

Smearing of Parameters in the Truth Seeding

Shyam Kumar, Annalisa Mastroserio, Domenico Elia
University and INFN Bari, Italy

<https://github.com/eic/ElCrecon/issues/215>

Track Parameters (Global to Local)

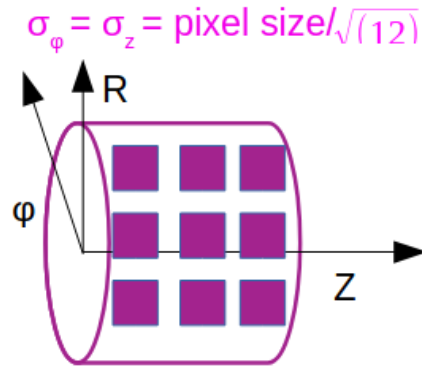
https://github.com/NicholasLukow/ATHENA_Tracking_GEM/blob/master/Fun4All_G4_Baseline2.C

Global coordinates: (x, y, z, p_x, p_y, p_z)

Track parameters on a surface (l_1, l_2 are local)

$$(l_0, l_1, \theta, \phi, q/p)$$

First two local parameters describes the pixel dimensions



Cylinder

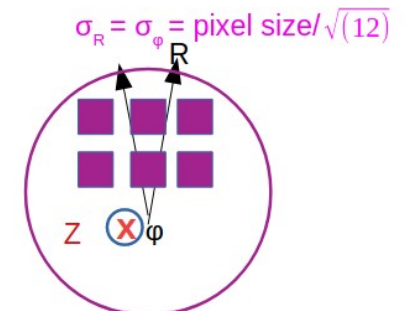
Track Parameters in Cylindrical surface:

$$(R\phi, z, \phi, \theta, q/p)$$

Cylinder (Fun4All)

Class PHG4TrackFastSim

```
PHG4TrackFastSim *kalman = new PHG4TrackFastSim("PHG4TrackFastSim");
// add Vertexing Layers
kalman->add_phg4hits(
    "G4HIT_SVTX",
    PHG4TrackFastSim::Cylinder,
    999.,
    pix_size_vtx/10000./sqrt(12.),
    pix_size_vtx/10000./sqrt(12.),
    1,
    0
    // const std::string& phg4hitsNames,
    // radial-resolution [cm]
    // azimuthal-resolution [cm]
    // z-resolution [cm]
    // efficiency,
    // noise hits
```



Disks

Track Parameters on Disk surface:

$$(R, R\phi, \phi, \theta, q/p)$$

Disks (Fun4All)

```
// add Disk Layers
kalman->add_phg4hits(
    "G4HIT_FBST",
    PHG4TrackFastSim::Vertical_Plane,
    pix_size_dis/10000./sqrt(12.),
    pix_size_dis/10000./sqrt(12.),
    999.,
    1,
    0
    // const std::string& phg4hitsNames,
    // radial-resolution [cm]
    // azimuthal-resolution [cm]
    // z-resolution [cm]
    // efficiency,
    // noise hits
);
```

General equation of a circle

$$(x - x_0)^2 + (y - y_0)^2 = R^2$$

If circle passed through the vertex (0,0): $x_0^2 + y_0^2 = R^2$

$$x^2 + x_0^2 - 2xx_0 + y^2 + y_0^2 - 2yy_0 = R^2$$

using

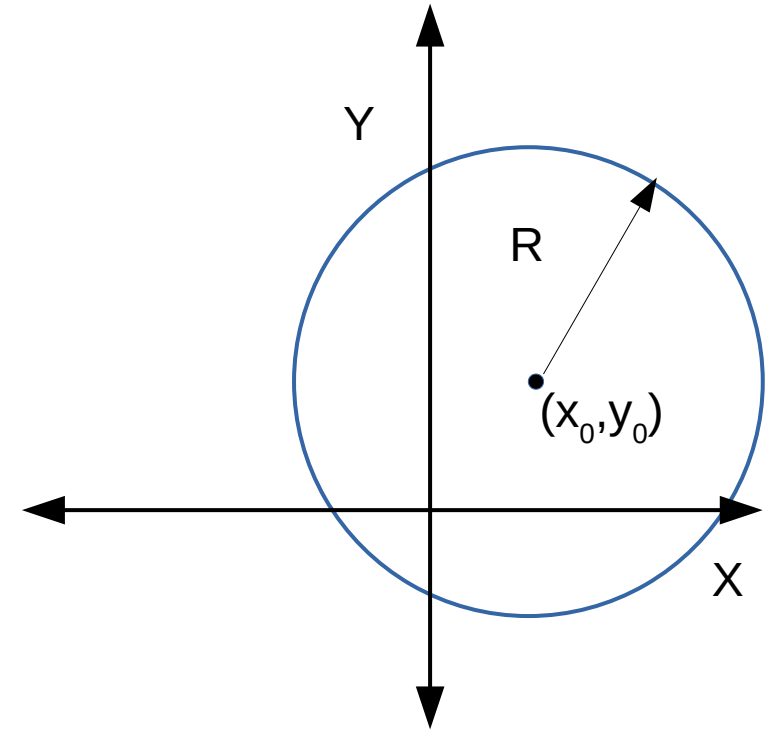
$$u = \frac{x}{x^2 + y^2} \quad v = \frac{y}{x^2 + y^2} \quad x_0^2 + y_0^2 = R^2$$

