Fast Parameterizations for Ray Tracing in Optical Calorimetry

Michael Bowler

May 11, 2023

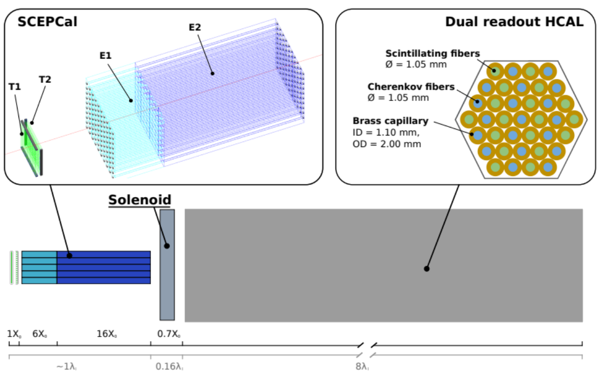

Figure 1: The Calvision detector, from Lucchini et al. (2020)

1 Introduction

Tracing optical photons produced by scintillation and Cherenkov processes in electromagnetic calorimeters is an important component of particle detector design. Unfortunately, propagating optical photons using standard ray tracing algorithms is computationally expensive, prohibitively so for modeling a large detector system. This work discusses a method for creating fast parameterizations that can replace a large fraction of the ray tracing necessary to model scintillation photons in an electromagnetic calorimeter, leading to substantial time savings while accurately reproducing the results obtained via full ray tracing.

2 The Calvision Detector

This project is motivated by the Calvision detector concept, shown in Figure 1. The detector consists of an electromagnetic calorimeter (ECAL) section (labeled SCEPCal in the diagram) which consists of two timing layers of a fast scintillator and two segments of homogeneous lead tungstate crystal with silicon photomultipliers (SiPMs) positioned at the end. The ECAL detects electrons, positrons, and gammas and has dual-readout capabilities, meaning that it is designed to separate the scintillation and Cherenkov components of the collected light. The Cherenkov light component comes from ultra-relativistic particles and is proportional to the electromagnetic component of the showers produced by incoming particles. Dual-readout techniques allow better energy resolution for hadronic showers, which is a major motivation for this detector design.

The detector also includes a dual-readout hadronic calorimeter (HCAL), which detects hadrons. This work focuses exclusively on scintillation photons within the ECAL, and in particular the E2 crystal.

3 Modeling Optical Photons

The primary question of interest for modeling the performance of the ECAL is: given a particle or set of particles incident on the E2 crystal, what is the distribution in time of scintillation photons arriving at the SiPM? The main tool for answering this question is the simulation package Geant4, which is able to model incident particles traversing the crystal as well as trace the resulting scintillation photons. Geant4 uses ray tracing algorithms to model these optical photons. These methods take each individual photon and propagate it through the crystal while accounting for the medium's properties, including transmission and reflection coefficients at boundaries, absorption, scattering, index of refraction, and more. As mentioned previously, a major drawback to ray tracing is that it is computationally demanding. Detector performance studies require modeling millions of optical photons, which becomes unreasonably time consuming when done via full ray tracing.

The idea behind the fast parameterization method is to use Geant4 to simulate a large sample of individual scintillation photons uniformly distributed in wavelength and initial longitudinal position within the crystal. We record information about whether each photon is detected and, if so, at what time it is detected, and then use this information to build models for the probability of detection and distribution of travel time until detection for the scintillation photons given their initial longitudinal position and wavelength. Having built this parameterized model, we run a Geant4 simulation of the incident particles we are interested in (e.g. electrons or muons of a certain energy) with ray tracing disabled. Instead, we have Geant4 record the information about the magnitude, time, and position of scintillating energy depositions and use the parameterized models and our knowledge of the physical properties of the medium to generate the expected distribution in time of photons arriving at the SiPM.

Finally, we compare the distribution of arrival times generated by a full Geant4 simulation to the distribution generated by the fast parameterization on the same events in order to verify that the fast parameterization is consistent. We also check to see the degree of time savings realized by the fast method.

