Subject: Vertex Fitter in Fast simulation
Posted by donghee on Mon, 31 Mar 2014 13:52:32 GMT
View Forum Message <> Reply to Message

Hi Ralf,

Does vertex fitting make sense in fast simulation?
PndKinVtxFitter for D+ or D- give constant chi2 value as 2e+06 in the fast simulation.
For PndVtxPRG I have same problem.

Best wishes,
Donghee

Subject: Re: Vertex Fitter in Fast simulation
Posted by Ralf Kliemt on Tue, 01 Apr 2014 06:52:50 GMT
View Forum Message <> Reply to Message

Hello Donghee,

Actually you should not try vertex fitting in the FastSim. Everything is already covered by the effective "Vertexing detectors" in the FastSim macro.

Cheers
Ralf

Subject: Re: Vertex Fitter in Fast simulation
Posted by donghee on Tue, 01 Apr 2014 08:28:23 GMT
View Forum Message <> Reply to Message

OK clear

Subject: Re: Vertex Fitter in Fast simulation
Posted by Elisa Fioravanti on Tue, 01 Apr 2014 08:52:29 GMT
View Forum Message <> Reply to Message

Hello,

if the vertex fitting is already covered in the fastsim macro, how can I extract in my macro the information about the vertexing?
For example if I want to see the x,y,z vertex resolutions, what can I do in my macro?

Thanks in advance

Elisa

Subject: Re: Vertex Fitter in Fast simulation
Posted by Ralf Kliemt on Tue, 01 Apr 2014 09:13:13 GMT
View Forum Message <> Reply to Message

Hi Elisabetta,

You should be able to get the position informatin of each track, which is simply smeared by the fsim vertexing.
TVector3 myVtx = myCandidate->GetPosition();

Cheers
Ralf

Subject: Re: Vertex Fitter in Fast simulation
Posted by donghee on Tue, 01 Apr 2014 09:21:22 GMT
View Forum Message <> Reply to Message

Hi,

I have same question, what Elisabetta pointed out.
You suggestion is not correctly working for composited candidate as like D+.
GetPosition() of D+ is everytime equal to zero, no meaning at all.

Let assume
I have three different tracklet for D+ decay into K- pi+ pi+, each composite track has different positions.
After composition to D+ we have total 3 variated positions according smearing of track smearing.

Quote:

```
RhoCandidate *dpk = dp[j]->Daughter(0); //
TVector3 dpVtx1 = dpk->Pos();
RhoCandidate *dppi1 = dp[j]->Daughter(1); //
TVector3 dpVtx2 = dppi1->Pos();
RhoCandidate *dppi2 = dp[j]->Daughter(2);
TVector3 dpVtx3 = dppi2->Pos();
```

where, dp = composite candidate list for D+ -> K- pi+ pi+

Best wishes,
Donghee

Subject: Re: Vertex Fitter in Fast simulation
Posted by Ralf Kliemt on Tue, 01 Apr 2014 09:31:03 GMT
View Forum Message <> Reply to Message

Hi.

Yes, that is correct. Just because all track vertices are smeared you may pick one and use it.

Cheers
Ralf

Subject: Re: Vertex Fitter in Fast simulation
Posted by donghee on Tue, 01 Apr 2014 09:35:40 GMT
View Forum Message <> Reply to Message

Hi Ralf,

I think that is not good way because one of track can have very close to correct positoin, but other one can have more offset than others.
If we select closest one, vertex resolution is too ideal.

I assume that best way should use the POCA approach, which can be accessed by PndPOCA as an alternative.

Best wishes,
Donghee

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Tue, 01 Apr 2014 09:40:07 GMT
View Forum Message <> Reply to Message

If you use always the same particle to take the vertex position, then you will have a consistent distribution and not a systematic deviation. I do not see problems on this.

Subject: Re: Vertex Fitter in Fast simulation
Posted by Ralf Kliemt on Tue, 01 Apr 2014 09:41:35 GMT
View Forum Message <> Reply to Message

Hi Donghee,

That is not what I meant. I think you should stick to one type of tracks, e.g. always the kaon.

Cheers
Ralf

Subject: Re: Vertex Fitter in Fast simulation
Posted by donghee on Tue, 01 Apr 2014 09:58:29 GMT
View Forum Message <> Reply to Message

We have few choice including your suggestion.

1. take a reference track, then get position.

