Principles of Interferometry & Synthesis Imaging

Sheng-Jun Lin (ASIAA)
EA ARC Node, Taiwan

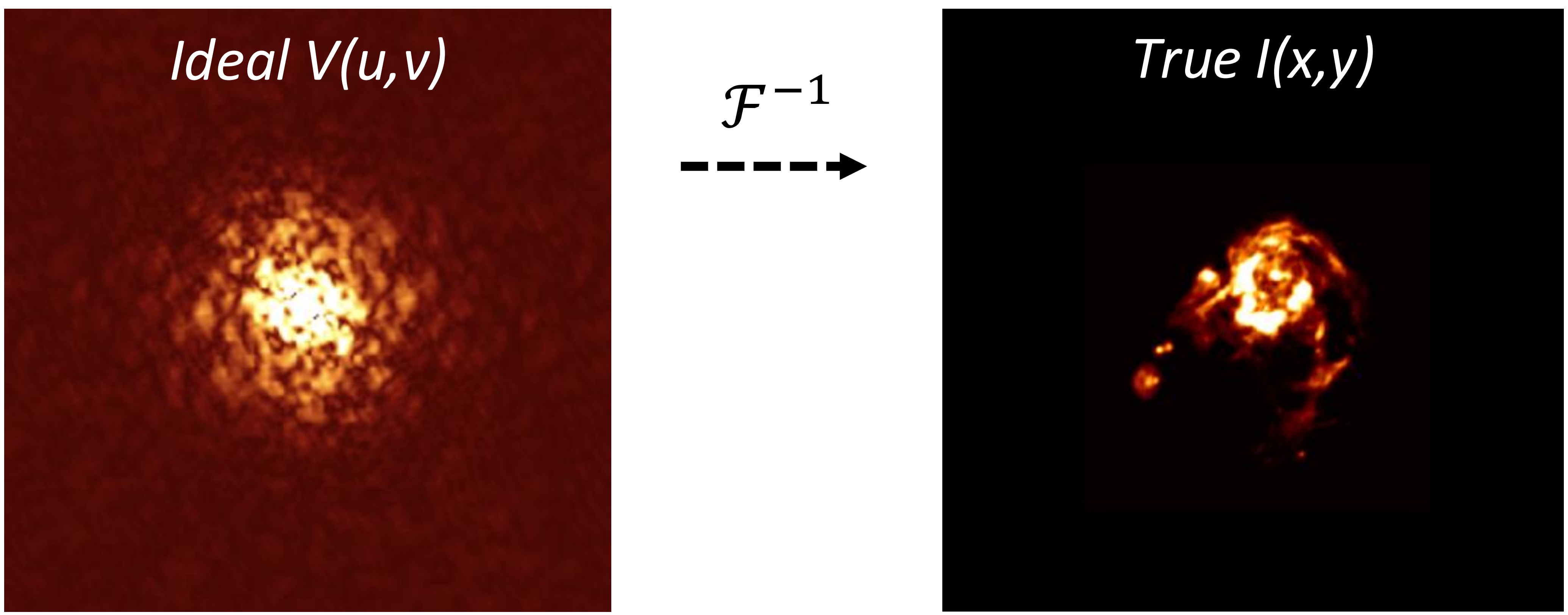
ALMA imaging workshop @ ASIAA 2026/09/02

CASA Imaging

Interferometric observations

Imaging the visibility data

- CASA plotms task: View the visibility data
- CASA tclean task: Image the visibility data
- Ideally, we obtain the image by $\mathcal{F}^{-1}[V(u,v)] = I(x,y)$

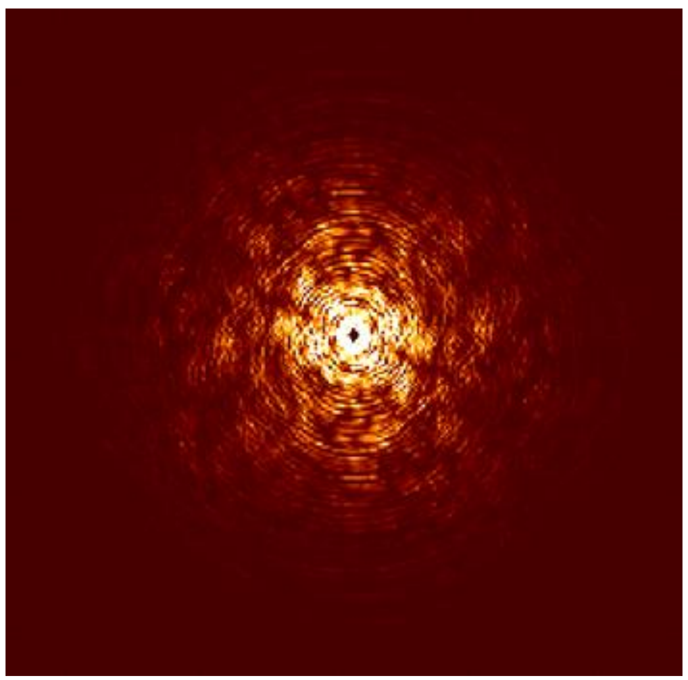


Interferometric observations

V(u, v) is not measured for all (u, v)…

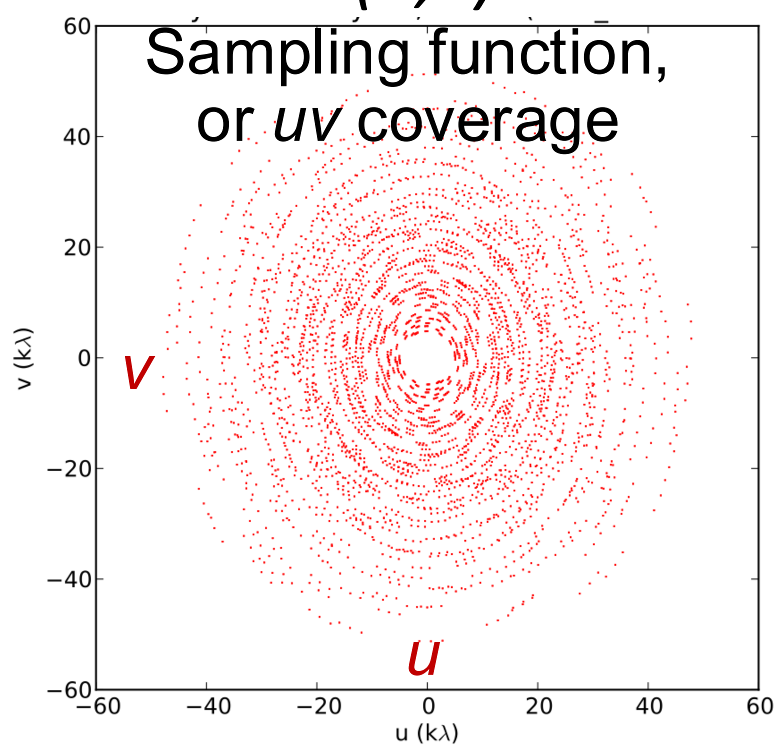
- Practically, $V_{\mathrm{obs}}(u, v) = S(u, v)V(u, v)$

Measured $V_{obs}(u,v)$



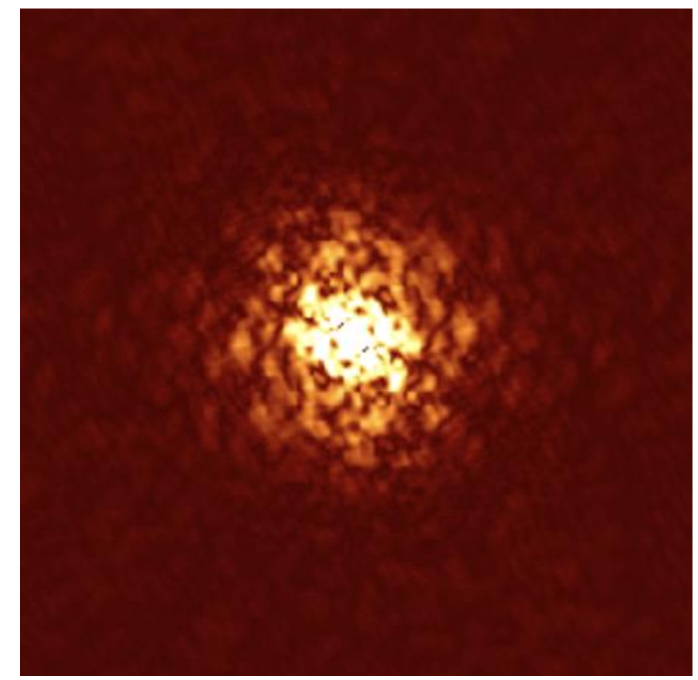
$=$

S(u,v)



$\times$

Ideal V(u,v)

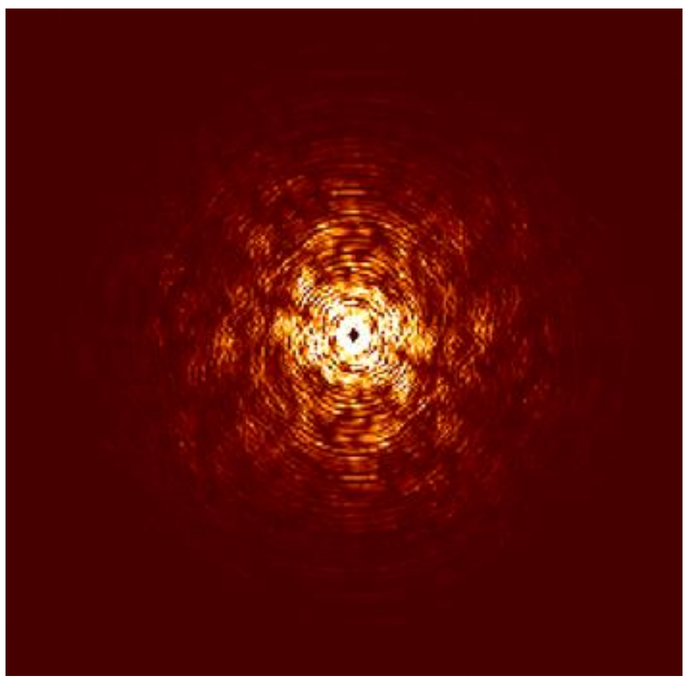


Interferometric observations

V(u, v) is not measured for all (u, v)…

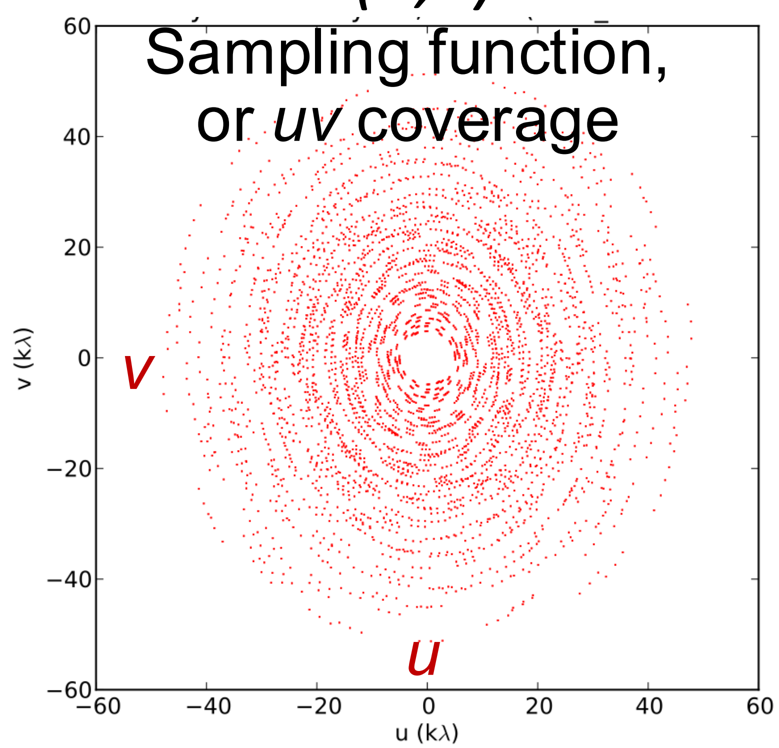
- Practically, $V_{\mathrm{obs}}(u,v) = S(u,v)V(u,v)$

Measured $V_{obs}(u,v)$



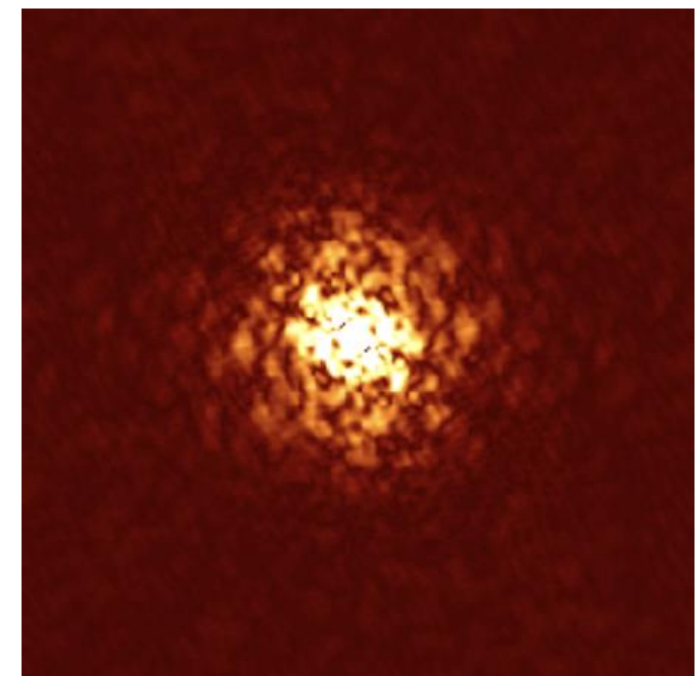
=

S(u,v)



×

Ideal V(u,v)



$$\Rightarrow \mathcal{F}^{-1}[V_{\mathrm{obs}}(u,v)] = \mathcal{F}^{-1}[S(u,v) \times V(u,v)]$$
$$= \mathcal{F}^{-1}[S(u,v)] * \mathcal{F}^{-1}[V(u,v)]$$

||

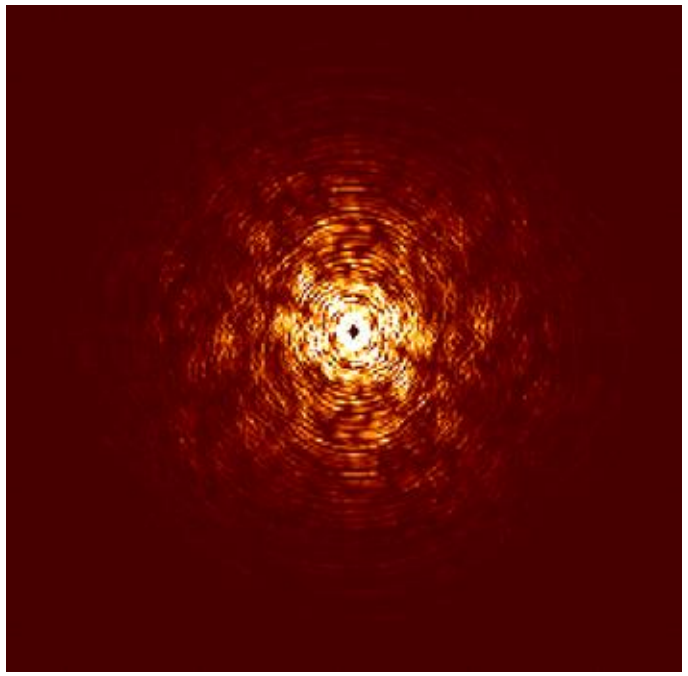
Convolution

True I(x,y)

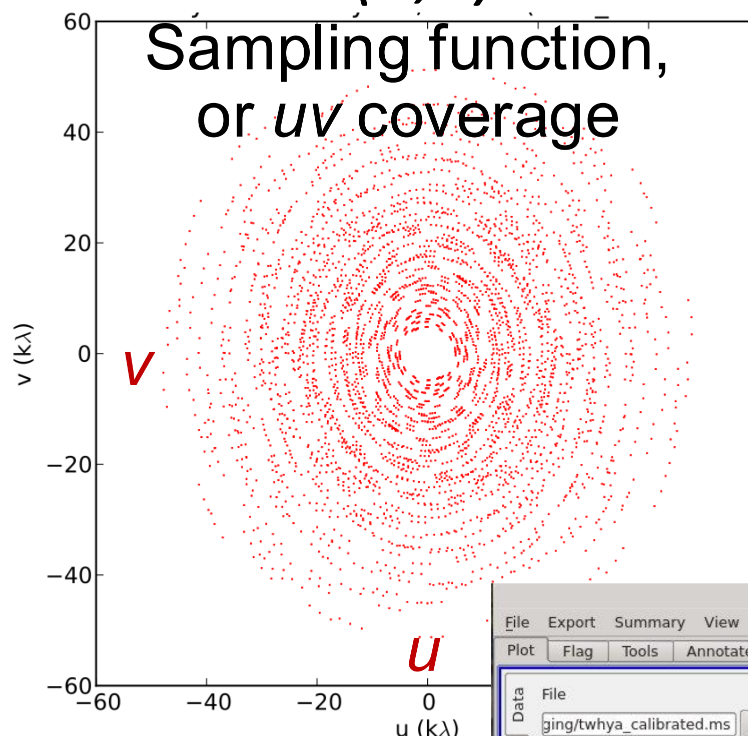
S(u, v) is not sampled
in a regular grid for V(u,v) either!

