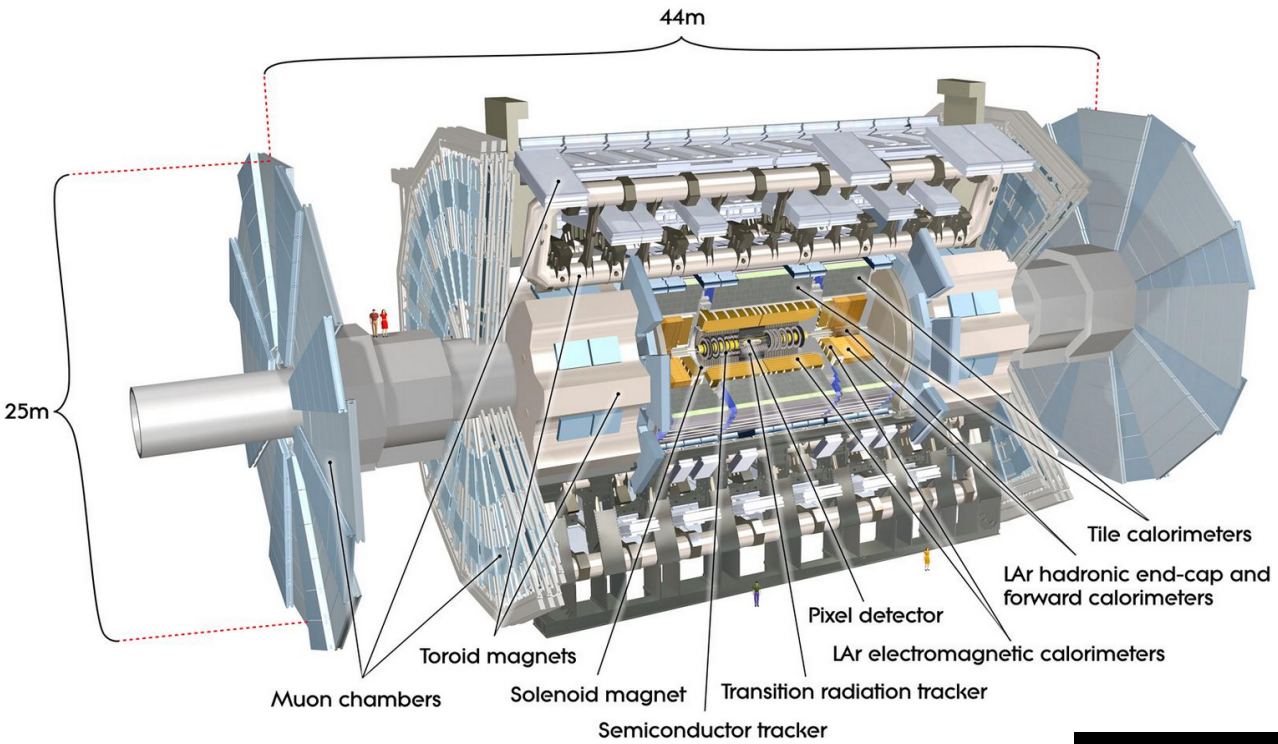
# The New ATLAS Fast Calorimeter Simulation (FastCaloSim)

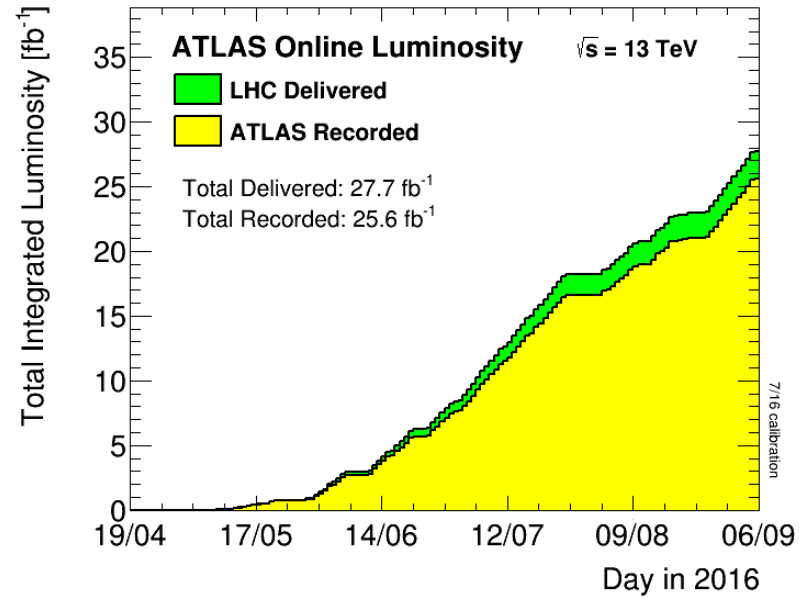
Jana Schaarschmidt (University of Washington), on behalf of the ATLAS collaboration

CHEP2016 - San Francisco - 10-14. October

# Introduction

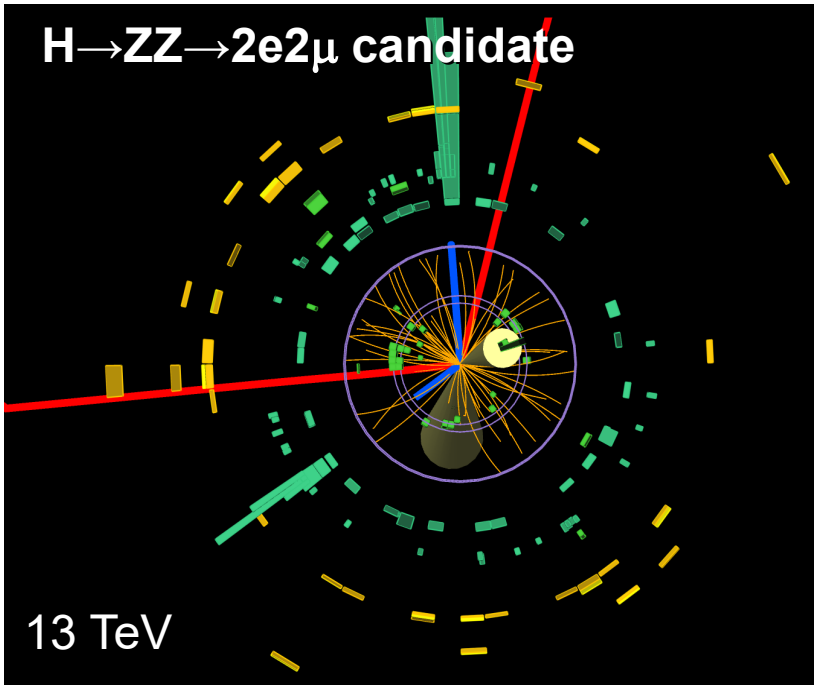





- Tremendous LHC and detector performances
- Lots of highly complex physics analysis
- Pile-up interactions $<\mu>=23$ at $10^{34}$ $cm^{-2}s^{-1}$

