ML-based pattern recognition for CLD/IDEA

Dolores Garcia, Brieuc Francois, Michele Selvaggi

FUTURE CIRCULAR COLLIDER

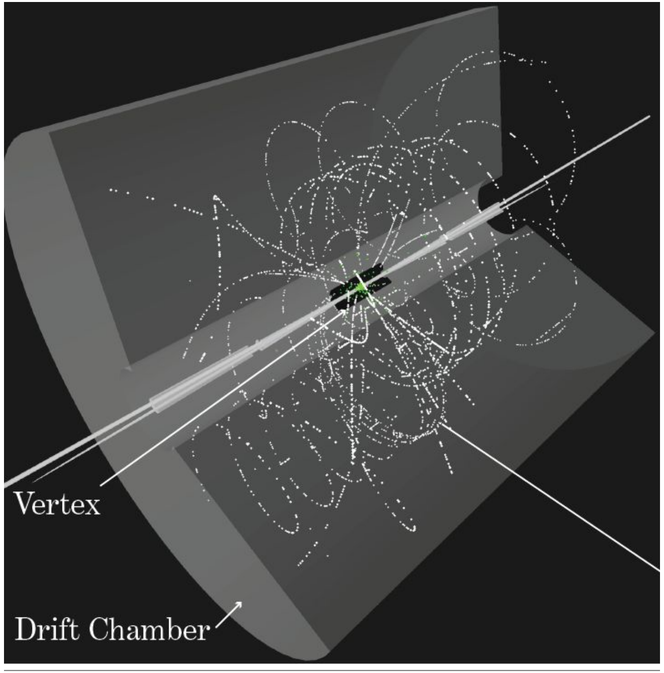
CERN

Objective

Obtain a pattern recognition algorithm for IDEA

- Classic algorithms are not easily applicable due to the left/right ambiguity
- Algorithm should be easily adaptable to new geometries
- As a result, the same pipeline can be applied to CLD

Disclaimer: these results were previously presented in more detail (key4hep implementation) in different full sim meetings



Dataset

Generated events of Z→qqbar 91GeV without background using Pythia

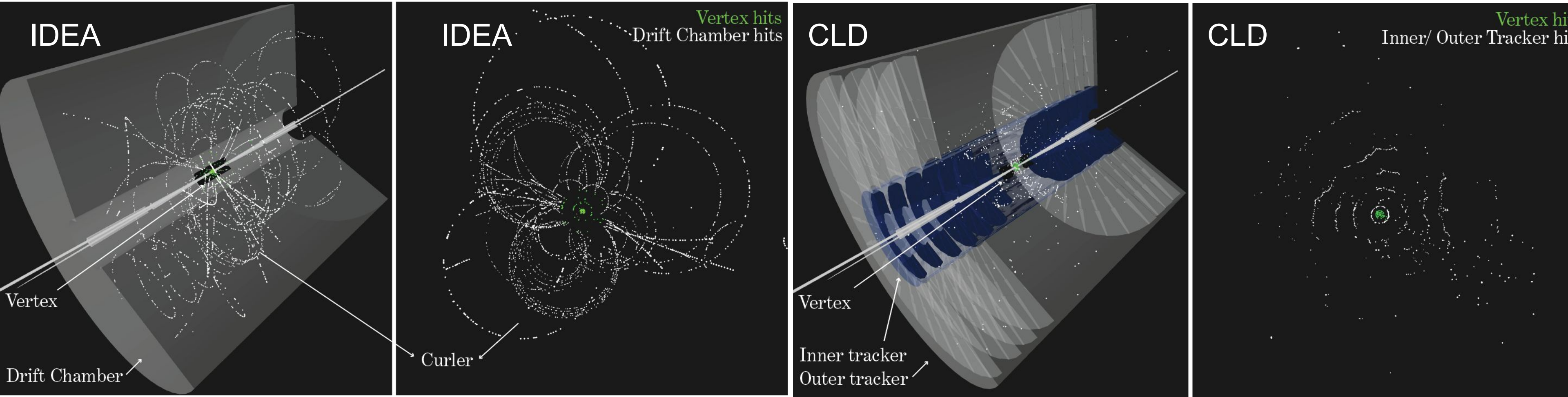
ddsim with CLD_02_v05 (key4hep 2024-05-09) [Note: TrackerHitRelations is broken in nightlies with v06 Github issue]

