NATIONAL RESEARCH UNIVERSITY

LAMBDA • HSE

Способы обработки нерегулярностей в симуляционных данных на коллайдерных экспериментах

The use of new methods for processing data of a physical experiment.
Application of machine learning methods on the NICA complex. 28-29 Aug 2023

Alexey Boldyrev[1], Denis Derkach[1],
Fedor Ratnikov[1,2], Andrey Shevelev[1]
1 — HSE University (Laboratory of Methods for Big Data Analysis); 2 — Yandex School of Data Analysis;

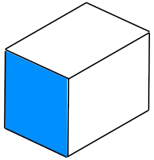
See also:
[1] JINST 15 C05032 (2020)
[2] JINST 15 C09030 (2020)
[3] EPJ Web of Conferences 245, 02019 (2020)
[4] J. Phys.: Conf. Ser.1740 012047 (2021)
[5] Reviews in Phys. 10 100085 (2023)

Challenges for detector sim+reco

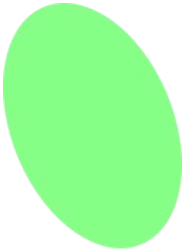
- R&D of HEP detector requires a lot of simulations
- Level of detail of a simulation may vary:
  - Toy model
    - Sketching
  - (Intermediate options)
  - Detailed modelling
    - Describes engineering constraints and detector alignment
    - Real data imitation

- Desired figure of merit of a simulation can be achieved by a reconstruction similar to that used in physical analysis

Simplest case of cells/pixels arrangement

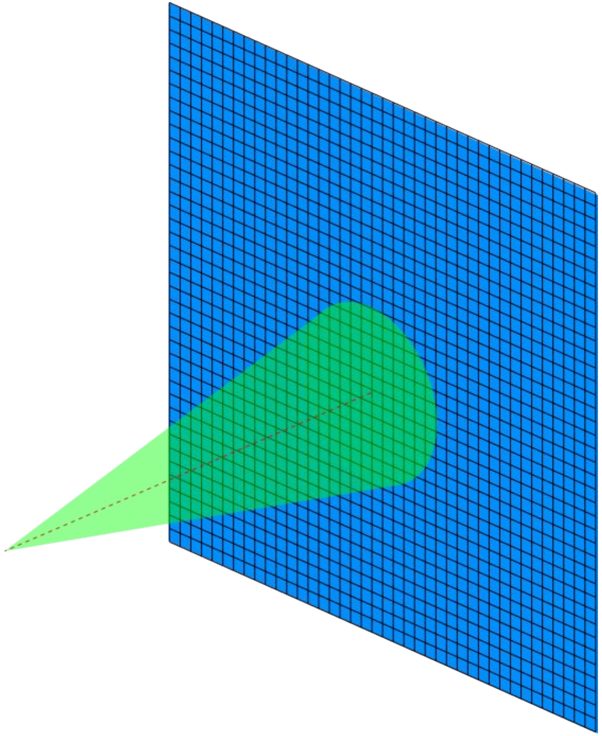


Sensitive element



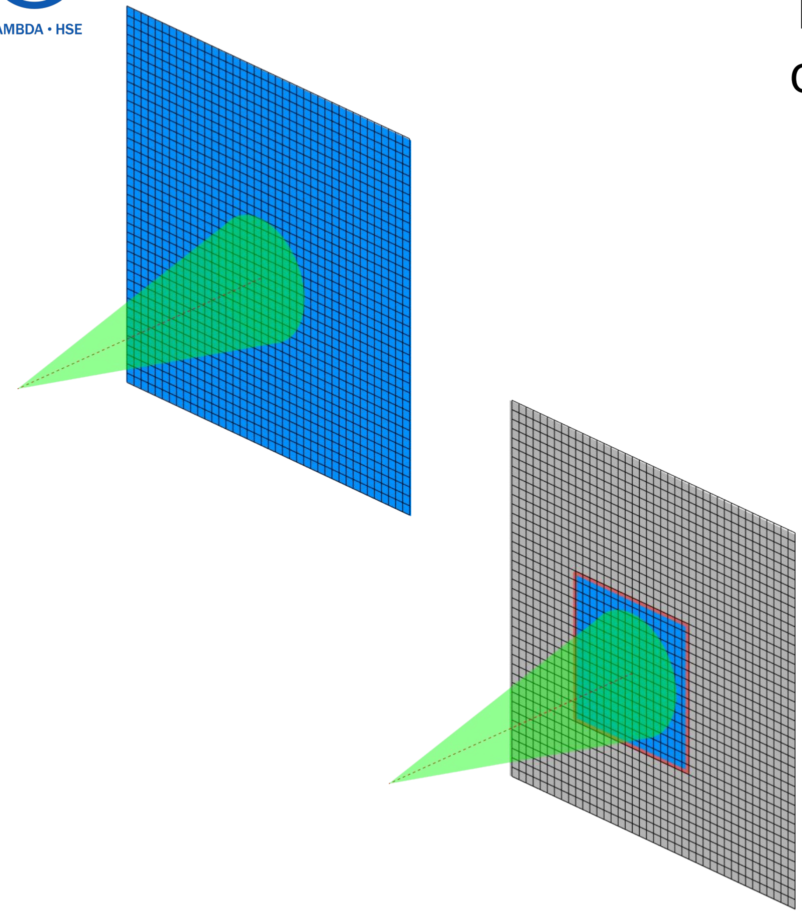
Typical area
of a response

Desired η coverage



(planar case)

