Hit Finders for Pileup Recognition in 2x2 Data

Tom Sonius

Requirements

- 97% efficiency for detecting pileup greater than 100 ns
- T0 resolution < 10 ns
- LCM noise threshold allowing 10 PE signal
- ACL noise threshold allowing 20 PE signal

Requirements

- 97% efficiency for detecting pileup greater than 100 ns
- T0 resolution < 10 ns
- LCM noise threshold allowing 10 PE signal
- ACL noise threshold allowing 20 PE signal

Outline

- Short overview of the hit finders and their parameters
- Data preprocessing + visualizations
- Fprompt parameter and selecting of clean events
- Waveform overlaying
- Results
- Conclusion

Important point

Peak: Cannot distinguish between peaks belonging to the same interaction or a new interaction

Hit Finder: Extended Single-Peak Finder

- Uses scipy’s find_peaks function (Note: params are stated in order in which the function evaluates it)
    - Noise threshold: Sets a floor at 3σ of the noise
    - Height: Threshold of 3σ of the noise
    - Distance: 6 ticks (96 ns)
    - Prominence: Waveform value with a 0.1% false positive rate (assumes Gaussian noise)

- SNR: Peaks must at least be 3x the noise level
- Rising edge: Strictly increasing

Hit Finder: ToT into find_peaks

- Groups consecutive bins (hits) that exceed a noise adjusted threshold (Time-Over-Threshold)
  - Noise threshold: Sets a floor at 3σ of the noise
  - Rolling average: Over 3 bins
  - Start time: First bin above either threshold
  - End time: First bin below threshold after start time
  - Save largest hit within a ToT region
- Applies Scipy’s find_peaks on hits with height > dynamic threshold
  - Dynamic Threshold: Rolling average + sqrt(rolling average)
  - Distance: Ignores peaks within 2 ticks (32 ns) of each other

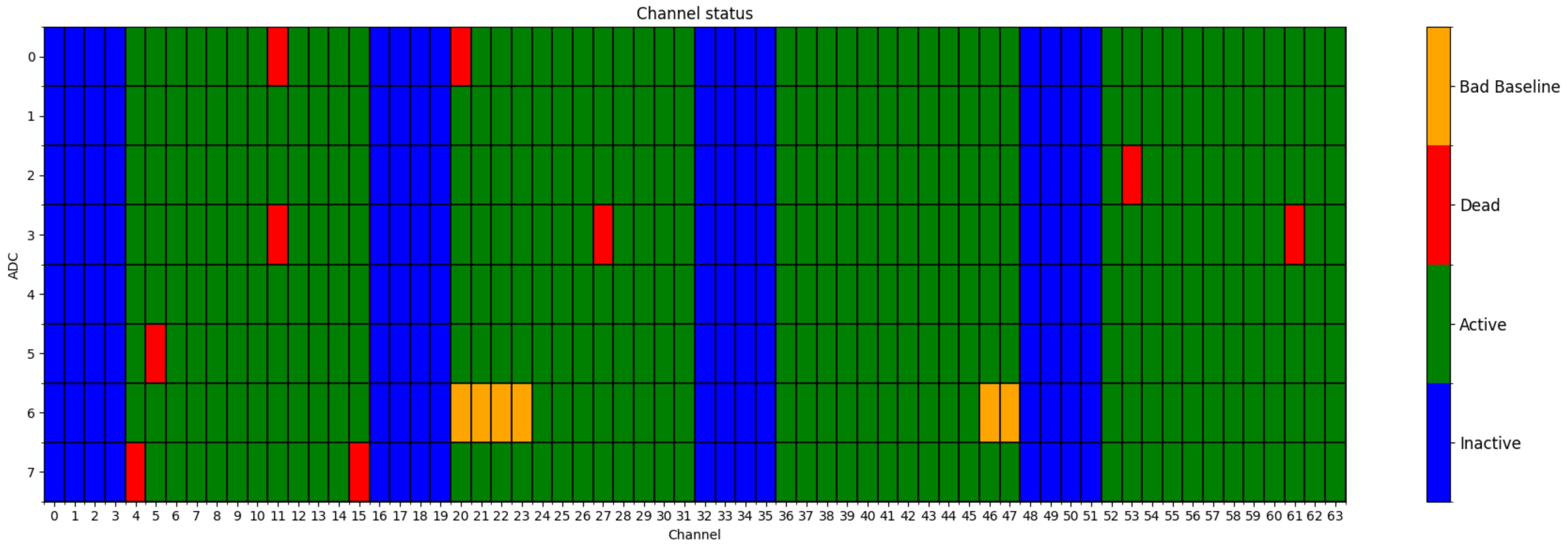
Hit Finder: ToT into Rising Edges

- Groups consecutive bins (hits) that exceed a noise adjusted threshold (Time-Over-Threshold)
  - Noise threshold: Sets a floor at 3σ of the noise
  - Rolling average: Over 3 bins
  - Start time: First bin above either threshold
  - End time: First bin below threshold after start time
  - Save largest hit within a ToT region
- Find the rising edges
  - Rising edge: Waveform exceeds dynamic threshold and noise threshold
  - Dynamic Threshold: rolling average + sqrt(rolling average)
  - Distance: Ignores peaks within 2 ticks (32 ns) of each other

