Subject: GEM tracking
Posted by Radoslaw Karabowicz on Tue, 14 Apr 2009 11:09:12 GMT
View Forum Message <> Reply to Message

Hi Christian and all the other tracking experts;)

I have just submitted my rather messy code for the tracking
in the GEM detector to the svn/trunk repository.
I have placed all the tracking code, as well as track related data container (like PndGemRecoHit, PndGemTrack) in the gem directory - later on it should be moved to pnddata. The code itself bases heavily on the dch and mvd tracking codes.

Could you please take a look at the code to check if I didn't do some (un)reasonable mistakes?

If you have any questions about the code, or if it is too messy to read, please let me know.
Thanks in advance,

yours,
radek

Subject: Re: GEM tracking
Posted by Anonymous Poster on Tue, 14 Apr 2009 15:51:20 GMT
View Forum Message <> Reply to Message

Hi,

I just took a first quick look. My question to Radek and Ralf is the following:

Do you have Pixel or Strip detetors? If you have both, do you try to handle then with the same recoHit?

Cheers, Christian

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Tue, 14 Apr 2009 16:06:43 GMT
View Forum Message <> Reply to Message

Dear Christian,

The GEM detector is to be a strip detector. Each of the 4 stations will be implemented with 2 sensitive planes, but remember, that each sensible plane sports two layers of strips. Thus there are 4 different views per station (see my slides from the last collaboration meeting, or the slides from last pandaroot EVO meeting). Because of the geometrical proximity of front (back) layers in one station it is possible to reconstruct 2 space points per station, separated by about 3 cm in "z".

Due to the above reasons I have decided to implement these space points for the fitter, and not the strips.

As a first attempt it is correct I think, especially that the PndGemIdealHitProducer produces space points out of the MC points, smeared with the resolution "expected" from the strip-to-hit finder.

Do you think it would be more correct to focus on the strip recoHit?

yours,
radek

Subject: Re: GEM tracking
Posted by Ralf Kliemt on Wed, 15 Apr 2009 09:47:18 GMT
View Forum Message <> Reply to Message

Dear Cristian and Radek,

As it is concerned to the Silicon Strip Sensors the different sides are at maximum 300 microns apart which I think to be sufficiant to start working with 2D points and error values. To match the two sides we also correlate the measured charges, which is not taken into account by the trackfitting.
I know you have forseen the possibility to fit tracks to onedimensional structures like wires and strips.

Regards, Ralf.

Subject: Re: GEM tracking
Posted by Anonymous Poster on Wed, 15 Apr 2009 11:01:56 GMT
View Forum Message <> Reply to Message

Hi Radek and Ralf,

I agree that things like amplitude correlation etc. are very helpful for pattern recognition, i.e. track finding, purposes. However, to my mind that does not mean, that in the track fitting this 2D information should be used. It is just as easy to use the 1D information in genfit. I tested this extensively. Even if the planes are only 300mu apart, this should be taken into account especially in the MVD were the spatial resolution will be much better than that. In the GEMs, it probably doenst matter so much.

Of course this choice is up to you guys. If you need any help in using the 1D info, please let me know and I will be happy to help.

Radek, about your RecoHit implementation in

PndGemRecoHit::PndGemRecoHit(PndGemHit* hit)

you set the hit coordinates and the covariance matrix like:

```
_hitCoord[0][0] = TMath::Sqrt(hitX*hitX+hitY*hitY);
_hitCoord[1][0] = 0.;
```

```
_hitCov[0][0] = hit->GetDr();
_hitCov[1][1] = hit->GetDp();
```

I dont understand, why one of the two hit dimensions is always 0. This should be the case if the PndGemHit is a 2D info.

Cheers, Christian

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Wed, 15 Apr 2009 14:03:02 GMT
View Forum Message <> Reply to Message

Yes, one of the two hit coordinates I set always to zero, while the other is set to the radius distance of the hit to the center. The errors are then corresponding to the error in radius (hit->GetDr()), and error perpendicular to it (hit->GetDp()).
Concequently, the whole DetectorPlane is rotated by the angle of the hit. I hope I did it right, at least the track position (trk->getPos() corresponds to the point I have insterted there.

Don't I do it correctly?, first setting the vectors uu and vv:

```
TVector3 oo (0.,0.,hit->GetZ()),
        uu ( TMath::Sin(phiAValue), TMath::Cos(phiAValue),0),
        vv ( TMath::Cos(phiAValue),-TMath::Sin(phiAValue),0);
```

and then setting the detector plane using the vectors:

```
setDetPlane(DetPlane(oo,uu,vv));
```

I hope it was tested before.
I wanted to have the errors expressed in Dr and Dp, as it seemed to me the best choise.

yours,
radek

Subject: Re: GEM tracking
Posted by Anonymous Poster on Mon, 27 Apr 2009 09:56:04 GMT
View Forum Message <> Reply to Message

Hi Radek,

I think you did it correctly. But I was wondering, why you do this rotation in the first place? The GEM measures in two strip projections, and it is in these direction, that you determine your detector resolutions. So, I would recommend to use the strip directions for determining the u and v vectors. Ans then your hit coordinates are just translated from the strips which fired. In the rotated case, correctly determining the hit covariance matrix is more difficult. For sure you wont have a diagonal matrix...

Concerning the 1D vs. 2D discussion: You can of course use the 2D information, there is no problem with that.

Did you obtain some results yet? If you do a gaussian smearing for your hits with known resolution, you should obtain correct pull and chi2 distributions. How does that look?

Cheers, Christian

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Tue, 28 Apr 2009 16:11:03 GMT
View Forum Message <> Reply to Message

Dear Christian,

Thank you for your answer. This post will be rather long, so prepare;).

First of all: I am using pndroot revision 5003, i will get a new version tomorrow, I did not wanted to change anything in the environment for the test.

I have made a lot of print-outs in my code, which allowed me to make several observations:

1. charge. Both in the Geane and LSL track represtantation constructor you are taking charge of the particle (charge or q/p). At least some people take the charge from root pdg database by:

```
TDatabasePDG *fdbPDG= TDatabasePDG::Instance();
TParticlePDG *fParticle= fdbPDG->GetParticle(PDGCode);
Double_t  fCharge= fParticle->Charge();
```

but the correct way should be:

```
Double_t  fCharge= fParticle->Charge()/3.;
```

because in TParticlePDG from root there is:

```
Double_t        fCharge;          // charge in units of |e|/3
```

Fixing this did not improve anything, so it is only a (meaningless) comment.

