ALMA simulators for Cycle 2

Eelco van Kampen, ESO

The use of simulators

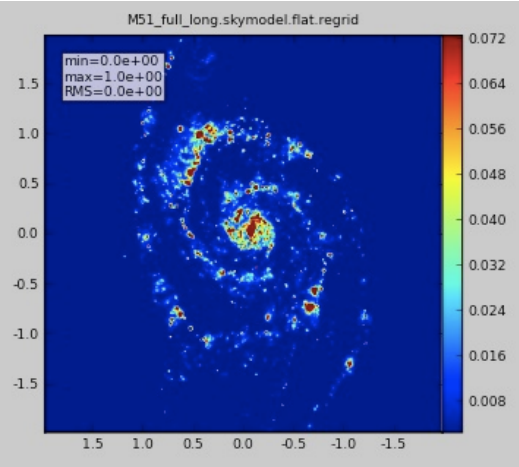
Proposers can use simulations to test their science case:
- can I do this with ALMA ? Do I need the ACA, or TP ?
- can I do this during Cycle 2 ? Should I ?
- how to optimize my science goal ?

PIs can use simulations to interpret their observed data

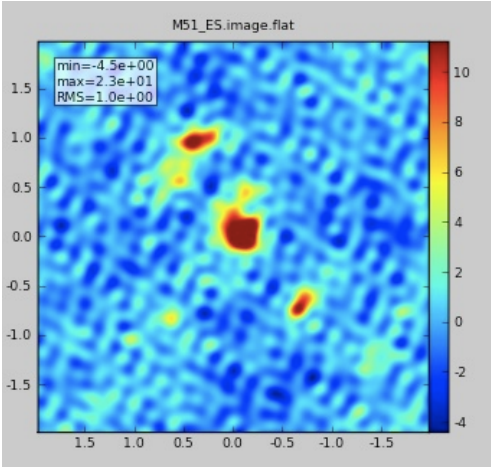
The simulation tools are also used by **the ALMA project** itself

Nearby galaxy (ALMA band 9)

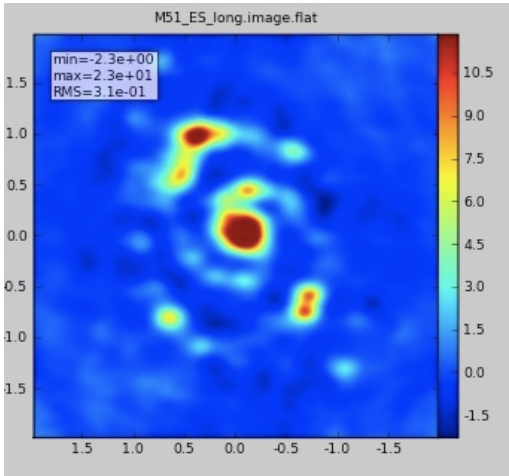
An H-alpha image of M51
provided by D. Thilker (NRAO)



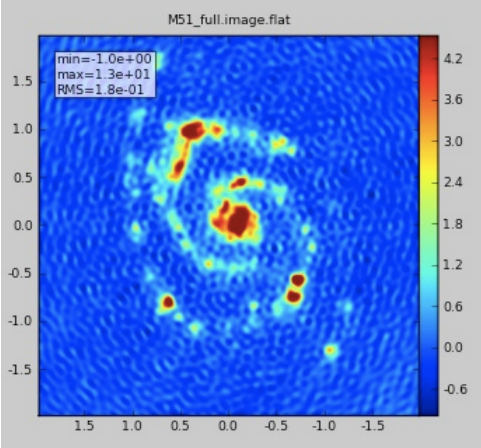
Skymodel



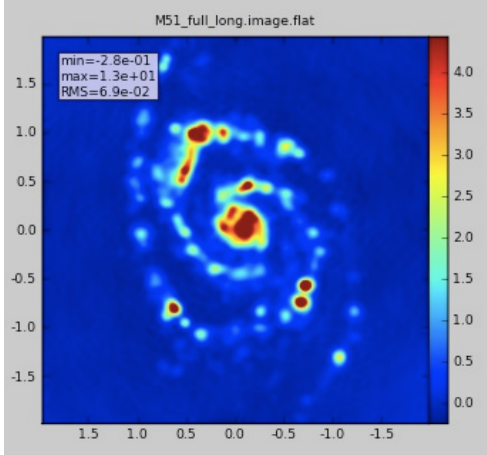
ES (30 min)



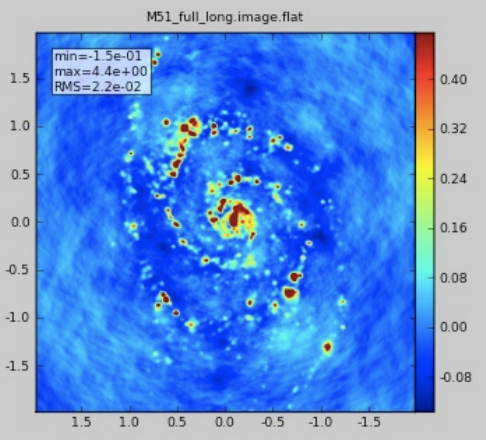
ES (4h)