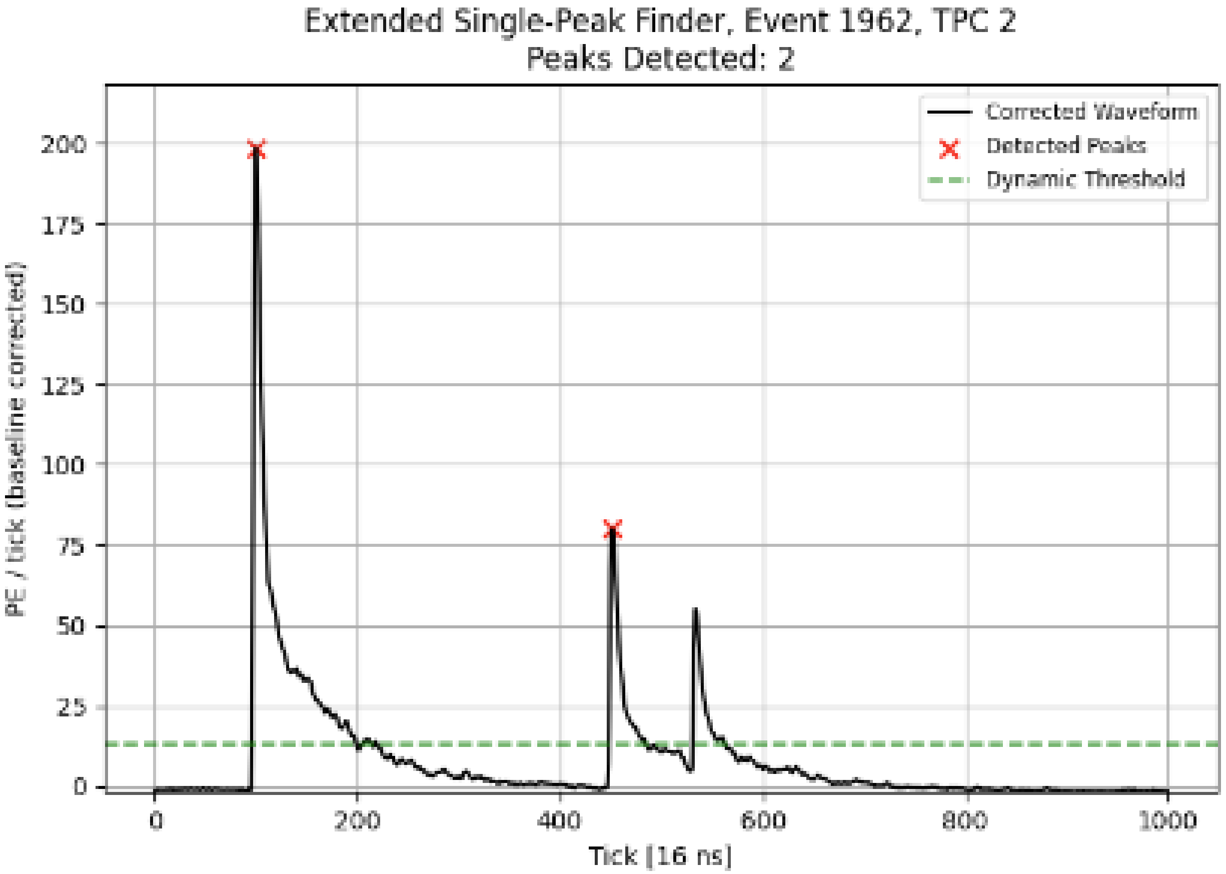
Data Pre-Processing

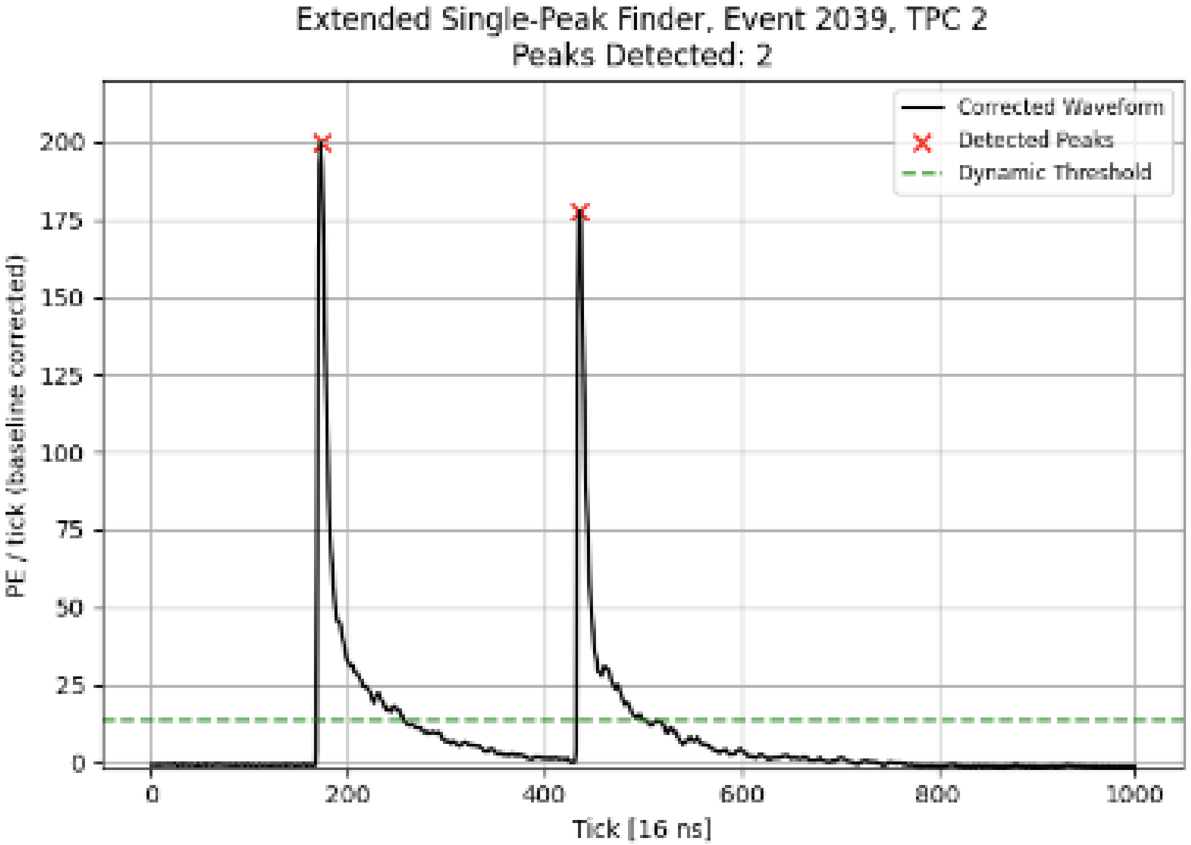
mpd_run_hvramp_rctl_104_p130.FLOW.hdf5

- Active channels only
- Waveforms are calibrated on a channel basis using the calibration values from July 2x2 run
- Waveforms are summed per TPC
- Waveforms are baseline corrected

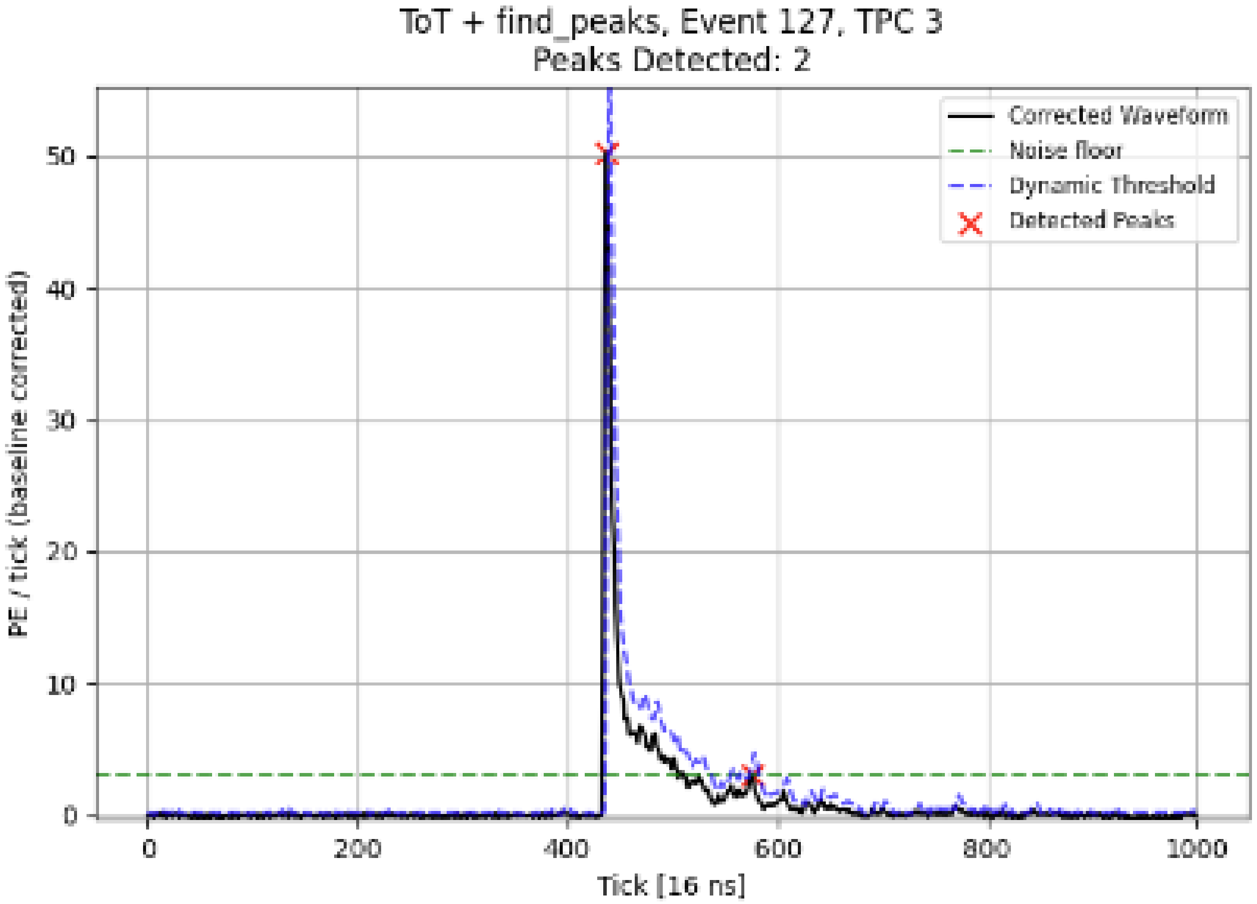


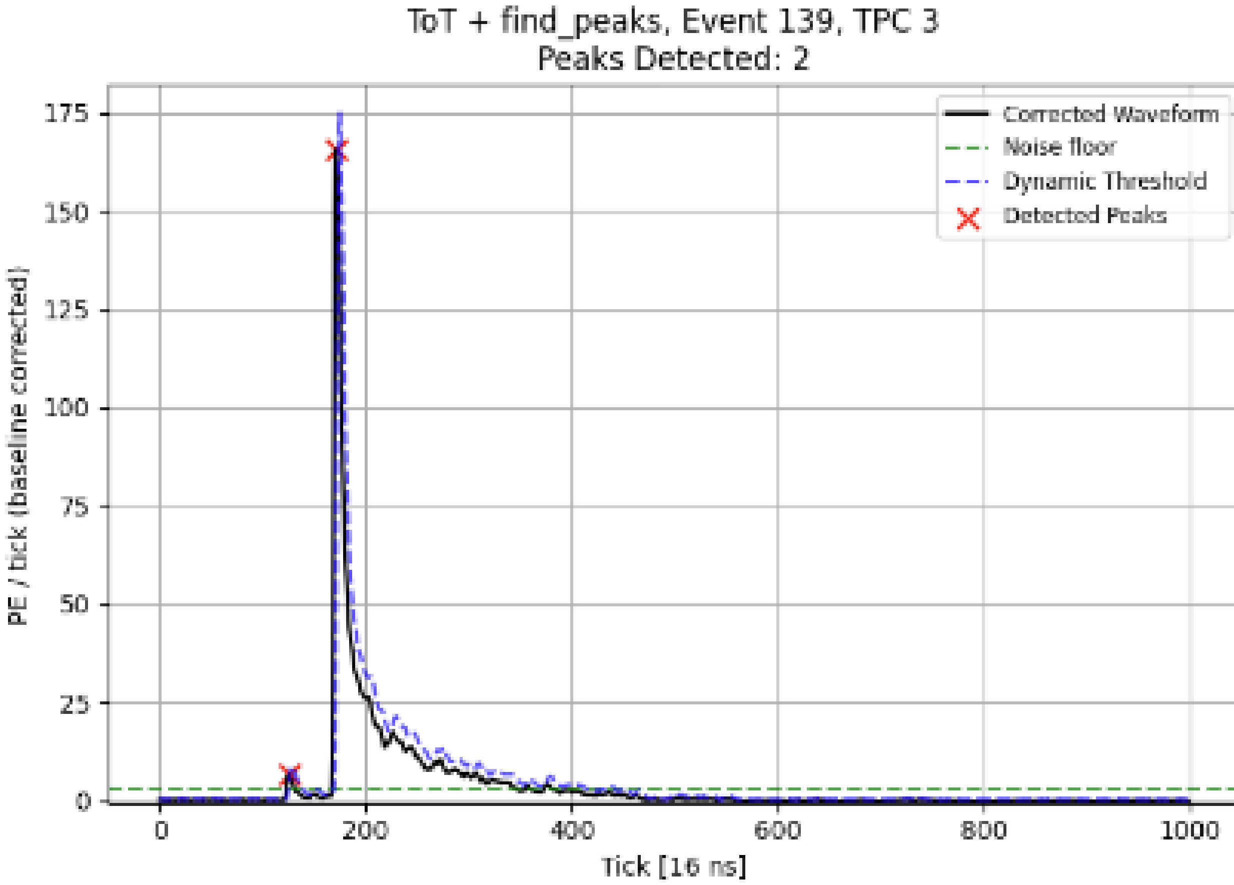
Extended Single-Peak Finder: Visualization



