(Outer) tracker simulations

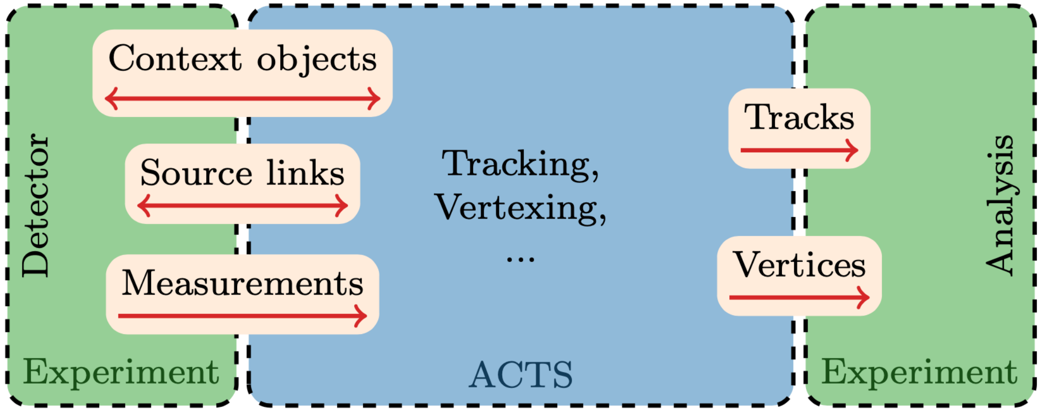
Berkin Ulukutlu, Henrik Fribert, Daniel Battistini,
Igor Altsybeev (TUM), Pavel Larionov (Inha University)

ALICE Upgrade Week
October 7, 2024

A Common Tracking Software Project

Overview:
Comp. and Soft. for Big Science, 6, 8 (2022)

Example of integration into experiment's framework:



- Being developed since 2016: Github
- Based on experience with track reconstruction in ATLAS
- **Experiment- and framework-independent toolkit**
- **High-level track reconstruction tools**
  - agnostic to the details of the detection technologies and magnetic field configuration

Why interesting for us:

- Allows track reconstruction for ALICE 3 geometry in the full range $|\eta| < 4$
- Our current O2 software tailored for central barrel reconstruction

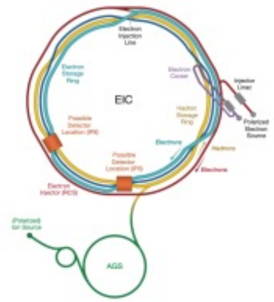
Some ACTS “clients”

[ACTS workshop, Nov 2023](#)

ATLAS EXPERIMENT
- ACTS Vertex reconstruction in Run-3
- Full ACTS powered reconstruction for Phase-2

sPHENIX — Full ACTS-powered track reconstruction

FASER — Full ACTS- powered track reconstruction



Electron-Ion Collider (EIC) software stack,
common track reconstruction software based on ACTS

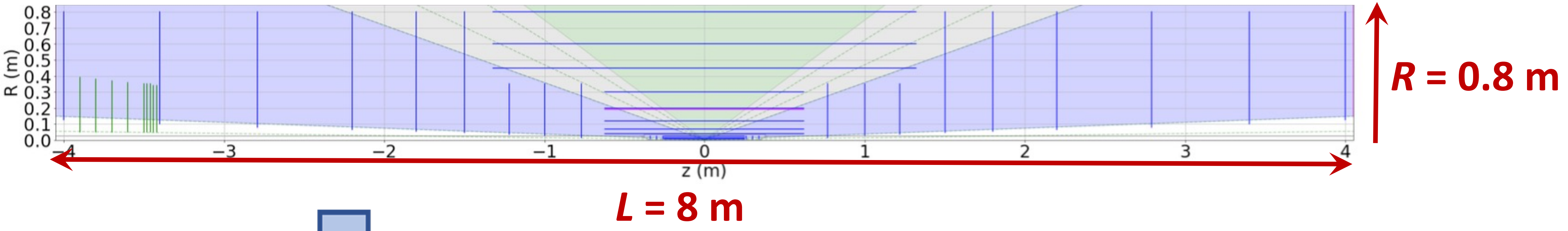
CEPC — Test implementation for CEPC design study

… also NA60+, LDMX, STCF, Lohengrin, BGV

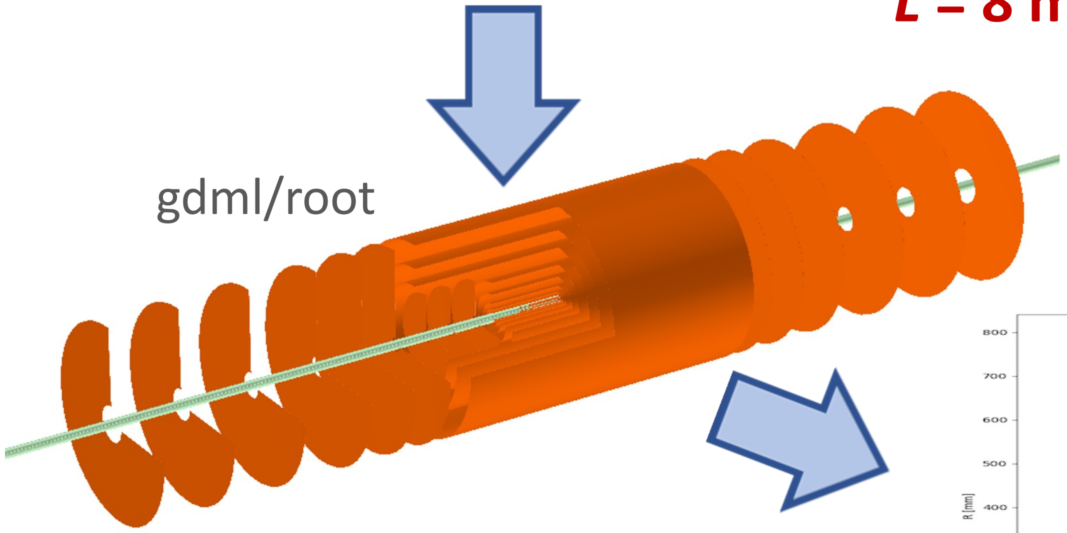
09:00	Introduction
	ATLAS
	ALICE
	Coffee Break
10:00	
	CEPC and STCF
	FASER
11:00	
	Food: Lunch
12:00	
13:00	
	NA60+
	LDMX
	Lohengrin
14:00	
	ePIC
	sPHENIX

Tracker layout in ACTS

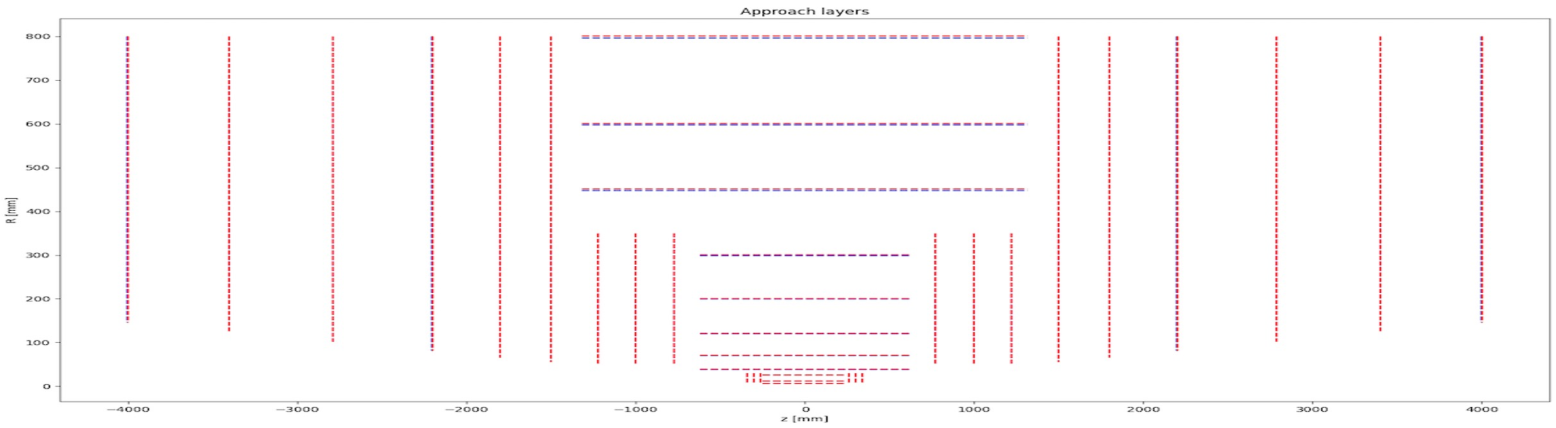
ALICE 3 tracker geometry



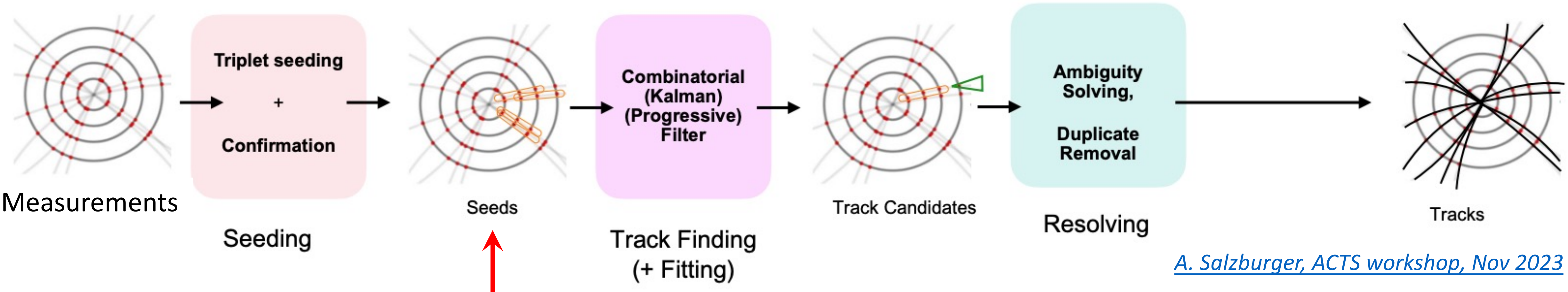
gdml/root



ACTS representation (cylinders + disks)
with material mapped on surfaces



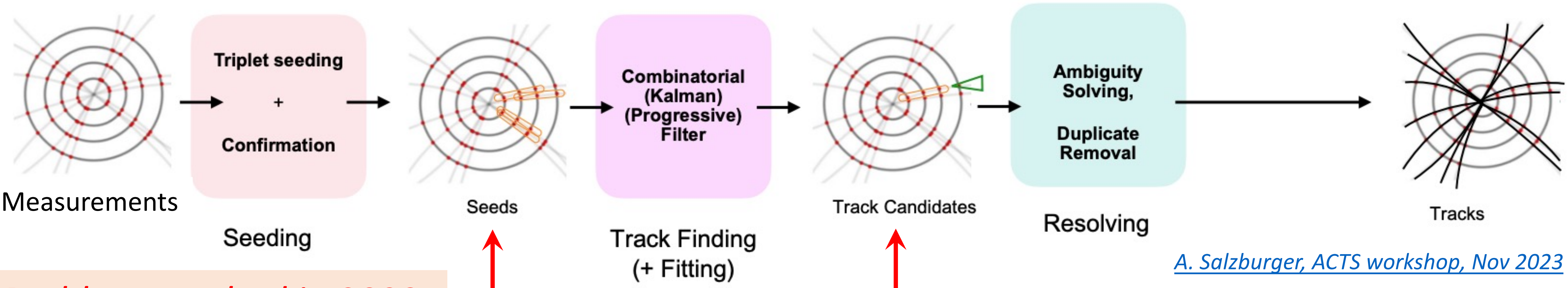
Reconstruction: **the "full chain"**



A. Salzburger, ACTS workshop, Nov 2023

Seeds = triplets of points
to initialize tracking

Reconstruction: **the ”full chain”**

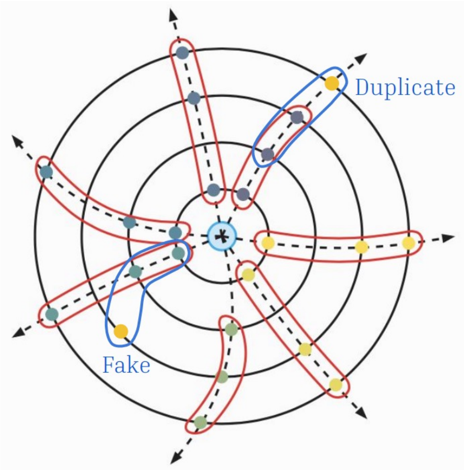


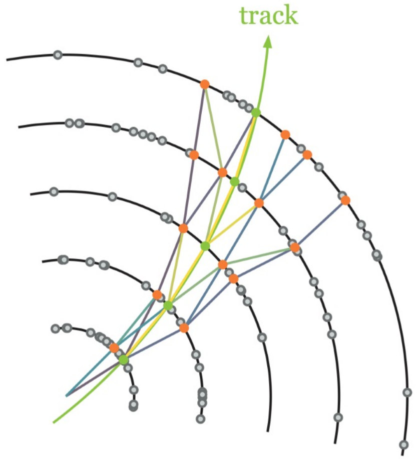
A. Salzburger, ACTS workshop, Nov 2023

Problems we had in 2023:

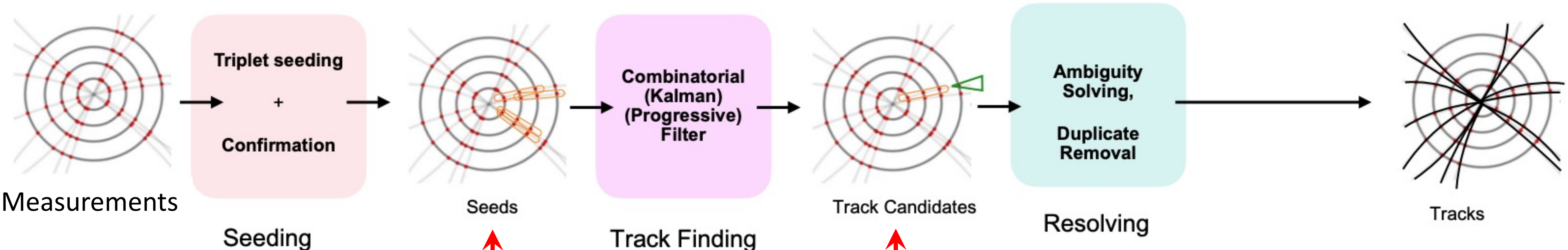
1. Multiple track seeds per particle
(a seed = triplet of points)

2. Combinatorial Kalman Filter → *branching*
(accepting multiple hits on a layer within a χ^2 window)





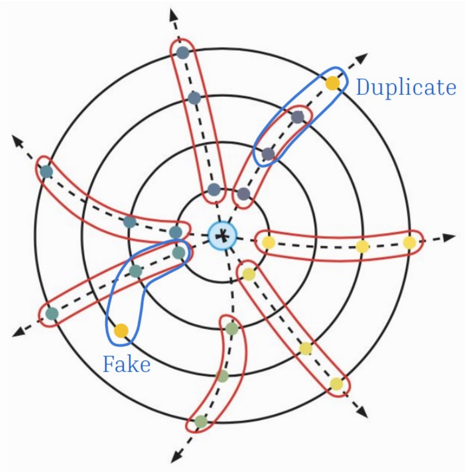
Reconstruction: **the ”full chain”**



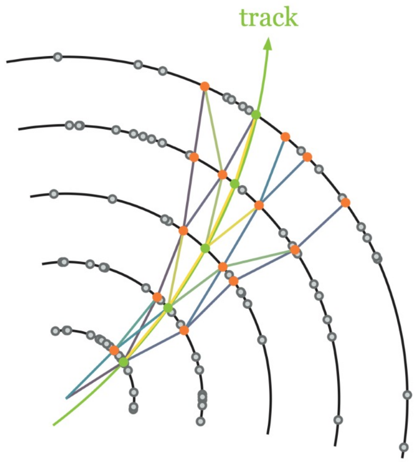
[A. Salzburger, ACTS workshop, Nov 2023]

Problems we had in 2023:

1. <u>Multiple</u> track seeds per particle
(a seed = triplet of points)



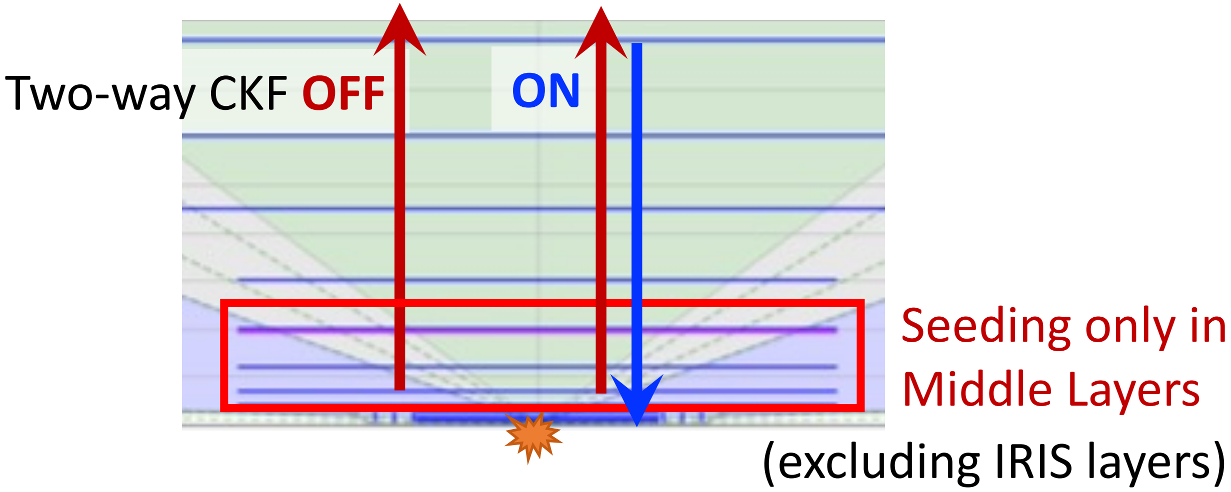
2. <u>Combinatorial</u> Kalman Filter → *branching*
(accepting multiple hits on a layer within a χ^2 window)

