IDEA detector concept Full Simulation description and performance

Flavia Cetorelli[1,2†], Wonyong Chung[1,2†], Marco T. Lucchini[1,2†], Armin Ilg[1,2†]

[1*]Department, Organization, Street, City, 100190, State, Country.
[2]Department, Organization, Street, City, 10587, State, Country.
[3]Department, Organization, Street, City, 610101, State, Country.

Contributing authors: flavia.cetorelli@cern.ch; wonyongc@princeton.edu; marco.toliman.lucchini@cern.ch; armin.ilg@cern.ch;
[†]These authors contributed equally to this work.

1 Introduction

Both the European Strategy for particle physicsand the Snowmass Community Planning Exercisepointed to an electron-positron Higgs factory as top priority after the LHC. The preferred option for a future collider at CERN is the Future Circular Collider (FCC), a collider with an integrated programme to push both intensity and energy frontiers [**?**]. The FCC program at CERN combines in the same 100 km infrastructure an electron-positron high luminosity Higgs/Electroweak factory, followed by a 100 TeV proton-proton collider.

The IDEA (Innovative Detector for an Electron–positron Accelerator) detector proposal for an experiment along the electron-positron collider, is designed to provide:

- the state-of-the-art momentum and angular resolution for charged particles, especially at low momentum, for precision electroweak physics at the Z pole and flavor physics;
- a suitable vertex resolution to separate g, c, b, τ final states;
- a suitable jet-jet invariant mass resolution to separate W, Z and Higgs bosons giving two jets;
- Good π^0 identification for τ and heavy flavor reconstruction;
- particle identification capability to classify final states and flavor tagging.

IDEA includes a silicon pixel detector in the innermost part, an ultralight drift chamber as the main tracking device, a silicon wrapper, a magnet, a preshower, a dual readout calorimeter and a muon system. The option of using a crystal electromagnetic calorimeter to improve the electron/photon energy resolution is also inspected. This document describes the full simulation of the IDEA subdetectors and the relevant performance.

2 Software framework and tools

Here quick intro to Geant, key4hep, DD4hep. Reference to other documents.

3 Vertex detector

The IDEA vertex detector geometry is described in detail in [1] and in the mid-term report of the machine-detector interface [2]. The inner vertex detector uses ARCADIA [3] monolithic active pixel sensors (MAPS) arranged in three barrel layers, while the outer vertex detector is constructed of ATLASPix3 [4] MAPS making up two additional barrel layers at larger radii and three disks per side to cover the forward region.

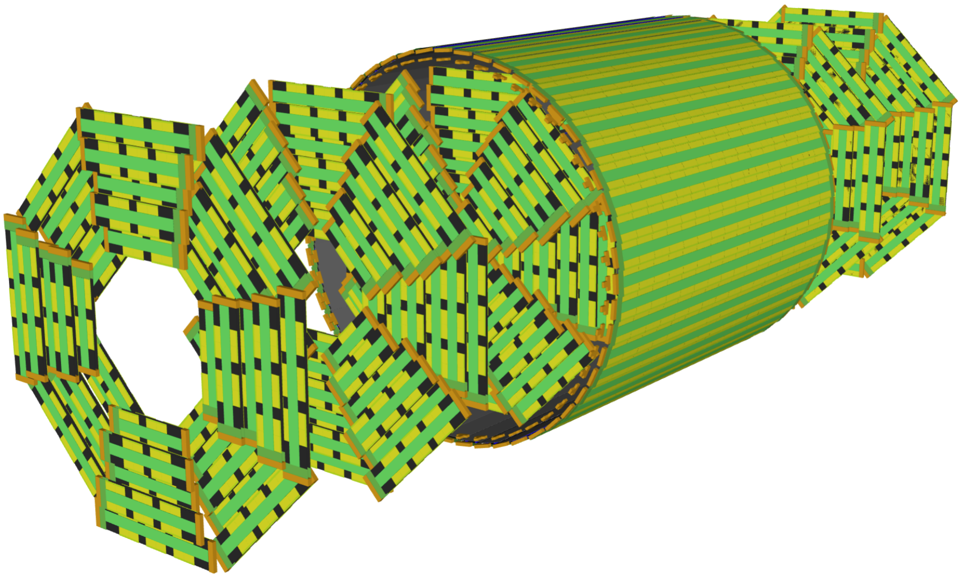

Fig. 1: Visualisation of the IDEA vertex detector geometry implemented in DD4hep.

The current implementation of the IDEA vertex detector in DD4hep describes most components of the detector geometry in detail. A visualisation of the DD4hep geometry is shown in Fig. 1. The silicon sensors are implemented as a collection of multiple sensitive and insensitive volumes to take the periphery in the sensors into account. This enables to estimate the material budget distribution in $\cos(\theta)$–ϕ and to eventually also check the hermeticity of the detector and a realistic estimation of the achievable impact and vertex resolutions.

The following sections describe how the IDEA vertex detector is described in DD4hep.

3.1 Inner vertex detector

The inner vertex detector consists of three layers with staves (or *ladders*) of dual ARCADIA modules with dimensions of $32 \times 8.4\,\mathrm{mm}^2$ in z–ϕ. The insensitive sensor periphery has a size of 2 mm in ϕ direction and is described by a separate volume. The modules are distributed along z with a spacing in between them of 0.2 mm. Fig. **??** shows the sensitive surfaces of the whole inner vertex detctor. The next component at larger r is the support structure that consists of a layer of glue and carbon fiber. On top of the staves there are two flex cables to power and read out the modules, described by stave-long volumes of Kapton and Aluminium. The stave's structural stability is furthermore insured by a light-weight holder structure made of carbon fiber and rohacell between the two flexes. The density of the holder material has been reduced to take into account the light-weight construction manner of it. At the two ends of the staves, a volume is placed to represent the hybrid circuit. Finally, the structure holding the inner vertex detector in place is imported using DDCAD, but the CAD file still needs to be adjusted to the latest changes in the detector design.

[Picture of sensitive surfaces]

Fig. 2 shows some staves of the innermost layer and the complete inner vertex detector. The material budget of this subdetector is shown in Fig. **??**, compatible with the original assumption of $0.3\% X/X_0$ in the FCC CDR.

