IDEA model in DELPHES

F. Bedeschi,
ECFA Higgs Factories:
1st Topical Meeting on Simulation,
OnLine, February 2022

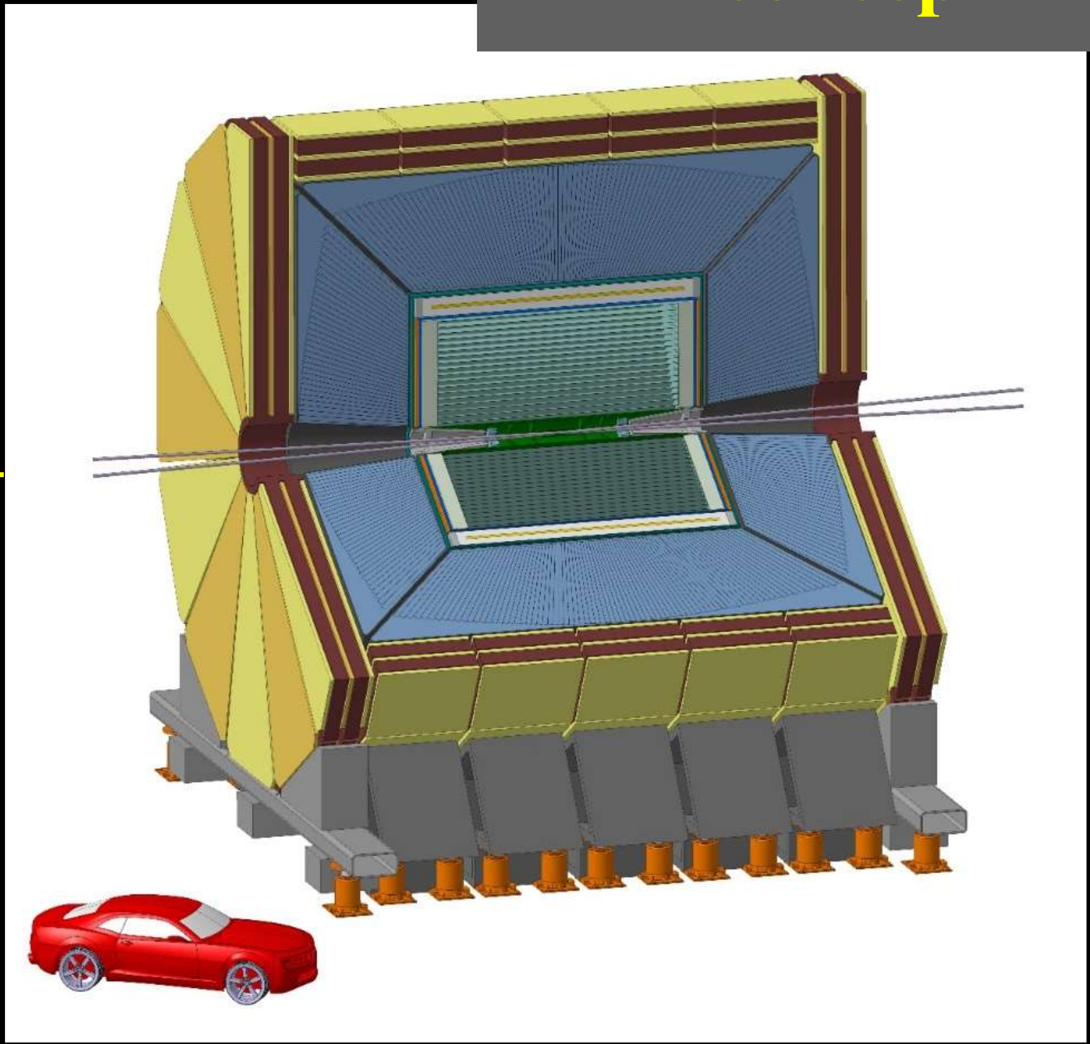
Outline

- IDEA detector
- Tracking
  - Application example
- Vertexing
- PID
- Calorimetry
- Conclusions

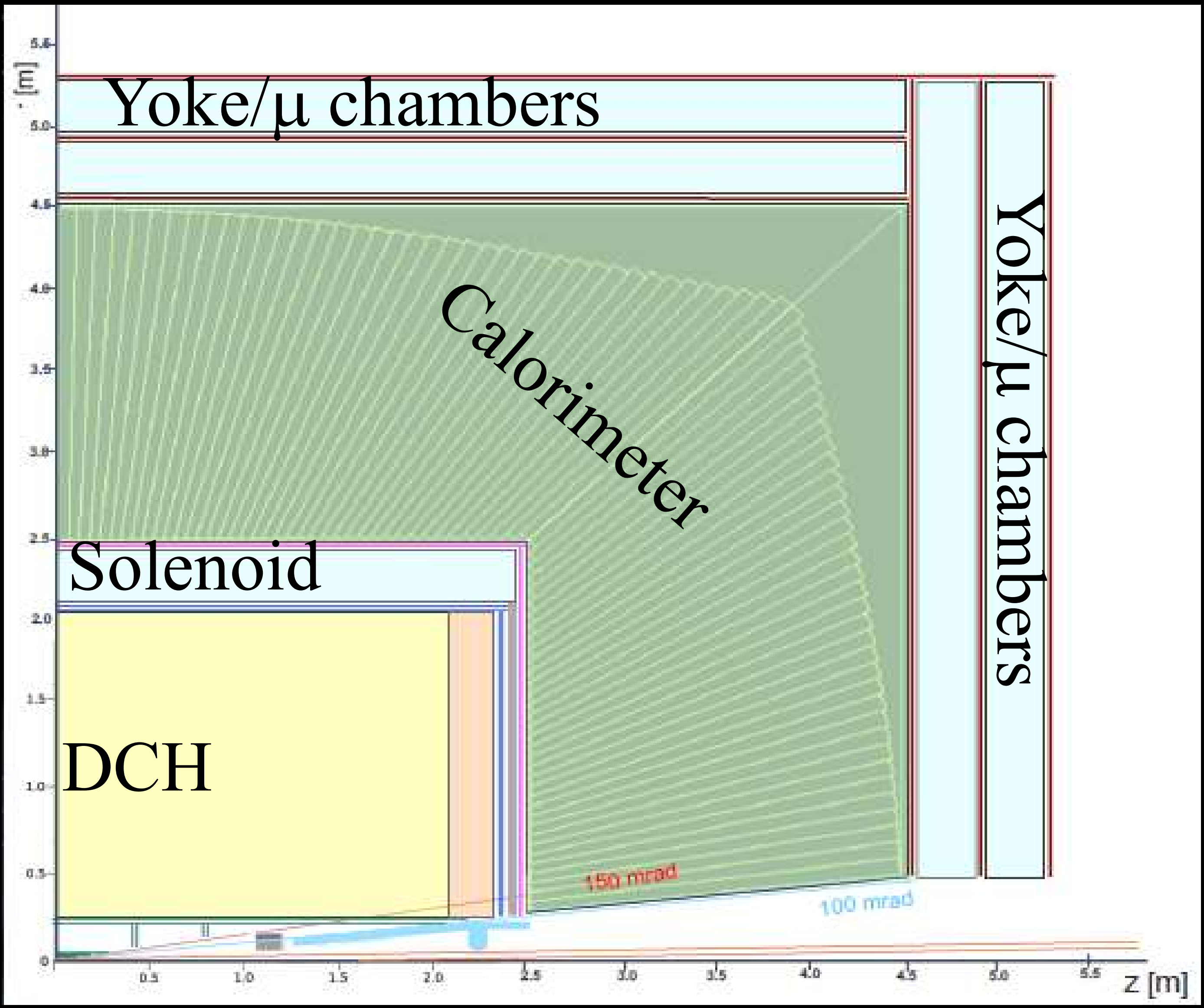
Detector concept IDEA

- Si pixel vertex detector
  - 5 MAPS layers
    - R = 1.7 – 34 cm
- Drift chamber (112 layers)
  - 4m long, r = 35 – 200 cm
- Si wrapper: strips
- Solenoid: 2 T - 5 m, r = 2.1-2.4
  - 0.74 X_0, 0.16 λ @ 90º
- Pre-shower: µRwell
- Dual Readout calorimetry
  - 2m deep/8 λ
- Muon chambers
  - µRwell

