AI-based reconstruction for highly granular calorimeters

Bishnu Karki

University of California, Riverside

Boost 23

05/03/2023

In Collaboration with:
LBNL - B. Nachman, F. Torales-Acosta
LLNL- A. Angerami, R. Soltz, P. Karande A. Sinha
UCR: M. Arratia, K. Barish, S. Paul, R. Milton, S. Vasquez

DOE grant award number DE-SC0022355

Detector optimization

Generators that mimics detector conditions

Reconstruction algorithm to produce physics object

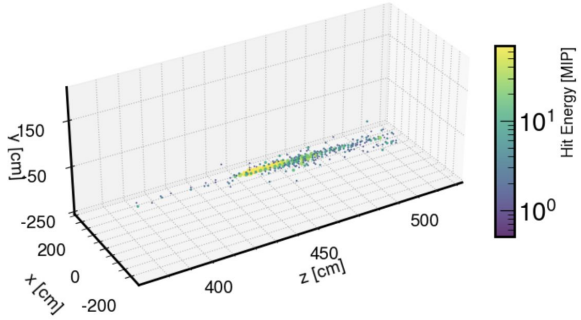
- Traditionally only small number of detector configurations are considered
- Reconstructions algorithm not optimized for a given detector condition
- Allows to have high-fidelity fast simulation and optimized reconstruction algorithms

Our group is working on deploying Artificial Intelligence (AI) methods for EIC hadronic calorimeter design

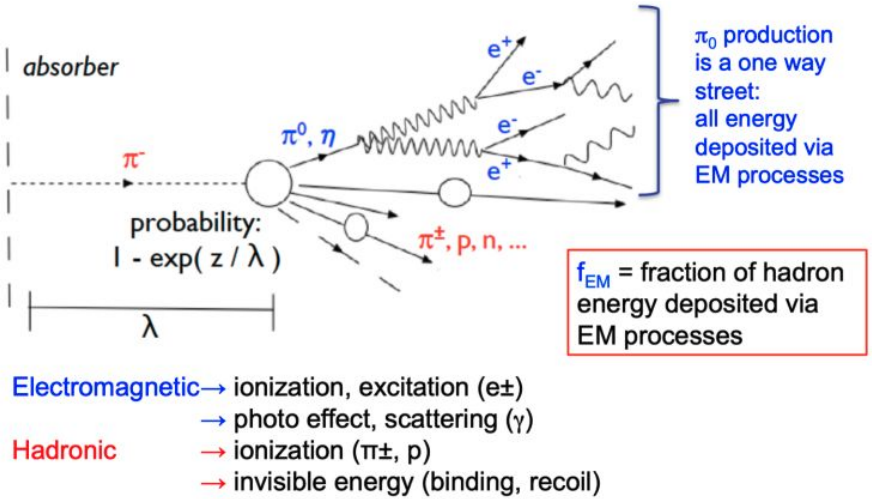
- Generative models (fast simulation) arXiv: 23307.04780
- **Reconstruction algorithm (Regression)**

Outline

- Challenges of non-compensating calorimeters
- “software compensation” for non-compensating calorimeters
  - Traditional Methods
  - AI/ML-based approach
- Impact of longitudinal segmentation and transverse cell information (cell Z, and XY) on model performance



Non-Compensation in Hadronic Calorimeters



Non-compensating calorimeter (e/h ≠1)

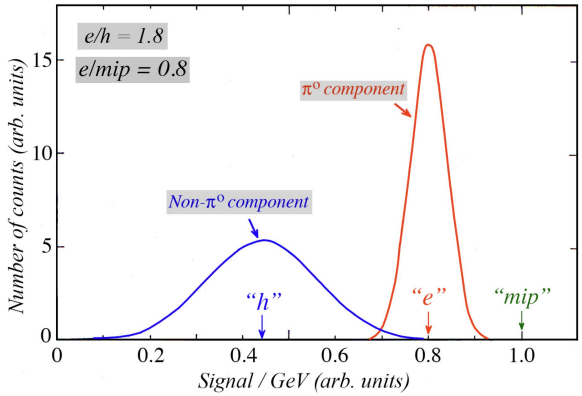


Fig. **arXiv:1710.10535v1**

- Smaller response to hadrons compared to EM particles of the same energy
- Difference in visible signal for EM and purely hadronic energy deposits deteriorates energy resolution

Ways to deal with non-compensation

Hardware compensation

- Imposes very strict requirements on the materials used and the overall geometry. E.g ZEUS Uranium/Sc calorimeter

Software compensation ("offline")

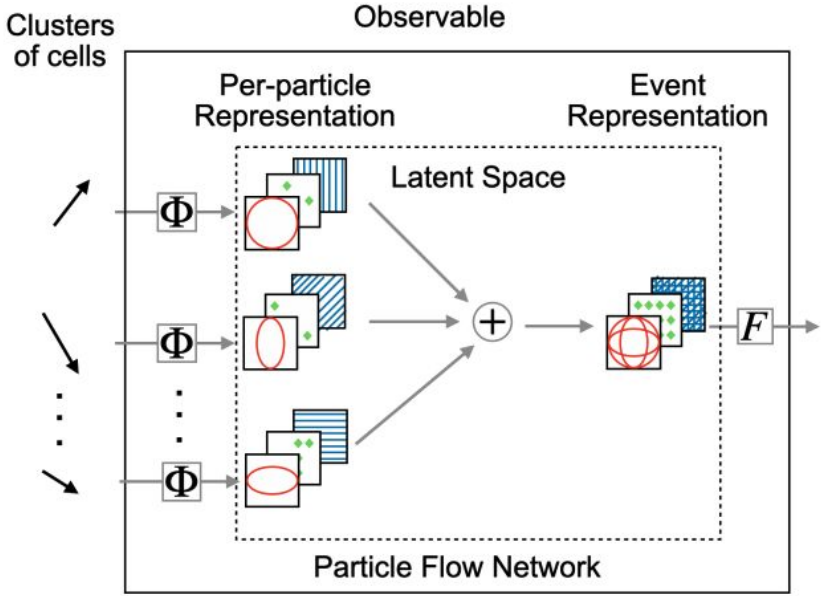
- Assigning weights to EM and HAD energy deposits event by event
- As argued in the YR report, the potential of software compensation motivates longitudinal segmentation in calorimeters

