

V0 analytical selection

Marian Ivanov, Alexander Kalweit

Overview

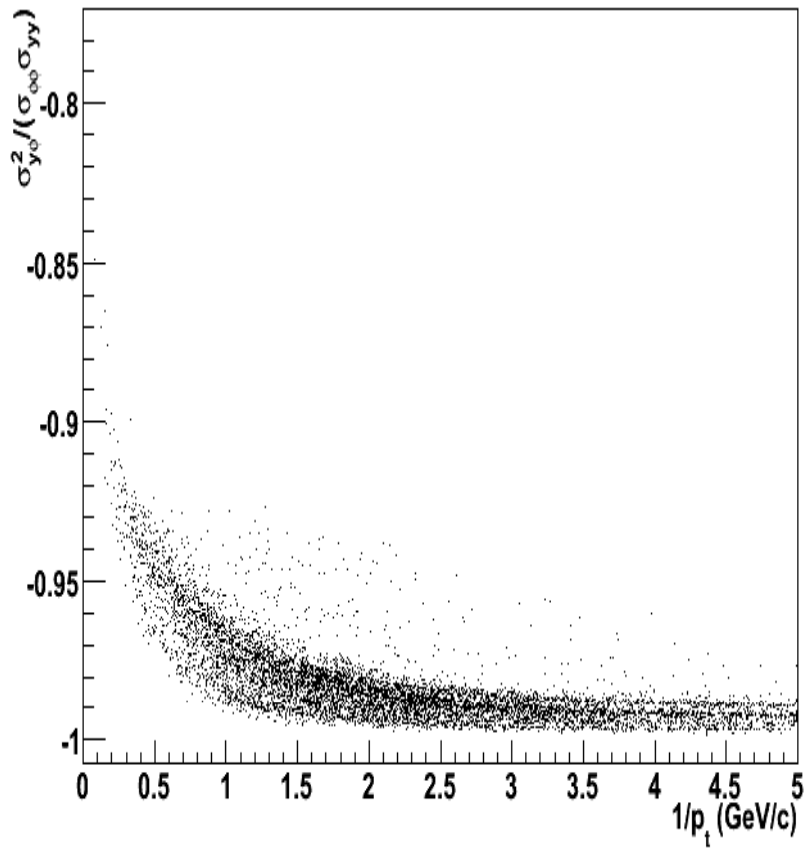
- Current status
- Kalman filter vertexing (I.Kisel, S.Gorbunov)
- Error parametrization
- Further plans

Current status - Approaches

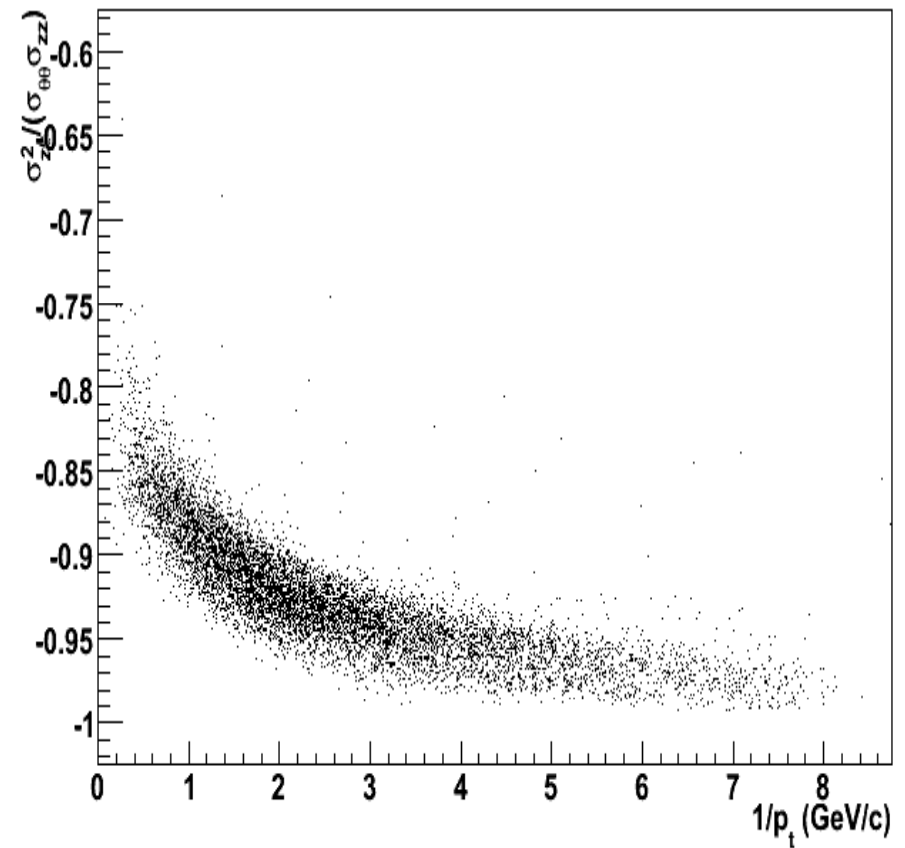
- Selection variable
 - Primary tracks selection (removal)
 - DCA, Pointing Angle
 - Causality (Online vertexing)
- Offline vertexing
 - Fixed cuts – selection
- Online vertexing
 - Cuts in terms of sigma of resolution
- Both cases the correlation between parameters not take into account
 - Linear discriminant method – not parametric

Correlation

Tracks[j].fC[3]/sqrt(Tracks[j].fC[0]*Tracks[j].fC[5])-abs(Tracks[j].fP[4]) (Tracks[j].fTSncls>5&&abs(Tracks[j].fX)<1)

