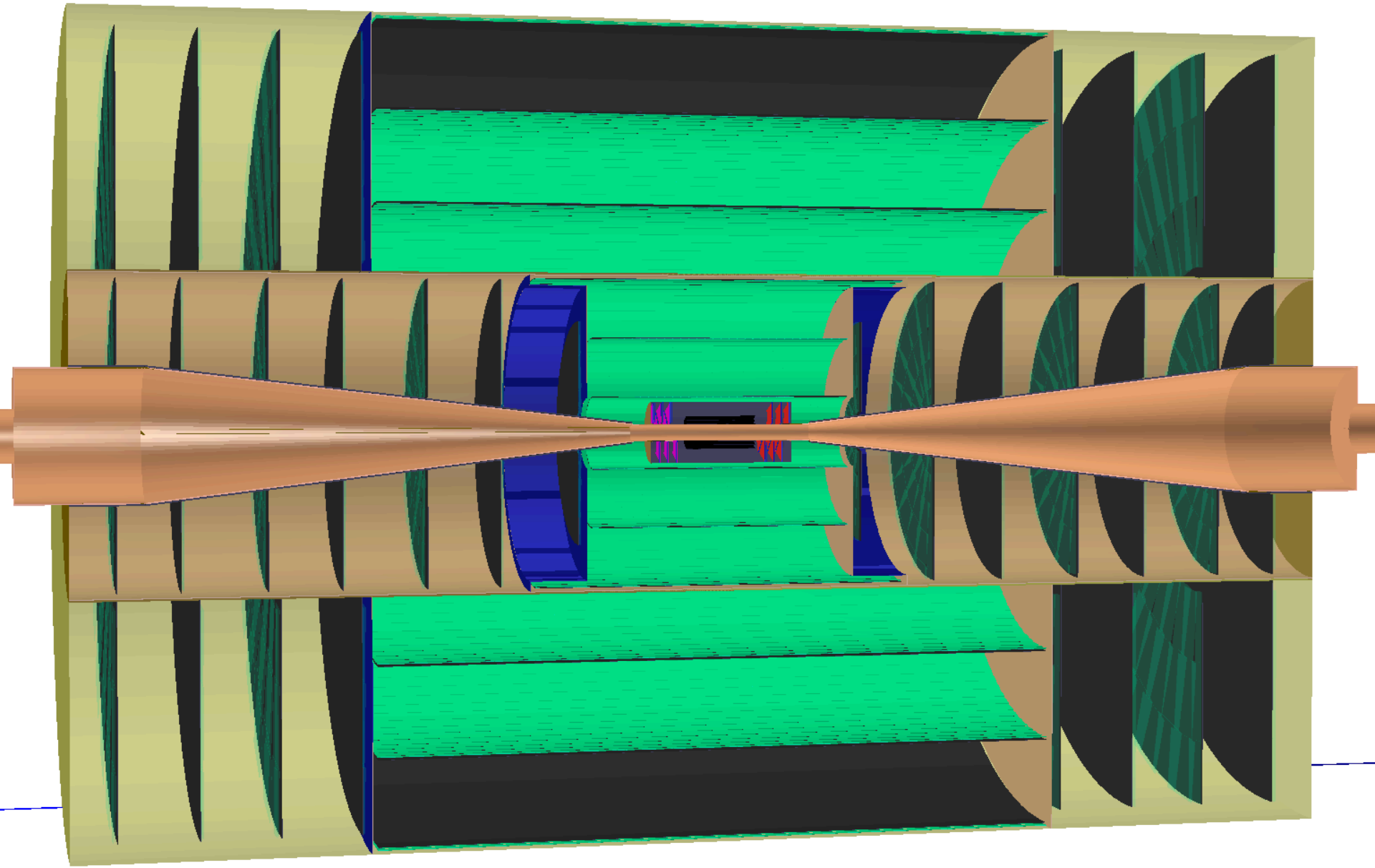
Tracker simulation and reconstruction

Rosa Simoniello

Outline

- Overview of the status of tracker in simulation
  - CLIC_o3_v06: last up-to date model with engineer layout
  - CLIC_o2_v04: frozen model to develop tracking
- News about tracking
  - New directions and ideas

