3D voxel direct reconstruction

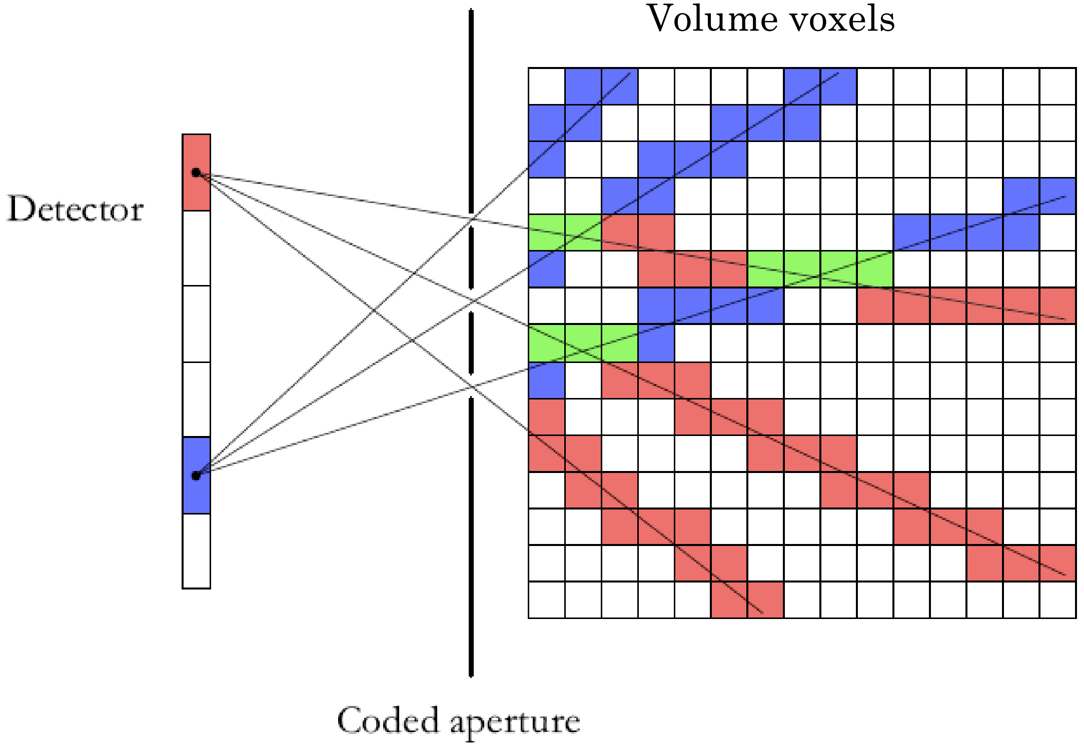
2021/11/11

Valentina Cicero

on behalf of the “LAr Optical System working group”

Core concept

- We want to develop a reconstruction algorithm that can be employed in conjunction with Coded Aperture Imaging techniques, but directly reconstruct a **3D image**
- **Combinatorial** approach: for each detected photon, project a probability amplitude in the segmented reconstruction volume through all possible mask holes



Core concept

Pros:

- This method can employ any multiple pinhole mask (but some masks patterns may be more effective than others).
- Timing can provide additional information for reconstruction.

Cons:

- Computationally heavy, but most part depends only on simulation geometry. It can be precomputed once and stored.

Software written in Python and openCL for GPU acceleration.

Now running on CNAF HPC node equipped with 80 cores CPU and 4 Nvidia Tesla-V100

Voxel probability amplitude

Reconstruction volume is segmented in voxels (*i,j,k* index)

A camera (*c* index) consists of a SiPM matrix (SiPM index *s*) and a coded aperture mask.

