A Neural-Network Clusterisation Algorithm for the ATLAS Silicon Pixel Detector

Katharine Leney
on behalf of the ATLAS Collaboration

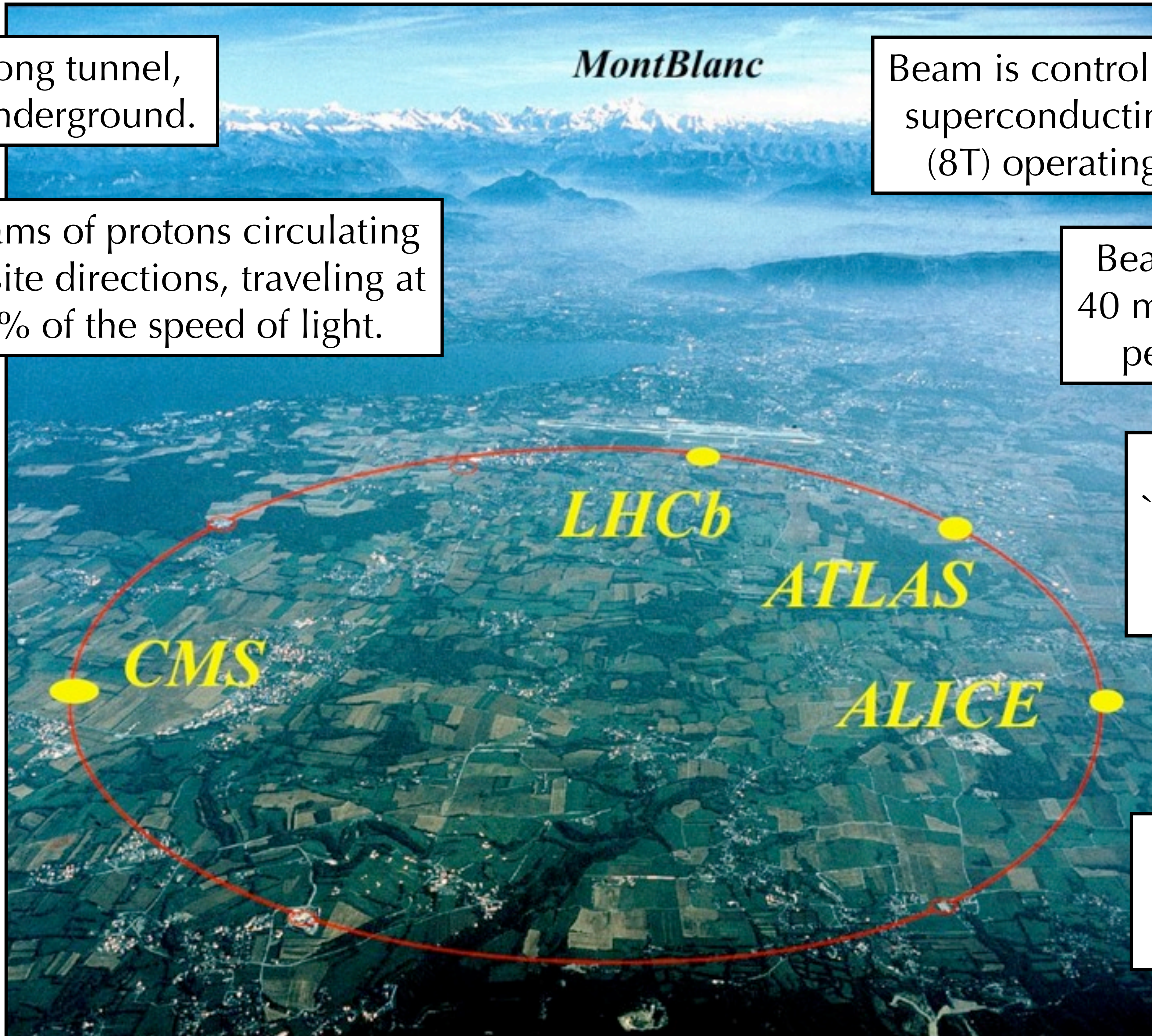
University of the Witwatersrand, South Africa

16th May 2013

CERN and the LHC

27km long tunnel, 100m underground.

Two beams of protons circulating in opposite directions, traveling at 99.99% of the speed of light.



Beam is controlled by 1800 superconducting magnets (8T) operating at 1.9 K.

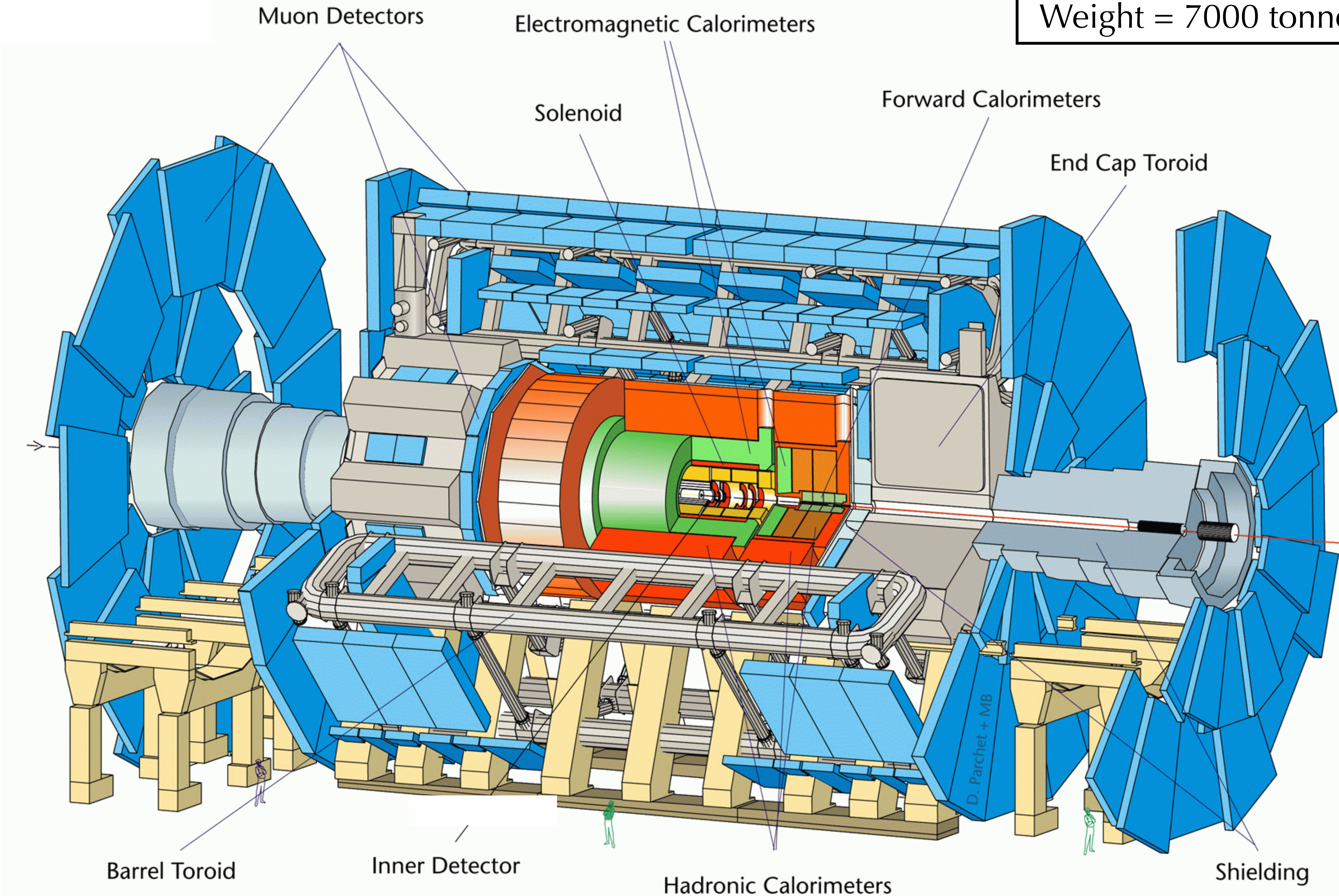
Beams collide 40 million times per second

3000 `bunches' of protons per beam.

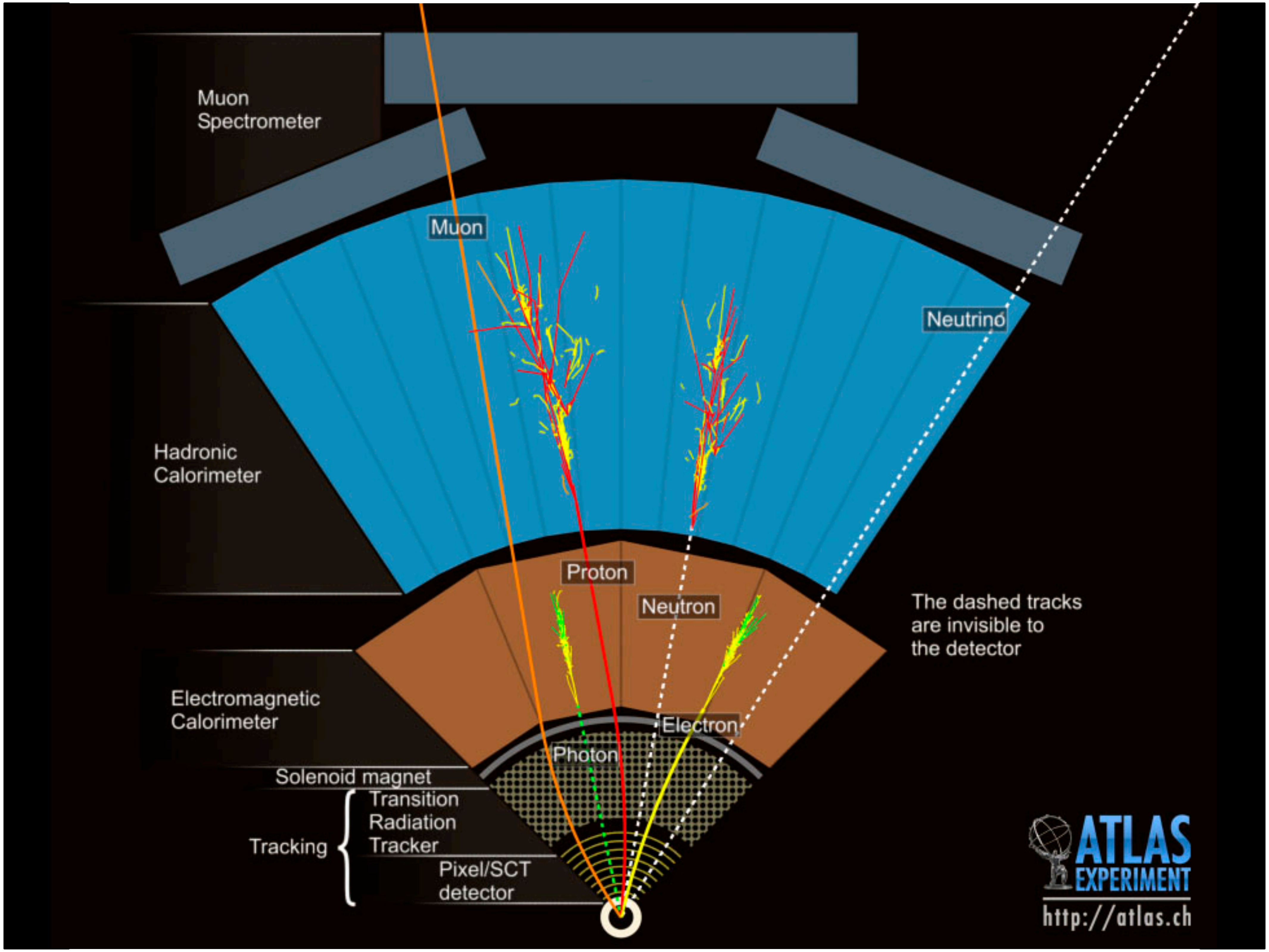
200 billion protons per bunch.

