Introduction to imaging*

Cristiana Spingola

INAF Istituto di Radioastronomia (Bologna, Italy)

INAF ISTITUTO NAZIONALE DI ASTROFISICA

OPTICON RadioNet Pilot

*Strongly based on previous CASA-VLBI workshops and ERIS schools -Thanks to all previous lecturers!

CASA VLBI Worskhop 2023, JIVE June 5-9, 2023

This event has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 101004719.

Intro

The output of an interferometer is basically a table of the correlation (amp. & phase) measured on each baseline every few seconds.

To get the final image out of our visibilities the steps are:

Calibration and data editing (all lectures so far)

Deconvolution = making a CLEANed image and a model of your source (this lecture!)

Refining calibration = self-calibration (next lecture)

This lecture + material

Deconvolution = making a CLEANed image and a model of your source

1) For experimenting with EVN data and for learning how to script CASA commands,

Follow this:
https://www.jb.man.ac.uk/DARA/unit4/Workshops/EVN_continuum.html

by Jack Radcliffe, Anita Richards and Des Small

2) Test data associated to this lecture are from VLBA, to have an alternative dataset

Imaging intro: Fourier Transform imaging and sampling function

B = Intrinsic source brightness distribution

Convolution

D = dirty beam = point spread function (PSF)

S = Sampling function

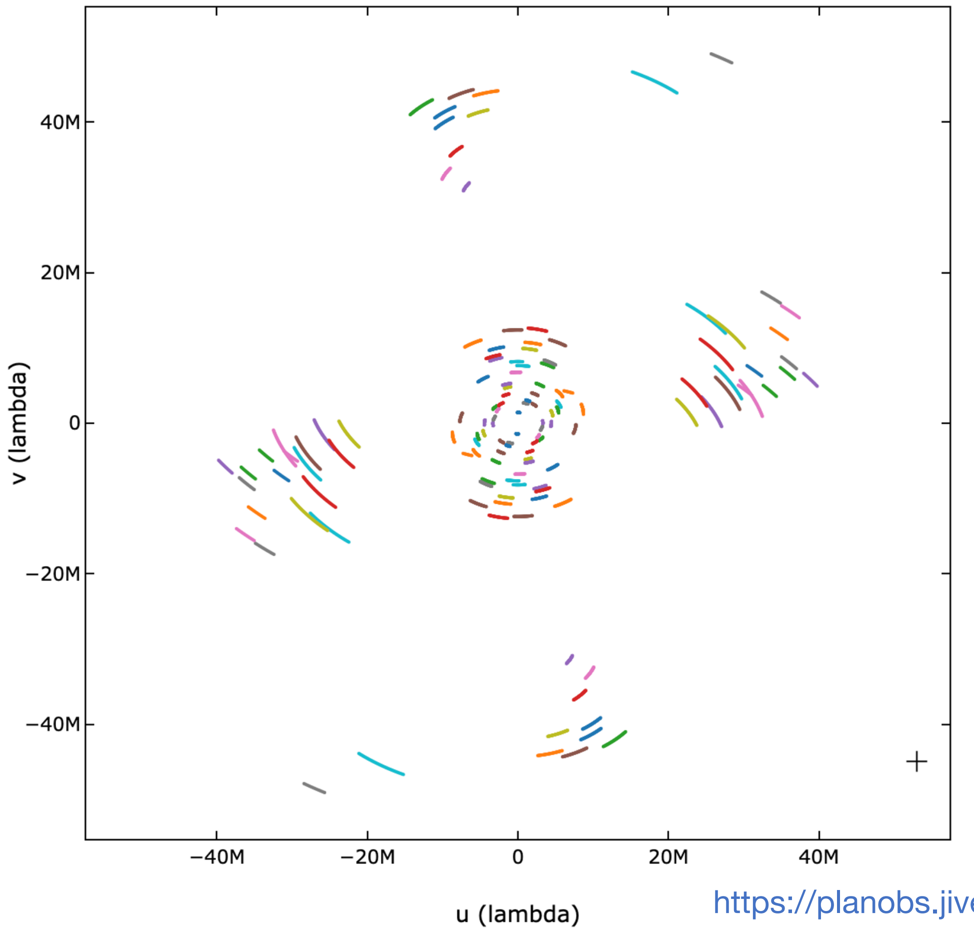
Observed visibilities = $B(l,m) * D(l,m) \approx \iint_{uv} S(u,v)V(u,v)e^{2\pi i(ul+vm)}dudv$

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

Dirty beam D(l,m) = Fourier transform of the sampling function
We know D(l,m) !!!

We need to **deconvolve** B(l,m) from the dirty beam D(l,m)

S = sampling function
= 1 where there is a measurement in the uv plane
= 0 otherwise



1 h EVN

https://planobs.jive.eu/

Imaging intro: Fourier Transform imaging and sampling function

B = Intrinsic source brightness distribution

Convolution

D = dirty beam = point spread function (PSF)

S = Sampling function

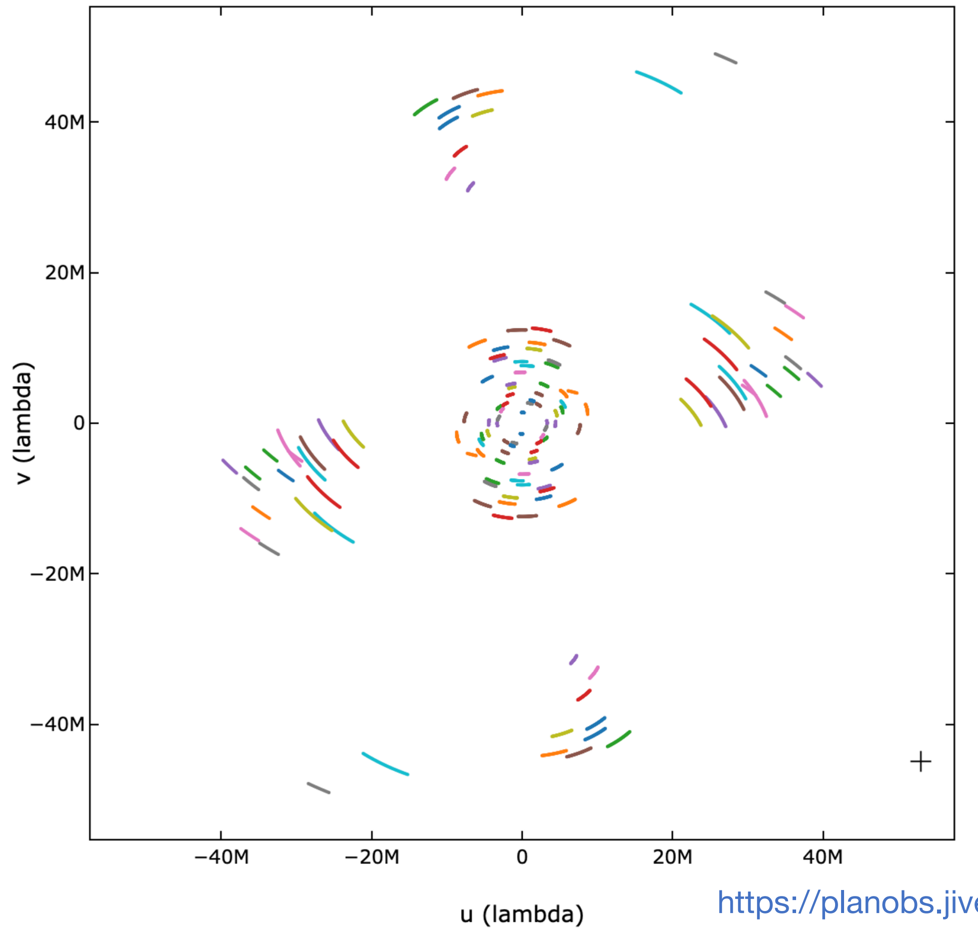
Observed visibilities = $B(l,m) * D(l,m) \approx \iint_{uv} S(u,v)V(u,v)e^{2\pi i(ul+vm)}dudv$

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

S = sampling function

= 1 where there is a measurement in the uv plane

= 0 otherwise



1 h EVN

https://planobs.jive.eu/

An ideal interferometer would deliver
on a regularly highly sampled rectangular grid.
An image of would then be made by simply applying
a Fourier transform

But, arrays provide (poorly) sampled Fourier Transform of the radio brightness region of sky

You need as many V(u,v) points as possible to reconstruct as robustly as possible the surface brightness distribution of the source

Imaging intro: Fourier Transform imaging and sampling function

B = Intrinsic source brightness distribution

Convolution

D = dirty beam = point spread function (PSF)

S = Sampling function

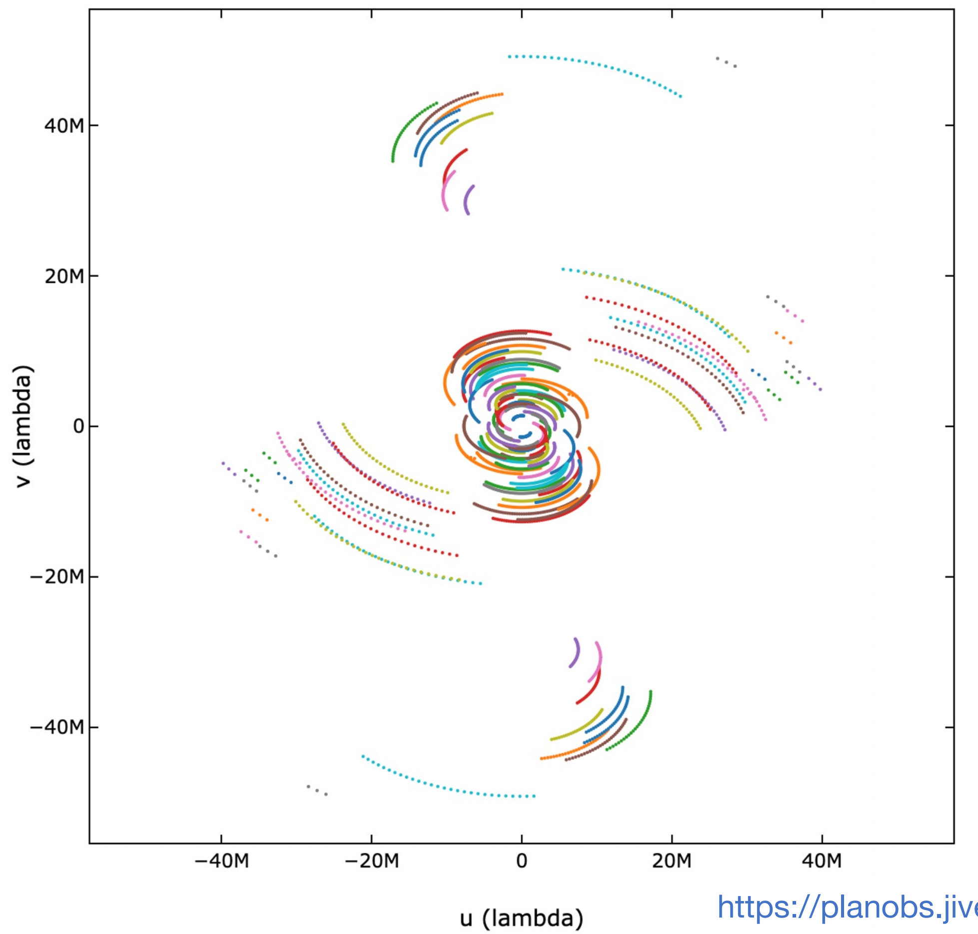
Observed visibilities $= \mathrm{B}(l,m) * \mathrm{D}(l,m) \approx \iint_{uv} \mathrm{S}(u,v)\mathrm{V}(u,v)e^{2\pi i(ul+vm)}dudv$

$$\mathrm{D}(l,m) = \iint_{uv} \mathrm{S}(u,v)e^{2\pi i(ul+vm)}dudv$$

S = sampling function

= 1 where there is a measurement in the uv plane

= 0 otherwise



6 h EVN

https://planobs.jive.eu/

An ideal interferometer would deliver
on a regularly highly sampled rectangular grid.
An image of would then be made by simply applying
a Fourier transform

But, arrays provide (poorly) sampled Fourier Transform of the radio brightness region of sky

You need as many V(u,v) points as possible to reconstruct as robustly as possible the surface brightness distribution of the source

Imaging intro: Fourier Transform imaging and sampling function

S = Sampling function

B = Intrinsic source brightness distribution

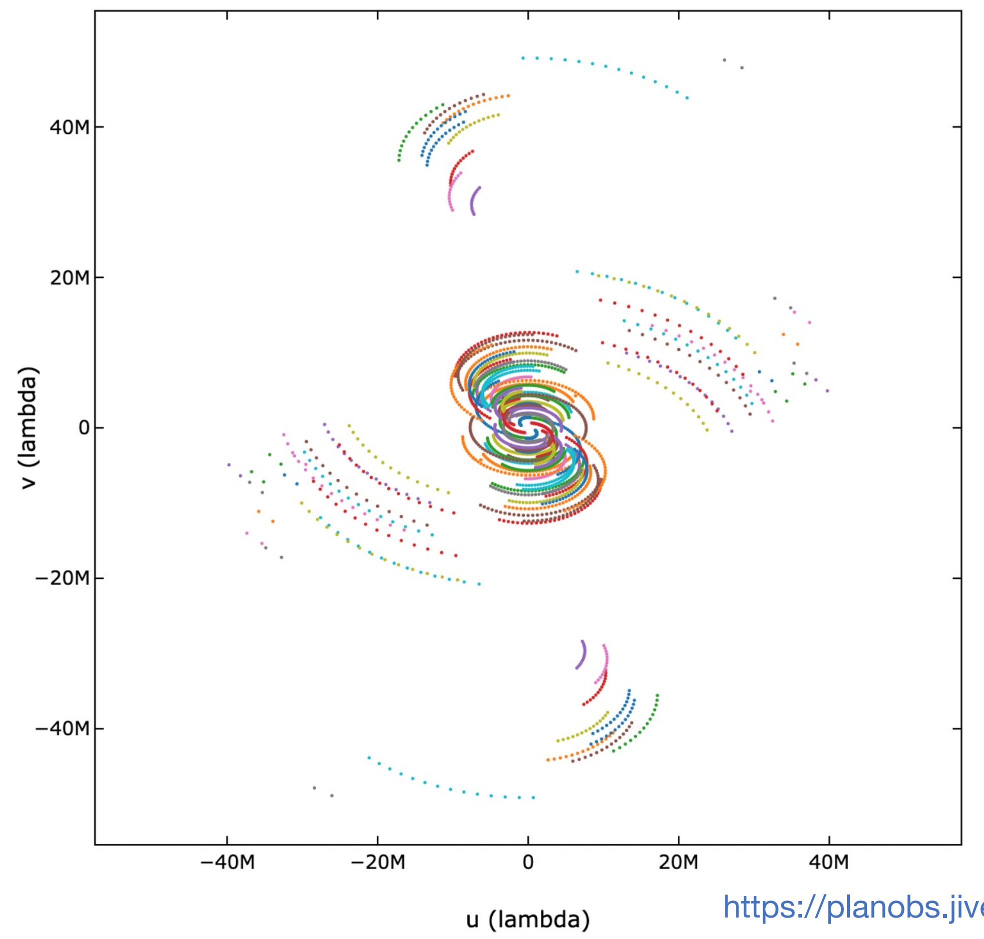
D = dirty beam = point spread function (PSF)

Convolution

$$\text{Observed visibilities} = B(l,m) * D(l,m) \approx \iint_{uv} S(u,v)V(u,v)e^{2\pi i(ul+vm)}dudv$$

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

S = sampling function
= 1 where there is a measurement in the uv plane
= 0 otherwise



12 h EVN

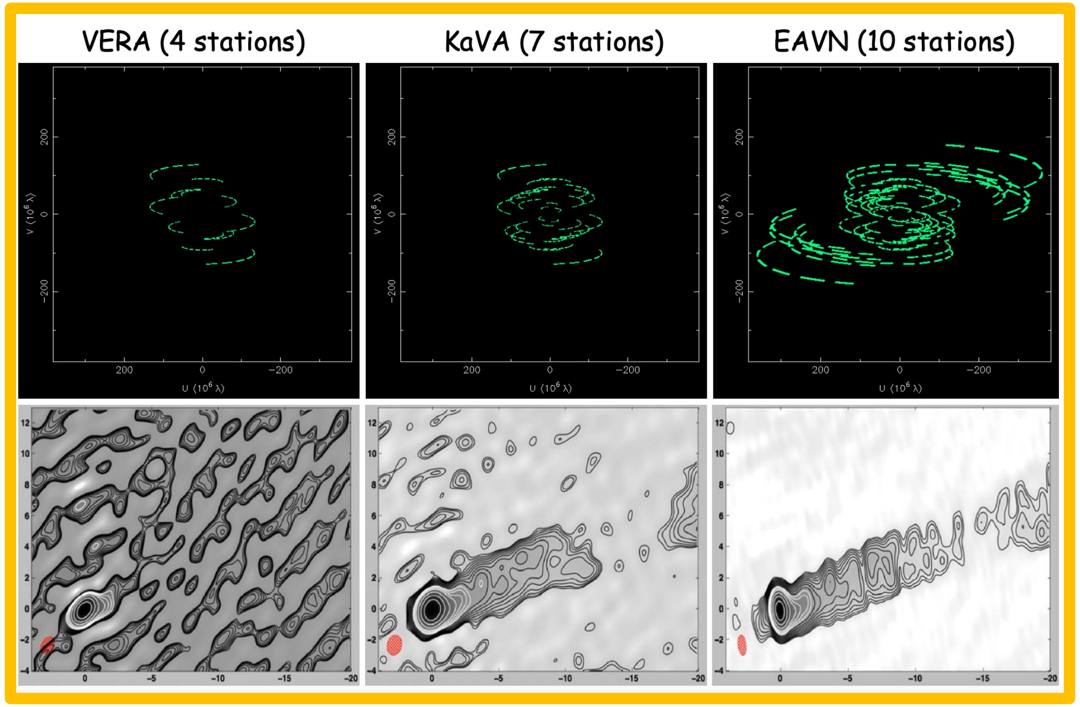
https://planobs.jive.eu/

An ideal interferometer would deliver
on a regularly highly sampled rectangular grid.
An image of would then be made by simply applying
a Fourier transform

But, arrays provide (poorly) sampled Fourier Transform of the radio brightness region of sky

You need as many V(u,v) points as possible to reconstruct as robustly as possible the surface brightness distribution of the source

Imaging intro: Fourier Transform imaging and sampling function



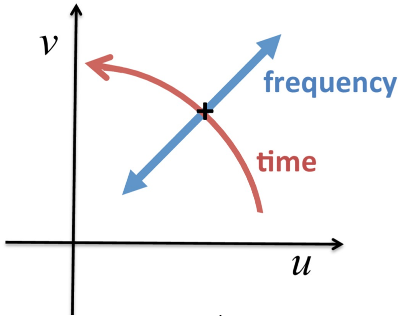
Credits: Prof. Kazuhiro Hada

An ideal interferometer = measurements on a regularly highly sampled rectangular grid.
An image of would then be made by simply applying a Fourier transform

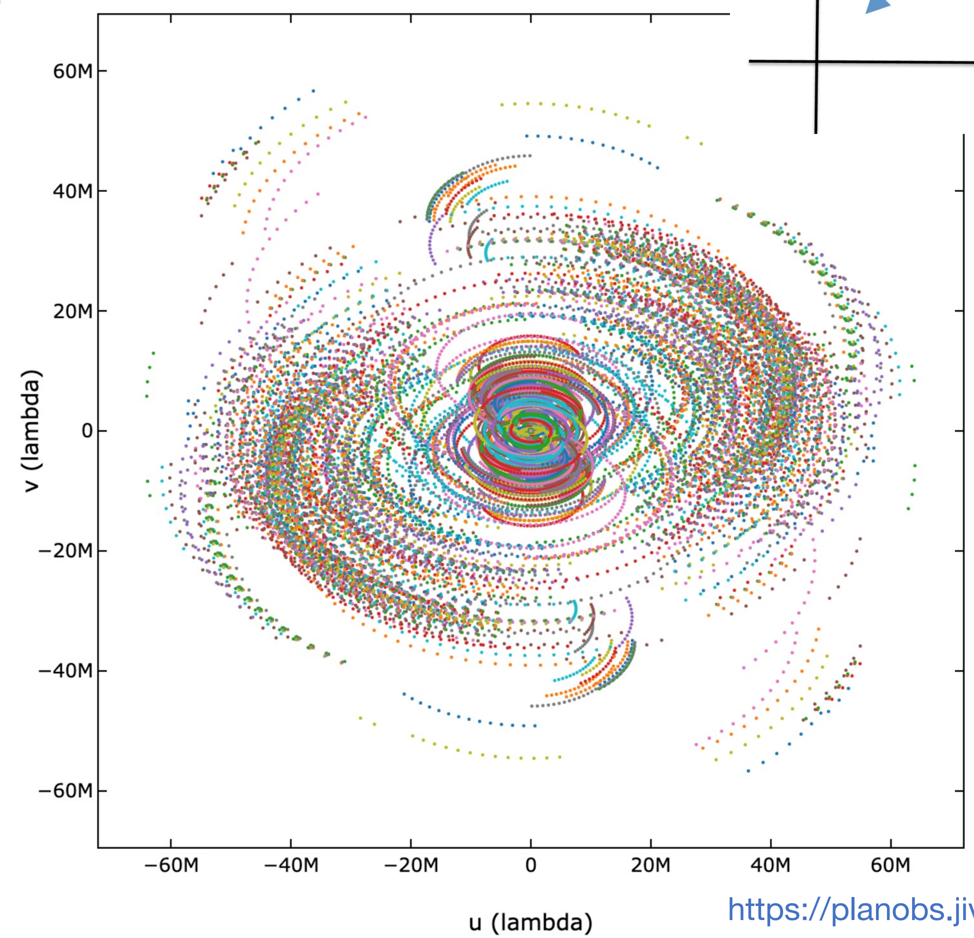
But, arrays provide (poorly) sampled Fourier Transform of the radio brightness region of sky

You need as many V(u,v) points as possible to reconstruct as robustly as possible the surface brightness distribution of the source

S = sampling function
= 1 where there is a measurement in the uv plane
= 0 otherwise



Credits: Prof. Scaife ERIS 2017



12 h
>30 antennas

https://planobs.jive.eu/

Gridding

But, arrays provide (poorly) sampled Fourier Transform of the radio brightness region of sky
AND
There will always be gaps in the u-v plane!

