# Using GEANE in the VMC

**A. Fontana**, P. Genova, L. Lavezzi, A. Panzarasa and A. Rotondi
*INFN-Pavia and University of Pavia*

*M. Al-Turany and D. Bertini*
*GSI*

- Brief review of the package GEANE.

- Updates to TGeant3.

- Updates to GEANE:
  - Coulomb *multiple scattering*: new parametrization.
  - Straggling in energy loss for gaseous materials: inclusion of *Urban model* for sub-Landau regime.
  - New option to extrapolate to *point of closest approach*.

- Results from the PANDA Straw Tube Tracker: pull distributions.

- Application for a GEANE-based Kalman Filter.

# What is GEANE?

- Package to calculate the **average trajectories** of particles through **dense materials** and to calculate the **transport matrix** as well as the propagated errors **covariance matrix** in a given track representation.

- It is a **track follower**: it predicts the trajectory of a charged particle in terms of mean values and errors both in forward and in backward direction. **Three effects** are taken into account:
  - energy loss (affects mean values and errors)
  - Coulomb multiple scattering (affects errors only)
  - magnetic field (affects mean values only)

This has to be checked against a real MC where one generates an ensemble of particles and compares the mean and sigma of the distributions with the predicted ones (**pull distributions**).

# Track parameters in GEANE

**Three** different **track representations** (beam directed along x axis):

- **Central System (SC, representation I)**
  $1/p, \lambda, \phi, y, z$
  Usually applied in the overall reference frame

- **Detector System (SD, representation II)**
  $1/p, v', w', v, w$
  where (u,v,w) orthonormal reference system with vw plane coincident with the detector plane.
  Usually applied when the trajectory has to be evaluated at different detector planes (colliding beam experiments where planes can take great variety of directions).

- **Spline System (SP, representation III)**
  $1/p, y'=py/px, z'=pz/px, y, z$
  Usually applied in fixed target experiments where the trajectory is evaluated on successive parallel planes (perpendicular to x axis)

GEANE provides routines to **switch** from one representation to the other.

# Track parameters in GEANE

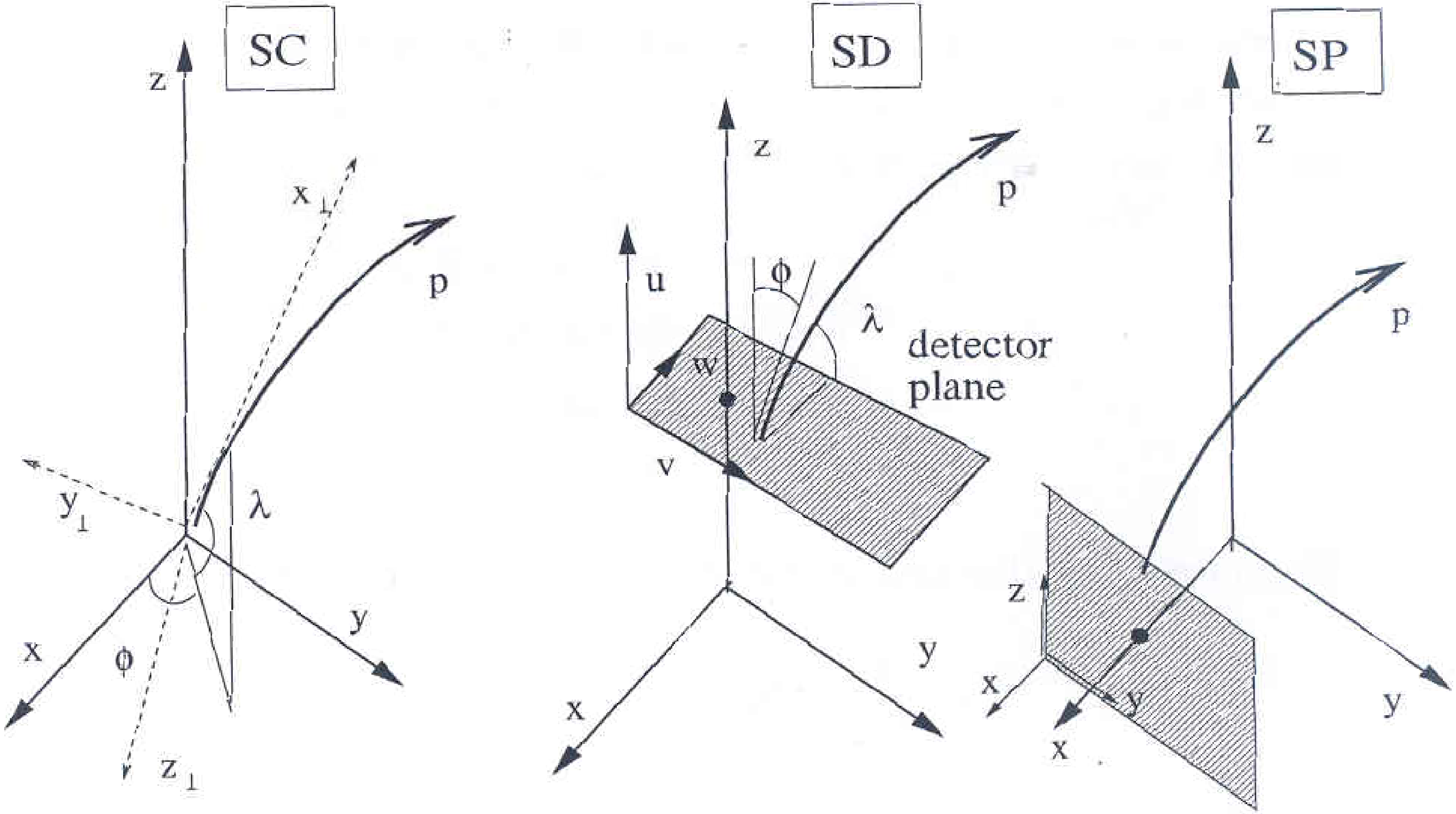


Figure 1.2: Systems of reference for the track following

# Old GEANE Input and Output

**Three** different **requests** can be made for a given representation.
Only two functions to be called (`Eufil(p/l/v)+Ertrak`):

- Extrapolation of the track parameters to a given plane (repr. III)

  Function `Eufilp(N,Ein,Pli,Plf)`

  ```
  N: number of predictions (<=10)
  Ein: initial error matrix
  Pli: initial plane
  Plf: final plane
  ```

- Extrapolation of the track parameters at a given track length (repr. I)

  Function `Eufill(N,Ein,Xlf)`

  ```
  Xlf: track length
  ```

- Extrapolation of the track parameters entering/exiting a given volume (repr. I)

  Function `Eufilv(N,Ein,Namv,Numv,Iovl)`

  ```
  Namv: volume name
  Numv: volume copy number
  Iovl: entering/exiting flag
  ```

# Old GEANE Input and Output

- Track parameters evaluation:

  Function `Ertrak(x1,p1,x2,p2,ip,chopt)`

```
x1: initial position
p1: initial momentum
x2: final position
p2: final momentum
ip: particle code (à la Geant)
chopt: extrapolation options
       ‘L’: tracking until length reached
       ‘P’: tracking until plane reached
       ‘V’: tracking until volume reached
       ‘O’: no error calculation
       ‘E’: exact error calculation
       ‘B’: backward tracking (energy
            loss added to current energy)
```

The output parameters are stored in the Fortran COMMONs.