The ATLAS Detector

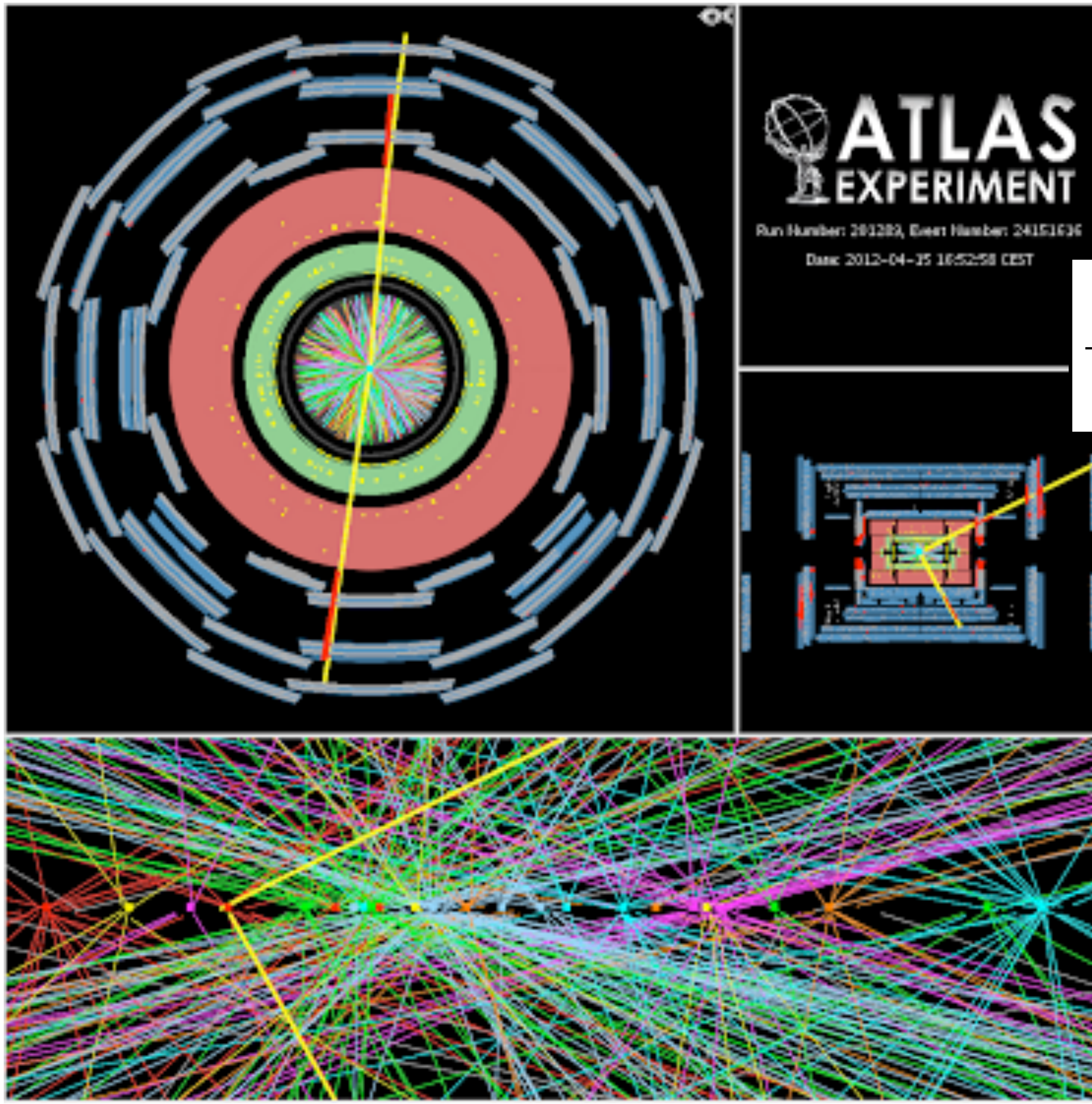
Diameter = 25m
Length = 45m
Weight = 7000 tonnes



The ATLAS Detector

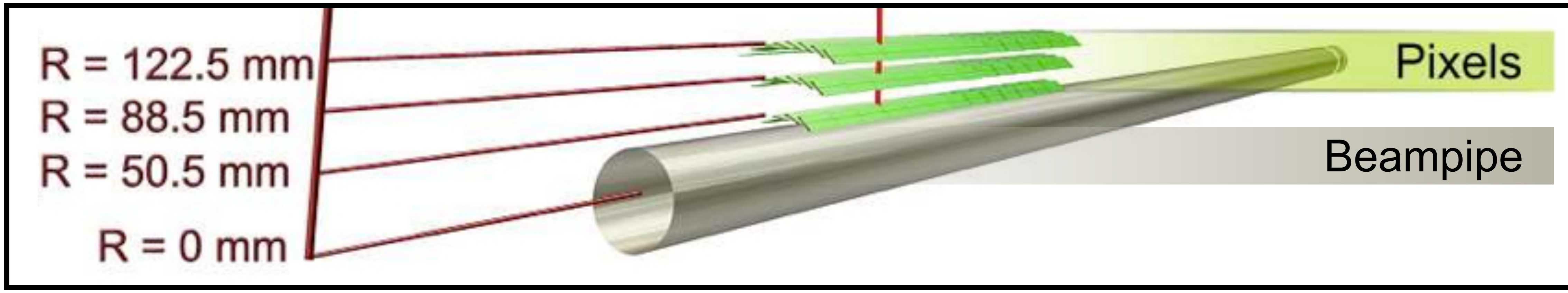


Data Events

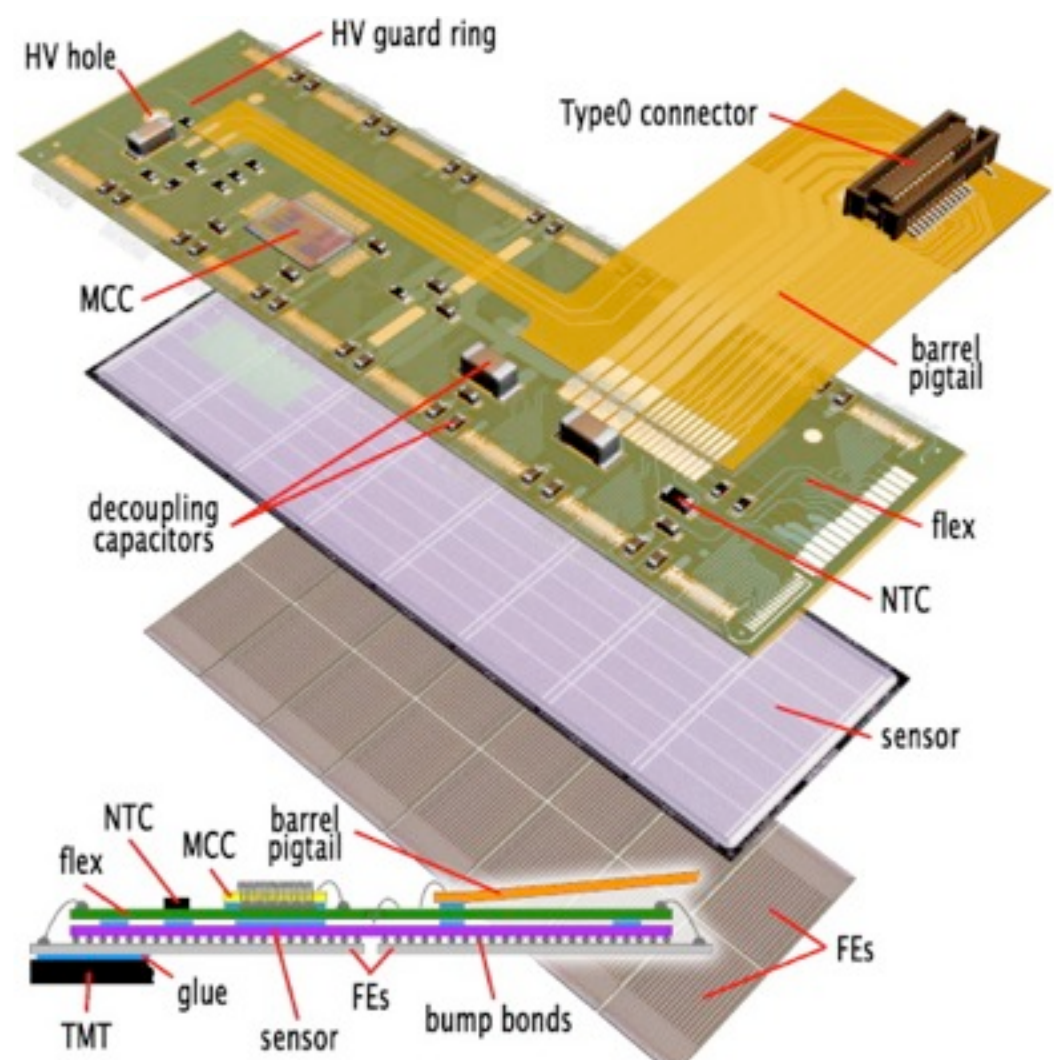


Z→ μμ event
+ 24 other p-p
interactions

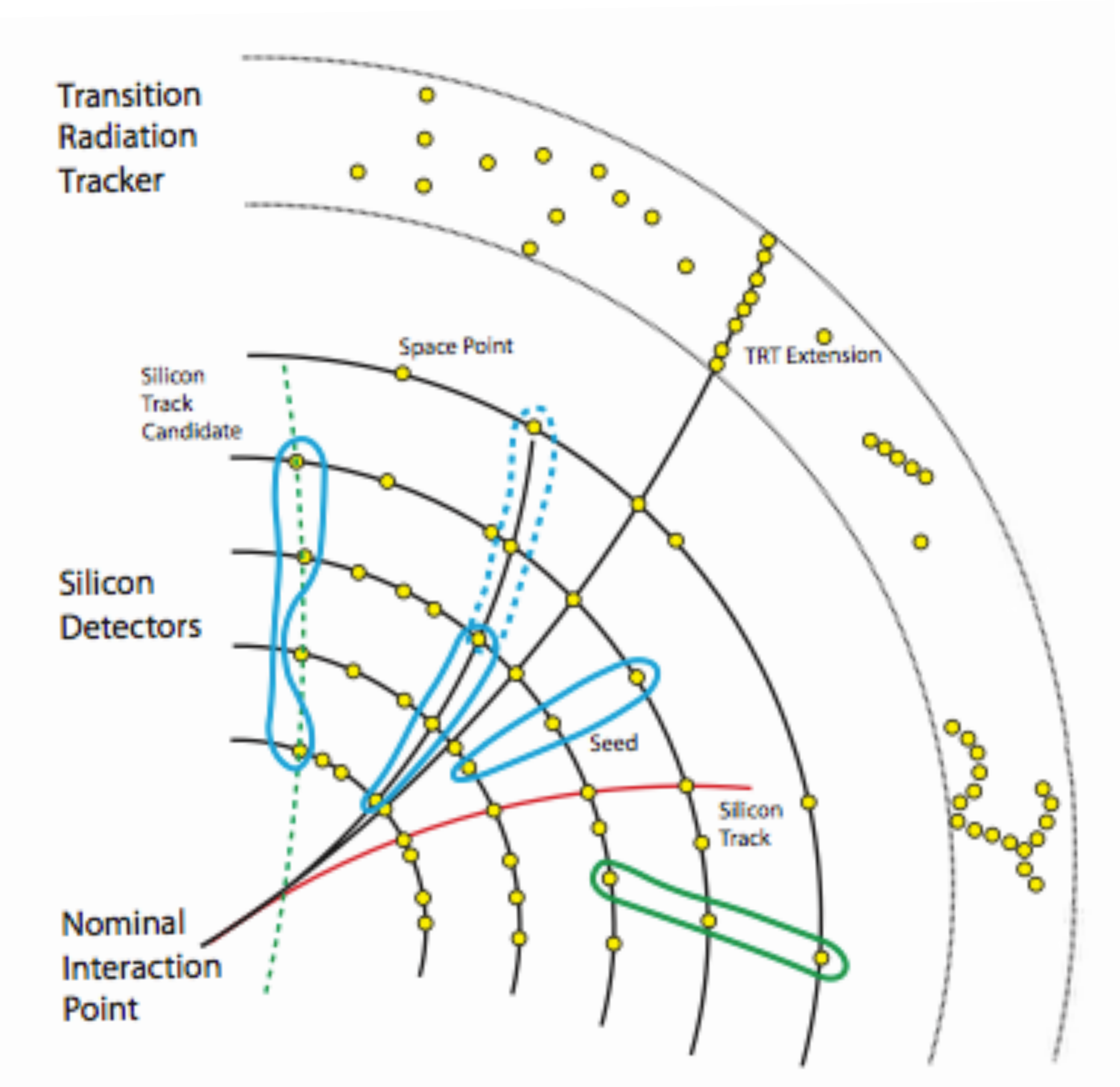
Pixel Detector



- Innermost layer of ATLAS detector.
- Provides precision measurements of positions of charged particles.
- Crucial for identification of long-lived particles via reconstruction of secondary vertices and estimate of number of proton-proton interactions per bunch crossing.
- 1744 identical modules.
- 46080 silicon pixel sensors per module.
  - Thickness: 250 μm
  - Transverse length: 50 μm
  - Longitudinal length: 400 μm
- Total of 80.8 million read-out channels.



