

Monte Carlo Simulation and Analysis Framework for a CdZnTe-based Spherical Coded Aperture and Compton Gamma-ray Imager

Final Report

NE 255 - Numerical Simulation in Radiation Transport
University of California, Berkeley
Department of Nuclear Engineering

Daniel Hellfeld

December 14, 2016

I. Introduction

Gamma-ray imaging allows for efficient detection, characterization, and localization of compact radioactive sources in complex environments. Fieldable detector systems employing active planar coded masks have demonstrated broad energy sensitivity via both coded aperture (tens to hundreds of keV) and Compton imaging (hundreds of keV to a few MeV) modalities [1]. However, the planar configuration of such systems suffer from an inherent limited field-of-view (FOV), especially in the case of coded aperture. Lawrence Berkeley National Laboratory (LBNL) has proposed a novel design by rearranging the detectors into an active coded spherical configuration. The Portable Radiation Imaging Spectroscopy and Mapping (PRISM) detector system currently under development (see Fig. 1a-b) consists of cm³ CdZnTe (CZT) coplanar grid (CPG) [2] gamma-ray detectors arranged on the inner surface of a 14 cm diameter spherical housing. There exists 192 available detector locations on the sphere, and an active coded arrangement, or a pattern of occupied and empty locations (see Fig. 1c), of the detectors facilitates gamma-ray imaging in 4π using both coded aperture and Compton imaging modalities.

This project aims to improve an existing Geant4 [3] Monte Carlo simulation for PRISM. The purpose of the simulation is to determine the response of the system to radioactive sources of varying energies, intensities, and spatial distributions in the entire 4π FOV. The simulated response can then be used as a tool to inform prototype design, for performance characterization, and for image reconstruction.

The original simulation was developed in order to generate a first-order approximation of the coded aperture response in the far-field limit (parallel rays at infinity). The simulation essentially functioned as a ray-tracer (i.e. no scattering, no secondary electron tracking), though, some physics was included to account for detector attenuation. This is important as the CPG detectors can measure the depth-of-interaction (DOI) in each detector, or the distance from the interaction to the anode. This greater degree of position resolution within each detector element is used to significantly improve the coded aperture reconstruction.

The purpose of this project is to upgrade the original simulation to include more functionality such as scattering physics, secondary electron production and tracking, multi-site events, geometry modification, and near-field and extended sources. In addition to the simulation, analysis tools will be developed to perform image reconstruction in both coded aperture and Compton imaging modes. The objective is to develop a high-fidelity, robust, and easy to use simulation and analysis framework that can be used to answer a variety of research questions for the novel concept of a spherical coded aperture and Compton imaging system.

This report will first discuss the mathematics and algorithms used in the simulation and analysis framework. Next, an overview of the developed code will be presented, outlining how to execute the simulation and analysis code. Some example test inputs will be included, as well as a discussion of the output results. Finally, the work will be summarized and future work will be outlined.

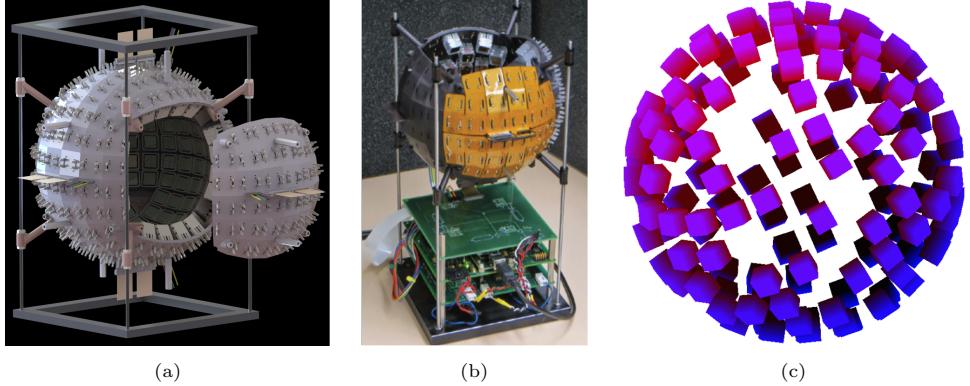


Fig. 1. (a) Modular design of PRISM. (b) PRISM prototype system currently under development at LBNL. (c) Example coded arrangement of detectors.

II. Mathematics

II.A Coded Aperture Imaging

Coded aperture is a technique first developed for X-ray astronomy and uses the principle of unique photon modulation for image reconstruction [4]. Typically an array of position sensitive photon detectors are placed behind a coded array (also referred to as an *aperture* or *mask*) of photon absorbing material (such as lead or tungsten). The concept is similar to a pinhole camera, however, several pinholes are used in the aperture (typically with an *open fraction* of 50% [5]) to increase the signal-to-noise-ratio (SNR) - which is necessary when attempting to image weak sources. The aperture arrays are designed such that the detected images (or *shadowgrams*) uniquely correspond to a location in the image space. Reconstruction algorithms are used to decode the detected image, typically using either correlation or iterative methods [5, 6].

In the case of PRISM, a passive lead or tungsten mask is not used. Instead, the cm³ CZT elements are used as both the mask and detectors (referred to as an *active* configuration). At low energies, the CZT easily attenuates the photons and acts as a highly absorbing mask (counts can also be rejected in post-processing). Moreover, an active mask increases the detection efficiency, improves weak source detection, and significantly reduces the weight of the system.

