Subject: alternative to LHE tracking
Posted by Radoslaw Karabowicz on Fri, 17 Dec 2010 10:59:32 GMT
View Forum Message <> Reply to Message

Dear All,

In the last few months I have developed a track finder that is using most of the tracking detectors to form global tracks. The task is called PndBarrelTrackFinder, and is located in trunk/global directory. At present it uses MVD, STT, TPC and GEM hits, although the TPC option was not tested yet. I have also developed a QA task which should test the performance of the track finder. It produces several histograms in the output file that show the tracking efficiency, resolution, and number of detector hits in reconstructed tracks as a function of theta.

Few of them are presented below:

You can try yourself the task with macro/global/sim_BARREL.C and macro/global/digi_BARREL.C macros. To run, you should type:

root -l -q 'sim_BARREL.C(100)'
and
root -l -q 'digi_BARREL.C(100,22)'

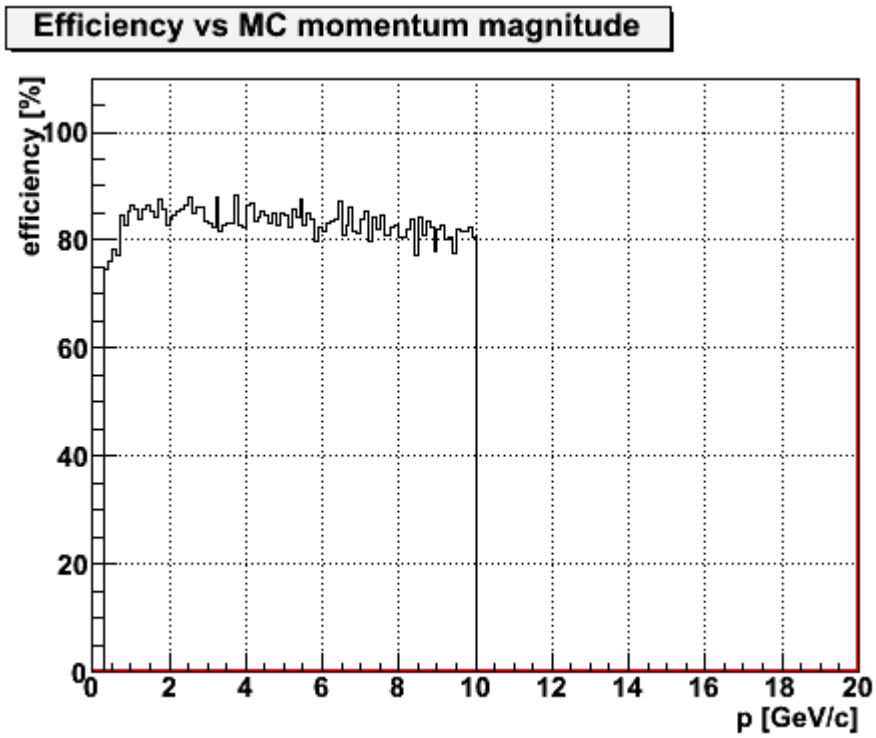
The output file: digi_22Part_n100_BTF.root will contain reconstructed tracks in BarrelTrack and BarrelTrackCand branches of cbmsim tree, and QA histograms in BarrelTrackFinderQA directory.

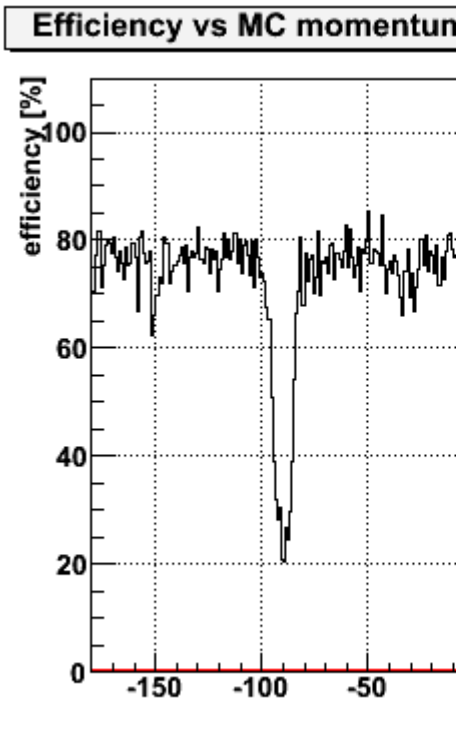
Please test it and report any bugs.

