Advanced Reconstruction Algorithms for the CMS High Granularity Calorimeter

Kevin Pedro (FNAL)
On behalf of the CMS Collaboration

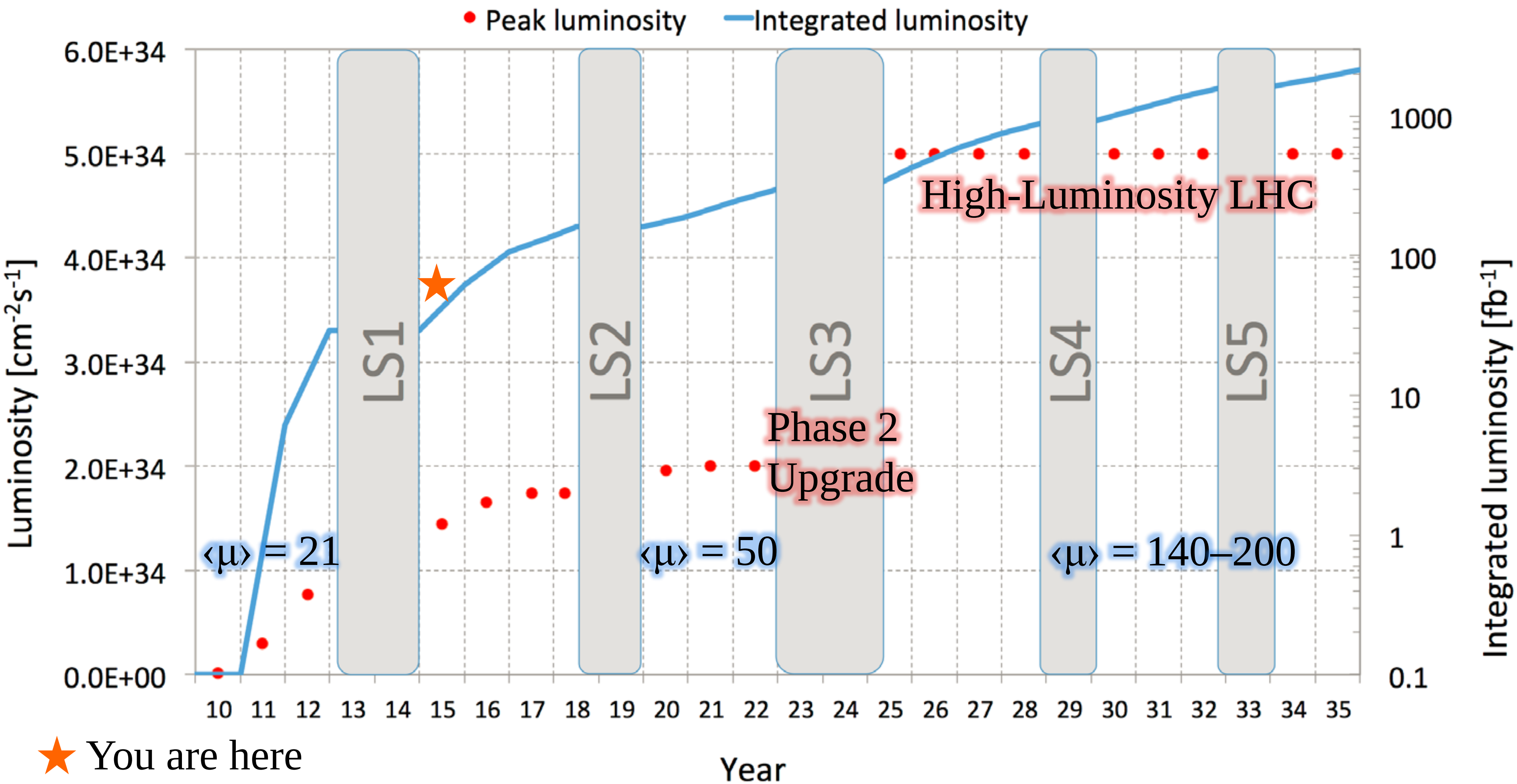
November 11, 2015

LHC Upgrade Schedule

Near the HL–LHC beamline: high radiation environment

After 3000 fb^{-1}, up to **150 Mrad** in EE, up to **30 Mrad** in HE

→ need new, radiation-hard endcap calorimeter technology



★ You are here

‹μ› = mean number of interactions per bunch crossing, or pileup (PU)

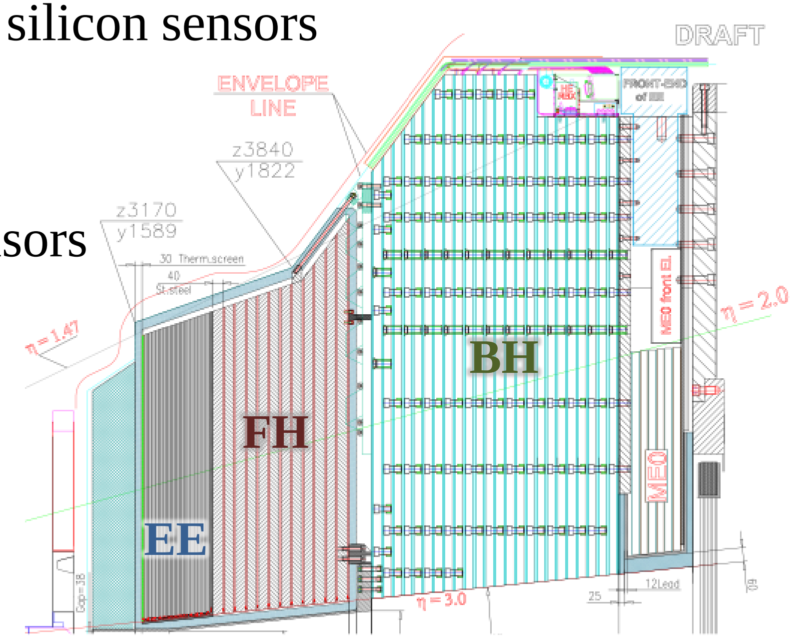
The High Granularity Calorimeter

CMS Phase 2 Upgrade: Replace the entire endcap calorimeter system (EE, HE) with an integrated high-granularity calorimeter

Inspired by CALICE designs:

