

# Measurements and Simulations of the Brighter-Fatter Effect in CCD Sensors.

Craig Lage, J. Anthony Tyson

Department of Physics

University of California - Davis

cslage@ucdavis.edu

## Abstract

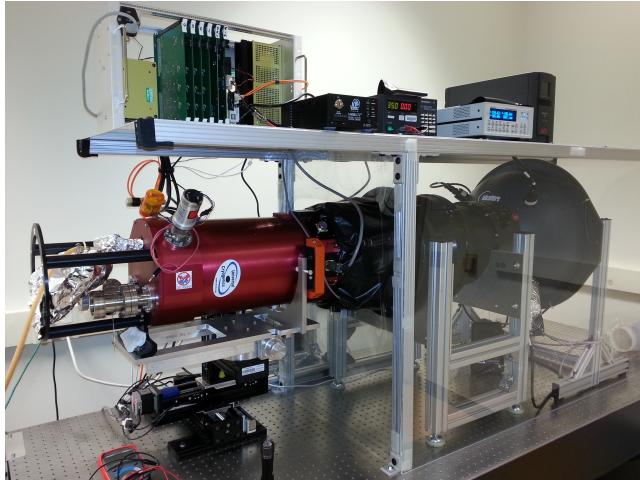
*Keywords:* LSST, modeling, camera, CCD, simulation, diffusion, image processing. Reduction of images and science analysis from ground-based telescopes such as the LSST requires detailed knowledge of the PSF of the image, which includes components attributable to the instrument as well as components attributable to the atmosphere. Because the atmospheric component is constantly changing, the PSF is typically extracted from each image by measuring the size and shape of star images across the CCD, then building a fitting function which is used to model the PSF for analysis of extended sources such as galaxies. Since the stars in each CCD field have a range of brightnesses, accurate knowledge of the PSF for point sources of varying brightness is essential. It has been found that in thick, fully-depleted CCDs, the electrostatic repulsion of charge stored in the collecting wells gives rise to a larger and slightly more elliptical PSF for brighter stars. This “brighter-fatter effect” has been reported on in some detail in the literature. In this work, we report on direct and indirect measurements of this effect in prototype LSST sensors, and describe a detailed physics-based model of the electrostatics and charge transport within the CCD. We then describe how we determine the free parameters of the model, and show the agreement between the model and the measurements.

## 1 Introduction

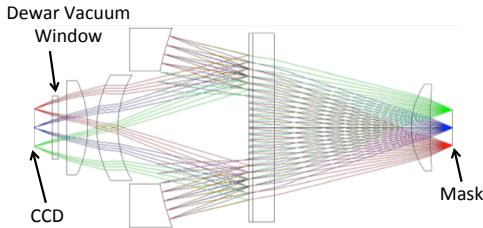
CCDs exhibit position and intensity dependent charge transport anomalies which must be understood and corrected in order to reach the weak gravitational lens shear measurement goals. Long misinterpreted as QE variations, we must correct for these small scale astrometric variations in the thick fully depleted CCDs. The effective pixel grid is not exactly rectangular it is warped. Several effects contribute to this charge transport redistribution: fringing transverse electric fields around the edge of the chip space charge repulsion of arriving photoelectrons near cores of bright stars and chemical potential variations frozen into the silicon. Nearly all the small scale features we see in flat field images actually represent shifted pixel boundaries (and hence shifting pixel positions). This means we should be including them as a remapping rather than dividing them out. The coordinate mapping is frozen in the chips, so there should be enough information to constrain the mapping from dithered star array positions. Correlations in flat fields provide added information, which combined with the dithered star array data can constrain a device physics model.

The “brighter-fatter” (hereafter “B-F”) effect has been reported on by several authors ([1], [2], [3], [4], [5], [6]). While there have been both empirical measurements of the effect and theoretical treatments, we believe that by building a detailed physics-based model of the CCD, we can help to understand the effect and point the way toward suppressing it through data management techniques.

This work is organized as follows. First, we describe the techniques used for measuring the B-F effect in the lab, both direct measurements of the sizes of artificial stars and indirect measurements of the B-F effect as extracted



(a) LSST Optical Simulator



(b) Schematic of f/1.2 Optical System

Figure 1: The UC Davis LSST Optical Simulator [9]. The CCD is contained in the Dewar chamber on the left, the catadioptric optical system is in the center, and the illuminating sphere is on the right. The control electronics are on top. The right-hand image shows the optical system, with the mask on the right, the optical system in the center and the Dewar vacuum window and CCD focal plane on the left.

from pixel-pixel correlations ([1]). We then describe the code which is used to simulate the electrostatic effects in the CCD sensor, how we tune in the free parameters of the model, and how well the model agrees with the measurements. We then discuss plans and progress for incorporating the results of these detailed simulations into the much faster PhoSim ([7], [8]) simulator, which simulates the entire LSST instrument on a photon-by-photon basis, and we then conclude.

## 2 Measurements of the Brighter-Fatter Effect

Measurements of the Brighter-Fatter Effect are performed using the UC Davis LSST Optical Simulator, which is described in detail in Tyson, et.al. [9]. The system and a schematic of the optical path are show in Figure 1.

The system enables a mask containing artifical stars to be projected onto the CCD being tested. Since the CCD used is a prototype version of the CCDs intended to be used in the final LSST instrument, and since the Optical Simulator has been engineered to duplicate the characteristics of the f/1.2 beam in the LSST telescope, it is expected that the results obtained will be representative of those in the final LSST images. In this work, the CCD used is an STA3800 prototype CCD produced by the Imaging Technology Laboratory [10]. The CCD is comprised of 16 segments, each containing 4000x509 pixels which are 10 microns square, for a total imaging area of 4000x4072 pixels or 40mm x 40.72 mm. In the final LSST instrument, the image scale is such that each pixel will have an angular size of 0.2 arcseconds. The CCD has a thickness of 100 microns of fully-depleted silicon in order to enhance the quantum efficiency in the near infrared (up to  $\lambda = 1.05\mu\text{m}$ ) region.

Two different techniques are used to characterize the B-F effect on the Optical Simulator. The first is direct imaging of artificial stars of differing intensities. The second is measurement of flat field images and extraction of pixel-pixel correlations, as described in [1]. The details of these measurements are described in the next sections.

### 2.1 Direct Measurements of Spot Sizes

Using the LSST Optical Simulator, we are able to directly measure the B-F effect on arrays of artificial stars (i.e.”spots”). To generate the spot images, a mask is placed at the focal plane of the optical simulator (on the right in Figure 1). The mask consists of a 2mm thick glass plate with an opaque metallic layer, in which has

been patterned an array of  $\approx 40000$  holes, each of which is about 30 microns in diameter [11]. The 30 micron size has been chosen to give images approximately 0.6 arcseconds in diameter, simulating the expected conditions at the LSST on nights of good seeing. The mask is illuminated using an illuminating sphere which illuminates the mask with a uniform intensity, and the image focused onto the CCD under test. For all of these tests, an LSST r-band filter is used (center wavelength of 621 nm), and the CCD is maintained at  $-100^\circ$  through the use of an LN2 reservoir and a feedback-controlled resistive heater.

The image of a single spot contains too much shot noise to allow determination of the spot size to the level of accuracy needed to characterize the B-F effect, which typically modifies the spot sizes by 1% or less. In order to reach the required level of accuracy, we image a large number (typically 500-1000) of spots simultaneously for each region of the CCD, and then use forward modeling techniques to extract the best estimate of the spot size. The details of the forward modeling technique used here are described in Appendix A, but a brief description of the modeling sequence is as follows:

1. Use sextractor to identify a list of spots.
  - (a) Use sextractor central pixel location, but not size or exact offset within pixel.
  - (b) Use a constant window (“postage stamp”) for all spots. Here we are using 9x9 pixels.
2. Assume all spots have the same shape, but allow variable peak intensity and offset within central spot.
  - (a) Calculate first moment of postage stamp to determine offset within central pixel.
3. Assume a 2D Gaussian, calculate expected signal in each pixel
4. Find  $(\sigma_x, \sigma_y)$  which minimizes:  $\sum_{N_{\text{spots}}} \sum_{x,y} (\text{Measured}_{n,x,y} - \text{Calculated}_{n,x,y})^2$

The forward modeling routine was tested on both measured and mock data to ensure it is giving sensible results, and the results of these tests are shown in Figure 2. These results are quite satisfactory, and give confidence in using this technique to characterize the B-F effect.

We next apply the above techniques to images generated with a range of photon fluxes to characterize the B-F effect. Typically we do this by keeping the light intensity fixed and increasing the exposure time to increase the spot flux. Figure 3 shows a typical result. There are several aspects of these plots which are universally seen. First, the slope of the line in the Y-direction (parallel to the CCD channel stops) is always greater than the slope in the X-direction (parallel to the CCD gates). This is because the electric field which confines the collected charge due to the channel stops is larger than the confining electric field due to the CCD transfer gates. This causes the spots to become slightly elliptical as the flux increases, and is an important effect to model because it will have a systematic effect on the measurement of galaxy shapes and consequently on measurements of weak lensing. Second, while we expect that the modeled X and Y sizes should be equal at zero intensity, we typically see a small offset even when extrapolated to zero flux. This is not fully understood, but is believed to be due to residual distortion in the optical system. Third, we typically see a change in slope at low intensities which is apparent in Figure 3. It is not yet known whether this is a real or an instrumental effect, but it is hoped that subsequent modeling work will help shed light on this.

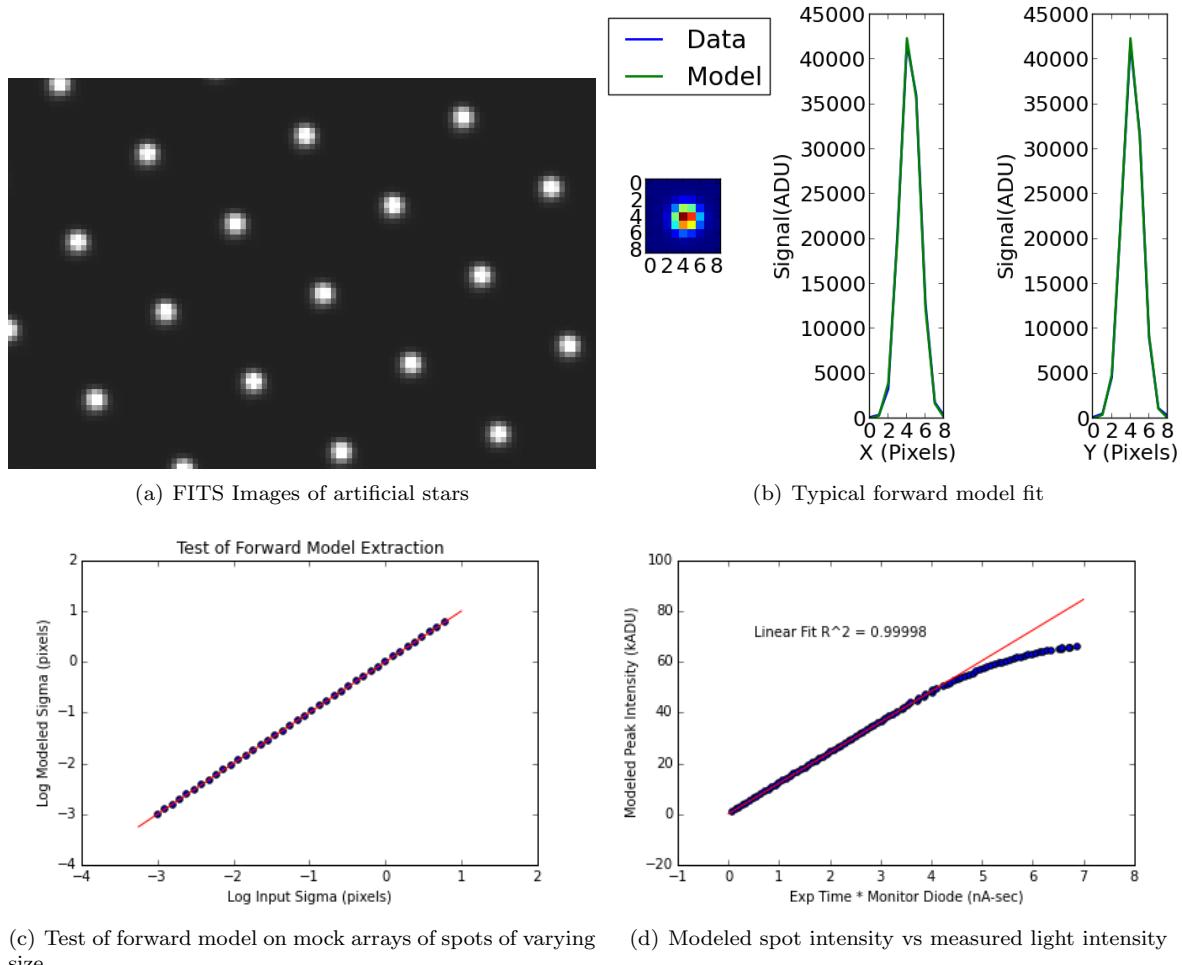


Figure 2: Forward modeling of spots. (a) A typical image of the array of spots. (b) The fit obtained between the forward model and the measurement on a representative spot. This shows the 9x9 postage stamp and X and Y slices through the central pixel. The units are in ADU, with a CCD gain of  $\approx 2.4$  electrons/ADU. (c) The results of the forward model fit obtained on mock arrays of spots of a range of sizes. (d) A comparison of the average modeled peak intensity of an array of spots to the measured light intensity times the exposure time.

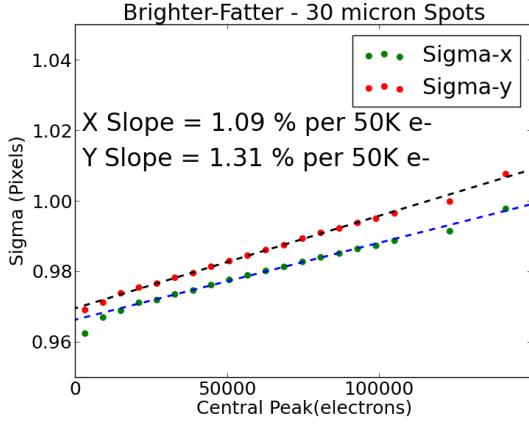


Figure 3: Forward modeled spot sizes as a function of the average stored charge level in the central (peak) spot. The Y-direction is parallel to the CCD channel stops, and the X-direction is parallel to the CCD gates. Note that the spots become both larger and more elliptical as the flux increases.

