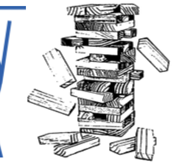


Facing the challenges of the reconstruction in High-Granularity Calorimeters

Erica Brondolin, **Felice Pantaleo**, Marco Rovere

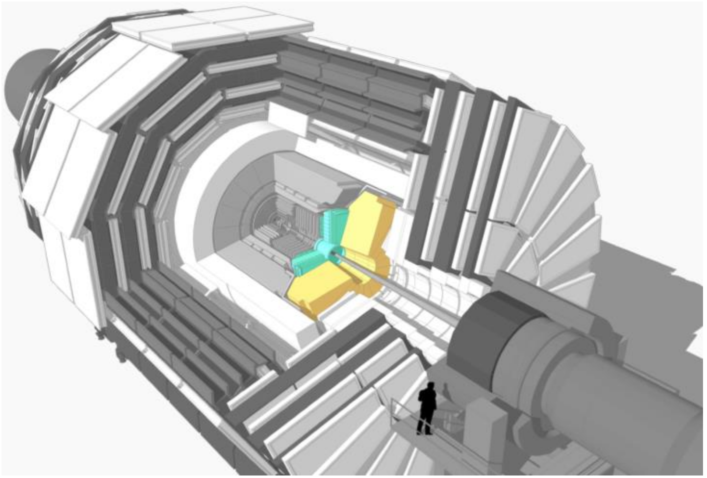
EP-CMG

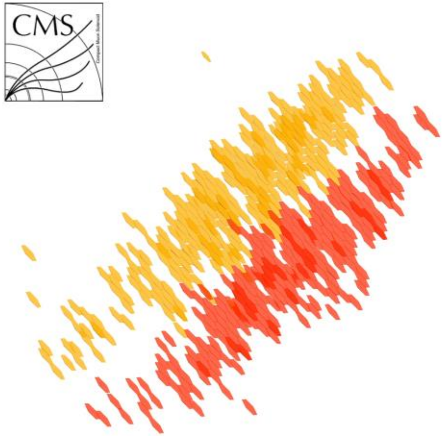
felice@cern.ch

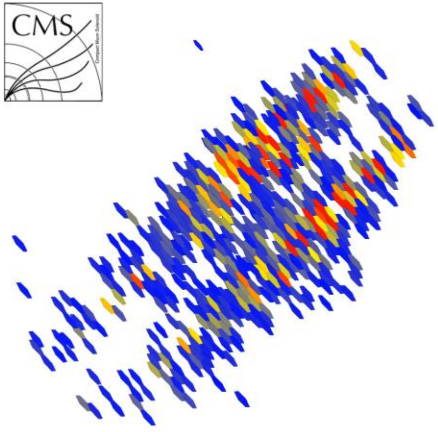
Motivation

High Granularity Calorimeters candidate technology for detectors at future accelerators (CMS@HL-LHC, CLIC, FCC-hh)

- Radiation hard detectors based on a mix of silicon and scintillator technologies
- High transverse/longitudinal granularity and timing allows for enhanced 5D pattern recognition, particle flow reconstruction, PID and pileup mitigation
- Preserve or even improve sensitivity in the interesting and busy forward region for VBF/VBS



