Rhett Cheek
PHYS 3995 report, Fall Semester, 2016
Faculty supervisor: Professor Xiaochao Zheng

[Figures 11-15 added on 2-15-2017]

This fall I worked under Professor Zheng on several C++ and ROOT programs of hexagonally-shaped scintillators with wavelength-shifting fibers (all of which have a circular cross-section and a 1mm diameter). The overarching objective of these programs is to simulate an arbitrarily high number of scintillating photon events (photons are generated with random position and direction vector) and acquire absorption efficiencies of the scintillators based on the reflectivity of their outer surfaces.

There are two types of scintillators that I modeled: the preshower scintillator and the shower scintillator. Each has the profile of a regular hexagon with side-lengths of 6.25cm, and they both have heights which retain identical hexagonal dimensions at both the base and the ceiling; the preshower scintillator has a height of 2.0cm and the shower has a height of 0.15cm.

The placements of the fibers in the two scintillator types differ significantly from one another, however. The preshower scintillator has one wavelength-shifting fiber spun around in a circle inside, in four loops, stacked atop one another. The stack of loops is spun at a radius of 4.5-4.6cm from the center of the hexagon, and the fiber tapers out of the scintillator from two points tangential to the loop. The shower scintillator, on the other hand, has 96 fibers that perforate it in straight lines from its floor to its ceiling, arranged symmetrically in a hex-pattern throughout.

The figures on the following pages illustrate this. However, one point of interest for the image of the shower (Fig. 3) scintillator is that the fiber at the very center, the larger holes near the corners of the hexagon, and the small hole on the lower right are used for alignment purposes only, and do not have wavelength-shifting fibers running through them.

My first task was to familiarize myself with hexagonal geometry, as the scintillators' design made such knowledge essential. With that in mind, the first C++ program I wrote was one which had the side-lengths of the preshower scintillator written in with a coordinate system set up such that the very

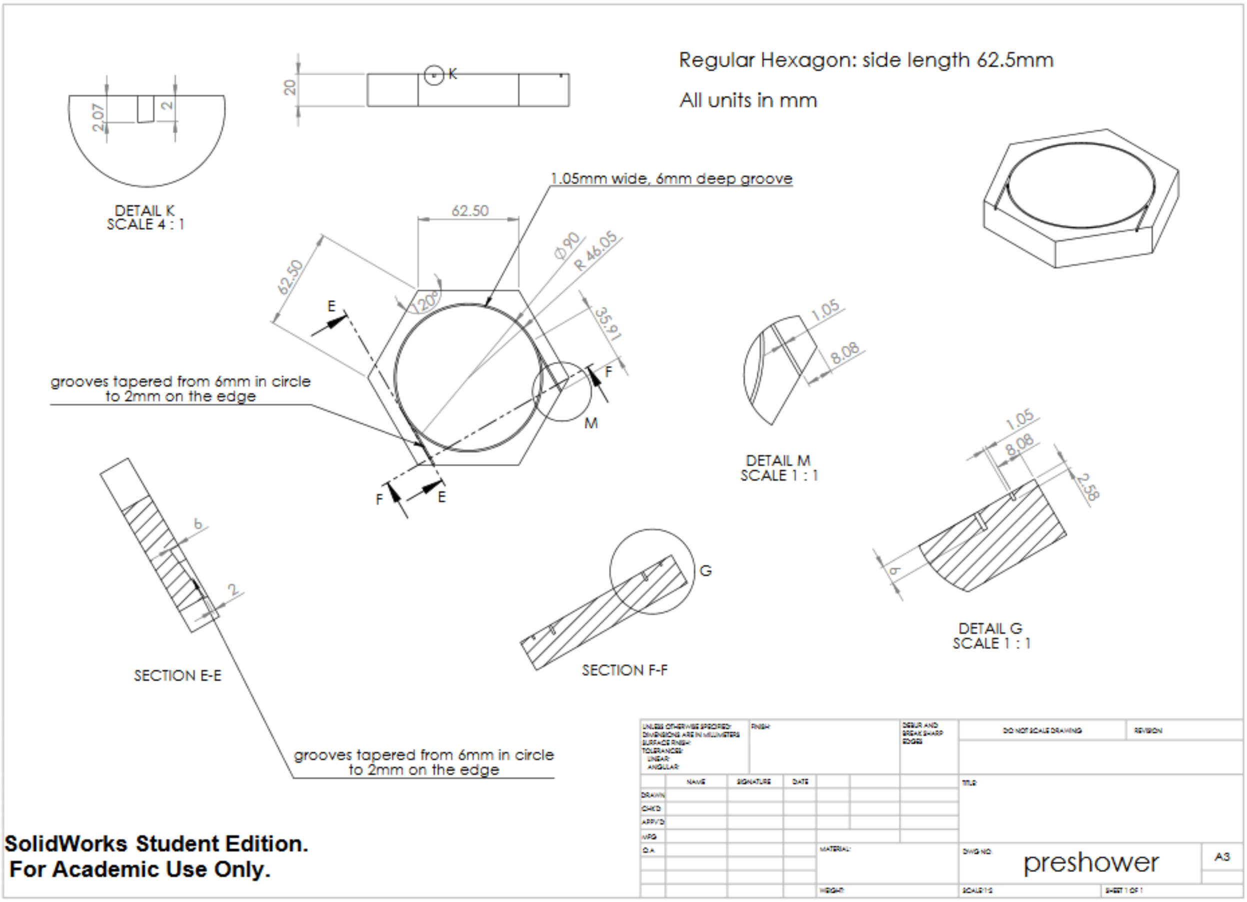

Fig. 1: Preshower cross-sections.

center of the hexagon is defined as the origin of the coordinate system, and the base of the scintillator is defined as z=0 (the range of possible z-values is [0, height of scintillator]). Next, the user would input values for the x, y, and z coordinates from the command line and the program would display whether or not the point given is inside or outside the scintillator.