## 2.2 Pixel-Pixel Correlation Measurements

A second method of characterizing the B-F effect, as originally described by Antilogus, et.al. [1] involves measurements of pixel-pixel correlations averaged over large numbers of flat images. If, through random variation, a pixel accumulates more charge than the average, electrons are repelled from this pixel into surrounding pixels. This causes positive correlations with surrounding pixels, since an increase in charge of a given pixel causes a corresponding increase in surrounding pixels. By measuring and averaging large numbers of flats, these correlations can be seen and quantified. However, there are several subtleties to the analysis of this data. First, because the light intensity of the flats is usually not perfectly uniform, it is best to subtract pairs of flats before calculating the pixel-pixel correlations. After subtracting pairs of flats, we are left with a pixelized difference image  $\text{Diff}(i,j)$ , from which the correlations are calculated as follows. Typically the correlations are divided by the variance in order to yield a covariance.

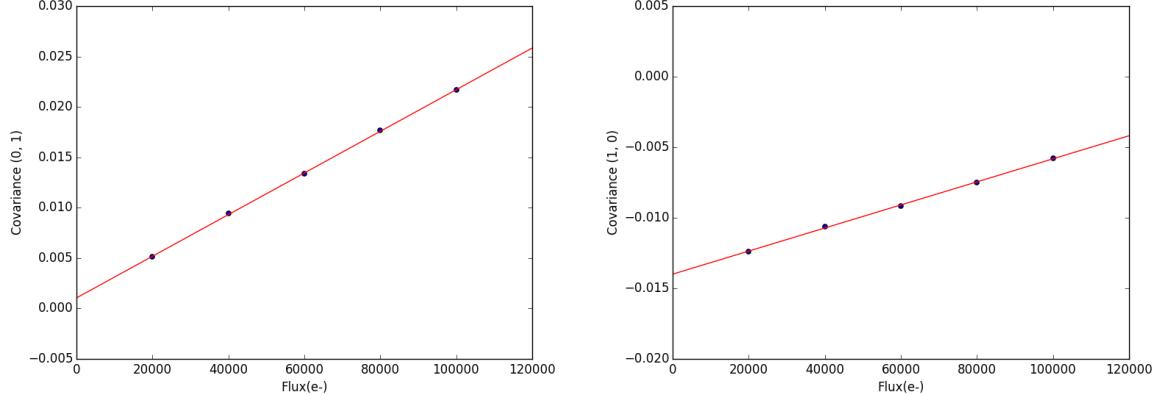
$$\text{Corr}(\delta i, \delta j) = \langle \text{Diff}(i, j) - \langle \text{Diff}(i, j) \rangle \rangle * \langle \text{Diff}(i + \delta i, j + \delta j) - \langle \text{Diff}(i, j) \rangle \rangle \quad (1)$$

$$\text{Cov}(\delta i, \delta j) = \frac{\text{Corr}(\delta i, \delta j)}{\text{Corr}(0, 0)} = \frac{\text{Corr}(\delta i, \delta j)}{\text{Var}(i, j)} \quad (2)$$

Where this last equality results because the correlation of the pixel value to itself is just the variance. A second complication is that there are other effects, such as time correlated electronics noise, that can cause pixel-pixel correlations, and these must be separated from the correlations due to the B-F effect. To do this, we calculate the covariances at a range of flux levels and extrapolate back to zero flux in order to identify the component due to the electronics. The results of this procedure are shown in Figure 4. It can be seen that for some pixels (such as pixel(0,1) shown in Figure 4 (a)), the correction is small, while for other pixels (such as pixel(1,0) shown in Figure 4 (b)) the effect is much larger. In fact, subtracting off the intercept converts the covariance of pixel(1,0) from a negative to a positive value, as shown in Figures 4 (c) and (d). This large impact on this pixel is believed to be due to the fact that the reference pixel(0,0) and pixel(0,1) are read out sequentially by the readout electronics, so incomplete recovery of the readout amplifier and correlated double sampler can induce strong correlations between these pixels.

## 3 Simulations of the Brighter-Fatter Effect

As part of this work, we have built a detailed physics-based model of the electric field and charge transport within the CCD, and compared the simulations which result to the measurements described in the last section. This



(a) Pixel covariance vs flux for pixel(0,1) (parallel direction). (b) Pixel covariance vs flux for pixel(1,0) (serial direction).

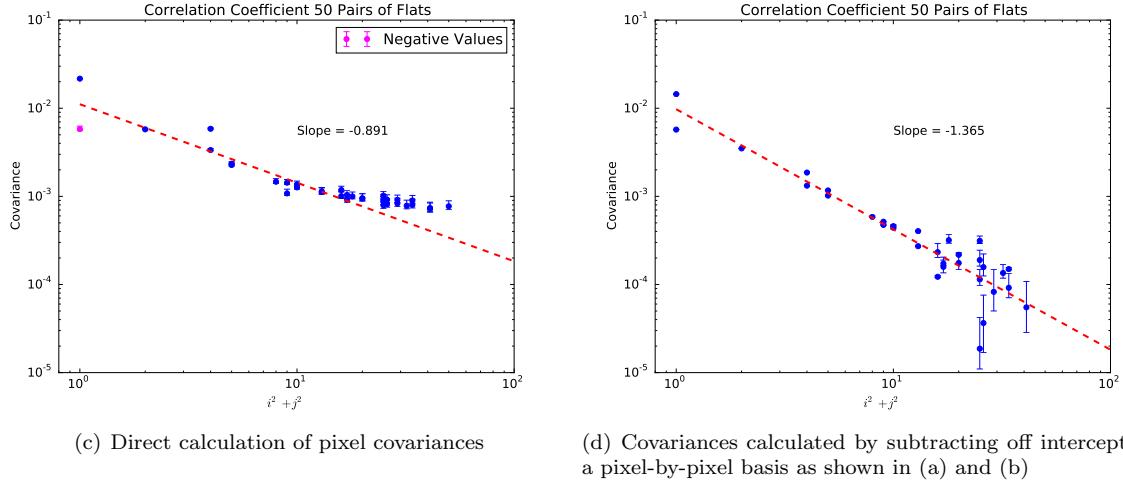


Figure 4: Measurements of pixel covariances taken by averaging 50 pairs of flats at each flux level. Figures (a) and (b) show the measured covariances as a function of flux. Figure (c) shows the directly measured covariances at the highest flux level, not accounting for the non-zero intercept. Figure (d) shows the calculated covariances after subtracting off the intercept on a pixel-pixel basis to account for noise in the electronics. Note that this subtraction changes the sign of the (1,0) covariance in addition to changing the slope of the curve.

section describes the simulation methods.

### 3.0.1 Solving for the Electric Field

Figure 5 shows cross sections of the basic CCD structure. An excellent overview of the physics of fully depleted CCDs is given by Holland, et.al. [12]. Figure 6(a) shows the basic operation of charge collection in these CCDs, where photons are incident on the “backside” of the CCD device. The incident photons create hole-electron pairs in the silicon, and a large electric field is set up so that positively charged holes are swept to the back side and negatively charged electrons are swept to the front side. The electrons are collected under the gates which define the pixels of the device, where they are read out during clocking of the collecting gates. The CCDs in this work are comprised of fully depleted p type silicon which is  $100\mu\text{m}$  thick, and pixels which are  $10\mu\text{m}$  square, which is a vertical to horizontal ratio of 10:1. Ideally the electrons propagate purely vertically so that a photon incident in a given pixel is collected in the same pixel. However, horizontal electric fields can cause electrons to propagate into pixels offset from the incident photon, resulting in distortions of astronomical images. In order to simulate this process, it is necessary to first simulate the electric field in the silicon. This is done by solving Poisson’s equation within a discretized volume given the potentials on the boundary of the volume and the charges within the silicon volume. Figure 6 (b) shows the typical simulation of a  $100\mu\text{m}$  cube comprised of  $10\times 10$  pixels, with a grid size comprising 32 grid cells across each pixel. This typical simulation is a  $320\times 320\times 320$  grid, with each grid cell representing a cube  $0.31\mu\text{m}$  on a side. In order to simulate the electron paths, it is necessary to know the potential and electric field within the silicon volume. These quantities are calculated by solving Poisson’s equation within this volume by using Multi-Grid methods. The details of solving Poisson’s equation are described in more detail in Appendix C.

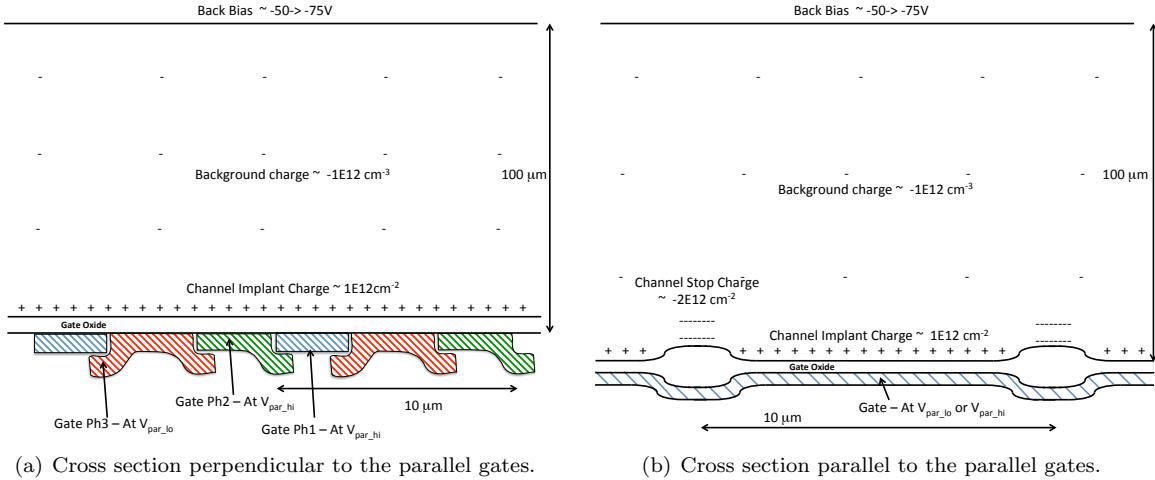
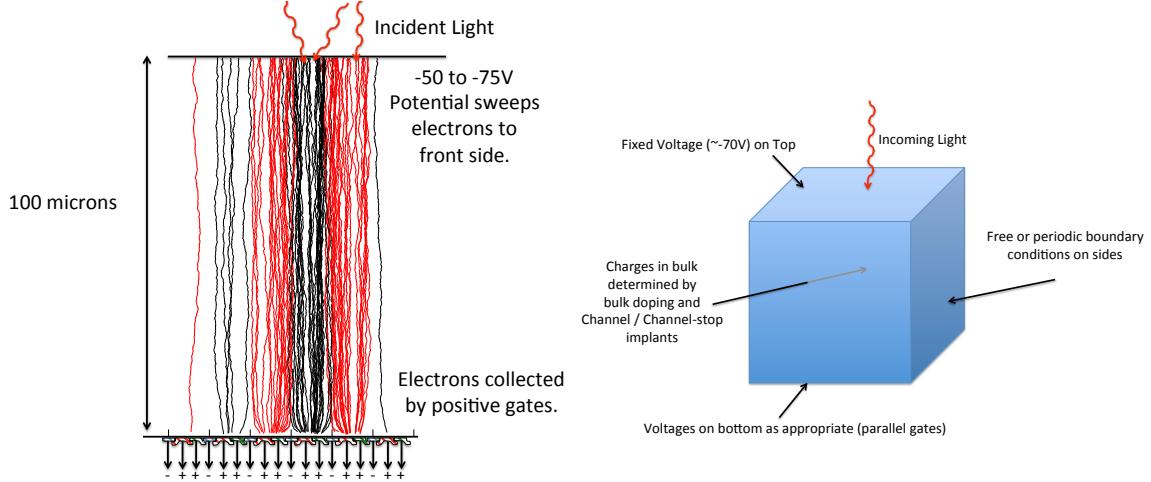


Figure 5: Basic cross section of the CCD structure. See Holland, et.al. [12] for a detailed description of the operation of the fully depleted CCD. Here light is incident at the top of the diagram and photoelectrons are collected above the parallel gates shown on the left. Note the difference in vertical and horizontal scale.



(a) Overview of electron paths. This diagram is to scale, (b) Basic simulation volume, comprised of a  $100\mu\text{m}$  cube. illustrating the very high aspect ratio of the pixel structure.

Figure 6: Overview of simulation strategy. The typical simulation is a  $100\mu\text{m}$  cube, containing  $10 \times 10$  pixels each  $10\mu\text{m}$  square and  $100\mu\text{m}$  thick. The volume is discretized on a  $320 \times 320 \times 320$  grid, with each grid cell representing a cube  $0.31\mu\text{m}$  on a side.

Figure 7 and 8 are plots of the potential solution for a typical set of assumptions. The contour plot in the lower left of Figure 7 clearly shows the islands of high potential where the photoelectrons are stored. The upper left of Figure 8 also clearly shows the buried channel region generated by the channel implant. This creates a potential well displaced from the silicon-silicon dioxide interface where the electrons are stored. Since recombination velocities at this interface are much higher than in the bulk silicon, storing the charge away from the interface greatly increase the charge lifetime and charge transfer efficiency. It can also be seen that the implant charge profiles in this simulation are assumed to be square wells, whereas a more physically correct assumption would be for them to have graded profiles. This will be discussed in more detail in the Section 3.0.3.

