Conformal Tracking for All-Silicon Trackers at future Electron-Positron Colliders

Frank Simon
Max-Planck-Institute for Physics

on behalf of the CLICdp Collaboration

presenting work by Erica Brondolin, Emilia Leogrande et al.

LCWS
Sendai, October 2019

Max-Planck-Institut für Physik
(Werner-Heisenberg-Institut)

Outline

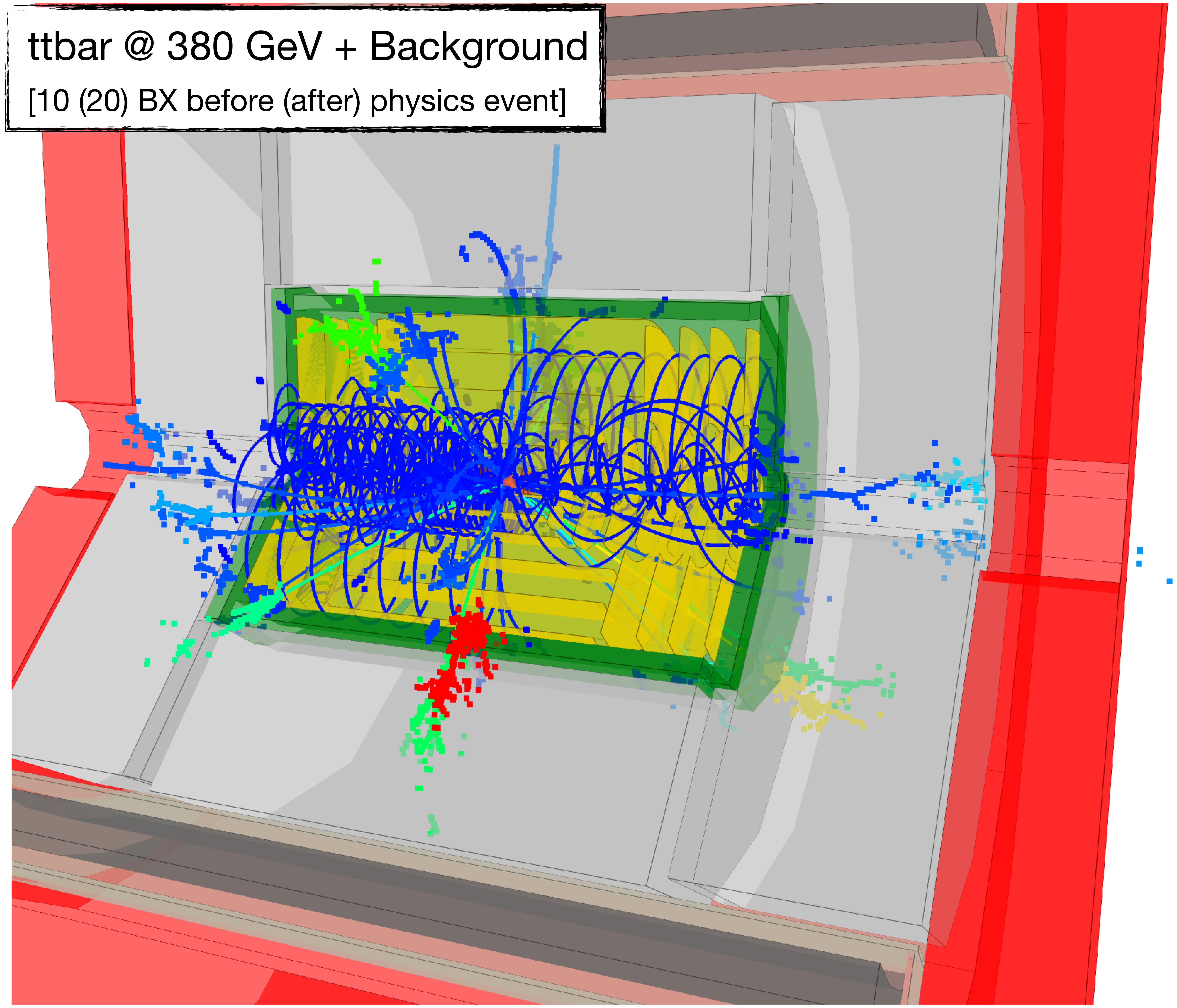
- Tracking Challenges at CLIC
- CLIC Detector Concept - with a focus on the Tracker
- Conformal tracking at CLIC
- Tracking Performance

Tracking Challenges at CLIC

What an Event looks like

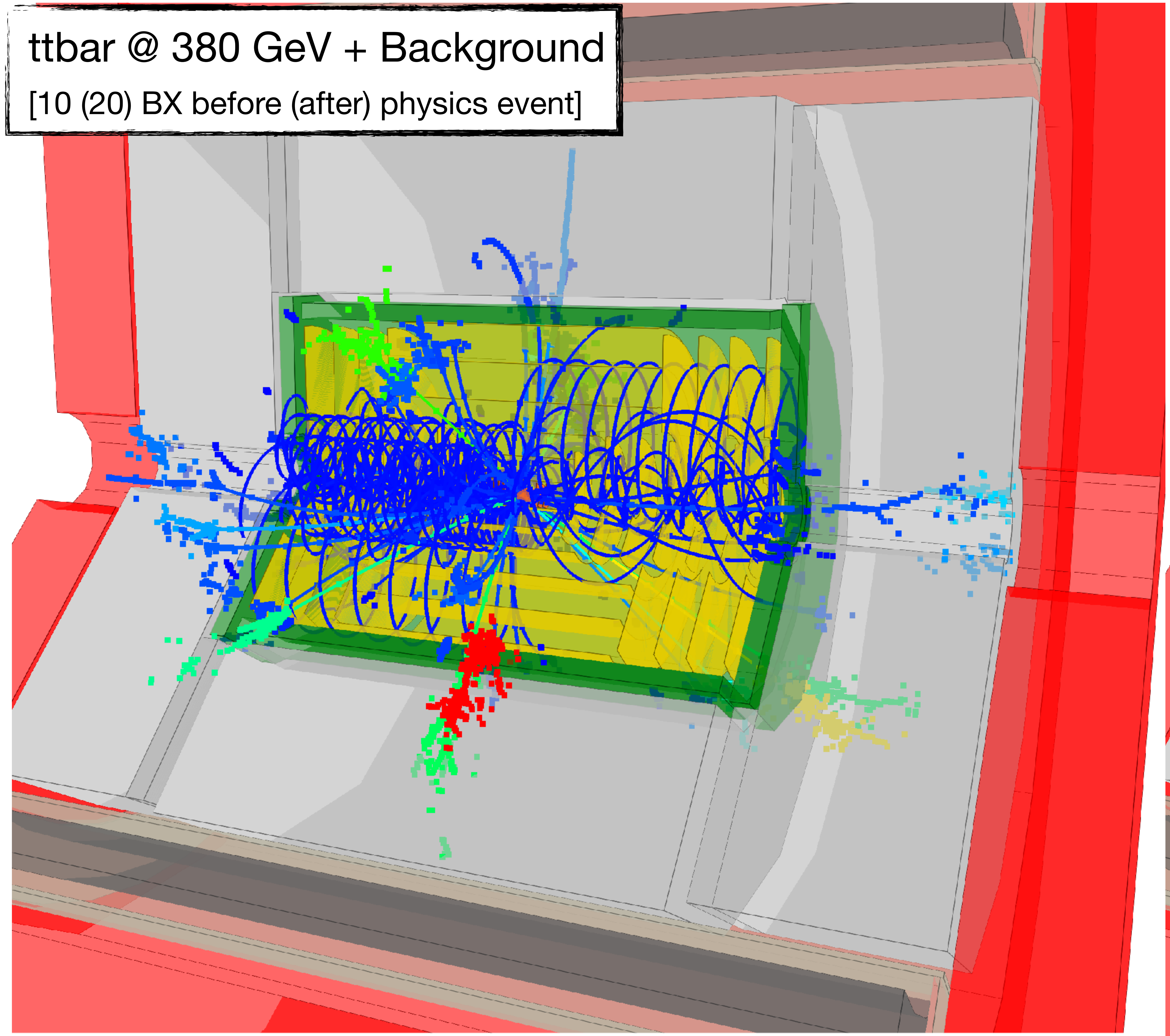
ttbar @ 380 GeV + Background

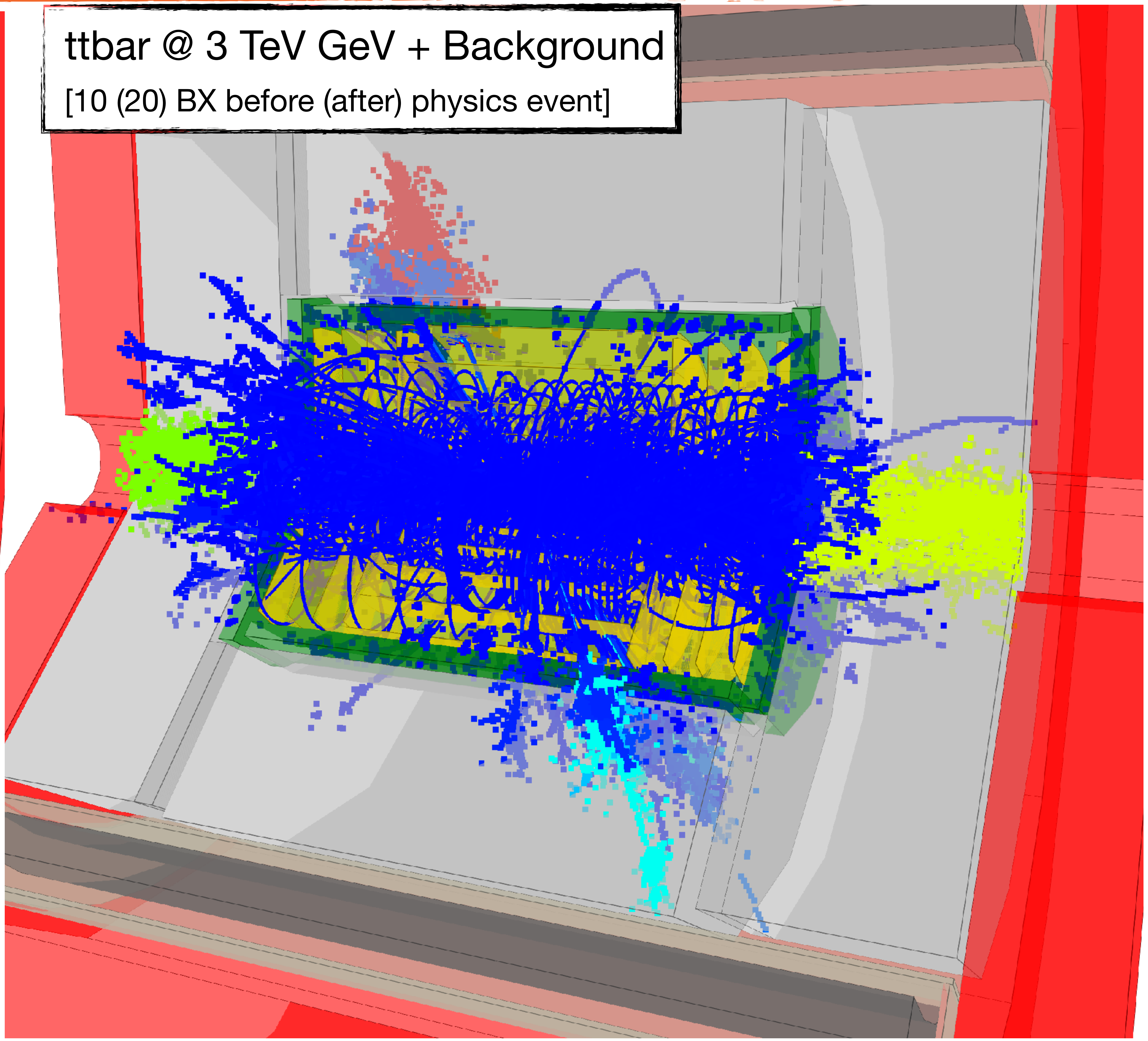
[10 (20) BX before (after) physics event]



Tracking Challenges at CLIC

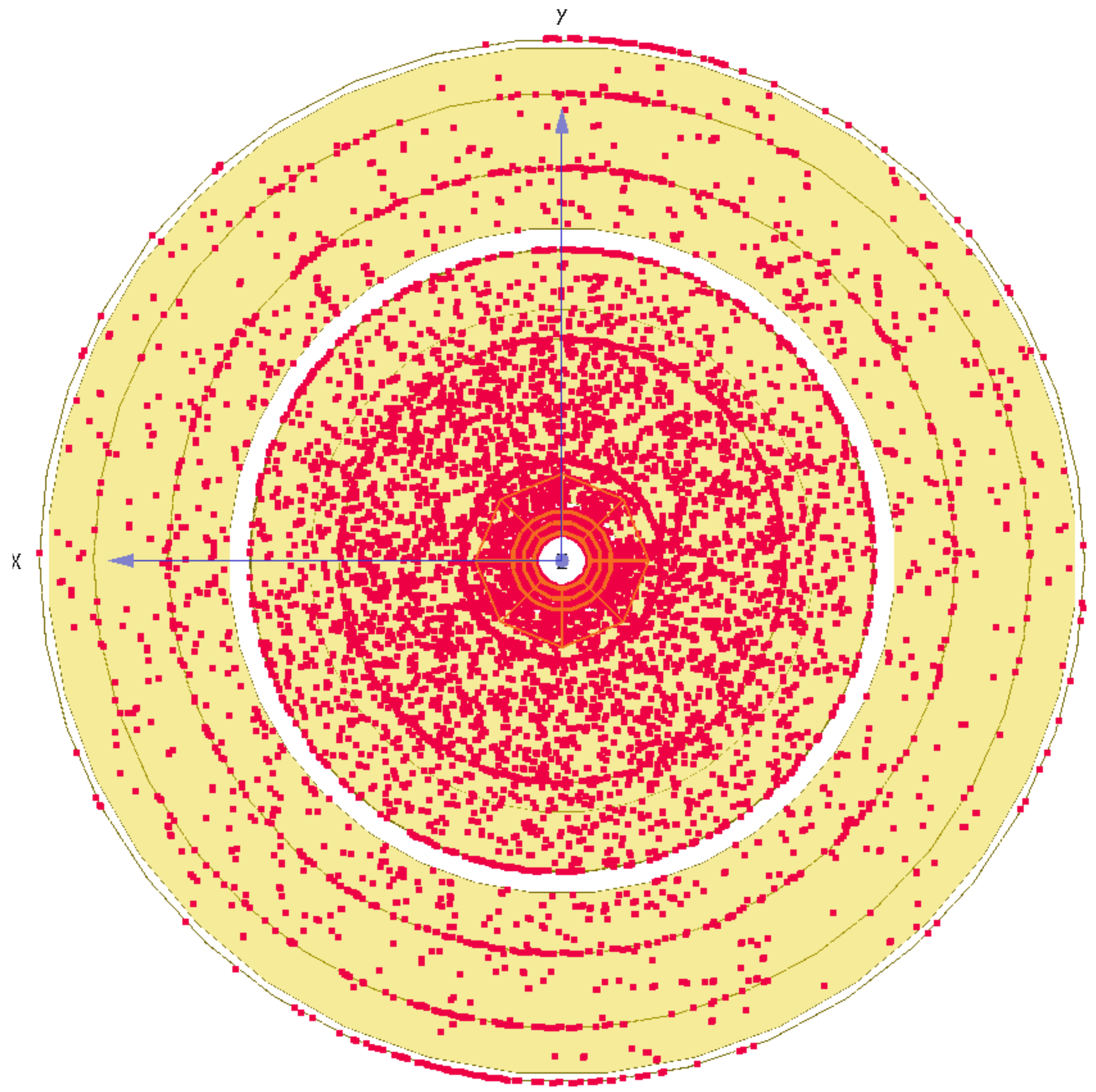
What an Event looks like

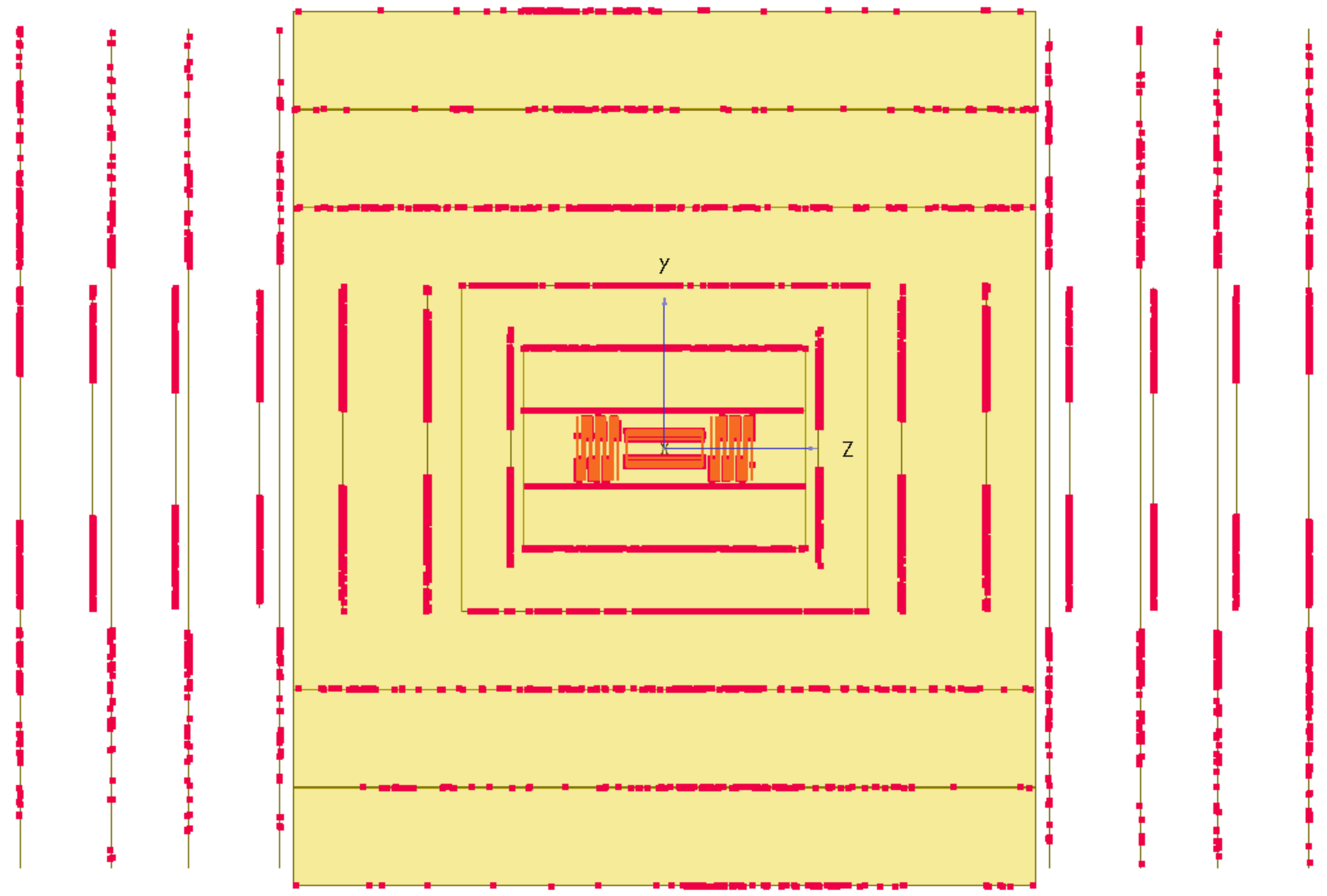




Tracking Challenges at CLIC

and Detector Requirements





ttbar @ 3 TeV + Background

[10 (20) BX before (after) physics event]

