PSF evaluation using full Bremsstrahlung data and MC

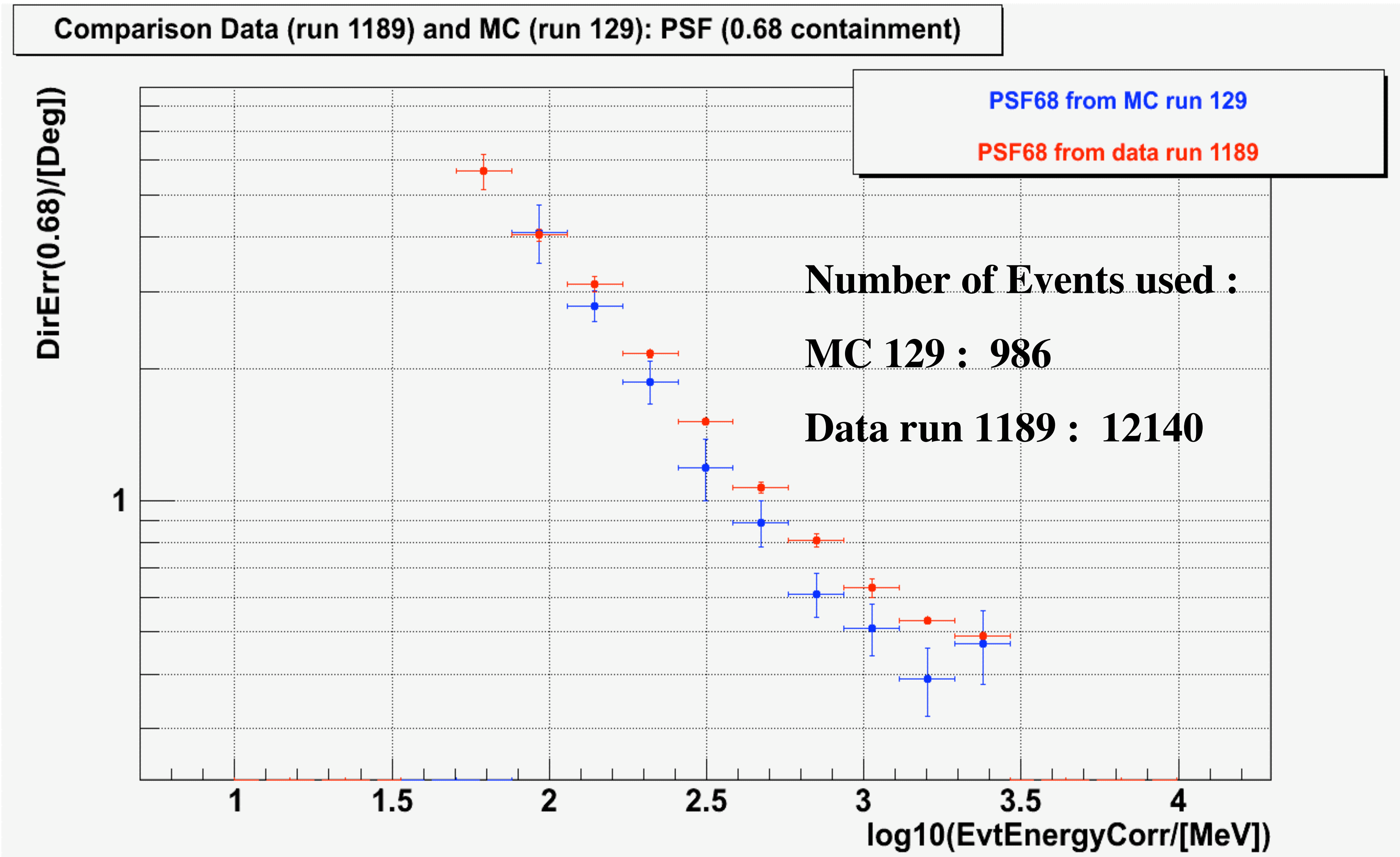
Some time ago I tried to evaluate the agreement data-mc in what concerns to the angular resolution (PSF)

https://confluence.slac.stanford.edu/download/attachments/13893/QuickInsectionPSFWithFullBrems_2.pdf?version=1

