Drift Chamber and Calorimeter Calibrations

Stephen J D Kay
University of Regina
28/11/2019

1 Introduction

This document outlines the principles and procedures behind the calibration of the HMS and SHMS drift chambers and calorimeters. The scripts utilised to carry out these procedures will be outlined; including instructions on how to run them. Also outlined are the procedures by which the performance of the calibrations can be measured and analysed after they have been carried out. Some of the procedures outlined here are specific to the calibrations for the analysis of the KaonLT and PionLT datasets due to modifications to the relevant scripts. This document effectively serves as a record of how these calibrations were carried out for these experiments.

1.1 Git Repositories

The following explanations and instructions assume that the following git repositories are being utilised -

- **hallc_replay_lt** - Offline branch [1].
- **UTIL_BATCH** - Develop branch [2]

Unless otherwise stated, paths provided are from the loaction to which hallc_replay_lt is cloned.

2 Drift Chamber Calibrations

Each spectrometer in Hall C has a pair of drift chambers that are used for particle tracking and timing. In the current experimental setup ([3] as of 2019), the chambers in each spectrometer are identical in terms of construction. As such, the procedure for calibrating the chambers in each spectrometer is identical. The performance of the drift chambers is very sensitive to the gas mix within the chamber. This gas mix is in turn dependent upon environmental conditions, both within the spectrometer huts and within the gas shed. With this in mind, it is advisable to calibrate the drift chambers regularly, particularly for a long experimental run. For the KaonLT and PionLT experiments, it was decided that a new calibration would be produced for every experimental shift (roughly every 8 hours). The principles and procedure behind the calibration of the drift chambers are outlined in this section.

2.1 Principle

Charged particles that traverse the drift chambers will ionise the gas within the chamber. Electrons from the ionised gas will drift towards sense wires in the chambers. The drift time of these electrons is recorded by a TDC. The purpose of the drift chamber calibration is to find the correct parameters to convert the recorded drift times to drift distances for each wire [4]. The correct t_0 offset for each wire must also be determined. For more detail on how the script carries out the calibration process see [5].

2.2 Procedure

The calibration process for the drift chambers is relatively straightforward. First, the file to be calibrated must be replayed with the parameter "h(p)_using_tzero_ per_wire=0" set in the relevant h(p)dc_cut.param file for the replay script. Note that h(p) refers to the spectrometer arm, "h" for HMS and "p" for SHMS.

As an explicit example, suppose you wish to calibrate the HMS drift chambers. The replay script will most likely look for the file -

PARAM/HMS/DC/hdc_cuts.param

Ensure that the line “h_using_tzero_per_wire=0” is in this file.

The replay script should preferably use a .def file that does not include branches that it does not need in order to speed up the process. The branches for the calorimeter and relevant cherenkovs for the spectrometer you are analysing should be included if you want to utilise PID in the calibration.

Once the replay is complete, the calibration script must be executed on the replayed rootfile. To run the calibration script, navigate to -

- **CALIBRATION/dc_calib/scripts** from the main hallc_replay directory
- Execute the script with the command -
  - **root -l -b -q “main_calib.C(“SPEC”, “ROOTFILE”, RUNNUMBER)”**
  - SPEC - Is the spectrometer to be analysed, either HMS or SHMS
  - ROOTFILE - Is the full path to the replayed rootfile to be analysed
  - RUNNUMBER - Is simply the run number for the replayed run, omitting any leading 0’s - i.e. 05888 should be entered as 5888

The -b and -q arguments for the root command can also be omitted but are generally a good idea to prevent a large number of windows popping up and to simply exit root once the command finishes.

Once this script runs and finishes, it should produce two new .param files if it has run correctly. These new param files are produced in the directory -

- **CALIBRATION/dc_calib/scripts/SPEC_DC_cardlog_RUNNUMBER/**

where SPEC and RUNNUMBER correspond to the arguments provided to the script above. The two new param files are

1. **h(p)dc_calib_RUNNUMBER.param**
2. **h(p)dc_tzero_per_wire_RUNNUMBER.param**

with h(p) referring to whether it was a HMS (h) or SHMS (p) calibration. At this point, these parameter files can be utilised in future replays. Before utilising these new parameter files for detailed analysis, it is highly advisable to verify the quality of the calibration.