$$x^2 + y^2 - 2xx_0 - 2yy_0 = 0$$

$$1 - 2\frac{xx_0}{x^2 + y^2} - 2\frac{yy_0}{x^2 + y^2} = 0$$

$$2ux_0 + 2vy_0 = 1$$

Line in u-v space



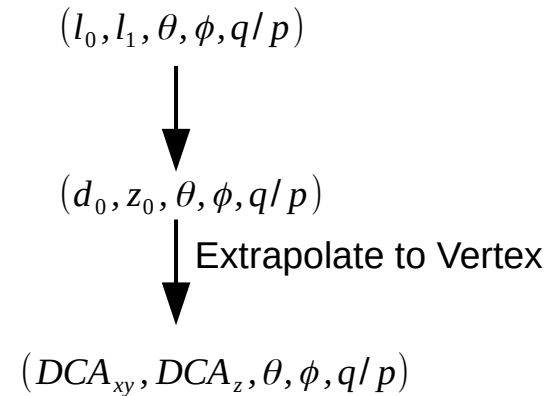
Find three hits belong to straight line in u-v space and use them as the seeding, later use combinatorial track finding and fitting using Kalman filter

How are the parameters initialized for fitting?

Truth Parameters in ACTS

```
// build some track cov matrix
Acts::BoundSymMatrix cov
cov(Acts::eBoundLoc0, Acts::eBoundLoc0) = Acts::BoundSymMatrix::Zero();
cov(Acts::eBoundLoc0, Acts::eBoundLoc1) = 1000*um*1000*um;
cov(Acts::eBoundLoc1, Acts::eBoundLoc1) = 1000*um*1000*um;
cov(Acts::eBoundPhi, Acts::eBoundPhi) = 0.05*0.05;
cov(Acts::eBoundTheta, Acts::eBoundTheta) = 0.01*0.01;
cov(Acts::eBoundQOverP, Acts::eBoundQOverP) = (0.1*0.1) / (GeV*GeV);
cov(Acts::eBoundTime, Acts::eBoundTime) = 10.0e9*ns*10.0e9*ns;

Acts::BoundVector params;
params(Acts::eBoundLoc0) = 0.0 * mm ; // cylinder radius
params(Acts::eBoundLoc1) = 0.0 * mm ; // cylinder length
params(Acts::eBoundPhi) = phi;
params(Acts::eBoundTheta) = theta;
params(Acts::eBoundQOverP) = charge / (pmag * GeV);
params(Acts::eBoundTime) = part->getTime() * ns;
```



Truth parameters are not smeared !!!

<https://github.com/eic/ElCrecon/blob/main/src/algorithms/tracking/TrackParamTruthInit.cc>

Truth Parameters in Fun4All

```
int PHG4TrackFastSim::process_event(PHCompositeNode* /*topNode*/)
```

```
int PseudoPatternRecognition(const PHG4Particle* particle,  
std::vector<PHGenFit::Measurement*>& meas_out, SvtxTrack* track_out,  
TVector3& seed_pos,  
TVector3& seed_mom, TMatrixDSym& seed_cov, const bool do_smearing = true);
```