The plan was to repeat (AND IMPROVE !!) these studies with customized MC runs

Evaluate the reliability of the quantities Sqrt(VtxS[XX,YY]) as errors of the quantities Vtx[X,Y]Dir (suggestion from Hiro Tajima)

Comparison of PSF performed using Data run 1189 and MC run 129
(From previous studies)



Estimation of the beam charactersitics show some differences …

Data

```
  Beam characteristics for RUN NUMBER  700001189
*****************************************************************

Beam incoming direction (cosinus directors):
       Tkr1ZDir =    -0.9992300746
       Tkr1XDir =   0.004465714435
       Tkr1YDir =    0.00228261912

Beam impact point on Calorimeter input (CalZ =  -47 mm):
      PosXAtCalZ (mm) =      208.0424863
      PosYAtCalZ (mm) =        8.4514553

Beam width (Sigma) estimated from projected beam width (X,Y)
on the first hit height, and incoming beam direction:
Beam Width (Sigma) in X direction (mm) = 14.29745645
Beam Width (Sigma)  in Y direction (mm) = 20.50043036

Upper limit for Beam divergence:
          CosMaxBeamDivergence() =      0.998685026
      MaxBeamDivergence(degrees) =      2.882293178
```

MC

```
  Beam characteristics for RUN NUMBER  129
*****************************************************************

Beam incoming direction (cosinus directors):
       Tkr1ZDir =    -0.9996026899
       Tkr1XDir =  0.0004366524459
       Tkr1YDir =  0.0003937221364

Beam impact point on Calorimeter input (CalZ =  -47 mm):
      PosXAtCalZ (mm) =      201.8669243
      PosYAtCalZ (mm) =      14.77194864

Beam width (Sigma) estimated from projected beam width (X,Y)
on the first hit height, and incoming beam direction:
Beam Width (Sigma) in X direction (mm) = 16.46691336
Beam Width (Sigma)  in Y direction (mm) = 13.83124008

Upper limit for Beam divergence:
          CosMaxBeamDivergence() =     0.9993201585
      MaxBeamDivergence(degrees) =      2.058589367
```

Unluckily, so far I have not been able to generate proper MC runs …
Work in progress…

Yet I would like to discuss a problem I found then, which I think it is still not solved…

Beam dispersion between several MC runs seem to give contradictory results…

That is very important if we want to compute PSF and compare with Full Brems photons from MC

0 - Full Brems data split into several energy bins

Logarithmic binning used: **23 bins in range 0.120-4.170**

Bin width increases by 50 % (suggested by Gary)

Description of bins in linear scale:

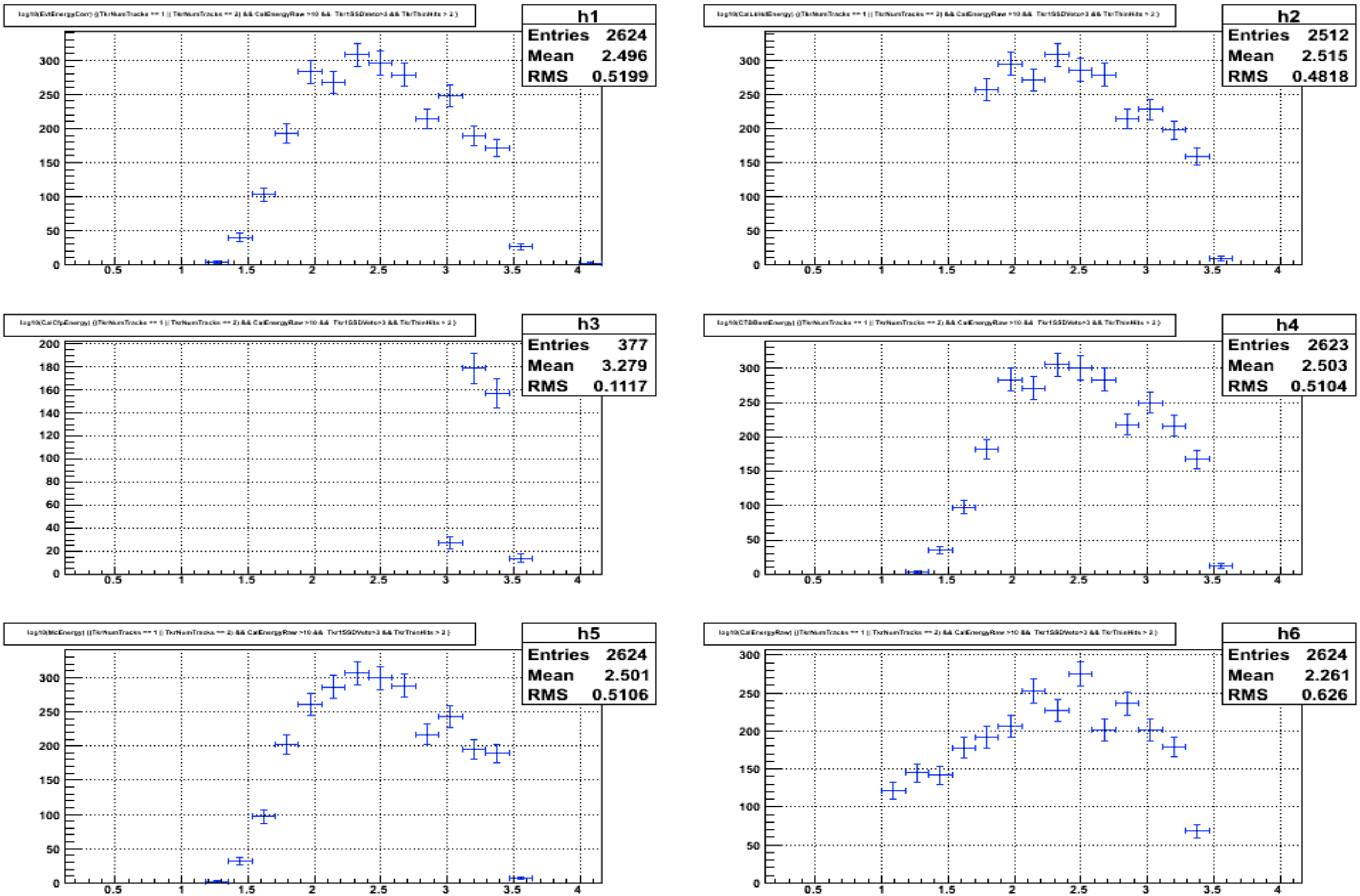
```
bin  1; 1.31687 - 1.97531 : Bin Width = 0.658436
bin  2; 1.97531 - 2.96296 : Bin Width = 0.987654
bin  3; 2.96296 - 4.44444 : Bin Width = 1.48148
bin  4; 4.44444 - 6.66667 : Bin Width = 2.22222
bin  5; 6.66667 -      10 : Bin Width = 3.33333
bin  6;      10 -      15 : Bin Width =       5
bin  7;      15 -    22.5 : Bin Width =     7.5
bin  8;    22.5 -   33.75 : Bin Width = 11.25
bin  9; 33.75 - 50.625 : Bin Width = 16.875
bin 10; 50.625 - 75.9375 : Bin Width = 25.3125
bin 11; 75.9375 - 113.906 : Bin Width = 37.9687
bin 12; 113.906 - 170.859 : Bin Width = 56.9531
bin 13; 170.859 - 256.289 : Bin Width = 85.4297
bin 14; 256.289 - 384.434 : Bin Width = 128.145
bin 15; 384.434 - 576.65 : Bin Width = 192.217
bin 16; 576.65 - 864.976 : Bin Width = 288.325
bin 17; 864.976 - 1297.46 : Bin Width = 432.488
bin 18; 1297.46 - 1946.2 : Bin Width = 648.732
bin 19; 1946.2 - 2919.29 : Bin Width = 973.098
bin 20; 2919.29 - 4378.94 : Bin Width = 1459.65
bin 21; 4378.94 - 6568.41 : Bin Width = 2189.47
bin 22; 6568.41 - 9852.61 : Bin Width = 3284.2
bin 23; 9852.61 - 14778.9 : Bin Width = 4926.31
```