4 Building the Fast Parameterization

Building the fast parameterization requires simulating a large number (~10 million) single optical photons traversing the crystal. The two variables that affect an optical photon's probability to be detected and time of travel if detected are its wavelength, λ, and its initial longitudinal position within the crystal, z. The effects of the other two spatial dimensions on the photon detection probabilities and travel times are negligible due to symmetry. Furthermore, it is assumed that scintillation photons are emitted isotropically, which means that the direction of emission of the scintillation photon can be ignored.

The simulated single photons are generated uniformly in z between 217.5 mm and 397.5 mm, which are the start and end positions of the E2 crystal in our model. Their wavelengths are also uniform between 300 nm and 1000 nm. For data recording purposes, the wavelength axis is divided into 70 bins of width 10 nm and the z axis is divided into 18 bins of width 1 cm. For each photon, Geant4 records whether the photon reached the SiPM or not and, if so, the time from its creation until it was detected. (Note that we are assuming that all photons that reach the SiPM are detected. This assumption is unrealistic and can be adjusted to match real SiPM parameters.)

The photon probability of detection P_{det} for each wavelength and z-position bin is estimated as the fraction of photons generated in that bin that are detected:

$$P_{\text{det}}(z_i, \lambda_j) = \frac{N_{\text{det}}(z_i, \lambda_j)}{N_{\text{gen}}(z_i, \lambda_j)} \tag{1}$$

where $N_{\text{gen}}(z_i, \lambda_j)$ is the number of photons generated in the bin and $N_{\text{det}}(z_i, \lambda_j)$ is the number of photons detected in the bin. A plot of these detection probabilities generated on a sample of 11 million photons is shown in Figure 2. Note that below about 350 nm, photons have no probability to be detected due to lead tungstate being opaque in that wavelength range. In addition, photons starting closer to the SiPM have a higher probability to be detected.

For recording the travel time distributions, the time axis ranges from 0 ns to 10 ns and is divided into 100 bins of 0.1 ns width. The time of travel distribution f_{travel} for each wavelength and position bin is given by the distribution of travel times for photons in each bin divided by the number of photons detected in that bin:

$$f_{\text{travel}}(z_i, \lambda_j, t_k) = \frac{N_{\text{det}}(z_i, \lambda_j, t_k)}{N_{\text{det}}(z_i, \lambda_j)} \tag{2}$$

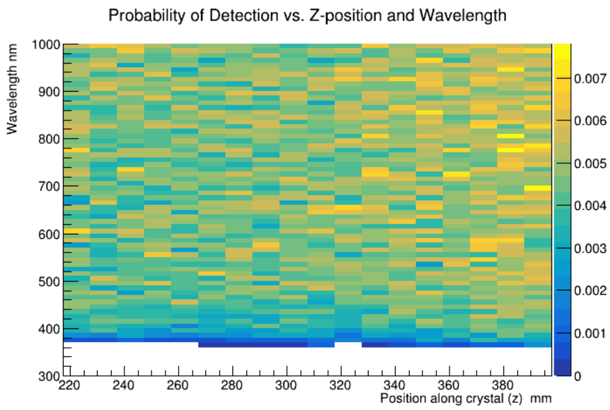

Figure 2: Probability of detection of a scintillation photon as a function of initial longitudinal position and wavelength

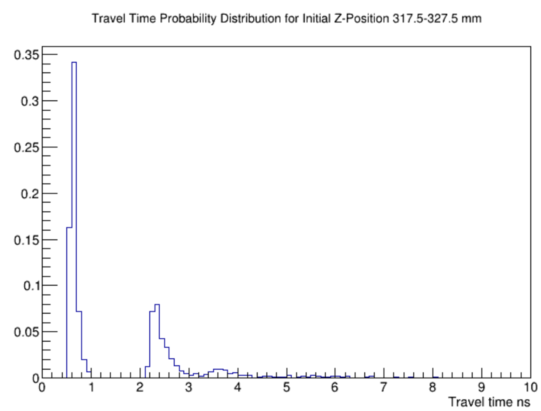

Figure 3: Probability distribution of travel times for detected photons originating between 317.5 and 327.5 mm, combining all wavelengths

An example of the travel time distribution for a particular z-bin and incorporating all wavelengths is shown in Figure 3. We can see that there is one large peak of photons detected at less than 1 ns, which comes from scintillation photons that travel directly to the SiPM and are absorbed. There is also a second peak between 2 and 3 ns from photons that travel to the far end of the crystal, reflect, and then are absorbed by the SiPM. There are a very small number of photons that reflect multiple times and take even longer to reach the SiPM.

