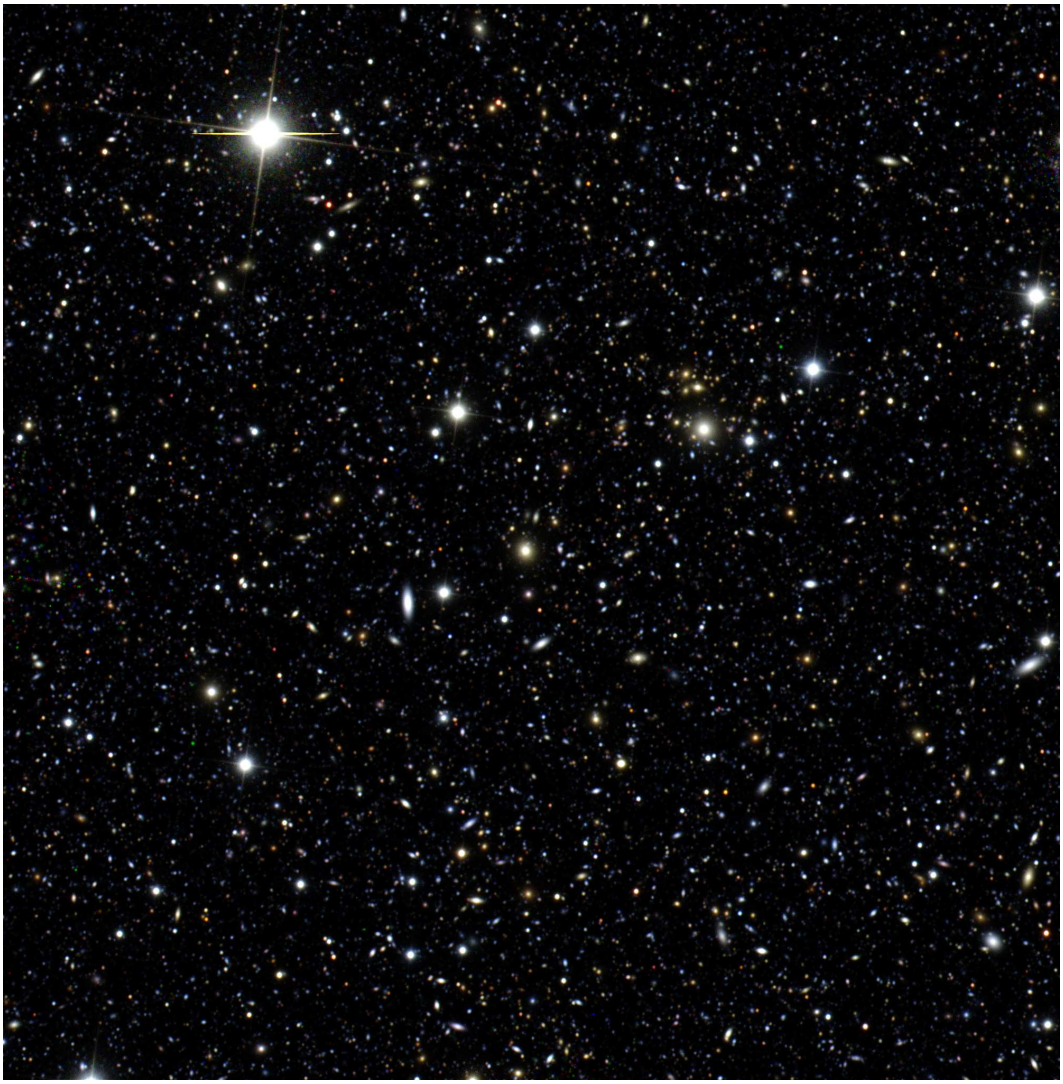
Technical Description of the LSST Photon Simulator and Its Validation Tests

John R. Peterson (Purdue) & J. Garrett Jernigan (SSL-UCB/KIPAC-SLAC/LSSTC)

Fall 2011 Release
November 29, 2011
Review Version



Preface

This work is a comprehensive and formal description of the photon simulator and its formal validation tests. It has been produced for the December 2011 validation review. This document is divided into four parts: I) an introduction to the photon simulator, II) a description of the physics of the photons simulator, III) a future improvement plan for the photon simulator, and IV) the description and results of the formal validation tasks. After the validation review, this document will become a living document and will be revised and re-approved coincident with image production runs. The software development process results in moving material from part III into part II, while updating the results in part IV.

In addition to this document, we also provide another document for the December 2011 review called *Results of the Validation of the LSST Photon Simulator*. It also contains the same validation tasks and part of the same improvement plan. It has additional informal validation tests, gives a more formal introduction to the validation approach, and discusses the general context of validation.

Contents

List of Tables

List of Figures

Part I

Introduction to the Photon Simulator

Chapter 1

Requirements, Scope & Overall Approach

Scope

The *photon simulator* is a novel set of codes that accomplish the task of taking the description of what astronomical objects are in the sky at a particular time (*the instance catalog*) as well as the description of the observing configuration (*the operational parameters*), and then produce a realistic data stream of images that are similar to what a real telescope would produce. To understand the approach used in the photon simulator, we first discuss the dual implicit requirements that have driven its software development. The *photon simulator* was also developed with some approximations. For example, we developed the simulator for large aperture wide field optical telescopes, such as the planned design of LSST. This class of telescopes are not diffraction limited in contrast to narrow field, large aperture telescope that are typical of adaptive optics system. The initial version of the simulator also targeted the LSST telescope and camera design. This model has now been broadened to include existing telescopes of a related nature. The atmospheric model, in particular, includes physical approximations that are limited to this general context.

Requirements

We have been working to meet essentially two overall requirements about how we decide to build and improve the simulator.

- **Fidelity Requirement:** *The physics of the simulator is required to have sufficient fidelity to be able to reproduce the photometric, astrometric, and image quality (point-spread-function size and shape) properties of a 17th magnitude astrophysical object in a ten year stack. The physics is required to include the full wavelength-dependent and time-dependent physics, so that the image quality are accurate for possible variations in the spectral energy distribution (SED) of the source and for the simulated exposure time.* A 17th magnitude source produces about 1 million photons for a typical LSST exposure and would be partially saturated, so this is a reasonable brightness to require complete fidelity. Brighter

objects should still be accurately rendered, but increasingly obscure physical effects that would not be observable in a 17th magnitude object would be not formally required. A stronger form of this requirement would be to accurately produce images for a 17th magnitude object in all exposures over the 10 years lifetime of LSST, as well as reproduce the sky background properties as well for the measurements of faint properties. This would result in about 1000 times more photons for that object, and a much stricter set of requirements. The astrometric, photometric, and PSF size/shape properties can be mapped to image quality accuracy tolerances that can be seen below the shot noise limit for a 17th magnitude object. However, it is clear that we are not at sub-percent accuracy in either the design of LSST itself or many of the physics models. This will probably be effectively accomplished after all the improvements described in part III are made. It can reasonably argued, however, that we are achieving complete fidelity (effects that can be seen above the shot noise) for the faintest objects (24th mag) in typical images already. These fainter objects are where a large fraction of the science will be done. In general, if you can measure a physical effect in an image for objects where actual measurements will be made, the simulator should eventually include that effect.

- **Speed Requirement:** *The speed of the overall simulation framework should be sufficient to be able produce images at a significant fraction of real-time LSST production rates (700 uncompressed Megabytes per second) using all available computational resources of the image simulation group.* A large fraction of the science goals of LSST involve complex image processing including stacking or differencing multiple 3-gigapixel images obtained over a several year baseline. This implies that in order to test algorithmic performance on many science topics sufficiently, a few sets of full stacks of LSST images obtained with all filters would be required. Thus, a reasonable estimate for the typical size of an algorithmically useful test is approximately $\approx$ 6 fields $\times$ 6 filters $\times$ 100 visits $\times$ 36 seconds per exposure $\approx$ a few nights of data. Thus, to make reasonable progress in understanding LSST algorithms and simulations and to have many software development cycles, a few nights of LSST data should be able to be simulated in at least a few week period. A reasonable long term goal, although not a requirement, would be to actually produce data at actual real-time rates as the Data Management software system would likely not plan on being able to process the data at far higher rates than the actual real time production rate. For data challenges to date we have acheived a simulation rate of order ~50 visits per day as compared to ~700 visits for each night of LSST operations. The focal plane of LSST is comprised of 21 rafts each with a 3x3 arrays of 4k x 4k CCDs. To date using thousands of processor cores we have approximately matched the real time image rate of a single LSST raft.

Thus, we argue that we have not met either requirement in its strongest form, but we are on a path to approaching these goals through continued software development, physics and design modeling, and additional validation.

Efficient Photon Monte Carlo on Grid Computing

In order to meet the dual requirements above, the photons simulator has used the novel approach of a sophisticated set of *numerically efficient photon Monte Carlo* codes. The codes have then be deployed for use on large-scale *grid computing* platforms. A photon Monte Carlo is the mechanism to produce an image in the following way:

$$\text{Catalogs} \xrightarrow[\text{photon sampling}]{} \text{Photons} \xrightarrow[\text{atmosphere, telescope, \& camera physics}]{} \text{Images}$$

The photon Monte Carlo approach involves first creating photons from the objects in listed in the input catalog that we designate the ”instance catalog” since it defines the celestial sources that are present in the LSST field of view for a pair of 15s exposures that is the basic minimum quantized period of time for LSST operations or an effective ”instance”. Then, we follow those photons through the atmosphere, telescope, and camera using wavelength and time-dependent atmosphere and instrument physics. The photon may convert into a photoelectron in the Silicon of a CCD, and the electrons are then collected into pixels. Finally, the readout physics is simulated to produce a highly accurate image. The final product is then the counts of electrons in each pixel. A photon Monte Carlo uses the general approach where the position and trajectory of a photons at a given time and wavelength are tracked. Often complex wavelength-dependent physics can then be accomplished with a single (or often few lines of code). The photon Monte Carlo approach enables us meet the fidelity requirement. We expect we can continue to improve the simulator further by having more sophisticated physical descriptions of where and how the photon can interact in its path. A Monte Carlo approach is the most efficient method for simulating the vast majority of primary science targets (galaxies or stars near the detection threshold). Signal celestial objects that are readily detected in single single chip scale images or in 10 years stacks of images that are as weak as a few tens of photons in a single 15s exposure. Modern sophisticated computational techniques are used throughout the photon simulator to efficiently produce images of these priority objects.

Grid computing is a method of dividing the computational process across thousands of processors. This is particularly useful for our problem because an individual processor can work on some fraction of the photons. The most practical division we have found is to have one processor work mostly the photons landing on an individual LSST CCD. The architecture of the code, is somewhat more complicated than this, however, as we describe in the following section. In general, efficient code that works on an individual workstation will also work efficiently on grid computing. However, keeping the memory per core below 1 to 2 Gbytes and the data I/O below few gigabyte per CPU hour is rather important. These limitations would not normally be constraints when developing code for a single desktop computer. The combination of modern efficient numerical techniques as well as grid computing make the second requirement achievable.

At the current time we have met most (but certainly not all) of the fidelity requirement by representing the photon physics in great detail. Similarly, in the most recent data challenges, we have produced data at 5% of actual real-time LSST production rates,

which has allowed us to produce millions of images using millions of CPU hours on a variety of grid computing platforms. The dual requirements are sometimes contradictory in that the code could have higher fidelity but would be slower (i.e. a fully diffractive EM code calculation), or could have lower fidelity but would be faster (i.e. a simple gaussian PSF convolver).

In the plot below, we show the number of photons emitted from stars and galaxies in the Universe. As the red dotted line shows, most of the photons detected from a telescope come from stars. However, stars brighter than 17th magnitude will saturate in a typical LSST exposure, making the accurate simulation of these photons less important. The photons from the objects in the 17th to 28th magnitude range, however, are critical. Since LSST has an effective aperture of 34 square meters and an exposure lasts 15 s, the black dashed curve can be multiplied by these numbers to see that the vast majority of objects will have either hundreds or thousands (but not millions) of photons detected from them. This is the ideal regime for a Monte Carlo. Each photon can be accurately simulated, but since there are only hundreds per object it is computationally efficient. Following, we show some examples of results achieved by the photon Monte Carlo approach.

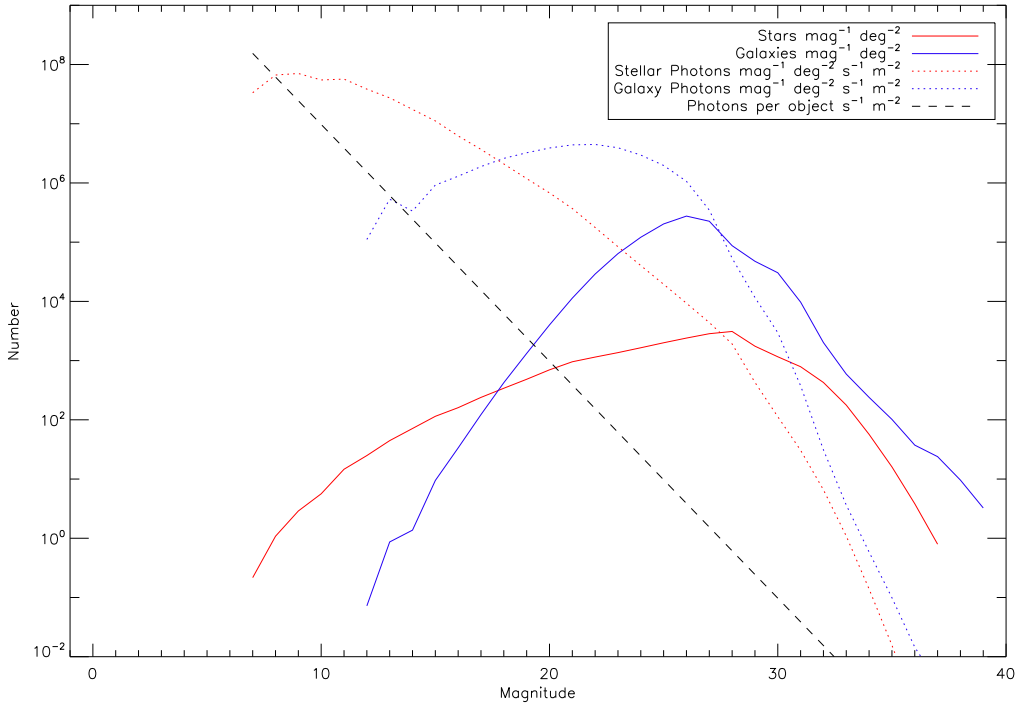


Figure 1.1: The photons emitted from the stars and galaxies in the Universe. The dotted lines show the total numbers of photons and the solid line shows a typical estimate for the numbers of stars and galaxies. The dashed line shows the number of photons per object. These distributions vary across the sky, and the number of photons (given between 0.3 and 1.2 microns) will depend on the SED of the source, but it will be within a factor ~2 of the calculation shown in the plot.

Sample Results

In the following figures, we show some examples of images where photons have been traced through the atmosphere, telescope, and camera. Figure 1.2 and 1.3 shows a detailed amplifier image with stars and galaxies. Currently, we have already simulated more than 10 million of these amplifer images. This is a comparable amount of data to that produced in the lifetime of many existing telescopes. It is also obviously more data than could ever be studied by a single human. Figure 1.4 shows the simulation of an entire LSST focal plane with its 3 gigapixels that cover 10 sq. degrees. This simulation takes about 500 CPU hours, and is now routine on our grid computing platforms. Figure 1.5 shows sample PSFs when we turn on successively more physical effects. The wide range of physical effects that will ultimately limit LSST's photometric astrometric, and PSF performance are visible. A system that can generate a large amount of data with high fidelity details can then be used to accurately determine whether science goals can be achieved as well as improve algorithmic performance.

The use of simulated images by the data management group is extensive. There have been four data challenges where simulated images have been used (DC3a, DC3b PT1.0, DC3b PT1.1, and DC3b PT1.2) over the last few years. We cannot even come close to describing the various ways the simulations have been used in this document and this is described elsewhere in data management reports. However, it is useful to look at the top line statistics of the last data run. In the table below, we list the statistics calculated from the pipeQA results demonstrating the full effectiveness of the combination of the simulation and data management software. Each statistic is a mixture of: 1) the current quality of the data management software, 1) the current realism of the photon simulator and astrophysical catalogs, 3) the survey design strategy, and 4) the design of the telescope and camera. This table shows the power of the simulations in that the statistics are impossible to know with real data, but they easily can be constructed with simulations by comparing with the truth. This also demonstrates the importance of high fidelity in the simulations, so that we are not misrepresenting realistic data for the data managment group. The fact that the results are somewhat reasonable at this stage in the project for the huge volume of simulated data is an important milestone.

Future Plans

Arguably, the photon simulator is not even close to being in a completed form. Since the photon simulator is a particularly novel approach in optical astronomy, the most efficient coding and self-consistent physics models are still being discovered by the ImSim team. In the Figure below, a rough set of metrics demonstrate that the code development is continuing at a rapid pace and is, in fact, accelerating. Therefore, what appears in the code now and in this document is a *snapshot* in time. However, it is a convenient time to describe the current configuration, obtain formal validation approval from the project, and discuss the future development priorities.

There are many planned future refinements in the physics and the efficiency that come in a variety of types: coding bugs, changes to the LSST design, internal consistency

Metric	Value	Affected by
Data Volume	449 visits; 2.7 million amplifier images	Scope of PT1.2
Completeness (Magnitude of 50% Completeness)	23.45 magnitudes (180 million detected sources)	DM source detection algorithms; Realism of photon simulator background/seeing; OpSim survey design strategy; Realism of magnitude distribution in astronomical catalogs
Photometry (Systematic Photometric Error Compared to Truth)	15 millimags (1.5%)	DM Photometric Algorithms; Realism of physics that causes photometric errors; (atmosphere opacity, clouds , instrumental chromatic effects, vignetting,etc.); SEDs of Astronomical Objects
Astrometry (Median Astrometric Error Compared to Truth)	40 milliarcseconds (0.22 arcseconds WCS accuracy)	DM WCS and centroiding algorithms ; Realism of astrometric physics (atmosphere, optical perturbations and misalignments, tracking) ; Design of Telescope and Camera ; Completeness because of photon statistics
PSF (Mean PSF Width Measurement)	0.90 arcseconds	DM PSF Characterization Algorithm; Realism of atmospheric refraction physics, Accuracy of optics misalignment physics & CCD Charge Diffusion; Design of Telescope and Camera ; Survey seeing conditions
PSF (Ellipticity Measurement)	typically 0.0 to 0.15	DM PSF Characterization Algorithm; Realism of atmospheric refraction physics, Accuracy of optics misalignment physics & CCD Charge Diffusion; Design of Telescope and Camera ; Completeness because of photon statistics

Table 1.1: The high level results of the entire PT1.2 simulation run that was used in data management pipelines. Neither the image simulation nor the data management pipelines are fully developed at this point. Nevertheless the production of these measurements is a major milestone.

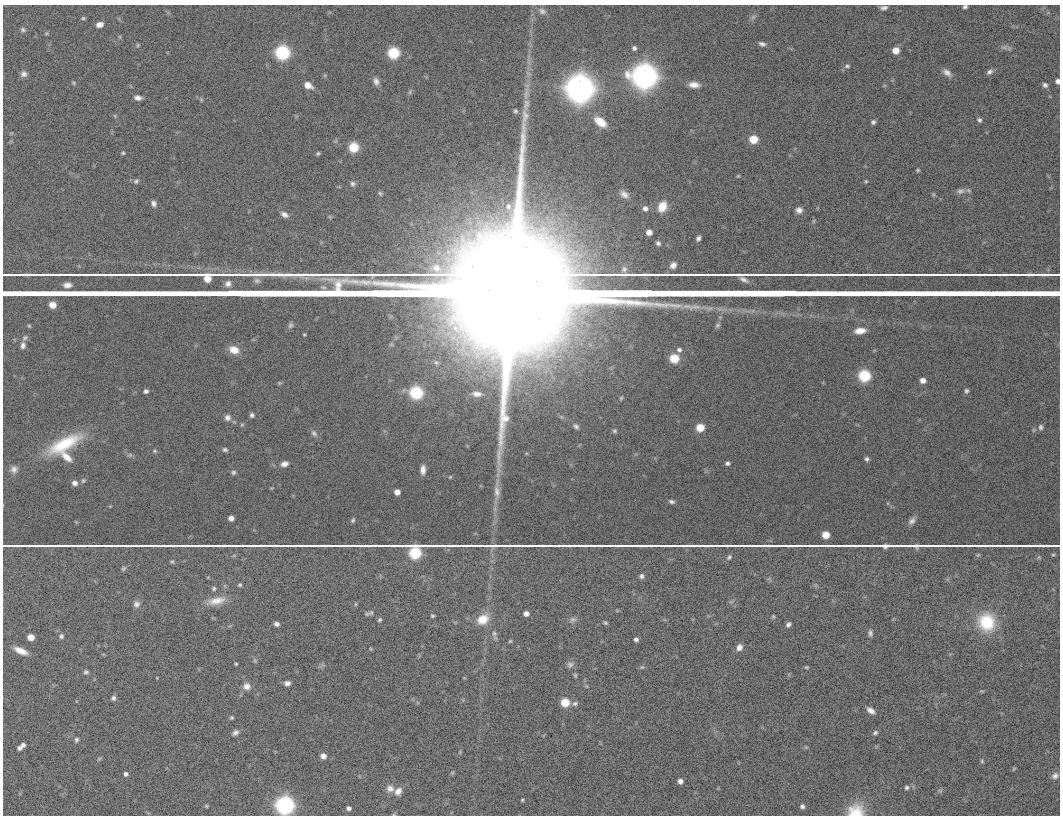


Figure 1.2: An example of three adjacent amplifiers. The amplifiers are 509 by 2000 pixels, and cover a region on the sky of about 1.7 by 6.7 arcminutes. The bright star that crosses the amplifier boundary shows diffraction spikes due to the secondary support spider and charge bleeding since many CCD pixels exceed a full well of charge.

Figure 1.3: Another example of three adjacent amplifiers. The amplifiers are 509 by 2000 pixels, and cover a region on the sky of about 1.7 by 6.7 arcminutes.

of multiple parts of the code, computational efficiency, grid computing restructuring, generalization to other telescopes, improved interfaces, and better physics models. In the Part II (the physics description) and Part IV (the validation tests) of this document, however, we restrict our discussion to what is in the current code at the moment. We describe a detailed plan for the future in part III of this document. In the following chapter describe the scope of what we call the photon simulator or *PhoSim*. As we describe in the following chapters, the photon simulator codes have considerable complexity. For the most part, we have followed a coding design approach where we attempt to meet our requirements and represent the physics as self-consistently as possible.

Additional Detailed Documentation

This work attempts to document the physics in considerable detail, but in many cases we will refer to the specific internal documentation about a given topic. The internal imSim documents are available at http://lsstdev.ncsa.uiuc.edu/trac/wiki/LSSTImageSim

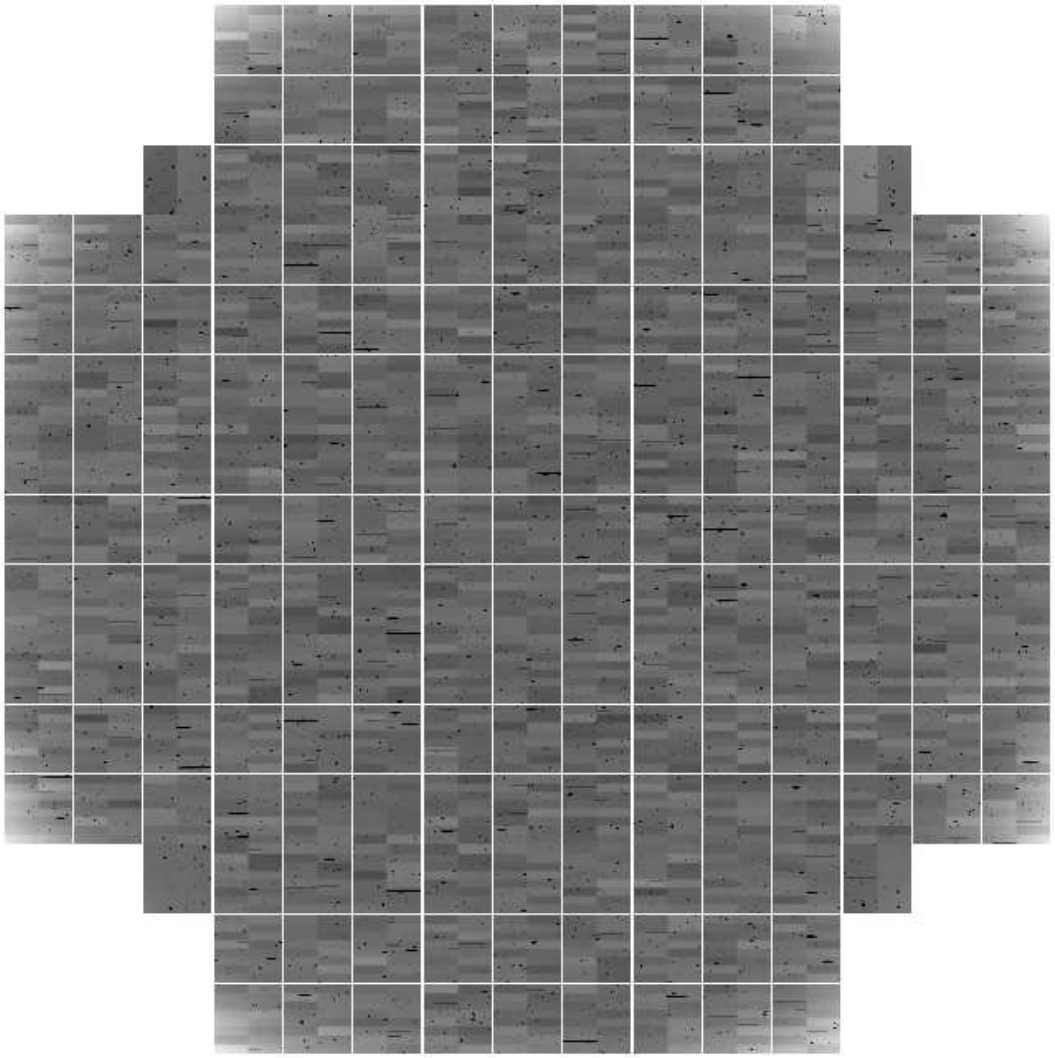


Figure 1.4: A simulation of an entire focal plane for LSST. The 193 different CCDs are represented by the squares, and the rectangular individual amplifiers (16 per chip) are also visible. These can be compared to the size of the previous plot to see the enormous detail that LSST collects in a single exposure. At this resolution, only the hot columns, and an occasional bright star can be seen. The vignetting of the light at the outer edge of the eight chips that are most distance from the center of the focal plane is clearly visible.

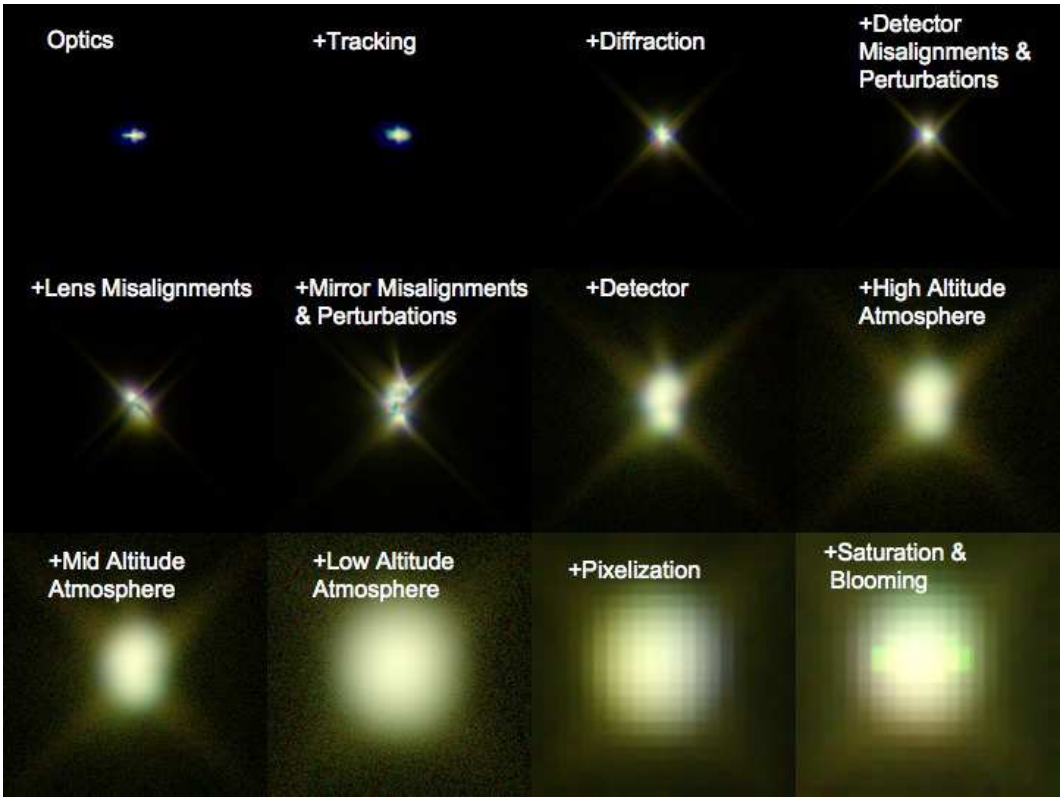


Figure 1.5: Example point-spread-functions with successively more physics included. The rgb colors represent the u-r-y filters, showing the wavelength dependence of some physical affects. The details of the physics can be rather important in determining the number of photons at each wavelength (photometry), the average position of a source (astrometry), and the objects image quality (PSF size and shape).

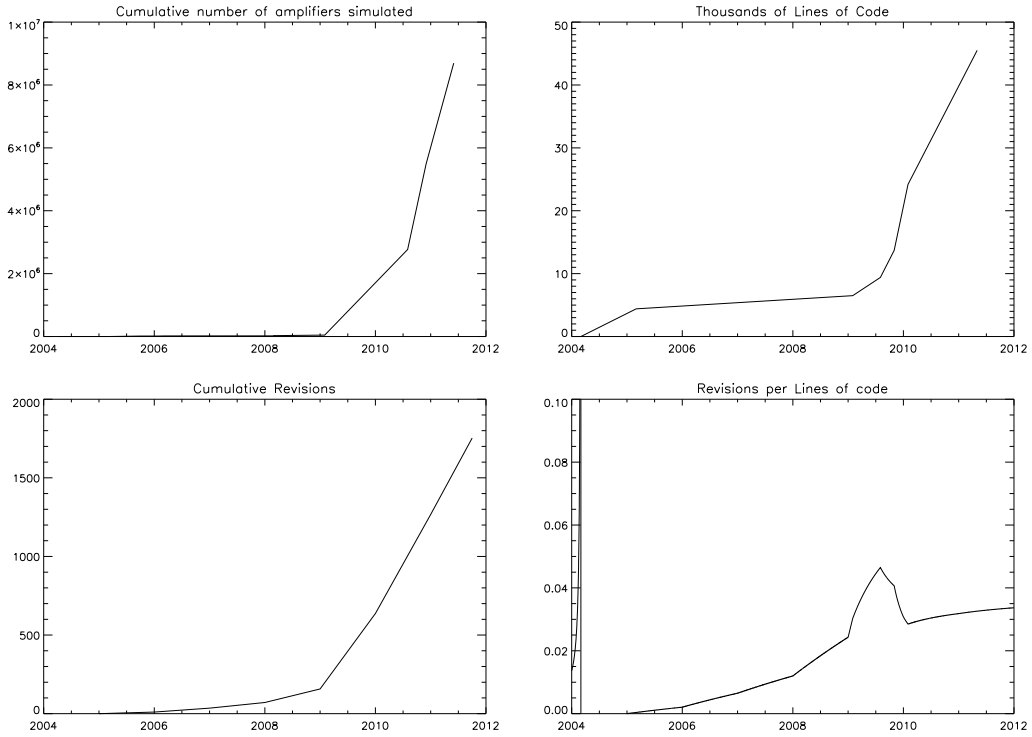


Figure 1.6: Some imperfect metrics that show the production and code development of the photon simulation and catalog constructor codes. These plot demonstrate that the codes are still in rapid development. The first few years were mostly spent developing the raytrace, and the increase in the last few years includes the infrastructure to simulate multiple observations (photon simulator non-raytrace codes), the catalog codes, the validation scripts, and grid computing infrastructure.

Chapter 2

Code Architecture & Interfaces

In this chapter, we describe the basic structure of the photon simulator (*PhoSim*) codes, and then discuss the input and output of the photon simulator. This then defines the scope of the photon simulator.