Full 2 km array (30 min)



Full 2 km array (4h)



Full 6 km array (4h)

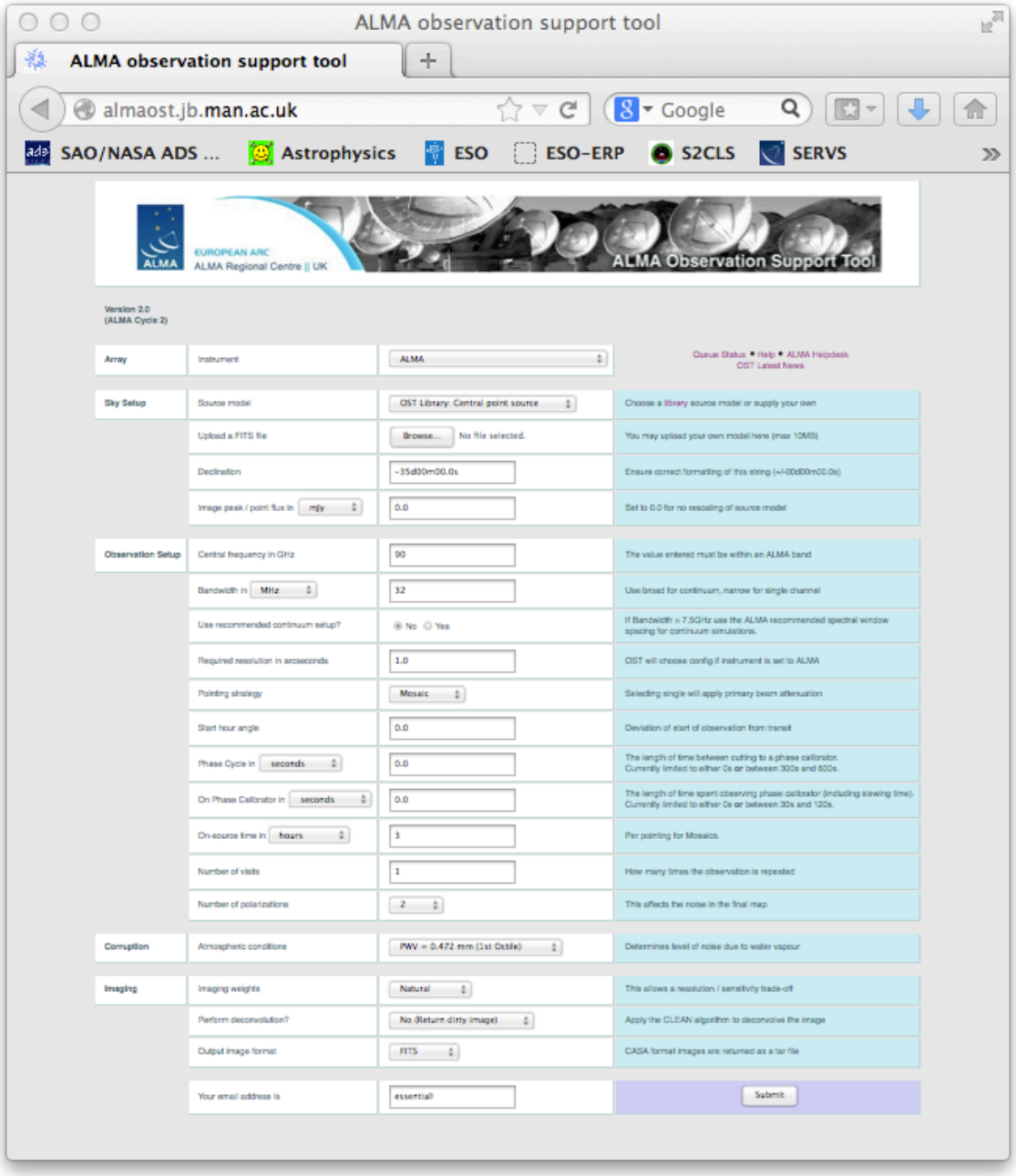
Cycle 2 ALMA simulators

Novice users: OST (hosted at the UK ARC: http://almaost.jb.man.ac.uk)
Expert users: simobserve/simanalyze/simalma (part of CASA)

Simobserve/simanalyze/simalma are CASA ***tasks*** used to produce mock ALMA data from an input sky model (theoretical model or previous observation).
The main work is done by the sm ***tool***: the three tasks (Python scripts) act as a user-friendly interface to this tool with additional work done on plotting and analysis

