BoloCalc User Manual
Version 0.b11.6

Charles Hill, UC Berkeley

Last updated: 2020-06-01

Contents

1 Revision History

All BoloCalc releases are tagged in the git repository. To retrieve the latest release, pull from the origin using `git pull`, list the revision history using `git tag`, and checkout the latest version using (e.g. v0.10.0) `git checkout v0.10.0`. Below is a revision history for BoloCalc's most recent releases.

1.1 v0.11

- Removed the option "ROOM" for the **Site** telescope parameter and added the parameter **Sky Temperature**, which allows the user to set the physical temperature of the sky explicitly (without having to build a custom atmosphere file first).
- Changed the "Percentile" simulation input to "Percentile Lo" and "Percentile Hi"
- New output handling class called `Unpack`, which unpacks the output sensitivity and parameter sweep files into Python dictionaries. See Section 5.2 for details on how to use the `Unpack` class.
- New folder `analyze/` with a new Jupyter notebook `output_handling_examples.ipynb`, which is an interactive tutorial for how to unpack and plot BoloCalc outputs. Additionally, a PDF render of this Jupyter notebook located in the `BoloCalc/MANUAL` directory. Again, see for more details.
- New capability for parameter sweeps using custom input files. See Section 4.2.1 for more information.
- New equation and parameter quick reference guide in the `BoloCalc/MANUAL/` directory

1.2 v0.10.

- BoloCalc no longer supports `Python 2` because it will no longer be maintained starting Jan 1, 2020.
- `mappingSpeed.py` and `mappingSpeed_vary.py` are no combined into a single command-line executable `calcBolos.py`. See section 2.5 to learn about the program's capabilities and command-line arguments.
- Probability distributions may now be passed for any input parameter using keyword "PDF." See section 3.2.2.
- Transmission, absorption, reflection, scattering, and spillover spectra may now be defined for any optic via the keyword "BAND." See section 3.6.3 for syntax details.
- `channels.txt` accepts a new parameter **Resp Factor**, which is used to help define detector responsivity, which impacts readout noise. See 6.7 for details about the readout noise calculation, and see Section 3.5.2 to review the parameter's usage.
- Foreground parameters are defined in friendlier units: MJysr^{-1} for dust and $\mathrm{K_{RJ}}$ for synchrotron. See section 3.3.1 for foreground parameter definitions and Sections 6.14.1 and 6.14.2 for the details of the foreground calculation.
- Output parameter spreads are now formatted as `median +/- [lo, hi]`, where `lo` and `hi` are user-defined in `config/simulationInputs.txt` as percentiles. See 4.1 for more details.
- Several new output parameters have been added, including optical throughput, telescope temperature, sky temperature, NET_{RJ}, RJ Map Depth, and the correlation factor. See Section 5.1.1 for more details about output parameters.
- Full outputs, with every simulated value for every experiment, sky, and detector realization, are now saved in `[Camera]/config/output.txt`. See section 5.1.1 for a full list of outputs and their descriptions and Section 5.1 for `output.txt` formatting.
- Atmosphere spectra have been moved from directories within `src/atmFiles` to the HDF5 file `src/atm.hdf5`. The specta themselves have not changed. Use the script `BoloCalc/update_atm.py` to retrieve the new HDF5 file. Note that if not already in the

Python environment, the user will need to download and source the Python package H5Py. For more information about how to get H5Py, see Section 2.1.

- Custom atmosphere files are now passed explicitly using the *Site* keyword "CUST" in the `telescope.txt` parameter file. See Section 6.14.3 for more details regarding atmosphere handling.
- Added a new option "Room" for the parameter **Site** in input file `telescope.txt` which is a useful utility for estimating optical load during lab testing.
- Custom bands are now passed explicitly using the keyword "BAND" in either the `channels.txt` parameter file or in the `optics.txt` parameter file. See Sections 3.6.3 and 3.6.4 for more details about custom bands.
- Full outputs, with every simulated value for every experiment, sky, and detector realization are also output for `calcBolos.py --vary` in an indexed file set. See 5.1.4 for details on vary output, and see Sections 2.5 and 4.2 for more details about simulating parameter sweeps.
- User-defined detector bands can now have their integrated efficiency modulated via the parameter **Def Eff** and their band center varied by the parameter `+/- [SPREAD]` in the parameter **Band Center**. See Section 3.6.4 for details about modulating custom detector bands.
- Multi-processing is no longer supported for BoloCalc at this time.
- BoloCalc now requires numpy v1.13 or higher, as opposed to the previous release, which required v1.8.
- BoloCalc has been restructured for improved efficiency and is now ten times faster, providing more feasibility and power to the Monte Carlo functionality.
- The BoloCalc source code has been substantially restructured for improved readability, hackability, and upgrade-ability. Doc strings have been added to all classes and methods, the logging has been substantially improved, code redudancy has been reduced, and all syntax now follows the PEP8 style guidelines.

Given the above list of upgrades, the following steps should be taken by the user to update their experiment data to be compatible with the latest BoloCalc version.

1. If you are a continuing BoloCalc user, run `update_atm.sh` to get the latest atmosphere HDF5 file and use `conda install` or `pip install` to obtain the `h5py` package. See Section 2.1 for more information about python package requirements and Section 2.3 for more information about downloading the HDF5 file.

2. The unit and equation definitions for the foregrounds have changed. If you are including foregrounds by setting the parameter **Foregrounds** to "True" in `simulationInputs.txt`, then you will need to change your `foregrounds.txt` file. See Section 3.3.1 for more details.

3. All bands are now called out explicitly, instead of being loaded implicitly when the parameter is "NA." The band files are called using the keyword "BAND" in the parameter files `channels.txt` and `optics.txt`. See Section 3.6.3 and 3.6.4 for more details about how these custom bands are handled.

4. All probability distributions are called out explicitly. This is an important change if you have been using **PWV** or **Elevation** distributions for `telescope.txt` parameters. The distribution files are called using the keyword "PDF," and the band files are stored in either the directory `config/Dist/Optics/` or the directory `config/Dist/Detectors/`. See Section 3.2.2 for more details about handling parameter distributions.

If you'd like to get a quick feel for how these new features might be implemented, you can download a fresh version of the `ExampleExperiment`, which demonstrates the new syntax and file structure in several areas. See Section 2.4 for more details about how to download experiment data.

1.3 v0.9

- Added the capability for the detector parameter **G** to be defined explicitly rather than calculated from **n**, **Tb**, and **Tc**. This is useful for when **G** is measured directly in lab tests.
- Added the capability for the detector parameter **Flink** to be defined explicitly rather than calculated from **n**, **Tb**, and **Tc**. This is useful for when **Flink** is measured directly in lab tests.

2 Quick Start Guide

This section covers how to get BoloCalc up and running. For more detailed information about how to define an experiment, run simulations, and read output files, see the following sections.

2.1 System Requirements

BoloCalc runs using Python 3 on either Linux or Windows and has the following Python package dependencies:

1. Python v3.7 or above
2. Numpy v1.13 or above
3. h5py, which handles HDF5 files

My recommendation to acquire all useful Python packages is to install Anaconda, which will also install Jupyter, a useful platform for creating analysis notebooks. Additionally, the following packages are needed to install auxiliary experiment and atmosphere data:

- unzip
- wget

If on Linux, these packages can be obtained using `apt-get`

```
$ apt-get install unzip
$ apt-get install wget
```

On Windows, you can download unzip here and wget here. On Windows, the executables will need to be added to my `PATH` variable. For me (Windows 10 Pro, 2018), `unzip.exe` and `wget.exe` are stored at `C:\Program Files (x86)\GnuWin32\bin`, so I added this directory to my `PATH` variable. For a tutorial about how to edit the `PATH` variable on Windows, see this walk-through.

For the duration of this manual, I will use command-line snippets on Linux to demonstrate example code and executions. However, all shown examples should, in theory, work just as well on Windows.

2.2 Download BoloCalc

BoloCalc is stored on Charles Hill's GitHub. To clone a local copy of the calculator repository, check that Git is installed on your machine (I've been using Git v2.13.5), and run

```
$ git clone git@github.com:chill90/BoloCalc.git
```

to use SSH or

```
$ git clone https://github.com/chill90/BoloCalc.git
```

to use HTTPS. Executables are stored in the top-most directory `BoloCalc/` and all modules are stored in the library `BoloCalc/src/`.

After pulling the BoloCalc source code, you will need to download atmosphere and experiment data, which are stored on a separate server. To obtain and set up this auxiliary data, first change directory to BoloCalc then run

```
BoloCalc $ ./init.sh
```

This command will add the directory `BoloCalc/Experiments/ExampleExperiment` and file `BoloCalc/src/atm.hdf5` to your working tree.

BoloCalc is still under construction and therefore is frequently being updated. Therefore, it is encouraged that you update your code using the commands `git pull`, `git tag`, and `git checkout [latest tag]` to keep their source code up to date.

2.3 Download Atmosphere Data

If you are downloading and using BoloCalc for the first time, then you can ignore this section, as `init.sh` will import the latest atmosphere data for you automatically. However, if you are a continuing user upgrading to BoloCalc `v0.10.0`+ for the first time, then you will need to update your atmosphere data.

BoloCalc `v0.10.0`+ utilizes an HDF5 file to store and organize atmosphere profile data. In order to obtain the HDF5 file, which is not stored in the BoloCalc git repository, run the script `update_atm.sh` as

```
BoloCalc $ ./update_atm.sh
```

If upgrading to `v0.10.0`+ for the first time, after the download is complete, we suggest that you remove the directory `BoloCalc/src/atmFiles/` via the command

```
BoloCalc $ rm -r src/atmFiles
```

as that directory is no longer used by BoloCalc and therefore is 1 GB of dead weight.

2.4 Download Experiment Data

BoloCalc experiments can be imported using the script `Experiments/importExperiments.py` as

```
BoloCalc $ cd Experiments/
Experiments $ python3 importExperiments.py [Experiment\_ID]
```

where the experiment IDs are

- `EX`: Example experiment
- `SA`: Simons Array (password-protected)
- `SO`: Simons Observatory (password-protected)

Note that the Example Experiment is downloaded by default upon running `init.sh`, as described in Section 2.2. When attempting to download Simons Array or Simons Observatory data, you will be prompted for a user name and password, which can be obtained at the following password-protected wiki pages:

- Simons Observatory
- Simons Array

Contact Charles Hill at chill@lbl.gov to obtain permissions for your experiment if you do not have access to the above links.

For this Quick Start Guide, we aren't going to modify any experiment parameters. For information about how to build your own experiment, see Section 3.3.

2.5 Run `calcBolos.py`

The BoloCalc executable is `BoloCalc/calcBolos.py`, which simulates an experiment directory passed via the command line. The executable should be run using Python 3, and the experiment directory `[Exp_Design]` should be passed as a positional argument. The script `BoloCalc/init.sh` downloads an example experiment (see Section 2.4 for more details) that you can use to get started:

p

```
BoloCalc $ python3 calcBolos.py Experiments/ExampleExperiment/V0/
```

By default, `calcBolos.py` will calculate one experiment realization, using the median values for all input parameters. See Section 4.1 for more information about how to set simulation parameters, Section 4.3 for more information about the command line arguments for `calcBolos.py`, and Section 4.2 for more information about how to sweep over experiment parameters.

2.6 Inspect Outputs

The executable `calcBolos.py` generates several output parameters for each frequency channel within the input experiment's telescopes and cameras. These sensitivity outputs are written to tables which show the median and (default) 68% confidence interval. The tables are stored in three files named `sensitivity.txt`, each located at a different level of the input experiment's directory structure:

- `Experiments/ExampleExperiment/V0/sensitivity.txt`: contains the sensitivity, combined by frequency band, of all telescopes in this experiment.
- `Experiments/ExampleExperiment/V0/Tel/sensitivity.txt`: contains the sensitivity, combined by frequency band, of all cameras in this telescope.
- `Experiement/ExampleExperiment/V0/Tel/Cam/sensitivity.txt`: contains the sensitivity for all frequency bands in this camera.

For more information about the contents of the `sensitivity.txt` files, as well as information about other output files, see Sections 5.1.

2.7 Contact and Citation

Thank you for using BoloCalc! Please send any comments, requests, questions, or bugs to Charles Hill at `chill@lbl.gov`. If you use this code as any component of a publication, please cite Hill et al, 2019 *BoloCalc: a sensitivity calculator for the design of Simons Observatory*.

3 Constructing a BoloCalc Experiment

BoloCalc has a modular object-oriented structure, which allows for arbitrary mixtures of sites, telescopes, cameras, optics, focal planes, and detectors. A BoloCalc project has the inheritance structure shown in Figure 1 and is built with four layers: experiments, telescopes, cameras, and channels, which are defined in Table 1. Each experiment can have an arbitrary set of telescopes (at different sites), each telescope an arbitrary set of cameras, and each camera an arbitrary set of channels.

Layer	**Definition**
Experiment	An assemblage of CMB telescopes.
Telescope	A platform that carries and points one or more cameras. It observes at a specified site with a specified observation strategy and can include warm reflectors.
Camera	A cryostat that houses cryogenic optics, filters, and detectors. Multiple cameras can be mounted on the same telescope.
Channel	A frequency band observed by some set of detectors within a camera. A multichroic camera will have multiple channels.

Table 1: Definitions of the layers used to build a BoloCalc project.

3.1 Experiment Directory Layout

A BoloCalc experiment's directory structure mimics the layered inheritance structure of the source code and is designed to be generalizable and modular. For information about how experiments are stored within the `BoloCalc/Experiments/` directory, see Section 3. Below is an overview of the file structure within an `[Experiment_Design]` directory, with the first telescope and camera directories unfolded. Identifiers within square brackets are user-definable directory names, while identifiers without square brackets are directories whose names cannot be changed. Identifiers with an asterisk denote that the directory or file is optional. Formatting rule: all [Experiment_Design], [Telescope], and [Camera] names should be camelCase and cannot contain underscores.

```
BoloCalc/Experiments/[Exp_name]/[Exp_Design]/
|-- config/
|    |-- foregrounds.txt
|    |-- Dist/
|    |    |-- [*ForegroundParam1.txt]
|    |    |-- [*ForegroundParam2.txt]
|    |    |-- ...
|-- [Telescope_1]/
|    |-- config/
|    |    |-- telescope.txt
|    |    |-- Dist/
|    |    |    |-- [*Tel_param1_PDF.txt]
|    |    |    |-- [*Tel_param2_PDF.txt]
|    |    |    |-- ...
|    |-- [Camera_1]/
|    |    |-- config/
|    |    |    |-- camera.txt
|    |    |    |-- channels.txt
|    |    |    |-- optics.txt
|    |    |    |-- *elevation.txt
|    |    |    |-- Bands/
|    |    |    |    |-- Optics/
|    |    |    |    |    |-- [*Optic1_Param.txt]
|    |    |    |    |    |-- [*Optic2_Param.txt]
```

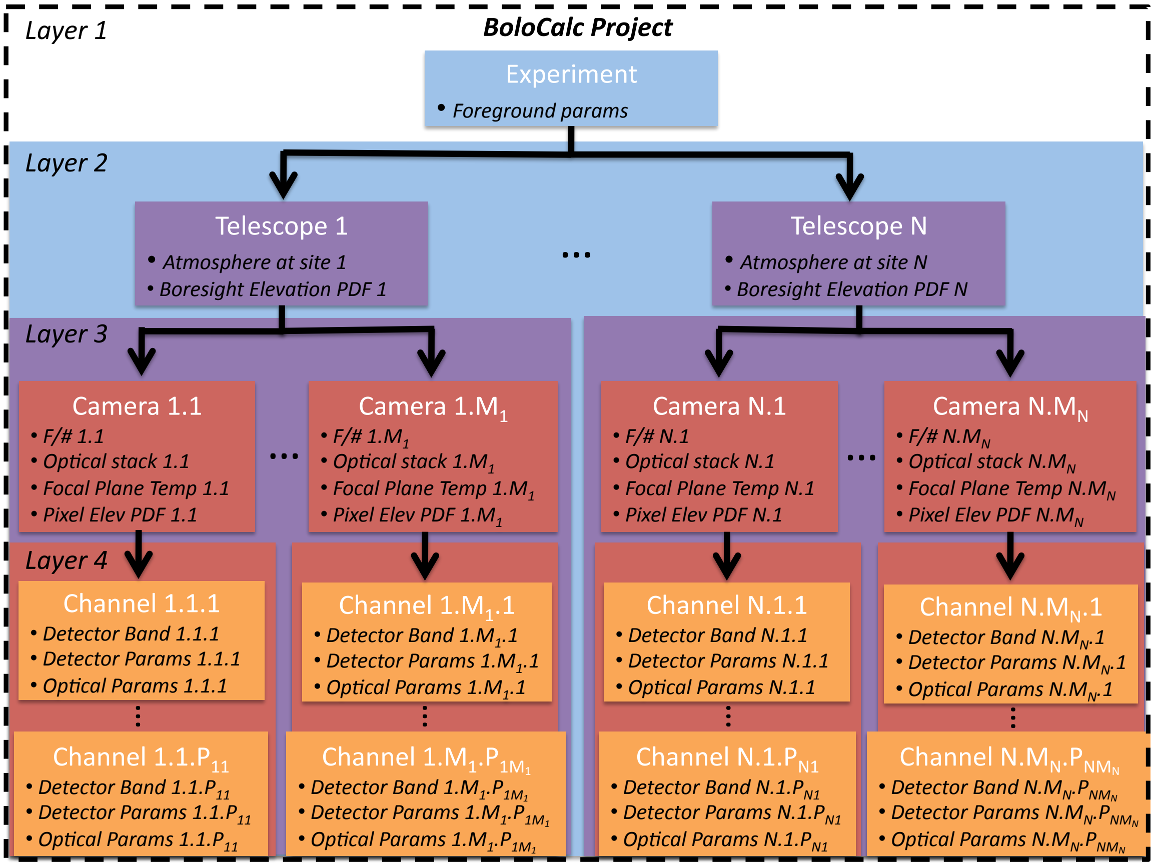


Figure 1: Generic layout of a BoloCalc project. There are four layers to a project, each with its own set of parameters: (1) experiments, (2) telescopes, (3) cameras, and (4) channels. There can be an arbitrary set of N telescopes in each experiment, an arbitrary set of M cameras in each telescope, and an arbitrary set of P channels in each camera. Each telescope inherits the parameters of its parent experiment, each camera inherits that of its telescope, and each channel that of its camera. The black bullet points highlight some important parameter definitions that occur within each layer.

```
    |    |    |    |    |    |-- ...
    |    |    |    |    |-- Detectors/
    |    |    |    |    |    |-- [*Channel1_Param.txt]
    |    |    |    |    |    |-- [*Channel2_Param.txt]
    |    |    |    |    |    |-- ...
    |    |    |    |-- Dist/
    |    |    |    |    |-- Optics/
    |    |    |    |    |    |-- [*Optic1_Param_BandID.txt]
    |    |    |    |    |    |-- [*Optic2_Param_BandID.txt]
    |    |    |    |    |    |-- ...
    |    |    |    |    |-- Detectors/
    |    |    |    |    |    |-- [*Channel1_Param.txt]
    |    |    |    |    |    |-- [*Channel2_Param.txt]
    |    |    |    |    |    |-- ...
    |    |    |    |    |-- ...
    |    |-- [*Camera_2]/
    |    |-- ...
```

```
    |-- [*Telescope_2]/
    |-- ...
```

Note that at least one telescope and camera need to be present within a given experiment. `Dist/` directories hold input probability distribution functions for foregrounds, telescope, or camera parameters, and `Bands/` directories hold reflection, absorption, spillover, or scattering spectra for optics, and transmission spectra for detectors. For more information about band definitions, see Sections 3.6.3 and 3.6.4. As stated in Section 3, the experiment has four layers, each which inherits the parameters of its parent layer:

1. Experiment (Layer 1)
    - Defines foreground parameters
    - Contains (multiple) telescope objects
2. Telescope (Layer 2)
    - Inherits foreground parameters from its parent experiment object
    - Defines telescope parameters (e.g. observation efficiency)
    - Defines site, atmospheric conditions, and scan strategy (elevation distribution)
    - Contains (multiple) camera objects
3. Camera (Layer 3)
    - Inherits foreground parameters from its parent experiment object
    - Inherits telescope parameters, atmospheric conditions, and scan strategy from its parent telescope object
    - Defines camera parameters (e.g. bath temperature)
    - Defines optics chain
    - Contains (multiple) channel objects
4. Channel (Layer 4)
    - Inherits foreground parameters from its parent experiment object
    - Inherits telescope parameters, atmospheric conditions, and scan strategy from its parent telescope object
    - Inherits camera parameters and optics chain from its parent camera object
    - Defines detector parameters and generates the detector array
    - Defines optics and detector bands for this frequency channel
    - **Calculates the power emitted, power attenuated, and power scattered/spilled for every optic and detector in this frequency channel**

The bold-faced operation is what's eventually passed to the sensitivity calculation, detailed in Section 6, making the channel objects the core of a BoloCalc experiment construction.