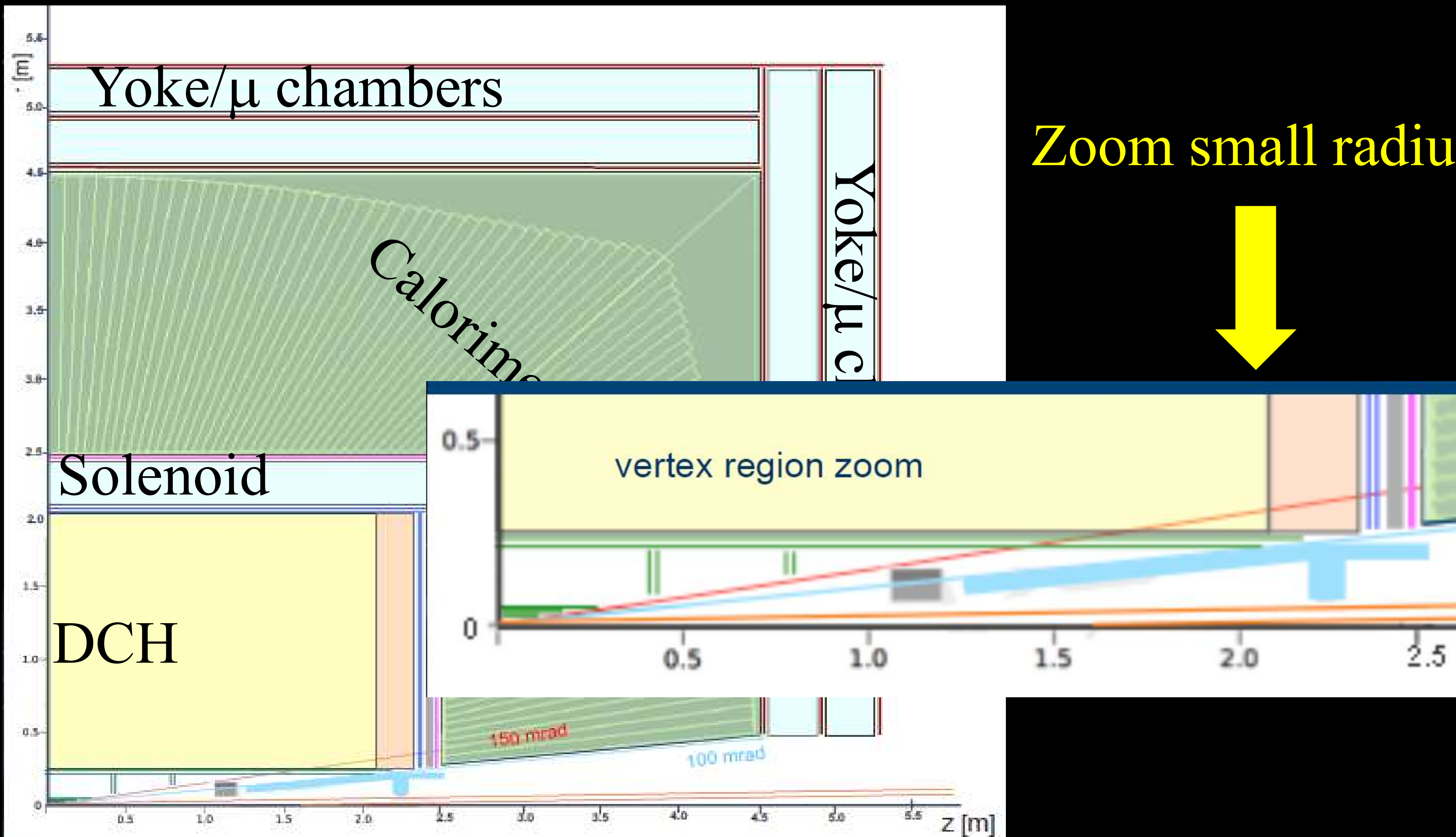
IDEA concept



IDEA details



IDEA details



Fast tracking simulation

- FCC tracking systems still evolving

Fast tracking simulation

- FCC tracking systems still evolving
- Need fast turn around in evaluation of various options
  - Easy implementation of modified geometry
  - Easy change of detector performances

Fast tracking simulation

- FCC tracking systems still evolving
- Need fast turn around in evaluation of various options
  - Easy implementation of modified geometry
  - Easy change of detector performances
- Need realistic input for fast simulation
  - Full covariance matrix
  - Dependence on pt and polar angle for generic configuration

Implementation (1)

- Track fit χ^2 linearized in the fit parameters:

$$\chi^2 = \vec{d}^t S^{-1} \vec{d} \simeq (\vec{d}_0 - \vec{d}^* + \frac{\partial \vec{d}}{\partial \vec{p}} \cdot \Delta\vec{p})^t S^{-1} (\vec{d}_0 - \vec{d}^* + \frac{\partial \vec{d}}{\partial \vec{p}} \cdot \Delta\vec{p})$$

  - d/d* = predicted/measured distance of track from wire or pixel
  - $\vec{p}$ = track parameters
  - S = covariance of all measurements:

$$S_{ij} = \sigma_i^2 \, \delta_{ij} + M_{ij}$$

$$M_{ij} = \sum_{1 \le k < \min(i,j)} (L_i - L_k)(L_j - L_k)\theta_k^2(i,j)$$

Implementation (1)

- Track fit χ^2 linearized in the fit parameters:

$$\chi^2 = \vec{d}^t S^{-1} \vec{d} \simeq (\vec{d}_0 - \vec{d}^* + \frac{\partial \vec{d}}{\partial \vec{p}} \cdot \Delta\vec{p})^t S^{-1} (\vec{d}_0 - \vec{d}^* + \frac{\partial \vec{d}}{\partial \vec{p}} \cdot \Delta\vec{p})$$

  - d/d* = predicted/measured distance of track from wire or pixel
  - $\vec{p}$ = track parameters
  - S = covariance of all measurements:

$$S_{ij} = \sigma_i^2 \, \delta_{ij} + M_{ij}$$

$$M_{ij} = \sum_{1 \le k < \min(i,j)} (L_i - L_k)(L_j - L_k)\theta_k^2(i,j)$$

- Parameter resolution depends only on S and derivatives:

$$C^{-1} = \frac{1}{2} \frac{\partial^2 \chi^2}{\partial \vec{p} \partial \vec{p}} = A^t S^{-1} A, \text{ where } A = \frac{\partial \vec{d}}{\partial \vec{p}}$$

Implementation (2)

- Simple geometry description in text file:

Implementation (2)

- Simple geometry description in text file:
  - Layer types allowed:
    - Cylinder shell (const R) or disk (constant z)
    - Measurement or inert (for MS)
    - Measurement is axial (Rϕ), small angle stereo, 90 deg. (Rz) or pixel

Implementation (2)

- Simple geometry description in text file:
  - Layer types allowed:
    - Cylinder shell (const R) or disk (constant z)
    - Measurement or inert (for MS)
    - Measurement is axial (Rϕ), small angle stereo, 90 deg. (Rz) or pixel
- Currently implemented:
  - IDEA in DELPHES card
  - CLD (tracking only) also available in private repository

Implementation (3)

- Track handling:
  - Initialize with $\vec{x}, \vec{p}$: point on track and momentum vector
  - Find intersection with any given layer (acceptance)
  - Calculates helix parameter covariance matrix
    - Include multiple scattering contributions with correlations

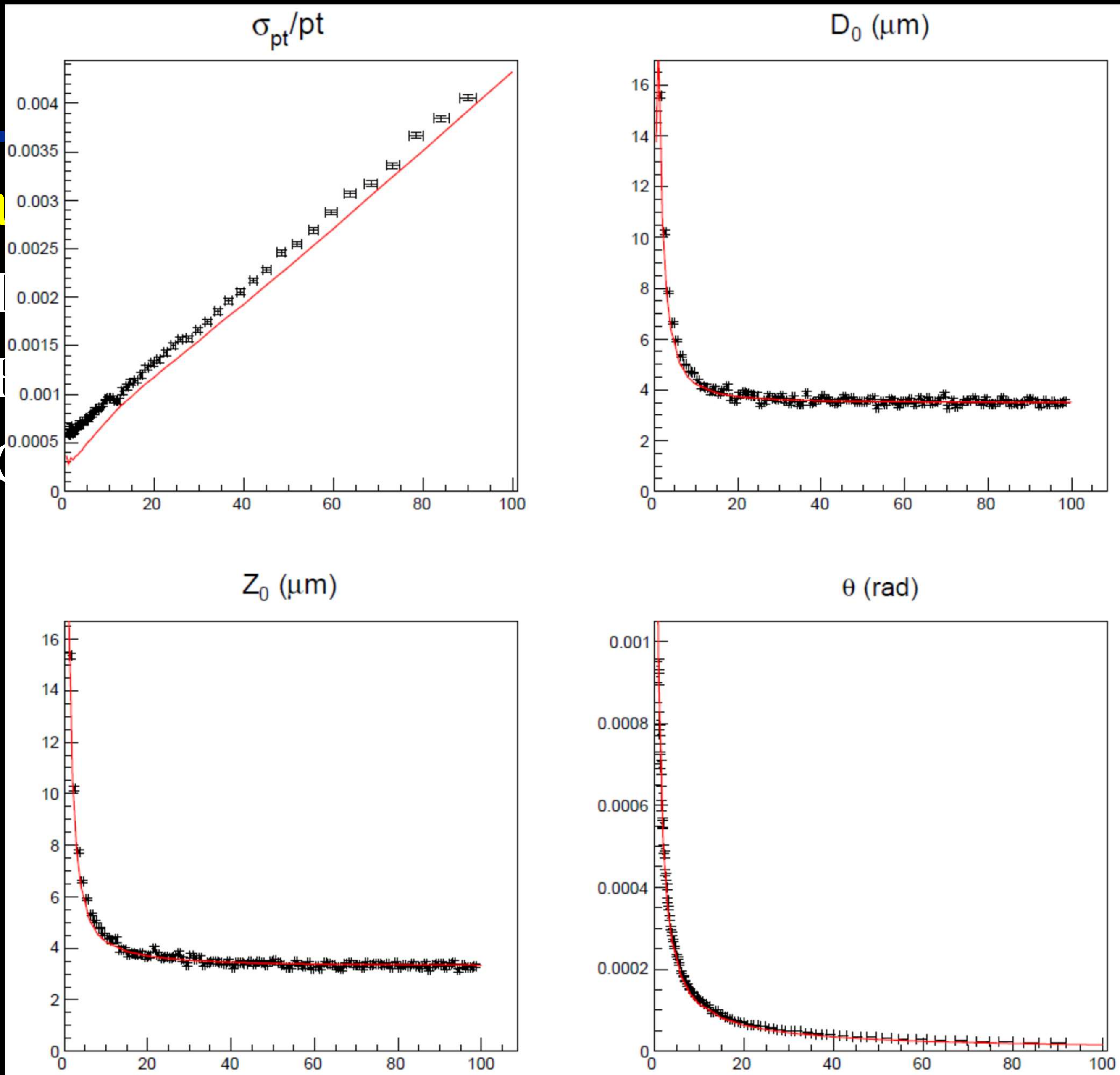
❖ Tra

➢ I ... um vector

➢ F ... ce)