Checking whether or not the z-value is inside the scintillator is a simple enough exercise as the value must be between 0 and the height of the scintillator. However, checking whether or not the x and y coordinates are within the bounds of the hexagon is a little more complicated, and knowledge of hexagonal geometry is crucial. With the hexagon's orientation as shown in Fig. 2, it is clear

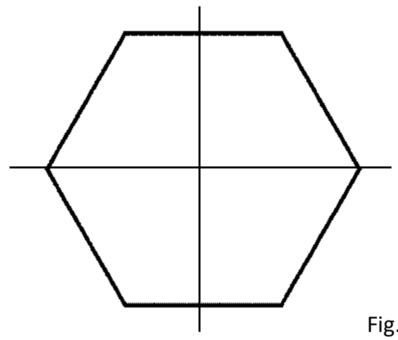

Fig. 2

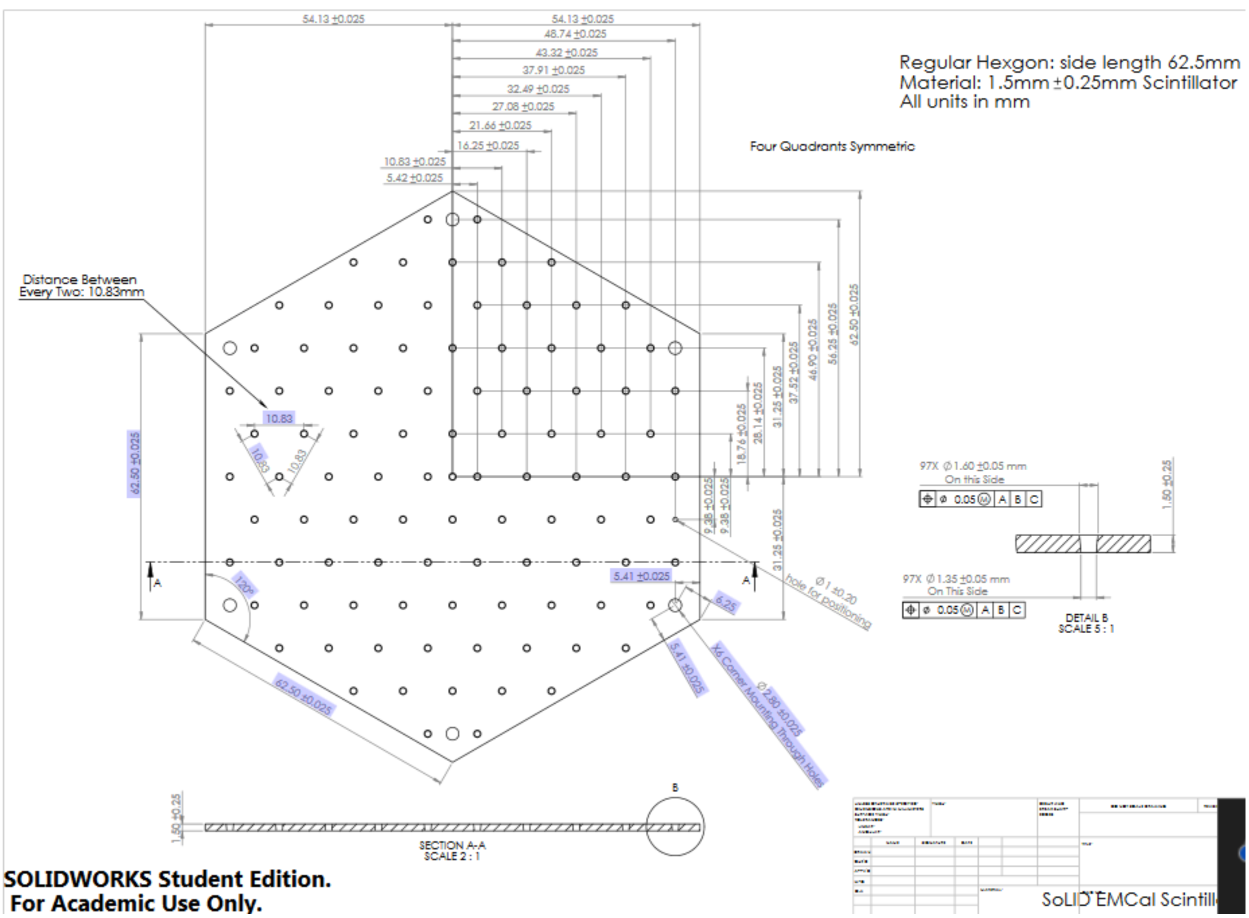

Fig. 3: Shower scintillator cross-sections.

that a regular hexagon has two major radii, which are generally dubbed the "outer radius" and the "inner radius". The outer radius (the distance from the center to either of the corners along the horizontal axis) is equal to the side-length of the hexagon, and the inner radius (the distance from the center to either the base or ceiling along the vertical axis) is directly proportional to the outer radius:

$$R_{outer} = 6.25cm$$

$$R_{inner} = \left(\frac{\sqrt{3}}{2}\right) * R_{outer}$$

In the case of both scintillators, R_{outer} = 6.25cm and R_{inner} =~ 5.413cm. With this in mind, it is clear that the range of the x-values is [-6.25, 6.25] and the range of the y-values is [-5.413, 5.413]. This is usually accurate, however there are points in this xy range where it is false: those that are near the corners of the rectangle with (x, y) coordinates $(\pm R_{outer}, \pm R_{inner})$ will be outside the hexagon.

It is at this point that recalling the point-slope form of the equation of a line is helpful. One can describe the 4 sides that are not parallel to the x-axis and determine whether or not a given set of (x,y) coordinates are within the hexagon. The lines describing the upper-right line, the lower-right line, the upper-left line, and the lower-left line are respectively:

$$y = -2(x - R_{outer}), y = 2(x - R_{outer}), y = 2(x + R_{outer}), y = -2(x + R_{outer})$$

Rearranging each of these shows that, for any x and y entered by the user, the point is OUTSIDE the hexagon if any of the below expressions are true:

$$y + 2x > 2 * R_{outer}, y + 2x < -2 * R_{outer}, y - 2x > 2 * R_{outer}, y - 2x < -2 * R_{outer}$$

Next, I wrote a C++ program which was designed to simulate an arbitrarily high number of events where a photon, generated with a random position and direction vector in 3-space, would travel and reflect throughout the scintillator until it escaped. The main physical law that this program simulated was the Law of Reflection, which states that when reflected, a light beam will rebound from the surface at the same angle relative to the normal vector of the surface, while remaining within the plane of both the initial direction vector and the normal of the surface it reflected from.