Two approaches

Direct Fourier Transform (DFT) = FT evaluated at every point of a rectangular grid – $O(N^2)$ operations

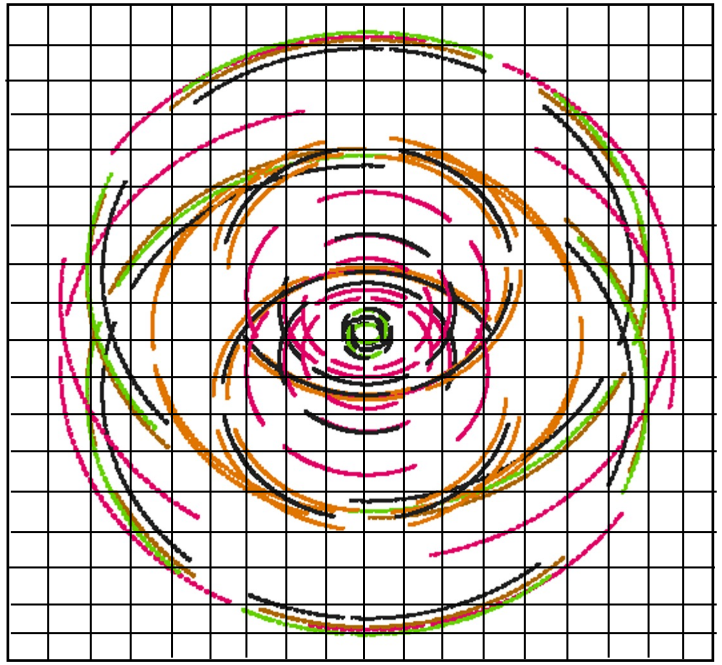
impractical for large number of visibilities

Fast Fourier Transform (FFT) = interpolate the data onto a rectangular grid – O(N log N) operations

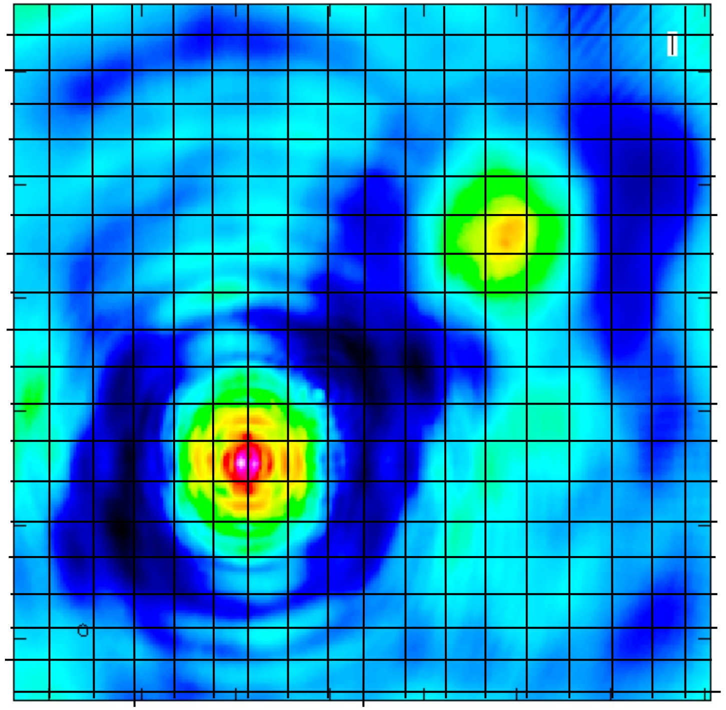
It saves a lot of computing time!!

This FFT method requires the observed visibilities to be interpolated on a regular grid

Usually we define the grid in the image plane, where grid spacing = cell size



NxN grid

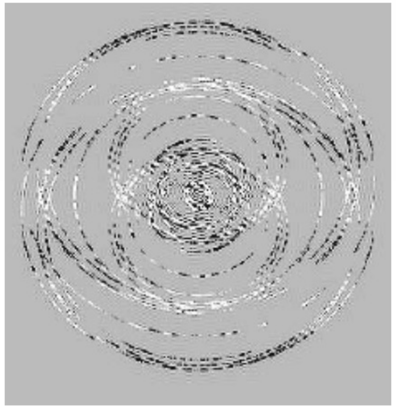


The need for deconvolution

«Visibility domain»

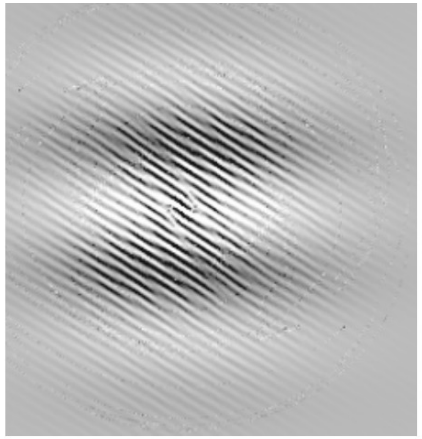
FT

Sampled visibilities V'(u,v)



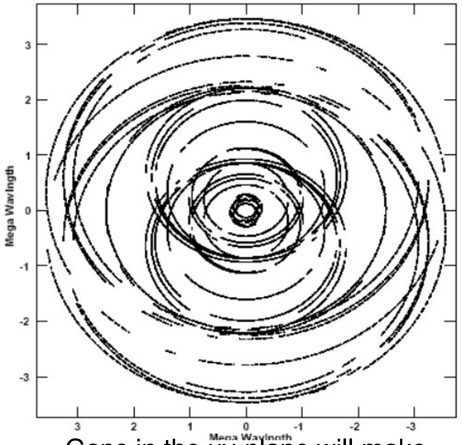
=

True visibilities V(u,v)



X

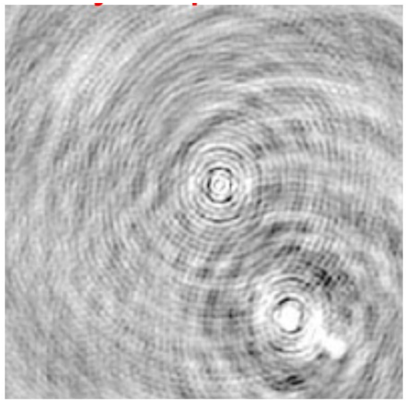
Sampling function S(u,v)



Gaps in the uv-plane will make sidelobes in the dirty beam

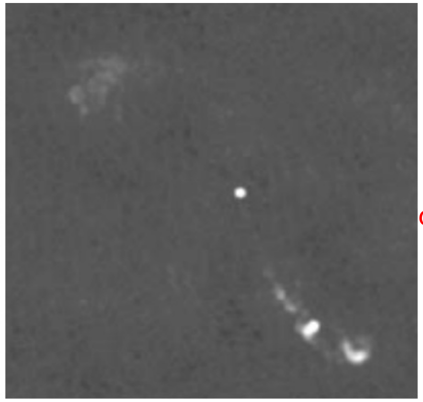
«Image domain»

Dirty image B' (l,m)



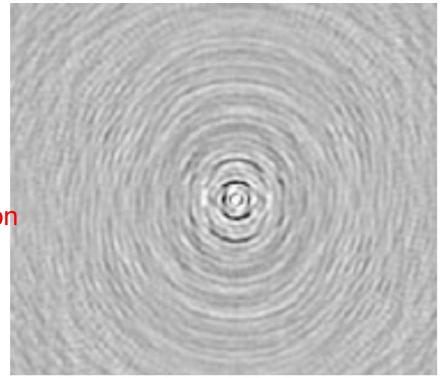
=

True sky B (l,m)



*
convolution

Dirty beam (l,m)



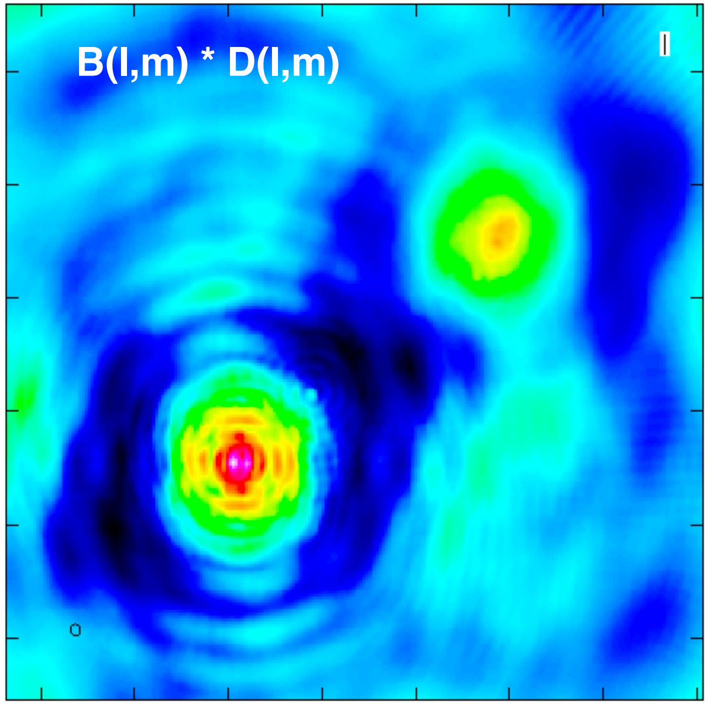
The dirty image is not the true image of the source, since the sampled visibilities are not the true visibilities

Corrections of the effect of Fourier sampling deficiencies on the dirty image = CLEAN algorithms

Original slide from
Prof. Garrington ERIS 2017

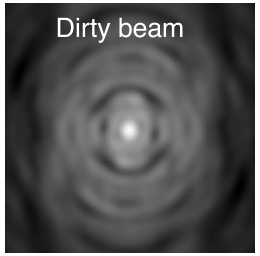
Deconvolution

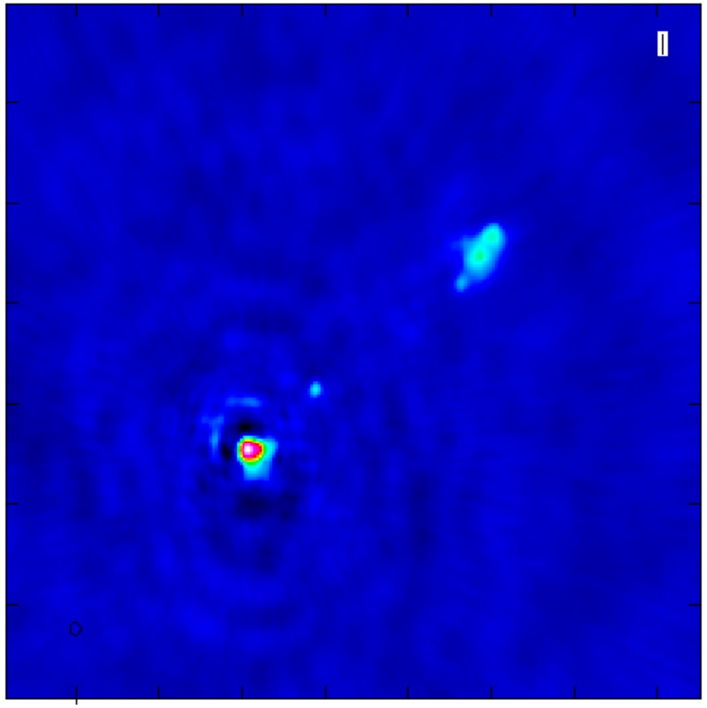
….Why do we need all of this again?



From «dirty image»

Deconvolve the intrinsic source brightness distribution B(l,m) from the **dirty beam** D(l,m)





To «CLEAN image»

https://www.jb.man.ac.uk/DARA/unit4/Workshops/EVN_continuum.html.

Complete tutorial using EVN data by Jack Radcliffe, Anita Richards and Des Small

Deconvolution

Since only a finite number of (noisy) samples are measured, to recover B(l,m) we need **some stable non-linear approach** + ***a priori*** **information**:

- B(l,m) must be positive
- Radio sources do not resemble the dirty beam (i.e. sidelobes-like patterns)
- Sky is basically empty with just a few localised sources

B = Intrinsic source brightness distribution

Convolution

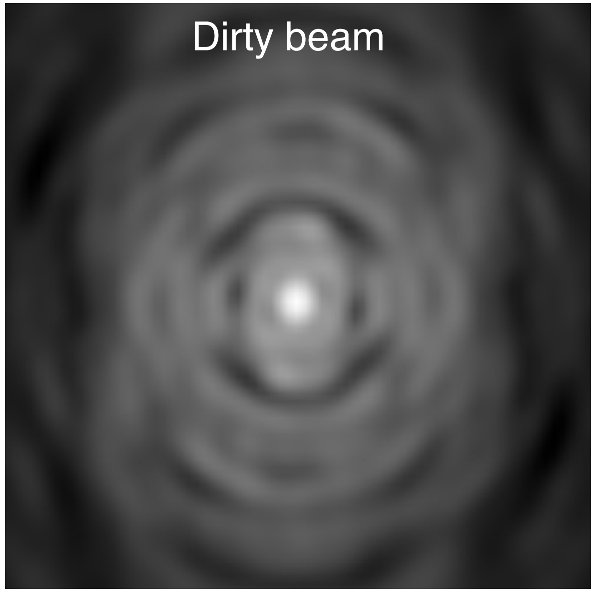
D = dirty beam = point spread function (PSF)

S = Sampling function

$$B(l,m) * D(l,m) \approx \iint_{uv} S(u,v)V(u,v)e^{2\pi i(ul+vm)}dudv$$

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

We know this!
To recover B we have **“just" to deconvolve** the D(l,m) term



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

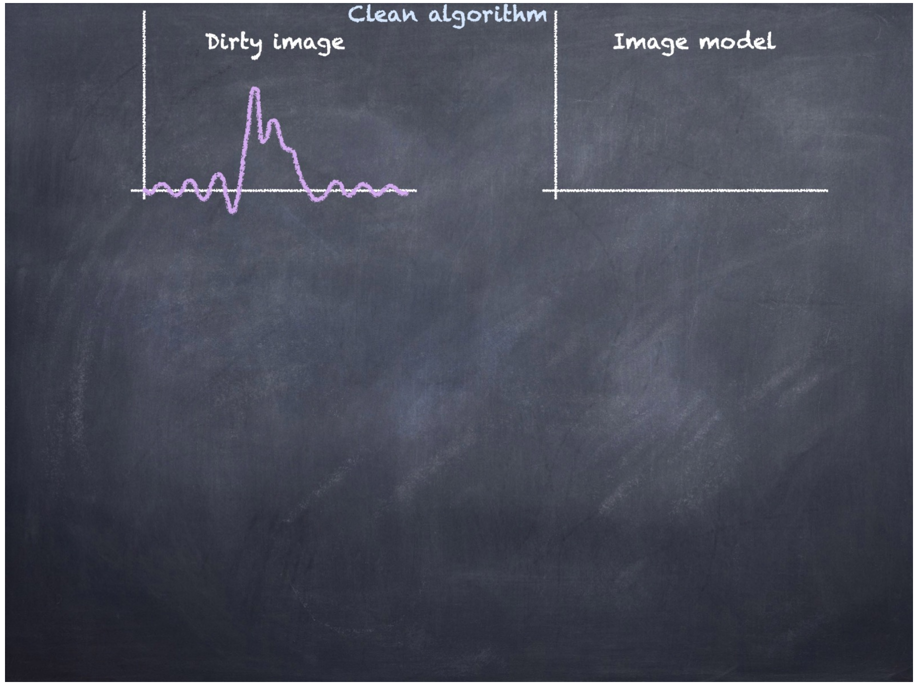
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

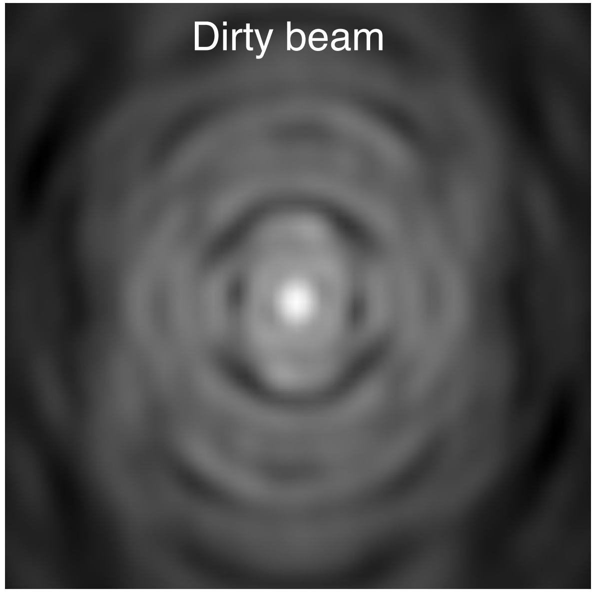
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

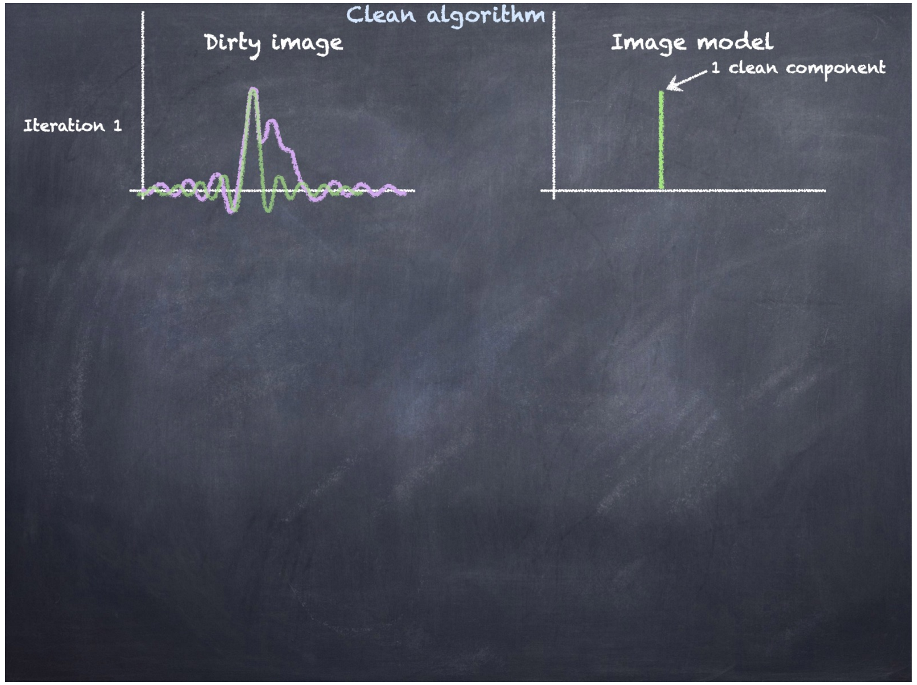
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

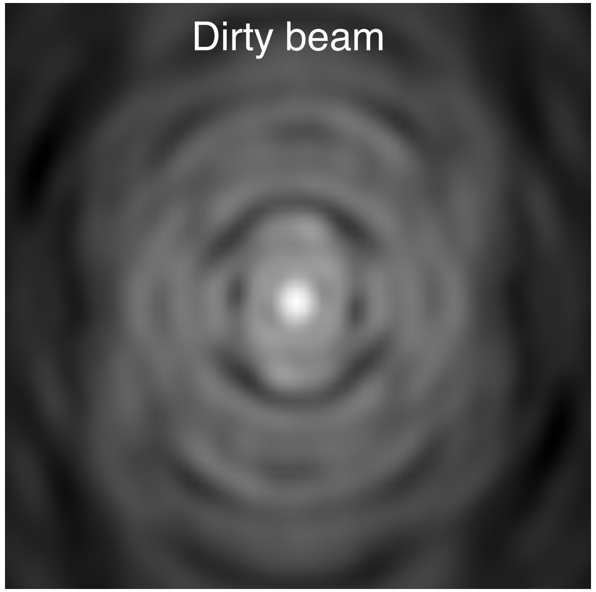
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

