Extrapolation Studies of Longitudinal Energy Distributions and Saving Simulated Detector Hits in the SLCIO File Format

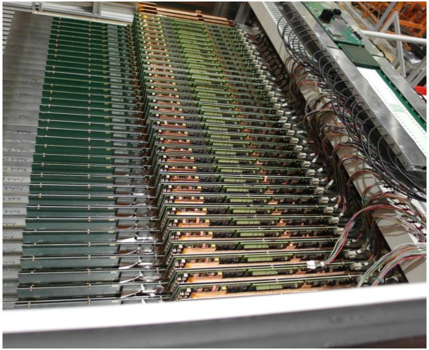
André Wilhahn
In collaboration with Stan Lai and Julian Utehs

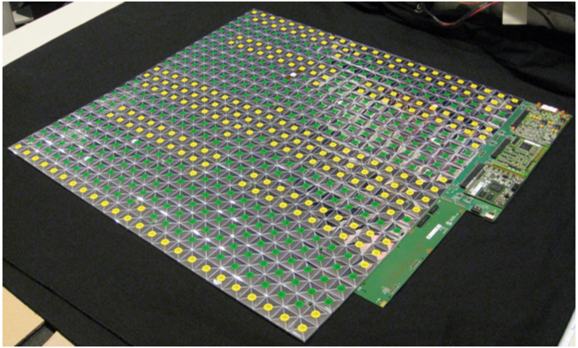
II. Physikalisches Institut, Georg-August-Universität Göttingen
CALICE Collaboration Meeting

September 27 - 29, 2023

Fast Simulation of Longitudinal Energy Distributions

- Working on the development of a **data-based fast simulation** of longitudinal AHCAL energy distributions
- Based upon 2018 test beam data:
  {10 GeV, 20 GeV, 30 GeV, 40 GeV, 60 GeV, 80 GeV, 120 GeV, 160 GeV, 200 GeV}



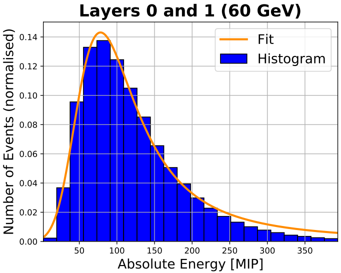


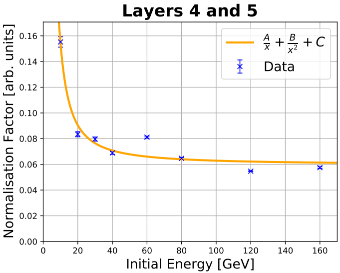
Fast Simulation of Longitudinal Energy Distributions

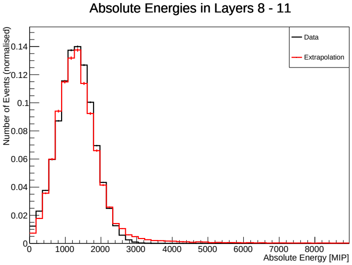
- Previous talk focused on angular-radial-longitudinal bins
  $\Rightarrow$ This talk focuses on only longitudinal bins (i.e. layers)
- Consider layers relative to shower start layer such that shower start layer has always number 0
- Have grouped layers together into "layer groups"
  $\Rightarrow$ Will use terms "layers" and "layer groups" analogously
- Layer groups: $\{0-1,\ 2-3,\ 4-5,\ 6-7,\ 8-11,\ 12-15,\ 16-23,\ 24-38\}$

Strategy for Extrapolating Energy Distributions to 200 GeV

1. Find curves that describe absolute energy deposition layerwise for target energy
   a) Fit one fit function to all PDFs of initial energies ≤ 160 GeV
   b) Plot fit parameters against initial energy
   c) Extrapolate fit parameters to 200 GeV
   d) Create curves for 200 GeV based on extrapolated fit parameters
2. Extrapolate events by integrating absolute energy distributions of 160 GeV and 200 GeV (explained in more detail during this talk)







Convoluting a Gaussian and a Landau Distribution

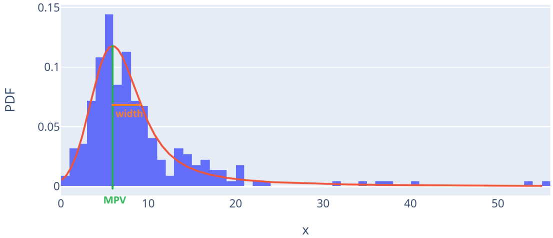
- Fit convolution of Gaussian and Landau distribution ("Langaus") to PDFs of absolute energies:

$$f_{\text{Gaus}}(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left\{-\frac{(x-\bar{x})^2}{2\sigma^2}\right\}$$

$$f_{\text{Landau}}(x) = \frac{1}{\pi\eta} \int_0^\infty e^{-t} \cos\left\{\frac{t(x-\mu)}{\eta} + \frac{2t}{\pi} \log\left(\frac{t}{\eta}\right)\right\} \mathrm{d}t$$

