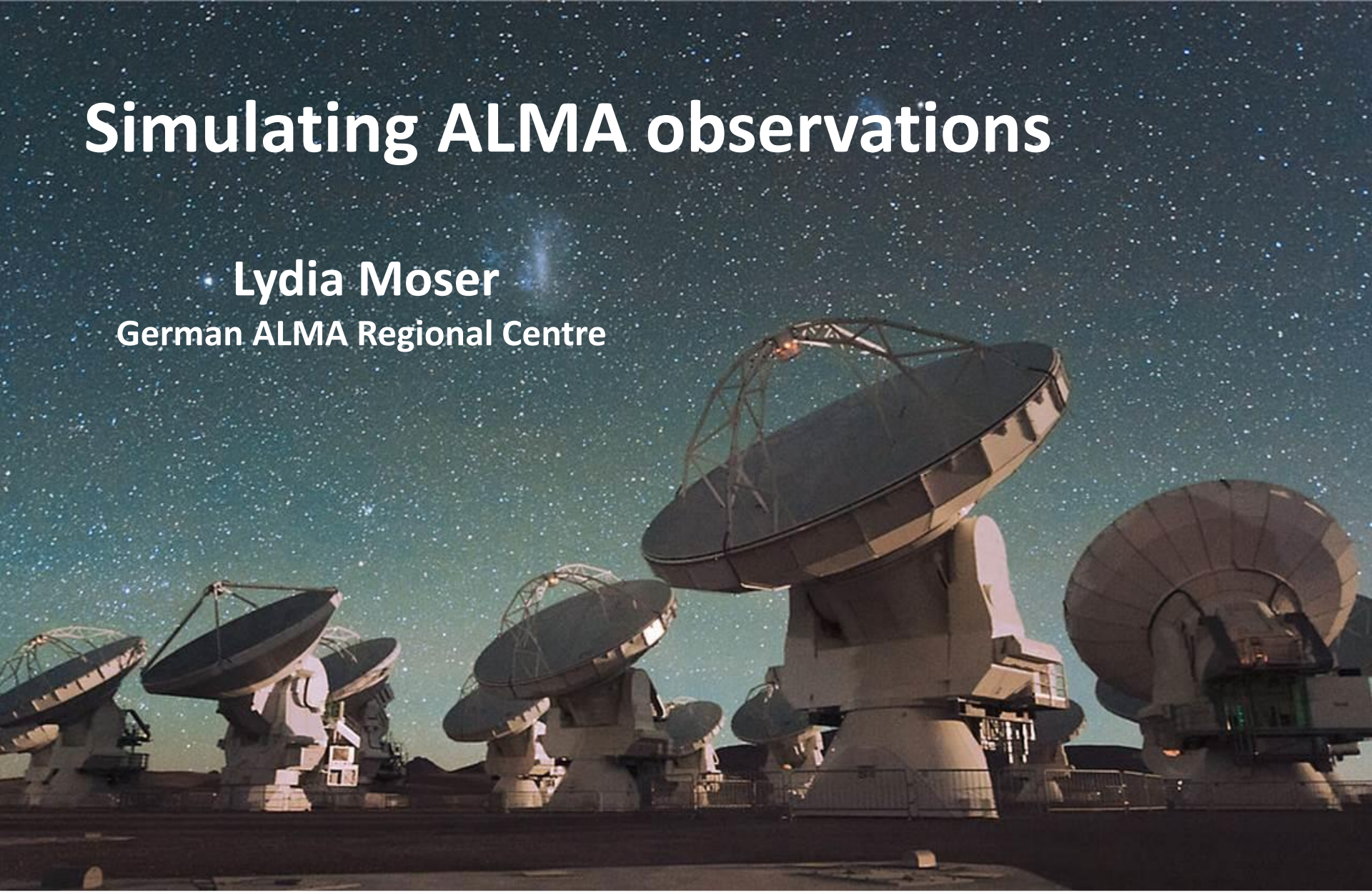
Simulating ALMA observations

Lydia Moser
German ALMA Regional Centre



Outline

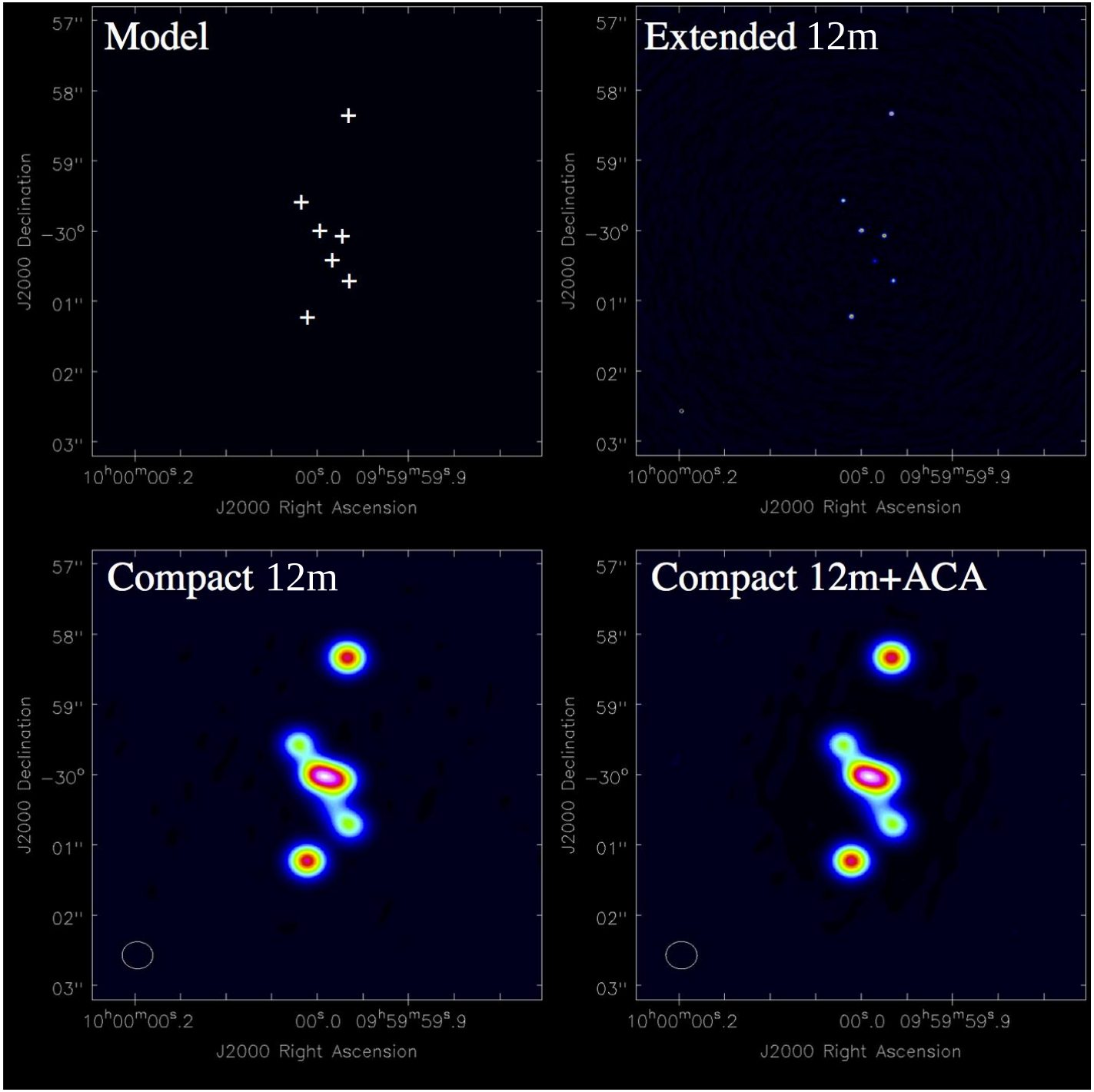
- why simulations?
- Examples for filtering effects and array combination
- Available tools and their concept
- CASA simulator
  - Simobserve
  - Simanalyze
  - Generate 12m + 7m +TP array data with simobserve
  - Simalma

Why simulations ?

