Tristan
Blackburn

University of Sussex

CHARGE DEPOSITIONS IN THE APA GAPS - #8

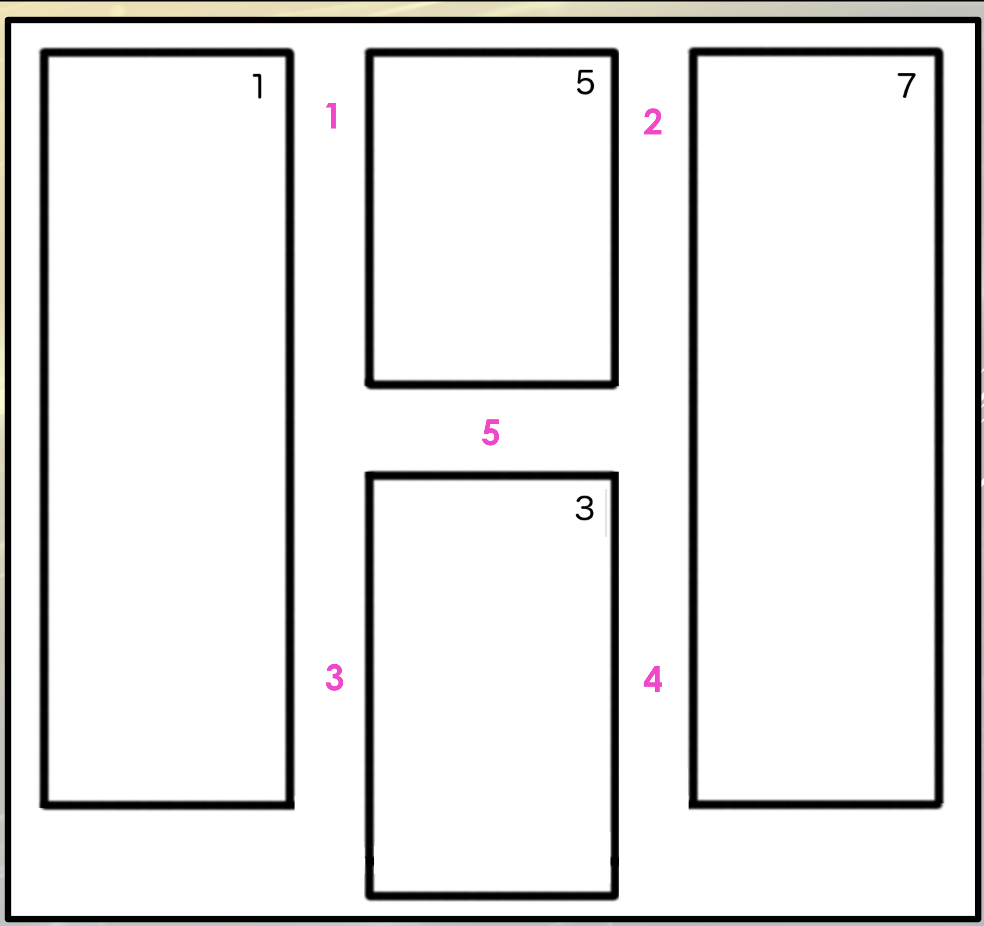
APPROACHING FINALIZATION

IN THIS PRESENTATION

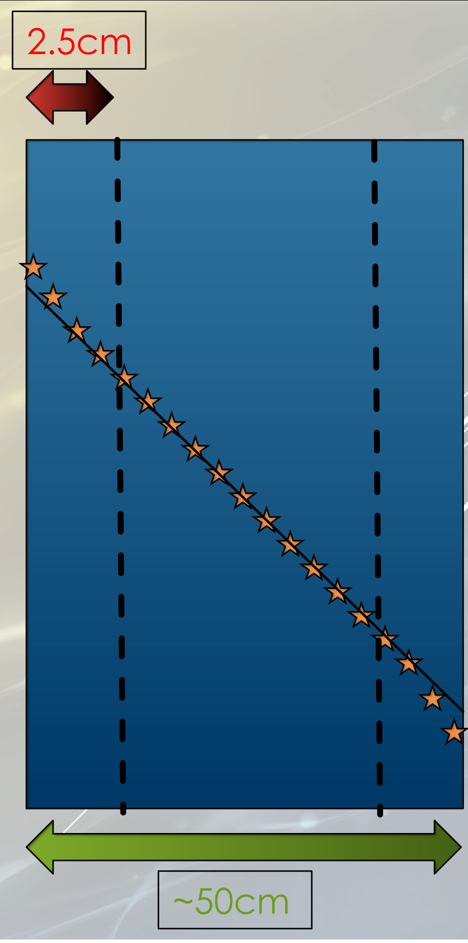
- Detailing of the gap width algorithm migration from standard track objects to a newly written ‘cheap’ track that is done internally by the analysis module.
- Results of module on idealized and MCC4 samples.
- Plots of 3D reconstructed tracks - detailing of new tools available in the module.
- Details of cuts employed to maximize track-space point co-linearity.
- Possible improvements, readiness for real data.

GAP NOMENCLATURE REMINDER

- 35-t APA gaps labeled according to which TPC's they bridge.
- The pink numbers indicate the gap label.
- The black numbers indicate the TPC label.
- View is of LDV as if standing in the SDV