# Update of TGeant3

- The class only contains the structures for Geane input/output and the function **ertrak()**.
- Structures for Geane input/output (Geane COMMONs) are set as public so that the user can access them:

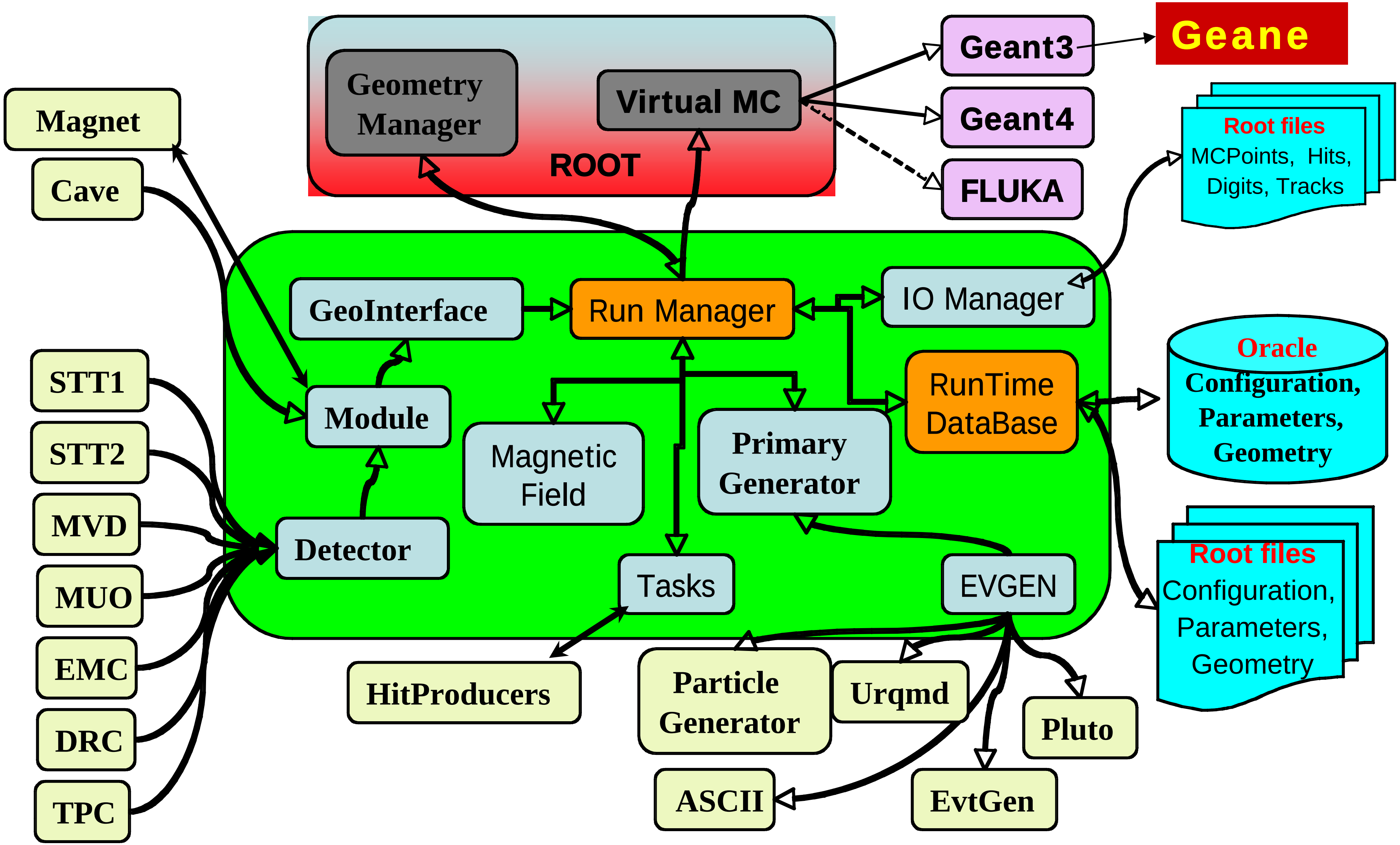
```
Ertrio_t *fErtrio
Eropts_t *fEropts
Eroptc_t *fEroptc
Erwork_t *fErwork
Trcom3_t *fTrcom3
```

- Geane interface functions are added:

```
void eufill(n, ein, xlf);
void eufilp(n,ein,pli,plf);
void eufilv(n,ein,namv,numv,iovl);
void trscsd(pc,rc,pd,rd,h,ch,ierr,spu,dj,dk);
void trsdsc(pd,rd,pc,rc,h,ch,ierr,spu,dj,dk);
void trscsp(ps,rs,pc,rc,h,ch,ierr,spx);
void trspsc(ps,rs,pc,rc,h,ch,ierr,spx);
```

**In this way the GEANE functionality has been reintroduced.**

# PANDA simulation in FairROOT



See talk by Denis Bertini (GSI) at ROOT2007

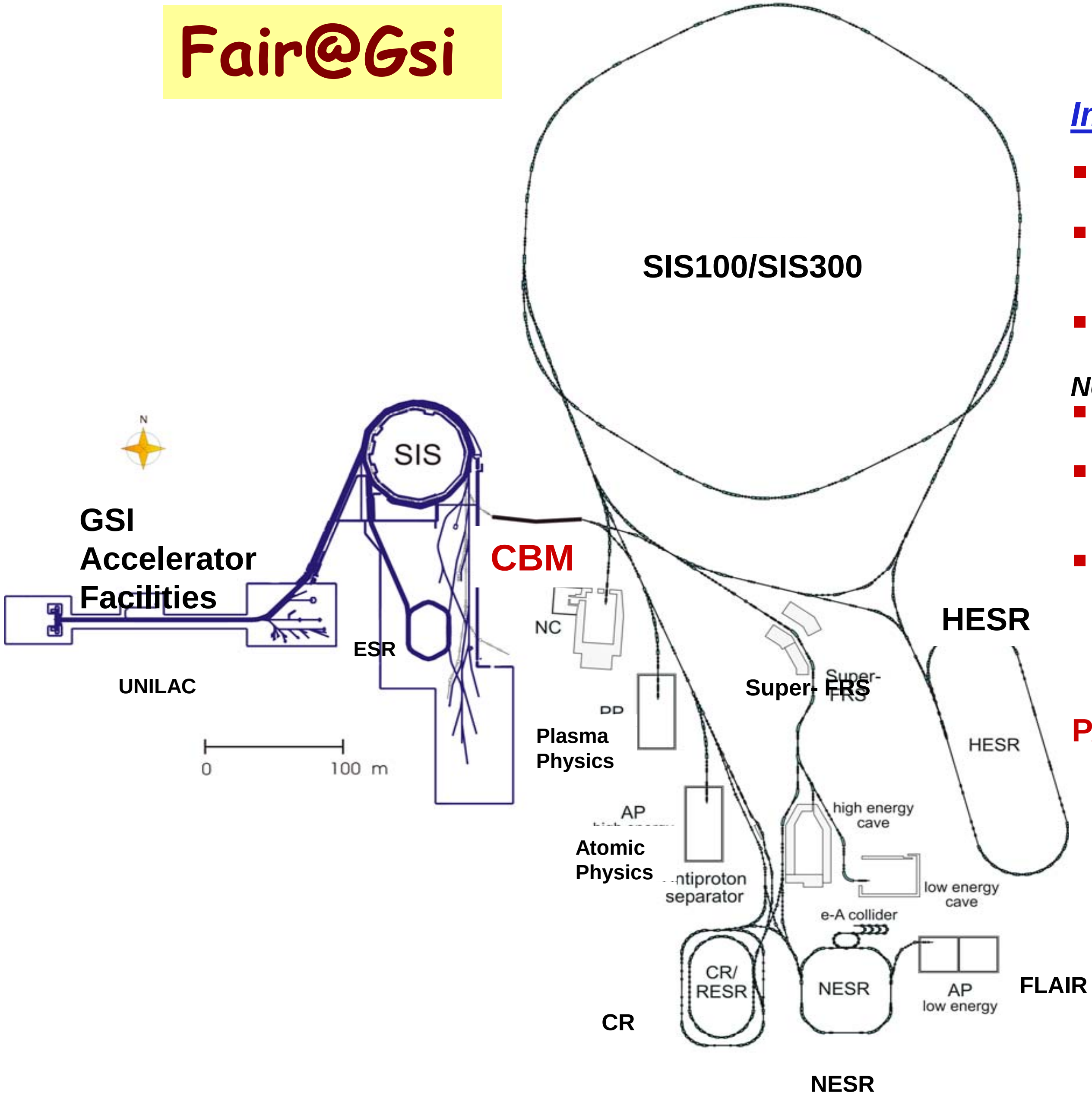
# Integration with FairROOT

- Geane is now fully integrated in PandaROOT as a package: definition of classes **`CbmGeane and CbmGeanePro`**
- Simplified and intuitive user interface in development: only four methods in **`CbmGeanePro`** class

```
Propagate(…);
PropagateToPlane(v0, v1, v2);
PropagateToVolume(VolName, CopyNo , option);
PropagateToLength(length);
```

- Track following can be done from macro or compiled code.
- The exact geometry and field used by the simulation are taken into account **automatically** by the track follower.
- GEANE works as a track follower in the TGeometry modeler and with both Geant3 and Geant4 as MC engines!
- Results are stored as a **tree** in the ROOT file.
- Pull distributions calculation is straighforward using **`TTree::Draw().`**