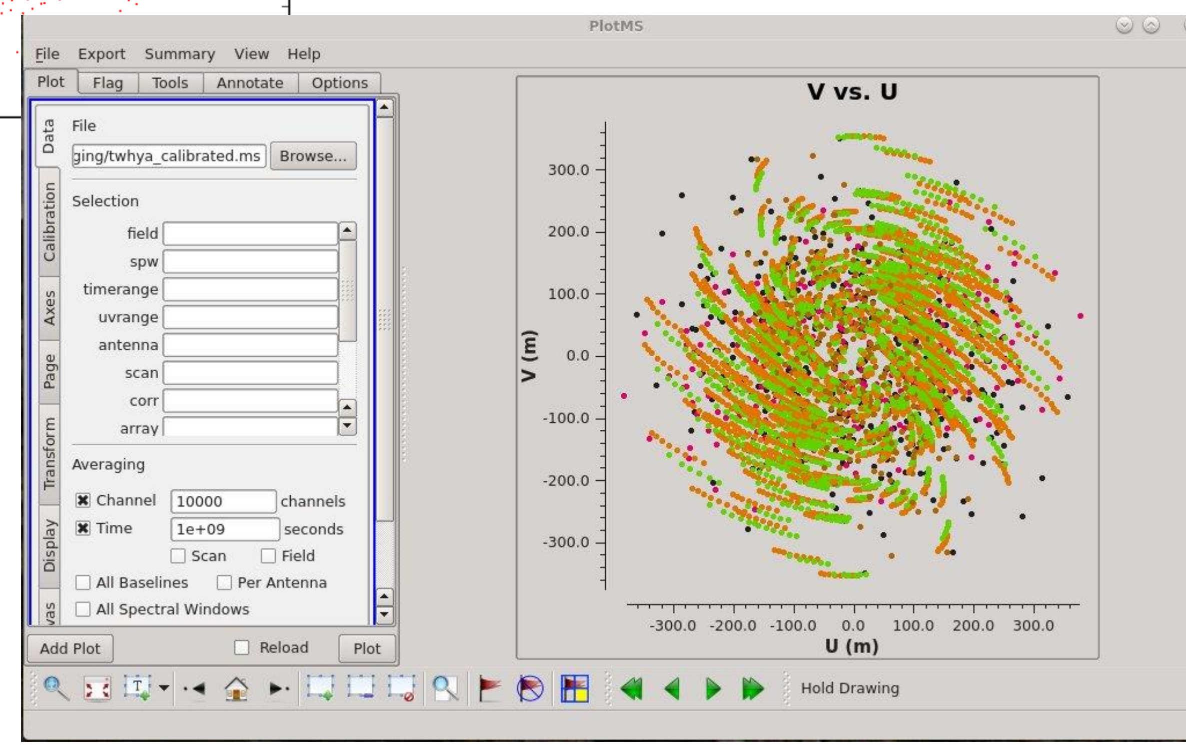
Interferometric observations

Measured $V_{obs}(u,v)$



$S(u,v)$



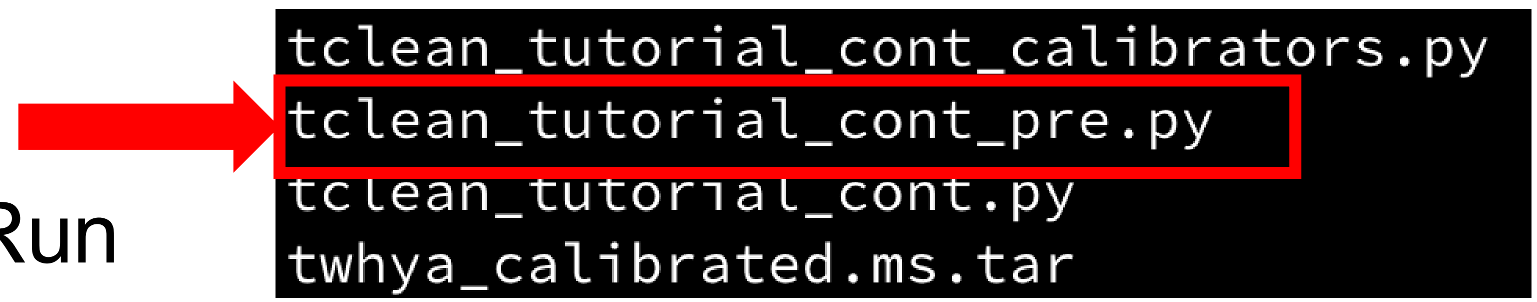


- CASA plotms task: View the visibility data

Task 1: Data Inspection & UV Coverage

First Look at Imaging CASA 6.6.6
(https://casaguides.nrao.edu/index.php/First_Look_at_Imaging_CASA_6.6.6)

Understand the basic properties of the TW Hya dataset and visualize the u,v visibility



1. Check Observation Log: Run
listobs(vis='twhya_calibrated.ms')

2. Plot UV Coverage: Run
plotms(vis='twhya_calibrated.ms', xaxis='u', yaxis='v')

3. Inspect Visibility Amplitude of the flux/other calibrators: Run
plotms(xaxis='uvdist', yaxis='amp')

Test dataset

```
tclean_tutorial_cont_calibrators.py
tclean_tutorial_cont_pre.py
tclean_tutorial_cont.py
twhya_calibrated.ms.tar
```

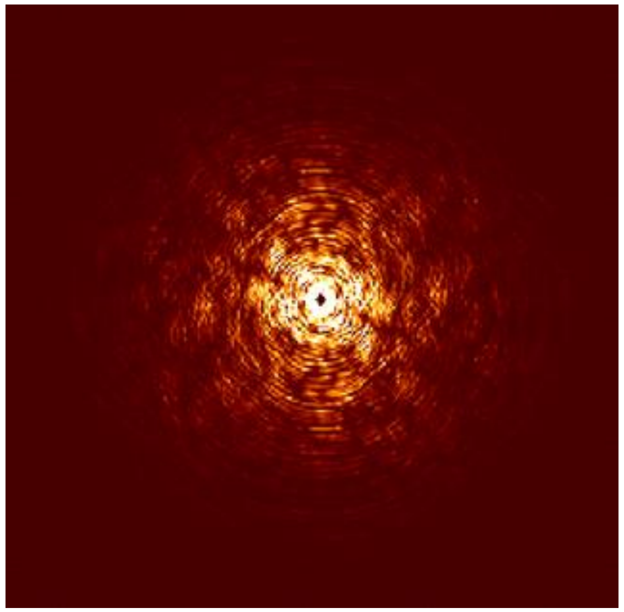
- inside CASA: execfile('tclean_tutorial_cont_pre.py')

```
listobs(vis='twhya_calibrated.ms', listfile='twhya_listobs.txt')

split(vis='twhya_calibrated.ms', field='5', width='8',
        outputvis='twhya_smoothed.ms', datacolumn='data')

listobs(vis='twhya_smoothed.ms', listfile='twhya_listobs_smoothed.txt')

plotms(vis='twhya_calibrated.ms',
       xaxis='u',
       yaxis='v',
       avgchannel='10000',
       avgspw=False,
       avgtime='1e9',
       avgscan=False,
       coloraxis='field',
       showgui=True)
```

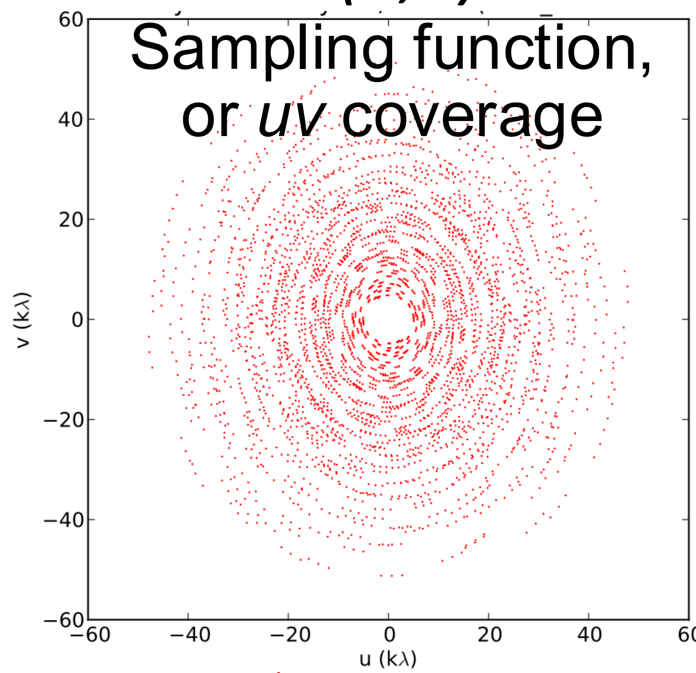
From Visibility to Image: $\mathcal{F}^{-1}$

Measured $V_{obs}(u,v)$ | *$S(u,v)$* | *Ideal $V(u,v)$*

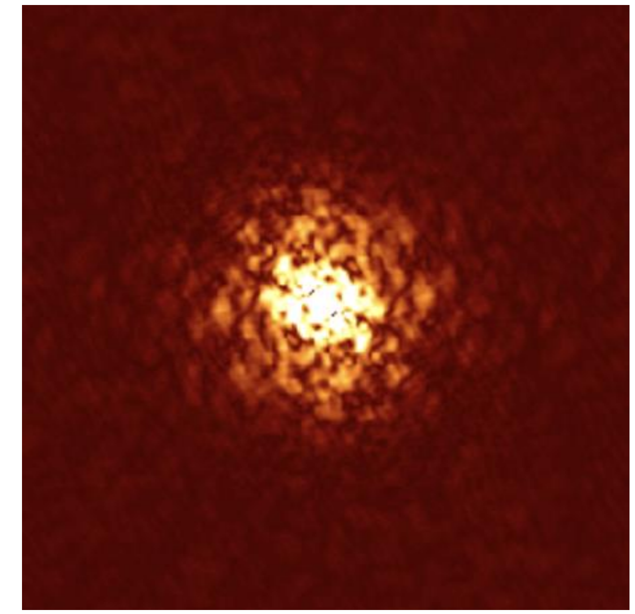
u, v domain:



=



×

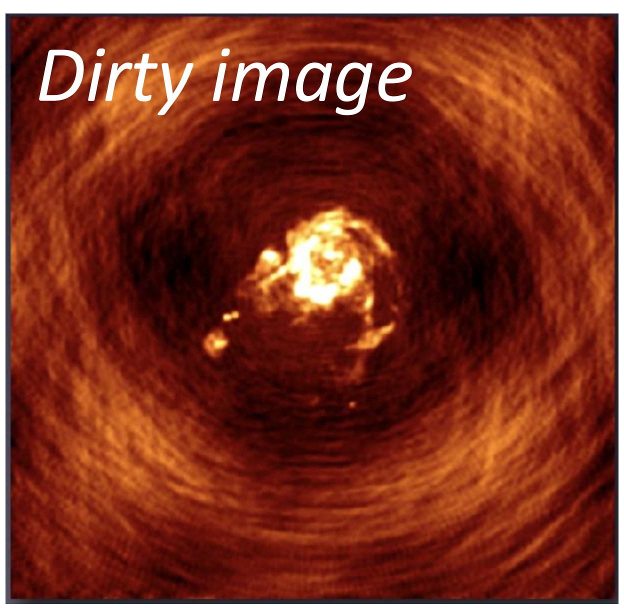


$\mathcal{F}^{-1}[V_{\text{obs}}(u,v)] = I_{dirty}(x,y)$

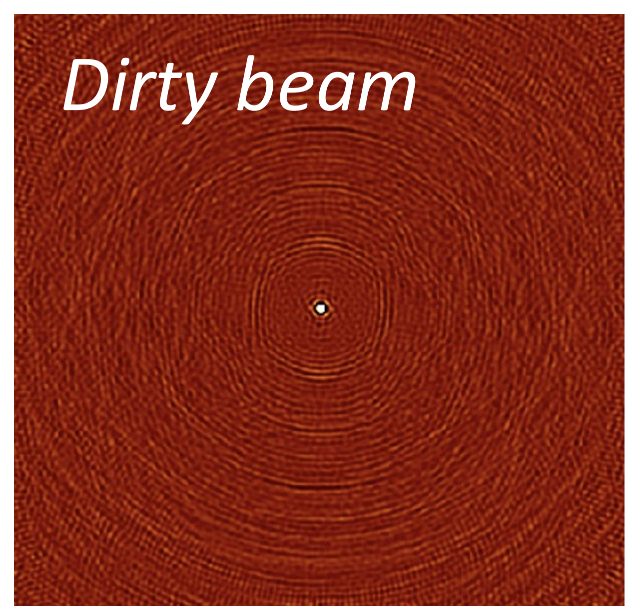
$\mathcal{F}^{-1}[S(u,v)] = B_{dirty}(x,y)$

$\mathcal{F}^{-1}[V(u,v)] = I_{true}(x,y)$

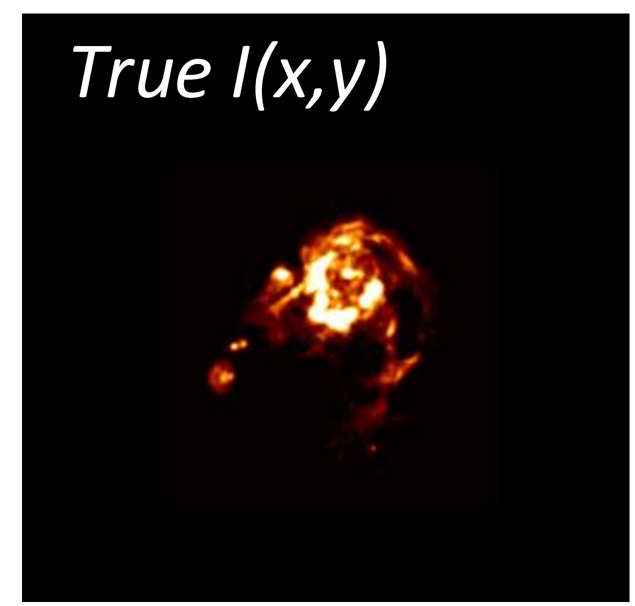
x, y domain:



=



$*$



To recover *True I(x,y)*, we must **deconvolve** *Dirty beam* from *Dirty image*.