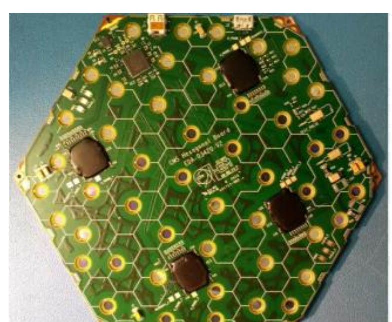


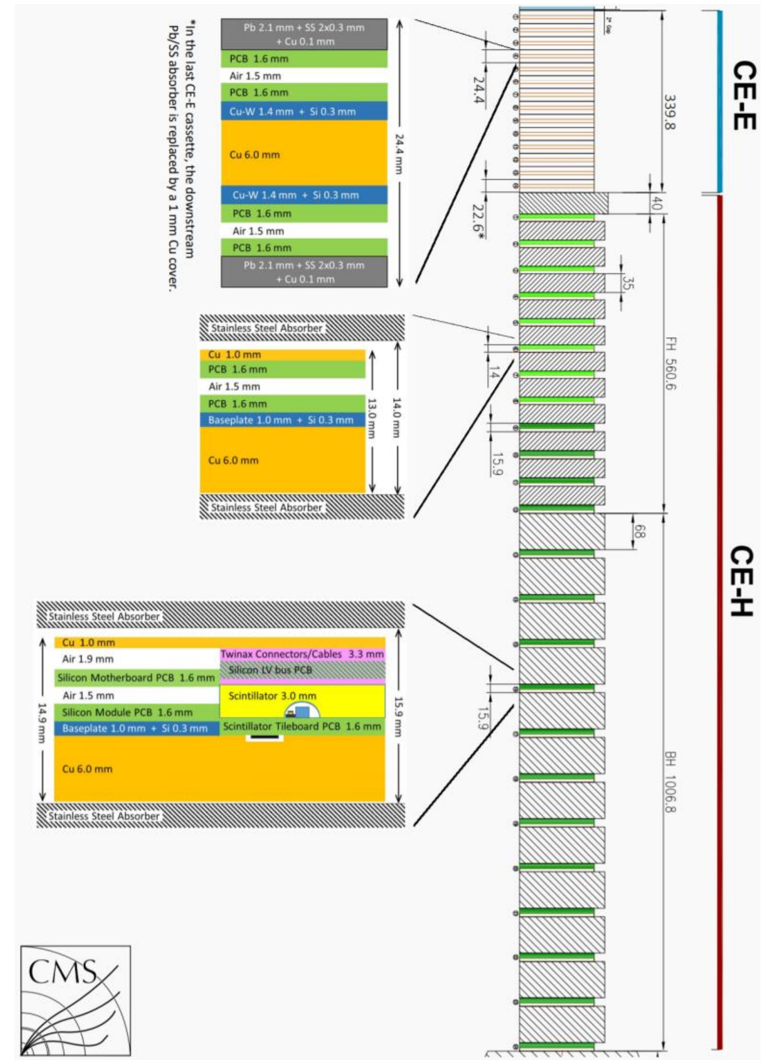


Sampling

The longitudinal sampling of the layers follows physics principles of shower development with two groups of sampling (em and hadronic)

In the transverse plane, sensitive Silicon cells of 0.5-1cm^2 contribute to disentangle two nearby showers or MIPs