3.2 Defining Parameters

All parameters in BoloCalc are defined in one of two ways:

1. With a mean and standard deviation in the input parameter TXT file
2. With an input probability distribution function (PDF) file

Parameters defined within the input parameter text files `foregrounds.txt`, `telescope.txt`, `camera.txt`, and `channels.txt` have the format

```
    [mean] +/- [stdev]
```

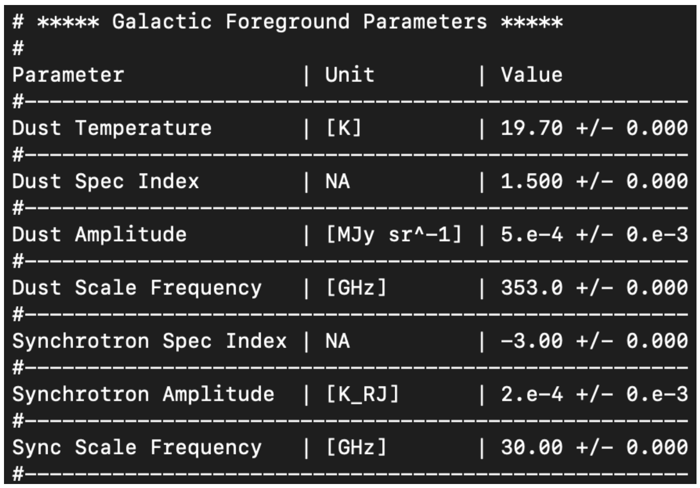
with the exception of the input parameter file `optics.txt`, which needs to define parameter values for every frequency channel, and therefore uses the format

```
    [band1_mean, band2_mean, ...] +/- [band1_stdev, band2_stdev, ...]
```

3.2.1 Input Parameter File Formatting

Below are snapshots of the input parameter files for `BoloCalc/Experiments/ExampleExperiment`, which we will use to discuss layout and formatting.

foregrounds.txt



```
# ***** Galactic Foreground Parameters *****
#
Parameter                | Unit        | Value
#---------------------------------------------------------
Dust Temperature         | [K]         | 19.70 +/- 0.000
#---------------------------------------------------------
Dust Spec Index          | NA          | 1.500 +/- 0.000
#---------------------------------------------------------
Dust Amplitude           | [MJy sr^-1] | 5.e-4 +/- 0.e-3
#---------------------------------------------------------
Dust Scale Frequency     | [GHz]       | 353.0 +/- 0.000
#---------------------------------------------------------
Synchrotron Spec Index   | NA          | -3.00 +/- 0.000
#---------------------------------------------------------
Synchrotron Amplitude    | [K_RJ]      | 2.e-4 +/- 0.e-3
#---------------------------------------------------------
Sync Scale Frequency     | [GHz]       | 30.00 +/- 0.000
#---------------------------------------------------------
```

Figure 2: Example `foregrounds.txt` file

The `foregrounds.txt` file, shown in Figure 2, is a one-dimensional table which defines the synchrotron and dust parameters. The *Parameter* column cannot be changed, the *Unit* column is not explicitly read by BoloCalc but shows the unit of the *Value* column, which is read by BoloCalc. All lines that begin with `#` are ignored. Formatting suggestion: to keep the table neat, we suggest keeping five characters for both the mean and standard deviation, for each parameter (e.g. 10.00, 5.e-4, 3.000).

telescope.txt
The `telescope.txt` file, shown in Figure 3, like the foregrounds file, is a one-dimensional table which defines the telescope's sky and observation parameters. The *Parameter* column cannot be changed, the *Unit* column is not explicitly read by BoloCalc but shows the unit of the *Value* column, which is read by BoloCalc. All lines that begin with `#` are ignored. Formatting suggestion: to keep the table neat, we suggest keeping five characters for both the mean and standard deviation, for each parameter (e.g. 10.00, 5.e-4, 3.000).

camera.txt
The `camera.txt` file, shown in Figure 4, like the foregrounds and telescope files, is a one-dimensional table which defines the camera's parameters. The *Parameter* column cannot be changed, the *Unit* column is not explicitly read by BoloCalc but shows the unit of the *Value* column, which is read by BoloCalc. All lines that begin with `#` are ignored. Formatting suggestion: to keep the table neat, we suggest keeping five characters for both the mean and standard deviation, for each parameter (e.g. 10.00, 5.e-4, 3.000).

```
# ***** Telescope Parameters *****
#
Parameter                | Unit  | Value
#----------------------------------------------
Site                     | NA    | Atacama
#----------------------------------------------
Elevation                | [deg] | 50.00 +/- 0.000
#----------------------------------------------
PWV                      | [mm]  | 2.000 +/- 0.500
#----------------------------------------------
Observation Time         | [yr]  | 5.000 +/- 0.000
#----------------------------------------------
Sky Fraction             | NA    | 0.200 +/- 0.000
#----------------------------------------------
Observation Efficiency   | NA    | 0.800 +/- 0.000
#----------------------------------------------
NET Margin               | NA    | 1.000 +/- 0.000
#----------------------------------------------
```

Figure 3: Example `telescope.txt` file

```
# ***** Camera Parameters *****
#
Parameter           | Unit  | Value
#-----------------------------------------
Boresight Elevation | [deg] | 0.000 +/- 0.000
#-----------------------------------------
Optical Coupling    | NA    | 1.000 +/- 0.000
#-----------------------------------------
F Number            | NA    | 2.500 +/- 0.000
#-----------------------------------------
Bath Temp           | [K]   | 0.100 +/- 0.000
#-----------------------------------------
```

Figure 4: Example `camera.txt` file

```
# ***** Detector Channels *****
#
Band ID | Pixel ID | Band Center     | Fractional BW   | Pixel Size       | Num Det per Wafer | Num Waf per OT | Num OT | Waist Factor    | Det Eff         | Psat            | P$
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------------$
#       | NA       | [GHz]           | NA              | [mm]             | NA                | NA             | NA     | NA              | NA              | [pW]            | N$
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------------$
1       | 1        | 90.00 +/- 0.000 | 0.350 +/- 0.000 | 6.800 +/- 0.000  | 542               | 1              | 3      | 3.000 +/- 0.000 | 0.700 +/- 0.000 | NA              | 3$
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------------$
2       | 1        | 150.0 +/- 0.000 | 0.300 +/- 0.000 | 6.800 +/- 0.000  | 542               | 1              | 3      | 3.000 +/- 0.000 | 0.800 +/- 0.000 | NA              | 3$
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------------$
```

Figure 5: Example `channels.txt` file. The image truncates the file on the right side.

channels.txt

The `channels.txt` file, shown in Figure 5 is a two-dimensional, wide-form table which defines the telescope detector and readout parameters for each of its frequency channels. Along the vertical axis are the frequency channels, identified by their **Band ID**. Along the horizontal axis are the channel parameters. All lines that begin with `#` are ignored.

The first row is the parameter names and cannot be changed. The second row is the parameter units, which are not explicitly read by BoloCalc because of the comment mark, and the subsequent rows are the parameter values for each channel. Formatting suggestions: this table is long and is a headache to handle when text wrapping is enabled. Therefore, we encourage turning off text wrapping in your text editor[1] when editing `channels.txt`. To keep the table neat, we suggest keeping five characters for both the mean and standard deviation, for each parameter and frequency channel (e.g. 10.00, 5.e-4, 3.000).

optics.txt

```
# ***** Optical Chain *****
#
Element   | Temperature    | Absorption                         | Reflection                         | Thickness     | Index         | Loss Tangent  | Conductivity  | Surfac$
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
#         | [K]            | NA                                 | NA                                 | [mm]          | NA            | [e-4]         | [e6 S/m]      | [um RM$
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
Primary   | 273.0 +/- 0.000 | [0.002,0.005] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
Mirror    | 273.0 +/- 0.000 | [0.002,0.005] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
Window    | 265.0 +/- 0.000 | [0.005,0.010] +/- [0.000,0.000] | [0.010,0.010] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
IRShader1 | 291.0 +/- 0.000 | [0.001,0.001] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
IRShader2 | 245.0 +/- 0.000 | [0.001,0.001] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
IRShader3 | 200.0 +/- 0.000 | [0.001,0.001] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
AbsFilter | 80.00 +/- 0.000 | [0.010,0.010] +/- [0.000,0.000] | [0.020,0.020] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
LowPass1  | 40.00 +/- 0.000 | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
Lens1     | 4.000 +/- 0.000 | [0.005,0.010] +/- [0.000,0.000] | [0.005,0.005] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
LowPass2  | 4.000 +/- 0.000 | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
Lens2     | 2.000 +/- 0.000 | [0.005,0.010] +/- [0.000,0.000] | [0.005,0.005] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
Aperture  | 2.000 +/- 0.000 | NA                              | NA                              | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
LowPass3  | 2.000 +/- 0.000 | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
Lens3     | 2.000 +/- 0.000 | [0.005,0.010] +/- [0.000,0.000] | [0.005,0.005] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
LowPass4  | 1.500 +/- 0.000 | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] | NA            | NA            | NA            | NA            | NA     $
#----------------------------------------------------------------------------------------------------------------------------------------------------------------------$
```

Figure 6: Example `optics.txt` file

The `optics.txt` file, shown in Figure 6 is a three-dimensional, wide-form table which defines the optical parameters for each optic and frequency channel. Along the vertical axis are the optics, identified by their **Element** name. Along the horizontal axis are the optical parameters. Along the depth axis are the values for each frequency channel, indexed by the row order in which the frequency channels are defined in `channels.txt`. All lines that begin with `#` are ignored.

The first row is the parameter names and cannot be changed. The second row is the parameter units, which are not explicitly read by BoloCalc because of the comment mark, and the subsequent rows are the parameter values. The parameters are defined according to the `[mean_list] +/- [std_list]` structure discussed in Section 3.2. Formatting suggestions: this table is long and is a headache to handle when text wrapping is enabled. Therefore, we encourage turning off text wrapping in your text editor[1] when editing `optics.txt`. To keep the table neat, I suggest keeping five characters for both the mean and standard deviation, for each parameter and frequency channel (e.g. 10.00, 5.e-4, 3.000).

3.2.2 Parameter Probability Distribution Functions

Defining input parameters in these input parameter files assumes a Gaussian distribution, which is not always a good approximation. In these cases, one can define an input probability distribution function (PDF) in the experiment, telescope, or camera's `Dist/` directory. The PDF file can be either a TXT file (whitespace delimited) or a CSV file (comma delimited) and has two columns

- Column 1 = value
- Column 2 = probability density

Note that if the sum of the probability density values do not add up to 1, then BoloCalc will scale the passed probabilities such that they do.

In order to flag the parameter as having an input PDF, its value in the input parameter file needs to be "PDF," and the PDF file name in `Dist/` needs to include the parameter name. As an

[1] I use emacs and therefore added `(set-default 'truncate-lines t)` to my `.emacs` file

example, Figure 7 shows the `telescope.txt` input parameter file, which instructs BoloCalc to load `Dist/pwv.txt`, the PWV PDF file, also shown in Figure 7.

```
# ***** Telescope Parameters *****
#
Parameter                | Unit  | Value
#-------------------------------------------------
Site                     | NA    | Atacama
#-------------------------------------------------
Elevation                | [deg] | 50.00 +/- 0.000
#-------------------------------------------------
#PWV                     | [mm]  | 2.000 +/- 0.500
PWV                      | [mm]  | PDF
#-------------------------------------------------
Observation Time         | [yr]  | 5.000 +/- 0.000
#-------------------------------------------------
Sky Fraction             | NA    | 0.200 +/- 0.000
#-------------------------------------------------
Observation Efficiency | NA    | 0.800 +/- 0.000
#-------------------------------------------------
NET Margin               | NA    | 1.000 +/- 0.000
#-------------------------------------------------
```

(a) `config/telescope.txt`

```
0.158375     0.0248262
0.25512      0.0387289
0.35187      0.0585898
0.44862      0.0711684
0.54537      0.0736511
0.64212      0.0670307
0.73887      0.0574313
0.83562      0.0513075
0.93237      0.0478318
1.02912      0.0484938
1.12587      0.0450182
1.22262      0.0326051
1.31937      0.0274743
1.41612      0.0261502
1.51287      0.0241641
1.60962      0.0195299
1.70637      0.0173783
1.80312      0.0163853
1.89987      0.0148957
1.99662      0.0168818
2.09337      0.0165508
2.19012      0.0135716
2.28687      0.0120820
2.38362      0.0095994
2.48037      0.0086064
2.57712      0.0100959
2.67387      0.0105925
2.77062      0.0089374
2.86737      0.0086064
2.96412      0.0081099
```

(b) `config/Dist/pwv.txt`

Figure 7: Example `config/telescope.txt` and `config/Dist/pwv.txt` file to illustrate how a PDF might be implemented for the parameter "PWV." Note that the text for the PWV PDF file is truncated a the bottom side.

In the case of `channels.txt` and `optics.txt`, which are two- and three-dimensional tables, the PDF file must be identified by not only the parameter name but also by the frequency channel Band ID, and in the case of optics parameters, also by the optical element.

PDF files for `channels.txt` are stored in `[Camera]/config/Dist/Detectors/` and PDF files for `optics.txt` are stored in `[Camera]/config/Dist/Optics/`. The required file formats are

- `channels.txt` PDF: `paramName_bandID.txt` or `paramName_bandID.csv`
- `optics.txt` PDF: `opticName_paramName_bandID.txt` or `opticName_paramName_bandID.csv`

where `bandID` is defined by the parameter **Band ID** in `channels.txt`.

For example, if passing a PDF file for parameter **SQUID NEI** for **Band ID** "2," the file name might be `config/Dist/Detectors/SquidNEI_2.txt` or `config/Dist/Detectors/squidNEI_2.csv`. In a similar manner, if passing a PDF file for parameter "Scatter Frac" for optic "Med Filt" and band ID "1," the PDF filename might be be `config/Dist/Optics/medFilt_scatterFrac_1.txt` or `config/Dist/Optics/MedFilt_ScatterFrac_1.csv`. Formatting note: note that all case comparisons are case insensitive, but we recommend using camel case (e.g. camelCase). File names cannot have spaces or underscores.

Figure 8 shows the `channels.txt` file when passing a PDF for **Det Eff** for **Band ID** "1," and Figure 9 shows the `optics.txt` file when passing a PDF for **Absorption** for optic "Lens1" and **Band ID** "2."

```
# ***** Detector Channels *****
#
Band ID | Pixel ID | Band Center      | Fractional BW     | Pixel Size        | Num Det per Wafer | Num Waf per OT | Num OT | Waist Factor      | Det Eff           |
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------
#       | NA       | [GHz]            | NA                | [mm]              | NA                | NA             | NA     | NA                | NA                |
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------
1       | 1        | 90.00 +/- 0.000  | 0.350 +/- 0.000   | 6.800 +/- 0.000   | 542               | 1              | 3      | 3.000 +/- 0.000   | PDF               |
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------
2       | 1        | 150.0 +/- 0.000  | 0.300 +/- 0.000   | 6.800 +/- 0.000   | 542               | 1              | 3      | 3.000 +/- 0.000   | 0.800 +/- 0.000   |
#--------------------------------------------------------------------------------------------------------------------------------------------------------------------
```

(a) `channels.txt`

```
0.4    0.10
0.5    0.15
0.6    0.30
0.7    0.35
0.8    0.10
```

(b) `detEff_1.txt`

Figure 8: Example `config/channels.txt` and `config/Dist/Detectors/detEff_1.txt` file to illustrate how a PDF might be implemented for parameter **Det Eff** for frequency channel with **Band ID** "1."

3.3 Experiment Parameters

An experiment is composed of an arbitrary set of telescopes, which are assumed to see the same celestial loading, including power from the CMB and galactic foregrounds.

3.3.1 `foregrounds.txt`

The below list overviews the parameters defined in `[Exp_name]/[Exp_Design]/config/foregrounds.txt`, the file that defines the galactic foreground parameters. **Note that these parameters have been updated as of BoloCalc v0.10.0**.

- **Dust Temperature**
  - Definition: T_d in Equation 48
  - Description: Modified dust blackbody temperature
  - Units: [K]
  - Allowed Values: Floating point value between $[0.000, +\infty)$
  - Suggested Default: 19.70 ± 0.000 K
- **Dust Spec Index**
  - Definition: n_d in Equation 48
  - Description: Spectral index of the modified blackbody dust spectrum
  - Units: [NA]
  - Allowed Values: Floating point value between $(-\infty, +\infty)$
  - Suggested Default: 1.500 ± 0.000
- **Dust Amplitude**
  - Definition: A_d in Equation 48
  - Description: Amplitude of the modified blackbody dust spectrum at the scale frequency

```
# ***** Optical Chain *****
#
Element   | Temperature      | Absorption                        | Reflection                        |
#---------------------------------------------------------------------------------------------------
#         | [K]              | NA                                | NA                                |
#---------------------------------------------------------------------------------------------------
Primary   | 273.0 +/- 0.000  | [0.002,0.005] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
Mirror    | 273.0 +/- 0.000  | [0.002,0.005] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
Window    | 265.0 +/- 0.000  | [0.005,0.010] +/- [0.000,0.000] | [0.010,0.010] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
IRShader1 | 291.0 +/- 0.000  | [0.001,0.001] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
IRShader2 | 245.0 +/- 0.000  | [0.001,0.001] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
IRShader3 | 200.0 +/- 0.000  | [0.001,0.001] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
AbsFilter | 80.00 +/- 0.000  | [0.010,0.010] +/- [0.000,0.000] | [0.020,0.020] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
LowPass1  | 40.00 +/- 0.000  | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
Lens1     | 4.000 +/- 0.000  | [0.005, PDF ] +/- [0.000,0.000] | [0.005,0.005] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
LowPass2  | 4.000 +/- 0.000  | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
Lens2     | 2.000 +/- 0.000  | [0.005,0.010] +/- [0.000,0.000] | [0.005,0.005] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
Aperture  | 2.000 +/- 0.000  | NA                                | NA                                |
#---------------------------------------------------------------------------------------------------
LowPass3  | 2.000 +/- 0.000  | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
Lens3     | 2.000 +/- 0.000  | [0.005,0.010] +/- [0.000,0.000] | [0.005,0.005] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
LowPass4  | 1.500 +/- 0.000  | [0.010,0.010] +/- [0.000,0.000] | [0.050,0.050] +/- [0.000,0.000] |
#---------------------------------------------------------------------------------------------------
```

(a) `optics.txt`

```
0.08    0.05
0.07    0.05
0.06    0.10
0.07    0.15
0.06    0.20
0.05    0.35
0.04    0.10
```

(b) `lens1_absorption_2.txt`

Figure 9: Example `config/optics.txt` and `config/Dist/Optics/lens1_absorption_2.txt` file to illustrate how a PDF might be implemented for optic "Lens1" for parameter **Absorption** for the second frequency channel defined in `channel.txt`, which in this case has **Band ID** "2."

