Subject: Kalman low efficiency
Posted by Susanna Costanza on Thu, 02 Jun 2011 10:04:07 GMT
View Forum Message <> Reply to Message

Dear all,
while analysing the data samples of single track events simulated for the STT/TPC comparison, we have noticed something strange: the reconstruction efficiency after the Kalman fit is much lower than the efficiency after the STT+MVD pattern recognition, especially for low momentum tracks.

In the following table, an example of the efficiencies of the STT pattern recognition, the STT+MVD PR and of the Kalman fit are reported:

Muons @ 0.3 GeV/c, theta = 40°
STT alone PR: 87%
STT+MVD PR: 97%
KALMAN : 65%

Muons @ 1 GeV/c, theta = 40°
STT alone PR: 87%
STT+MVD PR: 94%
KALMAN : 89%

The efficiency is calculated as (integral in peak)/(number of generated events), where the peak is defined as the range (mean-3sigma, mean+3sigma).
As you see, the high efficiency of the STT+MVD pattern recognition is reduced after the Kalman.

Two plots are attached to the message, showing the momentum distributions obtained with the STT alone (black), with the STT+MVD reconstruction (red) and after the Kalman (blue), again for muons at 0.3/1 GeV/c and theta = 40°.
As it is shown in the file "03GeV_40deg.pdf", the Kalman produces long tails in the momentum distribution, causing a lower efficiency in peak; these tails are not present in the other two distributions. In addition, the distribution is no more gaussian.
The tails are not present in the Kalman 1 GeV/c distribution; nevertheless, the efficiency is lower than the STT+MVD one also in this case.

Is there something going wrong in the Kalman?
Has someone already noticed this behaviour or has any idea/suggestion?
Is it possible for the detector experts to check if the recohit covariances are correct?

Thank you.
Ciao,
Susanna

File Attachments
1) 03GeV_40deg.pdf, downloaded 502 times
2) 1GeV_40deg.pdf, downloaded 501 times

Subject: Re: Kalman low efficiency

Posted by Anonymous Poster on Thu, 02 Jun 2011 10:19:50 GMT
View Forum Message <> Reply to Message

Hi Susanna,

in the tracks that enter into the plot for the Kalman filter, do you check that the getStatusFlag() of the track rep is giving 0?

Cheers, Christian

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Thu, 02 Jun 2011 10:31:57 GMT
View Forum Message <> Reply to Message

Hi Christian,
I check the flag of the Kalman output track with GetFlag().
If the flag is <0, then the track is not included in the plot.
Isn't it enough?

Ciao,
Susanna

Subject: Re: Kalman low efficiency
Posted by Anonymous Poster on Thu, 02 Jun 2011 10:33:42 GMT
View Forum Message <> Reply to Message

Hi,

what is the call and check you make, exactly?

Christian

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Thu, 02 Jun 2011 11:07:41 GMT
View Forum Message <> Reply to Message

Hi,
for each event, I do the following:

```
for (Int_t k = 0; k < kalfit->GetEntriesFast(); k++) {
  kalTrack = (PndTrack*) kalfit->At(k);
  if (!kalTrack) continue;
  if (kalTrack->GetFlag() < 0) continue;
```

where kalfit is the TClonesArray of PndTracks produced by the Kalman:

```
TClonesArray *kalfit = new TClonesArray("PndTrack");
```

reco->SetBranchAddress("SttMvdGenTrack", &kalfit);

Then, if the track is a primary track (I check the MCTrackId of the corresponding TrackCand), I fill the histogram of the momentum distribution:
hgen->Fill(kalTrack->GetParamFirst().GetMomentum().Mag());

Susanna

Subject: Re: Kalman low efficiency
Posted by Anonymous Poster on Thu, 02 Jun 2011 11:10:29 GMT
View Forum Message <> Reply to Message

Hi,

I see. Could you please check how this PnTrack::GetFlag() relates to the GFTrack::getStatusFlag() of GENFIT?

Cheers, Christian

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Thu, 02 Jun 2011 13:51:19 GMT
View Forum Message <> Reply to Message

Hi,
I had a look into the code.
The PndTrack flag can be filled both in PndRecoKalmanFit and in PndGenfitAdapters, with the following values:
i) -10 if pz=1e-9
ii) -2 if the conversion from GenfitTrack to PndTrack has failed
iii) -1 if the the number of degrees of freedom (NDF) = 0
iv) 1 if NDF != 0

Concerning fStatusFlag, it's a data member of GFAbsTrackRep. I see that in GFKalman it is set to 1 if you catch an exception in processHits(); then there is a continue and you go to the next rep.

So if I understand correctly, the values of these two flags do not seem to be related. Or am I wrong?

Susanna

Subject: Re: Kalman low efficiency
Posted by StefanoSpataro on Thu, 02 Jun 2011 14:14:06 GMT
View Forum Message <> Reply to Message

In the original PndGenfitAdapters::GenfitTrack2PndTrack (Christian was the author) there was

no flag propagation from genfit track to PndTrack.
The flags Susanna is speaking about where put by me in order to check if something had failed in the conversion procedure and to fix eventual bugs.
Therefore, reading the code, GFTrack::getStatusFlag() is not accessible from PndTrack at the moment.

