Increasing track reconstruction efficiency in dense environments at ATLAS

On behalf of the **ATLAS collaboration**

CTD Valencia

02-05 April 2019

Sofia ADORNI BRACCESI CHIASSI

Outline

- Introduction
- Tracking in ATLAS ➜ Focus on dense environments
- Tracking for high p_T B-hadrons:
  - Defining a region of interest
  - Tackling the efficiency loss
  - Improving track quality
- Tracking for high p_T τs:
  - Fixing merged tracks
- Conclusion

Introduction

The **long lifetime** of certain particles (B hadrons, τs, etc…) leads them to having characteristic properties that allow us to identify them:

- Large positive impact parameter
- Displaced secondary vertex
- Weak decay chain (PV ➜ b ➜ c)

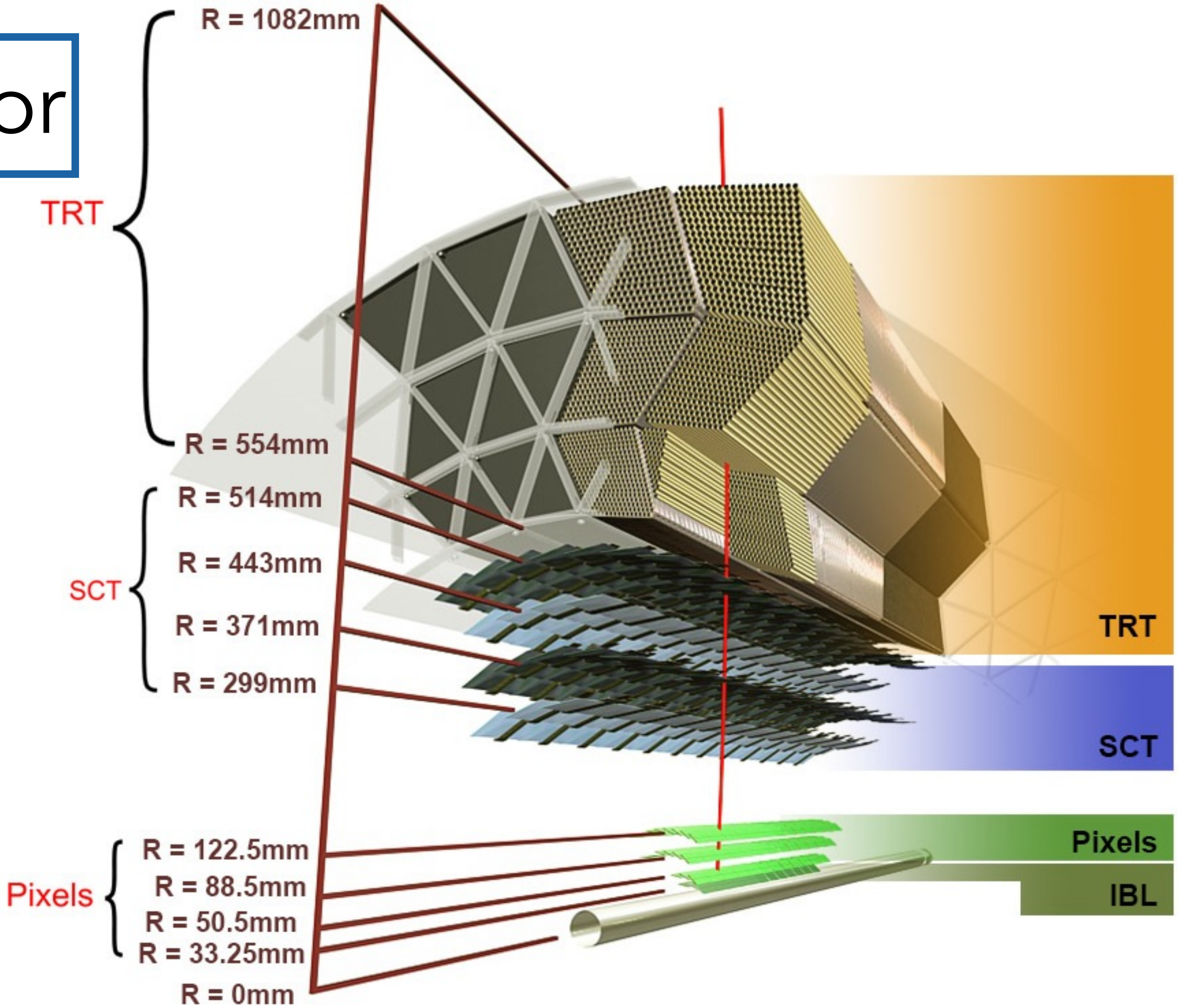
} ➡ **TRACKING IS ESSENTIAL**

The **identification** of jets originating from **B-hadrons** (B-tagging) is crucial for many interesting physics signature at the Large Hadron Collider (LHC):

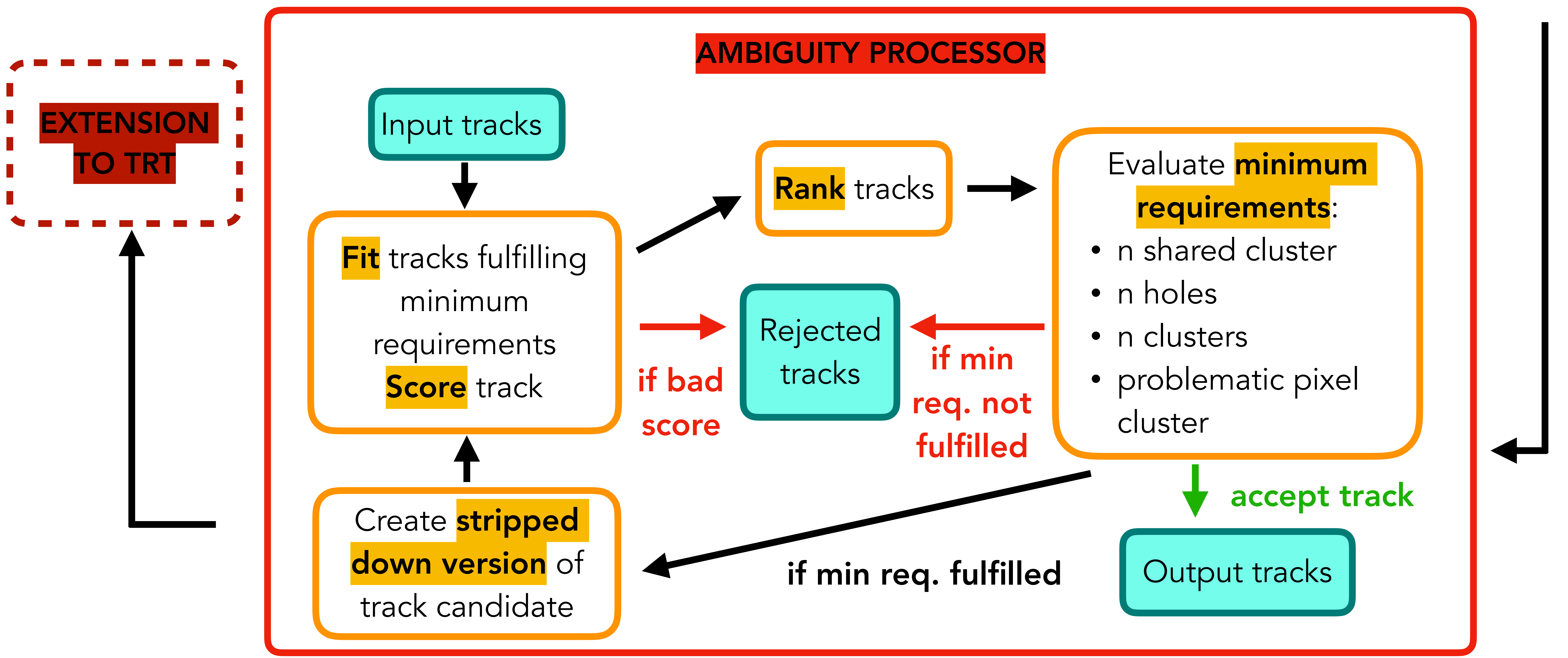
- Top quarks decay into W bosons and b-quarks about 100% of the time
- The Standard Model Higgs boson predominantly decays into b-anti b-quark pairs
- Many searches for new physics, e.g. supersymmetry, involve final states with b-quarks

ATLAS Inner Detector

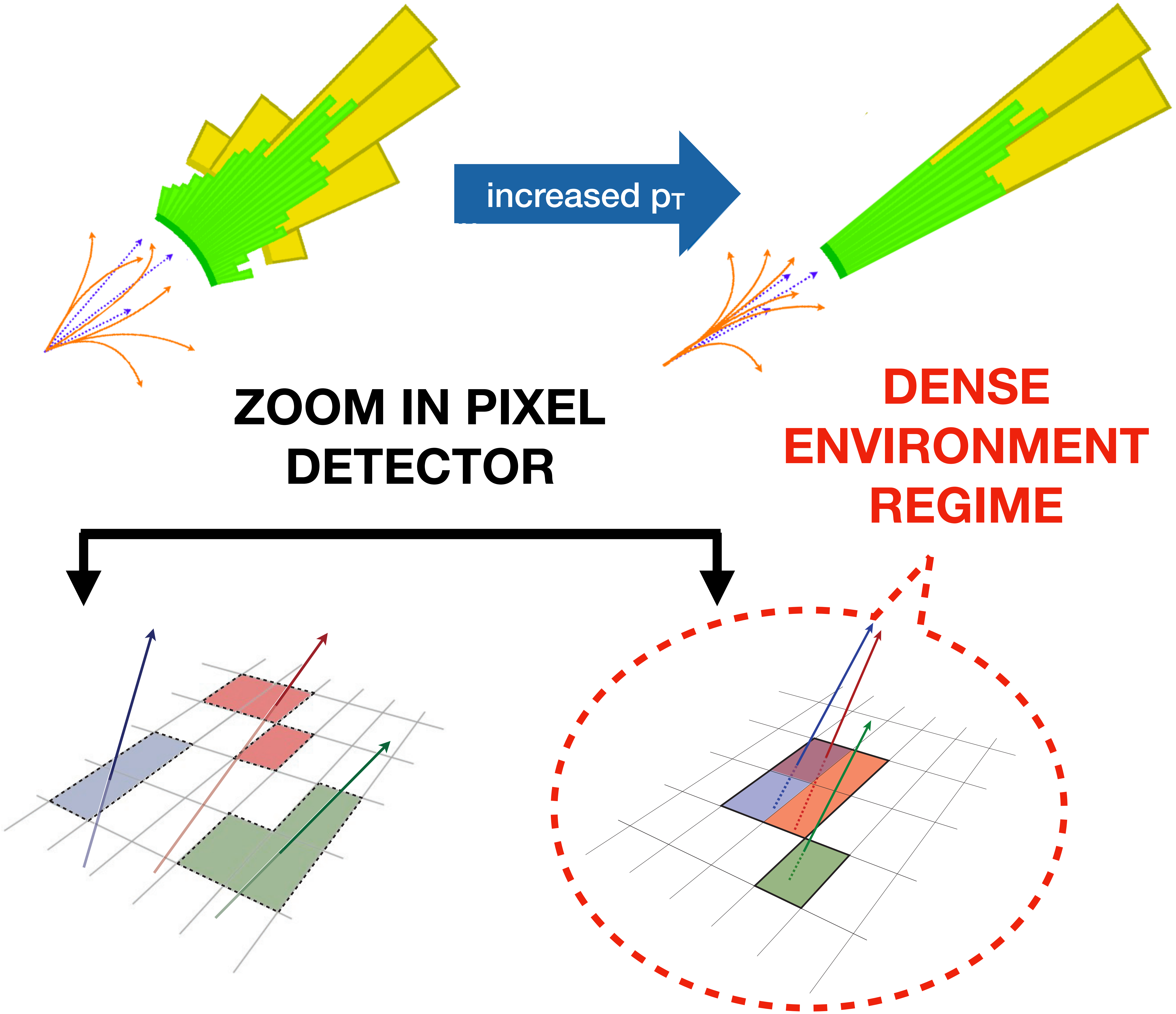
- The Inner Detector (ID) measures the **trajectories** of charged particles originating from the collision point
- **Three detector technologies**:
  - Silicon pixels (in order):
    ➜ IBL
    ➜ B Layer (BL)
    ➜ Pix 2
    ➜ Pix 3
  - Silicon strips (SCT)
  - Transition radiation (TRT)