➢ C

... ions



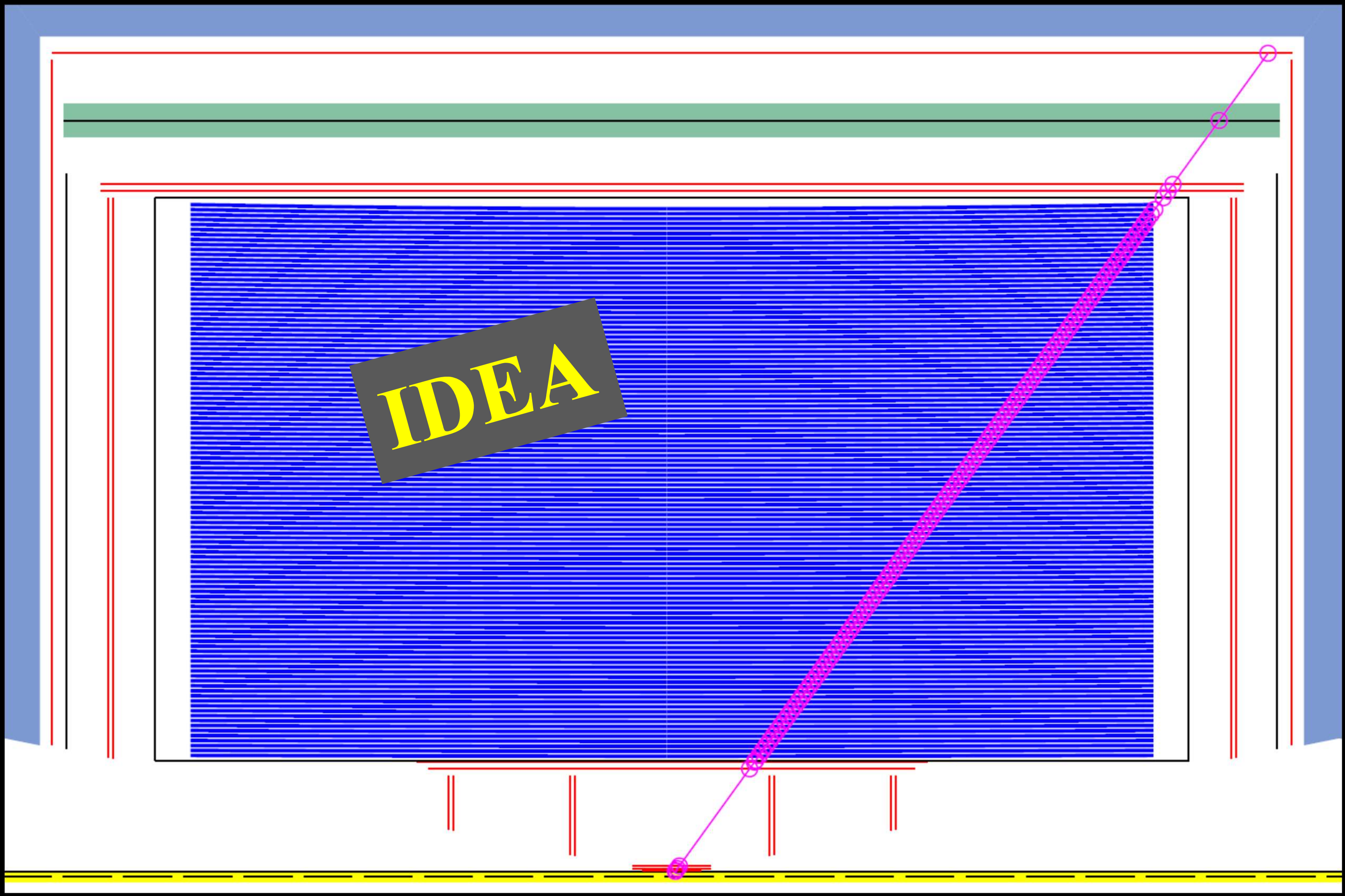
❖ Covariance calculation x-checked with full simulation

Examples (1)

- Implemented geometry and material summary

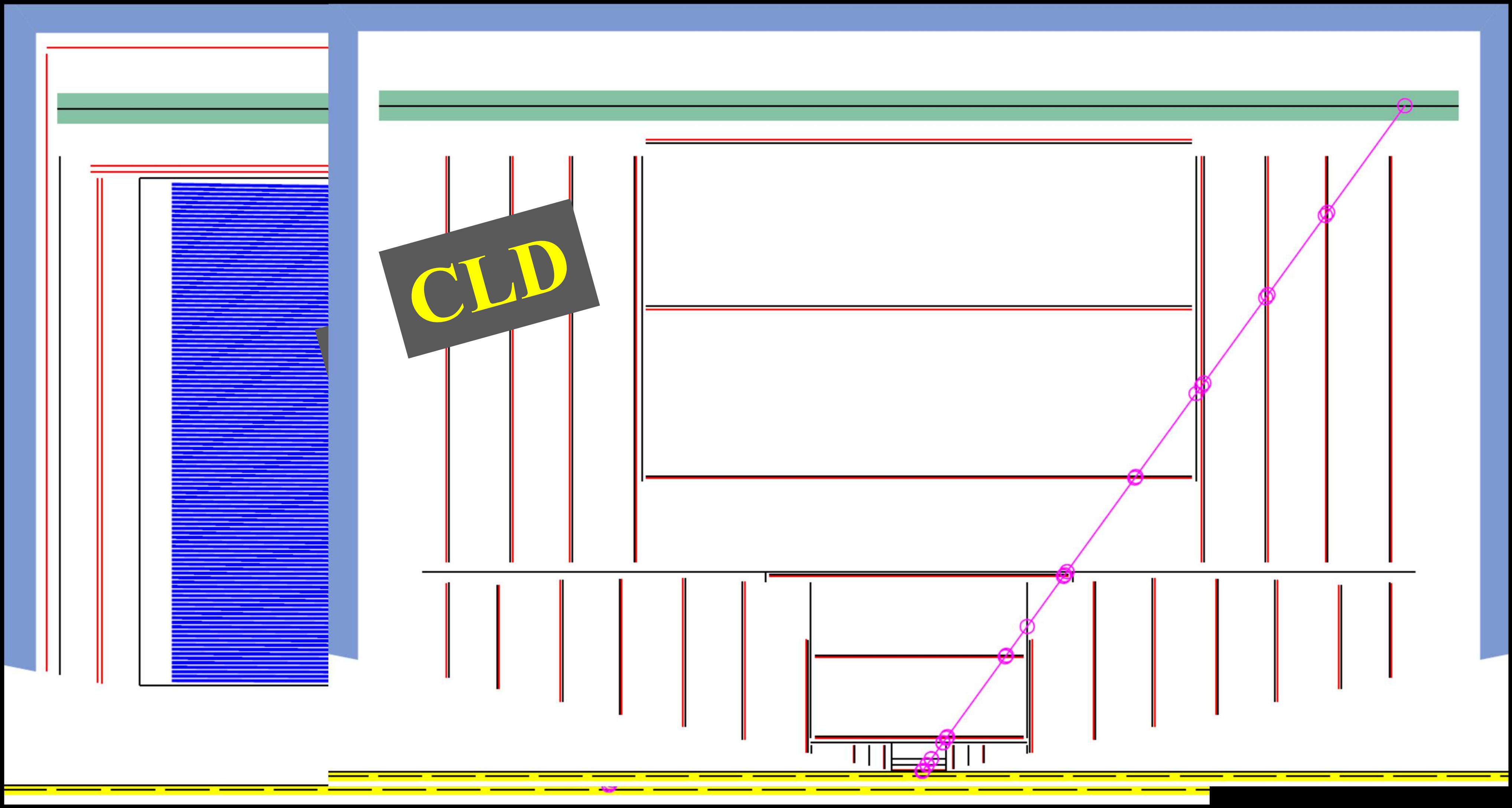
Examples (1)

❖ Implemented geometry and material summary



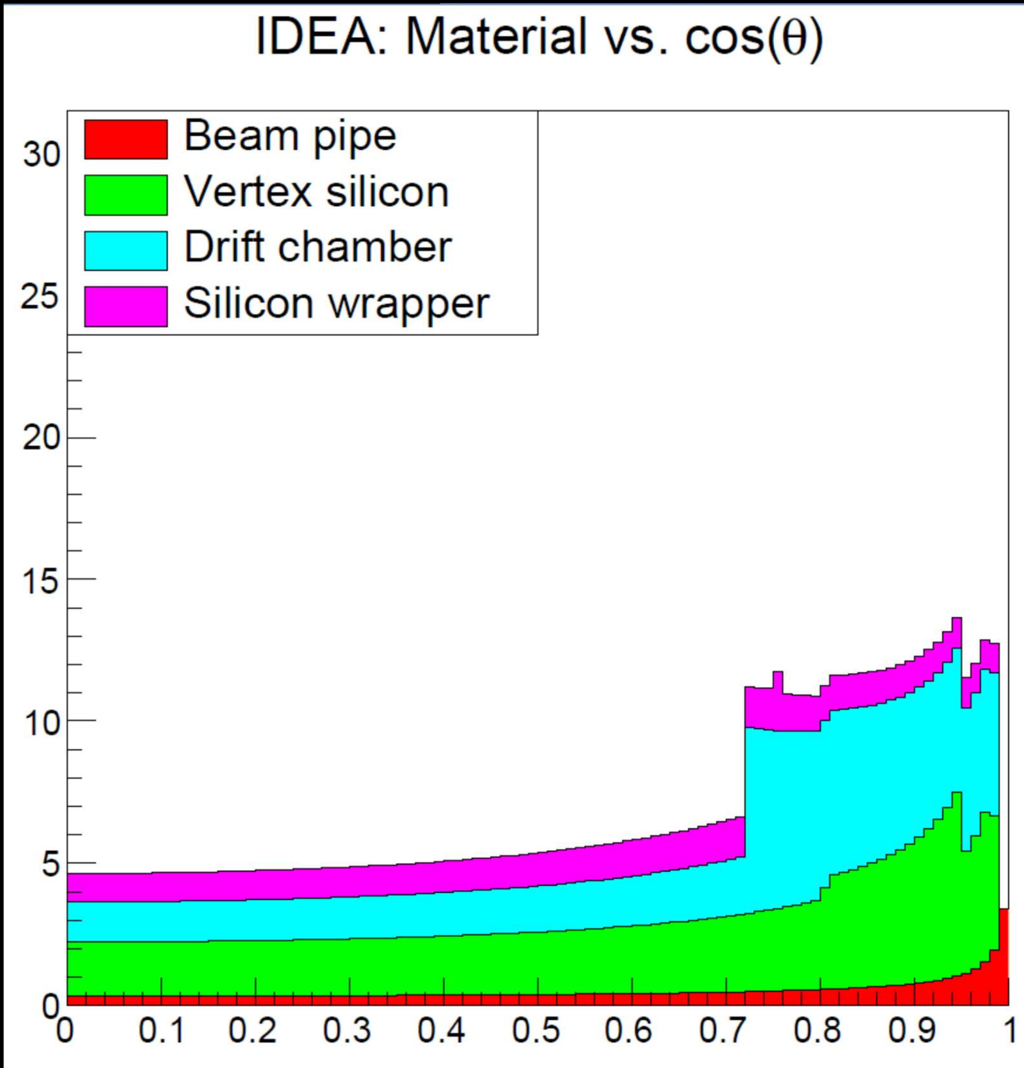
Examples (1)