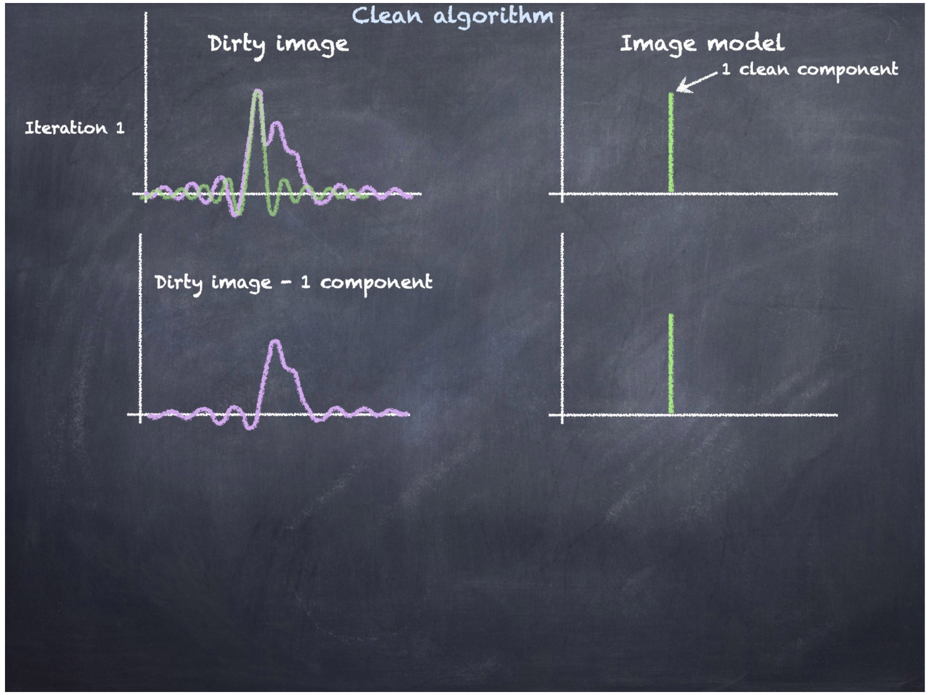
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

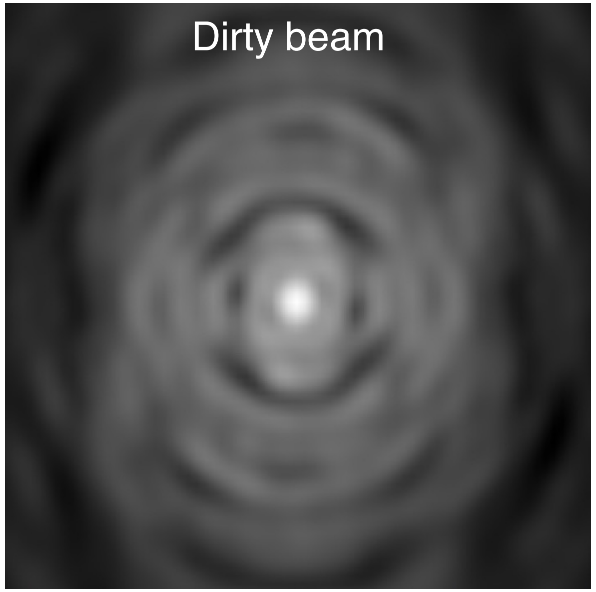
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

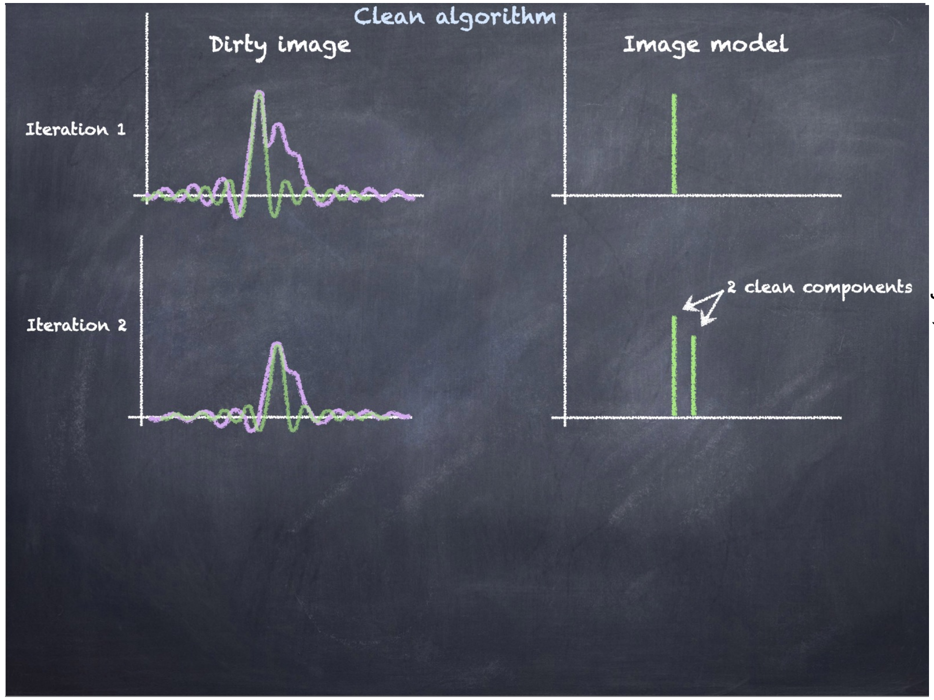
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

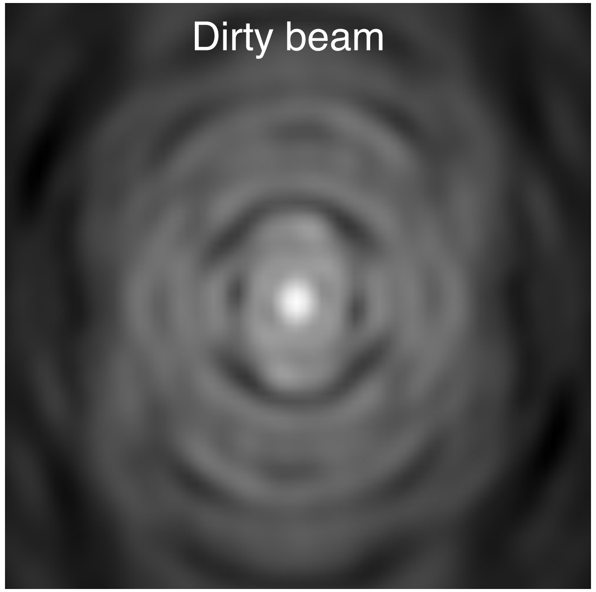
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (data - model)

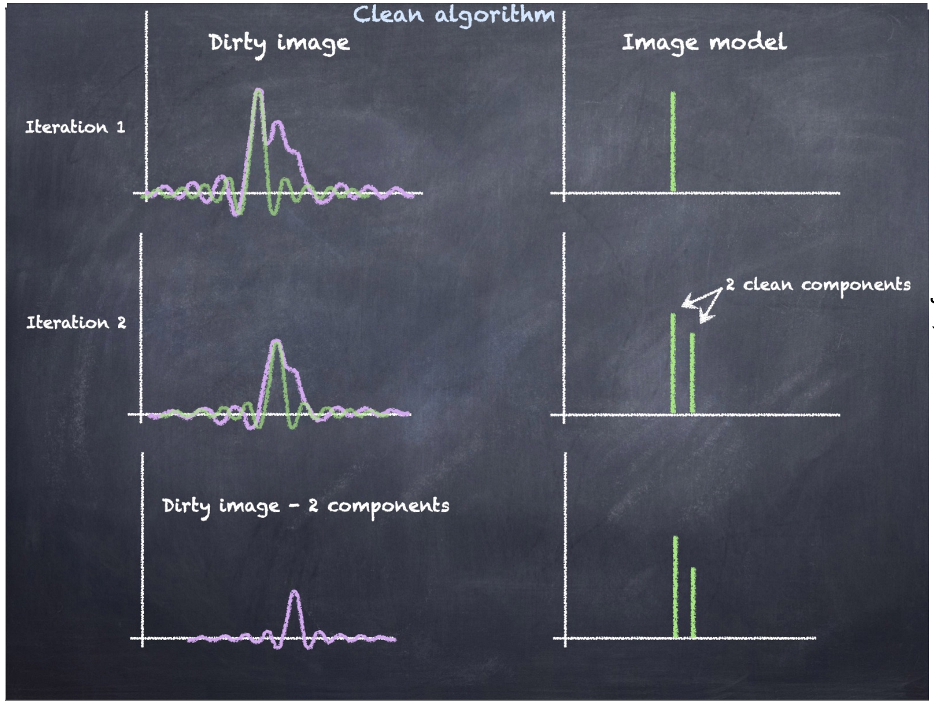
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

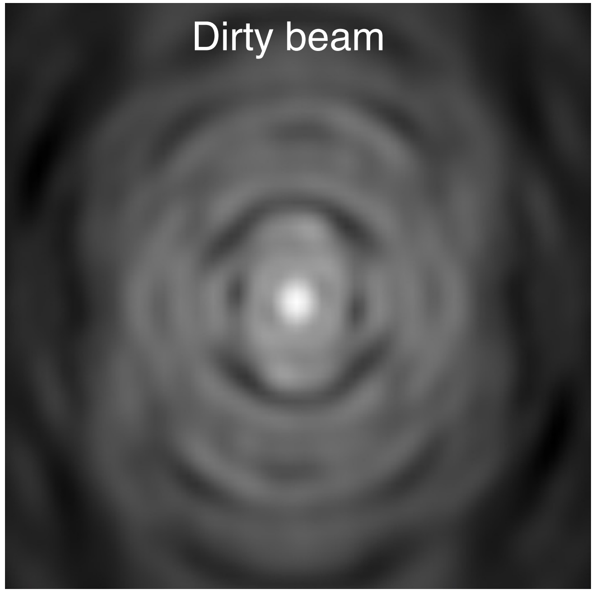
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

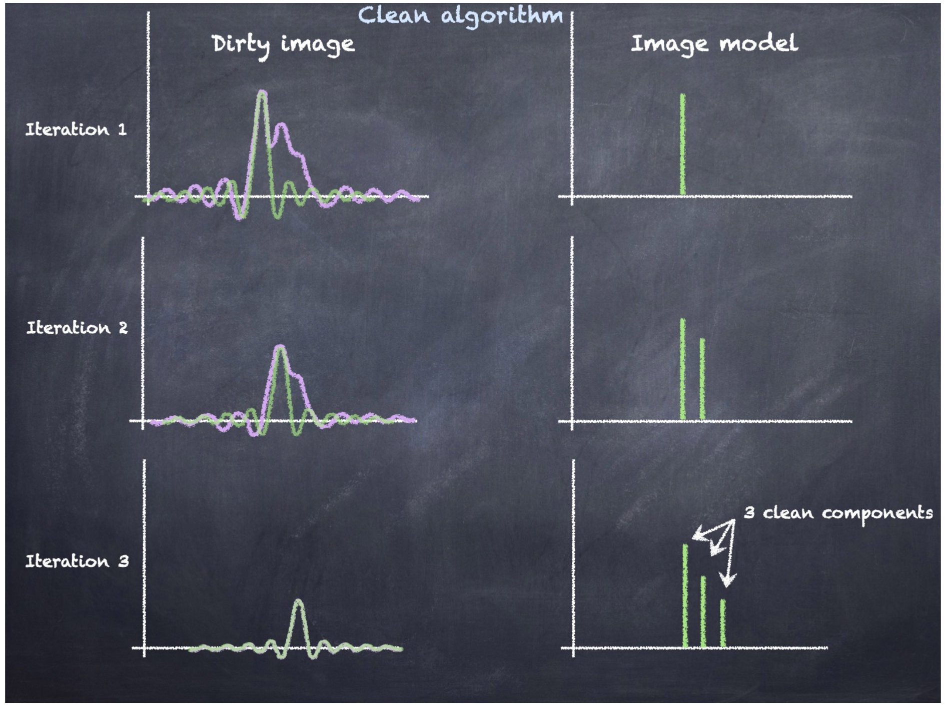
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

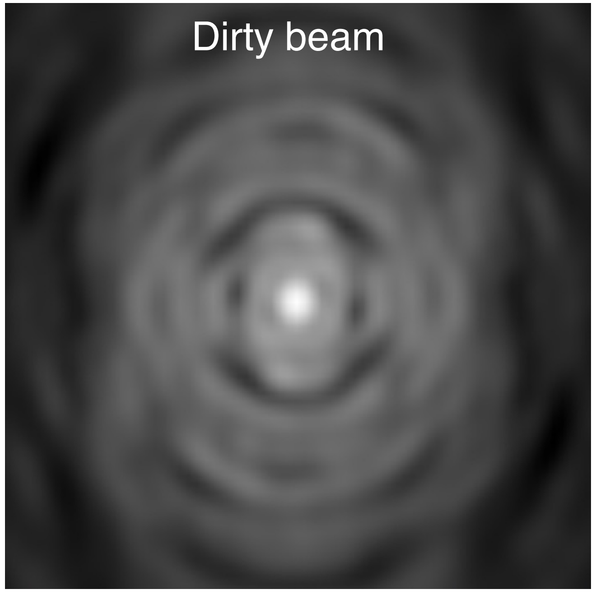
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

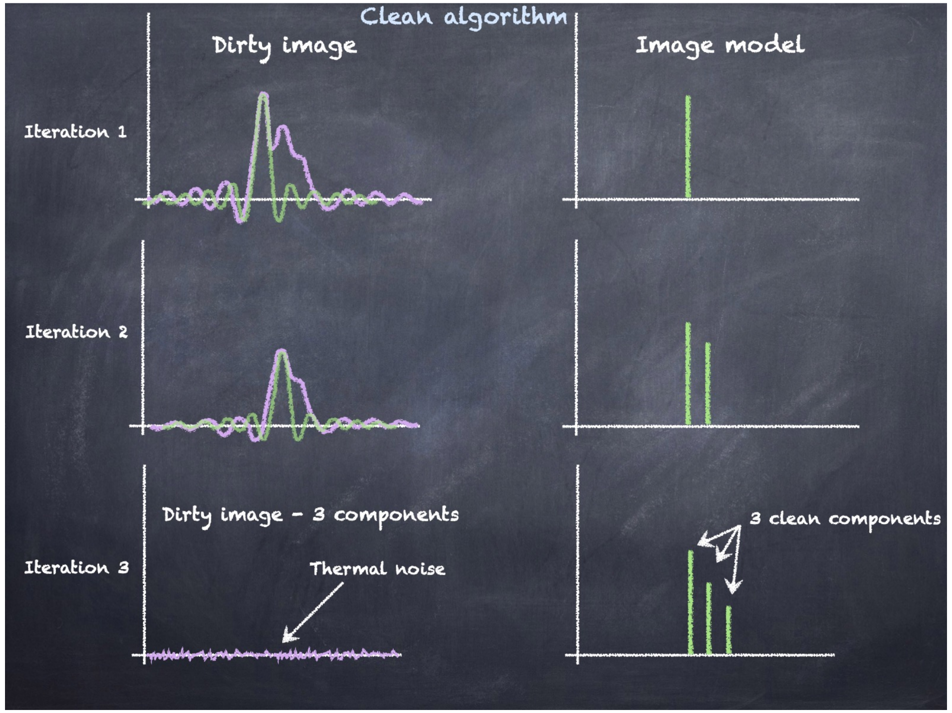
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

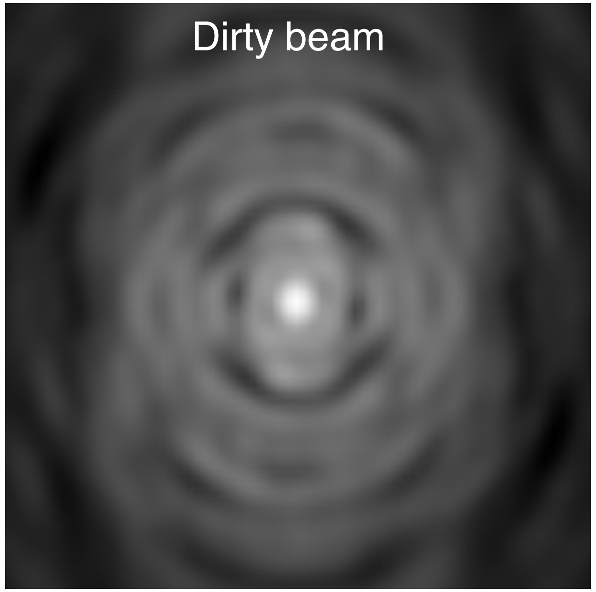
4) Go to 1) unless you reach the **stopping criterion**

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

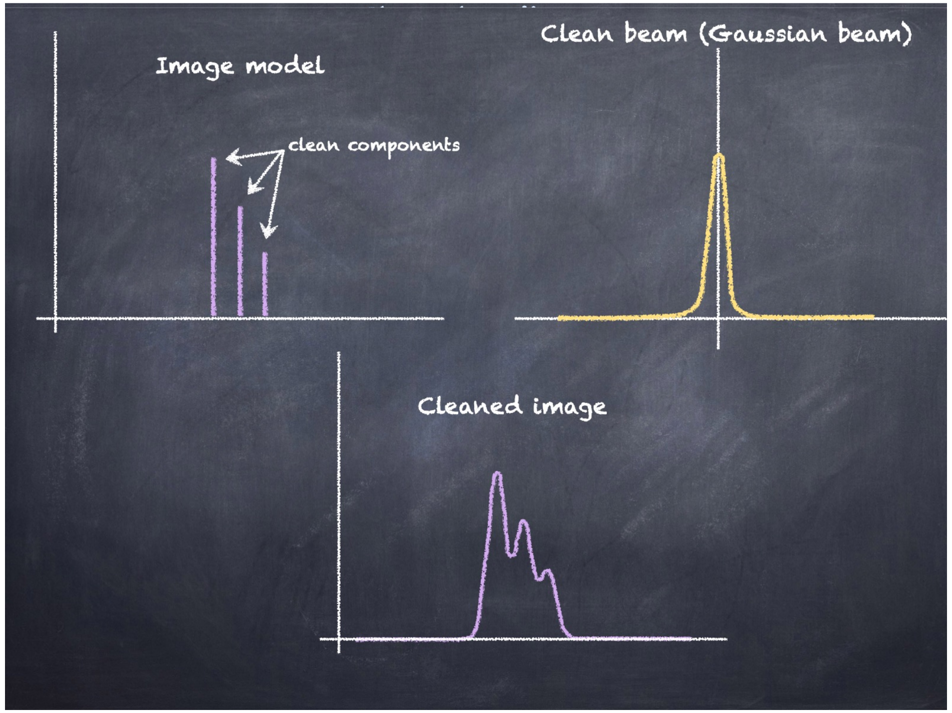
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

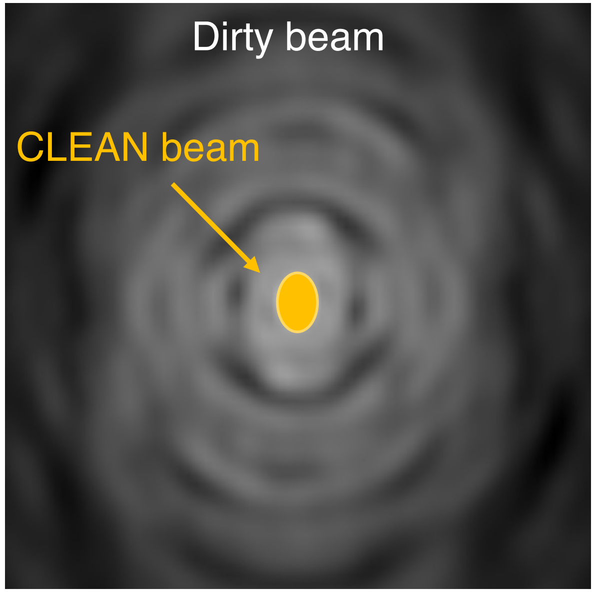
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

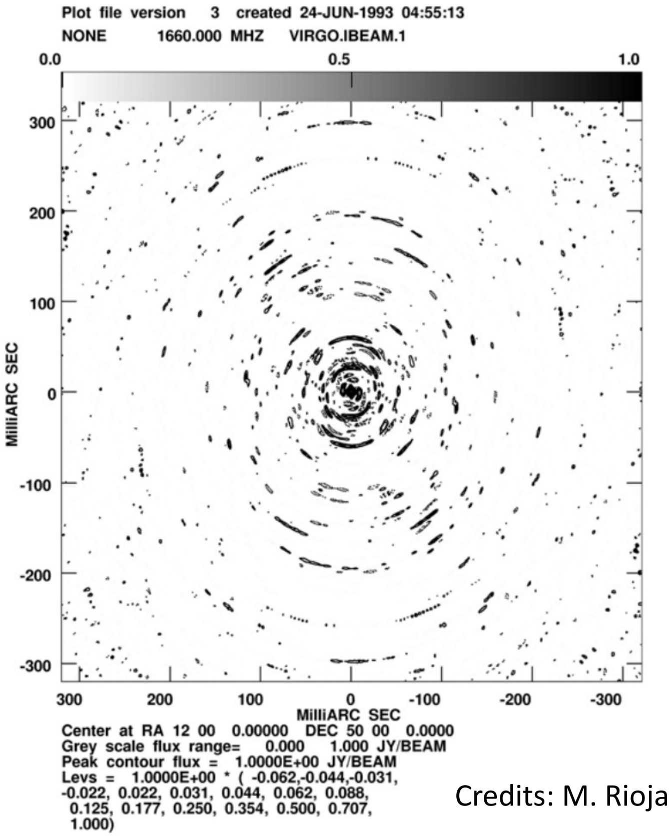
6) Add the residual of the dirty image to the CLEAN image