Imaging parameters

Handling the uneven visibility sampling

Fourier *uv* domain	Image *xy* domain

- Weighting $S(u,v)$ for re-sampling

  tclean/ "weighting" parameter

- Gridding $V_{\rm obs}(u,v)$ and $S(u,v)$ for FFT

  tclean/ "gridder" parameter

- Pixel size of the final image

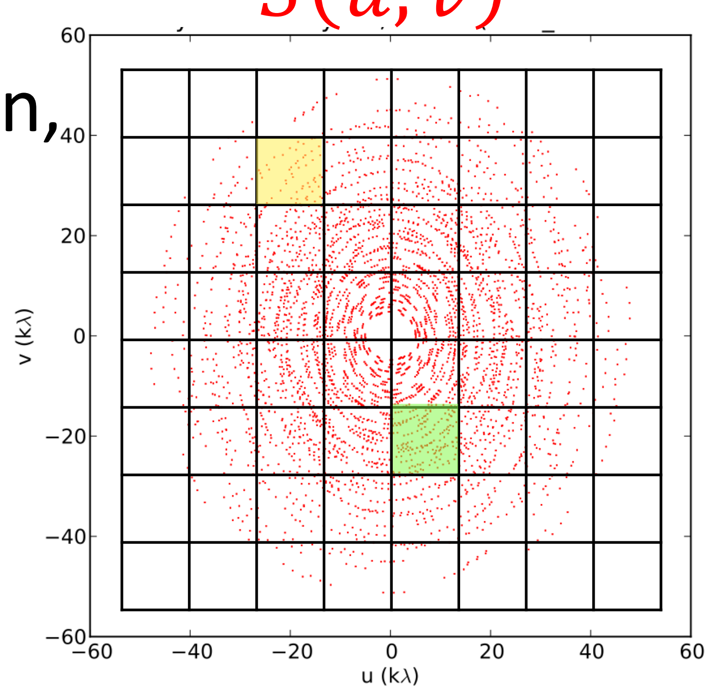
  tclean/ "cell" parameter

- Final image size

  tclean/ "imsize" parameter

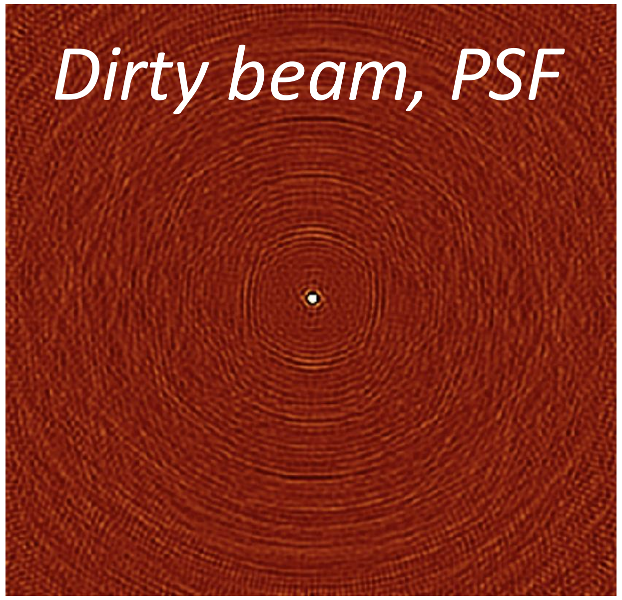
Sampling function, or *uv* coverage

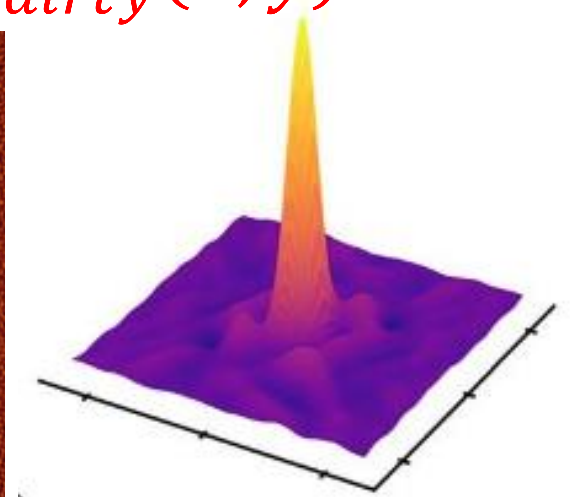
$S(u,v)$



$\xrightarrow{\mathcal{F}^{-1}}$

$\mathcal{F}^{-1}[S(u,v)] = B_{dirty}(x,y)$



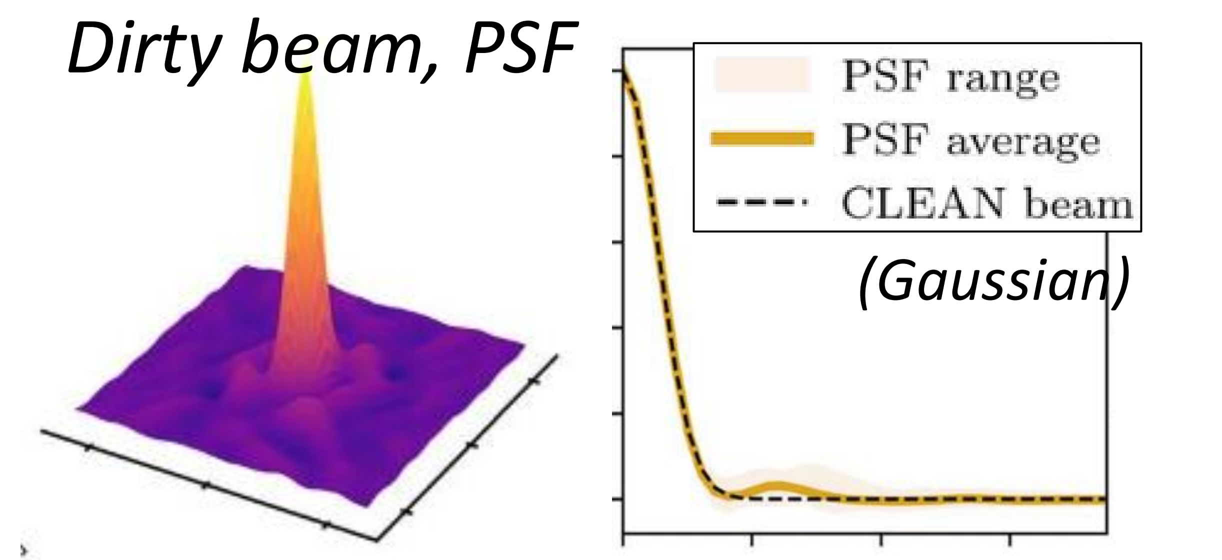


Czekala (2021)

Imaging parameters

tclean "cell" and "imsize" parameters

- EM wave $\lambda = c/\nu$
- Pixel size "cell" satisfies the Nyquist sampling for the longest baseline
  - Beam $\theta \sim \frac{\lambda}{b_{max}}$
  - (>2) 3, 5, or 7 pixels across the **dirty beam** main lobe (="**clean beam**")



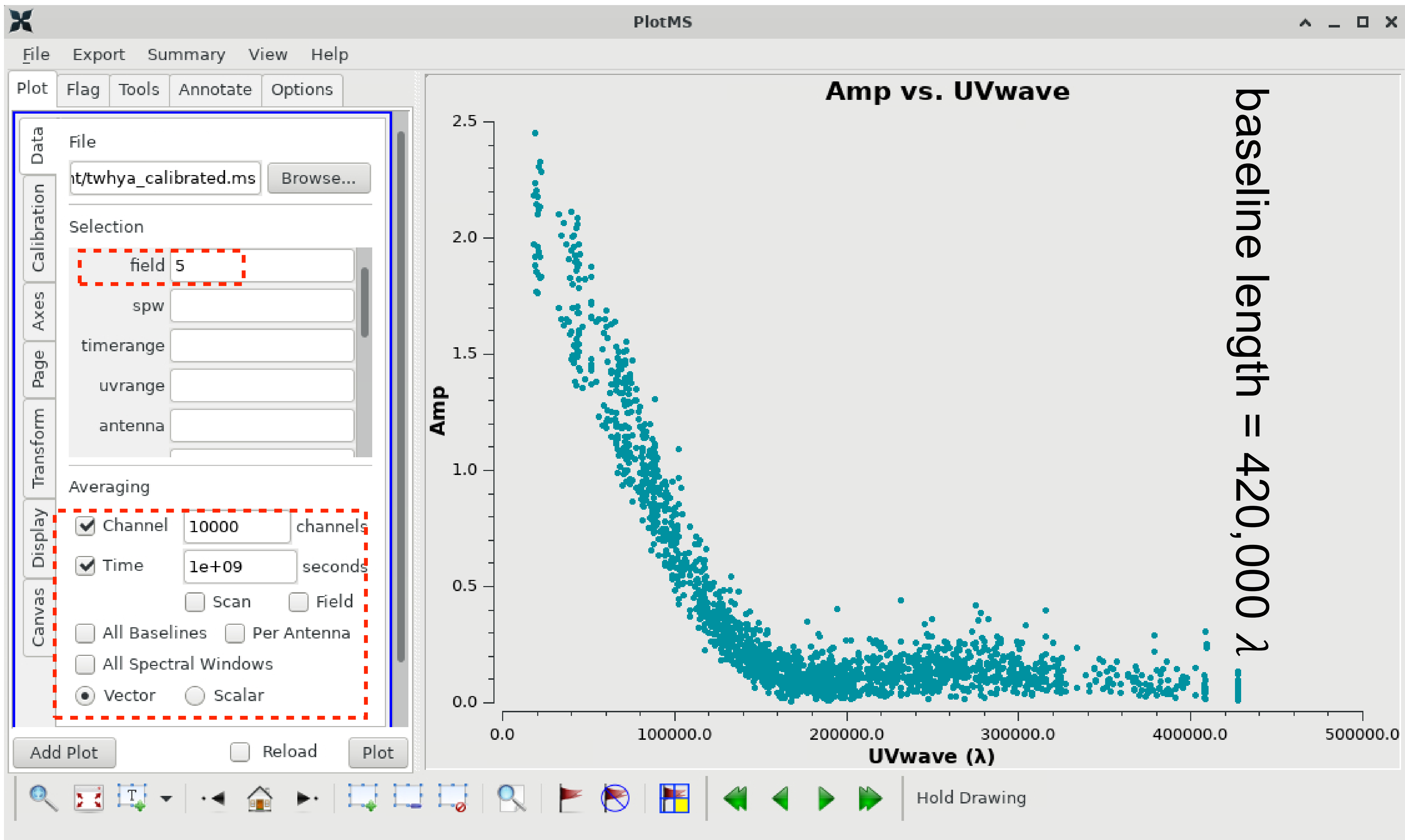
Czekala (2021)

- Image size "imsize" ≥ 2 × Field of view ("primary beam")
  - FOV $\sim 1.22\frac{\lambda}{D}$, where D is the dish diameter (12 m or 7 m)

Determine cell size and image size

simple math…

- Cell size (aka pixel size) is related to the clean beam size. Usually, we use 5~7 pixels to sample the short axis of the elliptical beam.



Estimate cell size and image size

simple math…

speed of light

frequency

$$c = \lambda\nu$$

wavelength

$$\lambda = \frac{c}{\nu} \longrightarrow 0.8\ \text{mm}$$

angular resolution (aka "beam")

$$\theta \sim \frac{\lambda}{B} \longrightarrow$$ 1/420,000 rad=0.5" → cell='0.1arcsec'

baseline length = 420,000 λ

field of view (aka "primary beam")

$$\text{FOV} \sim 1.22\frac{\lambda}{\text{D}} \longrightarrow 17'' \longrightarrow$$ imsize=[340]

dish size

2x FOV

Imaging parameters

tclean “weighting” parameter

Visibility weighting schemes

- Natural: Maximizes the sensitivity, degrades angular resolution
- Uniform: Best angular resolution, reduced point source sensitivity
- Robust: Smooth, tunable combination of Natural (robust=+2) & Uniform (-2)
- Taper: Similar to smoothing, degrades resolution

Task 2: Dirty Image & Weighting

Make dirty images (niter=0) for the bandpass calibrator

1. Use different weighting schemes

- Natural: Maximizes the sensitivity, degrades angular resolution
- Uniform: Best angular resolution, reduced point source sensitivity
- Robust: Smooth, tunable combination of Natural (robust=+2) & Uniform (-2)

2. Try robust from -2, -1, 0, 1, and 2.

3. Use CARTA to check their beam sizes.

4. Use python to plot “robust parameter” vs. “beam size”

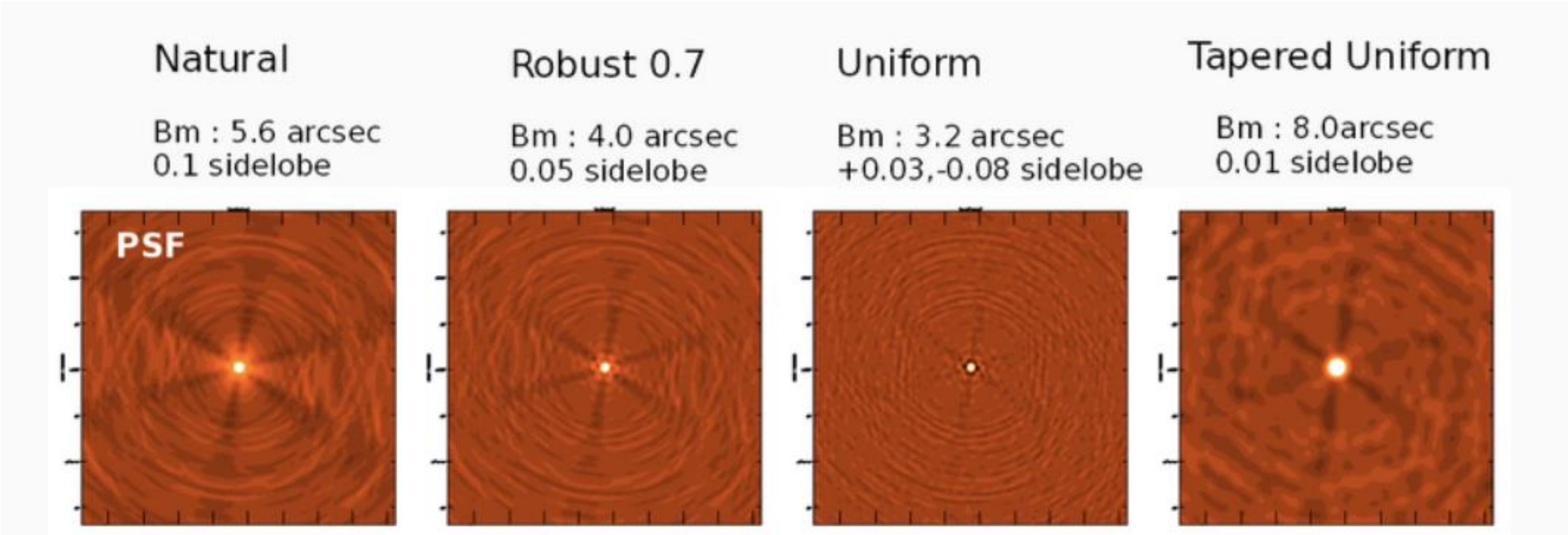
```
tclean_tutorial_cont_calibrators.py
tclean_tutorial_cont_pre.py
tclean_tutorial_cont.py
twhya_calibrated.ms.tar
```