→ Enormous need for simulation

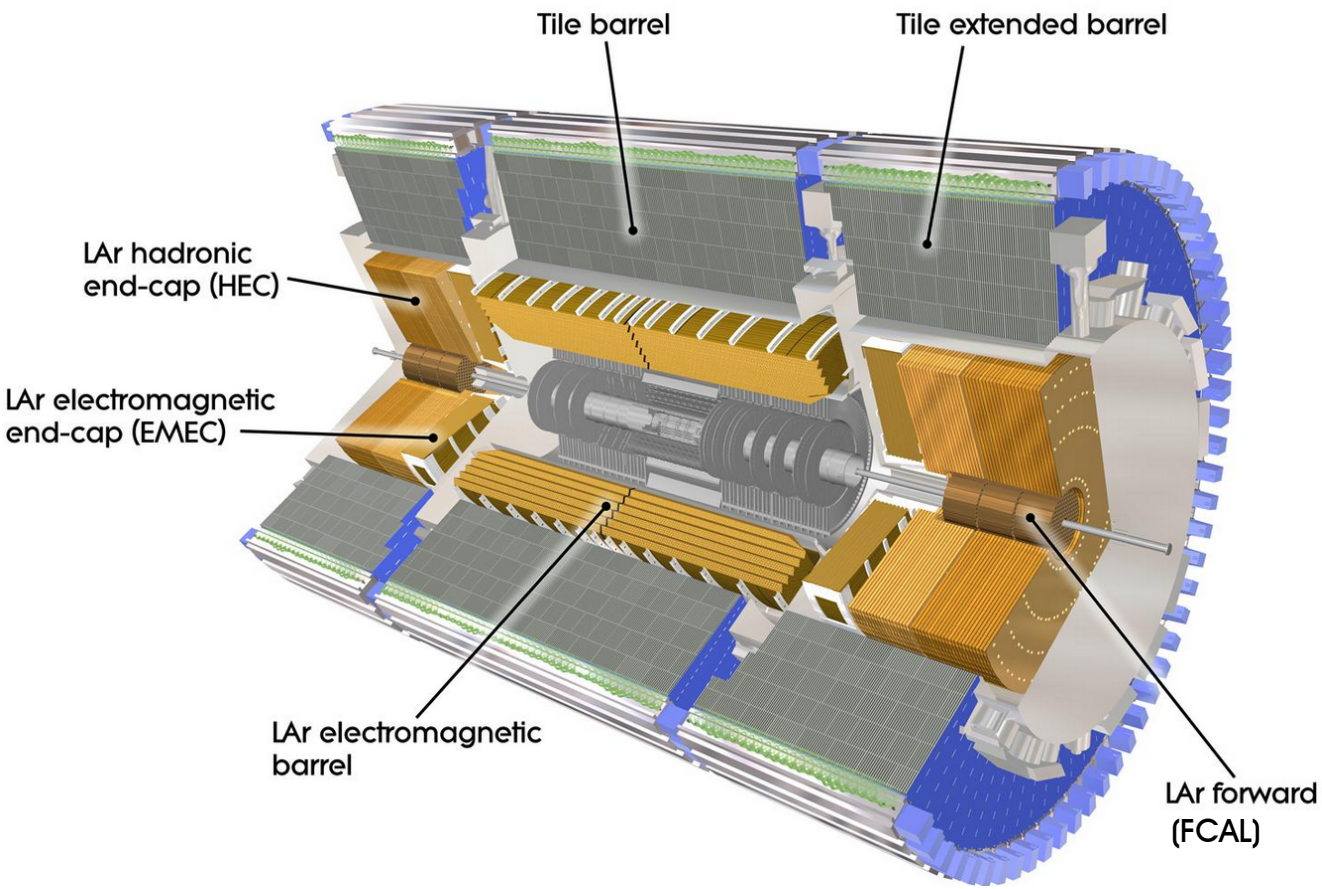



All ATLAS results:

https://twiki.cern.ch/twiki/bin/view/AtlasPublic

# ATLAS Calorimeters



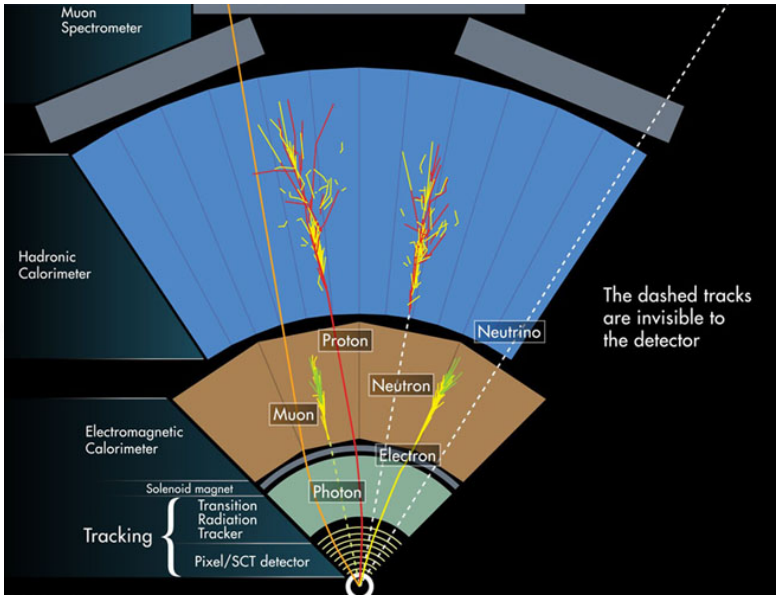



Covering $|\eta|<4.9$

Materials:

Liquid Argon + Lead, or copper or tungsten

Tile Cal: Steel + plastic



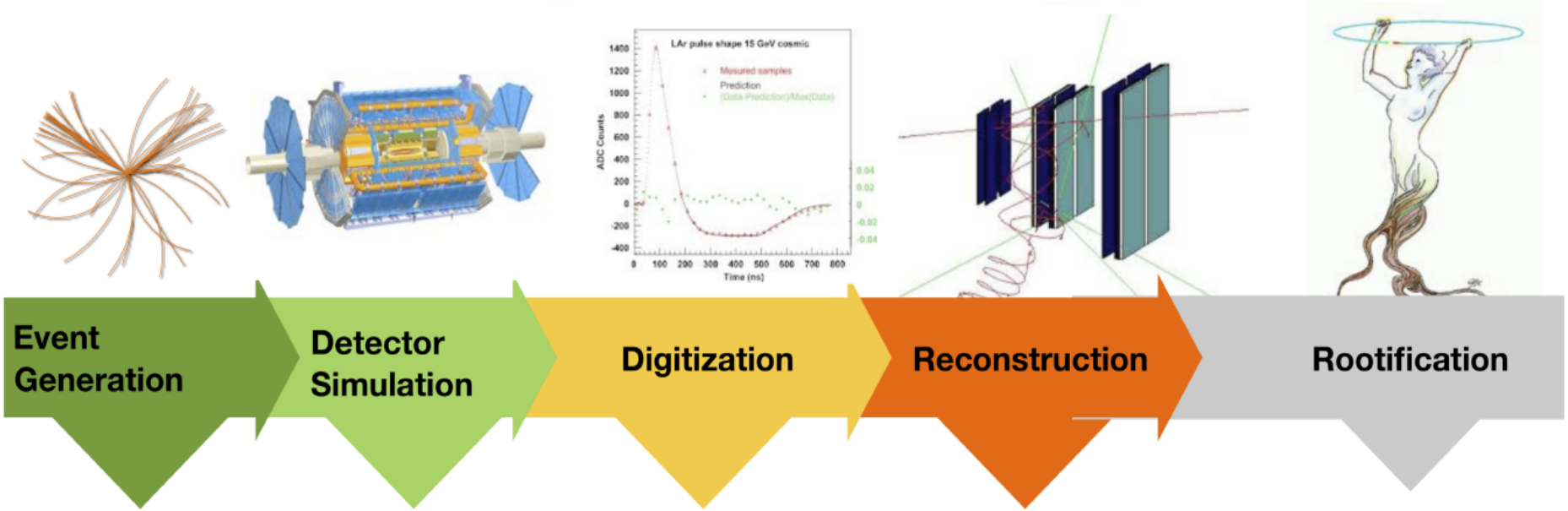

# Readout channels: ~190 k in total

| System | EM Barrel | EM EC | Hadronic EC | FCAL | Tile |
|---|---|---|---|---|---|
| #Channels | 110k | 64k | 5.6k | 3.5k | 9.8k |