"a fine granularity of a non-compensating calorimeter allows to improve the resolution by assigning weights to the detector signals ("off-line compensation)...at EIC such methods can be considered where a longitudinal segmentation of the ECAL and HCAL readout appears practical".

Deep Sets

JHEP 01 (2019) 121

- Deep sets are designed to operate on sets for permutation-invariant and variable length data
- Set collection of object without any order
- Each particle is mapped by Φ to an internal particle representation (latent space)



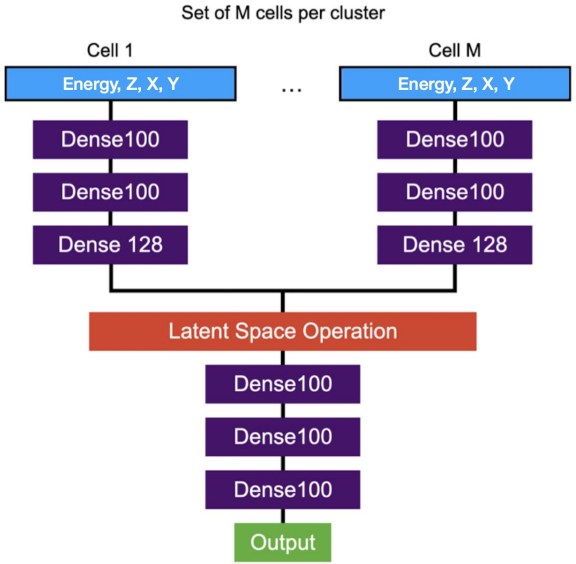
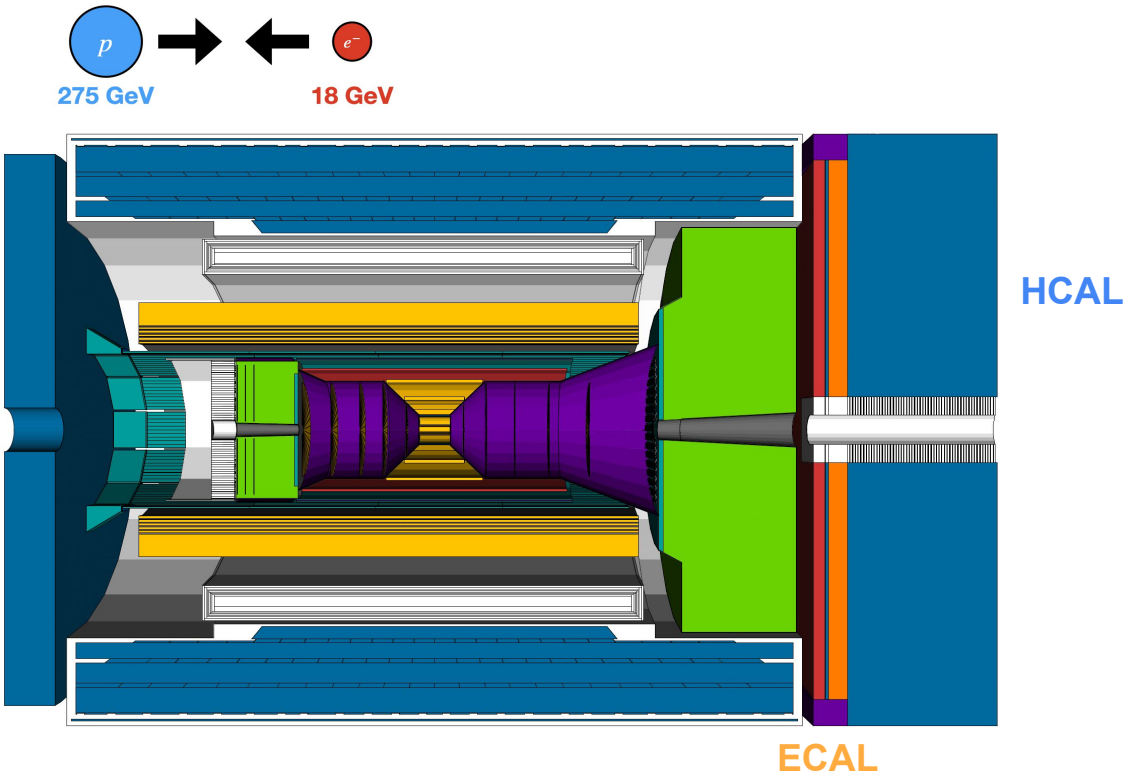


Fig. ATLAS PUB Note

arXiv: 1703.06114

Case Study: Optimization of forward HCAL in ePIC detector



- Proton/ion beam has significantly larger kinetic energy compared to e^- beam
- Most of the hadrons are emitted in the same direction as the hadron beam (“forward direction”)
- Granularity is key component to measure jets

Figure Courtesy ePIC

HCAL and Insert:

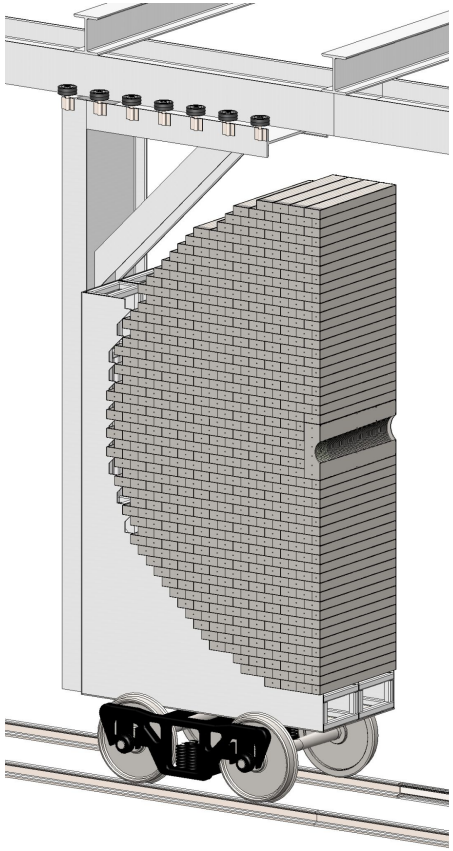
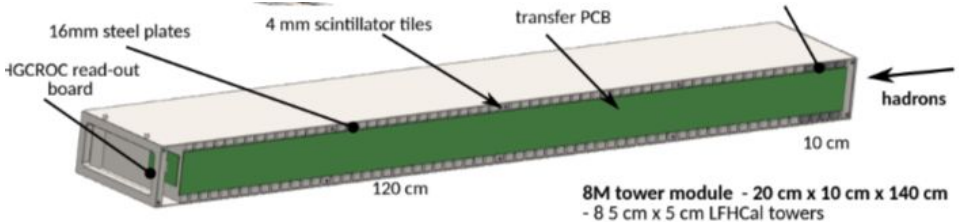
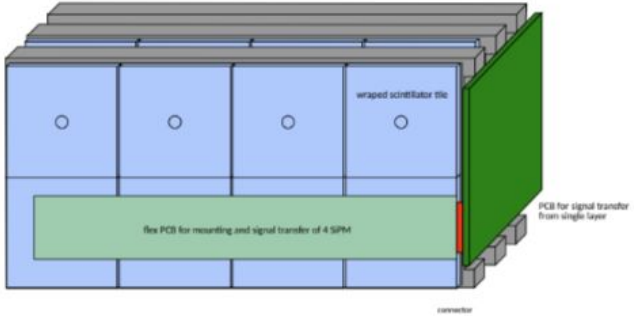


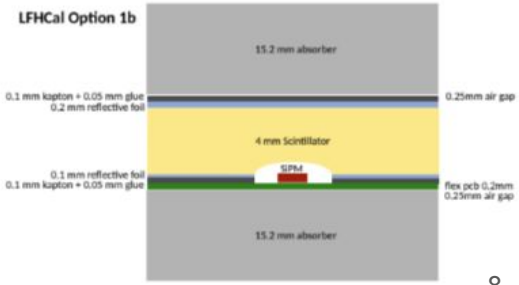
Figure Courtesy ePIC

Optimization Possibility in ePIC

- Technology in ePIC HCAL and Insert uses SiPM-on-tile approach.
- Number of longitudinal sections and their position can be easily changed in practice (summing SiPM pulses) before readout.
- Default is 7 equidistant z-sections regardless of radius.
- Energy density varies with radius, so this is likely non-optimal







Detector Simulation and reconstruction

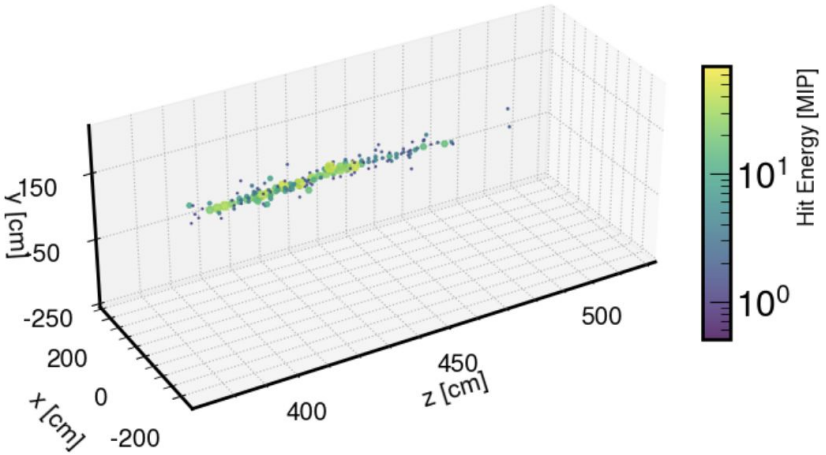
- Using standalone DD4HEP with simplified geometry similar to ePIC HCAL / insert

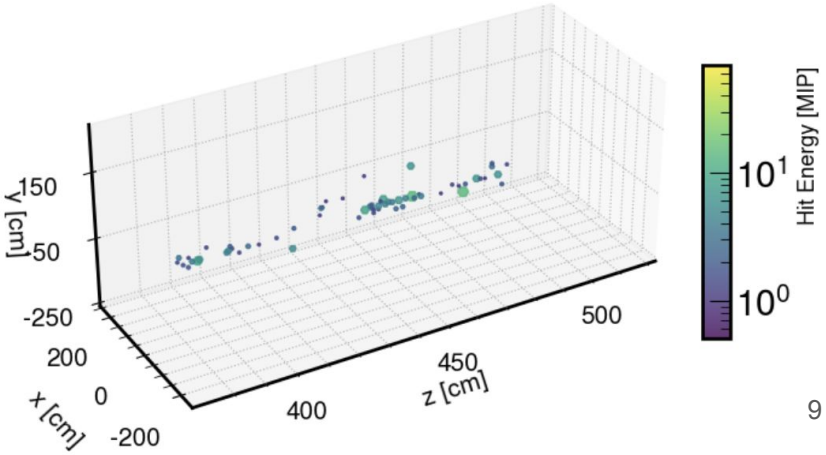
- Single particle Geant4 Simulation
  - Particle: π^+, Polar angle: 10< θ < 30 deg, Azimuthal angle: 0< ϕ< 360
  - Calorimeter Configuration: ECAL in front of HCAL