Code Architecture

We have designed a set of 10 novel C++ codes that optimally represent the physics self-consistently to describe the detection of a photon with a telescope looking through the atmosphere. In Figure 2.1, we show the overall architecture including the relationship to the operational simulator (OpSim) and the catalog simulations. The granularity of various codes are organized to parallelize the larger computational bottlenecks of the code. The raytrace, which is the bulk of the computation, simulates a single chip at a time on a given processor. This was a compromise between the simulator becoming I/O dominated if smaller regions were simulated, and to make the unit of computation as small as possible. We are I/O intensive at the beginning (inputs) and the end (outputs) of the simulation of a single chip scale image. We are computationally dominated during the phase when we track the path of every photon through the system (2 to 3 hours with one core per chip).

All code is now entirely written in C/C++, Python, or Shell scripting languages. It has been on the NCSA code subversion (SVN) repository since 2008, but developed since 2004. All revisions have been tracked through the SVN repository, so the revision history of both the code as well as the LSST-specific data files can be followed. There have been more than 2000 code revisions on the approximately 20,000 lines of code. The codes have been written by members of the ImSim group. A variety of simulation ideas and design parameters have been adapted from members of the entire LSST consortium.

The performance and structure of the code on grid computing is not fully described here. Table 2.1, however, lists of each of the codes and roughly points to the reasons for the current code architecture. The simulation time for a complete visit depends on the number of point-like and diffuse sources at a given magnitude, m

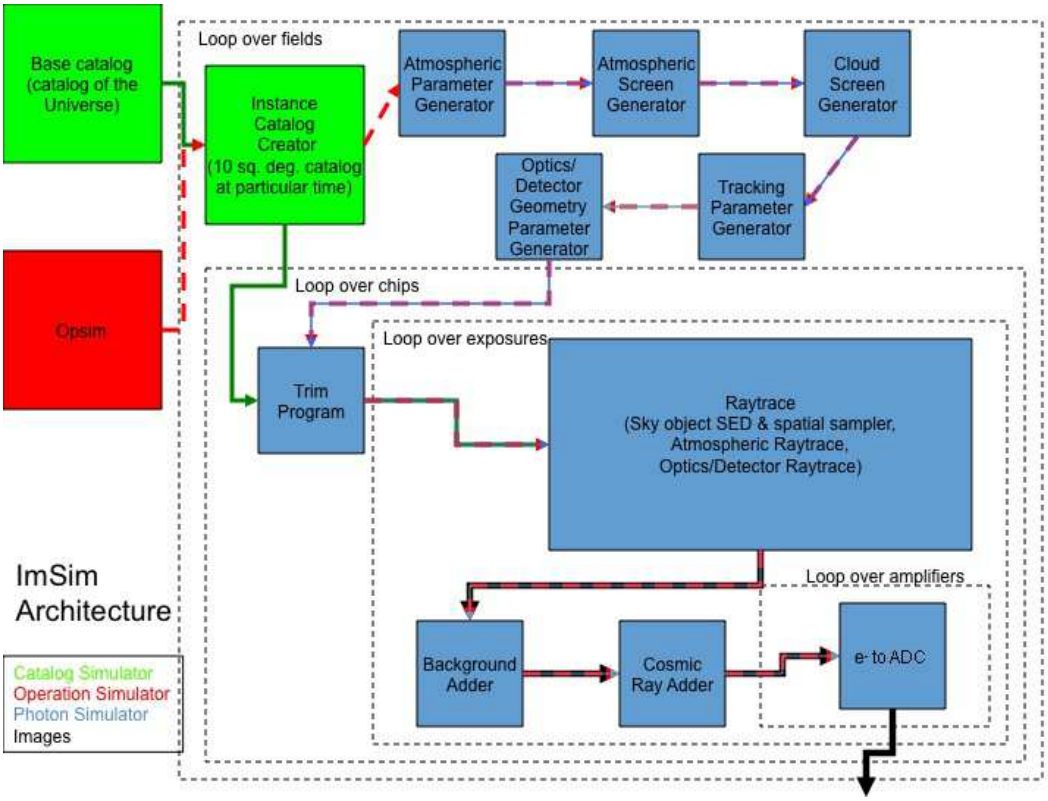


Figure 2.1: The basic architecture of the image simulation framework. The blue boxes show the 10 set of codes which comprise the photon simulation codes. Both the operation simulator (red) and the instance catalog constructor (purple) are the input to the simulator. The entire process results in the simulation of images. The dotted lines show the granularity of the parallelization.

Fixed Overhead:

$$30 \text{ minutes}$$

Stars:

$$+ \int dm \ \left(2.5^{16-14} \text{asinh}\left(2.5^{14-m}\right) N_{\text{point}}(m)\right) \text{ minutes}$$

Extended Sources:

$$+ \int dm \ \left(2.5^{16-m} N_{\text{diffuse}}(m)\right) \text{ minutes}$$

where 16th is about the magnitude where it can simulate a star in one minute, and 14th is the magnitude where saturation effects become dominant. The code optimizations cause

saturated objects to be simulated faster than for one photon at a time (faster than linear) for point-like objects. Therefore, the simulation speed has a rather complex form. The computational time of the codes that are not simulating photons (i.e. not the raytrace) are also a significant fraction of the computational time (fixed overhead for a chip scale image).

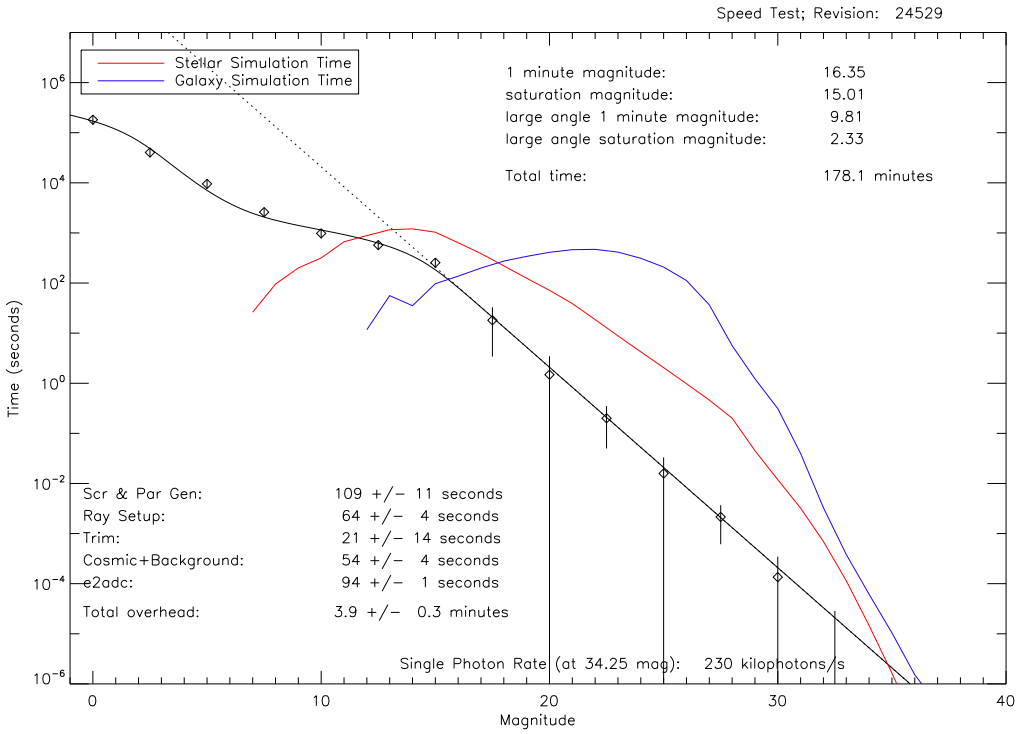


Figure 2.2: Measurement of the speed of the components of the photon simulator. This demonstrates that we have found a way to simulate the bright saturated stars more rapidly, so we spend most of the computational time simulating the vast numbers of stars and galaxies that LSST will measure near the detection threshold. The catalog construction time, and the I/O time are not included, but are significant.

Our purpose in this document is to describe the physics embodied in these codes rather than literally document the purpose of each code. In part II, we have organized the physics into Chapter 3 through 7, and these roughly follow the code organization. Chapters 3 through 5 describe the physics in the raytrace for the sky, atmosphere, and telescope/camera models. In addition, the atmospheric parameter, atmospheric screen, cloud screen, tracking parameter, and optics parameter codes have physics contained in these three chapters. Chapter 6 describes the simulation of non-astronomical sources as in the background and cosmic ray codes. Finally, chapter 7 describes the readout physics embodied in the e2adc code. The following sections first describe the input and output data of the photon simulator, and therefore define the scope of the photon simulator.

Code	Primary Author	Lines of code	CPU Time	Parallelization Factor	Wall Time	Fraction of Wall Time	I/O
Instance Catalog Generator	Connolly, Krughoff, Pizagno, Gibson	10000	15 minutes	field	15 minutes	10-15%	3 Gbytes
Photon Simulator Controlling Scripts	Peterson, Grace	500	negligible	field	negligible	0%	negligible
Atmospheric Parameter Generator	Young	500	negligible	field	negligible	0%	negligible
Atmospheric Screen Generator	Jernigan	1000	1 minute	field	1 minute	1%	1 Gbyte
Cloud Screen Generator	Jernigan	500	1 minute	field	1 minute	1%	negligible
Tracking Parameter Generator	Peterson	100	negligible	field	negligible	0%	negligible
Optics Parameter Generator	Todd	500	negligible	field	negligible	0%	negligible
Trim Program	Meert, Bankert	500	60 minutes	21 rafts pairs	1 minute	1%	3 Gbytes
Raytrace	Peterson	10000	500 hours	378 chips	90 minutes	60-70%	10 Gbytes
Background	Bankert	1500	3 hours	378 chips	1 minute	1%	10 Gbytes
Cosmic Rays	Young	500	2 hours	378 chips	1 minutes	1%	10 Gbytes
e^- to ADC	Peterson	1000	4 hours	6048 amps	1 minute	1%	10 Gbytes
Total I/O Latency					15-30 minutes?	10-20%	
CCD flatness (global)	Rasmussen	1000		all fields			1 Gbytes
Input to Background & Det defects (global)	Peterson, Bankert	1000		all fields			1 Gbytes
Validation Pipeline (global)	Peterson, Todd	5000		all fields			1 Gbytes

Table 2.1: A list of the various codes of the photon simulator and their computational efficiency & parallelization. Global scripts are not run on grid computing.

Input

The basic input of the photon simulator consists of three parts:

- **Observing configuration parameters:** These are parameters defining the pointing, position of moon and sun, etc. that are typically taken from OpSim simulations. The intention is to not use absolute time in the photon simulator, and so the details about when the observation is taken, precession, etc. are left outside the simulator. These parameters tend to be similar to the types of controls a telescope operator would specify, and parameters affected by the time of observation.

- **Instance Catalog**: A list of all astronomical objects that are within a *2.1 degree radius circle* (which covers the full focal plane) at the particular time of the *exposure pair*. There are many parameters that can describe the object that we list below. The catalog normally consists of parameterized models, but could also include lists of postage stamp truth images.

- **Spectral Energy Distribution (SED) files**: A reference set of SED files to be referenced in the instance catalog. Since the number of objects in a typical instance catalog can be around 10 million, there cannot be a different SED for every object. Similarly, the redshifting of photons have to be done in the simulator (to not make a unique SED for every galaxy), so these are the rest frame SEDs.

Observing Configuration Parameters

The 21 parameters that specify the observing configuration are given below. These are used by some of the 10 codes in the photon simulator to parameterize the physics used to simulate the images.

Type	Parameter	Notes
Telescope	Unrefracted_RA_deg	OpSim derived right ascension converted to unrefracted Value in units of decimal degrees
	Unrefracted_Dec_deg	OpSim derived declination converted to unrefracted Value in units of decimal degrees
	Unrefracted_Azimuth	OpSim azimuthal angle converted to decimal degrees
	Unrefracted_Altitude	OpSim altitude angle converted to decimal degrees
	Opsim_rotskypos	OpSim RotSkyPos (angle of sky relative to camera) in decimal degrees
	Opsim_rottelpos	OpSim RotTelPos (angle of spider relative to camera) in decimal degrees
	Opsim_filter	Filter configuration
	SIM_TELCONFIG	(0=normal, 1=dome/shutter closed, 2=dome closed/dome light on)
Moon	Opsim_moonra	OpSim moon right ascension in decimal degrees
	Opsim_moondec	OpSim moon declination in decimal degrees
	Opsim_moonalt	OpSim moon altitude in decimal degrees
	Opsim_dist2moon	OpSim distance to the moon in decimal degrees
	Opsim_moonphase	OpSim moon phase (0=new,100=full)
Sun	Opsim_sunalt	OpSim sun altitude in decimal degrees
Time	Opsim_ObsHistID	Labels files for a particular visit
	Opsim_expmjd	International Atomic Time (Informational Only)
	Slalib_date	Date calculated from TAI
	SIM_VISTIME	Total time of the exposure visit
Atmosphere	OpSim_rawseeing	FWHM in arseconds of atmospheric turbulence at zenith at 500 nm
Photon Simulator Specific	SIM_SEED	Random Seed
	SIM_MINSOURCE	Minimum number of sources to perform simulation on a chip (Often 0 or 1)

Table 2.2: Operational parameters that are necessary for the photons simulator.

Instance Catalog Format

The catalog of objects should be a gzipped ASCII file and have the following form. The parameters will be read in at single precision, if that is found necessary for any of these parameters. Each line should contain one model, but a single astronomical objects may be composed of multiple models.

object ID# ra dec mag SED_filename redshift γ_1 γ_2 μ Δra Δdec **spatial_model_name spatial_par_1 spatial_par_2 ... rest_frame_dust_model dust_par_1 dust_par_2 lab_frame_dust_model dust_par_1 dust_par_2**

An entry in the catalog is actually a command to the Monte Carlo as well, which begins with the word object. We define each entry as follows:

- **object:** The first entry of each line that also acts as a command for the raytrace.
- **ID#:** A floating point number that is unique identifier of the object (no internal use by the photon Monte Carlo).
- **ra:** The right ascension of the center of the object or image in decimal radians.
- **dec:** The declination of the center of the object in decimal radians in J2000 coordinates.
- **mag:** The normalization of the flux of the object. The units of the flux internal to the Monte Carlo are photons$\mathrm{cm}^{-2}\mathrm{s}^{-1}$, but we use AB magnitudes at 5000 Å/(1+z) (which is roughly equivalent to V (AB) or g (AB)).
- **SED_file:** The name of the SED file relative to the data directory.

- **redshift:** The redshift (or blueshift) of the object. Given that redshift is used quite often, it does not make sense to redshift the SED file externally. The wavelength of the photons are shifted by (1+z) within the photon Monte Carlo. Any cosmological dimming must be included in the flux normalization.
- γ_1**:** The value of the shear parameter γ_1 used in weak lensing.
- γ_2**:** The value of the shear parameter γ_2 used in weak lensing.
- μ**:** The value of the magnification parameter given in magnitudes.
- $\Delta\alpha$**:** The value of the declination offset in radians. This can be used either for weak lensing or objects that move relative to another exposure if you do not want to change the source position in the first two columns. $\Delta\alpha$ is a small offset for the value of the **ra** column that makes it easy to build multiple components of objects that all reference a common **ra**.
- $\Delta\delta$**:** The value of the declination offset in radians. This can be used either for weak lensing or objects that moved from another exposure if you do not want to change the source position in the first two columns. $\Delta\delta$ is a small offset for the value of the **dec** column that makes it easy to build multiple components of objects that all reference a common **dec**.
- **spatial_modelname:** The name of the spatial model to be used, which are coded in the Monte Carlo. We have currently implemented the following spatial models:
  - **point (no parameters):** This is a model primarily used for stars, but also unresolved objects.
  - **gaussian (sigma in arcseconds):** This is a model for a gaussian-shaped object.
  - **movingpoint:** (the derivative of the velocity arcseconds per second along the ra direction, the derivative of the velocity in arcseconds per second along the dec direction)
  - **sersic2D:** (size of axis 1 in arcseconds, size of axis 2 in arcseconds, position angle in radians, sersic index)
  - **sersic:** (size of axis 1 in arcseconds, size of axis 2 in arcseconds, size of axis 3 in arcseconds, sersic index, polar angle in radians, position angle in radians)
  - **image models:** If the spatial model name is a string which contains "fits" or "fit" the model is assumed to be a spatial image. Then there are two parameters for this model: pixel size (in arcseconds) and rotation angle (in degrees).
- **spatial_parameters:** The associated parameters for each spatial model. There could be none or many. While the parser is reading the model it looks for more parameters based on the name of the model.
- **Rest_Frame_Dust_modelname:** This is either the ccm for the CCM model, or calzetti for the calzetti model. If no dust model is desired, then put "none" for this field.

- **dust_parameters:** The parameters for both the calzetti and CCM are the A_v followed by the R_v value. If no dust model is used, do not use parameters
- **Lab_Frame_Dust_modelname:** This is either the ccm for the CCM model, or calzetti for the calzetti model. If no dust model is desired, then put "none" for this field.
- **dust_parameters:** The parameters for both the calzetti and CCM are the A_v followed by the R_v value. If no dust model is used, do not use parameters

Non-standard ways of using this format: We have also previously considered the concept of a data cube consisting of a series of images in small wavelength bands (several times smaller than a filter width). This approach is not optimal for a photon Monte Carlo of many sources, since this quickly grows too large because areas where there are no sources have many data points. However, this approach is useful for certain special applications. Technically, this format is supported in two different ways. First, one can use a series of images for a single source and have each one have a piece of the SED. Second, one can represent an extended source with a complex spatially dependent spectrum by a series of closely spaced point sources each with their own SED.

It is also possible to use a single truth image and a single SED for the entire field, and therefore just have a single entry of the catalog. This does not take full advantage of the wavelength-dependence capabilities in the Monte Carlo. In practice for a full LSST field of view a mosaic of images each modest in size would be practical as compared to s single extremely large image.

SED files

The SED files are gzipped ASCII files containing two columns: wavelength in nm and flux in units of $ergs/cm^2/s/nm$. A variable non-uniform wavelength spacing is allowed with arbitary start and end wavelengths. The flux is renormalized by the information in the instance catalog as the file is read, therefore the normalization the SED file is arbitrary. The SED is linearly interpolated within a wavelength bin. The gridding is arbitrary to accomodate any required resolution for the wavelength dependent structure.

Output

There are two forms of output for the simulator: 1) images and 2) additional diagnostic data.

Images

The most important output of the simulator is very straight-forward: a set of gzipped FITS files containing the ADU counts in each pixel for each amplifier. Calibration images (darks, flats, and biases) are also produced when the simulator is run in certain configurations.

Diagnostic Data

In the photon simulator, we also have the luxury of producing intermediate diagnostic data that would be impossible for a real telescope could produce. These have proven useful in validation, and also in DM algorithm development This includes: 1) centroid files (mean positions and number of photons for each astronomical object separately), 2) throughput files (counts of photons at every optical surface), 3) event files (locations of all the photon ray hits in 3-D), and 4) eimages (electron images before the background and detector defects and electronic effects are applied). We can easily accomodate new formats of diagnostic data in the future. All of the validation plots in this document use either the raw images or various kinds of diagnostic data.

Part II

Physics Description of the Simulator

Chapter 3

Photon Sampling from Astrophysical Sources

Properties of the Photon

We first create photons from astrophysical sources. The photons have several properties that are dictated by astrophysical models of the source: the direction, the wavelength, and the arrival time. The total number of photons are calculated from the magnitude normalization (described below). Finally, the photons evenly sample the entrance aperture projected to the top of the atmosphere which sets the initial three dimensional position.

Direction

The direction of the photon is computed using a spatial model for the source which defines a relative position of each photon as an offset from the right ascension and declination of the source. Internally, the direction is always converted to a three-vector relative to the coordinate system. The spatial model has parameters that model can also be the effect of gravitational lensing.

Wavelength

The wavelength of the photon is determined by sampling from the SED files (see below). The SED files are linearly interpolated between grid points. The wavelength of the photon is redshifted after it is sampled from the SED.

Time

The arrival time of a photon is uniformly chosen between 0 and the maximum exposure time. The absolute time offset relative to the beginning of the exposure pair is used in some time-dependent calculations.

Position

The initial position of the photon in 3-D is calculated in the following way. First, the entrance aperture (an annular shape) is sampled uniformly in x and y. Second, we look ahead to the shape of the primary mirror to calculate a z position given the values of x and y. Third, we then use the direction of the photon to move the ray backwards to a height of 20 km. This results in a new x and y position. This results in photons sampling the entrance aperture as efficiently as possible, but is approximately equivalent to sampling photons filling the projected annular aperture at the top of the atmosphere (~20 km)

an annulus at 20 km shifted by the angular offset.

Numbers of Photons

The total number of photons for a particular source are calculated by considering the AB magnitude at a particular wavelength (in our case, 500 nm divided by (1+redshift)) and converting that a flux in ergs per sq. centimeter per second per Hz. Then, the Spectral Energy Distribution (SED) is converted to a relative fraction of photons in each bin. Using the probability of finding a photon in the bin near the reference wavelength (500 nm/(1+redshift)), one can then calculate the total number of photons per sq. centimeter per second from that source at all wavelengths. Then the total number of photons are calculated by multiplying by the aperture size and exposure time.

Photons are randomly sampled from all the sources in proportion to their relative photon fluxes, until the total number of photons for all sources (with a Poisson error added) is simulated.

Astrophysical Models

Star & Asteroid Models

Our spatial model of a star is simply a point. The asteroid model specifies a velocity vector, so the spatial position will vary given the current arrival time for that photon.

Galaxy Models

Reference: Lorenz, S. (ImSim Internal Document-5), Sérsic, J. L. (1963)

The galaxy model normally used specifies an ellipsoidal sersic distribution in two dimensions. The sersic index, major and minor axis, and position angle are the parameters. Then a single galaxy is normally constructed out of two sersic models: one representing the bulge and another representing the disk. Each has its own sersic index.

SED sampling

The SED files are sampled by constructing cumulative distributions, and drawing photons according to that probability distribution. The wavelengths are linearly interpolated

between bin points. The photons are first absorbed by the first dust model using this wavelength, then the photons wavelengths are redshifted, and then they are absorbed by the second dust model at the new wavelength.

Dust Models

References: Cardelli, Clayton, Mathis (1989); Calzetti et al. (2000)

There are two dust models given by the Cardelli, Clayton, & Mathis (1989) model and the Calzetti et al. (2000) model. Each model has parameters: A_V and R_v. The models are stored in a variety of grid lookup tables and the dust is applied by first calculating the optical depth given the photon wavelength. Then we destroy photons with probability $e^{-\tau}$. Performing dust extinction through a Monte Carlo approach conveniently avoids construction of a unique SED for every single source.

Chapter 4

Atmosphere

The atmosphere model consists of a continuous model for the distribution of atmospheric molecules and a series of discrete layers of thin 2D screens that approximate the integrated effect of the nearbuy 3D layer of the air. Some screens have a fixed pattern atmospheric turbulence and other have a cloud pattern. The raytrace follows the photon from the top to the bottom of the atmosphere. We discuss first how the screens are constructed and positioned, then discuss how the photons interact with the screens, and describe the physics of propagation in the volume between the screens.

Screen & Bulk Geometry

Ray Intercept Calculation & Layer Geometry

The geometric model of the atmosphere consists of a series of screens and the molecular air model located between the screens. The screens are arranged perfectly parallel to the ground, making the calculation of the ray intercepts a simple analytic calculation. The screens represent both the turbulence and the clouds that comprise each 3D thin layer of the atmosphere.

Turbulence Screens

The screens of atmospheric turbulence follow the well-studied frozen-screen approximation, meaning they do not change their character as they drift over different regions of the atmosphere. Different photons arrive at different times, so even if they have the exact same path they may hit different parts of the screens. The approach is a known accurate way to simulate the time dependence of the physics.

Number of Layers and Spacing

Reference: Young, M. (ImSim Internal Document-9)

We use 20 layers of atmospheric turbulence arranged logarithmically in height between 20 m and 20 km. 20 km is a reasonable upper limit where the atmosphere is likely to

have a negligible impact on the observed seeing. 20 layers was determined to most accurately represent the degree of decorrelation of typical atmosphere using wind data (see M. Young document). We assume the screens are independent,so having more than 20 screens would falsely represent independent layers, and less than 20 layers would cause incomplete averaging of the photon refraction.

Pixel Scale

The screens are required to cover a horizontal region of about 1 km, in order to cover all paths that a photon could intersect the screens during a typical LSST exposure. On the small scale, the screen should be able to represent the turbulence on the 1 cm scale. In order to have meet reasonable memory and file size requirements, we found that we needed to construct the screen out of linear superpositions of three different pixel sizes: 1 cm, 10 cm, and 1 m. The screens all are 1024 by 1024 pixels, and the screens are toroidally wrapped when a pixel is accessed off the edge of a screen. The pixel sizes are constructed so the largest pixel screen (1.024 km across) will not wrap during two 15s exposures.

Kolmogorov Spectrum

Reference: Kolmogorov, A. N. (1941)

The screens are generated to model a density perturbation in every pixel. A Kolmorogov spectrum is used where the power as a function of inverse spatial wavelength decreases to power of $-\frac{5}{3}$ (3D index). Thus, the largest power is on the largest spatial scales. We flatten the power at the outer scale which matches the expected form of the spectrum for shear driven turbulence in air. We set the power to zero at the outer-outer scale to sensibly limit the physical form of the power on extremely large scales. The inner scale would normally be set to the viscous scale. However since the Kolmogorov spectrum rapidly falls at small scales we cutoff the power at 1 cm without effecting the path of any photons. We compute the gradient of the scalor density screen which yields a 2D vector screen of kicks that model the refractive effects of the variations in the density that are enduced by the turbulent flow. We also must filter the spectrum of the density and subsequent vector gradient with a Bessel function which depends on the Fried parameters r0 (implies the value of seeing). This extra filter step approximates the wavelength dependent effects that forms the expected speckle pattern.

Outer Scale

Reference: Bocass, M. (2004); Beland & Brown (1988); Coulman et al. (1988) Abahamid et al. (2004); Young, M. (ImSim Internal Document-9)

We use the outer scale model which predicts an outer scale a function of height, and match the mean parameters to the LSST site data of Bocass (2004) where a median outer scale of 26.7 m and mean outer scale of 35.6 m. The mean and median can be used to construct a log-normal distribution. To add altitude-dependence, we scale the mean as

a function of height by scalings found in Abadhamid et al. (2004) and references therein where the turbulence weighted median values are 1.27.

$$L_0 = \begin{cases} 3.21h^{-0.11} & h < 2km \\ \frac{4.0}{1.0+\left(\frac{h-8500}{2500}\right)^2} & 2km < h < 17km \\ 0.307 - 0.0324\left(\frac{h}{1000} - 17\right) & \\ +0.00167\left(\frac{h}{1000} - 17\right)^2 + 0.000476\left(\frac{h}{1000} - 17\right)^3 & h > 17km \end{cases}$$

Note, however, that estimates for the outer scale and its distribution vary considerably in the literature.

Outer-Outer Scale

We set the power of the density fluctuations to zero at 500 m for a screen size of 1 km. This parameter (newly invented here) is needed to set the outer most physical scale so the the overall model will pass a convergence test. The flat power spectrum that drives the Kolmogorov spectrum cannot sensibly extend to arbitary large scales. Since the aperture of LSST is much smaller than 500m the simulated paths of photons are not too sensitive to this unmeasured parameter.

Relative Turbulence Intensity

Reference: Vernin et al. (2000); Young, M. (ImSim Internal Document-9)

We use a Weibull distribution for the relative turbulence in each layer. The refractive index structure constant sigma values ($m^{-\frac{2}{3}}$) are given below. We have a seasonal model, so that the turbulence follows a realistic distribution based on LSST site data.

meters	January	April	July	October
23000	$0.00 \times 10^{+00}$	$0.00 \times 10^{+00}$	$0.00 \times 10^{+00}$	$0.00 \times 10^{+00}$
22000	$0.00 \times 10^{+00}$	$0.00 \times 10^{+00}$	$0.00 \times 10^{+00}$	$0.00 \times 10^{+00}$
21000	1.20×10^{-18}	8.78×10^{-19}	$0.00 \times 10^{+00}$	8.78×10^{-19}
20000	4.79×10^{-18}	1.76×10^{-18}	9.57×10^{-19}	1.60×10^{-18}
19000	7.18×10^{-18}	3.19×10^{-18}	2.39×10^{-18}	2.31×10^{-18}
18000	6.38×10^{-18}	4.79×10^{-18}	3.99×10^{-17}	2.79×10^{-18}
17000	3.99×10^{-18}	2.71×10^{-18}	1.20×10^{-17}	3.99×10^{-18}
16000	2.39×10^{-18}	3.03×10^{-18}	1.60×10^{-17}	4.39×10^{-18}
15000	2.39×10^{-18}	3.99×10^{-18}	1.36×10^{-17}	3.99×10^{-18}
14000	2.39×10^{-18}	3.99×10^{-18}	6.38×10^{-18}	2.39×10^{-18}
13000	1.76×10^{-18}	4.79×10^{-18}	4.79×10^{-18}	1.76×10^{-18}
12000	1.44×10^{-18}	4.31×10^{-18}	3.27×10^{-18}	2.39×10^{-18}
11000	3.27×10^{-18}	7.58×10^{-18}	4.79×10^{-18}	3.99×10^{-18}
10000	3.19×10^{-18}	6.38×10^{-18}	5.66×10^{-18}	6.62×10^{-18}
9000	5.19×10^{-18}	1.20×10^{-17}	1.04×10^{-17}	4.79×10^{-18}
8000	7.18×10^{-18}	8.78×10^{-18}	1.04×10^{-17}	4.87×10^{-18}
7000	2.55×10^{-17}	7.98×10^{-18}	1.44×10^{-17}	8.78×10^{-18}
6000	3.19×10^{-17}	1.36×10^{-17}	3.03×10^{-17}	2.07×10^{-17}
5000	2.15×10^{-17}	1.60×10^{-17}	1.76×10^{-16}	2.71×10^{-17}
4000	4.39×10^{-17}	3.59×10^{-17}	1.44×10^{-16}	5.98×10^{-17}
3000	7.18×10^{-17}	1.52×10^{-16}	1.44×10^{-16}	5.59×10^{-17}
2000	9.57×10^{-17}	6.38×10^{-17}	1.68×10^{-16}	1.12×10^{-16}

Table 4.1: The average turbulence intensity for various altitude as a function of season for the LSST site.

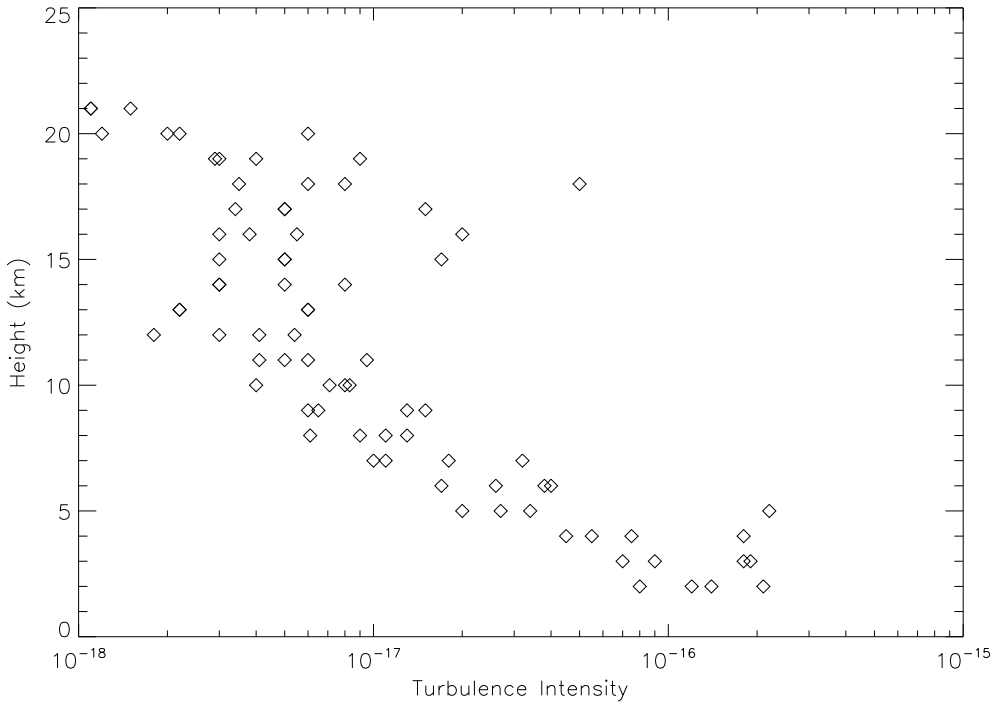


Figure 4.1: The mean turbulence intensity vs. height. The 4 points at each layer are the seasonal variation.

Absolute Turbulence Intensity

The turbulence values listed above predict an absolute level of seeing. However, we iteratively choose values for the structure function until we reproduce the seeing value input from the operation simulation. The value of turbulence intensities are then renormalized to produce the exactly predicted values. This assures us that we have the same seeing distribution as in the operations simulator. Alternately we can use the seasonal model of the structure function of the model to randomly select a structure profile independent of *OpSim*. Data challenges use the *OpSim* provided values for total seeing. Special studies of stacks of LSST images can use the internal structure funnction model and the associated integrated seeing values.