As in a vast variety of physical processes, the imaging system can be described by a convolution. In order to avoid working with reflected images, as is the case when using a pinhole camera, it is typical to instead describe the system as a correlation (see Appendix A). The imaging equation can written as

$$P = A \star O + N, \quad (1)$$

where P is the detected image, A is the aperture, O is the source object, N is noise, and \star is the correlation operator [5]. To analytically solve for the source object O , the convolution theorem (see Appendix A) is applied this to Eq. (1), ultimately leading to

$$\hat{O} = O + R\mathfrak{F}^{-1} \left[\frac{\mathfrak{F}(N)}{\mathfrak{F}(A)} \right], \quad (2)$$

where \hat{O} is the reconstructed source object, R is the reflection operator, \mathfrak{F} is the Fourier Transform, and \mathfrak{F}^{-1} is the inverse Fourier Transform. This method of *deconvolution* is simple and straightforward. However, since the

aperture is an array of ones and zeros, the $\mathfrak{F}(A)$ term can be small, leading to large noise gain in the image [5]. A variety of different reconstruction techniques exist, including simple (and filtered) back-projection [7], cross-correlation [5], and maximum likelihood expectation maximization (MLEM) [6]. Back-projection simply adds intensity to each image pixel proportional to the probability that the event originated from that pixel. However, since a single event is not unique to a single image pixel, the resulting images can be quite blurry. Cross-correlation typically uses the mask pattern to define a post-processing array, G , such that $A \star G$ approximates a delta function. G is applied to Eq. (1) and the equation is inverted to solve for O . However, $A \star G$ is never a true delta function and thus the images will contain some blur. Moreover, with the spherical array in PRISM, it is unclear how G would be defined. For these reasons, this report will focus solely on MLEM.

The derivation of the MLEM method is shown in Appendix B. The final result is concise and gives a relation between the current source distribution λ_j^n and the next iteration, λ_j^{n+1}

$$\lambda_j^{n+1} = \frac{\lambda_j^n}{\sum_{i \in J_j} C_{ij}} \sum_{i \in J_j} \frac{C_{ij} Y_i}{\sum_{k \in I_i} C_{ik} \lambda_k^n}, \quad (3)$$

where i is the detector index, j is the image pixel index, I_i are the set of image pixels that contribute to detector i , J_j are the set of detectors to which image pixel j contributes., C_{ij} is the probability of getting a count in detector i from source j , or *system response* (determined from simulation), and Y_i are the counts in detector i .

This method can be used to reconstruct the source distribution without the noise gain effects inherent in analytical deconvolution and without the blur of back-projection. However, because of its iterative nature, this methods comes at a cost of computational time. In most cases, the technique can be easily vectorized and computed quickly; however, it can become difficult when attempting to reconstruct in a list-mode fashion (count by count) in real time in a potentially large three dimensional voxelized space. Furthermore, the convergence of this method is a topic of debate in the field. MLEM attempts to reduce noise and increase resolution with each iteration, however if the input data contains a significant amount of noise, there can come a point where more iterations makes the image worse. Therefore it can be difficult to define convergence criteria. In some cases, the number of iterations is fixed based on previous experience [8, 9].

II.B Compton Imaging

Coded aperture imaging is powerful when the photon energy is low because the modulation (or *mechanical collimation*) is what produces the unique coding on the detected image. As the energy is increased, the photoelectric absorption cross-section decreases and the Compton scattering cross-section increases. The decrease in attenuation and increase in scattering will essentially remove the coded effect and blur the images. Therefore, an *electric collimation* technique such as Compton imaging must be used for higher energy photons. Gamma-rays can undergo Compton scattering in one detector, deposit a detectable amount of energy, and then eventually undergo a photoelectric absorption in another detector. A variety of other scenarios exists (scatter-scatter-absorb, scatter-escape, pair produce-scatter-absorb, etc.), but this report will focus only on two-interaction tracks. By measuring the interaction positions and energy depositions of the individual interactions, the photon path can be tracked through the system. The two interactions can then be used to determine a cone of incident angles from which the incident photon originated. The image is reconstructed by the overlap of multiple cones (in 3D) or circles (in 2D).

If the incident (E_γ) and scattered gamma-ray (E'_γ) energies are known, the scattering angle can be solved

with the Compton scattering formula

$$\mu = \cos(\theta) = 1 + \frac{m_e c^2}{E_\gamma} - \frac{m_e c^2}{E'_\gamma}, \quad (4)$$

where $m_e c^2$ is the rest mass of the electron ($= 511$ keV). The positions of the scattering interaction and the subsequent photoelectric absorption define an axis. A cone of possible gamma-ray directions can then be defined and back-projected into space from the scattering interaction location. In practice, detectors have finite energy, position, and time resolution. First, finite time resolution will obscure the exact ordering of events. Coincidence gates must be defined, which will introduce random coincidences into the data. Finite position resolution will introduce uncertainty into the cone axis. Typically for a given interaction, the interaction positions are assigned to the center of the detectors. If the *lever arm*, or the distance between the two interactions, is small, this effect can be large. Finally, finite energy resolution will introduce uncertainty into the calculated scattering angle and limit our ability to determine if two events came from the same initial gamma-ray [10].

