Subject: TPC dE/dx
Posted by StefanoSpataro on Tue, 08 Mar 2011 14:54:08 GMT
View Forum Message <> Reply to Message

Dear all,
I have tried to write a "quick and dirty" algorithm to calculate TPC dE/dx with a truncated mean
-> pid/PidCorr/PndPidTpcInfo.cxx
The energy is taken from PndTpcCluster::amp(), the dx is calculated somehow from the distance of consecutive hits, as a first approximation (in reality it is the sum of the semi-distance with the previous hit and the semi-distance with the next hit).

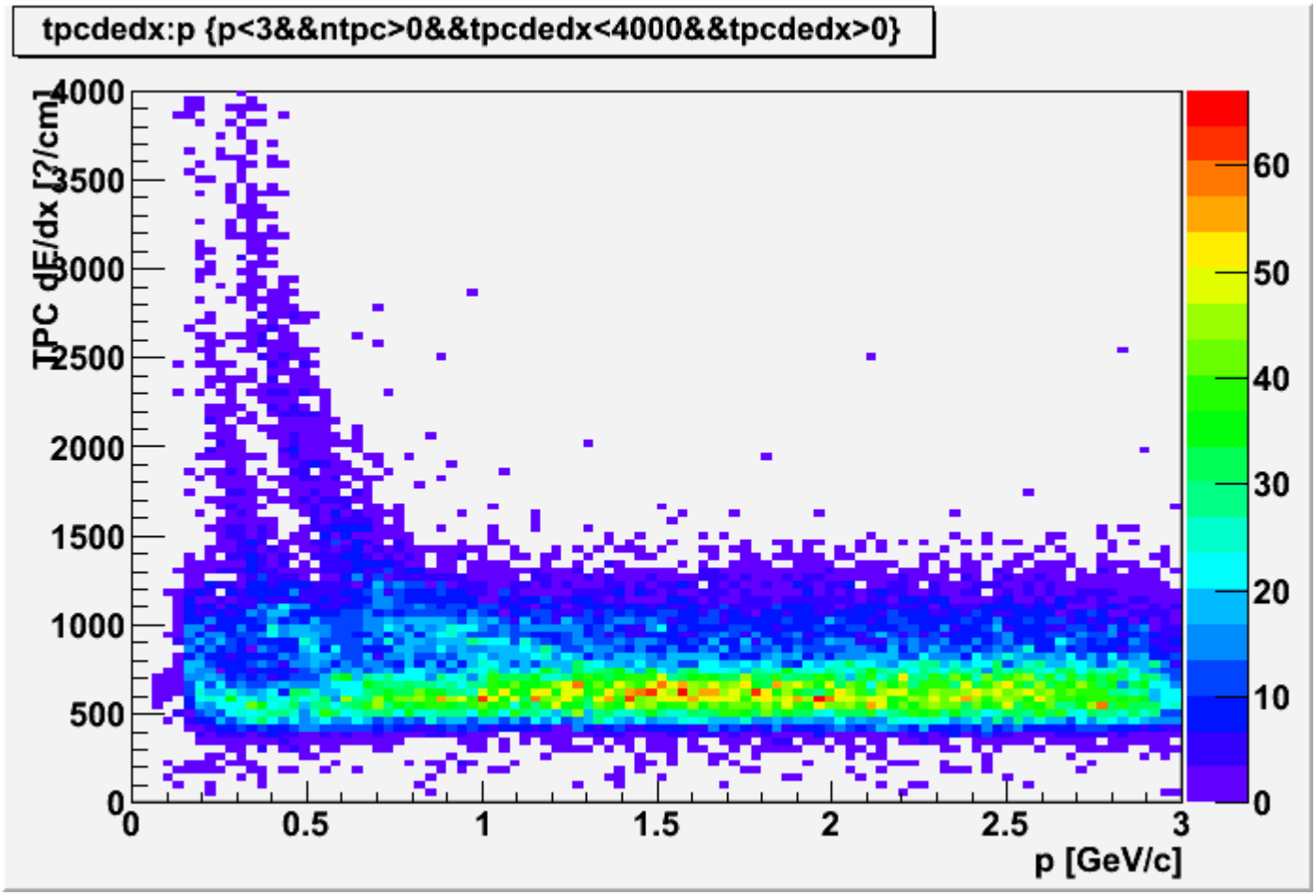
This is the plot I obtain, after 10k events * (p + e + pi + mu + k) with the box generator:

If I compare the plot to what present in the TPR, page 142 of the pdf, I have some questions:
a) What is the unity of PndTpcCluster::amp()? If I assume it is in GeV as the pandaroot standard, I obtain a around MeV/cm, while I suppose it should be something like keV/cm. And the numbers are still different. (I am using Geant3+Alice settings) Maybe the gas mixture was changed?
b) the plot is worse than the TPR one, because it is using reconstruction and not MC, but it looks like electrons are less separated now than the old plot (where they were quite far). Here they are overlapping with a bit higher de/dx, you are not able to find them like in the TPC plot. This is not clear to me.