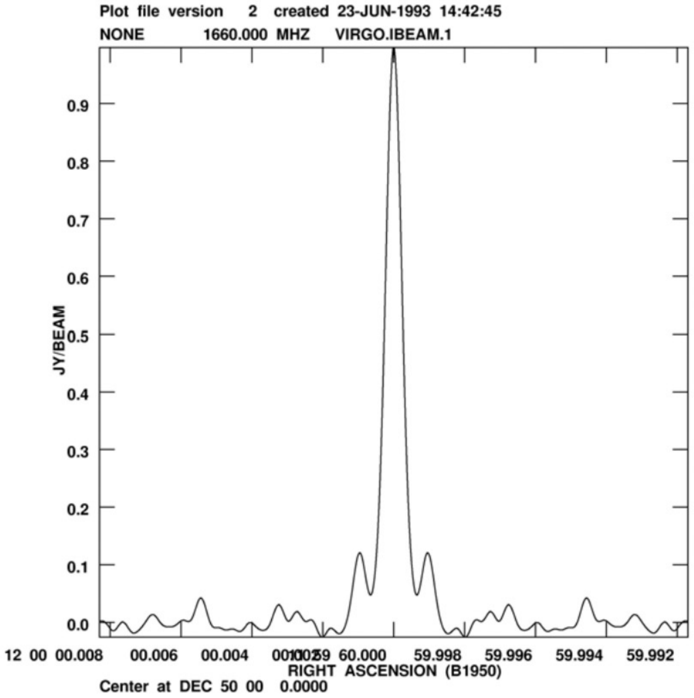
Credits: Daniel Tafoya, ERIS 2019

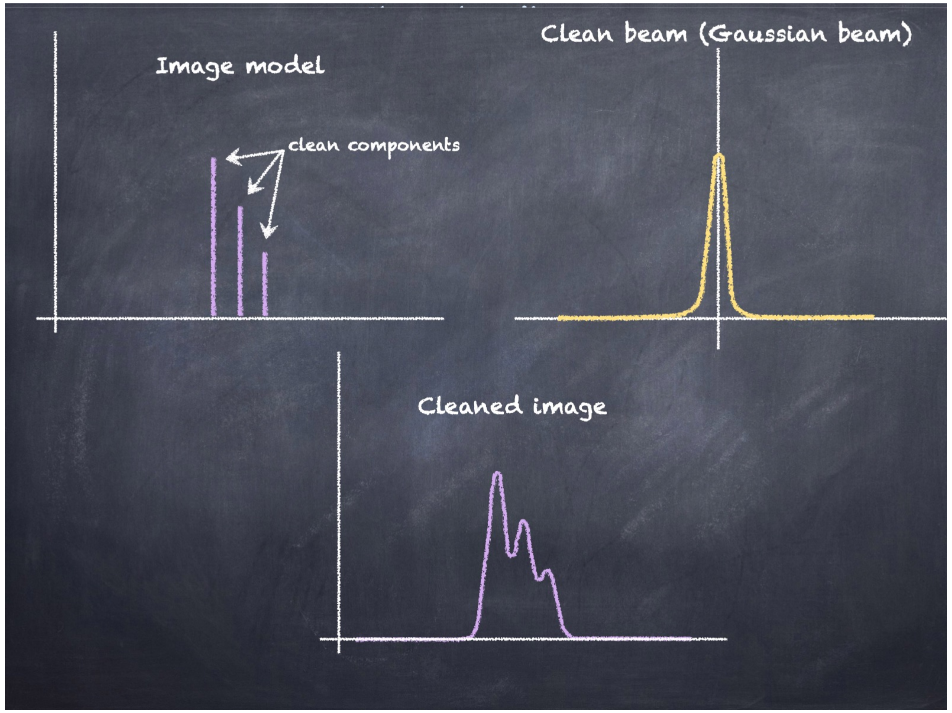


Deconvolution

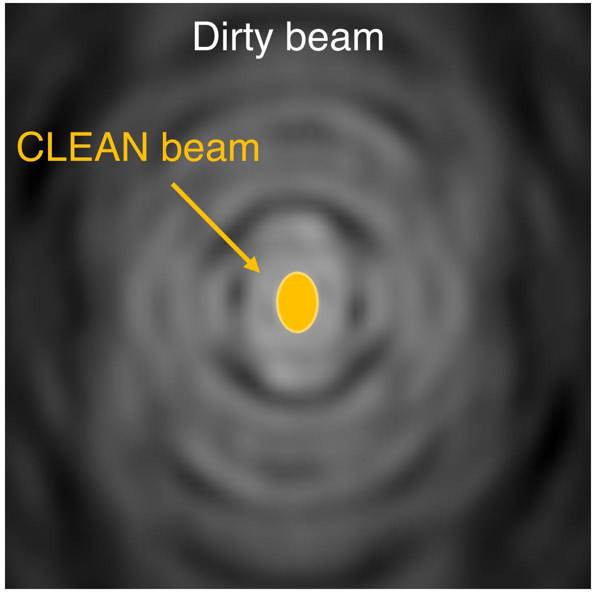


Credits: M. Rioja

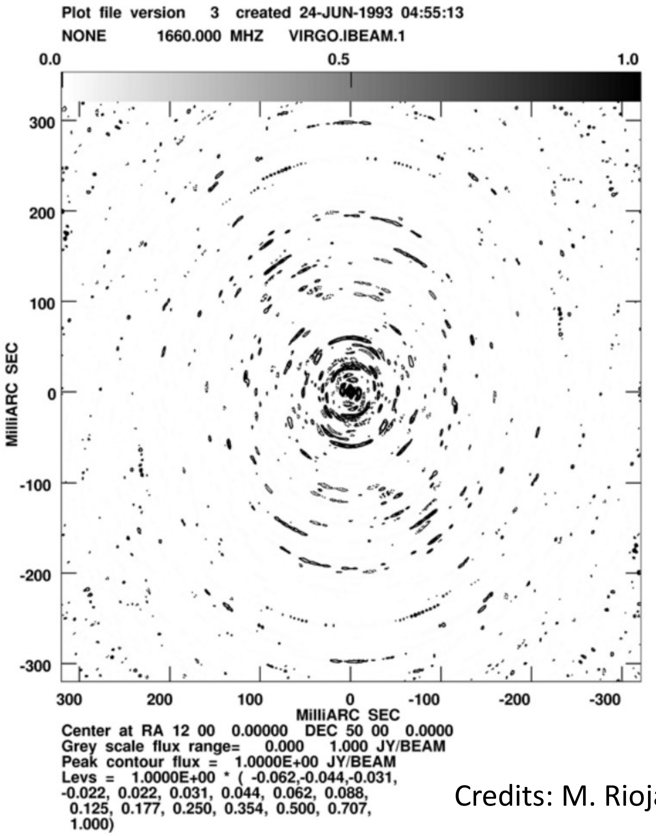




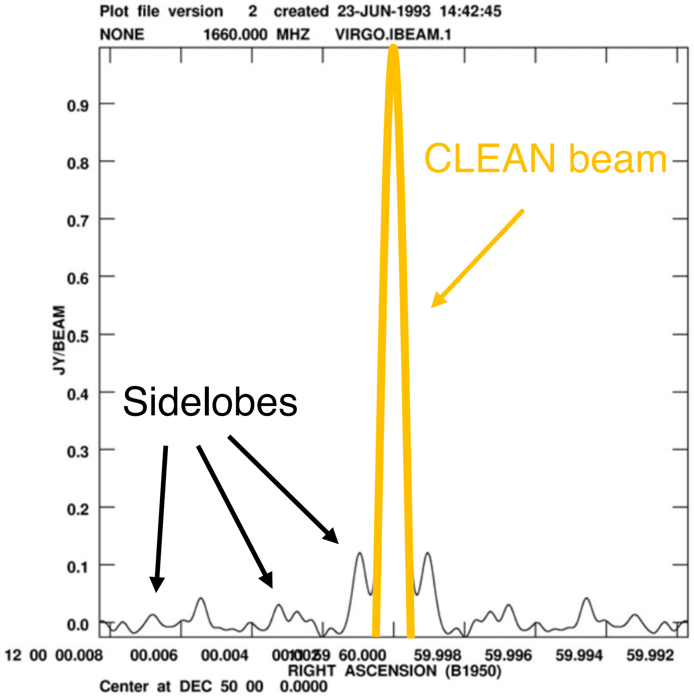
Credits: Daniel Tafoya, ERIS 2019

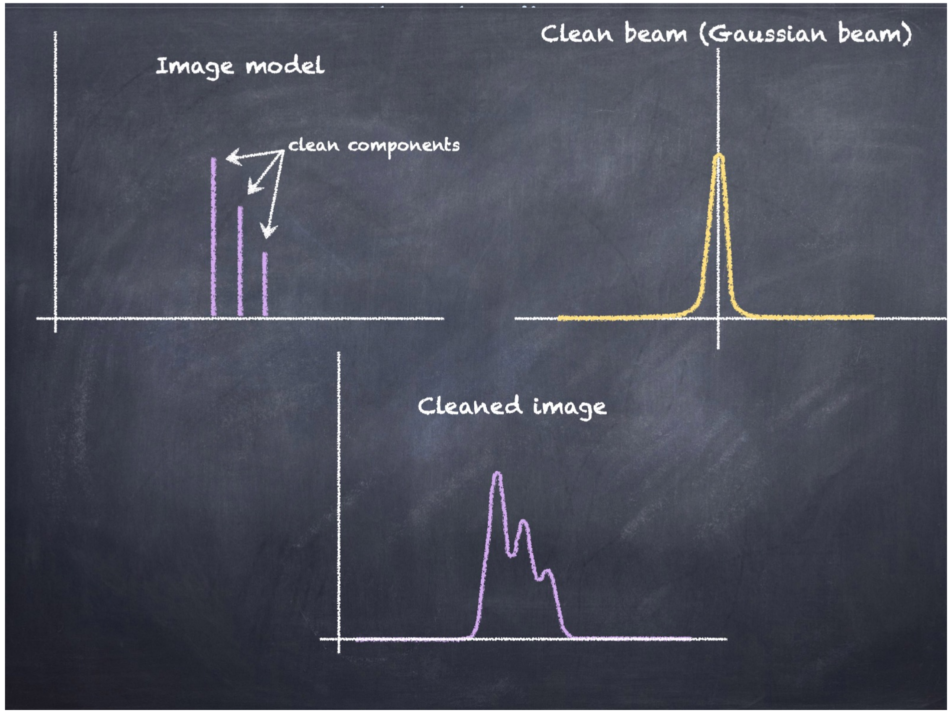


Deconvolution

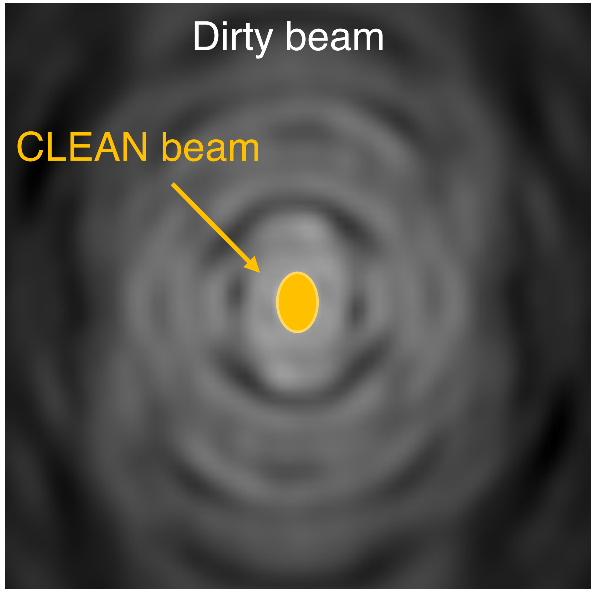


Credits: M. Rioja





Credits: Daniel Tafoya, ERIS 2019



Deconvolution

CLEAN method principal steps (Clark's algorithm):

1) Initialize a residual map (first image = dirty image)

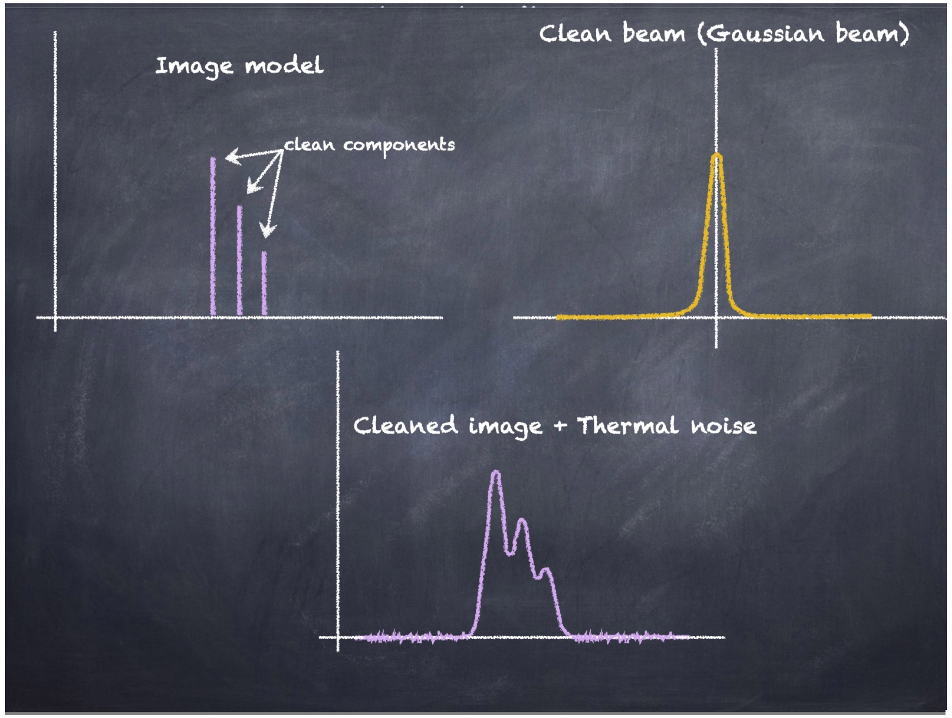
2) Identify strongest peak as a delta component

3) Record the position and magnitude in a model (clean components), subtract it from the dirty image

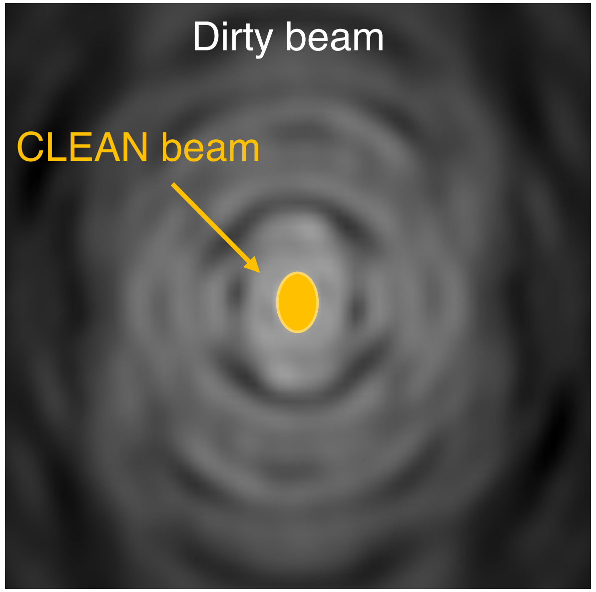
4) Go to 1) unless you reach the stopping criterion

5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)

6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019

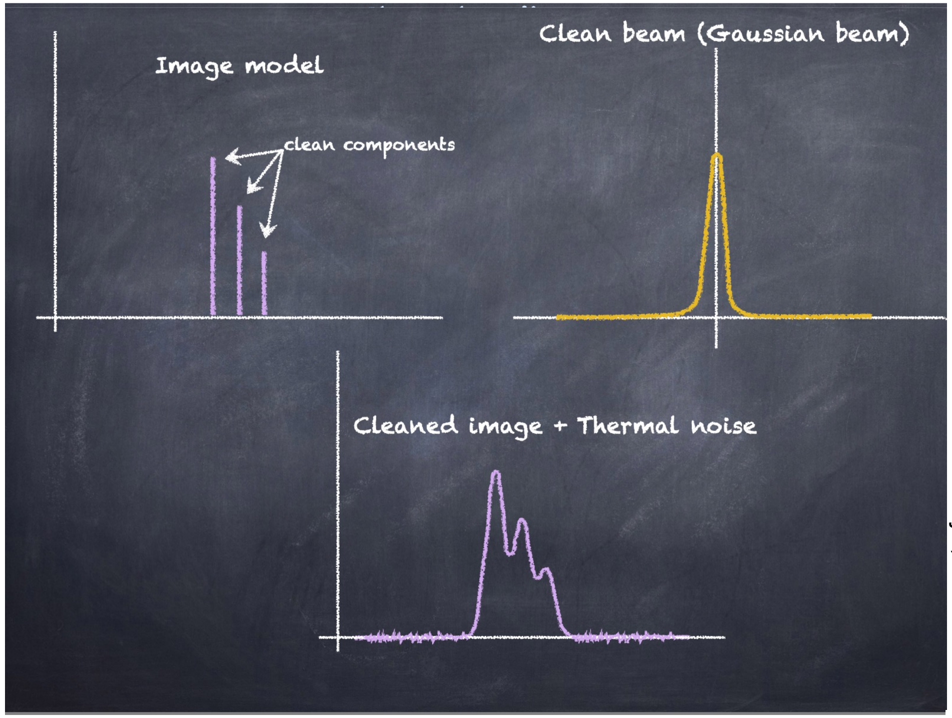


Deconvolution

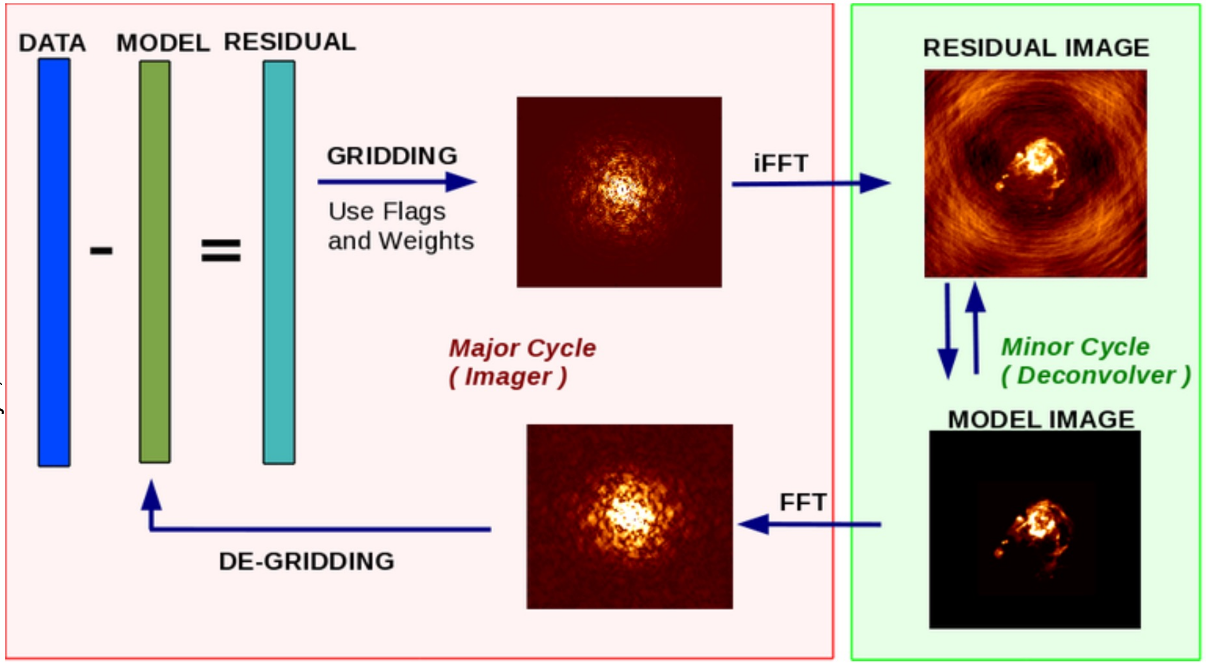
CLEAN method principal steps (Clark's algorithm):

Minor cycle

1) Initialize a residual map (first image = dirty image)
2) Identify strongest peak as a delta component
3) Record the position and magnitude in a model (clean components), subtract it from the dirty image
4) Go to 1) unless you reach the stopping criterion
5) Convolve the model (clean components) with an idealized CLEAN beam (elliptical Gaussian fit of the main lobe of the dirty beam)
6) Add the residual of the dirty image to the CLEAN image



