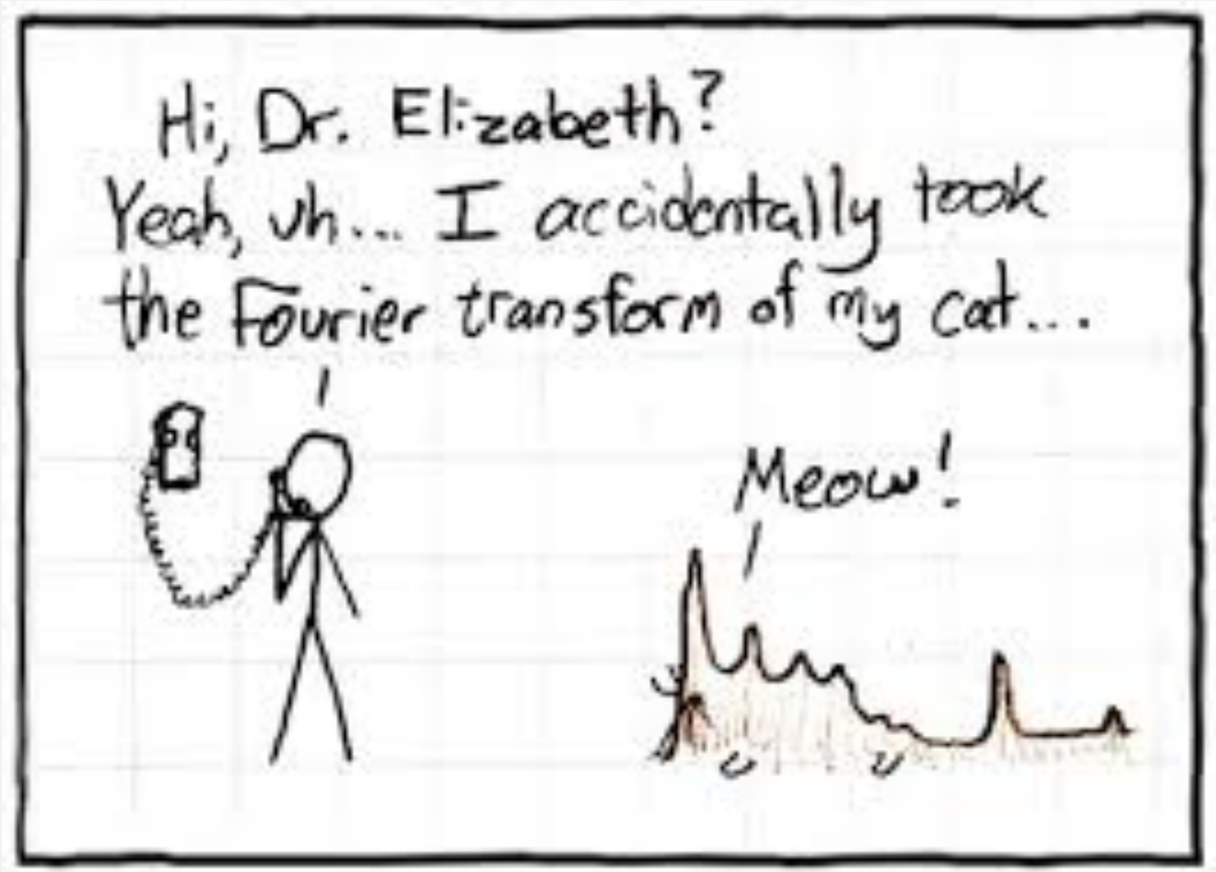
RADIO INTERFEROMETRIC IMAGING

The fun of Fourier Transforms…

Mubela Mutale

Zambia – DARA Unit 4 – July 2019

Talk credits: J. Radcliffe, based on A. Offringa’s & N. Jackson’s 2015 ERIS talks and T. Muxlow’s 2013 ERIS talk


Hi, Dr. Elizabeth?
Yeah, uh... I accidentally took the Fourier transform of my cat...
Meow!

OUTLINE

1. Deconvolution
   - CLEAN
   - Windowing
   - CASA clean/tclean
2. Data gridding & weighting
   - uv weighting
   - Telescope weighting
3. Wide-field imaging limits
   - Smearing
   - Non-coplanar baselines
   - Primary beam
4. Signal to noise & dynamic range

1. DECONVOLUTION

The basic operation of an (ideal) interferometer baseline measures (small sky approximation, w→0):

$$V(u,v) \approx \iint I(l,m) e^{-2\pi i(ul+vm)} dl dm$$

We can, in principle, measure I(l,m) for all u,v. We can then use a Fourier transform to recover the sky brightness distribution:

$$I(l,m) \approx \iint V(u,v) e^{2\pi i(ul+vm)} du dv$$

However V(u,v) is not known everywhere but is sampled at particular places on the u-v plane

Nb: (l, m, n) notation is essentially the same as (x, y, z) coordinates used in the prev. talks

DECONVOLUTION

This sampling function can be described by S(u,v) and is equal to 1 when the uv plane is sampled and zero otherwise:

$$I^D(l,m) = \iint V(u,v)S(u,v)e^{2\pi i(ul+vm)}dudv$$

$I^D(l,m)$ is known as the 'dirty image' and is related to the real sky brightness distribution by (using convolution theorem of FT):

$$I^D(l,m) = I(l,m) * B$$

Where *B* is known as the 'dirty beam' or the 'point spread function' and is the FT of the sampling function.

$$B(l,m) = \iint S(u,v)e^{2\pi i(ul+vm)}dudv$$

CASA IMAGE CONSTRUCTION

```
#  tclean :: Radio Interferometric Image Reconstruction
vis                 =         ''        #  Name of input visibility file(s)
selectdata          =       True        #  Enable data selection parameters
     field          =         ''        #  field(s) to select
     spw            =         ''        #  spw(s)/channels to select
     timerange      =         ''        #  Range of time to select from data
     uvrange        =         ''        #  Select data within uvrange
     antenna        =         ''        #  Select data based on antenna/baseline
     scan           =         ''        #  Scan number range
     observation    =         ''        #  Observation ID range
     intent         =         ''        #  Scan Intent(s)

datacolumn          = 'corrected'       #  Data column to image(data,corrected)
imagename           =         ''        #  Pre-name of output images
imsize              = [256, 256]        #  Number of pixels
cell                = ['1.0arcsec']     #  Cell size
phasecenter         =         ''        #  Phase center of the image
stokes              =        'I'        #  Stokes Planes to make
projection          =      'SIN'        #  Coordinate projection (SIN, HPX)
startmodel          =         ''        #  Name of starting model image
specmode            =      'mfs'        #  Spectral definition mode (mfs,cube,cubedata)
     reffreq        =         ''        #  Reference frequency

gridder             = 'standard'        #  Gridding options (standard, wproject, widefield,
                                        #   mosaic, awproject)
     vptable        =         ''        #  Name of Voltage Pattern table
     pblimit        =        0.2        #  >PB gain level at which to cut off normalizations

deconvolver         =   'hogbom'        #  Minor cycle algorithm
                                        #   (hogbom,clark,multiscale,mtmfs,mem,clarkstokes)
restoration         =       True        #  Do restoration steps (or not)
     restoringbeam  =       ['']        #  Restoring beam shape to use. Default is the PSF main
                                        #   lobe
     pbcor          =      False        #  Apply PB correction on the output restored image

outlierfile         =         ''        #  Name of outlier-field image definitions
weighting           =  'natural'        #  Weighting scheme (natural,uniform,briggs)
     uvtaper        =      False        #  uv-taper on outer baselines in uv-plane

niter               =        500        #  Maximum number of iterations
     gain           =        0.1        #  Loop gain
     threshold      =   '0.0mJy'        #  Stopping threshold
     nsigma         =        0.0        #  Multiplicative factor for rms-based threshold
                                        #   stopping
     cycleniter     =         -1        #  Maximum number of minor-cycle iterations
     cyclefactor    =        1.5        #  Scaling on PSF sidelobe level to compute the minor-
                                        #   cycle stopping threshold.
     minpsffraction =       0.05        #  PSF fraction that marks the max depth of cleaning in
                                        #   the minor cycle
     maxpsffraction =        0.8        #  PSF fraction that marks the minimum depth of
                                        #   cleaning in the minor cycle
     interactive    =      False        #  Modify masks and parameters at runtime

usemask             =     'user'        #  Type of mask(s) for deconvolution:  user, pb, or
                                        #   auto-multithresh
     mask           =         []        #  Mask (a list of image name(s) or region file(s) or
                                        #   region string(s) )
     pbmask         =        0.0        #  primary beam mask

restart             =       True        #  True : Re-use existing images. False : Increment
                                        #   imagename
savemodel           =     'none'        #  Options to save model visibilities (none, virtual,
                                        #   modelcolumn)
calcres             =       True        #  Calculate initial residual image
calcpsf             =       True        #  Calculate PSF
parallel            =      False        #  Run major cycles in parallel
```

- tclean is the CASA imaging routine (this replaced clean in CASA v4.6 and earlier)

- To achieve a basic image, need to set:
  - vis - your data (measurement set)
  - imagename (output image)
  - niter - no. of CLEAN iterations (next slide)
  - imsize - size of the image in pixels (needs to be as small as possible to decrease computation time)
  - cell - angular extent of each pixel (need to adequately sample the psf)

