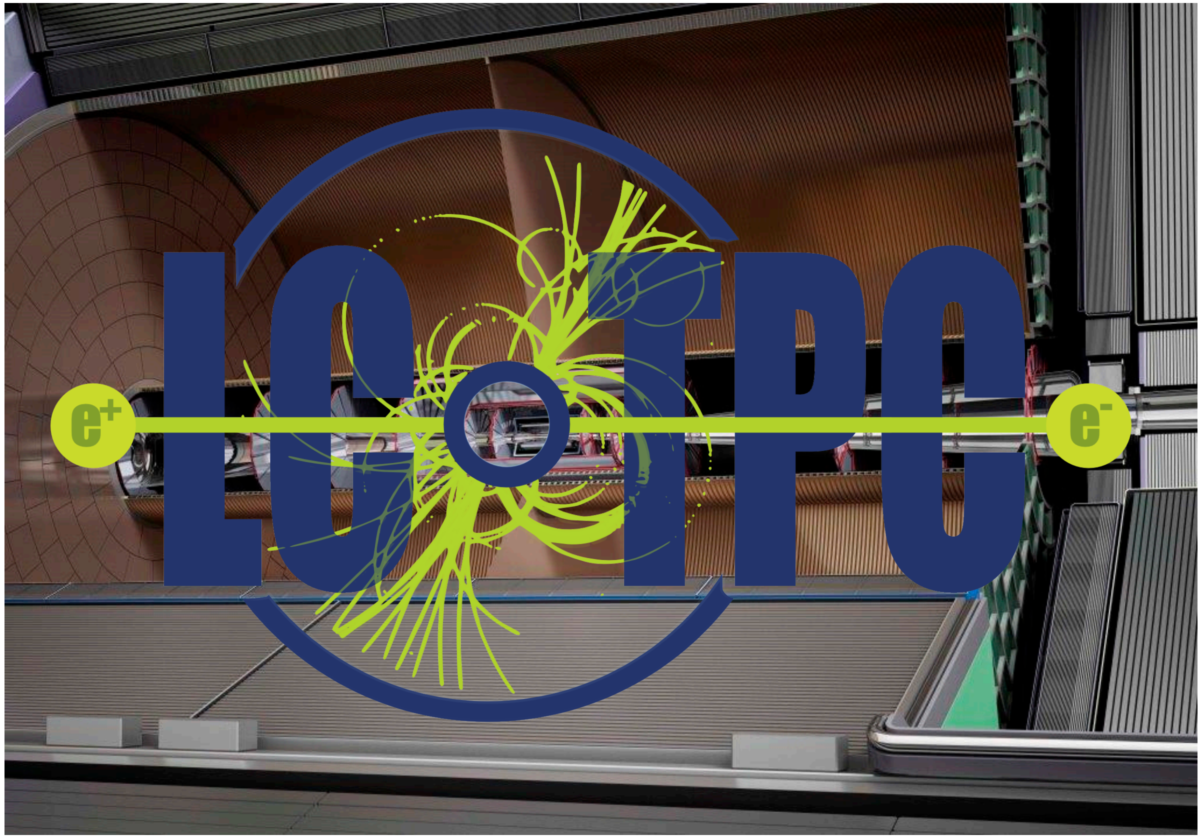
Pixel TPC dEdx

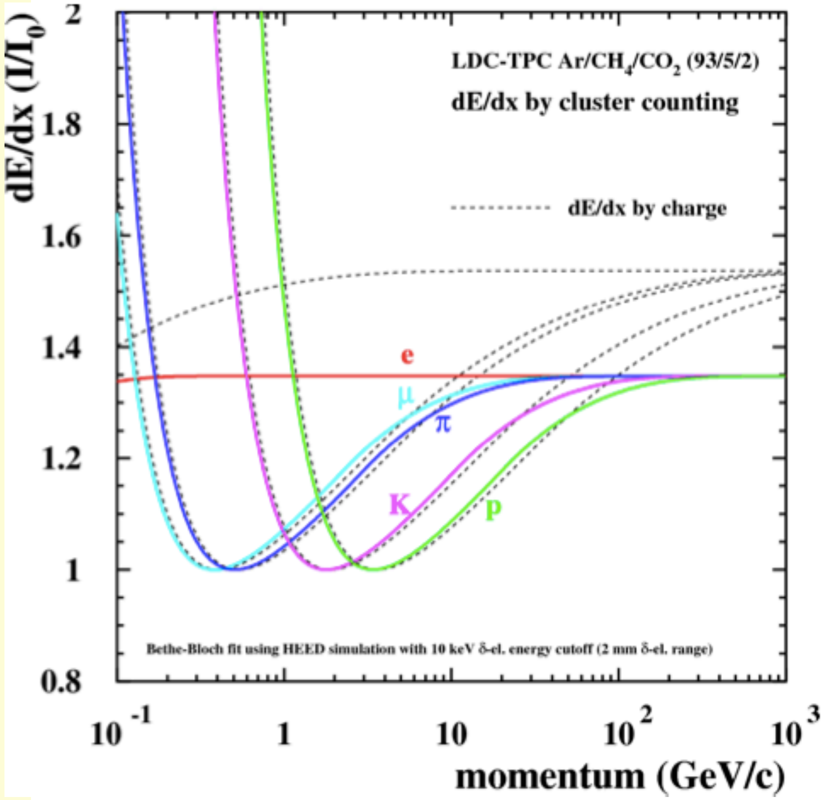


Fred Hartjes, Kees Ligtenberg, Jan Timmermans

Peter Kluit - Nikhef 20 november 2017

dEdx by charge and clusters

Energy loss can be measured from the charge e.g. using a pad read out scheme. However a pixel TPC allows an identification of the primary clusters as well as the charge from the total number of electrons (pixels).