Subject: Re: Kalman low efficiency
Posted by Anonymous Poster on Thu, 02 Jun 2011 15:28:20 GMT
View Forum Message <> Reply to Message

Hi Susanna,

so you you please add GENFIT flag to the PndTrack? The important thing is: status==0 means everything is fine, status!=0 fit failed. Then please apply a cut on this flag and redo the momentum spectrum. I am wondering if the tails get smaller...

Cheers, Christian

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Thu, 02 Jun 2011 16:19:08 GMT
View Forum Message <> Reply to Message

Yes, I can try but not immediately because other tests are running on my machine and I cannot change the code meanwhile.

Anyway, I fear that applying a cut on this flag will influence only the distribution tails (in case) and will not affect at all the problem of the efficiency lowered by the Kalman. Do you have any idea about that?

Ciao,
Susanna

Subject: Re: Kalman low efficiency
Posted by Anonymous Poster on Thu, 02 Jun 2011 16:34:29 GMT
View Forum Message <> Reply to Message

Hi,
at the end of the day we need to know:
-What fraction of tracks can be correctly fitted with GENFIT?
-Why are the tracks that are not fitted not fitted?
-Is it maybe due to false hits picked up in the pattern reco?

The last point you can check by looking at the efficiency after a MC truth pattern reco. If you implement the GEFIT flag, just make sure that the other cases the PndTrack flag can take (pz==1e-9, etc) do not superseed the GENFIT flag. When we isolate for which tracks GENFIT fails, we can try to find out why.

Cheers, Christian

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Mon, 06 Jun 2011 16:29:03 GMT
View Forum Message <> Reply to Message

Hi Christian,
I made the test on the Genfit flag and I found that only in 2 events (out of 10000) the GF status flag is equal to 1 (failure).
So this should not be the reason of the tails...
Concerning the other failures of Genfit, by looking at the flag implemented by Stefano, I found:
- 46 events with flag = -1 (NDF=0)
- 310 events with flag = -2 (GFTrack to PndTrack conversion failed).
As I told you, these events are rejected and do not enter in the plot.

Do you have any suggestion?

Ciao,
Susanna

Subject: Re: Kalman low efficiency
Posted by Anonymous Poster on Mon, 13 Jun 2011 08:21:11 GMT
View Forum Message <> Reply to Message

Hi Susanna,

sorry for not replying earlier, I was out of office...

I see, the GENFIT flag does not explain things. There is one more thing that just came to my mind:
What values for the momentum and position initialization due you use for building the GeaneTrackRep objects that go into GENFIT?

I ask this because I remember a similar behavior in cases where the init pos and mom values were not very good in some test I did in the past. What was happening was that I was basically falling into the left-right ambiguity normal for drift chambers.

Cheers, Christian

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Tue, 14 Jun 2011 09:33:34 GMT
View Forum Message <> Reply to Message

Hi Christian,
the GeaneTrackRep that goes into Genfit is built with the PndTrack that comes out from the STT+MVD pattern recognition, then backpropagated to the point (0,0,0).
The momentum distribution of the PndTracks after the pattern recognition is the red one shown

in the plots attached to the first message: it seems reasonable and without tails.
Then, before going into Genfit, the tracks are backpropagated to (0,0,0) but I don't have any plot for these.

Ciao,
Susanna

Subject: Re: Kalman low efficiency
Posted by Anonymous Poster on Mon, 20 Jun 2011 12:13:19 GMT
View Forum Message <> Reply to Message

Hi,

could you try as a test to initialize the GeaneTrackRep with MC truth values for the position and the momentum. This way we can exclude that there are problems with the seed values for the Kalman fit.

Cheers, Christian

Subject: Re: Kalman low efficiency
Posted by Anonymous Poster on Thu, 30 Jun 2011 08:17:41 GMT
View Forum Message <> Reply to Message

Hi Susanna,

did you find out more about this problem? Did the seed values have anything to do with your problem?

Bye, Christian

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Thu, 30 Jun 2011 08:51:21 GMT
View Forum Message <> Reply to Message

Hi Christian,
unfortunately I have problems with the latest version of the pandaroot, so I couldn't try the test you suggested yet.
As soon as I can perform the test, I will let you know the results.
Ciao,
Susanna

Subject: Re: Kalman low efficiency
Posted by Susanna Costanza on Wed, 06 Jul 2011 10:51:48 GMT
View Forum Message <> Reply to Message

Hi Christian,
I made the test you suggested, initializing the GeaneTrackRep with the MC position and momentum values.
The results are in the plot attached: the red histogram is the input one, with the MC values; the blue one is the Kalman output.
Also with the true values as input, the results are not good...
Any suggestion?

Ciao,
Susanna

File Attachments

1) 300MeV_MCseeds.pdf, downloaded 471 times