Crucial for photons & electrons, jets and missing energy reconstruction

# ATLAS MC Production Chain



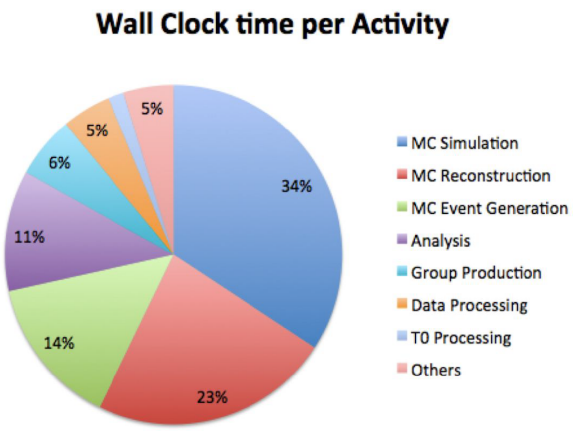



Format: EVGEN HITS RDO ESD/AOD NTUPLE/HIST

Typical times in s: (2010, simulation with Geant4)

| Sample | Generation | Simulation | Digitization | Reconstruction |
|---|---|---|---|---|
| Minimum Bias | 0.0267 | 551. | 19.6 | 8.06 |
| $t\bar{t}$ Production | 0.226 | 1990 | 29.1 | 47.4 |
| Jets | 0.0457 | 2640 | 29.2 | 78.4 |
| Photon and jets | 0.0431 | 2850 | 25.3 | 44.7 |
| $W^{\pm} \to e^{\pm}\nu_e$ | 0.0788 | 1150 | 23.5 | 8.07 |
| $W^{\pm} \to \mu^{\pm}\nu_\mu$ | 0.0768 | 1030 | 23.1 | 13.6 |
| Heavy ion | 2.08 | 56,000 | 267 | - |

Grid usage 2016:




→ Simulation takes the longest in our production chain, takes most of the grid resources
→ 90% of the simulation time is spent in the calorimeters (showering)

# (Current) FastCaloSim



- **FastCaloSim:**
  Parametrized calorimeter response simulation, based on the Geant4 simulation of single particles in a fine grid of particle energies and directions
- Single particles: Electrons & photons (EM interactions), charged pions (hadronic interactions)
- Parametrization split into longitudinal and lateral shower development
- Widely and very actively used: about half of all our MC events is simulated using FastCaloSim (examples: all SUSY signal samples, some ttbar samples, ...)
- FastCaloSim+Geant4 simulation of inner detector and muon system = ATLFASTII
- ATLFASTII is a factor 10 faster than full simulation with Geant4

Averaged simulation times per event in seconds:

| Sample | Full G4 Sim | Fast G4 Sim | ATLFast-II | ATLFast-IIF |
|---|---|---|---|---|
| Minimum Bias | 551 | 246 | 31.2 | 2.13 |
| $t\bar{t}$ | 1990 | 757 | 101 | 7.41 |
| Jets | 2640 | 832 | 93.6 | 7.68 |
| Photons, jets | 2850 | 639 | 71.4 | 5.67 |
| $W \rightarrow e\nu$ | 1150 | 447 | 57.0 | 4.09 |
| $W \rightarrow \mu\nu$ | 1030 | 438 | 55.1 | 4.13 |

Precalculated showers

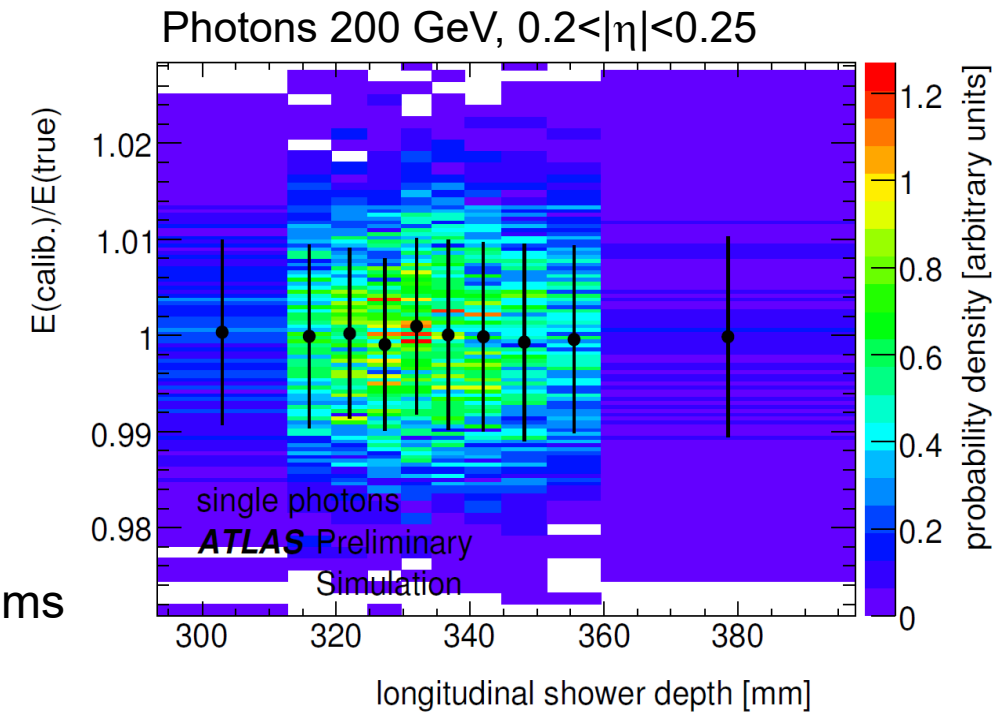
FastCaloSim+Fast Tracking Simulation

# (Current) FastCaloSim



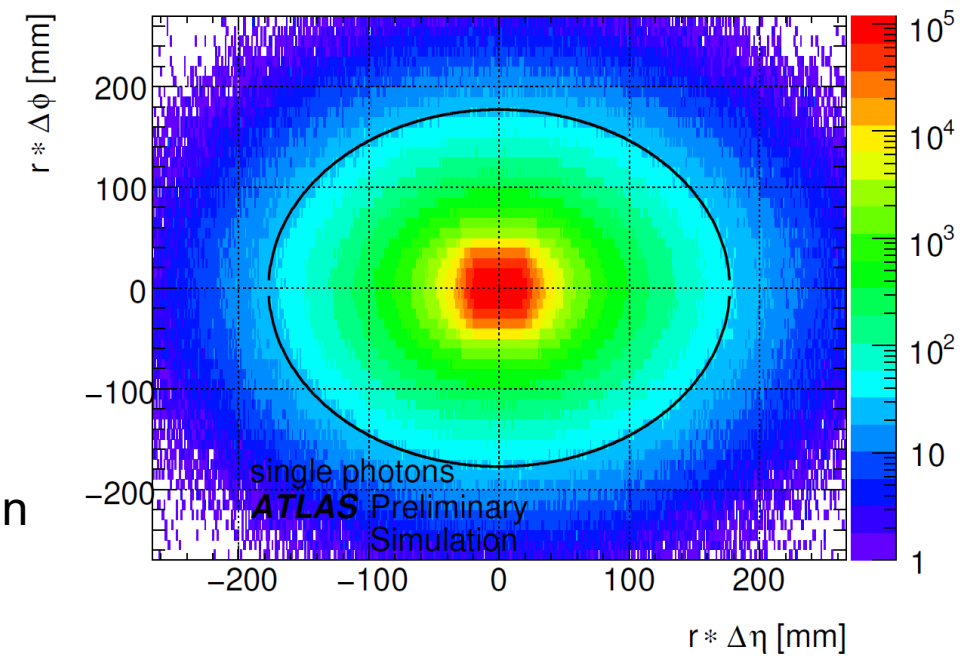
**Longitudinal energy parametrisation:**

- For each particle, energy and |η| store 2D histograms of energy vs. longitudinal shower depth (distance of the deposit from the calo surface), for total energy and energy fraction per layer
- Correlations between the deposits in each layer stored in correlation matrices
- Simulation: Randomly draw an energy value and energy fractions from the stored 2D histograms




**Lateral shower shape parametrization:**

- Radial symmetric function centered around the impact point of a particle in the calo layer, (3rd order polynomial function), modified with parameters to describe asymmetries when particles cross the calorimeter not perpendicular to the calo layer surface
- Parameters obtained from a fit to the Geant4 single particle lateral shape in each calo layer, for each particle type, energy, |η|, shower depth bin
- Good average shower description, poor modelling of substructure variables



