developer.skatelescope.org Documentation

Release 0.1.0-beta

Marco Bartolini

Mar 17, 2021

HOME

This package collects RASCIL scripts for various SKA-MID simulations: continuum imaging, continuum imaging with direction-dependent effects, high data rate observations, and observation sizing.

CHAPTER

ONE

CONTINUUM IMAGING

These scripts simulate MID continuum imaging observations of multiple point sources.

- The sky model is constructed from Oxford S3-SEX catalog. These are unpolarised point sources.
- The observation is by MID or MEERKAT+ over a range of hour angles.
- The visibility is calculated by Direct Fourier transform after application of the gaintable for each source.
- Dask is used to distribute the processing over a number of workers.
- Processing can be divided into chunks of time (default 1800s).

The core simulation functions reside in RASCIL. The RASCIL driver script for simulation in this repository is direction_dependent/src/mid_simulation.py. This driver scripts can be run directly using the command line arguments listed below. Canonical bash scripts have been provided in continuum_imaging/scripts. The scripts will need to be altered for location of the various files needed. We recommend use of the bash scripts at first.

These simulations typically requiring a cluster to run. In Cambridge They work well on the P3 cluster using 16 nodes of 128GB each.

There are types of output: MeasurementSets and Images. Each are calculated for actual, nominal, and difference. The time consumed in the calculation of the MeasurementSets is small compared to the time in writing the MeasurementSet and the time to make the images. Images may be calculated using the dist_imager.

The fits images can be viewed using the casaviewer or carta. The MeasurementSets can be viewed using casaviewer.

1.1 Effects simulated

1.1.1 Nominal

- A set of point sources is simulated and the relevant nominal voltage pattern for each end of interferometer is applied before Fourier transform. The nominal pattern is constructed from a tapered symmetric illumination pattern, with the diameter of the SKA dishes.
- stokesIQUV, band B2

1.1.2 Heterogenous

- A set of point sources is simulated and the relevant voltage pattern for each end of interferometer is applied before Fourier transform.
- Simulates observations with all SKA and MEERKAT dishes (on state) and all SKA dishes (off state).
- stokesIQUV, band B2

1.1.3 Heterogenous_meerkat+

- A set of point sources is simulated and the relevant voltage pattern for each end of interferometer is applied before Fourier transform.
- Simulates observations with MEERKAT+ configuration, all SKA and MEERKAT dishes (on state) and all SKA dishes (off state).
- stokesIQUV, band B2

1.1.4 Ionosphere

- A set of point sources is simulated and the phases calculated using a thin screen model for the ionosphere. The screen has units of meters of Total Electron Content. The phase is evaluated at places in the screen where the line of sight from source to a dish pierces the screen.
- Simulates observations with ionospheric screens on and off.
- This requires the screens to be calculated first or downloaded: see ../screens/README.md
- stokesI, band B1LOW

1.1.5 Polarisation

- Simulates observations with SKA and EMSS calculated primary beam models with cross pol (on) and no cross pol set to zero (off)
- Polarisation stokesIQUV, linear, band B2

1.1.6 Surface sag

- Simulates observations with sagging dish (on) and nominal dish (off).
- Models of the voltage pattern are available at +15, +45, +90 deg elevation.
- We interpolate between those to 5 degrees.
- Stokes I, band B2

For more details see: https://confluence.skatelescope.org/display/SE/Dish+deformation+simulations

1.1.7 Troposphere

- Simulates observations with tropospheric screens on and off.
- This requires the screens to be calculated first or downloaded: see ../screens/README.md
- stokesI, bands B2 and B5
- A set of point sources is simulated and the phases calculated using a thin screen model for the atmosphere. The phase is evaluated at places in the screen where the line of sight from source to dish pierces the screen. The screen has units of meters of delay.

1.1.8 Wind pointing

- Simulates observations with wind buffeting of dishes (on) and without (off)
- Stokes I, bands B2 and B5

For more details see: https://confluence.skatelescope.org/display/SE/MID+pointing+error+simulations