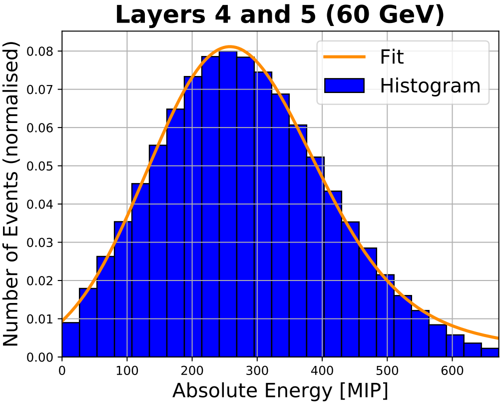
Four fit parameters:

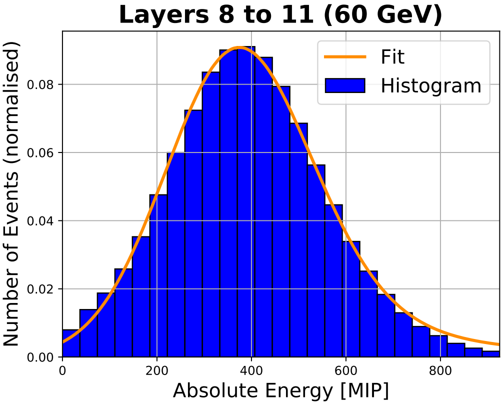
1. Normalisation factor A
2. Most probable value "mpv"
3. Scaling factor η (from Landau)
4. Standard deviation σ (from Gauss)



Fitting Absolute Energies

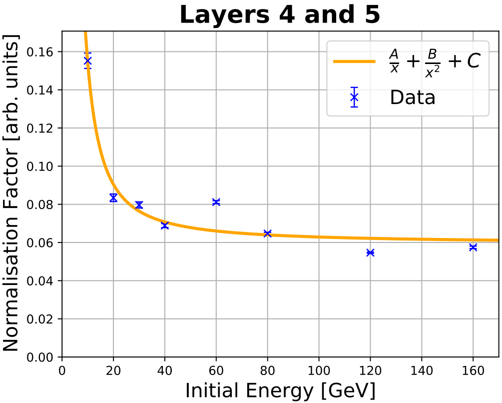
- Behaviour around maxima well described
- Deviations at tails, but negligible for now
- Similar results for other initial energies/layer groups

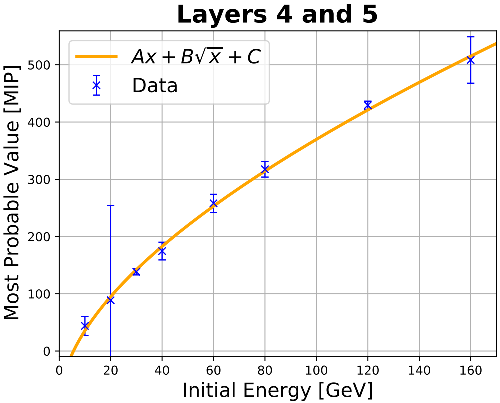




Fit Parameters vs. Initial Energy

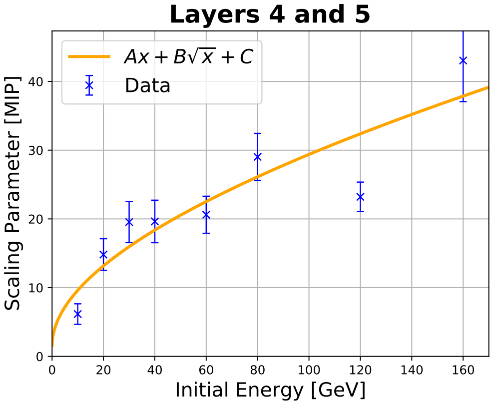
- Fit functions to points in order to describe behaviour of fit parameters as function of initial energy → trial and error, functions not based on any model
- Overall sensible results for A and mpv

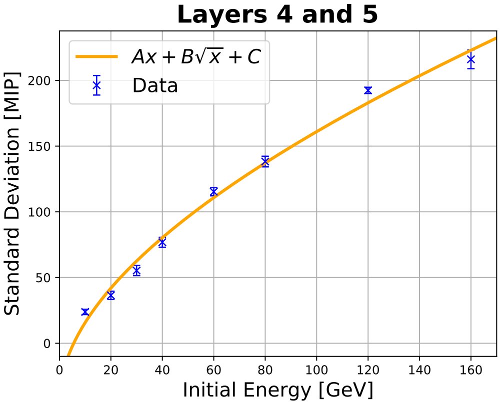




Fit Parameters vs. Initial Energy

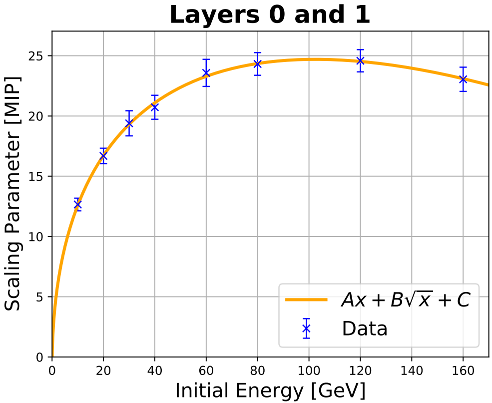
- Also mostly sensible results for η and σ
- However, some few curves exhibit weird behaviour $\rightarrow$ next slide