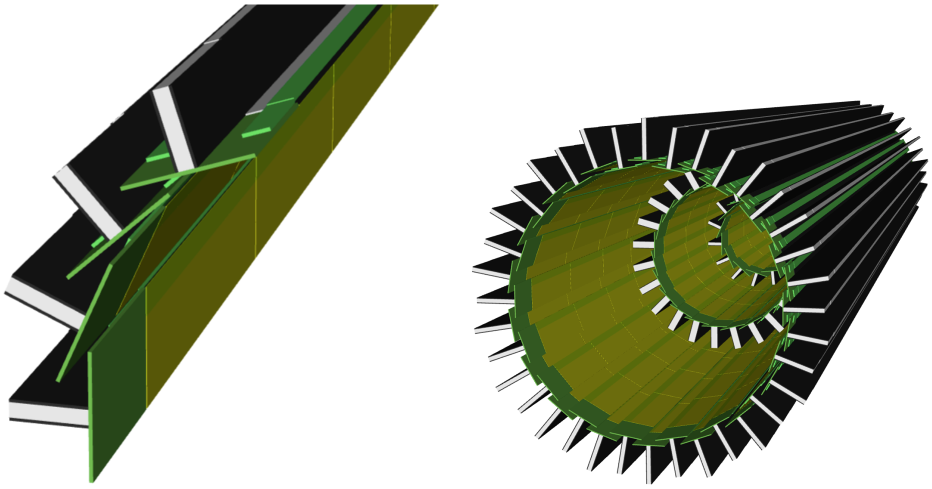

Fig. 2: One quarter of the first inner vertex barrel layer (left) and the complete IDEA inner vertex barrel (right).

[INNER SYSTEM MATERIAL BUDGET PLOTSincrease in luminosity []

3.2 Outer vertex detector

Both the outer vertex barrel and disks are made up of staves of ATLASPix3 quad modules. In the DD4hep geometry implementation, the quad module is described by four sensitive sensor volumes of $19.8 \times 18.6\,\mathrm{mm}^2$ (sensitive part of front-ends) and nine insensitive sensor volumes to model the front-end peripheries and the area in-between the four front-ends. The quad modules are placed along the z (outer vertex barrel) or the r–ϕ direction (outer vertex disks). Readout flexes, cooling pipes and a support structure consisting of carbon fiber, paper graphite and carbon fleece go along the stave, with some differences between the outer vertex barrel and disks: In the disks, the staves have ATLASPix3 quad modules on both sides of the stave to cover the full r–ϕ direction, while in the outer vertex barrel the modules are only mounted on one side of the barrel with a 0.2 mm gap in-between them.

The outer vertex barrel also features a hybrid circuit at both ends of the staves, which are supported by a light-weight truss structure made out of carbon fiber, described by a proxy volume with the according thickness. The staves are then hold in place by terminal structures made out of PEEK (Polyether ether ketone), again represented by proxy volumes of the correct mass. Fig. 3 shows a stave of the outer vertex barrel and the complete system, while Fig. **??**

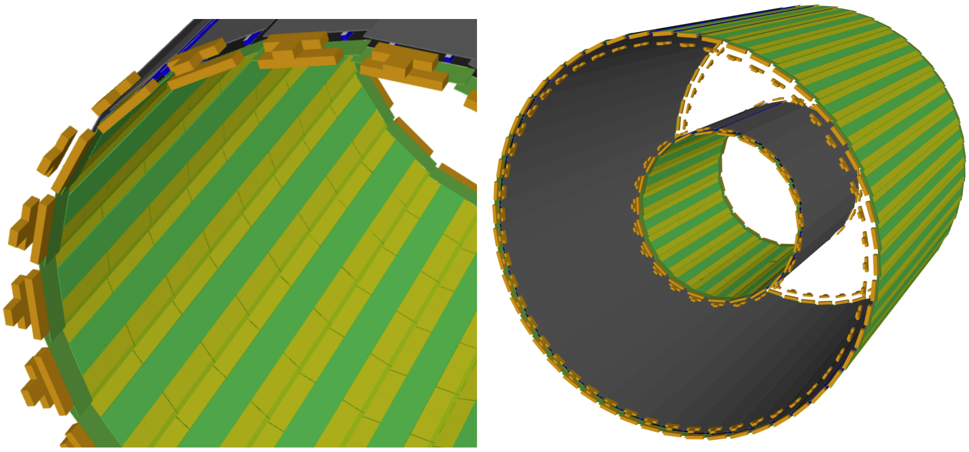

Fig. 3: Zoom-in to the first outer vertex barrel layer (left) and the full outer vertex barrel system (right).

[MATERIAL BUDGET PLOT]

The outer vertex disk staves feature the hybrid circuit only on one side, but again PEEK terminal structures on both sides. The staves are placed at different r and z to form petals. Petals at different ϕ for a face, where two faces make up a disk. The positioning of the various staves is intricate and it has to be checked that no holes in the detector coverage exist. The large structure that holds all disk staves in place is not added yet to the simulation, but should not add a lot of material budget compared to the existing components in the simulation. Fig. 4 shows a stave, the centre of the first disk and the complete outer vertex disk system while Fig. **??** shows the material budget.

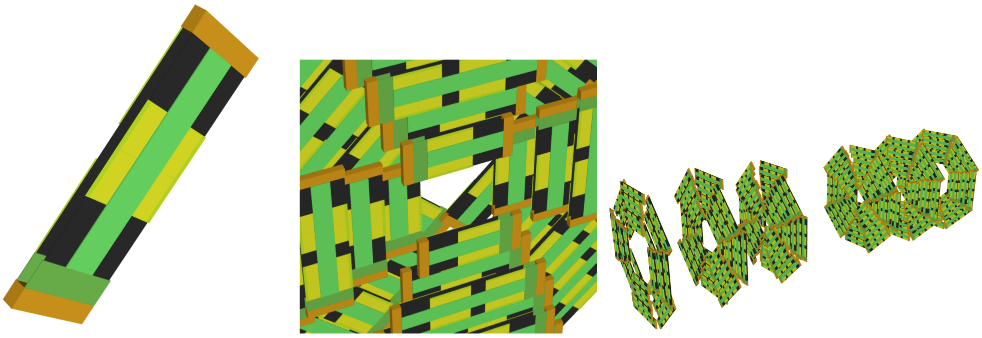

Fig. 4: One stave (left), arrangement of staves to form the first disk (middle) and the complete outer vertex disk system (right).

[MATERIAL BUDGET PLOT]

3.3 Digitisation of hits

The digitisation can be performed either using the DDPlanarDigiProcessor from iLCSoft using the k4MarlinWrapper (or directly using XXX) or with k4RecTracker. Tracking and vertexing can be done using the iLCSoft algorithms via k4MarlinWrapper.

The next step is to check for cracks in the detector coverage and estimate the impact parameter and vertex resolution of the whole system. The vertex detector model in DD4hep is detailed but a few structures are missing such as the disk support structure holding all the staves in place. Also the outer support tube, holding the central beam pipe and the vertex detector, needs to be implemented.