# Fair@Gsi



## *Improvements*:

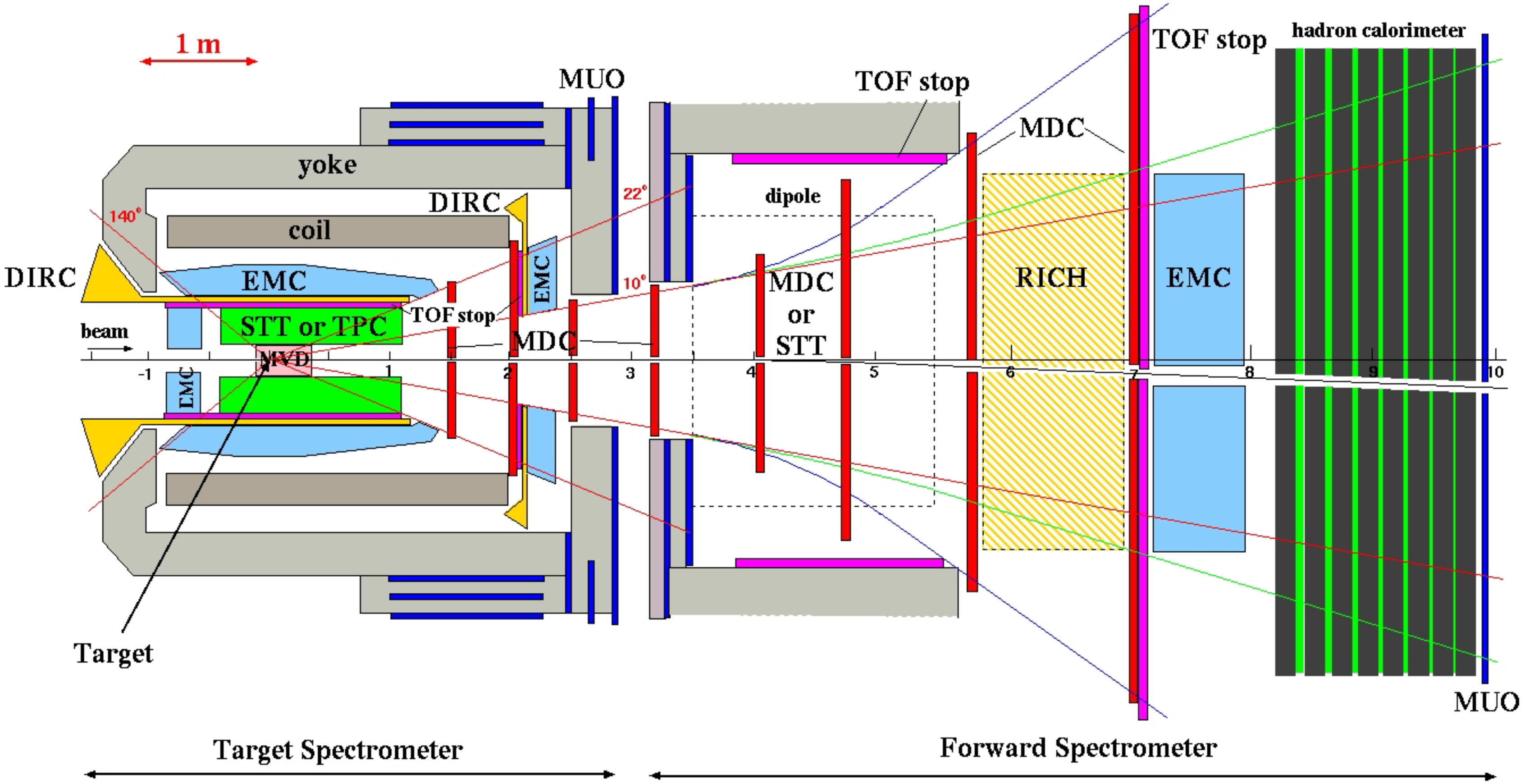
- Primary beam intensity 100 – 1000 x
- Secondary beam intensity
  for radioactive nucl.: up to 10 000x
- Ion energy: 20 x

*New features:*

- *fast* pulsed *superconducting* Magnet
- „cooled“ Antiproton beam
  up to 15 GeV
- Specific:
  intense „cooled“ beam
  radioactive ,exotic nucleus

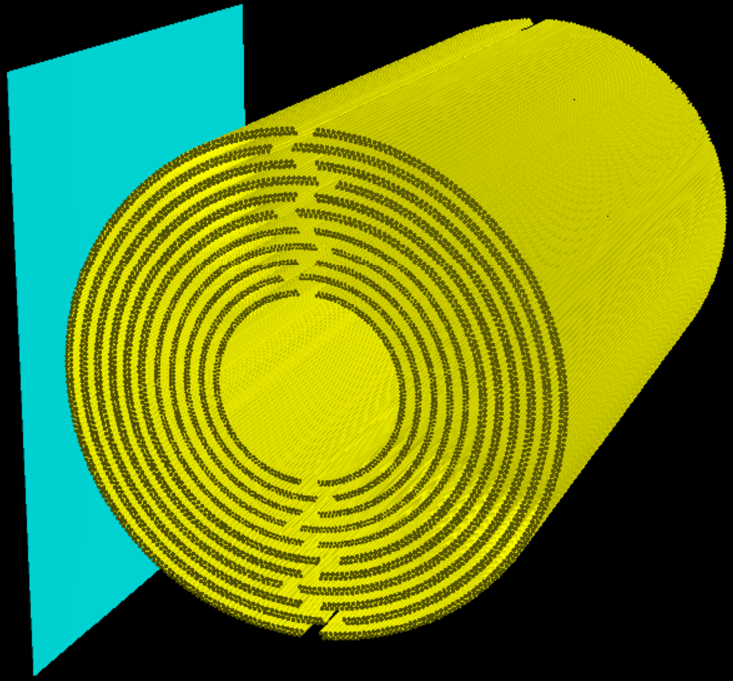
# The PANDA detector

Strong interaction studies with antiprotons.



# Pulls for the STT

We have defined a plane to which we extrapolate the track parameters.



Important issue: how GEANE treat multiple scattering and straggling in energy loss for gaseous materials?

# Multiple scattering

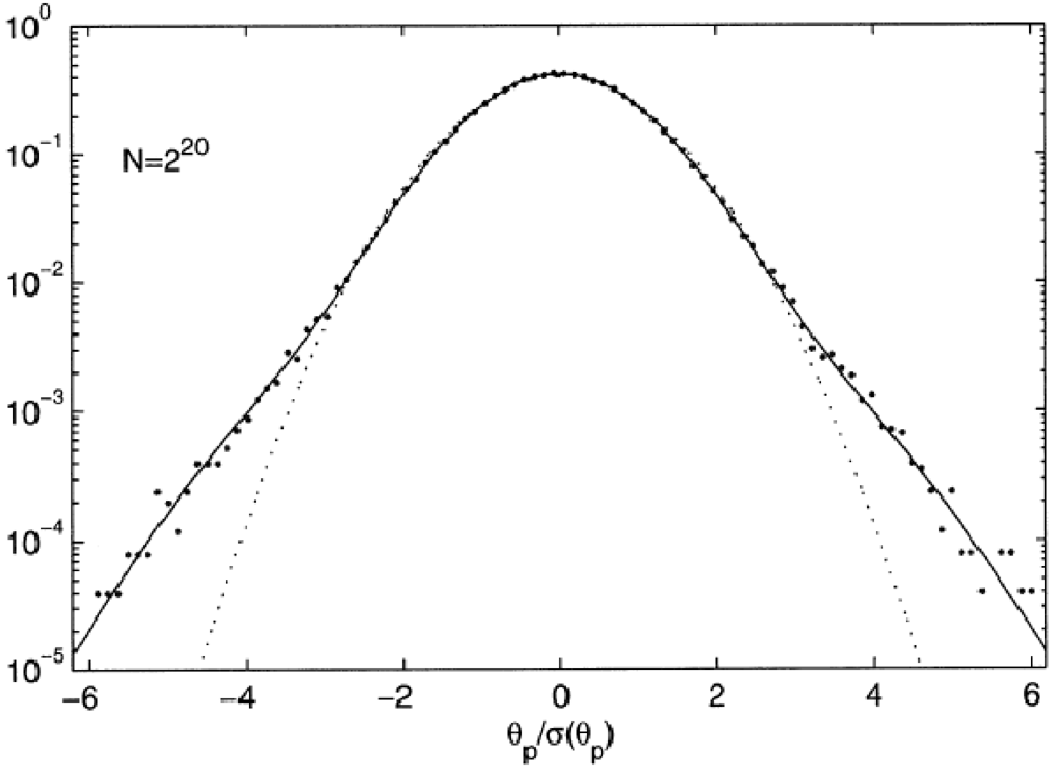
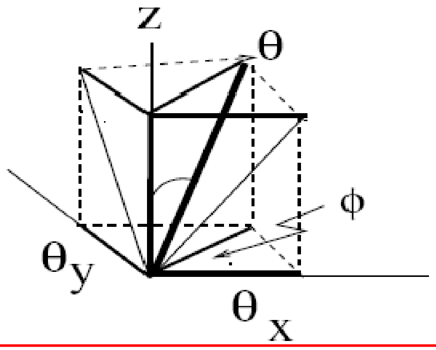


Fig. 3. The density of the projected multiple scattering angle in carbon, in standard measure, for $N = 2^{10}$ (top) and $N = 2^{20}$ (bottom). The dots are the frequencies of a simulated sample obtained by summing over single scatters. The dotted line is the density of a standard Gaussian.