- beginners: get familiar with spatial filtering effects
- advanced: test scenarios, mock data

Proposal preparation:

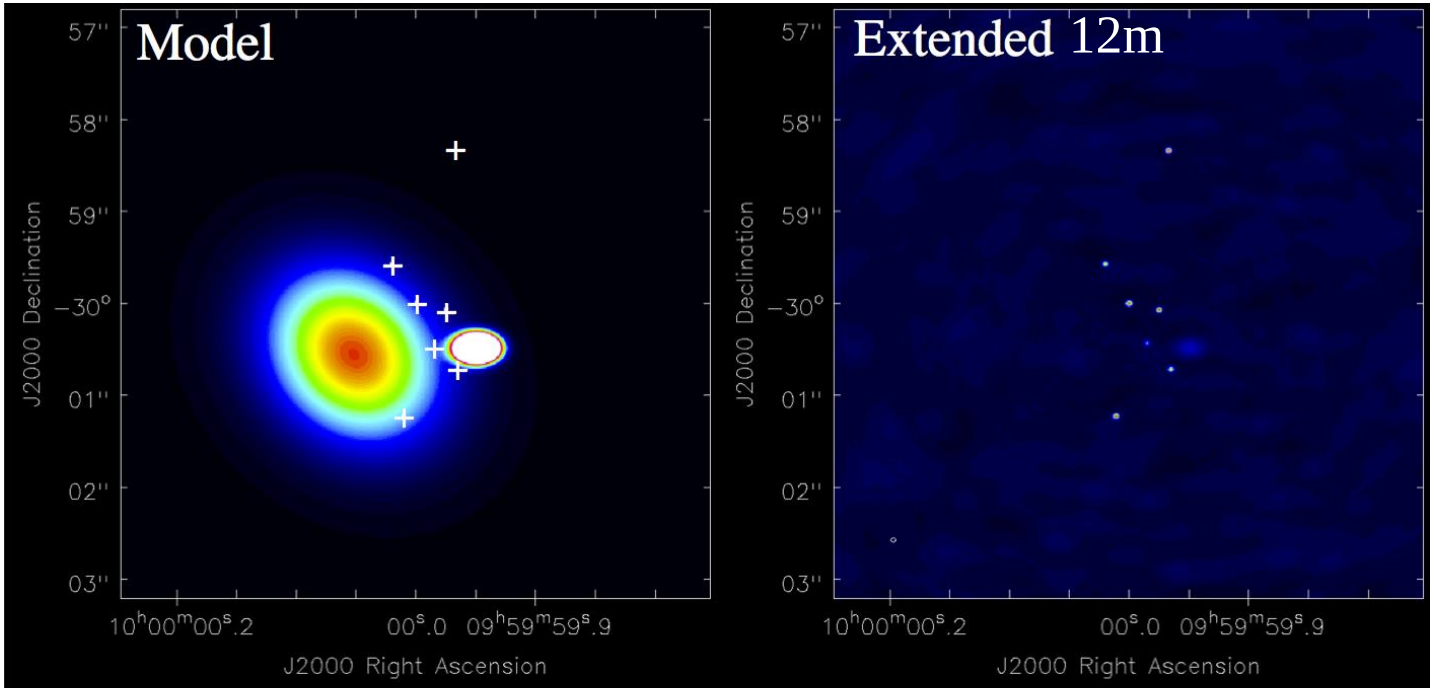
- Get relevant angular scales
    - Angular resolution (beam size) necessary
    - Largest angular scales (LAS) / Maximum Recoverable Scales (MRS)
    - ALMA compact array (ACA), i.e. 7m and/or total power (TP) ?
- Qualitative noise study – effect of artefacts, poor uv-sampling, source structure
- Test special observation setups – feasibility -> use results for technical justification in proposal



Point sources

- Extended: point sources
- Compact: smears
- Compact + ACA: same

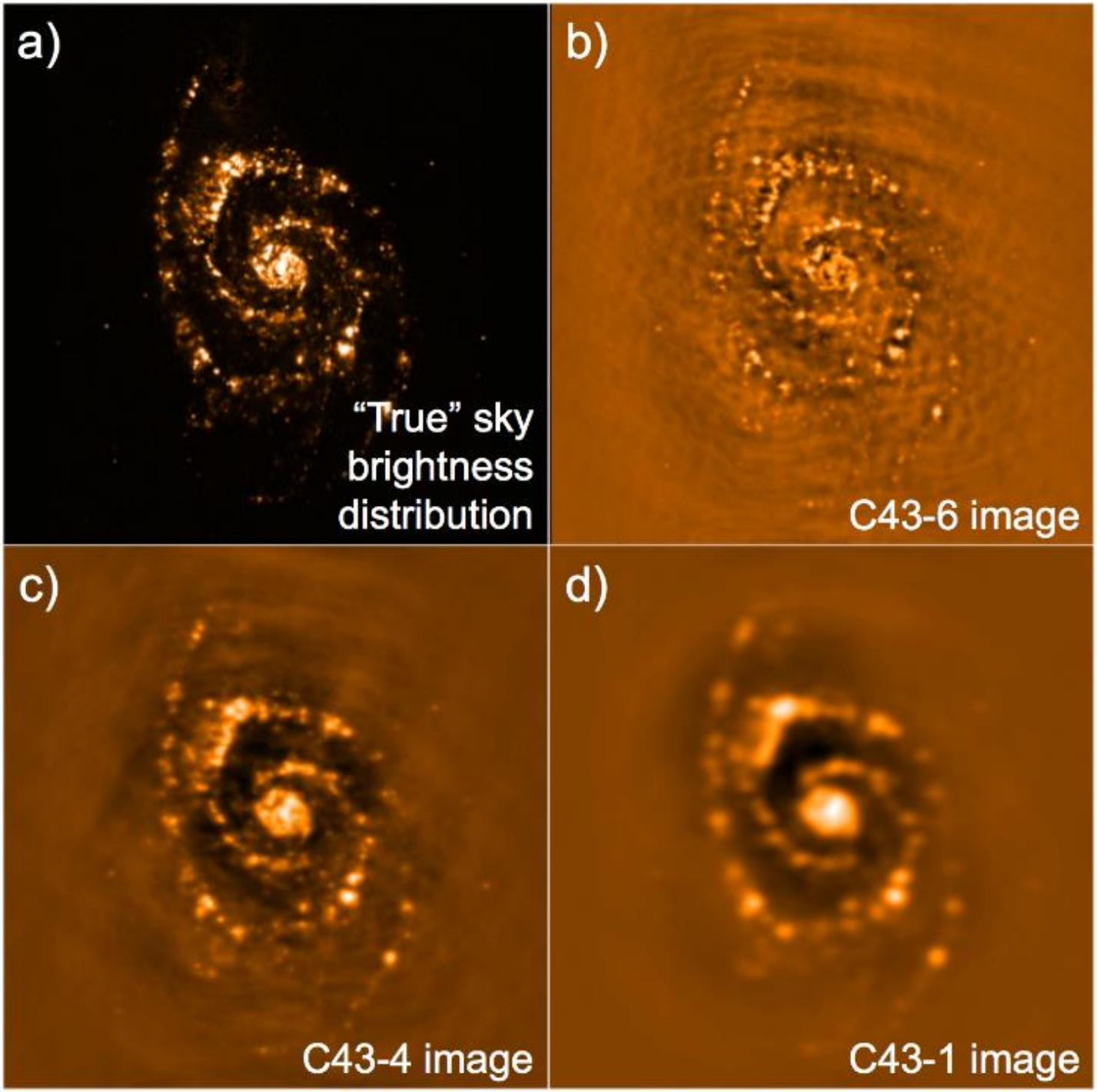
Beam size < ½ source distance



Point sources + elliptical Gaussians

- Extended: point sources
- Compact: missing large Gaussian
- Compact + ACA: better

All scales needed



M51

- Cfg 6: sub-structure
- Cfg 4: largest clusters
- Cfg 1: all + extended emission

