Hit Finding Update

Difference algorithm summary

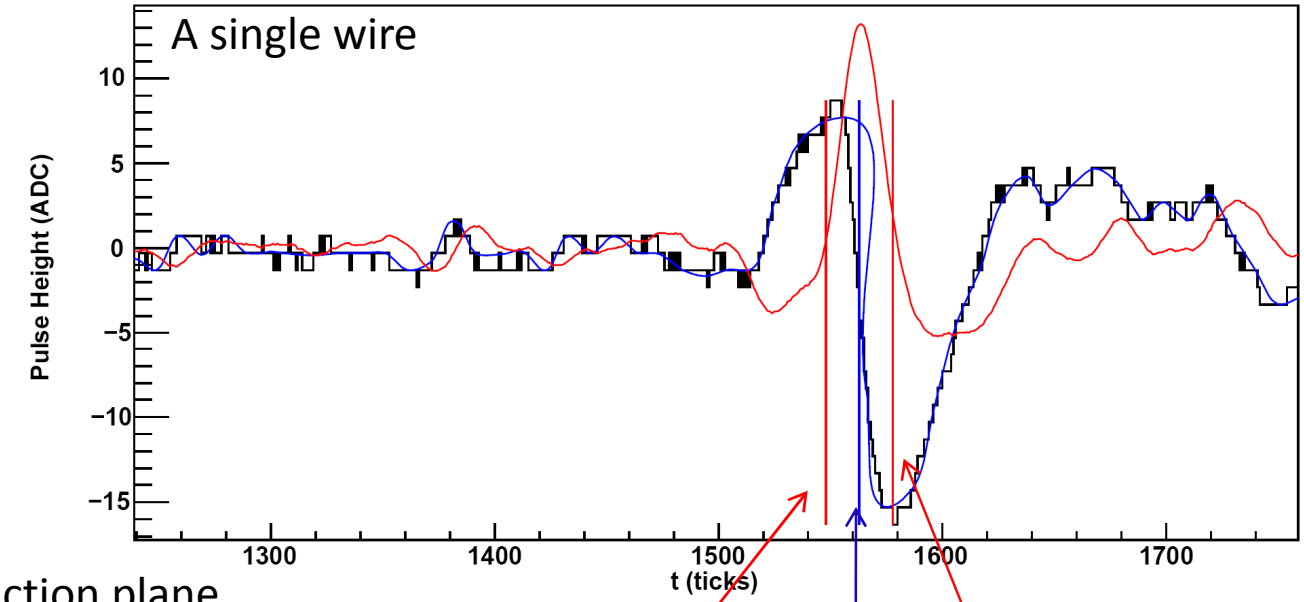
Joshua Spitz

1/20/2009

The ‘difference’ algorithm

The algorithm finds the difference between the integrated signal before and after a given time bin. The peak of the resulting difference is the central time of the hit.

1. Find the time bin *i* that gives the largest value of $Integral_i = \sum_{i-15}^{i} a_i - \sum_{i}^{i+15} a_i$
2. If Integral$_i$ is above the wire’s pedestal by *knob* sigma, count as hit