2.3 Performance Verification

A second replay that utilises the new .param files from the calibration should be carried out. Drift chamber parameters from this replay can be analysed to assess how well the new calibration performs. Two performance indicators are that for a well calibrated chamber -

1. Drift distance distributions for each plane should be flat
2. “Residuals” for each plane should be a narrow peak centred at 0

Note that the residual is a measure of the difference between the final position for a track and the hit location obtained from each individual chamber.

Of course, checking that these two indicators are good for the post calibration case does not necessarily indicate that the new calibration is better. The initial calibration may already have produced flat drift distance distributions and narrow residual peaks. To truly check whether the new calibration is "better" or not it must be compared to the initial calibration, a short new script was created for this purpose.

The script [6] simply plots the drift distance distributions and residuals for each chamber for the two replays utilised in the calibration process. This script assumes that the replays have been carried out using specific replay scripts (and hence have a set naming convention), but it could easily be modified to read in different files if needed. The script is located in -

- **CALIBRATION/dc_calib/Calibration_Checker/**

This script is intended to be utilised with the batch script (see Section 2.4) created to carry out the calibration process in one step. If you wish to utilise this script but use different replays, you will need to modify the comparison script slightly.

Assuming the relevant modifications to file name formats are made as needed, or they are already correct, then running the script is very straightforward. To run the script, simply navigate to the folder containing the script and execute the command -

- **root -l -b -q "run_DC_Calib_Check.C("RUNNUMBER", "MAXEVENTS", SPEC)"**
- RUNNUMBER - Is, as earlier, simply the run number for the replayed run
- MAXEVENTS - Is the number of events analysed in each replay, typically this will be -1 (i.e all events)
- SPEC - Is again the spectrometer to be analysed, either HMS or SHMS

The script will output a rootfile of the histograms plotted and some comparison plots. These will be saved in the location specified in the script, nominally, these are the normal OUTPUT and HISTOGRAM paths for replayed files but within a "DC_CALIB" sub directory. The comparison plots are saved as a .pdf file and show a comparison between the drift distances and residual distributions, on a chamber by chamber basis, for the pre and post calibration situation. Figures 1 and 2 show some examples of how these comparison plots appear in the output pdf file.

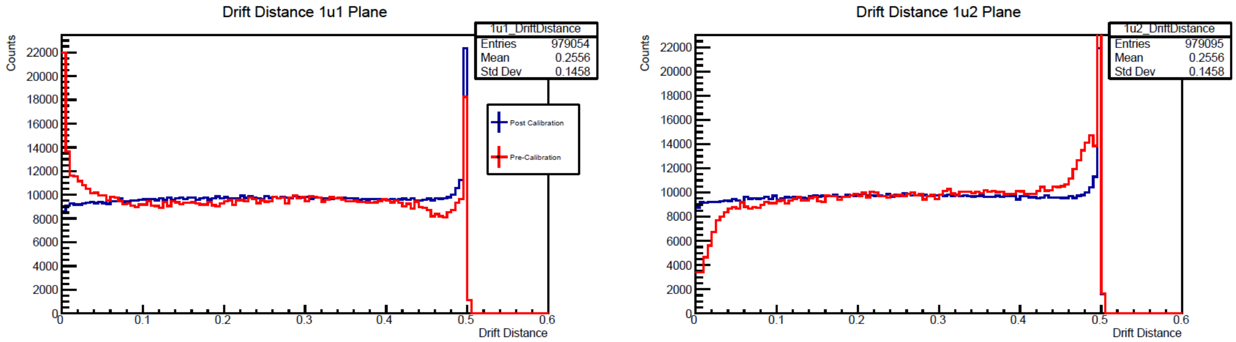


Figure 1: A sample of the output of the comparison script. The drift distance distribution for two chambers (1u1 and 1u2) is shown before and after calibration. Clearly the distribution after the calibration is much flatter, implying an improvement in the calibration. Note that the spike at the end of the distribution is due to events firing multiple wires.