- The edges of unstitched tracks have small distortions at the end arising from hits, that extend beyond the TPC edge, being associatively projected onto a track.
- Causes the unstitched track segments to be subject to gradient fluctuations which in turn causes the minimization function in the gap width module to lose accuracy.
- Have now migrated the code to use the space points associated with a track. Space points subject to edge effects can be effectively moved by making cuts in either track length or Z.



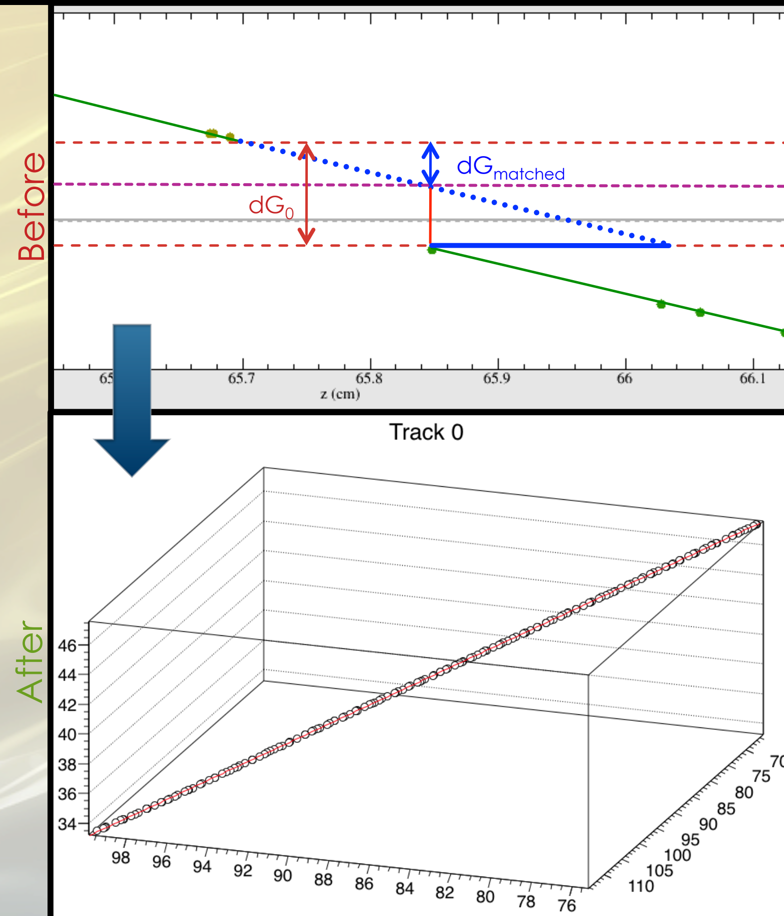
TRACK EDGE DISTORTIONS

The top right picture shows, in green, the ortho3D visualization of the previously used track objects.

dG0 is the minimum possible gap width given track end/start points and dGmatched is the calculated gap width. The other lines are there for ease of understanding the extrapolation procedure.