5 Applying the Fast Parameterization

The built fast parameterization can now be combined with data on energy depositions from Geant4 and information about the physical properties of the lead tungstate crystal to produce an expected distribution of photon arrivals given an incident particle without doing any additional ray tracing. In this section we will as an example consider the total scintillation photon arrival time distribution produced by 500 muons of energy 10 GeV each incident on the crystal. For each muon, we use Geant4 to simulate its path through the crystal and record information about the series of energy depositions it produces, which are referred to as hits. For each hit, indexed by l, we record the hit energy (E_l) in MeV, the hit z-position (z_l) in mm, and the hit time (t_l) in ns. The procedure for generating the expected photon arrival times is as follows:

1. Loop over all of the events (i.e. each muon).

2. For each event, loop over all hits generated by this event. For each hit l, calculate the expected number of scintillation photons n_l from this hit as the hit energy times the scintillation yield for lead tungstate, which is assumed to be 450 photons/MeV:

$$n_l = 450E_l \tag{3}$$

   Figure 4 shows the hit energies collected by Geant4. The average hit energy is 0.2 MeV.

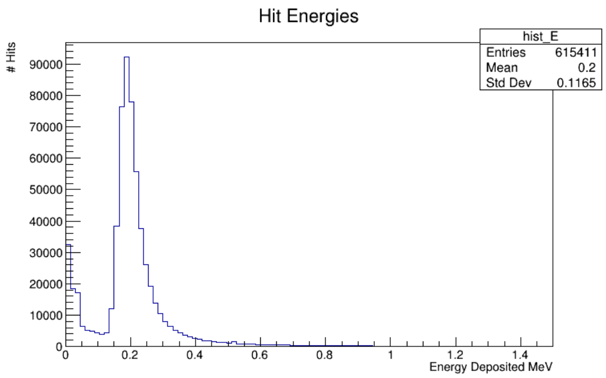

Figure 4: Hit energies of all hits collected by Geant4 over 500 10 GeV muons

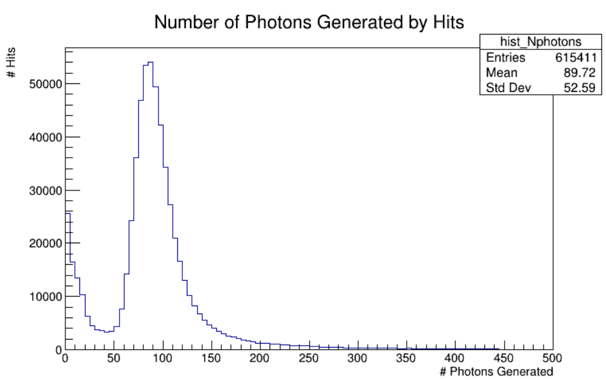

Figure 5: Number of photons generated by each hit in the fast parameterization

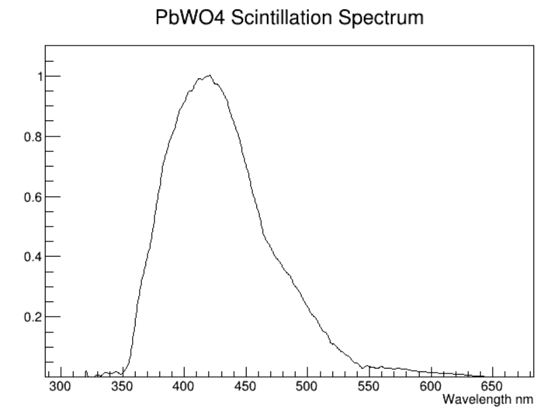

Figure 6: Scintillation spectrum of lead tungstate

3. Sample randomly around this expected number of photons to get the actual number of scintillation photons N_l generated by this hit. If $n_l < 100$ sample according to a Poisson distribution:

$$N_l = \text{random.Poisson}(n_l) \tag{4}$$

Otherwise, sample according to a Normal distribution:

$$N_l = \text{(int) random.Normal}(n_l, \sqrt{n_l}) \tag{5}$$

Figure 5 shows the number of photons generated by each hit in the fast parameterization. On average about 90 scintillation photons are emitted per hit.