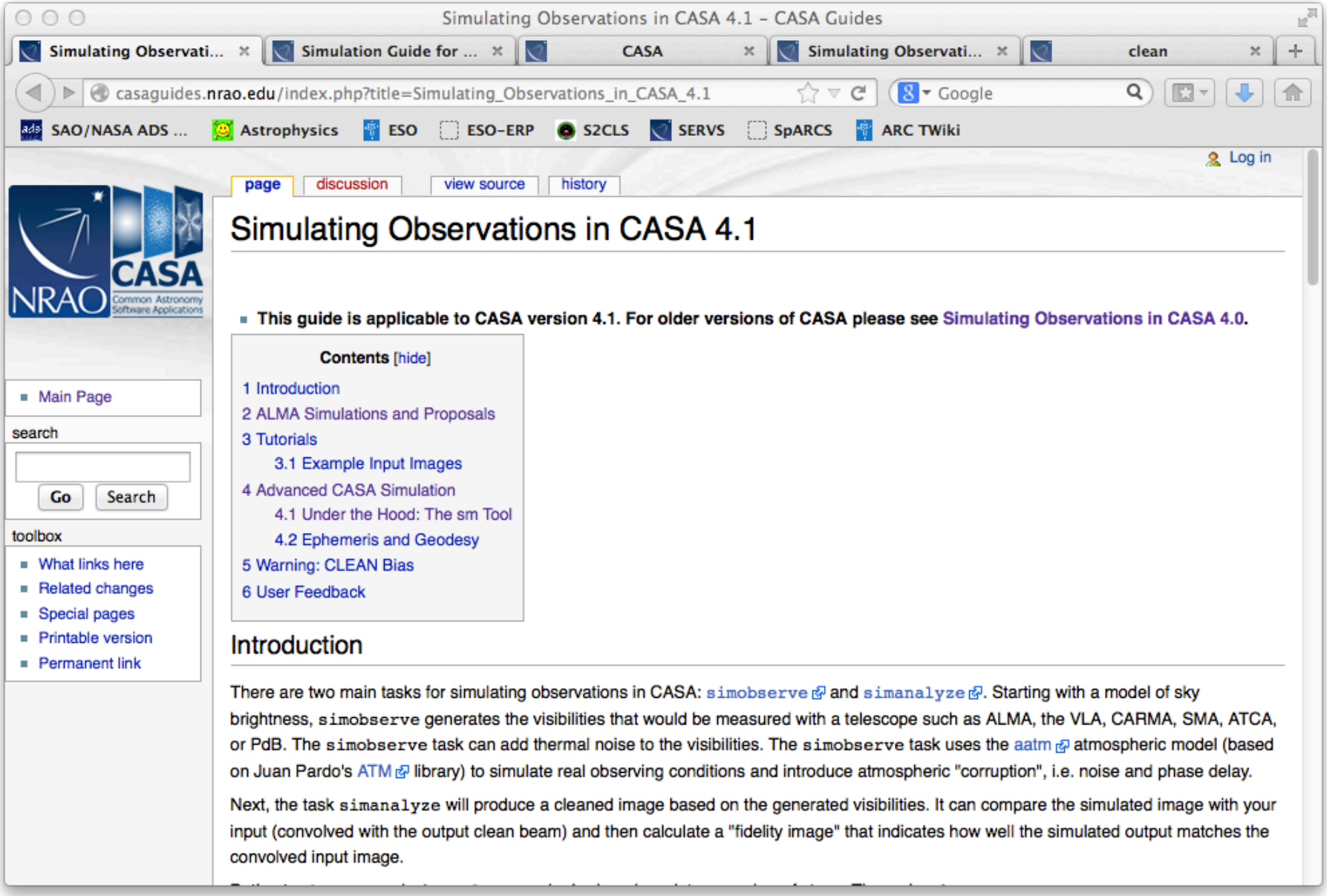
The OST (Observation Support Tool) is a webtool that also uses the sm ***tool*** under the hood, but is much simplified (hence faster) and restricted, with a website acting as a simple GUI to set parameters and run the simulation

The OST (Observation Support Tool)



almaost.jb.man.ac.uk

on-line ‘casaguides’ to *simobserve*



Note for proposers

Values from simobserve or the OST should not be used to calculate exposure times for ALMA Science Goals. Only values from the ALMA sensitivity calculator should be used for this purpose, as it is the ALMA sensitivity calculator that will be used in the technical assessment of ALMA proposals.

Representative Cycle 2 antenna configurations are included in CASA 4.2 and the OST (and can be downloaded from the Science Portal), but the actual configuration could differ somewhat when the observation is scheduled.