Density Screen vs. Screen Derivatives

The largest scale density screen is kept for the opacity calculation, and the derivative of density of all three screens is used for the refraction calculation. The derivative is calculated using a simple 2-point derivative. In the figure below, we show the density of the coarse screen, and the derivative of the screen on three different scales.

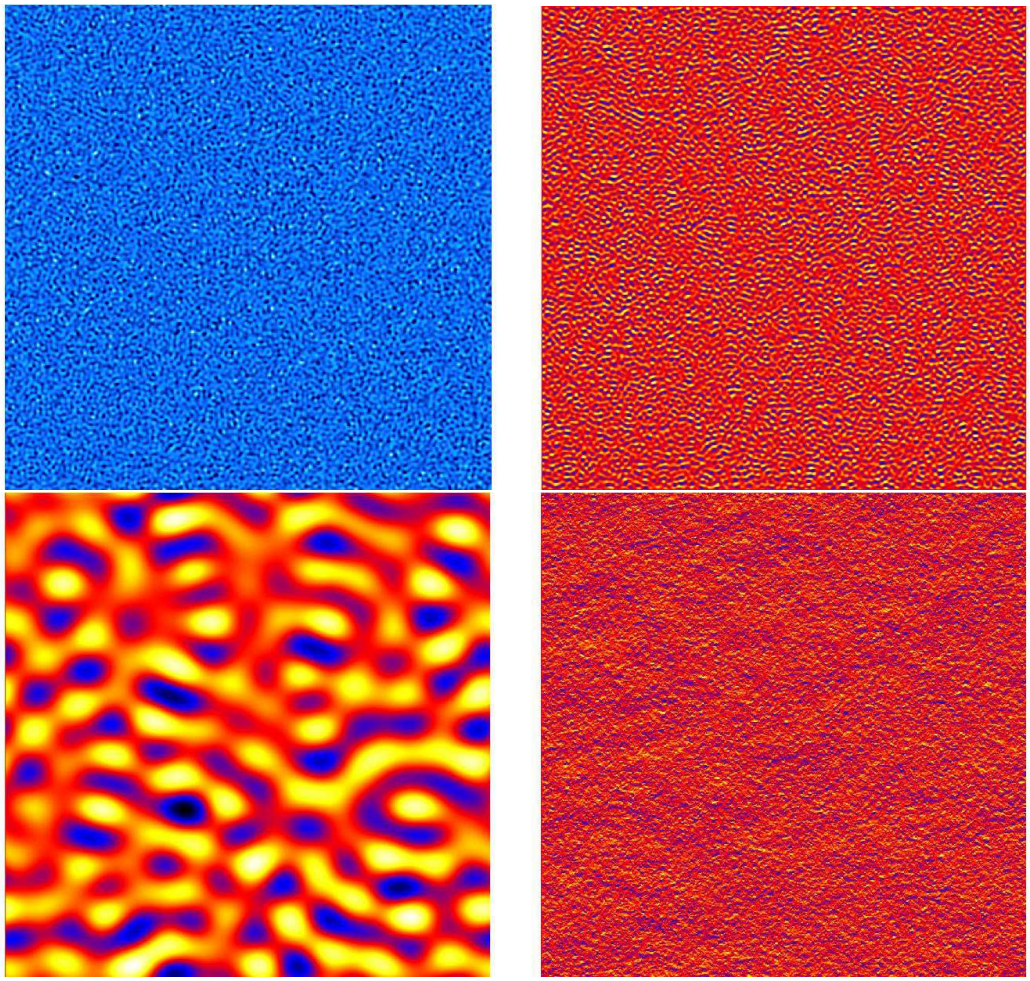


Figure 4.2: Example of Kolmorogov turbulence screens. The top left shows the density on a coarse scale, the top right shows the derivative, the bottom left shows the derivative on a medium scale, and the bottom right shows the screen on the smallest scales.

Cloud Screens

Reference: Ivezic (2007)

The cloud model consists of an absorptive probability for every pixel, and the photon has some probability of being absorbed by a photon as it propagates through the atmosphere.

Number of Layers

We use two clouds layers: one placed at the highest altitude of the turbulence screens and one place at the mid point of the turbulence screens.

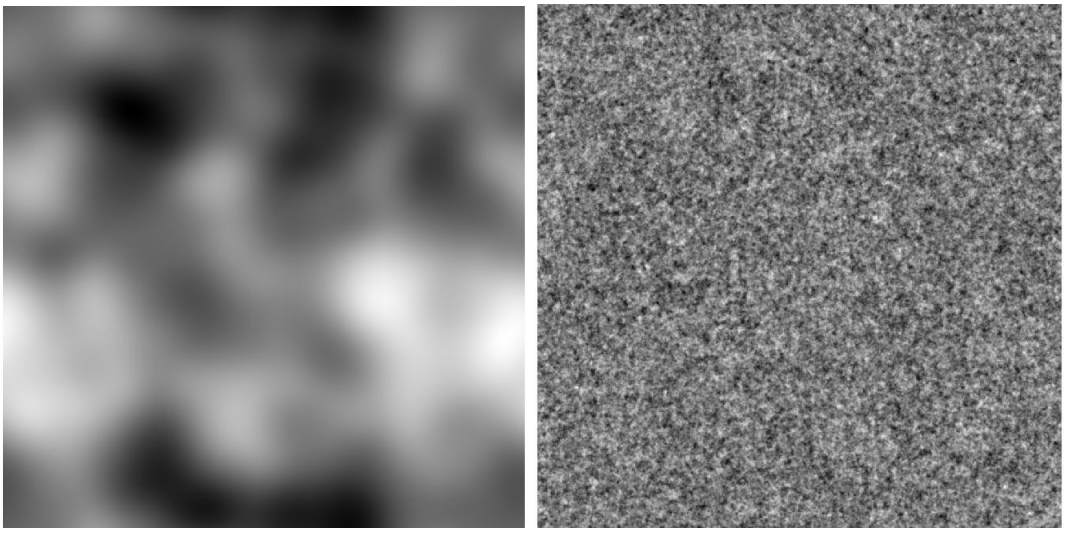


Figure 4.3: Examples of the cloud screens showing the variation of opacity on the top layer (left) and bottom layer (right).

Correlation Length

Reference: Ivezic (2007)

The cloud screens have an exponential structure function with a angular coherence scale of 2 degrees to be consistent with the measurements of Ivezic based on SDSS wide field images. This fixed angular scale is the reason that the images of the two cloud layers show a fine structure for the bottom layer and a coarse structure for the higher layer. Clouds of a fixed angular size will have a spatial size proportional to the height of the cloud layer. This is not a high fidelity model of actual clouds layers. We have not implemented a model that tracks the variations in cloud structure as function of height. The data is not readily available. Also the structure function estimated by Ivezic is assumed to be isotropic. Clouds clearly have non-isotropic structure that could be related to the wind direction. We decided on two layers of clouds each with a different wind velocity to create a partly realistic complexity that would at least mimic the difficulties for photometric calibration. Additional work is needed (see part IV).

Pixel Scale

The pixel scale of the cloud screen is 1 m (the same size as the coarse turbulence screen).

Wind Direction & Speed

Reference: NOAA NCEP/NCAR (database); Young, M. (ImSim Internal Document-9)

In order to determine where a photon hit a screen at a given layer, we first calculate the x and y position. We then calculate the pixel in the appropriate screen given two

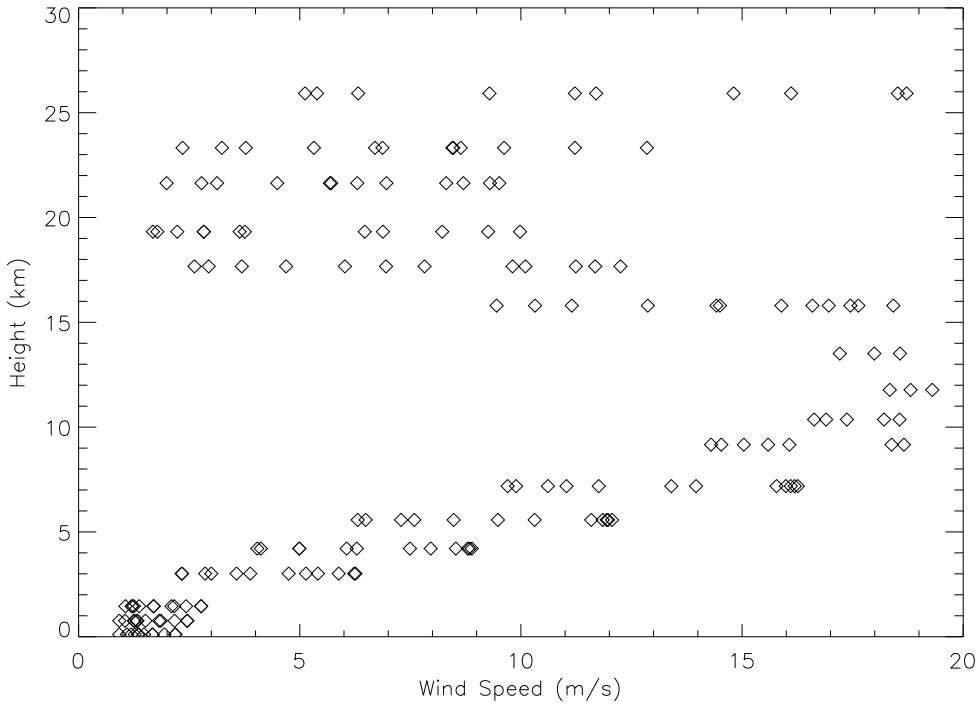


Figure 4.4: The sigma for the wind speed Weibull distribution parameter vs. height. The 12 points at each layer are the seasonal variation.

components of the wind vector for each screen. The arrival time of the photon then dictates exactly which pixel is used. For the wind model, we have a wind distribution that varies as a function of altitude. Both the wind direction and magnitude have a seasonal distribution at the LSST site. Using the NOAA NCEP/NCAR Reanalysis Monthly Database at the longitude and latitude of LSST, we fit the historical data to a Weibull distribution (k=2) for the wind velocity. The wind magnitude sigma values (m/s) are given below and shown in the figure. The wind measurements are a function of the pressure in the atmosphere which is converted to corresponding height.
We use a gaussian distribution for the wind direction, which is very anisotropic. The wind direction means (degrees from East) are given below.

The wind direction gaussian standard deviations (degrees) are given below.

Bulk Atmosphere Model

Reference: Bodhane et al. (1999); Green et al. (1988); Liou (2002); Grace, E. (ImSim Internal Document-10)

We track four components represent the bulk air located between the screens: the density of the atmosphere vs. height, the density of water vapor vs. height, the density of molecular oxygen vs. height, and the density of ozone vs. height. From the literature we

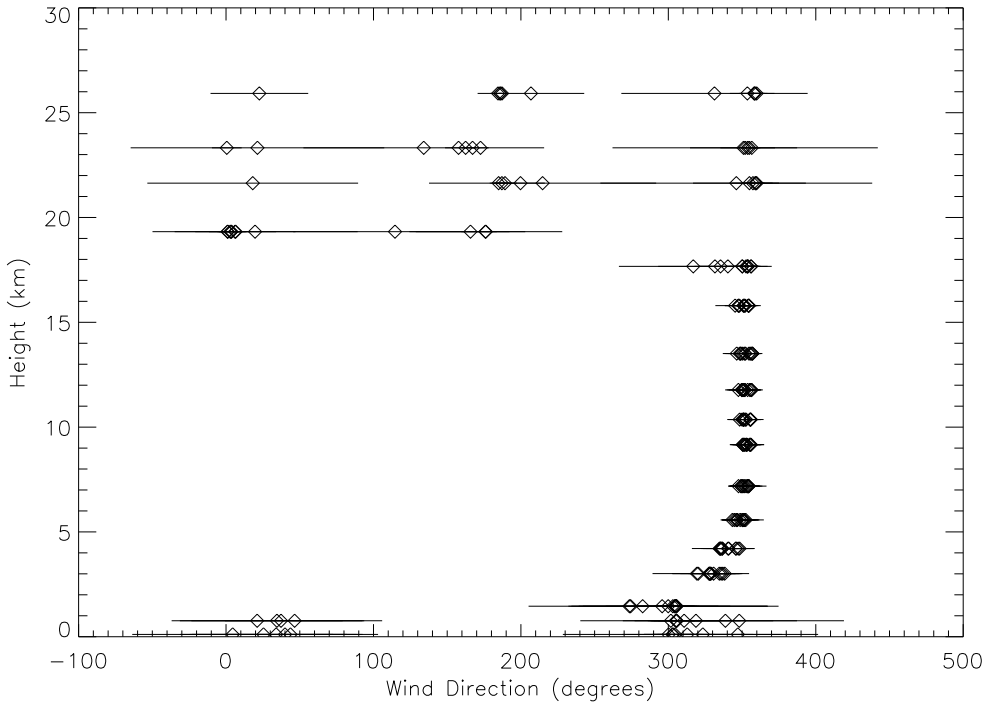


Figure 4.5: The average wind direction (degrees from east) vs. height. The 12 points at each layer are the seasonal variation. The lines show the variation in the wind directions. The wind direction is fairly anisotropic.

	Jan	Feb	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
10mb	18.72	18.52	9.29	5.12	16.11	22.79	20.25	14.81	11.7	6.32	5.39	11.22
20mb	8.45	9.62	6.7	2.35	8.64	12.85	11.22	8.47	6.87	3.78	3.24	5.32
30mb	8.7	9.51	5.71	1.99	6.3	9.3	8.31	6.96	5.68	3.13	2.78	4.49
50mb	2.83	3.64	1.78	2.83	6.88	9.98	9.26	8.22	6.47	3.76	2.23	1.68
70mb	2.94	2.62	3.69	6.95	10.1	12.25	11.68	11.24	9.81	7.82	6.02	4.69
100mb	10.32	9.45	11.15	14.42	16.96	18.42	17.63	17.45	16.59	15.89	14.5	12.87
150mb	17.99	17.21	18.57	20.76	23.51	25.76	25.18	25.22	25.21	25.35	22.78	20.46
200mb	18.81	18.34	19.3	20.83	23.65	26.35	25.99	26.55	26.79	27.04	23.54	20.7
250mb	16.9	16.63	17.37	18.56	21.32	24.06	23.71	24.51	24.66	24.84	21.16	18.21
300mb	14.53	14.3	15.04	16.07	18.66	21.17	20.84	21.63	21.64	21.8	18.38	15.59
400mb	9.89	9.7	10.61	11.76	13.96	15.99	15.78	16.26	16.19	16.1	13.4	11.03
500mb	6.49	6.31	7.29	8.48	10.31	11.97	11.85	12.06	11.94	11.59	9.48	7.59
600mb	4.12	4.04	4.99	6.06	7.49	8.89	8.84	8.8	8.53	7.96	6.29	4.99
700mb	2.33	2.34	3	3.88	5.14	6.26	6.23	5.88	5.41	4.75	3.57	2.86
850mb	1.06	1.21	1.37	1.7	2.16	2.77	2.77	2.43	2.1	1.69	1.25	1.2
925mb	1.25	1.05	0.92	1.26	1.82	2.45	2.46	2.17	1.86	1.51	1.33	1.31
1000mb	1.27	1.09	0.92	1.19	1.67	2.19	2.18	1.94	1.67	1.48	1.36	1.34

Table 4.2: The magnitude of the wind vector as a function of pressure (and therefore height) and month of the year for the LSST site.

constructed the density distribution in the plot below. We use the altitude of 2660 m for the location of LSST. These density distributions are used to determine the opacity as we describe in the opacity section. For the airmass dependence factor we use $\frac{1}{\cos z + 0.025 e^{-11 \cos z}}$. This factor then scales the density distributions below.

	Jan	Feb	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
10mb	186.2	185.7	184.5	22.6	359.9	358.6	358.8	358.2	353.6	331.3	206.8	187.2
20mb	162.5	167.2	172.6	21.3	0.5	354.9	356.6	353.8	351	352.1	134.1	157.7
30mb	187.1	184.9	189.2	18.1	359.9	357.4	359.3	358.9	355	346.1	214.7	199.7
50mb	165.8	175.9	176.2	6.5	3.7	1	3.4	2.9	1.2	6.1	19.7	114.6
70mb	331.6	316.9	335.4	350.2	353.7	353.9	356	356.2	353.9	352.7	350.1	340.4
100mb	347.5	345.3	347.6	351.6	354.5	354.2	354.6	354.5	352.1	351.2	350.9	348.2
150mb	348.4	349.1	349.1	351.7	355.6	356.4	357.1	357	354.7	352.3	350.4	346.4
200mb	349.8	350.4	349.9	352	355.3	356	356.4	356.5	353.8	351.6	350.9	347.5
250mb	350.9	351.4	350	352.3	355.4	355.9	355.8	355.6	352.7	350.9	351.1	348.4
300mb	353	353.4	351	352.9	355.6	356	355.6	355.1	352	350.5	351.6	350.4
400mb	354.3	354.5	350.4	351.8	353.8	354.6	353.8	352.7	349.2	347.6	349.5	350.7
500mb	350.2	350.5	346.9	348.7	351.1	352.5	351.7	350.2	346	343.4	344.7	346.5
600mb	336.7	334.7	335.3	340.9	346	348.3	347.4	345.7	340.5	336.5	335.8	335.3
700mb	328.9	319.4	320.3	327.9	334.6	338.2	336.9	335.8	330.7	327.6	328.4	328.4
850mb	299.9	273.6	274.4	282.6	295.8	305.4	305.3	305.6	304.9	304.4	303.1	304.5
925mb	46.5	37.3	348.1	305.7	301.7	305	305.5	310.7	318.7	338.7	21.1	34.5
1000mb	43.7	40	4.5	302.8	300.2	303.3	304.6	312.8	323.4	347.1	25.4	34

Table 4.3: The average direction of the wind vector as a function of pressure (and therefore height) and month of the year for the LSST site.

	Jan	Feb	Mar	Apr	May	June	July	Aug	Sep	Oct	Nov	Dec
10mb	1.03	1.15	5.24	33.14	12.06	3.78	4.03	9.03	11.88	63.18	36.16	2.47
20mb	2.63	2.48	5.58	86.02	10.06	5.3	5.77	18.54	36.32	89.97	81.64	9.14
30mb	6.44	6.04	7.96	71.49	10.85	6.62	6.07	16.19	38.27	92.19	77.01	16.49
50mb	37.12	16.38	51.85	27.44	9.56	5.91	6.38	7.38	12.82	40.96	69.57	69.05
70mb	38.52	50.44	17.71	8.69	9.06	6.87	6.02	5.38	6.23	7.25	11.18	19.9
100mb	9.33	13.43	9.13	7.71	8.22	6.95	5.65	5.44	5.94	6.26	6.15	8.06
150mb	9.17	12.17	8.6	7.76	8.19	6.54	4.93	5.87	5.72	6.05	6.44	8.21
200mb	8.9	11.89	9.04	8.34	8.81	6.75	5.27	6.56	5.91	6.31	6.88	7.96
250mb	8.48	11.46	9.18	8.91	9.18	6.96	5.97	7.39	6.18	6.48	7.22	7.77
300mb	7.91	11.43	9.33	9.53	9.42	7.17	6.65	7.89	6.44	6.63	7.41	7.58
400mb	7.65	12.09	9.69	10.3	9.98	7.86	7.93	8.74	7.14	6.96	8.16	8.33
500mb	8.33	14.27	10.54	10.25	10.74	8.81	9.15	9.51	7.91	7.18	9.27	9.56
600mb	13.05	18.74	12.03	10.88	11.66	10.28	10.62	9.92	8.72	8.34	10.25	13.26
700mb	25.88	30.15	17.85	13.15	11.21	12.02	10.97	9.58	9.97	11.58	14.8	20.26
850mb	67.73	68.39	32.42	25.12	18.96	15.65	12	12.5	14.46	29.59	63.9	70.25
925mb	47.24	68.6	70.99	65.41	32.57	17.25	12.42	24.66	30.25	48.54	58	58.02
1000mb	44.91	63.1	68.12	74.35	31.36	19.87	14.63	29.28	34.2	54.47	56.67	58.62

Table 4.4: The variation of the direction of the wind vector as a function of pressure (and therefore height) and month of the year for the LSST site.

Interactions with the Screens and Bulk Atmosphere

Refraction

Once the screen pixel location is determined for a given photon, we use the derivative of the turbulence to perturb the direction of a photon. This is the simple major interaction that will determine the seeing and contribute to the atmospheric PSF. The derivative is first normalized to the seeing at zenith at 500 nm, which is then input from the operational parameters. It is then scaled by two factors: the zenith dependence of the seeing ($\cos \text{zenith}^{\frac{-3}{5}}$) and the wavelength dependence of the refraction ($64.328 + \frac{29498.1}{146-\lambda_{\mu m}^{-2}} + \frac{255.4}{41-\lambda_{\mu m}^{-2}}$ divided by the same factor evaluated at 0.5 μm.

Cloud Grey Opacity

Reference: Ivezic (2007)

When a photon hits a cloud screen pixel it has some probability of being absorbed

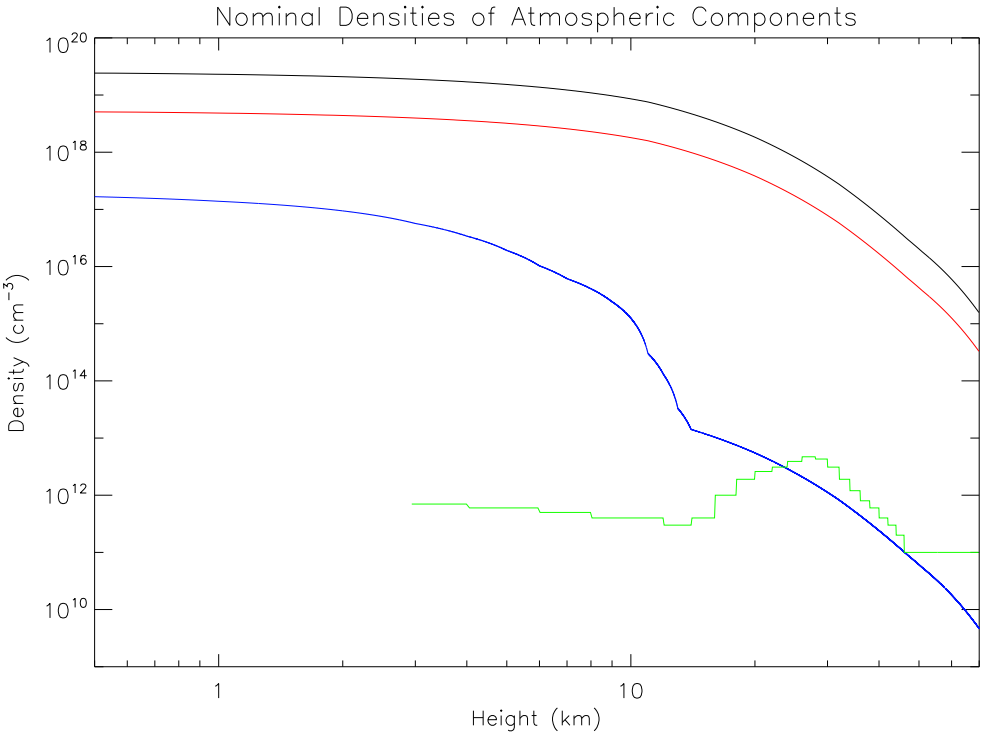


Figure 4.6: The atmospheric density distributions used in the opacity calculation. The black shows all molecular components, the green shows the ozone, the red shows the molecular oxygen, and the blue shows the water vapor.

(destroyed or lost by scattering outside the aperture). We use an average opacity of 0.1 magnitudes for each of two clouds screens, and a variation of 0.035 magnitudes to set the amplitude of variation from the screens. We vary each of these numbers by 1% from one atmosphere to the next. These values are typical of good conditions but are not really based on any database of measurements similar to what we use for wind data. Just as the spatial structure of the clouds needs more work so does the opacity model. For many purposes data can be used for science grade work with opacities up to ~1 magnitude of lost signal. Our current cloud model has an average total opacity of ~0.2 magnitudes of lost signal which is independent of wavelength (grey response). The cloud pattern moves with the same wind velocity as the turbulent screen at the same height. The cloud spatial structure has a resolution of 1m, which is the same as the coarse turbulence screen.

Opacity of the Atmosphere

Reference: Rothman (2009); Sander (2006); Thomas & Stramnes (1999); HITRAN (database); Grace, E. (ImSim Internal Document-10)

The opacity of the atmosphere in the photon raytrace is determined by calculating the local optical depth for each segment of the photons path. Between each atmospheric

screen the column density of each molecular species of the atmosphere is calculated by integrating the density profiles in our bulk atmosphere model and the path length the photon. This column density is then modulated by two factors. First, the overall column density is multiplied by a random gaussian (or log normal) with mean of 1 and σ given in the Table below for each component. Second, the local column density is also perturbed by a factor of $1\% \times \sqrt{N_{layers}}$ of the relative density of the height at each screen. In this way, the opacity will vary slightly from exposure to the next, and it will vary across the field in a complex way.

After the local column density is determined the optical depth is calculated by multiplying by the cross-sections as a function of wavelength from the plot below. The cross-sections are taken from the HITRAN atomic database as well as the Rayleigh scattering cross-section. The cross-sections are convolved with appropriate thermal and natural line width for a temperature model of the atmosphere that varies in height (see E. Grace technical note). Thus, the effective cross-section has a altitude dependence.

The probability the photon is destroyed along its particular path segment is then equal to $e^{-\tau(\lambda)}$ where $\tau = \sum_i \tau_i(\lambda) = \sum_i \int \sigma_i(\lambda, h) n_i(h) dh$.

Type	Variation	Distribution
Total Density	0.01	gaussian
Molecular Oxygen	0.01	gaussian
Ozone	0.01	gaussian
Water vapor	0.18	log-normal

Table 4.5: Variation of components of the atmosphere.

Atmospheric Dispersion

Reference: Filippenko (1982)

We use a formula below from the literature for atmospheric dispersion, which depends on air pressure (P, set to 520 mmHg), water vapor pressure (W, 8 mmHg),and ground temperature (T, 5 degrees C). This shifts the position of the photon depending on wavelength by a small angle. We subtract the mean positional difference of the entire field, so a photon in the center of the field with wavelength of 500 nm would receive no net dispersion. This is equivalent to the telescope operator knowing about the effect of atmospheric dispersion and then setting the pointing accordingly. However, photons of different wavelength and field positions would receive net displacement proportional to their distance from the zenith and in the direction to the zenith. The angular offset is given by the equation below,

$$\left(64.328 + \frac{29498.1}{146 - \frac{1}{\lambda^2}} + \frac{255.4}{41 - \frac{1}{\lambda^2}}\right) P \frac{1 + (1.049 - 0.0157T)\, 10^{-6} P}{720.883\,(1 + 0.003661T)} - \frac{0.0624 - \frac{0.000680}{\lambda^2}}{(1 + 0.003661T)W}$$

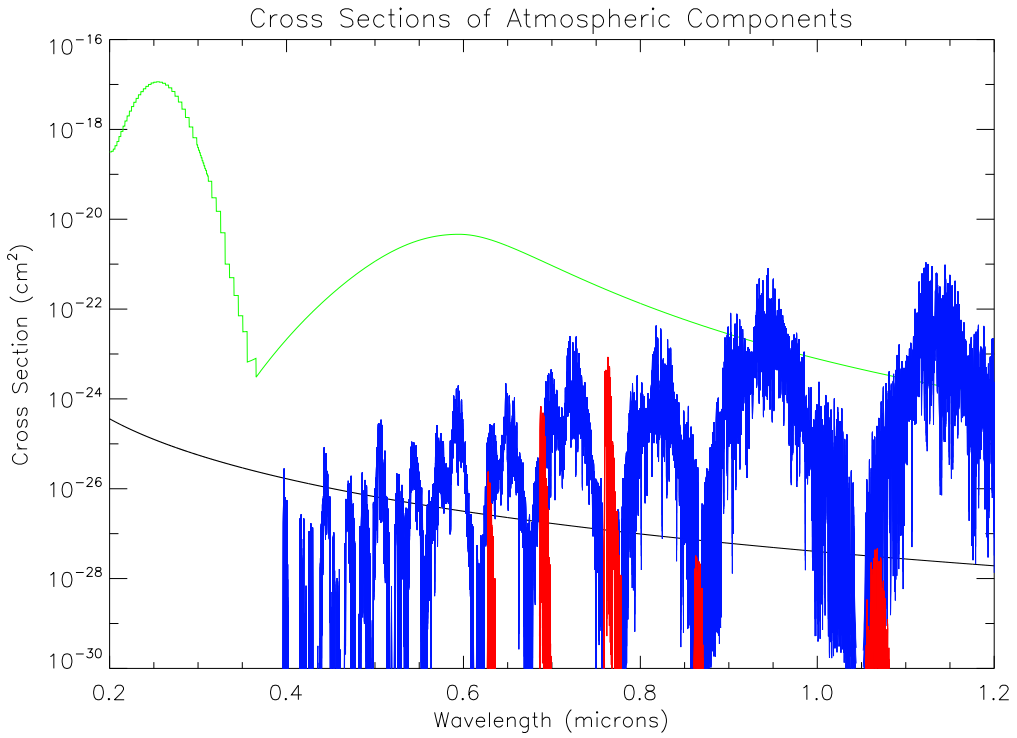


Figure 4.7: The atmospheric opacity cross-sections used in the opacity calculation. The black shows the rayleigh scattering cross-section, the green shows the ozone cross-section, the red shows the molecular oxygen cross-section, and the blue shows the water vapor cross-sections.

Chapter 5

Telescope & Camera

The telescope and camera model consists of a series of optical surfaces that define volumes of glass, silicon, and air. The raytrace follows the photons from the top of the telescope completely to the detector. We first discuss how the optical elements surfaces are defined and oriented, then discuss the possible photon interactions with optical surfaces, and finally describe the physics of photon (and electron) propagation in the volume between the surfaces.

Surface & Bulk Geometry

Below we describe the geometry and shape of the surfaces in the telescope and camera, as well as how the photons bounce between surfaces.

Ray Intercept Calculation

An essential calculation is to find the location of each photon hit on a given surface. Consider a photon at position (x, y, z) with a unit vector trajectory of (t_x, t_y, t_z). In order to find the intersection of that ray with a surface with height, $z = f(x, y)$, we move the ray a scalar distance, l, and minimize, δ

$$\delta = \left| z + t_z l - f(x + t_x l, y + t_y l) \right|$$

This equation can be solved exactly ($\delta = 0$) for quadratic surfaces, so we first approximate $f(x, y)$ with its parabolic form, and solve the equation for $l = l_0$. Then we adjust l near the value of l_0 using a golden bisection algorithm, and converge on the exact numerical answer. In practice, the function, f, for various optical surfaces is saved in memory and interpolated between grid points, which makes the computation extremely efficient and we set the tolerance to 0.01 microns. The mathematical efficiency of this part of the code has been studied in great detail. Following, we discuss the optical elements that make up the current model of the telescope and camera.

Optical Elements

Reference: Gressler, B., (LSST Collection-2097)

The basic optical design is described by a set of aspheres, using the standard asphere equation shown below. This is used to calculate the relative height of the optic as a function of radius between the inner and outer radius of the optic. The parameters listed below are input to the raytrace, and both the surface height as a function of radius and well as its derivative is pre-calculated and stored in a table for a fine radial grid.

$$z(r) = \frac{r^2}{R\left(1 + \sqrt{1 - (1+\kappa)\frac{r^2}{R^2}}\right)} + \alpha_3 r^3 + \alpha_4 r^4 + \alpha_5 r^5 + \alpha_6 r^6 + \alpha_7 r^7 + \alpha_8 r^8 + \alpha_9 r^9 + \alpha_{10} r^{10}$$

There is a different optical configuration when each of the 6 different filters are used. In the tables below, the higher order aspheric coefficient are listed so that the polynomial term will have unit of meters, whereas the other parameters are given in mm. The sign convention for the radius of curvature is opposite the ZEMAX convention.

Name	Type	R	Δz	Outer Rad	Inner Rad	κ	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}	Coating	Medium
M1	mirror	19835	0	4180	2558	-1.215	0	0	0	1.381e-27	0	0	0	0	mirror	vacuum
M2	mirror	6788	6156.2	1710	900	-0.222	0	0	0	-1.274e-23	0	-9.68e-31	0	0	mirror	vacuum
M3	mirror	8344.5	-6390	2508	550	0.155	0	0	0	-4.5e-25	0	-8.15e-33	0	0	mirror	vacuum
none	none	0	3630.5	0	0	0	0	0	0	0	0	0	0	0	none	vacuum
L1	lens	2824	3.568418	775	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L1E	lens	5021	82.23	775	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
L2	lens	0	412.64202	551	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L2E	lens	2529	30	551	0	-1.57	0	0	0	1.656e-21	0	0	0	0	lens A/R	vacuum
F	filter	5632	349.58	378	0	0	0	0	0	0	0	0	0	0	filter 0	glass
FE	filter	5530	26.60	378	0	0	0	0	0	0	0	0	0	0	none	vacuum
L3	lens	3169	42.40	361	0	-0.962	0	0	0	0	0	0	0	0	lens A/R	glass
L3E	lens	-13360	60	361	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
D	det	0	28.5	400	0	0	0	0	0	0	0	0	0	0	det A/R	silicon

Table 5.1: The aspheric coefficient and geometry of the optical elements for the u band configuration.

