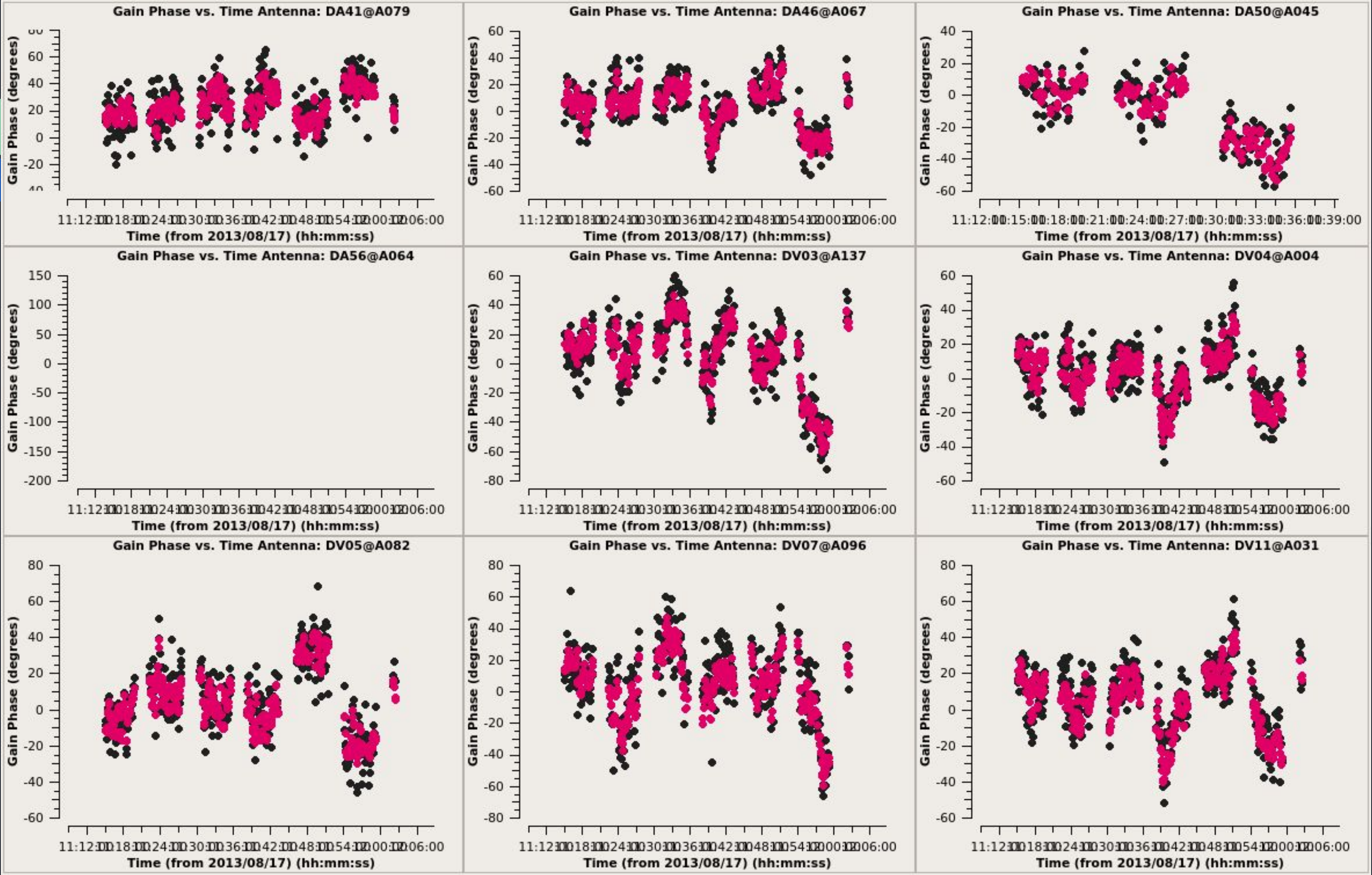
See Advanced slides for details one way to determine this practically.

For this tutorial, we decide to do ‘inf’ for the first round then a shorter interval for the second round.

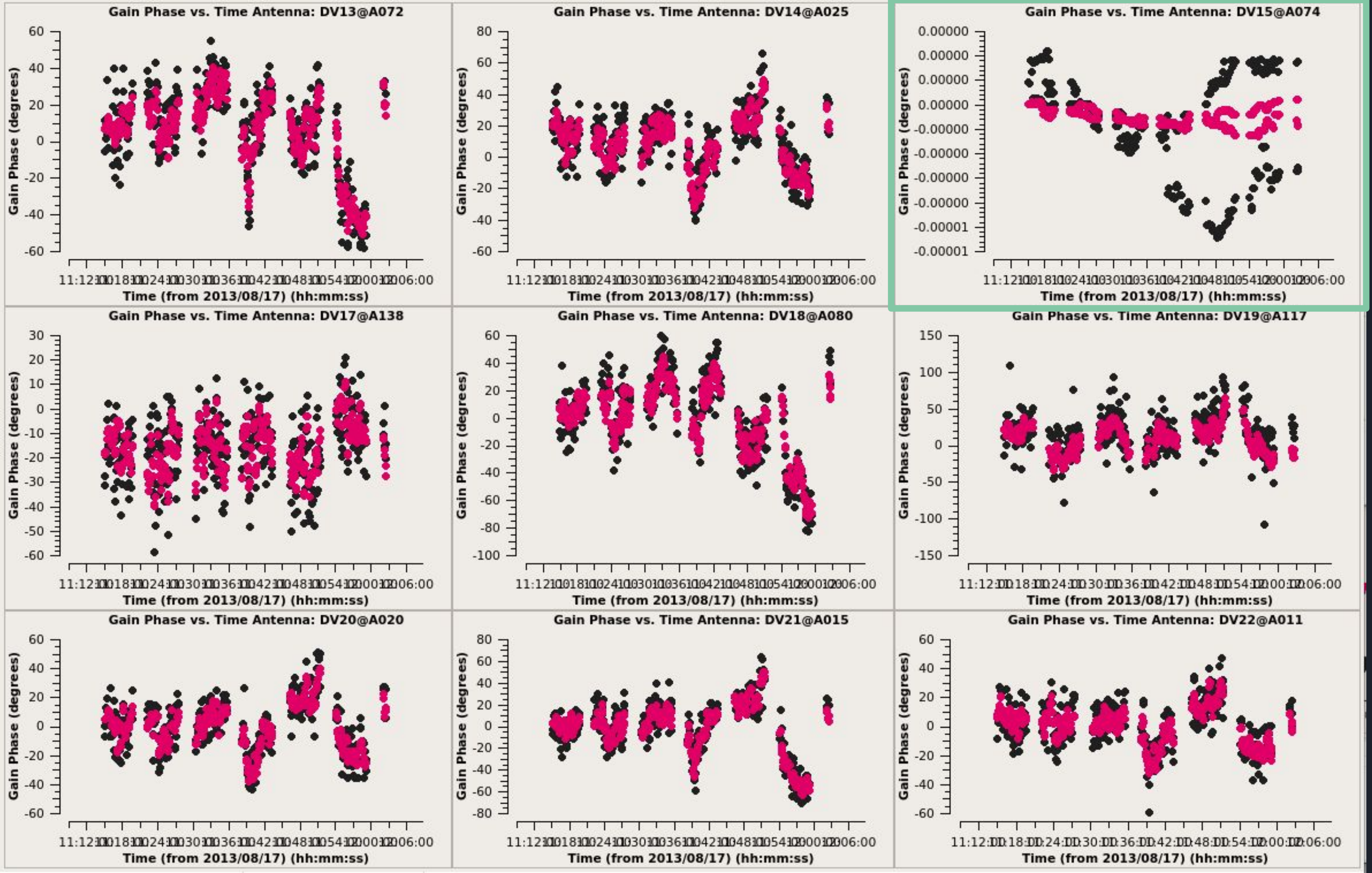
solint=10,60,inf, antenna=`10'



solint=int~10s



solint=int~10s

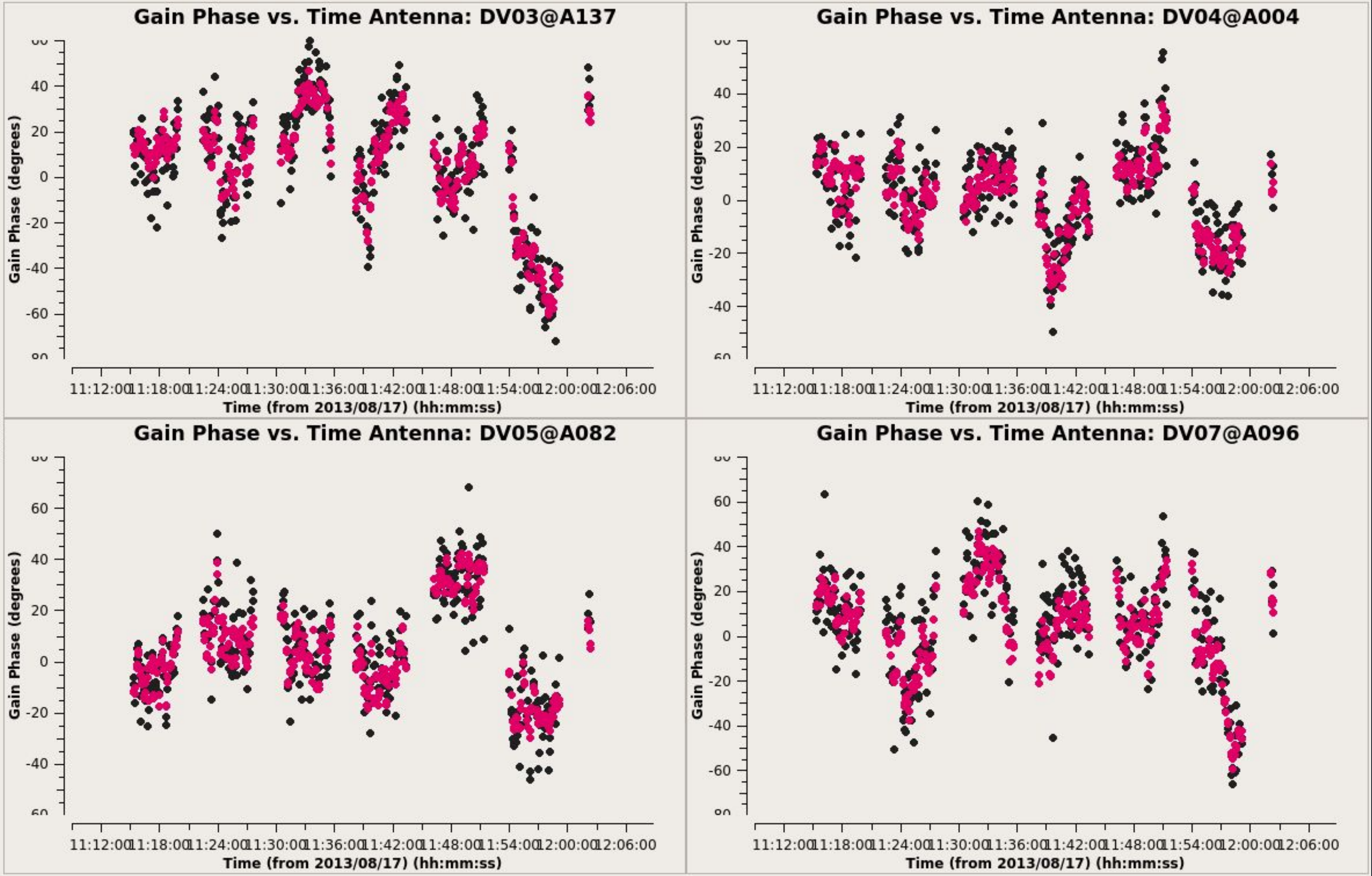


(5) Investigate Solution Intervals
mysteps = [5]

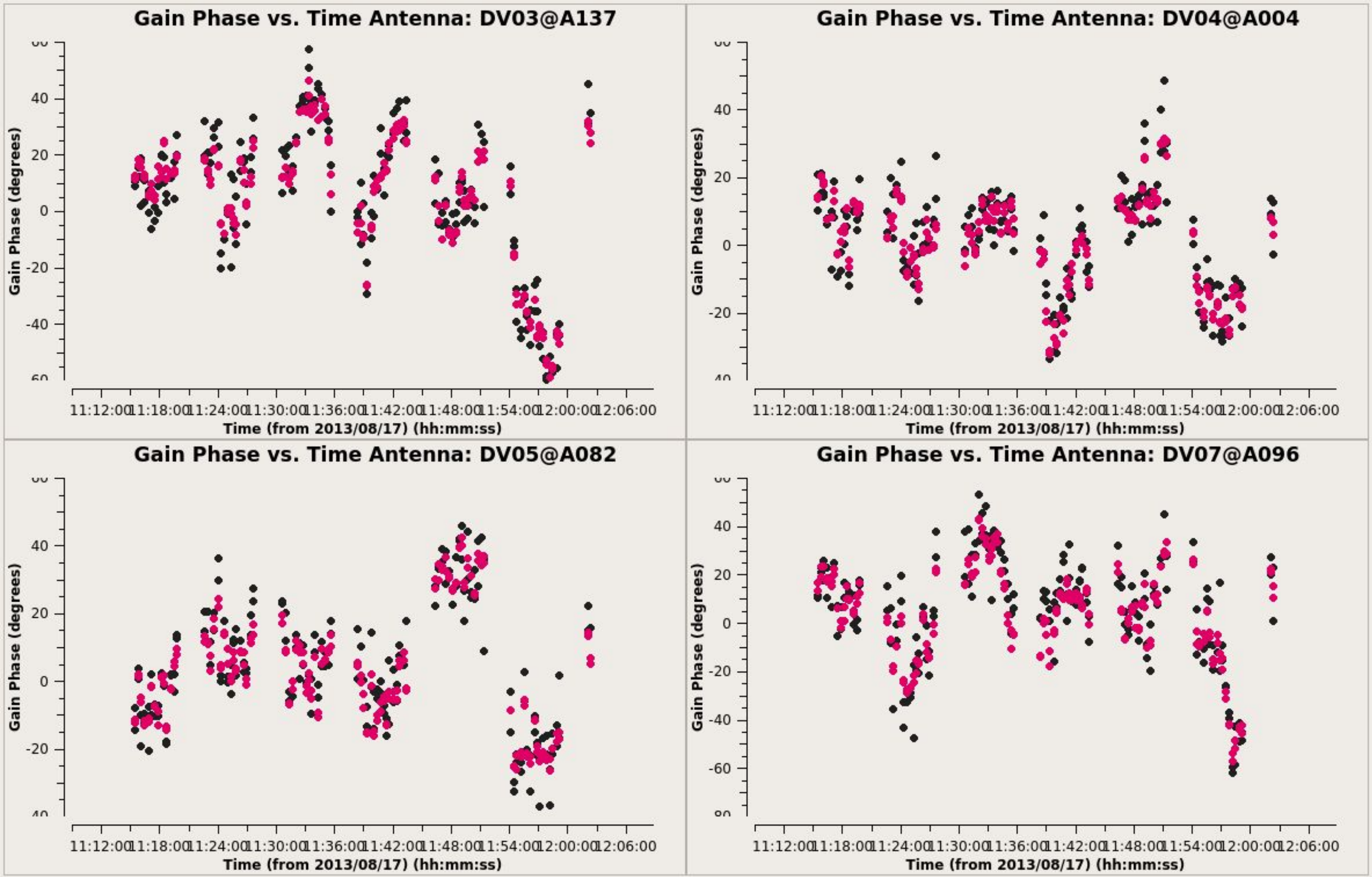
Gaincal + plotms in a loop -> /ph1_checks/