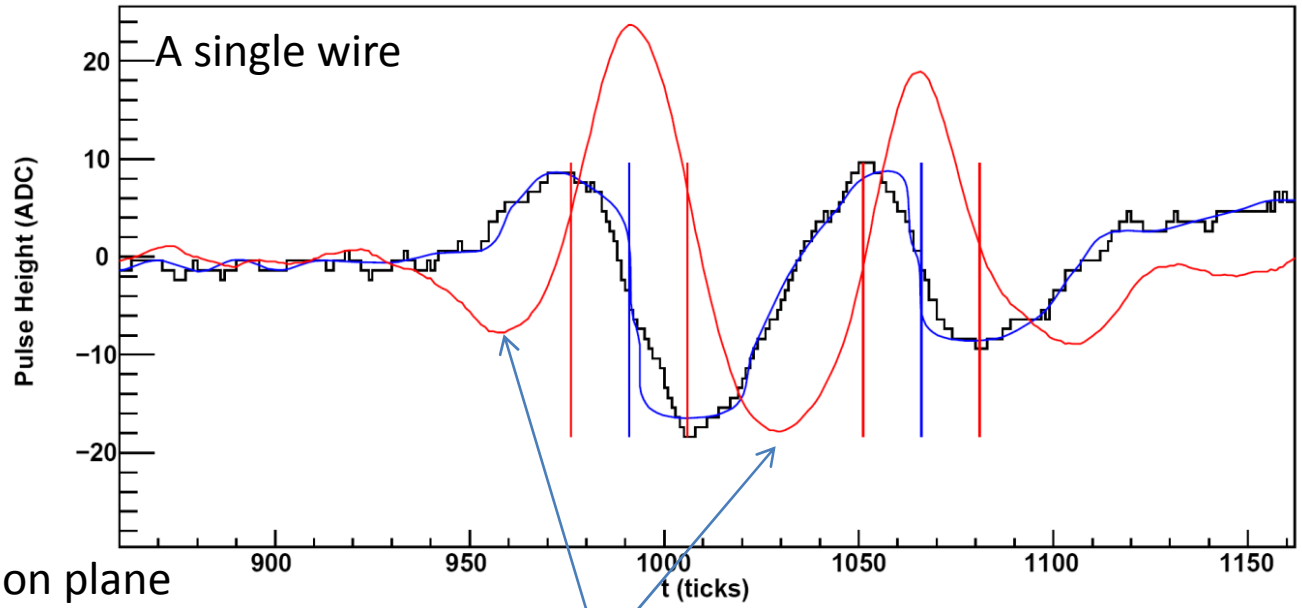
Induction plane

Multiple hits on the same wire

1. Find the time bin *i* that gives the largest value of → $Integral_i = \sum_{i-15}^{i} a_i - \sum_{i}^{i+15} a_i$

2. If Integral$_i$ is above the wire's pedestal by *knob* sigma, count as hit

3. Then, require a slope sign change in Integral$_i$, and start looking for hits again.



Induction plane

Slope sign changes

Raw data

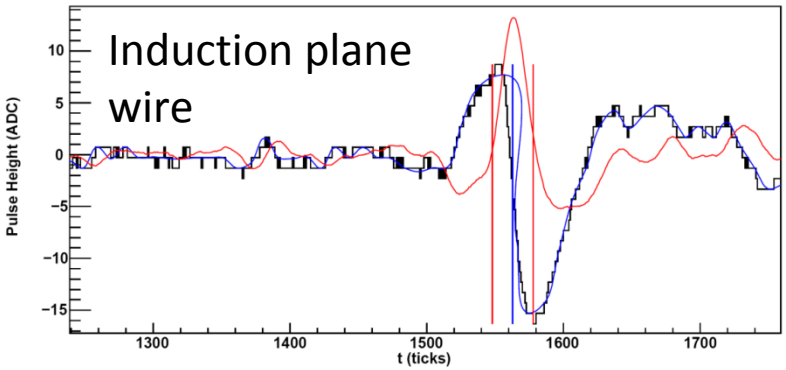
Raw data (smoothed)

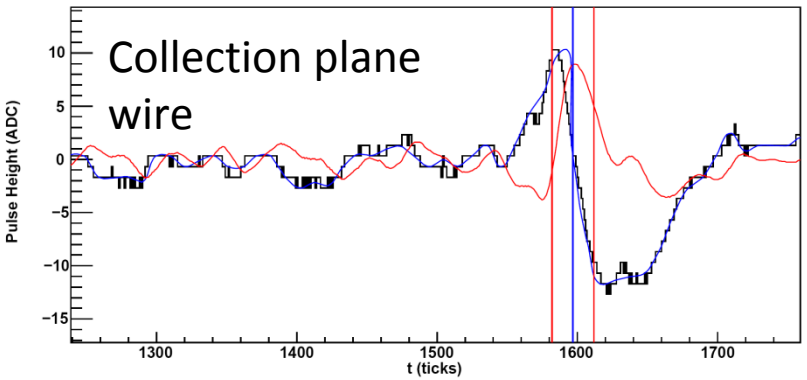
Integral$_i$ (arbitrary units, smoothed)

The knobs

$$Integral_i = \sum_{i-15}^{i} a_i - \sum_{i}^{i+15} a_i$$

- Difference time
- Sigma required to be called a hit
  - All the ADC pulse heights in a given wire are made into a distribution w/ mean (pedestal) and sigma.
- # of hits allowed
- a_i can be equal to ADC_i, (ADC_i-pedestal) , or (ADC_i-ADC_{i-1})
- Definition of “slope” for slope-change multiple hit requirement
  - A slope change over 1 time bin is too susceptible to fluctuation.
- Asymmetry
  - The negative signal is larger (and longer) than the positive signal on both planes. The difference time used can be different on the high and low sides.