CLIC_o3_v6



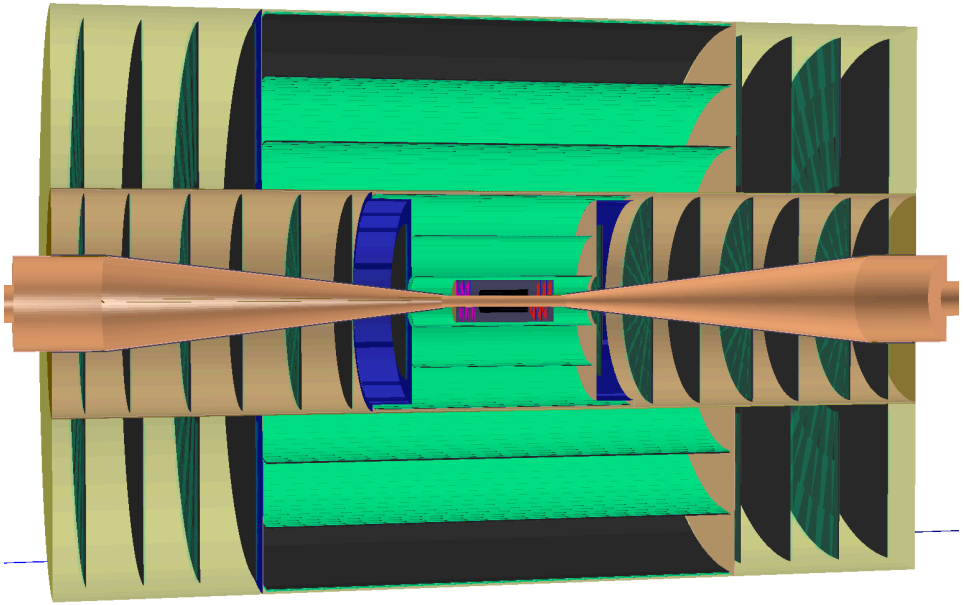
Size and number of hits

Inner barrels	IB1	IB2	IB3
R [mm]	127	340	554

Inner disks	ID1	ID2	ID3	ID4	ID5	ID6	ID7
z [mm]	524	808	1093	1377	1661	1946	2190

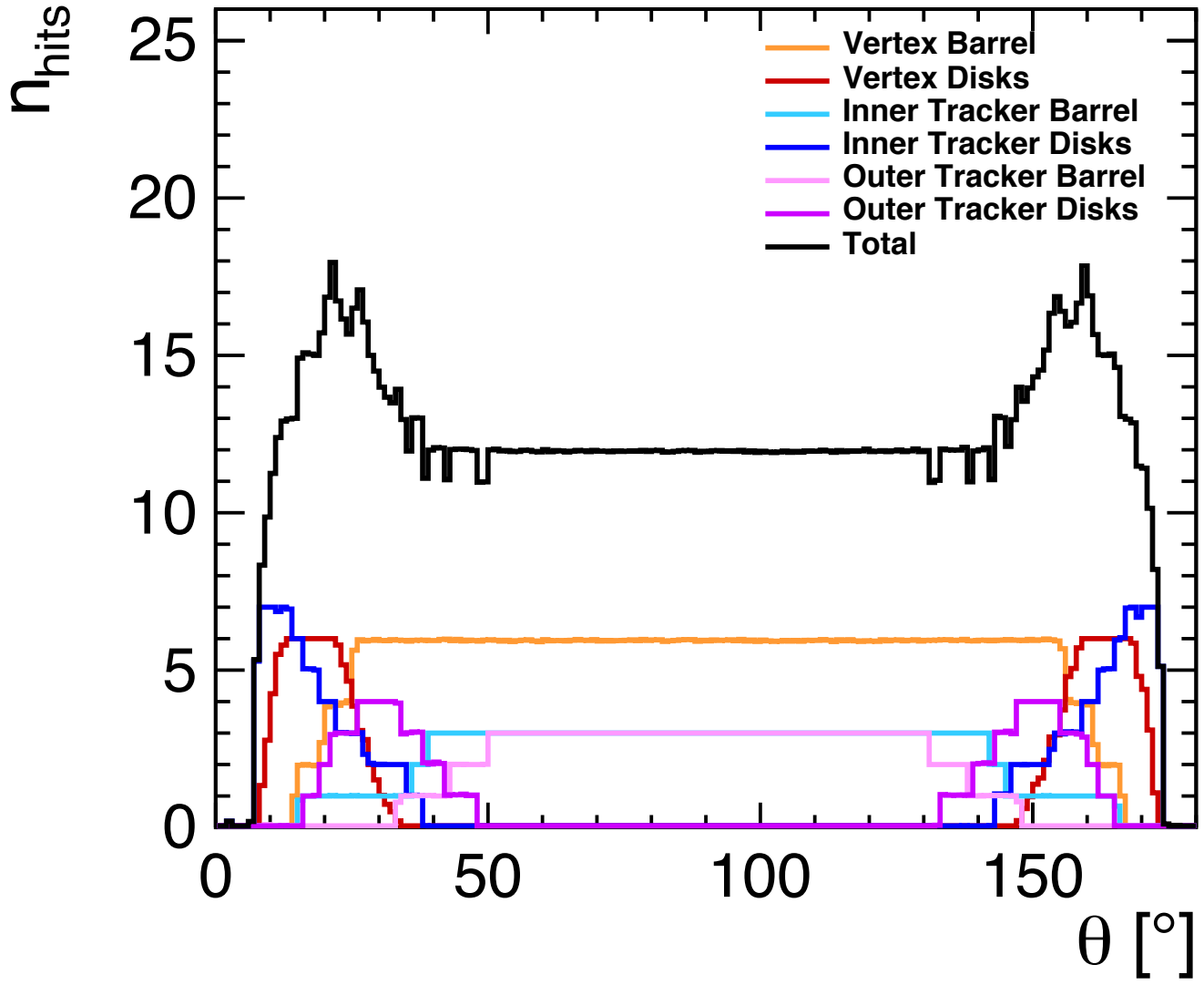
Outer barrels	OB1	OB2	OB3
R [mm]	819	1153	1486

