Subject: Changes in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Mon, 25 Jul 2011 13:21:32 GMT
View Forum Message <> Reply to Message

Dear Colleagues,

in the last days I made several changes and Improvements in the Pattern Recognition.
Many of the tracking parameters have changed their optimal settings, and new setters have been introduced.

In order to keep things simple, I set all parameters to reasonable default values in the PndTpcRiemannTrackingTask.

I also changed all the macros and took out the setter functions.
If you use your own macros, please use the PndTpcRiemannTrackingTask as follows, then all the default values will be used. They are already tuned and will be updated if necessary.

```
PndTpcRiemannTrackingTask* tpcSPR = new PndTpcRiemannTrackingTask();
tpcSPR->SetPersistence();
tpcSPR->useGeane(); // use RKTrackrep and GeaneTrackrep
tpcSPR->SetMCPid(); // use ideal particle identification
//tpcSPR->SetPDG(211);
fRun->AddTask(tpcSPR);
```

If you want you can use a specific pdg hypothesis, but then remember to comment out tpcSPR->SetMCPid();

If you know what you are doing and/or want to try different track finding parameters, please refer to PndTpcRiemannTrackingTask.h. I put comments there so that you understand what the parameters are doing.

Otherwise I recommend using the default values.

I hope recent changes will improve efficiency!

regards,

Johannes

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Dima Melnychuk on Mon, 25 Jul 2011 19:53:20 GMT
View Forum Message <> Reply to Message

Hi Johannes,

Unfortunately after your modification PndTpcRiemannTrackingTask still shows very low efficiency with ideal pid hypothesis.

For eta_c channel after your updates multiplicity of reconstructed charged particles looks like (with 4 generated kaons).

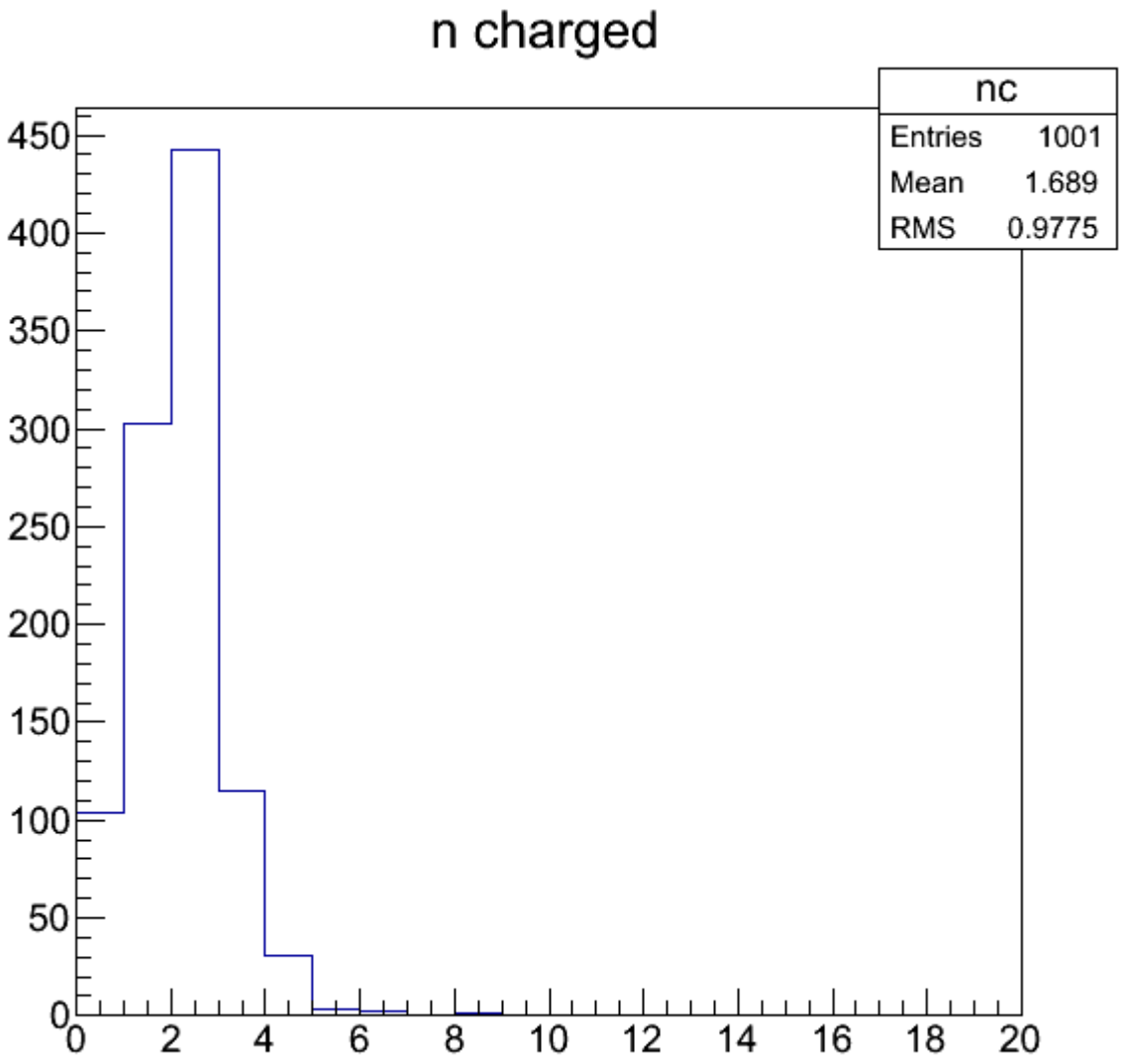
Wheras before these modifications with pion particle hypotesis