2. I still cannot make the fitter to run. I mean it is running, but it is not fitting at all. What I mean is that the parameters like momentum change insignificantly. I am doing: initialize track parameters with position and momentum taken from the first GEM point (most people use 0.,0.,0. as the starting point, is what I am using wrong?). I smear the position by (0.1,0.1,0.1) and increase the momentum by 0.1 to see if the fitter will fix it (should it?). The example printouts come from my PndGemKalmanTask and from yours Kalman.cxx (i print the track parameter in the loop of Kalman::processTrack(Track* trk)):

```
starting track0
FIRST GEM POINT AT:  (x, y, z)
******** -6.55763 27.2351 89.3803
WITH MOMENTUM: (px, py, pz ----> |p|)
******** -0.192626 0.445268 1.54116 ---> 1.61572
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
```

```
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
8 hits in track 0
starting fit
*****************************************************

before first iteration in Kalman::processTrack, 0.1 has been added to momentum.mag
0: result pos = (-6.47268,27.2066,89.3913)
0: result mom = (-0.204548,0.472827,1.63655) ---> 1.71572
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -6.47258 27.2067  89.3913
 FairGeanePro::Propagate --------------------------  -6.47133 27.2036  89.3803
 FairGeanePro::Propagate --------------------------  -6.70876 27.6091  90.6207
 FairGeanePro::Propagate --------------------------  -10.8173 35.6645  119.38
 FairGeanePro::Propagate --------------------------  -10.9983 35.9863  120.621
 FairGeanePro::Propagate --------------------------  -15.6744 43.7319  149.38
 FairGeanePro::Propagate --------------------------  -15.988 43.9828  150.621
 FairGeanePro::Propagate --------------------------  -20.9761 51.4469  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.7575 51.3515  180.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.5577 51.0004  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.4667 43.6138  150.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.136 43.3895  149.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -11.0507 35.9565  120.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.826 35.6454  119.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -6.22762 27.9268  90.6207

before second iteration in Kalman::processTrack
1: result pos = (-6.19682,27.4796,89.3803)
1: result mom = (-0.273583,0.43237,1.63766) ---> 1.71573
 FairGeanePro::Propagate --------------------------  -6.51044 27.267  89.3803
 FairGeanePro::Propagate --------------------------  -6.72045 27.6048  90.6207
 FairGeanePro::Propagate --------------------------  -11.0317 35.5209  119.38
 FairGeanePro::Propagate --------------------------  -11.1274 35.9024  120.621
 FairGeanePro::Propagate --------------------------  -15.7653 43.6395  149.38
 FairGeanePro::Propagate --------------------------  -16.0421 43.9262  150.621
 FairGeanePro::Propagate --------------------------  -21.0838 51.3342  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.771 51.3386  180.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.5622 50.9962  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.3376 43.7225  150.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.04 43.4701  149.38
GEANETRACKREP: USING BACKPROPAGATION!
```

```
 FairGeanePro::Propagate --------------------------  -10.9413 36.0422  120.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.7573 35.6996  119.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -6.18766 27.9631  90.6207

before third iteration in Kalman::processTrack
2: result pos = (-6.17669,27.4995,89.3803)
2: result mom = (-0.27435,0.431638,1.63773) ---> 1.71573
 FairGeanePro::Propagate --------------------------  -6.50937 27.2686  89.3803
 FairGeanePro::Propagate --------------------------  -6.7202 27.6054  90.6207
 FairGeanePro::Propagate --------------------------  -11.0291 35.5213  119.38
 FairGeanePro::Propagate --------------------------  -11.1251 35.9029  120.621
 FairGeanePro::Propagate --------------------------  -15.7649 43.6389  149.38
 FairGeanePro::Propagate --------------------------  -16.0422 43.9256  150.621
 FairGeanePro::Propagate --------------------------  -21.0853 51.3326  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.7713 51.3384  180.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.5623 50.9961  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.337 43.7229  150.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.0396 43.4704  149.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.9409 36.0425  120.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.757 35.6998  119.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -6.18749 27.9632  90.6207

before forth iteration in Kalman::processTrack
3: result pos = (-6.1766,27.4996,89.3803)
3: result mom = (-0.274355,0.431634,1.63773) ---> 1.71573
 FairGeanePro::Propagate --------------------------  -6.50936 27.2686  89.3803
 FairGeanePro::Propagate --------------------------  -6.7202 27.6054  90.6207
 FairGeanePro::Propagate --------------------------  -11.0291 35.5212  119.38
 FairGeanePro::Propagate --------------------------  -11.1251 35.9029  120.621
 FairGeanePro::Propagate --------------------------  -15.7649 43.6389  149.38
 FairGeanePro::Propagate --------------------------  -16.0422 43.9256  150.621
 FairGeanePro::Propagate --------------------------  -21.0853 51.3326  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.7713 51.3384  180.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.5623 50.9961  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.337 43.723  150.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.0395 43.4704  149.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.9409 36.0425  120.621
GEANETRACKREP: USING BACKPROPAGATION!
```

```
 FairGeanePro::Propagate --------------------------  -10.757 35.6998  119.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -6.18749 27.9632  90.6207

before fifth iteration in Kalman::processTrack
4: result pos = (-6.1766,27.4996,89.3803)
4: result mom = (-0.274355,0.431634,1.63773) ---> 1.71573
 FairGeanePro::Propagate --------------------------  -6.50936 27.2686  89.3803
 FairGeanePro::Propagate --------------------------  -6.7202 27.6054  90.6207
 FairGeanePro::Propagate --------------------------  -11.0291 35.5212  119.38
 FairGeanePro::Propagate --------------------------  -11.1251 35.9029  120.621
 FairGeanePro::Propagate --------------------------  -15.7649 43.6389  149.38
 FairGeanePro::Propagate --------------------------  -16.0422 43.9256  150.621
 FairGeanePro::Propagate --------------------------  -21.0853 51.3326  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.7713 51.3384  180.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.5623 50.9961  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.337 43.723  150.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.0395 43.4704  149.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.9409 36.0425  120.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.757 35.6998  119.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -6.18749 27.9632  90.6207

before sixth iteration in Kalman::processTrack
5: result pos = (-6.1766,27.4996,89.3803)
5: result mom = (-0.274355,0.431635,1.63773) ---> 1.71573
 FairGeanePro::Propagate --------------------------  -6.50936 27.2686  89.3803
 FairGeanePro::Propagate --------------------------  -6.7202 27.6054  90.6207
 FairGeanePro::Propagate --------------------------  -11.0291 35.5212  119.38
 FairGeanePro::Propagate --------------------------  -11.1251 35.9029  120.621
 FairGeanePro::Propagate --------------------------  -15.7649 43.6389  149.38
 FairGeanePro::Propagate --------------------------  -16.0422 43.9256  150.621
 FairGeanePro::Propagate --------------------------  -21.0853 51.3326  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.7713 51.3384  180.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -21.5623 50.9961  179.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.337 43.723  150.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -16.0395 43.4704  149.38
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.9409 36.0425  120.621
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -10.757 35.6998  119.38
GEANETRACKREP: USING BACKPROPAGATION!
```

```
 FairGeanePro::Propagate --------------------------  -6.18749 27.9632  90.6207
GEANETRACKREP: USING BACKPROPAGATION!
 FairGeanePro::Propagate --------------------------  -6.1766 27.4996  89.3803
ChiSq=13.963
*****************************************************

Resulting position and momentum after 6 iterations of  Kalman::processTrack
result pos = (-6.1766,27.4996,89.3803)
result mom = (-0.274355,0.431635,1.63773) ----> 1.71573
```

Observation:

The momentum |p| changes very little in the process of the fitting, the three components do change considerably, from
(-0.204548,0.472827,1.63655) to (-0.274355,0.431635,1.63773), but the |p| changes only from 1.71572 to 1.71573. This kind of behaviour I see for each and every particle that I tried to fit (you can check the attached file). Do you fix the overall particle momentum so that it cannot change during the fitting?