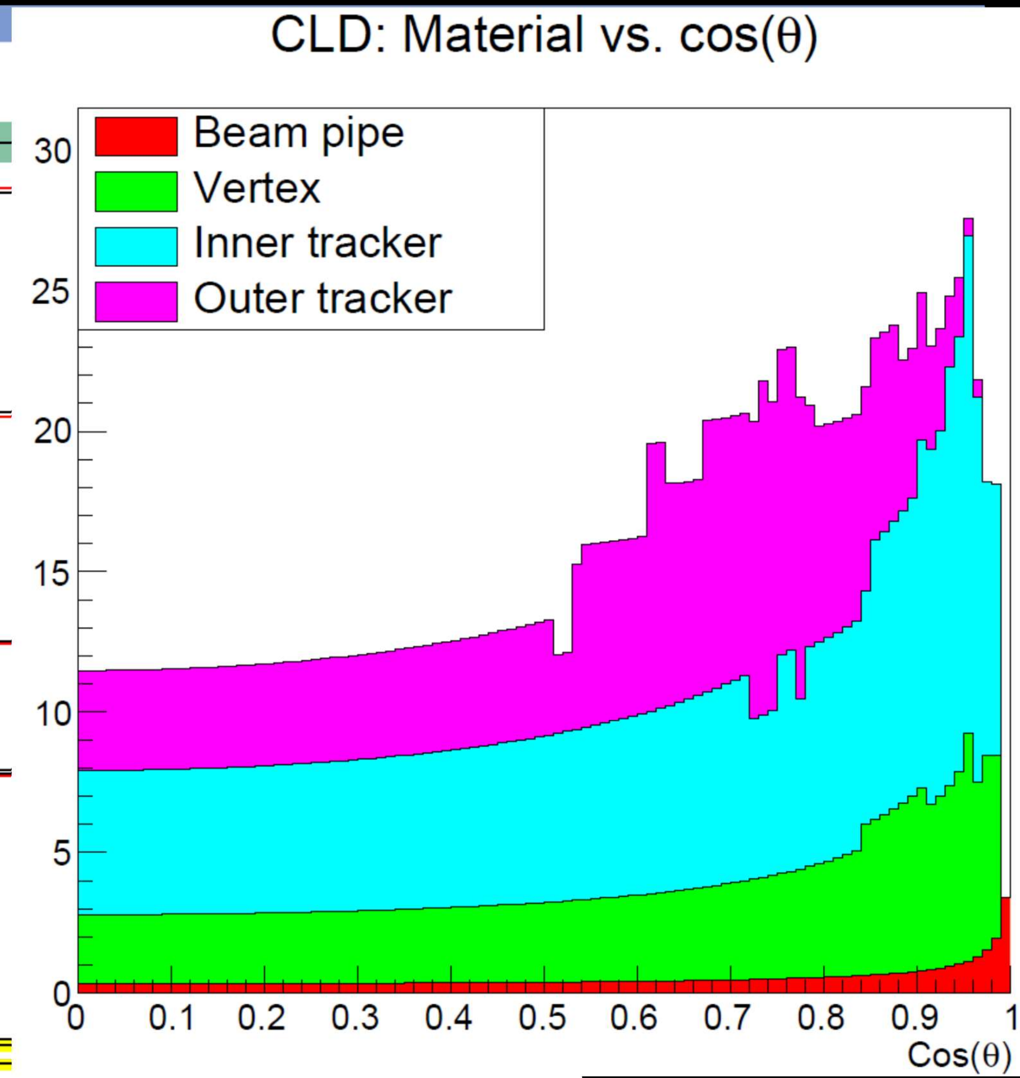
- Implemented geometry and material summary



Examples (1)

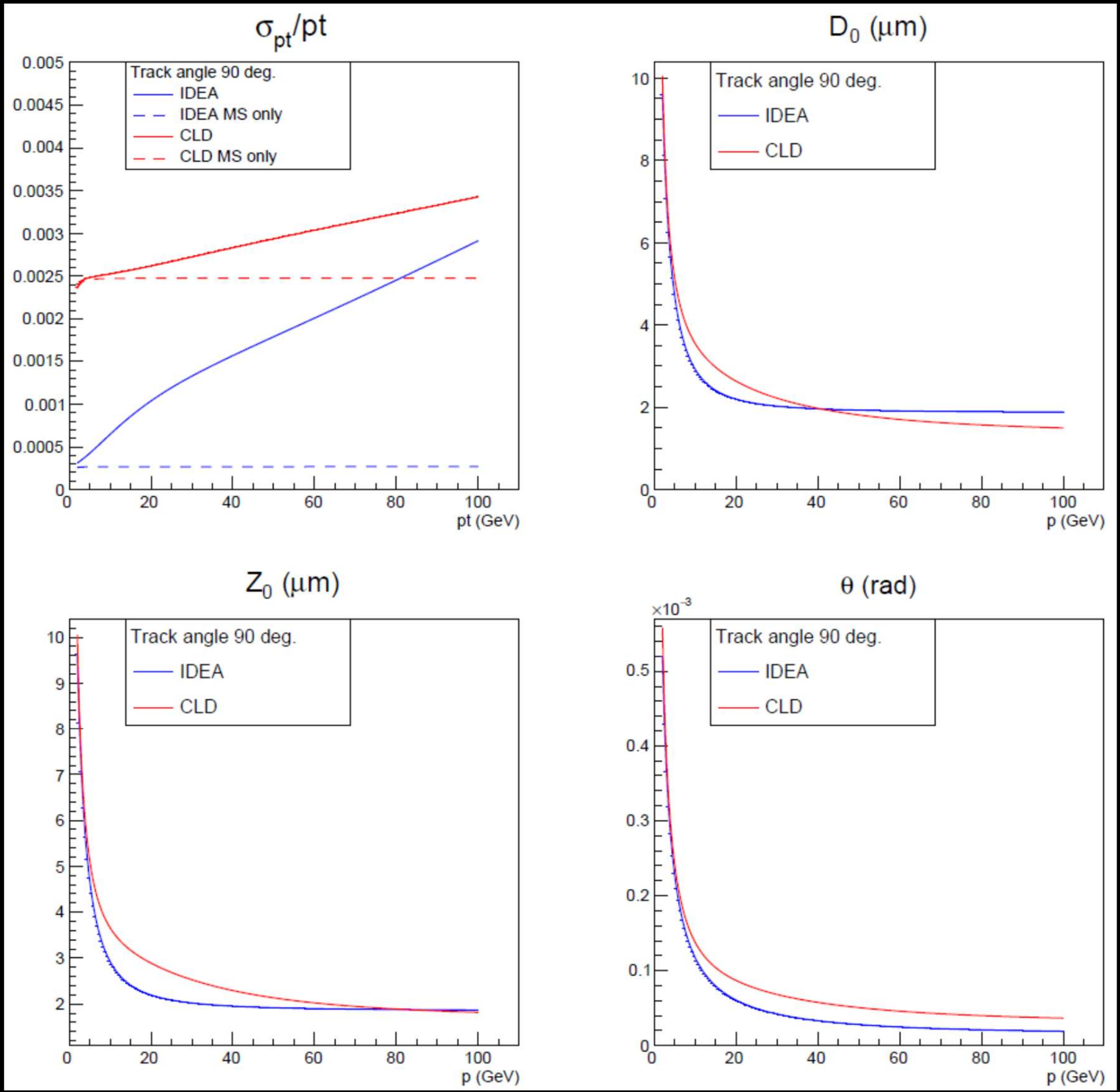
❖ Implemented geometry and material summary