- Units: [MJy sr^{-1}]
- Allowed Values: Floating point value between $[0.000, +\infty)$

- Suggested Default: $1 \times 10^{-2} \pm 0.000$

- **Dust Scale Frequency**
  - Definition: ν_d in Equation 48
  - Description: Scale (or pivot) frequency for the modified blackbody dust spectrum
  - Unit: [GHz]
  - Allowed Values: Floating point value between $(0.000, +\infty)$
  - Suggested Default: 353.0 ± 0.000
- **Synchrotron Spec Index**
  - Definition: n_s in Equation 47
  - Description: Spectral index of the synchrotron power law spectrum
  - Unit: [NA]
  - Allowed Values: Floating point value between $(-\infty, +\infty)$
  - Suggested Default: -3.000 ± 0.000
- **Synchrotron Amplitude**
  - Definition: T_s in Equation 47
  - Description: Amplitude of the synchrotron power law spectrum at the scale frequency in brightness temperature
  - Unit: [$\mathrm{K_{RJ}}$]
  - Allowed Values: Floating point value between $(0.000, +\infty)$
  - Suggested Default: $1.000 \times 10^{-3} \pm 0.000$
- **Sync Scale Frequency**
  - Definition: ν_s in Equation 47
  - Description: Scale (or pivot) frequency for the synchrotron power law spectrum
  - Unit: [GHz]
  - Allowed Values: Floating point value between $(0.000, +\infty)$
  - Suggested Default: 30.00 ± 0.000

3.4 Telescope Parameters

A telescope is composed of an arbitrary set of cameras, which are assumed to operate at the same site, with the same scan strategy, and with the same telescope parameters.

3.4.1 `telescope.txt`

The below list overviews the parameters defined in `[Tel_name]`. These parameters are used to define the site, atmospheric profile, and patch coverage for the telescope.

- **Sky Temperature**
  - Description: Equivalent brightness temperature (T in Eq. 12) of the sky through which the telescope observes, including the atmosphere, CMB, and galactic foregrounds.
  - Units: [K]
  - Allowed Values: Floating point value between $(0, \infty)$
  - Suggested Default: "NA"
  - If this parameter is not "NA," then the parameters **Site**, **Elevation**, and **PWV** are ignored
- **Site**
  - Description: Site at which the telescope observes, which defines the atmospheric profile
  - Allowed Values: "Atacama," "Pole," "McMurdo" (balloon), "Space," "CUST" (custom atmosphere file), or a float that represents the sky's effective brightness temperature. See section 6.14.3 for more details about handling of the atmosphere.
  - If the parameter **Sky Temperature** is not "NA," then this parameter is ignored.
- **Elevation**

- Definition: θ_{tel} in Equation 56
  - Description: Telescope boresight elevation
  - Units: [deg] above the horizon
  - Allowed Values: Floating point value with step size 1.0 between $[20, 90]$
  - Suggested Default: 50.00 ± 0.000
  - If the parameter **Site** is "CUST" of if the parameter **Sky Temperature** is not "NA," then this parameter is ignored.
- **PWV**
  - Description: Precipitable water vapor of the atmosphere through which the telescope observes in millimeters
  - Units: [mm]
  - Allowed Values: Floating point value with step size 0.1 between $[0.0, 8.0]$
  - Suggested Default: 1.000 ± 0.000
  - If the parameter **Site** is "CUST" of if the parameter **Sky Temperature** is not "NA," then this parameter is ignored.
- **Observation Time**
  - Definition: t_{obs} in Equation 46
  - Description: For how long the telescope will operate in years. Note that observation efficiency is a separate parameter
  - Units: [yr]
  - Allowed Values: Floating point value between $(0.000, +\infty)$
  - Suggested Default: 3.000 ± 0.000
- **Sky Fraction**
  - Definition: f_{sky} in Equation 46
  - Description: (Effective) fraction of the full sky observed by this telescope
  - Units: [NA]
  - Allowed Values: Floating point value between $(0.000, 1.0]$
  - Suggested Default: 0.700 ± 0.000
- **Observation Efficiency**
  - Definition: η_{obs} in Equation 46
  - Description: Fraction of the total operation time during which the telescope is making science observations
  - Units: [NA]
  - Allowed Values: Floating point value between $(0.000, 1.000]$
  - Suggested Default: 0.800 ± 0.000
- **NET Margin**
  - Definition: M in Equation 37
  - Description: Agnostic factor which multiplies all NETs for this telescope. Useful for incorporating contingencies.
  - Unis: [NA]
  - Allowed Values: Floating point value between $(0.000, +\infty)$
  - Suggested Default: 1.000 ± 0.000

3.5 Camera Parameters

A camera is composed of an arbitrary set of channels, which are assumed to share the same focal plane and have the same optical chain.

3.5.1 `camera.txt`

The below list overviews the parameters defined in `[Cam_name]`. These parameters are used to define focal plane tempearture, focal plane f-number, and focal plane optical coupling.

- **Boresight Elevation**
  - Definition: θ_{cam} in Equation 56
  - Description: Camera boresight elevation with respect to the telescope boresight elevation
  - Units: [deg] above telescope boresight
  - Allowed Values: Floating point value between $[-40.00, +40.00]$
  - Suggested Default: 0.000 ± 0.000
- **Optical Coupling**
  - Definition: ξ in Equations 39 and 38
  - Description: An overall coupling factor for all detectors pixels in this camera
  - Units: [NA]
  - Allowed Values: Floating point value between $(0.000, 1.000]$
  - Suggested Default: 1.000 ± 0.000
- **F Number**
  - Definition: F in Equation 49
  - Description: (Effective) F-number at the focal plane
  - Units: [NA]
  - Allowed Values: Floating point value between $(0.000, +\infty)$
  - Suggested Default: 2.000 ± 0.000
- **Bath Temp**
  - Definition: T_b in Equations 20, 23, and 22
  - Description: Focal plane temperature
  - Units: [K]
  - Allowed Values: Floating point value between $(0.000, +\infty)$
  - Suggested Default: 0.100 ± 0.000

3.5.2 `channels.txt`

Channel objects inherit the optical stack of its parent camera and contains an array of detector objects. `channels.txt` defines the detector parameters for all the channels in the camera, and the below list overviews those parameters.

- **Band ID**
  - Description: The identification for this frequency channel
  - Suggested Default: 1
  - Every channel within a camera needs to have a unique Band ID. **We strongly recommend that you index your bands in ascending order in `channels.txt`, using the Band ID parameter**.
- **Pixel ID**
  - Description: Pixel identification for this frequency channel
  - Suggested Default: 1
  - **As of BoloCalc v0.10.0, this parameter no longer has meaning. That said, it can be useful for organizing which detector channels share a detector pixel, in the case of multichroic pixels**. You can delete the parameter from your `channels.txt` file if you like.
- **Band Center**
  - Definition: ν_c in Equation 1
  - Description: Central frequency for this frequency channel
  - Units: [GHz]
  - Allowed Values: A floating point value between $(0.000, +\infty)$

  - Suggested Default: 150.0 ± 0.000
- **Fractional BW**
  - Definition: f_{BW} in Equation 1
  - Description: Fractional arithmetic bandwidth for this frequency channel
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, 2.0]$
  - Suggested Default: 0.300 ± 0.000
- **Pixel Size**
  - Definition: D_{pix} in Equation 49
  - Description: Detector pixel size (or equivalently, pixel spacing) for this frequency channel
  - Units: [mm]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: 6.800 ± 0.000
- **Num Det per Wafer**
  - Definition: $N_{det/waf}$ in Equation 60. Note that detector yield is an independent parameter.
  - Description: Number of detectors per detector wafer for this frequency channel
  - Units: [NA]
  - Allowed Values: An integer value between $[0, +\infty)$
  - Suggested Default: 542
  - Note that this parameter does not accept a spread
- **Num Waf per OT**
  - Definition: $N_{wav/OT}$ in Equation 60
  - Description: Number of wafers per optics tube (camera) for this frequency channel
  - Units: [NA]
  - Allowed Values: An integer value between $[0, +\infty)$
  - Suggested Default: 7
  - Note that this parameter does not accept a spread
- **Num OT**
  - Definition: N_{OT} in Equation 60
  - Description: Number of optics tubes (cameras) which contain detector wafers that have detectors of this frequency channel
  - Units: [NA]
  - Allowed Values: An integer value between $[0, +\infty)$
  - Suggested Default: 1
  - Note that this parameter does not accept a spread
- **Waist Factor**
  - Definition: w_f in Equation 49
  - Description: Ratio of pixel diameter to pixel Gaussian beam waist
  - Units: [NA]
  - Allowed Values: A floating point value between $[2.0, +\infty)$
  - Suggested Default: 3.000 ± 0.000
- **Det Eff**
  - Definition: η_{det} in Equations 14 and 5.
  - Description: The band-averaged detector optical efficiency
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, 1.000]$
  - Suggested Default: 0.700 ± 0.000
- **Psat**
  - Definition: P_{sat} in Equations 23 and 31
  - Description: The detector saturation power for this frequency channel

- Units: [pW/$\sqrt{\text{Hz}}$]
- Allowed Values: A floating point value between $(0.000, +\infty)$
- Suggested Default: "NA"
- If "NA," then use **Psat Factor** f_{psat} Equation 57 to calculate P_{sat}

- **Psat Factor**
  - Definition: f_{psat} in Equation 57
  - Description: The ratio of saturation power to optical power for detectors within this frequency channel
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: 3.000 ± 0.000
  - If parameter **Psat** is not "NA," this this parameter is ignored
- **Carrier Index**
  - Definition: n in Equations 23 and 22
  - Description: Thermal carrier index for bolometer conductivity to the bath
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: 2.700 ± 0.000
  - If parameter **G** is not "NA" and **Flink** is not "NA," this parameter is ignored
- **Tc**
  - Definition: T_{oper} in Equations 23, 22, and 20
  - Description: Bolometer operating temperature
  - Units: [K]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: 0.165 ± 0.000
  - If "NA," then use parameter **Tc Fraction** to calculate T_{oper}
- **Tc Fraction**
  - Definition: f_{oper} in Equation 21
  - Description: The ratio of bolometer operating temperature to bath temperature
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If parameter T_c is not "NA," then this parameter is ignored
- **Flink**
  - Definition: F_{link} in Equations 22 and 20
  - Description: Numerical factor that depends on the thermal carrier index
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If "NA," **Flink** is calculated using **Carroer Index** and Equation 22
  - This parameter was added in BoloCalc v0.9.0. If the parameter is absent from `channels.txt`, the assumed value is "NA"
- **G**
  - Definition: G in Equations 23 and 20
  - Description: Thermal conduction from the bolometer to the bath
  - Units: [pW / K]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If "NA," **G** is calculated using the parameters **Carrier Index**, **Tc**, **Bath Temp**, and **Psat** and Equation 23
  - This parameter was added in BoloCalc v0.9.0. If the parameter is absent from

channels.txt, the assumed value is "NA"

- **Yield**
  - Definition: Y in Equation 40
  - Description: Detector yield
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, 1.000]$
  - Suggested Default: 0.700 ± 0.000
- **SQUID NEI**
  - Definition: **NEI** in Equation 31
  - Description: SQUID amplifier noise
  - Units: $\mathrm{pA}/\sqrt{\mathrm{Hz}}$
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If "NA," then use parameter **Read Noise Frac** and Equation 32 to calculate $\mathrm{NEP}_{\mathrm{read}}$
- **Bolo Resistance**
  - Definition: R_{bolo} in Equation 31
  - Description: Bolometer operating resistance
  - Units: $[\Omega]$
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If "NA," then use parameter **Read Noise Frac** and Equation 59 to calculate $\mathrm{NEP}_{\mathrm{read}}$
- **Resp Factor**
  - Definition: S_{fact} in Equations 31, 25, and 26
  - Description: Responsively factor, which is set by bolometer operating loop gain
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: 1.000 ± 0.000
  - If "NA," then assume a value of 1
  - This parameter was added in BoloCalc v0.10.0. If the parameter is absent from channels.txt, the assumed value is "NA"
- **Read Noise Frac**
  - Definition: Δ_{read} in Equation 59
  - Description: Fraction of the total detector NEP that is due to readout noise
  - Units: [NA]
  - Allowed Values: A floating point value between $[0.000, +\infty)$
  - Suggested Default: 0.100 ± 0.000
  - If parameters **SQUID NEI** and **Bolo Resistance** are not "NA," then this parameter is ignored

3.5.3 optics.txt

Optics objects are used to construct the optical chain within the defined camera. optics.txt defines the parameters for each optic and each channel, and below are the parameters.

- **Element**
  - Description: Name of the optical element
  - Allowed Values: Any string, conventionally without spaces and in camel case format
  - RESERVED NAMES: "Primary" and "Mirror" trigger mirror calculations in Equation 54, and "Aperture," "Stop," and "Lyot," trigger aperture stop calculations in Equation 49. Use these names with intentionality
  - NOTE: Each optical element name must be unique
- **Temperature**

- Definition: T_{i} in Equation 11
- Description: Temperature of the optical element
- Units: [K]
- Allowed Values: A floating point value between $(0.000, +\infty)$

- **Absorption**
  - Definition: ϵ_{i} in Equation 11
  - Description: Emissivity/absorptivity of the optical element
  - Units: [NA]
  - Allowed Values: A floating point value between $[0.000, 1.000]$
  - If "NA"...
    - If **Element** is "Mirror" or "Primary," then use **Conductivity** and Equation 53 to calculate Ohmic losses in the conductor
    - If **Element** is not "Mirror" or "Primary," then use **Thickness**, **Index**, **Loss Tangent**, and Equation 52 to calculate loss in the dielectric
- **Reflection**
  - Definition: r_{i} in Equations 9, 14, and 55
  - Description: Reflectivity of the optical element
  - Units: [NA]
  - Allowed Values: A floating point value between $[0.000, 1.000]$
- **Thickness**
  - Definition: t_{i} in Equation 52
  - Description: Thickness of the dielectric optical element
  - Units: [mm]
  - Allowed Values: A floating point value between $[0.000, 1.000]$
  - Suggested Default: "NA"
  - If **Element** is "Mirror" or "Primary," then this parameter is ignored
  - If **Absorption** is not "NA," then this parameter is ignored
- **Index**
  - Definition: n_{i} in Equation 52
  - Description: Index of refraction of the dielectric optical element
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If **Element** is "Mirror" or "Primary," then this parameter is ignored
  - If **Absorption** is not "NA," then this parameter is ignored
- **Loss Tangent**
  - Definition: $\tan\delta_{\mathrm{i}}$ in Equation 52
  - Description: Loss tangent of the dielectric optical element
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If **Element** is "Mirror" or "Primary," then this parameter is ignored
  - If **Absorption** is not "NA," then this parameter is ignored
- **Conductivity**
  - Definition: σ_c in Equation 53
  - Description: Conductivity of the conductive optical element
  - Unit: [10^6 S/m]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: "NA"
  - If **Element** is not "Mirror" or "Primary," then this parameter is ignored
  - If **Absorption** is not "NA," then this parameter is ignored

- **Surface Rough**
  - Definition: σ_r in Equation 54
  - Description: Surface roughness of the optical element
  - Units: [μm RMS]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: “NA”
  - If “NA” and **Scatter Frac** “NA,” then scattering set to zero.
- **Scatter Frac**
  - Definition: δ_i in Equation 11
  - Description: Scattering loss at the optical element
  - Units: [NA]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: “NA”
  - If “NA,” then use **Surface Rough** and Equation 54 to calculate scattering loss
- **Scatter Temp**
  - Definition: $T_{\delta;i}$ in Equation 11
  - Description: Effective temperature to which power is scattered from this optical element
  - Units: [K]
  - Allowed Values: A floating point value between $(0.000, +\infty)$
  - Suggested Default: “NA”
  - If “NA,” then assume the same value as **Temperature**
- **Spillover**
  - Definition: β_i in Equation 11
  - Description: Spillover loss at the optical element
  - Units: [NA]
  - Allowed Values: A floating point value between $[0.000, 1.000]$
  - Suggested Default: “NA”
  - If “NA,” then spillover set to zero
- **Spillover Temp**
  - Definition: $T_{\beta;i}$ in Equation 11
  - Description: Effective temperature to which power is spilled over this optical element
  - Units: [K]
  - Allowed Values: A floating point value between $[0.000, +\infty)$
  - Suggested Default: “NA”
  - If “NA,” then assume the same value as **Temperature**

3.5.4 `elevation.txt`

Given a finite field of view within a camera, the pixels within that camera will have some distribution of elevations on the sky with respect to the camera boresight elevation. One can define such a distribution in one of two ways: either define a Gaussian spread on the parameter **Boresight Elevation** in the file `camera.txt` or define an explicit distribution in the file `[Camera]/config/elevation.txt`. The columns of the `elevation.txt` file are given as

- *Column 1*: Pixel elevation with respect to camera boresight
- *Column 2*: Fraction of detectors at this pixel elevation with respect to camera boresight

Figure 10 shows an example pixel elevation distribution file.

BoloCalc uses Equation 56 to assign elevation values to each simulated detector pixel, and given the number of detector realizations defined in `simInputs.txt`, the distribution in `elevation.txt` is sampled over as a probability distribution when generating Monte Carlo detector realizations.

```
-5      0.1
-3      0.2
-1      0.2
1       0.2
3       0.2
5       0.1
```

Figure 10: An example `elevation.txt` file

3.6 Defining Bands

A "band" is defined as the frequency dependence of a parameter, or equivalently the frequency spectrum of a parameter. A band most commonly refers to optical parameters, such as transmission vs frequency, and in BoloCalc, band functionalities are restricted to a few optical inputs:

- Channel **Det Eff**
- Optic **Absorption**
- Optic **Reflection**
- Optic **Spillover**
- Optic **Scattering**

Additionally, BoloCalc supports three different band types:

- Top hat
- Trapezoid
- Custom

which we will discuss in the following subsections.

3.6.1 Top-Hat Band

Top-hat bands are the "de facto" band shape for BoloCalc detectors and optics. They are fully defined by three parameters: band center, percent/fractional bandwidth and a frequency-independent parameter b

$$B(\nu) = \begin{cases} 0 & \text{if } \nu < \nu_{\mathrm{lo}} \\ b & \text{if } \nu_{\mathrm{lo}} \leq \nu \leq \nu_{\mathrm{hi}} \\ 0 & \text{if } \nu > \nu_{\mathrm{hi}} \end{cases} \tag{1}$$

where b might be efficiency, reflection, absorption, spillover, or scattering. ν is frequency, ν_{lo} is the lower band edge, and ν_{hi} is the upper band edge. We define the band edge frequencies using the band center ν_{c} and fractional bandwidth f_{BW} as

$$\nu_{\mathrm{lo}} = \nu_{\mathrm{c}} - \frac{\nu_{\mathrm{c}}\, f_{\mathrm{BW}}}{2} \tag{2}$$

$$\nu_{\mathrm{hi}} = \nu_{\mathrm{c}} + \frac{\nu_{\mathrm{c}}\, f_{\mathrm{BW}}}{2} \tag{3}$$

3.6.2 Trapezoidal Band

Trapezoidal bands are constructed very similarly to top-hat bands, but instead of a frequency-independent parameter b to define the band's admittence, they use a frequency-dependent parameter $\alpha(\nu)$. The frequency-dependent parameter doesn't necessarily have to be linear—which makes "trapezoidal" a bit of a misnomer—but often is in the small-value limit. The trapezoidal band is defined as

$$B(\nu) = \begin{cases} 0 & \text{if } \nu < \nu_{\text{lo}} \\ \alpha(\nu) & \text{if } \nu_{\text{lo}} \leq \nu \leq \nu_{\text{hi}} \\ 0 & \text{if } \nu > \nu_{\text{hi}} \end{cases} \tag{4}$$

where $\alpha(\nu)$ is the frequency-dependent parameter, ν is frequency, and ν_{lo} is the lower band edge and ν_{hi} is the upper band edge, which are defined in Equations 2 and 3.

