Improved Heavy Gas Cherenkov Detector Calibration Code

Vijay Kumar
July 31, 2020
University of Regina, Regina, SK, Canada
vkb135@uregina.ca

1 Heavy Gas Cherenkov (HGC) Detector

The HGC detector of the Super High Momentum Spectrometer (SHMS) is used for the separation of charged π/K. The HGC detector utilises four photomultiplier tubes (PMTs) to collect Cherenkov light. When charged particles pass through the detector with velocity greater than the velocity of light in the gas medium then they produce Cherenkov light. This Cherenkov light reflects from the four mirror quadrants inside the HGC tank and is then collected by the PMTs [1]. The purpose of the calibration is to produce a reliable translation from the raw FADC channel (charge in pC) to the physically accurate number of photo-electrons (NPE). This can be achieved by isolating the single photo-electron (SPE) peak to generate a calibration.

2 Calibration Code

The calibration code is available to download from Github [2]. The calibration scripts makes use of the TProof class in root to process the data [3]. The preliminary testing of the calibration code has been done with the Kaon LT experimental data and is showing promising results. Typically, a single run from the Kaon LT experiment does not have sufficient statistics to generate a reliable calibration, thus tests have generally been conducted by chaining together sets of 2-3 runs. The construction of the HGC is similar to the Noble Gas Cherenkov (NGC) detector. As such, with some adjustments to the HGC calibration code, it should be relatively straightforward to convert it to a code capable of calibrating the NGC.

2.1 Running the code

This package of the calibration code consists of three files, calibration.C, calibration.h and run_cal.C. These three files would be in the /hallc_replay_lt/CALIBRATION/shms_hgcer_calib directory. To run the code, users should -

2.1.1 Running steps

- Clone the calibration code from Github [2]
- Go to the code directory
- Create a new directory with the name of Calibration_plots for output files
- To run the calibration, use the run_cal.C macro with root

The pathing is set up for our environment on the farm but people should modify/add their own to the block around Line60 of the run_cal.C macro in this folder.

- It is highly recommend you execute the script without displaying any graphics (run in batch mode)
- This script requires between 4 and 6 arguments
- root -l -b -q run_cal.C(RunPrefix, NumEvents, nRuns, RunNumber1, RunNumber2, RunNumber3)
- RunPrefix The prefix to the replayed rootfile you wish to run, e.g if you files are Coin_Replay_#RUNNUMBER_#NUMEVENTS.root, the prefix would be Coin_Replay, omit the trailing _ before the runnumber
- NumEvents The number of events per run (must all be equal), e.g. if you ran all events, enter -1 here
- nRuns Number of runs you want to chain together and attempt to calibrate over, the script can handle 1-3 runs - ONLY enter 1, 2 or 3 here
- RunNumber1 The run number for the first run you want to add to the chain

If you entered 2 or 3 for nRuns, you will also need to specify RunNumber2 and RunNumber3.

- RunNumber2 The run number for the second run you want to add to the chain
- RunNumber3 The run number for the third run you want to add to the chain

As an example of running the script, suppose you want to analyse Coin_Replay_5555_-1.root, Coin_Replay_5556_-1.root and Coin_Replay_5557_-1.root. Execute in the calibration code directory,

- root -l -b -q run_cal.C(Coin_Replay, -1, 3, 5555, 5556, 5557)

2.1.2 Output files

The code will produce the calibration plots on a single pdf file with the file name is based on whether users are running the calibration code for a single root file or chaining of 2-3 root files. If calibration is performed for a single root file then the output pdf file name will be Calibration_plots_run1.pdf, where run1 is the run number. If calibration is carry out for more than one run number then the output pdf file name will be Calibration_plots_run1-run2.pdf, where run1 and run2 are the first and last run numbers. After running the calibration code for the first time, users should check the timing cuts in the produced pdf file. If the timing cuts are not correct then users should open the calibration.C script and correct the timing cuts in the block starting at line number 138. Without good timing cuts users will get nonsense calibration parameters.

The code generates two types of output files of the calibration parameters and both are available in the Calibration_plots directory. By default, their names will be phgcer_calib_run1.param

and phgcer_calib_err_run1.param or phgcer_calib_run1-run2.param and phgcer_calib_err_run1-run2.param depending on whether calibration did for one run number or more than one run number. The code will generate two guesses of the calibration parameters (more detail will be in the section 3). Both of the param files will have these two guesses of the calibration parameters. The first param file is of the same format that hcana takes in the replay of data therefore this can directly feed to the hcana after commenting the one guess of calibration parameters. The second param file is with calculated errors in the parameters and this is useful for studying the trend of the calibration over a period of run numbers.

