Progress Report #2

Daniel Gawerc
Mentor: Maria Spiropulu
Co-Mentor: Cristian Peña

California Institute of Technology

29 July 2016

1   Background

At the Large Hadron Collider (LHC), the High Luminosity upgrades plan to achieve luminosities of greater than 10^{34} cm^{-2}s^{-1}, with $140-200$ pileup events per bunch crossing every 25 ns [1]. Because of the physical spread of each bunch, the collisions occur over a large region rather than at an isolated point, which leads to a spread in time (~ 200 ps) [1]. Thus, detectors in the calorimeter must have a good (low) time resolution in order to resolve individual collisions. In order to reconstruct the vertex of a collision with an increased pileup, the electromagnetic calorimeter (ECAL) detectors should have a time resolution of about 10 ps [1].

2   Experimental Setup

This summer, the Caltech Precision Timing group took data at the Fermilab Test Beam Facility (FBTF). The experiment involves placing different calorimeter detectors in the beam line, in order to observe how the addition of detectors in the perpendicular transverse plane and/or parallel longitudinal direction along the beam line will improve the overall detector time resolution. The layers of detectors are meant to mimic the proposed High Granularity Calorimeter (HGC) detector, which shall replace the CMS ECAL endcap, and is described in detail below [2]. The results of this research may be useful in preparing the LHC High Luminosity upgrades.

In the FTBF experiment, different types of detectors were utilized. In the front of the setup, a scintillator crystal and PMT are used as a trigger. Depending on the configuration, there are a few radiation lengths (X_0) of either lead or tungsten absorbers behind the PMT in order to spread out and propagate the particle shower and produce more secondary electrons and photons. A depiction of the setup is given in Figure 1.

The High Granularity Calorimeter (or PicoSil when referring to a single pixel's readout that has been optimized for timing) layer has an array of separated silicon pixels, in a
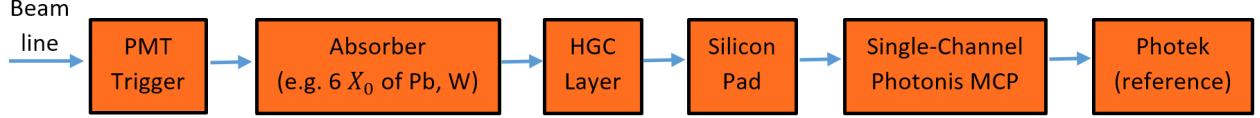
1

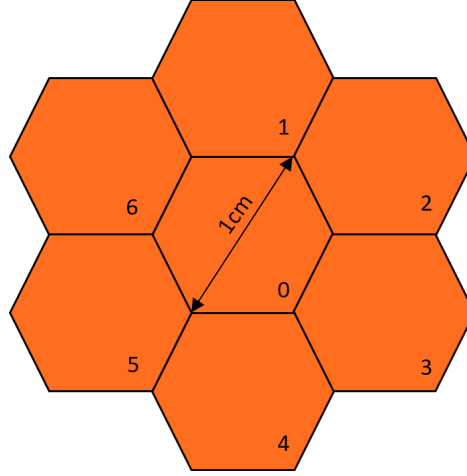Figure 1: A typical experiment configuration.

Figure 2: A depiction of the innermost 7 pixels of the HGC layer. The actual board has more than 7 pixels in a single layer.

hexagonal honeycomb geometry [2]. A depiction of the HGC layer is given in Figure 2. Because the beam is focused at the HGC center pixel (0 in Figure 2), its signal is reduced using a 6 dB attenuator due to the high charge detected relative to other pixels.

The Photek 240 MCP is used as a reference detector because its time resolution is good (~ 10 ps). Because the readout board for all the detectors is divided into 4 groups, which have some time delay between them, the Photek signal is split into 4 cables and plugged into each group as the reference signal from which the TOFs (time of flight, or Δt) are calculated. Specifically, the time taken for the particle to travel from a device to the reference Photek - the time of flight – is represented by Δt. In the data analysis, the Δt values over all events in the run (with certain *bad* events excluded) are combined in a histogram, forming a Gaussian peak. The peak σ parameter is denoted the time resolution.