# Multiple scattering



Changes to the ermcsc.f routine

$$p(x,\theta_p;d) = \frac{2\sqrt{3}}{\pi}\frac{1}{<\theta_p^2> d^2}\exp\left[-\frac{4}{<\theta_p^2>}\left(\frac{\theta_p^2}{d} - \frac{3x\theta_p}{d^2} + \frac{3x^2}{d^3}\right)\right]$$

$$<\theta_p^2> = \frac{(0.0136)^2\ d}{p^2\,\beta^2 X_0}\left[1 + 0.038\ \ln\left(\frac{d}{X_0}\right)\right]^2$$

**PDG: wrong**

$$<\theta_p^2> = \frac{184.96\cdot 10^{-6}}{p^2}\frac{d}{\beta^2 X_0}\ .$$

**GEANE: obsolete**

$$<\theta_p^2> = \frac{225\cdot 10^{-6}}{p^2}\frac{d}{\beta^2 X_s}\ , \qquad X_s = X_0\,\frac{Z+1}{Z}\,\frac{\ln(287\,Z^{-1/2})}{\ln(159\,Z^{-1/3})}$$

R. Frühwirth and M. Regler, Nucl. Instr. and Meth. A456(2001)369

# Modification of GEANE for PANDA

$\sigma(\lambda)$
pull

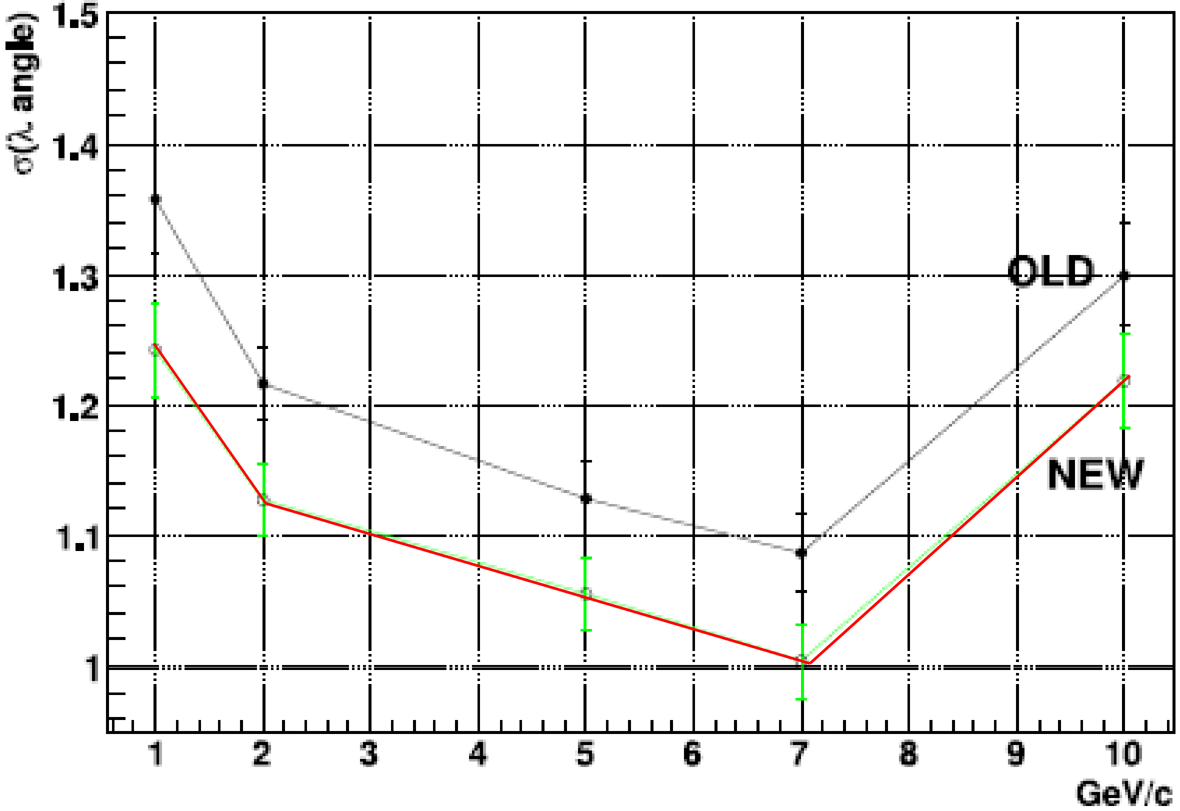


Figure 9: Behaviour with energy of the $\sigma$ of the pull distribution of the dip angle $\lambda$. OLD with eq. (47), NEW with eq. (45).

# Straggling of energy loss

The fluctuations in ionization for one particle of charge $z$, mass $m$, velocity $\beta$, are characterized by the parameter $\kappa$,

$$\kappa = \frac{\xi}{E_{\max}} \ , \tag{60}$$

which is proportional to the ratio of mean energy loss to the maximum allowed energy transfer $E_{\max}$ in a single collision with an atomic electron:

$$E_{\max} = \frac{2m_e\beta^2\gamma^2}{1 + 2\gamma m_e/m + (m_e/m)^2} \ , \tag{61}$$

where $\gamma = 1/\sqrt{1-\beta^2} = E/m$ and $m_e$ is the electron mass. The parameter $\xi$ comes from the Rutherford scattering cross section and is defined as [11]:

$$\xi = 153.4 \frac{z^2 Z}{\beta^2 A} \rho d \quad (\mathrm{keV}) \ , \tag{62}$$

where $\rho$, $d$, $Z$ and $A$ are the density (g/cm$^3$), thickness, atomic and mass number of the medium.

# Straggling of energy loss

1. for heavy absorbers, $\kappa > 10$ and the distribution is gaussian;

$$\sigma^2_{11}(l) = \frac{E^2}{p^6}\,\sigma^2(E) = \frac{E^2}{p^6}\,\xi\, E_{\max}(1-\beta^2/2)$$

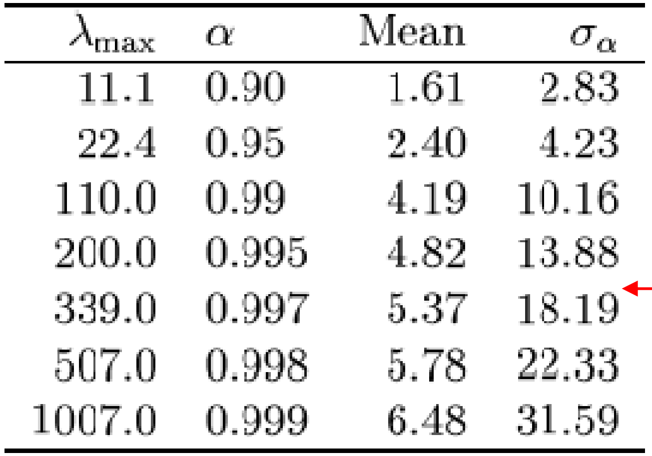
2. for moderate absorbers, $0.01 < \kappa < 10$ and the distribution follows the function of Vavilov [15], that tends smoothly to the gaussian by increasing the thickness;

3. when $\kappa < 0.01$, we are in the presence of thin absorbers. When the number of collisions $N_c > 50$, the distribution follows the Landau function [15];

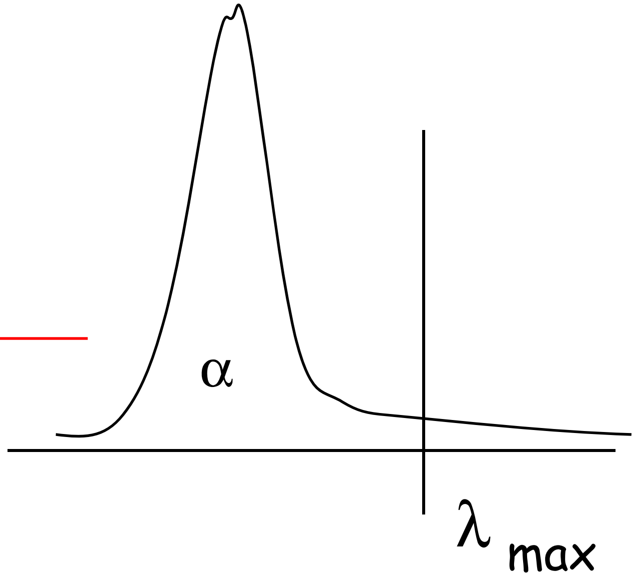
4. for very thin absorbers, $N_c < 50$ (the condition $\kappa \ll 0.01$ is implicitly fulfilled) and there are no universal straggling functions, but only approximated models [14]. We will call this as the *sub-Landau* condition or regime; it is the dominant one in gaseous detectors at PANDA energies.