3. I have modified the RecoHit from:

```
  TVector3 oo (0.,0.,hit->GetZ()),
        uu ( TMath::Cos(phiAValue),TMath::Sin(phiAValue),0),
        vv (-TMath::Sin(phiAValue),TMath::Cos(phiAValue),0);
  _hitCoord[0][0] = TMath::Sqrt(hitX*hitX+hitY*hitY);
  _hitCoord[1][0] = 0.;
  _hitCov[0][0] = hit->GetDr();
  _hitCov[1][1] = hit->GetDp();
```

to:

```
  TVector3 oo (0.,0.,hit->GetZ()),
    uu ( 1.0, 0.0, 0.0),
    vv ( 0.0, 1.0, 0.0);

  _hitCoord[0][0] = hitX;
  _hitCoord[1][0] = hitY; // by the way, should it be [1][0] or [1][1]?

  _hitCov[0][0] = 0.1;
  _hitCov[1][1] = 0.1;
```

Both versions work exactly the same.

4. I have tried to use the LSL track representation:

```
starting track0
******** -6.55759 27.2354 89.3803
******** -0.192626 0.445268 1.54116 ---> 1.61572
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
```

```
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
 -I- PndGemRecoHit::PndGemRecoHit(PndGemHit*) called.
8 hits in track 0
starting fit
*****************************************************
0: result pos = (-6.47264,27.2069,89.3913)
0: result mom = (-0.204548,0.472827,1.63655) ---> 1.71572
FitterException thrown with whatString:
cov[0][0]<1.-50
in line: 225 in file: /misc/karabowi/pandaroot-5003/trunk/genfit/Kalman.cxx
FitterException Info Output
==========================
==========================
1: result pos = (-6.47264,27.2069,89.3913)
1: result mom = (-0.204548,0.472827,1.63655) ---> 1.71572
2: result pos = (-6.47264,27.2069,89.3913)
2: result mom = (-0.204548,0.472827,1.63655) ---> 1.71572
3: result pos = (-6.47264,27.2069,89.3913)
3: result mom = (-0.204548,0.472827,1.63655) ---> 1.71572
4: result pos = (-6.47264,27.2069,89.3913)
4: result mom = (-0.204548,0.472827,1.63655) ---> 1.71572
5: result pos = (-6.47264,27.2069,89.3913)
5: result mom = (-0.204548,0.472827,1.63655) ---> 1.71572
*****************************************************
result pos = (-6.47264,27.2069,89.3913)
result mom = (-0.204548,0.472827,1.63655) ----> 1.71572
```

The exception using this representation is for every track, the parameters doesn't change at all... (check attached file, it unfortunately lacks information about exceptions).

5. I have similar exception in Geane as in LSL when I do not smear the position of the first GEM point as starting position for the Kalman.

Conclusion:

I seem to have some output from the fitter, but the problem is that the parameters do not change as I would like them to change.
In particular in the Geane rep. the momentum |p| stay constant in the process of fitting, the LSL doesn't work for me at all.
Could you take a look at the attached files and tell me if the fitter output looks normally?

That it for today.
Yours,
radek

ps. Thank you for patience, if you got down here;)

File Attachments

1) geaneOutput.dat, downloaded 522 times
2) lslOutput.dat, downloaded 498 times

Subject: Re: GEM tracking
Posted by StefanoSpataro on Tue, 28 Apr 2009 16:22:19 GMT
View Forum Message <> Reply to Message

Hi,
just a question: which momentum resolution are you giving at the beginning (I am not speaking about the smearing, just the momentum error).
Because I had a similar problem, and then I have found that I was setting the error to zero -> there was no fitting at all. Then I have started to put a 10% as initial momentum error, and the fitter started to fit

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Wed, 29 Apr 2009 07:55:59 GMT
View Forum Message <> Reply to Message

Dear Stefano,
I have actually thought about it. The first version of my PndGemKalmanTask had the errors set to zero. I have changed the line:

```
TVector3 StartMomErr = TVector3(0.,0.,0.);
```

to:

```
TVector3 StartMomErr = TVector3(1.,1.,1.);
```

some 2 weeks ago, but nothing changed.

Thank you anyways for a hint.

yours,
radek

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Wed, 29 Apr 2009 09:51:06 GMT
View Forum Message <> Reply to Message

Dear Stefano et al

Following the todays EVO meeting discussion, I have tried several options:

1. Look at the tracks in the outer regions of the GEM detector, or in other words, the tracks with high enough pt to use the bending power of the magnet. Comparison:
TRACK IN OUTER REGION
******** 3.40493 30.9044 89.3803
******** 0.110931 0.157997 0.53312 ---> 0.566997
ADDING 0.1 TO |p|
0: result pos = (3.37962,30.996,89.3649)
0: result mom = (0.130496,0.185863,0.627145) ---> 0.666997
FITTED MOMENTUM, difference with start momentum is 0.0003

result pos = (2.81193,32.2641,89.3803)
result mom = (0.1866,0.105973,0.631134) ----> 0.666618
TRACK IN INNER REGION WITH SIMILAR MOMENTUM
******** 3.45711 4.23193 89.3803
******** 0.00989272 0.0359984 0.576766 ---> 0.577973
ADDING 0.1 TO |p|
0: result pos = (3.58859,3.96869,89.5048)
0: result mom = (0.0116043,0.0422268,0.676558) ---> 0.677973
FITTED MOMENTUM, difference with start momentum is 0.00003
result pos = (3.70969,4.29123,89.3803)
result mom = (-0.00547421,0.0411893,0.676671) ----> 0.677946

In general, not a big improvement is seen for tracks with higher polar angle.

2. Square the hit errors when inserting into the covariance matrix. Change:
```
  _hitCov[0][0] = hit->GetDr();
  _hitCov[1][1] = hit->GetDp();
```
to:
```
  _hitCov[0][0] = hit->GetDr()*hit->GetDr();
  _hitCov[1][1] = hit->GetDp()*hit->GetDp();
```
No change in the track fitting performance what so ever.