Simulate SKA-MID direction dependent errors

```
usage: mid_simulation.py [-h] [--rmax RMAX] [--configuration CONFIGURATION]
                         [--antennas [ANTENNAS [ANTENNAS ...]]] [--ra RA]
                         [--declination DECLINATION] [--band BAND]
                         [--nchan NCHAN] [--channel_width CHANNEL_WIDTH]
                         [--integration_time INTEGRATION_TIME]
                         [--time_range TIME_RANGE TIME_RANGE]
                         [--image_pol IMAGE_POL] [--sim_pol SIM_POL]
                         [--vis_pol VIS_POL] [--pbradius PBRADIUS]
                         [--pbtype PBTYPE] [--seed SEED]
                         [--flux_limit FLUX_LIMIT] [--results RESULTS]
                         [--elevation_sampling ELEVATION_SAMPLING] [--r0 R0]
                         [--height HEIGHT] [--screen SCREEN]
                         [--time_chunk TIME_CHUNK] [--mode MODE]
                         [--duration DURATION]
                         [--wind_conditions WIND_CONDITIONS]
                         [--global_pe GLOBAL_PE GLOBAL_PE]
                         [--static_pe STATIC_PE STATIC_PE]
                         [--dynamic_pe DYNAMIC_PE]
                         [--pointing_directory POINTING_DIRECTORY]
                         [--vp_directory VP_DIRECTORY] [--nthreads NTHREADS]
                         [--processes PROCESSES] [--memory MEMORY]
                         [--nworkers NWORKERS] [--use_dask USE_DASK]
                         [--write_hdf WRITE_HDF]
                         [--imaging_dft_kernel IMAGING_DFT_KERNEL]
```

Named Arguments

--rmax Maximum distance of dish from centre (m)

Default: 200000.0

--configuration MID Configuration: MID | MEERKAT+

Default: "MID"

--antennas antenna names to include (default is all)

--ra Right ascension of phase centre (degrees)

Default: 0.0

--declination Declination of phase centre (degrees)

Default: -40.0

--band Band B1LOW | B1 | B2 | Ku

Default: "B2"

--nchan Number of frequency channels

Default: 1

--channel_width Channel bandwidth (Hz) (default is to calculate from frequency

--integration_time Integration time (s)

Default: 180

--time_range Time range in hour angle

Default: [-4.0, 4.0]

--image_pol RASCIL polarisation frame for image: stokesI | stokes IQ | stokesIQUV

Default: "stokesIQUV"

--sim_pol RASCIL polarisation frame for simulation: stokesI | stokes IQ | stokesIQUV

Default: "stokesI"

--vis_pol RASCIL polarisation frame for output visibility: linear | linearnp

Default: "linear"

--pbradius Radius of sources to include (in HWHM)

Default: 1.5

--pbtype Primary beam model: MID_B1 MID_B1LOW MID_B2 MID_Ku MEERKAT_B1 MEERKAT_B2

Default: "MID_B2"

--seed Random number seed

Default: 18051955

--flux_limit Flux limit (Jy)

Default: 0.01

--results Directory for results

Default: "./"

--elevation_sampling Sampling in elevation for surface (deg)

Default: 1.0

--r0 R0 (meters)

Default: 5000.0

--height Height of layer (meters)

Default: 300000.0

--screen Location of atmospheric phase screen

Default: <MagicMock name='mock()' id='140443122806992'>

--time_chunk Time for a chunk (s)

Default: 3600.0

--mode Mode of simulation: wind_pointing|random_pointing|polarisation|ionosphere|troposphere|heterogeneous

Default: "none"

--duration Type of duration: long or medium or short

Default: "long"

--wind_conditions SKA definition of wind conditions: precision or standard or degraded

Default: "precision"

--global_pe Global pointing error

Default: [0.0, 0.0]

--static_pe Multipliers for static errors

Default: [0.0, 0.0]

--dynamic_pe Multiplier for dynamic errors

Default: 1.0

--pointing_directory Location of wind PSD pointing files

Default: <MagicMock name='mock()' id='140443122806992'>

--vp_directory Location of voltage pattern files

--nthreads Number of threads per worker

--processes Number of processes per worker

Default: 1

--memory Memory per worker (GB)

--nworkers Number of workers

--use_dask Use dask processing?

Default: "True"

--write_hdf Write intermediate files as HDF

Default: "False"

--imaging_dft_kernel DFT kernel: cpu_looped | cpu_numba | gpu_raw

CHAPTER

TWO

DIRECTION-DEPENDENT EFFECTS

These scripts simulate MID observations of multiple point sources and calculate the effect of direction dependent gain errors.

