HEPHY
INSTITUTE OF HIGH ENERGY PHYSICS

Doktoratskolleg
Particles and Interactions

Track seeding in the Outer Tracker of CMS for HL-LHC

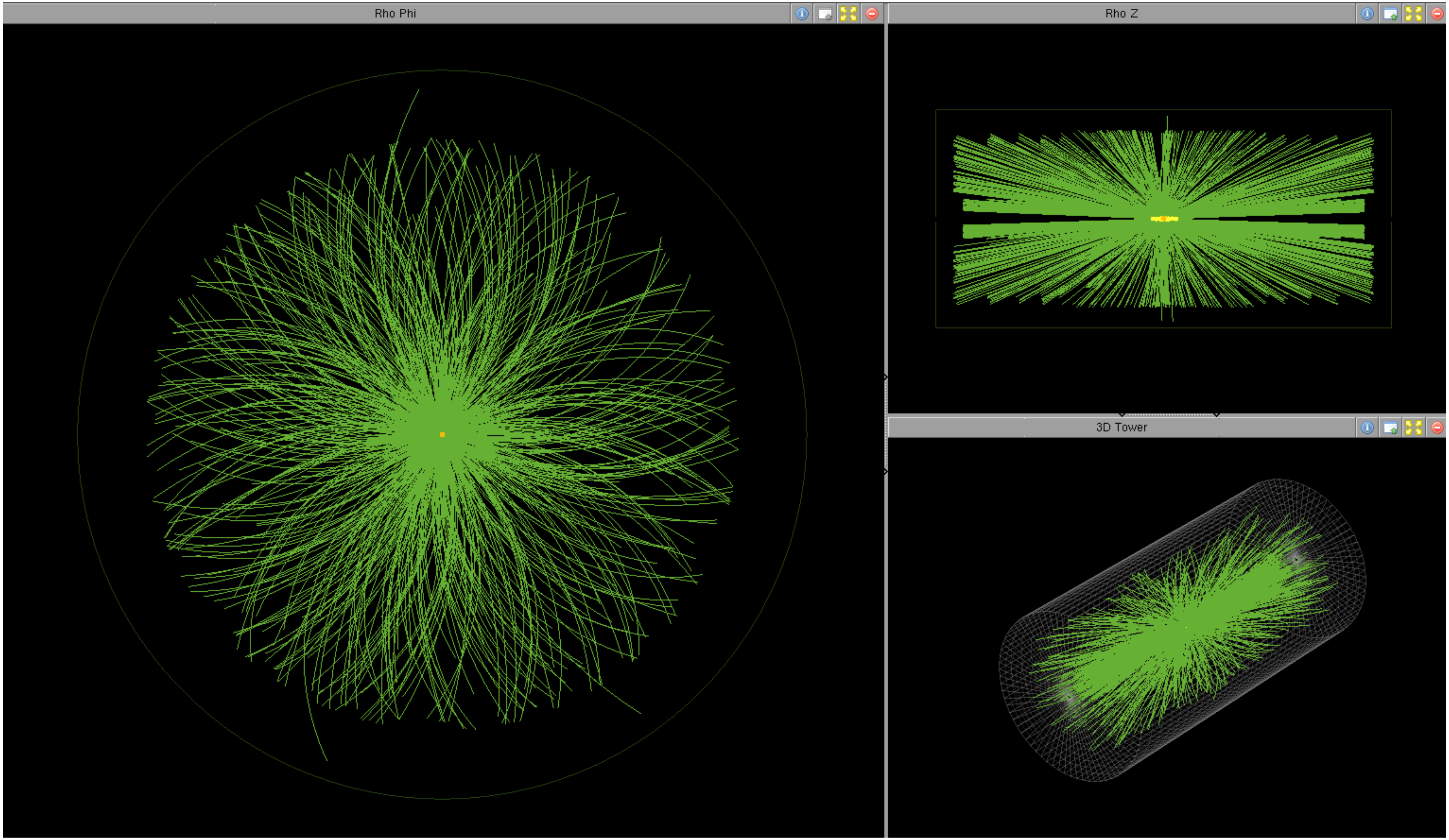
Erica Brondolin, Rudolf Frühwirth, Nikolaus Frohner
(HEPHY, Vienna)
On behalf of the CMS collaboration

18th International Workshop on Advanced Computing and Analysis Techniques in Physics Research
21st August 2017

Tracking Scenario in Phase 2

Reconstruction of CMS Simulated Event

$t\bar{t}$ event at <PU>=200 (126 vertices, 5934 tracks)



Phase-2 CMS Tracker

The Phase-2 detector is composed of:

- **Inner Tracker** : 4 barrel layer, 12 disks.

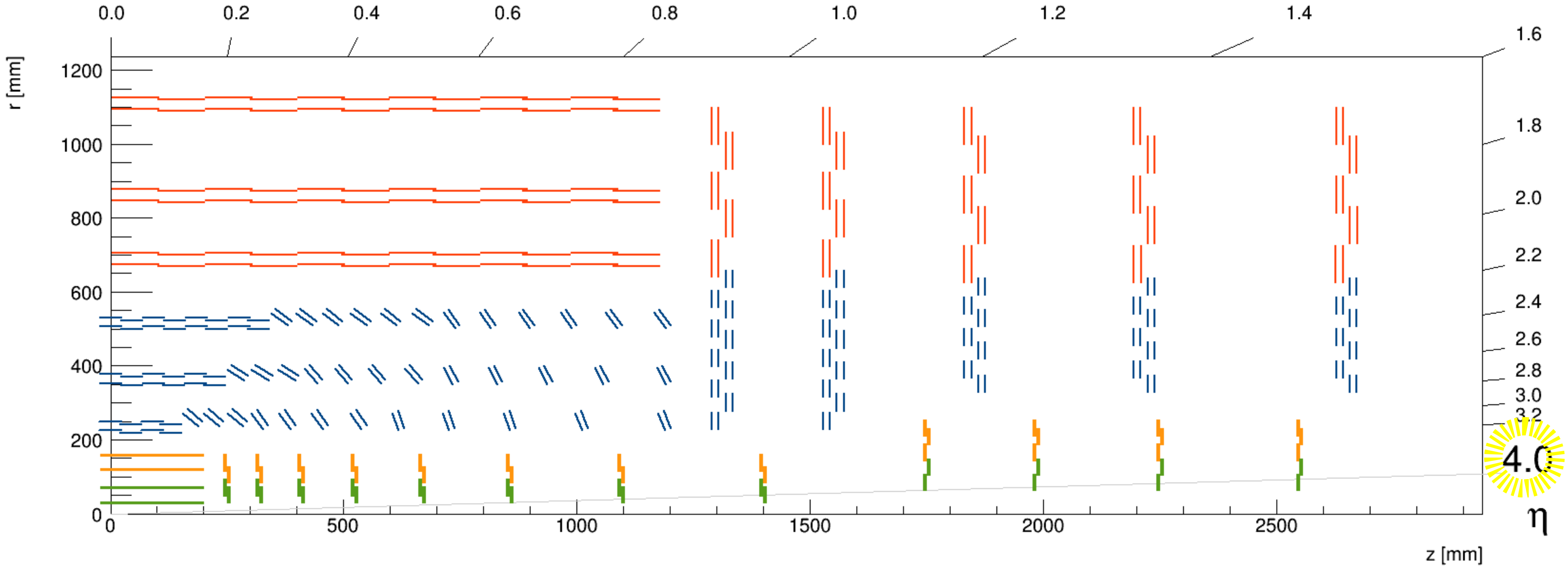
Extended up to eta = 4.0

- **Outer Tracker**: 6 barrel layer, 5 disks.

Each module consists of two closely spaced sensors:

 - **Pixel-strip (PS) modules**
 - **Strip-strip (2S) modules**

The modules are arranged in a tilted geometry for the Barrel Layers 1,2,3.

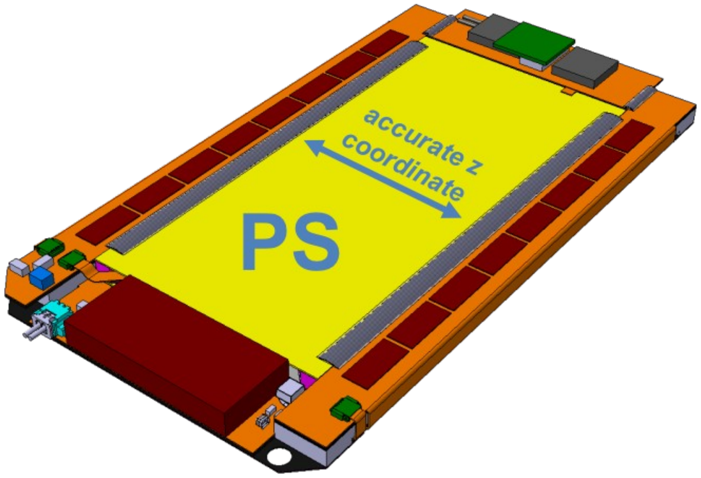


Geometry layout is the TDR one.

Modules in Phase-2 CMS tracker

Inner Tracker:

- 25 x 100 µm^2 or 50 x 50 µm^2 pixel size
- 6x pixel area reduction respect to current

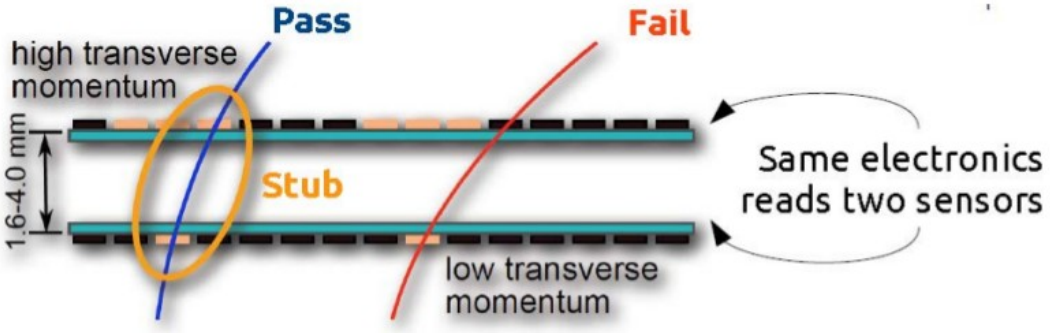


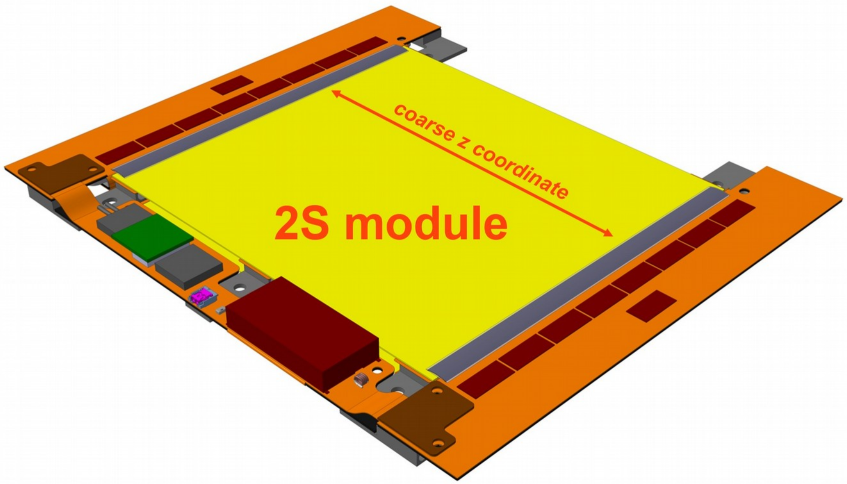
Outer Tracker:

Pixel + Strip Sensors (**PS**)
- Pixels: 1.5 mm x 100 µm
- Strips: 2.5 cm x 100 µm

2 Strips sensors (**2S**)
- Strips: 5 cm x 90 µm

Modules with two superposed sensors are able to filter tracks by momentum allowing for an L1 track trigger.