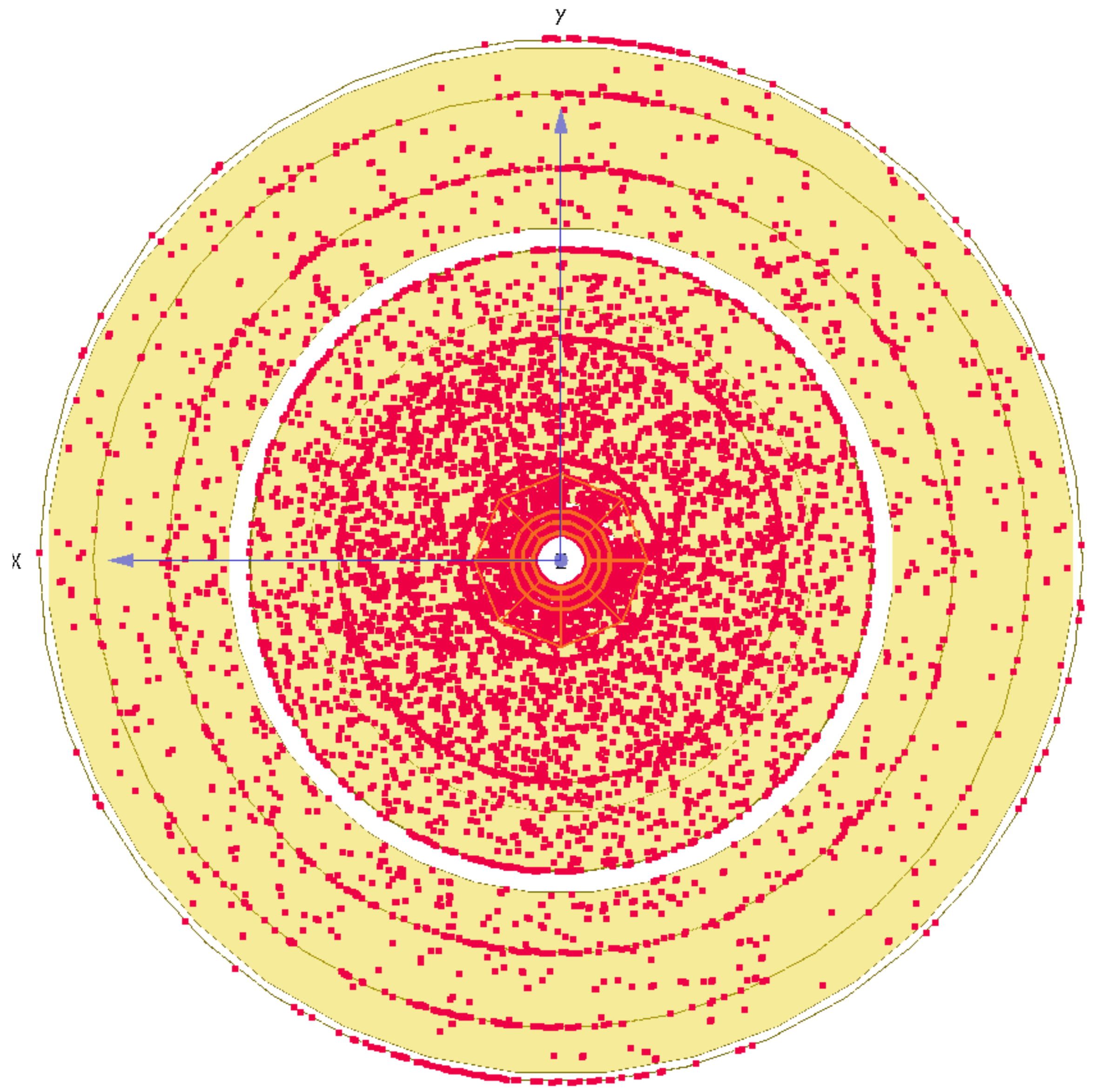
- Precisely reconstructing such events requires:

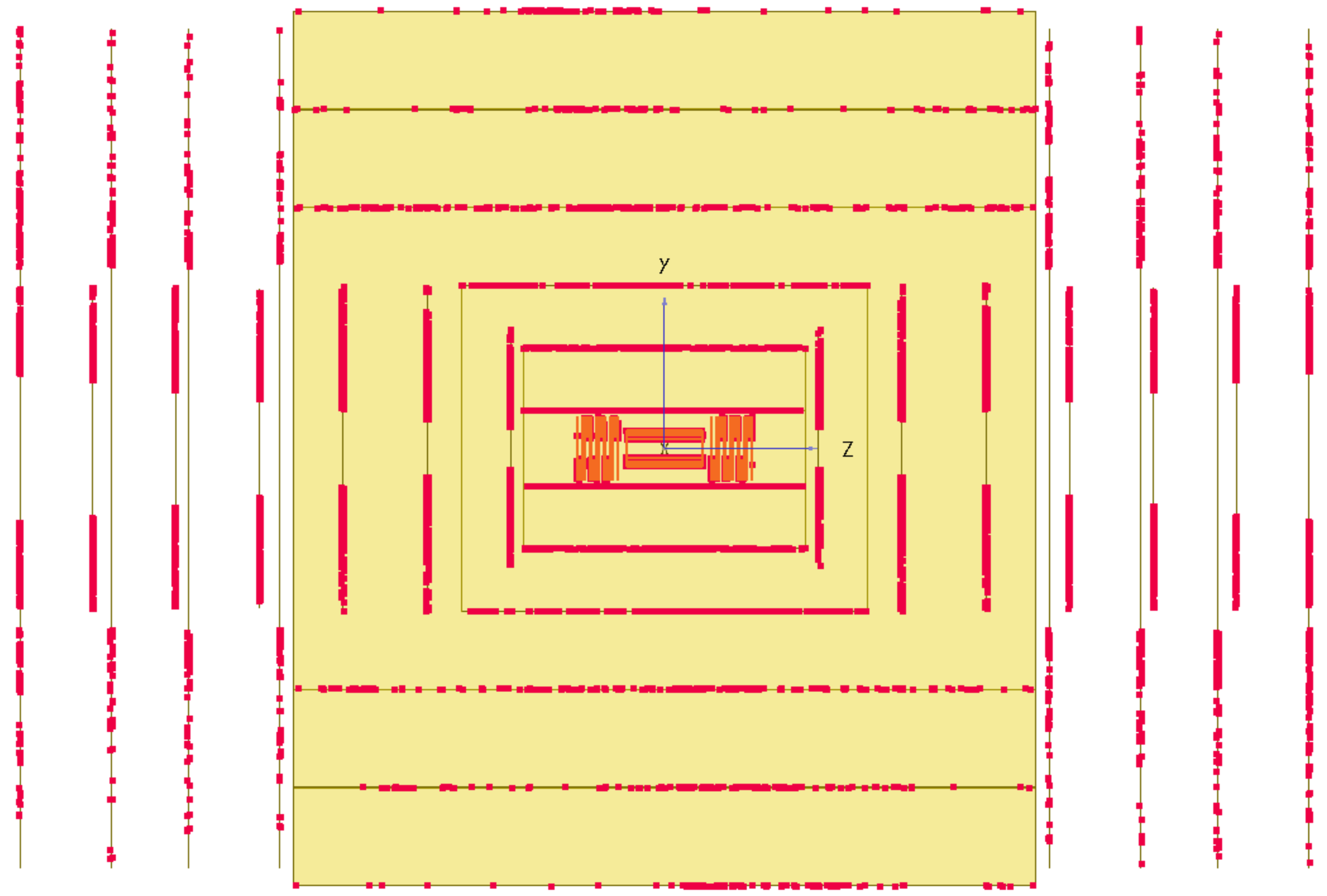
  Physics driven requirements:

  - Excellent momentum and impact parameter resolution
  - High angular coverage

Tracking Challenges at CLIC

and Detector Requirements





ttbar @ 3 TeV + Background

[10 (20) BX before (after) physics event]

- Precisely reconstructing such events requires:

  Physics driven requirements:

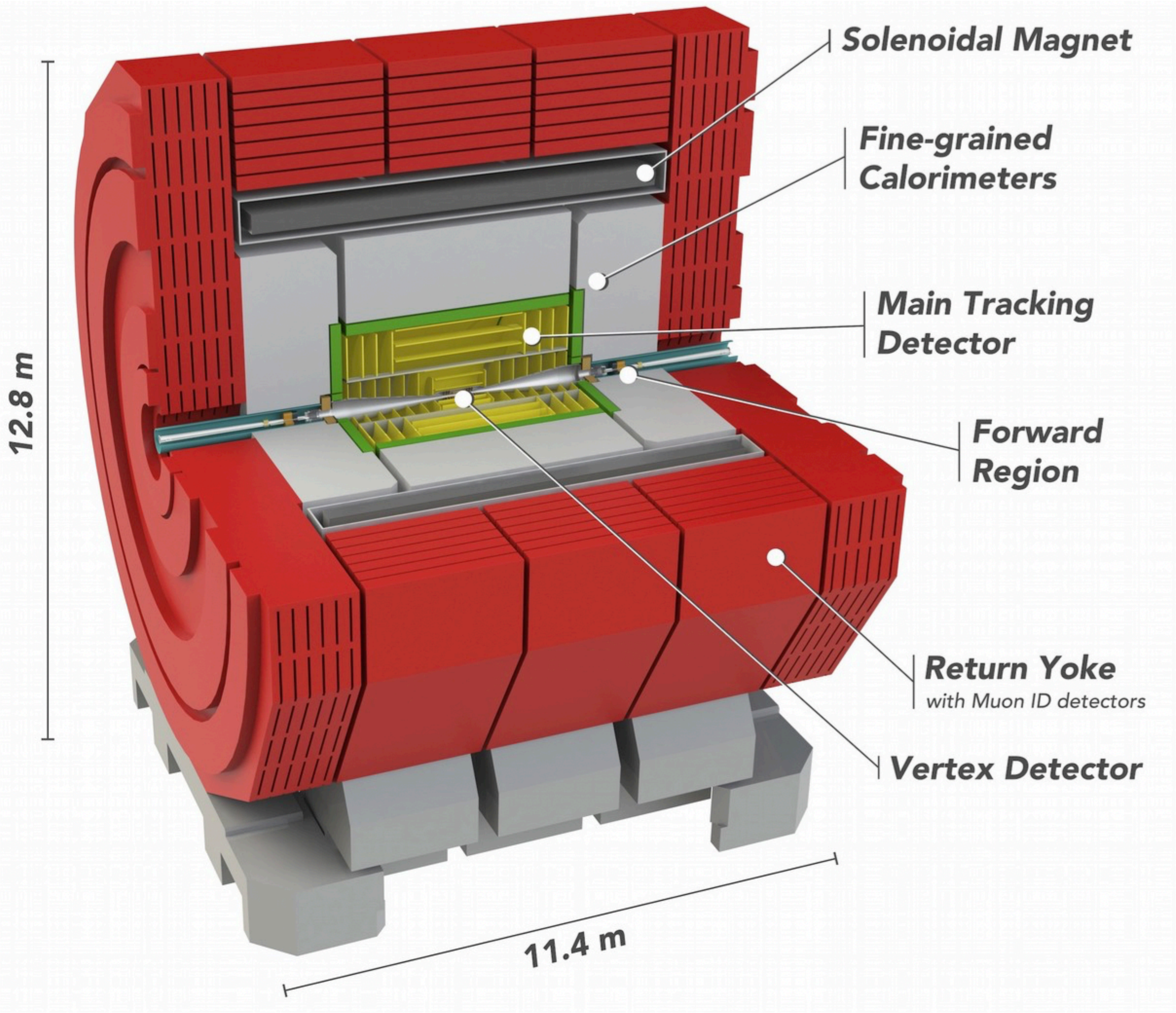
  - Excellent momentum and impact parameter resolution
  - High angular coverage

  Technological consequences:

  - Low material budget
  - Background rejection and separation of bunch crossings:
    - High graunlarity
    - < 10 ns hit time-stamping in tracking

The CLIC Detector Concept

With a tracking focus



Superconducting solenoid with 4T magnetic field

Vertex Detector

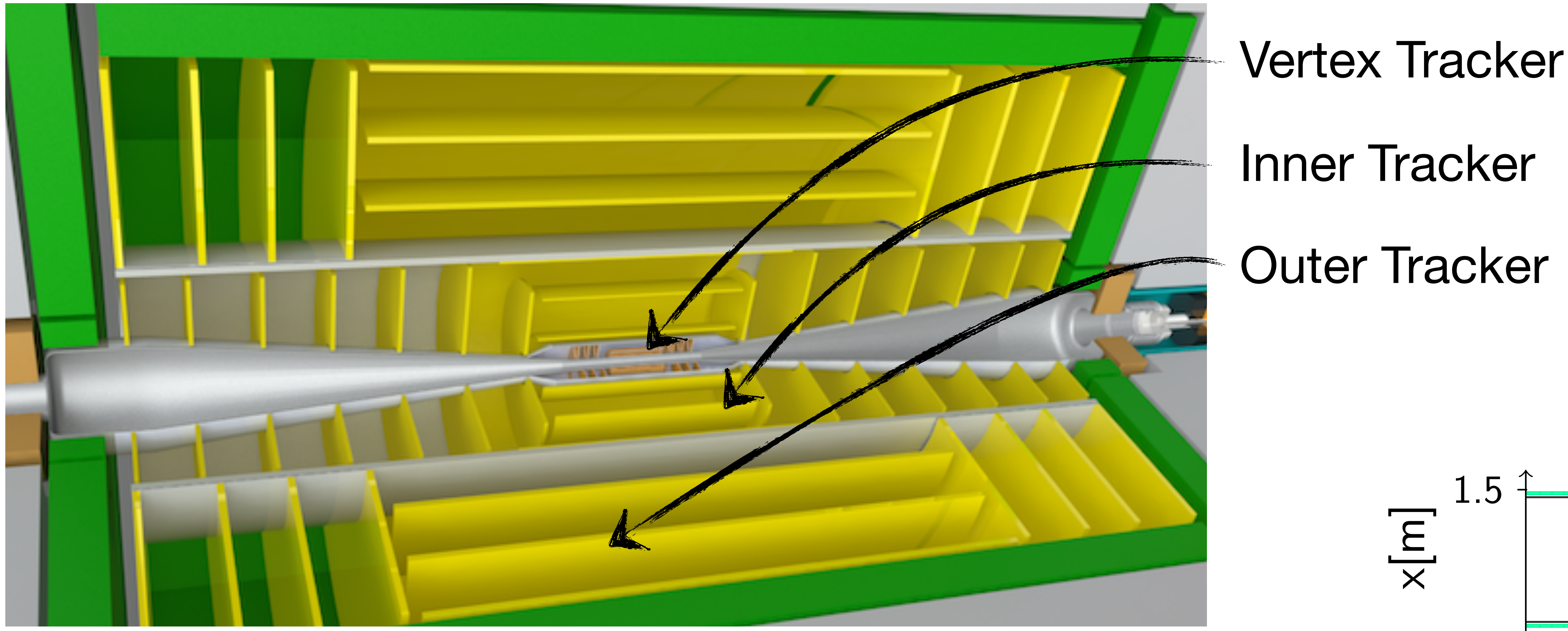
- 3 double layers (barrel) / forward spiral (3 crossings in z) with 25 × 25 μm² pixels; air cooling
- Extremely accurate and light:
  single point resolution = 3 μm
  material budget < 0.2 % X0 per layer

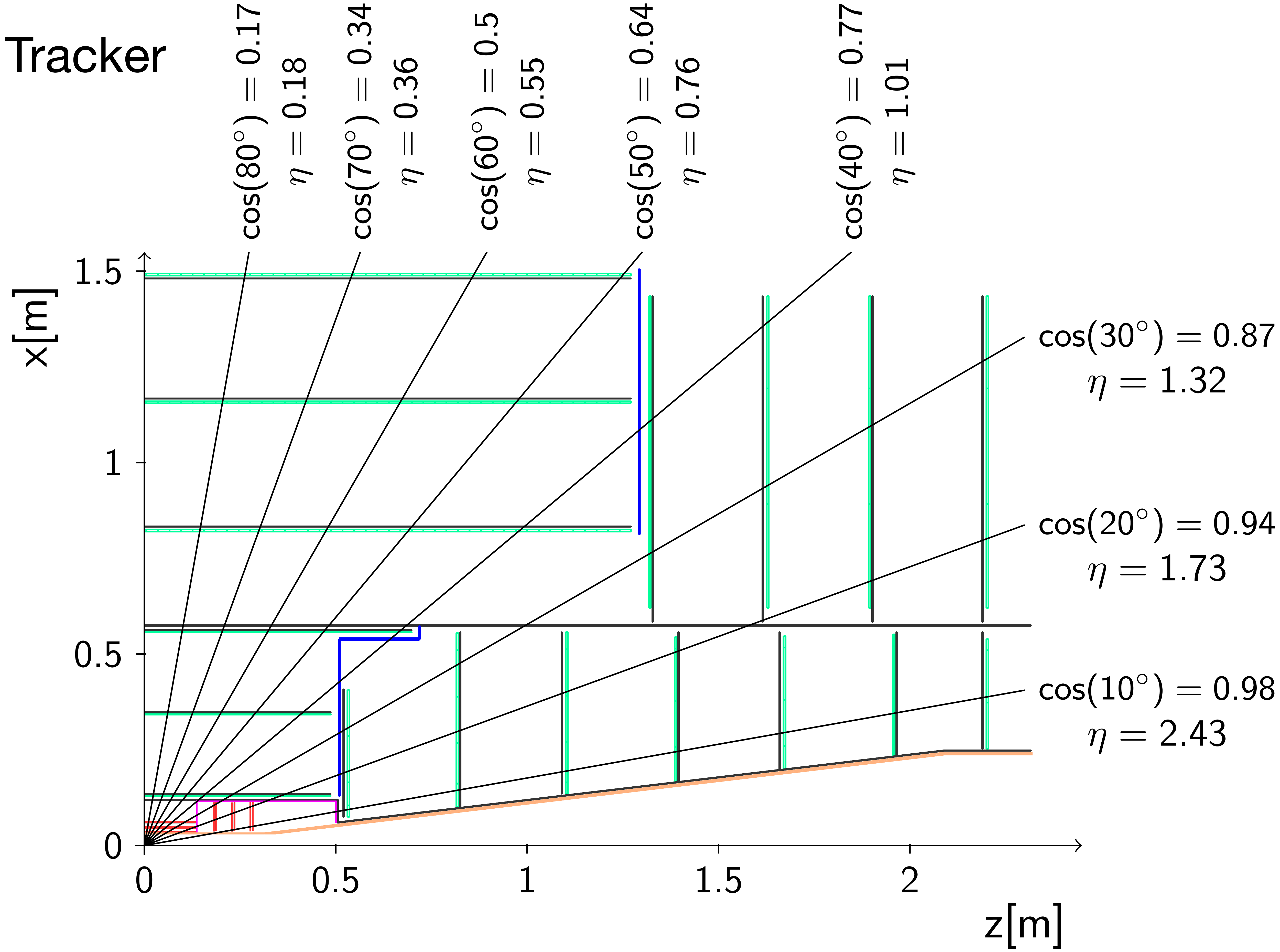
Silicon Tracker

- Tracker composed of large pixels/strips
- Outer radius ~ 1.5 m
- Single point resolution = 7 μm × 90 μm
- Material budget:
  Detector: ~1% X_0 per layer
  Support & services: ~2.5% X_0
- Precise timing for background rejection:
  < 10 ns hit time-stamping in tracking