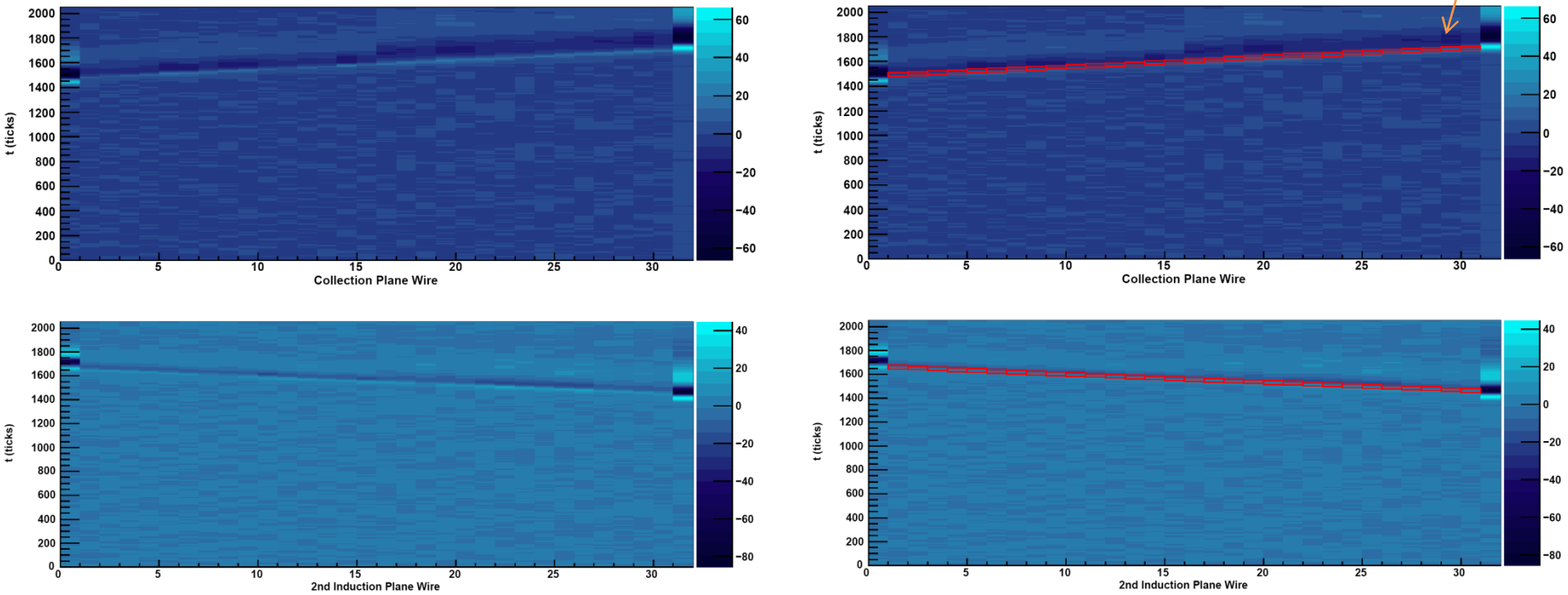


Optimizing hit finding: Large S/N, good multi-hit separation, position/vertex resolution

A one-track example

The height of these boxes is held constant at 30 ticks

Bo data

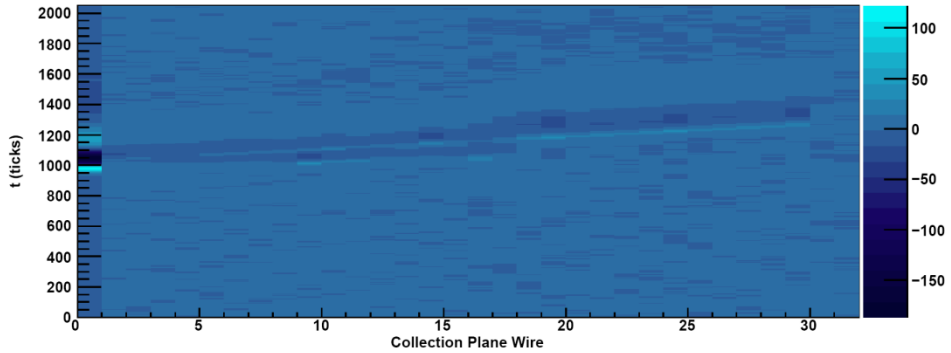


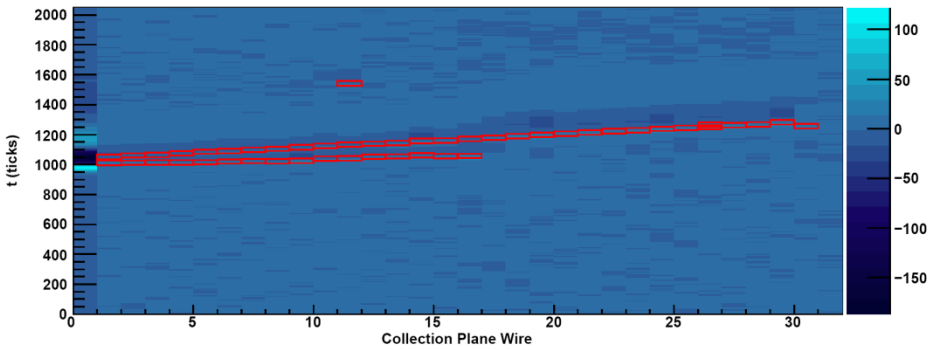
Wires 0 and 31 are not counted as they are really just bundles of wires