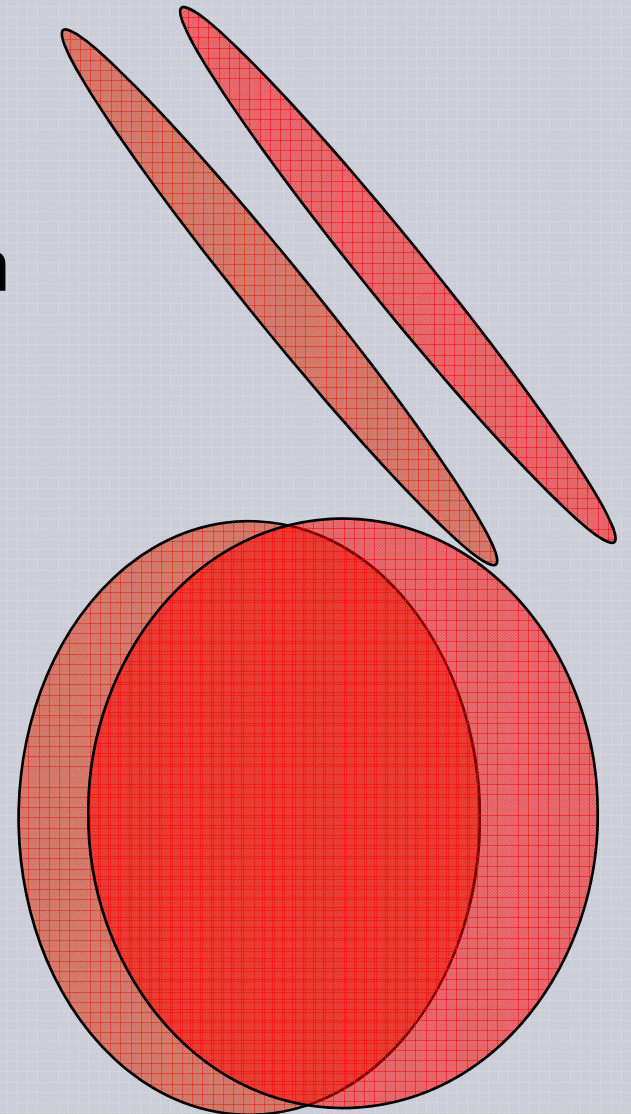


Tracks[j].fC[7]/sqrt(Tracks[j].fC[2]*Tracks[j].fC[9])-abs(Tracks[j].fP[4]) (Tracks[j].fTSncls>5&&abs(Tracks[j].fX)<1)



Proposed solution

- Use the selection based on resolution
- Take into account correlation
- The combinatorial background can be reduced by factor of volume reduction
 - => Use Kalman filter technique
- Chi2 selection define ellipsoid
 - the area of chi2 ellipsoid much smaller than enveloped ellipsoid
 - case of independent cuts



Proposed solution – KF (0)

- KF approach – used in the TPC dEdx calibration
 - To provide different S/N background – different χ^2 selection used
 - Works only to certain extend
 - The tracks with big covariance matrix are preferred
- Next improvement – instead cutting on χ^2 – cut on signal over background ratio - $\log(\text{likeS}/\text{likeB})$

Proposed solution – KF (1)

- Signal likelihood
 - $p_S(\chi^2) \sim \chi^2(\chi^2, \text{ndf})$
 - $\sim \exp(-\chi^2)$ – case of 1 NDF
 - In general small deviation from the χ^2 distribution observed (multiple scattering non gaussian) – General $p(\chi^2)$ preferred
- Background likelihood – (Product of space dependent occupancy and volume)
 - $p_B(\chi^2, \sigma) \sim N_{\text{part}}/r * \chi^2 * (s_y * s_z * s_{\phi} * s_{\theta})$
- Selection formula $\log(p_S(\chi^2)) - \log(p_B(\chi^2)) > \text{cut}$

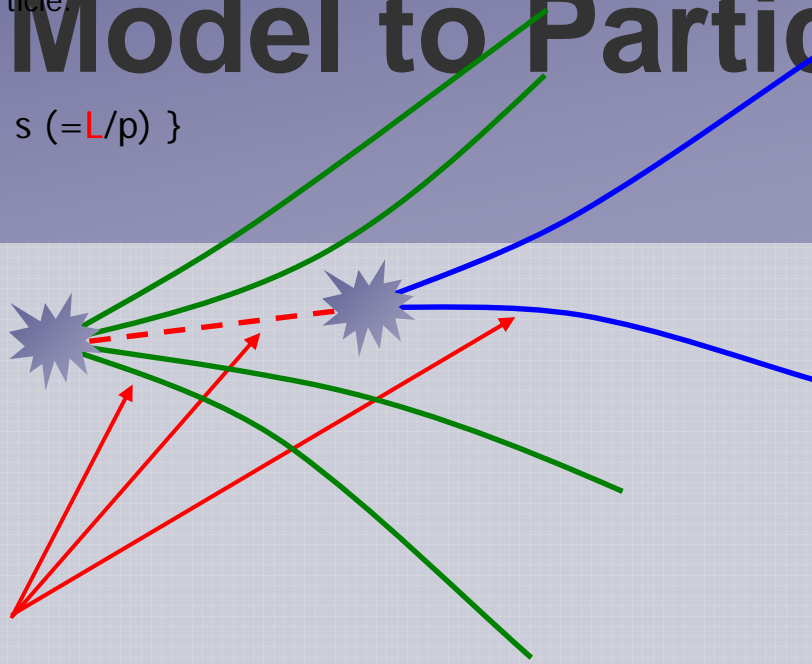
Track model focuses on geometrical trajectory of a particle.

Particle model contains full description of the particle.

From Track Model to Particle Model

Particle:

- state vector $r_{[8]} = \{ x, y, z, p_x, p_y, p_z, E, s (=L/p) \}$
- covariance matrix $C_{[8 \times 8]}$



All tracks in AliKFParticle are particles.

Two main reasons for that:

- This is a natural physics parameterization.
- A more general 3D track model, suitable for forward (CBM) and collider (ALICE) experiments.

Other technical advantages:

- all physics parameters are provided with their errors;
- other parameters and their errors can be easily calculated from the state vector;
- the state vector contains particle parameters at the production and decay vertices;
- suitable for vertex fit only;
- fixed size of the state vector;
- the state vector is measured directly by daughter tracks;
- natural treatment of a decay chain;
- ...

KF Algorithm (I.Kisel – CERN

July 11.07)

$$r_{[8]} = \{ x, y, z, p_x, p_y, p_z, E, s (=L/p) \}$$

1. Initial approximation v^0 of the decay vertex.
2. Transport of the k-th daughter particle r_k^d, C_k^d into the initial vertex position v^0 ; construction of the parameters of the daughter particle at the decay point.
3. If it is necessary to select daughter tracks, then the χ^2 probability of the fact that the k-th particle r_k^d is daughter particle is calculated.
4. Measurement of the state vector r_{k-1} by the k-th daughter particle adding the 4-momentum of the daughter particle to the 4-momentum of the mother particle.
5. Repeat from the step 2 for the next daughter particle, until all the daughters are treated.
6. Precision of the particle parameters obtained after the fit can be improved in the case of invariant mass M of the particle is known. In this case the parameters of the particle are measured by the one-dimensional measurement with the measured value M , the null error and the corresponding measurement matrix, using the Kalman filter.
7. If the production vertex is given, then the mother particle can be propagated into the production point and added to the production vertex, thus improving its resolution.
8. If the production vertex is given, then the constructed mother particle is transported into the production point and then is measured by the production vertex.

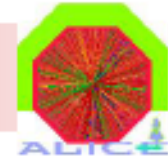
Since the problem is nonlinear, the algorithm is iterated several (3) times.