Another physical property to consider is total internal reflection, which states when a photon is going from some medium to a medium with a lower refractive index, there is a critical angle where the chance of transmission is incredibly low for any angle relative to the normal of the surface that is greater than or equal to that critical angle. As the scintillator is composed of plastic material (which has a refractive index of roughly 1.5) and the medium it would escape into is air (which has a refractive index approximately equal to 1), the formula for the critical angle is:

$$\varphi_{critical} = \arcsin(\frac{1}{n_{sci}})$$

Now, the scintillator has 8 planar surfaces that a photon might reflect off of: the six sides of the hexagon, the base, and the ceiling. Fortunately, there is a way to quickly determine which planar surface the photon will reflect off.

This is done as follows:

$$d * \boldsymbol{l} + \boldsymbol{l_0}$$

describes the point of intersection between a line and a plane where $\boldsymbol{l_0}$ is the initial point of the line, $\boldsymbol{l}$ is the direction vector of the line, and d is a scalar given by:

$$d = \frac{(\boldsymbol{p_o} - \boldsymbol{l_o}) \cdot \boldsymbol{n}}{\boldsymbol{l} \cdot \boldsymbol{n}}$$

where $\boldsymbol{p_o}$ and $\boldsymbol{n}$ are a sample point lying on the plane of interest and the unit normal vector of the planar surface, respectively. These equations are derived from the algebraic definitions of lines and planes[1].

Since a line will have one intersection with a plane (unless they are parallel), one can determine which surface the photon will impact by comparing the distances to collision for each. As a general rule, the surface with the shortest distance will be the surface that the photon impacts. The distance calculation, once the point of collision is known is, of course:

[1] These formulae were found at https://en.wikipedia.org/wiki/Line%E2%80%93plane_intersection

$$distance = \sqrt{(x' - x)^2 + (y' - y)^2 + (z' - z)^2}$$

However, one condition to consider is that this equation does not recognize which direction along the line the path leads. For example, if the photon is generated at the origin and center of the scintillator at an angle that leads to the ceiling, the calculation would give the distance to both the floor and ceiling as equal. This issue, however, can be circumvented by defining the angles where each surface is able to be impacted.

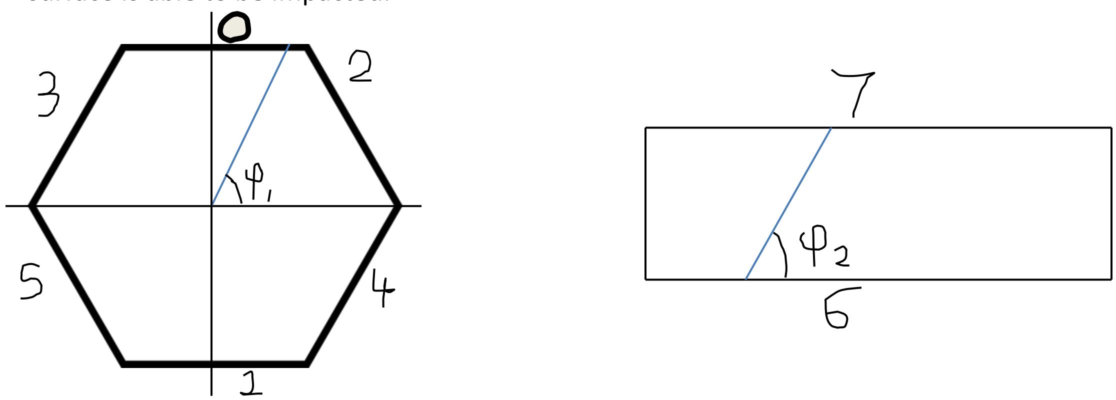

Fig.'s 4, 5: cross-sections along the horizontal and vertical plane respectively; each side is assigned the same designation as in my code.

Let's define the angle that the blue line makes with the horizontal axis in Fig. 4 as "ϕ1", (can be related to the azimuthal angle from spherical coordinates (r,θ,ϕ)). If the photon's path is along the blue line shown (beginning at the origin), the calculation I used would give equal distances to collision with both the uppermost boundary and the lowermost boundary. However, no matter what the point of origin is (provided it is inside the hexagon, of course), if ϕ1 is greater than 0 and less than π radians, it is incapable of hitting the lowermost boundary. Likewise, the uppermost boundary cannot be impacted if ϕ1 is greater than π and less than 2π radians. All of the other boundary lines have a similar constraint, making the calculations for which boundary the photon reflects feasible from a coding standpoint.

Angles in 3-space may be expressed in their totality as a set of two 2-dimensional angles (ϕ1 and ϕ2) or as a direction vector (cos(ϕ1)*cos(ϕ2), sin(ϕ1)*cos(ϕ2), sin(ϕ2)), where ϕ1 can have any value within [0,2π] and ϕ2 can have any value [0, π/2]U[3π/2, 2π] (I defined ϕ2 as the angle between the path of the photon and a plane that is parallel to the floor and ceiling of the scintillator; this is $\frac{\pi}{2}$ minus the usual polar angle in spherical coordinates and can be seen in Fig. 5). Using ϕ1 and ϕ2 is simpler in some calculations, namely when a photon reflects off a boundary.