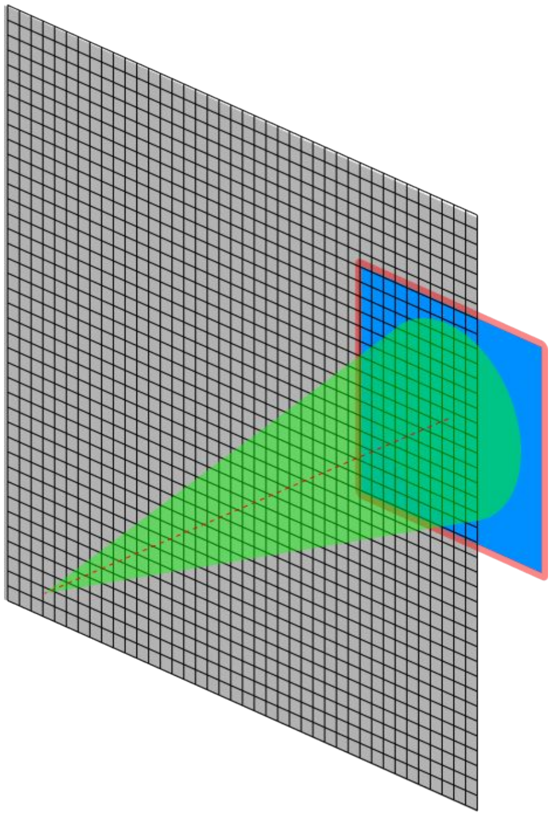
Simplest case processing



Two scenarios for the use of simulation outputs by following (ML) reconstruction:

- **All available channels at once**
  - Computation limitations (typically, $10^3 \ldots 10^6$ channels)
- **Scanning window**
  - Cells/pixels are selected only around fired sensitive elements
  - Rely on the seed finder algorithm
  - Window size should by larger than typical area of a response

Simplest case: scanning window



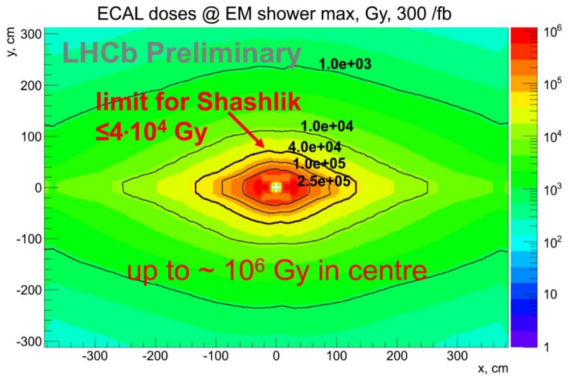
Possible pitfall

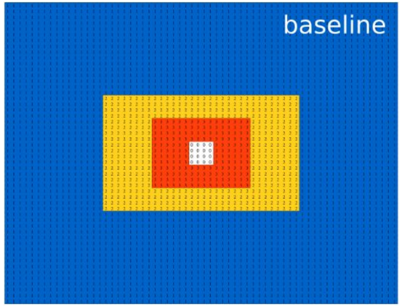
- The scannig window extends beyond the detector boundary

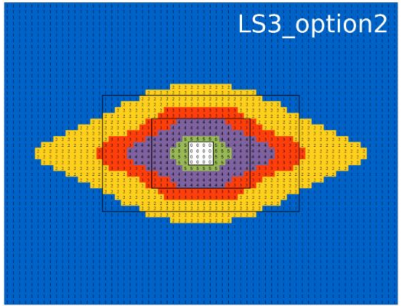
More realistic cases of pixels/cells arrangement

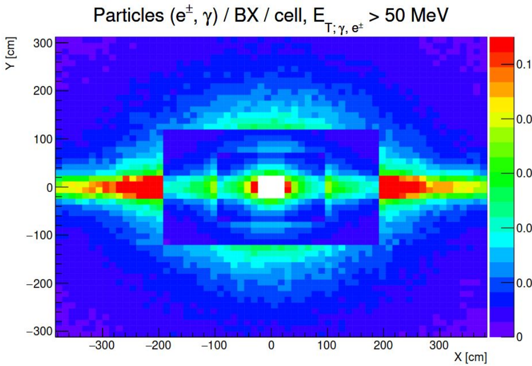
LAMBDA • HSE

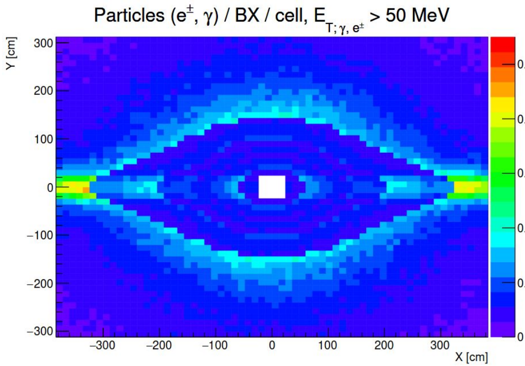
Driven by expected radiation doses or occupancy maps