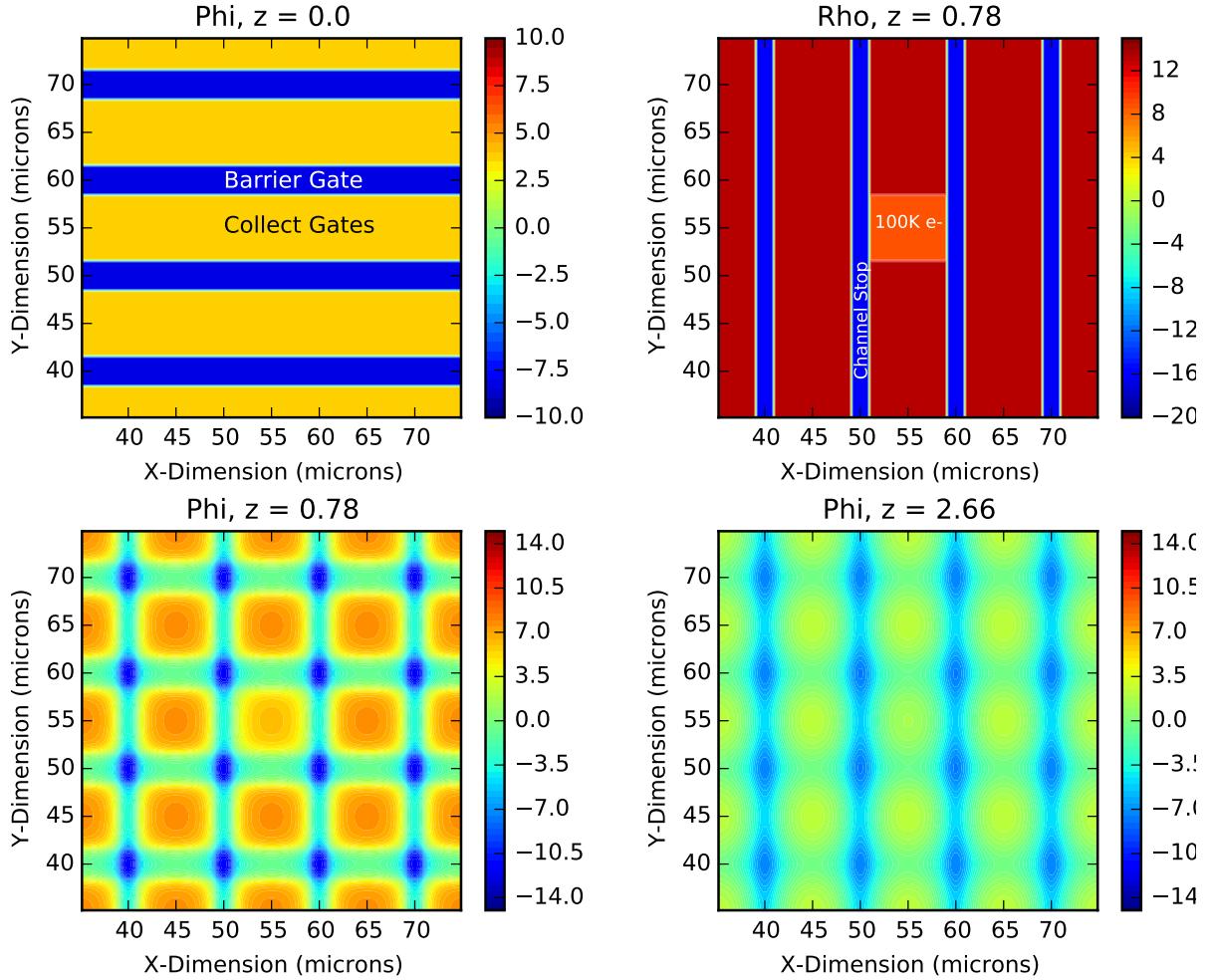


Figure 7: Poisson's equation solution of pixel array with one pixel containing 100K electrons. Upper left: Potential boundary conditions at  $z=0$ , showing the voltages on the collecting gates. Upper right: Charge density just above the gate oxide, showing the charge due to the channel stop implant and the charge collected in one of the potential wells. Lower left: 2D potential slice at  $z=0.78$  microns, showing the potential wells (most positive potentials) in which collected charges are stored. Lower right: 2D potential slice at  $z=2.66$  microns, as the potential is becoming more uniform.

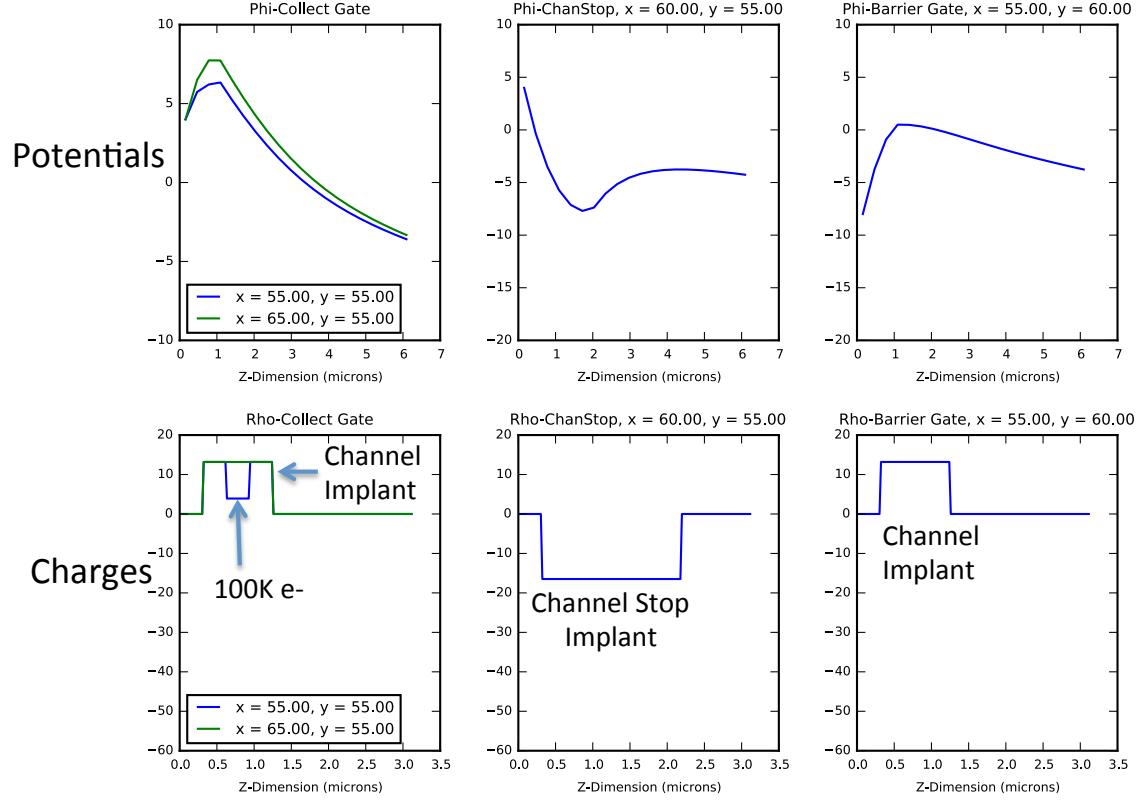


Figure 8: 1D profiles in the Z-direction of the Poisson's equation solution. Top Row: Potentials in the center of a collecting gate ( $V_{\text{parallel-hi}}$ ), the center of the channel stop region, and the center of a barrier gate( $V_{\text{parallel-lo}}$ ), respectively. Bottom Row: Charge densities in these same regions. The green lines are with all pixels empty, and the blue lines are with one collecting gate containing 100K electrons.

### 3.0.2 Simulating the Electron Paths

Once we have a solution for the potential and electric field within the silicon, the next step is to calculate the electron paths, so we can simulate where the photoelectrons created by the incident photons are ultimately stored. The details of how these calculations are done are in Appendix D and we focus here more on the results. Figure 9 shows the impact of diffusion on the electron paths and on the sharing of electrons between pixels.

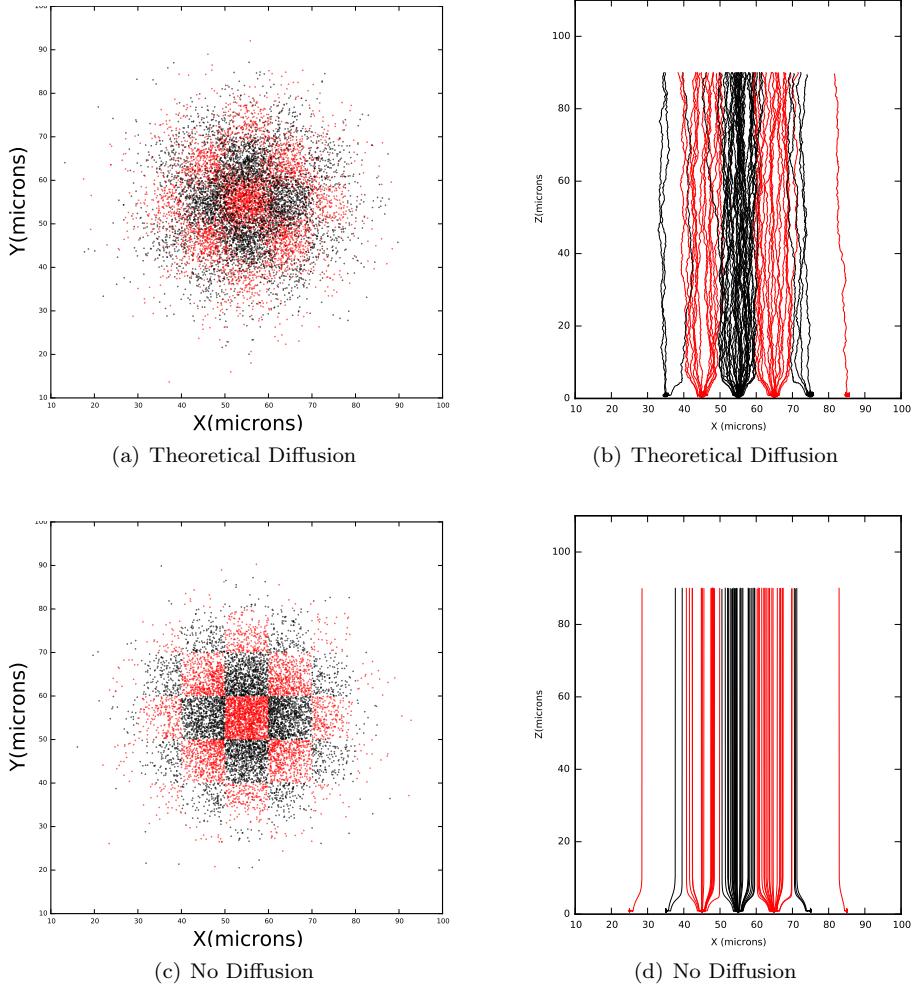
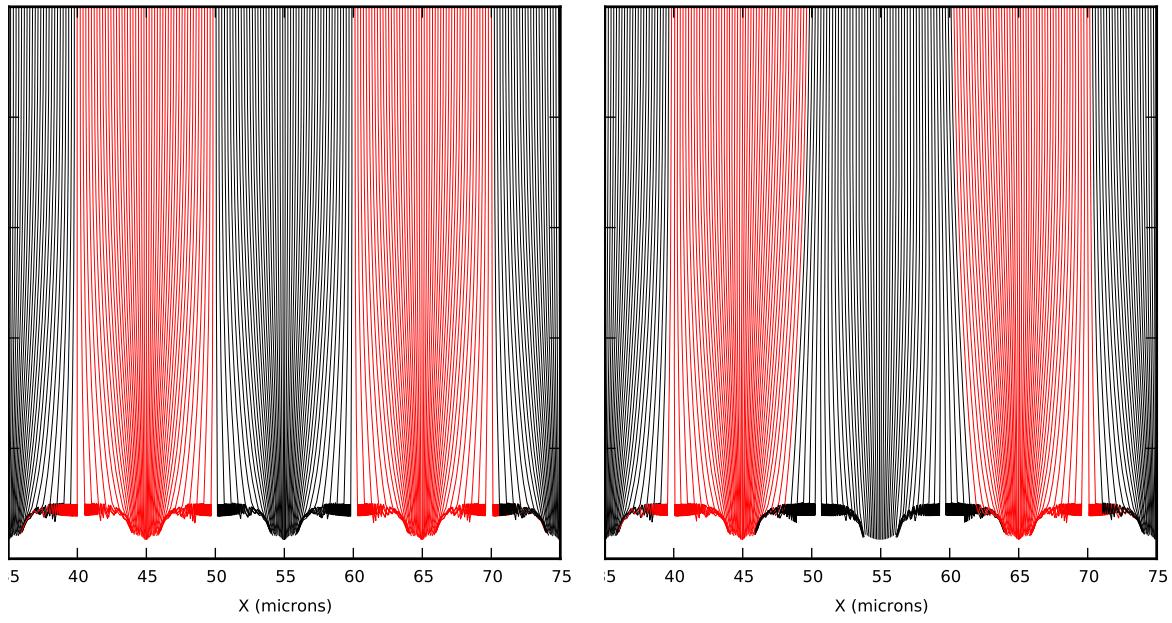


Figure 9: Impact of electron diffusion. Electron paths are color coded depending on within which undistorted pixel they were created.

The presence of stored charge in a pixel repels incoming electrons, pushing them into surrounding pixels; this is the cause of the B-F effect. Figure 10 shows the impact of one pixel containing 100K electrons on the surrounding electron paths.

Using the simulations of the electron paths, one can also study the distortion of apparent pixel boundaries due to the B-F effect. To do this, we use a binary search to locate the pixel boundaries, with results as shown in Figure 11. To make this plot, we first solve Poisson's equation with the central pixel containing charge (here 160,000 electrons). Then, with diffusion turned off to eliminate the stochastic effects, we propagate electrons down to the collecting wells to determine in which pixel they end up. We perform a binary search along a radial path starting at the pixel center until we have located the pixel boundary to a desired level of precision (here 0.001 micron). This allows us to determine the changes in the effective pixel boundaries and in the pixel areas. For these plots, the pixel corners are first found through binary search, and then each pixel edge is delineated by some number NumVertices. In these plots, NumVertices = 64, so the pixel polygon is delineated by 260 distinct vertices. The usefulness of these pixel characterizations will be discussed in the following sections.



(a) Electron paths with all pixels empty of charge.

(b) Electron paths with central pixel containing 100K electrons.

Figure 10: Impact of pixel charge on electron paths. Diffusion is turned off for this simulation. Electron paths are color coded depending on within which undistorted pixel they were created at the top of the diagram (CCD Backside).

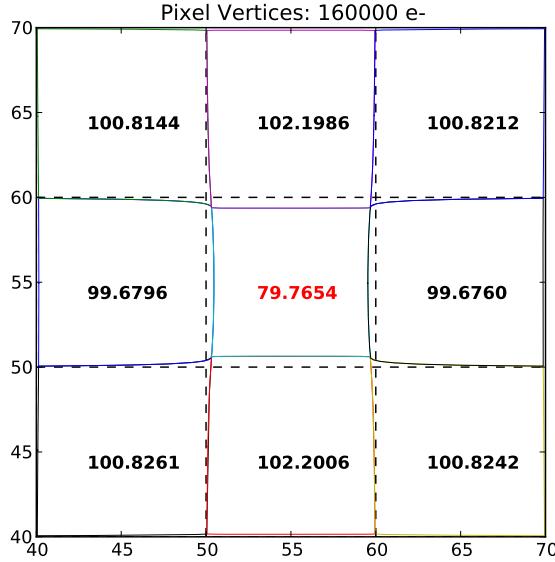


Figure 11: Distortion of pixel boundaries and areas due to 160,000 electrons in the central pixel. Undistorted pixel areas are 100 square microns. Each pixel polygon in this diagram contains 260 vertices. The vertices are found through binary search as described in the text. Diffusion is turned off to make this map.

### 3.0.3 Simulating the B-F Effect and Comparisons of Measurements with Simulations

We next discuss applying the simulation machinery discussed in the last sections to the B-F effect. The strategy for running these simulations is as follows:

1. Solve Poisson's equation for a postage stamp (using 9x9 pixels) with all pixels empty.
2. Choose a random location within the central pixel as the center of a 2D Gaussian spot. To match the measured 30 micron spots discussed in Section 2, the Gaussian spot has  $\sigma \approx 10$  microns.
3. Determine starting locations for N electrons in the 2D Gaussian spot.
4. Propagate these electrons down to their collecting gates.
5. Re-solve Poisson's equation with these wells now containing the appropriate charge. We have explored two different stored charge distributions:
  - (a) Charge distributed uniformly within the collecting region.
  - (b) Charge placed self-consistently as the stored charge builds up. This should be more accurate.
6. Repeat with N more electrons. Perfect accuracy would demand N = 1, but the electric field changes only slightly with < 1000 electrons / pixel. We have been using N=10,000 which places about 1000 electrons in the central pixel, so about 100 iterations are needed to fill the central pixel.
7. Repeat for more than one spot (typical  $\approx 100$ ), each with a different central location. Multiple spots can all be simulated in parallel.
8. Use the resulting array of spots as input to the same forward modeling routine used to analyze the measured data (see Section 2).

