Track Calorimetry Issues

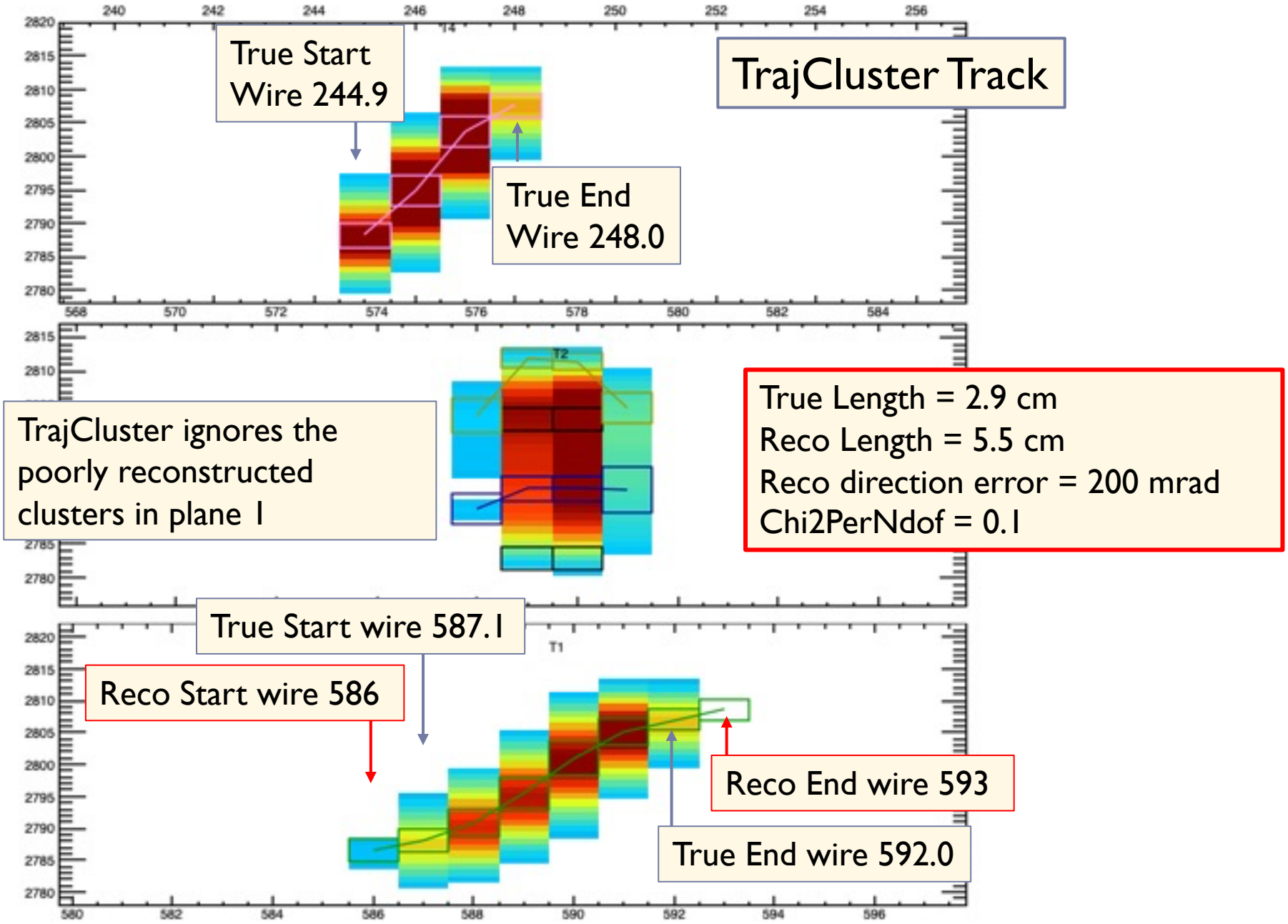
Bruce Baller
August 25, 2020

Motivation

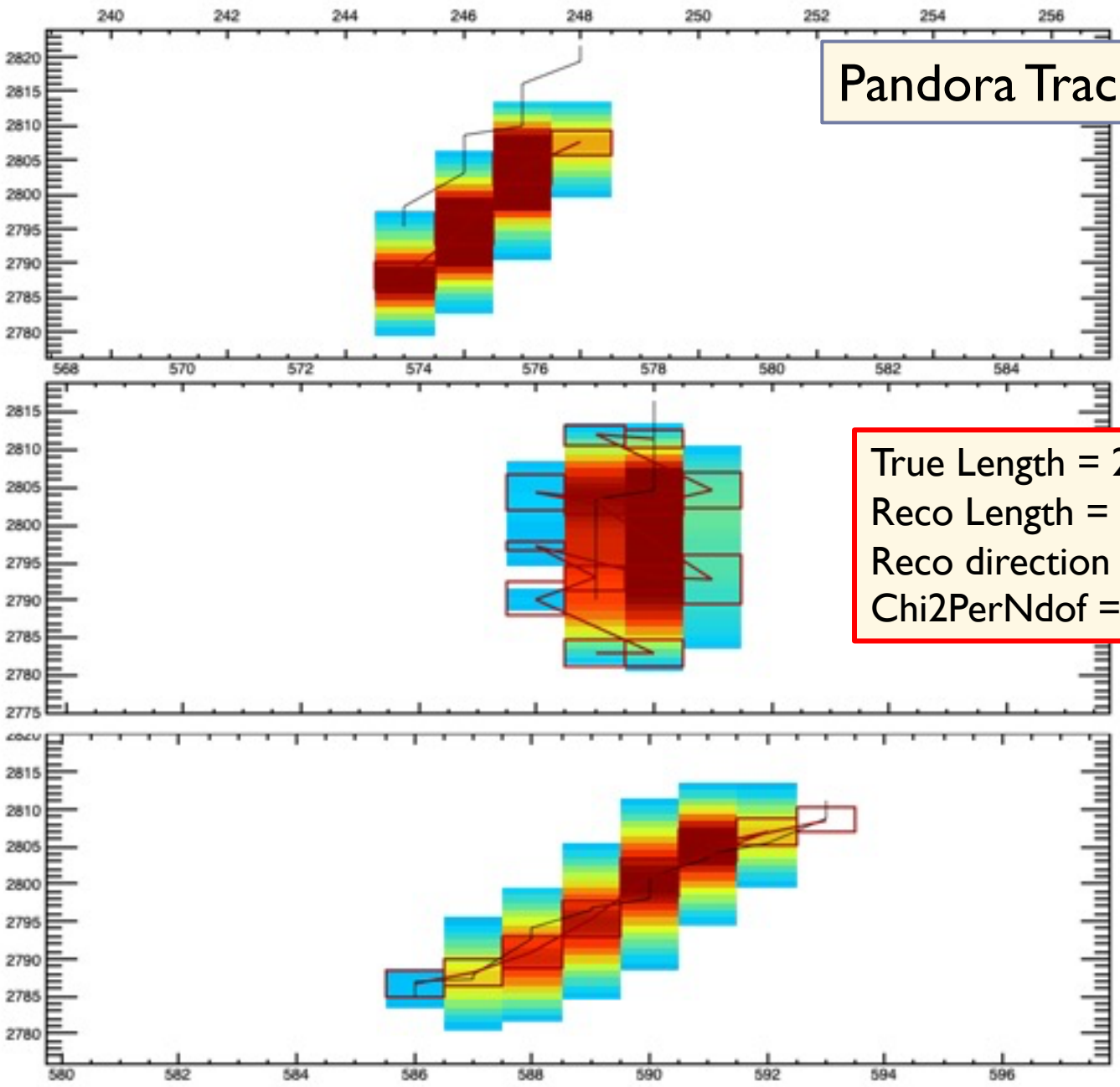
- I am testing a TrajCluster upgrade to produce Tracks with TrackHitMeta and SpacePoint collections
- Tests exposed an inaccuracy in how the Calorimetry module uses Track trajectory and TrackHitMeta data
- This is significant for short tracks

The Problem

- Short track calorimetry suffers from several effects
  - "Short" = 10 – 30 points in three wire planes
    - Example: 50 MeV (KE) protons
    - Generally at large angle wrt the wire planes
  - Large wire spacing (DUNE 0.47 cm) results in large values of dx (pitch > ~1 cm) in the dE/dx calculation
  - Transverse diffusion in long drift TPCs is not negligible
    - σ_T ~ 0.3 cm at max drift in DUNE
  - Calorimetry module assumes that a track stops directly over a wire on average → large uncertainty in dE/dx vs residual range → large uncertainty in PID
- Example on the next slide
  - MC proton, T = 56 MeV, p = 331 MeV, range = 2.9 cm
  - No SCE
  - Generated by Tingjun for this study