3. Change the initial track momentum errors from setting it to (1.,1.,1.) to 10% of the initial track momentum, i.e. change:
```
   TVector3 StartMomErr = TVector3(1.,1.,1.);
```
to:
```
   TVector3 StartMomErr = TVector3(0.1*StartMom);
```
BINGO!
The track fitting does now:

CORRECT TRACK POSITION, MOMENTUM AT FIRST GEM HIT:
******** -1.91359 7.80279 89.3803
******** -0.146677 0.492359 5.70466 ---> 5.72775
ADDING 0.1 TO MOMENTUM |p|
****
0: result pos = (-1.95901,7.91932,89.2568)
0: result mom = (-0.149238,0.500955,5.80426) ---> 5.82775
1: result pos = (-1.91288,7.80297,89.3803)
1: result mom = (-0.154325,0.491343,5.71709) ---> 5.74024
2: result pos = (-1.91288,7.80297,89.3803)
2: result mom = (-0.154326,0.491344,5.71709) ---> 5.74024
3: result pos = (-1.91288,7.80297,89.3803)
3: result mom = (-0.154326,0.491344,5.71709) ---> 5.74024
4: result pos = (-1.91288,7.80297,89.3803)
4: result mom = (-0.154326,0.491344,5.71709) ---> 5.74024
5: result pos = (-1.91288,7.80297,89.3803)
5: result mom = (-0.154326,0.491344,5.71709) ---> 5.74024
****
result pos = (-1.91288,7.80297,89.3803)
result mom = (0.154326,-0.491344,-5.71709) ----> 5.74024
****

Already after the first iteration we are close enough to initial track momentum. The next iterations do not change results so much.

Finally a success. Thank you all that contributed to solving the GEM problem, especially Christian and Stefano.

yours,
radek

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Thu, 30 Apr 2009 09:55:32 GMT
View Forum Message <> Reply to Message

Problem continues...

It seems that my yesterday enthusiastic mail was a bit preliminary.
I have been looking at several tracks at it looked rather ok, however when I look systematicly into the results I have realised that the fitting is still producing not optimal results.
As you may remember, I was giving to the Kalman fitter a Monte Carlo momentum increased by 100Mev/c as a starting value. What I have realized is that the fitter smears the momentum with some sigma, but the reconstructed mean momentum value of many tracks still differs from the MC momenta by a value of 100MeV/c.

I have therefore made two runs of my reconstruction, once with having the start momenta given to Kalman decreased by .5 GeV/c as compared to MC truth, and in the second run I set start momenta as MC truth plus .5 GeV/c. Plots below summarize the results:

Momenta decreased:

```
StartMom.SetMagThetaPhi(beforeMom.Mag()-0.5,
        beforeMom.Theta(),
    beforeMom.Phi());
```

Momenta increased:

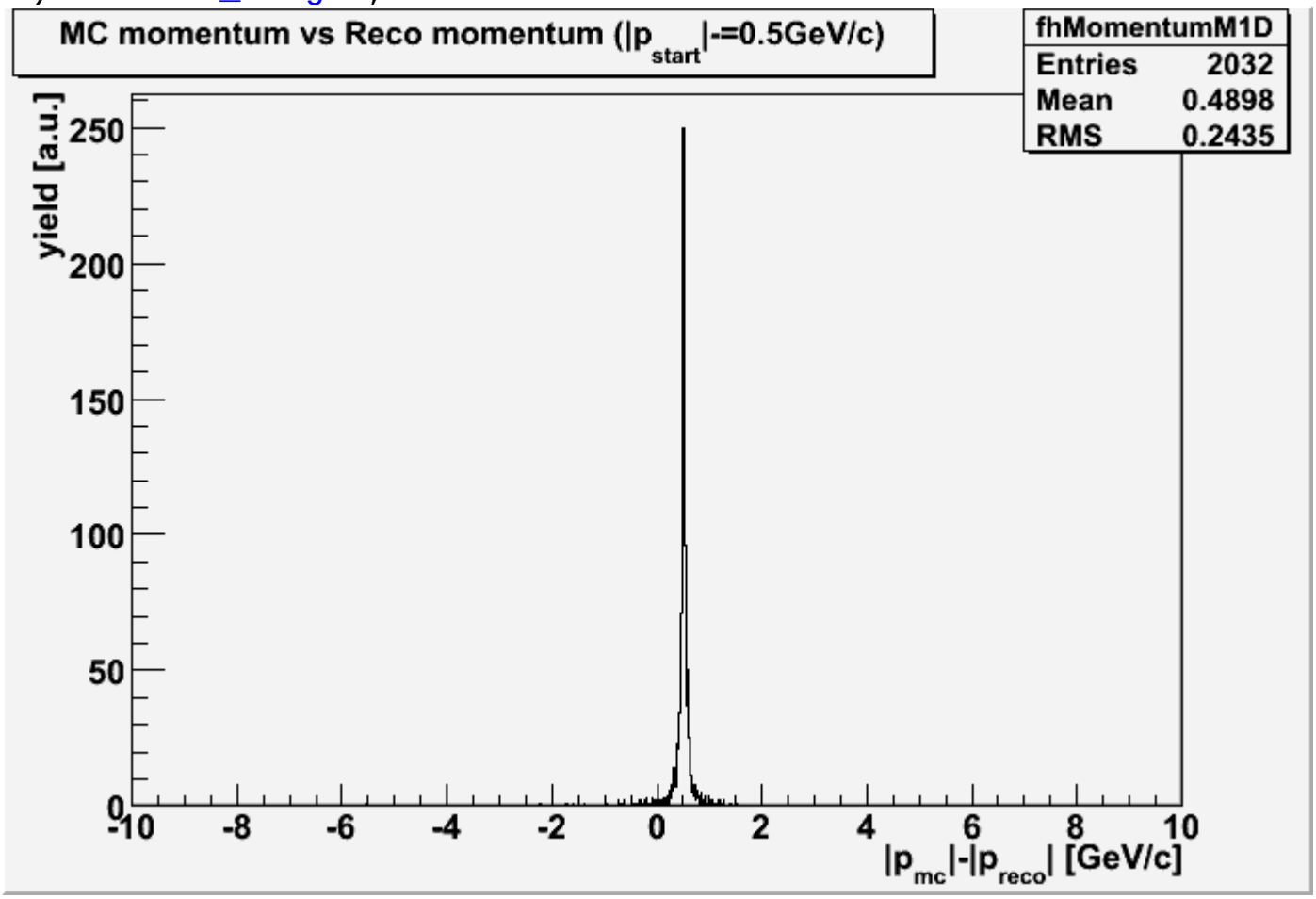
```
StartMom.SetMagThetaPhi(beforeMom.Mag()+0.5,
        beforeMom.Theta(),
    beforeMom.Phi());
```

Please take a look at the mean value of the distribution, the reco momenta is just smeard, it doesn't come back to MC truth:((.
The next thing I am planing to do is to use the tracks with high p_t. But that will come probably the next week.

