

# Cointime Peak Stability Studies

Stephen J D Kay  
stephen.kay@uregina.ca  
University of Regina  
17/11/2020

## 1 Introduction

This document outlines the principles and procedures behind the determination of the cointime peak position for different particle species in the SHMS. The prompt and random windows, which are used in subsequent physics analysis of the data, are also defined. These definitions also demonstrate how the coincidence time peak position relates to these timing windows. The scripts utilised to carry out these procedures will be outlined.

## 2 Cointime Peak Stability

The pion and kaon LT experiments (E12-09-11 [1] and E12-19-006 [2]) rely on identifying a scattered electron in the HMS in coincidence with a pion or kaon in the SHMS. The raw coincidence time must be corrected for path length differences through the spectrometer (which arise from deviations from the central trajectory) and for variations of particle velocity [3]. These corrections must be done independently for each hadronic particle species, i.e. we must determine a corrected coincidence time separately for pions, kaons and protons being detected in the SHMS.

In addition to the real (and desired) coincidence events such as  $p(e, e' \pi^+)n$  or  $p(e, e' K^+)\Lambda$ , there will also be random coincidence events. Random coincidences are due to scattered electrons and hadrons from different beam electrons within the resolving time of the coincidence trigger. Usually, they come from different beam bursts and can be separated by a coincidence time cut. But sometimes they come from two electrons in the same beam burst and they can only be removed via accidental window subtraction. The real coincidence events must be identified, by plotting the coincidence time (after applying particle velocity and path length corrections) the position of the coincidence time peak can easily be determined (detailed in Section 2.2). However, the position of the coincidence time peak may drift slightly on a run by run basis due to electronic jitter. For subsequent physics analysis, the position of the real peak (sometimes referred to as “prompt”) is used to determine where cuts should be placed on the cointime spectrum to select out real coincidences. To ensure that all of the real coincidences are retained, it is essential to carefully track the peak position and adjust any selection cuts accordingly.

By tracking the position of the coincidence time peak for each particle species across several runs, any general trends in the amount the peak positions drift or change by can be observed. We would not expect to observe different trends for each particle species, i.e. if the peak position for pions is drifting up for a series of runs, we would also expect it to drift up for kaons and protons. Any large shifts in the peak position should also be examined carefully, it is likely these will correspond to physical changes to the detector electronics, but this should be corroborated with documentation from the experiment logs. The rest of this report outlines the process by which the coincidence time peak was determined and discusses the trends that were observed in the kaon and pion LT separation experiments.

## 2.1 Definitions

The coincidence time (along with many other variables) is determined as part of the hcana replay process using the THcCoinTime.cxx [4] module. In hcana the raw coincidence time is determined via

$$T_{Coin} = (T_{pTDC} + T_{pFP}) - (T_{hTDC} - T_{hFP}),$$

where  $T_{pTDC}$ ,  $T_{hTDC}$ ,  $T_{pFP}$  and  $T_{hFP}$  are the SHMS/HMS TDC and focal plane times respectively. Numerous raw coincidence times are calculated for various trigger and ROC combinations. This is mainly for cross checking. The raw coincidence times need to be corrected for each different particle species. The corrections are determined in lines 288-292 of THcCoinTime.cxx [4]. For the kaon and pion LT experiments the SHMS was set to be the hadronic spectrometer so only the pion, kaon and proton corrections are relevant here. These corrections are applied to the raw coincidence time and a coin time for each particle species is determined.

The physics analysis of the experimental data requires a cut to select the real coincidence peak as well as cuts that select out random coincidence events. The random coincidence events are used to subtract the random background from the real coincidence events. The prompt and random cut windows are set dynamically depending upon the cointime peak position,  $T_{Peak}$ , as well as some other parameters that can be adjusted. For the prompt windows, the only other parameters that are needed are the bunch spacing,  $T_{BS}$ , and an offset value,  $T_{Offs}$ .  $T_{BS}$  is simply the bunch spacing of the beam in nanoseconds and  $T_{Offs}$  is an offset value to ensure that the whole prompt peak is included, this can be varied but is typically 0.25 ns. From these three parameters, the lower and upper bounds of the prompt peak window are defined by Equations (1) and (2).