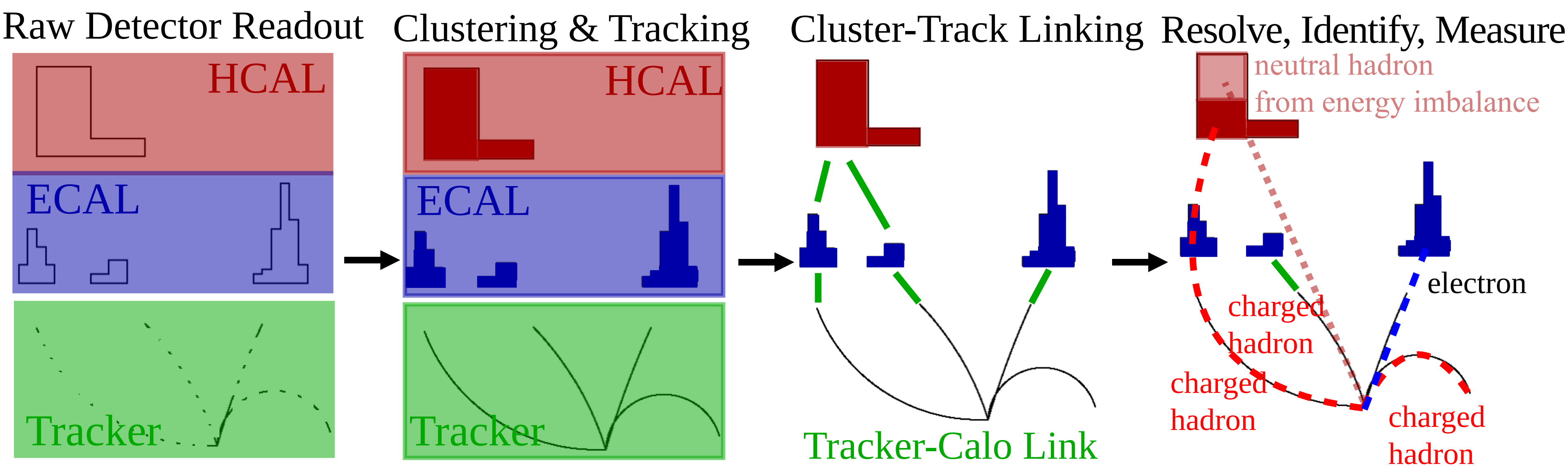
- ➢ radiation-hard components to survive the HL–LHC environment
- ➢ high granularity to record more information for physics in high pileup

- **EE: Endcap ECAL**
  - o 28 layers of tungsten/copper absorber and silicon sensors
  - o ~26 X_0 / 1.5 λ_0 thick, 4.3M channels
- **FH: Front HCAL**
  - o 12 layers of brass absorber and silicon sensors
  - o 3.5 λ_0 thick, 1.8M channels
- **BH: Back HCAL**
  - o 12 layers of brass absorber and (radiation-hard) plastic scintillator
  - o 5 λ_0 thick, 1K–10K channels



How can we exploit all of this information? Particle Flow!

What is Particle Flow?



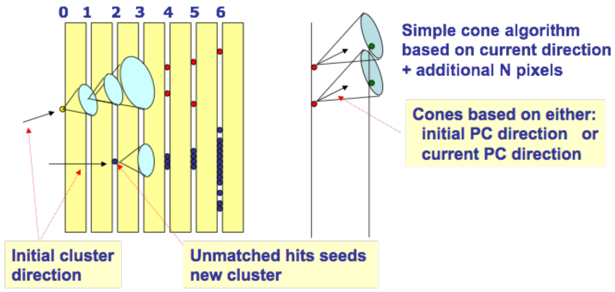
Reconstruction that yields *unambiguous* list of *identified* final particle states:

- *Cluster* detector hits together in each detector
- *Link* tracks to calorimeter deposits:
  tracking augments calorimeter response

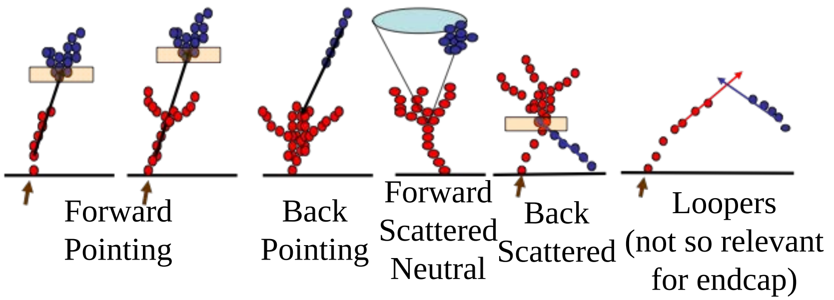
➢ Best use of *all* detector data to *measure* and *identify* all particles in a collision
→ performance depends on optimized use of all information