Store hits from

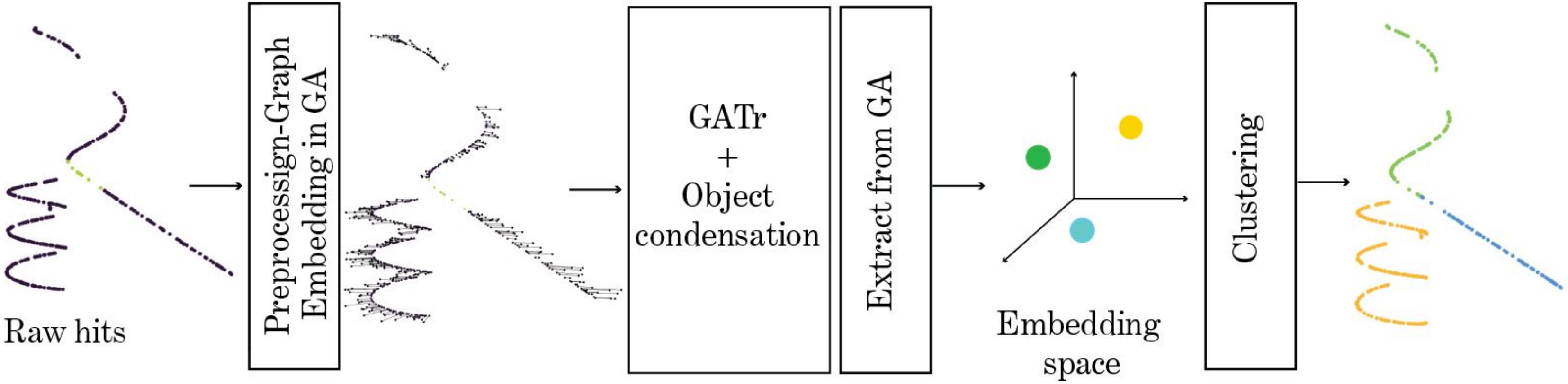
- Vertex Barrel, Vertex Endcap
- Inner Tracker Barrel, Inner Tracker Endcap
- Outer Tracker Barrel, Outer Tracker Endcap

For validation store MC association using “TrackerHitRelations”

```
! main03.cmnd.
! https://github.com/HEP-FCC/FCC-config/blob/winter2023/FCCee/Generator/Pythia8
! This file contains commands to be read in for a Pythia8 run.
! Lines not beginning with a letter or digit are comments.
! Names are case-insensitive  -  but spellings-sensitive!
! The settings here are illustrative, not always physics-motivated.

! 1) Settings used in the main program.
Main:timesAllowErrors = 5          ! how many aborts before run stops
Stat:showProcessLevel = on
Main:numberOfEvents = 100         ! number of events to generate
Random:setSeed = on
! 2) Settings related to output in init(), next() and stat().
Init:showChangedSettings = on      ! list changed settings
Init:showChangedParticleData = off ! list changed particle data
Next:numberCount = 100             ! print message every n events
Next:numberShowInfo = 1            ! print event information n times
Next:numberShowProcess = 1         ! print process record n times
Next:numberShowEvent = 0           ! print event record n times

! 3) Beam parameter settings. Values below agree with default ones.
Beams:idA = 11                     ! first beam, e
Beams:idB = -11                    ! second beam, e

! Beam energy spread: 0.132% x 45.594 GeV = 0.0602 GeV
Beams:allowMomentumSpread  = off


! 4) Hard process : Z->qq at Ecm=91 GeV
Beams:eCM = 91.188  ! CM energy of collision


WeakSingleBoson:ffbar2gmgZ = on
23:onMode = off
23:onIfAny = 1 2
```