Tracking Overview



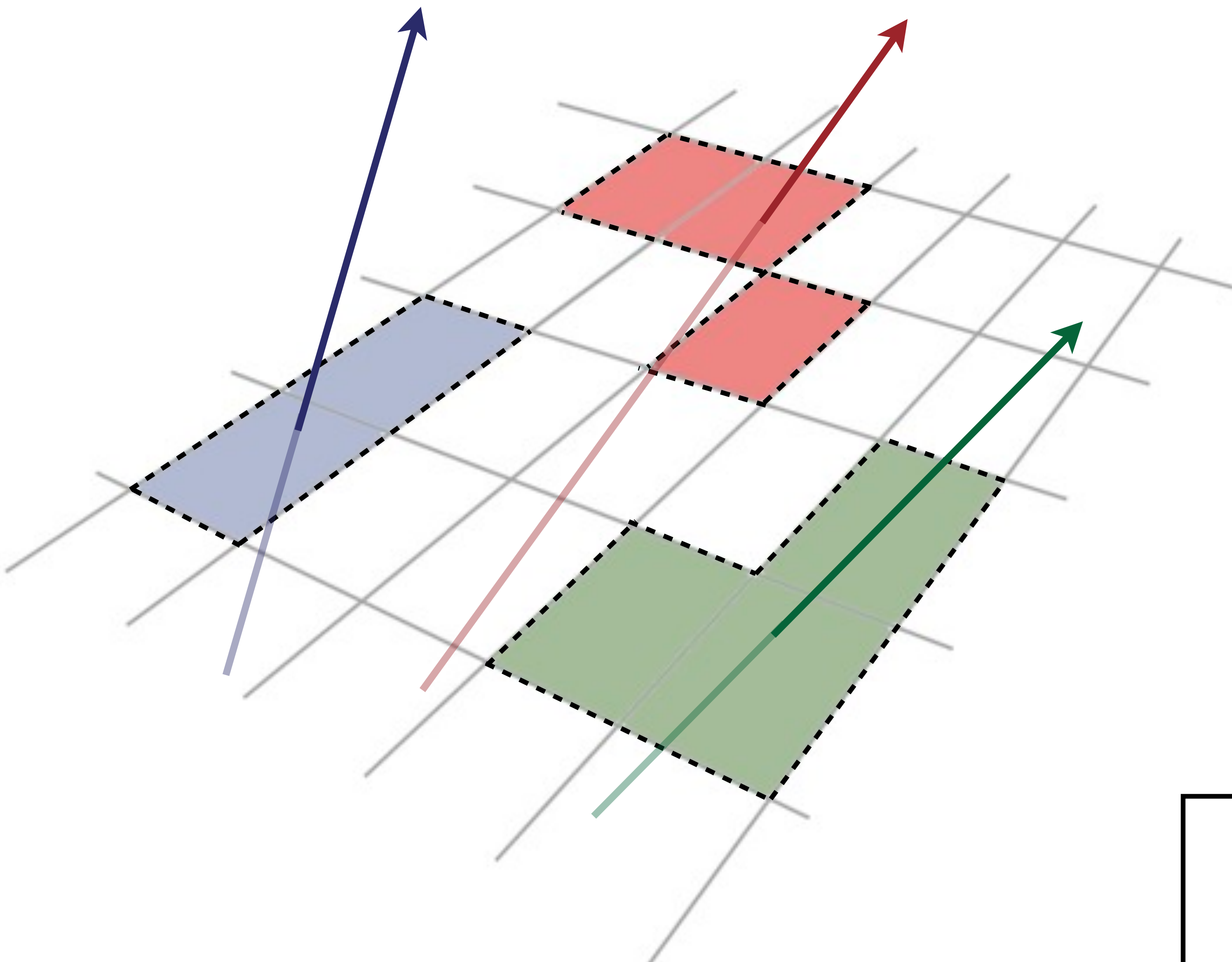
Track Pattern Recognition

- Start with blue “seeds” from 3 space point hits in silicon
- Use seed to build track candidate from inside out
- Use ambiguity solver to keep only most probable tracks
- Silicon track candidate connected to TRT extension

Track Fitting

- Fit track trajectory to collection of hits in track

Standard Clustering



- Particle traversing detector typically deposits charge in more than one pixel.
- Charge deposited in a pixel measured using pulse-height time-over-threshold.
- Pixels with deposited charge are grouped into clusters if they have a common edge or a common corner.

- Position of crossing is computed from the signal heights inside the cluster of pixels:

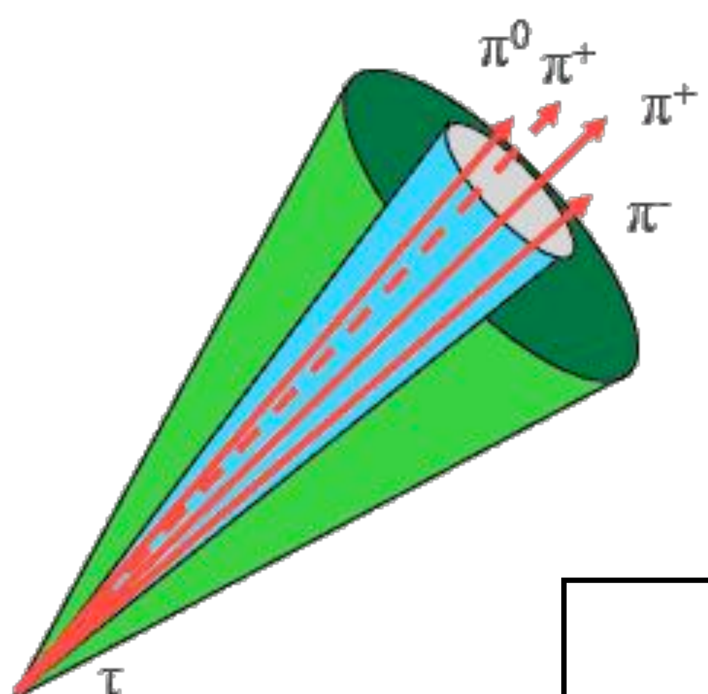
$$x_{cs} = x_{\text{center}} + \Delta_x \cdot \left(\Omega_x - \frac{1}{2}\right)$$

$$y_{cs} = y_{\text{center}} + \Delta_y \cdot \left(\Omega_y - \frac{1}{2}\right)$$

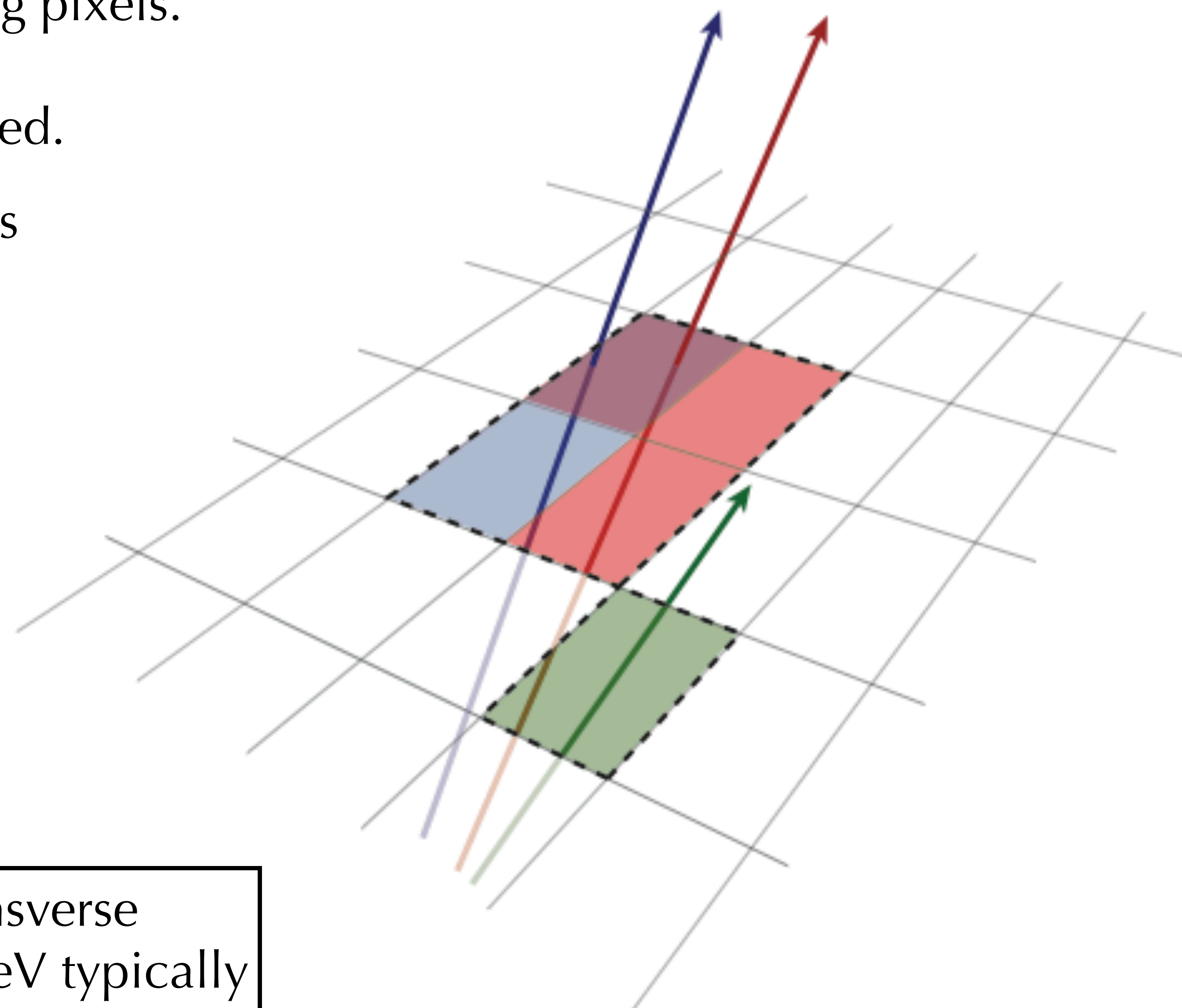
$$\Omega_{x(y)} = \frac{q_{\text{last row(col)}}}{q_{\text{first row(col)}} + q_{\text{last row(col)}}}$$

