Simulating ALMA Observations

How to get started and what to expect
Simulating ALMA data

Bjorn Emonts
NRAO (CASA User Liaison)
casa-feedback@nrao.edu
bemonts@nrao.edu

Andrew McNichols
NRAO (CASA Simulations, next-generation CASA)
casa-feedback@nrao.edu
amcnicho@nrao.edu

Credit:
Remy Indebetouw (NRAO)
Why simulate ALMA observations?

“Running a simulation can help convince the TAC that your proposed observations are feasible”
Why simulate ALMA observations?

“Running a simulation can help convince the TAC that your proposed observations are feasible”
Why simulate ALMA observations?

Proposed resolution / array configuration
Why simulate ALMA observations?

Extended emission
Resolved out on long baselines

$\lambda$!!
Why simulate ALMA observations?

Proposed resolution / array configuration

35 GHz

Point-source + Gaussian

Emonts et al 2018, ASPC, 517, 587
Why simulate ALMA observations?

Proposed resolution / array configuration

Emonts et al 2018, ASPC, 517, 587

Point-source + Gaussian

35 GHz

(u,v) model

Mm signal very easily resolved out!

ALMA Band 4 (150 GHz)

1km baseline: < 0.5 arcsec

CO(4-3) at z=2: < 4 kpc

Emonts et al 2018, ASPC, 517, 587

ALMA Band 4  (150 GHz)

1km baseline: < 0.5 arcsec

CO(4-3) at z=2: < 4 kpc
**How to simulate ALMA observations?**

**Option 1: CASA**

https://casa.nrao.edu

**Option 2: ALMA Observations Support Tool**

<table>
<thead>
<tr>
<th>OST</th>
<th>NEWS</th>
<th>HELP</th>
<th>QUEUE</th>
<th>LIBRARY</th>
<th>ALMA HELPDESK</th>
</tr>
</thead>
<tbody>
<tr>
<td>OST User Notice: Version 7.0 release - 19/03/2019 !!! (more info). OST Team</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Array Setup:**
- **Instrument:** ALMA
  - Select the desired ALMA antenna configuration.

**Sky Setup:**
- **Source model:** OST Library: Central point source
  - Choose a library source model or supply your own.
  - You may upload your own model here (max 10MB).
  - Ensure correct formatting of this string (+/-00d00m00.0s).
  - Rescale the image data with respect to new peak value.
  - Set to 0.0 for no rescaling of source model.

**Observation Setup:**
- **Observing mode:** Spectral
  - Spectral or continuum observations?
  - The value entered must be within an ALMA band.
  - Select the total bandwidth for continuum observations.
CASA

Latest release: CASA 6.1

Official ALMA version: **CASA 5.6.1**
*(CASA 6.1.1 with ALMA-pipeline pending)*
Download CASA versions

https://casa.nrao.edu

Latest version: CASA 5.7/6.1

The transition from CASA 5 to CASA 6 is based on the switch from Python 2 to Python 3. To ease in this transition, CASA is releasing version 6.1 (Python 3) simultaneously with CASA 5.7 (Python 2). The task and tools functionality of CASA 6.1 and 5.7 are scientifically equivalent.
## CASA

### Download CASA versions

[https://casa.nrao.edu](https://casa.nrao.edu)

---

### Alert!

Manual processing can be done with any of the versions below, but the ALMA and VLA pipelines may differ and are not always included in the official CASA releases. Please ensure to download the correct CASA version for running a pipeline.

<table>
<thead>
<tr>
<th>CASA 5.7 / 6.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>RedHat 7/6 (6.1.2)</td>
</tr>
</tbody>
</table>

(versions 6.1.2-7 and 5.7.2-4. CASA 6.1.2 includes a pipeline validated for VLA, CASA 5.7.2 does not include any pipeline)

### ALMA pipeline with CASA 5.6

| RedHat 7 | RedHat 6 |

(Version CASA 5.6.1-8, with ALMA pipeline version number 42866)

### VLA pipeline with CASA 6.1

| RedHat 7/6 |

(Version CASA 6.1.2-7, with VLA pipeline version number 2020.1.0.36)