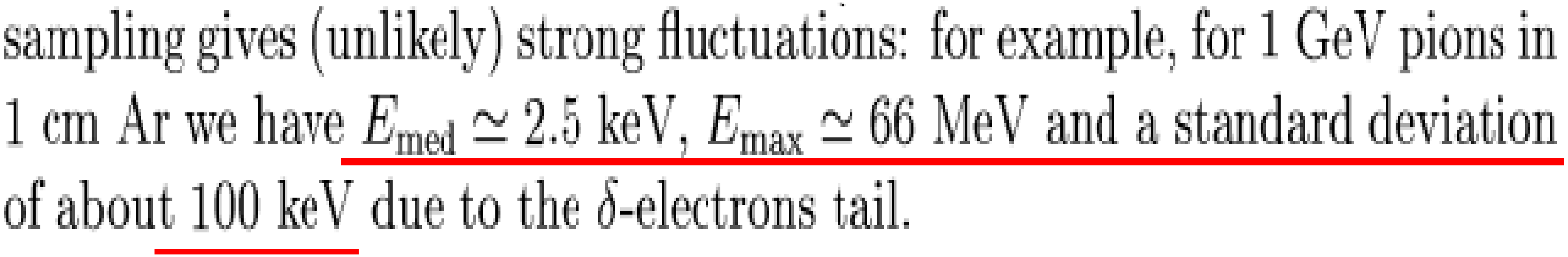
# Energy loss fluctuactions: there is no rigorous solution for track following

Landau has infinite variance!!



| $\lambda_{\max}$ | $\alpha$ | Mean | $\sigma_\alpha$ |
|---|---|---|---|
| 11.1 | 0.90 | 1.61 | 2.83 |
| 22.4 | 0.95 | 2.40 | 4.23 |
| 110.0 | 0.99 | 4.19 | 10.16 |
| 200.0 | 0.995 | 4.82 | 13.88 |
| 339.0 | 0.997 | 5.37 | 18.19 |
| 507.0 | 0.998 | 5.78 | 22.33 |
| 1007.0 | 0.999 | 6.48 | 31.59 |





sampling gives (unlikely) strong fluctuations: for example, for 1 GeV pions in 1 cm Ar we have $E_{\text{med}} \simeq 2.5$ keV, $E_{\max} \simeq 66$ MeV and a standard deviation of about 100 keV due to the $\delta$-electrons tail.

# GEANE modification for PANDA:

Changes to the ertrch.f and erland.f routines

a. for big and moderate absorbers when $\kappa > 0.005$, the variance $\sigma^2(E)$ is given by eq. (63) (old GEANE method);

$$\sigma^2(E) = \frac{\xi^2}{\kappa}(1 - \beta^2/2) = \xi\, E_{\max}(1 - \beta^2/2)\ .$$

b. for thin absorbers, $\kappa < 0.005$, when the number of collisions from eq. (67) is $N_c > 50$, $\sigma^2(E)$ is given by eq. (66) with $\alpha = 0.995$ and $\sigma_\alpha = 13.88$;

$$\sigma(E) = \xi\, \sigma_\alpha$$

**Truncated Landau**

New parameter alpha

c. for very thin absorbers, when $\kappa < 0.005$ and $N_c < 50$, the variance $\sigma_\alpha^2(E)$ is given by eq. (71). The default value for the area considered is $\alpha = 0.995$.

**Urban model**

$$\sigma^2(E) = N_1 e_1^2 + N_2 e_2^2 + N_3 \langle E_3 \rangle^2 + N_3 \sigma_\alpha^2(E_3)(N_3 + 1)$$

**GEANE old**

**GEANE new for PANDA**

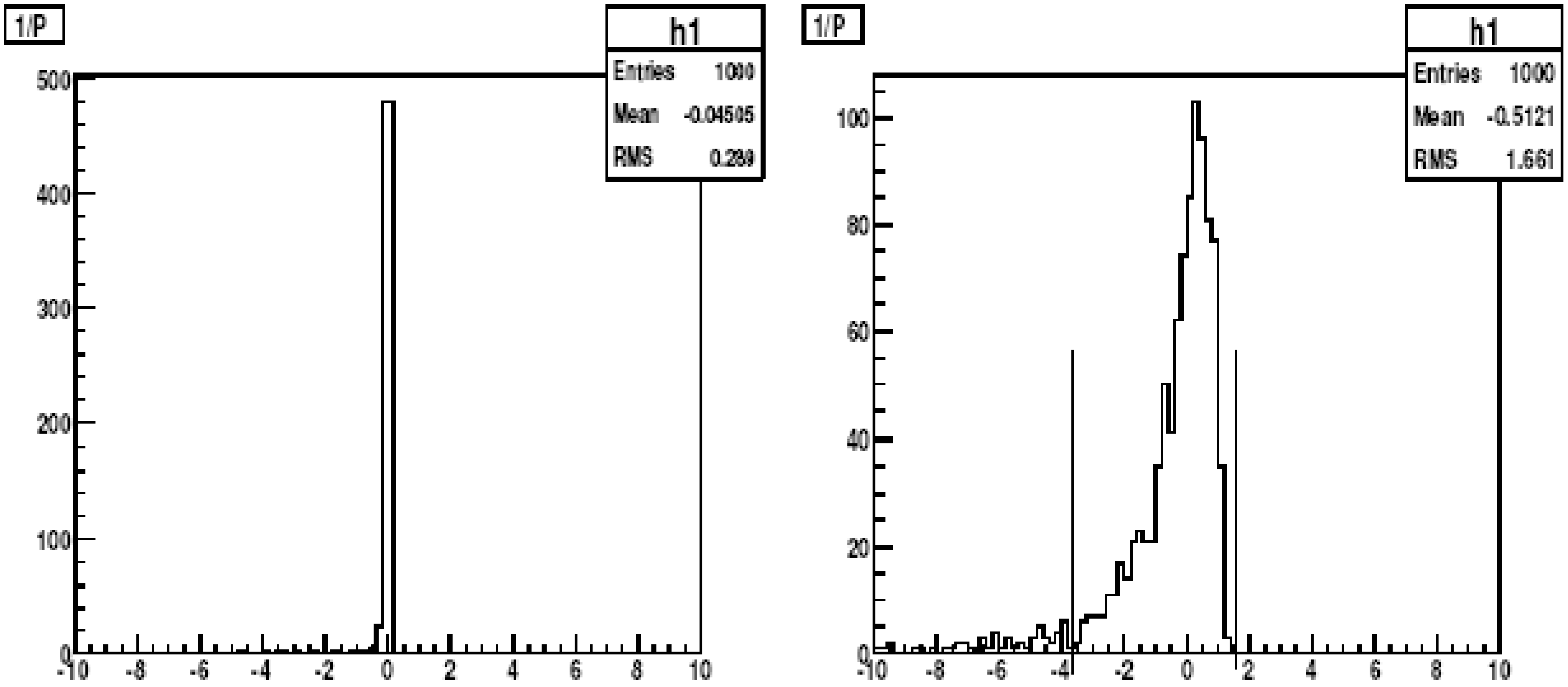


Figure 10: Pull distribution $\Delta(1/p)/\sigma$ for 1 GeV muons after passing through the PANDA straw tube detector. Left: Standard GEANE result (RMS$\simeq$ 0.3 in the displayed window); right: result after the modification with $\alpha = 0.995$ (see the text). The region between the vertical lines has RMS= 1.03.