There is a 64-channel and a single-channel Photonis MCP. The 64-channel Photonis was not present for most runs, and the single-channel Photonis was instead used. Lastly, a silicon pad (SiPad) detector was present for most runs, however it only detected pulses well in the very early runs, and does not seem to be useful in most of the analysis.

3   Data Analysis

This section aims to explain the general data analysis methods discussed later in this report.

3.1 Data Analysis: Overview

The Caltech Precision Timing group already had developed some analysis code for previous test beam experiments. The program *dat2rootCP* analyzes the .dat files (with low-level information) from the runs and turn them into .root files containing a tree with plots and fit results (higher-level information). Since some of the data contained ringing noise, which often resulted in the wrong peak being identified as the pulse, *dat2rootCP* was updated to correct for these events. Additionally, new macros were written in order to automate batch file analyses and merging the results into a single .root file (for when the setup configuration and beam type were the same over many runs). These programs utilize the ROOT programming language, which is based largely on C++. It is the main programming language used by scientists at CERN and in the field of HEP (see: root.cern.ch). The .root file extensions are associated with the ROOT language.

Because the HGC layer has multiple pixels, each pixel detects different signals and has a different time resolution. The *MultiChannelStudy* code was created (here channel and pixel are used interchangeably), which calculates and combines the TOF (Δt) histograms for different pixels in order to return a better overall time resolution. The Δt values are calculated for every event in a run by subtracting the pulse time in one device from the pulse time in the Photek (making sure that it is from the same cabling group 1-4 as the device). A 1-D histogram is then filled event-by-event with the Δt values. In an ideal world with perfect detectors and no variation in each event, the histogram would be a Dirac delta function with a time resolution of zero. However, due to detector jitter, the signal-to-noise ratio and the particle shower development in the material will vary event-by-event, resulting in Gaussian TOF histograms [2]. After fitting the histogram to a Gaussian, the σ parameter gives the time resolution of the detector/pixel. If weighted correctly, combining the HGC pixels' Δt histograms will result in a combined histogram with better time resolution (smaller σ). This improvement is due to the combination of non-overlapping information about the shower size and development in each of the detectors.

Because there are also multiple devices along the beam line as well as multiple HGC pixels, the analysis of the coplanar HGC pixels has been called *transverse analysis*, and the analysis of multiple devices *longitudinal analysis*. This report shall deal mainly with the longitudinal analysis: analyzing the improvement in the time resolution when taking the HGC layer, single-channel Photonis MCP, and SiPad into account. However since it is best to use the time resolution from the entire HGC layer, much of the transverse analysis will be included as well. Due to re-cabling the wires of the detectors into different groups between runs 52 and 53, multiple longitudinal analysis codes were written for each setup. Additionally, the silicon pad has noise dominating for most of the runs, and substantially deteriorates the combined detector time resolution. For example, Figures 3 and 4 give the time of flight (Δt) histograms for runs 65-83 (same configuration) for the HGC center pixel and the Photonis MCP, along with their Gaussian fit σ parameter. The Gaussian fit used the log-likelihood maximization method, as the χ^2 minimization method is incorrectly biased by histogram bins with few/empty events. Figure 5 gives the TOF histogram for the silicon pad. Figure 6 is one possible combination histogram (more details below on how the individual device Δt's can be combined). Clearly the combined histogram improves the silicon pad resolution by about 4 times, but it is 4-5 times *worse* than the individual resolutions of

3

the HGC and MCP. Additionally, the silicon pad histogram is peaked, but certainly not Gaussian. Thus, because it is usually more favorable (in most - but not all - runs) to simply exclude the silicon pad, code that does not account for the silicon pad was also created.