  Rule of thumb:

$$\mathrm{cell} \sim \lambda_{\mathrm{f}}/3B$$

λ_f - wavelength of highest frequency channel

B - longest baseline length

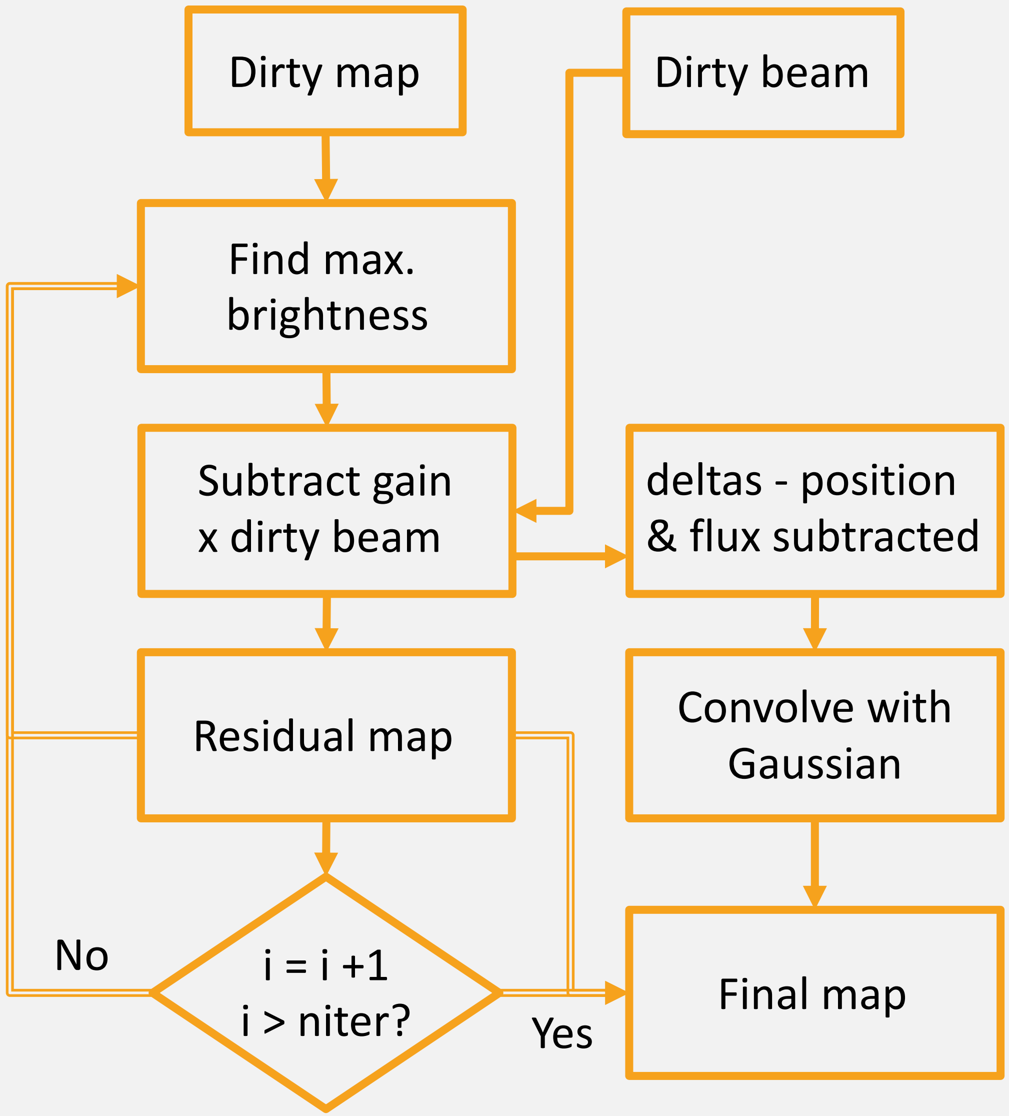
DECONVOLUTION

To recover the real brightness distribution we just need to deconvolve… easier said than done:

- A vast number of images are consistent with the data inc. the dirty beam.
- We need to take a Bayesian approach - supply priors (i.e. extra information/ assumptions) so we can find the most probable brightness distribution.
- Simplest scheme (but not only): Sky is mostly empty and consists of a finite number of unresolved point sources.

→ The basis of the Hogbom CLEAN algorithm (1974)

HOGBOM CLEAN & VARIANTS

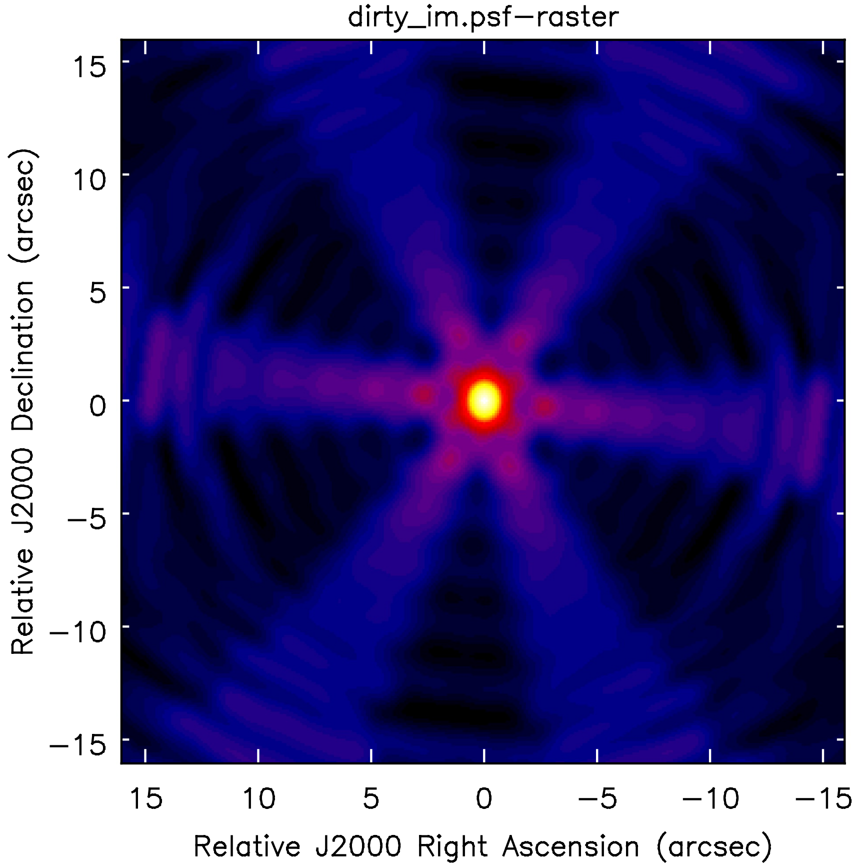


- A brute force deconvolution algorithm using the dirty beam
- Uses prior that the sky consists of unresolved point sources modelled by Dirac delta functions
- Other versions such as Clark, multiscale are variants of this algorithm

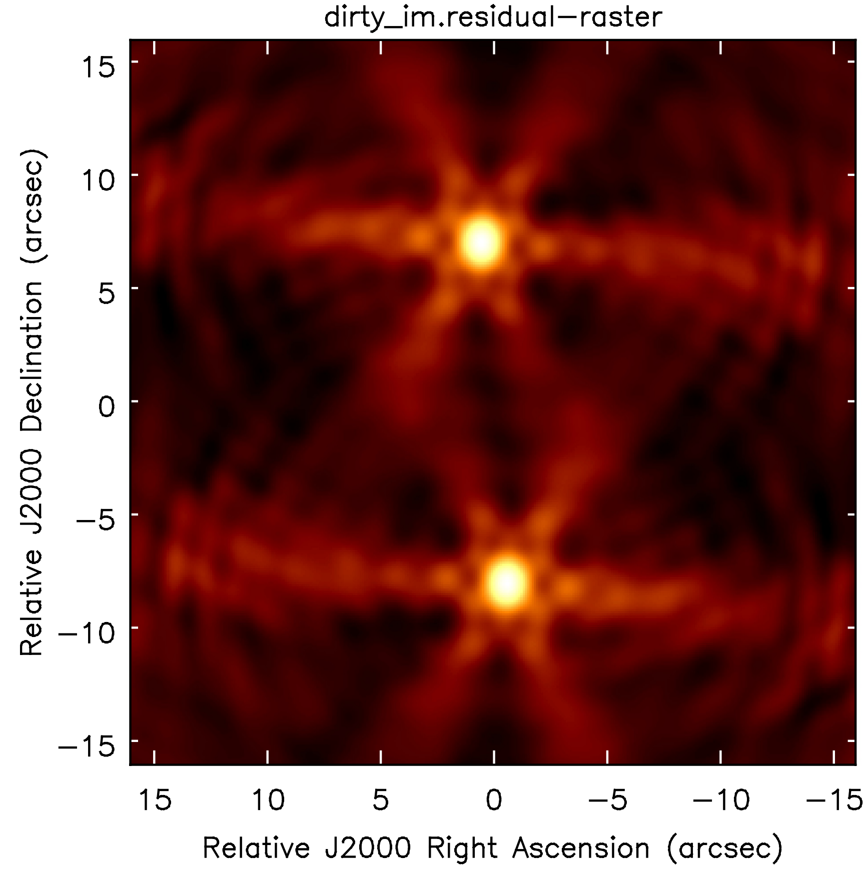
CLEAN DECONVOLUTION

JVLA simulation, 2hr observation targeting two 0.1 Jy point sources + some phase corruption included