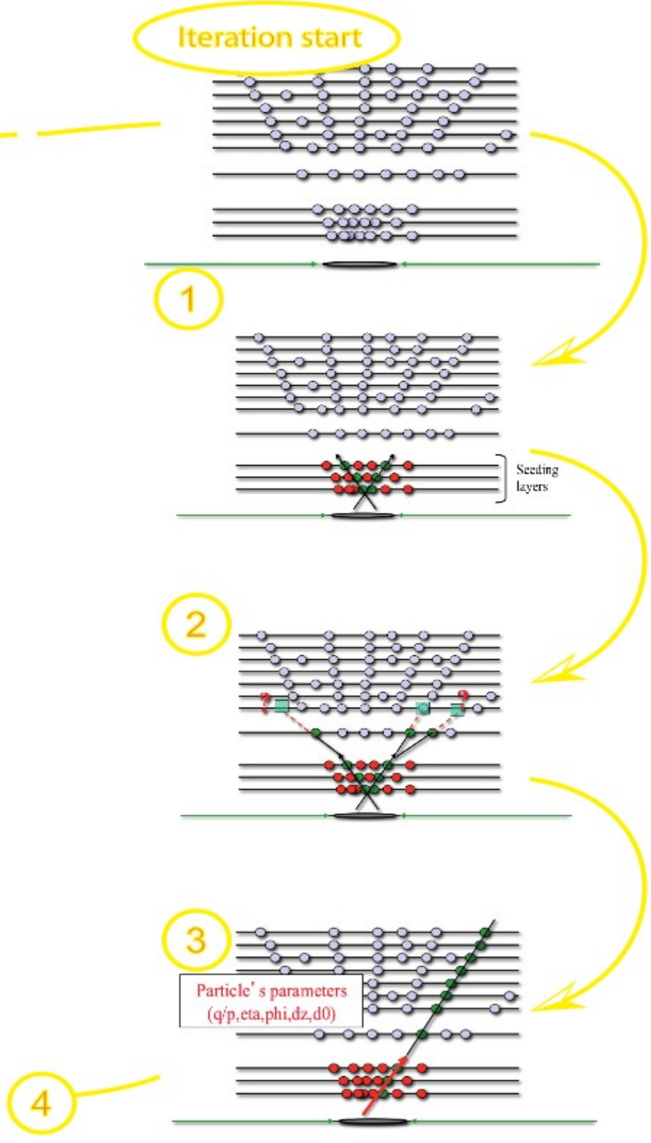




Track reconstruction

Iterative tracking philosophy

Tracks reconstructed in several iterations of the Combinatorial Track Finder
→ (search of easiest tracks + hits removing)



1. Seed generation

- Provide initial track candidates and trajectory parameters.

2. Track finding

- Extrapolate current trajectory parameters to the next layer and find compatible hits and update with Kalman filter.
- Continue until there are no more layers or there is more than 1 missing hit.

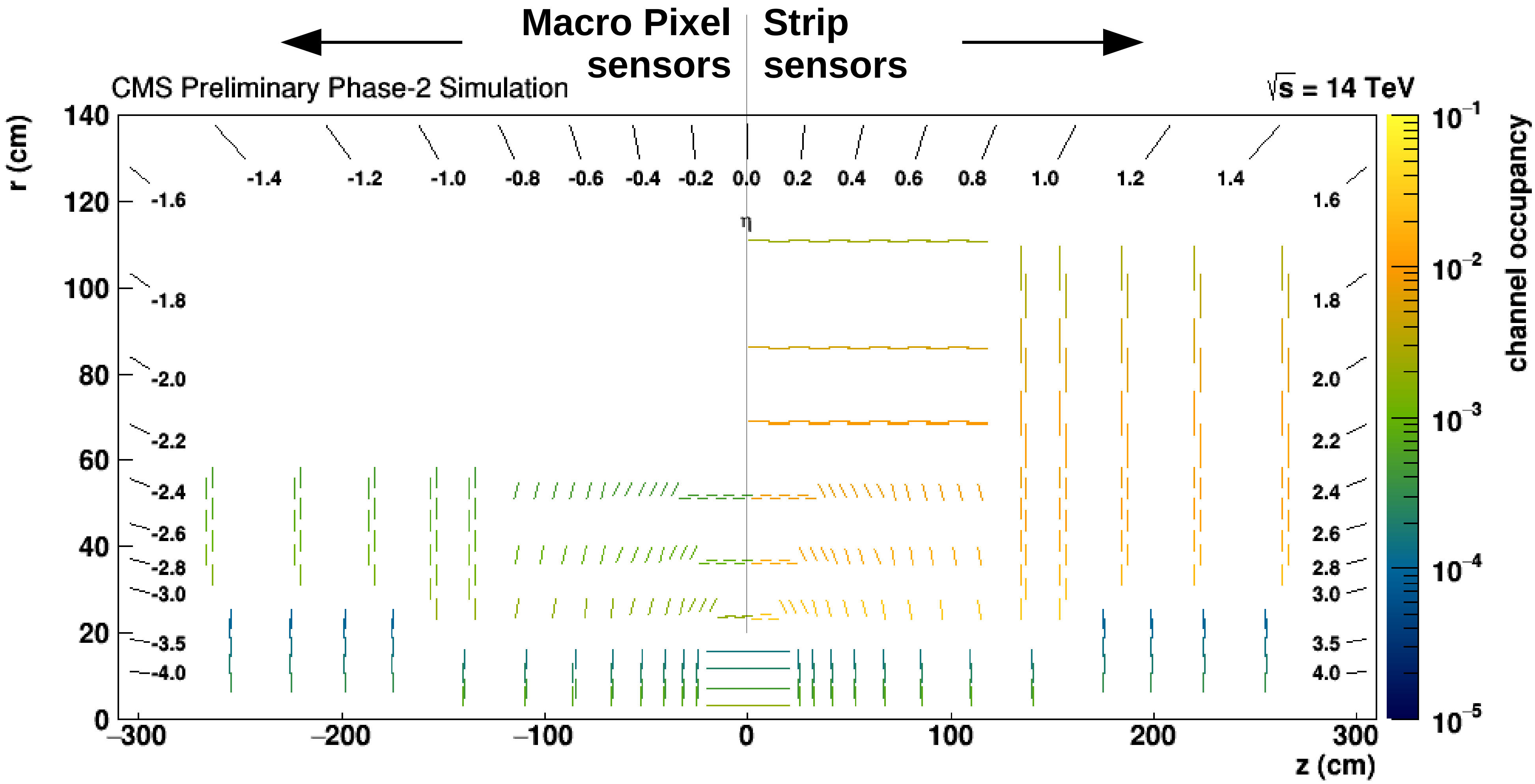
3. Track fitting

- Perform a final Kalman or Gaussian sum smoother to obtain the trajectory parameters at the interaction point.

4. Track selection

- Final selection and classification of tracks according to quality criteria

Seed reconstruction



Occupancy 10-100x lower in the Inner Tracker than in the Outer Tracker

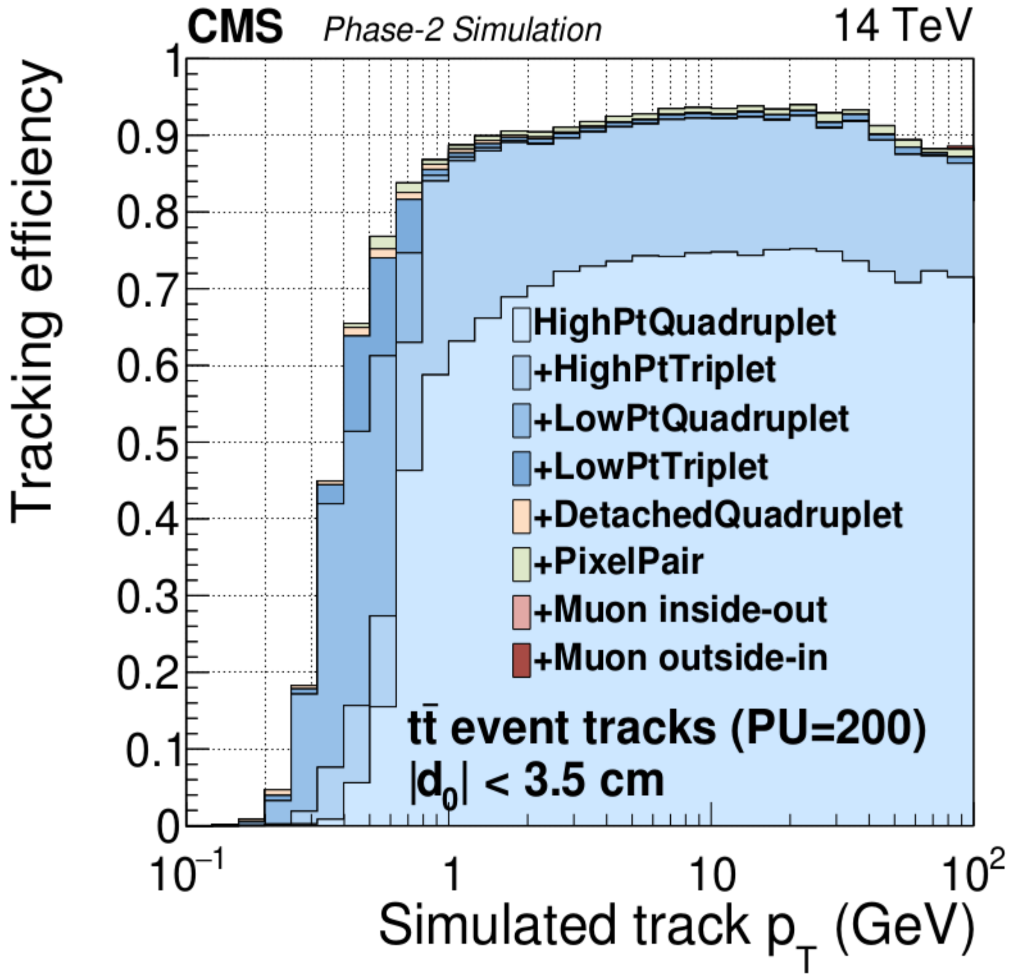
- Thanks to the high granularity of the pixel layers
- Same situation with the current detector
- Natural idea: start seeding in the Inner Tracker

Reconstruction Summary

Current Phase-2 iterative tracking

- CMS tracking iterative process → (search of easiest tracks + hits masking) x 8
- Tracking is the same for <PU> ~ 140/200 → some tuning required

Step Name	Seeding	Target Tracks
HighPtQuad	pixel quadruplets	prompt, high p_T
HighPtTriplet	pixel triplets	prompt, high p_T, recovery
LowPtQuad	pixel quadruplets	prompt, low p_T
LowPtTriplet	pixel triplets	prompt, low p_T, recovery
DetachedQuad	pixel quadruplets	displaced
PixelPair	pixel pairs	high p_T, recovery
Muon Inside-Out	muon-tagged tracks	muon
Muon Outside-In	muon-tagged tracks	muon



Tracking performance in Phase 2

Samples used

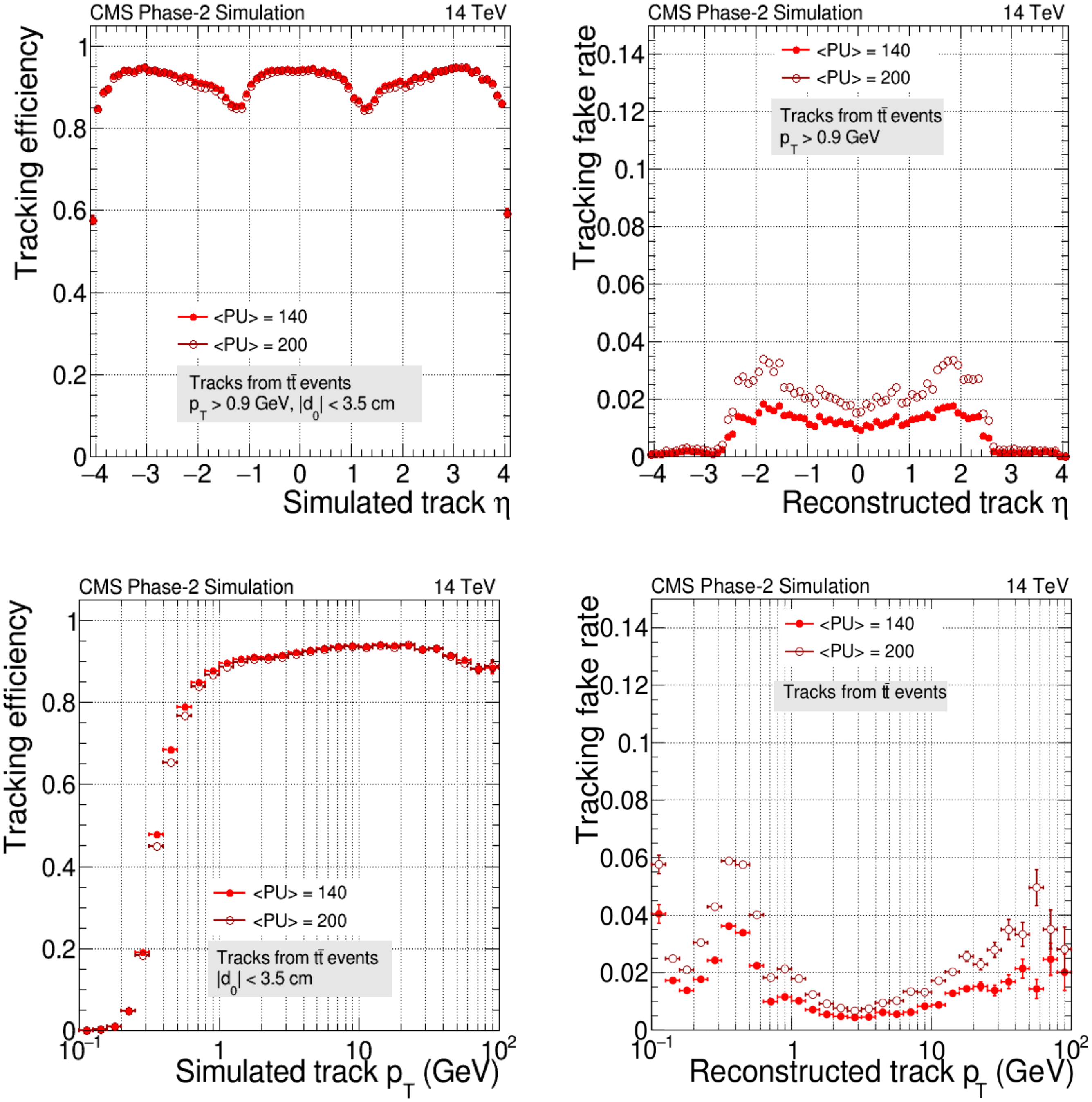
- ~9k $t\bar{t}$ events
- ~9k single muon events with p_T = 10 GeV
- Same sample used for Phase-1