It brings us regions with:

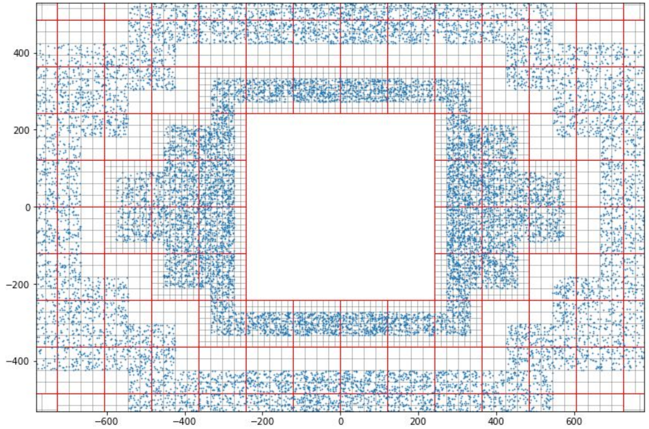
- Different technologies
- Different granularities

More realistic cases of pixels/cells arrangement

More realistic cases imply **additional boundaries** between regions of different granularities (technologies).

Naïve strategy: avoid irregularities of the geometry

- For 5x5 cells scanning window and ‘romboidal’ shape of the regions we can lose ~20% of reconstructible events

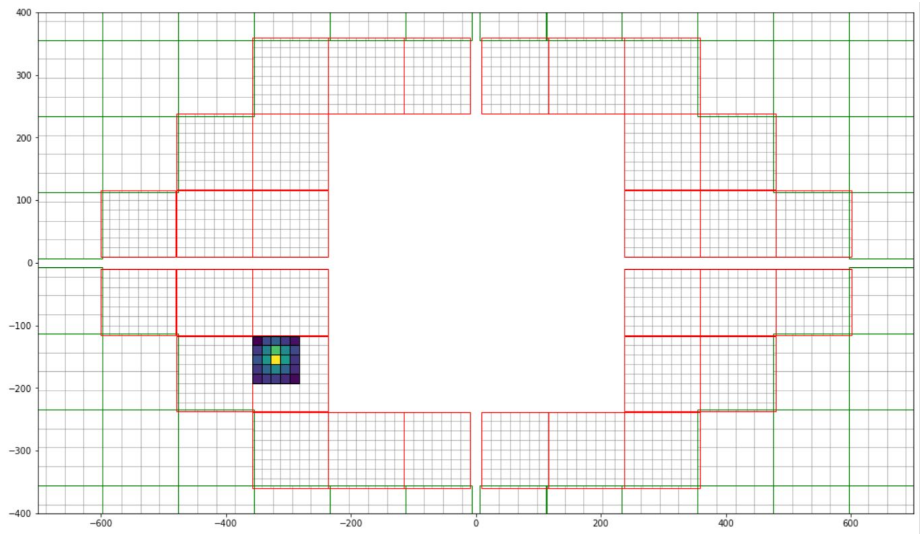


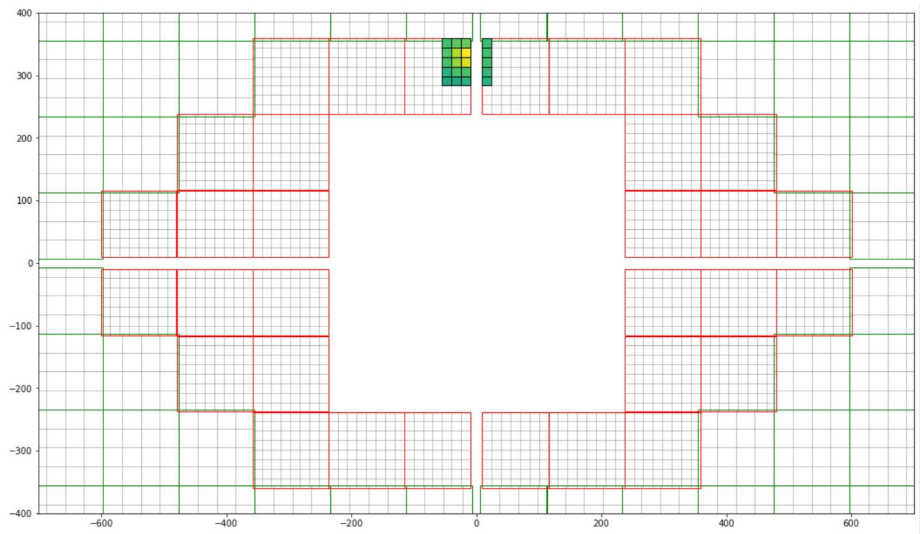
Better strategy: interpolation of the cells for equalization of granularity on both sides of the border