The CLICDet Tracker

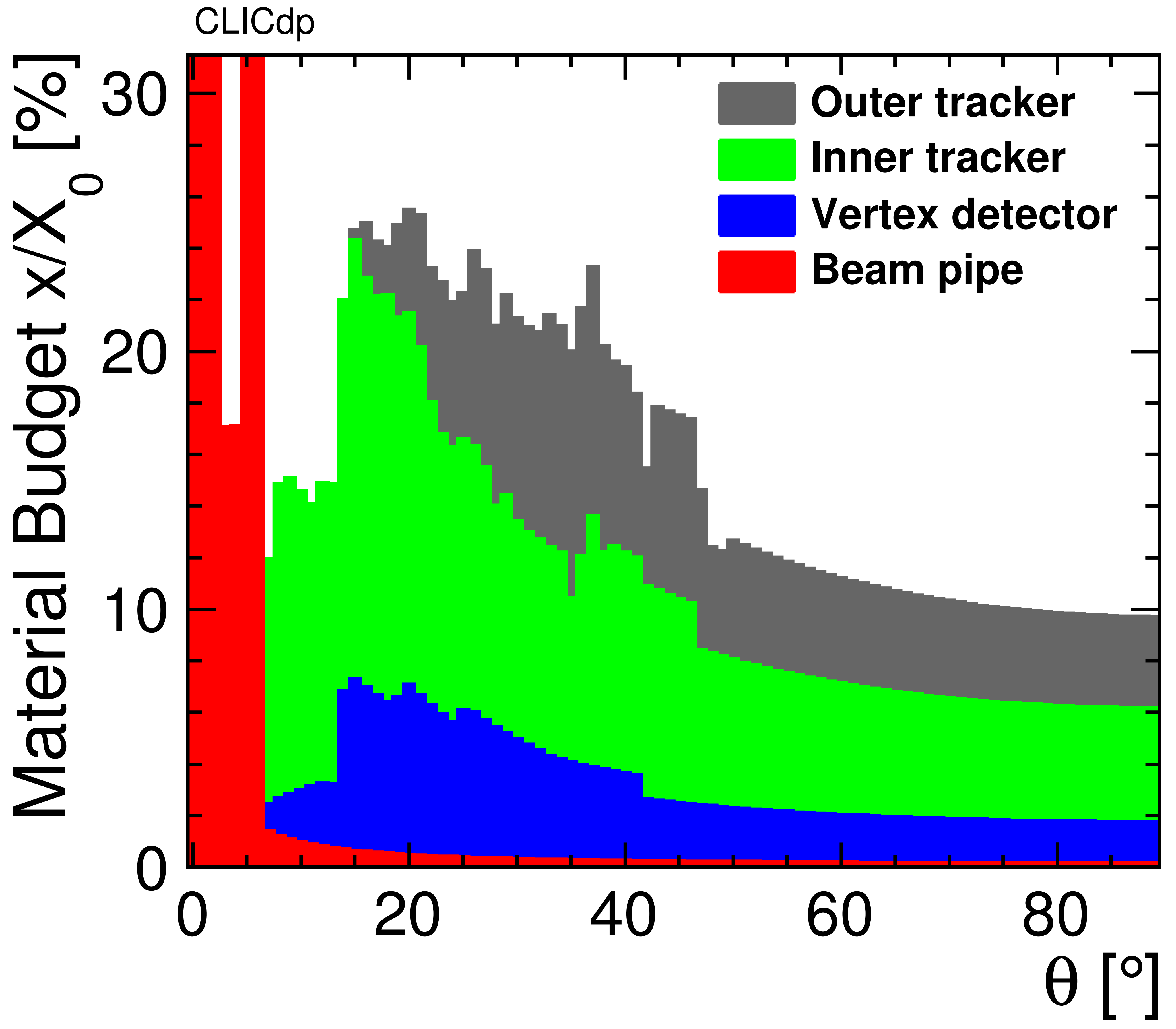
Geometrical Details





The CLICDet Tracker

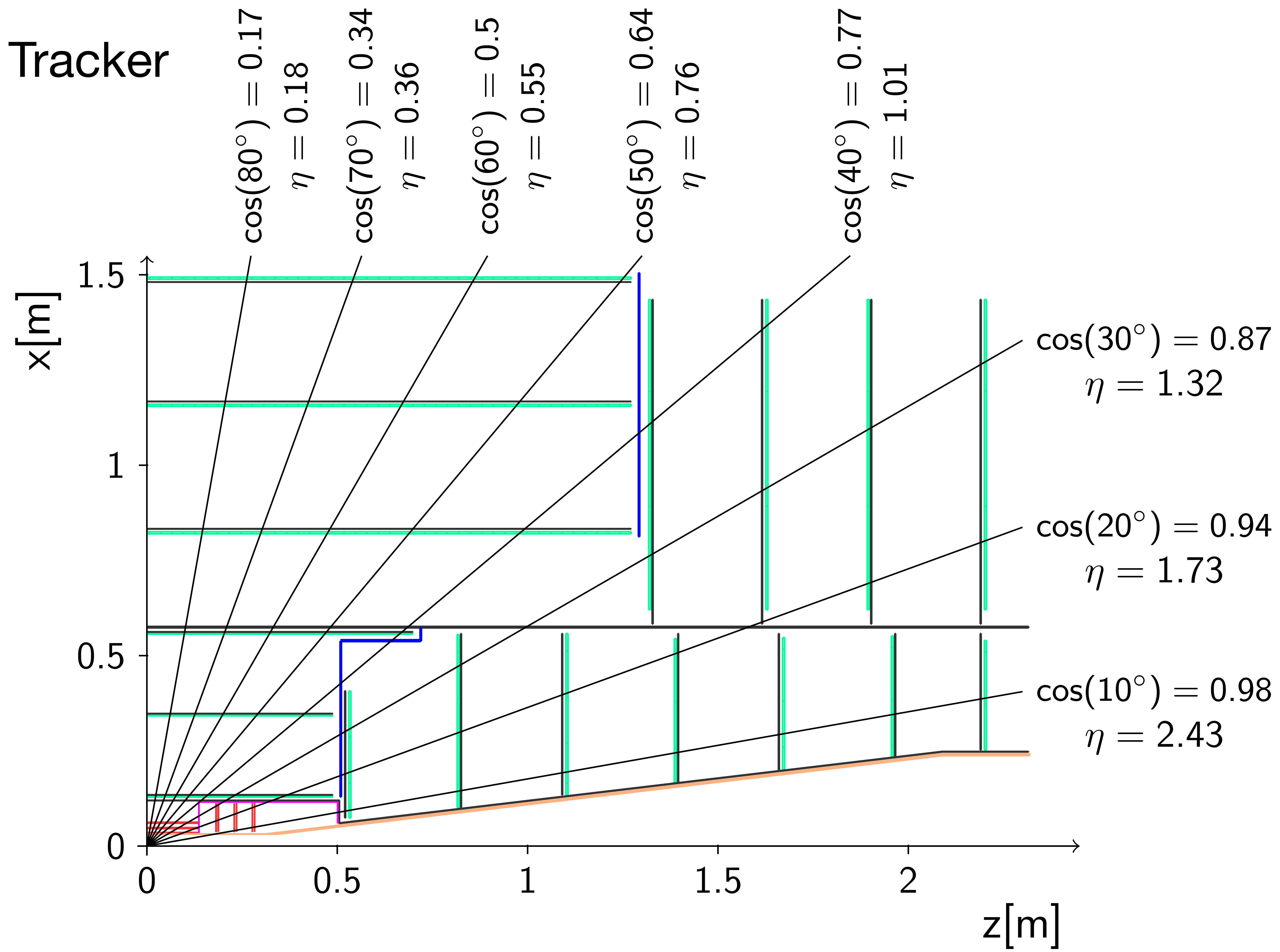
Geometrical Details



Material budget incl. support and services

[in units of λ_I ~ 4% at 90°, 10% at 20°]

- Vertex Tracker
- Inner Tracker
- Outer Tracker



Conformal tracking at CLIC

A new tracking concept, optimized for low-mass trackers

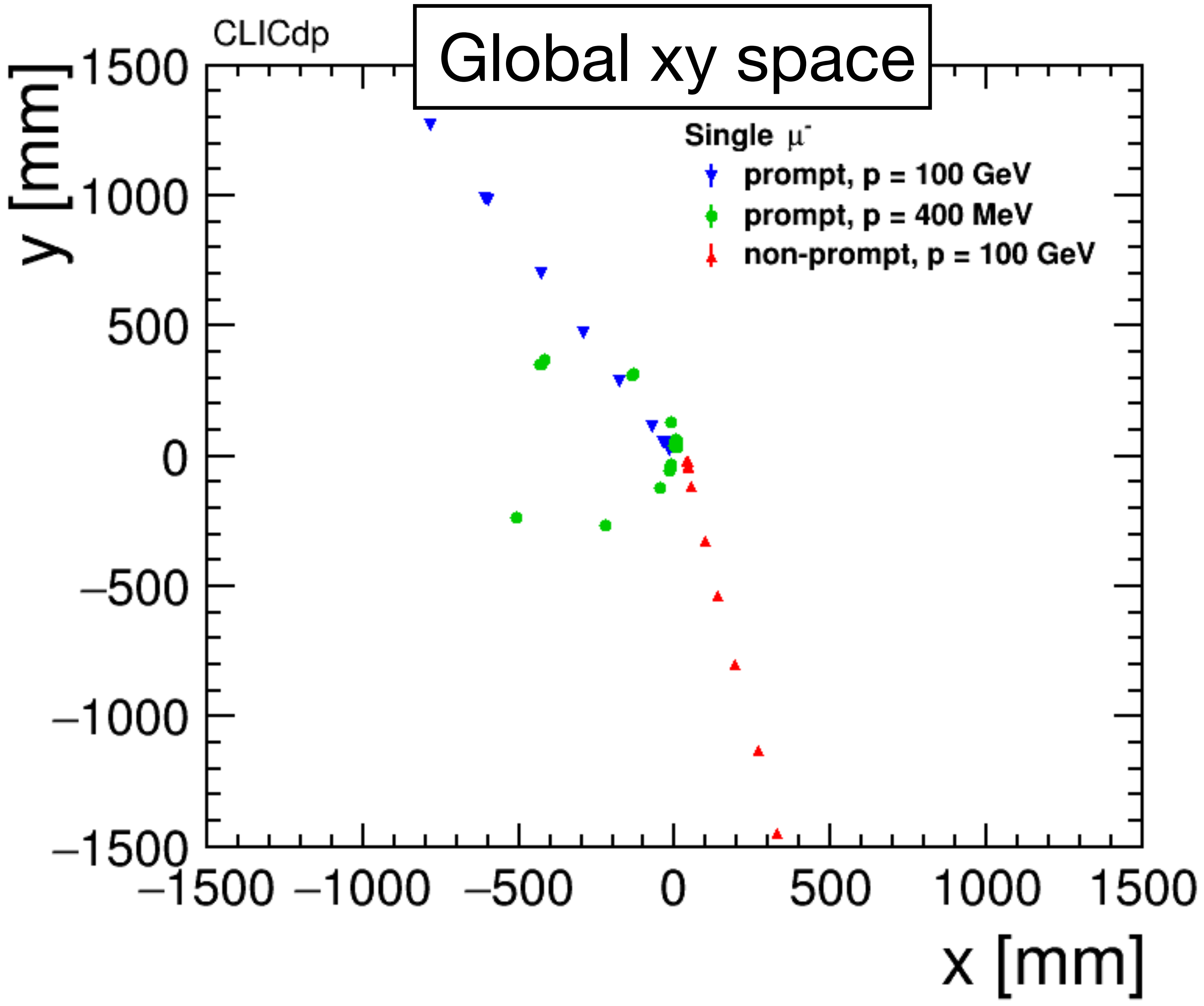
- ***Conformal mapping***: A coordinate transform to aid track finding / pattern recognition
- ***Cellular automaton***: Building tracks from hits in conformal space
- ***Track fitting*** using a Kalman filter and smoother
- ***Track selection*** to purify the track sample by removing mis-identified tracks

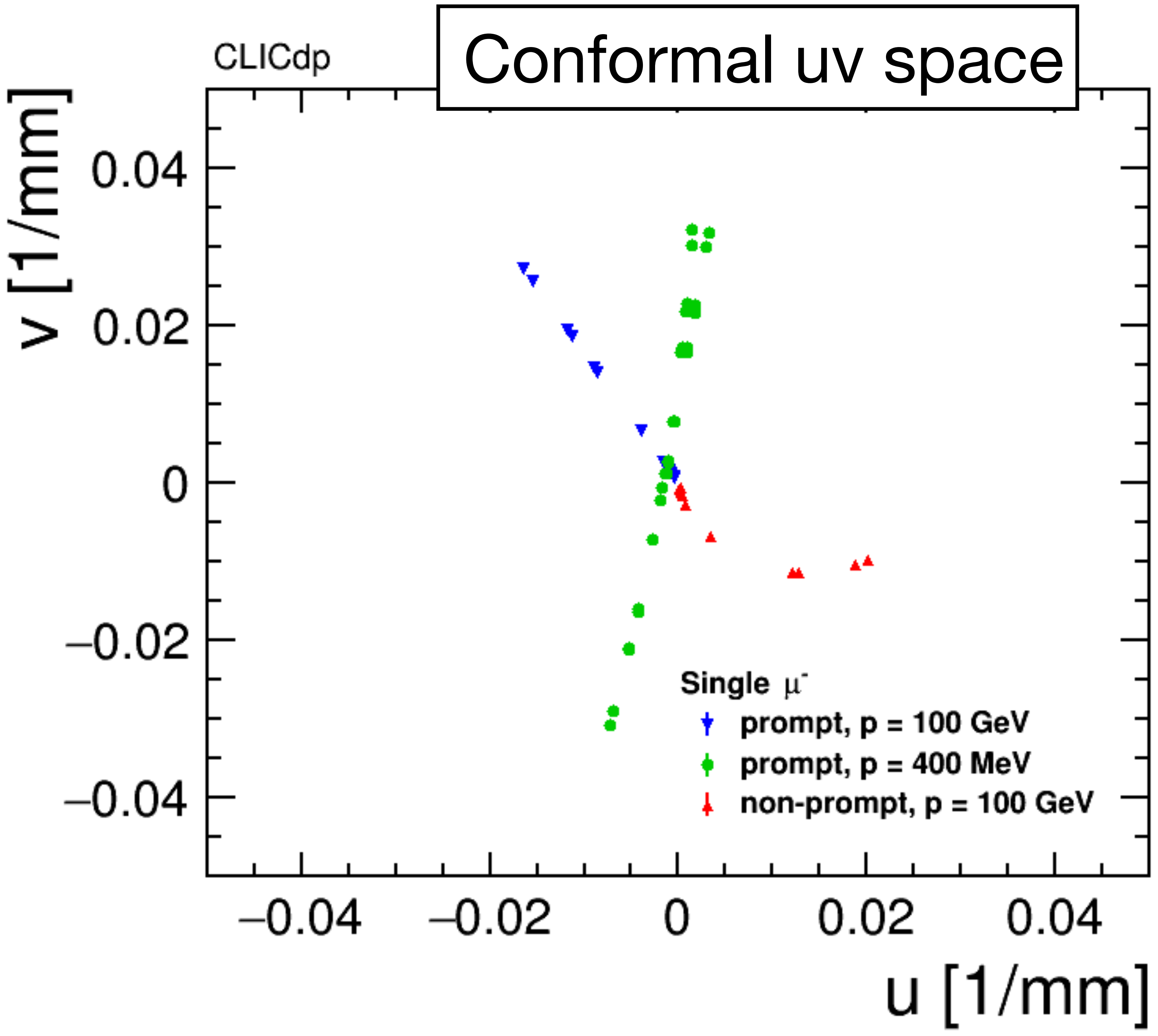
Conformal Mapping

A coordinate transform to aid pattern recognition

- The conformal mapping method is based on the fact that circles passing through the origin of a coordinate system xy can be translated onto straight lines in a new coordinate system uv

$$u = x / (x^2 + y^2) \qquad v = y / (x^2 + y^2)$$





NB:

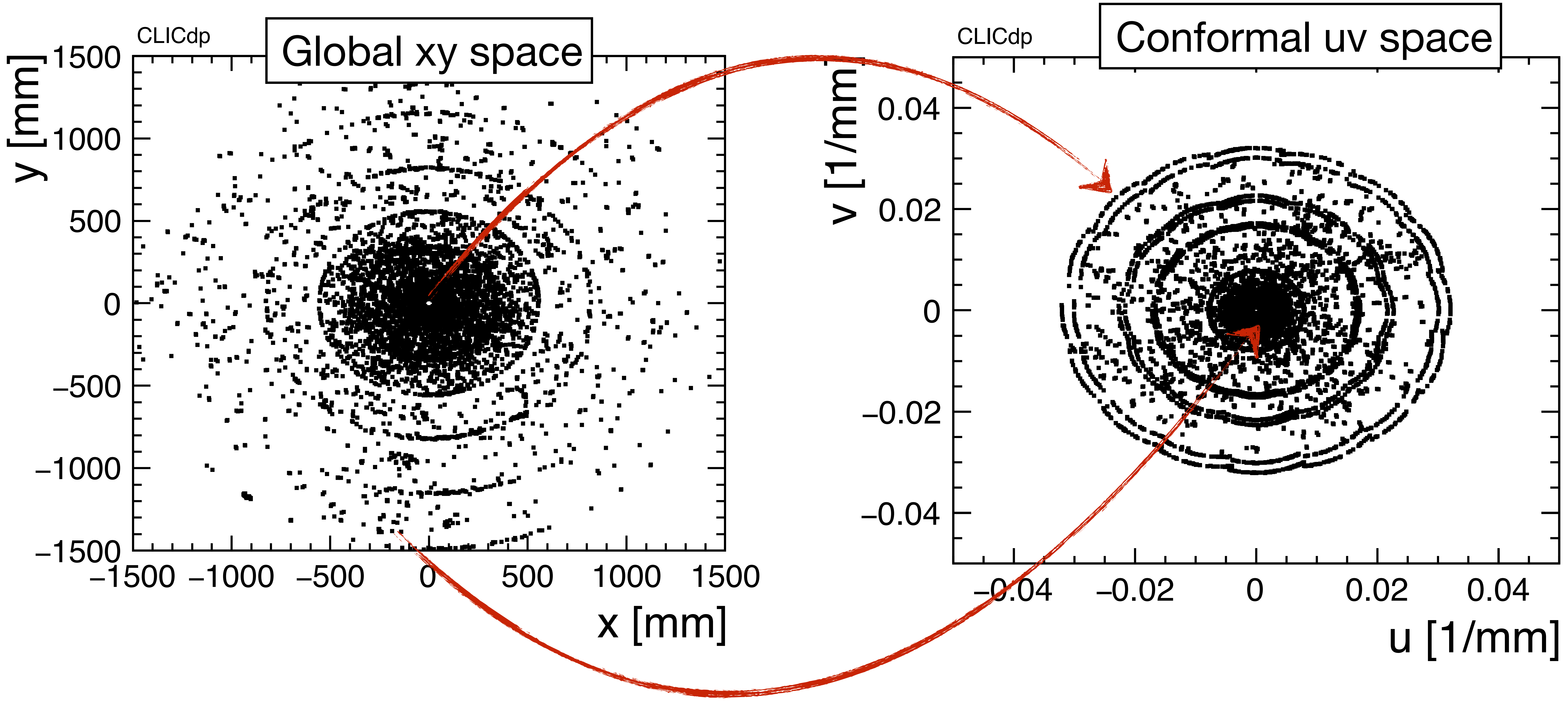
- Multiple scattering results in deviation from straight line
- Displaced tracks do not form a straight line

Conformal Mapping

A coordinate transform to aid pattern recognition

- The conformal mapping method is based on the fact that circles passing through the origin of a coordinate system xy can be translated onto straight lines in a new coordinate system uv

$$u = x / (x^2 + y^2) \qquad v = y / (x^2 + y^2)$$



NB:

- Multiple scattering results in deviation from straight line
- Displaced tracks do not form a straight line
- The innermost hits are in the main tracker, the vertex hits are mapped to the outside

