Overview of the CE Calibration Analysis using PDHD Pulser Data

Karla Téllez Girón Flores
Wenqiang Gu
Jay Hyun Jo
Xin Qian

@BrookhavenLab

Wire-Cell General Meeting, 10/10/2024

Dataset

- **NP04 TPC Electronics Studies Runs**.
- **Pulser Calibration Run 28286 from July of 2024.**
- **DAC = 30.**
- **7.8 mV/fC LArASIC gain.**
- **2 μs CE Shaping Time.**
- **LArASIC Output Mode: Single-ended.**

Waveform Fitting

1. A peak finding algorithm finds and isolates the positive and negative peaks for each channel in data.

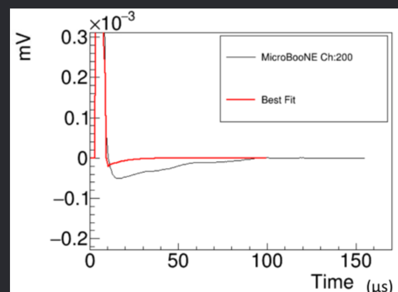
2. Averaging of all pulses for <u>each channel</u> effectively gets rid of most noisy tails.

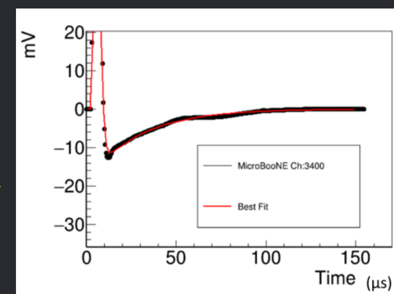
3. Fitting waveforms using a two-step process
   - Pre-fitting using Ideal Electronics Response Function from 0-10 μs.
   - Fit using *New* Electronics Response Function* from 0-50 μs.

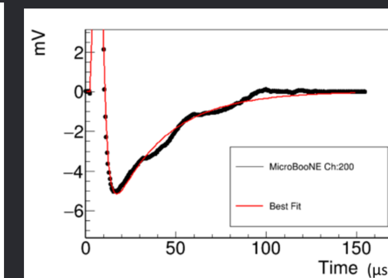
*New Electronics Response Function

$$T(s) = \frac{(s+k_3)(s+k_5)}{(s+k_4)(s+k_6)} \times \frac{A}{(s+p_0)(p_{1i}^2+(p_{1r}+s)^2)(p_{2i}^2+(p_{2r}+s)^2)}$$

