Simulations and Imaging in CASA

CASA

Common Astronomy Software Applications

Download latest version of CASA here

https://casa.nrao.edu/casa_obtaining.shtml

Veronica Allen

Written by Kristina Nyland

Associated Universities, Inc.

NRAO

Outline

- CASA overview
- Simulation guidelines
- CASA simulation example

Outline

- **CASA overview**
- Simulation guidelines
- CASA simulation example

What is CASA?

Online info: https://casaguides.nrao.edu

Welcome to CASA Guides

CASA (Common Astronomy Software Applications) is a comprehensive software package to calibrate, image, and analyze radio astronomical data from interferometers (such as ALMA and VLA) as well as single dish telescopes. This wiki provides tutorials for reducing data in CASA.

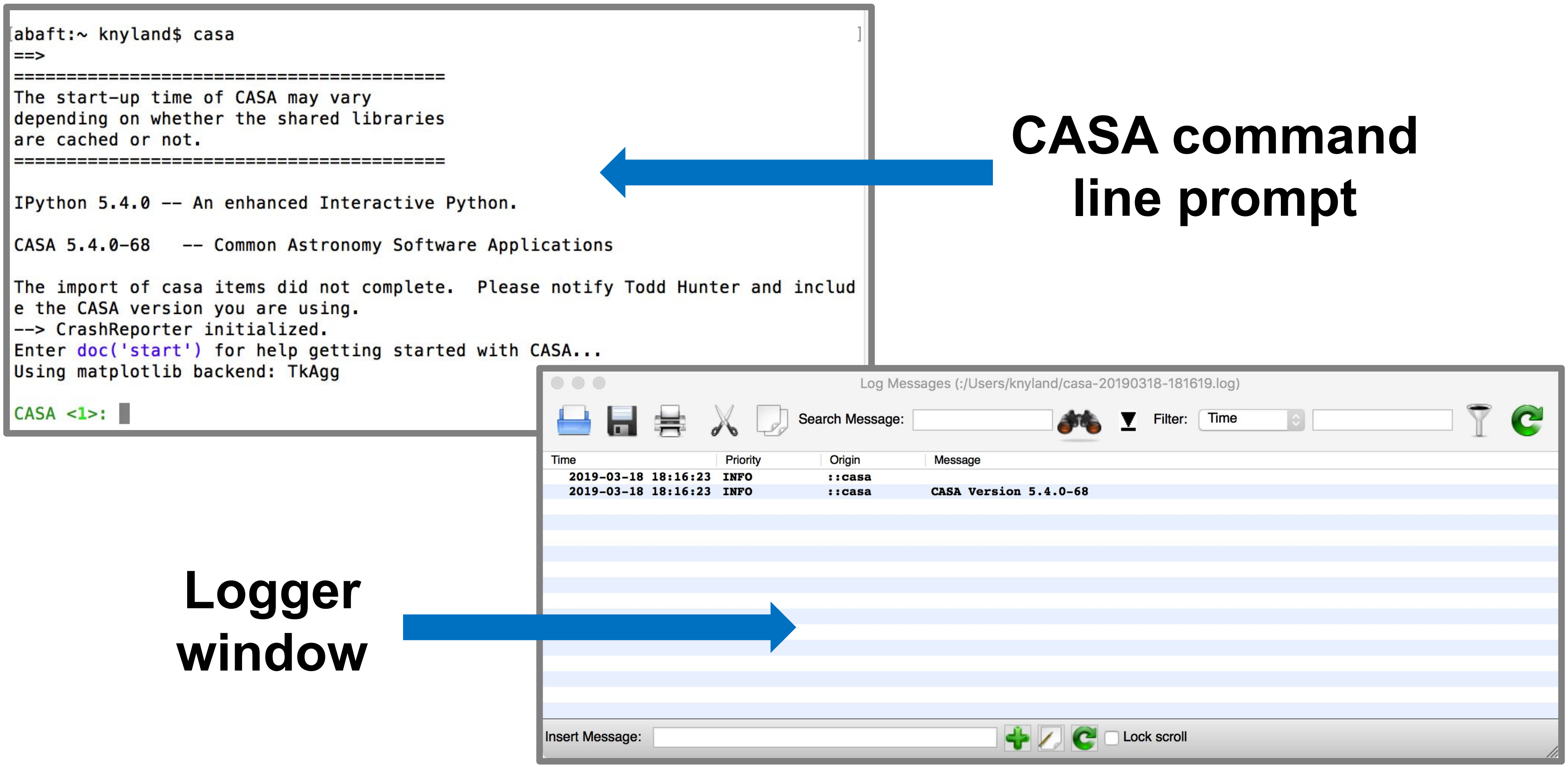
Homepage	Newsletter	CASA Docs	Download
Helpdesk	Subscribe	Forum	Tips

CASA is your go-to tool for simulations, calibration, imaging, and analysis with a friendly iPython interface

CASA Tutorials

Getting Started in CASA

https://casaguides.nrao.edu/index.php/Getting_Started_in_CASA



Pay attention to the logger window! Most tasks write important info to this window. All logger messages are also saved into a file labeled 'casapy.log' in the working directory

Working with Tasks

List available tasks
tasklist()

Get help info on a task
help(tclean)

Load default task params
default(tclean)

Review inputs
inp

Run task
go

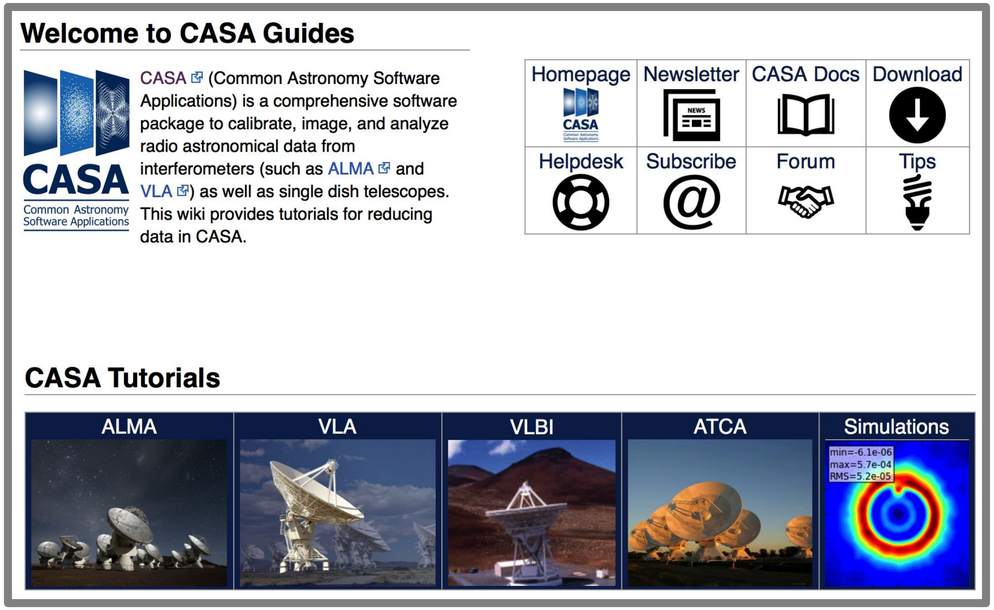
Restore previous params
tget(tclean)

```
CASA <15>: default(tclean)

CASA <16>: inp
---------> inp()
#  tclean :: Radio Interferometric Image Reconstruction
vis                 =          ''        #  Name of input visibility file
selectdata          =        True        #  Enable data selection paramet
     field          =          ''        #  field(s) to select
     spw            =          ''        #  spw(s)/channels to select
     timerange      =          ''        #  Range of time to select from
     uvrange        =          ''        #  Select data within uvrange
     antenna        =          ''        #  Select data based on antenna/
     scan           =          ''        #  Scan number range
     observation    =          ''        #  Observation ID range
     intent         =          ''        #  Scan Intent(s)

datacolumn          = 'corrected'        #  Data column to image(data,cor
imagename           =          ''        #  Pre-name of output images
imsize              =       [100]        #  Number of pixels
cell                = ['1arcsec']        #  Cell size
phasecenter         =          ''        #  Phase center of the image
stokes              =         'I'        #  Stokes Planes to make
projection          =       'SIN'        #  Coordinate projection (SIN, H
startmodel          =          ''        #  Name of starting model image
specmode            =       'mfs'        #  Spectral definition mode (mfs
     reffreq        =          ''        #  Reference frequency

gridder             =  'standard'        #  Gridding options (standard, w
     vptable        =          ''        #  Name of Voltage Pattern table
     pblimit        =         0.2        #  >PB gain level at which to cu

deconvolver         =    'hogbom'        #  Minor cycle algorithm (hogbom
restoration         =        True        #  Do restoration steps (or not)
     restoringbeam  =          []        #  Restoring beam shape to use.
     pbcor          =       False        #  Apply PB correction on the ou

outlierfile         =          ''        #  Name of outlier-field image d
weighting           =   'natural'        #  Weighting scheme (natural,uni
     uvtaper        =          []        #  uv-taper on outer baselines i

niter               =           0        #  Maximum number of iterations
usemask             =      'user'        #  Type of mask(s) for deconvolu
     mask           =          ''        #  Mask (a list of image name(s)
     pbmask         =         0.0        #  primary beam mask

restart             =        True        #  True : Re-use existing images
savemodel           =      'none'        #  Options to save model visibil
calcres             =        True        #  Calculate initial residual im
calcpsf             =        True        #  Calculate PSF
parallel            =       False        #  Run major cycles in parallel

CASA <17>:
```

Some CASA Words of Wisdom

CASA is a powerful tool, but it remains under active development . . .