Tracking in ATLAS → PRE-PROCESSING → COMBINATORIAL TRACK FINDING

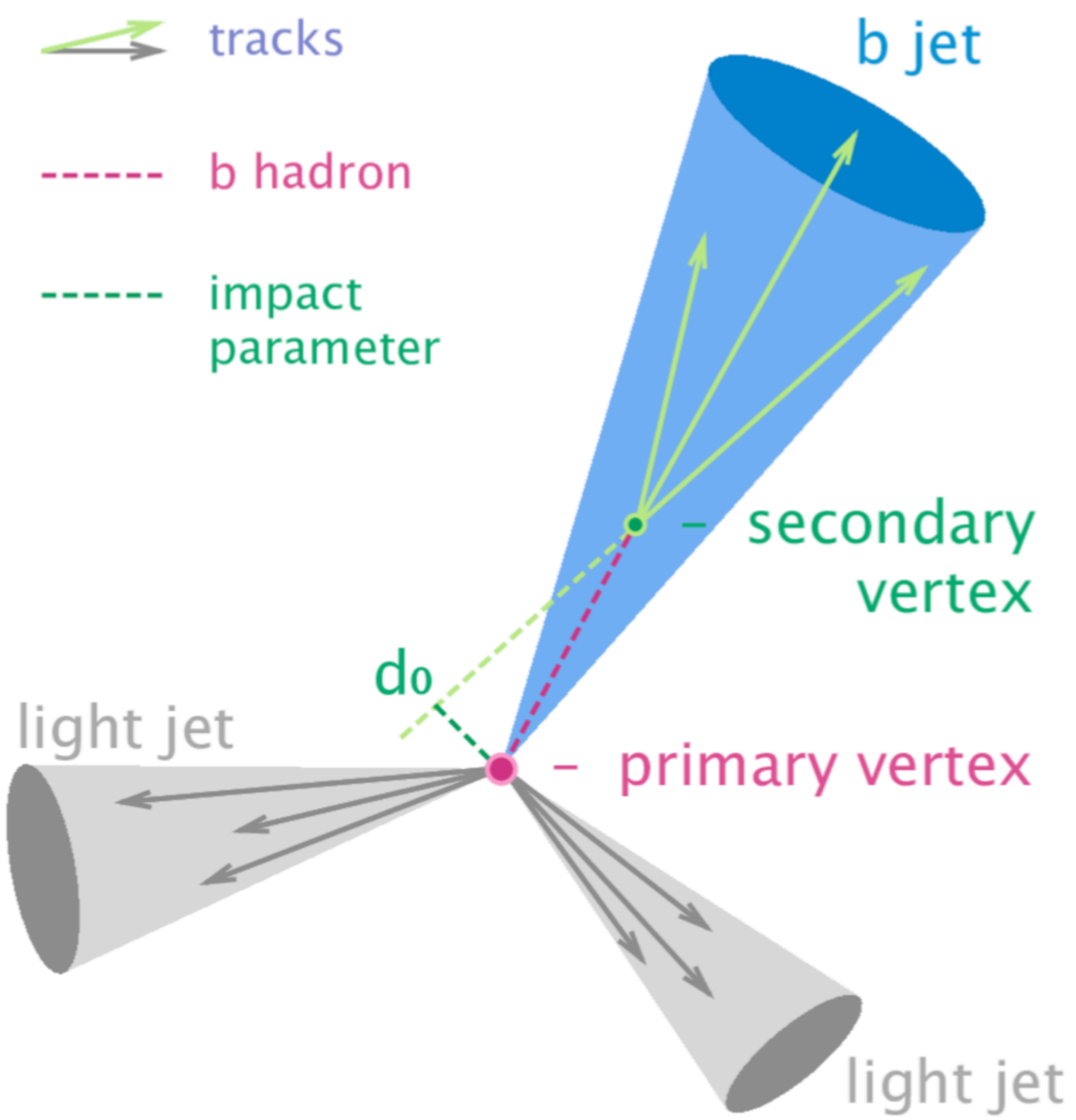


Tracking in Dense Environments : why are tracks rejected?



- In boosted environment
  ➜ separation between particles ~ cluster size
  ➜ reconstructed as merged cluster
  ➜ **shared clusters penalised** (reduce fakes/duplicates)
- ATLAS has a neural network (NN) to split shared cluster ➜ tackling remaining inefficiencies

Anatomy of a B jet



At high p_T Bs can be so **displaced** that they start decaying after the first layers of the ID

⟱

Our current tracking algorithm isn't optimised for this scenario

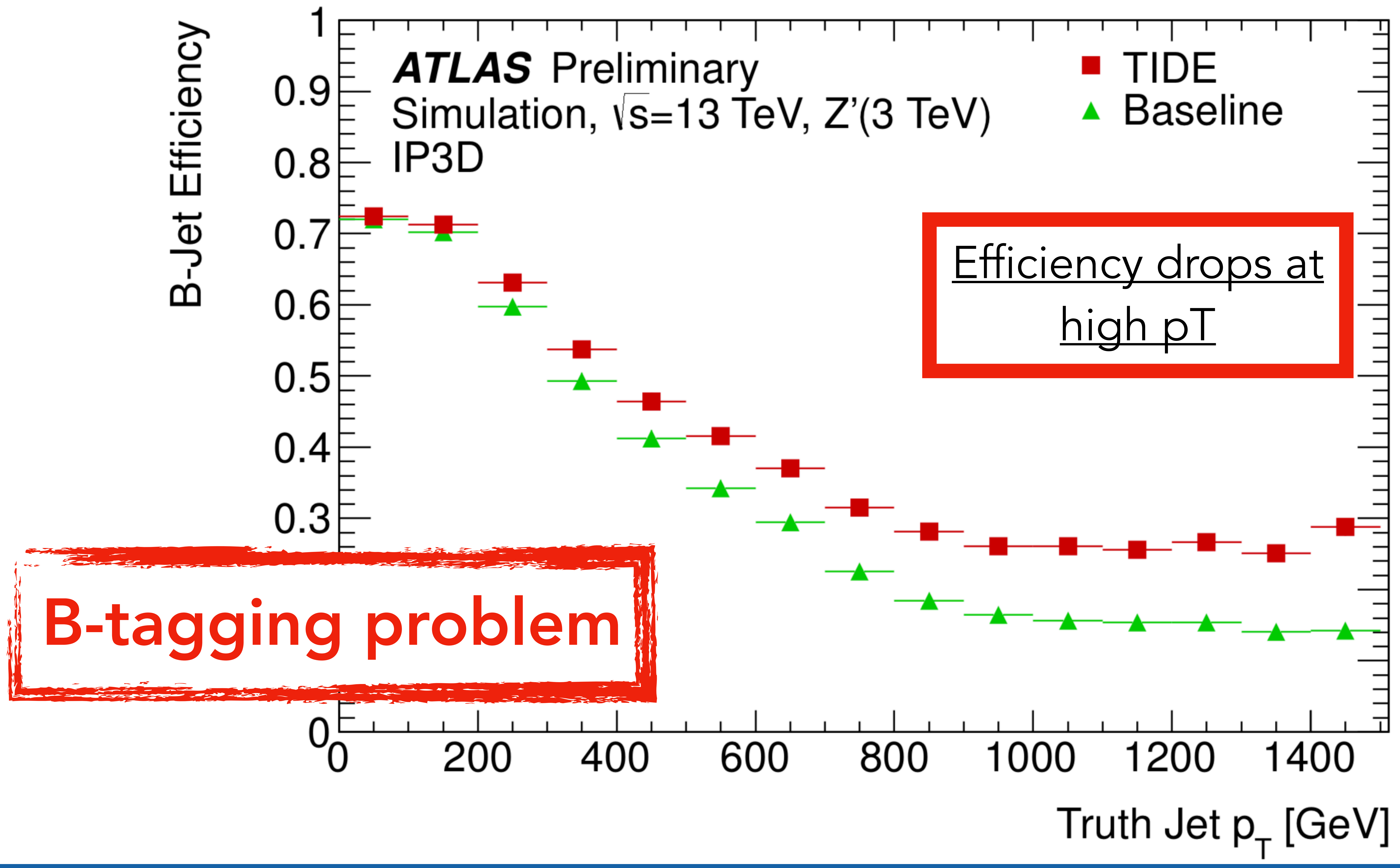
⟱

We **lose efficiency** in track reconstruction ➜ lose efficiency in B tagging

⟱

B-hadrons have ~4 tracks: whatever gain on per track efficiency results in ~4 fold gain in B tagging

Consequences for B-tagging at high p_T?



Tracking is a key part in this loss (2 main effects):

- **Loss of hits ➜** lower track reconstruction efficiency
- **Track quality ➜** degrades impact parameter measurement

Efficiency loss

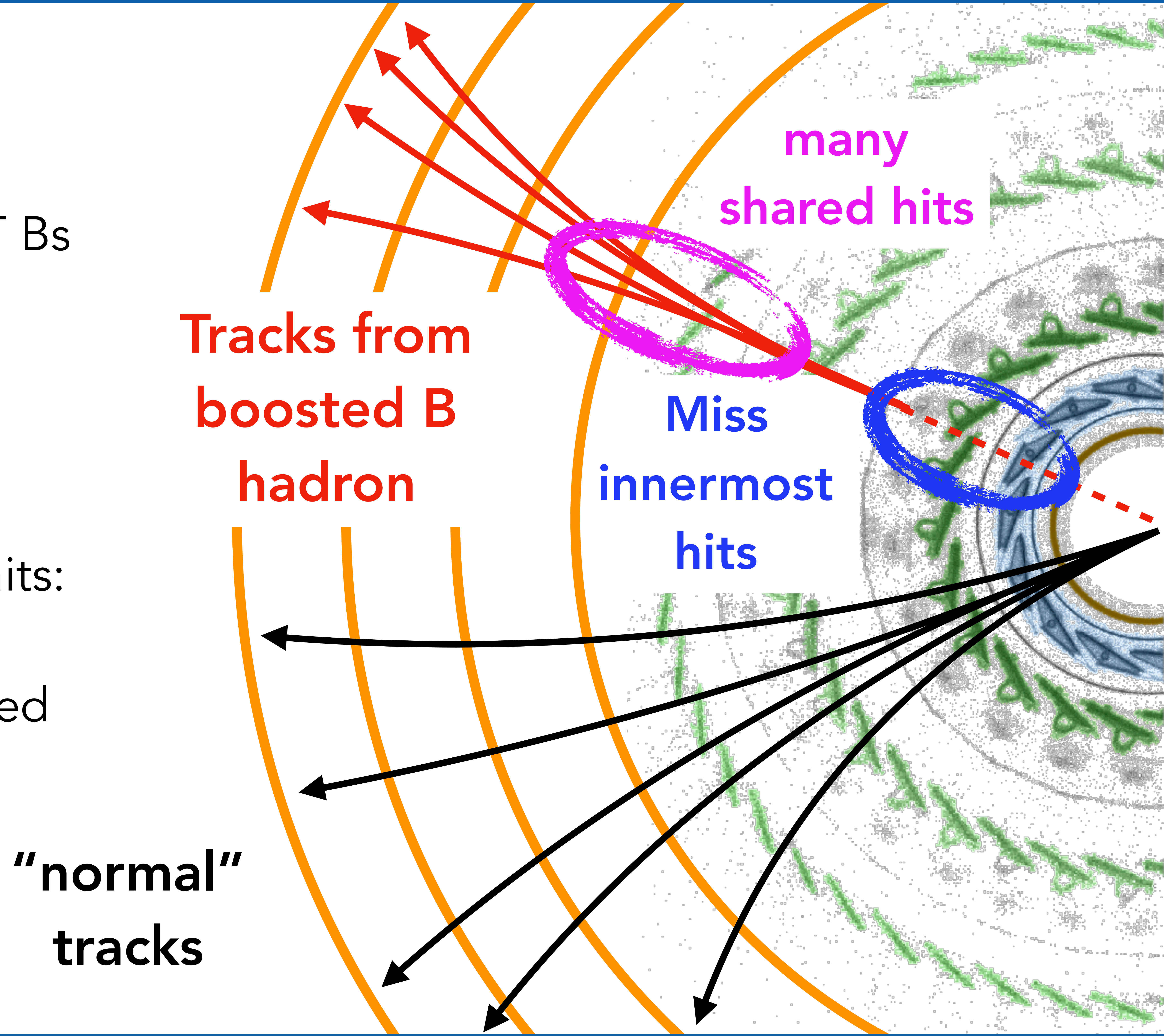
This **loss** for tracks coming from high pT Bs is the results of **tight cuts**