The above CASA versions can also be downloaded from our [NAOJ CASA mirror site](https://naoj-casa-mirror.nao.ac.jp) and [NAOJ CASA-pipeline mirror site](https://naoj-casa-pipeline-mirror.nao.ac.jp), or via [Google Drive](https://drive.google.com).
CASA

Latest release: CASA 6.1

• Monolithic tar-file (pipelines/manual)
  Plug-and-play (like always)

• Modular pip-wheels (manual)
  Integration Python (Jupyter Notebooks)

  CASA 6.1: tools & tasks (CASA 6.2: GUIs)
CASA

Latest release: CASA 6.1

ALMA Cycle 8: CASA 6.1.1

(release pending)

- simulator tclean
  (CASA 5.6 and earlier old clean)

- simulator Cycle-8 array-configuration files

Do **not** use CASA 5.3 for simulations!

(*bug tool `cl.addcomponents / ia.modify`*)
How to simulate ALMA observations?

1. CASA simulation tasks:
   - simobserve: *create MS*
   - simanalyze: *image MS*
How to simulate ALMA observations?

1. CASA simulation tasks:
   - simobserve: create MS
   - simanalyze: image MS

2. Simulator tools:
   sm tool / simutil
How to simulate ALMA observations?

1. CASA simulation tasks:
   - simobserve: *create MS*
   - simanalyze: *image MS*

2. Simulator tools:
   - sm tool / simutil

3. Configuration files:

   6.1.1: ALMA Cycle 0 – 8 + ACA
   VLA, ngVLA, ATCA, PdBI, WSRT, CARMA, MeerKAT, SMA, VLBA

   (Config files: ALMA Cycle 8 = Cycle 7)
How to simulate ALMA observations?

1. CASA simulation tasks:
   - simobserve: *create MS*
   - simanalyze: *image MS*

2. Simulator tools:
   - sm tool / simutil

3. Configuration files:
   - ALMA Cycle 0 – 8 + ACA
   - VLA, ngVLA, ATCA, PdbI, WSRT, CARMA, MeerKAT, SMA, VLBA

4. Visualization images:
   - CASA Viewer / CARTA
How to simulate ALMA observations?

Cube Analysis and Rendering Tool for Astronomy

4. Visualization images:
CASA Viewer / CARTA

CARTA version 1.4
Not all features of Viewer, but rapid progress!

Replace CASA Viewer in near future

https://cartavis.github.io/

Consortium:
ASIAA, IDIA, NRAO, Univ. Alberta
More information on ALMA simulations

1. **CASA Docs**: Official CASA documentation
   https://casa.nrao.edu/casadocs/
More information on ALMA simulations

1. **CASA Docs**: Official CASA documentation
   https://casa.nrao.edu/casadocs/

---

**CASA**, the *Common Astronomy Software Applications*, is the primary data processing software for the Atacama Large Millimeter/submillimeter Array (ALMA) and Karl G. Jansky Very Large Array (VLA), and is often used also for other radio telescopes.

CASA 5.7/6.1 can now be **downloaded** for general use. CASA 5.7 is based on Python 2, while CASA 6.1 is based on Python 3, but both contain the same functionality. CASA 6.1 is available either as a downloadable tar-file (much like CASA 5.7), or through **pip-wheel installation**, which gives flexibility to integrate CASA into a customized Python environment.

The CASA 5.7/6.1 release builds on CASA 5.6, but has the following main new features:

**New Features**

- A new task `sdintimaging` is available for joint deconvolution of Single Dish and Interferometer data.
- A new task `sdtimeaverage` is available for averaging single-dish spectral data over specified time.
- A new single-dish spectral imaging mode, `cubesource`, is available in the task `tsdimaging`.
- A new parameter `cordeflags` has been added to the `gaincal`, `bandpass`, and `fringefit` tasks to permit more control of flagging subsets of correlations.
- The `fringefit` task now includes support for dispersive delays.
- The CASA simulator now uses `tclean` instead of `clean`.
- The ability to correct for heterogeneous antenna pointing offsets stored in the POINTING sub-table of the MS has been added to `tclean` (`gridder=awproject`).
More information on ALMA simulations

1. **CASA Docs**: Official CASA documentation

   ![CASA Docs](https://casadocs.readthedocs.io/en/latest/)

   **Common Astronomy Software Applications**

   CASA, the Common Astronomy Software Applications, is the primary data processing software for the Atacama Large Millimeter/submillimeter Array (ALMA) and Karl G. Jansky Very Large Array (VLA), and is often used also for other radio telescopes.

