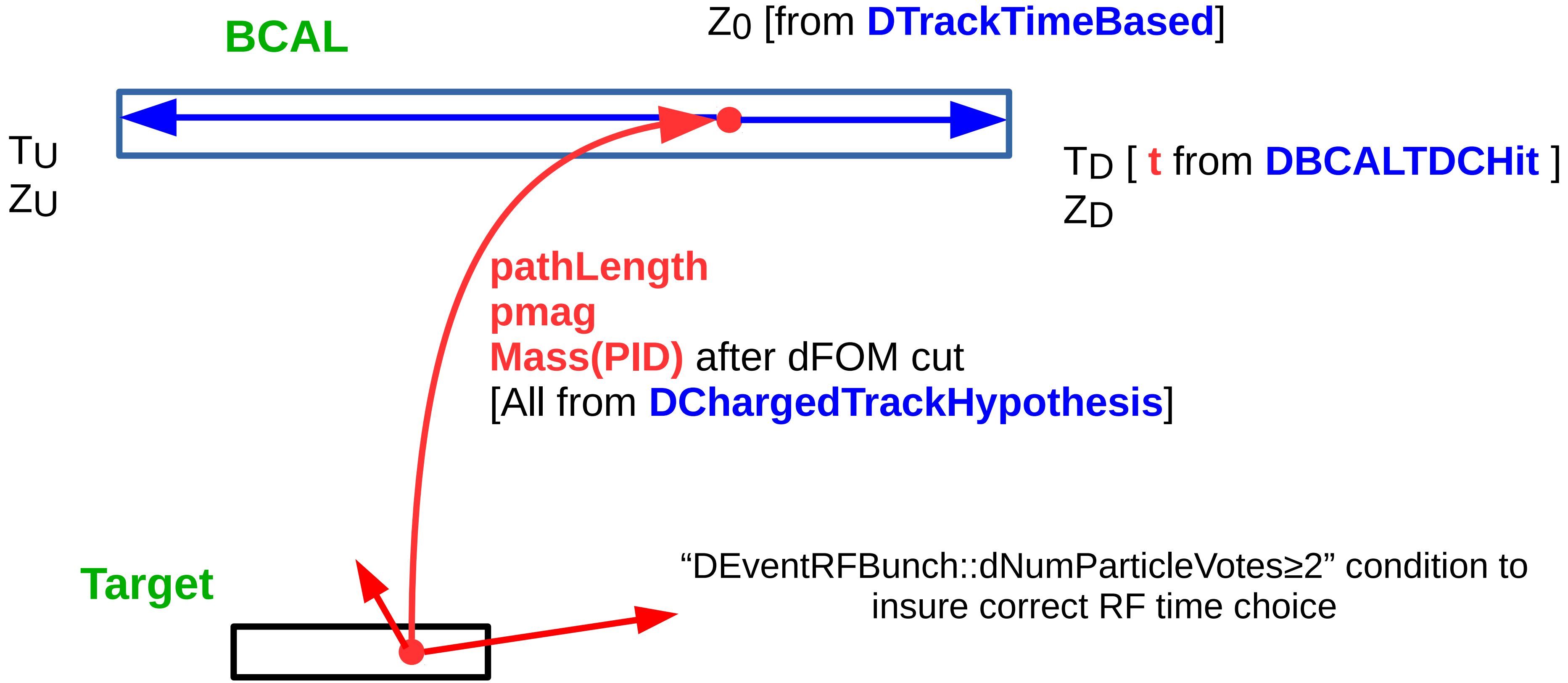
Time Calibration of BCAL

Andrei Semenov (UofR)

August 18, 2016

BCAL Time Calibration with Charged Tracks



T: chosen RF time, propagated to the z-vertex of the track in the target

[**T** from **DVertex** ~~or **t_0** from **DChargedTrackHypothesis**?~~]

BCAL Time Calibration with Charged Tracks (cont.)

- **Channel-by-channel spectra** for

$$T'_{U/D} = T_{U/D} - T - \mathit{Time\text{-}of\text{-}Flight} - \mathit{Time\text{-}of\text{-}Light\text{-}Propagation} =$$
$$= T_{U/D} - T - pathLength*pmag/SQRT(Mass^2+pmag^2)/c \; - |Z_0-Z_{U/D}|/v_{eff}$$

should be centered at zero (*time shift*), and the dependence

$$T'_{U/D} = f\,(Pulse\ Hight_{\ U/D})$$

should be removed (*time-walk correction*).

- **Problem:** Charged particles does not cover whole pulse-hight dynamic range.