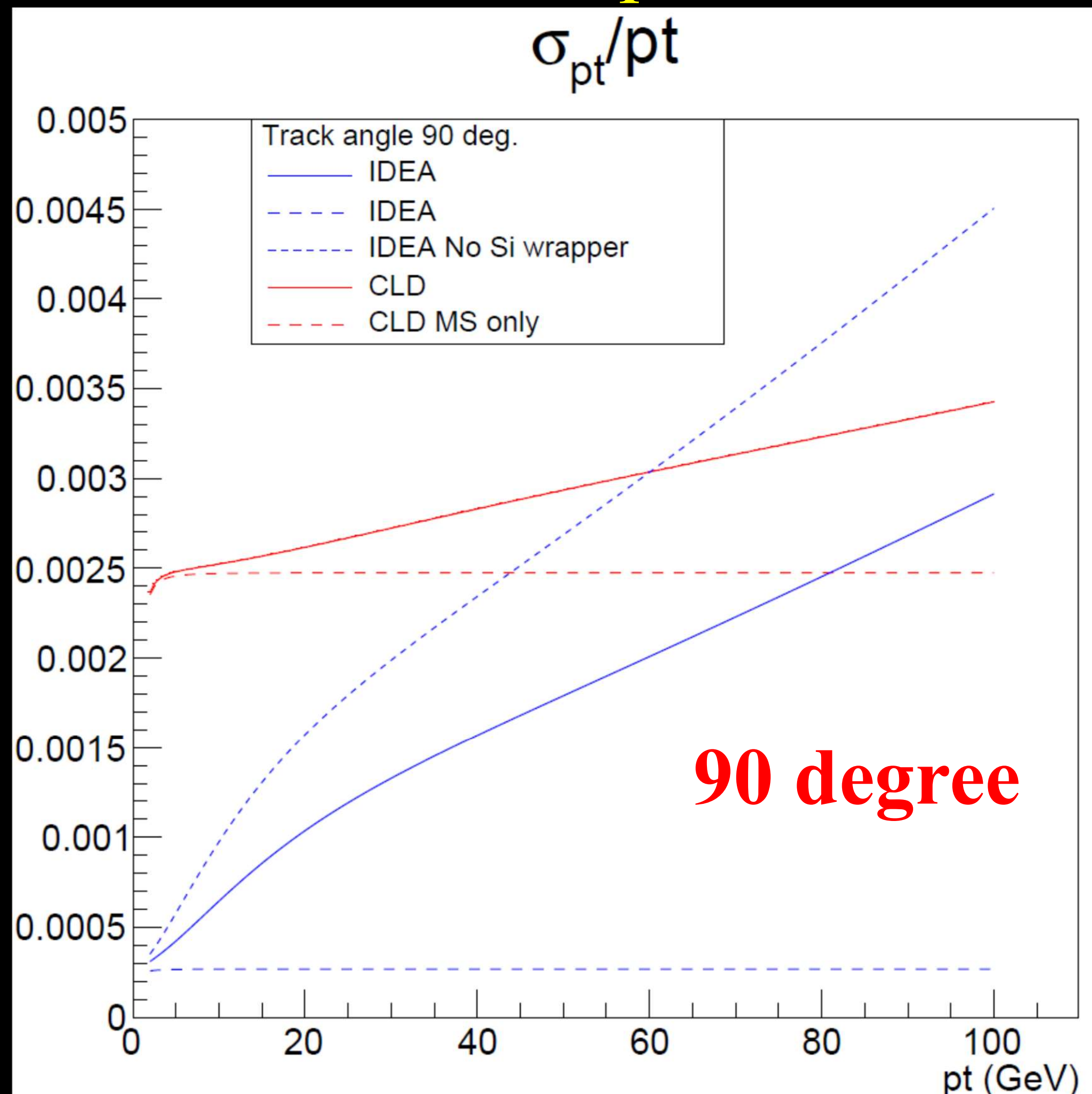
Examples (2)

❖ IDEA / CLD comparison



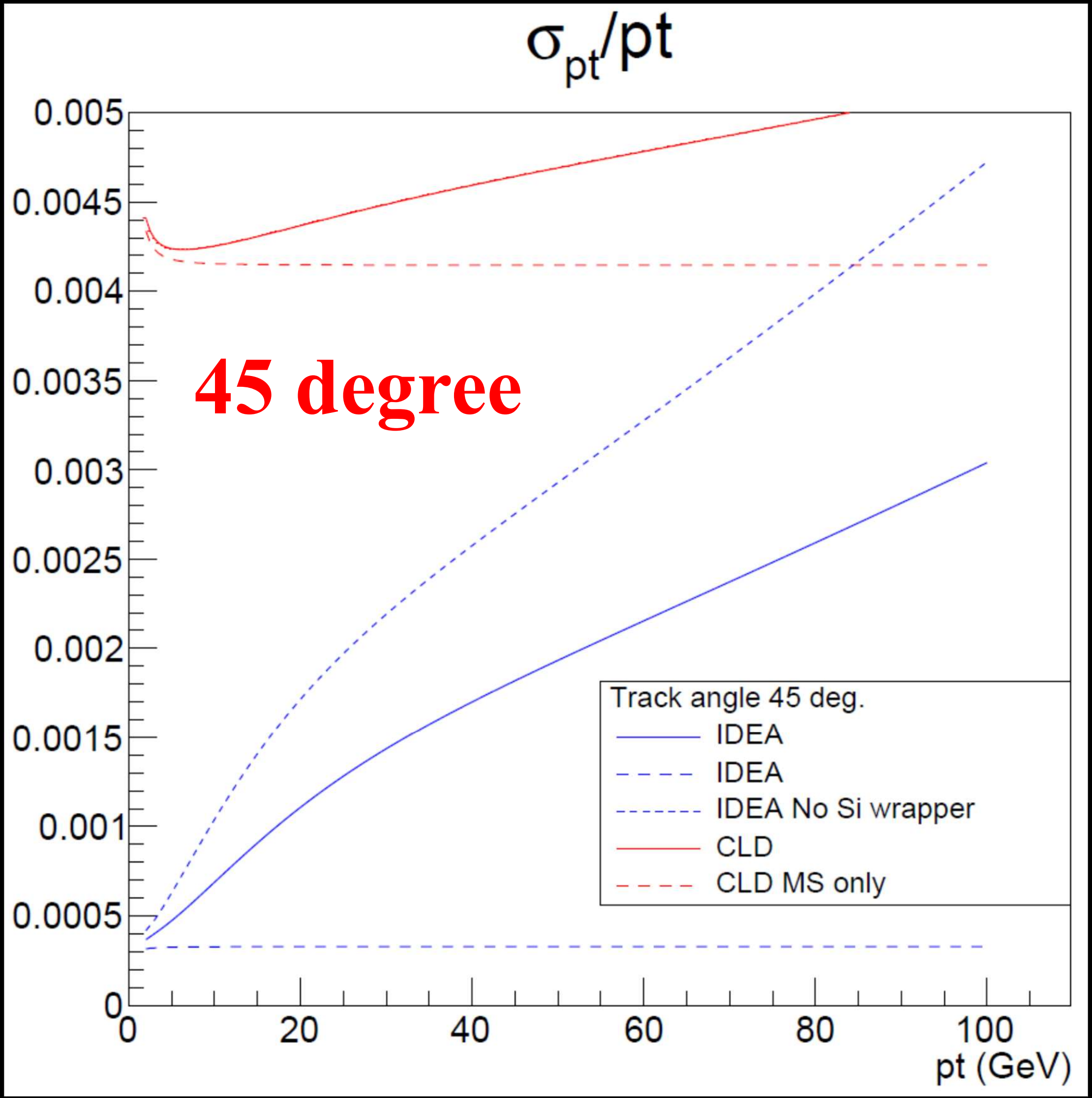
Examples (2)

- IDEA / CLD comparison →zoom pt resolution



Examples (2)

❖ IDEA / CLD comparison →zoom pt resolution

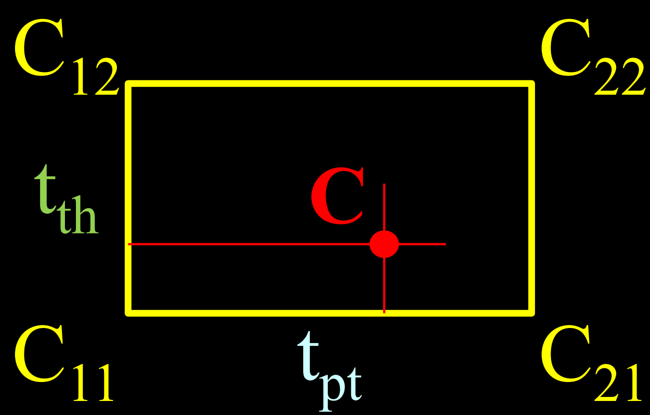


Make it faster

- Calculation of covariance matrix slow for IDEA
  - Involves inversion of matrix ~ 120x120

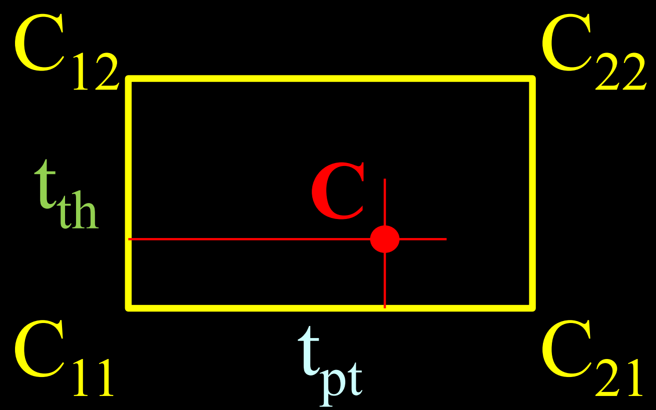
Make it faster

- Calculation of covariance matrix slow for IDEA
  - Involves inversion of matrix ~ 120x120
- Solution:
  - Store pt-polar angle grid of matrices in .root file
  - Get any matrix by bi-linear interpolation over 2D-grid ($p_t - \theta$)



Make it faster

- Calculation of covariance matrix slow for IDEA
  - Involves inversion of matrix ~ 120x120
- Solution:
  - Store pt-polar angle grid of matrices in .root file
  - Get any matrix by bi-linear interpolation over 2D-grid (p_t – θ)