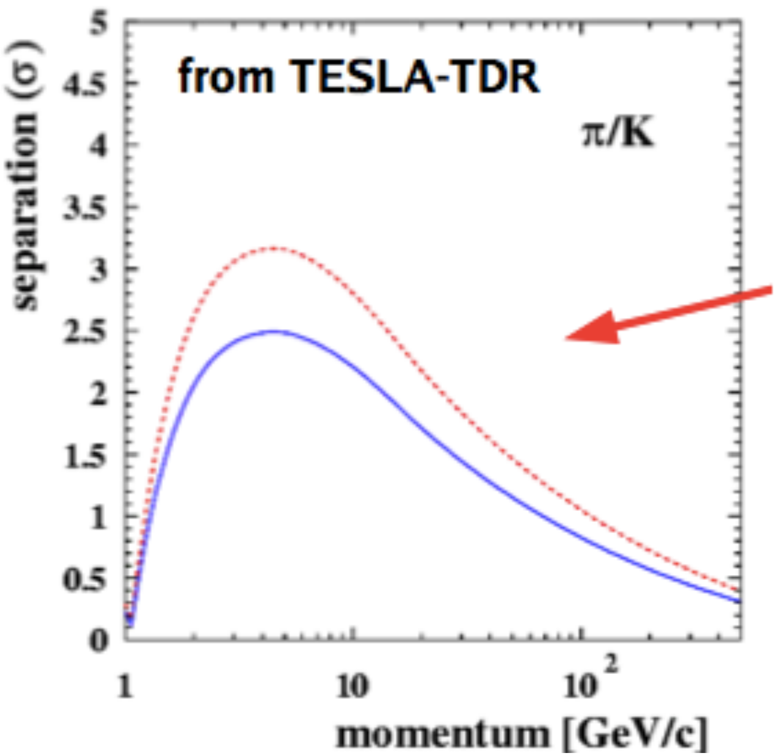
- Cluster counting
  a Poissonian distribution;
- Charge measurement
  a Landau like distribution with a long tail from secondary electrons and deltas.

Michael Hausschild ILC Tracking 2006 has shown that cluster counting can significantly improve the dEdx separation up to a factor 2.
Note that the mean measured dEdx is different for both methods. And depends on gas.

dEdx by charge and clusters

Comparison of the two methods is not trivial.
The σ dEdx is e.g. not a good measure: this e.g. a lot smaller for cluster counting than for the charge.
The final measure is the pion-Kaon separation in σ:

separation = (dEdx(π)-dEdx(K))/σ(π)

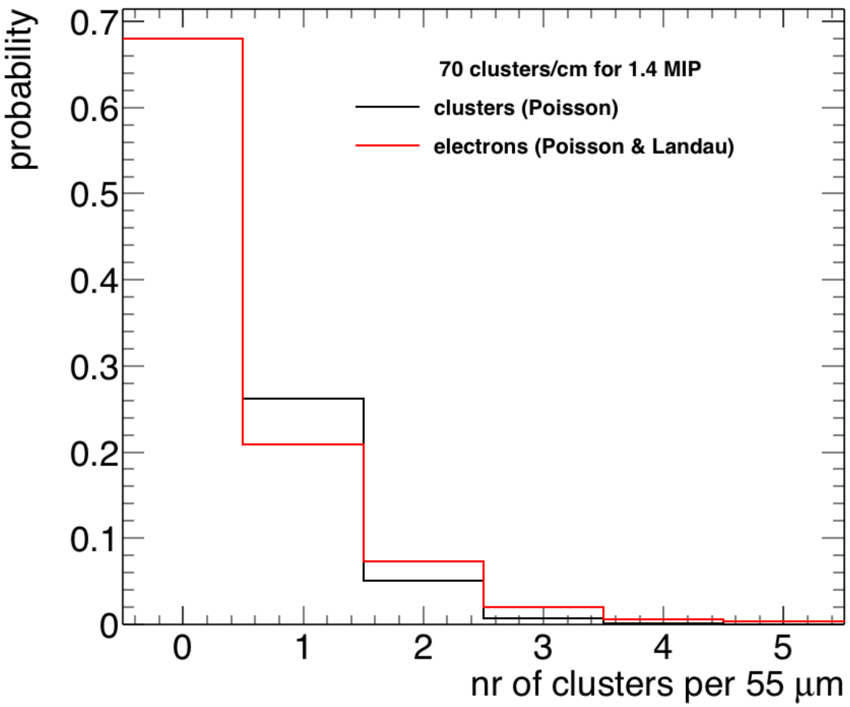


Method proposed: use the response of electrons and compare it to a 70% efficient electron (by hit dropping). This can be done on test beam data and simulation.

One could also think of comparing 4 GeV pions to kaons (as Michael proposed).

dEdx for a GridPix detector

The dEdx also strongly depends on the gas used, the granularity and efficiency of the device.
Further the diffusion in the drift process will smear the primary clusters and affect the capability to isolate in the reconstruction the clusters.
For the TPX3 chip the pixel granularity is 55 μm; in the T2K gas about 70 primary clusters per cm are made for 1.4 MIP.
In the GridPix detector this corresponds to (no diffusion):



So most of the time (68%) no cluster is expected. Clearly the Landau electron signal has a long tail

Test beam data and dEdx

Test beam data will be used to study the dEdx performance (see also talk Kees Ligtenberg in this session)
Data are taken in the Bonn test beam with a single GridPix detector based on one TPX3 chip using electrons of 2.5 GeV.

Tracks were reconstructed using a EUDET silicon beam telescope and the GridPix detector.
Selecting tracks that cross the detector the following procedure was followed:

- The maximum nr of pixels crossed is 256 (x 55 μm).
- The fully efficient region has a size of 220 pixels.
- In that range the hits associated to the track are projected along the track in the pixel plane and binned in 55 micron bins. Associated hits lie with 2 (pixel) and 3 (drift) sigma in both measurement planes.
- Mean resolution in pixel plane 250 μm
- run 347 is taken with 350 V grid voltage

G4 simulation samples

Geant4 was used to simulate different samples.
The ILD TPC pixel simulation was used for this purpose.
(see Kees Ligtenberg LCTPC 11 May 2017)

