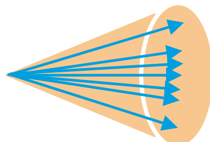
Clustering and Tracking in Dense Environments with the ITk

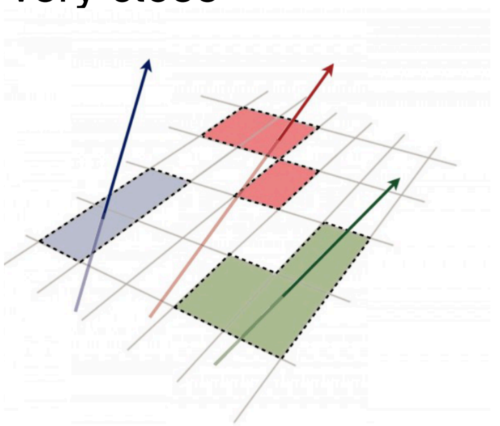
Nicola de Biase
DESY

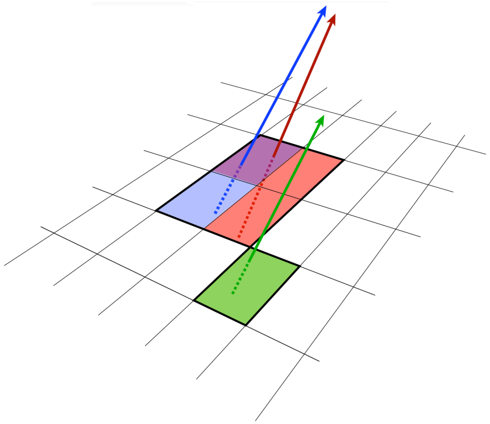
21.03.2022
DPG Spring Meeting

HELMHOLTZ RESEARCH FOR GRAND CHALLENGES

CTIDE
What is it?

- In a dense hadronic environment, reconstructed tracks may be very close
  - e.g. in core of high-p_T jets 
- Close-by tracks may share silicon clusters/hits
- Cluster merging affects
  - Tracking quality (fit quality is reduced)
  - Tracking efficiency (sharing clusters comes with a penalty)
- Currently, merged clusters are identified and split via NNs
- A new tracking detector will be introduced for HL-LHC
  - What will change for CTIDE?

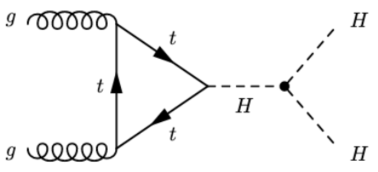




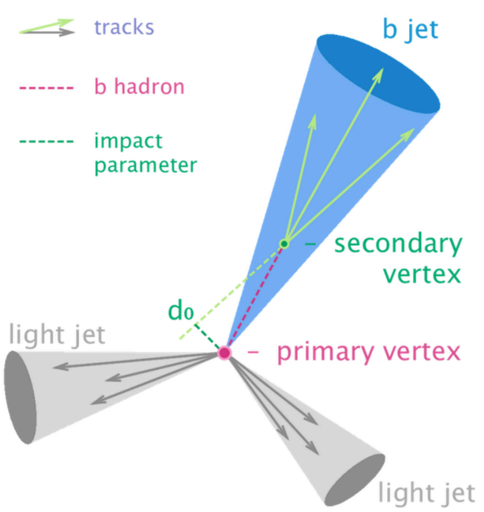
CTIDE
Why is it important?