Therefore, because of the change in cabling and the noise in the silicon pad, the following longitudinal analysis codes were used: *MultiDeviceStudy*, *MultiDeviceStudy_InitialWiring*, *MultiDeviceStudy_PicosilMCP*, and *MultiDeviceStudy_InitialWiring_PicosilMCP*. Because the most applicable file is *MultiDeviceStudy_PicosilMCP*, it is more extensive than the others, incorporating all of the ring 1 pixels (the 6 pixels surrounding the center pixel; refer to Figure 2) in the analysis. Thus, the code implements transverse analysis of the HGC as well as longitudinal analysis by adding the Photonis MCP.
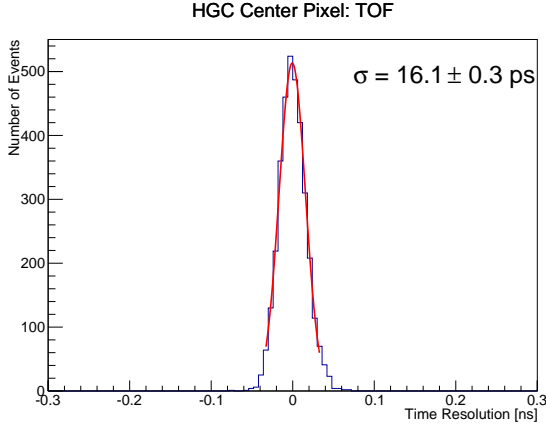

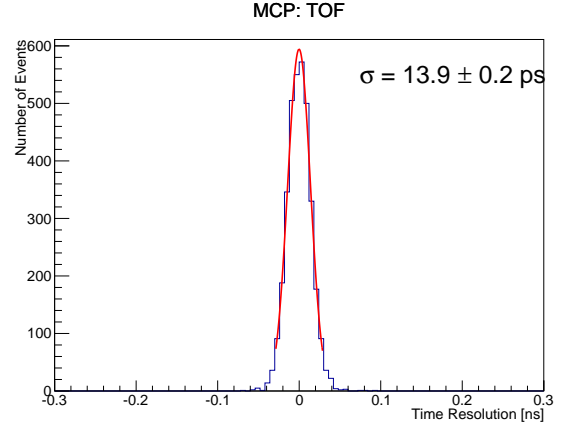Figure 3: TOF of the HGC Center Pixel

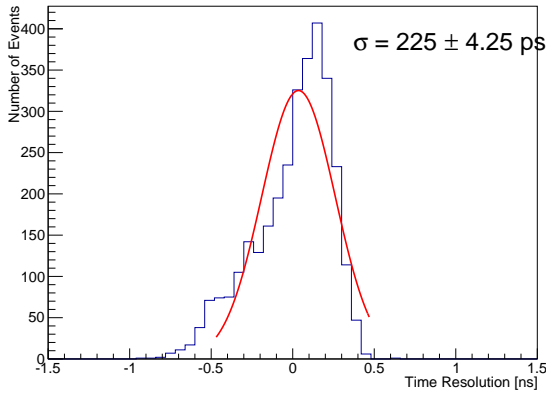
Figure 4: TOF of the Single-Channel Photonis MCP

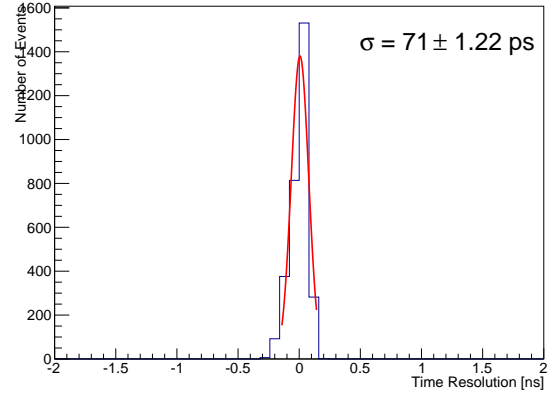Figure 5: TOF of the Silicon Pad

Figure 6: TOF of the HGC Center Pixel, Photonis, and Silicon Pad Combined with Total Device Charge Weighting

4

3.2 Data Analysis: Event Selection

Before discussing the various ways to combine the Δt values from different detectors, the methodology for selecting events that contribute to their individual TOF histograms should be reviewed. Because not every event contains a detected pulse, and some have a lot of ringing noise and a small signal-to-noise ratio, some events need to be cut out of the Δt calculations. In order to ensure that most of the *"bad"* events are removed from the histograms, a cut is applied to the calculated peak amplitude (in mA) and to the overall charge (in pC) of the event in that detector, forcing them both to be higher than some threshold. This requires that the peak found is not just noise and that the charge (found by integrating the plot) is significant enough to represent a pulse.