For a given two-interaction track, the two events must be sequenced in time. First, the measured energy depositions in the detectors are assumed to sum to the incident gamma-ray energy. If the $E_{tot} \leq 256$ keV, the kinematics of Eq. (4) states that the first interaction will always deposit less energy than the second. Therefore the sequence of the events can be known by comparing the energies. If $E_{tot} > 256$ keV, the sequencing will be ambiguous. Kinematic tests, such as the Compton Edge test can be used to discard forbidden sequences [10]. The Compton Edge refers to the maximum amount of energy than can be deposited for a given Compton scattering event. It occurs when the gamma-ray undergoes a complete backscatter ($\theta = \pi$). For each sequence, test can be performed to determine whether the first energy deposition exceeds the Compton Edge. If so, the sequence can be rejected.

In the far-field limit (parallel rays at infinity), each cone can be projected from the origin of the image space. The image space covers all of 4π and is defined as a unit sphere, \mathbb{S}^2 . We define a unit vector for the cone axis as $\hat{\omega}$ and a point in \mathbb{S}^2 as \hat{x} . The dot product of these two vectors than corresponds to the cosine of the angle between them. If this angle equals the scattering angle, then the point in \mathbb{S}^2 is on the cone. An infinitesimally thin cone back-projected into a discretized image space would result in pixellation effects. Therefore we give the cone a small width and intensity defined by a Gaussian function. The back-projected image is then the sum of all the cones

$$b(\hat{x}) = \sum_{i=1}^n \frac{w_i}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{(\hat{x} \cdot \hat{\omega} - \mu_i)^2}{2\sigma_i^2}\right), \quad (5)$$

where i runs over all the cones, σ is the width of the cone, and w is the weight of the cone [11]. The width of the cone is selected to be smaller than the expected resolution of the system, as to be able to see resolution-based effects in the image reconstruction. One may be inclined to include the uncertainty in the scattering angle from the uncertainty in the measured energies into the cone width. For a small number of cones, this will certainly help with source location, however with a large number of cones, this will simply degrade the resolution because the uncertainty is introduced twice. Imagine a large number of cones with a small width being back-projected - the uncertainty in the cones position in space will be built into its projection and over a large number of cones, will define the resolution of the system. If the uncertainty is included, the resulting image will be excessively blurred [11]. The weight of the cones are defined in two ways. In the case of ambiguous sequences, we apply a normalized weight to each sequence proportional to the Klein-Nishina differential scattering cross-section for the two scattering angles. The sequence with a larger probability of occurring is given a larger weight. The

Klein-Nishina formula is given by

$$\frac{d\sigma}{d\Omega}(E_\gamma, \theta) = \frac{\alpha^2 r_c^2 P(E_\gamma, \theta)^2}{2} [P(E_\gamma, \theta) + P(E_\gamma, \theta)^{-1} - 1 + \cos^2(\theta)] , \quad (6)$$

where α is the fine structure constant, r_c is the reduced Compton wavelength of the electron, and $P(E_\gamma, \theta)$ is given by

$$P(E_\gamma, \theta) = \frac{1}{1 + \frac{E_\gamma}{m_e c^2} (1 - \cos \theta)} . \quad (7)$$

For all cones, the lever arm is also incorporated into the weight. Events with smaller lever arms are more likely to occur and will produce images with worse angular resolution. Therefore we apply a weight of L^2 where L is the lever arm in order to more heavily weight the larger lever arm events [11].

III. Algorithms

The simulation is built around the Monte Carlo toolkit, Geant4. Individual particles, as well as any secondary particles produced, are tracked from birth to death given some initial conditions and underlying physics. This is accomplished by using random sampling techniques. The geometry is built first, detailing the size, position, and material of the objects in the system. Relevant physics are defined in all materials as well as for all initial and subsequently produced particles. The photoelectric effect, Compton scattering, Rayleigh scattering, and pair production were included for the gamma-rays and multiple scattering, ionization, and bremsstrahlung were included for the electrons. A particle is then given to the tracker with a particle type, energy, position, and momentum direction. Each step in the particle track is then determined by randomly sampling the next collision distance and comparing to the next boundary distance. The particle is moved accordingly and the interaction is then randomly sampled (including the interaction type, the outgoing energy and direction, and the production of any secondary particles). If any secondary particles are produced due to an interaction, the primary particle track is paused while the secondary particle tracks are followed to completion one by one. At each step (boundary crossing or interaction site) in the track, relevant information is stored in memory. This can include position, energy, direction, etc. This process continues until the particle is absorbed, leaves the system of interest, or falls below a defined threshold (e.g. < 1 eV). Once the particle track is completed, the next particle is given to the tracker and the process is repeated. Following the termination of the last particle sent to the tracker, the stored information for all particles is written to disk.

The simulation records the detector ID, event number, hit number, track ID, energy deposition, time, DOI, and the interaction process for every step in the track. In practice, the system would only be able to record the detector ID, the energy deposition, DOI, and the time of each triggered event.

The primary purpose of the simulation is to simulate a source of gamma-rays of some energy in space and measure the response of the detectors. The imaging system is small, therefore most sources can be considered far-field (all rays are parallel). To implement far-field sources, a disk slightly larger than the PRISM sphere is defined on the z -axis. A position on the disk is randomly sampled and then rotated through space according to the location of the source. The direction of the ray is then given as the opposite of the source position in space. The result is parallel rays emanating from a disk slightly larger than the system positioned at the source location (see Fig. 2a). A disk was chosen to reduce the number of photons that do not contribute to the detector response. Also implemented in the simulation are far-field ring sources and near-field point sources (see Fig. 2b-c). The ring source uses the same implementation as the far field source, but each position is shifted

randomly around a ring of some radius provided by the user. The near-field source is accomplished by randomly sampling photon directions in a cone that is slightly larger than the PRISM sphere, given a certain distance from the origin.