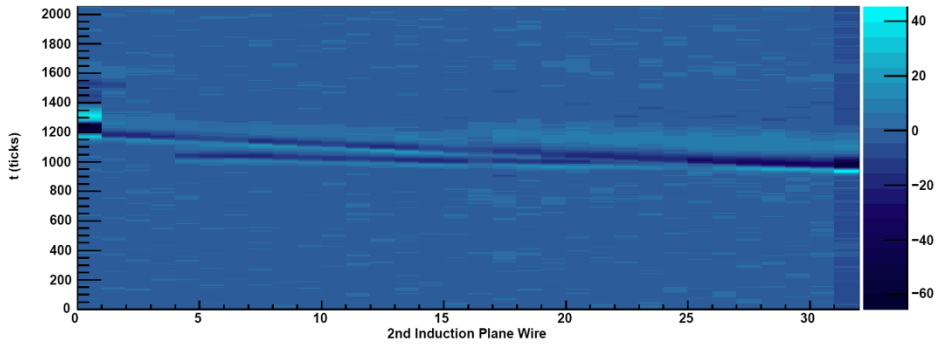
From a visual-scan standpoint, the hit finding works quite well for through-going muon tracks. Also, the algorithm is quite fast, analyzing at a rate of 27 Bo events/min on a 3.2GHz Xeon.

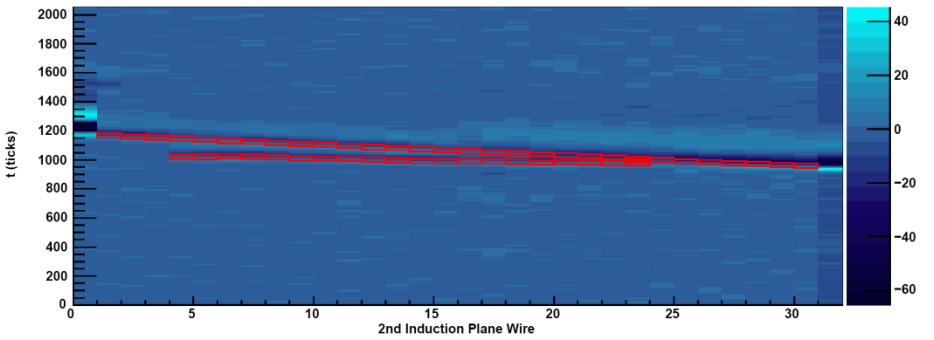
A two-track example

- Bo run 89, event 161 has a nearly-overlapping delta ray. I have used this event to optimize the knobs for multiple-hit hit finding.

