

KF Unit Test Development

Valentina Akishina

Goethe-Universität Frankfurt, Frankfurt am Main, Germany
Frankfurt Institute for Advanced Studies, Frankfurt am Main, Germany

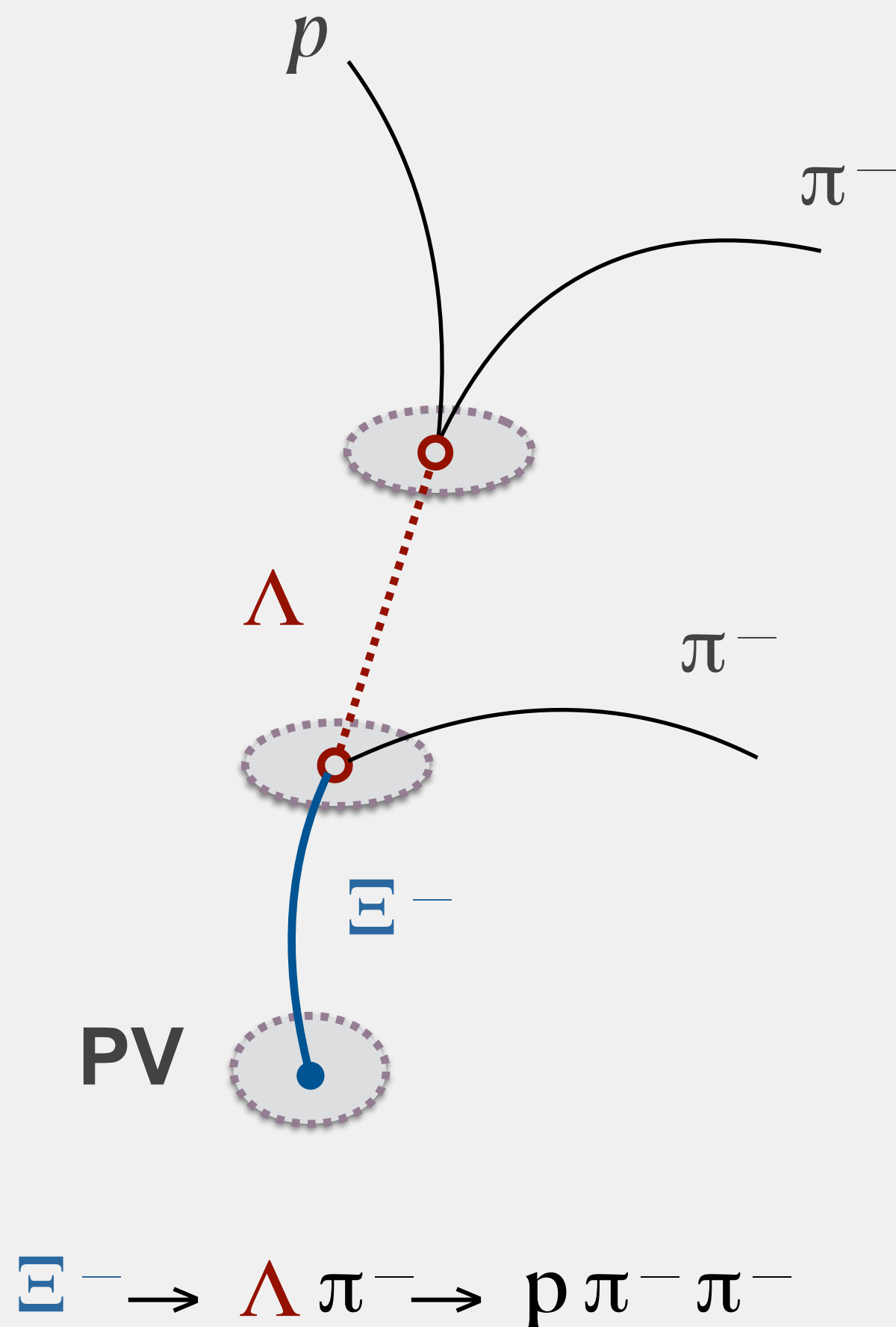
KFParticle Package

KF Particle package for reconstruction of short-lived particles:

- based on Kalman Filter method
- geometry-independent and portable to different experiments

Functionality:

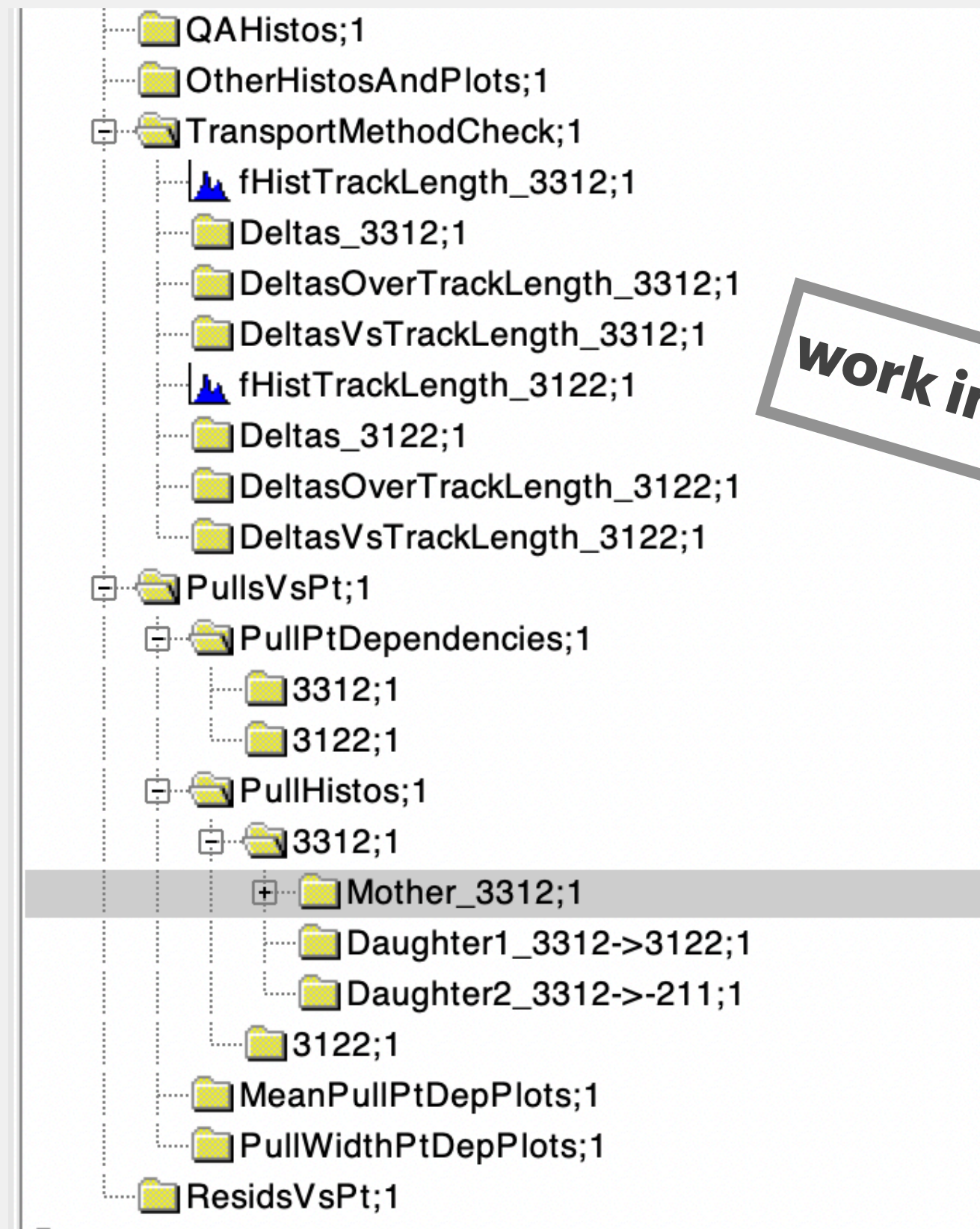
- Construction of the particles from tracks or another particles
- Decay chain reconstruction
- Transport of the particles
- Simple access to the particle parameters and their errors
- Calculation of the distance to point



Cascade Decays

Tree with MC decays (e.g. Ξ – cascade decay by Geant4)

Config.h: user-defined decays to study (mother, daughter PDGs)



<https://github.com/Volentina/KFParticle>



- MC Initial Tree
 - MC Final Tree
 - Reco Initial Tree
 - Reco Final Tree
- ↙ • Mother (PGD number)
 ↘ • Daughter



- Particle trees are used to fill histograms
- Currently the same procedure for all decays - to be changed (reco candidate should be used in future to study cascade effects)
- Smear MC daughter particle → reconstruct mother candidate
- Histograms are stored in folders with PDG number



How to organise constraints?

