BoloCalc User Manual
Version 0.3.0

Charles Hill, UC Berkeley

Last Updated: 2018-07-19
This document is still under construction and will be updated frequently

Contents

1 Quick Start

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.

1.1 System Requirements

BoloCalc runs on a Linux and Windows and has the following dependencies:

1. Python v2.7
2. Numpy v1.8 or above

My recommendation to acquire all useful Python packages is to install Anaconda2: https://www.anaconda.com/download/#macos Additionally, the following packages are needed to install auxiliary experiment and atmosphere data

- unzip
- wget

If on Linux, you can obtain these packages using apt-get or brew for Mac OSX:

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

If on Windows, I suggest downloading these packages from the GnuWin32

- http://gnuwin32.sourceforge.net/packages/unzip.htm
- https://eternallybored.org/misc/wget/

Once unzip is installed and wget is downloaded, you will need to add their executables to the PATH variable. For me, unxip.exe was stored at C:\Program Files (x86)\GnuWin32\bin, so I also moved wget.exe there. Then, I added this path to my PATH variable. For a tutorial about how to edit your PATH variable in Windows, see this example: https://www.howtogeek.com/118594/how-to-edit-your-system-path-for-easy-command-line-access/.

For the duration of this manual, I will use the Linux platform to show example code and executions. However, all shown examples **should** work just as well on Windows.

1.2 Code location

BoloCalc is stored on Charles Hill's GitHub account chill90/ within the repository chill90/BoloCalc. To clone a local copy of the calculator, make sure that you have Git installed on your machine (I've been using Git version 2.13.5) and run

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

All executables are stored at in the top-most directory

```
~/BoloCalc/
```

and all libraries are stored in the src/ directory

```
~/BoloCalc/src/
```

This code is still under construction and therefore is frequently being updated. Therefore, it is encouraged that the user git pull often, preferably before using the code at all.

1.3 Initialization

BoloCalc is downloaded from the GitHub repository only with source code. The example experiment data and atmosphere data needs to be installed from a separate server. To set up the auxiliary data, run init.sh

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

This will pull the directories Experiments/ExampleExperiment/ and src/atmFiles/ into your working tree.

1.4 Downloading Experiment Data

Additional experiment data is available at http://pbfs.physics.berkeley.edu/BoloCalc/ but is password protected. As of 2018-07-17, data for the following experiments is available for download:

- Simons Array
- Simons Observatory

Contact Charles Hill at chill@lbl.gov to obtain permissions for your experiment. If you're a member of SA or SO, you can find the permissions on the experiments' respective wiki pages:

- https://bolowiki.berkeley.edu/bin/view/Main/CharlesHill?topic=BoloCalc
- http://simonsobservatory.wikidot.com/sensitivity-calculators

1.5 Running mappingSpeed.py

The primary BoloCalc executable is called mappingSpeed.py, which calculates

- Optical power
- Noise-equivalent power (NEP) contributions from
  - Photon noise
  - Bolometer thermal carrier noise
  - Readout noise
- Per-detector noise-equivalent temperature (NET)
- Array NET
- Mapping speed
- Map depth

given some instrument configuration. mappingSpeed.py takes as input an experiment configuration directory, which is stored in BoloCalc/Experiments/ and has the following structure:

```
BoloCalc/
 |-- Experiements/
 |    |-- [Experiment1]/
 |    |    |-- [ExperimentDesign1]/
 |    |    |-- [ExperimentDesign2]/
 |    |    |-- ...
 |    |-- [Expierment2]/
 |    |-- ...
```

where the square brackets indicate the directories that can be named by the user.

To calculate the sensitivity an experiment, pass its [ExperimentDesign] directory to mappingSpeed.py. BoloCalc comes with an example experiment that can be used to get started:

```
BoloCalc$ python2.7 mappingSpeed.py Experiments/ExampleExperiment/V0/
```

By default, mappingSpeed.py will calculate one experiment realization, which should be shown by the status bar that appears upon execution. For information about how to tune the simulation parameters, see Section 4.1.

After it has finished running, mappingSpeed.py generates several tables that display the calculated quantities for the simulated instrument. These tables are stored in three files named sensitivity.txt, with each file located at a different level of the experiment directory structure and containing different information:

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

For more information about the contents of the sensitivity.txt files, see Section 5.1.

1.6 Running mappingSpeed_vary.py

The other BoloCalc executable is mappingSpeed_vary.py, which calculates mapping speed for a set of instrument configurations determined by a set of specified parameters varied with linear spacing over a specified range with and with a specified step size. To run mappingSpeed_vary.py, pass it the instrument configuration at the command line

```
BoloCalc$ python2.7 mappingSpeed_vary.py Experiments/ExampleExp/V0/
```

There are optional command line arguments that can be passed to the executable, which are defined below

- -vt: Vary the input parameters together instead of computing all possible combinations of parameters. The parameter array lengths, set by minimum, maximum, and step values, must be the same
- -fh [File Handle]: Choose a custom file handle for saving the parameter variations.

The input parameter variations are stored stored in BoloCalc/config/paramsToVary.txt. An example of the paramsToVary.txt is shown in Figure 1

```
# Telescope    | Camera     | Channel    | Optic          | Parameter      | Minimum    | Maximum    | Step Size
#-----------------------------------------------------------------------------------------------------------------
Tel            | Cam        | 1          | Primary        | Spillover      | 0.000      | 0.050      | 0.010
#-----------------------------------------------------------------------------------------------------------------
Tel            | Cam        | 1          | Aperture       | Temperature    | 1.000      | 5.000      | 1.000
```

Figure 1: Example of paramsToVary.txt

For information on how to define parameters to be varied by mappingSpeed_vary.py, see Section 4.2.

By default upon download, mappingSpeed_vary.py will calculate one experiment realization for each and every parameter combination (see Section 4.1 for details on simulation inputs). The status bar will show progression towards calculating all combinations of parameters defined over the ranges. mappingSpeed_vary.py saves three quantities[1] to files within the directory BoloCalc/Experiments/ExampleExp/V0/paramsToVary/. The file handle can be specified using the command line argument -fh [File Handle], or it if is not specified, it will default to

[1]More quantities can be added to this list upon request, or as needed.

[Tel]_[Cam]_[Channel]_[Param]. Below is the expected default file names that the data from the parameter-varied calculation will be saved to

- Optical power: mappingSpeedVary_Popt_[Tel]_[Cam]_[Channel]_[Param].txt
- Photon NEP: mappingSpeedVary_NEPph_[Tel]_[Cam]_[Channel]_[Param].txt
- Array NET: mappingSpeedVary_NETarr_[Tel]_[Cam]_[Channel]_[Param].txt

where [Tel]_[Cam]_[Channel]_[Param] is a file handle that contains the specifier for every parameter varied in the calculation. For example, if we looped over the following two parameters

- Telescope "Tel," camera "Cam," channel "1," optic "Mirror," Parameter "Spillover"
- Telescope "Tel," camera "Cam," channel "1," Parameter "Pixel Size"

the P_{opt} output file would have the filename

```
mappingSpeedVary_Popt_Tel_Cam_1_Mirror_Spillover_Tel_Cam_1_PixelSize.txt
```

For information about the contents of the mappingSpeed_vary.py output file, see Section 5.1.4. For more information about the output parameters, see Section 5.1.3.

1.7 Contact and Citation

Please send any comments, requests, questions, or bugs to Charles Hill at chill@lbl.gov. If you use this code for published results, please cite the SPIE proceeding from 2018 "BoloCalc: a sensitivity calculator for the design of Simons Observatory," which can be found at https://arxiv.org/abs/1806.04316.

2 Sensitivity Calculation

The primary function of BoloCalc is to import low-level instrument parameters and use them to estimate the instrument's noise-equivalent CMB 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 contributions from
   - Photon shot and wave noise
   - Bolometer thermal carrier noise
   - Readout 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 mapping speed (MS), a quantification of instrument noise in the power spectrum domain

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

2.1 Optical power

BoloCalc assumes an array of single-moded bolometers[2] within an instrument that is stationary in time. The propagation of optical power from the sky to the focal plane is represented by a one-dimensional chain of blackbody absorbers/emitters in thermal equilibrium. The power deposited on the detectors is then an analytic integral over the summation of each optical element's Planck

[2] If there is a demand, BoloCalc can be expanded to include other detector architectures, such as KIDs.

spectra modified by its frequency-dependent efficiency and emissivity. Explicitly, the optical power on a detector is given as

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

where ν is frequency, $p_i(\nu)$ is the power spectral density of optical element i referred to the detector input, the summation contains all $N_{\rm elem}$ 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 of all optics between it and the focal plane $[\eta_{i+1}(\nu), ..., \eta_{\rm N_{elem}}(\nu)]$, its emissivity $\epsilon_i(\nu)$, its spillover coefficient β_i, the effective temperature by which its spilled power is absorbed $T_{\beta;i}$, its scattering coefficient δ_i, and the effective temperature by which its scattered power is absorbed $T_{\delta;i}$

$$\begin{aligned} p_i\left(T_i,\, [\eta_{i+1}(\nu), ..., \eta_{\rm N_{elem}}(\nu)]\,,\, \epsilon_i(\nu),\, \beta_i,\, T_{\beta;i},\, \delta_i,\, T_{\delta;i},\, \nu\right) = \\ \prod_{j=i+1}^{N_{\rm elem}} \eta_j(\nu) \left[\epsilon_i(\nu)\, S(T_i, \nu) + \beta_i S(T_{\beta;i}, \nu) + \delta_i S(T_{\delta;i}, \nu)\right] \,. \end{aligned} \tag{2}$$