The bottom plot shows the new reconstructed track object. One such plot is made per track, per event.

It is produced by applying cuts to the space points, which remove edge distortions, and then putting the resultant XYZ points into a TGraph2D. Then, a tailored fitting algorithm, that invokes MIGRAD, is used to create the 3D line fit.



NEW TRACK OBJECTS

CUT CRITERIA

- The TGraph2D track segments are 3D, as opposed to the two 2D projections employed before. This allows better filtering of analyzable events.
- Tracks that are short in one 2D view, and would become useless given certain 1D cuts, can now pass selection and still provide gap width measurements
- For an event to be considered suitable it must pass the following (adjustable) cuts:

  1: Chi-Squared from the fit
  2: Minimum track length for matched tracks (cm)
  3: Gradient alignment checks (done by conversion from grad to tan space)

- To remove edge distortions, only the space point contributions from either:

  1: Hits > 2.5cm from the TPC edge; or
  2: 20% away from the 3D TGraph2D track edges

  are used for the fits. Whichever cut removes the least space points, whilst causing the Chi-Squared to fall into an 'acceptance' range, is employed. This way the total analyzable sample is maximized.

No of sppts before cut	No of sppts after 5cm cut	No of sppts after 2.5cm cut
208	138	169
13	0	13
293	293	293
180	111	141

Table showing the number of space points that are lost after 5cm and 2.5cm cuts from the TPC edges.

OUTPUT

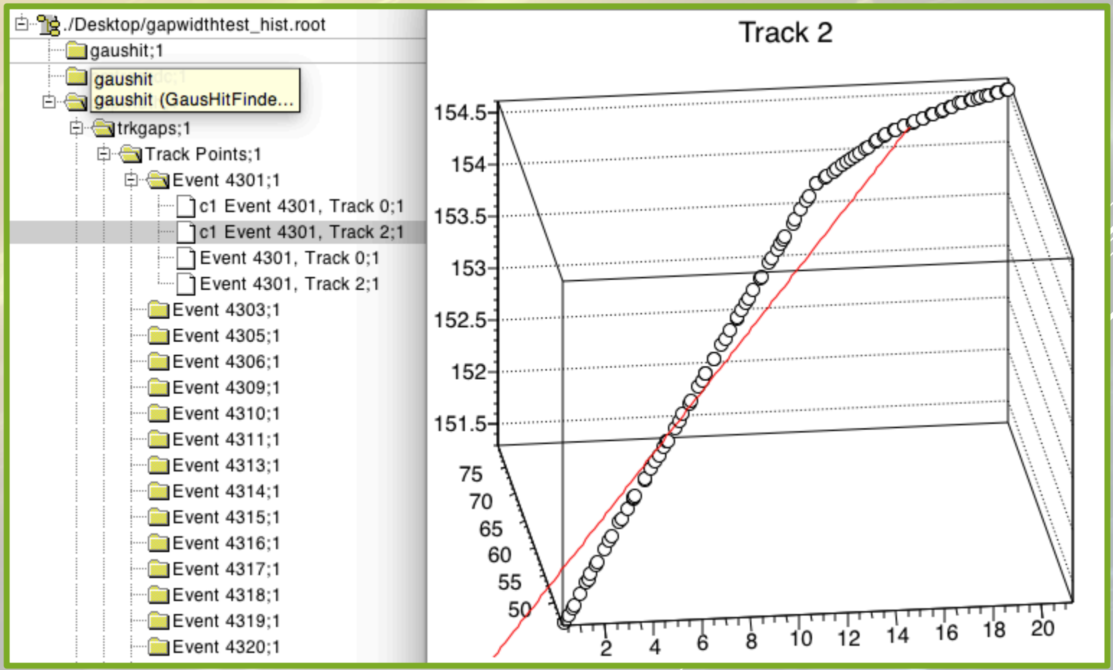
The Gap Width Module produces folders containing unique per track, per event plots and one large TTree.