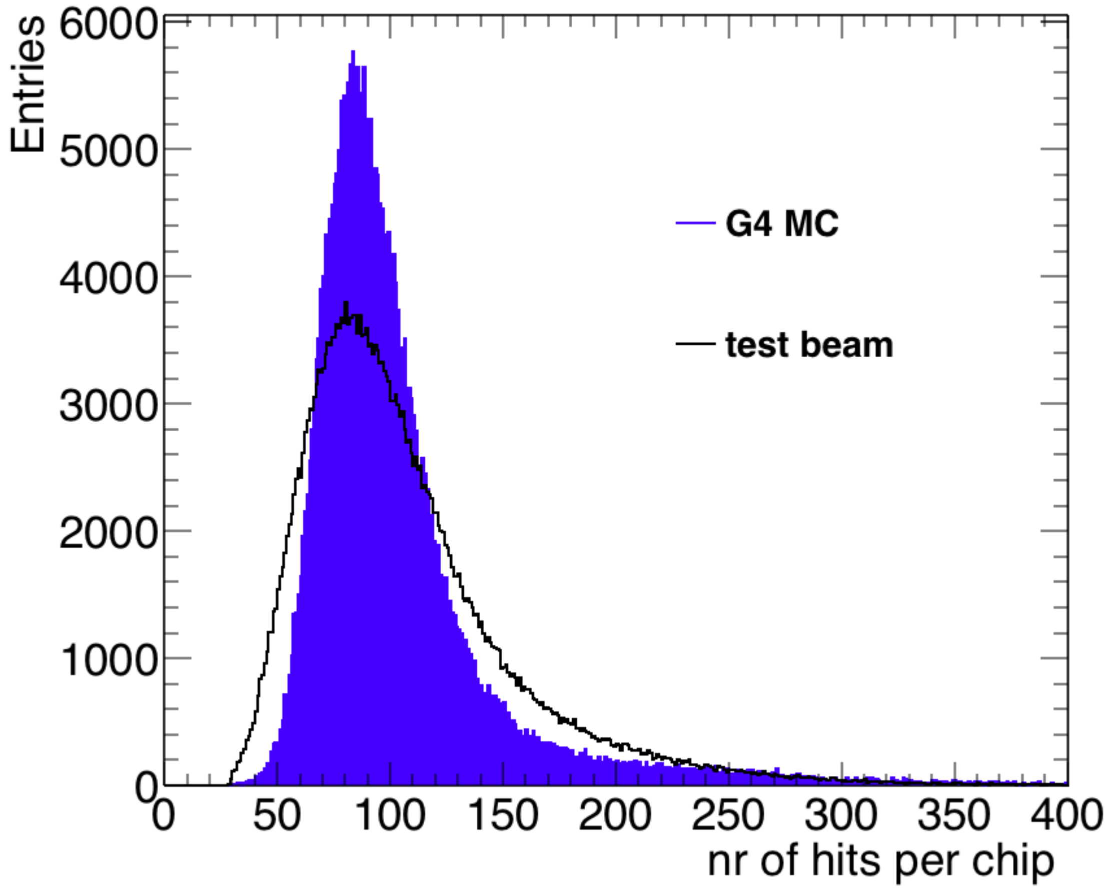
- 2.5 GeV electrons without B field. With a detailed simulation of the ionization with a step length of 55 μm;
- 250 GeV muons and in a 4 T B field; here a larger step length of 18 pixels was used interpolating the Eloss.

- both samples were generated at θ=85 degrees;
- The resolution in xy per hit in the simulation for this drift distance is 350 μm (default ILD diffusion values for B=4 T)

Also samples with the truth hit position without diffusion were made for the two samples.

Test beam data hits

Data run 347 300 k tracks



Below the total nr of hits associated to the track (for one chip).
Compared to the G4 electron sample.

One can observe that the data is more smeared around the MOP (most probable value).

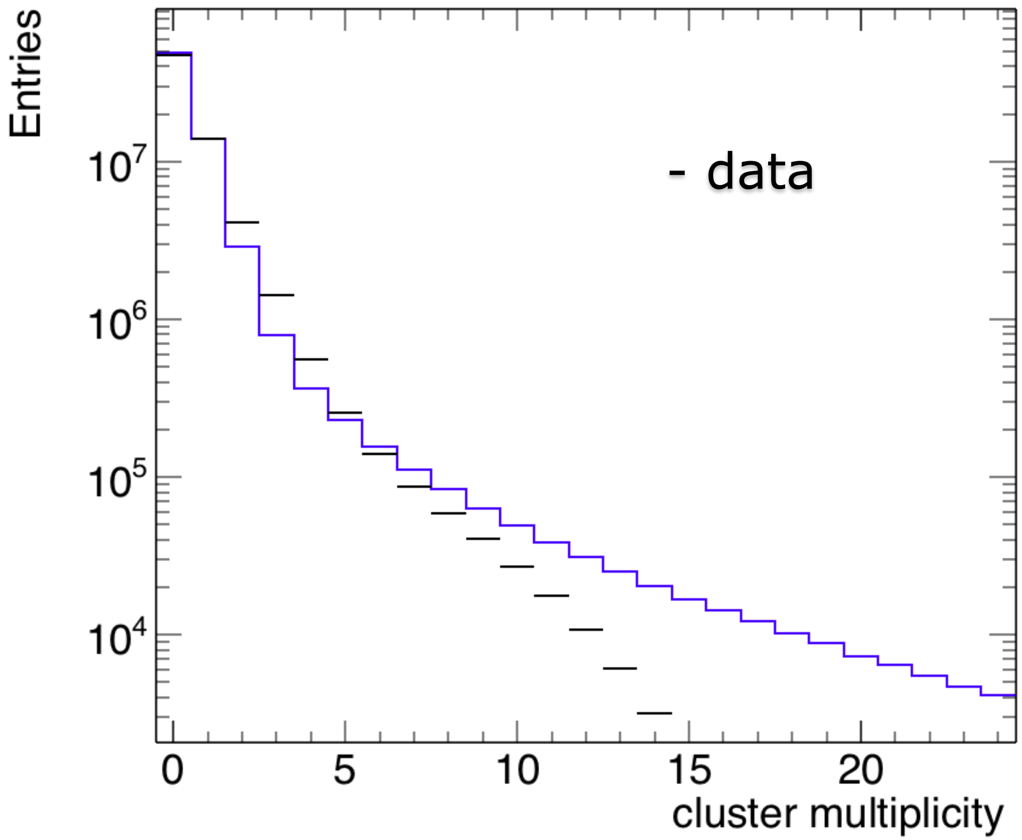
Test beam dEdx variables

The following array of information is used:
All the hits associated to the track are projected along the track into the pixel plane.
A typical event is 220 pixels long (efficient region):
0 1 5 1 0 0 0 2 2 0 1 0 0 1 1 0 6 3 0 etc.

From this one can calculate:
- The multiplicities 0, 1, 5, 1, etc.
- Distance between subsequent hits

More complicated are the cluster variables:
- cluster width and maximum: between two zero's
  1 5 1 -> width = 3 and max = 5
  2 2 -> width = 2 and max = 2