$\alpha(\nu)$ cannot be manipulated arbitrarily (see Section 3.6.3 for how to construct an arbitrary band) but is instead calculated using derived parameters for a few special cases:

- Absorption in a dielectric optic: $\alpha(\nu)$ defined by Equation 52
- Absorption in a conductive optic: $\alpha(\nu)$ defined by Equation 53
- Scattering due to surface roughness: $\alpha(\nu)$ defined by Equation 54

Note that a BoloCalc-derived trapezoidal band is not an option for detector transmission; it is only available for optical elements. For more information on how optics parameters can be used to construct a trapezoidal band, see the parameter definitions and descriptions in Section 3.5.3.

3.6.3 Custom Bands

As discussed in Sections 3.6.1 and 3.6.2, transmission bands for detectors, and absorption, reflection, spillover, and scattering bands or optics can be defined using BoloCalc parameters laid out in Section 3.2. However, often times you may want to use a measured or simulated band for a given detector or optic, and therefore BoloCalc accepts custom bands. The band file can be either a TXT file (whitespace delimited) or a CSV file (comma delimited) and has two or three columns

- Column 1 = frequency [GHz]
- Column 2 = quantity central value
- Column 3 = quantity standard deviation/error bar/spread (optional)

In order to flag an input parameter as having a custom-defined band, its value in the input parameter file needs to be "BAND" and the band file name in `Bands/` needs to include the parameter name. Band files for detectors are stored in the directory `[Cam_name]/config/Bands/Detectors/` and band files for optics are stored in the directory `[Cam_name]/config/Bands/Optics/`. Below are the required file formats for the detector and optics band names:

- `channels.txt` BAND: `camName_bandID.txt` or `camName_bandID.csv`
- `optics.txt` BAND: `opticName_paramName.txt` or `opticName_paramName.csv`

where `bandID` is defined by the parameter `[Band ID]` in `channels.txt`. Note that no spaces or underscores are allowed for `camName`, `opticName`, `bandID`, and `paramName`.

For example, if passing a band file for detector transmission for **Band ID** "1" within camera "MiddleCam," the file name might be `config/Band/Detectors/middleCam_1.csv`. In a similar manner, if passing a band file for parameter **Spillover** for optic "Aperture Lens," the file name might be `config/Band/Optics/spillover_apertureLens.txt`. Note that detector bands are defined for each channel within a camera, but optics bands are defined for all channels within the camera.

To demonstrate how to pass a detector band file, Figure 11 shows the `channels.txt` file when passing "BAND" to parameter **Band Center** for channel with **Band ID** "Band1." Similarly, Figure 12 shows the `optics.txt` file when passing "BAND" for optic "Window" and parameter **Reflection**.

```
# ***** Detector Channels *****
#
Band ID | Pixel ID | Band Center     | Fractional BW   | Pixel Size        | Num Det per Wafer | Num Waf per OT | Num OT | Waist Factor    | Det Eff         |
#-----------------------------------------------------------------------------------------------------------------------------------------------------------------
#       | NA       | [GHz]           | NA              | [mm]              | NA                | NA             | NA     | NA              | NA              |
#-----------------------------------------------------------------------------------------------------------------------------------------------------------------
Band1   | 1        | BAND  +/- 0.000 | 0.350 +/- 0.000 | 6.800 +/- 0.000   | 542               | 1              | 3      | 3.000 +/- 0.000 | PDF   +/- 0.000 |
#-----------------------------------------------------------------------------------------------------------------------------------------------------------------
2       | 1        | 150.0 +/- 0.000 | 0.300 +/- 0.000 | 6.800 +/- 0.000   | 542               | 1              | 3      | 3.000 +/- 0.000 | PDF   +/- 0.000 |
#-----------------------------------------------------------------------------------------------------------------------------------------------------------------
```

(a) `channels.txt`

```
70.     0.000   0.000
75.     0.500   0.000
80.     0.700   0.000
85.     0.650   0.000
90.     0.700   0.000
100.    0.800   0.000
105.    0.500   0.000
110.    0.000   0.000
115.    0.000   0.000
120.    0.000   0.000
```

(b) `Cam_Band1.txt`

Figure 11: Example `config/channels.txt` and `config/Bands/Detectors/Cam_Band1.txt` file to illustrate how a detector transmission band might be implemented for frequency channel with **Band ID** "Band1." The image of the `channels.txt` is truncated on the right side.

For more information regarding the handling of the band's central values and error bars in a Monte Carlo simulation, see Section 4.3.1.

3.6.4 Custom Detector Bands

As discussed in Section 3.2, parameters are either defined using a central value and a spread `[Central_Value] +/- [Spread]`, a probability distribution, or, for some parameters only, a `BAND`. As discussed in Section 3.6.3, custom bands can be defined for detectors by passing the value "BAND" to the parameter **Band Center**. However, there are two additional capabilities that are available for detector bands.

If the parameter **Band Center** in `channels.txt` is defined to be `BAND +/- [SPREAD]`, then an uncertainty in the location of the passed custom band is assumed to be `+/- [SPREAD]` GHz, and when running multiple Monte Carlos realizations of detectors (see Section 4.3.1 for more details regarding Monte Carlo simulations), the spread is assumed to be the one-sigma uncertainty in the custom band position. Note that you cannot pass "PDF" as a parameter for band shift.

Also, when passing a custom band by setting **Band Center** in `channels.txt` to `BAND` or `BAND +/- [SPREAD]`, the band-integrated efficiency can also be set independently using the parameter **Det Eff** (see Section 3.5.2 for more details about defining detector parameters). When both a custom band and **Det Eff** are defined simultaneously, the band-integrated efficiency is set as

$$B(\nu) = \frac{\Delta\nu_{3\mathrm{dB}}\ \eta_{\mathrm{det}}}{\int_0^\infty B_{\mathrm{input}}(\nu)\mathrm{d}\nu} B_{\mathrm{input}}(\nu) \tag{5}$$

where $B_{\mathrm{input}}(\nu)$ is the input custom band, $\Delta\nu_{3\mathrm{dB}}$ is the bandwidth between the input band's -3 dB points, η_{det} is the parameter **Det Eff**, and $B(\nu)$ is the adjusted custom bandpass. Note that you

```
# ***** Optical Chain *****
#
Element   | Temperature      | Absorption                        | Reflection                        |
#-----------------------------------------------------------------------------------------------------
#         | [K]              | NA                                | NA                                |
#-----------------------------------------------------------------------------------------------------
Primary   | 273.0 +/- 0.000  | [0.002,0.005] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#-----------------------------------------------------------------------------------------------------
Mirror    | 273.0 +/- 0.000  | [0.002,0.005] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#-----------------------------------------------------------------------------------------------------
Window    | PDF   +/- 0.000  | [0.005,0.010] +/- [0.000,0.000] | BAND                              |
#-----------------------------------------------------------------------------------------------------
IRShader1 | 291.0 +/- 0.000  | [0.001,0.001] +/- [0.000,0.000] | [0.000,0.000] +/- [0.000,0.000] |
#-----------------------------------------------------------------------------------------------------
```

(a) `optics.txt`

```
70.,0.000,0.000
75.,0.500,0.010
80.,0.050,0.010
85.,0.020,0.010
90.,0.020,0.010
100.,0.020,0.010
105.,0.020,0.010
110.,0.020,0.010
115.,0.020,0.010
120.,0.020,0.010
130.,0.020,0.010
135.,0.020,0.010
140.,0.020,0.010
145.,0.020,0.010
150.,0.020,0.010
155.,0.020,0.010
160.,0.020,0.010
165.,0.020,0.010
170.,0.050,0.010
175.,0.500,0.010
180.,0.000,0.000
```

(b) `window_reflection.csv`

Figure 12: Example `config/optics.txt` and `config/Bands/Optics/window_reflection.txt` file to illustrate how a an optical band might be implemented for parameter **Reflection** and optical element "Window." The image of the `optics.txt` is truncated on the right and bottom sides.

can pass "PDF" as a parameter for **Band Eff** if wanting an arbitrary probability distribution for detector efficiency on a custom band. See Section 3.2.2 for more details about defining probability distributions.

4 Simulate a BoloCalc Experiment

BoloCalc has one executable `calcBolos.py` that uses a few configuration fils and several command-line arguments to define the parameters of the simulation. We discss the simulation inputs, as well as how to run the simulation, in this section.

4.1 Define Simulation Inputs

Simulation inputs are stored in the file `BoloCalc/config/simulationInputs.txt`, and an example of the file is shown in Figure 13. Each parameter is defined in the following list

- **Experiments**
  - Definition: N_{exp} in Equation 8
  - Description: Number of experiment realizations in the Monte Carlo simulation
  - Allowed Values: Integer value between $[1, \infty)$
  - Suggested Default: "1"
  - If set to "1," then take the median value for all experiment parameters. If greater than 1, then sample the experiment parameter distributions the set number of times
- **Observations**
  - Definition: N_{obs} in Equation 8
  - Description: Number of observation realizations in the Monte Carlo simulation
  - Allowed Values: Integer value between $[1, \infty)$
  - Suggested Default: "1"
  - If set to "1," then take the median value for all PWV + telescope elevation parameters. If greater than 1, then sample the PWV + telescope elevation parameters distributions the set number of times
- **Detectors**
  - Definition: N_{dets} in Equation 8
  - Description: Number of detector realizations in the Monte Carlo simulation
  - Allowed Values: Integer value between $[1, \infty)$
  - Suggested Default: 1
  - If set to "1," then take the median value for all detector parameters. If greater than 1, then sample the detector parameters distributions the set number of times
- **Resolution**
  - Description: Frequency resolution used for the simulation
  - Allowed Values: Integer value between $(0, \infty)$
  - Units: [GHz]
  - Note that reducing the resolution—by increasing this parameter's value—is an efficient way to speed up BoloCalc calculations
  - Suggested Default: "0.5"
- **Foregrounds**
  - Description: A boolean value to specify whether or not to include foregrounds
  - Allowed Values: "True" or "False"
  - Suggested Default: "True"
- **Correlations**
  - Description: A boolean value to specify whether or not to include white-noise correlations
  - Allowed Values: "True" or "False"
  - Suggested Default: "True"
- **Percentile Lo**
  - Description: The low percentile to display on output parameter distributions
  - Allowed Values: A floating point value between (0.0, 100.0)
  - Units: [%]

  - Suggested Default: 15.9, which corresponds to a one-sigma deviation for a Gaussian distribution
- **Percentile Hi**
  - Description: The high percentile to display on output parameter distributions
  - Allowed Values: A floating point value between (0.0, 100.0)
  - Units: [%]
  - Suggested Default: 84.1, which corresponds to a one-sigma deviation for a Gaussian distribution

```
# ***** Simulation Parameters *****
#
#--------------------------------------------------------------------------------------------------------------------
Parameter    | Value        | Description
#--------------------------------------------------------------------------------------------------------------------
Experiments  | 1            | Number of Experiment MC realizations. Positive integer
#--------------------------------------------------------------------------------------------------------------------
Observations | 1            | Number of Observation MC realizations per experiment realization. Positive integer.
#--------------------------------------------------------------------------------------------------------------------
Detectors    | 1            | Number of Detector MC realizations per observation realization. Positive Integer.
#--------------------------------------------------------------------------------------------------------------------
Resolution   | 1.000        | Resolution for integration over bands and spectra in GHz. Positive floating point value.
#--------------------------------------------------------------------------------------------------------------------
Foregrounds  | False        | Include Foregrounds? True or False
#--------------------------------------------------------------------------------------------------------------------
Correlations | True         | Include white noise correlations? True or False
#--------------------------------------------------------------------------------------------------------------------
Percentile   | [15.9, 84.1] | Percentiles to be shown on either side of the median in output spreads
#--------------------------------------------------------------------------------------------------------------------
```

Figure 13: An example `simulationInputs.txt` file

4.2 Define Parameter Variation

`calcBolos.py` has an option, as detailed in Section 2.5, to sweep over a user-defined subset of experiment parameters, which are discussed in Section 3.3. The parameters to be varied are specified in the file `BoloCalc/config/paramsToVary.txt`, an example of which is shown in Figure 14.

```
# Telescope | Camera    | Channel   | Optic        | Parameter    | Minimum   | Maximum   | Step Size
#-----------------------------------------------------------------------------------------------------
Tel         | Cam       | 1         | Aperture     | Temperature  | 5         | 9         | 1
#-----------------------------------------------------------------------------------------------------
Tel         | Cam       | 2         |              | Det Eff      | 0.1       | 0.9       | 0.2
```

Figure 14: Example `paramsToVary.txt` file

Parameters are defined by row and have the following column descriptors

- *Column 1*: Telescope in which the parameter is defined. If left blank, the parameter is assumed to be an experiment/foreground parameter defined `foregrounds.txt`
- *Column 2*: Camera in which the parameter is defined. If left blank, the parameter is assumed to be a telescope parameter defined in `telescope.txt`
- *Column 3*: Channel for which the parameter is defined. If left blank, the parameter is assumed to be a camera parameter defined in `camera.txt`
- *Column 4*: Optic for which the parameter is defined. This should be left blank **unless** this is an optic parameter defined in `optics.txt`
- *Column 5*: Parameter name
- *Column 6*: Minimum parameter value
- *Column 7*: Maximum parameter value

```
# Telescope | Camera    | Channel   | Optic         | Parameter    | Minimum   | Maximum   | Step Size
#---------------------------------------------------------------------------------------------------
Tel         | Cam       | 2         |               | Pixel Size** | 5         | 10        | 1
#---------------------------------------------------------------------------------------------------
```

Figure 15: Example `paramsToVary.txt` varying pixel size given a fixed field of view

- *Column 8*: Step size with which the parameter will be varied over the range [Minimum, Maximum]

If the parameter belongs to an optic for a specific channel ("Absorption" for Band 2), then all column entries will be filled. If the parameter belongs to an optic but applies to all bands (e.g. "Temperature"), then the channel column entry will be empty. If the parameter belongs to a detector in a specific channel (e.g. "Psat"), then the optic column entry will be blank. If the parameter belongs to the camera (e.g. "Boresight Elevation"), then the optic and channel column entries will be empty. If the parameter belongs to the telescope (e.g. "Elevation"), then the optic, channel, and camera column entries will be blank. Finally, if the parameter belongs to the experiment (e.g. "Dust Temperature"), then the optic, channel, camera, and telescope column entries will be blank.

The user can define as many simultaneous parameter variations as desired, and by default, `calcBolos.py` will calculate all unique combinations of those inputs. If the `--vary_tog` command-line argument is passed to `calcBolos.py`, then all parameters will be swept together. If the input parameter sweeps are of unequal length when `--vary_tog` is called, `calcBolos.py` will throw an error. See the below Section 2.5 for more details about how to use the `calcBolos.py` executable.

There is one "special," useful channel parameter that can be varied named **Pixel Size****; see Figure 15 for an example of how it might be implemented in the `paramsToVary.txt` file. When the identifier **Pixel Size**** is specified, `calcBolos.py` varies pixel diameter while maintaining a fixed focal plane size. Therefore, at each pixel size, detector number is re-calculated to be

$$N_{\mathrm{det;new}} = \left(\frac{D_{\mathrm{pix;old}}}{D_{\mathrm{pix;new}}} \right)^2 N_{\mathrm{det;old}} \tag{6}$$

where $D_{\mathrm{pix;old}}$ is the old pixel size, and $D_{\mathrm{pix;new}}$ is the new pixel size. When the pixel size is modulated, so is the aperture absorption ϵ_{stop}, which is assumed to be $\eta_{\mathrm{stop}} = 1 - \epsilon_{\mathrm{stop}}$, where η_{stop} is the aperture efficiency. The new aperture efficiency η_{stop} is calculated either using Equation 49 explicitly if **Absorption** for the aperture is defined as "NA" in `optics.txt`. If aperture absorption is not "NA," then the existing aperture efficiency is scaled as

$$\eta_{\mathrm{ap;new}} = \left(\frac{\eta_{\mathrm{stop}}(D_{\mathrm{pix;new}})}{\eta_{\mathrm{stop}}(D_{\mathrm{pix;old}})} \right) \eta_{\mathrm{ap;old}} \tag{7}$$

This functionality is useful when optimizing NET for pixel size, as smaller pixels give more detectors which gives more noise averaging in Equation 40, but smaller pixels also leads to lower optical throughput, which raises the per-detector NET in Equation 37.

4.2.1 Custom Parameter Variation Files

It's often the case that the user will want to sweep over arbitrary sets of experiment input parameters that aren't definable by a minimum, maximum, and step value. In this case, BoloCalc can accept custom-generated files with lists of input parameters. These files should have a file name that is the values in the `paramsToVary.txt` file columns, joined by underscores, and it should exist in the directory `BoloCalc/config/customVary`. For example, for if you want to vary the telescope `Tel`, camera `Cam`, channel `2`, optic `Primary`, parameter `Spillover`, then the custom input file should be `Tel_Cam_2_Primary_Spillover.txt` in the directory `BoloCalc/config/customVary`.

```
# Telescope      | Camera          | Channel         | Optic        | Parameter        | Minimum         | Maximum         | Step Size
#-----------------------------------------------------------------------------------------------------------------------------------
Tel              | Cam             | 2               | Primary      | Spillover        | CUST            | CUST            | CUST
#-----------------------------------------------------------------------------------------------------------------------------------
Tel              | Cam             | 2               |              | Det Eff          | CUST            | CUST            | CUST
```

Figure 16: A `paramsToVary.txt` file that calls for custom input files to be loaded for each parameter

```
(base) chilly-air:customVary chill$ ls
Tel_Cam_2_DetEff.txt                Tel_Cam_2_Primary_Spillover.txt
```

Figure 17: List of files whose names match those parameters that are called by `'CUST'` in the correponsinding `paramsToVary.txt` file

To invoke this file for the parameter variation, all three of the entries for `'Minimum'`, `'Maximum'`, and `'Step Size'` need to be called out as `'CUST'`.

As described in Section 4.2, in order to use the `--vry_tog` command-line argument for `calcBolos.py`, the lists in the custom-defined input parameter vary files must be of the same length.

4.3 Run Simulation

]

Once your simulation input parameters are defined in `simulationInputs.txt` as described in Section 4.1, and, if you want to sweep over a subset of experiment parameters, after your parameter sweeps are defined in `paramsToVary.txt`, you are ready to begin simulating your BoloCalc experiment.

- `--vary` induces the use of the `config/paramsToVary.txt` file to sweep over a user-defined set of parameters. For more information about how to do parameter sweeps, see Section 4.2.
- `--vary_tog` can only be passed when `--vary` is also passed. This causes the parameters defined in `config/paramsToVary.txt` to vary together, rather than varying all possible parameter combinations. For more information about how to do parameter sweeps, see Section 4.2.
- `--vary_name [VARY_NAME]` can only be passed when `--vary` is also passed. This causes the vary parameter output files to be saved to a directory within `[Camera]/paramVary/` named `[VARY_NAME]/`. When this keyword argument is not specified, the vary parameter output will be sent to a directory called `[date]_[hour]_[min]_[sec]/` within a given camera's `paramVary/` directory. See Section 5.1.4 for more information about vary output files.
- `--log_name [LOG_NAME]` causes the logging output to be saved to a file `BoloCalc/log/log_[LOG_NAME].txt`. When this keyword argument is not specified, the

```
0.010        |0.95
0.015        |0.90
0.018        |0.88
0.019        |0.86
0.022        |0.75
0.025        |
0.035        |
```

Figure 18: An example of the contents of the for the files `Tel_Cam_2_Primary_Spillover.txt` and `Tel_Cam_2_DetEff.txt`. The data must be organized column-wise in the top-to-bottom order of desired simulation (it doesn't have to be in increasing/decreasing order).

logging output is sent to the file `BoloCalc/log/log_[date].txt`. For more information about logging output, see Section 5.1.5.