4. Loop over all N_l emitted photons for this hit. For each photon, assign it a wavelength λ_γ randomly according to the scintillation spectrum $f_{\text{scint}}(\lambda)$ of lead tungstate:

$$\lambda_\gamma = f_{\text{scint}}.\text{GetRandom}() \tag{6}$$

This spectrum is shown in Figure 6.

5. Using the hit position z_l and photon wavelength λ_γ, calculate the probability of detection for this photon using the fast parameterization as $P_{\text{det}}(z_l, \lambda_\gamma)$. Draw a uniform random number

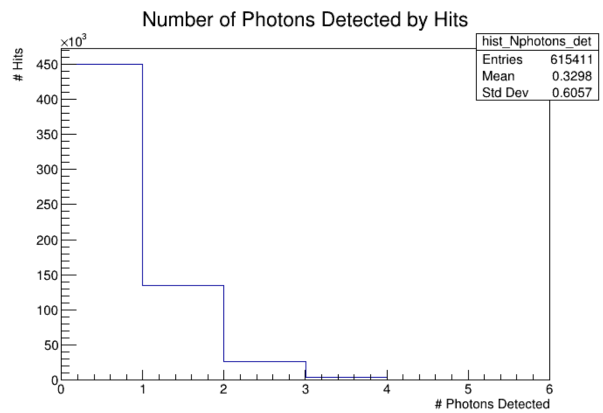

Figure 7: Number of photons detected from each hit in the fast parameterization

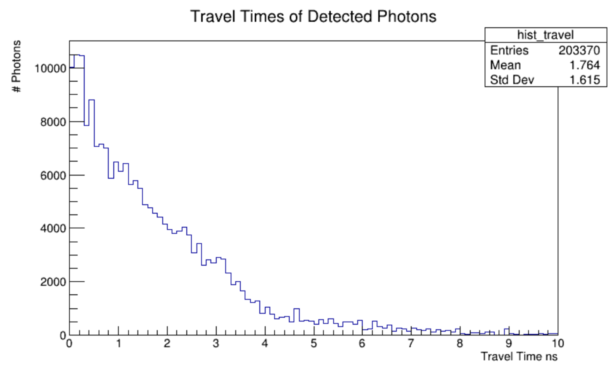

Figure 8: Travel times of detected photons in the fast parameterization

between 0 and 1. If the random number is less than this probability of detection, proceed. Otherwise, the photon is considered not detected and we skip to the next photon. Figure 7 shows that number of photons detected from each hit. On average only about 0.33 photons per hit are being detected, which is a reflection of the very low detection probability for any given photon.

6. Generate a random travel time t_{travel} for the detected photon using the corresponding distribution from the fast parameterization:

$$t_{\text{travel}} = f_{\text{travel}}(z_l, \lambda_\gamma).\text{GetRandom}() \qquad (7)$$

Figure 8 shows the distribution of travel times of all detected photons after sampling their travel times from the fast parameterization.

7. Generate the emission delay time t_{decay} for this photon by randomly sampling from the scintillation decay distribution for lead tungstate $f_{\text{decay}}(t)$ given by:

$$f_{\text{decay}}(t) = \frac{0.3}{5}e^{-t/5} + \frac{0.7}{15}e^{-t/15} \qquad (8)$$

$$t_{\text{decay}} = f_{\text{decay}}.\text{GetRandom}() \qquad (9)$$

This corresponds to a scintillation decay distribution which is 30% a 5 ns decay component and