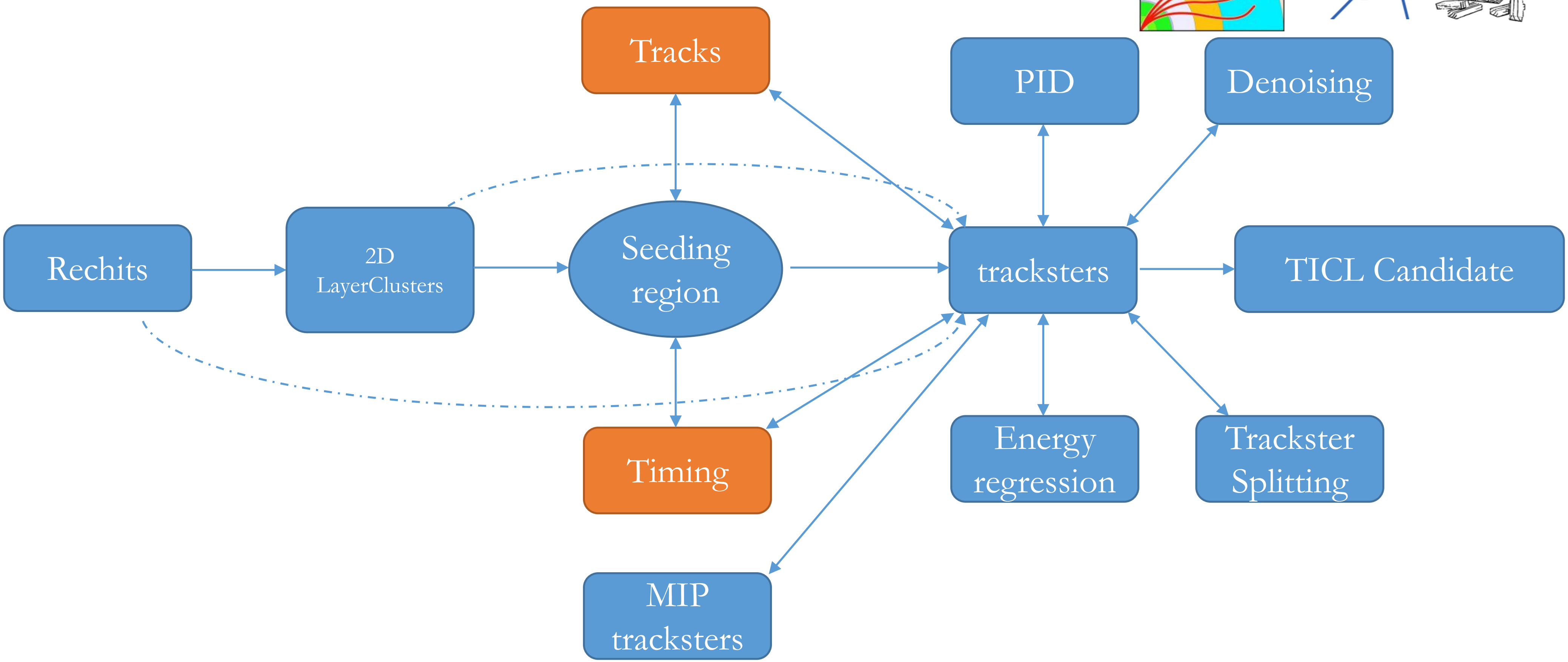
Reconstruction in HGCals

- Reconstruction in High Granularity Calorimeters can be challenging in high PU environments as
  - overlapping showers are the norm
  - naive reconstruction algorithms exploring many combinations among all possible paths are expected to fail due to memory/timing explosion
  - Fertile ground for new techniques and algorithms: clustering, machine learning, graph theory, and modern computer architectures
    - Must be planned and designed, taking into account the information from the surrounding tracking and timing detectors.
    - The CMS HGCal is a good opportunity as its software reconstruction cannot be obtained by adapting any existing CMS sub-detector software

TICL - [ˈtɪkəl]: The Iterative Clustering

What is TICL?

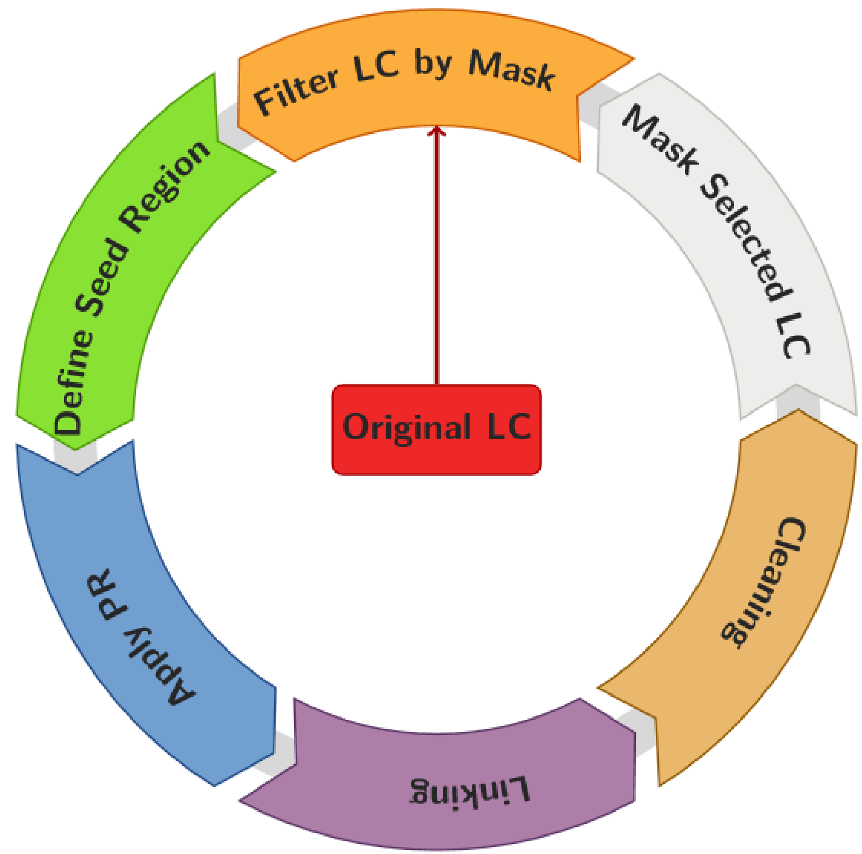
- TICL is a modular framework integrated and under development in the CMS reconstruction
- Its final purpose is to process HGCal rechits (x, y, z, t, E) and return particle properties and probabilities
- Modules and interfaces are defined so that new developers don't have to know anything about the CMSSW core framework in order to contribute
- Geometry agnostic
- Algorithms implemented as plugins