Name	Type	R	Δz	Outer Rad	Inner Rad	κ	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}	Coating	Medium
M1	mirror	19835	0	4180	2558	-1.215	0	0	0	1.381e-27	0	0	0	0	mirror	vacuum
M2	mirror	6788	6156.2	1710	900	-0.222	0	0	0	-1.274e-23	0	-9.68e-31	0	0	mirror	vacuum
M3	mirror	8344.5	-6390	2508	550	0.155	0	0	0	-4.5e-25	0	-8.15e-33	0	0	mirror	vacuum
none	none	0	3630.5	0	0	0	0	0	0	0	0	0	0	0	none	vacuum
L1	lens	2824	1.916631	775	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L1E	lens	5021	82.23	775	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
L2	lens	0	412.64202	551	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L2E	lens	2529	30	551	0	-1.57	0	0	0	1.656e-21	0	0	0	0	lens A/R	vacuum
F	filter	5632	349.58	378	0	0	0	0	0	0	0	0	0	0	filter 1	glass
FE	filter	5576	21.50	378	0	0	0	0	0	0	0	0	0	0	none	vacuum
L3	lens	3169	47.50	361	0	-0.962	0	0	0	0	0	0	0	0	lens A/R	glass
L3E	lens	-13360	60	361	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
D	det	0	28.5	400	0	0	0	0	0	0	0	0	0	0	det A/R	silicon

Table 5.2: The aspheric coefficient and geometry of the optical elements for the g band configuration.

Name	Type	R	Δz	Outer Rad	Inner Rad	κ	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}	Coating	Medium
M1	mirror	19835	0	4180	2558	-1.215	0	0	0	1.381e-27	0	0	0	0	mirror	vacuum
M2	mirror	6788	6156.2	1710	900	-0.222	0	0	0	-1.274e-23	0	-9.68e-31	0	0	mirror	vacuum
M3	mirror	8344.5	-6390	2508	550	0.155	0	0	0	-4.5e-25	0	-8.15e-33	0	0	mirror	vacuum
none	none	0	3630.5	0	0	0	0	0	0	0	0	0	0	0	none	vacuum
L1	lens	2824	0.772588	775	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L1E	lens	5021	82.23	775	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
L2	lens	0	412.64202	551	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L2E	lens	2529	30	551	0	-1.57	0	0	0	1.656e-21	0	0	0	0	lens A/R	vacuum
F	filter	5632	349.58	378	0	0	0	0	0	0	0	0	0	0	filter 2	glass
FE	filter	5606	17.90	378	0	0	0	0	0	0	0	0	0	0	none	vacuum
L3	lens	3169	51.10	361	0	-0.962	0	0	0	0	0	0	0	0	lens A/R	glass
L3E	lens	-13360	60	361	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
D	det	0	28.5	400	0	0	0	0	0	0	0	0	0	0	det A/R	silicon

Table 5.3: The aspheric coefficient and geometry of the optical elements for the r band configuration.

Name	Type	R	Δz	Outer Rad	Inner Rad	κ	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}	Coating	Medium
M1	mirror	19835	0	4180	2558	-1.215	0	0	0	1.381e-27	0	0	0	0	mirror	vacuum
M2	mirror	6788	6156.2	1710	900	-0.222	0	0	0	-1.274e-23	0	-9.68e-31	0	0	mirror	vacuum
M3	mirror	8344.5	-6390	2508	550	0.155	0	0	0	-4.5e-25	0	-8.15e-33	0	0	mirror	vacuum
none	none	0	3630.5	0	0	0	0	0	0	0	0	0	0	0	none	vacuum
L1	lens	2824	082134	775	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L1E	lens	5021	82.23	775	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
L2	lens	0	412.64202	551	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L2E	lens	2529	30	551	0	-1.57	0	0	0	1.656e-21	0	0	0	0	lens A/R	vacuum
F	filter	5632	349.58	378	0	0	0	0	0	0	0	0	0	0	filter 3	glass
FE	filter	5623	15.70	378	0	0	0	0	0	0	0	0	0	0	none	vacuum
L3	lens	3169	53.30	361	0	-0.962	0	0	0	0	0	0	0	0	lens A/R	glass
L3E	lens	-13360	60	361	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
D	det	0	28.5	400	0	0	0	0	0	0	0	0	0	0	det A/R	silicon

Table 5.4: The aspheric coefficient and geometry of the optical elements for the i band configuration.

Name	Type	R	Δz	Outer Rad	Inner Rad	κ	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}	Coating	Medium
M1	mirror	19835	0	4180	2558	-1.215	0	0	0	1.381e-27	0	0	0	0	mirror	vacuum
M2	mirror	6788	6156.2	1710	900	-0.222	0	0	0	-1.274e-23	0	-9.68e-31	0	0	mirror	vacuum
M3	mirror	8344.5	-6390	2508	550	0.155	0	0	0	-4.5e-25	0	-8.15e-33	0	0	mirror	vacuum
none	none	0	3630.5	0	0	0	0	0	0	0	0	0	0	0	none	vacuum
L1	lens	2824	-0.322542	775	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L1E	lens	5021	82.23	775	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
L2	lens	0	412.64202	551	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L2E	lens	2529	30	551	0	-1.57	0	0	0	1.656e-21	0	0	0	0	lens A/R	vacuum
F	filter	5632	349.58	378	0	0	0	0	0	0	0	0	0	0	filter 4	glass
FE	filter	5632	14.40	378	0	0	0	0	0	0	0	0	0	0	none	vacuum
L3	lens	3169	54.60	361	0	-0.962	0	0	0	0	0	0	0	0	lens A/R	glass
L3E	lens	-13360	60	361	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
D	det	0	28.5	400	0	0	0	0	0	0	0	0	0	0	det A/R	silicon

Table 5.5: The aspheric coefficient and geometry of the optical elements for the z band configuration.

Baffles

Reference: Gressler, B., (LSST Collection-2097)

The rays start begin the surface of the primary mirror as we describe in Chapter 3 as we sample the geometry pupil plane. In the current optical design (v3.3), however, the baffle locations would only prohibit rays from entering the optical path from other off-axis angles. Hence, our current sampling of the pupil plane already eliminates any rays that

Name	Type	R	Δz	Outer Rad	Inner Rad	κ	α_3	α_4	α_5	α_6	α_7	α_8	α_9	α_{10}	Coating	Medium
M1	mirror	19835	0	4180	2558	-1.215	0	0	0	1.381e-27	0	0	0	0	mirror	vacuum
M2	mirror	6788	6156.2	1710	900	-0.222	0	0	0	-1.274e-23	0	-9.68e-31	0	0	mirror	vacuum
M3	mirror	8344.5	-6390	2508	550	0.155	0	0	0	-4.5e-25	0	-8.15e-33	0	0	mirror	vacuum
none	none	0	3630.5	0	0	0	0	0	0	0	0	0	0	0	none	vacuum
L1	lens	2824	-0.571889	775	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L1E	lens	5021	82.23	775	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
L2	lens	0	412.64202	551	0	0	0	0	0	0	0	0	0	0	lens A/R	glass
L2E	lens	2529	30	551	0	-1.57	0	0	0	1.656e-21	0	0	0	0	lens A/R	vacuum
F	filter	5632	349.58	378	0	0	0	0	0	0	0	0	0	0	filter 5	glass
FE	filter	5640	13.60	378	0	0	0	0	0	0	0	0	0	0	none	vacuum
L3	lens	3169	55.40	361	0	-0.962	0	0	0	0	0	0	0	0	lens A/R	glass
L3E	lens	-13360	60	361	0	0	0	0	0	0	0	0	0	0	lens A/R	vacuum
D	det	0	28.5	400	0	0	0	0	0	0	0	0	0	0	det A/R	silicon

Table 5.6: The aspheric coefficient and geometry of the optical elements for the y band configuration.

the real baffle design would eliminate from sources in the field of view. However, we need to study the highly off-axis rays and the other support structure that would ultimately affect the background and possibly result in structured straylight.

Spider

Reference: Gressler, B., (LSST Collection-2097)

Below are the parameters describing the spider model. The spider structure itself is more complicated than our model, but this should serve as a good approximation. We model the diffraction as two layers of spiders at the two locations and at each location we have a spider offset at two different locations. The structure is considerably more complicated but this should get the average diffraction shadowing correct. We also model the diffraction of the innermost baffle. There are two ways the spider structure affects the rays in the raytrace. First, rays are eliminated that intersect the spider structure, or are outside of the spider design. Second, the spider structure induces diffraction, which we model using a full 3-D Monte Carlo diffraction calculation (described in a later section). We have also implemented the effective rotation of the spider relative to the camera that occurs during the exposure.

Type	Height	Width	X Center	Y Center
Outer Ring	8577	4419.6	0.0	0.0
Cross	8577	25.4	431.8	431.8
Cross	8577	25.4	-431.8	-431.8
Cross	7211.27	25.4	431.8	431.8
Cross	7211.27	25.4	-431.8	-431.8

Table 5.7: The geometric parameters of the spider support structure.

Focal Plane Layout

Reference: Rasmussen, A. (LSST Document-7822&7821)

Below is the nominal focal plane layout. The main effect of the focal plane layout is to determine if the rays intercepted a piece of active silicon. We have the flexibility to

specify arbitrary centers and numbers of pixels for each sensor (either amplifier or chip or arbitrary size). This is specified as a chipID, center in microns, pixel size, number of pixels, and the sub-amplifier region. We have a focal plane map where we assumed uniform pixels and a specified number of pixels in each direction for each chip with 16 amplifier regions. There are no gaps between amplifiers. This layout is then modified by the focal plane perturbation parameters described later. The surface heights of these chips from a perfectly flat layout is described later. The chip locations are plotted below for the STA model.

Gap Description	e2v	STA
Chip Gap Parallel (mm)	2.21	2.25
Chip Gap Serial (mm)	1.29	1.53
Raft Gap Parallel (mm)	2.46	2.50
Raft Gap Serial (mm)	1.54	1.78
Pixels Parallel	2002	2000
Pixels Serial	512	509
Pixel Size (microns)	10	10

Table 5.8: Parameters specifying the nominal focal plane layout.

Perturbations on the Nominal Design

Reference: Claver (priv. comm.); Todd, N. (ImSim Internal Document-11); Meert, A. (ImSim Internal Document-6); Lupu, A. (ImSim Internal Document-1)

The previous sections specified the nominal geometry of the optical elements. The positions of these elements, however, will not be perfect, and plays a critical role on the image quality. We have implemented five different kinds of perturbations on the optical elements: body-centered motions of the optical elements, surface perturbations of the optical elements described through Zernike polynomials, displacements and rotations of the chips in the focal plane, surface distortion of the CCD chip surface, and tracking errors of the entire system. In some cases, the parameterizations are redundant with each other. In this section, we describe the first two kinds of perturbations, and describe the other three in the following sections.

Body-centered motions are accomplished by considering the 6 degrees of freedom of each optical elements (3 rotations and 3 displacements). We have implemented this in the most efficient way by performing a set of displacements and Euler rotations on the *photon itself* prior to its intersection with the surface. This makes the calculation of the intercept efficient, because we calculate it in the geometric frame of the optic. The Euler transformation are saved in 3x3 numerical table matrices that are pre-calculated at the start of the simulation for every optic.

We have also implemented an algorithm a surface deformation on every mirror, lens, and detector surface by having a function describe the non-ideal perturbed surface. We use Zernike polynomials, the orthogonal set of polynomials for a cylindrical geometry, where the surface height perturbations is represented by

$$f(x,y) = z_{\mathrm{a}sphere}(r) + \sum_i a_i z_i(r,\theta)$$

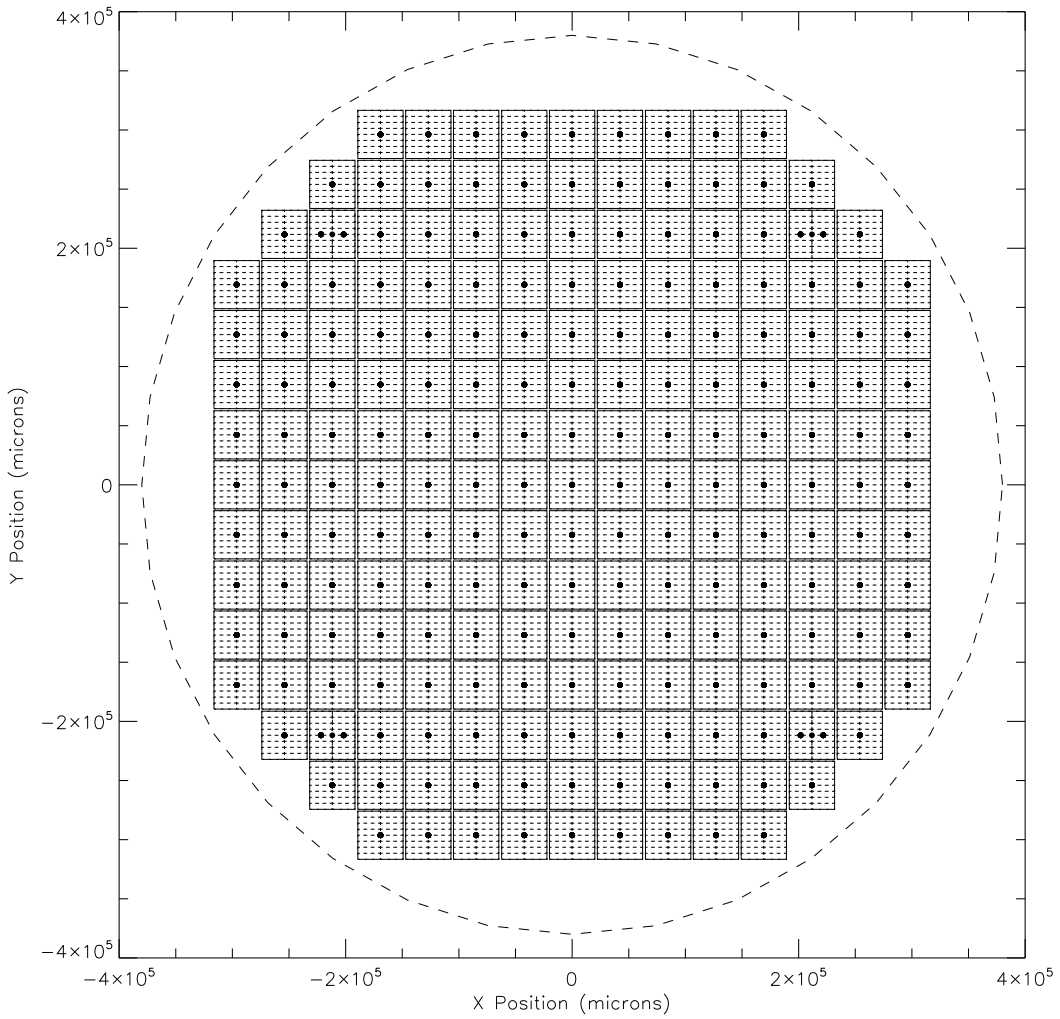


Figure 5.1: The map of the chip geometry of focal plane, and amplifier boundaries. The circle shows the full 2.1 degree circle, where an instance catalog is normally defined.

where the first term uses the asphere formula described previously, and the second term describes the Zernike polynomials (z_i) and their coefficients (a_i). We have implemented the first 22 terms (up to 5th order) for every mirror, lens, and detector surface. The zernikes are stored in numerical tables as a function of the polar variables for efficient computation.

The level of perturbation for every optic is ultimately set by the material physics of each optic as well as the effectiveness of the control system, and will vary as a function of time. Additionally, there are a large number of parameters (22+6 parameters × 12 optics = 336). However, several simplications and intuitive guesses for the level of perturbations can be made using the experience of the telescope and camera team.

The overall level of perturbations will be set by the current conditions at the site. Most likely the mechanical and thermal stresses will depend on the wind shake of the

telescope, temperature variation within the dome, gravitational stress which depends on the telescope elevation, pressure-induced misalignments due to the vacuum/air interfaces, and fabrication, support structures, or actuation errors. Furthermore, the control system is likely to behave more poorly when atmospheric seeing is higher, as the active optics system will attempt to correct some of the perturbations by sensing the misaligments in the wavefront images. We do not have a complete model for the perturbing physics and the effectiveness for response of the control system. Currently, and we scale the overall complete set of parameters so that the perturbations meet the engineering requirements for the total perturbation budget and scale in proportion to the seeing zenith dependence,

$$\text{Non-atmospheric PSF size} \sim (\sec(\text{zenith}))^{\frac{3}{5}}$$

In this way, our level of optical perturbations is set to the nominal expectation and also scales with the elevation of the observation. Some additional studies could be performed to increase the complexity of this scaling, and make it depend on other parameters (e.g. seeing, elevation, etc.). The perturbations are only one part of the non-atmospheric PSF, so we simulate the complete PSF and scale the perturbations (see documentation by N. Todd and validation task 2B)

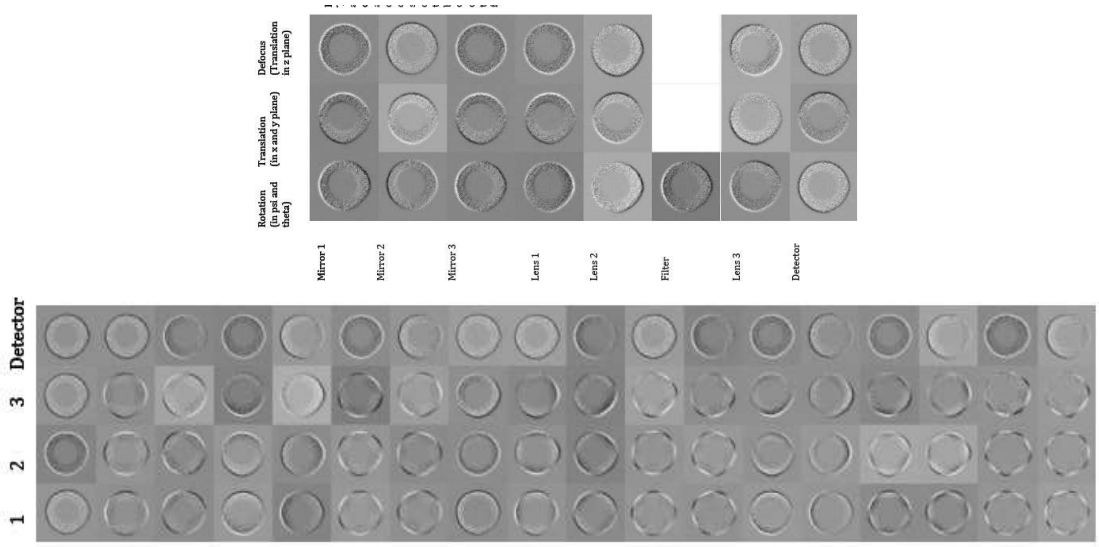


Figure 5.2: The resulting wavefront images in the corner raft when various perturbations are applied. The left plot shows the body misalignments of each optic, and the right plot shows perturbations places on the three mirrors and the detector surface.

We set the perturbations only on the possible parameters that are degrees of freedom of the active optics system. The assumption is that non-controlled degrees of freedom will be sufficiently well-designed as to not affect the PSF size significantly, compared to the degrees of freedom that are deliberately moved during telescope operation. This is not entirely correct. Furthermore, we assume the parameters are uncorrelated (except when the optics are fused each other), an assumption that may not be fully justified. We

copy the parameters of M1 to M3, since those optics are fused. We also couple the camera lens body parameters, since the camera is controlled together. We ignore the first four zenikes polynomials, since the linear terms are redundant with the body motions. We currently set the surface perturbations on the lenses to 0. The detector perturbations are set by a separate mechanism describe in another section. The perturbations are either drawn from a gaussian distribution or uniform distribution, and then possibly scaled by our SRD function above. The complete set of perturbed parameters are given in the table below. The relative normalization of each parameter is determined by looking at the relative change in the shape of a wavefront image for each parameters (described in detail by N. Todd documentation). The previous figures demonstrates examples of the wavefront images. Currently the perturbations are assumed to be independent for every visit. Normally, we are not simulating adjacent visits but this may change in the future. In summary, there are approximations to: 1) the degree of freedom chosen to have perturbations, 2) the time-decoherrence of the perturbations, and 3) the correlations between the perturbations that will have to be revisited in future versions.

Detector Misalignments

Reference: Rasmussen, A. (LSST Document-7822&7821)

The decenter of each CCD as well as an overall rotation is set by the focal plane data of Rasmussen in 7822 & 7821. Typically, the displacements are around 40 μm and the rotations are around 1 arcminute. These detector misalignments are used to infer the location of the pixel grid in the raytrace, and are also calculated by moving the photon to the local reference frame of the relevant detector. These displacements are fixed for all of the simulations and are not varied for different exposures nor are scaled with the scaling function.

Detector Surfaces

Reference: Rasmussen, A. (LSST Document-7822&7821)

Each detector surfaces has a distortion pattern as given by the focal plane data of Rasmussen in 7822 & 7821. The local surface height also includes a tilt as expressed through a Zernike expansion, which predicts a relative chip height for every pixel and is stored in a table. The surface height variations are typically around 5 μm. The detector surface pattern is fixed for all the simulations and are not varied for the different exposures nor are scaled with the scaling function.

Tracking

Reference: Claver (priv. comm.); Warner, M. (priv. comm.)

The entire telescope and camera system is essentially perturbed by a tracking model. The model simply perturbs the position of the photons in the reference frame of the camera and telescope, and represents the residual tracking perturbations that might be expected

Name	Degree of Freedom	σ or low value	μ or high value	Type	Multiplied by Scale?	Optics
M13	phi	0	6.28	uniform	no	M1,M3
M13	psi	0	6.28	uniform	no	M1,M3
M13	theta	4.94e-7	0	gaussian	yes	M1,M3
M13	x	2.28e-3	0	gaussian	yes	M1,M3
M13	y	2.28e-3	0	gaussian	yes	M1,M3
M13	z	2.28e-3	0	gaussian	yes	M1,M3
M13	Zernike 4	5.16e-5	0	gaussian	yes	M1,M3
M13	Zernike 5	5.16e-5	0	gaussian	yes	M1,M3
M13	Zernike 6	5.16e-5	0	gaussian	yes	M1,M3
M13	Zernike 7	5.16e-5	0	gaussian	yes	M1,M3
M13	Zernike 8	5.16e-5	0	gaussian	yes	M1,M3
M13	Zernike 9	5.16e-5	0	gaussian	yes	M1,M3
M13	Zernike 10	2.356e-5	0	gaussian	yes	M1,M3
M13	Zernike 11	2.356e-5	0	gaussian	yes	M1,M3
M13	Zernike 12	2.356e-5	0	gaussian	yes	M1,M3
M13	Zernike 13	2.356e-5	0	gaussian	yes	M1,M3
M13	Zernike 14	2.356e-5	0	gaussian	yes	M1,M3
M13	Zernike 15	1.59e-5	0	gaussian	yes	M1,M3
M13	Zernike 16	1.59e-5	0	gaussian	yes	M1,M3
M13	Zernike 17	1.59e-5	0	gaussian	yes	M1,M3
M13	Zernike 18	9.88e-6	0	gaussian	yes	M1,M3
M13	Zernike 19	9.88e-6	0	gaussian	yes	M1,M3
M13	Zernike 20	9.88e-6	0	gaussian	yes	M1,M3
M13	Zernike 21	9.88e-6	0	gaussian	yes	M1,M3
M2	phi	0	6.28	uniform	no	M2
M2	psi	0	6.28	uniform	no	M2
M2	theta	4.94e-7	0	gaussian	yes	M2
M2	x	2.28e-3	0	gaussian	yes	M2
M2	y	2.28e-3	0	gaussian	yes	M2
M2	z	2.28e-3	0	gaussian	yes	M2
M2	Zernike 4	8.36e-5	0	gaussian	yes	M2
M2	Zernike 5	8.36e-5	0	gaussian	yes	M2
M2	Zernike 6	8.36e-5	0	gaussian	yes	M2
M2	Zernike 7	8.36e-5	0	gaussian	yes	M2
M2	Zernike 8	8.36e-5	0	gaussian	yes	M2
M2	Zernike 9	8.36e-5	0	gaussian	yes	M2
M2	Zernike 10	4.332e-5	0	gaussian	yes	M2
M2	Zernike 11	4.332e-5	0	gaussian	yes	M2
M2	Zernike 12	4.332e-5	0	gaussian	yes	M2
M2	Zernike 13	4.332e-5	0	gaussian	yes	M2
M2	Zernike 14	4.332e-5	0	gaussian	yes	M2
M2	Zernike 15	4.332e-5	0	gaussian	yes	M2
M2	Zernike 16	2.8e-5	0	gaussian	yes	M2
M2	Zernike 17	2.8e-5	0	gaussian	yes	M2
M2	Zernike 18	2.8e-5	0	gaussian	yes	M2
M2	Zernike 19	2e-5	0	gaussian	yes	M2
M2	Zernike 20	2e-5	0	gaussian	yes	M2
M2	Zernike 21	2e-5	0	gaussian	yes	M2
camera	phi	0	6.28	uniform	no	L1,L1E,L2,L2E,F,FE,L3,L3E
camera	psi	0	6.28	uniform	no	L1,L1E,L2,L2E,F,FE,L3,L3E
camera	theta	3.34e-5	0	gaussian	yes	L1,L1E,L2,L2E,F,FE,L3,L3E
camera	x	6.8e-3	0	gaussian	yes	L1,L1E,L2,L2E,F,FE,L3,L3E
camera	y	6.8e-3	0	gaussian	yes	L1,L1E,L2,L2E,F,FE,L3,L3E
camera	z	6.8e-3	0	gaussian	yes	L1,L1E,L2,L2E,F,FE,L3,L3E

Table 5.9: Optical perturbation parameter tolerances and how they are correlated with one another.

after a successful tracking system operated. We have implemented a gaussian random walk model that varies throughout the exposure pair. Every 0.1 seconds, a random walk step is taken in both elevation, azimuth, and rotation of the camera. The mean step is calculated so that the *final* RMS size of the jitter after 15 seconds is given by the table below. Thus, the temporal spectrum is purely white up to 0.1 seconds. Between every 0.1 seconds, the jitter is interpolated.

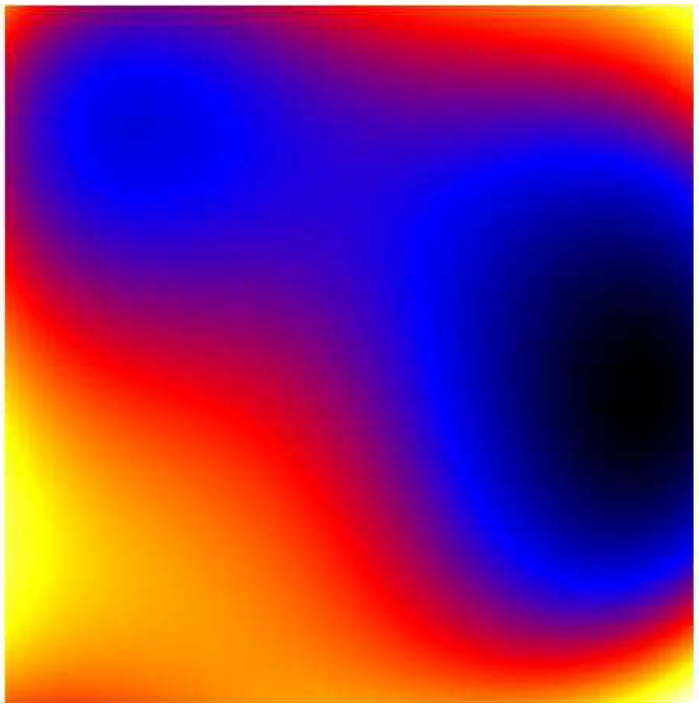


Figure 5.3: An example of the height variations on an individual CCD.

Parameter	Value
Rotation Rate	Varies
RMS Elevation Tracking Jitter	0.02 arcseconds
RMS Azimuth Tracking Jitter	0.02 arcseconds
RMS Rotation Tracking Jitter	1 arcseconds
Jitter Time	0.1 second

Table 5.10: Tracking model parameters.

Interactions with Surfaces and Media

Transmission and Reflection

Most of the optical elements have a coating that may affect the photon. The different types are listed in Table 5.1-5.6, and are described as either a reflection probability as a function of wavelength or a transmission probability as a function of wavelength. The transmission probability is always 1 minus the reflection probability. When the photon reaches a coating, we use its wavelength to decide whether it is reflected or transmitted. In the case of a lens, filter, or detector if it is reflected backwards, it is considered stray light and could be either destroyed or continued to be simulated when the ghost option is used. Photons that are reflected have their direction changed relative to the normal of that surface by the reflected angle equal to the incident angle.

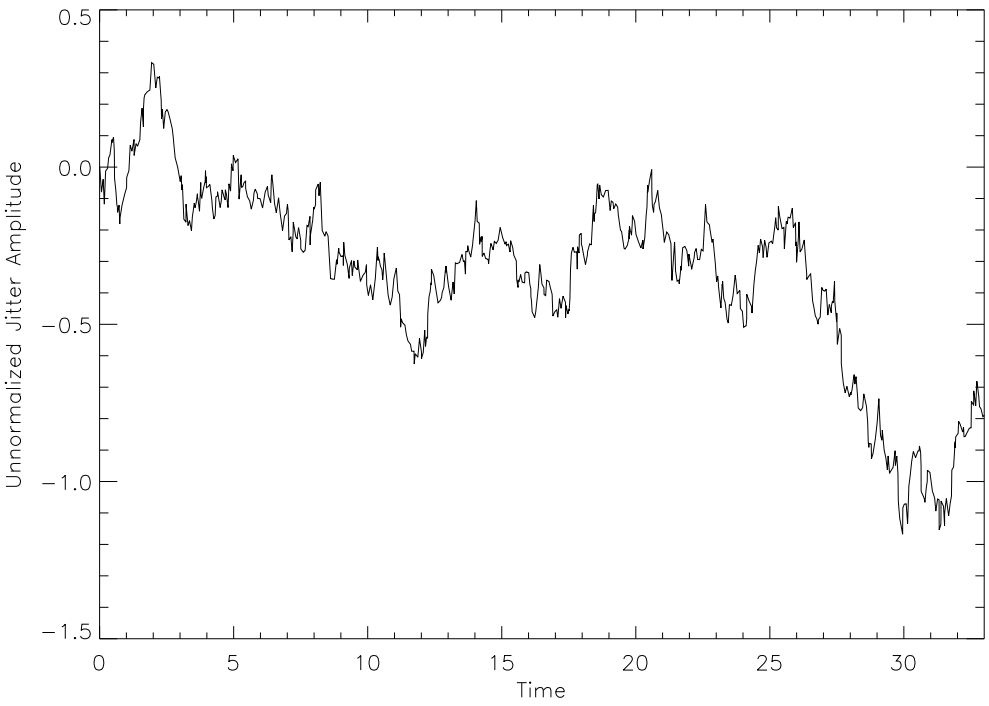


Figure 5.4: An example of the random walk of the tracking model, which has a characteristic time scale of 0.1 second. Photons are shifted in angle by a multiple of this sort of tracking jitter.

With the anti-reflective coatings and filter coatings for LSST, they will be achieved by depositing a multi-layer coating. With a multi-layer, the reflection and transmission probabilities are not only a function of wavelength but depend on the angle, that can only be calculated properly having a complete layering design and performing a full interference calculation. In the absence of this design, we assume that the transmission function behaves as the nominal transmission curve at the average incidence angle (θ_0 = 19.94 degrees). When the photon with wavelength, λ, hits a multi-layer at a different angle, θ, we use a wavelength $\lambda' = \frac{\cos \frac{\arccos \theta}{n_m}}{\cos \frac{\arccos \theta_0}{n_m}}$. Here n_m is the index of the refraction of the multi-layer, which we set to be 2. This slightly shifts the transmission curve depending on the wavelength, but is not particularly significant unless n_m is close to 1.

With reflection and transmission, we have implemented a rather important optimization of the raytrace process we call dynamic transmission. Because a large number of photons get removed because of the process of transmission, it is not optimal to continue to trace photons through the entire simulation that begin with a wavelength that makes it likely to get destroyed. This is obviously true for photons with wavelength far outside the characteristic filter transmission curve. Therefore, at the beginning of the simulation we record the probability, p, that photons make it through each optic. Later photons can then be removed at the start of the simulation at slightly lower probability (in practice, 1-(p

+ 1%)) than the actual running set of probabilities. In this way, the simulation quickly learns that photons of certain wavelength are unlikely to make it to detection. As the algorithm optimizes, photons that actually reach a given optic have the random number recalled from before, and make it through the optic with much higher probability. In this way, the filter leaks, for instance, can still be simulated without wasting significant computation time.