We can track the buildup of charge in the collecting well as the simulation progresses. Figure 12 shows this progression, and also a comparison of the self-consistent charge placement with an assumed uniform charge distribution.

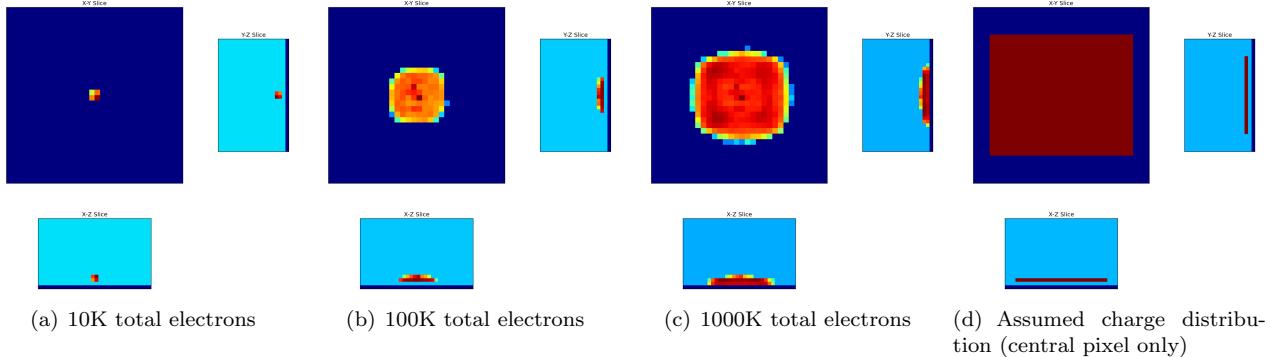


Figure 12: Self-consistent build-up of charge during simulation of Gaussian spot ((a) - (c)), as compared to an assumed charge distribution (central pixel only). Charge initially accumulates in the center of the well where the potential is highest, then spreads as more charge builds up since the accumulating negative charge modifies the potential in the potential well.

Following the procedure outlined above with no additional adjustable parameters results in a simulated plot of the spot size as a function of flux. Typically we run 64-256 spots in parallel with dithered central locations to mimic the measured data. These plots can then be compared to the measured results, and this has been done for a range of back-bias voltages, as shown in Figure 13. The agreement is quite satisfactory.

The measurements of pixel-pixel correlations, as shown in Figure 4, can also be compared to simulation. To do this, one compares a measurement of correlations where each pixel has an average of  $N$  electrons, to the simulated area change due to a central pixel with  $N$  electrons while the surrounding pixels are empty (as in Figure 11). For reasons explained in Appendix B, these should give the same results. This comparison is shown in Figure 14. The agreement is reasonable, and it is believed that the agreement can be improved through adjustment of the model free parameters, which are due to incomplete knowledge of the doping profiles in the CCD. The plan for doing this is discussed in the next Section.

### 3.0.4 Tuning of Model Free Parameters

Any simulation model necessarily includes free parameters which need to be adjusted. We have endeavored to make the model we are simulating in this work as physics-based as possible, which minimizes the free parameters. The electron transport for example, is based on well-known silicon parameters which have been well measured, and thus has no free parameters. Ideally the solution of Poisson's equation would also have no free parameters, but in practice there are properties of the CCD device that are not known exactly. For the device we are simulating, the horizontal geometries are well known, but the charges present in the channel and channel stop regions are known only within about a factor of two. These then are the main free parameters; the range of these parameters could be reduced with more detailed knowledge of the doping profiles.

1. The total charge, depth, and profile in the channel region.
2. The total charge, depth, and profile in the channel stop region.

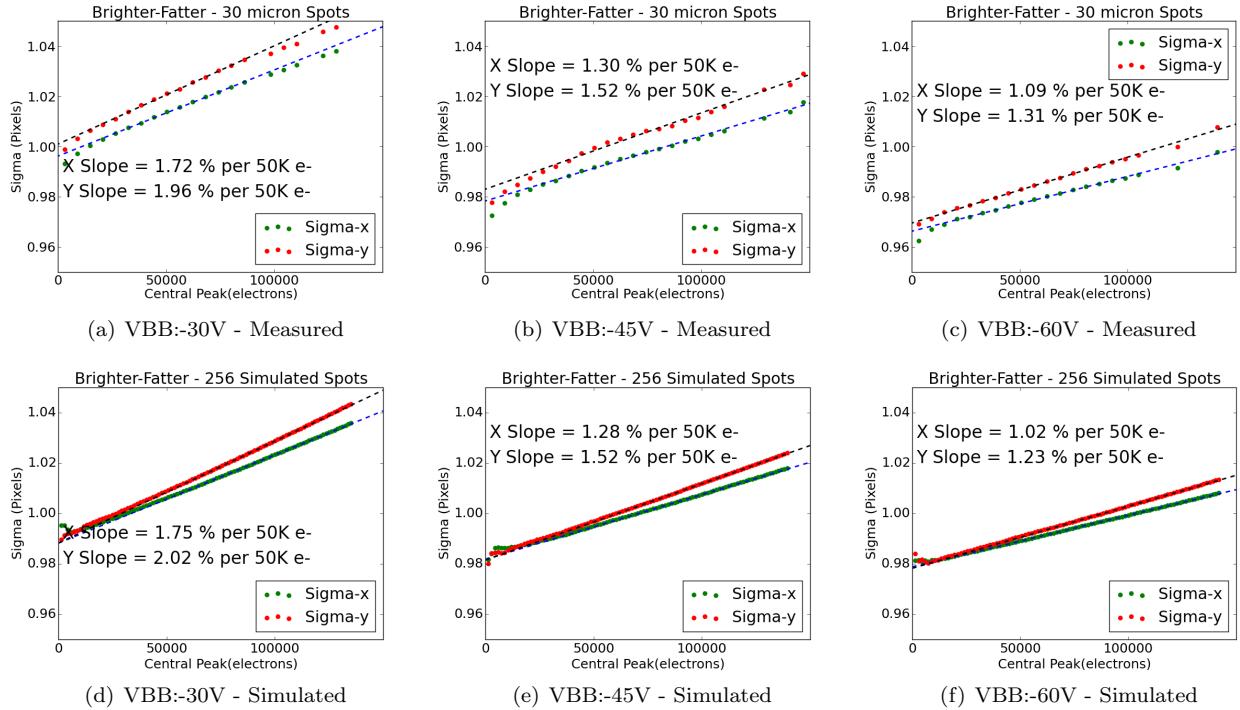


Figure 13: B-F slopes at different back-biases: Measurements and Simulations. The simulations are correctly capturing the B-F slopes as a function of back-bias, as well as capturing the X-Y asymmetry. This asymmetry is due to the fact that the confining electric field due to the channel stop implant is stronger than that due to the parallel gate potentials.

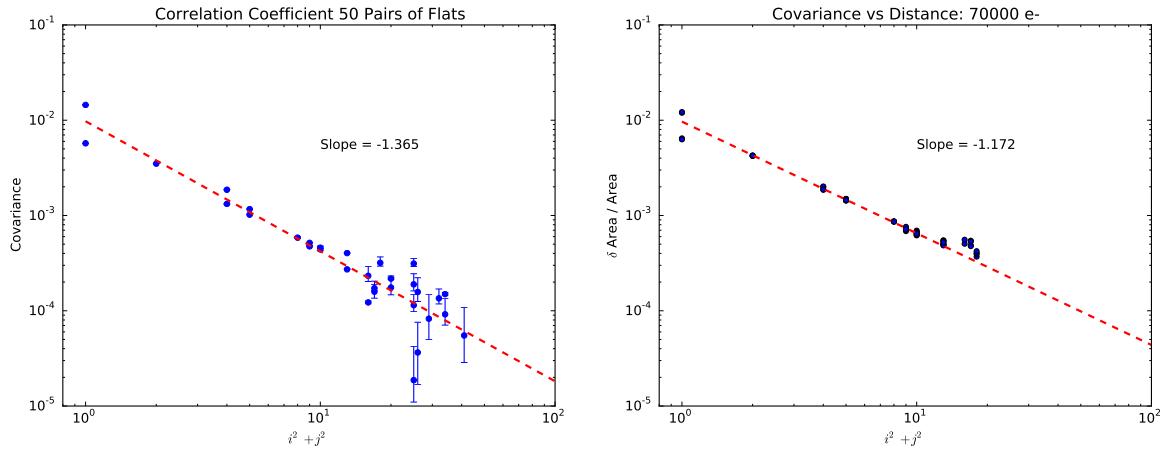


Figure 14: Measured pixel-pixel correlations and pixel area change as a function of radial distance.

A preliminary investigation of their impact has been done. Figure 15 shows the impact of moving the stored charge location deeper into the silicon. This has the effect of increasing the effective dipole moment of the dipole created by the stored charge and its image, which is located an equal distance below the equipotential defined by the parallel gates. Increasing this dipole moment increases the magnitude of the B-F effect and the corresponding slopes of the curves in Figure 15. These simulations were done assuming a certain location of the charge stored in the pixel. Using the method of calculating this charge location self-consistently, as we will do in the future, we expect the (negative) charge to be stored at the potential maximum of the buried channel. Thus, as we move the channel implant deeper, we expect the B-F slopes to increase.

We have also characterized the impact of the channel stop strength on the pixel area changes and corresponding pixel-pixel correlations. This effect is shown in Figure 16, where it is seen that making the channel stop stronger and deeper results in a smaller impact of the collected charge on the area of the pixels in the direction of the channel stop. In fact, it is possible to completely change the sign of the area change of the (1,0) pixel.

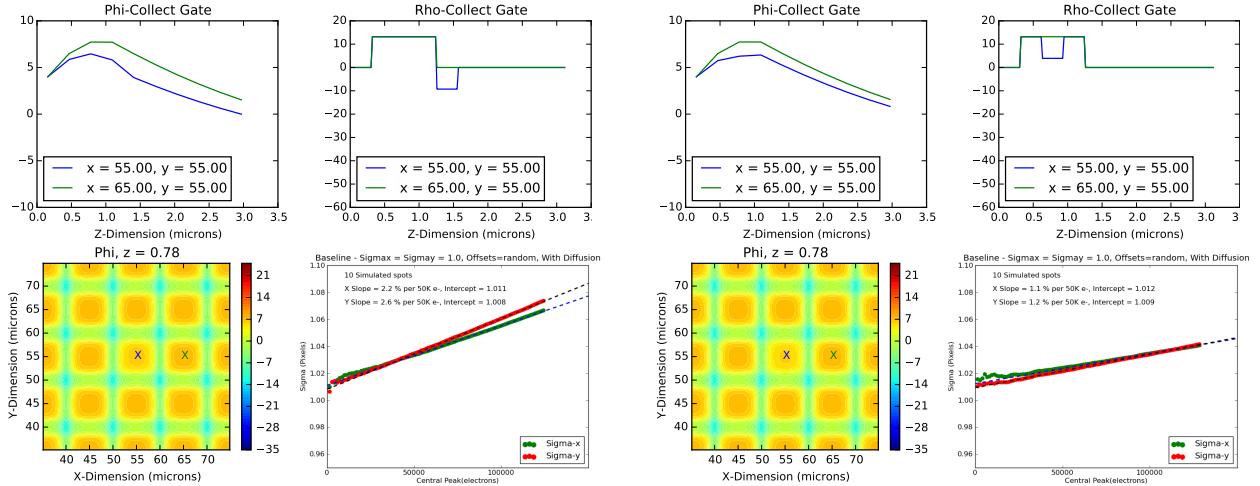
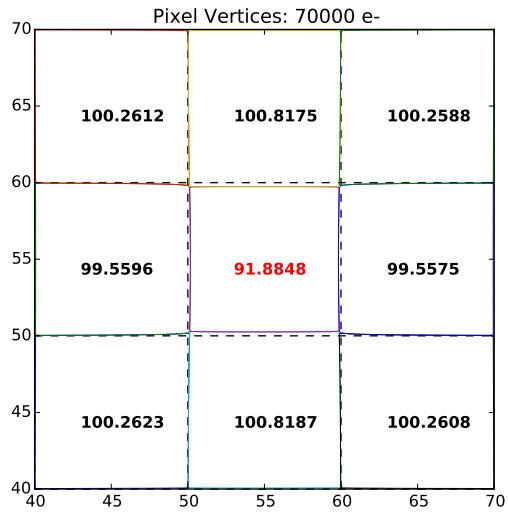
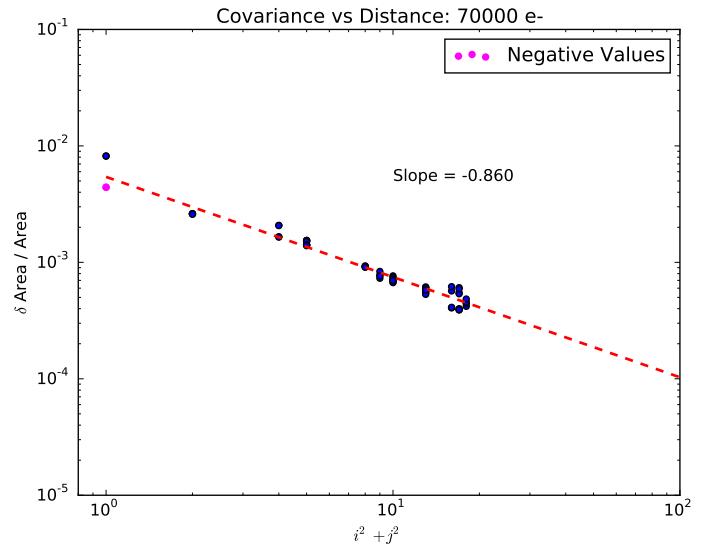


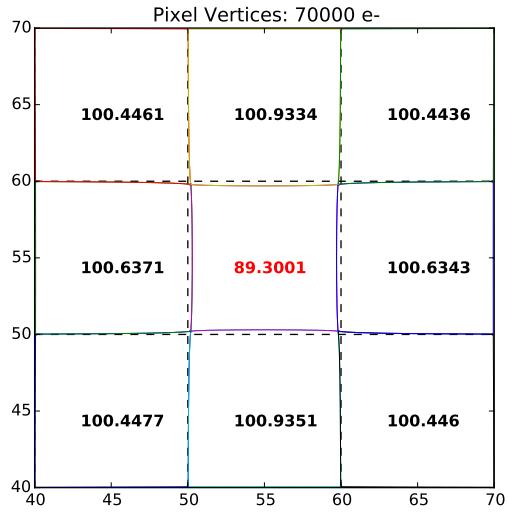
Figure 15: Impact of stored charge location on B-F slopes. As the charge is stored further from the Si/SiO<sub>2</sub> interface, the magnitude of the B-F effect increases.