- ***Bugs do exist*** *in CASA* – when in doubt contact the helpdesk
- Some aspects of pipeline heuristics may be ***dependent on the observing date and CASA version*** used to produce the data products – always check documentation
- ***Task names and inputs do change*** as new features are enabled/improved – always review CASA release notes
- CASA has many ***online resources*** – use them!



Outline

- CASA overview
- **Simulation guidelines**
- CASA simulation example

General Simulation Guidelines

- **Simulations are *not required* for ALMA proposals**, but they may strengthen proposals in some cases
- If performed, simulations should appear in the Science Justification (and optionally in the technical justification)
- The Helpdesk provides assistance with simulations if needed!

Simulations may help justify observations of:

- **Sources with complex, extended morphologies:** To demonstrate the need for specific configs, combinations of configs, or array components (12m-array, ACA, TP)
- **Low-elevation sources**: To verifiy adequate uv-coverage, check synthesized beam shape, etc.
- **Distant analog to a given source model:** To ensure angular resolution is sufficient

Steps for Simulating Observations

1) Use the ALMA sensitivity calculator to determine the necessary observing time for your science goals

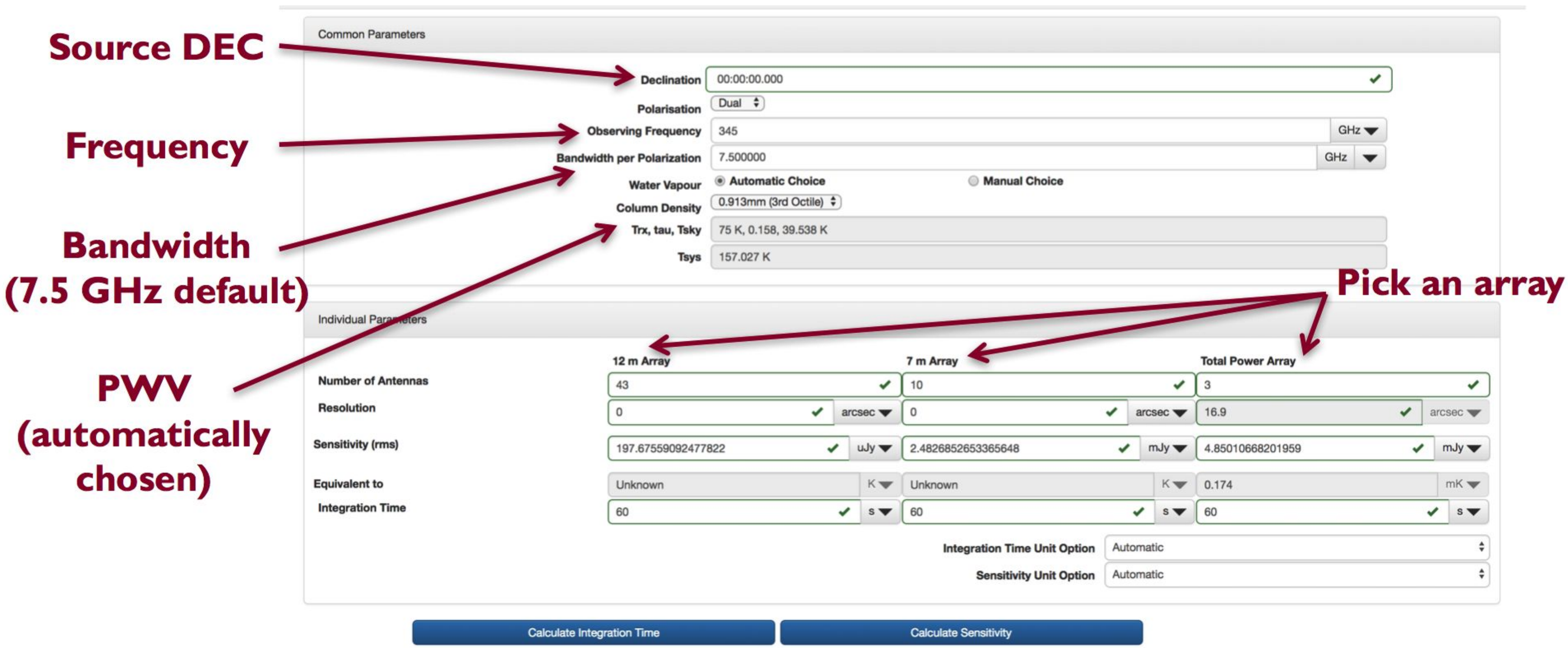
2) Generate simulated visibilities using the 'simobserve' task in CASA (takes FITS input)

3) Image, analyze, and evaluate the resulting visibilities

Requires trial and error - repeat for different antenna configurations, observing times, etc.

Sensitivity Calculator

https://almascience.nrao.edu/tools/proposing/sensitivity-calculator



Either enter a sensitivity (rms) and calculate integration time or enter an integration time and calculate sensitivity

Outline

- CASA overview
- Simulation guidelines
- **CASA simulation example**

Simulations in CASA

CASA can take an input model image, smooth it, change its location/resolution, and create a mock dataset + images

Important tasks/tools for simulations:

- SIMOBSERVE
- SIMALMA
- SIMANALYZE
- SIMULATOR TOOLKIT
- PLOTMS
- TCLEAN
- VIEWER (CARTA)
- EXPORTFITS

WARNING: Do not use predicted sensitivity from simulated images for proposals – use values calculated in the OT or Sensitivity Calculator

Simulations Examples on CASA Guides

Online info: https://casaguides.nrao.edu

Welcome to CASA Guides

CASA (Common Astronomy Software Applications) is a comprehensive software package to calibrate, image, and analyze radio astronomical data from interferometers (such as ALMA and VLA) as well as single dish telescopes. This wiki provides tutorials for reducing data in CASA.

Homepage	Newsletter	CASA Docs	Download
Helpdesk	Subscribe	Forum	Tips

Simulation tutorials

CASA Tutorials

Simulations in CASA

Tutorials

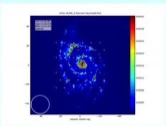
A detailed overview of **how to simulate data in CASA** is given in the "Simulation" pages of the CASA Docs documentation. The following tutorials provide additional examples:

Simulating ngVLA Data (CASA 5.4)

This tutorial shows how to create simulated data for the next generation Very Large Array (ngVLA) either by using simobserve or the sm toolkit. Additionally, it shows how to estimate the scaling parameter for adding thermal noise using the sm.setnoise function and the simplenoise parameter.

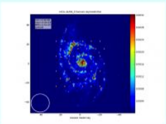
Simalma (CASA 5.4)

This tutorial demonstrates how to use **simalma**, a task that simplifies simulations that include the main 12-m array plus the ACA. Like the previous guide, this one is of particular interest to those wishing to explore multi-component ALMA observations.



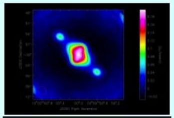
ACA Simulation (CASA 5.4)

A tutorial for simulating ALMA observations that use multiple configurations or use the 12-meter array in combination with the ALMA Compact Array. This tutorial demonstrates combining data from each ALMA component "by hand". This guide is of particular interest to those wishing to explore using the 12-m array in combination with the ACA, and those interested in combining data from multiple 12-m array configurations.



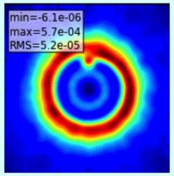
Simulation Guide Component Lists (CASA 5.4)

Tutorial for simulating data based on multiple sources (using both a FITS image and a component list). If you are interested in simulating from a list of simple sources (point, Gaussian, disk), rather than or in addition to a sky model image, then read the considerations here.



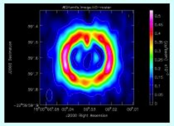
Protoplanetary Disk Simulation (CASA 5.4)

A sky model with a lightly annotated script that simulates a protoplanetary disk. Uses a theoretical model of dust continuum from Sebastian Wolff, scaled to the distance of a nearby star. This is another fairly generic simulation - if you're short on time, you probably don't need to go through this one and the New Users guide, but it can be useful to go through multiple examples.