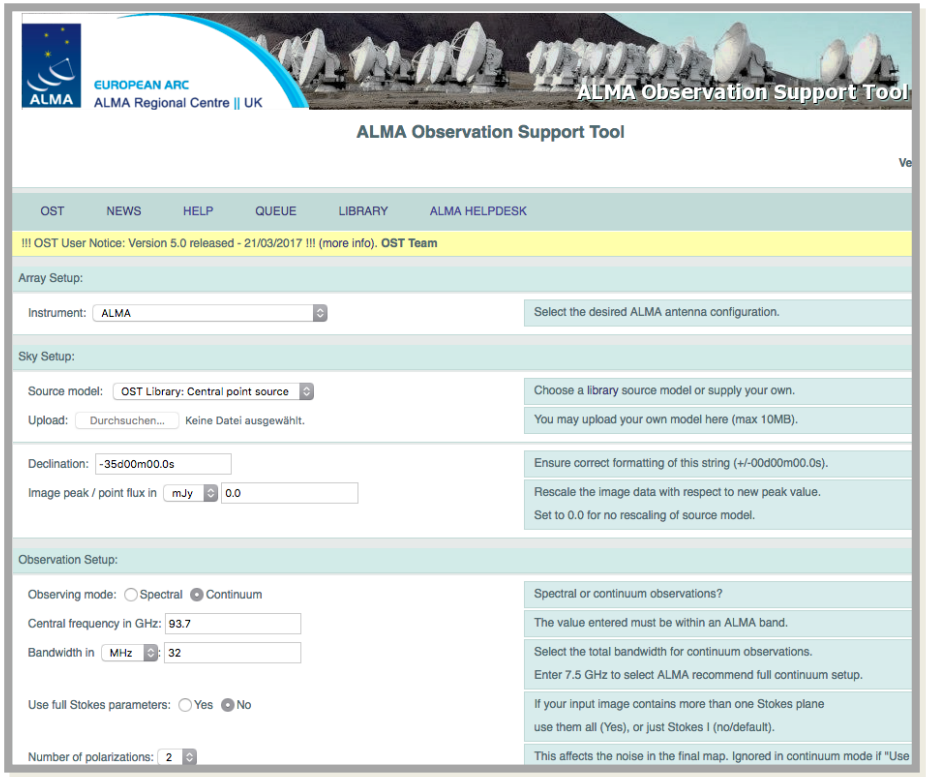
Note: clean bowls

Available simulators

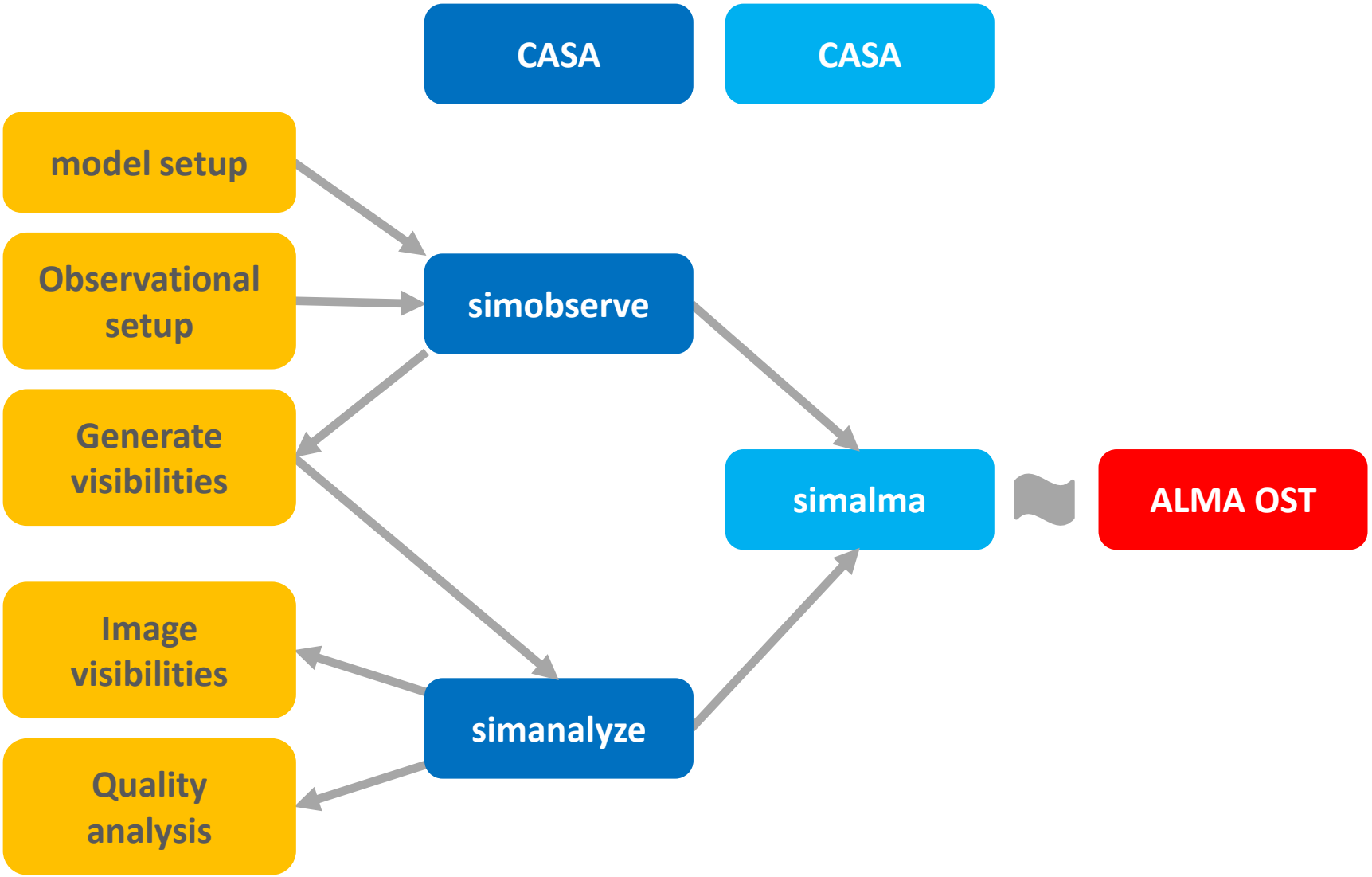
CASA simulator
shell

```
CASA <134>: inp simobserve
----------> inp(simobserve)
#  simobserve :: visibility simulation task
project             = 'simobs-M51Ha'     #  root prefix for output file names
skymodel            = 'M51Ha.fits'       #  model image to observe
     inbright       =    '0.004'         #  scale surface brightness of brightest
     indirection    = 'J2000 23h59m59.96s -34d59m59.50s' #  set new direction e.
     incell         = '0.1arcsec'        #  set new cell/pixel size e.g. "0.1arcs
     incenter       = '330.076GHz'       #  set new frequency of center channel e
     inwidth        =    '50MHz'         #  set new channel width e.g. "10MHz" (r

complist            =         ''         #  componentlist to observe
setpointings        =       True
     integration    =      '10s'         #  integration (sampling) time
     direction      =         ''         #  "J2000 19h00m00 -40d00m00" or "" to c
     mapsize        =  '1arcmin'         #  angular size of map or "" to cover mo
     maptype        =      'hex'         #  hexagonal, square (raster), ALMA, etc
     pointingspacing =  '9arcsec'        #  spacing in between pointings or "0.25

obsmode             =       'int'        #  observation mode to simulate [int(int
     antennalist    = 'ALMA;0.5arcsec'   #  interferometer antenna position file
     refdate        = '2012/12/03'       #  date of observation - not critical un
     hourangle      =  'transit'         #  hour angle of observation center e.g.
                                         #   hours), or "transit"
     totaltime      =    '3600s'         #  total time of observation or number o
     caldirection   =         ''         #  pt source calibrator [experimental]
     calflux        =      '1Jy'

thermalnoise        =         ''         #  add thermal noise: [tsys-atm|tsys-man
leakage             =        0.0         #  cross polarization (interferometer on
graphics            =     'both'         #  display graphics at each stage to [sc
verbose             =      False
overwrite           =       True         #  overwrite files starting with $projec
```

ALMA Observation Support Tool (OST)
http://almaost.jb.man.ac.uk/



Simulation procedure



Simobserve

```
#  simobserve :: visibility simulation task
project             = 'simobs-M51Ha'    #  root prefix for output file names
skymodel            = 'M51Ha.fits'      #  model image to observe
     inbright       =    '0.004'        #  scale surface brightness of brightest pixel e.g. "1.2Jy/pixel"
     indirection    = 'J2000 23h59m59.96s -34d59m59.50s' #  set new direction e.g. "J2000 19h00m00 -40d00m00"
     incell         = '0.1arcsec'       #  set new cell/pixel size e.g. "0.1arcsec"
     incenter       = '330.076GHz'      #  set new frequency of center channel e.g. "89GHz" (required even for 2D model)
     inwidth        =    '50MHz'        #  set new channel width e.g. "10MHz" (required even for 2D model)

complist            =         ''        #  componentlist to observe
setpointings        =       True
     integration    =      '20s'        #  integration (sampling) time
     direction      =         ''        #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize        =  '1arcmin'        #  angular size of map or "" to cover model
     maptype        =     'ALMA'        #  hexagonal, square (raster), ALMA, etc
     pointingspacing =         ''       #  spacing in between pointings or "0.25PB" or "" for ALMA default INT=lambda/D/sqr
                                        #   SD=lambda/D/3

obsmode             =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist    = 'Cycle5-cfg/C43-2.cfg' #  interferometer antenna position file
     refdate        = '2017/12/02'      #  date of observation - not critical unless concatting simulations
     hourangle      =  'transit'        #  hour angle of observation center e.g. "-3:00:00", "5h", "-4.5" (a number without
                                        #   units will be interpreted as hours), or "transit"
     totaltime      =     '0.5h'        #  total time of observation or number of repetitions
     caldirection   =         ''        #  pt source calibrator [experimental]
     calflux        =      '1Jy'

thermalnoise        = 'tsys-atm'        #  add thermal noise: [tsys-atm|tsys-manual|""]
     user_pwv       =        0.5        #  Precipitable Water Vapor in mm
     t_ground       =      269.0        #  ambient temperature
     seed           =      11111        #  random number seed

leakage             =        0.0        #  cross polarization (interferometer only)
graphics            =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose             =      False
overwrite           =       True        #  overwrite files starting with $project
```