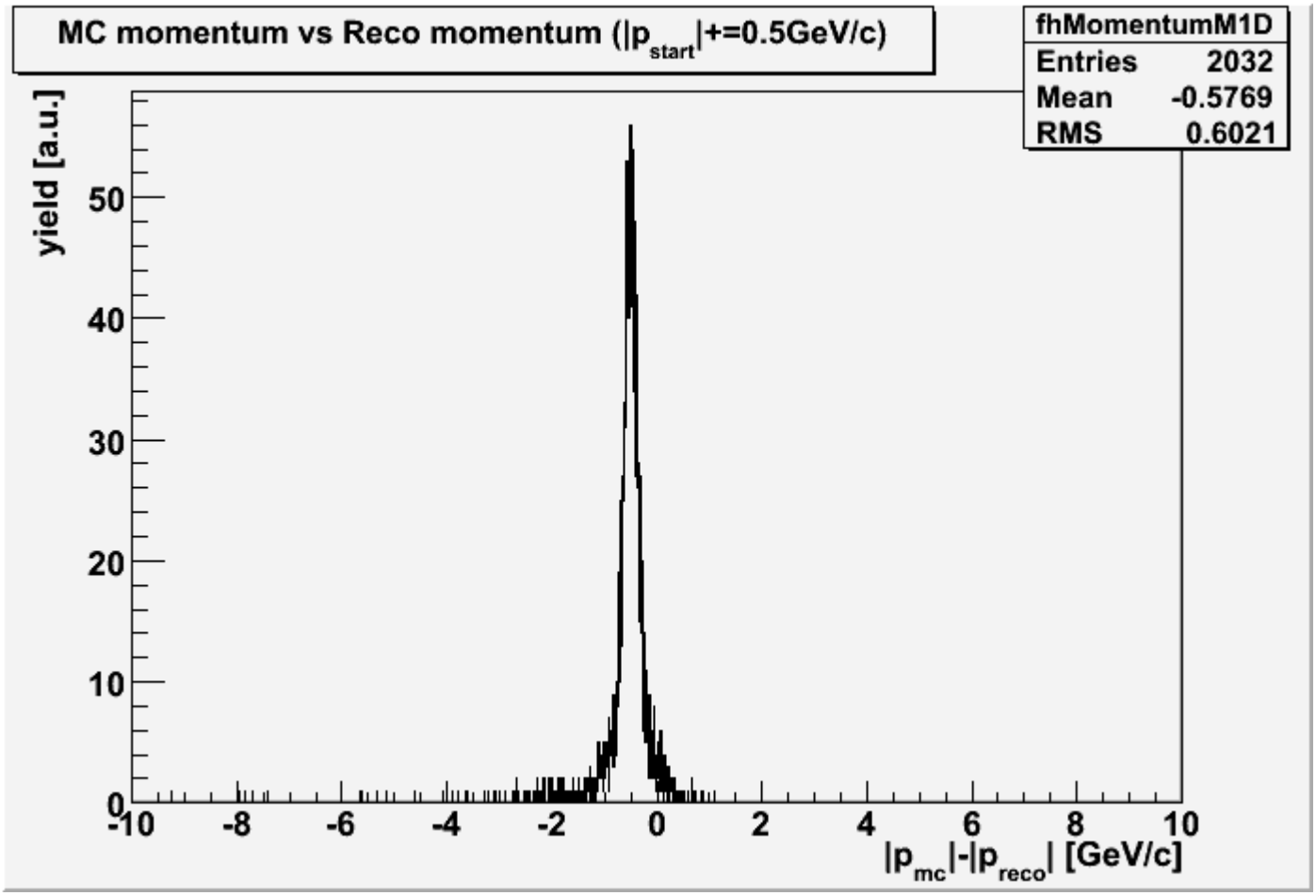
yours,
radek

File Attachments

1) momDiff_m5.gif, downloaded 1228 times



2) momDiff_p5.gif, downloaded 1313 times



Subject: Re: GEM tracking
Posted by StefanoSpataro on Thu, 30 Apr 2009 10:11:20 GMT
View Forum Message <> Reply to Message

Hi,
just a question: how much is the momentum value of the simulated tracks?

Subject: Re: GEM tracking
Posted by Anonymous Poster on Thu, 30 Apr 2009 10:12:52 GMT
View Forum Message <> Reply to Message

And more: How is the theta of the tracks?

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Thu, 30 Apr 2009 13:52:11 GMT
View Forum Message <> Reply to Message

I am using DPM events at 15 GeV, and reconstruct particles with MC momenta larger than 0.5 GeV/c (just an arbitrary cut).

So I have various momenta from 0.5 up to 15 GeV/c lets say, and all possible thetas.
yours.
radek

ps. have a nice weekend;)

Subject: Re: GEM tracking
Posted by Anonymous Poster on Thu, 30 Apr 2009 16:26:34 GMT
View Forum Message <> Reply to Message

Hi Radek,

I propose the following:

Please use the box generator, since in the beginning the momena of particles is important until you understand everything. Also, I would like to see what happens as a proof of principle of your recoHit implementation what happens if you change your geometry a little:

- Remove all other stuff from the setup
- make your GEMs much larger, so they will cover a higher theta acceptance
- use some more planes in between the ones you have.

If we make the reconstruction work in this scenario, then we can try to find out, why in the real case it doesnt work. I still suspect that you dont get enough bending to measure |p|.

Looking forward to hear about your tests...

Christian

Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Thu, 04 Jun 2009 16:10:34 GMT
View Forum Message <> Reply to Message

Dear Christian,

I have to apologize for my late answering, but the test and the results analysis took a bit longer I expected. The change from linux to mac didn't help either.
But here I am with something interesting, I hope.

I took the box generator, and used:

```
FairBoxGenerator* boxGen = new FairBoxGenerator(211,1);
boxGen->SetThetaRange(theta   -0.01,theta   +0.01);
boxGen->SetPhiRange  (phi     -0.01,phi     +0.01);
boxGen->SetPRange    (momentum-0.01,momentum+0.01);
primGen->AddGenerator(boxGen);
```

I have been varying the theta parameter from 2 to 40, phi from 0 to 360, and momentum was {0.1,0.2,0.5,1.0,2.0,4.0,8.0}. I have generated 1000 pions in each of the bins and tried to reconstruct them using start parameters:

```
TVector3 StartPos    = TVector3 (0.,0.,0.);
TVector3 StartPosErr = TVector3(0.01,0.01,0.01);

TVector3 StartMom    = TVector3 (1.,0.,1.);
StartMom.SetMagThetaPhi(fMomentum+0.05,
                        TMath::DegToRad()*(Double_t)fTheta,
                        TMath::DegToRad()*(Double_t)fPhi);
TVector3 StartMomErr = TVector3(0.1*StartMom);
```

where fMomentum, fTheta and fPhi were taken from the real parameters of the simulated pions.

As the results I have been looking at the distribution of the reconstructed momentum:

This plot shows the distribution of reconstructed momentum magnitude for pions shot at phi = 0, theta = 10. The distribution look OK, maybe a bit to broad, but at least at correct momentum.