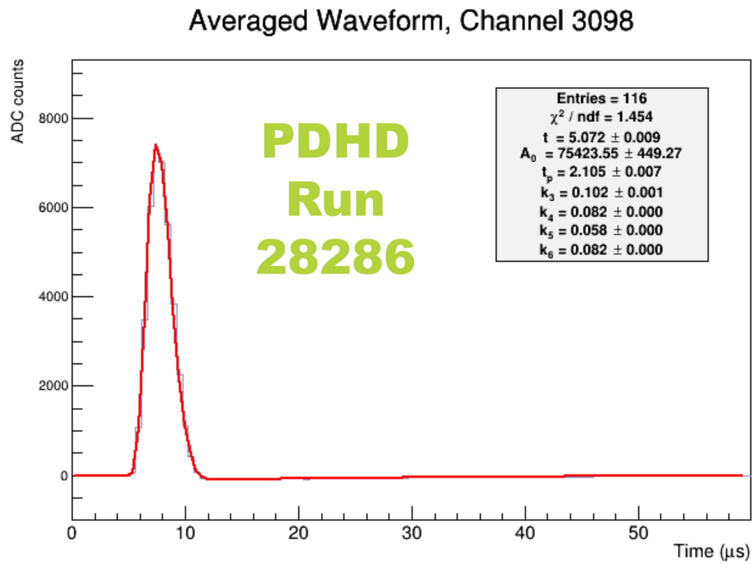
```
double response(double *x, double *par){
  Double_t t = x[0]-par[0];
  Double_t A0 = par[1];
  Double_t tp = par[2];
  Double_t CT = 1./1.996;
  Double_t A = A0 * 2.7433/pow(tp*CT,4);
  Double_t p0 = 1.477/tp/CT;
  Double_t pr1 = 1.417/tp/CT;
  Double_t pr2 = 1.204/tp/CT;
  Double_t pi1 = 0.598/tp/CT;
  Double_t pi2 = 1.299/tp/CT;


  double k3 = par[3];
  double k4 = par[4];
  double k5 = par[5];
  double k6 = par[6];

  double value = A*((-(k3*k4) + pow(k4,2) + k3*k5 - k4*k5)/(exp(k4*t)*(k4 - k6)*(k4 - p0)*(pow(k4,2) + pow(pi1,2) - 2*k4*pr1 + pow(pr1,2))*(pow(k4,2) + pow(pi2,2) -
      2*k4*pr2 + pow(pr2,2))) +
         (-(k3*k5) + k3*k6 + k5*k6 - pow(k6,2))/(exp(k6*t)*(k4 - k6)*(k6 - p0)*(pow(k6,2) + pow(pi1,2) - 2*k6*pr1 + pow(pr1,2))*(pow(k6,2) + pow(pi2,2) - 2*k6*pr2 +
             pow(pr2,2))) +
         (-(k3*k5) + k3*p0 + k5*p0 - pow(p0,2))/(exp(p0*t)*(k4 - p0)*(-k6 + p0)*(pow(p0,2) + pow(pi1,2) - 2*p0*pr1 + pow(pr1,2))*(pow(p0,2) + pow(pi2,2) - 2*p0*pr2
             + pow(pr2,2))) +
     (pi1*((pow(pi1,2) + pow(pr1,2))*(2*k6*(pow(pi1,2) + pow(pr1,2))*(pr1 - pr2) + k6*p0*(-pow(pi1,2) + pow(pi2,2) - pow(pr1,2) + pow(pr2,2)) + (pow(pi1,2) +
         pow(pr1,2))*(pow(pi1,2) - pow(pi2,2) + (pr1 - pr2)*(2*p0 - 3*pr1 + pr2)) +
              k5*(2*pow(pi1,2)*(-2*pr1 + pr2) + p0*(pow(pi1,2) - pow(pi2,2) - 3*pow(pr1,2) + 4*pr1*pr2 - pow(pr2,2)) + 2*pr1*(pow(pi2,2) + 2*pow(pr1,2) - 3*pr1*pr2
                 + pow(pr2,2)) +
                 k6*(pow(pi1,2) - pow(pi2,2) + (pr1 - pr2)*(2*p0 - 3*pr1 + pr2)))) + k4*((pow(pi1,2) + pow(pr1,2))*(2*(pow(pi1,2) + pow(pr1,2))*(pr1 - pr2) +
                     p0*(-pow(pi1,2) + pow(pi2,2) - pow(pr1,2) + pow(pr2,2))) +
              k5*(2*k6*(pow(pi1,2) + pow(pr1,2))*(pr1 - pr2) - k6*p0*(pow(pi1,2) - pow(pi2,2) + pow(pr1,2) - pow(pr2,2)) + (pow(pi1,2) + pow(pr1,2))*(pow(pi1,2) -
                 pow(pi2,2) + (pr1 - pr2)*(2*p0 - 3*pr1 + pr2))) +
              k6*(-pow(pi1,4) + pow(pi1,2)*(pow(pi2,2) - 2*pow(pr1,2) + 2*p0*pr2 + pow(pr2,2)) + pr1*(pr1*(pow(pi2,2) - pow(pr1,2) + pow(pr2,2)) - 2*p0*(pow(pi2,2)
                 - pr1*pr2 + pow(pr2,2)))))) +
           k3*(-((pow(pi1,2) + pow(pr1,2))*(4*pow(pi1,2)*pr1 - 2*pow(pi2,2)*pr1 - 4*pow(pr1,3) - 2*pow(pi1,2)*pr2 + 6*pow(pr1,2)*pr2 - 2*pr1*pow(pr2,2) +
               p0*(-pow(pi1,2) + pow(pi2,2) + 3*pow(pr1,2) - 4*pr1*pr2 + pow(pr2,2)) +
                   k6*(-pow(pi1,2) + pow(pi2,2) - (pr1 - pr2)*(2*p0 - 3*pr1 + pr2)))) + k5*(-pow(pi1,4) + pow(pi1,2)*(pow(pi2,2) - 4*p0*pr1 + 10*pow(pr1,2) +
```