Credits: Daniel Tafoya, ERIS 2019



Credits: Daniel Tafoya, ERIS 2019

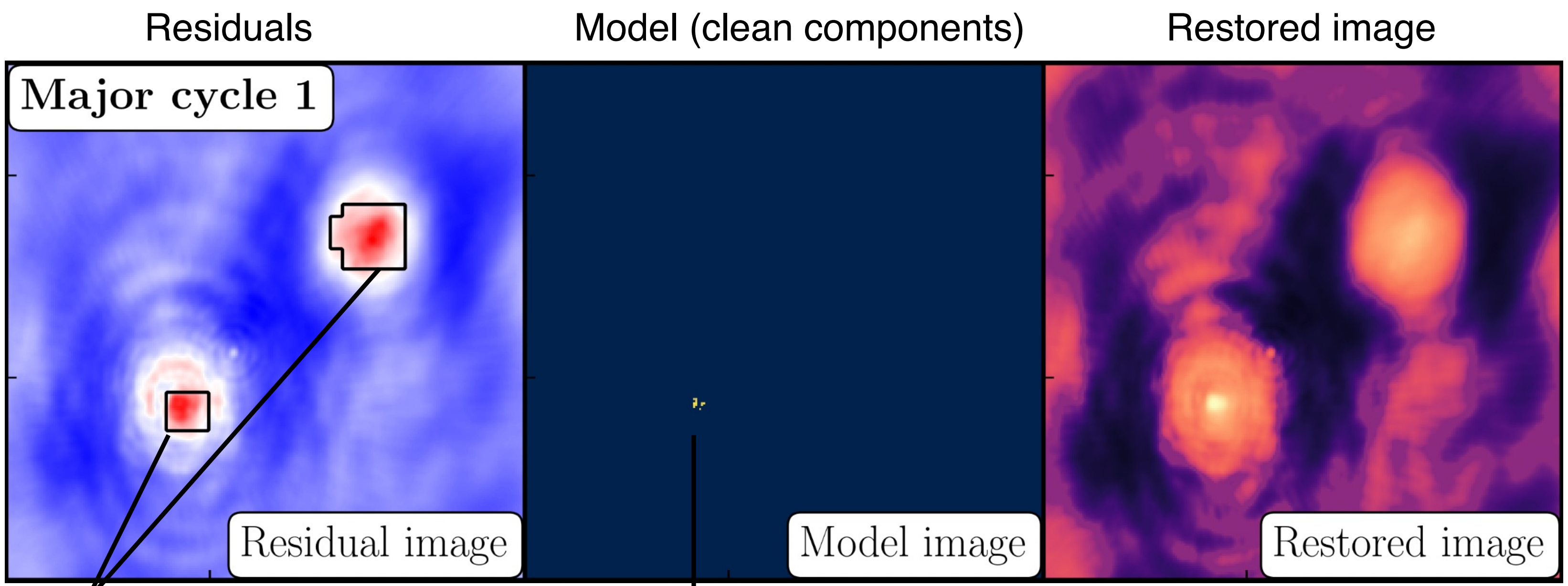
The major cycle implements FT between the data and image domains

The minor cycle operates purely in the image domain

(The 2-cycles approach makes the deconvolution faster)

Deconvolution in practice

CLEAN in action



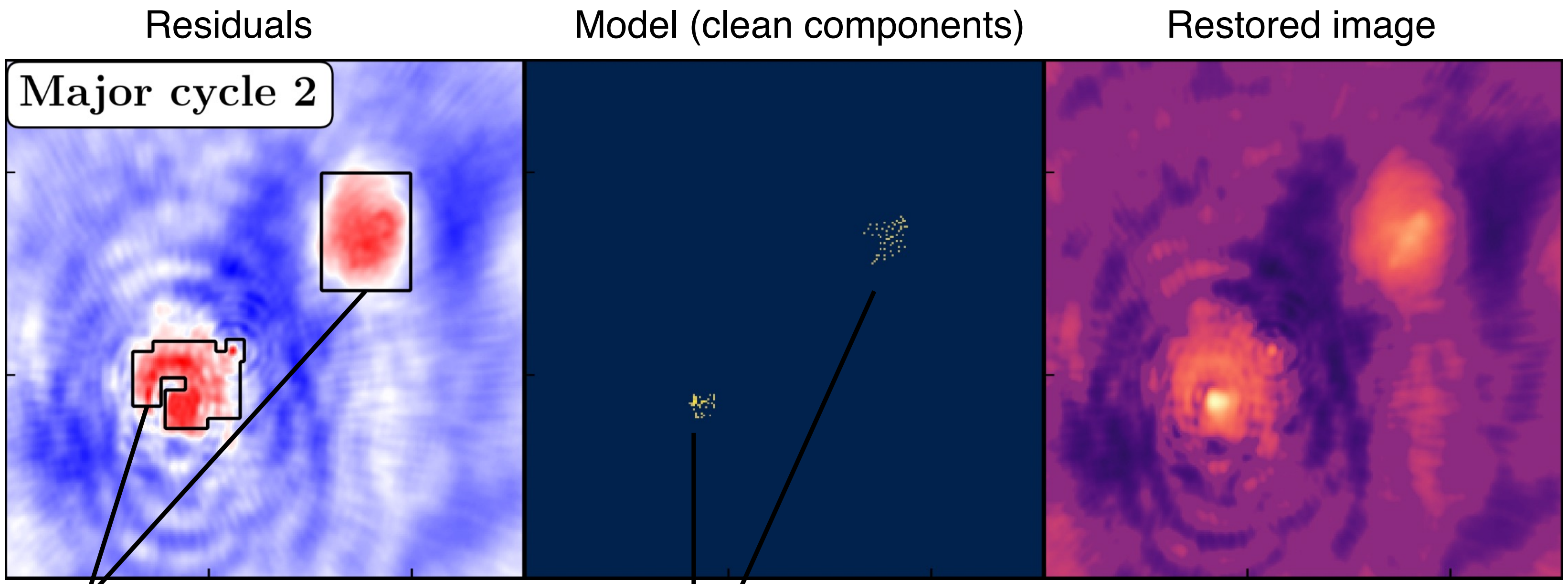
To make this process converge faster
we use the so-called CLEAN boxes (mask)

Also useful to not let CLEAN go to sidelobes
(see next slides)

CLEAN components obtained
during several minor cycles

Deconvolution in practice

CLEAN in action

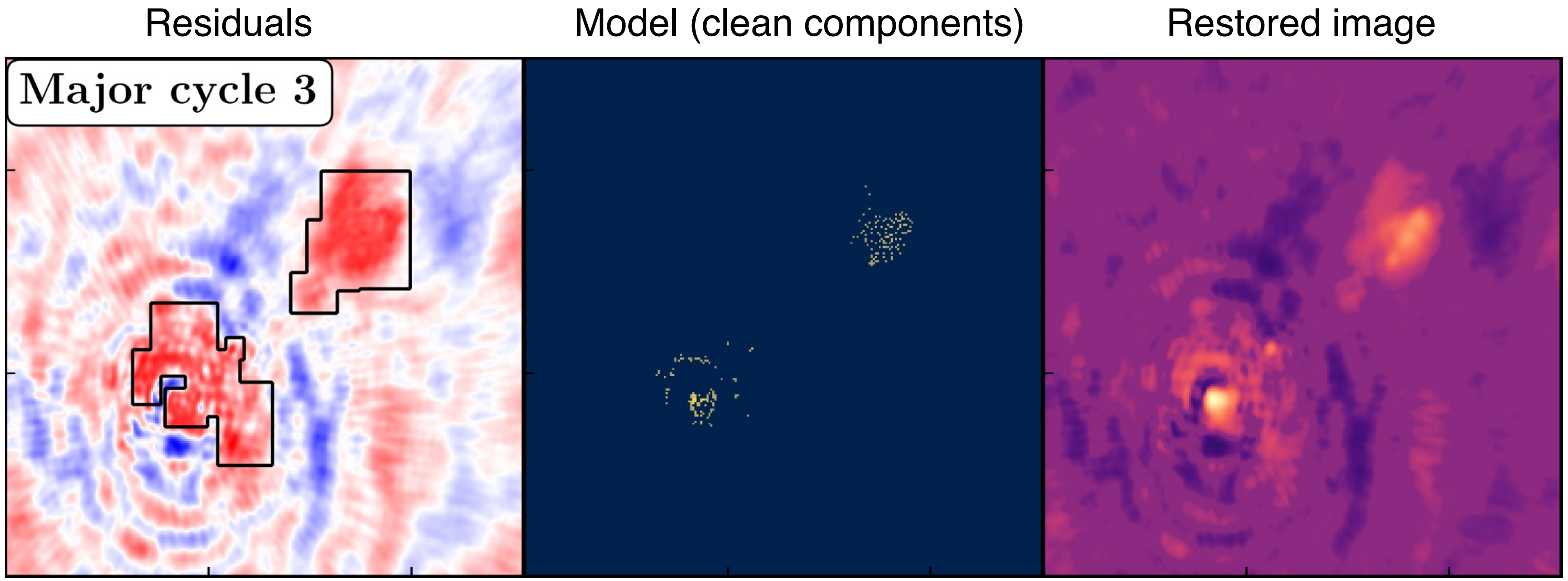


The residual image now shows fainter emission: we enlarge the CLEAN boxes to cover this

New CLEAN components added to the previous ones

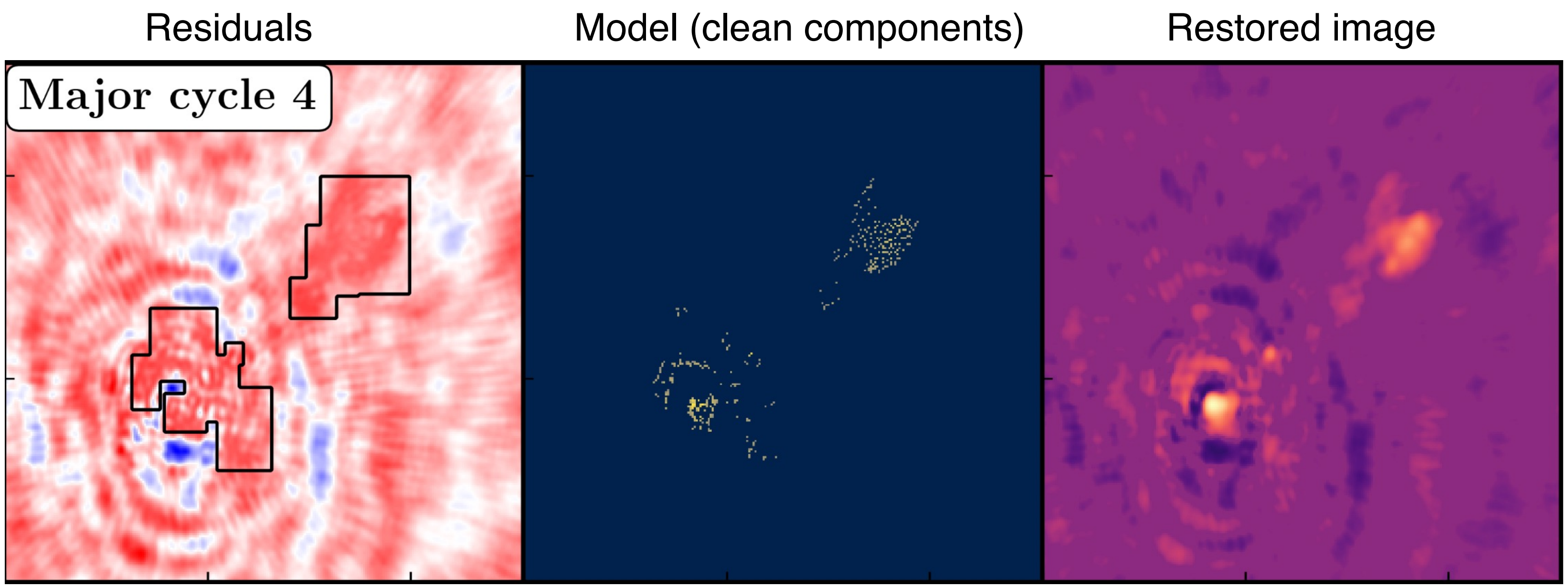
Deconvolution in practice

CLEAN in action



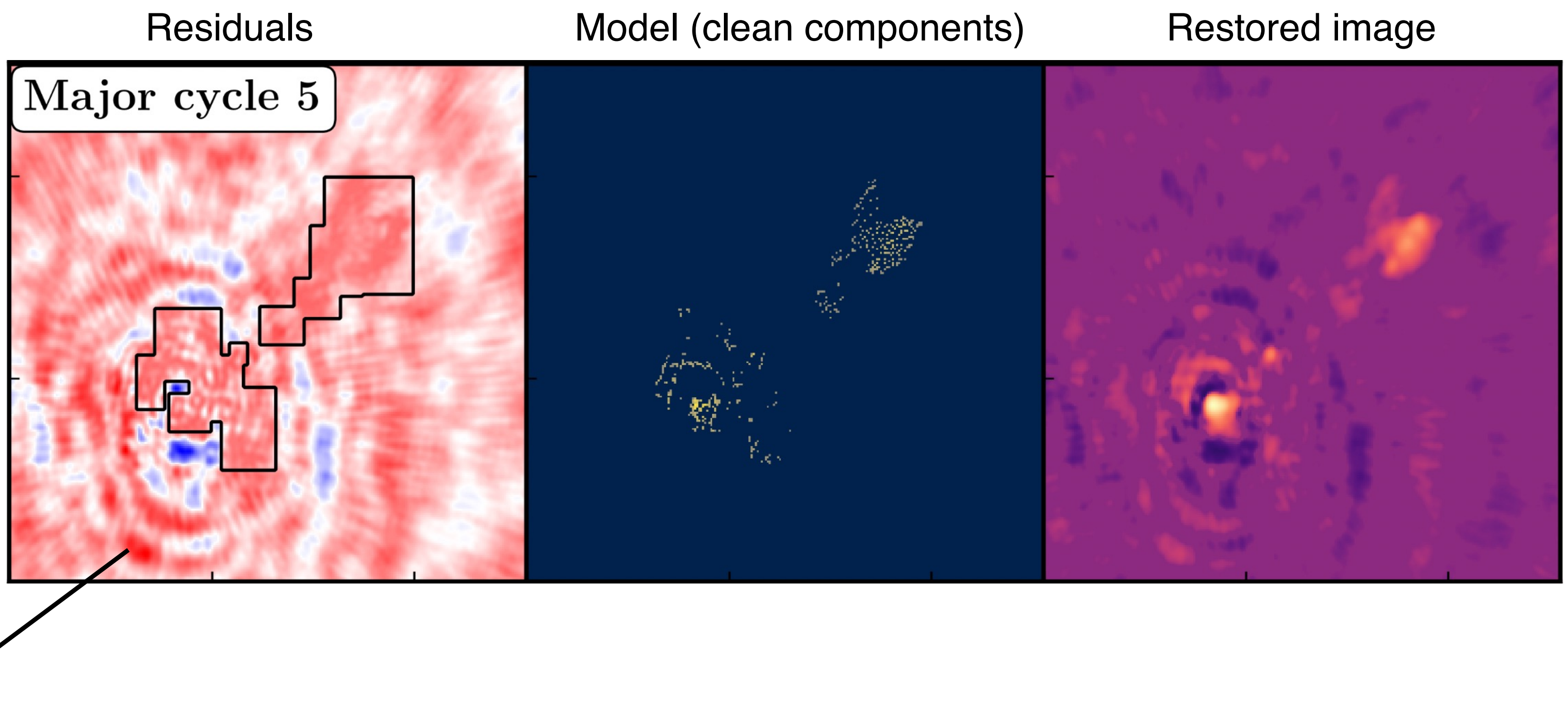
Deconvolution in practice

CLEAN in action



Deconvolution in practice

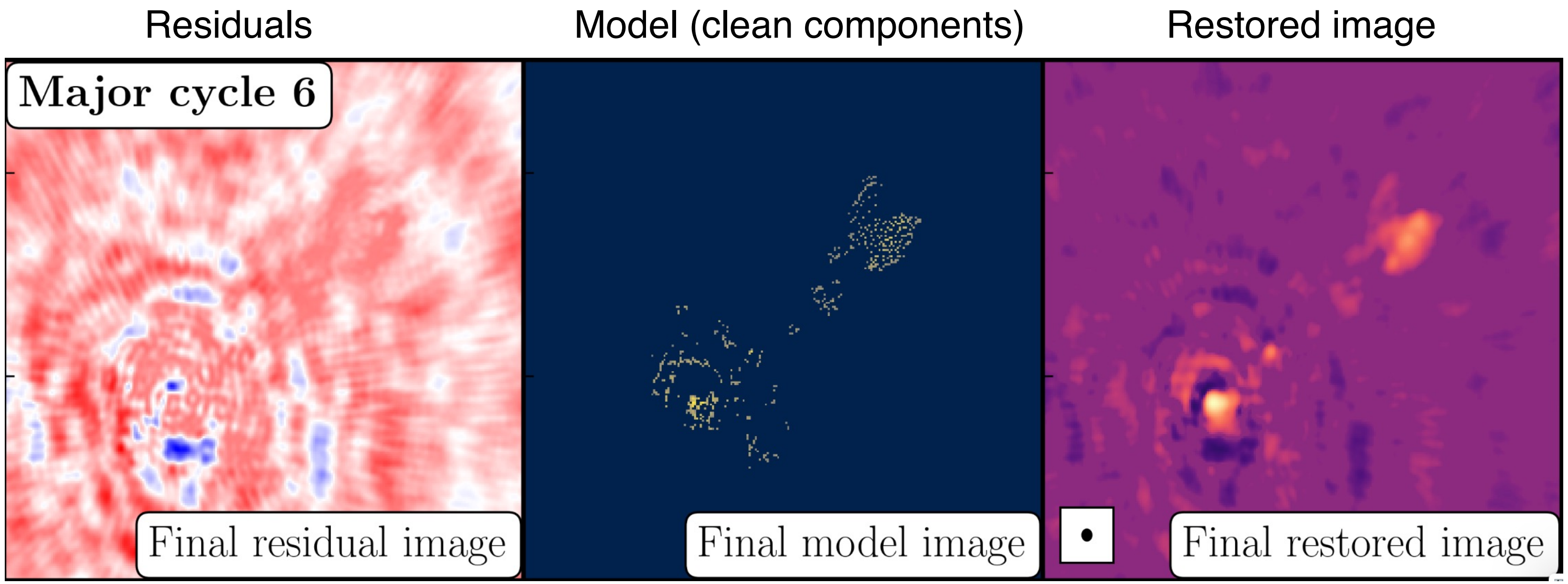
CLEAN in action



This emission is brighter BUT it's due to sidelobes!
It's always a good idea take a look at the dirty beam before starting cleaning
+ CLEAN boxes prevent the CLEANing of sidelobes

Deconvolution in practice

CLEAN in action



Residual image should look like «only noise»

CLEANing stopping criteria

- **Visually**, when your residuals contain only noise – this means that you cleaned all the flux density of the source

- **Convergence**: Check the logger for max-min (possibily symmetrical), total flux density should increase while cleaning (if not, stop), noise level should decrese (if it does not change anymore, stop → overcleaning)

- **Negative peak identified** (negatives can indicate that CLEAN is now working on sidelobes/noise, but it can also indicate that CLEAN is trying to fix earlier mistakes)

- **Smallest peak identified below a threshold** – which can be noise-based (e.g. 3 x theoretical noise estimated with exposure calculator – thermal noise)

- **Number of iterations** (not the best criterion, as you may end up doing too much or too little cleaning)

CLEANing issues and recognizing errors

CLEANING-related

- Interpolation of **unsampled (u,v) spacings** (in particular short spacings) : reconstruction of largest spatial scales is always an extrapolation (CLEAN boxes help)
- Assumption of **point-sources for extended structure** is not great
- **Under- and over-cleaning** are often an issue
(over-cleaning: rms in logger does not change anymore)
- **Computationally expensive**, as it requires iterative, non-linear fitting process (CLEAN boxes help this too)

Calibration and data-handling related

- **Bandwidth and time smearing**
- **Amplitude/phase errors** from previous calibration and/or unflagged data **(symmetric/antisymmetric artefacts)**

Source-related

- **Variability** of the source
- **Spectral variations of the source** – multi frequency synthesis
(gridding different frequencies on the same (u,v) grid is now standard)