- Set Production Vertex (mother, daughters)
- Set Non-linear Mass constraint (mother, daughters)

Track Covariance Matrix Parametrisation

Parametrisation of daughter covariance matrix
as function of transverse momentum

```
std::vector<float> MakeCovMatrix(const MCParticleStruct& part){
    // 6x6 matrix has 21 indep. elements. Lets define all of them.
    std::vector<float> covmat(21);

    float pt = -1.;
    if (part.isMother)
        pt = part.finalPT();
    else
        pt = part.initialPT();

    covmat[0] = -3.59281e-06 + 1.02589e-05/(2.90359e-02+exp(-1.23947/pt)) ; //  $\sigma_{xx}$ 
    covmat[1] = -1.03361e-04 - 2.13155e-05/pt + 8.35099e-05*log(3.34901+(1./pt)) ; //  $\sigma_{xy}$ 
    covmat[2] = -6.17021e-06 + 1.21346e-05/(2.73856e-02+exp(-1.23947/pt)) ; //  $\sigma_{yy}$ 
    covmat[3] = 4.75842e-06 - 2.16231e-07/pt; //  $\sigma_{xz}$ 
    covmat[4] = 5.55892e-06 - 3.34455e-07/pt ; //  $\sigma_{yz}$ 
    covmat[5] = -1.24273e-04 + 2.09914e-04/(1.94591e-01+exp(-1.25084/pt)) ; //  $\sigma_{zz}$ 
    covmat[6] = -9.77974e-06 - 1.26268e-07*pow(1./pt,3) - 4.13286e-06*log(1./pt) ; //  $\sigma_{pxx}$ 
    covmat[7] = 1.04643e-06 + 5.47427e-09*pow(1./pt,3) + 1.02030e-07*log(1./pt); //  $\sigma_{pyp}$ 
    covmat[8] = -6.07308e-07 - 6.06728e-11*pow(1./pt,5) - 2.71488e-07*log(1./pt) ; //  $\sigma_{zpx}$ 
    covmat[9] = 2.46840e-05 + 1.73881e-06/(4.59998e-03+exp(-2.28556*pt)) ; //  $\sigma_{pxpx}$ 
    covmat[10] = 1.44735e-06 + 5.22400e-09*pow(1./pt,3) - 1.47596e-09*log(1./pt) ; //  $\sigma_{xpy}$ 
    covmat[11] = 2.43044e-04 - 9.99721e-08*pow(1./pt,3) - 8.62114e-05*log(1.79634e+01+1./pt); //  $\sigma_{ypy}$ 
    covmat[12] = -2.09525e-06 - 1.63962e-08*pow(1./pt,3) - 1.37700e-06*log(1./pt) ; //  $\sigma_{zpy}$ 
    covmat[13] = 7.32129e-05 + 2.35547e-05*pt - 7.30680e-05*log(2.63928e+00+pt) ; //  $\sigma_{pxpy}$ 
    covmat[14] = 2.15703e-05 + 1.3404e-06/(4.31348e-03+exp(-2.29888*pt)) ; //  $\sigma_{pypy}$ 
    covmat[15] = 3.98892e-07 - 1.70505e-09*pow(1./pt,3) + 8.76380e-08*log(1./pt) ; //  $\sigma_{xpz}$ 
    covmat[16] = 4.56639e-07 + 3.78791e-11*pow(1./pt,3) + 4.69908e-08*log(1./pt) ; //  $\sigma_{ypz}$ 
    covmat[17] = 4.60901e-04 - 1.04157e-07*pow(1./pt,3) - 1.85200e-04*log(1.35021e+01+1./pt) ; //  $\sigma_{zpz}$ 
    covmat[18] = -5.45974e-07 - 3.80233e-07*pow(pt,3) + 2.88266e-07*log(pt) ; //  $\sigma_{pxpz}$ 
    covmat[19] = 3.19847e-04 - 3.22758e-04*pow(1./pt,0.1) + 3.55611e-05*log(-4.93008e-02+1./pt); //  $\sigma_{ypyz}$ 
    covmat[20] = 1.57546e-05 + 2.39963e-06/(9.59981e-03+exp(-2.02381*pt)) ; //  $\sigma_{pzpz}$ 
```