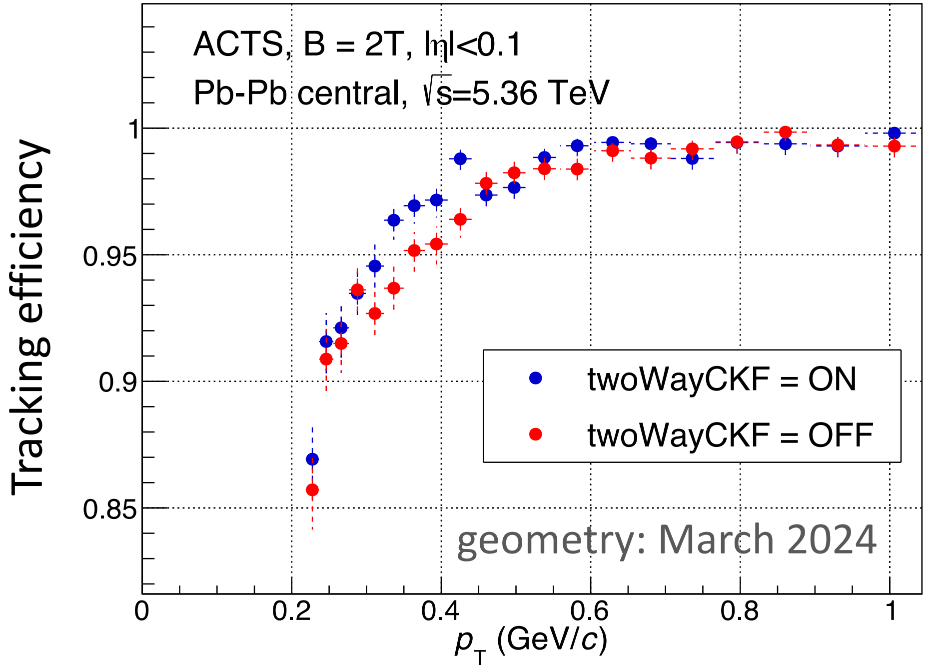


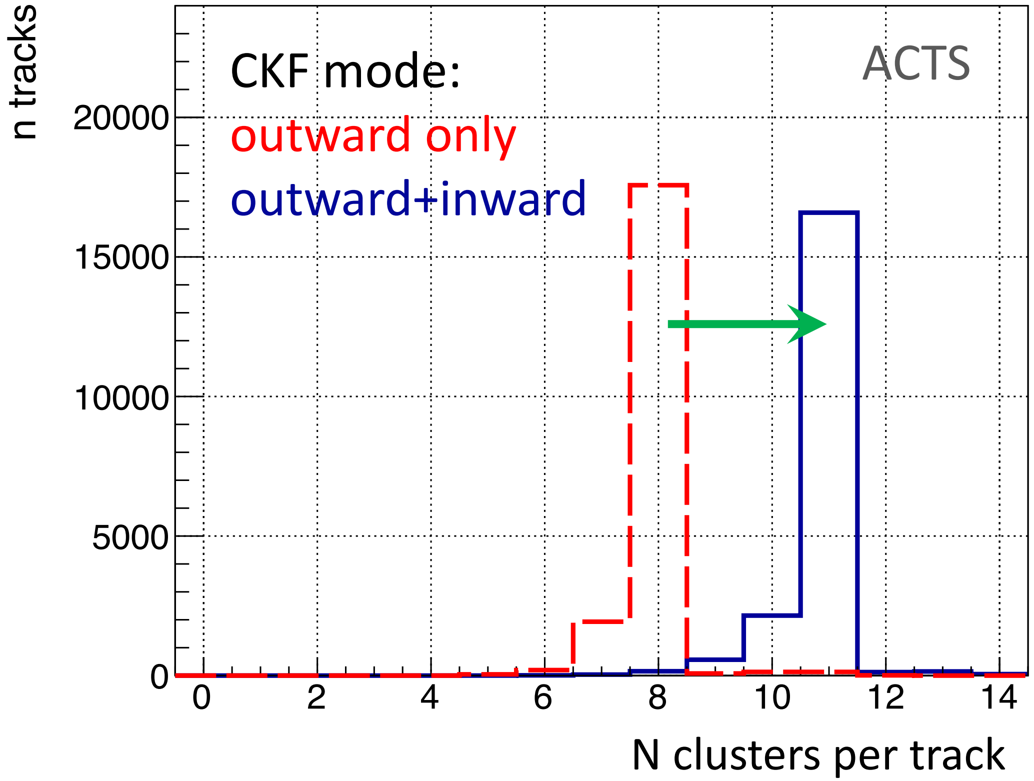
<u>ACTS updates 2024:</u>
Significant changes in CKF
In particular, two options:
1. **Seed de-duplication**
2. **Outward-inward fitting**

Current version used in this talk: **36.3**

Demonstration: outward+inward Combinatorial KF

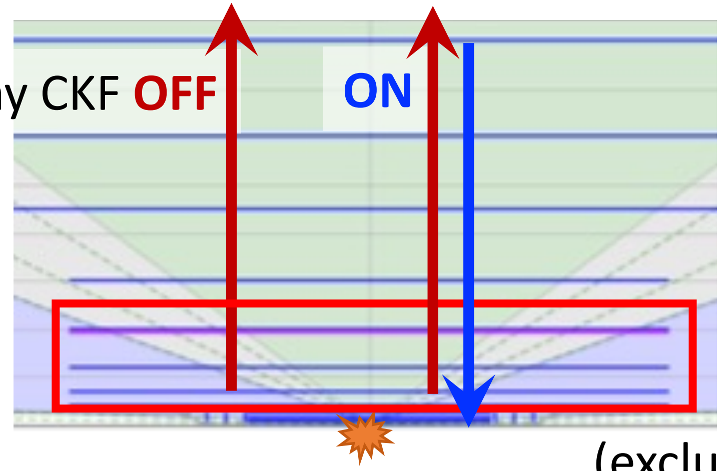






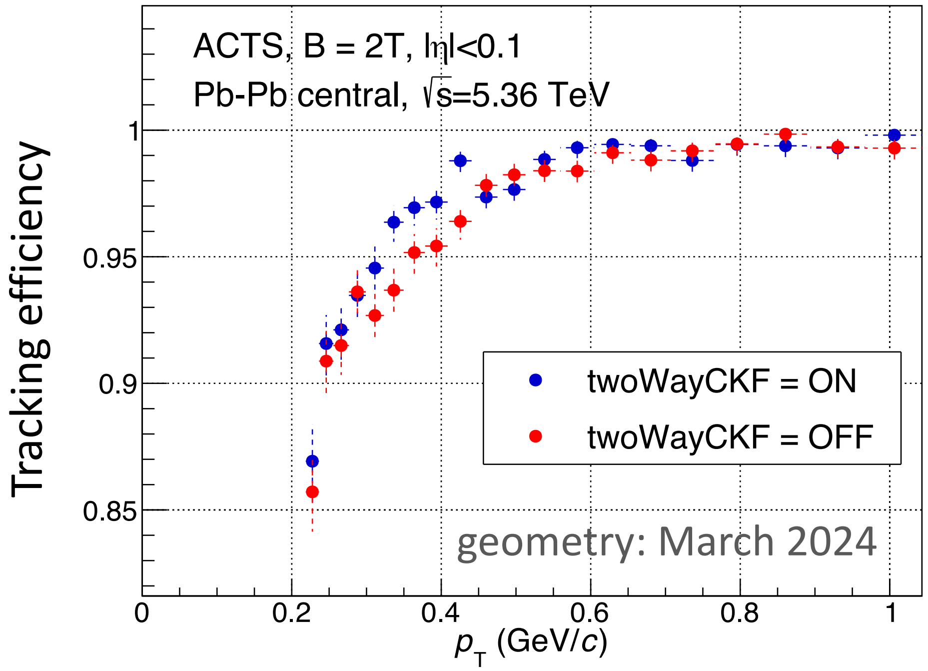
- Two-way track finding allows to **recover missing inner clusters,** improving efficiency

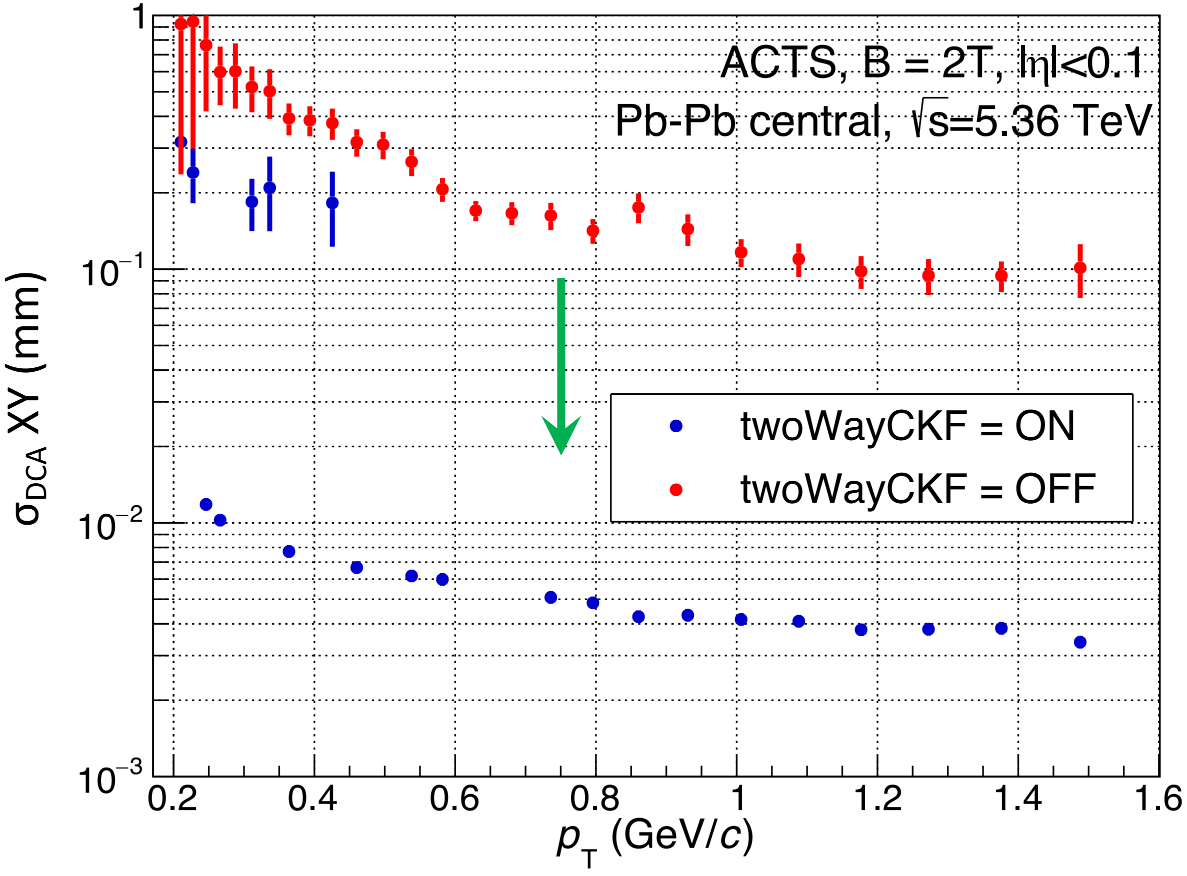
Demonstration: outward+inward Combinatorial KF



Seeding only in Middle Layers

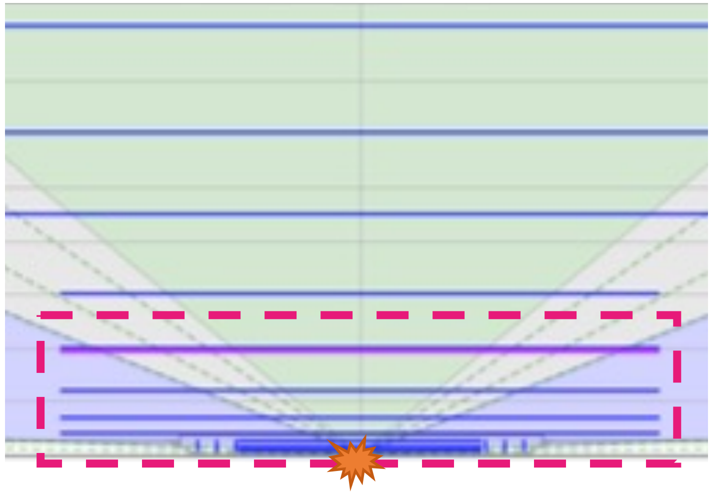
(excluding IRIS layers)



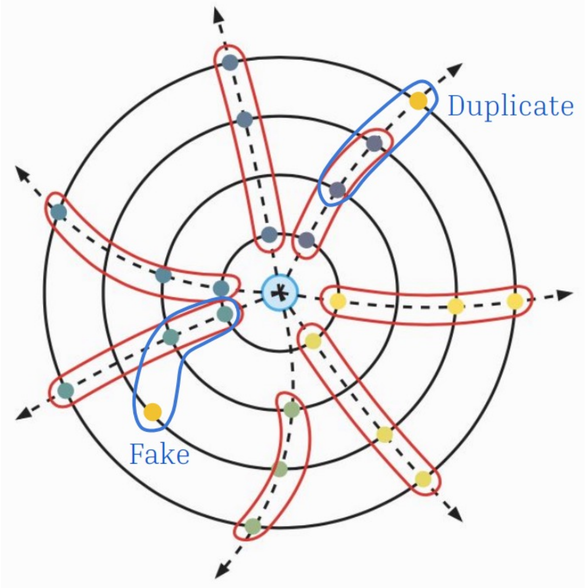


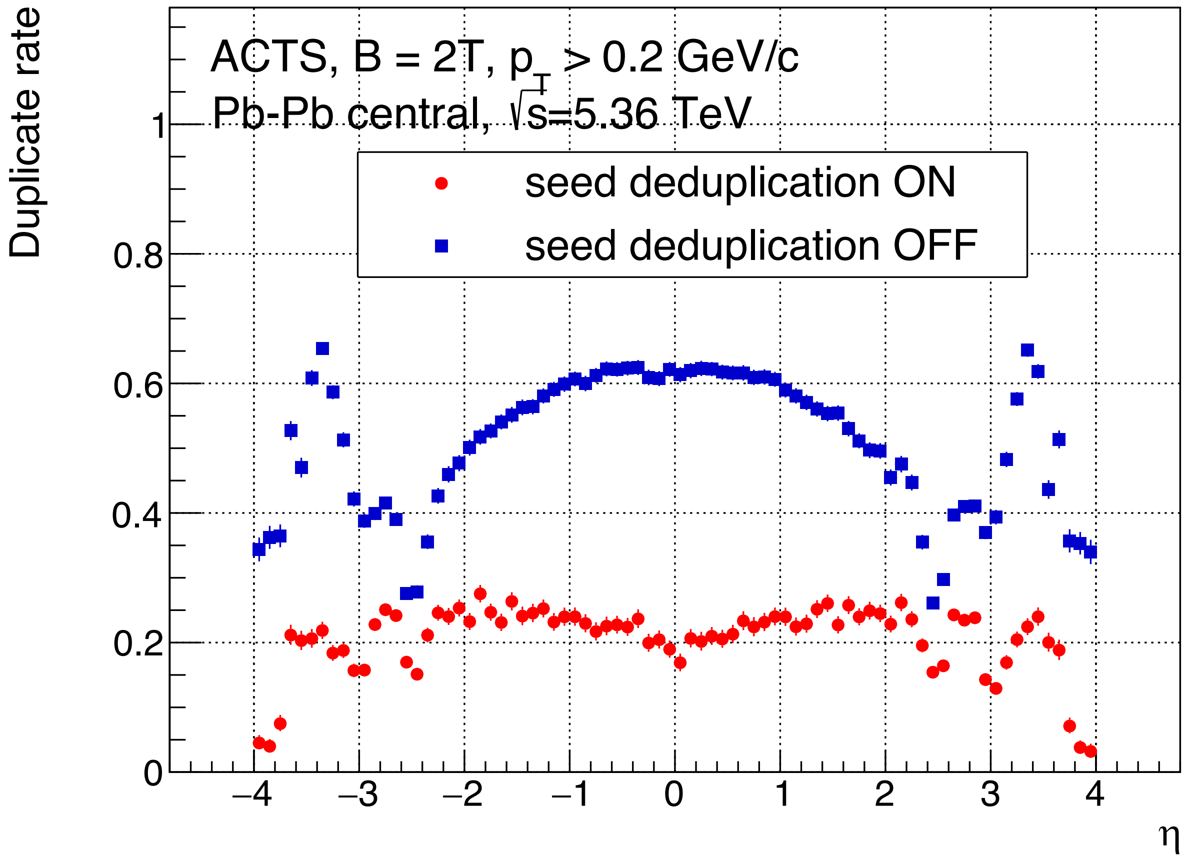
- Two-way track finding allows to **recover missing inner clusters,** improving efficiency
- Crucial also e.g. for DCA xy