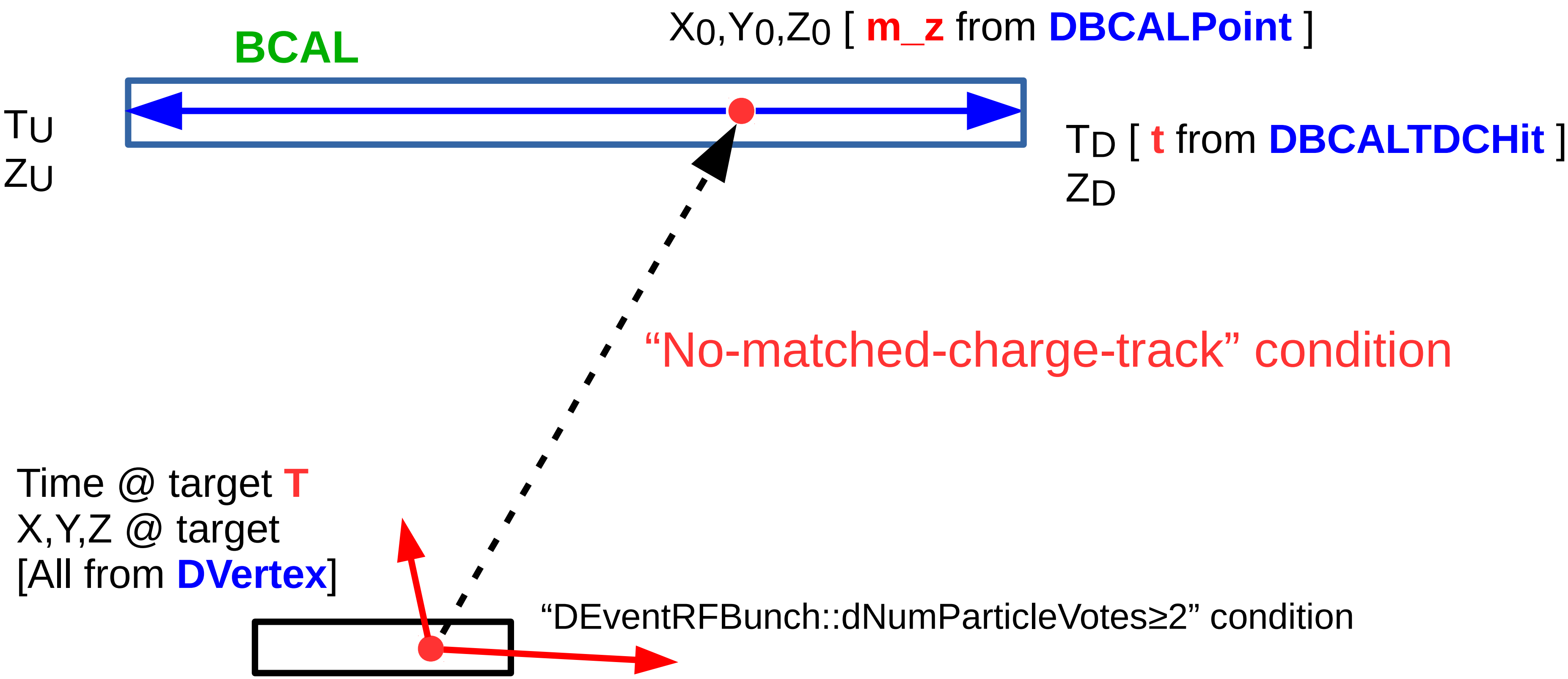
Solution: Use neutral (viz., “not-matched-with-charged-tracks”) showers for the time calibration in addition to the charged tracks.

- **Cross-check:** Invariant-mass spectra for charged particles with calibrated

$$TOF = (T_U+T_D)/2 \; - T$$

should provide correct pion, proton and kaon masses.

BCAL Time Calibration with Neutral Showers

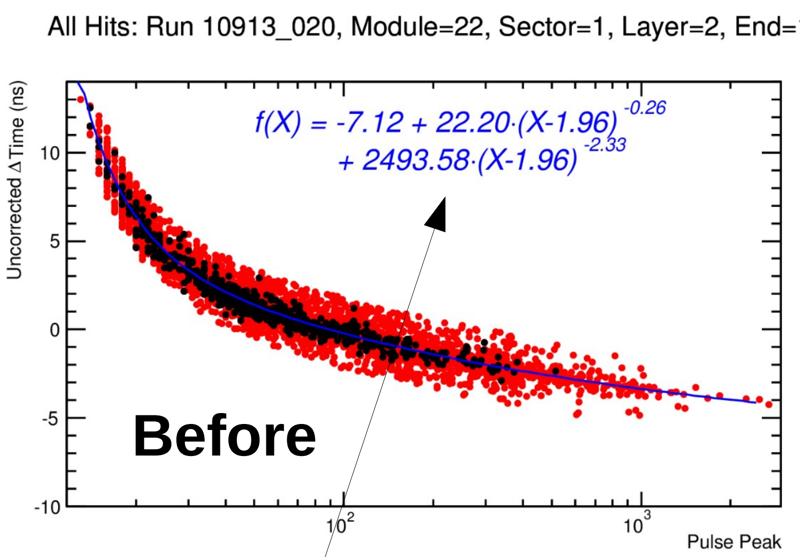


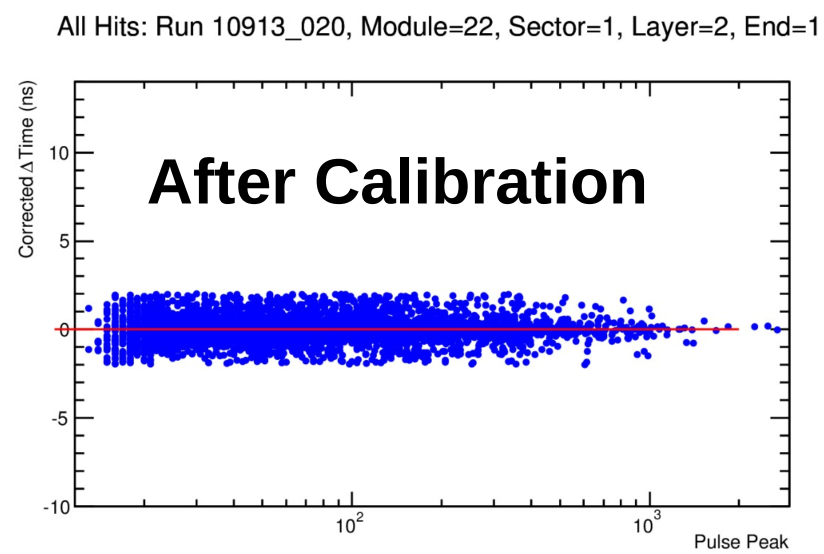
$$T'_{U/D} = T_{U/D} - T - \textit{Time-of-Flight} - \textit{Time-of-Light-Propagation} =$$

$$= T_{U/D} - T - SQRT((X_0-X)^2+(Y_0-Y)^2+(Z_0-Z)^2)/c - |Z_0-Z_{U/D}|/v_{eff}$$