Outer disks	OD1	OD2	OD3	OD4
z [mm]	1310	1617	1883	2190



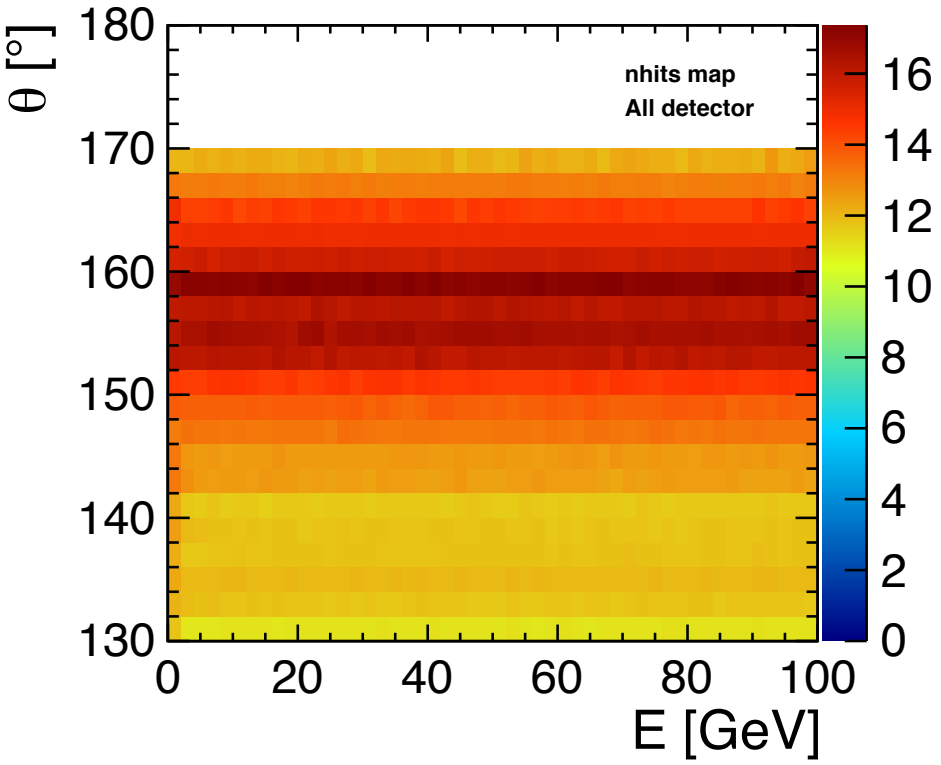
Number of hits

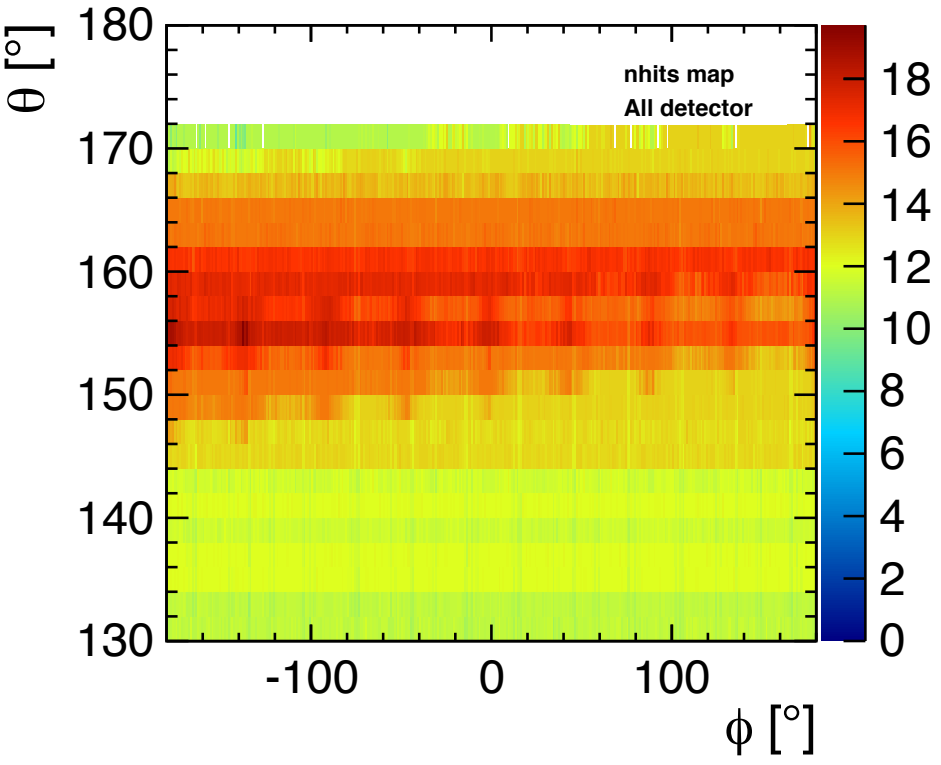
- Good coverage, few missing hits in transition between barrels and disks



Hit map

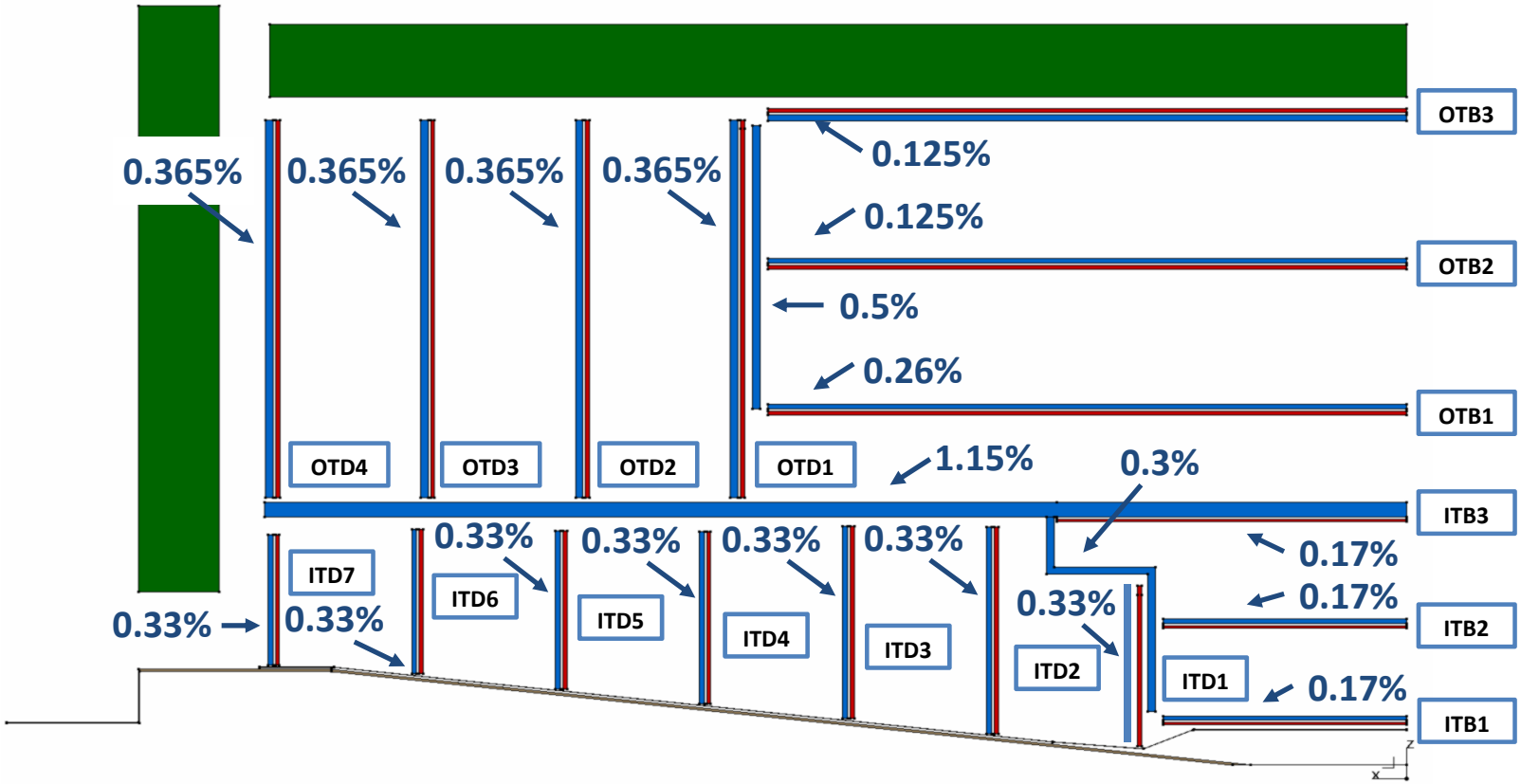
- N of hits investigated as a function of particle energy (in case at low momentum the particle “curles” into holes
  - To be study also as a function of pT
- Dependence on phi due to spiral vertex geometry in the disks





Material budget

- Total material budget per layer for barrels (w/o st): **1.19% - 1.145%**
- Total material budget per layer for disks: **1.35% - 1.385%**
- **0.3% - 0.5%** for take out of services



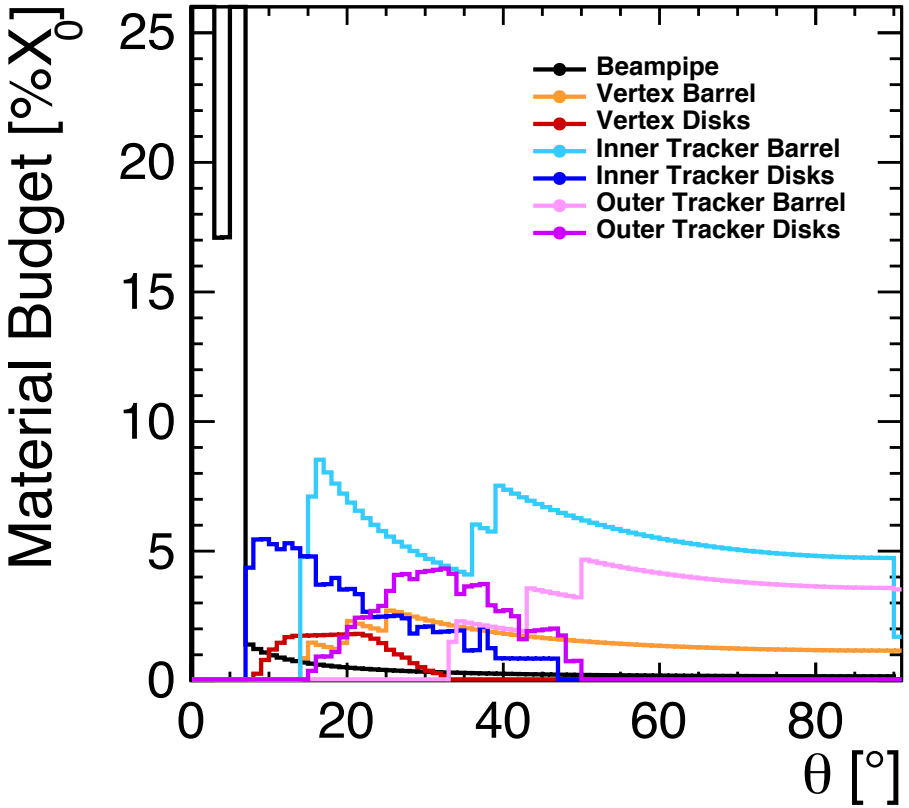
Red - Module + Cold Plate+ Power Bus = 1.02%X0