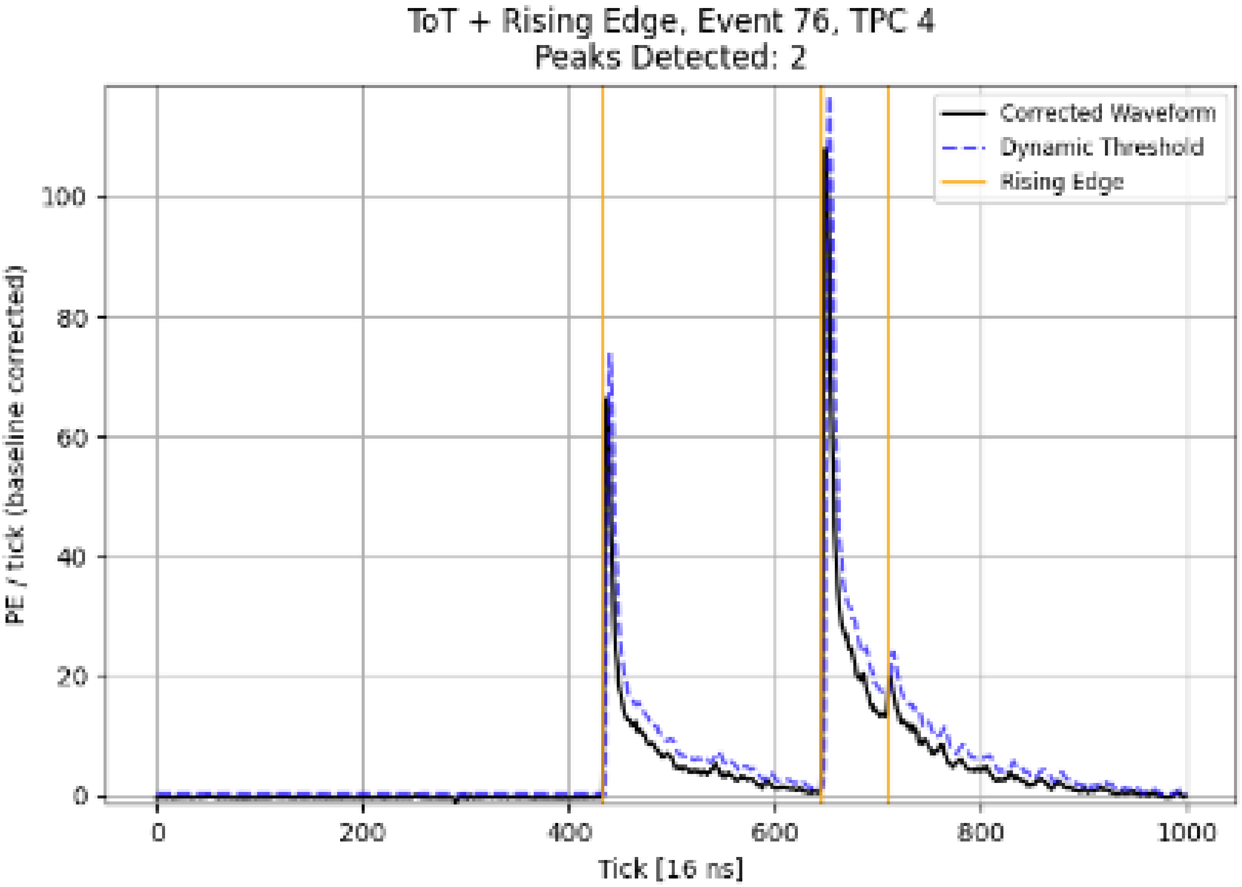


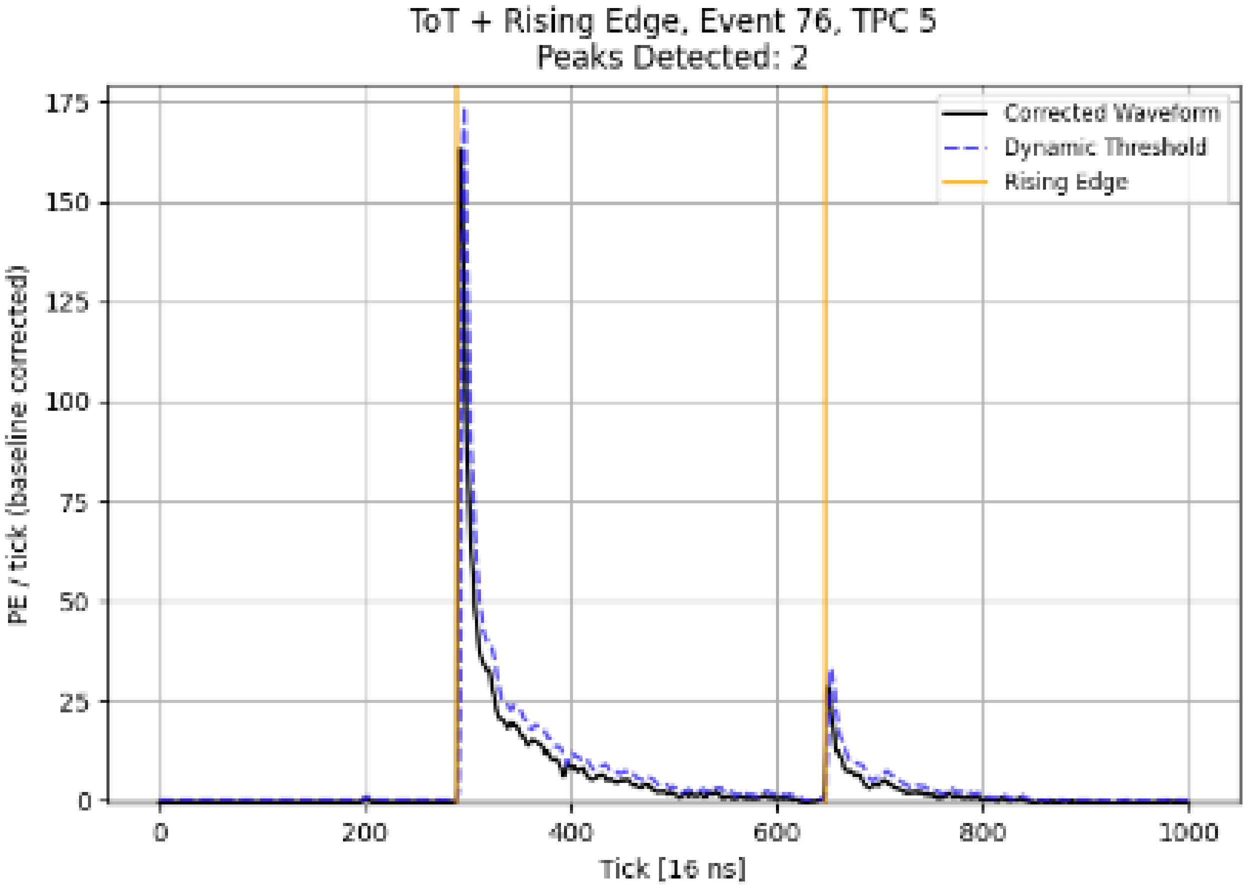
ToT + find_peaks : Visualization



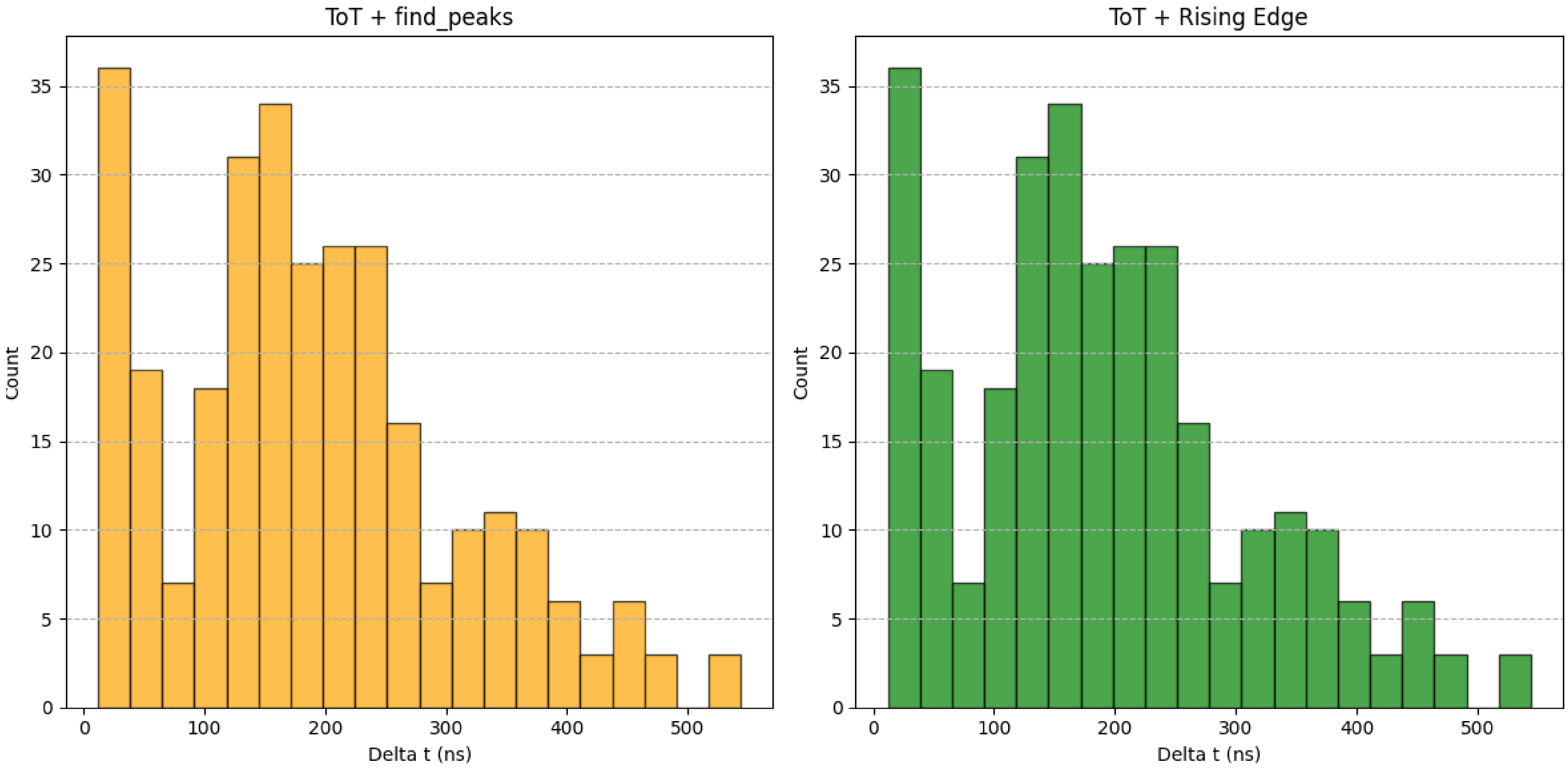


ToT + Rising Edge : Visualization


ToT + Rising Edge, Event 76, TPC 4
Peaks Detected: 2


ToT + Rising Edge, Event 76, TPC 5
Peaks Detected: 2

Data delta t histograms

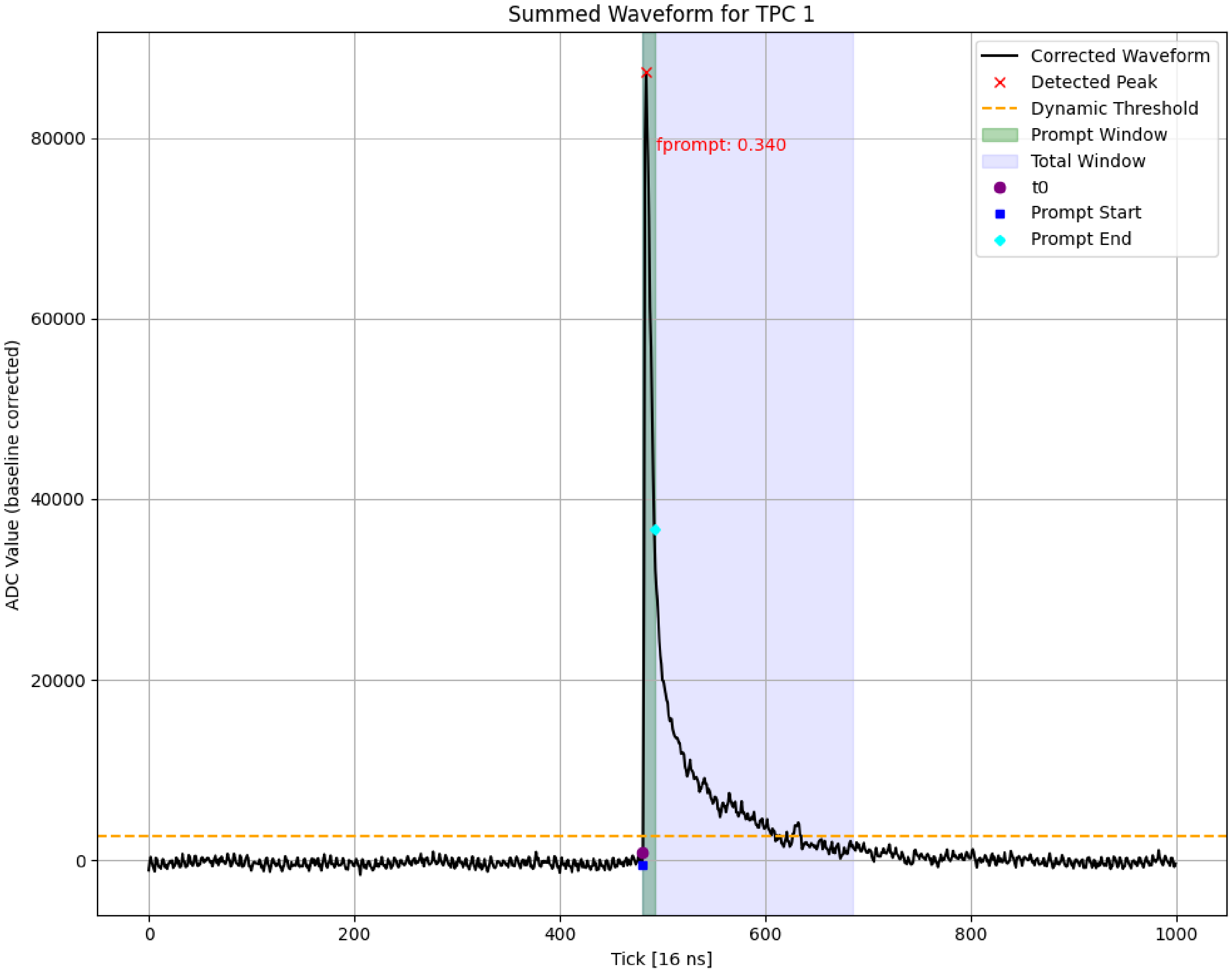


fprompt Parameter

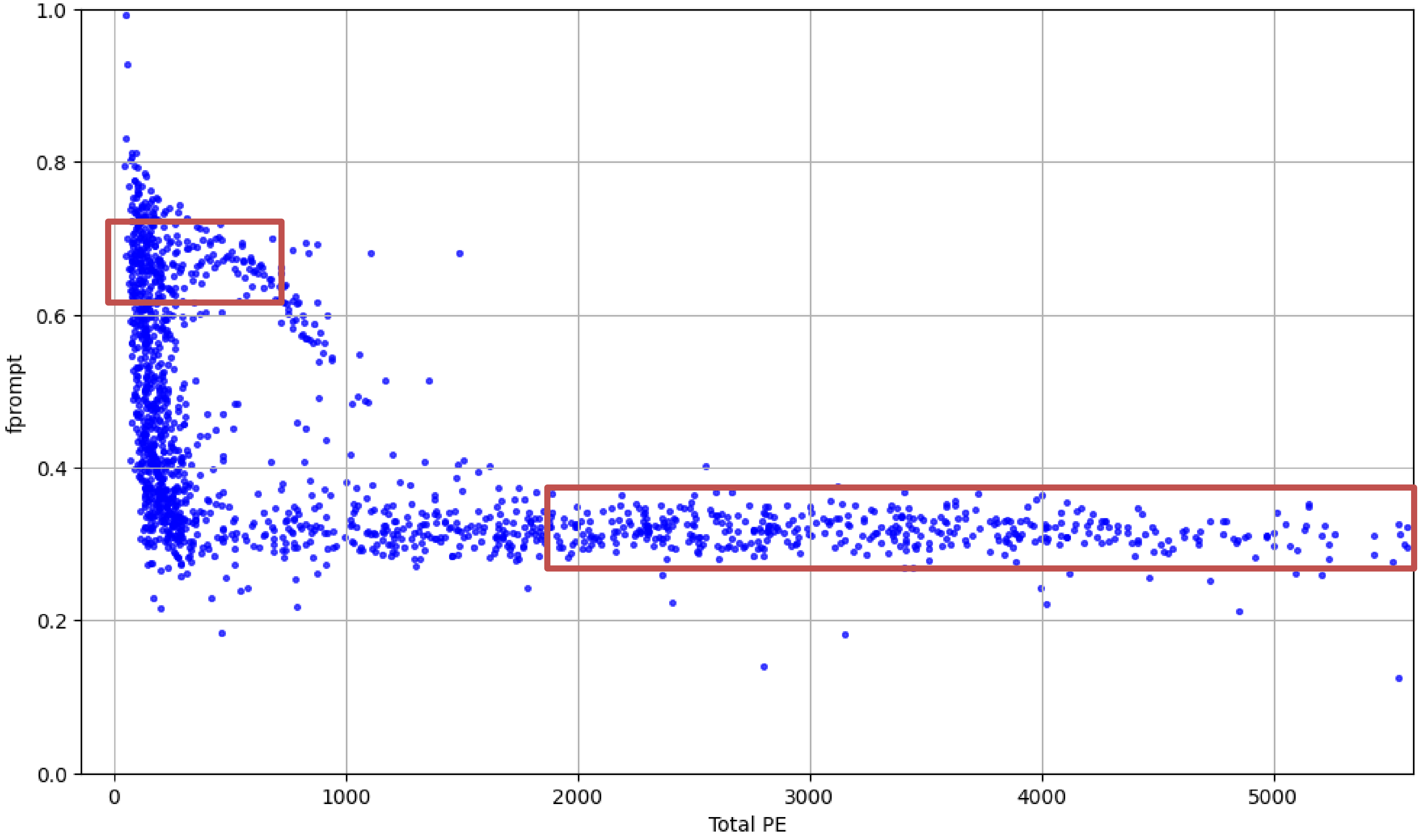
$$f_{\mathrm{prompt}} = \frac{\int_{t_0}^{t_{cut}} I(t)\,dt}{\int_{t_0}^{t_{\mathrm{total}}} I(t)\,dt}$$

- ***I (t)*: Scintillation intensity**
- ***t_0*: Start time of the signal.**
- ***t_{cut}*: Time defining the prompt window.**
- ***t_{total}*: Total integration time.**

- ***Baseline correction:***
  - Subtract median waveform value
- ***Fprompt calculation:***
  - Prompt window: 200 ns
  - Total window: 3200 ns