The BCAL Z-hit position (Z_0) comes from BCAL timing => Iterative procedure

BCAL Time Calibration with Neutral Showers

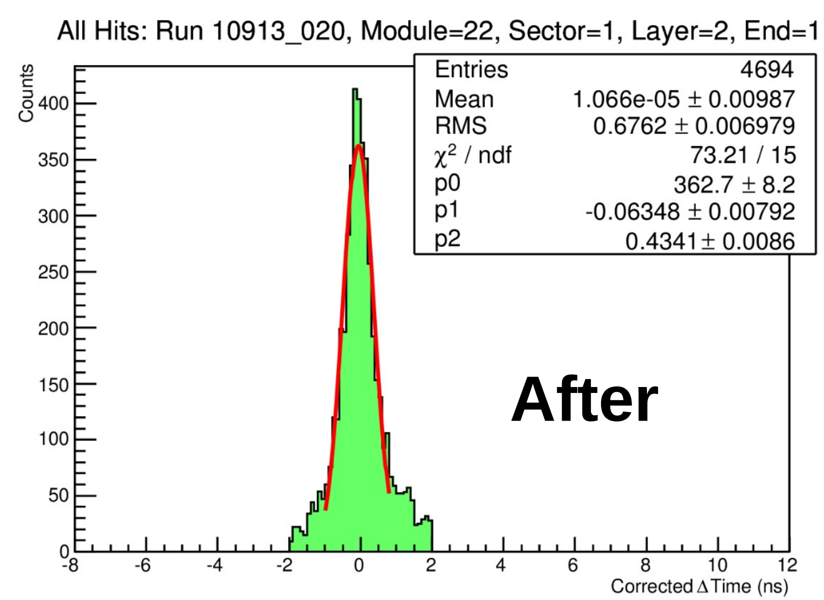




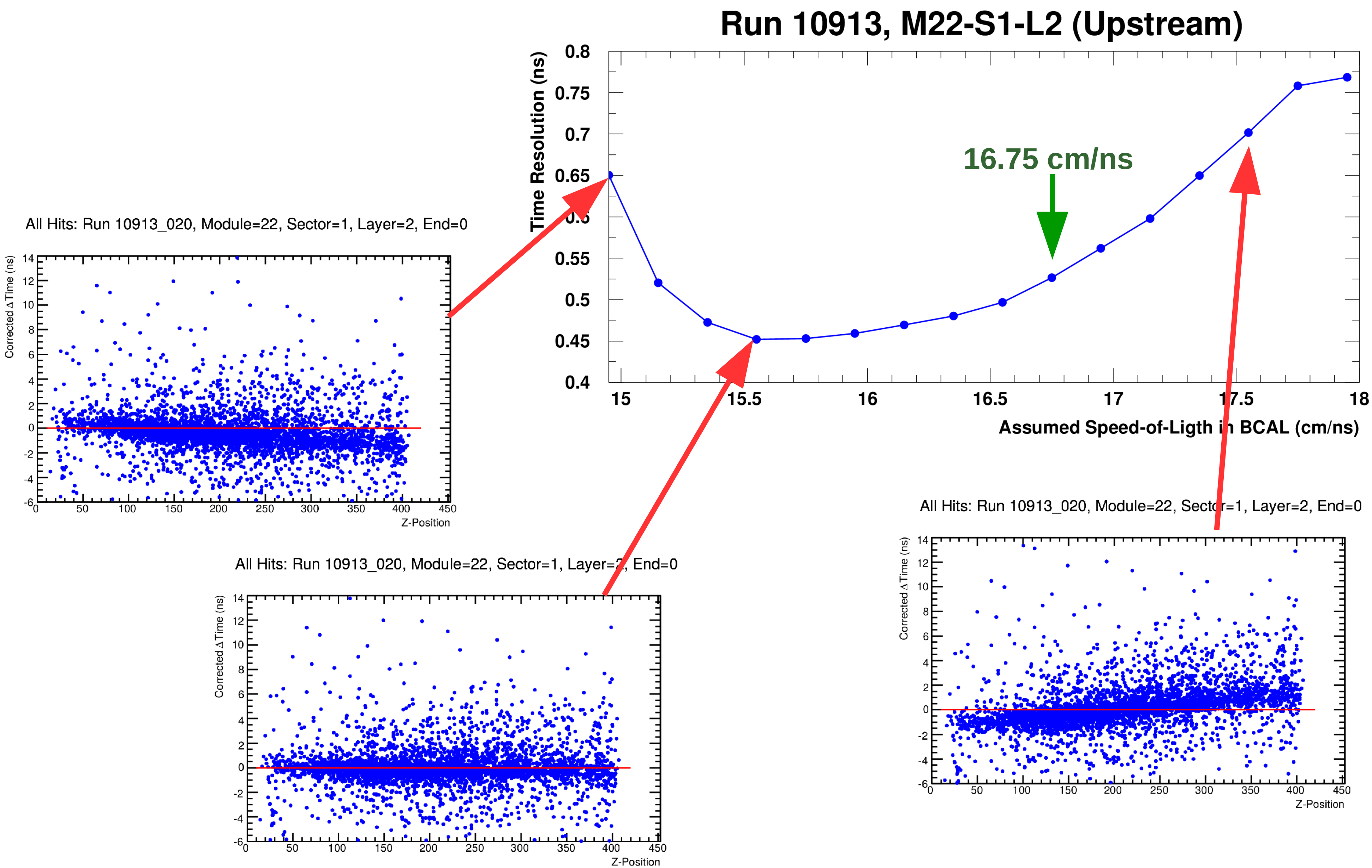
Calibration Function:

Level +
Power Func. (Middle-High ADC Range) +
Another Power Func. (Low ADC Range)

[Note: Red points are for the neutrals, and black points are for the charged tracks)