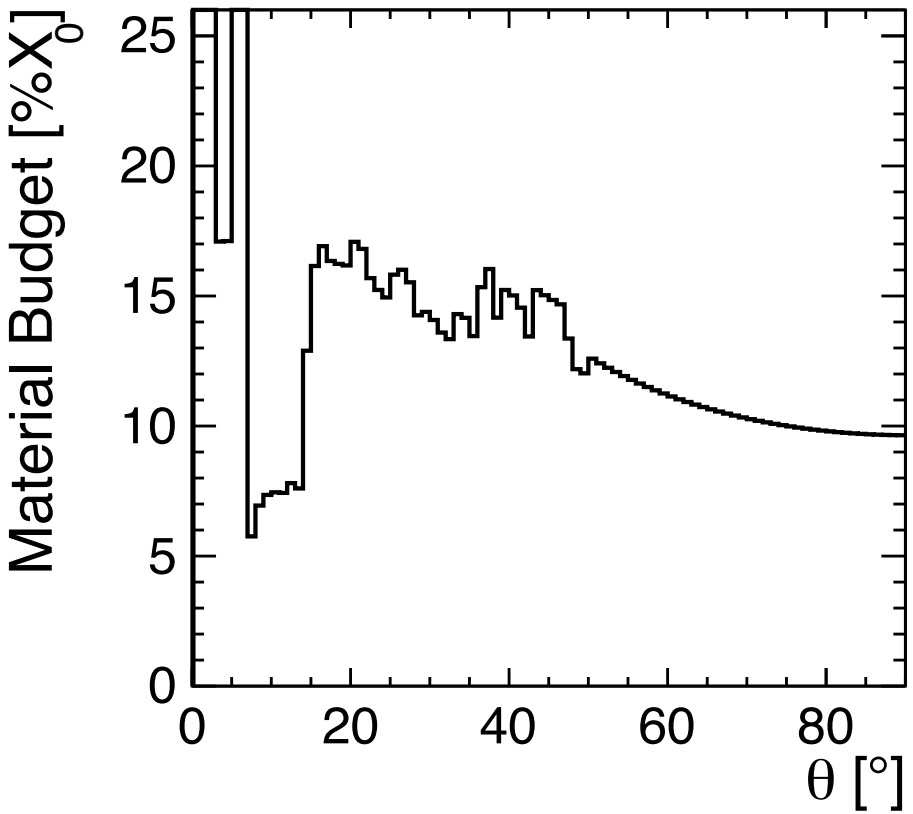
Dark Blue– Carbon fiber supports

Work is ongoing

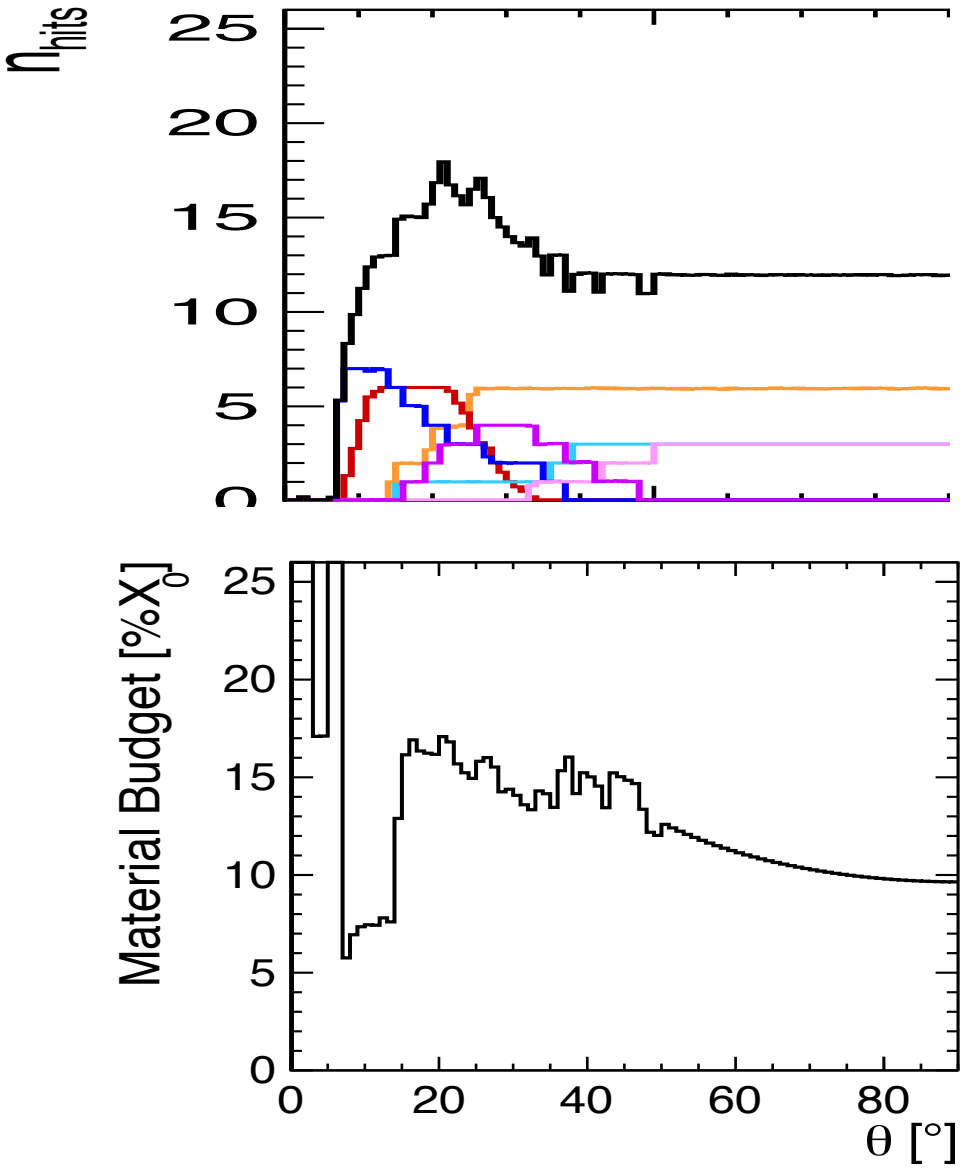
Material budget

- Inner tracker barrel includes also the support tube
- Beam pipe gives peak of material ~5-6° due to its conical shape