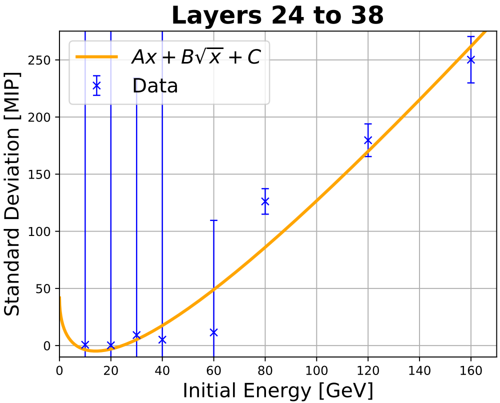




Some Outliers

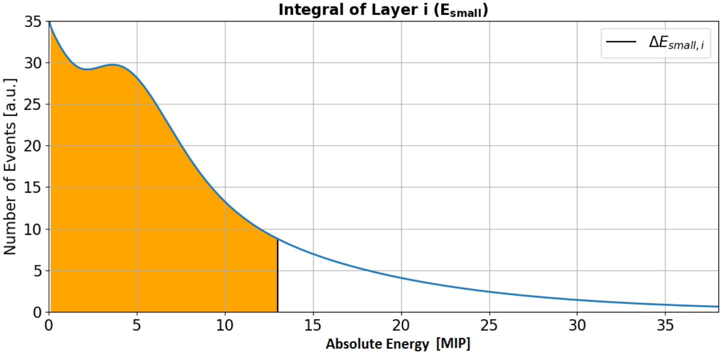
- Sometimes fit parameters are really hard to find (e.g. at small initial energies in large detector layers)
- In particular, η very hard to control $\Rightarrow$ probably σ controls most of the curve's width





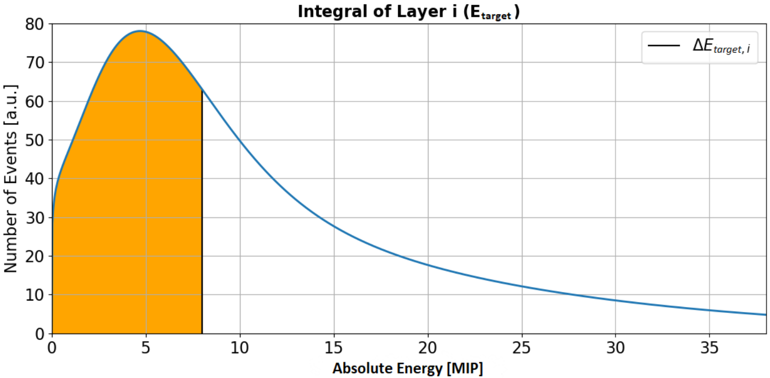
Generating extrapolated Events by Integration of PDFs

- Extrapolate from one initial energy ($E_{\text{small}} = 160\,\text{GeV}$) to target energy ($E_{\text{target}} = 200\,\text{GeV}$)
- Generate one event of absolute energies randomly $\rightarrow$ ($E_{\text{small},\,0}$, $E_{\text{small},\,1}$, ..., $E_{\text{small},\,n}$)
- Integrate PDF of layer i until $E_{\text{small},\,i}$ is reached



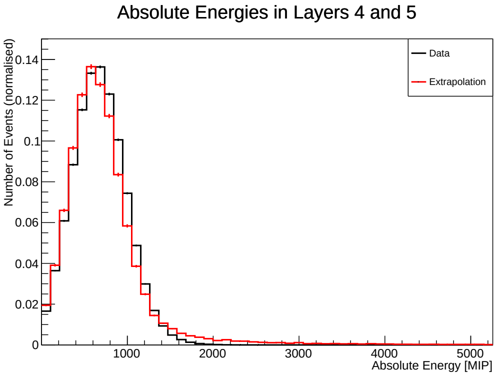
Generating extrapolated Events by Integration of PDFs

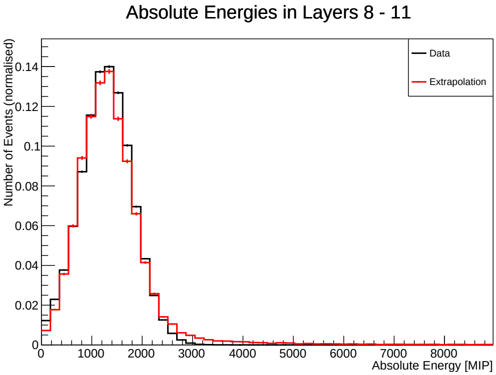
- Integrate PDF of same layer but for E_{target}, until area equals A_i → at $E_{\text{target},\,i}$
- Repeat for all layers → $(E_{\text{target},\,0}, E_{\text{target},\,1}, \dots, E_{\text{target},\,n})$
  ⇒ Preserves correlations between layers