- Segmentation:
  Longitudinal segmentation: 55 z-sections
  Transverse segmentation: 10 x 10 cm^2 (55 cells)

- Point cloud representations of calorimeter showers

- Established models to predict the generated energy from given cell information
  - With different number of Z- sections
  - With given transverse and longitudinal cell hits (Z, XY)

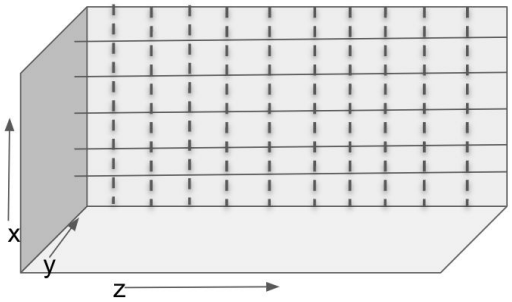


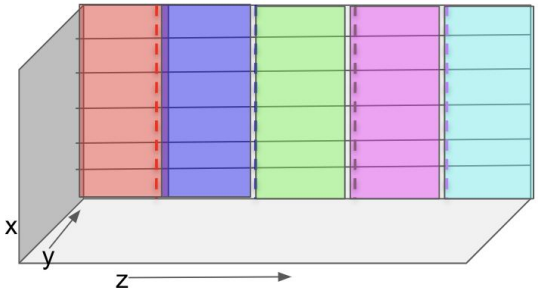


Varying longitudinal segmentation

Regrouping illustration with 5 z sections

3821.5	4046.1	4270.8	4495.4	4720.1	4944.7	**Z edges**
3933.8	4158.5	4383.1	4607.7	4832.4		**Z centers**



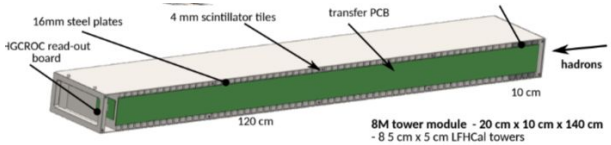


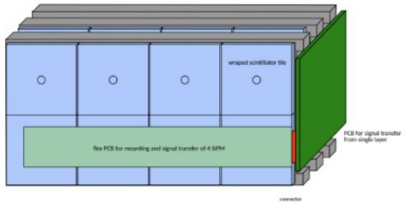
E	Z	Y	X
[[0.028	3821.500	300.000	-1100.000]
[0.058	3844.900	300.000	-1100.000]
[0.092	3938.500	300.000	-1100.000]
[0.070	3961.900	300.000	-1100.000]
[0.109	3868.300	300.000	-1100.000]
[0.132	3891.700	300.000	-1100.000]
[0.116	3915.100	300.000	-1100.000]
[0.001	4429.900	300.000	-1100.000]
[0.001	4359.700	300.000	-1100.000]
[0.003	4055.500	300.000	-1100.000]
[0.016	4008.700	300.000	-1100.000]
[0.032	3985.300	300.000	-1100.000]
[0.003	4032.100	300.000	-1100.000]
[0.003	4125.700	300.000	-1100.000]
[0.001	4172.500	300.000	-1100.000]
[0.003	4078.900	300.000	-1100.000]
[0.001	4149.100	300.000	-1100.000]
[0.001	4640.500	300.000	-1100.000]
[0.001	4242.700	300.000	-1100.000]
[0.002	4102.300	300.000	-1100.000]
[0.001	4195.900	300.000	-1100.000]
[0.001	4336.300	300.000	-1100.000]
[0.001	4289.500	300.000	-1100.000]]

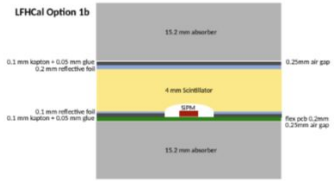
Re-grouping →

Esum	Z centers	Y	X
[[0.656	3933.820	300.000	-1100.000]
[0.016	4158.460	300.000	-1100.000]
[0.003	4383.100	300.000	-1100.000]
[0.001	4607.740	300.000	-1100.000]]

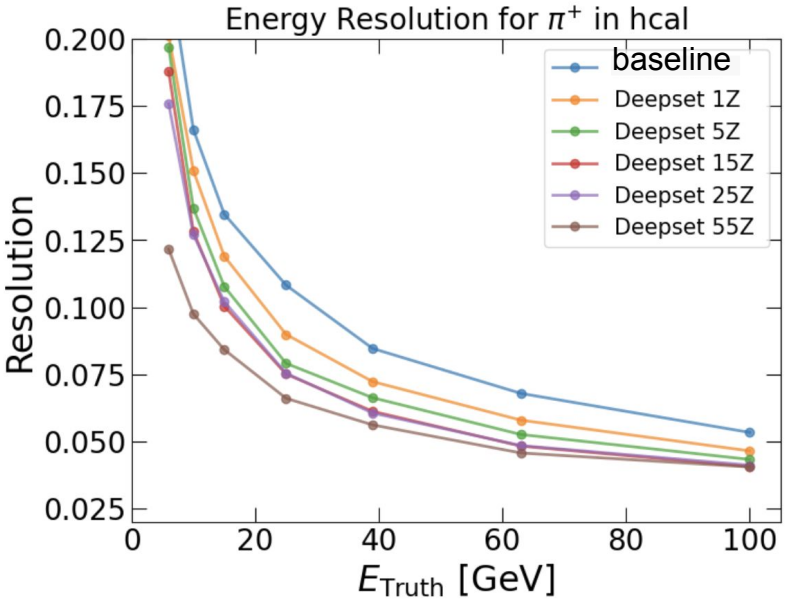
Regrouping in real world is just summing SiPMs outputs