Many of these chains can co-exists in "iterations" targeting different particles

An iteration in TICL

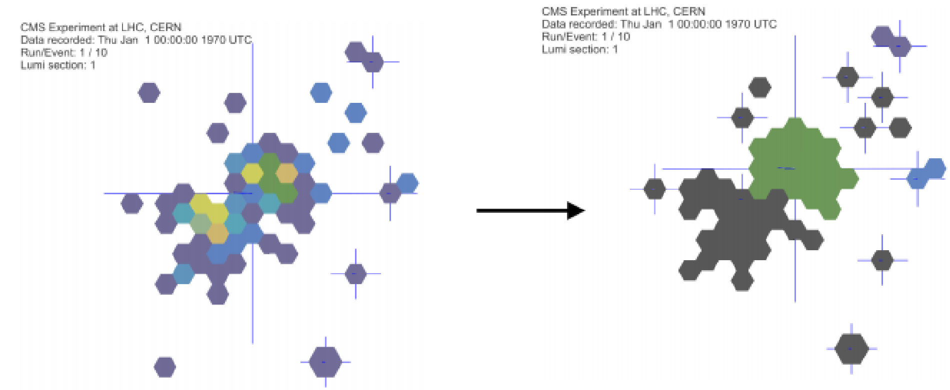
- hits on the same layer are clustered together in Layer Clusters (LC), in parallel
- LCs belonging to physics objects are masked-out and won't be available for the next iteration
- In CMS at PU200, O(10^5) hits and O(10^4) LCs per event
- Pattern Recognition algorithms are plugins
- Linking/cleaning/classification
- Make use of timing information in PR when available



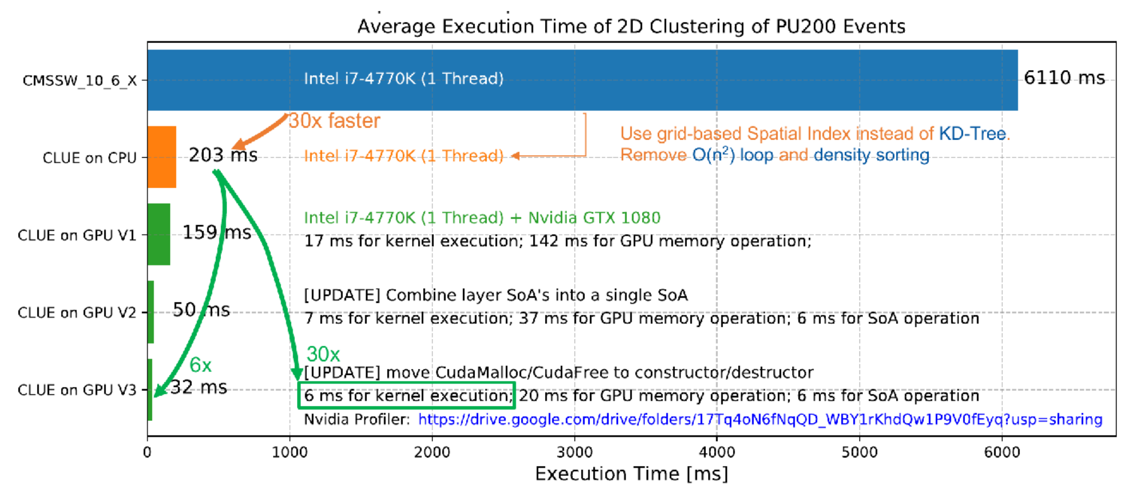
Layer Clusters

TICL's modularity starts with the Layer Clustering:

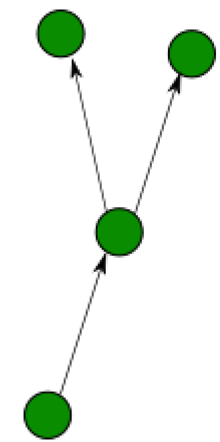
- Input rechits, output layer clusters
- Swappable plugins for 2d clustering algo
- CLUE default clustering algo:



https://arxiv.org/abs/2001.09761



Pattern recognition: Tracksters



Tracksters are basically small DAGs whose vertices are layerClusters

Pattern recognition can be any algorithm that
connects layer clusters together to produce a Trackster

- Cellular Automaton, Kalman Filter, 3d clustering algo, Final Armageddon algo
- Can be constrained in a seeding region

- Tracksters have to be pruned, linked to off-hgcal objects, and filled with:
    - regressed energy, probabilities for each particle-id class and fitted timing for the 3d-object

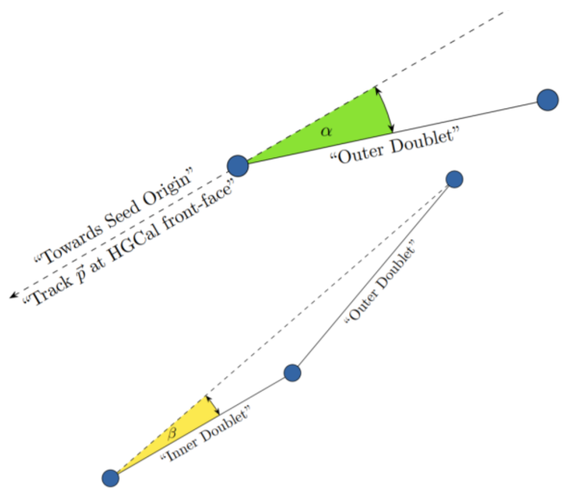
Tracksters can then be fed to PF

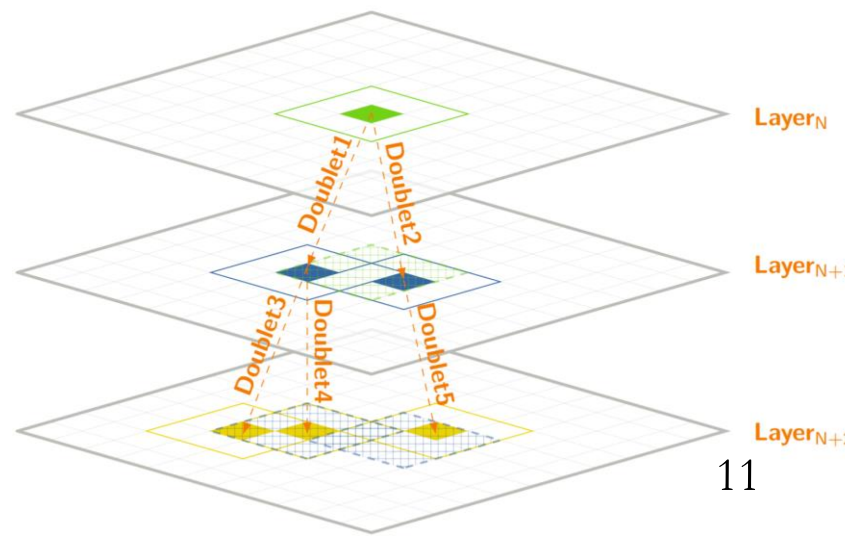
Pattern Recognition by CA

For each 2D layer cluster in a LayerN, open a search window in LayerN+1

- Search uses a 2D tiling date structure in η, ϕ
- A layer cluster in this search window will make a doublet with the original layer cluster
  - Requirement on the compatibility of the timings of the two LCs
- Doublets are linked if two angular requirements are satisfied:
  - A requirement on the direction (α) of each doublet wrt the vertex (or wrt a track direction if this is a track seeded iteration)
  - A requirement on the angle (β) between the doublet