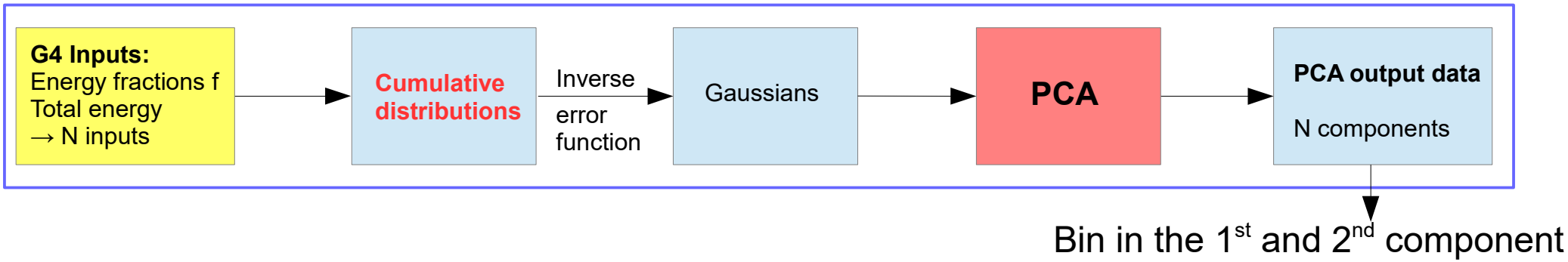
# New FastCaloSim: Energy Parametrisation



1. Step: Transform the correlated inputs into a set of uncorrelated ones
→ **Principal component analysis**

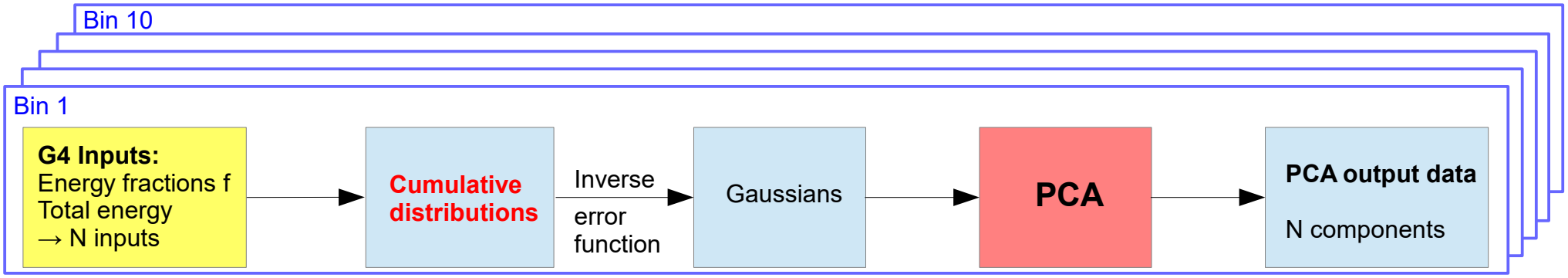
Inputs: Total energy deposited, and the fractions of the energy deposited in each calo layer
Typically 5-10 variables (electrons & photons impact less calo layers than charged pions)

1st PCA chain:




The 1st PCA is used only to derive a binning, then the procedure starts all over, in each bin:

2nd PCA chain:



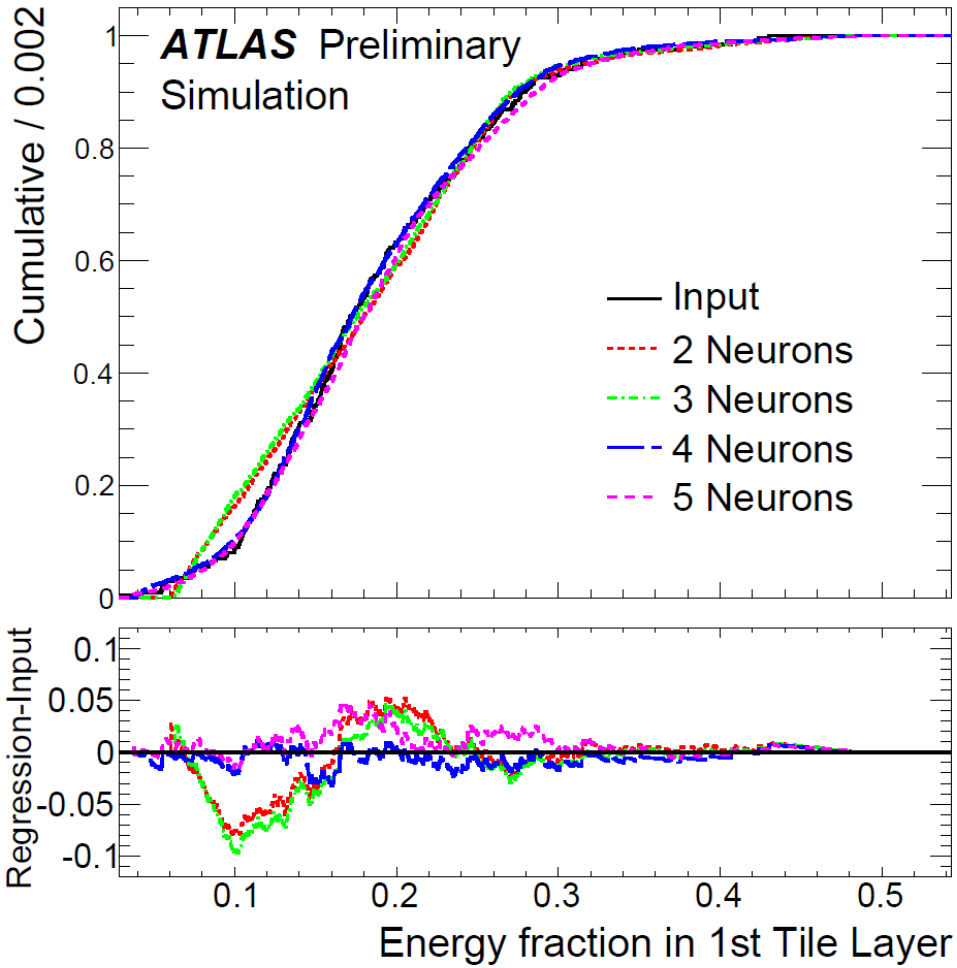

Output is a set of uncorrelated, Gaussian-shaped distributions
To store: Cumulative distributions, PCA matrices, mean and RMS of the output Gaussians

# New FastCaloSim: Energy Parametrisation



2. Step: **Storage optimisation**

- The cumulative energy distributions take a lot of memory space
- TMVA is used to perform a multivariate regression to approximate the functional form
  → Only the weights of the multilayer perceptron (MLP) method are stored
- The number of neurons of the MLP are optimized in an iteration, the less weights the less storage




Iteration stopped when good agreement between input and output is achieved

Number of weights stored:
1+n+(2*n)