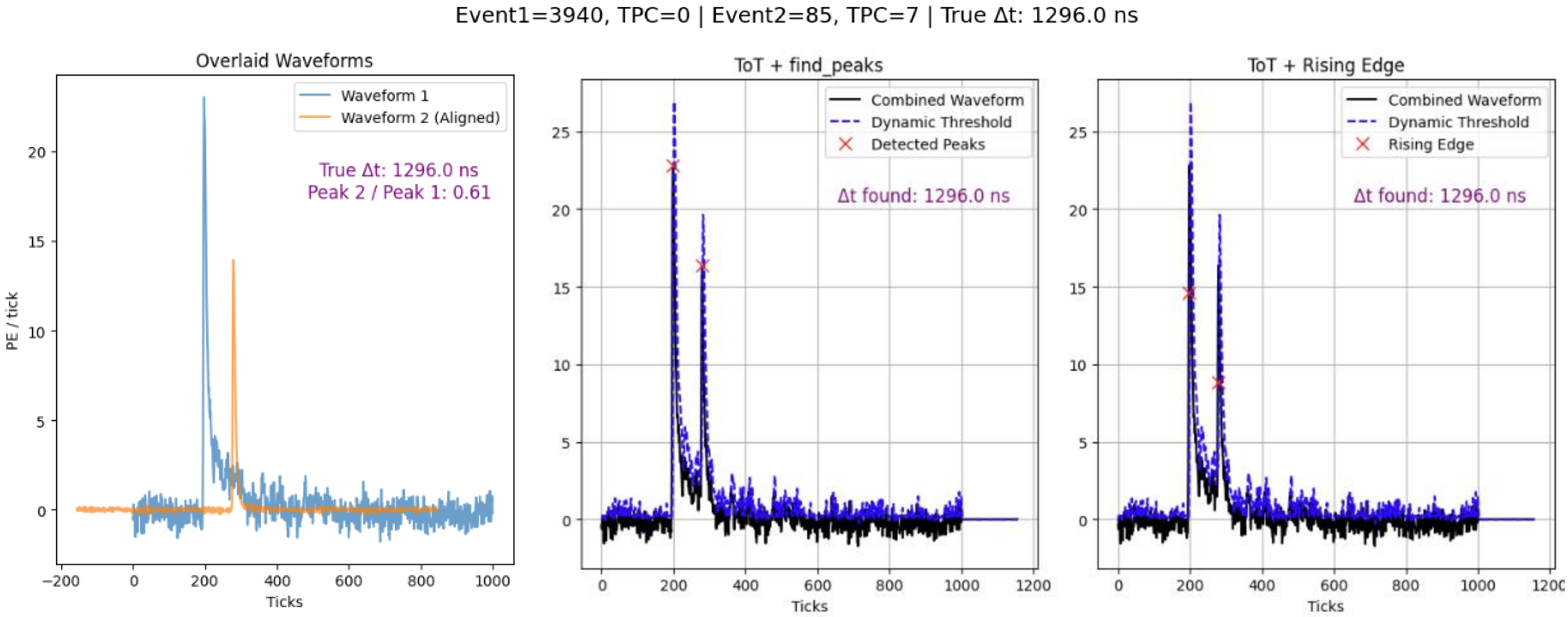
Fprompt parameter scatter plot



How Artificial Pileup is Created

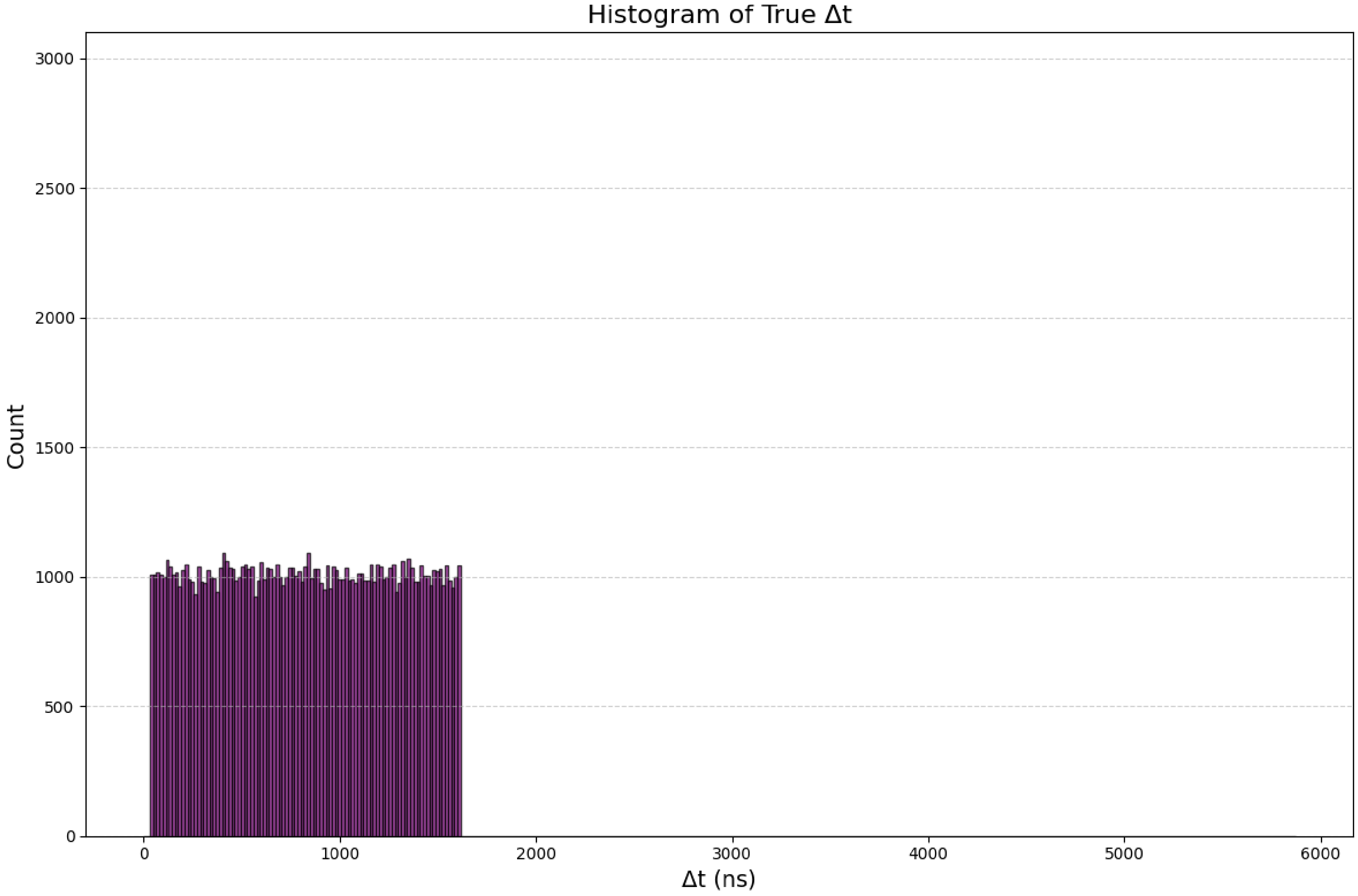
- **Select events from data**
  - Choose two events with single peaks
    - One from region 1 (waveform 1) and the other from region 2 (waveform 2)
- **Extract peak information**
  - Find the index of the peak in waveform 1 and waveform 2
  - Calculate the ratio of peak 2 to peak 1
- Generate a random offset between the peaks (2 ticks – maximum_offset)
- **Calculate the shift**
  - See what shift is needed to align waveform 2 to 1 such that its peak is offset by the random amount generated
- **Overlay waveforms**
  - Generate an empty array of the length of waveform 2 + its shift
  - Add waveform 1 to the array starting at 0 ticks
  - Add waveform 2 to the array starting at the shift position
- **True time difference**
  - The “true” delta t between the two peaks is then the randomly generated offset