2. Access poca in RhoTupleQA ntuple, which is already stored as "prefix"+pocax(y,z,qa) in your ntuple.

3. Re-do poca by hand as like this where dp[j] is your candidate list.

```
PndVtxPoca POCAvtx;
    TVector3 POCAvertex(0,0,0);
double POCA=-10000;
    double POCAdist=-10000;
    POCA = POCAvtx.GetPocaVtx(POCAvertex,dp[j]);

double poca_x = POCAvertex.);
double poca_y = POCAvertex.Y();
double poca_z = POCAvertex.Z();
```

I think that this way one may avoid ambiguity for choice of reference track.
But disadvantage is that POCA approach cannot be used with neutral composite as like pion0 k_s or Lambda0, unfortunately.

Best wished,
Donghee

Subject: Re: Vertex Fitter in Fast simulation
Posted by Ralf Kliemt on Tue, 01 Apr 2014 10:04:04 GMT
View Forum Message <> Reply to Message

Hi Donghee,

For the sake of transparency of the FastSim I would not use the poca. You can already set the smearing in the fsim macro and you'll know what the resolution in your analysis code will be.

Cheers

Ralf

Subject: Re: Vertex Fitter in Fast simulation
Posted by donghee on Tue, 01 Apr 2014 10:05:17 GMT
View Forum Message <> Reply to Message

Ok clear

Subject: Re: Vertex Fitter in Fast simulation
Posted by Elisabetta Prencipe (2) on Wed, 02 Apr 2014 16:04:17 GMT
View Forum Message <> Reply to Message

Hi Ralf,

I have one more question for today.
In the trunk 24275 now I get an analysis macro (fast-sim). I can check basic plot of 4-momentum, event shape, ... fine.
I want now to use the vertex fitter and then check the mass resolution.

So, I write for Ds-, in the chain pbarp --->Ds- Ds(2536)+:

```
qa.qaComp("Ds", Dslist[j], ntp2);  ///this was already in the analysis macro provided
                                   //with this line I obtain all info on 4-mom and mass

//Here I add info related to the vertex fitter
PndKinVtxFitter vtxfitter(Dslist[j]);
vtxfitter.Fit();
double chi2_vtx=vtxfitter.GetChi2();
if (chi2_vtx>0.){
  RhoCandidate *jfit = Dslist[j]->GetFit();
  TVector3 jVtx=jfit->Pos();
}
qa.qaComp("Ds_vtx_", XXXX, ntp2);
```

What shall I write where I put XXXX? I can't write Dslist[j], otherwise I get exactly same value as without the vertex fitter.
Could you help me, please, to understand how to get this in?

Thank you very much for your help!

Elisabetta

Subject: Re: Vertex Fitter in Fast simulation

Posted by donghee on Wed, 02 Apr 2014 23:08:52 GMT
View Forum Message <> Reply to Message

Hi Elisabetta,

First of all, you don't need to do vertex fitting in the fast simulation. Ralf pointed out the concept for vertexing in fast simulation campaign is different.
Quote:
Actually you should not try vertex fitting in the FastSim. Everything is already covered by the effective "Vertexing detectors" in the FastSim macro.

Vertex fit or 4C whatever any vertexing doesn't make sense in fast simulation study. I confirm that every fit doesn't work in fast simulation except POCA.

Concerning a fitted list.
You want to have a list after doing the vertex/kinematic fitting, probably at full simulation.
Here is a simple way to do.
Quote:

```
RhoCandList Dslist;        //This is a list for Ds
RhoCandList Dslist_fitted; //here define a new list!

for (j=0;j<Dslist.GetLength();++j)
{
PndKinVtxFitter vtxfitter(Dslist[j]);
vtxfitter.Fit();
double chi2_vtx=vtxfitter.GetChi2();
if (chi2_vtx>0.){
RhoCandidate *jfit = Dslist[j]->GetFit();
TVector3 jVtx=jfit->Pos();
Dslist_fitted.Add(jfit); //here you should put your fitted candidate into the new list
}
qa.qaComp("Ds_vtx_", Dslist_fitted, ntp2);
```

You should have fitted information inside the list.
I hope it helps.

Best wishes,
Donghee

Subject: Re: Vertex Fitter in Fast simulation
Posted by Elisabetta Prencipe (2) on Thu, 03 Apr 2014 10:48:00 GMT
View Forum Message <> Reply to Message

Hi Donghee,

thank you for your reply.
Actually my question is not how to address the vertex fitter(s) in the full simulation macro: there I have, and it works.
Different vtx fitters give different mass resolutions. As well as I understood (but I may be wrong) the sense of performing the fast simulation campaign and repeat what we already set up in the release jan14 (with full simulation) is to check differences in efficiency and resolution, so we can decide if the fast simulation is compatible with what we already did with full simulation. Right now I am repeating my simulations (full- and fast-) in the new release. In the case of positive answer from the fast simulation tests, it will be easier to try and check the 4 detector setups proposed in one of the previous computing meeting, using the fast simulation only.