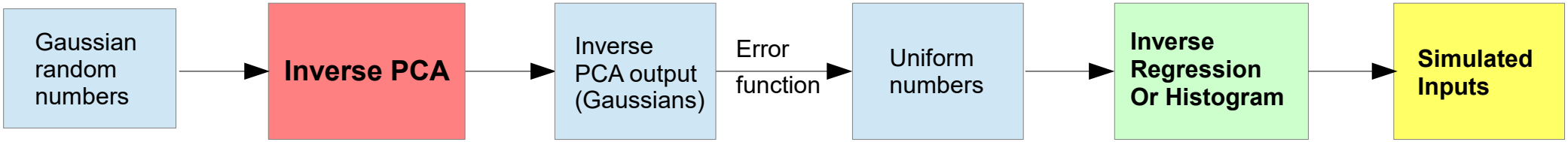
n = number of neurons

# New FastCaloSim: Energy Parametrisation

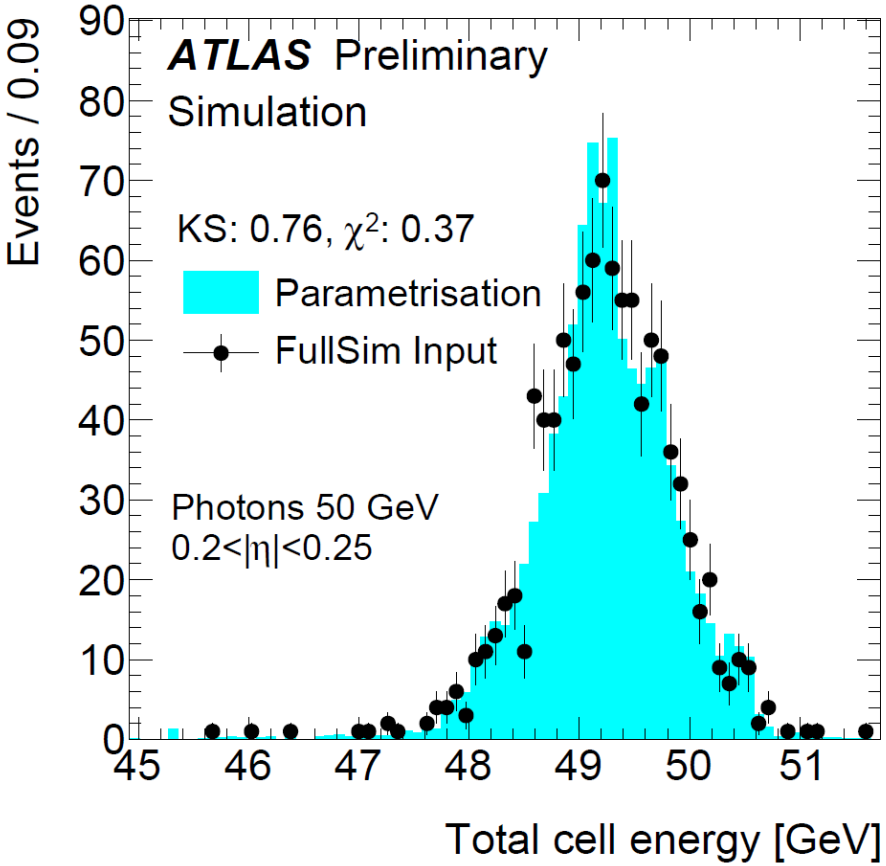


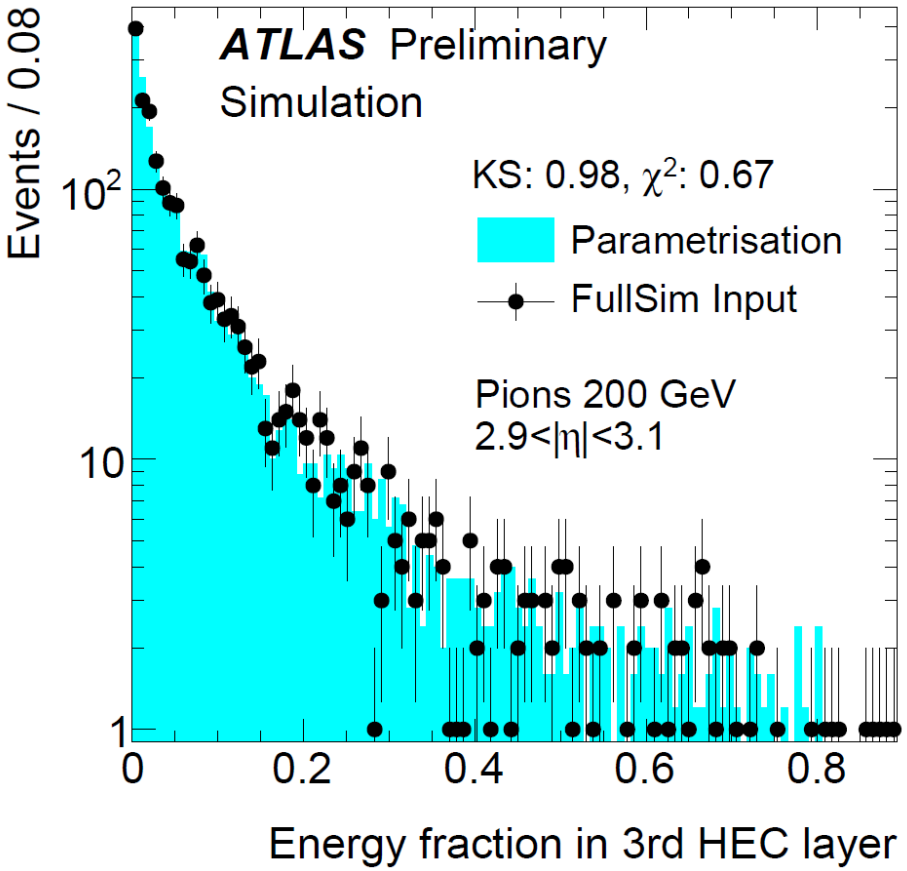
3. Step: **Simulation**

For each PCA bin:




Results from the various PCA bins are then summed together



# New FastCaloSim: Lateral Shower Shape

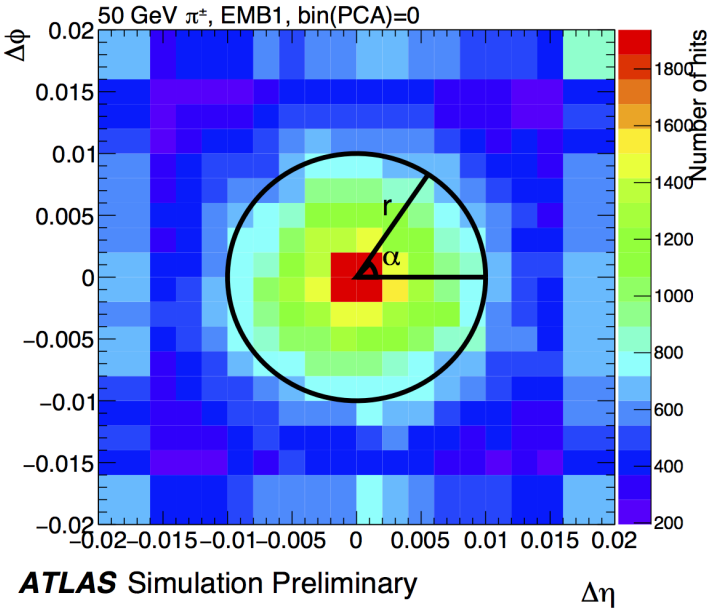


**Shower geometry:**

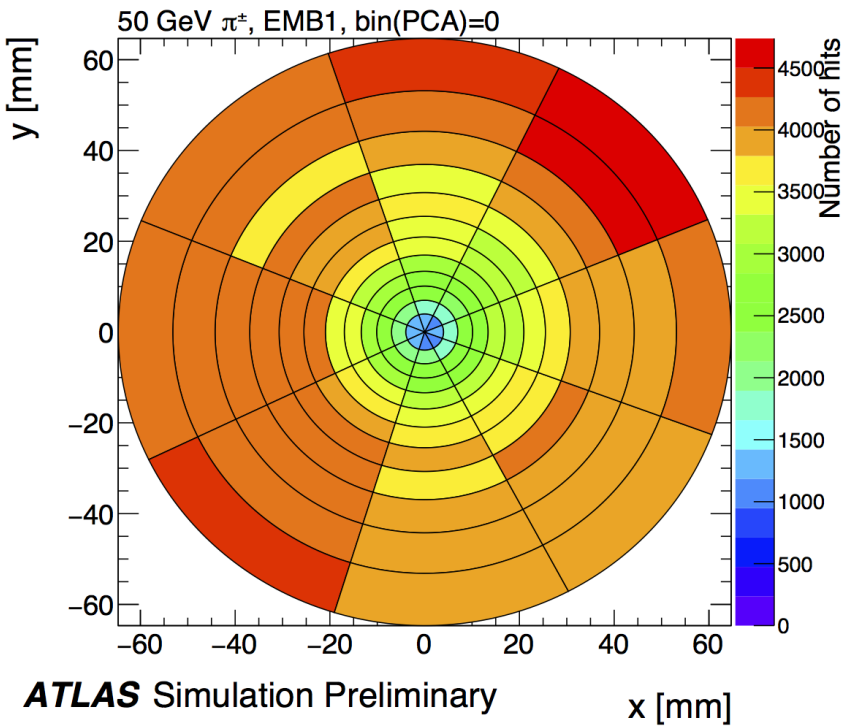
Radial distance to shower center r
Angle $\alpha$

$$r[\mathrm{mm}] = \sqrt{(\delta\eta[\mathrm{mm}])^2 + (\delta\phi[\mathrm{mm}])^2}$$

$$\alpha = \arctan(\delta\phi[\mathrm{mm}], \delta\eta[\mathrm{mm}]), \text{ defined between } [0, 2\pi]$$