Simobserve

```
#  simobserve :: visibility simulation task
project             = 'simobs-M51Ha'     #  root prefix for output file names
skymodel            = 'M51Ha.fits'       #  model image to observe
     inbright       =    '0.004'         #  scale surface brightness of brightest pixel e.g. "1.2Jy/
     indirection    = 'J2000 23h59m59.96s -34d59m59.50s' #  set new direction e.g. "J2000 19h00m00
     incell         = '0.1arcsec'        #  set new cell/pixel size e.g. "0.1arcsec"
     incenter       = '330.076GHz'       #  set new frequency of center channel e.g. "89GHz" (requir
     inwidth        =    '50MHz'         #  set new channel width e.g. "10MHz" (required even for 2D

complist            =         ''         #  componentlist to observe
setpointings        =       True
     integration    =      '20s'         #  integration (sampling) time
     direction      =         ''         #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize        =  '1arcmin'         #  angular size of map or "" to cover model
     maptype        =     'ALMA'         #  hexagonal, square (raster), ALMA, etc
     pointingspacing =       ''         #  spacing in between pointings or "0.25PB" or "" for ALMA
                                        #   SD=lambda/D/3

obsmode             =      'int'         #  observation mode to simulate [int(interferometer)|sd(sin
     antennalist    = 'Cycle5-cfg/C43-2.cfg' #  interferometer antenna position file
     refdate        = '2017/12/02'       #  date of observation - not critical unless concatting sim
     hourangle      =  'transit'         #  hour angle of observation center e.g. "-3:00:00", "5h",
                                        #   units will be interpreted as hours), or "transit"
     totaltime      =     '0.5h'         #  total time of observation or number of repetitions
     caldirection   =         ''         #  pt source calibrator [experimental]
     calflux        =      '1Jy'

thermalnoise        = 'tsys-atm'         #  add thermal noise: [tsys-atm|tsys-manual|""]
     user_pwv       =       0.5          #  Precipitable Water Vapor in mm
     t_ground       =     269.0          #  ambient temperature
     seed           =     11111          #  random number seed

leakage             =       0.0          #  cross polarization (interferometer only)
graphics            =    'both'          #  display graphics at each stage to [screen|file|both|none
verbose             =     False
overwrite           =      True          #  overwrite files starting with $project
```

Input model

Spectal setup

Array and pointing setup

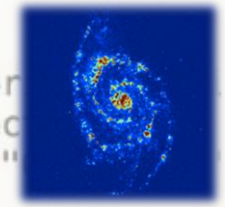
Date and Duration

Noise corruption

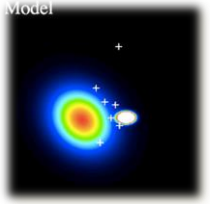
Input model

- fits-file (also cube)

```
skymodel            = 'M51Ha.fits'      #  model image to observe
      inbright      =    '0.004'        #  scale surface brightness of br
      indirection   = 'J2000 23h59m59.96s -34d59m59.50s' #  set new direc
      incell        = '0.1arcsec'       #  set new cell/pixel size e.g. "
```



- create fits-file from GIF or JPG with ImageMagik: `convert myfile.jpg myfile.fits`
- simulate object at higher redshift → rescale brightness and angular scale
  (see https://casaguides.nrao.edu/index.php/M51_at_z_%3D_0.1#Flux_Density_Scaling)



- component list (point sources, disks, Gaussians, ...)

```
complist            =           ''      #  componentlist to observe
```

- See https://casaguides.nrao.edu/index.php/Simulation_Guide_Component_Lists_(CASA_3.3)

Spectral setup

```
      incenter      = '330.076GHz'      #  set new frequency of center channel e.g.
      inwidth       =    '50MHz'        #  set new channel width e.g. "10MHz" (requi
```

Array configuration

- interferometer

```
obsmode                 =       'int'          #  observation mode to simulate [int(interfe
     antennalist        = 'Cycle5-cfg/C43-2.cfg' #  interferometer antenna position file
```

- antennalist:
  - = 'alma; 0.5arcsec'
  - = os.getenv('CASAPATH').split(' ')[0] + '/data/alma/simmos/>choose_a_list<.cfg‘
  - Recent configurations: https://almascience.eso.org/tools/casa-simulator
  - Beam size > model image cell size
  - Array resolution and LAS: https://almascience.eso.org/proposing/proposers-guide#section-59
  - Arrays to combine like in a real ALMA observation: https://almascience.eso.org/proposing/proposers-guide#section-63

- single dish

```
obsmode                 =         'sd'         #  observation mode to simulate [int(interfe
     sdantlist          = 'aca.tp.cfg'         #  single dish antenna position file
     sdant              =            0         #  single dish antenna index in file
```

Pointing setup

- provide a list of pointing positions

```
setpointings          =      False
     ptgfile          = '$project.ptg.txt' #  list of pointing positions
     integration      =      '20s'         #  integration (sampling) time
```

- OT generated pointings cannot be used?

- calculate a map of pointings

```
setpointings          =       True
     integration      =      '20s'        #  integration (sampling) time
     direction        =        ''         #  "J2000 19h00m00 -40d00m00" or "" to cente
     mapsize          =  '1arcmin'        #  angular size of map or "" to cover model
     maptype          =    'ALMA'         #  hexagonal, square (raster), ALMA, etc
     pointingspacing =          ''        #  spacing in between pointings or "0.25PB" c
                                          #   SD=lambda/D/3
```

- If `obsmode = 'sd'` : mapsize <0.5PB larger than for 'int' – avoid edge effects
- $project.ptg.txt generated automatically

Date and Duration

```
integration    =      '20s'        #  integration (sampling) time
```

- time per pointing

```
refdate        = '2017/12/02'      #  date of observation - not critical unless
hourangle      =  'transit'        #  hour angle of observation center e.g. "-3
                                   #   units will be interpreted as hours), or
totaltime      =     '0.5h'        #  total time of observation or number of re
```

- refdate: use different dates for each configuration/data set within your project
- totaltime:
  - also: = # repetion per map
  - Observing time ratios for multiple arrays to combine like in a real ALMA observation:
    https://almascience.eso.org/proposing/proposers-guide#section-63

Noise corruption

- Atmospheric Transmission at Microwaves (ATM) model

```
thermalnoise          = 'tsys-atm'        #  add thermal noise: [tsys-atm|tsys-manual
     user_pwv         =         0.5       #  Precipitable Water Vapor in mm
     t_ground         =       269.0       #  ambient temperature
     seed             =       11111       #  random number seed
```

- Zenith opacity

```
thermalnoise          = 'tsys-manual'     #  add thermal noise: [tsys-atm|tsys-manual|
     t_ground         =       269.0       #  ambient temperature
     t_sky            =       263.0       #  atmospheric temperature
     tau0             =         0.1       #  zenith opacity
     seed             =       11111       #  random number seed
```

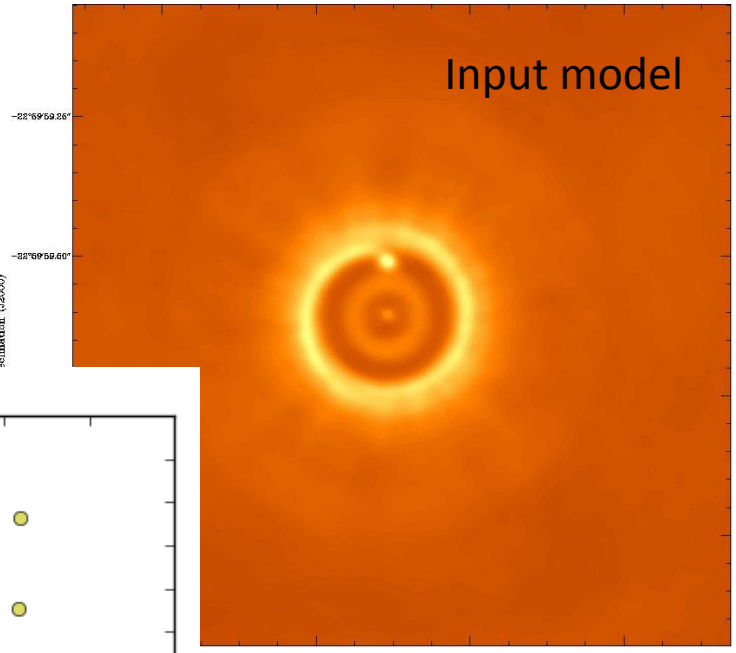
- No noise: thermalnoise= ''
- RMS in the resulting map should not be used in the proposal
- use ALMA sensitivity calculator instead