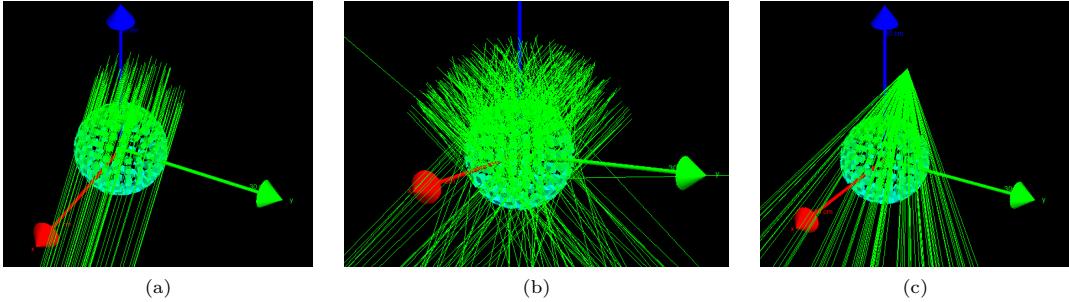


Fig. 2. Geant4 visualization of a (a) far-field point source, (b) far-field ring source, and (c) near-field point source

The current state of the simulation is meant to be used to either generate a full system response or produce a random data set in which image reconstruction can be performed. In the case of the full system response, which is essential for coded aperture imaging, a point source is simulated at each point in the image space in order to determine the probability that a photon detected in detector i originated from image pixel j . The 4π image space is discretized into 3072 equal area pixels on \mathbb{S}^2 defined using the HEALPix library [12]. For each image pixel, a user defined number of particles are simulated, and the relative responses of all the detectors are obtained. These distributions across all of the image pixels act as the coding that is used in the coded aperture reconstruction. A single source can then be simulated (potentially with a smaller number of detected events and noise included) in order to test reconstruction algorithms. The same can be said for Compton imaging, which is dependent on the number of cones to back-project. A single source can be simulated in space with some arbitrary amount of particles in order to generate a data set of sequenced events. The imaging capability can then be determined as a function of the number of cones used.

In the coded aperture analysis code, only full energy absorption events are considered, so a filter is used to reject any events with less than the full energy deposited. A histogram routine is used to bin the response in each detector for each source location into a matrix. This matrix is then passed to a vectorized MLEM algorithm written in Python.

In the Compton imaging analysis code, the output data must first be parsed for coincident events. In practice, this would be done by taking all events within some coincidence time gate following an interaction. Currently the code searches for two sequential events in different detectors that sum near a known incident energy. The coincidence events are checked against the Compton Edge test and the accepted events are placed into a sequence array. The array is then looped over and the cone back-projection is performed, including the Klein-Nishina and lever arm weighting. Currently a cone width of 3° is used to avoid pixelation effects. This is smaller than the expected angular resolution of PRISM ($\sim 10^\circ$). The back-projection intensities are added to the image array in each iteration. The dot products in Eq. (5) can be easily vectorized in Python, therefore the number of image pixels can be significantly increased as compared to coded aperture (in which the image discretization is defined on how computationally expensive it is to generate the system response at every point).

IV. Code Use

The Geant4 simulation is written in C++ and compiled with CMake on OSX version 10.10.5. The Geant4 version is Geant4.10.2.p1 and requires a CMake version 3.5+. The version of Python used was Python 2.7.12 (Anaconda 4.1.1). This report will assume that all the required software has been installed and sourced correctly. The `compile.sh` shell script will initialize the compilation of the code. It can be run with `$sh compile.sh`. The user must input their own directory paths to the CMake executable and the Geant4 directory. A final `$make` command in the top level directory will compile the code and produce an executable called `PRISM`. The code can be run by calling the executable and providing a macro file. For example: `./PRISM macros/run.mac`. The macro file is where the user can input various parameters for the simulation, including particle energy, source location, image pixel indexing, detector size, mask configuration (in a hexadecimal format), number of particles to run, output filename, and output file type. The `run.mac` is provided as an example of the available commands and the `vis.mac` macro is provided as a visualization example. Visualization in Geant4 depends heavily on the system architecture and the installed software on the machine. The visualization is not essential for this report and will not be mentioned any further. More information on each user command in the macros can also be accessed during the simulation.

The system response simulation uses the macro files provided in the `macros/response` directory. The set up of the simulation is done in the `hp.mac` macro and the number of particles to run at each source location is specified in the `main.mac` macro. To run the system response simulation, run `./PRISM macros/response/hp.mac`.

Outputs will be placed in the `output/` directory. The data can then be read into memory and manipulated with the Python scripts in the `tools/` directory. The coded aperture system response data can be histogrammed and stored into a Python numpy array using the `HistogramCodedApertureResponse.py` script by providing the output file path and energy of interest in the command line execution. The response output files can be quite large (a few gigabytes), therefore the script will perform the histogramming once and save the numpy array into a `.npy` file which can be quickly read in at a later time. Currently the entire output data file is placed into memory during execution, it is the users responsibility to ensure enough RAM is available. The `CodedApertureReconstruction.py` script will read in the system reponse numpy array and will perform the MLEM reconstruction on either a user defined row of the system response (to test reconstruction) or a provided output data file. The `ComptonReconstruction.py` script will read in the binary simulation output data and perform the coincidence selecting, event sequencing, and cone back-projection for Compton imaging. The scripts requires the Python HEALPix package, `healpy` (<https://healpy.readthedocs.io/en/latest/>).