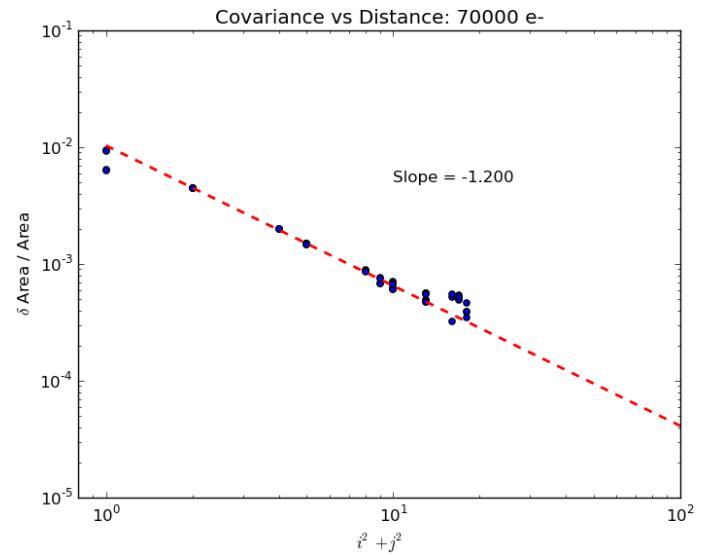
(a) Channel Stop charge of 4E12, 2 microns deep



(b) Channel Stop charge of 4E12, 2 microns deep



(c) Channel Stop charge of 1E12, 1 micron deep



(d) Channel Stop charge of 1E12, 1 micron deep

Figure 16: Impact of channel stop charge and depth on pixel areas and inferred pixel-pixel correlations. In addition to the change in slope, note that the covariance of the (1,0) pixel actually changes sign.

We are in the process of doing a more detailed simulation study to enable us to “tune in” these free parameters based on measurements of the B-F slopes and pixel-pixel correlations. Prior to doing this, we have made the model more physical by transitioning from square doping profiles and assumed stored charge distributions to Gaussian doping profiles and self-consistent stored charge distributions. The comparison between these two is shown in Figure 17.

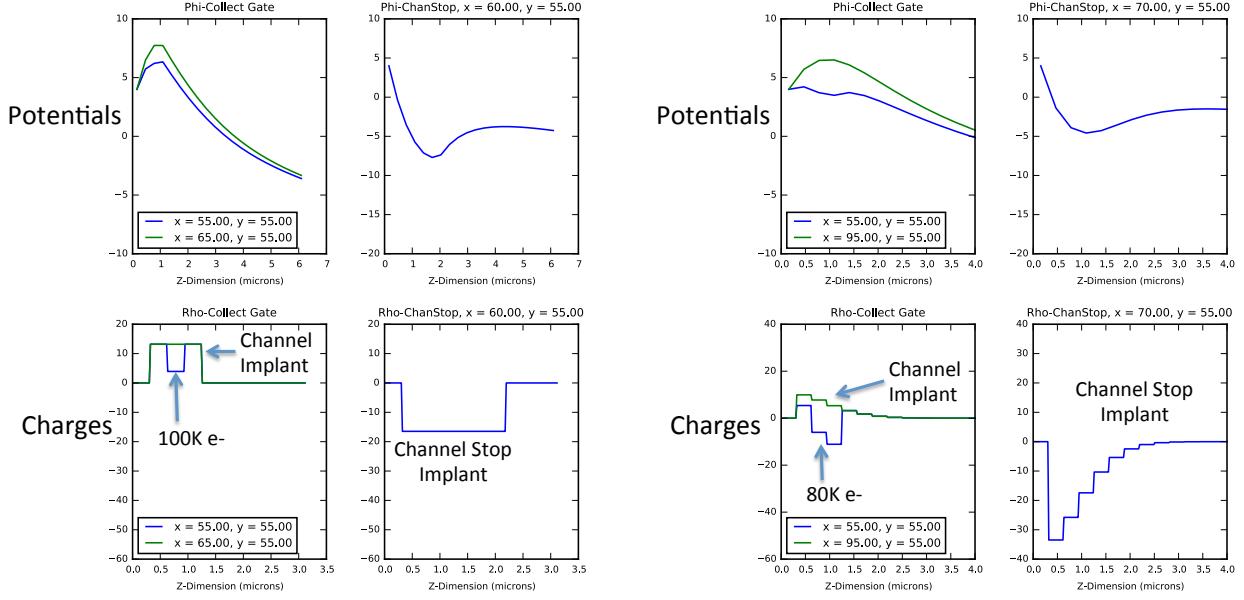


Figure 17: Prior to running a series of simulations to enable us to tune in the model free parameters, we have made the model more physical by transitioning from square doping profiles and assumed stored charge distributions (left image) to Gaussian doping profiles and self-consistent stored charge distributions (right image).

To determine the best fit values for the free parameters, approximately 50 simulations were run with varying values. These included both values randomly chosen from within the parameter space and targeted variations of a single parameter. After running these simulations, a four variable second order fit was done to the results of the simulations to determine a response surface for each of five measured parameters. The results of this analysis are shown in Figure 18. While the channel stop parameters are nicely aligned between the direct B-F measurements and the correlations measurements, the channel parameters show a discrepancy between what is required to fit the direct B-F measurements and what is required to fit the correlation measurements. The cause of the discrepancy is being investigated. Possibilities being considered include an error or missing physics in the model, an error in the measurements, and that the Gaussian profile assumption is not a good assumption.

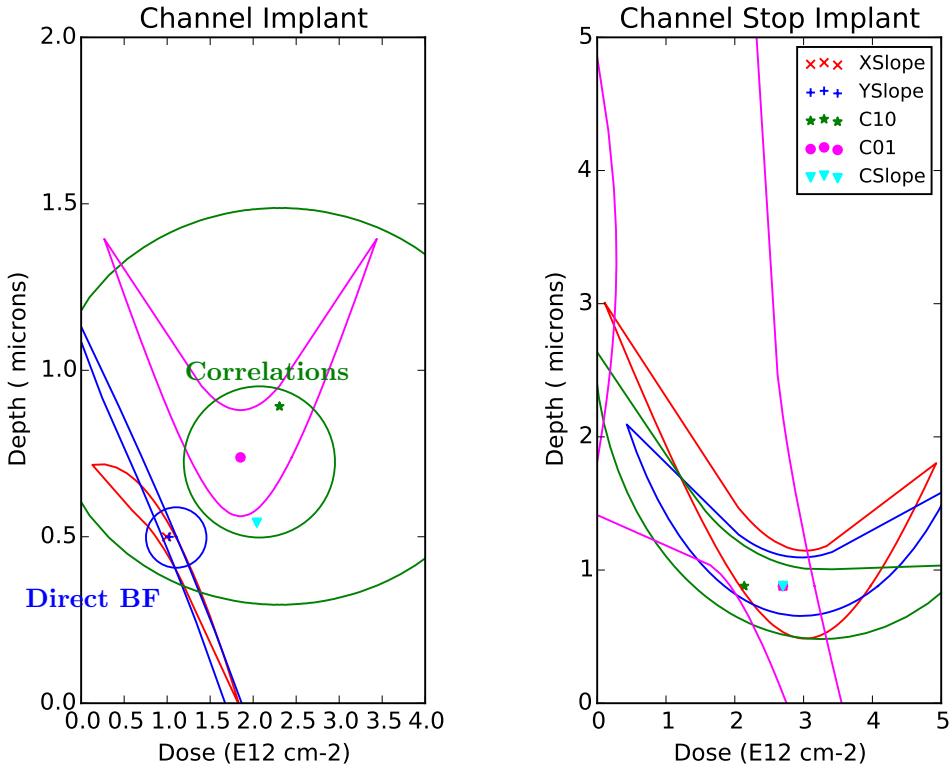


Figure 18: Results of a response surface analysis based on approximately 50 simulations with varying channel and channel stop parameters. The contours represent  $\pm 10\%$  variations from the measured values. While the channel stop parameters are nicely aligned between the direct B-F measurements and the correlations measurements, the channel parameters show a discrepancy between what is required to fit the direct B-F measurements and what is required to fit the correlation measurements. The cause of the discrepancy is being investigated.

### 3.1 Simulations using PhoSim

The physics-based model described in the preceding sections has been developed with an eye toward making the model as physically accurate as possible at the expense of simulation speed. However, there are applications in the LSST project where a much faster simulator is desired. The PhoSim simulator (Peterson, et.al. [7], [8]) is a detailed simulator of the entire LSST instrument that operates on a photon-photon basis. The Poisson solver code described here is far too slow to be incorporated directly into PhoSim, so a way to accurately incorporate the results of this code into PhoSim was needed. After examining several options, it was decided that capturing the pixel vertex displacements as a function of stored charge electron creation depth would serve. Figures 19 and 20 show the changes in pixel area and displacement of pixel vertices as a function of charge in the central pixel. Figure 21 shows the displacement of pixel vertices as a function of depth in the silicon at which the electron is created. The fact that the pixel vertex displacement (Figure 20) is a linear function of pixel charge is an encouraging sign that it is possible to superpose the pixel vertex displacements which are due to charges in different pixels. To test this hypothesis, two tests of this superposition plan have been done, and the results are shown in Figure 22. These plots show that this superposition very accurately describes the distortion of the pixel shapes. Figure 21 shows that it is possible to write an analytic function that captures the change in pixel shape with depth of electron creation.

Building on this success, this strategy was implemented into the PhoSim code. A single physics-based simulation

Code	Polygon Vertices	Pixel Area Error (%)	Photons/second
Existing	—	—	180,000
New	12	0.51	168,000
New	20	0.23	152,000
New	36	0.11	130,000
New	260	$\approx 0$	42,000

Table 1: Performance of the pixel distortion algorithm as compared to existing PhoSim code. Figure 23 is run with 36 vertices. This could possibly be reduced.

is run with a large quantity of charge in one pixel and the surrounding pixels empty. Typically 160,000 electrons is used for this; a large charge is used to maximize the pixel distortions, but it is desired to stay below saturation. This simulation takes on the order of one hour, but only needs to be re-run in the event of a change in CCD conditions. When PhoSim starts up, this table of pixel displacements is read into an array. As PhoSim runs and an electron is created at some location, the shape of the pixels near where the electron is created are calculated based on the table of pixel displacements and the charge stored in the surrounding pixels. This allows us to calculate in which pixel the electron will end up, and the electron is than placed at a random location within the collecting region of that pixel. This procedure is quite fast, and the implementation in PhoSim resulted in little to no slow down of the PhoSim code. Table 3.1 quantifies the performance/pixel shape accuracy trade-off. More accurate characterization of the distorted pixel shape of course requires more computer time.

Figure 23 shows the results. Since PhoSim generates FITS files directly, this plot is generated by inputting the FITS files from PhoSim directly in to the forward modeling routine described in Section 2.1. One can see that this procedure accurately simulates the B-F effect, and the B-F slopes are quite close to what is found with the full physics-based simulation (Figure 13(f)). More work needs to be done to complete the PhoSim implementation, but it is clear that this strategy will result in accurate simulations at acceptable speed.

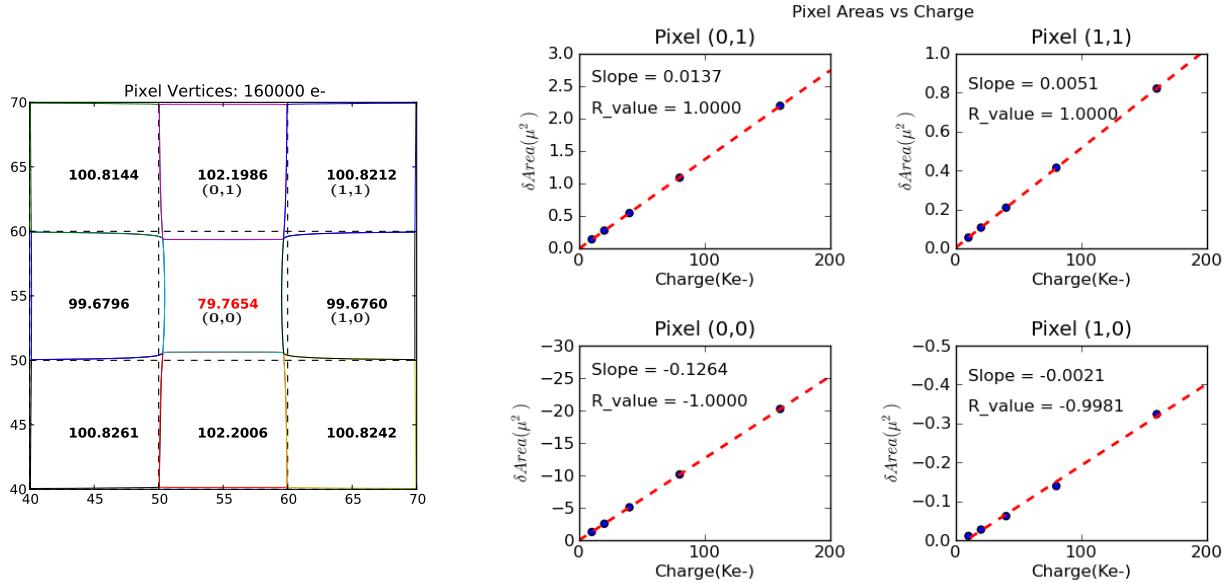


Figure 19: Characterization of pixel area trends as a function of central pixel charge. Note that they are quite linear.

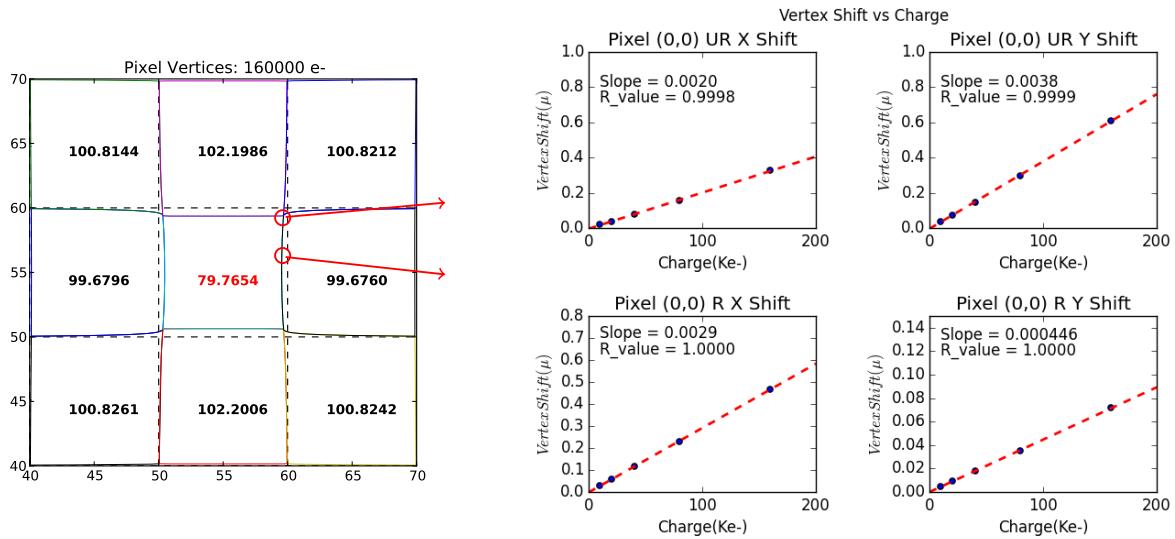


Figure 20: Characterization of pixel vertex displacement as a function of central pixel charge. These are also quite linear.