Imaging parameters

tclean "weighting" parameter

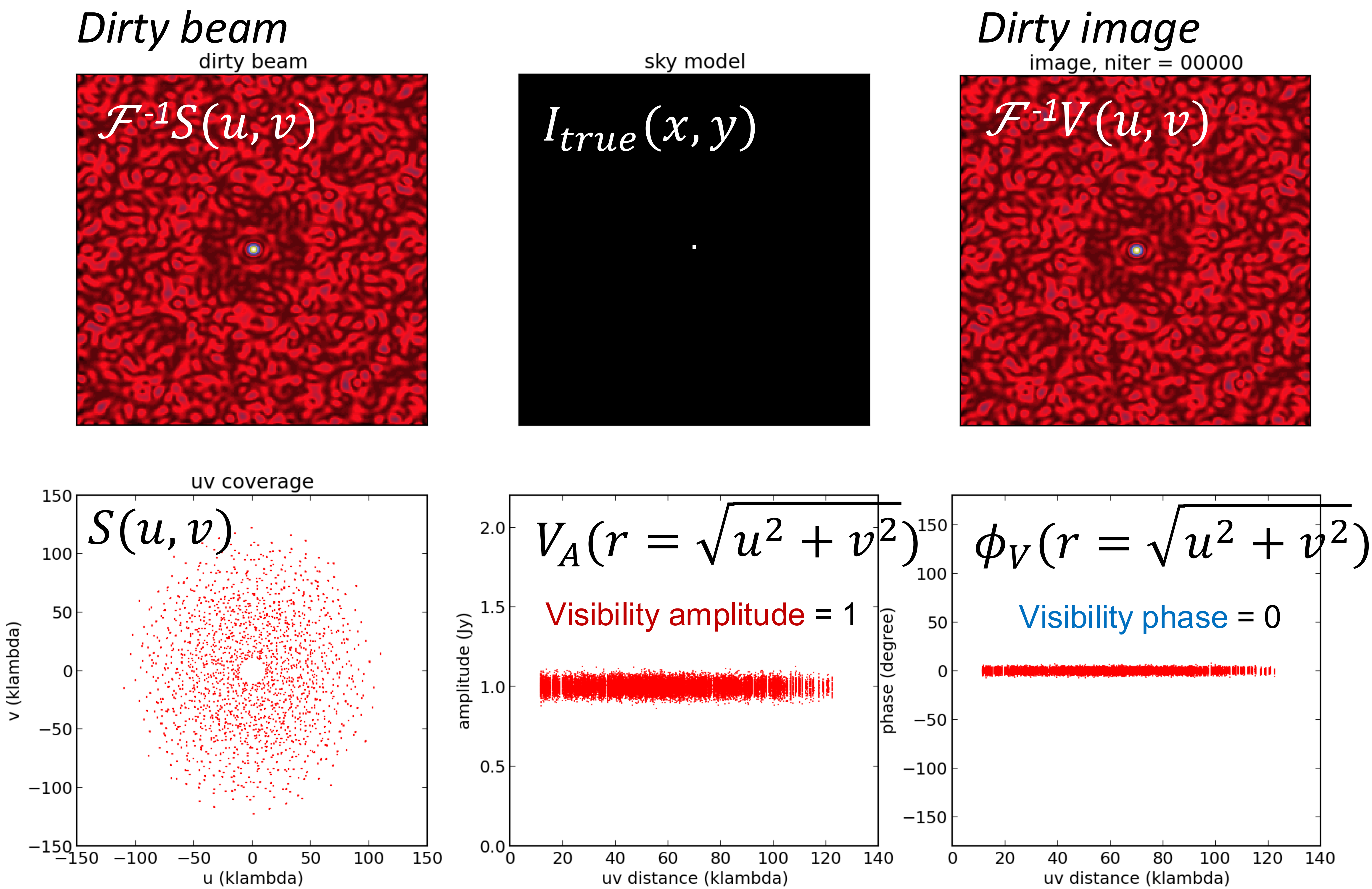
Visibility weighting schemes

- Natural: Maximizes the sensitivity, degrades angular resolution
- Uniform: Best angular resolution, reduced point source sensitivity
- Robust: Smooth, tunable combination of Natural (robust=+2) & Uniform (-2)
- Taper: Similar to smoothing, degrades resolution



Simulated visibilities and images

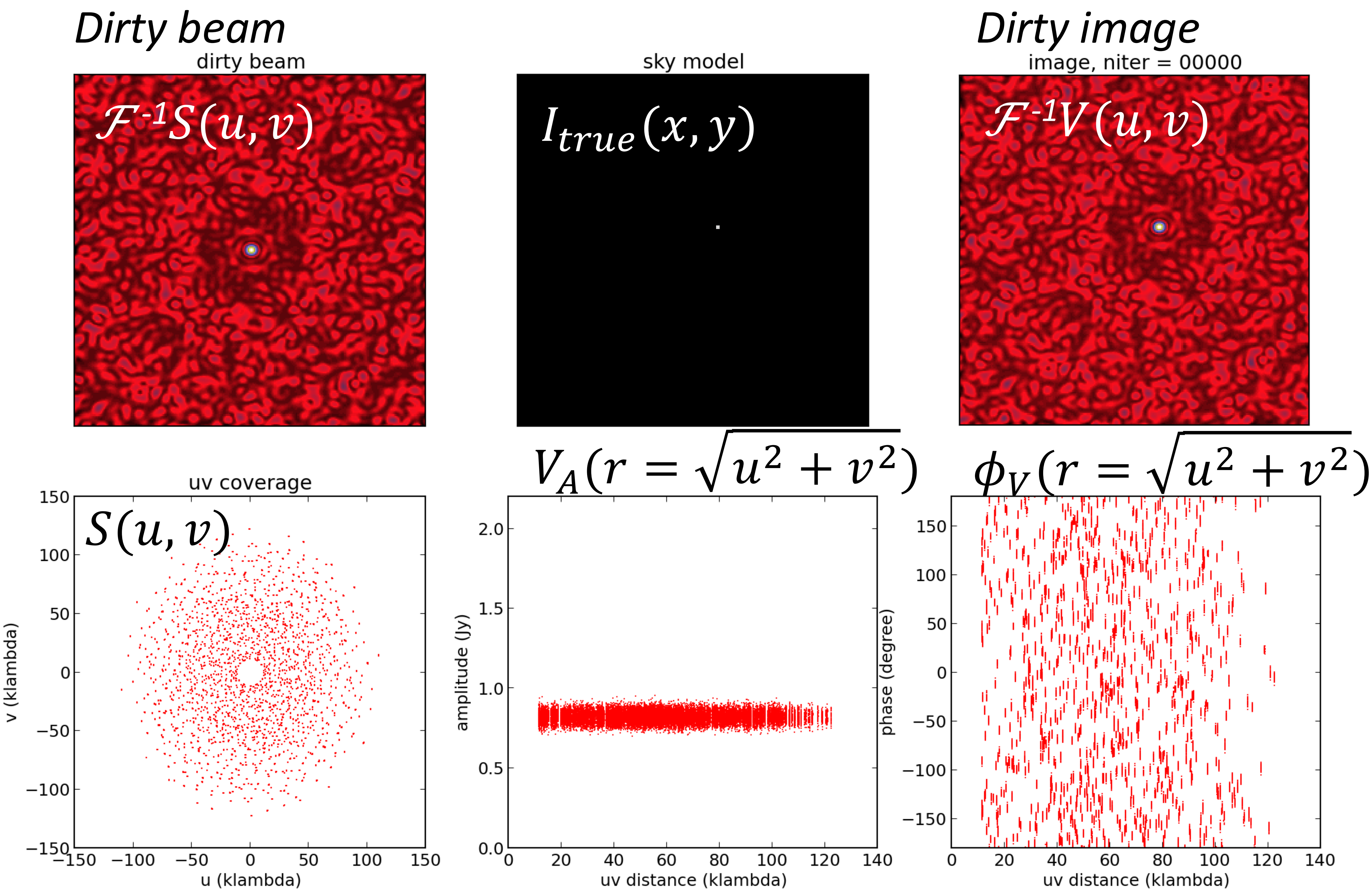
Point source



Credit: Kuo-Song Wang (ASIAA)

Simulated visibilities and images

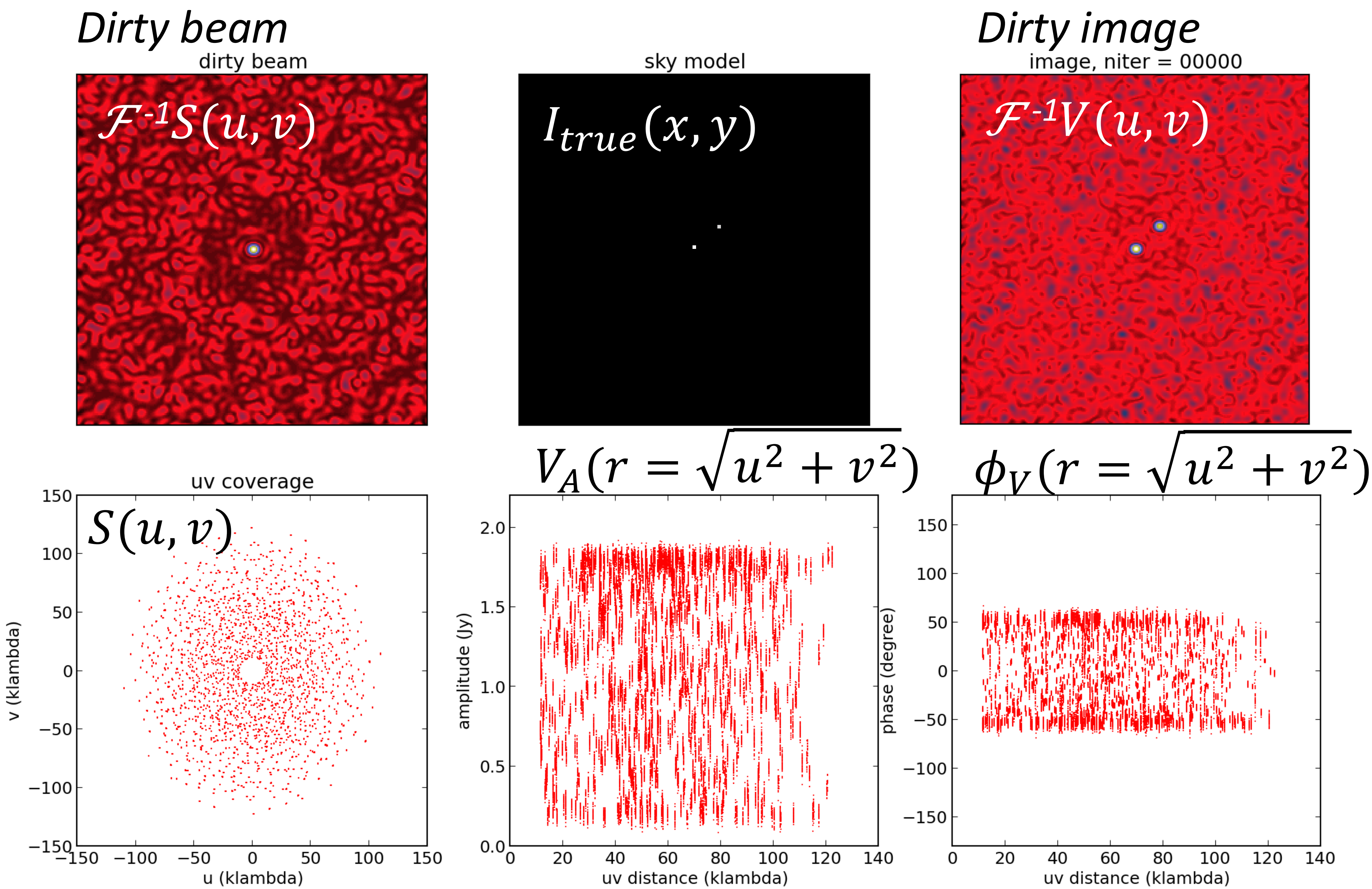
Point source (off-center)



Credit: Kuo-Song Wang (ASIAA)

Simulated visibilities and images

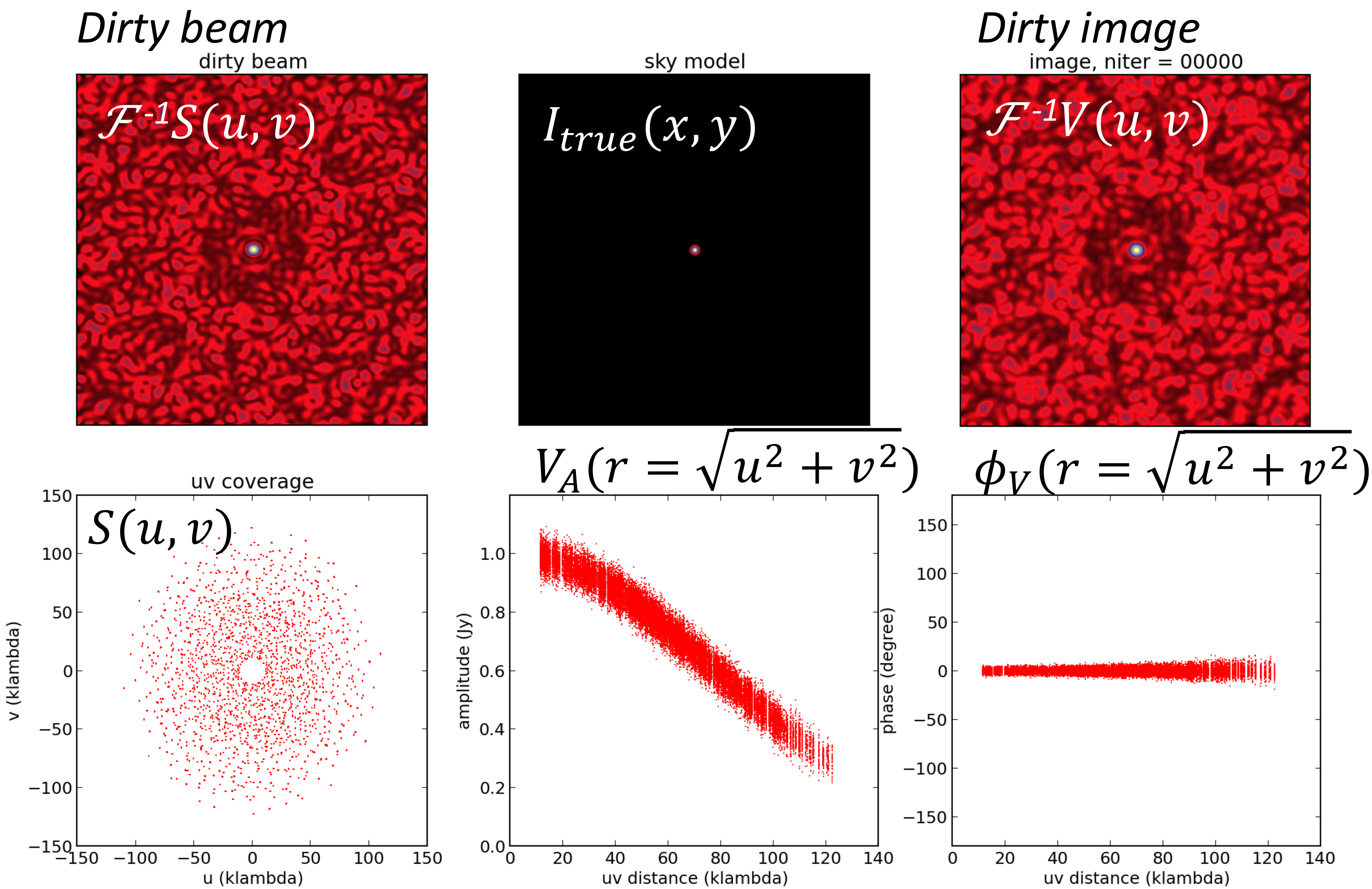
Double point sources



Credit: Kuo-Song Wang (ASIAA)