Designed with parallelism in mind





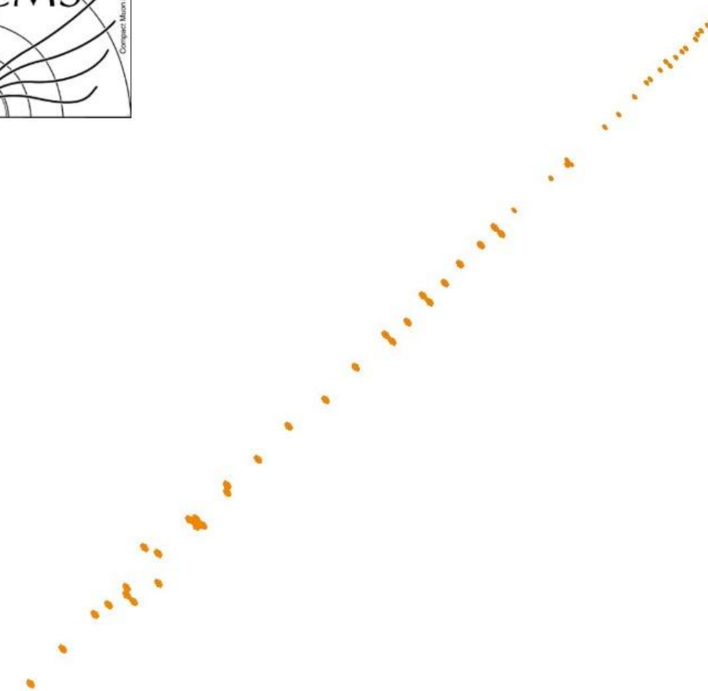
TICL Iterations

Current prototype iterations:

- track-seeded (e.g., electrons, charged hadrons)
- electromagnetic (e.g., photons)
- hadronic (e.g., neutral hadrons)
- MIP (e.g., muons)