General comments

There are two types of input models:
- a theoretical model (a science simulation)
- a previous observation, in the same waveband or a related one

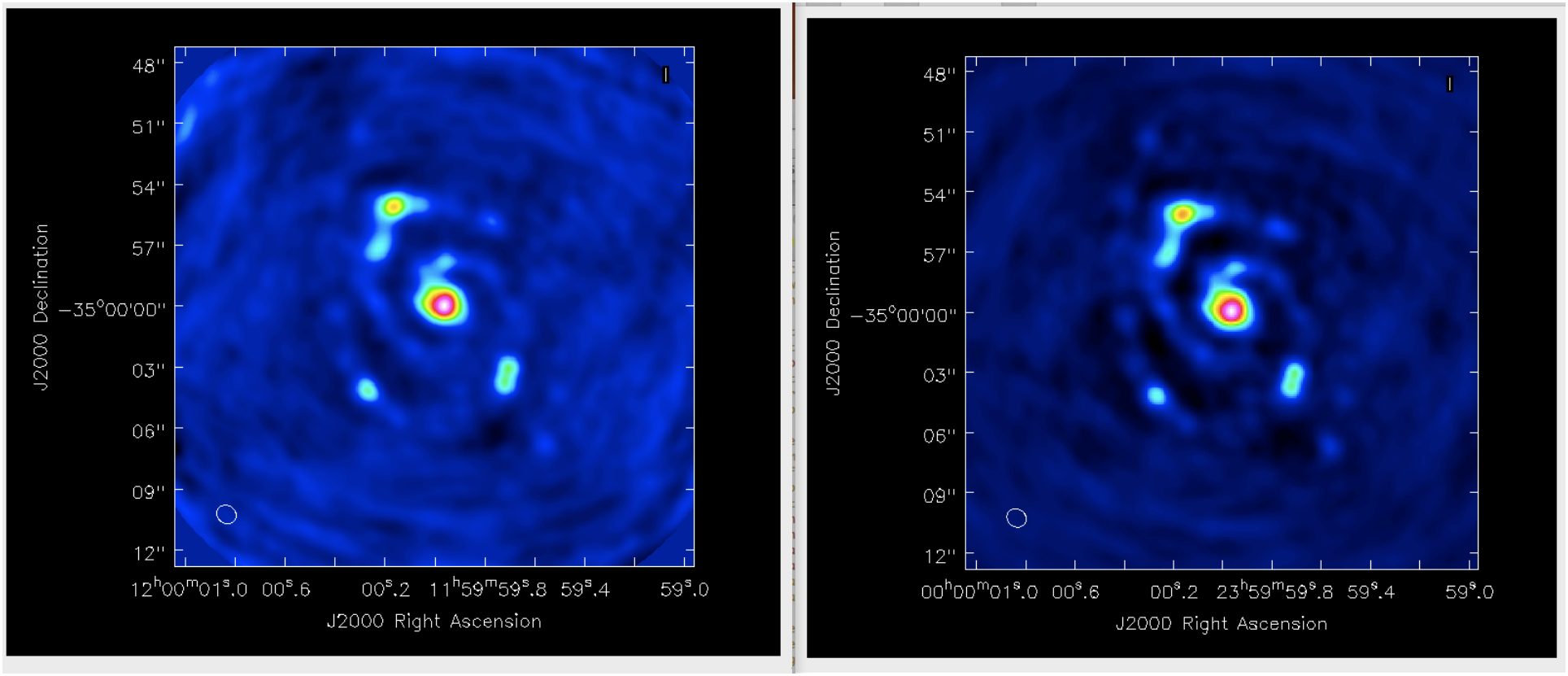
In the first case the input image is noise free, in the second case it is not. Therefore the noise estimated by *simobserve* or the OST comes on top of the original noise, and could result in overestimation.

Both *simobserve* and the OST are based on the sm tools in CASA, but do not include the same noise terms, and have different default clean parameters.

Warning: CLEAN Bias
As is the case for real images, cleaning images produced by simobserve can lead to a spurious decrease in object fluxes and noise on the image (an effect known as "clean bias"). This is particularly true for observations with poor coverage of the uv-plane, i.e. using telescopes with small numbers of antennas, such as the ALMA Early Science configurations, and/or in short "snapshot" observations. Users should always clean images with care, using a small number of iterations and/or a conservative (3-5sigma) threshold and boxing bright sources.