- We can use all reconstructible events

More realistic cases of pixels/cells arrangement

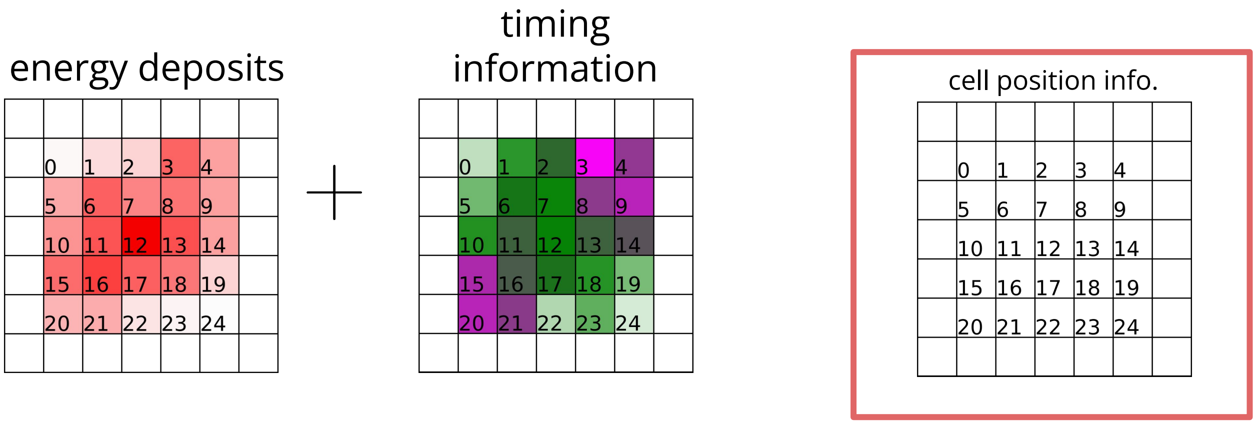
More realistic cases also involve engineering gaps or infrastructure objects.





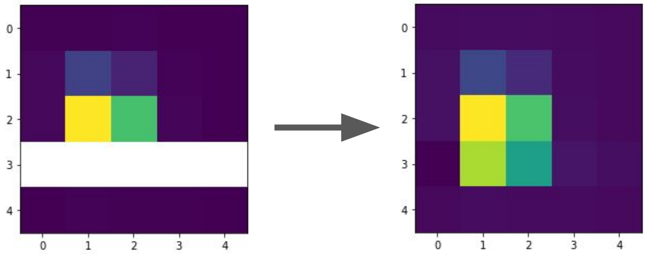
Naïve strategy also leads to additional inefficiency close to such gaps or objects.

Strategies to have geometry agnostic inputs



Better strategy:

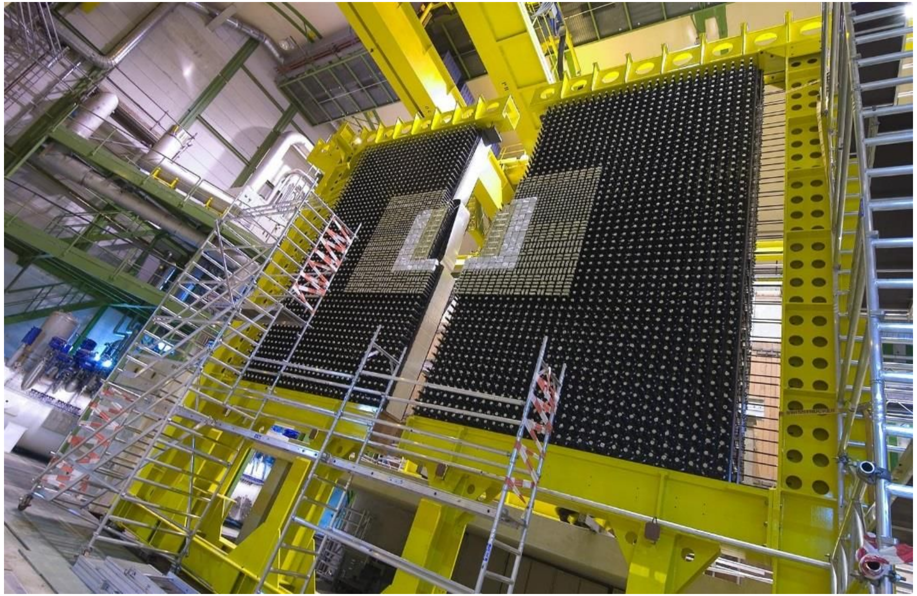
- Cell position matrix as addition input to ML regressors
- Interpolation of non-existing cells outside of the outer borders
- Interpolation of cells for equalization of granularity on both sides of the boundaries between