Plots are created of every set of space points associated with every track for every event.

Further canvases are produced, for all events that pass minimum fitting criteria, containing both the space points and the subsequent fits of the new track segments.

A further two TH1D plots are made for each event, for each possible track matching. The first contains the calculated gap width against the translation in the second track. The second contains the difference between the calculated value of the gap width and the truth gap width – as known from the V5 geo.

The TTree contains (amongst many other track properties) five branches, one for each gap, that hold all the resulting calculated values for each gap from each event the module ran over.

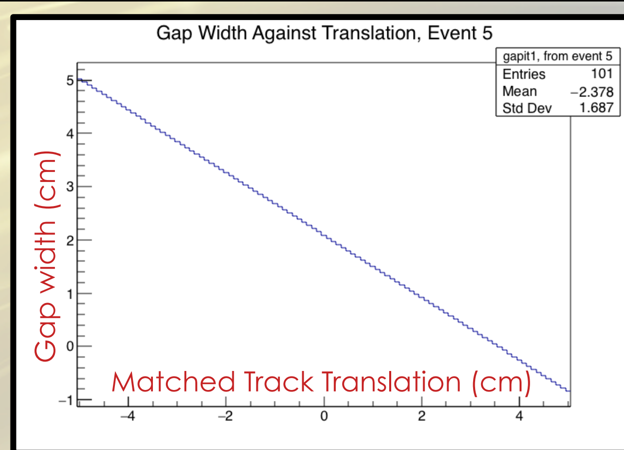


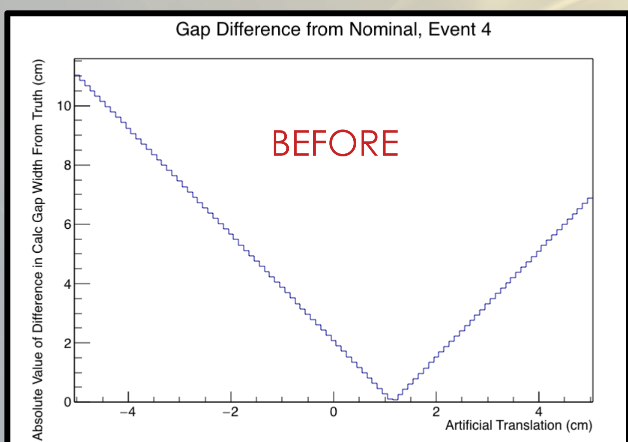
TRACK MATCH RESULTS

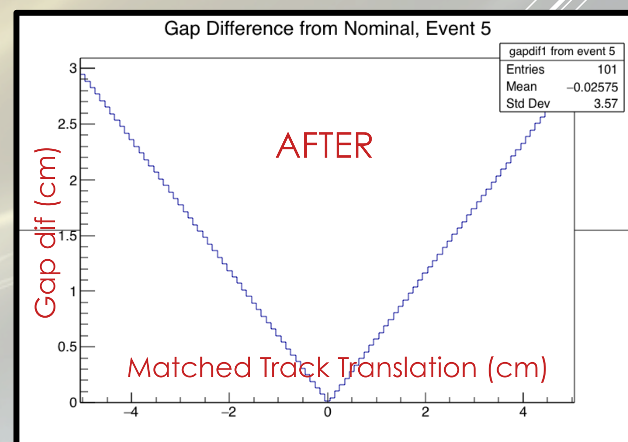
Output of the calculated gap width (cm) against the matched track artificial translation is shown in the top plot.

Output of the calculated gap width difference, with respect to truth, (cm) against the matched track artificial translation is shown in the bottom two plots.