The scintillator has eight planar boundaries that a photon might reflect off; in my C++ programs I defined the boundaries thusly: the six sides of the hexagon are boundaries 0,1,…5, the floor is boundary 6, and the ceiling is boundary 7 (I began at 0 to make coding its calculations a little simpler; I defined an array with 2 dimensions, where the boundary number was the first and its sample point was the second. Recall that calculating the intersection of a line in 3-space and a plane requires the sample point and a normal vector for the plane). The coordinates in 3-space for a point on each planar boundary are (in the order boundary 0,1,…7 respectively):

(0, R_{inner}, 0) [uppermost boundary of hexagon],

(0, -R_{inner}, 0) [lowermost boundary of hexagon],

($\frac{R_{outer}}{2}$, R_{inner}, 0) [upper-right side of the hexagon],

($-\frac{R_{outer}}{2}$, R_{inner}, 0) [upper-left side],

($\frac{R_{outer}}{2}$, $-R_{inner}$, 0) [lower-right side],

($-\frac{R_{outer}}{2}$, $-R_{inner}$, 0) [lower-left side],

(0, 0, 0) [floor],

(0, 0, height) [ceiling]

In addition, the normal vectors of these planes are (in the same order as above):

(0, -1, 0), (0, 1, 0),

$(-\sqrt{3}/2$, -1/2, 0), ($\sqrt{3}/2$, -1/2, 0),

$(-\sqrt{3}/2$, 1/2, 0), ($\sqrt{3}/2$, 1/2, 0),

(0, 0, 1), (0, 0, -1)

If the photon reflects from any of the hexagonal boundaries, the angle defined earlier as ϕ1 will change, and if it reflects off either the ceiling or the floor, the angle defined and ϕ2 will change. While the Law of Reflection seems very straightforward (the reflected angle is the same size as the incident angle), keeping ϕ1 and ϕ2 internally consistent in the code makes this a little more complicated. ϕ1 and ϕ2 need to remain within the ranges [0,2π] and [0,π/2]U[3π/2,2π] respectively, but the reflections from the boundaries are such that, if calculated without the precaution that the new ϕ1 and ϕ2 remain within their respective ranges, the new angles would make future calculations incorrect.

I'll elucidate this with two examples: the ceiling of the scintillator and the wall on the top-right of the hexagon. The ceiling's (as well as the floor's) incident angle is dependent solely on what I've defined as ϕ2. After a reflection, the angle ϕ2 simply changes to –ϕ2, but to keep all angles positive I'd also add 2π, giving a final expression of:

$$\varphi 2' = 2\pi - \varphi 2$$

The top-right wall of the hexagon, however, is a bit more complicated. The possibility for a photon to reflect off that boundary is geometrically constrained by ϕ1 ϵ [0,2π/3]U[5π/3,2π]. The normal vector of this surface makes an angle of π/6 radians with the horizontal axis. Therefore, if ϕ1 is less than 2π/3,

$$\varphi 1' = \frac{4\pi}{3} - \varphi 1$$

and if ϕ1 is greater than 2π/3,

$$\varphi 1' = \frac{10\pi}{3} - \varphi 1$$

In this instance, the incident angle is not $\phi 1$ as it has been previously defined, and each of the other 4 diagonal sides of the hexagon share similar reflection calculations (the uppermost and lowermost sides of the hexagon do, however, have the simple $\phi 1' = -\phi 1$). The angle along the vertical axis ($\phi 2$) will not be changed during a reflection along this boundary as there is no slope along that axis, just as the ceiling and floor do not change the angle along the horizontal axis ($\phi 1$).

After coding the model for the boundary reflections, I did a study on how often photons with randomly-generated initial positions and directions would impact each boundary using ROOT; the following pages will display the results of this. Fig. 6 will show the shower scintillator's results boundary-by-boundary, Fig. 7 will show the shower scintillator's results with all boundaries taken together, and Fig. 8 will show the preshower's results, boundary-by-boundary. The histograms are organized thusly: the horizontal axis states the number of times a photon impacts the given boundary, and the vertical axis states the number of photon events which impacted the boundary said number of times (e.g., if the bar above the horizontal axis value of 4 reaches up to 20, it means that 20 photons struck that boundary 4 times in the simulation). The data in these graphs are for 1000-event simulations, where a photon's chance to escape the scintillator during a collision is defined to be 20% if the incident angle is less than the critical angle, and 1% if the incident angle is greater in the simulation. The preshower graphs omit the 0-bin (where the photon does not impact a given boundary for a given event), while the shower graphs do not.

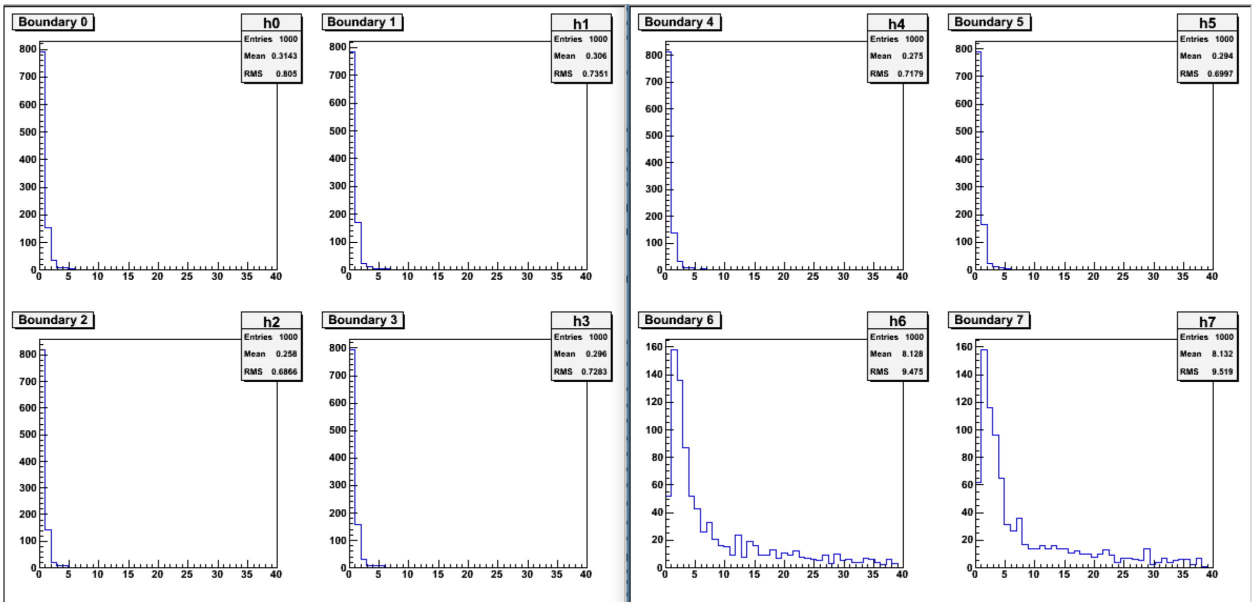

Fig. 6: Number of reflections on each of the 8 boundaries for the shower scintillator. These were obtained in a simulation with 1000 events.