Dirty beam

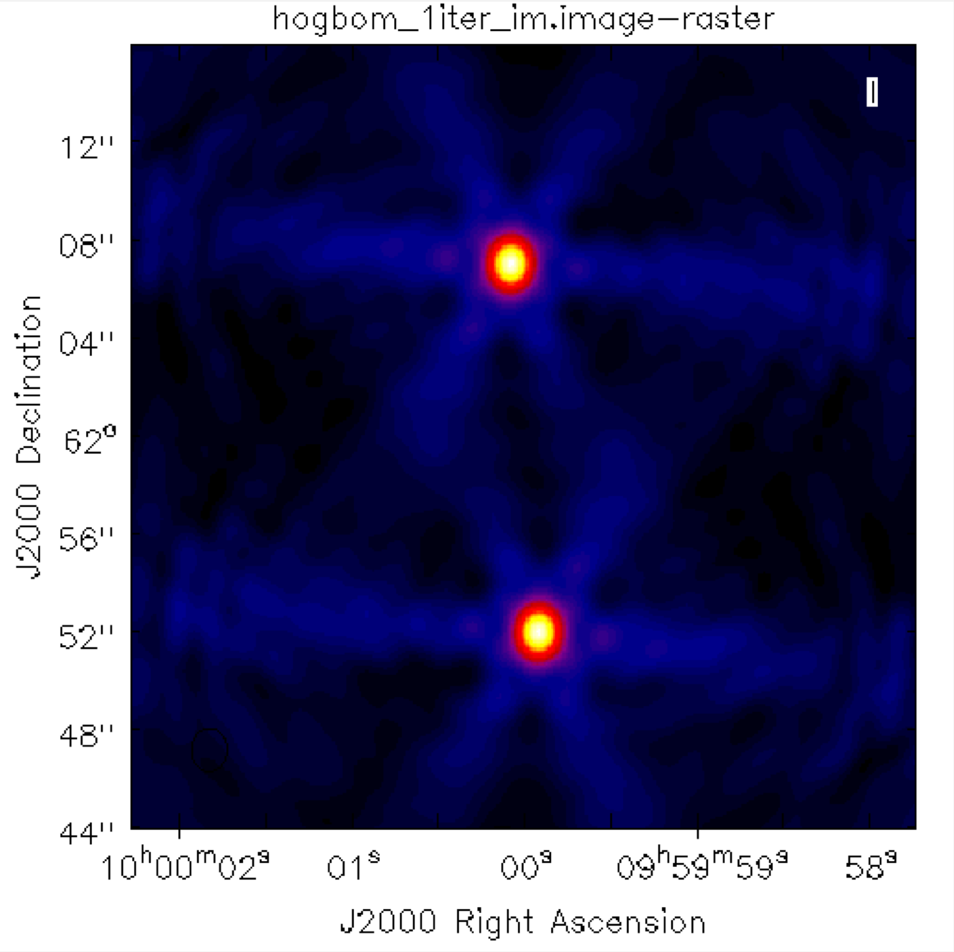


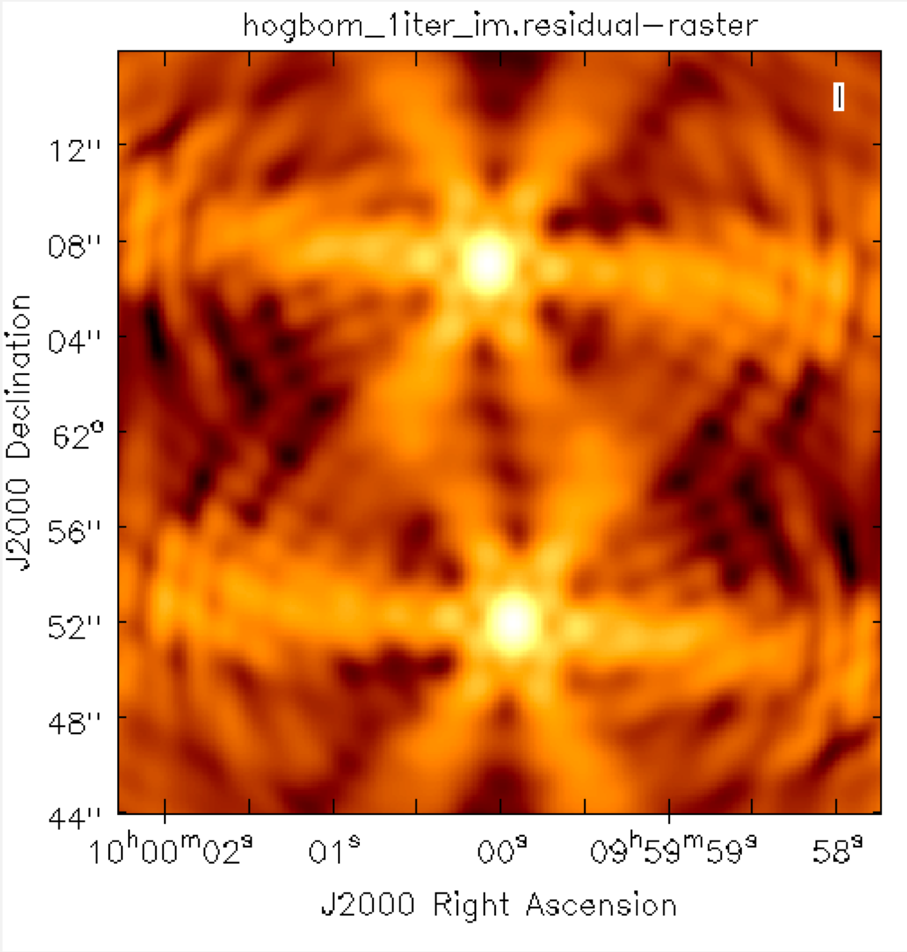
Dirty image



CLEAN DECONVOLUTION

Hogbom CLEAN
Image & residual after 1 iteration with 0.5 gain

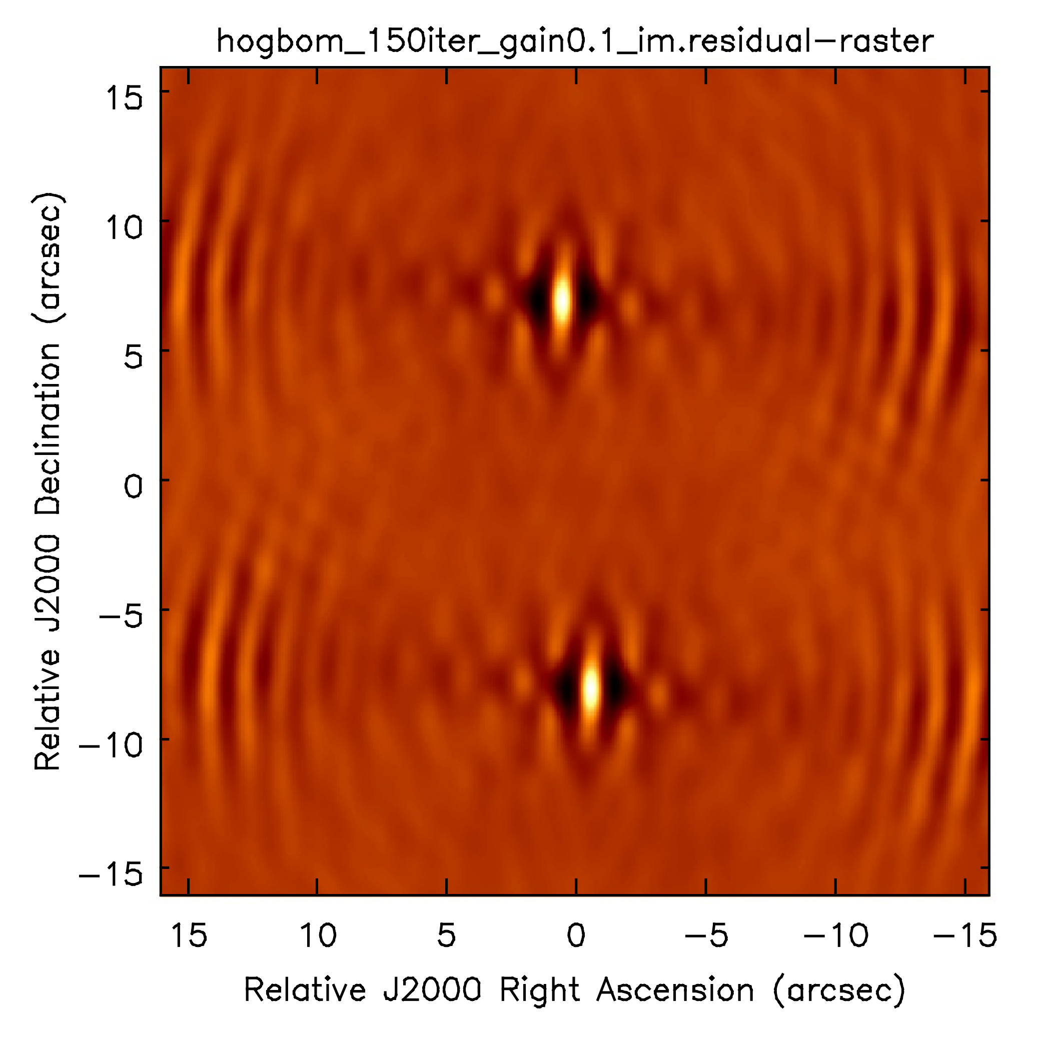




CLEAN DECONVOLUTION

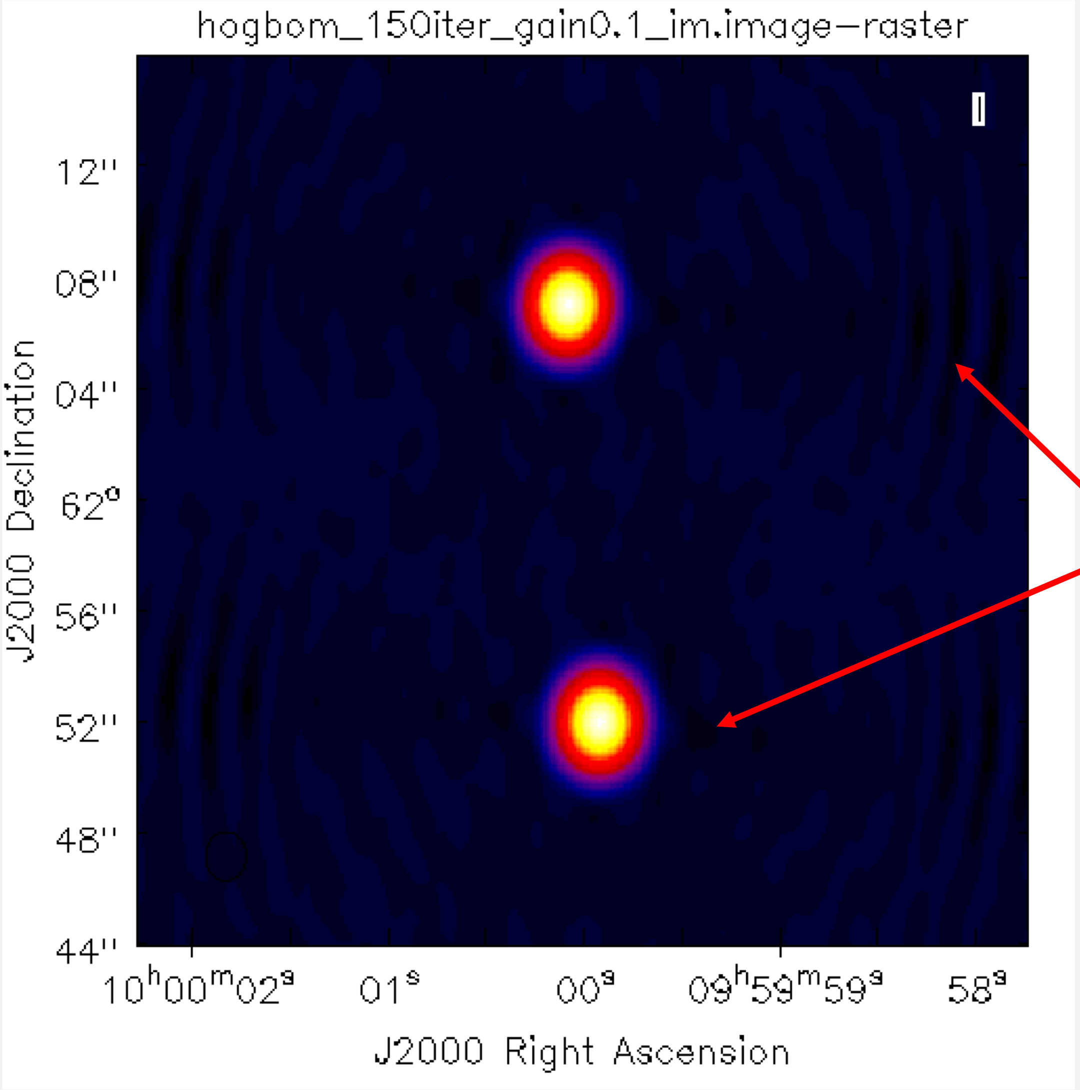
Hogbom CLEAN

Residual after 150 iterations with 0.1 gain



CLEAN DECONVOLUTION

CLEAN map (residual+CLEAN components) after 150 iterations



Some artefacts left from deconvolution

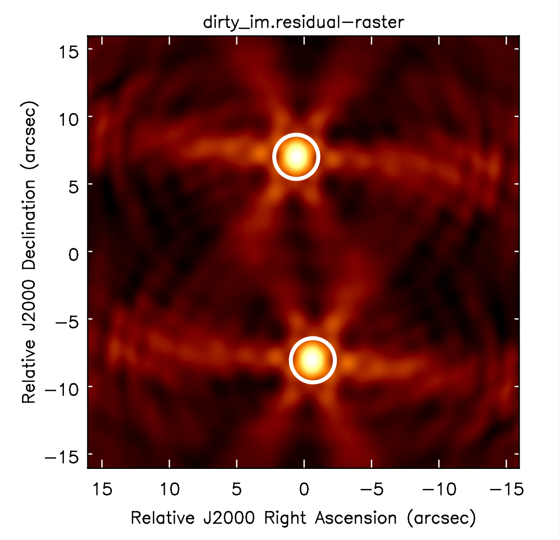
CLEAN DECONVOLUTION

CLEAN is far from perfect, but we can lend it a hand:

CLEAN consists of two 'cycles':

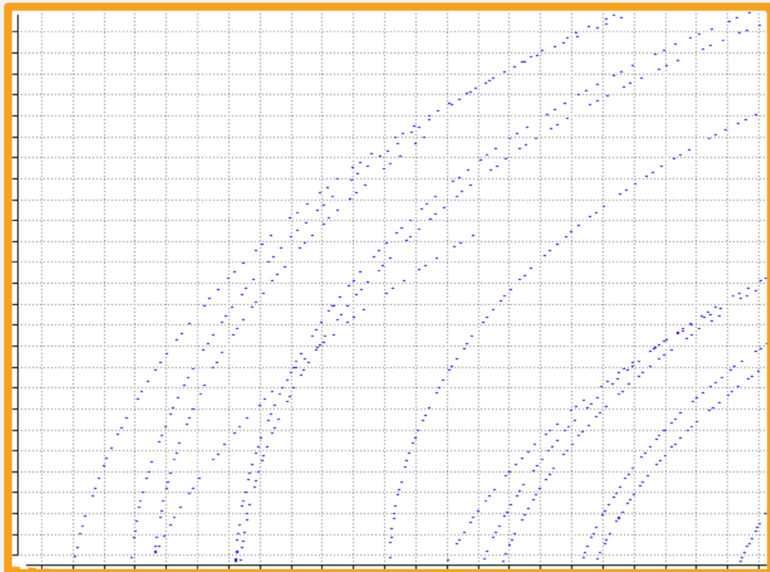