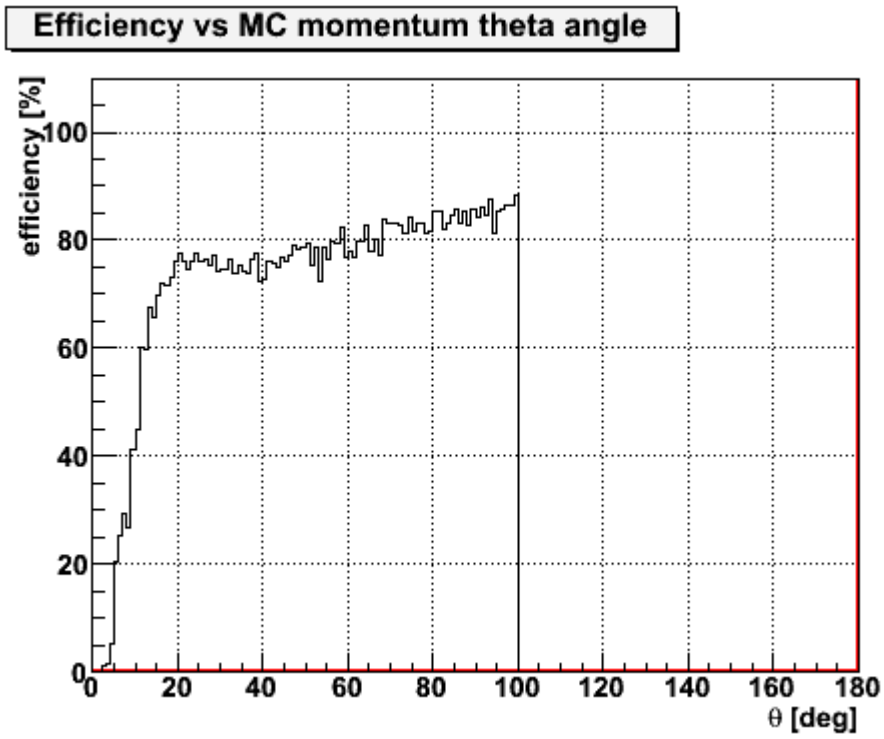
Sincerely yours,
radek

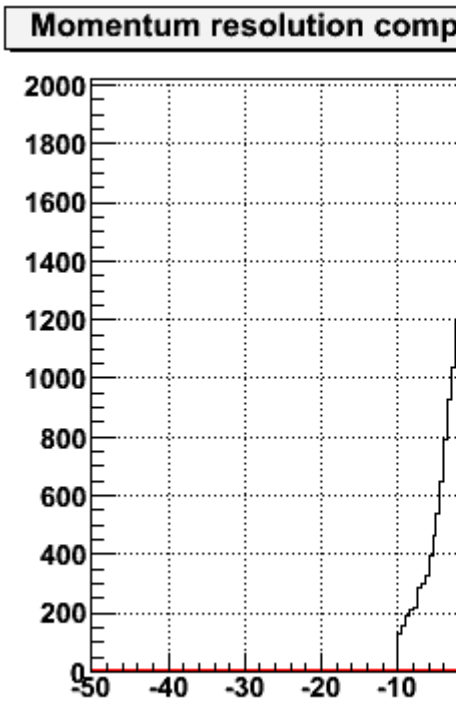
File Attachments

1) eff_and_res.gif, downloaded 1540 times

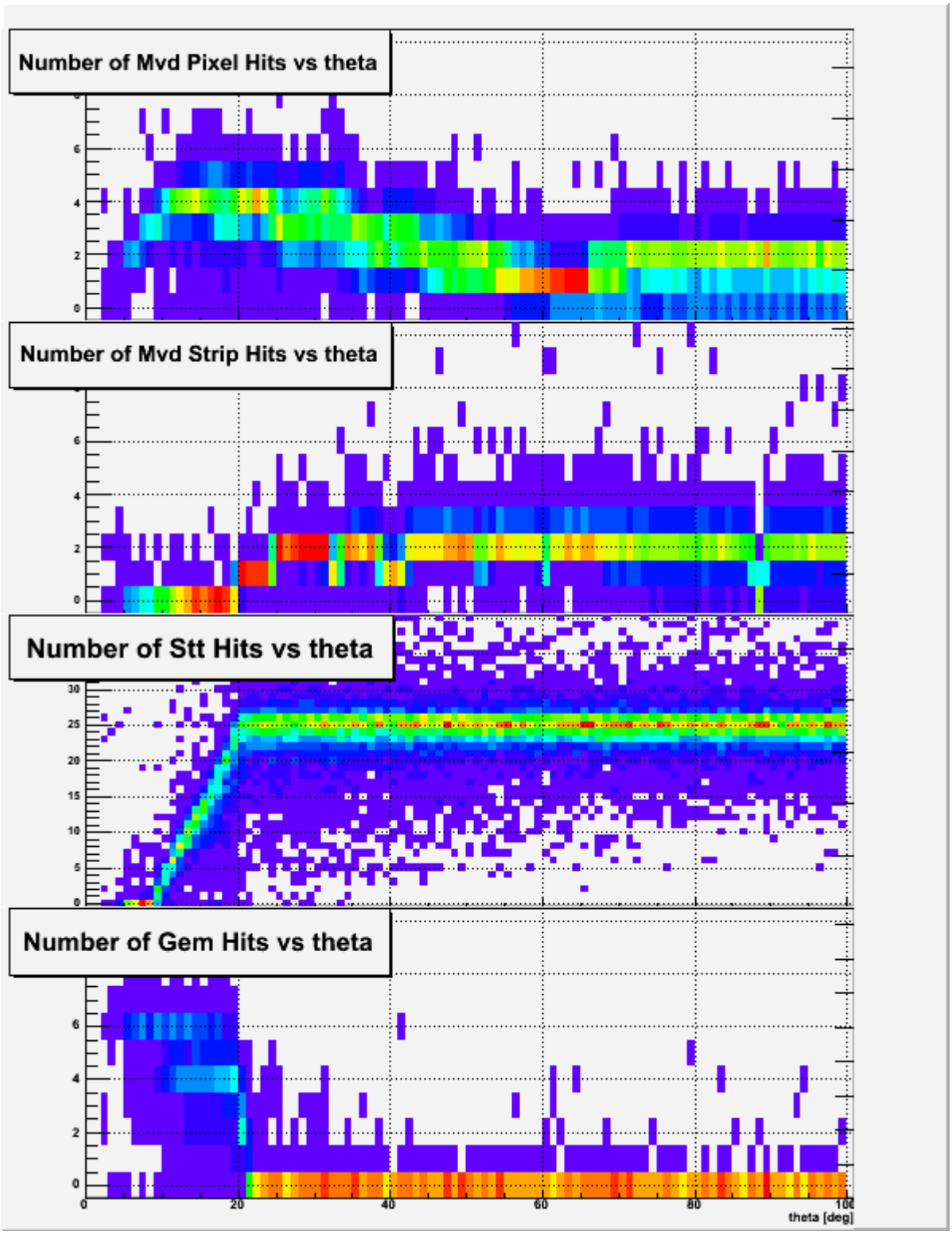








2) detectorHits.gif, downloaded 1622 times



Subject: Re: alternative to LHE tracking
Posted by StefanoSpataro on Fri, 17 Dec 2010 11:10:56 GMT
View Forum Message <> Reply to Message

Hi,
could you please shortly explain on which track finding principle is it based?

Dear Stefano and All,

I am not sure how to describe the tracking shortly. The track model is a helix, that is a circle in x-y plane plus some pitch along the z-axis. Since this assumes the constant magnetic field parallel to z, the tracking will not work in the Forward Detector, and is not doing very well in the GEM region. Of course some future extensions are possible.

The main part of the program is the loop over hits. The hits in one event are mixed so that there is no correlation possible to MC.
Inside the loop the hit is matched to already found tracks. If it doesn't match to any track, the hit is being combined with previously unused hits to form tracks. If no such combination is possible, it is put in unused hit array.