The power spectral density function $S(T, \nu)$ of the emitted and scattered power from each element is given by the Planck spectral density[3] for a diffraction-limited, single-moded polarimeter

$$S(T,\, \nu) = \frac{h\nu}{\exp\left[\frac{h\nu}{k_{\rm B}T}\right] - 1} \,. \tag{3}$$

2.2 Photon noise

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

$$\mathrm{NEP_{ph}} = \sqrt{2 \int_0^\infty \left[h\nu \sum_{i=1}^{N_{\rm elem}} p_i(\nu) + \Big(\sum_{i=1}^{N_{\rm elem}} p_i(\nu) \Big)^2 \right] B(\nu)\, {\rm d}\nu} \,. \tag{4}$$

There are two contributions to $\mathrm{NEP_{ph}}$. The first term represents shot noise $\mathrm{NEP_{shot}}$, which dominates when the photon occupation number $\ll 1$ (e.g. optical wavelengths) and is $\propto \sqrt{P_{\rm opt}}$. The second term represents wave noise $\mathrm{NEP_{wave}}$, which dominates when the photon occupation number is $\gg 1$ (e.g. radio wavelengths) and is $\propto P_{\rm opt}$. 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.

2.3 Bolometer thermal carrier noise

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

$$\mathrm{NEP_g} = \sqrt{4\, k_{\rm B} F_{\rm link} T_{\rm oper}^2 G} \,, \tag{5}$$

[3] `BoloCalc` can be expanded to incorporate non-thermal spectral densities if there is a demand for them.

where $T_{\rm oper}$ is the operating temperature of the bolometer, G is the thermal conductance from the absorbing element to the bath, and $F_{\rm link}$ is a numerical factor that depends on the link's thermal conduction index n. $F_{\rm link}$ is can be theoretically calculated via the equation

$$F_{\rm link} = \frac{n+1}{2n+3} \frac{1-(T_{\rm bath}/T_{\rm oper})^{2n+3}}{1-(T_{\rm bath}/T_{\rm oper})^{n+1}} \,, \tag{6}$$

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

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

$$G = P_{\rm sat}(n+1) \frac{T_{\rm oper}^{n}}{T_{\rm oper}^{n+1} - T_{\rm bath}^{n+1}} \;. \tag{7}$$

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

2.4 Readout noise

Modern CMB detectors are low-impedance, voltage-biased bolometers read out using superconducting quantum interference device (SQUID) transimpedance amplifiers. Therefore, amplifier-induced readout noise can be modeled as a noise-equivalent current (NEI), referred to the power at the detector by the inverse of the detector responsivity. For a voltage-biased bolometer operating with negative feedback and high loop gain $\mathcal{L} \gg 1$, responsivity is $\approx 1/V_{\rm bias}$, and readout NEP is given by

$$\mathrm{NEP_{read}} = \sqrt{P_{\rm elec}\, R_{\rm bolo}} \times \mathrm{NEI} \,, \tag{8}$$

where the bias power $P_{\rm elec} = P_{\rm sat} - P_{\rm opt}$ and $R_{\rm bolo}$ is the bolometer operating resistance.

2.5 Noise-equivalent CMB temperature

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

$$\mathrm{NET_{det}} = M \frac{\sqrt{\mathrm{NEP_{ph}^2} + \mathrm{NEP_g^2} + \mathrm{NEP_{read}^2}}}{\sqrt{2}\,(\mathrm{d}P/\mathrm{d}T_{\rm CMB})} \,, \tag{9}$$

where the M is a "margin" applied to the expected per-detector NET, and the $\sqrt{2}$ arises due to a unit conversion from output bandwidth $1/\sqrt{\mathrm{Hz}}$ to integration time $\sqrt{\mathrm{s}}$. The conversion factor from optical power to CMB temperature is defined as

$$\mathrm{d}P/\mathrm{d}T_{\rm CMB} = \xi \int_0^\infty \left[\prod_{i=1}^{N_{\rm elem}} \eta_i(\nu) \frac{1}{k_{\rm B}} \left(\frac{h\nu}{T_{\rm CMB}\, (\exp{[h\nu/k_{\rm B}T_{\rm CMB}]} - 1)} \right)^2 \exp{[h\nu/k_{\rm B}T_{\rm CMB}]} \right] B(\nu)\mathrm{d}\nu \,, \tag{10}$$

and has units of $\mathrm{W/K_{CMB}}$. ξ is an optical coupling factor that quantifies SNR degradation associated with image degradation at the focal plane.

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

$$\mathrm{NET_{arr}} = \frac{\mathrm{NET_{det}}}{\sqrt{Y\, N_{\mathrm{det}}}}\,\Gamma\ , \tag{11}$$

where N_{det} is the number of deployed bolometers, Y is the yield, and Γ is a factor ≥ 1 that quantifies the degree to which white noise is correlated between detector pixels on the focal plane.

2.6 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

$$\sigma_S = \sqrt{\frac{4\pi f_{\mathrm{sky}}\,\mathrm{NET^2_{arr}}}{\eta_{\mathrm{obs}}\,t_{\mathrm{obs}}}}\left(\frac{10800\ \mathrm{arcmin}}{\pi}\right)\ , \tag{12}$$

where f_{sky} the fraction of sky observed, t_{obs} is the observation time of the experiment, and η_{obs} is the observation efficiency.

2.7 Mapping speed

While NET is useful for estimating noise in the time and map domain, mapping speed (MS) is useful for describing instrument performance in the power spectrum domain

$$\mathrm{MS} = \frac{1}{\mathrm{NET^2_{arr}}} \tag{13}$$

and has units of $\mathrm{K^{-2}\,s^{-1}}$.

2.8 Auxiliary Calculations

There are a few calculations that are carried out "under the hood" if specific sets of parameters are defined in the input files (for more information about parameter definitions and dependencies, see . Ultimately, as shown in Equation , the calculate needs a temperature, emissivity, and efficiency in order to calculate the power from each element.

2.8.1 Synchrotron Emission

Synchrotron power in units of W/Hz is given by the following equation

$$p_{\mathrm{synch}}(\nu) = A_{\mathrm{s}}\nu^{n_{\mathrm{s}}} \tag{14}$$

where A_{s} is the amplitude, ν is the observation frequency, and n_{s} is the spectral index. This quantity is integrated over the observation band when calculating the in-band optical power and signal attenuation referenced to the detector.

2.8.2 Dust Emission

Thermal dust power in units of W/Hz is given by the following equation

$$p_{\rm dust}(\nu) = A_{\rm d} \left(\frac{\nu}{\nu_{\rm d}} \right)^{n_{\rm d}} S(T_{\rm d}, \nu) \tag{15}$$

where $A_{\rm d}$ is the amplitude, ν is the observation frequency, $\nu_{\rm d}$ is the pivot frequency, $n_{\rm d}$ is the power law index, and $T_{\rm d}$ is the effective blackbody temperature. This quantity is integrated over the observation band when calculating the in-band optical power and signal attenuation referenced to the detector.

2.8.3 Atmosphere Emission

BoloCalc utilizes atmospheric simulations of the atmosphere at the Atacama and South Pole sites[4] generated by the AM atmospheric modeling code[5], which uses data from the MERRA-2 meteorological reanalysis[6] as input. The output from AM produces results consistent with measured sky loading in existing Atacama experiments. The range of input elevations handled by BoloCalc is from 20–90 deg, and the range of input PWV values is from 0–8 mm.

2.8.4 Cold Stop Spillover Efficiency

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

$$\eta_{\rm stop}(\lambda) = 1 - \exp\left[-\frac{\pi^2}{2} \left(\frac{D}{F \lambda w_{\rm f}} \right)^2 \right], \tag{16}$$

where D is the pixel diameter, F is the F/# at the focal plane, 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.

2.8.5 Dielectric Absorption/Emission

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

$$\epsilon(\lambda) = 1 - \exp\left[-2\pi t n \tan\delta / \lambda \right] \tag{17}$$

where t is the thickness of the 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 calcuating the in-band optical power and signal attenuation referenced to the detector.

[4]More site profiles exist and can be added if requested.
[5]`https://www.cfa.harvard.edu/~spaine/am/`
[6]`https://gmao.gsfc.nasa.gov/reanalysis/MERRA-2/`

2.8.6 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

$$\epsilon(\nu) = 4\sqrt{\frac{\pi\nu\mu_0}{\sigma_c}}\frac{1}{Z_0} \tag{18}$$

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 optical power and signal attenuation referenced to the detector.

2.8.7 Ruze Scattering

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

$$\delta(\lambda) = \exp\left[\left(\frac{4\pi\sigma_r}{\lambda}\right)^2\right] \tag{19}$$

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 optical power and signal attenuation referenced to the detector.