Artificial Pileup Visualization

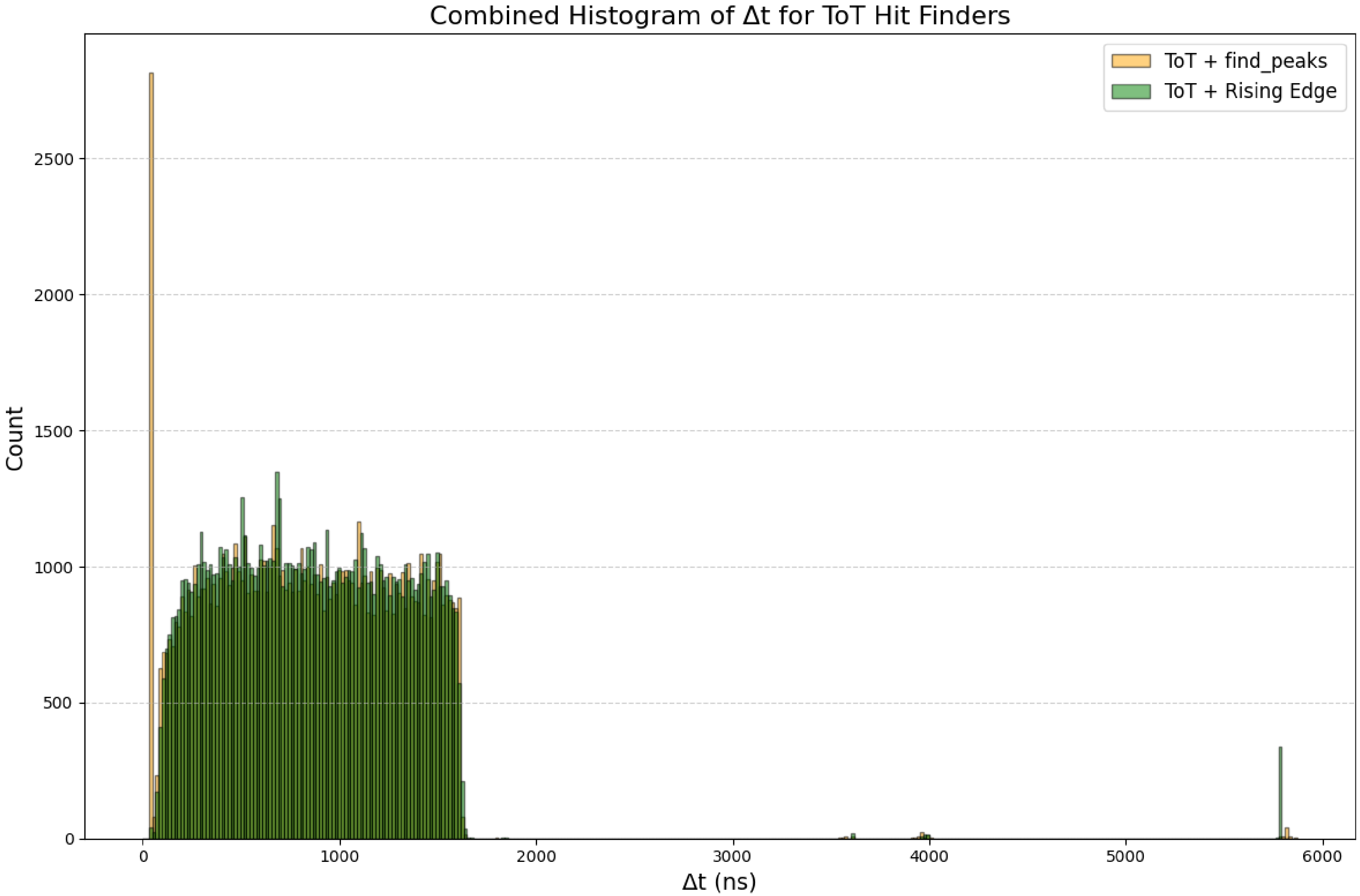


“True” delta T histogram

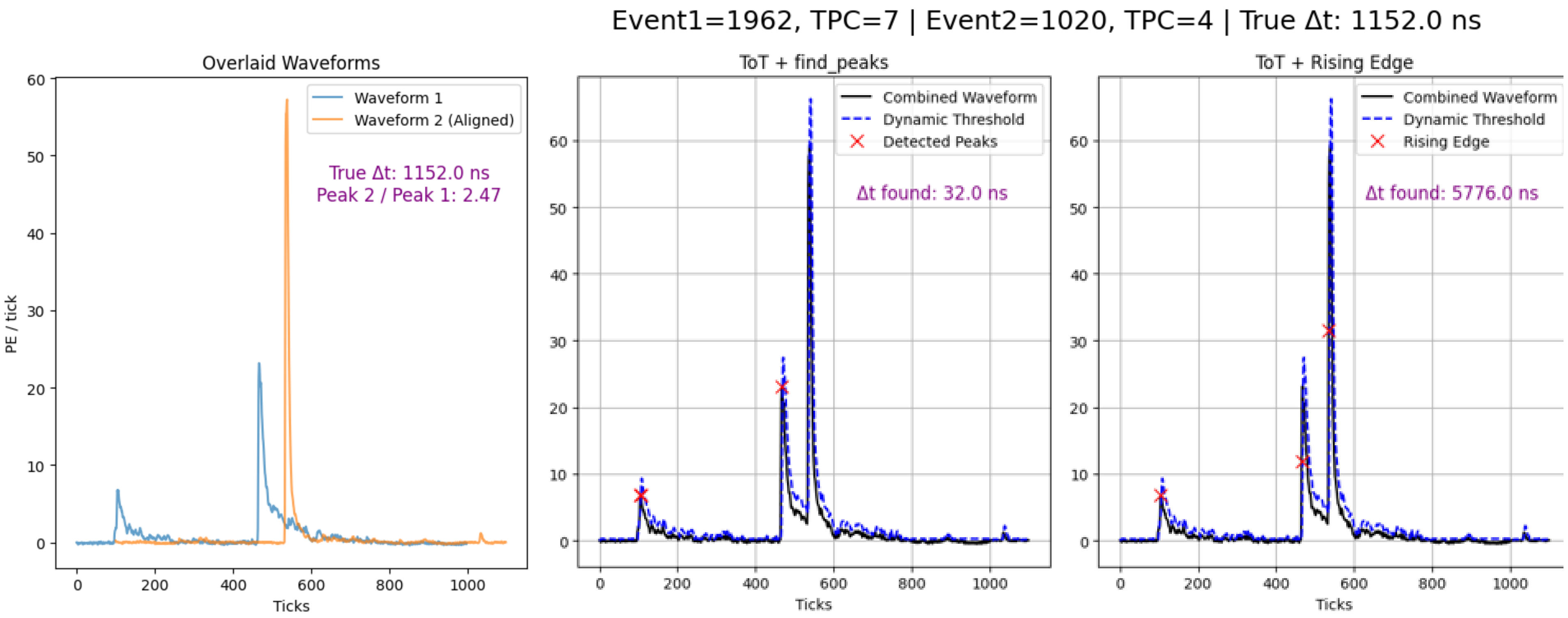
- Max offset of 1600



Found delta t histogram



Current issue: Visualization



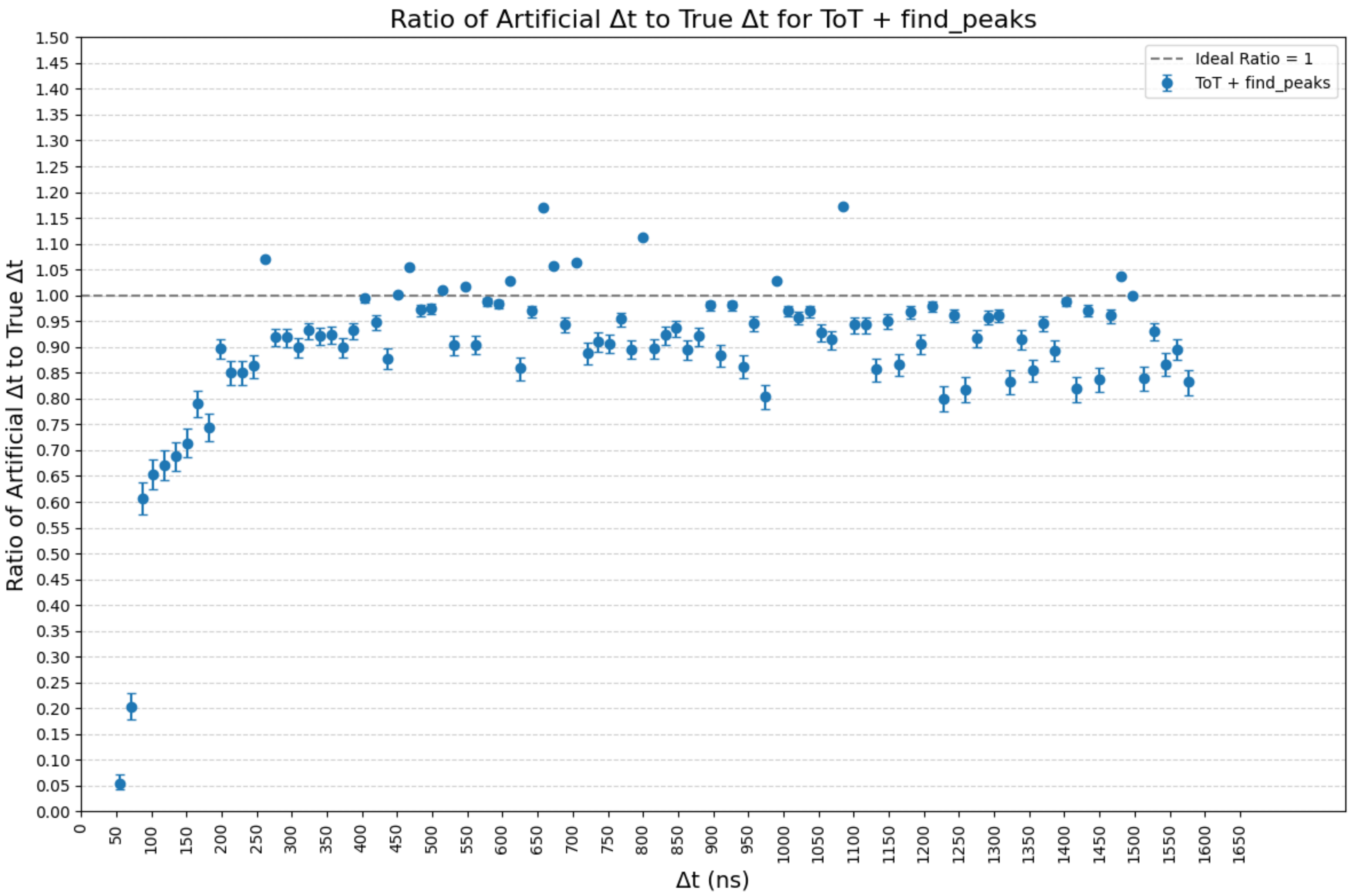
Event1=1962, TPC=7 | Event2=1020, TPC=4 | True Δt: 1152.0 ns