After estimation of the parameters of the particle, additional physics parameters, which are not explicitly included into the state vector, the particle momentum P , the invariant mass M , the length of flight L in the laboratory coordinate system, and the time of life of the particle cT in its own coordinate system:

$$\begin{aligned} P &= \sqrt{p_x^2 + p_y^2 + p_z^2}, \\ M &= \sqrt{E^2 - P^2}, \\ L &= s \cdot P, \\ cT &= s \cdot M, \end{aligned}$$

and the corresponding errors can be easily calculated.

AliKF: Unconstrained Fit



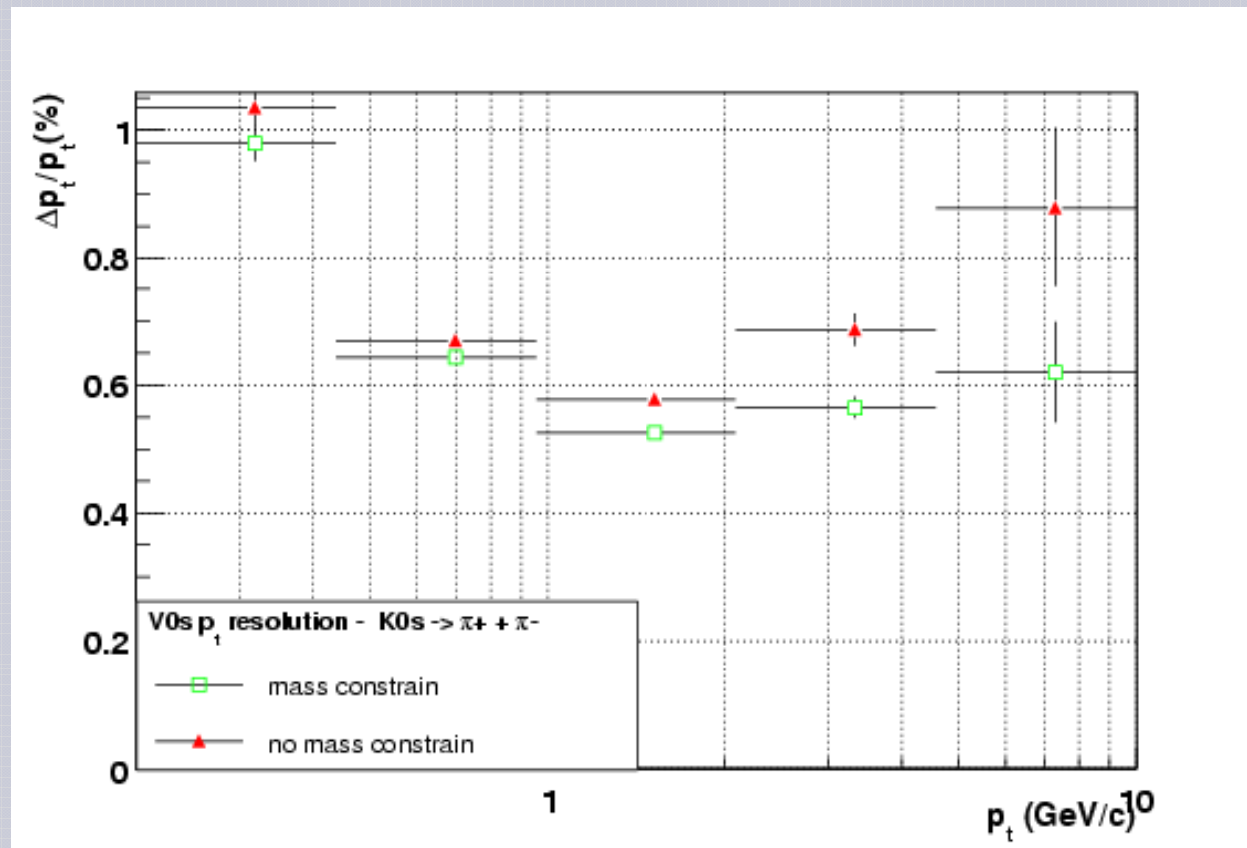
1 Create AliKFParticles:

```
AliKFParticle Proton(*ESDproton,2212);  
AliKFParticle Kaon(*ESDkaon,321);  
AliKFParticle Pion(*ESDpion,211);  
AliKFParticle Lambdac(Kaon,Pion,Proton);
```

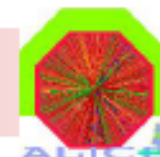
2 Use functionalities:

```
GetX(),...  
GetPx(),...  
GetCovariance(int i)  
GetQ()  
GetChi2()  
GetNDF()  
GetMass(Double_t &M, Double_t &SigmaM)
```

Momentum Resolution with Mass Constraint



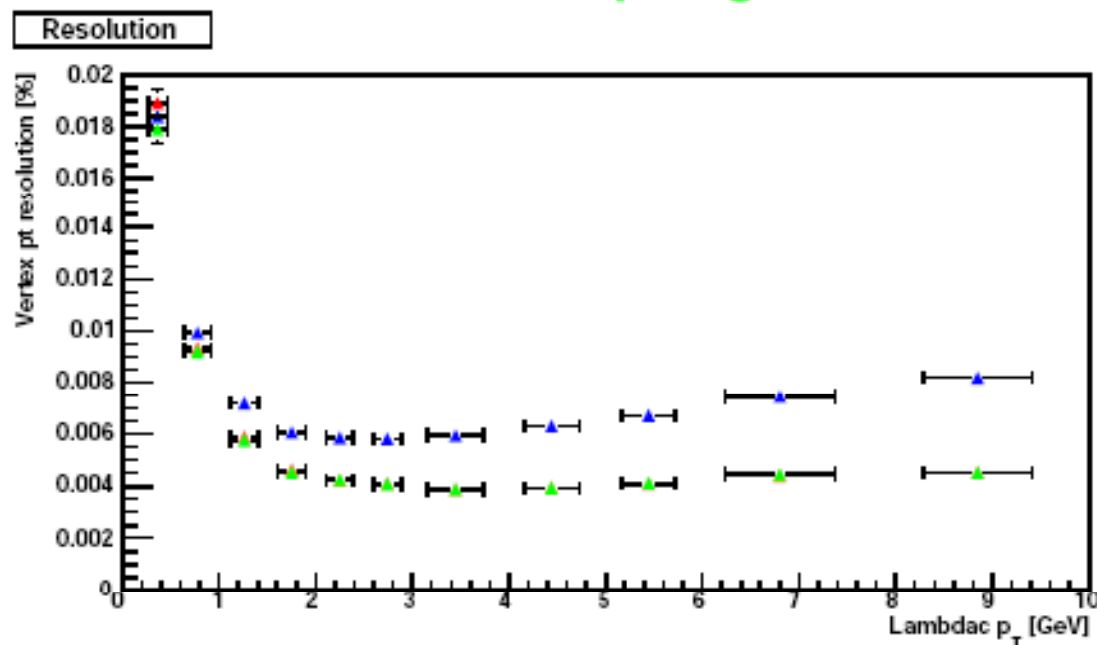
Pt Resolution



No constraints

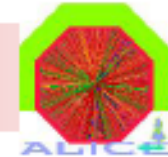
Mass constrained fit

Fit with topological constraint



SIGNIFICANT IMPROVEMENT!!!