Z-Dependence



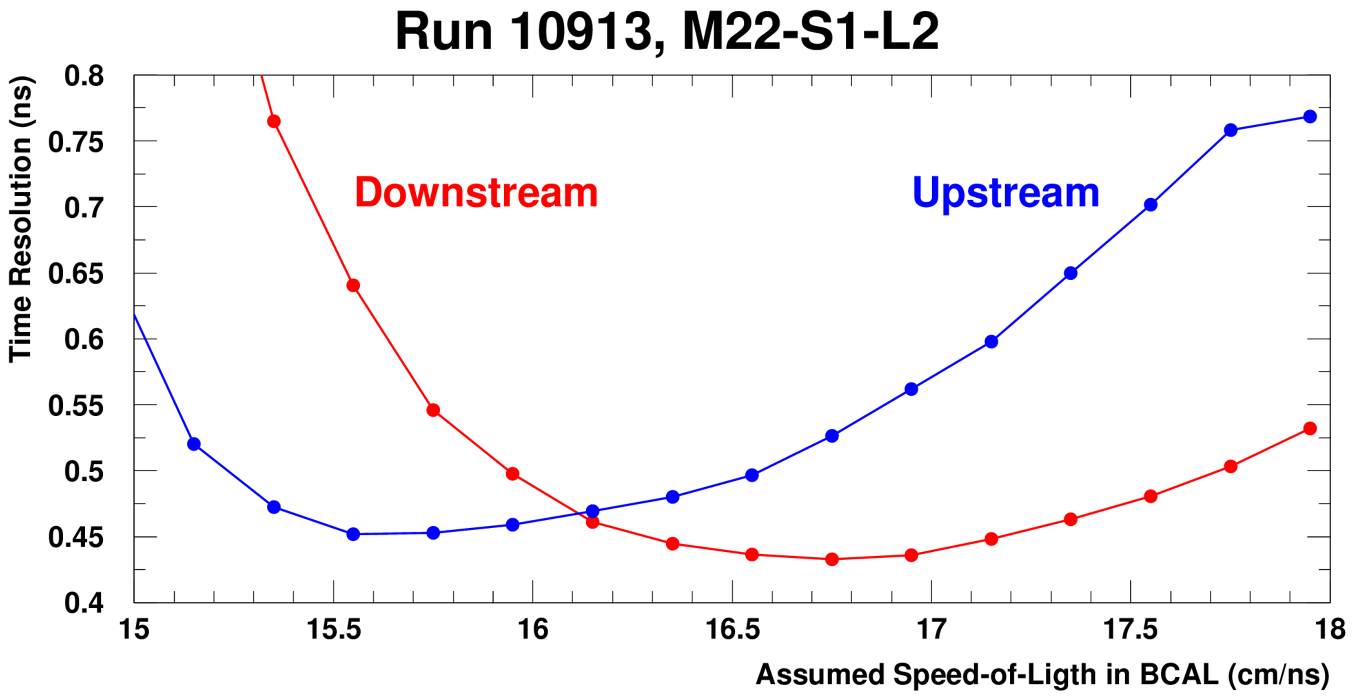
Hypothesis #1: Different cells have different light-propagation velocities. Different distances from the photodetector creates different error in timing in the case of wrong assumed light-propagation velocity => Slope => Varying the assumed light-propagation velocity, we can find the minumim in the time resolution (viz., no z-slope) that corresponds to the “real” velocity.

The minima for upstream and downstream readout must be in the same place.

Hypothesis #2 (by Mark and George): Shower size removes part of the travelling distance for the light inside the calorimeter => Slope even if the velocity is correct => Varying the assumed velocity, we can compensate the “original” slope, but the minumum in time resolution is not correspond to the “real” velocity.

Because the upstream readout is more affected with the shower-size distortion, the minimun for the upstream readout should be for lower velocities than the one for the downstream readout.

Hypothesis #2 is Correct



To address Z-dependence, we can:

1. Do nothing. (Bad idea: time resolution will be about 600 ps instead of 450 ps + “z-systematics” in time of about 1 ns from upstream end of BCAL to downstream end)

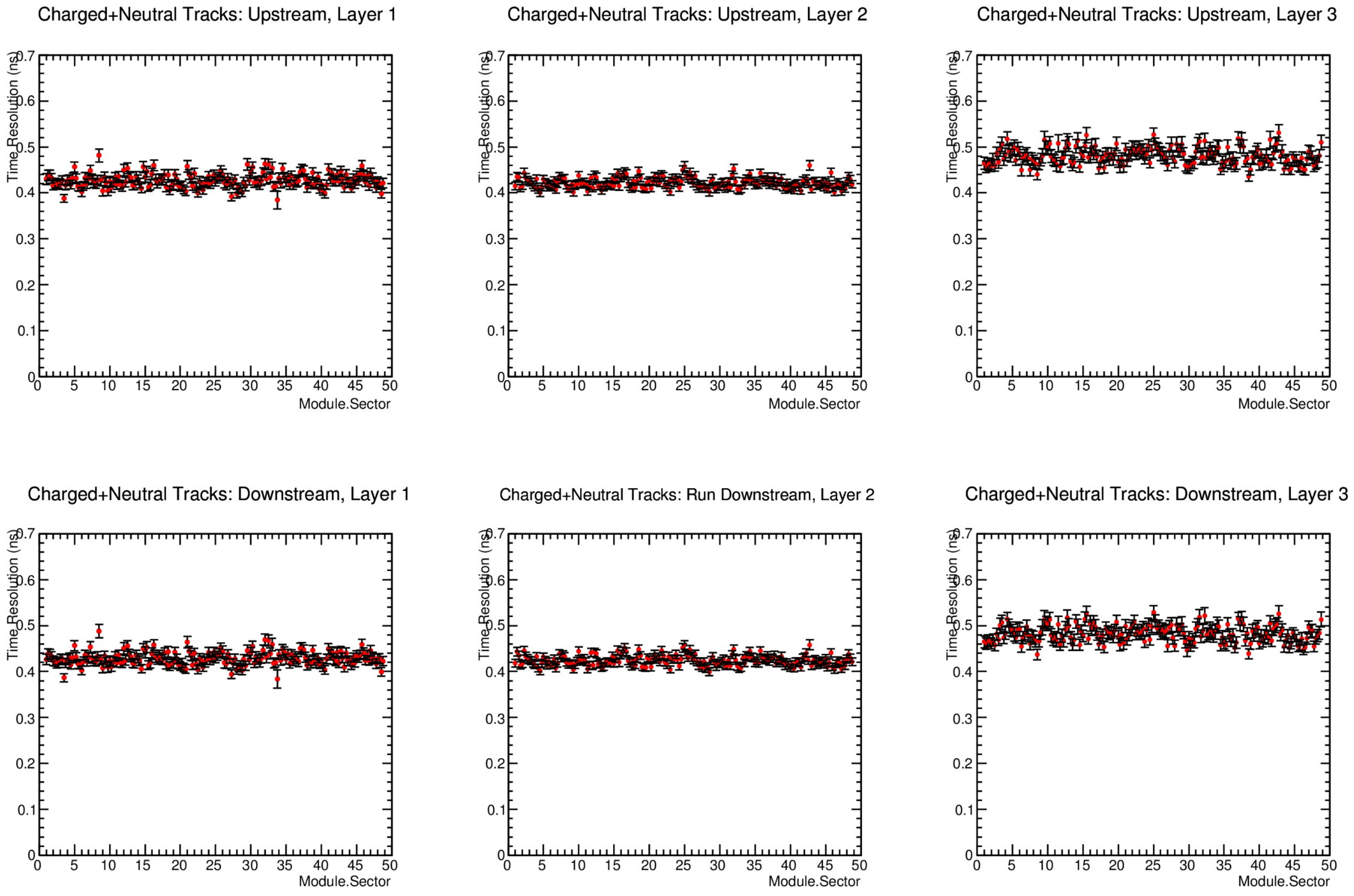
2. Try to calculate or simulate the shower size and use it in the calibration/timing (Bad idea: too complicated)