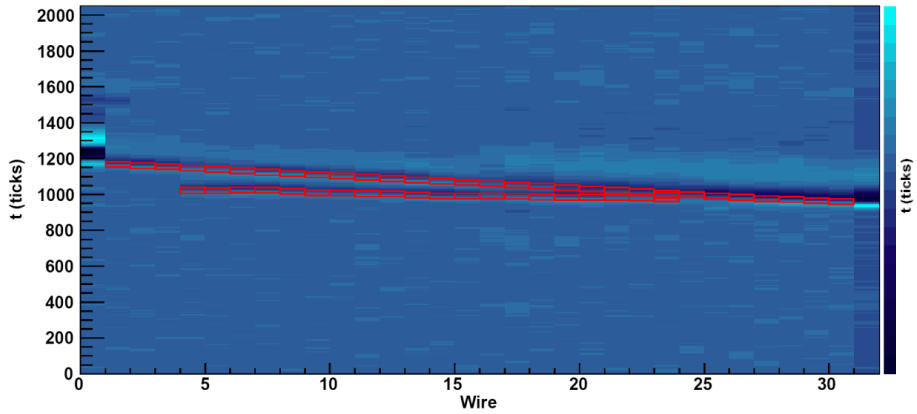


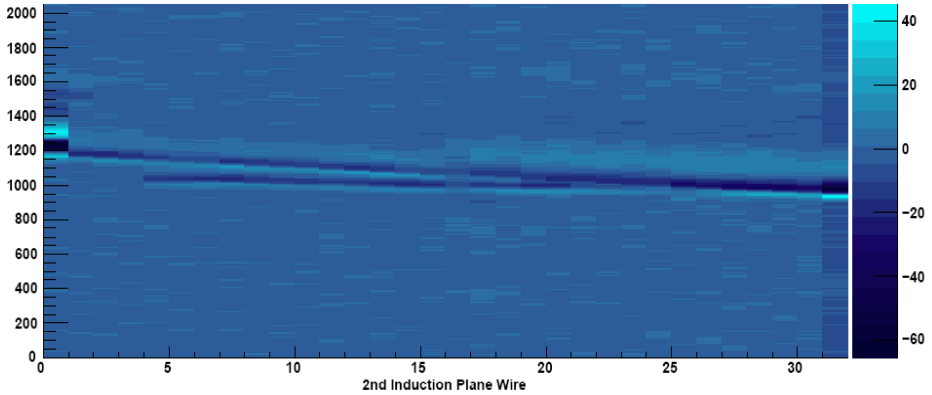


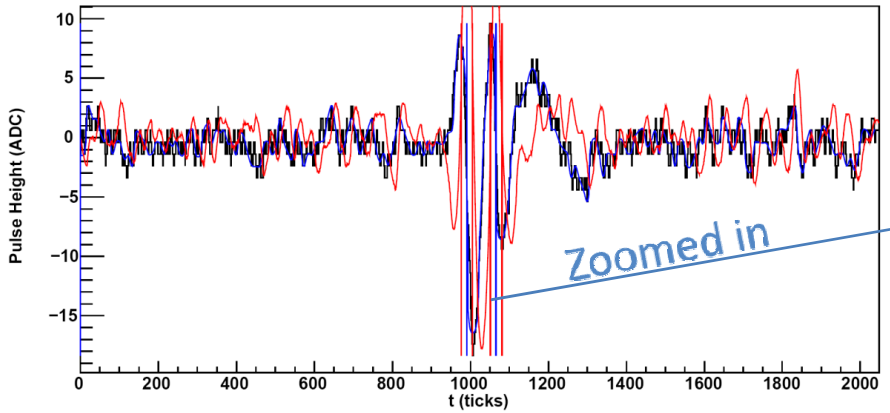


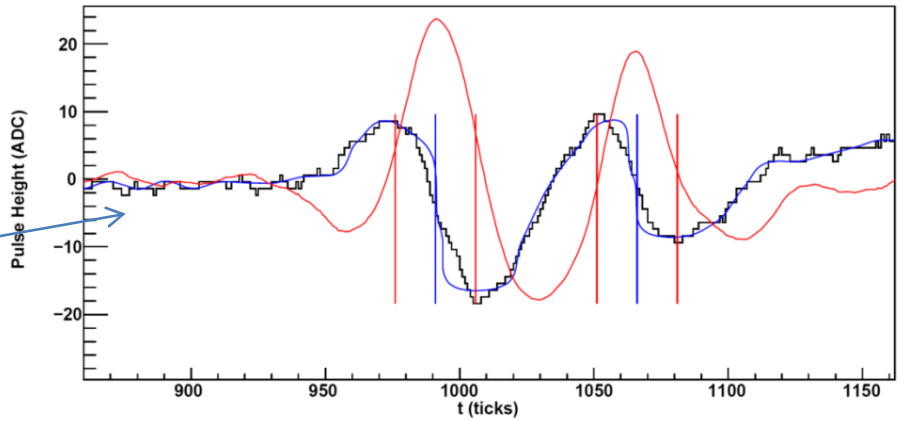


A two-track example cont'd









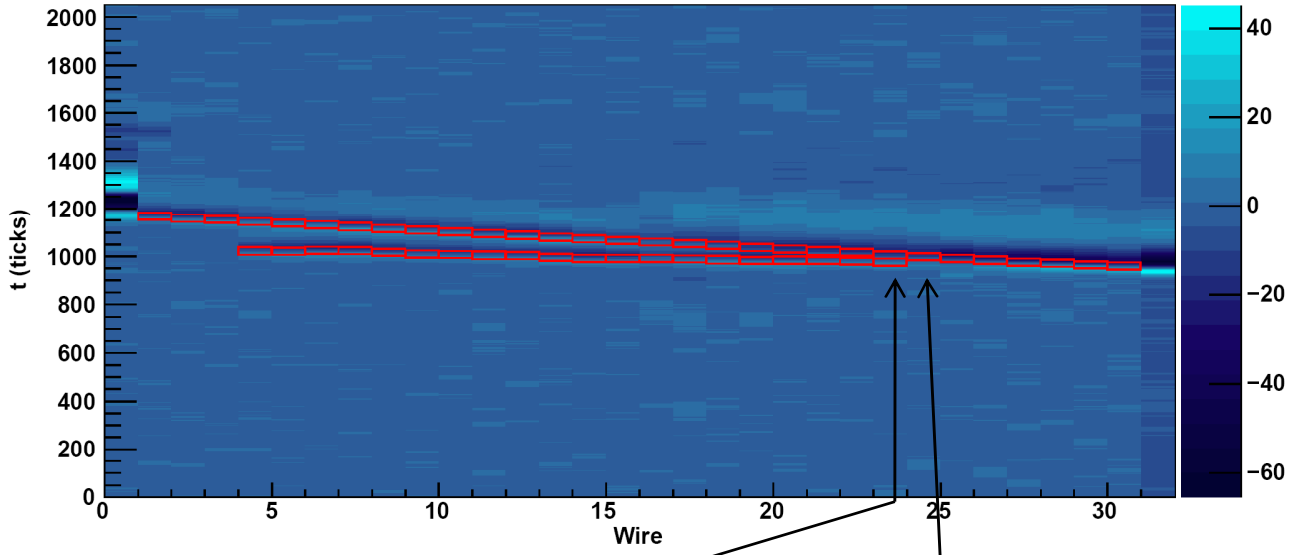
Wire 15

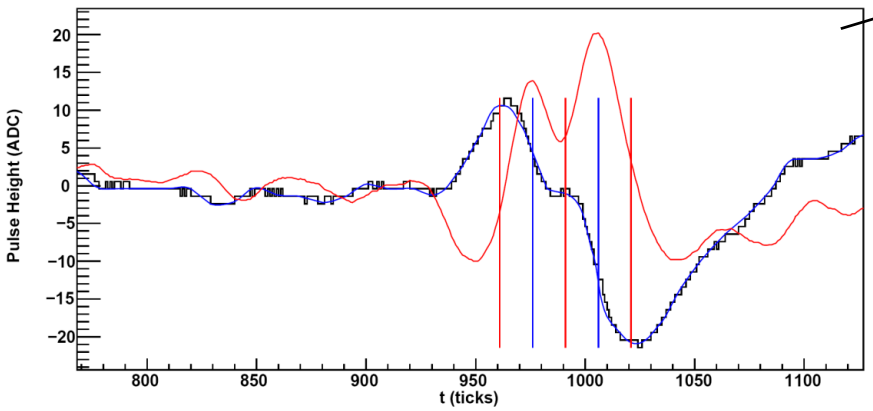
