Subject: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Mon, 04 Jul 2011 18:11:05 GMT
View Forum Message <> Reply to Message

Submitting TPC macros on the grid, only 20 events for each job, we have seen that almost 50% of the jobs are crashing because of PndTpcRiemannTrackingTask.
In particular, the last message is the following:

PndTpcRiemannTrackingTask::Exec; Event Number: 16
Looping over 58 riemann tracks to write out
RKTrackRep::setPDG particle -22 not known to TDatabasePDG -> abort

This comes from the MonteCarlo particle hypothesis (why is it used inside a pattern recognition task?). Somehow some "photon" tracks are reconstructed, multiplied by some charge and finally -22 is not taken by the particle DB -> crash.

If we run 500 or 1000 events we expect to have a very low success rate.
I would ask to tpc tracking experts to take a look and to fix it as soon as possible, becase this prevents us to run production on the grid.

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Tue, 05 Jul 2011 09:47:21 GMT
View Forum Message <> Reply to Message

Dear Stefano,

I commited a fix: If the pdg is -22 it is reset to 22, and an additional check is performed if the pdg is valid. If not, the track is skipped.

Please update the PndTpcRiemannTrackingTask to version 12579.

regards,

Johannes

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Tue, 05 Jul 2011 10:56:32 GMT
View Forum Message <> Reply to Message

Are we sure that genfit+geane is able to propagate photons correctly? I would say that this was never tested before.

And I think it makes no sense for tracking of charged particles. How can be that a photon is creating a track, which is also reconstructed?

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Fri, 08 Jul 2011 11:14:37 GMT
View Forum Message <> Reply to Message

Hi,

I have been working on the bug that we adressed in the meeting yesterday (if the Riemann PR gets the track in a wrong direction, it flips the pdg id).

What I'm doing now is sorting the track in a way that the winding matches the charge gathered from the mc pdg (instead of flipping the charge resp. pdg id).

The problem here is that secondary tracks will get the pdg of the primary track they stem from, because for secondaries we have no more pdg ids available at this point.
We still have to think about a solution here.

Dear Analysts, could you please update the RiemannTrackingTask and check if efficiencies increase?

regards,

Johannes

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Tue, 12 Jul 2011 14:56:20 GMT
View Forum Message <> Reply to Message

Hi,

I just checked in a new version of the PndTpcRiemannTrackingTask.
You can now specify a pdg in the reco macro, which can then be flipped by the RiemannTracker according to the winding.
The default pdg is 211.
Please remember to switch off MCPid which is on by default.

For example:

```
tpcSPR->SetPDG(15);
tpcSPR->SetMCPid(false);
```

Please have a try.

regards,

Johannes

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by Dima Melnychuk on Thu, 14 Jul 2011 08:25:34 GMT
View Forum Message <> Reply to Message

Hi,

I checked TPC performance for eta_c reconstruction after latest update of PndTpcRiemannTrackingTask.

I used kaon hypothesis in tpc reconstruction macro

tpcSPR->SetPDG(321);
tpcSPR->SetMCPid(false);

and eta_c peak appeared back but efficiency is still low in comparison with STT (5.4-5.9% for TPC vs 20-29% for STT).

So with 2000 events multiplicity of charged tracks:

And maximum efficiency we can expect is 54%.