V. Test Problems and Results

Far-field point source system response simulations were run for 60 keV and 186 keV gamma-rays with 100,000 particles at each of the 3072 source locations. A random mask was used with hexadecimal code 21524FA478BD521AB44791322B545C979943A029753854BB. Electron tracking was turned off and all energy deposition was assumed to be at the point of interaction. The output file was parsed and the system response was generated with and without DOI. The responses are shown in Fig. 3.

The detectors and image pixels are indexed from the north pole to the south pole in a spiraling fashion. In each plot, the detectors are indexed to 192, however because this is a coded arrangement, some detectors are missing (empty rows). It is clear that the detectors closest to the source see the largest proportion of the flux, as expected. When DOI is included with 10 depth bins, the number of detectors is increased by a factor of 10. In

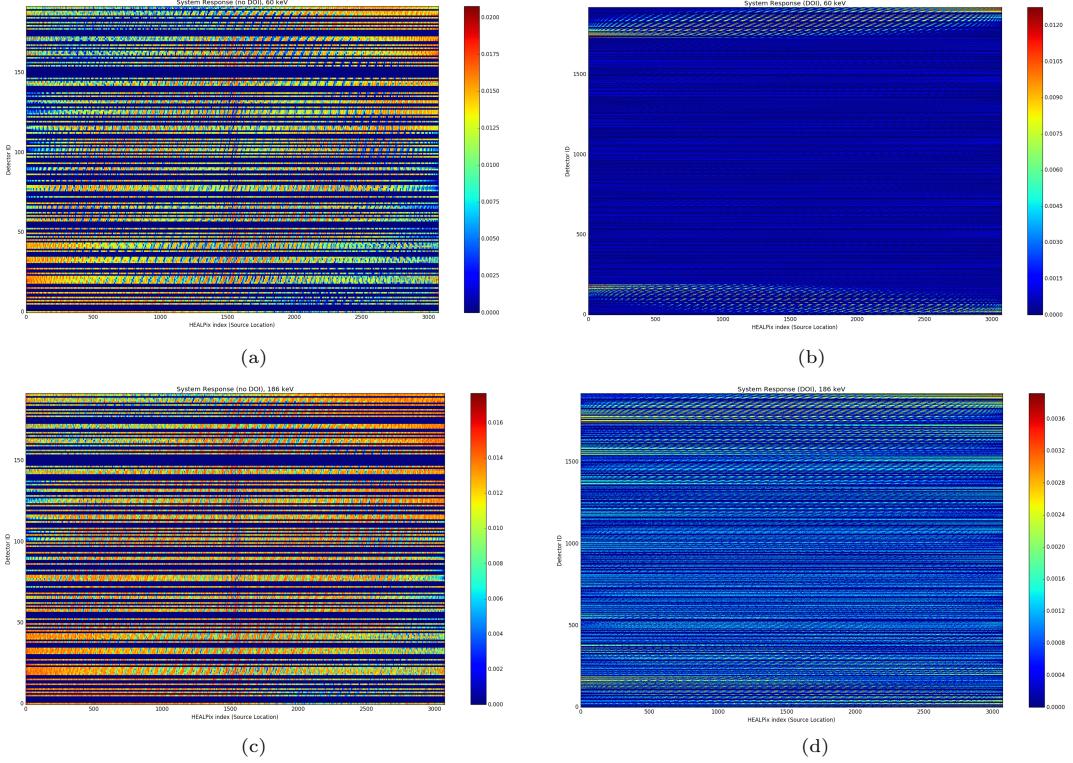


Fig. 3. System response of a randomly populated mask to 60 keV gamma-rays (a) without DOI and (b) with DOI. System response of a randomly populated mask to 186 keV gamma-rays (b) without DOI and (c) with DOI.

this case, detectors 1 through 192 correspond to the inner bins (closest to the origin) and detectors 1728 to 1920 correspond to the bins furthest from the origin. In the 60 keV case, the gamma-ray attenuation is quite large (mean free path is a fraction of a millimeter). Most of the intensity is seen on the outer bins because they are in clear view of the source. Significant intensity is seen in the inner bins as well, accounting for the gamma-rays that pass through the open spaces of the mask and interact. The middle bins see the least amount of flux because of the large occlusion from the other detectors and the small surface area they present to the source. The large modulation of gamma-rays facilitates a higher level of coding, and thus the image reconstruction should be better when DOI is included. When the energy is increased, the DOI behavior degrades as the gamma-rays can now penetrate deeper into the detectors. Therefore, the intensity is distributed more throughout all the detector bins.

An arbitrary column of the system response was taken as the signal and run through the MLEM image reconstruction. The results for 60 keV with and without DOI are shown in Fig. 4. In the DOI case, only the inner detector responses were used as they present a compromise between the best encoding and counts. The images using 10 and 30 MLEM iterations are also shown. The true source location is shown with a blue “X”.