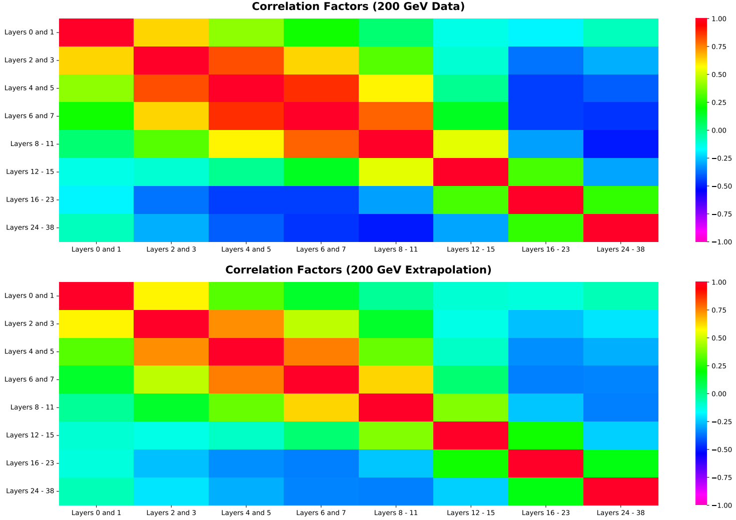
Extrapolated Absolute Energies

- After fit parameter extrapolation, extrapolate absolute energies via integration shown on previous slides
- Very good behaviour around maximum, but tails too large (too many events with too much energy)





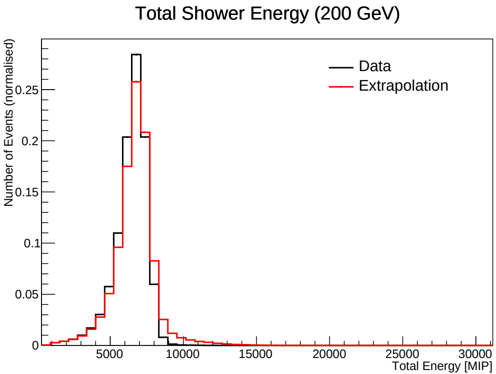
Extrapolated Correlation Factors

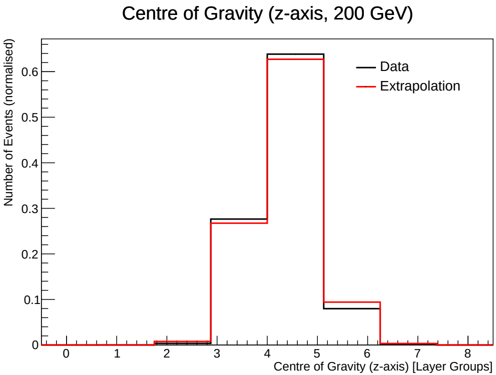


- Extrapolated correlations slightly “weaker” than expectations
- Shows that extrapolation already deteriorates for differences in initial energies of only 40 GeV

Kinematic Variables

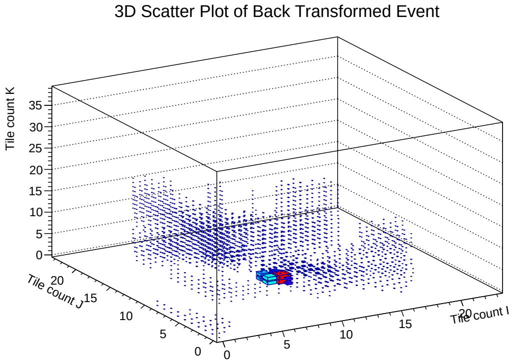
- Check total shower energy and centre of gravity as examples to make sure that shower behaviour is correctly extrapolated too
- Similar case as for absolute energies: maximum well recreated by extrapolation, but high-end tail deviates from data



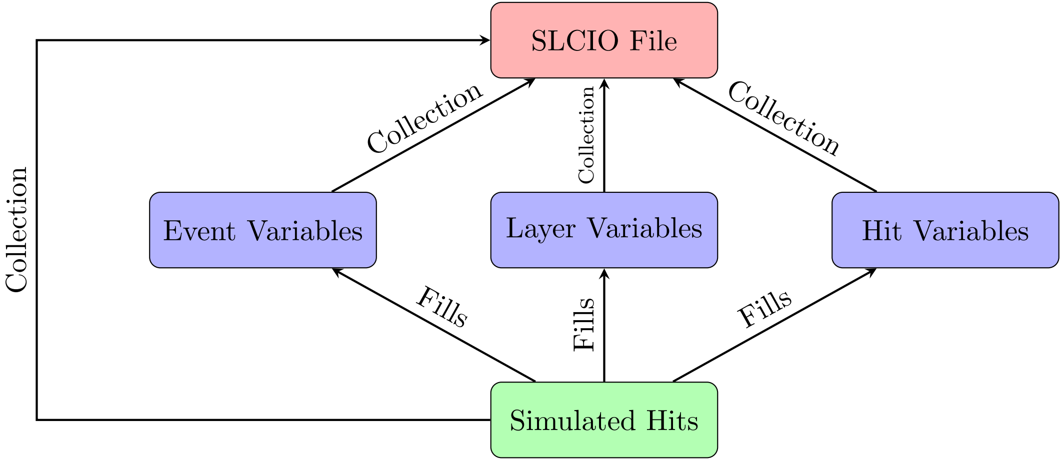


Convertion from Simulated Hits into SLCIO Files

- Until now have only worked with textfiles for both temporary and permanent saving
- Standard file format in CALICE are SLCIO files
- SLCIO files include certain collections $\rightarrow$ vectors of object pointers
- Goal: create dummy SLCIO file with test event/hits, then convert SLCIO file into ROOT file



SLCIO Structure and Collections



SLCIO Collections - Calorimeter Hits and Hit Variables

Calorimeter Hits:

- Two Cell IDs: ID 0 depends on detector hit position (IJK-coordinates), ID 1 irrelevant for simulation (always zero)
- Hit energy and hit energy error
- Double array with global x-, y-, and z-coordinates of hit (x and y measured w.r.t. detector center, z measured from beginning of first layer)

Hit Variables:

- Hit position in detector coordinates (IJK)
- Cell size (area of single cell in millimetres squared)
- Radial distance of hit from center of gravity in x-y plane
- Hit energy density (hit energy divided by cell size)

SLCIO Collections - Layer and Event Variables

Layer Variables:

- Number of hits per layer
- Total energy and energy density per layer
- Mean event radius and weighted event radius per layer (latter weighted by hit density)
- Centers of gravity in all three spatial directions

Event Variables:

- Total number of hits
- Shower start layer number
- Total energy and energy density
- Mean event radius and weighted event radius per layer (latter weighted by hit density)
- Centers of gravity in all three spatial directions

Summary and Limitations of the Extrapolation Algorithm

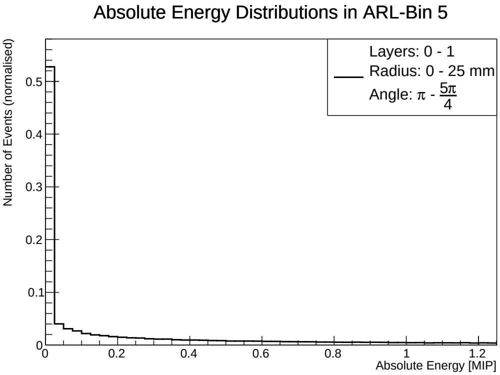
Pros:

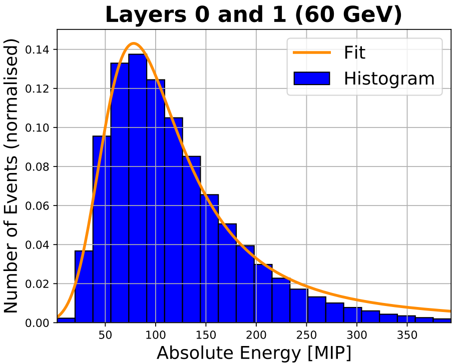
- Langaus curves fit absolute energy distributions (mostly) well
- Extrapolation of fit parameters and distributions works reasonably well too
- Extrapolated distributions match those from data (except for high energy tails)

Cons:

- Requires to fit Langaus curve binwise
  $\Rightarrow$ Nobody wants to fit (and adjust) Langaus distribution to hundreds of PDFs
- With finer binning, absolute energy PDFs do not exhibit Langaus behaviour anymore
- During extrapolation, one also has to calculate cumulative Langaus distributions of the extrapolated target energy
  $\Rightarrow$ Computing time already quite long for only longitudinal bins (only layerwise energy deposition), but would take ages to do same calculation for even finer binning

Summary and Limitations of the Extrapolation Algorithm





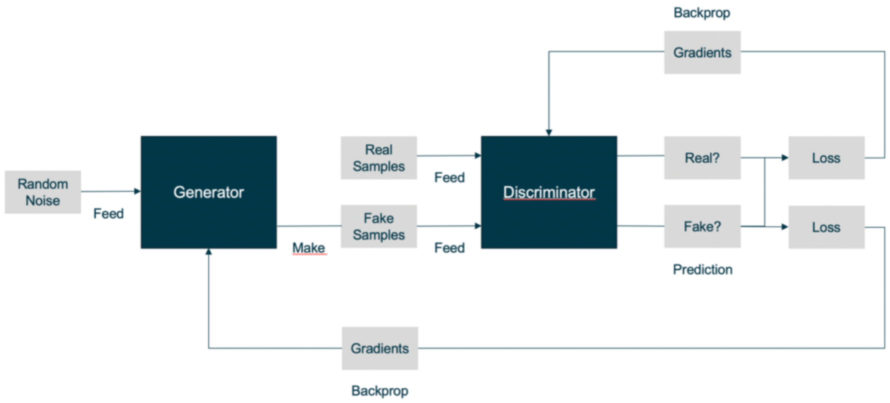
- Soon have to replace old extrapolation algorithm by new one involving machine learning techniques if we want to keep fine resolution
- Ideal case would be to have no fitting at all involved because it requires too much fine tuning

Summary of the SLCIO File Creation

- Fast simulation almost so far that consistent events can be produced
- Events (and hits) are stored in SLCIO files, together with certain variables
- Finding out how SLCIO files are structured, what needs to be saved, and how filling works took quite some time
- Was able to create first SLCIO file with dummy values and convert it into ROOT file
- ROOT tree writer is also able to convert dummy SLCIO files into ROOT files

Outlook

- Next project will be to simulate showers with “Generative Adversarial Networks”
- Feed cell information (energy, timing, ...) to GAN and try to generate showers that look as if they came from test beam data

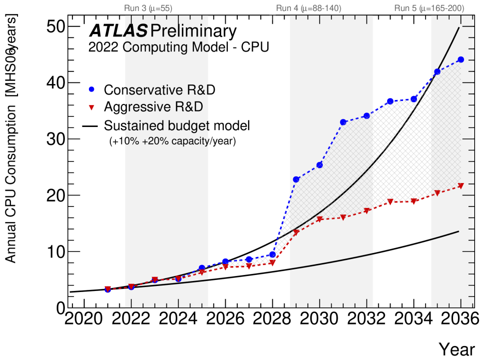


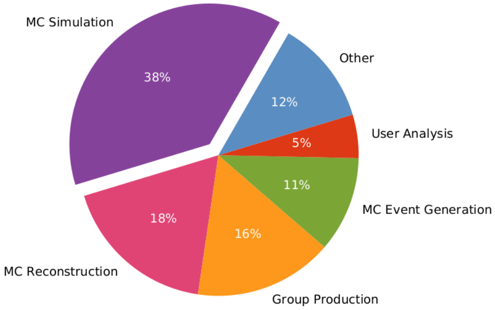
Thanks for your attention!
Questions?