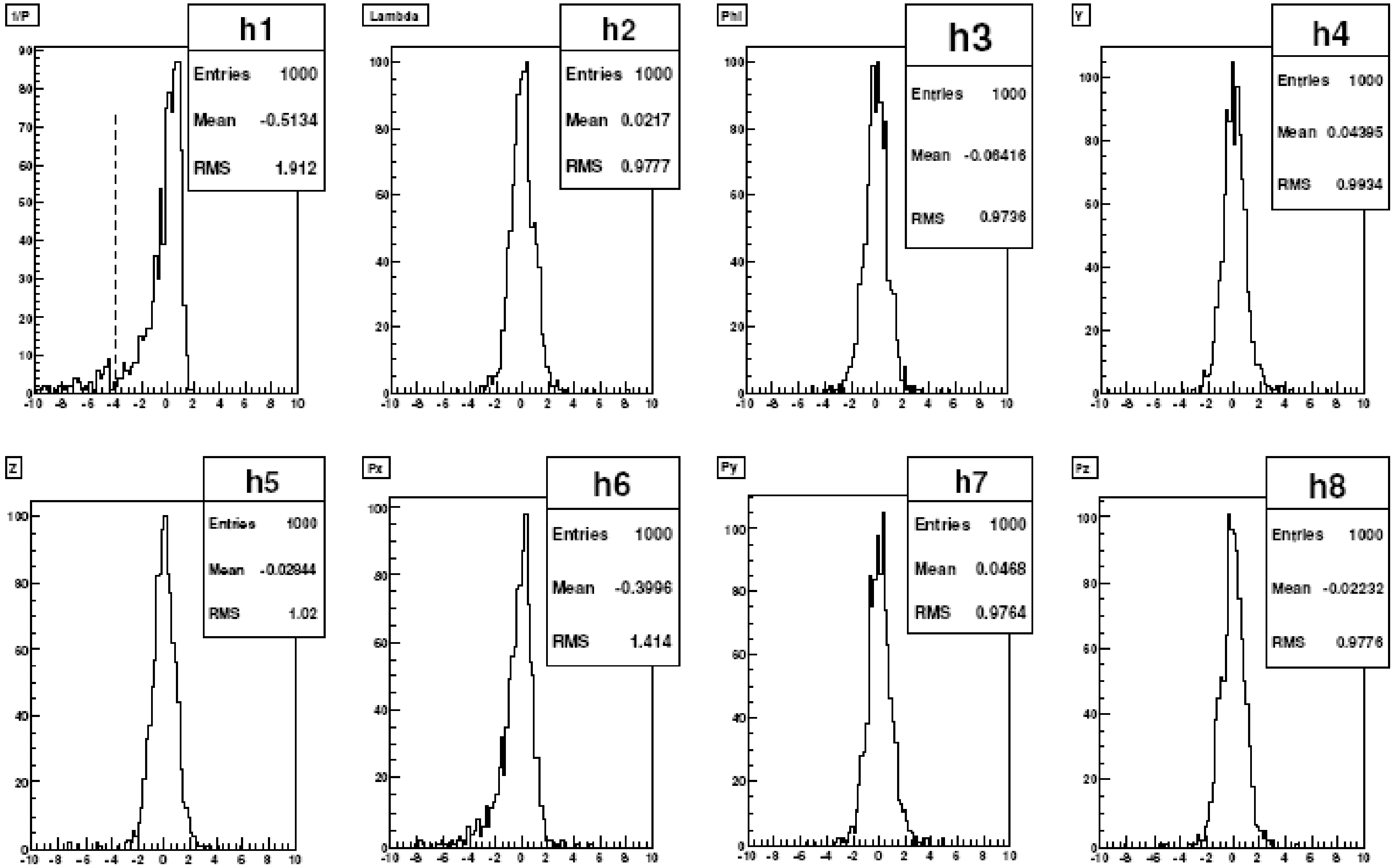
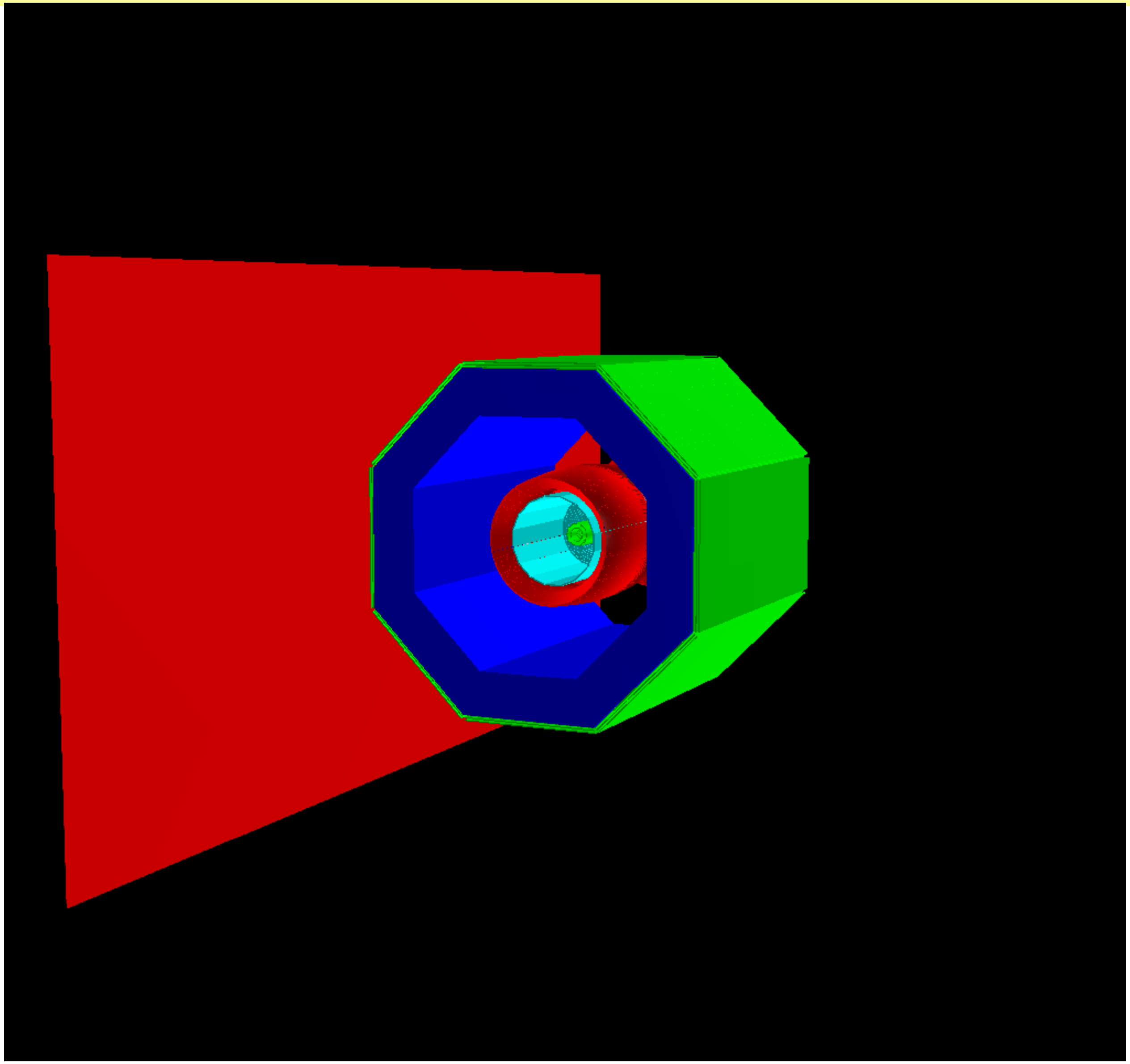


Figure 12: Pull distributions of the 5 track parameters and of $p_x$, $p_y$, $p_z$ in the case of 2 GeV muons after passing through the 22 layers of the straw tube detector only. For the $1/p$ histogram dispersion up to the dotted line one has RMS= 1.08 (top left).