Tracking in Dense Environments

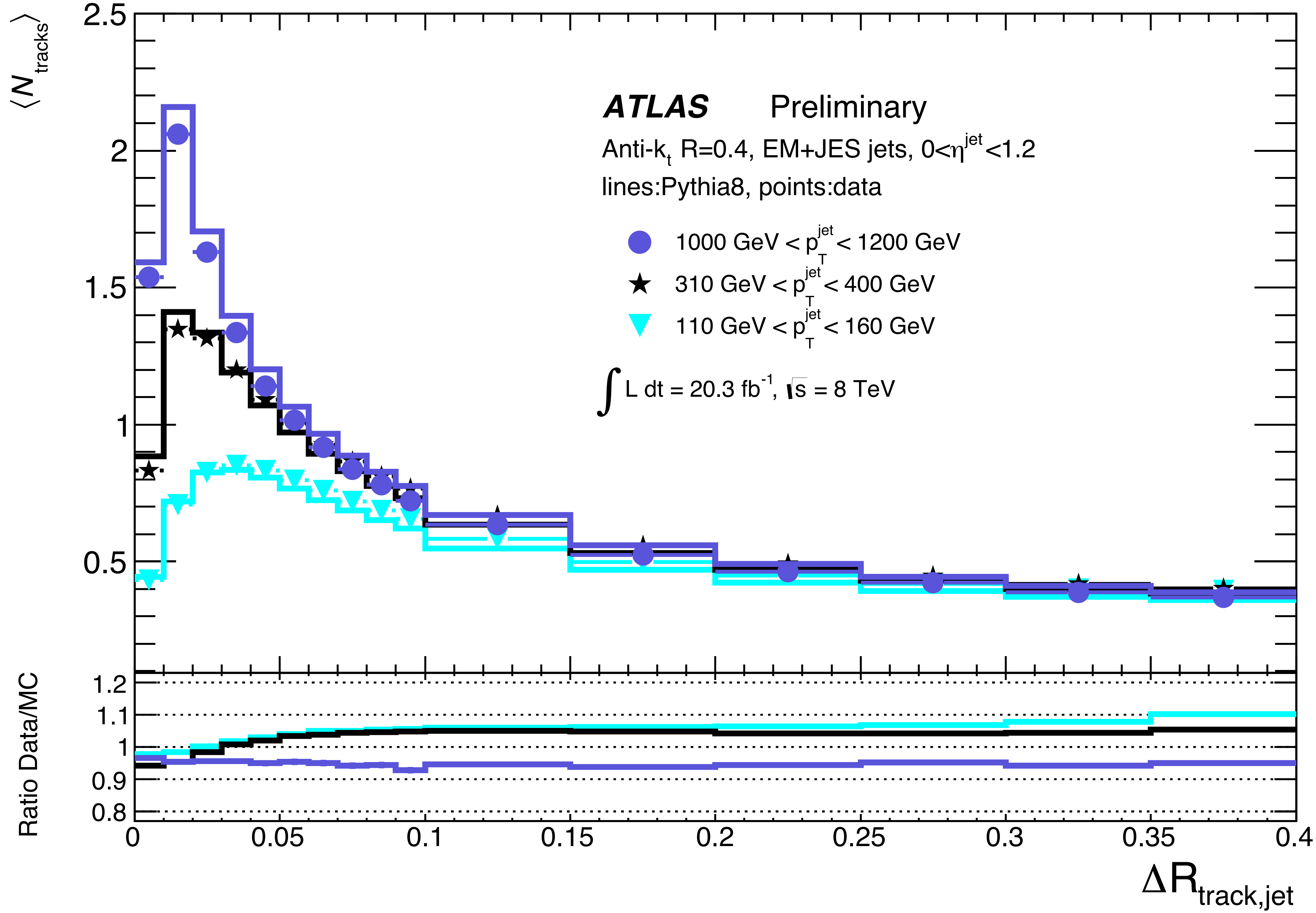
- Standard clustering provides excellent resolution for most clusters.
- Inadequate for dense environments with multiple charged particles:
  - Charge deposited in neighbouring pixels.
  - Clusters are shared.
  - Track parameters are mis-estimated.
- Objects such as highly energetic jets and hadronically decaying tau leptons are most affected.



Jets with transverse momentum > 1 TeV typically produce merged clusters.

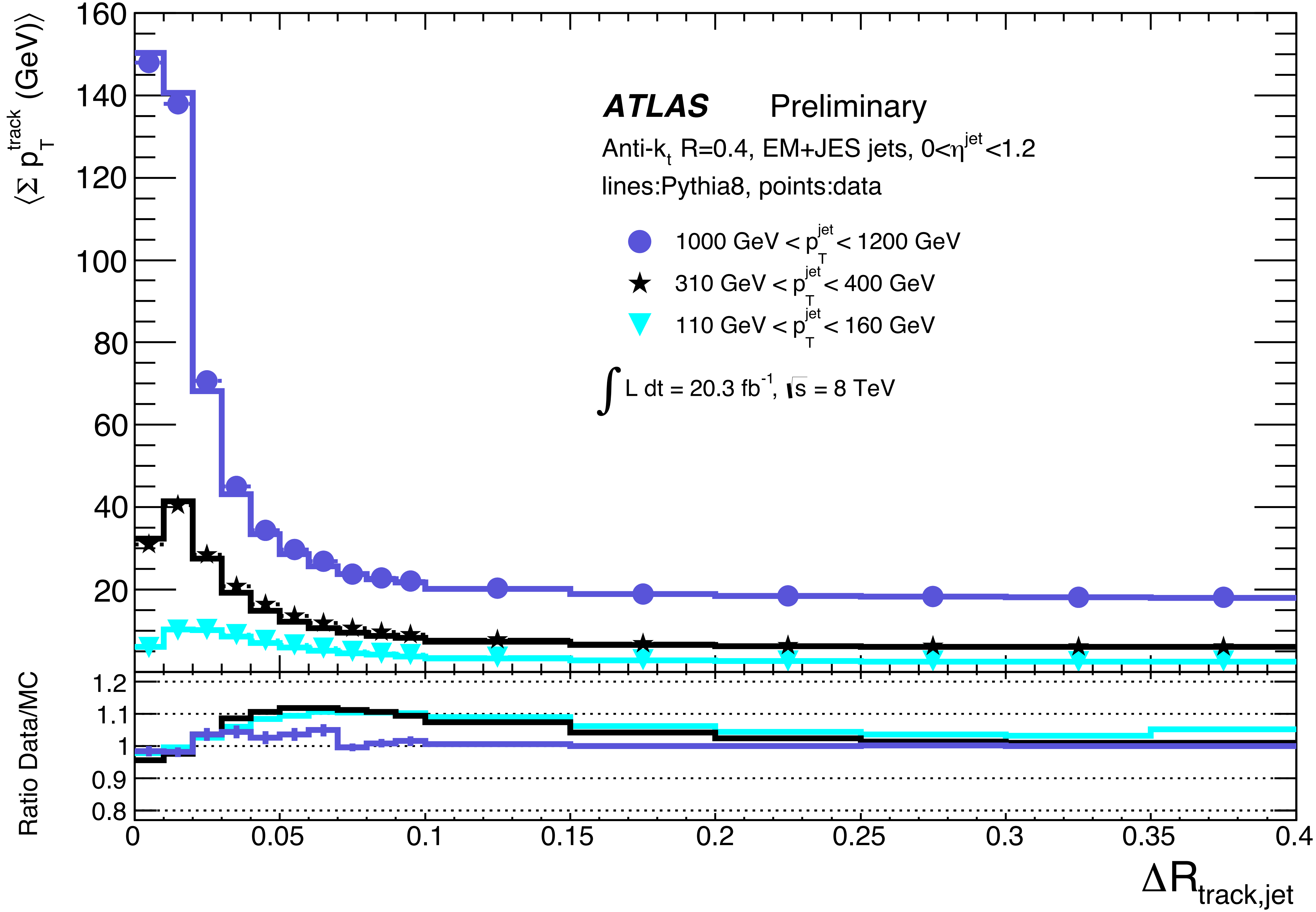


Tracking in Dense Environments



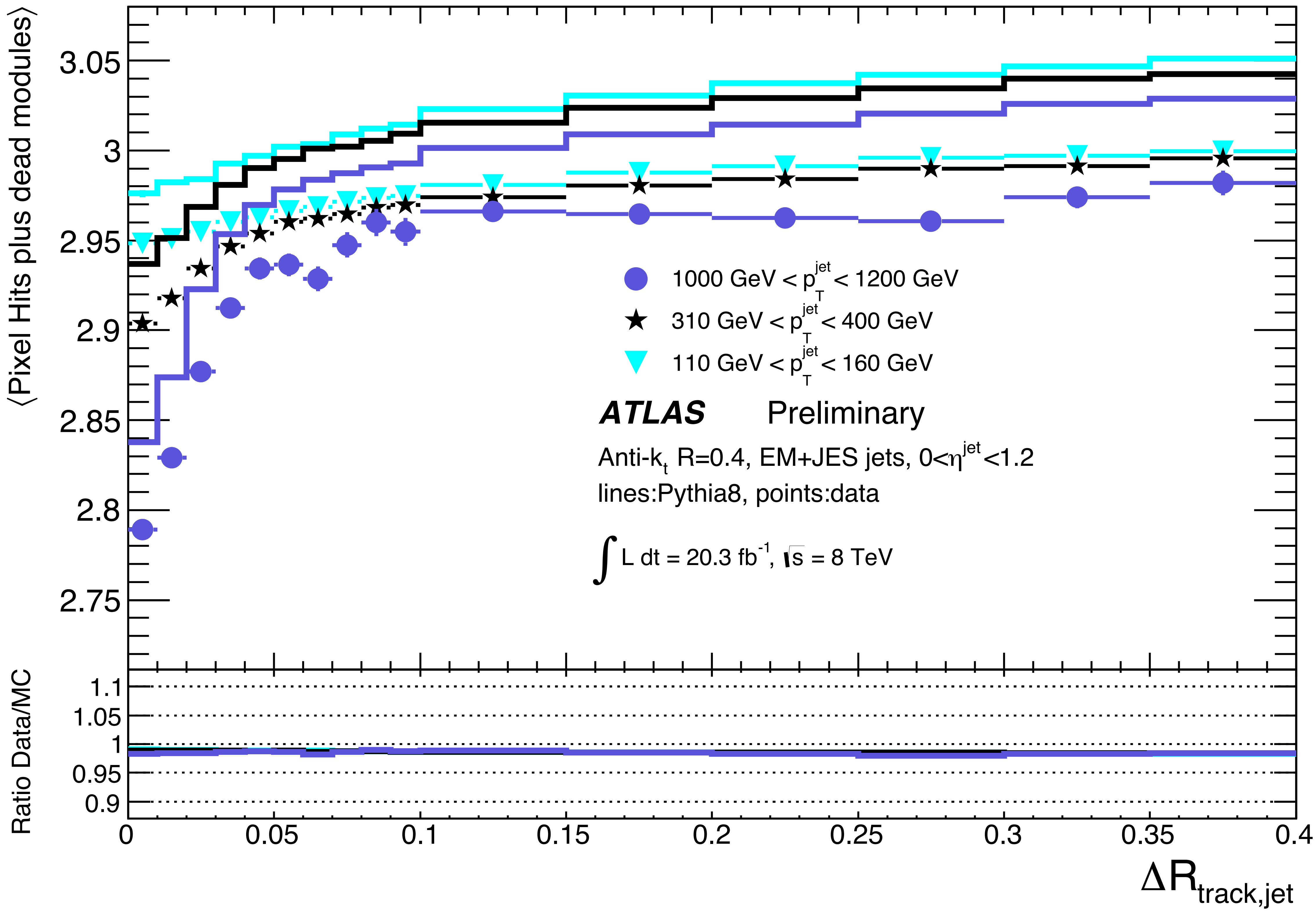
More energetic jets → more tracks in core.