- Pseudorapidity range [-4.0, +4.0]
- <PU> = 140, 200
- Highest quality tracks with the following selections:

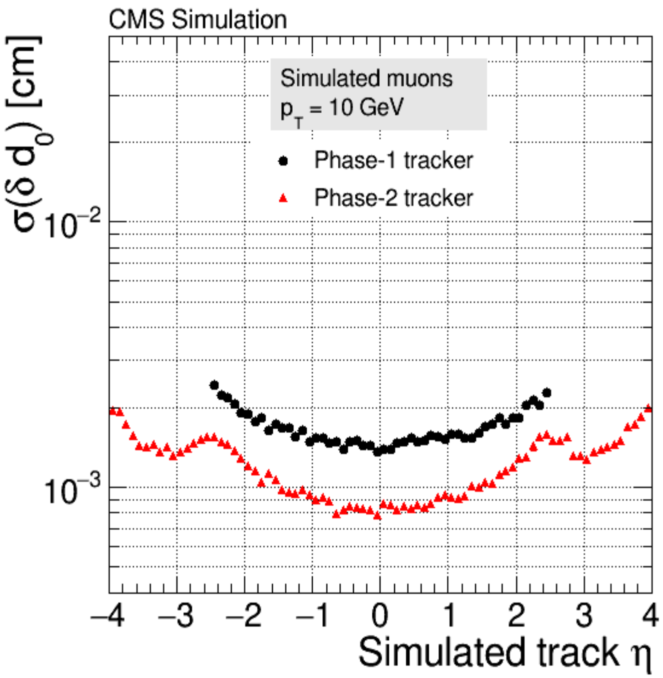
- For efficiency measurement: simulated track p_T > 0.9 GeV, d_0 < 3.5 cm
- For fake rate measurements: reconstructed track p_T > 0.9 GeV

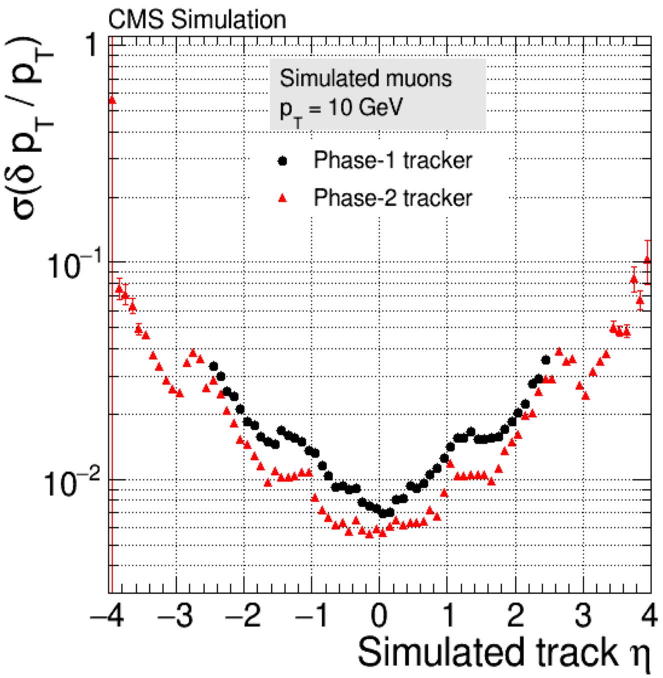
- Optimization for Phase-2 era on-going, results can be considered conservative

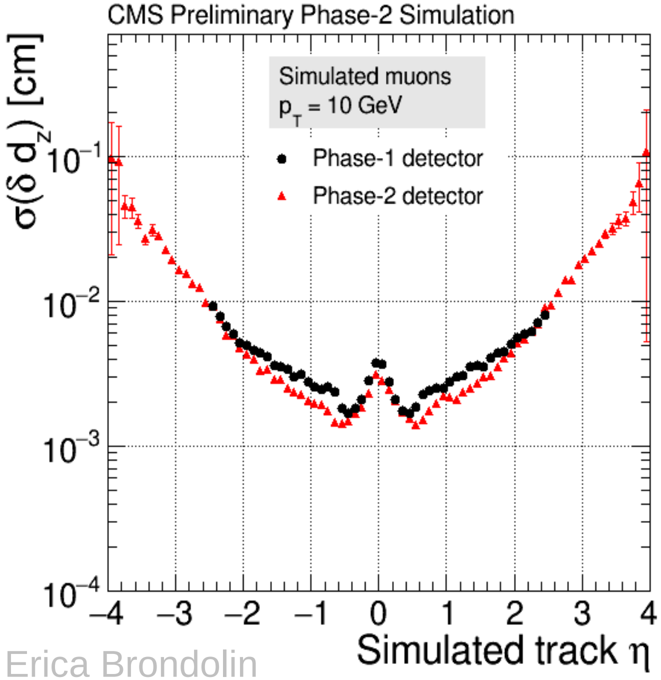
Tracking performance in Phase 2

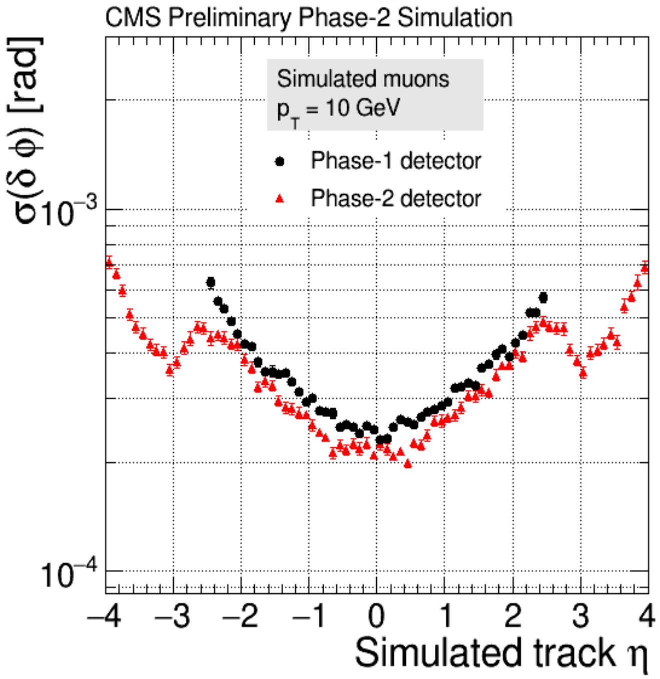


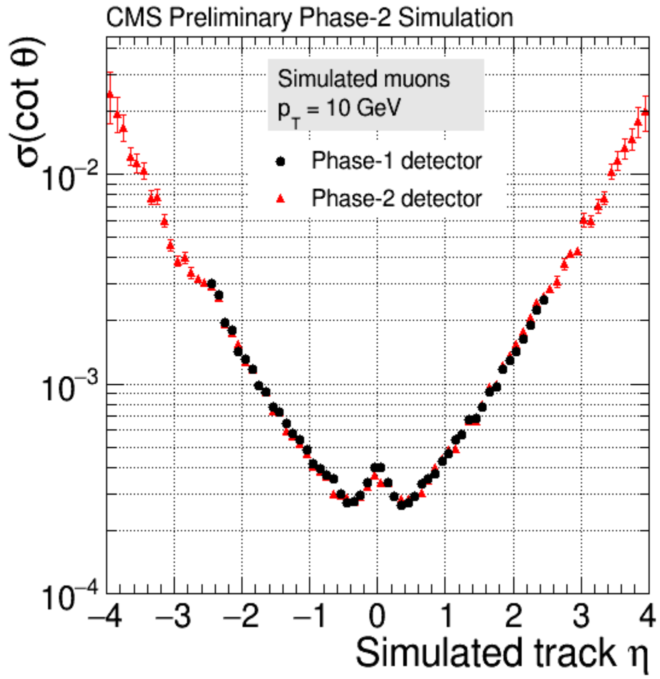
Tracking performance in Phase 2











Summary of baseline performance

Efficiency and fakerate:

- Good efficiency and low fakerate in the entire range of pseudorapidity
- From 1 GeV to 100GeV about 90% efficiency
- For <PU> = 140(200) the fake rate is lower than 4(6)% up to 100 GeV

- Efficiency almost independent of PU
- Further optimization of fakerate possible for high PU

Track parameters resolution:

- Better performance compared to the current detector
- Larger eta range

Vertex reconstruction:

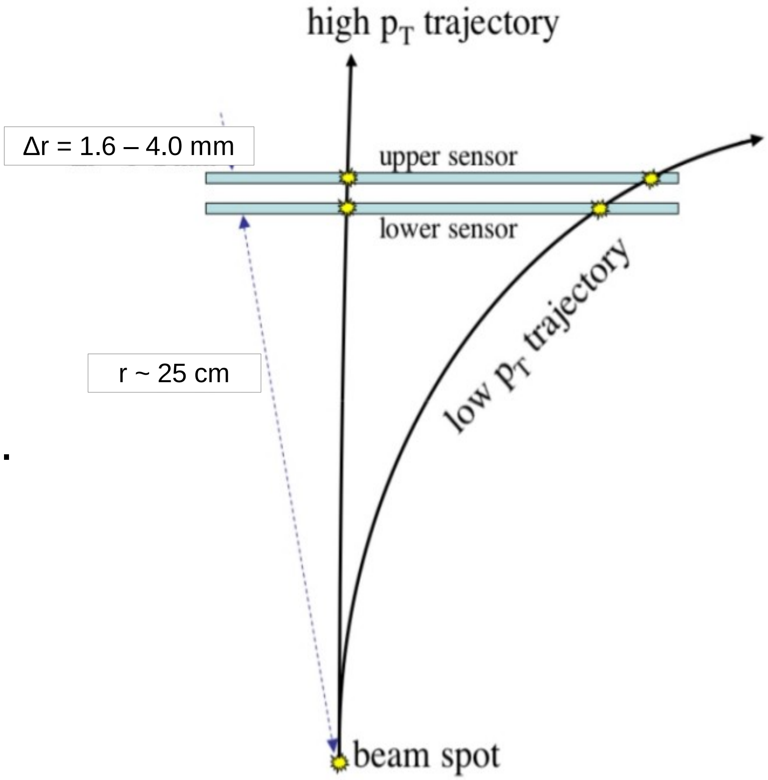
- Resolving power Δz < 1 mm
- Merging rate smaller than 1 even at 25 μm
- Vertex resolutions are almost independent of PU

More performance plots in backup

Vector Hits

New kind of hits in the outer tracker [1]

Vector hits (~offline stubs) are short track segments reconstructed from two clusters in stacked sensors – upper and lower clusters.
A vector hit contains direction information



Local reconstruction status:

Vector Hits have been implemented in CMS software framework
Implemented window cut:

- width = $x_{lower} – x_{upper}$ where x is local precise coordinates.
- comparison between two approaches:
  1) Fixed cut on the width
  2) Layer-dependent cut on the width

Global reconstruction status:

Comparison with the baseline using Kalman Filter
Introducing Outer Tracker seeded iteration to reconstruct displaced tracks

Everything run and tested in latest release
Work in progress – Test at high PU is needed also for timing and memory consumption