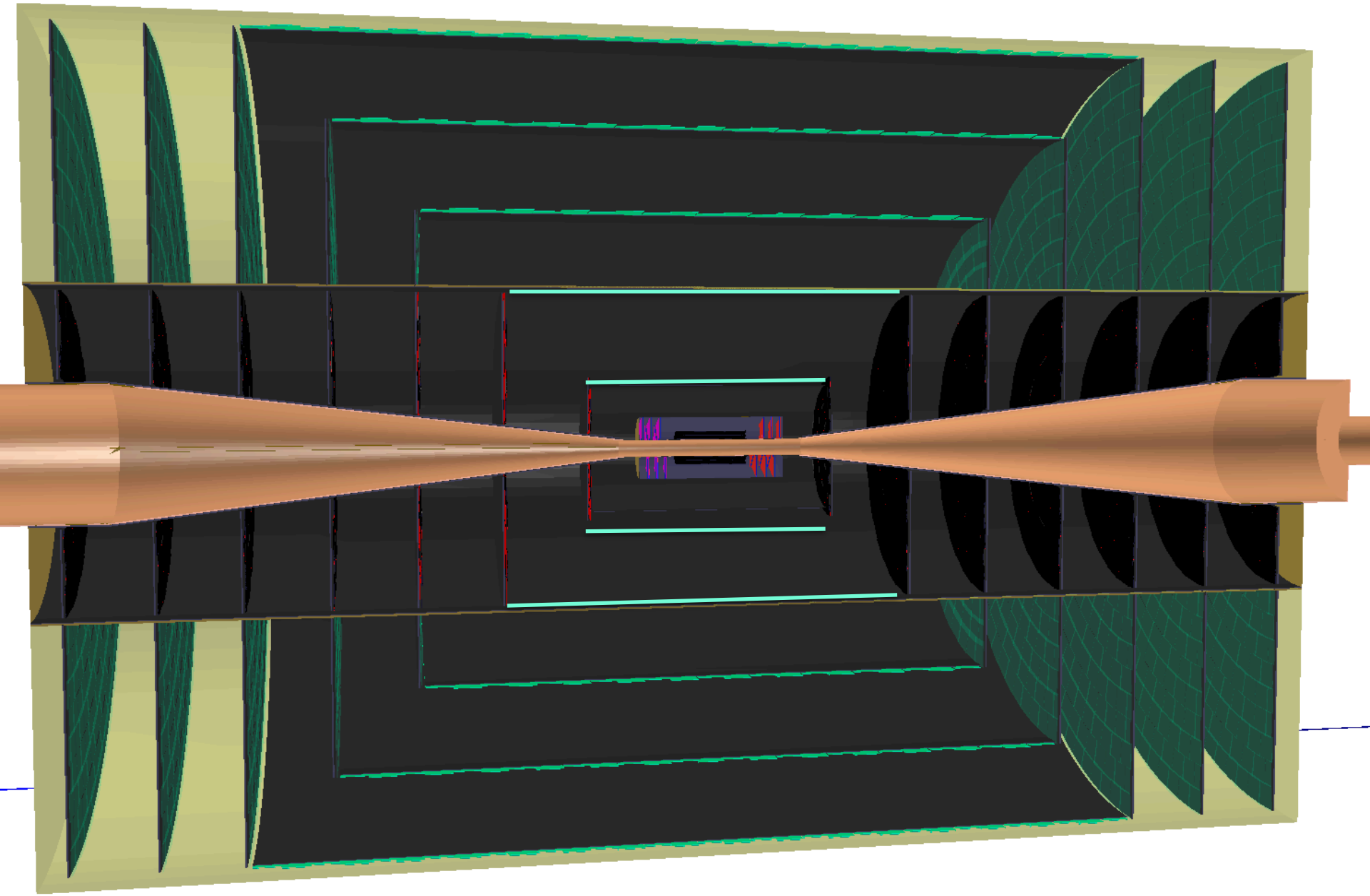


Summary of nhits and mb



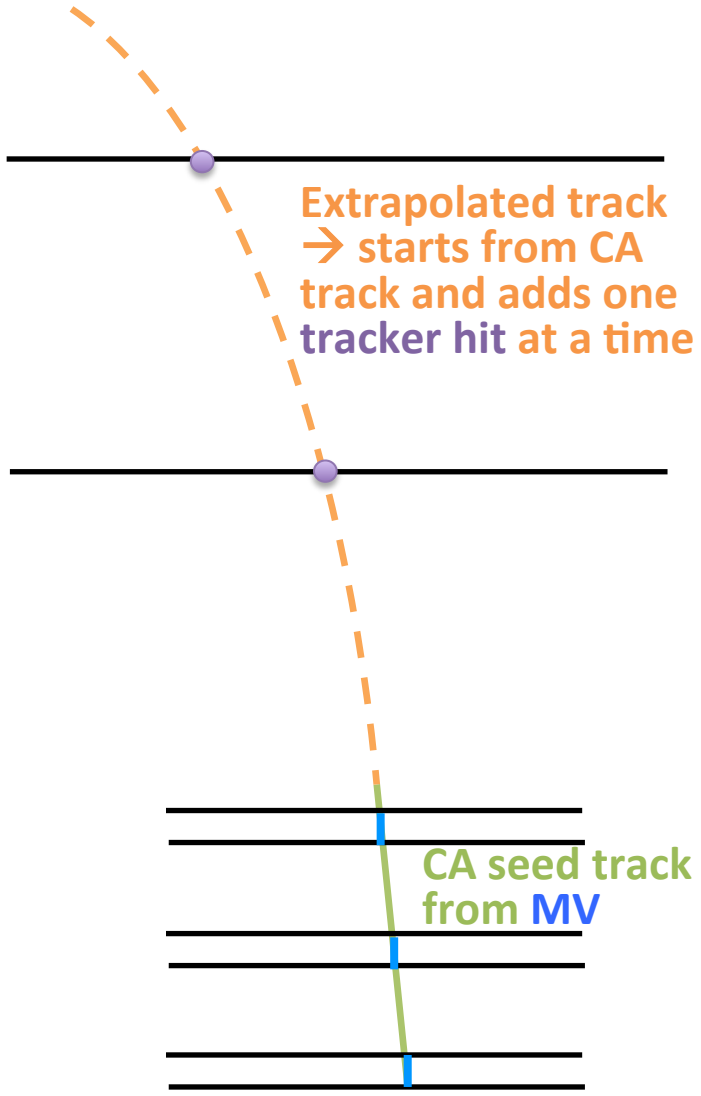
- The region with more material budget is also the region with more hits