Coatings

The coatings for each of the optics is described below. The appropriate coating for each optical surface is listed in Table 5.1-5.6.

Mirror Reflectivity

Reference: Jones, Ivezic, Claver (Document-1777)

The reflectivity for the aluminized coatings on the mirrors is given by the wavelength-dependent function shown in the Figure below. This curve is identical to the Document-1777 data.

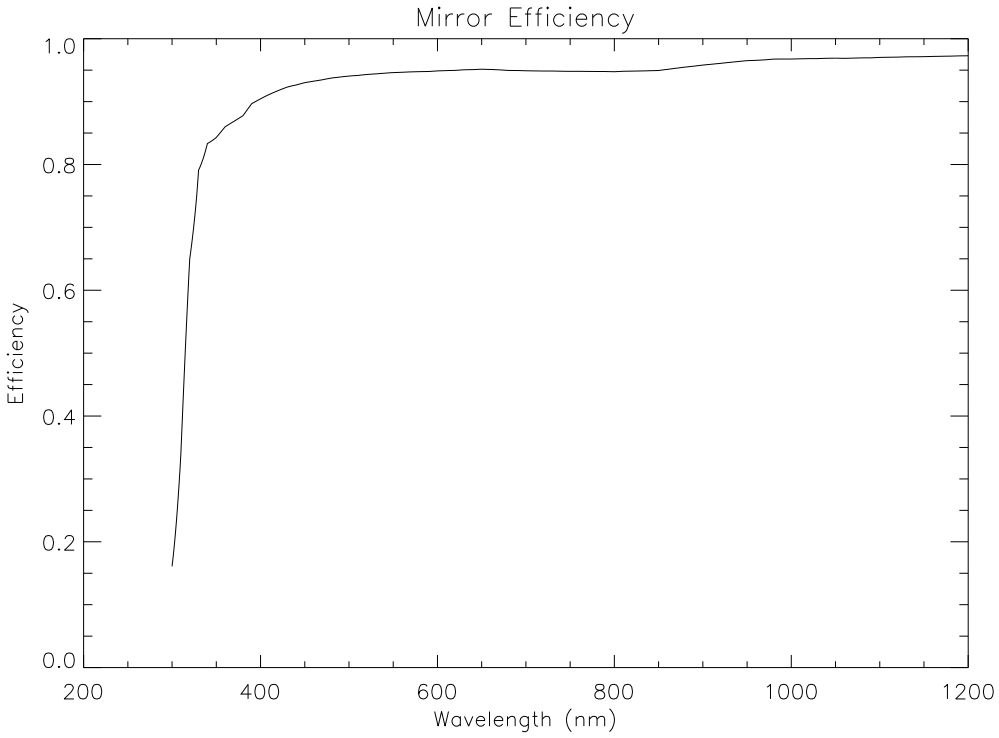


Figure 5.5: The mirror reflectivity as a function of wavelength.

Lens A/R Coating

Reference: Jones, Ivezic, Claver (Document-1777)

The lens anti-reflective coatings are shown in the wavelength-dependent function shown below. This coating is place on all the lens surfaces as listed in Table 5.1-5.6. The curve is computed by calculating the square root of the Document-1777 data, since we place the curve on all pairs of surfaces. This distinction is important for accurate ghost simulations. We use the multi-layer physics formula described in the previous section.

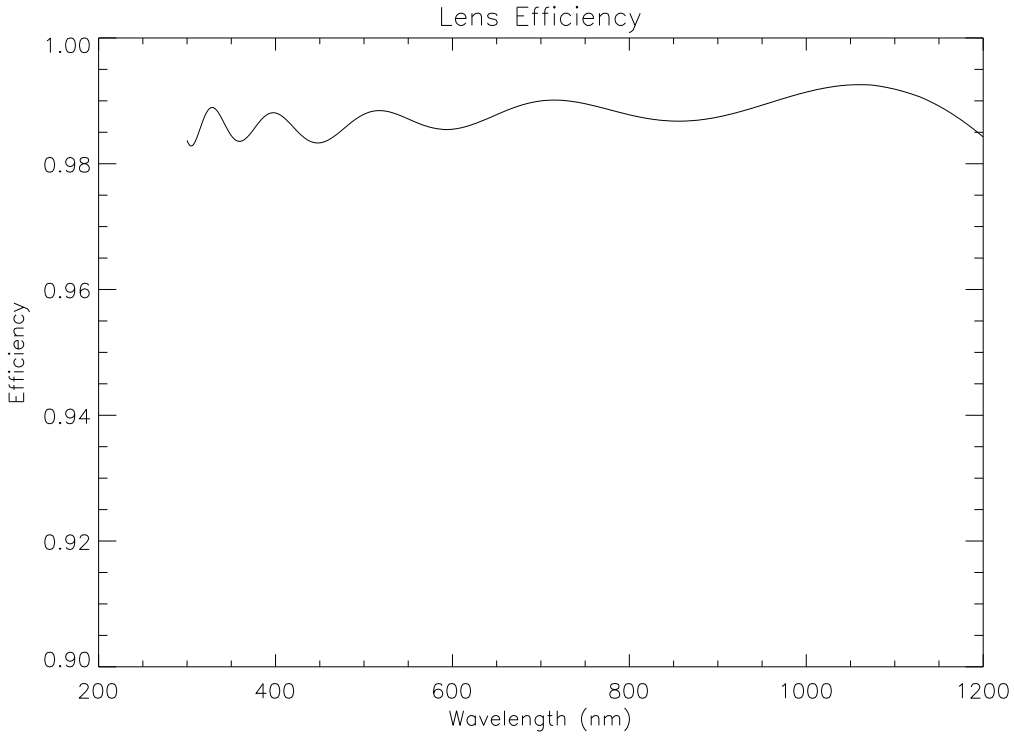


Figure 5.6: The lens anti-reflective coating transmission probability as a function of wavelength.

Filter Transmission

Reference: Jones, Ivezic, Claver (Document-1777)

The filter multi-layer coatings transmission curves are shown in the wavelength-dependent functions plotted below. The curve is identical to the the Document-1777 data. We currently place this coating only on the front surface of the filter (and no coating on the back surface), which may or may not be appropriate with the final filter design. This will affect the ghost simulations. We use the multi-layer physics described in the previous section.

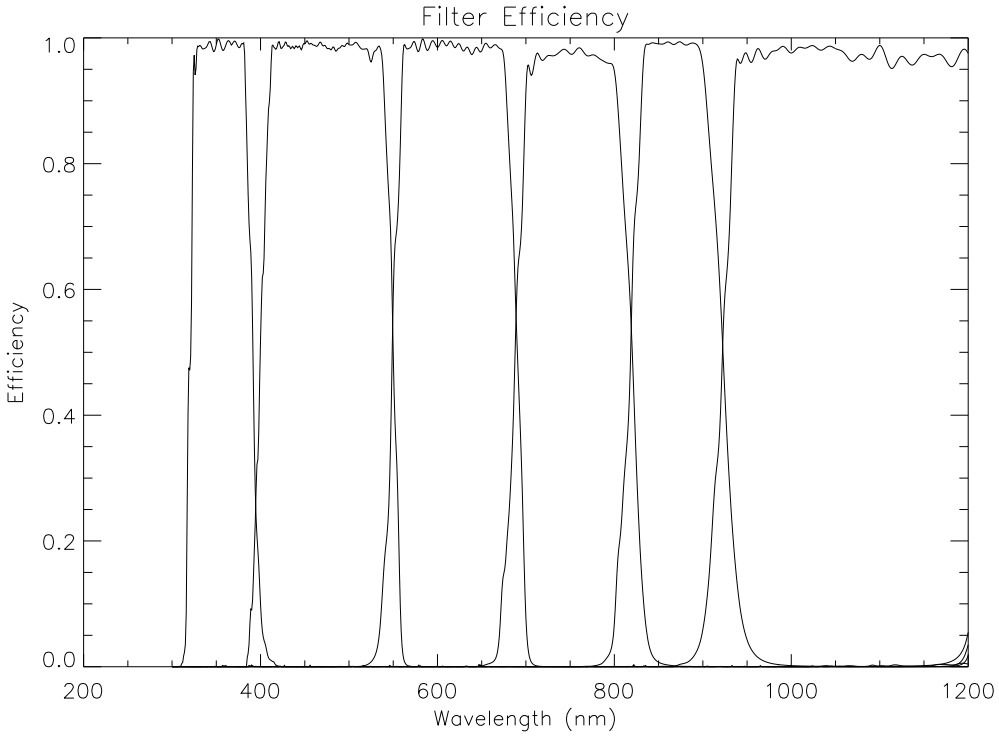


Figure 5.7: The filter transmission probability curve as a function of wavelength. The filters are from left to right: u,g,r,i,z,y.

Detector A/R

Reference: Jones, Ivezic, Claver (Document-1777)

The detector A/R coating transmission curve is shown in the curve plotted below. The curve is calculated by first measuring the photon conversion efficiency through the silicon layer and then divided that by the QE curve in document-1777. Since the transmission efficiency cannot be greater than 1, the implied QE of the photon simulator does not agree perfectly with document-1777 at long wavelength.

Large Angle Scattering

Reference: Grace, E. (ImSim Internal Document-4)

To represent the incoherent scattering that occurs from micro-roughness on the mirror surfaces, we have a simple empirical model for large angle scattering. The micro-roughness of mirrors (at the nm level) primarily causes photons to very large angle (few arcminutes). At the current time, we have not implemented a physical model for this, but instead studied the large angle light from stars from SDSS and Gemini. We use a model that empirically fit the Gemini data (which was a smaller amount of scattering) that followed a radial probability of

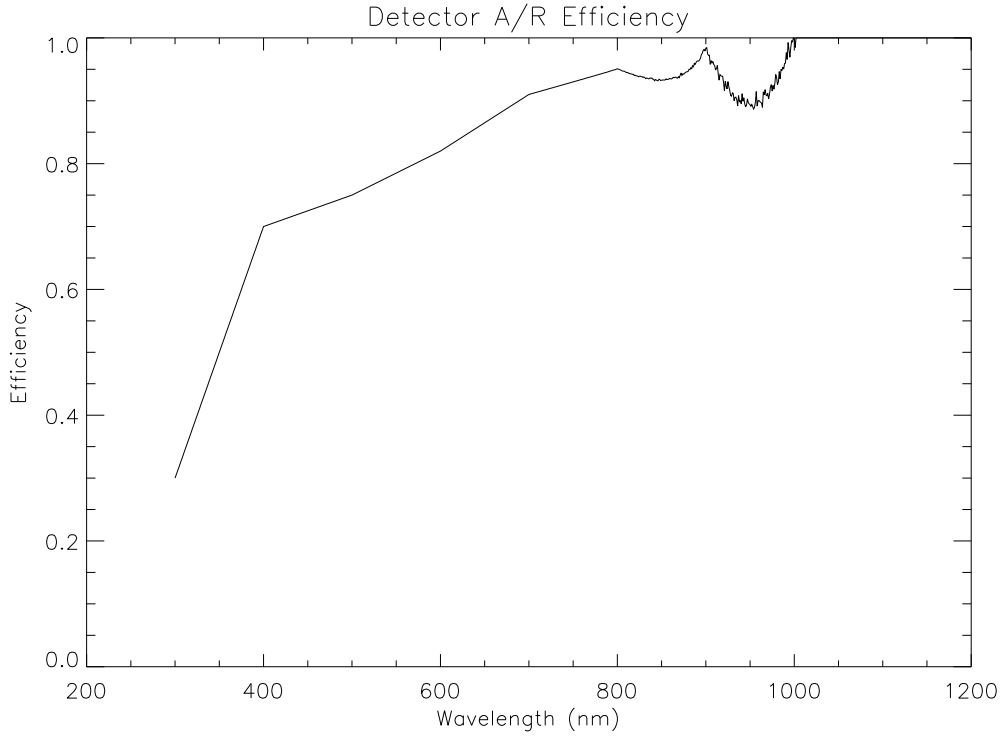


Figure 5.8: The anti-reflective and photo-absorption transmission probability curve as a function of wavelength on the front surface of the CCD.

$$\frac{1}{1+\left(\frac{r}{1^\circ}\right)^{3.5}}$$

The fraction of light in this diffuse halo compared to the core is 13.5%. Therefore, at the start of the telescope simulation the photon has a 13.5% probability of simply scattered by the angle implied by the above formula.

Refraction

There are essentially three kinds of material in the bulk volume of the telescope simulation: glass, silicon, and air. These get defined by the optical surfaces in Table 5.1-5.6. When a ray passes from one material to the next, the path of the ray will get refracted. We refract the ray by computing the normal to the surface, and apply Snell's law using a three vector formulation. Following, we discusses the index of refraction of the different media.

Glass

Reference: Sellmeier (1871); Schott Glass Catalog (database)

The glass in the lenses has an index of refraction as a function of wavelength as shown below. We use the Sellmeier equation where the index of refraction is given by

$$n = \sqrt{1 + \frac{B_1\lambda^2}{\lambda^2 - C_1} + \frac{B_2\lambda^2}{\lambda^2 - C_2} + \frac{B_3\lambda^2}{\lambda^2 - C_3}}$$

where the values for the fused silica constants are given by the table below.

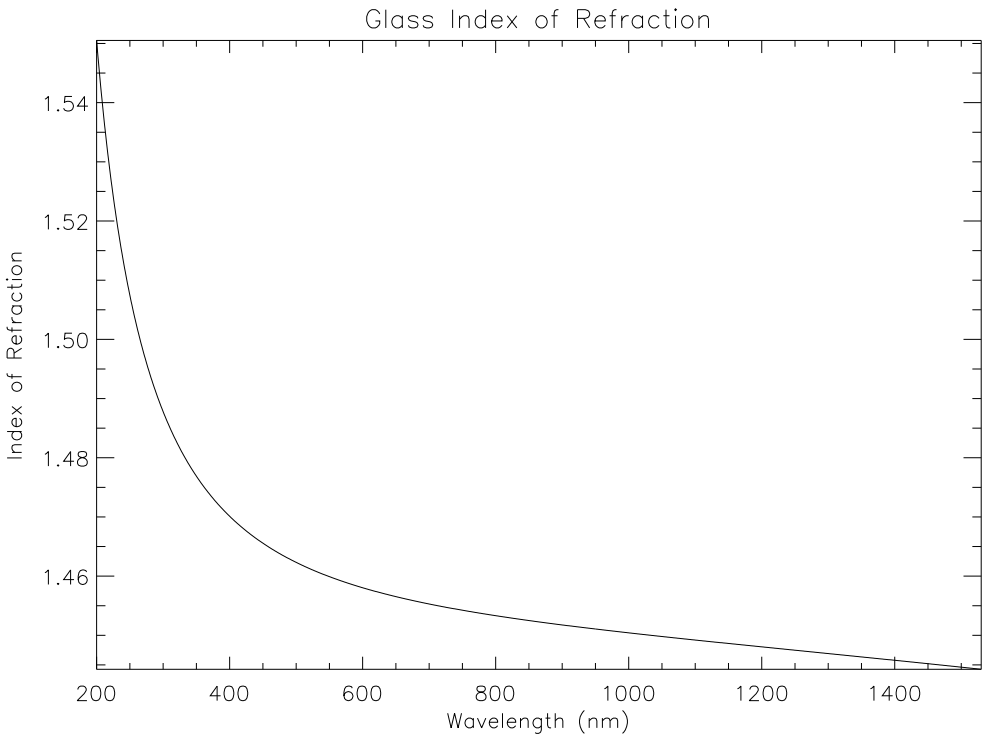


Figure 5.9: The index of refraction of the glass in the lenses as a function of wavelength.

Parameter	Value
B_1	0.6961663
C_1	0.00467914826
B_2	0.4079426
C_2	0.0135120631
B_3	0.8974794
C_3	97.9340025

Table 5.11: Glass coefficients for the index of refraction glass model.

Silicon

Reference: Phillip & Taft (1960)

The index of refraction of Silicon as a function of wavelength is shown below.

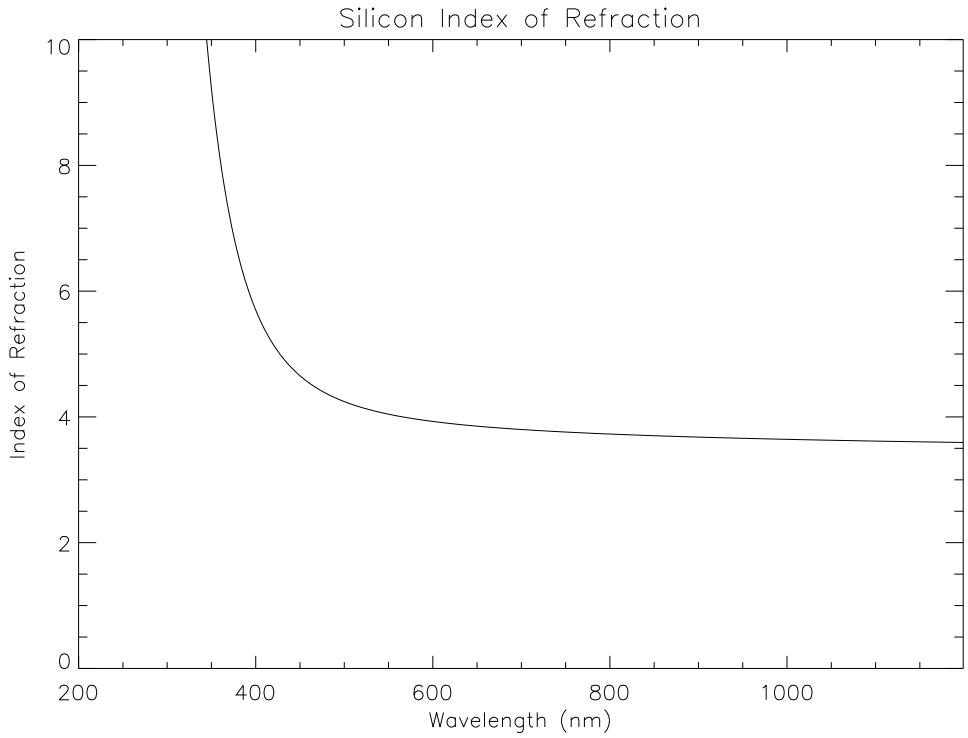


Figure 5.10: The index of refraction of the silicon in the CCDs as a function of wavelength.

Dome Air

When the photon is in the air, we assume the index of refraction is 1 (a perfect vacuum). However, we perturb the angle of the photon by a gaussian with with of 0.1 arcseconds before it reaches the primary mirror to represent the refraction due to micro-turbulence inside the dome. This induces a perfectly isotropic PSF addition due to the dome seeing with no correlation pattern (a model that may oversimplify the actual case).

Diffractive Effects

Reference: Freniere, Gregory, & Hassler (1999)

Diffraction of light does not occur in the geometric optics approximation of tracing rays. However, we use a clever algorithm to mimic its effects. Freniere, Gregory, & Hassler describe a numerical algorithm where the shortest distance to edge of any optic

(d) is recorded as the rays pass through the normal raytracing process. Then as the photons arrive on the detector their position is shifted by a distance

$$p_l \frac{\lambda}{4\pi d}$$

where p_l is the plate-scale of the telescope, and λ is the wavelength of the photon. We made the algorithmic modification to shift the photon in the direction to the nearest optic edge as well, and the method remarkably reproduces both the Airy diffraction envelope as well as the correct power as a function of angle in the diffraction spikes. Notice, that most photons will experience a small shift since d is typically very large. We consider the surfaces described in the spider section along with the primary mirror as possibly diffracting surfaces.

Photoelectric Conversion in Silicon

Reference: Rasmussen, A. (priv. comm. model); Rajkanan, Singh, & Shewchun (1979)

Once the photon enters the silicon, it has a probability of undergoing photoelectric conversion according to the mean free path of a photon in Silicon. Below the mean free path vs. wavelength is plotted as a function of various temperatures (where we use the nominal CCD temperature at 173 K in the simulations). Thus, this curve implies an effective quantum efficiency at long wavelength, since the long wavelength photons have a significant probability of not converting at all after propagating through 100 microns of Silicon.

Parameter	Value
Silicon Thickness (microns)	100
CCD temperature (K)	173

Table 5.12: CCD parameters affecting the photoelectric conversion model.

Electron Diffusion

Reference: Rasmussen, A. (priv. comm. model)

After conversion, the electrons are drifted to the backside of the CCD following Rasmussen's electron drift model. An electric field profile vs. height is calculated using a dopant model and applied bias voltage using the parameters and formula listed below.

$$E(z) = \frac{V}{t_{Si}} + \frac{q}{\epsilon_0 \epsilon_{Si}} \int_{t_{Si}}^{z} dz n_d(z)$$

where V is the overdepletion potential, t_{Si} is the silicon thickness, ϵ_{Si} is the permittivity in Silicon, and n_d is the doping density function which is given by

$$n_d(z) = n_{bulk} + n_b \times e^{-\frac{(t_{Si}-z)}{s_b}} + n_f \times e^{-\frac{z}{s_f}}$$

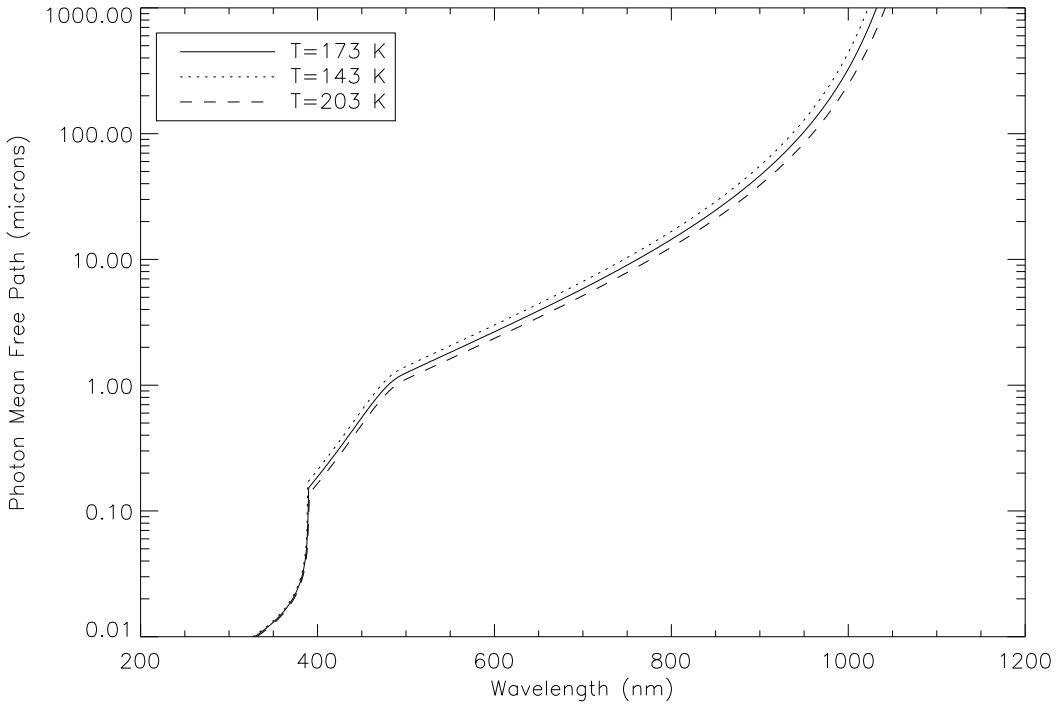


Figure 5.11: The mean free path of a photon in silicon as a function of wavelength for different temperatures.

Then the relevant electron transverse diffusion at each height is calculated with gaussian diffusion width, $\sqrt{2Dt_c}$, where D is the diffusion coefficient, $D = \frac{\mu_q kT}{q}$, and the collection time is $t_c = \int_{z_c}^{z} \frac{dz}{|\mu_q E_z(z)|}$.

Parameter	Value
Bulk doping density	10^{12}
Front side doping density	0.0
Back side doping density	0.0
Front side doping scale	0.01
Back side doping scale	0.01
Over depletion bias (volts)	-12
Silicon Thickness (microns)	100
CCD temperature (K)	173

Table 5.13: CCD parameters affecting the electron diffusion model.

Saturation & Blooming

After the electrons reach the backside of the CCD, the exact pixel is accurately assigned to each electron. Normally, electrons accumulate during the simulated exposure without any further effect. However, once the full well depth is reached (100,000 electrons), no more electrons can accumulate in that particular pixel. At that point, the electrons are

diffused to the closest pixel that is not at full well depth on that particular column. The diffusion across columns is not allowed, and the diffusion across the midpoint of the CCD is not allowed either. Once the entire column is at full well depth, the electron is destroyed. This model roughly reproduces the ideal process of saturation and blooming in real CCDs. In the future additional refinement may be done to account for a softer variation in bleed trails, as well as any variation in the saturation between devices.

Parameter	Value
Full well depth	100,000 electrons

Table 5.14: The parameters affecting the saturation and blooming model.

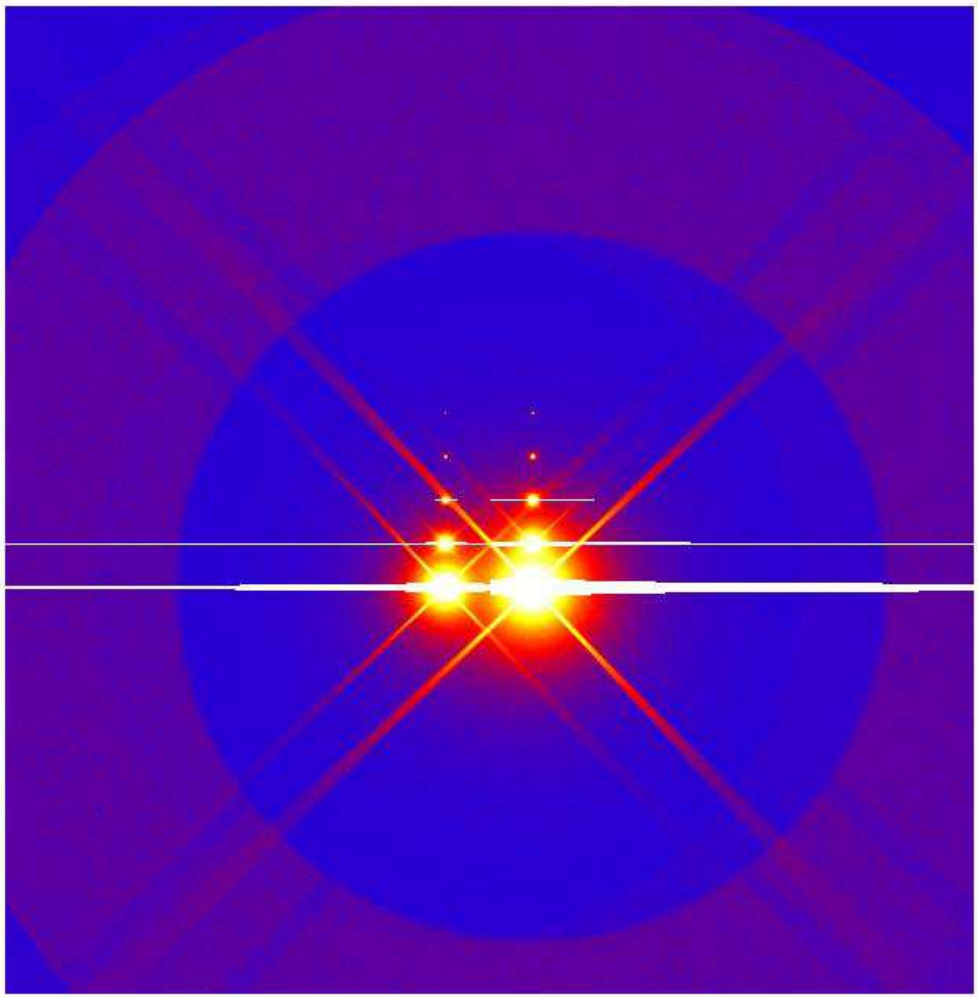


Figure 5.12: Images of bright stars showing the bleed trail (left to right and top to bottom are magnitude 20, 18, 16, 14, 12, 10, 8, 6, 4 and 2 stars.

Bright Object Optimizations

Throughout most of the previous sections, we interchangeably talked about photons as if they were identical to the simulated rays. For the vast majority of simulated sources, we simply make one photon represented by one ray. This then accurately gets the photon statistics correctly, and performs the Monte Carlo integration of all the physical effects accurately. For particularly bright source (i.e. sources that saturate the pixels with more photons than the full well-depth), we do not really require this accuracy.

To see why this is true, consider the photons in a bleed trail. They will gradually add up to the full well depth and produce rectangularly shaped trail whether photons are simulated one at a time or 10,000 at a time. Therefore, we introduced the concept of increasing the photons represented by a single ray. However, there is great subtlety on how this is implemented, because we need to not represent too many photons per ray on regions which are unlikely to become saturated (e.g. in the diffraction spikes, ghosts, and large angle scattering wing).

To perform this optimization, we calculate the fraction of rays that end up in a saturated pixel from a particular source (f_s). We then increase the number of rays per source by a function of f_s. In practice, we found that the function $p = min\left(1 + s\frac{f}{1-f}, w\right)$ works well, where $s \approx 10$ and w is the full well depth. Then, in order to accurately render the photons outside of saturated pixels, we first determine the radius of pixels where the source is becoming saturated, r_s. Then we enhance the probability of three events happening in the path of the photon by a factor, q. We repeat the calculations of large angle scattering q times, and stop the repetition of the calculation if the photon scatter more than a distance r_s. Similarly, we repeat the calculation of the Monte Carlo diffraction q times, and also stop the calculation if the photon scatters more than a distance, r_s. Also, we repeat the calculation of reflection backwards for photons that have already become stray-light to enhance the probability for ghosts. If any of these events occur, then we use $\frac{p}{q}$ photons per ray, otherwise we use p photons per ray. Thus, the simulator simulates the saturated core of the source at a faster speed than the outer regions, but still preserves the correct probabilities of each. The implementation details can be subtle, but the optimization only affects photons from extremely bright sources that have already saturated during the simulation. This avoids massive computation for a single 5th magnitude source. We also use this optimization for sources that produce many photons off the edge of the chip.

Chapter 6

Non-Astronomical Source Simulation

Following, we describe how we generate background photons, diffuse dome flat photons, and cosmic ray events.

Moon & Sky Background

Reference: Krisciunas & Shaefer (1991); Patat et al. (2006); Bankert J. (ImSim Internal Document-3); Gemini spectra

We use a background model where we model the background out of two components: the dark sky and the moon. The dark sky's is assumed isotropic on the sky and has a wavelength dependence given by the SED described below. The dark sky's brightness is assumed to be 22.09 AB magnitudes at 5000 angstroms plus a variation related to the variation in water vapor we used in the opacity model (18%). The assumption is that all of the dark sky background is due to water lines, which is mostly true. The water vapor variation is then squared, because the emission is proportional to density squared and this is converted to the actual magnitude per sq. arcseconds. The mean sky background and its variation are consistent with that found by Krisciunas & Shaefer (1991). The sky brightness is increased near twilight according to the sun's altitude using a color-dependent model of Patat et al. (2006).

The moon's intrinsic brightness as a function of its phase and altitude follows the calculation of Krisciunas & Shaefer. We use the lunar spectrum below to represent the wavelength dependence. We then need to predict the brightness where the telescope is pointing. Here we use the Krisciunas & Shaefer formula which has a term for the Rayleigh scattering of the moonlight as well as the Mie scattering of the moonlight. Krisciunas & Shaefer were only calculated the lunar brightness for one band so we simply scale the Rayleigh term by inverse wavelength to the fourth power. Mie scattering is wavelength independent.

We then can make a prediction for the sky brightness as a function wavelength for every pixel in the image. The background brightness varies across the field because of the moonlight. We then need to predict the actual counts in the image, where we need to describe the calculation in the next section. We also note that we also a trivial extension

of this model is that we remove the background when the shutter is closed to simulate darks.