LHCb ECAL case

LHCb ECAL

Current configuration

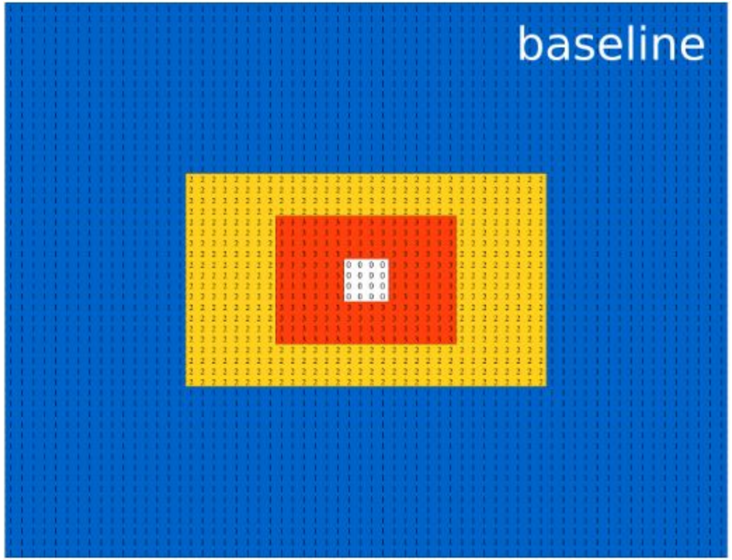


Calo modules of size 12x12 cm^2

Wall dims: 7.8x6.3x0.5 m^3

176 inner: 9 cells with size 4x4 cm^2
448 middle: 4 cells with size 6x6 cm^2
2688 outer: 1 cell with size 12x12 cm^2

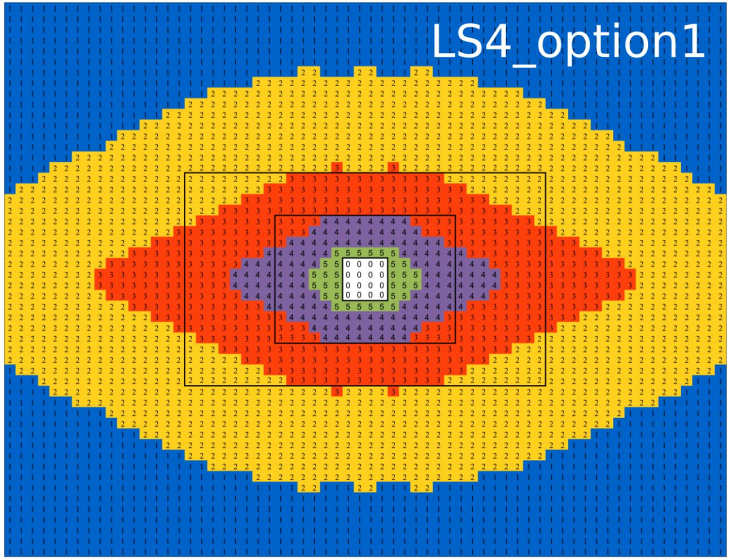
LHCb ECAL



Module type	# of modules
(inner): 3x3 cells (4.04x4.04 cm^2 each)	176 (1536 ch.)
(middle): 2x2 cells (6.06x6.06 cm^2 each)	448 (1792 ch.)
(outer): single cell (12.12x12.12 cm^2)	2688 (2688 ch.)

Starting from current configuration

Future LHCb ECAL



Reuse of current "Shashlik" modules

- 1: Outer region, cell size = 12.12×12.12 cm^2
- 2: Middle region, cell size = 6.06×6.06 cm^2
- 3: Inner region, cell size = 4.04×4.04 cm^2

New "SpaCal" modules

- 4: cell size = 3.03×3.03 cm^2
- 5: cell size = 1.515×1.515 cm^2

(+ longitudinal split)

Questions for future ECAL:

- What is the best configuration for given modules (fix cost) in terms of given physics metric?
- What is the best way to arrange a certain number of new modules?

Towards differentiable Calo framework

LAMBDA • HSE

Input	Differentiable chain	Output

Input:
- MC Truth
- Signal
- Background

Detector simulation (+ digitization)		Detector reconstruction + digitization	Physics reconstruction
Particle transport	Geant4 Calo simulation		
	Optimized parameters: Granularity, Molière radius, Timing	Energy, Spatial and Timing resolutions	Signal yield, Significance + their derivatives
Yes	Hardly	Yes	Yes
	Local surrogate modelling using CaloGAN		

Is differentiable? (row above: Yes / Hardly / Yes / Yes)

Output:

Label:
- 1 event accepted
- 0 event rejected

Possible inputs for Calo framework

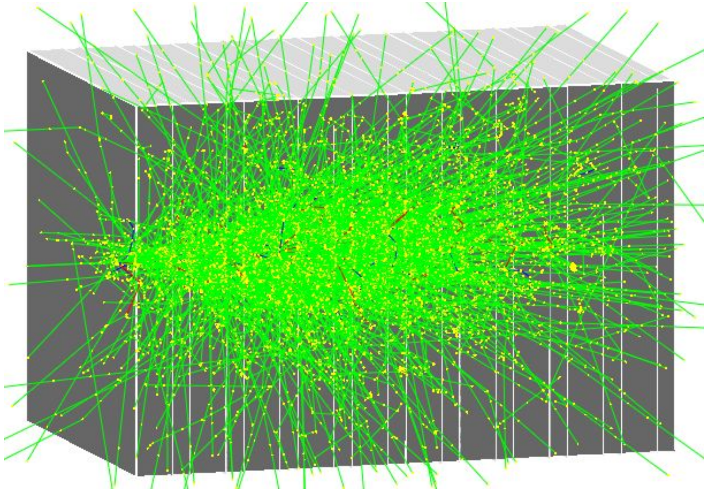
- Simulation input:
    - Particle gun (single photons)
    - Pythia with reference physics sample like $B_s^0 \to J/\psi(\to \mu^+\mu^-)\pi^0(\to \gamma\gamma)$
        - Background sample(s)
            - Minimum Bias
                - Arbitrary pile-up modelling due to PV extraction