Induction plane

Wire 15

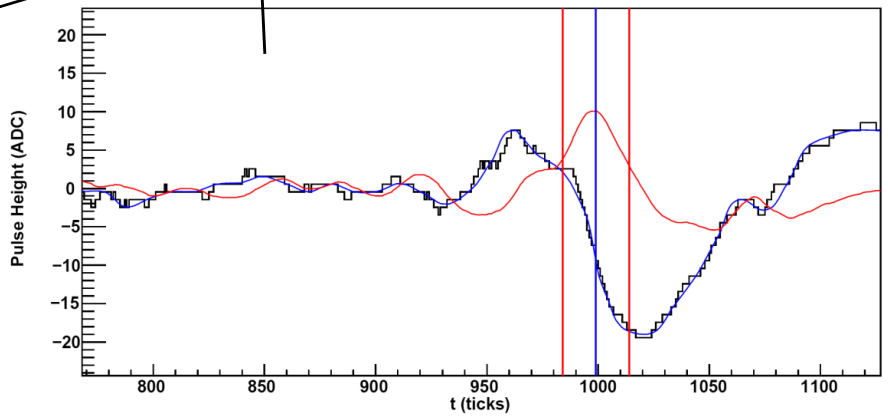
Two hit separation

Induction plane





Wire 23

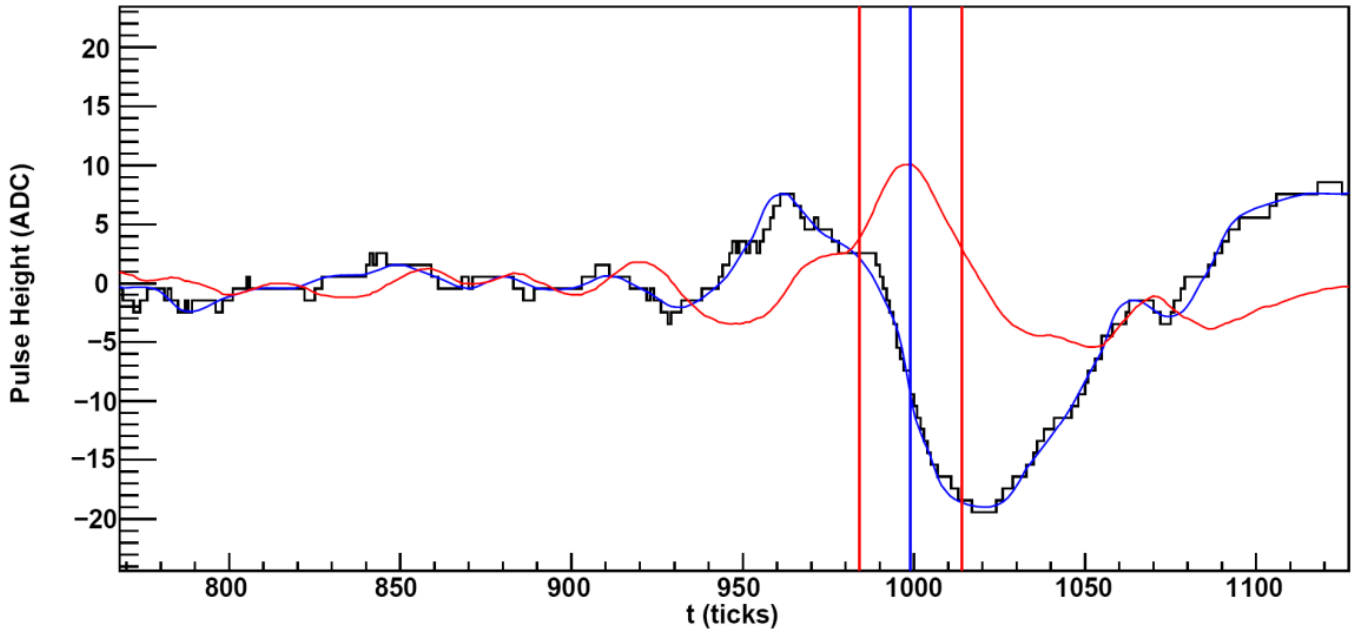


Wire 24 w/ missed hit

~30 tick=6 microsec resolution

FFT is known to be better. See Carl's lartpc docDB #395.