$$P_{Low} = T_{Peak} - \frac{T_{BS}}{2} - T_{Offs}, \quad (1)$$

$$P_{High} = T_{Peak} + \frac{T_{BS}}{2} + T_{Offs}. \quad (2)$$

Two random windows are needed, one either side of the prompt peak. The random windows require two additional parameters to fully define,  $n_{skip}$  and  $n_{win}$ .  $n_{win}$  is the total number of random windows to include and  $n_{skip}$  is the number of random peaks to skip. Half of the desired windows will be to the left (lower time) of the prompt peak and half will be to the right (higher time). The lower and upper bounds of the two random windows are defined by

$$R_{Low1} = T_{Peak} - \frac{T_{BS}}{2} - T_{Offs} - n_{skip}T_{BS} - \frac{n_{win}}{2}T_{BS}, \quad (3)$$

$$R_{High1} = T_{Peak} - \frac{T_{BS}}{2} - T_{Offs} - n_{skip}T_{BS}, \quad (4)$$

$$R_{Low2} = T_{Peak} + \frac{T_{BS}}{2} + T_{Offs} + n_{skip}T_{BS}, \quad (5)$$

$$R_{High2} = T_{Peak} + \frac{T_{BS}}{2} + T_{Offs} + n_{skip}T_{BS} + \frac{n_{win}}{2}T_{BS}. \quad (6)$$

An example of how these values define the prompt and random windows can be seen in Figure 1. In this figure the red lines correspond to the values from Equations 1 and 2 (and are correspondingly labelled in the figure), clearly, these values enclose the prompt peak in the data. Equations 3/4 and 5/6 enclose 3 random peaks each on either side of the prompt peak, clearly,  $n_{skip} = 2$  and  $n_{win} = 6$  in this case. If desired,  $n_{skip}$  and  $n_{win}$  could easily be tweaked to select a different number of random windows. For the run in question (6650, Winter 2018), the bunch spacing was 2.004 ns.

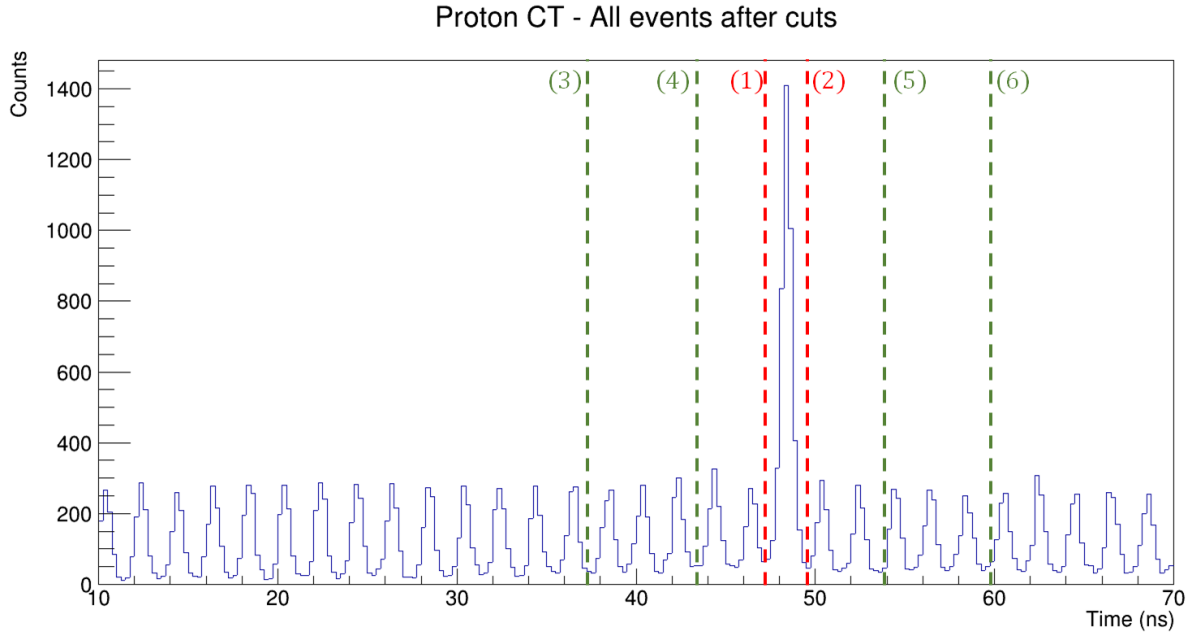


Figure 1: Proton coincidence time after proton PID cuts have been applied. The coloured dashed lines indicate the values calculated from Equations 1 – 6 for this particular run (6650). The line numbered (1) corresponds to the value from Equation 1 and so on.