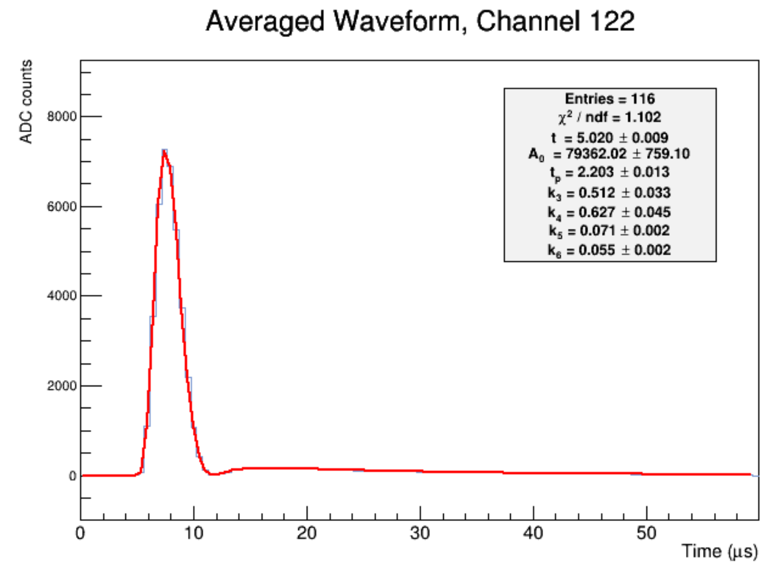


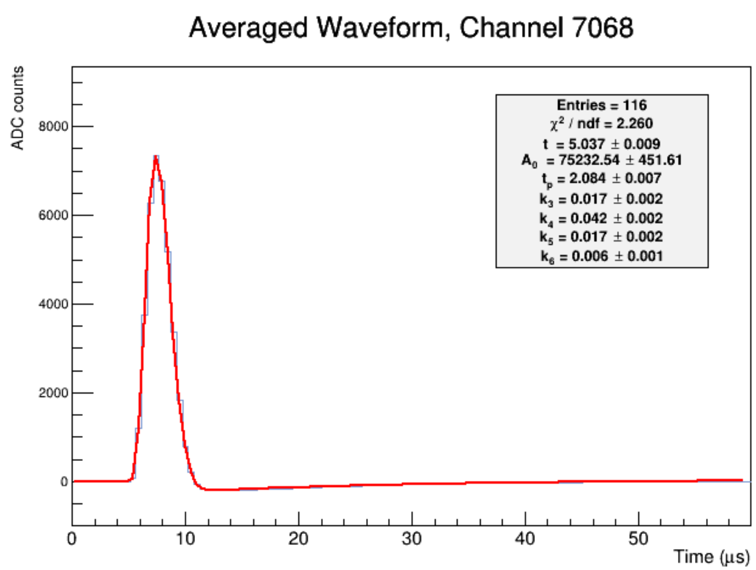


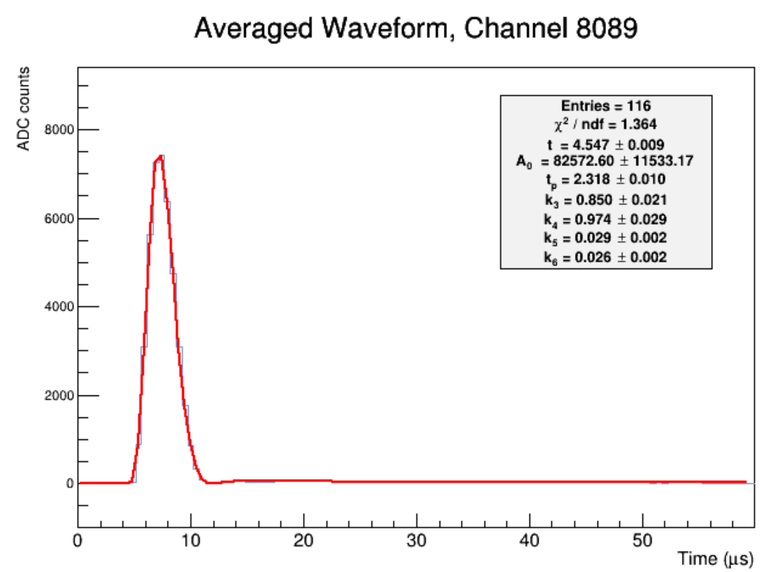


Examples of Fitted Waveforms



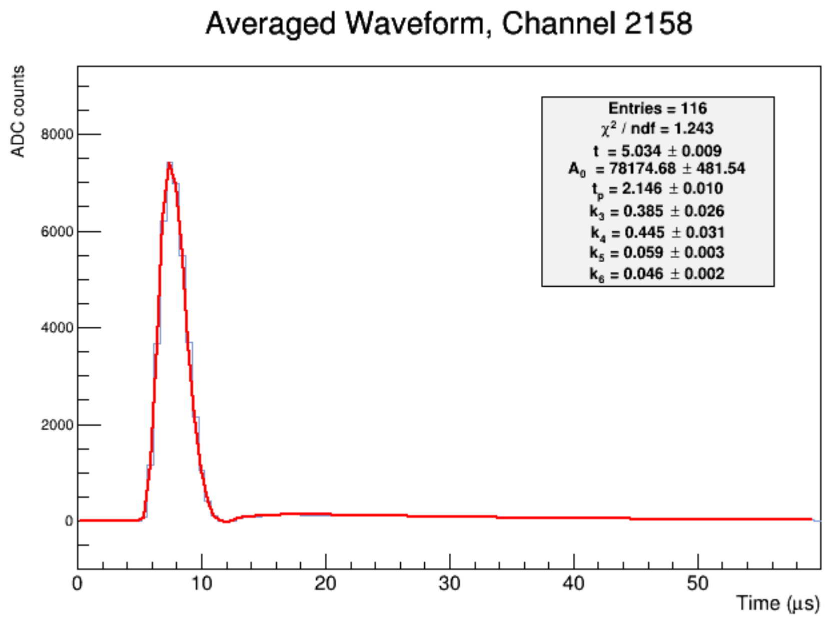


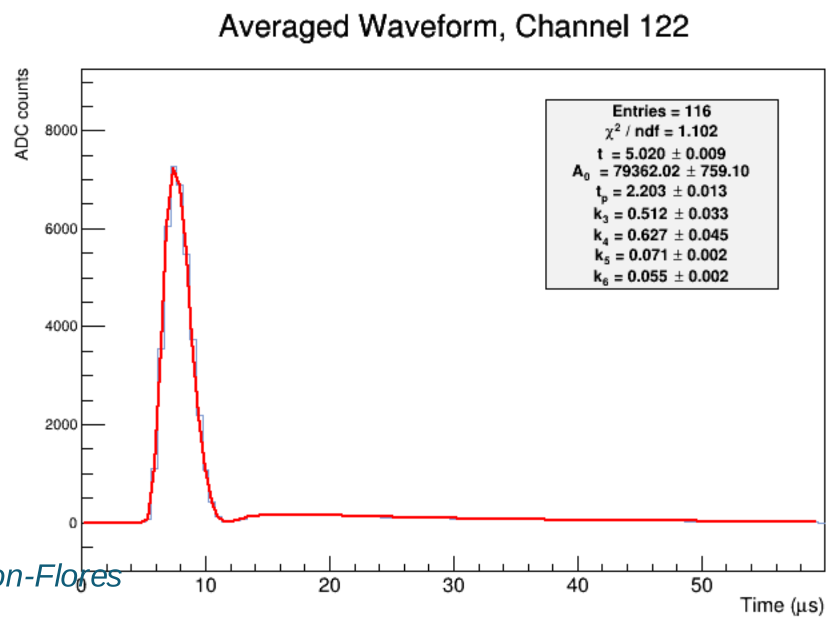




Waveform Correction

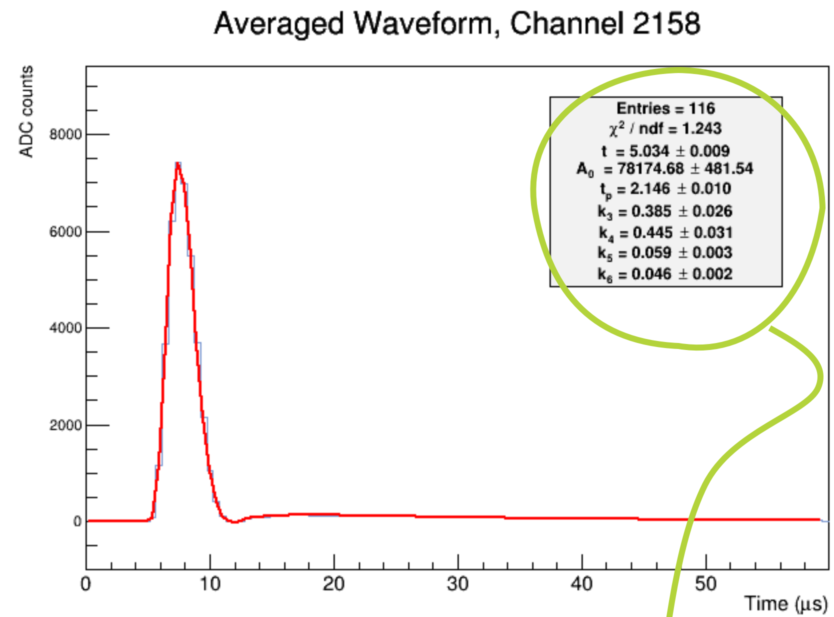
1. **Run the full fitter on our dataset.**
2. Extract fit parameters.
3. Run the waveform correction (Wire-Cell).
4. Fit corrected waveforms with Ideal Electronics Response Function.
5. Retrieve Amplitude and Shaping Time.