Selection of events applied (Events converted in thin layers)

(TkrNumTracks == 1 || TkrNumTracks == 2) &&
CalEnergyRaw >10 && Tkr1SSDVeto>3 && TkrThinHits > 2

Distributions of McEnergy, CalEnergyRaw and Reconstructed energies

MC Run 125 (PSF will be computed using events from each of these bins)



1 - PSF (from MCDirErr) for each of these energy bins (68% and 95%) (MC runs 125, 127, 129, 130)

Two PSF are computed, the one which contains 68% and the one which contains 95% of the events.

Selection of events applied (Events converted in thin layers)

(TkrNumTracks == 1 || TkrNumTracks == 2) &&
CalEnergyRaw >10 && Tkr1SSDVeto>3 && TkrThinHits > 2

*PSF; Position at which IntegratedNumOfEvents = Fraction*NumEvents*

Where fraction is 0.68 and 0.95

It also computes an error for each of the PSFs. Arbitrary definition:
PosHelp; Position at which

*IntegratedNumOfEvents = Fraction*NumEvents*+Sqrt(N*fraction*(1-fraction))

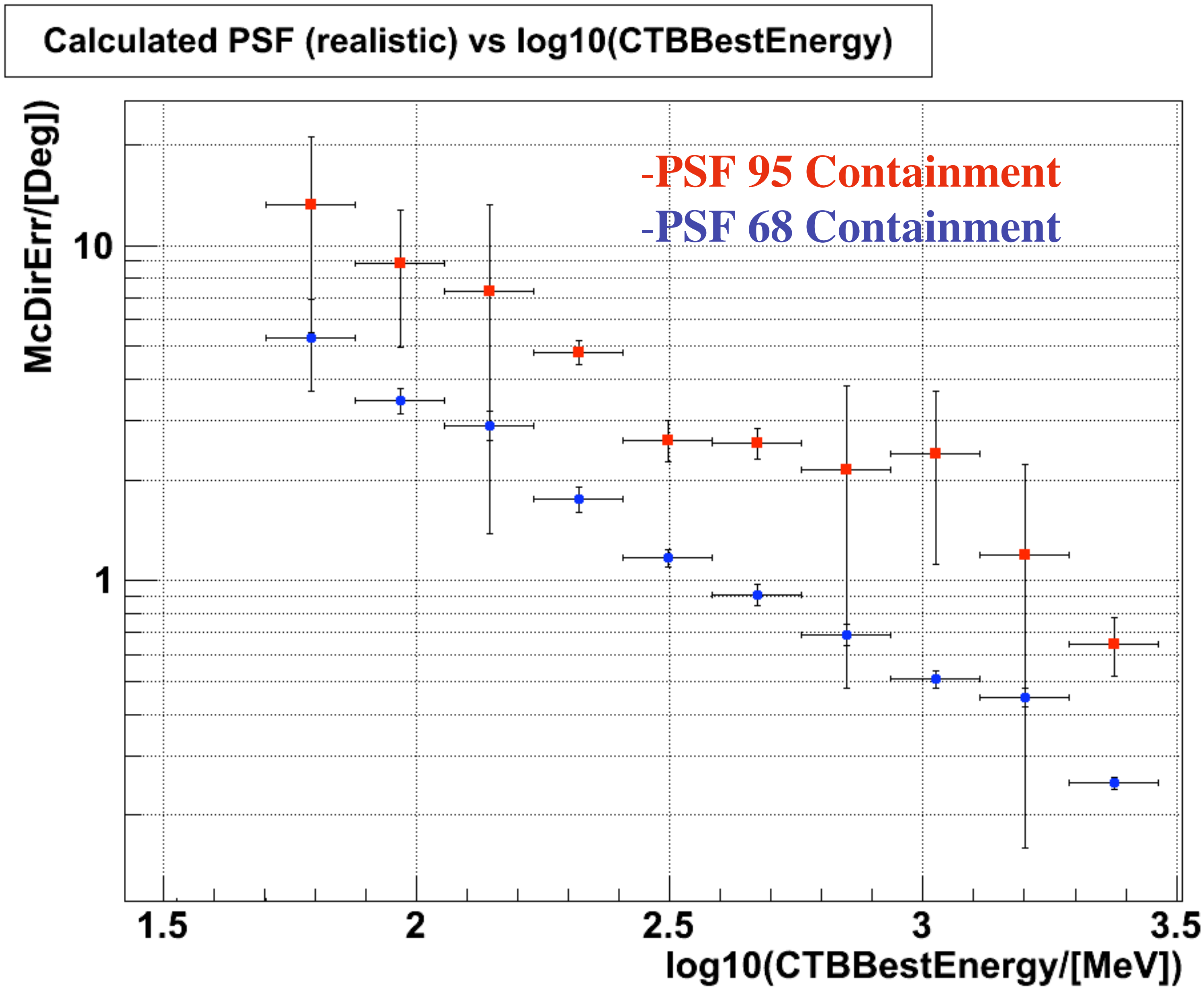
PSFErr = PosHelp-PSF

With this definition, the magnitude of this error depends on:

1. **The number of events in that particular energy bin**
2. **Shape of the distribution of McDirErr**

1 - PSF (from MCDirErr) for each of these energy bins (68% and 95%)

MC runs 125 (0 incidence angle)