If PANDA likes the comparison in this way, then I go ahead.
My concern now is which vtx fitter is used by default in the fast simulation. In the case of my analysis, it depends which one is used, results can change in a sensitive way. So, what shall I compare, and with what?

My next post will be in the proper forum dedicated to the fast simulation.

cheers, Elisabetta

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Thu, 03 Apr 2014 12:07:14 GMT
View Forum Message <> Reply to Message

In reality the vertex fitters do not change so much the invariant mass resolution, while the kinematic fits do change.
However I suppose at this stage the fitters should not be used, we compare only the values obtained directly from momentum reconstruction.

Subject: Re: Vertex Fitter in Fast simulation
Posted by Elisabetta Prencipe (2) on Thu, 03 Apr 2014 12:11:27 GMT
View Forum Message <> Reply to Message

Hi Stefano,

exactly. Probably I had to specify "kinematic fit, mass constraint fit". So, you say that I have to use (if I do not misunderstood you) _only_ the ftm values of the full simulation. Correct?
If so, I go ahead.

Elisabetta

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Thu, 03 Apr 2014 12:17:10 GMT
View Forum Message <> Reply to Message

I believe so, we are validating our parametrization.
Maybe somebody has some different idea about...

Subject: Re: Vertex Fitter in Fast simulation
Posted by Malte Albrecht on Thu, 03 Apr 2014 15:15:20 GMT
View Forum Message <> Reply to Message

Dear all,

i also have a question concerning kinematic fits (mass constraint fits), I hope this is the right thread to ask:

When we want to produce physics results using the fast sim, we will need a kinematic fit to have a better background suppression and improve the mass resolution.
So, I am not talking about checking the parameterizations or comparison of full/fast simulation, but really to perform simulations of physics channels for the scrutiny process.

How - and which - kinematic fitter can I use in the fast sim, lets say for mass constraint fits? Is the fitter usable in the same way as in the full simulation?

Thanks!

Greetings,
Malte

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Fri, 04 Apr 2014 13:06:23 GMT
View Forum Message <> Reply to Message

Hi,
the fitters should work exactly in the same way in both fast simulation and full reconstruction, the only thing which changes is the pid selection (I presume).

Subject: Re: Vertex Fitter in Fast simulation
Posted by Malte Albrecht on Tue, 15 Apr 2014 22:05:47 GMT
View Forum Message <> Reply to Message

Hi!

I tried to perform 4C- as well as MassConstraint Fits in the fast sim. Using the PndKinFitter, as in Klaus' FastSim example works fine.
The MassConstraintFit does however not work... I tried the following:

```
PndKinFitter etaFit(etaCand);
etaFit.AddMassConstraint(0.547862);
etaFit.Fit();
```

etaCand in this example is a RhoCandidate* composed of two gammas.
The Fit always gives me:
Error in <TDecompLU::DecomposeLUCrout>: matrix is singular
Error in <TDecompLU::InvertLU>: matrix is singular, 0 diag elements < tolerance of 2.2204e-16

chi2 is always zero, and it seems the fit does not do anything...
Is it correct to use the PndKinFitter in that way? Whats going wrong here?

Thanks!
Best regards,
Malte

Subject: Re: Vertex Fitter in Fast simulation
Posted by Klaus Götzen on Wed, 16 Apr 2014 05:51:42 GMT
View Forum Message <> Reply to Message

Hi Malte,

I guess this is because Ralf only implemented the covariance for the tracks so far. Probably it should work in the same with the neutrals, but unfortunately both he and myself are in vacations for the time being until easter. Sorry for that, I didn't think about it either.

We are going to fix it as soon as somebody can take care.

Best,
Klaus

Subject: Re: Vertex Fitter in Fast simulation
Posted by Malte Albrecht on Thu, 17 Apr 2014 17:49:35 GMT
View Forum Message <> Reply to Message

Hi Klaus!