> CTIDE affects physics analyses

> Measurement of Higgs tri-linear self coupling at HL-LHC



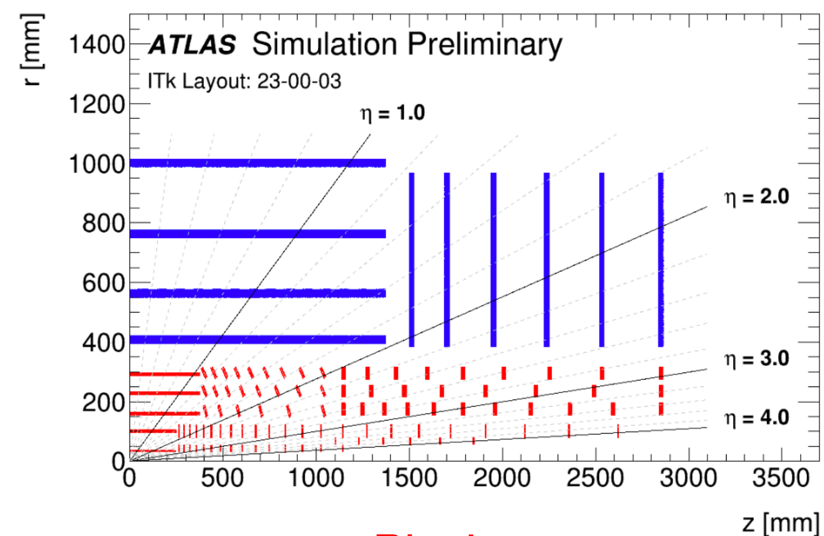
- Via Di-Higgs measurements with at least one $H \rightarrow b\bar{b}$
- Need accurate b-tagging
- Need precise measurement of displaced vertices (Inside of jets)



The ITk
Inner tracker for HL-LHC

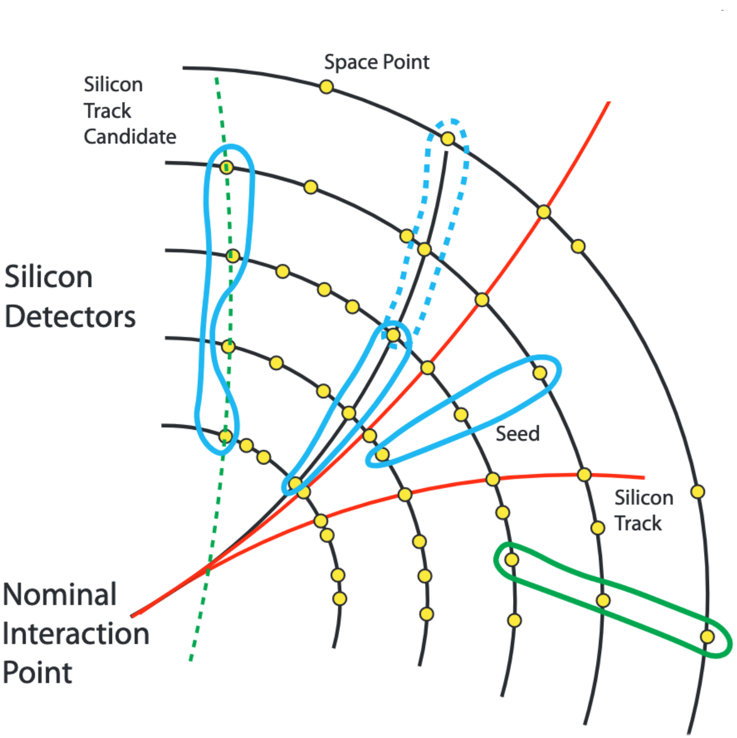
	ID (current)	ITk (HL-LHC)
Technology	Silicon + TRT	Full-silicon
Pixel pitch	50μm×300μm	25μm×100μm/ 50μm×50μm
Strip pitch	80μm	69-85μm
<Pile-up> (Interactions per bunch crossing)	Run-2: 13÷38 Run-3: 60	HL-LHC: 200

https://cds.cern.ch/record/2800852/files/ATL-ITK-PUB-2022-001.pdf



Pixel
Strip

Tracking in ATLAS
General



https://indico.cern.ch/event/504284/contributions/2023875/attachments/1240146/1823137/HGray_Zurich_Tracking.pdf