- ML-based reconstruction based on 3 sets of regressors to estimate:
    - Position
    - Energy
    - Time

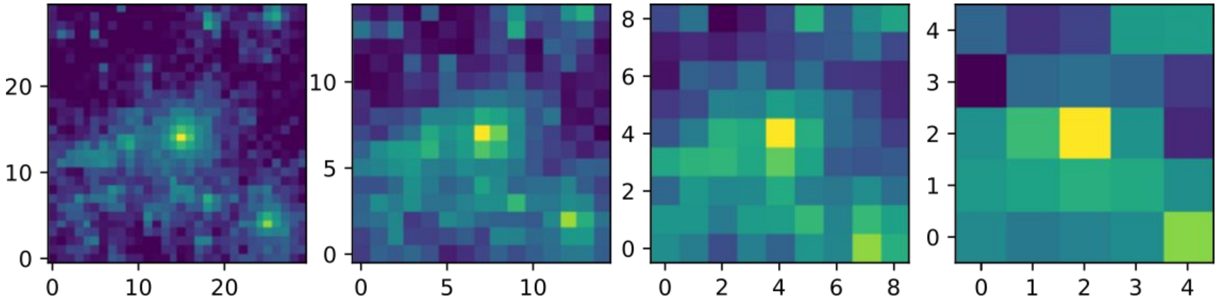
Challenges for new ECAL configurations

- Thousands of configurations are possible within a budget
- For each configuration one should decide:
  - Module technology options (Shashlik/SpaCal/...)
  - Granularity (cell size)
  - Longitudinal segmentation
  - Timing information
- How to factorize the above?

Geant4 simulation for granularity study

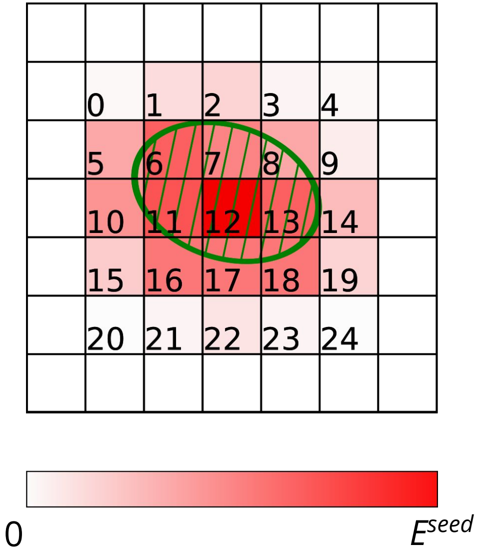


- Sampling structure with alternating scintillating tiles and lead plates
  - Roughly emulates LHCb ECAL
- {x, y, z, t} of all hits in the ECAL are recorded
- Can represent arbitrary (regular) granularity of the ECAL cells



Input features for Reconstruction

Signal energy deposits and shower spot



Simulated Geant4 response is an array of cells

- Used as base features for the regressors on Energy and Position
- Regressor on time uses weighted energy deposits

What we have so far

- Single ECAL module:
  - ML reco performance is compatible with conventional Reco performance using detailed simulation input (& with beam tests)

- Full ECAL
  - Requires geometrical irregularities
    - There are 4 borders between the regions of different granularity
    - Some modules have to be rotated due to technology limitations

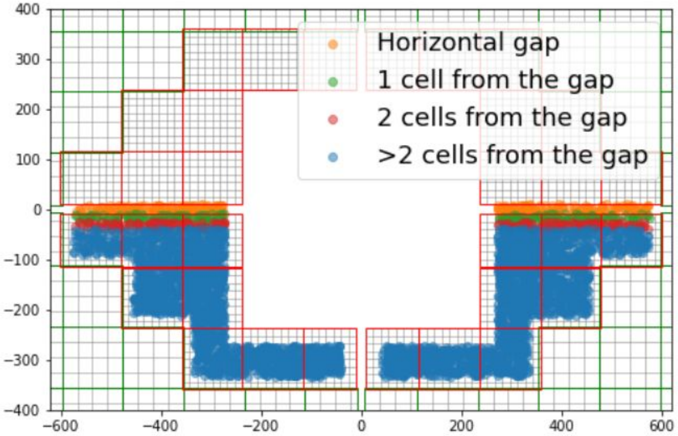
How does that fit in with the fact that reco algorithm needs to be geometry agnostic?