simobserve versus OST: images

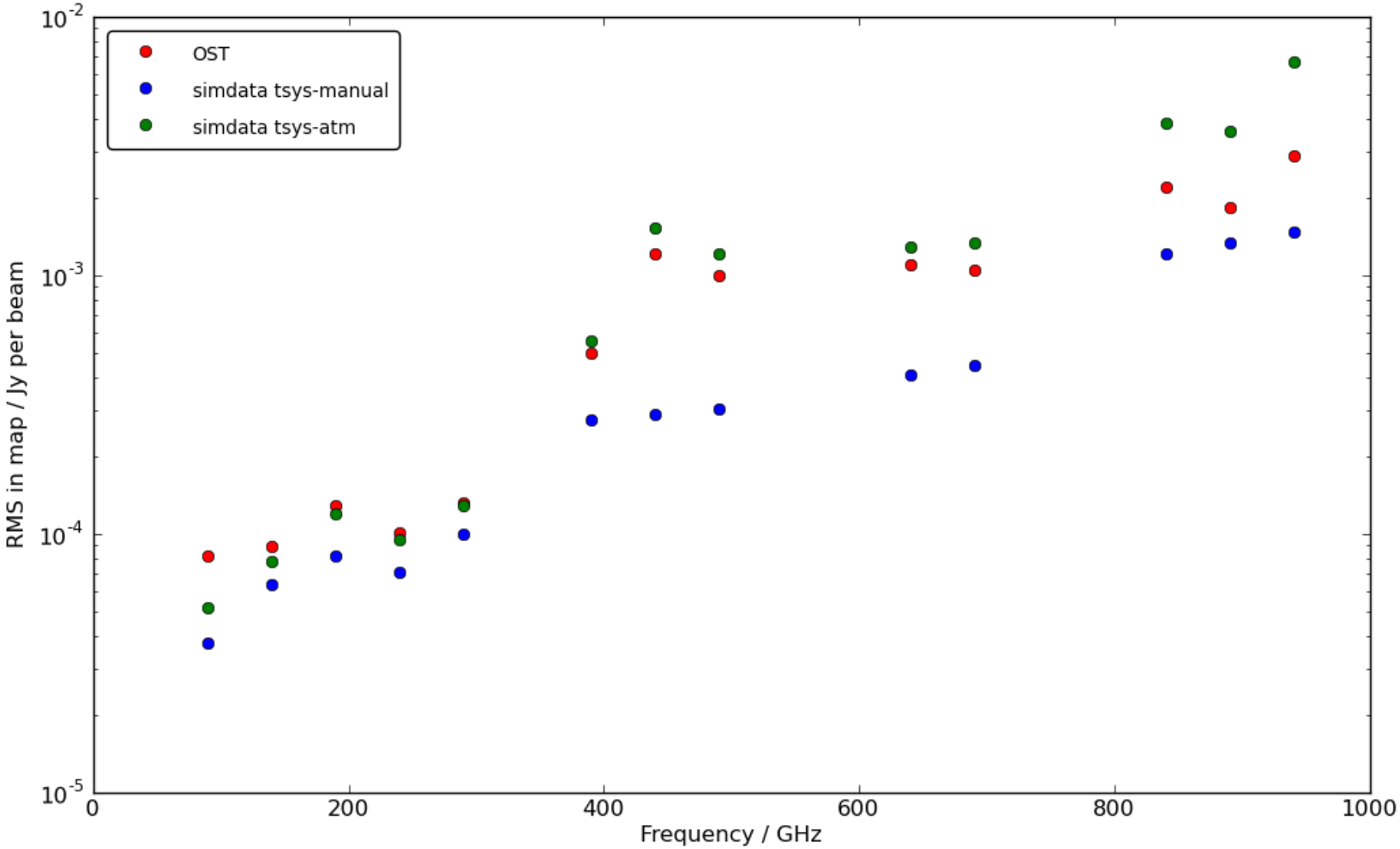
simobserve OST



M51@z=0.5, Early Science configuration, band 6

simobserve versus OST: noise levels

rms noise comparison for all ALMA bands (using pure noise maps)



ALMA+ACA+TP

Deconvolving the predicted visibilities back into an image

Many ways of doing this, including:

1) use the total power image as a model when deconvolving the ACA image, and then use the result as a model when deconvolving the 12m interferometric image (this tends to give low weight to the large spatial scales)

2) use multiscale clean, in the clean task (again using the lower resolution image as a model when deconvolving the higher resolution one)