TrajCluster Track
True Start
Wire 244.9
True End
Wire 248.0
TrajCluster ignores the
poorly reconstructed
clusters in plane 1
True Length = 2.9 cm
Reco Length = 5.5 cm
Reco direction error = 200 mrad
Chi2PerNdof = 0.1
True Start wire 587.1
Reco Start wire 586
Reco End wire 593
True End wire 592.0

Pandora Track



True Length = 2.9 cm
Reco Length = 4.5 cm
Reco direction error = 226 mrad
Chi2PerNdof = -999

Truth Study using Bethe-Bloch in Excel
DUNE doc #19353

- Used to develop the PIDA algorithm
  - Calculates dE/dx, energy loss, recombination and charge (Q) for user-selected step size and initial kinetic energy
  - Charge Q = ($\mathcal{R}$ * dE/dx) is summed on pseudo-wires

Set so that the proton range is 6.00 wires

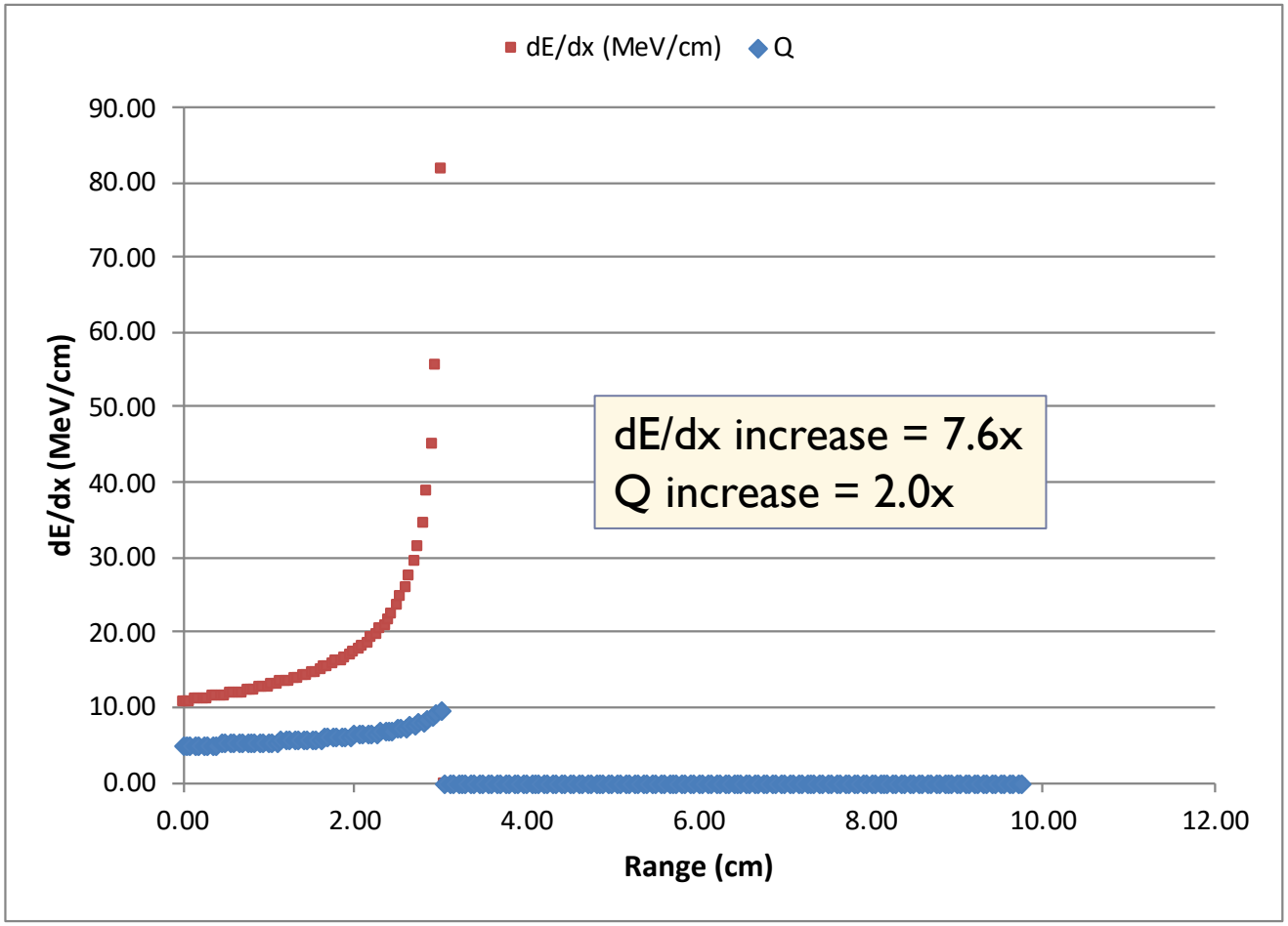
Start T	57.049	MeV
Step	0.050	cm
Range	3.025	cm