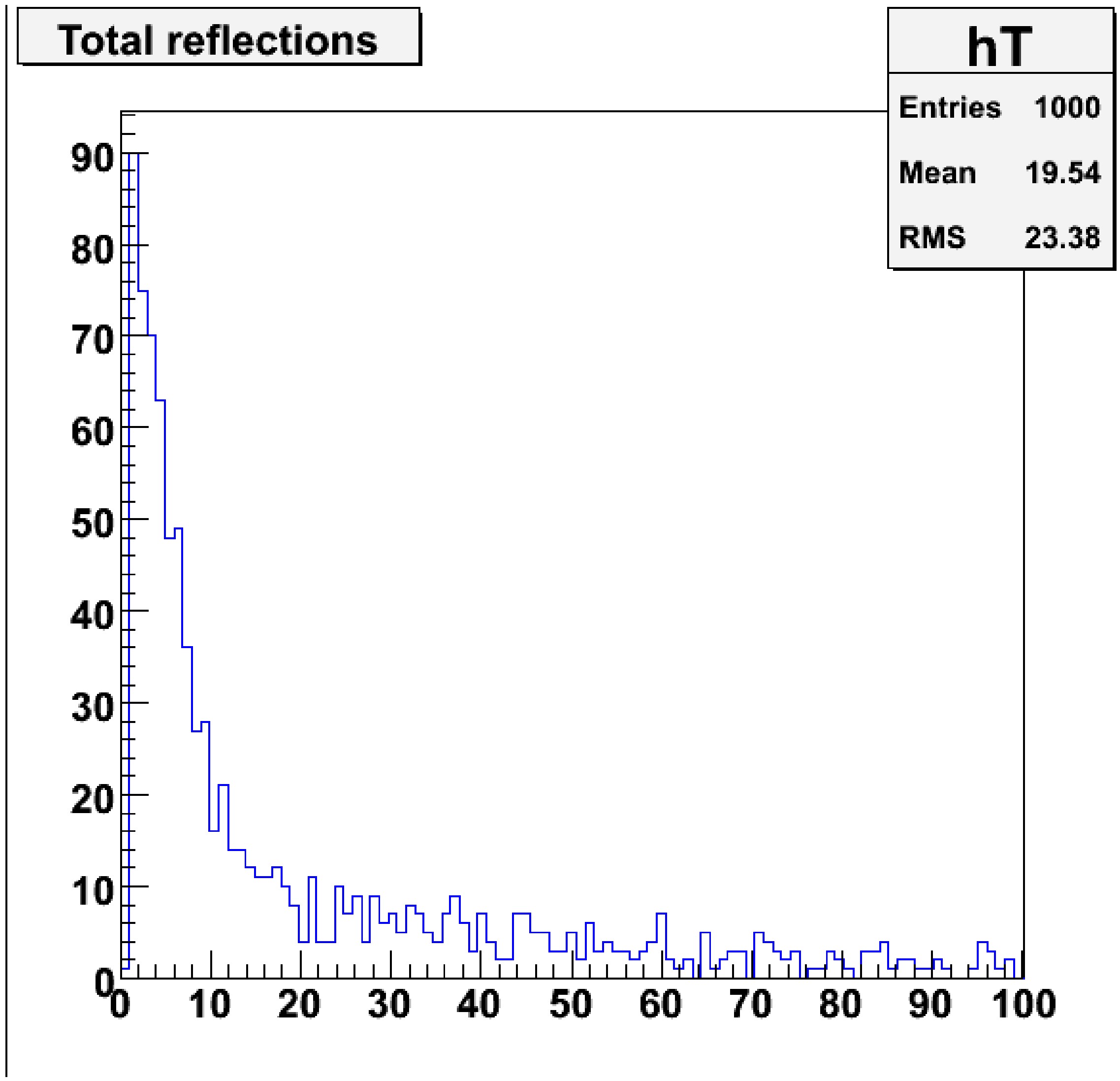

Fig. 7: Composite shower boundary reflections. This was obtained in a simulation with 1000 events.