For example, analyzing the data (with *dat2rootCP*) from runs 65-83 returns a maximum amplitude value in the Photek for every event. Filling a histogram with all of these values in the command-line in ROOT gives the left plot in Figure 7. Clearly the amplitudes that are near zero or negative are events without pulses or very noisy events. These events should be ignored, so a cut is implemented to select for amplitudes above a certain value. In this case the value appears to be $0.1 \times \sqrt{10}$ mA. The right plot in Figure 7 shows the histogram of the amplitudes with only the events that passed the cut. Figure 8 shows the same methodology being applied to the charge distribution, with a cut at $2 \times \sqrt{10}$ pC. Similar cuts were made in the HGC pixels and Photonis MCP.
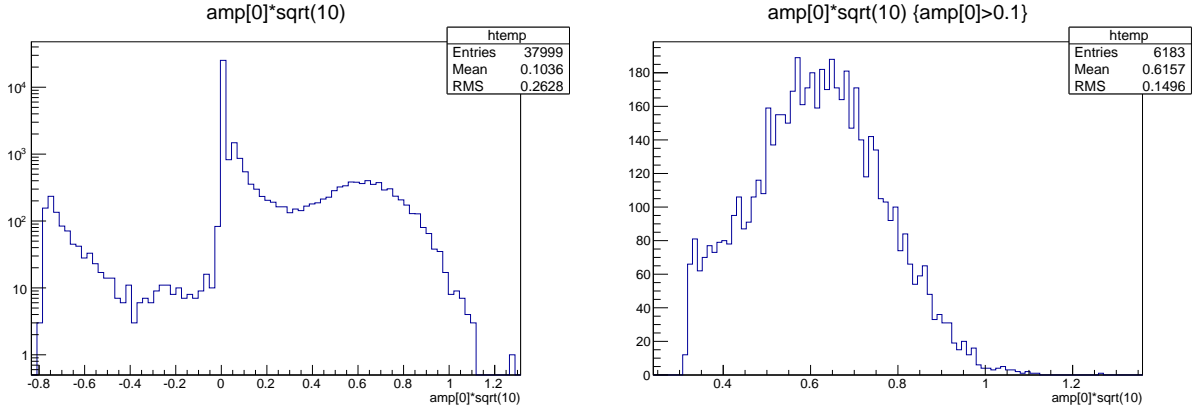

Figure 7: Pulse peak amplitude distribution for the Photek 240 MCP before and after the cut. Non-positive amplitudes represent *"bad"* events. The $\sqrt{10}$ factor accounts for a 6 dB attenuator.

3.3 Data Analysis: Device Combination

The *MultiDeviceStudy_PicosilMCP* code generates many variations of TOF combination histograms. The simplest are the HGC center pixel and the single-channel Photonis MCP histograms (Figures 3 and 4). These TOFs are generated without combining the Δt values from multiple detectors. There are 4 main ways that have been utilized to combine different detectors.
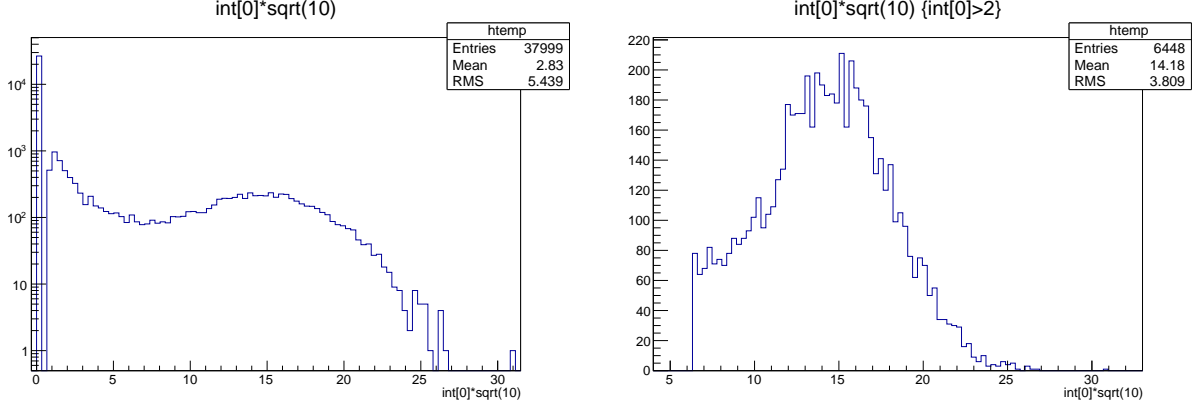
Figure 8: Pulse charge distribution for the Photek 240 MCP before and after the cut. Zeroed charges represent events with only noise. The $\sqrt{10}$ factor accounts for a 6 dB attenuator.