4 Drift Chamber

Description by Brieuc Francois, Walaa Elmetenawee, Franco Grancagnolo.

4.1 Geometry simulation

4.2 Local reconstruction

4.3 Cluster counting technique for PID

The IDEA drift chamber is designed to provide efficient tracking, a high-precision momentum measurement and excellent particle identification by exploiting the application of the cluster counting technique. To investigate the potential of the cluster counting techniques on physics events, a simulation of the ionization cluster generation is needed, therefore we developed algorithms that can use the energy deposit information provided by the Geant4 toolkit to reproduce, in a fast and convenient way, the cluster number and cluster size distributions. Indeed, the ionization of matter by charged particles is the primary mechanism used for particle identification (dE/dx), but the large uncertainties in the total energy deposition represent a limit to the particle separation capabilities. The cluster counting technique (dN/dx) takes advantage of the primary ionization Poissonian nature and offers a more statistically significant way to infer the mass information [5].

4.3.1 Principles of cluster counting techniques

In general, the drift chambers can provide a measurement of the energy loss along the particle trajectory, that, if associated with a measurement of the momentum, allows for the derivation of the mass of an ionizing particle (the traditional dE/dx method). However, the large and inherent uncertainties in the total energy deposition, described by the Landau function, represent a serious limit to the particle identification capabilities [6].

The process of energy loss of a charged particle crossing a medium is a discrete process: a particle traversing a gas leaves a track of ionization consisting of a sequence of clusters with one or more electrons which are all released in a single act of primary ionization (see Fig. 5). The cluster counting technique takes advantage of the Poissonian nature of the primary ionization and offers a more statistically significant way to infer mass information. The main advantage of the Poissonian distribution is that its Gaussian limit is achieved when the mean value reaches 20, that is of the order of 1 cm track length for the most commonly used gas mixtures. Instead, the energy distribution follows a Gaussian shape just in thick and dense material (because of central limit theorem), but this situation is not helpful for the drift chambers [5]. We expect that the cluster counting could improve the particle identification capabilities and the analytical results shown in Fig. 6 confirm this expectation.

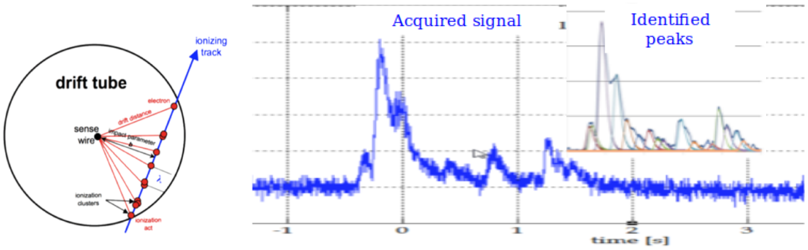

Fig. 5: Left: The section of a drift tube, with an ionizing track (blu arrow) and some ionization clusters (red dots). Right: A typical signal with the identified peaks.

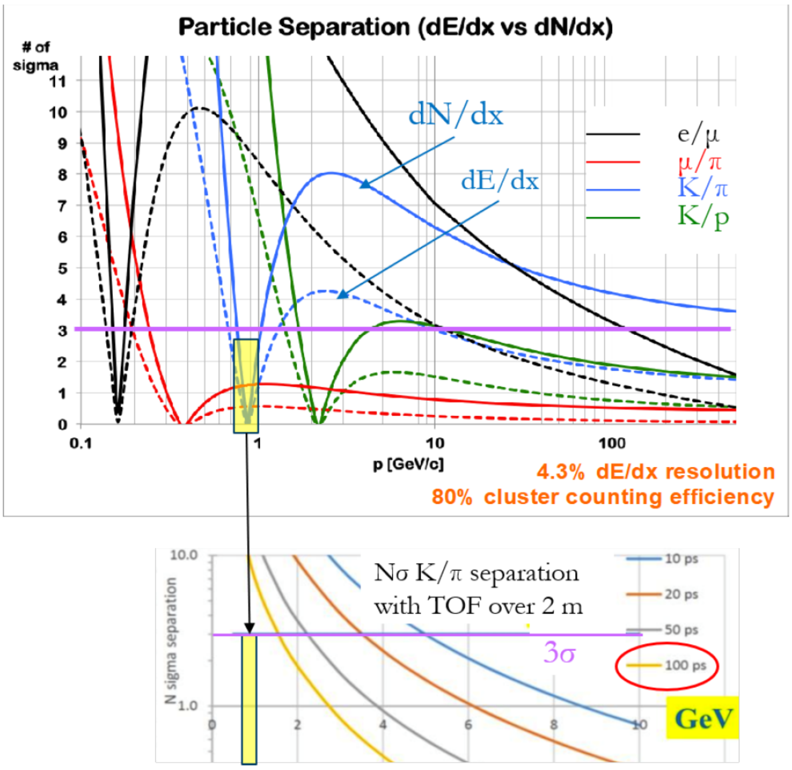

Fig. 6: Top: Analytic evaluation of particle separation capabilities achievable with dE/dx (solid curves) and dN/dx (dashed curves). The region between 0.85 GeV/c and 1.05 GeV/c where a different technique is needed is highlighted in yellow. Bottom: PID performance as a function of the time resolution by using a time of flight technique to recover the particle identification in the range 0.85 GeV/c and 1.05 GeV/c.

4.3.2 Cluster counting simulation

To assess the potential of cluster counting techniques for He-based drift chambers in physics events, it is crucial to conduct a realistic simulation of ionization cluster generation. Two suitable software tools for drift chamber simulations are Garfield++ and Geant4.

Garfield++ excels in providing detailed descriptions of the properties and performance of individual cells or drift chambers. However, it is not well-suited for simulating large-scale detectors or studying collider events. In contrast, Geant4 can simulate elementary particle interactions within complex detectors and analyze collider events. Nonetheless, it requires parameterization of fundamental properties or the development of "ad-hoc" physics models for sensitive elements like drift cells.

Our primary objective is to develop an algorithm capable of leveraging the energy deposition data provided by Geant4 to quickly and conveniently reproduce the distribution of cluster numbers and cluster sizes. While it is possible to create a physics model that integrates into Geant4 and reproduces the ionization process in detail, this approach demands significant disk space and computational time. A simpler algorithm, utilizing the energy deposition data simulated by the software, seems to be a more efficient solution.