⟱

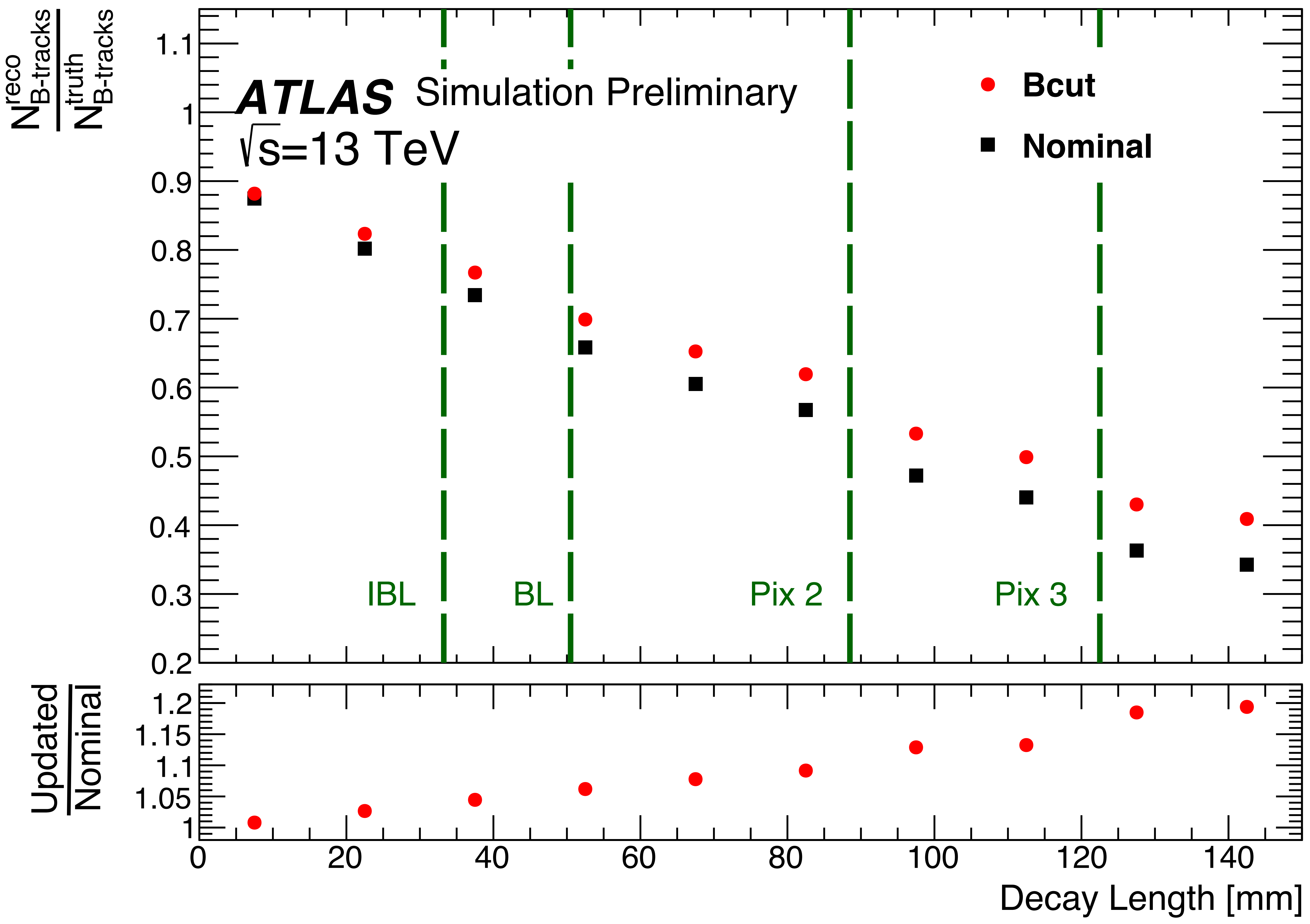
We **loosen** the following cuts

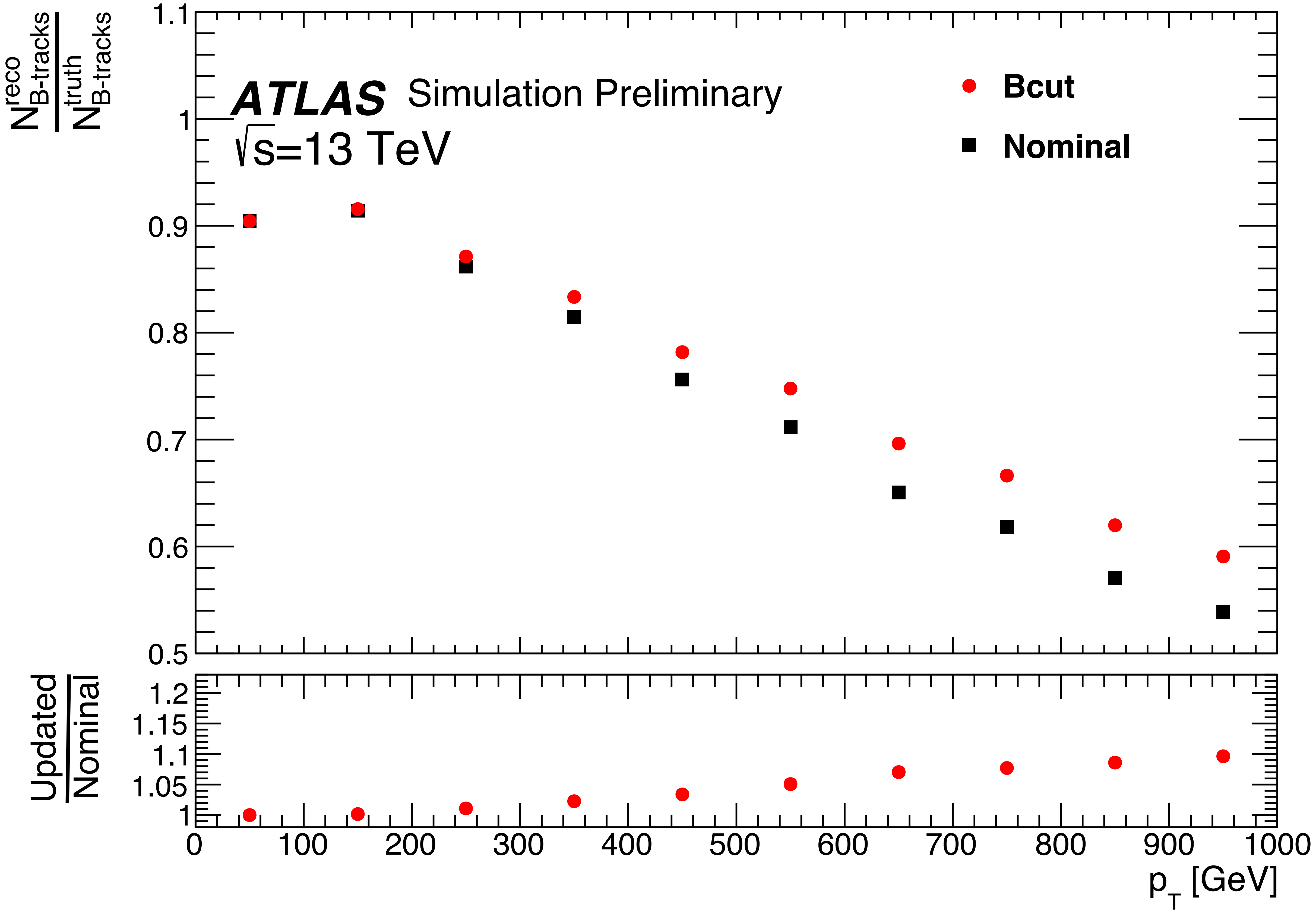
⟱

- nominal maximum number of shared hits: $N^{hits}_{shared} < 3$ ➜ $N^{hits}_{shared} < 7$
- nominal minimum number of non shared hits: $N^{hits}_{non-shared} > 5$ ➜ $N^{hits}_{non-shared} > 1$
- nominal minimum number of Si hits to allow splitting: $N^{hits}_{Si} > 8$ ➜ $N^{hits}_{Si} > 6$



Regaining efficiency





Recover efficiency

- ~15 % for B-hadrons decaying after Pix2 and ~10% after BL
- 4 tracks per B ➜ increase in B-tagging should be larger

Defining a Region Of Interest (ROI)

We want to improve our tracking only for tracks coming from B otherwise ➜ **time consuming** and **fake tracks**

IDEALLY

Want to apply changes in the tracking only to B tracks

IN REALITY

Can limit the changes to high energy calorimeter clusters corresponding to high pT jets

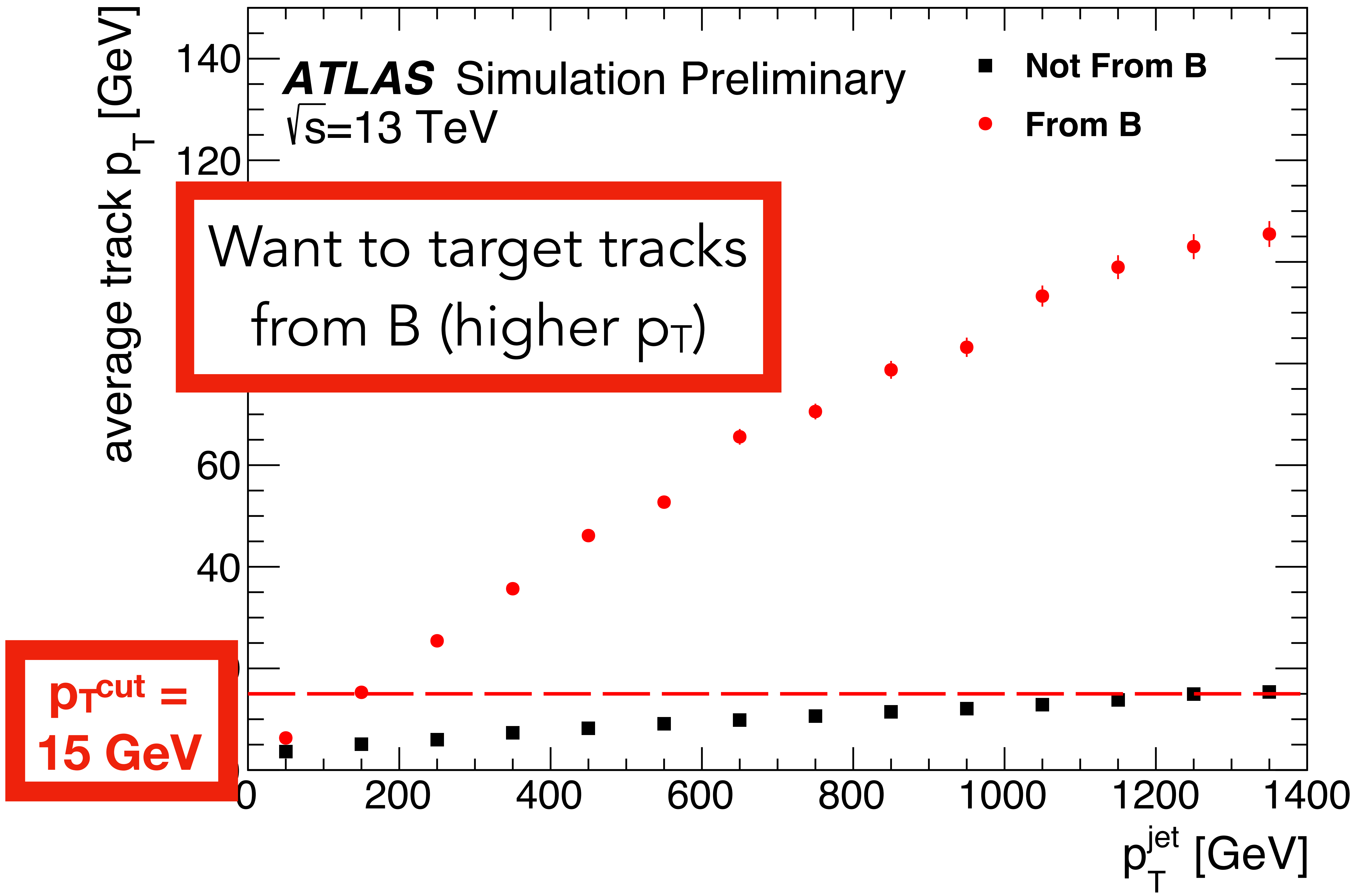
THREE PARAMETERS

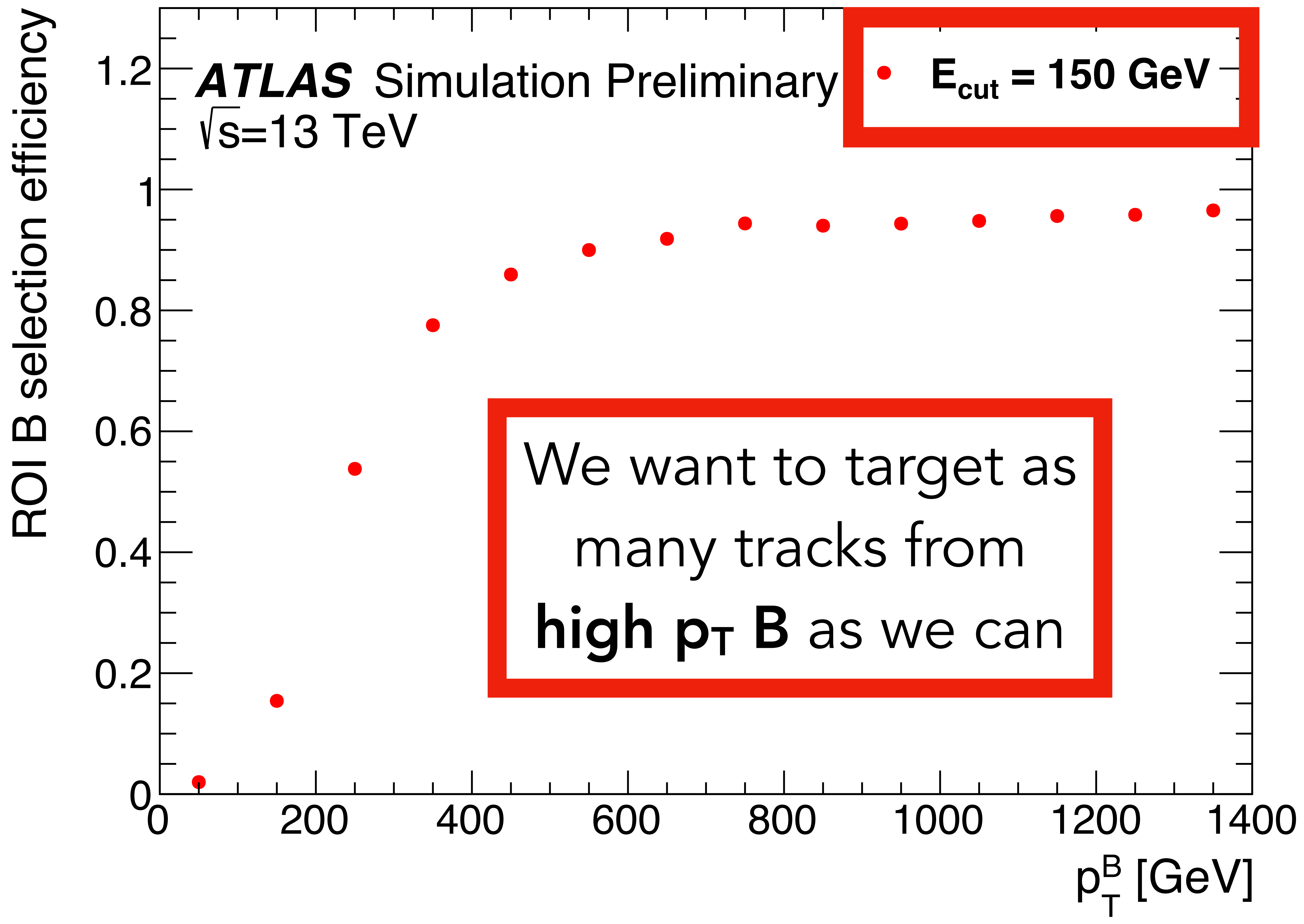
$\mathbf{P_t}^{\mathbf{cut}}$ on track candidates