- `--help` which prints descriptions of these the positional and keyword arguments for this executable to standard output.

4.3.1 Monte Carlo

BoloCalc is set up to iterate over uncertainties or probability distributions for experiment, observation, and detector parameters. Below is the basic structure of the calculation

- Generate N_{exp} experiment + telescope + camera realizations. Each experiment realization includes an independent sampling of foreground, telescope (minus PWV and elevation, which are handled separately), camera + focal plane, and optical chain parameters, including optics bands.
- For each experiment realization, generate N_{obs} observation realizations. Each observation realization includes an independent sampling of PWV and telescope boresight elevation.
- For each observation realization, generate N_{dets} detector realizations. Note that the number of detector realizations is distinct from the number of detectors within a given frequency channel. Each detector realization includes an independent sampling of channel parameters, including detector bands.

In all the number of frequency channel realizations is

$$N_{\mathrm{sims}} = N_{\mathrm{exp}} \times N_{\mathrm{obs}} \times N_{\mathrm{dets}} \tag{8}$$

5 Handle Outputs

Running `calcBolos.py` generates several output files, which quantify the performance of the simulated experiment. This files can either be inspected directly in their ASCII format, or they can be loaded into nested dictionaries using the `Unpack` class. In Section 5.1, we go through the layout and contents of the raw output files, and in Section 5.2, we review how one can use an `Unpack` object to easily generate tables and plots.

5.1 Output Files

`calcBolos.py` generates several types of output text files, either for a single simulation or for a parameter sweep, which are designed to human readable. In this section, review the output files, highlighting their contents and formatting.

5.1.1 `sensitivity.txt`

BoloCalc generates sensitivity tables at three levels of the experiment directory structure. The most informative of these is `[Experiment]/[Telescope]/[Camera]/sensitivity.txt` and contains 15 output parameters

- **Chan**
  - Description: Frequency channel name, constructed as `[Cam name]_[Band ID]`
- **Num Det**
  - Definition: N_{det} in Equation 60
  - Description: total number of detectors in this frequency channel + camera + telescope
- **Optical Throughput**
  - Definition: η_{inst} in Equation 14
  - Description: optical throughput for this frequency channel + camera + telescope
- **Optical Power**
  - Definition: P_{opt} in Equation 9
  - Description: optical power sensed by the detector in this frequency channel + camera + telescope
  - Units: pW
- **Telescope Temp**
  - Definition: T_{tel} in Equation 16
  - Description: telescope temperature for this frequency channel + camera + telescope
  - Units: K_{RJ}
- **Sky Temp**
  - Definition: T_{sky} in Equation 18
  - Description: sky temperature for this frequency channel + camera + telescope
  - Units: K_{RJ}
- **Photon NEP**
  - Definition: NEP_{ph} in Equation 19
  - Description: photon noise equivalent power for this frequency channel + camera + telescope
  - Units: $\mathrm{aW}/\sqrt{\mathrm{Hz}}$
- **Bolometer NEP**
  - Definition: NEP_{g} in Equation 20
  - Description: thermal carrier noise equivalent power for this frequency channel + camera + telescope
  - Units: $\mathrm{aW}/\sqrt{\mathrm{Hz}}$
- **Readout NEP**

  - Definition: NEP_{read} in Equation 31
  - Description: readout noise equivalent power for this frequency channel + camera + telescope
  - Units: $\mathrm{aW}/\sqrt{\mathrm{Hz}}$
- **Detector NEP**
  - Definition: NEP_{det} in Equation 36
  - Description: detector noise equivalent power for this frequency channel + camera + telescope
  - Units: $\mathrm{aW}/\sqrt{\mathrm{Hz}}$
- **Detector NET_CMB**
  - Definition: NET_{det} in Equation 37 assuming $\mathrm{d}P/\mathrm{d}T_{\mathrm{sky}} = \mathrm{d}P/\mathrm{d}T_{\mathrm{CMB}}$
  - Description: detector noise equivalent CMB temperature for this frequency channel + camera + telescope
  - Units: $\mu\mathrm{K}_{\mathrm{CMB}}\sqrt{\mathrm{s}}$
- **Detector NET_RJ**
  - Definition: NET_{det} in Equation 37 assuming $\mathrm{d}P/\mathrm{d}T_{\mathrm{sky}} = \mathrm{d}P/\mathrm{d}T_{\mathrm{RJ}}$
  - Description: detector noise equivalent RJ temperature for this frequency channel + camera + telescope
  - Units: $\mu\mathrm{K}_{\mathrm{RJ}}\sqrt{\mathrm{s}}$
- **Array NET_CMB**
  - Definition: NET_{arr} in Equation 40 assuming $\mathrm{d}P/\mathrm{d}T_{\mathrm{sky}} = \mathrm{d}P/\mathrm{d}T_{\mathrm{CMB}}$
  - Description: array noise equivalent CMB temperature for this frequency channel + camera + telescope
  - Units: $\mu\mathrm{K}_{\mathrm{CMB}}\sqrt{\mathrm{s}}$
- **Array NET_RJ**
  - Definition: NET_{arr} in Equation 40 assuming $\mathrm{d}P/\mathrm{d}T_{\mathrm{sky}} = \mathrm{d}P/\mathrm{d}T_{\mathrm{RJ}}$
  - Description: array noise equivalent RJ temperature for this frequency channel + camera + telescope
  - Units: $\mu\mathrm{K}_{\mathrm{RJ}}\sqrt{\mathrm{s}}$
- **Correlation Factor**
  - Definition: Γ in Equation 45
  - Description: array NET correlation factor for this frequency channel + camera + telescope
- **CMB Map Depth**
  - Definition: σ_{s} in Equation 46 assuming $\mathrm{d}P/\mathrm{d}T_{\mathrm{sky}} = \mathrm{d}P/\mathrm{d}T_{\mathrm{CMB}}$
  - Description: map depth in CMB temperature units for this frequency channel + camera + telescope
  - Units: $\mu\mathrm{K}_{\mathrm{CMB}}$ amin
- **RJ Map Depth**
  - Definition: σ_{s} in Equation 46 assuming $\mathrm{d}P/\mathrm{d}T_{\mathrm{sky}} = \mathrm{d}P/\mathrm{d}T_{\mathrm{RJ}}$
  - Description: map depth in RJ temperature units for this frequency channel + camera + telescope
  - Units: $\mu\mathrm{K}_{\mathrm{RJ}}$ amin

Figure 19 shows a truncated example of the `sensitivity.txt`.

Chan	Num Det	Optical Throughput	Optical Power	Telescope Temp	Sky Temp	Photon NEP	Bolometer NEP
			[pW]	[K_RJ]	[K_RJ]	[aW/rtHz]	[aW/rtHz]
Cam_Band1	1626	0.102 +/- (0.000,0.000)	2.16 +/- (0.00 , 0.00)	39.21 +/- (0.00 , 0.00)	8.81 +/- (0.00 , 0.00)	25.18 +/- (0.00 , 0.00)	11.79 +/- (0.00 , 0.00)
Cam_2	1626	0.222 +/- (0.000,0.000)	5.64 +/- (0.00 , 0.00)	36.33 +/- (0.00 , 0.00)	4.68 +/- (0.00 , 0.00)	51.75 +/- (0.00 , 0.00)	19.04 +/- (0.00 , 0.00)
Total	3252						

Figure 19: An example `[Experiment]/[Telescope]/[Camera]/sensitivity.txt` file

There are also output sensitivity files at levels `[Experiment]/[Telescope]/sensitivity.txt` and `[Experiment]/sensitivity.txt` which contain the parameters

- Num Det
- Array NET_CMB
- Array NET_RJ
- CMB Map Depth
- RJ Map Depth

which combine the sensitivities of all frequency channels within it that have the same **Band ID**. Therefore, if looking to quickly compute telescope- or experiment-wide NETs or map depths, it's important to label the **Bands ID**s such that the total is computed at this level. We recommend indexing all bands within a given experiment increasing with band center frequency.

5.1.2 `optical_power.txt`

`calcBolos.py` also generates a table for each frequency channel of the following

- **Element**: name of optical element, as defined within `optics.txt` for this camera
- **Power from Sky**: power incident on this optical element from the sky side, as defined between the detector band's -3 dB points
- **Power to Detect**: power emitted from the optical element onto the detector. This quantity is integrated over the detector bandpass
- **Cumulative Eff**: cumulative efficiency between this optical element and the sky, integrated over the detector bandpass

These values are useful for understanding how optical power propagates to the detector through the telescope and for identifying sources of parasitic loading. Figure 20 shows an example table from within an `optical_power.txt` file.

5.1.3 `output.txt`

The file `[Camera]/output.txt` holds the parameter outputs for every Monte Carlo simulation, for each frequency channel within the camera. These numbers form the histograms behind the central values and spreads in the `sensitivity.txt` file; see Section 5.1.1 for more details about the sensitivity outputs. Figure 21 shows an example `output.txt` file. The full output is useful for better understanding the shape of the output parameter distributions, as opposed to relying only on median and percentile values.

5.1.4 `paramVary`

As discussed in Section 2.5, BoloCalc can calculate sensitivity while varying the central values of a specified set of input parameters. The outputs of these parameter variations are stored in the camera-level directory `[Experiment]/[Telescope]/[Camera]/paramVary/` and have the structure shown below

```
[Camera_1]/paramVary/
|-- [vary_name]/
|    |-- [Camera_1]_[Channel_1].txt
|    |-- [Camera_1]_[Channel_2].txt
|    |-- ...
|    |-- [Camera_1]_[Channel_1]/
|    |    |-- output_001.txt
|    |    |-- output_002.txt
|    |    |-- ...
|    |-- [Camera_1]_[Channel_2]/
|    |-- ...
```

************************************ Cam_2 ************************************

Element	Power from Sky	Power to Detect	Cumulative Eff
	[pW]	[pW]	
CMB	0.000 +/- (0.000 , 0.000)	0.076 +/- (0.000,0.000)	0.222 +/- (0.000,0.000)
ATM	0.209 +/- (0.000 , 0.000)	0.567 +/- (0.000,0.000)	0.225 +/- (0.000,0.000)
Primary	1.741 +/- (0.000 , 0.000)	0.573 +/- (0.000,0.000)	0.228 +/- (0.000,0.000)
Mirror	3.221 +/- (0.000 , 0.000)	0.192 +/- (0.000,0.000)	0.230 +/- (0.000,0.000)
Window	3.708 +/- (0.000 , 0.000)	3.879 +/- (0.000,0.000)	0.260 +/- (0.000,0.000)
IRShader1	12.094 +/- (0.000 , 0.000)	0.047 +/- (0.000,0.000)	0.261 +/- (0.000,0.000)
IRShader2	12.189 +/- (0.000 , 0.000)	0.039 +/- (0.000,0.000)	0.261 +/- (0.000,0.000)
IRShader3	12.267 +/- (0.000 , 0.000)	0.032 +/- (0.000,0.000)	0.261 +/- (0.000,0.000)
AbsFilter	12.328 +/- (0.000 , 0.000)	0.128 +/- (0.000,0.000)	0.269 +/- (0.000,0.000)
LowPass1	12.243 +/- (0.000 , 0.000)	0.065 +/- (0.000,0.000)	0.287 +/- (0.000,0.000)
Lens1	11.645 +/- (0.000 , 0.000)	0.016 +/- (0.000,0.000)	0.306 +/- (0.000,0.000)
LowPass2	10.920 +/- (0.000 , 0.000)	0.003 +/- (0.000,0.000)	0.326 +/- (0.000,0.000)
Lens2	10.270 +/- (0.000 , 0.000)	0.000 +/- (0.000,0.000)	0.331 +/- (0.000,0.000)
Aperture	10.117 +/- (0.000 , 0.000)	0.026 +/- (0.000,0.000)	0.522 +/- (0.000,0.000)
LowPass3	6.484 +/- (0.000 , 0.000)	0.001 +/- (0.000,0.000)	0.556 +/- (0.000,0.000)
Lens3	6.096 +/- (0.000 , 0.000)	0.001 +/- (0.000,0.000)	0.564 +/- (0.000,0.000)
LowPass4	6.005 +/- (0.000 , 0.000)	0.000 +/- (0.000,0.000)	0.600 +/- (0.000,0.000)
Detector	5.645 +/- (0.000 , 0.000)	0.000 +/- (0.000,0.000)	1.000 +/- (0.000,0.000)

Figure 20: Example of a table within an optical_power.txt file

Eff	OptPow	TelTemp	SkyTemp	PhNEP	BoloNEP	ReadNEP	DetNEP	DetNET	DetNETRJ	ArrNET	ArrNETRJ	CorrFact	MapDepth	MapDepthRJ	Eff	OptPow	TelTemp
0.1074	1.8865	33.0740	10.7836	22.5463	11.0071	11.4975	27.5986	563.0938	447.1715	20.0566	15.9276	1.2017	13.7589	10.9264	0.2223	5.9492	37.1918
0.1052	2.1428	36.8109	10.7740	24.8270	11.7308	12.5833	30.2048	590.3090	468.7840	21.2402	16.8675	1.2726	14.5709	11.5712	0.2223	5.9492	37.1918
0.1476	1.4363	21.9542	10.0836	19.3200	9.6041	9.8871	23.7330	473.1460	375.7409	16.5803	13.1670	0.9934	11.3741	9.0326	0.2594	6.9407	37.1918
0.0853	1.9125	39.9062	10.8327	22.5211	11.0824	11.5024	27.6103	644.4256	511.7598	22.9470	18.2230	1.3748	15.7417	12.5010	0.2594	6.9407	37.1918
0.1210	0.9089	22.0958	9.9131	14.5124	7.6402	7.5157	18.0407	560.4237	445.0511	19.3393	15.3580	1.1587	13.2668	10.5356	0.2594	6.9407	37.1918
0.1080	1.8307	32.9076	9.4373	22.1067	10.8429	11.2835	27.0850	549.8474	436.6521	19.5619	15.5347	1.1426	13.4195	10.6569	0.2170	6.7927	38.0999
0.1058	2.0835	36.6233	9.4101	24.4006	11.5675	12.3746	29.7040	577.5782	458.6741	20.7678	16.4924	1.2131	14.2468	11.3139	0.2170	6.7927	38.0999
0.1484	1.3736	21.8352	8.6378	18.6788	9.3921	9.5809	22.9979	456.0169	362.1382	15.9205	12.6430	0.9299	10.9215	8.6731	0.2532	7.9249	38.0999
0.0858	1.8630	39.7032	9.4708	22.1701	10.9380	11.3288	27.1937	631.4923	501.4891	22.4752	17.8483	1.3128	15.4181	12.2440	0.2532	7.9249	38.0999
0.1216	0.8716	21.9832	8.5544	14.0750	7.4816	7.3046	17.5339	541.9103	430.3490	18.6369	14.8002	1.0886	12.7850	10.1530	0.2532	7.9249	38.0999
0.1080	1.8307	32.9076	9.4373	22.1067	10.8429	11.2835	27.0850	549.8474	436.6521	19.5619	15.5347	1.4472	13.4195	10.6569	0.2202	6.2845	37.5477
0.1058	2.0835	36.6233	9.4101	24.4006	11.5675	12.3746	29.7040	577.5782	458.6741	20.7678	16.4924	1.5365	14.2468	11.3139	0.2202	6.2845	37.5477
0.1484	1.3736	21.8352	8.6378	18.6788	9.3921	9.5809	22.9979	456.0169	362.1382	15.9205	12.6430	1.1778	10.9215	8.6731	0.2569	7.3320	37.5477
0.0858	1.8630	39.7032	9.4708	22.1701	10.9380	11.3288	27.1937	631.4923	501.4891	22.4752	17.8483	1.6628	15.4181	12.2440	0.2569	7.3320	37.5477
0.1216	0.8716	21.9832	8.5544	14.0750	7.4816	7.3046	17.5339	541.9103	430.3490	18.6369	14.8002	1.3788	12.7850	10.1530	0.2569	7.3320	37.5477
0.1077	1.8584	32.9899	10.1033	22.3244	10.9246	11.3896	27.3395	556.3968	441.8532	19.8062	15.7288	1.0475	13.5871	10.7900	0.2192	6.4564	37.7327
0.1055	2.1129	36.7161	10.0848	24.6116	11.6488	12.4780	29.9520	583.8697	463.6703	21.0010	16.6776	1.1107	14.4068	11.4409	0.2192	6.4564	37.7327
0.1480	1.4047	21.8941	9.3529	18.9970	9.4979	9.7330	23.3629	464.4982	368.8734	16.2469	12.9022	0.8593	11.1455	8.8510	0.2557	7.5325	37.7327
0.0855	1.8875	39.8036	10.1446	22.3438	11.0099	11.4148	27.4000	637.8845	506.5653	22.7081	18.0332	1.2010	15.5778	12.3709	0.2557	7.5325	37.7327
0.1213	0.8901	22.0389	9.2265	14.2921	7.5607	7.4094	17.7856	551.0791	437.6302	18.9845	15.0762	1.0041	13.0234	10.3424	0.2557	7.5325	37.7327
0.1081	1.8167	32.8661	9.1021	21.9970	10.8014	11.2300	26.9565	546.5531	434.0360	19.4392	15.4373	1.2206	13.3354	10.5900	0.2192	6.4564	37.7327
0.1059	2.0687	36.5765	9.0705	24.2943	11.5262	12.3225	29.5789	574.4161	456.1629	20.6508	16.3995	1.2967	14.1666	11.2501	0.2192	6.4564	37.7327
0.1486	1.3579	21.8056	8.2779	18.5179	9.3382	9.5039	22.8131	451.7415	358.7430	15.7561	12.5125	0.9894	10.8088	8.5836	0.2557	7.5325	37.7327
0.0859	1.8505	39.6526	9.1317	22.0827	10.9016	11.2855	27.0897	628.2790	498.9373	22.3583	17.7554	1.4039	15.3379	12.1803	0.2557	7.5325	37.7327
0.1218	0.8622	21.9552	8.2162	13.9652	7.4413	7.2515	17.4064	537.2869	426.6774	18.4618	14.6611	1.1593	12.6649	10.0576	0.2557	7.5325	37.7327
0.0838	0.6330	21.2676	9.1423	11.2068	6.3759	5.9086	14.1829	608.1316	482.9376	20.2647	16.0928	1.1242	13.9016	11.0398	0.2210	5.7831	36.2302
0.1065	1.8266	32.7206	10.0995	22.1283	10.8307	11.2899	27.1003	557.5930	442.8031	19.8507	15.7641	1.1013	13.6177	10.8142	0.2947	7.7108	36.2302
0.0905	1.3316	29.3913	10.0767	17.4660	9.2477	9.0566	21.7394	564.1779	448.0324	19.5702	15.5414	1.0857	13.4253	10.6615	0.1474	3.8554	36.2302
0.1438	2.5171	32.1684	10.0884	28.2924	12.7141	14.2142	34.1197	504.0840	400.3099	18.3593	14.5797	1.0185	12.5946	10.0018	0.2210	5.7831	36.2302
0.1015	1.9505	37.8841	10.1093	23.7021	11.1922	12.0117	28.8329	621.0352	493.1847	22.3868	17.7781	1.2419	15.3574	12.1958	0.2579	6.7469	36.2302
0.0838	0.6330	21.2676	9.1423	11.2068	6.3759	5.9086	14.1829	608.1316	482.9376	20.2647	16.0928	1.2261	13.9016	11.0398	0.2179	6.2875	36.7573
0.1065	1.8266	32.7206	10.0995	22.1283	10.8307	11.2899	27.1003	557.5930	442.8031	19.8507	15.7641	1.2011	13.6177	10.8142	0.2905	8.3834	36.7573
0.0905	1.3316	29.3913	10.0767	17.4660	9.2477	9.0566	21.7394	564.1779	448.0324	19.5702	15.5414	1.1841	13.4253	10.6615	0.1452	4.1917	36.7573
0.1438	2.5171	32.1684	10.0884	28.2924	12.7141	14.2142	34.1197	504.0840	400.3099	18.3593	14.5797	1.1108	12.5946	10.0018	0.2179	6.2875	36.7573
0.1015	1.9505	37.8841	10.1093	23.7021	11.1922	12.0117	28.8329	621.0352	493.1847	22.3868	17.7781	1.3545	15.3574	12.1958	0.2542	7.3355	36.7573
0.0833	0.6623	21.3847	10.6066	11.5590	6.5216	6.0819	14.5990	629.4122	499.8371	21.0463	16.7136	1.2267	14.4378	11.4656	0.2137	6.9571	37.4805
0.1060	1.8827	32.8892	11.4744	22.5683	10.9959	11.5043	27.6149	571.0918	453.5230	20.3542	16.1640	1.1864	13.9631	11.0885	0.2849	9.2761	37.4805
0.0901	1.3756	29.5410	11.4362	17.8499	9.3989	9.2445	22.1906	578.8061	459.6492	20.1114	15.9711	1.1722	13.7965	10.9563	0.1424	4.6380	37.4805
0.1431	2.5965	32.3362	11.4810	28.8875	12.9131	14.5003	34.8066	516.8991	410.4867	18.8478	14.9677	1.0986	12.9297	10.2679	0.2137	6.9571	37.4805
0.1010	2.0034	38.0768	11.4672	24.0857	11.3427	12.2001	29.2852	633.9708	503.4573	22.8650	18.1578	1.3327	15.6855	12.4564	0.2493	8.1166	37.4805
0.0833	0.6623	21.3847	10.6066	11.5590	6.5216	6.0819	14.5990	629.4122	499.8371	21.0463	16.7136	1.3737	14.4378	11.4656	0.2179	6.2875	36.7573
0.1060	1.8827	32.8892	11.4744	22.5683	10.9959	11.5043	27.6149	571.0918	453.5230	20.3542	16.1640	1.3285	13.9631	11.0885	0.2905	8.3834	36.7573
0.0901	1.3756	29.5410	11.4362	17.8499	9.3989	9.2445	22.1906	578.8061	459.6492	20.1114	15.9711	1.3126	13.7965	10.9563	0.1452	4.1917	36.7573
0.1431	2.5965	32.3362	11.4810	28.8875	12.9131	14.5003	34.8066	516.8991	410.4867	18.8478	14.9677	1.2302	12.9297	10.2679	0.2179	6.2875	36.7573
0.1010	2.0034	38.0768	11.4672	24.0857	11.3427	12.2001	29.2852	633.9708	503.4573	22.8650	18.1578	1.4924	15.6855	12.4564	0.2542	7.3355	36.7573
0.0835	0.6475	21.3255	9.8667	11.3818	6.4486	5.9947	14.3898	618.6792	491.3137	20.6517	16.4002	1.1105	14.1672	11.2506	0.2210	5.7831	36.2302
0.1063	1.8544	32.8040	10.7797	22.3462	10.9129	11.3962	27.3554	564.2681	448.1040	20.0994	15.9616	1.0808	13.7883	10.9497	0.2947	7.7108	36.2302
0.0903	1.3534	29.4654	10.7494	17.6563	9.3230	9.1498	21.9632	571.4171	453.7813	19.8378	15.7538	1.0667	13.6088	10.8072	0.1474	3.8554	36.2302
0.1434	2.5565	32.2515	10.7774	28.5870	12.8132	14.3559	34.4600	510.4180	405.3399	18.6005	14.7713	1.0002	12.7600	10.1332	0.2210	5.7831	36.2302