Investigate multiples of int (~10s) (1,2,4,8,16) * int

solint=int~10s

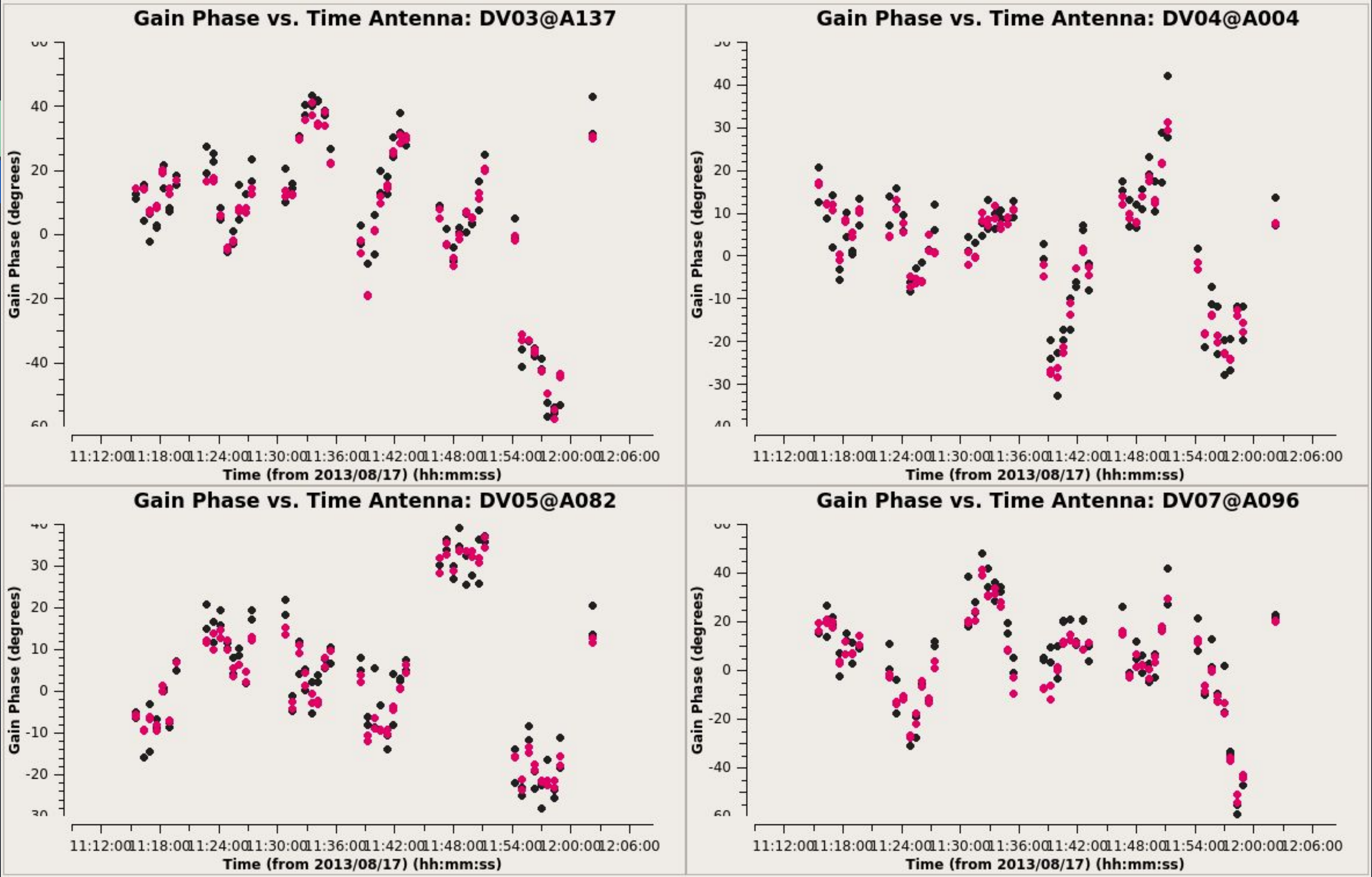


solint=20s

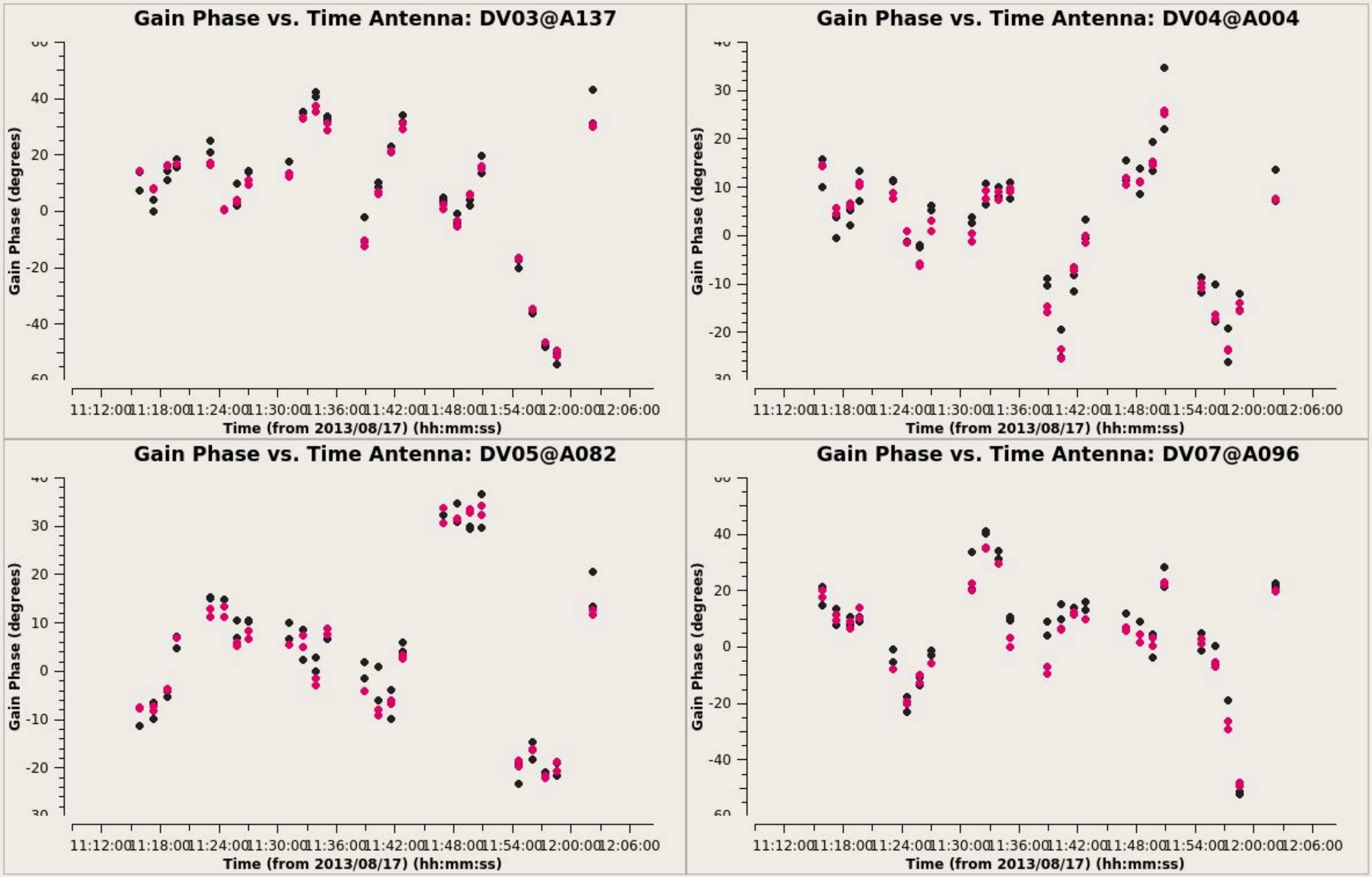


Gain Phase vs. Time Antenna: DV03@A137
Gain Phase vs. Time Antenna: DV04@A004
Gain Phase vs. Time Antenna: DV05@A082
Gain Phase vs. Time Antenna: DV07@A096
Gain Phase (degrees)
Time (from 2013/08/17) (hh:mm:ss)
11:12:00
11:18:00
11:24:00
11:30:00
11:36:00
11:42:00
11:48:00
11:54:00
12:00:00
12:06:00
40
20
0
-20
-40
-60

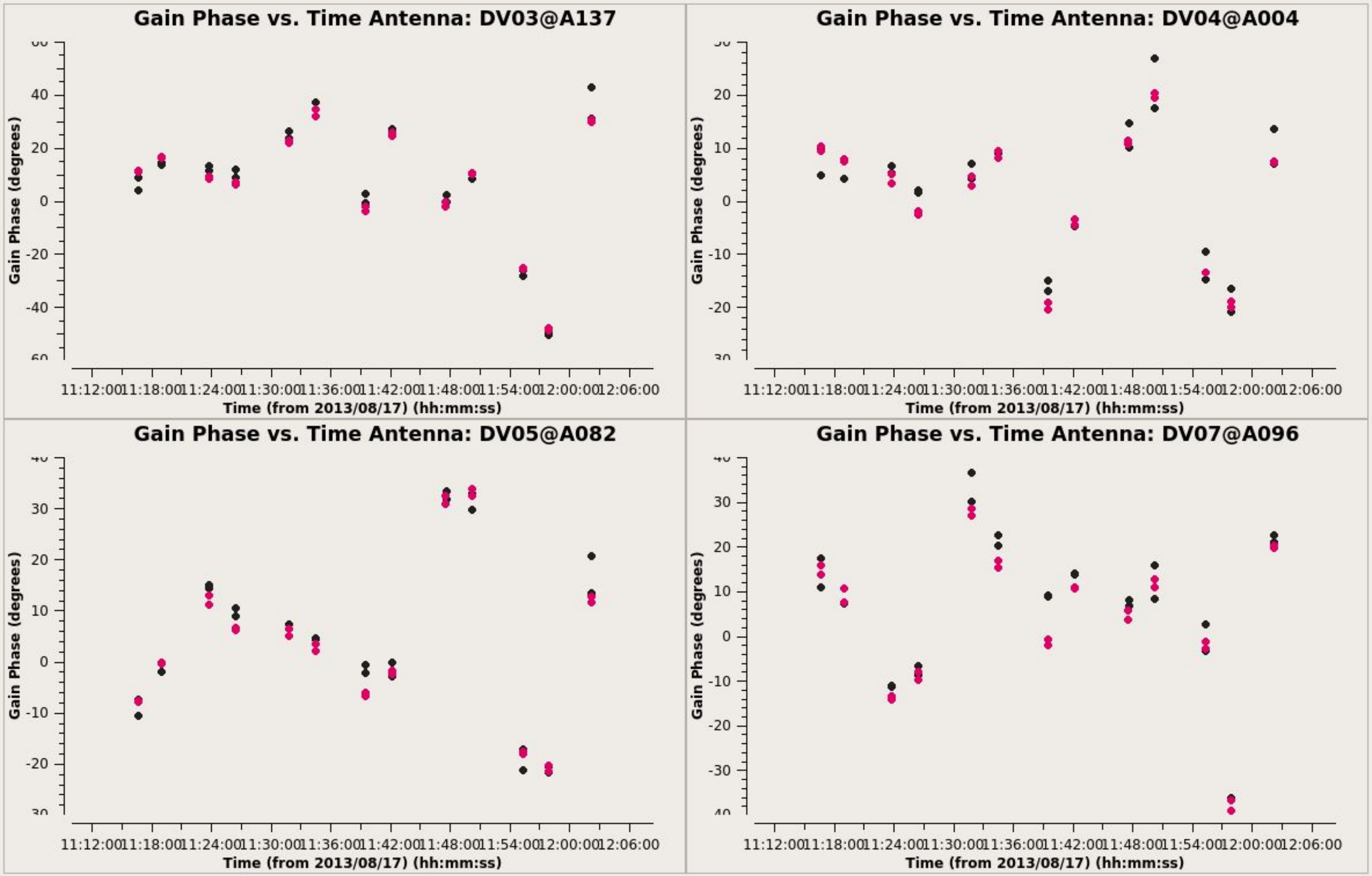
solint=40s


Gain Phase vs. Time Antenna: DV03@A137
Gain Phase vs. Time Antenna: DV04@A004
Gain Phase vs. Time Antenna: DV05@A082
Gain Phase vs. Time Antenna: DV07@A096
Gain Phase (degrees)
Time (from 2013/08/17) (hh:mm:ss)
11:12:00 11:18:00 11:24:00 11:30:00 11:36:00 11:42:00 11:48:00 11:54:00 12:00:00 12:06:00