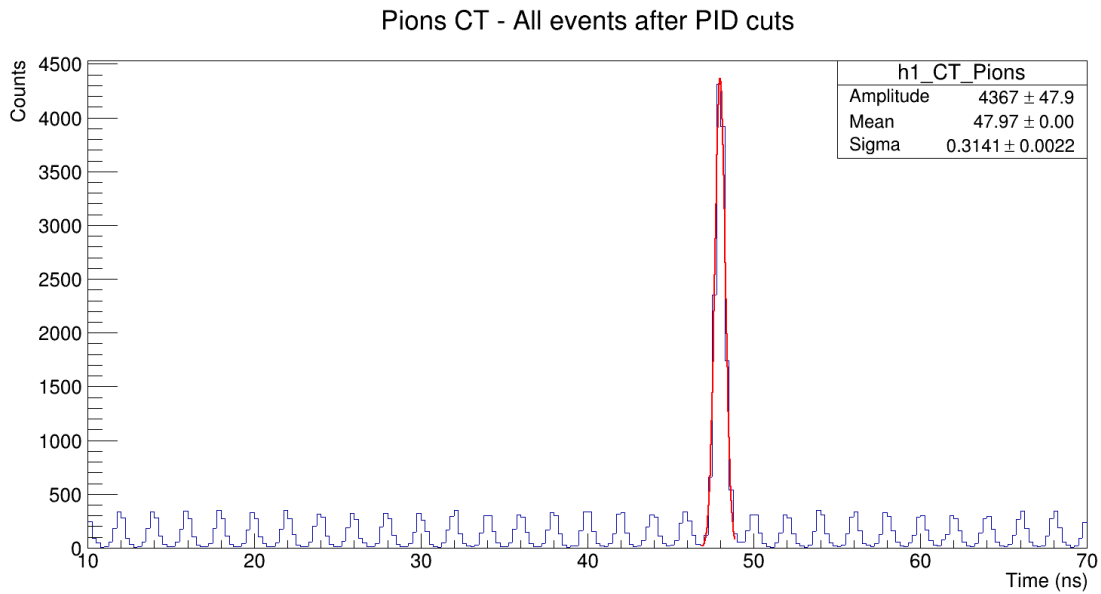


Figure 2: Pion coincidence time distribution with a simple gaussian fitted to the prompt peak. The resulting fit parameters can be seen in the top right of the plot.

## 2.2 Analysis and Fitting

As defined in Equations 1 – 6, the prompt and random timing windows are calculated dynamically based upon the cointime peak position. As such, determining this peak position accurately is an important step in establishing a set of good timing windows for further physics analysis. During the pion and kaon LT experimental runs the cointime peak was determined in a very approximate manner during the online analysis. Shift workers would simply roughly determine the peak position by eye from plots of the coincidence time and set the window positions based upon this. In order to establish how accurate this was, and to refine it as needed, a set of new analysis scripts was created to determine the cointime peak position in a more reliable and systematic manner.

The process of determining the cointime peak position for a single run is a two step process involving a python script and a root macro. The general process and logic of the scripts will be explained here, for details on how to run the scripts see [5]. The first new script [6] makes use of the new python analysis framework [7, 8] to apply selection cuts to the data and trims down a large hcana replay file to a far smaller root file which contains only the desired information from the original replay file. In this case, the output is a slimmed down root file with a handful of trees containing some timing and PID information. Each tree corresponds to a different set of cuts applied to the data. Many of the generated trees are simply for debugging purposes, of relevance to the next analysis step are the “Pions\_All”, “Kaons\_All” and “Protons\_All” trees which simply contain the pion/kaon/proton cointime information for events which passed the corresponding PID cuts (defined in [9]).