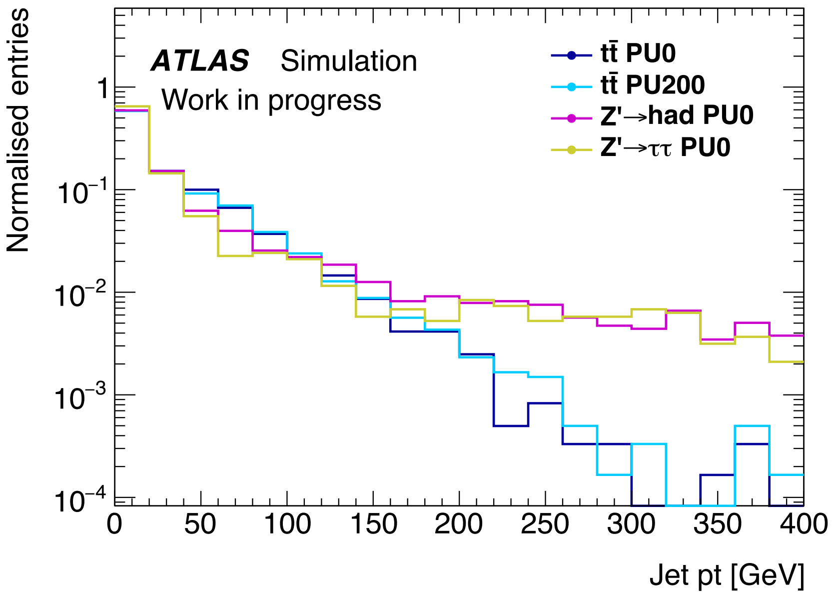
Clustering
↓
Track seeding
↓
Track finding
↓
Ambiguity solving
↓
Fitting — CPU intensive!
↓
Selection

Monte Carlo Samples

> Detector simulation obtained via Geant4

> Processes:

- $Z' \to$ had: Inclusive Z' hadronic decay with flat pT spectrum and no pile-up
- $Z' \to \tau\tau$: Z' decay to hadronically decaying τ, no pile-up
- $t\bar{t}$ PU0: Standard Model (SM) top pair production, no pile-up
- $t\bar{t}$ PU200: Standard Model (SM) top pair production, 200 pile-up interactions per bunch crossing



Is merging relevant?

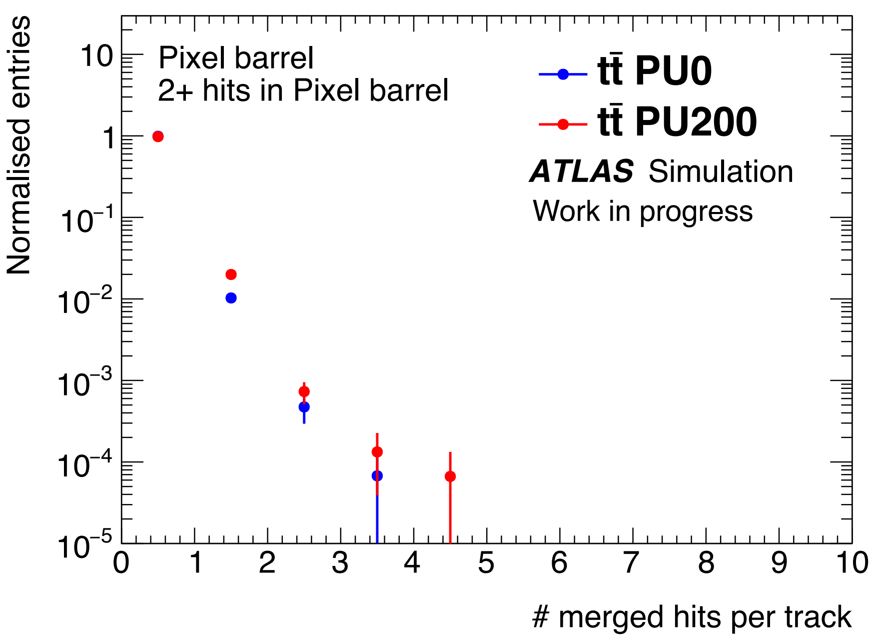
Track merging

> $$\text{Merging-affected tracks (section S)} = \frac{\text{\# tracks with a merged cluster in S}}{\text{\# tracks with} \geq 2 \text{ clusters in S}}$$

- Merged cluster = ≥2 contributing particles

> Relevant in all sections (Especially at high p_T - relevant for new physics!)