Protoplanetary Disk Simulation - VLA (CASA 5.4)

This tutorial explains the steps for simulating VLA observations using the same protoplanetary disk sky model that was used for the analogous ALMA tutorial. Observational and analysis parameters are changed step by step and the results are compared to the VLA exposure calculator.



Advanced: Corrupting Simulated Data (Simulator Tool)

simobserve calls methods in the **simulator** tool. For advanced CASA users, the 'simulator' tool has methods that can add to simulated data: phase delay variations, gain fluctuations and drift, cross-polarization, and bandpass and pointing errors. 'simulator' also has more flexibility than simobserve in adding thermal noise. The tutorial linked from this page describes the simulation of data using the task interface only. To learn more about the 'simulator' tool, see the CASA Toolkit Reference Manual. An examples of advanced techniques for corrupting a simulated MeasurementSet can be found in this CASA Guide on Corrupting Simulated Data (Simulator Tool).

Simulations in CASA

Tutorials

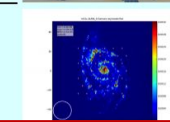
A detailed overview of **how to simulate data in CASA** is given in the "Simulation" pages of the CASA Docs documentation. The following tutorials provide additional examples:

Simulating ngVLA Data (CASA 5.4)

This tutorial shows how to create simulated data for the next generation Very Large Array (ngVLA) either by using simobserve or the sm toolkit. Additionally, it shows how to estimate the scaling parameter for adding thermal noise using the sm.setnoise function and the simplenoise parameter.

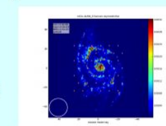
Simalma (CASA 5.4)

This tutorial demonstrates how to use **simalma**, a task that simplifies simulations that include the main 12-m array plus the ACA. Like the previous guide, this one is of particular interest to those wishing to explore multi-component ALMA observations.



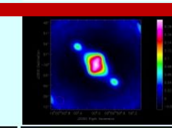
ACA Simulation (CASA 5.4)

A tutorial for simulating ALMA observations that use multiple configurations or use the 12-meter array in combination with the ALMA Compact Array. This tutorial demonstrates combining data from each ALMA component "by hand". This guide is of particular interest to those wishing to explore using the 12-m array in combination with the ACA, and those interested in combining data from multiple 12-m array configurations.



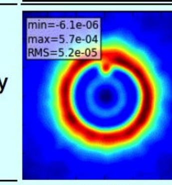
Simulation Guide Component Lists (CASA 5.4)

Tutorial for simulating data based on multiple sources (using both a FITS image and a component list). If you are interested in simulating from a list of simple sources (point, Gaussian, disk), rather than or in addition to a sky model image, then read the considerations here.



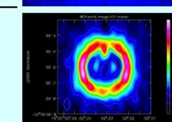
Protoplanetary Disk Simulation (CASA 5.4)

A sky model with a lightly annotated script that simulates a protoplanetary disk. Uses a theoretical model of dust continuum from Sebastian Wolff, scaled to the distance of a nearby star. This is another fairly generic simulation - if you're short on time, you probably don't need to go through this one and the New Users guide, but it can be useful to go through multiple examples.



Protoplanetary Disk Simulation - VLA (CASA 5.4)

This tutorial explains the steps for simulating VLA observations using the same protoplanetary disk sky model that was used for the analogous ALMA tutorial. Observational and analysis parameters are changed step by step and the results are compared to the VLA exposure calculator.



Advanced: Corrupting Simulated Data (Simulator Tool)

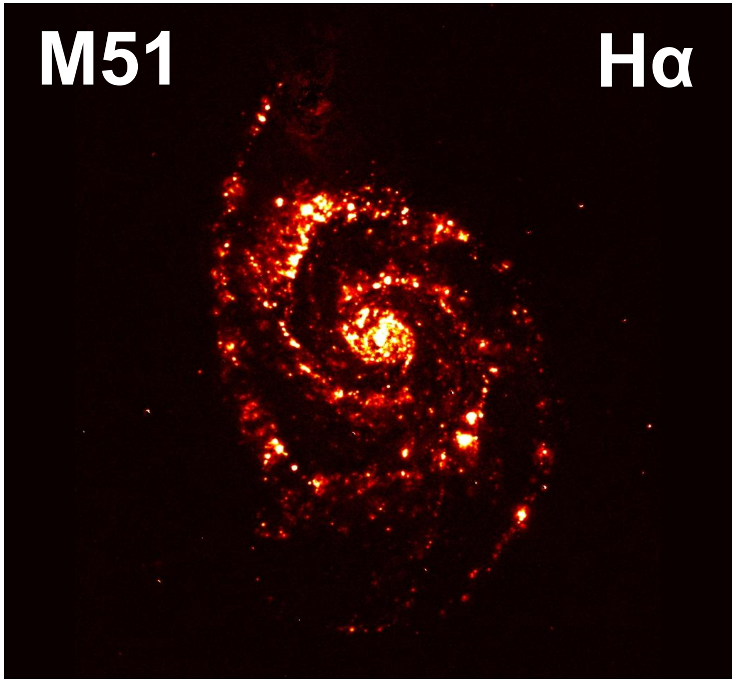
simobserve calls methods in the **simulator** tool. For advanced CASA users, the 'simulator' tool has methods that can add to simulated data: phase delay variations, gain fluctuations and drift, cross-polarization, and bandpass and pointing errors. 'simulator' also has more flexibility than simobserve in adding thermal noise. The tutorial linked from this page describes the simulation of data using the task interface only. To learn more about the 'simulator' tool, see the CASA Toolkit Reference Manual. An examples of advanced techniques for corrupting a simulated MeasurementSet can be found in this CASA Guide on Corrupting Simulated Data (Simulator Tool).

Simulations in CASA: M51

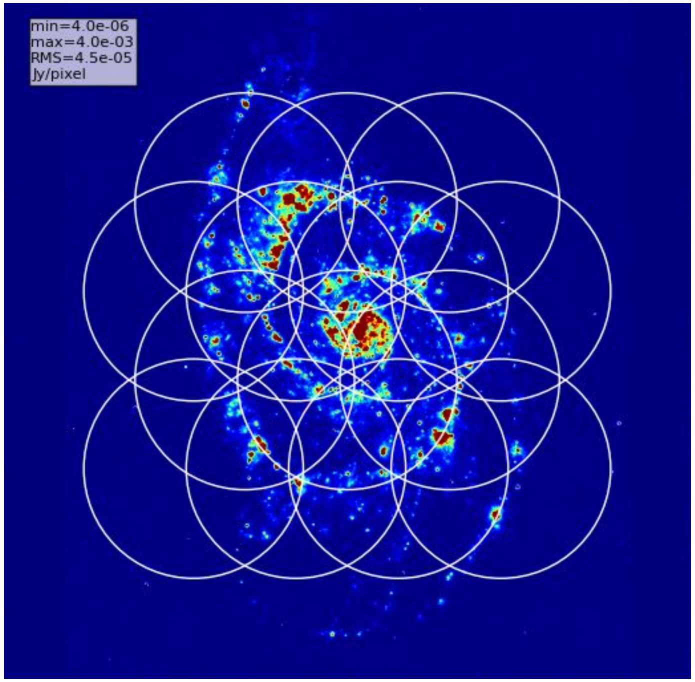
https://casaguides.nrao.edu/index.php/ACA_Simulation_(CASA_5.4)

curl https://casaguides.nrao.edu/images/3/3f/M51ha.fits.txt -f -o M51ha.fits

Input sky model



Model + pointings



Mosaicking + combining multiple configs/array elements

Notes on Mosaicking/Multiple Configs

- **Mosaics combine multiple pointings into a single image**
  - If your target does not fit within 1/3 of the primary beam width, mosaicking may be necessary
  - **Beware of bug** in CASA 5.4.0 mosaicking – update!