No parametrisation of vertex covariance matrix

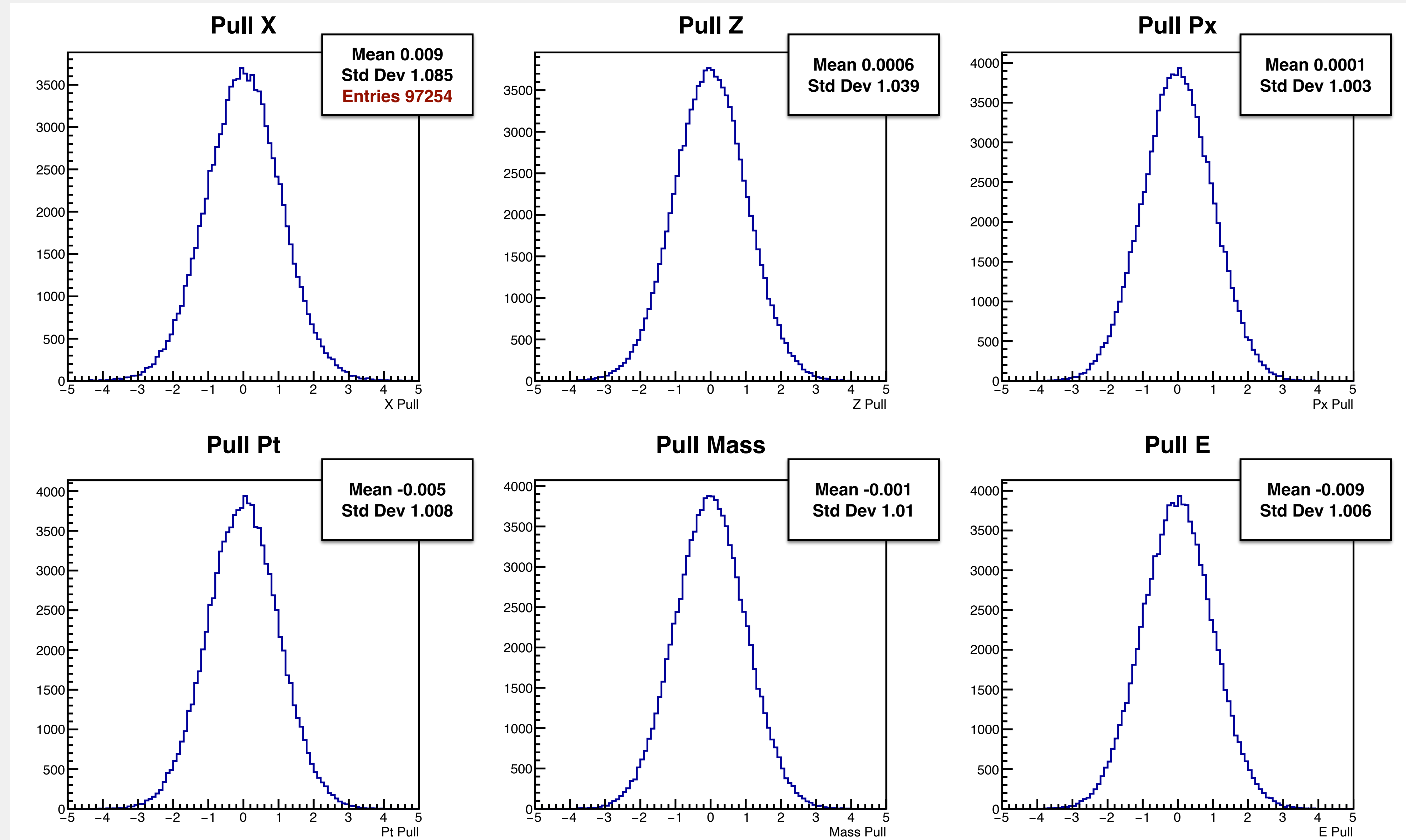
```
std::vector<float> MakeVertexCovMatrix(const MCParticleStruct& part){
    // 3x3 matrix has 6 indep. elements. Lets define all of them.
    std::vector<float> covmat(6);

    /*float pt = -1.;
    if (part.isMother)
        pt = part.finalPT();
    else
        pt = part.initialPT();*/

    covmat[0] = 1e-6; //  $\sigma_{xx}$ 
    covmat[1] = 1e-6 * (-1 + (double)rand() / RAND_MAX * 2); //  $\sigma_{xy}$ 
    covmat[2] = 1e-6; //  $\sigma_{yy}$ 
    covmat[3] = 1e-6 * (-1 + (double)rand() / RAND_MAX * 2); //  $\sigma_{xz}$ 
    covmat[4] = 1e-6 * (-1 + (double)rand() / RAND_MAX * 2); //  $\sigma_{yz}$ 
    covmat[5] = 1e-6; //  $\sigma_{zz}$ 
```

Pulls of D^0 at decay point

Pull – normalised errors: $P_x = \rho_x / \sqrt{C_{xx}}$



Study of diverged covariance matrix

- Particles only get rejected if CheckCovMatrix fails (positive diagonal elements, non-diagonal: $c_{ij}^2 > c_{ii}c_{jj}$)
- Bug in CheckCovMatrix cached (wrong calculation of diagonal element index)
- Particle smearing are done with matrix correction
- Still some decays diverge after SetProductionVertex of mother