Backup Slides

Why Fast Simulations?

- CPU consumption of MC simulations increases with occupancy/granularity
- Up to 90 % of calculation time is needed for the calorimeter
- Saving of computational resources will become necessary sooner or later





Kernel Density Estimators

- Want to find PDF of dataset x_1, x_2, ..., x_n
- Define Kernel Density Estimator (KDE) with bandwidth h as:

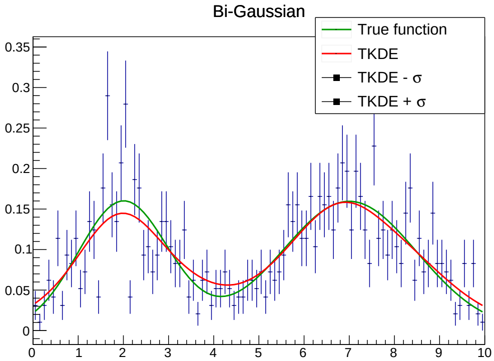
$$f(x) = \frac{1}{nh} \sum_{i=1}^{n} K\left(\frac{x - x_i}{h}\right)$$

with

$$K(x) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2}x^2\right)$$

- PDF = sum of all Gaussian kernels
- Choice of bandwidth determines smoothness of PDF
- Apply KDE of energy differences simultaneously on layer groups

Kernel Density Estimators



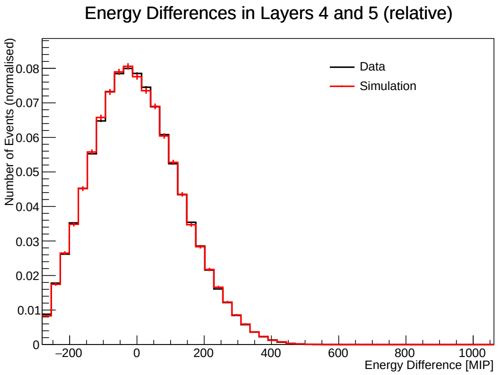
- Generalise to d dimensions:

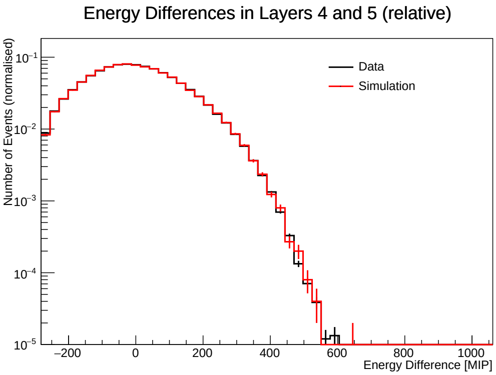
$$f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} |\mathbf{H}|^{-1/2} K\left(\mathbf{H}^{-1/2}(\mathbf{x} - \mathbf{x}_i)\right)$$

- $\mathbf{x}$: d-dimensional data vector; $\mathbf{H}$: $d \times d$ bandwidth matrix
- Have chosen $\mathbf{H} = h^2\mathbf{C}$ where $\mathbf{C}$ is the covariance matrix of the dataset

Simulated Energy Differences for longitudinal 60 GeV Pion PDFs

- Generate 100 000 events with KDEs
- Each event containing eight energy differences corresponding to eight layer groups





Excellent agreement between data and simulation!

Convolution

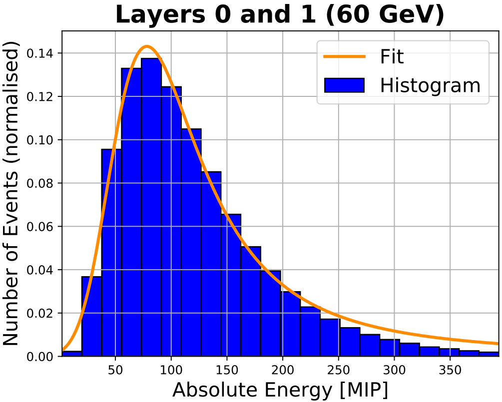
The convolution of two functions f and g (denoted as $f * g$) is defined as:

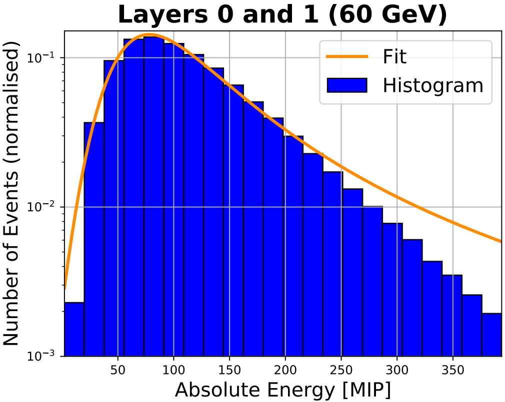
$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t-\tau)\mathrm{d}\tau$$

or equivalently:

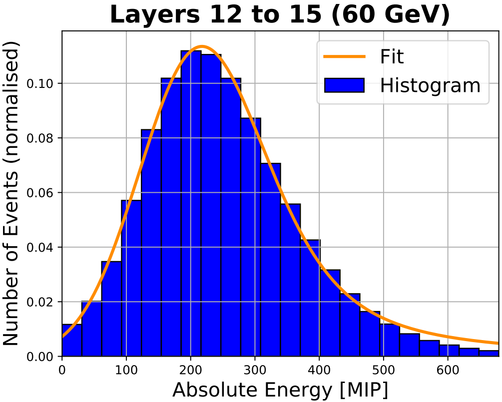
$$(f * g)(t) = \int_{-\infty}^{\infty} f(t-\tau)g(\tau)\mathrm{d}\tau$$

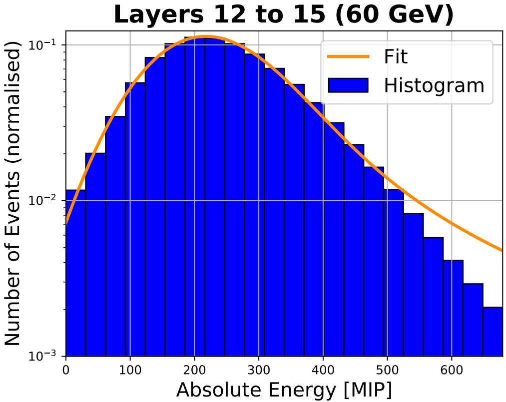
Fitting Langaus Curves to Absolute Energy PDFs I



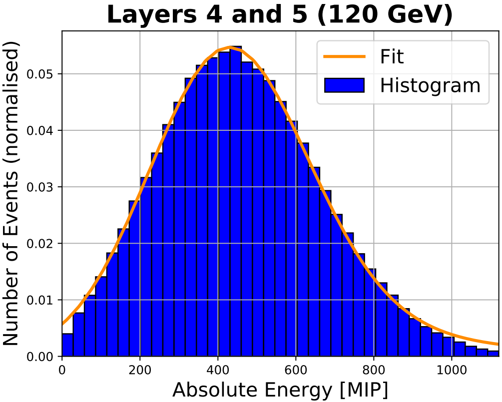


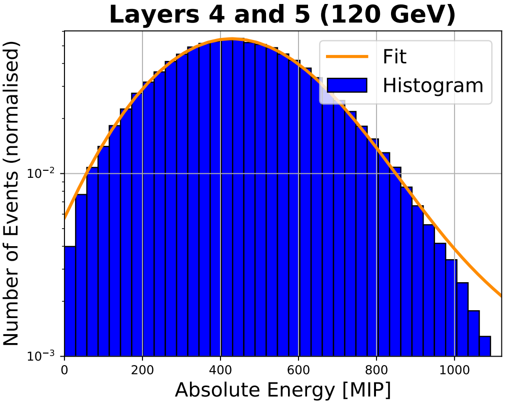
Fitting Langaus Curves to Absolute Energy PDFs II



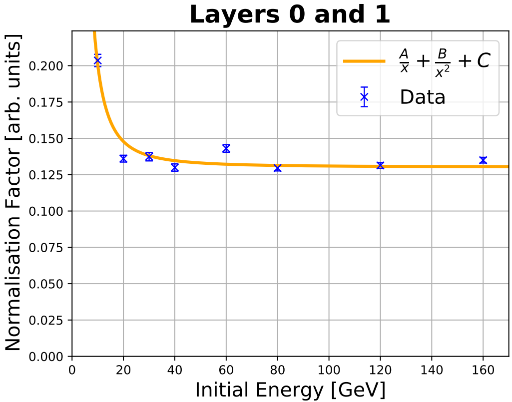


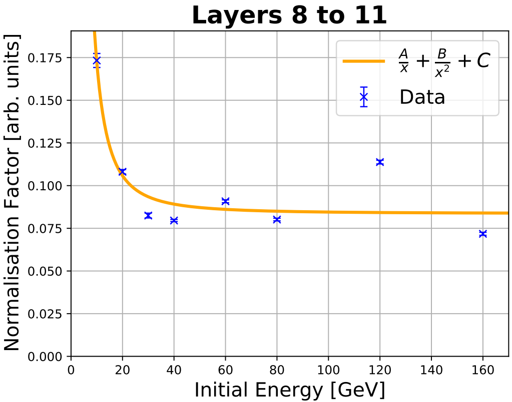
Fitting Langaus Curves to Absolute Energy PDFs III