Voxel probability amplitude:

$$A(i,j,k) = \sum_c \frac{\sum_s H(c,s) \cdot w(c,s,i,j,k)}{w_{norm}(c,i,j,k)}$$

- ***H(c,s)*** *:* n photons detected by sensor pixel *s* of camera *c*
- ***w*** weight factor computed as Bayesian probability of (*c,s*) detecting a photon that originated in (*i,j,k*)
- $\boldsymbol{w_{norm}} = \sum_s w(c,s,i,j,k)$ weight normalization factor per voxel

Weight

Weight w is computed as the Bayesian probability that a photon that is detected by sensor (c,s) originated from voxel (i,j,k)

Events:
vox = photons originated in voxel (i,j,k)
sens = photon is detected by sensor (c,s)

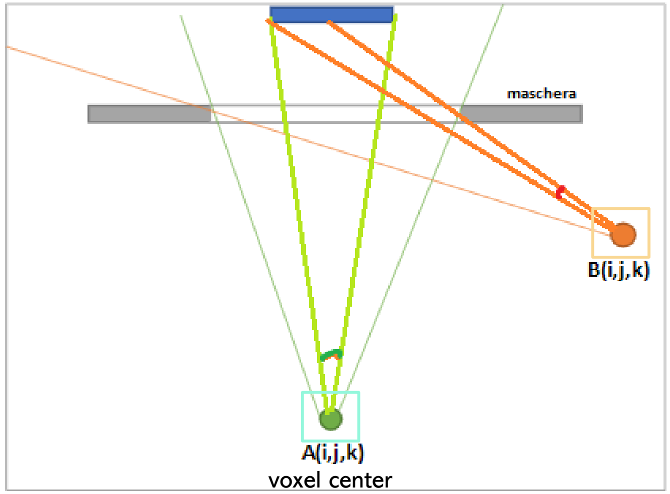
$$w = P(vox|sens) = \frac{P(sens|vox) \cdot P(vox)}{P(sens)}$$

P(sens | vox) = P geometrical acceptance ·
· P attenuation · P detection · (…)

P(vox) = uniform (prior) → simulated particle tracks can cross any voxel in the scintillating medium volume