$\mathbf{E_{cut}}$

$\mathbf{dR} = \sqrt{\Delta\phi^2 + \Delta\eta^2}$ between cluster and track < 0.1

Defining a Region Of Interest (ROI)





Wrong hits assignment

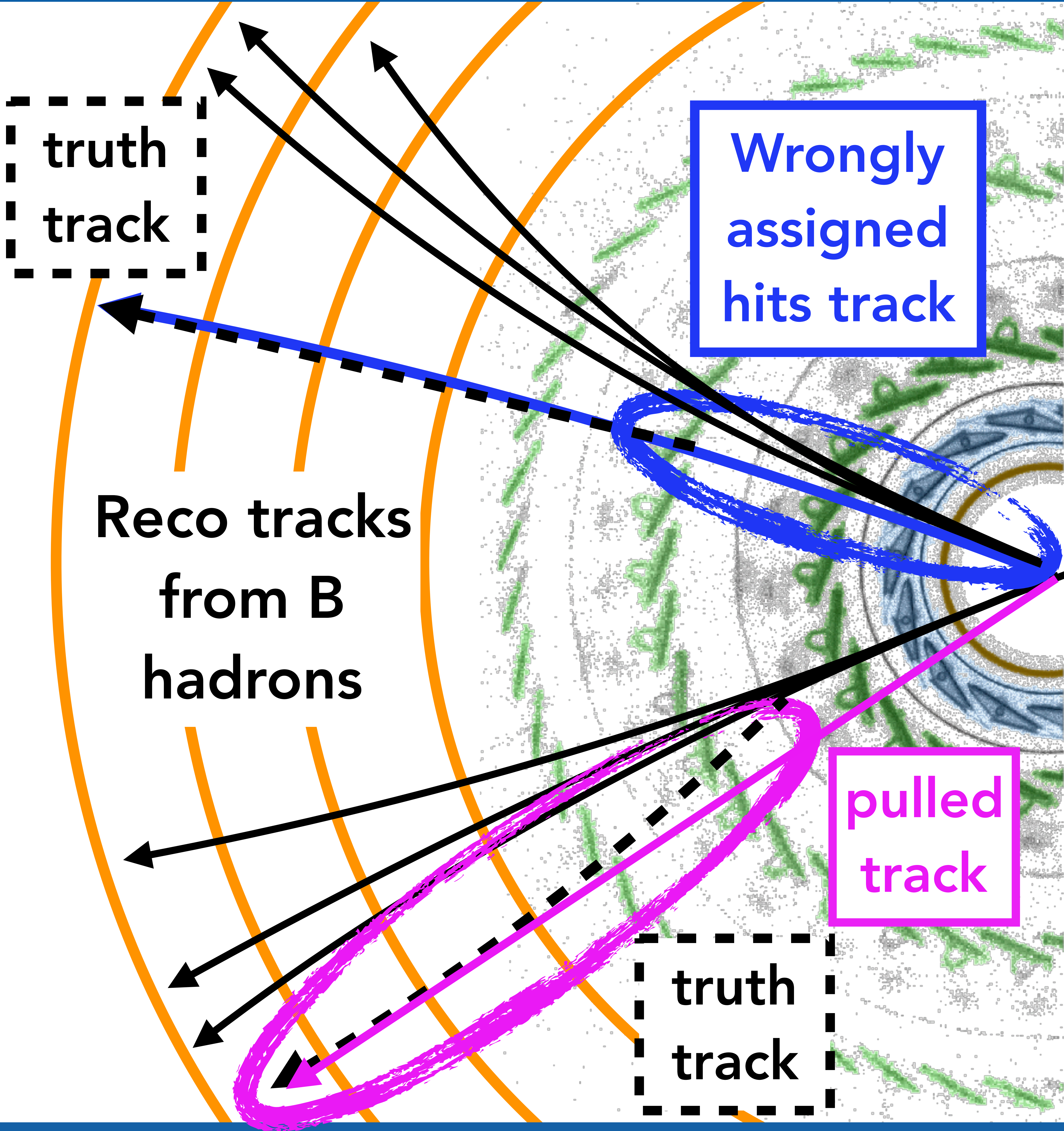
Picking up **wrong** inner layer **hits** means **pulling** the track which results in poor fit quality

⟱

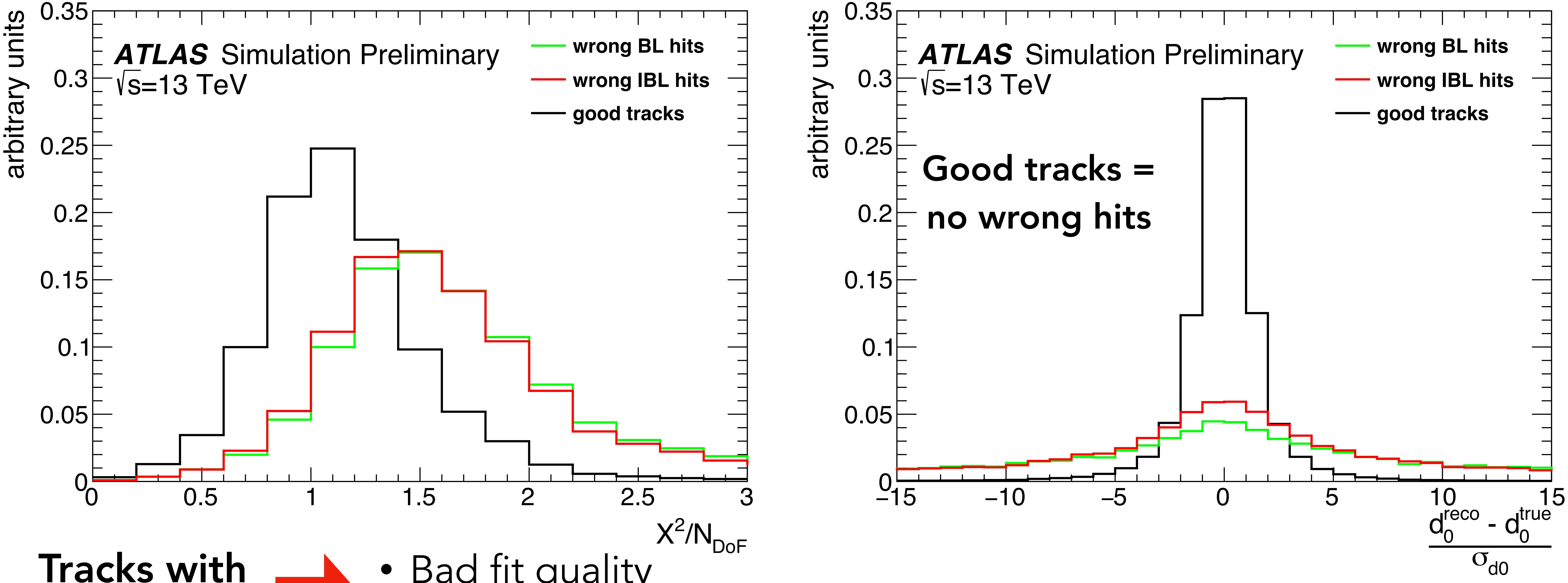
We try to remove recursively the hits (starting from IBL up to Pix2) and we refit the track each time

⟱

if the quality improves (20% improvement) we keep the **stripped down version** of the track



The wrong hits assignment problem

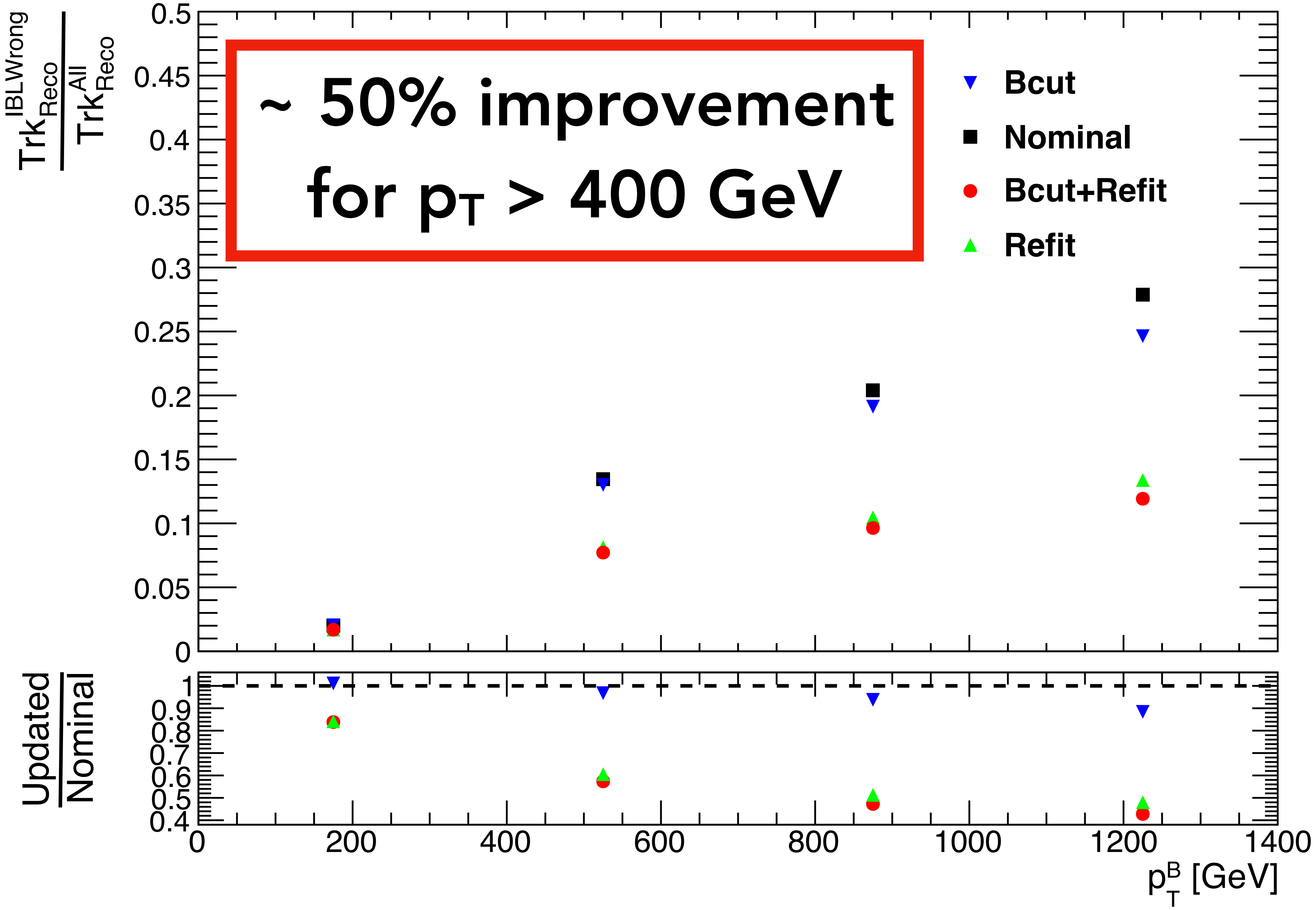


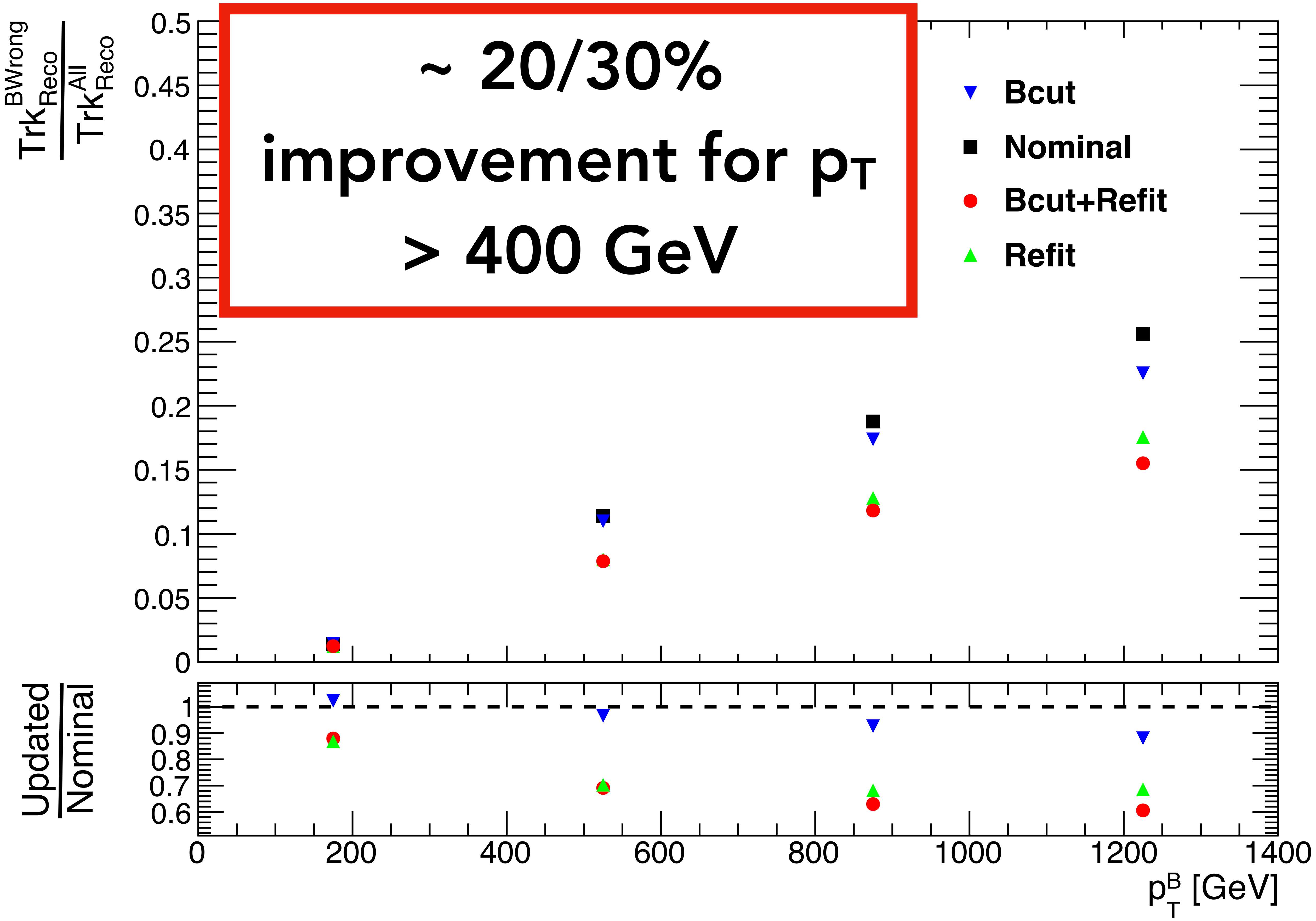
Tracks with wrong hits ➜

- Bad fit quality
- Large pull distribution ➜ lose information on impact parameter