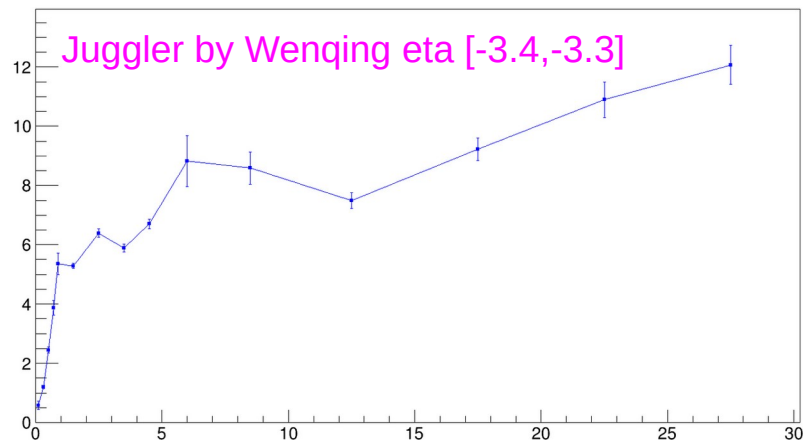
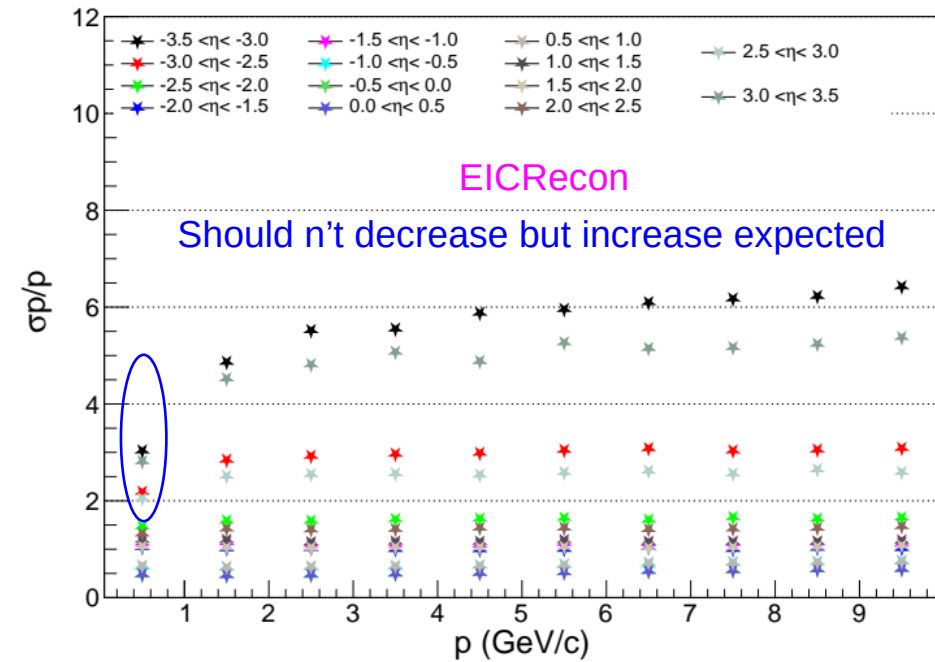
<https://github.com/SPHENIX-Collaboration/coresoftware/blob/master/simulation/g4simulation/g4trackfastsim/PHG4TrackFastSim.cc#L734>

