Subject: box generator
Posted by asanchez on Wed, 16 May 2007 15:24:22 GMT
View Forum Message <> Reply to Message

hi I want to use the box generator
to simulate one uniforme momentum distribution
my detector is at a Z-distance -76.5 cm
from the center of Panda.

I do the following selection

```
boxGen->SetPRange(.1,1.); // GeV/c
 // boxGen->SetPtRange(1.,1.); // GeV/c
 boxGen->SetPhiRange(0., 360.); // Azimuth angle range [degree]
 boxGen->SetThetaRange(0., 180.); // Polar angle in lab system range [degree]
 boxGen->SetXYZ(0.,10., -760.); // vertex coordinates [mm]
 primGen->AddGenerator(boxGen);
```

but I get any hit in the detector.
is it something wrong with
the parameters.
thanks a lot
alicia

Subject: Re: box generator
Posted by StefanoSpataro on Mon, 21 May 2007 06:25:43 GMT
View Forum Message <> Reply to Message

Hi Alicia,
I have the feeling that the position should be expressed in cm, and not in mm (yes I know, even in my Frascati presentation it was written wrong).
Try to take out one order of magnitude and see if it will work after.

Subject: Re: box generator
Posted by Pablo Genova on Mon, 21 May 2007 07:54:08 GMT
View Forum Message <> Reply to Message

Hi Stefano and Alicia,

one warning on the box generator: is it really uniform in theta? As Kasha and Felice Iazzi noticed the p components are generated uniformly in phi and in theta,
not uniformly in phi and cos(theta).

Do you agree that there is something strange?

ciao, Pablo

The code is the following from CmbBoxGenerator.cxx:

```
// Generate particles
  for (Int_t k = 0; k < fMult; k++) {
    phi = gRandom->Uniform(fPhiMin,fPhiMax) * TMath::DegToRad();

    if      (fPRangeIsSet ) pabs = gRandom->Uniform(fPMin,fPMax);
    else if (fPtRangeIsSet) pt   = gRandom->Uniform(fPtMin,fPtMax);

    if      (fThetaRangeIsSet) {
      theta = gRandom->Uniform(fThetaMin,fThetaMax) * TMath::DegToRad();
    }
    else if (fEtaRangeIsSet) {
      eta   = gRandom->Uniform(fEtaMin,fEtaMax);
      theta = 2*TMath::ATan(TMath::Exp(-eta));
    }
    else if (fYRangeIsSet) {
      y     = gRandom->Uniform(fYMin,fYMax);
      mt = TMath::Sqrt(fPDGMass*fPDGMass + pt*pt);
      pz = mt * TMath::SinH(y);
    }

    if (fThetaRangeIsSet || fEtaRangeIsSet) {
      if      (fPRangeIsSet ) {
 pz = pabs*TMath::Cos(theta);
 pt = pabs*TMath::Sin(theta);
      }
      else if (fPtRangeIsSet)
 pz = pt/TMath::Tan(theta);
    }

    px = pt*TMath::Cos(phi);
    py = pt*TMath::Sin(phi);

    if (fBoxVtxIsSet) {
      fX = gRandom->Uniform(fX1,fX2);
      fY = gRandom->Uniform(fY1,fY2);
    }
```

Subject: Re: box generator
Posted by StefanoSpataro on Mon, 21 May 2007 08:05:13 GMT
View Forum Message <> Reply to Message

Hello,
the aim of the "uniform theta distribution" function is to have a uniform distribution in theta (which means same number of counts per each theta bin), not in cos(theta).

I mean, the uniform distribution in cos(theta) is another different thing, that probably can be added under request if needed.

So I think there is nothing strange.

Subject: Re: box generator
Posted by Pablo Genova on Mon, 21 May 2007 08:30:30 GMT
View Forum Message <> Reply to Message

Hi Stefano,

so we misunderstood the aim of the box generator.

The box generator is not a generator which generates uniformly on the solid angle, but it generates uniformly on theta, phi angles.

I mean, if one wants to generate isotropically in 3D with a fixed total momentum, or fixed transverse momentum, which is a common request, one has to generate uniformly on cos\theta and \phi and the cbmbox generator is NOT correct for this.

I thought it were!

I think it is really useful to have a 3d uniform generator not to be confused with the present box generator.

I do not understand completely the need for uniform generation on theta and phi variables, but this is not important.

ciao, Pablo

Subject: Re: box generator
Posted by StefanoSpataro on Mon, 21 May 2007 14:32:37 GMT
View Forum Message <> Reply to Message

Well,
it could be possible to add a function SetCosThetaRange() (or something similar), in order to have what you ask without adding a new generator.

Subject: Re: box generator
Posted by Pablo Genova on Tue, 22 May 2007 07:14:20 GMT
View Forum Message <> Reply to Message

Yes for me it's OK.

It could be useful, in order to prevent errors, to explain in the comments that, for uniform generation on the solid angle, one has to call setphirange and setcosthetarange and not settheta.

Maybe also a function setuniform() which automatically sets the whole (phi, theta) range and

generates uniformly on the solid angle could be useful.

ciao, Pablo

Subject: Re: box generator
Posted by Katarzyna Szymanska on Tue, 22 May 2007 11:01:40 GMT
View Forum Message <> Reply to Message

Hi Stefano,

Few remarks concerning the direction generation:

1) the possible distributions of the (physically meaningful) directions are infinite: the uniform direction is the simplest, corresponds to the s-wave in the two-body kinematics and is mostly used to evaluate the relative acceptances and efficiencies at different solid angles

2) therefore, we think that the collaboration needs such a distribution which is generated by -1 < cos(theta) < +1 and 0 < phi < 360, both uniform.