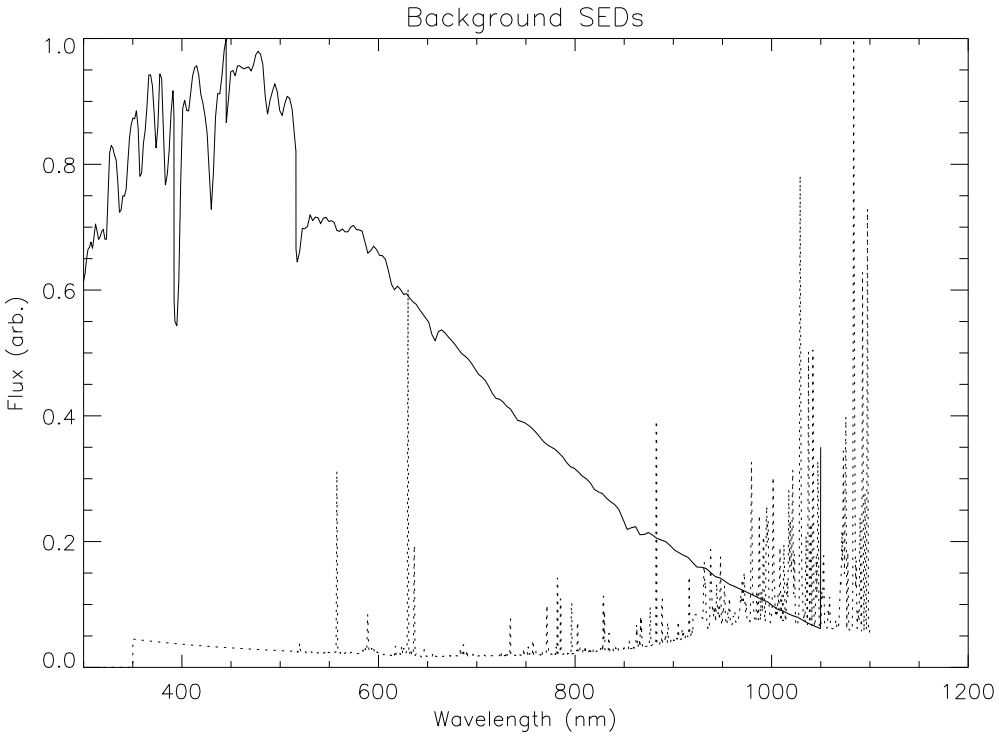


Figure 6.1: The spectral energy distribution of the background components: the dark sky and the moon. The SEDs are taken from Gemini data.

Count Rate Prediction

Reference: Bankert J. (ImSim Internal Document-3)

Simulating the actual moon and sky background, one photon at a time would increase the simulation time by a factor of at least 30. Therefore, we developed a fast technique to calculate the count rate in every pixel approximately. To predict the actual count rate in each pixel, we pre-run the simulator using scaling the sky brightness and SEDs used for the two background components. This involves first simply recording the conversion formulae for each filter given the SEDs used for each component. Then when a new sky brightness is considered the zeropoints are already known exactly, we do not have to run the simulation for every background exactly. The one major telescope effect that must be included is the vignetting as a function of field angle. To do this, we now simulate patches of a flat illuminating sources as a function of field angle and record it in a numerical table. The background program then accurately predicts the count rate in every pixel. This is then distorted by the Poisson counting error.

Dome Light

A flat that would be produced from a dome light illumination pattern actually uses the exact same calculation described for the background. Therefore, to simulate flat we use the same program, but instead use the dome light SED (currently assumed to be flat) and the dome light illuminating profile (currently assumed to have a perfectly isotropic pattern). We can use this same infrastructure to simulate monochromatic flats.

Cosmic Rays

Reference: Gilmore, K. (priv. comm.); Doty (priv. comm.)

Real images of cosmic rays are added to the simulated images using real data from thick devices. We first have constructed postage stamp images of 130 different actual cosmic ray events. We then add these randomly to our simulated images using two important calculations. To determine how often to place a cosmic ray in the image, we use the production rate of 0.04 cosmic rays per sq. centimeter of Silicon per second. This produces about 1 cosmic ray per amplifier per 15 second exposure. The actual cosmic rays are expected to be a combination of gamma rays from local ground radiation and muons and other particles from atmospheric particle interactions. Our data has some combination of the two, but it may not be in correct proportions. A second calculation gives us the scaling of electrons in the cosmic ray data, to the appropriate volume of silicon in the LSST pixels. This correctly normalizes the intensity of the cosmic ray images (i.e. makes the correct number of ionized electrons in our devices).

Chapter 7

Electron Readout Physics

Below we describe the physics the electron readout. This mostly involves taken the image from the previous step and adding camera defects or readout complications with some physics models.

Amplifier Segmentation, Readout Orientation, Pre & Over Scans

Reference: Rasmussen, A. (LSST Document-7822&7821)

The amplifier segmentation and readout direction are taken from the Rasmussen data in document-7822 & 7821. The implementation of the segmentation is straightforward, because the pixels can simply be copied from the physical piece of silicon in the raytrace to new image files representing the individual amplifier data. The readout direction is important for both in arranging the pixels in readout order in the amplifier images as well as determining the direction of charge transfer inefficiency calculations. It is also used for the hot column algorithm.

The pre-scan (virtual pixels read out earlier) and over-scan (virtualpixels read out later) regions are added to the edges of the image to represent the virtual pixels normally represented in the digitization process or pixels unexposed to photons. Currently, the pre-scan pixels in the serial direction add a buffer of 4 pixels, and the pre-scan pixels in the parallel direction adds 1 pixel. There are currently no over-scan pixels. The number of pre-scan pixels is currently insufficient for a robust offset calculation.

Relative QE Variation

Reference: Gilmore, K. (priv. comm.)

For the pixel to pixel variation of the Quantum efficiency, we used a simple model where the QE was given in each pixel as 1 minus a single-sided gaussian with mean 0 and a σ of 1%. We considered having a correlation length in the variations, but found that the

correlation length was not much more than 1 pixel in real flats. The QE variation is fixed throughout all simulations, and currently has no time-dependence.

Figure 7.1: An example of the quantum efficiency variation on the individual pixel level for various regions.

Read noise and dark current

Reference: Gilmore, K. (priv. comm.)

Read noise is implemented by adding a gaussian with mean of 2 electrons to the value in each pixel (but a negative value is not added). Read noise is added to the pre/overscan pixels as well. There is a 3% variation between amplifiers in the read noise value that is fixed throughout all simulations. Dark current is added at the rate of 2 electrons per second of exposure. The total number of electrons added per pixel is given a Poisson error. There is no amplifier dark current variation between amplifiers or between exposures.

Parameter	Value	Variation
Read Noise	$2\ e^{-1}$	3%
Dark Current	$2\ e^{-1}/s$	0%

Table 7.1: The read noise and dark current model parameters.

Hot pixels, Hot Columns, & Dead Pixels

Reference: Gilmore, K. (priv. comm.); Ahmad, Z. (ImSim Internal Document-12)

Hot pixels are added to the image in the same location throughout all exposures. Hot pixels are always set to the full well depth with no variation. The rate of hot pixels is set to 0.0125%, which is somewhat high compared to Calypso measurements (see Z. Ahmad note). Similarly, dead pixels defined as pixels with 0% quantum efficiency are included at the rate of 0.0125% and do not vary between exposures.

The total number of pixels in the full camera that are assigned to hot columns is 0.025%. This is accomplished by first randomly deciding some fraction of hot pixels that will produce a hot column and then also marking the pixels after that pixel in the readout process as hot. There is no variation in hot columns between exposures.

Parameter	Value	Variation
Hot Pixels	0.0125%	0%
Dead Pixels	0.0125%	0%
Hot Columns	0.025%	0%

Table 7.2: The hot pixel & column and dead pixel rates.

Gain, Bias, Non-Linearity, Charge Transfer Inefficiency

Reference: Gilmore, K. (priv. comm.); Ahmad, Z. (ImSim Internal Document-12)

After every pixel in the image has its complete number of e^{-1} determined from all sources, the conversion of the the image from electrons into analog-to-digital units (ADU) that would occur by a real analog-to-digital converted (ADC) is a straight-forward calculation. We use the formula below

$$ADU = \frac{eps}{G\left(1 - N\frac{e}{W}\right)} + B$$

where e is the number of electrons, G is the gain, W is the full well depth, B is the bias, N is the non-linearity factory, p is the non-linearity in the par. The gains are set to 1.7 with a 3% variation between amplifiers. Note a value of 1.7 gets the maximum well depth of 100,000 electrons below the 16-bit limit of 65,536. The amplifier biases are set to 1000 ADU with a 2% variation. The non-linearity factor is currently set to 0 with no variation. We have studied the variation of bias as a function of column to build a more realistic model, but this is not currently implemented (see Z. Ahmad note). The charge transfer inefficiency

(CTE) factors calculate the number of pixels from the readout in both the parallel and series direction, and then scale the number of counts by the number of pixels from the readout raised to the power of 0.999555 for the parallel CTE and 1.0 for the serial CTE.

Parameter	Value	Variation
Bias	1000 ADU	2%
Gain	1.7 e^{-1}/ADU	3%
Non-linearity	0.0	0%
Serial CTE	1.0	0%
Parallel CTE	0.999995	0%

Table 7.3: The electron count rate to ADU conversion model.

ADC errors

The digitization process can produce errors it the conversion from electrons to ADU. We implemented a digitization sequence simulation by taking the predicted ADU in every pixel from the previous step and calculating the value in the ith bit by

$$\mathrm{b}it_i = \left(\frac{ADU}{2^i} + \mathrm{A}DCERR_i\right) \mod 2$$

and then the final ADU value is given by

$$ADU = \sum_i bit_i 2^i$$

where the $ADCERR_i$ values keep the digitization from being perfect. Currently, however we have set all those errors to 0.

Part III

Future Improvement Plan

Chapter 8

Physics Model Improvements

Below is a list of planned future updates to the photon simulator. In some cases, extensive research has already been done, and in other cases we have done limited preliminary work. We roughly list them in the same order as the physics description document.

Astrophysical Model Improvements

- **Galaxy Morphology:** The current galaxy model uses ten parameter to specify a separate ellipsoidal sersic distribution for both a galaxy's bulge and its disk. The next step would be to add a slightly more complex morphology by representing spiral arms and non-ellipsoidal shapes.

- **More flexible SED interface:** To represent a greater diversity of objects and types of stars and galaxies, we probably need to move to a system where SEDs are represented by additive components in order to keep the total number of SEDs to a reasonable number (10,000). This requires re-working the interface, and modifying the method of choosing wavelengths of photons.

- **Other features:** Some features such as the weak lensing shearing and simulating galaxies from truth images are tested in a limited sense, but have not been deployed on the full grid computing system. We expect the implementation of these will be straight-forward, but could require some changes in the interfaces.

- **Continued OpSim interface:** We will continue to use the OpSim parameters as input along with the astrophysical parameters, and will continue to evolve the parameters so that interface is as self-consistent as possible.

Atmospheric Model Improvements

- **Improved Turbulence Screen Model:** The measured distributions for the outer scale (and also the outer-outer scale) are very limited. This model clearly has a significant effect on the atmospheric PSF, so it is important to study not only the

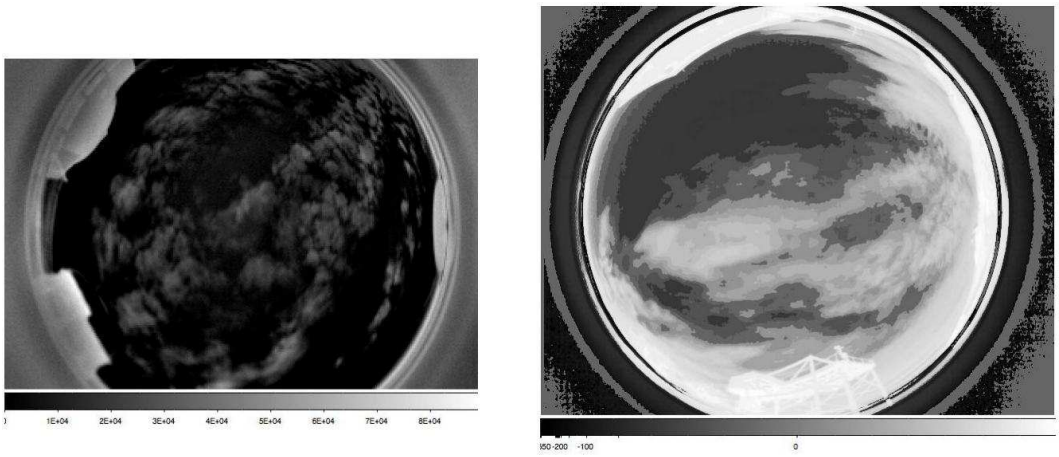


Figure 8.1: Sample whole-sky cloud images provided by Sabag (see J. Sebag et al., SPIE, Vol. 70-12, 2008).

mean values for these parameters, but their distributions as well. The numbers of layers has proved to be rather important in the distribution of PSF properties, so it must be studied in detail about how many independent layers (or a vertical correlation scale) should be used. With the current model, we can show that the atmospheric PSF are reasonable but may not be the ultimate model.

- **LSST Site parameter data:** An extensive set of planned site measurements in the next few years can be incorporated into the simulator. This includes study of the wind velocity vector vs. height, the outer scale vs. height, and the turbulence intensity vs height. We have working reasonable models for all three, but it would be useful to study the correlations in time, the correlation scale height of each three, and the correlations of each distribution with each other. These distributions affects the atmospheric PSF shape, size, correlation with angle, and correlation between one exposure and the next.

- **Improved Cloud model:** The cloud model can be update based on LSST site data using images of actual clouds. The current model uses a simple structure function derived using SDSS data. The number of effective layers, correlation scales, and variation from one exposure to the next could be included in a more realistic model. Also direct measurement of clouds over time will also provide information about the wind profile. Figure 8.1 shows an example of an whole sky IR image of clouds.

- **Improved Opacity model:** The opacity model consist of atomic absorption of molecular oxygen, ozone, water vapor, as well as Rayleigh scattering. The variation of some of these component is currently just a guess, and other trace components could be added, such as aerosols. The correlations between their relative abundance and other atmospheric parameters is currently assumed independent, which may not be the case. These could be updated with further study.

- **Improve Water Vapor Model:** Water vapor is included in the current model both in terms of emission in the sky background and the absorption of light from celestial sources. It is clearly the most important component that has signficant variability (10s of percents). Currently, we have a crude log normal variance of the water vapor valiability based on measurement at various sites, and couple the emission in the background to that variability. We really need better information about the water vapor variability at the LSST site. We also need to vary the emission across the field in the background model. This topic is particularly important for the Z and Y band simulations.

Telescope/Camera Geometry & Perturbation Improvements

- **LSST geometric design updates**: We will continue to track the basic design of LSST (optics design, detector layout, and coating specifications. This will evolve in the project, and we will have to track these changes.

- **Baffle models**: Currently we do not explicitly model the light baffles, because we do not simulate light outside of the field of view. To accurately simulate the straylight from the sides of the telescope, implementing the baffle model would be necessary. We could include a real model of the baffles which would also necessitate the addition of a model for all significant stray light sources including interactions with the physical structure of the support structure of the telescope and camera. These complex additions would be especially significant near dust and dawn and near full Moon.

- **Dynamic camera geometry model to match dynamic telescope model:** One change in choosing parameters for the optical perturbations is that some of the parameters describing the camera geometry are static (the same throughout the entire run) and most involving the optics are dynamic (change every exposure). We need to build the capability to model the time dependent changes for the camera parameters in the optics_parameter code and therefore avoid the need for large data files.

- **Time-scales associated with each physics perturbation model:** Once the two perturbation models are on the same level of fidelity, we then can set the perturbations to evolve on different time scales. So two adjacent observations will have similar perturbations, but two observations separated by a year will have a different realization entirely. This is important, because the mirror control system is expected to evolve on ten minute timescales.

- **Perturbation/Misalignment parameters predicted from physics models and build tolerances:** The perturbations then can be set from models of physics (vibrational, thermal, gravity, pressure) with their own external parameters (e.g. the gravity vectors) or build tolerances (fabrication errors). These models will be developed with engineers in the camera and telescopes teams. Therefore, we will end up replacing the single line gaussian random number called that sets the perturbations

currently, with a several line code model that sets the parameters from a physics models with observing parameters. Note that many of the possible degrees of freedom are currently set to 0, which is only a crude approximation (i.e. the lens surfaces).

- **Perturbation/Misalignment parameters coupled through simulation of feedback loop:** To truly address the correlations between the parameters, we have to simulate the feedback loop of generating wavefront images, and then simulating how well the correction occurs through the active optics system. The model would then use the detailed atmospheric conditions at the time of the observation. It may be necessary to simulate in a crude sense the sequence of perturbations that may occur over the half hour prior to an exposure to accurately predict what the perturbations will be during a particular exposure pair. This sort of improvement will likely occur through study with the active optics group, and then we will be able to couple the correlated parameters through some kind of simplified model.

- **New tracking model:** The current tracking model has a white spectrum and is simulated independently for each exposure pair. We have sample power spectra for the fine pointing system from the engineering team that is not white that could be implemented. The engineering tolerance of the tracking model needs further clarification from the engineering team.

- **Filter changer:** The cadence is effected by the filter changes. We may have repeatable slight errors in filter locations after a filter change. Its unclear how important this effect is.

- **Large angle scattering physical model:** Our current large angle scattering model is an empirical distribution in angle determined from Gemini data. We could replace this empirical model by a real physical model with a spectrum of perturbations and then predict the angular distribution directly from scalar diffraction theory.

Telescope/Camera Interactions Improvements

- **Verification of multi-layer approximation:** We use a multi-layer approximation to simulate the effect of the angle dependent transmission in the filter coatings, lens coating, and the detector coatings. In this model, we have an approximate formula for the angular dependence as well as an effective index of refraction in the multi-layers. Alternatively, the exact angle and wavelength transmission function can be calculated if the exact multi-layer prescription (composition and height pattern) is known, and the reflective and transmission probabilities can be calculated directly. We can then either verify our approximations, or replace this with a more detailed model. Figure 8.3 shows recently provided models of filter designs from a number of vendors. After the down selection for the baseline filter is finished we will update the models.

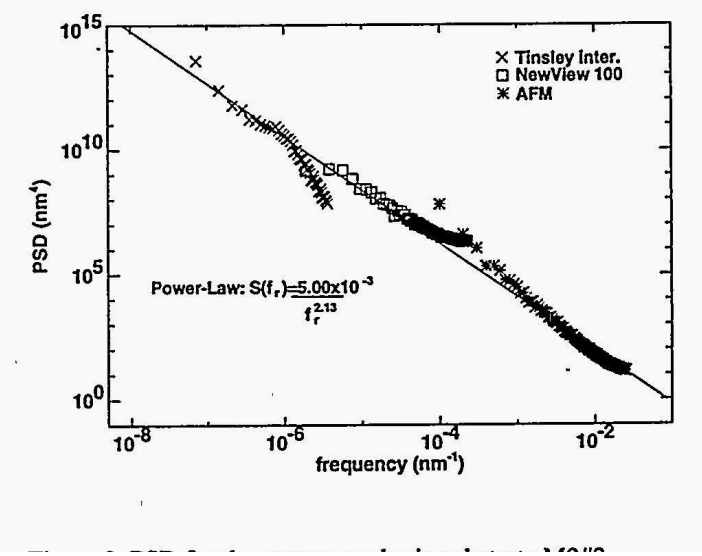


Figure 8.2: From: "Surface characterization of optics for EUV lithography", P. Gaines, D. W. Sweeney, K. W. DeLong, S. P. Vernon, S. L. Baker, D. A. Tichenor, and R. Kestner, LLNL, 1996

- **Coating non-uniformity:** The coatings currently have no positional dependence, but we could include such an effect where the coatings on the mirrors, filters, lenses, or detectors are not perfectly uniform.

- **Update to filter, lens, detector coating models:** The filter, lens, and detector coating models need to be updated based on the actual design and not on requirements. We have, for instance, no coating on the back of the filter, and the actual design may change. This ultimately affects the accuracy of the simulation of photometry as well as the ghost patterns (see figure 8.4. The model for the back surface reflectivity of the CCD currently has a simple 20% wavelength-independent reflectivity that could be improved.

- **Lens surface fabrication:** We could model the surface errors in the fabrication of the lens. Figure 8.2 shows an example of the type of power law model that would be useful for modeling these surface effects.

- **Glass non-uniformities:** The glass has a single index of refraction in all of the filters and lenses. We can vary this as a function of position according to engineering studies. The pressure and temperature changes across L3, in particular, are predicted

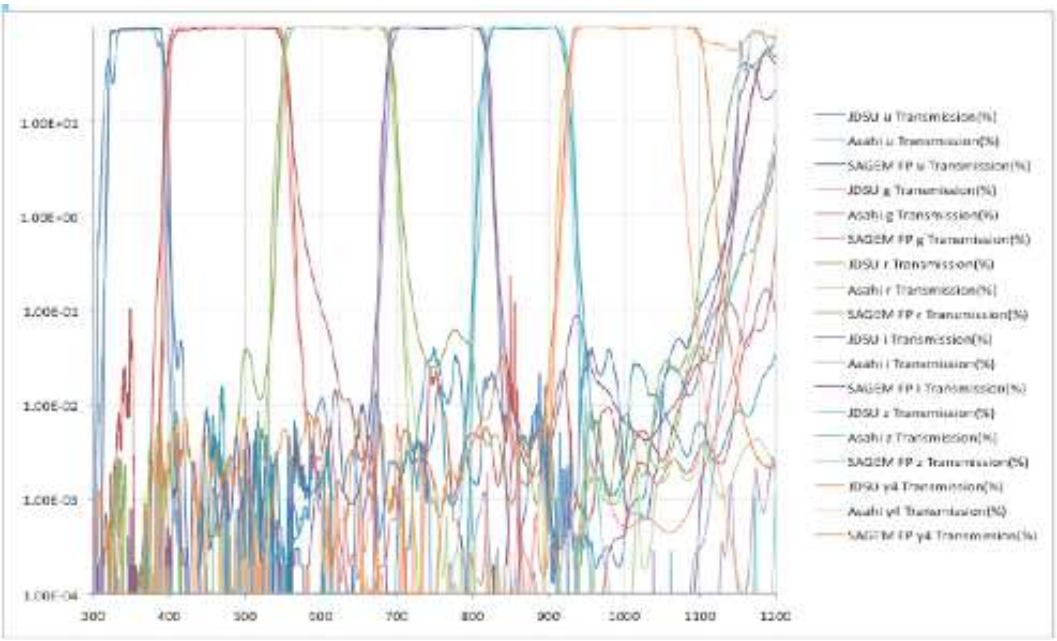


Figure 8.3: New Filter designs from Vendors (figure provided by Gilmore)

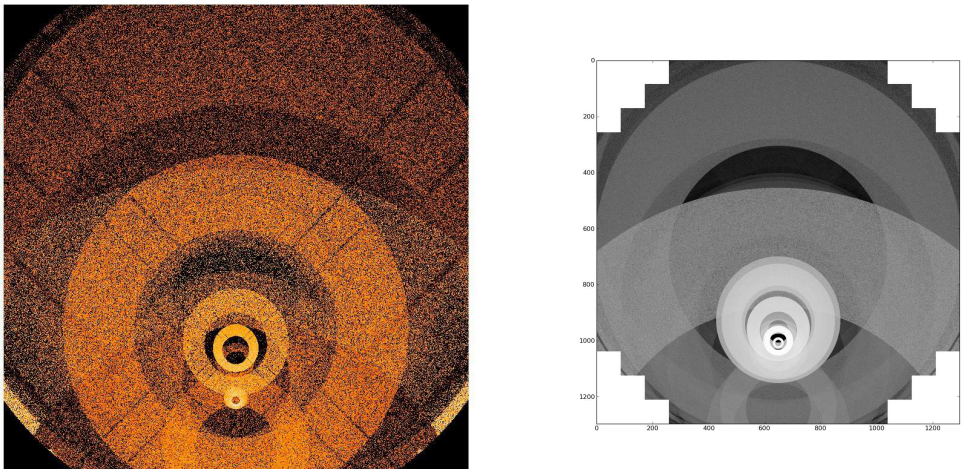


Figure 8.4: Left panel is a calculation of the Ghost pattern of a bright star using the photon simulator; the right panel is the ghost pattern derived from an independent investigation by the LSST engineering team. These images are similar but with differences that require further evaluation.

to have a noticeable change in the index of refraction throughout the L3 glass. In addition, there could be surface pitting or macroscopic defects internal to the lenses, this could be modeled by volume variation in the index of refraction.

- **Electron diffusion physics:** The model of electron diffusion can be updated to include both the fact the the electron's velocity saturates in Silicon as well as the

curved field lines exhibited in thick devices. The tree ring effects identified by the engineering team are likely in the 3-D, which modifies the pixel boundaries.

- **Model for CCD traps** This effect is thermally sensitive and is not currently part of the CCD model. This has the potential to have a significant impact on PSF prediction and could imply a loss of some data for weak lensing.

- **More complex dome seeing models:** Our dome seeing model is a simple gaussian model with the width equal to the design expectation. This can be improved by something similar to our atmospheric model, and requires further study.

- **Saturation and bleeding:** The CCDs saturate at exactly 100,000 electrons per pixel, and the excess charge is shifted to exactly the closest pixel. In reality, this is an ideal approximation and the charge can shift further, and become saturated with some variance. A study of bleed trails with prototype devices can be used to come up with a better model.

- **Fringing:** Fringing from strong emission lines in the sky background has been studied by Rasmussen, but not implemented in the simulator. We could add this capability.

- **Study of ghosts:** Ghost have been simulated, but not studied in great detail because they haven't been turned on in recent data challenges. We expect to start simulating them in the near future. Particular attention to the coating design is important, because it produces a very different ghost pattern. There are computational challenges for this, but we have done prototype testing (see figure 8.4.

- **Glints:** Glints off certain edges of surfaces should be considered.

- **Photo-response non-uniformity (PRNU):** This effect is not included and is especially important for u band.

- **Polarization:** Photons enter the Camera from a wide range of angle (+- 22 degrees) so that the outcome could be dependent on the polarization which could effect the PSF.

- **Shutter model:** The current model assumes that all pixels of all CCDs have the same start and stop times for 15 second exposure. We need to model the shutter time dependence with the correct start and stop times for each pixel of each CCD. The full details are known including mechanical repeatability, but the model is not yet implemented in the code.

- **Real flatness data for CCDs** We current have a model that matches requirement but may not be physically reasonable. There is some possibility that there is significant high frequency spatial structure. The flatness interface supports arbitrary shapes.

- **Dust on surfaces:** We have included only the coating reflectivity functions on all surfaces, but we could include models to degrade their reflectivity over time (or

improve it through re-coating or cleaning), or produce additional empirical realism in their efficiency.

Non-astronomical source simulation improvements

- **More components in background model:** The background model consists of two components: the dark sky and the light reflected off the moon. The model does not, however, self-consistently include the clouds used in the current atmosphere model nor the variation of the emission components across the sky (although we do couple the water vapor opacity to the normalization of the dark sky and use the average cloud opacity). Stray light from large angles is not included currently either. Future improvements could accurately model the spatial variation, its wavelength dependence, and time dependence of the actual sky. Comparison to real data, and models in the literature could provide more detail.

- **Dome light and dome screen model:** We have approximated the dome light as a perfectly flat SED source and the dome screen as a perfectly uniform reflector. We could update both of these models to produce more realistic dome flats.

- **Cosmic ray site data:** A simple improvement to the cosmic ray model would be to use template images based on actual rays taken at the actual site. The actual observed rate in rays per sq. centimeter can be used. We also can use GEANT simulation to build a complete model including secondaries from the entire telescope structure. It would also be an improvement to increase the library of cosmic rays images.

Readout physics improvements

- **Device design:** There are two amplifier layout designs that also may change over time that we will need to update and track. Further details, such as edge pixels, can be added as designs become more specific.

- **Edge Pixels:** An obvious change to the device design is to correctly describe the pixels at the very edge from either prototype devices or a model. We need models for both pixels at the edge of the overall CCD and for some CCD designs for pixel at the edge of the 16 amplifier regions.

- **Update to ADC modeling:** We use a simple model based on a single equation to predict the conversion from electrons to ADU counts. To enhance this model, we can include a position dependent offset, a more complex pre-scan/over-scan model based on actual electronic response, read noise/dark current predictions from actual devices, dark current pixel to pixel variation, gain non-linearity in the high and low signal range, ADC non-linearity, ADC digitization errors, and detector cross-talk (both neighboring amplifiers and external electronics cross-talk).

- **Detector defect model:** The current detector defects contains QE variation, hot pixels, hot columns, and dead pixels. We have simple models for each of these. We could improve these models from studying the rate of defects from actual prototype devices, as well as adding new models, such as edge pixels, and making these model time-dependent. The hot pixel rates are temperature dependent, and that could be modelled with prototype-devices. We could also include a detector failure rate, and an amplifier failure rate.

Further validation

- **More detailed alternative calculations:** There are a variety of more detailed validation tests that can be done to compare with alternative calculations that can be done within the project. This often involves simulating some specific configuration that can be calculated from an alternative code. The ghost pattern from a bright star, for instance, would be an ideal more complex test.

- **Validation and Studies with other telescopes:** We have the prototype capability to simulate the Subaru telescope and the calibration telescope and hopefully others soon. The calibration telescope can help us to understand the common atmospheric model by simulating simultaneous LSST and calibration telescope observations. These simulations will help to validate the simulator in much greater detail, and are also scientifically useful. A number of other possibilities would help the project and help us with validation. For example, we could simulate the Camera Calibration Optical Bench (CCOB), the photon counting calibration concept, the BNL CCD test system, or other optical systems.

Chapter 9

Efficiency and Automation Improvements

There are several categories of improvements that involve more efficiently running the photon simulator codes. Automation improvements are improvements that minimize the human effort of running the code. Efficiency improvements minimize the CPU time that is used while running the code. There is a general strategy to continue to simplify and improve the automation of the code, and make improvements that are straight-forward that can enhance efficiency in some simple way. Needless to say, the fact that grid computing is used as the primary platform, means these improvements can be somewhat complicated. In general, although one might be argue whether efficiency/automation should take precedence over fidelity improvements, it is generally true that as efficiency/automation improvements are done, we have more development time for fidelity improvements.

Efficiency Improvements

- **General numerical efficiency improvements in raytrace:** Part of the simulation time is used by tracing individual photons through the entire framework. Optimizations can simplify the calculations and make them more efficient. The efficiency is sufficient, but there are always improvements to be made.

- **Bright object optimization improvements in raytrace:** Bright stars would dominate the simulation time, if there were not already optimizations in place (dynamic transmission and increasing the photon per ray when stars saturate). In general improving these optimizations involves carefully maintaining the fidelity on effects that matter for bright star haloes (diffraction, large angle scattering, and ghosts).

- **General reduction in overall file I/O:** Grid computing often is dominated by I/O transfer speeds rather than actual computing speeds. We have often seen increases in production rates just by simplifying the amount of data each processor receives. The instance catalogs, atmospheric screens, SED files, and other telescope and camera data need to be kept small in size.

- **General reduction of memory footprint:** Most grid computing is done with only 1 or 2 Gigabytes per core in RAM. Therefore, the memory of the raytrace and other codes need to be kept small.
- Grid computing multi-platform improvements: There are a variety of grid computing specific optimizations that need to be done to keep the production process efficient.
- **Alternative methods:** On a longer term basis we could consider simulation methods other than grid computing. This could include using bootstrap methods usings either grid computing or GPUs.

Automation

- **Connect instance catalogs to image production via an automated mechanism:** We currently do not have an automated mechanism to transfer instance catalogs to the image production sites.
- **Improve the QA of image production and error catching:** A large fraction of our data challenge production runs is spent understanding strange file transfer errors intrinsic to grid computing that occurs about 1 out of every 100,000 images. A simpler mechanism to check for this, and rerun simulations automatically without human effort would increase our efficiency.
- **Improve the automation of file transfer:** The file transfer has been automated to some extent, but we need to improve this to be more flexible and involve less human interaction.
- **Continue reorganization of LSST-specific data separated from general code:** The code has generally started separating the LSST-specific data from the actual physics code. This process should continue because it allows us to update the telescope and camera models more efficiently as we have defined clear interfaces.
- **Maintenance of the validation scripts:** We will continue to maintain the validation code for the ten task in the previous section. This involves making sure the scripts remain consistent with the current code interface. This is a non-trivial effort, but should enable more aggressive software development, because the validation can be rerun to check for mistakes than invalidate the validation.
- **Operation on opportunistic resources quasi-continuously:** If much of the automation tasks are obtained we can operate the local resources throughout the year automatically. This would allow us to produce more images with more time to improve the code.

Part IV

Validation Tests of the Simulator

Appendix A

Validation Approach & Framework

A large fraction of the physics discussed in the previous chapters has been validated by extensive informal testing. This includes testing by the code developer to make sure the designed outputs are reasonable, and testing by the entire ImSim team when the photon simulator is run for a large number of different purposes. For the validation process described below, however, we selected a set of 10 well-designed tests that would test a sub-set of the simulation outputs. In general, the tasks are designed to easily fail if a large fraction of the code is not operating perfectly. The tests also attempt to validate the more important characteristics of images that would most affect the scientific measurements.

The tests are all administered by constructing small instance catalogs of objects (in most cases one or more stars in a certain pattern) with a variety of special commands defined in a command file to isolate the physics of a certain type that is being validated. The instance catalogs and command files are located in the SVN repository, so the tests can be performed by every user on every version of the code. We plan to maintain and refine these tasks, so any version can be validated with at least these 10 tasks, before it is deployed on large scale computing and used for large-scale image production. Undergoing this process has made the code development actually more efficient because it is easier to notice problems earlier in development. The 10 tasks also actually run all 10 photon simulation codes even if that is not necessary, so the complete suite of code is always tested.