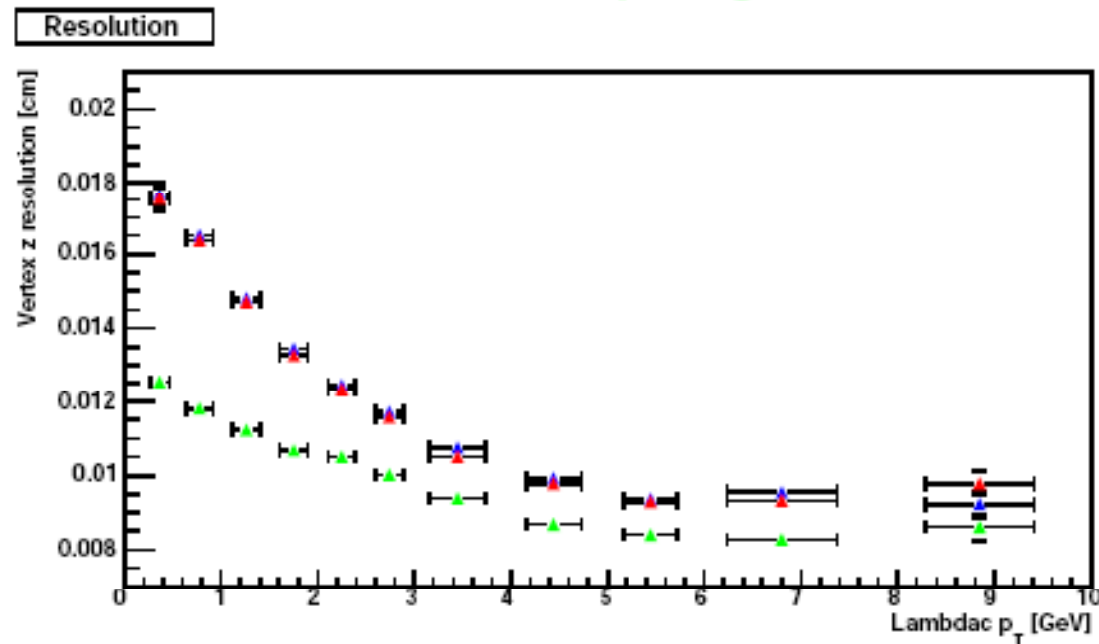
Lambdac Resolution: z



No constraints

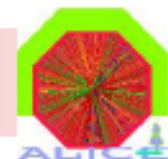
Mass constrained fit

Fit with topological constraint



x,y,z: VERY IMPORTANT IMPROVEMENT!!!

After Topological Constraint

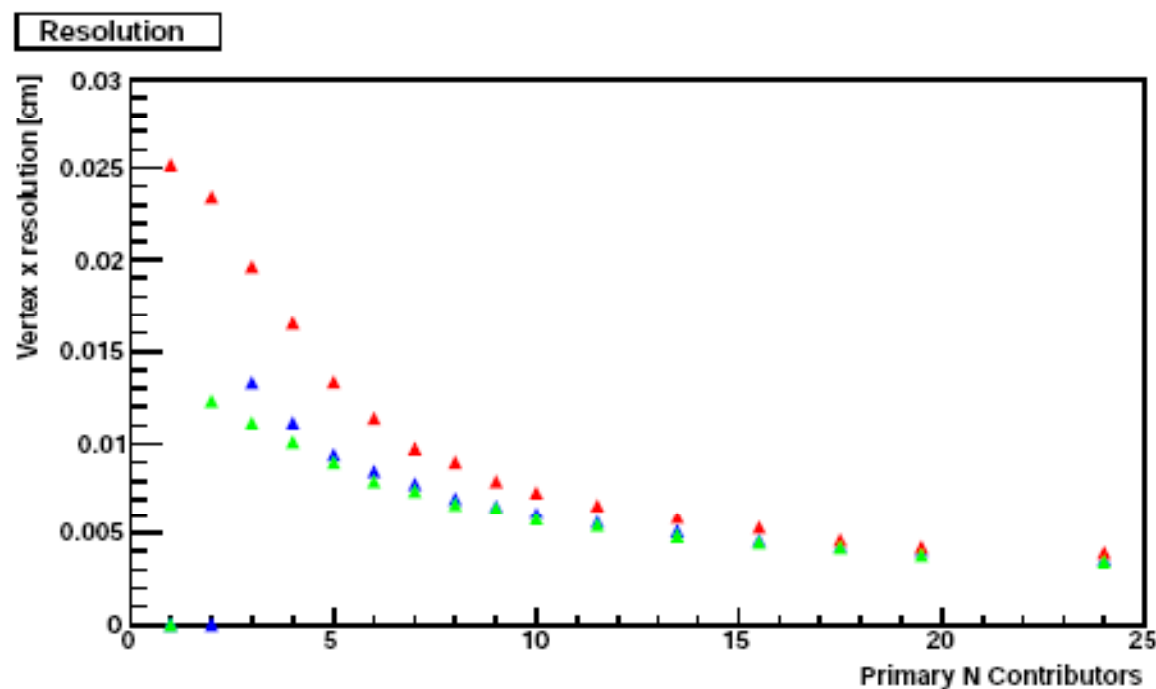


Λ_c with multiple interactions: 600,000 events

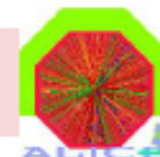
x: NContributors

x: NContributors - Λ_c daughters

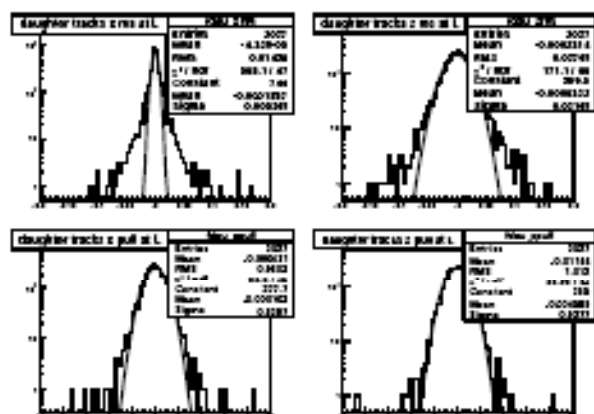
x: NContributors - Λ_c daughters + Λ_c mother track



Track Errors and Pulls



ESD daughter tracks extrapolated
to Λ_c MC decay vertex



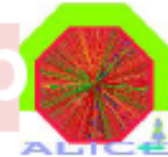
	res. (μm)	pull
x	53	0.84
y	51	0.84
z	216	0.93

Track
errors
overestimated!!