3 Interpreting the calibration

In this report, the output details of the calibration plots and parameters are given from a calibrated set of run numbers 6918, 6919 and 6920. This set is taken from the 4.9 GeV beam energy run plan of the Kaon LT experiment. In the HGC detector, four PMTs receive Cherenkov light from the four mirror quadrants, therefore this process produces 16 histograms. Since only the SPE is desired, the histogram corresponding to a PMT looking at its own mirror quadrant is ignored, as it receives too much light which is difficult to isolate the SPE peak. Therefore, for each PMT three histograms are then fit with the sum of two Gaussian functions.

The code is setup to print out a text box of the Good fit or Bad fit on each quadrant. This is based on the number of events at the peak of the distribution and χ^2 per degree of freedom. By default the χ^2 condition for a good fit is between of $0.5-10$ and the max events at the peak is at least 40. Users can adjust these two statements in the calibration.C macro in the block starting at line number 546. The SPE mean (and error on the mean) of any "Good" quadrants are used to calculate an error weighted average (first guess calibration parameter) for each PMT. The fitted plot of the separated photo-electron peaks is shown in Figure 1. The red function is the sum of two Gaussian functions, while the black and green functions are the individual Guassian functions that are summed to form the red function.

The next study is for the checking of the quality of the calibration parameters. A new histogram is created to store the full calibrated spectrum for each individual PMT. This is done by filling the new histograms bin-by-bin with the previous histogram data scaled by the previously obtained calibration parameters. This is depicted by the equation

$$b_i' = \frac{b_i}{\overline{\mu}},$$

where b_i' and b_i are the i^{th} bins of the calibrated and uncalibrated spectra respectively and $\overline{\mu}$ is the error weighted average calculated previously for the PMT. This new histogram of the full distribution of Cherenkov light is then fit with a new function which is the sum of four Gaussian and two Poisson functions. The fitted new histogram is depicted in Figure 2, where the yellow function is the new function and others are the individual function drawn from the summed function. The range of 0-5 NPE is the most interesting region therefore Figure 2 is zoomed in the 0-5 NPE range. The zoomed version of Figure 2 is shown in Figure 3.