> PU increases merging



ATLAS Simulation - Work In Progress

Process	Whole	Barrel	Inclined	Ring
$t\bar{t}$ PU0	1.9%	1.1%	0.4%	2.3%
$t\bar{t}$ PU200	3.3%	2.1%	0.5%	3.4%
Z' PU0	10.7%	11.2%	2.8%	3.7%

Table 2: Track merging rate in different sections of the pixel detector.

ATLAS Simulation - Work In Progress

Process	Whole	Barrel	Endcap
$t\bar{t}$ PU0	1.8%	1.6%	1.4%
$t\bar{t}$ PU200	6.8%	5.3%	5.9%
Z' PU0	18.9%	18.6%	9.0%

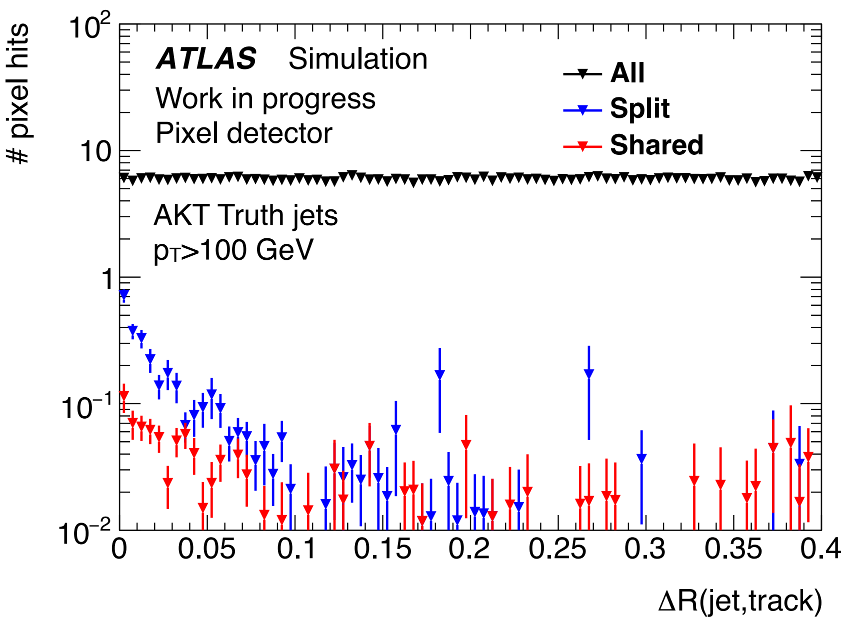
Table 3: Track merging rate in different sections of the strip detector.