3.3.1 Combination Method: Equal Weighting

This weighting method is the most straightforward. If there are 2 devices which all pass the cuts, then their Δt values are computed. These 2 Δt values are then combined into a single Δt value by computing the arithmetic average. This final Δt is then used to populate the histogram. Mathematically, this is given by,

$$\Delta t_{final} = \frac{\Delta t_1 + \Delta t_2}{2}.$$

It gets slightly harder to compute this Δt_{final} when some combinations do not require that *all* detectors pass the cuts, such as when trying to combine the center HGC layer pixel with the 6 inner ring pixels. Because many events have 3 or 4 pixels passing cuts, but only a few events have all 7 passing cuts, the Δt_{final} is calculated at every event using whichever pixels were able to pass the cuts. This means the number of $\Delta t_{initial}$'s will vary with event. Mathematically, it can more generally be written:

$$\Delta t_{final_i} = \frac{\sum_{k=1}^{N} \Delta t_{k_i}}{N}$$

for N detectors passing cuts, where Δt_{k_i} represents the kth detector's TOF at event i.

3.3.2 Combination Method: Event Charge Weighting

This weighting method uses the detected event pulse charge in each device to weight each $\Delta t_{initial}$ value. As illustrated in Figure 8, the pulse charge varies with event, so the relative weightings between detectors will change on an event-by-event basis. Mathematically, an event's charge weighting is given by,

6

$$\Delta t_{final_i} = \frac{\sum\limits_{k=1}^{N} \Delta t_{k_i} q_{k_i}}{\sum\limits_{k=1}^{N} q_{k_i}}$$

for N detectors passing cuts, where q_{k_i} represents the kth detector's charge at event i.

3.3.3 Combination Method: Total Charge Weighting

Whereas the event charge method weights each detector differently for different events, the total charge method weights each detector the same way over the entire run. Rather than use the event charge, this method sums a detector's event charge for every event in the run, and uses that value as the weighting factor. Mathematically,

$$\Delta t_{final_i} = \frac{\sum\limits_{k=1}^{N} \Delta t_{k_i} Q_k}{\sum\limits_{k=1}^{N} Q_k}, \qquad Q_k = \sum\limits_{all\ events\ i} q_{k_i}$$

for N detectors passing cuts, where q_{k_i} represents the kth detector's charge at event i.

3.3.4 Combination Method: Charge MPV Weighting

Like the total charge method, this method also assigns weighting factors to each detector that do not change with the event. This method is additionally only used within the HGC layer, for combining the TOFs of different pixels. Rather than being determined by the total charge detected, this method fits a Landau distribution to the charge distribution histogram of each of the inner ring pixels. The peak – most probable value (MPV) – of the fit is then used as the weighting factor for the inner ring pixels. Because there are far more events passing cuts – and with a higher charge – in the center pixel, the distribution is Gaussian, and its weighting factor is the mean of a Gaussian fit. Both types of fits are accomplished by maximizing the log-likelihood in order to avoid the biasing of the χ^2 minimization fit for low-event bins. Mathematically, this method is represented by the same formula as the total charge method, letting Q_k represent the detector's MPV, instead of total charge.

4 Time of Flight Plots: Simple Combinations

Now that all the different combination methods have been described, understanding the resultant plots is now possible. In order to show results in the simplest manner, this section will only use plots coming from 1 data set with the same configuration. Specifically, the results in this section will all be for runs 104-116 excluding runs 111-114, which have a test beam of 32 GeV electrons, and a tungsten (W) absorber 1mm in front of the HGC layer (it may help to recall the setup from Figure 1).