Improving two hit resolution

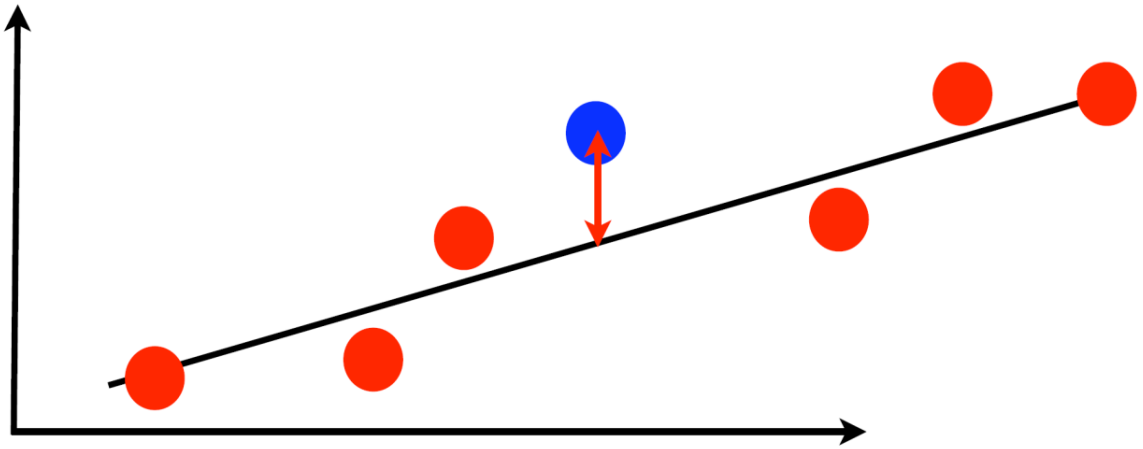


Wire 24 w/ missed hit

How to improve this? Perhaps I should write a sub-hit finder (w/ different characteristic integration time, sigma requirement, etc.) and call it for each hit. Maybe sub-hits can be given a weight or likelihood?