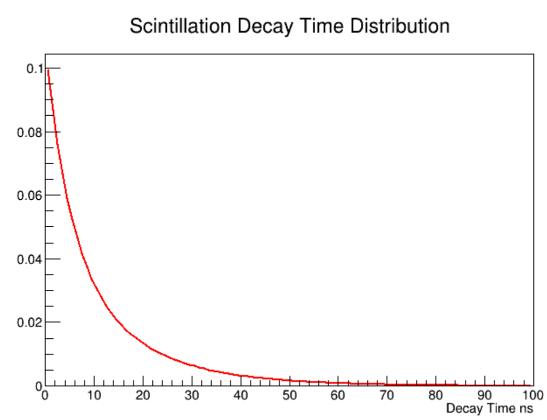

Figure 9: Scintillation decay time distribution of lead tungstate

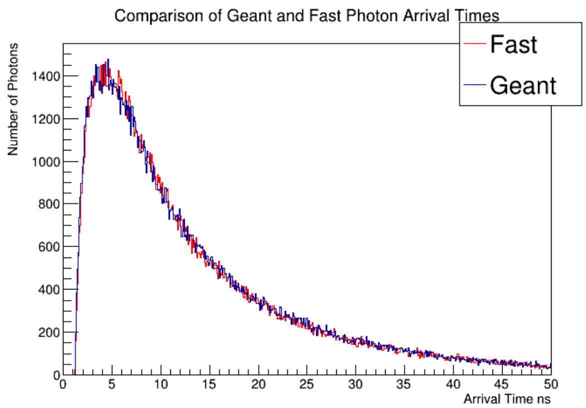

Figure 10: Comparison of scintillation photon arrival times at SiPM generated by Geant4 and the fast parameterization

70% a 15 ns decay component. It is plotted in Figure 9.

8. The arrival time t_{arrival} of this photon is the sum of the hit time, scintillation decay time, and travel time:

$$t_{\text{arrival}} = t_l + t_{\text{decay}} + t_{\text{travel}} \tag{10}$$

We add t_{arrival} to the final distribution of arrival times with weight 1.

9. Continue to the next photon, or next the hit if all photons for the current hit have been accounted for, or the next event (i.e. next muon) if all hits in the current event have been accounted for. Stop when all events have been processed.

The distribution of photon arrival times output by this procedure can then be compared to the distribution of photon arrival times generated by full ray tracing in Geant4. The two distributions are calculated using the exact same set of events and hits. Figure 10 shows an overlay of the Geant4 and fast distributions generated on one set of 500 10 GeV muons. The agreement between the two distributions is excellent. The fast parameterization therefore successfully reproduces the results obtained through full ray tracing.

Step	**Time Required**
Full Geant4 simulation for 500 muon events	59 min
Geant4 ray tracing of 11 million single optical photons	16 min 45 s
“Hit only” Geant4 simulation for 500 muon events	0.23 s
Processing of hit information into arrival times for 500 muon events	28 s
Total for fast parameterization	16 min 45 s one-time + 28.2 s for every 500 muon events

Table 1: Time needed to complete the various steps in each method for 500 10 GeV muons

Table 1 shows information about the time required to run each method on Rivanna for 500 10 GeV muons. The fast parameterization method takes about 30% of the time of the full Geant4 simulation for this number of events, which is a significant time reduction.

A key benefit of the fast parameterization is that the single optical photon simulation only needs to run once for each detector geometry. As long as the detector geometry and materials remain the same, the single optical photon data can be reused with the hit data generated by any number and type of events. Assuming that the single optical photon simulations have been generated previously and are being reused, running a new set of 500 muons through “hit only” Geant4 and processing the hits using the fast parameterization takes only 28.2 seconds, versus nearly an hour to run full Geant4 on the 500 events. In this case the fast method takes about 0.8% the time of running full Geant4, which is a huge reduction in the time required.

At the same time, the models are flexible enough that regenerating them with different detector geometries and material properties is not difficult. For example, it would be easy to compare the performance of the detector with and without optical silicone grease in the gap between the crystal and the SiPM by performing the single photon simulations for both configurations and comparing the results of running the same set of hit data through both models.