Ah, okay I see.
I looked a bit into fsim/PndFastSim.cxx and tried to understand how the covariances are handled. As far as I see this, smearTrack() is called for each particle from the main Exec()-loop. From there, cutAndSmear() is called - again for each particle. Further on in cutAndSmear() the function SetFlatCovMatrix(t,dp,dtheta,dphi,dE,dV.),dV.Y(),dV.Z()) is called with the appropriate values. Back in the main Exec()-loop the 7x7-covariance matrix (Cov7) is only set for charged particles (first time, that uncharged are cut out).

Wouldnt it be possible, to just call SetFlatCovMatrix with appropriate values for neutrals, and fill the Cov7 then? If so, what are good values for neutral tracks and where to get them from? Maybe you can point me into the right direction...

Best regards and happy easter,
Malte

P.S.: I tried that already with values similar to the ones for charged particles. Technically the mass constraint fit works then (Fitter does not produce any errors any more, and the gamma gamma spectrum shows a needle at the nominal eta-mass. However, the chi2 and pull-distributions look wrong. That must be due to the fact, that I simply used wrong values in the covariance matrix... therefore my question above

---

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Thu, 17 Apr 2014 17:58:55 GMT
View Forum Message <> Reply to Message

Hi Malte,
you are correct. Klaus sent me an email exactly with what you said, I checked the code but I had no time to correct it.
Hopefully I will try tomorrow.

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Fri, 18 Apr 2014 08:07:51 GMT
View Forum Message <> Reply to Message

Dear Malte,
could you please provide me (upload here in the forum) your fastsim and analysis macro, so that I can play and check?

Subject: Re: Vertex Fitter in Fast simulation
Posted by Malte Albrecht on Fri, 18 Apr 2014 12:39:53 GMT
View Forum Message <> Reply to Message

Hi Stefano!

I uploaded my PndScrutAnaTask.cxx and .h, as well as the decay file.
I did not change the simfast.C yet, so you can use the one provided in macro/scrut. The way I called it was (like in the example):
root -l -b -q 'simfast.C("ppbar", "ppbar_phiphieta.dec", 7.0, 10000, "pbarpSystem")'

The PndScrutAnaTask.cxx doesnt do much yet, but in the ntp1 tree you should find the (fitted) variables beginning with "fEta", especially the mass fEtam. Chi2 and pull of the fit are in fchi2Eta and fpullEta.

Hope this helps!

Malte

File Attachments

1) PndScrutAnaTask.cxx, downloaded 521 times
2) PndScrutAnaTask.h, downloaded 509 times
3) ppbar_phiphieta.dec, downloaded 520 times

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Fri, 18 Apr 2014 20:26:20 GMT
View Forum Message <> Reply to Message

Hi,
I think I found the problem.
First, the covariance matrix was not filled for neutrals, as you were saying. Second, in the EMC class the error in Energy was set but not the error in momentum. then all the elements with p were set to zero.
I fixed in my private version of the code, and I can see something reasonable. I wait for the Klaus' OK before committing the code.
Meanwhile, if you wanna try, these are the changes:

```
Index: ../../fsim/PndFsmEmcBarrel.cxx
===================================================================
--- ../../fsim/PndFsmEmcBarrel.cxx (revisione 24552)
+++ ../../fsim/PndFsmEmcBarrel.cxx (copia locale)
@@ -107,12 +107,14 @@
   if (wasDetected && fabs(t->charge())<1e-8)
   {
     result->setdE(dE(t));
+    result->setdp(dE(t));
     result->setdphi(dphi(t));
     result->setdtheta(dtheta(t));
   }
   else
   {
     result->setdE(0.);
+    result->setdp(0.);
     result->setdphi(0.);
     result->setdtheta(0.);
   }
Index: ../../fsim/PndFastSim.cxx
===================================================================
--- ../../fsim/PndFastSim.cxx (revisione 24552)
+++ ../../fsim/PndFastSim.cxx (copia locale)
@@ -465,7 +465,7 @@

       RhoCandidate tcand(ft->p4(),ft->charge(),svtx);

-      if( (fPropagate||fUseFlatCovMatrix) && fabs(ft->charge())>1e-6)
+      if( (fPropagate||fUseFlatCovMatrix))// && fabs(ft->charge())>1e-6)
       {
         tcand.SetCov7(ft->Cov7());
         pidCand->SetCov7( ft->Cov7() );
```

```
Index: ../../fsim/PndFsmEmcBwCap.cxx
===================================================================
--- ../../fsim/PndFsmEmcBwCap.cxx (revisione 24552)
+++ ../../fsim/PndFsmEmcBwCap.cxx (copia locale)
@@ -99,12 +99,14 @@
   if (wasDetected && fabs(t->charge())<1e-8)
   {
     result->setdE(dE(t));
+    result->setdp(dE(t));
     result->setdphi(dphi(t));
     result->setdtheta(dtheta(t));
   }
   else
   {
     result->setdE(0.);
+    result->setdp(0.);
     result->setdphi(0.);
     result->setdtheta(0.);
   }
Index: ../../fsim/PndFsmEmcFwCap.cxx
===================================================================
--- ../../fsim/PndFsmEmcFwCap.cxx (revisione 24552)
+++ ../../fsim/PndFsmEmcFwCap.cxx (copia locale)
@@ -100,12 +100,14 @@
   if (wasDetected && fabs(t->charge())<1e-8)
   {
     result->setdE(dE(t));
+    result->setdp(dE(t));
     result->setdphi(dphi(t));
     result->setdtheta(dtheta(t));
   }
   else
   {
     result->setdE(0.);
+    result->setdp(0.);
     result->setdphi(0.);
     result->setdtheta(0.);
   }
Index: ../../fsim/PndFsmEmcFS.cxx
===================================================================
--- ../../fsim/PndFsmEmcFS.cxx (revisione 24552)
+++ ../../fsim/PndFsmEmcFS.cxx (copia locale)
@@ -100,12 +100,14 @@
   if (wasDetected && fabs(t->charge())<1e-8)
   {
     result->setdE(dE(t));
+    result->setdp(dE(t));
     result->setdphi(dphi(t));
     result->setdtheta(dtheta(t));
   }
   else
   {
```

```
    result->setdE(0.);
+    result->setdp(0.);
    result->setdphi(0.);
    result->setdtheta(0.);
  }
```