2.8.8 Efficiency

Efficiency can be defined explicitly for any optic or detector by passing a band file (see Section **??** and **??** for more details about band files). However, if a band file is not passed, efficiency is derived using the following equation

$$\eta(\nu) = 1 - r(\nu) - \epsilon(\nu) - \beta - \delta \tag{20}$$

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

2.8.9 Pixel Elevation

In order to calculate the expected loading on each detector, we need to know the elevation of its projected detector pixel on the sky. This is calculated as the sum of the telescope boresight elevation, the camera boresight elevation w.r.t. telescope boresight, and the pixel elevation w.r.t. camera boresight. Therefore, the pixel elevation is given by

$$\theta_{\text{elv}} = \theta_{\text{tel}} + \theta_{\text{cam}} + \theta_{\text{pix}} \tag{21}$$

2.8.10 Saturation Power Factor

`BoloCalc` has an option to set saturation power P_{sat} (see Section 2.3 for more details) as a fraction of the optical power (as defined in Equation 1) if the parameter "Saturation Power" is set to "NA"

$$P_{sat} = P_{opt} \times f_{\text{psat}} \tag{22}$$

where $f_{\rm psat}$ is the saturation power factor.

2.8.11 Operating Temperature Fraction

`BoloCalc` has an option to set the operating temperature $T_{\rm oper}$ (see Section 2.3 for more details) as a fraction of the camera base temperature $T_{\rm base}$ if the parameter "Tc" is set to "NA"

$$T_{\rm oper} = T_{\rm bath} \times f_{\rm oper} \tag{23}$$

where $f_{\rm oper}$ is the operating temperature fraction.

2.8.12 Fractional Readout Noise

`BoloCalc` has an option to set readout noise not explicitly but instead as a fractional increase on the total NEP. If either $R_{\rm bolo}$ or NEI from Equation 8 are "NA," and if the parameter "Read Noise Frac" is not "NA" in the `channels.txt` file (see Section 3.5.3 for more detail), then the following equation is used to calculate readout noise:

$$NEP_{\rm read} = \sqrt{(1+\Delta_{\rm read})^2 - 1} \times \sqrt{NEP_{\rm ph}^2 + NEP_{\rm g}^2} \tag{24}$$

where $\Delta_{\rm read}$ is the fractional increase in the total NEP due to readout noise.

2.8.13 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 $N_{\rm OT}$.

$$N_{\rm det} = N_{\rm det/waf} \times N_{\rm waf/OT} \times N_{\rm OT} \tag{25}$$

3 Defining an 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 parent-child structure shown in Figure 2 and is built with four layers: experiments, telescopes, cameras, and channels, which are defined in Tab. 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.

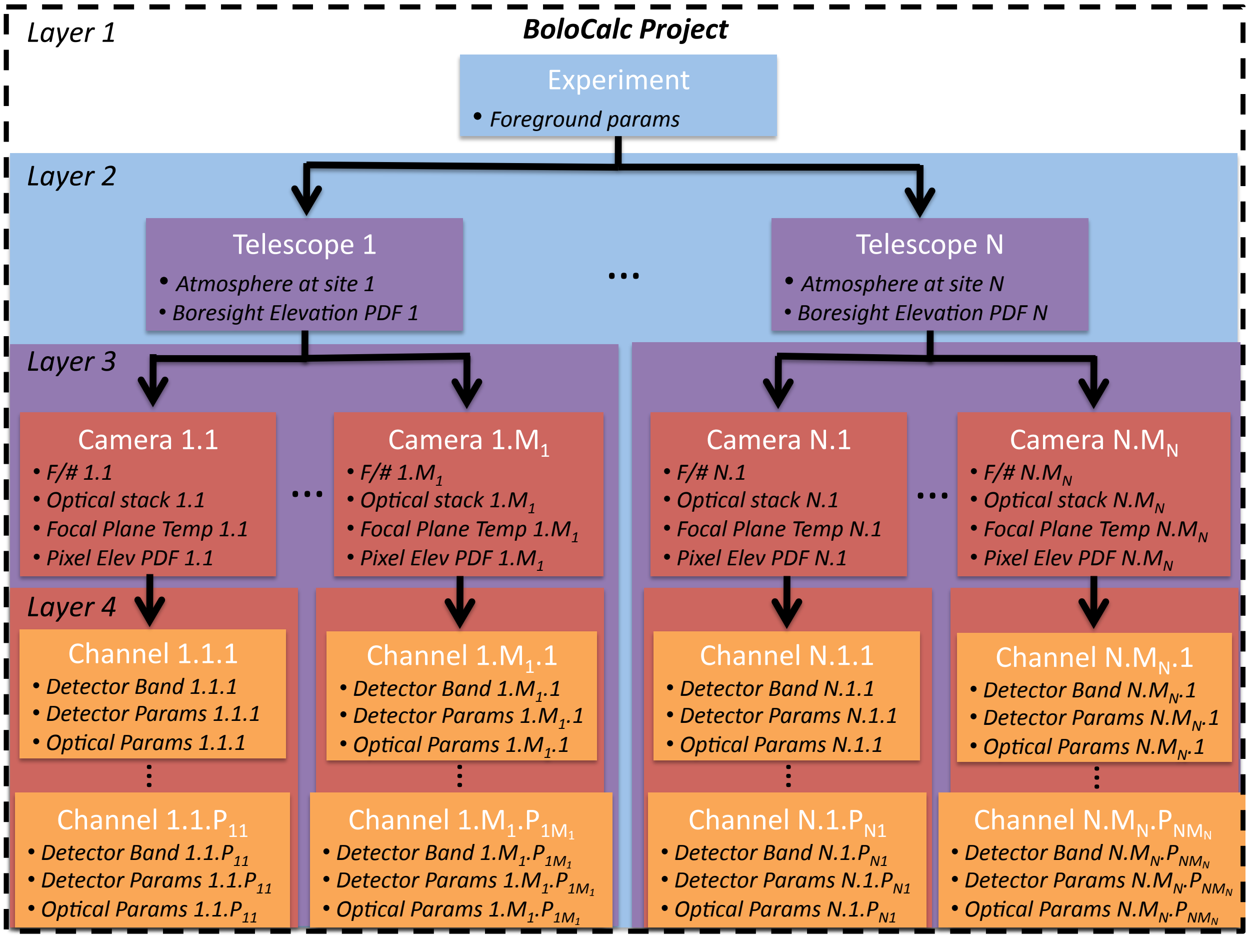

Figure 2: Layout of a BoloCalc project.

3.1 Experiment Directory Layout

The experiment directory is meant to be easily generalizable to various kinds of experiments and telescopes. For information about how experiments are stored within the BoloCalc/Experiments/ directory, see Section 1.5. Below is an overview of the file structure within an experiment directory.

Each layer of an experiment has several parameters that define the input experiment. In this section, we go through them by input file.

```
BoloCalc/Experiements/[Experiment Name]/[Experiment Design]
 |-- config
 |     |--foregrounds.txt
 |--Telescope1/
 |     |-- config/
 |     |     |-- telescope.txt
 |     |     |-- elevation.txt
 |     |     |-- [Custom Atmospheric Profile] (optional)
 |     |-- Camera1/
 |     |      |-- config/
 |     |            |-- camera.txt
 |     |            |-- channels.txt
 |     |            |-- optics.txt
 |     |            |-- elevation.txt
```

```
  |     |           |-- Bands/
  |     |               |-- [Optic Band File 1] (optional)
  |     |               |-- [Optic Band File 2] (optional)
  |     |               |-- ...
  |     |               |-- [Detector Band File 1] (optional)
  |     |               |-- [Detector Band File 2] (optional)
  |     |        |-- ...
  |     |-- Camera2/
  |     |-- ...
  |-- Telescope2/
  |-- ...
```

As stated in Section 3, an experiment definition has four layers:

1. Experiment (Layer 1)
   - Composed of (multiple) telescopes
   - All telescopes in a given experiment see the same galactic foregrounds
2. Telescopes (Layer 2)
   - Composed of (multiple) cameras
   - All cameras in a given telescope see the same atmosphere
   - All cameras in a given telescope have the same scan strategy
3. Cameras (Layer 3)
   - Composed of (multiple) frequency channels
   - All frequency channels in a given camera have the same camera parameters
   - All frequency channels in a given camera see the same optical chain
4. Channels (Layer 4)
   - Each channel has independent detector and optical element parameters

These layers allow telescopes to added to an experiment in an arbitrary location, each with arbitrary cameras, each with arbitrary frequency channels.

3.2 Parameter Definitions

All parameters in `BoloCalc` are defined with a central value and an error bar. Therefore, all parameter variation is assumed to have an underlying Gaussian distribution, and all parameters are assumed to be uncorrelated.

Parameters defined `foregrounds.txt`, `telescope.txt`, `camera.txt`, and `channels.txt` have the following format:

```
Mean +/- Std Dev
```

The following formatting rules should be followed for the code to work robustly and the tables to render neatly:

1. All central values and standard deviations should have five characters. For instance, `1.000` and `10.00` are valid, while `10.000` is not.

2. All central values and standard deviations should be separated by ` +/- `, noting the spaces on either side of the $+/-$ sign.