Details: Test beam data multiplicity

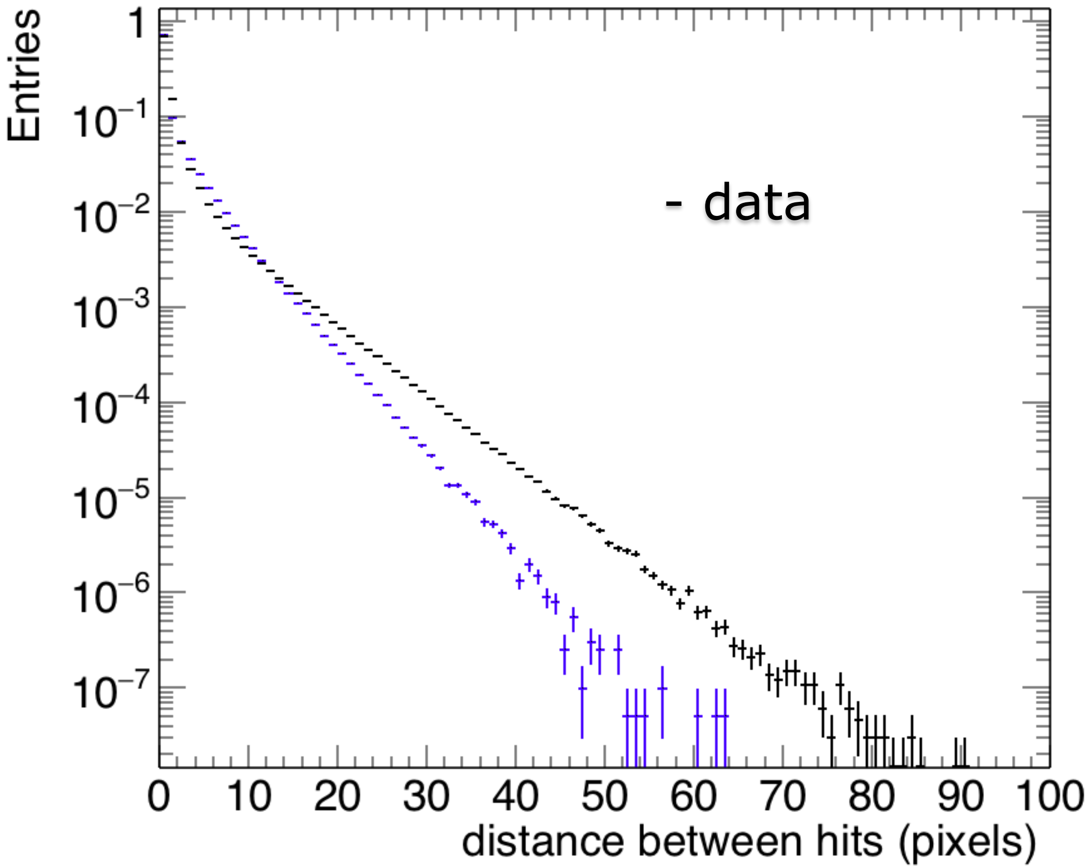


Here the distribution for data and G4 MC for cluster multiplicity (so nr of hits in the particular bin) along the track.

One can observe that in simulation there are more events with large (>5) multiplicities.

Data has more clusters 2-4

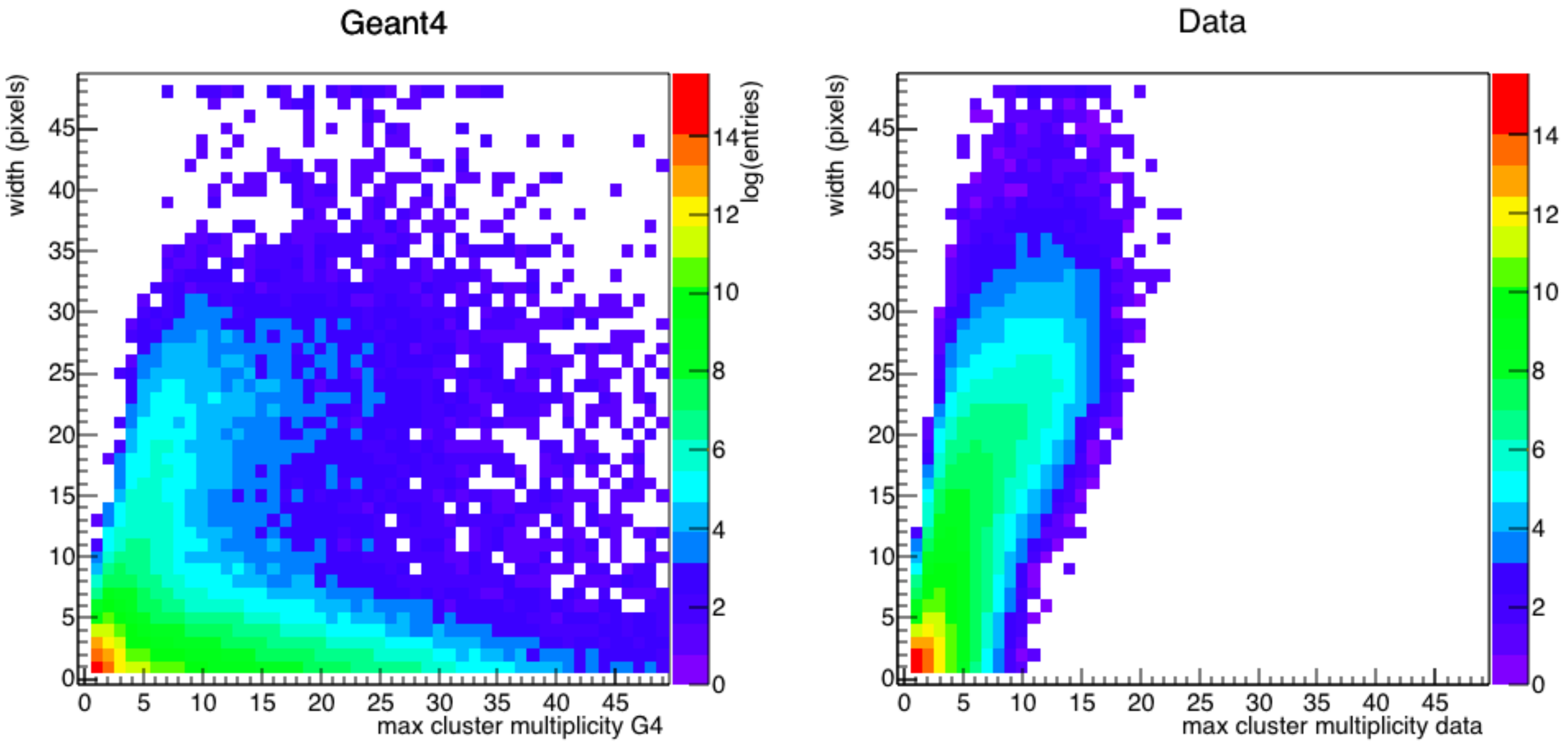
Details: Test beam data distance



Here the distribution for data and G4 MC for the distance between subsequent hits along the track in the pixel plane.

One can observe that in the data there are more events with large distances.