Demonstration: seed de-duplication for CKF



seeding in 3 inner + 4 middle layers



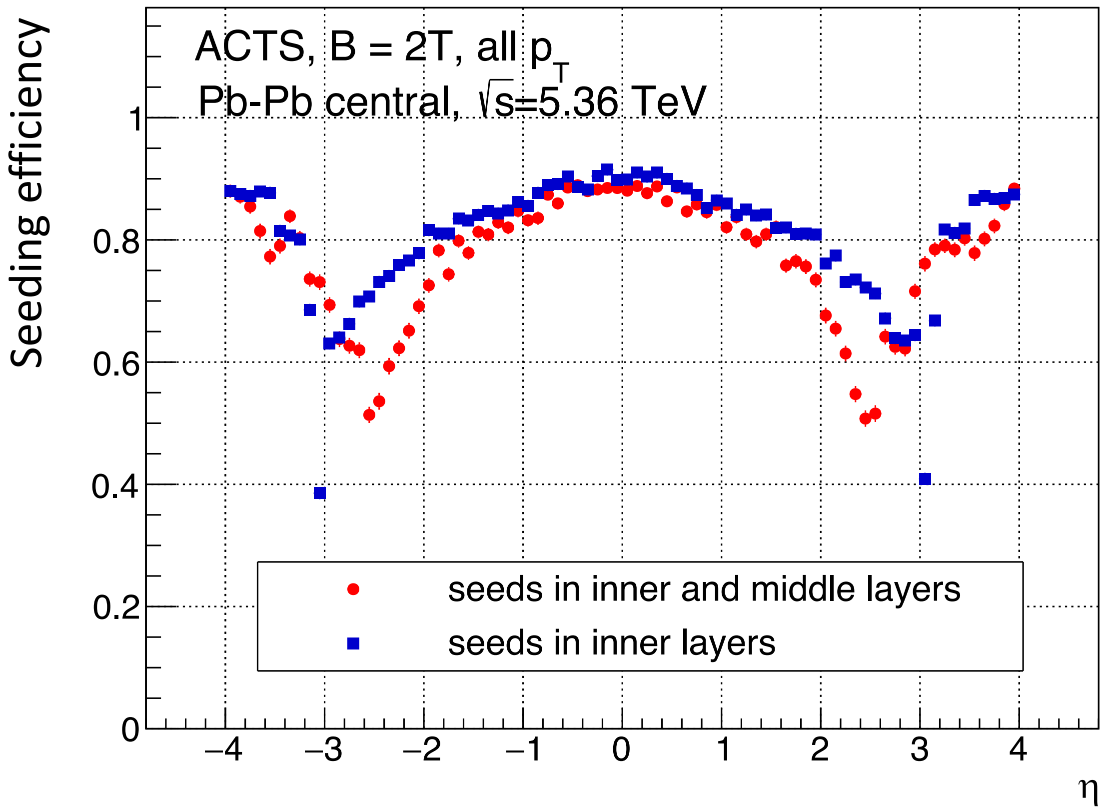


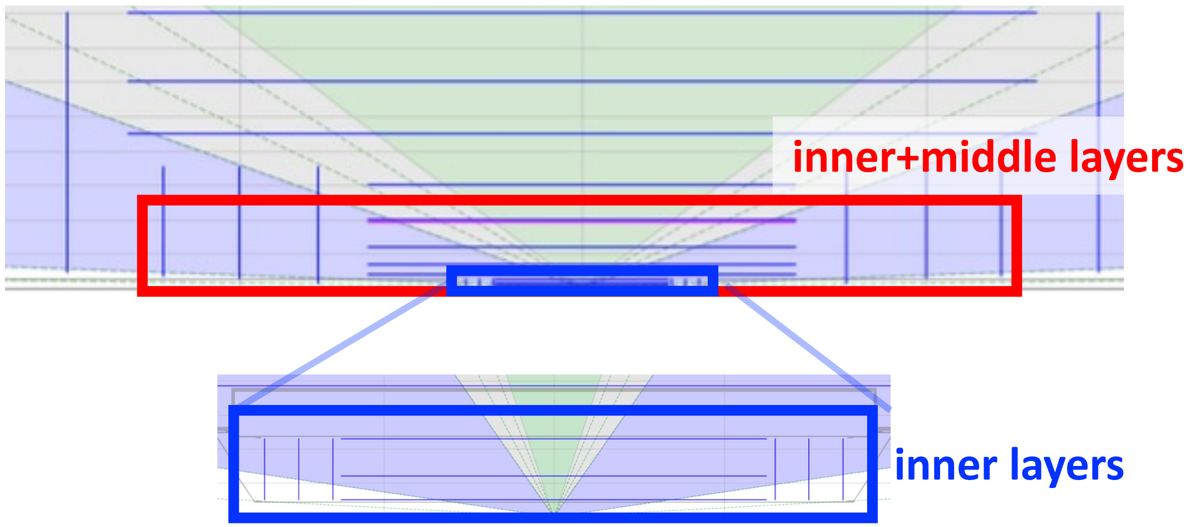
- Central Pb-Pb: ~40% faster with de-duplication=ON

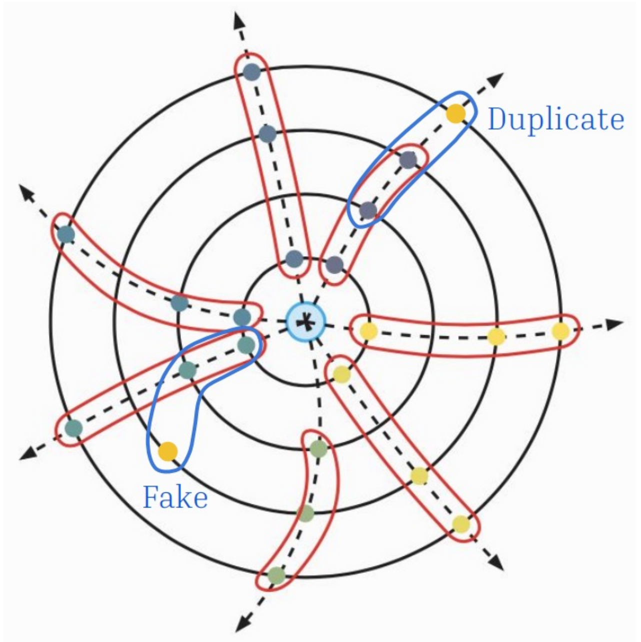
(~130 sec / ev)

TrackFinding printouts:
- total seeds: **193k**
- deduplicated seeds: **58k**

Which layers are actually optimal for track seeding?

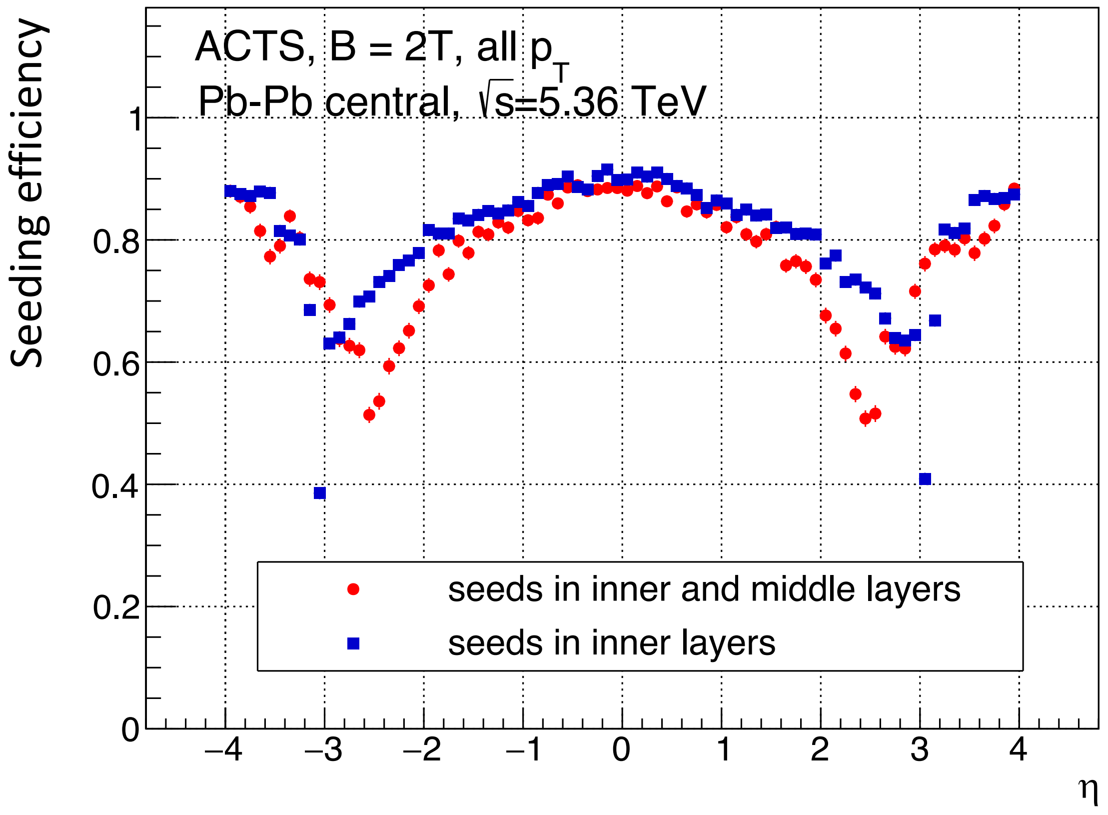


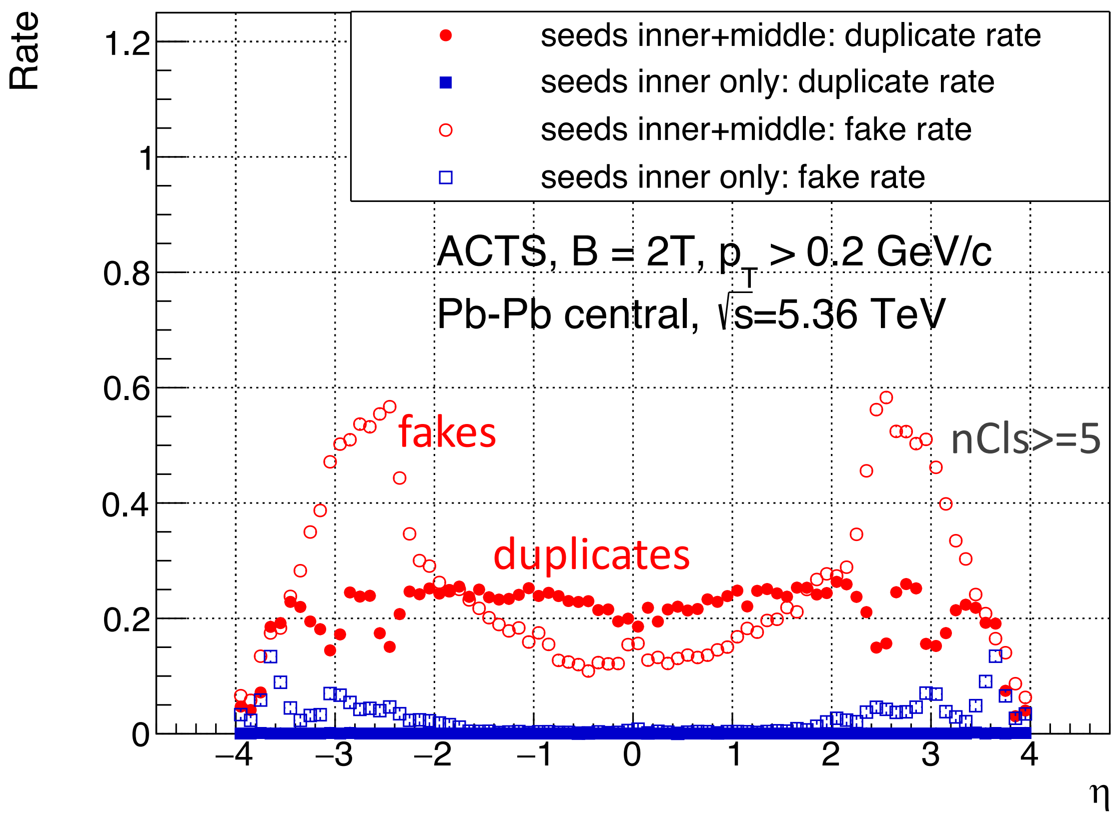




- Seeding from **inner+middle layers** gives reduced efficiencies, in contrast to seeding from **inner-only layers**

Which layers are actually optimal for track seeding?



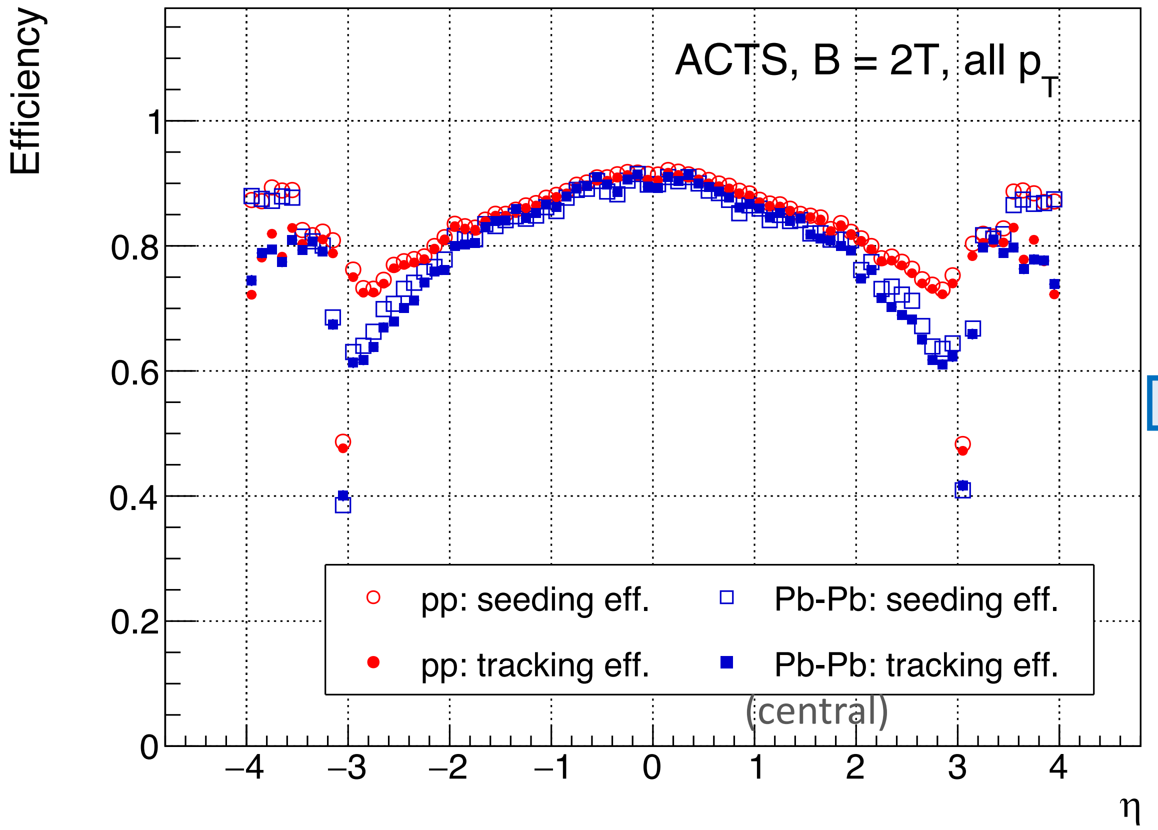


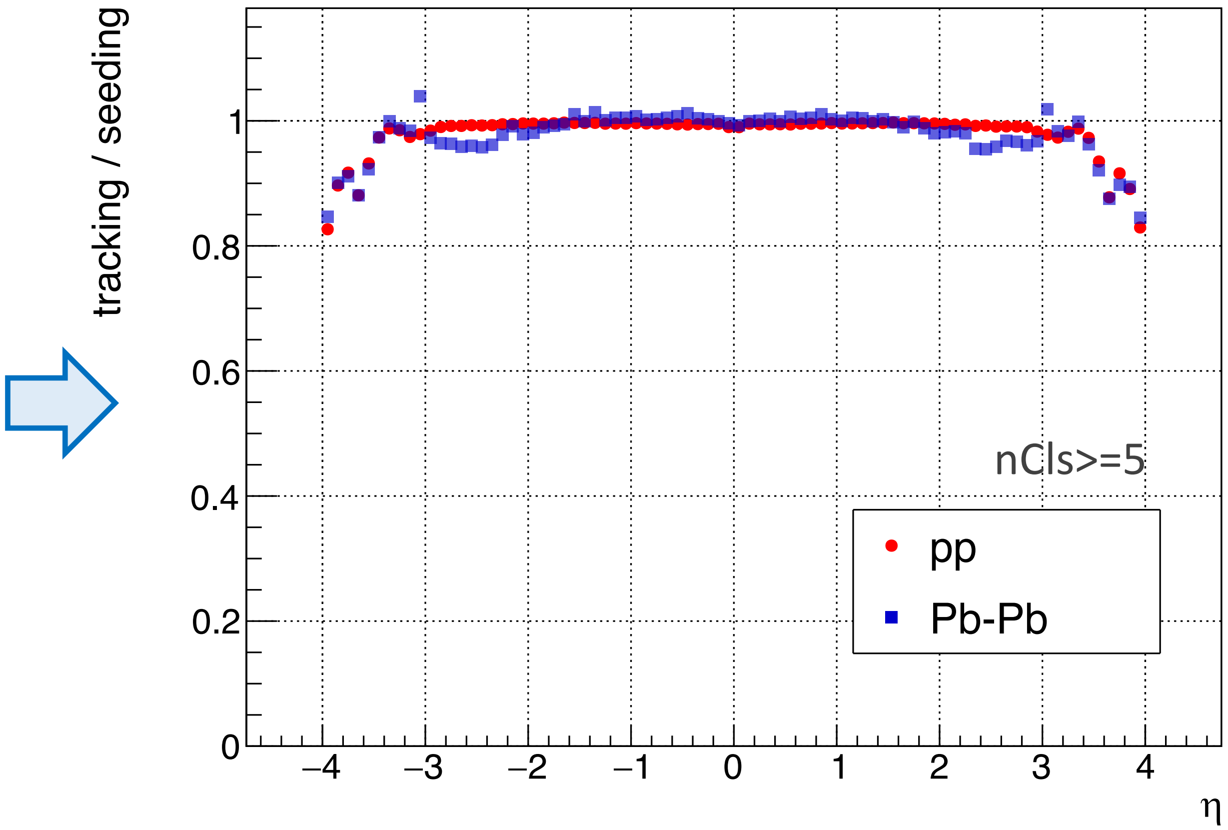
- Seeding from **inner+middle layers** gives reduced efficiencies, in contrast to seeding from **inner-only layers**