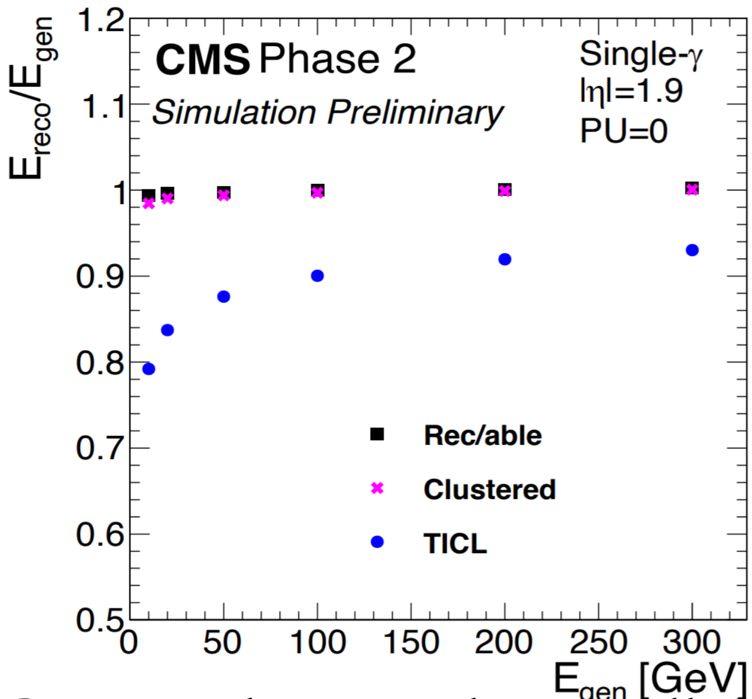
MIPs

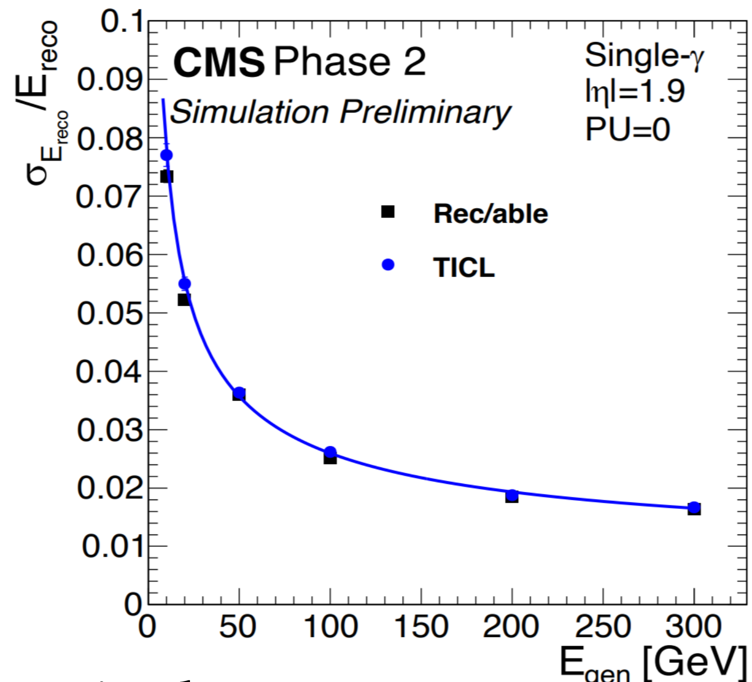
- Aims at reconstructing track-like objects within HGCAL
- The inputs are small LCs
- Cellular Automaton-based pattern recognition
- Useful to link tracksters together or to find the MIP at the beginning of an hadronic shower



A muon traversing one endcap of the HGCal detector

Electromagnetic Iteration





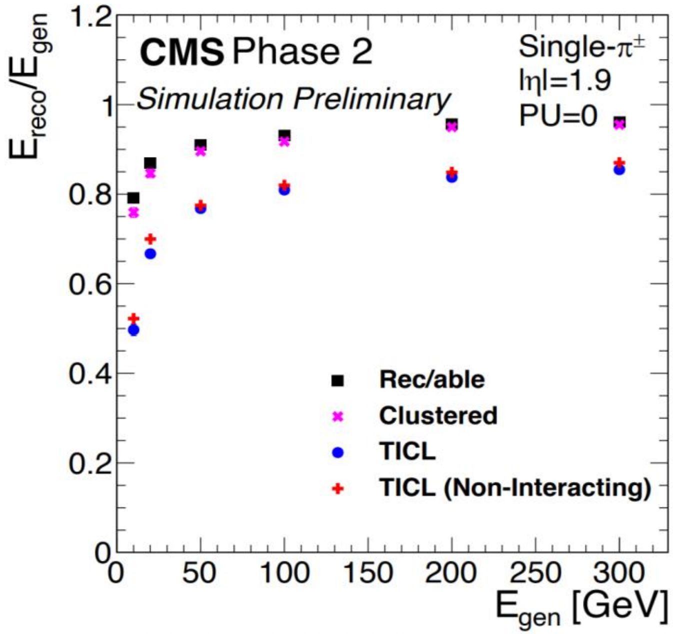
CLUE: clusters almost all deposited energy

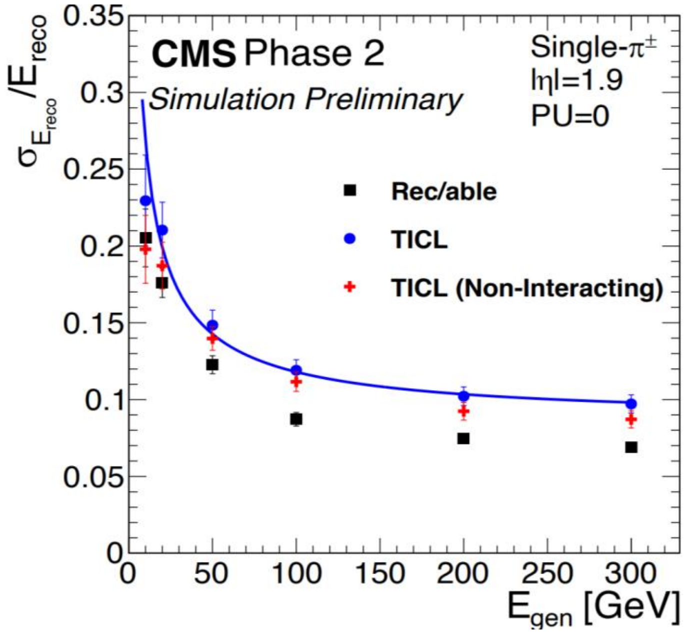
TICL showers: contain a large fraction of it

Energy resolution: similar resolution between TICL & Rec/able

	Stochastic	Constant
Rec/able	23%	0.9 %
TICL	25%	0.9 %

Hadronic Iteration





Hadronic showers can have different shapes and topologies, going from single blob to many linked blobs of energy