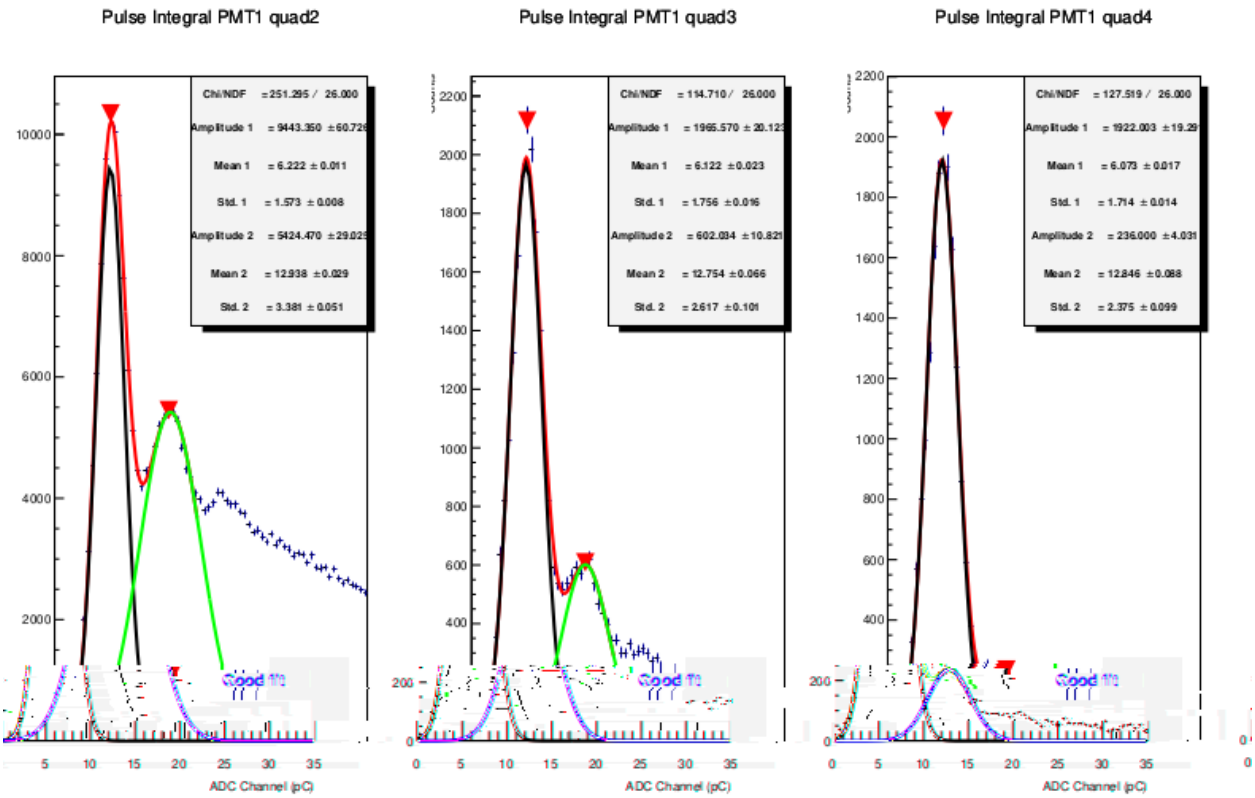

Figure 1: The separated photo-electron peaks fitted with the sum of two Gaussian.

The last plot that is produced is the linear spacing of the number of photo-electrons. This plot is fit with two linear functions, one function is fitted with a fixed intercept (intercept = 0) while the second is fitted with the intercept as a free parameter in the fit (intercept $\neq$ 0). This is shown in Figure 4 where two linear functions (black and red) are overlapping each other completely. This shows a high quality calibration.

The strategy of extracting second guess calibration parameters is just to multiply the previous calibration parameter with the mean of the first photoelectron peak in the full distribution plot (see line967 in calibration.C file) for each PMT. Finally, in Table 1 there is a comparison of new calibration parameters compared to the original parameters that were used in the online analysis. The final conclusion from our preliminary investigation is that the second guess calibration parameters yield a more consistent set of results across a range of run numbers. It would be advisable for other users to check the trends of both parameters across runs in their own datasets however.

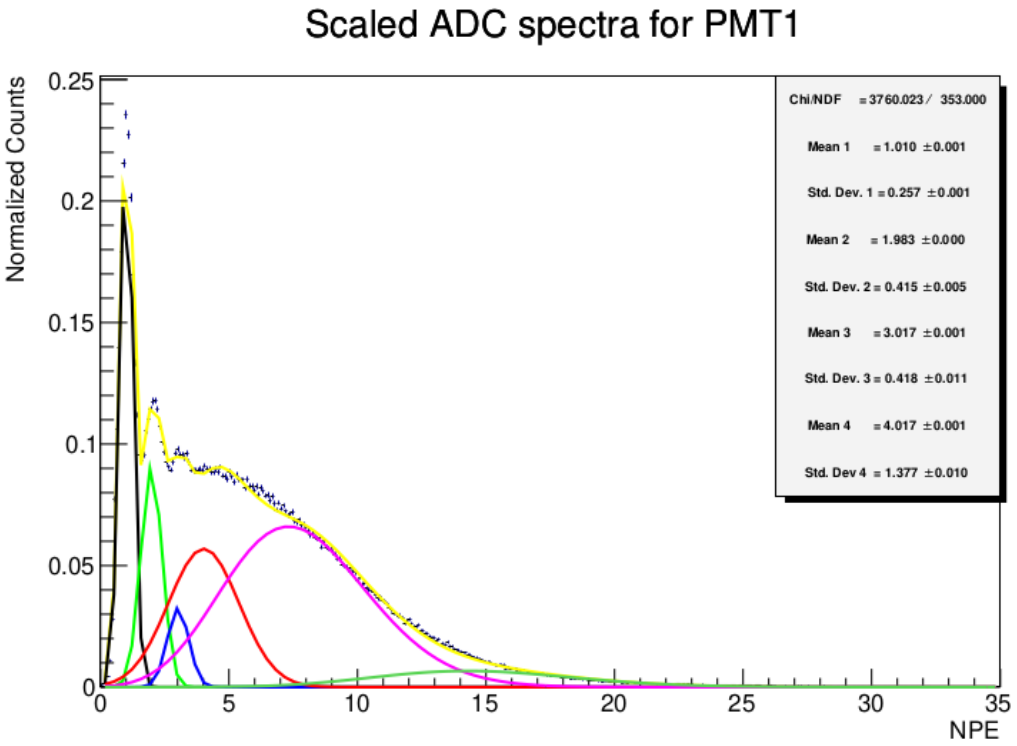

Figure 2: The full distribution of Cherenkov light scaled with the previously obtained calibration parameter for PMT 1.

PMT No	First Guess Parameter	Second Guess Parameter	Old Parameter
1	6.17 ± 0.008	6.237 ± 0.011	6.26
2	6.82 ± 0.010	6.839 ± 0.013	7.30
3	5.36 ± 0.013	5.384 ± 0.015	5.85
4	6.86 ± 0.039	6.893 ± 0.045	6.98

Table 1: The comparison of the new and old calibration parameters.