6. Convert Amplitude to Gain.





Waveform Correction

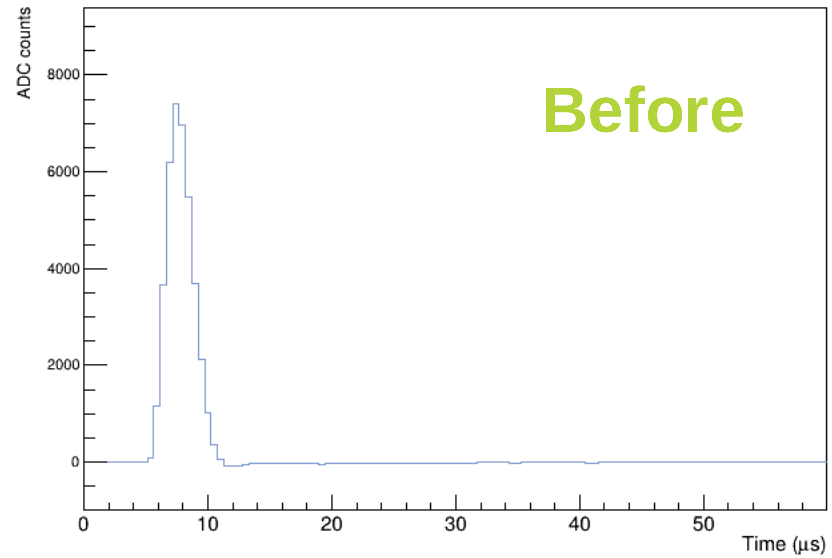
1. Run the full fitter on our dataset.
2. **Extract fit parameters.**
3. Run the waveform correction (Wire-Cell).
4. Fit corrected waveforms with Ideal Electronics Response Function.
5. Retrieve Amplitude and Shaping Time.
6. Convert Amplitude to Gain.

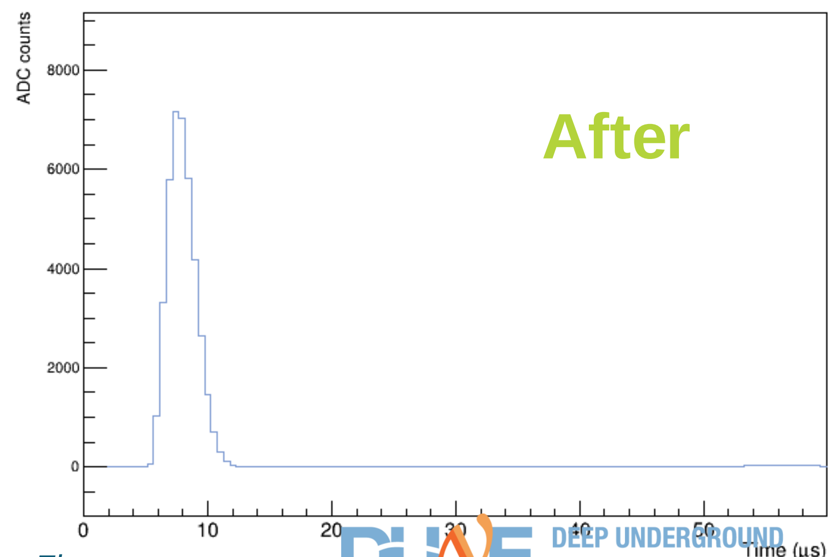


"channels": [
2158
],
"gain": 1.345697e-12,
"shaping": 2146.0,
"k3": 0.3846,
"k4": 0.4447,
"k5": 0.05915,
"k6": 0.04606
},