solint=80s

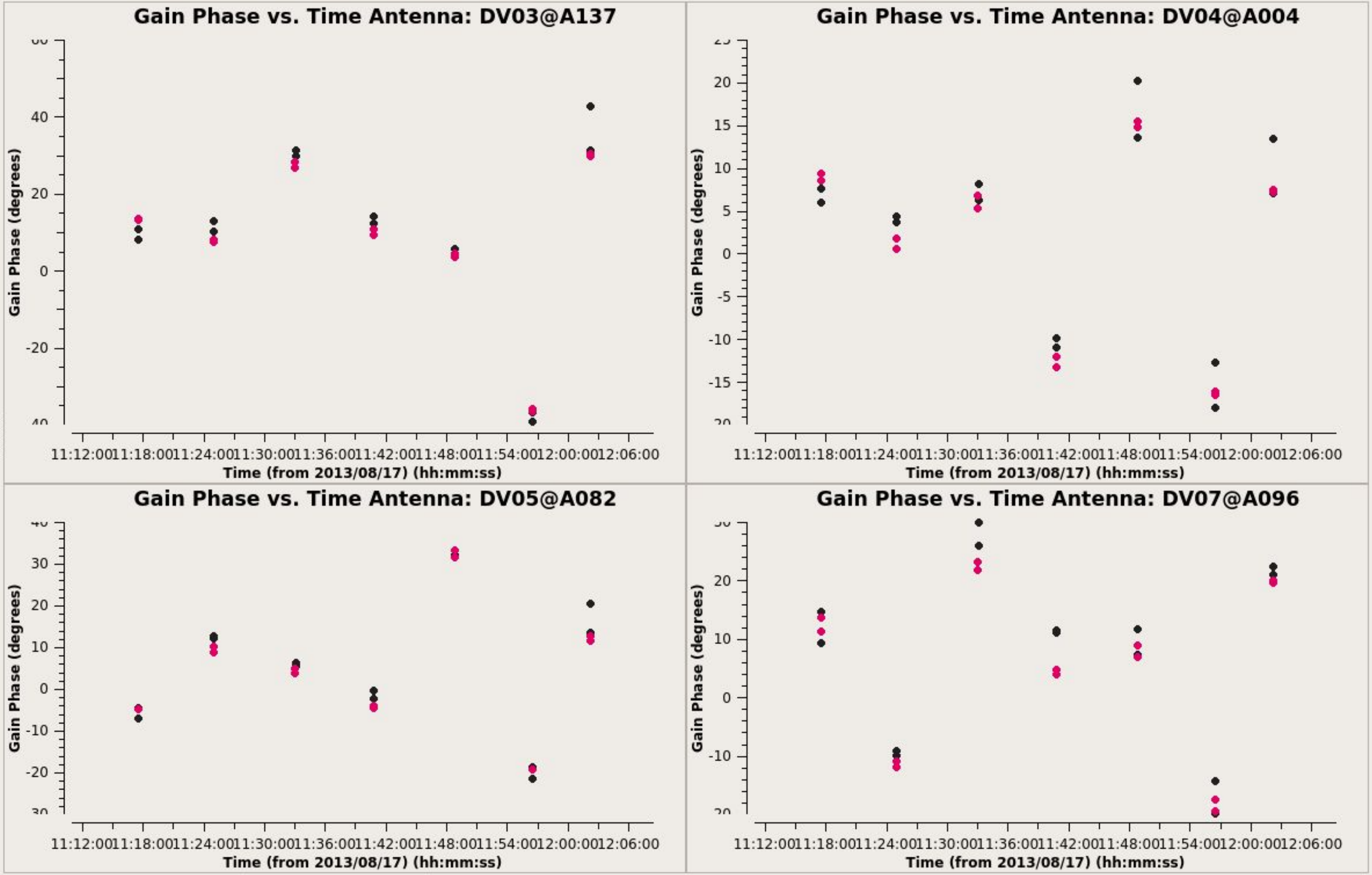

Gain Phase vs. Time Antenna: DV03@A137
Gain Phase vs. Time Antenna: DV04@A004
Gain Phase vs. Time Antenna: DV05@A082
Gain Phase vs. Time Antenna: DV07@A096
Gain Phase (degrees)
Time (from 2013/08/17) (hh:mm:ss)
11:12:00
11:18:00
11:24:00
11:30:00
11:36:00
11:42:00
11:48:00
11:54:00
12:00:00
12:06:00

solint=160s



Gain Phase vs. Time Antenna: DV03@A137
Gain Phase vs. Time Antenna: DV04@A004
Gain Phase vs. Time Antenna: DV05@A082
Gain Phase vs. Time Antenna: DV07@A096
Gain Phase (degrees)
Time (from 2013/08/17) (hh:mm:ss)
11:12:00 11:18:00 11:24:00 11:30:00 11:36:00 11:42:00 11:48:00 11:54:00 12:00:00 12:06:00

solint=inf~285s



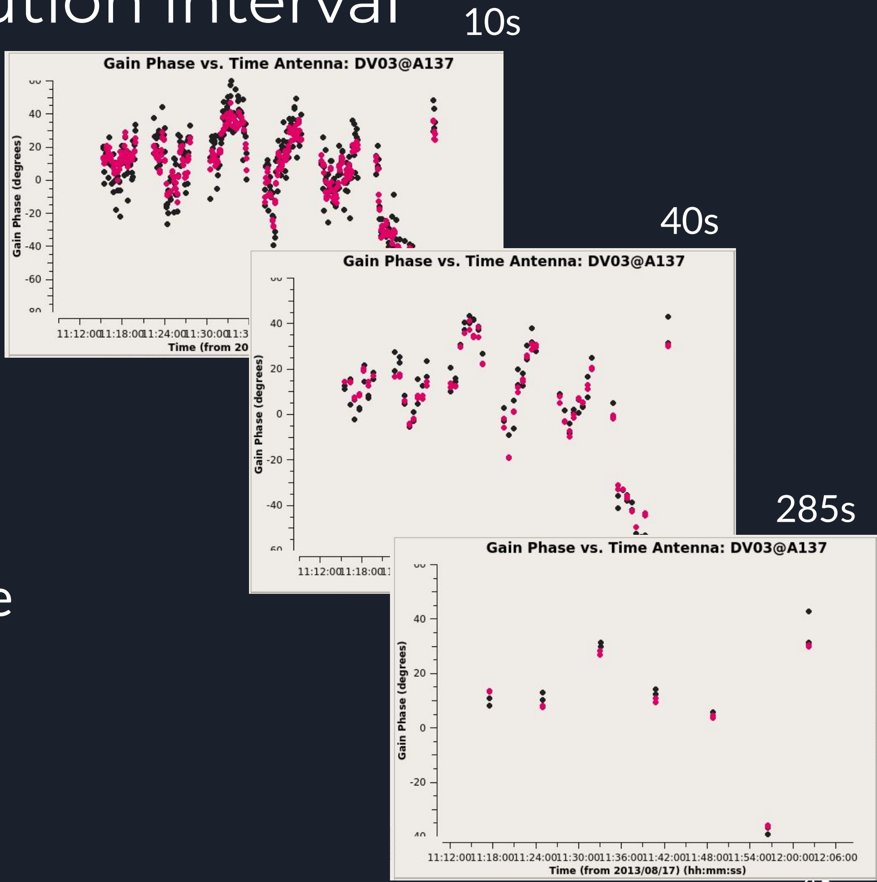
What you want in a solution interval

A balance between:

(1) sampling the variations in the solutions

(2) flagging the least amount of solutions

For this tutorial, we decide to do ‘inf’ for the first round then a shorter interval for the second round.



Flagging messages from Gaincal

- You can keep an eye on how many solutions are failed/flagged by checking in the terminal
- Notice how the number of messages decreased as we increased the solution interval
- The failed solutions are flagged in the gaintables produced from gaincal
- Don't need to worry until > 10% flagged

```
Step  5 Explore different solution intervals
Solint: int
2021-05-11 09:41:00     SEVERE  MeasTable::dUTC(Double) (file casacore/measures/Measures/MeasTable.cc, line 4290) L
le TAI_UTC seems out-of-date.
2021-05-11 09:41:00     SEVERE  MeasTable::dUTC(Double) (file casacore/measures/Measures/MeasTable.cc, line 4290)+
he table is updated (see the CASA documentation or your system admin),
2021-05-11 09:41:00     SEVERE  MeasTable::dUTC(Double) (file casacore/measures/Measures/MeasTable.cc, line 4290)+
nd coordinates derived from UTC could be wrong by 1s or more.
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:15:21.1
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:15:33.2
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:15:42.2
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:15:52.8
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:04.9
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:14.0
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:24.6
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:36.7
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:45.8
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:56.4
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:08.5
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:17.6
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:28.2
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:40.3
5 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:49.3
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:18:00.0
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:18:31.7
3 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:18:52.9
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:03.5
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:15.6
3 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:24.7
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:35.3
3 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:47.4
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:56.4
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:22:30.6
```

Skip Step 6 [ADVANCED]

(7) Apply calibration table (applycal)
mysteps = [7]

Applies the caltable output from gaincal to the SPWs selected.

Command: applycal(vis = obj, field='Vy Cma', spw='0,1', spwmap=[0,1], gaintable='X1de2_VYCMa_325_cont.ph1.solint_inf.tb', calwt = False, applymode='calonly', flagbackup = False)

Applycal -- basic set of parameters:

  vis, spw, spwmap, **gaintable**, calwt, applymode, ...

Output: None but adds data to CORRECTED DATA column.

Flagging Solutions

Gaincal tables have flagging information when apply tables you can control flagging behavior

`applycal applymode`

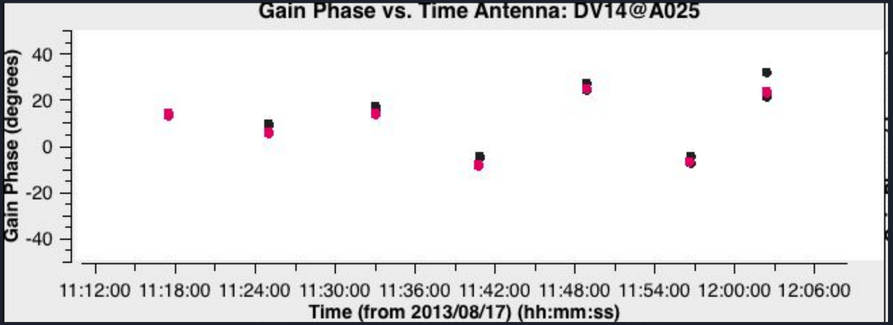
- applymode = ‘calflag’: calibrate data and apply flags from solutions

```
3 of 27 solutions flagged due to SNR < 6 in spw=0 at 2017/10/13/20:59:21.0
```

- applymode=’calonly’ : calibrate data only, flags from solutions NOT applied
- Flagging messages (threshold for worry > 10% of solutions) (set by minsnr and combining)(see advanced).

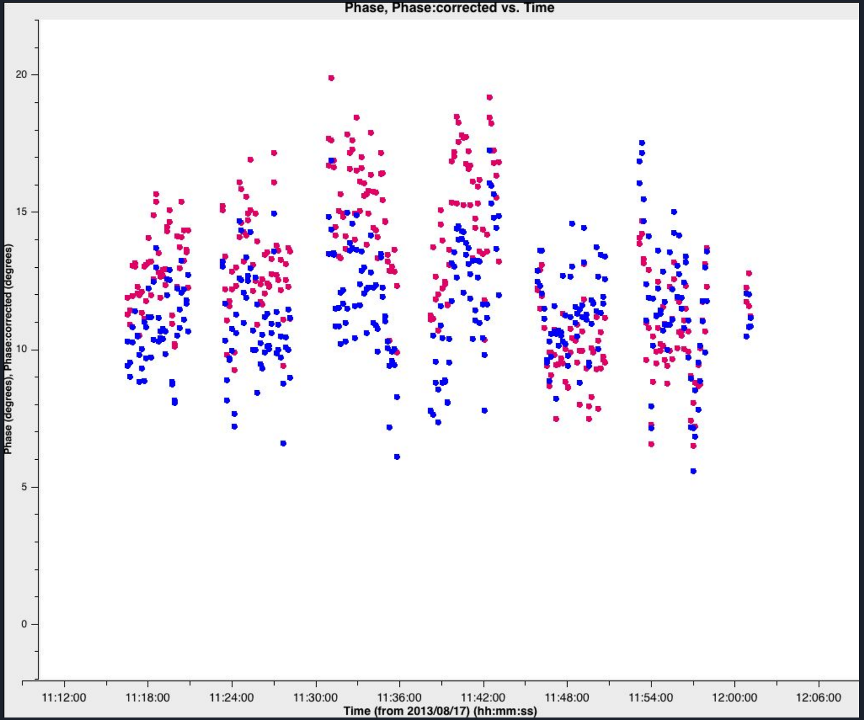
Demonstration of Gaintable Application

gaintable ph1 solint 'inf'



antenna='10' is DV14
(see listobs)

visibilities



red=datacolumn
blue=corrected datacolumn after ph1 table applied
spw='1', antenna='10'