- Keep full calculation for tracks originating far from IR
  - Deal with long lived particles

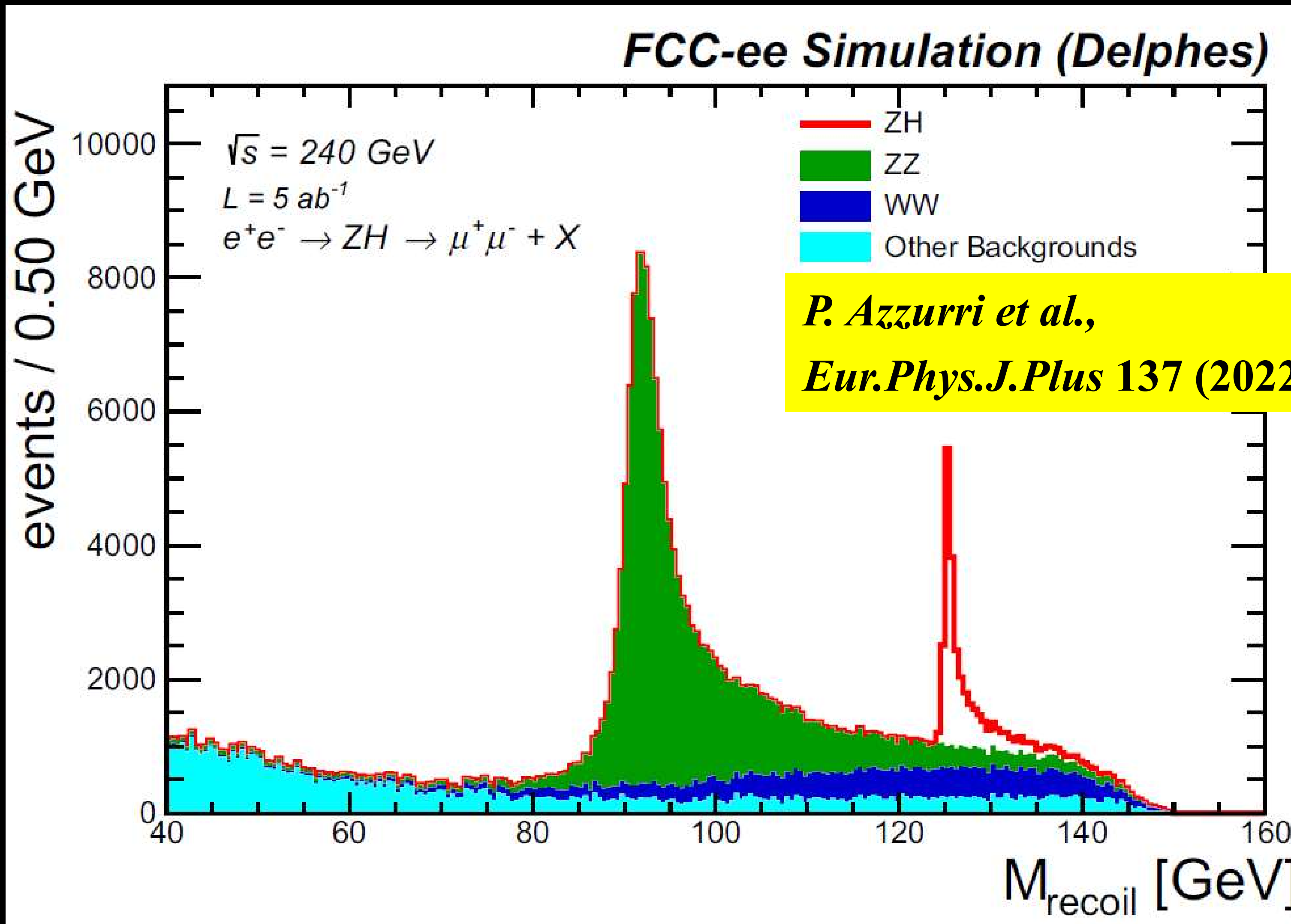
Application to simulation

- Track handling:
  - Perfect track → observed track
    - Check acceptance
    - Get covariance matrix from interpolation or full calculation
    - Smear helix parameters according to covariance matrix using Choleski decomposition

$$
\begin{aligned}
C &= \text{Covariance matrix} \\
C &= U^T U \text{ (U is upper triangular matrix) - Choleski decomposition} \\
\vec{r} &= \text{vector of normal random numbers } \mu = 0,\ \sigma = 1 \\
\vec{x} &= U^T \vec{r} \rightarrow \vec{x} \text{ has covariance } C\text{. Proof:} \\
Cov(\vec{x}) &= <\vec{x} \cdot \vec{x}^T> = U^T <\vec{r} \cdot \vec{r}^T> U = U^T I U = U^T U = C
\end{aligned}
$$

Example (3)

- Higgs recoil from HZ ($Z \rightarrow \mu^+\mu^-$)



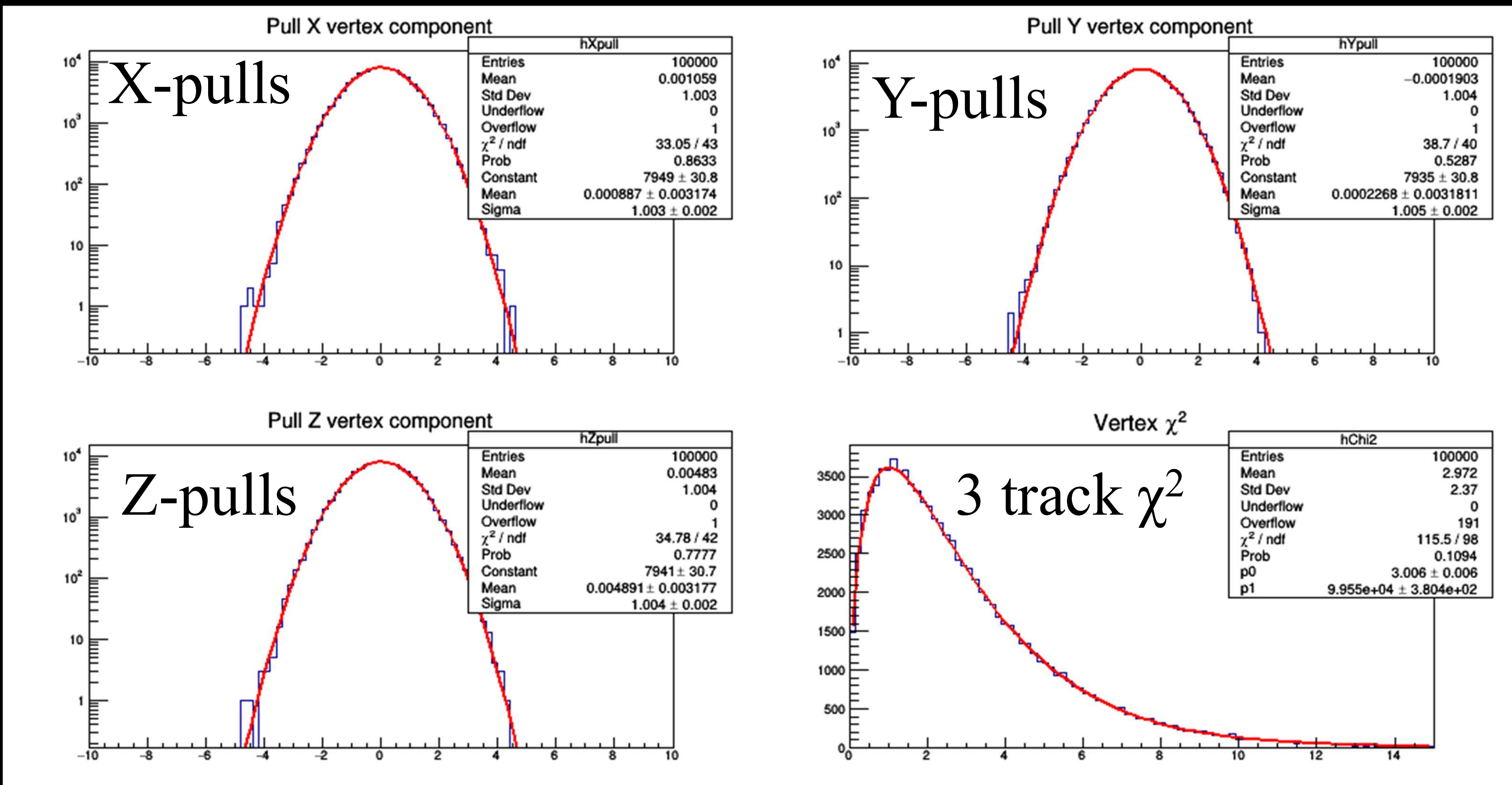
***P. Azzurri et al., Eur.Phys.J.Plus* 137 (2022) 1, 23**

Vertex finding/fitting

- Simple class for vertex fitting for post-processing DELPHES simulated events (external/TrackCovariance/VertexFit.cc)
  - Input: array of tracks (parameters + covariance matrices)
    - Only charged → assume simple helix
  - Output:
    - Vertex + associated covariance matrix
    - Chi2 and chi2 contribution of each track
  - Additional features:
    - External gaussian constraint
    - Add/Remove track
- Simple root macro to find primary vertex:
  - ExamplePVtxFind.C included in DELPHES repository