1. Step: **Optimized input binning**




Optimize binning such that each bin contains about the same number of hits

8 bins in $\alpha$

Iterative procedure to find the number of bins in r

No empty bins allowed

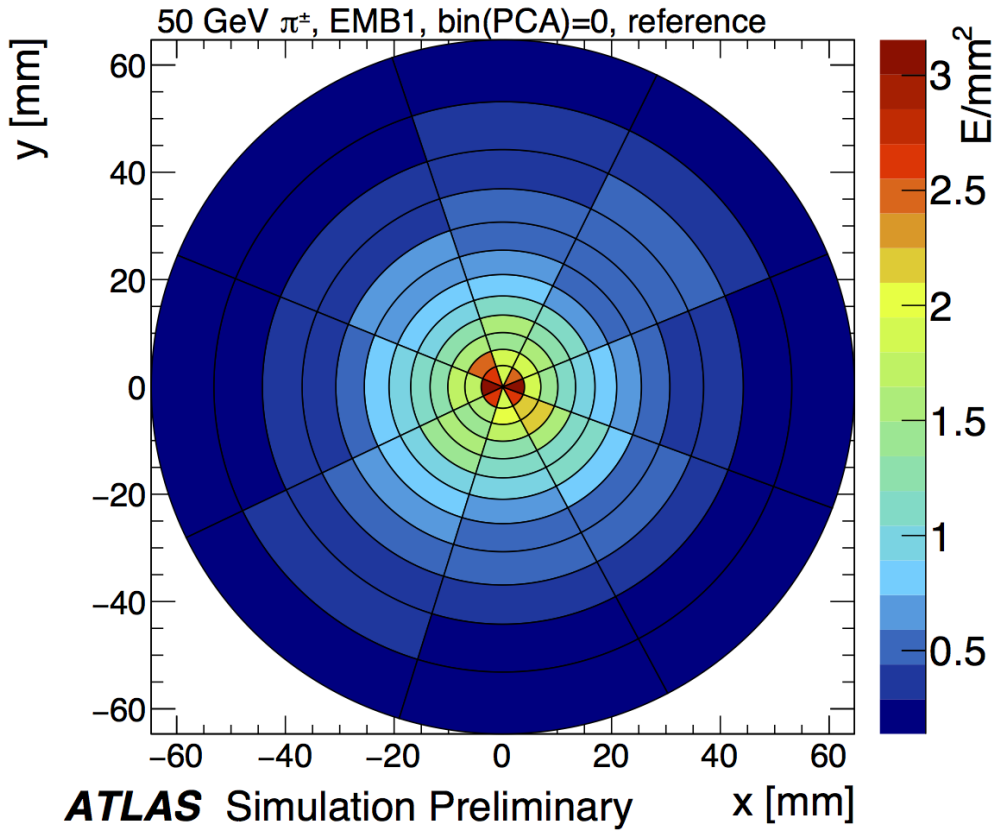
Finer bins in the shower center

# New FastCaloSim: Lateral Shower Shape

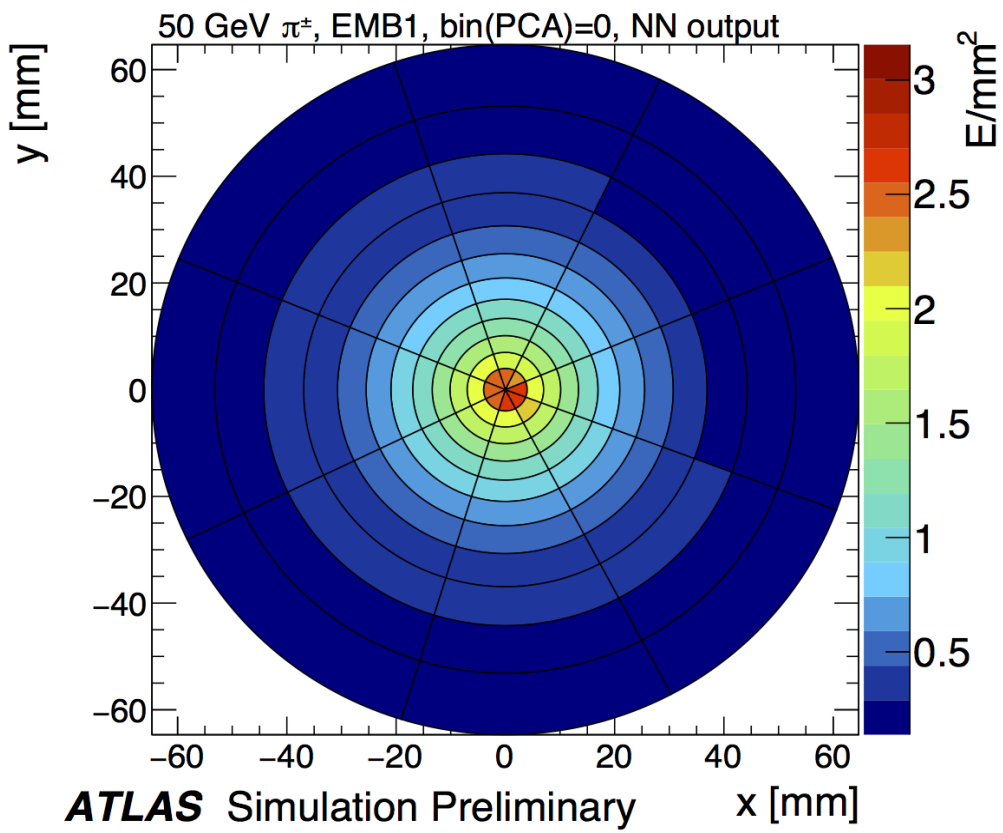


2. Step: **Neural network regression (MLP) of the energy density as a function of r and α**

Input




Output



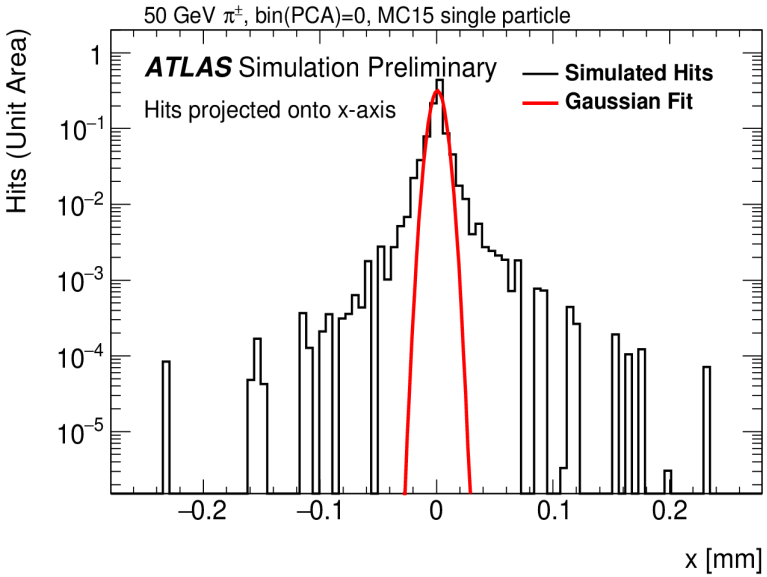

Some technical parameters (used in TMVA):

| Hidden Layers | Test rate | Neuron Type | Training method | Convergence Improve | Convergence Tests |
|---|---|---|---|---|---|
| 2-10 | 6 | tanh | BFGS | 1e-6 | 15 |