I. Minor cycles - subtract subimages of the dirty beam
II. Major cycles - Fourier Transform residual map and subtract

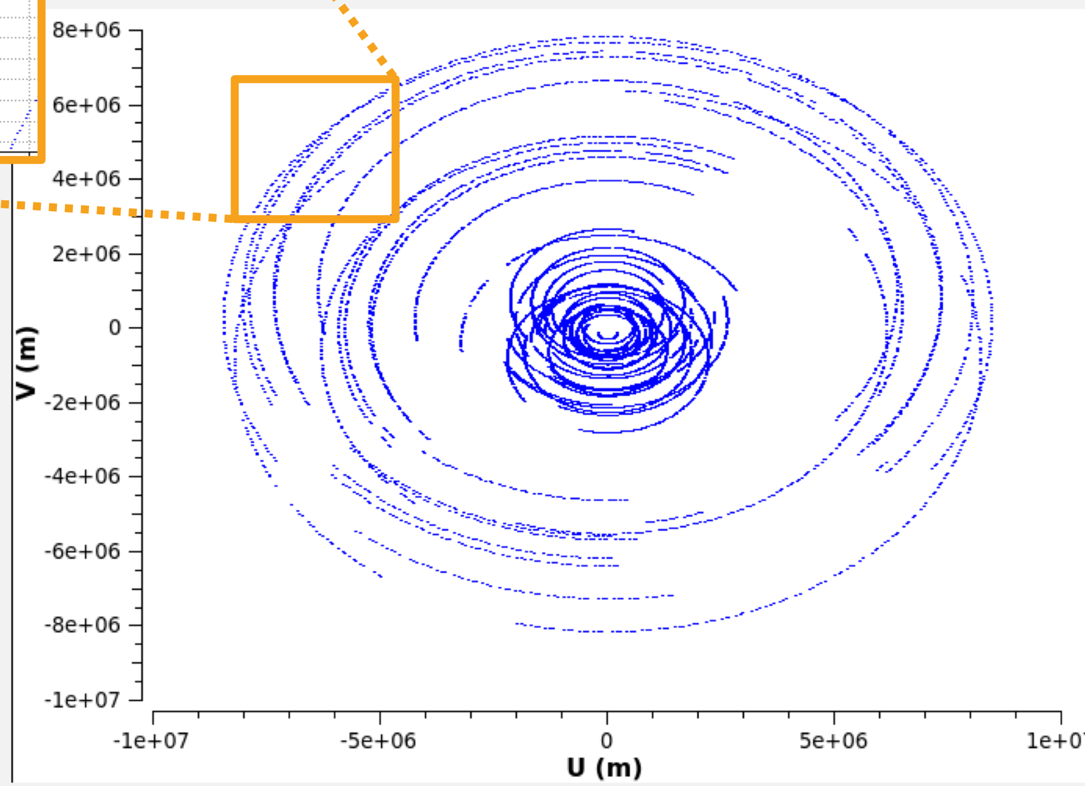
We can use windowing to tell the algorithm where the flux lies.
This should be used when you **know** the flux you see is real!



2. WEIGHTING

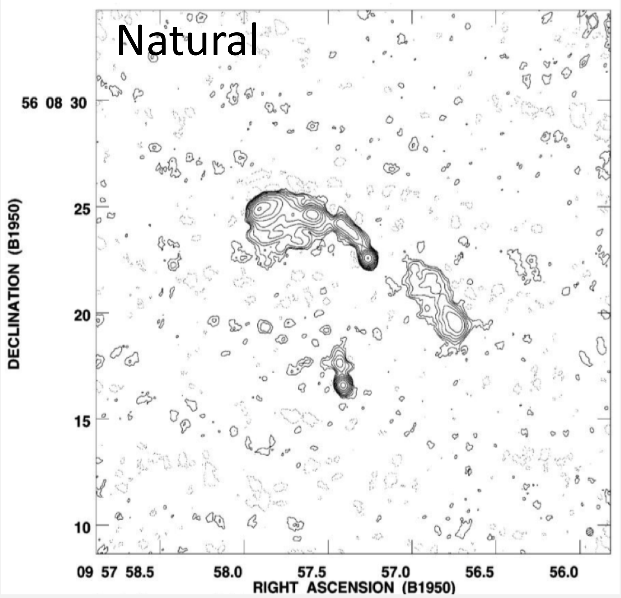
Integrations are distributed over a greater number of sampled grid points in the outer uv plane than the inner regions

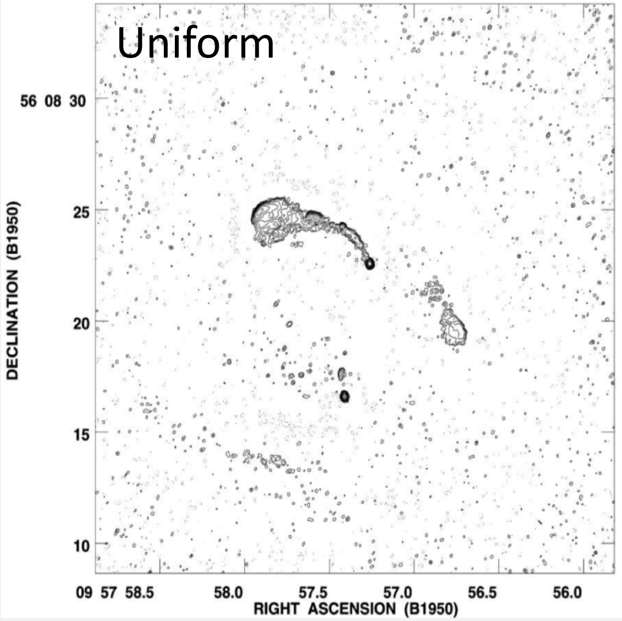




- Data interpolated on 2^n grid
- Weights unmodified by local density - ‘Natural’
- Weights divided by local density of points - ‘Uniform’