A Garfield++ simulation analysis

To achieve this goal, we conducted simulations involving tracks passing through 200 cells, each measuring 1 cm per side, filled with a mixture of 90% He and 10% iC4H10. Our initial focus was on analyzing the cluster number distribution and energy loss distribution for muons, pions, electrons, protons, and kaons across a momentum range from 200 MeV up to 1 TeV, as reported in Fig. 7 [7]. Subsequently, we evaluated particle separation power by implementing two methods: one based on dE/dx with a truncated mean of 70% and the other based on dN/dx, as shown in Fig. 8.

The plots clearly demonstrate that the cluster counting technique enhances particle separation capabilities by a factor of 2 compared to traditional methods. For instance, at approximately 5 GeV, the separation power between a pion and a kaon (indicated by the blue squares) achieved with the traditional method is around 4 sigma, while the cluster counting technique yields a separation power of approximately 8 sigma.

In order to achieve the same results by using Geant4, we have developed an algorithm that reconstructs the distribution of cluster numbers and cluster sizes using the information provided by Geant4.

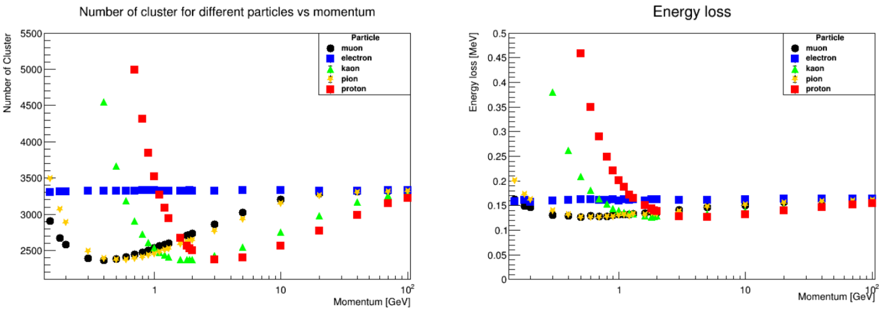

Fig. 7: Left: Clusters number distribution. Right: Energy loss distribution. [7]

Algorithm Parameters

The initial step of the algorithm begins with Garfield++ simulations. Initially, we analyze the distribution of kinetic energy for clusters with a cluster size equal to 1, as

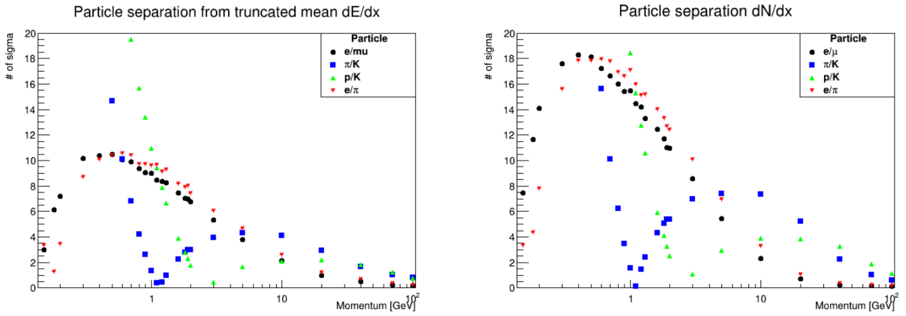

Fig. 8: Left: Particle separation power with dE/dx (truncated mean at 70%). Right: Particle separation power with dN/dx. [7]

depicted on the left side of Fig. 9, and for clusters with a cluster size greater than 1, as illustrated on the right side of Fig. 9.

Furthermore, we conduct a separate study of the distribution of clusters with a cluster size greater than 1 but up to 1 keV, which corresponds to a cutoff equivalent to the default single interaction range set in Geant4. This distribution is presented in Fig. 10.

The distribution on the left side of Fig. 9 is fitted with an exponential function combined with a Gaussian function, while the distribution on the right side of the same figure and the one in Fig. 10 are fitted with a Landau function.

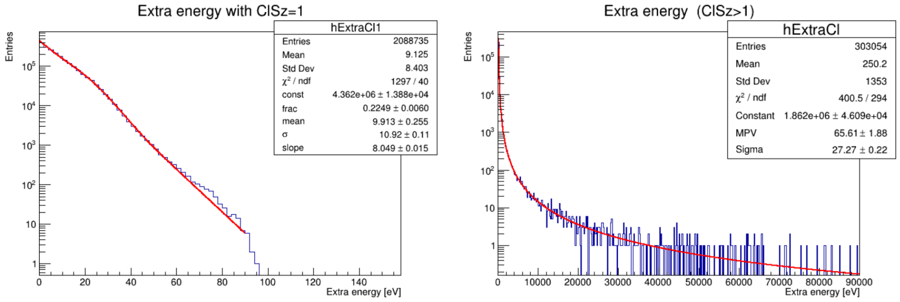

Fig. 9: Kinetic energy distribution for a muon at 300 MeV for cluster with cluster size equal to 1 (left) and for cluster size higher than 1 (right). [7]

This analysis is conducted for all particles across the entire momentum range. Subsequently, the fit results are stored and analyzed independently to be utilized during

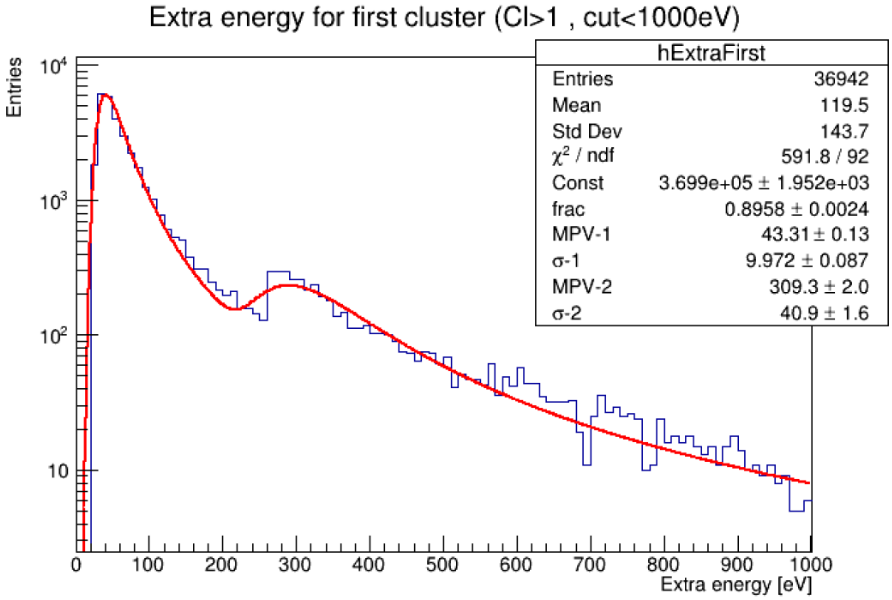

Fig. 10: Kinetic energy distribution for a cluster with cluster size higher than one for a muon at 300 MeV, up to 1 keV. [7]

the algorithm implementation. As an illustration, Fig. 11 displays the distribution of mean values resulting from the Gaussian fit of the kinetic energy for clusters with a cluster size equal to 1. This distribution is fitted with an exponential function plus a plateau function.