Parameters defined in `optics.txt` are defined for multiple bands in multi-chroic cameras and therefore follow a different format:

```
[Mean for Band 1, Mean for Band 2, ...] +/- [Std Dev for Band 1, Std Dev for
    Band 2, ...]
```

In a similar way as with the single values, the following formatting rules should be followed for the code to work robustly and the tables to render neatly:

1. All central values and standard deviations should have five characters. For instance, `1.000` and `10.00` are valid, while `10.000` is not.
2. All arrays of central values and arrays of standard deviations should be separated by ` +/- `, noting the spaces on either side of the +/− sign

3.3 Experiment Parameters (Layer 1)

The experiment is comprised of an arbitrary set of telescopes, which are assumed to see the same 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 `[Experiment Name]/[Experiment Version]/config/foregrounds.txt`, the file that defines the galaxy through which the telescope must observe the CMB. This galactic parameters will be the same for all telescopes.

- **Synchrotron Spec Index**
  - *Definition*: "n_{s}" in Equation 14
  - *Description*: Spectral index of the synchrotron power law spectrum
  - *Allowed Values*: Floating point value between $(-\infty, +\infty]$
  - *Default Value*: 2×10^{-3} +/- 0×10^{-3}
- **Synchrotron Amplitude**
  - *Definition*: "A_{s}" in Equation 14
  - *Description*: Amplitude of the synchrotron power law spectrum
  - *Allowed Values*: Floating point value between $(0.0, +\infty)$ W/Hz
  - *Default Value*: 6×10^{3} +/- 0×10^{3} W/Hz
- **Dust Temperature**
  - *Definition*: "T_{d}" in Equation 15
  - *Description*: Dust modified blackbody temperature
  - *Allowed Values*: Floating point value between $(0.0, +\infty)$ K
  - *Default Value*: 19.70 +/- 0.00 K
- **Dust Spec Index**
  - *Definition*: "n_{d}" in Equation 15
  - *Description*: The spectral index of the dust modified blackbody spectrum
  - *Allowed Values*: Floating point value between $(-\infty, +\infty)$
  - *Default Value*: 1.50 +/- 0
- **Dust Amplitude**
  - *Definition*: "A_{d}" in Equation 15
  - *Description*: Amplitude of the dust modified blackbody spectrum
  - *Allowed Values*: Floating point value between $(0.0, +\infty)$
  - *Default Value*: 2×10^{-3} +/- 0×10^{-3}
- **Dust Scale Frequency**
  - *Definition*: "ν_{d}" in Equation 15
  - *Description*: Scale frequency for the dust modified blackbody spectrum
  - *Allowed Values*: Floating point value between $(0.0, +\infty)$

- *Default Value*: 353.0 +/- 0.0 GHz

3.4 Telescope Parameters (Layer 2)

The following sections overview the input parameters for each of the telescope input files.

3.4.1 `telescope.txt`

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

- **Site**
  - *Description*: Site at which the telescope observes, which defines the atmospheric conditions.
  - *Allowed Values*: Either "Atacama" or "Pole"
  - *Default Value*: "Atacama"
- **Elevation**
  - *Definition*: θ_{tel} defined in 21
  - *Description*: Elevation at which the telescope observes.
  - *Allowed Values*: [20.0, 90.0] deg above the horizon
  - *Default Value*: 50.0 +/- 0.0 deg
  - If "NA," the elevation distribution defined in `Telescope/config/elevation.txt`, will be assumed.
- **PWV**
  - *Description*: Precipitable water vapor of the atmosphere through which the telescope observes.
  - *Allowed Values*: [0.0, 8.0] mm
  - *Default Value*: 1.0 +/- 0.0 mm
  - If "NA," the PWV distribution defined in `Telescope/config/pwv.txt`, will be assumed.
- **Observation Time**
  - *Definition*: t_{obs} in Equation 12
  - *Description*: How long this telescope will observe
  - *Allowed Values*: Floating point value between (0.0, $+\infty$) years
  - *Default Value*: 3.0 +/- 0.0
- **Sky Fraction**
  - *Definition*: f_{sky} in Equation 12
  - *Description*: What fraction of the sky this telescope will observe
  - *Allowed Values*: Floating point value between (0.0, 1.0]
  - *Default Value*: 0.7 +/- 0.0
- **Observation Efficiency**
  - *Definition*: η_{obs} in Equation 12
  - *Description*: What fraction of the observation time will the telescope be actually observing
  - *Allowed Values*: Floating point value between (0.0, 1.0]
  - *Default Value*: 0.8 +/- 0.0
- **NET Margin**
  - *Definition*: M in Equation 9
  - *Description*: Agnostic factor which multiplies the detector NETs for this telescope
  - *Allowed Values*: Floating point value between (0.0, $+\infty$)

- *Default Value*: 1.0 +/- 0.0

3.4.2 `elevation.txt`

`elevation.txt` is an optional file that defines the elevation probability distribution function (PDF), which is determined by the scan strategy of the telescope. The first column is the elevation above the horizon in degrees, and the second column is the fraction of time spent at that elevation. This file is only used if the parameter "Elevation" in `Telescope/config/telescope.txt` is defined to be "NA" (see Section 3.4.1 for more information about `telescope.txt`). The probabilities should add up to one, and if they do not, the fraction will be determined by the provided probability divided by the sum of the provided probabilities.

Below is a made-up example for `elevation.txt`. Notice that each row is separated by a commented line of hyphens, while each column is separated by a vertical bar.

```
#*****Elevation Distribution******
#
Elevation      | Probability
#----------------------------
[deg]          | NA
#----------------------------
30.0           | 0.100
#----------------------------
40.0           | 0.100
#----------------------------
50.0           | 0.300
#----------------------------
60.0           | 0.400
#----------------------------
70.0           | 0.100
#----------------------------
```

3.4.3 `pwv.txt`

`pwv.txt` is an optional file that defines the precipitable water vapor (PWV) probability distribution function (PDF), which is determined by the weather conditions at the observation site. The first column is the PWV in mm, and the second column is the fraction of time spent at that PWV value. This file is only used if the parameter "PWV" in `Telescope/config/telescope.txt` is defined to be "NA" (see Section 3.4.1 for more information about `telescope.txt`). The probabilities should add up to one, and if they do not, the fraction will be determined by the provided probability divided by the sum of the provided probabilities.

Below is a made-up example for `pwv.txt`. Notice that each row is separated by a commented line of hyphens, while each column is separated by a vertical bar.

```
#*****Site PWV Distribution******
#
PWV            | Probability
#----------------------------
[mm]           | NA
#----------------------------
0.200          | 0.100
#----------------------------
0.400          | 0.100
#----------------------------
0.600          | 0.200
```

```
#---------------------------
0.800         | 0.200
#---------------------------
1.000         | 0.100
#---------------------------
1.200         | 0.100
#---------------------------
1.400         | 0.100
#---------------------------
1.600         | 0.100
#---------------------------
```

3.4.4 `atm.txt`

There is an option to override the code's handling of the atmosphere by providing a custom atmospheric profile. The atmosphere file must be in `[Telescope Name]/config/`, and the filename must have the form `atm_[Elevation]deg_[PWV]um.txt`. For example, a valid filename is `atm_60deg_1000um.txt`.

The atmosphere file should have four columns, delimited by white space:

1. Column 1: Frequency [GHz]
2. Column 2: Optical Depth
3. Column 3: Planck Temperature [K]
4. Column 4: Transmission

```
| Frequency [GHz] | Optical Depth | Planck Temp [K] | Transmission |
```

Note that the temperature is in Planck units, not Rayleigh-Jeans. The passed frequency range in the atmosphere file needs to be large enough to cover the frequency bands observed by the telescope—which are defined in the "channels.txt" and "Bands/" files at the camera directory level (see Section 3.5.3 and Section for more details)—with at least a 15% buffer on the bandwidths of the highest and lowest bands. For instance, if your detector bands range from 240 to 300 GHz, your atmosphere file should have values that go from at least 205 to 345 GHz.

Below is an example of an atmospheric profile generated using the AM simulation code and a configuration file designed using the 10-year MERRA dataset for the Chajnantor Observatory.

```
1.0000000e+01 3.664259e-03 3.589557e+00 9.963424e-01
1.0020000e+01 3.667538e-03 3.590339e+00 9.963392e-01
1.0040000e+01 3.671011e-03 3.591163e+00 9.963357e-01
1.0060000e+01 3.674741e-03 3.592043e+00 9.963320e-01
1.0080000e+01 3.678826e-03 3.593001e+00 9.963279e-01
1.0100000e+01 3.683414e-03 3.594069e+00 9.963234e-01
1.0120000e+01 3.688744e-03 3.595301e+00 9.963181e-01
1.0140000e+01 3.695220e-03 3.596788e+00 9.963116e-01
1.0160000e+01 3.703583e-03 3.598698e+00 9.963033e-01
1.0180000e+01 3.715317e-03 3.601374e+00 9.962916e-01
1.0200000e+01 3.733976e-03 3.605660e+00 9.962730e-01
...
```

3.5 Camera Parameters (Layers 3 and 4)

In the following sections, we will overview the input parameters for each of the camera input files.

3.5.1 `camera.txt`

`camera.txt` defines parameters that are the same for all frequency bands observed with this camera. The file must be located within `[Telescope Name]/[Camera Name]/config/`.