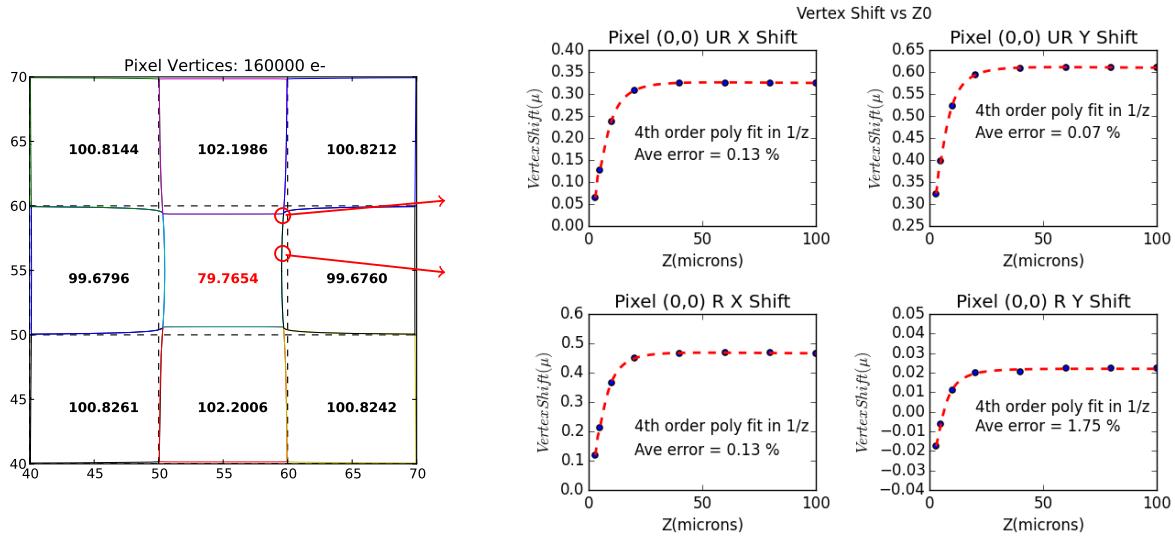
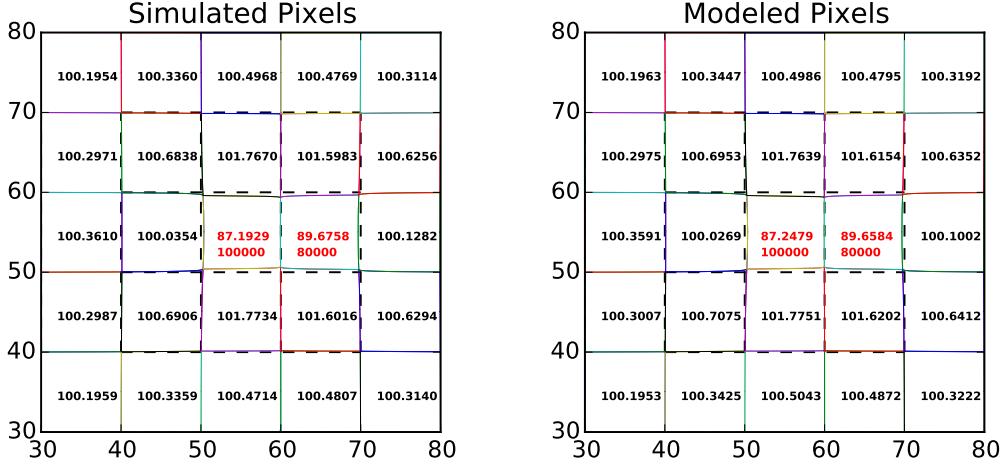
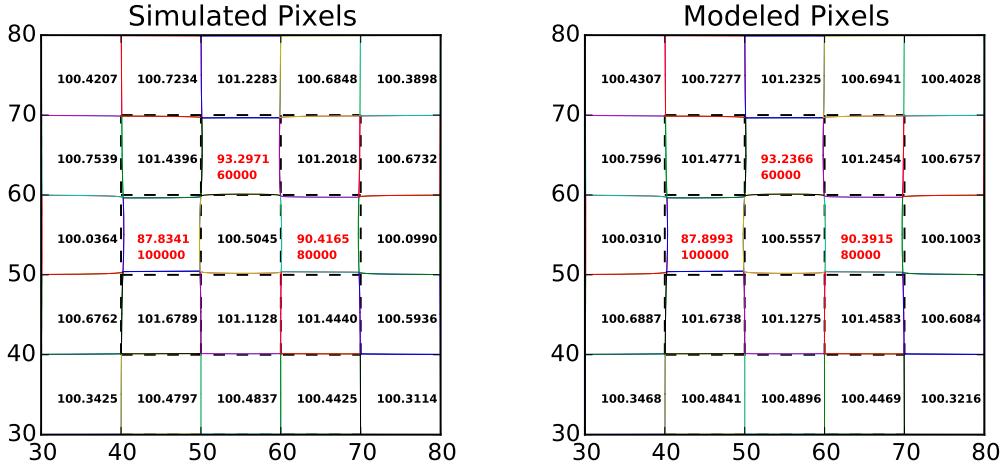


Figure 21: Characterization of pixel vertex displacement as a function of initial electron depth.



Average Area Error = 0.0139 percent  
 Average Vertex Error = 0.0003 microns  
 Worst Case Area Error = 0.0551 percent  
 Worst Case Vertex Error = 0.0669 microns

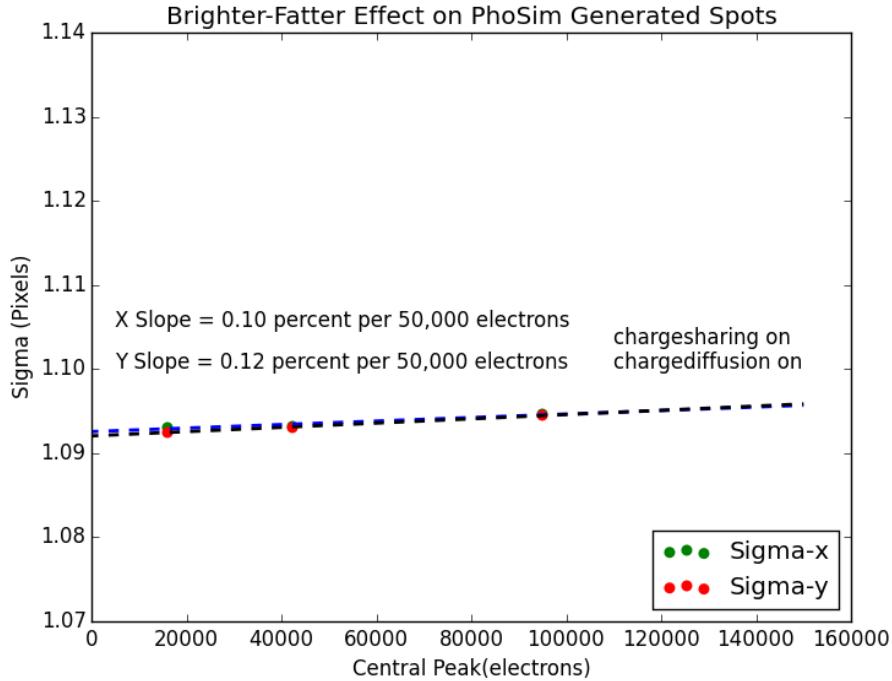
(a) Superposition Test 1



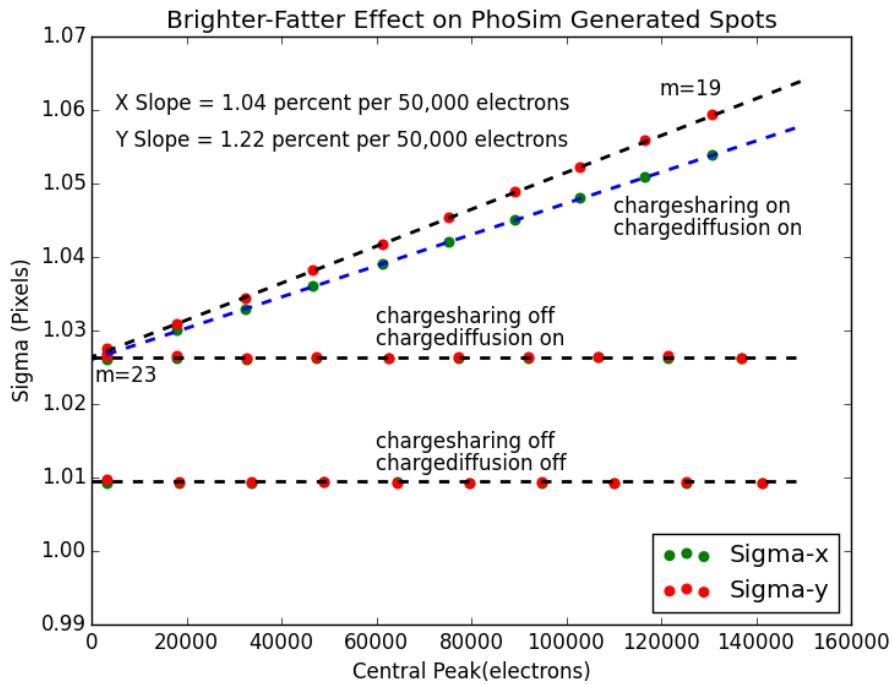
Average Area Error = 0.0242 percent  
 Average Vertex Error = 0.0003 microns  
 Worst Case Area Error = 0.0682 percent  
 Worst Case Vertex Error = 0.0638 microns

(b) Superposition Test 2

Figure 22: Superposition tests. In both cases the left-hand plot is a full physics-based simulation, solving Poisson's equation and finding pixel vertices through binary search, while the right-hand plot takes pixel vertex displacement from a single physics-based run with one pixel containing 80K e-, and superposes the displacements for the pixels containing charge. We expect the displacements to add linearly as a consequence of the linearity of Maxwell's equations, and this confirms that intuition.



(a) Current released version of PhoSim



(b) Modified version of PhoSim with modifications as described in the text.

Figure 23: B-F effect as simulated using PhoSim to generate a FITS file with an array of 400 dithered spots, which is then analyzed with the forward modeling routine as described in Appendix A. On the left are the results with the current released version of PhoSim, and on the right are the results after PhoSim has been modified as described in the text. Note that the B-F slopes in the right hand plot simulated using PhoSim are virtually identical to those simulated with the full physics-based Poisson solver, as seen by comparing this plot with Figure 13 (f).

## 4 Plans for Future Work

As can be seen, more work is needed, both measurements and simulations, to achieve the goal of a verified physical model of the CCD. Figure 24 shows a proposed schedule for completing this work. Of particular importance are the planned measurements involving injecting a sky background with an intensity separately adjustable from the intensity of the spots. This is shown in Line 18 of Figure 24.

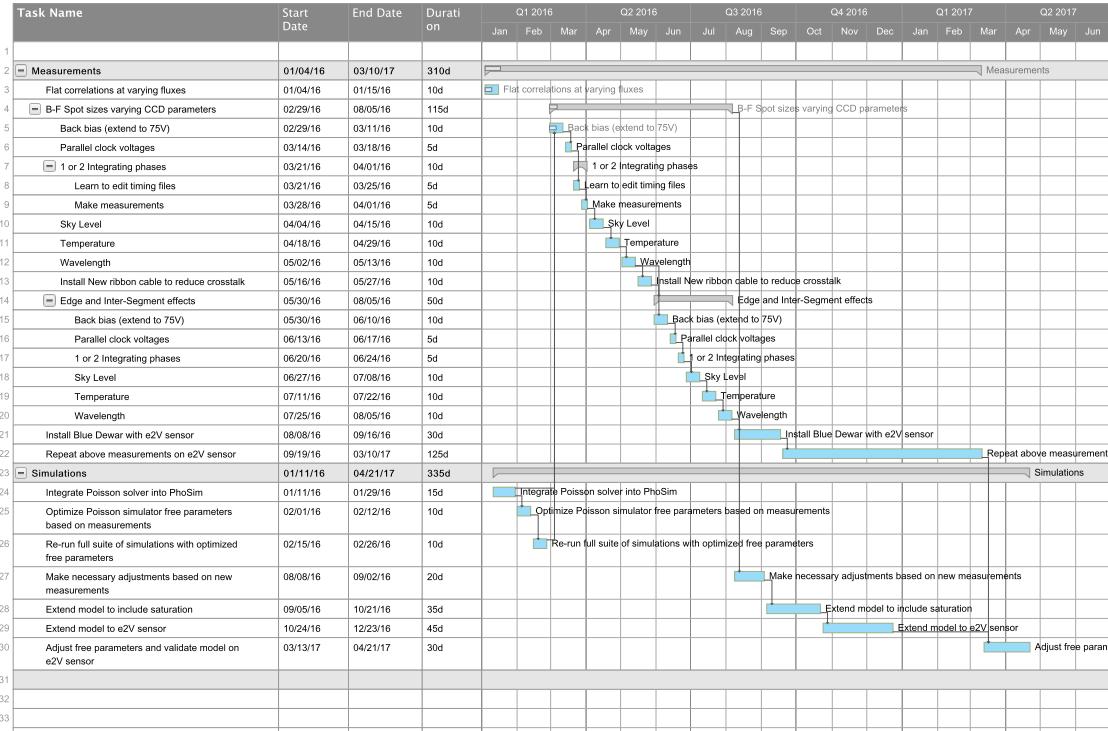


Figure 24: Schedule for continued measurement and simulation work at UC Davis

## 5 Conclusions and Next Steps

In this work we have described in detail a physics-based simulation of the CCD which accurately models the B-F effect, as determined by both direct measurements of spot sizes and by measurements of pixel-pixel correlations in CCD flats. The model is capable of simulating the B-F effect to the accuracy required for determining the PSF in LSST images and suppressing the systematic impact of this effect on measurements of galaxy shapes to the level required by the weak lensing science.

Preliminary work has been done on characterizing the response of the model to the model free parameters, primarily the strength and depth of the implants in the channel region and the channel stop region. However, more work, both measurements and simulations, is need to effectively “tune in” these free parameters to have a fully data-verified model of the CCD. This work will be completed in 2016.