Vertex finding/fitting

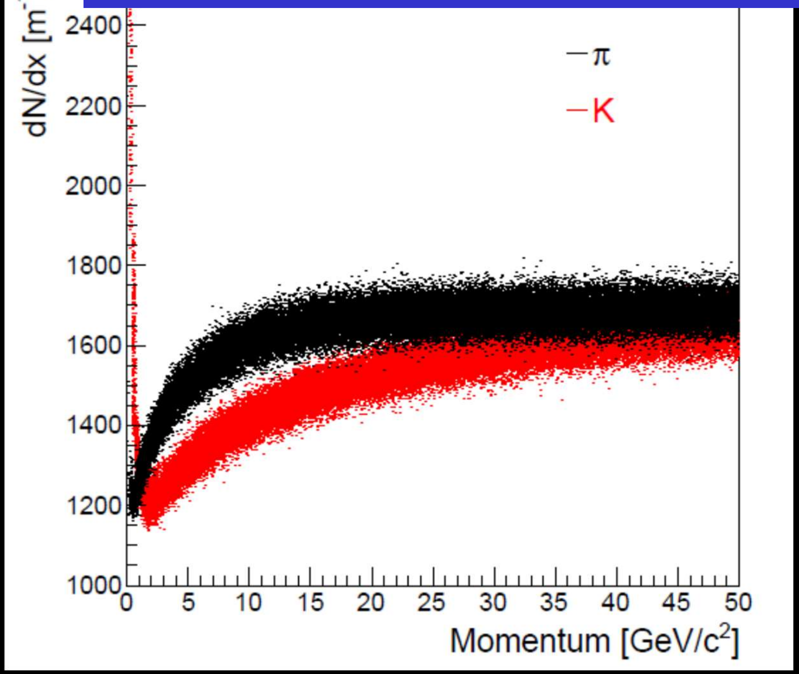
- Simple class for vertex fitting for post-processing



PID (1)

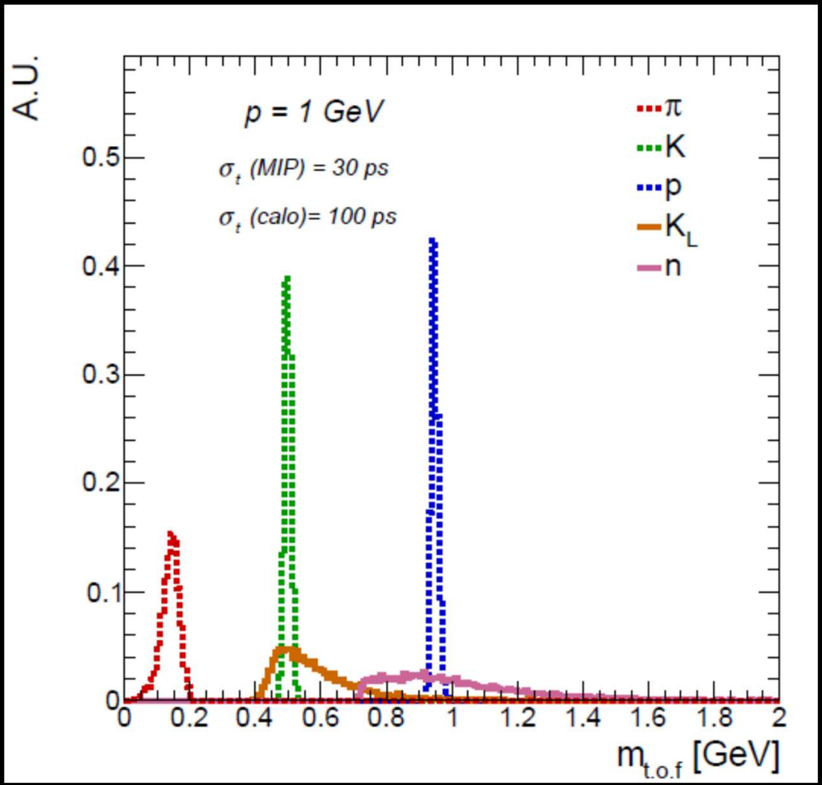
- **Cluster counting:**
  - Count # ionization clusters in drift chamber gas
    - ~ 2x better than dE/dx
  - # clusters/length vs βγ from HEED simulation embedded in Garfield++
  - Procedure:
    - Get track length in gas volume
    - Avg. # clusters from βγ
    - Extract from Poisson distribution

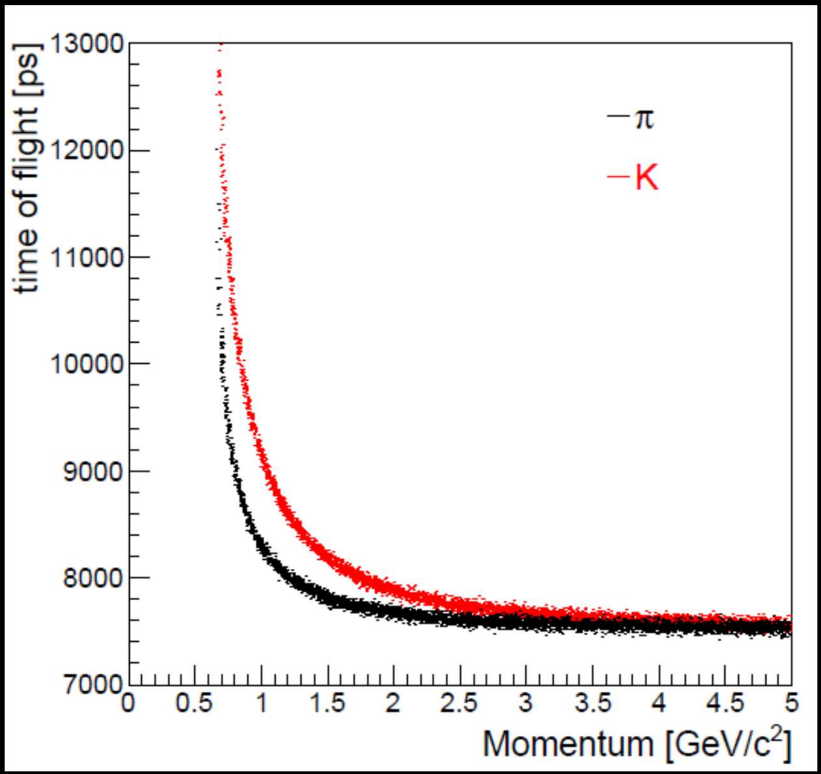
90% He, 10% Isobutane



PID (2)

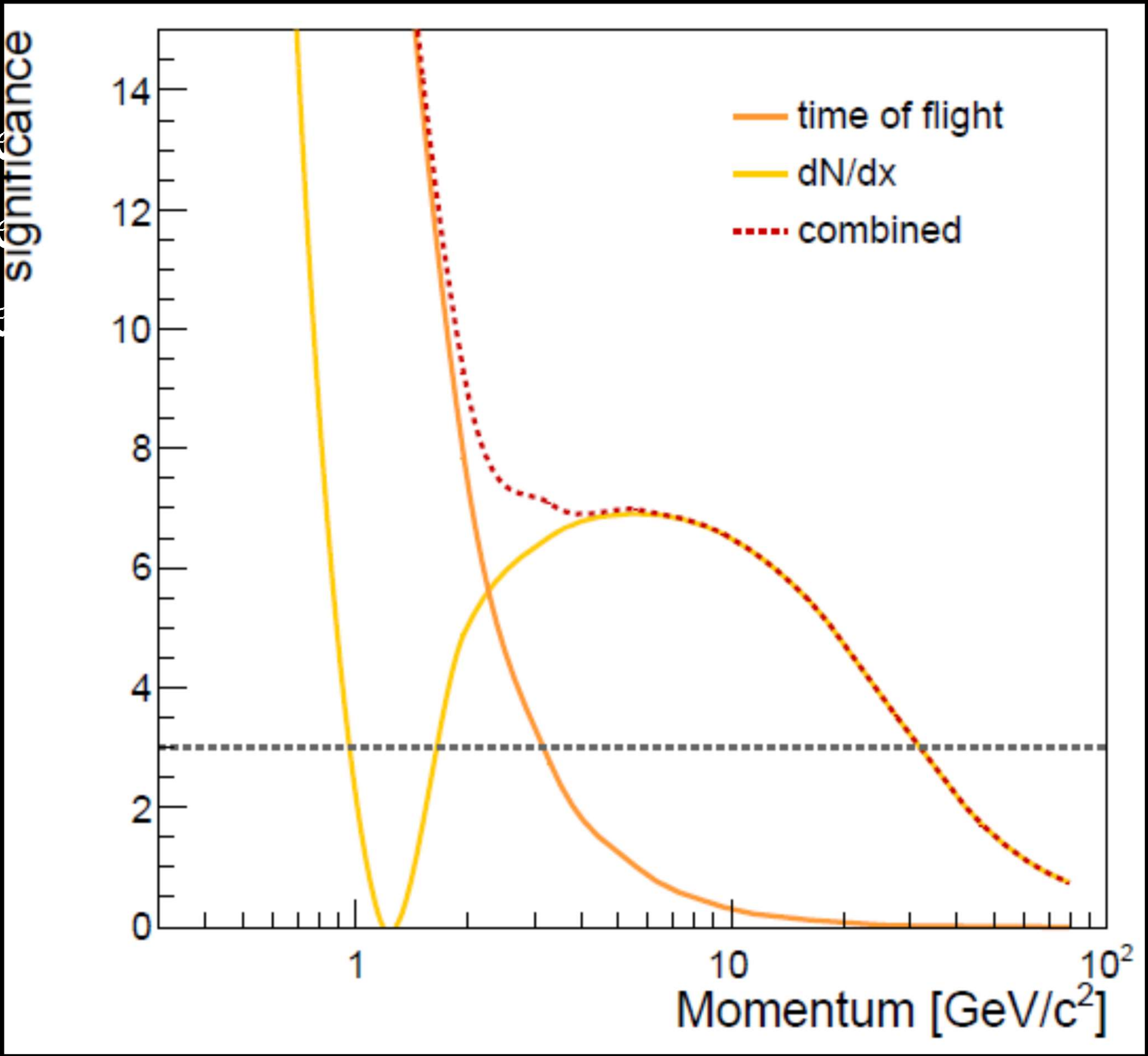
- Timing:
  - Assume timing layers at end of tracking volume
  - Assume track starting vertex has negligible error
  - 30 ps resolution for charged/ 100 ps for neutrals