Tracking in Dense Environments



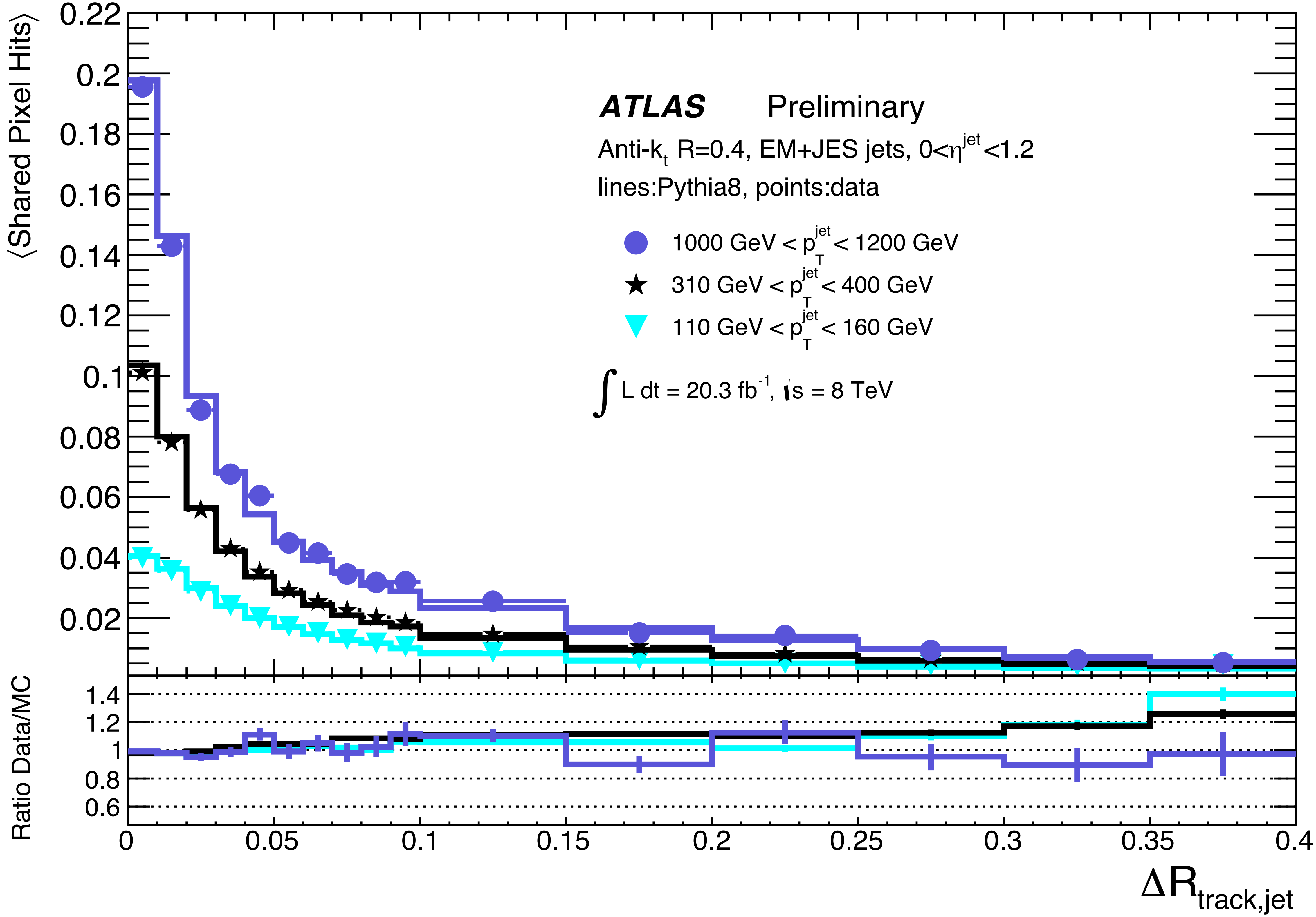
Most energetic tracks in jet core.

Tracking in Dense Environments



Tracks in core have fewer hits associated to them.

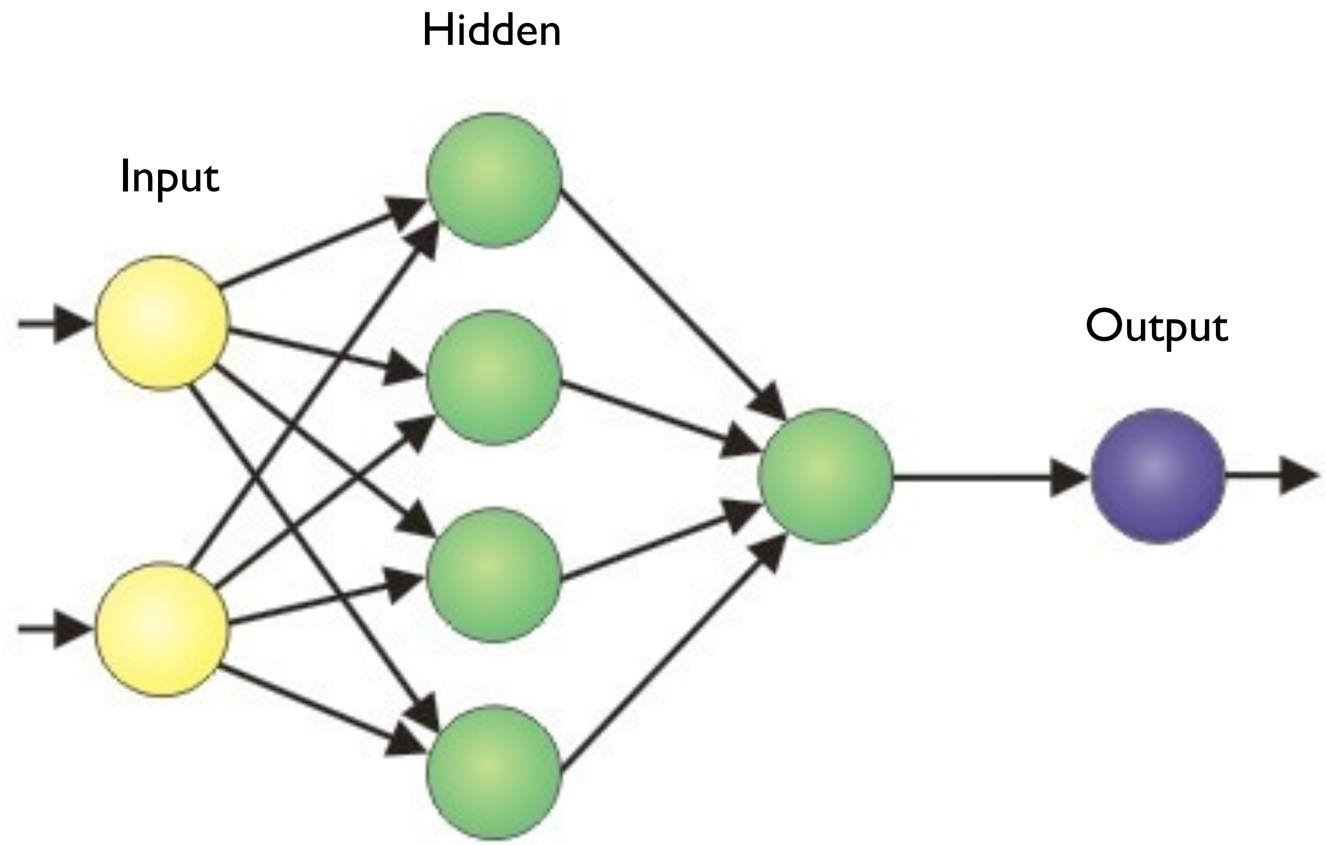
Tracking in Dense Environments



Energetic tracks in jet core share more hits with neighbouring tracks.

Neural Networks

- Powerful tools for pattern recognition problems.
- Can handle non-linear correlations between input variables.
- Attractive for problems with many degrees-of-freedom.
- Inputs are differently weighted in the hidden layers of the NN to finally determine the output.



Good choice for pixel clustering algorithm:

- Many cluster properties are nearly meaningless when alone (e.g. charge of a single pixel).
- Combine cluster properties to put into context (e.g. knowing charges of adjacent pixels).
- Variables then contain all information required for successful pattern recognition.

Neural Network Cluster Splitter

Feed-forward multi-layer perceptron network:

weight parameters

threshold parameters

$$F_i\left(\vec{x}\right) = h\left(\sum_j \omega_{ij} g\left(\sum_k \omega_{jk} x_k + \theta_j\right) + \theta_i\right)$$

output nodes

activation functions for computing values of intermediate and output nodes

input nodes/variables (k = [0, N_{inputs}]

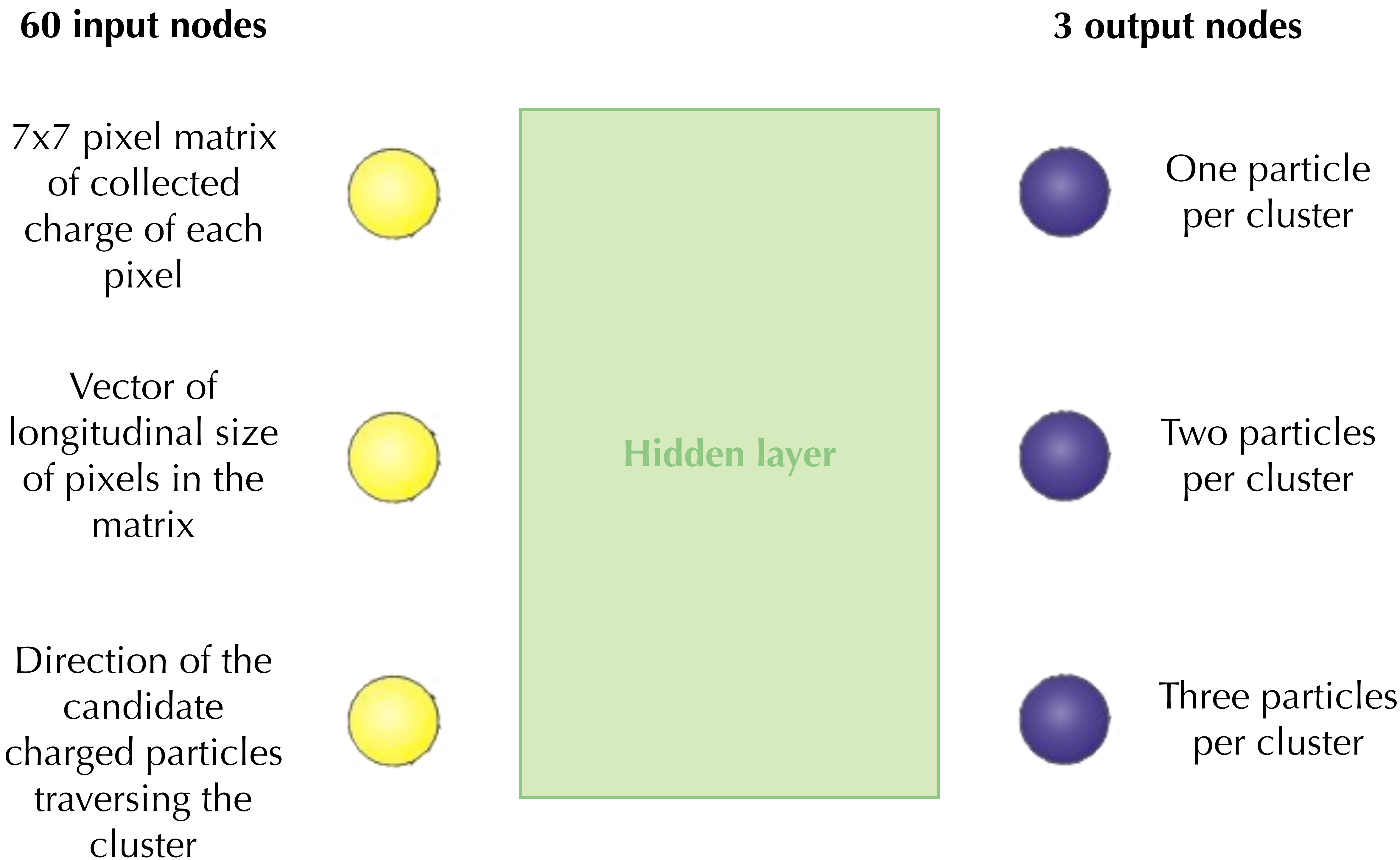
$$g\left(x\right) = \left(1 + \exp^{-2x}\right)^{-1}$$

output nodes are confined between 0 and 1

$$h(x) = x$$

Neural networks used to compute:
- Number of particles per cluster.
- Cluster position and error.

Number of Particles Per Cluster



Cluster Position & Error

Additional set of neural networks used to estimate:

Cluster position:

- Configured for interpolation.
  - Obtain a function where outputs get as close as possible to one or more continuous target variables.
  - Exploit dependence of such targets on the input variables.
- Different neural networks for different number of particles scenarios.
  - Trained on true number of particles in simulation.
- Same input variables as for classification neural network.

Probability density function for residual of estimated impact point:

- $\Delta\vec{P} = \vec{P} - \vec{P}_{true}$
- Separate neural networks for transverse and longitudinal directions.
- Translated to the cluster rest frame.