As expected, the blur in the image is reduced as the number of MLEM iterations is increased. No noise was included in the simulations therefore the source location can be reconstructed exactly as the number of iterations becomes large. It is also clear that including depth-of-interaction significantly improves the image reconstruction. Each image tends to have a slight bias, however this is not surprising because an optimized mask pattern was not used.

To test the Compton imaging analysis, far-field point source simulations were run with 200 keV and 662 keV gamma-rays and 500,000 particles in each case. The mask was fully populated. Again, electron tracking was

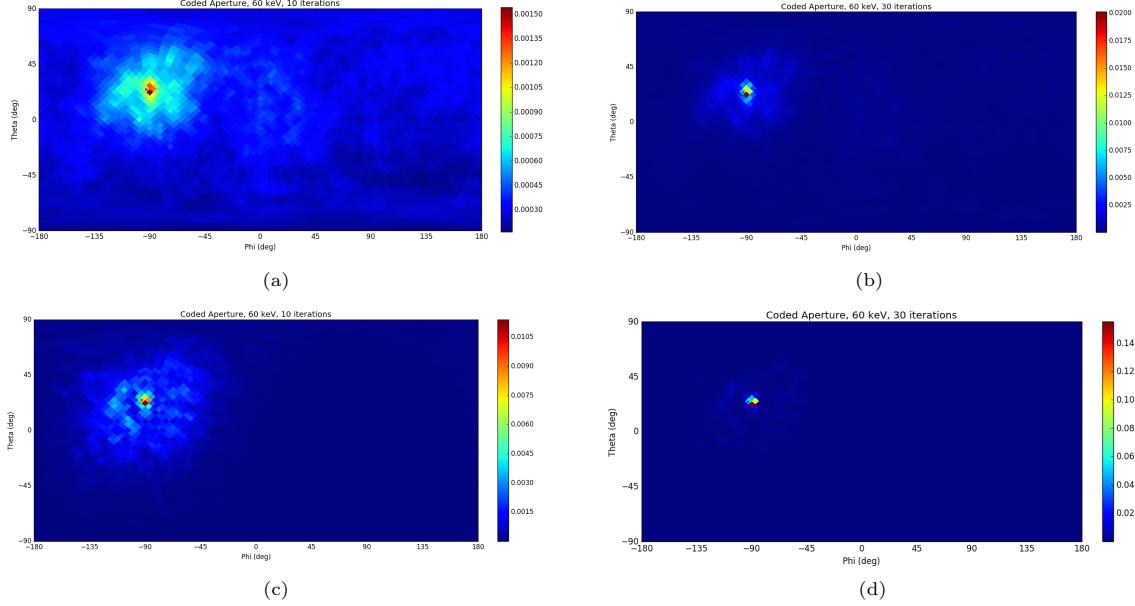


Fig. 4. MLEM reconstruction of a 60 keV far-field point source without DOI using (a) 10 MLEM iterations and (b) 30 MLEM iterations. MLEM reconstruction of a 60 keV far-field point source with DOI using (c) 10 MLEM iterations and (d) 30 MLEM iterations.

turned off and all energy deposition was assumed to be at the point of interaction. The output was parsed for coincident detections assuming the initial gamma-ray energy was known, and the events were sequenced. Accepted sequences were back-projected with respective Klein-Nishina and lever arm weighting. The reconstructed images with 50 and 2000 cones are shown in Fig. 5.

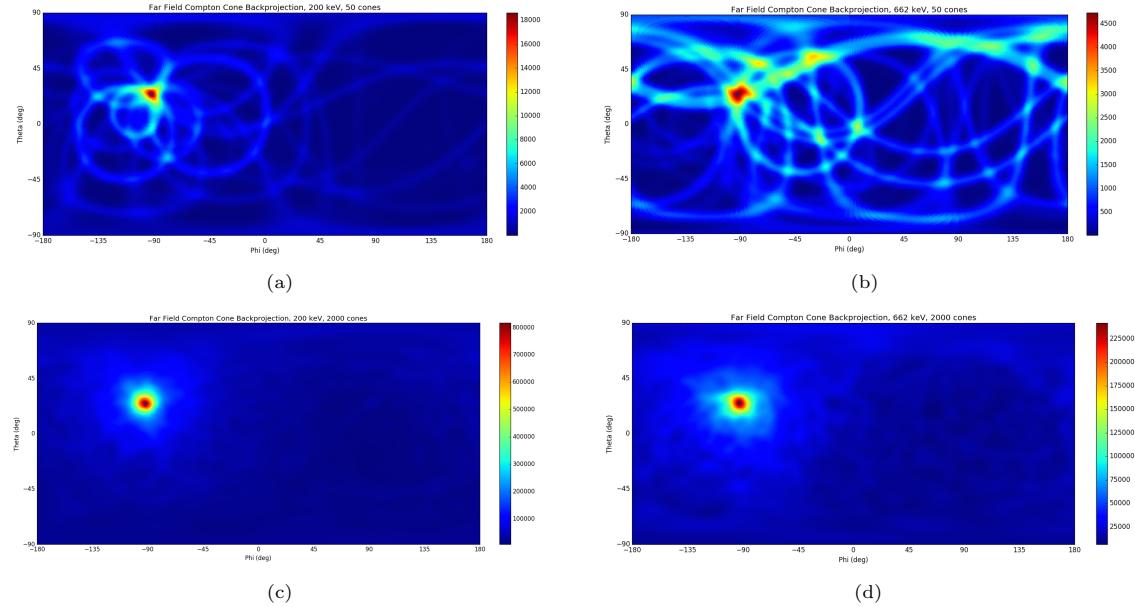


Fig. 5. Compton cone back-projection image for a far-field (a) 200 keV and (b) 662 keV point source with 50 cones. Compton cone back-projection image for a far-field (c) 200 keV and (d) 662 keV point source with 2000 cones.