Mass distribution for phi and eta_c (using 4C-fit) efficiency - 5.9%

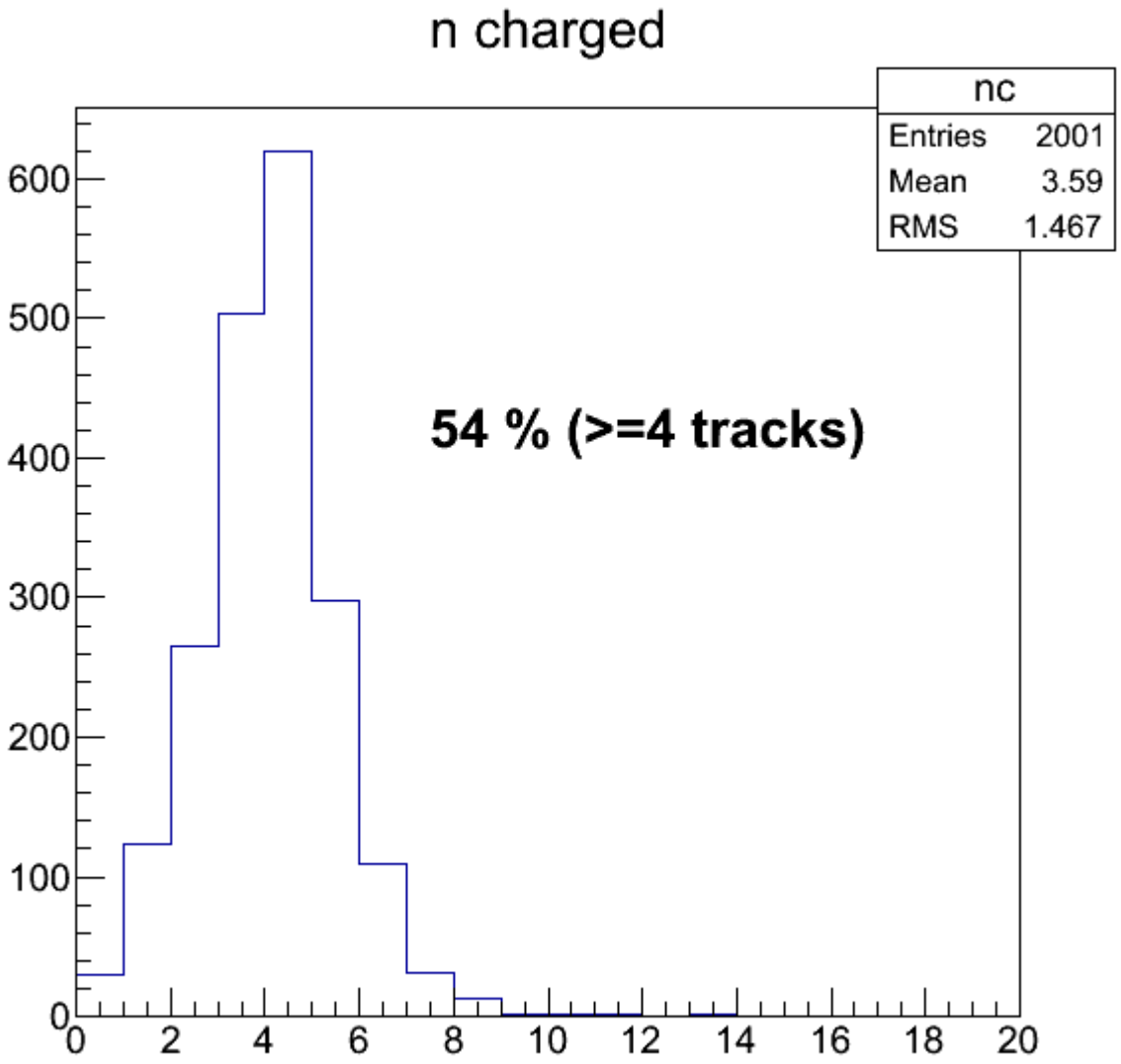
Mass distribution for phi and eta_c (using vertex fit) efficiency - 5.4%

Efficiencies in principle depend on cuts on chi2 of the fit and can be increased, but the 4-times difference between TPC and STT is with the same cuts.