The hit to hit matching assumes the track is primary, that is one of the 3 needed points to construct a circle in x-y plane is set to origin. The other two points (or circles for STT) are set to (x,y) parameters of the hits. In case of hits in MVD, TPC or GEM, where there is no ambiguity, only one circle is fit to 2 hits. With one STT hit there are two circles found (one including, one excluding STT hit, both tangent to it). With two STT hits there are 4 combinations. All the combinations are stored in the track.

The hit matches to already found track, if its distance to any of the track circles in x-y plane is smaller than some parameter (for now around 1mm, but should depend on hit resolution). If the hit matches to the track, then circles with all other track hits are formed. If other track hits are close enough to the circle, then the circle parameters are stored in the track.

After the track contains more than 3 hits and a lot of possible circle parameters from various hit combinations I try to see if there exists one common circle that would fit all the hits. If such mean circle exists, the criterium to attach next hits will be their proximity to this mean circle.

In fact only after this mean circle is found do I try to attach SKEWED STT hits to the track. When I attach them no new information about the circle in xy plane is obtained, only the pure information about z position of the place where skewed stt tube is tangent to the track circle... If there are enough hits in the track (presently more than 3) that carry the z information then I try to obtain the pitch of the track's helix. Again by checking if there is some mean helix pitch from which all the track hits are not too far.

After the main hit loop is finished, I loop over found tracks and get rid of the tracks that had no mean circle nor mean pitch found. They are usually short tracks formed from hits belonging to different MC tracks. This getting rid of tracks consist of moving track hits into the unused hits array, and removing the track from the list.

The last step is to attach the hits from the unused hits array to existing tracks.

The algorithm may be summarized by following simplified C++ code:

```
nofHits = fHitArray->GetEntriesFast();
nofTracks = 0;
nofUnusedHits = 0;
```

```
for ( ihit = 0 ; ihit < nofHits ; ihit++ ) {
   Bool_t hitUsed = kFALSE;
   for ( itr = 0 ; itr < nofTracks ; itr++ )
      if ( MatchHitToTrack(ihit,itr) ) hitUsed = kTRUE;
   if ( hitUsed ) continue;
   for ( iuh = 0 ; iuh < nofUnusedHits ; iuh++ )
      if ( HitMatchesToHit(ihit,iuh) ) hitUsed = kTRUE;
   if ( hitUsed ) continue;
   AddHitToUnusedHits(ihit)
}
RemoveShortTracks();
for ( iuh = 0 ; iuh < nofUnusedHits ; iuh++ )
   for ( itr = 0 ; itr < nofTracks ; itr++ )
      MatchHitToTrack(iuh,itr);

WriteTracks();
```

The tracks are written to the output tree as PndTrack (BarrelTrack) and PndTrackCand (BarrelTrackCand). The last difficulty is to retrieve the momentum information of the track from the helix parameters. For the time, there is some hardcoded values used to make the transformation.

If the explanation is not satisfactory, messy or unclear, please let me know.

yours,
radek

Subject: Re: alternative to LHE tracking
Posted by StefanoSpataro on Fri, 17 Dec 2010 16:01:14 GMT
View Forum Message <> Reply to Message

Thanks for your explanations.
I was looking at your efficiency, which seems being around 85% for "high" momenta, comparable to the lhe one. It is ven true that efficiency depends also on how one defines "efficiency", which could be different from my way or the Gianluigi version.
In lhe case, the code is not optimized for stt, I am wondering if here is room for improvement in your code to reach some higher value, or if maybe the low efficiency is connected to the stt geometry itself (but if I remember well Gianluigi had better values).
I think it would be interesting to check resolution as a function of theta and of momentum. Have you used only prefit values or have you run also the kalman?
Regards

Subject: Re: alternative to LHE tracking
Posted by Radoslaw Karabowicz on Sat, 18 Dec 2010 18:48:26 GMT
View Forum Message <> Reply to Message

Dear Stefano,

I am showing the values as they come after the PndBarrelTrackFinder. I did not yet try the Kalman. From what I remember for the genfit one needs to provide space points, and for the STT I am not creating the PndSttHelixHits, but use the "original" PndSttHits. If my assumption is wrong, please correct it.