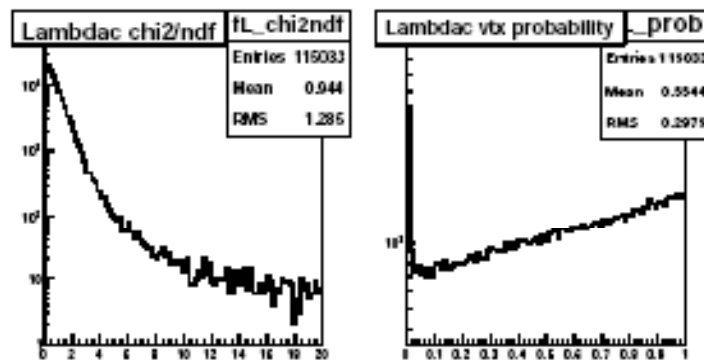
NOW:

	res. (μm)	pull
x	47	0.92
y	47	0.92
z	200	0.96

Secondary Vertex Pulls and Prob



Λ_c

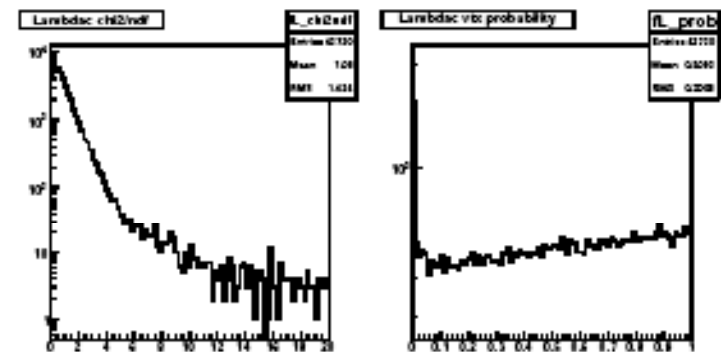


	res. (μm)	pull
x	111	0.91
y	112	0.92
z	137	0.99

prob not flat

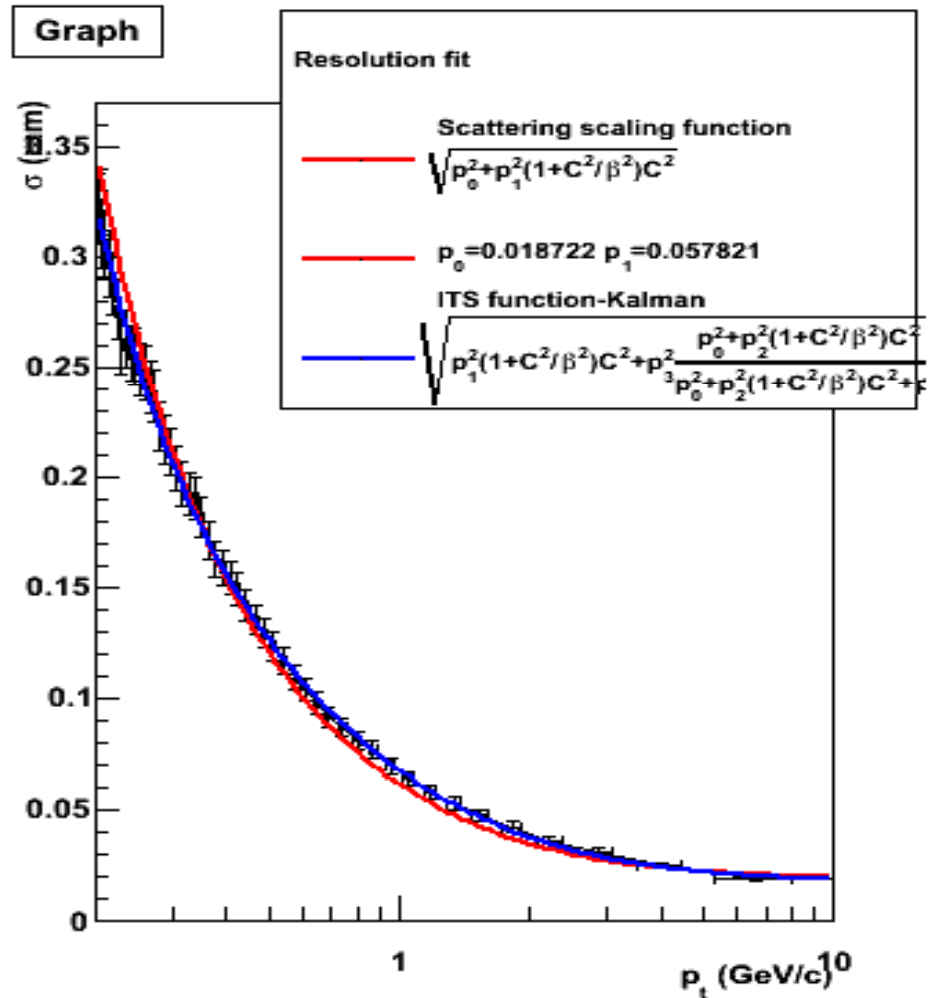
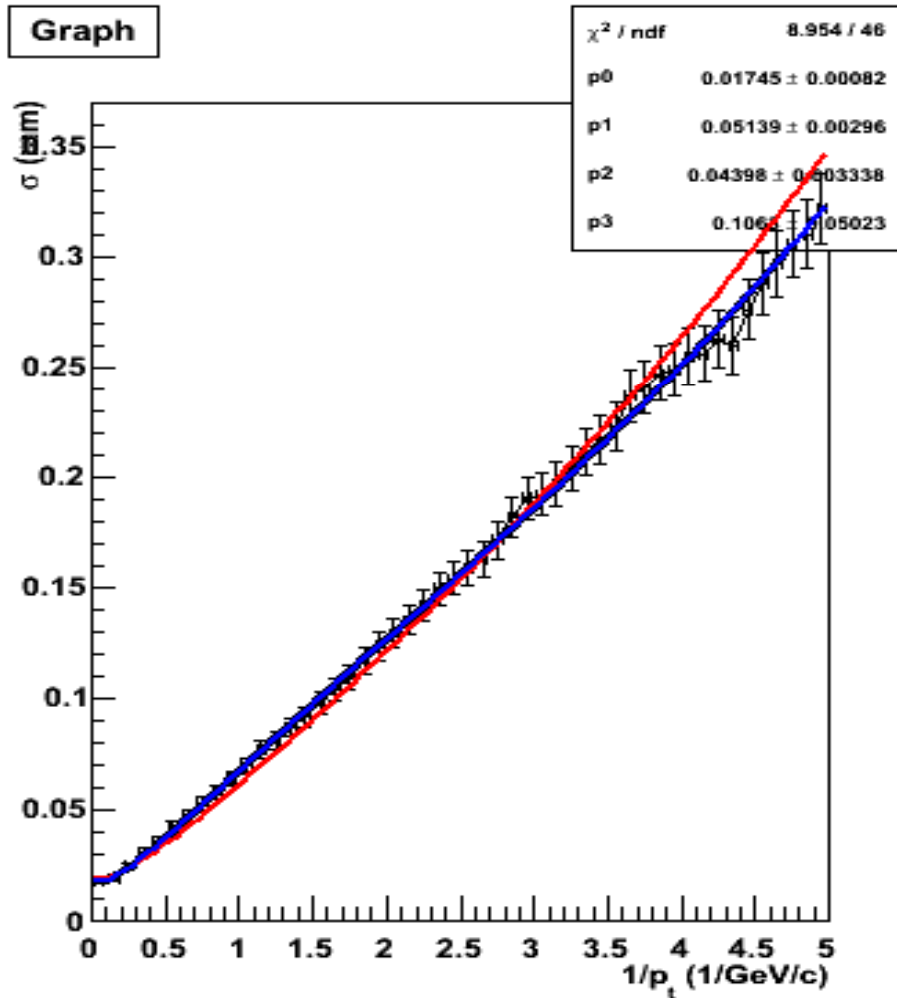
pulls < 1