1 - PSF calculation using the beam direction

ReconstructedDirectionVector *= VtxXDir, VtxYDir, VtxZDir*

IncomingPhotonDirectionVector *= McXDir,McYDir, McZDir*
BeamDirectionVector *= cos(XthetaBeam), cos(YthetaBeam), cos(ZThetaBeam)*

I can use 3 DirErr s: **McDirErr**, **MyDirErr** and **BeamDirErr:**

Cos(MyDirErr) = McXDir* VtxXDir +
McYDir* VtxYDir +
McZDir* VtxZDir

Cos(BeamDirErr) = cos(XthetaBeam)* VtxXDir +
cos(YthetaBeam)* VtxYDir +
cos(ZThetaBeam)* VtxZDir

McDirErr is exactly the same **MyDirErr**

BeamDirErr >= MyDirErr because of the photon beam dispersion

1 - Estimation of the photon beam dispersion in the MC data

beam dispersion for the selected energy bins can be calculated as:

Cos(PhotonBeamDispersion) = cos(XthetaBeam)* McXDir +
cos(YthetaBeam)* McYDir +
cos(ZThetaBeam)* McZDir

I computed the "PSF" exactly in the same way (counting up to 68%, and 95% containment), but this time using ***PhotonBeamDispersion instead of***

McDirErr or MyDirErr

```
// Incoming direction of the photon beam 0 deg
Double_t cosXTheta = 0.0;
Double_t cosYTheta = 0.0;
Double_t cosZTheta =-1.0;
// Incoming direction of the photon beam 40 deg
Double_t cosXTheta = -6.4273634724861605 8e-01;
Double_t cosYTheta = 0.0;
Double_t cosZTheta = -7.6604311646595 9573e-01;
```