Results for tracks from B hadrons

Showing the % of tracks from B with a wrong IBL and/or BL hit

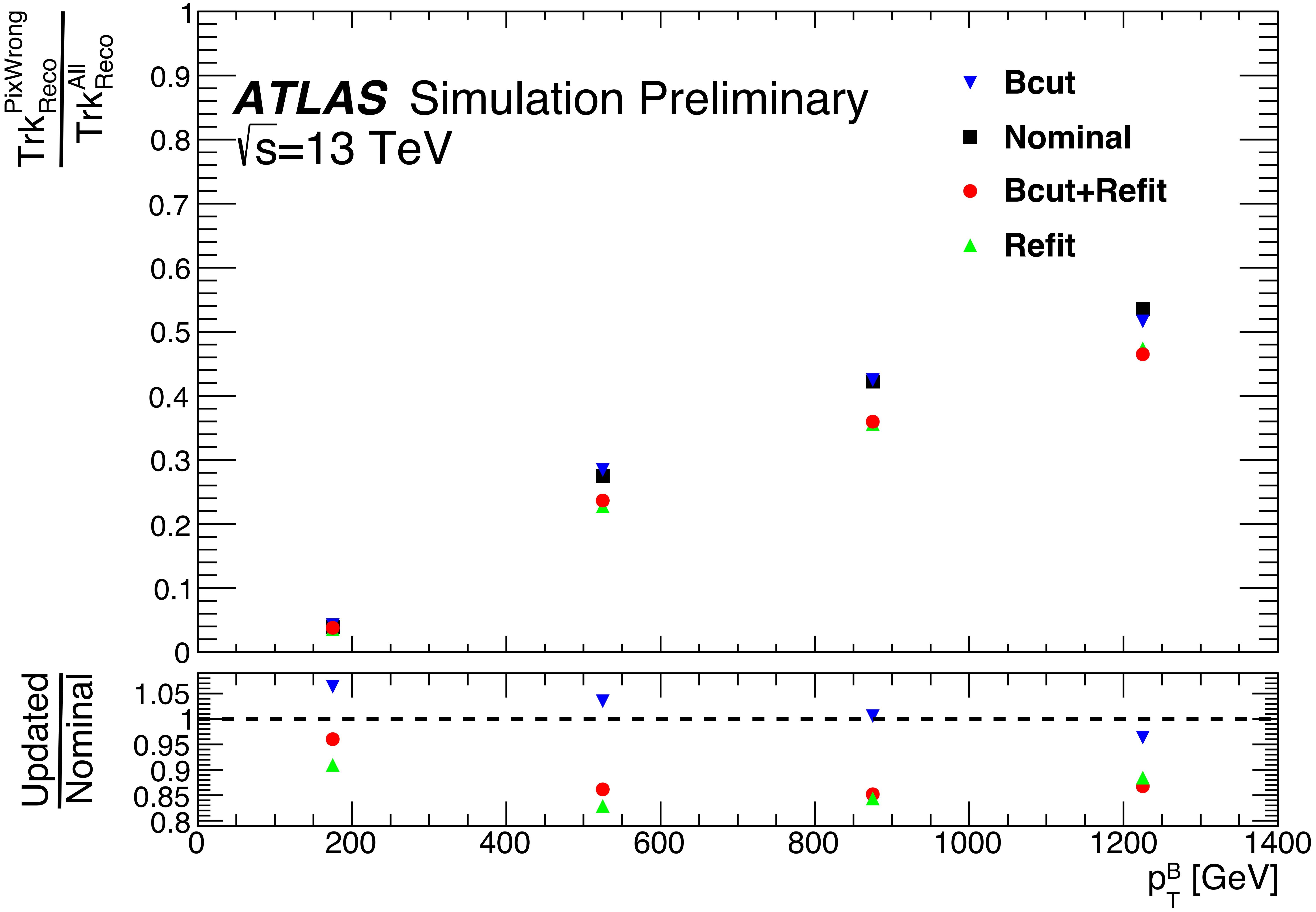




Results for tracks from B hadrons

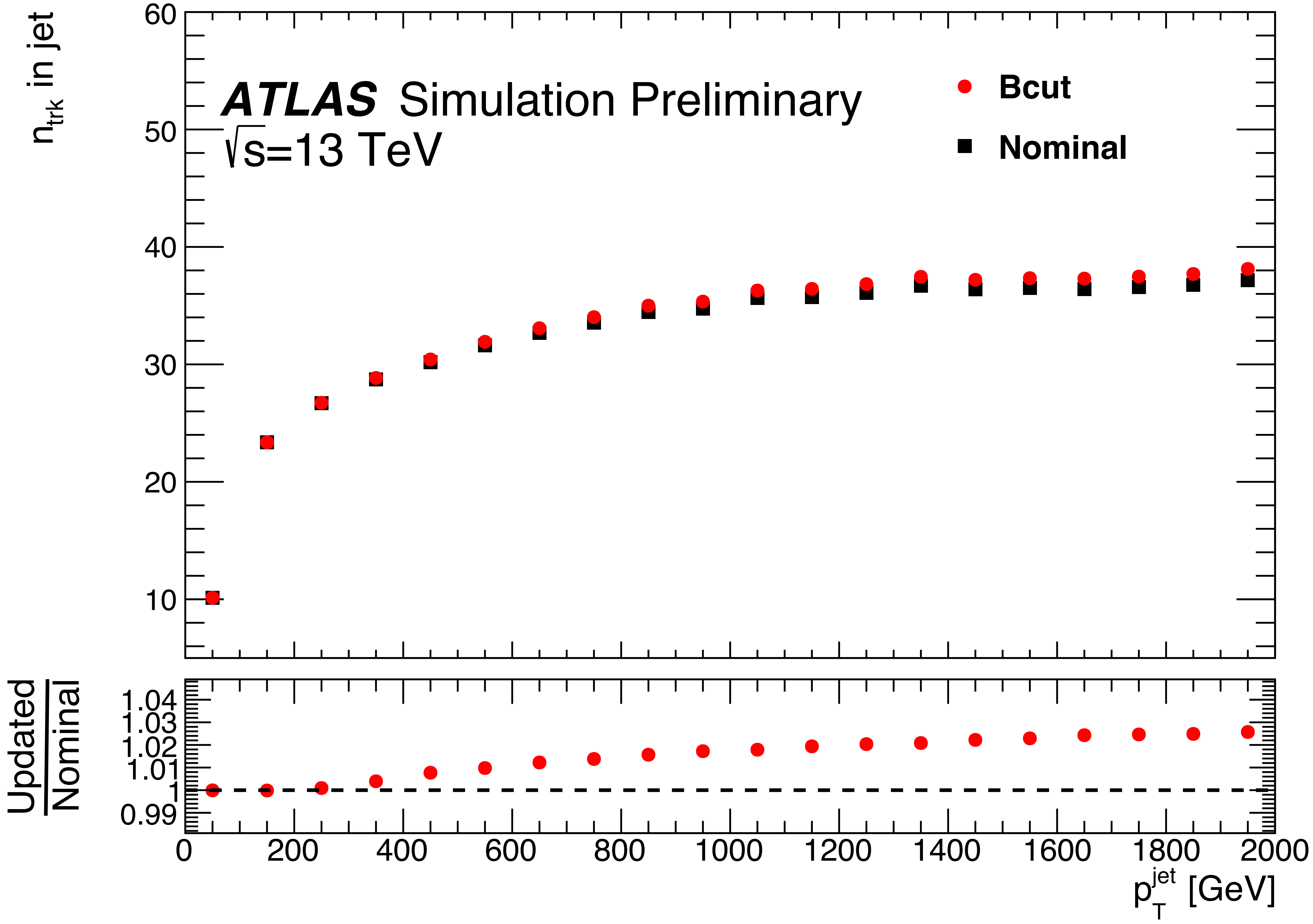
Overall effect if we consider all pixel hits we have an improvement of ~10%

Only apply the refit procedure up to Pix2



Impact on number of tracks per jet

Increase in number of tracks in jet < 2%, convoluted effect of fake rate and increase in efficiency



Outline

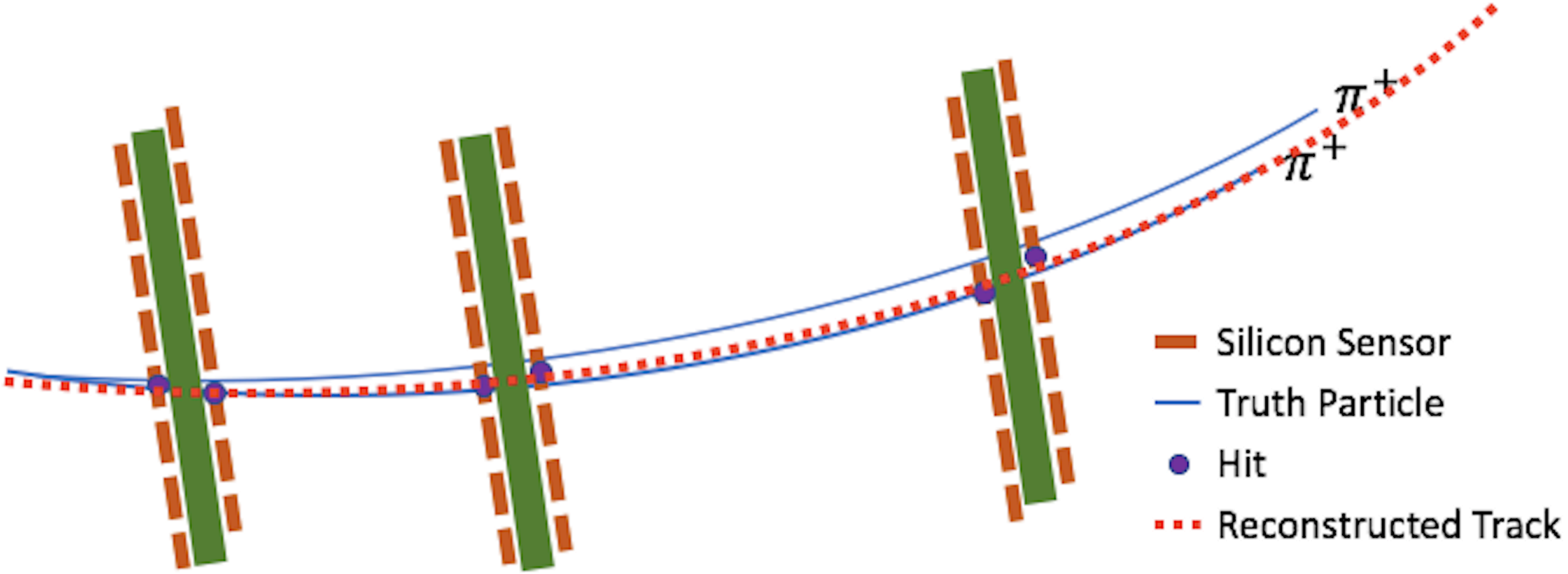
- Introduction
- Tracking in ATLAS ➜ Focus on dense environments
- Tracking for high p_T B-hadrons:
  - Defining a region of interest
  - Tackling the efficiency loss
  - Improving track quality
- Tracking for high p_T τs:
  - Fixing merged tracks
- Conclusion

High p_T τs

William Patrick McCormack

- Highly **collinear** particles ➜ many shared hits
- Two tracks reconstructed as one ➜ merged tracks
- Problematic for high p_T τs: analysis level has specific selections for 3-prong τs ➜ missing tracks = τs not identified
- Hard to tackle with simple cut-based approach
- Special MC sample: no pileup, particle gen. with η = 0, single τ to 3 π