NOW

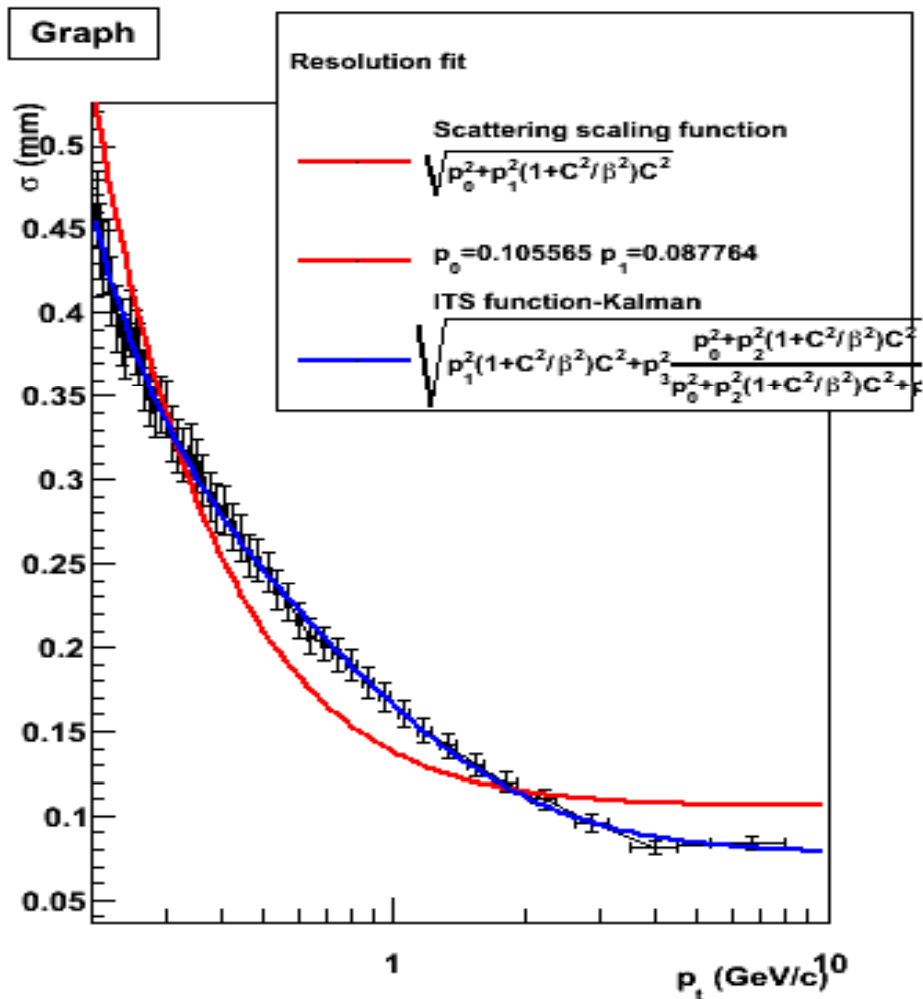
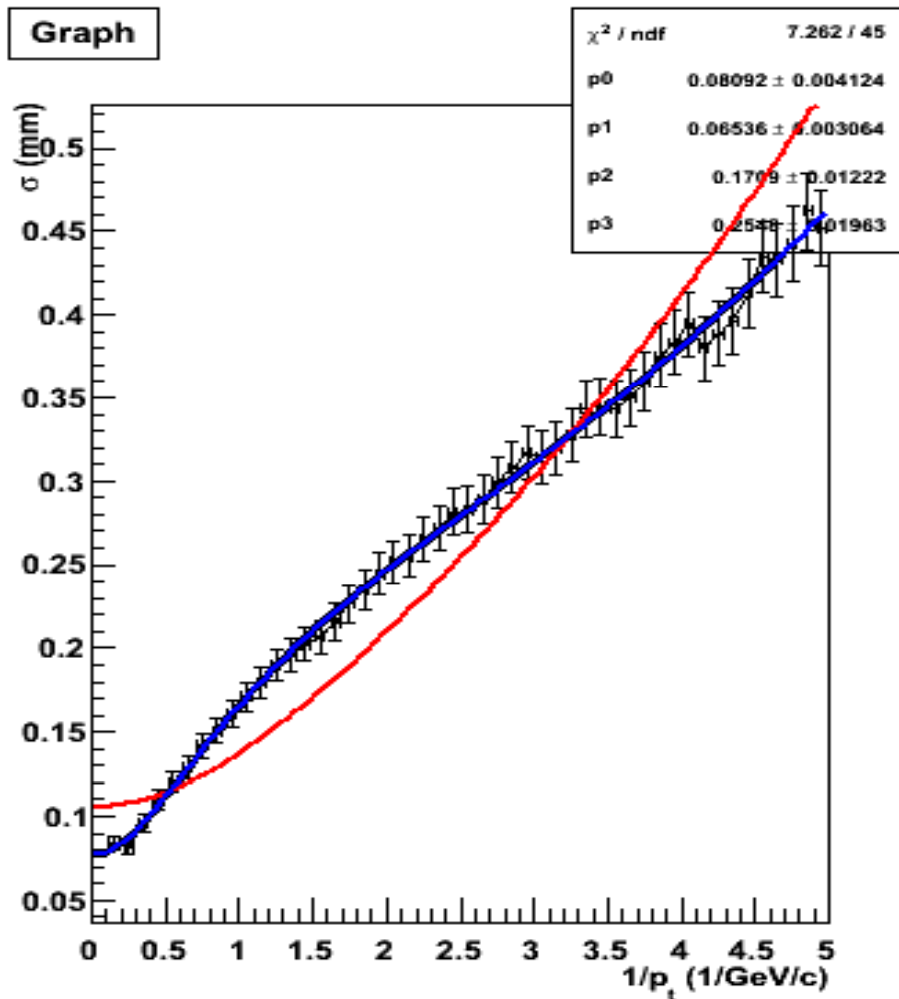