3. Use additional step in the time calibration to remove z-dependence (Not so bad idea but requires the knowledge of the z-position of the point for the processing time on the hit level that is not good)

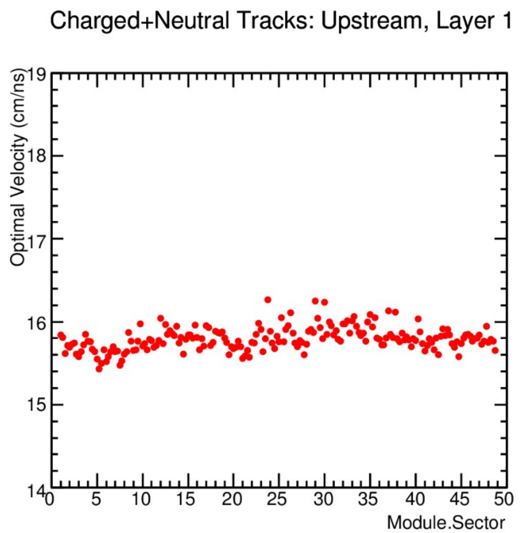
4. Use different “optimal” effective velocities for upstream and downstream times of each cell in the calibration and time processing. (Good idea: different “optimal” velocities remove the z-dependence, and adjusting the “level” constant in the calibration function, we bring the calibrated time to the “correct” physical value again.) In this case, z-coordinate in the local BCAL system will be:

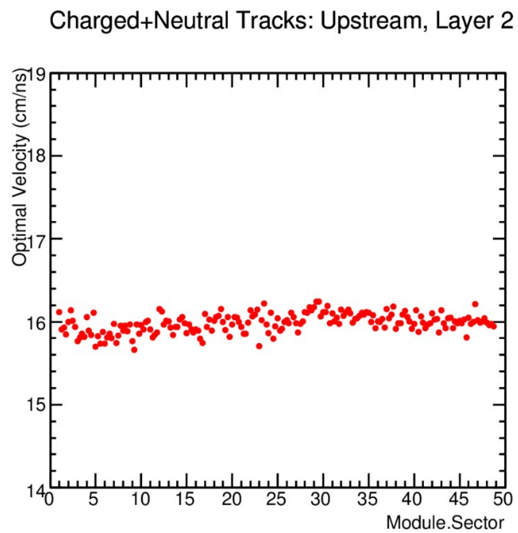
$$Z = ((tUp-tDo)*vUp*vDo + 0.5*fiblen*(vUp-vDo)) / (vDo+vUp)$$

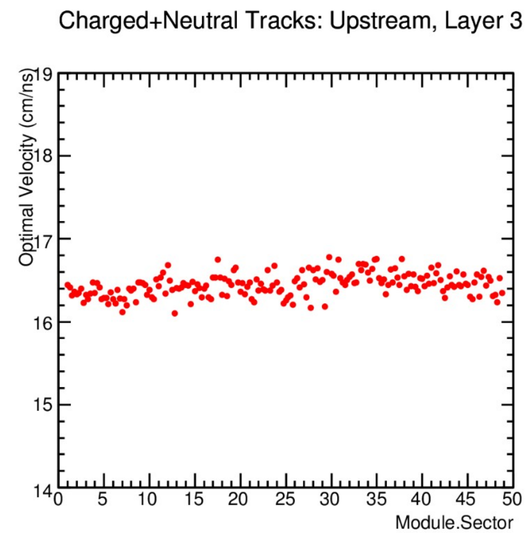
Time Resolutions After the Calibration

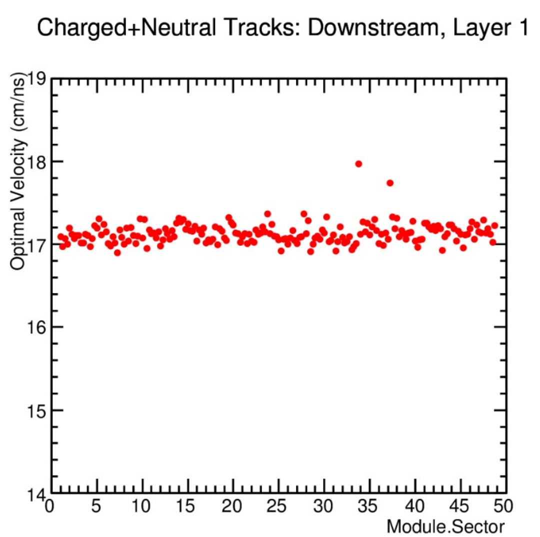


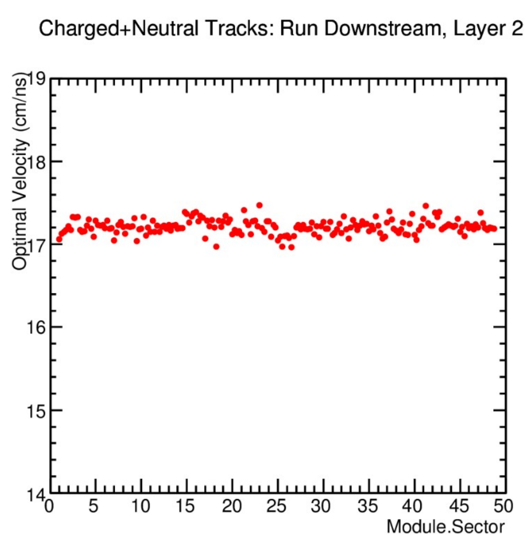
Effective Velocities After the Calibration