6 Second Method for Fast Parameterization

It is also possible to build a parameterized model that replaces ray tracing without needing to draw so many random numbers. In order to build this model, we return to the single optical photon simulations and for each time and z-position bin take a weighted average over all wavelengths of the number of single optical photons detected in that bin. The weights are given by the scintillation spectrum, which is normalized to one. This is expressed as follows:

$$N^{\mathrm{int}}_{\mathrm{det}}(z_i, t_k) = \sum_{\lambda_j} N_{\mathrm{det}}(z_i, \lambda_j, t_k) f_{\mathrm{scint}}(\lambda_j) \qquad (11)$$

where $N^{\mathrm{int}}_{\mathrm{det}}(z_i, t_k)$ is the number of single optical photons detected in bin (z_i, t_k) after integrating over all wavelengths weighted by the scintillation spectrum.

The next step is to convolve the distribution in time of detected photons in each position bin with the scintillation decay time distribution:

$$F_{\mathrm{conv}}(z_i, t_m) = \sum_{t_k} N^{\mathrm{int}}_{\mathrm{det}}(z_i, t_k) f_{\mathrm{decay}}(t_m - t_k) \qquad (12)$$

Here $F_{\mathrm{conv}}(z_i, t_m)$ represents the unnormalized distribution of the sum of the photon travel and scintillation decay times for each longitudinal position bin. Each possible value of the sum of the travel and decay times is given by a t_m, while each possible value of the travel time alone is denoted by a t_k. The capital F is a reminder that this distribution is not normalized yet.

In the last step we normalize F_{conv} within each layer z_i such that the normalization equals the probability of detection for a photon originating in that layer after integrating over the scintillation spectrum. This can be expressed as:

$$f_{\mathrm{conv}}(z_i, t_m) = \frac{F_{\mathrm{conv}}(z_i, t_m)}{\sum_{t_m} F_{\mathrm{conv}}(z_i, t_m)} \frac{N^{\mathrm{int}}_{\mathrm{det}}(z_i, t_k)}{N^{\mathrm{int}}_{\mathrm{gen}}(z_i, t_k)} \qquad (13)$$

where $N^{\mathrm{int}}_{\mathrm{gen}}(z_i, t_k)$ is calculated by the exact same weighting procedure as for $N^{\mathrm{int}}_{\mathrm{det}}(z_i, t_k)$ in Eqn. 11 and $f_{\mathrm{conv}}(z_i, t_m)$ is the distribution, normalized to the weighted probability of detection for a photon in position bin z_i, of the sum of photon travel and decay times for that position bin. In other words, $f_{\mathrm{conv}}(z_i, t_m)$ is the probability that a photon originating in position bin z_i has the sum of its scintillation decay and travel to detector times equal to t_m.

Performing these manipulations obviates the need to draw random numbers for the wavelengths, travel times, and scintillation decay times of individual photons. We also choose to use the average number of photons expected to be emitted from each hit rather than performing a random fluctuation around the average. Instead of drawing randoms numbers, we take a sum over all hits of the $f_{\mathrm{conv}}(z_i, t_m)$ distributions shifted by their hit times and weighted by their expected numbers of emitted photons. The procedure is now as follows:

1. Loop over all of the events.
2. For each event, loop over all hits generated by this event. For each hit l, calculate the expected number of scintillation photons n_l from this hit as the hit energy times the scintillation yield for lead tungstate (see Eqn. 3).
3. Add the distribution $n_l f_{\mathrm{conv}}(z_l, t_m - t_l)$ to the final distribution of arrival times. Note that since for each z_i, $f_{\mathrm{conv}}(z_i, t_m)$ is normalized to the wavelength-weighted probability of detection of a photon, multiplying by the expected number of photons generated by this hit will result in a distribution that is normalized to the expected number of detected photons for this hit. We also shift the $f_{\mathrm{conv}}(z_i, t_m)$ distribution in time by the hit time in order to account for the hit occurring later than time zero.
4. Continue to the next hit, or the next event (i.e. next muon) if all hits in the current event have

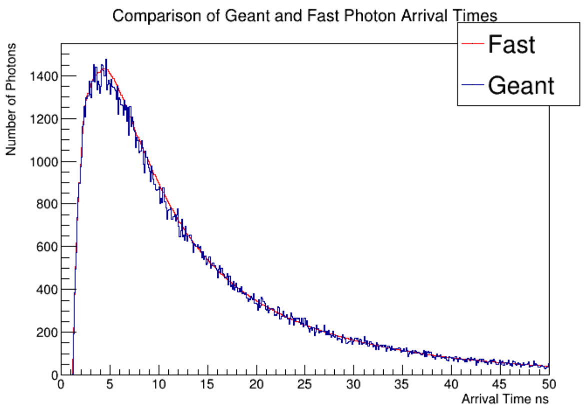

Figure 11: Comparison of scintillation photon arrival times at SiPM generated by Geant4 and the second fast parameterization

been accounted for. Stop when all events have been processed.