UV WEIGHTING





Natural weighted images have low spatial frequencies are weighted up (due to gridding) and gives:

- Best S/N
- Worse resolution

Uniform weighted images low have spatial frequencies weighted down and the data are not utilised optimally (may be subject to a deconvolution striping instability)
resulting in:

- Worse S/N
- Best resolution

Compromises exist:

- Briggs (robust) weighting parameter -5 to +5. (next slide)

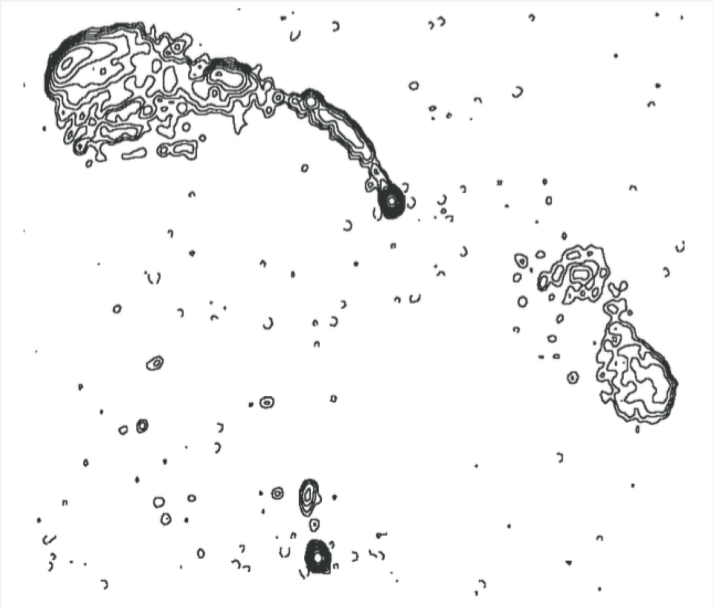
Implementation in CASA tclean/clean

```
weighting          =   'natural'        #  Weighting of uv (natural, uniform,
                                        #   briggs, ...)
```

UV WEIGHTING: 'BRIGGS WEIGHTING'

- Originally derived as a cure for striping – Natural weighting is immune and therefore most 'robust'
- Varies effective weighting as a function of local u-v weight density
  - Where weight density is low – effective weighting is natural
  - Where weight density is high – effective weighting is uniform

Robust 0 image

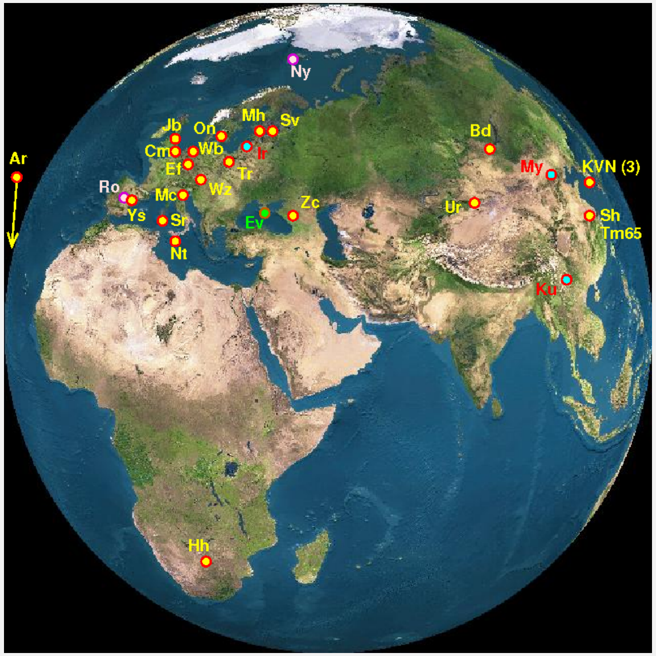


- Modifies the variations in effective weight found in uniform weighting → more efficient use of data & lower thermal noise
- ROBUST = – 5 is nearly pure uniform ROBUST = + 5 is nearly pure natural
  ROBUST = 0 is a good compromise (Contoured)

- Can produce images close to uniform weighting resolution with noise levels close to natural weighting. See CASA webpage for other weighting schemes!

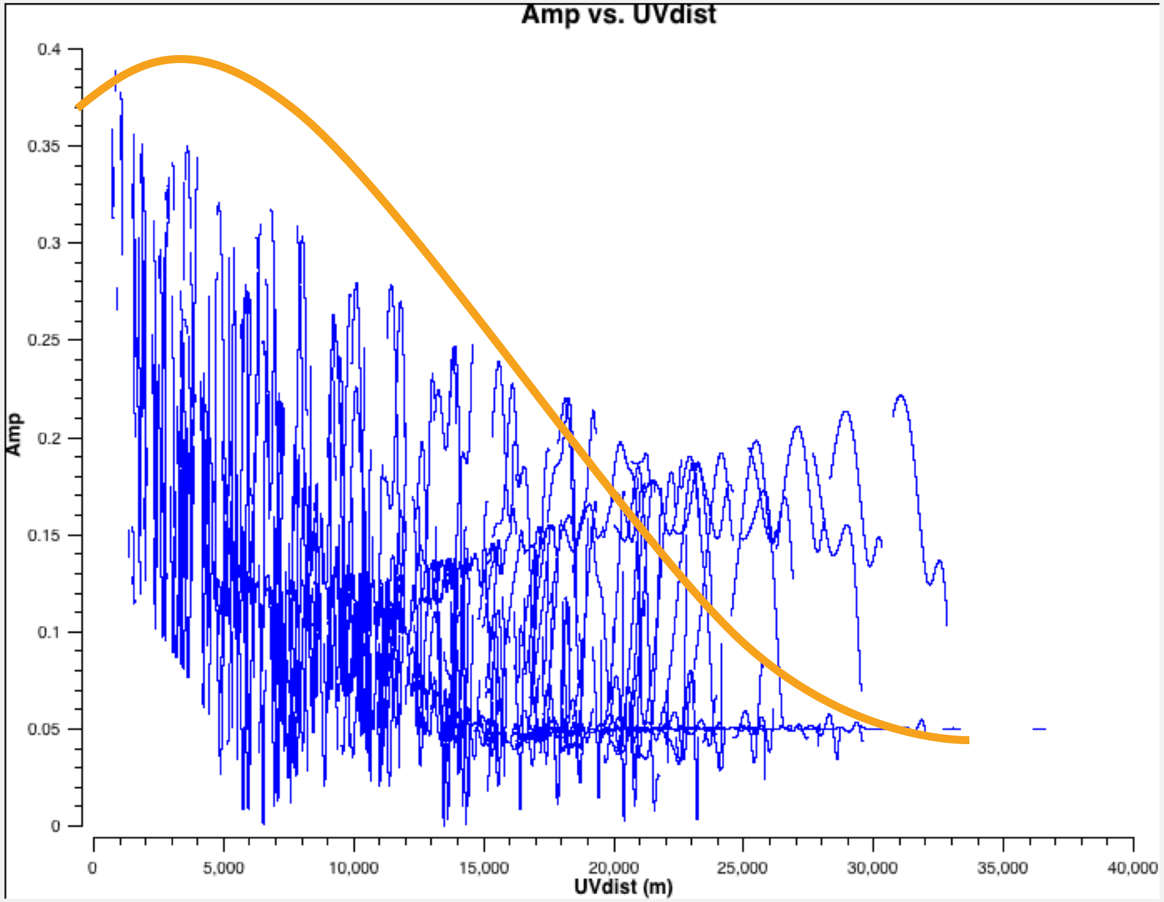
WEIGHTING BY TELESCOPE

- Many arrays are heterogeneous e.g. e-MERLIN, EVN & AVN (when built)
- To get the best S/N need to increase weighting on larger telescopes so they contribute more.
- Nb. this can change the resolution depending on the baseline distribution.



UV TAPERING

Gaussian u-v taper or u-v range can smooth the image but at the expense of sensitivity since data are excluded or data usage is non-optimal