Overlaid Waveforms

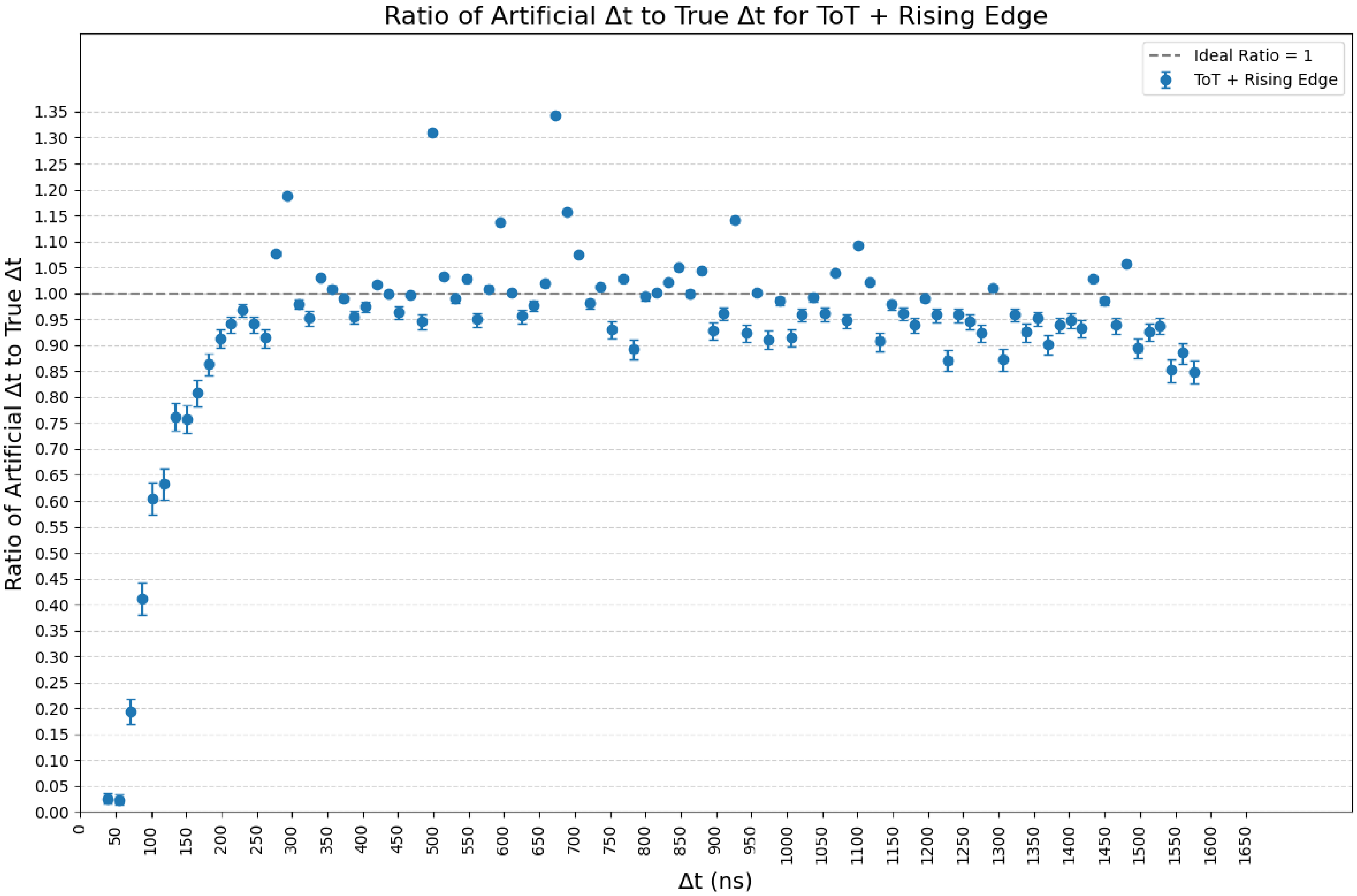
ToT + find_peaks

ToT + Rising Edge

ToT + find_peaks performance

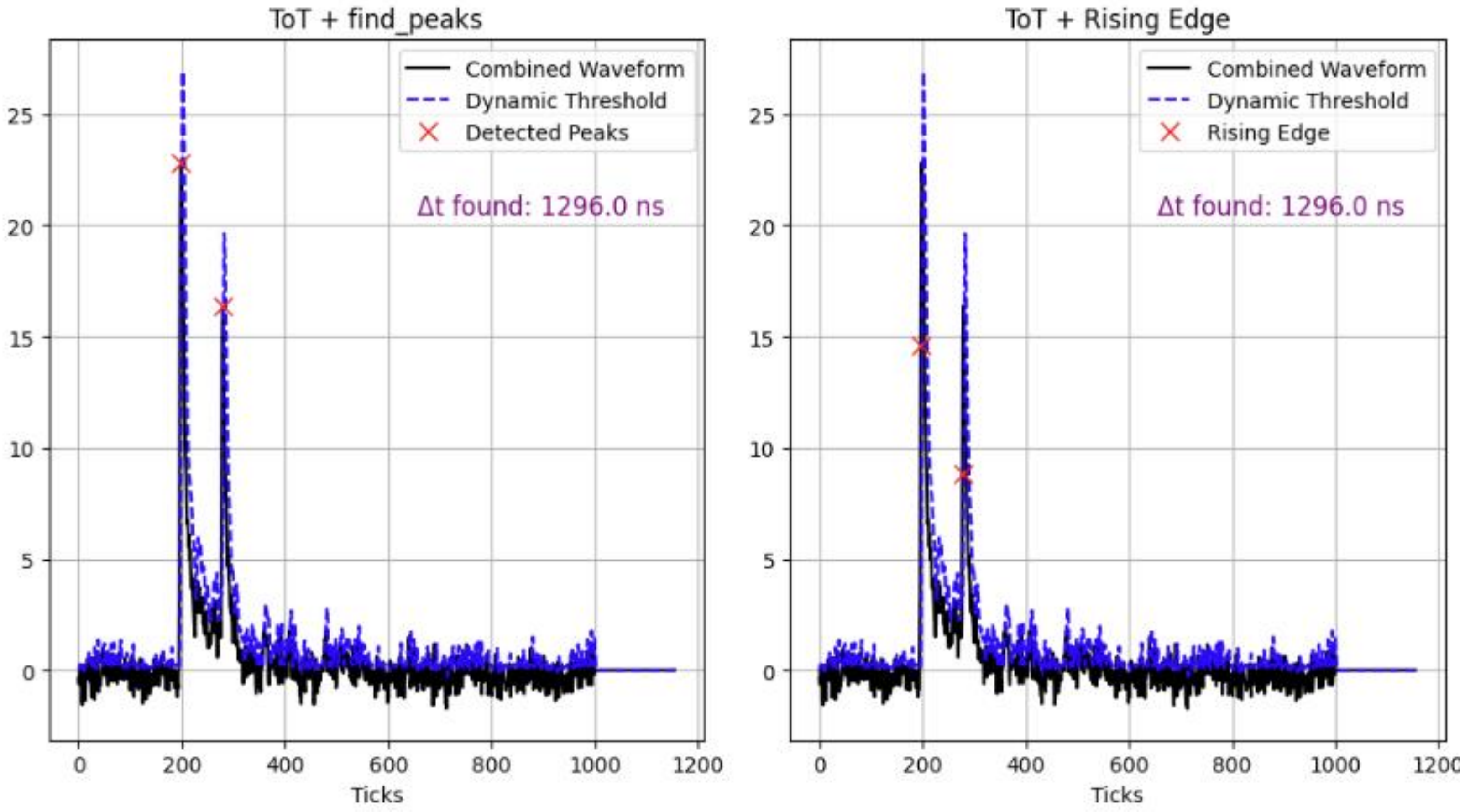


ToT + Rising Edge performance



ToT + Rising Edge performance

- Hit finders are sensitive to the ratio of the peaks
  - Great at detecting peaks (even in 100 ns) of similar size
  - Not so great if they're very different



Conclusion + Next Steps

- ***Conclusion***
  - Both ToT hit finders are able to distinguish peaks -> not yet reaching the 100 ns threshold that we want

- **Next steps**
  - Show sensitivity of hit finders to peak ratio
  - Go to MC – where true amount of interactions are known
  - Fix bugs (32 ns) and play around with parameters
  - Go to data and separate LCMs vs ACLs – determine if we can meet noise threshold requirements
  - Make steps towards determining the t0 resolution
  - More randomizations, more pileup, multiple data files

Data Pre-Processing

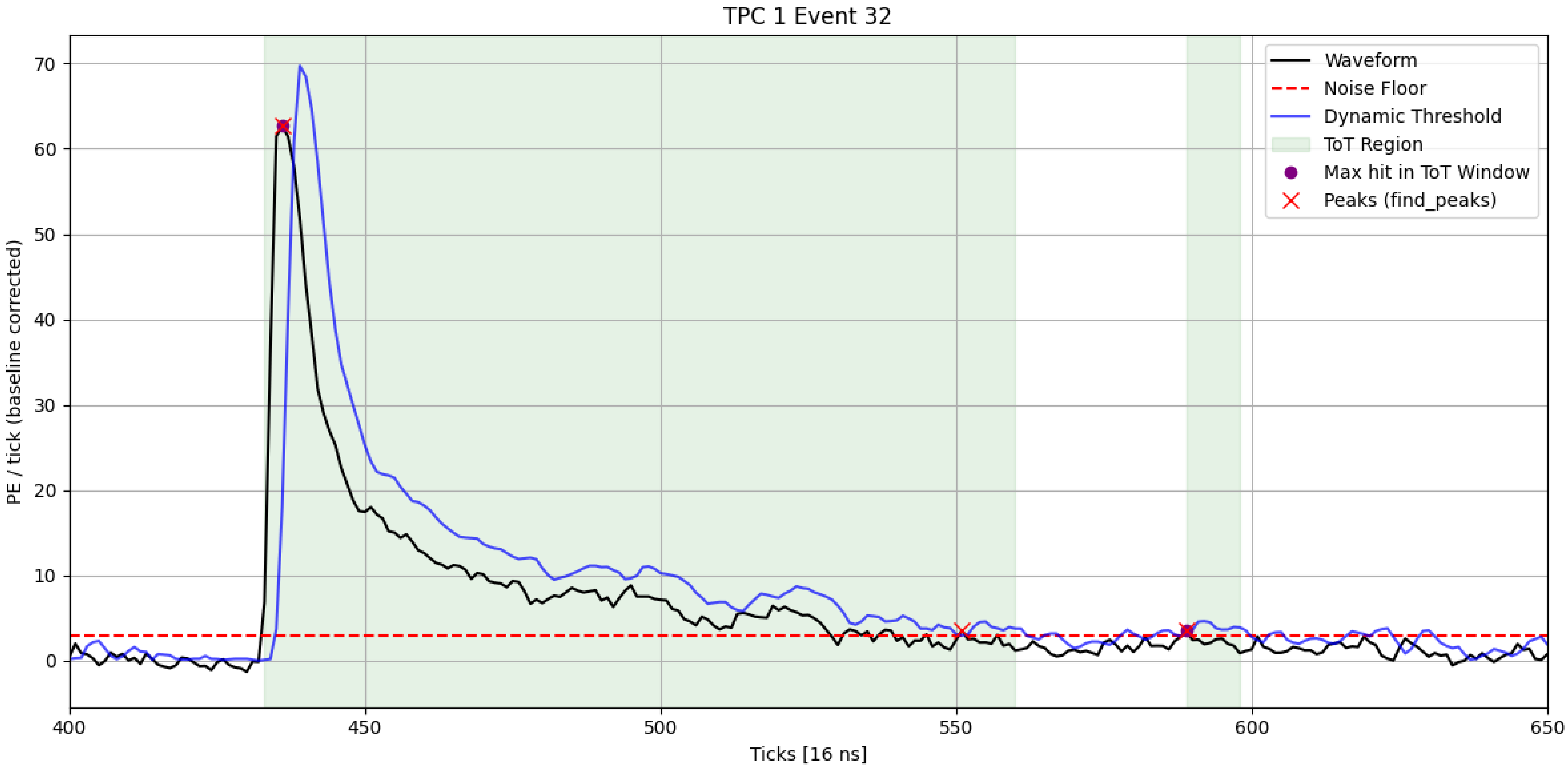
- Explain how waveforms are overlayed
- Explain how I defined pass or fail and why for accepting a delta t compared to true delta t
- Next steps in terms of determing t0 resolution
- Doing it for separated LCMs for ACLs

- Explain how I randomly select two waveforms and why I choose those specific regions
- Explain how this can be extended