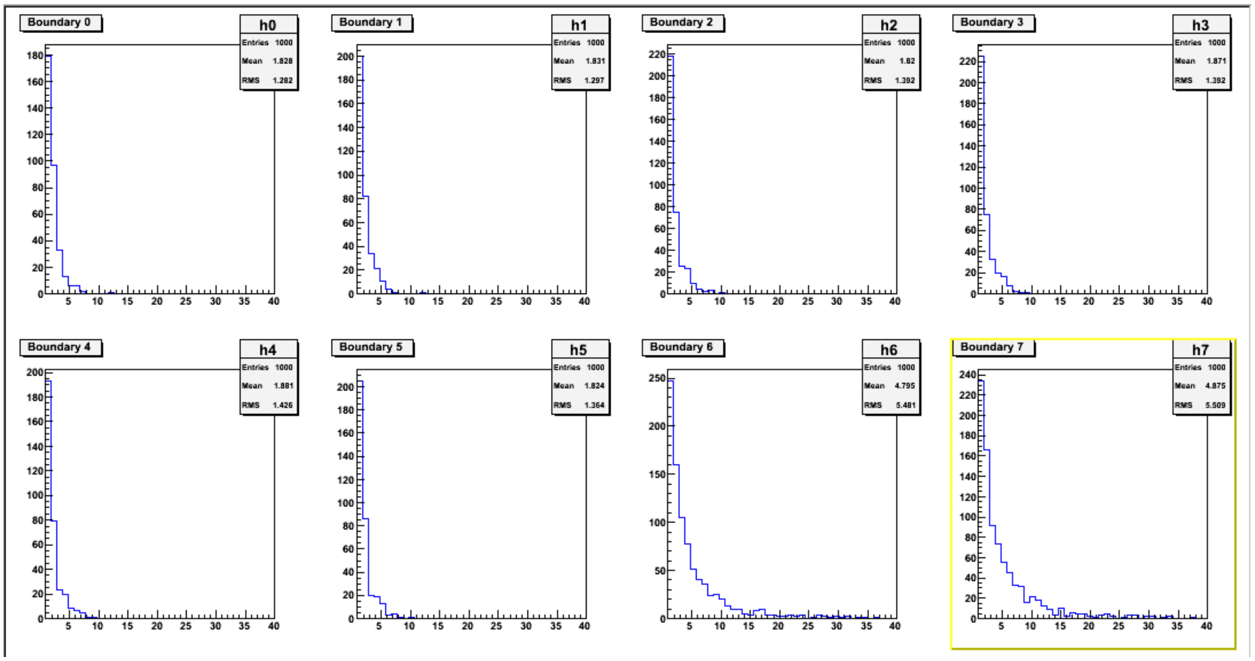

Fig. 8: Number of reflections on each of the 8 boundaries for the preshower scintillator. These were obtained in a simulation with 1000 events.

As expected, the six boundaries that represent the sides of the hexagon are impacted less often on average than the floor and the ceiling, and the disparity is more pronounced in the shower scintillator (whose height is far smaller than that of its preshower counterpart). Another expected value is 90 out of the 1000 photons escaping in the first collision in the shower scintillator: when the initial position and direction are random and unweighted, most of the photons will first collide with either the floor or ceiling. Since the critical angle for escaping the glass is just under 45°, just under 500 of the photons will impact at an angle less than the critical angle and thus have a 20% chance of escaping during the first boundary collision, while the remaining photons will impact at an angle greater than the critical angle and have a 1% chance of escaping; the value from the simulation is well within the expected ballpark.

Moving on, it's about time I explained how the wavelength-shifting fibers operate physically. The fibers themselves are developed by Kuraray Co., a Japanese company who's provided the following information regarding their absorption capabilities. The absorption chance of the fiber used in this project is determined by how far in 3-space the photon's path travels through the fiber before exiting:

$$Absorption\ chance = 1 - 10^{-1.276*d}$$

where d is the distance traveled through the fiber (in millimeters).

For the shower scintillator, the method used to calculate collisions and distance traveled through the fibers was to reduce the first step to a two-dimensional problem, and use the formula to calculate the intersection points between a line and a circle in 2-space. The formula employs the forms of a line and a circle, respectively:

$$y = mx + D,\ (x - a)^2 + (y - b)^2 = r^2$$

The nature of the simulation requires that no line be perfectly tangential to any circle, and using randomly- generated doubles in C++, the chance of generating a line perfectly tangential to a given circle is miniscule. The shower scintillator has 96 fibers, all symmetric about the vertical axis and with a circular cross-section and a diameter of 1mm (each of the 96 fibers will have a different a and b in the formula above, but all will have $r = 0.5mm$). The m from the equation of the line is equal to what I've already defined as $tan(\varphi 1)$ due to geometric convention ($slope\ = \frac{rise}{run}$), and the D is

$$D = y_0 - \tan(\varphi 1) * x_0$$

where y_0 and x_0 are taken from the initial point on the photon's path.

Another value of interest is Δ, defined as:

$$\Delta = r^2(1 + m)^2 - (b - m * a - D)^2$$

When $\Delta > 0$, a 2-dimensional line intersects the circle in 2-dimensional space at the points

$$x_{1,2} = \frac{\left(a + b * m - D * m \pm sqrt(\Delta)\right)}{1 + m^2}$$ [2]

with corresponding

$$y_{1,2} = m * x_{1,2} + D$$

[2] These formulas were found at http://www.ambrsoft.com/TrigoCalc/Circles2/circlrLine_.htm

At this point, it is necessary to again account for all three special dimensions. The z-coordinates corresponding to each collision point with the fiber's circular cross-section are:

$$z_{1,2} = \tan(\varphi 2) * sqrt((x_{1,2} - x_0)^2 + (y_{1,2} - y_0)^2)$$

Finally, we have the distance the photon travels through a given fiber:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

This will be used to calculate the absorption chance when a photon passes through a fiber. When I coded it into my C++ program, I would calculate absorption chance and then generate a random double between 0 and 1; if the random double was less than or equal to the absorption chance, the photon would be absorbed and the event would end. As there are 96 fibers, 96 of these calculations must be made for each photon, every time before it impacts a boundary; I assigned the center-point for each fiber to two arrays (one for the x-coordinates, another for the y) and then ran the previous calculations through a loop, checking each fiber, before each boundary reflection.

Again, there are a couple of constraints to consider. One is that the line may intersect the cross-section of one of the fibers in 3-space, but do so outside the scintillator (at a negative z or a z greater than the height of the scintillator). There are two conditions of this that I considered: when both z-coordinates are outside the scintillator, and when one is inside and the other outside. In the former condition, the photon simply doesn't collide with the fiber, striking a boundary first. In the latter, however, it is necessary to know that, when fully assembled into the experimental apparatus, the shower scintillator would in fact be many showers stacked one atop the other and separated by a reflective metal. With this in mind, I would exploit the symmetry of the apparatus and treat the photon's path through the fiber as a free reflection (chance of escape = 0) and still use the prior formulas to calculate its absorption chance. Fig. 8 demonstrates this condition:

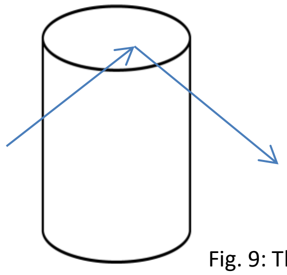
Fig. 9: The cylinder represents the fiber, rays represent a possible path of the photon.