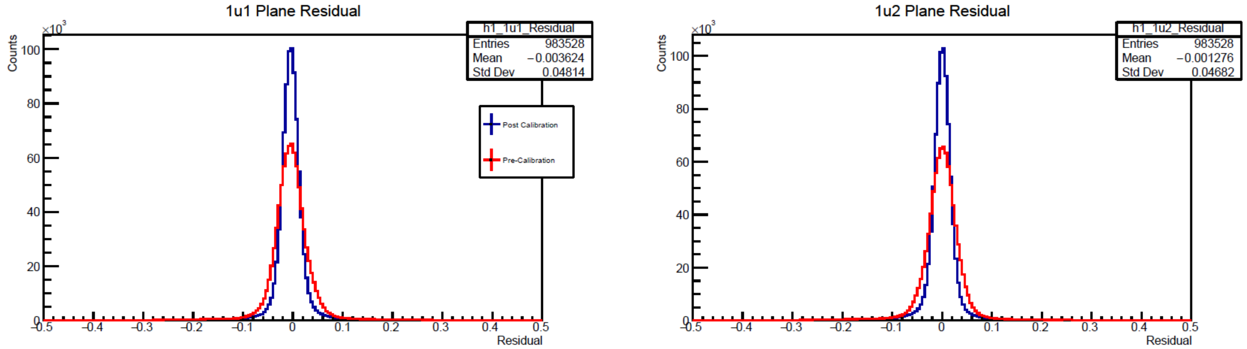


Figure 2: A sample of the output of the comparison script. The residual distribution for two chambers (1u1 and 1u2) is shown before and after calibration. Clearly the distribution after the calibration shows a much sharper peak, this implies the calibration has improved performance.

2.4 Batch Script

Due to the volume of calibrations to be carried out for data analysis, a short shell script to speed up the process was created. This shell script combines all of the commands needed to calibrate and check a single data file from the run into a single command. The intention is that this script could be called in an auger style job on the JLab farm.
The shell script to calibrate the drift chambers is located at -

- **UTIL_BATCH/Analysis_Scripts/DCCalib_Batch.sh** [7].

This script can be executed interactively via the command -

- **./DCCalib_Batch.sh RUNNUMBER SPEC MAXEVENTS**
  - RUNNUMBER - Is, as earlier, simply the run number for the replayed run
  - SPEC - Is again the spectrometer to be analysed, either HMS or SHMS
  - MAXEVENTS - Is the number of events analysed in each replay, this argument is optional, if not provided it is assumed that all events will be analysed

The script will run all of the steps outlined above using some fixed hallc replay scripts. These scripts are defined in the shell script, the default replay scripts that the shell script looks for are included in the main hallc_replay_lt git repository [1]. The shell script also provides some initial setup and will set paths depending upon the environment in which the script is being run. By default, environment setup for the JLab farm, cdaq and machines at the University of Regina are defined. If you wish to run the script in an environment not listed above, then simply add it to the start of the shell script.

In addition to running the script interactively, it can also be provided to a job to be run on the JLab farm. Some example job submission scripts are included in

- **UTIL_BATCH/batch_scripts**

Typically, these scripts have a hard coded list of input runs (a list of run numbers) and simply need to be executed along with the spectrometer to be analysed. The format of these scripts will vary depending upon the job submission system but the basic principle is similar.

As an example, to submit a set farm jobs to calibrate the HMS or SHMS drift chambers on the JLab farm there is already an existing script, **run_batch_DCCalib.sh** [8]. To run a set of runs simply -

1. Modify line 29 of the script to a path to the list of runs to be analysed
    - The run list should be a simple text file with the runs listed one per line
    - Some template run lists are already included in the **InputRunLists** directory included in [2]
2. Execute the script with the spectrometer (HMS or SHMS) provided as an argument
3. If desired, provide a max number of events to analyse per run as a second argument

3 Calorimeter Calibrations

3.1 Principle

The HMS and SHMS calorimeter calibrations are carried out in a very similar manner, any differences in the procedure arise from the slightly different construction of each calorimeter. The purpose of the calibration is to correctly convert the detected ADC signal from the calorimeter into an equivalent energy. The calibration script utilises electron events to perform the calibration. For detected electron events, the deposited energy in the calorimeter, normalised to the particle momentum, should yield a peak centred at 1, if the calorimeter is well calibrated. Further details on the calibration process are included in [9].

3.2 Procedure

Running the calibration script for the calorimeter in each spectrometer is straightforward. Before calibrating, a replay of the run to be calibrated must be carried out. The calorimeter calibration script is designed to calibrate the detector using electrons, as such a run where the spectrometer was set to negative polarity must be used. The replay should use a .def file which contains information from all of the detectors in the spectrometer to be calibrated.