The output of the first script is fed into a second script [10] which is simply a short root macro. This script simply fills histograms of the cointime data for each particle species and fits a simple Gaussian to the prompt peak in the data. Satisfactory fits are achieved by forcing some constraints on the fitting ranges. The mean,  $\mu$ , of the Gaussian fit is initially set to be equal to the value of the bin with the largest number of entries. For any run with anything more than a few events, this will almost certainly be a bin which roughly corresponds to the position of the coincidence time peak. The mean of the fit is only allowed to vary within  $\pm 0.5$  of this value. The amplitude of the Gaussian fit is also restricted to vary between  $\frac{1}{2}$  and  $2 \times$  the number of entries in the bin with the largest number of entries. Finally, the standard deviation,  $\sigma$ , of the fit is limited to vary between 0.1 and 2. The fit is applied to each particle species *only* in the range  $\mu_{init} - 1$  to  $\mu_{init} + 1$  where  $\mu_{init}$  is the initial mean value. Without this, MINUIT has a tendency to produce broad, flat peaks. An example of a fit to the pion cointime distribution from run 6650 can be seen in Figure 2. A fit is produced for all three particles species and the resulting three fits are saved to a .pdf and .root file, these files are named dynamically based upon the input arguments.

To process a large number of files in this manner, a simple shell script [11] that can be used as part of a job submission script [12] on the JLab farm was created. This script simply carries out a replay for a given run and processes the python script and root macro described above. To examine how the position of the cointime peak varies across a kinematic, an additional shell script and root macro were created [13, 14]. The shell script simply compiles the fit data for a range of runs and then feeds it to the root macro which plots the data. The macro plots the peak position and width by run number for all three particle species and saves the plots as a .pdf and .root file (which is named dynamically by the input arguments). Some examples of these plots for a selection of kinematic settings can be seen in Section 2.3.

If the new cointime peak values are to be used in further analyses, they must be placed in a parameter file for future scripts to read in. To this end, a final python script [15] was created which compiles all of the data from a set of processed kinematics (which are fed in as an argument) and produces a new parameter file. The produced file is sorted by run number and contains all of the relevant parameters needed to define the prompt and random windows. The parameter file is simply a comma separated variable file with each line corresponding to a run numbers (or block of run numbers) with each comma separated field being a different parameter. An example of several lines from a produced .csv file can be seen in Figure 3.

```

Run_Start,Run_End,Bunch_Spacing,Coin_Offset,nSkip,nWindows,Pion_Prompt_Peak,Kaon_Prompt_Peak,Proton_Prompt_Peak,RF_Offset
4865,4866,4.008,0.250,2,6,47.306,47.138,49.272,-1.030
4866,4867,4.008,0.250,2,6,47.357,47.163,49.328,-1.030
4867,4868,4.008,0.250,2,6,47.327,47.214,49.326,-1.030
4868,4869,4.008,0.250,2,6,47.283,47.176,49.255,-1.030
4871,4872,4.008,0.250,2,6,44.704,44.526,46.631,0.119
4873,4874,4.008,0.250,2,6,44.715,44.612,46.659,0.119
4874,4875,4.008,0.250,2,6,44.731,44.635,46.707,0.119
4875,4876,4.008,0.250,2,6,44.399,44.346,46.466,0.119
4876,4877,4.008,0.250,2,6,44.420,44.354,46.516,0.119

```

1	Run_Start	Run_End	Bunch_Spacing	Coin_Offset	nSkip	nWindows	Pion_Prompt_Peak	Kaon_Prompt_Peak	Proton_Prompt_Peak	RF_Offset
2	4865	4866	4.008	0.250	2	6	47.306	47.138	49.272	-1.030
3	4866	4867	4.008	0.250	2	6	47.357	47.163	49.328	-1.030
4	4867	4868	4.008	0.250	2	6	47.327	47.214	49.326	-1.030
5	4868	4869	4.008	0.250	2	6	47.283	47.176	49.255	-1.030
6	4871	4872	4.008	0.250	2	6	44.704	44.526	46.631	0.119
7	4873	4874	4.008	0.250	2	6	44.715	44.612	46.659	0.119
8	4874	4875	4.008	0.250	2	6	44.731	44.635	46.707	0.119
9	4875	4876	4.008	0.250	2	6	44.399	44.346	46.466	0.119
10	4876	4877	4.008	0.250	2	6	44.420	44.354	46.516	0.119

Figure 3: An example of an output .csv parameter file. The top part of the image shows the raw text in the file whereas the bottom part of the image shows how this file should be interpreted, each line in the file is a row and each comma separated value is effectively a column of data.