Comments from experts are welcome.

File Attachments

1) tpc_dedx_test.gif, downloaded 1453 times



Subject: Re: TPC dE/dx
Posted by Anonymous Poster on Tue, 08 Mar 2011 15:02:53 GMT
View Forum Message <> Reply to Message

Hi Stefano,

could you please use another momentum region for you generation? For example [0.1,1.] GeV/c? The interesting region in this plot is not very visible and populated.

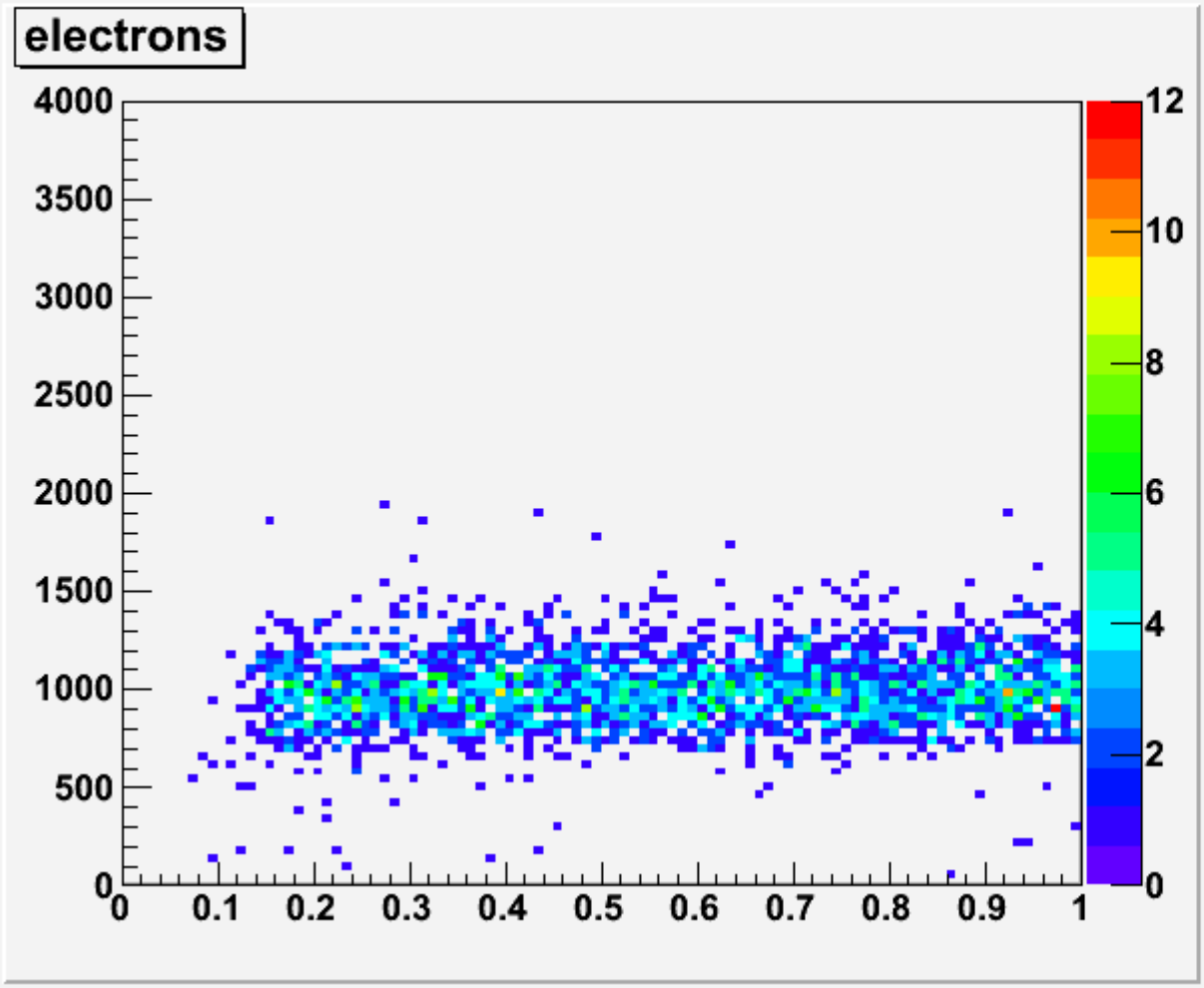
Cheers, Chrstian

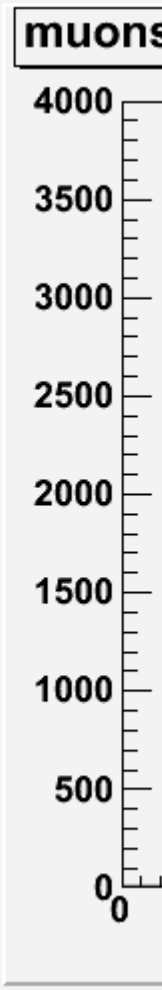
Subject: Re: TPC dE/dx
Posted by StefanoSpataro on Tue, 08 Mar 2011 15:28:19 GMT
View Forum Message <> Reply to Message

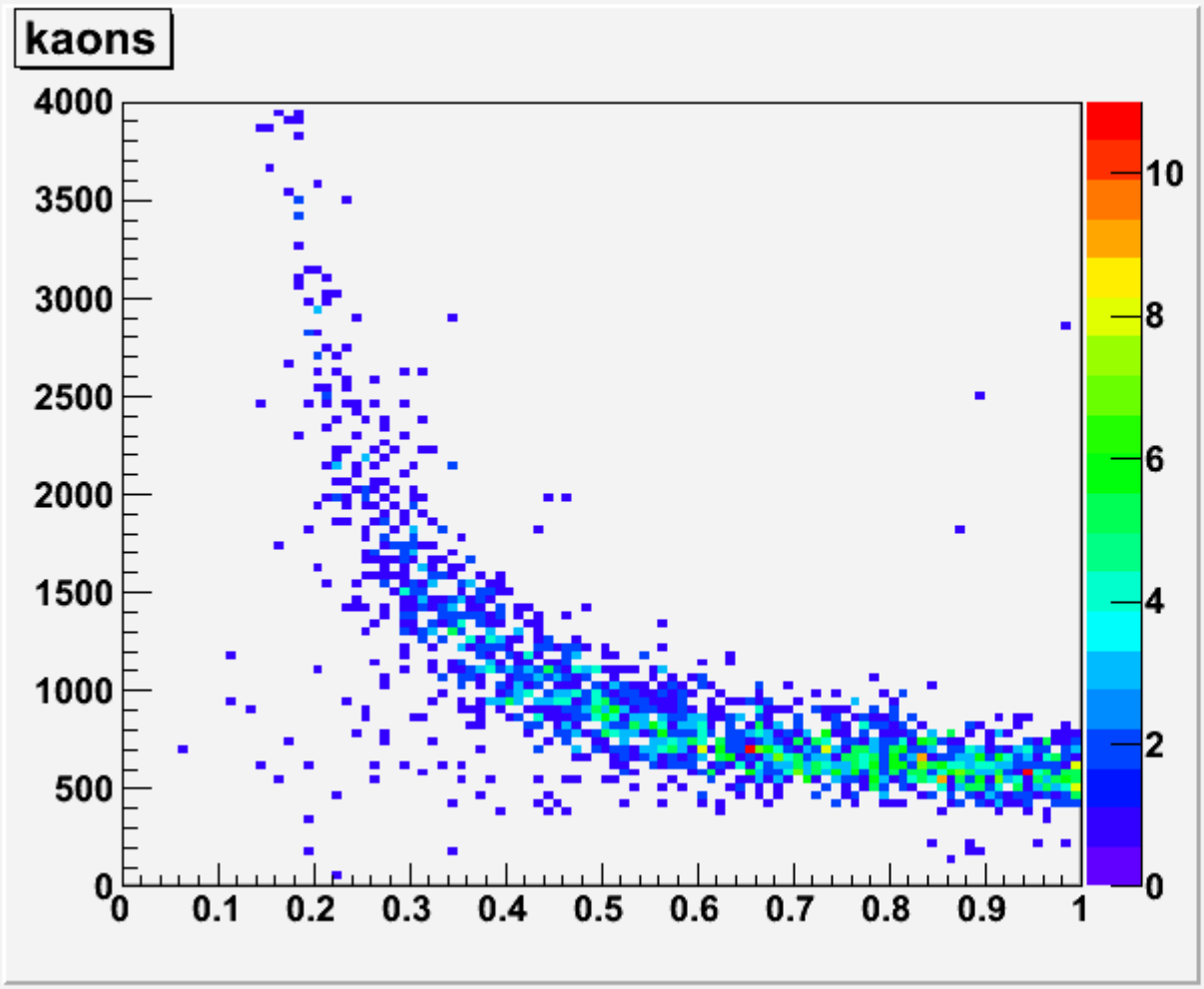
Probably this plot will help.