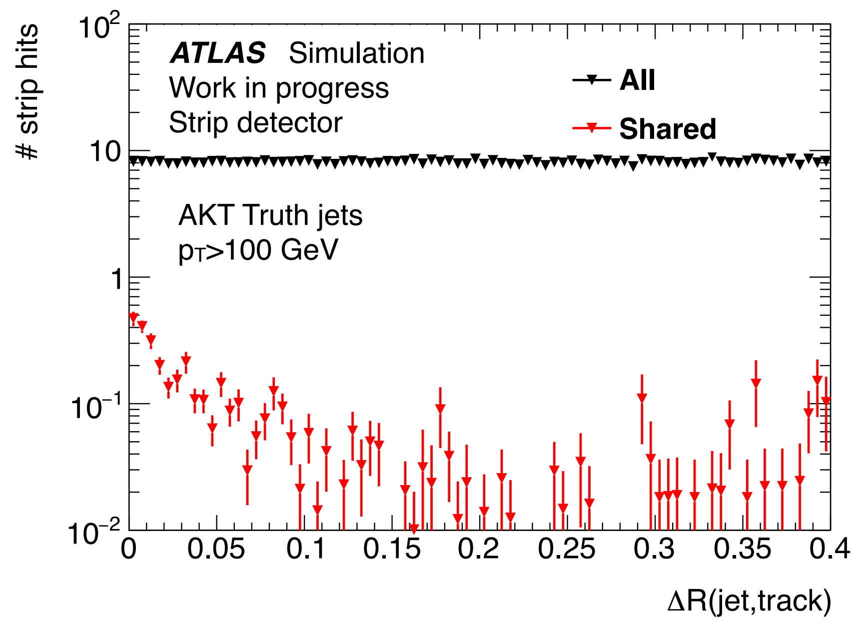
Is merging relevant?
Tracks inside of jets

> Pixel split hits = used for multiple tracks and merged

> Pixel shared = used for multiple tracks but NOT merged

> Strip shared = used for multiple tracks





Track overlap is relevant at the centre of jets

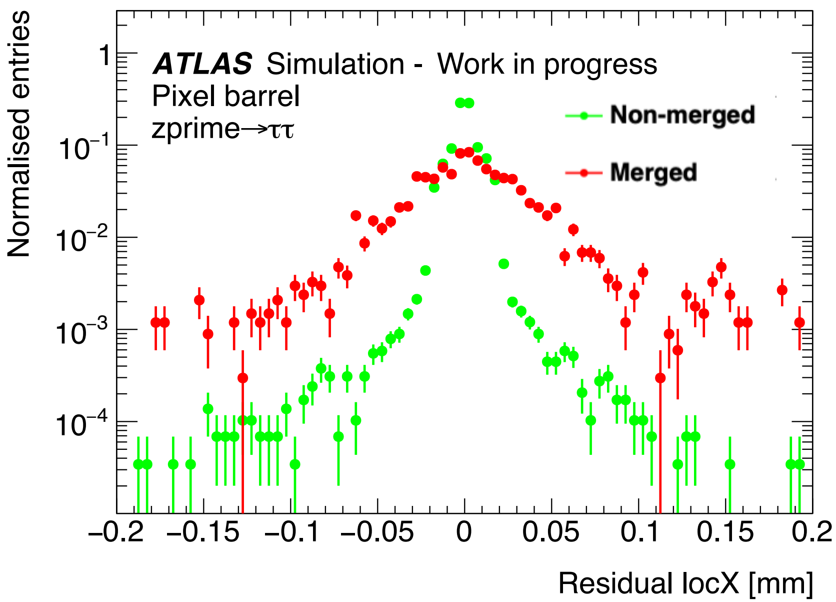
How does merging degrade tracking?
Residuals and pulls

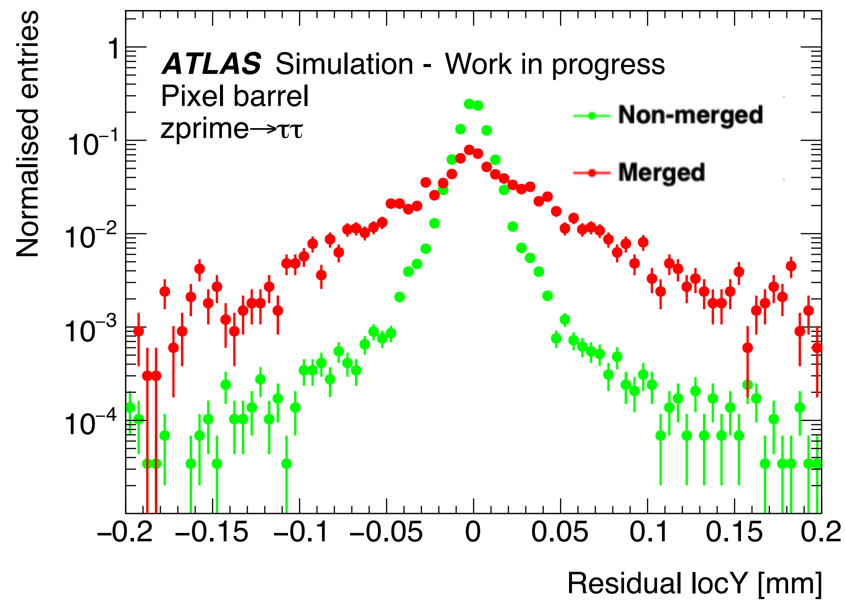
$$\text{Residual} = (\text{Reco}) - (\text{truth})$$