Recom A	0.8	
Recom k	0.0486	
Efield	0.5	kV/cm
Wire Space	0.5	cm

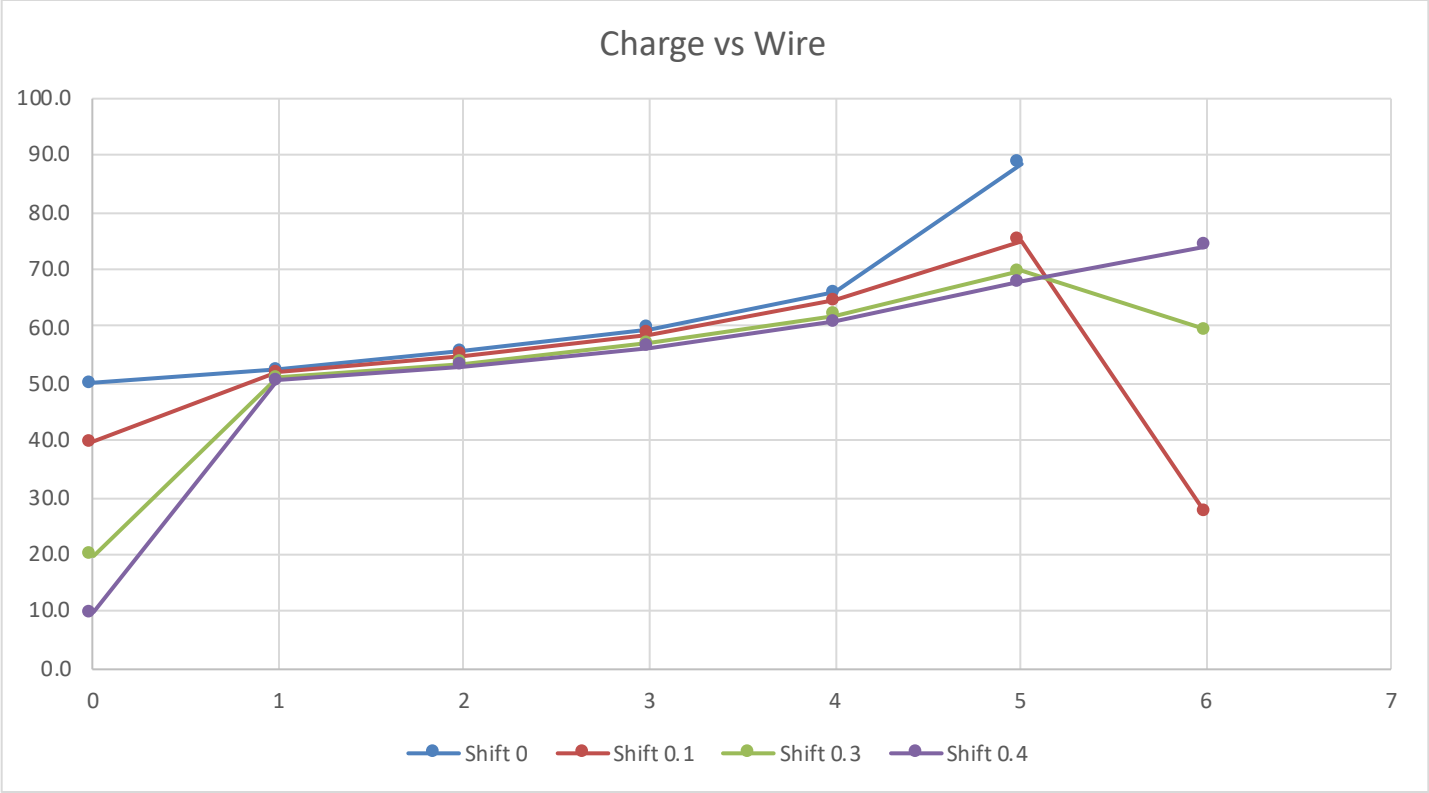
$\mathcal{R}$ = Birks recombination

Range (cm)	Res Range	T (MeV)	dE/dx (MeV/cm)	E Dep (MeV)	Bin Frac	R	flt Wire	int Wire	Chg	Chg * fltWire
	3.03	57	10.77	0.5	1.00	0.46	1.050	1	4.90	5.15
0.05	2.98	57	10.85	0.5	1.00	0.45	1.150	1	4.92	5.66
0.10	2.93	56	10.93	0.5	1.00	0.45	1.250	1	4.94	6.18
0.15	2.88	55	11.01	0.6	1.00	0.45	1.350	1	4.97	6.70