CLEANing issues and recognizing errors

CLEANING-related

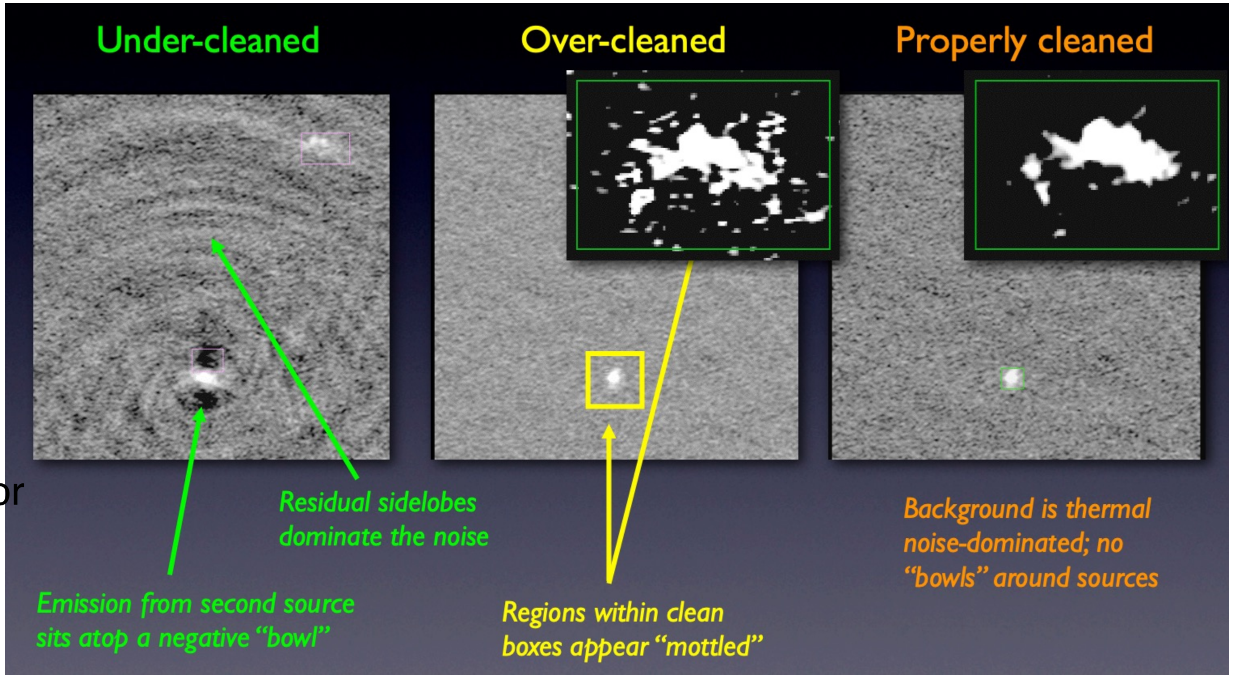
- Interpolation of **unsampled (u,v) spacings** (in particular short spacings) : reconstruction of largest spatial scales is always an extrapolation (CLEAN boxes help)
- Assumption of **point-sources for extended structure** is not great
- **Under- and over-cleaning** are often an issue (over-cleaning: rms in logger does not change anymore)
- **Computationally expensive**, as it requires iterative, non-linear fitting process (CLEAN boxes help this too)

Calibration and data-handling related

- **Bandwidth and time smearing**
- **Amplitude/phase errors** from previous calibration and/or unflagged data **(symmetric/antisymmetric artefacts)**

Source-related

- **Variability** of the source
- **Spectral variations of the source** – multi frequency synthesis (gridding different frequencies on the same (u,v) grid is now standard)



Credits: Michael Wise UVA lectures

CLEANing issues and recognizing errors

CLEANING-related

- Interpolation of **unsampled (u,v) spacings** (in particular short spacings) : reconstruction of largest spatial scales is always an extrapolation (CLEAN boxes help)
- Assumption of **point-sources for extended structure** is not great
- **Under- and over-cleaning** are often an issue (over-cleaning: rms in logger does not change anymore)
- **Computationally expensive**, as it requires iterative, non-linear fitting process (CLEAN boxes help this too)

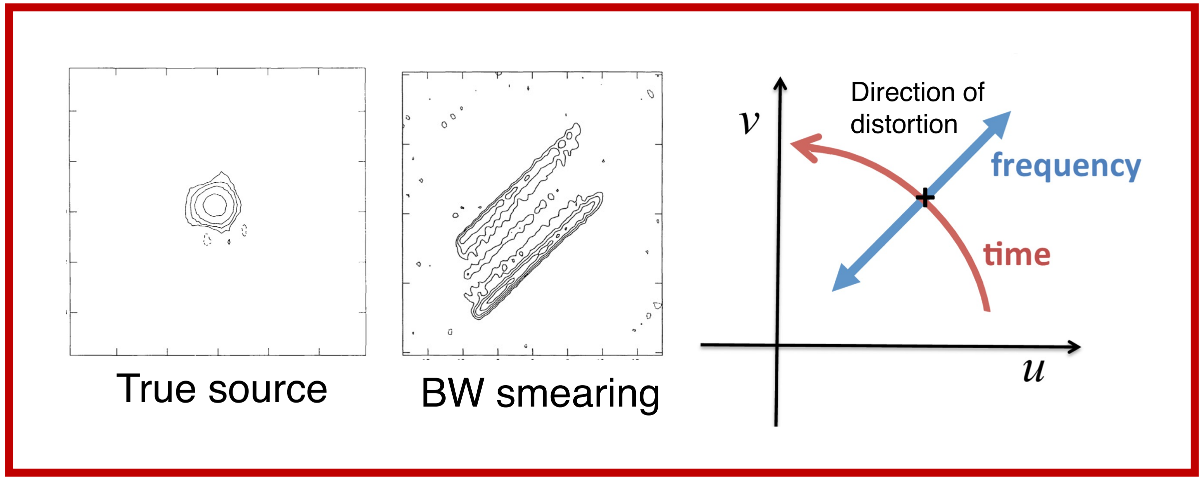
Calibration and data-handling related

- **Bandwidth and time smearing**
- **Amplitude/phase errors** from previous calibration and/or unflagged data **(symmetric/antisymmetric artefacts)**

Source-related

- **Variability** of the source
- **Spectral variations of the source** – multi frequency synthesis (gridding different frequencies on the same (u,v) grid is now standard)

Cotton 1989



CLEANing issues and recognizing errors

CLEANING-related

- Interpolation of **unsampled (u,v) spacings** (in particular short spacings) : reconstruction of largest spatial scales is always an extrapolation (CLEAN boxes help)

- Assumption of **point-sources for extended structure** is not great

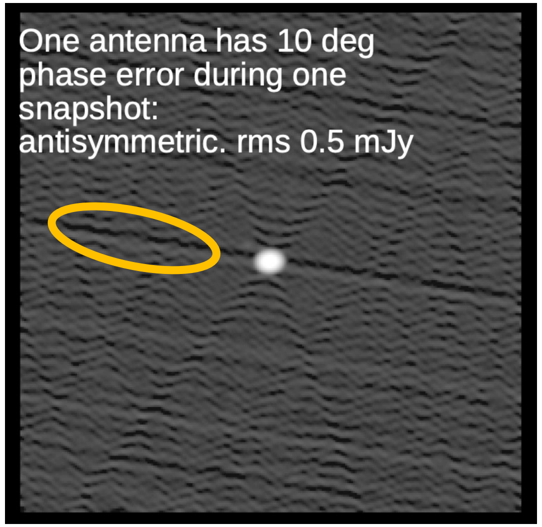
- **Under- and over-cleaning** are often an issue
(over-cleaning: rms in logger does not change anymore)

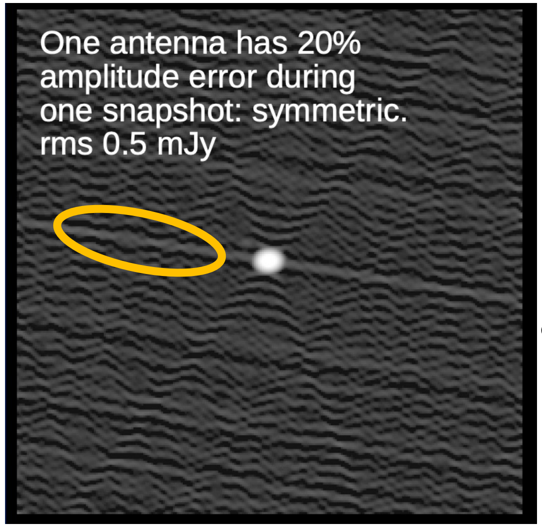
- **Computationally expensive**, as it requires iterative, non-linear fitting process (CLEAN boxes help this too)

Calibration and data-handling related

- **Bandwidth and time smearing**

- **Amplitude/phase errors** from previous calibration and/or unflagged data **(symmetric/antisymmetric artefacts)**





Credits: Laing ERIS 2013

Source-related

- **Variability** of the source

- **Spectral variations of the source** – multi frequency synthesis
(gridding different frequencies on the same (u,v) grid is now standard)

CLEANing issues and recognizing errors

CLEANING-related

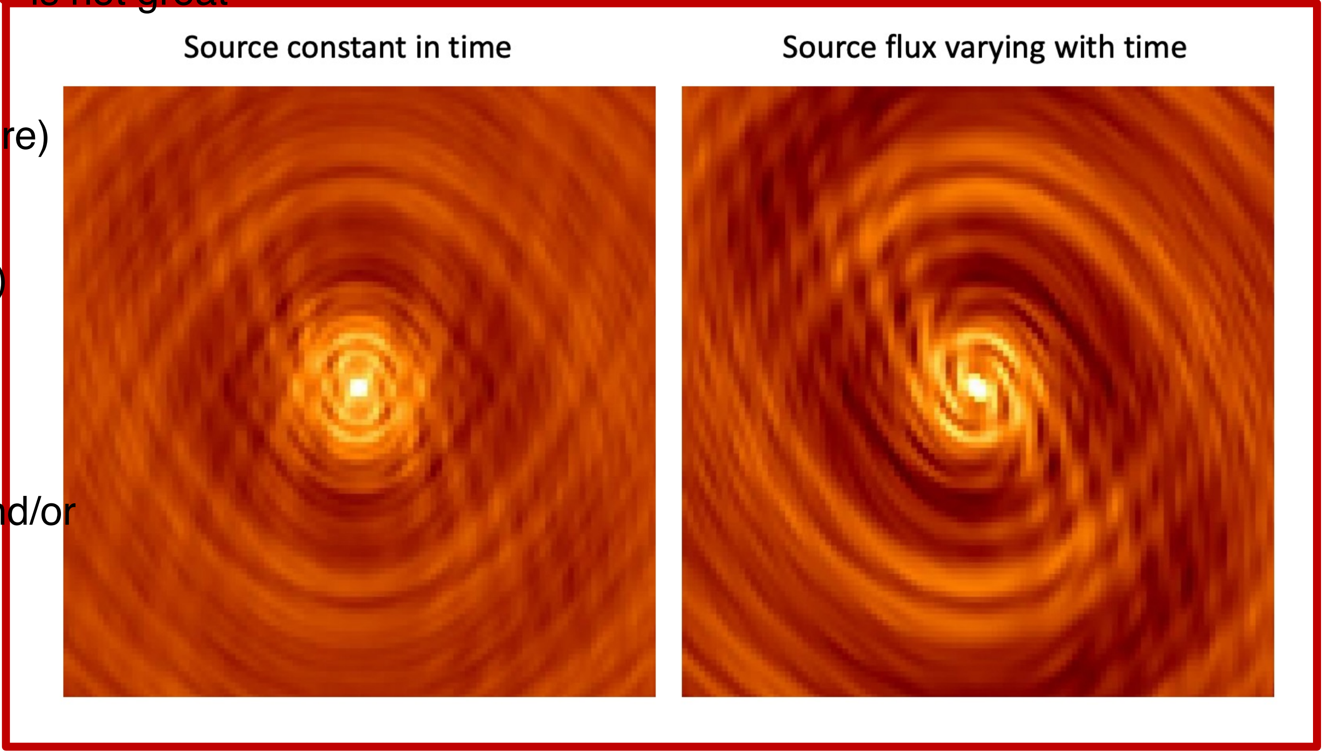
- Interpolation of **unsampled (u,v) spacings** (in particular short spacings) : reconstruction of largest spatial scales is always an extrapolation (CLEAN boxes help)
- Assumption of **point-sources for extended structure** is not great
- **Under- and over-cleaning** are often an issue (over-cleaning: rms in logger does not change anymore)
- **Computationally expensive**, as it requires iterative, non-linear fitting process (CLEAN boxes help this too)

Calibration and data-handling related

- **Bandwidth and time smearing**
- **Amplitude/phase errors** from previous calibration and/or unflagged data **(symmetric/antisymmetric artefacts)**

Source-related

- **Variability** of the source
- **Spectral variations of the source** – multi frequency synthesis (gridding different frequencies on the same (u,v) grid is now standard)

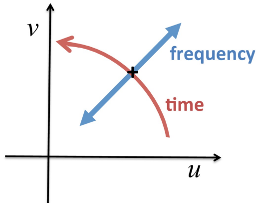


Credits: Stewart 2012 GLOW school

CLEANing issues and recognizing errors

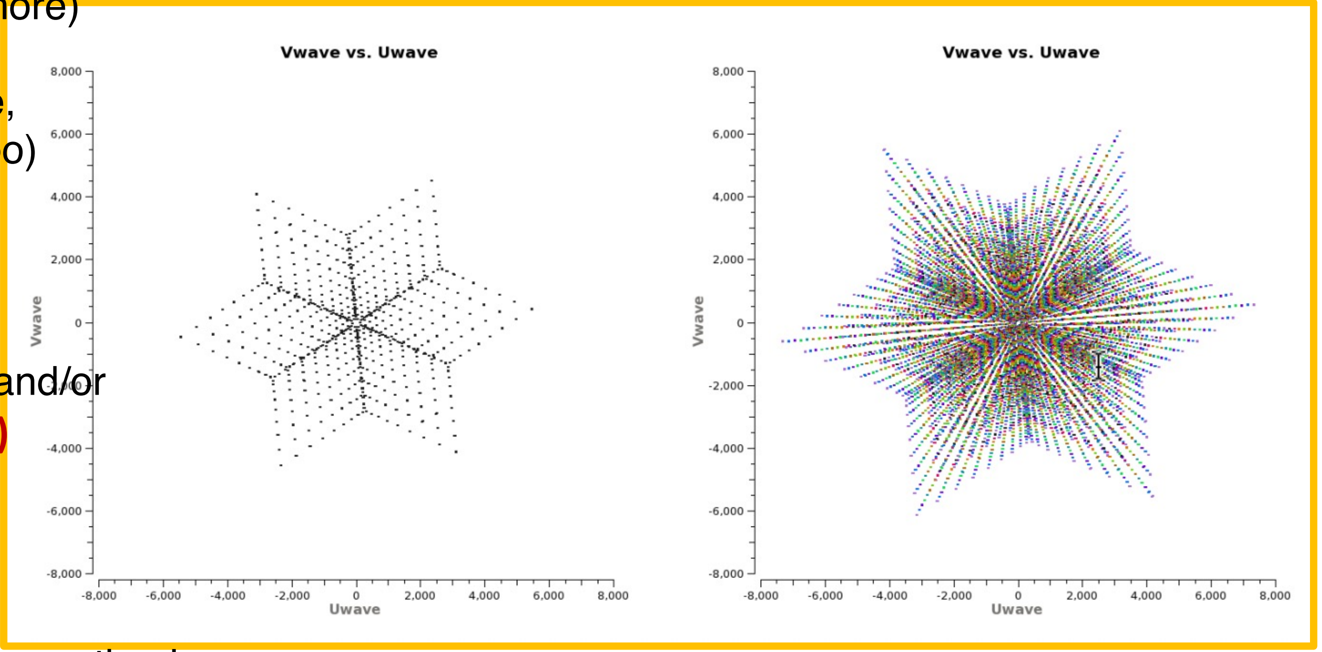
CLEANING-related

- Interpolation of **unsampled (u,v) spacings** (in particular short spacings) : reconstruction of largest spatial scales is always an extrapolation (CLEAN boxes help)
- Assumption of **point-sources for extended structure** is not great
- **Under- and over-cleaning** are often an issue
(over-cleaning: rms in logger does not change anymore)
- **Computationally expensive**, as it requires iterative, non-linear fitting process (CLEAN boxes help this too)



Calibration and data-handling related

- **Bandwidth and time smearing**
- **Amplitude/phase errors** from previous calibration and/or unflagged data **(symmetric/antisymmetric artefacts)**



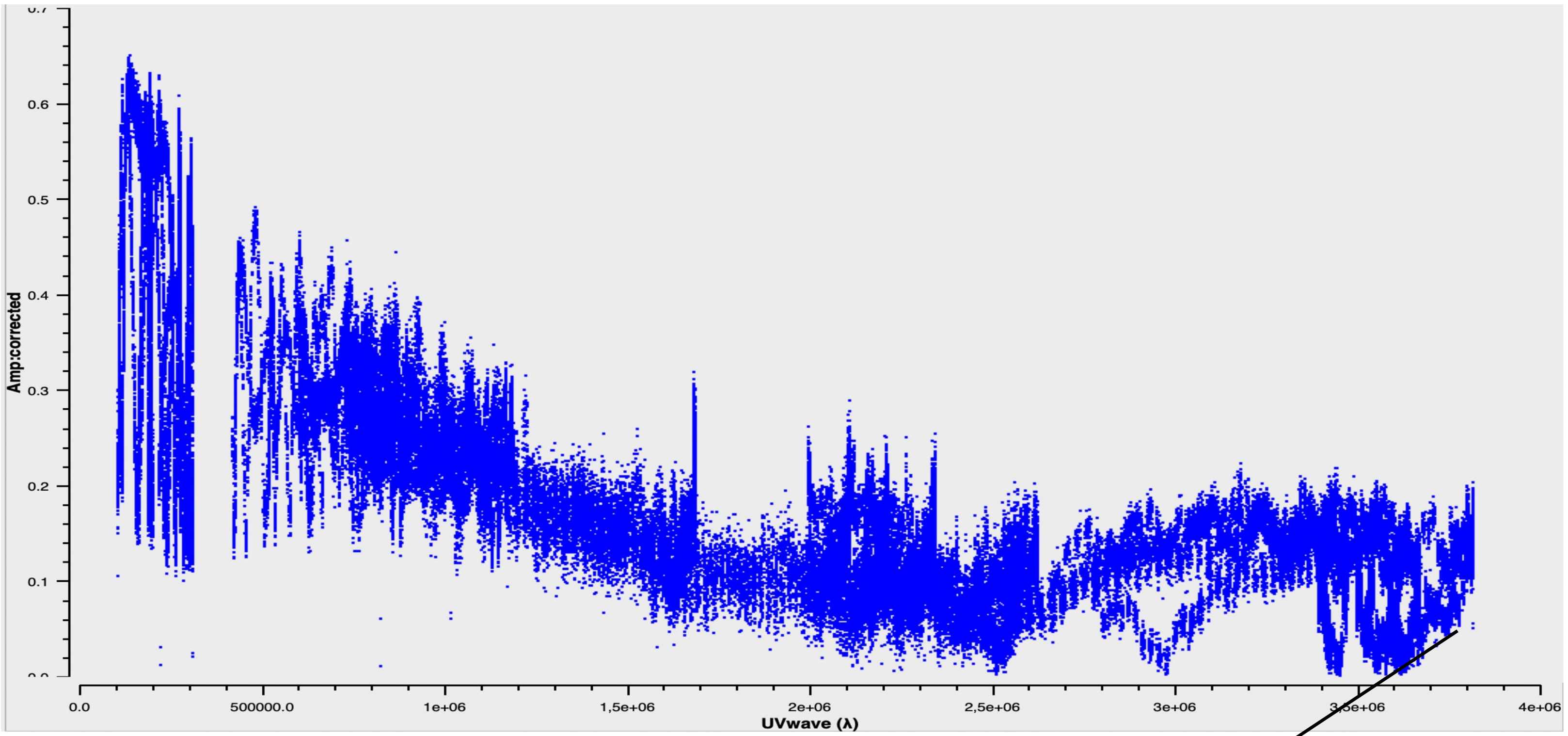
Credits: Michael Wise UVA lectures

Source-related

- **Variability** of the source
- **Spectral variations of the source** – multi frequency synthesis
(gridding different frequencies on the same (u,v) grid is now standard)

Determining imaging parameters

Imaging parameters: pixel size



Nyquist sampling theorem in astronomical terms

minimum number of pixels required to obtain a fit to a Gaussian must be at least 3 pixels

$$\text{cell} \approx \frac{180}{\pi N_s} \times \frac{1}{D_{\max}\,[\lambda]}\ [\text{deg}]$$

N_s typically = 5-7

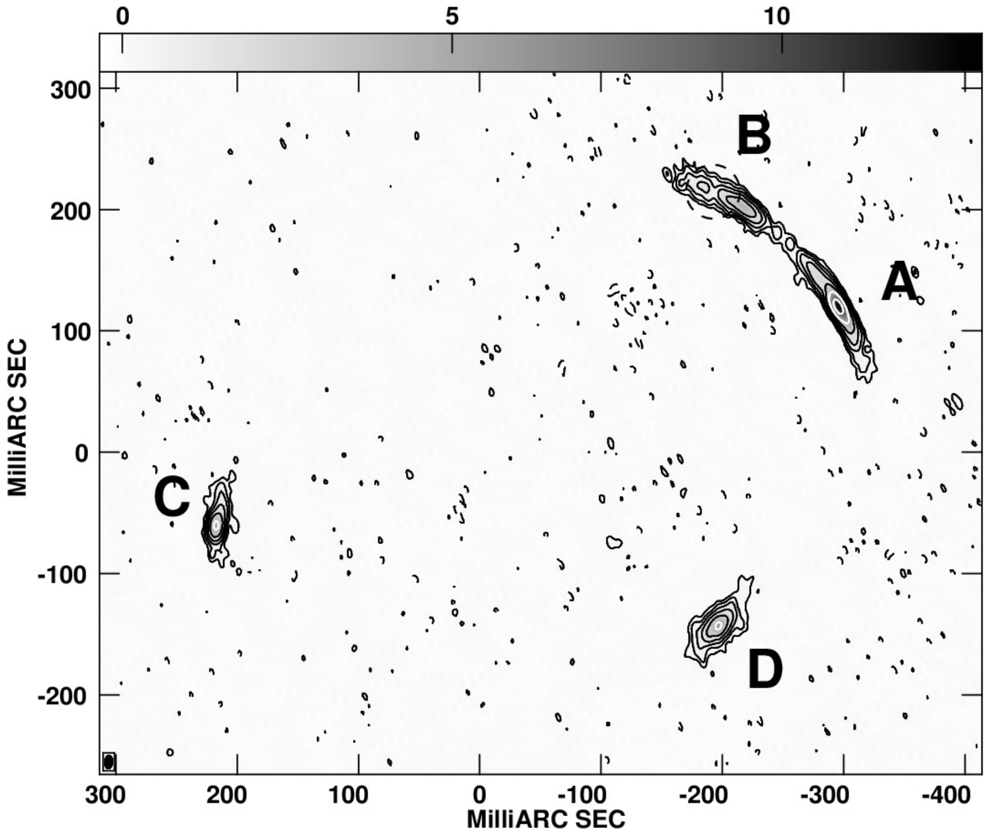
must not be smaller than 3!