Subject: Re: Vertex Fitter in Fast simulation
Posted by Klaus Götzen on Sat, 19 Apr 2014 06:57:48 GMT
View Forum Message <> Reply to Message

Hi Stefano,

great, thanks for taking a look! If you think your results with a e.g. 4C-fit and mass constraint fit look reasonable, of course check in your modifications! Good to know that it works now for neutrals as well!

Best,
Klaus

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Sat, 19 Apr 2014 09:33:04 GMT
View Forum Message <> Reply to Message

I have updated both trunk and scrut14, please try now.

Subject: Re: Vertex Fitter in Fast simulation
Posted by Malte Albrecht on Sat, 19 Apr 2014 12:15:08 GMT
View Forum Message <> Reply to Message

Hi Stefano!

Yes, actually I tried exactly the same in PndFastSim.cxx and I saw that the momentum errors are all set to zero in the covariance matrix, but I did not proceed to the EMC-classes, where you set the dp() correctly now...

OK, I checked out the most recent version and 4C- as well as mass constraint fit are running. Good thing is: When using a mass constraint for e.g. the eta in my channel, I get a needle at the eta-mass (as expected). But the chi2 values are all extremely small (also for 4C-Fit!), which results in a probability of ~0. The pull-distribution is also a needle at 0 (sigma is much too small), so I guess we need to tune the errors a bit?

Best regards and thanks for your efforts!
Malte

Subject: Re: Vertex Fitter in Fast simulation
Posted by StefanoSpataro on Sat, 19 Apr 2014 17:16:27 GMT
View Forum Message <> Reply to Message

Hi,
this "feature" was seen also in the full reco (if you check my "bug&activities" pdf of each software meeting you will find it since months), we were suspecting it was correlated to our tracking covariance matrix, but since it appears also in fast sim, where we set exactly the correct errors, this means that most probably the bug is in the fitter itself.
Since the developer of both the fitters is not active in Panda since years, trying to find the bug will take more time. At present I would suggest to not use the chi2/prob values. Or if you want to check what is happening wrong there...

Subject: Re: Vertex Fitter in Fast simulation
Posted by Klaus Götzen on Sun, 20 Apr 2014 06:35:08 GMT
View Forum Message <> Reply to Message

Hi,

thinking a bit more about the approach of setting dE = dp for neutrals, there is some problem with the way it is done now I think.

Since E and p are 100% correlated quantities, the covariance cannot be diagonal if you set both at the same time. That has to be taken into account right from the beginning. However when the smearing is techincally done in Fast Sim, it is in an uncorrelated way, so that E and p get an individual random modification.

I guess, a proper way would be for neutrals:

- Just set the dE in the detector classes.

- Chose a random smearing base on this dE value.

- Smear both E and p with this value.

- Set the according diagonal elemens for E and p and the correlated term sigma_E,p in covariance matrix before transformation to px, py, pz, E domain.