We then focused on determining the maximum kinetic energy required to create clusters with a cluster size greater than one. To derive this parameter, referred to as maxExEcl, we study the correlation plot displayed in Fig. 12. This plot illustrates the relationship between the total energy loss by particles traversing the gas mixture and the total kinetic energy of clusters with a cluster size greater than 1. Additionally, we evaluate the parameter ExSgm to account for the smearing effect around the mean value of the total energy loss.

The profile plot is fitted with a linear function, and the formula for determining maxExEcl is expressed as follows:

$$maxExEcl = \frac{E_{tot} - p0 + Random(Gaus(0, ExSgm))}{p1} \tag{1}$$

where, p0 and p1 represent the fit parameters of the linear fit, and E_{tot} represents the total energy loss incurred by particles traversing the 200 cells of gas. All these elements are integral to the algorithm implementation.

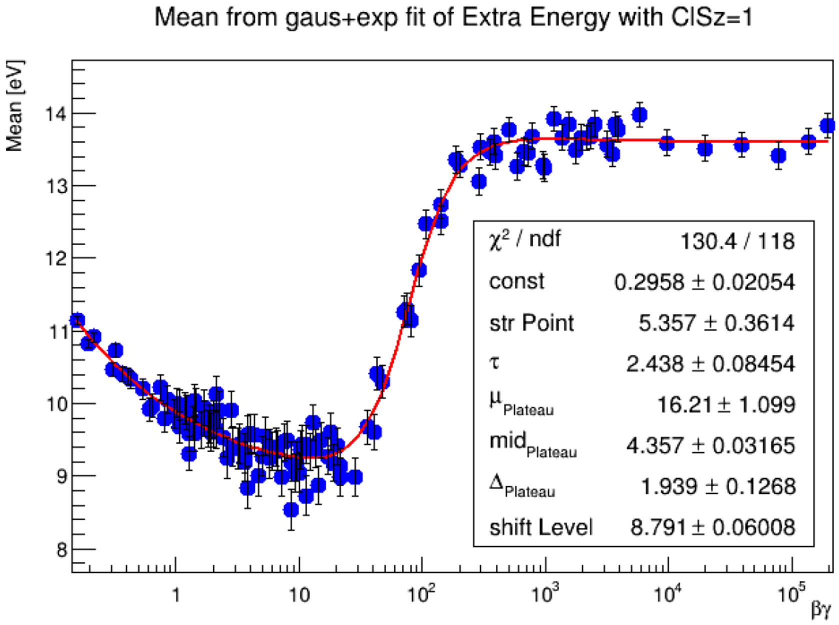

Fig. 11: Mean value distribution of Gaussian fit of the kinetic energy distribution for clusters with cluster size equal to 1, fitted with an exponential plus a plateau function. [7]

4.3.3 Algorithm implementation in Geant 4

Using the results from the Garfield++ analysis described in the section above, we tested three different algorithms (for detailed information, see ref1) and implemented the one which gave the best results consistent with the ones expected from simulations.

The algorithm starts by utilizing maxExEcl (see equation 1). If maxExEcl is higher than zero, generates the kinetic energy for clusters with cluster size higher than 1 by using its distribution and evaluating the cluster size. This process iterates until the cumulative sum of primary ionization energy and the kinetic energy per cluster reaches the event's maxExEcl threshold. Subsequently, with the remaining energy, the algorithm generates clusters with a cluster size equal to 1, assigning their kinetic energy in accordance with the appropriate distribution.

To validate the algorithms, we conducted a comparative analysis of their results against those simulated by Garfield++. As a case study, we considered a muon with a momentum of 300 MeV, for which the cluster number and cluster size distributions are presented in Fig. 13.

Fig. 14 illustrates the clusters number distribution and the cluster size distribution obtained from the Geant 4 full simulation framework for the IDEA drift chamber.

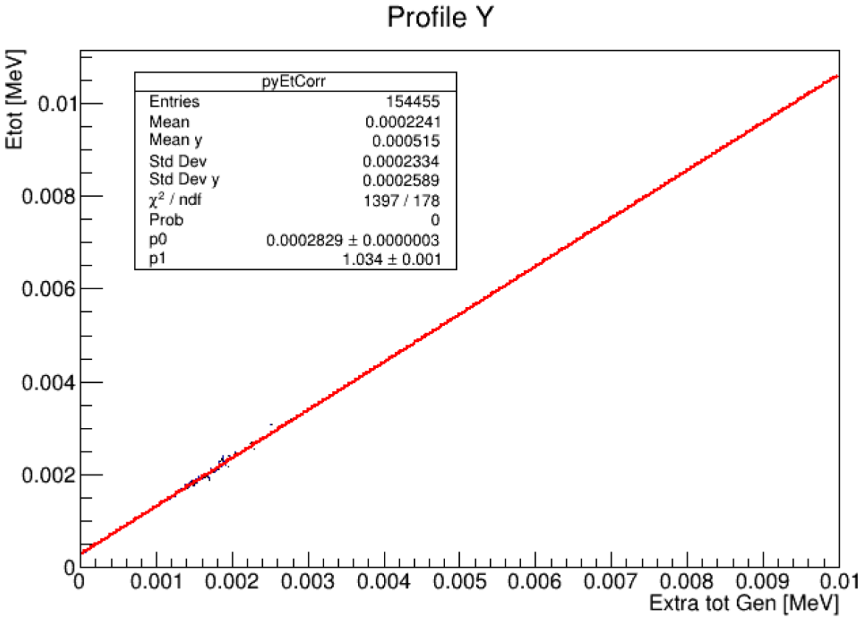

Fig. 12: Correlation plot between the total energy loss by a muon at 300 MeV traversing gas and the total kinetic energy for clusters with cluster size higher than one. [7]

The algorithm successfully reproduces the expected Poissonian shape in the clusters number distribution. Additionally, the cluster size distribution exhibits a mean value that closely aligns with our expectations.

5 Silicon Wrapper

The Silicon Wrapper is located just outside the drift chamber and will provide an outer high-precision measurement of the tracks. The additional timing capabilities of $O(30\,\mathrm{ps})$ of the system will complement particle identification as previously discussed in Fig. 6. The Silicon Wrapper geometry is described using the same detector constructors as for the IDEA vertex detector.