High Granularity Clustering

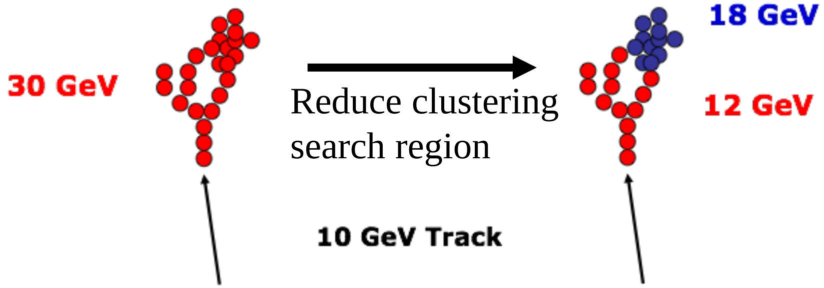
1. Initial Clustering



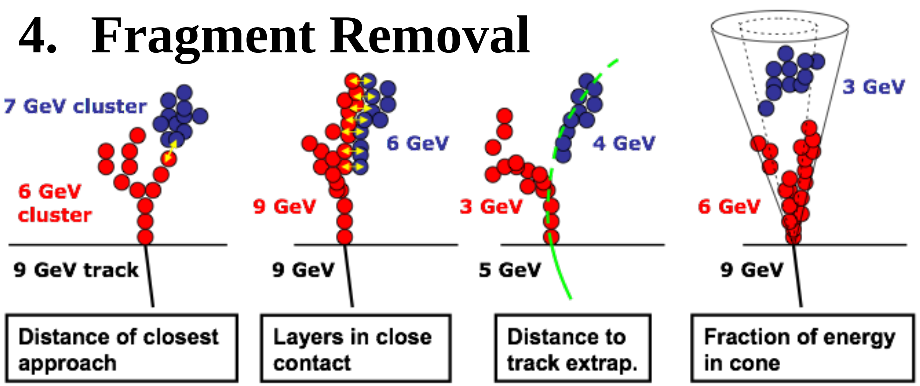
2. Topological Associations



3. Iterative Clustering

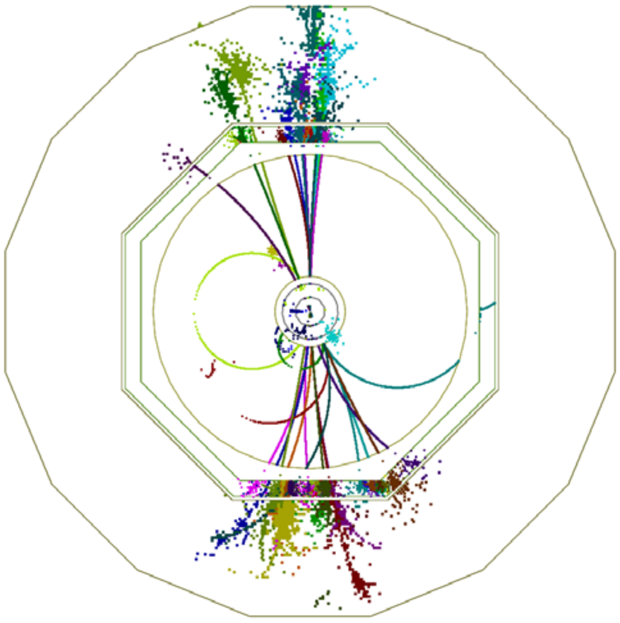


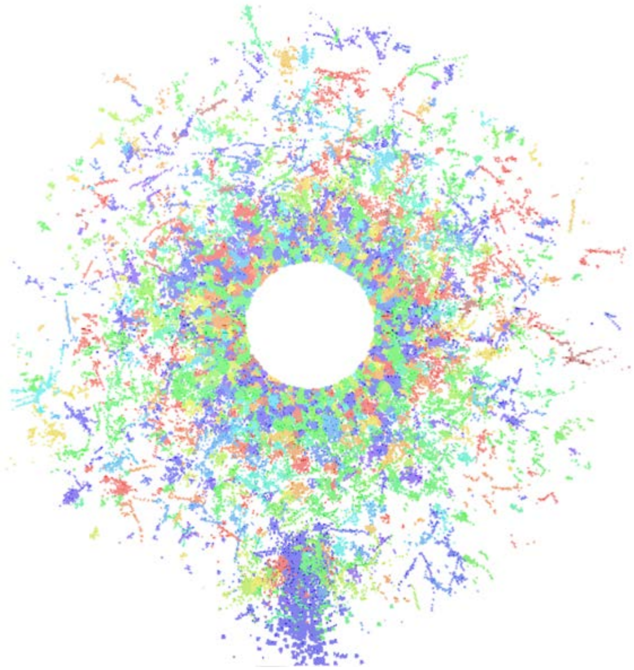
4. Fragment Removal



Clustering approach based on the Pandora Particle Flow Algorithm developed by Mark Thomson for ILD and CALICE

From ILD to CMS



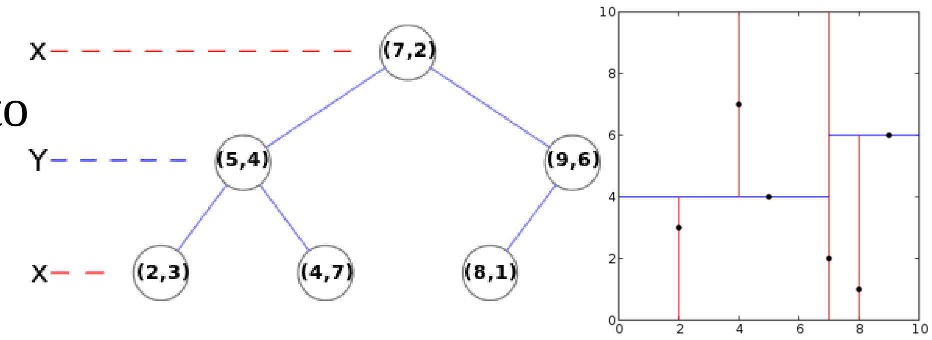


ILD and CALICE	**HGCAL at CMS**
• e^+e^- collisions (ILC, CLIC)	• pp collisions (HL–LHC)
• No/low pileup	• High pileup (140–200 interactions per event)
• Optimized for barrel	• Endcap calorimeter

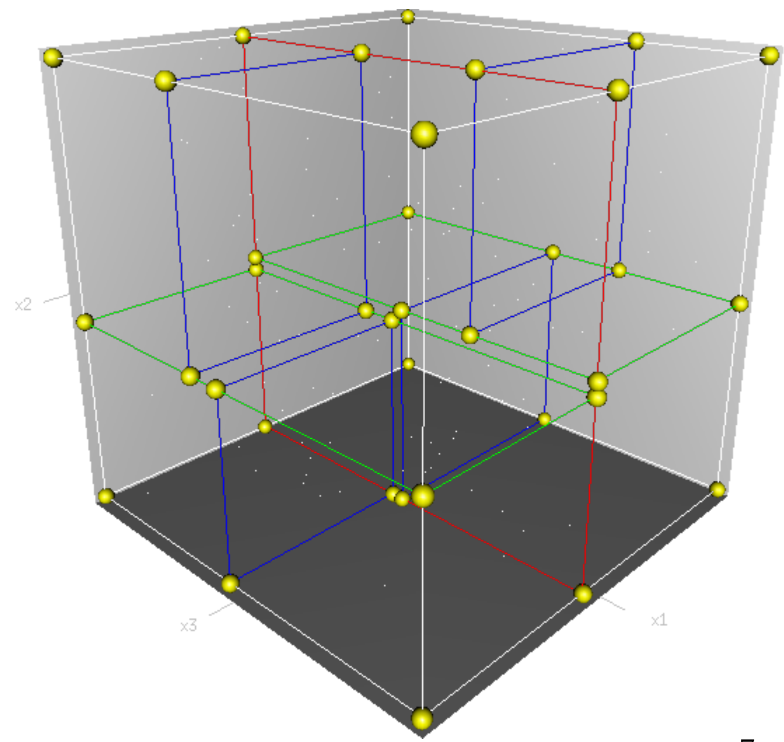
Computational Geometry

- Nearest neighbor search: core of any clustering algorithm
- Naïve approach: compare each RecHit to every other RecHit
  - o $\mathcal{O}(N^2)$ behavior
  - o With high pileup, N = 200,000!
- *k*-d trees: a binary tree in *k* dimensions
  - o Change the splitting dimension at each depth (examples at right)
  - o $\mathcal{O}(N \cdot \log(N))$ to search for neighbors of hits in a region
- Hull finding: get set of outermost points
  - o More efficient when comparing two existing clusters
- These algorithms provide orders-of-magnitude speedup over naïve approaches

k-d tree in 2 dimensions:



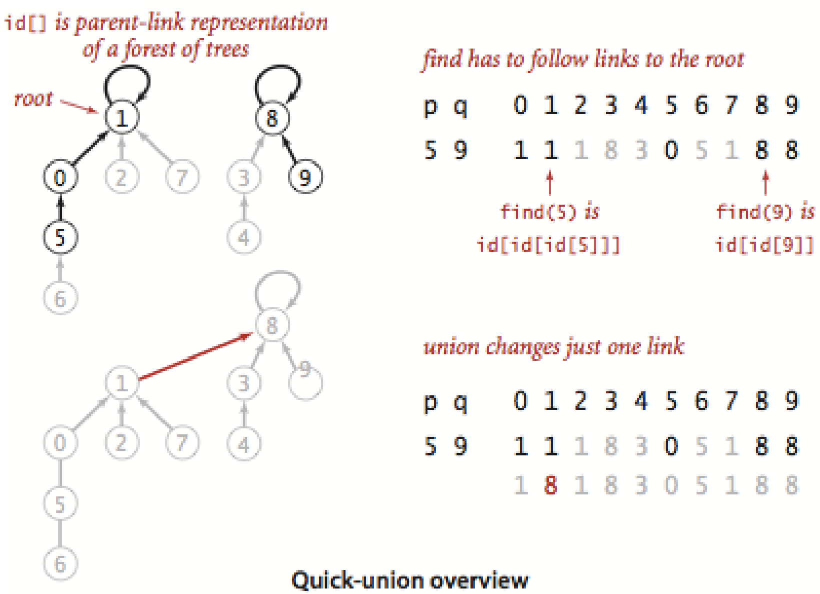
k-d tree in 3 dimensions:



Graph Theory

- Efficient way to manage associated sets of points
  - o Start with hits as vertices of disconnected graph
  - o Associate hits by building edges between vertices in the graph
- No need to search over and over again when adding a hit to an associated cluster
  - o $\mathcal{O}(N^2) \rightarrow \mathcal{O}(N \cdot \log^*(N))$ (almost linear)

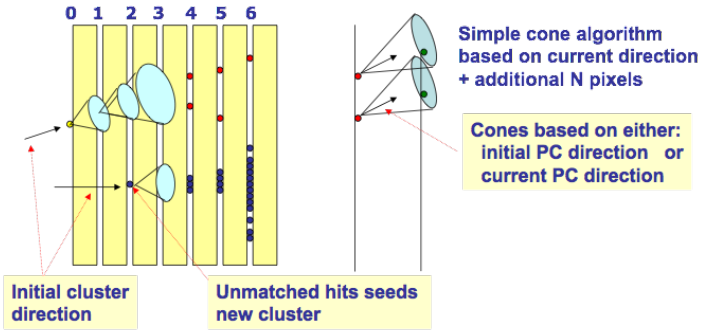
QuickUnion efficiently represents associated sets of points:



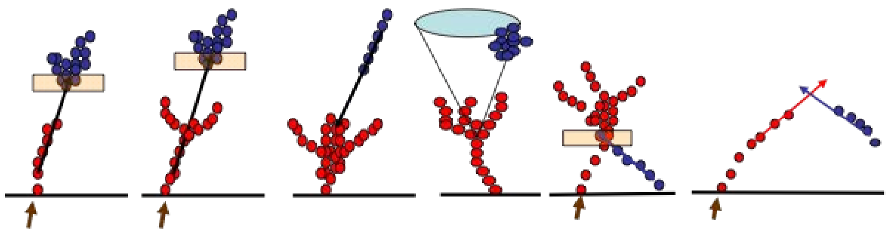
Quick-union overview

Advanced Algorithms for HL–LHC

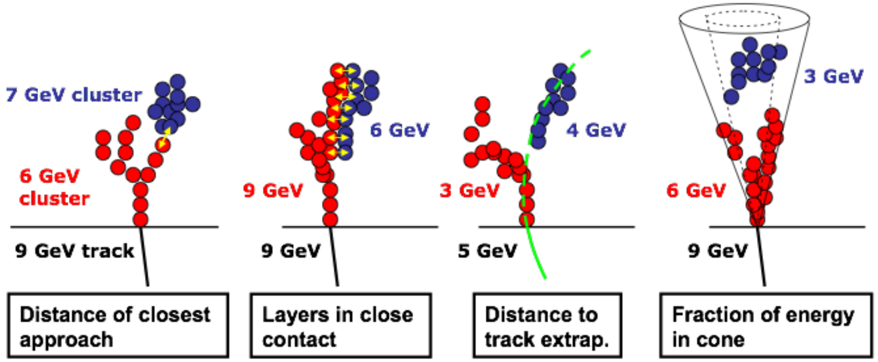
Cone clustering: 10–20×
k-d trees very important



Topological Assc.: ~3×
k-d trees, QuickUnions both useful



Frag. Removal: 3–4×
k-d trees help, hull finding also important



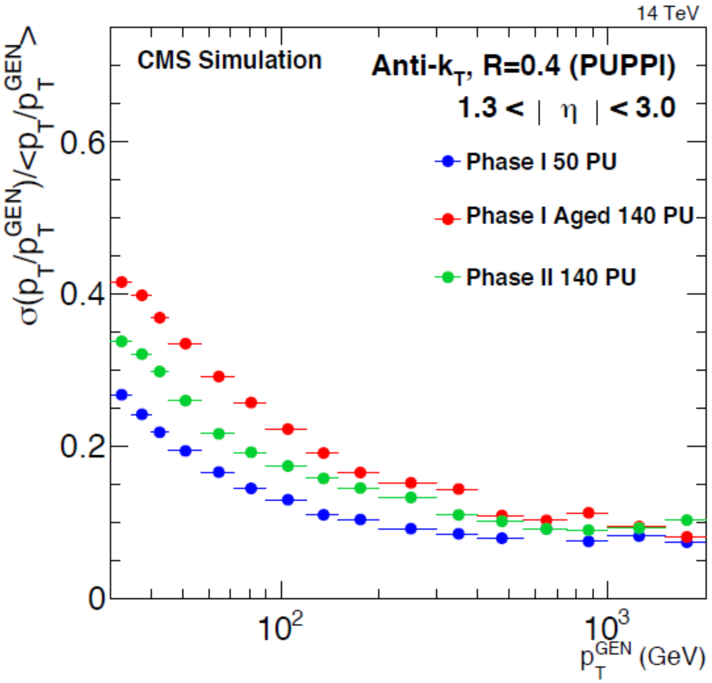
Advanced algorithms provide significant speedups in the reconstruction code

Without these computing performance improvements, simulations at high pileup would be impossible

140 pileup:
1 hour/event → 10 min/event

Conclusions & Future Considerations

- Initial effort with advanced algorithms provided reconstruction and performance results for the CMS Phase 2 Technical Proposal



- Imaging calorimetry is the future!