Simulated visibilities and images

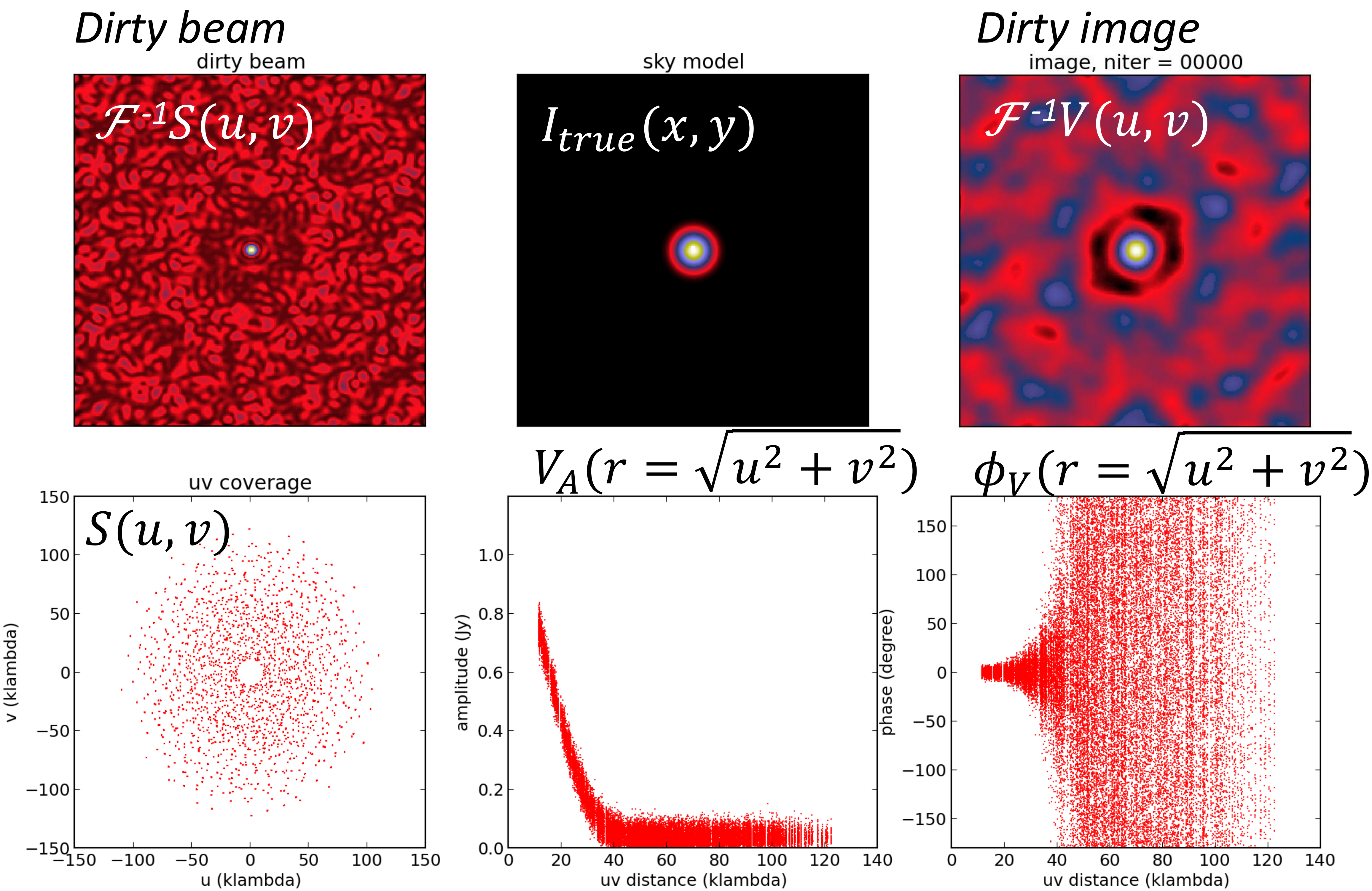
Small gaussian



Credit: Kuo-Song Wang (ASIAA)

Simulated visibilities and images

Large gaussian

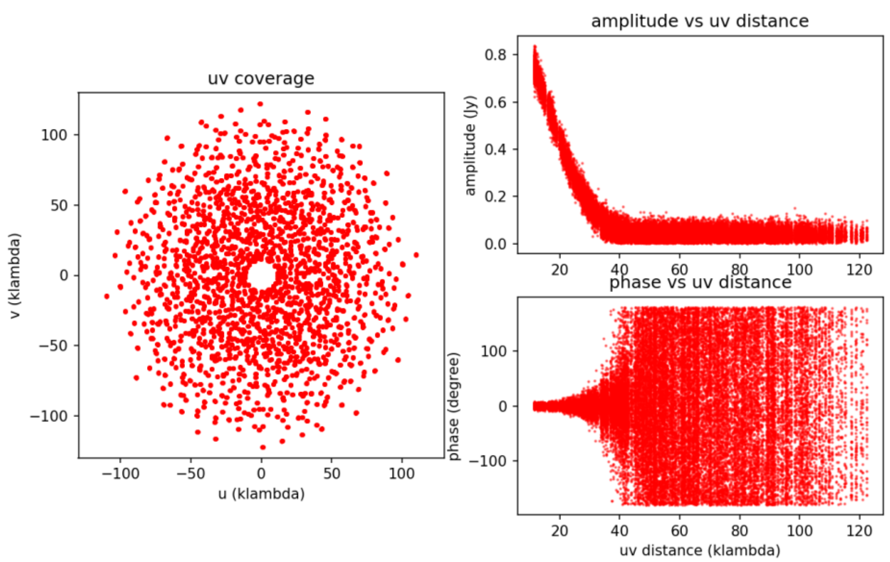


Credit: Kuo-Song Wang (ASIAA)

Deconvolution procedure

CLEAN algorithm

Inversion ($\mathcal{F}^{-1}$): Obtain *Dirty image*, $I_{dirty}(x, y)$.
Deconvolution: Perform on the image domain
Restoration: Obtain *Clean image*, $I_{clean}(x, y)$.

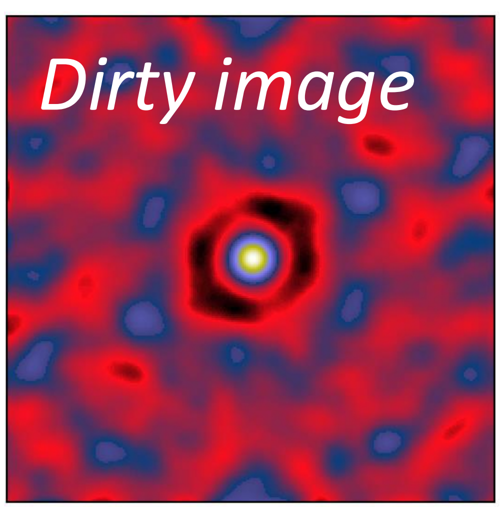


$$V_{\mathrm{obs}}(u, v) = S(u, v) V(x, y)$$

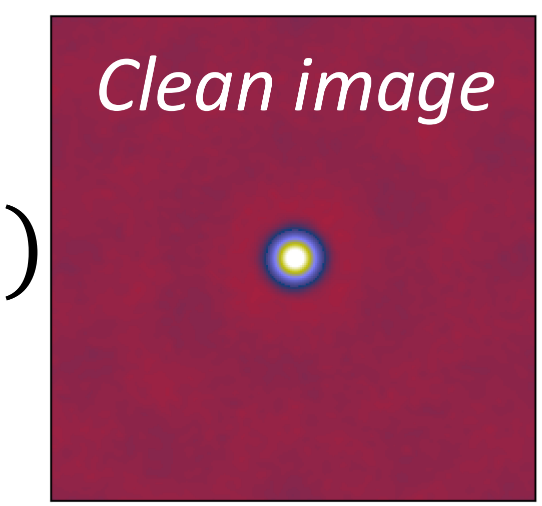
$$V_{\mathrm{obs}}(u, v) = S(u, v) \iint \boxed{I_{\mathrm{true}}(x, y)} e^{-2\pi i (ux + vy)} dxdy$$

$\mathcal{F}^{-1}$

Approximation



$$I_{dirty}(x, y) \xrightarrow{\hspace{3em}} I_{clean}(x, y)$$



Deconvolution

CLEAN

Högbom (1974): tclean/ "deconvolver='hogbom' "

Assumption: Target source consists of multiple point sources.

1. [**Inversion**] Compute dirty beam ($\mathcal{F}^{-1}[S(u,v)]$) and dirty map ($\mathcal{F}^{-1}[V(u,v)]$)

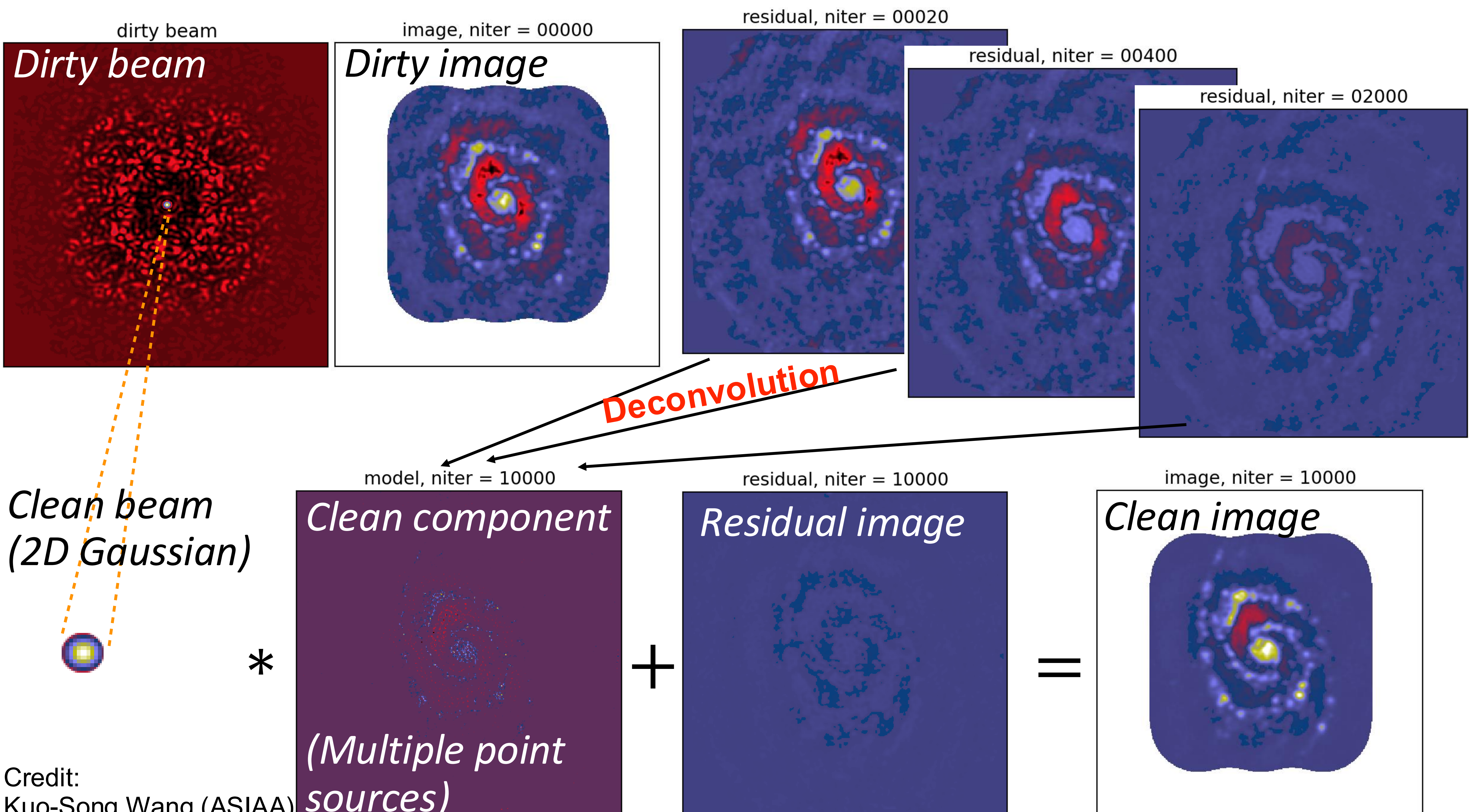
2. [**Deconvolution**] Subtract over the whole dirty map with a dirty beam pattern (multiplied by a factor, e.g., 0.1) which is centered at the point where the dirty map has its maximum of absolute value of intensity

3. Repeat step 2, each time replacing the dirty map by the remaining map (i.e., residual map) from the previous iteration

4. [**Restoration**] Once the iteration is satisfied by
(a. tclean/ "niter") reaching the iteration limit or
(b. tclean/ "threshold") that the brightness in the remaining map is lower than a threshold,
generate the clean map by adding the final residual map with the clean component map convolved with the clean beam