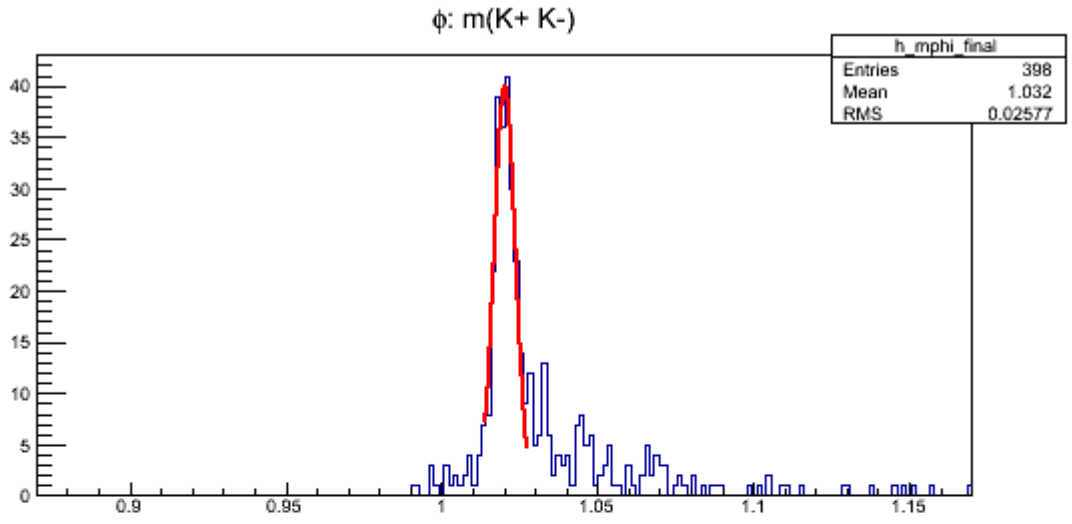
Dima

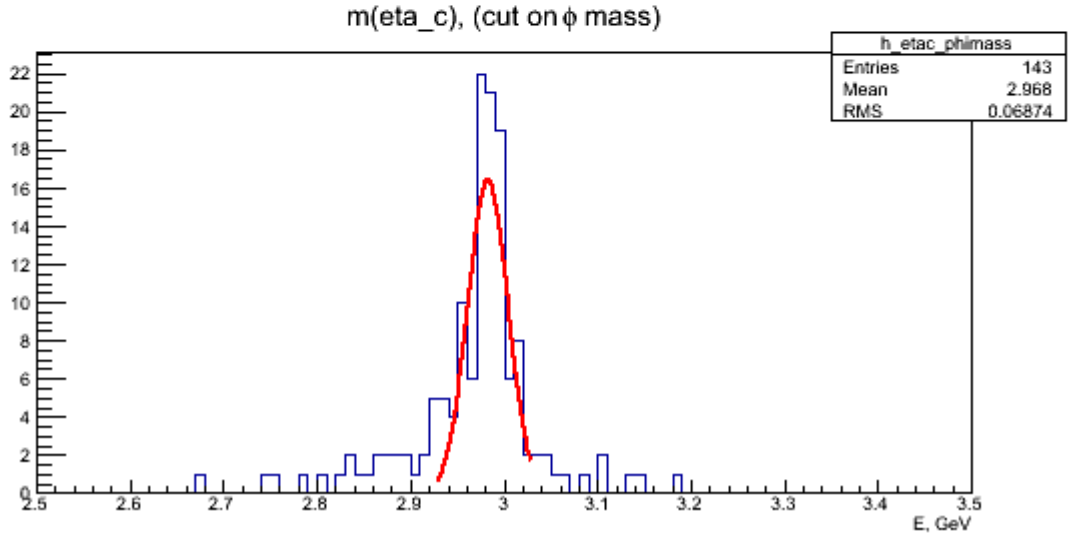
File Attachments

1) n_charged_tpc.png, downloaded 580 times

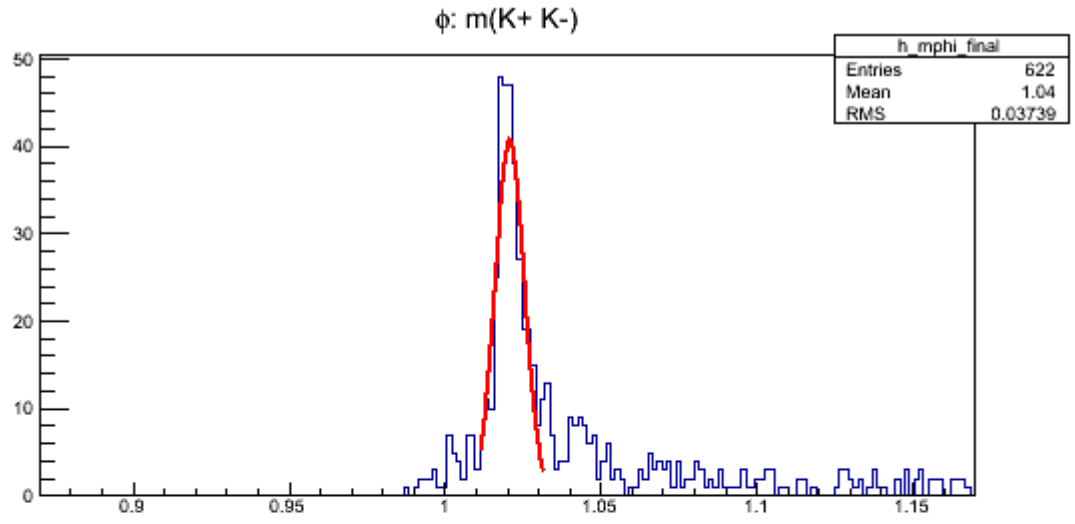


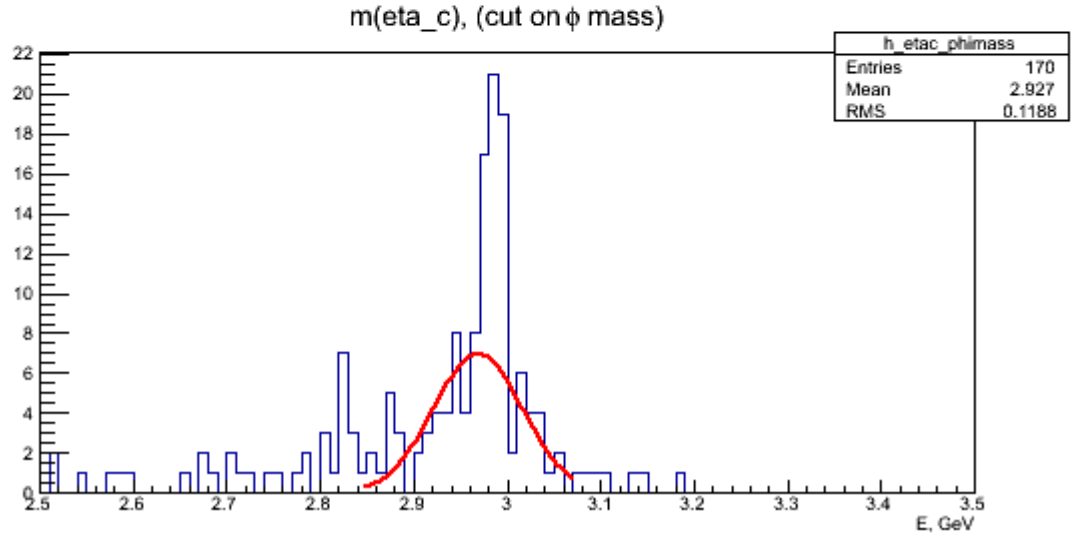
2) m_etac_tpc.png, downloaded 557 times





3) m_etac_tpc_vtx.png, downloaded 523 times





Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by Johannes Rauch on Thu, 14 Jul 2011 08:53:11 GMT
View Forum Message <> Reply to Message

Hi Dima,

its very strange that the TPC efficiency is so low. Yesterday I started to run the tpc macros in the /macro/run/tdrct/eta_c folder, to see whats happening, but I ran into a problem:
When runnin run_ana_eta_c_tpc.C, he complains:

Warning in <TTree::AddFriend>: FriendElement cbmsim in file evt_points_tpc.root has less entries 10 than its parent Tree: 11

How can I get it running?

regards,

Johannes

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by Dima Melnychuk on Thu, 14 Jul 2011 08:58:17 GMT
View Forum Message <> Reply to Message

Hi Johannes,

This warning according to Tobias is harmless.
(http://forum.gsi.de/index.php?t=tree&th=3267&start=0&rid=78&
amp;S=226ae0a4cb62b7d1cbdcce6c470541d1#page_top)
It happens when you use trunk version of base and additional entries in digitization, reconstruction should be simply ignored.

Dima

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Thu, 14 Jul 2011 09:06:20 GMT
View Forum Message <> Reply to Message

However wil will run the old base packages on the grid, to avoid such a problem.

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Sat, 16 Jul 2011 21:27:34 GMT
View Forum Message <> Reply to Message