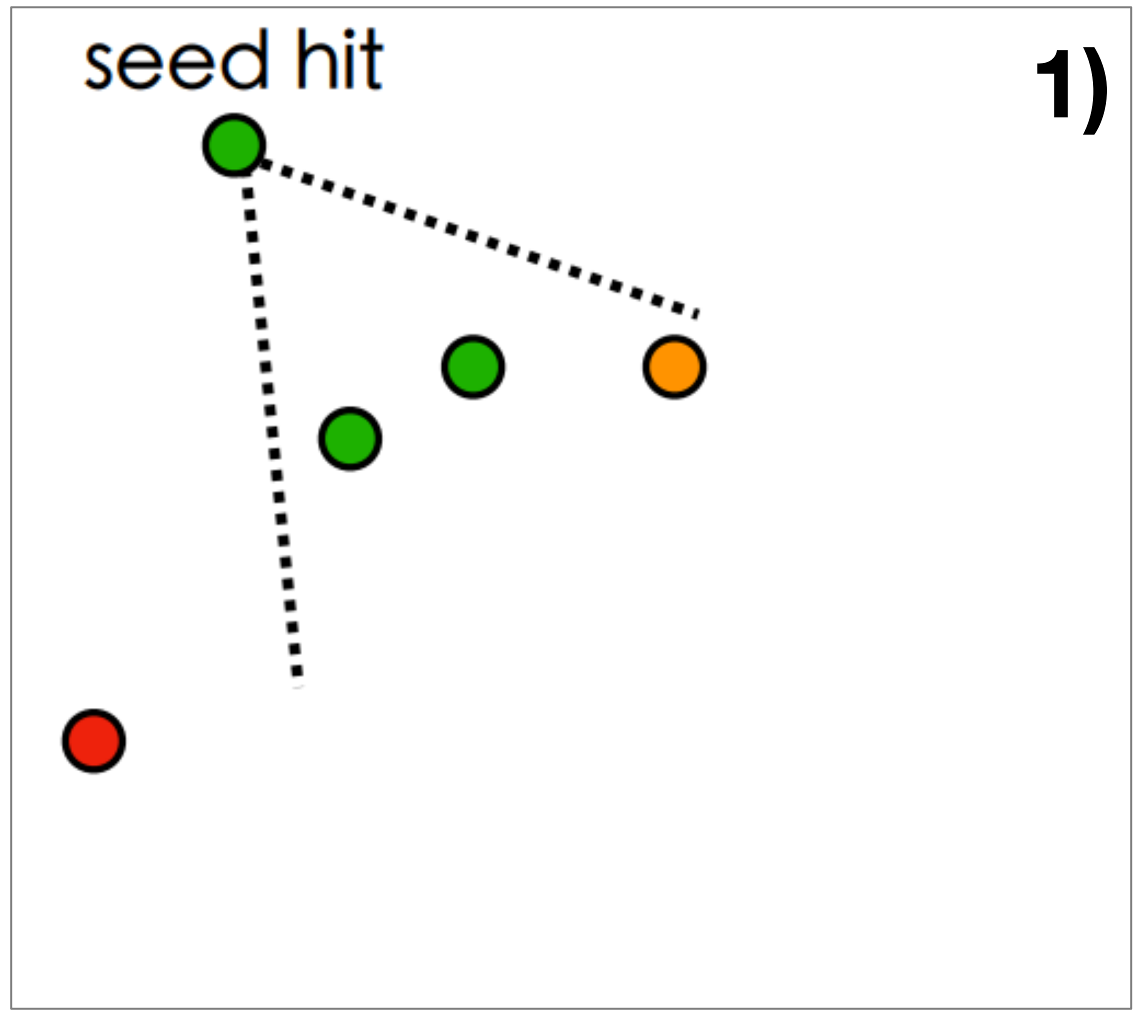
Track Finding: Cellular Automaton

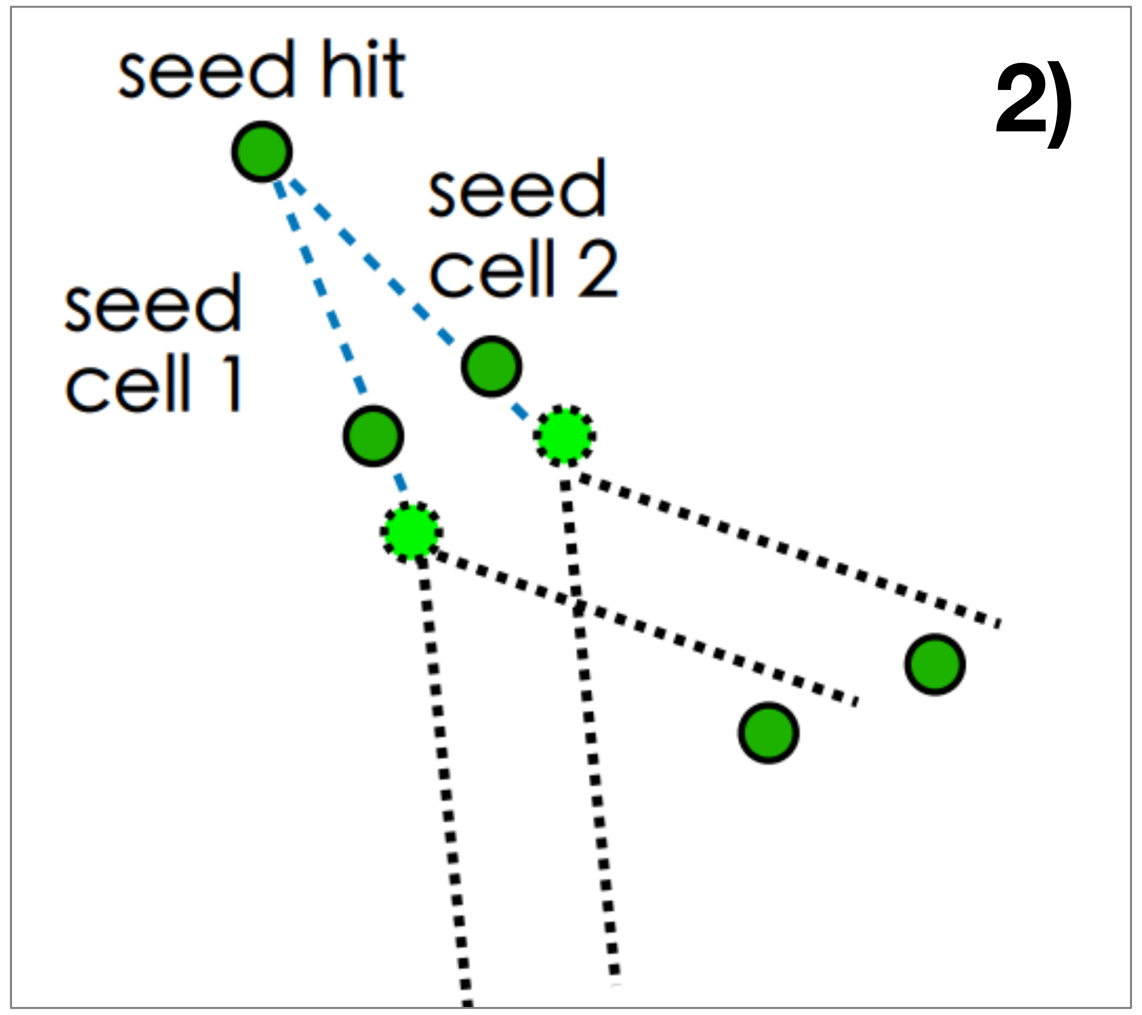
A straight line fit is not enough

- Cell is a segment between two hits with a weight associated
- Cellular tracks are vectors of cells
- Cellular automaton (creation/extension of cells) can be used to:
  - Build cellular tracks
  - Extend cellular tracks

Selecting the tracks

1. Define seed hits
2. Create cellular track candidate
   1. Define hit neighbor if in cone, not too far and not yet used
   2. Cell is created with weight, each link increments weight
   3. Seed cell is extended to virtual hits
3. Select cellular tracks among candidates
4. Mark hits in cellular tracks as used





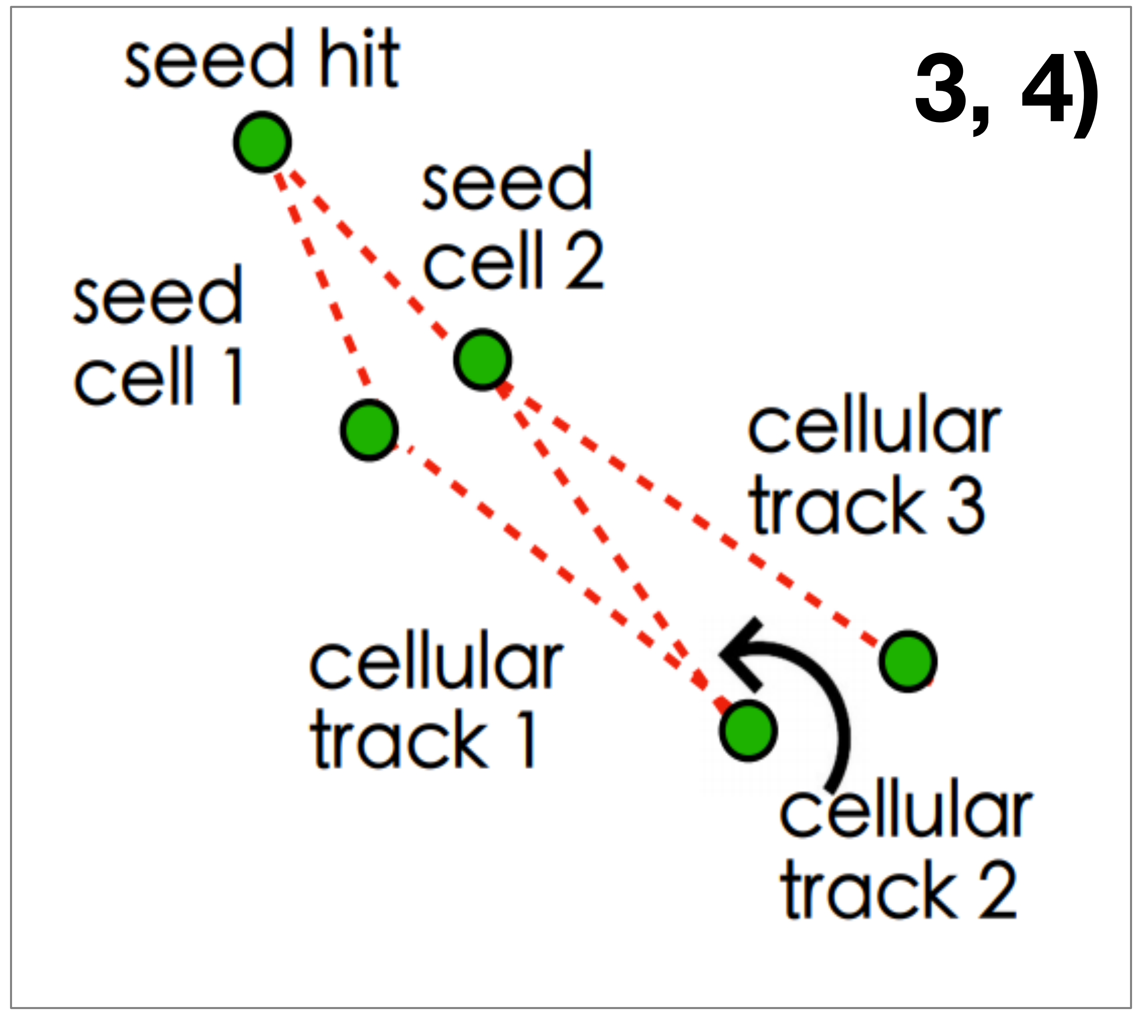
Track Finding: Cellular Automaton

A straight line fit is not enough

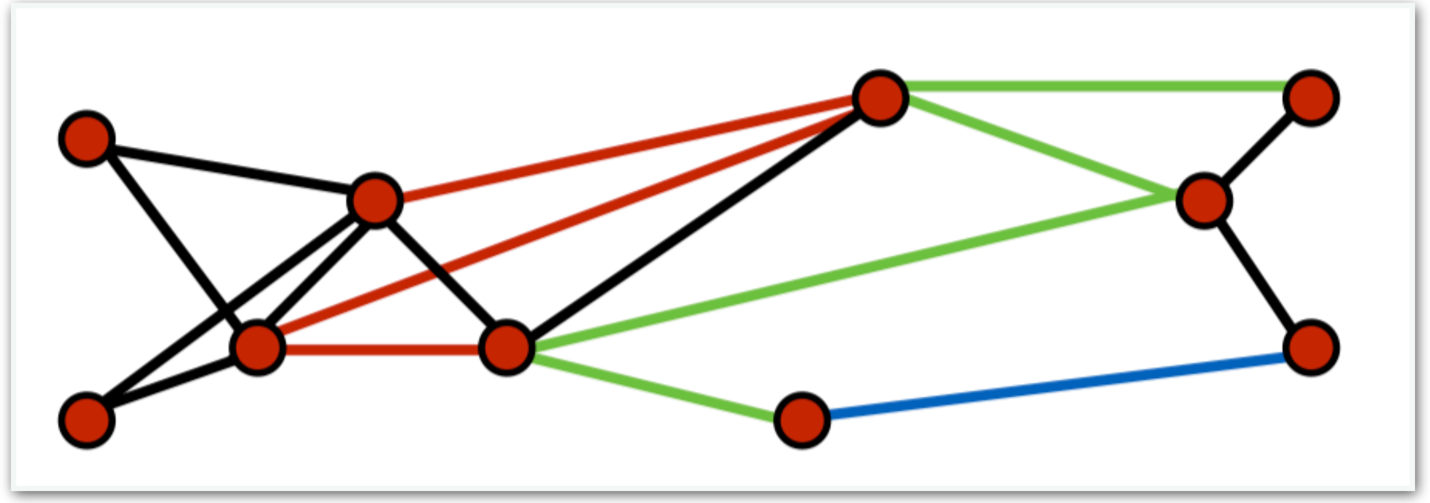
- Cell is a segment between two hits with a weight associated
- Cellular tracks are vectors of cells
- Cellular automaton (creation/extension of cells) can be used to:
  - Build cellular tracks
  - Extend cellular tracks

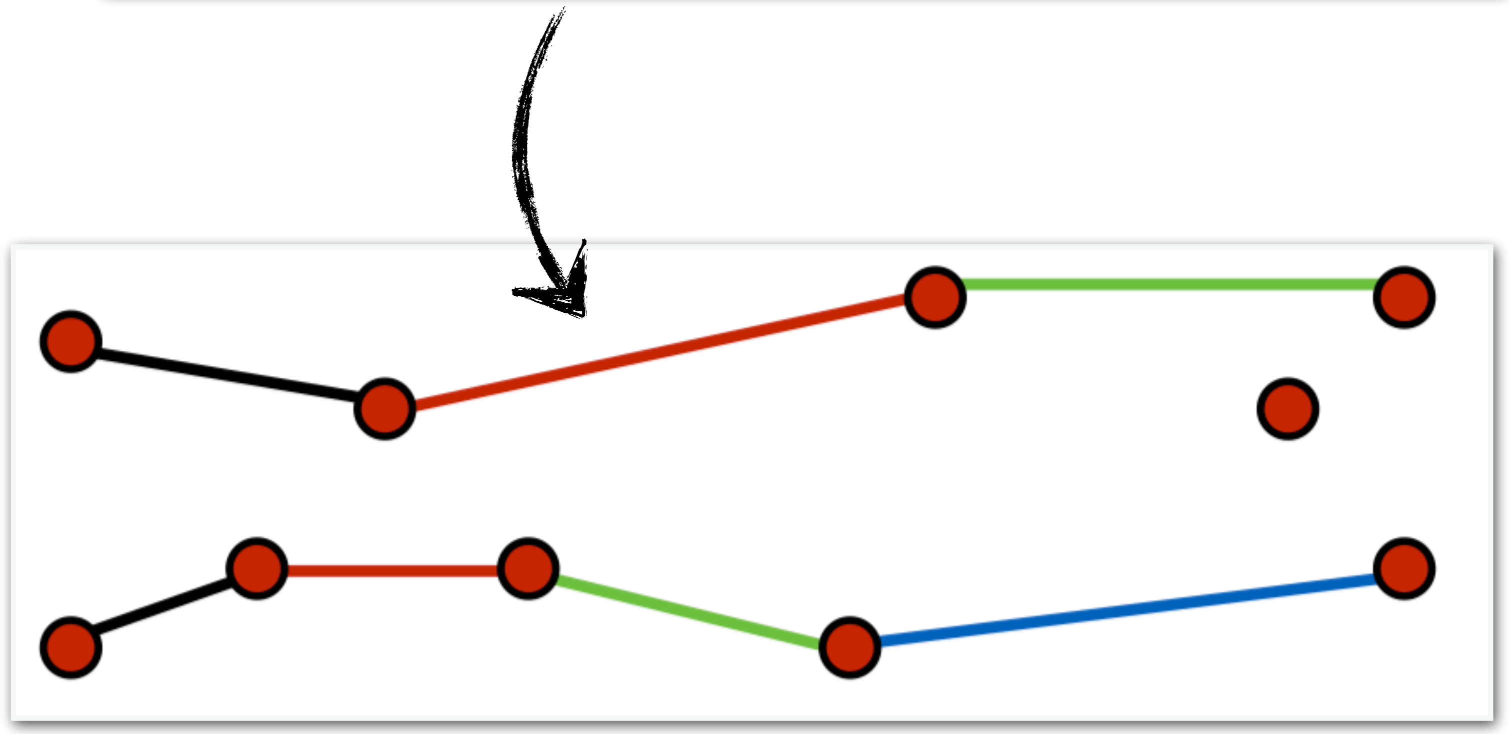
Selecting the tracks

1. Define seed hits
2. Create cellular track candidate
3. Select cellular tracks among candidates
   - create valid tracks starting from higher weight back to the seed cell
     - if more paths available, branch the track
4. Mark hits in cellular tracks as used



w = 0
w = 1
w = 2
w = 3





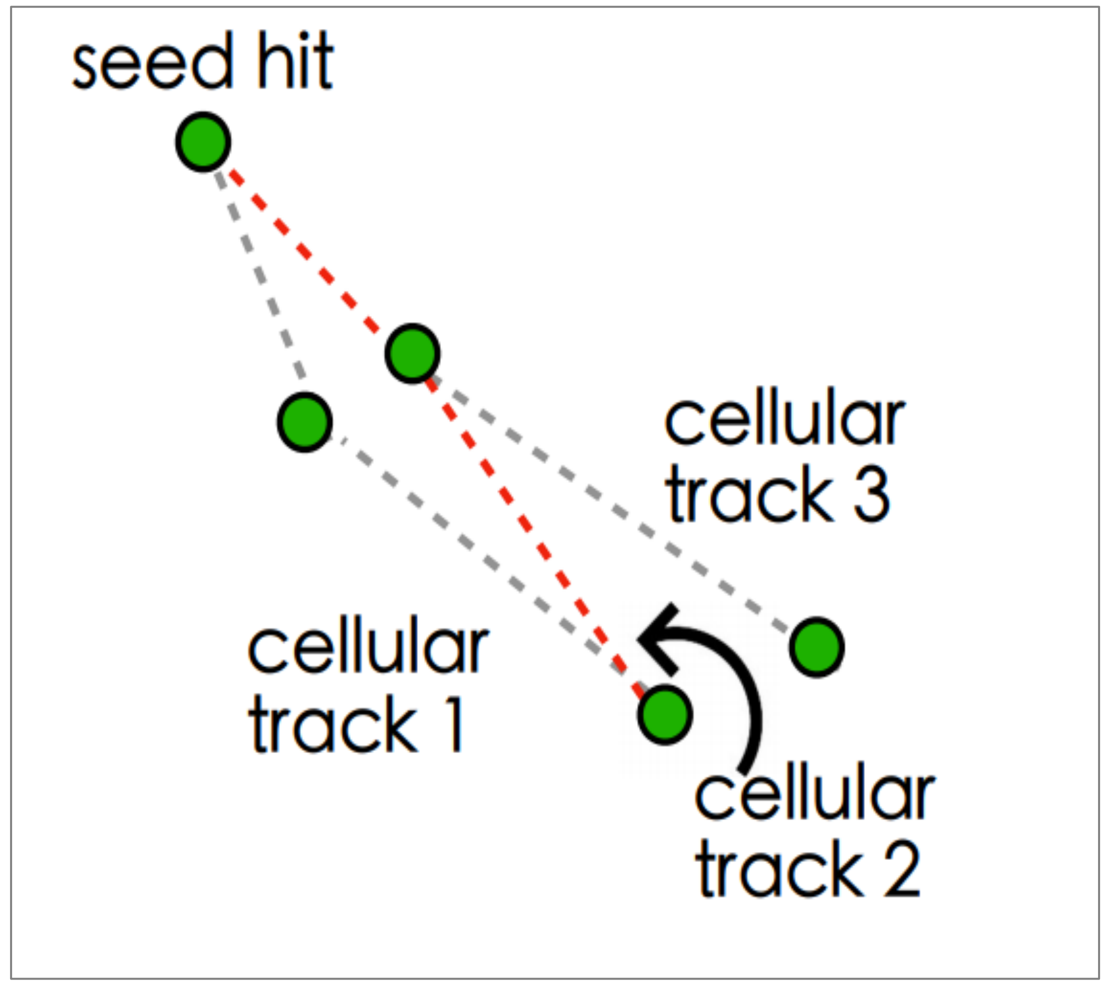
Track Finding: Cellular Automaton

A straight line fit is not enough

- Cell is a segment between two hits with a weight associated
- Cellular tracks are vectors of cells
- Cellular automaton (creation/extension of cells) can be used to:
  - Build cellular tracks
  - Extend cellular tracks