$$\text{Pull} = \frac{(\text{Reco}) - (\text{truth})}{\sigma(\text{Reco})}$$

How does merging degrade tracking?

Cluster-truth residuals





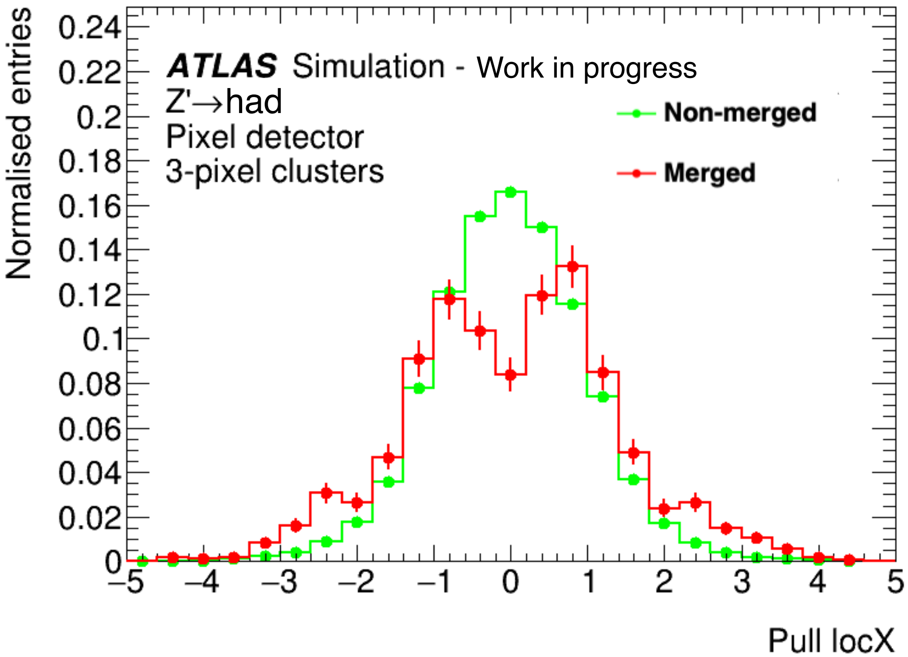
> Merging degrades hit position resolution