As for the definition of efficiency. It is always a problem and I have always saw it problematic. In the PndBarrelTrackFinderQA the efficiency is defined as a ratio of foundTracks to simulatedTracks as a function of MC momentum, theta or phi angles. In the denominator there are all the MC tracks (primaries and secondaries) that had at least 2 hits in MVD, STT, TPC and GEM (the sum counts) - at this very moment I don't know how was it possible for me to set the limit so low, it should be at 4 hits at least. The criterion for calling a track: foundTrack is based purely on momentum comparison. Presently there are no any hits comparisons (which should in fact be done). The reconstructed momentum matches MC momentum, when all three following conditions are satisfied:
- abs(mcMom.Mag-recoMom.Mag) < 0.1*mcMom;
- abs(mcMom.Phi-recoMom.Phi) < 1.5; // in degress, not rads
- abs(mcMom.Theta-recoMom.Theta) < 1.5; // in degress, not rads

In fact I just saw that in the PndBarrelTrackFinderQA there is a bug, and I am not checking for the theta angle... oops. Luckily a quick check and rerun confirms the results I've already presented. I will commit the fix on Monday.

yours,
radek

Subject: Re: alternative to LHE tracking
Posted by Anonymous Poster on Sat, 18 Dec 2010 18:57:47 GMT
View Forum Message <> Reply to Message

Hi Radek,

your assumption that GENFIT needs space point hits is not correct. The STT hits are fitted with the wire position and the drift distance to the wire since a long time. Can an STT expert maybe point out the corresponding reco hit class for Radek?

Cheers, Christian

Subject: Re: alternative to LHE tracking
Posted by StefanoSpataro on Sun, 19 Dec 2010 00:11:52 GMT
View Forum Message <> Reply to Message

If the PndBarrelTrackFinder produces PndTrack objects, then it is possible to use this TCA as input of the PndRecoKalmanTask. This was done for Gianluigi sttmvd pattern recognition, there is no need to write additional code.

Subject: Re: alternative to LHE tracking
Posted by Lia Lavezzi on Sun, 19 Dec 2010 11:21:54 GMT
View Forum Message <> Reply to Message

Quote:The STT hits are fitted with the wire position and the drift distance to the wire since a long time. Can an STT expert maybe point out the corresponding reco hit class for Radek?
Christian is right, the helix hits were necessary for LHE, not for genfit.

To run the Kalman with the PndSttHits, the PndSttRecoHit class (GenfitTools/recohits) with the ctor:
PndSttRecoHit(PndSttHit *currenthit, TClonesArray *tubeArray)
has to be used (where tubeArray is the array of tubes obtained from the map creator).

As Stefano pointed out this is already there in the PndRecoKalmanFit.cxx, so I think you can just run the PndRecoKalmanTask after you PR to have the Kalman results.

Cheers,
Lia.

Subject: Re: alternative to LHE tracking
Posted by Radoslaw Karabowicz on Tue, 21 Dec 2010 13:56:57 GMT
View Forum Message <> Reply to Message

Dear All, you are absolutely right. I have tested it yesterday, and the track fitting with Kalman/genfit works good, but slow:(.

Anyways, I have spotted and fixed two bugs:
- theta comparison now is taken into account when calculating efficiency
- particle charge is now correctly associated to track

I have also created histograms showing momentum resolution as functions of momentum magnitude, phi and theta. The first two dependences look rather OK, but for the smallest theta angles (in the region of GEM) the reconstructed momentum is... well, ugly.

The following plot base on the momentum as taken after the track finder, without fit.
The histograms are 2D, with X axis being the momentum, phi or theta, and Y axis being the momentum resolution. The histogram is then sliced and to each slice a gaussian function is fitted. The sigma of the fit is plotted as the black curve.