   **6.2.0 Development Build**

   You are viewing the latest build of master, currently in the 6.2.0 development cycle.

   CASA is developed by an international consortium of scientists based at the National Radio Astronomical Observatory (NRAO), the European Southern Observatory (ESO), the National Astronomical Observatory of Japan (NAOJ), the Academia Sinica Institute of Astronomy and Astrophysics (ASIAA), CSIRO Astronomy and Space Science (CSIRO/CASS), and the Netherlands Institute for Radio Astronomy (ASTRON), under the guidance of NRAO.

   **Upcoming: CASA 6.2!**
More information on ALMA simulations

1. **CASA Docs**: Official CASA documentation
   [https://casa.nrao.edu/casadocs/](https://casa.nrao.edu/casadocs/)

2. **CASA Guides**: Telescope-specific CASA strategies
   [https://casaguides.nrao.edu/](https://casaguides.nrao.edu/)

**CASA Tutorials**

<table>
<thead>
<tr>
<th>ALMA</th>
<th>VLA</th>
<th>VLBI</th>
<th>ATCA</th>
<th>Simulations</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1.png" alt="ALMA" /></td>
<td><img src="image2.png" alt="VLA" /></td>
<td><img src="image3.png" alt="VLBI" /></td>
<td><img src="image4.png" alt="ATCA" /></td>
<td><img src="image5.png" alt="Simulations" /></td>
</tr>
</tbody>
</table>
More information on ALMA simulations

<table>
<thead>
<tr>
<th>Simulating ngVLA Data (CASA 5.4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Simalma (CASA 5.4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>ACA Simulation (CASA 5.4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>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-meter array in combination with the ACA, and those interested in combining data from multiple 12-m array configurations.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Simulation Guide Component Lists (CASA 5.4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Protoplanetary Disk Simulation (CASA 5.4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Protoplanetary Disk Simulation - VLA (CASA 5.5)</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Advanced: Corrupting Simulated Data (Simulator Tool)</th>
</tr>
</thead>
<tbody>
<tr>
<td>simobserve tool 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 tool' 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).</td>
</tr>
</tbody>
</table>
More information on ALMA simulations: Tutorials

### 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 5.5)
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](https://casa.nrao.edu/casadocs/). An examples of advanced techniques for corrupting a simulated MeasurementSet can be found in this CASA Guide on [Corrupting Simulated Data (Simulator Tool)](https://casaguides.nrao.edu/).
# Model sky = Halpha image of M51
os.system('curl https://casaguides.nrao.edu/images/3/3f/M51ha.fits.txt -f -o M51ha.fits')
skymodel = "M51ha.fits"

# Set model image parameters:
indirection="J2000 23h59m59.96s -34d59m59.50s"
inell="0.1arcsec"
inbright="0.004"
incenter="330.076GHz"
inwidth="50MHz"

antennalist=["alma.cycle6.3.cfg","aca.cycle6.cfg"]

totaltime="1800s"
tpnant = 2
tptime="7200s"
pwv=0.6
mapsize="1arcmin"

inp
go
# Model sky = Halpha image of M51
os.system('curl https://casaguides.nrao.edu/images/3/3f/M51ha.fits.txt -f -o M51ha.fits')

skymodel = "M51ha.fits"