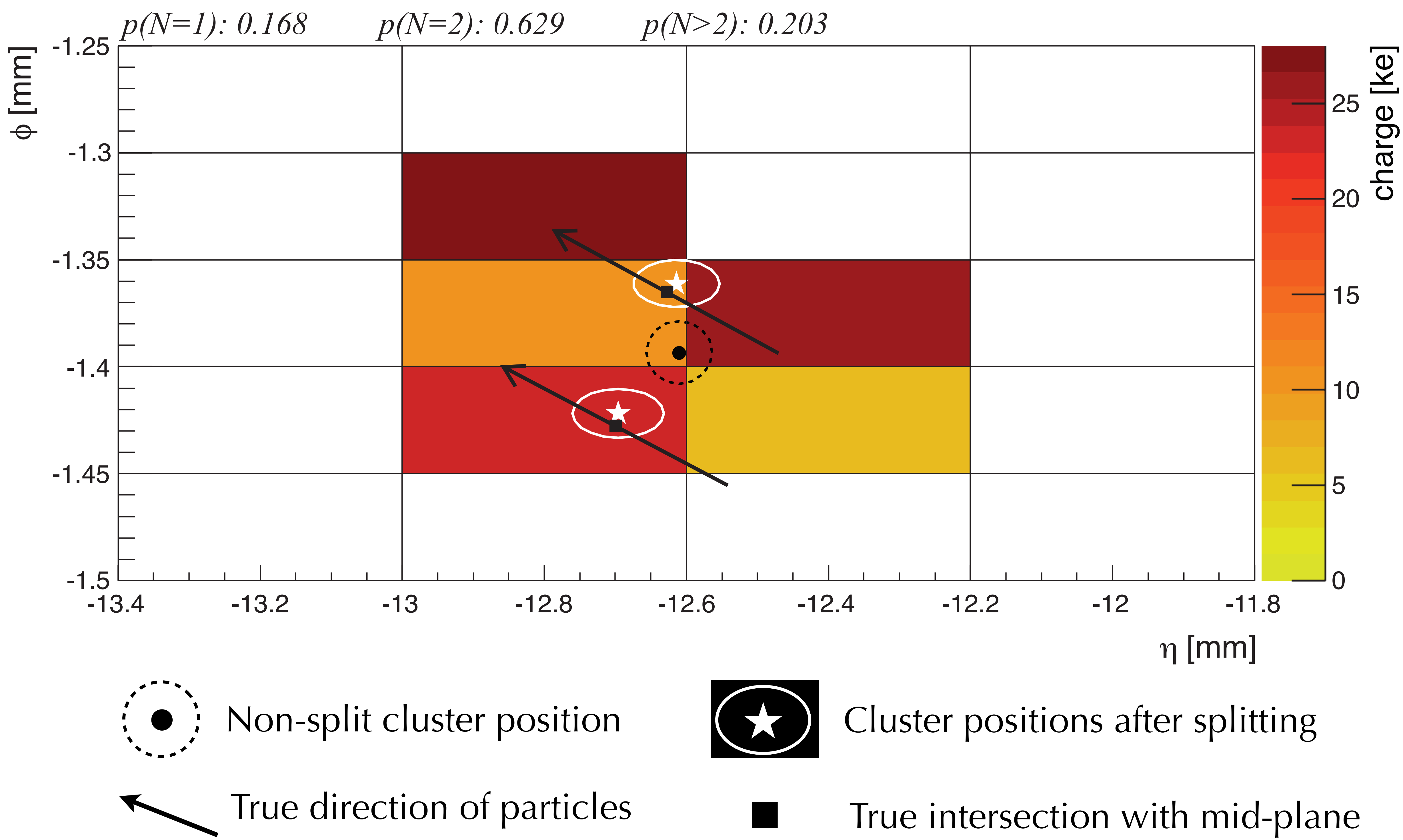
Training

- Ten neural networks needed for up to three sub-clusters:

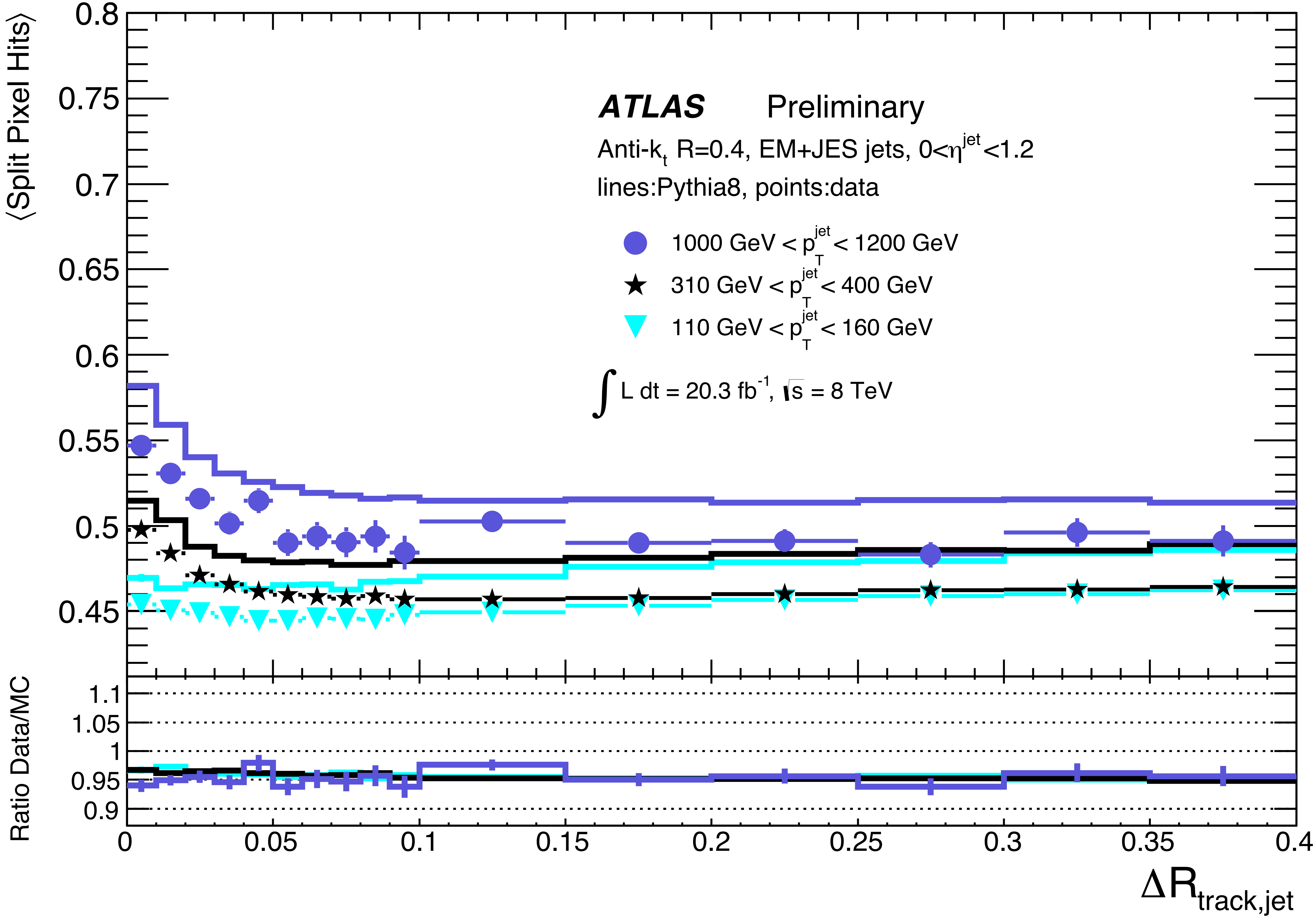
Number of charged particles traversing cluster	Particle 1 position	Particle 2 position	Particle 3 position
	Particle 1 x-error	Particle 2 x-error	Particle 3 x-error
	Particle 1 y-error	Particle 2 y-error	Particle 3 y-error

- Trained on simulations of pair produced top-quarks, and highly energetic di-jet events.
- Simulations divided into test samples and training samples.
  - Number of training patterns exceeds number of network parameter by at least 1000.