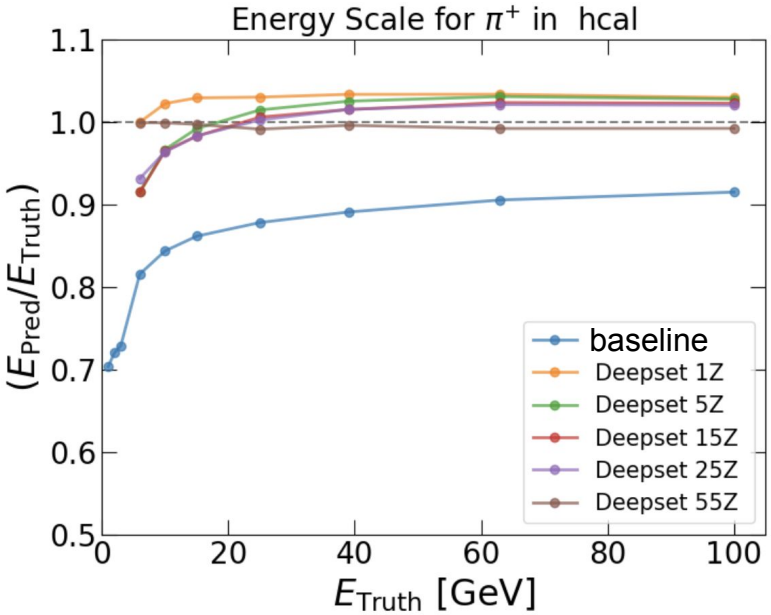






Performance with Z- sections



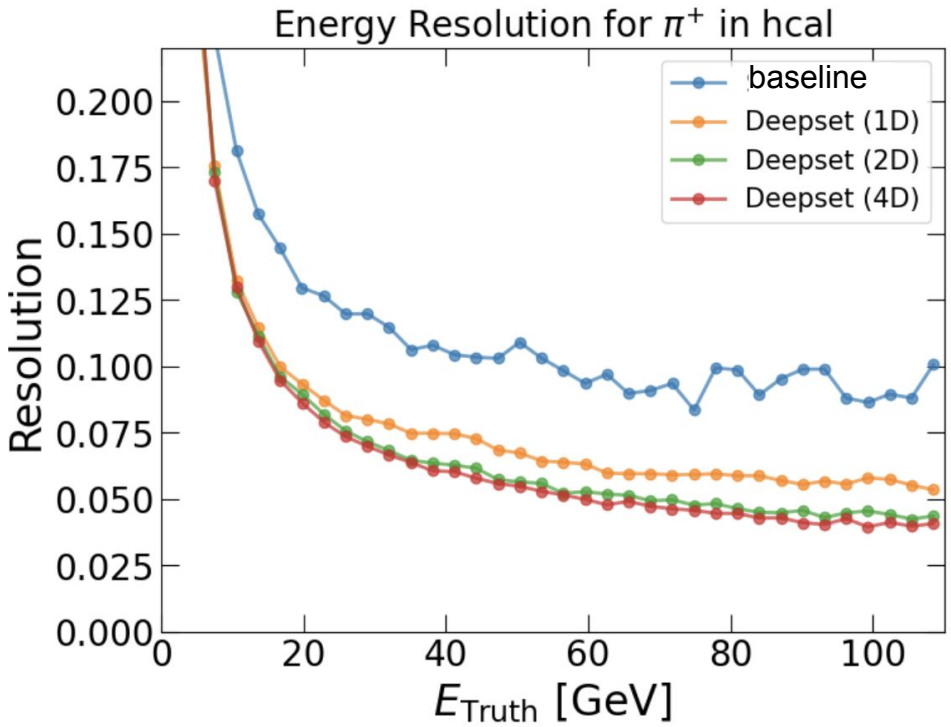


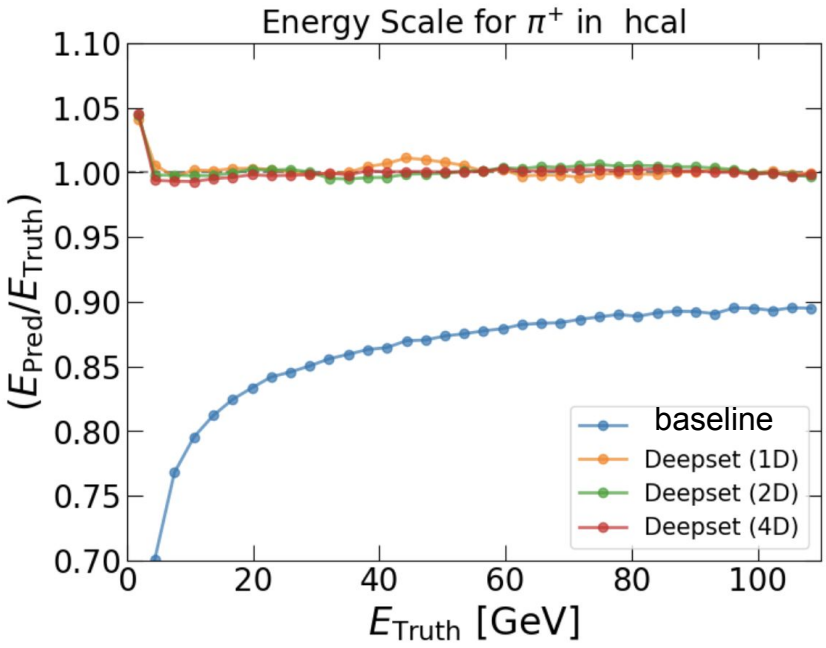
$$Resolution = \sigma\left(\frac{E_{Pred}}{E_{Truth}}\right)_{distribution}$$