No engineering design of the Silicon Wrapper exists yet, which is why the current description in DD4hep is preliminary and based on many assumptions. The detector layout is guided by the following assumptions:

- Two barrel layers and two disks per side are used.
- The resulting material budget is assumed to be $\mathcal{O}(0.5\% X/X_0)$ per layer/disk.
- Two layers/disks are needed to ensure at least one hit in the Silicon Wrapper down to $\theta \approx 150\,\mathrm{mrad}$, but for large parts of the detector coverage two hits will be available.

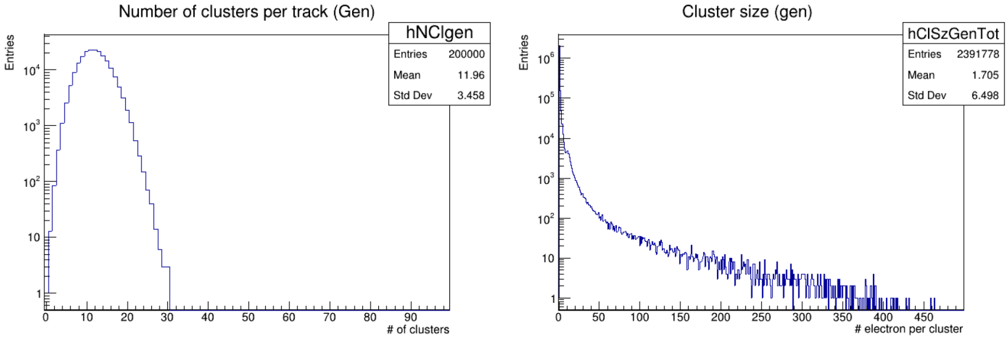

Fig. 13: Clusters number distribution (left) and cluster size distribution (right) for a muon at 300 MeV traversing 200 cells, 1 cm per side, filled with 10 % He and 90 % iC_4H_{10}. [7]

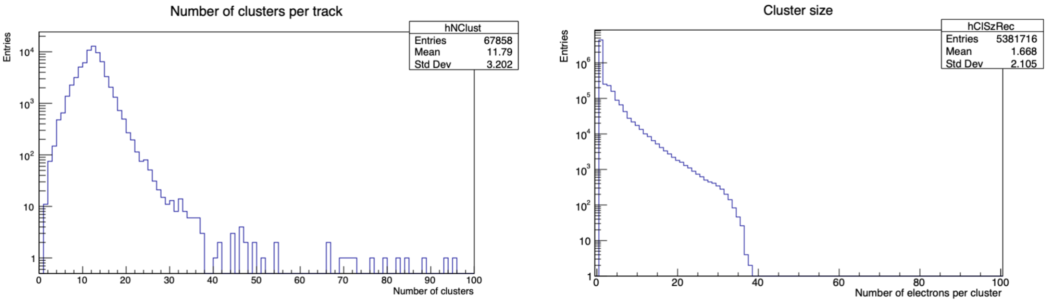

Fig. 14: Clusters number distribution (left) and cluster size distribution (right) for a muon at 300 MeV as obtained from the Geant 4 full simulation framework for the IDEA drift chamber.

- No specific sensor technology is assumed. The current implementation foresees 50 μm of silicon for the sensor.
- The pitch of the sensors is assumed to be $0.05 \times 1\,\mathrm{mm}^2$ with a timing resolution of 40 p per hit, leading to $\lesssim 30$ ps time resolution if ≥ 2 hits are recorded in the Silicon Wrapper.
- The sensors are assumed to have a size of $42.2 \times 40.6\,\mathrm{mm}^2$ – equivalent to the size of an ATLASPix3 quads, used in the vertex detector.

- Two such sensors are put next to each other, which is repeated on both sides of a carbon fiber support to form a long stave (barrel) or tile (disks), similar to the design proposed for the CEPC tracker (see [8, 9]).

The Silicon Wrapper full simulation model is built in DD4hep using the same constructor code as for the vertex detector (see Sec. section 3).

5.1 Silicon Wrapper barrel

The two layers are at $r = 2040, 2060\,\text{mm}$ and extend to $z = |2400\,\text{mm}|$. With a spacing of 3.8 mm between the staves in r–ϕ, a total of 151 staves are used for each layer with 262 sensors each. Each side of the staves has two flex cables (same as in IDEA outer vertex detector), two layers of 0.7 mm thick carbon fiber to represent the support structure and cooling pipes in between them. The two layers are offset in ϕ by half a stave width to cover the holes between the staves of each other layer. Fig. 15 shows parts of one stave as well as one quadrant of the whole Silicon Wrapper Barrel system.

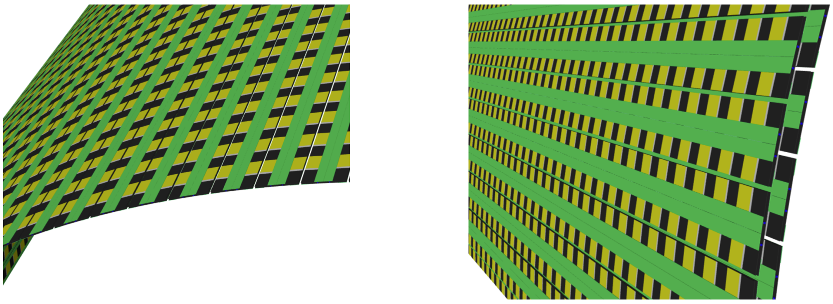

Fig. 15: Staves making up the first Silicon Wrapper layer (left) and usage of two layers (right).

[MATERIAL BUDGET PLOTS]

5.2 Silicon Wrapper disks

The disks are composed of tiles sensors. Three sizes of tiles exist: 6, 12 and 24 sensors with resulting lengths of 116.2, 227.2 and 454.4 mm. These tiles are used to fill the whole area of the disks, similarly to how is done for the CMS endcap timing layer [10]. A hole of $r = 350\,\text{mm}$ is left in the middle, where the beam pipe, vertex detector and their support tube will be inserted through, and extending to $r = 2040\,\text{mm}$. This outer radius is more than what was previously assumed (2020 mm) to better close the gap in coverage between the Silicon Wrapper barrel and disks. The second disk is offset by half a tile width to cover the gaps in detector coverage of the first, leading to also at least one silicon hit. Fig. 16 shows how sensors are placed on both sides of a tile of 24 modules and how the disk is assembled out of many such tiles.