- **Rule of thumb** for multi-config imaging with ALMA: Aim for relative observing times that yield similar S/N ratios
  - Use the OT/sensitivity calculator, simulations
  - **Beware of flux bias** during multi-configuration imaging and analysis (see **Jorsater & van Moorsel 1995**)

- Multi-configuration projects are processed and delivered separately, but ***PI's are responsible for combining these products together*** (e.g., using feathering)

Notes on Mosaicking/Multiple Configs

Table A-2 from the Cycle 7 Proposer's Guide

Most Extended configuration	Allowed Compact configuration pairings	Extended 12-m Array Multiplier	Multiplier if compact 12-m Array needed	Multiplier if 7-m Array needed	Multiplier if TP Array needed and allowed
7-m Array	TP			1	1.7
C43-1	7-m Array & TP	1		7.0	11.9
C43-2	7-m Array & TP	1		4.7	7.9
C43-3	7-m Array & TP	1		2.4	4.1
C43-4	C43-1 & 7-m Array & TP	1	0.34	2.4	4.0
C43-5	C43-2 & 7-m Array & TP	1	0.26	1.2	2.1
C43-6	C43-3 & 7-m Array & TP	1	0.25	0.6	1.0
C43-7	C43-4	1	0.23		
C43-8	C43-5	1	0.22		
C43-9	C43-6	1	0.21		
C43-10	-	1			

simobserve example

Observing time ratios for allowed combinations of configurations and arrays in Cycle 7 (see Chapter 7 of the Technical Handbook)

Simobserve

```
CASA <5>: inp
--------> inp()
#  simobserve :: visibility simulation task
project             =     'm51c'        #  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)
```

- **project:** Name of folder for simulation output
- **skymodel:** Input FITS image for simulations
- **inbright**: Peak brightness - assumes Jy/pixel units
- **indirection:** Sky coordinates of map center
- **incell**: Spatial pixel size (include units)
- **incenter:** Central observing frequency
- **inwidth:** Channel width (set to 7.5 GHz for continuum)

Simobserve

```
complist            =         ''      #  componentlist to observe
setpointings        =       True
     integration    =      '10s'      #  integration (sampling) time
     direction      =         ''      #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize        =  '1arcmin'      #  angular size of map or "" to cover model
     maptype        =      'hex'      #  hexagonal, square (raster), ALMA, etc
     pointingspacing =  '9arcsec'     #  spacing in between pointings or "0.25PB" or "" for ALMA default INT=lambda/D/sqrt(3),
                                      #   SD=lambda/D/3
```

- **setpointings:** Calculate mosaic pointing positions; if False, ptgfile parameter must be set (see “help”)
- **integration:** Sampling time interval
- **direction:** Mosaic center direction (defaults to input image center). Can also be a list of pointings.
- **mapsize:** Angular size of map. Defaults to model image size
- **maptype:** Sets pattern for mosaic if not specified elsewhere
- **pointingspacing:** Spacing in between pointings for mosaic (leave unset for automatic pointing spacing determination)

Simobserve

```
obsmode             =      'int'         #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist    = 'ALMA;0.5arcsec'   #  interferometer antenna position file
     refdate        = '2014/05/21'       #  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      =    '3600s'         #  total time of observation or number of repetitions
     caldirection   =         ''         #  pt source calibrator [experimental]
     calflux        =      '1Jy'

outframe            =     'LSRK'         #  spectral frame of MS to create
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
```

- **obsmode:** “int” for interferometers or “sd” for singledish
- **antennalist:** Antenna configuration file (“alma.cycle6.3.cfg”) or angular resolution (“ALMA; 0.5arcsec”)
- **refdate:** Date of simulated observation (e.g. “yyyy/mm/dd”)
- **hourangle:** hour angle of observation (HA = LST - RA; sources transit at HA = 0); impacts source elevation
- **totaltime:** Total on-source observing time

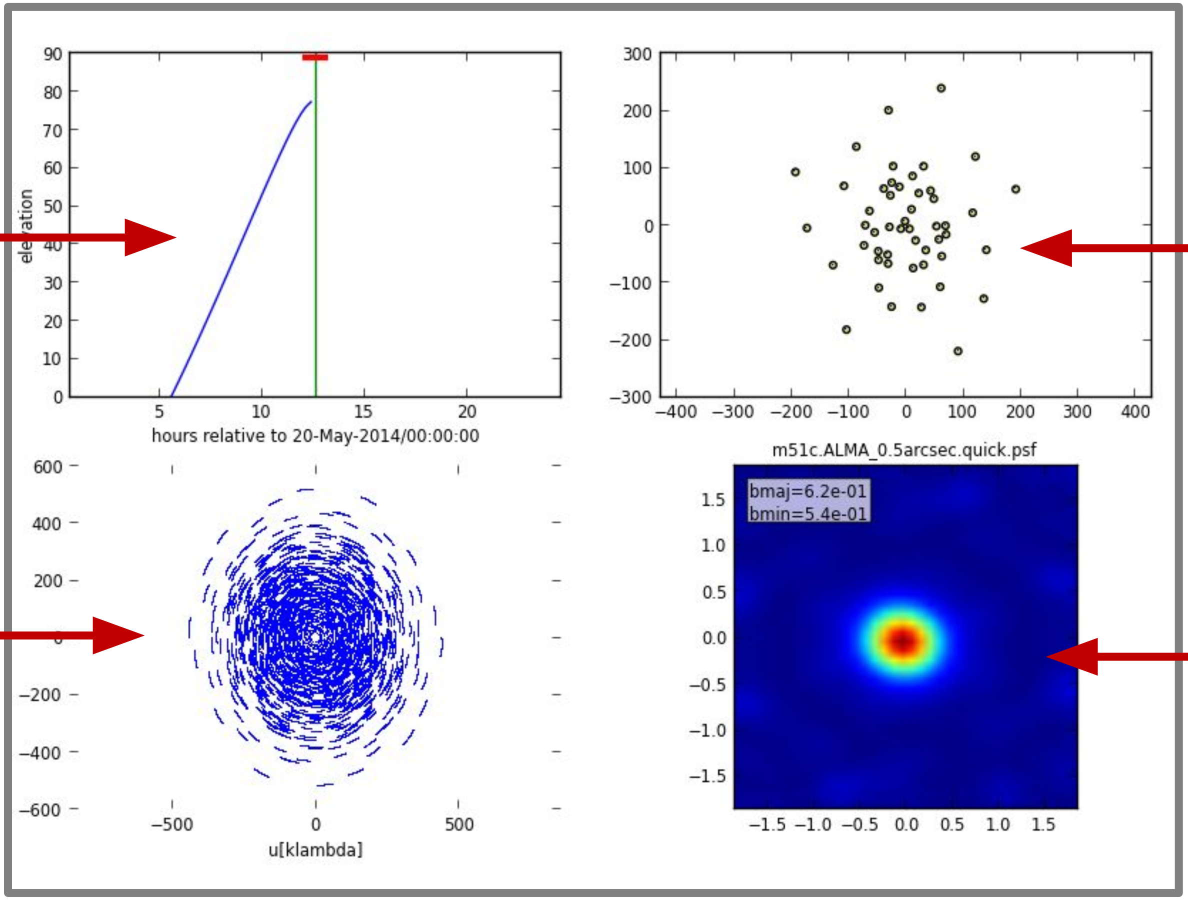
Note on Antenna Configs in CASA

```
mydir = os.getenv('CASAPATH').split()[0]+'/data/alma/simmos/'
os.system('ls ' + mydir)
```

```
CASA <17>: mydir = os.getenv('CASAPATH').split()[0]+'/data/alma/simmos/'

CASA <18>: os.system('ls ' + mydir)
WSRT.cfg                        alma.cycle4.2.cfg               alma.out11.cfg                  atca_ew214.cfg
aca.all.cfg                     alma.cycle4.3.cfg               alma.out12.cfg                  atca_ew352.cfg
aca.cycle1.cfg                  alma.cycle4.4.cfg               alma.out13.cfg                  atca_ew367.cfg
aca.cycle2.i.cfg                alma.cycle4.5.cfg               alma.out14.cfg                  atca_h168.cfg
aca.cycle2.ns.cfg               alma.cycle4.6.cfg               alma.out15.cfg                  atca_h214.cfg
aca.cycle3.cfg                  alma.cycle4.7.cfg               alma.out16.cfg                  atca_h75.cfg
aca.cycle4.cfg                  alma.cycle4.8.cfg               alma.out17.cfg                  atca_ns214.cfg
aca.cycle5.cfg                  alma.cycle4.9.cfg               alma.out18.cfg                  carma.a.cfg
aca.cycle6.cfg                  alma.cycle5.1.cfg               alma.out19.cfg                  carma.b.cfg
aca.i.cfg                       alma.cycle5.10.cfg              alma.out20.cfg                  carma.c.cfg
aca.ns.cfg                      alma.cycle5.2.cfg               alma.out21.cfg                  carma.d.cfg
aca.tp.cfg                      alma.cycle5.3.cfg               alma.out22.cfg                  carma.e.cfg
aca_cycle1.cfg                  alma.cycle5.4.cfg               alma.out23.cfg                  meerkat.cfg
alma.all.cfg                    alma.cycle5.5.cfg               alma.out24.cfg                  ngvla-core-revB.cfg
alma.cycle0.compact.cfg         alma.cycle5.6.cfg               alma.out25.cfg                  ngvla-gb-vlba-revB.cfg
alma.cycle0.extended.cfg        alma.cycle5.7.cfg               alma.out26.cfg                  ngvla-plains-revB.cfg
alma.cycle1.1.cfg               alma.cycle5.8.cfg               alma.out27.cfg                  ngvla-revB.cfg
alma.cycle1.2.cfg               alma.cycle5.9.cfg               alma.out28.cfg                  ngvla-sba-revB.cfg
alma.cycle1.3.cfg               alma.cycle6.1.cfg               alma_cycle1_1.cfg               pdbi-a.cfg
alma.cycle1.4.cfg               alma.cycle6.10.cfg              alma_cycle1_2.cfg               pdbi-b.cfg
alma.cycle1.5.cfg               alma.cycle6.2.cfg               alma_cycle1_3.cfg               pdbi-c.cfg
alma.cycle1.6.cfg               alma.cycle6.3.cfg               alma_cycle1_4.cfg               pdbi-d.cfg
alma.cycle2.1.cfg               alma.cycle6.4.cfg               alma_cycle1_5.cfg               sma.compact.cfg
alma.cycle2.2.cfg               alma.cycle6.5.cfg               alma_cycle1_6.cfg               sma.compact.n.cfg
alma.cycle2.3.cfg               alma.cycle6.6.cfg               atca_1.5a.cfg                   sma.extended.cfg
alma.cycle2.4.cfg               alma.cycle6.7.cfg               atca_1.5b.cfg                   sma.subcompact.cfg
alma.cycle2.5.cfg               alma.cycle6.8.cfg               atca_1.5c.cfg                   sma.vextended.cfg
alma.cycle2.6.cfg               alma.cycle6.9.cfg               atca_1.5d.cfg                   viewer.last
alma.cycle2.7.cfg               alma.out01.cfg                  atca_122c.cfg                   vla.a.cfg
alma.cycle3.1.cfg               alma.out02.cfg                  atca_6a.cfg                     vla.b.cfg
alma.cycle3.2.cfg               alma.out03.cfg                  atca_6b.cfg                     vla.bna.cfg
alma.cycle3.3.cfg               alma.out04.cfg                  atca_6c.cfg                     vla.c.cfg
alma.cycle3.4.cfg               alma.out05.cfg                  atca_6d.cfg                     vla.cnb.cfg
alma.cycle3.5.cfg               alma.out06.cfg                  atca_750a.cfg                   vla.d.cfg
alma.cycle3.6.cfg               alma.out07.cfg                  atca_750b.cfg                   vla.dnc.cfg
alma.cycle3.7.cfg               alma.out08.cfg                  atca_750c.cfg                   vlba.cfg
alma.cycle3.8.cfg               alma.out09.cfg                  atca_750d.cfg
alma.cycle4.1.cfg               alma.out10.cfg                  atca_all.cfg
```