Details: Test beam data cluster width



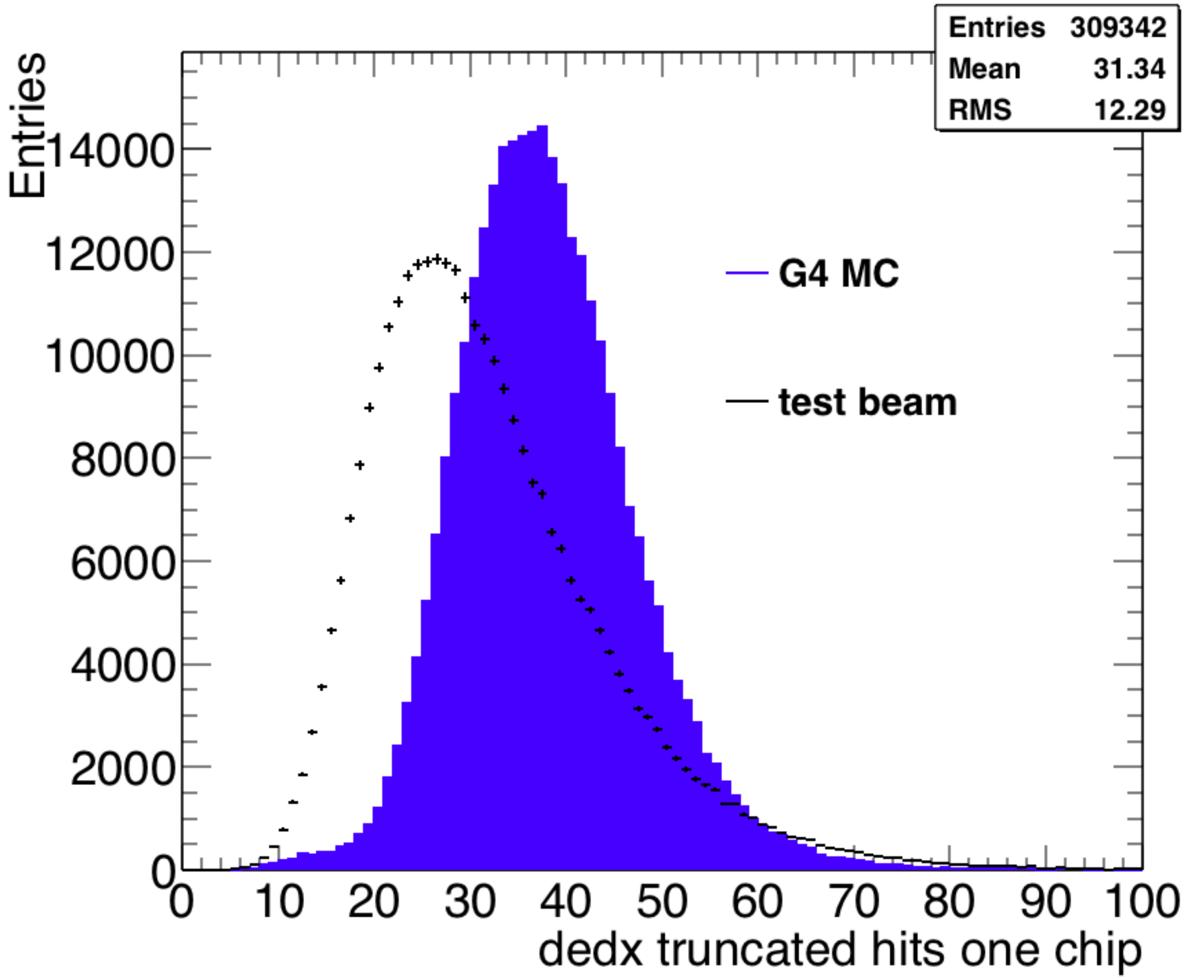
Here along the tracks isolated clusters are looked for and there width (how long along the track) vs max nr of hits in one bin. One can observe that the distribution in simulation has higher max cluster multiplicities.

Test beam dEdx algorithms

"Truncated hits" algorithm to calculate the total hits (electrons) truncated. Use the method proposed by Michael Lupberger to make 20 pixel samples. So 11 samples per track. Then only the 70 percent lowest samples are used to measure the total nr of hits.

The full performance of a Pixel TPC is obtained by combining 8 (rows) times 9 (chips per module).

Test beam dEdx truncated hits



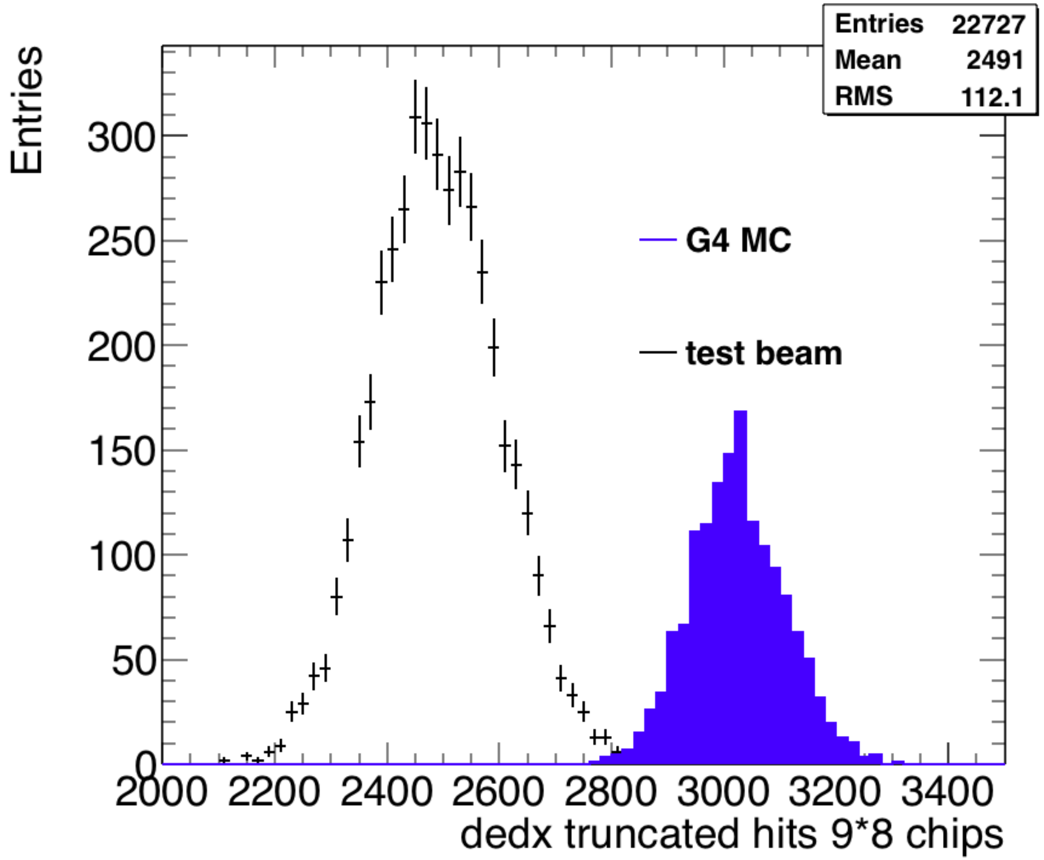
Truncated nr of hits for data and G4 for one chip.