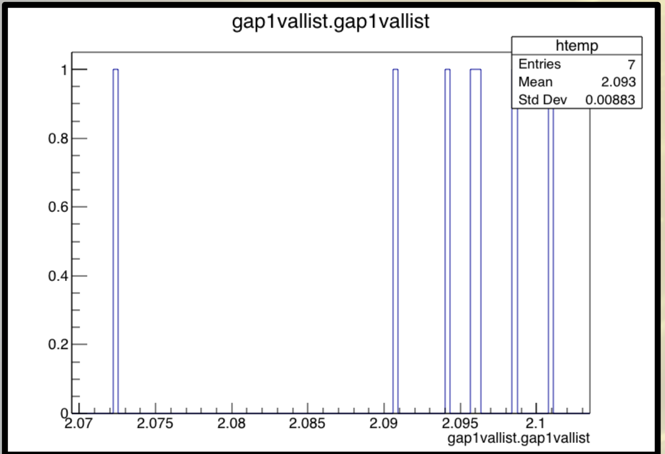
The before and after in the two bottom plots are for when using the previous objects (the unstitched tracks from PMalg) and when using the new track objects respectively.

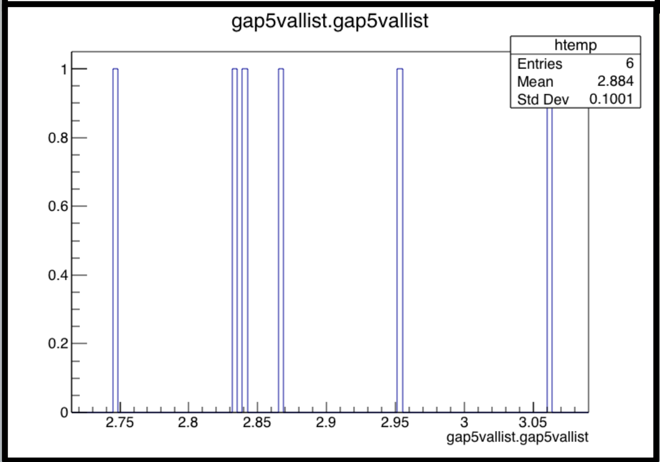






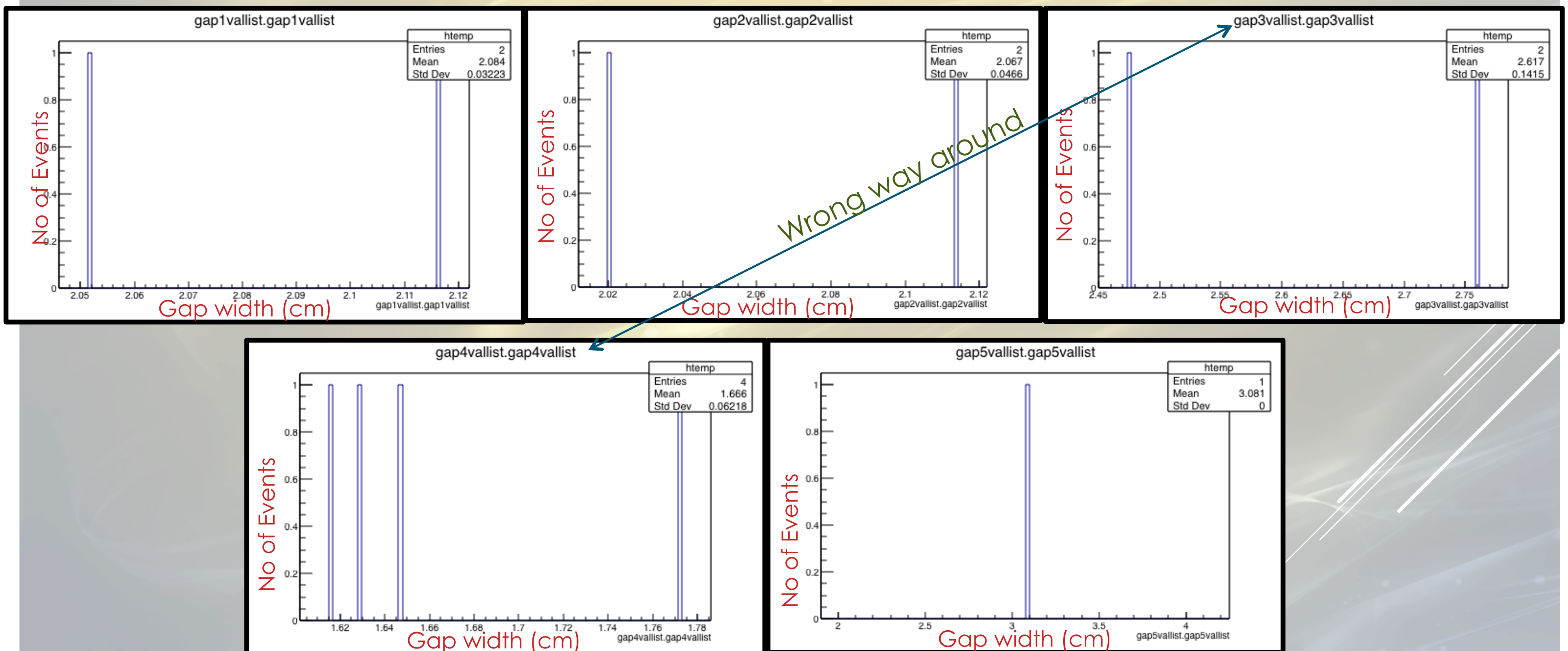
RESULTS OF GAP WIDTH MODULE OVER 10 EVENT COUNTER HITTING SAMPLE





In a sample of 10 anti-muon events, both of which are known from truth (and/or the gap filtering module), to cross gaps 1 & 5, 7 events