10% smearing in p_{true}

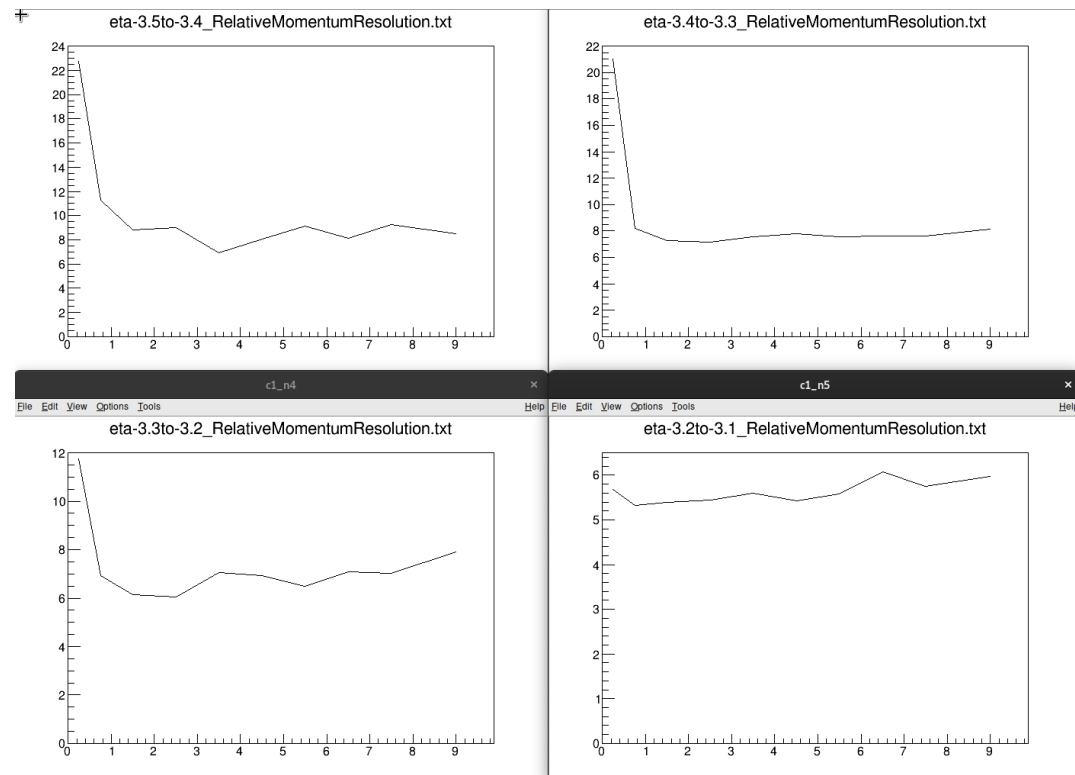
root [0] 3.0/180.*3.1416
(double) 0.052360000
0.05236 radian smearing in theta and Phi

```
int PHG4TrackFastSim::PseudoPatternRecognition(const PHG4Particle* particle,  
                                                std::vector<PHGenFit::Measurement*>& meas_out,  
                                                SvtxTrack* track_out,  
                                                TVector3& seed_pos,  
                                                TVector3& seed_mom, TMatrixDSym& seed_cov, const bool do_smearing = true)  
{  
    assert(track_out);  
  
    seed_cov.ResizeTo(6, 6);           Initialization default  
  
    seed_pos.SetXYZ(0, 0, 0);  
    // reset the seed resolution to the approximate position resolution of the last detector  
    seed_cov[0][0] = .1 * .1;  
    seed_cov[1][1] = .1 * .1;  
    seed_cov[2][2] = 30 * 30;  
    // for (int i = 0; i < 3; i++)  
    // {  
    //     seed_cov[i][i] = _phi_resolution * _phi_resolution;  
    // }  
  
    seed_mom.SetXYZ(0, 0, 10);  
    for (int i = 3; i < 6; i++)  
    {  
        seed_cov[i][i] = 10;  
    }  
  
    if (particle)                       If there is a truth information  
    {  
        TVector3 True_mom(particle->get_px(), particle->get_py(),  
                           particle->get_pz());  
  
        seed_mom.SetXYZ(particle->get_px(), particle->get_py(),  
                          particle->get_pz());  
  
        if (do_smearing)                 Option for smearing  
        {  
            const double momSmear = 3. / 180. * M_PI; // rad  
            const double momMagSmear = 0.1;           // relative  
  
            seed_mom.SetMag(  
                True_mom.Mag() + gsl_ran_gaussian(m_RandomGenerator,  
                                                    momMagSmear * True_mom.Mag()));  
            seed_mom.SetTheta(True_mom.Theta() + gsl_ran_gaussian(m_RandomGenerator, momSmear));  
            seed_mom.SetPhi(True_mom.Phi() + gsl_ran_gaussian(m_RandomGenerator, momSmear));  
        }  
    }  
}
```