CLIC_o2_v4



Pattern recognition I – Reminder

- For tracks traversing the vertex barrel
  1. Compute *mini vectors* (MV) exploiting the double layer structure of the vertex barrel
  2. Run *Cellular Automaton on MV* → obtain vertex tracks
  3. Use vertex tracks as seed for track *extrapolation* to Inner and Outer Tracker in Barrel and Disks
- To be combined with *independent Cellular Automaton in the forward*

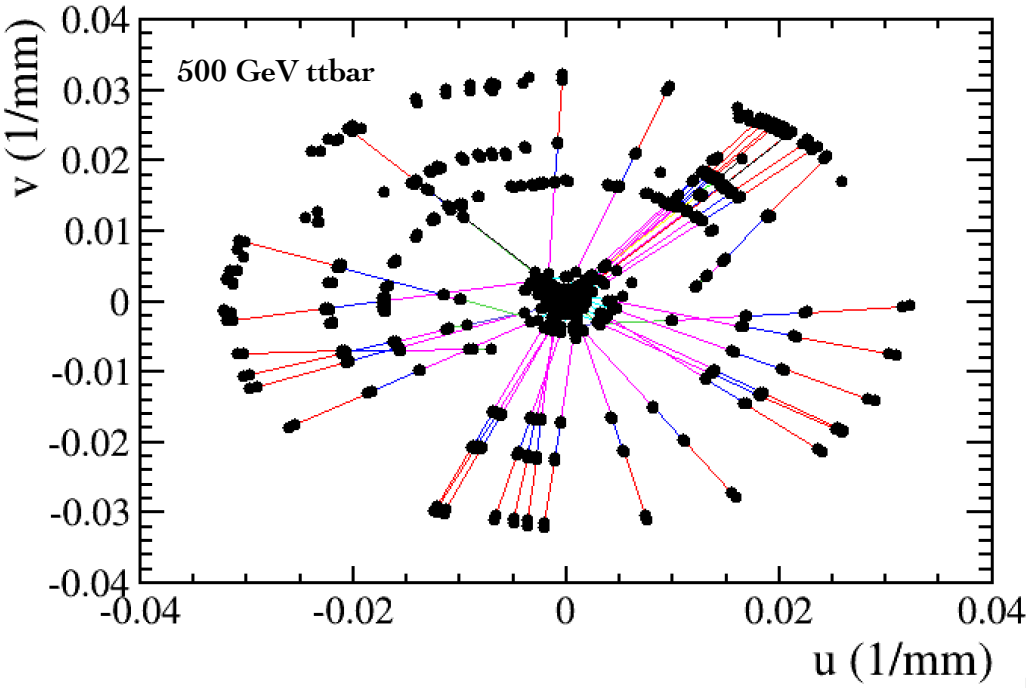


Mini Vectors:
Create a mini vector out of 2 hits on adjacent layers (d < 5 mm) with $\delta\theta < 1°$

Pattern recognition II – Reminder

- *Conformal mapping:* coordinate transformation that preserves local angles
  - (x,y)-plane in the (u,v)-plane
  - Tracks are straight lines
- *Run CA sequentially on all hit collections*
  - same approach in the full tracker system
  - no sub-track combination needed
- Track ambiguity resolved with linear fit

$$\begin{cases} u = x/(x^2+y^2) \\ v = y/(x^2+y^2) \end{cases}$$



Tracking

- For both methods, good efficiency shown at CLICdp August workshop
  - Now Conformal mapping tested also with background overlay
- Decision to merge the 2 methods:
  - Seed vertex track with conformal mapping + Cellular Automaton
  - Extrapolate the tracks in the tracker → better fit propagation including material effects
  - Saved not used hits and try something else on these hits to recover late decays → few ideas available

→ Very model independent, hopefully easy transition between detector models

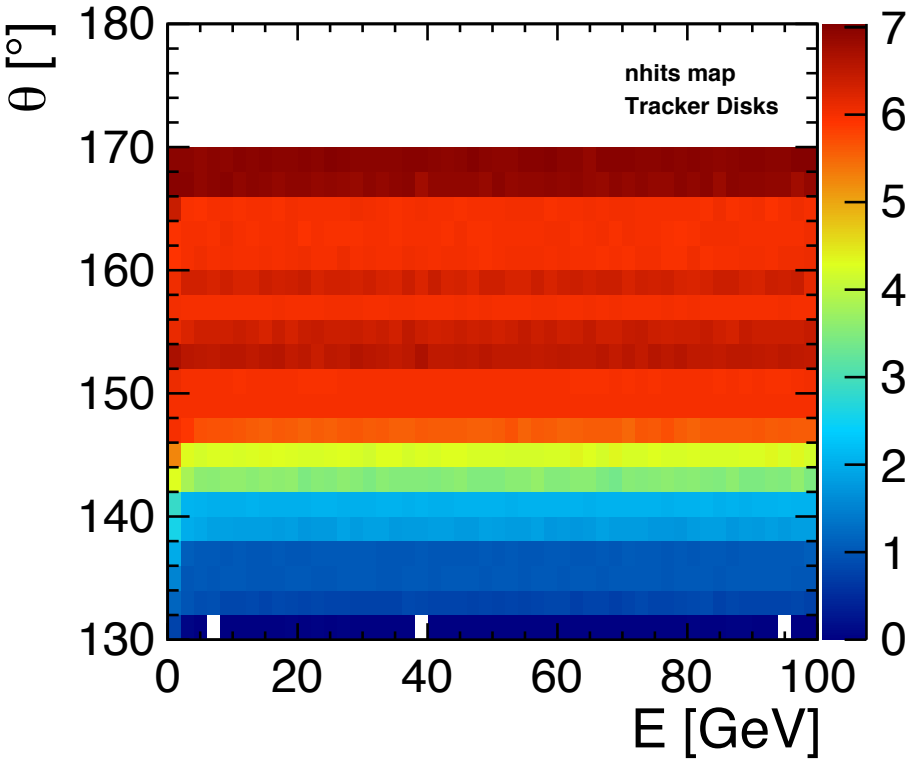
→ Test just started, more checks need to make sure that everything is working properly (n hits in the track, purity, fakes, etc…)