Merged tracks problem

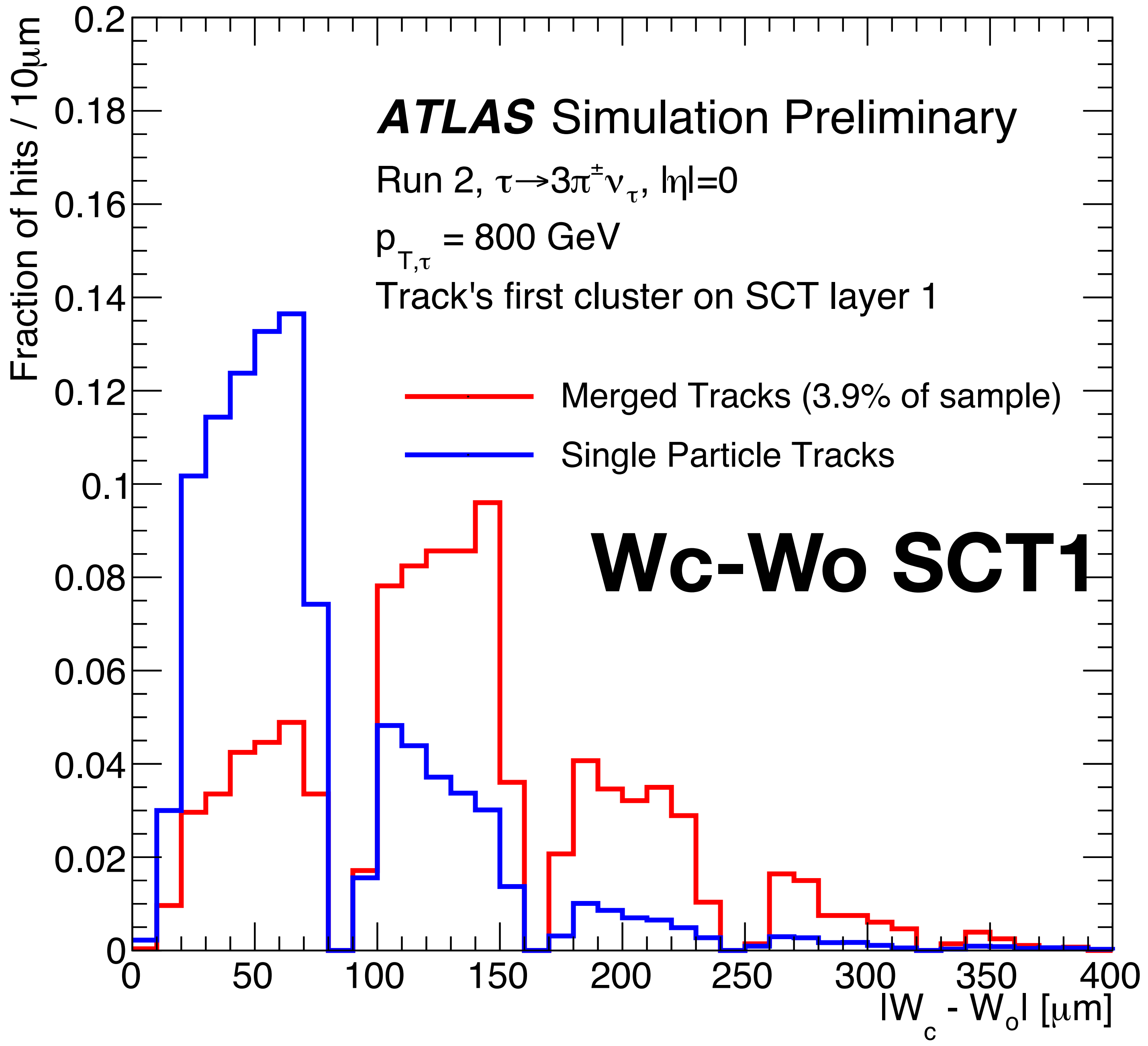


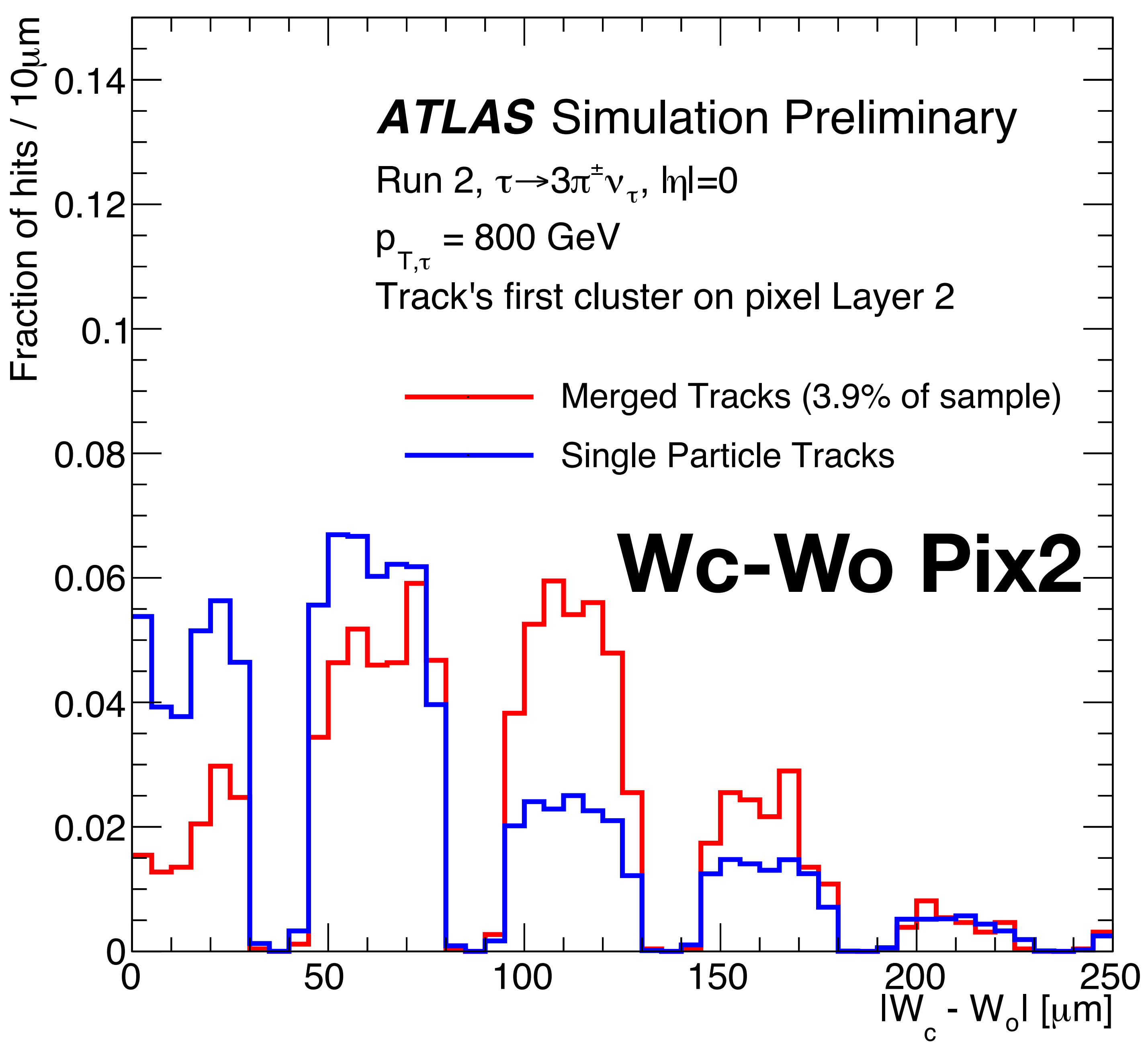
Building the BDT

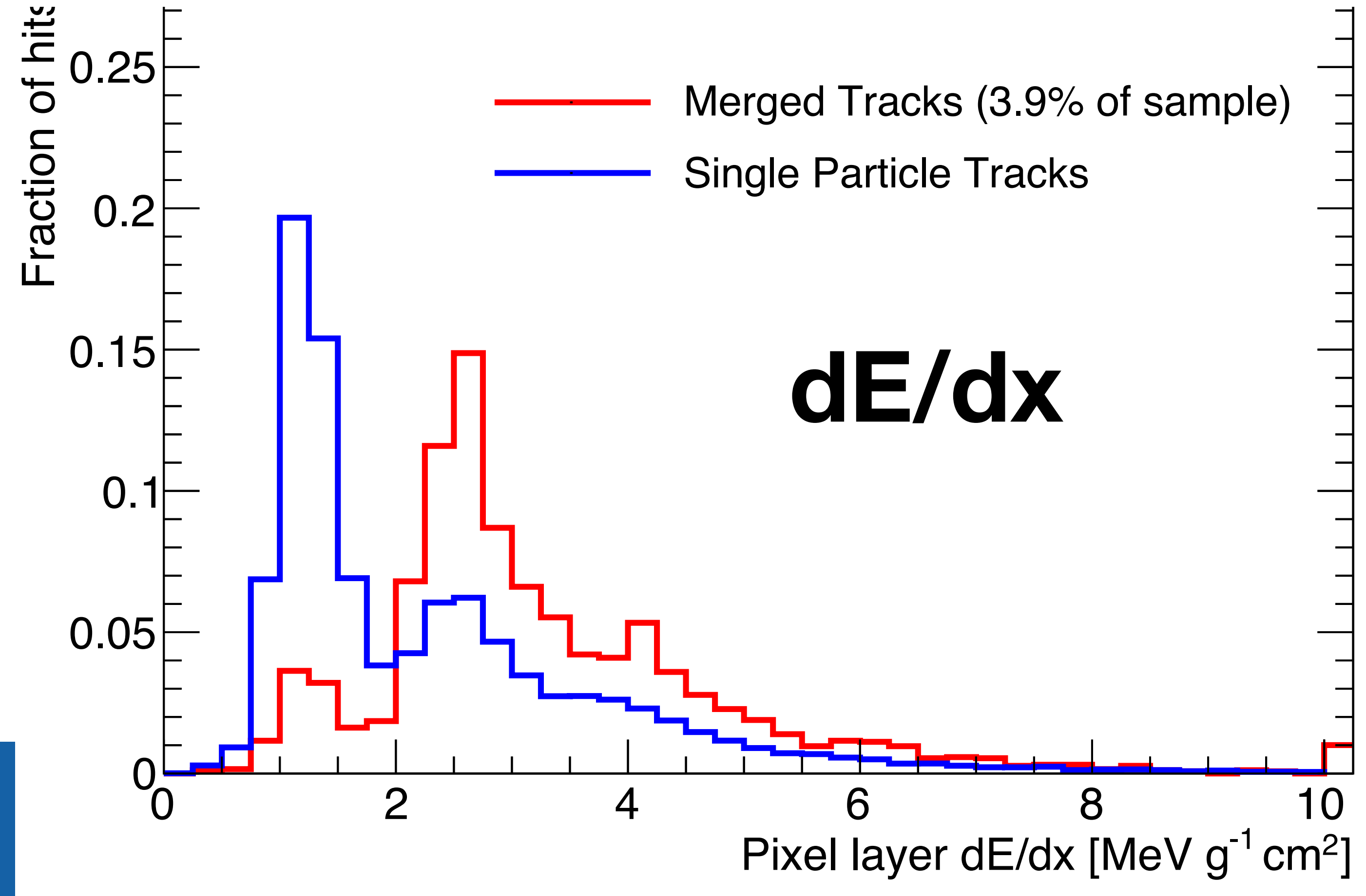
Variable	Num. of Features
Track p_{T}	1
Track η	1
Track ϕ	1
Num. clusters on each pixel layer	4
Highest charge deposited in a cluster on each pixel layer	4
dE/dx in each pixel layer	4
$\lvert W_{\mathrm{c}} - W_{\mathrm{o}}\rvert_{pix}$ for pixel cluster with the highest charge on each pixel layer	4
Boolean for whether a hit on each pixel layer is flagged as split	4
$\lvert L_{\mathrm{c}} - L_{\mathrm{o}}\rvert_{pix}$ for pixel cluster with the highest charge on each pixel layer	4
Num. clusters on each SCT layer	4
$\lvert W_{\mathrm{c}} - W_{\mathrm{o}}\rvert_{SCT}$ for SCT cluster with the lowest pull on each SCT layer	8
Num. shared clusters on each SCT layer	4