P(sens) = $\sum_{i,j,k} P(sens|vox)$ → law of total probability

P geometrical acceptance



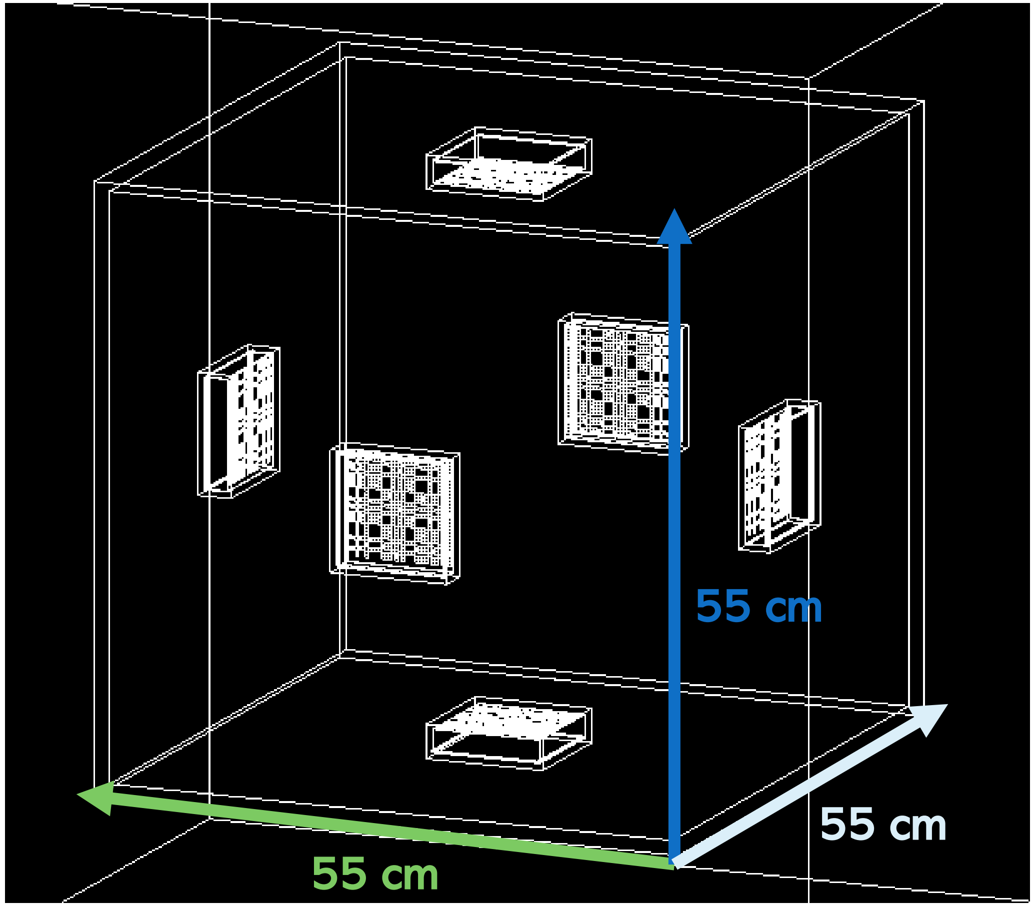
Simulation: Cold demonstrator

- 55x55x55 cm LAr volume
- 6 cameras

- SiPM Matrix 16x16
- Mask rank 17
- SiPM active area 3x3mm^2

- SiPM full electrical response simulated (pde ~20%, small DCR noise and crosstalk)