Building the tracks

- For all cellular tracks originating from one seed hits
  - linear regression in (u, v) and (s, z) to determine χ^2
  - hits progressively removed, refit, select track with lowest χ^2
- Clones (with 2 or more overlapping hits):
  - Keep longest track
  - If similar length, prefer best χ^2



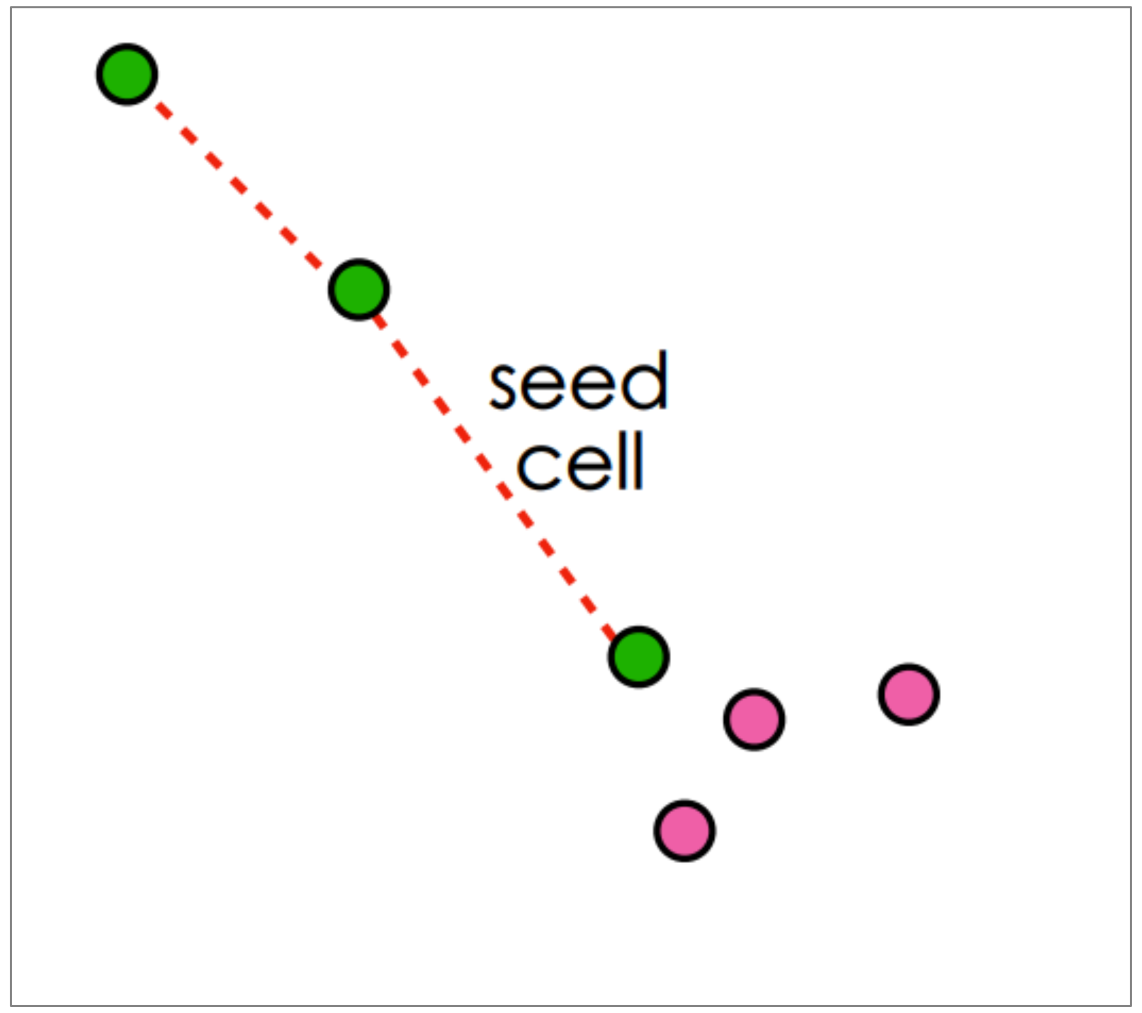
Track Finding: Cellular Automaton

A straight line fit is not enough

- Cell is a segment between two hits with a weight associated
- Cellular tracks are vectors of cells
- Cellular automaton (creation/extension of cells) can be used to:
  - Build cellular tracks
  - Extend cellular tracks

Extending the tracks

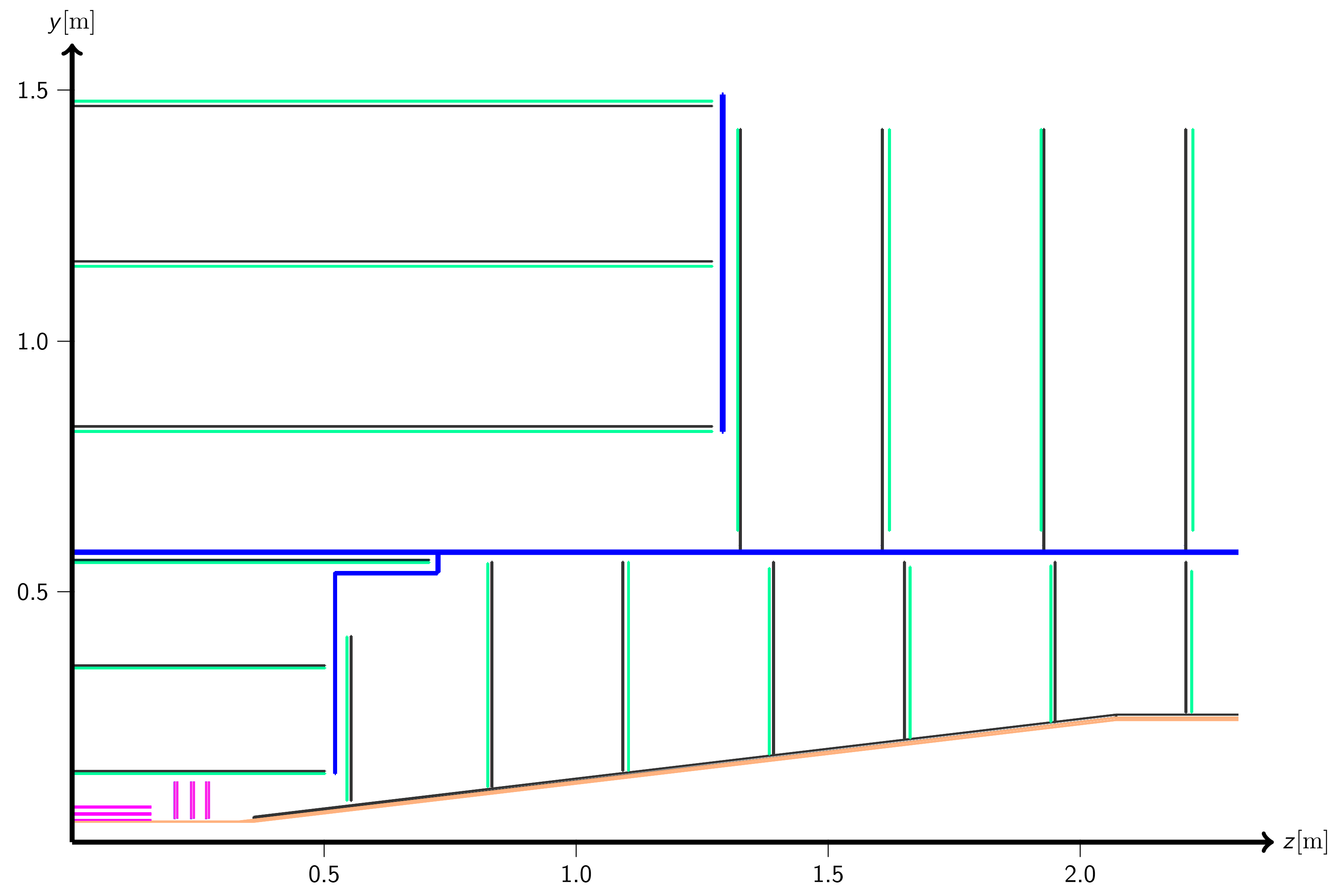
- Starting from the last cell of cellular track
- Search for neighbors in the following layer
- The track is refitted for each hit candidate
  hit is accepted/rejected based on a χ^2 cut



Full Pattern Recognition Chain in CLICDet

A concrete application

Algorithm	Hit collection	Configuration
Build tracks	Vertex barrel	Standard cuts
Extend tracks	Vertex endcap	Standard cuts
Build tracks	Vertex	Looser cuts (angle x 5)
Build tracks	Vertex	Looser cuts (angle x 10; χ^2 x 20))
Extend tracks	Tracker	Looser cuts (angle x 10; χ^2 x 20)
Build tracks	Vertex + Tracker	Displaced cuts



5 steps to reconstruct prompt tracks:

- from vertex detector to silicon tracker
- min number of hits = 4
- standard or looser (angle or χ^2) cuts

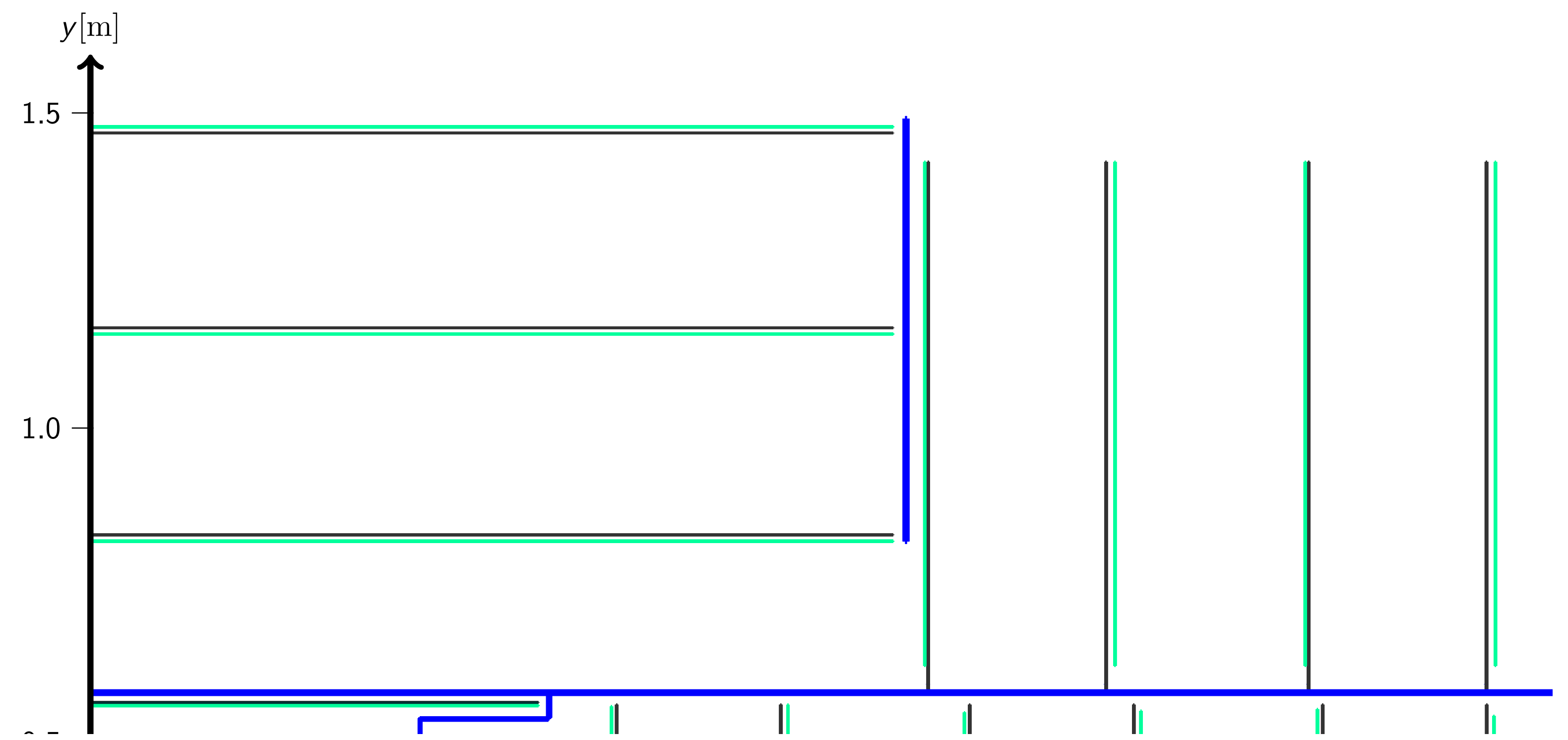
1 final step to reconstruct displaced tracks:

- quadratic terms in the CT fit added
- inverted order search: from silicon tracker to vertex detector
- broader search angle than for prompt tracks

Full Pattern Recognition Chain in CLICDet

A concrete application

Algorithm	Hit collection	Configuration
Build tracks	Vertex barrel	Standard cuts
Extend tracks	Vertex endcap	Standard cuts
Build tracks	Vertex	Looser cuts (angle x 5)
Build tracks	Vertex	Looser cuts (angle x 10; χ^2 x 20))
Extend tracks	Tracker	Looser cuts (angle x 10; χ^2 x 20)
Build tracks	Vertex + Tracker	Displaced cuts

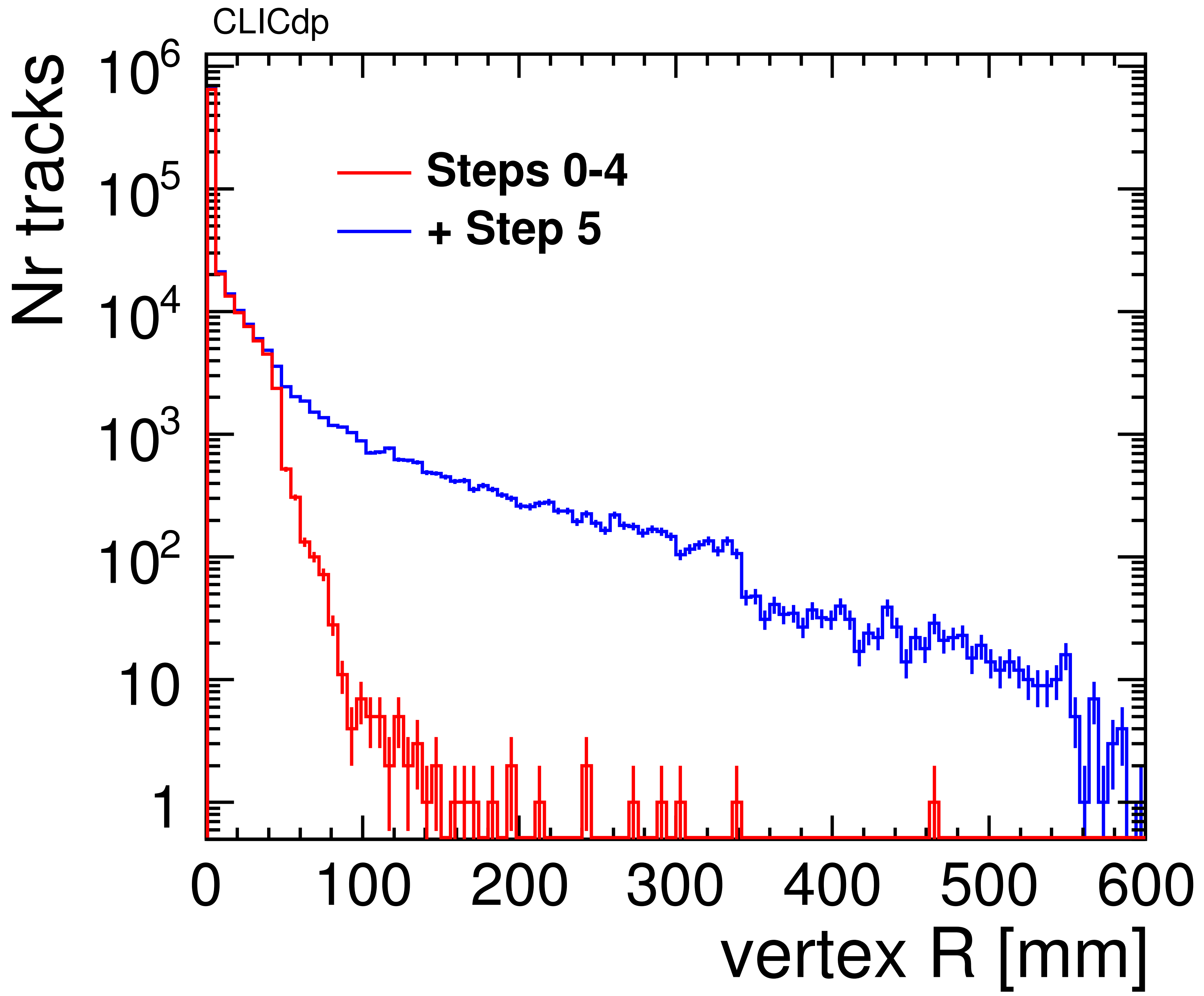


5 steps to reconstruct prompt tracks:

- from vertex detector to silicon tracker
- min number of hits = 4
- standard or looser (angle or χ^2) cuts