Waveform Correction

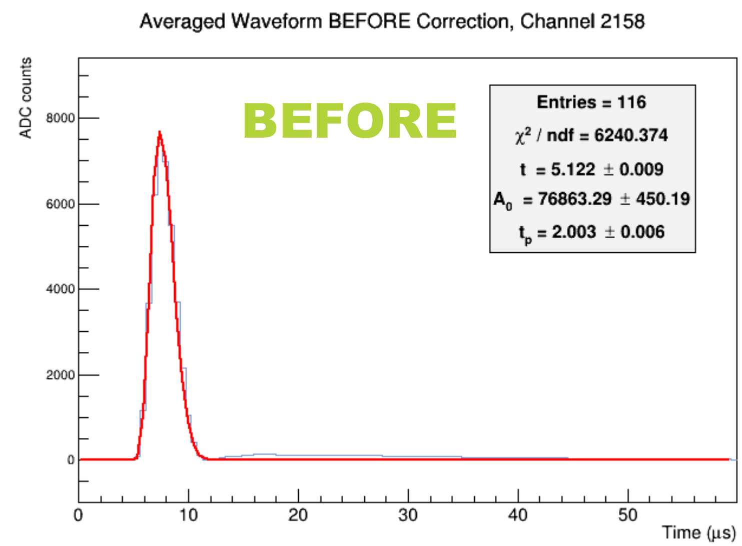
1. Run the full fitter on our dataset.
2. Extract fit parameters.
3. **Run the waveform correction (Wire-Cell).**
4. Fit corrected waveforms with Ideal Electronics Response Function.
5. Retrieve Amplitude and Shaping Time.
6. Convert Amplitude to Gain.

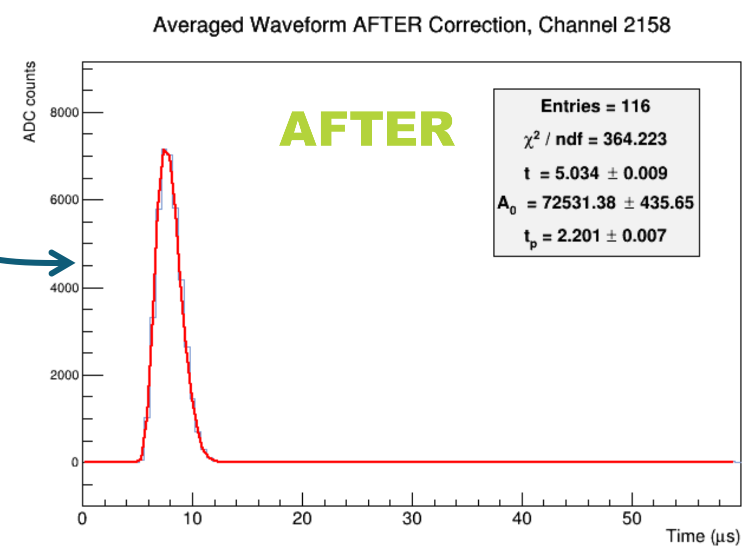




Waveform Correction

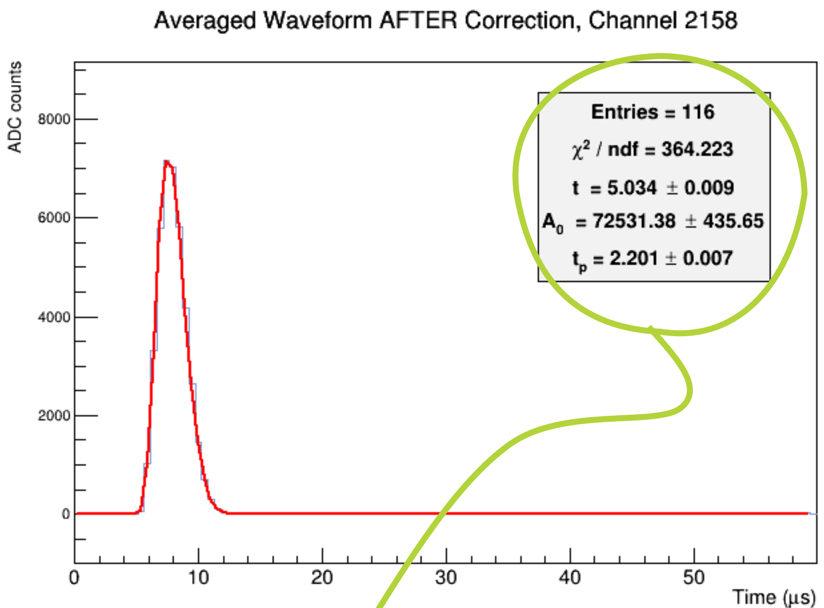
1. Run the full fitter on our dataset.
2. Extract fit parameters.
3. Run the waveform correction (Wire-Cell).
4. **Fit corrected waveforms with Ideal Electronics Response function.**
5. Retrieve Amplitude and Shaping Time.
6. Convert Amplitude to Gain.





Waveform Correction

1. Run the full fitter on our dataset.
2. Extract fit parameters.
3. Run the waveform correction (Wire-Cell).
4. Fit corrected waveforms with Ideal Electronics Response Function.
5. **Retrieve Amplitude and Shaping Time.**
6. Convert Amplitude to Gain.



