Subject: [FIXED] Problem with PndVertexFitter for particles with neutral charge
Posted by Jennifer Pütz on Fri, 29 May 2015 08:44:07 GMT
View Forum Message <> Reply to Message

Hi everyone,

anti-lambda0 decays into an anti-proton and a pi+.

If I combine the proton and the pi- to the lambda0 the vertex fitter is working fine. Same for the antilambda0. But when I combine lambda0 and pi- to Xi- the vertex fitter does not work.

The vertex fitter is working fine for combining proton and pi- to lambda0. But if I combine lambda0 and anti-lambda0 to the pbar p system, here again the vertex fitter does not work.

I checked the 4-Mom-Constraint Fitter and it is working.

Is there already a solution for this problem?

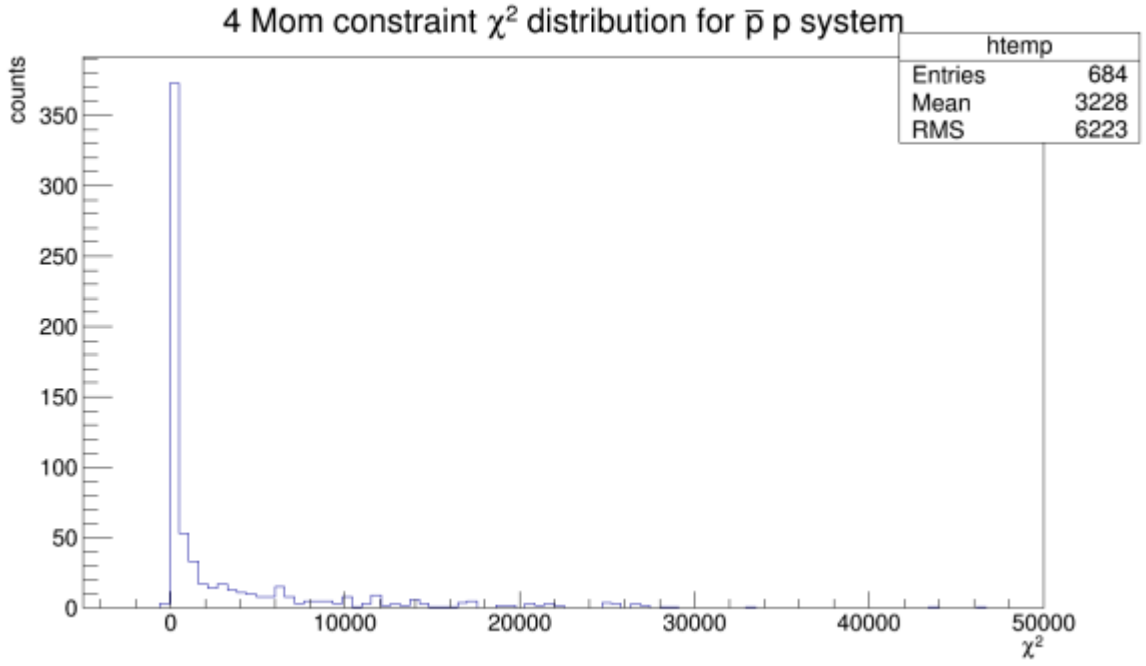
I'm using on my system:

Fairsoft Version: March 15
FairRoot: Master
PandaRoot: Trunk Rev. 27718

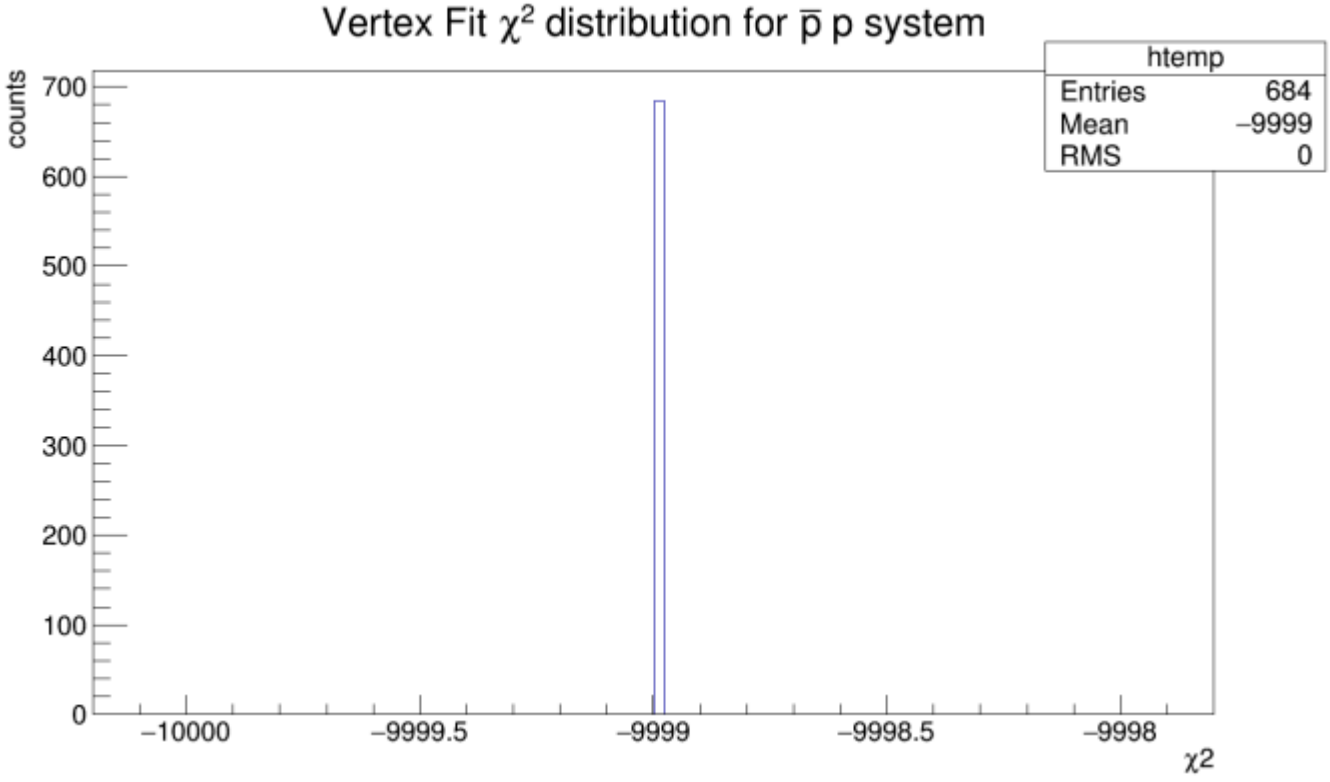
Best
Jenny

File Attachments

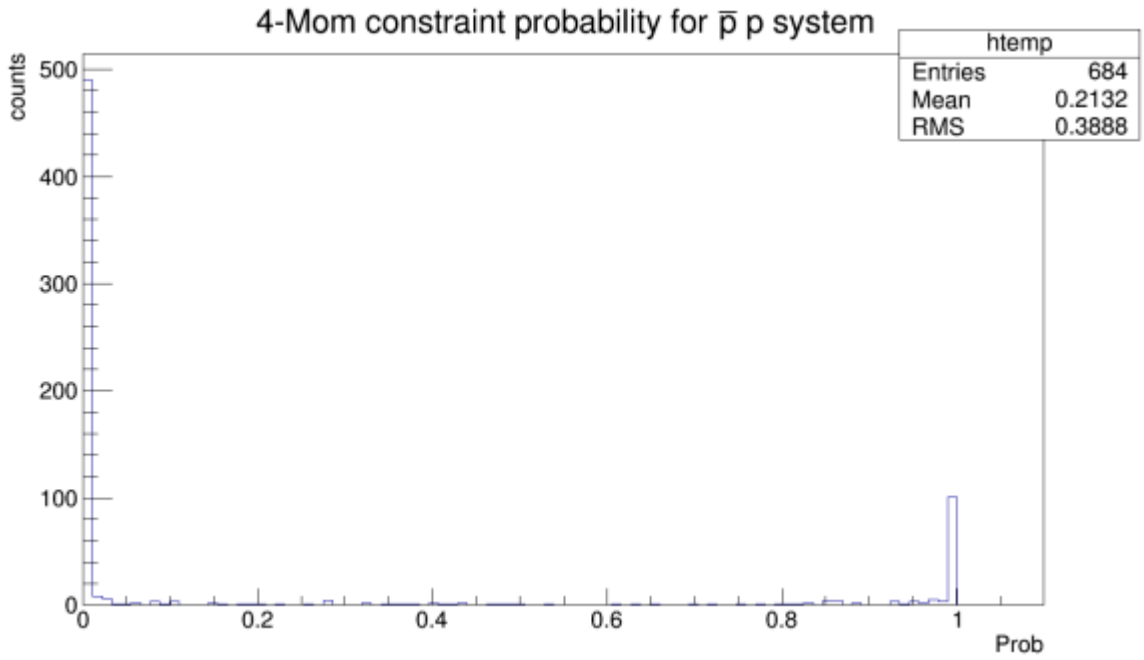
1) 4CFit_chi2.png, downloaded 805 times



2) VertexFit_chi2.png, downloaded 865 times



3) 4CFit_prob.png, downloaded 869 times



Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by StefanoSpataro on Fri, 29 May 2015 09:20:29 GMT
View Forum Message <> Reply to Message

Can you write the piece of code where you are doing such vertex fitting?