A test was also performed to verify that the first interaction always deposits less energy than the second if the incident gamma-ray energy is less than 256 keV. This is a good step towards validating the random sampling

procedure that is being done internally in Geant4 for each Compton scattering interaction. The ratio of the first to second energy deposition is shown for 1000 accepted sequences for both 200 keV and 662 keV in Fig. 6. The 200 keV sequences never exceed a ratio of 1, indicating that the first interaction always deposits less energy than the second. The 662 keV sequences produce ratio both above and below 1, indicating that the amount of energy deposited first is ambiguous.

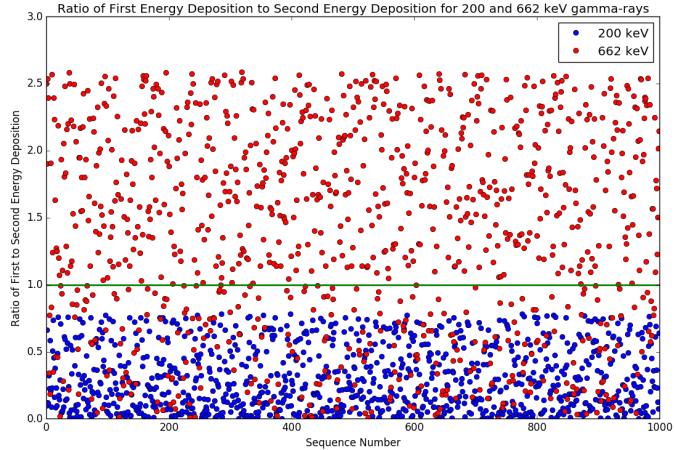


Fig. 6. Ratio of the first energy deposition to the second energy deposition for 200 keV (blue) and 662 keV (red) gamma-rays.

Finally, a far-field ring source was simulated at 60 keV (with a random mask) and 662 keV (with a fully populated mask) with 1×10^6 particles each. The image reconstructions with coded aperture and 50 iterations of MLEM including DOI (60 keV) and Compton cone back-projection (662 keV) are shown in Fig. 7. The coded aperture image has low noise, but fails to uniformly reconstruct the ring this may be due to the unoptimized mask configuration or insufficient statistics. The Compton reconstruction contains more noise but captures the entire ring.

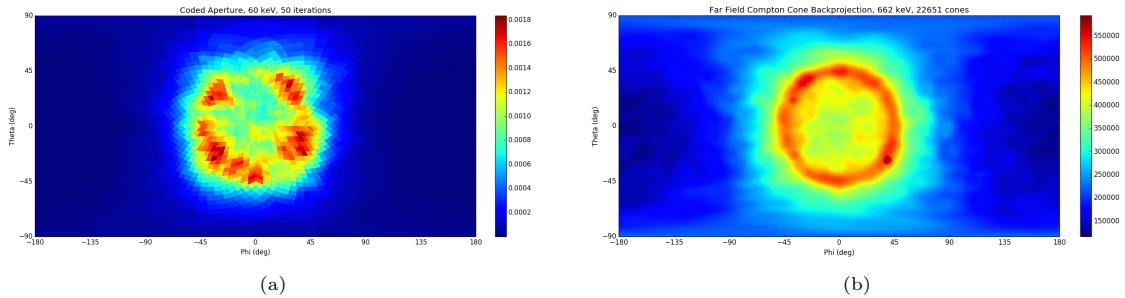


Fig. 7. (a) MLEM coded aperture image reconstruction (50 iterations) of a far-field ring source with 60 keV gamma-rays with a random mask and DOI. (b) Compton cone back-projection image of a far field ring source with 662 keV gamma-rays and a fully populated mask.

VI. Summary/Conclusions

The PRISM imaging system is a handheld, broad energy sensitive, high efficiency gamma-ray imager currently under development at LBNL. The spherical design of PRISM is the first of its kind and will facilitate 4π imaging in both coded aperture and Compton imaging modalities. The initial Geant4 simulation used as

a ray-tracer for preliminary studies has been upgraded to include scattering, secondary electron production, simple geometry modifications, and near-field and extended sources. Analysis tools were developed to parse the Geant4 output files and perform MLEM image reconstruction of coded aperture data and cone back-projection for Compton data. In both cases, the reconstruction behaved as expected as a function of energy, iterations, DOI, and number of cones. The results of this project have demonstrated the 4π imaging concept of a spherical CZT based gamma-ray imaging system and have laid the groundwork for further analysis.