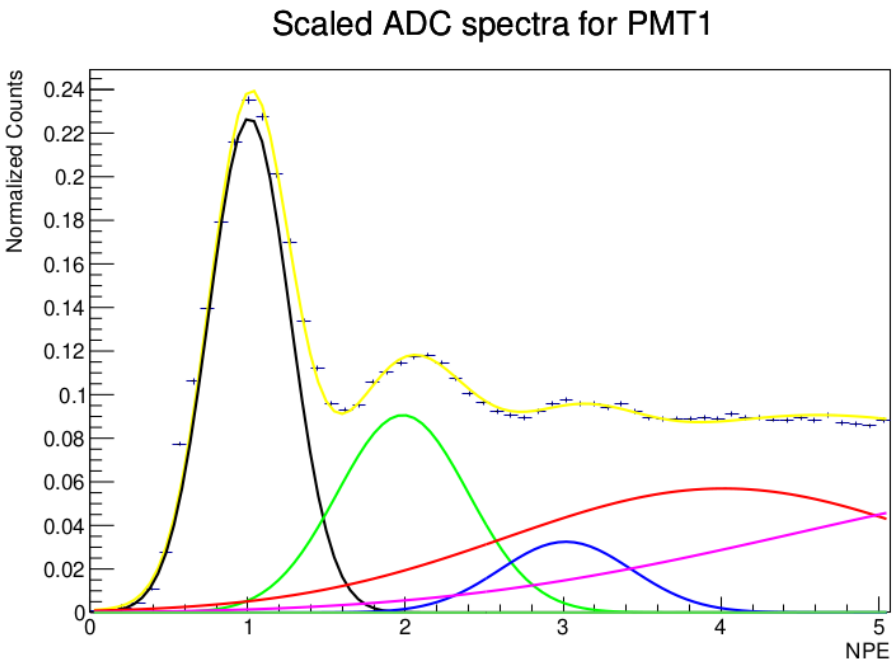

Figure 3: The zoomed (0- 5 NPE) version of the full distribution of Cherenkov light scaled with the previously obtained calibration parameter for PMT 1.

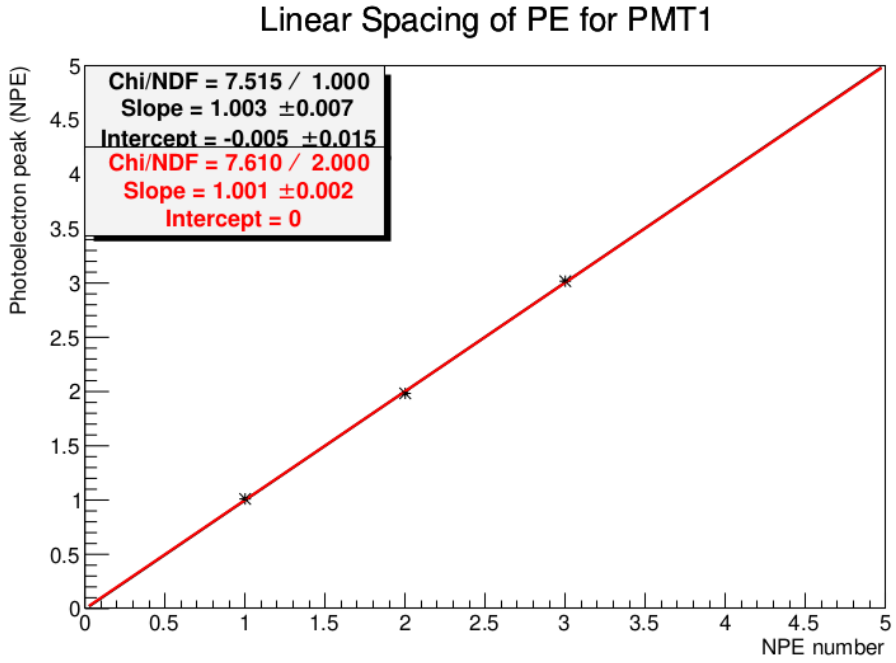

Figure 4: The plot of the linear spacing of the PE from the 1st, 2nd and 3rd Gaussian peaks for PMT 1. The plot has been fitted with two functions (black and red), the resulting fit parameters can be seen in the statistics box in the top left of the plot.

References

[1] George Ryan Ambrose (2019), Blinded by the Light: Commissioning of the Hall C SHMS Heavy Gas Cherenkov Detector. Master's thesis, University of Regina.

[2] Vijay Kumar, KaonLT Collaboration, https://github.com/JeffersonLab/hallc_replay_lt/tree/master/CALIBRATION/shms_hgcer_calib. Accessed, Jul 30, 2020.

[3] ROOT @ cern, https://d35c7d8c.web.cern.ch/developing-tselector. Accessed, Jul 30, 2020.