- Can further exploit shower shape info: software compensation (EM vs. hadronic), pileup rejection

- Add time and energy information for 4D or 5D clustering and further pileup rejection

- Multithreading will provide more speedup: clustering is local

- Research in computer science shows 50–100× speedup for clever implementations of *k*-d trees on GPUs

Backup

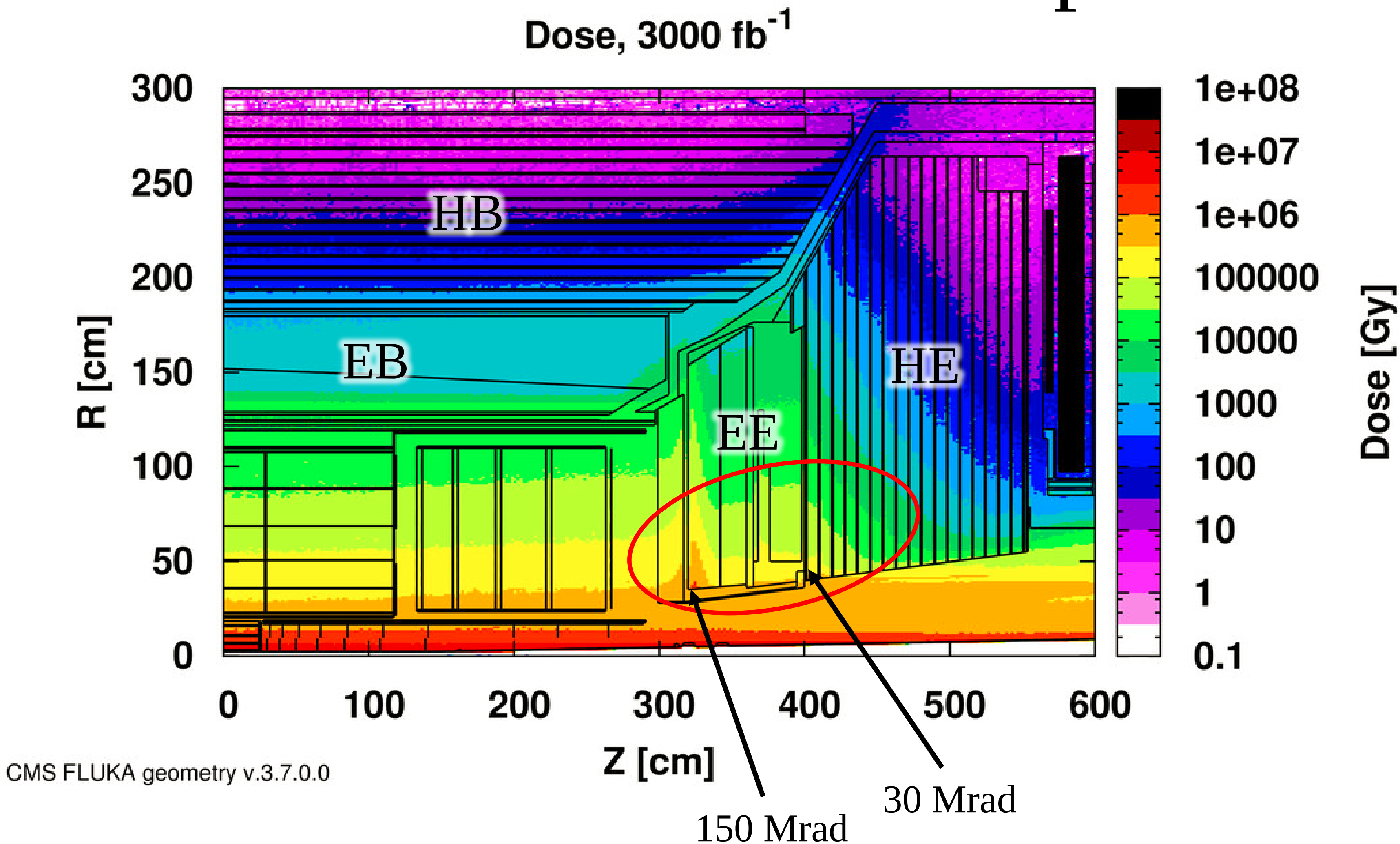
References

- The LHC Experiments Committee, "Technical Proposal for the Phase 2 Upgrade of the CMS Detector", LHCC-P-008, June 2015
- M. A. Thomson, "Particle Flow Calorimetry and the PandoraPFA Algorithm", *Nucl. Instr. Meth. A* **611** (2009) 25, arXiv:0907.3577
- F. Gieseke et al., "Buffer k-d Trees: Processing Massive Nearest Neighbor Queries on GPUs", ICML 32 (2014) 172

Images borrowed from:

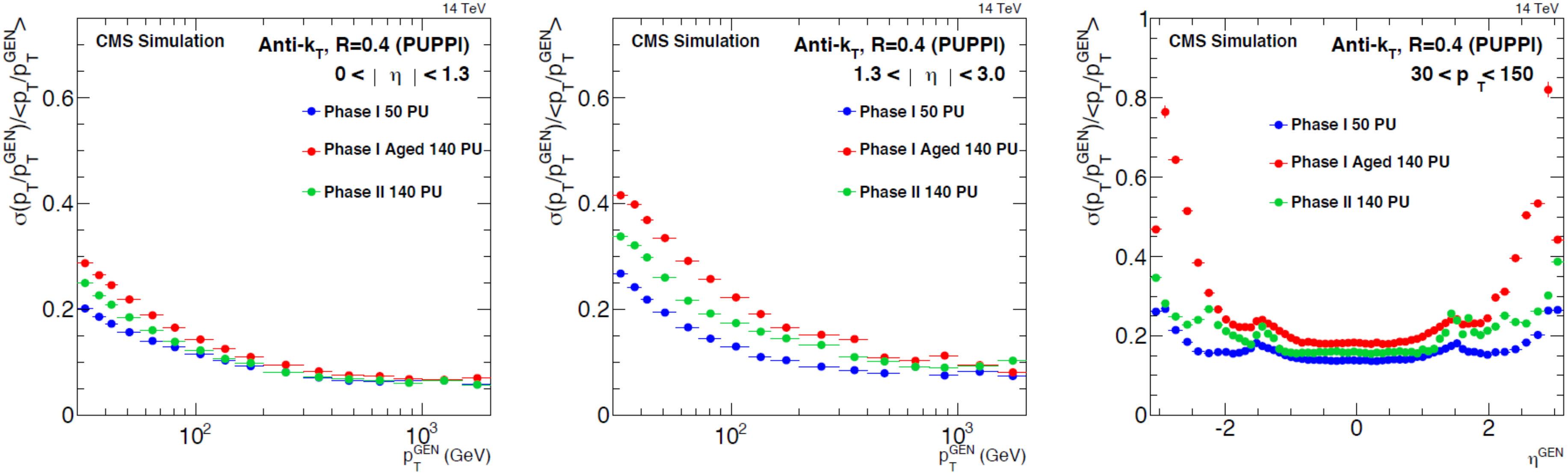
- http://www-jlc.kek.jp/~miyamoto/evdisp/html/
- https://en.wikipedia.org/wiki/K-d_tree
- http://algs4.cs.princeton.edu/15uf/