## A Appendix - Details of Forward Modeling of Spot Images

This appendix describes the techniques that are used to generate a forward model of a set of spot images. We start with a number  $N$  of pixelized images, each  $N_x \times N_y$  pixels in size. These images are referred to as “postage stamps”. We want to simultaneously fit all of these images with a single function, typically a 2D Gaussian. We

assume that this function has the same shape for all of the postage stamps, but we allow the magnitude of the function to vary from stamp to stamp, and we also allow an offset of the origin of the fitting function which varies from stamp to stamp. Let  $M_{nij}$  be the measured data of each postage stamp, where  $n$  is an index that identifies the stamp, and  $i$  and  $j$  are indices which identify the X-Y location within the stamp. Let  $f_{nij}$  be the normalized fitting function. The form of the fitting function does not matter for this analysis, but we next illustrate a typical fitting function. This fitting function for a 2D Gaussian is of the following form, with  $p$  the pixel size, and  $x_{\text{off}_n}$  and  $y_{\text{off}_n}$  the offsets of the Gaussian function origin from the pixel center.

$$f_{nij} = I = \frac{1}{4}(\operatorname{erf}\left(\frac{xh}{\sqrt{2}\sigma_x}\right) - \operatorname{erf}\left(\frac{xl}{\sqrt{2}\sigma_x}\right)) * (\operatorname{erf}\left(\frac{yh}{\sqrt{2}\sigma_y}\right) - \operatorname{erf}\left(\frac{yl}{\sqrt{2}\sigma_y}\right)) \quad (\text{A.1})$$

where:

$$xh = (i + \frac{1}{2})p + x_{\text{off}_n}; xl = (i - \frac{1}{2})p + x_{\text{off}_n}; yh = (j + \frac{1}{2})p + y_{\text{off}_n}; yl = (j - \frac{1}{2})p + y_{\text{off}_n} \quad (\text{A.2})$$

The offset from the pixel center are determined by taking the first moment of the postage stamp image to find the image centroid. Since we allow each stamp to have a different intensity, the function that we want to minimize to find the best fit is as follows:

$$F = \sum_n^N \sum_i^{N_x} \sum_j^{N_y} (M_{nij} - I_n f_{nij})^2 \quad (\text{A.3})$$

Expanding:

$$F = \sum_n^N \sum_i^{N_x} \sum_j^{N_y} (M_{nij}^2 - 2M_{nij}I_n f_{nij} + I_n^2 f_{nij}^2) \quad (\text{A.4})$$

Since we are minimizing this function, we can take the partial derivatives with respect to the  $I_n$  and set them equal to zero. This gives  $N$  equations, each of which can be solved for one of the coefficients  $I_n$ , giving:

$$I_n = \frac{\sum_i^{N_x} \sum_j^{N_y} M_{nij} f_{nij}}{\sum_i^{N_x} \sum_j^{N_y} f_{nij}^2} \quad (\text{A.5})$$

We introduce the following shorthand notations:

$$a_{2n} = \sum_i^{N_x} \sum_j^{N_y} M_{nij}^2 \quad (\text{A.6})$$

$$b_{2n} = \sum_i^{N_x} \sum_j^{N_y} f_{nij}^2 \quad (\text{A.7})$$

$$ab_n = \sum_i^{N_x} \sum_j^{N_y} M_{nij} f_{nij} \quad (\text{A.8})$$

So we can then write:

$$I_n = \frac{ab_n}{a_{2n}} \quad (\text{A.9})$$

We can now substitute the result for  $I_n$  back in to the expression for  $F$ , giving:

$$F = \sum_n^N \left( \sum_i^{N_x} \sum_j^{N_y} M_{nij}^2 - 2 \frac{\sum_i^{N_x} \sum_j^{N_y} M_{nij} f_{nij}}{\sum_i^{N_x} \sum_j^{N_y} f_{nij}^2} \sum_i^{N_x} \sum_j^{N_y} M_{nij} f_{nij} + \left( \frac{\sum_i^{N_x} \sum_j^{N_y} M_{nij} f_{nij}}{\sum_i^{N_x} \sum_j^{N_y} f_{nij}^2} \right)^2 \sum_i^{N_x} \sum_j^{N_y} f_{nij}^2 \right) \quad (\text{A.10})$$

Which, using the shorthand notation, can be expressed as:

$$F = \sum_n^N \left( a_{2n} - 2 \frac{ab_n}{b_{2n}} ab_n + \left( \frac{ab_n}{b_{2n}} \right)^2 b_{2n} \right) = \sum_n^N \left( a_{2n} - \frac{ab_n^2}{b_{2n}} \right) \quad (\text{A.11})$$

This is the form that is implemented in the forward modeling code. A Python extension implemented in C++ has been written to perform this calculation as rapidly as possible. It takes as inputs a list of N postage stamps and values of  $\sigma_x$  and  $\sigma_y$ , and returns the intensity for each stamp  $I_n$ , as well as the result F, which is the sum-of-squares fit between the measured postage stamps and the fitting function. The value of F is then used in a steepest descent algorithm (Powell's method) to find the values of  $\sigma_x$  and  $\sigma_y$  which give the best fit to the measured data.

## B Appendix - Relation of Pixel-Pixel Correlations to Pixel Area Change

The purpose of this appendix is to clarify the relationship between the pixel-pixel correlations and the pixel area changes caused by the lateral electric fields which distort the pixel boundaries. Referring to Figure 14, we see that the two are definitely related, but it is not immediately obvious whether the covariance of a given pixel is equal to the fractional change in area, proportional to it, or whether they are related in some more complex way. To clarify this, this appendix discusses a simple toy model which has been both analyzed analytically and simulated numerically.

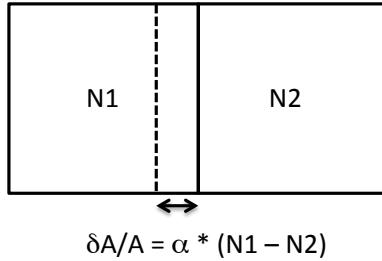


Figure 25: Toy Model of the B-F effect with two pixels with a movable partition between them.

The basics of the toy model are shown in Figure B. There are two initially equal pixels with a movable partition between them. Each pixel has a number of stored electrons ( $N_1, N_2$ ), and the partition is assumed to move by an amount proportional to the difference in stored charge between the two pixels, as shown in the figure. We assume that the collection of charge in the two pixels is a Poisson process with rate parameter  $\lambda$ , so that the probability of collecting N electrons is:

$$P(N; \lambda) = \frac{\lambda^N e^{-\lambda}}{N!} \quad (B.1)$$

As is well known, a Poisson process can be decomposed into two subprocesses as follows:

$$P(N_1, N_2; \lambda_1, \lambda_2) = \frac{\lambda_1^{N_1} e^{\lambda_1}}{N_1!} \frac{\lambda_2^{N_2} e^{\lambda_2}}{N_2!} \quad (B.2)$$

In this case, because the rate of collecting electrons is proportional to the pixel area, the two rate parameters ( $\lambda_1, \lambda_2$ ) are related to the global parameter  $\lambda_0$  as follows:

$$\lambda_1 = \lambda_0(1 - \frac{\alpha}{2}(N_1 - N_2)); \lambda_2 = \lambda_0(1 + \frac{\alpha}{2}(N_1 - N_2)); \quad (B.3)$$

The factor of 2 in the denominator is inserted because the initial areas are equal, and the final areas are determined by the final counts ( $N_1, N_2$ ). Integrating from an initial value of  $\delta A = 0$  to the final value  $\delta A = A\alpha(N_1 - N_2)$  gives this factor of two. Now we insert  $\lambda_1, \lambda_2$  into P and expand, keeping terms to first order in  $\alpha$ :

$$P(N_1, N_2; \lambda_0, \alpha) = P(N_1; \lambda_0)P(N_2; \lambda_0)(1 - \frac{\alpha}{2}(N_1 - N_2)^2) \quad (B.4)$$

The means, variances, and covariance are then (to first order in  $\alpha$ ):

$$\text{Mean}(N_1) = \sum_{N_1=0}^{\infty} \sum_{N_2=0}^{\infty} N_1 P(N_1, N_2; \lambda_0, \alpha) = \lambda_0 (1 - \frac{\alpha}{2}) \quad (\text{B.5})$$

$$\text{Var}(N_1) = \sum_{N_1=0}^{\infty} \sum_{N_2=0}^{\infty} (N_1 - \text{Mean}(N_1))^2 P(N_1, N_2; \lambda_0, \alpha) = \lambda_0 (1 - \frac{\alpha}{2}(\lambda_0 + \frac{1}{2})) \quad (\text{B.6})$$

$$\text{Correlation}(N_1, N_2) = \sum_{N_1=0}^{\infty} \sum_{N_2=0}^{\infty} (N_1 - \text{Mean}(N_1))(N_2 - \text{Mean}(N_2)) P(N_1, N_2; \lambda_0, \alpha) = \lambda_0^2 \alpha \quad (\text{B.7})$$

$$\text{Covariance}(N_1, N_2) = \frac{\text{Correlation}(N_1, N_2)}{\text{Var}(N_1)} = \lambda_0 \alpha \quad (\text{B.8})$$

When calculating the changes in area as in Figure 14, we set  $N_1 = N_0, N_2 = 0$ , so that  $\frac{\delta A}{A} = \alpha N_0 = \alpha \lambda_0$ . So we find that the fractional change in area of a pixel and the covariance of that pixel are in fact equal. As a further check, we have built a Monte-Carlo version of the toy model, simply “throwing darts” at the two pixels using a pseudo-random number generator, and moving the central partition after each “dart”. The results of running the Monte-Carlo code for 20 different values of  $\alpha$  and two different values of  $\lambda_0$  are show in Figure 26, and confirm Equation B.8.

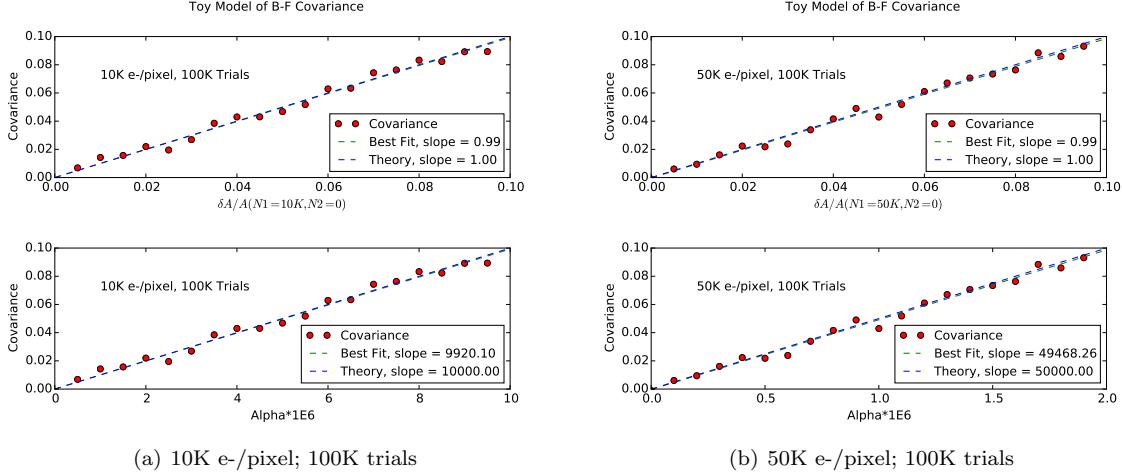


Figure 26: Monte-Carlo simulations of the toy model.

## C Appendix - Multi-Grid Solution of Poisson’s Equation

There are several methods in use for solving Poisson’s equation numerically. The most important are the use of the Fast Fourier Transform (FFT) and the use of Multi-Grid Methods, which is the method we have chosen in this work. There is a wealth of literature describing the use of Multi-Grid methods for solving PDE’s; a good reference is Press, et.al. [13]. First, one discretizes Poisson’s equation on a 3-dimensional grid, as follows. Here  $\varphi$  is the electrostatic potential, and  $\rho$  is the charge density.

$$\nabla^2 \varphi = \rho \quad (\text{C.1})$$

$$\frac{\partial^2 \varphi_{i,j,k}}{\partial x^2} \approx \frac{(\varphi_{i+1,j,k} - \varphi_{i,j,k}) - (\varphi_{i,j,k} - \varphi_{i-1,j,k})}{h^2} \quad (\text{C.2})$$

$$(\varphi_{i+1,j,k} + \varphi_{i-1,j,k} + \varphi_{i,j+1,k} + \varphi_{i,j-1,k} + \varphi_{i,j,k+1} + \varphi_{i,j,k-1} - 6 * \varphi_{i,j,k}) = h^2 * \rho_{i,j,k} \quad (C.3)$$

$$\varphi_{i,j,k} = \frac{1}{6} * (\varphi_{i+1,j,k} + \varphi_{i-1,j,k} + \varphi_{i,j+1,k} + \varphi_{i,j-1,k} + \varphi_{i,j,k+1} + \varphi_{i,j,k-1} - h^2 * \rho_{i,j,k}) \quad (C.4)$$

$$\varphi_{i,j,k}^{(n+1)} = \frac{1}{6} * (\varphi_{i+1,j,k}^{(n)} + \varphi_{i-1,j,k}^{(n)} + \varphi_{i,j+1,k}^{(n)} + \varphi_{i,j-1,k}^{(n)} + \varphi_{i,j,k+1}^{(n)} + \varphi_{i,j,k-1}^{(n)} - h^2 * \rho_{i,j,k}) \quad (C.5)$$

$$\varphi_{i,j,k}^{(n+1)} = (1 - \omega) * \varphi_{i,j,k}^{(n)} + \frac{\omega}{6} * (\varphi_{i+1,j,k}^{(n)} + \varphi_{i-1,j,k}^{(n)} + \varphi_{i,j+1,k}^{(n)} + \varphi_{i,j-1,k}^{(n)} + \varphi_{i,j,k+1}^{(n)} + \varphi_{i,j,k-1}^{(n)} - h^2 * \rho_{i,j,k}) \quad (C.6)$$

Equation C.5 gives a relation for the potential at location  $(i,j,k)$  in terms of the charge density at  $(i,j,k)$  and the potential at the surrounding points. This can be solved iteratively, and the usual method is through the use of Successive Over-Relaxation (SOR), as described in Equation C.6. It can be shown that this process will converge provided that the parameter  $\omega < 2$ . In practice however, it is found that real problems require a very large number of iterations in order to converge. This is because errors in the grid are only propagated to nearest neighbors, so that long wavelength errors that span the entire grid require many iterations to be corrected. The solution is to use Multi-Grid methods, as described schematically in Figure 27.