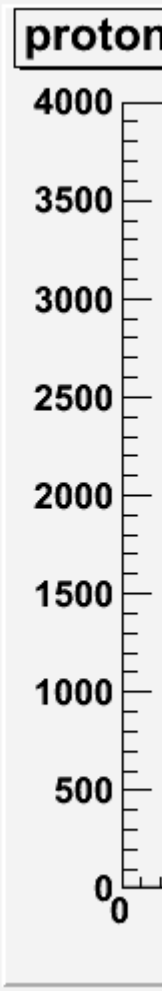
File Attachments

1) tpc_dedx_test_6.gif, downloaded 1095 times


electrons
4000
3500
3000
2500
2000
1500
1000
500
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
12
10
8
6
4
2
0


muons
4000
3500
3000
2500
2000
1500
1000
500
0
0


kaons
4000
3500
3000
2500
2000
1500
1000
500
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1
10
8
6
4
2
0


proton
4000
3500
3000
2500
2000
1500
1000
500
0
0

Subject: Re: TPC dE/dx
Posted by Felix Boehmer on Tue, 08 Mar 2011 16:03:13 GMT
View Forum Message <> Reply to Message

Hi Stefano,

very cool that you look into this!

Just some comments:
The cluster amplitude is defined after the FrontEnd and given in ADC channels! It is calculated from the analogue signal amplitude (on the pad) using some parameters: ADC_amp = Sig_amp /(adcmax/adcmaxcount) (See PndTpcFrontend).

You have to be very careful when defining the dx - a lot of effects are at work here (clustering!). The best approach is to use the actual track fit, as it is done in the task that I sent you a while ago

Cheers

Felix

Subject: Re: TPC dE/dx
Posted by StefanoSpataro on Tue, 08 Mar 2011 16:16:59 GMT
View Forum Message <> Reply to Message

Felix Boehmer wrote on Tue, 08 March 2011 17:03Hi
Just some comments:
The cluster amplitude is defined after the FrontEnd and given in ADC channels! It is calculated from the analogue signal amplitude (on the pad) using some parameters: ADC_amp = Sig_amp /(adcmax/adcmaxcount) (See PndTpcFrontend).

Ok, now I understand why the numbers do not match.

Quote:
You have to be very careful when defining the dx - a lot of effects are at work here (clustering!). The best approach is to use the actual track fit, as it is done in the task that I sent you a while ago

I know, this was just the first test, but it seems somehow to work and it is very fast. Using track extrapolation would improve a bit the resolution (even if I would not aspect huge improvements), but it will take much more computing time. I will see if I will be able to implement this updated version, however at least we have now a starting point working with PndTrack/PndPidCandidate.

Subject: Re: TPC dE/dx
Posted by Felix Boehmer on Wed, 09 Mar 2011 12:53:54 GMT
View Forum Message <> Reply to Message

Hi Stefano,

Quote:I know, this was just the first test, but it seems somehow to work and it is very fast. Using track extrapolation would improve a bit the resolution (even if I would not aspect huge improvements), but it will take much more computing time. I will see if I will be able to implement this updated version, however at least we have now a starting point working with PndTrack/PndPidCandidate.

this is surely nice for a first test. However, the improvement are not going to be small!
First of all, for low-momentum tracks the error you make by assuming a linear distance of clusters compared to the true helical track become significant. Secondly, one has to think about the (large!) effects of the clustering itself. What is the best measure for energy loss between the center of masses of two clusters, for instance? Not necessarily the amplitude of the second cluster! Here one probably needs to go back to the single digis! (I think this is how we implemented it in the task anyway). Stepping along the fitted track in fixed intervals, summing up all the single digis amplitudes is probably the best solution.

What I want to convey is that the problem is quite complex, and I expect LARGE systematic differences between your method and a more involved implementation. There is a reason why we still don't have something nice and easy to use

Cheers

Felix