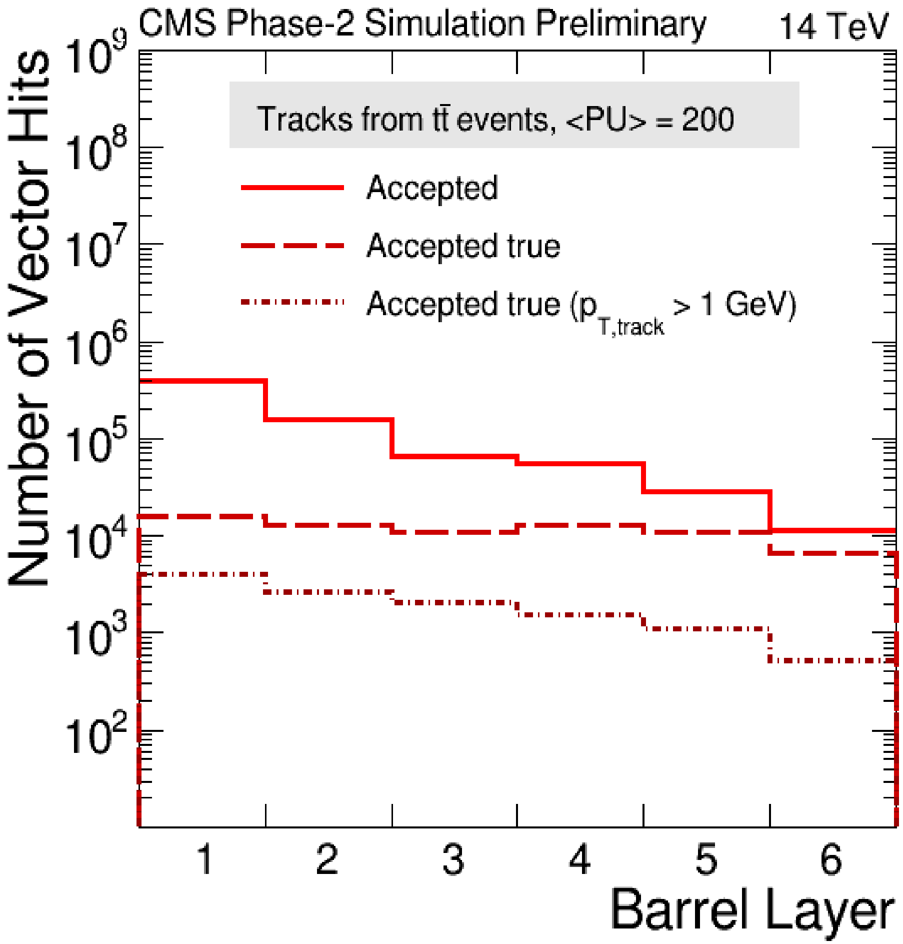
[1] Pattern recognition with vector hits, R. Frühwirth

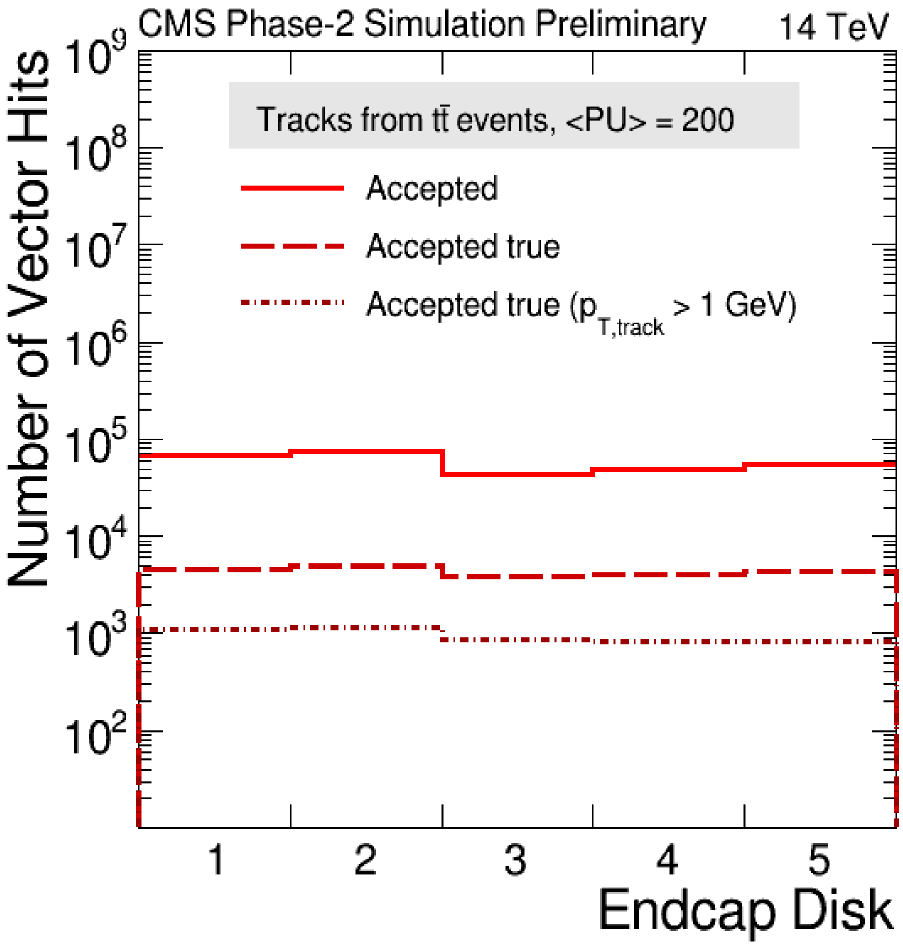
Challenges at high PU

True Vector Hits:

Both upper and lower clusters are associated to the same simulated track.

If no window cut is applied → all Vector Hits created in the event are labeled as “accepted”.



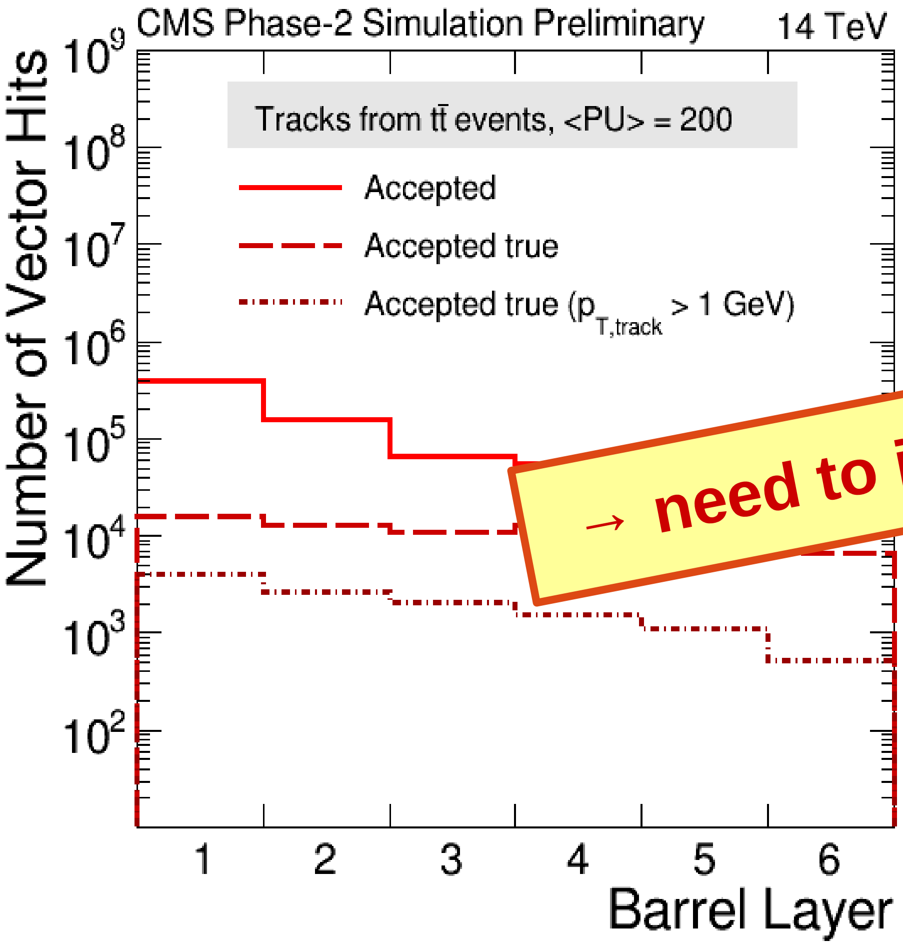


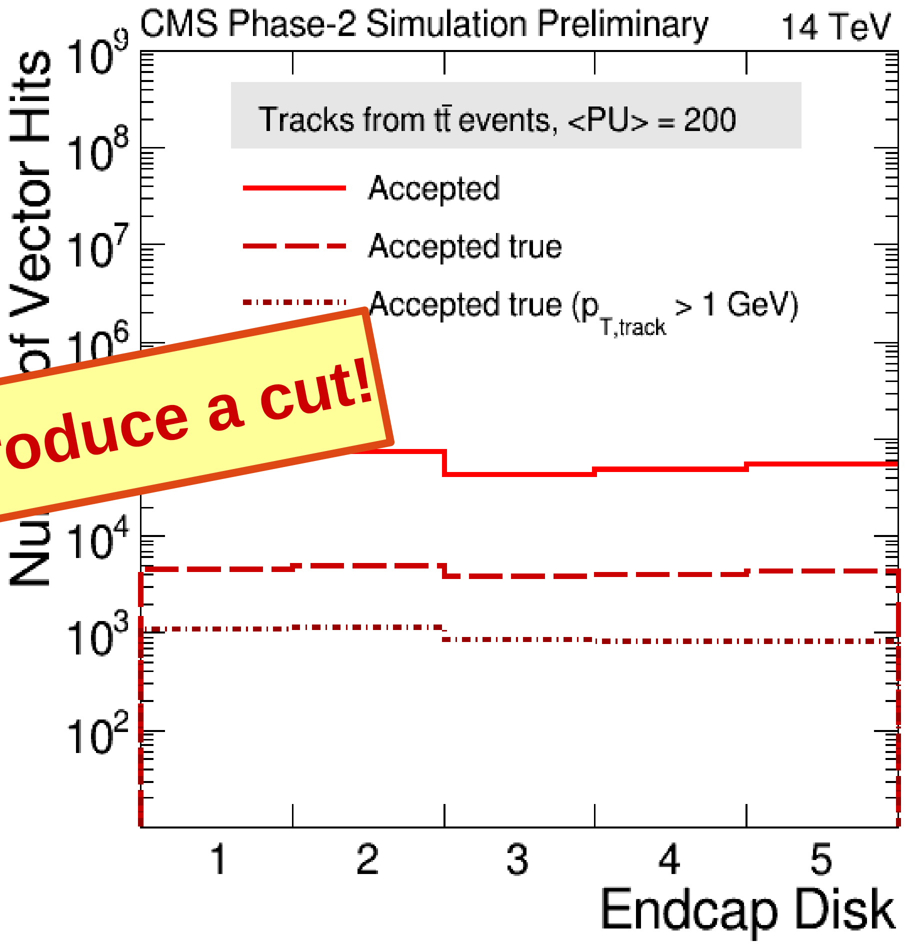
Challenges at high PU

True Vector Hits:

Both upper and lower clusters are associated to the same simulated track.

If no window cut is applied → all Vector Hits created in the event are labeled as “accepted”.





→ need to introduce a cut!

Fixed width cut

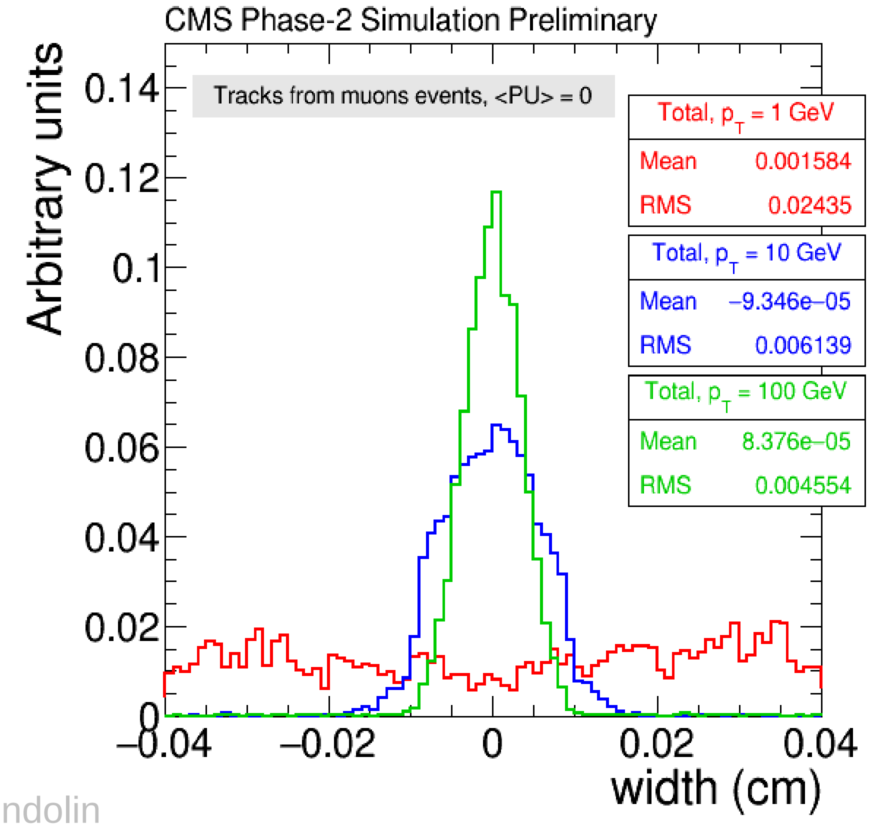
- Correcting cluster position with parallax correction
- width = $x_{lower} - x_{upper}$