Starting from the lower-level plots, Figure 9 contains the TOF histograms for the HGC center pixel and Photonis MCP.
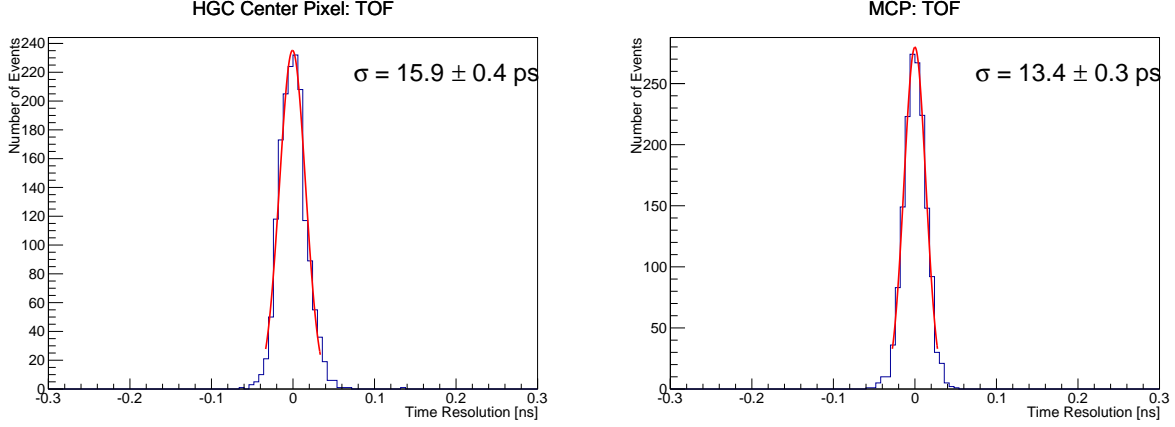
Figure 9: TOF histograms of HGC center pixel and of Photonis MCP.

In order to see how the time resolution improves when adding in the inner ring pixels, Figure 10 contains different weighting methods based on the charge. Equal weighting is not used here because the center pixel has more events and a better resolution than the other pixels, so it should be weighted more heavily. In Figure 10, there is a decrease in σ (from 15.9 ps in Figure 9) and thus an improvement in the time resolution for all combinations. Although the uncertainties in σ are too large to be conclusive, Figure 10 suggests that the event charge weighting is the worst of the 3 methods.
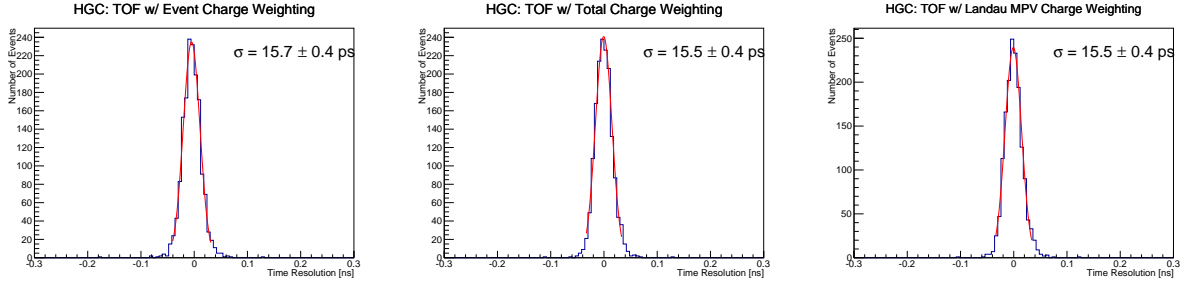

Figure 10: TOF histograms of HGC pixels combined by event charge, total charge, and charge MPV.

The above illustrates the transverse portion of this analysis. For the longitudinal aspect, the Δt values that are used to populate the HGC layer histograms in Figure 10 will be combined with the Photonis MCP Δt values used to populate the right histogram in Figure 9. There are some basic and naive ways to combine the HGC layer with the Photonis MCP. The first method includes assigning an equal weighting to every pixel/detector that passes the cuts. The second method includes assigning an equal weight to all HGC pixels that pass the cuts and calculating the HGC Δt, which is then weighted equally with the Photonis Δt. This second method can be written mathematically,

$$\Delta t_i = \frac{\Delta t_{MCP_i}}{2} + \frac{\sum\limits_{j=1}^{N} \Delta t_{HGC_{ij}}}{2 \times N}, \quad event\ i, \quad N\ HGC\ pixels\ passing\ cuts.$$