What we have done thus far

Image source to deconvolve create model
`tclean`

↓

Place model in MS
`ft`

MS now has MODEL column

↓

Calibrate to match data to model
`gaincal`

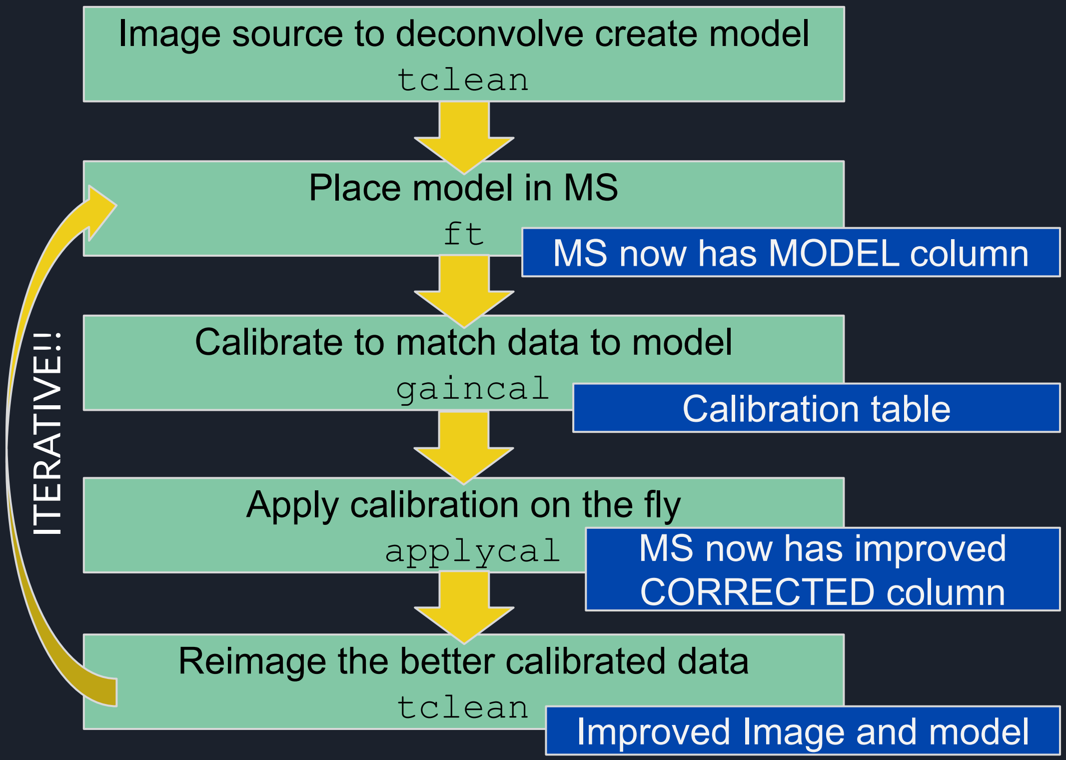
Calibration table

↓

Apply calibration on the fly
`applycal`

MS now has improved CORRECTED column

What we have done thus far

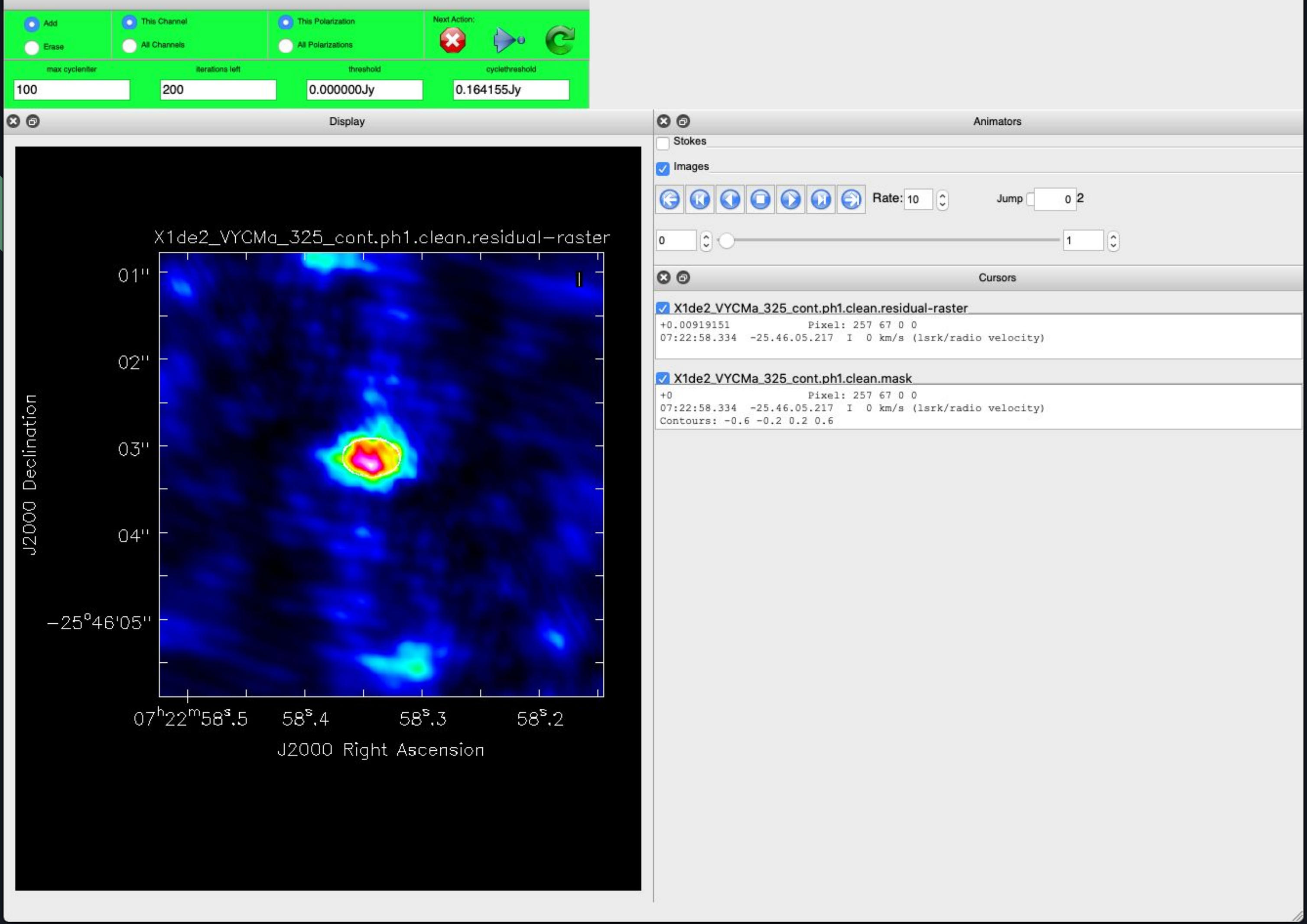


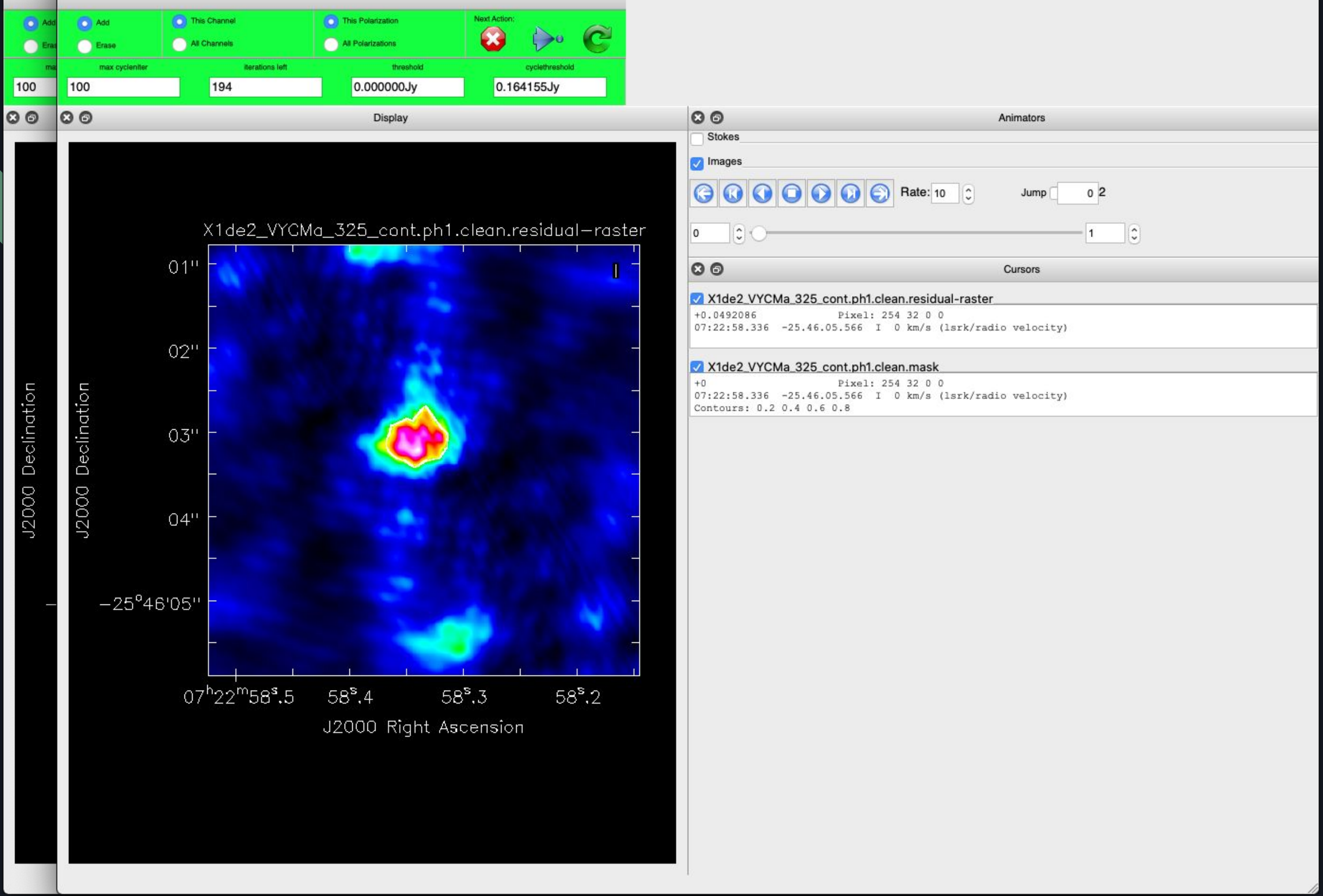
(8) Begin Second Round of Self-Cal to Compare mysteps = [8]

Use `tclean` to update model and use `ft` to save the model

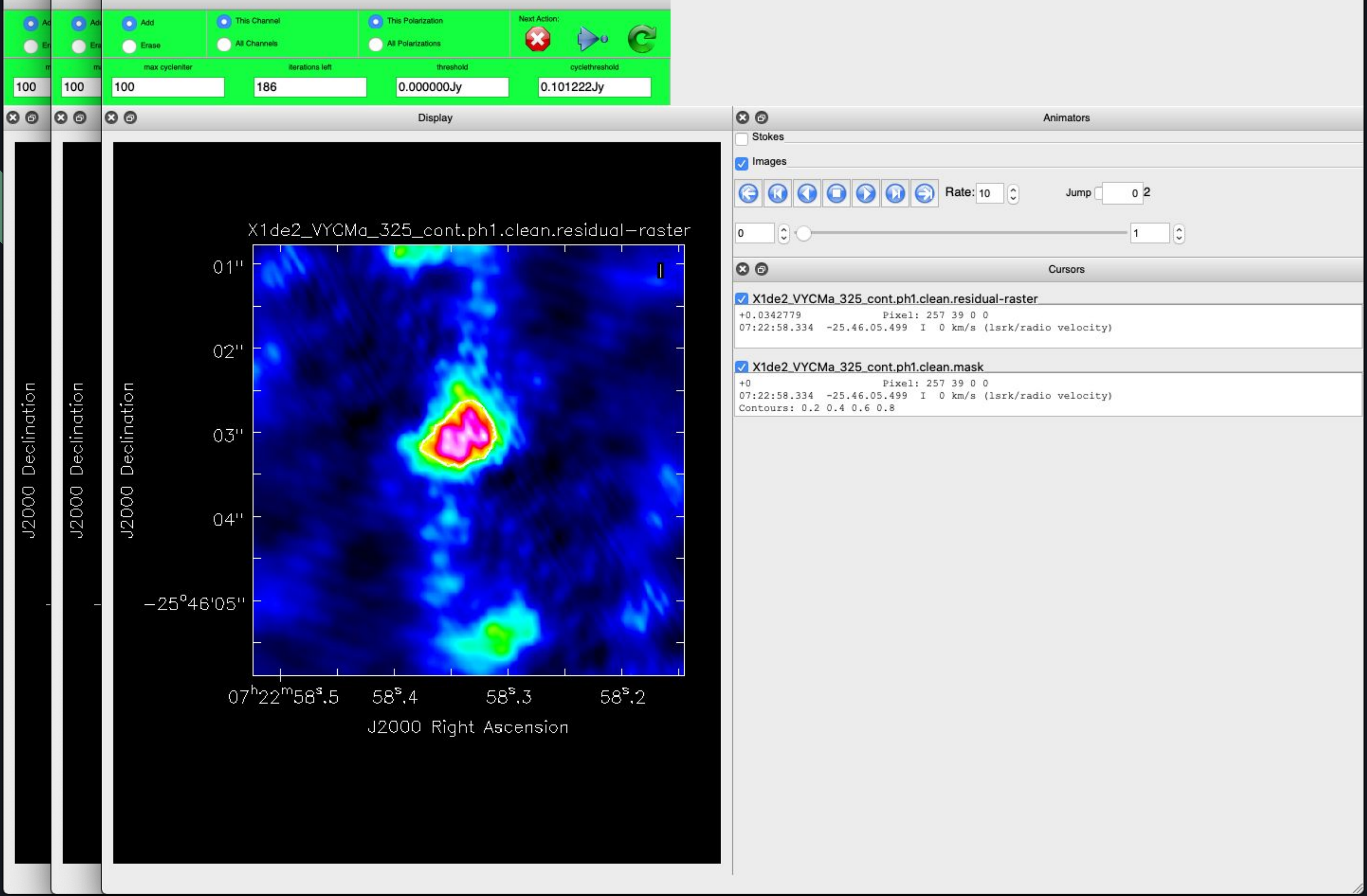
```
tclean(vis=vis,
    imagename='X1de2_VYCMa_325_cont.ph1.clean',
    field=field,
    spw=contchans,
    specmode='mfs',
    cell='0.01arcsec',
    imsize=512,
    niter=200,
    interactive=True)
```