- The sky model is constructed from Oxford S3-SEX catalog. These are unpolarised point sources.
- The observation is by MID or MEERKAT+ over a range of hour angles.
- The visibility is calculated by Direct Fourier transform after application of the gaintable for each source.
- Dask is used to distribute the processing over a number of workers.
- Processing can be divided into chunks of time (default 1800s). This allows

See the following presentation for an overview of the results:

> Simulating MID direction dependent gain effects SPO-1057

The core simulation functions reside in RASCIL. The RASCIL driver script in this repository is direction_dependent/src/mid_simulation.py. This can be run directly using the command line arguments listed below, or some typical bash scripts have been provided in direction_dependent_src/scripts. The bash scripts allow for looping over duration and declination. The scripts will need to be altered for location of the various files needed. We recommend use of the bash scripts at first.

Note that the simulation has two steps: first the visibilities are calculated and written to HDF files, using mid_simulation.py and then all the HDF files are combined into one MeasurementSet using convert_to_ms.py. In this conversion step, a number of diagnostic plots are written.

These simulations typically requiring a cluster to run. In Cambridge They work well on the P3 login node, which has 512GB and 64 threads/32 cores, using a large number of threads. For this approach –use_slurm should be False.

There are types of output: MeasurementSets and Images. Each are calculated for actual, nominal, and difference. The time consumed in the calculation of the MeasurementSets is small compared to the time in writing the MeasurementSet and the time to make the images. Images may be calculated using the dist_imager.

The fits images can be viewed using the casaviewer or carta. The MeasurementSets can be viewed using casaviewer.

2.1 Effects simulated

2.1.1 Nominal

- A set of point sources is simulated and the relevant nominal voltage pattern for each end of interferometer is applied before Fourier transform. The nominal pattern is constructed from a tapered symmetric illumination pattern, with the diameter of the SKA dishes.
- stokesIQUV, band B2

2.1.2 Heterogenous

- A set of point sources is simulated and the relevant voltage pattern for each end of interferometer is applied before Fourier transform.
- Simulates observations with all SKA and MEERKAT dishes (on state) and all SKA dishes (off state).
- stokesIQUV, band B2

2.1.3 Heterogenous_meerkat+

- A set of point sources is simulated and the relevant voltage pattern for each end of interferometer is applied before Fourier transform.
- Simulates observations with MEERKAT+ configuration, all SKA and MEERKAT dishes (on state) and all SKA dishes (off state).
- stokesIQUV, band B2

2.1.4 Ionosphere

- A set of point sources is simulated and the phases calculated using a thin screen model for the ionosphere. The screen has units of meters of Total Electron Content. The phase is evaluated at places in the screen where the line of sight from source to a dish pierces the screen.
- Simulates observations with ionospheric screens on and off.
- This requires the screens to be calculated first or downloaded: see ../screens/README.md
- stokesI, band B1LOW

2.1.5 Polarisation

- Simulates observations with SKA and EMSS calculated primary beam models with cross pol (on) and no cross pol set to zero (off)
- Polarisation stokesIQUV, linear, band B2

2.1.6 Surface sag

- Simulates observations with sagging dish (on) and nominal dish (off).
- Models of the voltage pattern are available at +15, +45, +90 deg elevation.
- We interpolate between those to 5 degrees.
- Stokes I, band B2

For more details see: https://confluence.skatelescope.org/display/SE/Dish+deformation+simulations

2.1.7 Troposphere

- Simulates observations with tropospheric screens on and off.
- This requires the screens to be calculated first or downloaded: see ../screens/README.md
- stokesI, bands B2 and B5
- A set of point sources is simulated and the phases calculated using a thin screen model for the atmosphere. The phase is evaluated at places in the screen where the line of sight from source to dish pierces the screen. The screen has units of meters of delay.

2.1.8 Wind pointing

- Simulates observations with wind buffeting of dishes (on) and without (off)
- Stokes I, bands B2 and B5

For more details see: https://confluence.skatelescope.org/display/SE/MID+pointing+error+simulations