- **sm tool**: advanced users; apply after simobserve without noise
  - thermal noise, phase delay variations, gain fluctuations and drift cross-polarization, etc. (see https://casaguides.nrao.edu/index.php/Corrupt)
    → more flexibility in adding thermal noise

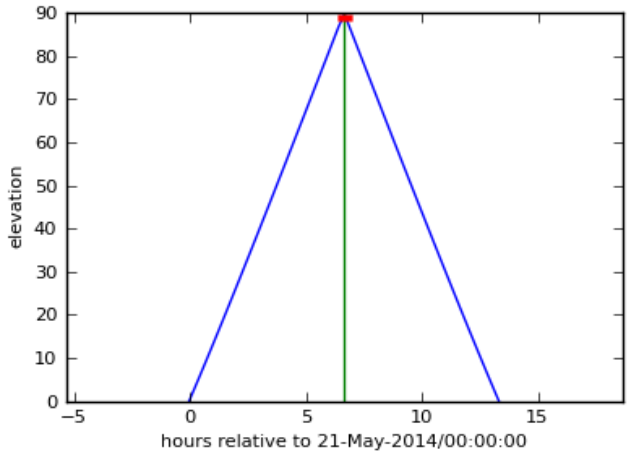
Simobserve

Diagnostic image products

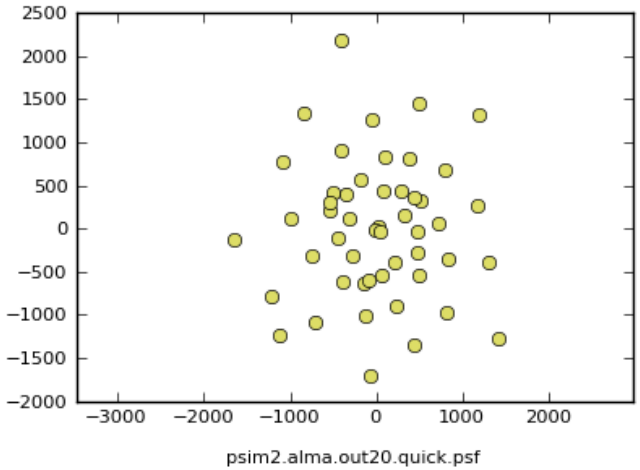
Example: Protoplanetary disk with 12m array

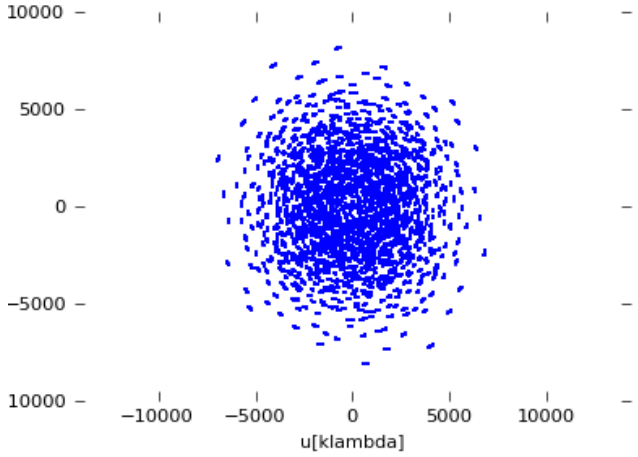


Elevation

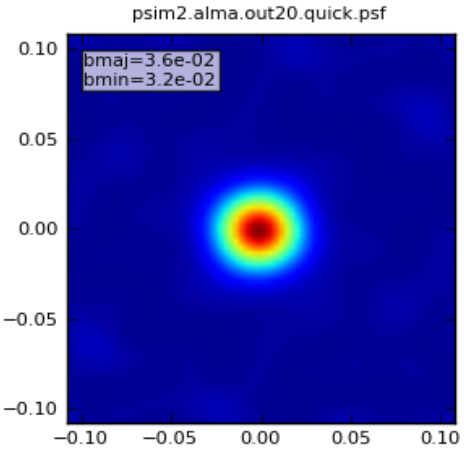


Antenna positions



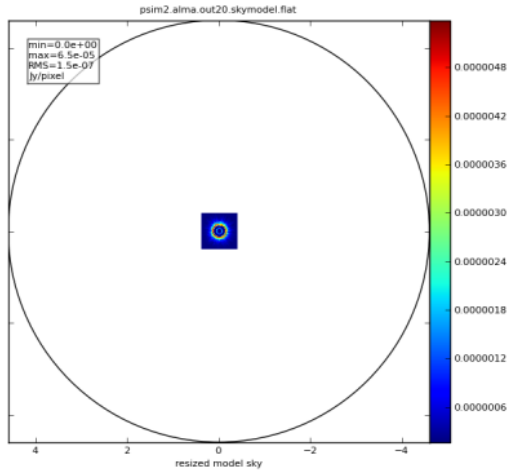


uv coverage



Synthesized beam

skymodel



Simanalyze

```
#  simanalyze :: image and analyze measurement sets created with simobserve
project             = 'simobs-M51Ha'     #  root prefix for output file names
image               =      True          #  (re)image $project.*.ms to $project.image
     vis            = '$project.C43-2.noisy.ms' #  Measurement Set(s) to image
     modelimage     =          ''        #  lower resolution prior image to use in clean e.g. existing total power image
     imsize         =           0        #  output image size in pixels (x,y) or 0 to match model
     imdirection    =          ''        #  set output image direction, (otherwise center on the model)
     cell           =          ''        #  cell size with units e.g. "10arcsec" or "" to equal model
     interactive    =     False          #  interactive clean?  (make sure to set niter>0 also)
     niter          =           0        #  maximum number of iterations (0 for dirty image)
     threshold      =   '0.1mJy'         #  flux level (+units) to stop cleaning
     weighting      =  'natural'         #  weighting to apply to visibilities.  briggs will use robust=0.5
     mask           =          []        #  Cleanbox(es), mask image(s), region(s), or a level
     outertaper     =          []        #  uv-taper on outer baselines in uv-plane
     pbcor          =       True         #  correct the output of synthesis images for primary beam response?
     stokes         =         'I'        #  Stokes params to image
     featherimage   =          ''        #  image (e.g. total power) to feather with new image

analyze             =        True        #  (only first 6 selected outputs will be displayed)
     showuv         =       False        #  display uv coverage
     showpsf        =       False        #  display synthesized (dirty) beam (ignored in single dish simulation)
     showmodel      =        True        #  display sky model at original resolution
     showconvolved  =        True        #  display sky model convolved with output clean beam
     showclean      =        True        #  display the synthesized image
     showresidual   =        True        #  display the clean residual image (ignored in single dish simulation)
     showdifference =        True        #  display difference between output cleaned image and input model sky image cor
     showfidelity   =        True        #  display fidelity (see help)

graphics            =      'both'        #  display graphics at each stage to [screen|file|both|none]
verbose             =       False
overwrite           =        True        #  overwrite files starting with $project
dryrun              =        True        #  only print information [experimental; only for interfermetric data]
logfile             =          ''
```