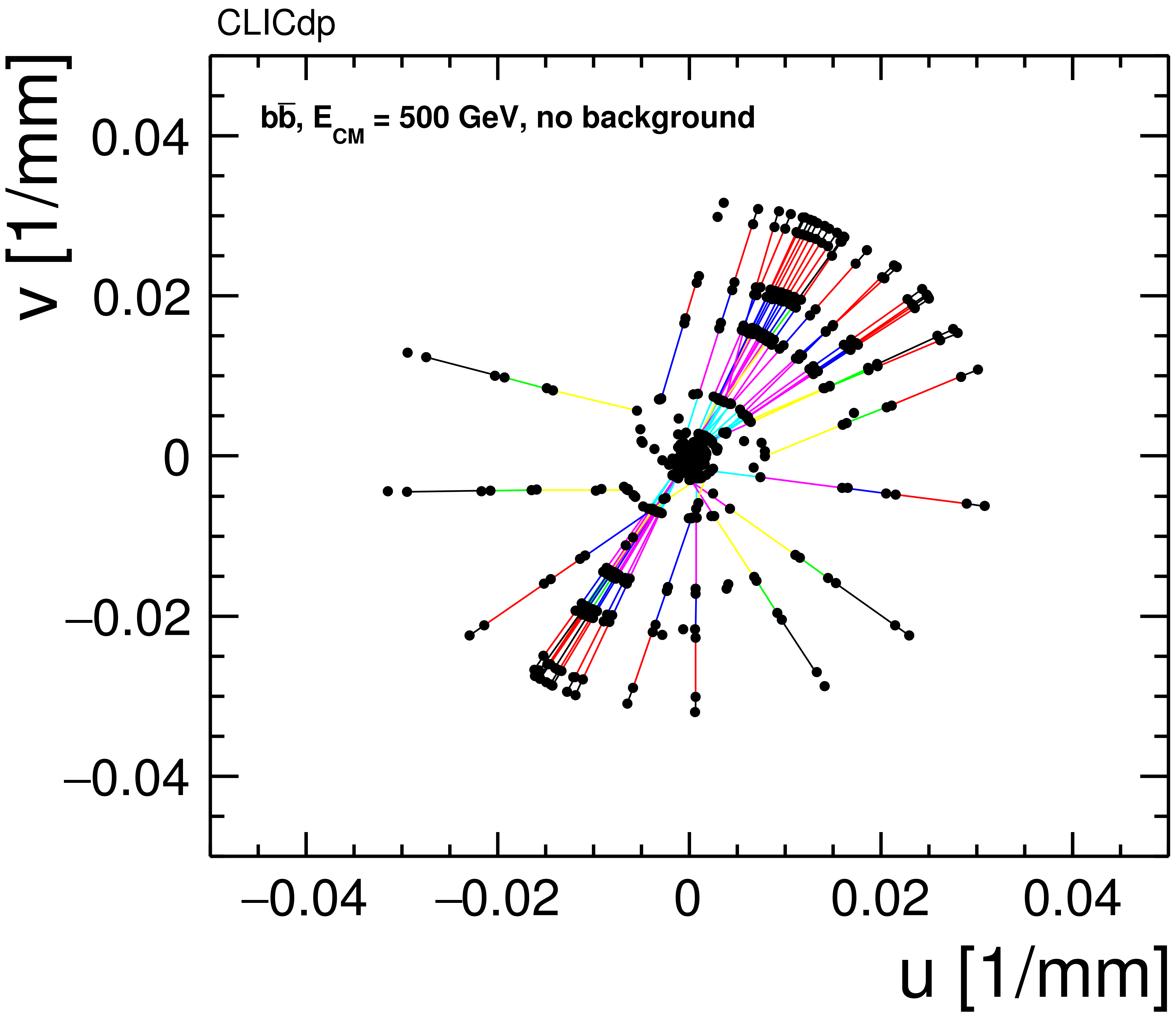
1 final step to reconstruct displaced tracks:

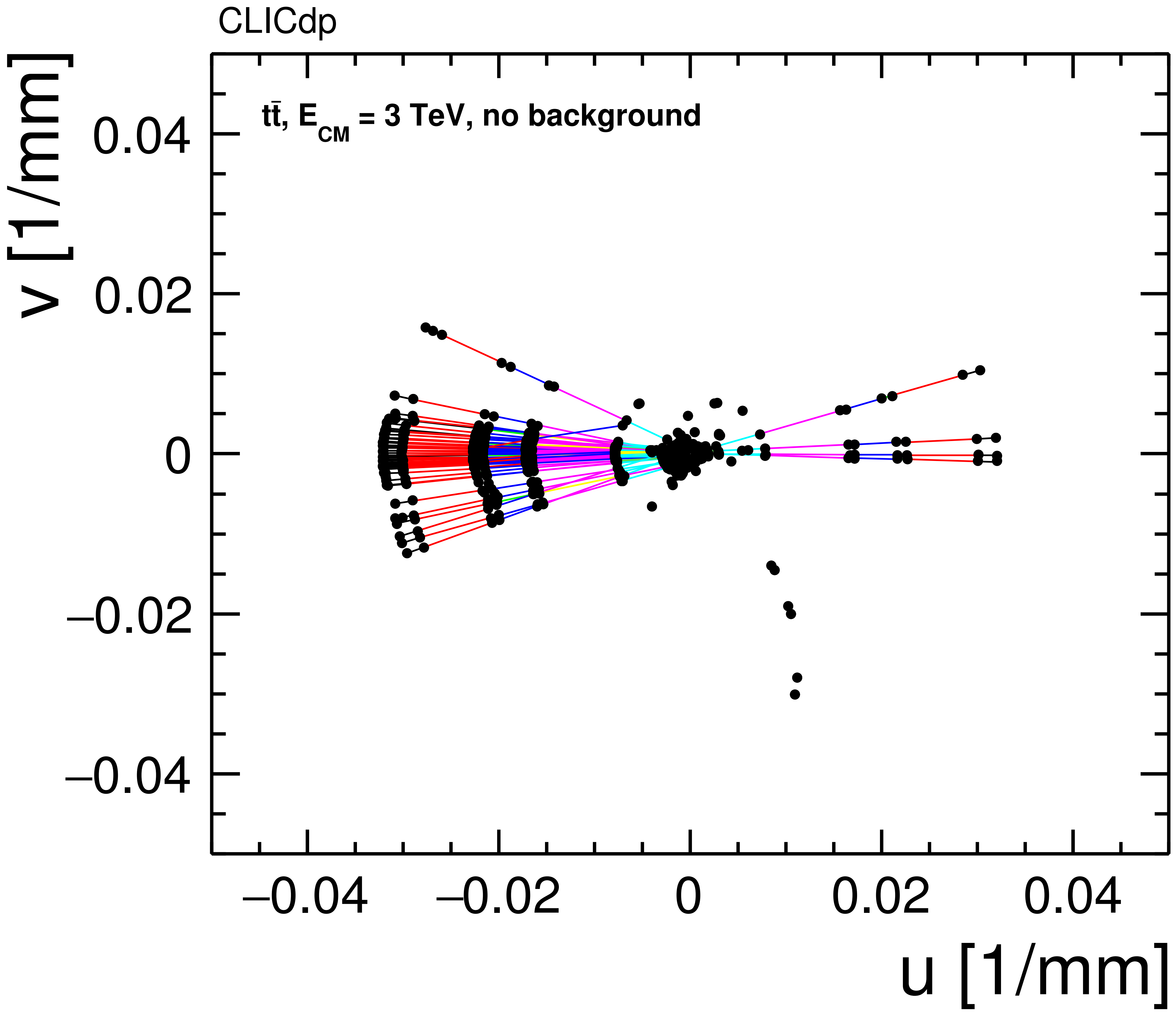
- quadratic terms in the CT fit added
- inverted order search: from silicon tracker to vertex detector
- broader search angle than for prompt tracks



Full Pattern Recognition Chain in CLICDet

Two example events



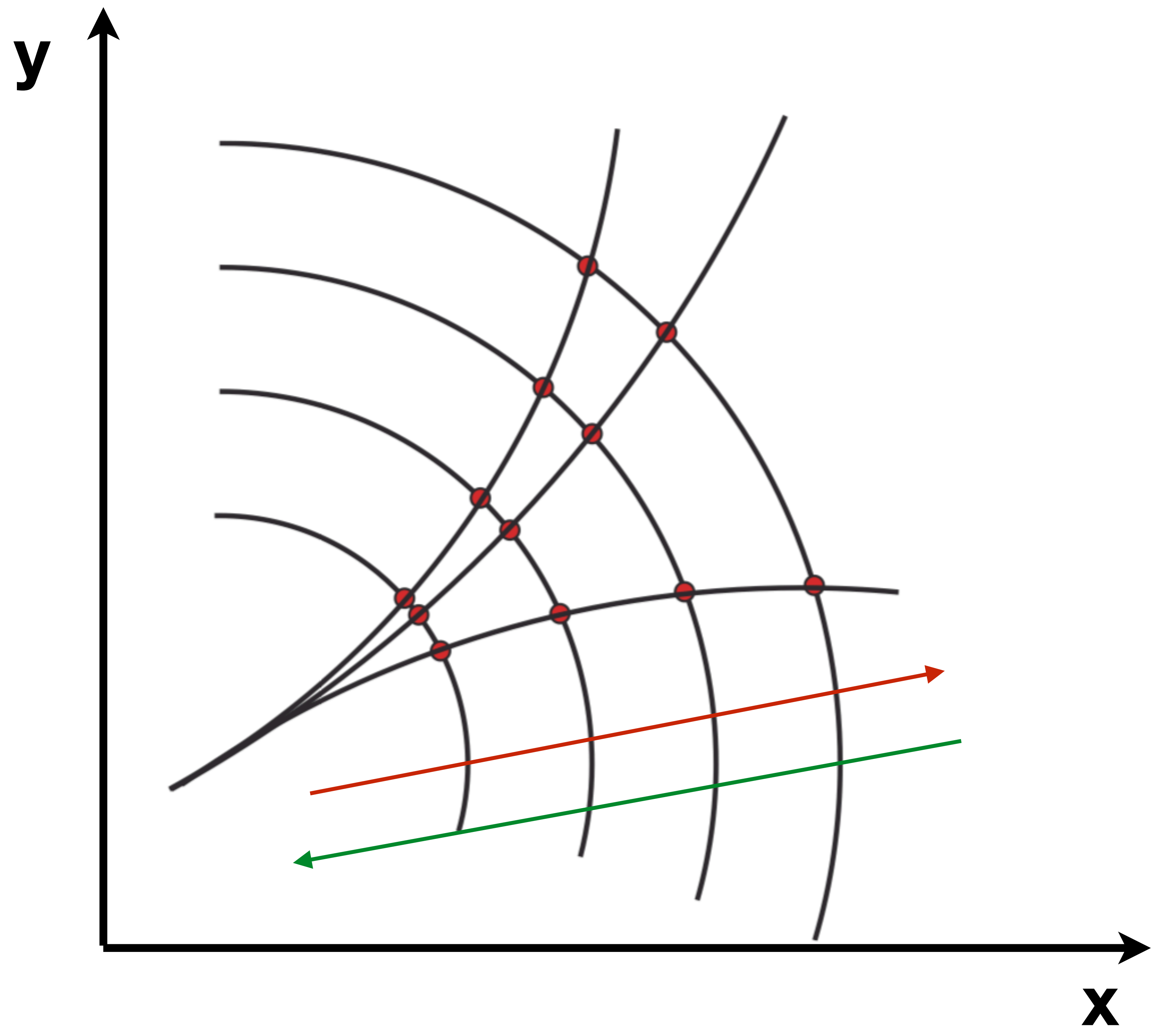


Track Fitting, Selection and CPU Performance

The final steps

Track fit: Kalman filter (KF) and smoother in global coordinate space

- Pre-fit: helix fit with three hits (first, middle, last) to initialize KF
- Kalman filter from innermost hits outwards
  - Hits added one by one, acceptance / rejection based on χ^2
  - If forward KF fit fails, fit is tried from outside in



Track selection:

- minimum number of hits = 4
- treatment of clones repeated once more as very last step

CPU time:

- 3 TeV ttbar takes ~50 longer with background than w/o
- most time consuming: Kalman filter w/o background, Cellular Automaton with background
  most time consuming in CA: displaced track reconstruction -
  50% of total CA time w/o background, 75% with background

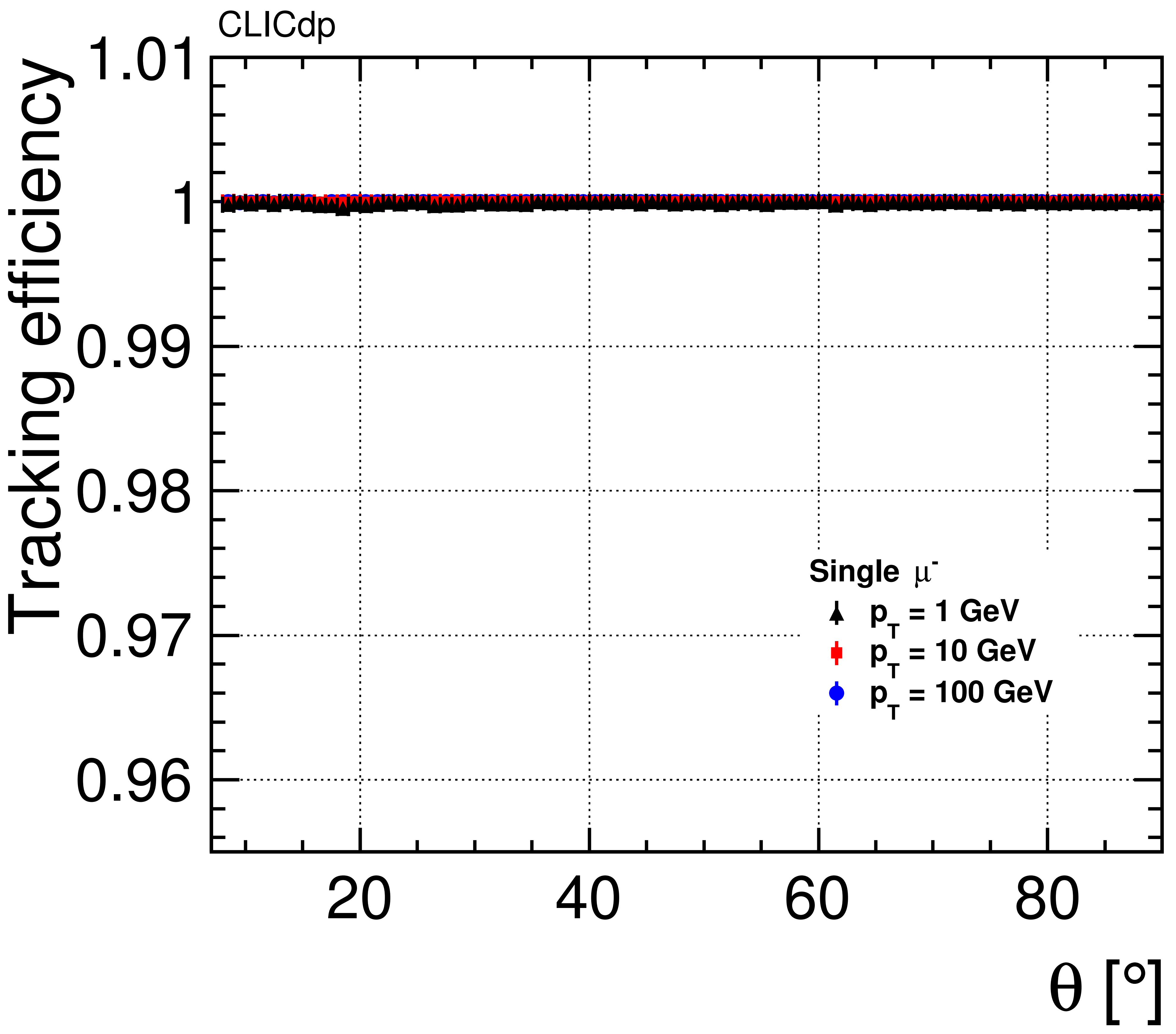
Performance Results

- Definitions for performance evaluation:
  - *Associated particle*: Simulated MC particle from which the majority of the hits on a track originate
  - *Reconstructable particles*: (Quasi-)stable particles with $p_T > 0.1$ GeV, $|\cos\theta| < 0.99$, unique hits ≥ 4
  - *Purity*: Fraction of track hits associated to the same particle
    - *pure track* for purity ≥ 75%, *fake track* for purity < 75%
  - *Efficiency*: $\dfrac{\text{\# MC particles associated to at least one pure track}}{\text{\# reconstructable MC particles}}$
  - *Fake rate*: $\dfrac{\text{\# fake tracks}}{\text{\# reconstructed tracks}}$

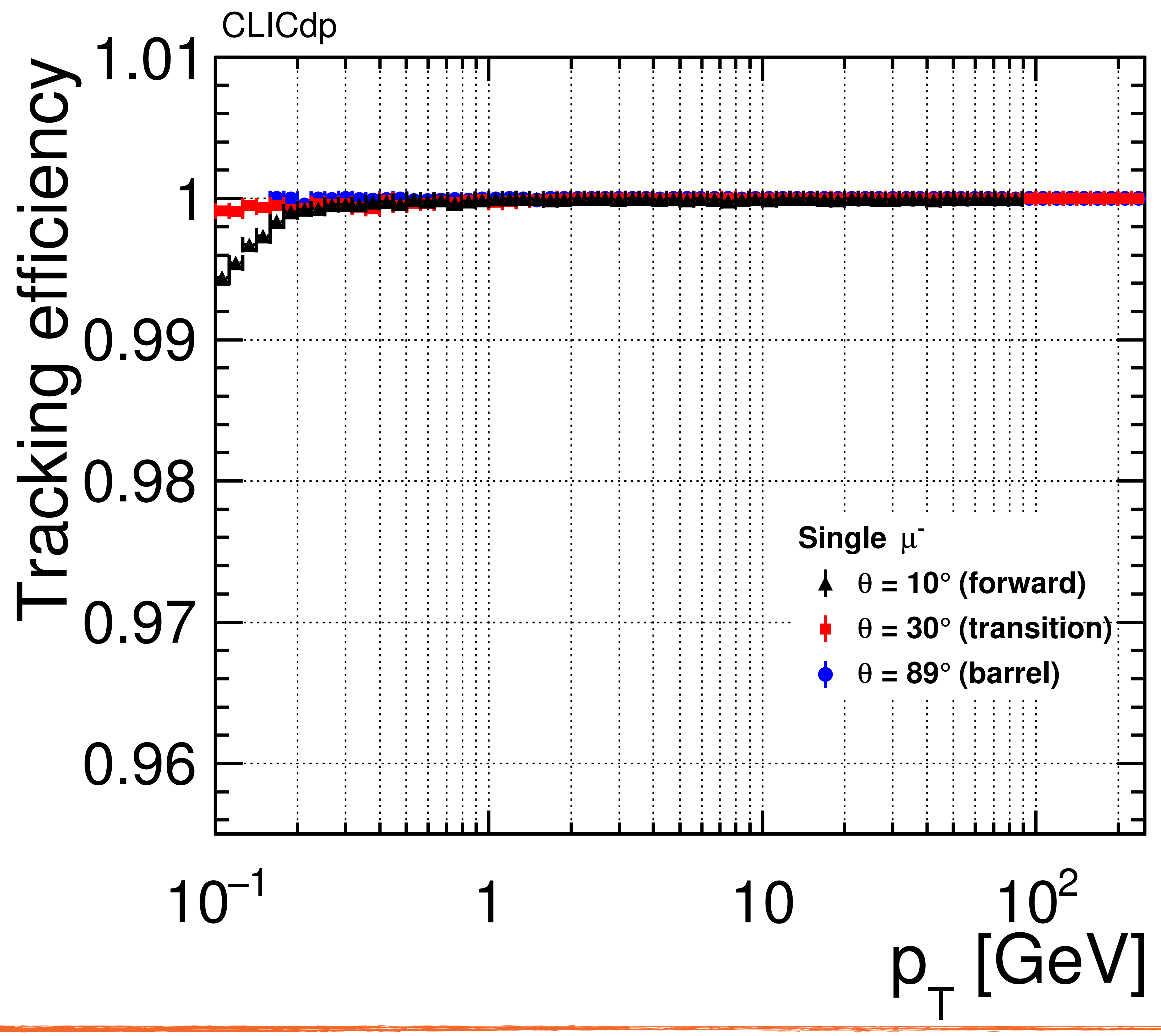
Note: only MC tracks with a minimum angular isolation of 0.02 rad considered in efficiency calculations