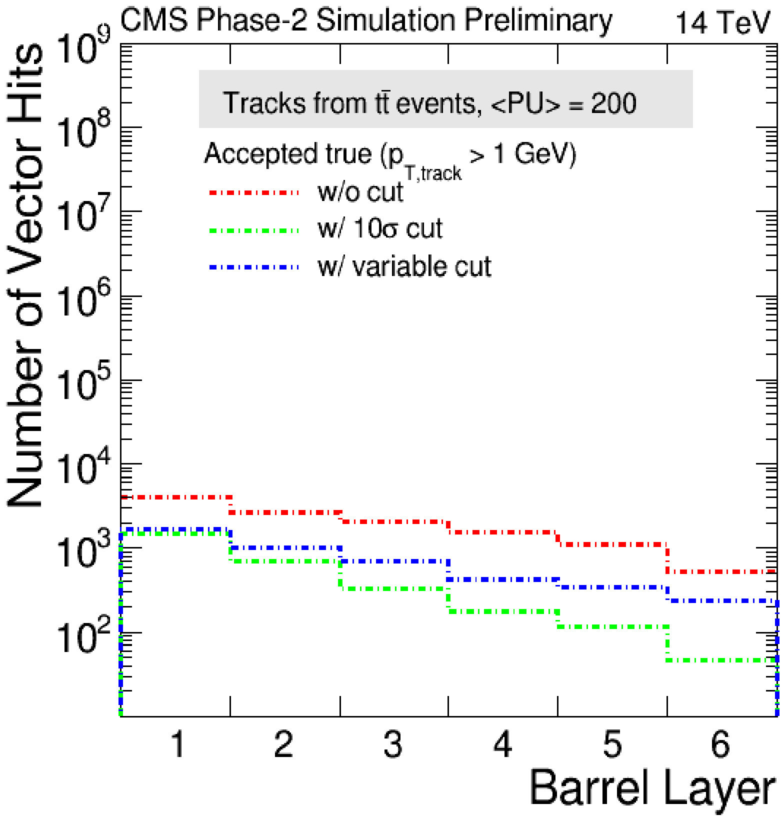
Fixed cut:

- $|width| < 10\sigma$, where $\sigma = \mathrm{sqrt}(\sigma_{x,lower}^2 + \sigma_{x,upper}^2)$

Limitations:

- Not good for low pT tracks
- Losing efficiency for outermost layers in the barrel



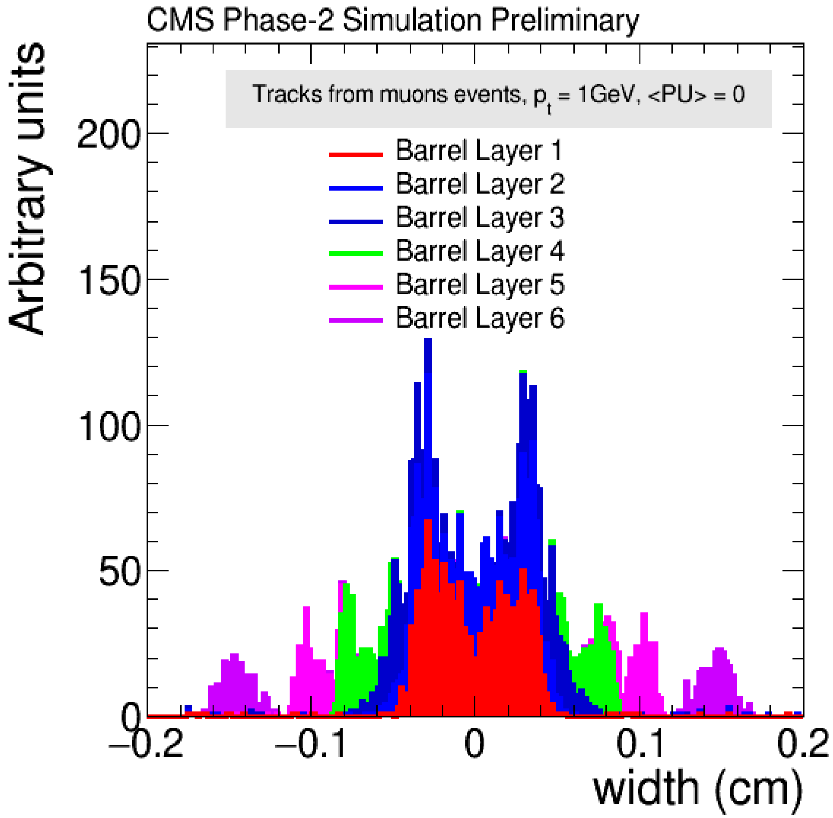


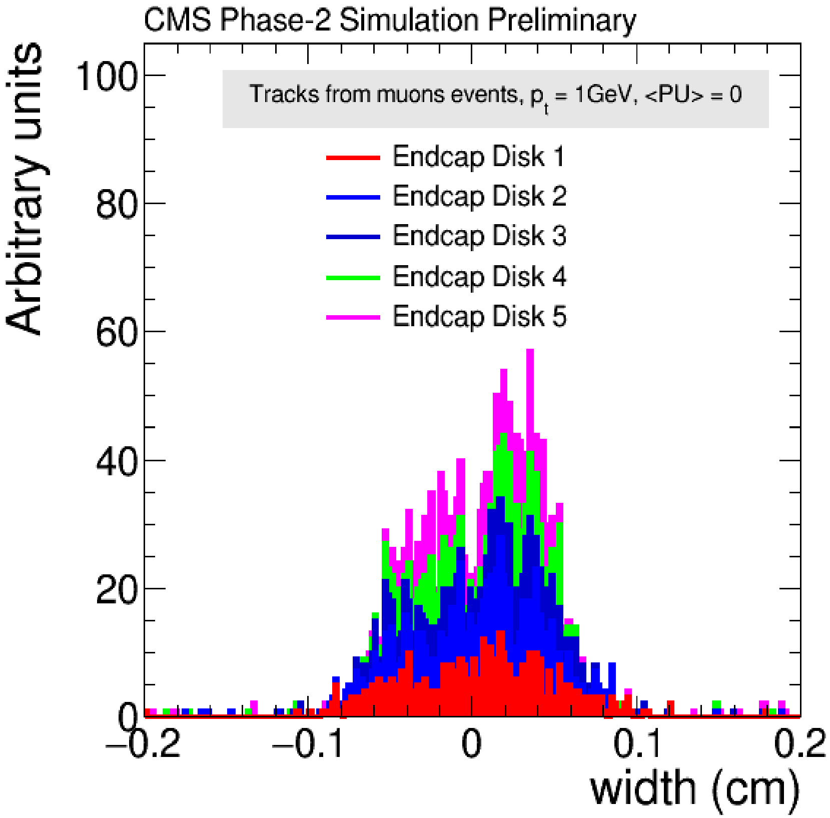
Variable width cut

- Correcting cluster position with parallax correction
- width = $x_{lower} – x_{upper}$

Layer-dependent cut:

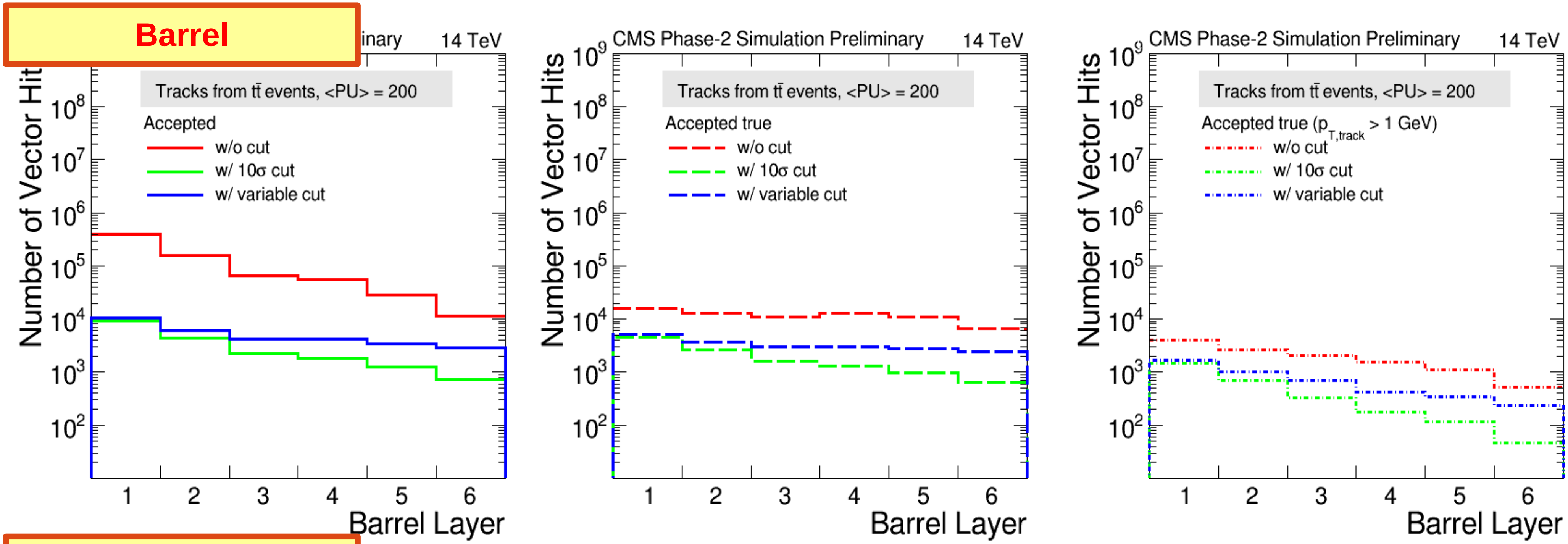
- |width| < value for a specific Layer/Disc
- Values chosen using single muon with p_T= 1 GeV