Simulate SKA-MID direction dependent errors

```
usage: mid_simulation.py [-h] [--rmax RMAX] [--configuration CONFIGURATION]
                         [--antennas [ANTENNAS [ANTENNAS ...]]] [--ra RA]
                         [--declination DECLINATION] [--band BAND]
                         [--nchan NCHAN] [--channel_width CHANNEL_WIDTH]
                         [--integration_time INTEGRATION_TIME]
                         [--time_range TIME_RANGE TIME_RANGE]
                         [--image_pol IMAGE_POL] [--sim_pol SIM_POL]
                         [--vis_pol VIS_POL] [--pbradius PBRADIUS]
                         [--pbtype PBTYPE] [--seed SEED]
                         [--flux_limit FLUX_LIMIT] [--results RESULTS]
                         [--elevation_sampling ELEVATION_SAMPLING] [--r0 R0]
                         [--height HEIGHT] [--screen SCREEN]
                         [--time_chunk TIME_CHUNK] [--mode MODE]
                         [--duration DURATION]
                         [--wind_conditions WIND_CONDITIONS]
                         [--global_pe GLOBAL_PE GLOBAL_PE]
                         [--static_pe STATIC_PE STATIC_PE]
                         [--dynamic_pe DYNAMIC_PE]
                         [--pointing_directory POINTING_DIRECTORY]
                         [--vp_directory VP_DIRECTORY] [--nthreads NTHREADS]
                         [--processes PROCESSES] [--memory MEMORY]
                         [--nworkers NWORKERS] [--use_dask USE_DASK]
                         [--write_hdf WRITE_HDF]
                         [--imaging_dft_kernel IMAGING_DFT_KERNEL]
```

Named Arguments

--rmax Maximum distance of dish from centre (m)

Default: 200000.0

--configuration MID Configuration: MID | MEERKAT+

Default: "MID"

--antennas antenna names to include (default is all)

--ra Right ascension of phase centre (degrees)

Default: 0.0

--declination Declination of phase centre (degrees)

Default: -40.0

--band Band B1LOW | B1 | B2 | Ku

Default: "B2"

--nchan Number of frequency channels

Default: 1

--channel_width Channel bandwidth (Hz) (default is to calculate from frequency

--integration_time Integration time (s)

Default: 180

--time_range Time range in hour angle

Default: [-4.0, 4.0]

--image_pol RASCIL polarisation frame for image: stokesI | stokes IQ | stokesIQUV

Default: "stokesIQUV"

--sim_pol RASCIL polarisation frame for simulation: stokesI | stokes IQ | stokesIQUV

Default: "stokesI"

--vis_pol RASCIL polarisation frame for output visibility: linear | linearnp

Default: "linear"

--pbradius Radius of sources to include (in HWHM)

Default: 1.5

--pbtype Primary beam model: MID_B1 MID_B1LOW MID_B2 MID_Ku MEERKAT_B1 MEERKAT_B2

Default: "MID_B2"

--seed Random number seed

Default: 18051955

--flux_limit Flux limit (Jy)

Default: 0.01

--results Directory for results

Default: "./"

--elevation_sampling Sampling in elevation for surface (deg)

Default: 1.0

--r0 R0 (meters)

Default: 5000.0

--height Height of layer (meters)

Default: 300000.0

--screen Location of atmospheric phase screen

Default: <MagicMock name='mock()' id='140443122806992'>

--time_chunk Time for a chunk (s)

Default: 3600.0

--mode Mode of simulation: wind_pointing|random_pointing|polarisation|ionosphere|troposphere|heterogeneous

Default: "none"

--duration Type of duration: long or medium or short

Default: "long"

--wind_conditions SKA definition of wind conditions: precision or standard or degraded

Default: "precision"

--global_pe Global pointing error

Default: [0.0, 0.0]

--static_pe Multipliers for static errors

Default: [0.0, 0.0]

--dynamic_pe Multiplier for dynamic errors

Default: 1.0

--pointing_directory Location of wind PSD pointing files

Default: <MagicMock name='mock()' id='140443122806992'>

--vp_directory Location of voltage pattern files

--nthreads Number of threads per worker

--processes Number of processes per worker

Default: 1

--memory Memory per worker (GB)

--nworkers Number of workers

--use_dask Use dask processing?

Default: "True"

--write_hdf Write intermediate files as HDF

Default: "False"

--imaging_dft_kernel DFT kernel: cpu_looped | cpu_numba | gpu_raw

CHAPTER

THREE

PSI SIMULATIONS

These are simulated visibility data sets in Measurement Set format, as needed to test the operation of the CBF-SDP interface and the real-time signal displays.

These simulations generate fully time-sampled, high number of spectral channel observations using the simulated Prototype System Integration (PSI) test arrays: a small number of antennas or stations (between 4 and 8), with a large number of frequency channels (of order 10000).

The simulation currently does:

- 10 minutes of 0.28s integrations
- six antennas chosen to be close to the centre but to give baselines upto 200m
- 50% fractional bandwidth: centre frequency 1.369GHz +/- 0.5 * 1.369GHz

The simulation requires a large amount of memory, primality because of inefficiencies in the RASCIL visibility format. We expect to reduce these over time. The best Dask setup on Alaska P3 is to use one worker and a large number of threads (up to 64).