43 track variables fed into the BDT

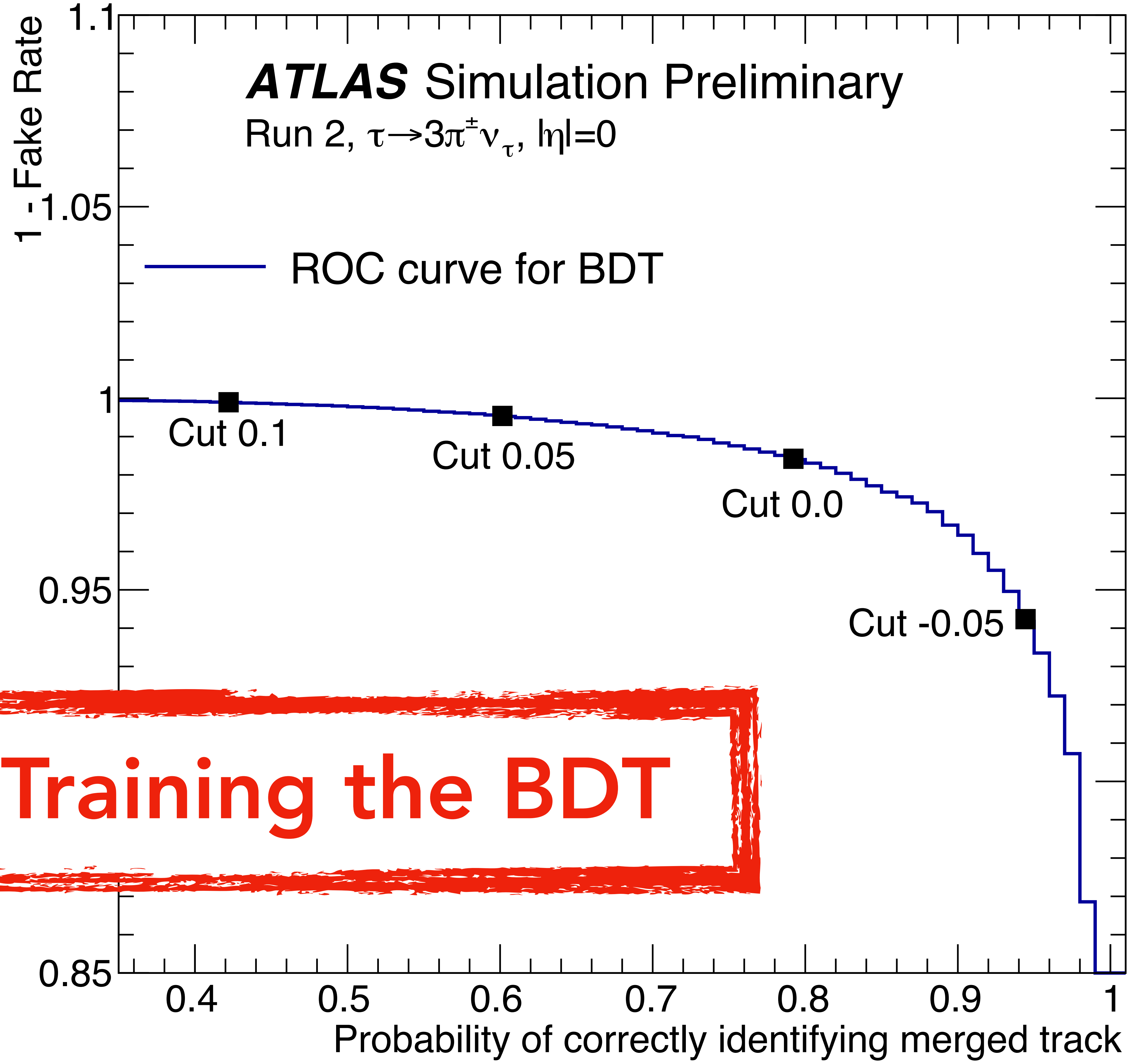
Fixing the merged tracks







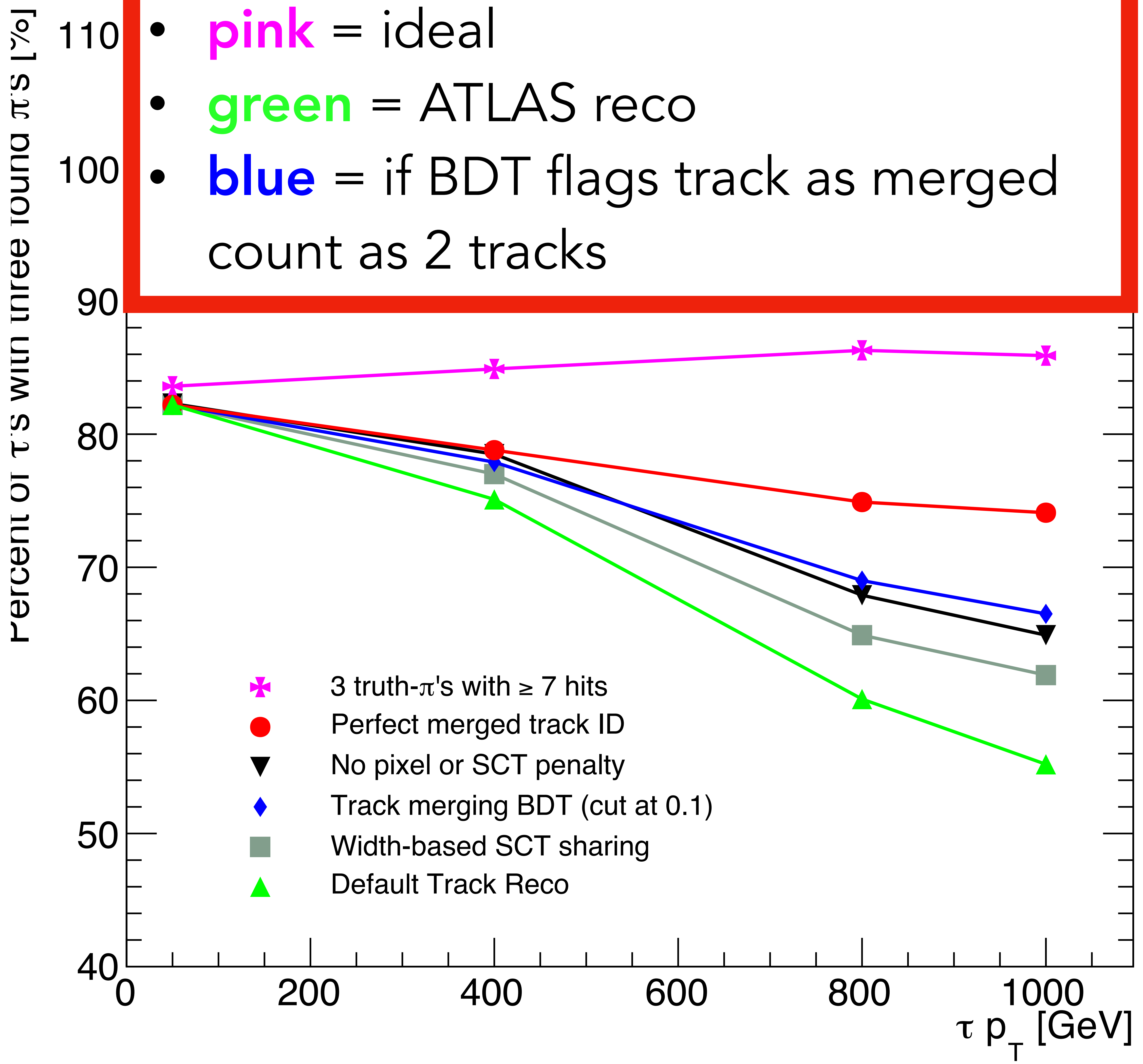
Results



Training the BDT

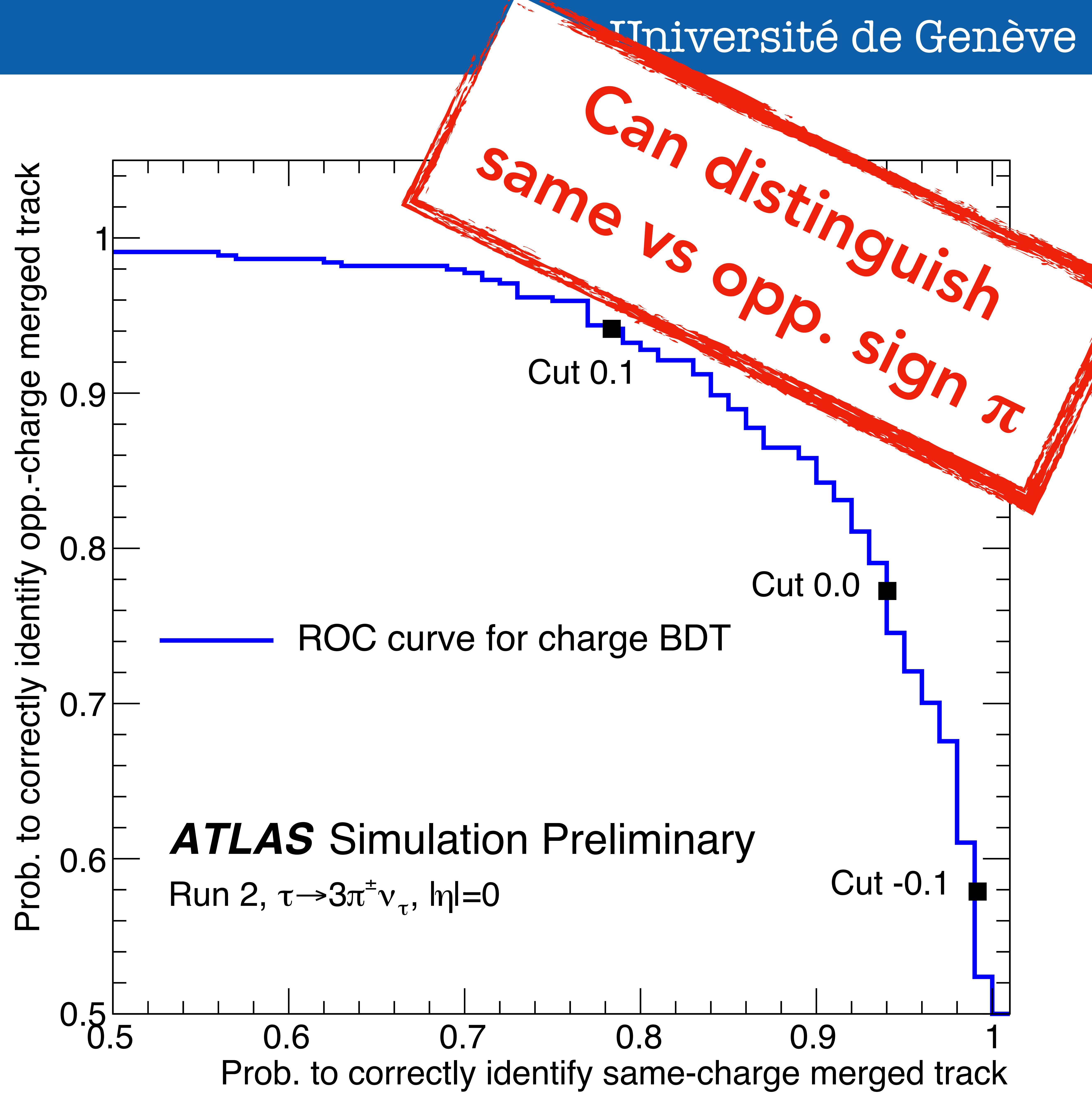
Recover τs ➜ larger with increasing p_T:

- **pink** = ideal
- **green** = ATLAS reco
- **blue** = if BDT flags track as merged count as 2 tracks



Pion charge BDT

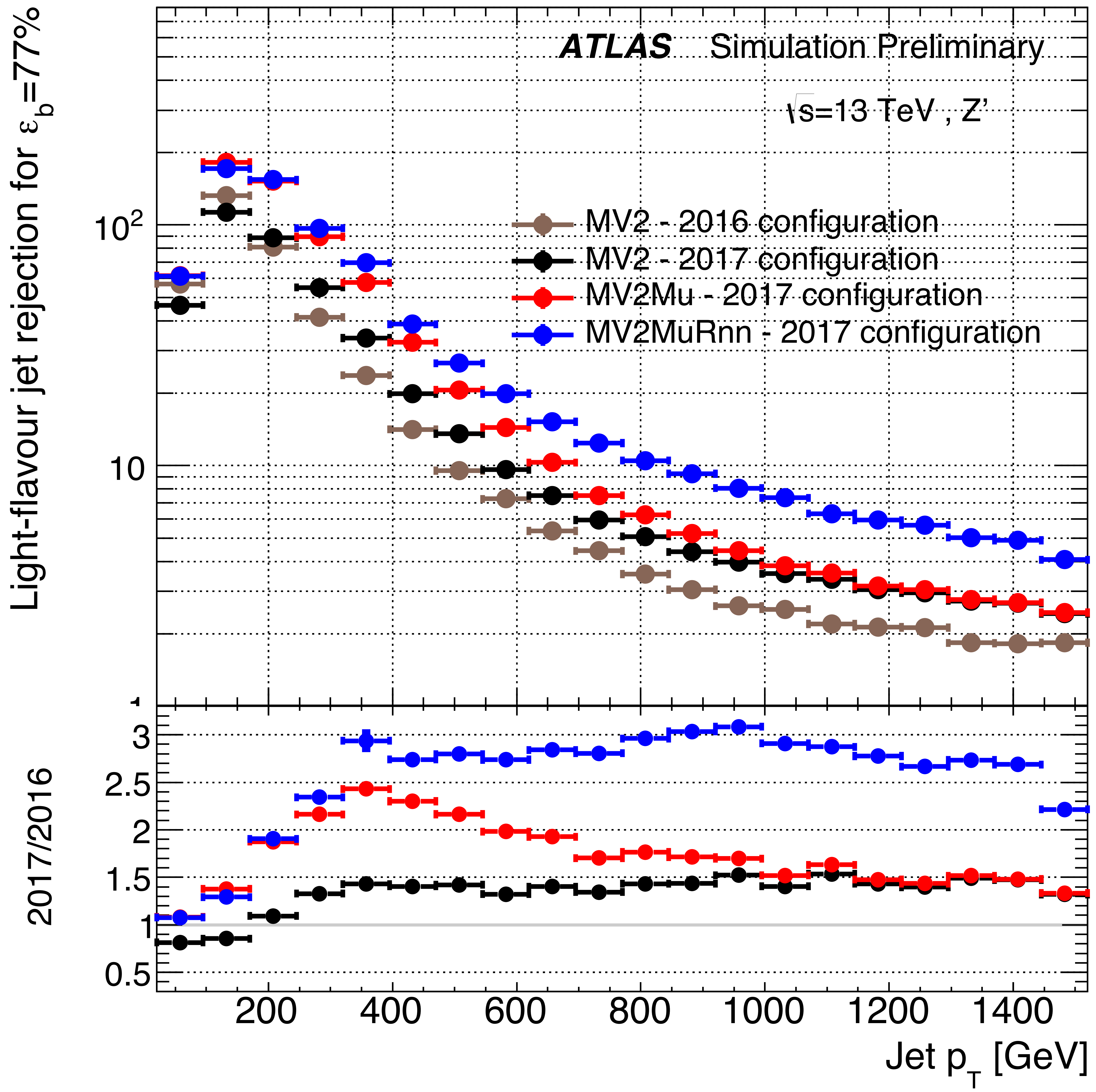
- Using same input but different training
  - on tracks identified as merged at truth-level and flagged by original track-merging BDT
- Only runs on merged tracks
- **With charge BDT cut of 0, charge-tagging accuracy is over 85%**