produce a 'good' gap width calculation for gap 1 and 6 events do the same for gap 5

Gap	Truth Gap Size (cm)	Calc Gap Size (cm) [no of events]
1	2.0789	2.093 [7]
2	2.0789	n/a
3	1.6299	n/a
4	2.5280	n/a
5	2.9241	2.884 [6]

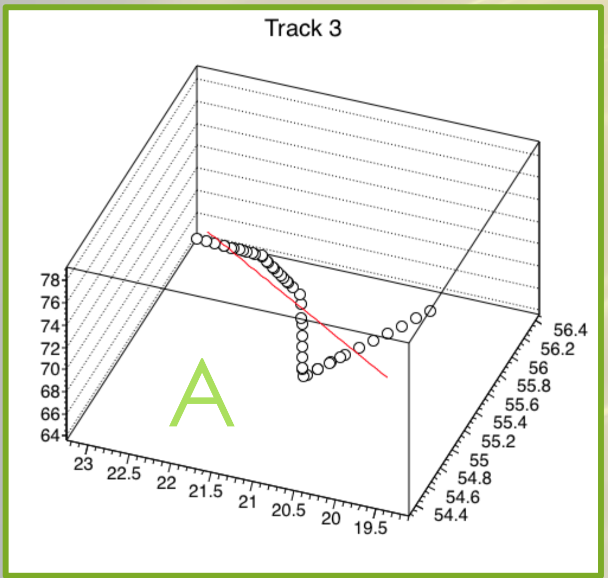


RESULTS OF GAP WIDTH MODULE OVER 100 MCC4 EVENTS

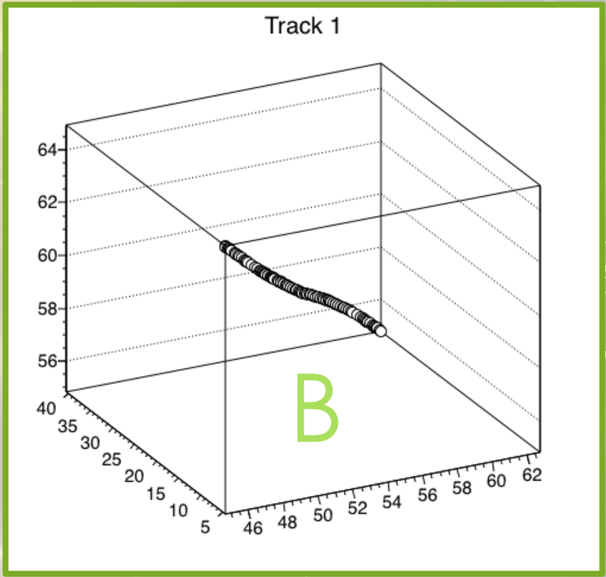
Gap	Truth Gap Size (cm)	Calc Gap Size (cm) [no of events]
1	2.0789	2.084 [2]
2	2.0789	2.067 [2]
3	1.6299	1.666 [4]
4	2.5280	2.617 [2]
5	2.9241	3.081 [1]