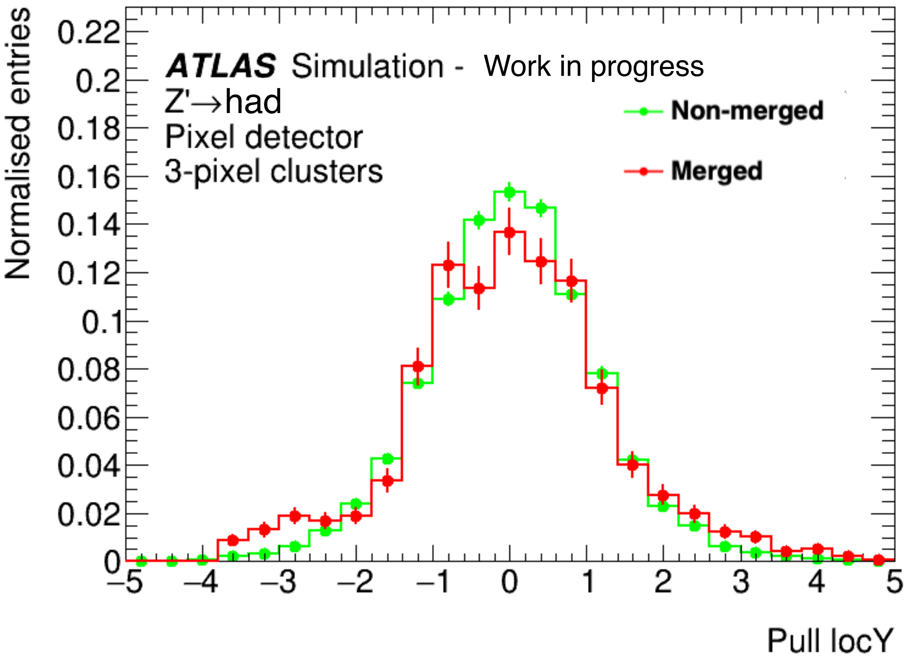
> Local coordinates:

- X: in the r-ϕ plane
- Y: in the η direction

How does merging degrade tracking?