Output from Simobserve

Elevation vs. LST for observations (in red)



Antenna positions

uv coverage

Synthesized beam

Output from Simobserve

```
CASA <22>: ls
m51c.ALMA_0.5arcsec.ms/                m51c.ALMA_0.5arcsec.simobserve.last
m51c.ALMA_0.5arcsec.noisy.ms/          m51c.ALMA_0.5arcsec.skymodel/
m51c.ALMA_0.5arcsec.observe.png        m51c.ALMA_0.5arcsec.skymodel.flat/
m51c.ALMA_0.5arcsec.ptg.txt            m51c.ALMA_0.5arcsec.skymodel.png
m51c.ALMA_0.5arcsec.quick.psf/
```

Don’t forget to check the log – useful info is printed there!

```
…observe::::   Position: [1761.87, -4307.63, -1977.71]
…bserve::::+   Epoch: 56798::00:00:00.0000
…bserve::::+   Epoch: 56797::12:38:05.3948
…bserve::::+   LAST of rise= 16:52:10.61
…observe::::   LAST of  set= 07:09:16.22
…observe::::   UTC of rise= 2014/05/19/05:35:55
…bserve::::+   UTC of  set= 2014/05/19/19:50:41
…ephemeris::   peak=20-May-2014/12:38:05
…er::open()    Opening MeasurementSet /Users/knyland/Desktop/outreach/ALMA_ambassadors/si
…imobserve::   using default model cell {'value': 0.1, 'unit': 'arcsec'} for PSF calculat
…fineimage()   Defining image properties:nx=128 ny=128 cellx='0.1arcsec' celly='0.1arcsec
…fineimage()    phaseCenter='field-0 ' mStart='Radialvelocity: 0' qStep='0 '' mFreqStart=
…matepsfs()    Calculating approximate PSFs using full sky equation
…TMachine()    Performing interferometric gridding...
…eApproxPSFs   bmaj: 0.620481", bmin: 0.536341", bpa: 80.227 deg
```

Adding 7m + TP ACA Simulations

There will be additional time to try the ACA simobserve commands after this talk

```
tget(simobserve)
integration          = "10s"
mapsize              = "1arcmin"
maptype              = "hex"
pointingspacing      = ''
obsmode              = "int"
refdate              = "2012/12/02" # NOTE: change the date from 12m array sims
antennalist          = "aca.i.cfg"
totaltime            = "2.4h"
simobserve()
```

7m array

```
tget(simobserve)
integration          = "10s"
mapsize              = "1.3arcmin"
maptype              = "square"
obsmode              = "sd"
sdantlist            = "aca.tp.cfg"
sdant                = 0
refdate              = "2012/12/01" # NOTE: change the date from 7m and 12m array sims
totaltime            = "4.1h"
simobserve()
```

TP

NOTE: When simulating observations that combine multiple configs/arrays, be sure to change the refdate parameter in simobserve

Analyzing the Output of Simobserve

Use **simanalyze** in CASA, which creates images using **clean**

–OR–

Use **tclean** to image the resulting visibilities

Another approach to ALMA simulations is the **simalma** task:

https://casaguides.nrao.edu/index.php/Simalma

But, **simobserve** is more generalized and has more features and flexibility - best for complex simulations

Simanalyze Inputs

```
#  simanalyze :: image and analyze measurement sets created with simobserve
project             =      'sim'        #  root prefix for output file names
image               =       True        #  (re)image $project.*.ms to $project.image
     vis            =  'default'        #  Measurement Set(s) to image
     modelimage     =         ''        #  lower resolution prior image to use in clean e.g. existing total power
     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
```

- **project:** Name of same folder from simobserve run(s)
- **vis:** Output corrupted (noisy) dataset from simobserve
- **featherimage:** single-dish or TP map to combine with interferometric image using the ‘feathering’ technique

Additional params imsize, imdirection, cell, interactive, niter, threshold, weighting, mask, outertaper – same as in **tclean**

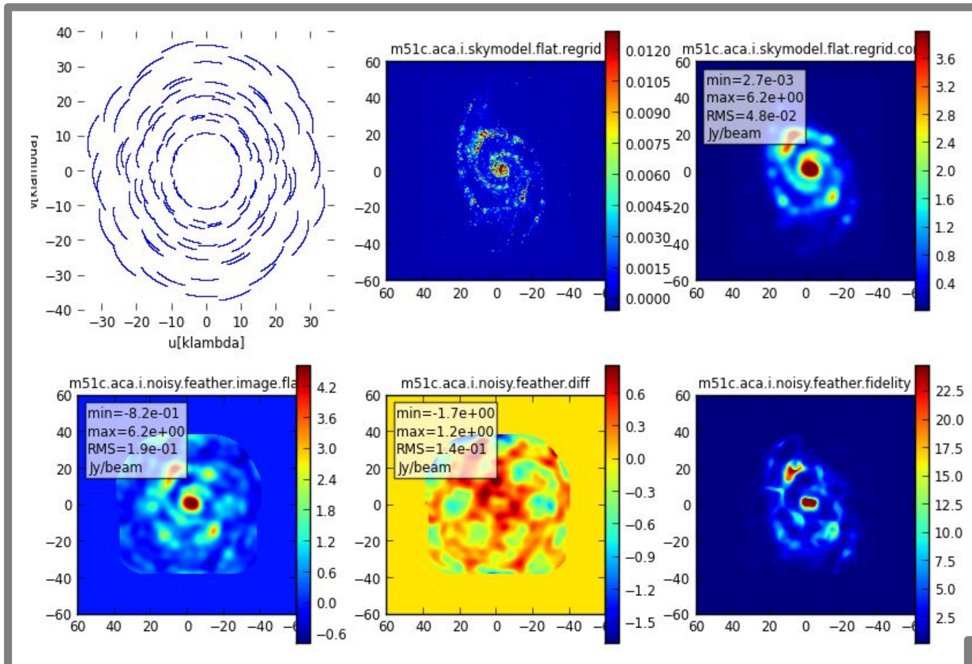
Simanalyze Inputs

```
analyze          =       True         #  (only first 6 selected outputs will be displayed)
     showuv      =       True         #  display uv coverage
     showpsf     =       True         #  display synthesized (dirty) beam (ignored in single dish simulation)
     showmodel   =       True         #  display sky model at original resolution
     showconvolved =    False         #  display sky model convolved with output clean beam
     showclean   =       True         #  display the synthesized image
     showresidual =     False         #  display the clean residual image (ignored in single dish simulation)
     showdifference =    True         #  display difference between output cleaned image and input model sky :
                                      #   clean beam
     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           =      False         #  only print information [experimental; only for interfermetric data]
logfile          =         ''
```