The results form these two methods can be seen in Figure 11. Note that these two methods may seem too elementary – especially the first method, which weights the inner ring pixels too much, and the center pixel and Photonis too little – but they are brought up now because one of them will be used later in a smarter way. Here however, both methods (utilizing 8 detectors) gave a worse time resolution than just the central HGC pixel.
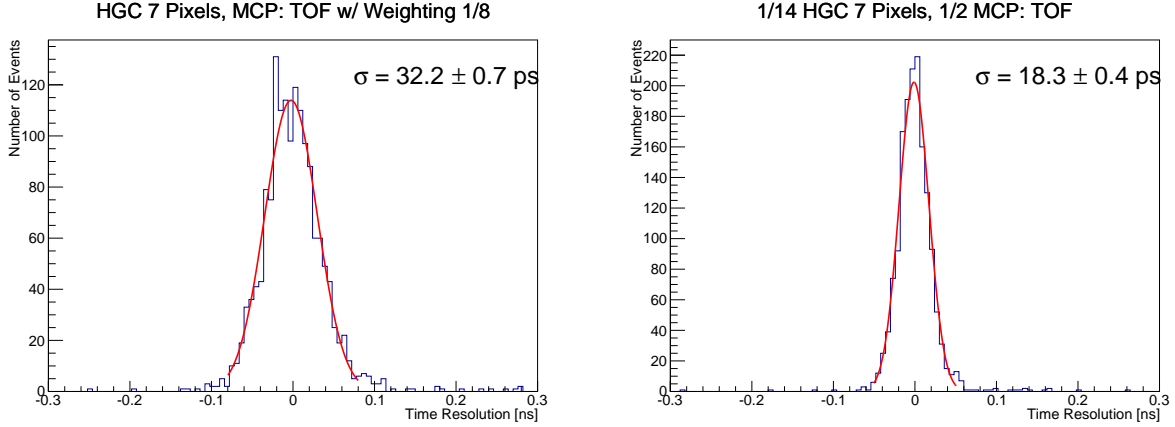

Figure 11: Implementation of basic methods 1 (left) and 2 (right) as described in the text.

Smarter ways to combine the HGC layer with the Photonis MCP includes weighting each pixel and the Photonis by their event or total charge. Mathematically, these methods would be given by,

$$\Delta t_i = \frac{\Delta t_{MCP_i} q_{MCP_i} + \sum\limits_{all\ pixels} \Delta t_{HGC_i} q_{HGC_i}}{q_{MCP_i} + \sum\limits_{all\ pixels} q_{HGC_i}}$$

$$\Delta t_i = \frac{\Delta t_{MCP_i} Q_{MCP} + \sum\limits_{all\ pixels} \Delta t_{HGC_i} Q_{HGC}}{Q_{MCP} + \sum\limits_{all\ pixels} Q_{HGC}}$$

Figure 12 gives the TOF histograms using these combination methods. While the total charge method time resolution shows an undeniable improvement, the event charge method time resolution of 14.2 ps is better than any HGC layer combination (Figure 10) but worse than the resolution of just the Photonis (Figure 9).

It turns out, however, that an HGC/Photonis combination using just charges is not the most accurate way to weight each detector. The first indication that there may be a smarter method comes from the equal weighting combination of the HGC center pixel (left Figure 9; 15.9 ps) and the Photonis. Figure 13 shows this combination. While the total charge-combined HGC layer has a better time resolution than the center pixel, the latter's
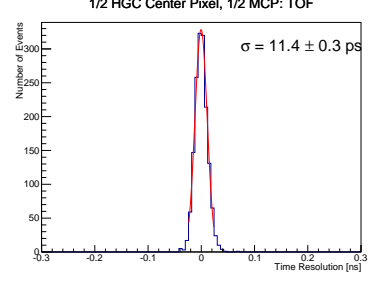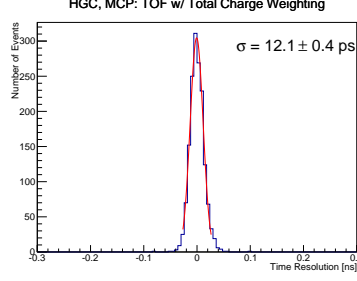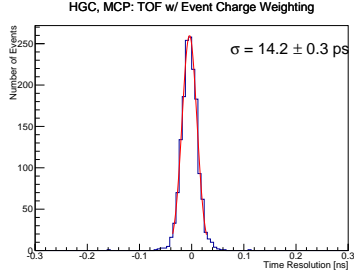
Figure 12: TOF histograms of the HGC pixels and Photonis MCP, using event (left) and total (right) charge weighting.