# Set model image parameters:
indirection="J2000 23h59m59.96s -34d59m59.50s"
incell="0.1arcsec"
inbright="0.004"
incenter="330.076GHz"
inwidth="50MHz"

antennalist = ["alma.cycle6.3.cfg","aca.cycle6.cfg"]

totaltime="1800s"
tonant = 2
tptime="7200s"
pwv=0.6

mapsize="1arcmin"

inp
go
## SIMALMA: Antenna Position Lists

### # Model sky = Halpha image of M51

<table>
<thead>
<tr>
<th>Start date</th>
<th>Configuration</th>
<th>Longest baseline</th>
<th>LST for best observing conditions</th>
</tr>
</thead>
<tbody>
<tr>
<td>2021 October 1</td>
<td>C-8</td>
<td>8.5 km</td>
<td>~ 22–10 h</td>
</tr>
<tr>
<td>2021 October 20</td>
<td>C-7</td>
<td>3.6 km</td>
<td>~ 23–11 h</td>
</tr>
<tr>
<td>2021 November 10</td>
<td>C-6</td>
<td>2.5 km</td>
<td>~ 1–13 h</td>
</tr>
<tr>
<td>2021 December 1</td>
<td>C-5</td>
<td>1.4 km</td>
<td>~ 2–14 h</td>
</tr>
<tr>
<td>2021 December 20</td>
<td>C-4</td>
<td>0.78 km</td>
<td>~ 4–15 h</td>
</tr>
<tr>
<td>2022 January 10</td>
<td>C-3</td>
<td>0.50 km</td>
<td>~ 5–17 h</td>
</tr>
<tr>
<td>2022 February 1</td>
<td>No observations due to maintenance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2022 March 1</td>
<td>C-1</td>
<td>0.16 km</td>
<td>~ 9–21 h</td>
</tr>
<tr>
<td>2022 March 26</td>
<td>C-2</td>
<td>0.31 km</td>
<td>~ 9–23 h</td>
</tr>
<tr>
<td>2022 April 20</td>
<td>C-3</td>
<td>0.50 km</td>
<td>~ 11–0 h</td>
</tr>
<tr>
<td>2022 May 10</td>
<td>C-4</td>
<td>0.78 km</td>
<td>~ 12–2 h</td>
</tr>
<tr>
<td>2022 May 31</td>
<td>C-5</td>
<td>1.4 km</td>
<td>~ 13–4 h</td>
</tr>
<tr>
<td>2022 June 23</td>
<td>C-6</td>
<td>2.5 km</td>
<td>~ 15–6 h</td>
</tr>
<tr>
<td>2022 July 28</td>
<td>C-5</td>
<td>1.4 km</td>
<td>~ 17–7 h</td>
</tr>
<tr>
<td>2022 August 18</td>
<td>C-4</td>
<td>0.78 km</td>
<td>~ 19–8 h</td>
</tr>
<tr>
<td>2022 September 10</td>
<td>C-3</td>
<td>0.50 km</td>
<td>~ 20–9 h</td>
</tr>
</tbody>
</table>

### CASA Simulator

The CASA Simulator allows a user to simulate interferometric observations, including the ALMA observatory. The simulations consider the configuration of the ALMA array at the start date and allow for the generation of data products that can be used for calibration and scientific analysis. The CASA Simulator is part of the ALMA Science Planning System and can be accessed through the ALMA website.

**CASA Guides:**
- [https://casaguides.nrao.edu/](https://casaguides.nrao.edu/)
- [https://almascience.nrao.edu/tools/casa-casa-simulator](https://almascience.nrao.edu/tools/casa-simulator)
# Model sky = Halpha image of M51
os.system('curl https://casaguides.nrao.edu/image
skymodel = "M51ha.fits"

# Set model image parameters:
indirection="J2000 23h59m59.96s -34d59m59.50s"
incell="0.1arcsec"
inbright="0.004"
incenter="330.076GHz"
inwidth="50MHz"

antennalist=["alma.cycles6.3.cfg","aca.cycles6.cfg"

totaltime="1800s"
tpnant = 2
tptime = "7200s"
pwv=0.6
mapsize="larcmin"

inp
go
1. Simobserve

Simulate visibilities (MS) for each configuration

2. Simanalyze

Imaging using simulated MSs
1. Simobserve
Simulate visibilities (MS) for each configuration

2. Simanalyze
Imaging using Simulated MSs
### 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 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 5.5)