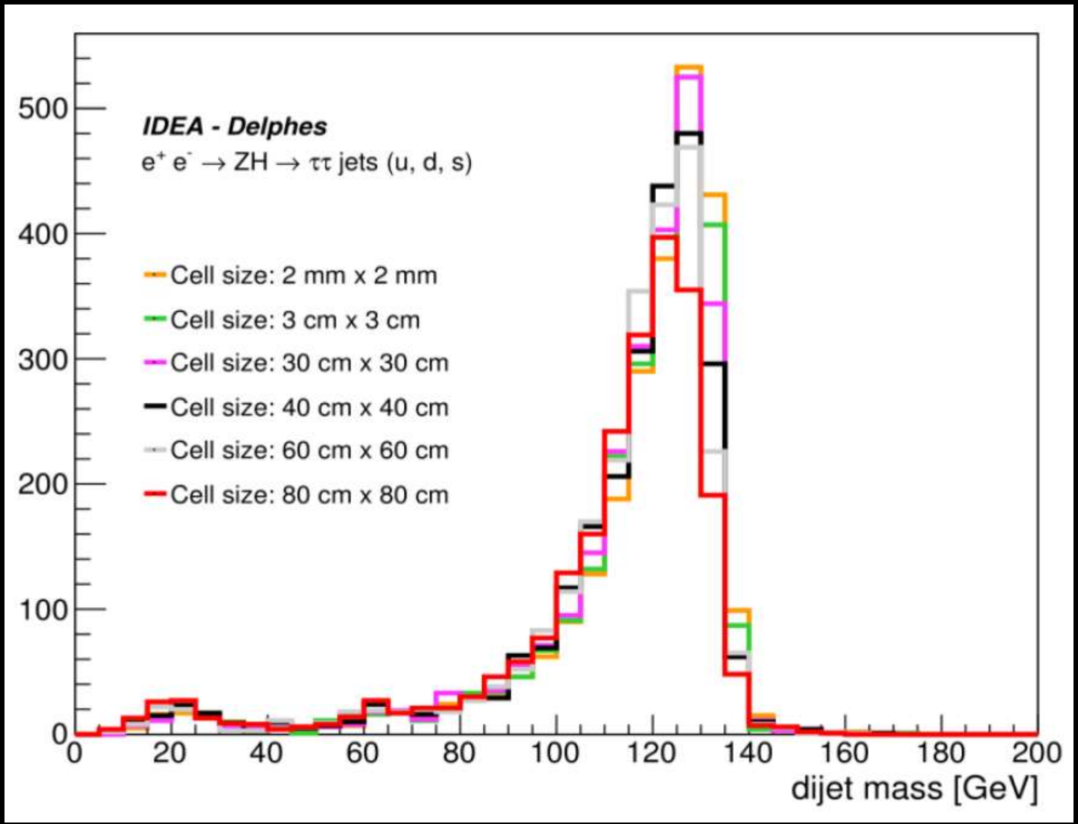
PID (2)

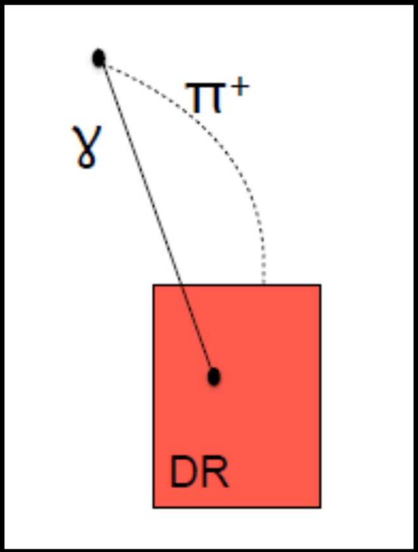
- Timing:
  - Assume
  - Assume
  - 30 ps re



DR calorimetry

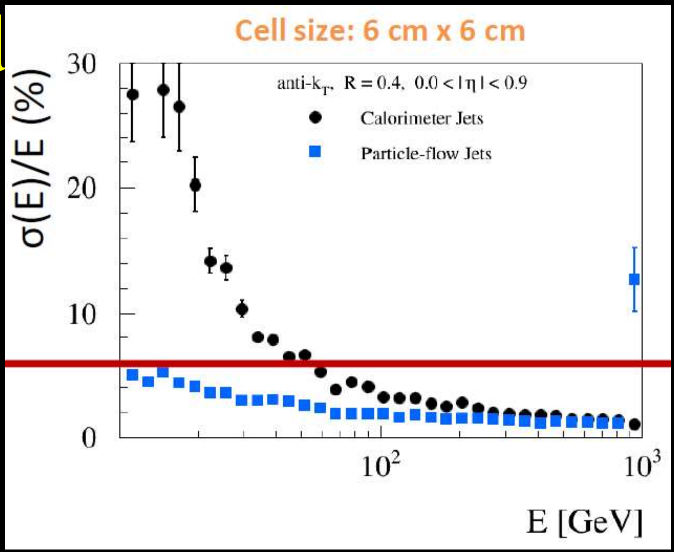
- Approximate simulation of a complex detector
  - Define virtual towers (optimal 6x6 cm^2)
  - Assign EM resolution if only e, γ hit tower
  - Assign HAD resolution if at least a hadron hits the tower





DR calorimetry

- Approximate simulation of a complex d
  - Define virtual towers (optimal 6x6 cm^2)
  - Assign EM resolution if only e, γ hit tower
  - Assign HAD resolution if at least a hadron



- Apply simple particle flow
  - Case 1: tower energy consistent with tracks only
    - Use track momentum measurement
  - Case 2: tower energy consistent with track+neutrals
    - Use calorimeter tower energy

Conclusion

- IDEA detector fully implemented in DELPHES

Conclusion

- IDEA detector fully implemented in DELPHES
- Realistic fast tracking and timing modules can be used for any detector concepts

Conclusion

- IDEA detector fully implemented in DELPHES
- Realistic fast tracking and timing modules can be used for any detector concepts
- Cluster counting is also general for any gas detector including a large TPC

Additional Slides

Implementation details

❖ Typical geometry block

```
//
// Vertex  detector (inner)
if (fEnable[1])
{
        const Int_t NlVtx = 3;                              // Assume 3 vertex pixel layers
        Double_t rVtx[NlVtx] = { 1.7, 2.3, 3.1 };           // Vertex layer radii in cm
        Double_t lVtx[NlVtx] = { 11.0, 15.0, 20.0 };        // Vertex layer half length in cm
        for (Int_t i = 0; i < NlVtx; i++)
        {
                ftyLay[fNlay] = 1;                          // Layer type 1 = R (barrel) or 2 = z (forward/backward)
                fxMin[fNlay] = -lVtx[i] * 1.e-2;            // Minimum dimension z for barrel  or R for forward
                fxMax[fNlay] = lVtx[i] * 1.e-2;             // Maximum dimension z for barrel  or R for forward
                frPos[fNlay] = rVtx[i] * 1.e-2;             // R/z location of layer
                fthLay[fNlay] = 280.E-6;                    // Thickness (meters)
                frlLay[fNlay] = 9.370e-2;                   // Radiation length (meters)
                fnmLay[fNlay] = 2;                          // Number of measurements in layers (1D or 2D)
                fstLayU[fNlay] = 0;                         // Stereo angle (rad) - 0(pi/2) = axial(z) layer - Upper side
                fstLayL[fNlay] = TMath::Pi() / 2.;          // Stereo angle (rad) - 0(pi/2) = axial(z) layer - Lower side
                fsgLayU[fNlay] = 4.E-6;                     // Resolution Upper side (meters) - 0 = no measurement
                fsgLayL[fNlay] = 4.E-6;                     // Resolution Lower side (meters) - 0 = no measurement
                fflLay[fNlay] = kTRUE;                      // measurement flag = T, scattering only = F
                fNlay++; fBlay++;
                fNm++;
        }
}
```

Beam energy spread

- Courtesy of P. Janot (FCC week 2018 – Amsterdam)

Z pole scan strategy and √s spread measurement

	Z	W	H (ZH)	ttbar
beam energy [GeV]	45.6	80	120	175-182.5
arc cell optics	60/60	60/60	90/90	90/90
emittance hor/vert [nm]/[pm]	0.27/1.0	0.84/1.7	0.63/1.3	1.4/2.8
β* horiz/vertical [m]/[mm]	**0.15/.8**	**0.2/1**	**0.3/1**	**1/1.6**
SR energy loss / turn (GeV)	0.036	0.34	1.72	9.21
total RF voltage [GV]	0.10	0.75	2.0	8.8-10.3
energy acceptance [%]	±1.3	±1.3	±1.7	±2.4-2.8
energy spread (SR / BS) [%]	0.038 / **0.132**	0.066 / **0.165**	0.099 / **0.165**	0.15 / **0.20**
bunch length (SR / BS) [mm]	3.5 / 12.1	3.0 / 7.5	3.15 / 5.3	2.75 / 3.80
bunch intensity [10^{11}]	1.7	2.3	1.8	3.2-3.35
no. of bunches / beam	16640	1300	328	40-33
beam current [mA]	1390	147	29	6.4-5.4
SR total power [MW]	100	100	100	100
luminosity [10^{34} cm^{-2}s^{-1}]	230	34	8.5	1.9-1.7
luminosity lifetime [min]	70	24	18	25
allowable asymmetry [%]	±5	±3	±3	±3

Patrick Janot FCC Week in Amsterdam 10 Apr 2018 1

$$\sigma_{beam} = 0.165\%$$

$$\sigma_{\sqrt{s}} = \sigma_{beam}/\sqrt{2} = 0.136\%$$

Higgs from Z recoil