Figure 21: An example `output.txt` file for two frequency channels within a camera. The image truncates the file on the right hand and bottom sides. This particular file was generated by 5 experiment realizations, each with 5 observations, each with 5 detector realizations.

```
[Camera_2]/paramVary/
|-- ...
```

The directory in which the parameter vary files are stored `[vary_name]/` defaults to `[date]_[hour]_[min]_[sec]` unless otherwise specified by the `calcBolos.py` command-line argument `--vary_name`. See Section 4.3 for more details about the command-line options for `calcBolos.py`.

Figure 22 shows an example `varyParam` output file for a specific channel given 5 experiment realizations, each with 5 observations, each with 5 detector realizations. Figure 23 shows the list of all the raw output files generated, with the index in the file name corresponding to the parameter combination index in the `varyParm` output file. Finally, Figure 24 shows an example of the raw input file, which holds the output parameters for every Monte Carlo simulation.

	Tel	Tel Cam Band1					
Index	Observation EfficIency [NA]	Det Eff [NA]	Optical Throughput	Optical Power [pW]	Telescope Temp [K_RJ]	Sky Temp [K_RJ]	Photon NEP [aW/rtHz]
000	0.100	0.100	0.074 +/- (0.054,0.020)	0.53 +/- (1.48 , 0.51)	20.68 +/- (17.78, 1.10)	8.22 +/- (2.86 , 1.11)	11.08 +/- (13.25, 4.83)
001	0.100	0.300	0.066 +/- (0.071,0.026)	0.57 +/- (1.65 , 0.56)	20.80 +/- (18.88, 1.61)	7.91 +/- (3.20 , 1.44)	11.62 +/- (14.00, 4.51)
002	0.100	0.500	0.068 +/- (0.072,0.028)	0.98 +/- (1.69 , 0.26)	21.55 +/- (19.30, 1.10)	8.17 +/- (2.79 , 1.15)	14.99 +/- (15.31, 3.00)
003	0.100	0.700	0.065 +/- (0.078,0.030)	0.60 +/- (1.69 , 0.33)	20.75 +/- (19.44, 0.58)	8.30 +/- (2.70 , 1.01)	11.15 +/- (15.27, 3.57)
004	0.100	0.900	0.067 +/- (0.079,0.023)	0.72 +/- (1.44 , 0.57)	19.42 +/- (19.10, 1.55)	8.64 +/- (2.18 , 0.67)	13.32 +/- (11.93, 3.88)
005	0.200	0.100	0.080 +/- (0.059,0.020)	0.72 +/- (1.78 , 0.41)	21.45 +/- (19.63, 0.55)	8.24 +/- (2.44 , 1.02)	12.34 +/- (16.48, 3.44)
006	0.200	0.300	0.064 +/- (0.075,0.033)	0.66 +/- (1.78 , 0.32)	20.47 +/- (21.06, 1.29)	7.95 +/- (3.22 , 1.13)	11.97 +/- (15.70, 3.43)
007	0.200	0.500	0.075 +/- (0.062,0.008)	0.42 +/- (1.85 , 0.41)	19.79 +/- (19.18, 2.96)	7.91 +/- (2.93 , 1.27)	9.26 +/- (17.02, 3.98)
008	0.200	0.700	0.076 +/- (0.066,0.015)	0.62 +/- (1.48 , 0.48)	20.75 +/- (16.90, 1.16)	8.20 +/- (2.94 , 1.12)	11.22 +/- (13.65, 4.97)
009	0.200	0.900	0.066 +/- (0.073,0.019)	0.63 +/- (1.63 , 0.45)	20.98 +/- (18.03, 1.14)	8.51 +/- (2.32 , 0.67)	11.15 +/- (15.59, 4.80)
010	0.300	0.100	0.069 +/- (0.076,0.021)	0.74 +/- (1.49 , 0.35)	19.94 +/- (19.75, 1.17)	8.18 +/- (2.60 , 0.92)	12.76 +/- (13.07, 3.16)
011	0.300	0.300	0.063 +/- (0.076,0.032)	0.65 +/- (1.62 , 0.24)	21.43 +/- (19.33, 0.65)	8.50 +/- (2.19 , 0.63)	12.11 +/- (14.24, 2.28)
012	0.300	0.500	0.082 +/- (0.060,0.021)	0.74 +/- (1.78 , 0.40)	21.09 +/- (19.53, 0.82)	8.04 +/- (2.21 , 1.06)	12.59 +/- (16.88, 3.45)
013	0.300	0.700	0.067 +/- (0.079,0.019)	0.77 +/- (1.60 , 0.35)	20.64 +/- (17.74, 0.72)	8.03 +/- (2.79 , 1.04)	12.80 +/- (14.69, 3.43)
014	0.300	0.900	0.074 +/- (0.075,0.029)	0.76 +/- (1.42 , 0.39)	20.05 +/- (21.28, 2.10)	7.92 +/- (3.34 , 1.05)	13.69 +/- (11.90, 3.12)
015	0.400	0.100	0.068 +/- (0.062,0.019)	0.62 +/- (1.54 , 0.56)	20.51 +/- (16.16, 0.69)	8.08 +/- (2.78 , 1.24)	12.27 +/- (13.03, 3.89)
016	0.400	0.300	0.067 +/- (0.076,0.021)	0.48 +/- (2.09 , 0.64)	21.68 +/- (18.42, 0.63)	7.90 +/- (2.26 , 1.09)	9.90 +/- (19.01, 5.52)
017	0.400	0.500	0.067 +/- (0.066,0.019)	0.54 +/- (1.84 , 0.57)	20.99 +/- (18.11, 1.38)	8.52 +/- (2.60 , 0.82)	10.41 +/- (17.11, 6.17)
018	0.400	0.700	0.065 +/- (0.067,0.034)	0.62 +/- (1.51 , 0.54)	19.47 +/- (18.96, 2.48)	7.94 +/- (3.17 , 1.11)	11.38 +/- (13.61, 4.96)
019	0.400	0.900	0.071 +/- (0.065,0.033)	0.63 +/- (1.64 , 0.68)	21.33 +/- (19.03, 2.40)	7.78 +/- (2.97 , 1.31)	12.99 +/- (13.72, 4.24)
020	0.500	0.100	0.070 +/- (0.066,0.020)	0.53 +/- (2.10 , 0.87)	20.66 +/- (19.09, 4.22)	8.31 +/- (2.86 , 1.10)	10.31 +/- (19.11, 7.99)
021	0.500	0.300	0.070 +/- (0.070,0.020)	0.59 +/- (1.71 , 0.31)	20.64 +/- (18.74, 0.60)	8.04 +/- (2.44 , 0.97)	11.37 +/- (15.14, 3.03)
022	0.500	0.500	0.084 +/- (0.052,0.005)	0.52 +/- (1.77 , 0.74)	19.90 +/- (21.22, 2.72)	8.35 +/- (2.58 , 0.71)	10.85 +/- (15.96, 6.58)
023	0.500	0.700	0.082 +/- (0.050,0.019)	0.38 +/- (1.84 , 0.53)	20.38 +/- (20.38, 1.50)	8.11 +/- (2.91 , 0.83)	8.59 +/- (17.38, 5.63)
024	0.500	0.900	0.080 +/- (0.049,0.006)	0.79 +/- (1.73 , 0.47)	21.90 +/- (18.01, 4.21)	8.22 +/- (2.90 , 1.06)	12.31 +/- (16.23, 4.94)

Figure 22: An example `paramVary` output file for channel with **Band ID** "Band1." The image truncates the file on the right side. In this particular instance, 5 experiment realizations, each with 5 observations and 5 detector realizations were simulated for every parameter combination.

```
(base) chilly-air:testing chill$ ls -l Cam_Band1/
total 1000
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_000.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_001.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_002.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_003.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_004.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_005.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_006.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_007.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_008.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_009.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_010.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_011.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_012.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_013.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_014.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_015.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_016.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_017.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_018.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_019.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_020.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_021.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_022.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_023.txt
-rw-r--r--  1 chill  staff  18751 Sep  9 18:33 output_024.txt
```

Figure 23: A list of `output.txt` files for channel with **Band ID** "Band1." The output file index corresponds to the parameter combination index shown in the `paramVary` output file in Figure 22

Note that the `[vary_name]/` directory will only be created or overwritten in cameras which are influenced by the simulated parameter variation. In other words, if you're only sweeping over parameters in `Cam_1` using the `calcBolos.py` command line argument `--vary_name [vary_name]`, you won't see the directory `Cam_2/paramVary/[vary_name]/` update or be created.

5.1.5 Logging Files

BoloCalc has three classifications of logging output:

- Logging (normal operation): no standard output, save messages to a logging file
- Warning: print the warning message to standard output, but continue BoloCalc operation. Also save the warning messages to a logging file.
- Error: this raises an Exception with a custom "BoloCalc ERROR:" message and terminates the program

Logging messages are saved to a file in the directory `BoloCalc/log/` with the default file name

0.1186	1.4158	22.9179	9.1096	18.0536	9.5353	9.3562	22.4587	449.8639	357.2519	15.4982	12.3077	1.1623	10.6319	8.4431
0.0802	0.7573	22.3558	8.7430	12.1192	6.9736	6.4075	15.3806	559.3704	444.2146	18.6265	14.7919	1.3969	12.7779	10.1473
0.1291	0.8595	21.8012	8.3492	14.3312	7.4296	7.3975	17.7568	560.9314	445.4543	19.3193	15.3421	1.4488	13.2531	10.5247
0.1413	1.4069	22.5906	8.7561	18.7346	9.5055	9.6271	23.1089	457.1300	363.0221	15.9458	12.6631	1.1958	10.9389	8.6870
0.1235	1.1035	22.2313	8.5909	16.0499	8.4184	8.3053	19.9361	496.2992	394.1277	17.0856	13.5683	1.2813	11.7208	9.3079
0.1184	1.4301	22.9469	9.4459	18.1886	9.5834	9.4213	22.6147	453.5595	360.1867	15.6386	12.4191	0.9359	10.7281	8.5196
0.0801	0.7651	22.3839	9.0764	12.2050	7.0096	6.4498	15.4821	563.7683	447.7071	18.7875	14.9198	1.1243	12.8883	10.2351
0.1289	0.8694	21.8307	8.7067	14.4508	7.4721	7.4551	17.8951	566.0610	449.5279	19.5142	15.4969	1.1678	13.3868	10.6309
0.1412	1.4216	22.6193	9.0936	18.8813	9.5548	9.6973	23.2774	461.0456	366.1316	16.0963	12.7826	0.9632	11.0421	8.7689
0.1233	1.1154	22.2600	8.9337	16.1761	8.4635	8.3661	20.0820	500.5768	397.5247	17.2481	13.6973	1.0322	11.8323	9.3964
0.1184	1.4301	22.9469	9.4459	18.1886	9.5834	9.4213	22.6147	453.5595	360.1867	15.6386	12.4191	0.9321	10.7281	8.5196
0.0801	0.7651	22.3839	9.0764	12.2050	7.0096	6.4498	15.4821	563.7683	447.7071	18.7875	14.9198	1.1197	12.8883	10.2351
0.1289	0.8694	21.8307	8.7067	14.4508	7.4721	7.4551	17.8951	566.0610	449.5279	19.5142	15.4969	1.1630	13.3868	10.6309
0.1412	1.4216	22.6193	9.0936	18.8813	9.5548	9.6973	23.2774	461.0456	366.1316	16.0963	12.7826	0.9593	11.0421	8.7689
0.1233	1.1154	22.2600	8.9337	16.1761	8.4635	8.3661	20.0820	500.5768	397.5247	17.2481	13.6973	1.0280	11.8323	9.3964
0.1183	1.4452	22.9776	9.8020	18.3310	9.6340	9.4898	22.7793	457.4686	363.2910	15.7872	12.5371	1.1449	10.8301	8.6005
0.0800	0.7734	22.4136	9.4295	12.2955	7.0475	6.4944	15.5892	568.4175	451.3992	18.9579	15.0551	1.3749	13.0052	10.3279
0.1287	0.8798	21.8618	9.0854	14.5769	7.5167	7.5158	18.0410	571.4860	453.8361	19.7205	15.6607	1.4302	13.5284	10.7433
0.1410	1.4370	22.6497	9.4511	19.0362	9.6066	9.7713	23.4551	465.1880	369.4212	16.2556	12.9091	1.1789	11.1514	8.8557
0.1232	1.1279	22.2904	9.2967	16.3092	8.5109	8.4303	20.2360	505.1010	401.1175	17.4201	13.8339	1.2634	11.9502	9.4901
0.1183	1.4452	22.9776	9.8020	18.3310	9.6340	9.4898	22.7793	457.4686	363.2910	15.7872	12.5371	1.0545	10.8301	8.6005
0.0800	0.7734	22.4136	9.4295	12.2955	7.0475	6.4944	15.5892	568.4175	451.3992	18.9579	15.0551	1.2663	13.0052	10.3279
0.1287	0.8798	21.8618	9.0854	14.5769	7.5167	7.5158	18.0410	571.4860	453.8361	19.7205	15.6607	1.3172	13.5284	10.7433
0.1410	1.4370	22.6497	9.4511	19.0362	9.6066	9.7713	23.4551	465.1880	369.4212	16.2556	12.9091	1.0858	11.1514	8.8557
0.1232	1.1279	22.2904	9.2967	16.3092	8.5109	8.4303	20.2360	505.1010	401.1175	17.4201	13.8339	1.1635	11.9502	9.4901
0.0736	1.1107	29.2986	9.7340	15.1566	8.4458	7.9512	19.0860	588.7651	467.5579	20.0210	15.8993	1.1472	13.7345	10.9070
0.0903	1.5811	32.5051	9.7585	19.5896	10.0766	10.0951	24.2322	569.9435	452.6111	19.9573	15.8487	1.1436	13.6908	10.8723
0.1341	2.9082	40.8372	9.8312	32.2540	13.6663	16.0527	38.5328	588.9352	467.6930	21.8100	17.3201	1.2498	14.9618	11.8816
0.0909	1.3361	29.6741	9.7553	17.4673	9.2633	9.0605	21.7487	561.8632	446.1942	19.4708	15.4624	1.1157	13.3570	10.6073
0.0764	0.9030	24.6845	9.5694	13.2486	7.6151	7.0027	16.8094	559.2219	444.0967	18.7309	14.8748	1.0733	12.8494	10.2042
0.0734	1.1291	29.3727	10.4076	15.3227	8.5155	8.0332	19.2829	596.3346	473.5691	20.2966	16.1182	1.1871	13.9235	11.0571
0.0901	1.6056	32.5881	10.4397	19.7860	10.1543	10.1914	24.4635	576.8457	458.0923	20.2117	16.0508	1.1821	13.8653	11.0109
0.1338	2.9452	40.9401	10.5030	32.4992	13.7531	16.1716	38.8184	594.7859	472.3393	22.0308	17.4954	1.2885	15.1132	12.0019
0.0907	1.3578	29.7481	10.4203	17.6585	9.3380	9.1539	21.9731	569.0689	451.9166	19.7379	15.6745	1.1544	13.5403	10.7528
0.0762	0.9199	24.7465	10.2393	13.4203	7.6864	7.0872	17.0122	567.3868	450.5808	19.0305	15.1128	1.1130	13.0550	10.3674
0.0736	1.1107	29.2986	9.7340	15.1566	8.4458	7.9512	19.0860	588.7651	467.5579	20.0210	15.8993	1.1469	13.7345	10.9070
0.0903	1.5811	32.5051	9.7585	19.5896	10.0766	10.0951	24.2322	569.9435	452.6111	19.9573	15.8487	1.1433	13.6908	10.8723
0.1341	2.9082	40.8372	9.8312	32.2540	13.6663	16.0527	38.5328	588.9352	467.6930	21.8100	17.3201	1.2494	14.9618	11.8816
0.0909	1.3361	29.6741	9.7553	17.4673	9.2633	9.0605	21.7487	561.8632	446.1942	19.4708	15.4624	1.1154	13.3570	10.6073
0.0764	0.9030	24.6845	9.5694	13.2486	7.6151	7.0027	16.8094	559.2219	444.0967	18.7309	14.8748	1.0730	12.8494	10.2042
0.0735	1.1201	29.3363	10.0768	15.2412	8.4814	7.9930	19.1864	592.6190	470.6184	20.1612	16.0107	1.2027	13.8307	10.9834
0.0902	1.5936	32.5474	10.1052	19.6896	10.1163	10.1442	24.3501	573.4562	455.4007	20.0867	15.9515	1.1983	13.7795	10.9428
0.1339	2.9271	40.8896	10.1731	32.3788	13.7106	16.1132	38.6782	591.9106	470.0559	21.9223	17.4092	1.3078	15.0388	11.9428
0.0908	1.3472	29.7117	10.0938	17.5647	9.3015	9.1081	21.8631	565.5313	449.1072	19.6067	15.5703	1.1697	13.4503	10.6813
0.0763	0.9116	24.7161	9.9103	13.3362	7.6515	7.0458	16.9128	563.3805	447.3992	18.8834	14.9960	1.1265	12.9541	10.2873
0.0736	1.1107	29.2986	9.7340	15.1566	8.4458	7.9512	19.0860	588.7651	467.5579	20.0210	15.8993	1.2078	13.7345	10.9070
0.0903	1.5811	32.5051	9.7585	19.5896	10.0766	10.0951	24.2322	569.9435	452.6111	19.9573	15.8487	1.2040	13.6908	10.8723
0.1341	2.9082	40.8372	9.8312	32.2540	13.6663	16.0527	38.5328	588.9352	467.6930	21.8100	17.3201	1.3158	14.9618	11.8816
0.0909	1.3361	29.6741	9.7553	17.4673	9.2633	9.0605	21.7487	561.8632	446.1942	19.4708	15.4624	1.1746	13.3570	10.6073
0.0764	0.9030	24.6845	9.5694	13.2486	7.6151	7.0027	16.8094	559.2219	444.0967	18.7309	14.8748	1.1300	12.8494	10.2042

Figure 24: An example raw `output_001.txt` file. In this particular instance, 5 experiment realizations, each with 5 observations and 5 detector realizations were simulated for every parameter combination. the image truncates the file on the bottom side. This specific example holds all outputs listed in Section 5.1.1 for every Monte Carlo simulation. The index in the file name corresponds this simulation to the parameter combination indexed in the `paramVary` output file.