When dealing with the preshower scintillator, the method employed is similar to that of the shower; however, this time when looking at the 2-dimensional cross-section, one sees two fiber rings, one of radius $r = 4.5cm$ and the other of radius $r = 4.6cm$. In the interest of time, I treated the cross-section of the fiber ring as rectangular rather than 4 fibers of 1mm diameter each stacked atop one another. This approximation will over-estimate the number of photons absorbed. The x- and y-intersections will have to be calculated for each ring separately. Fig. 10 is a display of the cross-section used in my approximation and the ring's actual cross-section, respectively:

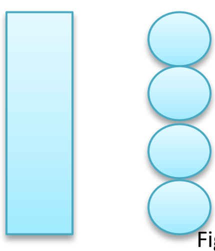
Fig. 10

This time, I used the equation for a line $ax + by = c$ while retaining the equation for a circle used before, yielding the x- and y-coordinates (assuming sum under the square-root is positive):

$$x_{1,2} = \frac{a*c \pm b*\sqrt{(r^2(a^2+b^2)-c^2)}}{a^2+b^2}, y_{1,2} = \frac{b*c \mp a*\sqrt{(r^2(a^2+b^2)-c^2)}}{a^2+b^2}$$ [3]

The z-coordinates are found, again, by

$$z_{1,2} = \tan(\varphi 2)\sqrt{\left(x_{1,2} - x_0\right)^2 + \left(y_{1,2} - y_0\right)^2}$$

And the distance is found by

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

[3] These formulas were found at https://en.wikipedia.org/wiki/Intersection_(Euclidean_geometry)#A_line_and_a_circle

These calculations must be made twice, once for $r = 4.5cm$, and once for $r = 4.6cm$. Like the shower-scintillator, there is more than one way the photon path can pass through the fibers; the simplest is passing through both rings without incident, another is similar to the dilemma explored with the shower scintillator (and has a similar solution) where the photon's path exits the scintillator before it exits the fiber, and yet another is the case where the photon's path is nearly tangential to the fiber ring such that it passes though the outer radius (4.6cm) but not the inner radius (4.5cm). In this case, there will be no real solution to the line-circle intersection formula for the smaller radius and only the entry and exit points along the outer radii are to be considered.

With that, the foundations of the simulation programs for both the preshower and shower scintillators are completed; they are capable of generating an arbitrarily high number of photon events and calculating the number absorbed. My final task this fall was to study how the reflectivity of the boundaries influences the absorption efficiencies of the scintillators. The default reflectivity for the boundaries for both scintillators up until now was 80% when the incident angle is less than the critical angle, and 99% if the incident angle (non-total internal reflection) is greater than or equal to the critical angle.

For the preshower scintillator, I then varied both of these values: the reflectivity for smaller incident angles (non-total internal reflection) beginning at 70%, increasing by increments of 5% (75%,…100%), and the reflectivity when the incident angle is greater than the critical angle (total internal reflection) beginning at 90% and increasing by increments of 2% up to 100%. For the shower I followed a similar process; however, I added another dimension of values as I changed the reflectivity of the floor and ceiling separately from that of the six hexagonal sides. When varying the values of one aspect of its reflectivity, I held the others constant at their defaults (99% for critical angle, 80% for smaller angles). The values presented were, unless stated otherwise, calculated by simulations of 10000 randomly-generated photon events. The value "Absorption efficiency" is defined as:

$$\frac{total\ number\ of\ photons\ absorbed\ by\ the\ fibers}{(total\ number\ of\ photons\ generated)}$$

Table 1 and Fig. 11 describe the preshower scintillator varying the boundaries' normal reflectivity while keeping the reflectivity of the total internal reflection constant at 99%:

Table 1: Absorption efficiency of preshower with variable reflectivity (non-total internal reflection)

Reflectivity	70%	75%	80%	85%	90%	95%	100%[4]
Absorption efficiency	68.13%	70.19%	73.16%	75.69%	79.69%	85.94%	97.25%

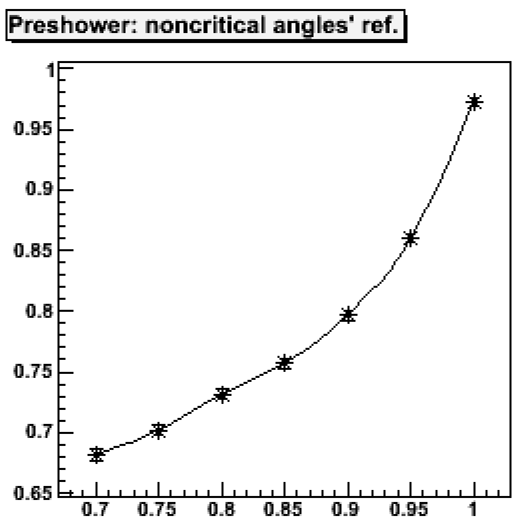

Fig. 11

Table 2 and Fig. 12 describe the preshower scintillator varying reflectivity when dealing with the critical angle, while keeping the reflectivity of the non-total internal reflection constant at 80%:

[4] The percentage value used 2000 photon events and reflectivity of 99.9999% due to the limitations of the .dat file size. However, the absorption efficiency is less than 100% primarily because the reflectivity of the total internal reflection is held constant at 99%.

Table 2: Absorption efficiency of preshower with variable reflectivity (total internal reflection)

Reflectivity	90%	92%	94%	96%	98%	100%[5]
Absorption efficiency	69.66%	70.20%	71.30%	71.81%	72.92%	73.20%

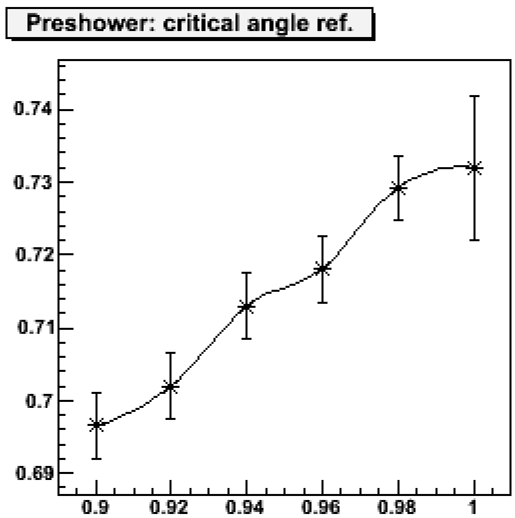

Fig. 12

Table 3 and Fig. 13 describe the shower scintillator varying the floor and ceiling's reflectivity, while keeping the 6 hexagonal sides' reflectivity constant at 80% and the reflectivity of total internal reflection constant at 99%:

Table 3: Absorption efficiency of shower with variable reflectivity (non-total internal reflection, floor and ceiling only)

Reflectivity	70%	75%	80%	85%	90%	95%	100%[6]
Absorption efficiency	70.51%	71.32%	72.19%	72.82%	74.25%	76.25%	88.00%

[5] For this value I used 2000 photon events rather than 10000, and the value 99.9999% rather than 100% for the same reason as above. However, the absorption efficiency is less than 100% primarily because the reflectivity of the non-total internal reflection is held constant at 80%.