Taking the different distributions for pions shot at different theta and phi, I was able to construct a map of the mean values of the reconstructed momentum magnitude as functions of theta and phi. The resulting map for 2 GeV pions is:

It certainly looks strange, but the behaviour is consistently repeated with different particle momenta (here 0.5 GeV):

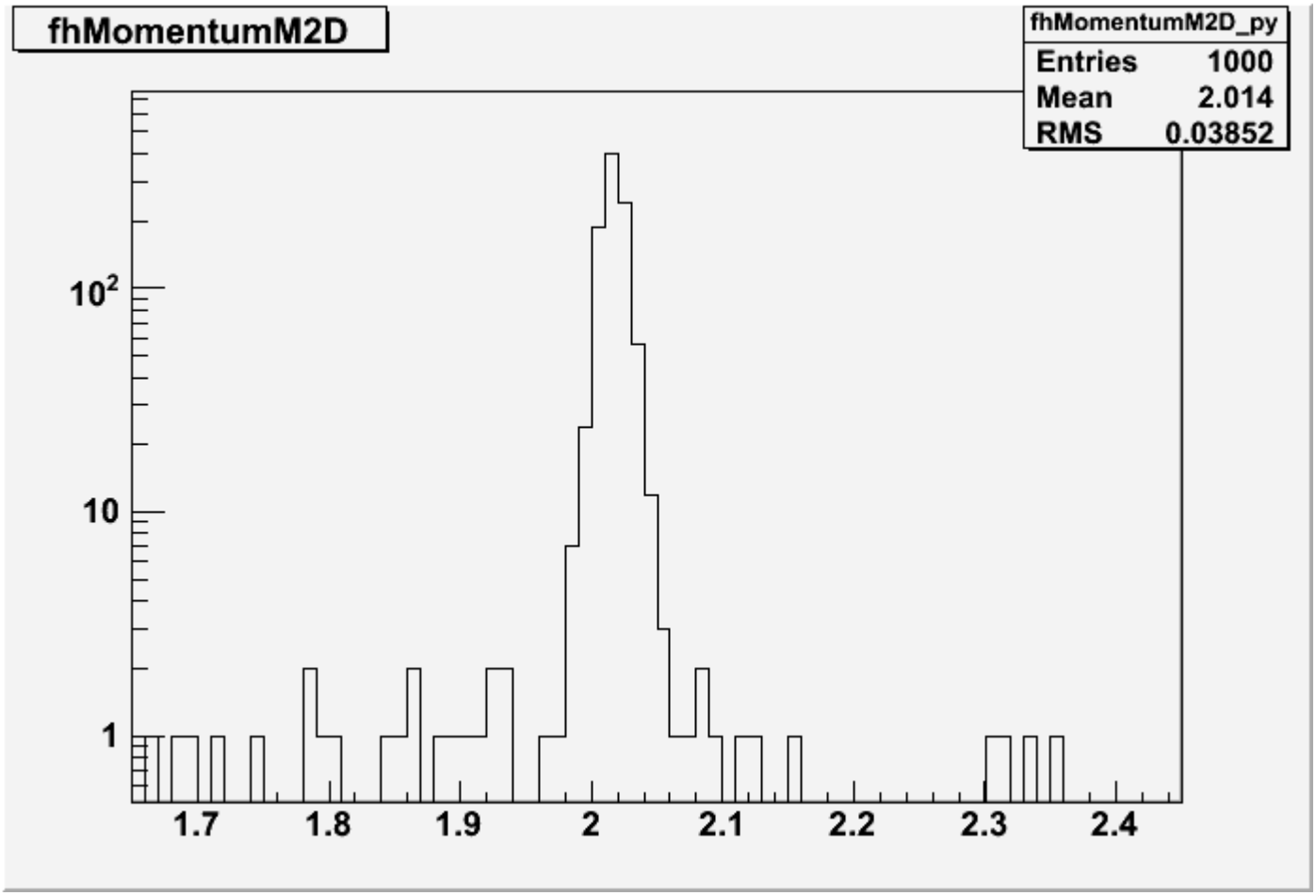
Have you ever seen anything so strange? For some phi angles the tracking performs good for all theta, but for some angles the fitting gives WAY too small or too large momenta, consistently for all theta. Even worse, the larger the theta angle, the greater the discrepancy.

The only thing I am afraid of is that I messed with the angles somehow, i mean rads and degs, I am checking it now... (well, tomorrow).