So ideal pid still does not work here properly.

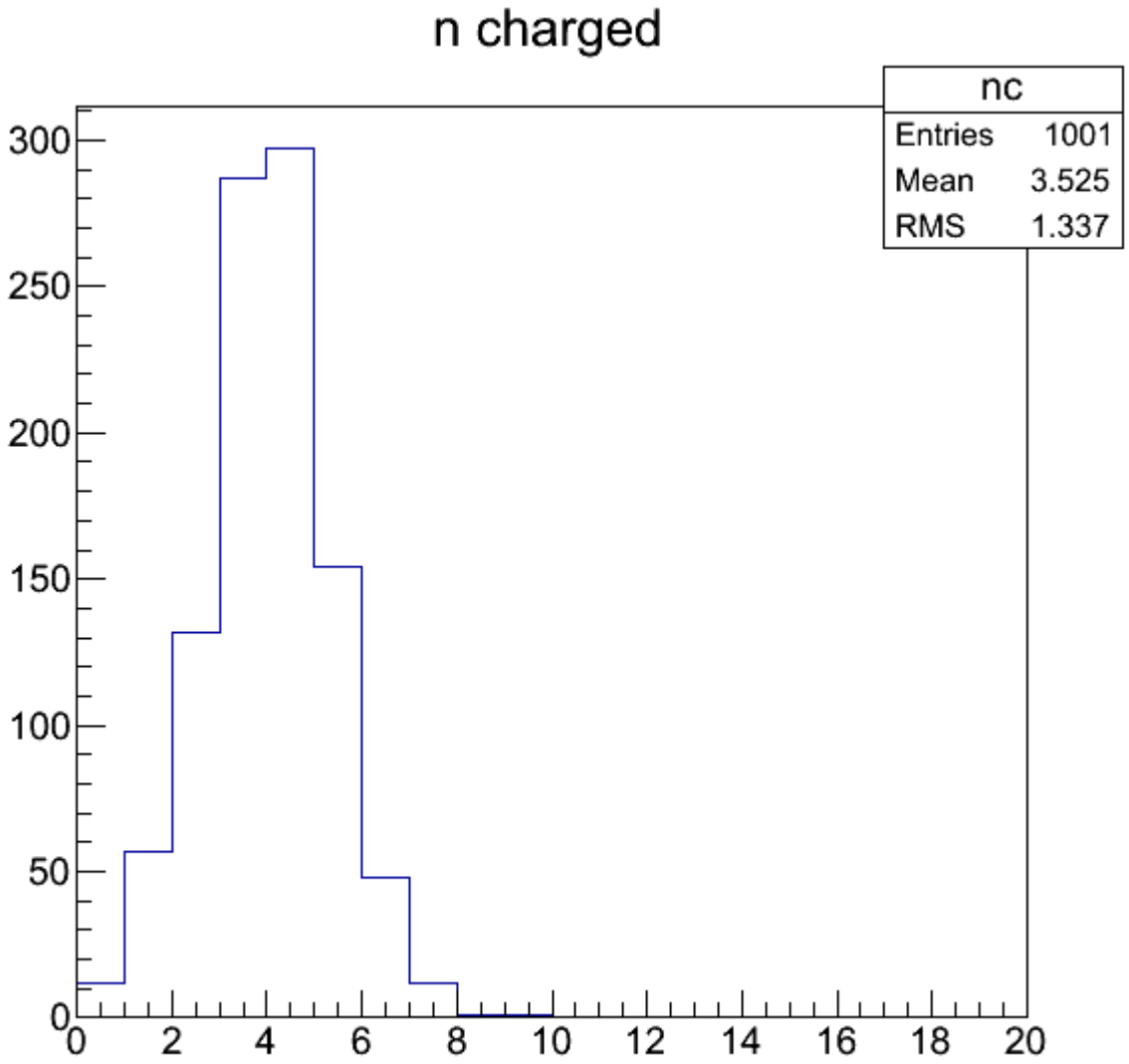
Dima

File Attachments

1) c1_tpc_mc.png, downloaded 1375 times



2) c1_tpc_pion.png, downloaded 1406 times



Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Mon, 25 Jul 2011 21:06:50 GMT
View Forum Message <> Reply to Message

Hi Dima,
Have you maybe tried to run kaon hypothesis with tpc, just to understand if this changes a lot the results (also vertex fitting) compared to pion hyp. If not, maybe the problem stays elsewhere.

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Dima Melnychuk on Mon, 25 Jul 2011 21:24:51 GMT
View Forum Message <> Reply to Message

Hi Stefano,

I tried kaon hypothesis and results (eta_c reconstruction efficiency and mass resolution) were

very similar. I cannot quantitate the differnce between kaon and pion hypothesis since I tried them on different data sets, but the difference was within 10 %.

And I can try it once again to check the vertex resolution.

Dima

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Dima Melnychuk on Tue, 26 Jul 2011 07:30:10 GMT
View Forum Message <> Reply to Message