CMS Radiation Map

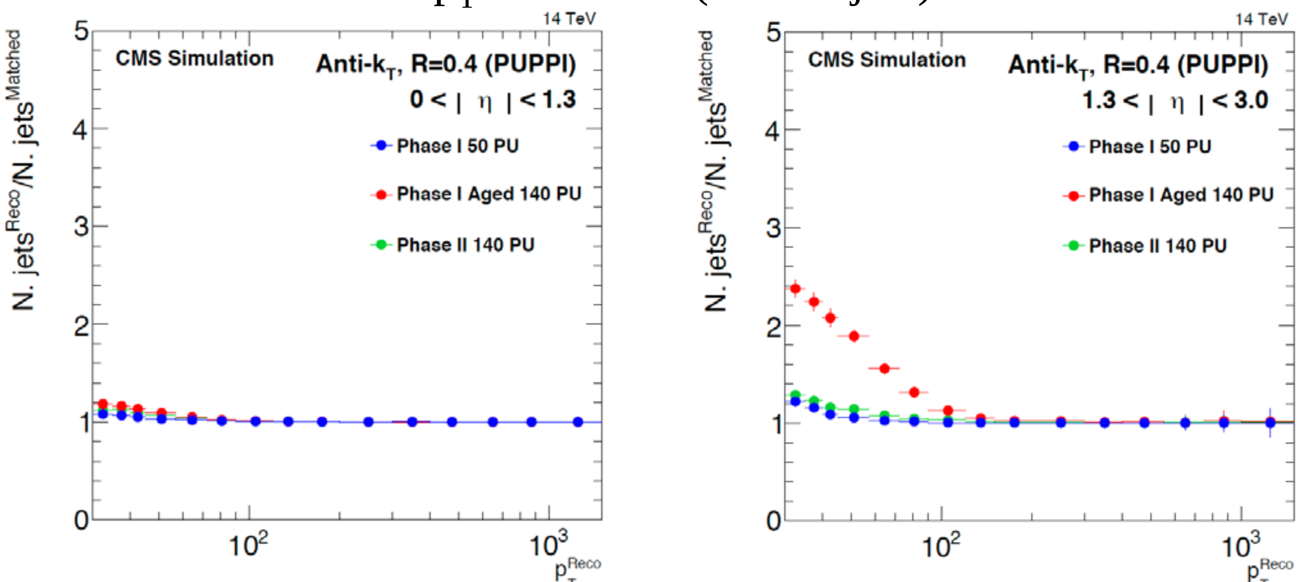


Existing endcap calorimeters will not survive the high radiation dose expected after 3000 fb^{-1} delivered by the HL–LHC → need to be replaced

Current Performance: Jets

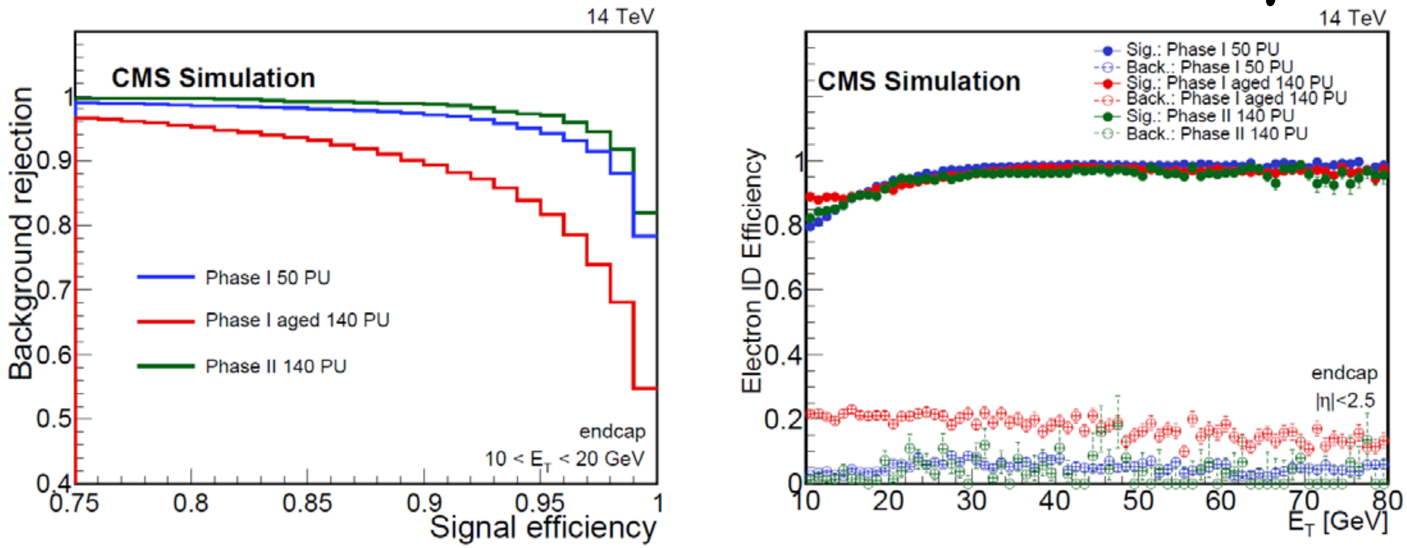


Jet p_T resolution (PUPPI jets)

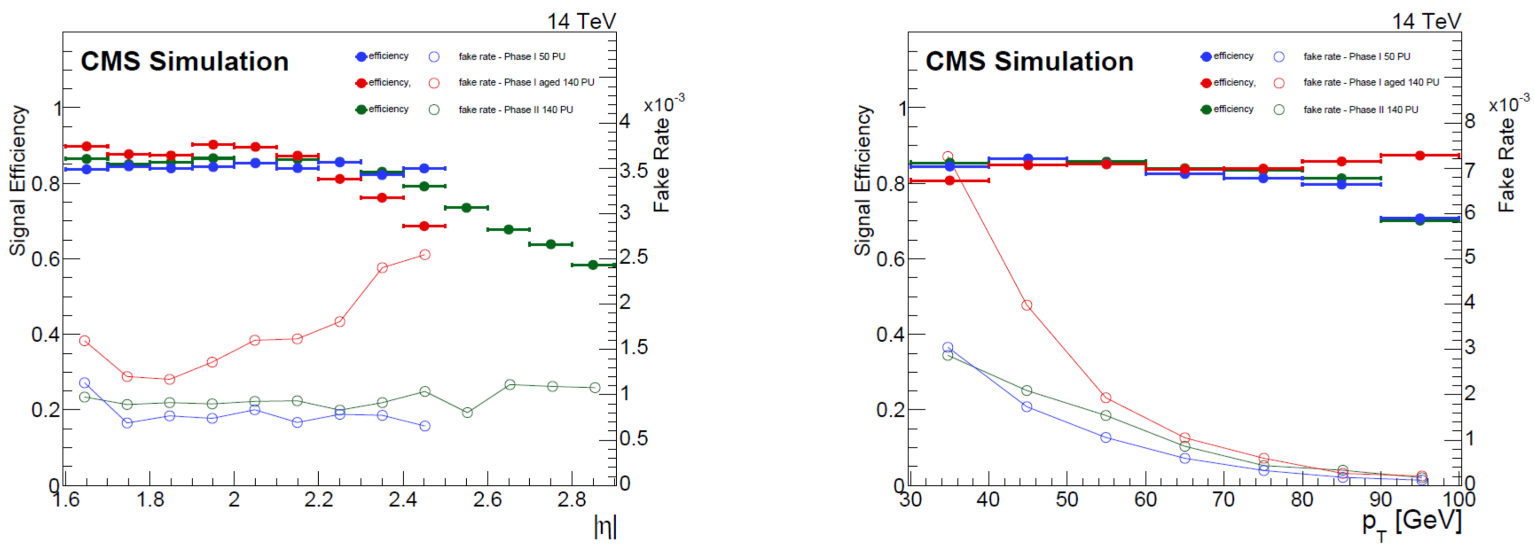


Pileup jet rate (PUPPI jets)