Simulate time and frequency dense PSI observations

```
usage: mid_write_psi.py [-h] [--rmax RMAX] [--configuration CONFIGURATION]
                        [--antennas [ANTENNAS [ANTENNAS ...]]]
                        [--badantenna BADANTENNA] [--howbad HOWBAD] [--ra RA]
                        [--declination DECLINATION] [--band BAND]
                        [--nchan NCHAN] [--nspw NSPW]
                        [--channel_width CHANNEL_WIDTH]
                        [--integration_time INTEGRATION_TIME]
                        [--time_range TIME_RANGE TIME_RANGE]
                        [--image_pol IMAGE_POL] [--vis_pol VIS_POL]
                        [--duration DURATION] [--results RESULTS]
                        [--msname MSNAME] [--split SPLIT]
                        [--nthreads NTHREADS] [--processes PROCESSES]
                        [--memory MEMORY] [--nworkers NWORKERS]
                        [--use_dask USE_DASK] [--interface INTERFACE]
```

3.1 Named Arguments

--rmax	Maximum distance of dish from centre (m) Default: 200.0
--configuration	MID Configuration: MID | MEERKAT+ Default: "MID"
--antennas	antenna names to include (default is all)
--badantenna	Bad antenna Default: -1
--howbad	How many meters to move bad antenna Default: 1
--ra	Right ascension of phase centre (degrees) Default: 0.0
--declination	Declination of phase centre (degrees) Default: -55.0
--band	Band B1LOW | B1 | B2 | Ku Default: "B2"
--nchan	Number of frequency channels Default: 1024
--nspw	Number of spectral windows Default: 8
--channel_width	Channel bandwidth (Hz) (default is to calculate from frequency
--integration_time	Integration time (s) Default: 0.14
--time_range	Time range in hour angle Default: [0, 0.16667]
--image_pol	RASCIL polarisation frame for image: stokesI | stokes IQ | stokesIQUV Default: "stokesIQUV"
--vis_pol	RASCIL polarisation frame for visibility: linear | linearnp Default: "linear"
--duration	Type of duration: long or medium or short Default: "long"
--results	Directory for results Default: "./"
--msname	Root name of MS (i.e. without .ms) Default: "sim_mid_psi"

--split In concat, type of tree e.g. 2->binary

Default: 2

--nthreads Number of threads per worker

Default: 1

--processes Number of processes per worker

Default: 1

--memory Memory per worker (GB)

--nworkers Number of workers

Default: 4

--use_dask Use dask processing?

Default: "True"

--interface Network interface

Default: "ib0"

Todo:

- Insert todo's here

CHAPTER

FOUR

SIZING OBSERVATIONS

This script estimates the size of CASA MeasurementSets for various MID observations. it does this by constructing a scaled down BlockVisibility and writing it to a MeasurementSet.

The maximum size of the array (maximum distance from the array centre) can be set. This determines the angular resolution. See parameter rmax.

The field of view to be imaged is set to a multiple of the primary beam. See parameter guardband.

The time and frequency sampling is adjusted to match the SKA specification of 2% decorrelation at the half-power point of the primary beam. See parameter dela.

Calculate MeasurementSets sizes for MID observations

```
usage: mid_size.py [-h] [--output_msname OUTPUT_MSNAME] [--context CONTEXT]
                   [--ra RA] [--declination DECLINATION] [--rmax RMAX]
                   [--band BAND] [--integration_time INTEGRATION_TIME]
                   [--time_range TIME_RANGE TIME_RANGE]
                   [--fractional_bandwidth FRACTIONAL_BANDWIDTH]
                   [--guardband GUARDBAND] [--dela DELA] [--verbose VERBOSE]
```

4.1 Named Arguments

--output_msname MeasurementSet to be written

Default: “mid_simulation.ms”

--context s3sky or singlesource or null

Default: “s3sky”

--ra Right ascension (degrees)

Default: 15.0

--declination Declination (degrees)

Default: -45.0

--rmax Maximum distance of station from centre (m)

Default: 1000.0

--band Band

Default: “B2”

--integration_time Integration time (s)

Default: 80.0

--time_range Time range in hours

Default: [-4.0, 4.0]

--fractional_bandwidth Fractional bandwidth

Default: 0.1

--guardband Size of field of view relative to primary beam

Default: 2.0

--dela Allowable degradation of point source at half power point of primary beam

Default: 0.02

--verbose Verbose output?