# Pulls for the whole Panda detector



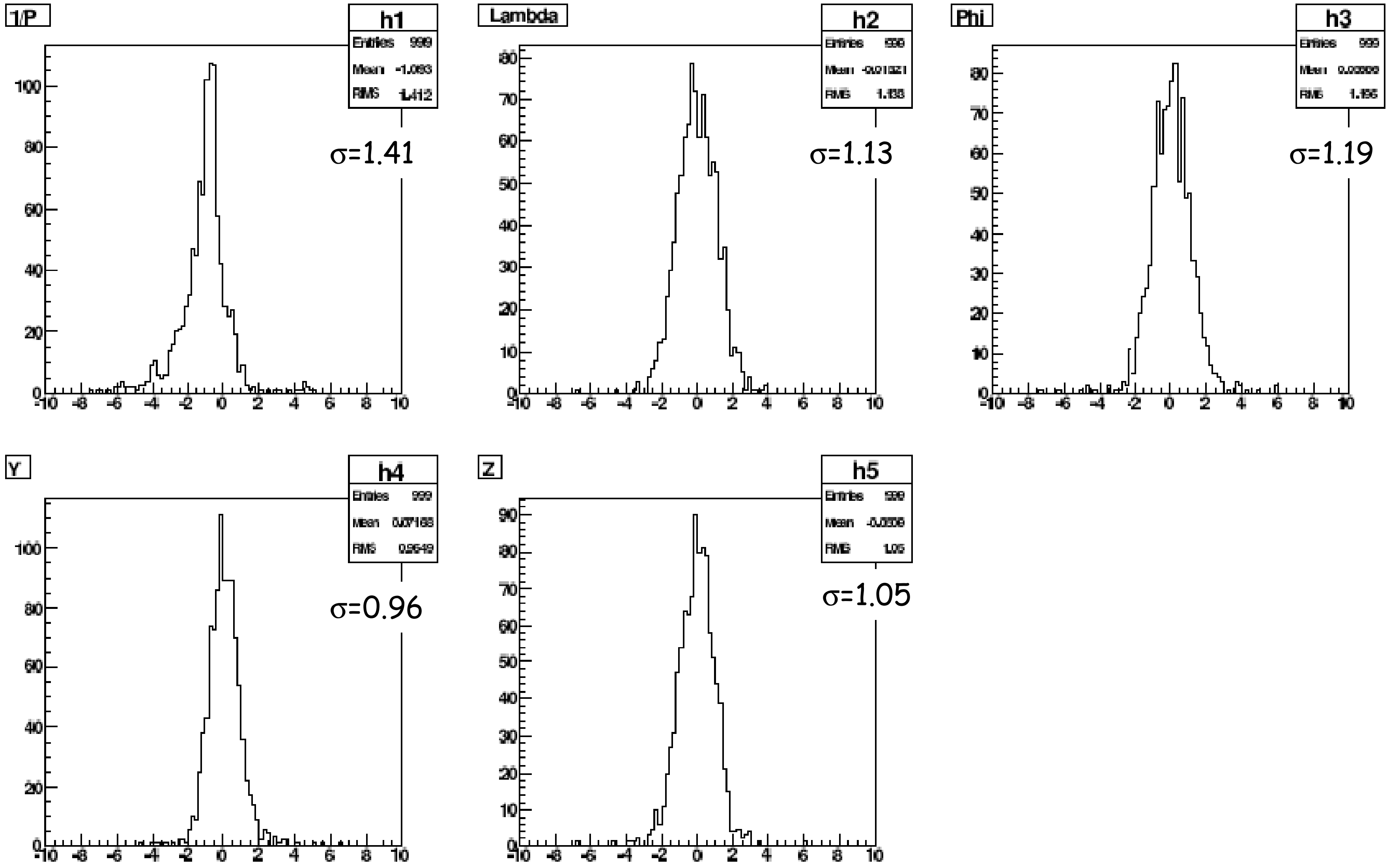


Figure 11: Pull distributions of the 5 track parameters in the case of 2 GeV muons that have passed through the whole detector, just before the PANDA

# New option for extrapolation

## Changes to the eustep.f routine

Extrapolation along the track to the point of closest approach to an experimental hit: case of non-planar detectors (*Virtual Detector Plane*). Useful for boh Panda STT and TPC. Problem of GEANE stepping.

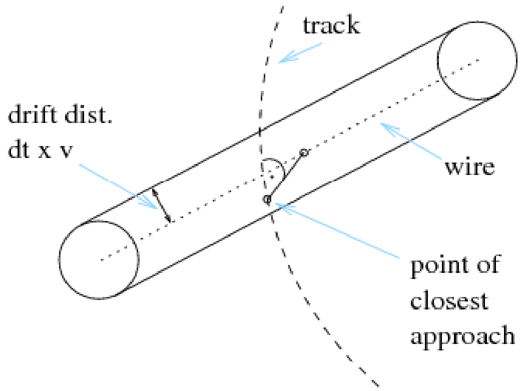


Figure 1: Point of closest approach of track and straw tube wire. The measurement of the straw tube defines that the particle traversed somewhere on the drift cylinder.

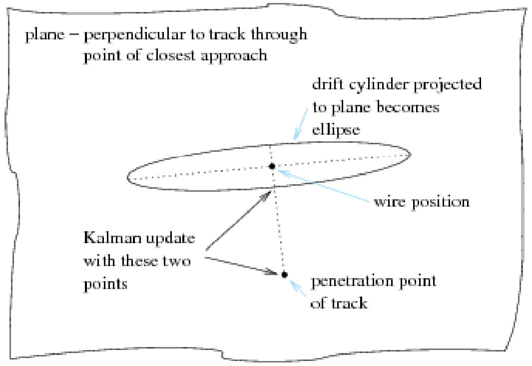
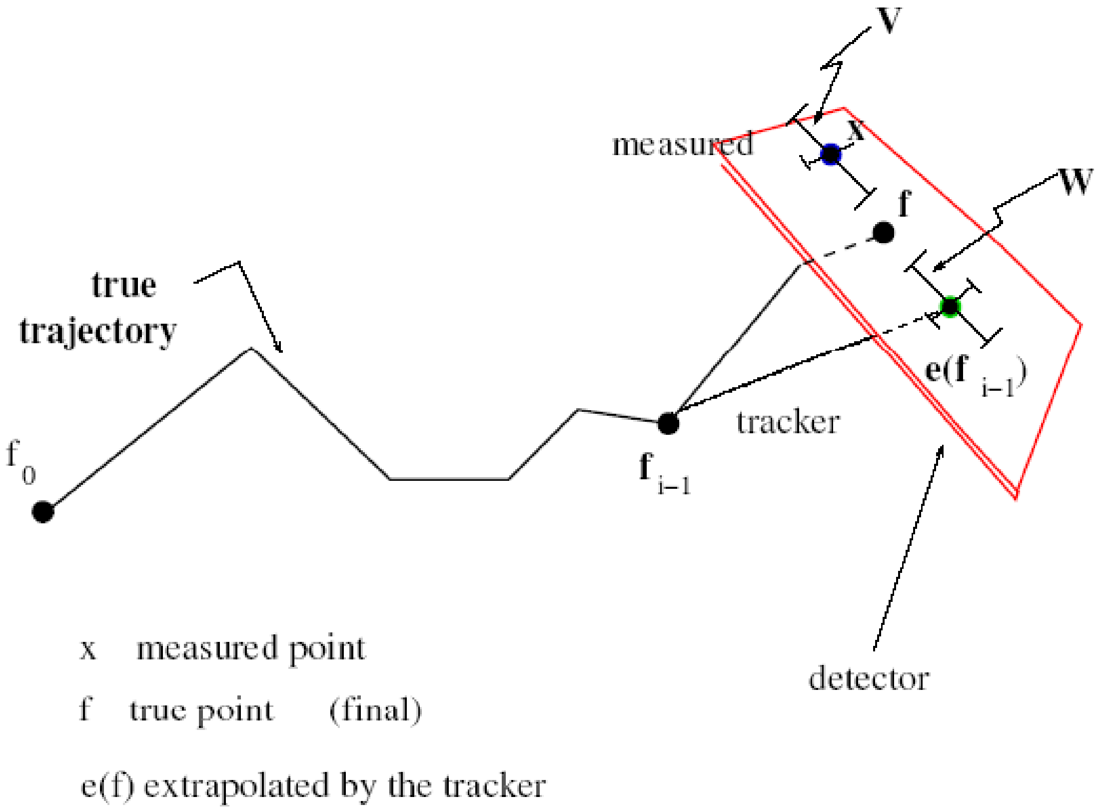


Figure 2: Virtual detector plane. The intersection of the plane and the drift cylinder defines the ellipse seen in the figure. The point on the ellipse closer to the track point is the position measurement used for the update.

# Tracking with Kalman: application of GEANE.



The best estimate of the track is given by minimizing w.r.t the $f$ variables:

$$\chi^2(f) = \sum_i [(e_i[f_{i-1}]-f_i)\boldsymbol{W}_{i-1}(e_i[f_{i-1}]-f_i)]+(x_i-f_i)\boldsymbol{V}_i(x_i-f_i) \tag{1}$$

Note the $\boldsymbol{W}$ matrix associated to $e_i$ because the extrapolation start from the true point.

# Tracking with Kalman

The minimization gives:

$$\frac{\partial\chi^2}{\partial f_i} = \boldsymbol{W}_{i-1,i}(e_i[f_{i-1}] - f_i) + \boldsymbol{V}(x_i - f_i) \qquad (2)$$
$$+\boldsymbol{T}_{i,i+1}\boldsymbol{W}_{i,i+1}(e_{i+1}[f_i] - f_{i+1}) = 0$$

where the last (*extra*) term comes from the extrapolation procedure (tracker).