Current Performance: e/γ

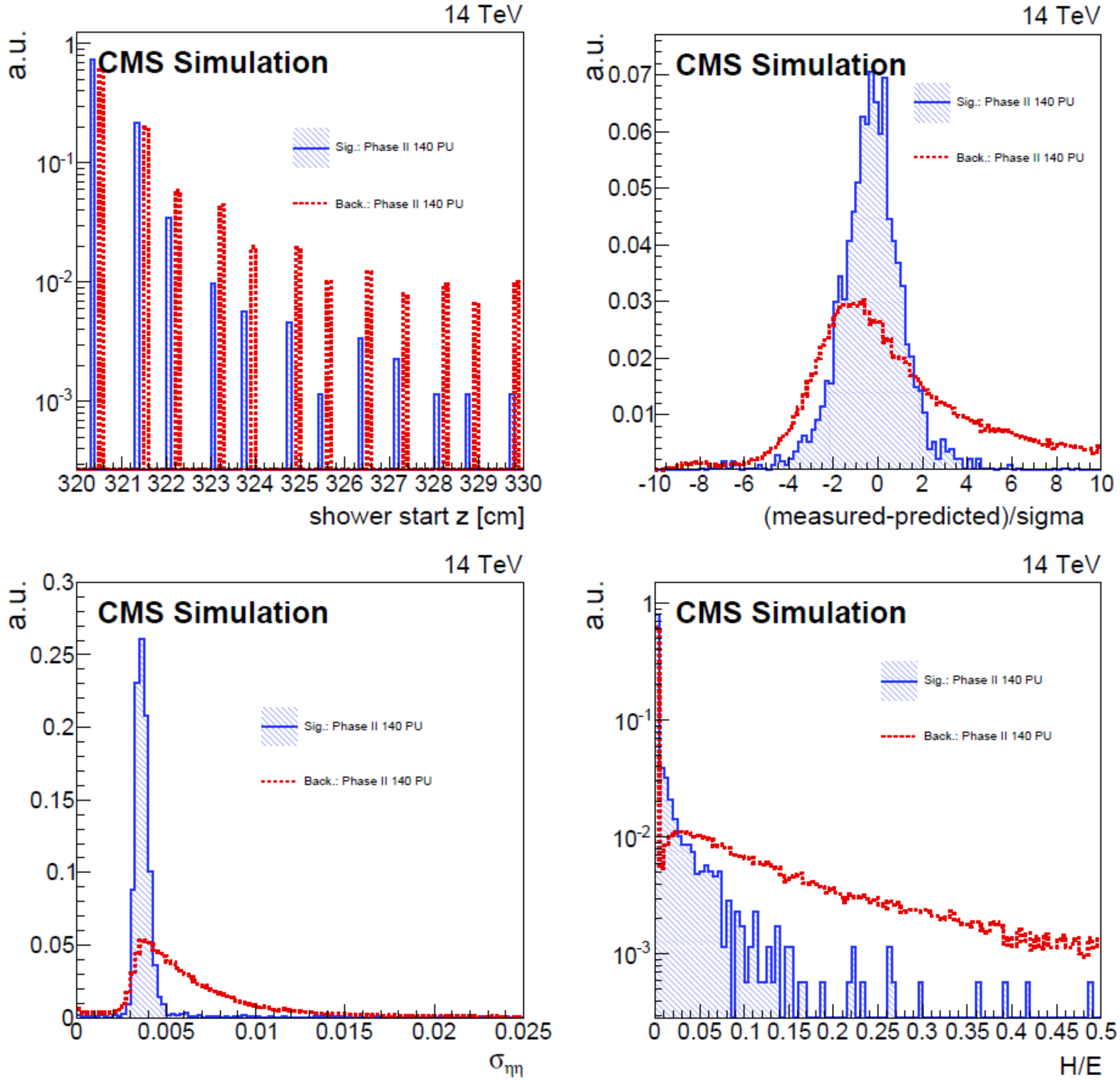


Electron identification performance (using BDT)