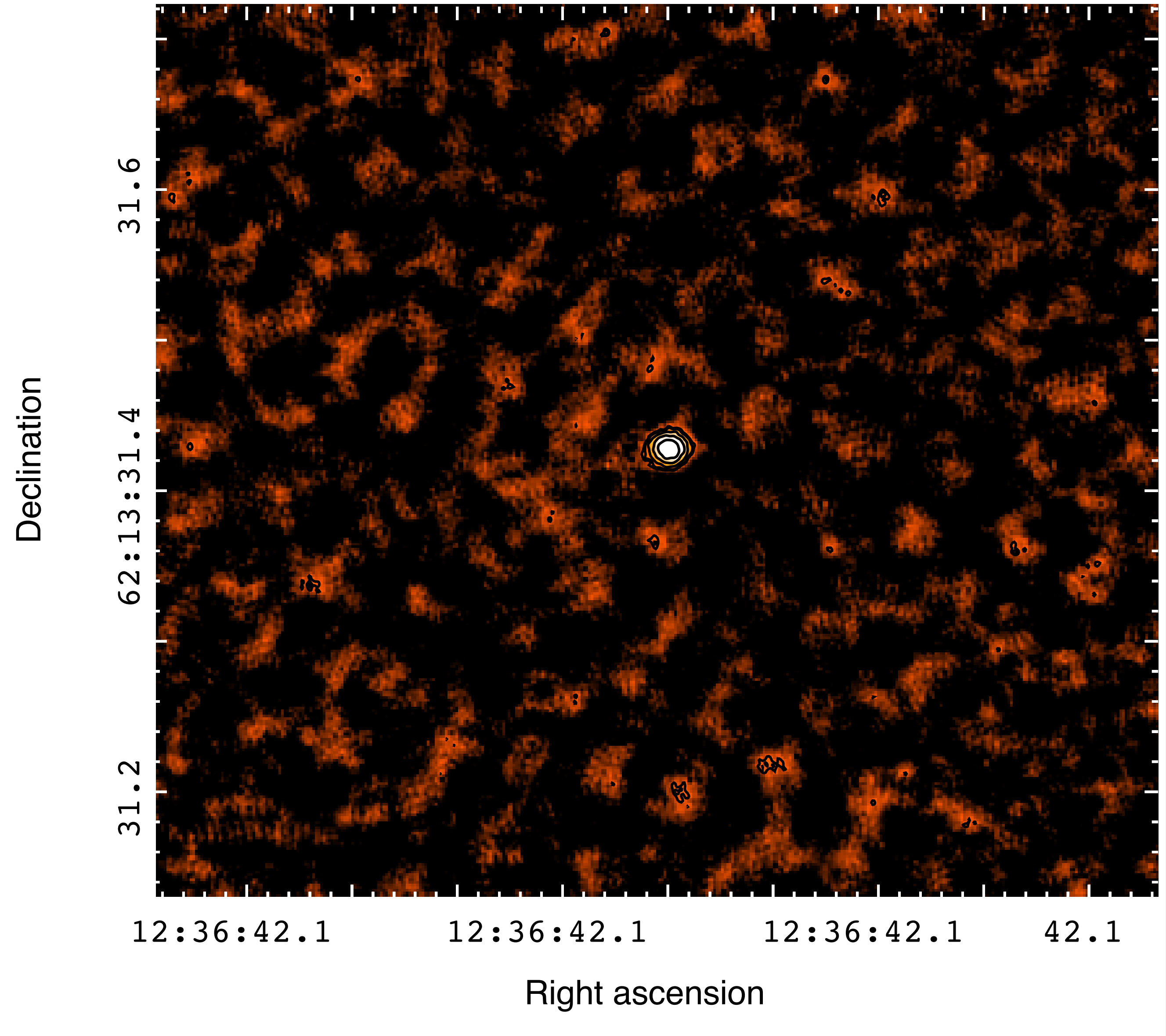


Can compromise image quality in VLBI arrays by severely restricting the *u-v* coverage

Controlled by the uvtaper parameter in CASA task tclean/clean

UV TAPERING

eMERLIN + EVN images of J123642+621331



3. WIDE-FIELD IMAGING

A 'wide-field' image is defined as:

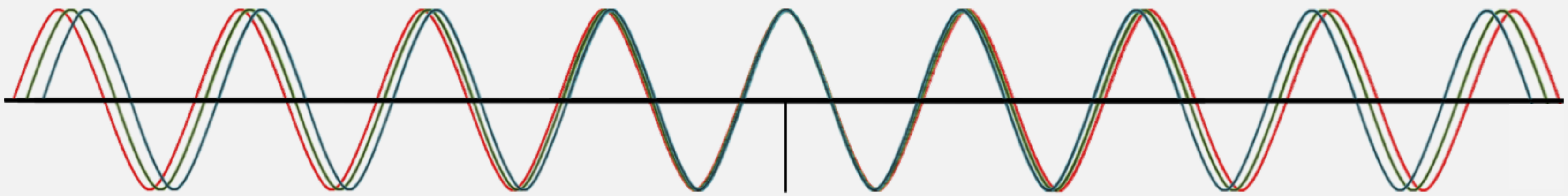
- An image with large numbers of resolution elements across them
- Or multiple images distributed across the interferometer primary beam

In order to image the entire primary beam you have to consider the following distorting effects:

1. Bandwidth smearing
2. Time smearing
3. Non-coplanar baselines (or the 'w' term) - Covered in advanced imaging
4. Primary beam response

BANDWIDTH SMEARING

Given a finite range of wavelengths



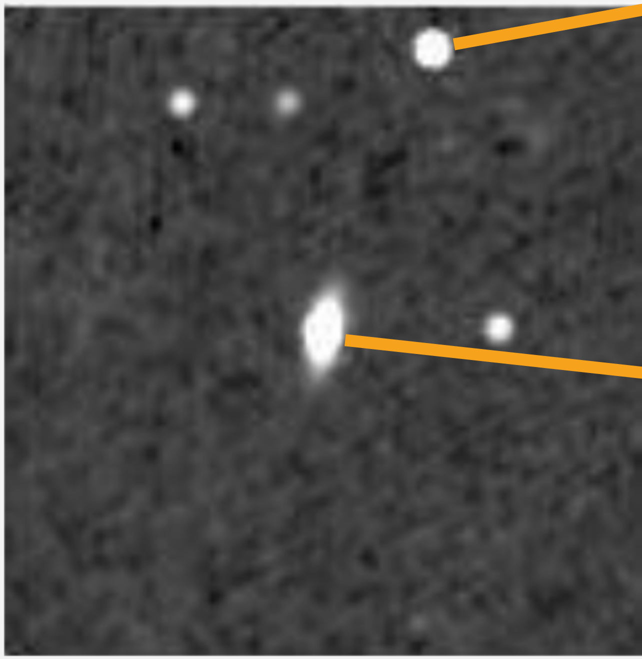
increasing radius ← pointing centre → increasing radius

- Fringe pattern is ok in the centre but, with higher relative delay, different colours are out of phase
- BW smearing can be estimated using: $\mathrm{FoV} \sim \dfrac{\lambda}{\Delta\lambda}\dfrac{\lambda}{B}$
- Can be alleviated by observing and imaging with high spectral resolution with many narrow frequency channels gridded separately prior to Fourier inversion (reduces $\Delta\lambda$).
- Detailed form of response depends on individual channel bandpass shapes

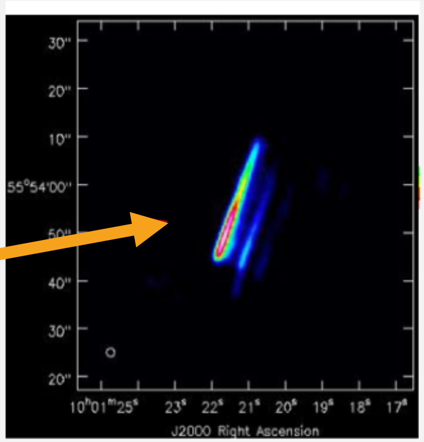
BANDWIDTH SMEARING

Credit T. Muxlow

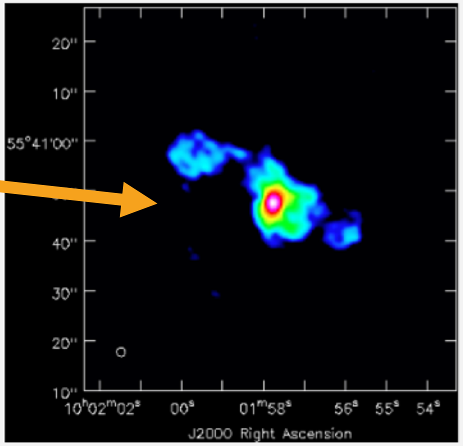
Example



NVSS image



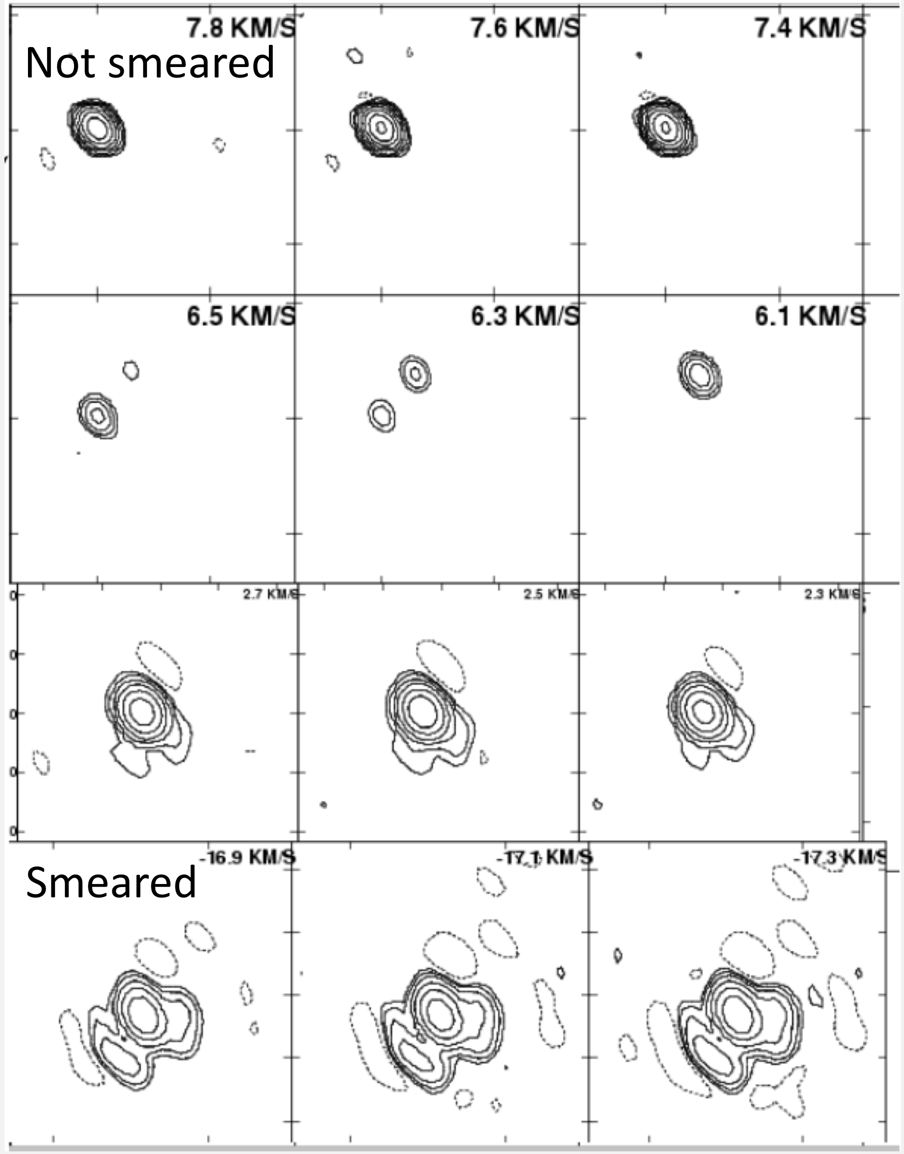
e-MERLIN



Effect is radial smearing, corresponding to radial extent of measurements in uv plane

TIME SMEARING

Credit N. Jackson



- Time-average smearing (de-correlation) produces tangential smearing

- Not easily parameterized. At declination +90° a simple case exists where percentage time smearing is given by:

$$\omega_e \delta t_{\rm int} \frac{\theta}{\theta_{\rm HPBW}}$$

- At other declinations, the effects are more complicated.