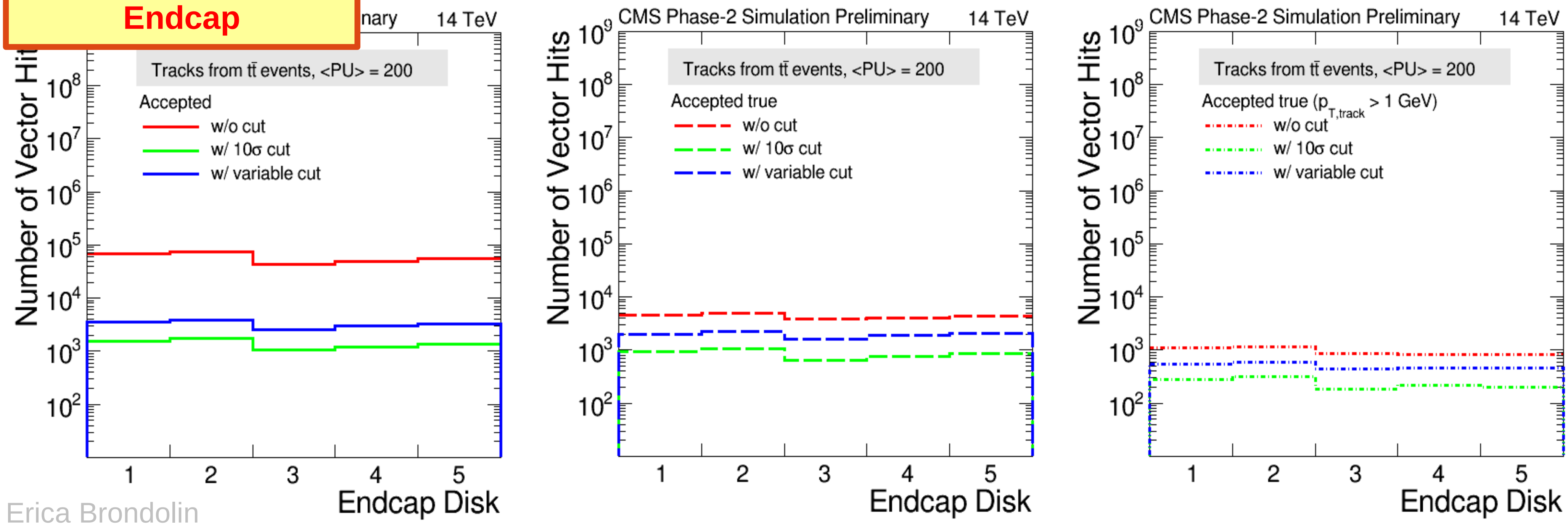


Challenges at high PU

Barrel



Endcap



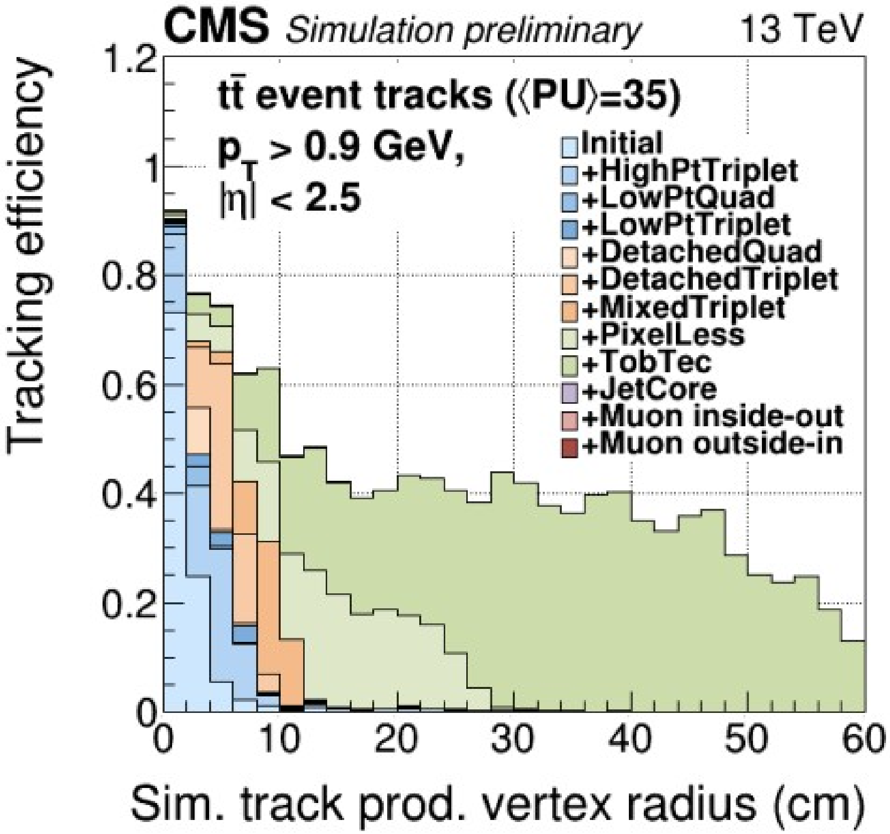
Seeding in the Outer Tracker

- Comparison with the baseline using Kalman Filter:
  all iterations using the Vector Hits in the Outer Tracker

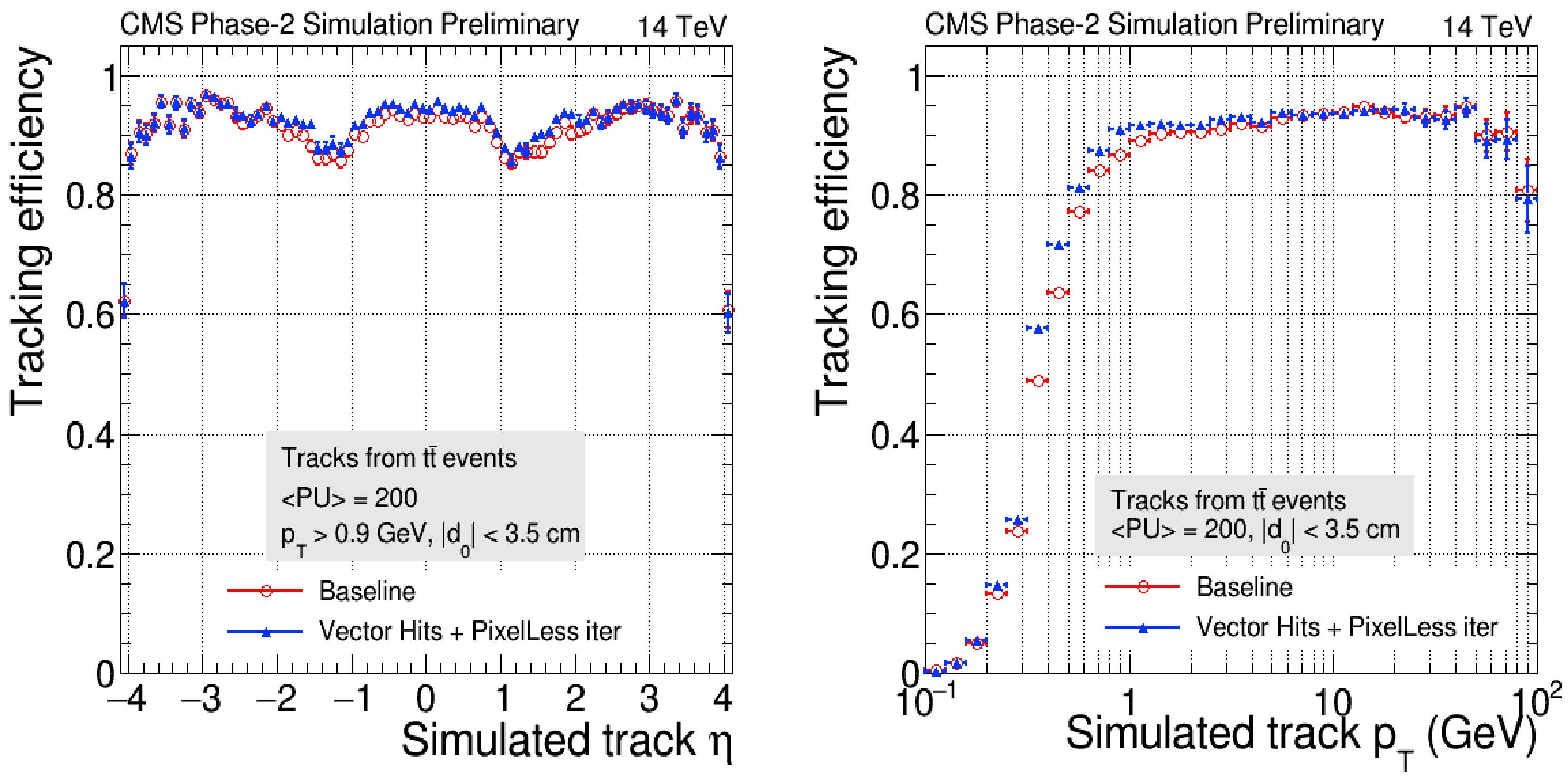
- Introducing Outer Tracker seeded iteration to focus on displaced tracks:
  - seeds are pairs of VHs in adjacent layers
  - selection of pairs using geometrical cuts
  - referred to as PixelLess
  - promising results with high PU samples

- Exploiting Vector Hits possibilities with ML techniques using Convolutional Neural Network (CNN)
  - seeds are triplets of VHs in the inner three barrel layers
  - Neural Network Classifier
  - training/testing with CMSSW Full Simulation samples

2017 tracking

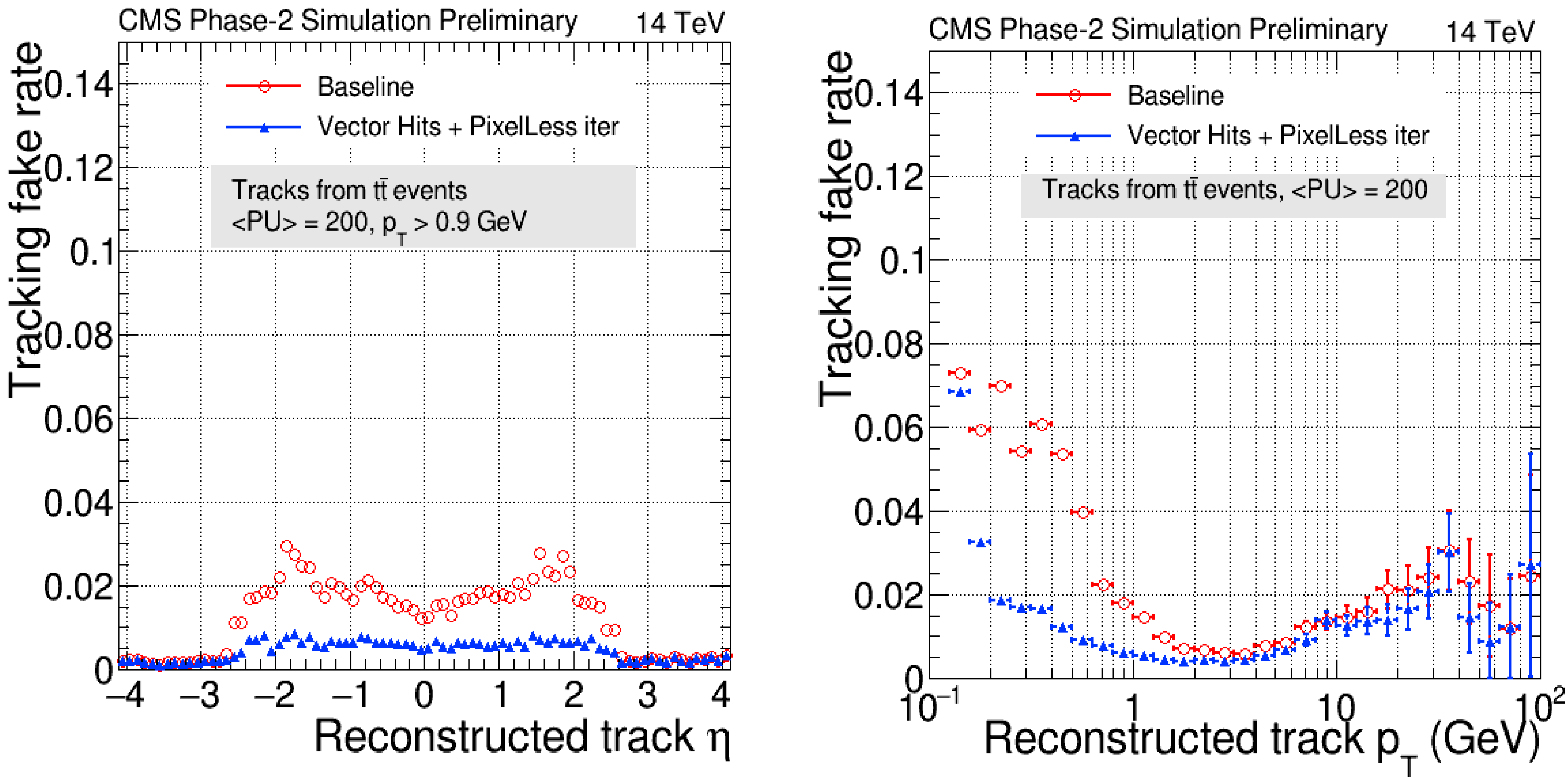


Tracking performance with VHs



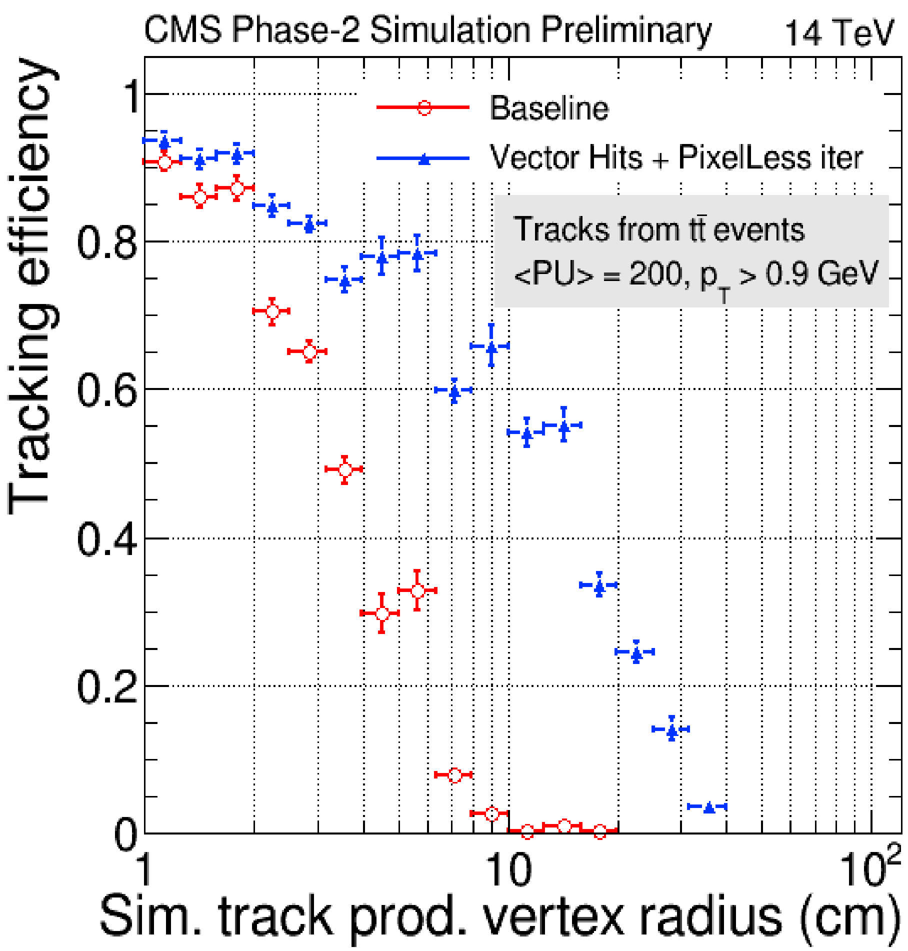
- The baseline (red) is compared with track reconstruction using Vector Hits and including the Outer Tracker seeded iteration (blue)
- Efficiency compatible with baseline for high PU scenario