3) neverthless, if someone uses the "uniform theta" distribution, which should peak the directions forward, one has the remember that the relative and absolute acceptances, efficiencies must be suitable normalized.

4) in any case, it will be useful to comment clearly the "uniform theta" distribution and its physical meaning in term of directions.

Regards,

Felice and Kasia

Subject: Re: box generator
Posted by Pablo Genova on Thu, 24 May 2007 17:43:18 GMT
View Forum Message <> Reply to Message

Hi Stefano and generators' people,

triggered also by Katarzyna and Felice, I checked the two generators, the former CbmBoxGenerator and the PndBoxGenerator with option SetCosTheta(), which Stefano wrote recently.

As they suggested, I used a detector consisting of a sphere, everything in vacuum.

You can see the results in the attached plots

1. box_NON_uniform.jpg : with uniform theta.
2. box_uniform.jpg : with uniform cos theta.

As expected the first plot shows enhanced momenta in region x=0 & y=0, i. e. along the zed axis , and only using the SetCosTheta option you get the truly uniform generation.

So everything is OK, but I would strongly suggest to put a comment a like: "if you want to generate uniformly in the solid angle use the SetCosTheta() function" or even call that function SetUniform3D() , to avoid errors or misunderstandings.

Maybe even in the macros it is better to default to uniform distributions, with suitable comment to clear this to the user.

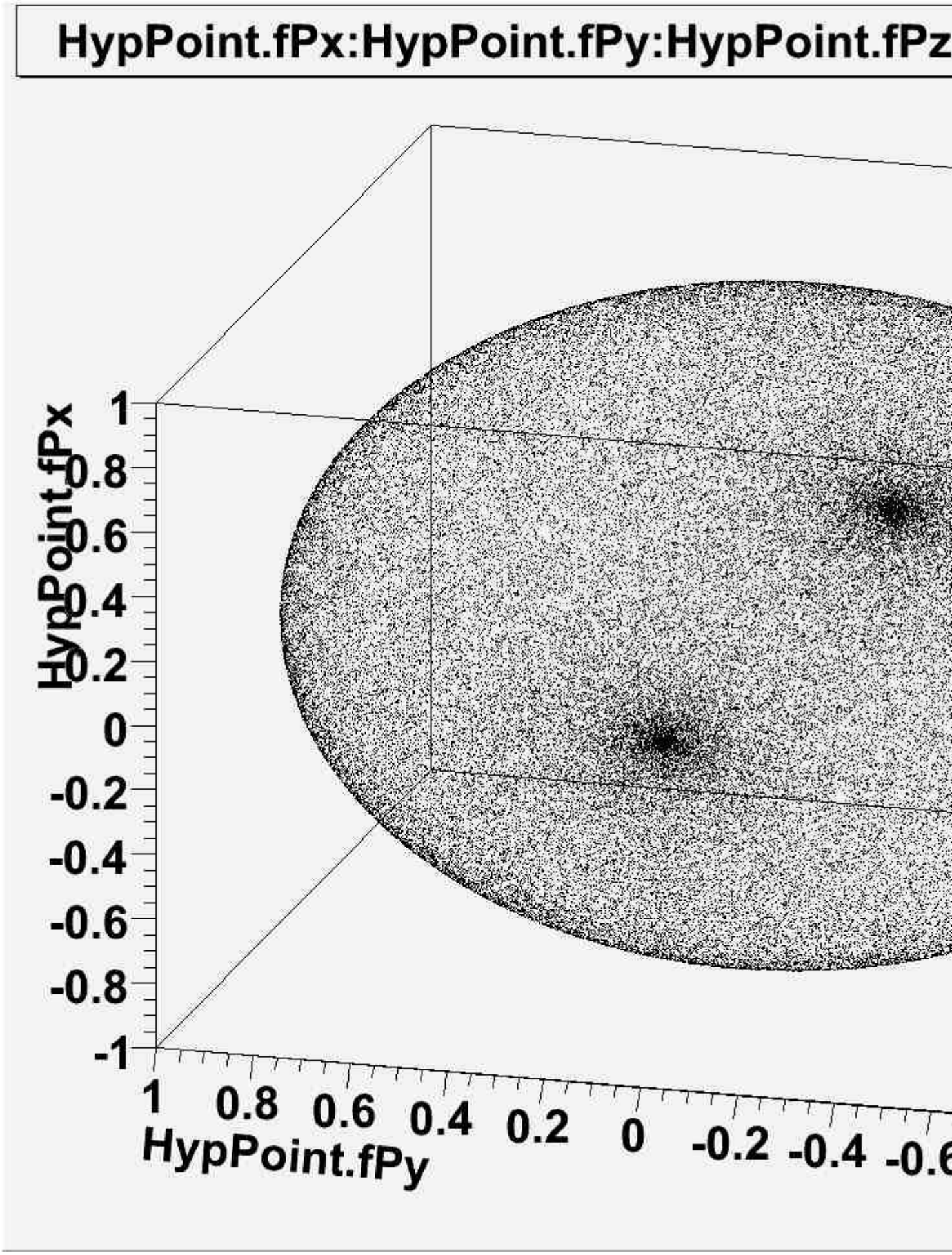
(I went into the mistake of supposing uniform what uniform was not!)

ciao, Pablo

ps: this forum is wonderful!

File Attachments

1) box_NON_uniform.jpg, downloaded 3760 times


HypPoint.fPx:HypPoint.fPy:HypPoint.fPz
HypPoint.fPx
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
-0.6
-0.8
-1
1
0.8
0.6
0.4
0.2
0
-0.2
-0.4
HypPoint.fPy

2) box_uniform.jpg, downloaded 3764 times