- **analyze:** Set to True to open options for diagnostic plots
  - showuv = True
  - showpsf = False
  - showmodel = True
  - showconvolved = True
  - showclean = True
  - showresidual = False
  - showdifference = True
  - showfidelity = True

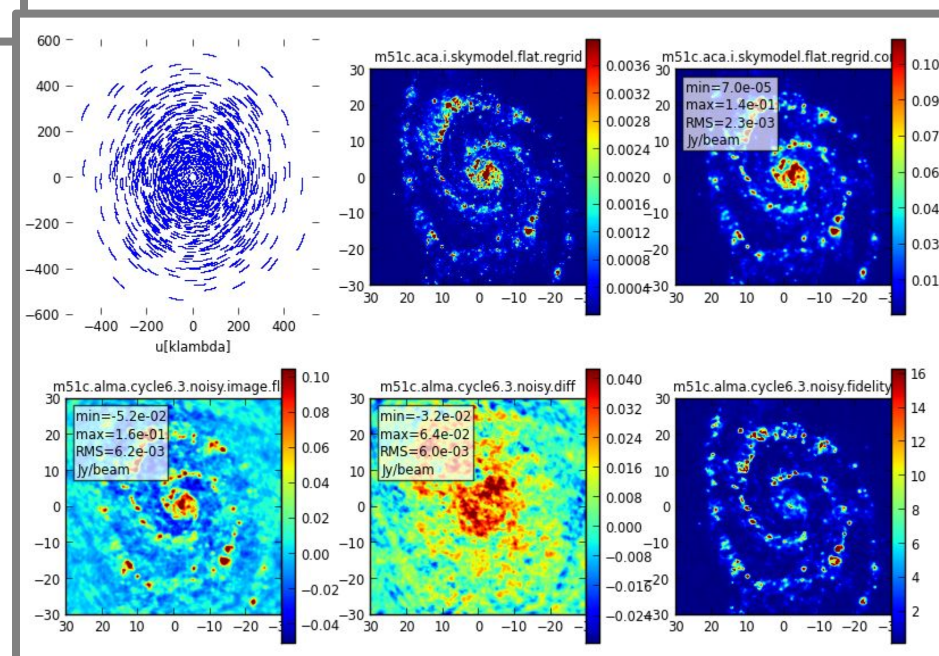
fidelity image measures how well the simulated output matches the convolved input model

Running Simanalyze

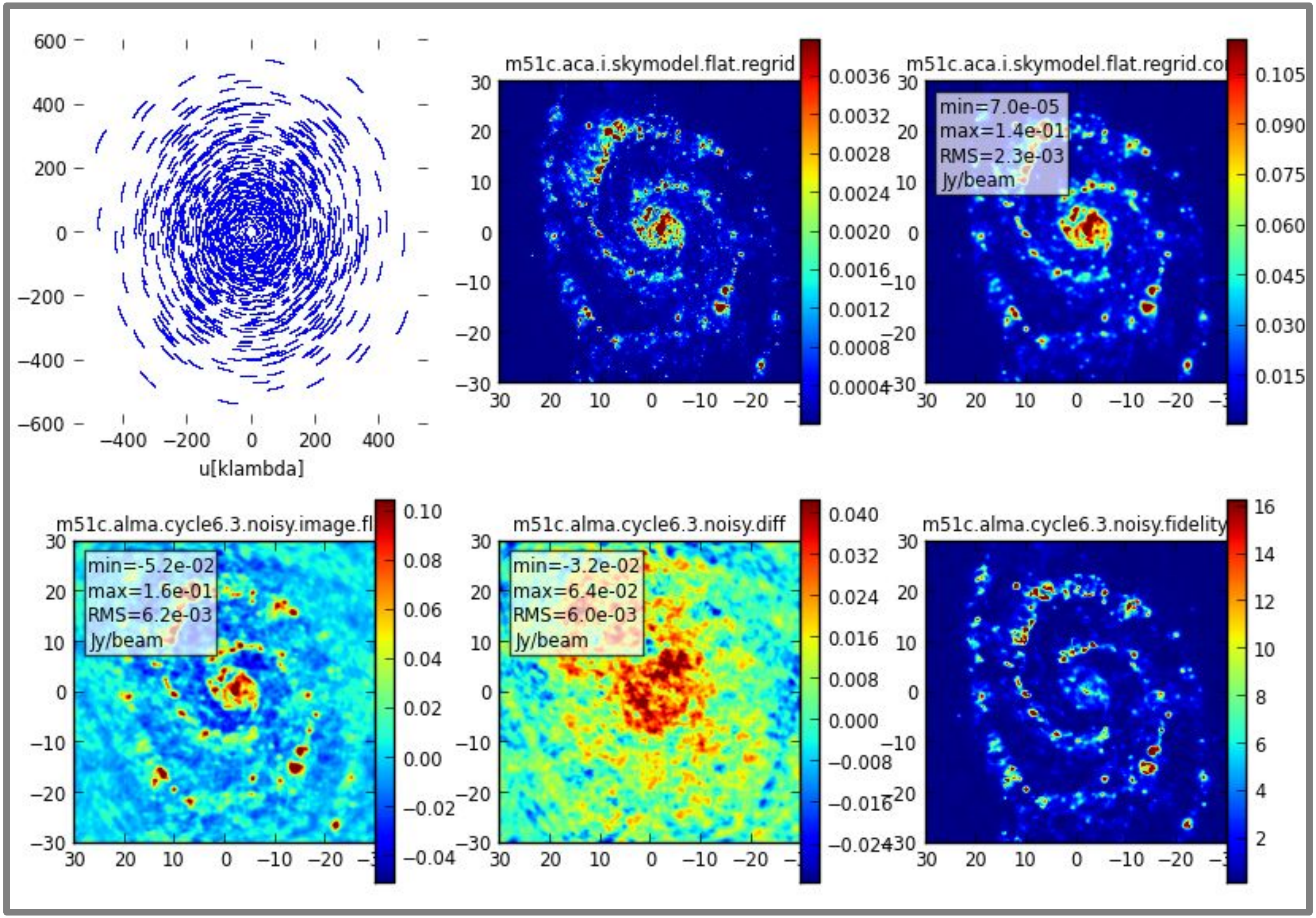


Simanalyze is first run on the simulated ACA 7m + TP noisy datasets and creates a feathered ACA image

Next, simanalyze is run on the noisy simulated 12m array dataset using the feathered ACA image from the last run as a starting model



Simanalyze Output



NRAO

www.nrao.edu
science.nrao.edu

NSF

Associated Universities, Inc.

Atacama Large Millimeter/submillimeter Array
Karl G. Jansky Very Large Array
Robert C. Byrd Green Bank Telescope
Very Long Baseline Array

NRAO

Additional Slides

Atacama Large Millimeter/submillimeter Array
Karl G. Jansky Very Large Array
Robert C. Byrd Green Bank Telescope
Very Long Baseline Array

Basic imaging guidelines

Intro to Tclean

- Imaging capabilities of **clean** have been refactored and improved in **tclean** in the current version of CASA
- The ALMA pipeline now uses **tclean** instead of **clean** for imaging
- **Major syntax changes** are summarized here: https://casaguides.nrao.e du/index.php/TCLEAN_an d_ALMA
- **WARNING:** Avoid killing **tclean/clean** using CTRL+C □ this may corrupt your dataset

```
#  tclean :: Radio Interferometric
vis                 =           ''
selectdata          =         True
     field          =           ''
     spw            =           ''
     timerange      =           ''
     uvrange        =           ''
     antenna        =           ''
     scan           =           ''
     observation    =           ''
     intent         =           ''

datacolumn          =  'corrected'
imagename           =           ''
imsize              =        [100]
cell                = ['1arcsec']
phasecenter         =           ''
stokes              =          'I'
projection          =        'SIN'
startmodel          =           ''
specmode            =        'mfs'
     reffreq        =           ''

gridder             =   'standard'
     vptable        =           ''
     pblimit        =          0.2

deconvolver         =     'hogbom'
restoration         =         True
     restoringbeam  =           []
     pbcor          =        False

outlierfile         =           ''
weighting           =    'natural'
     uvtaper        =           []

niter               =            0
usemask             =       'user'
     mask           =           ''
     pbmask         =          0.0

restart             =         True
savemodel           =       'none'
calcres             =         True
calcpsf             =         True
parallel            =        False
```