** *Could use mask from last clean*

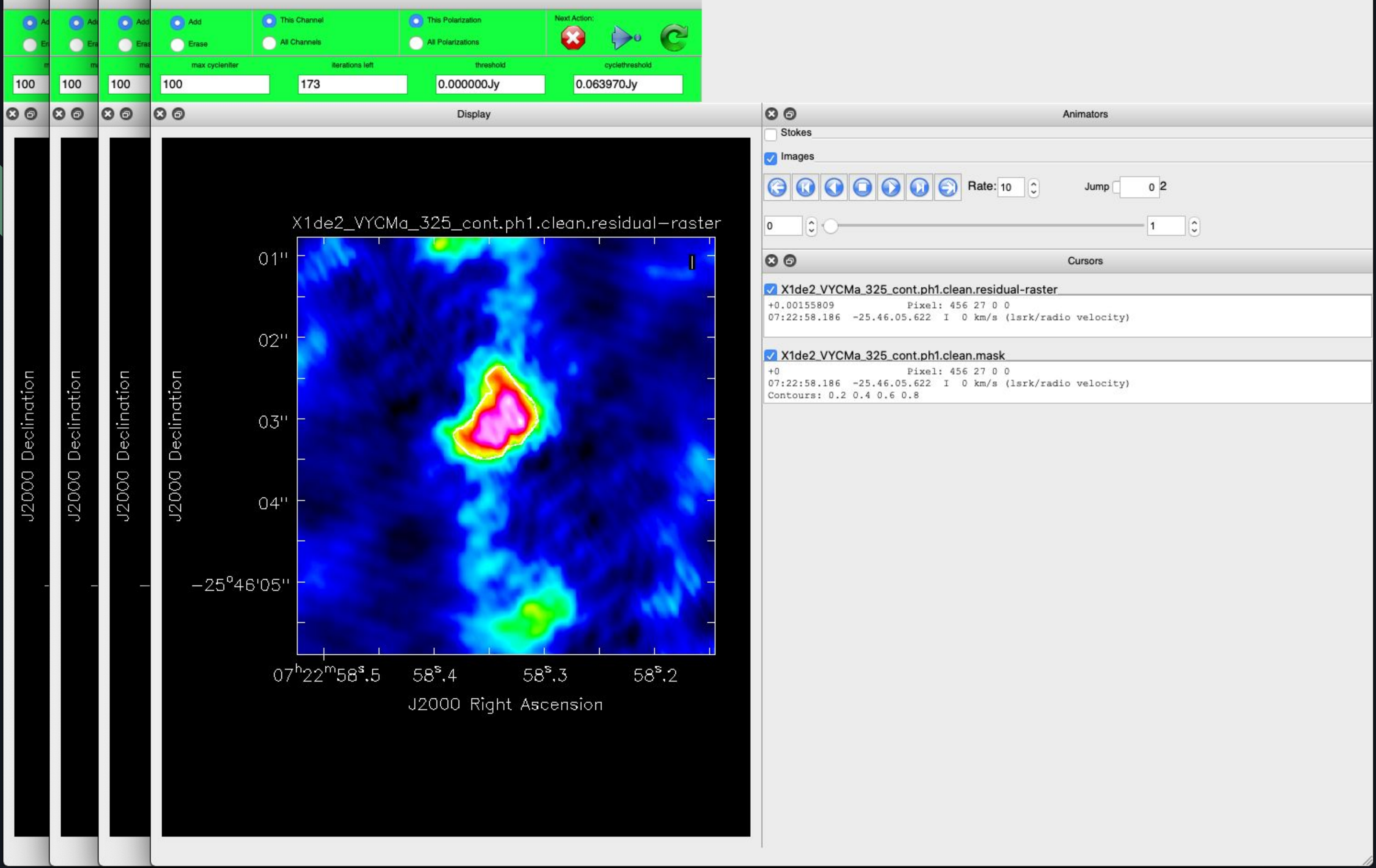

Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
200
threshold
0.000000Jy
cyclethreshold
0.164155Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual–raster
J2000 Declination
J2000 Right Ascension
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.00919151 Pixel: 257 67 0 0
07:22:58.334 -25.46.05.217 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 257 67 0 0
07:22:58.334 -25.46.05.217 I 0 km/s (lsrk/radio velocity)
Contours: -0.6 -0.2 0.2 0.6



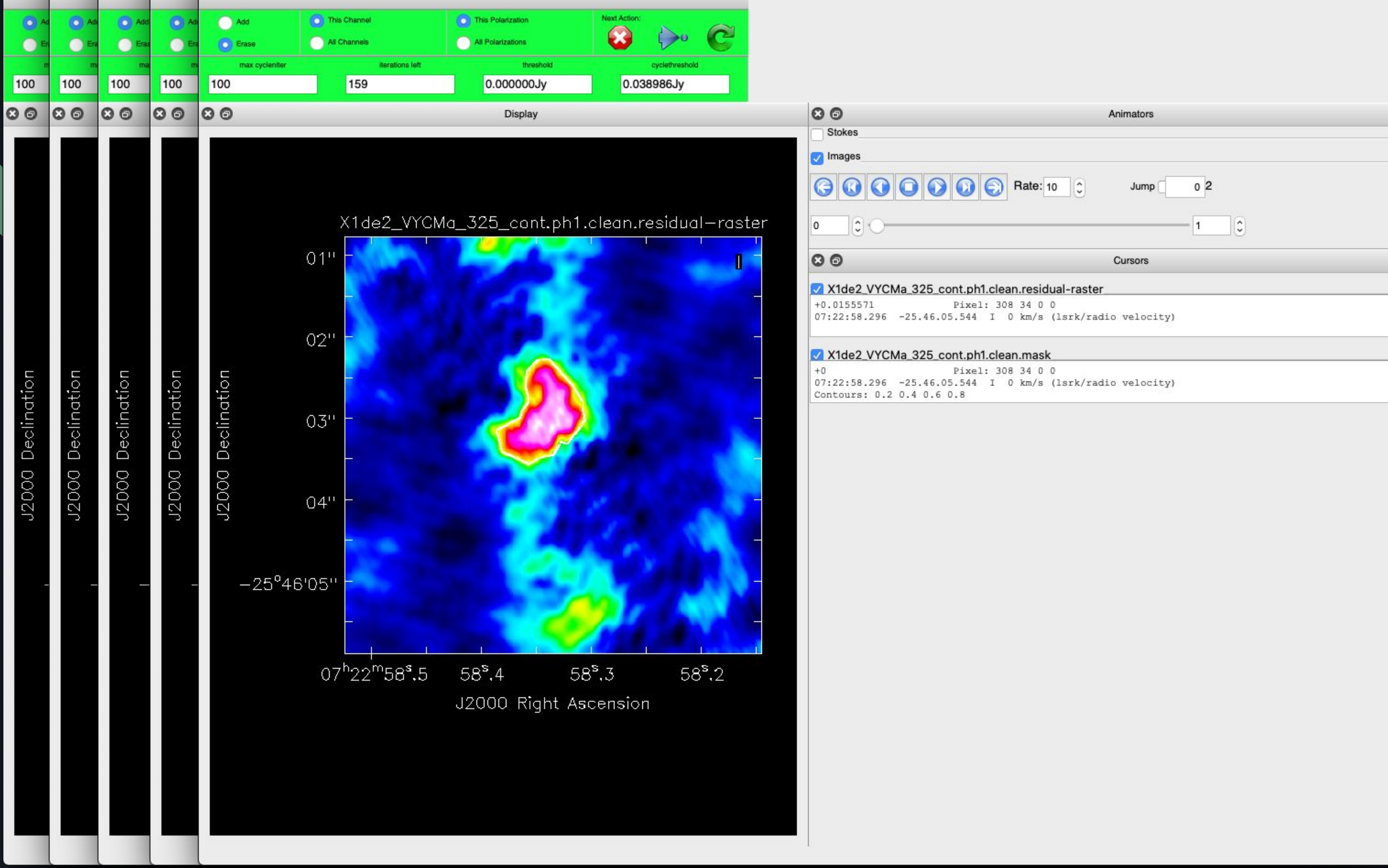
Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
iterations left
threshold
cyclethreshold
100
100
194
0.000000Jy
0.164155Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual−raster
01''
02''
03''
04''
−25°46'05''
J2000 Declination
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.0492086 Pixel: 254 32 0 0
07:22:58.336 -25.46.05.566 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 254 32 0 0
07:22:58.336 -25.46.05.566 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8



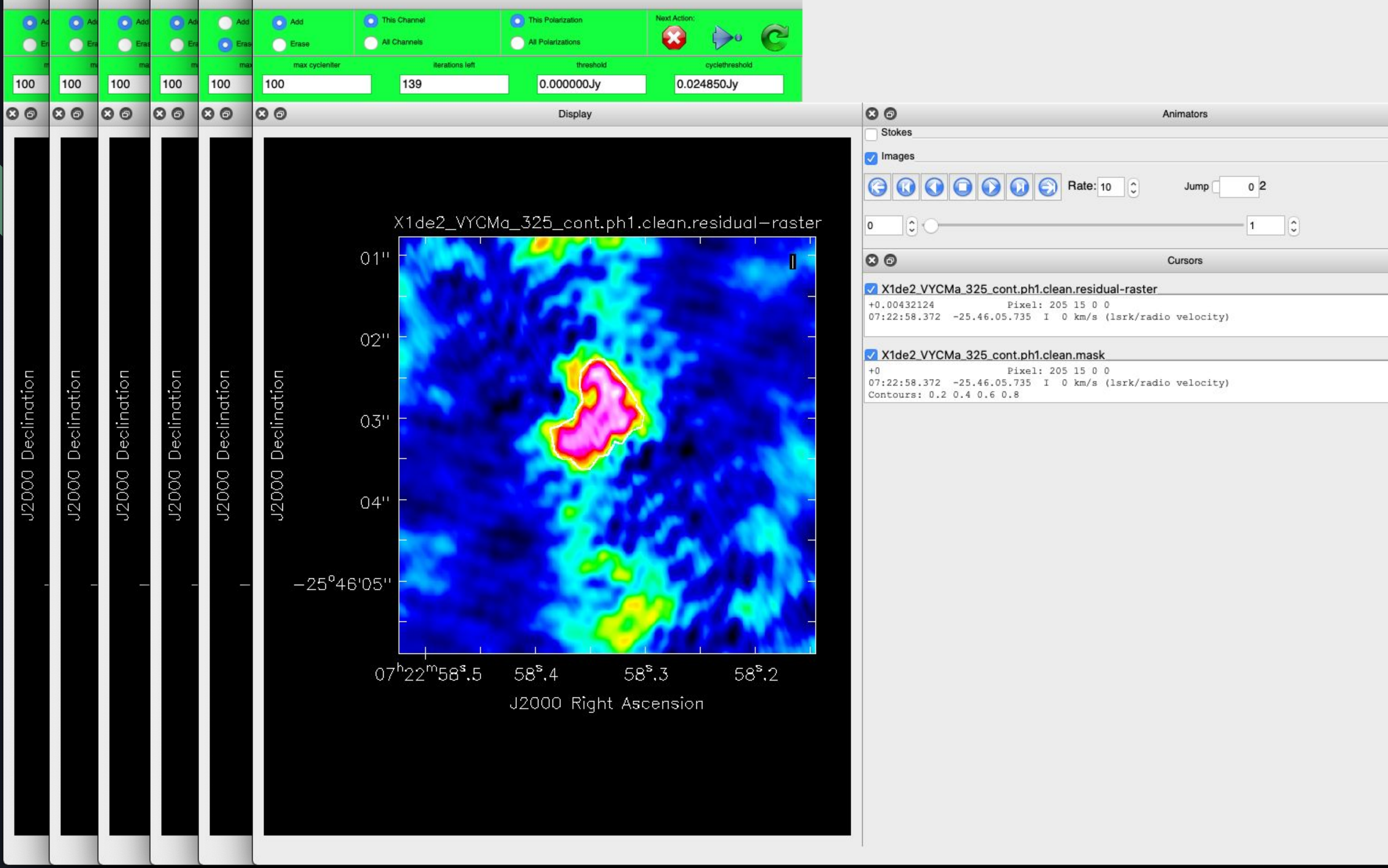
Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
186
threshold
0.000000Jy
cyclethreshold
0.101222Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual−raster
01''
02''
03''
04''
−25°46'05''
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
J2000 Declination
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.0342779 Pixel: 257 39 0 0
07:22:58.334 -25.46.05.499 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 257 39 0 0
07:22:58.334 -25.46.05.499 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8



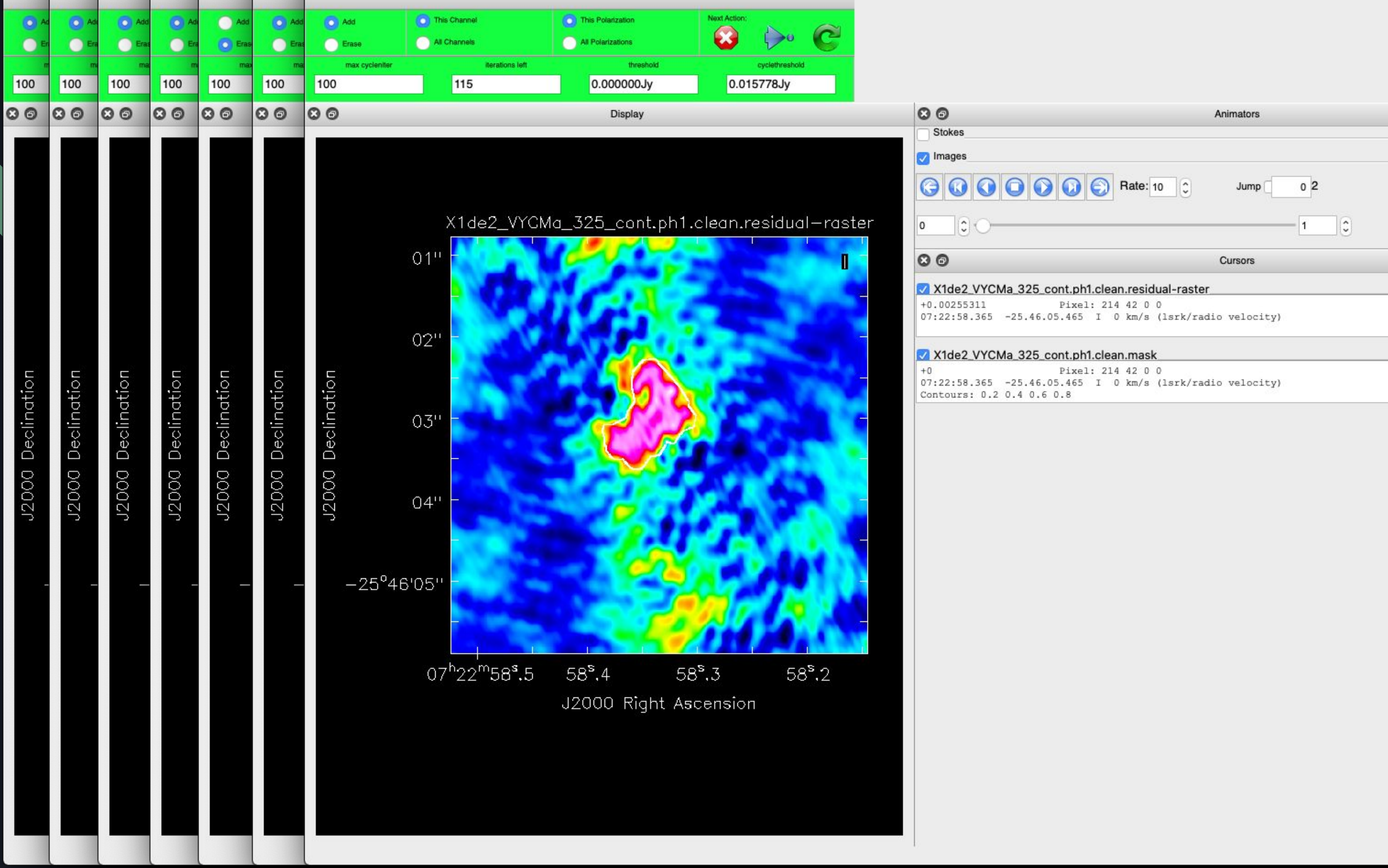
Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
173
threshold
0.000000Jy
cyclethreshold
0.063970Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual−raster
01''
02''
03''
04''
−25°46'05''
J2000 Declination
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.00155809 Pixel: 456 27 0 0
07:22:58.186 -25.46.05.622 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 456 27 0 0
07:22:58.186 -25.46.05.622 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8