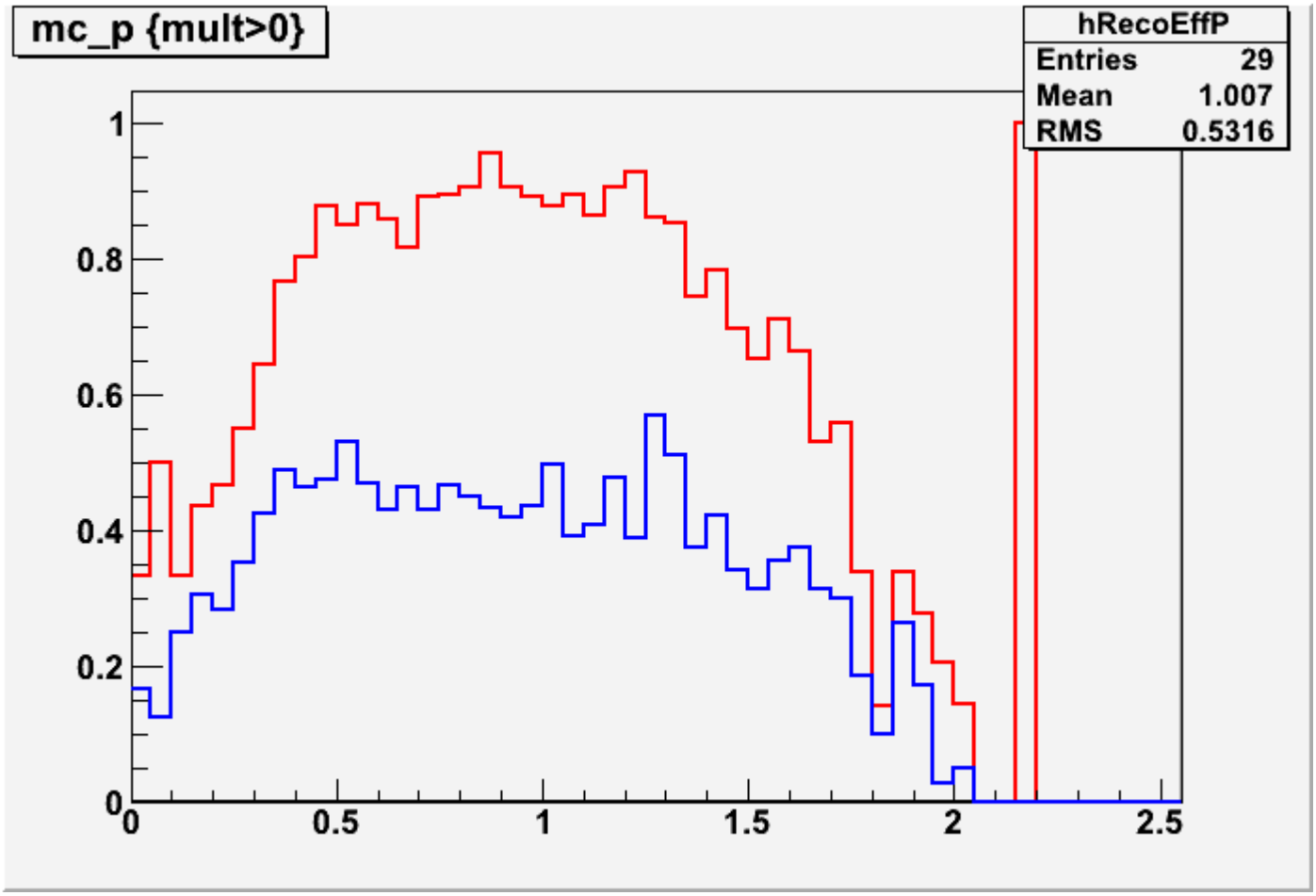
Hi,
I have tried with the new july11 release (12718) to run eta_c events with tpc, ideal riemann and using the pion hypothesis.

The following plot shows the single kaon efficiency as a function of momentum, as showed in the last tracking meeting. In blue with ideal pid, in red with pion pid:

You can see that the low efficiency is connected really to the ideal id code.
I suppose tomorrow I will start new files on the grid setting the pion hypothesis in the tpcriemann, to have a "decent" efficiency.

File Attachments

1) plot_tpceff_comp.gif, downloaded 508 times



Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by Dima Melnychuk on Sat, 16 Jul 2011 23:18:26 GMT
View Forum Message <> Reply to Message

Hi Stefano,

I have a clarification here.
Pion hypothesis gives better efficiencies than ideal pid, but not better then kaon hypothesis.

Here is the plot done with your macro when in reconstruction kaon hypothesis was used:

I would say that efficiency is not worse that with pion hypotheis.