0.20	2.83	55	11.09	0.6	1.00	0.45	1.450	1	4.99	7.23
0.25	2.78	54	11.18	0.6	1.00	0.45	1.550	1	5.01	7.76
0.30	2.73	54	11.27	0.6	1.00	0.45	1.650	1	5.03	8.30

dE/dx and Charge vs Range



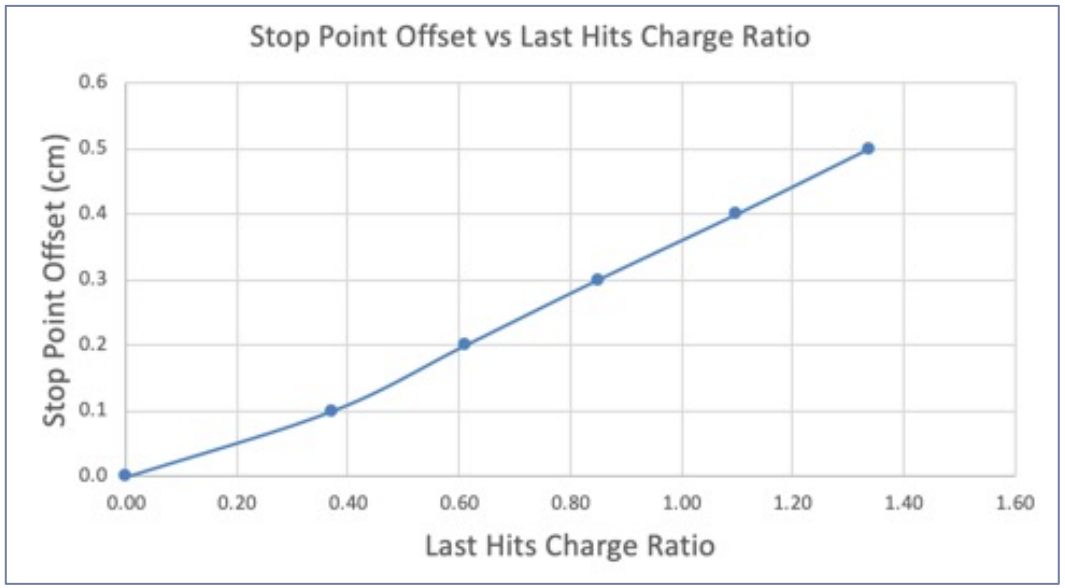
Deposit Q on Pseudo-Wires
Shift the start point by (0, 0.1, 0.3, 0.4) cm



Track Length() will usually be reconstructed as 3.5 cm.
This will result in a PID hypothesis uncertainty.