The distribution in data has more spread and a lower average.

To get rid of the Landau tail one has to add more samples

NB this corresponds to 1.1 cm "pad"

Test beam dEdx truncated hits



Truncated mean nr of hits for data for 72 chips.

Performances dEdx:

	mean	rms	reso (%)
data	2491	112	4.5
G4	3023	83	2.7

NB track length 72*220*55 μm
or 87 cm
while full TPC 133 cm

Took a realistic (pessimistic) coverage including dead zones for the Pixel TPC.

Test beam dEdx algorithms

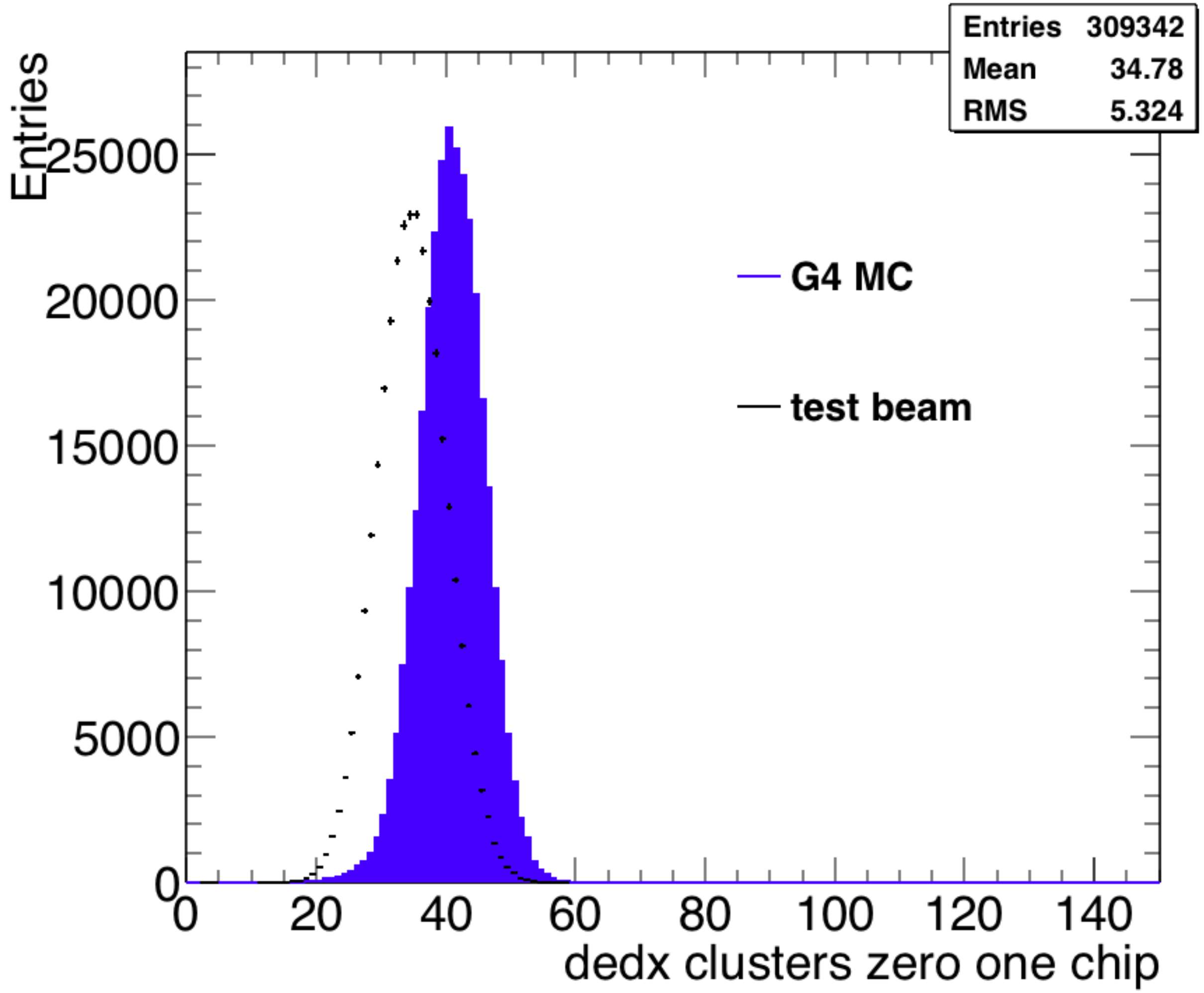
"Clusters zero" algorithm
method counts all hits between two zero's as 1

So the typical event:
0 1 5 1 0 0 0 2 2 0 1 0 0 1 1 0 6 3 0 etc.
1 1 1 1 1
So in total 5 clusters.

"Clusters 1" algorithm
method counts all hits as 1 cluster
0 1 5 1 0 0 0 2 2 0 1 0 0 1 1 0 6 3 0 etc.
1 1 1 1 1 1 1 1 1 1

One can think of smarter algorithms that looks at the hits between zero's and then fits a peak and assigns a cluster count depending on the cluster width and peak height.

Test beam dEdx clusters "zero"