Algorithm



The algorithm is independent of the detector geometry (same pipeline for IDEA)

- Embedding of raw hits
- Graph neural network
- Clustering step → outputs are Track candidates (collection of hits)

Performance for complex events CLD: tracking efficiency

Definitions from [CLD paper]

Track hit purity: is the ratio of the number of hits in the track that belong to the MC particle and the total number of hits of the reconstructed track

Track hit efficiency: is the ratio of the number of hits in the track that belong to the MC particle and the total number of hits this particle left in the detector

Reconstructable particle: stable at generator level, pT>100 MeV, |cosθ|<0.99 and at least 4 unique hits

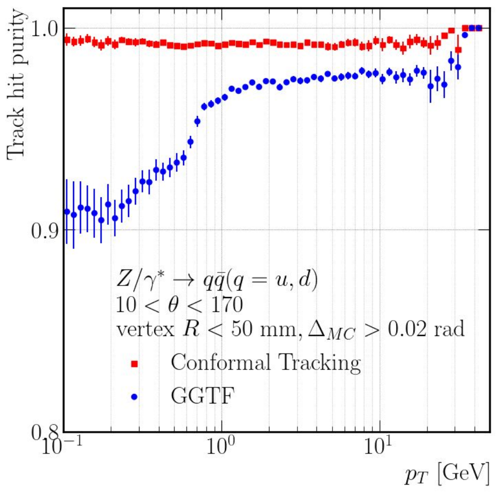
Compare with SiTracks_Refitted

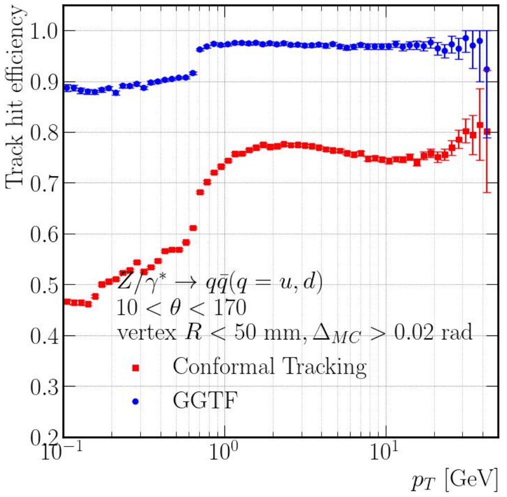
Evaluated on 10k events

Fakes: no MC is assigned to the reconstructed track

The fakes can not be evaluated per pT bin since the track is not reconstructed but the total number of fakes is:

- ML: 4.2%
- Conformal: 4.4%



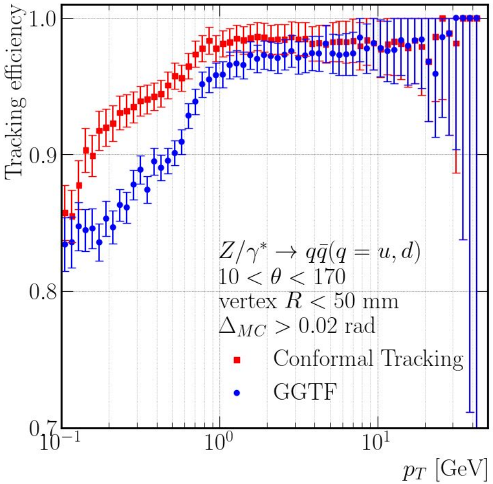


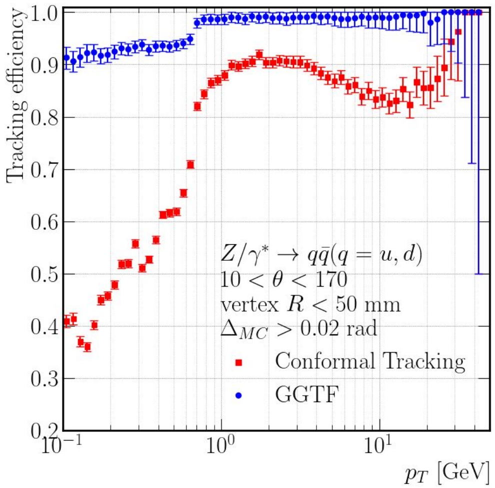
Performance for complex events CLD: tracking efficiency

Definitions from [CLD paper]

Efficiency def 1. Percentage of reconstructable particles with track hit purity >75% (track segments)

Efficiency def 2. Percentage of reconstructable particles with track hit purity >50% and track hit efficiency > 50%



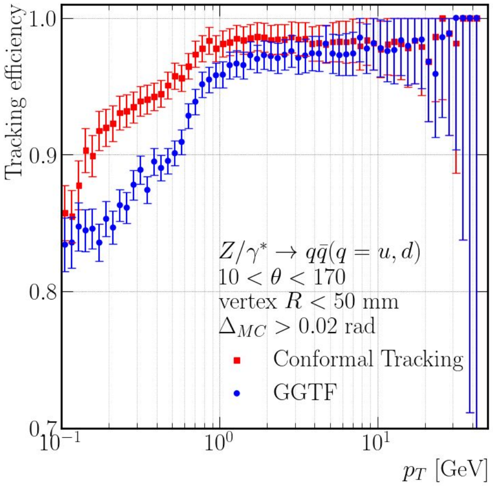


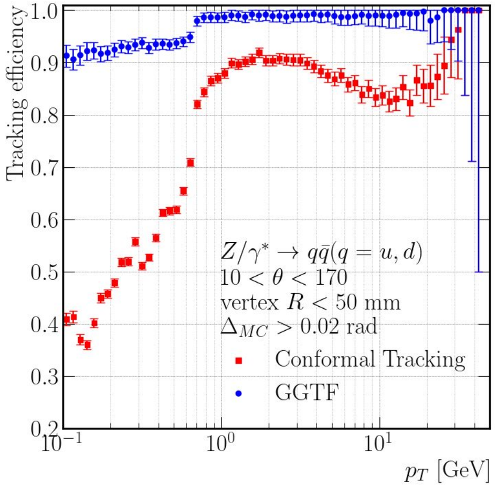
Performance for complex events CLD: tracking efficiency

Definitions from [CLD paper]

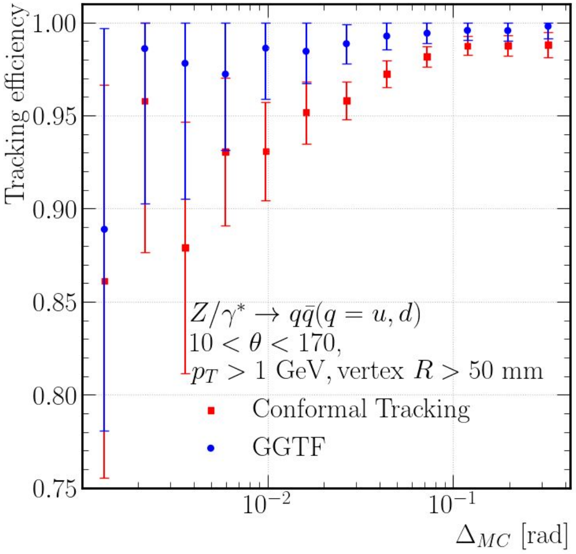
Efficiency def 1. Percentage of reconstructable particles with track hit purity >75% (track segments)