## 2.3 Examples

All of the kinematic settings from the kaon LT and pion LT experimental runs were processed using the scripts described in Section 2.2. Generally, most kinematics did not exhibit any clear trends in the cointime peak position. Figure 4 shows a typical distribution of the pion cointime peak position across a range of run numbers, whilst the position does shift slightly point to point, there is no clear trend for the peak position to increase or decrease. This lack of any trend is also observed in the kaon cointime peak position for the same kinematic setting as seen in Figure 5. However, in a select few kinematic settings, there is evidence of a slight increasing (Figure 6) or decreasing (Figure 7) trend. There is also sometimes quite a clear shift in the peak position after a large gap of runs as seen in Figure 6. Some kinematic settings had large gaps in run numbers due to the running conditions, this was typically due to “moving on” to a different setting to ensure it was measured before going back to take more data at the previous setting if time permitted.

Typically, any shifts or trends seen for one particle species were also observed in the others, on some occasions the proton cointime peak position did not shift in exactly the same manner as the pion/kaon peak position. However, this is likely just due to the more limited statistics available for the proton on some runs. This is reassuring as any trends or shifts due to the electronics should be present in all three particle species. The reason for the appearance of any of these trends is not immediately clear. However, it is likely just due to electronic jitter. By measuring the cointime peak position on a run by run basis any shifts can be accounted for in subsequent analyses anyway. It is also worth noting that the  $y$  axis scales of Figures 4–7 vary, what appear to be “large” shifts from the plot are actually typically only shifts of  $\sim 0.1$  ns at most. The shifts and trends observed are not particularly concerning. Using the script discussed at the end of Section 2.2, a new parameter file was created with timing parameters defined for every production run. This can be used in future physics analysis to accurately select out the prompt and random windows.

## 3 Summary

A new, systematic approach to determine the position of the pion, kaon and proton prompt coincidence time peak was established. This new approach utilises a series of python and root scripts to determine the prompt coincidence time peak. In addition, these scripts are capable of creating a new parameter file for use in subsequent physics analysis, with the parameters in this file more accurate prompt and random windows can be established. The new scripts can also be utilised to investigate any run dependent trends in the position of the coincidence time peak. Such an investigation was carried out for all of the kinematic points taken during the kaon and pion LT experiments, no consistent or significant trends were observed. Any minor shifts in the peak position can be accounted for in future with the new parameter file that was created. This process can also be utilised in future data taking to provide a more accurate and reliable value of the cointime peak position in the online analysis. This will improve the quality of the preliminary online analysis.

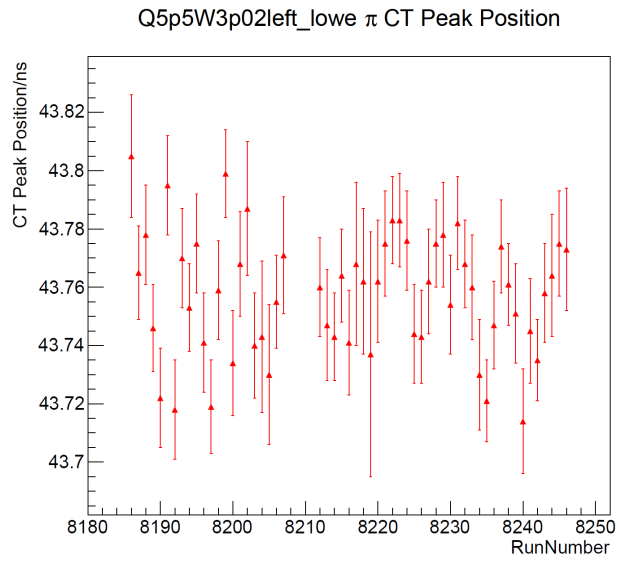


Figure 4: Pion cointime peak position as a function of run number for the  $Q^2 = 5.5, W = 3.02$ , left,  $\epsilon_{low}$  kinematic setting. Although the peak position shifts run to run, there is no clear upward or downward trend apparent in the data.