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).
Simulating Component Lists

```python
# In CASA
direction = "J2000 10h00m00.0s -30d00m00.0s"
cl.done()
cl.addcomponent(dir=direction, flux=1.0, fluxunit='Jy', freq='230.0GHz', shape="Gaussian",
               majoraxis='0.1arcmin', minoraxis='0.05arcmin', positionangle='45.0deg')
#
ia.fromshape("Gaussian.im",[256,256,1,1],overwrite=True)
cs=ia.coordsys()
cs.setunits([('rad', 'rad', '', 'Hz')]
cell_rad=qa.convert(qa.quantity("0.1arcsec"),"rad")['value']
cs.setincrement([-cell_rad,cell_rad], 'direction')
ks.setreferencevalue([qa.convert("10h",'rad')]['value'],qa.convert("-30deg",'rad')['value'],type="direction")
ks.setreferencevalue("230GHz","spectral")
ks.setincrement("1GHz","spectral")
ia.setcoordsys(cs.torecord())
ia.setbrightnessunit("Jy/pixel")
ia.modify(cl.torecord(),subtract=False)
exportfits(imagename='Gaussian.im',fitsimage='Gaussian.fits',overwrite=True)

# In CASA
os.system('rm -rf point.cl')
cl.done()
cl.addcomponent(dir="J2000 10h00m00.08s -30d00m02.0s", flux=0.1, fluxunit='Jy', freq='230.0GHz', shape="point")
cl.addcomponent(dir="J2000 09h59m59.92s -29d59m58.0s", flux=0.1, fluxunit='Jy', freq='230.0GHz', shape="point")
cl.addcomponent(dir="J2000 10h00m00.40s -29d59m55.0s", flux=0.1, fluxunit='Jy', freq='230.0GHz', shape="point")
cl.addcomponent(dir="J2000 09h59m59.60s -30d00m05.0s", flux=0.1, fluxunit='Jy', freq='230.0GHz', shape="point")
cl.rename('point.cl')
cl.done()

# In CASA
default("simobserve")
project = "FITS_list"
skymodel = "Gaussian.fits"
inwidth = "1GHz"
complist = 'point.cl'
comppwidth = '1GHz'
direction = "J2000 10h00m00.0s -30d00m00.0s"
obsmode = 'int'
antennalist = 'alma.cycle6.1.cfg'
totaltime = "28800s"
mapsize = "10arcsec"
thermalnoise = "" 
simobserve()
```
Simulating Component Lists

Use CASA tools to create FITS file of Gaussian

Create component list

Create simulated MS of “skymodel”
Simulating Component Lists

CASA Guides: https://casaguides.nrao.edu/

Create component list
Create simulated MS of "skymodel"
Simulating Component Lists

CASA Guides: https://casaguides.nrao.edu/

```python
# In CASA
direction = "J2000 10h00m00.0s -30d00m00.0s"
cl.done()
cl.addcomponent(dir=direction, flux=1.0, fluxunit='Jy', freq='230.0GHz', shape='Gaussian',
default("simanalyze")
project = "FITS_list"
vis="FITS_list.alma.cycle6.1.ms"
imsize = [256,256]
imdirection = "J2000 10h00m00.0s -30d00m00.0s"
cell = '0.1arcsec'
niter = 5000
threshold = '10.0mJy/beam'
analyze = True
simanalyze()
```

Observations

True sky model

Importance of ALMA simulations!!
Questions?
Five Minute Break
Hands on
SIMALMA DEMO
SIMALMA

• simalma — simulate an ALMA observation including multiple configurations of the 12-m interferometric array, the 7-m ACA, and total power measurements by streamlining the capabilities of both simobserve and simanalyze.

• Simulating interferometric observations using the simobserve and simanalyze tasks proceeds in the following steps:
1. Make a model image or component list. The model is a representation of the sky brightness distribution that you would like to simulate observing.
   - Existing previous image of your target or similar target
   - Component list (point sources, Gaussians, disks, and limb-darkened disks)

1. Uses the `simobserve` task to create a Measurement Set (uv data) that would be measured by a telescope observing the specified input model of sky brightness. `simobserve` can also introduce corruption modeling thermal noise or atmospheric effects.
Generating visibilities with `simobserve`

The task `simobserve` takes several steps to generate observed visibilities. The major steps are:

- **Modify Model**: If desired, you can modify the header parameters in your data model to mimic different observing targets. For example, if you start with a model of M100 you might wish to scale the axes to simulate an observation of an M100-like galaxy that is 4X more distant.
- **Set Pointings**: If the angular size of your model image is comparable or larger than the 12-m primary beam, you can simulate observing the target as a mosaic. In this step, the individual pointings are determined and saved in a text file. You can also generate such a text file yourself.
- **Generate visibilities**: The visibilities are determined based on the telescope and configuration specified, and the length in time of the observation.
- **Finally, noise can be added to the visibilities**: The `simobserve` task uses the `aatm` atmospheric model (based on Juan Pardo's `ATM` library) to simulate real observing conditions. It can corrupt the data with thermal noise and atmospheric attenuation. Corruption with an atmospheric phase screen, or adding gain fluctuations or drift, can be added subsequently using the `simulator` tool `sm` as described in this CASA guide.
SimObserve: Files Created

Task output

Below is a list of the products produced by the `simobserve` task. Not all of these will necessarily be produced, depending on input parameters selected.

**NOTE:** To support different runs with different arrays, the names have the configuration name from antenna list 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 clean 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 MeasurementSet`
- `[project].[cfg].noisy.ms = corrupted MeasurementSet`
- `[project].[cfg].observe.png = diagnostic figure of uv coverage and visibilities`
- `[project].[cfg].simobserve.last = saved input parameters for `simobserve` task`
3. Image (grid, invert, and deconvolve) the simulated observation(s) with the simanalyze task. simanalyze can also compare the simulated image with your input (convolved with the output clean beam) and then calculate a "fidelity image" that indicates how well the simulated output matches the convolved input image.

- Alternately, you can create an image yourself with the tclean task, and then use simanalyze to compare that to the sky model input.
**Summary**

This task is for imaging and analyzing MeasurementSets (MSs) simulated with `simobserve` or `simalma`.

`simanalyze` analyzes one or more MeasurementSets - interferometric and/or single dish, using CASA's `tclean` task. It can also calculate and display the difference between the simulated observation and the original model data, and generate a "fidelity image". Fidelity is defined as:

\[
\frac{I}{|I - T|}
\]

where \(I\) is the observed image intensity and \(T\) is the true image intensity, given in this case by the sky model (see ALMA memo 398 for description of fidelity). The input parameters are therefore grouped by the two main pieces of functionality:

1. **Image** - Image the visibility data with CASA's `tclean` task. Most of the parameters are passed to the wrapper method `simutil.intclean`, which in turn calls `tclean`.

2. **Analyze** - Calculate and display the difference between output and input, and the fidelity image. Different diagnostic images can be chosen to plot on a multi-panel figure, with the different show parameters. That figure can be saved as a .png file if `graphics='both' or graphics='file'`.

The output is a synthesized image, a difference image between the synthesized image and your sky model convolved with the output synthesized beam, and a fidelity image.

**NOTE:** If you prefer to run `tclean` manually (e.g., to interactively clean with a mask), you can do that, and then use `simanalyze` to convolve the sky model and create difference and fidelity images by setting `image=False`. 
Simanlyze: Files Created

Task output

Below is a list of the products produced by the simanlyze task. Not all of these will necessarily be produced, depending on the input parameters selected.

NOTE: To support various runs using differing arrays, the file names have the configuration name from the antenna list appended.

- [project].[cfg].skymodel.flat.regrid.conv = input sky regridded to match the output image, and convolved with the output clean beam
- [project].[cfg].image = synthesized image
- [project].[cfg].pb.pbcov = primary beam correction for a mosaic image
- [project].[cfg].residual = residual image after cleaning
- [project].[cfg].tclean.last = parameter file of what parameters were used in the tclean task
- [project].[cfg].psf = synthesized (dirty) 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].simanlyze.last = saved input parameters for simanlyze task, available in CASAshell
casa-feedback@nrao.edu

CASA website: https://casa.nrao.edu/
CASA Docs: https://casa.nrao.edu/casadocs/
CASA Guides: https://casaguides.nrao.edu/
www.nrao.edu
science.nrao.edu
public.nrao.edu

The National Radio Astronomy Observatory is a facility of the National Science Foundation operated under cooperative agreement by Associated Universities, Inc.