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by Jennifer Pütz on Fri, 29 May 2015 09:24:25 GMT
View Forum Message <> Reply to Message

Hi Stefano,

for the vertex fit I'm using:

```
//*** Cross check: pbar + p -> Lambda0 + AntiLambda0

crossCheck.Combine(lambda0, antiLambda0);
crossCheck.SetType(88888);


for (int j=0; j<crossCheck.GetLength(); ++j){

    //do vertex fit
    PndKinVtxFitter vertexFitter_cc (crossCheck[j]);
    vertexFitter_cc.Fit();
    RhoCandidate * ccFit = crossCheck[j]->GetFit();

}
```

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by StefanoSpataro on Fri, 29 May 2015 09:26:06 GMT
View Forum Message <> Reply to Message

lambda0 and antiLambda0?

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by Jennifer Pütz on Fri, 29 May 2015 09:37:08 GMT
View Forum Message <> Reply to Message

Ok, sorry I should have give you some more information

```
theAnalysis->FillList(piminus, "PionAllMinus", "PidAlgoIdealCharged");
theAnalysis->FillList(piplus, "PionAllPlus", "PidAlgoIdealCharged");
theAnalysis->FillList(proton, "ProtonAllPlus", "PidAlgoIdealCharged");
theAnalysis->FillList(antiProton, "ProtonAllMinus", "PidAlgoIdealCharged");


//***Lambda0 -> PiMinus + Proton
lambda0.Combine(piminus,proton);
lambda0.SetType(3122);



//***AntiLambda0 -> PiPlus + AntiProton
antiLambda0.Combine(piplus,antiProton);
antiLambda0.SetType(-3122);


//*** Cross check: pbar + p -> Lambda0 + AntiLambda0

crossCheck.Combine(lambda0, antiLambda0);
crossCheck.SetType(88888);


for (int j=0; j<crossCheck.GetLength(); ++j){

//do vertex fit
PndKinVtxFitter vertexFitter_cc (crossCheck[j]);
vertexFitter_cc.Fit();
RhoCandidate * ccFit = crossCheck[j]->GetFit();

}
```

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by StefanoSpataro on Fri, 29 May 2015 09:41:09 GMT

View Forum Message <> Reply to Message

Don't you have to first vertex fit the lambdas, before giving them into the global fitter?

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by Jennifer Pütz on Fri, 29 May 2015 09:47:12 GMT
View Forum Message <> Reply to Message

I do first vertex fit lambdas. I just leave it out here.

```
for (int j=0; j<lambda0.GetLength(); ++j){
    PndKinVtxFitter vertexfitterLambda0 (lambda0[j]);
    vertexfitterLambda0.Fit();
    RhoCandidate * lambda0Fit = lambda0[j]->GetFit();
}

for (int j=0; j<antiLambda0.GetLength(); ++j){
      PndKinVtxFitter vertexfitterAntiLambda0 (antiLambda0[j]);
      vertexfitterAntiLambda0.Fit();
      RhoCandidate * antiLambda0Fit = antiLambda0[j]->GetFit();
}
```

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by StefanoSpataro on Fri, 29 May 2015 09:57:11 GMT
View Forum Message <> Reply to Message

In this case maybe you should fit lambda0Fit and not lambda0. Just trying to guess, Klaus and Ralf are the experts.

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by André Zambanini on Fri, 29 May 2015 10:56:05 GMT
View Forum Message <> Reply to Message

I actually have the same problem with a non-working vertex fitter combining a (anti-)lambda with some other particle. If I remember well, Dariusch saw the same.

Other than Jenny, I'm using the particle candidate after the vertex fitting as you just suggested. So my guess is that this is not the issue.

Subject: Re: Problem with PndVertexFitter for particles with neutral charge
Posted by Ralf Kliemt on Fri, 29 May 2015 12:27:02 GMT
View Forum Message <> Reply to Message

Hi,

The vertex fitters (PndKinVtxFitter/PndKalmanVtxFitter) operate on a minimum of two charged tracks. For anything else you may use PndVtxPoca, which analytically calculates the vertex. From the top of my head:

```
PndVtxPoca pocafinder;
for (int j=0; j<lambda0.GetLength(); ++j)
{
  PndKinVtxFitter vertexfitterLambda0 (lambda0[j]);
  vertexfitterLambda0.Fit();
  RhoCandidate * lambda0Fit = lambda0[j]->GetFit();
  for (int j=0; j<antiLambda0.GetLength(); ++j)
  {
    PndKinVtxFitter vertexfitterAntiLambda0 (antiLambda0[j]);
    vertexfitterAntiLambda0.Fit();
    RhoCandidate * antiLambda0Fit = antiLambda0[j]->GetFit();

    TVector3 pbarpvertex;
    crossCheck.Combine(lambda0Fit, antiLambda0Fit);
    crossCheck.SetType(88888);

    double doca = pocafinderGetPocaVtx(pbarpvertex,crossCheck);
    // now you have a vertex written to pbarpvertex and a measure of the distances between the
trajectories (i.e. a quality criterium)
  }
}
```

If you have charged(!) particles combining to crossCheck, you may want to update the four-momentum along the helix to the vertex via the analysis:

```
bool check = theAnalysis->PropagateToPoint(crossCheck, pbarpvertex);
```

Cheers
Ralf