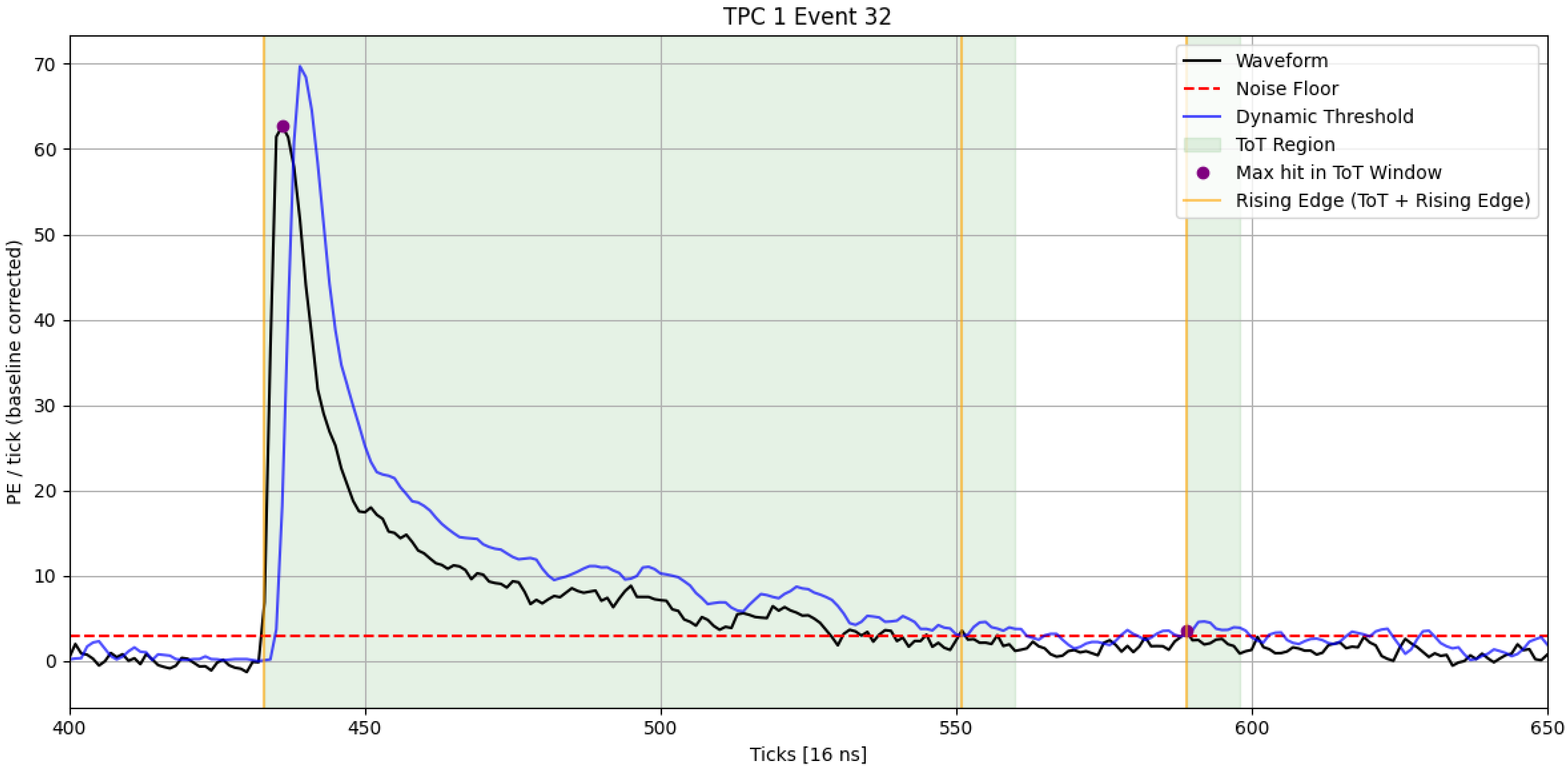
- Think about every step you do and every parameter you've set and explicitly state these and try to reason them
- Next steps – go to MC where we will know true amount of interactions
- Improve current steps -> stronger motivation behind parameter selections
- Select more than the region I do now, increase statistics
- Go to data and separate LCMs vs ACLs -> determine if noise threshold requirement can be met
- Make steps towards determining the t0 resolution
- Go to multiple data files

Backup Slides

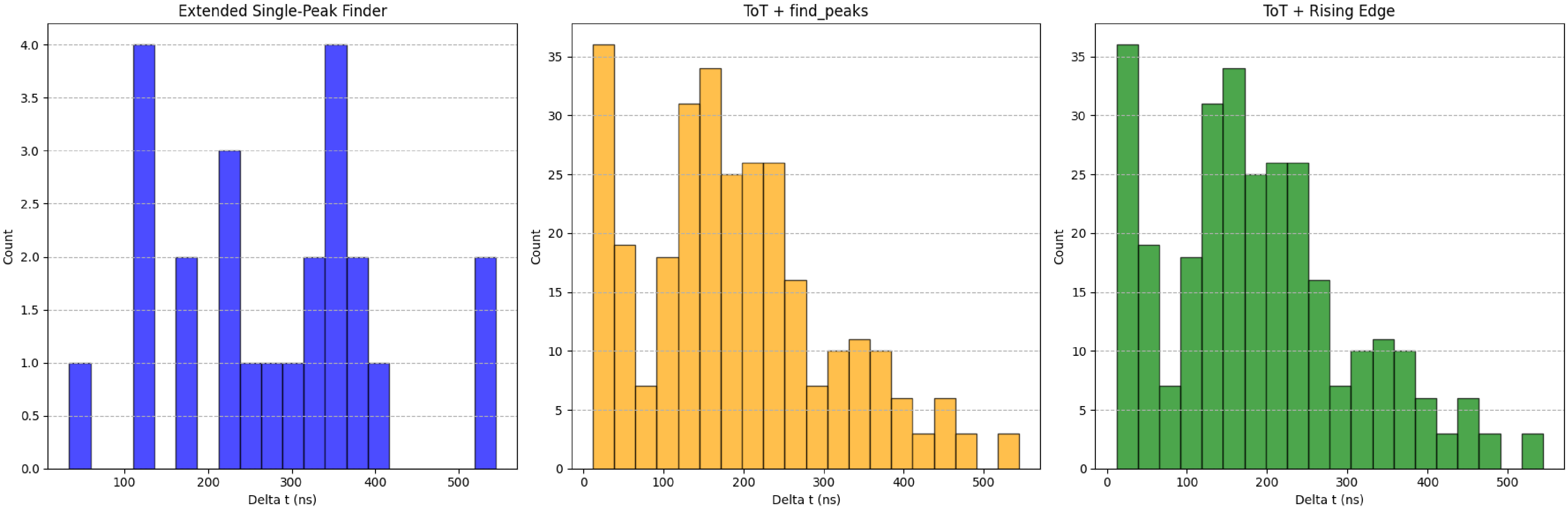
Data Pre-Processing



Data Pre-Processing



Data Pre-Processing



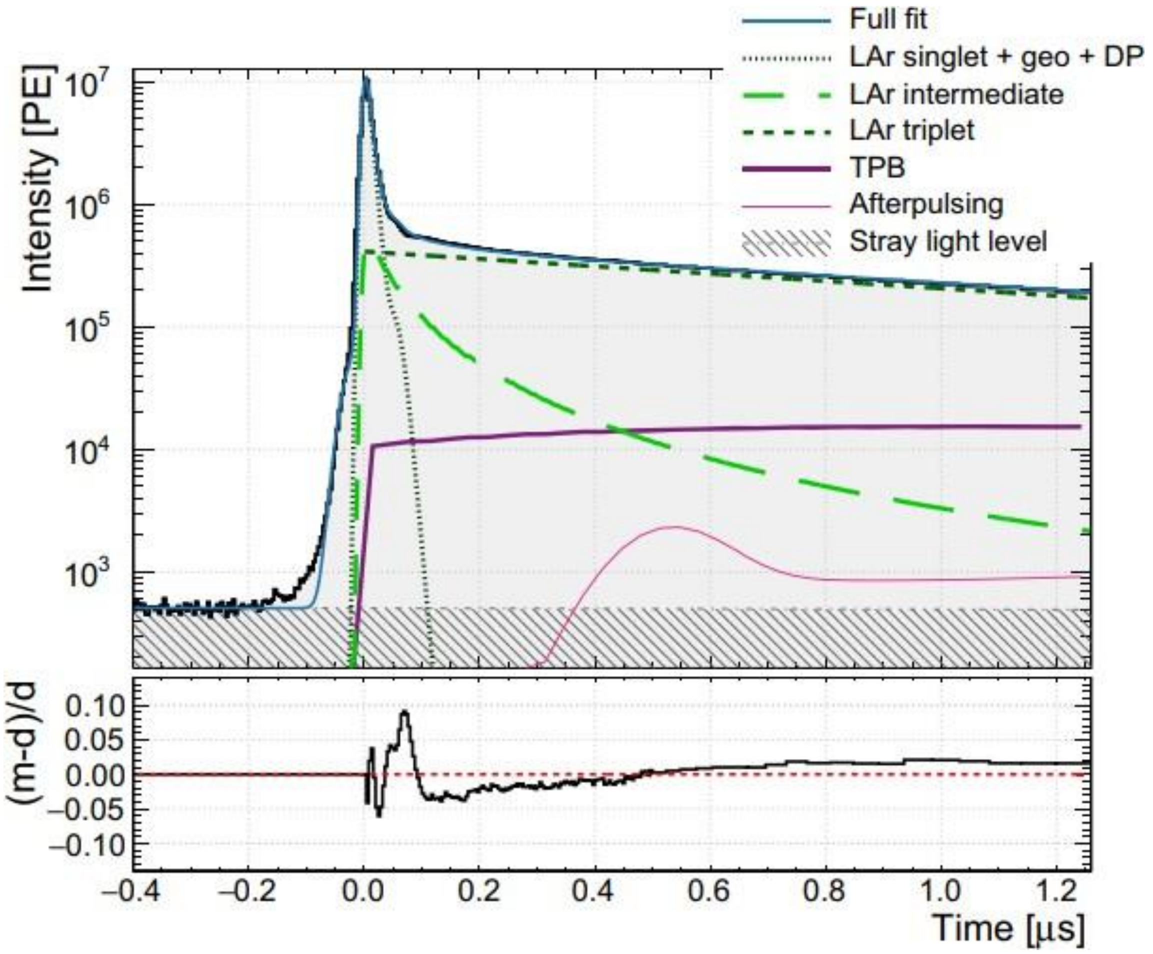
Pulse Shape Components

Experiment:

- DEAP-3600 (Ar-39)
- Pulse shape characteristics

Fits:

- geometric effect + detector response
- intermediate (later recomb.)
- TPB late emission
- Afterpulsing (residual charge effects)


Full fit
LAr singlet + geo + DP
LAr intermediate
LAr triplet
TPB
Afterpulsing
Stray light level
Intensity [PE]
10^7
10^6
10^5
10^4
10^3
(m-d)/d
Time [µs]