CLEAN

As an iterative process

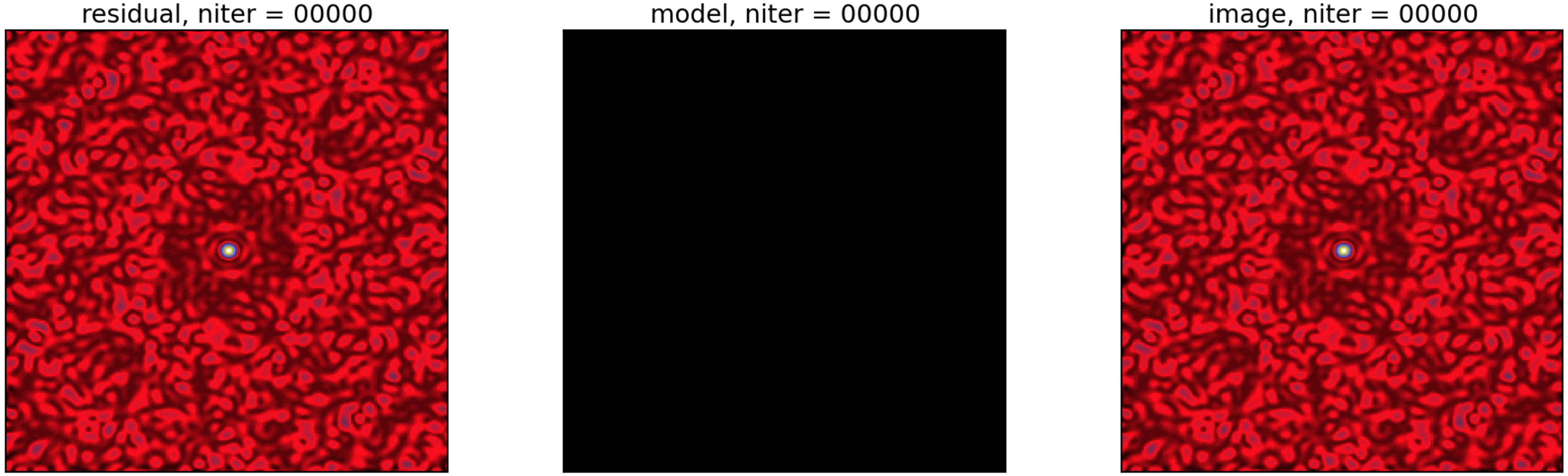


Credit:
Kuo-Song Wang (ASIAA)

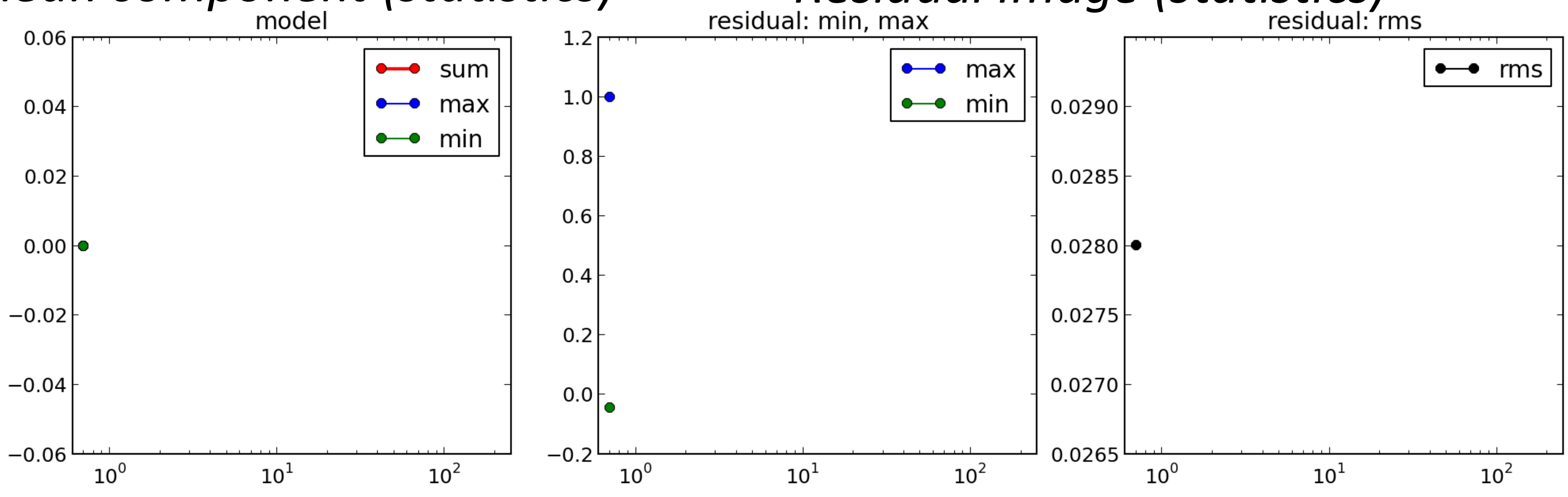
CLEAN examples

Point source

Residual image *Clean component* *Clean image*



Clean component (statistics) *Residual image (statistics)*

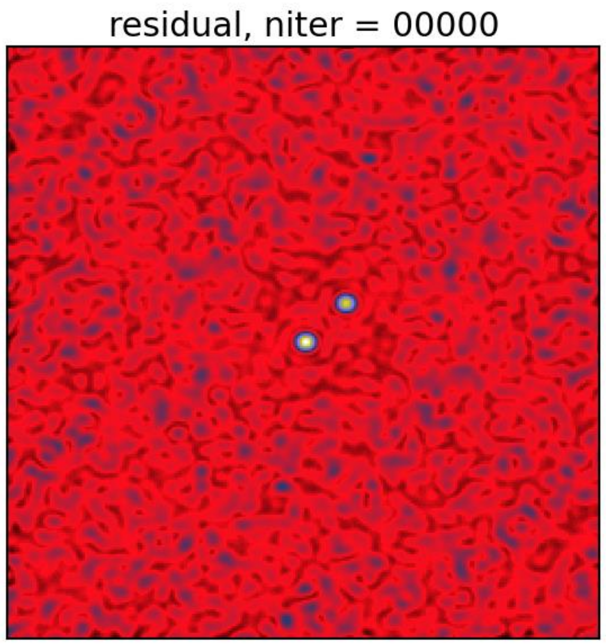


Credit: Kuo-Song Wang (ASIAA)

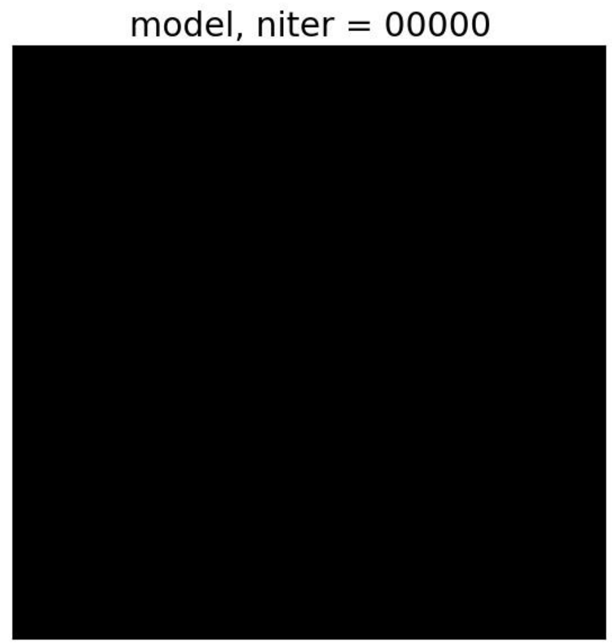
CLEAN examples

Double point sources

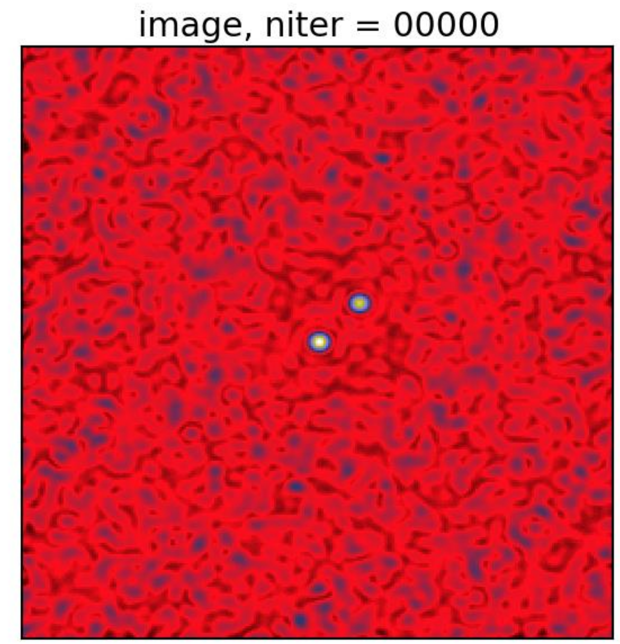
Residual image



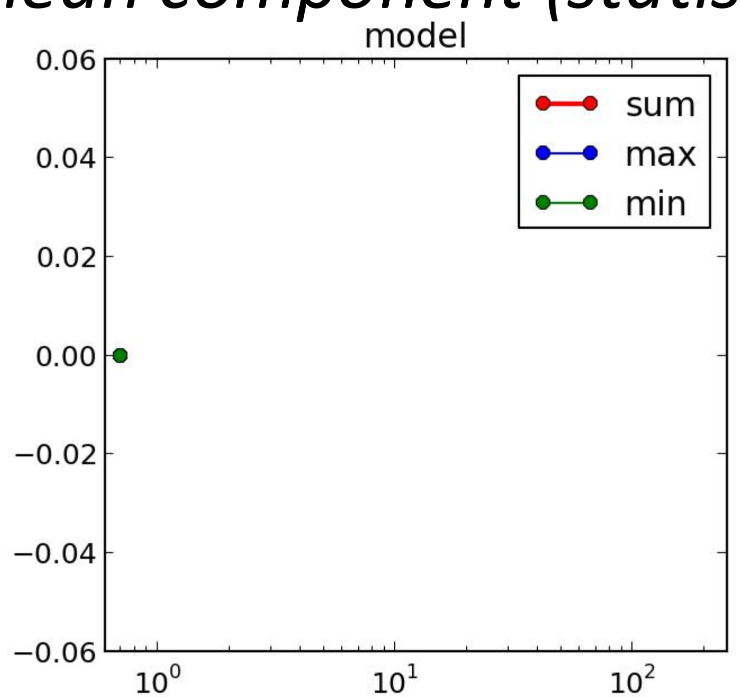
Clean component



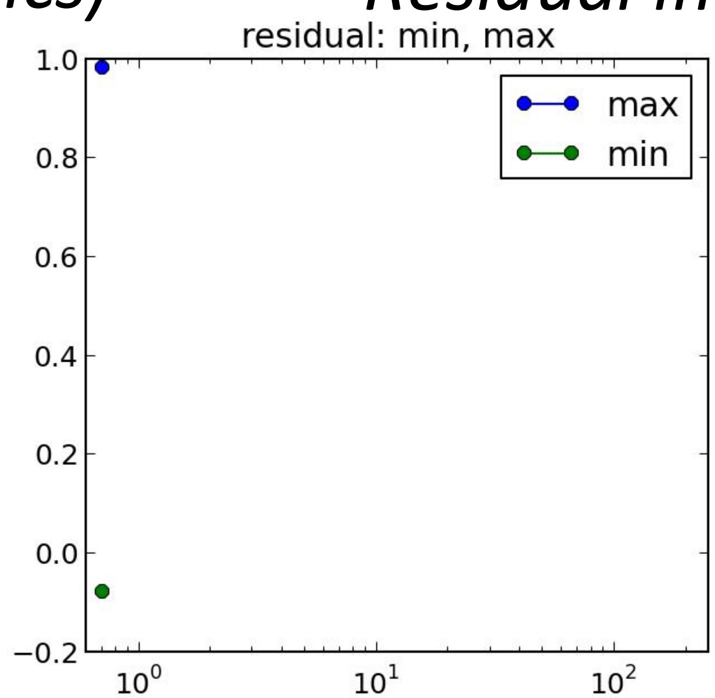
Clean image

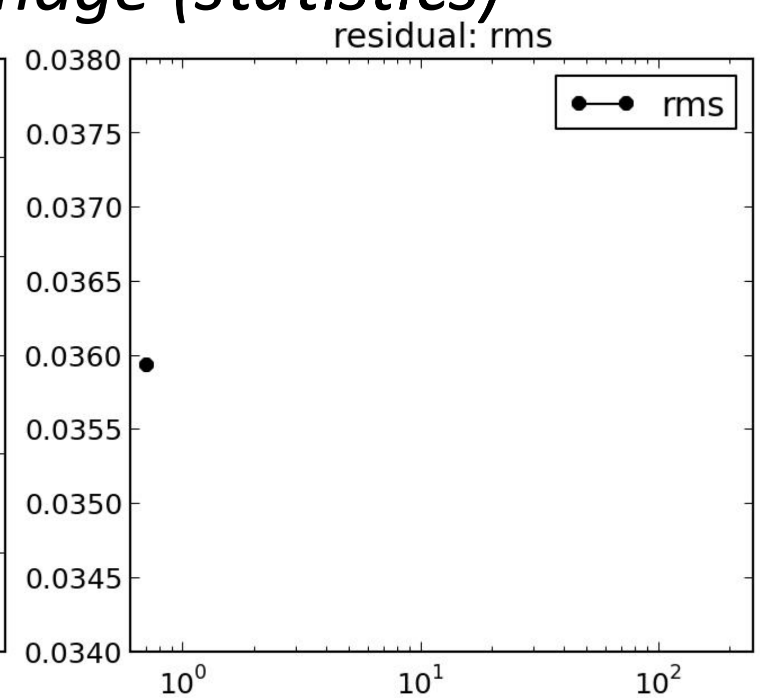


Clean component (statistics)



Residual image (statistics)



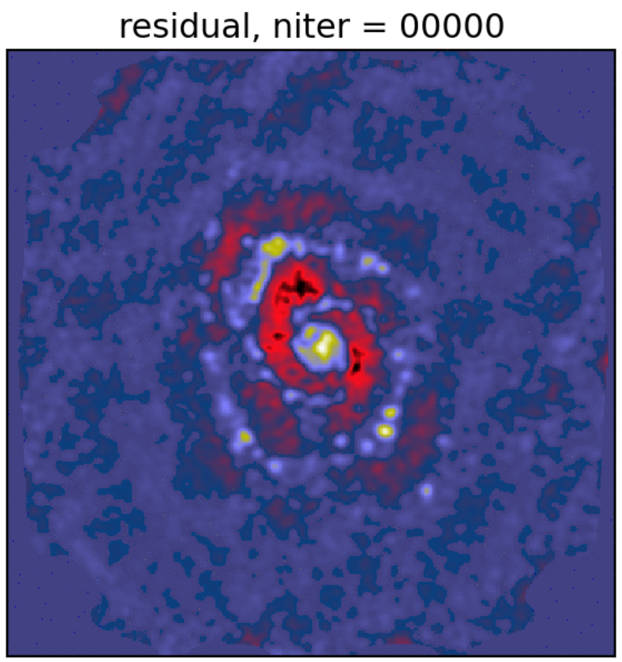


Credit: Kuo-Song Wang (ASIAA)