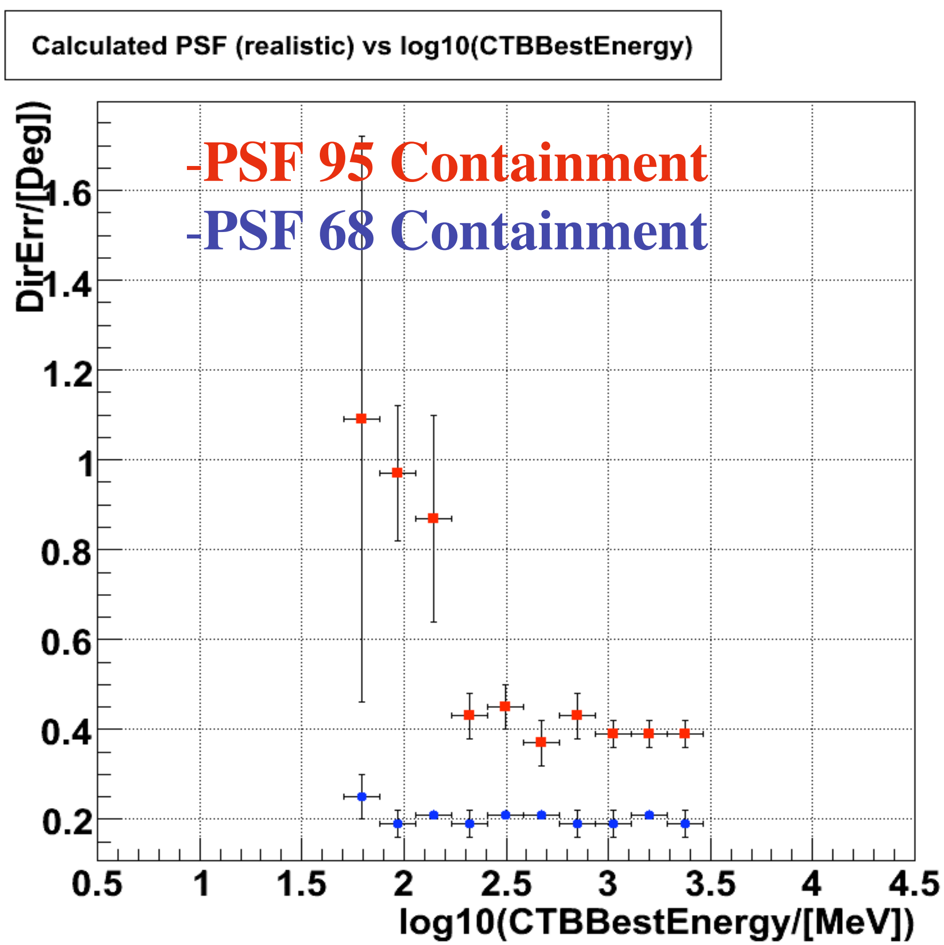
1 - Photon beam dispersion for each of these energy bins

For MC 125 (3GeV) , “PSF68” from this dispersion is FLAT, about 0.2 deg.