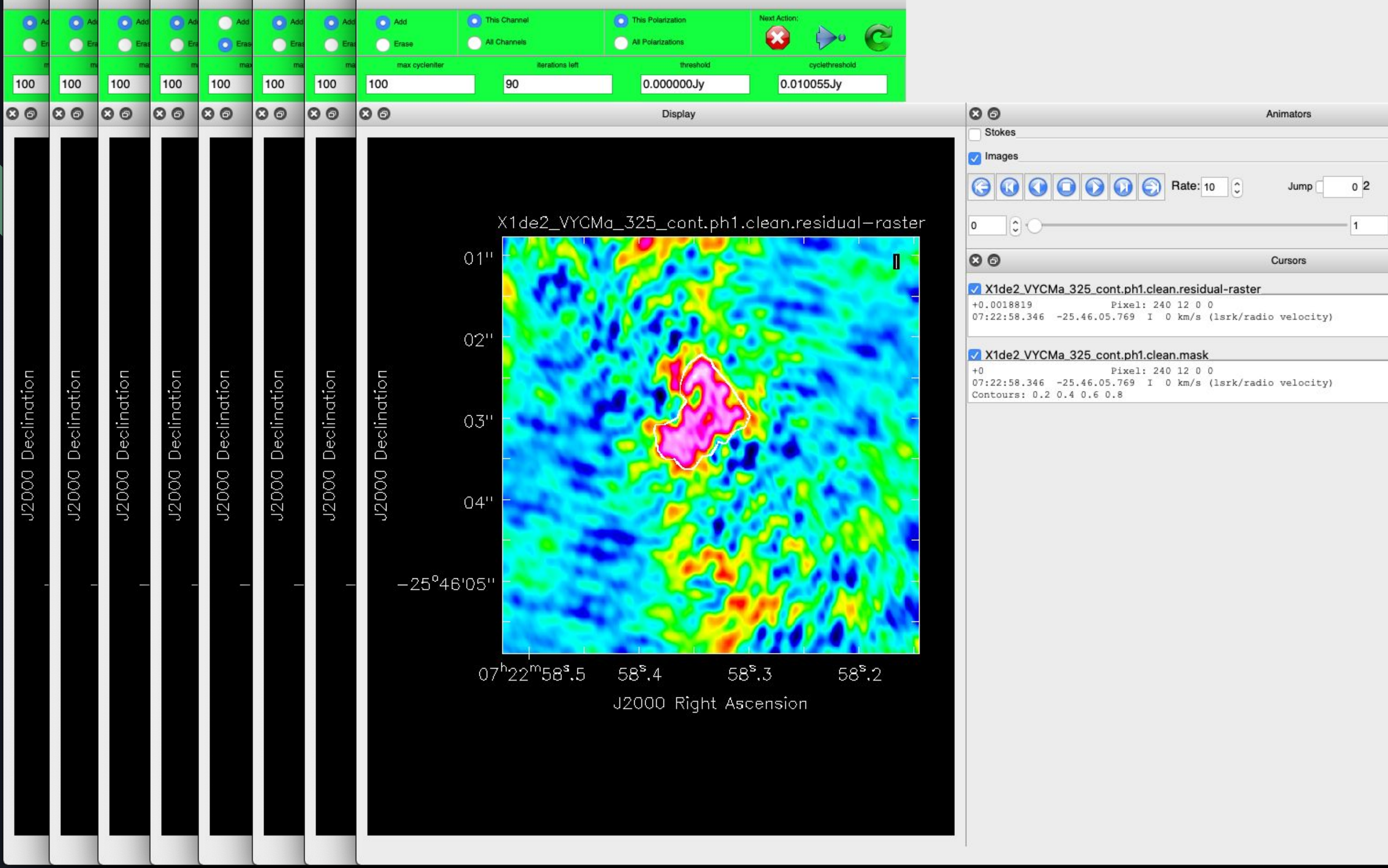


Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
iterations left
threshold
cyclethreshold
100
159
0.000000Jy
0.038986Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual–raster
01''
02''
03''
04''
–25°46'05''
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
J2000 Declination
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.0155571 Pixel: 308 34 0 0
07:22:58.296 -25.46.05.544 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 308 34 0 0
07:22:58.296 -25.46.05.544 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8

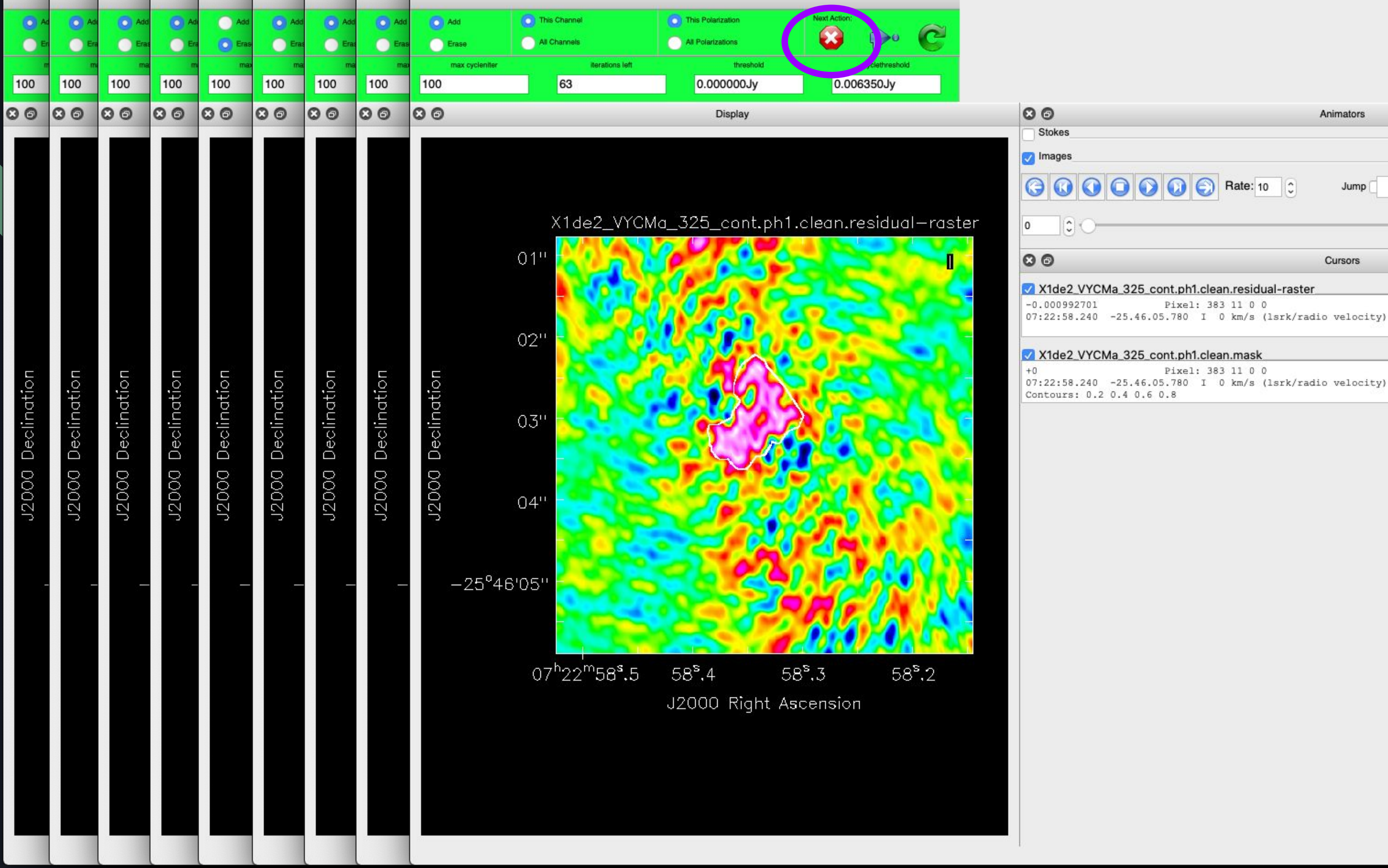


Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
139
threshold
0.000000Jy
cyclethreshold
0.024850Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual−raster
J2000 Declination
01''
02''
03''
04''
−25°46'05''
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.00432124 Pixel: 205 15 0 0
07:22:58.372 -25.46.05.735 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 205 15 0 0
07:22:58.372 -25.46.05.735 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8


Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
115
threshold
0.000000Jy
cyclethreshold
0.015778Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
01''
02''
03''
04''
−25°46'05''
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
J2000 Declination
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.00255311 Pixel: 214 42 0 0
07:22:58.365 -25.46.05.465 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 214 42 0 0
07:22:58.365 -25.46.05.465 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8

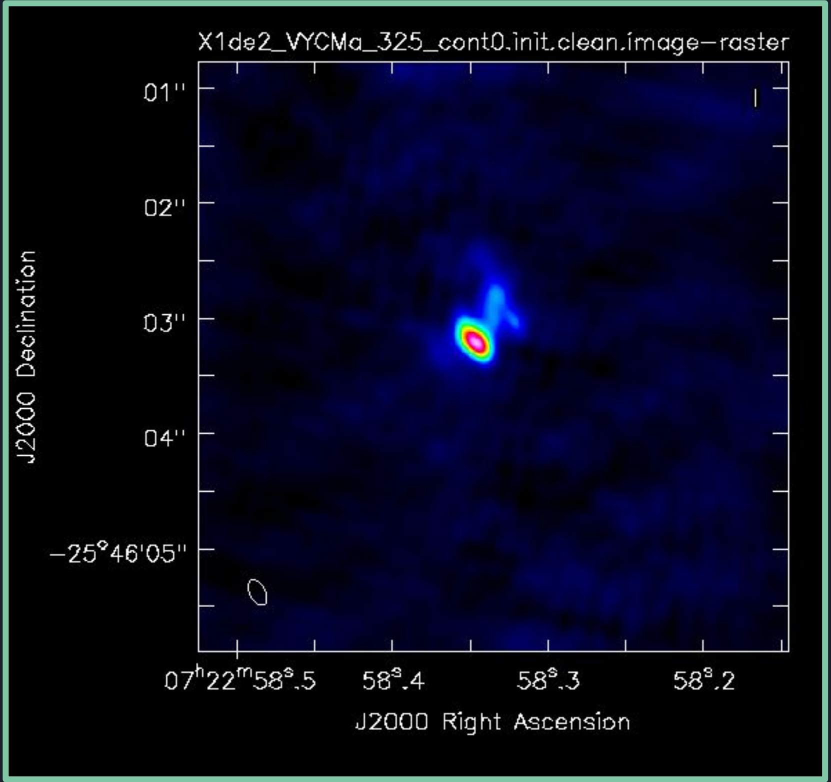


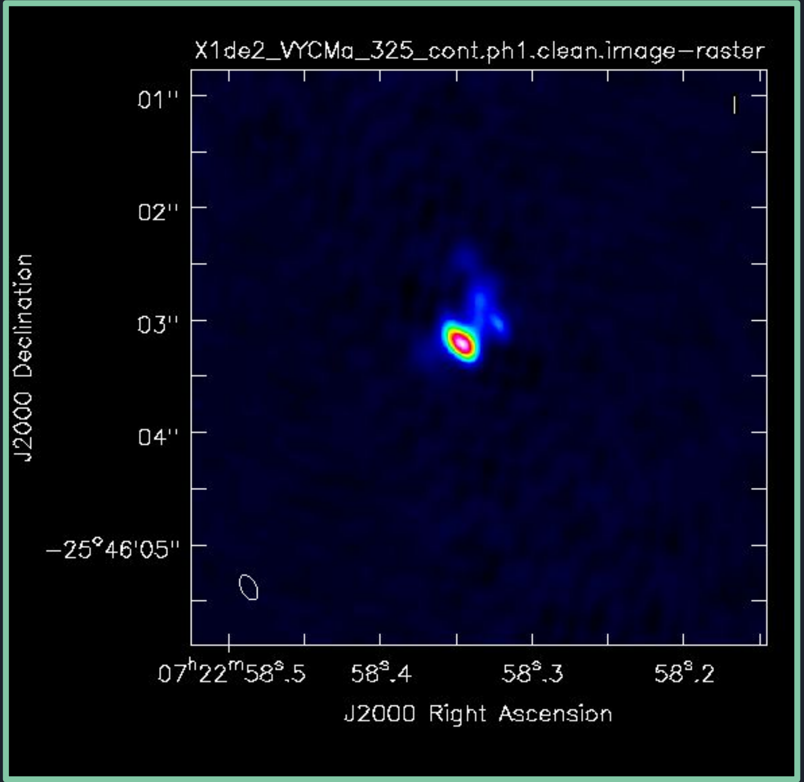
Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
100
iterations left
90
threshold
0.000000Jy
cyclethreshold
0.010055Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual–raster
01"
02"
03"
04"
–25°46'05"
J2000 Declination
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
Animators
Stokes
Images
Rate: 10
Jump
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
+0.0018819 Pixel: 240 12 0 0
07:22:58.346 -25.46.05.769 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 240 12 0 0
07:22:58.346 -25.46.05.769 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8



Add
Erase
This Channel
All Channels
This Polarization
All Polarizations
Next Action:
max cycleniter
iterations left
threshold
cyclethreshold
100
63
0.000000Jy
0.006350Jy
Display
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
01"
02"
03"
04"
−25°46'05"
J2000 Declination
07h22m58s.5
58s.4
58s.3
58s.2
J2000 Right Ascension
Animators
Stokes
Images
Rate: 10
Jump
0
Cursors
X1de2_VYCMa_325_cont.ph1.clean.residual-raster
-0.000992701 Pixel: 383 11 0 0
07:22:58.240 -25.46.05.780 I 0 km/s (lsrk/radio velocity)
X1de2_VYCMa_325_cont.ph1.clean.mask
+0 Pixel: 383 11 0 0
07:22:58.240 -25.46.05.780 I 0 km/s (lsrk/radio velocity)
Contours: 0.2 0.4 0.6 0.8