VI.A Future work

Many studies still need to be conducted, such as multiple far-field point sources, different extended sources, and near-field sources. As the source is brought closer to the detector system, static 3D imaging may be possible with coded aperture (using magnification effects) and Compton imaging (back-projecting cones into \mathbb{R}^3 instead of \mathbb{S}^2). Moreover, since the system is handheld, the concept of including motion into the simulation must be explored. This will also include 3D tomographic imaging and real-time imaging - both quite challenging problems. PRISM will also be equipped with a LiDAR and visual camera in order to fuse contextual information with the radiation map. This will require an investigation into Simultaneous Location And Mapping (SLAM) algorithms and exploring how contextual information can be used to constrain the imaging procedure [13]. More complex Compton reconstruction will be explored, including filtered back-projection and MLEM. The effect of including DOI in the Compton reconstruction will be studied. Event sequencing for events with greater than two interactions will be implemented, as well as appropriate energy and position blurring in the detector response and image reconstruction. Finally, geometry manipulation will be made more flexible to facilitate systematic studies of system performance as a function of detector size, sphere radius, detector spacing, mask configuration, etc. This will be critical in the future design versions of PRISM to optimize the coded aperture and Compton imaging capability for a given geometry.

VII. Appendix

VII.A Correlation and Convolution

In one dimension, the correlation (\star) of two functions is given by

$$f(x) \star g(x) = \int_{-\infty}^{\infty} du g^*(u-x)f(u), \quad (\text{A.1})$$

where $g^*(x)$ referred to the complex conjugate of $g(x)$. Correlation can be related to the convolution operation ($*$) by the adjoint operation (or simply a reflection operation, as all functions are strictly real in this case)

$$f(x) * g(x) = \int_{-\infty}^{\infty} du g(x-u)f(u). \quad (\text{A.2})$$

The convolution theorem is given by

$$\mathfrak{F}[f(x) * g(x)] = \mathfrak{F}[f(x)]\mathfrak{F}[g(x)] = F(k)G(k), \quad (\text{A.3})$$

where \mathfrak{F} is the Fourier Transform and k is spatial frequency.

VII.B Derivation of MLEM

Assume the measured detector counts are Poisson distributed, such that the probability of observing a count k in detector λ is

$$P(z = k) = \frac{e^{-\lambda} \lambda^k}{k!}, \quad (\text{B.1})$$

and thus the likelihood over all detector bins and image pixels is

$$L(X, \lambda) = \prod_{i,j \in I_i} \frac{e^{-C_{ij}\lambda_j} (C_{ij}\lambda_j)^{X_{ij}}}{X_{ij}!}, \quad (\text{B.2})$$

where i is the detector index, j is the image pixel index, I_i are the set of image pixels that contribute to detector i , C_{ij} is the probability of getting a count in detector i from source j , or *system response* (determined from simulation), λ_j is the intensity at image pixel j , and X_{ij} is the number of photons emitted from source j to detector i . Also note that the total number of counts in detector i is given by $Y_i = \sum_{j \in I_i} X_{ij}$. To make Eq. (6) more manageable, the log-likelihood function is typically used

$$\log L(X, \lambda) = \sum_i \sum_{j \in I_i} [-C_{ij}\lambda_j + X_{ij} \log(C_{ij}\lambda_j) - \log(X_{ij}!)] . \quad (\text{B.3})$$

The conditional expectation of the log-likelihood is taken with respect to Y and the current source distribution λ^n

$$E[\log L(X, \lambda)|Y, \lambda^n] = \sum_i \sum_{j \in I_i} \left[-C_{ij}\lambda_j + \frac{C_{ij}\lambda_j^n Y_i}{\sum_{k \in I_i} C_{ik}\lambda_k^n} \log(C_{ij}\lambda_j) \right] . \quad (\text{B.4})$$

The expectation is then maximized, by taking a partial derivative with respect to λ_j and setting it equal to 0

$$\frac{\partial}{\partial \lambda_j} E[\log L(X, \lambda)|Y, \lambda^n] = 0 . \quad (\text{B.5})$$

Solving the above results in a relation between the current source distribution and the next iteration

$$\lambda_j^{n+1} = \frac{\lambda_j^n}{\sum_{i \in J_j} C_{ij}} \sum_{i \in J_j} \frac{C_{ij} Y_i}{\sum_{k \in I_i} C_{ik} \lambda_k^n} , \quad (\text{B.6})$$

where J_j are the set of detectors to which image pixel j contributes.

References

- [1] M. Galloway and other, Nucl. Instrum. Methods A **652**, 641 (2011).
- [2] P. N. Luke, IEEE Trans. Nucl. Sci. **4**, 207 (1995).
- [3] S. Agostinelli *et al.*, (GEANT4 Collaboration), Nucl. Instrum. Methods A **506**, 250 (2003).
- [4] R. Dicke, The Astrophysical Journal **153**, 101 (1968).
- [5] E. E. Fenimore and T. M. Cannon, Applied Optics **17**, 3 (1978).
- [6] K. Lange and R. Carson, Journal of Computer Assisted Tomography **8(2)**, 306 (1984).
- [7] C. Wahl, Ph.D. Thesis, University of Michigan (2011).
- [8] L. A. Shepp and Y. Vardi, IEEE Trans. Med. Imag. **1**, 113 (1982).
- [9] N. Bissantz, B. A. Mair, and A. Munk, IEEE NSS Conference Record, 3376 (2006).

- [10] C. E. Lehner, Ph.D. Thesis, University of Michigan (2004).
- [11] A. Haefner, D. Gunter, R. Barnowski, and K. Vetter, IEEE Trans. on Nucl. Sci. **62**, 1911 (2015).
- [12] K. Górska *et al.*, The Astrophysical Journal **662**, 759 (2005).
- [13] R. Barnowski, A. Haefner, L. Mihailescu, and K. Vetter, Nucl. Instrum. Methods A **800**, 65 (2015).