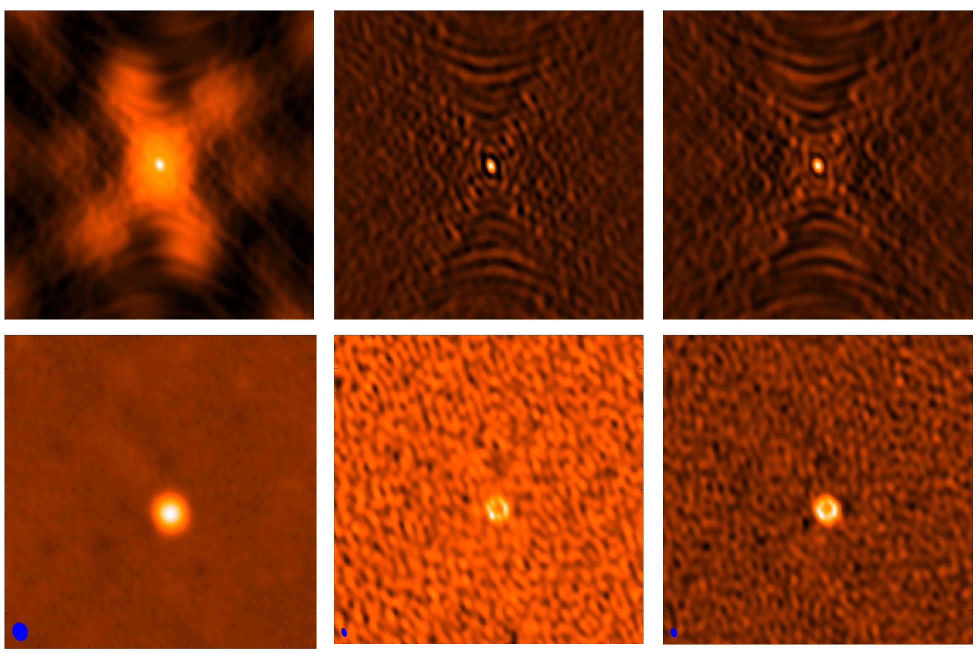
Visibility Weighting

	Robust/Uniform	Natural	Taper
Resolution	**higher**	medium	lower
Sidelobes	lower	higher	depends
Point source sensitivity	lower	**maximum**	lower
Extended source sensitivity	lower	medium	**higher**

Natural Uniform Robust=0

Dirty beam

CLEAN image



Multiplying the uv distribution, S(u,v), by a weighting function, W(u,v), changes the resolution, sensitivity, and sidelobe levels of the dirty beam

Recommended starting point: **briggs** weighting with **robust = 0.5**

Tclean Inputs

```
#  tclean :: Radio Interferometric Image Reconstruction
vis                 =           ''        #  Name of input visibility file(s)
selectdata          =        False        #  Enable data selection parameters
datacolumn          = 'corrected'         #  Data column to image(data,corrected
imagename           =           ''        #  Pre-name of output images
imsize              =        [100]        #  Number of pixels
cell                = ['1arcsec']         #  Cell size
phasecenter         =           ''        #  Phase center of the image
stokes              =          'I'        #  Stokes Planes to make
projection          =        'SIN'        #  Coordinate projection (SIN, HPX)
startmodel          =           ''        #  Name of starting model image
```

- **vis:** input uv dataset (MS file)
- **imagename:** root name of output images
- **imsize:** size of image in pixels – if possible, image the full widath at half-power of the primary beam (~λ/D)
- **cell:** angular size of each pixel in arcsec – need 5-8 pixels across the synthesized beam (~λ/B_{max})
- **phasecenter:** image center - typically only set for mosaics
- **startmodel:** (optional) – initial clean model; useful to set this to TP image for ALMA 12m-array + TP observations

Tclean Inputs

```
#  tclean :: Radio Interferometric Image Reconstruction
vis                 =          ''        #  Name of input visibility file(s)
selectdata          =       False        #  Enable data selection parameters
datacolumn          = 'corrected'        #  Data column to image(data,corrected
imagename           =          ''        #  Pre-name of output images
imsize              =       [100]        #  Number of pixels
cell                = ['1arcsec']        #  Cell size
phasecenter         =          ''        #  Phase center of the image
stokes              =         'I'        #  Stokes Planes to make
projection          =       'SIN'        #  Coordinate projection (SIN, HPX)
startmodel          =          ''        #  Name of starting model image
```

Use **selectdata = True** to specify field, spw, etc. to be imaged – important when data includes multiple targets/calibrators

```
selectdata          =        True
     field          =          ''
     spw            =          ''
     timerange      =          ''
     uvrange        =          ''
     antenna        =          ''
     scan           =          ''
     observation    =          ''
     intent         =          ''
```

Tclean Inputs

```
specmode            =       'mfs'        #  Spectral definition mode
                                         #   (mfs,cube,cubedata)
    reffreq         =          ''        #  Reference frequency

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

deconvolver         =    'hogbom'        #  Minor cycle algorithm (hogbom,clark,m
                                         #   ultiscale,mtmfs,mem,clarkstokes)
```

- **specmode:** use ‘mfs’ for continuum images and ‘channel/velocity/frequency’ for spectral line imaging*
- **gridder:** ‘standard’ and ‘mosaic’ most common for ALMA
- **deconvolver:** algorithm for reconstructing the sky brightness from the dirty image and the PSF (“deconvolution”)

*For line imaging, you will also need to set the dimensions of the cube, rest frequency, velocity frame, and Doppler definition

Tclean Inputs

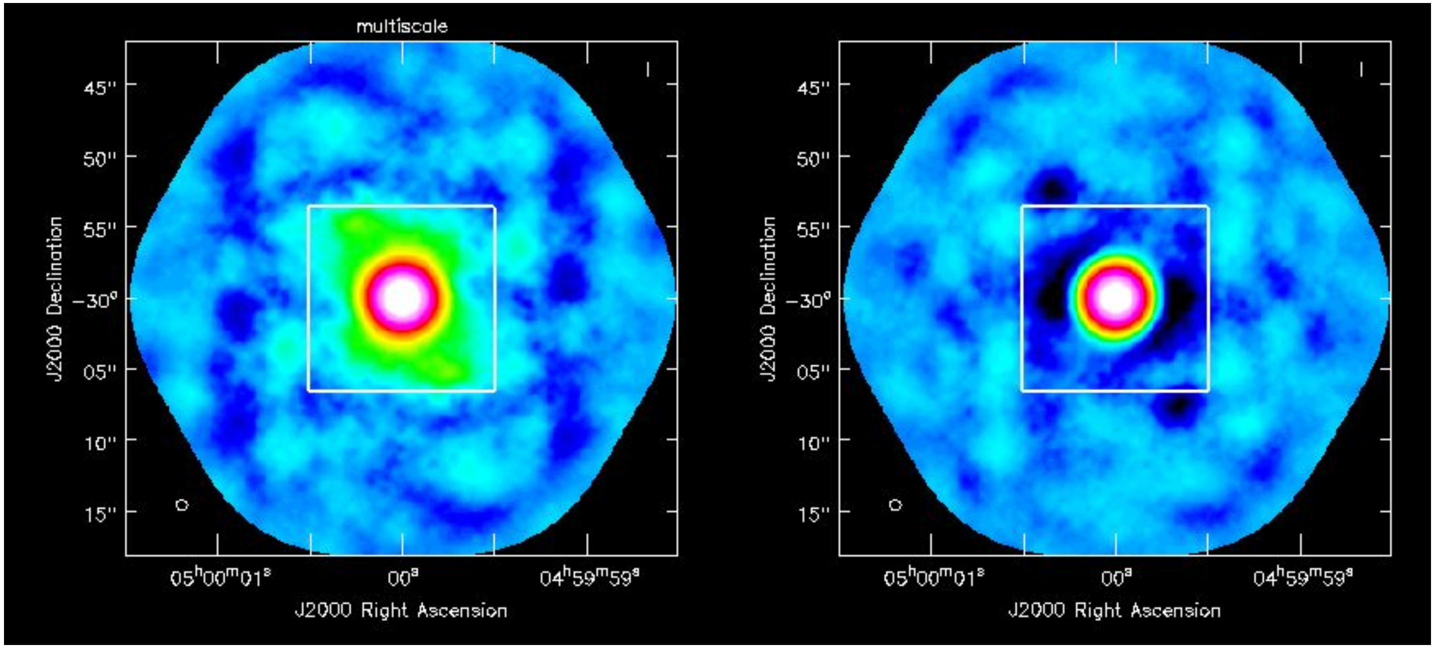
```
                                        #   definitions
weighting            =  'natural'       #  Weighting scheme
                                        #   (natural,uniform,briggs)
     uvtaper         =          []      #  uv-taper on outer baselines in uv-
                                        #   plane

niter                =         100      #  Maximum number of iterations
     gain            =         0.1      #  Loop gain
     threshold       =         0.0      #  Stopping threshold
     cycleniter      =          -1      #  Maximum number of minor-cycle
                                        #   iterations
     cyclefactor     =         1.0      #  Scaling on PSF sidelobe level to
                                        #   compute the minor-cycle stopping
                                        #   threshold.
     minpsffraction  =        0.05      #  PSF fraction that marks the max depth
                                        #   of cleaning in the minor cycle
     maxpsffraction  =         0.8      #  PSF fraction that marks the minimum
                                        #   depth of cleaning in the minor cycle
     interactive     =        True      #  Modify masks and parameters at
                                        #   runtime
```