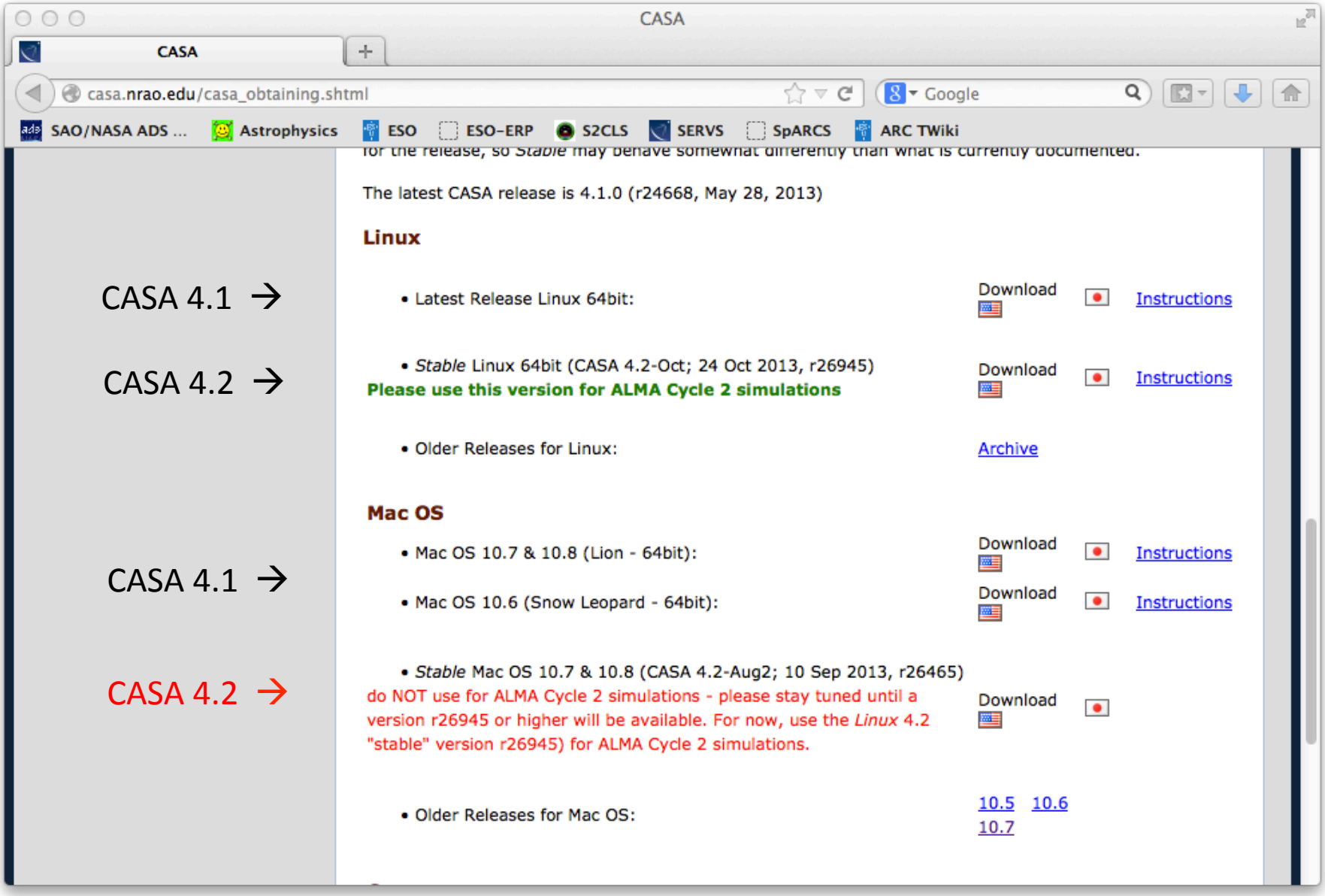
3) create each image independently, and then use the CASA feather task to combine them entirely in the image plane

Note: use different dates for ALMA, ACA and TP simulated visibilities, as generic antenna names are used by *simobserve* … (dataset concatenation !).

New ***simalma*** task in CASA to help you do this in one go … but see next slide !

CASA 4.1 or 4.2 ?

From the CASA download page:



Default *simobserve* parameters

```
CASA <3>: inp
--------> inp()
#  simobserve :: mosaic simulation task
project             =      'sim'        #  root prefix for output file names
skymodel            =        ''         #  model image to observe
complist            =        ''         #  componentlist to observe
setpointings        =      True
     integration    =      '10s'        #  integration (sampling) time
     direction      =        ''         #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize        =   ['', '']        #  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 Nyquist

obsmode             =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist    = 'alma.out10.cfg'  #  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, or "transit"
     totaltime      =    '7200s'        #  total time of observation or number of repetitions
     caldirection   =        ''         #  pt source calibrator [experimental]
     calflux        =      '1Jy'

thermalnoise        =        ''         #  add thermal noise: [tsys-atm|tsys-manual|""]
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
async               =     False        #  If true the taskname must be started using simobserve(...)
```

Setting *simobserve* parameters

skymodel=<filename>
thermalnoise='tsys-atm'

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

complist            =          ''       #  componentlist to observe
setpointings        =       True
     integration    =       '10s'       #  integration (sampling) time
     direction      =          ''       #  "J2000 19h00m00 -40d00m00" or "" to center on model
     mapsize        =    ['', '']       #  angular size of map or "" to cover model
     maptype        =      'ALMA'       #  hexagonal, square, etc
     pointingspacing =          ''      #  spacing in between pointings or "0.25PB" or "" for 0.5 PB

obsmode             =      'int'        #  observation mode to simulate [int(interferometer)|sd(singledish)|""(none)]
     antennalist    = 'alma.out10.cfg'  #  interferometer antenna position file
     refdate        = '2012/05/21'      #  date of observation - not critical unless concatting simulations
     hourangle      =  'transit'        #  hour angle of observation center e.g. -3:00:00, or "transit"
     totaltime      =    '7200s'        #  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       =       1.0         #  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
async               =     False         #  If true the taskname must be started using simobserve(...)
```

Default *simanalyze* parameters