RESULTS OF GAP WIDTH MODULE OVER 100 MCC4 EVENTS

EXAMPLES OF 'FAILURE' TRACKS



Track A is an example of a collection of space points, which pass the requirement to fit (no hits close to edge and sufficiently large number of points to fit to) but subsequently fails the gradient matching criteria. No Chi-Squared cut is applied.

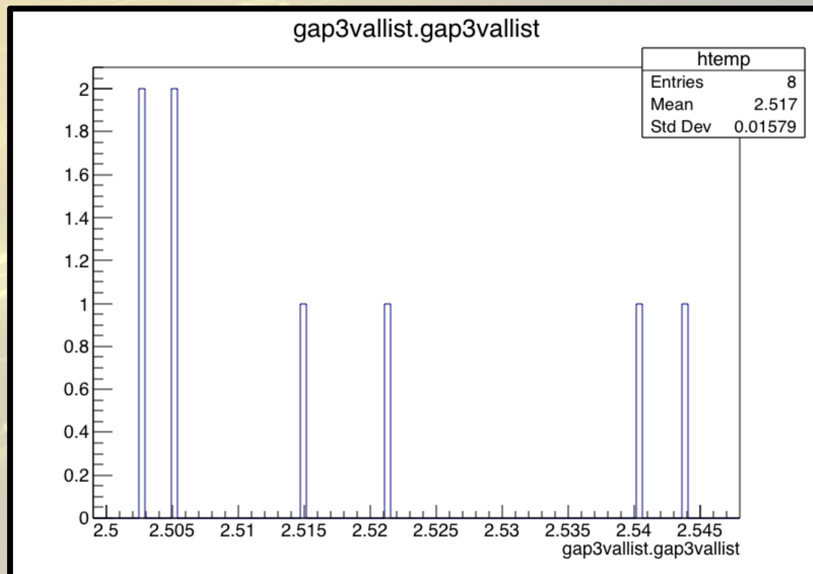


Track B is an example of a track that is sufficiently non linear for a long enough period such that it fails the Chi-Squared cut and has no fit made to it.

When comparing A and B notice the dimensions of the TGraph2D. Track A may look less clean but the distance scale is much smaller.

FAILURES OF THE MODULE

The module occasionally calculates values of gap width for non gap crossing events. The plot to the right is taken from the 10 anti-muon sample, in which 8 values are calculated for Gap 3. No event crosses gap 3 in this sample. This happens because track end and start points can be very close if a track 'clips' the corner of TPC5 in it's trajectory from TPC 1/7 to TPC3. In this case, it has crossed gap 1, not 3.



This can be fixed by first running the gap filtering module, and passing the output to the gap calculating module. A FHICL parameter to do this is on its way!

The MINUIT & MIGRAD minimizations employed by the width calculation and by the fitter are not 100% optimized. This is harder to 'fix' but just requires time, trial and error.

Errors on the result are very hard to determine. Going to produce a statistical estimate on error after running over a sufficiently large data set.

- Implement FHICL parameters for all cuts and selection criteria. These should all be adjustable to user preference/use case.
- Pass information from gap filtering module to gap calculating module to bypass error described on slide 13.
- Need errors in calculation – going to look at the S.D and distribution of values about the truth value for a sample that provides a large number of calculated widths.
- Optimize the TMinuit functionality for the code.
- Suggestions?