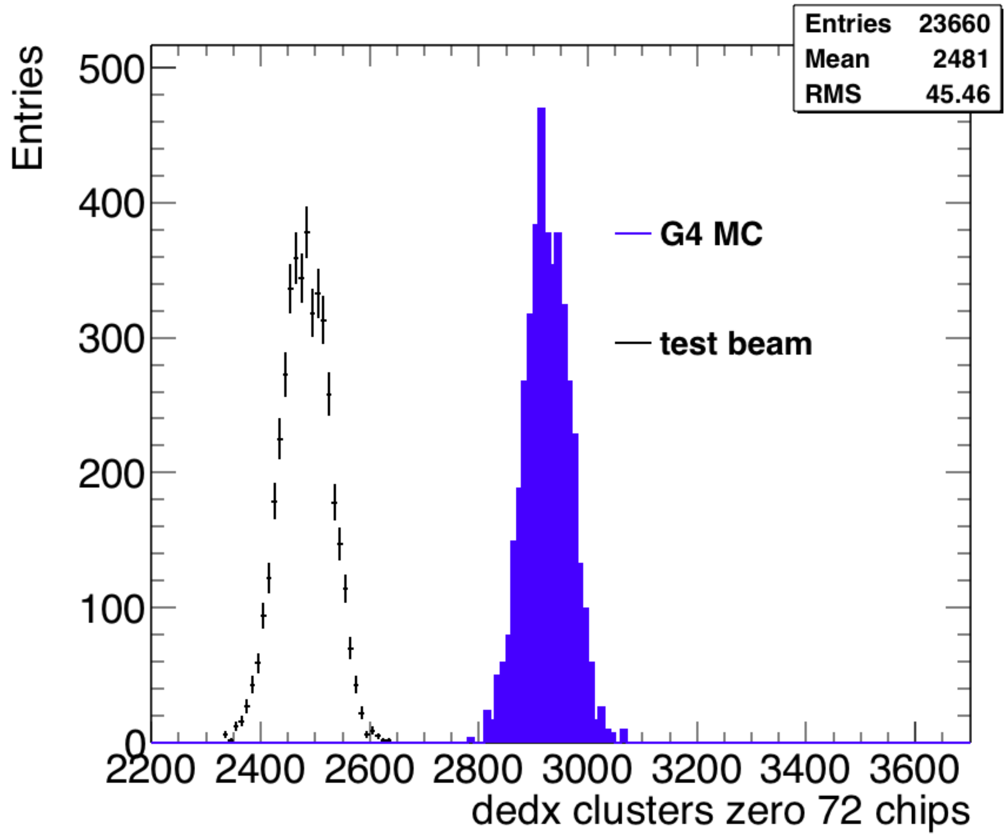


Cluster zero hits for data and G4 for one chip.

The distribution has clearly less Landau tail!

And also a quite small rms.

Test beam dEdx clusters "zero"



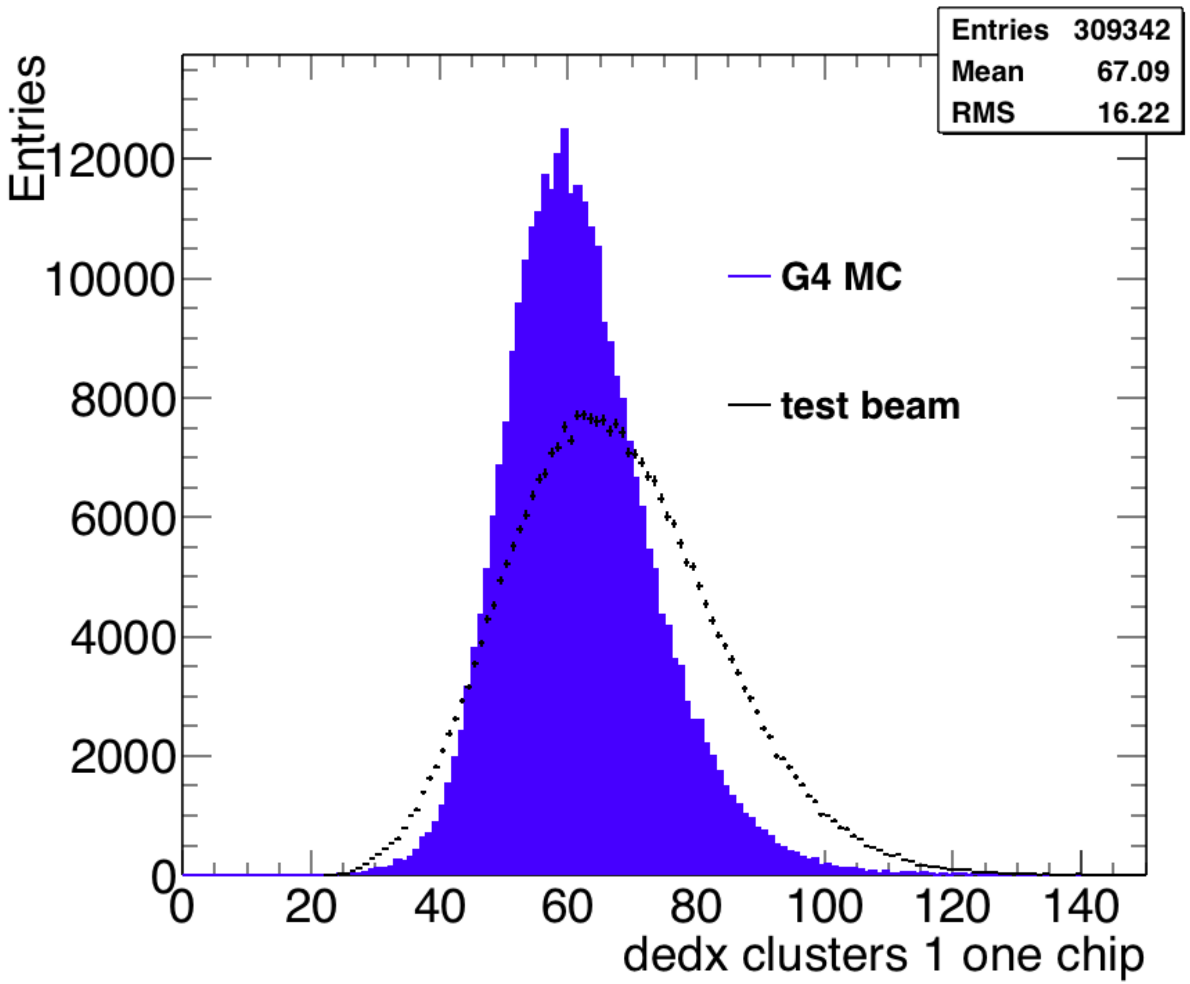
Cluster zero hits for data and G4 for 72 chips.