We had to compromise between designing tests that were too simple (but fast) and would not really test the desired simulation performance, and tests that were too elaborate (but too slow) that would be more like a research study of a given topic. Consequently, the goal was to have entire validation pipeline run in less then 24 hours on a single laptop or desktop. Note, in many cases we decreased the pixel size and increase the brightnesses of the source, so measurements made off the image were not affected by shot noise error or finite pixel effects.

Following, we describe each task and the requirement for validation. Appendix B studies the validation of the atmosphere (which was described in Chapter 4), Appendix C studies the validation of the telescope & camera physics (which was described in Chapter 5), and Appendix D studies the validation of the electron physics and diffuse simulation (which was described in Chapter 6 and 7).

The validation summary of the ten tasks is shown below. The tolerance for each task is

shown, so the developer can quickly determine if something was done to change the code so it no longer agreed with the validation specifications. In the following appendices we describe these results.

Validation Summary for Revision: 24812

Validation Type	Validation Task	Accuracy	Pass/Fail	(Tolerance)
γ Propagation in Atmosphere	1A (Refractive Approximation):		Pass	(< 1.00)
	1B (Screen Convergence):	0.31%	Pass	(< 0.50)
	1C (Atmospheric PSF):	0.29 std dev	Pass	(< 2.00)
	1D (Atmospheric Astrometry):	27.91% p−value	Pass	(> 10.00)
γ Propagation in Optics	2A (Spot Diagram):	98.55% overlap	Pass	(> 95.00)
	2B (PSF Budget):	0.33 arcsec	Pass	(< 0.38)
	Optics Design	0.07 arcsec	Pass	(< 0.10)
	Dome Seeing	0.08 arcsec	Pass	(< 0.09)
	Perturbations/Misalignments	0.24 arcsec	Pass	(< 0.26)
	Charge Diffusion	0.21 arcsec	Pass	(< 0.25)
	2C (Throughput):	49.93 millimags	Pass	(< 50.00)
	u flat AB_{500nm} zeropoint	27.22 mags	Pass	(27.23 +/− 0.20)
	g flat AB_{500nm} zeropoint	28.72 mags	Pass	(28.72 +/− 0.20)
	r flat AB_{500nm} zeropoint	28.86 mags	Pass	(28.86 +/− 0.20)
	i flat AB_{500nm} zeropoint	28.89 mags	Pass	(28.89 +/− 0.20)
	z flat AB_{500nm} zeropoint	28.62 mags	Pass	(28.63 +/− 0.20)
	y flat AB_{500nm} zeropoint	27.96 mags	Pass	(28.08 +/− 0.20)
	2D (Spider Diffraction):	0.30%	Pass	(< 0.50)
e^- Propagation	3A (Diffuse/Camera Defects):	0.17%	Pass	(< 0.50)
	3B (Charge Diffusion):	12.92 microns	Pass	(12.00 +/− 2.00)

Figure A.1: Output of the suite of validation tasks.

Appendix B

Validation of Photon Propagation in the Atmosphere

Task 1A: Test of the Refractive Approximation

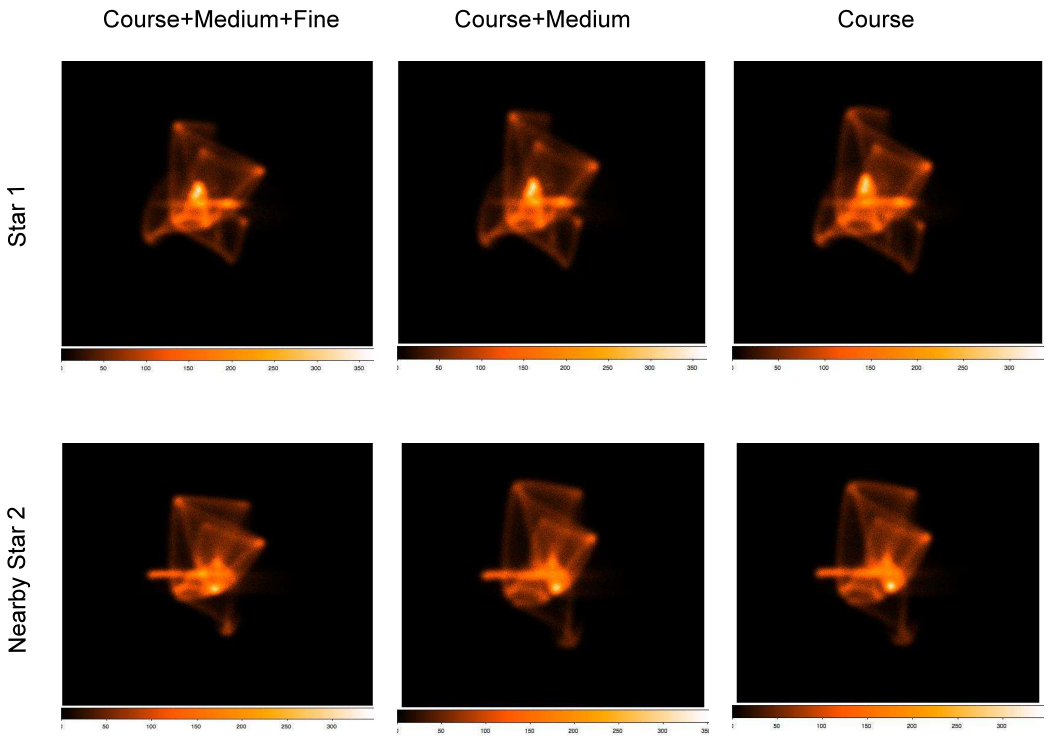


Figure B.1: Sample PSF with speckles. The top row shows the separate effects of the coarse, medium anf fine screens. The bottom shows the similar PSF from a nearby star. The two PSFs share part of the path through the air as defined by the partly overlapping projected aperture of LSST.

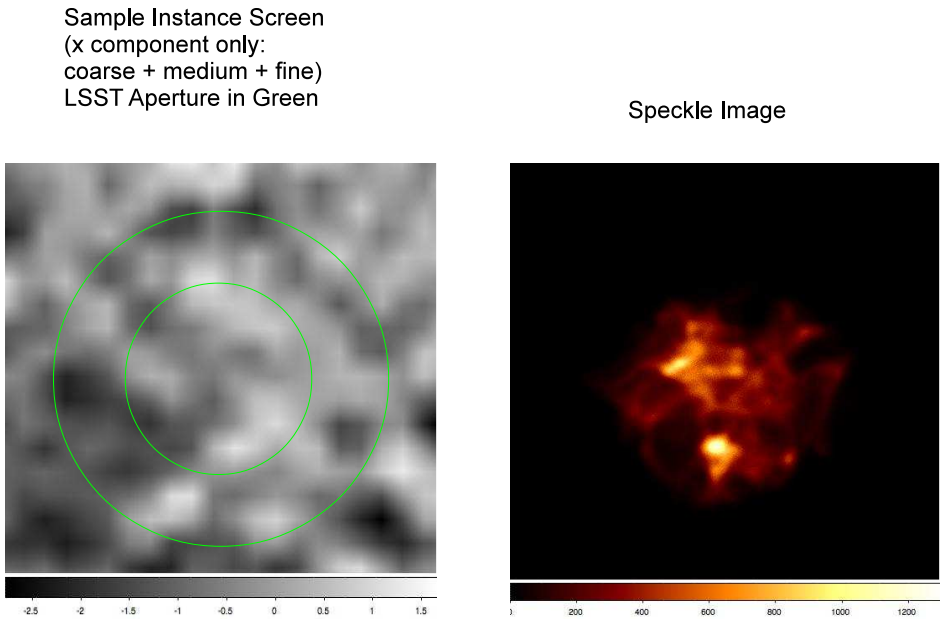


Figure B.2: A instance screen at the scale of the LSST annular aperature with the associated speckled image.

Requirement: Show that important diffraction effects are includes in the photon path algorithm (screen functions for each layer and the raytrace code). The validation requires a direct compare son of simulated PSF with the geometric raytrace and full diffraction.

The purpose of this task is to show that each simulated PSF for a point source (star) is in a form of speckles that are expected from a full calculation based on diffraction. The light that passes throught the layers of the atmosphere that form an unwanted time dependent optic formed by the density fluctuations enduced by the turbulent flow of the air. These density variations are modeled by a Kolmogorov spectrum which imprints a variation in the index of refraction. The path of the photon changes direction through each layer due to the refraction that is basically the same physical model that we use to follow the path through the lens of the camera. We model each layer by the sum of three screens each 1024x1024 elements in 2D. The resolution (and size) of the coarse, medium and fine screens is 1 m (1 km), 10 cm (100 m) and 1 cm (10 m) respectively. The projected aperture of the LSST captures the light from a star at a given instance that passes through each screen that is translating according to the vector wind of that layer. The dominant component of the refraction of the light is set by the coarse screen with a 1 m resolution that maps the density of the air over the ~8 m aperture of the LSST. Figure B.1 shows an example of the image of a star (top left) with the full screen, the same star (top middle) with the fine screen removed, and finally the same star (top right) with only the coarse

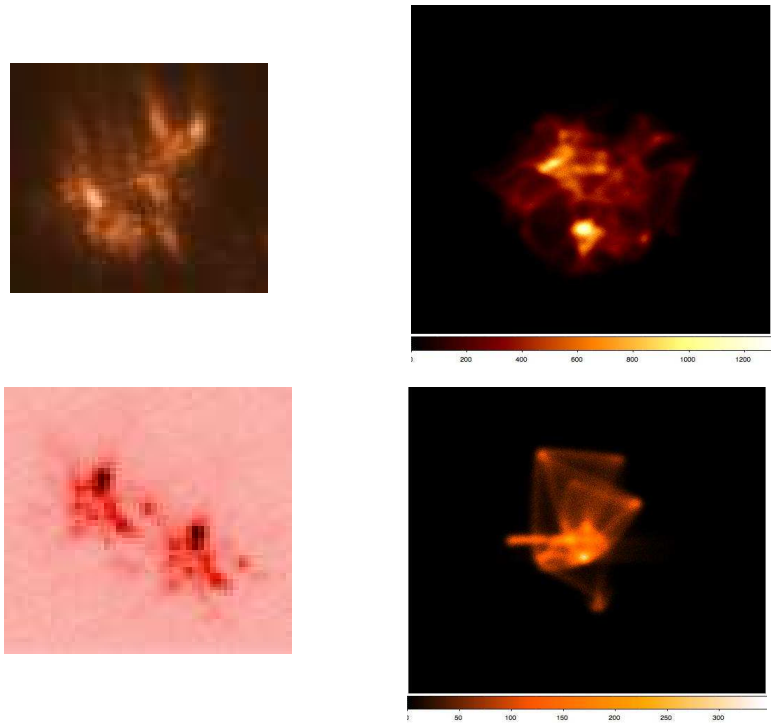


Figure B.3: Comparison of real and simulated images.

screen. As expected the number and placement of the speckles is mostly set by the coarse screen with some shifts in the speckles as the effects of the medium and fine screen are included. The lower three images are computed for the same screens for a second nearby star that shares a portion of the screen with the first star. However the geometric path through the screens is shifted enough to change the speckle pattern for the second star. Clearly the speckle pattern is highly correlated as expected from two stars separated by a small angle. The simulation ensures that the full nonlinear PSF is computed for each instance in time during a 15 s exposure for each separate point within the full LSST field of view. Overall the correlations in the spatial and temporal structure of the PSF are accurately represented. The wavelength dependence of the speckle pattern is reproduced by a special Fourier filter that is applied to the screen to set the scale of the pattern of the speckles. This spatial filter is a function of the Fried parameter approximated for the average wavelength of the particular LSST filter passband. This model produces a realistic speckle pattern but does not capture the detailed diffraction effects of the Airy pattern that each speckle would show at high sub-pixel resolution. This limitation is not an issue for an LSST simulation since the optics of the telescope and the camera are not diffraction limited. This approach works well for a wide field large aperture telescope (degrees across the field) but would not be appropriate for an AO system that seeks diffraction limited performance over a narrow field of view (less than ~1 arc minute).

Figure B.2 an example of an instance screen (sum of the coarse, medium and fine

parts) and the corresponding speckle pattern. We show the full screen at the scale of the LSST aperture (left panel) and compute the associated PSF (right panel). The left part of Figure B.3 shows two examples of actual speckle patterns from existing telescopes. The right part of the figure shows two similar examples of speckle patterns for two examples of simulated stars.

Task 1B: Test of Screen Convergence

Requirement: Perform a convergence test for screens that shows that physics is independent of grid resolution

The atmospheric model captures the physics of the turbulence by implementing a series of numerical screens. The numerical screens are rather complex in that there are multiple layers with multiple grid resolutions. The use of the screens by the raytrace involves determining the kick vector depending on where each photon hit at a given time. The final positions of the photons, and therefore the size, shape, and centroid of the PSF is affected by these photon kicks. Therefore, it is important to test even if the atmospheric model is incorrect, if the positions of the photon are independent of the grid resolution.

Below we simulated a single star with an identical set of parameters for 9 separate simulations. For each simulation, we measured the ellipticity, centroid, and size (5 parameters). The only variation was that the turbulence screens were used in nine different ways. First, the medium & fine screens were ignored in the calculation completely, and only every fourth pixel was used in the coarse screen. In this way, we were crudely representing the atmospheric screens by a very coarse screen that was undersampled. The second simulation was identical except we used every second pixel in the coarse screen. The third simulation simply used all pixels on the coarse screen. The fourth simulation continued using all pixels on the coarse screen, but added every fourth pixel on the medium size screen. This continued until we used all pixels on all three screens for the ninth simulation. Below we show the effective resolution (by calculated the effective pixel size) and the five measurements of the star compared to the value when we used all pixels. The data indicated that the size, shape, and position of the star have approximately converged after adding the last fine screen. We compare the last to second last point of each of the five measurements to calculate the screen convergence.

Task 1C: Test of Atmospheric PSF Effects

Requirement: Validate the multi-layer model of the atmosphere with wind speeds and structure function of the turbulence and demonstration that the PSF including form and correlations across the field of view are present at the correct levels.

The atmosphere model creates a different PSF for every point in the field (as well as a different one depending on the SED of the source). It does this because the time averaged set of photons will result in a different set of displacements as they pass through the series

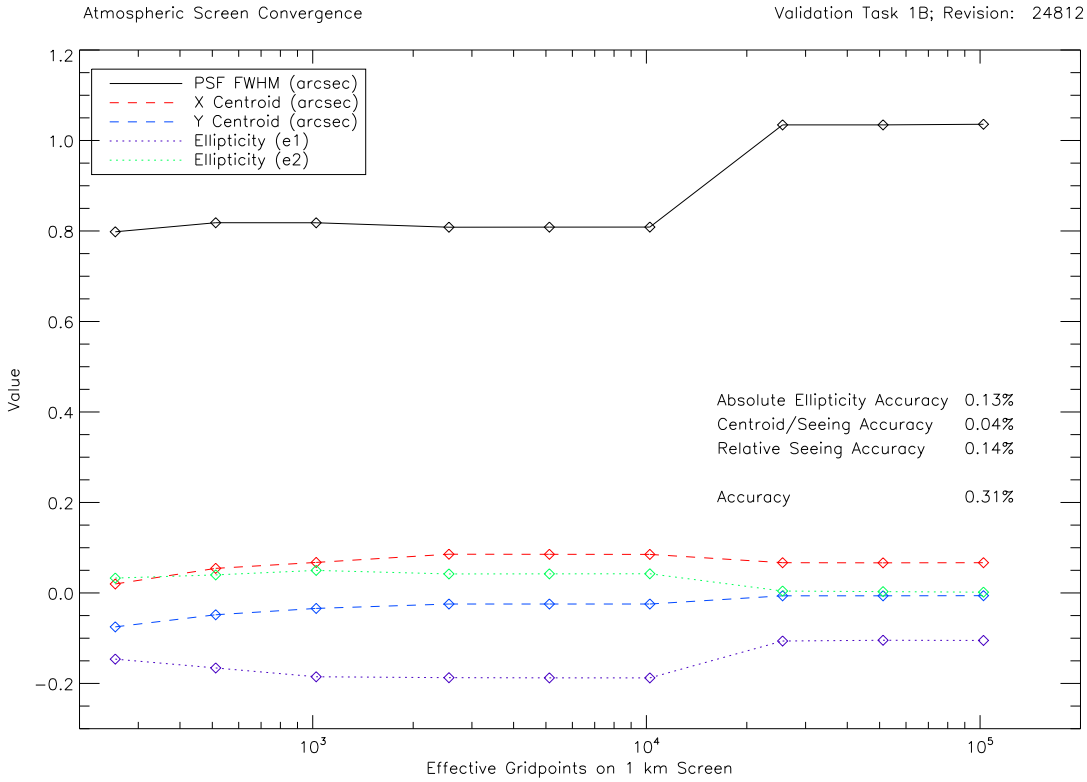


Figure B.4: Validation of the convergence of the turbulence screens for task 1B.

of turbulence screens drifted at different velocities. If the screens has a very fine scale pattern (say a coherence scale of 1 cm), then the resulting PSFs would be perfectly round because the angular displacements would average out with more and more photons. However, with the real atmosphere the turbulence pattern has a much larger scale to the pattern (several meters), so over reasonably short integration times, the PSF has a finite and measureable ellipticity. Below, we first test whether the atmospheric seeing is what is intended, and then we compare the ellipticity distribution to Subaru data. The spatial correlation scale is then compared.

To validate the atmospheric PSF, we simulated grids of stars across the full focalplane (one on each raft, and one on each chip for the central 5 by 5 chips). We decreased the pixel size to 1 microns, turned off all optics perturbations, and increased the brightness to 18th magnitude so their were no statistical or pixel effects on the results. We repeated this simulation for 5 different realizations of the atmosphere with 1 arcsecond seeing. We then measured the PSF size and ellipticity for all the stars. The figures below show those measurements. The upper left plot shows that the measured sizes approximately agree with the input to the simulator (1”). There is some variation at the few percent level across the field. The upper right plot shows the ellipticity distribution of stars. The ellipticity is produced by the fact that the turbulence screens produce an incomplete averaging of the isotropy of the PSF. We compared this to measurements of the ellipticity

of PSFs using 15 Subaru exposures. Since Subaru has a similar aperture size to LSST, and the exposures were the same as the LSST exposure time, this is probably the most appropriate data to make a comparison. The bottom plots show the 2 point correlation of both the PSF size and the ellipticity. The fact that the size is highly correlated is not surprising because of the frozen-screen approximation. The ellipticity correlation has a angle around 1 degree. This is roughly expected because the ratio of the typical outer scale to the height of the atmospheric turbulence sets this angular scale, since stars larger than this angle will experience a completely different turbulence pattern. We have compared both of these correlations to the same Subaru data. It has a similar fall off in the ellipticity correlation and a flat profile in the size correlation. In our model, the exact correlation varies from one atmosphere to the next, so with only the current data we can it roughly agrees given the uncertainties. We do not yet have data that validates the correlation on large scales, because of the inability to have a telescope with a large enough field of view, similar aperture size, and have the ability to take 15 second exposures. We calculate three quantities from these measurements: 1) the number of standard deviations the average PSF size is from 1.0, 2) the number of standard deviations the mean ellipticity is from the Subaru mean ellipticities given their width, and 3) whether the log of the ellipticity correlation angle is within 1 dex of 1 degree. The sum of these three statistics gives our validation value.

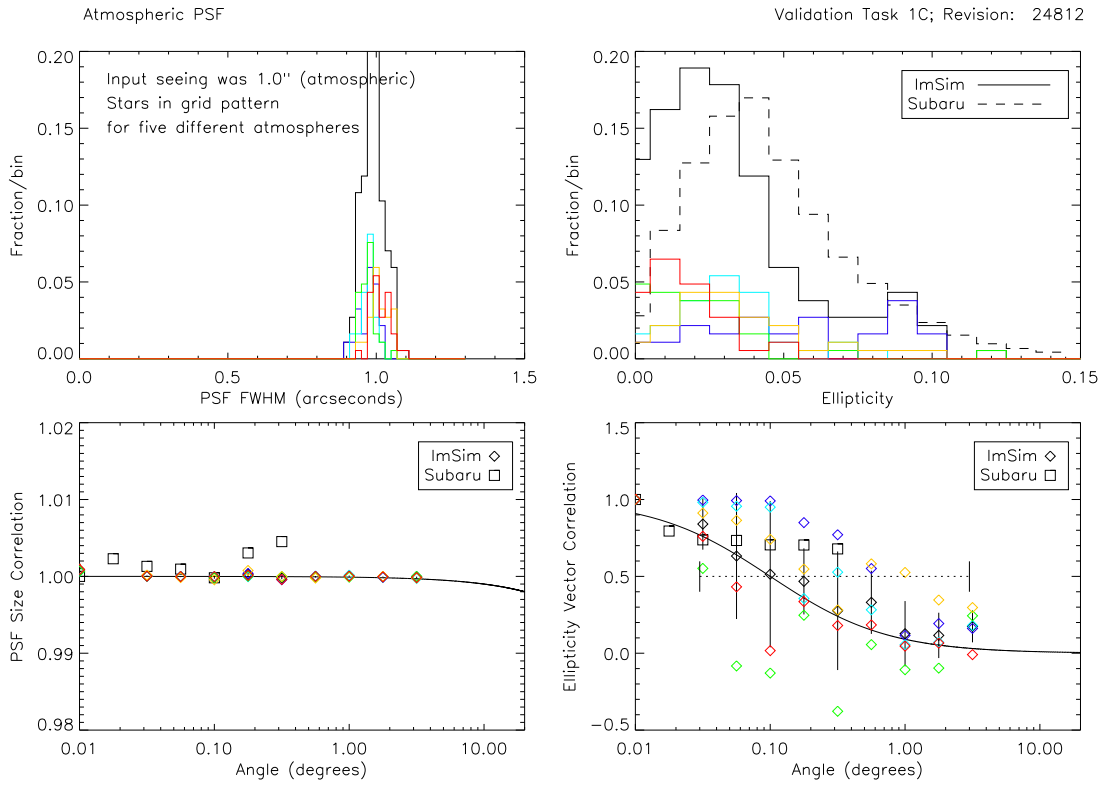


Figure B.5: Validation of the Atmospheric PSF for Task 1C.

Task 1D: Test of Atmospheric Astrometric Effects

Requirement: Validate the variation in the differential location of stars across the full field of view between pairs of 15 second images including the correlations of these different vectors.

The atmosphere induces small astrometric motions. The exact position of a star on the focal plane will shift even if all optics perform perfectly. Furthermore, the atmospheric shifts are predicted to vary across the field, so the guiders in the corner raft cannot remove this effect. There will also be some small shift in the positions of stars between the two exposures. This validation test determined if the astrometric shifts are reasonable.

Tim Axelrod obtained some calibration data from the LBT telescope where a crowded star field was deliberately trailed across the CCD during the exposure. The stars motion in the dimension transverse to the trailing then accurately measures the astrometric motions of the atmosphere in one-dimension. In general, he found that stars a few arcminutes from each other had highly correlated trails (also predicted by the atmospheric model), so we did not have a large enough field to accurately measure the correlation scale. On the other hand, we can accurately compare the astrometric jitter due to the atmosphere. Axelrod measured the mean 1-D shift of the atmospheric astrometric jitter in 15 second segments and produced the red dashed histogram in the bottom plot. We then simulated the a series of trails with the same trailing speed for 10 atmospheres. The atmospheric jitter is seen in the top plot. We also produced the average shift in 15 second intervals, and compared with the LBT data in the bottom plot. We then calculate the KS p-value probability of these two distributions coming from the same parent distribution. We do not expect perfect agreement since the results will vary from one value to the next, but it appears consistent.

Task 1E: Atmospheric Description

Describe in Detail the Physical Models for: 1) the turbulence in the dome, 2) the model and variation of cloud opacity, 3) the multi-layer/multi-component model (including wind shift) of atmospheric transmission (including water, molecular oxygen, ozone, and Rayleigh scattering) with cross-sections and density variation as a function of altitude, 4) the wavelength dependence of the seeing, 5) the model for wind speed and direction as a function of altitude and its annual variation appropriate for the LSST site, 6) the model for turbulence intensity as a function of altitude appropriate for the LSST site, 7) the model for the outer Kolmogorov scale as a function of altitude as well as the maximum of the Kolmogorov scale (the outer-outer scale), 8) the model for the total atmospheric seeing and its annual variation, 9) the model for atmospheric dispersion and its wavelength dependence

The description of the atmosphere and more detailed other atmospheric physical effects are found in Chapter 4.

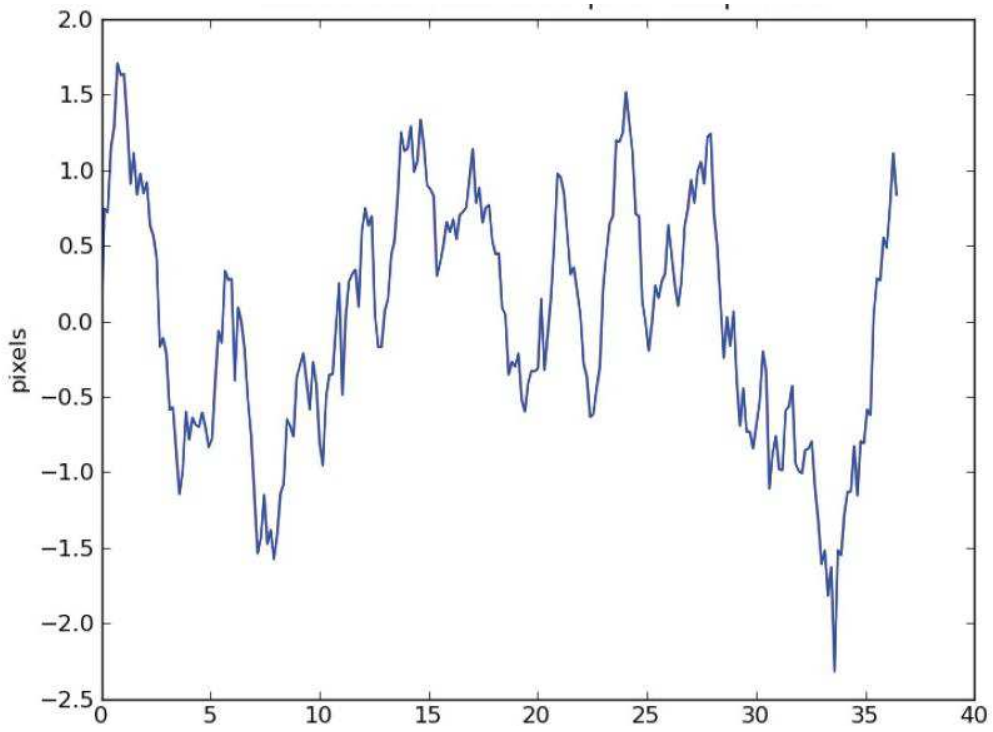


Figure B.6: 1-D centroid motion of the LBT data showing the atmospheric astrometry variation. The pixel size is 0.224 arcseconds. This motion compares favorably with the simulations in Task 1D.

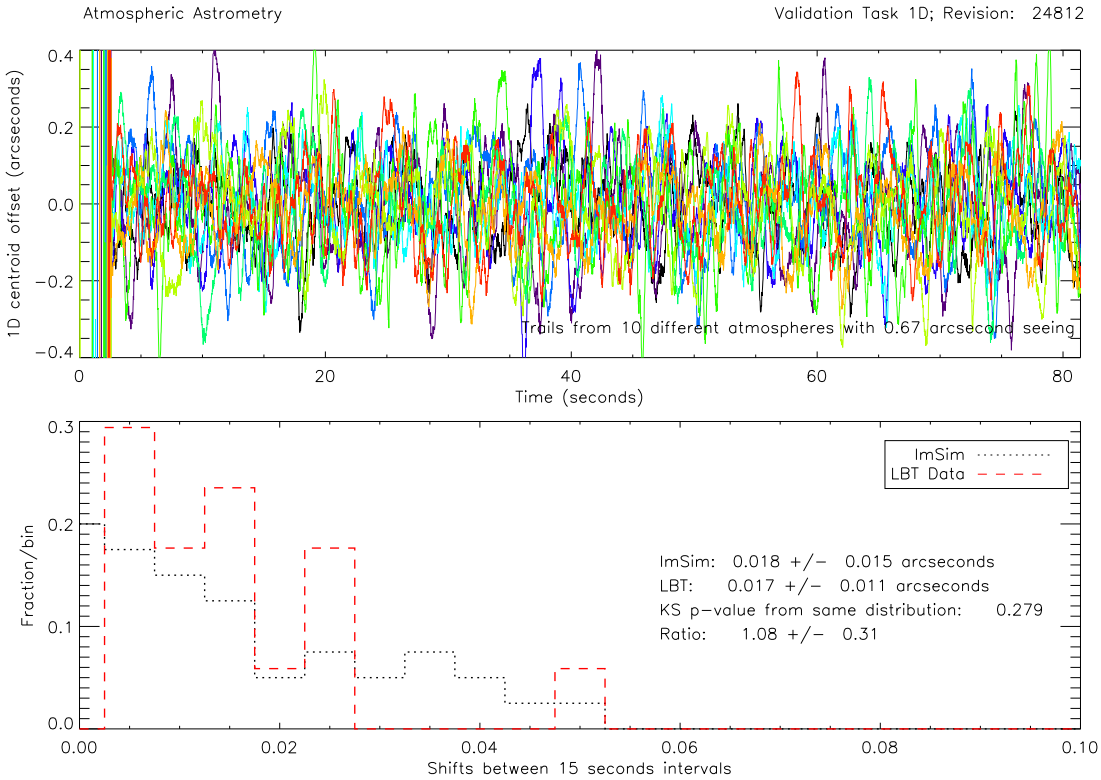


Figure B.7: Validation of the Atmospheric Astrometry for Task 1D.

Appendix C

Validation of Photon Propagation in the Telescope & Camera

Task 2A: Test of Raytrace Accuracy and Astrometric Scale

Requirement: Perform simulations to test the accuracy of the raytrace using a detailed spot diagram for off-axis pointings for the ideal positions of the optical elements. The focal plane position of the spot diagram tests the astrometric accuracy of the simulator. This will simultaneously test the reflection and refraction calculations, the wavelength-dependent index of refraction model, and the implementation of the correct optical prescription.

The accuracy of the entire raytrace system is best tested by a spot diagram. A spot diagram tests the ray intercept calculations, the ray reflection and refraction calculation, and the implementation of the optical design. Constructing a spot diagram involves tracing the rays, but turning off many components of the physics including the perturbations, charge diffusion, large angle scattering, and atmospheric physics. In this way, the final positions of the rays will have a complex pattern due to the details of the ideal optical design.

We compare the distribution of photon positions with that of optical engineering codes, such as ZEMAX, which are well-studied. The positions of photons using ZEMAX were provided by Ming Liang (NOAO). To do this comparison, we simulated a single star at 1.41 degrees off-axis (the center of raft 42). The resulting spot diagram has a highly complicated pattern that is wavelength-dependent, so we chose 5 wavelengths: 0.54, 0.58, 0.62,0.66, and 0.70 microns. This pattern is impossible to reproduce if there is any inaccuracy in the raytracing calculation or there is any mistake in the implementation of the optical design. Note this also tests the PSF shape from the perfect optics, as well as the astrometric scale since we also compare the absolute position in the focal plane. The figure below compares our calculation with that calculated from ZEMAX with the current optics model. For each spot diagram, we calculate the overlap of the rays in the ImSim positions with that of the ZEMAX rays. The photons are binned on the same scale (0.08 microns) and the overlap fraction is calculated on bins where we have at least ten rays. The raytrace is probably accurate on scales much smaller than this, but would require even more photons to test, and the accuracy is clearly already below 1% of a pixel.