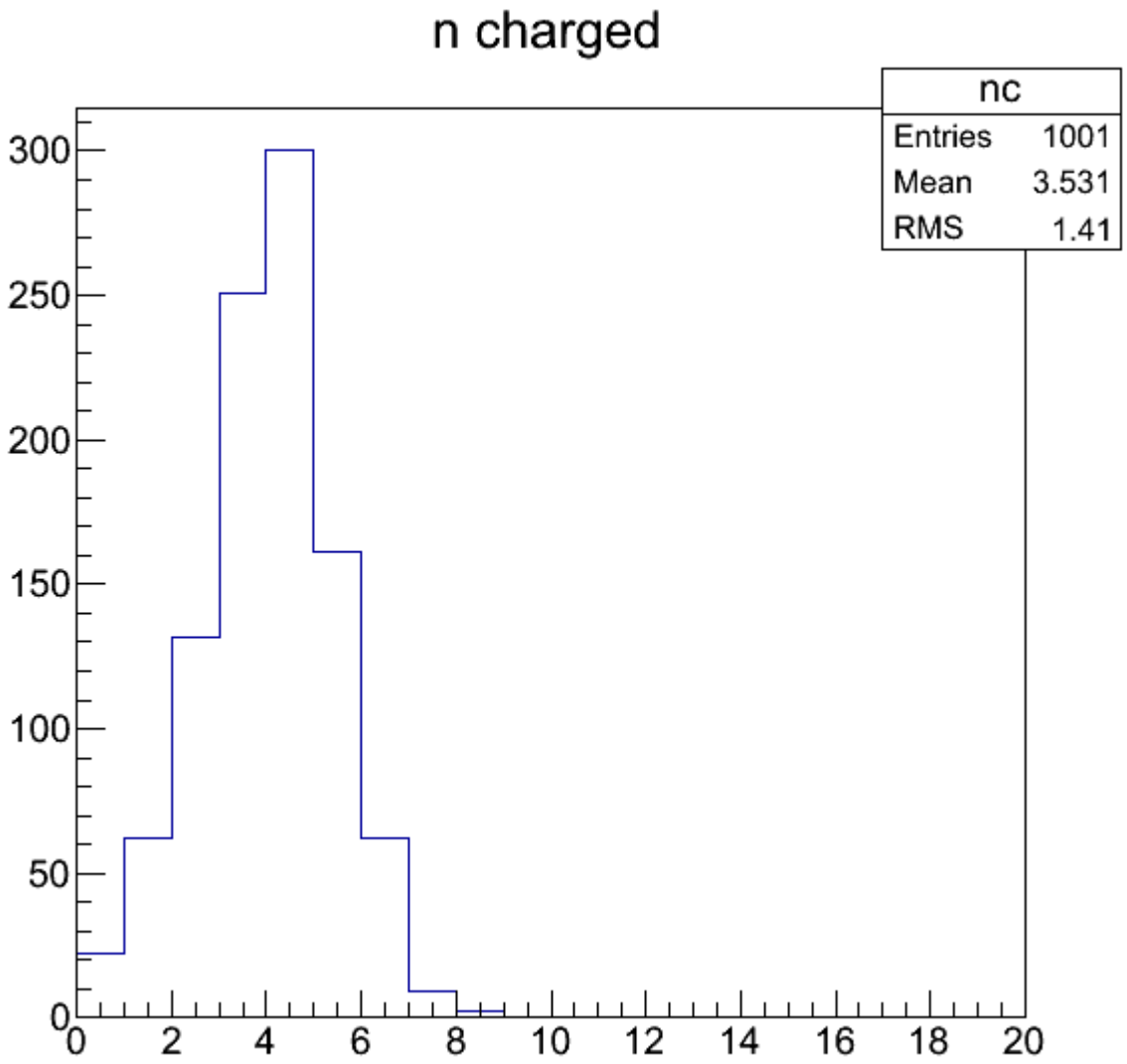
Hi Johannes,

despite the problem with ideal pid, running updated tpc code with pion hipotesis I see an improvement in eta_c reconstruction efficiency 8.6% vs 5.9% before the update. However it's still too low in comparison with STT.

And muliplicity of reconstructed charged particles with pion hipothesis and updated tpc code.

Dima

File Attachments
1) c1_tpc_pion_new.png, downloaded 1474 times



Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Tue, 26 Jul 2011 09:33:37 GMT
View Forum Message <> Reply to Message

Hi,

we are still working on some other problems:
- There is a bug when the Kalman tries to invert the charge
- We suffer from inefficiencies due to sectorization

I hope we can submit some fixes still today.

regards,

Johannes

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Lia Lavezzi on Tue, 26 Jul 2011 09:53:58 GMT
View Forum Message <> Reply to Message

Hi Johannes,
Quote:There is a bug when the Kalman tries to invert the charge
just to understand, what do you mean by "the Kalman tries to invert the charge"? Is this some kind of feature which is inside the base Kalman classes and affects also the STT or it is only in the TPC reconstruction classes?

Cheers,
Lia.

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Tue, 26 Jul 2011 10:12:45 GMT
View Forum Message <> Reply to Message

Hi Lia,

the GFKalman treats q/p as a free parameter (which can also change sign).
But in the RKTrackRep, the charge was fixed. I fixed this bug, and now the sign of the charge is set according to fState[0][0] in setData().

I didn't check GeaneTrackRep yet, but maybe there is the same problem.

This problem resulted in a very bad fit if the charge was set wrong.

regards,

Johannes

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Lia Lavezzi on Tue, 26 Jul 2011 11:15:40 GMT
View Forum Message <> Reply to Message

Hi Johannes,
thanks for your reply.

Quote:the GFKalman treats q/p as a free parameter (which can also change sign).
But in the RKTrackRep, the charge was fixed. I fixed this bug, and now the sign of the charge is set according to fState[0][0] in setData().
With your fix in the RKTrackRep, is the charge set to a fixed value along all the Kalman procedure, i.e. the Kalman is not allowed to change the sign of q/p anymore, or does the fix just set the final charge back to the input one when the Kalman output has the wrong sign (after all the fitting procedure)?

Quote:I didn't check GeaneTrackRep yet, but maybe there is the same problem.
Are you looking also into GeaneTrackRep to check whether there is the problem or not? Just to know whether I have to investigate this by myself or not.

Cheers,
Lia.