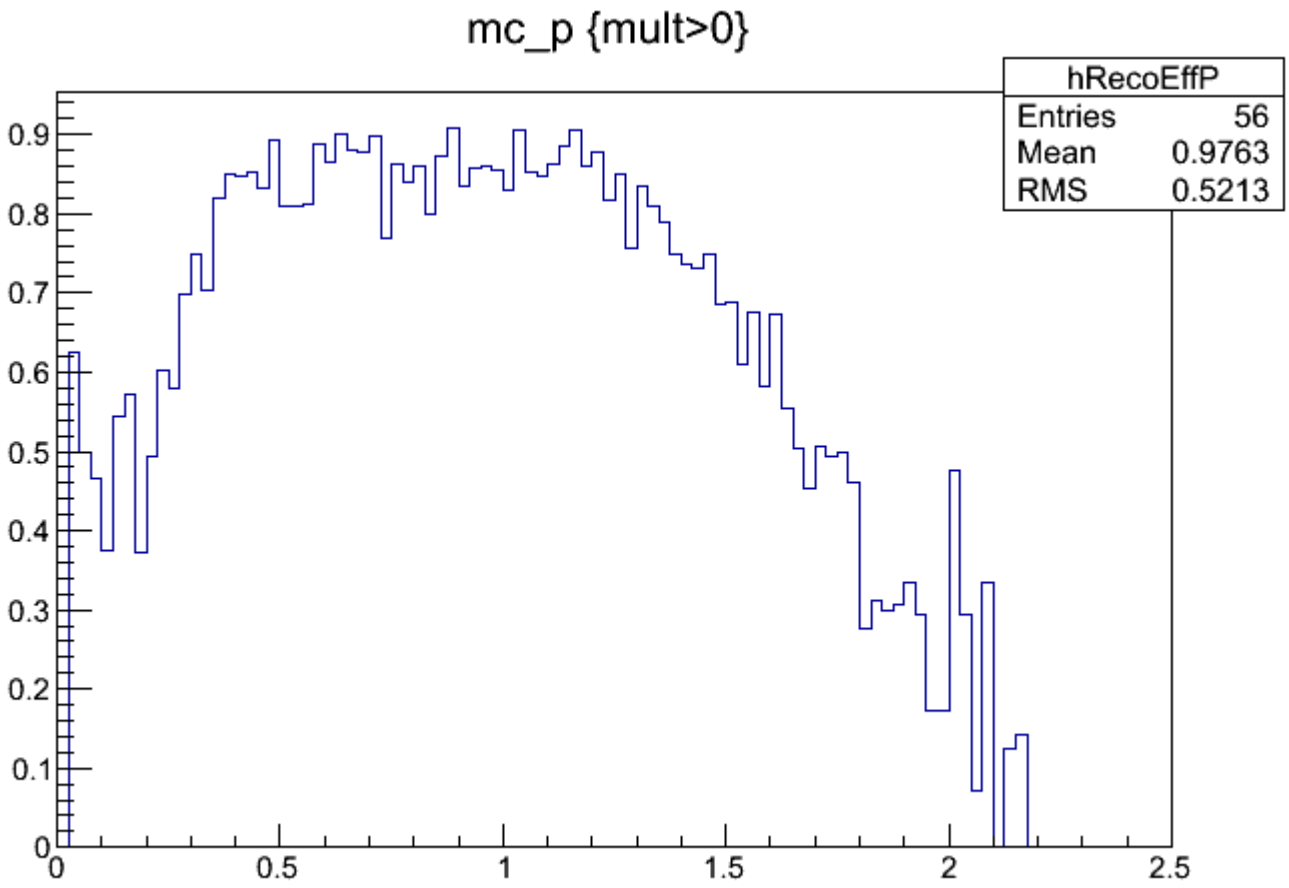
And this efficiency is with with reconstructed data, for which I obtained eta_c reconstruction efficiency 6%, i.e. 3-4 times lower than for STT.

So in the grid reconstruction macro specific hypotheis is prefered over ideal pid at the moment, but for eta_c channel it could be both pion or kaon.

Dima

File Attachments

1) eff.png, downloaded 501 times



Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Sun, 17 Jul 2011 08:05:57 GMT
View Forum Message <> Reply to Message

I think the problem is independent from the particle hypothesis, simply the "ideal" option has some problems in the code and does not work properly.
However, it is too early to use specific particle hyp at the pattern recognition level, i think.

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Sun, 17 Jul 2011 14:58:54 GMT
View Forum Message <> Reply to Message

Meanwhile I have tried to check also, tpc with pion riemann, phi and eta_c peaks with standard pion hypothesis (red) and with ideal hypothesis (blue) in PndPidCorrelator.