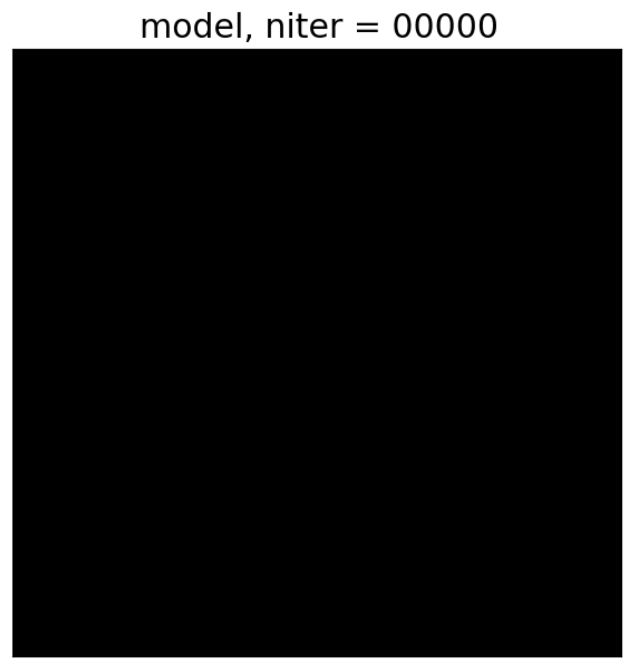
CLEAN examples

Galaxy

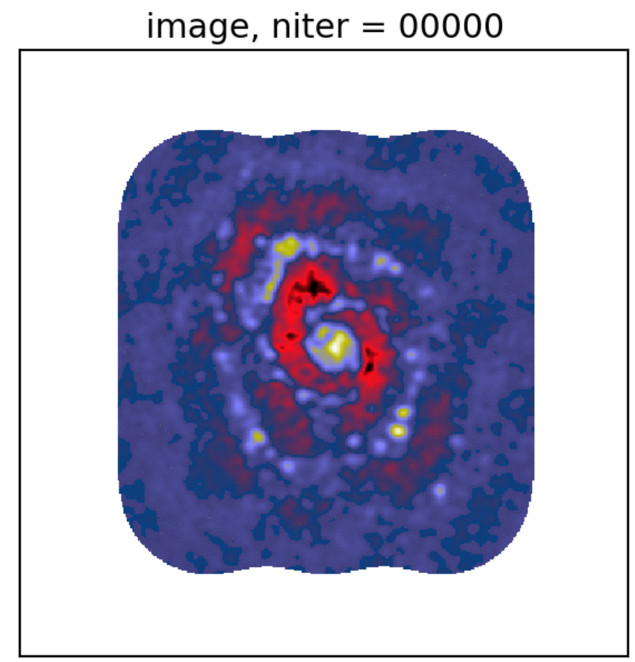
Residual image



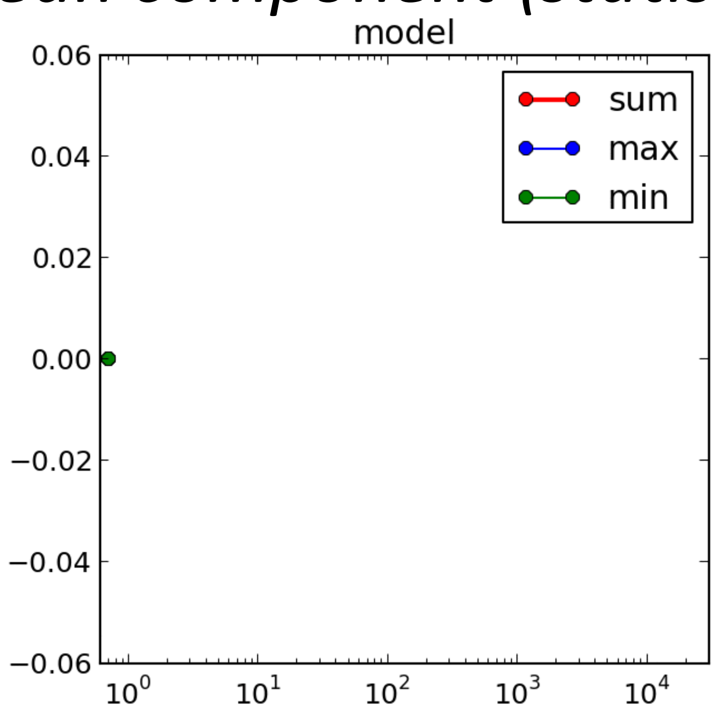
Clean component



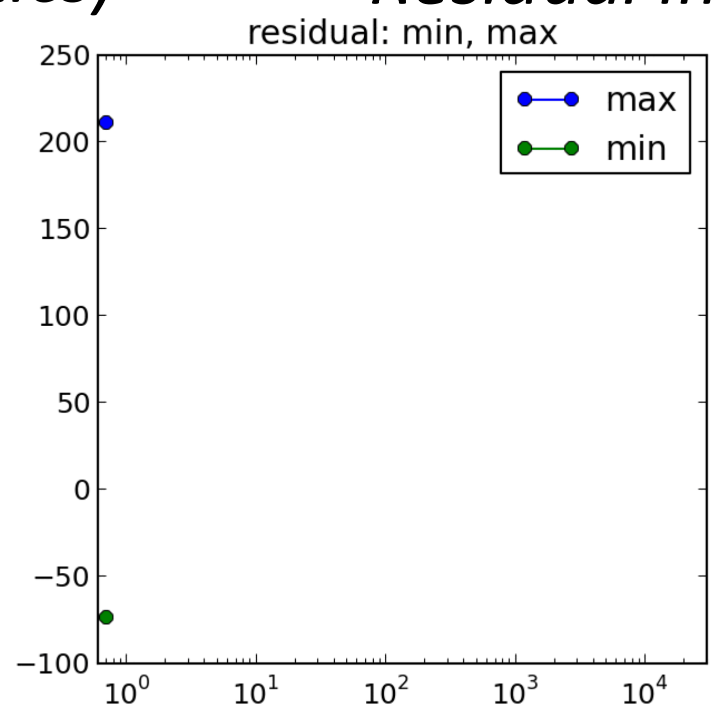
Clean image

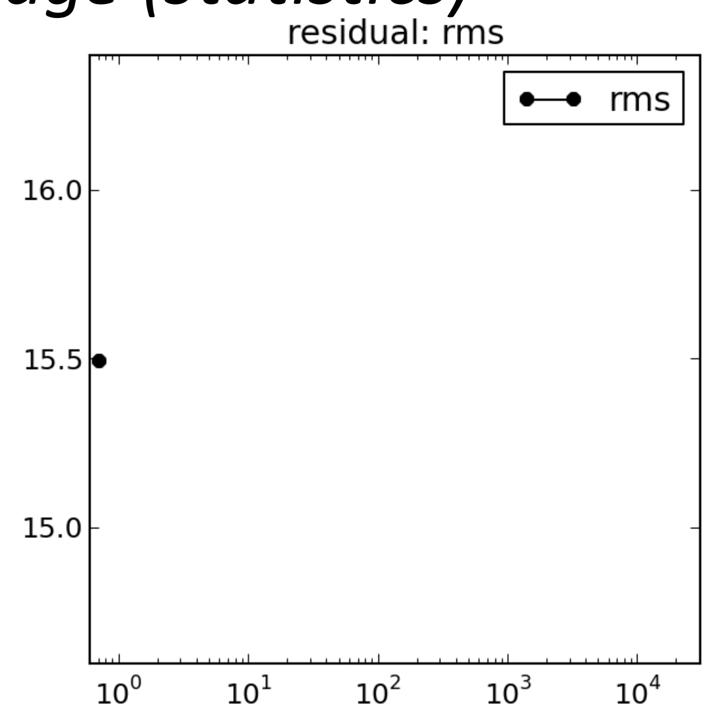


Clean component (statistics)



Residual image (statistics)





Credit: Kuo-Song Wang (ASIAA)

Terms associated with CLEAN

clean, dirty, beam, model, residual…

Dirty beam: point-source response (point spread function) of the data; $\mathcal{F}^{-1}$ of uv-coverage ($\mathcal{F}^{-1}[S(u,v)]$) and normalize the peak to unity

Dirty map: $\mathcal{F}^{-1}$ of visibilities ($\mathcal{F}^{-1}[V(u,v)]$)

Clean beam: (usually) a 2D Gaussian approximation of the main lobe of the dirty beam

Clean map: the sum of clean component map convoluted with clean beam and the residual map

Clean component (model) map: an assumable of point sources determined from CLEAN algorithm [Jy/pixel]

Residual map: a dirty map with clean components (convolved with dirty beam) removed, ideally noise-like at final stage of CLEAN

Generated images after tclean

check your working directory

- After tclean finished, we will see that new images are generated
  - twhya_cont.image — Cleaned and restored image (Jy/beam)
  - twhya_cont.model — Clean component (model) image (Jy/pixel)
  - twhya_cont.pb — Primary beam model
  - twhya_cont.psf — Dirty beam
  - twhya_cont.residual — Residual (Jy/beam)
  - twhya_cont.sumwt — Sum of weights

- Try to visualize them with casa viewer or CARTA

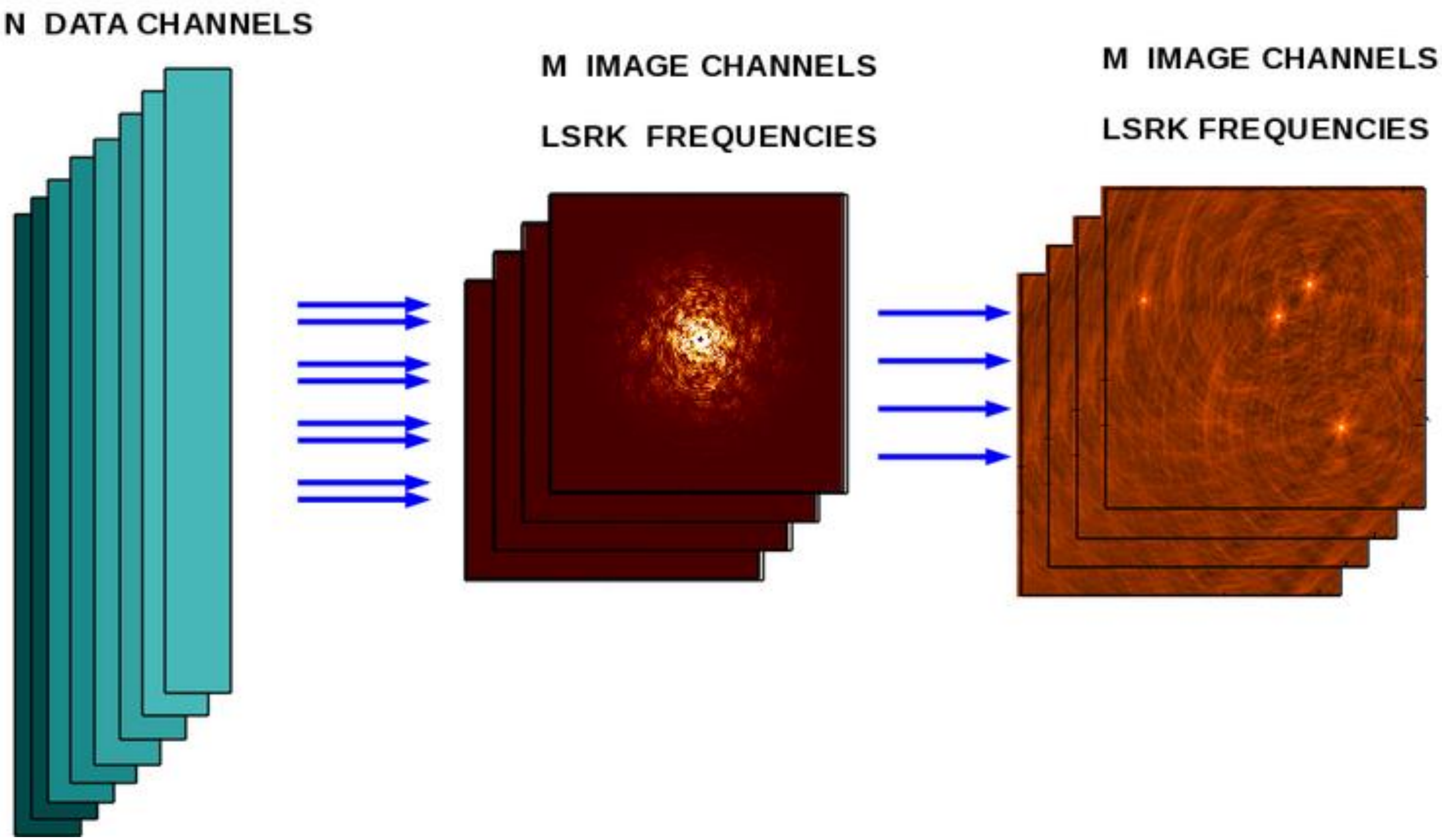
- To run CARTA
  ```
  > cd <your folder>
  > ./CARTA.AppImage --remote
  ```
  Copy and paste the url to your local web browser (Chrome, Firefox, Safari)