# New FastCaloSim: Simple Lateral Shower Shape

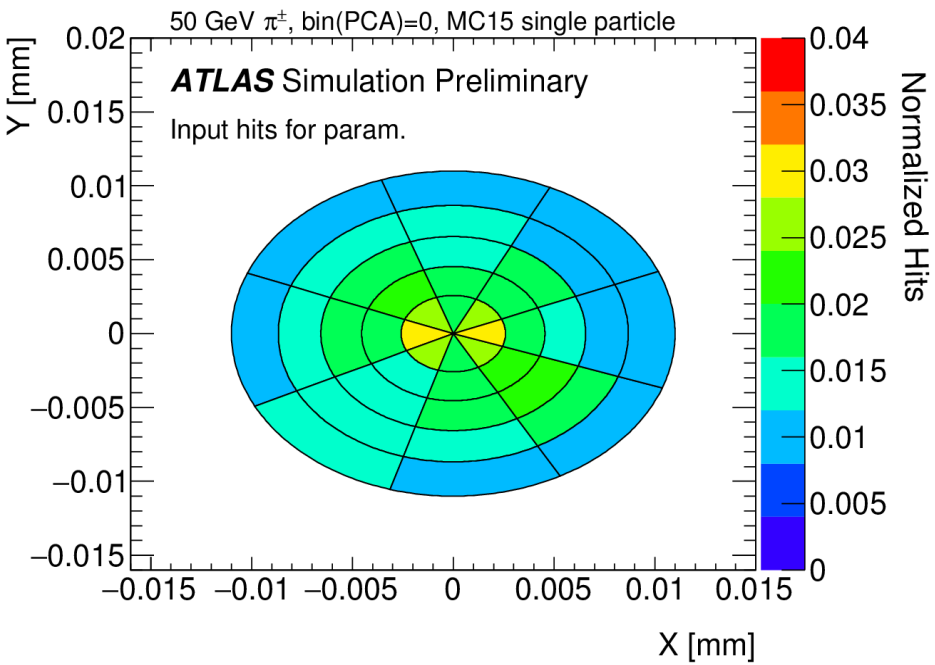


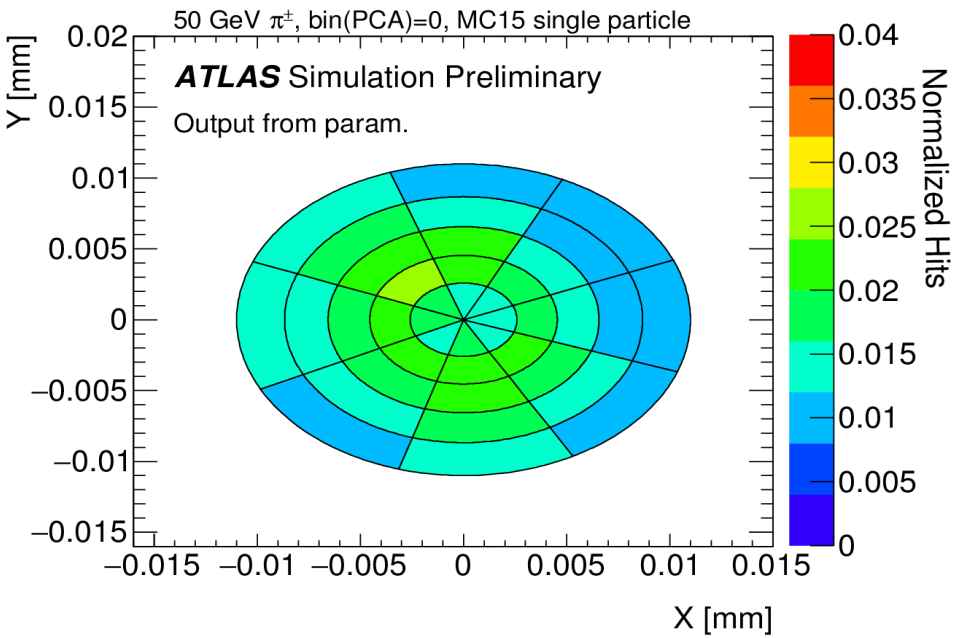
- Alternative to the neural-network fit based shower shape
- For the case of low energy particles, or when the NN fit fails




**Method:**

- Projection of the shower shape to two axes x and y
- Fit each 1D distribution with an analytic function, eg. simple Gaussian
- Generate random hits from the fit, in 2D
- Shower core well reproduced, tails not well described



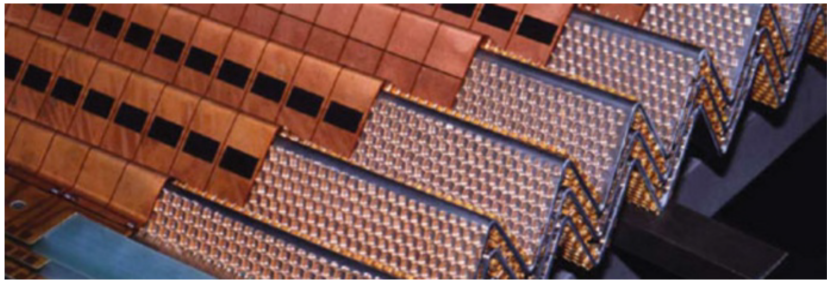
# New FastCaloSim: Energy Assignment to Cells



Discussed so far: - total energy and the energy in each calo layer
- the energy distribution in lateral direction

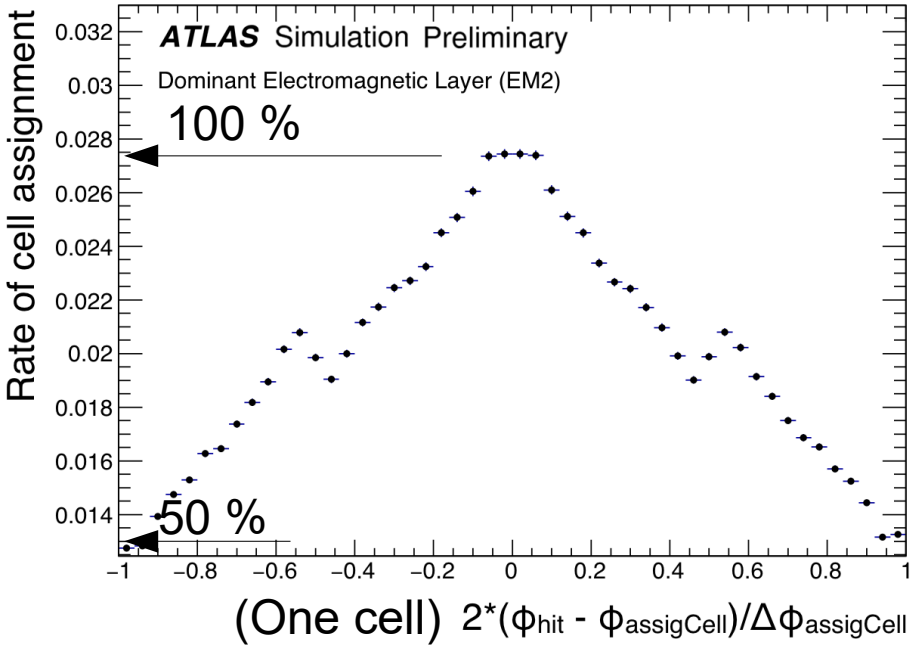
→ Now: Assign the energy to the actual calo cells using the right detector geometry