Performance for Prompt Isolated Muons

The easy case



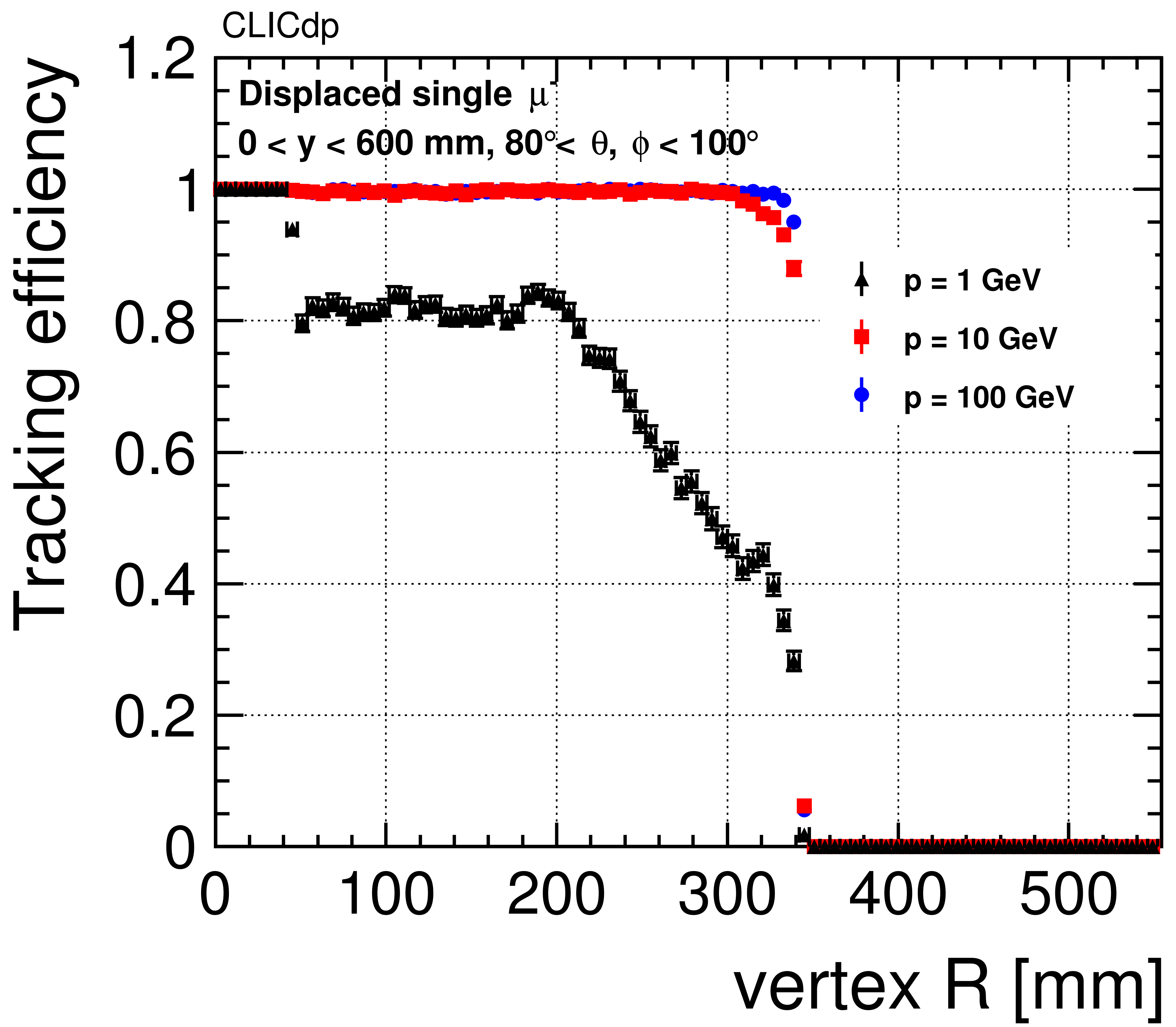
- Tracking fully efficient over full acceptance for high p_T tracks

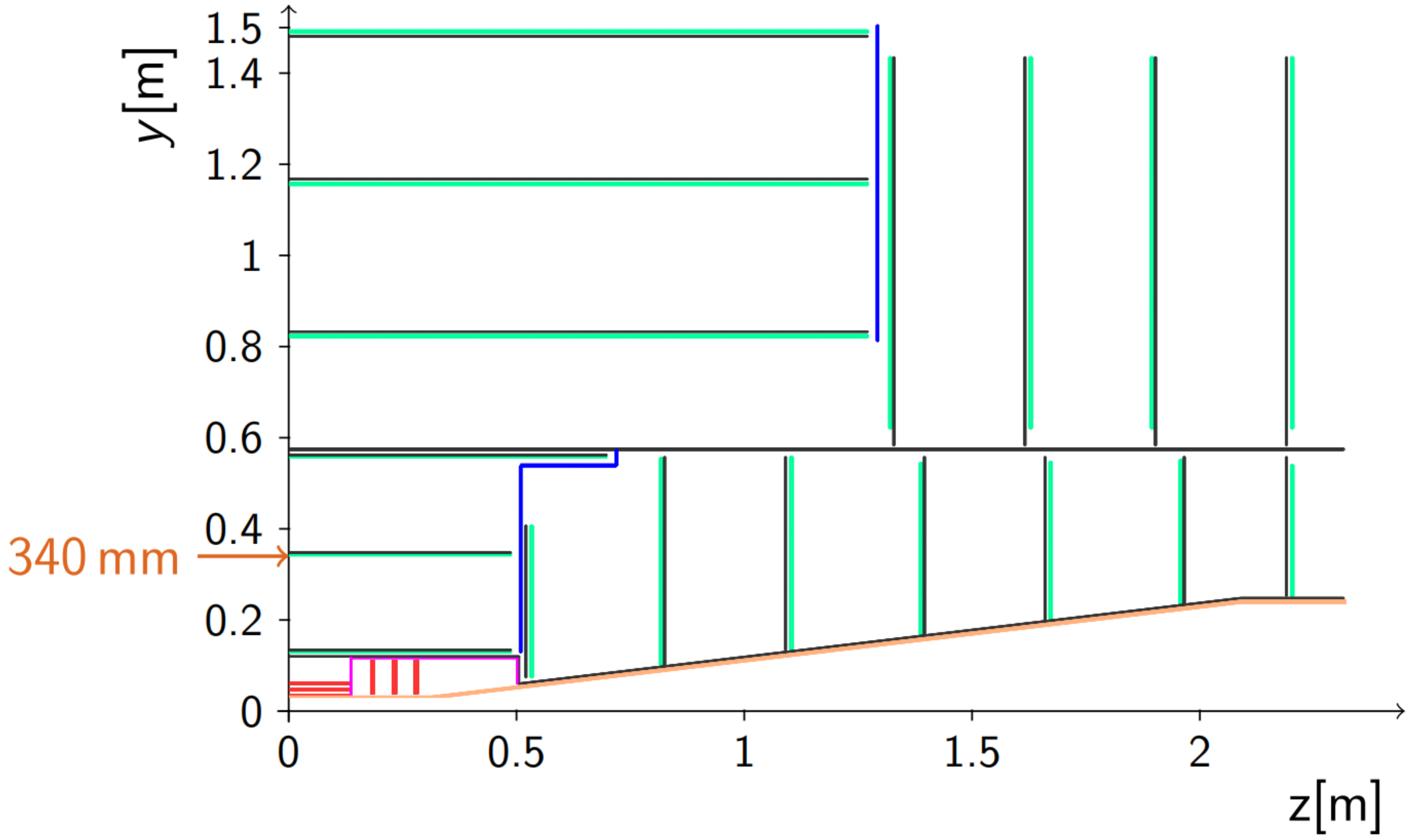


- Full efficiency everywhere from ~ 0.2 GeV p_T

Performance for Displaced Isolated Muons

Adding (some) complexity

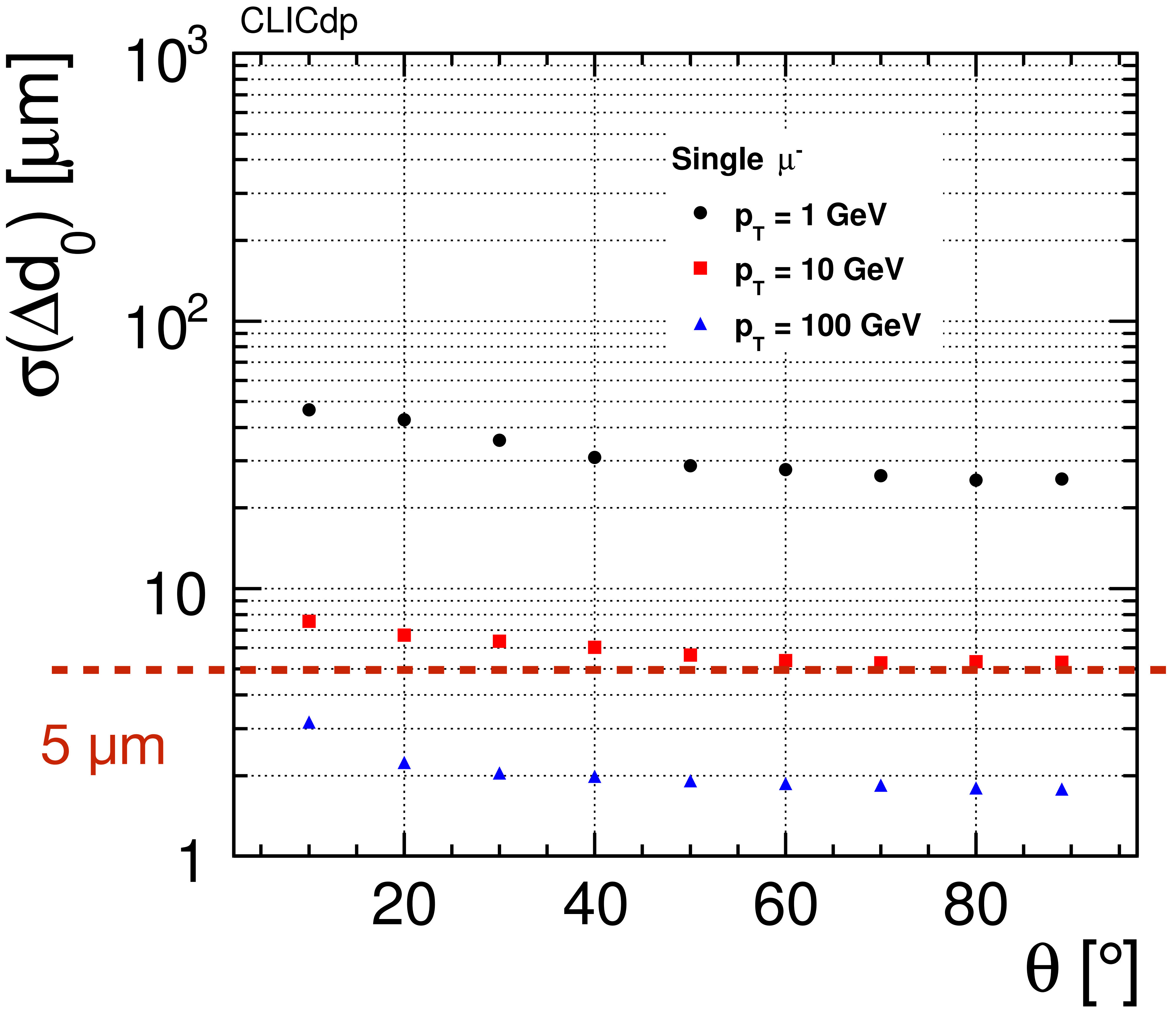


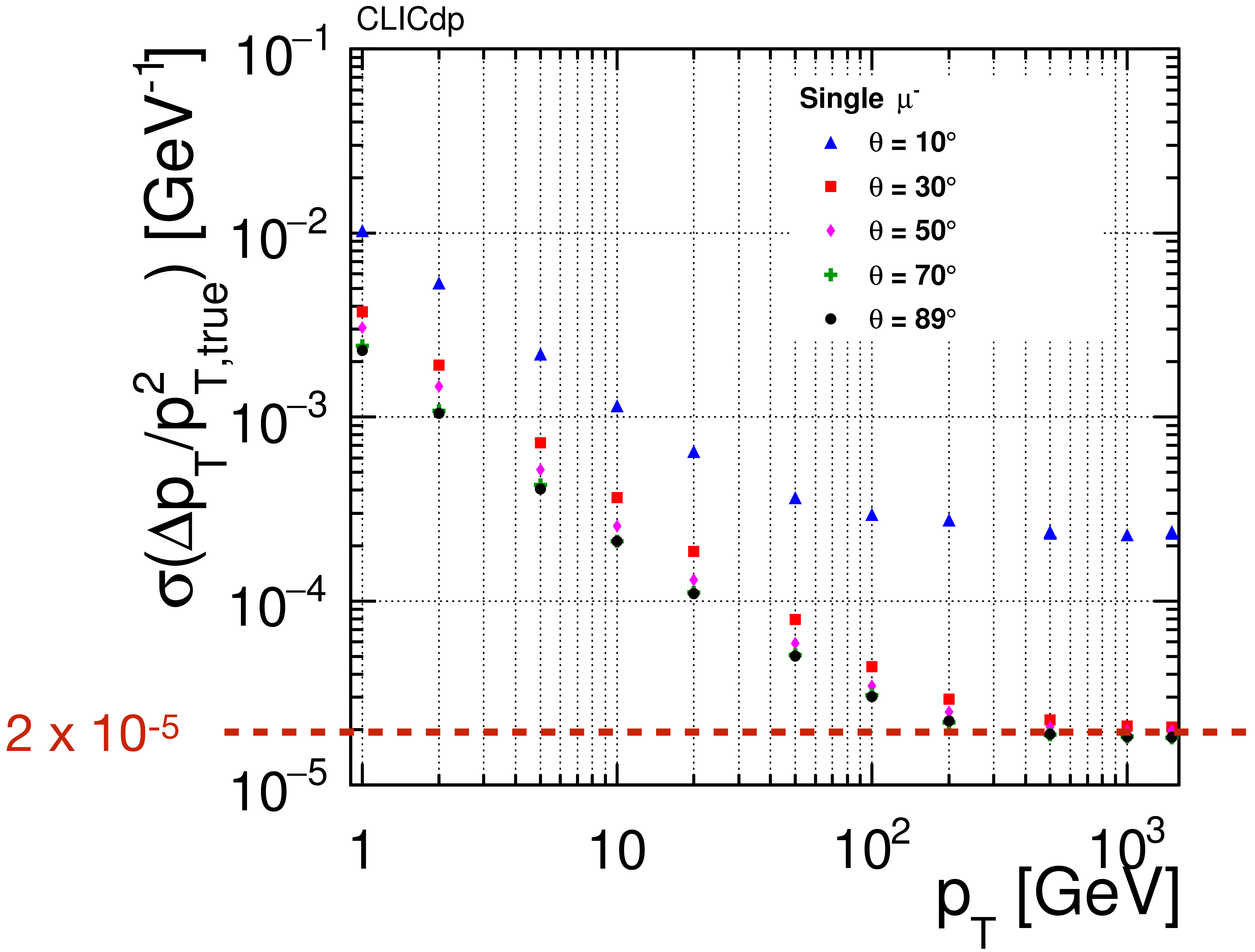


- For lower p_T, tracking remains efficient up to ~ 40 mm: 2nd VTX double layer
- At high p_T, tracking remains efficient up to ~ 340 mm: 2nd main tracker layer