Default: "False"

Typical output:

```
mid_size: Starting MID simulation sizing

{'band': 'B2',
 'context': 's3sky',
 'declination': -45.0,
 'dela': 0.02,
 'fractional_bandwidth': 0.1,
 'guardband': 2.0,
 'integration_time': 80.0,
 'output_msname': 'mid_simulation.ms',
 'ra': 15.0,
 'rmax': 1000.0,
 'time_range': [-4.0, 4.0],
 'verbose': 'False'}
mid_size: Using only dishes within 1000.0m of the array centre requires 12 channels␣
↪of 1.13e+07Hz and integration time 683.8s, the MS size is 0.338GB
mid_size: The time and frequency sampling is to provide no more than 2% smearing at␣
↪the half power point of the primary beam.
mid_size: Image has shape 768 by 768, 4 polarisations, and size 0.018GB
mid_size: W processing requires 49 w planes of step 238.8 wavelengths and maximum␣
↪support 0 pixels
mid_size: Allowing fractional bandwidth 0.100  to fill in the uv sampling.
nmid_size: Starting MID simulation sizing

{'band': 'B2',
 'context': 's3sky',
 'declination': -45.0,
 'dela': 0.02,
 'fractional_bandwidth': 0.1,
 'guardband': 2.0,
 'integration_time': 80.0,
 'output_msname': 'mid_simulation.ms',
 'ra': 15.0,
 'rmax': 10000.0,
 'time_range': [-4.0, 4.0],
 'verbose': 'False'}
```

(continues on next page)

(continued from previous page)

```
mid_size: Using only dishes within 10000.0m of the array centre requires 107 channels␣
↪of 1.27e+06Hz and integration time 80.4s, the MS size is 50.382GB
mid_size: The time and frequency sampling is to provide no more than 2% smearing at␣
↪the half power point of the primary beam.
mid_size: Image has shape 8192 by 8192, 4 polarisations, and size 2.000GB
mid_size: W processing requires 428 w planes of step 238.8 wavelengths and maximum␣
↪support 12 pixels
mid_size: Allowing fractional bandwidth 0.100  to fill in the uv sampling.
nmid_size: Starting MID simulation sizing

{'band': 'B2',
 'context': 's3sky',
 'declination': -45.0,
 'dela': 0.02,
 'fractional_bandwidth': 0.1,
 'guardband': 2.0,
 'integration_time': 80.0,
 'output_msname': 'mid_simulation.ms',
 'ra': 15.0,
 'rmax': 200000.0,
 'time_range': [-4.0, 4.0],
 'verbose': 'False'}
mid_size: Using only dishes within 200000.0m of the array centre requires 1161␣
↪channels of 1.17e+05Hz and integration time 7.4s, the MS size is 8512.629GB
mid_size: The time and frequency sampling is to provide no more than 2% smearing at␣
↪the half power point of the primary beam.
mid_size: Image has shape 98304 by 98304, 4 polarisations, and size 288.000GB
mid_size: W processing requires 4065 w planes of step 238.8 wavelengths and maximum␣
↪support 132 pixels
mid_size: Allowing fractional bandwidth 0.100  to fill in the uv sampling.
```

INSTRUCTIONS

To install these scripts, use git:

```
git clone https://gitlab.com/ska-telescope/sim/ska-sim-mid.git
```

Or download the source code from https://gitlab.com/ska-telescope/sim/ska-sim-mid

None of these simulations need to be installed into the python path. Instead please set the environment variable SMSROOT to the location of code:

```
export SMSROOT=/path/to/code
```

RASCIL must be installed - the options are via pip, git clone, or docker. The simplest approach is to use pip. In the ska-sim-mid directory, do:

```
pip3 install -r requirements.txt
```

A script can be run as:

```
export SMSROOT=`pwd`
cd sizing/scripts
sh make_sizes.sh
```

The resource files for the SKA direction-dependent MID simulations are 80GB in size and are kept on the Google Cloud Platform. The python command line tool gsutil allows for interacting with the Google Cloud Platform:

```
https://cloud.google.com/storage/docs/gsutil
```

After installing gsutil, you may download the resources as follows:

```
cd resources
gsutil -m rsync -r gs://ska1-simulation-data/ska1-mid/beam_models/ beam_models
gsutil -m rsync -r gs://ska1-simulation-data/ska1-mid/screens screens
```