The best way to solve eq (2) is the Kalman algorithm (Kalman, 1961). It is based on three steps:

- **EXTRAPOLATION**: calculation of $e_i$ and $\boldsymbol{W}$. Deterministic step made by the tracker.

$$
\begin{aligned}
e_i &= \boldsymbol{G}_{i-1,i}[k_{i-1}] && (3)\\
\boldsymbol{\sigma}^2[e_i] &= \boldsymbol{T}_{i-1,i}\,\boldsymbol{\sigma}^2[k_{i-1}]\,\boldsymbol{T}^T_{i-1,i} + \boldsymbol{W}^{-1}_{i-1,i} && (4)
\end{aligned}
$$

Square brackets mean function argument

$e_i$ = EXTRAP. extrapolation

$k_i$ = result of the Kalman filter

$\boldsymbol{T}_{i-1,i}$ = EXTRAP. transport matrix

$\boldsymbol{\sigma}(k)^2$ = Kalman error matrix

$\boldsymbol{\sigma}^2[e_i]$ = EXTRAP. error matrix

$\boldsymbol{W}_{i-1,i}$ = EXTRAP. energy loss and multiple scattering weight matrix

$\boldsymbol{W}^{-1}_{i-1,i}$= covariance matrix inverse of $W$

- **FILTERING**: minimizes the first two terms of eq. (2). It is simply the weighted mean;

$$\begin{aligned} k_i &= \boldsymbol{\sigma}^2[k_i]\left(\boldsymbol{\sigma}^{-2}[e_i]\,e_i + \boldsymbol{V}_i x_i\right) && (5) \\ \boldsymbol{\sigma}^{-2}[k_i] &= \boldsymbol{\sigma}^{-2}[e_i] + \boldsymbol{V}_i && (6) \end{aligned}$$

$x_i$ = measured points
$k_i$ = Kalman average value
$\boldsymbol{\sigma}(k)^2$ = Kalman error matrix
$\boldsymbol{\sigma}^2(e)$ = EXTRAP. error matrix
$\boldsymbol{V}$ = original weight matrix of the measured points

- **SMOOTHING**: necessary to minimize a $\chi^2$ in the presence of the extrapolation term (last term in eq. (2)).

$$f_i = k_i + \boldsymbol{A}_i \left( f_{i+1} - e_{i+1} \right) \tag{7}$$

$$\boldsymbol{\sigma}^2[f_i] = \boldsymbol{\sigma}^2[k_i] + \boldsymbol{A}_i \left( \boldsymbol{\sigma}^2[f_{i+1}] - \boldsymbol{\sigma}^2[e_{i+1}] \right) \boldsymbol{A}_i^T \tag{8}$$

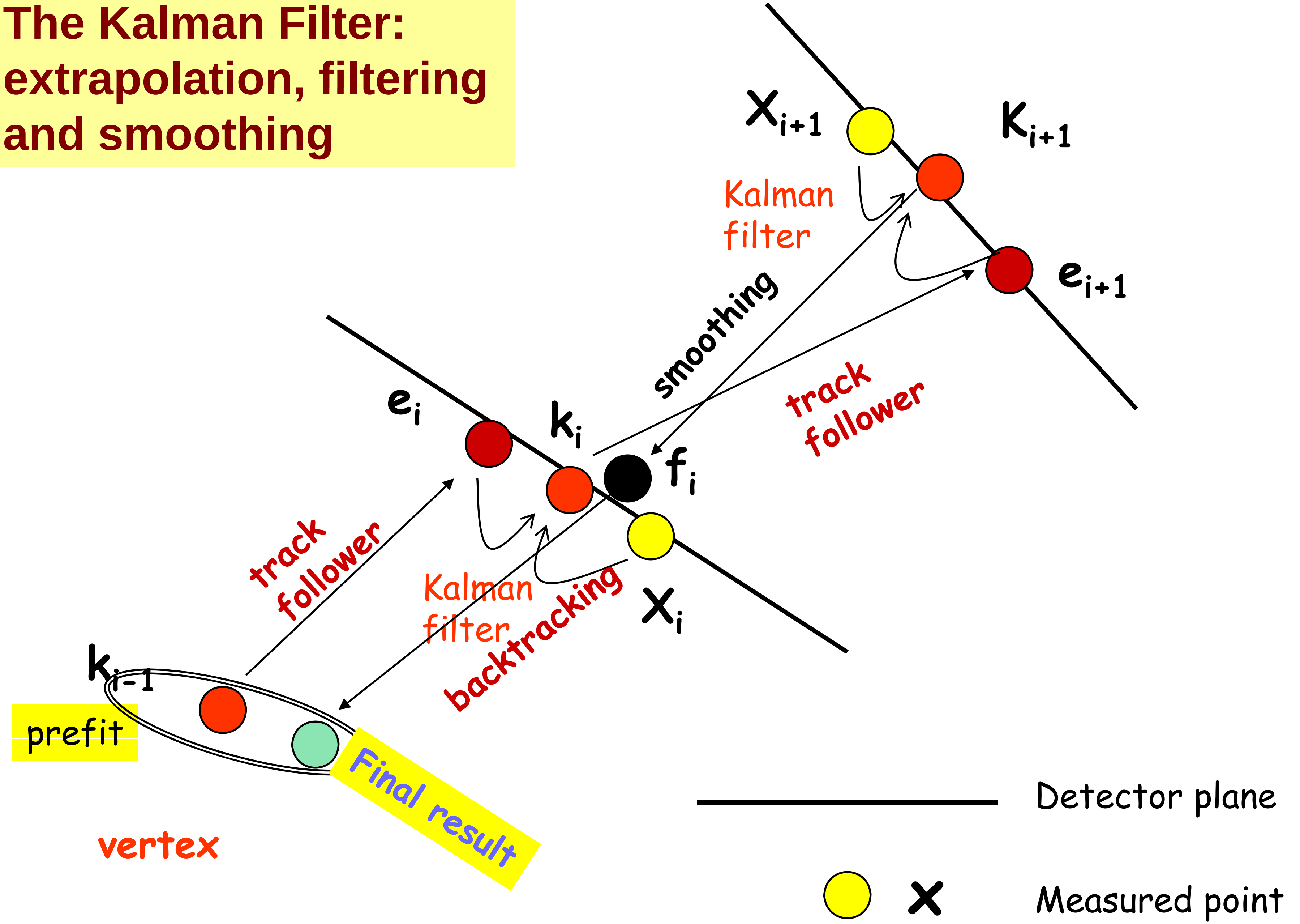
$$\boldsymbol{A}_i = \boldsymbol{\sigma}^2[k_i]\, \boldsymbol{T}_{i,i+1}^T\, \boldsymbol{\sigma}^{-2}[e_{i+1}] \tag{9}$$

$f_i$ = final average value

$\boldsymbol{\sigma}(k)^2$ = Kalman error matrix

$\boldsymbol{\sigma}^2(e)$ = EXTRAP. error matrix

# The Kalman Filter: extrapolation, filtering and smoothing



# Summary

- The **GEANE functionality** has been **reintroduced** in TGeant3.
- GEANE is now fully integrated in FairROOT with a **new C++ interface**: it can be used with the TGeometry modeler and with both G3 and G4!
- **Some modifications** for multiple scattering and energy loss have been introduced in GEANE to allow its use for gaseous detectors: old GEANE by default (no need to change existing code), new features only by explicit request (rule: keep changes to Fortran to the minimum).
- The results are good enough to allow the use of this track following in global fits (**Panda Kalman filter**).
- The energy loss part (errors on 1/p) is not completely under control (further investigations are necessary).
- All the mathematical and physical details are contained in a **technical report** from our group.

# Detailed report from our group available at:
# http://www.pv.infn.it/~fontana/tracking.pdf

## Main References

- V. Innocente, M. Maire and E. Nagy, GEANE: Average Tracking and Error Propagation Package, CERN Program Library W5013-E (1991)
- A. Strandlie, W. Wittek, Propagation of Covariance Matrices of Track Parameters in Homogeneous Magnetic Fields in CMS, CMS Note 2006/001
- V. Innocente and E. Nagy, Trajectory fit in presence of dense materials NIM A324(1993)297
- R. Fruewirth et al., Data Analysis Techniques for High Energy Physics, 2° edition, Cambridge University Press, 2000
- K. Lassilla-Perini and L. Urban, NIM A362(1995)416
- Geant3 and Geant4 manuals