A_0 = 72531.38 ± 435.65 ADCs
t_p = 2.201 ± 0.007 µs

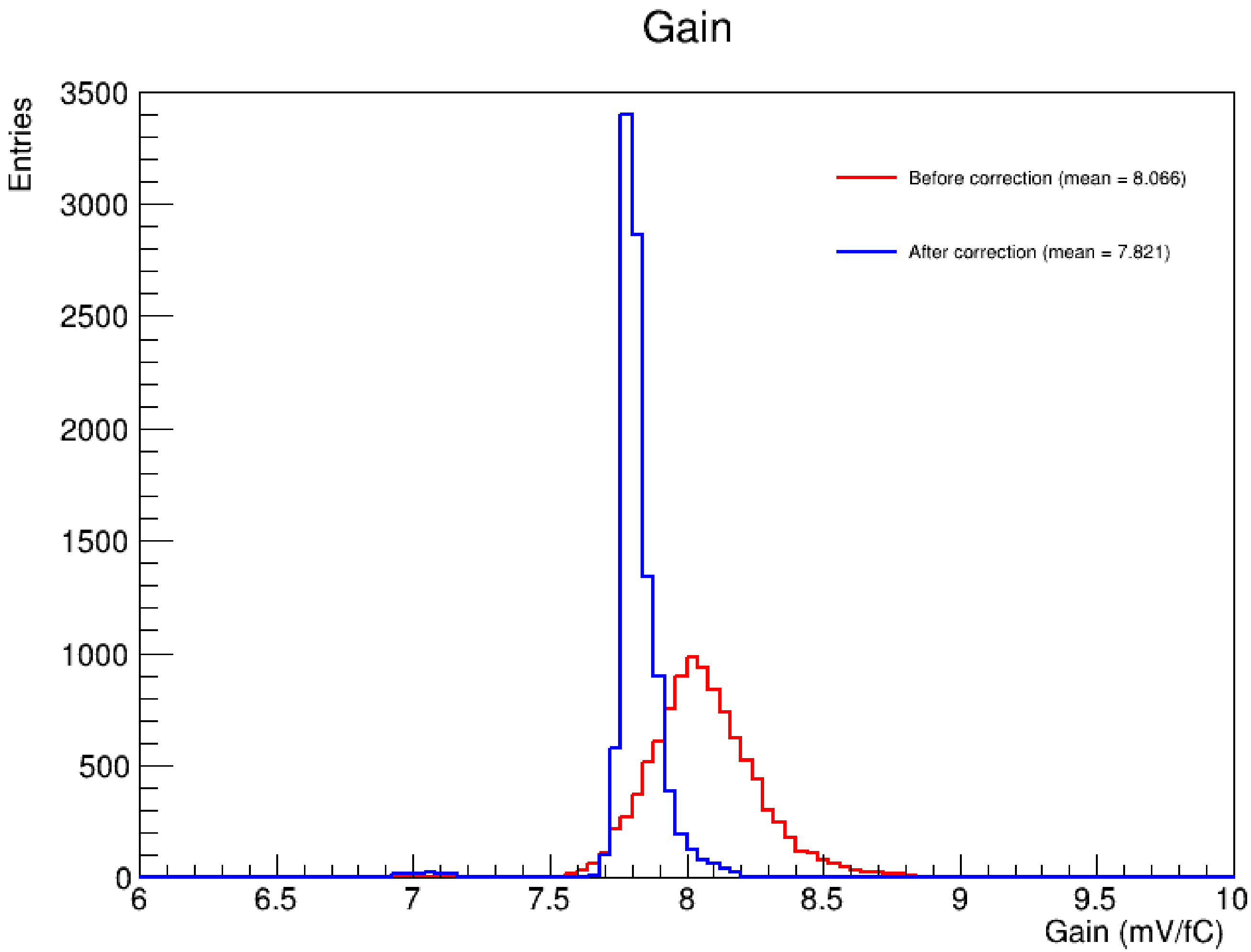
Waveform Correction

1. Run the full fitter on our dataset.
2. Extract fit parameters.
3. Run the waveform correction (Wire-Cell).
4. Fit corrected waveforms with Ideal Electronics Response Function.
5. Retrieve Amplitude and Shaping Time.
6. **Convert Amplitude to Gain.**