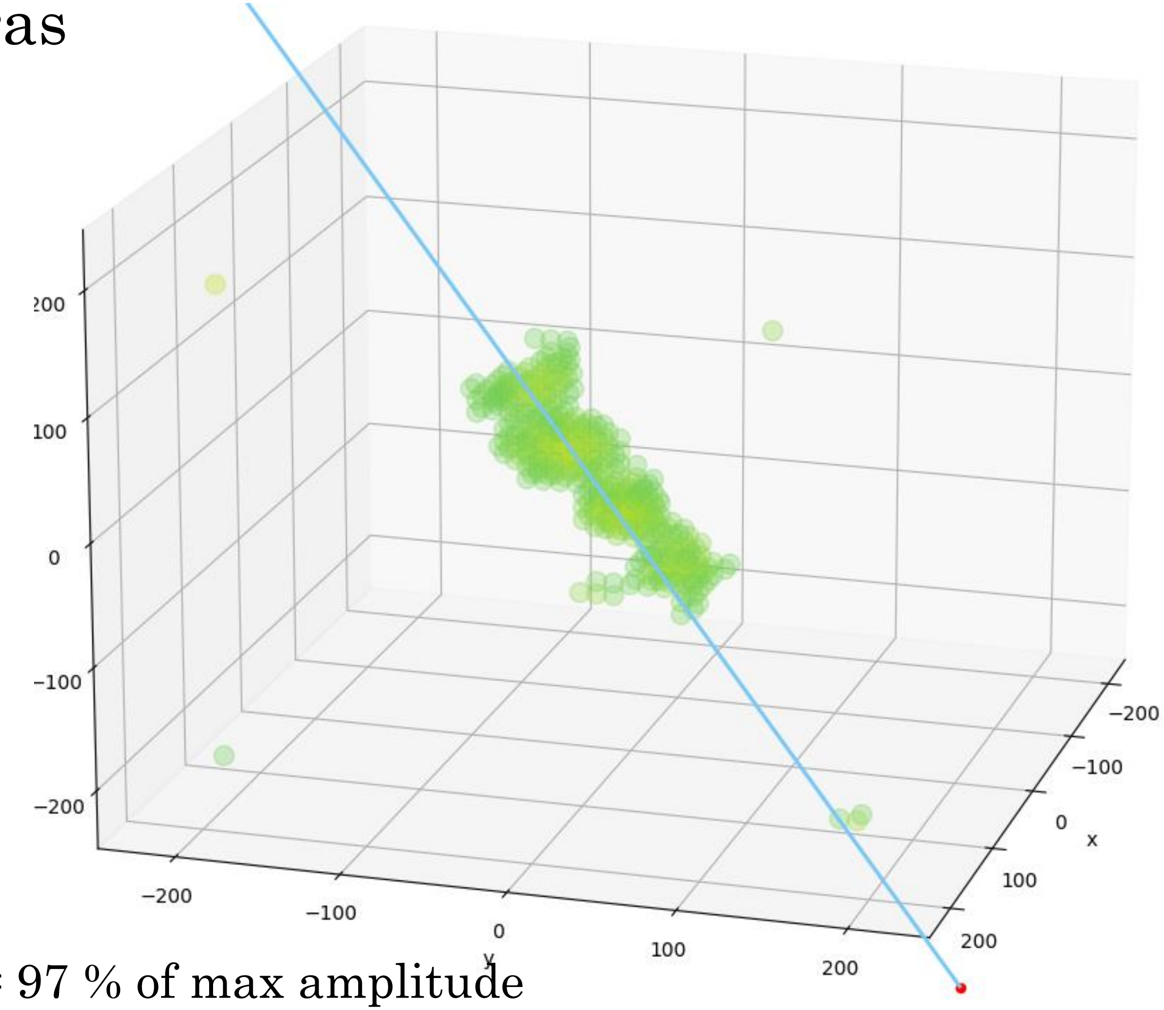
- DAQ simulation: ADC, TDC



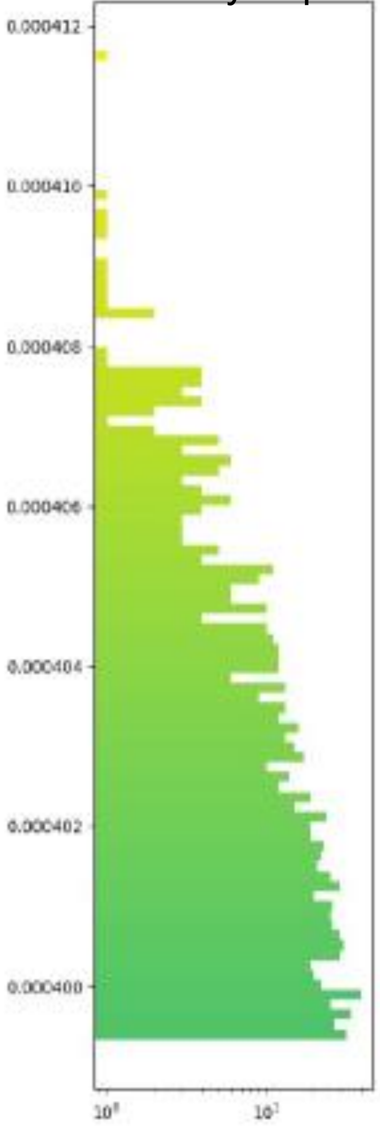
Simulated Muon Tracks

3 GeV diagonal muons

6 cameras



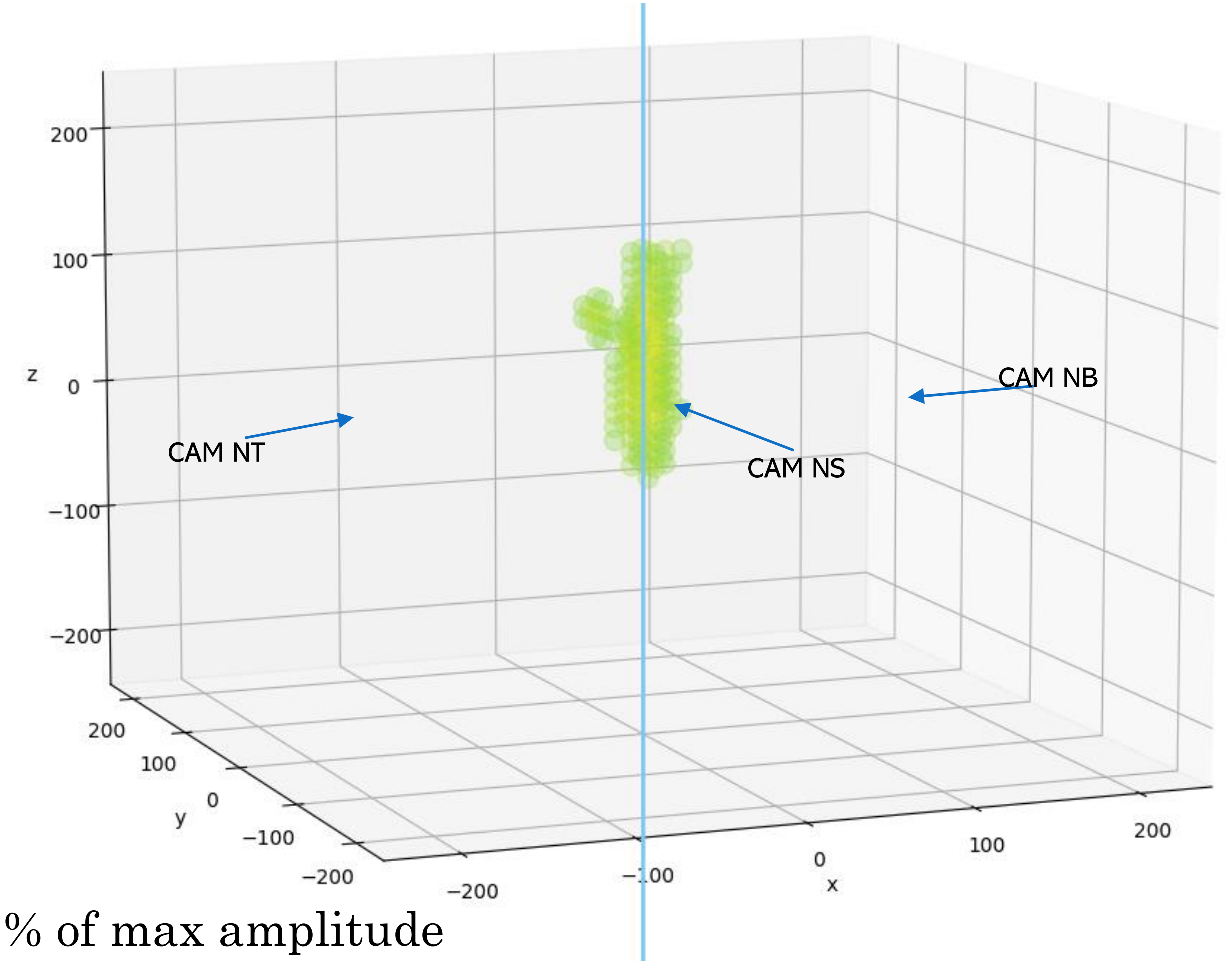
Cut = 97 % of max amplitude

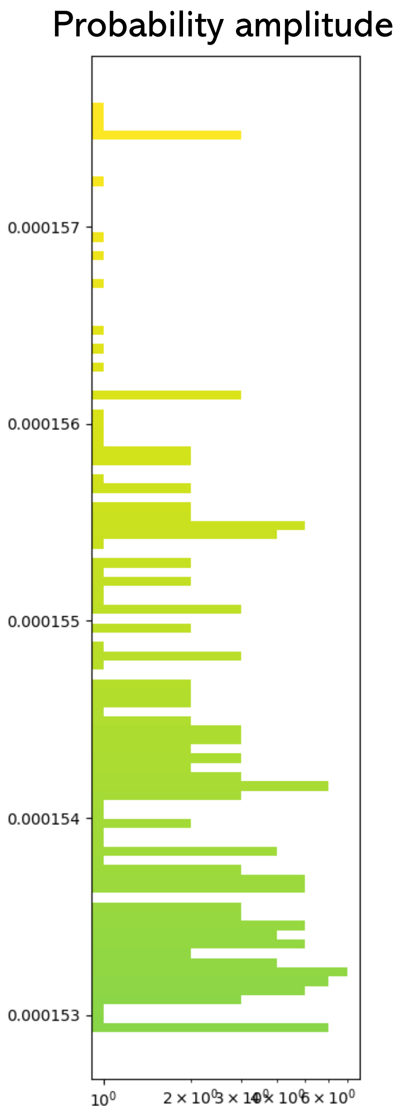


Simulated Muon Tracks

3 GeV vertical muons

3 cameras



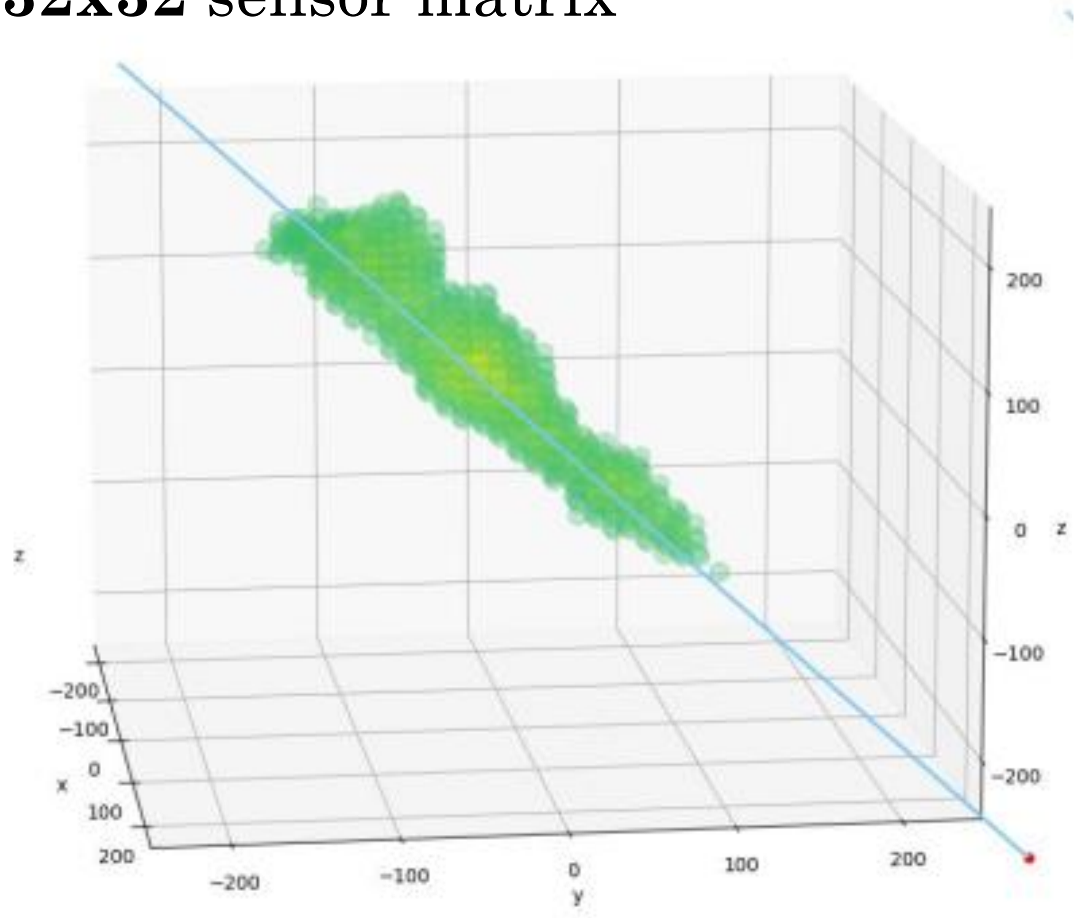


Cut = 97 % of max amplitude

Simulated Muon Tracks