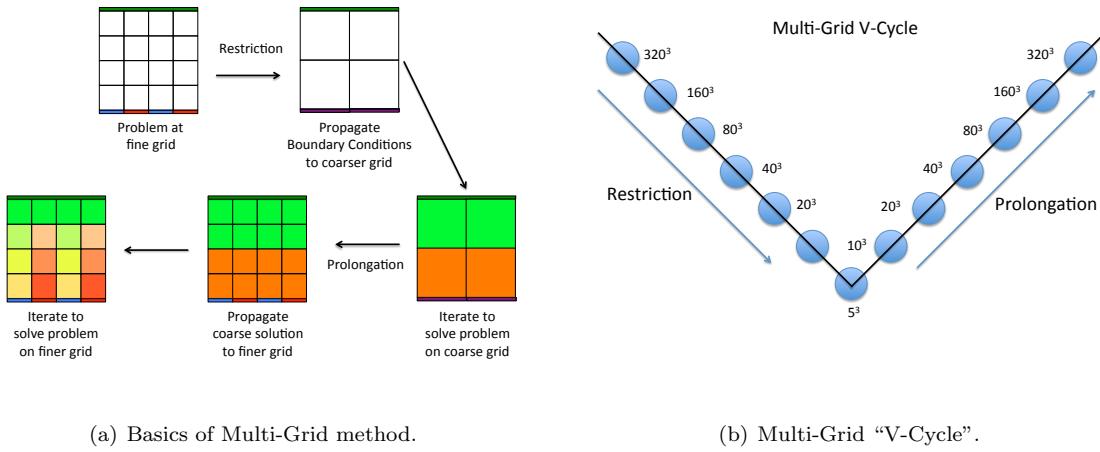


Figure 27: Summary of Multi-Grid method as used to solve Poisson’s equation in this work.

In the application of these methods to the problem at hand, one starts with a grid at the desired final resolution, and defines the boundary conditions, which include the potentials on the boundaries, the charges throughout the volume, and an initial guess at the potentials within the volume. One then propagates these conditions down to successively coarser grids through averaging, a process referred to in the literature as “restriction”, until one arrives at a grid with only a small number of grid cells (usually  $5 \times 5 \times 5$  cells in this work). Because of the small number of grid cells, this can be solved iteratively to machine precision in a very short time. One then propagates this solution up to the next finest grid by interpolation, a process referred to as “prolongation”. A relatively small number of iterations then suffices to solve the problem at this level of resolution, after which one repeats the process at the next finest grid until one arrives at the final (finest) resolution. This speeds up convergence of the problem dramatically, because long wavelength modes are determined at low resolution, and each successively finer level of resolution only requires enough iterations to converge at the shorter wavelengths which are introduced at this finer scale.

In this work, we typically use a single “V-Cycle” (see Figure 27 (b)). At each finer level of resolution, we use 2X fewer iterations, ending with 128 iterations at the finest grid. This suffices to find a solution of Poisson’s equation with better than 1 mV accuracy. In the code we have developed, all of these parameters, including the number of V-Cycles, the value of the SOR parameter  $\omega$ , and the number of iterations at the finest grid, can be controlled in the configuration file.

## D Appendix - Details of Electron Path Simulation

This section describes the details of the electron propagation model used in this work.

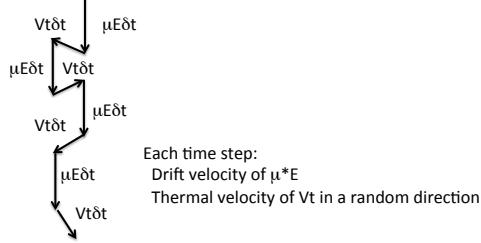


Figure 28: Basics of Electron propagation model

As is well known, charges in semiconductors under the influence of a electric field move at a constant velocity  $v = \mu E$ , where  $\mu$  is the mobility and  $E$  is the electric field. The mobility is dependent on temperature and electric field, and the mobility model used here is that due to Jacobini, et.al. [14], which is given in Equation D.1. For the CCDs used here, the mobility is typically on the order of  $6000 \frac{\text{cm}^2}{\text{V sec}}$ . The constant velocity is due to repeated scattering events which keep the charge from accelerating. The time between scattering events is given by Equation D.2, where  $m_e^*$  is the carrier effective mass,  $q_e$  is the carrier charge, and  $\mu$  is the mobility. For the conditions in the CCDs we are simulating, the scattering time is on the order of 1 picosecond.

$$\mu(E, T) = \frac{v_m}{E_c} \left(1 + \left(\frac{E}{E_c}\right)^\beta\right)^{-1/\beta}; E_c = 1.01 \text{ T}^{1.55} \frac{\text{V}}{\text{cm}}; v_m = 1.53 \times 10^9 \text{ T}^{-0.87} \frac{\text{cm}}{\text{sec}}; \beta = 0.0257 \text{ T}^{0.66} \quad (\text{D.1})$$

$$\tau_{\text{scatt}} = \frac{m_e^*}{q_e} \mu \quad (\text{D.2})$$

$$v_t = \sqrt{\frac{3kT}{m_e^*}} \quad (\text{D.3})$$

The basic electron transport model is shown in Figure 28. A time  $\delta t$  is drawn from an exponential distribution with an average of  $\tau_{\text{scatt}}$ . At each time step  $\delta t$ , the electron is assumed to move in the direction of the electric field by an amount  $\mu E \delta t$  and to move in a random direction by an amount  $v_t \delta t$ , where  $v_t$  is a thermal velocity given by Equation D.3. The process is repeated until the electron reaches the bottom. For the CCDs in this work, it typically requires about 1000 steps for the electron to propagate from the point of photocreation to the point of collection. The code developed here continues the electron propagation for some number of steps (typically 1000) after the electron has reached the collecting well, then takes the average of these last steps to determine the final location. The code allows one to turn the diffusion component off or reduce the amount of diffusion by an arbitrary multiplier. There is also an option to “coarse-grain” the electron transport by adding together a number (called NumDiffSteps) of steps. When this is done, the step due to the electric field is increased by NumDiffSteps, and the step due to diffusion is increased by  $\sqrt{\text{NumDiffSteps}}$ .

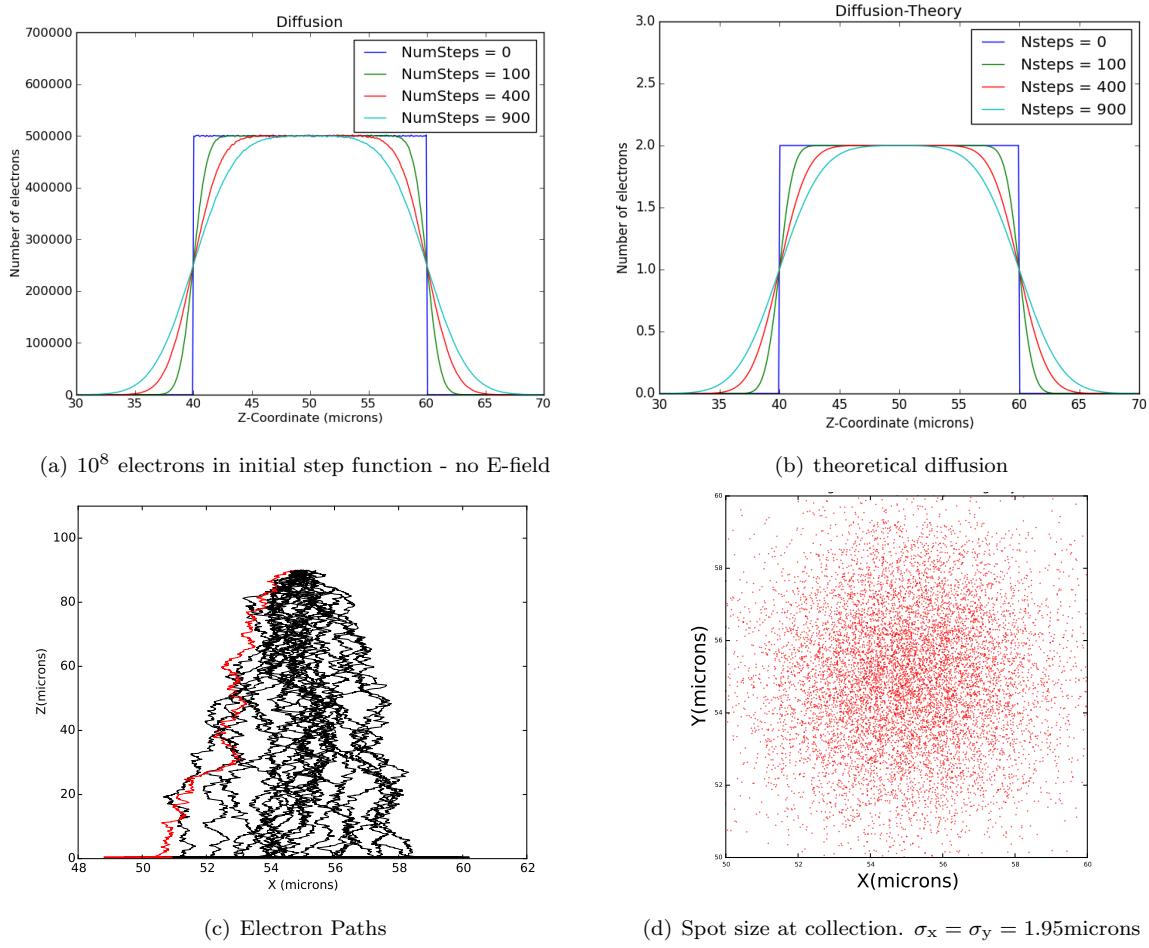


Figure 29: Tests of diffusion model. (a) shows a simulation of electrons diffusing away from an initial step function as compared to (b) a theoretical calculation of an initial step function given by  $\text{erf}\left(\frac{x-x_1}{\sqrt{4D\tau N_{\text{steps}}}}\right) - \text{erf}\left(\frac{x-x_2}{\sqrt{4D\tau N_{\text{steps}}}}\right)$  with  $D = \frac{kT}{q}\mu$ . (c) and (d) show the spreading of an initial spot with a diameter of 0.1 microns.

Two tests were done to check the accuracy of the diffusion model, and the results of these tests are shown in Figure 29. Figures (a) and (b) show a pure diffusion check (no E-field) showing that an initial step function of electrons diffuses as expected according to diffusion theory. Figures (c) and (d) show the spreading of an initial distribution of photoelectrons with a diameter of 0.1 microns starting at  $z=90$  microns as they propagate down to  $z=0$ . We expect (see for example Holland, et.al. [12]) that the lateral spread of a spot due to diffusion is given by:

$$\sigma = \sqrt{2Dt_{\text{tr}}} \quad (\text{D.4})$$

where the transit time  $t_{\text{tr}}$  is given by

$$t_{\text{tr}} = \frac{T_{\text{si}}}{\mu E} \quad (\text{D.5})$$

Since the diffusion coefficient and the mobility are related by the Einstein relation:

$$D = \frac{kT}{q}\mu \quad (\text{D.6})$$

this simplifies to:

$$\sigma = T_{si} \sqrt{\frac{2kT}{qV}} \quad (D.7)$$

where V is the applied voltage across the CCD. For the conditions simulated here ( $T_{si} = 90\text{ microns}$ ;  $V = 64\text{V}$ ;  $T = -100\text{K}$ ), we expect  $\sigma = 1.95\text{ microns}$ , in excellent agreement with the simulated value.

## E Acknowledgements

Andrew Bradshaw and Kirk Gilmore have contributed significantly to this work. Many thanks to Perry Gee for lab and IT support, and to Peter Pederson for Optical Simulator insights. We also acknowledge helpful suggestions from Pierre Astier, Andrei Nomerotski, Andy Rasmussen, John Peterson, Rachel Mandelbaum, and Debbie Bard.

## References

- [1] P. Antilogus, P. Astier, P. Doherty, A. Guyonnet, and N. Regnault. The brighter-fatter effect and pixel correlations in CCD sensors. *Journal of Instrumentation*, 9:C03048, March 2014.
- [2] P. E. Doherty, P. Antilogus, P. Astier, J. Chiang, D. K. Gilmore, A. Guyonnet, D. Huang, H. Kelly, I. Kotov, P. Kubanek, A. Nomerotski, P. O'Connor, A. Rasmussen, V. J. Riot, C. W. Stubbs, P. Takacs, J. A. Tyson, and K. Vetter. Electro-optical testing of fully depleted CCD image sensors for the Large Synoptic Survey Telescope camera. In *High Energy, Optical, and Infrared Detectors for Astronomy VI*, volume 9154 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 915418, July 2014.
- [3] A. Guyonnet, P. Astier, P. Antilogus, N. Regnault, and P. Doherty. Evidence for self-interaction of charge distribution in charge-coupled devices. *"Astro. & Astrophys."*, 575:A41, March 2015.
- [4] D. Gruen, G. M. Bernstein, M. Jarvis, B. Rowe, V. Vikram, A. A. Plazas, and S. Seitz. Characterization and correction of charge-induced pixel shifts in DECam. *Journal of Instrumentation*, 10:C05032, May 2015.
- [5] C. W. Walter. The Brighter-Fatter and other sensor effects in CCD simulations for precision astronomy. *Journal of Instrumentation*, 10:C05015, May 2015.
- [6] A. Rasmussen. Use of sensor characterization data to tune electrostatic model parameters for LSST sensors. *Journal of Instrumentation*, 10:C05028, May 2015.
- [7] J. R. Peterson, J. G. Jernigan, S. M. Kahn, A. P. Rasmussen, E. Peng, Z. Ahmad, J. Bankert, C. Chang, C. Claver, D. K. Gilmore, E. Grace, M. Hannel, M. Hodge, S. Lorenz, A. Lupu, A. Meert, S. Nagarajan, N. Todd, A. Winans, and M. Young. Simulation of Astronomical Images from Optical Survey Telescopes Using a Comprehensive Photon Monte Carlo Approach. *"Astrophys. J. Supp."*, 218:14, May 2015.
- [8] J. R. Peterson and J. G. Jernigan. phoSim: Photon Simulator. Astrophysics Source Code Library, July 2013.
- [9] J. A. Tyson, J. Sasian, K. Gilmore, A. Bradshaw, C. Claver, M. Clint, G. Muller, G. Poczulp, and E. Ressegueie. LSST optical beam simulator. In *High Energy, Optical, and Infrared Detectors for Astronomy VI*, volume 9154 of *Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series*, page 915415, July 2014.
- [10] University of Arizona. Imaging Technology Laboratory, 2016. <http://www.itl.arizona.edu>.
- [11] A. Bradshaw, C. Lage, E. Ressegueie, and J. A. Tyson. Mapping charge transport effects in thick CCDs with a dithered array of 40,000 stars. *Journal of Instrumentation*, 10:C04034, April 2015.
- [12] S. E. Holland, C. J. Bebek, W. F. Kolbe, and J. S. Lee. Physics of fully depleted CCDs. *Journal of Instrumentation*, 9:C03057, March 2014.

- [13] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes*. Cambridge University Press, third edition, 2007.
- [14] C. Jacobini, C. Canali, G.. Ottaviani, and A. Quaranta. A review of some transport properties of silicon. *Solid-State Electronics*, 20:77–89, 1977.