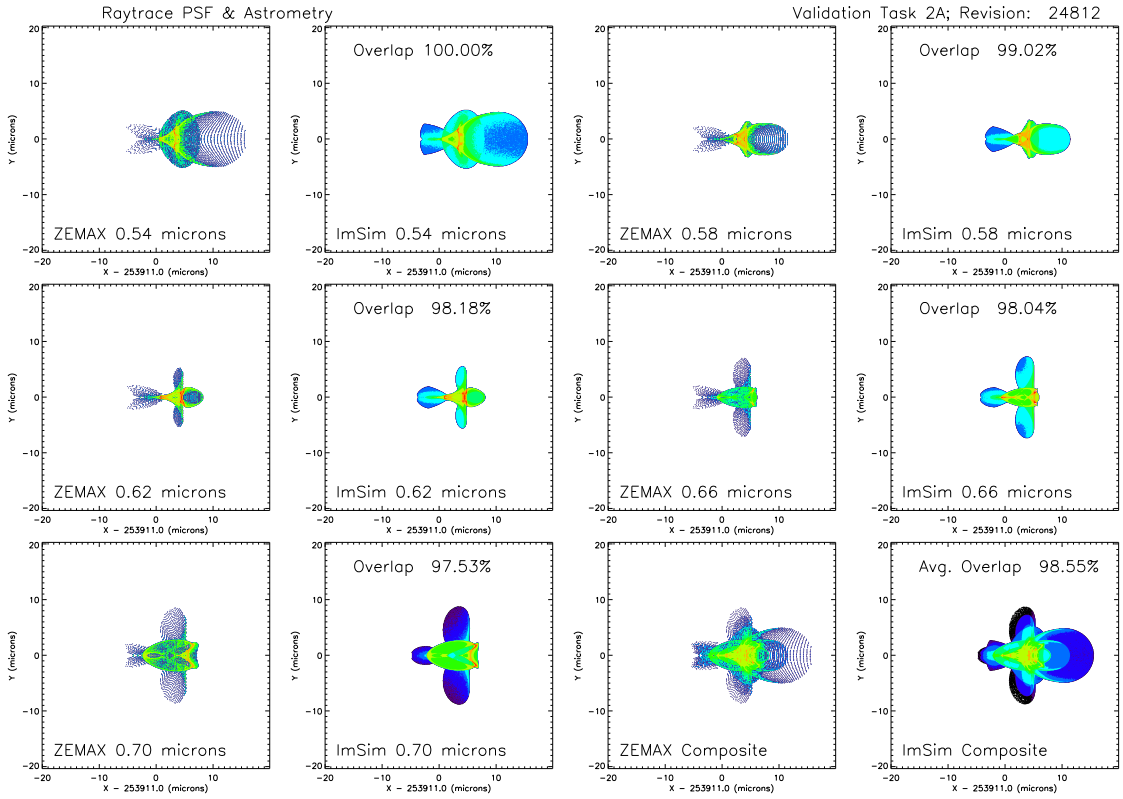


Figure C.1: Validation of the Raytrace accuracy for Task 2A.

Task 2B: Test of Perturbations of the Optical System on the PSF Size and Shape

Requirement: Perform simulations to demonstrate that the perturbation on the optical elements including misalignments of mirrors, lenses, filter, detector and surface deformations on the mirrors, lenses, filters, and detectors agree with the PSF budget.

There are various components of the model that have a significant effect on the actual PSF without atmospheric effects. Although, the details divided into the tolerances for every single imperfection of the telescope and camera (as in the engineering flowdown documents, Document-3535), we can roughly divide the components on the non-atmospheric PSF into four parts: the aberations from the intrinsic optical design, the sum of all perturbations and misalignments of all parts of the optical system, the charge diffusion in the silicon, and the internal seeing inside the dome and camera. In this task, we will test the tolerances with the output of the photon simulator for these four parts, and leave the validation at a great level of detail until the design matures and the physics models improve. The engineering flowdown lists the maximum allowable tolerance for these components at: 0.097, 0.259, 0.246, and 0.090 arcseconds, respectively. The assumption here is that if the telescope and camera is built properly then the simulator should also

meet these requirements.

First, we have tested a variety of realizations of the optics perturbations and then measured the PSF size with the atmosphere, internal seeing, and charge diffusion turned off. In general, the PSF from the optic design itself plus the perturbations has a very complex looking PSF. It will be important to keep this PSF contribution to also keep the PSF ellipticity low. This is shown in the first two rows in the Figure below. In the bottom row, we show the effect of the tracking, the designed (aberrated) PSF, and the internal seeing. Note that all of these PSF have the effect of the optical aberrations, since that cannot be turned off. The effect of charge diffusion is measured in Task 3B. We then measure the size using the formula in the SRD,

$$\mathrm{FWHM} = 0.663 \sqrt{\frac{(\sum_i I_i)^2}{\sum I_i^2}}$$

where I_i is the intensity of the ith pixel. The measured sizes are within tolerances in the engineering flowdown requirements (design 0.097″, internal dome seeing 0.090″, perturbations and misalignments 0.259″, and charge diffusion 0.246″). Thus, our optics PSF components are set so they are reasonably close to the tolerance, but do not have a significant chance of exceeding that tolerance.

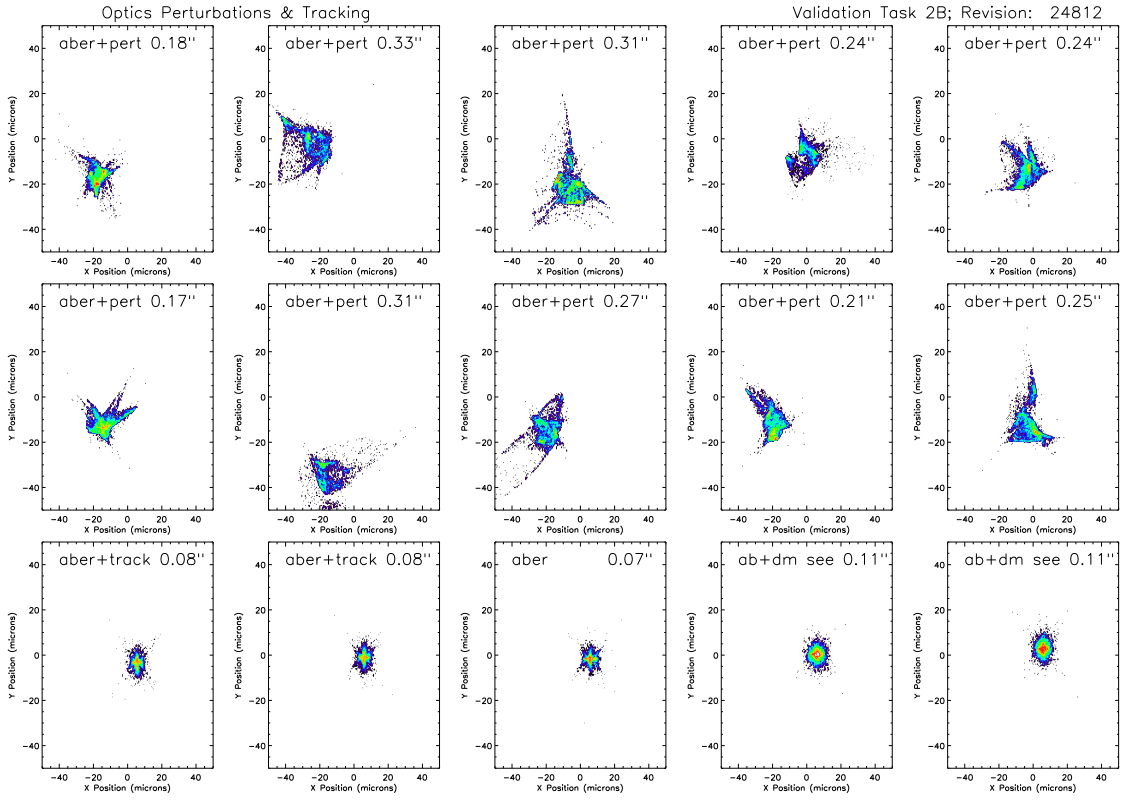


Figure C.2: Validation of the perturbations for Task 2B.

Task 2C: Test of Throughput of Optical Elements

Requirement: Demonstrate the throughput versus wavelength of the entire optical system is correct. This includes testing the mirror efficiency, the lens coatings, the detector A/R coatings, the filters, and the QE.

The throughput of the simulation involves making sure that the correction efficiency as a function of wavelength agrees with the nominal expectation. This insures an accurate photometric simulation. In the photon Monte Carlo approach, this simply involves counting photons at each stage of the simulation and making sure the correct amount were accepted and rejected at each stage of the process. In some cases, the agreement should be perfect since we applied a simple transmission curve (e.g. with the mirrors). In other cases, we have a more complex model (e.g. the multi-layer angle-dependence or the mean free path of the photon in Silicon) and it is our goal to match the throughput curve (in document-1777) by adjusting the physics parameters. In the following two sets of figures, we compare the throughput after each stage of the simulation: atmosphere, mirrors, L1+L2, Filter, L3, Detector, and total throughputs. The black line of document-1777 should be compared with the on-axis, airmass=1, no clouds or opacity variation simulation shown in the red curve. We also show a higher airmass and a larger off-axis angle (blue and green curves) to show how the throughput can vary across the field and from one exposure to the next. The only visible discrepancy occurs with comparison with the QE curve at wavelengths longer than 1.1 microns. This occurs because the mean free path in silicon at 1.1 microns in the physics model is much larger than the thickness of the device, so we cannot change any parameters to get those photons to convert.

Task 2D: Test of Monte Carlo Diffraction Model

Requirement: Model for Diffraction pattern due to the multi-layer spider support structure. Validation requires a comparison of the fast single photon ray trace method with the Fourier transform of the aperture.

We can compare the diffraction from the Monte Carlo edge diffraction model we described previously, with the exact calculation from diffraction theory. In the Figure below, we demonstrate that the basic pattern is reproduce in the Airy envelope and the spike pattern. On the left, we did a simulation of a star with a flat SED with diffraction turned on, but with the charge diffusion, optical perturbations, and atmosphere turned off. The simulation is done on-axis with no rotation of the spider. The right plot shows an elaborate calculation doing the full Fraunhofer integral where we integrate over the spider design. We chose to perform the integral averaging the results at wavelengths of 0.57, 0.62, and 0.67 microns. The calculation takes several days. The power in the core (less than 50 microns), spikes (within 25 microns of the pattern and not in the core) disk (all other regions) are similar. We construct our validation statistic by computing the rms difference between the intensity in these three components.

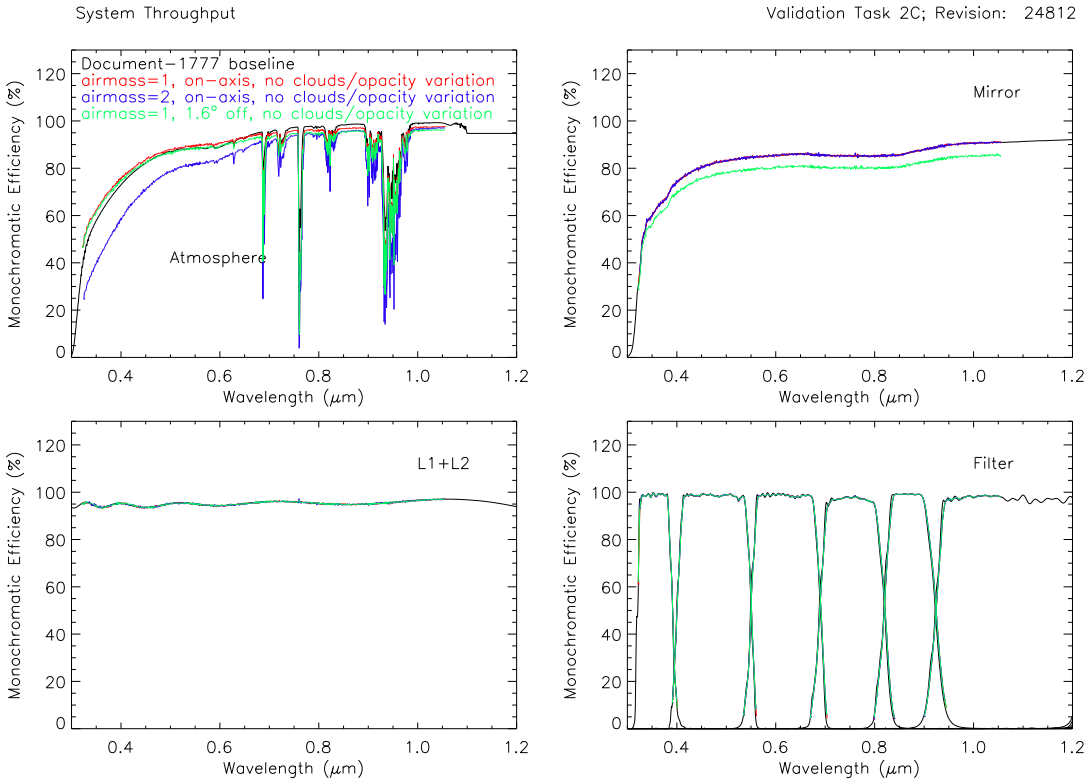


Figure C.3: Validation of the system throughput for Task 2C.

Task 2E: Description of Telescope & Camera Models

Requirement: Describe in Detail the Physical Models for: 1) the control system and the effect on perturbation/misalignment parameters including the values defined by physics and those that are under active control, 2) describe the dependence of the model on elevation effects (gravity), thermal effects and the performance variation in time (sunrise to sunset), 3) the focal plane layout model, 4) the detector misalignment model, 5) the use of OpSim inputs and their application in ImSim, 6) the time-dependent tracking model including rotation tracking, 7) the model for bore site pointing, field rotation, and focus control, 5) L1, L2 and L3 model of fixed surface variation due to fabrication effects, 8) the 3D model for the pre-compensation design L3 which includes pressure gradient and thermal gradient effects on the index of refraction and the resulting photon path 9) models of filters including off-axis effects, 10) the guider model: tracking performance with and without guiding, 11) vibration effects, temperature and gravity gradient effects on camera optics, 12) surface pitting, volume effects (bubbles) - glass quality, index of refraction variations in camera optics, 13) the model for the shutter variation in performance (sub-frame time dependence), 14) special 3D model for the pre-compensation design L1, 15) the thickness and flatness map, 16) the mechanical control of chip within rafts (gaps, function of elevation (gravity) and thermal drift), 17) the model the photo-response non-uniformity, 18) the large angle (incoherent) scattering model, 19) coherent light scattering (ghosts), 20) effects that rotate during an exposure (spider, atmospheric

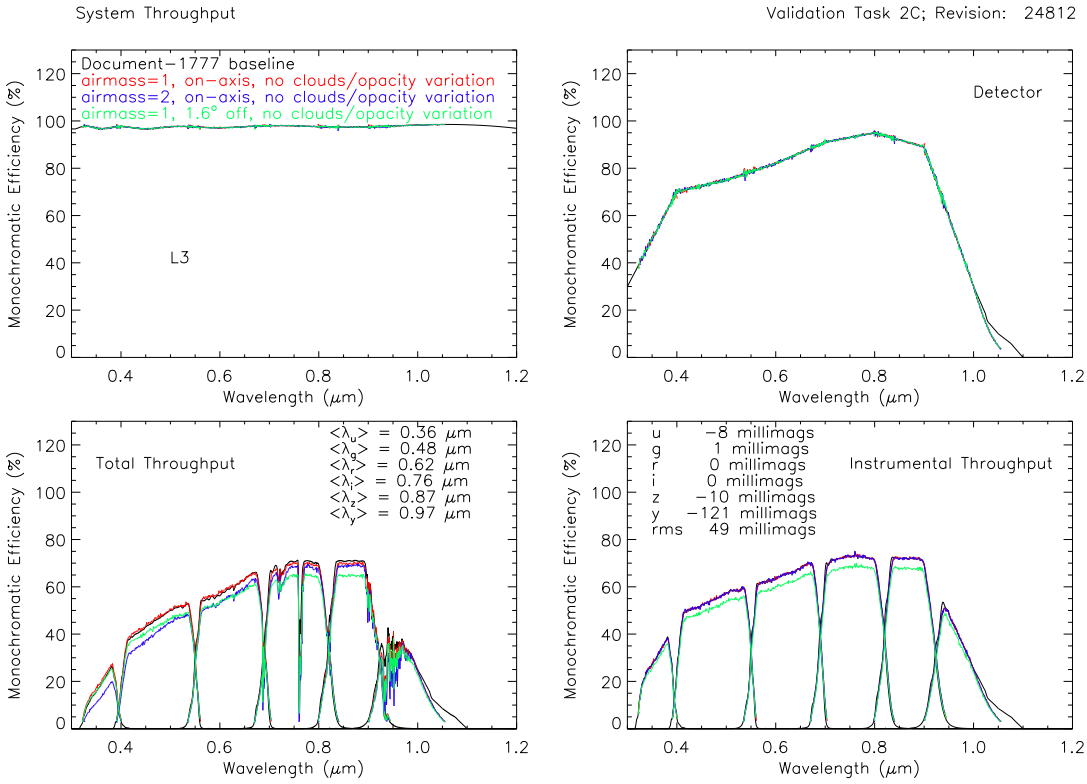


Figure C.4: Validation of the system throughput for Task 2C.

dispersion)

The description of the telescope and camera and more detailed other telescope and camera physical effects are found in Chapter 5.

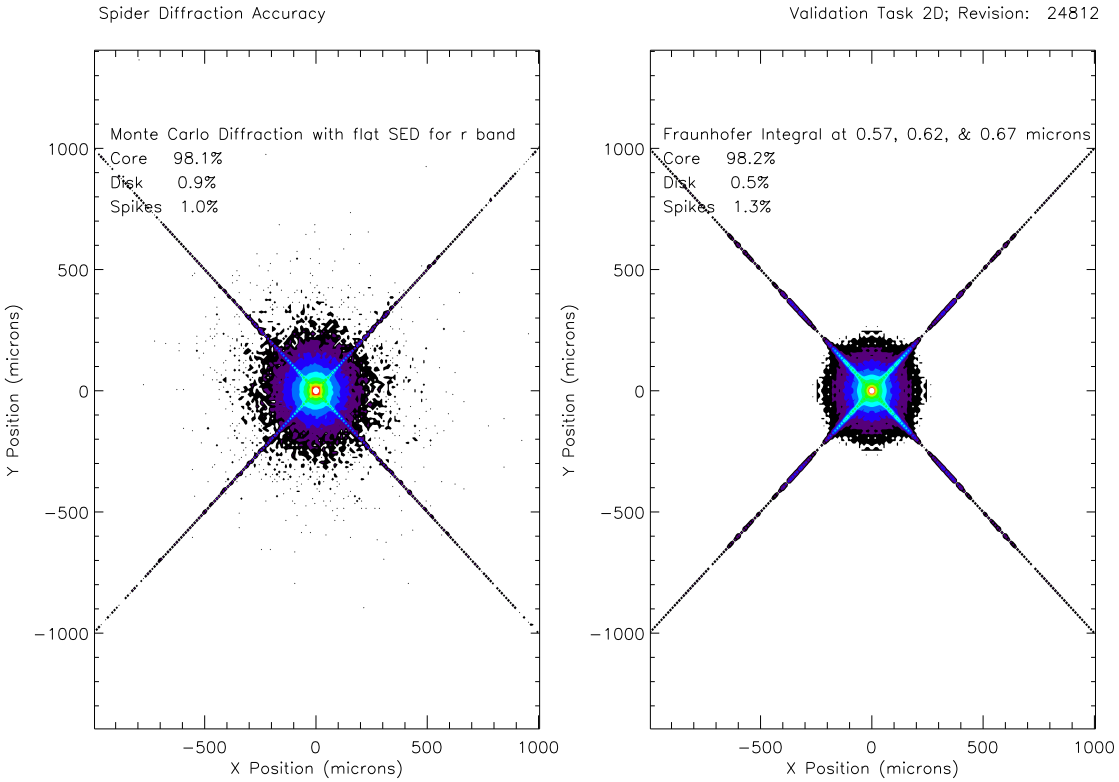


Figure C.5: Validation of the diffraction model for Task 2D.

Appendix D

Validation of Electron Propagation & Diffuse Physics Models

Task 3A: Test of Diffuse Simulation (for background & flats) and Camera Defect Model

Requirement: Demonstrate the parameterized model that produces either sky background or a dome flat due to a uniform illumination pattern and a given SED accurately reproduces large-scale effects and detector defects across the focal plane (vignetting, other gradients, QE variation, cosmic rays, hot pixels, other detector defects)

By simulating an exposure without any astrophysical sources but with a uniform background, we can study not only the telescope response to a flat illumination pattern but also quantify the level of detector defects. We can simulate a flat illumination pattern in two ways: either one photon at a time or using the empirical model for the background where we precomputed the illuminating response. We simulated a diffuse illuminating source both methods using an edge chip (R10_S00). We also compared that to the flats using the data where LSST prototype devices were tested on the Calypso telescope. The top plot shows the histogram of all the values in the pixels. In this way, the cosmic rays, hot pixels, light from the diffuse source, as well as the prescan/overscan pixel variations are shown. The Calypso data contained a large amount of excess electronic variation in response, so at this time we would not want to reproduce this distribution. It is likely though our hot pixel distribution and readout model need more variation. The bottom plot shows the count rate comparison between the empirical model and the exact photon calculation in 500 by 500 pixel regions. This demonstrates that we can match the detailed photon count rate as a function of radius of the empirical model (described in Chapter 6) with the exact detailed calculation. This also checks the overall count rate level prediction using our empirical approximation in Chapter 6. This task validates the approximate accuracy of the dome flats as well as the background.

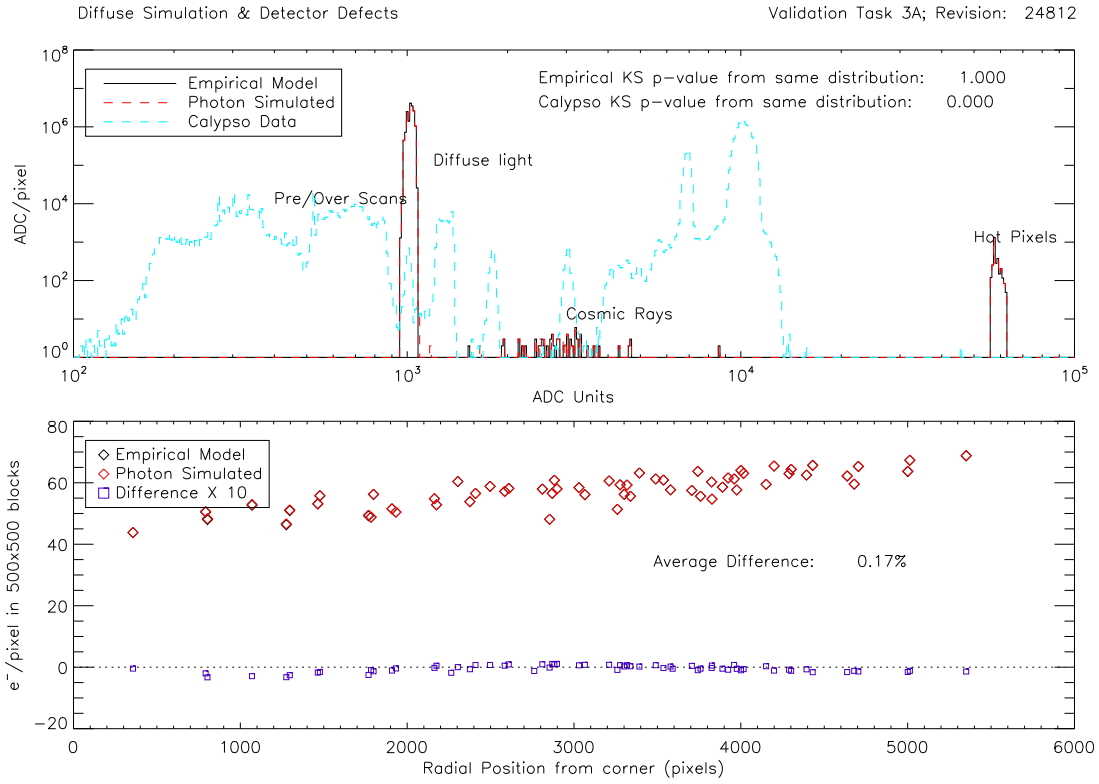


Figure D.1: Validation of the diffuse source and camera defect model for Task 3A.

Task 3B: Test of Charge Diffusion on the PSF Size

Requirement: Measure the effective diffusion of the electron diffusion in the Silicon and match the PSF budget in all bands

As a test of the charge diffusion model we ran the simulator with stars with a flat SED in different bands and varied the Silicon thickness. The atmosphere and optical perturbation were turned off, so the PSF size was mostly dominated by the charge diffusion. The PSF size at 0 thickness was subtracted in quadrature, so the pure charge diffusion contribution could be accurately measured. In the Figure below the square root dependence is clearly visible. A series of measurements were taken on prototype devices where X-rays from a Fe 55 source were measured. Andy Rasmussen then converted the X-ray event size (one or two pixel size) distribution into an estimate for the effective charge diffusion. We compare that measurement with the appropriate parameters at 100 microns and there is reasonable agreement. The size is also compared with the system design tolerances in Task 2B.

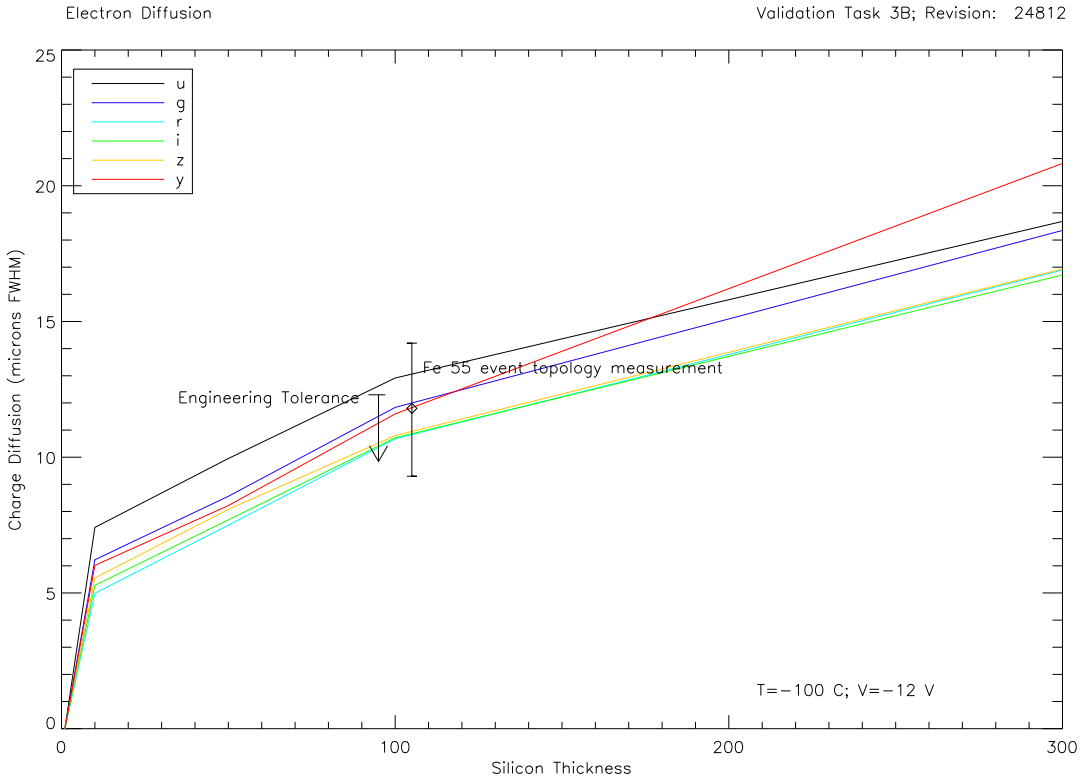


Figure D.2: Validation of the electron diffusion model for Task 3B.

Task 3C: Description of Diffuse and Camera Defect Models

Requirement: Describe in Detail the Physical Models for: 1) the sky background brightness, SED, and variation in time and moon phase, 2) the cosmic ray rates and pattern of cosmic ray tracks, 3) method of dome flat, darks, and bias generations, 4) model for dome illumination, 5) loss of light due to dust on surfaces, 6) full well blooming, 7) faulty pixel model (hot pixels, bad columns, dead or low response), 8) CCD non-linearity (low and high signal effects), 9) ADC non-linearity, 10) on-chip amplifier cross-talk, 11) off-chip amplifier cross-talk, 12) map of traps density (temperature and time-dependent model), 13) rate of failure of CCD, telescope, and camera, 14) readout noise model, 15) dark current model, 16) gain/offset for each amplifier and its variation, 17) the amplifier layout & readout orientation, 18) the pre-scan over-scan model, 19) charge transfer inefficiency, 20) fringing effects for the red response, 21) non-parallel field effects in CCD (variation in pixel size, 22) model of edge pixels (amplifiers boundaries and chip physical boundaries)

The description of most of these effects are included in Chapter 6 and 7.

Acknowledgments

We greatly acknowledge the support of the many members of the successful team that have either helped by writing photon simulation code, are involved with its extensive testing, or are creating instance catalogs for it. This team includes at Purdue: Zarah Ahmad, Justin Bankert, Emily Grace, Mary Ann Hodge, Suzanne Lorenz, Alexandra Lupu, Alan Meert, Satya Nagarajan, En-Hsin Peng, Nathan Todd, Mallory Young; at SLAC: Andy Rasmussen, Chihway Chang, Stuart Marshall; at U Washington: Andy Connolly, Rob Gibson, Lynne Jones, Simon Krughoff, Jim Pizagno, Nicole Silvestri, and Kreso Cosic at SLIP.

We also greatly acknowledge the contribution of enumerable ideas, models, designs, and external data by many members of the LSST collaboration: Tim Axelrod, Chuck Claver, Wei Cui, Bill Gressler, Kirk Gilmore, Zeljko Ivezic, Mario Juric, Steve Kahn, Jeff Kantor, Victor Krabbendam, Robert Lupton, Dave Monet, Phil Pinto, Jacques Sebag, Ian Shipsey, Don Sweeney, Tony Tyson, and Michael Warner.

Finally, we appreciate the support of the entire LSST data management team who have already requested and processed over 1 quadrillion photons.

Bibliography

Abahamid et al. 2004, A&A 422, 1123.

Ahmad, Z., ImSim Internal Document-12, "Hot Pixels & Offsets from Calypso Prototype Devices".

Bankert, J., ImSim Internal Document-3, "Background Model".

Beland, R. & J. Brown 1988, PhyS 37, 419B.

Bocass, M. 2004, Gemini Technical Note: CP Seeing & GS IQ.

Bodhane. B., Norman, B. Ellsworth, G. James, R. 1999, J Atmos. Ocean Tech. 16, 1884.

Cardelli, Clayton, Mathis 1989, ApJ 345, 245C.

Calzetti, D. et al. 2000, ApJ 533, 682C.

Coulman, C., Vernin, J. Coqueugniot, Y, Caccia, J. L. 1988, ApOpt 27, 155.

Claver, C., LSST control system model, priv. comm.

Claver, C., LSST tracking model, priv. comm.

Doty, J., Images of cosmic rays in thick devices, priv. comm.

Filippenko, A. V. 1982, PASP 94, 715.

Freniere, Gregory, & Hassler 1999, "Edge Diffraction in Monte Carlo Raytracing", SPIE 3780.

Grace, E., ImSim Internal Document-10, "Atmospheric Transmittance".

Grace, E., ImSim Internal Document-4, "Mie Scattering Verification".

Green, A. Wagner, J., Mann A. 1988, Appl. Opt. 27, 2266.

High-Resolution Transmission Molecular Absorption Database (HITRAN) (http://www.cfa.harvard.edu/hitran).

Ivezic, Z. et al. 2007, AJ 134, 3, 973.

Jones, L., Ivezic, Z., Claver, C., Document-1777, LSST Document, System Throughput.

Kolmogorov, A. N., Proc. of USSR Ac. Sci. 30, 299.

Krisciunas & Shaefer 1991, PASP 103, 1033.

Liou, K. N., "Introduction to Atmospheric Radiation", San Diego: Academic P, 2002.

Lorenz, S., ImSim Internal Document-5, "Galaxy Model Parameters".

Gilmore, K., cosmic ray rate, priv. comm.

Gilmore, K., QE variation, priv. comm.

Gilmore, K., read noise and dark current, priv. comm.

Gilmore, K., hot pixel/column rate, priv. comm.

Gilmore, K., ADC model, priv. comm.

Gressler, B., Collection-2097, LSST Document, Optical Design v3.3.

Lupu, A., ImSim Internal Document-1, "First Set of Optics Parameters".

Meert, A., ImSim Internal Document-6, "Second Set of Optics Parameters".

NOAA NCEP/NCAR Reanalysis Monthly Database (appropriate for LSST site location).

Patat., F., Ugolnikov., O. S., & Postylyakov, O. V. 2006, A&A 455, 385.

Phillip & Taft 1960.

Rajakan, K. Sing, R., Shechun J., Solid State Electronics 22, 793.

Rasmussen, A., model of electron interactions in silicon, priv. comm.

Rasmussen, A., Document-7821 & 7822, LSST Document, Focal Plane Geometry.

Rothman, L. S. et al. 2009, J. Quant. Spectr. Radiat. Transfer 110, 533.

Sander, S. P. et al., 2006, Chem. Kin. & Photochemical Data for Use in Atmospheric Studies, Pasadena JPL Pub, 2006.

Sellmeier, 1871.

Schott Glass Catalog Online Database.

Sérsic, J. L., 1963, BAAA 6, 41S.

Thomas & Stramnes, Radiative Transfer in the Atmosphere and Ocean, Cambridge: Cambridge UP, 1999.

Todd, N., ImSim Internal Document-11, "DC3b Optical Distortions".

Vernin, J. et al. 2000, Gemini RPT-AO-G0094.

Warner, M., LSST tracking model, priv. comm.

Young, M., ImSim Internal Document-9, “Atmosphere Parameters”.