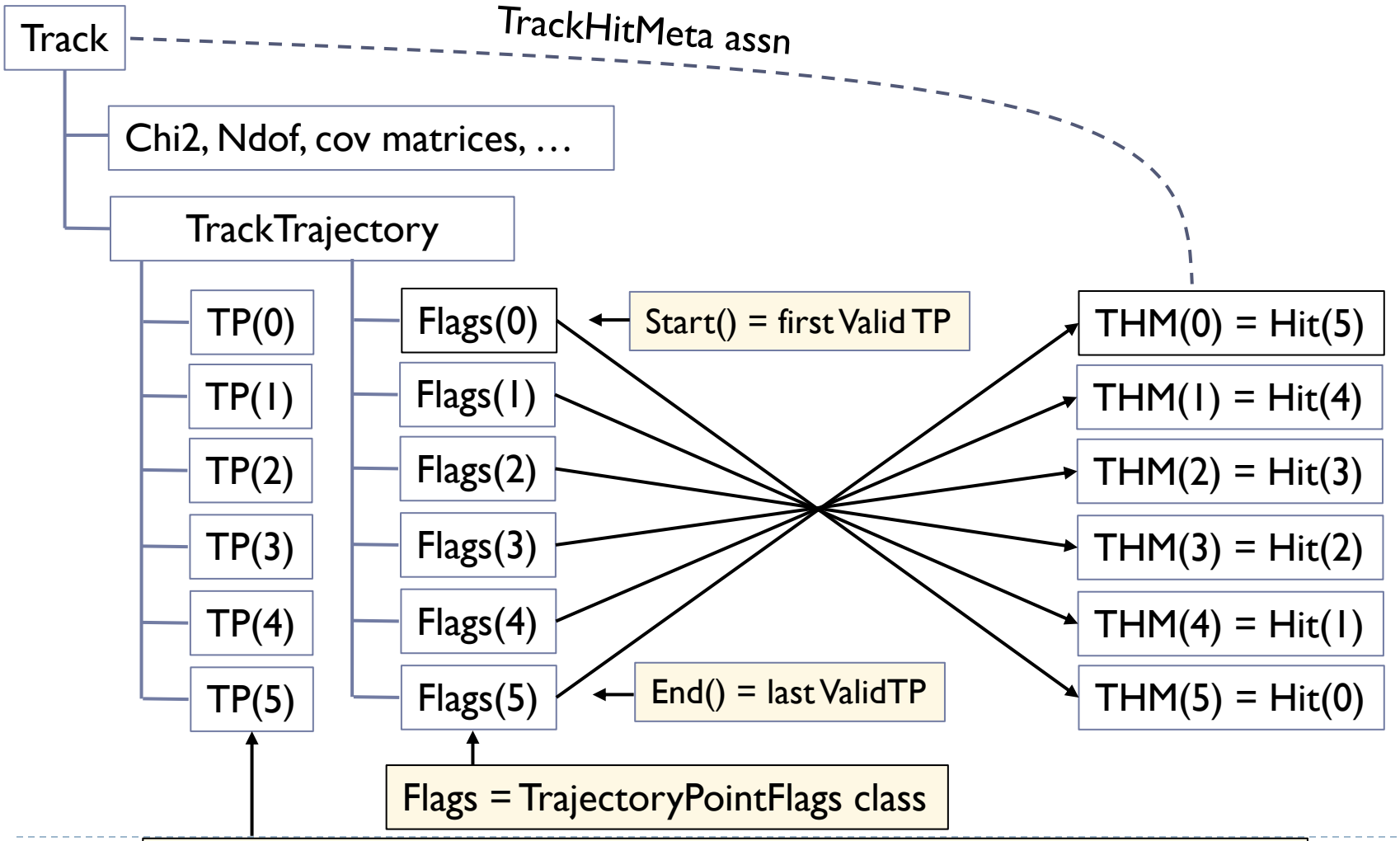
Refine the End Point Positions

- Example using the last two hits at the Bragg peak



- Better results using charge ratio pattern matching of ~6 hits
  - This method was used in the ArgoNeuT recombination study
  - It should work at the Start end as well
- Issues
  - IntersectionPoint(wire1, wire2, … , y, z) integer wire accuracy is insufficient
  - Need a better representation for recob::Track Start and End

Current recob::Track Conventions



Comments, Opinions and a Question

- C: Flags are not fully utilized by track modules
  - Ex: HitIgnored, NoPoint (aka not Valid), Suspicious, Shared, DetectorIssue
  - O: TPs that are Shared with another track shouldn't be used for calorimetry
- C: TrackHitMeta data provides a Hit → Track assn
  - O: There should be a Hit → TP assn
  - C: Calorimetry throws an exception if there is not a one-to-one TrackHitMeta – TP assn
- C: The Track Start (End) method returns the position of the first (last) Valid point
  - C: The Calorimetry module has a "work-around" to set the stopping point = ResidualRange[0] = ½ * track pitch
  - O: The Start and End positions should be defined by the Track module
- C: It is implicit that Valid Trajectory points are ordered from Start position to End position
- O: Ideally, Flags and TrackHitMeta would be packaged in the TrajectoryPoint_t struct
  - This feature might be accomplished using Gianluca's track proxy class
- Q: Should the Start() of a track that is attached to a vertex be the vertex position or should it be the first Valid hit?

Proposal

- Track modules insert two special TPs to define the Start and End positions
  - The positions are defined using an algorithm like that described on slide 9
  - Define two new TP FlagTraits: TrackStart and TrackEnd
  - No hits are associated with these TPs, but they need to be Valid and be the first and last Valid points in the trajectory
    - Requires modifications to Calorimetry_module and maybe other
  - Hit charge in the (!Valid) TPs outside of the Start – End range would need to be included in the sum of the total track charge
- Add an IntersectionPoint method that allows using wires with float precision