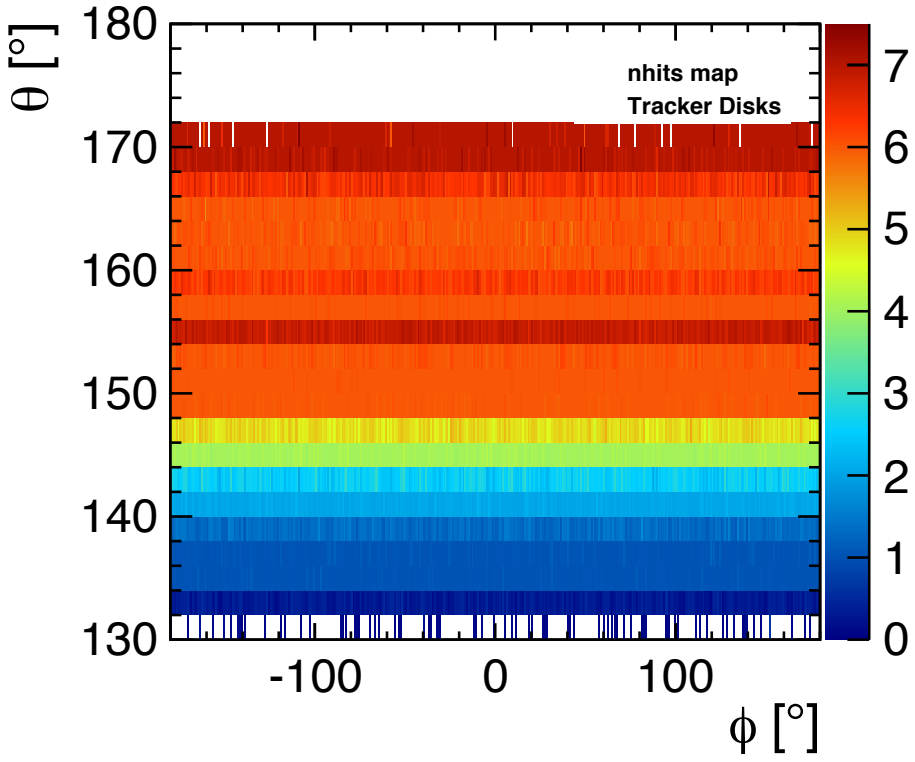
- Time spent mostly in track fit → few improvements in AIDATT to test

Conclusion

- The tracker model in the full simulation is up-to-date
- The track reconstruction is being finalised
  - New idea to combine Conformal Mapping with Tracker Extrapolator is under test → potentially very flexible to run on different detector models
  - More metric plots need to be added
- Coming soon: results by Emilia Leogrande
  - Initial task: validation with single μ (at the moment low stat)
  - Run for the first time on new tracker model in CLIC_o3_v06
  - Results to be investigated: some inefficiency in fit

BACK-UP

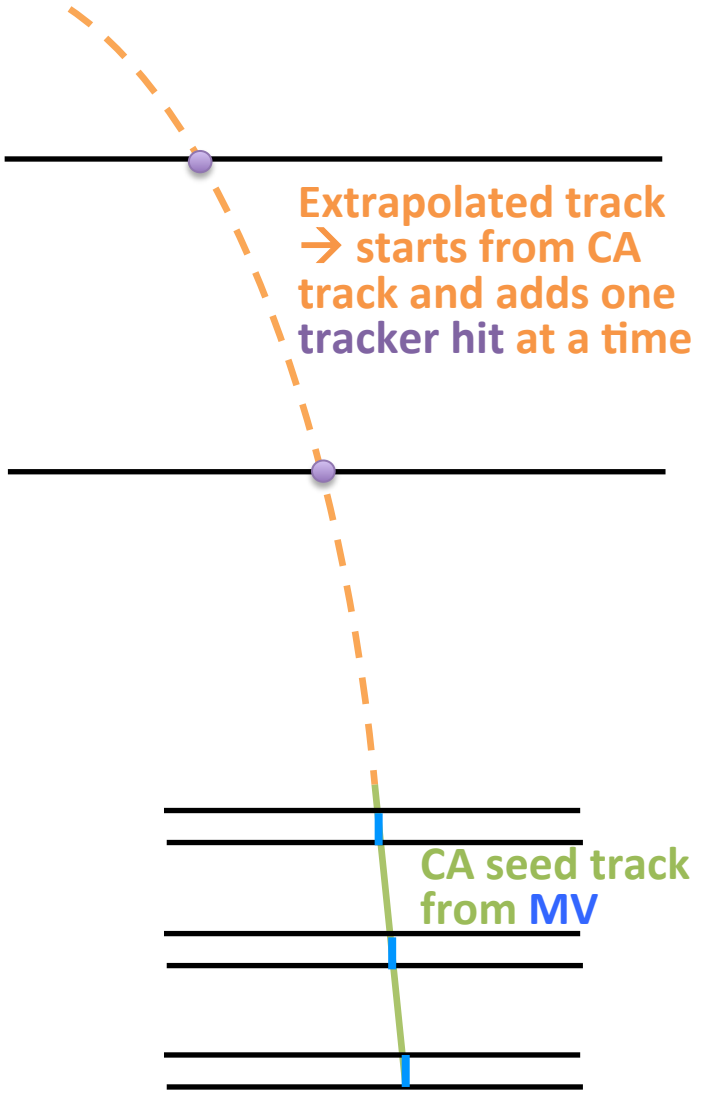

nhits map
Tracker Disks
θ [°]
180
170
160
150
140
130
0
20
40
60
80
100
E [GeV]
7
6
5
4
3
2
1
0


nhits map
Tracker Disks
θ [°]
180
170
160
150
140
130
-100
0
100
ϕ [°]
7
6
5
4
3
2
1
0

Pattern recognition I – Reminder

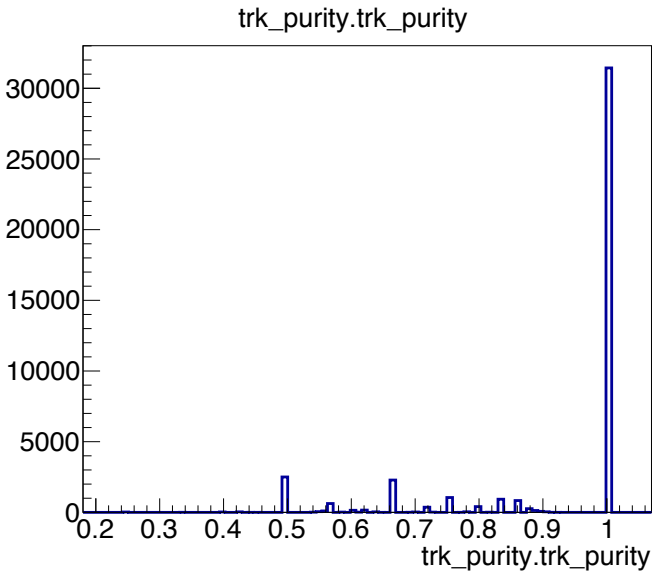
- For tracks traversing the vertex barrel
  1. Compute *mini vectors* (MV) exploiting the double layer structure of the vertex barrel
  2. Run *Cellular Automaton on MV* → obtain vertex tracks
  3. Use vertex tracks as seed for track *extrapolation* to Inner and Outer Tracker in Barrel and Disks
- To be combined with *independent Cellular Automaton in the forward*

Mini Vectors:
Create a mini vector out of 2 hits on adjacent layers (d < 5 mm) with $\delta\theta < 1°$