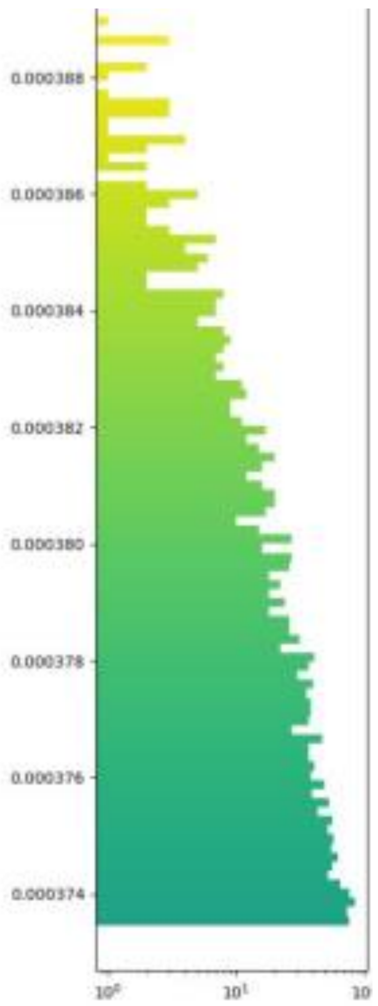
3 GeV diagonal muons

6 cameras, **32x32** sensor matrix



Cut = 97 % of max amplitude

Probability amplitude



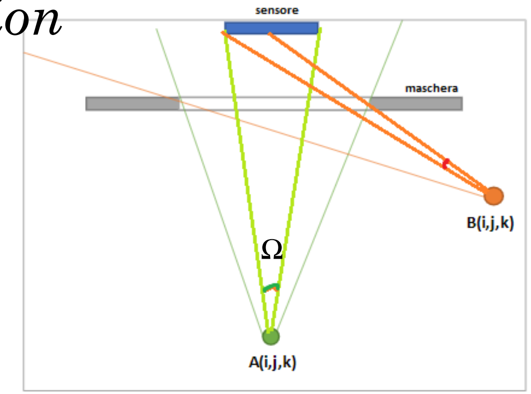
Next steps

- Optimize demonstrator geometry:
  - Demonstrator size, cameras field of view and position
- Simulation of cosmic ray events
- 3D tracks clustering and fitting
- Move to GRAIN geometry

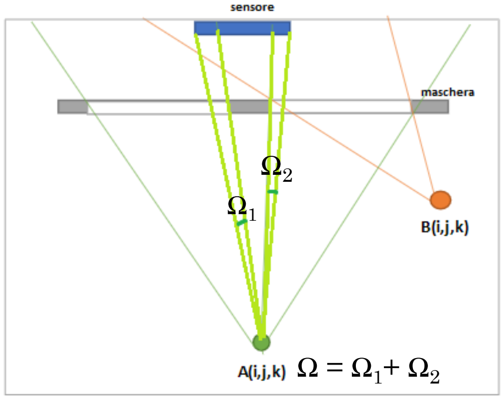