```
#  simanalyze :: image and analyze simulated datasets
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     =         ''        #  prior image to use in clean e.g. existing single dish 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 or "" to equal model
     niter          =        500        #  maximum number of iterations (0 for dirty image)
     threshold      =   '0.1mJy'        #  flux level (+units) to stop cleaning
     weighting      =  'natural'        #  weighting to apply to visibilities
     mask           =         []        #  Clearbox(es), mask image(s), region(s), or a level
     outertaper     =         []        #  uv-taper on outer baselines in uv-plane
     stokes         =        'I'        #  Stokes params to image

analyze             =      False        #  (only first 6 selected outputs will be displayed)
graphics            =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose             =      False
overwrite           =       True        #  overwrite files starting with $project
async               =      False        #  If true the taskname must be started using simanalyze(...)
```

Setting *simanalyze* parameters

analyze=True

```
#  simanalyze :: image and analyze simulated datasets
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     =         ''        #  prior image to use in clean e.g. existing single dish 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 or "" to equal model
     niter          =        500        #  maximum number of iterations (0 for dirty image)
     threshold      =   '0.1mJy'        #  flux level (+units) to stop cleaning
     weighting      =  'natural'        #  weighting to apply to visibilities
     mask           =         []        #  Cleanbox(es), mask image(s), region(s), or a level
     outertaper     =         []        #  uv-taper on outer baselines in uv-plane
     stokes         =        'I'        #  Stokes params to image

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 beam
     showclean      =       True        #  display the synthesized image
     showresidual   =      False        #  display the clean residual image (ignored in single dish simulation)
     showdifference =       True        #  display difference image
     showfidelity   =       True        #  display fidelity

graphics            =     'both'        #  display graphics at each stage to [screen|file|both|none]
verbose             =      False
overwrite           =       True        #  overwrite files starting with $project
async               =      False        #  If true the taskname must be started using simanalyze(...)
```

Antenna configurations

Antenna configurations are simple text files, listing all antennas that are part of the array of choice. Many come with CASA, and can be found in the CASA repository directory in the subdirectory data/alma/simmos

For example, to select ALMA full science configuration #20, use (in simobserve)

```
> antennalist='alma.out20.cfg'
```

One can also choose a configuration corresponding to a certain resolution, eg:

```
> antennalist = "alma;0.05arcsec"
```

NOTE: the 'full operations' antenna configurations are meant to be representative. The Cycle 2 antenna configuration files are all included in CASA 4.2, but need to be requested separately for CASA 4.1 (can be downloaded from the Science Portal !).

Example antenna configuration for full operations (included with CASA)

alma.out20.cfg

```
# observatory=ALMA
# coordsys=UTM
# datum=SAM56
# zone=19
# hemisphere=S
#UTM-X       UTM_Y       Z     Diam (m)   Pad #
627801.31  7453100.27  5029.4  12.0       3
627762.59  7453069.82  5029.9  12.0       23
627808.00  7453045.89  5028.3  12.0       43
628103.00  7453218.00  5022.2  12.0       102
627454.00  7453191.00  5023.4  12.0       103
627980.00  7452724.00  5029.8  12.0       104
627856.00  7453486.00  5026.6  12.0       105
627499.00  7452791.00  5023.5  12.0       106
628250.00  7453047.00  5015.9  12.0       107
627422.00  7453453.00  5029.0  12.0       108
627837.00  7452578.00  5032.9  12.0       109
628059.00  7453493.00  5022.3  12.0       110
627320.00  7452981.00  5025.1  12.0       111
628242.00  7452816.00  5015.9  12.0       112
627593.00  7453611.00  5031.0  12.0       113
627615.00  7452488.00  5028.0  12.0       114
628287.00  7453384.00  5016.4  12.0       115
627237.00  7453285.00  5026.9  12.0       116
628261.00  7452578.00  5019.8  12.0       117
627878.00  7453858.00  5029.4  12.0       118
627369.00  7452511.00  5019.2  12.0       119
628488.00  7453134.00  5007.2  12.0       120
627265.00  7453482.00  5026.1  12.0       121
628003.00  7452241.00  5018.9  12.0       122
628166.00  7453836.00  5022.7  12.0       123
627021.00  7452792.00  5011.4  12.0       124
628593.00  7452742.00  5012.6  12.0       125
627364.00  7453932.00  5025.9  12.0       126
627640.00  7452147.00  5028.4  12.0       127
628567.00  7453703.00  5010.7  12.0       128
626779.00  7453196.00  5013.0  12.0       129
628571.00  7452164.00  5030.1  12.0       130
627725.00  7454268.00  5029.5  12.0       131
627047.00  7452073.00  5015.2  12.0       132
628948.00  7453327.00  4984.2  12.0       133
626693.00  7453811.00  5011.8  12.0       134
```

All antenna configurations included in CASA 4.1

all files in CASA's subdirectory /data/alma/simmos :

WSRT.cfg
aca.i.cfg
aca.ns.cfg
aca.tp.cfg
aca_cycle1.cfg
alma.cycle0.compact.cfg
alma.cycle0.extended.cfg
alma.out01.cfg
alma.out02.cfg
alma.out03.cfg
alma.out04.cfg
alma.out05.cfg
alma.out06.cfg
alma.out07.cfg
alma.out08.cfg
alma.out09.cfg
alma.out10.cfg
alma.out11.cfg
alma.out12.cfg
alma.out13.cfg
alma.out14.cfg
alma.out15.cfg
alma.out16.cfg
alma.out17.cfg
alma.out18.cfg
alma.out19.cfg
alma.out20.cfg
alma.out21.cfg
alma.out22.cfg
alma.out23.cfg
alma.out24.cfg
alma.out25.cfg
alma.out26.cfg
alma.out27.cfg
alma.out28.cfg
alma_cycle1_1.cfg
alma_cycle1_2.cfg
alma_cycle1_3.cfg
alma_cycle1_4.cfg
alma_cycle1_5.cfg
alma_cycle1_6.cfg
carma.a.cfg
carma.b.cfg
carma.c.cfg
carma.d.cfg
carma.e.cfg
meerkat.cfg
pdbi-a.cfg
pdbi-b.cfg
pdbi-c.cfg
pdbi-d.cfg
sma.compact.cfg
sma.compact.n.cfg
sma.extended.cfg
sma.subcompact.cfg
sma.vextended.cfg
vla.a.cfg
vla.b.cfg
vla.bna.cfg
vla.c.cfg
vla.cnb.cfg
vla.d.cfg
vla.dnc.cfg

Output filenames