`log[date].txt`. However, the loging filename tag can be specified via the `calcBolos.py` command line argument `--log_name`, which is discussed explicitly in Section 4.3. Figure 25 shows an example BoloCalc logging file. Note that the logging output from every BoloCalc run is appended to that day's logging file and starts with the statement `***** Starting BoloCalc Program 'calcBolos.py' *****`

5.2 Unpack Outputs

BoloCalc has a class called "Unpack" in its source directory `BoloCalc/src/` that unpacks the outputs into layered dictionaries. There is an interactive Jupyter notebook tutorial `unpack_tutorial.ipynb` located at `BoloCalc/analyze/` that shows how to use the `Unpack` class in some detail, and there is a notebook-generated PDF in `BoloCalc/MANUAL/` that can be referenced if most convenient. To really see how the `Unpack` class works, we strongly recommend that you consult that PDF and notebook, but nonetheless, we briefly overview the class and its functional members here.

The unpack class can be imported and instantiated via the following code block

```
src_path = os.path.join("..", "src")
if src_path not in sys.path:
    sys.path.append(src_path)
import unpack as up

```

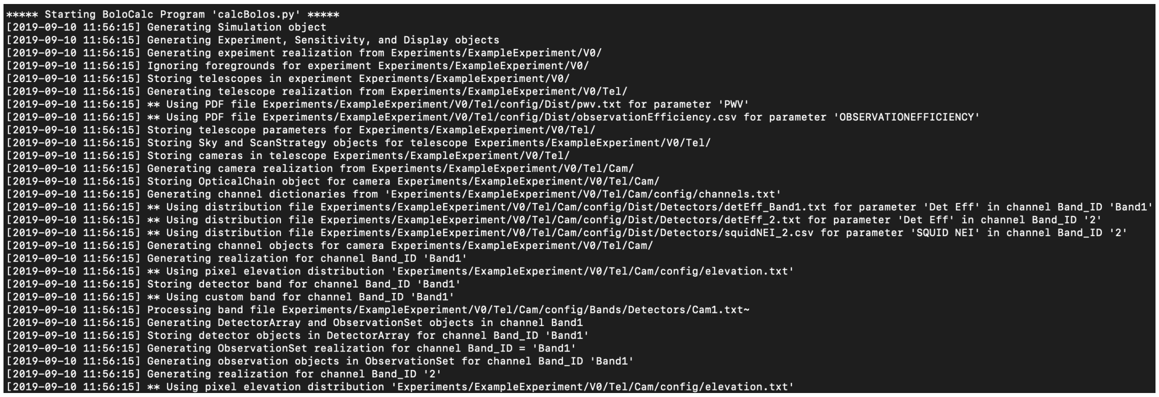


Figure 25: An example logging file. Important file imports, such as those for bands, probability distributions, or pixel elevation distributions, are demarcated by a "**."

```
unpack = up.Unpack()
```

The `unpack` object has two public methods that build three dictionaries

- `unpack.unpack_sensitivities(exp_dir)`: method to unpack the sensitivity data (not for parameter sweeps), written in `sensitivity.txt` files, for the experiment version directory `exp_dir`
- `unpack.unpack_parameter_vary(exp_dir, vary_name)`: method to unpack sensitivity data for parameter sweeps, written in `paramVary/vary_name/`, for the experiment version directory `exp_dir`
- `unpack.sens_outputs`: dictionary of sensitivity outputs (not for parameter sweeps)
- `unpack.vary_inputs`: dictionary of parameter-sweep inputs
- `unpack.vary_outputs`: dictionary of parameter-sweep outputs

For the methods, the input `exp_dir` experiment version directory must be the same experiment directory passed to `calcBolos.py`, and the `vary_name` argument must match the desired directory name in `paramVary`, which may be specified by the `calcBolos.py` command-line argument `--vary_name`, or if not specified, will be of the format `yyyy_mm_dd_hh_mm_ss`. See Section 5.1.4 for more details on how the `paramVary` subdirectories are created.

The two output dictionaries `unpack.sens_outputs` and `unpack.vary_outputs` are embedded in a way that traces the directory structure of the simulated experiment. Each has an experiment version, telescope, and camera layer, with the available sensitivity summary data keyed as `'Summary'` and the totality of the MC-simulated output marked as `'All'`. Within each of the `'Summary'` and `'All'` sub-dictionaries, there are dictionaries of the sensitivity outputs, keyed by the frequency channel names. The sensitivity outputs can be accessed via the list of keys below

1. `Num Det` = number of detectors
2. `Optical Throughput` = optical throughput, withOUT the aperture stop efficiency divided out
3. `Optical Power` = optical power in `[pW]`
4. `Telescope Temp` = telescope temperature, referenced to the instrument's first optic, in `[K_RJ]`
5. `Sky Temp` = sky temperature in `[K_RJ]`
6. `Photon NEP` = photon noise-equivalent power in `[aW/rtHz]`
7. `Bolometer NEP` = bolometer thermal carrier noise-equivalent power in `[aW/rtHz]`
8. `Readout NEP` = readout noise-equivalent power in `[aW/rtHz]`

9. `Detector NEP` = total detector noise-equivalent power in `[aW/rtHz]`
10. `Detector NET_CMB` = per-detector noise-equivalent CMB temperature in `[uK_CMB-rts]`
11. `Detector NET_RJ` = per-detector noise-equivalent CMB temperature in `[uK_RJ-rts]`
12. `Array NET_CMB` = array noise-equivalent CMB temperature in `[uK_CMB-rts]`
13. `Array NET_RJ` = array noise-equivalent RJ temperature in `[uK_RJ-rts]`
14. `Correlation Factor` = degree of white-noise correlation across the focal plane
15. `CMB Map Depth` = map depth in `[uK_CMB-amin]`
16. `RJ Map Depth` = map depth in `[uK_RJ-amin]`

The parameter-vary input layered dictionary has keys which label the input parameters that were varied in long-form arrays.

To obtain a better handle on how the `Unpack` methods and dictionaries can be used, you will either need to look at the Jupyter notebook interactive tutorial `unpack_tutorial.ipynb` in `BoloCalc/analyze/` or look at the notebook-generated PDF in `BoloCalc/MANUAL`.

6 Descriptions of Calculations

The primary functionality of BoloCalc is to import low-level instrument parameters and use them to estimate the instrument's noise-equivalent temperature (NET). This calculation is outlined by the following steps

1. Collect input parameters and construct an instrument model
2. Calculate per-detector noise-equivalent power (NEP) due to contrubutions from
   - Photon shot and wave noise
   - Bolometer thermal carrier noise
   - Readout amplifier noise
   - Johnson noise
3. Convert NEP to NET
4. Calculate the array-averaged NET, where the noise contributions from each detector are inverse-variance weighted to estimate the instantaneous sensitivity of the full camera
5. Calculate map depth, a quantification of the noise achieved in the final sky image considering sky fraction and observation time

We detail the assumptions and inputs for this calculation in the following subsections.

6.1 Optical Power

BoloCalc assumes an array of single-moded bolometers within an instrument that is stationary in time. The propagation of optical power from the sky to the focal plane is repreented by a one-dimensional chain of blackbody absorbers/emitters/attnuators in thermal equilibrium. The power deposited on the detectors is then an integral over the summation of each optical element's Planck spectra modified by its frequency-dependent efficiency and emissivity. Explicitly, the optical power on a detector is given as

$$P_{\mathrm{opt}} = \int_0^\infty \left[\sum_{i=1}^{N_{\mathrm{elem}}} p_i(\nu) \right] B(\nu) \mathrm{d}\nu \tag{9}$$

where the ν is frequency, $p_i(\nu)$ is the power spectral density of optical element i referred to the detector input, the summation contains all N_{elem} optical elements in the sky + telescope + camera and runs from the CMB to the focal plane, and $B(\nu)$ is the detector bandpass.

The power spectral density $p_i(\nu)$ for optical element i is determined by its blackbody temperature T_i, the transmission efficiency for all optics between it and the focal plane $[\eta_{i+1}(\nu), \ldots, \eta_{N_{\mathrm{elem}}}(\nu)]$, its emissivity $\epsilon_i(\nu)$, its spillover coefficient $\beta_i(\nu)$, the effective temperature by which its spilled power is absorbed $T_{\beta;i}$, its scattering coefficient $\delta_i(\nu)$, and the effective temperature by which its scattered power is absorbed $T_{\delta;i}$

$$p_i\left(T_i, \left[\eta_{i+1}(\nu), \ldots, \eta_{N_{\mathrm{elem}}}(\nu)\right], \epsilon_i(\nu), \beta_i(\nu), T_{\beta;i}, \delta_i(\nu), T_{\delta;i}\right) = \tag{10}$$

$$\prod_{j=i+1}^{N_{\mathrm{elem}}} \eta_j(\nu) \left[\epsilon_i(\nu) S(T_i, \nu) + \beta_i(\nu) S(T_{\beta;i}, \nu) + \delta_i(\nu) S(T_{\delta;i}, \nu)\right] \tag{11}$$

The power spectral density function $S(T, \nu)$ of the emitted and scattered power from each element is given by the Planck spectral density [2] for a polarimeter

[2] BoloCalc can be expanded to incorporate non-thermal spectral densities upon request

$$S(T,\nu) = A\Omega \frac{h\nu^3}{c^2} \frac{1}{\exp\left[\frac{h\nu}{k_\mathrm{B}T} - 1\right]} = \frac{h\nu}{\exp\left[\frac{h\nu}{k_\mathrm{B}T} - 1\right]} \tag{12}$$

where for a diffraction-limited, single-moded detector, the etendue $A\Omega$ is given by the square of the detected wavelength

$$A\Omega = \left(\frac{c}{\nu}\right)^2 \tag{13}$$

6.2 Optical Throughput

The throughput of the instrument is defined to be the total transmission through all optical elements in the telescope + camera, including the detector, and is defined as

$$\eta_\mathrm{inst} = \prod_{i=0}^{N_\mathrm{inst}} \eta_i \tag{14}$$

where N_inst represents all optical elements within in the instrument, which are defined in the `optics.txt` file, and for the detector, whose efficiency is defined in `channels.txt`.

6.3 Telescope Temperature

Another useful parameter when assessing instrument performance is the telescope temperature, which is defined as the Rayleigh-Jeans temperature of an object with emissivity 1 emitting into a 0 K instrument to reproduce the total power seen by the detector due to instrument emission. The total power due to emission from the instrument is a subset of Equation 9 and is defined as

$$P_\mathrm{inst} = \int_0^\infty \left[\sum_i^{N_\mathrm{inst}} p_i(\nu)\right] B(\nu)\mathrm{d}\nu \tag{15}$$

where N_inst are elements within the telescope (excluding the sky elements). The telescope temperature is defined to be

$$T_\mathrm{tel} = \frac{P_{inst}}{k_\mathrm{B}\eta_\mathrm{inst}} \tag{16}$$

where η_inst is the throughput defined in Equation 14. Note that the temperature units are $\mathrm{K_{RJ}}$.

6.4 Sky Temperature

In a similar manner to the telescope temperature, we can define the sky temperature, which is simply the Rayleigh-Jeans temperature of the detected sky power. Total power due to emission from the sky is a subset of Equation 9 and is defined as

$$P_\mathrm{sky} = \int_0^\infty \left[\sum_i^{N_\mathrm{sky}} p_i(\nu)\right] B(\nu)\mathrm{d}\nu \tag{17}$$

where N_sky are elements outside of the telescope, which are explicitly

- CMB
- Galactic dust (if foregrounds are enabled)
- Galactic synchrotron (if foregrounds are enabled)
- Atmosphere (if the **Site** parameter is not "Space")

- Room (if the **Site** parameter is "Room")

The sky temperature is defined to be

$$T_{\text{sky}} = \frac{P_{sky}}{k_{\text{B}}\eta_{\text{inst}}} \tag{18}$$

where η_{inst} is the throughput defined in Equation 14. Note that the temperature units are K_{RJ}.

6.5 Photon Noise Equivalent Power

Photon noise in bolometric detection is the result of fluctuations in the arrival times of photons at the absorbing element

$$NEP_{\text{ph}} = \sqrt{2\int_0^{\infty}\left[h\nu\sum_{i=1}^{N_{\text{elem}}} p_i(\nu) + \left(\sum_{i=1}^{N_{\text{elem}}} p_i(\nu)\right)^2\right] B^2(\nu)\text{d}\nu} \tag{19}$$

There are two contributions to NEP_{ph}. The first term represents shot noise NEP_{shot}, which dominates when the photon occupation number $\ll 1$ (e.g. optical wavelengths) and is $\propto \sqrt{P_{\text{opt}}}$. The second term represents wave noise NEP_{wave}, which dominates when the photon occupation number is $\gg 1$ (e.g. radio wavelengths) and is $\propto P_{\text{opt}}$. For ground-based experiments, the photon occupation number at ~100 GHz is ~1, and therefore a careful handling of both terms is necessary for an accurate NET estimate.

6.6 Thermal Carrier Noise Equivalent Power

Thermal carrier noise in bolometers arises due to fluctuations in heat flow between the absorbing element and the bath to which it is weakly connected and given by the equation

$$NEP_{\text{g}} = \sqrt{4k_{\text{B}}F_{\text{link}}T_{\text{oper}}^2 G} \tag{20}$$

where T_{oper} is the bolometer operating temperature, G is the thermal conductance from the absorbing element to the bath, and F_{link} is a numerical factor that depends on the link's thermal conduction index n. BoloCalc allows the bolometer operating temperature to be defined as a ratio of the bath temperature as

$$T_{\text{oper}} = f_{\text{oper}}T_{\text{bath}} \tag{21}$$

Similarly, F_{link} can be theoretically estimated via the equation

$$F_{\text{link}} = \frac{n+1}{2n+3}\frac{1-(T_{\text{bath}}/T_{\text{oper}})^{2n+3}}{1-(T_{\text{bath}}/T_{\text{oper}})^{n+1}} \tag{22}$$

where T_{bath} is the bath temperature. However, NEP_{g} can vary depending on the specifics of the bolometer geometry, composition and fabrication. For example, transition-edge sensors (TES) have known pathologial noise sources, such as flux flow noise and non-equilibrium Johnson noise, that increase the measured NEP_{g} beyond that of Mather's theoretical prediction. Therefore, BoloCalc provides an option for F_{link} to be set independent of T_{bath} and n, allowing NEP_{g} to be tuned phenomenologically.

Thermal conductance can be parameterized in terms of n, T_{bath}, and the bolometer saturation power P_{sat}—or the power conducted from the bolometer to the bath—as

$$G = P_{\text{sat}}(n+1)\frac{T_{\text{oper}}^n}{T_{\text{oper}}^{n+1} - T_{\text{bath}}^{n+1}} \tag{23}$$

Therefore, $NEP_{\rm g} \propto P_{\rm sat}$, making the tuning of saturation power important to optimizing detector sensitivity.

6.7 Readout Noise Equivalent Power

Modern CMB detectors are low-impedance, voltage-biased bolometers read out using superconducting quantum interference device (SQUID) transimpedance amplifiers. SQUIDs are current sensors, and a modulation of power on a voltage-biased bolometer corresponds to a modulation of current at the amplifier input. Therefore, SQUID amplifier noise is typically characterized in terms of a noise-equivalent current NEI, which has units of $\frac{\rm A}{\sqrt{\rm Hz}}$. In order to refer NEI to an NEP, we need to consider the bolometer responsivity $S_{\rm I} = {\rm d}I_{\rm elec}/{\rm d}P$. BoloCalc quantifies responsivity via the relationship

$$S_{\rm I} = -S_{\rm fact} \frac{1}{V_{\rm elec}} \tag{24}$$

where $V_{\rm elec}$ is the bias voltage across the bolometer. In the presence of electrothermal feedback, the bolometer responsivity can typically be written in terms of the bolometer DC loop gain $\mathcal{L}$, the bolometer time constant τ, and the modulation mode frequency ω as

$$S_{\rm I} = -\tilde{S}_{\rm fact} \frac{1}{V_{\rm elec}} \frac{\mathcal{L}}{\mathcal{L}+1} \frac{1}{1+i\omega\tau} \tag{25}$$

where $\tilde{S}_{\rm fact}$ is a factor that can depend on whether the bias voltage $V_{\rm elec}$ is DC (e.g. time-division multiplexing or microwave multiplexing) or AC (frequency-domain multiplexing) RMS:

$$\tilde{S}_{\rm fact} = \begin{cases} 1 & \text{if } V_{\rm elec} \text{ is DC} \\ \sqrt{2} & \text{if } V_{\rm elec} \text{ is AC RMS} \end{cases} \tag{26}$$

BoloCalc allows the user to set the parameter $S_{\rm fact}$ explicitly, but one common starting assumption is that in the limit of high loop gain $\mathcal{L} \gg 1$ and at frequencies much slower than the bolometer time constant $\omega\tau \ll 1$, a simple approximate relationship for the responsivity is

$$S_{\rm I} \approx -\tilde{S}_{\rm fact} \frac{1}{V_{\rm elec}} \tag{27}$$

or, equivalently

$$S_{\rm fact} \approx \tilde{S}_{\rm fact} \tag{28}$$

where common-use cases for $\tilde{S}_{\rm fact}$ are given in Equation 26.

The bias voltage is determined from the bolometer resistance $R_{\rm bolo}$ and the bias power $P_{\rm elec}$ as

$$V_{\rm elec} = \sqrt{R_{\rm bolo} P_{\rm elec}}\,, \tag{29}$$

and furthermore, the bias power $P_{\rm elec}$ is determined from the bolometer saturation power $P_{\rm sat}$ and the absorbed optical power $P_{\rm opt}$ as

$$P_{\rm elec} = P_{\rm sat} - P_{\rm opt} \tag{30}$$

Finally, $NEP_{\rm read}$ can be written in terms of the bolometer responsivity, bolometer resistance, and optical power as

$$NEP_{\rm read} = \frac{NEI}{S_{\rm I}} = \frac{\sqrt{R_{\rm bolo}\left(P_{\rm sat} - P_{\rm opt}\right)}}{S_{\rm fact}} NEI \tag{31}$$

Note that a decreased bolometer responsivity factor S_{fact} leads to increased readout noise.