- **Boresight Elevation**
  - *Definition*: θ_{cam} in Equation 21.
  - *Description*: The elevation of the camera boresight with respect to the telescope boresight.
  - *Allowed Values*: Floating point value between (-40.0, 40.0] deg
  - *Default Value*: 0.0 +/- 0.0 deg
- **Optical Coupling**
  - *Definition*: ξ in Equation 10.
  - *Description*: A parameter that quantifies the SNR degradation associated with how well the far-field image couples to the detector pixels on the focal plane.
  - *Allowed Values*: Floating point value between (0.0, 1.0]
  - *Default Value*: 1.0 +/- 0.0
- **F-number**
  - *Definition*: F in Equation 16.
  - *Description*: The focal ratio at the focal plane
  - *Allowed Values*: Floating point value between (0.0, $+\infty$)
  - *Default Value*: 2.5 +/- 0.0
- **Bath Temp**
  - *Definition*: T_{bath} in Equation 6.
  - *Description*: The temperature of the focal plane
  - *Allowed Values*: Floating point value between (0.0, $+\infty$) K
  - *Default Value*: 0.100 +/- 0.000 K

3.5.2 `elevation.txt`

`elevation.txt` is an optional file that defines the elevation distribution with respect to the camera boresight of pixels on the focal plane. This file sets the value for θ_{elv} defined in equation 21.

This pixel elevation distribution file has the exact same structure as the file `Telescope/config/elevation.txt`, which instead defines the scan strategy and is described in section 3.4.3. The first column is the elevation above the horizon in degrees, and the second column is the fraction if time spent at that elevation.

This file is only used if the parameter "Elevation" in `Telescope/config/elevation.txt` is defined to be "NA" (see Section 3.4.1 for more information about `telescope.txt`).

The probabilities should add up to one, and if they don not, the fraction will be determined by the provided probability divided by the sum of the provided probabilities.

Below is a made-up example of correct formatting for `elevation.txt`. Notice that each row is separated by a commented line of hyphens, while each column is separated by a vertical bar.

```
#*****Elevation Distribution******
#
Elevation      | Probability
#---------------------------
[deg]          | NA
#---------------------------
30.0           | 0.100
#---------------------------
40.0           | 0.100
```

```
#---------------------------
50.0           | 0.300
#---------------------------
60.0           | 0.400
#---------------------------
70.0           | 0.100
#---------------------------
```

3.5.3 `channels.txt`

`channels.txt` defines the parameters of the frequency bands and detectors that are observing in this camera. The file must be located within `[Telescope Name]/[Camera Name]/config/`.

- **Band ID**
  - *Description*: The identification number for this band.
  - *Allowed Values*: An integer value between $[1, +\infty)$
  - *Default Value*: 1
  - Cannot have two bands with same Band ID on the same pixel
- **Pixel ID**
  - *Description*: The identification number for the pixel this band observes from
  - *Allowed Values*: An integer value between $[1, +\infty)$
  - *Default Value*: 1
  - Cannot have two bands with same Band ID on the same pixel
- **Band Center**
  - *Definition*: ν_{c} in Equation 26.
  - *Description*: The central frequency for this band.
  - *Allowed Values*: A floating point value between $(0.0, +\infty)$ GHz
  - *Default Value*: [90, 150] +/- [0, 0] GHz
- **Fractional BW**
  - *Definition*: f_{BW} in Equation 27.
  - *Description*: Fractional arithmetic bandwdith for this band
  - *Allowed Values*: A floating point value between (0.0, 2.0]
  - *Default Value*: [0.350, 0.275] +/- [0.000, 0.000]
- **Pixel Size**
  - *Definition*: D in Equation 16.
  - *Description*: Size of the pixel via which this band is observing
  - *Allowed Values*: A postive floating point value between $(0.0, +\infty)$ mm
  - *Default Value*: 6.8 +/- 0.0 mm
- **Num Det per Wafer**
  - *Definition*: $N_{\mathrm{det/waf}}$ in Equation 25
  - *Description*: Number of detectors per wafer within this band
  - *Allowed Values*: An integer value between $[1, +\infty)$
  - *Default Value*: 542
- **Num Waf per OT**
  - *Definition*: $N_{\mathrm{waf/OT}}$ in Equation 25.
  - *Description*: Number of wafers with this band per camera.
  - *Allowed Values*: An integer value between $[1, +\infty)$
  - *Default Value*: 7
- **Num OT**
  - *Definition*: N_{OT} in Equation 25.

- *Description*: Number of cameras in this telescope observing at this frequency.
- *Allowed Values*: An integer value between [1, $+\infty$)
- *Default Value*: 1

- **Waist Factor**
  - *Definition*: w_{f} in Equation 16.
  - *Description*: The ratio of the Gaussian beam waist coming from the pixel aperture to the pixel aperture diameter for this band.
  - *Allowed Values*: A floating point value between [2.0, $+\infty$)
  - *Default Value*: 3.0 +/- 0.0
- **Det Eff**
  - *Definition*: η_{det} in Equation 26.
  - *Description*: Efficiency of the detector from the entry point of the pixel aperture to the bolometer.
  - *Allowed Values*: A floating point value between (0.0, 1.0]
  - *Default Value*: 0.700 +/- 0.000
- **Psat**
  - *Definition*: P_{sat} in Equation 7.
  - *Description*: Saturation power of the bolometer.
  - *Allowed Values*: A floating point value between (0.0, $+\infty$] pW
  - *Default Value*: "NA"
  - If "NA," use Equation 22 to calculate P_{sat}
- **Psat Factor**
  - *Definition*: f_{psat} in Equation 22.
  - *Description*: Safety factor used to caculate saturation power from optical power
  - *Allowed Values*: A floating point value between (0.0, $+\infty$)
  - *Default Value*: 3.0 +/- 0.0
  - If "Psat" is not "NA," this parameter is ignored.
- **Carrier Index**
  - *Definition*: n in Equations 7 and 6.
  - *Description*: Thermal carrier index for bolometer leg conductivity
  - *Allowed Values*: A floating point value between (0.0, $+\infty$)
  - *Default Value*: 2.7 +/- 0.0
- **Tc**
  - *Definition*: T_{oper} in Equations 5, 6, and 7.
  - *Description*: Bolometer operating temperature
  - *Allowed Values*: A floating point value between (0.0, $+\infty$) K
  - *Default Value*: 0.165 +/- 0.000 K
  - If "NA," use Equation 23 to calculate operating temperature.
- **Tc Fraction**
  - *Definition*: f_{oper} in Equation 23.
  - *Description*: Factor used to calculate transition temperature from the bath temperature
  - *Allowed Values*: A floating point value between (1.0, $+\infty$)
  - *Default Value*: "NA"
  - If "Tc" is not "NA," this parameter is ignored.
- **Yield**
  - *Definition*: Y in Equation 11.
  - *Description*: Fraction of deployed detectors that are observing. This is also commonly referred to as "end-to-end" yield.

- *Allowed Values*: A floating point value between (0.0, 1.0]
- *Default Value*: 0.7 +/- 0.0

- **SQUID NEI**
  - *Definition*: NEI in Equation 8.
  - *Description*: SQUID noise equivalent current.
  - *Allowed Values*: A floating point value between (0.0, $+\infty$) pA/$\sqrt{\mathrm{Hz}}$
  - *Default Value*: "NA"
  - If "NA," use Equation 24 to calculate readout noise $\mathrm{NEP}_{\mathrm{read}}$
- **Bolo Resistance**
  - *Definition*: R_{bolo} in Equation 8.
  - *Description*: Bolometer operating resistance.
  - *Allowed Values*: A floating point value between (0.0, $+\infty$) Ω
  - *Default Value*: "NA"
  - If "NA," use Equation 24 to calculate readout noise $\mathrm{NEP}_{\mathrm{read}}$
- **Read Noise Frac**
  - *Definition*: Δ_{read} in Equation 24.
  - *Description*: Fraction of the total NEP that is due to readout noise.
  - *Allowed Values*: A floating point value between [0.0, $+\infty$)
  - *Default Value*: 0.1 +/- 0.0
  - If "SQUID NEI" and "Bolo Resistance" are not "NA," this parameter is ignored.

3.5.4 `optics.txt`

`optics.txt` defines the parameters the frequency bands and detectors that are observing in this camera. The file must be located within `[Telescope Name]/[Camera Name]/config/`.