CLUE reconstructs almost all of the Rec/able energy, TICL a large fraction of it

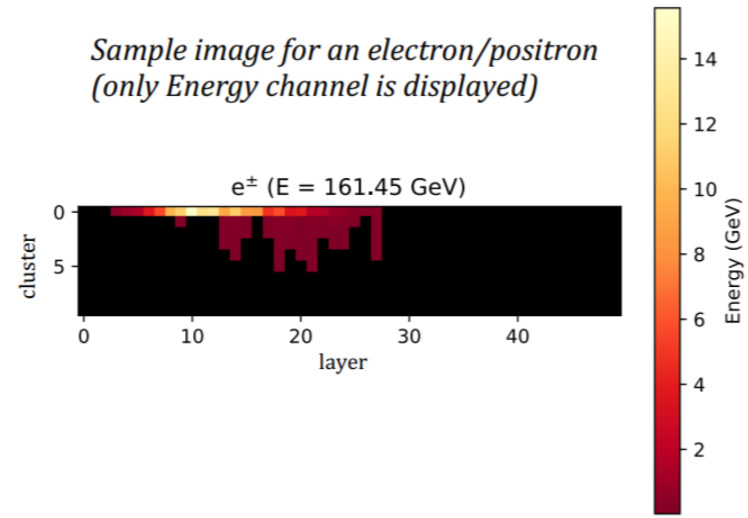
	Stochastic	Constant
Rec/able	72%	5.5 %
TICL	80%	8.7 %
TICL (Non-int/ing)	72%	8.0 %

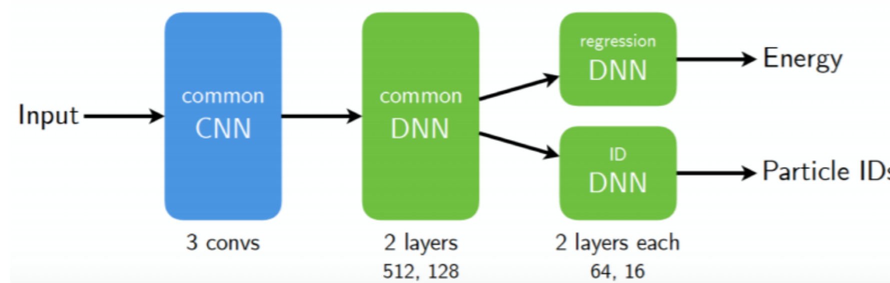
PID and Energy regression

Particle ID and Energy regression can be applied on Tracksters with a single CNN

PID probabilities and energy value as output

Sample image for an electron/positron (only Energy channel is displayed)





Validation and Testing

As TICL has been integrated in the CMS reconstruction, tests are performed automatically on daily basis and for each PR.

A validation has been developed to monitor the physics performance of 2D layer clustering and Pattern recognition

TICL as a library

Repo: https://gitlab.cern.ch/kalos

Very loose coupling with the CMSSW framework and Geometry

- templated design allowed its usage out-of-the-box for another endcap calorimeter with different geometry in half a day of work

Assumption on highly granularity in the transverse plane and in the longitudinal sampling (closeby layers)

Started by making CLUE standalone

- Made it easy to use it on test beam data
- https://gitlab.cern.ch/kalos/clue
- Ramp up work starting from end of August/September with Thursday biweekly meetings with the rest of the TICL team

Summary

- The reconstruction in High Granularity Calorimeters at future colliders has many unprecedented challenges:
  - A "tracking" device with high hit multiplicity and precise time information
- TICL is being designed with throughput and modern sw techniques
  - A unique opportunity since it is impossible to reuse software
  - Fertile ground for the exploitation of neural networks
  - Integrated in CMSSW but loosely coupled to it
  - Allows to exercise the code frequently, validate it and profit from the CMS software development infrastructure
  - TICL is being developed with parallelism in mind
    - many parts have been/are being ported to GPUs
- Making it a common library would benefit experiments at future colliders and contribute to TICL's improvement by using it with different topologies/geometries
- Increase the interaction with groups developing similar PF-like reconstruction
  - e.g. fruitful discussion after our presentation at CLIC workshop