[MATERIAL BUDGET PLOTS]

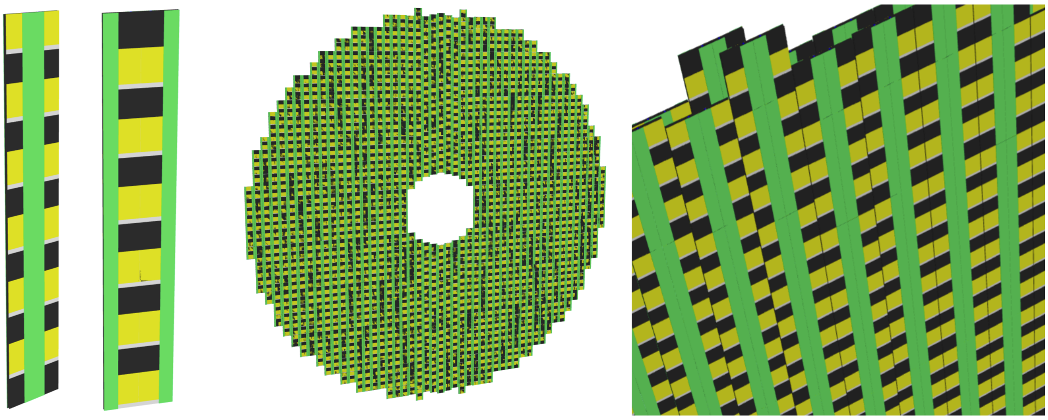

Fig. 16: Tile from front and back (left) making up the first disk (middle) and how the two disks are offset one to another (right).

6 Magnet

7 Dual Readout Calorimeter

8 Crystal Electromagnetic Calorimeter Option

Description by **Flavia Cetorelli, Wonyong Chung and Marco Lucchini**.

8.1 Overview of the calorimeter concept

- Overall motivation for high precision measurements of photons
- Description of hybrid dual-readout method and performance
- Potential for dual-readout particle flow based reconstruction
- Comment on overall detector/calorimeter performance-cost optimization

8.2 Description of geometry implementation in DD4HEP

- Projective geometry
- Granularity in dtheta, dphi (approximate crystal dimensions), dimensions of front/rear segment, description of barrel and endcap geometry and coverage
- Channel count for 10x10 mm front face
- Pictures shown by Wonyong in other talks

The projective geometry of the segmented crystal ECAL is implemented in dd4hep and takes as input the following parameters:

- Half-barrel length of the detector
- Inner radius of the detector R_{inner}
- Nominal front-face width of the front crystals
- Fixed longitudinal length of the front and rear crystals, F_{dz} and R_{dz}

Each crystal is implemented as an asymmetric trapezoid with the front face exactly normal to the IP for a fully projective and hermetic design. The rear crystals are continuous with the front crystals and the rear crystal face widths are therefore determined in part by the length of the front crystal.

To account for projective gaps, additional theta segments totaling a z span of 10 cm are inserted at $z = 0$ after the main construction loop, shifting each half of the detector $\pm$ 5 cm in z from the IP.

An integer number of segmentations in theta and phi are calculated by taking the floor of the barrel length and inner radius against the nominal crystal face width; each crystal therefore captures a constant solid angle projection from the IP, but the actual dimensions of the crystal faces will differ in theta due to the varying distances from the IP.

In the barrel no special checks are done on the crystal face dimensions since the nominal face width is generally much smaller than the detector dimensions and the aspect ratio will generally be near unity. In the endcap however, the number of phi segmentations must decrease with radius in order to maintain the aspect ratio and and a check is implemented to keep the ratio to within 15% of unity.

The theta delineation between the barrel and endcap and the $d\theta$ and $d\phi$ extent of each crystal are determined by the input dimensions. For the barrel, a single phi slice of the detector is first constructed, then rotated and copied in phi by the number of phi segments. In the endcap, since the number of phi segments decreases with radius, a single crystal is first constructed for each radius then rotated and copied in phi to form the concentric rings of the endcap.

A single crystal is implemented as an asymmetric trapezoid using the EightPointSolid shape in dd4hep, which takes as input the length of the crystal (the height z of the trapezoid) and an ordered vector of 8 (x, y) vertices that define the two bases of the trapezoid. At initialization, the EightPointSolid is centered at the origin with extent from $-z/2$ to $z/2$. The crystal is then rotated and translated to its final position based on its theta number. The position of each crystal is taken to be the center of its front face.

The vertices of each crystal trapezoid are determined by the following calculations.

[diagrams to be added]

First, we define the distance r from the IP for each of the crystal faces:

$$r_0 = r_{\text{front crystal, front face}} = R_{\text{inner}}/\sin(\theta_n) \tag{2}$$

$$r_1 = r_{\text{front crystal, rear face}} = r_0 + F_{\text{dz}} = r_{\text{rear crystal, front face}} \tag{3}$$

$$r_2 = r_{\text{rear crystal, rear face}} = r_1 + R_{\text{dz}} \tag{4}$$

$$\tag{5}$$

Projecting onto the $y - z$ plane, the half y-height of the bases of the trapezoids corresponding to the crystal faces are

$$y_0 = r_0 \tan(d\theta/2) \tag{6}$$

$$y_1 = r_1 \tan(d\theta/2) \tag{7}$$

$$\begin{align}
y_2 &= r_2 \tan(d\theta/2) \tag{8}\\
& \tag{9}
\end{align}$$

A similar simple projection to the $x - z$ plane is not sufficient to calculate the x-widths since pairs of corners at one width x are at different heights in y. The r-term is adjusted to no longer be the distance from the IP to the center of the crystal face, but rather the distance from the IP to the midpoint of the top and bottom width segments of the trapezoid bases. A total of six unique widths are therefore defined for a pair of front and rear crystals:

$$\begin{align}
x^t_{y_0} &= x^{\text{top pair}}_{\text{at } y_0} && = r^t_{y_0} \tan(d\phi/2) \tag{10}\\
x^b_{y_0} &= x^{\text{bottom pair}}_{\text{at } y_0} && = r^b_{y_0} \tan(d\phi/2) \tag{11}\\
x^t_{y_1} &= x^{\text{top pair}}_{\text{at } y_1} && = r^t_{y_1} \tan(d\phi/2) \tag{12}\\
x^b_{y_1} &= x^{\text{bottom pair}}_{\text{at } y_1} && = r^b_{y_1} \tan(d\phi/2) \tag{13}\\
x^t_{y_2} &= x^{\text{top pair}}_{\text{at } y_2} && = r^t_{y_2} \tan(d\phi/2) \tag{14}\\
x^b_{y_2} &= x^{\text{bottom pair}}_{\text{at } y_2} && = r^b_{y_2} \tan(d\phi/2) \tag{15}\\
& \tag{16}
\end{align}$$