Figure 13: TOF histogram weighting Δt_{Center} and $\Delta t_{Photonis}$ by $\frac{1}{2}$.

equal combination with the Photonis is significantly better (11.4 ps compared to 12.1 ps). The equal weighting between devices is better than one of the charge weightings because it approximates that the same amount of charge flows into both detectors. Even though the charge is not actually equal, this approximation is better than using the charge weighting *because the relative gain of the HGC and Photonis is not known.* Put another way, a particle of a specific energy detected in the HGC would not return the same charge as a detection in the Photonis. Therefore, a better way to weight the Δt values would be to do either an event charge, total charge, or charge MPV weighting for the HGC pixel Δt's, then weight that value and the Photonis Δt equally (Figure 14).
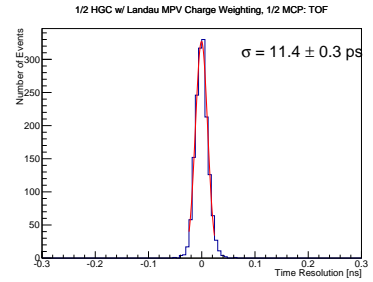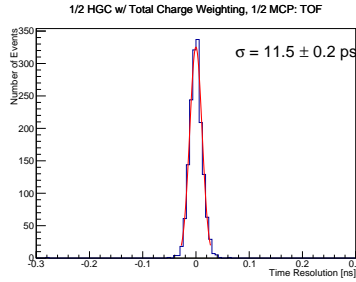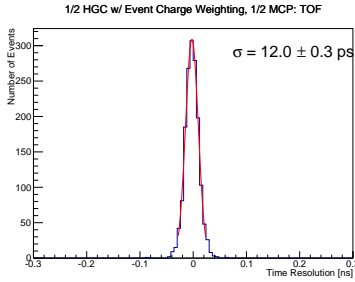

Figure 14: TOF histogram weighting Δt_{HGC} and $\Delta t_{Photonis}$ by $\frac{1}{2}$, with Δt_{HGC} computed via event (left), total (middle), and MPV (right) charge weightings.

5   Time of Flight Between HGC and Photonis

6   SKIROC2 Emulation

6.1   Background

6.2   500 ps Smear

6.2.1   Add Pixels Individually: Use All Pixels

6.2.2   Add Pixels Individually: Exclude Center Pixel

6.3   50 ps Smear

6.3.1   Add Pixels Individually: Use All Pixels

6.3.2   Add Pixels Individually: Exclude Center Pixel

6.4   Minimize χ^2 or Maximize Log-Likelihood

——————Move this subsection somewhere more appropriate—————-

---------------------- Stopped my revisions here ----------------------

----

7   Preliminary Results

I am currently using my analysis code for the HGC and MCP to look at how these time resolutions change for different configurations (i.e. absorber type, spacing, and thickness; beam energy; etc). So far, I have only looked at the following configurations:

The last run is a reminder that the cuts should be edited for every run in order to correctly select the proper data. There may also be a beam energy dependence for the value of the cuts. So far, I have determined that the following cuts are the best for the above runs:

As far as the rest of my research goes: I am going back to Caltech at the end of next week to continue my data analysis. I cannot really plan out how my analysis will evolve until I finish looking at the different runs, but then I will most likely take my results to someone in the Caltech Precision Timing group and figure out the next steps of my research.

References

[1] Adolf Bornheim et al. "Calorimeters for Precision Timing Measurements in High Energy Physics". In: *Journal of Physics: Conference Series* 587.1 (2015), p. 012057. URL: http://stacks.iop.org/1742-6596/587/i=1/a=012057.

[2] Cristian Pena. For revisions.