Performances dEdx:

	mean	rms	reso (%)
data	2481	45	1.8
G4	2924	41	1.4

NB track length 72*220*55 μm
or 87 cm
while full TPC 133 cm

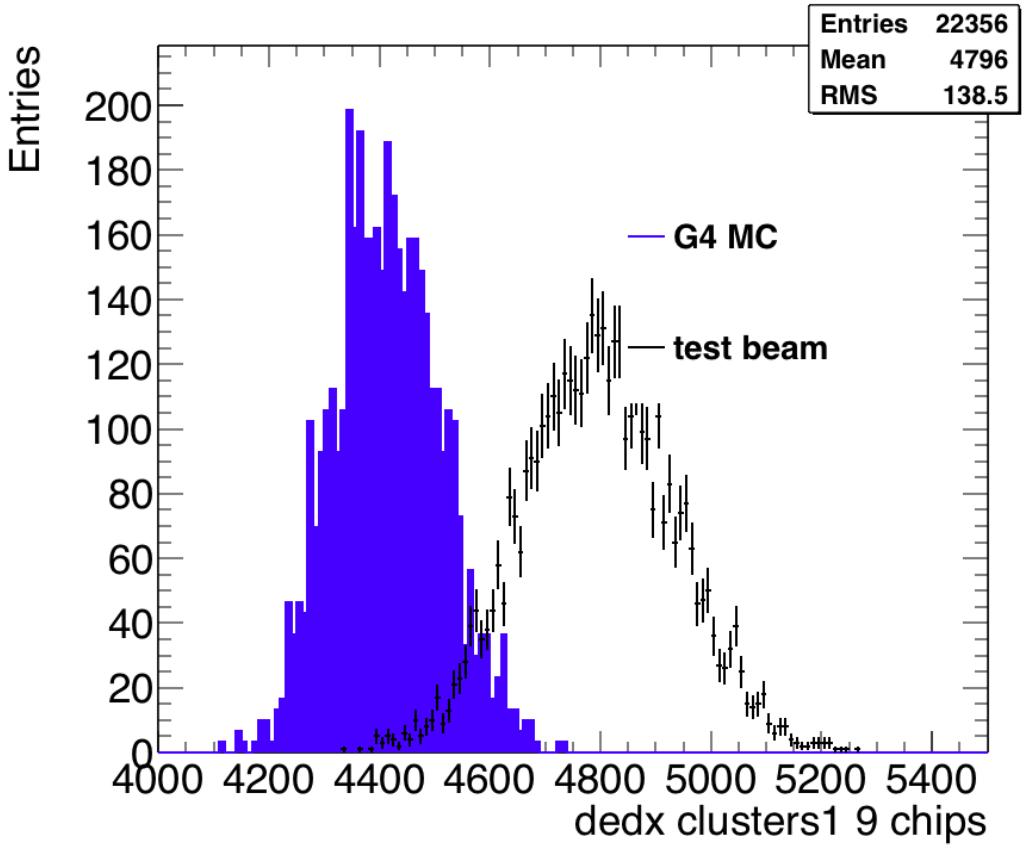
Test beam dEdx clusters "1"



Cluster "1" hits
for data and G4
for one chip.

The distribution
has more Landau
tail than "zero"
but less than
the truncated hits

Test beam dEdx clusters "1"



Cluster 1 hits for data and G4 for 72 chips.

Performances dEdx:

	mean	rms	reso (%)
data	4796	138	4.4
G4	4414	95	2.7

NB track length 72*220*55 μm
or 87 cm
while full TPC 133 cm

Separation: algorithm performance

As I stated in the beginning of the talk. The dEdx resolution is not the whole story. The cluster “zero” algorithm has a high resolution. But is it really the best performant?

The way to address this question is by measuring the separation. This can be done by using the data and the simulation and dropping hits with an efficiency of 70%.

This case corresponds to something like e/K separation at 2 GeV.

So before concluding anything on the dEdx performance this study has to be performed. Results will be presented on the next slides.

Separation: dEdx results I

separation = (dEdx(e)-dEdx(0.7*e))/σ(e)

Data results

algorithm	difference	error	separation
Trunc hits	786.	111.	7.0
clusterz	283.	45.	6.2
cluster1	1063.	138.	7.6

G4 MC results

Trunc hits	966.	83.	11.5
clusterz	472.	40.	11.5
cluster1	1044.	95.	10.9

NB a 5% σ(dEdx) for electrons corresponds to 6.0 σ

So the difference between the three algorithms is not that large. The simulation is a factor 1.6 too optimistic.

Can we do better? E.g. combining cluster"z" and "1"?

Separation: dEdx results II

Made simple linear combinations and optimized;

clusterc1 = clusterz + 0.5 cluster1

clusterc2 = (clusterc1 + 2 trunc)/3

Data results

algorithm	difference	error	separation
Trunc hits	786.	111.	7.0
clusterz	283.	45.	6.2
cluster1	1063.	138.	7.6
clusterc1	815.	99.	8.2
clusterc2	1330.	120.	11.0

G4 MC results

algorithm	difference	error	separation
Trunc hits	966.	83.	11.5
clusterz	472.	40.	11.5
cluster1	1044.	95.	10.9
clusterc1	993.	77.	12.8
clusterc2	1533.	85.	18.0

- a 5% σ(dEdx) for electrons corresponds to 6.0 σ
- Clusterc2 data would then correspond to σ=2.7%

Separation: dEdx results III

One can wonder how well the numbers extrapolate to a full ILD TPC and whether the results are:

1) Sensitive to the diffusion
   generate electrons 2.5 GeV without diffusion
2) Sensitive to the particle type
   generate muons 250 GeV with diffusion

G4 MC results			electron separation	no diff	muon
Trunc hits	966.	83.	11.5	11.9	11.8
clusterz	472.	40.	11.5	11.8	11.5
cluster1	1044.	95.	10.9	12.8	11.5
clusterc1	993.	77.	12.8	13.3	13.4
clusterc2	1533.	85.	18.0	18.7	18.4

One can conclude that there is a weak sensitivity to diffusion and particle type. The separation numbers stay pretty similar within a relative 10%.

Summary: dEdx performance

The simulation of dEdx in G4 needs further understanding. In general the dEdx in G4 has less fluctuations and the TPC dEdx resolutions are smaller by a factor of 1.6. One can further see that too frequently large clusters with a small width are produced.

Using the test beam data several events are combined to extrapolate to the full TPC performance.
The dEdx extrapolations are conservative and take into account dead space in the pixel TPC module

- Pixel active track length 8 (padrows)*9*220 pix = 87 cm
- while full TPC track length = 133 cm

Several algorithms were developed using the truncated number of hits, the cluster zero and one counts.
The performance of the different algorithms is measured using the electron data comparing that to data with a 70% efficiency.

Summary: dEdx performance

The – current - best algorithm combines cluster counting and the number of truncated hits.
To quantify the performance we use:
separation = $(dEdx(e)-dEdx(0.7*e))/\sigma(e)$ ("e-K" at 2 GeV)

A 5% σ(dEdx) for electrons corresponds to 6.0 σ
The best algorithm gives on data a separation of 11 σ
This corresponds to a σ(dEdx) =2.7%

The results for the separation are similar within a relative 10% if the diffusion is put to zero or the particle changed to a muon.

We can therefore conclude that a pixel TPC will provide a separation of 11σ corresponding to a σ(dEdx) of 2.7%.

Pattern recognition exploiting the cluster shape and machine learning can probably improve on this result.