The radial distances are:

$$\begin{align}
r^t_{y_0} &= r_0 \cos(\theta_n) + y_0 \sin(\theta_n) \tan(\theta_n - d\theta/2) \tag{17}\\
r^b_{y_0} &= r_0 \cos(\theta_n) - y_0 \sin(\theta_n) \tan(\theta_n + d\theta/2) \tag{18}\\
r^t_{y_1} &= r_1 \cos(\theta_n) + y_1 \sin(\theta_n) \tan(\theta_n - d\theta/2) \tag{19}\\
r^b_{y_1} &= r_1 \cos(\theta_n) - y_1 \sin(\theta_n) \tan(\theta_n + d\theta/2) \tag{20}\\
r^t_{y_2} &= r_2 \cos(\theta_n) + y_2 \sin(\theta_n) \tan(\theta_n - d\theta/2) \tag{21}\\
r^b_{y_2} &= r_2 \cos(\theta_n) - y_2 \sin(\theta_n) \tan(\theta_n + d\theta/2) \tag{22}\\
& \tag{23}
\end{align}$$

The Euler rotation routine exists in dd4hep but does not in fact follow the Euler rotation convention. Instead, the RotationZ class must be imported from ROOT to achieve a proper Euler rotation:

```
$\phi_m = m\,d\phi$
RotationZYX eulerRotation($\pi/2$, $\theta_n$, 0);
ROOT::Math::RotationZ rotZ = ROOT::Math::RotationZ($\phi_m$);
eulerRotation = rotZ*eulerRotation;
```

Finally,

$$r_t = r_{\text{trapezoid}} = r_0 + (F_{\text{dz}} + R_{\text{dz}})/2 \tag{24}$$

$$r_{\text{displacement from origin}} = (r_t \sin(\theta_n)\cos(\phi_m), r_t\ \sin(\theta_n)\sin(\phi_m), r_t\cos(\theta_n)) \tag{25}$$

$$\tag{26}$$

with placement inside an assembly volume:

```
experimentalHall.placeVolume(
    crystalTowerAssemblyVol,
    Transform3D(eulerRotation,r_displacement from origin)
);
```

8.3 DD4HEP geometry validation

- First plots obtained with photon particle gun:
  - Hit multiplicity vs photon energy as a function of hit energy threshold
  - Reconstructed energy vs eta and vs phi
  - Energy resolution vs photon energy (including artificial smearing)
  - Spatial/angular resolution

8.4 Towards reconstruction

The path towards reconstructing (clustering strategy).

8.4.1 Simulation of optical part

Some comments on why this is not critical/useful to implement now.

9 Pre-Shower and Muon detector

10 Performance plots

Here of in the various sub-sections...

11 Conclusions

Acknowledgments. Acknowledgments are not compulsory. Where included they should be brief. Grant or contribution numbers may be acknowledged.

Please refer to Journal-level guidance for any specific requirements.

Appendix A Beam pipe simulation

References

[1] M. Boscolo, F. Palla, F. Bosi, F. Fransesini, S. Lauciani, Mechanical model for the FCC-ee interaction region. EPJ Techniques and Instrumentation **10**(1)

(2023). https://doi.org/10.1140/epjti/s40485-023-00103-7. URL https://doi.org/10.1140/epjti/s40485-023-00103-7

[2] M. Boscolo, F. Palla, A. Abramov, M. Aleksa, K.D.J. Andre, f. bosi, G. Broggi, A. Ciarma, M.D. Da Rocha Rolo, B. Francois, F. Fransesini, A. Gaddi, A. Ilg, S. Lauciani, D. Mergelkhul, A. Novokhatski, E. Perez, L. Reichenbach, A. Sailer, L. Watrelot, The fcc-ee interaction region, design and integration of the machine elements and detectors, machine induced backgrounds and key performance indicators (2023). https://doi.org/10.17181/W4KWS-RNE05. URL https://new-cds.cern.ch/doi/10.17181/w4kws-rne05

[3] L. Pancheri, R.A. Giampaolo, A.D. Salvo, S. Mattiazzo, T. Corradino, P. Giubilato, R. Santoro, M. Caccia, G. Margutti, J.E. Olave, M. Rolo, A. Rivetti, Fully depleted maps in 110-nm cmos process with 100–300-μm active substrate. IEEE Transactions on Electron Devices **67**(6), 2393–2399 (2020). https://doi.org/10.1109/ted.2020.2985639. URL http://dx.doi.org/10.1109/TED.2020.2985639

[4] I. Peric, A. Andreazza, H. Augustin, M. Barbero, M. Benoit, R. Casanova, F. Ehrler, G. Iacobucci, R. Leys, A.M. Gonzalez, P. Pangaud, M. Prathapan, R. Schimassek, A. Schoning, E.V. Figueras, A. Weber, M. Weber, W. Wong, H. Zhang, High-voltage cmos active pixel sensor. IEEE Journal of Solid-State Circuits **56**(8), 2488–2502 (2021). https://doi.org/10.1109/jssc.2021.3061760. URL http://dx.doi.org/10.1109/JSSC.2021.3061760

[5] G. Cataldi, F. Grancagnolo, S. Spagnolo, Cluster counting in helium based gas mixtures. Nuclear Instruments and Methods in Physics Research **386**(2), 458–469 (1997). https://doi.org/https://doi.org/10.1016/S0168-9002(96)01164-3

[6] C. Lippmann, Particle identification. Nuclear Instruments and Methods in Physics Research **666**, 148–172 (2012). https://doi.org/https://doi.org/10.1016/j.nima.2011.03.009

[7] F. Cuna, N. De Filippis, F. Grancagnolo, G. Tassielli, Simulation of particle identification with the cluster counting technique (2021)

[8] T. Jones. Cepc silicon /lhcb mt tile (2020). https://indico.ph.ed.ac.uk/event/65/contributions/814/Presentation at the First UK workshop on HV-CMOS technology for future e+e- colliders, University of Edinburgh

[9] H. Zhu. A large tracking system with novel HV-CMOS sensors for the CEPC (2021). https://indico.inp.nsk.su/event/42/contributions/2186/attachments/1355/1777/CEPC_Silicon_Tracker_AFAD.pdfPresentation at the Asian Forum for Accelerators and Detectors (AFAD), BINP

[10] M. Tornago. Detector optimization and physics performance of the CMS Phase-2 Endcap Timing Layer (2023). URL https://cds.cern.ch/record/2848200. Presented 13 Feb 2023