NON-COPLANAR BASELINES

Standard Fourier synthesis assumes planar arrays or small (l,m) - Only true for E-W interferometers e.g. WSRT

$$V(u,v,w) = \iint \frac{I(l,m)}{\sqrt{1-l^2-m^2}} e^{-2\pi i (ul+vm+w(\sqrt{1-l^2-m^2}-1))} dl dm$$

Need to take into account the 'w' term properly in wide-fields as:

- Errors increase quadratically with offset from phase-centre
- Serious errors result if: $\theta_{\mathrm{offset}}[\mathrm{rad}] \times \theta_{\mathrm{offset}}[\mathrm{beams}] > 1$
- Effects are severe when imaging the entire primary beam

NON-COPLANAR BASELINES

Result: We need to deal with $V(u,v,w)$ rather than just $V(u,v)$

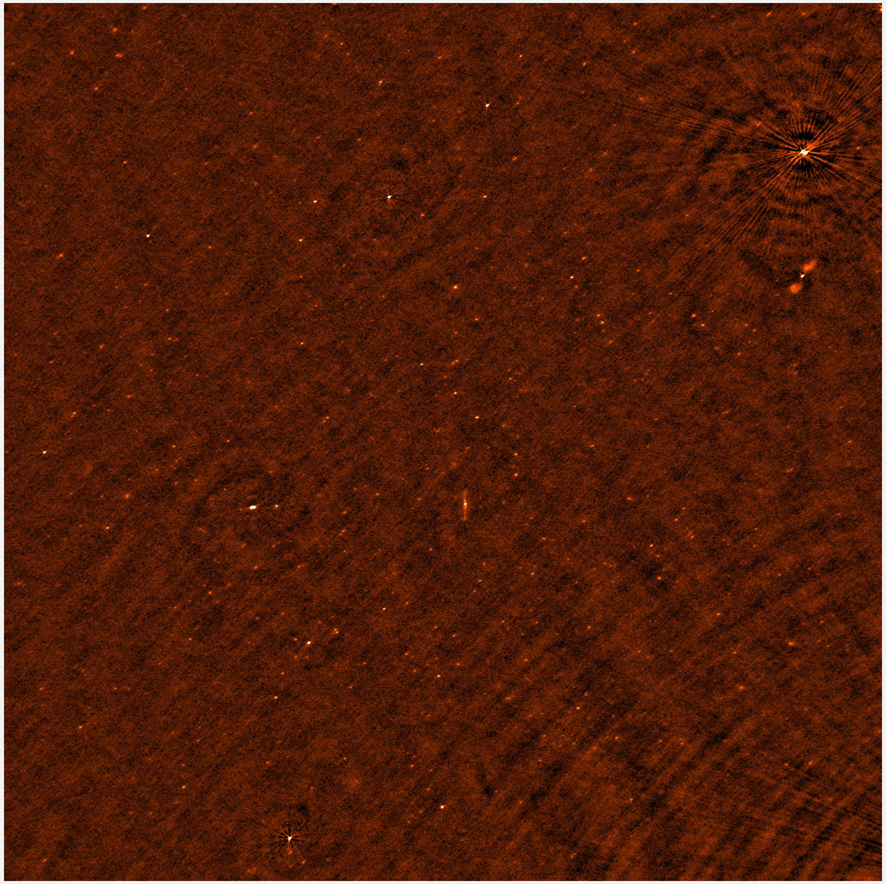
Two solutions available:

i. Faceting - split the field into multiple images to maintain l, m, w $\backsim$ 0 and stitch them together.

ii. w-projection - most used solution, project 3D sky brightness onto 2D tangent plane using w kernel.

See lecture on Advanced Imaging!

CONFUSION



JVLA image of GOODS-N showing confusion from a 0.25Jy source to the SE

- Bright radio sources on the edge of the primary beam give rise to ripples in the centre of the field of view
- The primary beam is spectrally dependent, so image subtraction should include such corrections and be performed in full spectral-line mode
- Pointing errors introduce gain and phase changes on the edge of the primary beam. If severe, the apparent source structure may change – attempt multiple snapshot subtraction on timescales comparable with pointing error change

CONFUSION

So how do we deal with these sources?

1. Outlier fields (the CASA default option) - deconvolve the confusing source while imaging the field of interest

2. Peeling - self-cal. on confusing source (to remove phase errors), get model & subtract source. Return to original calibration & insert model into visibilities

3. Direction-dependent calibration - see Advanced Imaging lecture

These are listed in order of complexity - note that direction dependent calibration is not available for all telescope arrays

CONFUSION

1. Outlier fields

If the source is out of your desired target area, then you can set a small area around the confusing source and deconvolve with the main image.

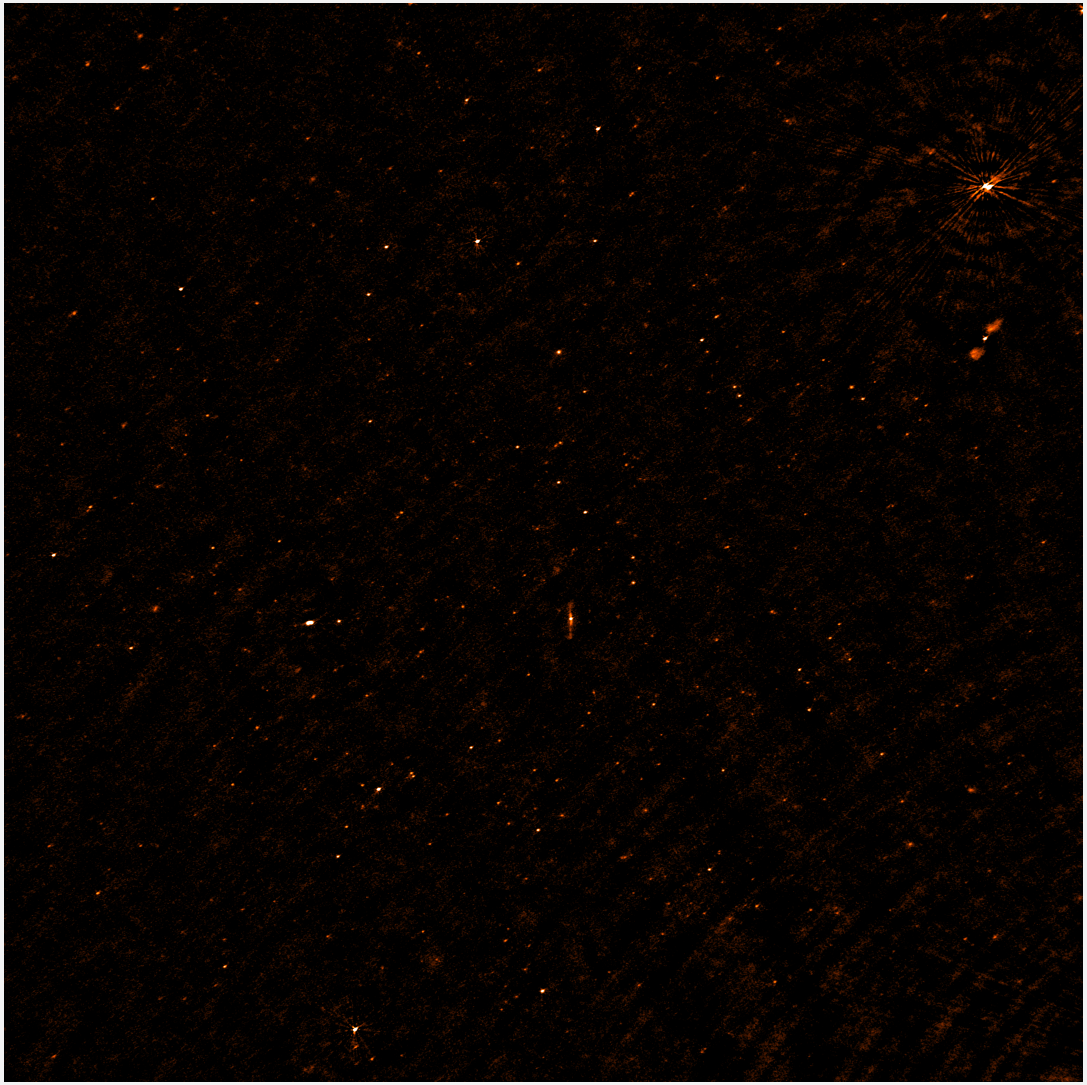
In CASA, this is achieved by setting multiple images (see right) or set an outlier file (orange box & example below)

```
#  clean :: Invert and deconvolve images with selected algorithm
vis                 = 'JVLA_combined_GOODSN.ms' #  Name of input visibility file
imagename           = ['main', 'outlier'] #  Pre-name of output images
outlierfile         =         ''          #  Text file with image names, sizes, centers for
                                          #   outliers
field               =         ''          #  Field Name or id
spw                 =         ''          #  Spectral windows e.g. '0~3', '' is all
selectdata          =       True          #  Other data selection parameters
     timerange      =         ''          #  Range of time to select from data
     uvrange        =         ''          #  Select data within uvrange
     antenna        =         ''          #  Select data based on antenna/baseline
     scan           =         ''          #  Scan number range
     observation    =         ''          #  Observation ID range
     intent         =         ''          #  Scan Intent(s)

mode                =      'mfs'          #  Spectral gridding type (mfs, channel, velocity,
                                          #   frequency)
     nterms         =          1          #  Number of Taylor coefficients to model the sky
                                          #   frequency dependence
     reffreq        =         ''          #  Reference frequency (nterms > 1),'' uses central
                                          #   data-frequency

gridmode            =         ''          #  Gridding kernel for FFT-based transforms, default=''
                                          #   None
niter               =        500          #  Maximum number of iterations
gain                =        0.1          #  Loop gain for cleaning
threshold           =   '0.0mJy'          #  Flux level to stop cleaning, must include units:
                                          #   '1.0mJy'
psfmode             =    'clark'          #  Method of PSF calculation to use during minor cycles
imagermode          =  'csclean'          #  Options: 'csclean' or 'mosaic', '', uses psfmode
     cyclefactor    =        1.5          #  Controls how often major cycles are done. (e.g. 5
                                          #   for frequently)
     cyclespeedup   =         -1          #  Cycle threshold doubles in this number of iterations

multiscale          =         []          #  Deconvolution scales (pixels); [] = standard clean
interactive         =      False          #  Use interactive clean (with GUI viewer)
mask                =         []          #  Cleanbox(es), mask image(s), region(s), or a level
imsize              = [[8000, 8000], [50, 50]] #  x and y image size in pixels.  Single value:
                                          #   same for both
cell                = ['0.33arcsec']      #  x and y cell size(s). Default unit arcsec.
phasecenter         = ['J2000 12h36m49.4 62d12m58.0', 'J2000 12h34m52.2 62d02m34.53'] #  Image
```