- **Element**
  - *Description*: Name of optical element
  - *Allowed Values*: Any string
  - *Default Value*: "Element"
  - **RESERVED NAMES**: "Primary," "Mirror," "Aperture," and "Stop" trigger additional calculations and therefore must be used purposefully
- **Temperature**
  - *Definition*: T_i in Equation 2.8.
  - *Description*: Temperature of the optical element
  - *Allowed Values*: An floating point value between (0.0, $+\infty$) K
  - *Default Value*: 4 +/- 0 K
- **Absorption**
  - *Definition*: ϵ_i in Equation 1.
  - *Description*: Fractional power attenuation/emission due to absorption within the optical element
  - *Allowed Values*: A floating point value between [0.0, 1.0)
  - *Default Value*: 0.000 +/- 0.000
  - If "NA" ...
    - If "Mirror" or "Primary" is in "Element," Equation 18 is used to calculate ohmic losses in the reflective optical element.
    - If "Mirror" or "Primary" is not in "Element," Equation 17 is used to calculate dielectric loss in the refractive optical element.
- **Reflection**

- *Definition*: r in Equation 20.
- *Description*: Fractional power lost due to reflection at the optical element.
- *Allowed Values*: A floating point value between [0.0, 1.0)
- *Default Value*: 0.000 +/- 0.000
- If "NA," Equation 20 is used to calculate reflection loss

- **Thickness**
  - *Definition*: t in Equation 17
  - *Description*: Thickness of the optical element
  - *Allowed Values*: A floating point value between $(0.0, +\infty)$ mm
  - *Default Value*: "NA"
  - If "Mirror" or "Primary" is in "Element," this parameter is ignored
  - If "Absorption" is not "NA," this parameter is ignored
- **Index**
  - *Definition*: n in Equation 17.
  - *Description*: Index of refraction of the optical element.
  - *Allowed Values*: A floating point value between $[1.0, +\infty)$
  - *Default Value*: "NA"
  - If "Mirror" or "Primary" is in "Element," this parameter is ignored
  - If "Absorption" is not "NA," this parameter is ignored
- **Loss Tangent**
  - *Definition*: $\tan\delta$ in Equation 17.
  - *Description*: Loss tangent of the optical element.
  - *Allowed Values*: A floating point value between $[0.0, +\infty)$
  - *Default Value*: "NA"
  - If "Mirror" or "Primary" is in "Element," this parameter is ignored
  - If "Absorption" is not "NA," this parameter is ignored
- **Conductivity**
  - *Definition*: σ_{c} in Equation 18.
  - *Description*: Electrical conductivity of the optical element.
  - *Allowed Values*: A floating point value between $(0.0, +\infty)$
  - *Default Value*: "NA"
  - If "Mirror" or "Primary" is not in "Element," this parameter is ignored
  - If "Absorption" is not "NA," this parameter is ignored
- **Scatter Frac**
  - *Definition*: δ_i in Equation 2.8.
  - *Description*: Fractional power lost due to scattering.
  - *Allowed Values*: A floating point value between [0.0, 1.0]
  - *Default Value*: "NA"
  - If "NA," Equation 19 is use to calculate scattering loss.
- **Surface Rough**
  - *Definition*: σ_{r} in Equation 19.
  - *Description*: Surface roughness of the optical element.
  - *Allowed Values*: A floating point value between $[0.0, +\infty)$
  - *Default Value*: "NA"
  - If "Scatter Frac" is not "NA," this parameter is ignored.
  - If "NA" and "Scatter Frac" is "NA," scattering is assumed to be zero.
- **Scatter Temp**
  - *Definition*: $T_{\delta;i}$ in Equation 2.8.

- *Description*: The effective temperature that the scattered power lands on.
- *Allowed Values*: A floating point value between [0.0, $+\infty$)
- *Default Value*: "NA"
- If "NA," scattered power is assumed to land at "Temperature," the temperature of the optical element itself

- **Spillover**
  - *Definition*: β_i in Equation 2.8.
  - *Description*: Fractional power that spills over the optical element.
  - *Allowed Values*: A floating point value between [0.0, 1.0)
  - *Default Value*: "NA"
  - If "NA," spillover is assumed be zero
- **Spillover Temp**
  - *Definition*: $T_{\beta;i}$ in Equation 2.8.
  - *Description*: The effective temperature that the spilled power lands on.
  - *Allowed Values*: A floating point value between [0.0, $+\infty$)
  - *Default Value*: "NA"
  - If "NA," spillover is assumed to land at "Temperature," the temperature of the element itself

3.6 Custom Bands

By default, `BoloCalc` assumes top hat bands for all detectors and optical elements, whose height is determined soley by the mean and standard deviation of the end-to-end optical efficiency. However, custom bands can be input for any optical element.

The general file format for all band files is the same:

```
| Frequency [GHz] | Mean Efficiency | Standard Deviation (optional) |
```

Each column is separated by a vertical bar, and the final column, which contains the error bars, can be omitted. If only two columns are present, all stand deviations are assumed to be zero.

An example of a made-up detector bandpass text file, which is space-delimited, is shown below:

```
70.     0.000   0.000
75.     0.500   0.100
80.     0.700   0.100
85.     0.650   0.100
90.     0.700   0.100
100.    0.800   0.100
105.    0.600   0.100
110.    0.700   0.100
115.    0.500   0.100
120.    0.000   0.000
```

3.6.1 Detectors

Detector band files are stored in

```
[Telescope Name]/[Camera Name]/config/Bands/Detectors/
```

The band is identified using its filename, which must have the format `[Camera Name][Band ID].txt` for text files, or `[Camera Name][Band ID].csv` for comma-separated value files; both file formats work equally well. For example, to load a text file for a camera named "MF" (in the directory `[Telescope Name]/MF/` and a Band ID "1," the file name should be `MF1.txt`. Note that for a multi-chroic camera, there should be multiple band files, one for each frequency channel.

3.6.2 Optics

Optics band files are stored in

```
[Telescope Name]/[Camera Name]/config/Bands/Optics/
```

The band is identified using its filename, which must have the format `[Optical Element Name].txt` for text files, or `[Optical Element Name].csv` for comma-separated value files; both file formats work equally well. For example, to load a text file for an optical element named "Lens1," the file name should be `Lens1.txt`. Note that there is only one band file for optical elements, even in multi-chroic cameras, as all detectors are all frequencies see the same bandpass. Also note that you can apply the same band to multiple optical elements by duplicating matching names in `optics.txt`.

3.7 Default Bands

If no custom bands are defined, `BoloCalc` assumes a top-hat band function, which is determined by the detector bandpass.

3.7.1 Detectors

If no custom detector bandpass is provided, then a top-hat band is assumed with an overall efficiency factor for the detector η_{det}

$$B(\nu) = \begin{cases} \eta_{\mathrm{det}} & \text{if } \nu_{\mathrm{lo}} \leq \nu \leq \nu_{\mathrm{hi}} \\ 0 & \text{otherwise} \end{cases} , \tag{26}$$

where the high and low band edges are defined as

$$\nu_{\mathrm{lo}} = \nu_c \left(1 + \frac{f_{\mathrm{BW}}}{2}\right) \;\; ; \;\; \nu_{\mathrm{hi}} = \nu_c \left(1 - \frac{f_{\mathrm{BW}}}{2}\right) \tag{27}$$

where ν_{c} is the band central frequency and f_{BW} is the fractional bandwidth.

3.7.2 Optics

If no custom bandpass is provided for an optical element i, then the bandpass is assumed to be a top-hat with an overall efficiency factor defined in Equation 20

$$B(\nu) = \begin{cases} \eta_i & \text{if } \nu_{\mathrm{lo}} \leq \nu \leq \nu_{\mathrm{hi}} \\ 0 & \text{otherwise} \end{cases} \tag{28}$$

If a custom detector bandpass is defined, then the frequency range covered ν_{lo} to ν_{hi} are the highest and lowest frequencies defined in the custom band. Otherwise, ν_{lo} and ν_{hi} are set by Equation 27.

4 Monte Carlo Simulation

`BoloCalc` has two executables `mappingSpeed.py` and `mappingSpeed_vary.py` that, when run, use independent sampling of all parameters using the Monte Carlo (MC) method to generate NET distributions from variations in the underlying instrument parameters. For information on how to run the executables, see Sections 1.5 and 1.6.

The MC simulation iterates using a nest of the following structure:

- N_{exp} Experiment realizations
  - N_{obs} Observations per experiment realization
    - N_{det} Detector realizations per observation

4.1 Simulation Inputs

Simulation inputs are stored in the file `BoloCalc/config/simulationInputs.txt`. An example of the file is shown in Figure 3. Below, we define the parameters for the simulation explicitly.

```
#***** Simulation Parameters *****
#
#------------------------------------------------------------------------------------------------------------------------
Parameter    | Value   | Description
#------------------------------------------------------------------------------------------------------------------------
Multiprocess | False   | Whether or not to use multiprocessing
#------------------------------------------------------------------------------------------------------------------------
Cores        | 1       | Number of cores to use for parallel processing. Positive integer
#------------------------------------------------------------------------------------------------------------------------
Verbosity    | 1       | Verbosity for printing logging messages to standard output. 0 for low, 1 for moderate, 2 for high
#------------------------------------------------------------------------------------------------------------------------
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   | 0.100   | Spectral 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
#------------------------------------------------------------------------------------------------------------------------
```

Figure 3: Example of simulation input file `simulationInputs.txt`

- **Multiprocess**
  - *Description*: Whether or not to use multiprocessing to calculate statistics in parallel
  - *Allowed Values*: "True" or "False"
  - *Default Value*: "False"
- **Cores**
  - *Description*: Number of cores to pool for parallel processing
  - *Allowed Values*: Positive integer between $[1, +\infty)$
  - *Default Value*: 1
- **Verbosity**
  - *Description*: Logging verbosity. 2 to print all logging output, 1 to print some output, 0 to print little output
  - *Allowed Values*: Integer between [0, 2]
  - *Default Value*: 1
- **Experiments**
  - *Description*: Number of experiment realizations to Monte Carlo; each time an experiment is MC-ed, its foreground, telescope, camera, and optical parameters are re-sampled.
  - *Allowed Values*: Positive integer $[1, +\infty)$
  - *Default Value*: 1
  - If 1, assume central value (i.e. ignore the spread) for all parameters set by the experiment realization.
- **Observations**