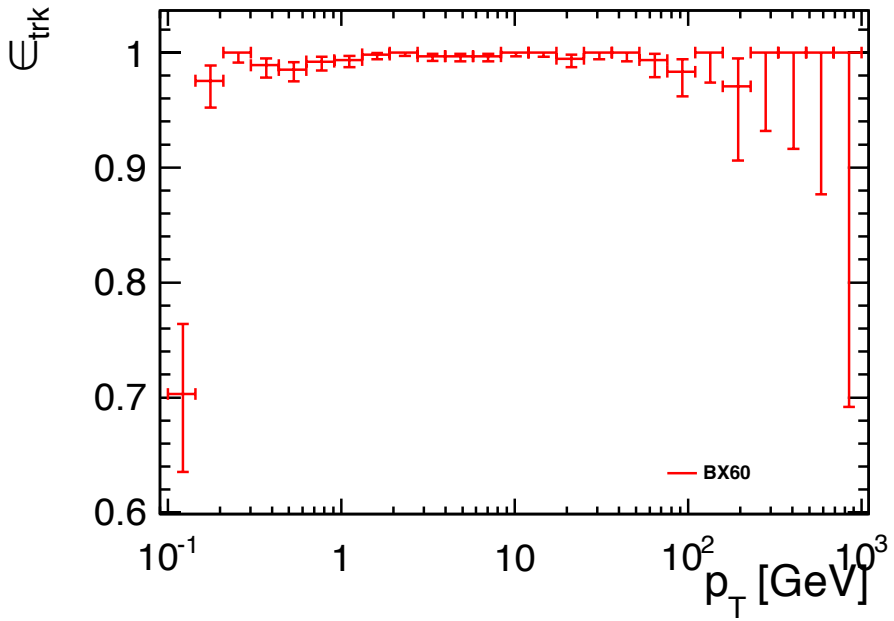


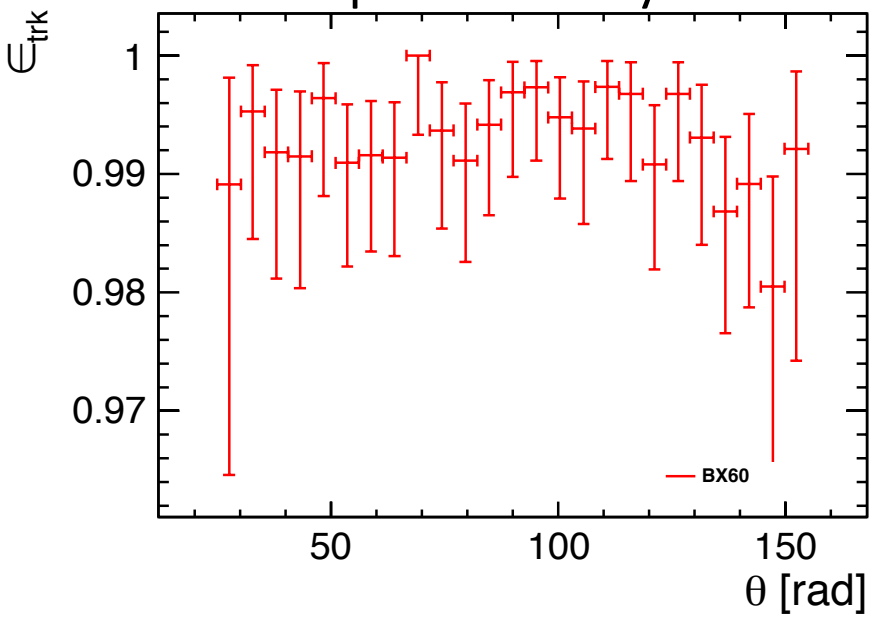
Pattern recognition I – Results

- 3 TeV ttbar sample with 60BX overlayed
  - For new samples: 30BX vs the 60BX?
- Reoptimisation necessary
- Purity ok but room for improvement
- Not run any profiler yet to optmise timing (anyway no longer of current tracking)
  - Optimised loop in hits
  - Extrapolation procedure takes time



Mini Vector + Cellular Automaton + Extrapolation only

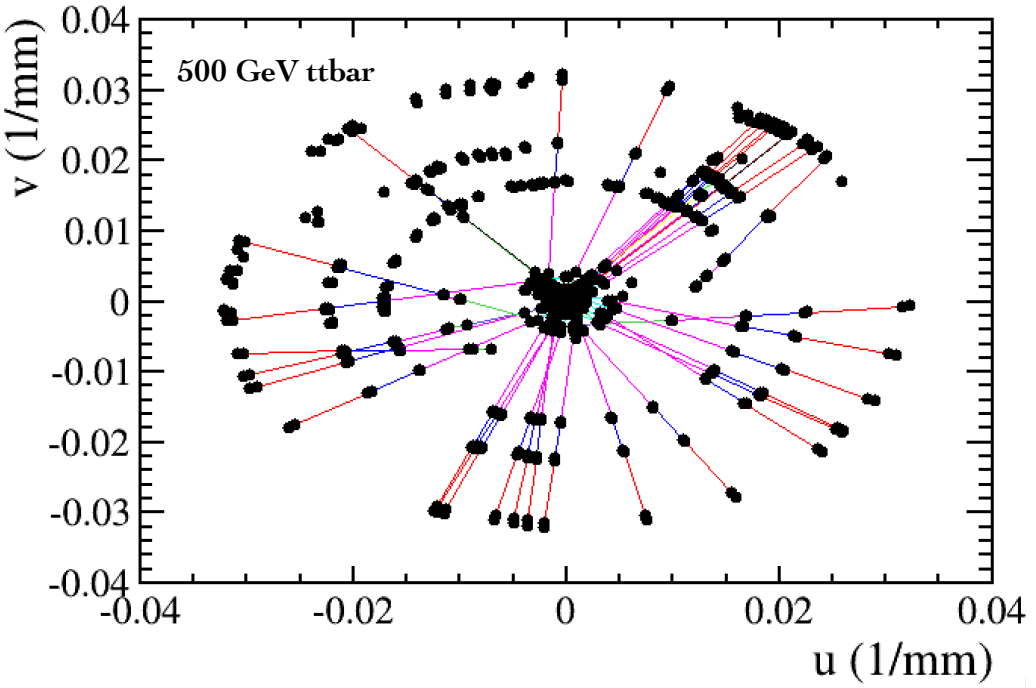




Pattern recognition II – Reminder

- *Conformal mapping:* coordinate transformation that preserves local angles
  - (x,y)-plane in the (u,v)-plane
  - Tracks are straight lines
- *Run CA sequentially on all hit collections*
  - same approach in the full tracker system
  - no sub-track combination needed
- Track ambiguity resolved with linear fit

$$\begin{cases} u = x/(x^2+y^2) \\ v = y/(x^2+y^2) \end{cases}$$



Pattern recognition II – Results

- 3 TeV ttbar sample without overlay
- Cell following logic revised, keep more tracks candidate till the end
- Added fit in sz-fit to gain 3D-info
- Timing improved → 0.3 s/event (no Marlin track fit)
  - Linear regression implemented instead of chi2 for fit
  - Code profiled to find hotspot
- Good purity, numbers to be computed

Vertex only