Figure 11 shows a comparison of the photon arrival time distribution generated by Geant4 and that obtained from the second method for fast parameterization. These are again built on 500 10 GeV muons. The agreement is once again excellent, with the main difference being that the fast distribution is now much smoother than before due to the lack of drawing random numbers.

The only difference in the time needed to run the second method, referring to Table 1, is in the "processing of hit information into arrival times" step, which now takes 14 s for 500 muons events rather than 28 s. The second fast parameterization method therefore offers significantly greater time savings than the first method, cutting the time needed to process the hit and single photon information into an arrival time distribution in half.

In summary, the last three sections have demonstrated that using fast parameterizations to replace full ray tracing in simulations both reproduces the results obtained from full ray tracing and does so in much less time.

7 Future Work

There are several aspects of this method that could be explored and improved in the future. One would be the sensitivity of the method to applying a cut on the energy of the hits that are recorded by Geant4. If hits below 0.05 GeV or some other cutoff were to be discarded, we might be able to trade some acceptable amount of bias in the resulting expected photon arrival distribution for reduced memory usage and time needed to process a smaller number of hits.

The method is already flexible enough to model non-ideal light detectors if desired. If the detector has some photon detection efficiency as a function of wavelength then this can be incorporated into the parameterized models by multiplying this efficiency function with the wavelength-dependent probabilities of photon detection. There is no need to regenerate the single optical photon files. This feature is helpful for characterizing and comparing light detectors with different efficiencies.

An additional area of interest is finding a way to apply this method to Cherenkov photons. Cherenkov photons must be handled differently than scintillation photons because they are not emitted isotropically, but rather at an angle determined by the velocity of the emitting particle relative to the speed of light in the medium. The assumption that photons are emitted isotropically was important for building the fast parameterizations for scintillation light. It might be possible to build a similar parameterization where some information about the angle of emission of the photons is also recorded, but this would require significantly more single photons to be simulated in order to sample the parameter space adequately, which would cost more time.

Another approach that could complement these fast parameterizations is GPU-accelerated ray tracing. GPUs (graphics processing units) are specialized hardware designed to perform ray tracing quickly, and packages such as CaTS+Opticks exist that allow Geant4 to offload optical photon ray tracing to a GPU to speed up simulation. It would be interesting to compare the time needed for GPU-accelerated ray tracing to that needed for the fast parameterization. A hybrid approach could also be developed where scintillation photons are handled by the fast parameterization and Cherenkov photons are handled by full Geant4 with or without GPU-acceleration. Since fewer Cherenkov photons are produced than scintillation photons this could be a feasible approach even if the Cherenkov component is not done with GPU-acceleration.

In all cases, the ultimate goal is to be able to quickly and efficiently characterize the detection of both scintillation and Cherenkov light in order to apply dual-readout techniques.

8 Conclusion

This work has demonstrated that fast parameterization methods of the kind presented can substantially reduce the time needed to model scintillation photons propagating through a detector. Further work with GPU acceleration could also advance the efficiency of simulations and could potentially be combined with these fast parameterizations for even greater performance gains. Developing a framework such as this one to reliably calibrate and apply fast simulations can be very useful in detector performance studies. In turn, understanding the properties of optical calorimeters is an important element of detector design in collider and neutrino applications, dark matter searches, and other areas. Fast parameterizations for optical ray tracing can therefore be a useful tool for advancing these physics goals.

9 Acknowledgements

I would like to thank Bob Hirosky for advising me on this project. I would also like to thank Alexander Ledovskoy and Hayden Hollenbeck for their help.