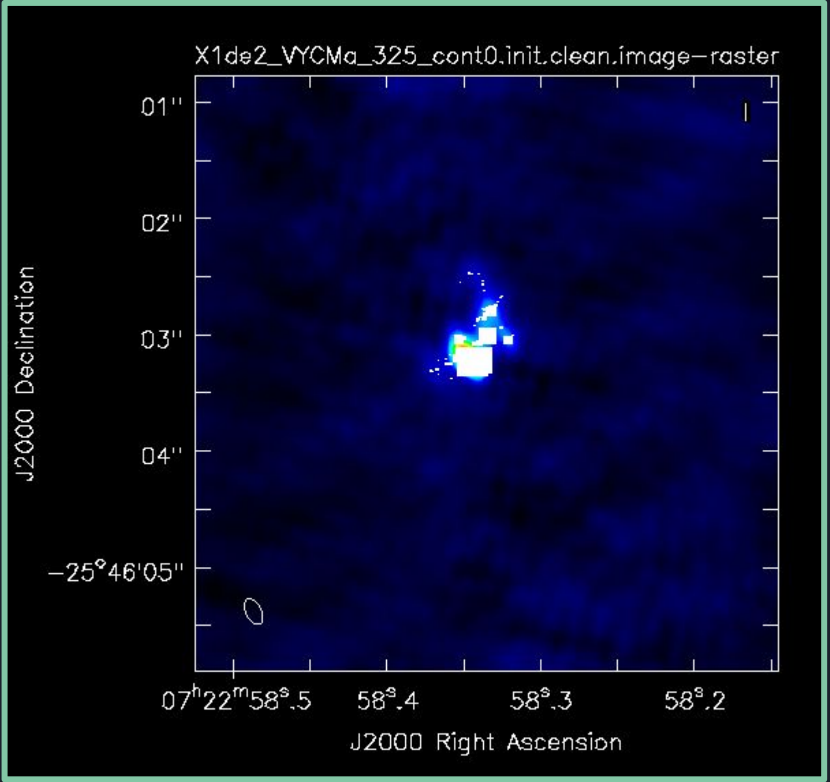
Comparison - Image

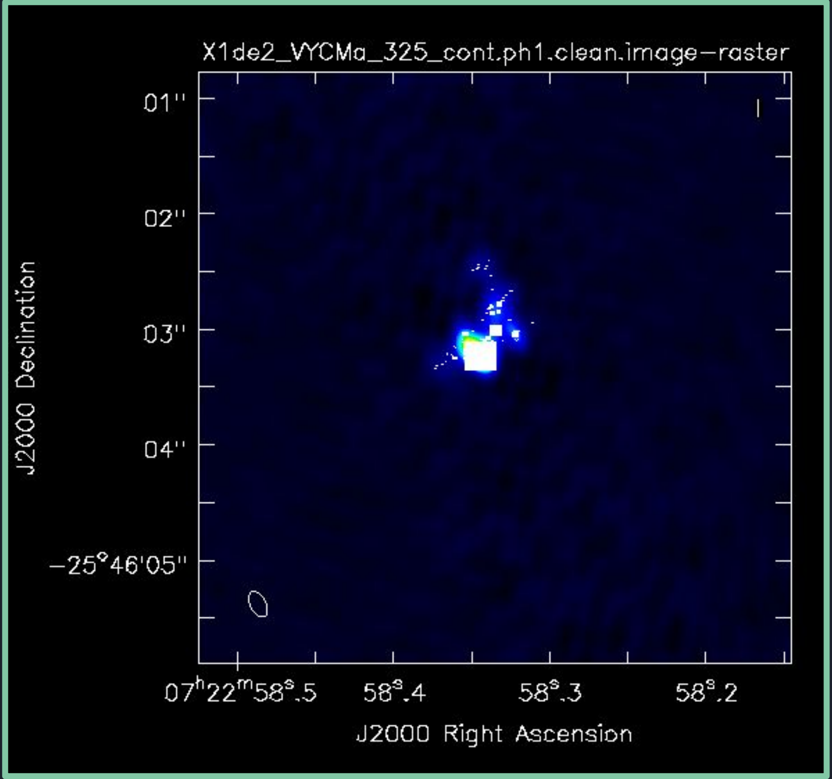




SNR~ 51 -> SNR ~ 126 is a factor of 2.5

Comparison - Model





PAUSE: What solution interval would you apply next?

Hints:

- What interval do the solutions vary on?
- Balance between SNR/flagging solutions and capturing variations.

Commands to use to investigate:

- Gaincal with `solint='int'`
- Gaincal with `solint='60s'`
- Plotms
- **Put your `solint` choice in step 11, apply (step 12) and image (step 13)**
- Look at the output of Step 9 (and execute Step 10 if you so desire, which will be covered in the *Advanced* presentation)

Tutor’s Second Round of Self-cal -- solint=’60s’ Apply gain tables ‘on-the-fly’

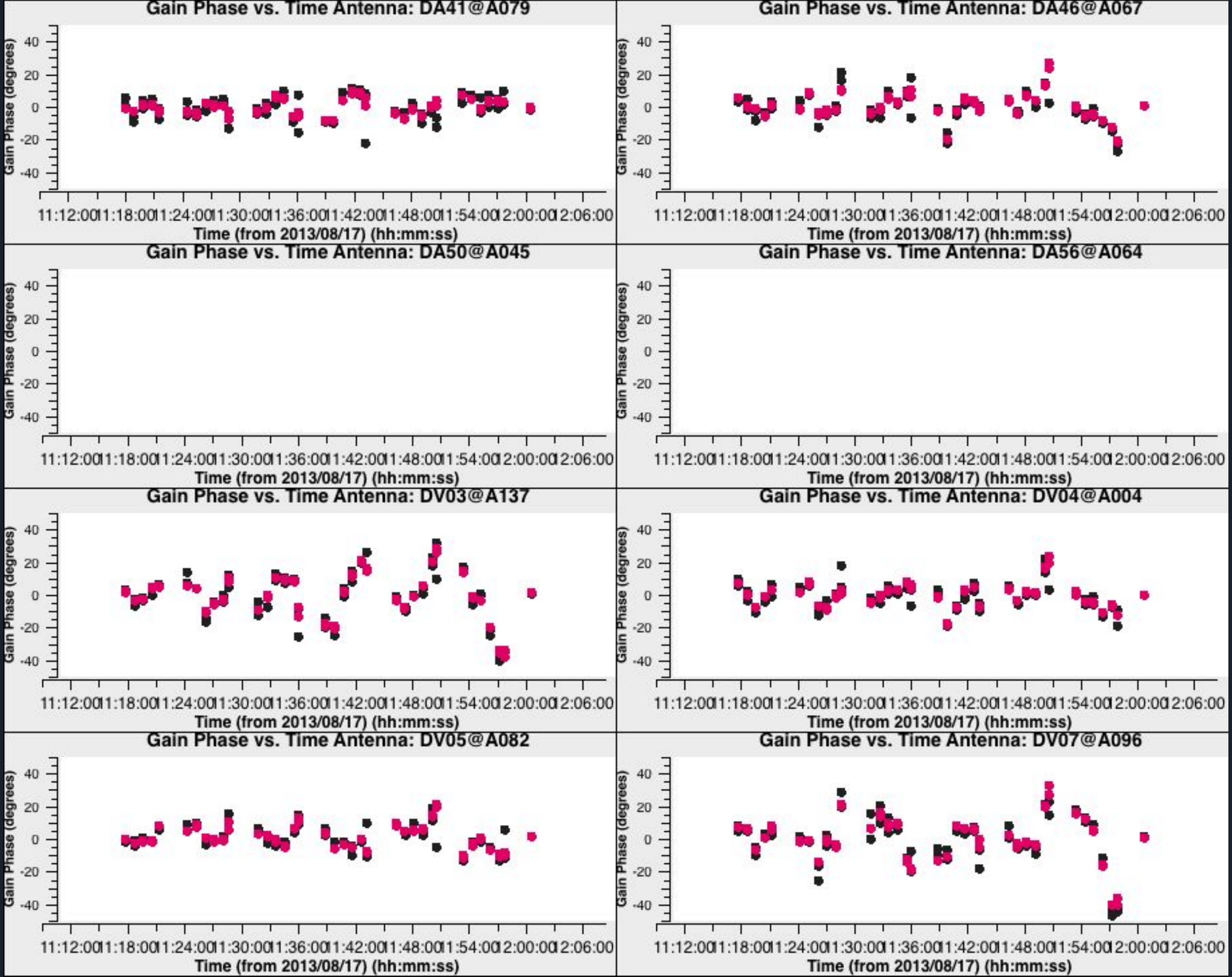
```
gaincal(vis=obj,
        caltable='X1de2_VYCMa_325_cont.ph2.solint_60s.tb',
        solint='60s',
        gaintable=['X1de2_VYCMa_325_cont.ph1.solint_inf.tb'],
        spwmap=[0,1],
        refant=refantenna,
        spw=contchans,
        calmode='p',
        gaintype='G',
        minsnr=3)
mysteps = [11]
```

```
applycal(vis = obj,
         field='Vy Cma',
         spw='0,1',
         spwmap=[[0,1],[0,1]],
         gaintable=['X1de2_VYCMa_325_cont.ph1.solint_inf.tb',
                    'X1de2_VYCMa_325_cont.ph2.solint_60s.tb'],
         calwt = False,
         applymode='calonly',
         flagbackup = False)
mysteps = [12]
```

When we make our new image (step 13): `rms 0.001, peak 0.199, snr 151`

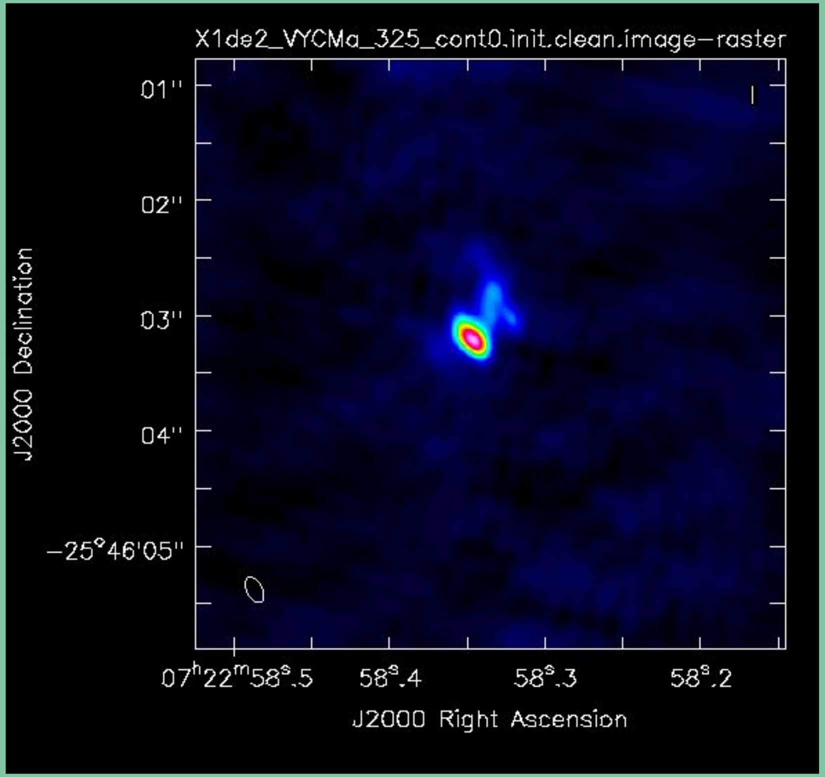
Any solution interval ~40-80s is a good balance between tracing the phase stream and the amount of flagged solutions (see ADVANCED presentation for the choice of solution interval based on SNR)

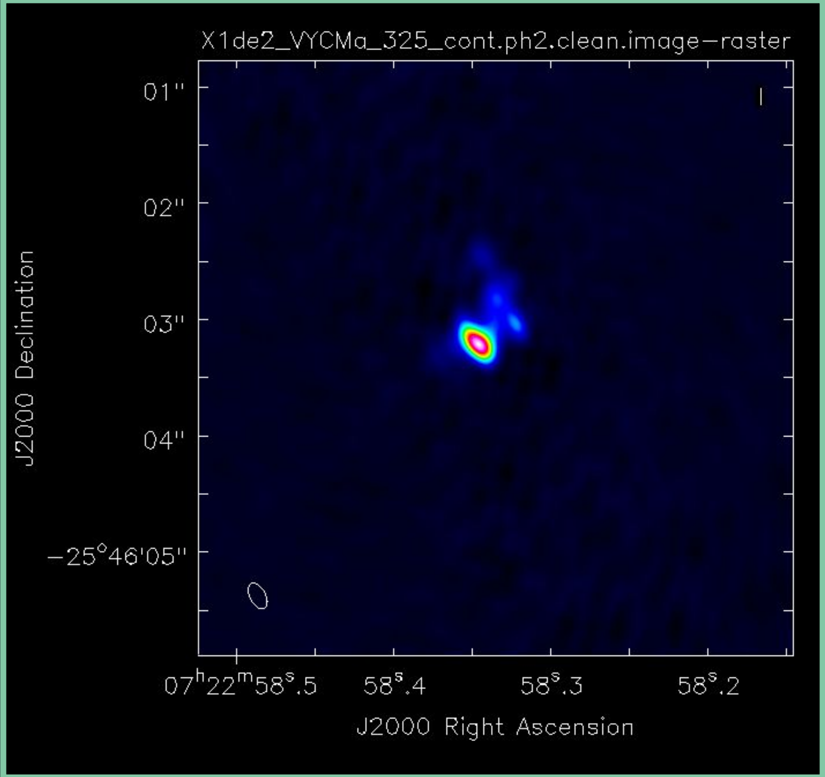
solint=60s


Gain Phase vs. Time Antenna: DA41@A079
Gain Phase vs. Time Antenna: DA46@A067
Gain Phase vs. Time Antenna: DA50@A045
Gain Phase vs. Time Antenna: DA56@A064
Gain Phase vs. Time Antenna: DV03@A137
Gain Phase vs. Time Antenna: DV04@A004
Gain Phase vs. Time Antenna: DV05@A082
Gain Phase vs. Time Antenna: DV07@A096
Gain Phase (degrees)
40
20
0
-20
-40
11:12:00
11:18:00
11:24:00
11:30:00
11:36:00
11:42:00
11:48:00
11:54:00
12:00:00
12:06:00
Time (from 2013/08/17) (hh:mm:ss)

```
4 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/12:02:26.9
1 of 36 solutions flagged due to SNR < 3 in spw=1 at 2013/08/17/12:02:26.9
Solint: 40s
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:15:36.7
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:21.0
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:02.2
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:43.8
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:41.1
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:22:46.2
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:27:25.7
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:30:48.9
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:32:14.5
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:32:56.0
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:33:34.0
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:34:53.4
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:35:28.4
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:39:15.5
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:39:56.8
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:40:38.3
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:41:54.4
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:42:35.7
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:46:34.6
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:47:18.9
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:48:00.2
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:48:41.7
3 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:49:19.7
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:50:39.1
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:51:14.1
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:54:16.9
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:55:01.2
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:55:42.5
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:56:24.0
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:57:40.1
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:58:21.4
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:58:56.4
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/12:02:14.8
Solint: 60s
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:15:46.1
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:16:49.6
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:50.1
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:41.1
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:22:55.6
4 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:27:37.8
1 of 36 solutions flagged due to SNR < 3 in spw=1 at 2013/08/17/11:27:37.8
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:33:02.4
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:35:40.5
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:38:40.7
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:39:44.2
3 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:43:22.8
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:46:44.1
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:48:48.0
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:50:51.9
4 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:51:26.2
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:55:29.9
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:56:30.3
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:57:30.6
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:58:34.2
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:59:08.5
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/12:02:14.8
Solint: 80s
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:17:24.6
2 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:19:41.1
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:38:53.5
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:40:19.2
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:46:56.8
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:48:22.6
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:50:55.0
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:54:39.1
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:56:04.9
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/11:57:24.3
1 of 36 solutions flagged due to SNR < 3 in spw=0 at 2013/08/17/12:02:14.8
```