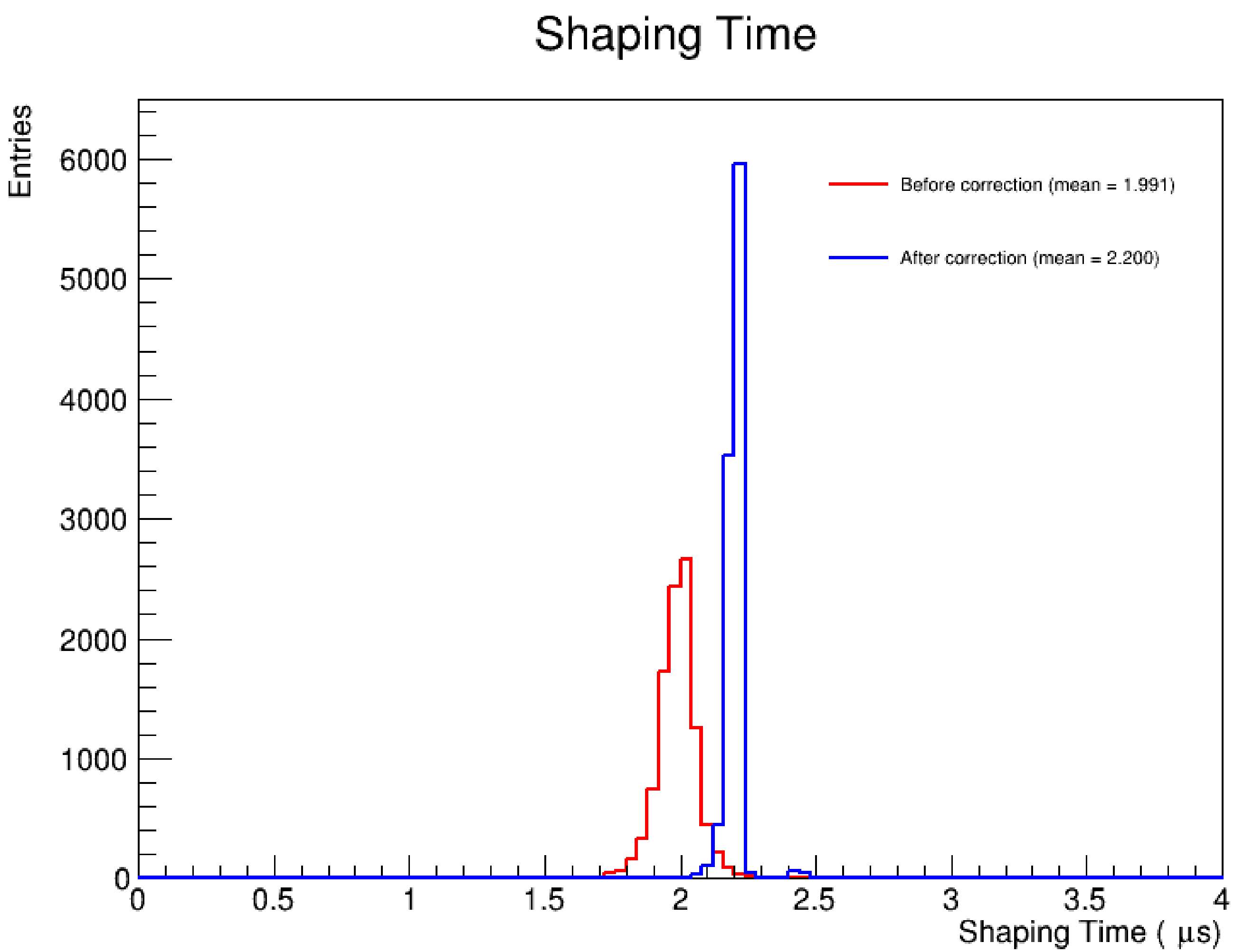
$$\mathbf{gain} = \frac{\mathbf{1400\ mV} * \mathbf{A_0}}{\mathbf{79.5315\ fC} * \mathbf{16384} * \mathbf{10}}$$

- **1400 mV is the maximum voltage in our voltage range.**
- **79.5315 fC is the injected charge**
    - **0.185 pF: test capacitance**
    - **14.33 mV/bit : DAC-to-voltage conversion factor for our LArASIC gain setting.**
    - **30 DAC: DAC setting.**
- **16384 : 14-bit ADC resolution (2^{14})**