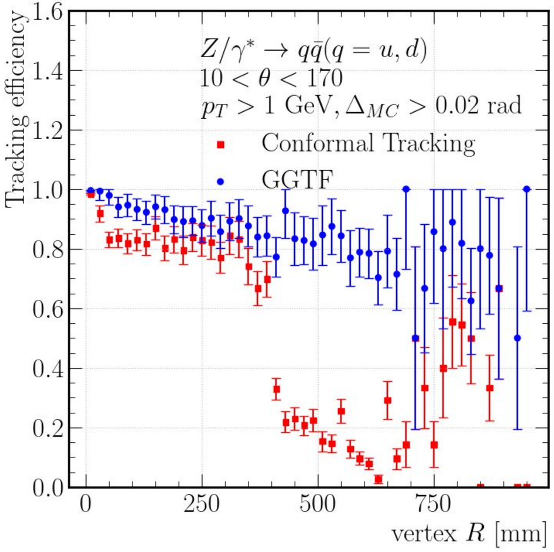
Efficiency def 2. Percentage of reconstructable particles with track hit purity >50% and track hit efficiency > 50%





Performance for complex events CLD: tracking efficiency



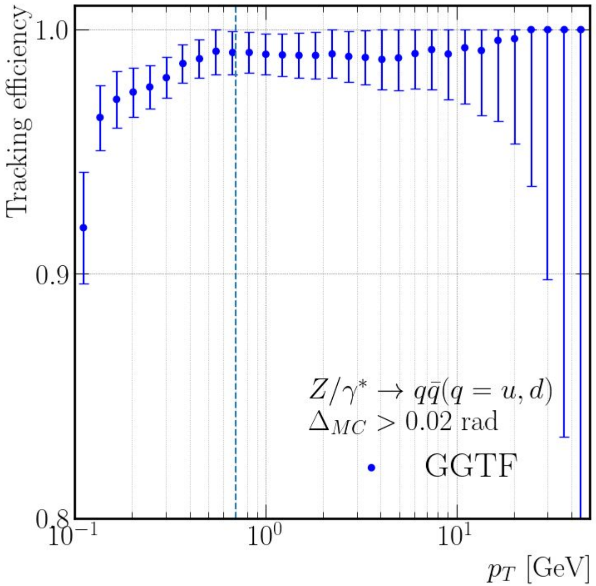


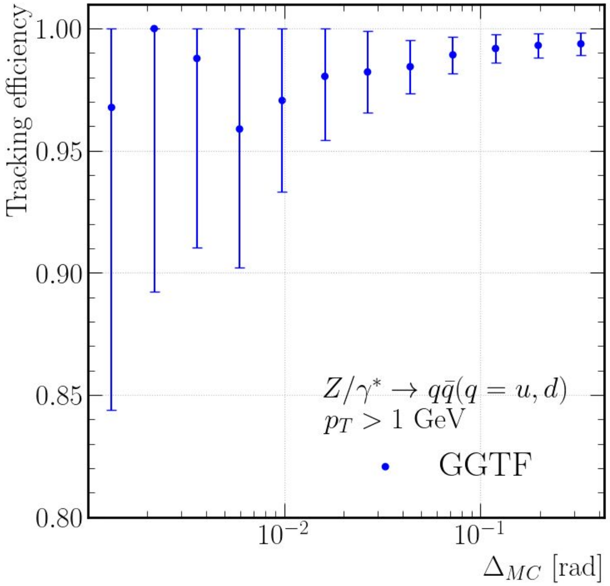
Efficiency as a function of particle proximity:

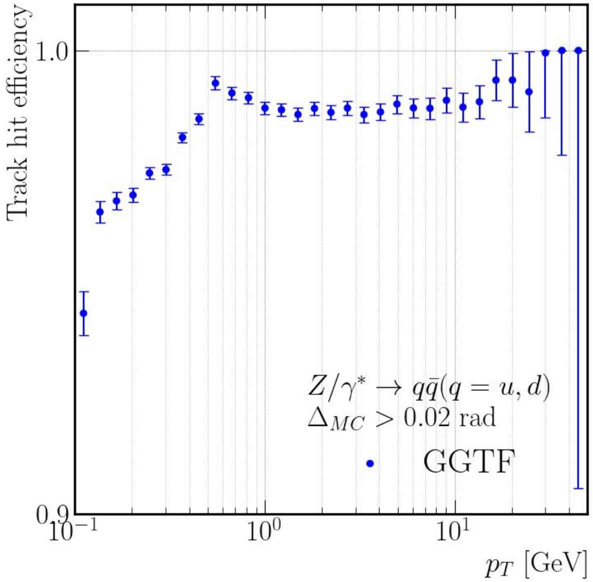
ΔMC = $\sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$

Efficiency as a function of production vertex radius

Performance for complex events IDEA: tracking efficiency







Tracking efficiency def 2)

Tracking efficiency vs ΔMC

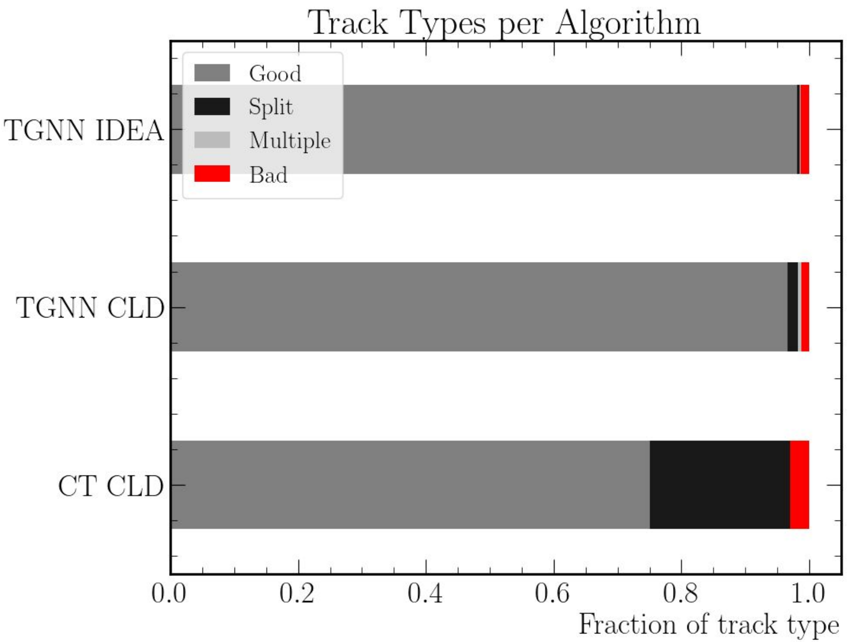
Track hit efficiency

Performance for complex events IDEA vs CLD

Track hit purity (THP) Track hit efficiency (THE)

- Good: THP>50 % THE >50 %
- Split : THP>50 % THE <50 % (only a fraction of the track is reconstructed)
- Multiple: : THP<50 % THE >50 %
- Bad: THP<50 % THE <50 %