Simanalyze

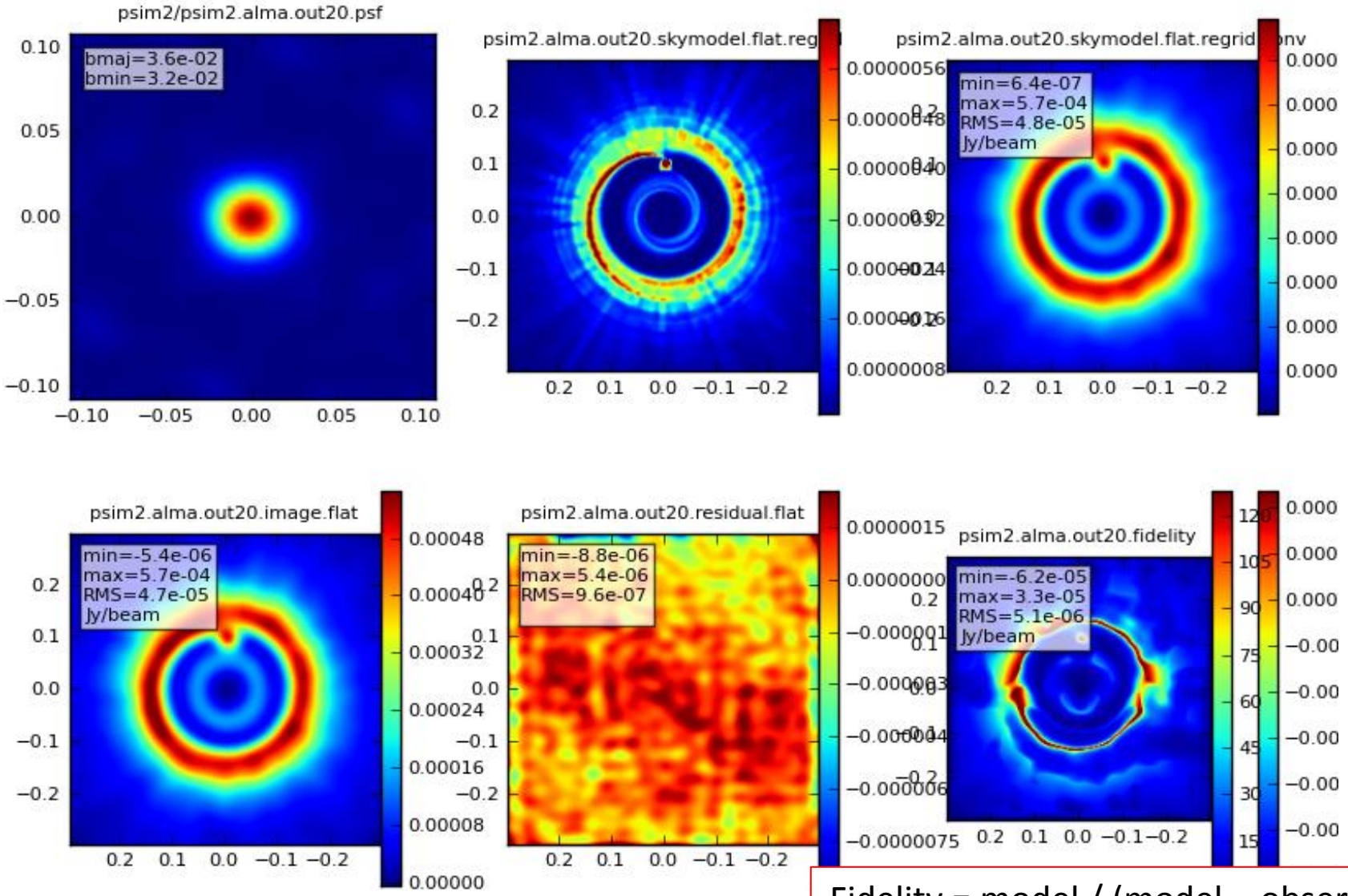
```
#  simanalyze :: image and analyze measurement sets created with simobserve
project             = 'simobs-M51Ha'     #  root prefix for output file names
image               =      True          #  (re)image $project.*.ms to $project.image
     vis            = '$project.C43-2.noisy.ms' #  Measurement Set(s) to image
     modelimage     =          ''        #  lower resolution prior image to use in clean e.g. existing
     imsize         =           0        #  output image size in pixels (x,y) or 0 to match model
     imdirection    =          ''        #  set output image direction, (otherwise center on the model
     cell           =          ''        #  cell size with units e.g. "10arcsec" or "" to equal model
     interactive    =     False          #  interactive clean?  (make sure to set niter>0 also)
     niter          =           0        #  maximum number of iterations (0 for dirty image)
     threshold      =    '0.1mJy'        #  flux level (+units) to stop cleaning
     weighting      =   'natural'        #  weighting to apply to visibilities.  briggs will use robus
     mask           =          []        #  Cleanbox(es), mask image(s), region(s), or a level
     outertaper     =          []        #  uv-taper on outer baselines in uv-plane
     pbcor          =        True        #  correct the output of synthesis images for primary beam re
     stokes         =         'I'        #  Stokes params to image
     featherimage   =          ''        #  image (e.g. total power) to feather with new image

analyze             =        True        #  (only first 6 selected outputs will be displayed)
     showuv         =       False        #  display uv coverage
     showpsf        =       False        #  display synthesized (dirty) beam (ignored in single dish s
     showmodel      =        True        #  display sky model at original resolution
     showconvolved  =        True        #  display sky model convolved with output clean beam
     showclean      =        True        #  display the synthesized image
     showresidual   =        True        #  display the clean residual image (ignored in single dish s
     showdifference =        True        #  display difference between output cleaned image and input
     showfidelity   =        True        #  display fidelity (see help)

graphics            =      'both'        #  display graphics at each stage to [screen|file|both|none]
verbose             =       False
overwrite           =        True        #  overwrite files starting with $project
dryrun              =        True        #  only print information [experimental; only for interfermetric data]
logfile             =          ''
```

clean

diagnostic products

Simanalyze

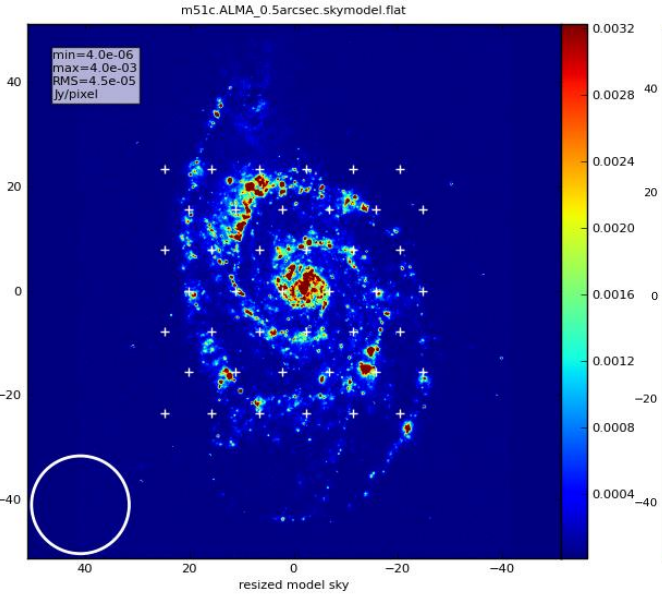
Diagnostic image products - Example: Protoplanetary disk with the 12m array

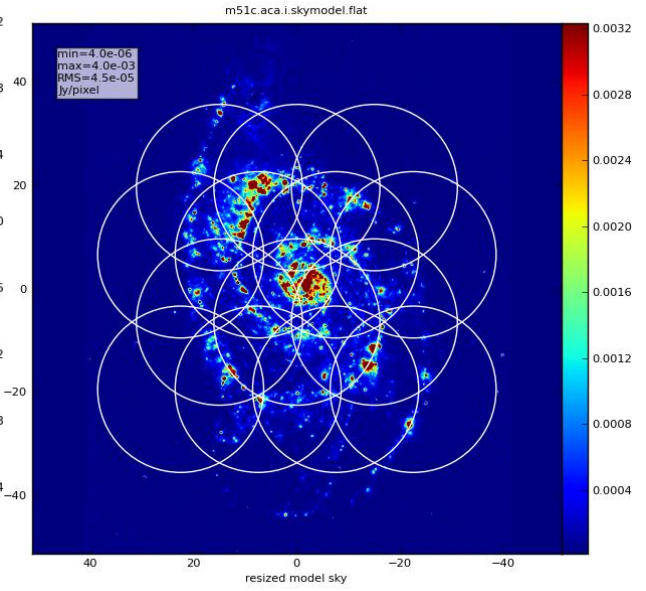


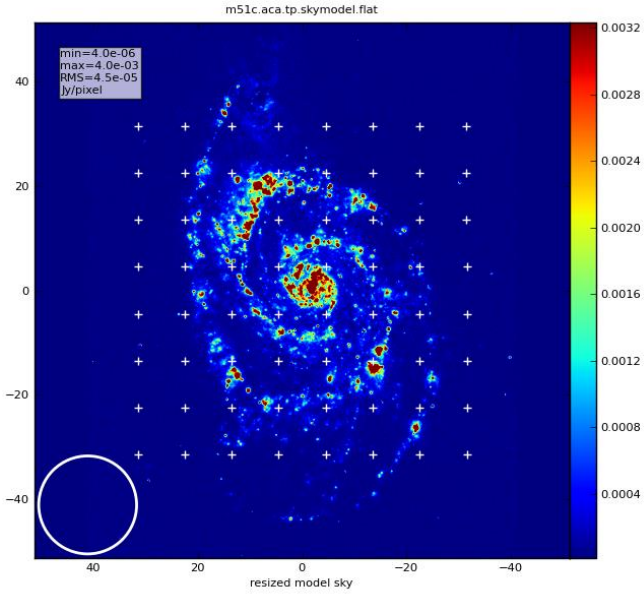
Fidelity = model / (model – observed)