Imaging parameters: field of view / image size

The source size is typically much smaller than the entire Field-of-View (FoV), which corresponds approximately to the single-dish beam $\approx \lambda/D$ (homogeneous array)

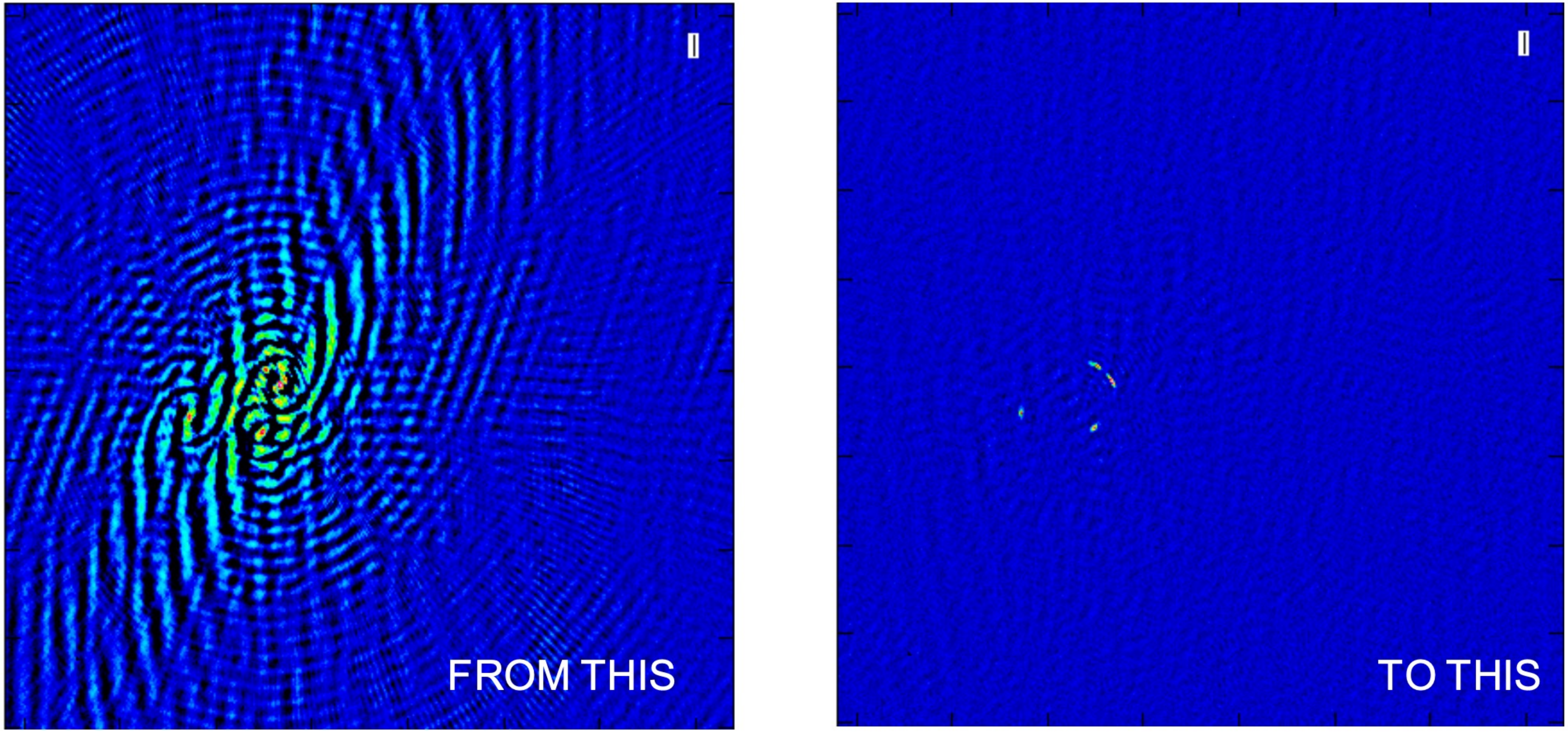
But it is also limited by time and bandwidth smearing:
a typical FoV for VLBI is of the order of a few arcseconds

Also: it's always good to check
what is already known about your target!
For the test data for this CASA-VLBI workshop
the target is JVAS B0128+437
(L-band VLBA observations)



Biggs et al. 2004, MNRAS, 350, 949
and https://arxiv.org/pdf/astro-ph/0412677.pdf (Biggs+ 2004 EVN proceeding)

Let's create a CLEAN image



Imaging parameters in CASA: tclean

If you don't know what a parameter means, its units… just type **`help tclean`**

`imagename`

`field`

`cell`

`imsize`

`deconvolver`

`niter`

`weighting`

Imaging parameters in CASA: tclean

If you don't know what a parameter means, its units… just type **`help tclean`**

`imagename` this is up to you

`field` Name of the field (source) that you would like to image

`cell`

`imsize`

`deconvolver`

`niter`

`weighting`

Imaging parameters in CASA: tclean

If you don't know what a parameter means, its units…
just type
`help tclean`

`imagename` this is up to you

`field` Name of the field (source) that you would like to image

`cell` Angular size of your pixel

`imsize`

`deconvolver`

`niter`

`weighting`

Imaging parameters in CASA: tclean

If you don't know what a parameter means, its units… just type
`help tclean`

`imagename` this is up to you

`field` Name of the field (source) that you would like to image

`cell` Angular size of your pixel

`imsize` Size of image in pixels – typically power of two 2^n (128x128, 256x256, 512x512 etc.)

`deconvolver`

`niter`

`weighting`

Imaging parameters in CASA: tclean

If you don't know what a parameter means, its units… just type **`help tclean`**

`imagename` this is up to you

`field` Name of the field (source) that you would like to image

`cell` Angular size of your pixel

`imsize` Size of image in pixels – typically power of two 2^n (128x128, 256x256, 512x512 etc.)

`deconvolver` CLEAN algorithm -- Clark or Hogbom algorithms are fine for starting

`niter`

`weighting`

Imaging parameters in CASA: tclean

If you don't know what a parameter means, its units… just type **`help tclean`**

`imagename` this is up to you

`field` Name of the field (source) that you would like to image

`cell` Angular size of your pixel

`imsize` Size of image in pixels – typically power of two 2^n (128x128, 256x256, 512x512 etc.)

`deconvolver` CLEAN algorithm -- Clark or Hogbom algorithms are fine for starting

`niter` If = 0 computes the dirty image; if > 0 runs major and minor cycles (sub-parameter *'cycleniter'*)

`weighting`

Imaging parameters in CASA: tclean

If you don't know what a parameter means, its units… just type **`help tclean`**

`imagename` this is up to you

`field` Name of the field (source) that you would like to image

`cell` Angular size of your pixel

`imsize` Size of image in pixels – typically power of two 2^n (128x128, 256x256, 512x512 etc.)

`deconvolver` CLEAN algorithm -- Clark or Hogbom algorithms are fine for starting

`niter` If = 0 computes the dirty image; if > 0 runs major and minor cycles (sub-parameter *'cycleniter'*)

`weighting` $V_{obs}(l,m) = V_{true}(u,v)\ S(u,v)$

S(u,v) is 1 if there's a measurement and 0 elsewhere
so-called **natural weights**

Imaging parameters in CASA: a slide about weights

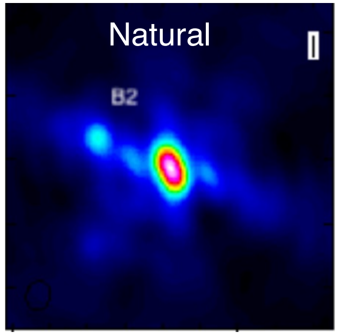
weighting

$V_k \rightarrow$ AMP(a_k) PHASE(ϕ_k) NOISE(σ_k) WEIGHT (w_k)

Better rms, worse beam

Natural $w_k = 1 / \sigma_k^2$

«more weights on short baselines», best sensitivity (important for more extended structures) but poor beam shape with overemphasized sidelobes

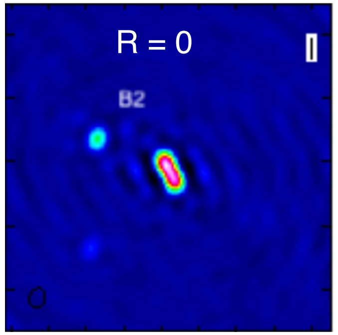


Robust (Briggs 1995) $w_k = 1 / (S^2 + \sigma^2_k)$

$$S^2 = \frac{(5 \times 10^{-R})^2}{\overline{w}}$$

R = robustness (or robust factor) and it goes from -2 to 2

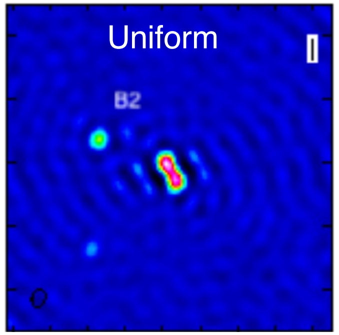
Average variance weighting factor over the grid cell in the image



Uniform $w_k = 1 / \varrho(u_k, v_k)$

Sampling density function

«more weights on long baselines», better resolution (tighter main lobe) and lower sidelobes

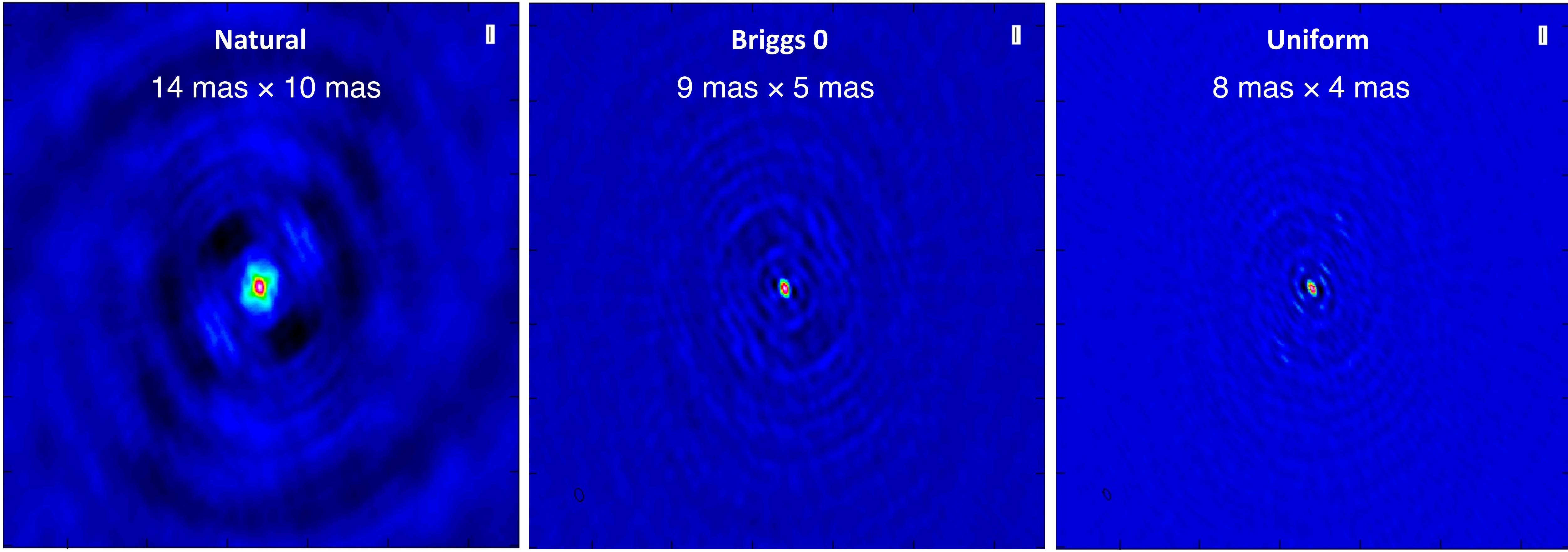


Better beam, worse rms

Imaging parameters in CASA: a slide about weights

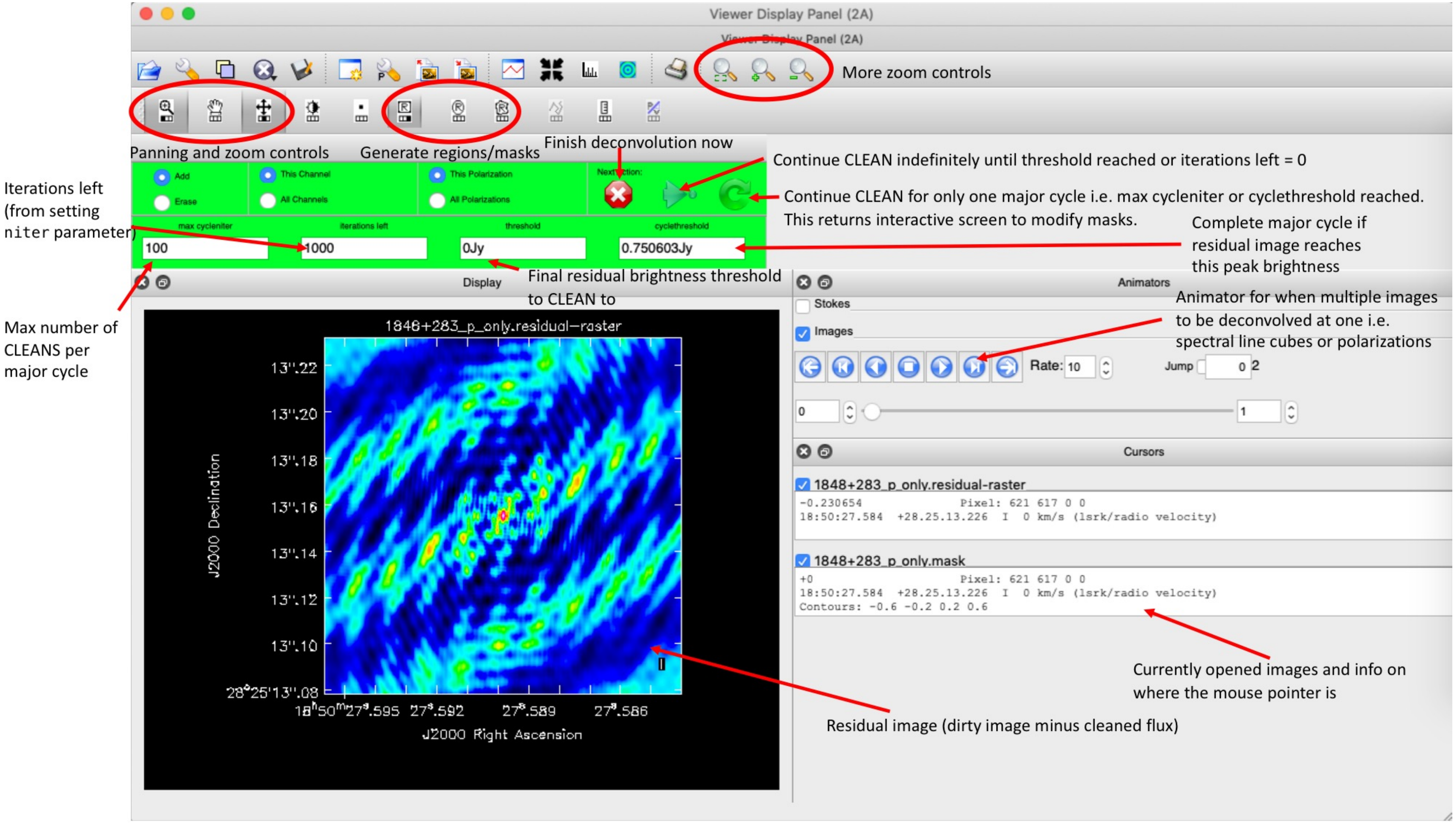
weighting

Different weights lead to different PSF (Dirty Beam) hence we should adjust the cellsize according to the chosen weights



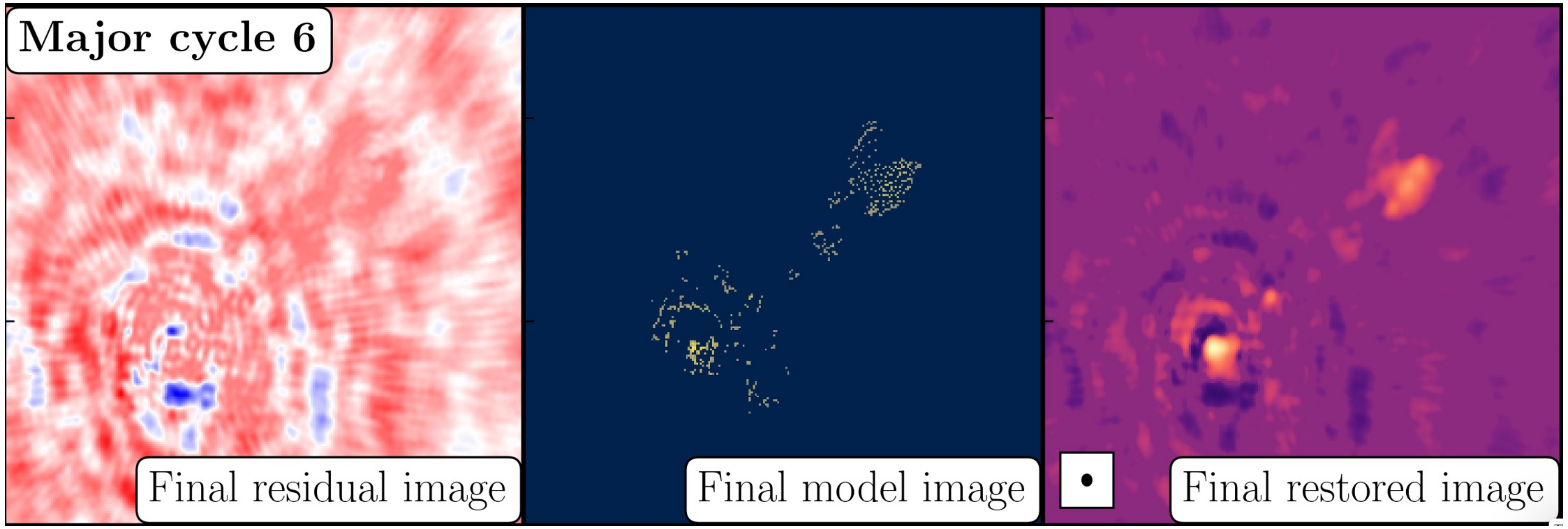
Imaging in CASA: interactive cleaning

Once you are happy with the choice of parameters … `go tclean`



Imaging in CASA: interactive cleaning

Continue the cleaning process unitl your image looks like noise



ALWAYS take a look at the logger!