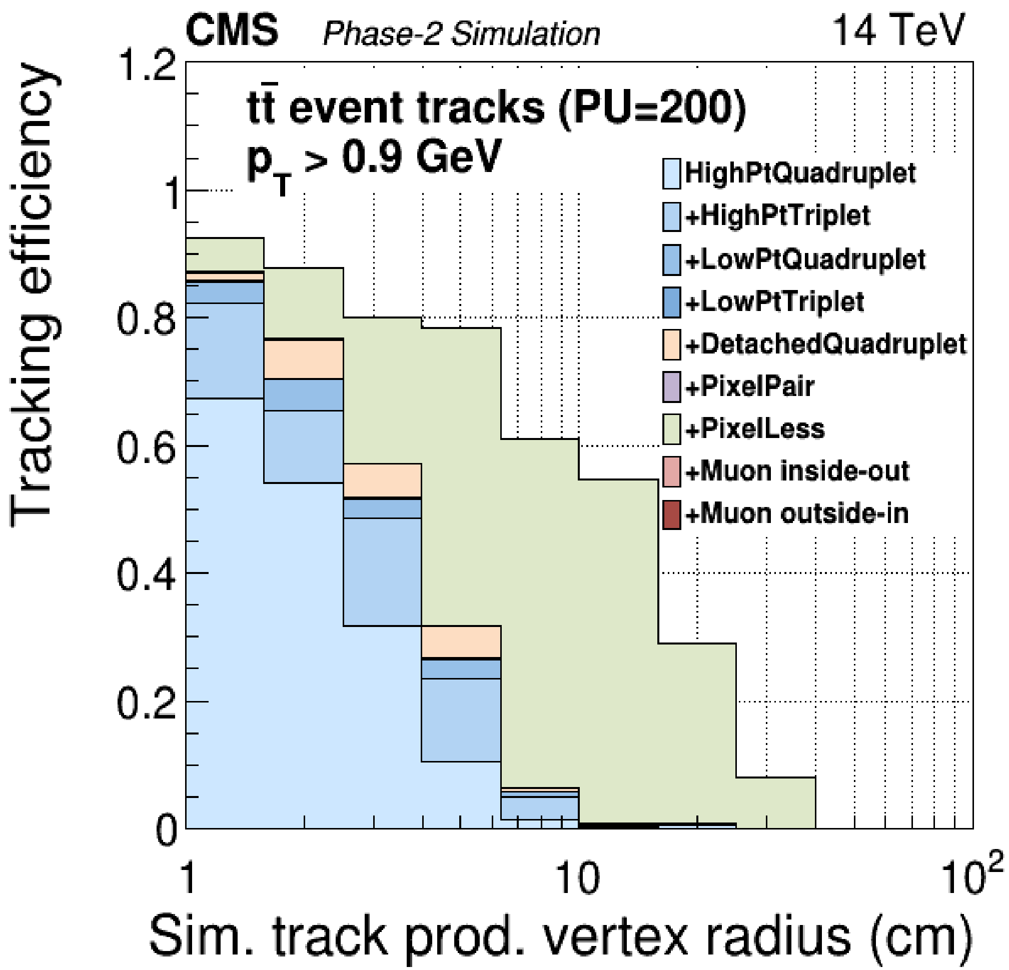
Tracking performance with VHs



- The baseline (red) is compared with track reconstruction using Vector Hits and including the Outer Tracker seeded iteration (blue)
- Fake rate reduced of almost a factor 10

Tracking performance with VHs



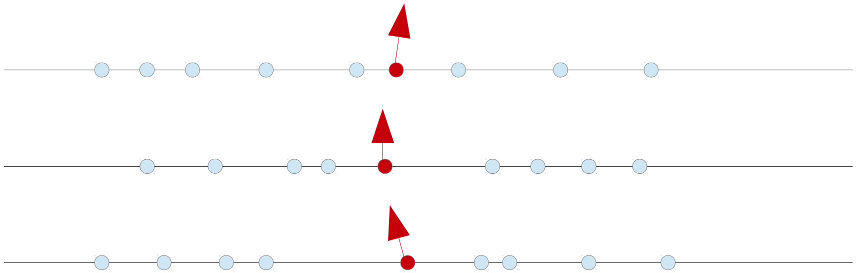


- The baseline (red) is compared with track reconstruction using Vector Hits and including the Outer Tracker seeded iteration (blue)
- With Outer Seeded iteration the efficiency for large vertex radius is improved
- Dramatic increase in efficiency for long-living neutrals

Seeding with VHs triples using ML

- Very preliminary results

- ~100 $t\bar{t}$ + <PU> = 200 events
- Using first three layers of outer tracker
- Events split into training/validation/test: 60%/20%/20%
- True triples:
  all Vector Hits belonging to the same SimTrack (pT≥ 1GeV)
- False triples:
  generate combinatorial background using
  9 nearest neighbors in each layer



- Training input data:
  balanced set with 1:1 ratio false:true
- Test input data:
  unbalanced set with 27:1 ratio false:true

- Network Input:
  - convert cluster coordinates to cylindrical coordinates (18 inputs in total)
  - order clusters by r
  - subtract polar angle of innermost cluster (symmetry)

$r_{1,i}$	$\Phi_{1,i} - \Phi_{1,i}$	$z_{1,i}$	$r_{1,o}$	$\Phi_{1,o} - \Phi_{1,i}$	$z_{1,o}$
$r_{2,i}$	$\Phi_{2,i} - \Phi_{1,i}$	$z_{2,i}$	$r_{2,o}$	$\Phi_{2,o} - \Phi_{1,i}$	$z_{2,o}$
$r_{3,i}$	$\Phi_{3,i} - \Phi_{3,i}$	$z_{3,i}$	$r_{3,o}$	$\Phi_{3,o} - \Phi_{1,i}$	$z_{3,o}$

- Detailed architecture description in backup

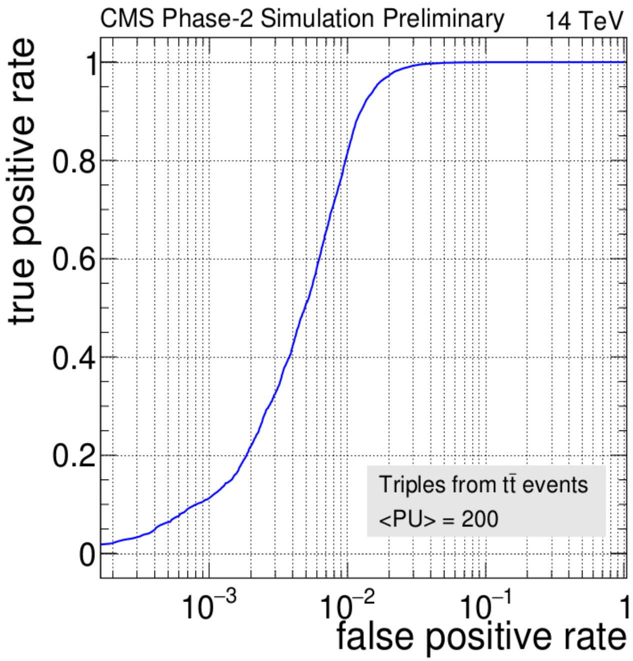
Seeding with VHs triples using ML

- Very preliminary results

- true positive rate: number of accepted true triples with SimTrack (pT≥ 1GeV) / total number of true triples with SimTrack (pT≥ 1GeV) (13365)

- false positive rate: number of accepted false triples / total number of false triples in the test set (360855)

- Large area under the ROC curve

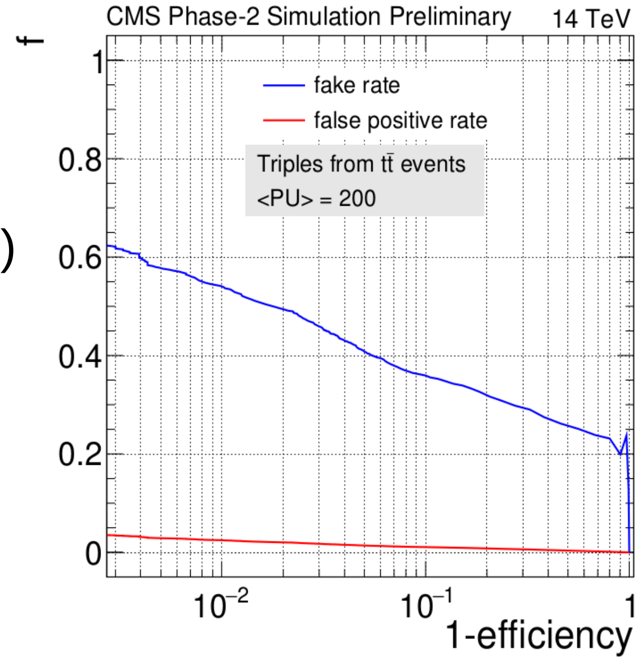


- efficiency: number of SimTracks (pT≥ 1GeV) corresponding to at least one accepted true triple / total number of SimTracks (pT≥ 1GeV) (7652)

- fake rate: number of accepted false triples / (number of accepted false triples + number of SimTracks (pT≥ 1GeV) corresponding to at least one accepted true triple)

- Large reduction of false triples

- <u>Comparison with baseline in progress</u>



Conclusions & Outlook

TDR results public since few months

- Excellent tracking and vertexing performance with high PU
- Phase-2 tracking specific development started!

Exploiting new Outer Tracker possibilities

- Local reconstruction of Vector Hits already in place
- On-going further optimization

Outer Tracker seeding developing on several fronts

- Seeding with Vector Hits pairs already working with excellent performance
- Application of ML to seeding with Vector Hits triples in initial phase

To be continued in the next future… stay tuned!

Thank you for your attention

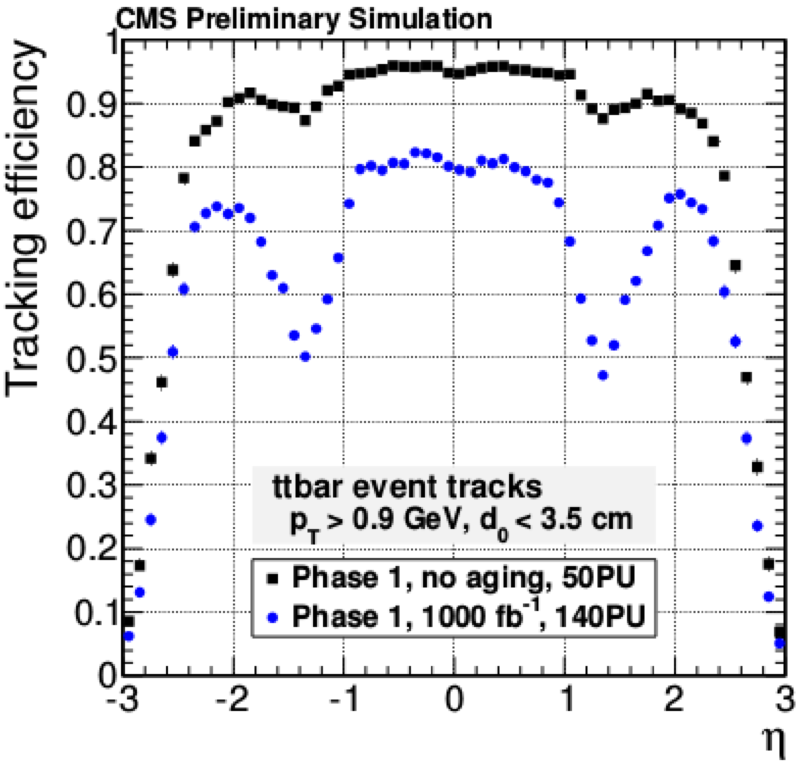
Limitation of the Current Tracker

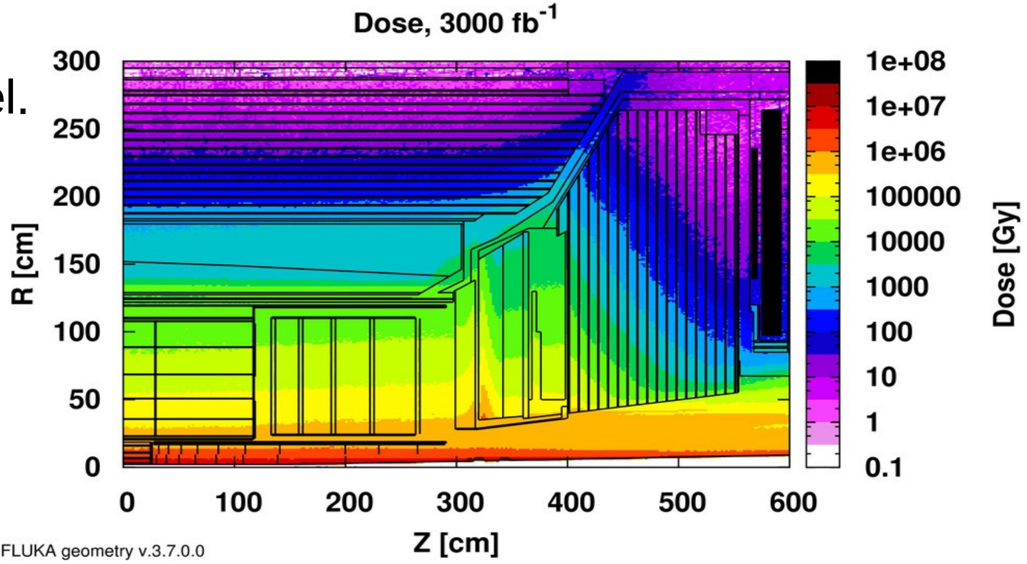
Radiation damage and performance degradation

- Present tracker designed for a integrated lumi of 500/fb and <PU> ~ 30-50.
- Radiation damage in the pixels reduces charge collection and Lorentz angle while in the strip tracker it increases depletion voltage and leakage current.
- Pixel dynamic inefficiency becomes not negligible.