The calibration script requires as an input file with several thresholds and ranges defined as well as the initial gain constants in utilised in the replay. To set up the input file for a specific run number, from the base hallc_replay_lt directory navigate to

- **CALIBRATION/SPEC_cal_calib/**
    - SPEC is the spectrometer to be calibrated, in this case either hms or shms (in lower case)

Within this directory, a symbolic link to the location of the replayed rootfile must be created, this symbolic link should have the name “ROOTfiles”. From the same directory -

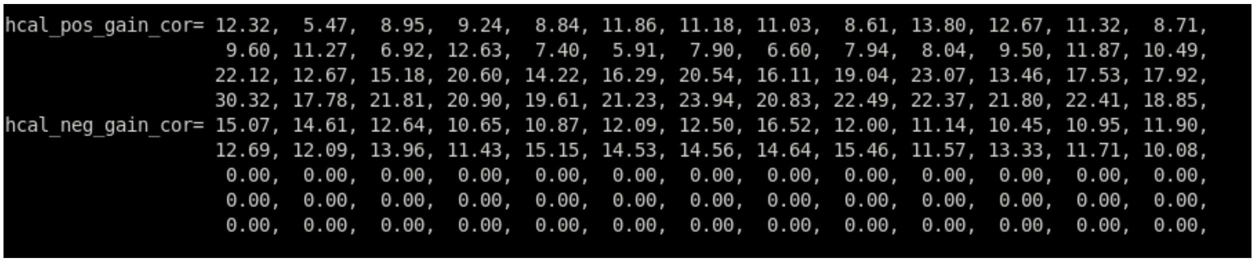
- Copy the template **input.dat** file in this directory to a new file with the format **input_RUNNUMBER.dat**
    - RUNNUMBER - Is simply the run number for the replayed run, omitting any leading 0’s - i.e. 05888 should be entered as 5888
- Replace the block of gain constants in this input file with the block from the relevant parameter file for the replay in question
- The parameter file typically be found in
    - **PARAM/SPEC/CAL/h(p)cal_calib.param**
        - SPEC being HMS or SHMS respectively
        - The corresponding file will be hcal... for the HMS and pcal... for the SHMS

  - The param file name may differ and will depend upon the file specified in standard.database for the chosen replay script
  - The provided path is from the base level hallc_replay_lt directory
  - The block of constants to copy in will be of the form shown in Figures 3 and 4 for the HMS and SHMS respectively

- Modify any of the default thresholds and ranges (e.g. delta range) in the input file as needed
- Move the new input file to the **Input** directory

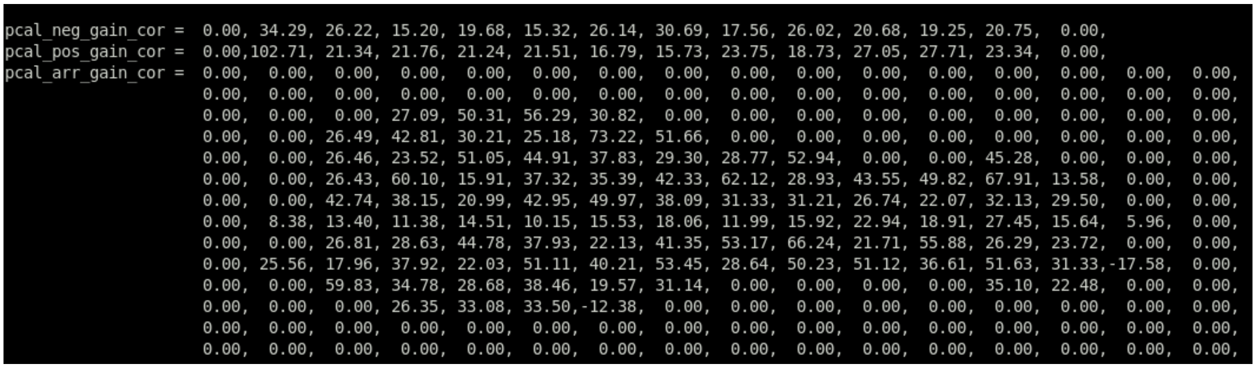
With the input file and symbolic link created and moved to the correct location, the calibration script can be executed. To run the script, simply execute the command -