- Baseline is sum of cell hit energy corrected by sampling fraction
- Resolution improves with larger number of Z-section (default configuration 7 section)

Performance longitudinal, transverse cell information

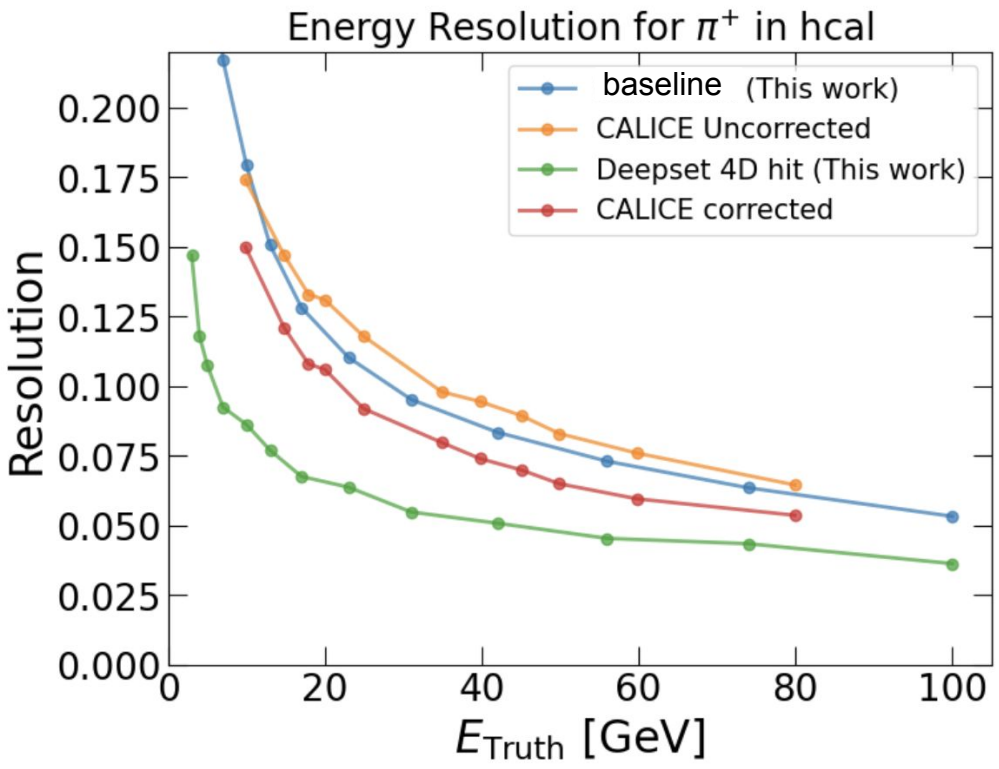
- 1D: cell hits E
- 2D: cell hits E, Z
- 4D: cell hits E, Z, X, Y





Resolution improves most given longitudinal cell information. Transverse cell information improves model performance at high energy

Comparison of performance with existing result



CALICE arXiv:1207.4210v2,2012

- CALICE Fe-Sc calorimeter similar in design
- Our baseline, CALICE uncorrected are sum of cell energy corrected by sampling fraction
- AI based methods yields better performance compared to traditional reconstruction methods

Conclusion:

- **Established a deepset model trained on point cloud to predict the generated energy**
  - **Given different number of longitudinal segments**
  - **Given Transverse and longitudinal cell information (Z, XY)**

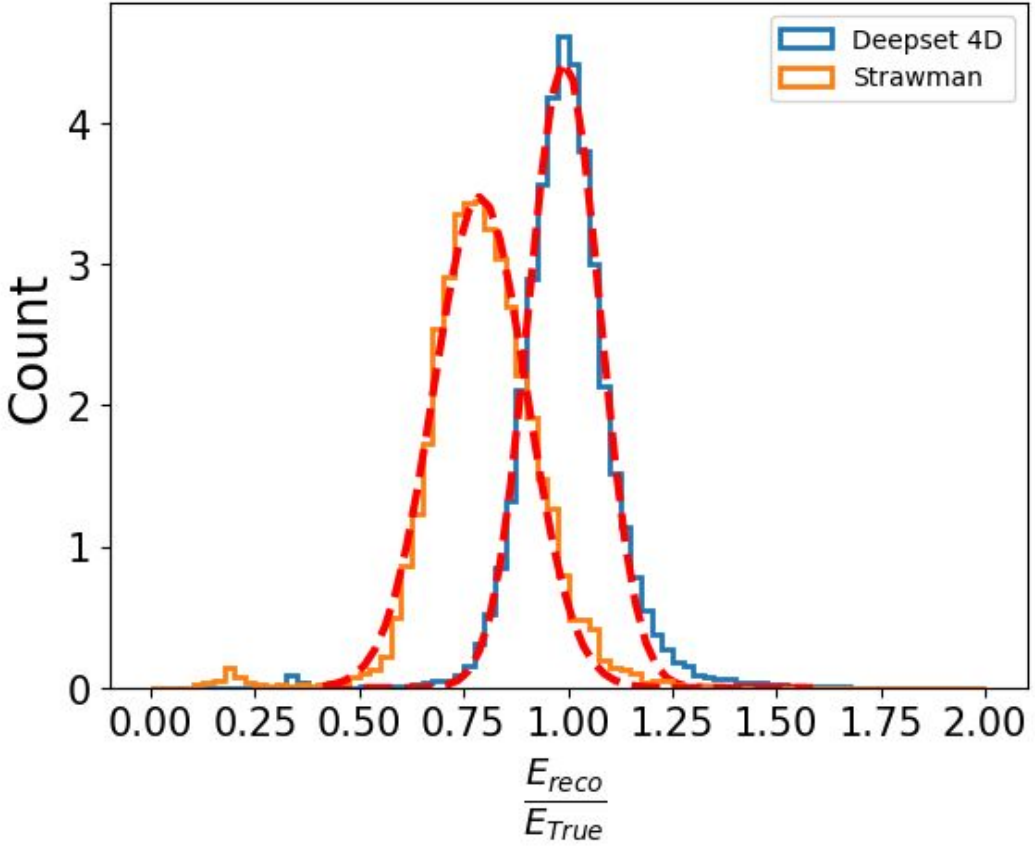
- **Resolution improves most given longitudinal cell information**
- **Transverse cell information improves model performance**
- **AI based reconstruction performs better than traditional reconstruction methods**