Cluster Splitting



Split Clusters

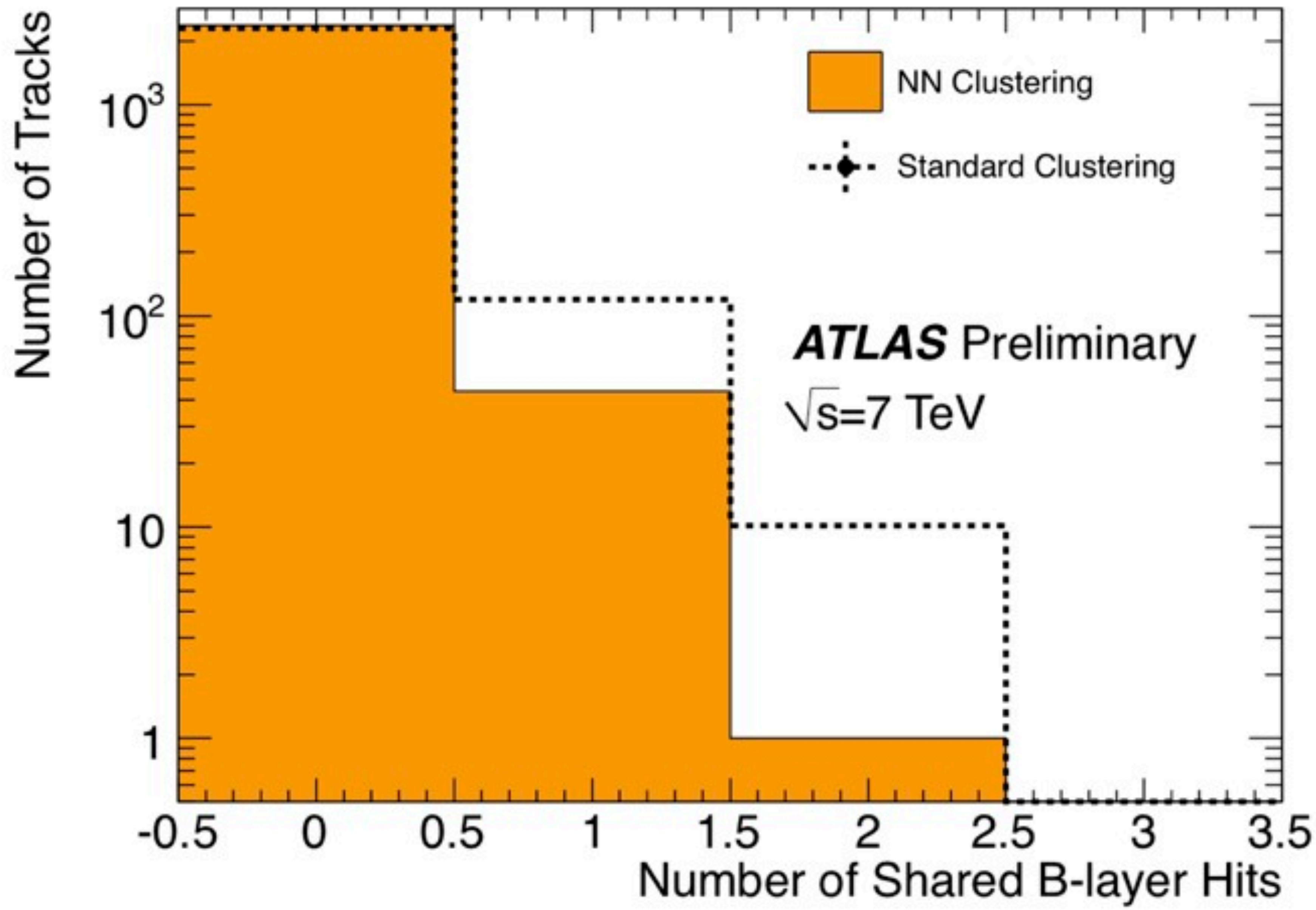


Two-Particle Separation

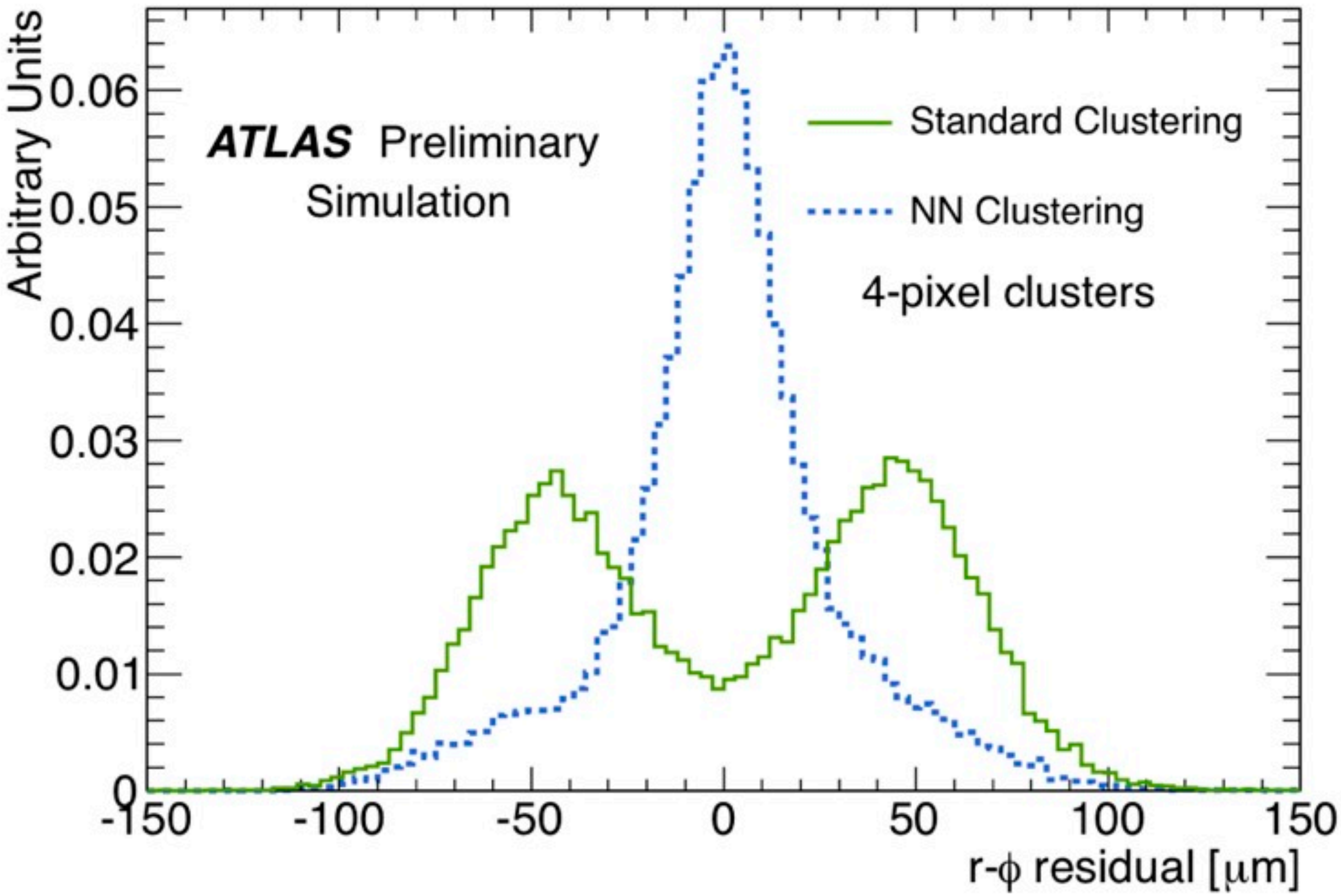
- Track is allowed to share a pixel cluster with another track only if cluster is not already split, and the neural network output is compatible with a possible merged cluster.

- Most noticeable improvements in innermost layer of pixel detector (b-layer) where particle density is highest.

- Ambiguities reduced by order of magnitude when using neural network.

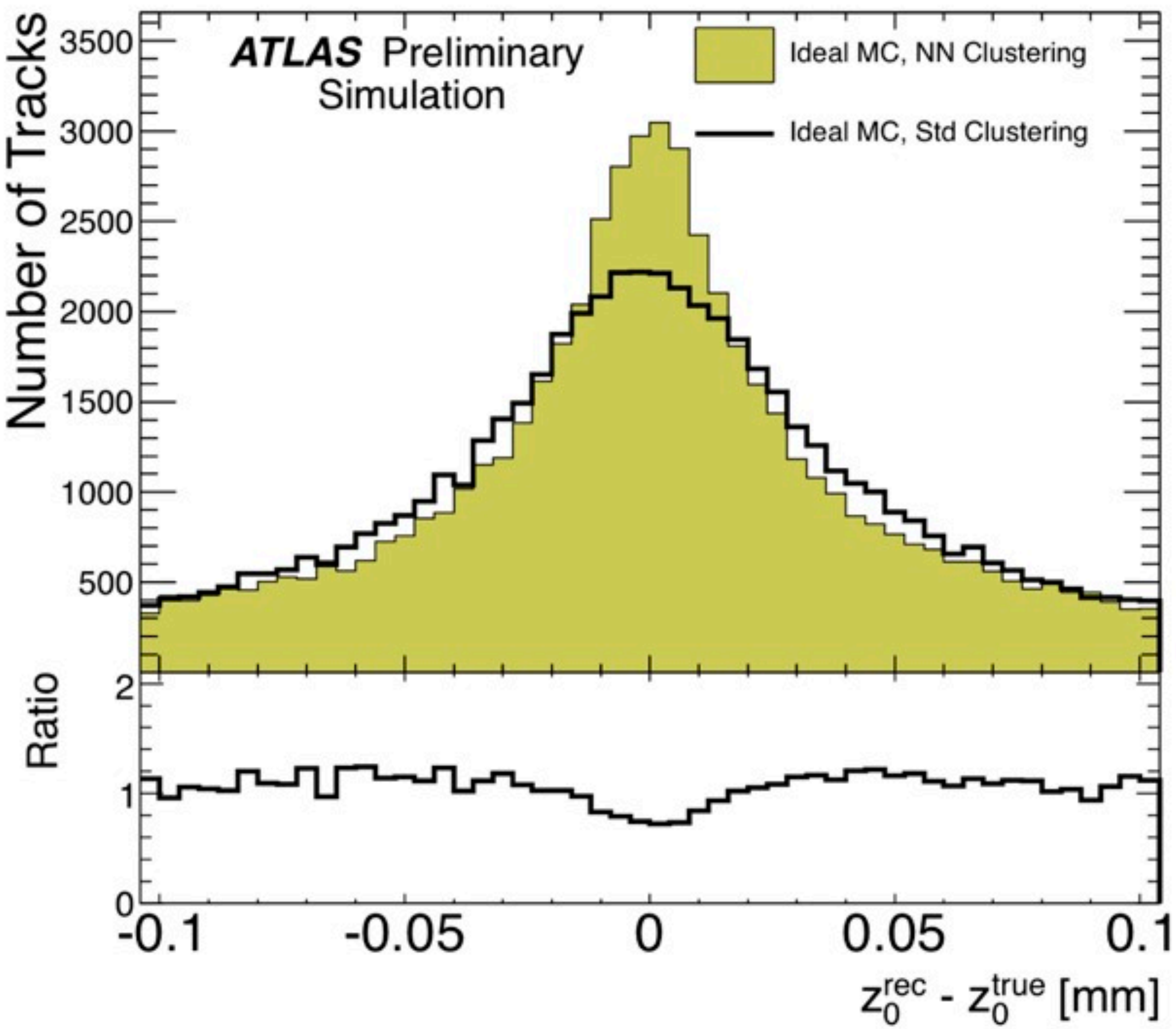


Cluster Resolution



- Dramatic improvement in resolution (track-to-measurement residual).
- Non-linear treatment of charge resolution allows recovery of single peak in track-to-hit residuals.

Track Resolution



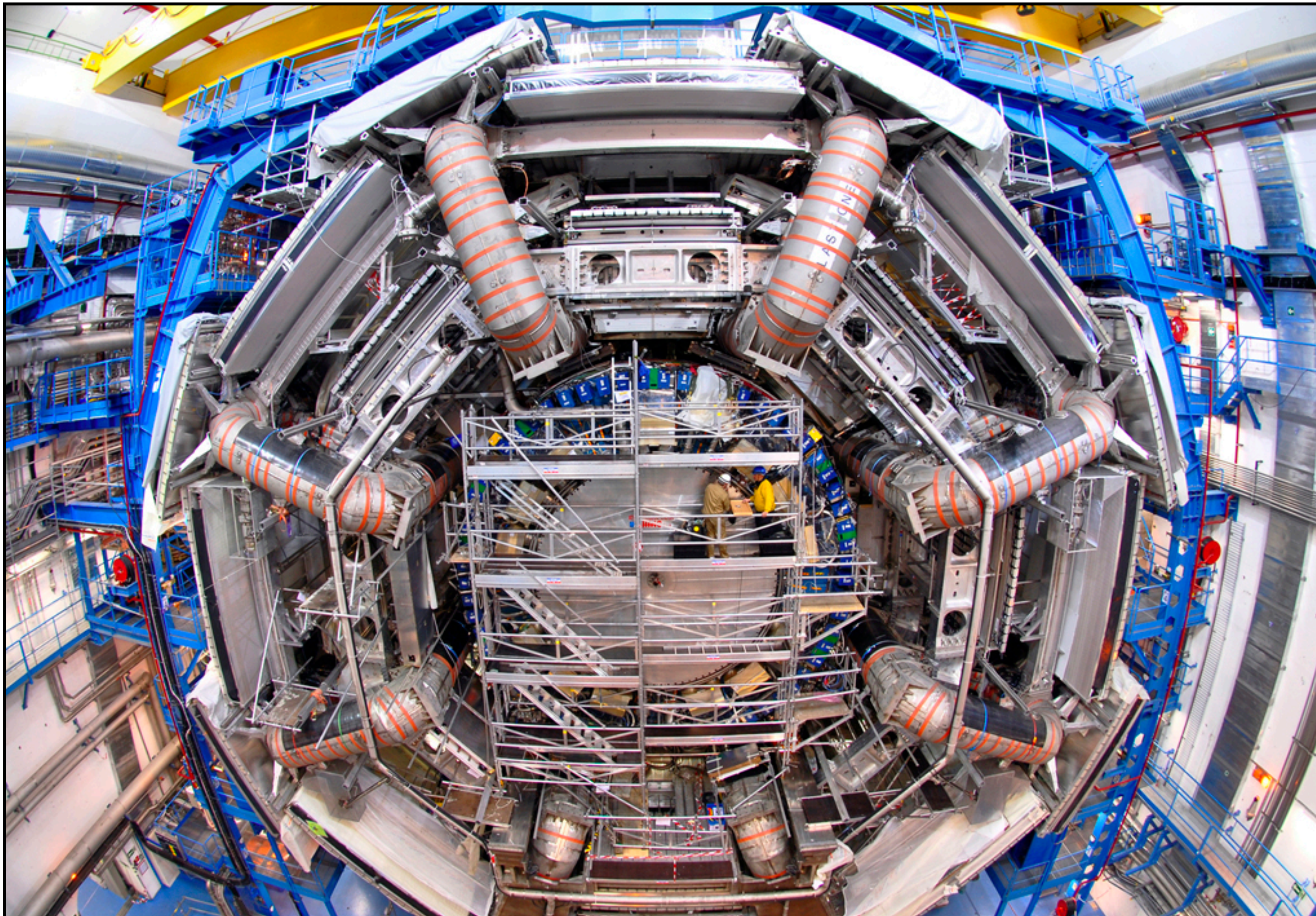
- Improved cluster resolution leads to improved track parameters.
- 15% improvement in longitudinal impact parameter.
  - Used for identification of long-lived particles (e.g. heavy flavour quarks).

Summary

- Neural network approach used to boost detector performance and make full use of detector design potential.
- All correlations inside pixel cluster are taken into account.
- Identify and split merged clusters created by multiple charged particles.
- Sizeable improvement in track measurements, particularly in dense environments such as in jet cores and hadronic tau decays.
- Non-linear treatment of charge collection improves impact parameter resolution even for isolated tracks.
- Improved two-particle separation will become even more important during future upgrades as particle density increases.