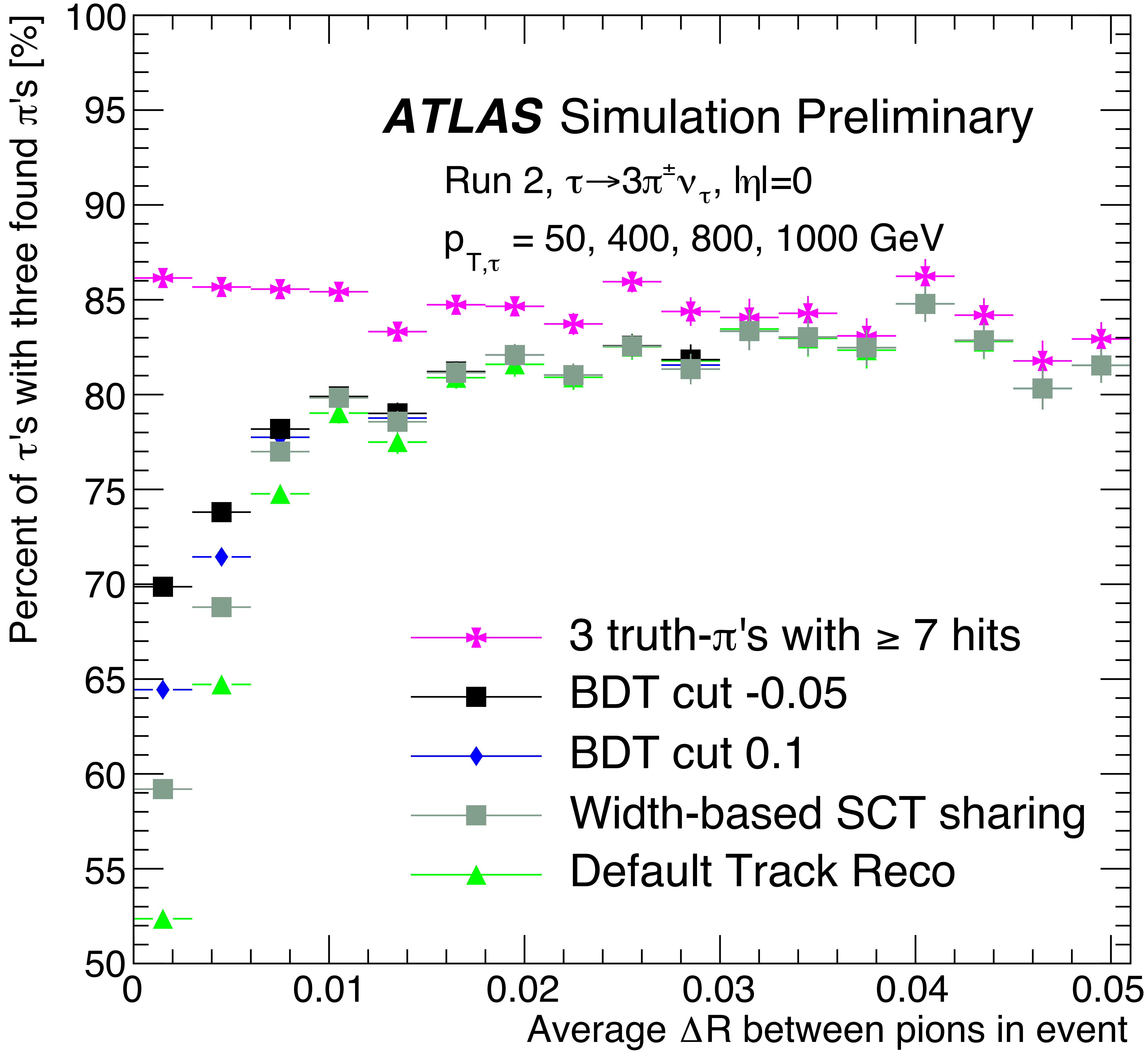


Conclusions

- **Increasing efficiency for tracks coming from B :**
  - By loosening the cuts we manage to improve the efficiency by ~5/10% per track coming from high pT Bs ➜ should result in even larger efficiency for B hadrons
  - The refit procedures allows us to remove some of the wrongly assigned tracks and thus improve the quality of tracks coming from B
  - The ROI selects tracks coming from B ➜ high efficiency and purity
  - Further improvement possible with better optimisation
- **Identifying merged tracks from τ :**
  - Promising results by using BDT to identify merged tracks ➜ recover efficiency for 3 prong τs
  - Low impact on number of duplicate tracks
  - Can also be used to distinguish same vs opp. sign pions

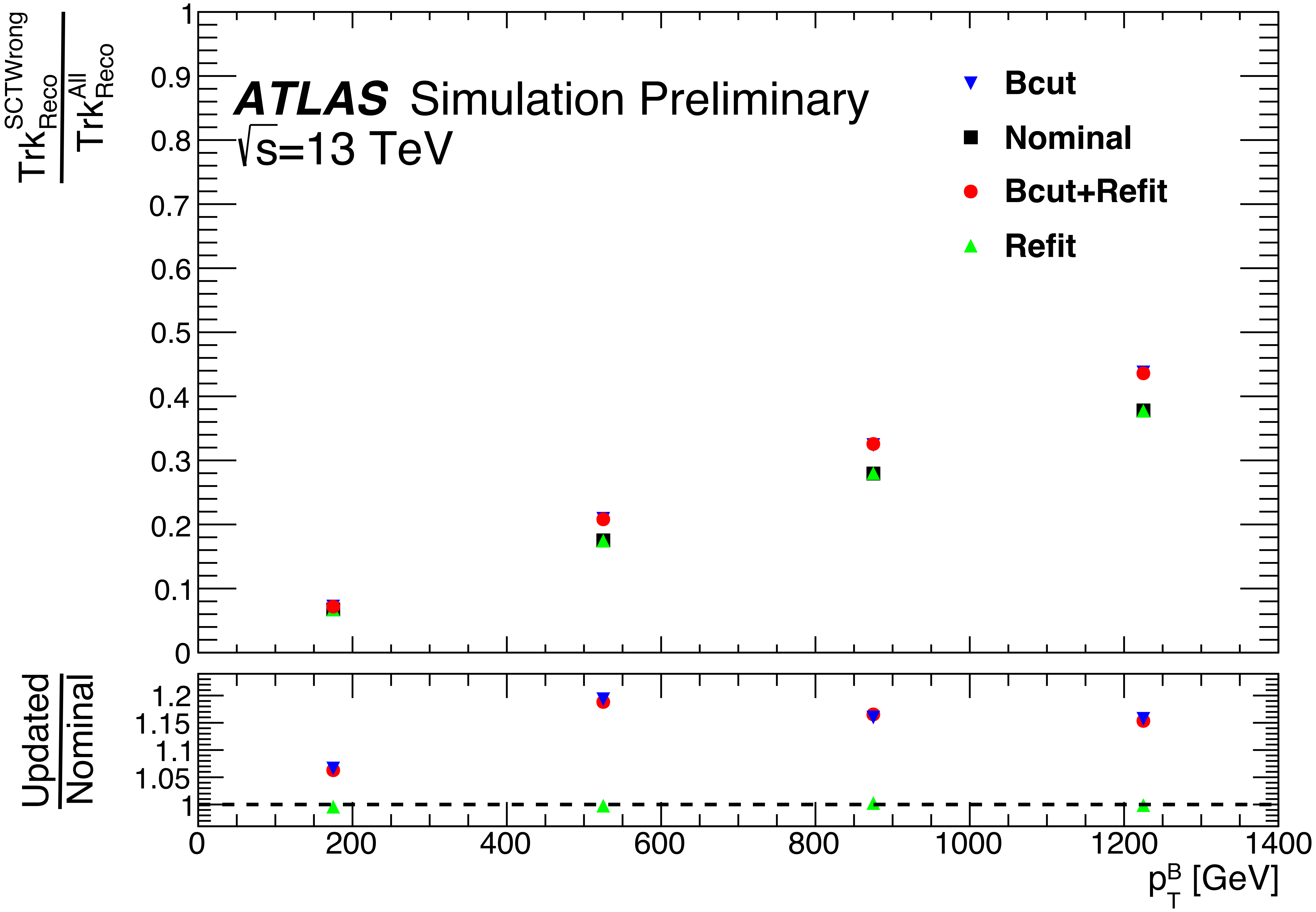
BACKUP

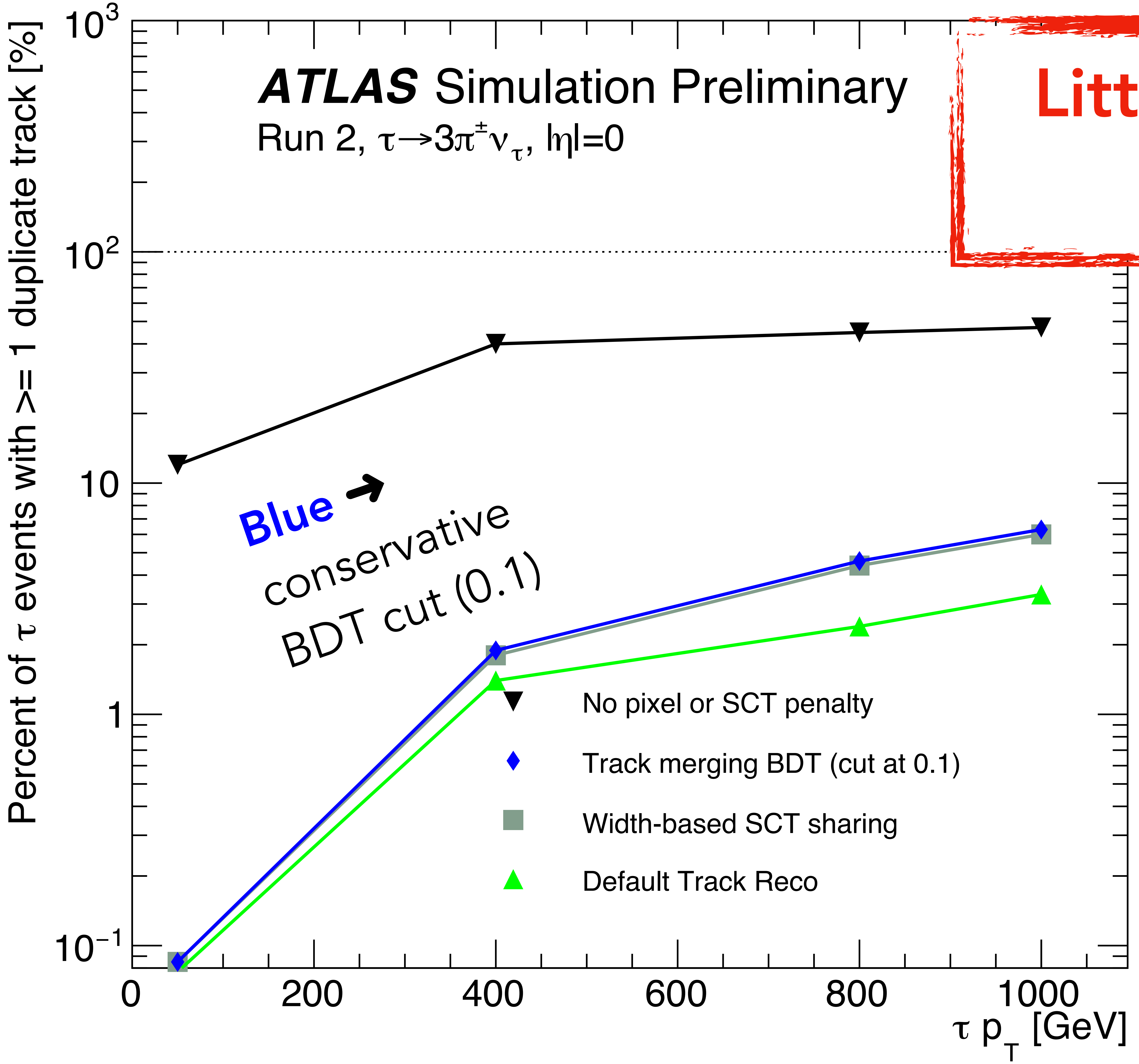

ATLAS Simulation Preliminary
√s=13 TeV , Z'
MV2 - 2016 configuration
MV2 - 2017 configuration
MV2Mu - 2017 configuration
MV2MuRnn - 2017 configuration
Light-flavour jet rejection for ε_b=77%
10^2
10
1
2017/2016
3
2.5
2
1.5
1
0.5
200
400
600
800
1000
1200
1400
Jet p_T [GeV]


ATLAS Simulation Preliminary
Run 2, τ→3π±ντ, |η|=0
pT,τ = 50, 400, 800, 1000 GeV
3 truth-π's with ≥ 7 hits
BDT cut -0.05
BDT cut 0.1
Width-based SCT sharing
Default Track Reco
Percent of τ's with three found π's [%]
Average ΔR between pions in event
100
95
90
85
80
75
70
65
60
55
50
0
0.01
0.02
0.03
0.04
0.05

Results for tracks from B hadrons

Increased number of tracks with wrong SCT hits ➜ loosening the cuts allows for more errors but we have increased efficiency





Little impact on fake rate