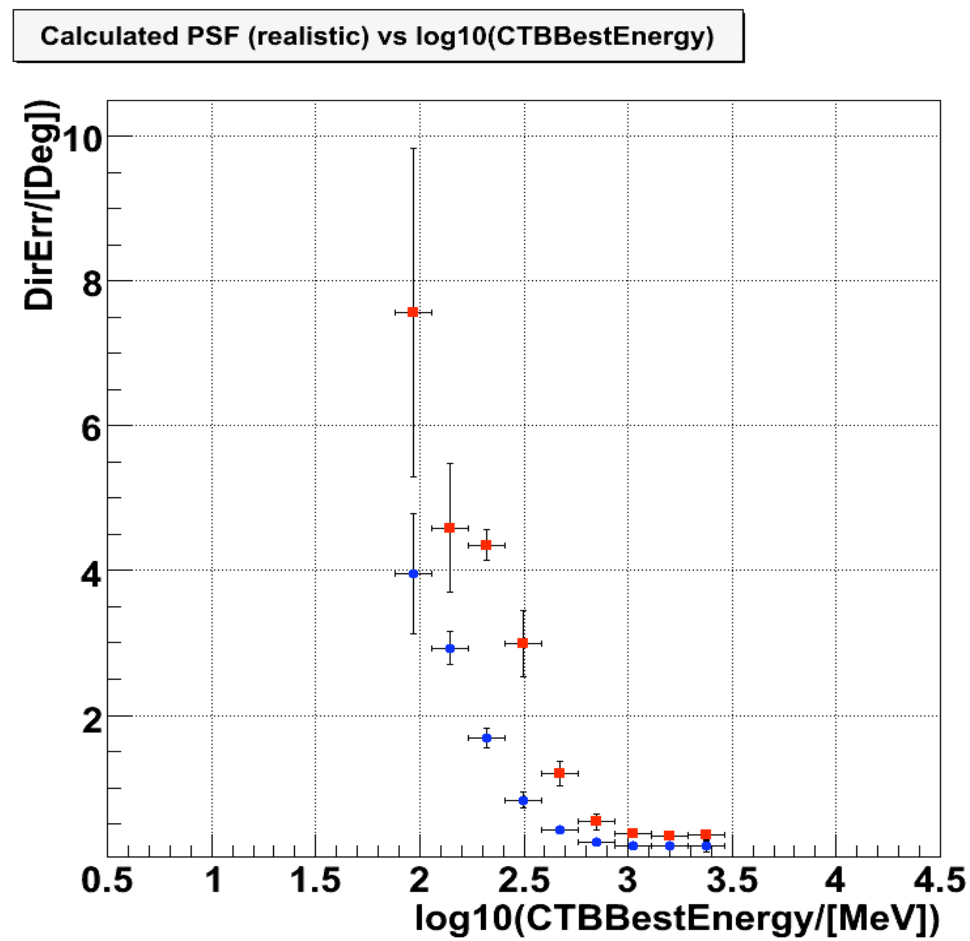
For MC 129 (2.5GeV), the “PSF68” from this dispersion is ENERGY dependent. It converges assimptotically to 0.2 at high energies.

WHY this difference ??

Run MC 125 (0 deg)



Run MC 129 (0 deg)



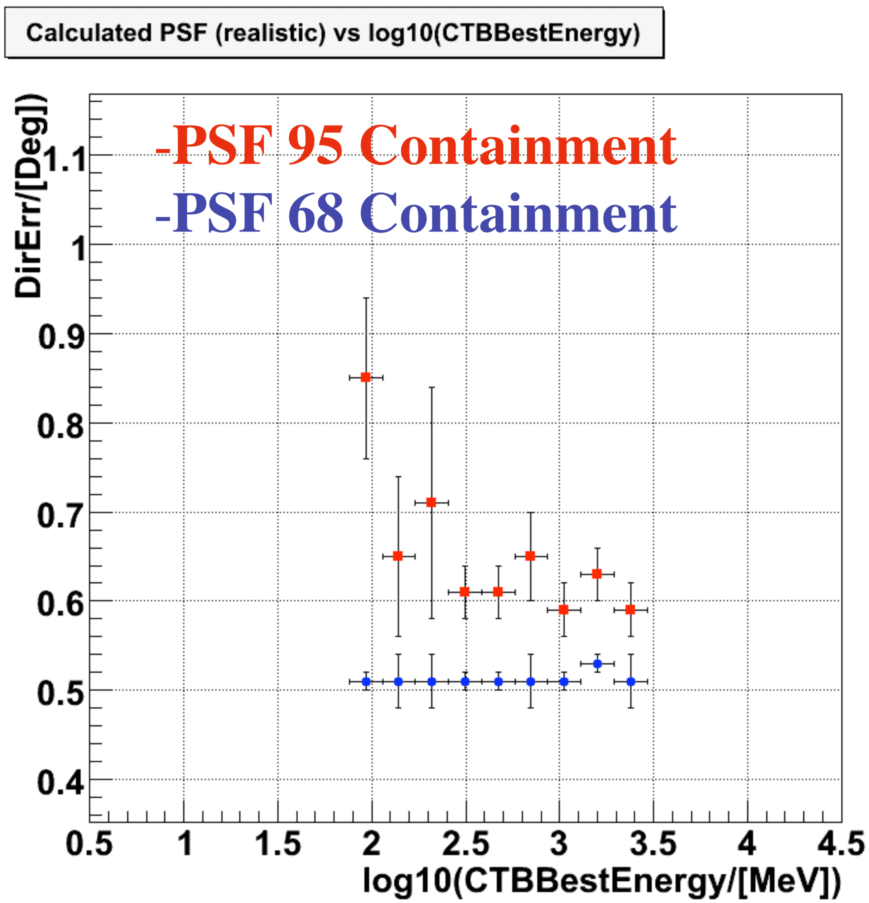
1 - Photon beam dispersion for each of these energy bins

For MC 127 (3GeV), "PSF68" from this dispersion is FLAT, about 0.5 deg.