Outlook

- **Manuscript in preparation**
- **Develop a model condition on Z-sections**

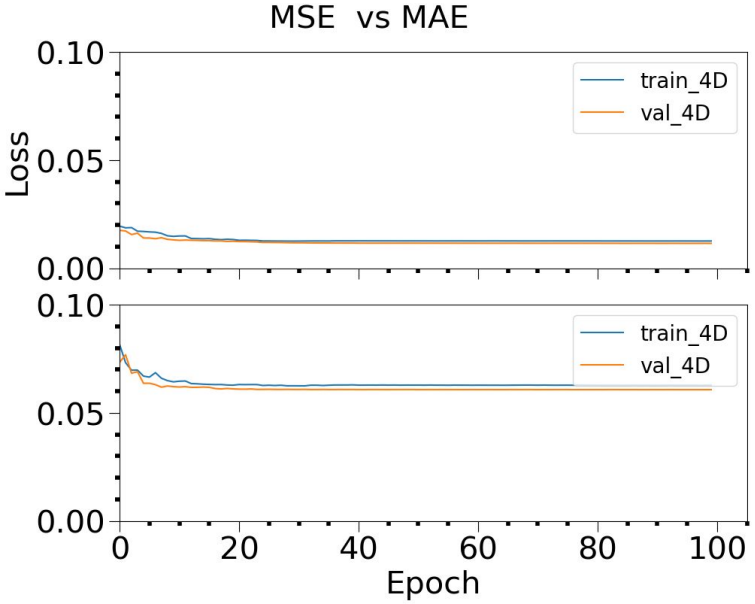
Backup

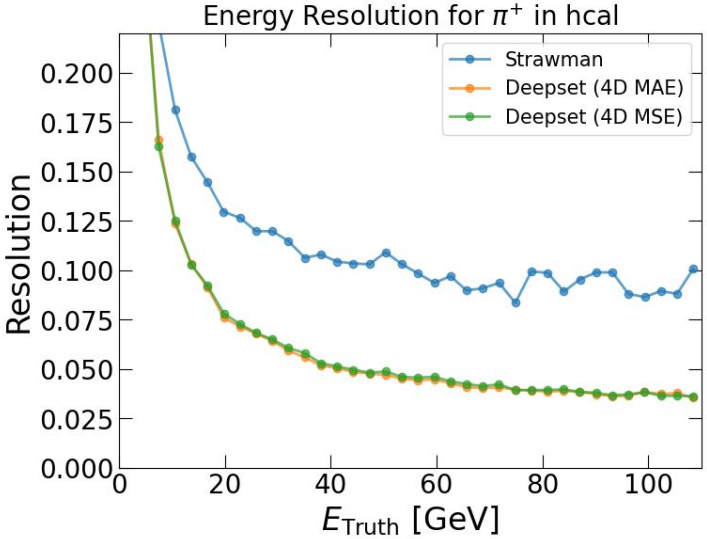
Typical Gaussian Fit



Training input and tuned hyperparameters and architecture

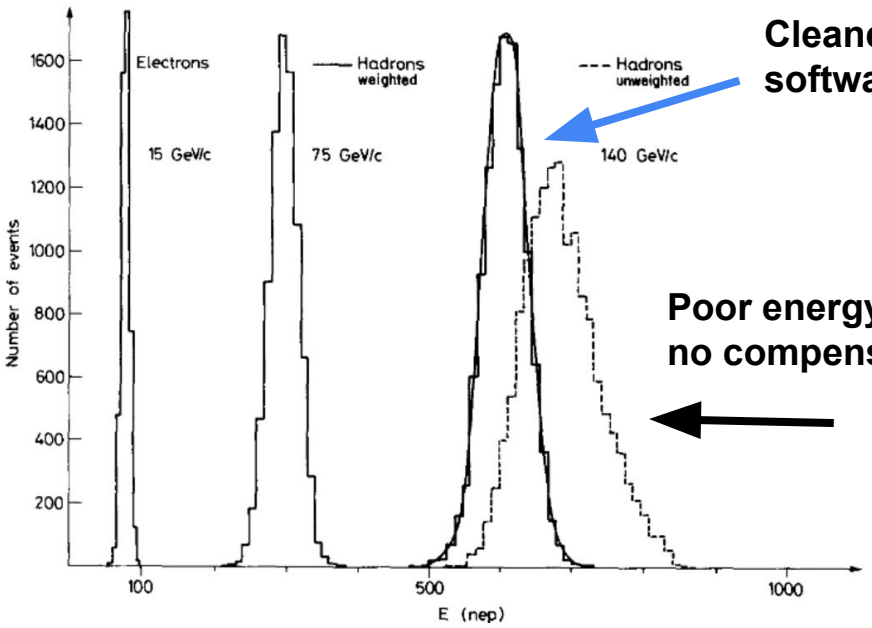
- Used simulated data, 2 M π^+ events, Data splitted: Training, Validation, and Test
- Batch size =2048, number of layers=4, latent size = 64
- Each dense layer uses Rectified Linear Unit (ReLu) activation functions
- Adam optimizer , Mean Squared Error (MSE) for loss
- Trained until converges (approximately 100 Epochs)