E.g., Flux density in the model should increase

```
Major Cycle 1    model=0->0.177024,
Major Cycle 2    model=0.177024->0.381401,
Major Cycle 3    model=0.381401->0.504565
```

If it keeps decreasing: STOP CLEANING!

```
model=0.686961->0.683864
```

Imaging in CASA: output of tclean

.image → Final cleaned map

.tt0, .tt1 … → Suffix to indicate Taylor terms for multi-term wideband imaging if «mtmfs» is used as deconvolver

.alpha and .alpha.error → spectral index and its error map

.mask → mask used (clean boxes)

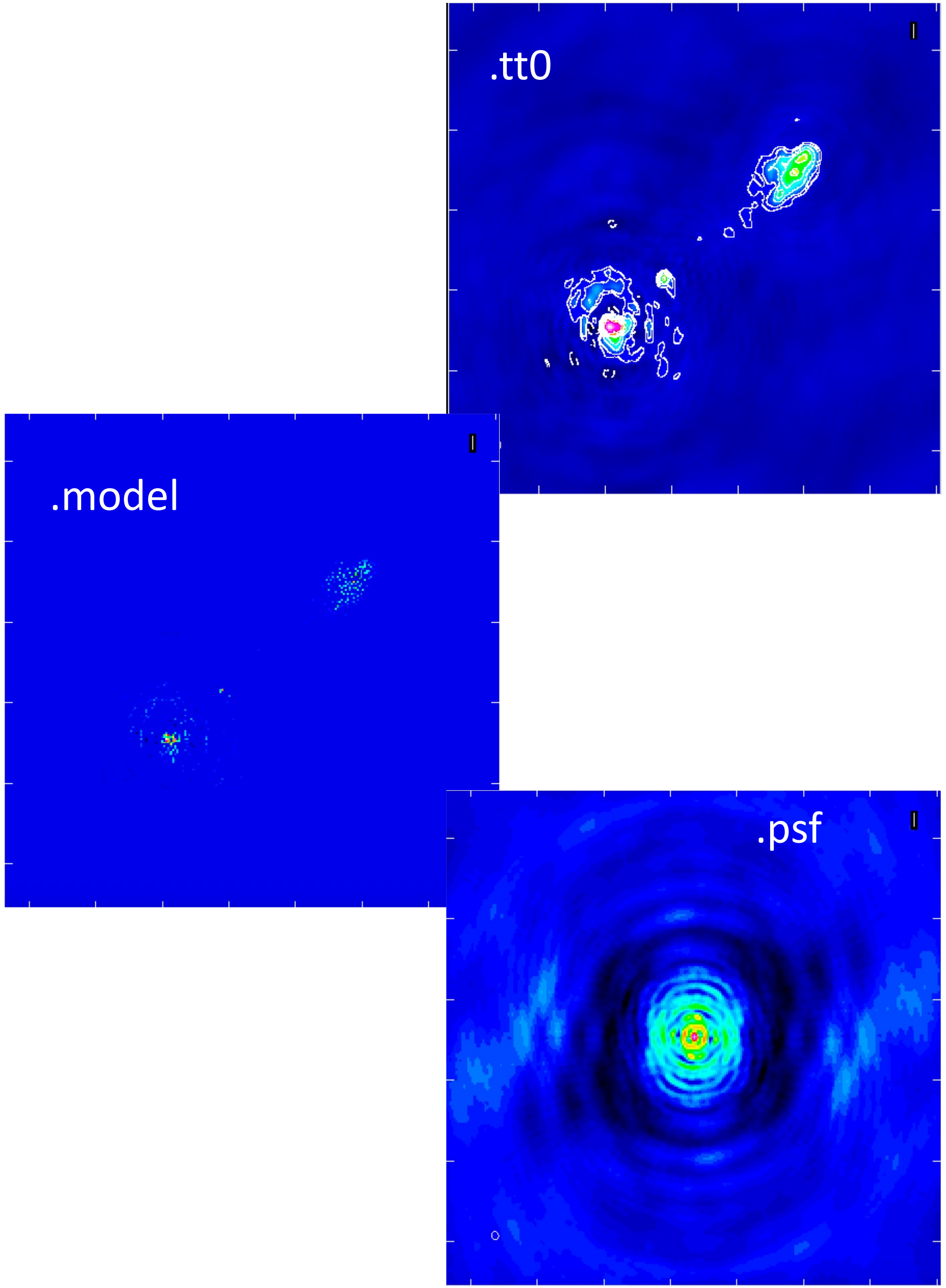
.model→ clean components

.psf (for tt0, tt1,...) → dirty beam

.pb → primary beam

.residual → residual image (data – model)

.sumwt → sum of the weights



Other details can be found here
https://casa.nrao.edu/docs/taskref/tclean-task.html

Measuring image properties

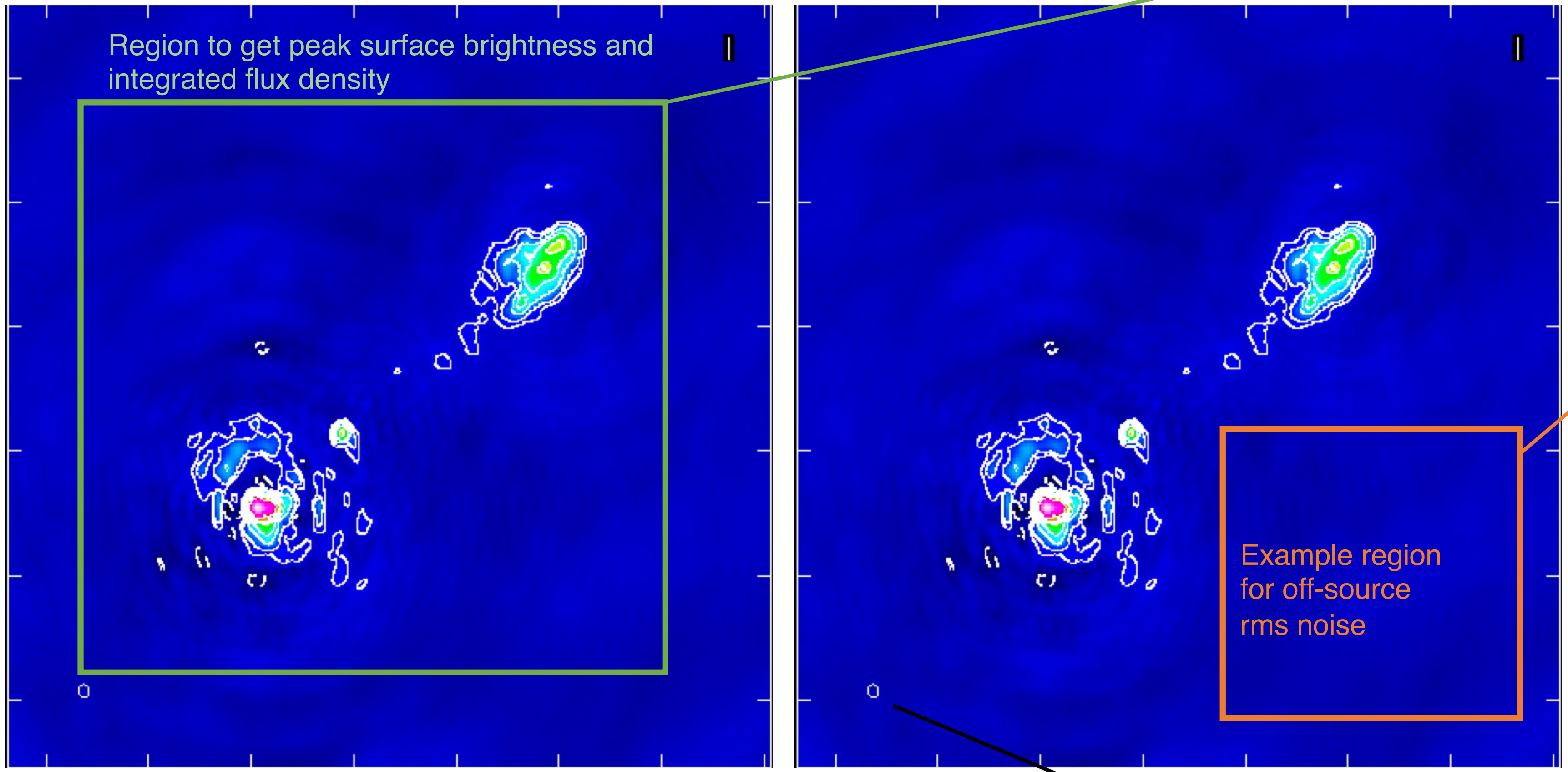
See also https://www.jb.man.ac.uk/DARA/ERIS22/imaging.html#image_properties

Imaging in CASA: measuring image properties

Directly using `IMVIEW`
Or using the task `IMSTAT`

The peak surface brightness is in Jy/beam

While the integrated flux density is in **Jy**



The rms of the image is in Jy / beam

Restoring beam (CLEAN beam)

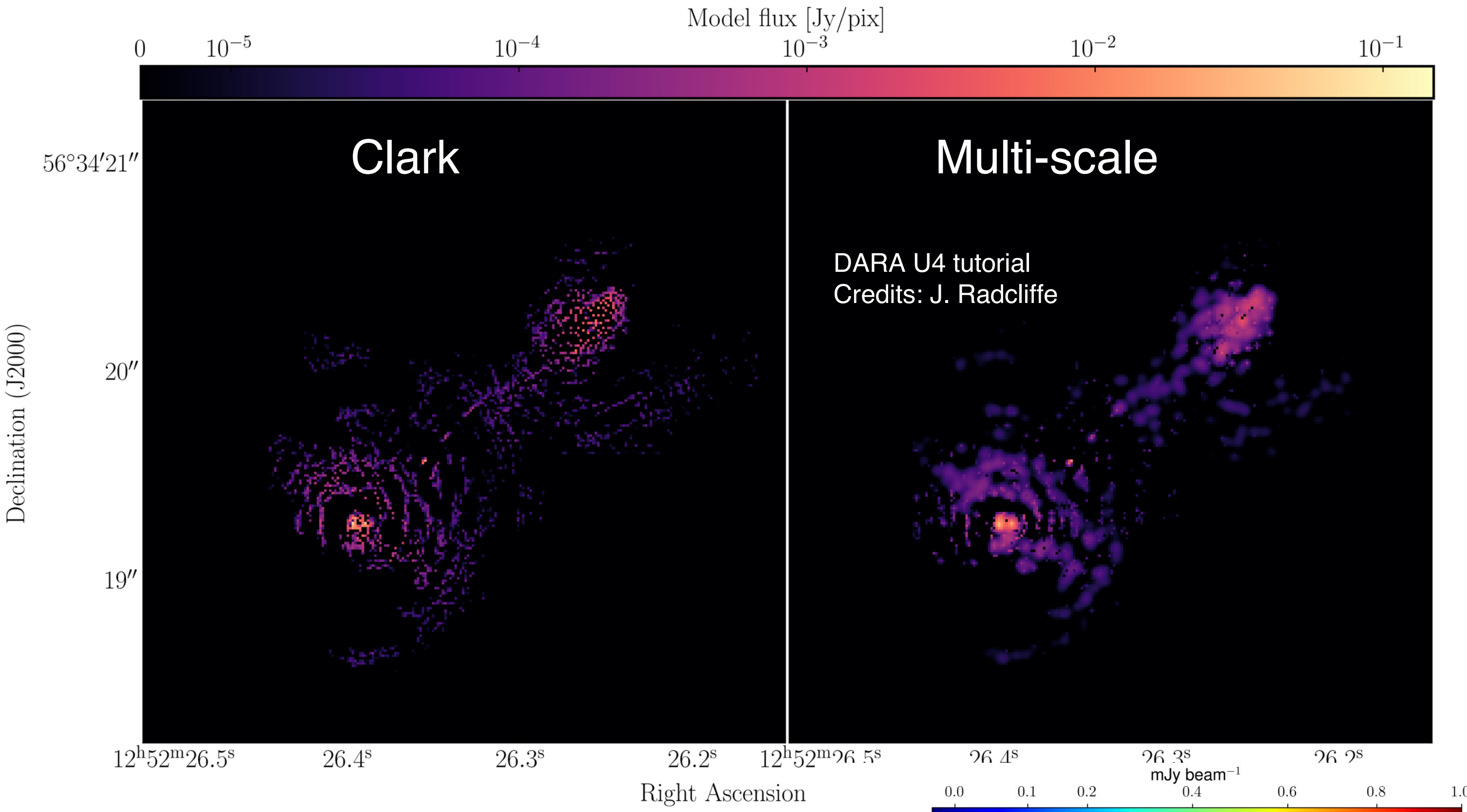
What is the dynamic range
(peak / off-source rms)
of your image
???

Imaging in CASA: measuring image quality

- **Off-source rms** noise close to theoretical noise
- **Dynamic range** (peak / off-source rms) -- typical (good) values 10^2- 10^6
- **«Fidelity»** – difference with an input model (need *a priori* info)
- Off-source rms **noise structure quite uniform**, close to a Gaussian random field («no stripes»): check for any phase and amplitude errors (see previous slides)
- any «weird» structure might be a symptom that something went wrong (at the deconvolution stage and/or during calibration)

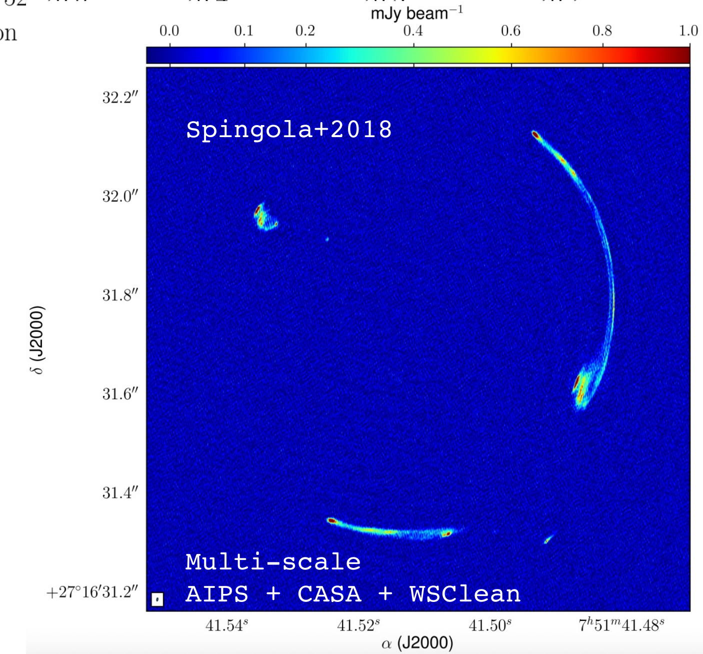
Multi-scale cleaning

`Deconvolver: mtmfs`



Default: clean components are *delta functions* – it does not work very well for extended sources!

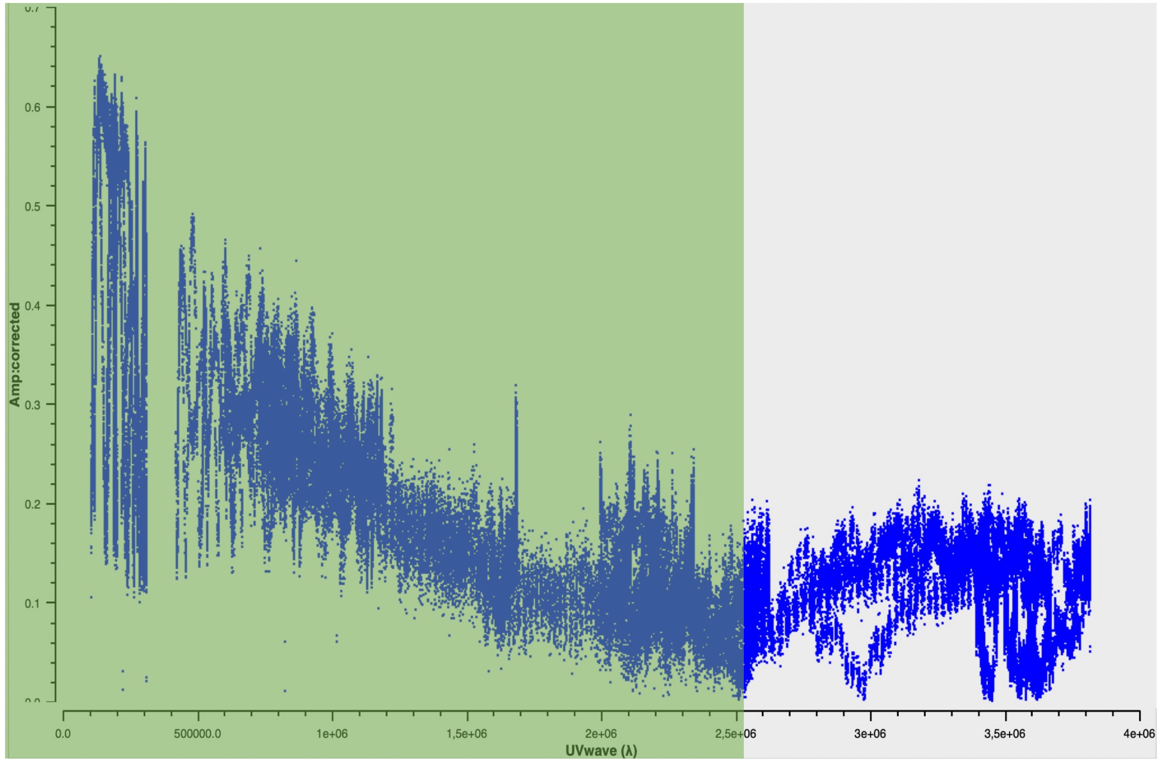
Multi-scale = clean components that can be «extended» (e.g., Gaussian functions) on different scales



See Cornwell 2008; Rau & Cornwell 2011; Offringa & Smirnov 2016

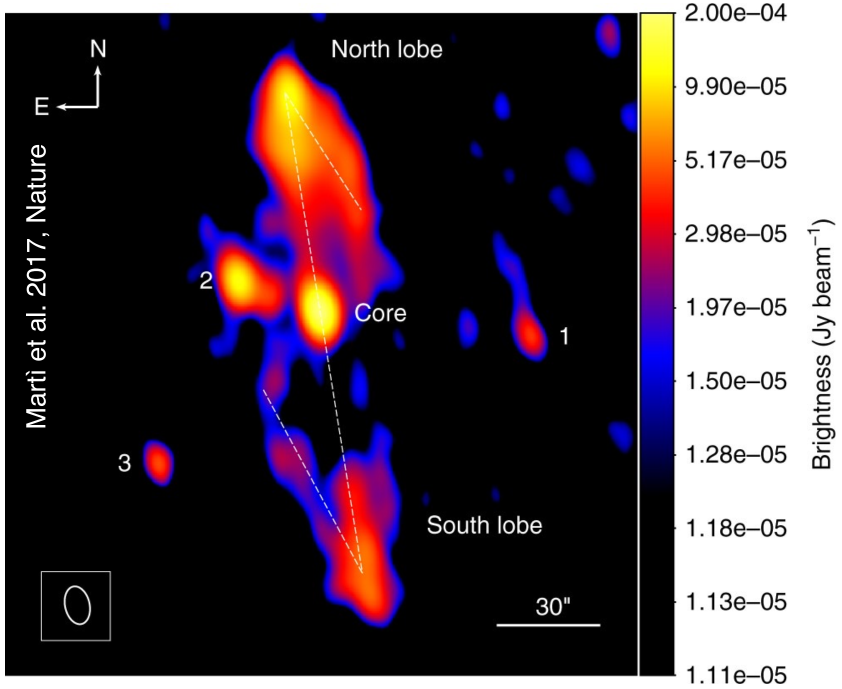
Tapering

It's like smoothing the data in the image plane
by convolving with a Gaussian PSF with FWHM corresponding to the chosen **kλ**
to some extent increases sensitivity to medium-large angular scale structures



uvtaper = [‘2.5 Mλ’] (tclean)

circular taper of HWHM=25 Mega-lambda



Tapering on 25 kλ was applied to the interferometric visibilities to better enhance the extended emission

References

Chapters 7 and 8 of «Synthesis imaging in radio astronomy II» (Edited by Taylor Carilli and Perley)

Campbell 2019 http://old.evlbi.org/user_guide/fov/fovSFXC.pdf

Interferometry and Synthesis in radio imaging (Thompson, Moran and Swenson) https://link.springer.com/book/10.1007/978-3-319-44431-4

Previous ERIS imaging lectures can be found here https://www.astron.nl/events/eris-2022/

Lecture on imaging by Michael Wise https://www.astron.nl/astrowiki/lib/exe/fetch.php?media=ra_uva:ra_uva_lecture8.pdf

Images in the first slide: Spingola+2018, Giovannini+2018(NatAs), Boccardi+2016, Hartley+2019, McKean+2011, Giroletti+2020, Radcliffe+2016, Johnston+2020, Kellermann+2007

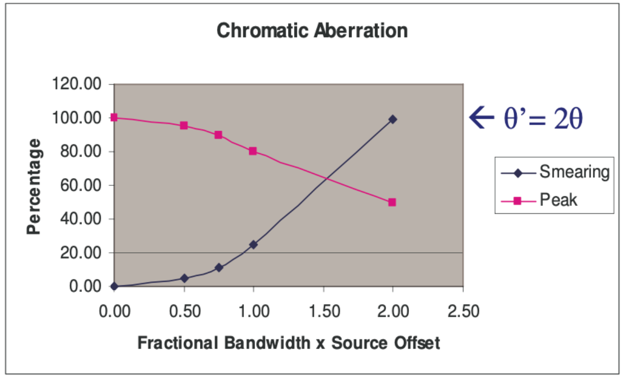
Extra slide: bandwidth and time smearing

Bandwidth smearing

- Bandwidth smearing (chromatic aberration) will produce radial smearing and reduction in source peak

- Parameterized by the product of the fractional bandwidth and the source offset in synthesised beamwidths

$$\delta\upsilon / \upsilon_0 \;\; x \;\; \theta/\theta_{HPBW}$$



- Can be alleviated by observing and imaging in spectral line mode with many narrow frequency channels gridded separately prior to Fourier inversion – reduces $\boldsymbol{\delta\upsilon}$

- Detailed form of response depends on individual channel bandpass shapes.

Time smearing

- Time-average smearing (de-correlation) will produce tangential smearing

- In general cannot be easily parameterized. At Declination=+90° a simple case exists where the effects can be parameterized by the equivalent product:

$$\omega_e \delta t_{int} \;\; x \;\; \theta/\theta_{HPBW}$$

Where $\boldsymbol{\omega_e}$ is the Earth's angular rotation rate and $\boldsymbol{\delta t_{int}}$ is the integration time interval in the dataset

- For other Declinations the effects are more complicated. However they can be alleviated by ensuring that $\boldsymbol{\delta t_{int}}$ is small enough such that there at least 4 samples per turn assuming a maximum rate of $\boldsymbol{\theta/\theta_{HPBW}}$ turns in 6 hours