The simple change of the macro to include the genfitter:

```
FairGeane *Geane = new FairGeane();
fRun->AddTask(Geane);
PndRecoKalmanTask* recoKalman = new PndRecoKalmanTask();
recoKalman->SetTrackInBranchName("BarrelTrack");
recoKalman->SetTrackOutBranchName("BarrelFitTrack");
fRun->AddTask(recoKalman);
```

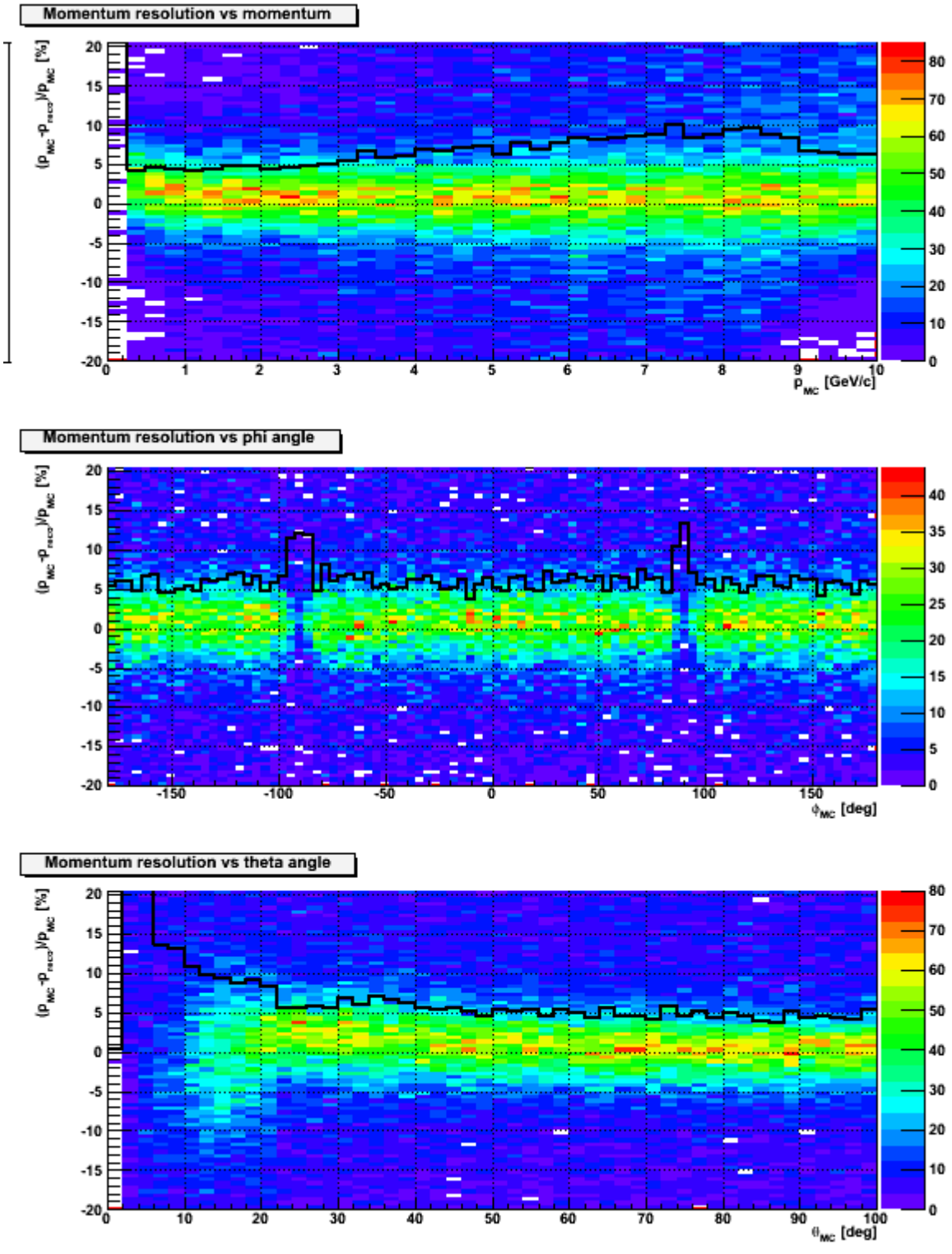
Works and fits the tracks. Below is the histogram showing the reconstructed momentum after PndBarrelTrackFinder (black curve) and after the fit applied (red curve). In this particular case