Results



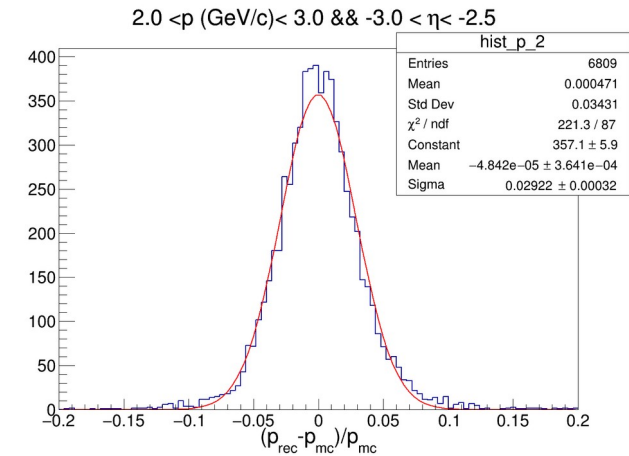
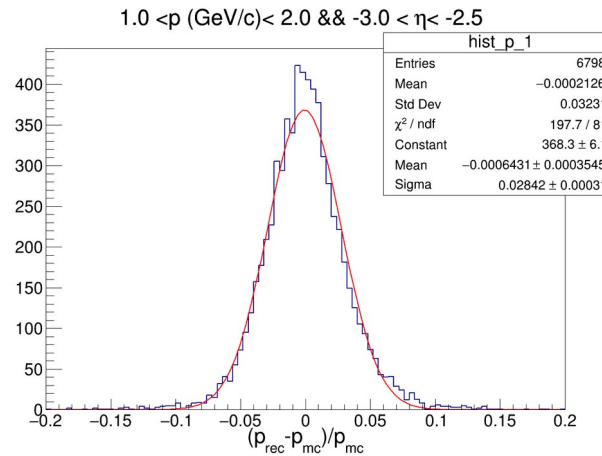
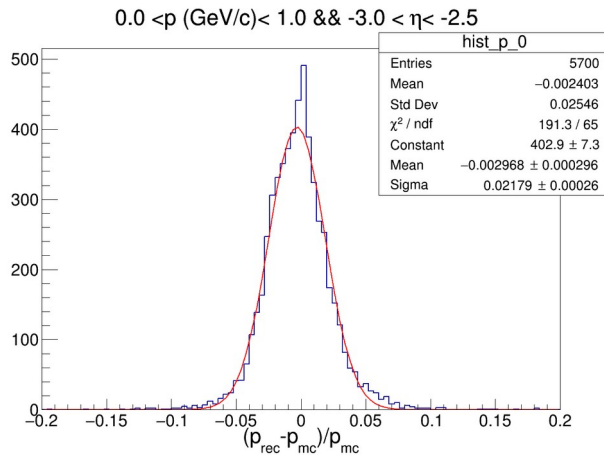
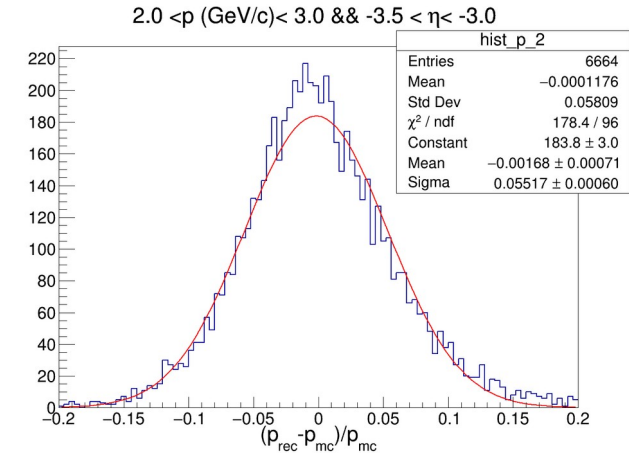
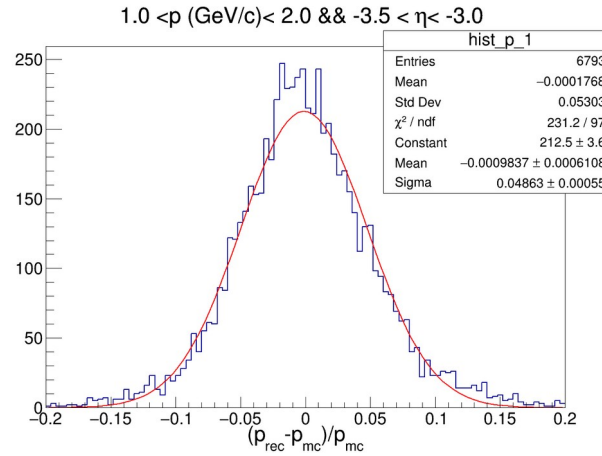
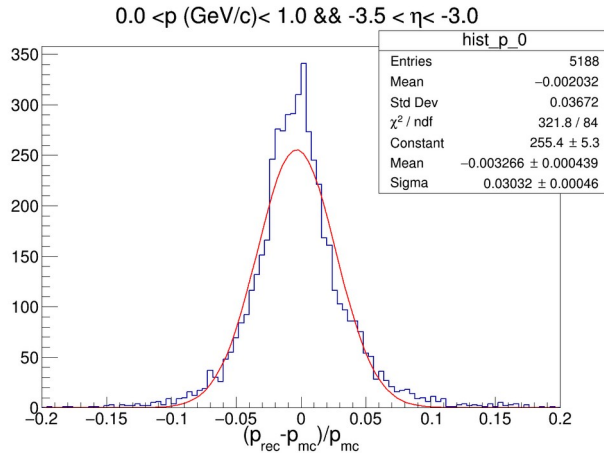
From Stephen using Fun4All (geometry is same)



Fun4All is giving correct results at low momentum !!

Few Debug Plots

At the very low momentum peak structure can be seen



Summary and Future Plan

- Origin of decrease of Momentum resolution at low momentum and higher eta is understood
- We need to come with proper smearing solution
- Need to cross check with smearing and realistic seeding the momentum resolution should increase at low momentum