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Tue, 26 Jul 2011 11:45:45 GMT
View Forum Message <> Reply to Message

Quote:With your fix in the RKTrackRep, is the charge set to a fixed value along all the Kalman procedure, i.e. the Kalman is not allowed to change the sign of q/p anymore, or does the fix just set the final charge back to the input one when the Kalman output has the wrong sign (after all the fitting procedure)?

The Kalman is fitting q/p, and the sign of q/p can change during the fitting process.
The problem was that a change in sign of q/p was not correctly passed to the RK propagator, which is now fixed.

That means: The absolute value of the charge is fixed, its sign can change.

I just looked into GeaneTrackRep and there everything seems to be fine.

cheers,

Johannes

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Lia Lavezzi on Tue, 26 Jul 2011 12:01:22 GMT
View Forum Message <> Reply to Message

Quote:That means: The absolute value of the charge is fixed, its sign can change.
Then you have a found track with a charge sign as input to the Kalman and AFTER the Kalman fit you can have that the charge has changed sign, right?

Since the Kalman changes the sign at runtime, could it happen that the charge associated to the last hit (last params I mean) has opposite sign with respect to the one of the first hit?
Cheers,
Lia.

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Tue, 26 Jul 2011 12:07:44 GMT
View Forum Message <> Reply to Message

Quote:Then you have a found track with a charge sign as input to the Kalman and AFTER the Kalman fit you can have that the charge has changed sign, right?

Yes.

Quote:Since the Kalman changes the sign at runtime, could it happen that the charge associated to the last hit (last params I mean) has opposite sign with respect to the one of the first hit?

When smoothing is not turned on, the track has only one state, and therefore only one charge.

I don't know what happens with smoothing turned on.
But what I saw in my display ist that usually in the second iteration the Kalman flips the charge if necessary. So with 3 iteration the charge should be consistent for all hits.

cheers,

Johannes

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Anonymous Poster on Tue, 26 Jul 2011 12:17:30 GMT
View Forum Message <> Reply to Message

Hi,

even without smoothing we do forward and backward fits. So in principle we could have a different charge in the last point (from the forward fit) and the first point (from the backward fit).

However: Since we do repeated Kalman iterations this should never happen. A wrong charge hypothesis could be turned around in the first forward fit, but then it will stay. So with >=2 iterations this should never happen.

Cheers, Christian

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Lia Lavezzi on Tue, 26 Jul 2011 13:47:48 GMT
View Forum Message <> Reply to Message

Hi Christian

Quote:However: Since we do repeated Kalman iterations this should never happen. A wrong charge hypothesis could be turned around in the first forward fit, but then it will stay. So with >=2 iterations this should never happen.
Actually for the STT we saw that more than 1 iteration does not improve the results, for some momenta (lower ones) it even worsen the efficiency.

For this reason we would like to run the reconstruction with only one iteration. This is the best option for us.

Anyway, we will check how many times, with one iteration, we get that the Kalman sets the first and last hit charges to a different value and how many times it changes the charge sign to the wrong one. Maybe it does not happen so often to make us to worry about.

Ciao,

Lia.

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Tue, 26 Jul 2011 18:03:05 GMT
View Forum Message <> Reply to Message

Quote:we are still working on some other problems:
- There is a bug when the Kalman tries to invert the charge
- We suffer from inefficiencies due to sectorization

I hope we can submit some fixes still today.

I just commited a new version of the ClusterFinderSimple which gets rid of cluster splitting at the edges of the sectors.

This should improve efficiency a bit.

cheers,

Johannes

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Wed, 27 Jul 2011 09:20:04 GMT
View Forum Message <> Reply to Message

I forgot to commit the updated padplane file which is necessary.
It is now checked in.

Subject: Re: Changes in PndTpcRiemannTrackingTask
Posted by Dima Melnychuk on Wed, 27 Jul 2011 10:00:40 GMT
View Forum Message <> Reply to Message

Hi Johannes,

I've redone the eta_c reconstruction with updated code but still without updated padplane file.

And results are worse then before.
Using ideal pid in PndTpcRiemannTrackingTask I still have as before 0 reconstructed eta_c with 1000 events statistics.
But using kaon hypothesis efficiency decreased from 8.3% to 3.7%.

So I hope it is related to wrong padplane file and I will rerun digitiation/reconstruction now.

Dima