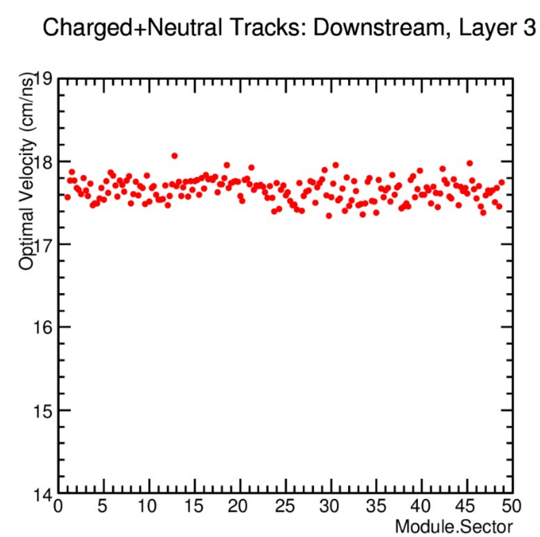




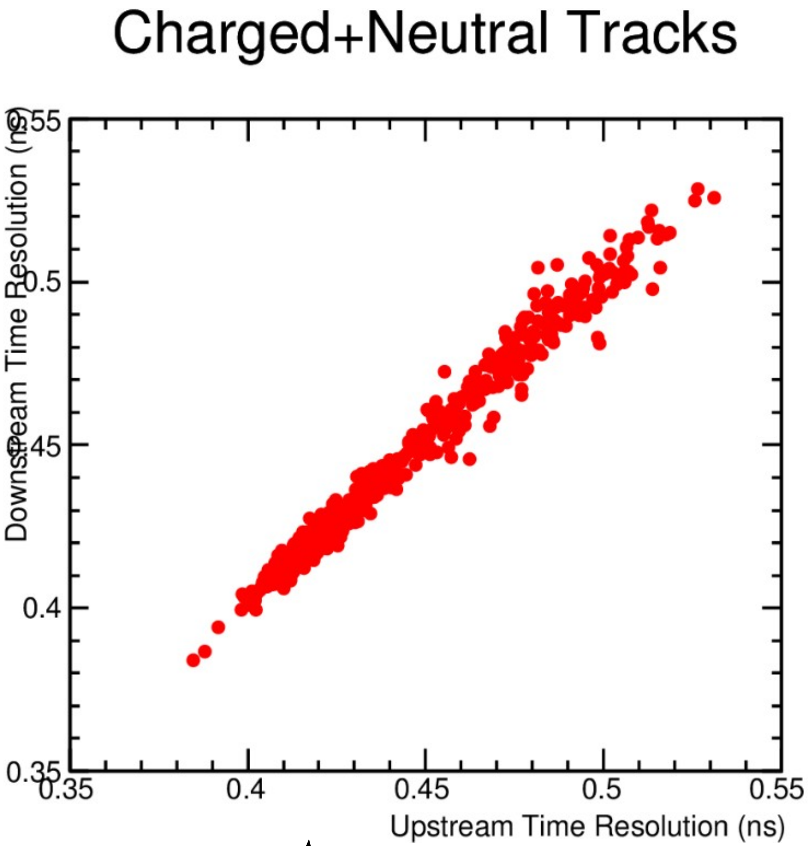


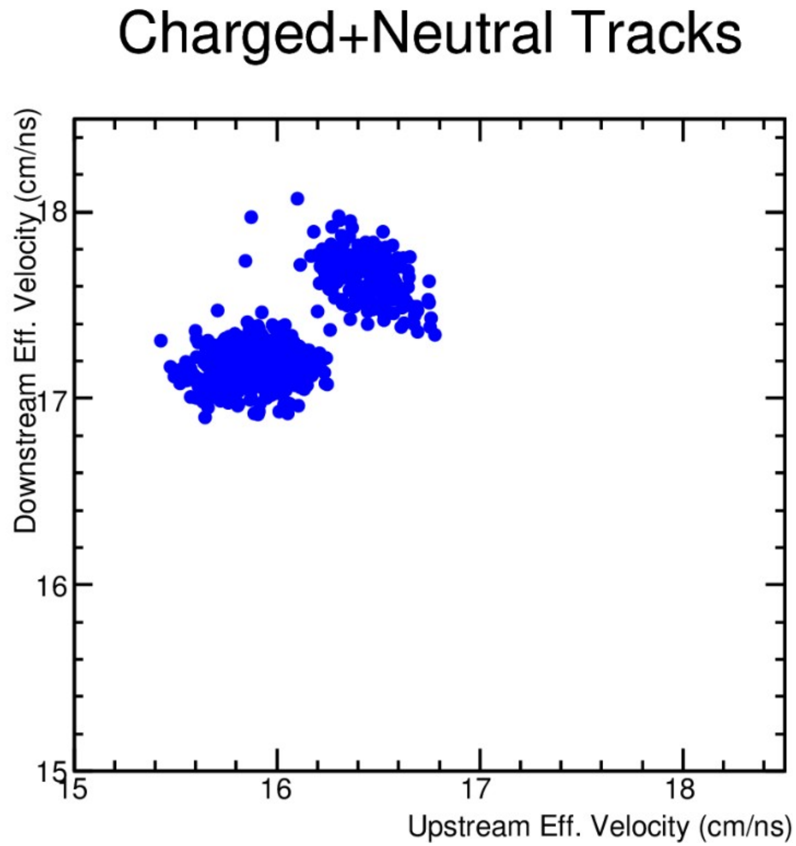






Upstream-Downstream Correlations

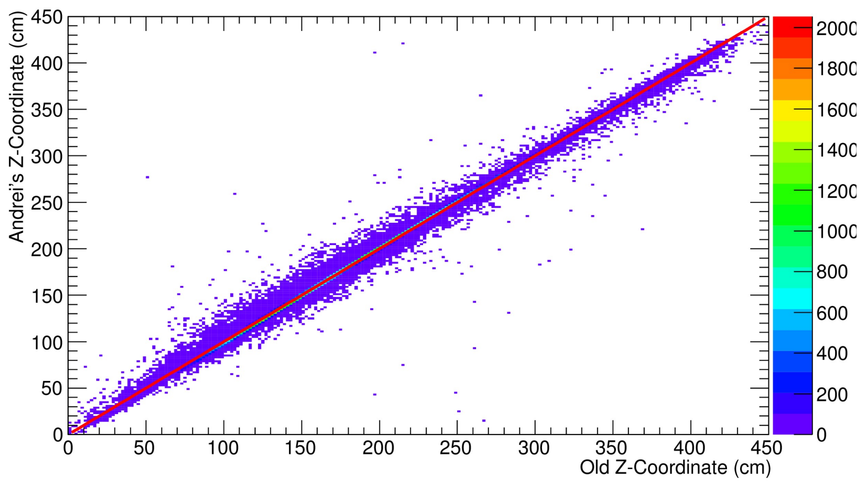




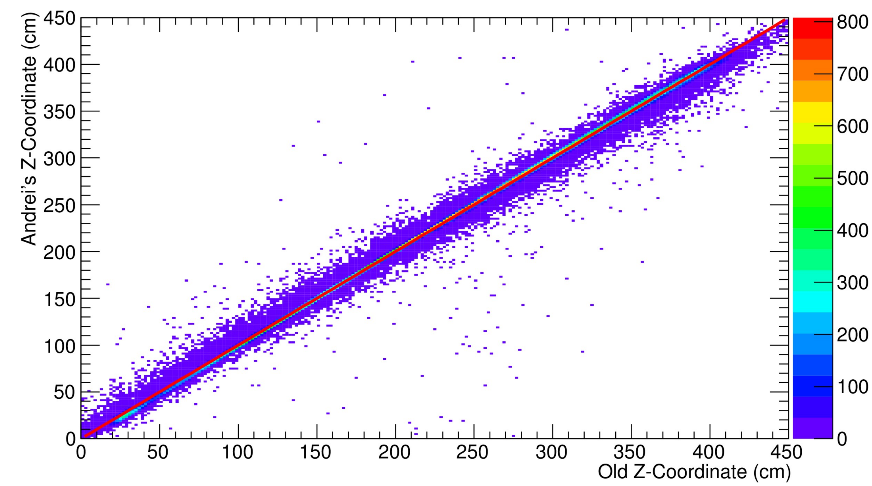
Major fraction of the time resolution is belong to the BCAL cell (not photodetector)