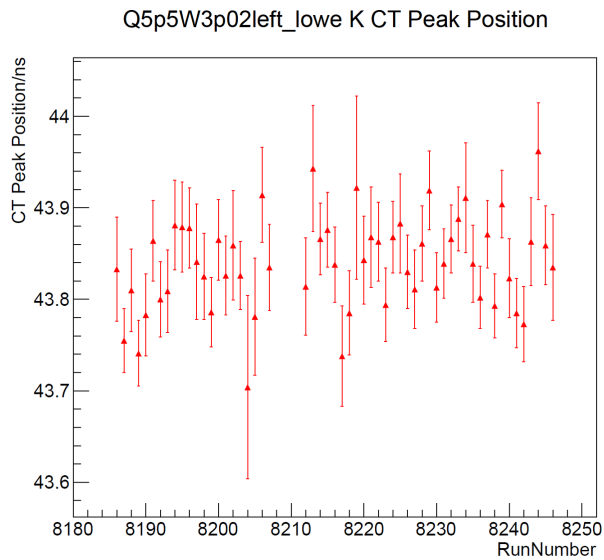


Figure 5: Kaon cointime peak position as a function of run number for the  $Q^2 = 5.5, W = 3.02$ , left,  $\epsilon_{low}$  kinematic setting. As in Figure 4 there is no clear trend to the data.

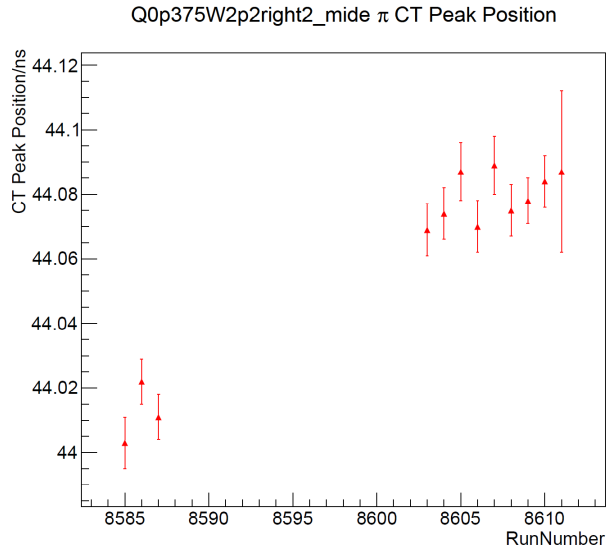


Figure 6: Pion cointime peak position as a function of run number for the  $Q^2 = 0.375$ ,  $W = 2.2$ , right2,  $\epsilon_{mid}$  kinematic setting. A clear upwards trend is apparent in the data. This trend appears to occur as a distinct shift in the peak position in between the two sets of run numbers.

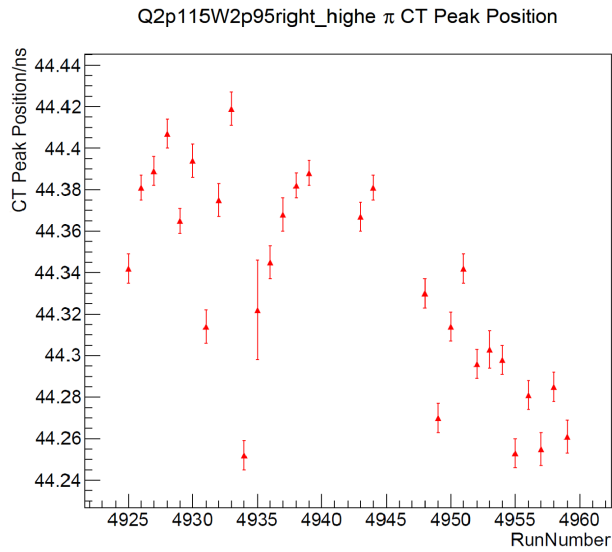


Figure 7: Pion cointime peak position as a function of run number for the  $Q^2 = 2.115$ ,  $W = 2.95$ , right,  $\epsilon_{high}$  kinematic setting. A slight downward trend in the cointime peak position is apparent.