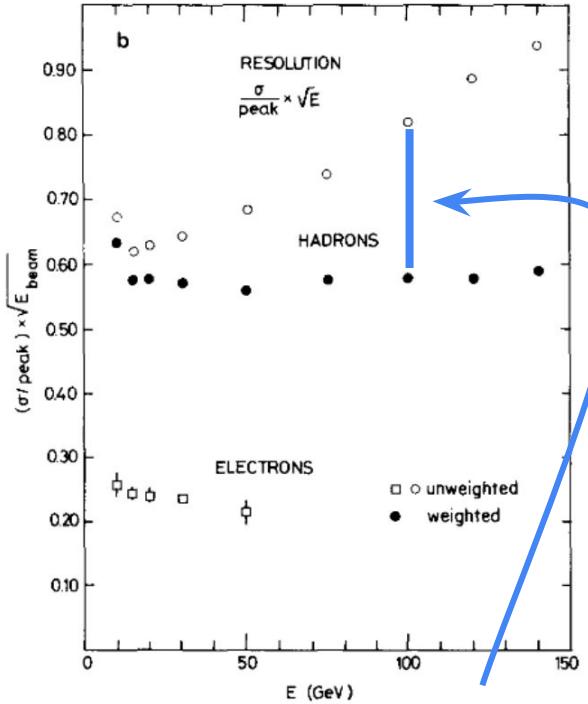
Software compensation has been around since at least 1980!

- CERN study of a longitudinally segmented Fe/Sc scintillator [H. Abramowicz et al., NIM 180 (1981) 429]
- Simple adjustment of cell event energy:
  - $E_{cell,\ weighted} = E_{cell,\ unweighted}(1 - C \cdot E_{cell,\ unweighted})$, $C = 0.03/\sqrt{E_{total}}$



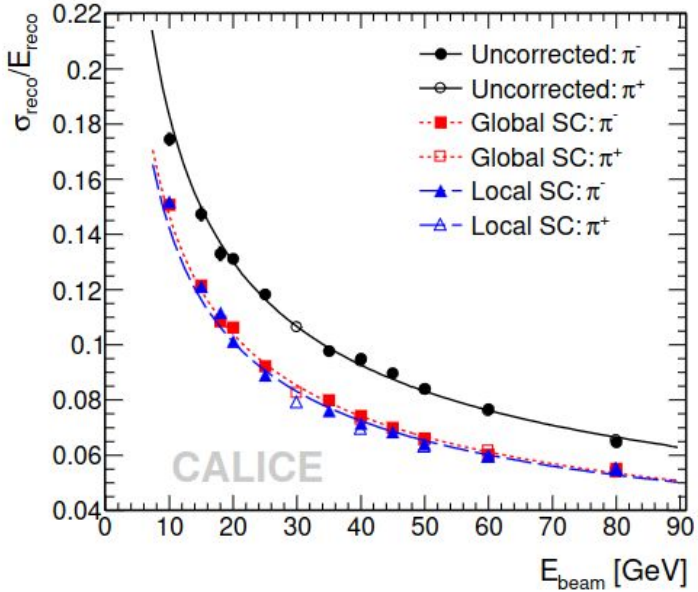
Cleaner Gaussian with software compensation

Poor energy resolution with no compensation

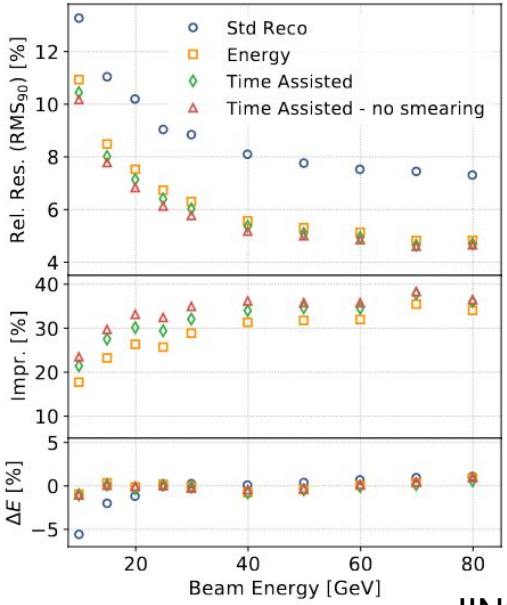


~20% improvement at high energies

Modern software compensation with imaging calorimetry (CALICE Collaboration)



CALICE arXiv:1207.4210v2,2012



(b) Local software compensation

JINST 17 (2022) P08027

- **Human-made algorithm, culmination of many decades of study**
- **Improves resolution by up to 30-40%**

Detector optimization

Generators that mimics detector conditions

Reconstruction algorithm to produce physics object

- Typically neither of components are optimized using the automated tools
- Traditionally only small number of detector configurations are considered as consequence reconstructions algorithm not optimized for a given detector condition
- This challenges can be addressed by deep learning,
- Allows to have high-fidelity fast simulation and optimized reconstruction algorithms