… and also increased duplication and fake track rates…

twoWay=On
seedDeduplication=On

Seeding *vs* tracking efficiency in pp and central Pb-Pb





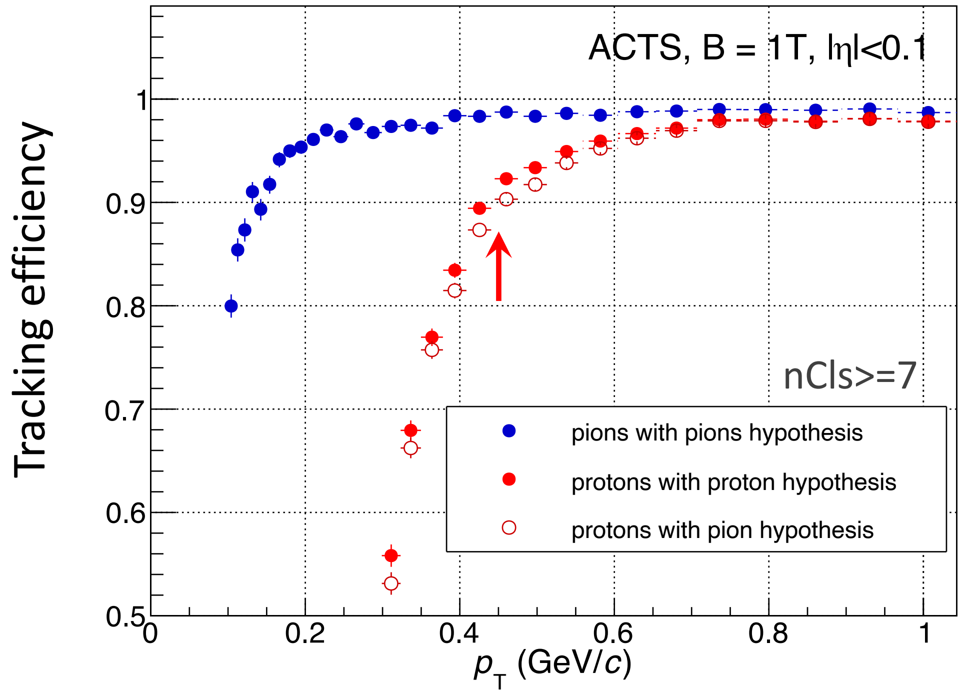
- Seeding is still the main issue for total tracking efficiency

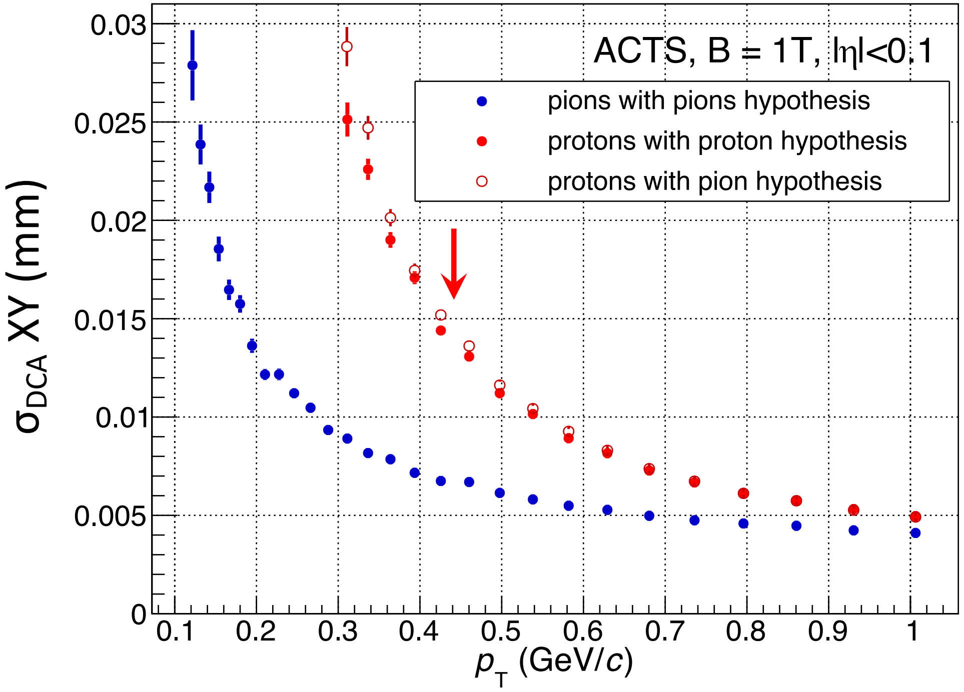
- For found seeds, **tracking efficiency is >95%** within |η|<3.5

PID hypothesis for tracking

```
245  def addSeeding(
        ...
271          particleHypothesis: Optional[
272              acts.ParticleHypothesis
273          ] = acts.ParticleHypothesis.pion,
```

- PID hypothesis in ACTS is **propagated together with track parameters from seeds to final tracks**
- **Pions are the default → try protons with proton hypothesis:**





Low p_T~0.4 GeV/c: 2-3% gain in proton efficiency

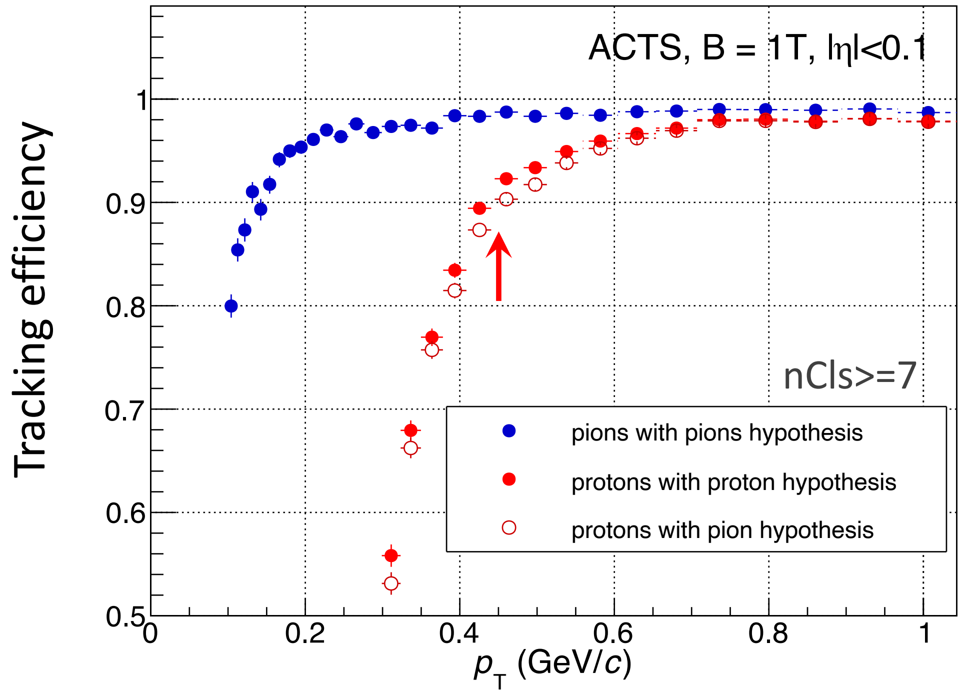
~5% better proton DCAxy

.. however, low-p_T resolution worse by ~30%.

PID hypothesis for tracking

```
245  def addSeeding(
        ...
271         particleHypothesis: Optional[
272             acts.ParticleHypothesis
273         ] = acts.ParticleHypothesis.pion,
```

- PID hypothesis in ACTS is **propagated together with track parameters from seeds to final tracks**
- **Pions are the default → try protons with proton hypothesis:**

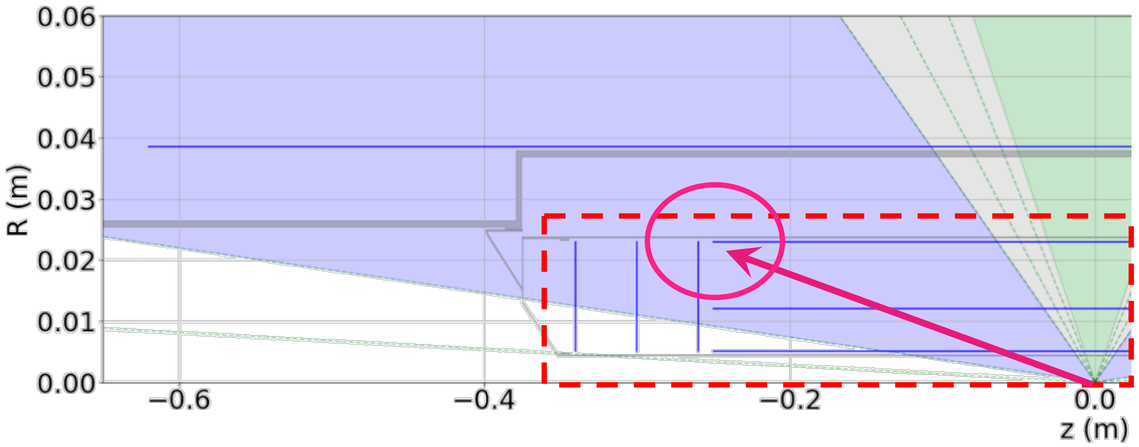


Low p_T~0.4 GeV/c: 2-3% gain in proton efficiency

Can we assign PID hypotheses for each track?

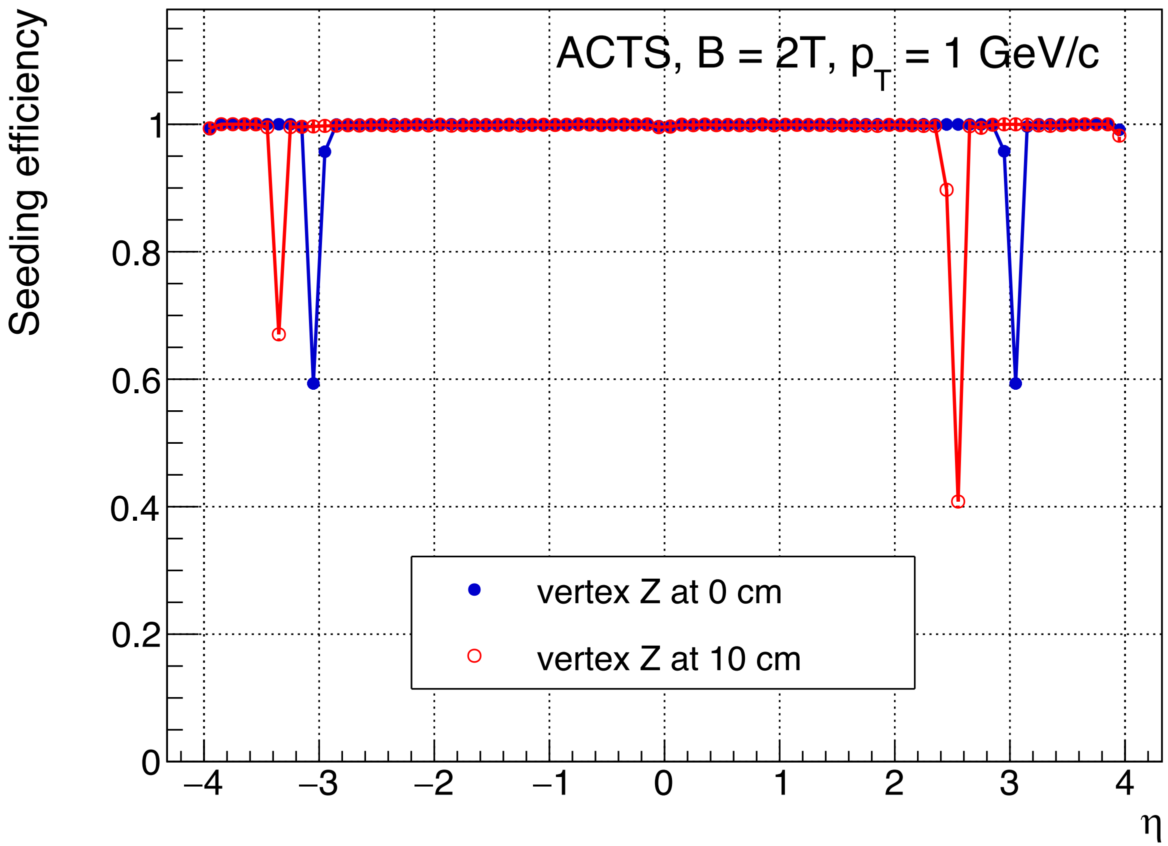
- One would have to hack the code…
  - e.g. assume pions and afterwards refit the found track using PID from TOF
- Somewhere here we hit the boundary of the current ACTS Examples…
  - The usual argument is that experiments will have to come up with their own framework if it gets too complicated.