Backup

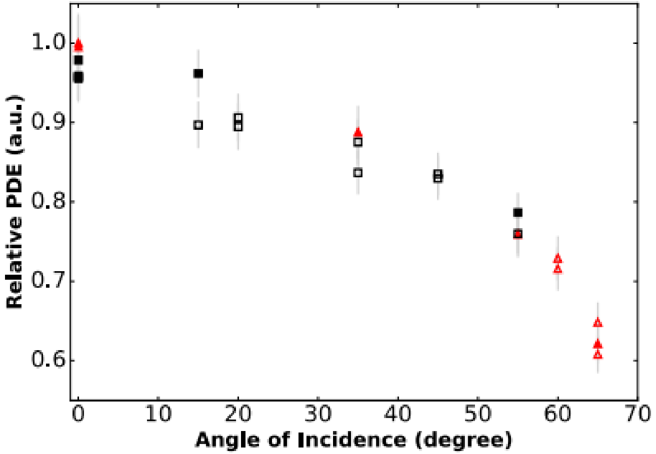
$P(sens \mid vox)$

$P(sens \mid vox) = P\ geometrical\ acceptance \cdot P\ attenuation \cdot P\ detection$



- $P\ geom.\ acc. = \Omega\ /\ 4\ \pi$ oppure $A_{visibile}/A_{sens}$
  - Ω : angolo solido sotteso dall'area del sensore (c,s) che si vede attraverso i fori della maschera dal centro del voxel (i,j,k)
  - A : area del sensore (c,s) che si vede attraverso i fori della maschera dal voxel (i,j,k)
- $P\ attenuation = e^{-d/\lambda\ att}$
  - d: distanza(centro sensore, centro voxel)
  - $\lambda\ att$: lunghezza di attenuazione
- *P detection* → la PDE di un SiPM dipende anche dall'angolo di incidenza del fotone (non ancora implementato nella simulazione)





Nakarmi P. et al. *"Reflectivity and PDE of VUV4 Hamamatsu SiPMs in* ***liquid xenon****"*
https://arxiv.org/pdf/1910.06438.pdf