Z-Coordinate: Andrei's vs OLD (fADC timing)

Charged

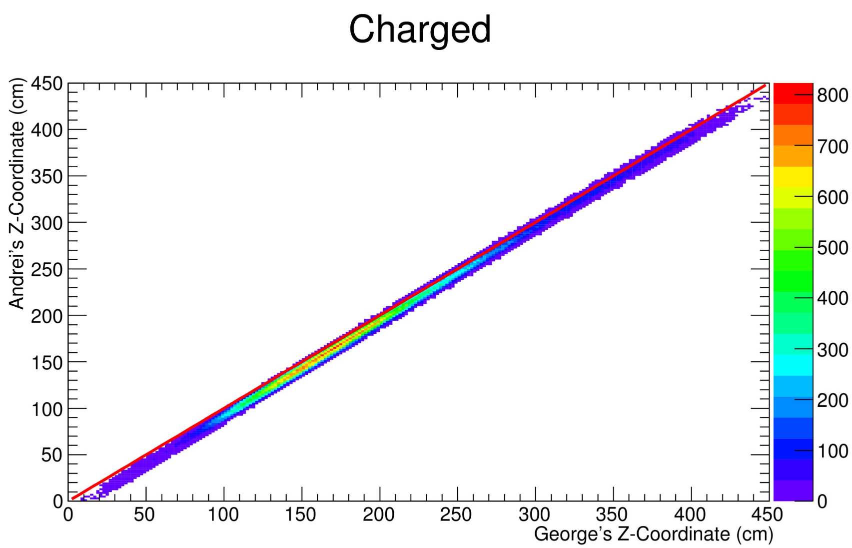


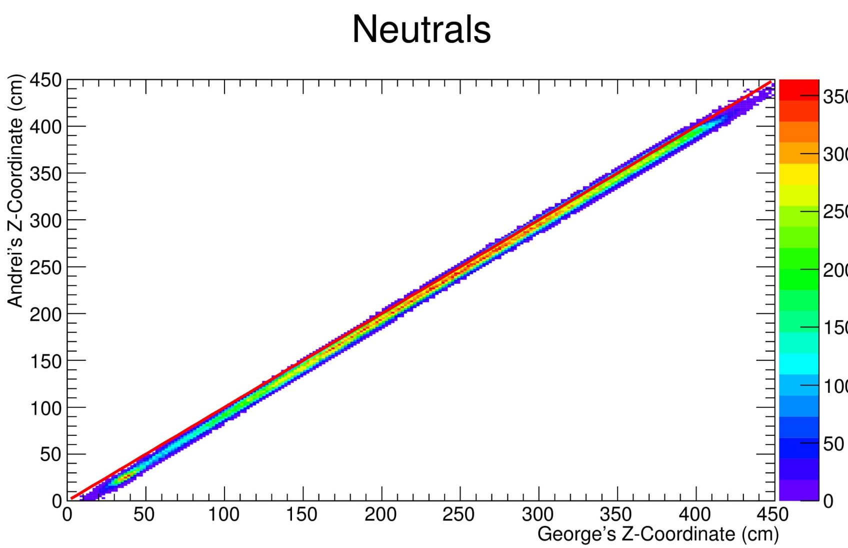
Neutrals



Centroid for Andrei's z-coordinates agrees with the OLD ones,
but “scattering points” assume
that OLD z-coordinates have worse accuracy
(for the proof, see the next slide)