	res. (μm)	pull
x	100	0.98
y	100	0.98
z	125	1.02

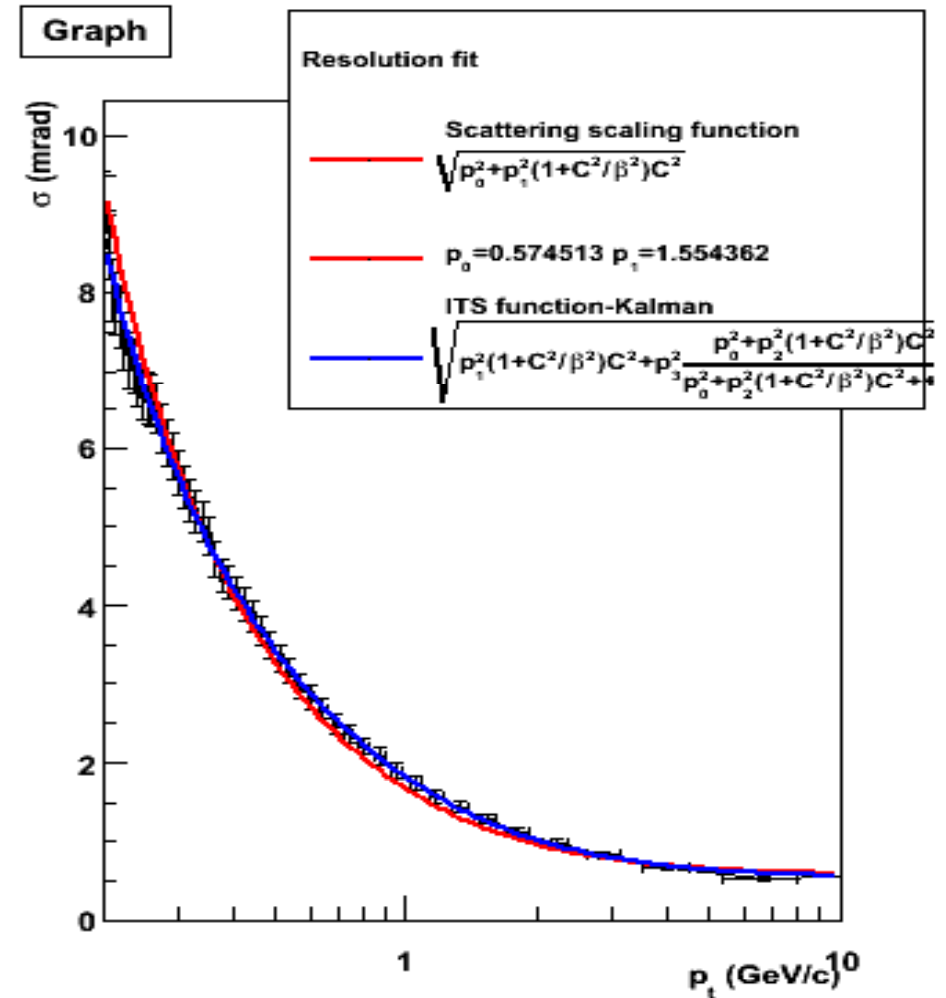
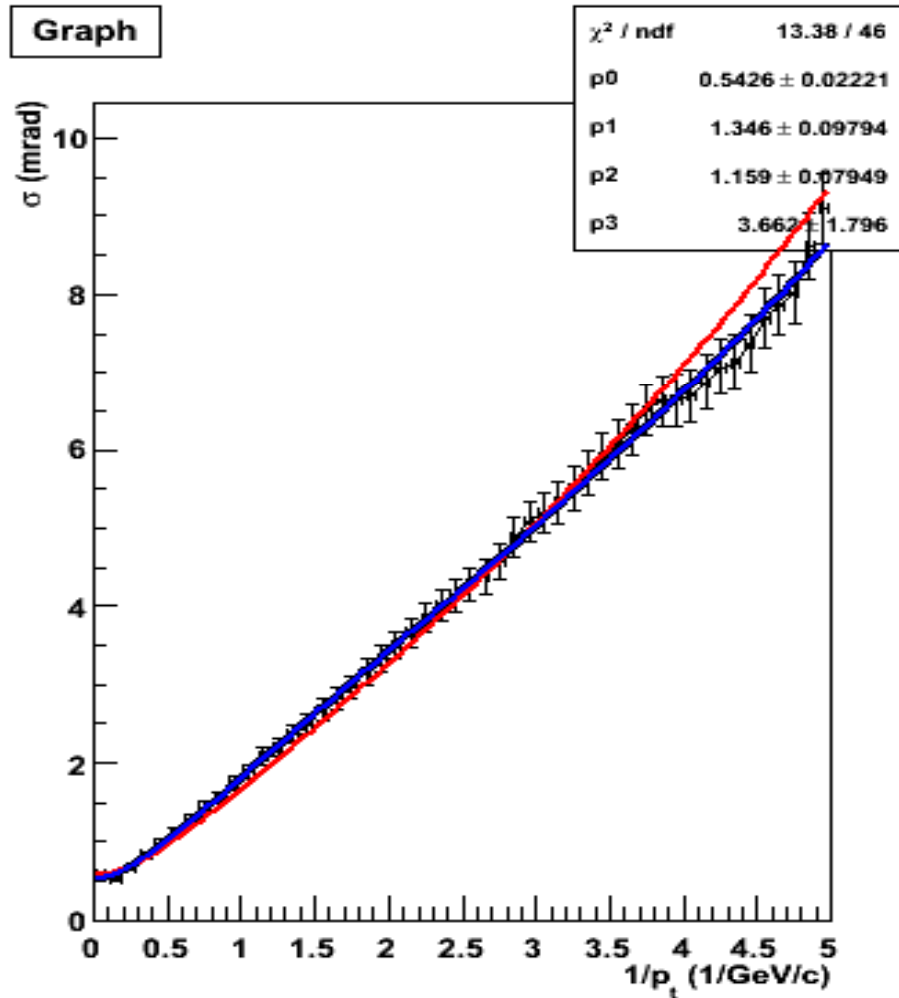
ITS y resolution