Last (geometrical) note about seeding



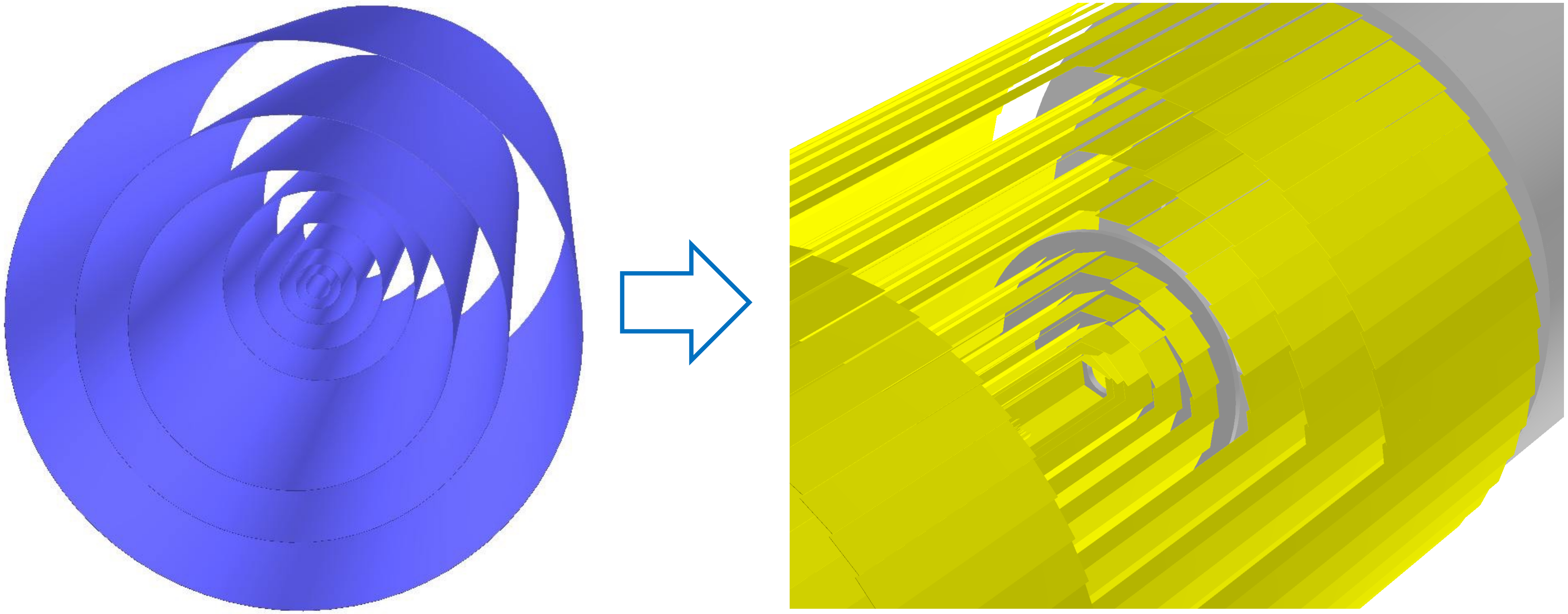
- If seeding is done in innermost layers (to have 3-point "tracklets"):

a small gap b/n Barrel and Endcap influence the performance



Segmentation of Middle and Outer tracker barrel

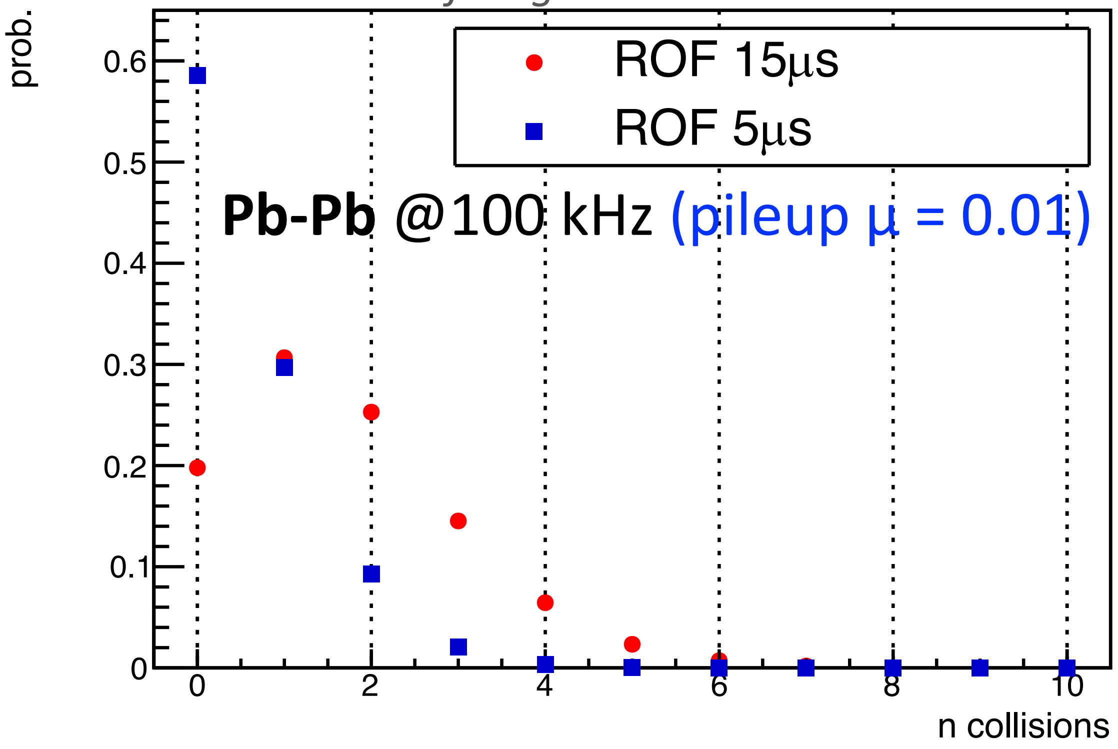
Daniel Battistini
Matteo Concas

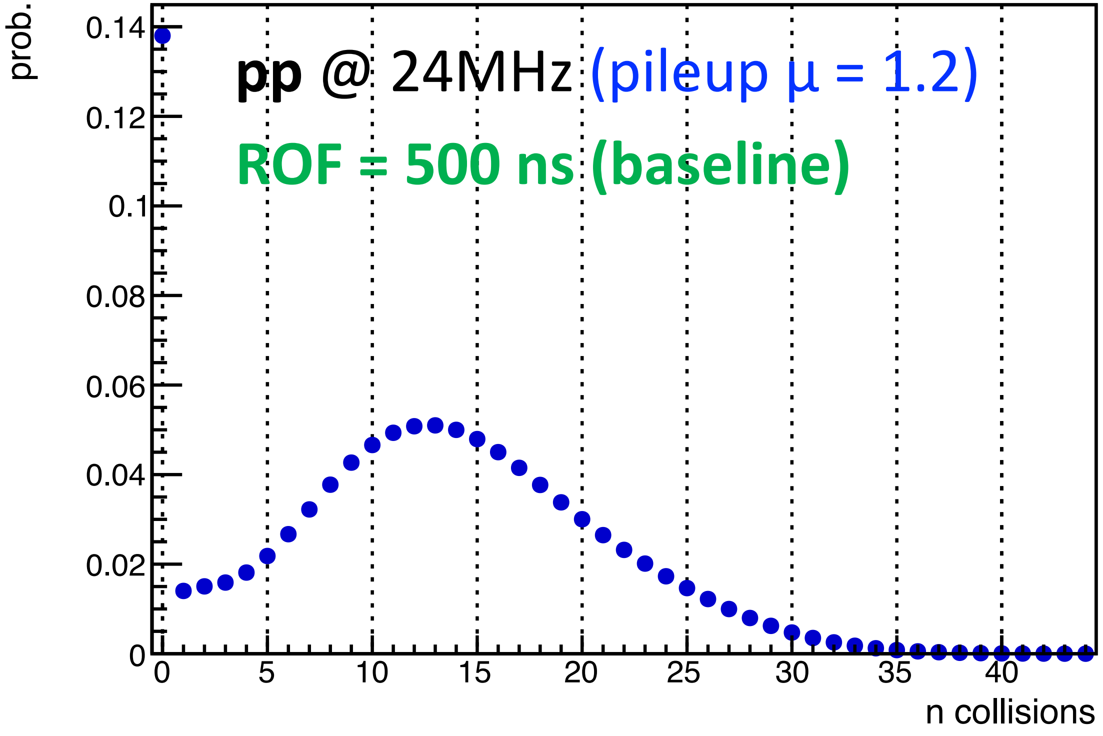


- Implementation of a realistic segmentation in the geometry of the detector is in progress
- **To be tried with ACTS** (currently – simple cylinders in barrel)

Number of collisions in Tracker Readout frames

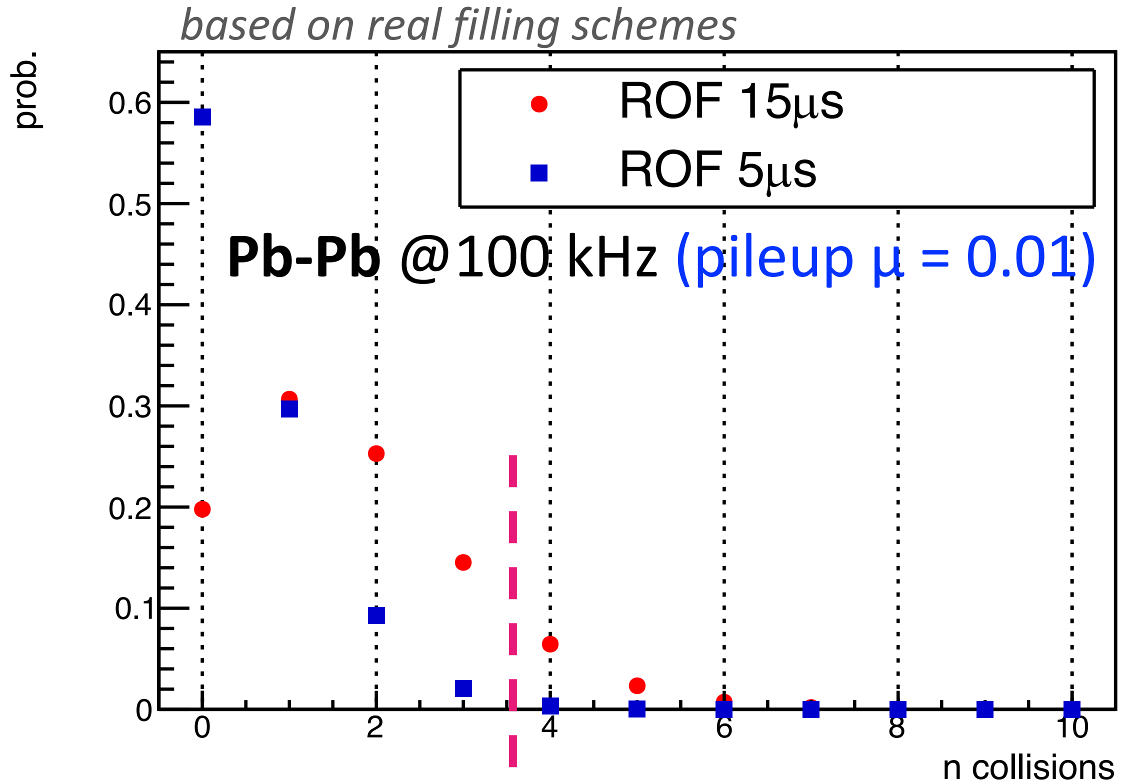
based on real filling schemes





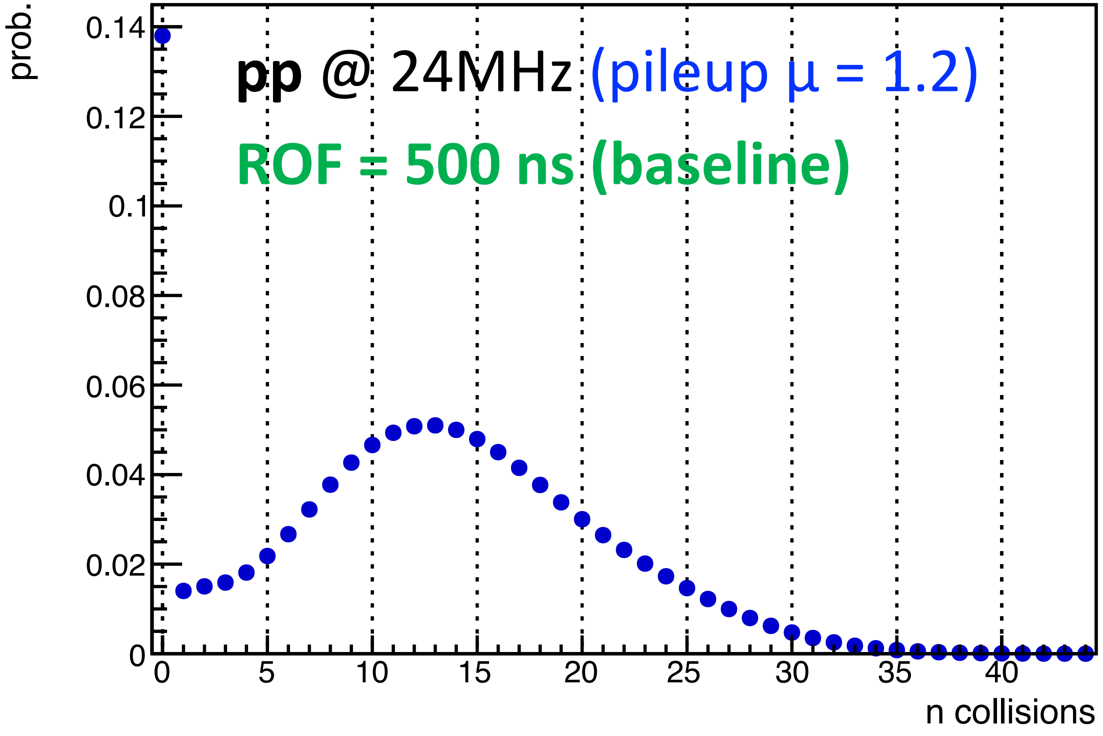
- Currently, ITS2 ROF in Pb-Pb is 15 μs
- **Can we keep the same <u>15 μs ROFs for Pb-Pb</u> in ALICE3?**

Number of collisions in Tracker Readout frames



ROF = 15 μs:

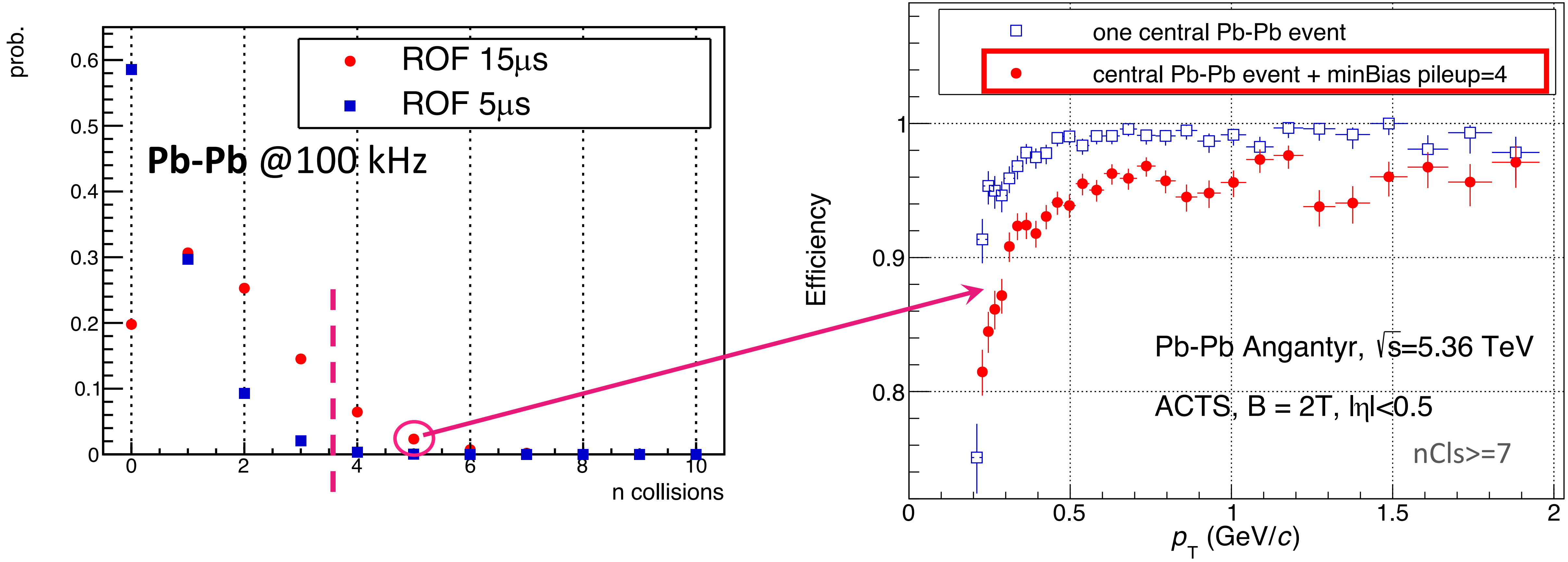
in 10% cases, >=4 coll. per ROF



- Currently, ITS2 ROF in Pb-Pb is 15 μs
- **Can we keep the same <u>15 μs ROFs for Pb-Pb</u> in ALICE3?**

→ Try tracking (with ACTS)

Tracking Efficiency for central Pb-Pb + pileup



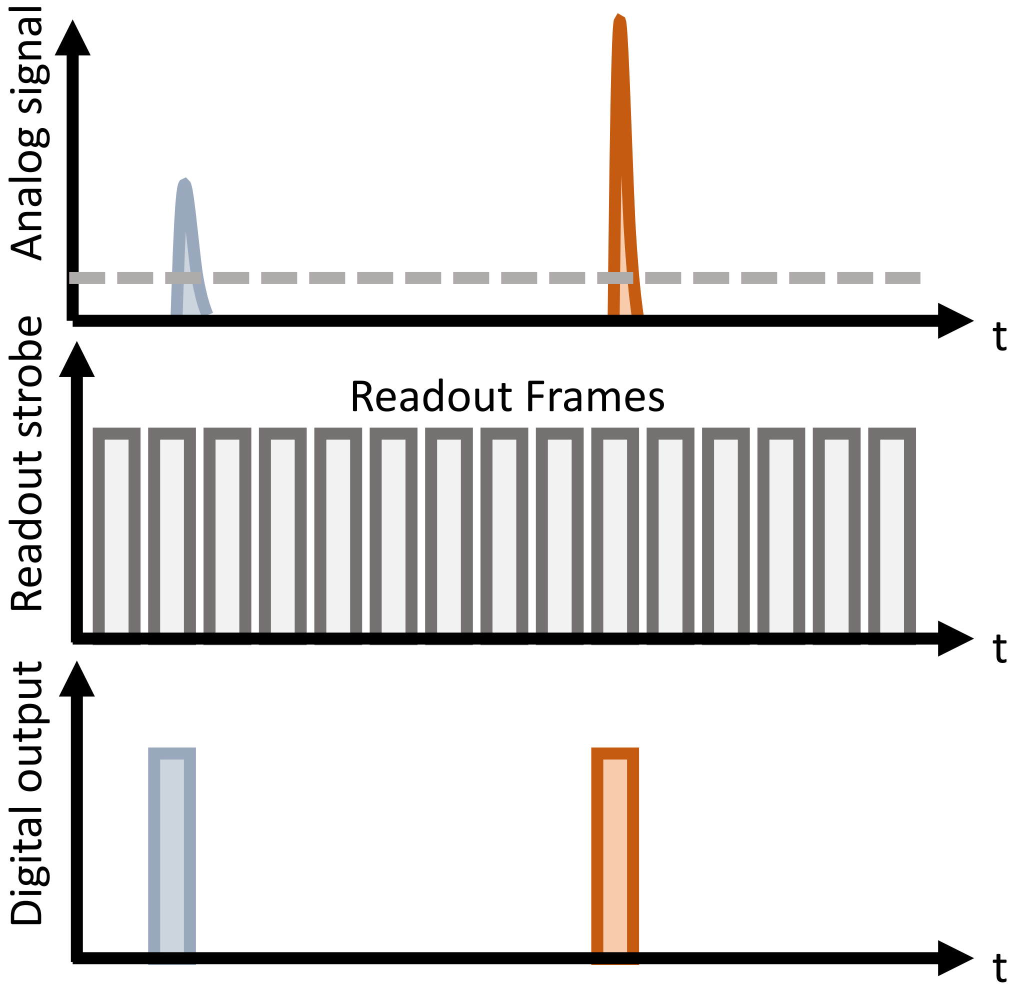
- Visible drop of the tracking efficiency with high in-ROF pileup
  - yet another thing to optimize