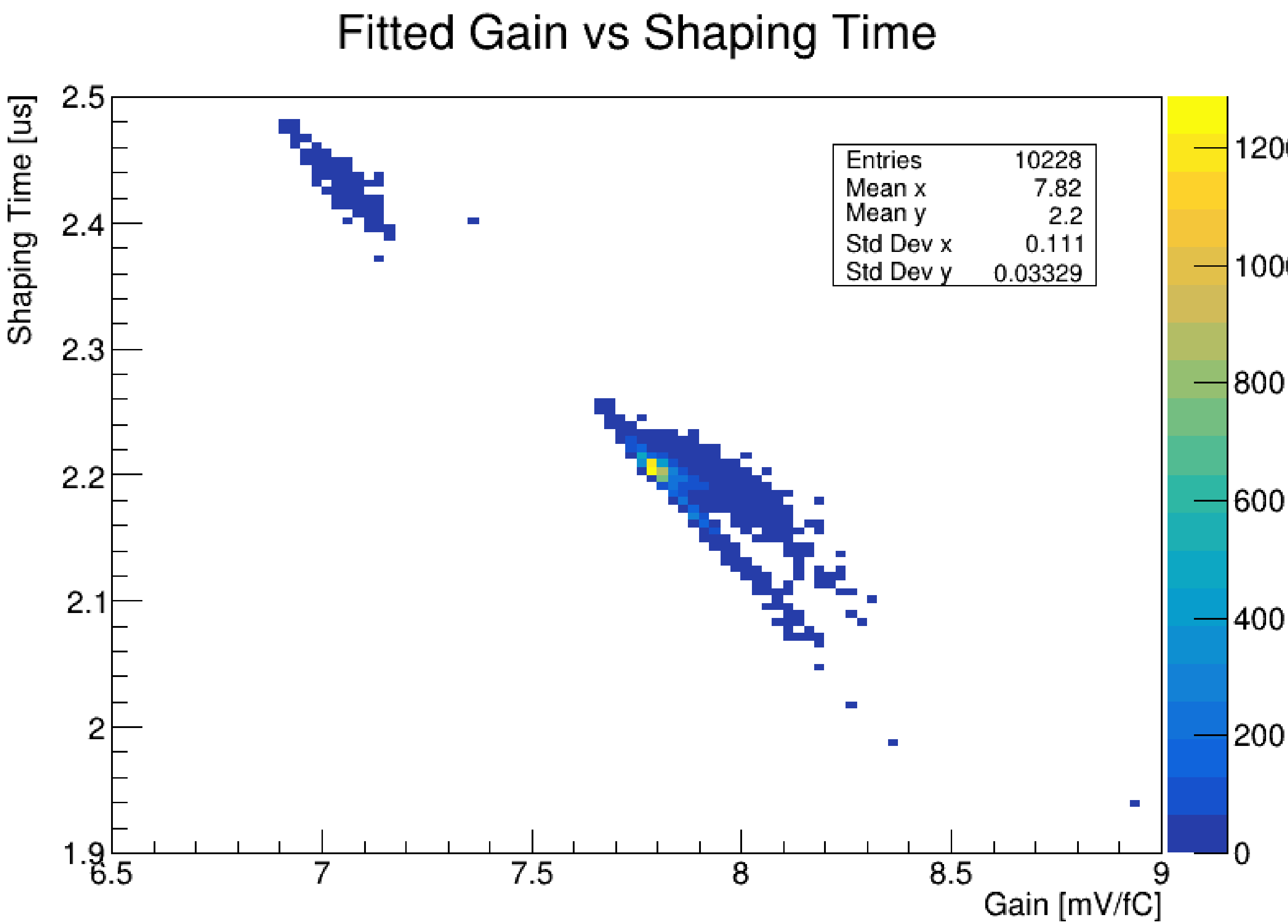
First Results



First Results



First Results

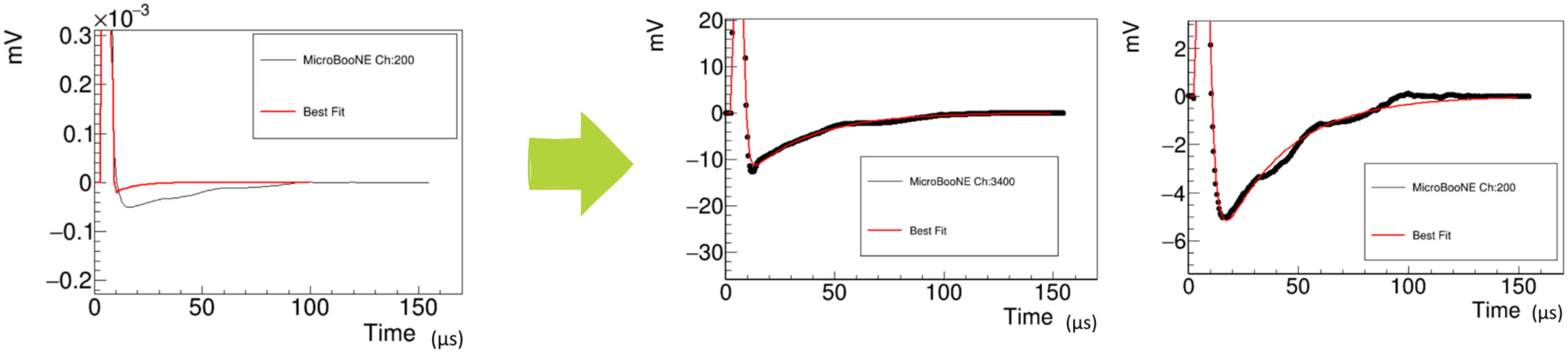


What's Next?

1. **Fix any remaining problems.**
2. **Gather all the components of the analysis and place them in a single place for the public to access.**
3. **Build a toolbox for CE performance analysis and troubleshooting.**

Backup Slides

Imperfect Pole-zero Cancelation



- From Prashansa Mukin (BNL ASIC designer):

$$T(s) = \frac{(s+k_3)(s+k_5)}{(s+k_4)(s+k_6)} \times \frac{A}{(s+p_0)(p_{1i}^2+(p_{1r}+s)^2)(p_{2i}^2+(p_{2r}+s)^2)}$$

Gain (A); Frequency (s); $\propto 1/t_p$

- Adding a canceled pole can introduce a tail.

- Mismatch between (k_3, k_4) and (k_5,k_6) is the **imperfect pole-zero cancellation**.
- When $k_3=k_4$ and $k_5=k_6$, the transfer function recovers its **ideal function** form.
- When $k_3 > k_4$ we have a **positive tail**.
- When $k_5 > k_6$ we have a **negative tail**.
- **Length** of tail is controlled by k_4.

Waveform Correction

1. Run the full fitter on our dataset.
2. Extract fit parameters.
3. **Run the waveform correction.**
4. Fit corrected waveforms with Ideal Electronics Response Function.
5. Retrieve Amplitude and Shaping Time.
6. Convert Amplitude to Gain.

Electronics Response in Waveform Data

$$M_i(t_0) = \int_{-\infty}^{\infty} R_i(t - t_0) \cdot I(t) \cdot dt$$

Digitized Waveform — Channel "i" Elec. Response — Induced Current

Frequency Domain

$$M_i(\omega) = R_i(\omega) \cdot I(\omega).$$

Electronics Response Correction

$$M_i^{Corr}(\omega) = M_i(\omega) \cdot \frac{R_{nominal}(\omega)}{R_i(\omega)}$$

Channel "i" measured response FFT

https://arxiv.org/pdf/1804.02583