## References

- [1] T. Horn, G.M. Huber, P. Markowitz. UPDATE E12-09-011: Studies of the L-T Separated Kaon Electroproduction Cross Section from 5-11 GeV. [https://www.jlab.org/exp\\_prog/PACpage/PAC38/proposals/Previously\\_Approved/E12-09-011\\_Update.pdf](https://www.jlab.org/exp_prog/PACpage/PAC38/proposals/Previously_Approved/E12-09-011_Update.pdf). Accessed 06/11/20.
- [2] D. Gaskell, T. Horn, G.M. Huber. Update on E12-06-101: Measurement of the Charged Pion Form Factor to High Q<sup>2</sup> and E12-07-105: Scaling Study of the LT Separated Pion Electroproduction Cross Section at 11 GeV. [https://www.jlab.org/exp\\_prog/proposals/19/E12-19-006.pdf](https://www.jlab.org/exp_prog/proposals/19/E12-19-006.pdf). Accessed 06/11/20.
- [3] T. Horn. *The Pion Charge Form Factor Through Pion Electroproduction*. PhD thesis, University of Maryland, 2006.
- [4] Carlos Yero. <https://github.com/JeffersonLab/hcana/blob/develop/src/THcCoinTime.cxx>. Accessed 17/11/2020.
- [5] Stephen Kay. [https://github.com/JeffersonLab/UTIL\\_KAONLT/tree/offline/scripts/CoinTimePeak/README](https://github.com/JeffersonLab/UTIL_KAONLT/tree/offline/scripts/CoinTimePeak/README). Accessed 17/11/20.
- [6] Stephen Kay. [https://github.com/JeffersonLab/UTIL\\_KAONLT/tree/offline/scripts/CoinTimePeak/src/CoinTimePeak.py](https://github.com/JeffersonLab/UTIL_KAONLT/tree/offline/scripts/CoinTimePeak/src/CoinTimePeak.py). Accessed 17/11/20.
- [7] Stephen Kay. Python Analysis Code Update. [https://redmine.jlab.org/attachments/download/1000/Analysis\\_Meeting\\_14\\_05\\_20.pdf](https://redmine.jlab.org/attachments/download/1000/Analysis_Meeting_14_05_20.pdf). HallC Commissioning Analysis Meeting, 13/05/20, Accessed 17/11/20.
- [8] Richard Trotta. Kaon LT Status Update. [https://redmine.jlab.org/attachments/download/998/KaonLTMeeting\\_5\\_13\\_20.pdf](https://redmine.jlab.org/attachments/download/998/KaonLTMeeting_5_13_20.pdf). HallC Commissioning Analysis Meeting, 13/05/20, Accessed 17/11/20.
- [9] Stephen Kay. [https://github.com/JeffersonLab/UTIL\\_KAONLT/blob/offline/DB/CUTS/run\\_type/coinpeak.cuts](https://github.com/JeffersonLab/UTIL_KAONLT/blob/offline/DB/CUTS/run_type/coinpeak.cuts). Accessed 17/11/20.
- [10] Stephen Kay. [https://github.com/JeffersonLab/UTIL\\_KAONLT/blob/offline/scripts/CoinTimePeak/PlotCoinPeak.C](https://github.com/JeffersonLab/UTIL_KAONLT/blob/offline/scripts/CoinTimePeak/PlotCoinPeak.C). Accessed 17/11/20.
- [11] Stephen Kay. [https://github.com/JeffersonLab/UTIL\\_BATCH/blob/develop/Analysis\\_Scripts/CTPeak\\_Analysis.sh](https://github.com/JeffersonLab/UTIL_BATCH/blob/develop/Analysis_Scripts/CTPeak_Analysis.sh). Accessed 17/11/20.
- [12] Stephen Kay. [https://github.com/JeffersonLab/UTIL\\_BATCH/blob/develop/batch\\_scripts/run\\_batch\\_CTPeak\\_Analysis.sh](https://github.com/JeffersonLab/UTIL_BATCH/blob/develop/batch_scripts/run_batch_CTPeak_Analysis.sh). Accessed 17/11/20.
- [13] Stephen Kay. [https://github.com/sjdkay/UTIL\\_KAONLT/blob/offline/scripts/CoinTimePeak/AnalyseKinematic\\_CTPeak.sh](https://github.com/sjdkay/UTIL_KAONLT/blob/offline/scripts/CoinTimePeak/AnalyseKinematic_CTPeak.sh). Accessed 17/11/20.
- [14] Stephen Kay. [https://github.com/sjdkay/UTIL\\_KAONLT/blob/offline/scripts/CoinTimePeak/PlotKinematic.C](https://github.com/sjdkay/UTIL_KAONLT/blob/offline/scripts/CoinTimePeak/PlotKinematic.C). Accessed 17/11/20.
- [15] Stephen Kay. [https://github.com/sjdkay/UTIL\\_KAONLT/blob/offline/scripts/CoinTimePeak/Paramfile.py](https://github.com/sjdkay/UTIL_KAONLT/blob/offline/scripts/CoinTimePeak/Paramfile.py). Accessed 17/11/20.