Amplitude measurement via Time-Over-Threshold

Berkin Ulukutlu
Henrik Fribert

- Similar to color runs in ITS2 (see slide 18 in ALICE week talk)
  - Over sampling the signal → measure same hit multiple times
  - Signal decay time (ToT) depends on signal amplitude (deposited charge)

Standard operation mode



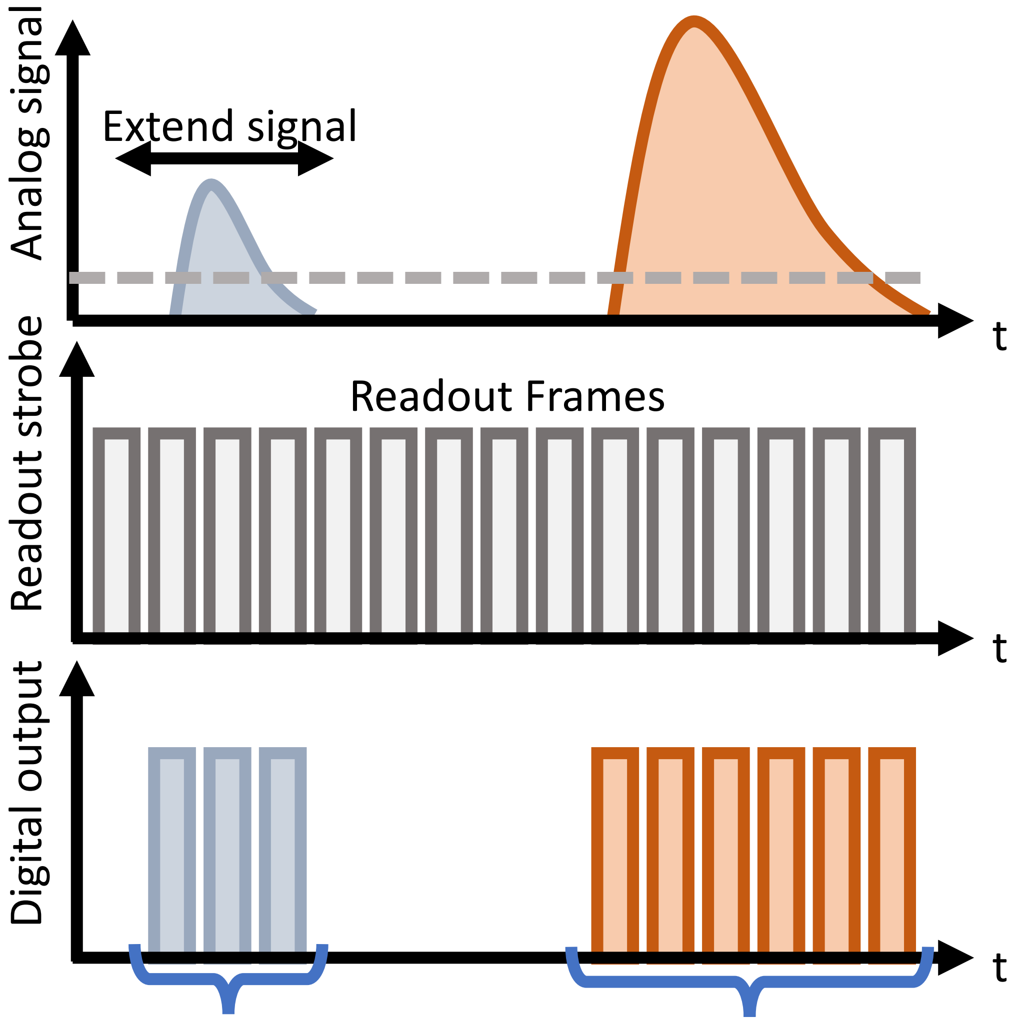
Only hit/ no-hit info in output

Amplitude measurement via Time-Over-Threshold

Berkin Ulukutlu
Henrik Fribert

- Similar to color runs in ITS2 (see slide 18 in ALICE week talk)
  - Over sampling the signal
    → measure same hit multiple times
  - Signal decay time (ToT) depends on signal amplitude (deposited charge)
  - Data rate becomes too high to be feasible

ToT operation mode



Amplitude dependent output
But data rate too high!

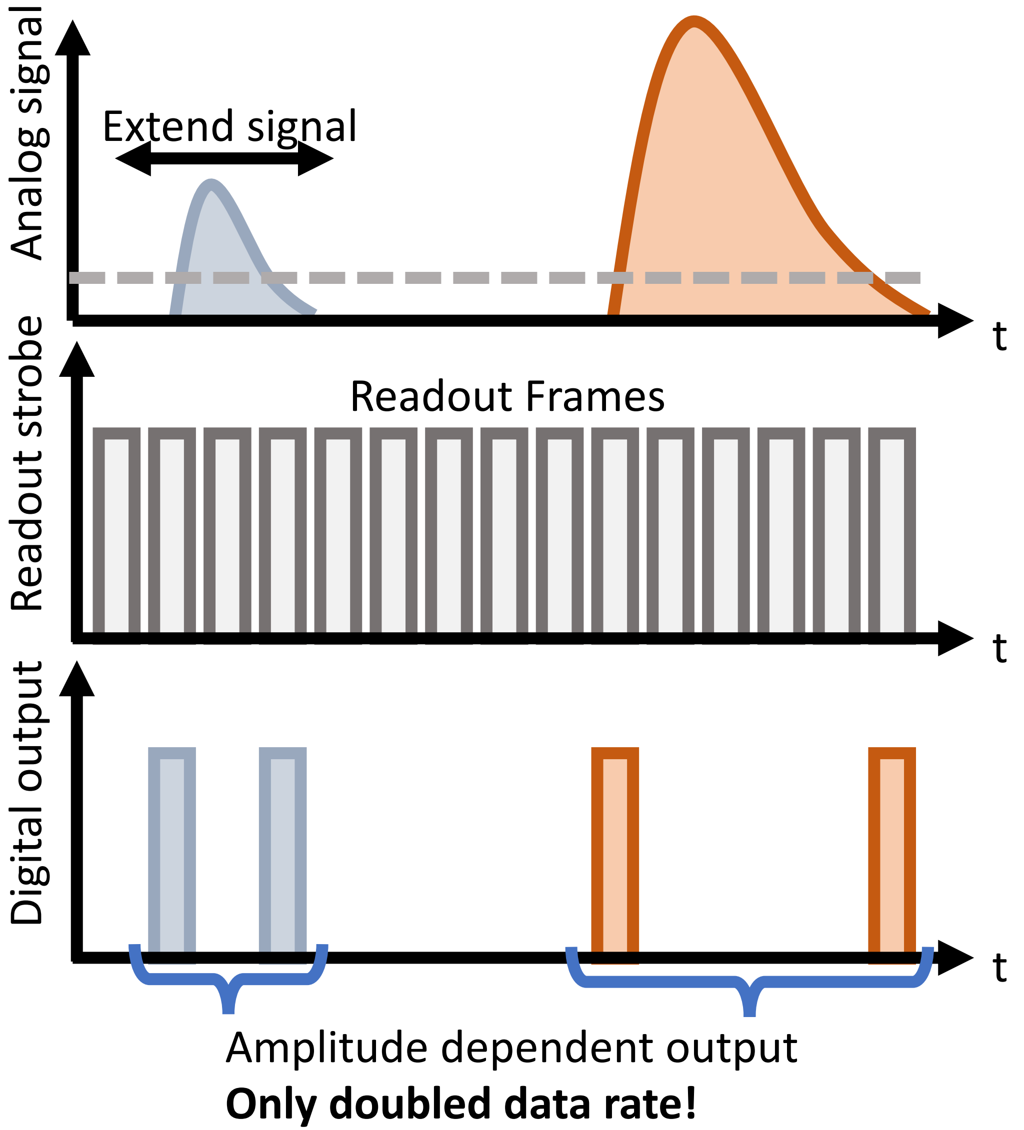
Amplitude measurement via Time-Over-Threshold

Berkin Ulukutlu
Henrik Fribert

- Similar to color runs in ITS2 (see slide 18 in ALICE week talk)
  - Over sampling the signal
    → measure same hit multiple times
  - Signal decay time (ToT) depends on signal amplitude (deposited charge)
  - Data rate becomes too high to be feasible
- Front-end sensitive to both rising and falling edge
  → data rate only doubled from standard operation mode

ALICE3: use the time difference for PID?..

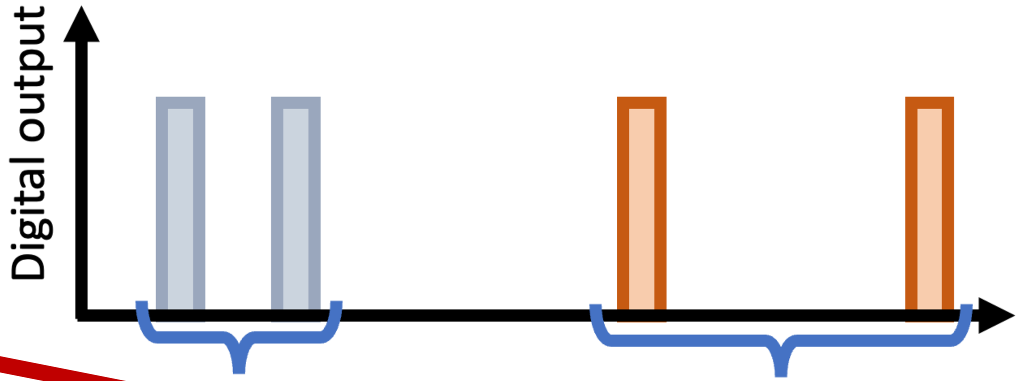
ToT operation mode



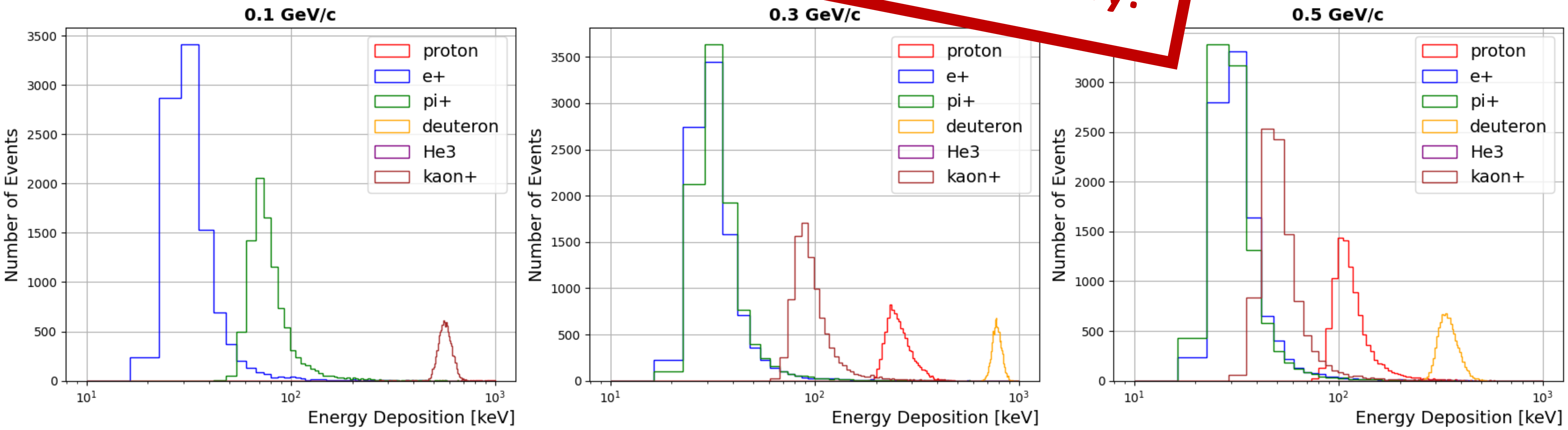
PID Performance estimation

Berkin Ulukutlu
Henrik Fribert

- Average energy loss signal measured over 11 hits
- Separation of protons up to ~500 MeV/c particle momentum

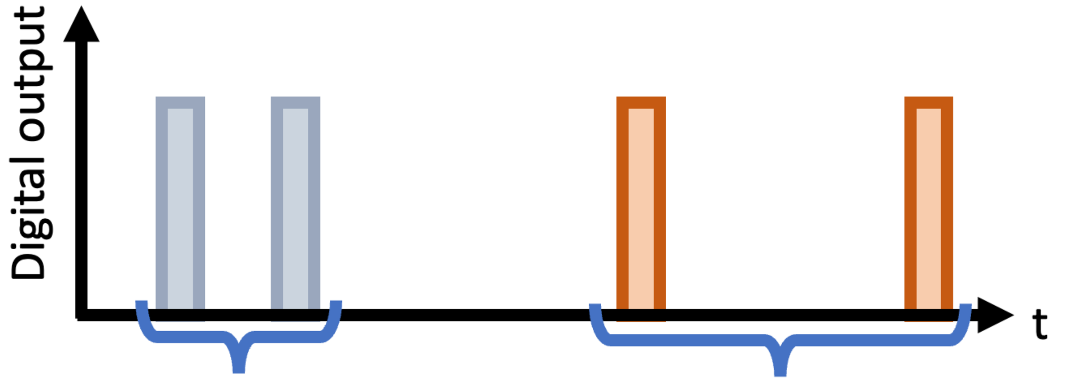


Very preliminary!



PID Performance estimation

Berkin Ulukutlu
Henrik Fribert



Potential use cases

- Improved PID for low p_T tracks and nuclei
- Seeding and ambiguity resolving
- Fake-hit rejection via ToT cuts
  - Noise hits do not lead to long ToT signal
  - Effect on efficiency needs to be studied for different configurations

Challenges

- ToT measurement not applicable for high pixel occupancies
- Complication of tracking due to ”phantom” hits from ToT-end signal
  - Introducing an additional bit (rising/falling edge) might be necessary

Concept to be proved

Outlook

- Introducing ToT measurement into ACTS to estimate impact on tracking
- More detailed Geant4 simulations for differential PID performance based on ToT-resolution

Summary

- Important tracking improvements in ACTS, including:
  - outward+inward tracking, seed deduplication
  - now we can fairly well operate at multiplicities of central Pb-Pb
- If continue using ACTS, we will soon need to customize it for ALICE3:
  - use our own seeding (CA)? (via “code intervention” or via plug-in mechanism)
  - PID hypothesis, etc.
- Segmentation of layers is added to the geometry → to be tried with ACTS
- ROF size for Pb-Pb – first performance checks
- PID using time difference b/n rising and falling edges of amplitudes? → concept to be proved

Backup

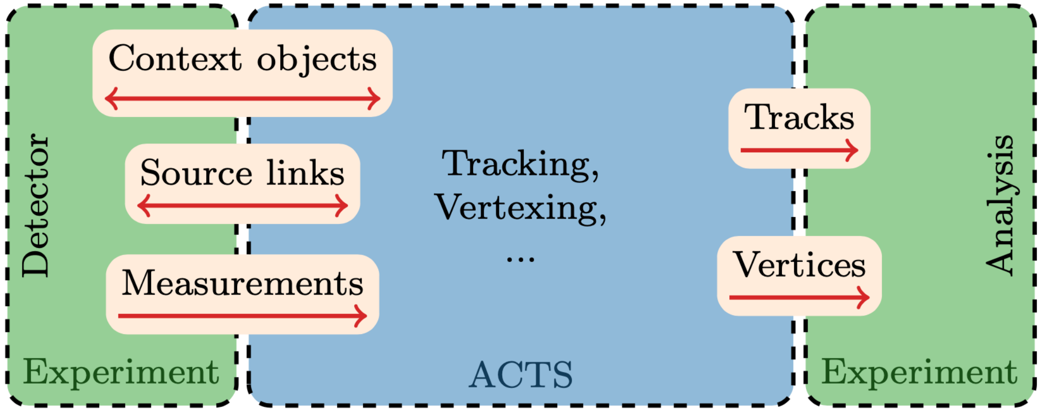
A Common Tracking Software Project

Overview:
Comp. and Soft. for Big Science, 6, 8 (2022)

Implementation for sPHENIX
Comp. and Soft. for Big Science, 5, 23 (2021)

github , acts_alice3

Example of integration into experiment's framework:



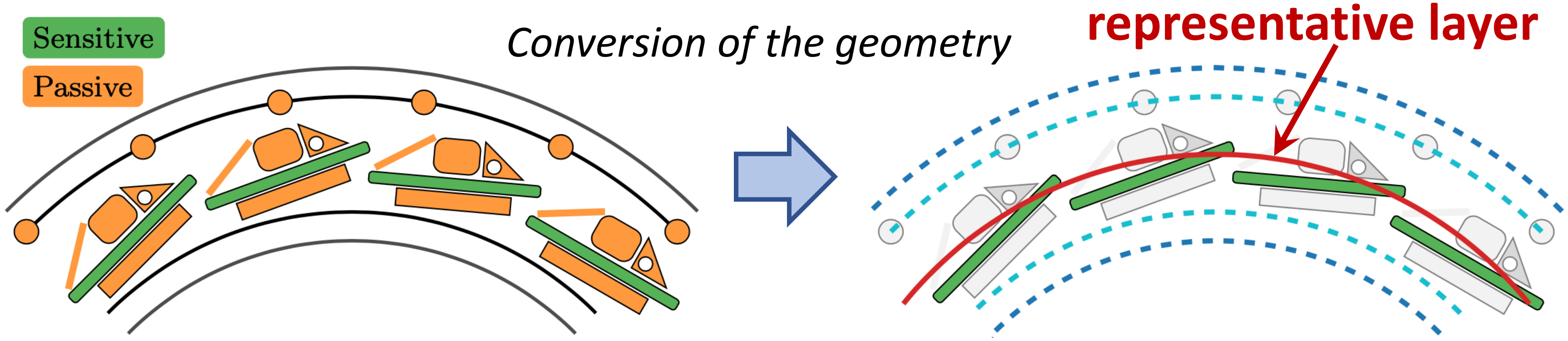
Full chain:

[*sim*] gen. particles → sim. hits → digitization

[*rec*] → seeding → track finder→ track ambiguity resolver

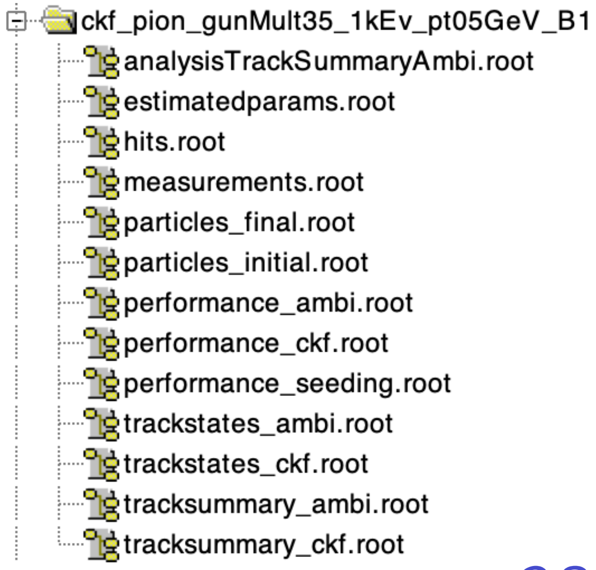
- can be run as a simple Python script

Conversion of the geometry

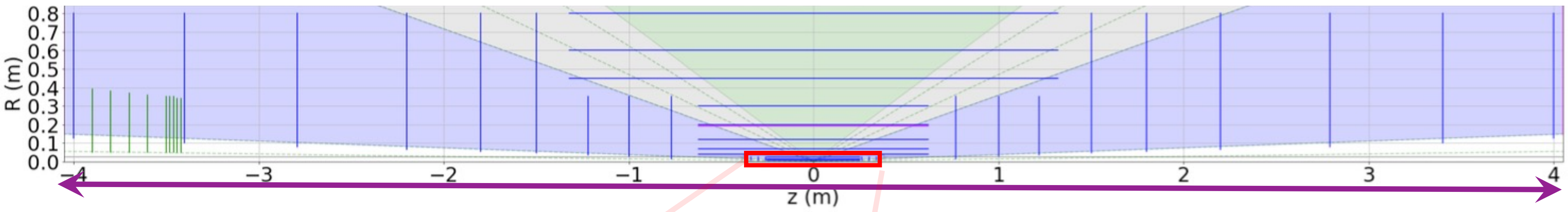


Full geometry (GEANT) ACTS tracking geometry

details: link



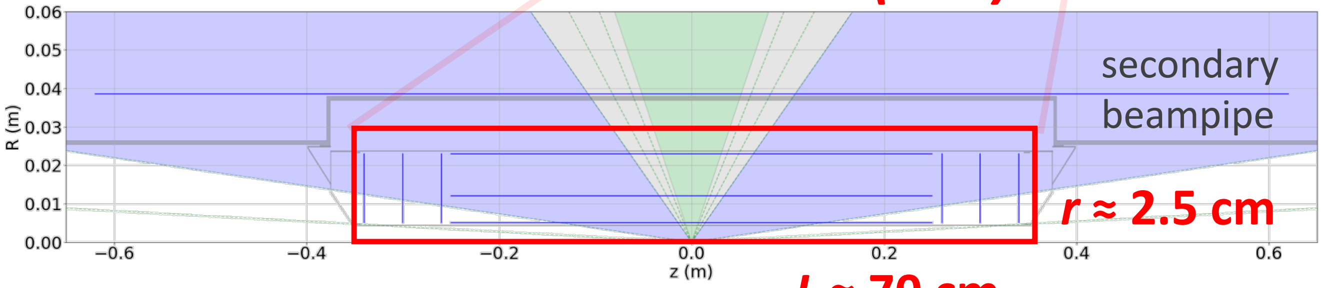
ALICE 3 LoI – arxiv.2211.02491



R = 0.8 m

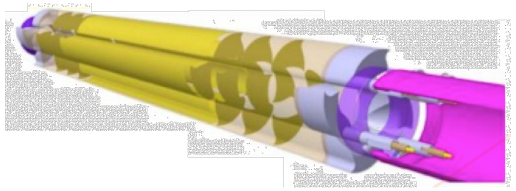
L = 8 m

Vertex detector (IRIS)



r ≈ 2.5 cm

L ≈ 70 cm



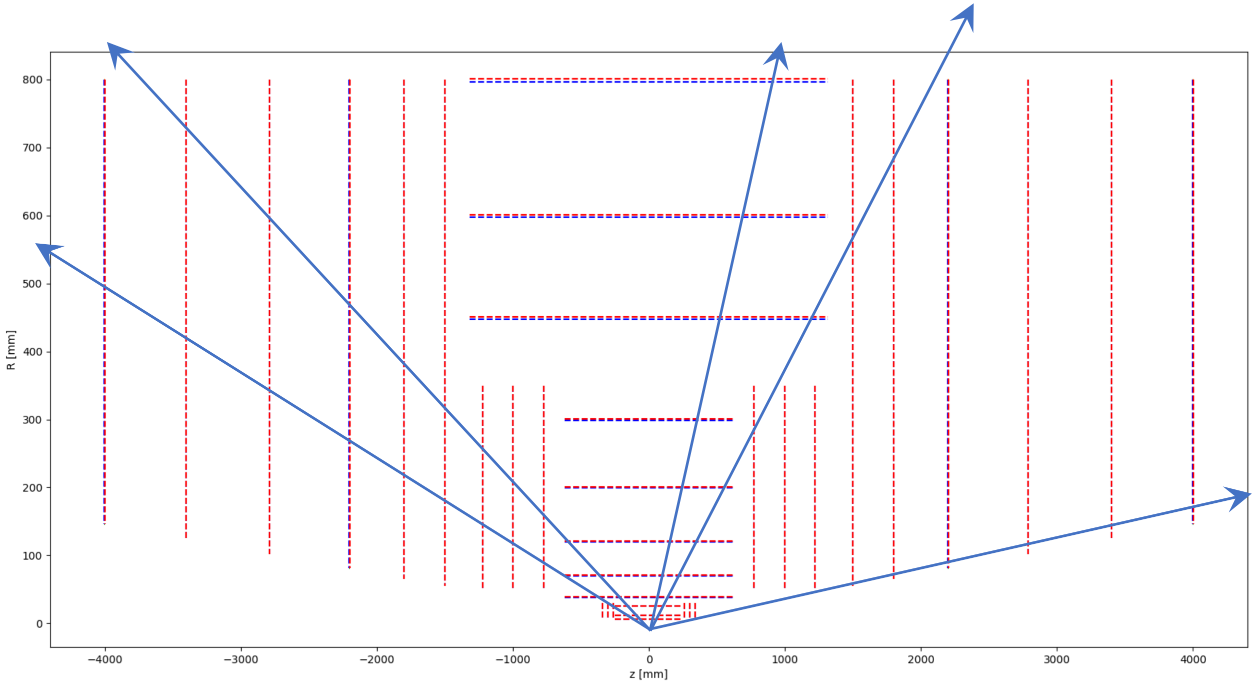
first layer at midrapidity:
5 mm from beam!

Layer	Material thickness ($\%X_0$)	Intrinsic resolution (µm)	Barrel layers: Length ($\pm z$) (cm)	Barrel layers: Radius (r) (cm)	Forward discs: Position ($\|z\|$) (cm)	Forward discs: R_{in} (cm)	Forward discs: R_{out} (cm)
0	0.1	2.5	50	0.50	26	0.50	3
1	0.1	2.5	50	1.20	30	0.50	3
2	0.1	2.5	50	2.50	34	0.50	3
3	1	10	124	3.75	77	5	35
4	1	10	124	7	100	5	35
5	1	10	124	12	122	5	35
6	1	10	124	20	150	5	80
7	1	10	124	30	180	5	80
8	1	10	264	45	220	5	80
9	1	10	264	60	279	5	80
10	1	10	264	80	340	5	80
11	1				400	5	80

Table 8: Geometry and key specifications of the tracker.

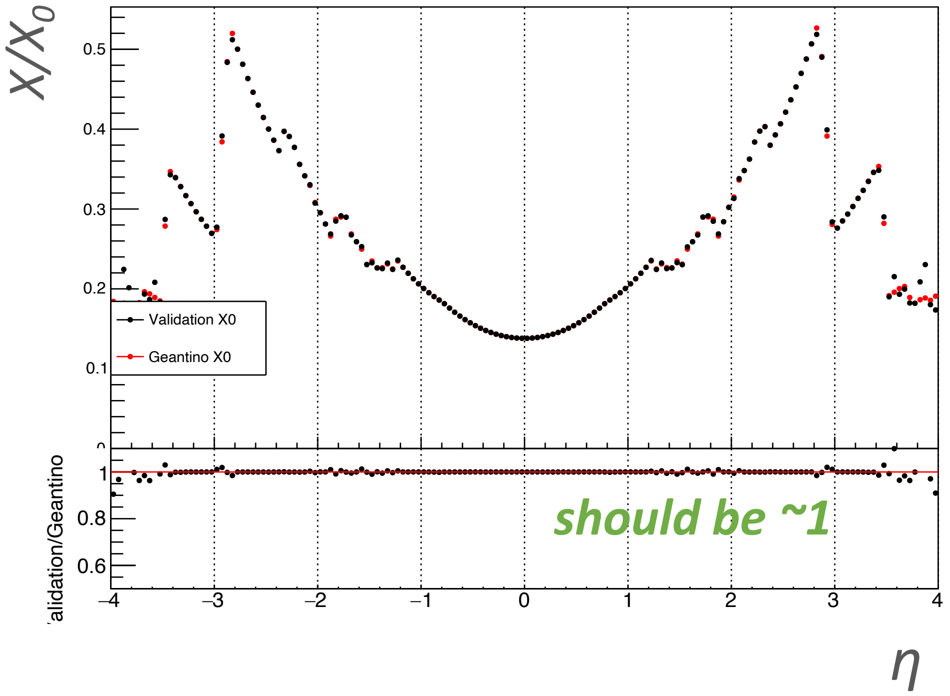
How to prepare a layout?
2) Do "material mapping"

- **"scan"** geometry with a particle gun with GEANT4,
- **project material** on pre-defined surfaces

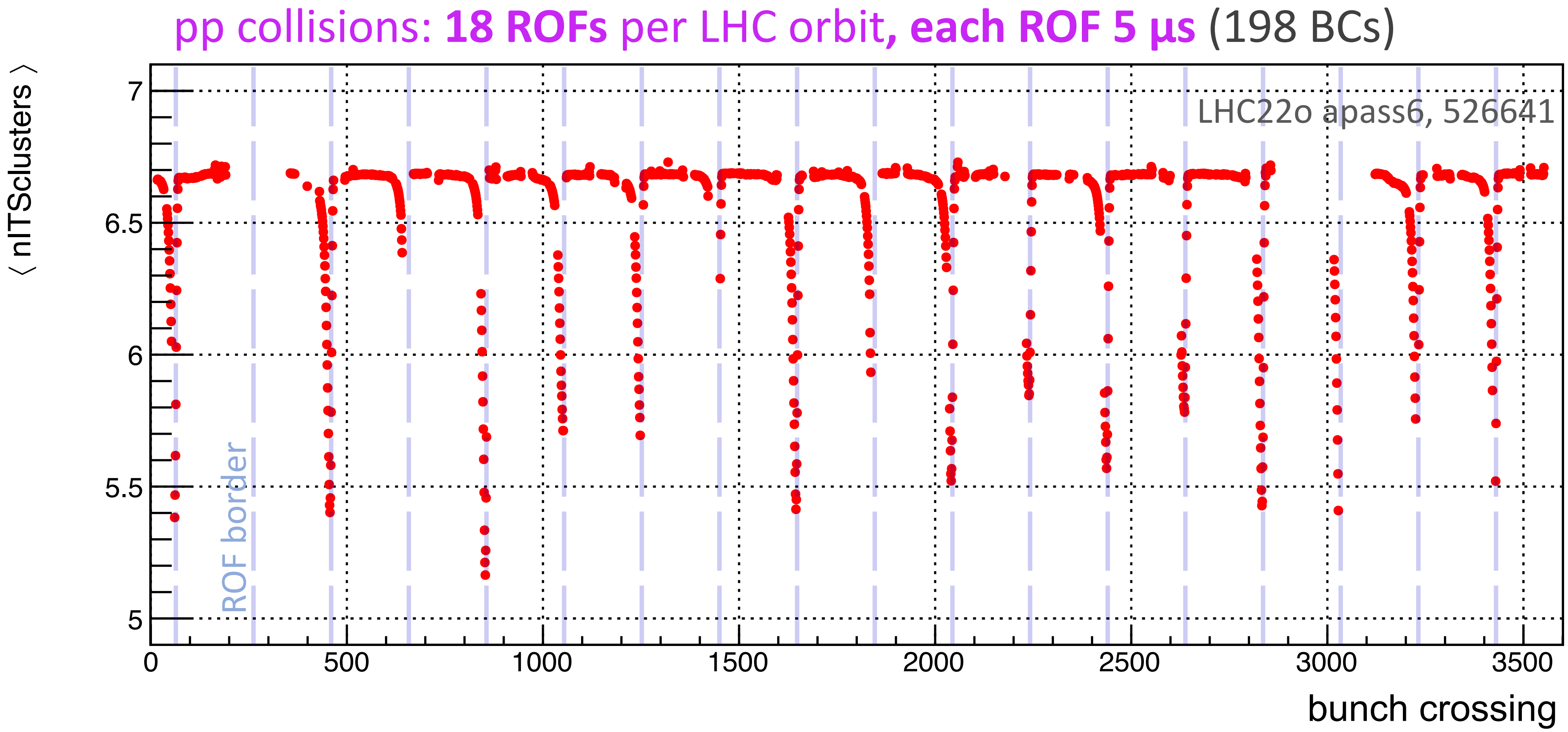


→ *.json file*

QA: GEANT4 *vs* "mapped"