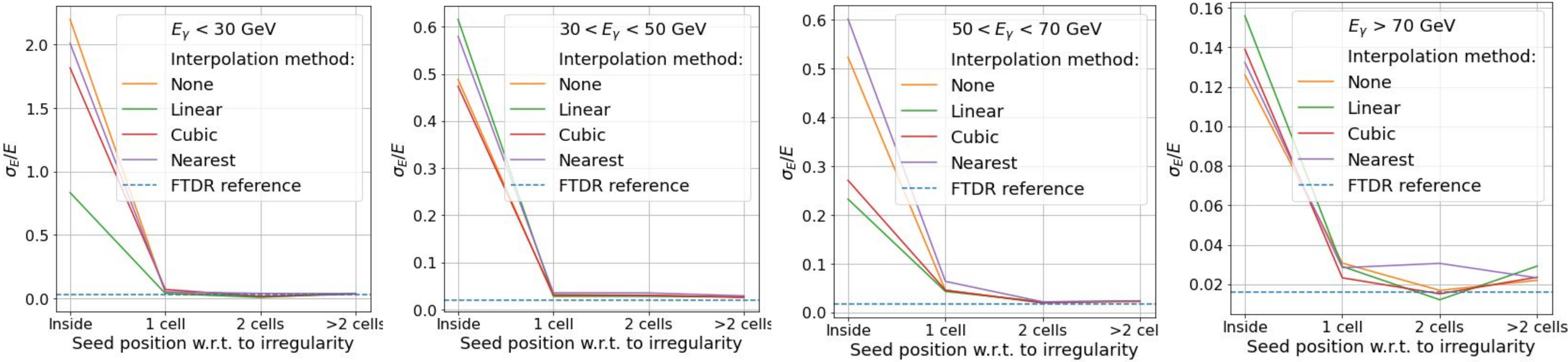
Energy reconstruction on geometry agnostic inputs

LAMBDA • HSE

XGB-based ML regressor
- 5x5 matrix of energy deposits
  - Missing cells recovered using

Linear, Cubic and Nearest-neighbor interpolation
- Cell position info
- Additional features



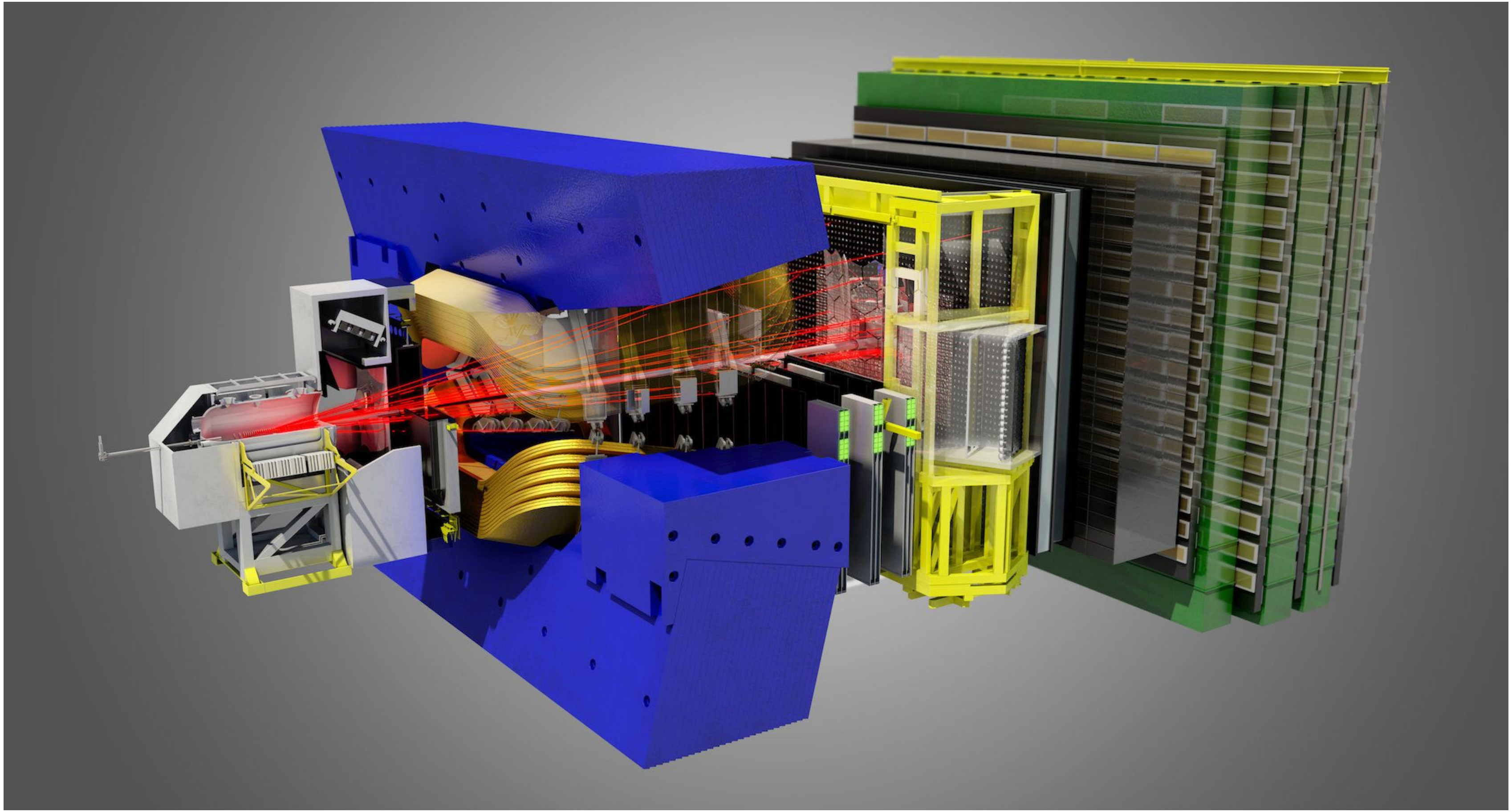


Conclusions

- The R&D process requires time consuming computation steps to evaluate physics performance for different detector techniques and configurations.
- ML reco is consistent with conventional reconstruction for single ECAL module and regular geometry
- ML reco is able to handle detailed simulation inputs using cell position matrix, interpolation of missing cells, and interpolation of low granularity cells close to high granularity cells
- Automatic training speeds up the turnover for the performance studies and ensures consistency and uniformity of obtained results