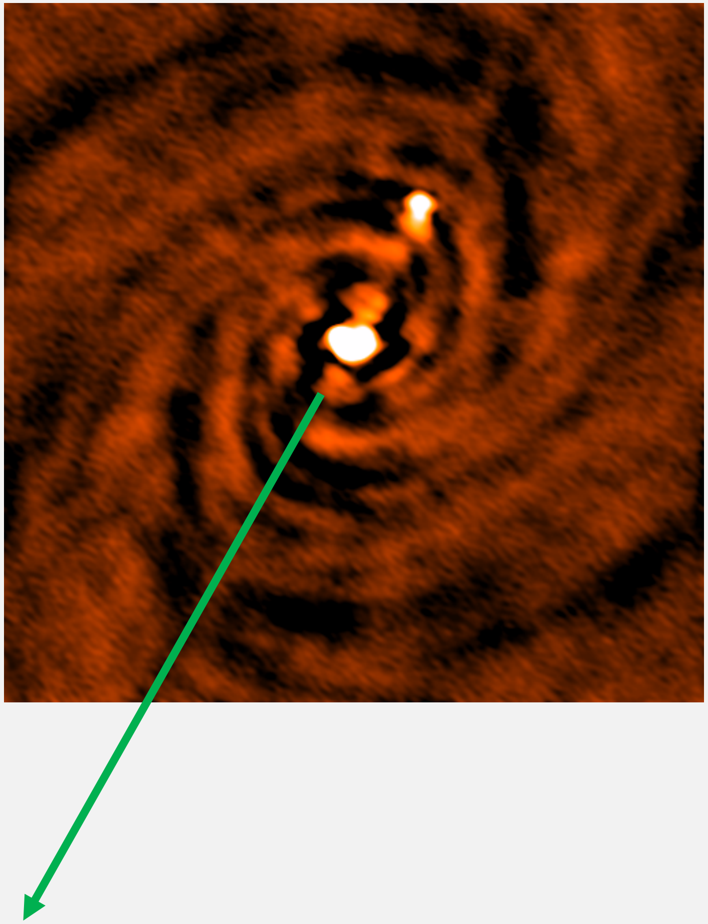
```
#content of outliers.txt
#
#outlier field1
imagename='outlier1'
imsize=[512,512]
phasecenter = 'J2000 12h34m52.2 62d02m34.53'
mask='box[[245pix,245pix],[265pix,265pix]]'
```

CONFUSION

1. Outlier fields



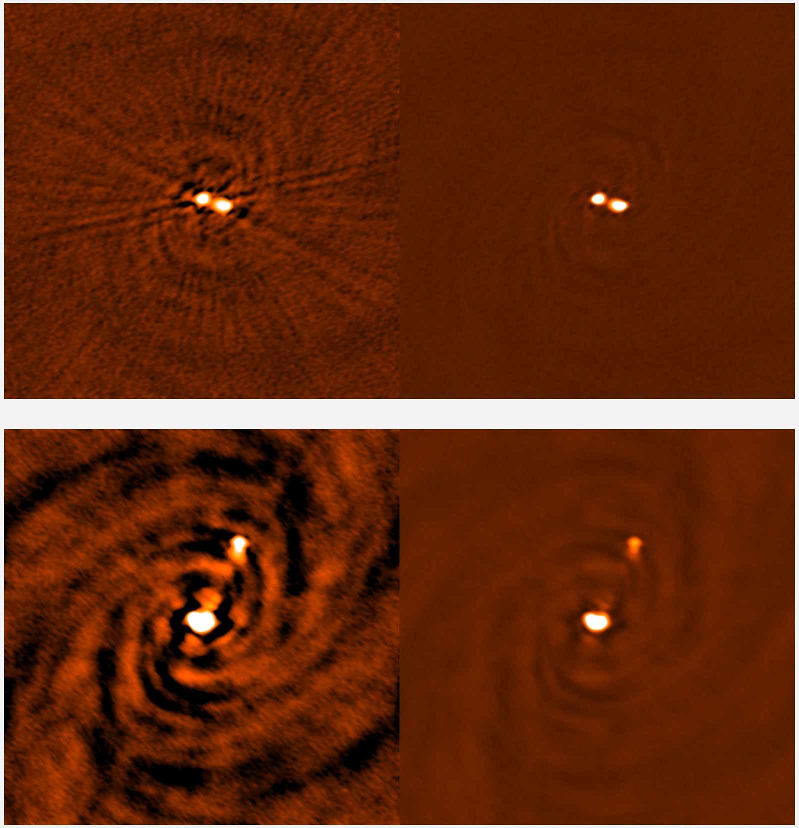
0.25 Jy confusing source using outlier field assigned



CONFUSION

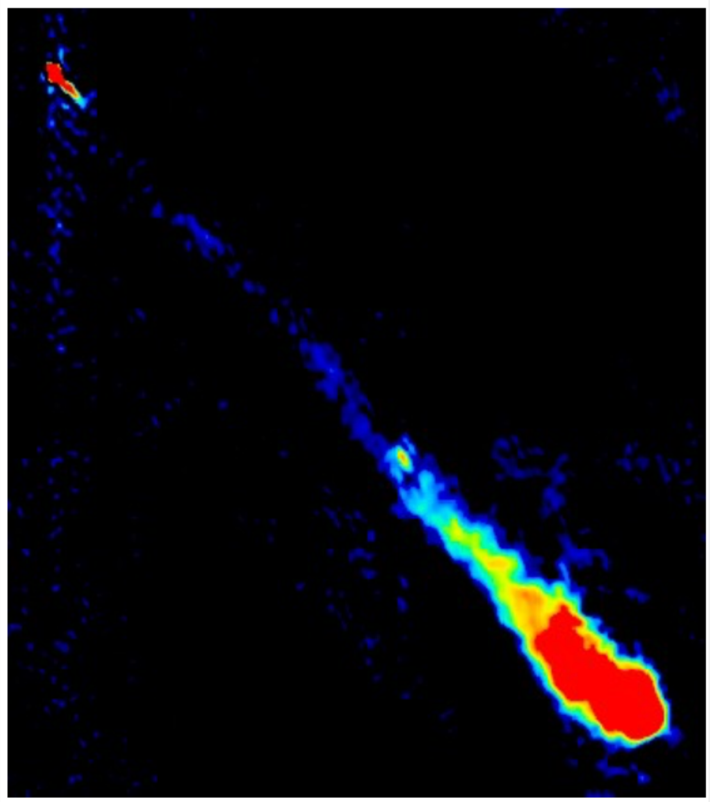
2. Peeling If outlier fields do not work try peeling!

- After phase calibrating the data, perform self-calibration for the brightest confusing source – then subtract it out
- Delete phase solutions derived for previous confusing source (1)
- Move to next brightest confusing source, perform self-calibration/imaging cycles – then subtract that source from the dataset (2)
- Perform (1) and (2) until all confusing sources are subtracted. Delete all self-calibration solutions and image central regions



HIGH DYNAMIC RANGE IMAGING

- Present dynamic range limits (on axis):
  - Phase calibration – up to 1000:1 ! improve with self-calibration
  - Non-closing data errors – continuum ~20,000:1, line >100,000:1
  - After non-closing error correction ~10,000,000:1
- Non-closing errors thought to be dominated by small changes in telescope passbands.
- Spectral line data configurations are the default for all new wide-band radio telescopes.
- In order to subtract out confusion we will need to be able to image with these very high dynamic ranges away from the beam centre



3C273, Davis et al. (MERLIN)
1,000,000:1 peak – RMS

Credit T. Muxlow

SIGNAL TO NOISE

Noise level of a (perfect) homogeneous interferometer:

$$\text{Noise} = \frac{\sqrt{2} k_B T_{\text{sys}}}{\sqrt{n_b t \Delta\nu} A \eta}$$

where:

T_{sys} - system temperature [K]
n_b - number of baselines
t - integration time [s]
$\Delta\nu$ - bandwidth [Hz]
A - area of apertures [m]
η - aperture efficiency

Many factors increase noise level above this value:

- Confusion
- Calibration errors
- Bad data
- Non-closing data errors
- Deconvolution artefacts

Rarely get this from an image. Dependent of flagging accuracy, calibration & adequate deconvolution

But techniques presented in this workshop can get you closer!