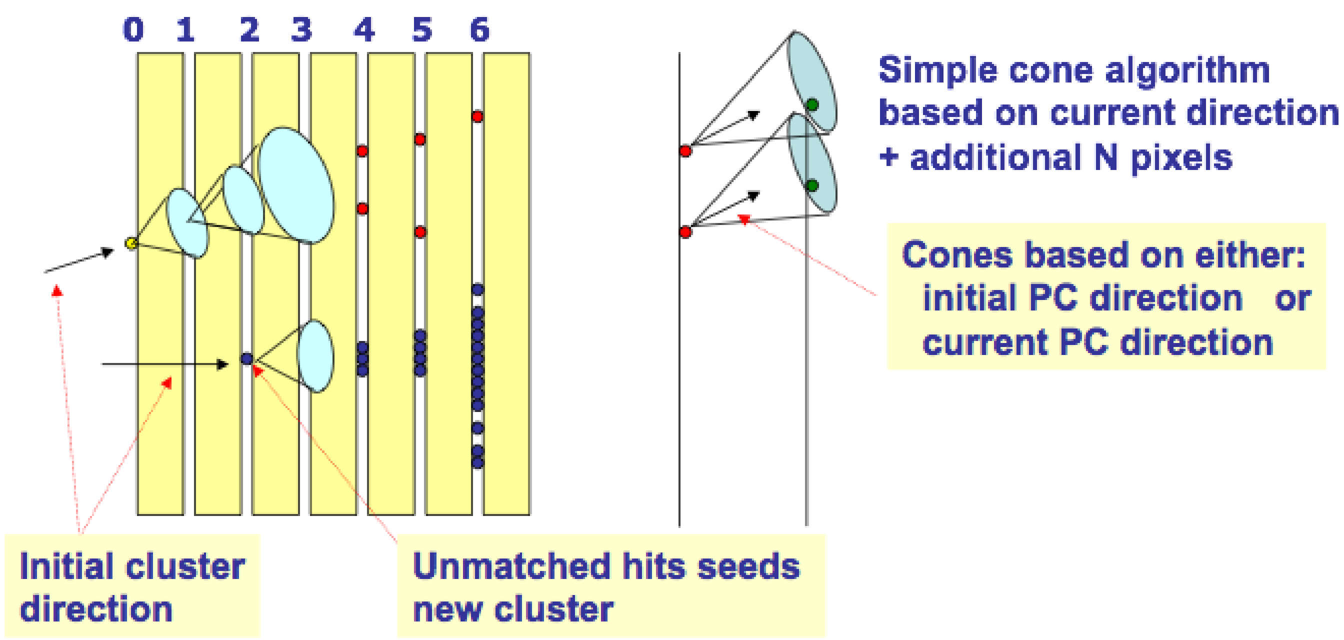
Photon identification performance

Electron ID Variables



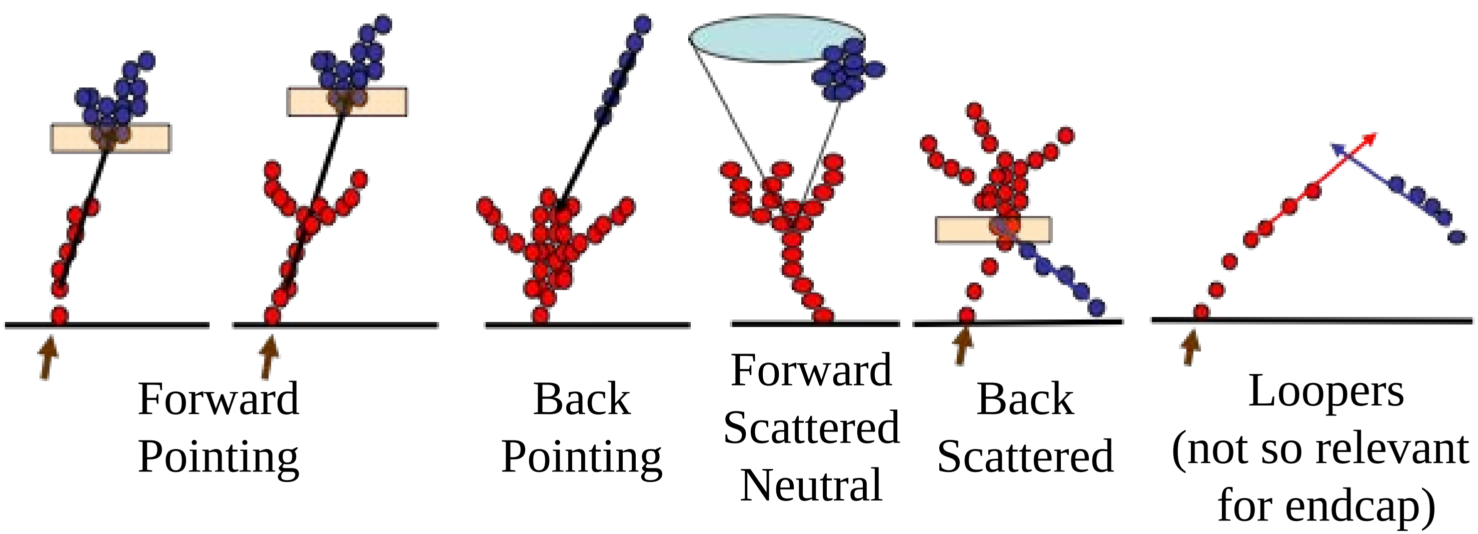
Initial Clustering

- Track-seeded initial cluster positions and directions (optional)
- Loop over calorimeter hits to find nearest cluster
  - Look in narrow region in few previous layers, then same layer
  - If no match *at all*, seed new cluster w/ expected direction pointing back to IP
- Gives reasonable clustering to start, though it fragments clusters apart
  - Other algorithms put event back together
  - Easier and more efficient to detect patterns that should be merged (vs. detecting patterns that should be split)



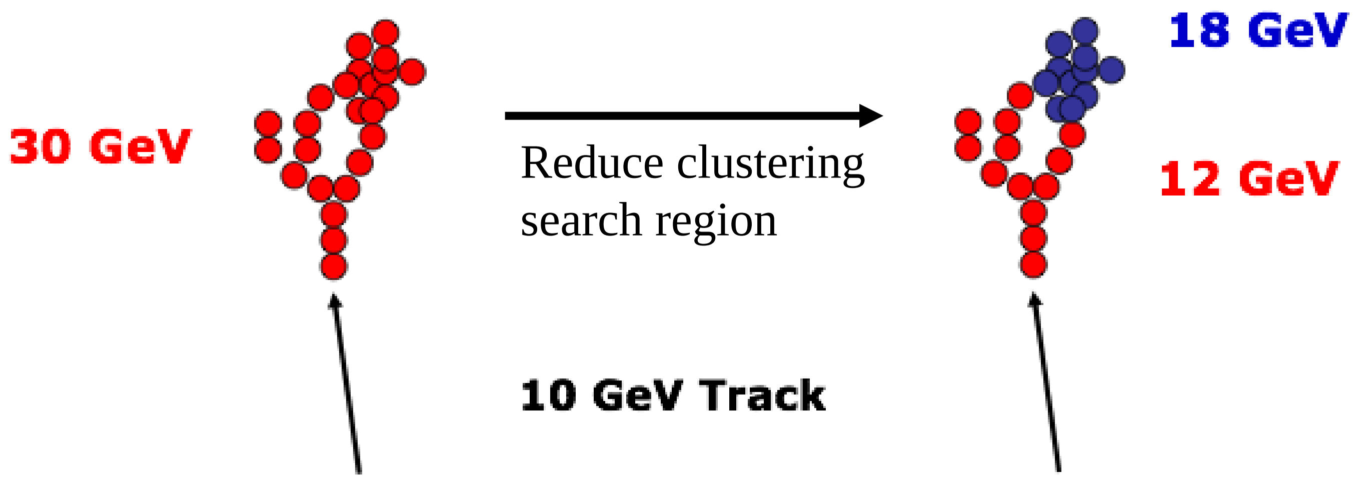
Topological Associations

- Use longitudinal granularity and tracking capabilities of HGCAL to gather fragmented clusters
  - MIP-like clusters point with very high precision
    → most cluster-cluster associations are accurate
  - Exploit in-situ cluster direction fit from initial clustering step
- Prevent gross mistakes in charged energy component by requiring merged clusters to be consistent with parent tracks in E/p



Iterative Clustering

- Look at all track-cluster associations in which cluster energy > track energy
  - Typically 3σ deviations
    (Requires a clean set of tracks → need *a priori* fake rejection in CMS)
  - Attempt reclustering for better match with track:
    alter the clustering parameters, from coarser clustering to very narrow clustering
- Keep reclustering result with best energy balance in local charged component
  - Sensitive to both upwards and downward fluctuations in the cluster energy gathering efficiency (can make a cluster bigger if track energy is much too large)
  - Get the best calorimeter-defined clustering with respect to input track information



Fragment Removal

- Final clustering step before particle flow
- Previous clustering steps naturally seed "fragments"
  - Smaller, split-off clusters on periphery of larger ones
  - Causes double counting or "confusion" if that cluster belongs to a charged object (energy usually taken from track)
- Look for residual topological associations
  - Clusters with shared boundaries or contained within projection of cluster envelope
  - Clusters along track propagation in calorimeter