- *Description*: Number of independently observations, which draw the PWV and telescope boresight elevation from the distributions defined in `Telescope/config/pwv.txt` and `Telescope/config/elevation.txt`.
  - *Allowed Values*: Positive integer $[1, +\infty)$
  - *Default Value*: 1
- **Detectors**
  - *Description*: Number of independently-sampled detector parameters defined in `channels.txt` per observation.
  - *Allowed Values*: Positive integer $[1, +\infty)$
  - *Default Value*: 1
  - If 1, assume central value for all detector parameeters
- **Resolution**
  - *Description*: Frequency resolution assumed in the simulation.
  - *Allowed Values*: A positive floating-point value between (0.0, 20.0] GHz
  - *Default Value*: 0.1
  - Reducing the resolution can have a dramatic impact on the speed of the MC simulations.
- **Foregrounds**
  - *Description*: Whether or not to include foregrounds in the estimate of the optical loading.
  - *Allowed Values*: "True" or "False"
  - *Default Value*: "False"
  - Celestial loading doesn't have a noticeable impact on ground-based telescopes but can be important for satellite experiments.
- **Correlations**
  - *Description*: Whether to impose white noise correlations when calculating array NET
  - *Allowed Values*: "True" or "False"
  - *Default Value*: "True"
  - This switch is useful for computing the impact of white-noise correlations on array NET.

4.2 Parameter Variation

`mappingSpeed_vary.py` calculates the NET distribution using the MC method for a set of experiment configurations defined by the variation in user-defined parameters, which are defined in `BoloCalc/config/paramsToVary.txt`. An example of the `paramsToVary.txt` file is shown in Figure 1.

Parameters are defined by row and have the following descriptors, which are organized by column:

- Column 1: Telescope in which the parameter is defined
  - Leave blank if the parameter is defined in `config/foregrounds.txt`
- Column 2: Camera in which the parameter is defined
  - Leave blank if the parameter is defined in `Tel/config/telescope.txt`
- Column 3: Channel for which the parameter is defined
  - Leave blank if the parameter is defined in `Cam/config/camera.txt` of if Column 2 is blank.
- Column 4: Optic for which the parameter is defined
  - Leave blank **unless** the parameter is defined in `Cam/config/optics.txt`
- Column 5: Parameter name, as displayed in the input text files
- Column 6: Minimum parameter value to be calculated
- Column 7: Maximum parameter value to be calculated

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

The user can define as many simultaneous parameter variations as desired, and `mappingSpeed_vary.py` will calculate the NET distribution for all combinations of those parameters. The results are then stored in `[ExperimentDesign]/config/paramVary/`. For more information regarding the contents of the output files, see Section 5.1.4.

5 Output Files

Running `mappingSpeed.py` and/or `mappingSpeed_vary.py` generates several output files, which quantify the performance of the simulated experiment. In this section, we go through the layout of the output files as well as their contents.

5.1 Sensitivity Tables

`BoloCalc` produces tables of outputs related to the white noise performance of the instrument. All sensitivity tables are in files named `sensitivity.txt`, and there are multiple tables generated at multiple directory tree levels, describing the performance of the camera, telescope, and entire experiment.

5.1.1 `[ExperimentDesign]/sensitivity.txt`

When `mappingSpeed.py` is run, it generates the following parameters in the file `[ExperimentDesign]/sensitivity.txt`:

- `Chan`
  - *Description*: Frequency channel name `[Camera Name][Band ID]`
- `Frequency`
  - *Definition*: ν_{c} in Equation 26
  - *Description*: Central frequency of the frequency channel
- `Frac Bandwidth`
  - *Definition*: f_{BW} in Equation 27
  - *Description*: Fractional bandwidth of frequency channel
- `Num Det`
  - *Definition*: N_{det} in Equation 25
  - *Description*: Total number of detectors deployed in this frequency channel, combined for all telescopes within the experiment
- `Array NET`
  - *Definition*: $\mathrm{NET}_{\mathrm{arr}}$ in Equation 11
  - *Description*: Array-averaged noise-equivalent CMB temperature of this frequency channel, combined for all telescopes within the experiment
- `Mapping Speed`
  - *Definition*: MS in Equation 13
  - *Description*: Mapping speed of this frequency channel, combined for all telescopes within the experiment
- `Map Depth`
  - *Definition*: σ_{s} in Equation 12
  - *Description*: Map depth achieved by this frequency channel, combined for all telescopes within the experiment

Note that if identical band names are shared between telescopes and/or cameras, the detectors are combined in this table, with the combined Array NET of that combined band is found by taking the inverse-variance average of the duplicated bands.

Figure 4 shows an example of `sensitivity.txt` at the Experiment directory level, using the provided `ExampleExperiment/V0/` configuration.

Chan	Frequency	Frac Bandwidth	Num Det	Array NET	Mapping Speed	Map Depth
	[GHz]			[uK-rtSec]	[(uK^2 s)^-1]	[uK-arcmin]
Cam1	90.0 +/- 0.0	0.350 +/- 0.000	1626	10.79 +/- 0.00	0.0086 +/- 0.0000	7.4 +/- 0.0
Cam2	150.0 +/- 0.0	0.275 +/- 0.000	1626	10.38 +/- 0.00	0.0093 +/- 0.0000	7.1 +/- 0.0
Total			3252	7.48 +/- 0.00	0.0179 +/- 0.0000	5.1 +/- 0.0

Figure 4: `ExampleExperiment` sensitivity at the experiment level.

5.1.2 `[ExperimentDesign]/[Telescope]/sensitivity.txt`

When `mappingSpeed.py` is run, it generates the following parameters in the file `[ExperimentDesign]/[Telescope]/sensitivity.txt`:

- `Chan`
  - Frequency channel name `[Camera Name][Band ID]`
- `Frequency`
  - *Definition*: ν_c in Equation 26
  - *Description*: Central frequency of the frequency channel
- `Frac Bandwidth`
  - *Definition*: f_{BW} in Equation 27
  - *Description*: Fractional bandwidth of frequency channel
- `Num Det`
  - *Definition*: N_{det} in Equation 25
  - *Description*: Total number of detectors deployed in this frequency channel, combined for all cameras within the telescope
- `Array NET`
  - *Definition*: $\mathrm{NET}_{\mathrm{arr}}$ in Equation 11
  - *Description*: Aarray-averaged noise equivalent temperature of this frequency channel, combined for all cameras within the telescope
- `Mapping Speed`
  - *Definition*: MS in Equation 13
  - *Description*: Mapping speed of this frequency channel, combined for all cameras within the telescope
- `Map Depth`
  - *Definition*: σ_s in Equation 12
  - *Description*: Map depth achieved by this frequency channel, combined for all cameras within the telescope

Note that if identical band names are shared between cameras, the detectors are combined in this table, with the combined Array NET of that band is found by taking the inverse-variance average of the duplicated bands.

Figure 5 shows an example of `sensitivity.txt` at the telescope directory level, using the provided `ExampleExperiment/V0/` configuration.

Chan	Frequency	Frac Bandwidth	Num Det	Array NET	Mapping Speed	Map Depth
	[GHz]			[uK-rtSec]	[(uK^2 s)^-1]	[uK-arcmin]
Cam1	90.0 +/- 0.0	0.350 +/- 0.000	1626	10.79 +/- 0.00	0.0086 +/- 0.0000	7.4 +/- 0.0
Cam2	150.0 +/- 0.0	0.275 +/- 0.000	1626	10.38 +/- 0.00	0.0093 +/- 0.0000	7.1 +/- 0.0
Total			3252	7.48 +/- 0.00	0.0179 +/- 0.0000	5.1 +/- 0.0

Figure 5: `ExampleExperiment` sensitivity at the telescope level. Note that for the simple, single-telescope design of `ExampleExperiment`, this file is identical to what is shown in Figure 4.