Comparison - Image





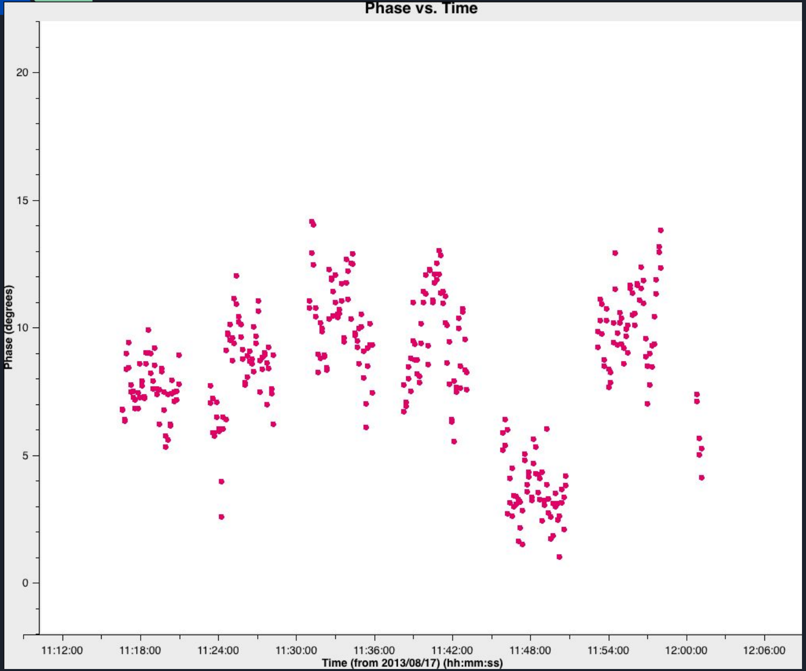
SNR~ 51 -> SNR ~ 151

Self-calibrated visibilities

You can see the effect of our phase-only self-calibration steps in the phases of the visibilities

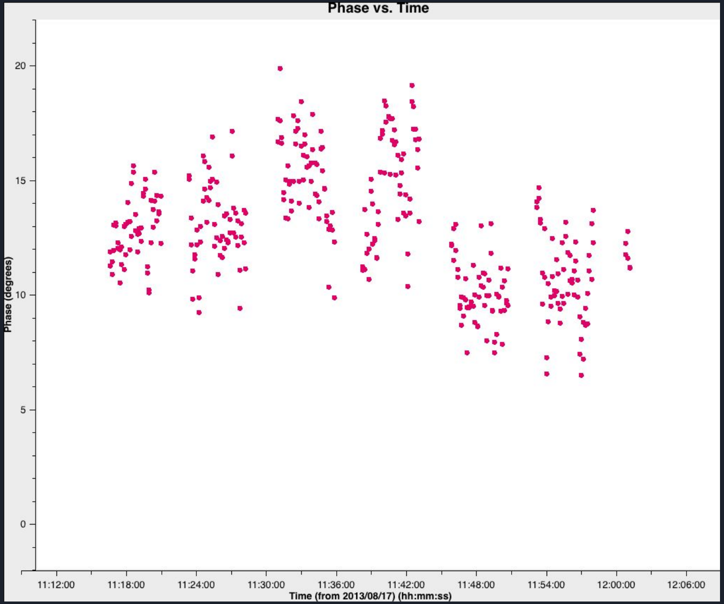
Prior to Self-Cal (initial map, `data` column)

Spw1, All baselines



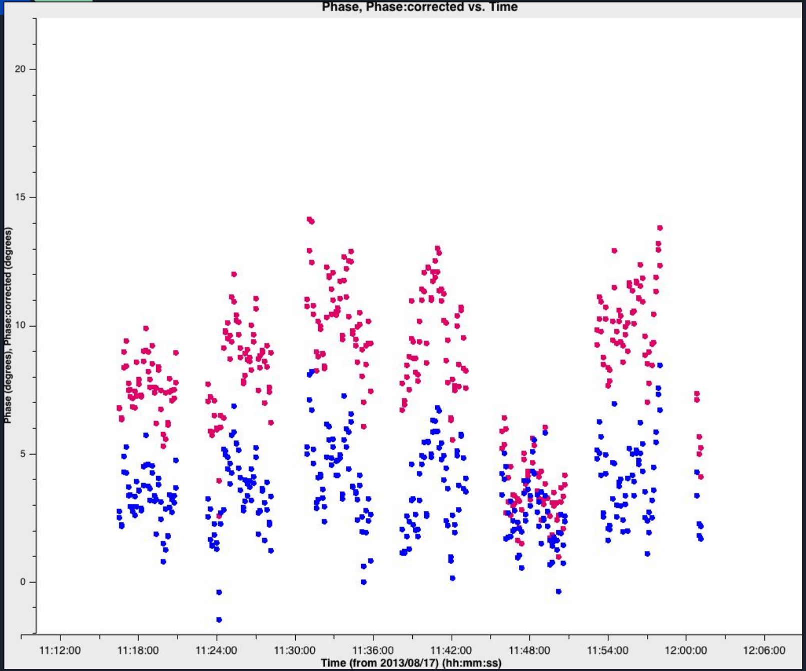
All antennas

antenna = 10



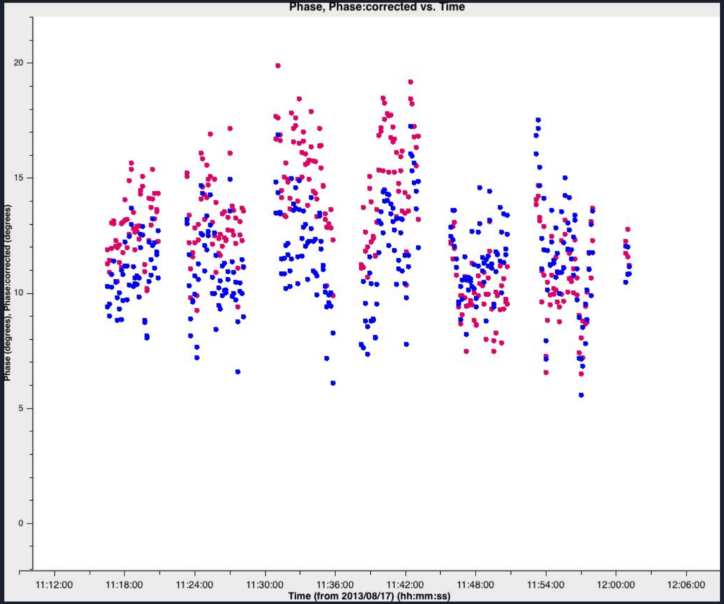
After 1st round (solint='inf')

Spw1, All baselines

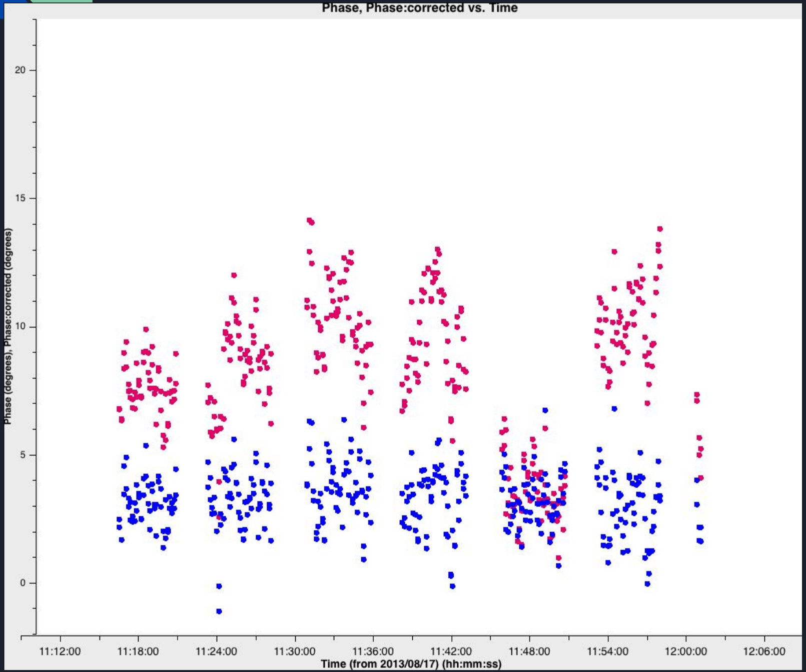


All antennas

antenna = 10



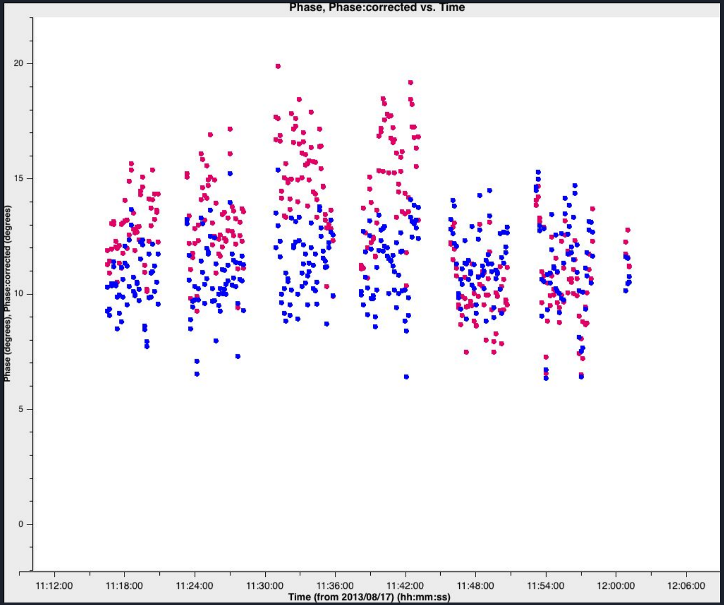
After 2nd round (solint='inf' + ’60s’)



All antennas

Spw1, All baselines

antenna = 10



Q: When do I stop doing rounds of self-cal?
A: When things stop getting better

- Continue self-calibration while you can provide a better model or a more accurate `solint`
  - No point in going to a `solint` so short it is just moving noise around
    - See Advanced section for choosing `solint`
  - If a set of solutions look like noise, or after applying them the S/N or target flux goes down:
    - Don't apply them! Go back a step,
      - Maybe you have done enough?
      - Need a better model or different averaging?
  - As long as solutions are good:
    - Stop when you reach the predicted image noise and/or there is no S/N improvement
      - Always check that the image is reasonable
- For good S/N (here, ~100) try amplitude self-cal applying phase solutions
  - See Advanced section
- Stop (or try something different) if S/N does not increase or target flux falls

Now we start Self-calibration Advanced session

Additional slides

Resources/More information

VLA Self-cal Tutorial (https://casaguides.nrao.edu/index.php?title=VLA_Self-calibration_Tutorial-CASA5.7.0)

ALMA Self-cal Tutorial (https://casaguides.nrao.edu/index.php?title=First_Look_at_Self_Calibration)

Advanced Gain Calibration Techniques (Brogan et al. 2018): https://arxiv.org/abs/1805.05266

INAF (http://www.alma.inaf.it/images/Selfcalibration.pdf)

NAASC (https://science.nrao.edu/facilities/alma/naasc-workshops/nrao-cd-wm16/Selfcal_Madison.pdf)

ALLEGRO (https://www.alma-allegro.nl/wp-content/uploads/2018/10/Allegro_CASAtrainingDay2018_selfcalupdate.pdf)

ERIS (https://www.chalmers.se/en/researchinfrastructure/oso/events/ERIS2019)

DARA (http://www.jb.man.ac.uk/DARA/)

Synthesis Imaging Taylor, Carilli & Perley (http://www.phys.unm.edu/~gbtaylor/astr423/s98book.pdf)

Calibration application modes: the `'applymode'` parameter in `applycal`

In `applycal` you can select different modes of applying your calibration tables by means of the **`applymode`** parameter:

- `applymode = 'calflag'`: this is the default mode. It will apply all flags from a calibration table and the calibration itself to the remaining visibilities after flagging.
- `applymode = 'calonly'`: this will apply the calibration data and weights but will not flag.
- `applymode = 'flagonly'`: it will apply flags but not the calibration itself.

CASA applycal documentation

The `spwmap` parameter in `applycal`

This parameter controls the way gaintables are applied to the different spectral windows. Some examples for a dataset with 4 spectral windows [0,1,2,3]:

- `spwmap = [0,0,1,1]` : this option means apply the caltable solutions from spectral window 0 to spectral windows 0 and 1 and solutions from spectral window 1 to spectral windows 2 and 3.
- `spwmap = [[0,0,1,1],[0,1,0,1]]`: this is an example on how to specify the spectral window mapping when two gaintables are applied. If there are multiple gaintables to be applied, this should be reflected in the parameter `spwmap`.
- The task `browsetable` can help to identify the SPW IDs to use in `spwmap`.

CASA applycal documentation