BoloCalc also has an option to define the readout noise as a fraction of all other quadrature-summed noise sources NEP_{other} as

$$NEP_{\mathrm{read}} = \Delta_{\mathrm{read}} \times NEP_{\mathrm{other}} \tag{32}$$

See Section 3.5.2 for more details about setting channel noise parameters.

6.8 Johnson Noise Equivalent Power

Johnson noise arises due to thermal fluctuations in the bolometer which cause resistance fluctuations. The relationship for Johnson noise equivalent current NEI_{johnson} is given by

$$NEI_{\mathrm{johnson}} = \frac{1}{\mathcal{L}}\sqrt{\frac{4k_{\mathrm{B}}T_{\mathrm{oper}}}{R_{\mathrm{bolo}}}} \tag{33}$$

The Johnson current noise can be converted to an NEP using the bolometer responsivity S_{I}, which is defined in Equation 25 as

$$NEP_{\mathrm{johnson}} = \frac{NEI_{\mathrm{johnson}}}{S_{\mathrm{I}}} = \tilde{S}_{\mathrm{fact}}\frac{\mathcal{L}+1}{\mathcal{L}^2}\frac{1}{1+i\omega\tau}\sqrt{4k_{\mathrm{B}}T_{\mathrm{oper}}P_{\mathrm{elec}}}\,, \tag{34}$$

where P_{elec} is defined in Equation 30. Two things to note regarding this equation. First, in the limit of large loop gain $\mathcal{L} \gg 1$, $NEP_{\mathrm{johnson}} \rightarrow 0$. Second, Johnson noise is suppressed by a factor of factor of $1/\mathcal{L}$ with respect to readout noise.

Therefore, BoloCalc, as of v0.10.0, assumes that the bolometers have a high enough loop gain to render Johnson noise negligible compared to other noise sources.

$$NEP_{\mathrm{johnson}} = 0 \tag{35}$$

6.9 Detector Noise Equivalent Power

Assuming that the all detector noise sources add in quadrature and that Johnson noise is zero, as explained in Section 6.8, then the total detector NEP is given as

$$NEP_{\mathrm{det}} = \sqrt{NEP_{\mathrm{ph}}^2 + NEP_{\mathrm{g}}^2 + NEP_{\mathrm{read}}^2} \tag{36}$$

where NEP_{ph} is the photon NEP, NEP_{g} is the thermal carrier NEP, and NEP_{read} is the readout NEP.

6.10 Noise-equivalent Temperature

A bolometer is built to measure fluctuations in the incident power due to fluctuations in the sky temperature. Therefore, it is useful to convert bolometer NEP into a noise-equivalent sky temperature (NET). The total noise in the bolometer output is the quadrature sum of photon noise, thermal carrier noise, and readout noise as described in Section 6.9, and the conversion to NET is given by

$$NET_{\mathrm{det}} = M\frac{NEP_{\mathrm{det}}}{\sqrt{2}\,(\mathrm{d}P/\mathrm{d}T_{\mathrm{sky}})} \tag{37}$$

where M is a "margin factor" applied to the expected per-detector NET, and the $\sqrt{2}$ arises due to a unit conversion from output bandwidth $\frac{1}{\sqrt{\mathrm{Hz}}}$ to integration time $\sqrt{s}$. BoloCalc considers two different sky temperature units: $\mathrm{K}_{\mathrm{CMB}}$ and K_{RJ}, corresponding to a signal reference of a 2.725 K blackbody and a 1 K Rayleigh-Jeans source, respectively. The conversion factor for $\mathrm{K}_{\mathrm{CMB}}$ is

$$\frac{\mathrm{d}P}{\mathrm{d}T_{\mathrm{CMB}}} = \xi \int_0^\infty \left[\prod_{i=1}^{N_{\mathrm{elem}}} \frac{1}{k_{\mathrm{B}}} \left(\frac{h\nu}{T_{\mathrm{CMB}} \left(\exp\left[h\nu / k_{\mathrm{B}} T_{\mathrm{CMB}} \right] - 1 \right)} \right)^2 \exp\left[h\nu / k_{\mathrm{B}} T_{\mathrm{CMB}} \right] \right] B(\nu) \mathrm{d}\nu \tag{38}$$

where ξ is an overall signal degradation factor, which might be, for example, associated with poor far-field image formation at the focal plane, and $B(\nu)$ is the detector bandpass. Note that Equation 38 has units of $\mathrm{W/K_{CMB}}$. The conversion factor for $\mathrm{K_{RJ}}$ is

$$\frac{\mathrm{d}P}{\mathrm{d}T_{\mathrm{RJ}}} = \xi \int_0^\infty k_{\mathrm{B}} B(\nu) \mathrm{d}\nu \tag{39}$$

where $B(\nu)$ is the detector bandpass. Note that Equation 39 has units of $\mathrm{W/K_{RJ}}$.

6.11 Array Noise-equivalent Temperature

When reconstructing the sky during analysis, data from each detector are co-added in the map domain to improve signal to noise in the final map. To quantify this SNR increase in the time domian, we defined "array NET" as the inverse-variance-weighted average of the NETs of all yielded detectors within a given frequency channel

$$NET_{\mathrm{arr}} = \frac{NET_{\mathrm{det}}}{\sqrt{Y N_{\mathrm{det}}}} \Gamma \tag{40}$$

where N_{det} is the number of detectors in this frequency channel, Y is the detector yield, and Γ is a factor which quantifies the degree to which white noise is correlated between detector pixels on the focal plane.

6.12 Correlation Factor

The optical correlation coefficient between detector i and detector j when observing a source through the aperture stop is given by

$$\gamma_{i,j} = \frac{\langle |e_i|^2 |e_j|^2 \rangle - \langle |e_i|^2 \rangle \langle |e_j|^2 \rangle}{\mathrm{RMS}\left(|e_i|^2\right) \, \mathrm{RMS}\left(|e_j|^2\right)} \,, \tag{41}$$

where e_i is the integral of the source electric field at the aperture plane $a(x,y)$ for detector i with beam $b_i(x,y)$ and optical path length to the source $\ell_i(x,y)$

$$e_i = \iint \mathrm{d}x \, \mathrm{d}y \, e^{2\pi i \ell_i(x,y)} \, b_i(x,y) \, a(x,y) \,. \tag{42}$$

The correlation coefficient depends on the $F\lambda$ spacing—where F is the F-number at the focal plane—between pixels, the intensity, etendue, and angular location of the source, and whether the source is viewed within or outside of the aperture. The cumulative correlation coefficient γ is given by a summation of the correlation coefficients between all N_{pix} detector pixels on the focal plane

$$\gamma = \frac{1}{N_{\mathrm{pix}} - 1} \sum_i \sum_{j \neq i} \gamma_{i,j} \,. \tag{43}$$

Correlations then propagate to MS by suppressing the degree to which wave noise is averaged down when inverse-variance averaging the detector data

$$\mathrm{NET_{arr}} = \sqrt{\frac{\mathrm{NET}_{\mathrm{shot}}^2 + (1+\gamma)\mathrm{NET}_{\mathrm{wave}}^2 + \mathrm{NET}_{\mathrm{g}}^2 + \mathrm{NET}_{\mathrm{read}}^2}{Y N_{\mathrm{det}}}} \,. \tag{44}$$

We can now write the array NET correlation suppression factor Γ defined in Equation 40 as

$$\Gamma = \sqrt{1 + \frac{\gamma\,\mathrm{NET}^2_{\mathrm{wave}}}{\mathrm{NET}^2_{\mathrm{shot}} + \mathrm{NET}^2_{\mathrm{wave}} + \mathrm{NET}^2_{\mathrm{g}} + \mathrm{NET}^2_{\mathrm{read}}}}\,. \tag{45}$$

The impact of correlations on array NET depends on the contribution of wave noise to the total noise in the system and on the optical correlation factor γ. The BoloCalc authors are actively working on upgrading this part of the calculation and on publishing the details of the assumptions behind it.

6.13 Map Depth

Finally, array NET—white noise in the time domain—is converted from units of $\mathrm{K}\sqrt{\mathrm{s}}$ to units of K-arcmin—white noise in the map domain— using the equation

$$\sigma_{\mathrm{S}} = \sqrt{\frac{4\pi f_{\mathrm{sky}} NET^2_{\mathrm{arr}}}{\eta_{\mathrm{obs}} t_{\mathrm{obs}}}} \left(\frac{10800\ \mathrm{arcmin}}{\pi}\right) \tag{46}$$

where f_{sky} is the (effective) fraction of sky observed, NET_{arr} is defined in Equation 40, η_{obs} is the telescope observation efficiency, and t_{obs} is the total operation time of the telescope. Note that BoloCalc computes Map Depth in both units of $\mathrm{K_{CMB}}$-arcmin and $\mathrm{K_{RJ}}$-arcmin.

6.14 Auxiliary Optical Calculations

There are a few calculations that are carried out "under the hood" of Eqution 9 if specific sets of parameters are defined in the experiment input files. For more information about optical parameter definitions and dependencies, see Section 3.2. Ultimately, as described in Equation 11, the sensitivity calculation needs temperature, emissivity, and efficiency to calculate the optical power absorbed by, emitted from, and reflected from each optical element. This section details how some of those optical powers are estimated from a few different classes of optical elements.

6.14.1 Synchrotron Emission

The synchrotron power spectral density for a single polarization in units of W Hz^{-1} sr^{-1} is given by the equation

$$p_{\mathrm{synch}}(\nu) = A\Omega \frac{\nu^2}{c^2} k_{\mathrm{B}} T_{\mathrm{s}} \left(\frac{\nu}{\nu_{\mathrm{s}}}\right)^{n_{\mathrm{s}}} = k_{\mathrm{B}} T_{\mathrm{s}} \left(\frac{\nu}{\nu_{\mathrm{s}}}\right)^{n_{\mathrm{s}}} \tag{47}$$

where T_{s} is the sychrotron brightness temperature in $\mathrm{K_{RJ}}$ at pivot frequency ν_{s}, n_{s} is the synchrotron spectral index, and the etendue $A\Omega$ for a diffraction-limited detector is given by Equation 13. This spectral density is included in the summation described in Equation 9 when the **Foregrounds** parameter in `BoloCalc/config/simInputs.txt`, as described in Section 4.1, is set to `True`. While sychrotron brightness is typically much lower than the CMB brightness at high galactic latitudes at peak CMB frequencies, an accurate estimate of optical power from synchrotron emission may become important at lower frequencies or for observations near the galactic plane.

6.14.2 Dust Emission

The dust power spectral density in units of W Hz^{-1} sr^{-1} is given by the following equation

$$p_{\mathrm{dust}}(\nu) = A_{\mathrm{d}} A\Omega \left(\frac{\nu}{\nu_{\mathrm{d}}}\right)^{n_{\mathrm{d}}} \frac{S(T_{\mathrm{d}},\nu)}{S(T_{\mathrm{d}},\nu_{\mathrm{d}})} \times 10^{-20} = A_{\mathrm{d}} \frac{c^2}{\nu^2} \left(\frac{\nu}{\nu_{\mathrm{d}}}\right)^{n_{\mathrm{d}}} \frac{S(T_{\mathrm{d}},\nu)}{S(T_{\mathrm{d}},\nu_{\mathrm{d}})} \times 10^{-20} \tag{48}$$

where A_d is the dust amplitude in MJy sr^{-1} at pivot frequency ν_d, n_d is the dust spectral index, $S(T_\mathrm{d}, \nu)$ is the blackbody spectral density function defined in Equation 12, and T_d is the dust temperature. Note that the conversion from MJy to W is $\times 10^{-26}$W/MJy. This spectral density is included in the summation described in Equation 9 when the **Include Foregrounds** parameter described in Section 4.1 is set to `True`. While dust brightness is typically much lower than the CMB brightness at high galactic latitudes at peak CMB frequencies, an accurate estimate of optical power from dust emission may become important at higher frequencies or for observations near the galactic plane.

6.14.3 Atmosphere Emission

BoloCalc utlilizes atmosheric simulations of the atmosphere at the Atacama and South Pole sites, as well as from the high altitude of a balloon, generated by the AM atmsopheric modling code [3], which uses data from the MERRA-2 meterological reanalysis [4] as input. Note that more sights may be able to be added to BoloCalc upon request.

The output from the AM simulation produces a spectrum of physical sky temperature and sky efficiency/transmission, and the band-integrated values produce loading results consistent with measure sky loading in existing Atacama experiments. The allowed range of input elevations handled by BoloCalc is 20 - 90 deg, and the range of input precipitable water vapor (PWV) values is from 0 - 8 mm.

6.14.4 Aperture Stop Spillover Efficiency/Absorption

Aperture efficiency can be specified explicitly, or, if it is defined as "NA," it can be derived from a combination of other detector pixel parameters. The equation for aperture efficiency assumes a Gaussian beam at the detector pixel which is parameterized by the ratio of the pixel diameter D_pix to the Gaussian beam waist w_0 as $w_\mathrm{f} = D_\mathrm{pix}/w_0$ using the equation

$$\eta_\mathrm{stop}(\nu) = 1 - \exp\left[-\frac{\pi^2}{2}\left(\frac{D_\mathrm{pix}}{F\,\lambda\,w_\mathrm{f}}\right)^2\right] \tag{49}$$

where F is the F-number at the focal plane and $\lambda = c/\nu$ is the observation wavelength. This quantity is integrated over the observation band when calculating the in-band optical power and signal attenuation referenced to the detector.

Note that the aperture is assumed to be black absorber, and therefore, aperture reflection is assumed be zero

$$r_\mathrm{stop} = 0 \tag{50}$$

and the aperture absorption is assumed to be

$$\epsilon_\mathrm{stop} = 1 - \eta_\mathrm{stop} \tag{51}$$

6.14.5 Dielectric Absorption/Emission

Absorption in a refractive optical element can be specified explicitly, or, if it is defined as "NA," it can be derived using the following equation

$$\epsilon(\lambda) = 1 - \exp\left[-2\pi\, t\, n\, (\tan\delta)\, \lambda\right] \tag{52}$$

[3]https://www.cfa.harvard.edu/ spaine/am/

[4]https://gmao.gsfc.nasa.gov/reanalysis/MERRA-2/

where t is the thickness of the dielectric substrate, n is the refractive index, $\tan\delta$ is the loss tangent, and λ is the observation wavelength. This quantity is integrated over the observation band when calculating the in-band optical power and signal attenuation referenced to the detector.

6.14.6 Reflector Absorption/Emission

Absorption in a reflective optical element can be specified explicitly, or, if it is left as "NA," it can be derived using the following equation

$$\epsilon(\nu) = 4\sqrt{\frac{\pi\nu\mu_0}{\sigma_c}}\frac{1}{Z_0} \tag{53}$$

where ν is the observation frequency, μ_0 is the permeability of free space, $Z_0 = \sqrt{\mu_0/\epsilon_0}$ is the impedance of free space, and σ_c is the reflector conductivity at the reflector's operating temperature. This quantity is integrated over the observation band when calculating the in-band power and signal attenuation referenced to the detector.

6.14.7 Ruze Scattering

Scattering off of a reflective or refractive optical element can be specified explicitly, or, it if is defined as "NA," it can be derived using Ruze's equation

$$\delta(\nu) = \exp\left[\left(\frac{4\pi\sigma_r}{\lambda}\right)^2\right] \tag{54}$$

where σ_r is the RMS surface roughness of the optical element and λ is the observation wavelength. This quantity is integrated over the observation band when calculating the in-band power and signal attenuation referenced to the detector.

6.14.8 Efficiency

Efficiency can be defined explicitly for any optic by the total power loss as

$$\eta(\nu) = 1 - r(\nu) - \epsilon(\nu) - \beta(\nu) - \delta(\nu) \tag{55}$$

where $r(\nu)$ is the reflection spectrum, $\epsilon(\nu)$ is the dielectric absorption, $\beta(\nu)$ is the spillover spectrum, and $\delta(\nu)$ is the scattering spectrum. This quantity is integrated over the observation band when calculating the in-band power and signal attenuation referenced to the detector.

6.15 Auxiliary Detector Calculations

There are a few calculations that are carried out "under the hood" of the detector noise estimates. For more information about detector parameter definitions and dependencies, see Section 3.2. Ultimately, as described in Equations 20 and 31, the sensitivity calculation comes down to bolometer operating temperature, operating resistance, saturation power, amplifier noise-equivalent current, bath temperature, thermal conduction, and optical efficiency. In this section we overview how some of these inputs can be estimated from other parameters.

6.15.1 Pixel Elevation

In order to calculate the expected loading on each detector, we need to know the elevation of its detector pixel projected onto the sky. This is calculated as the sum of the telescope boresight elevation θ_{tel}, the camera boresight elevation with respect to the telescope boresight θ_{cam}, and the pixel elevation with respect to the camera boresight θ_{pix}. Therefore, the net elevation is given as

$$\theta_{\rm elv} = \theta_{\rm tel} + \theta_{\rm cam} + \theta_{\rm pix} \tag{56}$$

See section 6.15.1 for more details about defining the distribution of $\theta_{\rm pix}$.

6.15.2 Saturation Power

BoloCalc has an option to set the saturation power $P_{\rm sat}$ explicitly, or, as another option, as fraction of the optical power $P_{\rm opt}$, as defined in Equation 9. If the parameter **Saturation Power** is set to "NA," then $P_{\rm sat}$ is calculated as

$$P_{\rm sat} = f_{\rm psat} \times P_{\rm opt} \tag{57}$$

where $f_{\rm psat}$ represents the parameter **Psat Factor**.

6.15.3 Operating Temperature

BoloCalc has an option to set the operation temperature $T_{\rm oper}$ explicitly, or, as another option, as a fraction of the bath temperature $T_{\rm b}$. If the parameter **Tc** is set to "NA," then operating temperature is defined using the equation

$$T_{\rm oper} = f_{\rm oper} \times T_{\rm bath} \tag{58}$$

where $f_{\rm oper}$ represents the parameter **Tc Fraction**.

6.15.4 Fractional Readout Noise

BoloCalc has an option to calculate readout noise using the bolometer resistance $R_{\rm bolo}$, amplifier noise NEI, detector responsivity $S_{\rm I}$ and saturation power $P_{\rm sat}$, as shown in Equations 30 and 31. However, BoloCalc also offers an option to ignore the details of the readout architecture and set the readout noise as a fixed fractional noise increase to the quadrature sum of other noise sources (photon noise and thermal carrier noise). If either the parameter **Bolo Resistance** or **SQUID NEI** is defined as "NA," and if the parameter **Read Noise Frac** is not "NA," then readout noise is calculated via the equation

$$NEP_{\rm read} = \sqrt{(1 + \Delta_{\rm read})^2 - 1} \times \sqrt{NEP_{\rm ph}^2 + NEP_{\rm g}^2} \tag{59}$$

where $\Delta_{\rm read}$ represents the parameter **Read Noise Frac** and the noise contributions $NEP_{\rm ph}$ and $NEP_{\rm g}$ are calculated using Equations 19 and 20, respectively.

6.15.5 Number of Detectors

The number of total detectors in a given channel $N_{\rm det}$ is determined by the number of detectors per wafer $N_{\rm det/waf}$, the number of wafers per optics tube (or equivalently, per camera) $N_{\rm waf/OT}$, and the number of optics tubes per telescope $N_{\rm OT}$

$$N_{\rm det} = N_{\rm det/waf} \times N_{\rm waf/OT} \times N_{\rm OT} \tag{60}$$