- **hcana -l -q -b "h(p)cal_calib.cpp+(PREFIX, RUNNUMBER, MAXEVENTS)")**
  - Again, the script will be h or p cal_calib.cpp for the HMS and SHMS respectively
  - Prefix is simply the prefix of the replay rootfile
  - RUNNUMBER - Is again simply the run number in the same format utilised earlier
  - MAXEVENTS - Is the number of events analysed in the replayed run
- E.g. to calibrate the replay "HMS_TEST_5500_100000.root" execute
  - **hcana -l -q -b "h(p)cal_calib.cpp+("HMS_TEST", 5500, 100000")**



```
hcal_pos_gain_cor= 12.32,  5.47,  8.95,  9.24,  8.84, 11.86, 11.18, 11.03,  8.61, 13.80, 12.67, 11.32,  8.71,
                    9.60, 11.27,  6.92, 12.63,  7.40,  5.91,  7.90,  6.60,  7.94,  8.04,  9.50, 11.87, 10.49,
                   22.12, 12.67, 15.18, 20.60, 14.22, 16.29, 20.54, 16.11, 19.04, 23.07, 13.46, 17.53, 17.92,
                   30.32, 17.78, 21.81, 20.90, 19.61, 21.23, 23.94, 20.83, 22.49, 22.37, 21.80, 22.41, 18.85,
hcal_neg_gain_cor= 15.07, 14.61, 12.64, 10.65, 10.87, 12.09, 12.50, 16.52, 12.00, 11.14, 10.45, 10.95, 11.90,
                   12.69, 12.09, 13.96, 11.43, 15.15, 14.53, 14.56, 14.64, 15.46, 11.57, 13.33, 11.71, 10.08,
                    0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                    0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                    0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
```

Figure 3: An example block of gain constants from a HMS calorimeter parameter file that must be copied to the new input file to carry out the calorimeter calibration.



```
pcal_neg_gain_cor =  0.00, 34.29, 26.22, 15.20, 19.68, 15.32, 26.14, 30.69, 17.56, 26.02, 20.68, 19.25, 20.75,  0.00,
pcal_pos_gain_cor =  0.00,102.71, 21.34, 21.76, 21.24, 21.51, 16.79, 15.73, 23.75, 18.73, 27.05, 27.71, 23.34,  0.00,
pcal_arr_gain_cor =  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                     0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                     0.00,  0.00,  0.00, 27.09, 50.31, 56.29, 30.82,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                     0.00,  0.00, 26.49, 42.81, 30.21, 25.18, 73.22, 51.66,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                     0.00,  0.00, 26.46, 23.52, 51.05, 44.91, 37.83, 29.30, 28.77, 52.94,  0.00,  0.00, 45.28,  0.00,  0.00,  0.00,
                     0.00,  0.00, 26.43, 60.10, 15.91, 37.32, 35.39, 42.33, 62.12, 28.93, 43.55, 49.82, 67.91, 13.58,  0.00,  0.00,
                     0.00,  0.00, 42.74, 38.15, 20.99, 42.95, 49.97, 38.09, 31.33, 31.21, 26.74, 22.07, 32.13, 29.50,  0.00,  0.00,
                     0.00,  8.38, 13.40, 11.38, 14.51, 10.15, 15.53, 18.06, 11.99, 15.92, 22.94, 18.91, 27.45, 15.64,  5.96,  0.00,
                     0.00,  0.00, 26.81, 28.63, 44.78, 37.93, 22.13, 41.35, 53.17, 66.24, 21.71, 55.88, 26.29, 23.72,  0.00,  0.00,
                     0.00, 25.56, 17.96, 37.92, 22.03, 51.11, 40.21, 53.45, 28.64, 50.23, 51.12, 36.61, 51.63, 31.33,-17.58,  0.00,
                     0.00,  0.00, 59.83, 34.78, 28.68, 38.46, 19.57, 31.14,  0.00,  0.00,  0.00,  0.00, 35.10, 22.48,  0.00,  0.00,
                     0.00,  0.00,  0.00, 26.35, 33.08, 33.50,-12.38,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                     0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
                     0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,  0.00,
```

Figure 4: An example block of gain constants from a SHMS calorimeter parameter file that must be copied to the new input file to carry out the calorimeter calibration.