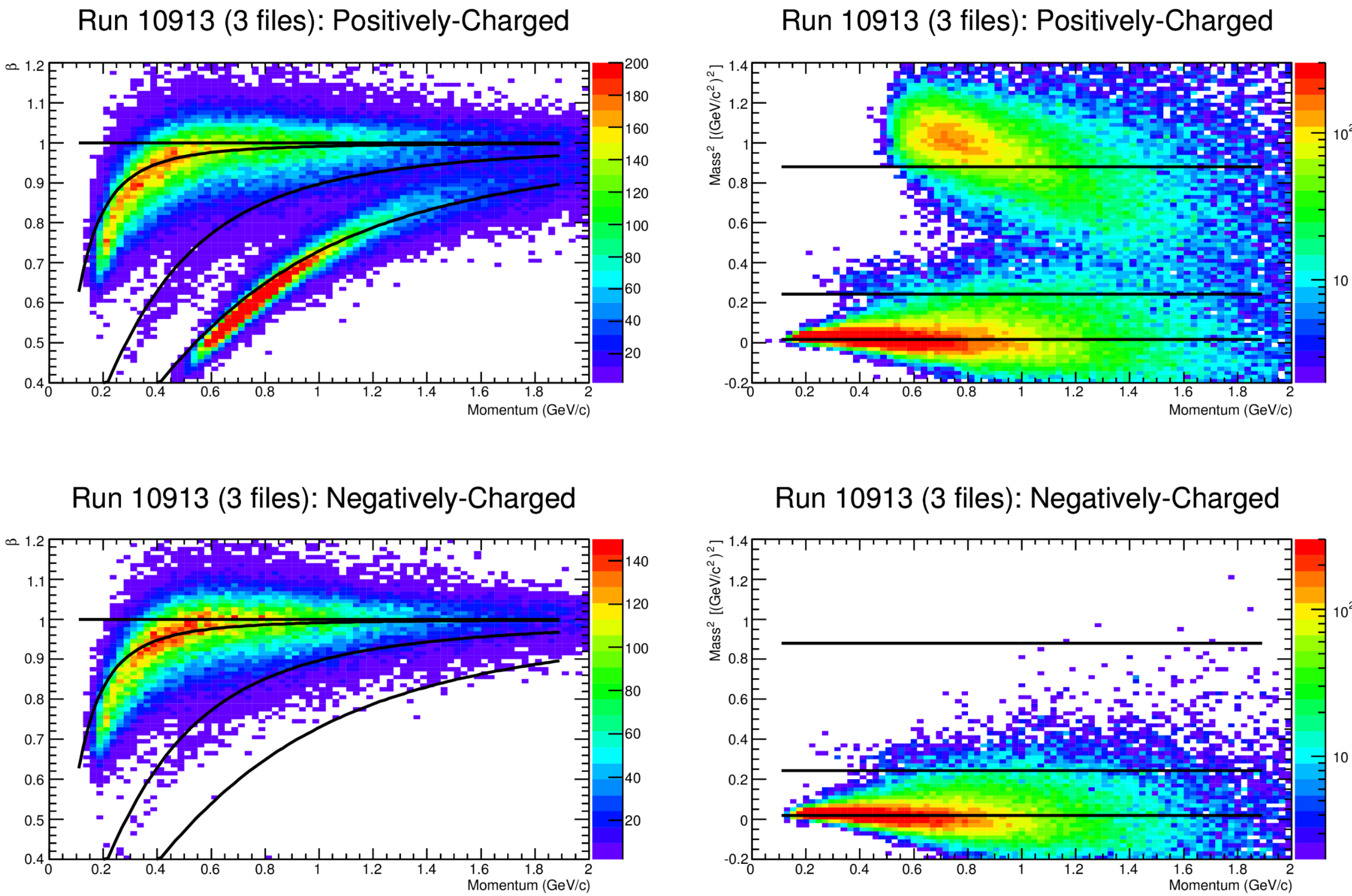
Z-Coordinate: Andrei's vs George's





No “scattering points” => both Andrei's and George's z-coordinates are accurate,
but George's z-coordinates are shifted sistematically by about 10 cm to the higher values
(for the proof, see the previous slide)

Charged Particles: β and Mass² Distributions



The calibration was done with the pions, and the pions are pretty good, but the protons and kaons have some problems (shifted positions and the momentum slope).

Possible explanation: β- and mass²-calculations were done using the momentum from the chambers, but the charged particle drops its momentum during the passage through BCAL material => Corrections in the PID are needed (not for now but for the future high-level corrections.)

Required Changes: Data Base

We need to use **new calibration table of 7 constants** for each readout cell (6 constants for the calibration function + 1 constant for the effective velocity)

Also, we need to read this table in the code.

Required Changes: DBCALUnifiedHit_factory.cc

We need to replace the line 00211 in the existing code:

```
t_TDC -= tdc_coeff.c0 + tdc_coeff.c1/pow(pulse_peak/tdc_coeff.a_thresh, tdc_coeff.c2);
```

With the following formula:

```
t_TDC -= tdc_coeff.c1 + tdc_coeff.c2*pow((pulse_peak-tdc_coeff.c0),tdc_coeff.c3)
                      + tdc_coeff.c4*pow((pulse_peak-tdc_coeff.c0),tdc_coeff.c5);
```

Required Changes: DBCALPoint_factory.cc

We need to replace the line 00164 in the existing code:

```
double zLocal = 0.5 * cEff * ( uphit->t - dnhit->t );
```

With the following formula:

double zlocal =
((uphit->t – dnhit->t)*cEff_Up*cEff_Dn+0.5*fiblen*(cEff_Up-cEff_Dn))/(cEff_Up+cff_Dn);

Also cEff_Up, cEff_Dn, and fiblen should be correctly introduced (NOTE that the effective velocities are the 7th calibration coefficients in the data base table).

Required Changes: DBCALPoint.cc

We need to replace the line 00085 in the existing code:

```
m_t = 0.5 * ( tUp + tDown - fibLen / c_effective );
```

With the following formula:

m_t =0.5*(tUp+tDown - (0.5*fiblen-m_zlocal)/cEff_Dn – (0.5*fiblen+m_zlocal)/cEff_Up);

Also cEff_Up and cEff_Dn should be correctly introduced (NOTE that the effective velocities are the 7th calibration coefficients in the data base table).