5.1.3 `[ExperimentDesign]/[Telescope]/[Camera]/sensitivity.txt`

When `mappingSpeed.py` is run, it generates the following parameters in the file `[ExperimentDesign]/[Telescope]/[Camera]/sensitivity.txt`:

- `Chan`
  - Frequency channel name `[Camera Name][Band ID]`
- `Frequency`
  - *Definition*: ν_c in Equation 26
  - *Description*: Central frequency of the frequency channel
- `Frac Bandwidth`
  - *Definition*: f_{BW} in Equation 27
  - *Description*: Fractional bandwidth of frequency channel
- `Num Det`
  - *Definition*: N_{det} in Equation 25
  - *Description*: Total number of detectors deployed in this frequency channel within this camera
- `Lyot Efficiency`
  - *Definition*: η_{stop} in Equation 16
  - *Description*: Lyot stop spillover efficiency of the frequency channel
- `Optical Power`
  - *Definition*: P_{opt} in Equation 1
  - *Description*: Total optical power on detectors within this frequency channel
- `Photon NEP`
  - *Definition*: NEP_{ph} in Equation 4
  - *Description*: Noise-equivalent power due to photon noise for detectors within this frequency channel
- `Bolometer NEP`
  - *Definition*: NEP_{g} in Equation 5
  - *Description*: Noise-equivalent power due to bolometer thermal carrier noise for detectors within this frequency channel
- `Readout NEP`
  - *Definition*: NEP_{read} in Equation 8
  - *Description*: Noise-equivalent power due to readout for detectors within this frequency channel

- `Detector NEP`
  - *Definition*: $\sqrt{\mathrm{NEP}_{\mathrm{ph}}^2 + \mathrm{NEP}_{\mathrm{g}}^2 + \mathrm{NEP}_{\mathrm{read}}^2}$ in Equation 9
  - *Description*: Total noise-equivalent power due to the combination of photon, thermal carrier, and readout noise for detectors within this frequency channel
- `Detector NET`
  - *Definition*: $\mathrm{NET}_{\mathrm{det}}$ in Equation 9
  - *Description*: Per-detector noise-equivalent temperature for detectors within this channel
- `Array NET`
  - *Definition*: $\mathrm{NET}_{\mathrm{arr}}$ in Equation 11
  - *Description*: Array-averaged noise equivalent temperature of this frequency channel within this camera
- `Mapping Speed`
  - *Definition*: MS in Equation 13
  - *Description*: Mapping speed of this frequency channel within this camera
- `Map Depth`
  - *Definition*: σ_{s} in Equation 12
  - *Description*: Map depth achieved by this frequency channel within this camera

for every frequency channel in the provided camera.

Figure 6 shows an example of `sensitivity.txt` at the camera directory level. Note that this sensitivity file contains the most information about each individual channel. No combining of like channels happens at this low level.

Chan	Frequency	Frac Bandwidth	Num Det	Stop Efficiency	Optical Power	Photon NEP	Bolometer NEP	Readout NEP	Detector NEP
	[GHz]			[%]	[pW]	[aW/rtHz]	[aW/rtHz]	[aW/rtHz]	[aW/rtHz]
Cam1	90.0 +/- 0.0	0.350 +/- 0.000	1626	30.75 +/- 0.00	1.82 +/- 0.00	20.75 +/- 0.00	10.82 +/- 0.00	10.72 +/- 0.00	25.74 +/- 0.00
Cam2	150.0 +/- 0.0	0.275 +/- 0.000	1626	63.40 +/- 0.00	4.81 +/- 0.00	45.97 +/- 0.00	17.57 +/- 0.00	22.55 +/- 0.00	54.13 +/- 0.00
Total			3252						

Detector NET	Array NET	Mapping Speed	Map Depth
[uK-rtSec]	[uK-rtSec]	[(uK^2 s)^-1]	[uK-arcmin]
307.3 +/- 0.0	10.79 +/- 0.00	0.0086 +/- 0.0000	7.4 +/- 0.0
345.3 +/- 0.0	10.38 +/- 0.00	0.0093 +/- 0.0000	7.1 +/- 0.0
	7.48 +/- 0.00	0.0179 +/- 0.0000	5.1 +/- 0.0

Figure 6: `ExampleExperiment` sensitivity at the camera level.

5.1.4 `[ExperimentDesign]/paramVary/*.txt`

`mappingSpeed_vary.py` saves three quantities[7] to files within the directory `BoloCalc/Experiments/[Experiment]/[ExperimentDesign]/paramVary/`

- Optical power: `mappingSpeedVary_Popt_[Tel]_[Cam]_[Channel]_[Param].txt`
- Photon NEP: `mappingSpeedVary_NEPph_[Tel]_[Cam]_[Channel]_[Param].txt`
- Array NET: `mappingSpeedVary_NETarr_[Tel]_[Cam]_[Channel]_[Param].txt`

where `[Tel]_[Cam]_[Channel]_[Param]` is a string that contains the specifier for every parameter varied in the calculation. For example, if we looped over the following two parameters

- Telescope "Tel," camera "Cam," channel "1," optic "Mirror," Parameter "Spillover"
- Telescope "Tel," camera "Cam," channel "1," Parameter "Pixel Size"

[7] More quantities can be added to this list upon request, or as needed.

the P_{opt} output file would have the filename

```
mappingSpeedVary_Popt_Tel_Cam_1_Mirror_Spillover_Tel_Cam_1_PixelSize.txt
```

An example of the P_{opt} output file generated by running the provided example experiment, as described in Section 1.6, is shown in Figure 7. A similar file will be generated for photon NEP NEP_{ph} and array NET $\mathrm{NET}_{\mathrm{arr}}$. As the example output shows,

5.2 Optical Power Table

`BoloCalc` also outputs tables of optical powers, including the the following information in columns:

- `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
- `Power to Detect`
    - Power emitted from this optical element which is seen by the detector
- `Cumulative Eff`
    - Cumulative Efficiency between this optical element and the detector

These values are useful for understanding how signal propagates to the detector through the telescope optical chain, as well as seeing what the major sources of parasitic loading are.

5.2.1 `[ExperimentDesign]/[Telescope]/[Camera]/opticalPower.txt`

Optical powers are listed for each channel in the camera, formatted as one table per frequency. Figure 7 is an example optical power table for

Cam 1

Element	Power from Sky [pW]	Power to Detect [pW]	Cumulative Eff
CMB	0.00 +/- 0.00	0.08 +/- 0.00	0.169 +/- 0.000
ATM	0.49 +/- 0.00	0.74 +/- 0.00	0.175 +/- 0.000
LowPass4	4.85 +/- 0.00	0.00 +/- 0.00	0.186 +/- 0.000
Lens1	4.56 +/- 0.00	0.00 +/- 0.00	0.188 +/- 0.000
LowPass2	4.52 +/- 0.00	0.00 +/- 0.00	0.200 +/- 0.000
LowPass1	4.25 +/- 0.00	0.03 +/- 0.00	0.213 +/- 0.000
Lens2	4.16 +/- 0.00	0.00 +/- 0.00	0.215 +/- 0.000
IRShader3	4.12 +/- 0.00	0.02 +/- 0.00	0.215 +/- 0.000
IRShader2	4.21 +/- 0.00	0.02 +/- 0.00	0.215 +/- 0.000
IRShader1	4.31 +/- 0.00	0.03 +/- 0.00	0.215 +/- 0.000
Primary	4.43 +/- 0.00	0.31 +/- 0.00	0.218 +/- 0.000
LowPass3	5.79 +/- 0.00	0.00 +/- 0.00	0.232 +/- 0.000
Window	5.44 +/- 0.00	0.13 +/- 0.00	0.235 +/- 0.000
Mirror	5.93 +/- 0.00	0.06 +/- 0.00	0.236 +/- 0.000
Aperture	6.16 +/- 0.00	0.13 +/- 0.00	0.768 +/- 0.000
AbsFilter	2.02 +/- 0.00	0.27 +/- 0.00	0.792 +/- 0.000
Lens3	2.30 +/- 0.00	0.00 +/- 0.00	0.800 +/- 0.000
Detector	2.28 +/- 0.00	0.00 +/- 0.00	1.000 +/- 0.000

Cam 2

Element	Power from Sky [pW]	Power to Detect [pW]	Cumulative Eff
CMB	0.00 +/- 0.00	0.11 +/- 0.00	0.340 +/- 0.000
ATM	0.32 +/- 0.00	1.93 +/- 0.00	0.353 +/- 0.000
LowPass4	5.71 +/- 0.00	0.00 +/- 0.00	0.375 +/- 0.000
Lens1	5.36 +/- 0.00	0.00 +/- 0.00	0.381 +/- 0.000
LowPass2	5.29 +/- 0.00	0.00 +/- 0.00	0.405 +/- 0.000
LowPass1	4.98 +/- 0.00	0.09 +/- 0.00	0.431 +/- 0.000
Lens2	4.89 +/- 0.00	0.00 +/- 0.00	0.438 +/- 0.000
IRShader3	4.82 +/- 0.00	0.05 +/- 0.00	0.438 +/- 0.000
IRShader2	4.93 +/- 0.00	0.06 +/- 0.00	0.439 +/- 0.000
IRShader1	5.06 +/- 0.00	0.07 +/- 0.00	0.439 +/- 0.000
Primary	5.22 +/- 0.00	1.03 +/- 0.00	0.446 +/- 0.000
LowPass3	7.44 +/- 0.00	0.00 +/- 0.00	0.474 +/- 0.000
Window	7.00 +/- 0.00	0.72 +/- 0.00	0.484 +/- 0.000
Mirror	8.35 +/- 0.00	0.37 +/- 0.00	0.486 +/- 0.000
Aperture	9.08 +/- 0.00	0.03 +/- 0.00	0.765 +/- 0.000
AbsFilter	5.84 +/- 0.00	0.34 +/- 0.00	0.789 +/- 0.000
Lens3	6.10 +/- 0.00	0.00 +/- 0.00	0.801 +/- 0.000
Detector	6.01 +/- 0.00	0.00 +/- 0.00	1.001 +/- 0.000

Figure 7: `ExampleExperiment` optical tables for camera `Cam` in both of its bands. Each `opticalTable.txt` table is labeled by "[camera name] + [band name]."