Back-Up

The ATLAS Detector



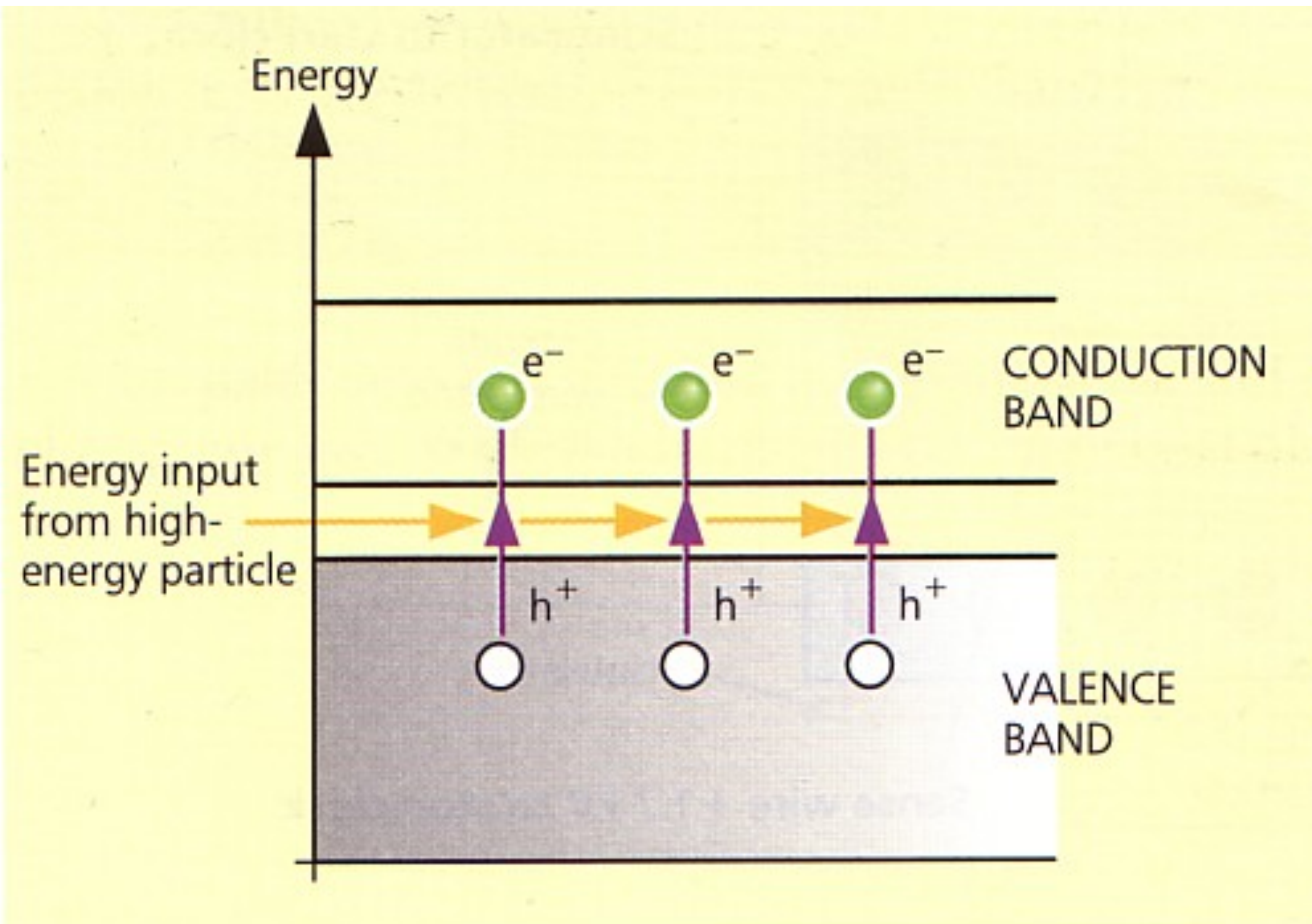
How much is an eV?

A single electron accelerated by a potential difference of 1 volt will have a discreet amount of energy, E=qV joules, where q is the charge on the electron in coulombs and V is the potential difference in volts.

$$\boxed{1 \text{ eV} = (1.602 \times 10^{-19} \text{ C}) \times (1 \text{ V}) = 1.602 \times 10^{-19} \text{ J.}}$$

10^3	k (kilo)
10^6	M (mega)
10^9	G (giga)
10^{12}	T (tera)

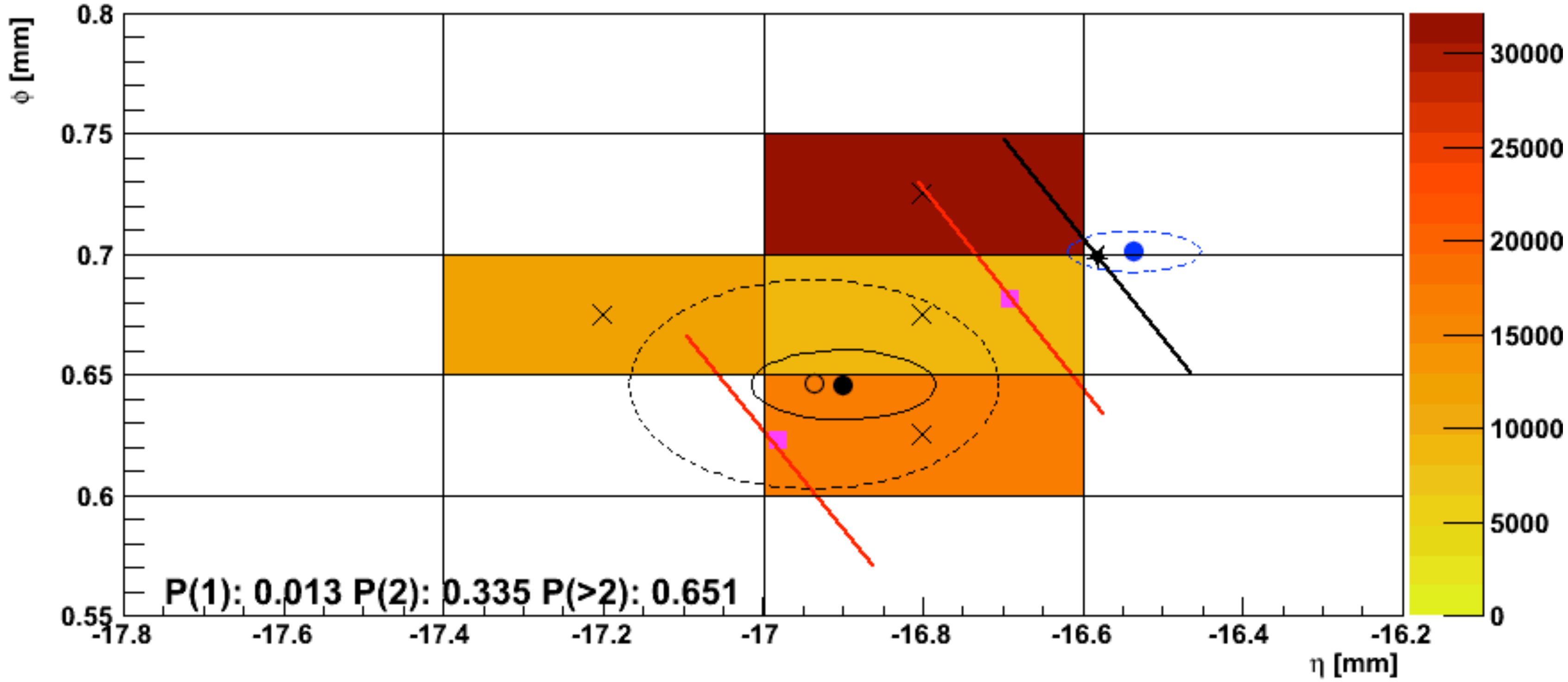
Semi-Conductor Trackers



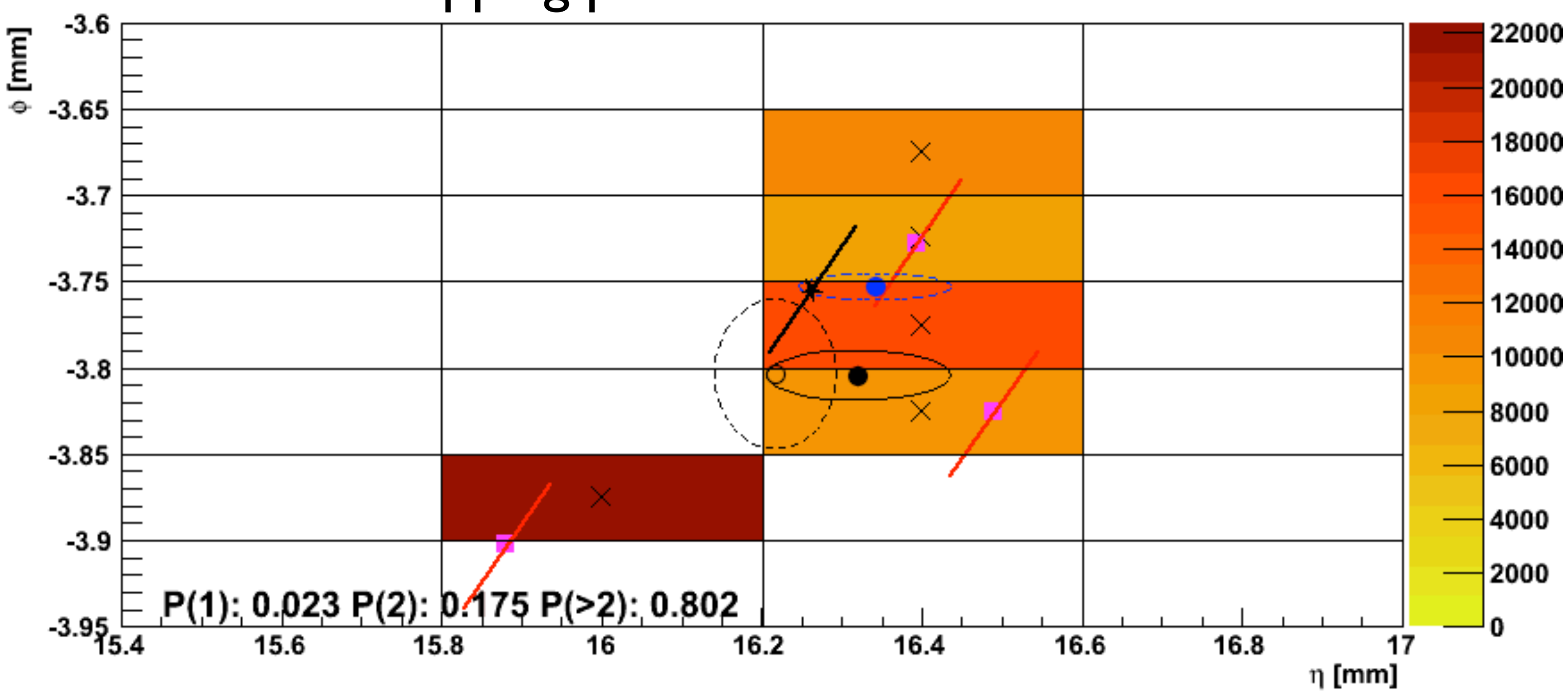
- In a semi-conductor, the gap between the valence band and conduction band is less than 1GeV.
- When high energy particle hits semi-conductor, some of the energy is absorbed by electrons which are then promoted to the conduction band.
- Number of charge carriers (both electrons and holes) is increased, and so resistance decreases.

Cluster Shapes

Two very close particles



Three overlapping particles



Abstract

We present a novel technique using a set of artificial neural networks to identify and split merged measurements created by multiple charged particles in the ATLAS pixel detector. Such merged measurements are a common feature of boosted physics objects such as tau leptons or strongly energetic jets where particles get highly collimated. The neural networks are trained using Monte Carlo samples produced with a detailed detector simulation.

The performance of the splitting technique is quantified using LHC data collected by the ATLAS detector in 2011 and Monte Carlo simulation. The number of shared hits per track is significantly reduced, particularly in boosted systems, which increases the reconstruction efficiency and quality. The improved position and error estimates of the measurements lead to a sizable improvement of the track and vertex resolution.

CPU Performance of NN Clusterisation

- Neural network clustering runs 6 times slower than traditional clustering.
- In context of the full event reconstruction, the re-evaluation of the splitting during track fitting and the increased combinatorics from additionally found track candidates leads at maximum to a 5% increase of the per event execution time in the highest pile-up conditions experienced during the 2012 data taking.