K+K- invariant mass

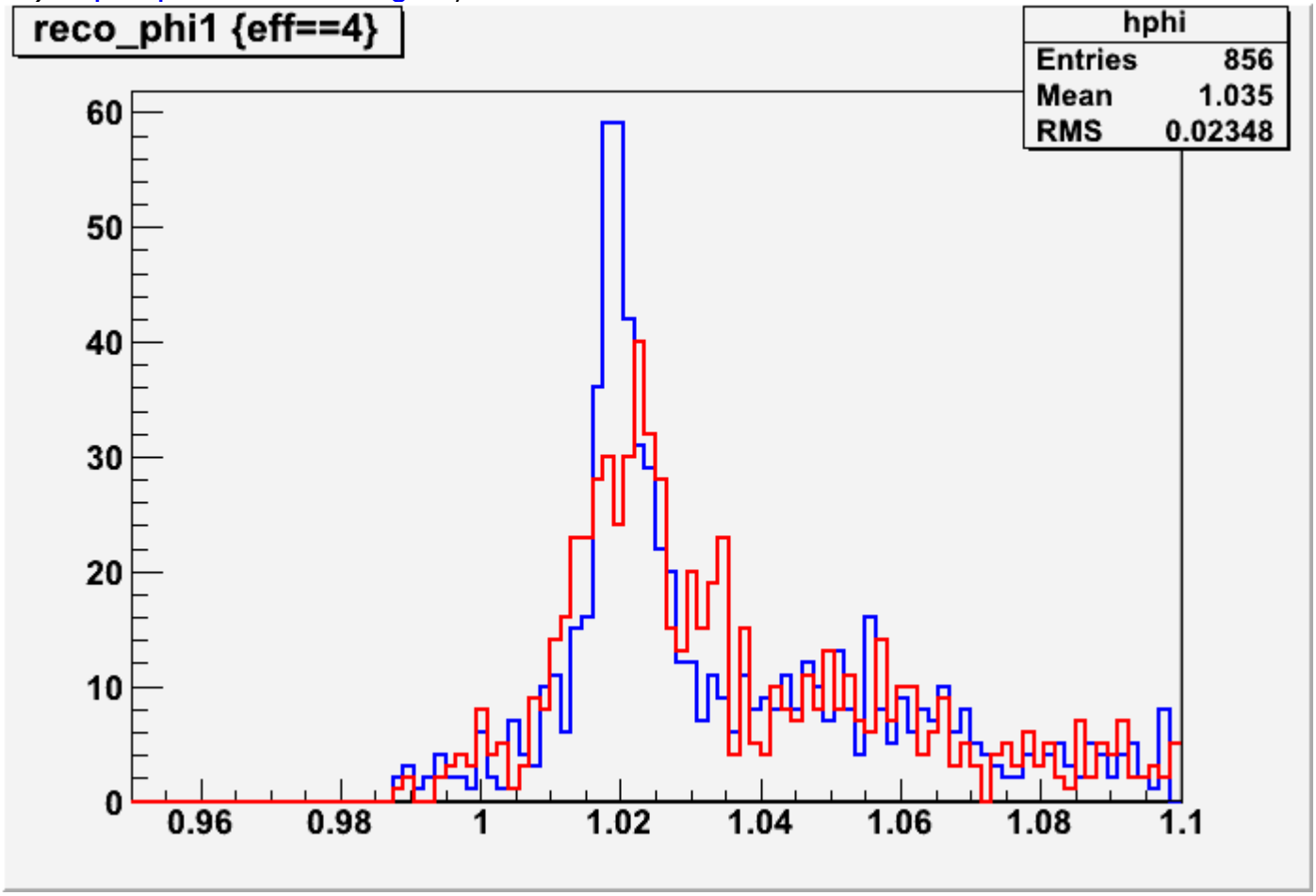
eta_c invariant mass

It seems that the ideal PidCorrelator (back extrapolation to IP using ideal particle hypothesis) produces a much sharper peak than with the default pion hypothesis. It seems a bit strange to me that the phi-phi invariant mass shows the same behaviour... however I will run the ideal pid correlator for tpc.