Momentum and Vertex Resolution for Isolated Muons

Achieving physics-driven performance goals

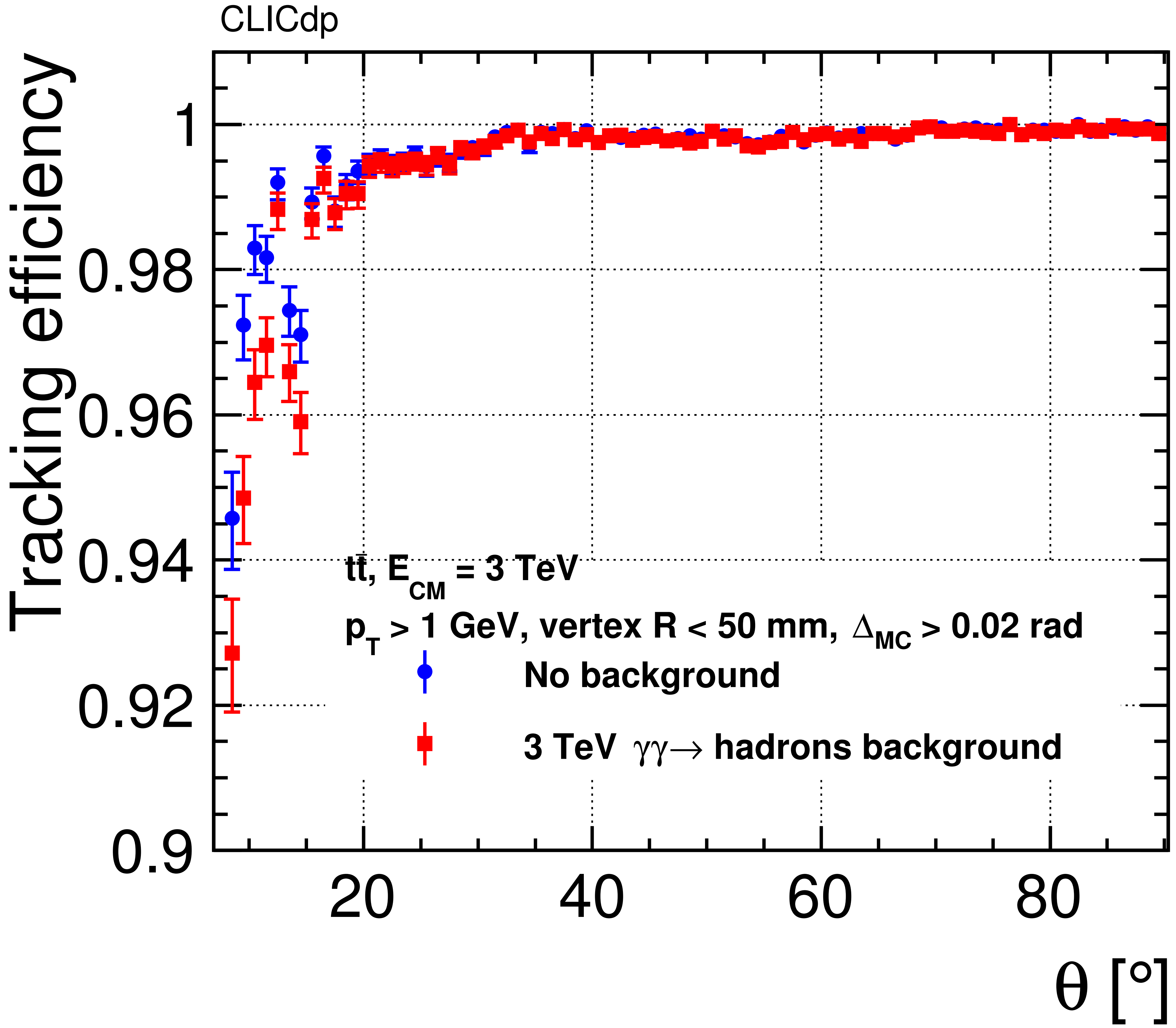


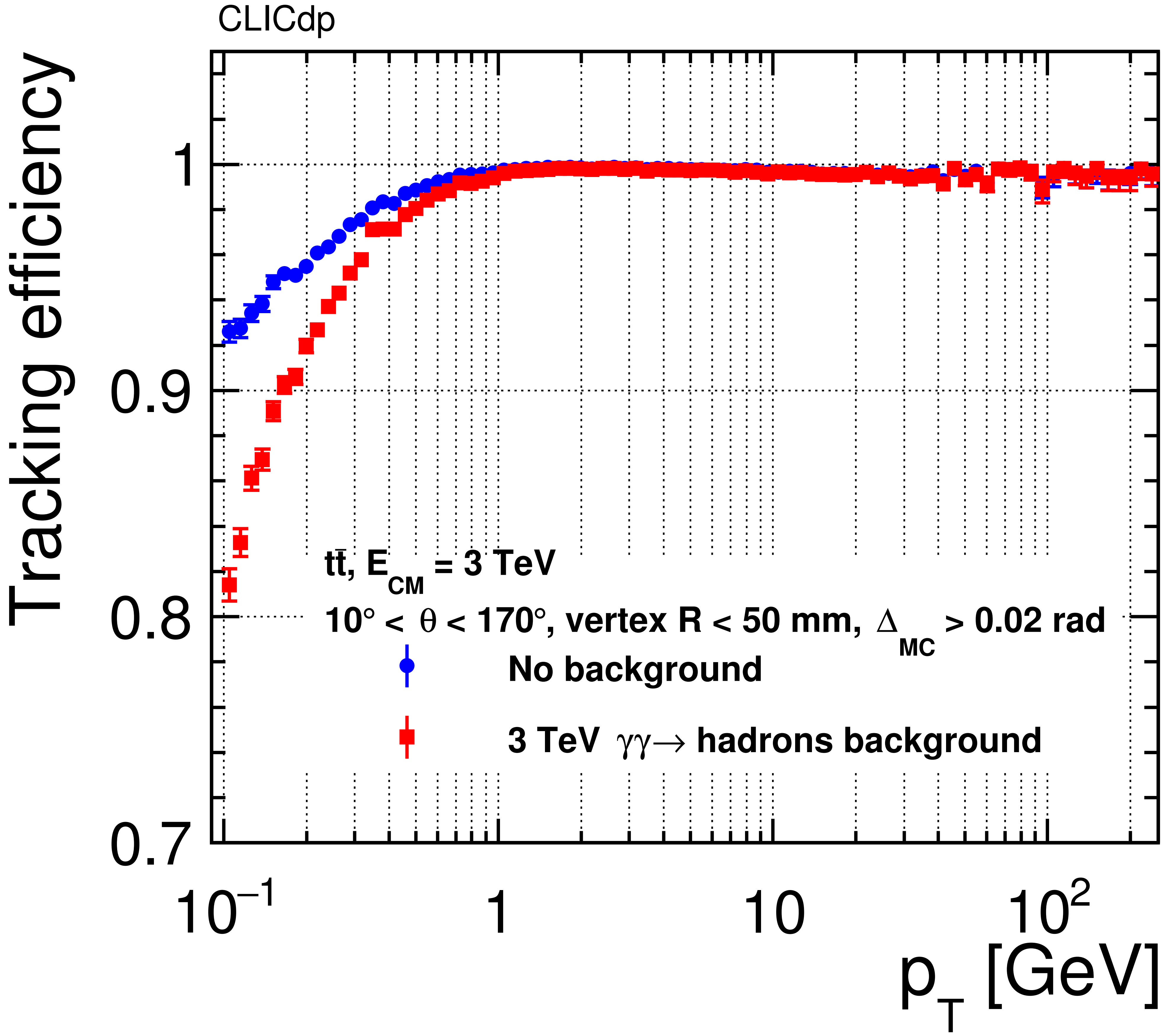


Performance for ttbar Events at 3 TeV

High degree of event complexity

- Global Efficiency:



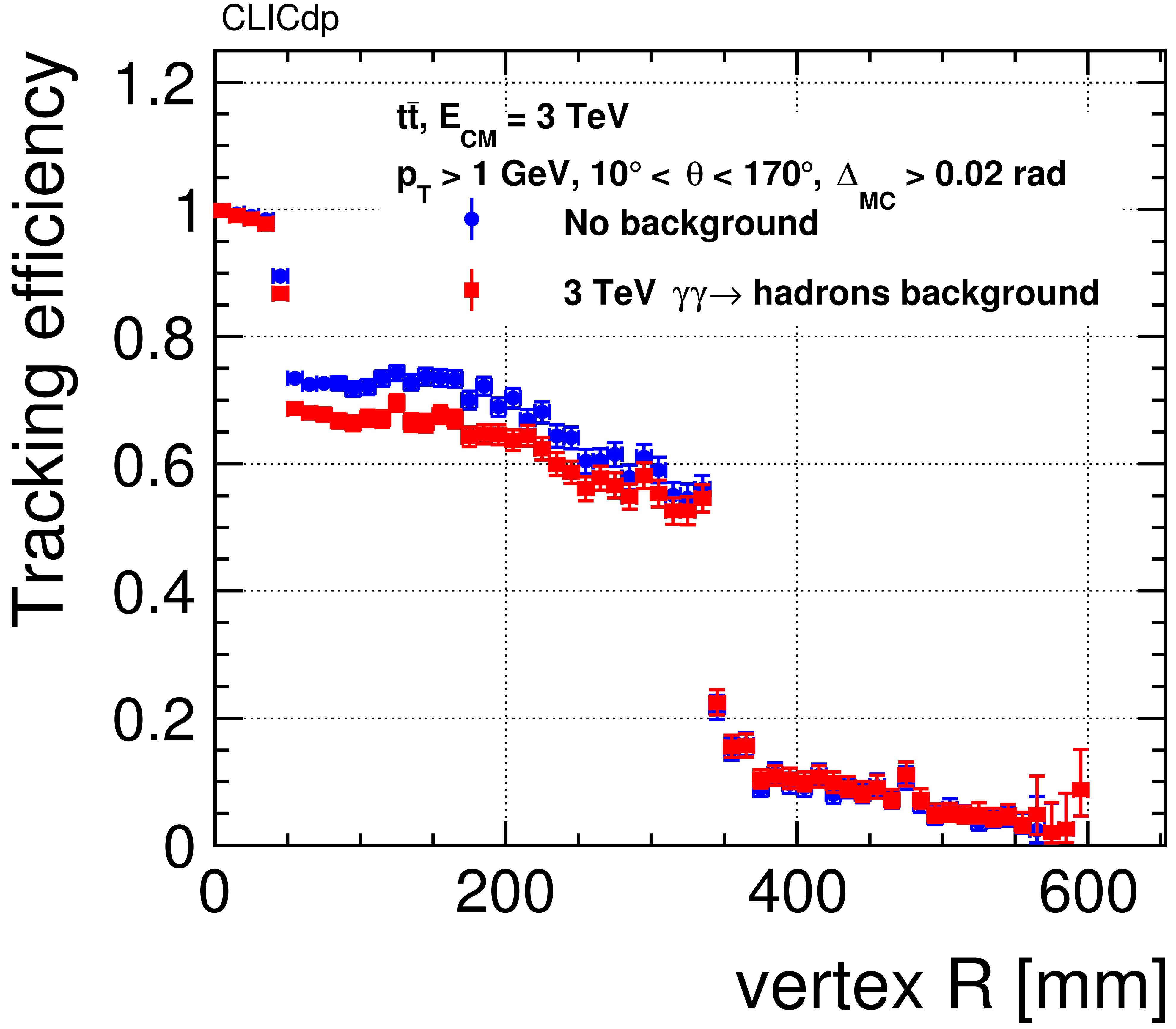


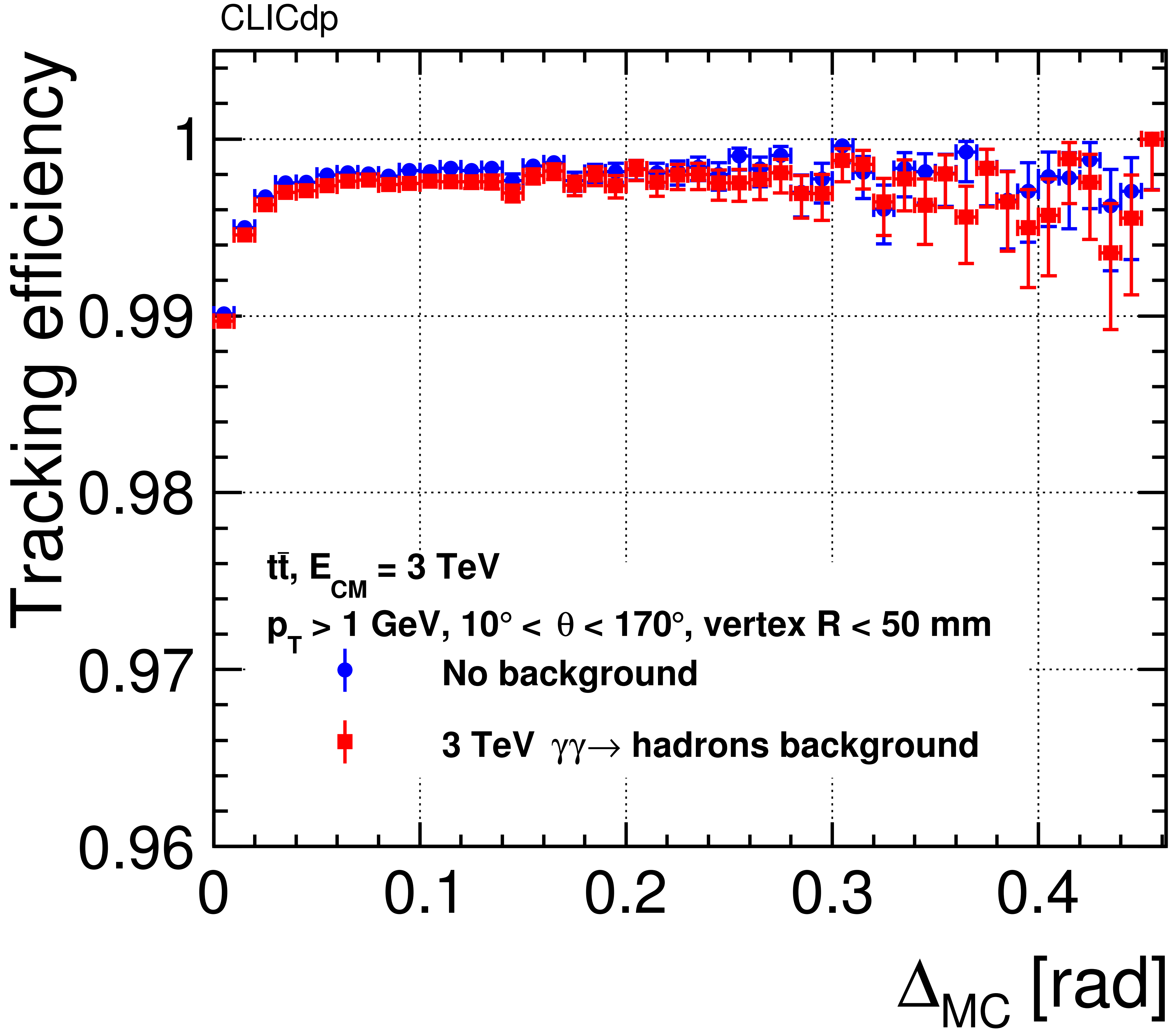
- High efficiency in barrel region and above 1 GeV p_T: plateau region insensitive to background
- Background impact mostly far forward and at low p_T

Performance for ttbar Events at 3 TeV

High degree of event complexity

- Efficiency for displaced tracks and high density:



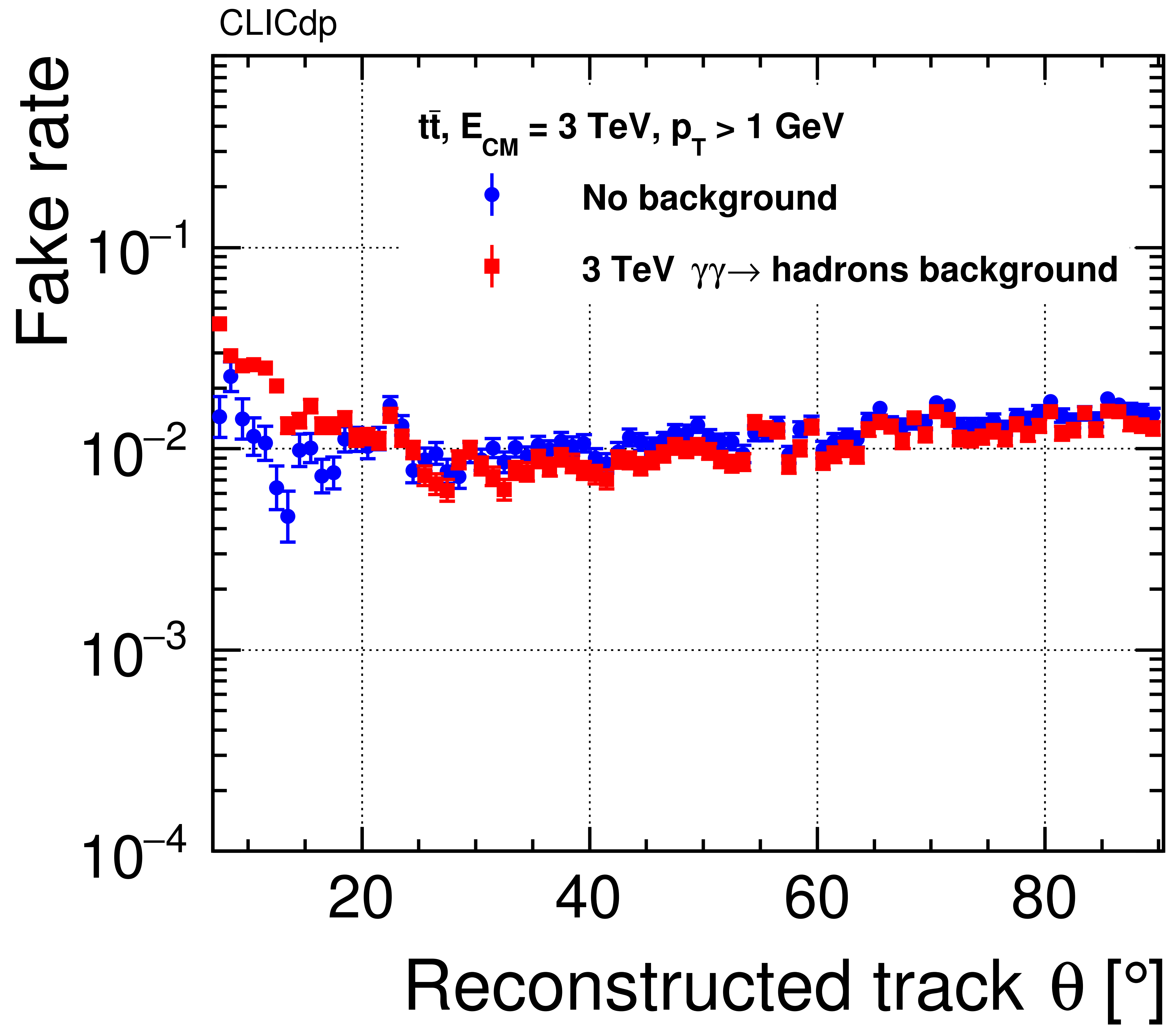


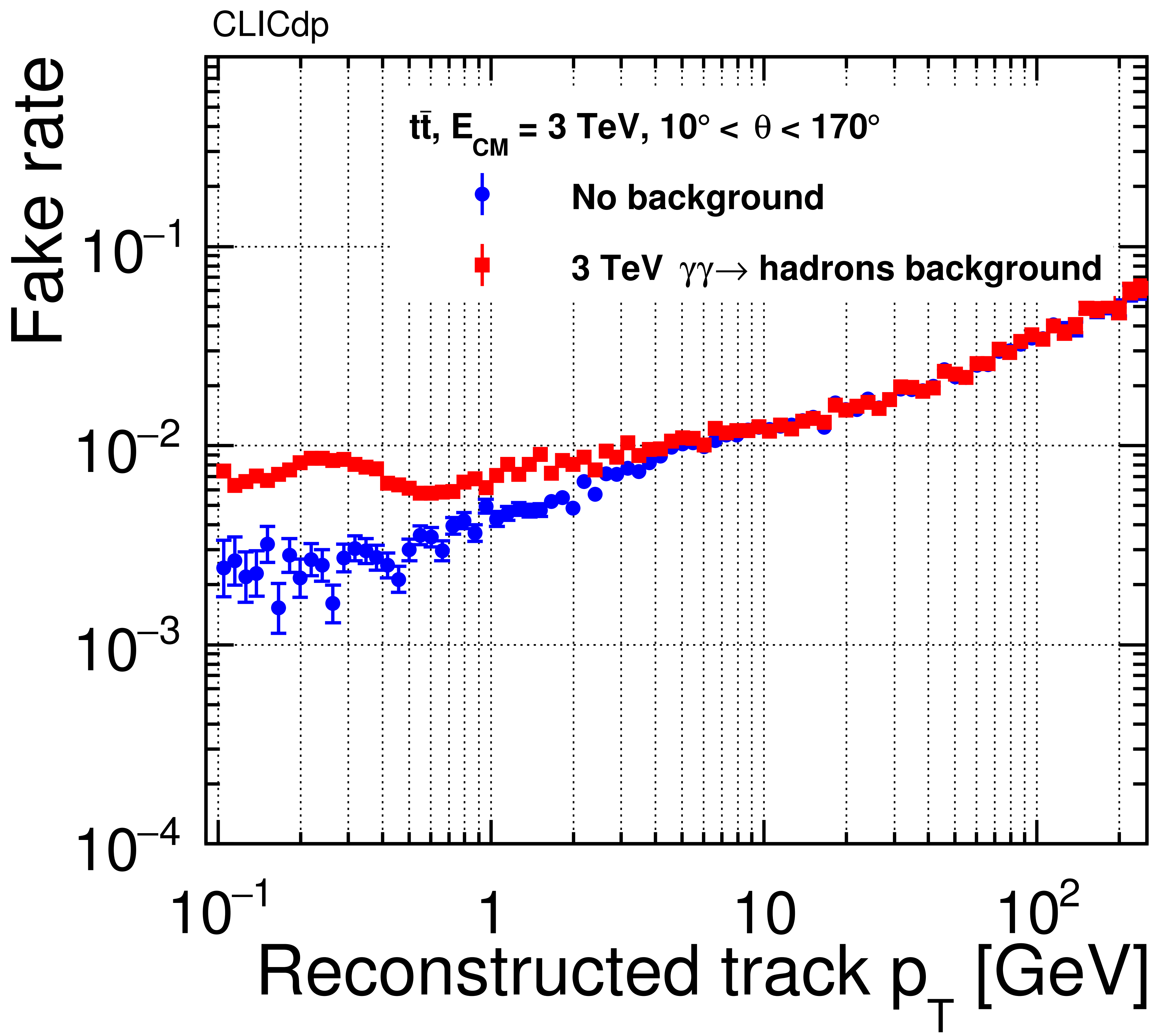
- Efficiency for vertices outside of VTX reduces - as seen for low energy muons
- Slight efficiency drop for very close-by tracks

Performance for ttbar Events at 3 TeV

High degree of event complexity

- Fake rates





- Background results in an increase in fake rates at very forward angles and at low p_T, in line with observed efficiency drop in same regions

Conclusions

- Tracking at CLIC is an interesting challenge:
  - High precision required by ambitious physics goals
  - Significant beam-induced backgrounds and multi-TeV final states result in highly complex events

- The CLIC tracker is optimised for these conditions:
  Timing capabilities on the sub - 10 ns level, very low mass to limit scattering, enable resolution goals

- Low-mass trackers allow the use of novel tracking algorithms:
  Conformal mapping + cellular automaton + Kalman filter
  - Provides robust pattern recognition in complex events with background
  - Performs well for tracks from displaced vertices
  - High efficiency, low fake rates, meets CLIC requirements

  ➫ Developed and tested for CLICdet, but inherently flexible and geometry-independent

More details: Paper recently submitted to NIM A: arXiv:1908.00256