- download CARTA at https://cartavis.github.io

CASA tclean task

what tclean can do…

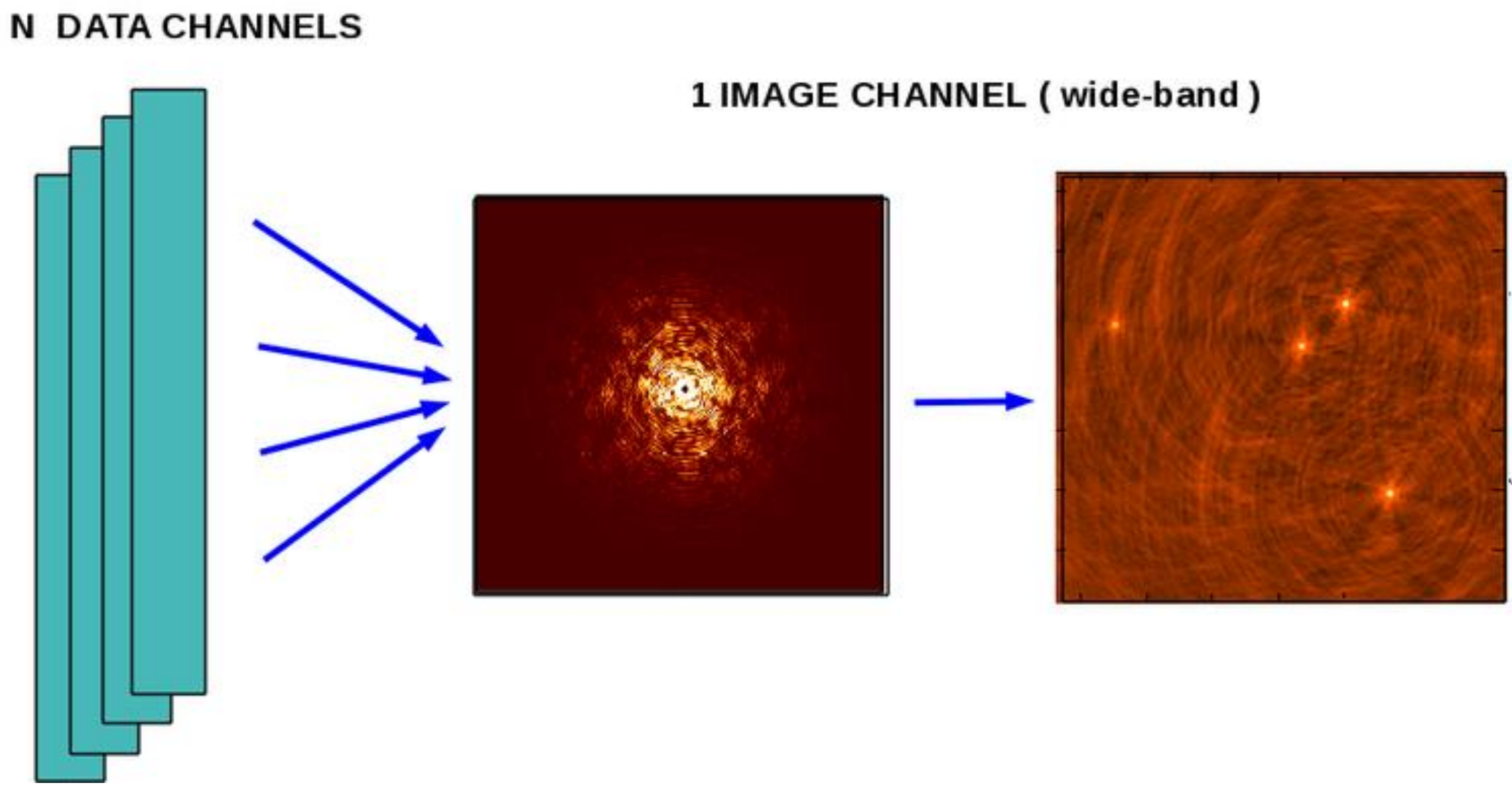
- Spectral cubes



CASA tclean task

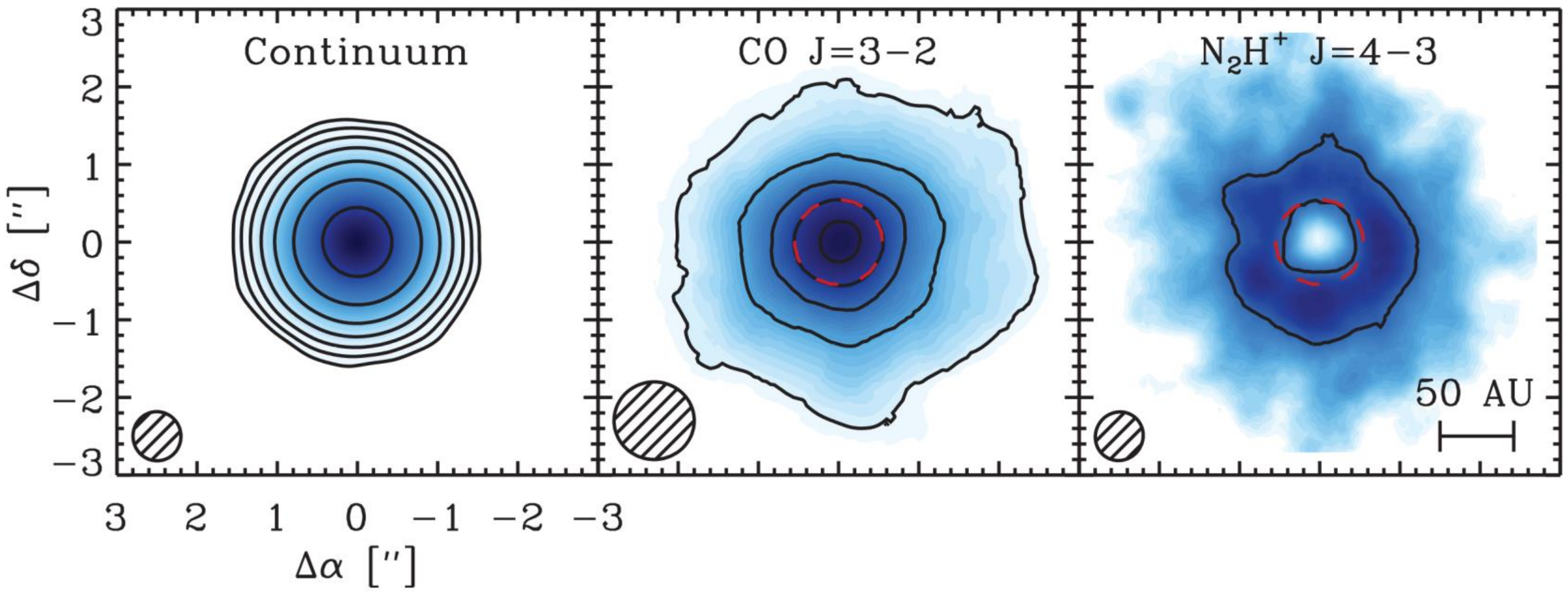
what tclean can do…

- Multi Frequency Synthesis (MFS) - single wideband image



Task 3: CLEAN & Masking

Make clean images (niter!=0) for the science target.



Continuum and N2H+ in TW Hydra at Band 7 (Qi et al. 2013)

```
tclean_tutorial_cont_calibrators.py
tclean_tutorial_cont_pre.py
tclean_tutorial_cont.py
twhya_calibrated.ms.tar
```

Review all parameters before ‘go’

After ‘go’

pay attention to the logger

```
Completed 2042 iterations.
----------------------------------------------------------- Run Major Cycle 5 ---------------------
 Absolute Peak residual within mask : 0.00056293, over full image : 0.0152164
Minor Cycle controls : {'cycleniter': 2958, 'cyclethreshold': 0.00010189844761043787, 'loopgain': 0.10000000149
[phase_cal_uniform] Run Hogbom minor-cycle | CycleThreshold=0.000101898, CycleNiter=2958, Gain=0.1
[phase_cal_uniform] iters=0->2958 [2958], model=0.620792->0.619962, peakres=0.00056293->0.000337509, Reached cy
Completed 5000 iterations.
----------------------------------------------------------- Run Major Cycle 6 ---------------------
 Absolute Peak residual within mask : 0.000337502, over full image : 0.0152439
Reached global stopping criterion : iteration limit
getSummary call: fullsummary=False
[phase_cal_uniform] : Restoring model image.
Beam for chan : 0 : 0.499474 arcsec, 0.395111 arcsec, -44.3226 deg
```

```
Completed 84 iterations.
------------------------------------------------------------ Run Major Cycle 2 ----------------------
 Absolute Peak residual within mask : 0.0536286, over full image : 0.0536286
Minor Cycle controls : {'cycleniter': 9916, 'cyclethreshold': 0.014999999664723873, 'loopgain': 0.10000
[twhya_cont] Run Hogbom minor-cycle | CycleThreshold=0.015, CycleNiter=9916, Gain=0.1
[twhya_cont] iters=0->263 [263], model=0.74245->1.49335, peakres=0.0536286->0.0149695, Reached cyclethr
Completed 347 iterations.
------------------------------------------------------------ Run Major Cycle 3 ----------------------
 Absolute Peak residual within mask : 0.0149695, over full image : 0.020731
Reached global stopping criterion : threshold
getSummary call: fullsummary=False
[twhya_cont] : Restoring model image.
Beam for chan : 0 : 0.560291 arcsec, 0.415713 arcsec, -55.9733 deg
Searching for images with prefix 'twhya_cont'... Found these, writing history into them: ['twhya_cont.ps
```

Image visualization

Dirty images when niter=0

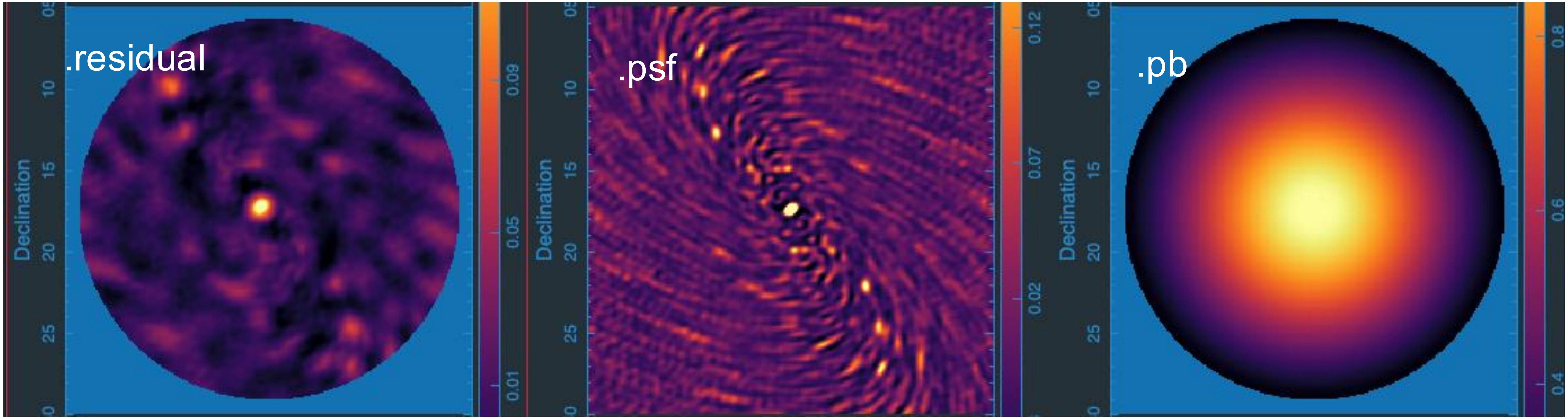


Image visualization

Clean-ed images

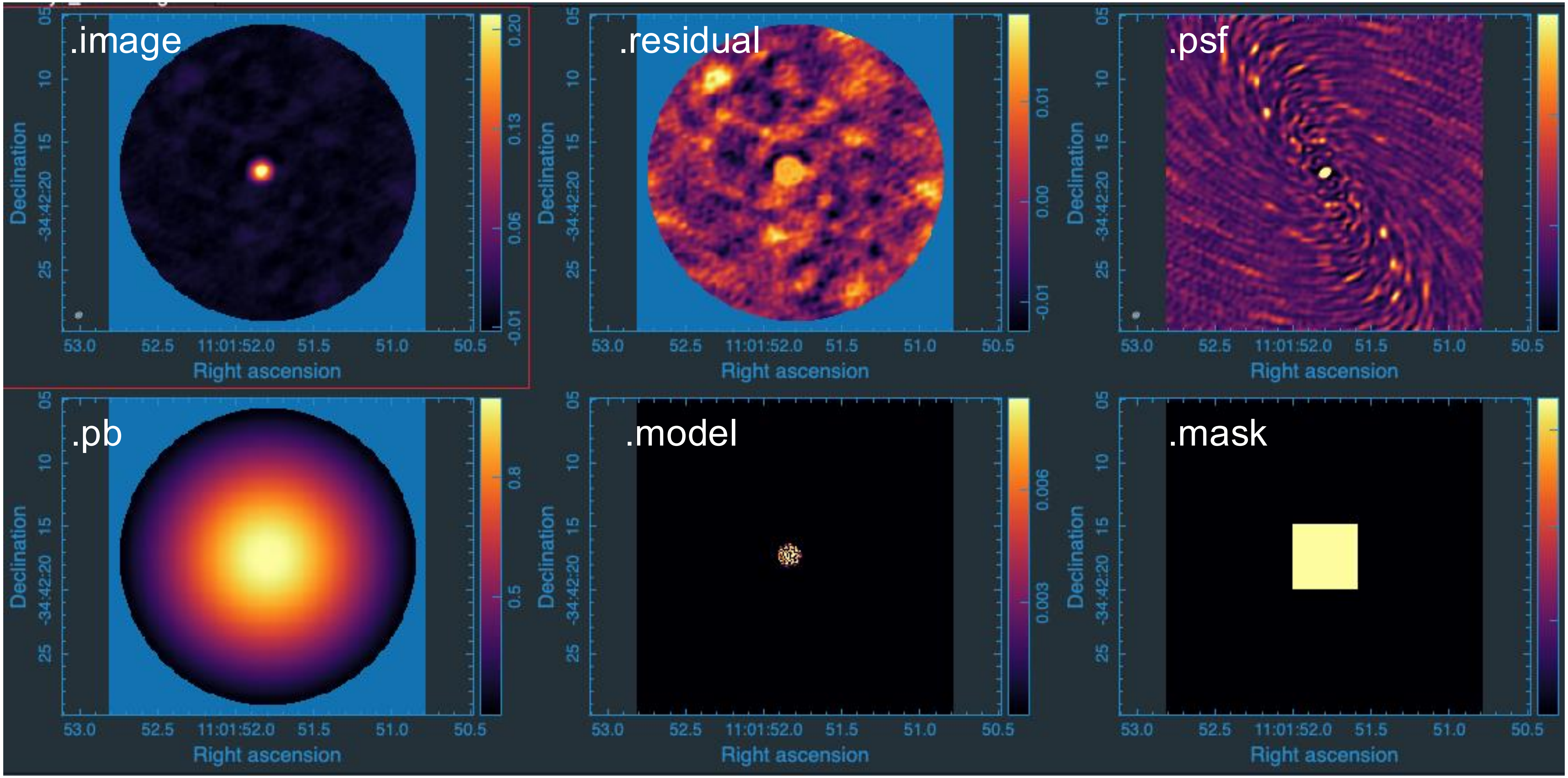
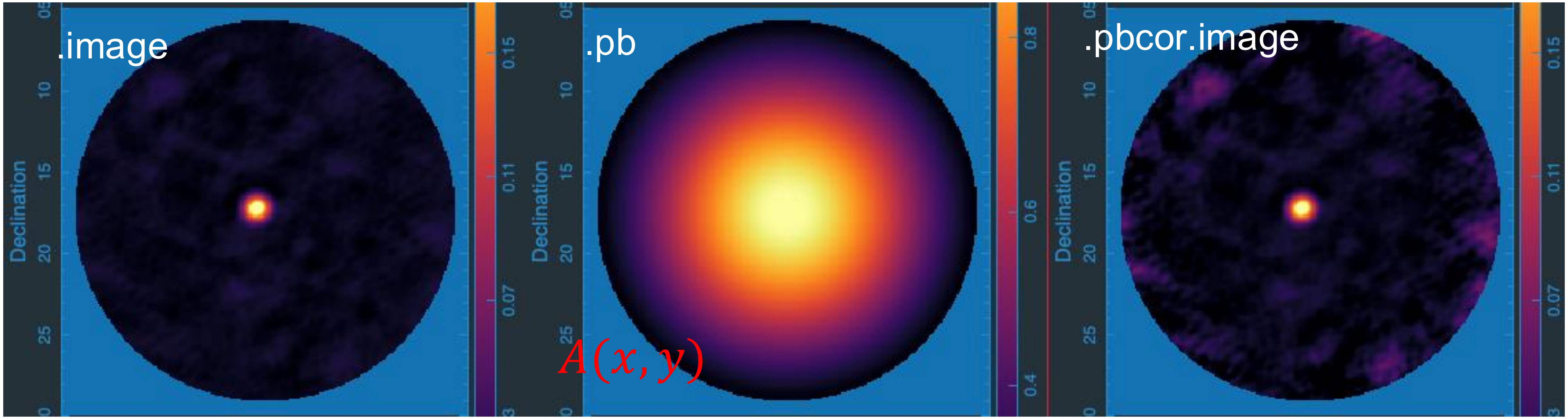


Image visualization

Primary-beam corrected image = *.image/*.pb



$$V_{\rm obs}(u,v) = S(u,v) \iint I_{\rm true}(x,y) e^{-2\pi i(ux+vy)} dxdy$$

$$\Rightarrow V_{\rm obs}(u,v) = S(u,v) \iint A(x,y) I_{\rm true}(x,y) e^{-2\pi i(ux+vy)} dxdy$$

Image visualization

Phase calibrator with different weighting