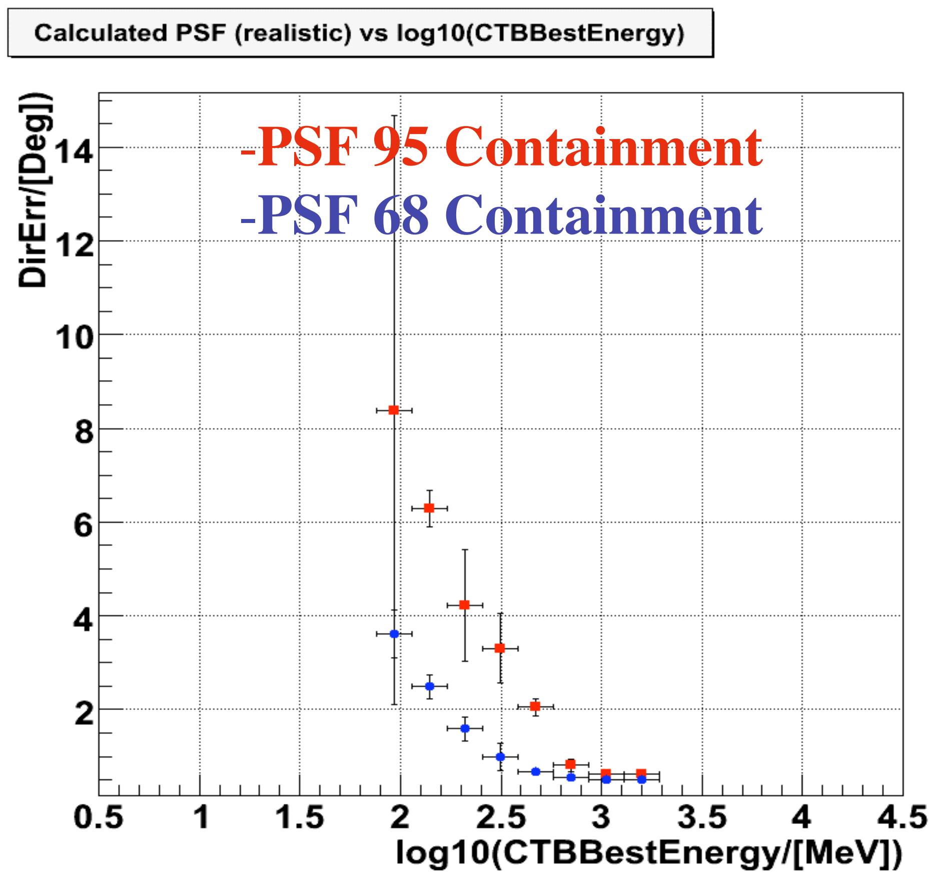
For MC 130 (2.5GeV), "PSF68" from this dispersion is ENERGY dependent. It converges assimptotically to 0.5 at high energies.

WHY this difference ?? Why dispersion larger than at 0 deg ?? Do I make a mistake in the argumentation ??

Run MC 127 (40 deg)



Run MC 130 (40 deg)



Two issues:

Change in beam dispersion when going from 0 to 40 deg:

Perhaps there is another factor in the calculation of the beam dispersion that I do not take into account, and which becomes important when increasing incoming angle… any idea ?

Change in beam dispersion with energy for the 2.5 GeV MC runs (which are the ones generated last)

This is very probably something related to the MC generation. Any idea ?

In principle we do not expect energy dependence in the brems photons with energy, *unless we have a electrons with different energies*