Track-cluster pulls

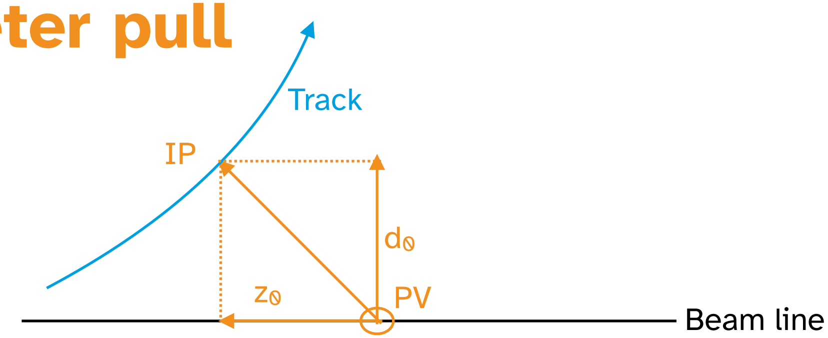


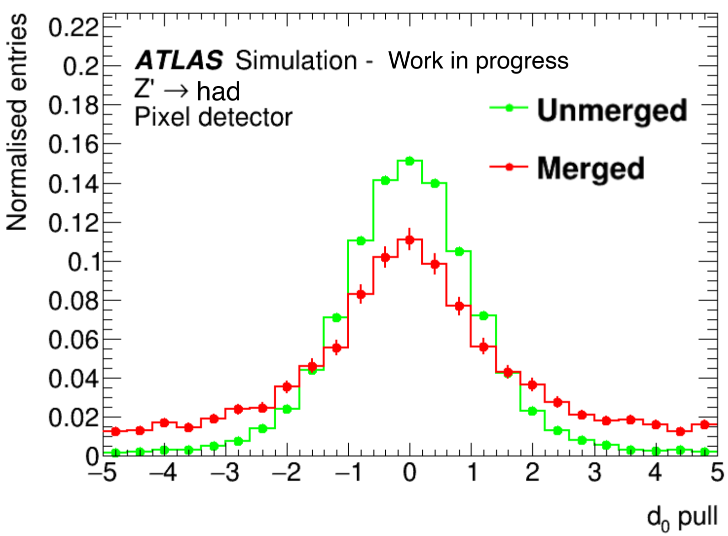


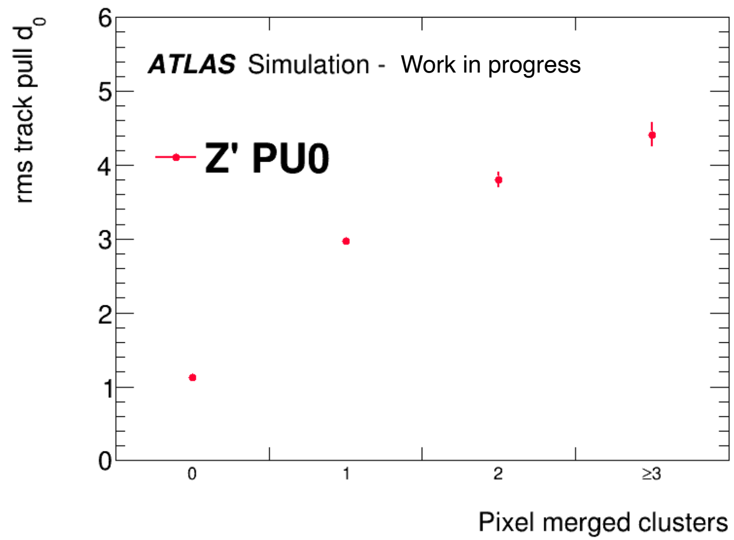
> Merging degrades tracking by pulling clusters away from tracks

How does merging degrade tracking?

Impact parameter pull







> Merging degrades impact parameter estimation
  - Impacts flavour tagging

> IP estimation worsens with increasing number of merged clusters

Conclusions

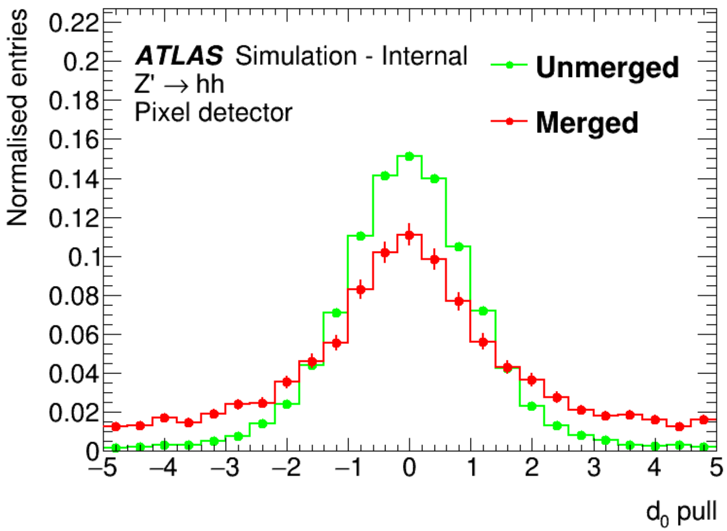
> Rate of merging-affected tracks relevant in the ITk

> Cluster merging relevant for track performance

- Degrades clustering performances
- Degrades track parameter estimation
    - d_0 degradation → Worse flavour tagging

> Remaining question:

- How much room for improvement via splitting?



Backup

Tracking in ATLAS
Ambiguity solving

> Cluster classification (Only for clusters on multiple tracks)
  - Evaluate expected number of contributing particles
    - NN with ID, based on MC truth with ITk

> Cluster correction
  - Correct the cluster position and relative error (Splitting)
    - Machine learning algorithm with ID, not performed with ITk
    - Is splitting needed with the ITk?

Only for Pixel clusters

Tracking in ATLAS
General

- Combine signals in the tracker into clusters and convert them into space-points
  - Clusters interpreted as deposits left by individual particles
- Create track candidates
  - Combine space-point triplets into track seeds
  - Extend track seeds
- Ambiguity solving and fitting
  - Reject combinations of unrelated clusters
  - Resolve overlaps between candidates
  - Fit
- Track selection:
  - Bad tracks rejected via a score (penalising cluster sharing)

Tracking in ATLAS
General

> Form silicon clusters

> Create track candidates

> Ambiguity solving and fitting

- Reject combinations of unrelated clusters
- Resolve overlaps between candidates
- Estimate track parameters

> Track selection:

- Bad tracks rejected via a score (penalising cluster sharing)