- **weighting:** natural, uniform or robust
- **uvtaper:** apply Gaussian uv taper to visibilities (helpful for imaging extended/diffuse emission)
- **niter:** maximum number of iterations (niter=0 □ dirty image)
- **threshold:** flux stopping criterion
- **interactive:** run clean interactively or non-interactively

Multiscale

Uses extended clean components to better match emission scales unlike hogbom or clark, which use delta functions

multi-scale **"classic" scale**



Suggested (trial) parameter setting is **scales = [0,5,15]**: **(1)** point source, **(2)** the size of the synthesized beam, and **(3)** 3-5 times the synthesized beam

Tclean Inputs

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

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

- **usemask:** type of clean mask - ‘user’ or ‘auto-multithresh’ useful for ALMA)
- **restart:** If ‘tclean’ is started again with same image name, it will try to continue deconvolution from where it left off. Make sure this is what you want. If not, set a new imagename or move/delete existing image files.
- **savemodel:** controls how CASA stores deconvolution model – set savemodel = ‘modelcolumn’ during self calibration
- **parallel:** implements parallel processing; requires launching CASA with ‘mpicasa’

Notes on Mosaicking/Multiple Configs

θ_{res} (arcsec)	θ_{LAS} (arcsec)	Array combination	Time ratios	Total Time
0.042	< 0.496	C43-10	1	$1.0 \times \Delta_{extended}$
0.042	> 0.496	-	-	-
0.057	< 0.814	C43-9	1	$1.0 \times \Delta_{extended}$
0.057	0.814-4.11	C43-9 + C43-6	1 : 0.21	$1.21 \times \Delta_{extended}$
0.057	> 4.11	-	-	-
0.096	< 1.42	C43-8	1	$1.0 \times \Delta_{extended}$
0.096	1.42-6.7	C43-8 + C43-5	1 : 0.22	$1.22 \times \Delta_{extended}$
0.096	> 6.7	-	-	-
0.211	< 2.58	C43-7	1	$1.0 \times \Delta_{extended}$
0.211	2.58-11.2	C43-7 + C43-4	1 : 0.23	$1.23 \times \Delta_{extended}$
0.211	> 11.2	-	-	-
0.306	< 4.11	C43-6	1	$1.0 \times \Delta_{extended}$
0.306	4.11-16.2	C43-6 + C43-3	1 : 0.25	$1.25 \times \Delta_{extended}$
0.306	16.2-66.7	C43-6 + C43-3 + 7-m	1 : 0.25 : 0.6	$1.8 \times \Delta_{extended}$
0.306	> 66.7	C43-6 + C43-3 + 7-m + TP	1 : 0.25 :0.6 : 1.0	$2.3 \times \Delta_{extended}$
0.545	< 6.7	C43-5	1	$1.0 \times \Delta_{extended}$
0.545	6.7-22.6	C43-5 + C43-2	1 : 0.26	$1.26 \times \Delta_{extended}$
0.545	22.6-66.7	C43-5 + C43-2 + 7-m	1 : 0.26 : 1.21	$2.5 \times \Delta_{extended}$
0.545	> 66.7	C43-5 + C43-2 + 7-m + TP	1 : 0.26 : 1.21 : 2.1	$3.3 \times \Delta_{extended}$
0.918	< 11.2	C43-4	1	$1.0 \times \Delta_{extended}$
0.918	11.2-28.5	C43-4 + C43-1	1 : 0.34	$1.3 \times \Delta_{extended}$
0.918	28.5-66.7	C43-4 + C43-1 + 7-m	1 : 0.34 : 2.4	$3.7 \times \Delta_{extended}$
0.918	> 66.7	C43-4 + C43-1 + 7-m + TP	1 : 0.34 : 2.4 : 4.0	$5.3 \times \Delta_{extended}$
1.42	< 16.2	C43-3	1	$1.0 \times \Delta_{extended}$
1.42	16.2-66.7	C43-3 + 7-m	1 : 2.4	$3.4 \times \Delta_{extended}$
1.42	> 66.7	C43-3 + 7-m + TP	1 : 2.4 : 4.1	$5.1 \times \Delta_{extended}$
2.3	< 22.6	C43-2	1	$1.0 \times \Delta_{extended}$
2.3	22.6-66.7	C43-2 + 7-m	1 : 4.7	$5.7 \times \Delta_{extended}$
2.3	> 66.7	C43-2 + 7-m + TP	1 : 4.7 : 7.9	$8.9 \times \Delta_{extended}$
3.38	< 28.5	C43-1	1	$1.0 \times \Delta_{extended}$
3.38	28.5-66.7	C43-1 + 7-m	1 : 7	$8.0 \times \Delta_{extended}$
3.38	> 66.7	C43-1 + 7-m + TP	1 : 7 : 11.9	$12.9 \times \Delta_{extended}$
12.5	< 66.7	7-m	1	$1.0 \times \Delta_{extended}$
12.5	> 66.7	7-m + TP	1 : 1.7	$2.7 \times \Delta_{extended}$

Table 7.4 from the Cycle 7 Technical Handbook

Guidelines on time ratios, angular scales for different array and config combinations

Note total time < sum of the individual times b/c TP and 7m Array observations are run in parallel

ALMA Bands

Band	Wavelength (mm)	Frequency (GHz)
1	8,6 – 6	35 – 50
2	4,6 – 3,3	65 – 90
3	3,6 – 2,6	84 – 116
4	2,4 – 1,8	125 – 163
5	1,8 – 1,4	163 – 211
6	1,4 – 1,1	211 – 275
7	1,1 – 0,8	275 – 373
8	0,8 – 0,6	385 – 500
9	0,5 – 0,4	602 – 720
10	0,4 – 0,3	787 – 950

Cycle 8 Configurations

	Band	3	4	5	6	7	8	9	10
	Frequency (GHz)	100	150	185	230	345	460	650	870
Config.									
7-m	θ_{res} (arcsec)	12.5	8.35	6.77	5.45	3.63	2.72	1.93	1.44
	θ_{MRS} (arcsec)	66.7	44.5	36.1	29.0	19.3	14.5	10.3	7.67
C-1	θ_{res} (arcsec)	3.38	2.25	1.83	1.47	0.98	0.735	0.52	0.389
	θ_{MRS} (arcsec)	28.5	19.0	15.4	12.4	8.25	6.19	4.38	3.27
C-2	θ_{res} (arcsec)	2.30	1.53	1.24	0.999	0.666	0.499	0.353	0.264
	θ_{MRS} (arcsec)	22.6	15.0	12.2	9.81	6.54	4.9	3.47	2.59
C-3	θ_{res} (arcsec)	1.42	0.943	0.765	0.615	0.41	0.308	0.218	0.163
	θ_{MRS} (arcsec)	16.2	10.8	8.73	7.02	4.68	3.51	2.48	1.86
C-4	θ_{res} (arcsec)	0.918	0.612	0.496	0.399	0.266	0.2	0.141	0.106
	θ_{MRS} (arcsec)	11.2	7.5	6.08	4.89	3.26	2.44	1.73	1.29
C-5	θ_{res} (arcsec)	0.545	0.363	0.295	0.237	0.158	0.118	0.0838	0.0626
	θ_{MRS} (arcsec)	6.7	4.47	3.62	2.91	1.94	1.46	1.03	0.77
C-6	θ_{res} (arcsec)	0.306	0.204	0.165	0.133	0.0887	0.0665	0.0471	0.0352
	θ_{MRS} (arcsec)	4.11	2.74	2.22	1.78	1.19	0.892	0.632	0.472
C-7	θ_{res} (arcsec)	0.211	0.141	0.114	0.0917	0.0612	0.0459	0.0325	0.0243
	θ_{MRS} (arcsec)	2.58	1.72	1.4	1.12	0.749	0.562	0.398	0.297
C-8	θ_{res} (arcsec)	0.096	0.064	0.0519	0.0417	0.0278	-	-	-
	θ_{MRS} (arcsec)	1.42	0.947	0.768	0.618	0.412	-	-	-
C-9	θ_{res} (arcsec)	0.057	0.038	0.0308	0.0248	0.0165	-	-	-
	θ_{MRS} (arcsec)	0.814	0.543	0.44	0.354	0.236	-	-	-
C-10	θ_{res} (arcsec)	0.042	0.028	0.0227	0.0183	0.0122	-	-	-
	θ_{MRS} (arcsec)	0.496	0.331	0.268	0.216	0.144	-	-	-