Once the script runs, it should produce two files in the **"Output"** directory, a .pdf file and a new parameter file. These files will be named based upon the arguments provided to the calibration script. The parameter file is the new set of gain constants generated by the calibration script, these can be utilised in subsequent replays. The .pdf contains a series of plots that can be used to assess the performance of the calibration script as outlined in the next subsection.

3.3 Performance Verification

As mentioned in Section 3.1, when well calibrated, the normalised energy peak for electrons should be a sharp peak centred at 1. This peak should also be flat across δ too. The pdf file produced by the calibration includes four plots to assess the performance of the calibration script. These plots show the calibrated and uncalibrated normalised energy distributions, as well as a plot of the normalised energy as a function of $\delta(\Delta P)$ for the calibrated data. Also shown is a plot of the shower versus pre-shower normalised energy distribution to assess the purity of the electron selection. An example output pdf file is shown in Figure 5. Broad normalised energy peaks post calibration imply the calibration script has not performed very well, it may be that some of the threshold parameter in the input file need to be adjusted to more closely match the initial case.

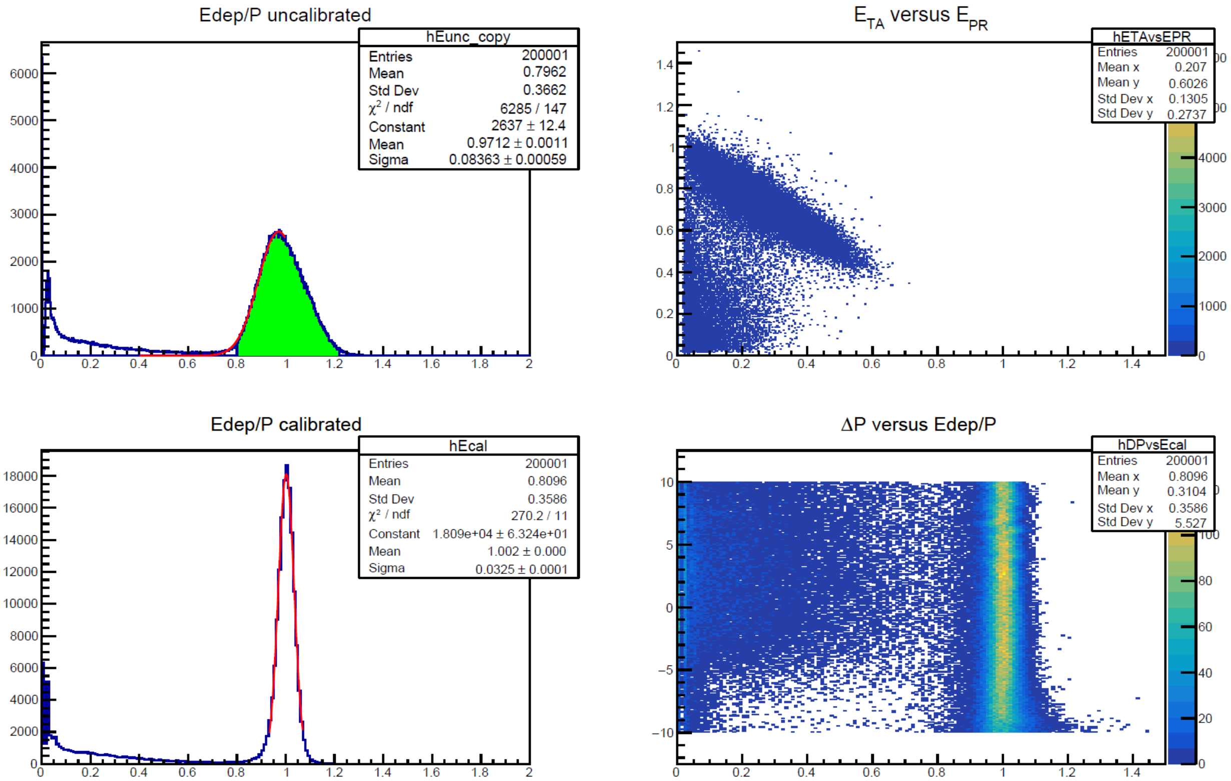


Figure 5: An example of a pdf output file after calibrating the HMS calorimeter. Clearly the normalised energy peak post calibration (bottom left) is improved, the peak is centred more closely to 1 and is much sharper than the initial case (top left). Also, the plot of the normalised energy as a function of $\delta(\Delta P)$ (bottom right) is a relatively straight line barring a few minor wiggles at high δ.