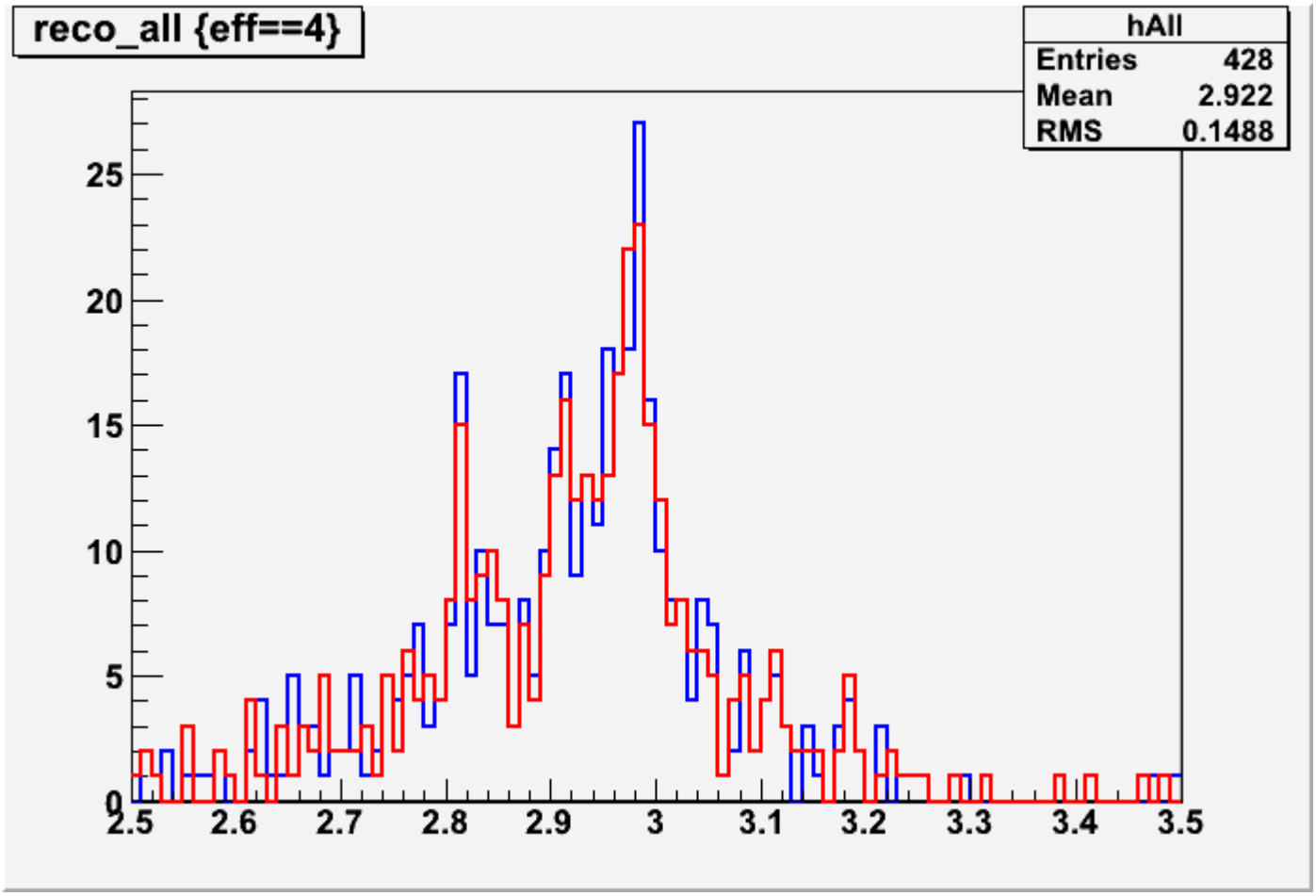
I will check the same also with stt.

File Attachments

1) tpc_phi_corrid.gif, downloaded 528 times



2) tpc_psi_corrid.gif, downloaded 486 times



Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Mon, 18 Jul 2011 07:59:09 GMT
View Forum Message <> Reply to Message

Sorry,
my fault,
in reality blue peak is with pion hypothesis, while in read the ideal hypothesis.
Something is going wrong in the PidCorrelator with ideal hyp, it seems in the backward extrapolation the phi angle is wrong. I will investigate. The same for stt...

Subject: Re: Crash in PndTpcRiemannTrackingTask
Posted by StefanoSpataro on Mon, 18 Jul 2011 12:46:49 GMT
View Forum Message <> Reply to Message

Hi,
there was a problem in the ideal settings of the PndPidCorrelator, which is now fixed in the trunk (12725). Now phi and psi peaks ae almost the same w/ and w/o ideal pid.