2 GeV muons were simulated.

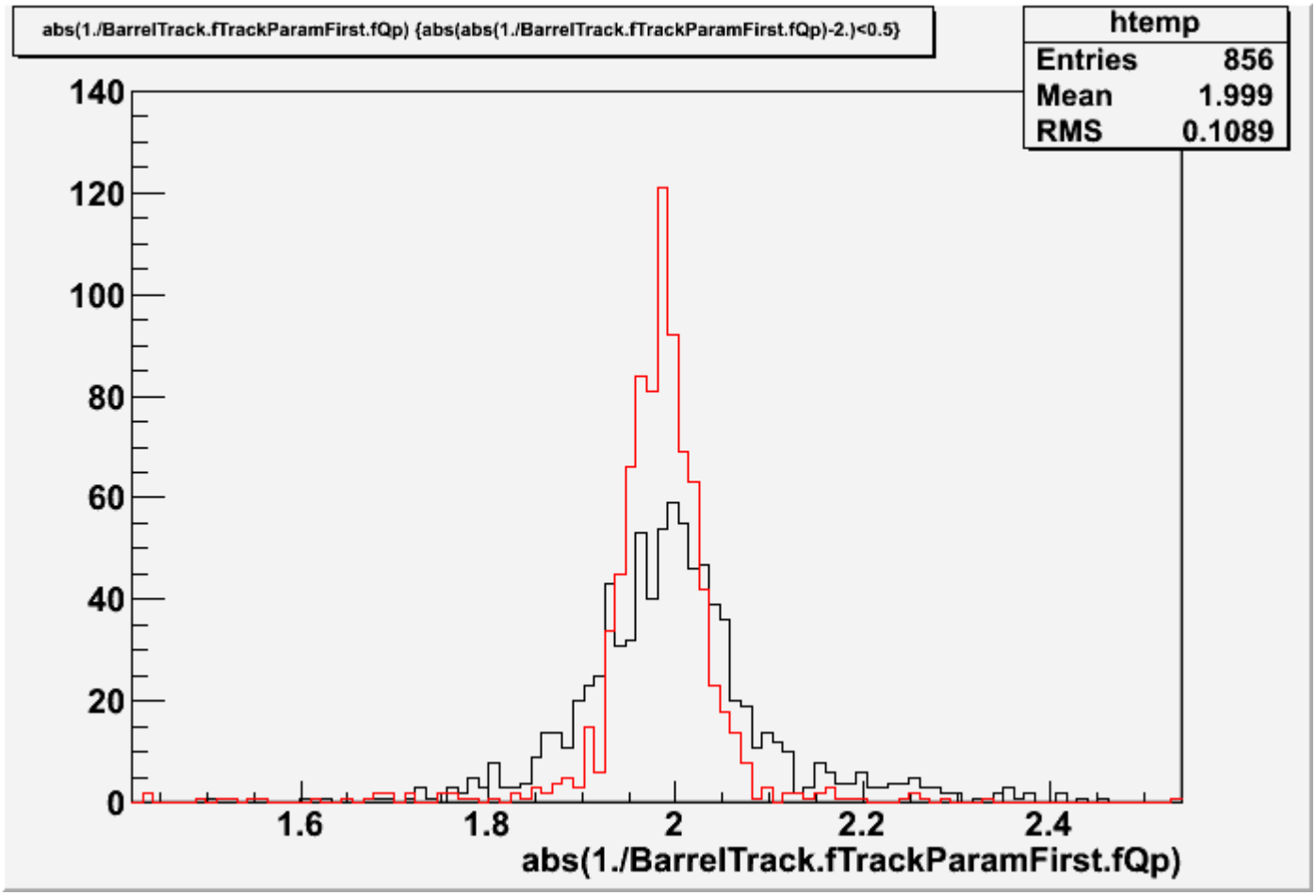
yours
radek

File Attachments

1) resolution2D.gif, downloaded 1320 times



2) fitMom.gif, downloaded 1393 times


abs(1./BarrelTrack.fTrackParamFirst.fQp) {abs(abs(1./BarrelTrack.fTrackParamFirst.fQp)-2.)<0.5}
htemp
Entries 856
Mean 1.999
RMS 0.1089
140
120
100
80
60
40
20
0
1.6
1.8
2
2.2
2.4
abs(1./BarrelTrack.fTrackParamFirst.fQp)