3.4 Batch Script

As with the drift chamber calibrations, a large number of calorimeter calibrations are expected to be performed during the analysis of the KaonLT and PionLT data. As with the drift chamber process, a short shell script was again created to expedite the calibration process. The script to calibrate the calorimeters can be found in -

- **UTIL_Batch/Analysis_Scripts/CalCalib_Batch.sh**

within the git repository [10]. To execute the script interactively, execute the command -

- **./CalCalib_Batch.sh RUNNUMBER SPEC MAXEVENTS**
  - RUNNUMBER - Is, as earlier, simply the run number for the replayed run
  - SPEC - Is again the spectrometer to be analysed, either HMS or SHMS
  - MAXEVENTS - Is the number of events analysed in each replay, this argument is optional, if not provided it is assumed that all events will be analysed

The script will run a short replay of the file in question to determine the initial calibration parameter file that will be examined, this is used later to set up the calibration input file. The script will then run a replay with the specified number of maximum events. Once the replay is complete, the script automatically sets up an input file for the calibration and executes the calibration script as described in Section 3.2. Once completed, the script will move the generated parameter and pdf files to the Output directory of the calibration folder.

As with the drift chamber calibrations, this shell script utilises hcana replay scripts that are included in [1]. To utilise different replay scripts simply modify the relevant lines in the script. Again, this shell script also performs some initial environment setup, the same defaults present in the drift chamber script are present in this script too.

The calorimeter calibration script can also be utilised in a job submission script. A template script is included in the **UTIL_BATCH** git repository [11]. This script is almost identical the the job submission script described for the drift chambers in Section 2.4 and it is executed in an identical manner. Simply modify the run list file as needed and execute the script with the spectrometer (and optionally the max number of events) as an argument.

References

[1] Stephen Kay, Richard Trotta, KaonLT Collaboration. https://github.com/JeffersonLab/hallc_replay_lt/tree/offline. Accessed 21/11/2019.

[2] Stephen Kay, Richard Trotta, KaonLT Collaboration. https://github.com/JeffersonLab/UTIL_BATCH/tree/develop. Accessed 21/11/2019.

[3] Hall C Staff and Editor S. A. Wood Users. *Jefferson Lab Hall C Standard Equipment Manual*. Hall C. https://hallcweb.jlab.org/safety-docs/current/Standard-Equipment-Manual.pdf, 06/11/2019.

[4] Carlos Yero. Hms/shms drift chambers calibration. https://hallcweb.jlab.org/DocDB/0008/000863/003/HallC-Software-Workshop_pdf.pdf. Hall C Software Workshop 2017, Updated June 2019, Hall C DocDB access required.

[5] Carlos Yero. Calibration of the high momentum spectrometer drift chambers. https://hallcweb.jlab.org/DocDB/0008/000842/003/report.pdf. Hall C DocDB access required.

[6] Stephen Kay. https://github.com/JeffersonLab/hallc_replay_lt/blob/offline/CALIBRATION/dc_calib/Calibration_Checker/run_DC_Calib_Check.C. Accessed 21/11/2019.

[7] Stephen Kay. https://github.com/JeffersonLab/UTIL_BATCH/blob/develop/Analysis_Scripts/DCCalib_Batch.sh. Accessed 21/11/2019.

[8] Stephen Kay, Richard Trotta. https://github.com/JeffersonLab/UTIL_BATCH/blob/master/batch_scripts/run_batch_DCCalib.sh. Accessed 21/11/2019.

[9] H. Mkrtchyan et. al. The lead-glass electromagnetic calorimeters for the magnetic spectrometers in Hall C at Jefferson Lab. *Nuclear Instruments and Methods in Physics Research Section A*, 719:85–100, 2013.

[10] Stephen Kay. https://github.com/jeffersonlab/UTIL_BATCH/blob/develop/Analysis_Scripts/CalCalib_Batch.sh. Accessed 28/11/2019.

[11] Stephen Kay, Richard Trotta. https://github.com/sjdkay/UTIL_BATCH/blob/develop/batch_scripts/run_batch_CalCalib.sh. Accessed 28/11/2019.