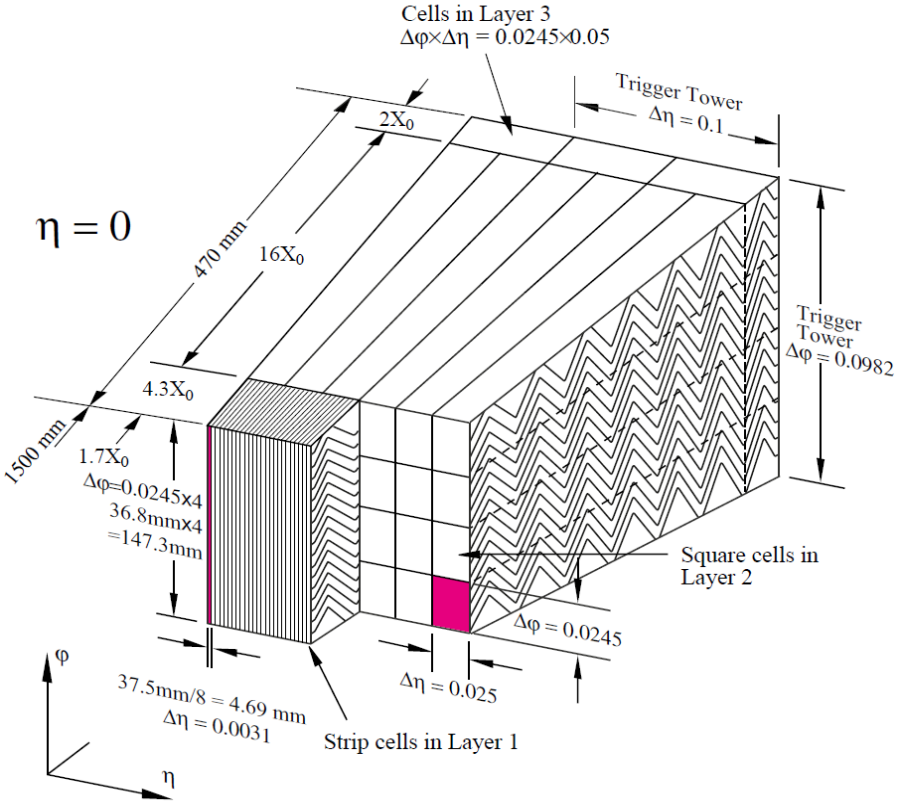
Difficulty: **Accordion shape** of the readout electrodes



FastCaloSim makes the simplification of taking cells as cuboids. In reality: Cells are defined by 4 electrodes

"Wiggle function" (hit displacement):




Probability function describing the chance that the energy belongs to this cell (if <1, some chance it belongs to a neighbour cell)

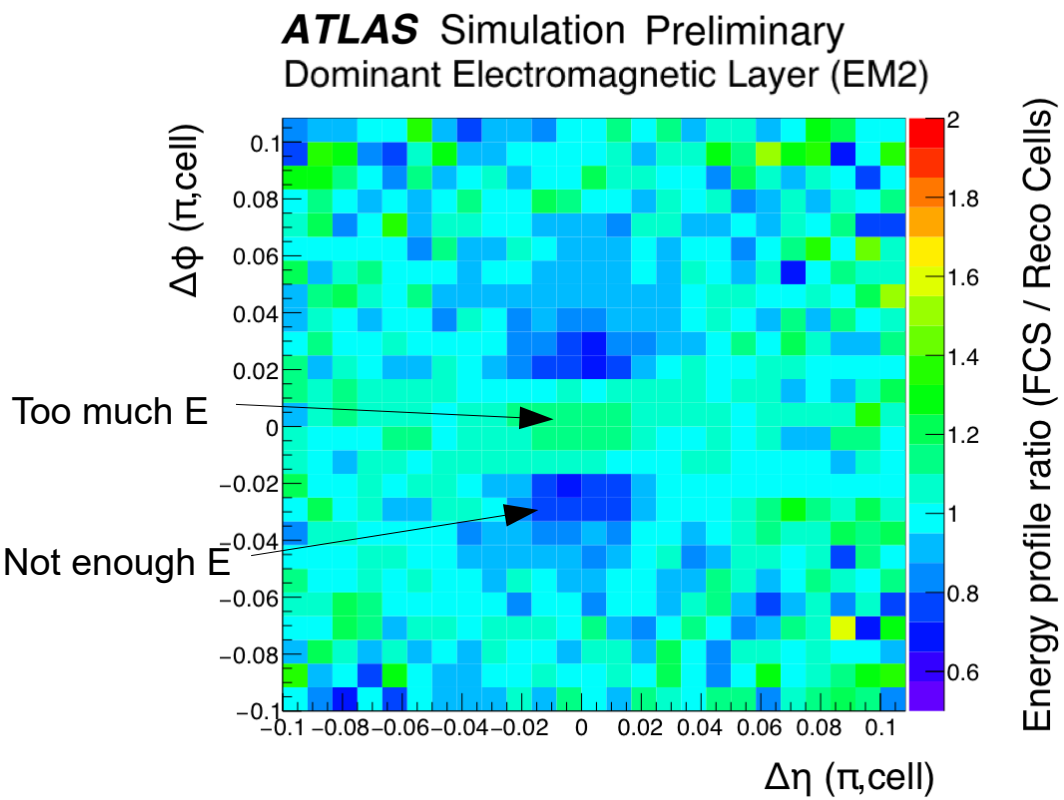
# New FastCaloSim: Energy Assignment to Cells

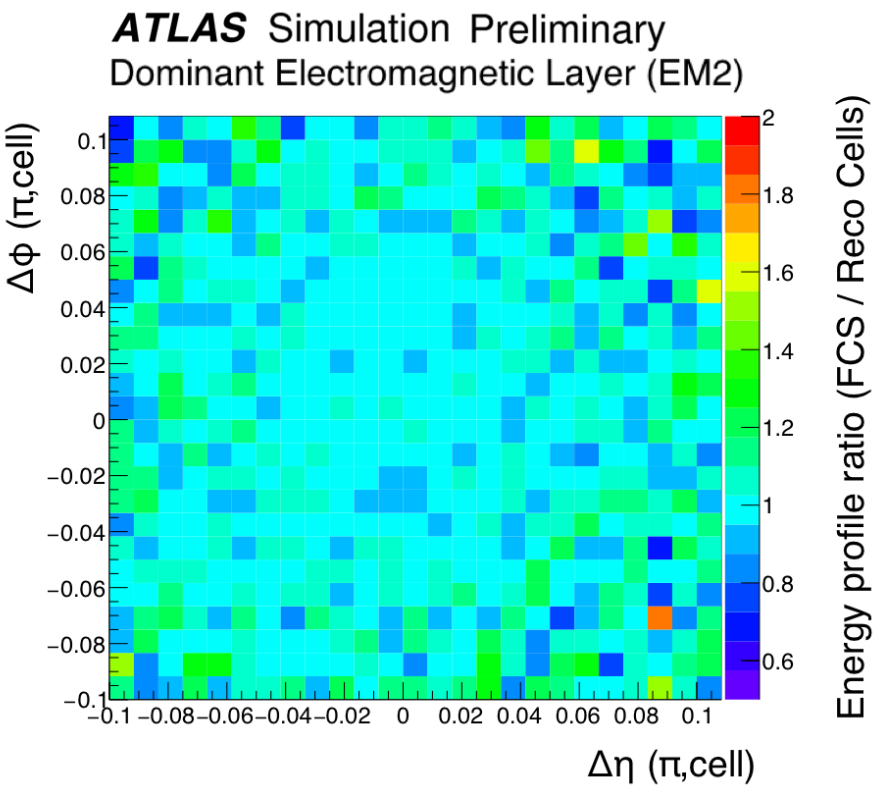


Ratio of energies as a function of distance of the hit from the pion shower center

Not considering the “wiggling”, structures in phi direction

Considering the “wiggle function” → Flat in phi direction




Reco cell: Geant4 cell
FCS cell: assigned cell

→ Good closure when compared to the Geant4 inputs

# New FastCaloSim: Prototype development



**Steps still in development:**

- Putting the components all together
  (Energy value, shape distribution, cell assignment)

- Integrating into the ATLAS software (Athena)

- Introducing parameters for tuning to data

- Validation

# Summary



- Physics and performance studies require huge samples of simulated MC events
- Geant4 simulation in the calorimeters takes a lot of time and resources
- FastCaloSim is a parametrized calorimeter response, widely used in ATLAS
- New FastCaloSim development was reported in this talk,
  the most important steps are implemented and in principle tested:

  - PCAs to decorrelated the inputs
  - Storage optimisation using neural-network techniques
  - Lateral shower shape: Optimized binning and neural network fit, simple functional fit
  - Output hit collection with correct cell assignment
- Prototype will be available soon

Backup