Bo Resolution

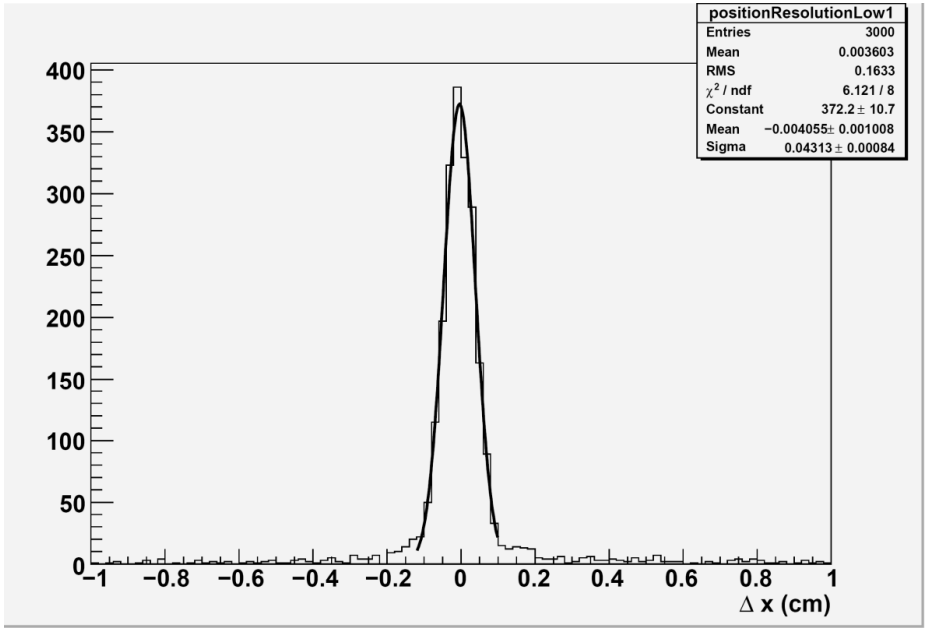
- Use window of 6 nearest neighboring wires to determine resolution for a given hit
- Each wire measures a position in the direction orthogonal to the wire axis (transverse position)
- Use hits to determine distance e- traveled to plane
- Plot transverse position vs distance traveled and fit line to it
- Resolution for a given hit is the residual between the measured position and the fit using the other wires



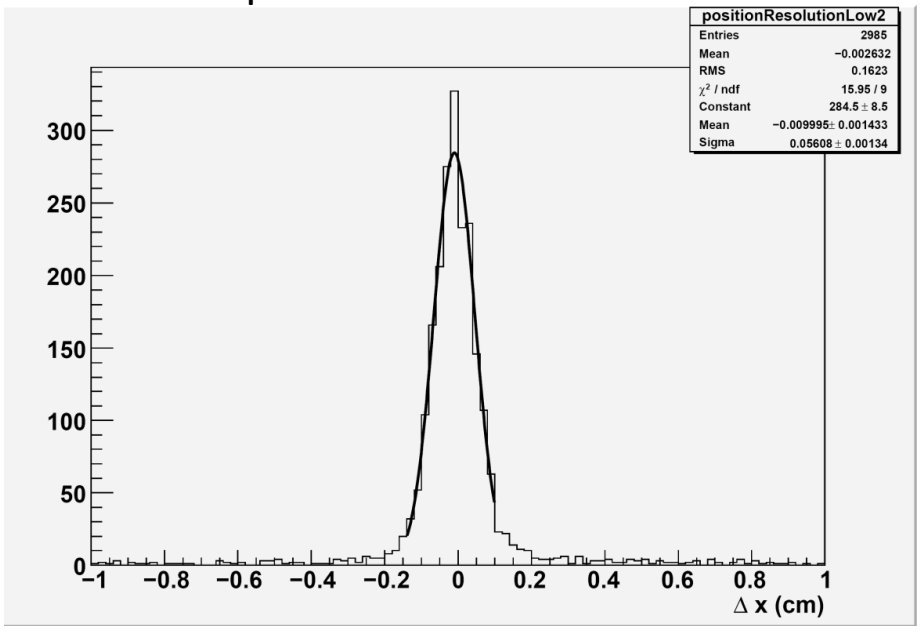
*copied and pasted from Brian's talk (lartpc docDB #377)

Position resolution

Induction plane



Collection plane



These plots were obtained looking only at events where all wires in the plane had at least 1 hit.

These plots are for wires 0-15. Remember that wires 0-15 and 16-31 have different electronics. Resolution on wires 16-31 is ~20% worse.

σ=.043 cm for induction plane and σ =.056 cm on collection plane.

Conclusions

- The difference algorithm is quite simple and has only a few knobs.
- The algorithm is effective for 1-2 track scenarios and should work for higher track events.
  - It will be very interesting to see vertex resolution and multiple hit separation for ArgoNeuT neutrino events with >=2 tracks.
- We could use the difference algorithm wherever fast event processing is needed.
  - Used with trigger to identify spills w/ or w/o a neutrino event.
- Also, it will be useful to study the effects of different electronics choices with the difference method. It might be useful to do a systematic study of Bo electronics configurations using this algorithm.
- FFT is known to be better for two-hit resolution.
  - How well does FFT do with position resolution and how long does the actual FFT wire-analysis process take?