I'm not really sure whether this is the right way, but one could at least try like this.

Best and happy easter,
Klaus

Subject: Re: Vertex Fitter in Fast simulation

Posted by Malte Albrecht on Mon, 21 Apr 2014 19:28:57 GMT
View Forum Message <> Reply to Message

Hi,

I tried a lot around with this fitting issue today. Unfortunately, it wont work as I want it to
I started off with what Klaus said, and implemented that.

Klaus Goetzen wrote on Sun, 20 April 2014 08:35

Hi,

I guess, a proper way would be for neutrals:

- Just set the dE in the detector classes.

- Chose a random smearing base on this dE value.

- Smear both E and p with this value.

- Set the according diagonal elemens for E and p and the correlated term sigma_E,p in covariance matrix before transformation to px, py, pz, E domain.

I'm not really sure whether this is the right way, but one could at least try like this.

Best and happy easter,
Klaus

- I changed the files, so that just dE is set in the detector classes again.
- Now, I dont see how the smearing of dE and dp should be handled really...
  If you compare the implementations for the smearing of energy and momentum, in case of neutrals there is only a scaling of the three momentum vector components done - both in smearMomentum and smearEnergy!
There is no sense in doing that twice, right? So... what is actually the meaning of smearing p and E seperately - for a neutral? Then it also makes no sense to have the smearing based on the same random number, since actually the same mathematical operation (the scaling) would be done twice. Please correct me if Im wrong...

Anyway, I still tried it. The results dont change (chi2 is ~0), everything looks approximately the same. Then I introduced the off-diagonal elements to the polar covariance matrix.
Am I correct, that in case of neutrals the diagonal elements, as well as the off-diagonals should contain dE*dE then?
When I put that into the covariance matrix, it seriously breaks things... the gammagamma mass spectrum looks like a barcode afterwards So, that is apparently not correct.

Any suggestions?

Best regards,
Malte

Subject: Re: Vertex Fitter in Fast simulation
Posted by Ralf Kliemt on Tue, 22 Apr 2014 14:16:10 GMT
View Forum Message <> Reply to Message

Hi,

I second the opinion that in the EMCs shall be only dE, because for neutrals the momentum is then calculated in the smearing function.

Furthermore I checked that in no detector class of the fastsim momentum and energy smearing are used simultaneously. So for the purpose of the neutrals covariance I check for the charge and use a momentum-smearing based cov for charged particles and an energy-smearing based cov for neutrals.

I still get reasonable pulls for the final state articles in the standard pion/electron sample from the scrut/ folder. Using the macros above, I find good pulls for the gammas, too. The pulls for the Eta and your 4C fit look good, too.

Less problems.

Now one issue lies with the energies of charged Kaons and I still have to understand that.
More Problems.

Cheers
Ralf

Subject: Re: Vertex Fitter in Fast simulation
Posted by Malte Albrecht on Tue, 22 Apr 2014 17:41:33 GMT
View Forum Message <> Reply to Message

Hi Ralf,

thanks for your changes today, now my pulls and chi2 / Prob histograms look good. Actually what did the trick, are your changes to the calculation of the jacobian and the covPol, right? Anyway, I did just a quick check, but 4C and mass constraint fits seem to be operational now.

Now one more question: How are staged fits handled? If I want a mass constraint fit (e.g. for an eta) and after that a 4C fit using the already pre-fitted 4vector of my mass-constrained-eta (which would then actually be a 5C-fit, but ok...) ?
From other fitters I just know, that one can add MassConstraints, or FourVector-constraints as many as one likes, but in the end one can still fit everything _together_ with one fitter object. Can I do something similar here, or do I always have to create new RhoCandidates and give them to the next fitter?

Greetings,
Malte

Subject: Re: Vertex Fitter in Fast simulation
Posted by Ralf Kliemt on Wed, 23 Apr 2014 08:01:34 GMT
View Forum Message <> Reply to Message

Hi,

Yes, the jacobian is now from (p,theta,phi) to (px,py,pz,e) for charged particles and from (e,theta,phi) to (px,py,pz,e) for gammas.

You may want to try PndKinFitter and set more constraints. No warrenty here. However, our kinematic fitters fit the input node of the decay with all the final state particles. No intermediate decay is considered.
To circumvent that you can apply Lock() (and UnLock()) to a composite candidate. The fitters will then treat it as a final state particle. You'd be the first one to apply that feature. See RhoFitterBase for details.

Cheers
Ralf