[6] For this value I used 2000 photon events rather than 10000 for the same reason as the note just above. However, the absorption efficiency is less than 100% primarily because the 6 hexagonal sides' reflectivity is held constant at 80% and the reflectivity of the total internal reflection is held constant at 99%.

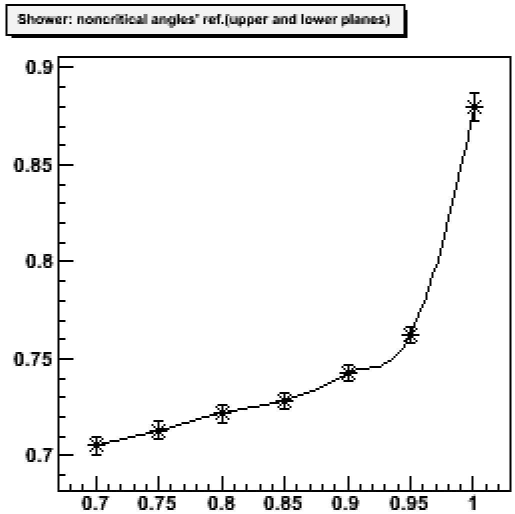

Fig. 13

Table 4 and Fig. 14 describe the shower scintillator varying the hexagonal sides' reflectivity, while keeping the floor and ceiling's reflectivity constant at 80%, and the reflectivity of the total internal reflection constant at 99%:

Table 4: Absorption efficiency of shower with variable reflectivity (non-total internal reflection, hexagonal sides only)

Reflectivity	70%	75%	80%	85%	90%	95%	100%
Absorption efficiency	70.87%	71.81%	72.62%	72.84%	73.56%	74.08%	74.55%

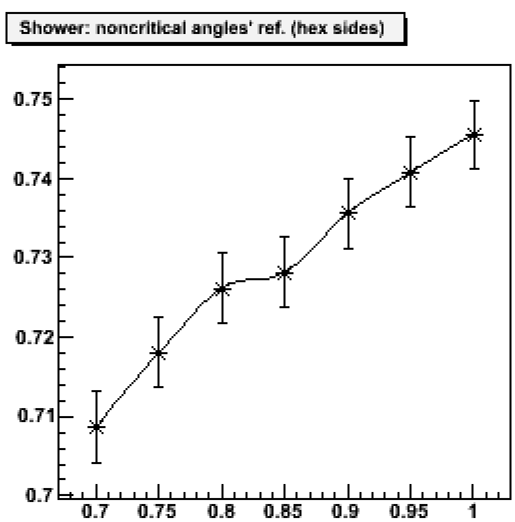

Fig. 14

Table 5 and Fig. 15 describe the shower scintillator varying the reflectivity when dealing with the critical angle, while keeping the reflectivity of the non-total internal reflection constant at 80%:

Table 5: Absorption efficiency of shower with variable reflectivity (total internal reflection)

Reflectivity	90%	92%	94%	96%	98%	100%[7]
Absorption efficiency	63.23%	64.50%	65.86%	67.51%	70.09%	75.85%

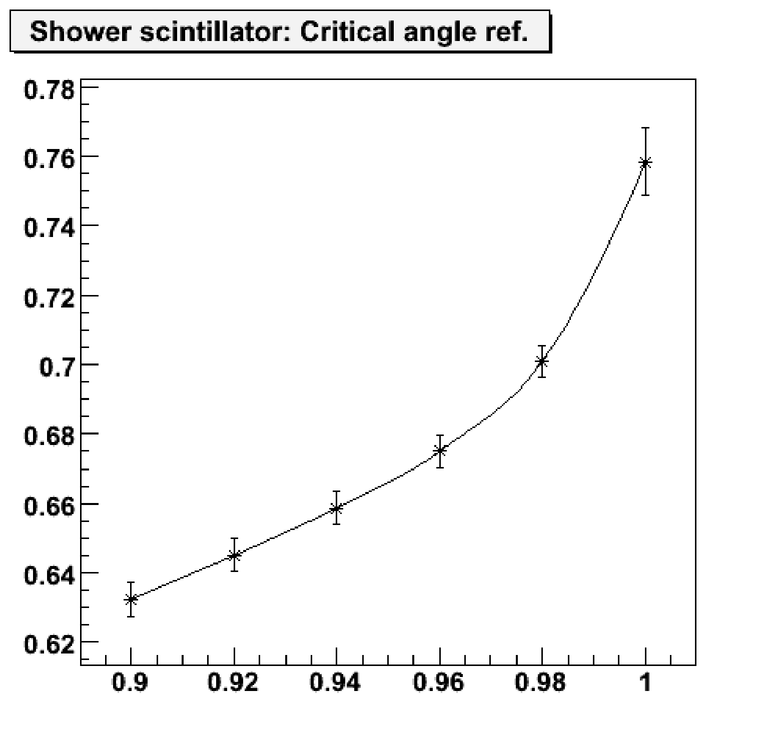

Fig. 15

When viewing the graphs, one aspect to consider is the apparent visual differences in the widths of the error-bars. These differences can be attributed to the vertical axes covering differing ranges, thus the scale is different in each. Error-bar widths were calculated from efficiency (e) and the number of events for a given efficiency calculation (N): $Standard\ deviation = \frac{\sqrt{N*e*(1-e)}}{N}$

[7] For this value I used the value 99.999% rather than 100% and 2000 photons instead of 10000 for the same reason as the notes above. However, the absorption efficiency is less than 100% primarily because the reflectivity of the non-total internal reflection is held constant at 80%.