Simobserve - ALMA 12m, ACA (7m) and TP

	12 m array	7 m array	Single dish
setpointings	True	True	True
integration	'10s'	'10s'	'10s'
direction	''	''	''
mapsize	'1arcmin'	'1arcmin'	'1.3arcmin'
maptype	'hex'	'hex'	'square'
pointingspacing	'9arcsec'	'15arcsec'	'9arcsec'
obsmode	'int'	'int'	'sd'
antennalist	'ALMA;0.5arcsec'	'aca.i.cfg'	'aca.tp.cfg'
refdate	'2012/12/03'	'2012/12/02'	'2012/12/01'
hourangle	'transit'	'transit'	'transit'
totaltime	'3600s'	'3'	'2h'





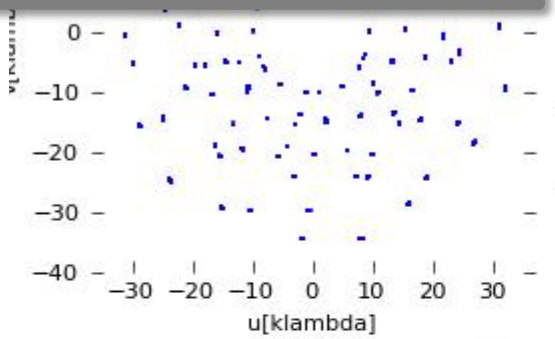


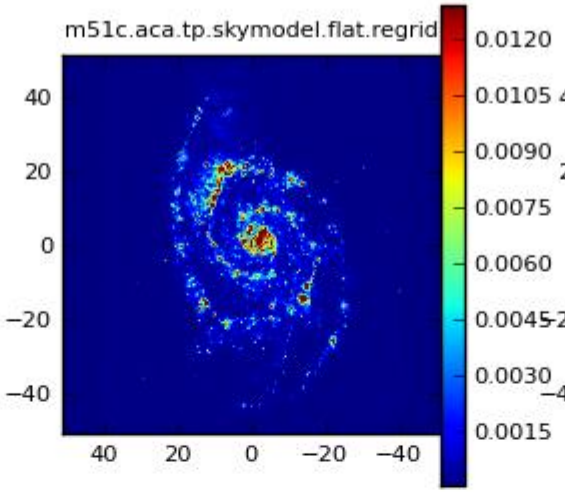
Simanalyze - combine ALMA 12m, ACA (7m) and TP

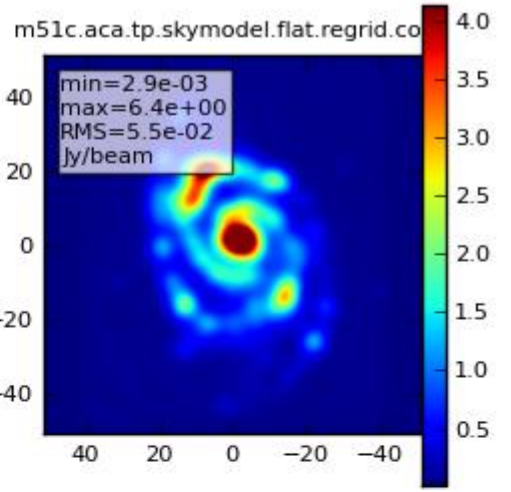
One of many ways: use lower resolution image as model for the higher resolution

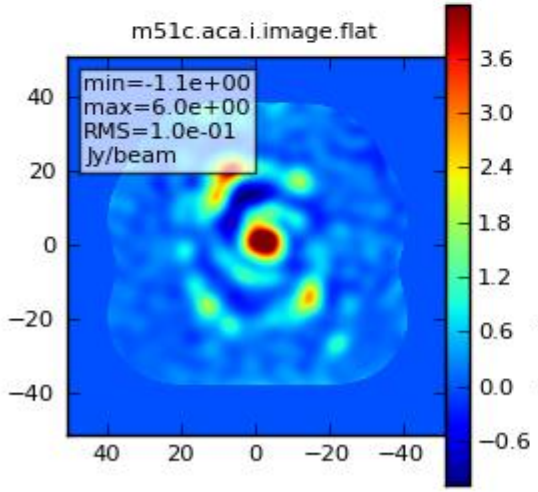
```
project   = "m51c"
vis       = '$project.aca.i.ms,
             $project.aca.tp.sd.ms'
imsize    = [512,512]
cell      = '0.2arcsec'
```

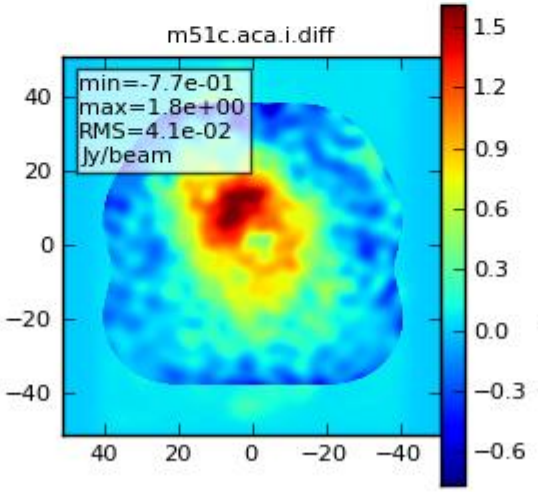
Combine TP + 7m

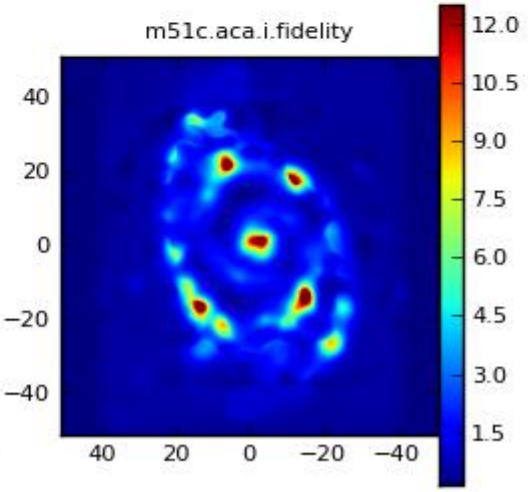










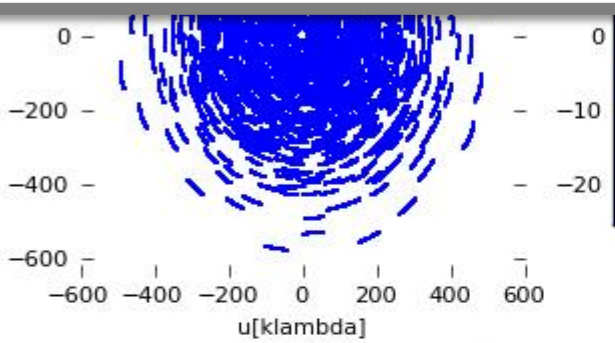


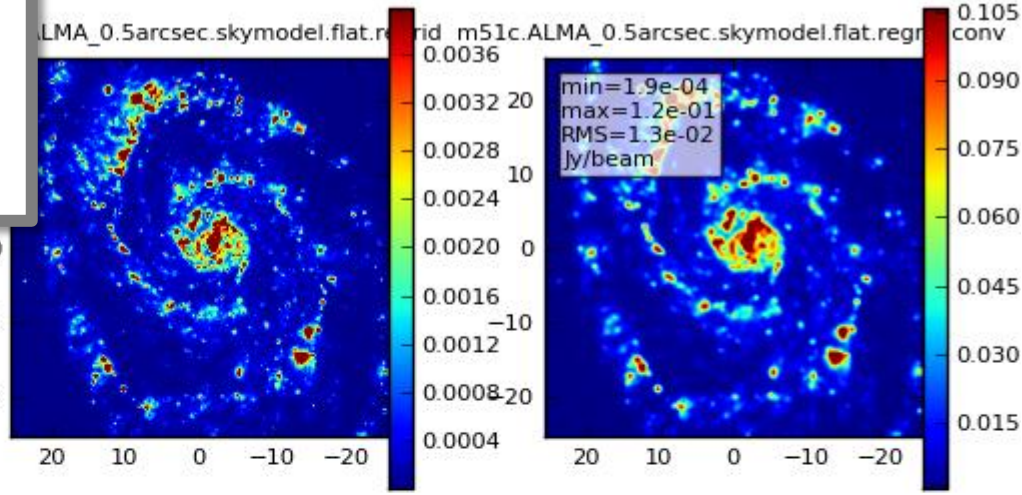
Simanalyze - combine ALMA 12m, ACA (7m) and TP