```
------------------------------
Output produced: (not all will always exist, depending on input parameters)
To support different runs with different arrays, the names have the
configuration name from antennalist appended.
------------------------------

project.[cfg].skymodel = 4d input sky model image (optionally) scaled
project.[cfg].skymodel.flat.regrid.conv = input sky regridded to match the
    output image, and convolved with the output synthesized beam
project.[cfg].skymodel.png = diagnostic figure of sky model with pointings

project.[cfg].ptg.txt = list of mosaic pointings
project.[cfg].quick.psf = psf calculated from uv coverage
project.[cfg].ms = noise-free measurement set
project.[cfg].noisy.ms = corrupted measurement set
project.[cfg].observe.png = diagnostic figure of uv coverage and
     visibilities

project.[cfg].image = synthesized image
project.[cfg].flux.pbcoverage = promary beam correction for mosaic image
project.[cfg].residual = residual image after cleaning
project.[cfg].clean.last = parameter file of what parameters were used in
      the clean task
project.[cfg].psf = synthesized beam calculated from weighted uv
      distribution
project.[cfg].image.png = diagnostic figure of clean image and residual

project.[cfg].fidelity = fidelity image
project.[cfg].analysis.png = diagnostic figure of difference and fidelity

project.[cfg].simdata.last = saved input parameters for simdata task
```

Running simobserve interactively

1: start up CASA: casapy

2: default("simobserve")

3: inp()

4: manually set the various parameters

5: go() or simobserve()

repeat 4+5

This works the same way for *simanalyze* (or any other CASA task)

Running simobserve using a script

1: open yourscript.py in your favorite editor

2: start up CASA: casapy

3: execfile("yourscript.py")

edit yourscript.py and repeat 3

Things you (could) need

Tutorial material, to be found at ftp://ftp.eso.org/pub/general/simdata/ :
- input models (skymodels): FITS files (or use your own !)
- scripts / parameter settings: python files
- Cycle 2 antenna configuration files (only for CASA 4.1 - they are included in CASA 4.2)

Downloading 'ALL.tar' or 'ALL.zip' gets you all of these in one go.

OST: http://almaost.jb.man.ac.uk/

CASA (includes *simobserve*, *simanalyze, simalma*): http://casa.nrao.edu/

Simulation CASAguides starting point:
http://casaguides.nrao.edu/index.php?title=Simulating_Observations_in_CASA_4.1

Simulating datacubes (advanced users):
http://casaguides.nrao.edu/index.php?title=N891_simdata_(CASA_3.4)

Useful tools:
- FITS viewer/editor: for example Fv (http://heasarc.nasa.gov/lheasoft/ftools/fv/)
- CosmoCalc: http://www.astro.ucla.edu/~wright/CosmoCalc.html

Things we need

Feedback on the simulators !

- bug reports (especially *almasim*)

- suggestions for improvement (always possible)

- interesting input models (for a casaguide)

Send feedback to evkampen@eso.org