Requirements

- High radiation tolerance to operate efficiently up to 3000/fb.
- Increased granularity to maintain channel occupancy around or below the per cent level.
- Reduced material in the tracking volume.
- Contribution to the Level-1 trigger.
- Extended tracking acceptance.
- Robust pattern recognition.





Vertex reconstruction & performance

Vertex Reconstruction

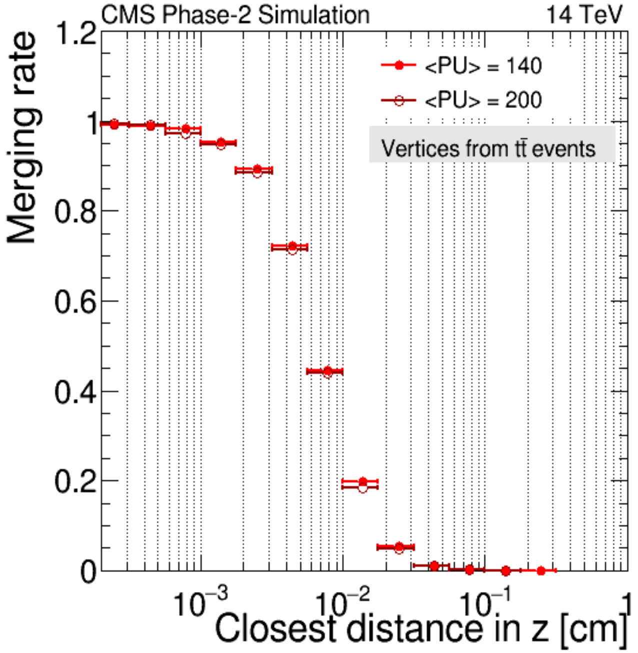
- Run-I based vertexing algorithm
- Track clustering with Deterministic Annealing
- Adaptive Vertex fit to estimate vertex position

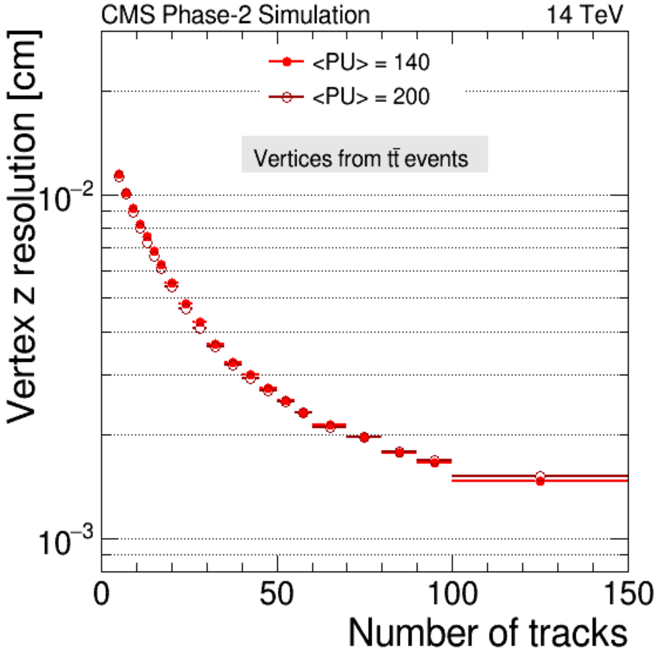
Primary vertex tagging

- Reconstructed vertices sorted by ascending Σp_T^2, using jets originating from the same vertex, remaining isolated single tracks and missing transverse momentum.
- The vertex with the highest Σp_T^2 is tagged as Primary Vertex.

Efficiency of primary vertex reconstruction and tag

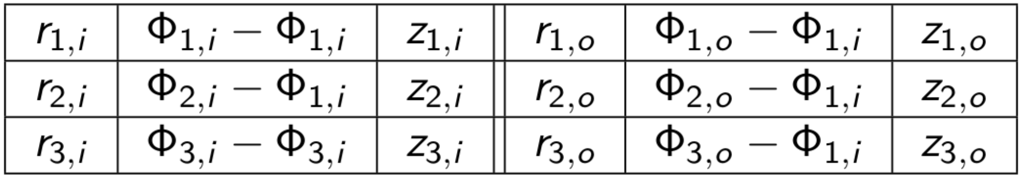
- ~94% for simulated tt events + 140PU
- ~89% for simulated tt events + 200PU





Seeding with VHs triples using ML

- Input dimension for network 6@1x3
  (one triple with three vector hits, each vector hit has 6 features)

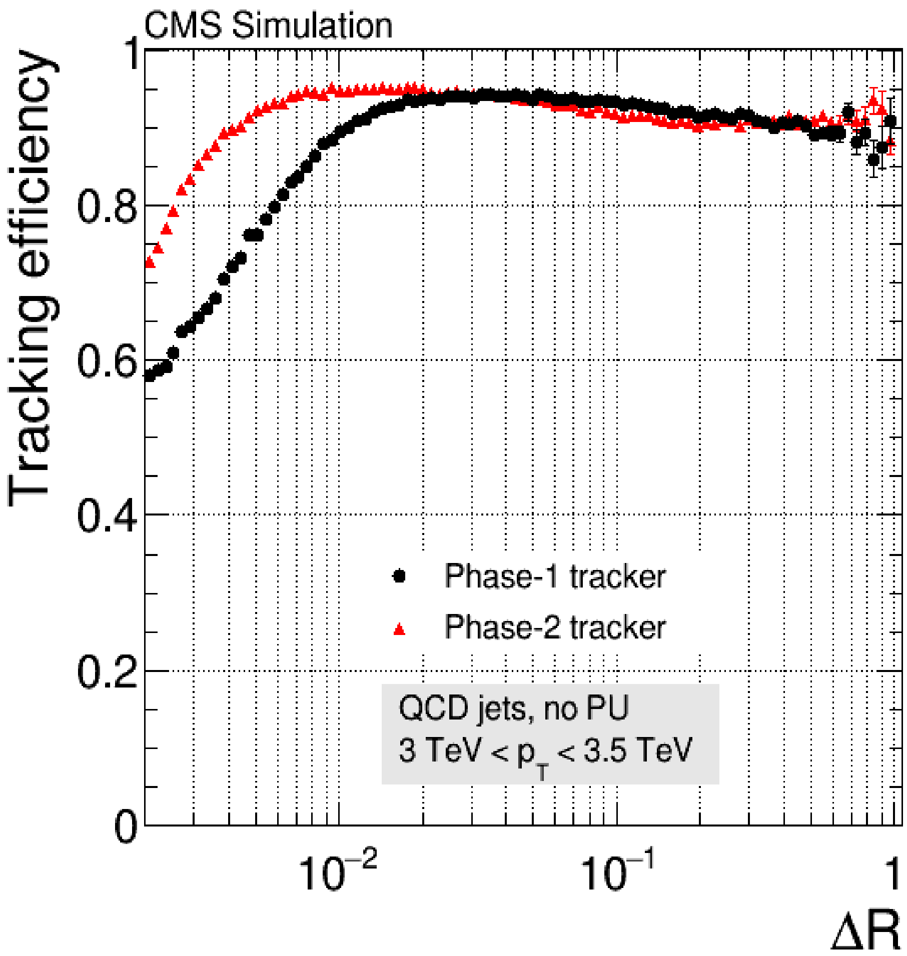


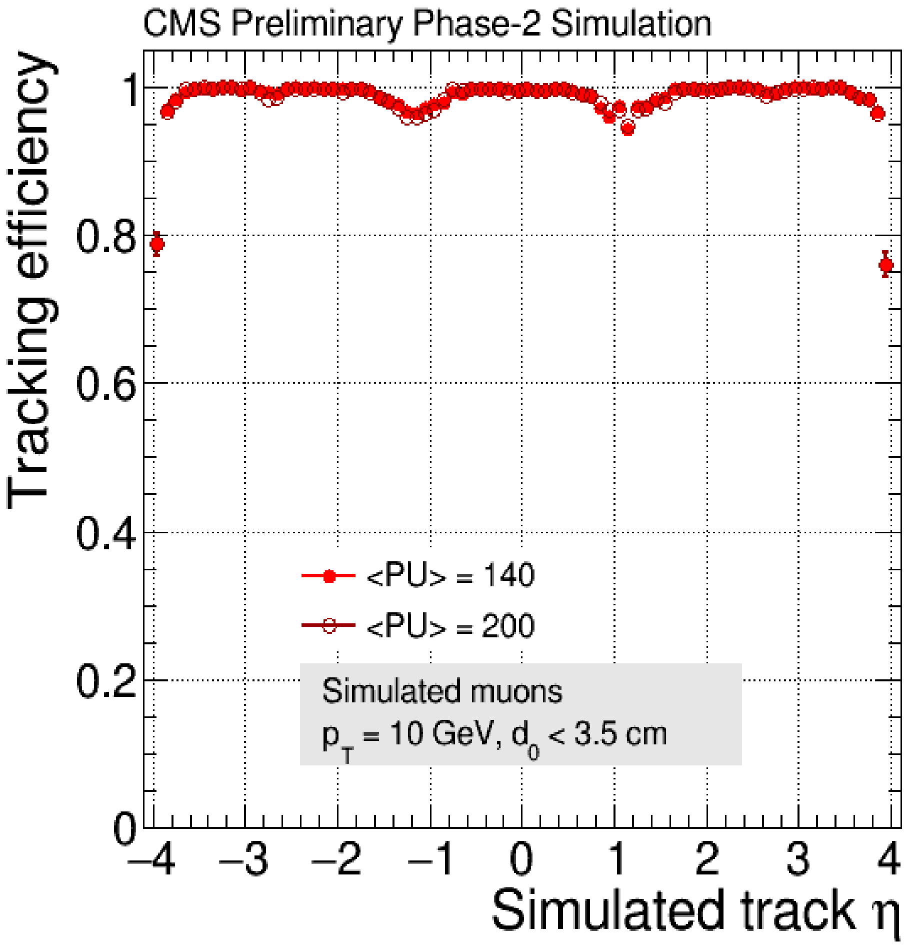
$r_{1,i}$	$\Phi_{1,i} - \Phi_{1,i}$	$z_{1,i}$	$r_{1,o}$	$\Phi_{1,o} - \Phi_{1,i}$	$z_{1,o}$
$r_{2,i}$	$\Phi_{2,i} - \Phi_{1,i}$	$z_{2,i}$	$r_{2,o}$	$\Phi_{2,o} - \Phi_{1,i}$	$z_{2,o}$
$r_{3,i}$	$\Phi_{3,i} - \Phi_{3,i}$	$z_{3,i}$	$r_{3,o}$	$\Phi_{3,o} - \Phi_{1,i}$	$z_{3,o}$

- Architecture:
  - 2 stacked 1D convolution layers 32@1x3 with kernel size 2, stride 1 and padding to keep dimensionality
  - 1D max pooling of size 2 and stride 1 reducing dimensionality to 32@1x2
  - 2 stacked 1D convolution layers 64@1x2 with kernel size 2, stride 1 and padding to keep dimensionality
  - 1D max pooling of size 2 and stride 1 reducing dimensionality to 64@1x1
  - flattening and feeding fully connected backend with 1 layer of 50 units and dropout of 0.5
  - softmax activation in output layer, everywhere else ReLU activation
  - trained for binary cross entropy loss, output corresponds to probability of a triple being true

- Tools:
  - Keras
  - TensorFlow
  - Python

Other objects performance





Jet core reconstruction:

- No tuning applied yet
- Already significant improvement can be seen for small values of ΔR thanks to the higher granularity of the new detector

Muons reconstruction:

- Efficiency almost 100%