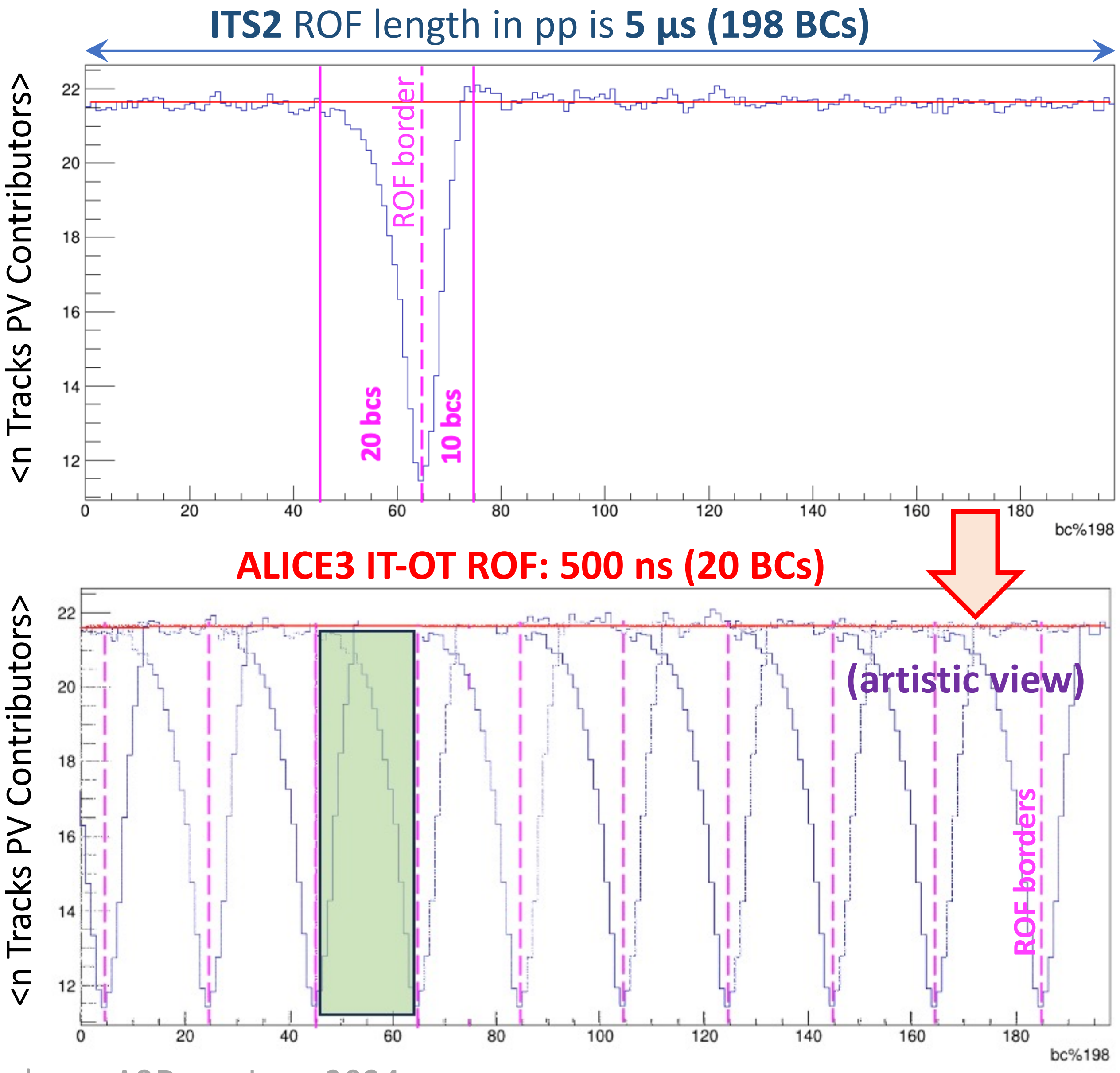


ITS2 Readout Frames in Run 3



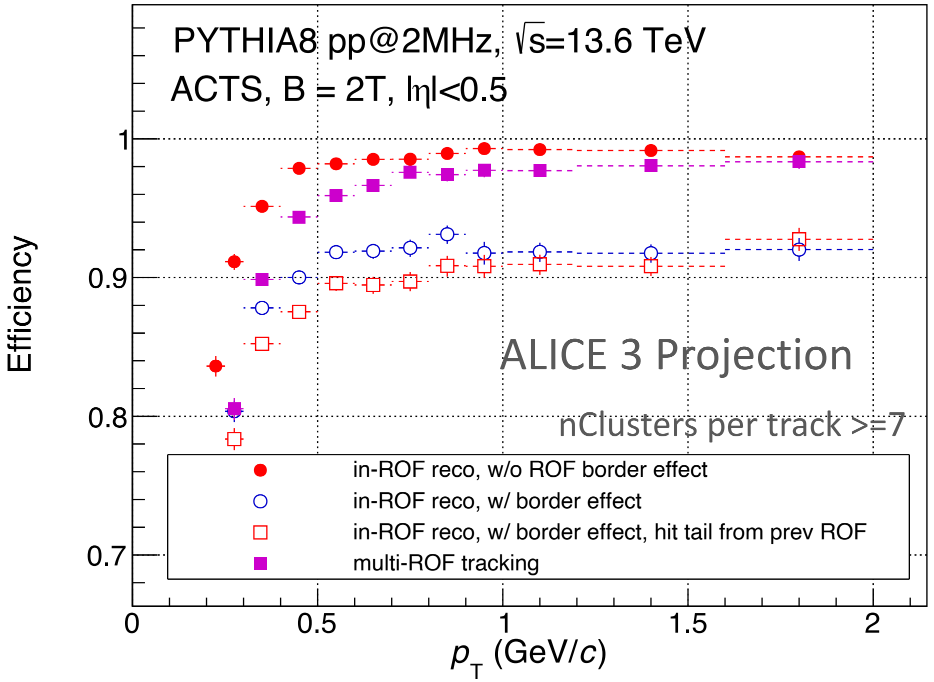
- Cluster loss on the ROF boundary due to the ALPIDE *time walk:*
  - "Time walk" determines the **variations of the time distance** between when a particle crosses the detector and the corresponding pulse in the front-end goes above the threshold

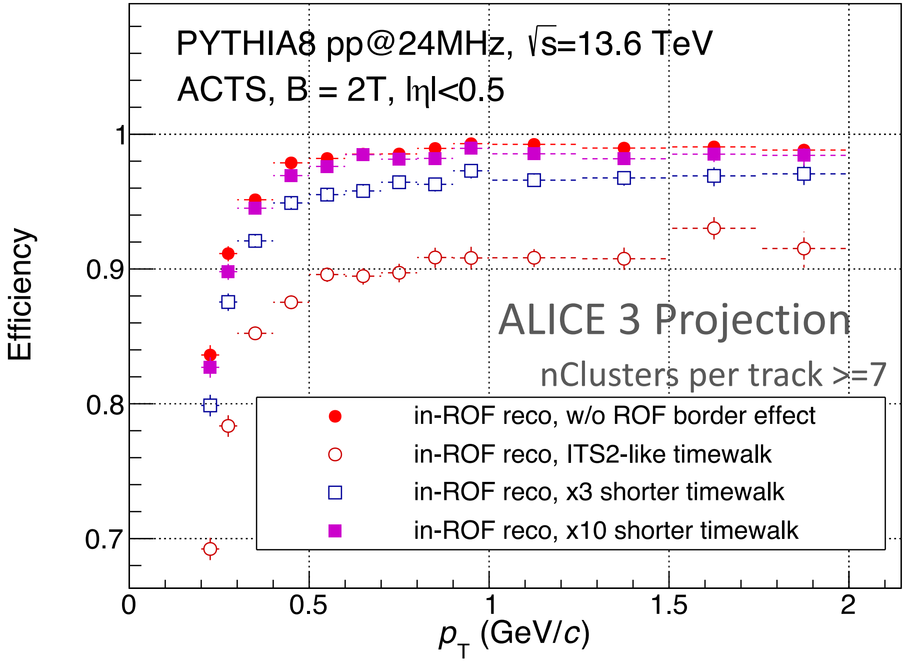
“Projection” of the border effect on ALICE 3 ROFs



- The *time walk* in ITS3 is expected to be similar to ITS2
- If **assume the same also for IT-OT of ALICE3,** + take narrower **ROFs of 500 ns** → the “gaps” in performance can even overlap!

- ROF border effect should be very significant for short ROF lengths (500 ns for IT/OT @ ALICE 3)
- Multi-ROF tracking allows one to recover tracking efficiency in ROFs of ALICE3
  - demonstrated with simulations (assuming the same time-walk as in ITS2)
- The reduction of the time walk allows to recover the in-ROF rec. efficiency, important aspect of R&D

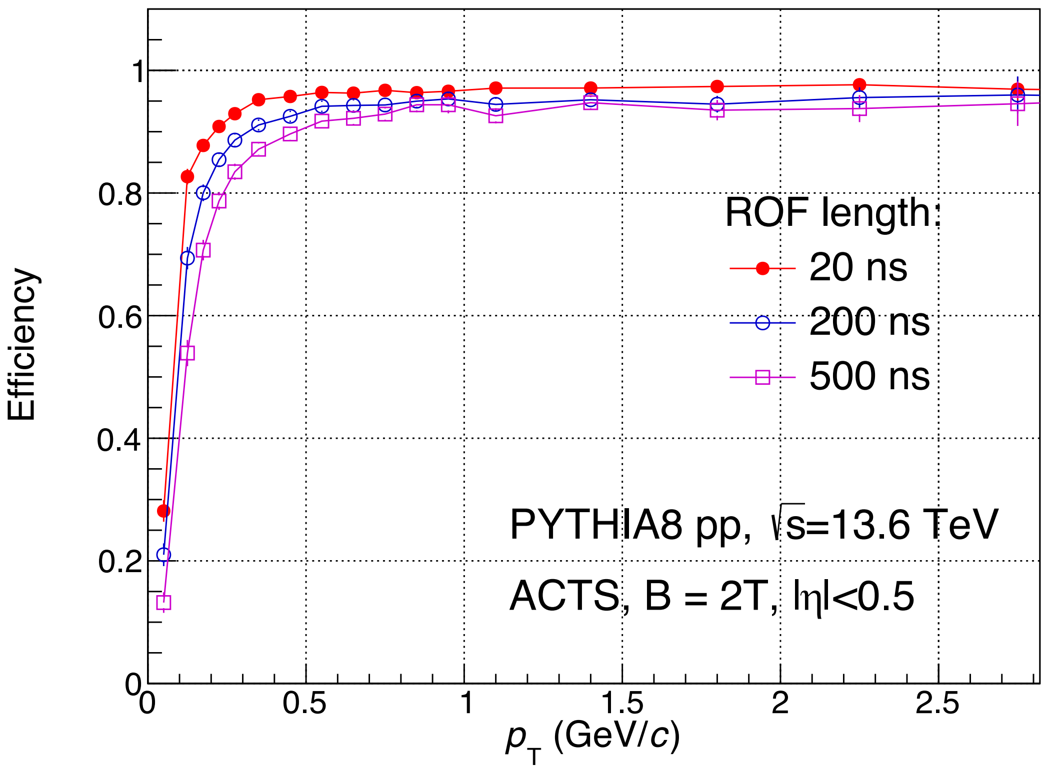




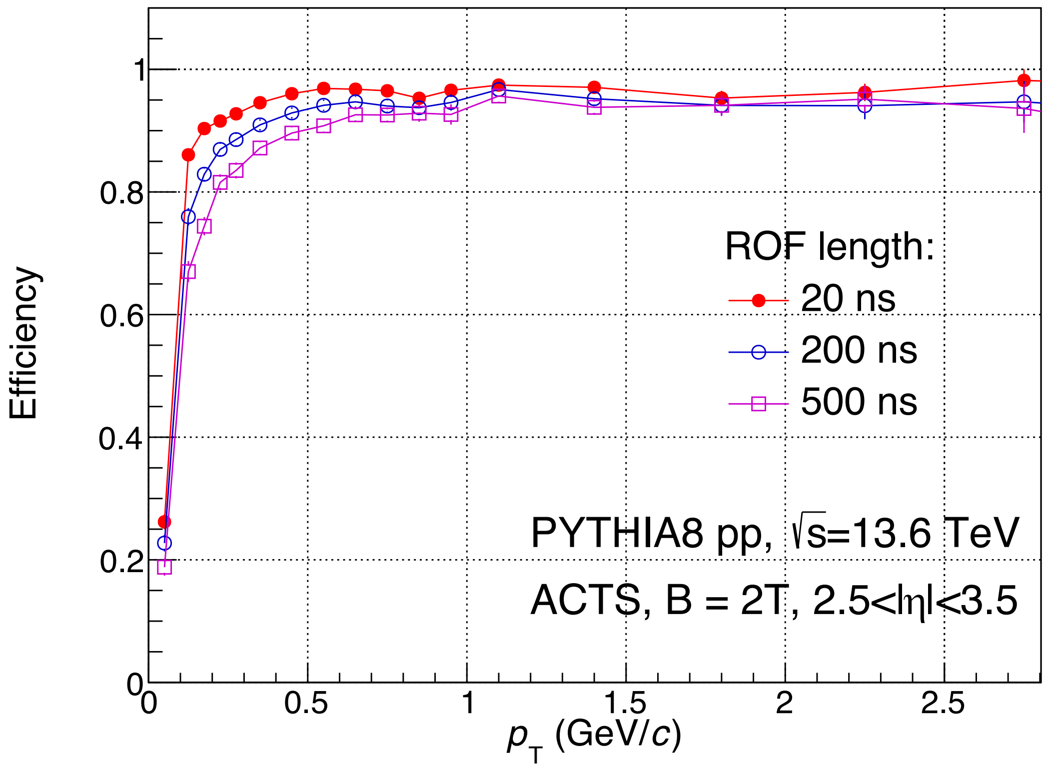
Efficiency vs p_T at different *pp* pileup

Updated March 25:
now with reconstructed seeds & in-ROF pileup

$|\eta| < 0.5$



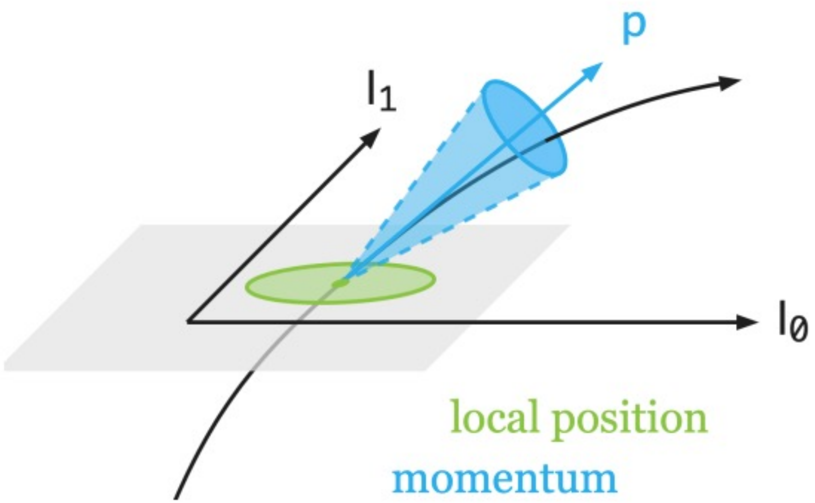
$2.5 < \eta < 3.5$

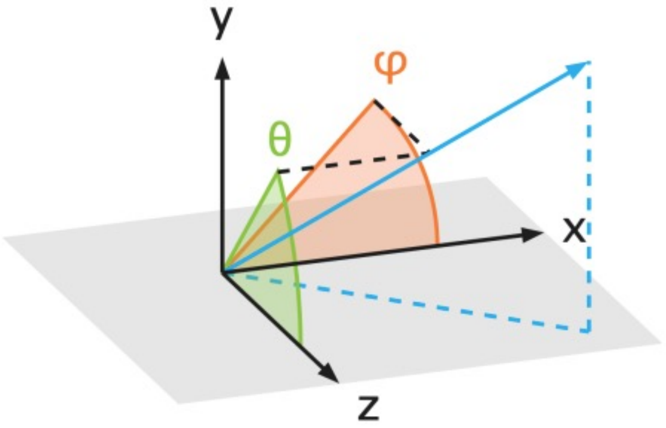


ACTS Event Data Model

Track parameterization:

local coordinates of the surface + global momentum **+ timing info**

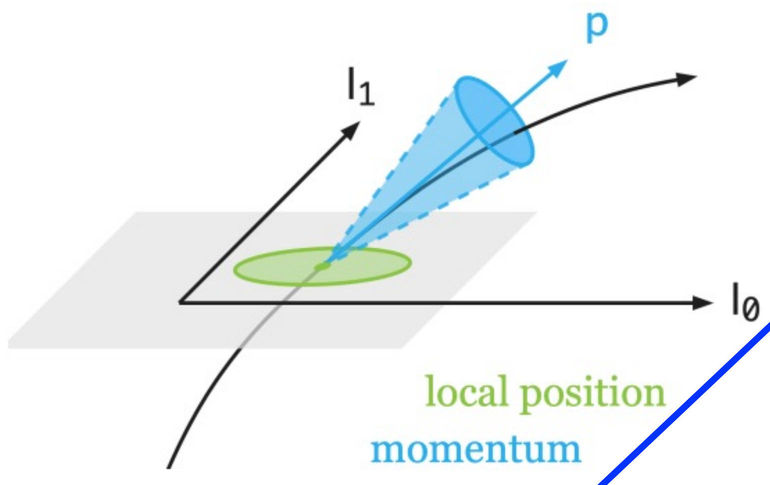




$$\vec{x} = (l_0, l_1, \phi, \theta, q/p, t)^T$$

$$C = \begin{bmatrix} \sigma^2(l_0) & \mathrm{cov}(l_0, l_1) & \mathrm{cov}(l_0, \phi) & \mathrm{cov}(l_0, \theta) & \mathrm{cov}(l_0, q/p) \\ . & \sigma^2(l_1) & \mathrm{cov}(l_1, \phi) & \mathrm{cov}(l_1, \theta) & \mathrm{cov}(l_1, q/p) \\ . & . & \sigma^2(\phi) & \mathrm{cov}(\phi, \theta) & \mathrm{cov}(\phi, q/p) \\ . & . & . & \sigma^2(\theta) & \mathrm{cov}(\theta, q/p) \\ . & . & . & . & \sigma^2(q/p) \end{bmatrix}$$

ACTS Event Data Model



$$\vec{x} = (l_0, l_1, \phi, \theta, q/p, t)^T$$

Parameter	l_0	l_1	phi	theta	q/p	t
Bound track parameters						
Pixel measurement						
Pixel measurement with time						
Strip measurement (along local x)						
Strip measurement (along local y)						
Drift time/circle measurement						
Track segment (straight line)						
...						

possible types of measurements

- Measurements can be represented as *subsets* of the full bound parameter space

A. Salzburger, ACTS workshop, Nov 2023