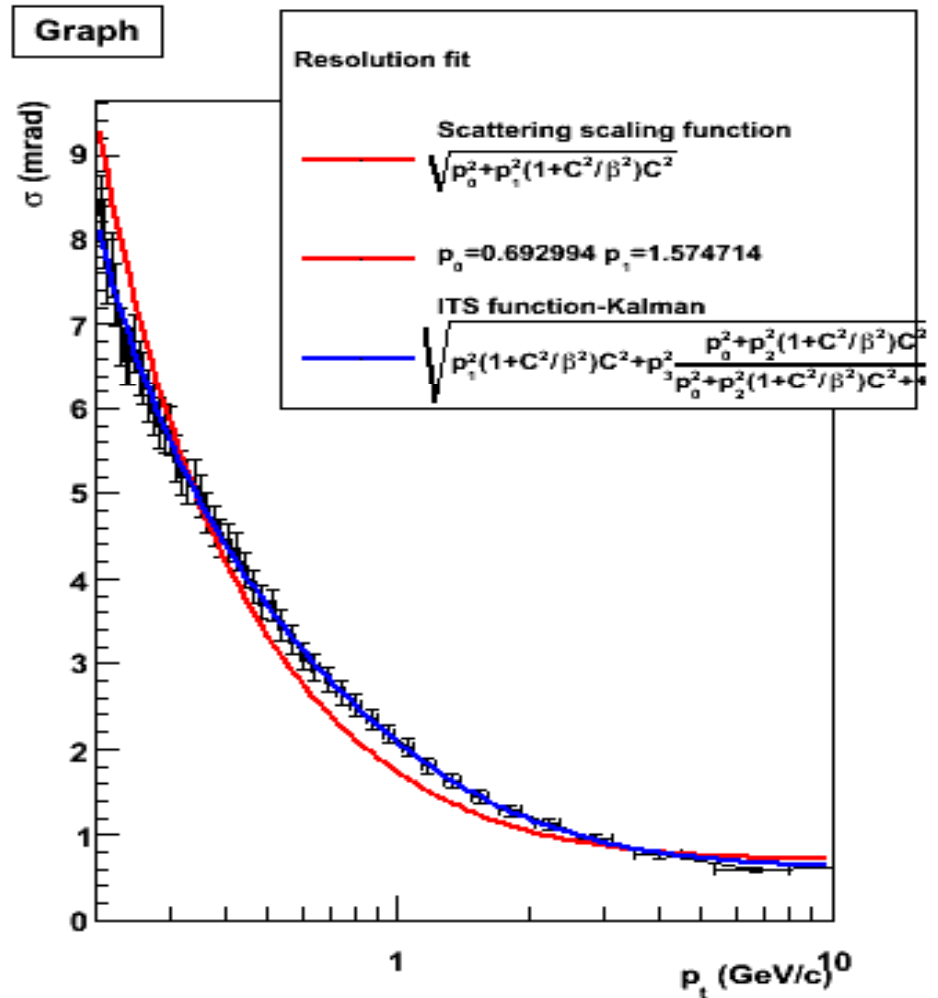
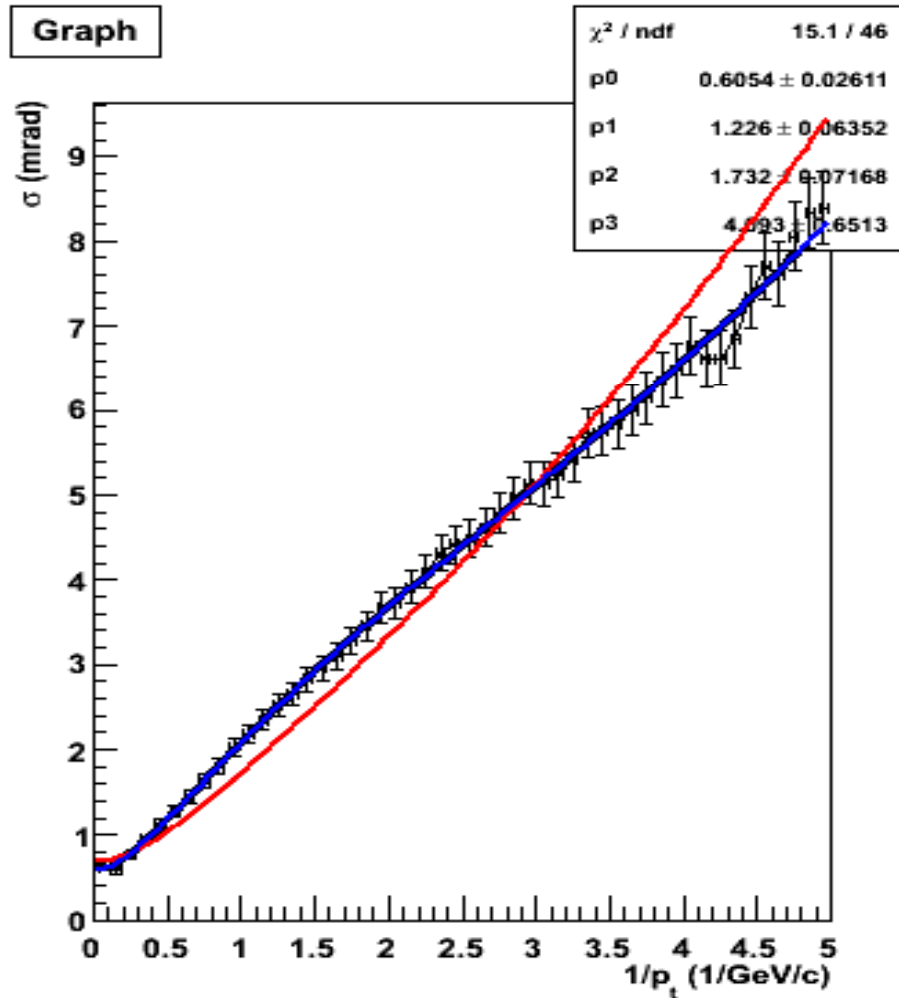
ITS z resolution



ITS phi resolution



ITS theta resolution



Further plans

- The strategy – usage of AliKF Package defined
- Tested on already preselected data
- Next step - Use it directly during V0 finder
- AliESDv0Cuts
 - Class to be used in the Online and Offline vertexing (Problem to be solved – extract correct covariance matrix in case of offline vertexer)
 - The same class planned to be used in User analysis (tightened criteria + PID)