Backup slides

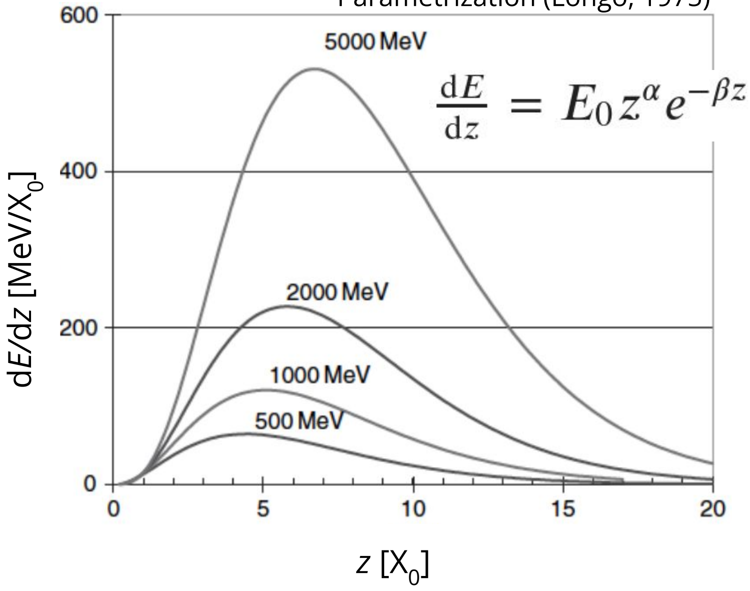
LHCb detector



Calorimetry in a nutshell

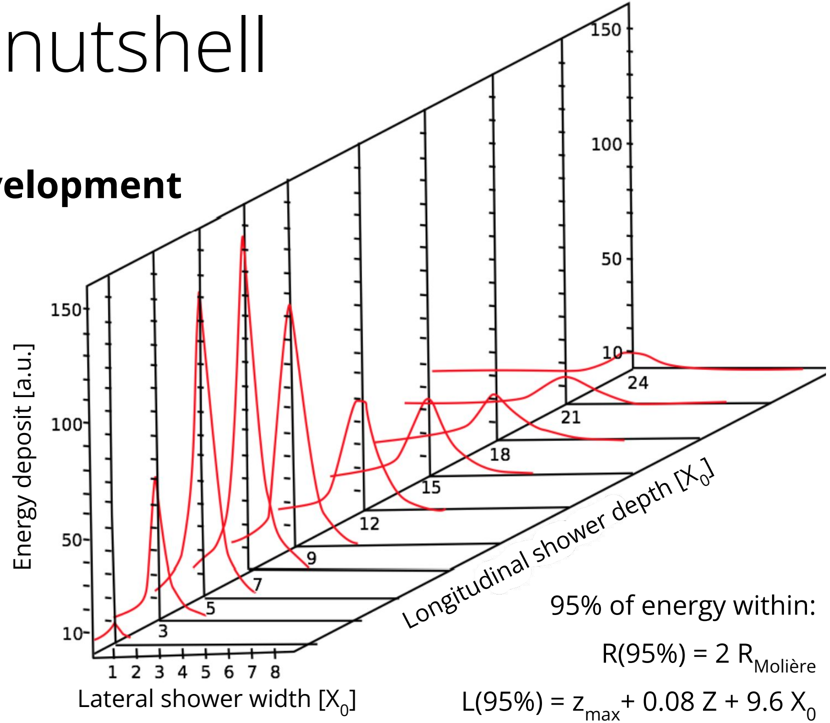
Longitudinal EM shower profile

Parametrization (Longo, 1975)



$$\frac{dE}{dz} = E_0 z^{\alpha} e^{-\beta z}$$

EM Shower development



95% of energy within:

$R(95\%) = 2\ R_{Molière}$

$L(95\%) = z_{max} + 0.08\ Z + 9.6\ X_0$

Differences between showers induced by γ & e

$$z_{max} = \frac{\alpha - 1}{\beta} = \ln\left(\frac{E_0}{E_C}\right) + C$$

$C_\gamma = -0.5$, $C_e = -1.0$

Energy resolution

$$\frac{\sigma_{reco}}{E_{reco}} = \frac{a}{\sqrt{E_{gen}}} \oplus b \oplus \frac{c}{E_{gen}}$$

Timing resolution

$$\sigma_t = A/\sqrt{E} \oplus B$$

Generating responses using GAN (WIP)

- Collect GEANT responses for the calorimeter technology of track parameters in standalone setup
- Train conditional generative model on simulated data
- Use the model to generate response for the given particle