THINGS TO TIDY UP

MEASURING CHARGE LOSS

- Once a gap width has been determined it is straightforward to evaluate the charge lost using the shown equation.
- This equation only works for the collection planes.
- The equation assumes a linear relationship between gap distance and charge lost.

$$Q_{edge} = \frac{Q_{wire}}{dx_{wire}} dx_{gap} \times (1 - Q_{lost})$$

Where:

Q_{edge} = Charge on edge channel
Q_{wire} = Average charge on wire in same TPC
dx_{wire} = Distance of LAr experienced by a channel
dx_{gap} = Distance of LAr experienced by a gap adjacent edge channel
Q_{lost} = Fractional charge lost to the detector

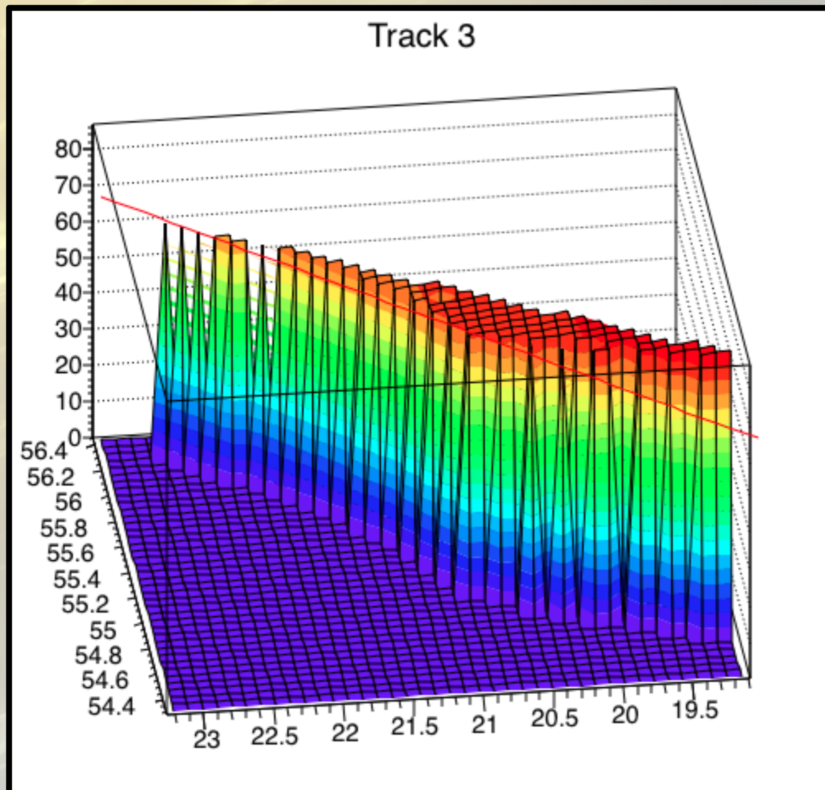
TGraph2Ds are useful for creating visualizations of associated values that vary across a series of associated hits.

Can use the module to produce heat maps of hit charge distribution, hit peak time etc in 3D.

The plot to the right shows a series of space points, to which a track has been fitted, weighted by the associated position in Z. The weighting (which is reflected in color) can be changed to any applicable 'per hit' variable.

Requests for any such plots?

SCOPE FOR FUN PLOTS!



CONCLUSIONS

- The module has been updated to minimize the distortions that negatively influence gap width results.
- This has been done through implementation of a TFitter that fits to the space points associated with a track. The space points themselves are where the cuts are put in place.
- Calculated gap widths are now within 1% of truth values, when 10 or more determinations are made. Before the adjustments, calculated gap widths were within 5-10% of the truth value.
- There's a few things left to tidy with regard to code and implementing FCL pars.
- Pretty much ready for data!
- There's plenty of room for interesting plots to be made with the module given the new track objects.
- The module output is large. 100 events, of which only ~15% cross gaps, requires 6MB of disk space. This can be reduced by reducing the number of output plots. Again the output verbosity could be adjusted with a FHICL par.