Overall, more splitted tracks are recovered using the TGNN method

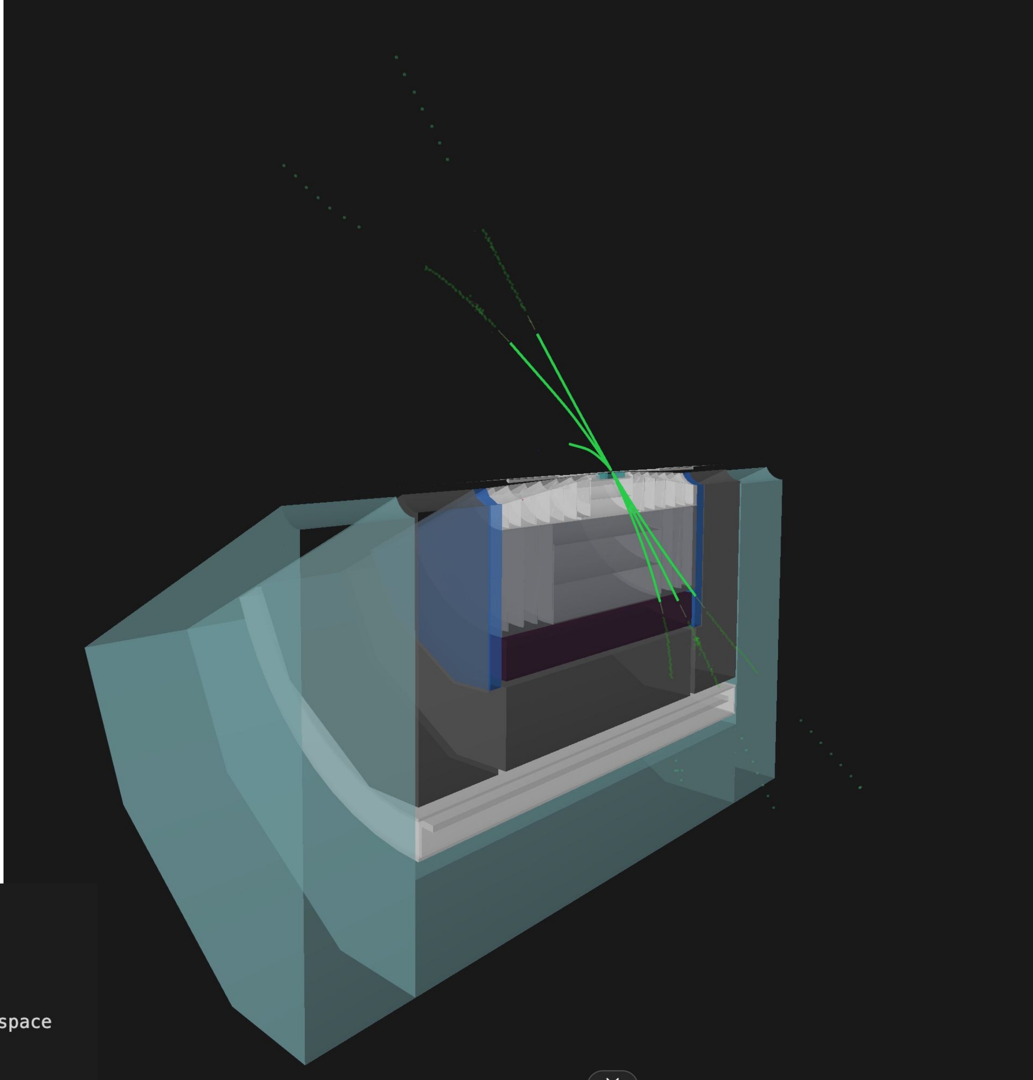


Summary

- Performance is improved in terms of efficiency compared to the Conformal tracking 'out of the box'
- The purity is lower as the tracks include more hits but remains high
- The effect on the track fit still needs to be evaluated
- A similar pipeline is available in key4hep for IDEA so it could be adaptable for CLD
- Preparing a paper on this to be submitted to ICLR

Z→ ττ→(3μ)(3μ)

- Force pythia decay
- Same data for CLD (02_v06) and IDEA (01_v02)
- Performance comparison



```
WeakSingleBoson:ffbar2gmZ = on
23:onMode = off
23:onIfAny = 15
15:onMode = off
15:AddChannel =  on 0.00001 0 13 13 -13 ! forced tau -> 3mu decay, pure phase space
```

Efficiency for Z→ ττ→(3μ)(3μ)

- Tracking efficiency defined as hit purity>50% (in order to be able to compare IDEA on the same grounds)
- Very similar performance with the 3 algorithms
- Gains expected in IDEA (currently there is some merging on tracks that are close due to imbalance in training)