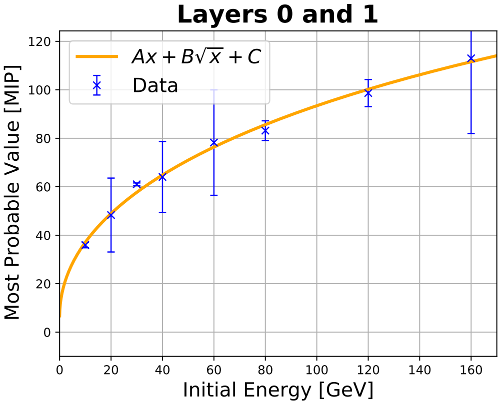


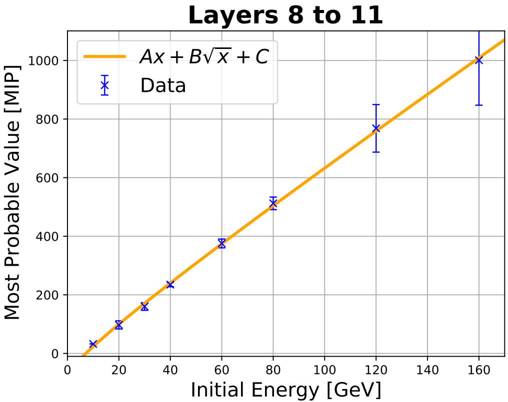
Normalisation Factor



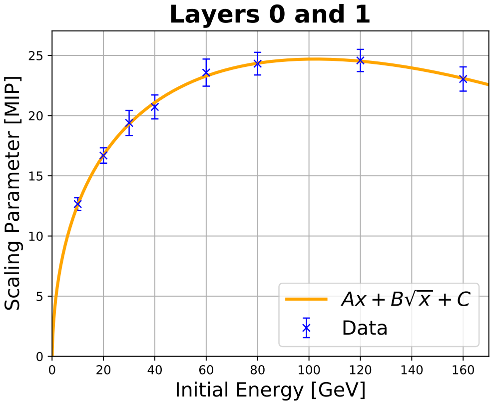


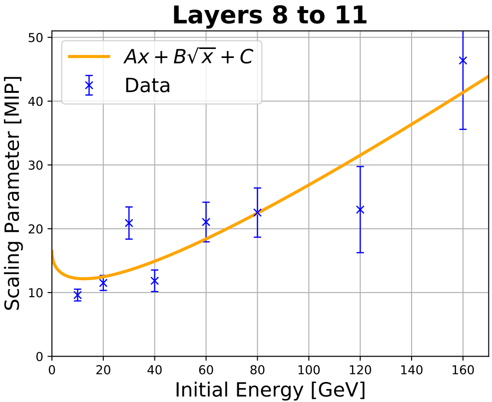
Most Probable Value



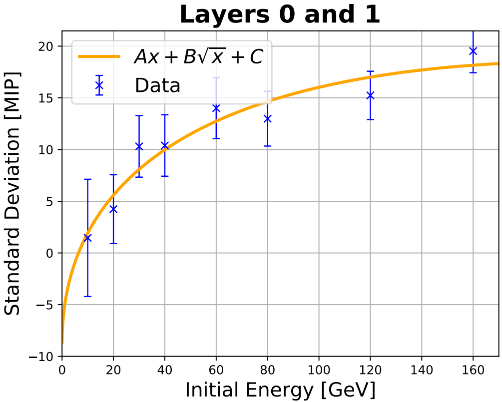


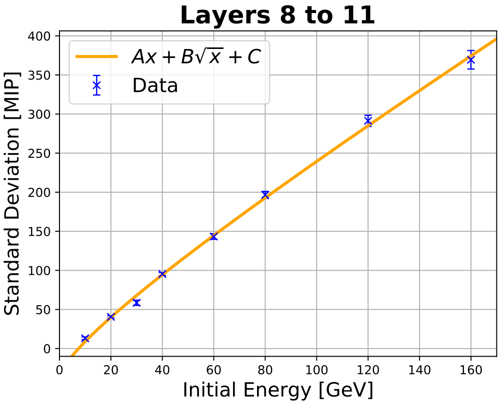
Scaling Factor





Standard Deviation

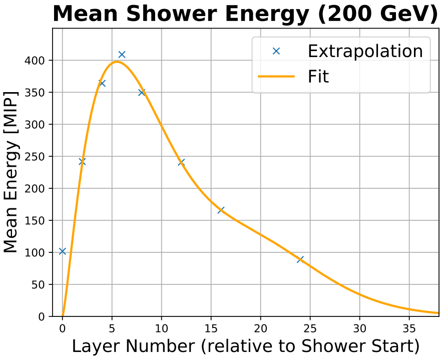




Distributing Energy of Layer Groups among Layers

$$E(n) = E_0 \cdot \left\{ \frac{f}{\Gamma(\alpha_s)} \left(\frac{n}{\beta_s}\right)^{\alpha_s - 1} \frac{e^{\frac{-n}{\beta_s}}}{\beta_s} + \frac{1-f}{\Gamma(\alpha_l)} \left(\frac{n}{\beta_l}\right)^{\alpha_l - 1} \frac{e^{\frac{-n}{\beta_l}}}{\beta_l} \right\}$$

- Electromagnetic core and hadronic halo
- Integrate curve from point i to $i+1$ $\rightarrow$ area for certain layer group
- Within layer group, integrate for single layers
  $\Rightarrow$ **Example:** LG 1 comprises layers 0 and 1 $\rightarrow$ integrate from 0 to 2
  $\Rightarrow$ For layers 0 and 1: integrate from 0 to 1 and from 1 to 2, respectively



Centre of Gravity in Single Layers