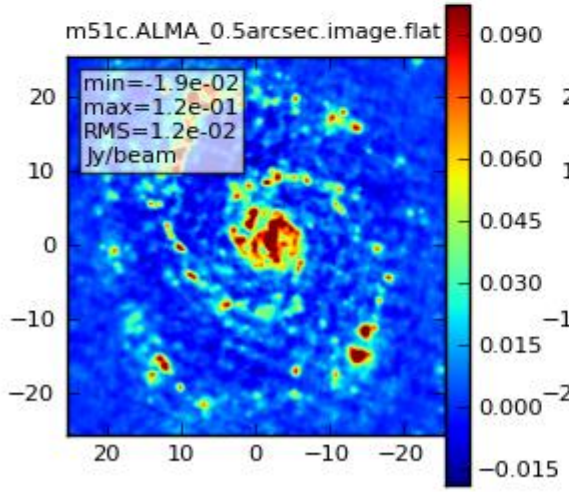
One of many ways: use lower resolution image as model for the higher resolution

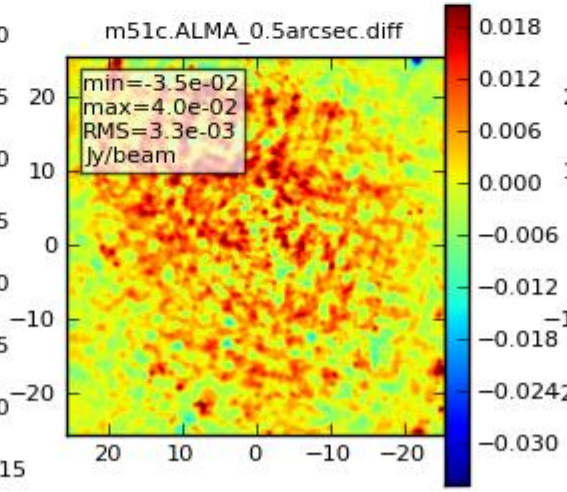
```
project   = "m51c"
vis       = '$project.ALMA_0.5arcsec.ms'
imsize    = [512,512]
cell      = '0.1arcsec'
modelimage = "$project.aca.i.image"
```

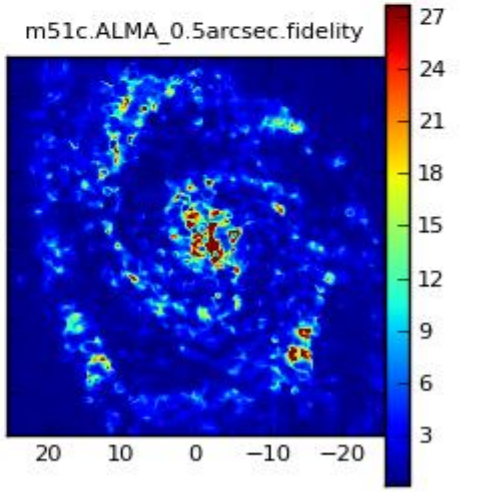
Combine 7m (TP) +12m







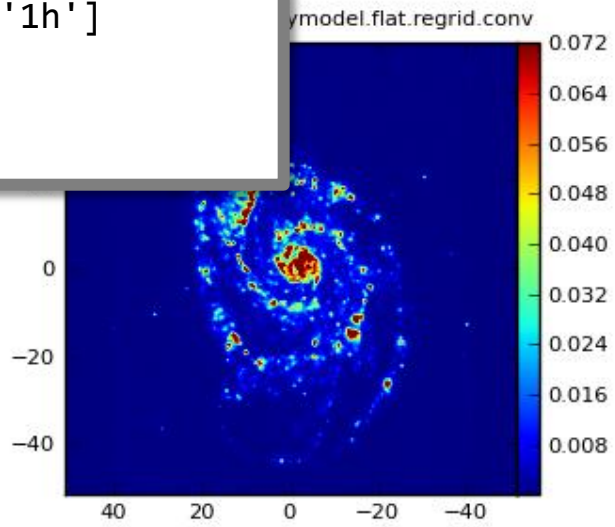


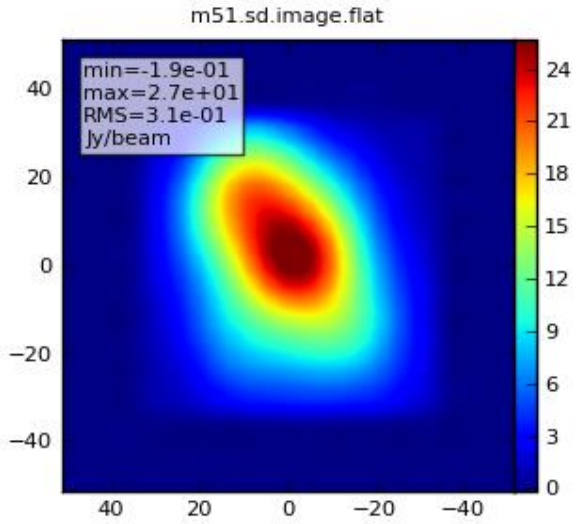


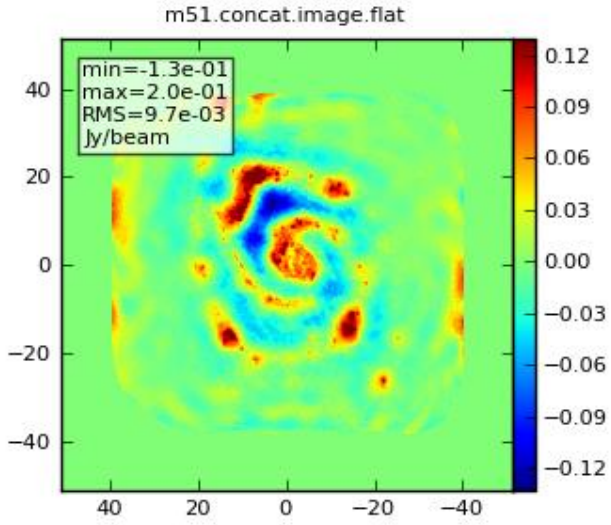
Simalma - generate and combine ALMA 12m, ACA (7m) and TP

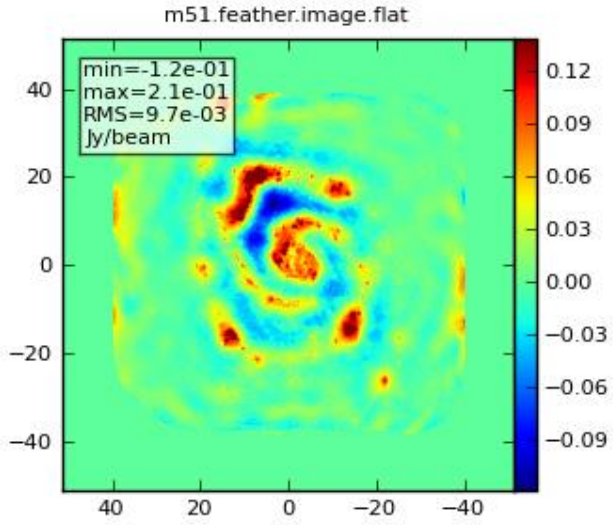
```
mapsize   = "1arcmin"
antennalist= ["alma_cycle1_3.cfg",
               "aca_cycle1.cfg"]
totaltime= ['30min', '1h']
tpnant    = 2
tptime    = '2h'
pwv       = 0.6
```

- Simobserve + simanalyze for each array
- Simanalyze of concatenated interferometic data (weigths!)
- Feather concatenated interferometic image with SD image









Summary

- Simulations – useful tool to study observation setup/ feasibility
- basic simulator procedure demonstrated for CASA simulator
  - Simobserve
  - Simanalyze
  - Simalma
- More details?! → CASA inline help (e.g. `help simobserve`), tutorials, manual, Google

Based on https://casaguides.nrao.edu/index.php/Simulating_Observations_in_CASA_4.4

- https://casaguides.nrao.edu/index.php/Guide_To_Simulating_ALMA_Data
- https://casaguides.nrao.edu/index.php/Protoplanetary_Disk_Simulation_(CASA_4.4)
- https://casaguides.nrao.edu/index.php/ACA_Simulation_(CASA_4.4)
- https://casaguides.nrao.edu/index.php/Simalma_(CASA_4.4)