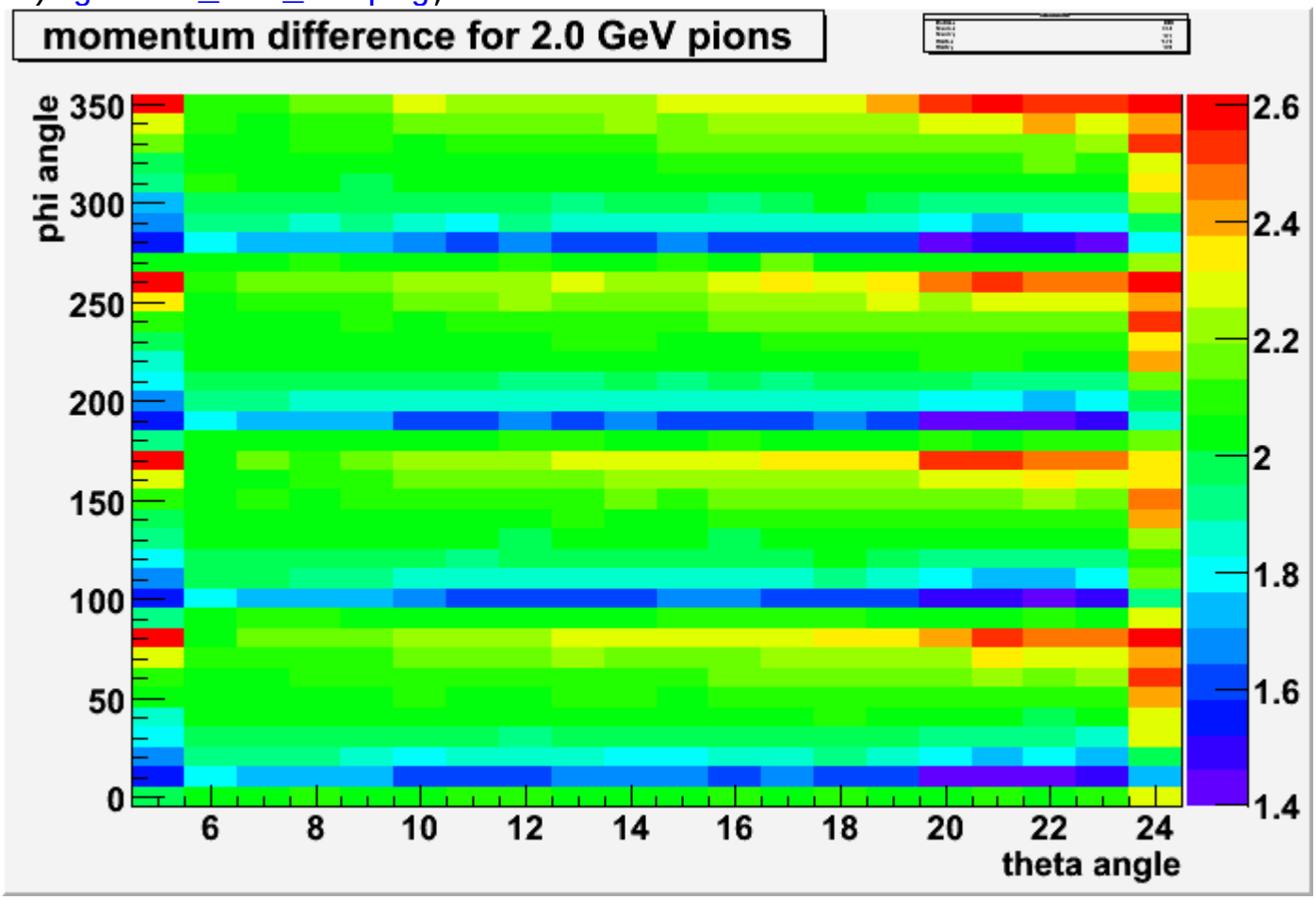
yours,
radek

File Attachments

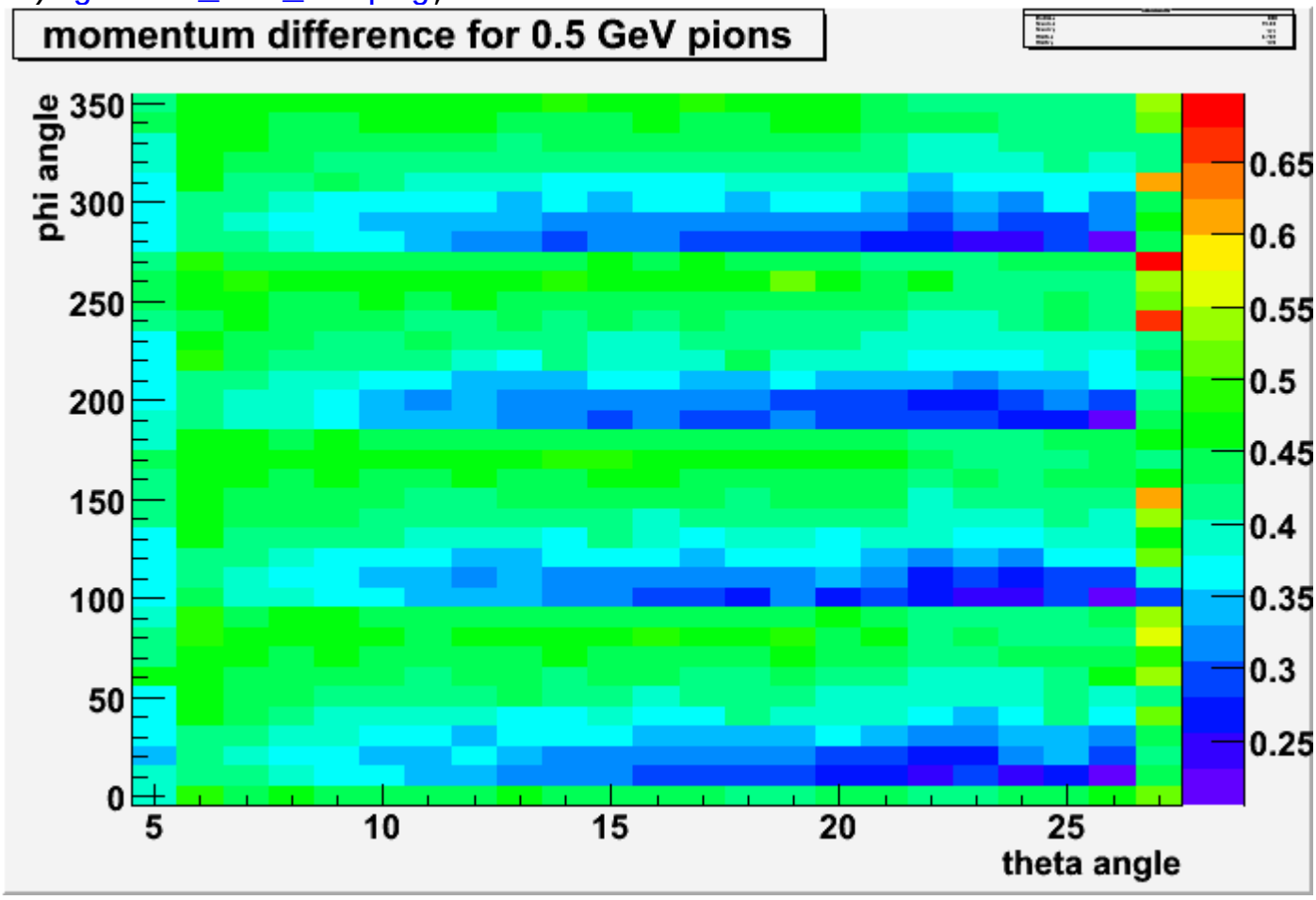
1) mom_reco_2_10_0.gif, downloaded 1153 times



2) gev2.0_mom_cl.png, downloaded 1129 times



3) gev0.5_mom_cl.png, downloaded 1208 times



Subject: Re: GEM tracking
Posted by Radoslaw Karabowicz on Thu, 25 Jun 2009 14:22:17 GMT
View Forum Message <> Reply to Message

Happy news finally!

During the Panda Torino meeting we sat together with Christian and looked at the GEM track fitting. After several hours we pinned down the problem and realized that the GEANE was not aware of having any magnetic field at all.
We found out that in the GEM MC simulation macro the magnetic field was not put into the par.root file. To be more precise, the following lines were missing:

```
 PndMultiFieldPar* fieldPar =
    (PndMultiFieldPar*) rtdb->getContainer("PndMultiFieldPar");
  if(fField)  {  fieldPar->SetParameters(fField); }
  fieldPar->setInputVersion(fRun->GetRunId(),1);
  fieldPar->setChanged(kTRUE);
```

Also in the reconstruction macro, the magnetic field was not sent to the Geane. After adding line:

```
  Geane->SetField(fRun->GetField());
```

the tracking now performs perfect.

I have again shot 1000 pions with momentum 2GeV/c, at some theta and phi angles. I have used the distribution of the fitted tracks' momenta to draw the mean reconstructed momentum as function of theta and phi. The resulting plot is here:

The fitted momenta are very close to 2.0 GeV/c which was the MC momentum and much lower than the initial GENFIT momentum, which was set to 2.1GeV/c.

I have also constructed similar map, but this time with the width of gaussian fits to the reconstructed momentum distribution:

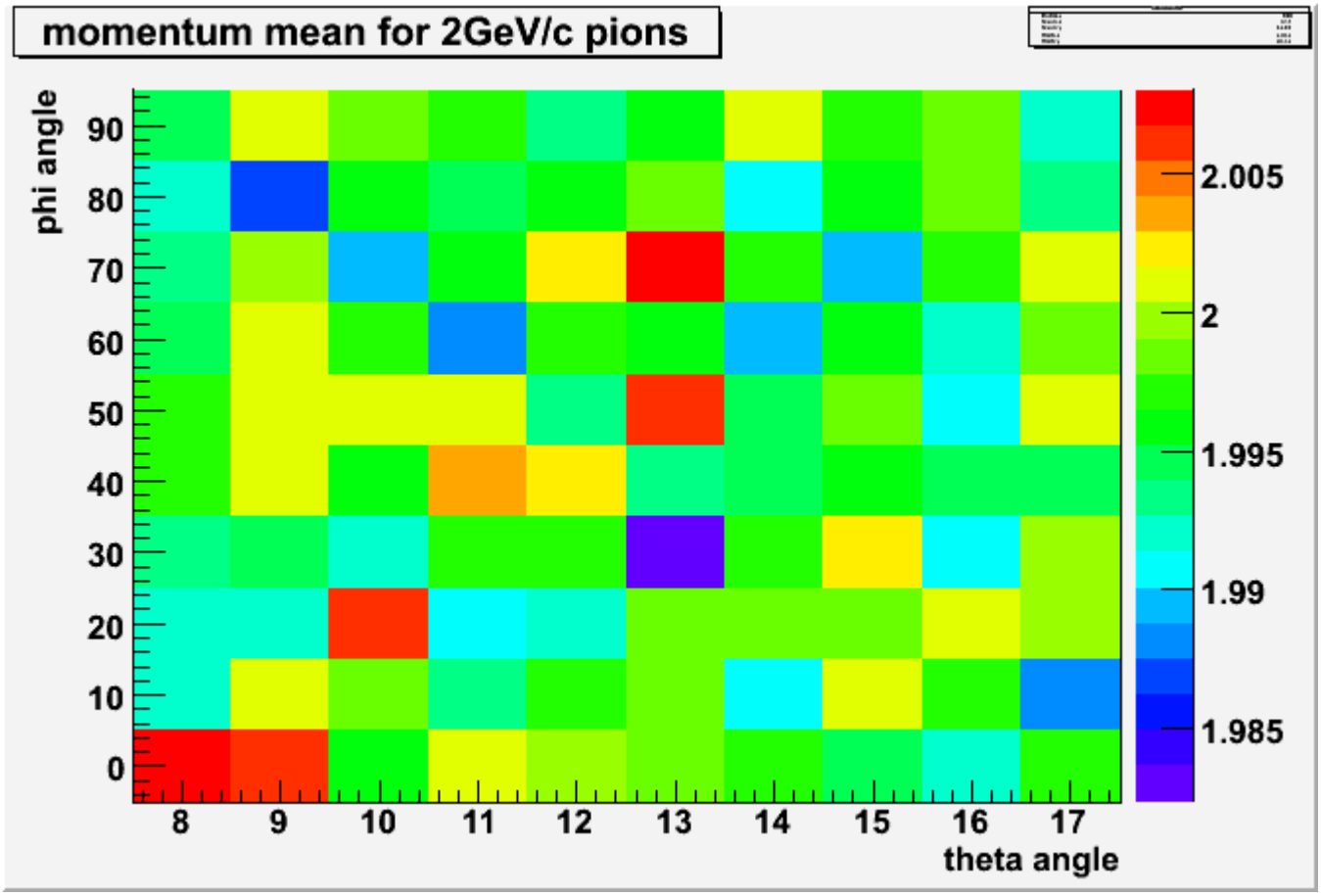
From the plot you can deduce that the larger the emission theta angle the better the momentum resolution. This is understood, since it also means the bigger curvature in the magnetic field. To put some numbers, it looks that momentum resolution varies between 4% and 8% depending on the theta emission angle.

More studies to come, but I would like to send special thanks to Ola W., Christian and Stefano, who particularly helped me to get the results.

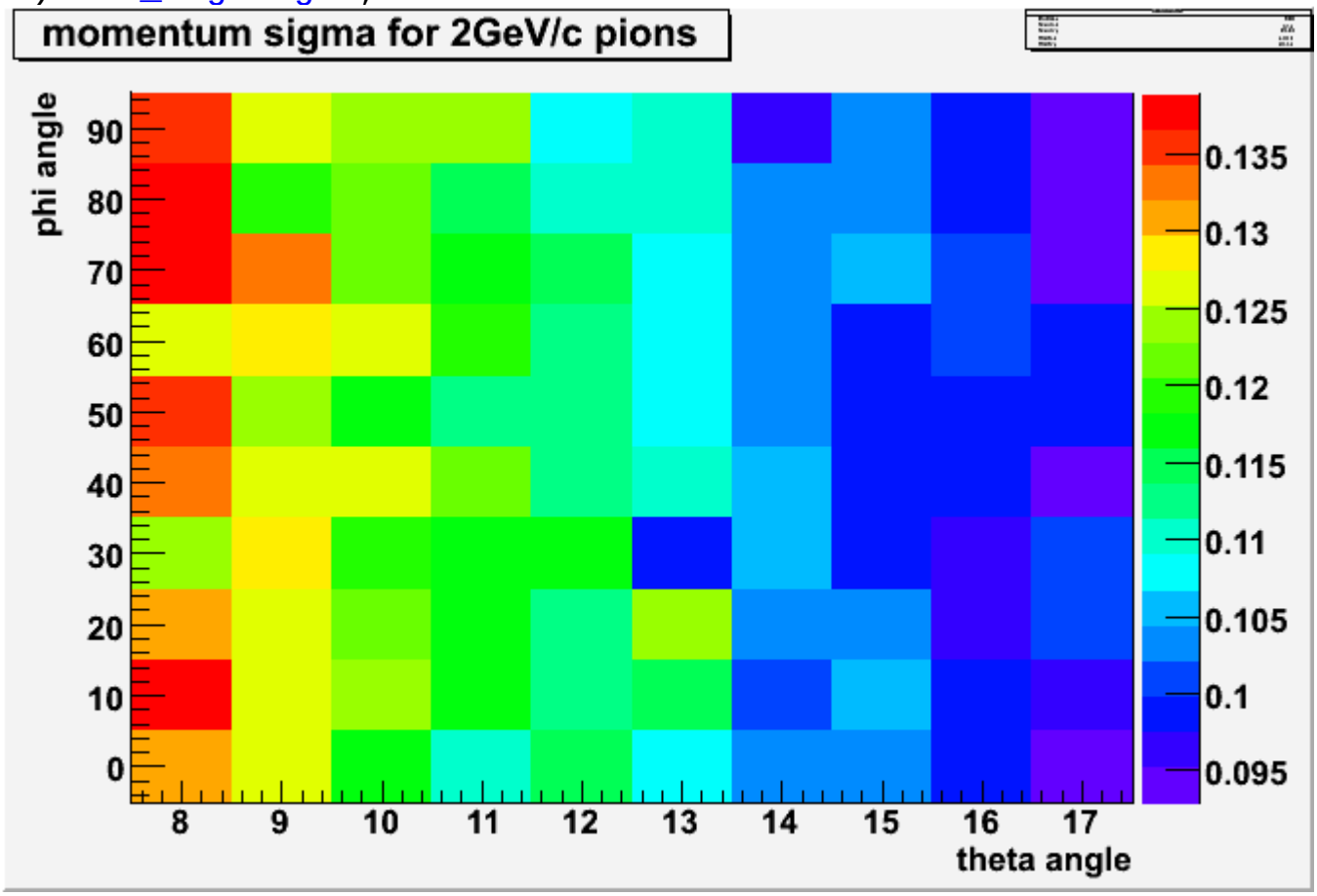
yours,
radek

File Attachments

1) mom_mean.gif, downloaded 1197 times



2) mom_sigma.gif, downloaded 1338 times



Subject: Re: GEM tracking
Posted by Johan Messchendorp on Fri, 26 Jun 2009 06:53:03 GMT
View Forum Message <> Reply to Message

very good news, indeed! Up to the next hurdle

Johan.