classification of the MCTrackCands

outline

- ★ Introduction
- ★ Classification Criteria
- ★ Results
- ★ ROI finding

Giulia Casarosa

INFN - Sezione di Pisa

INFN

Belle II Face2Face Tracking Meeting ~ January, 11st 2016

The Project

- Append a module following the TrackFinderMCTruth that classifies the MCTrackCands on the basis of their traceability and produces subsets of MCTrackCands:
  - **ideal**MCTrackCands
  - non-idealMCTrackCands
    - **fine**MCTrackCands
    - **nasty**MCTrackCands
- Estimate the pattern recognition (and fitting) efficiencies on the lists separately: expect ~1 on idealMCTrackCands, …
- The VXD and CDC pattern recognition modules will have separate lists
  - focused first on the classification of the MCTrackCands from the VXD

TrackFinderMCTruth TrackCands

current PR efficiency definition:

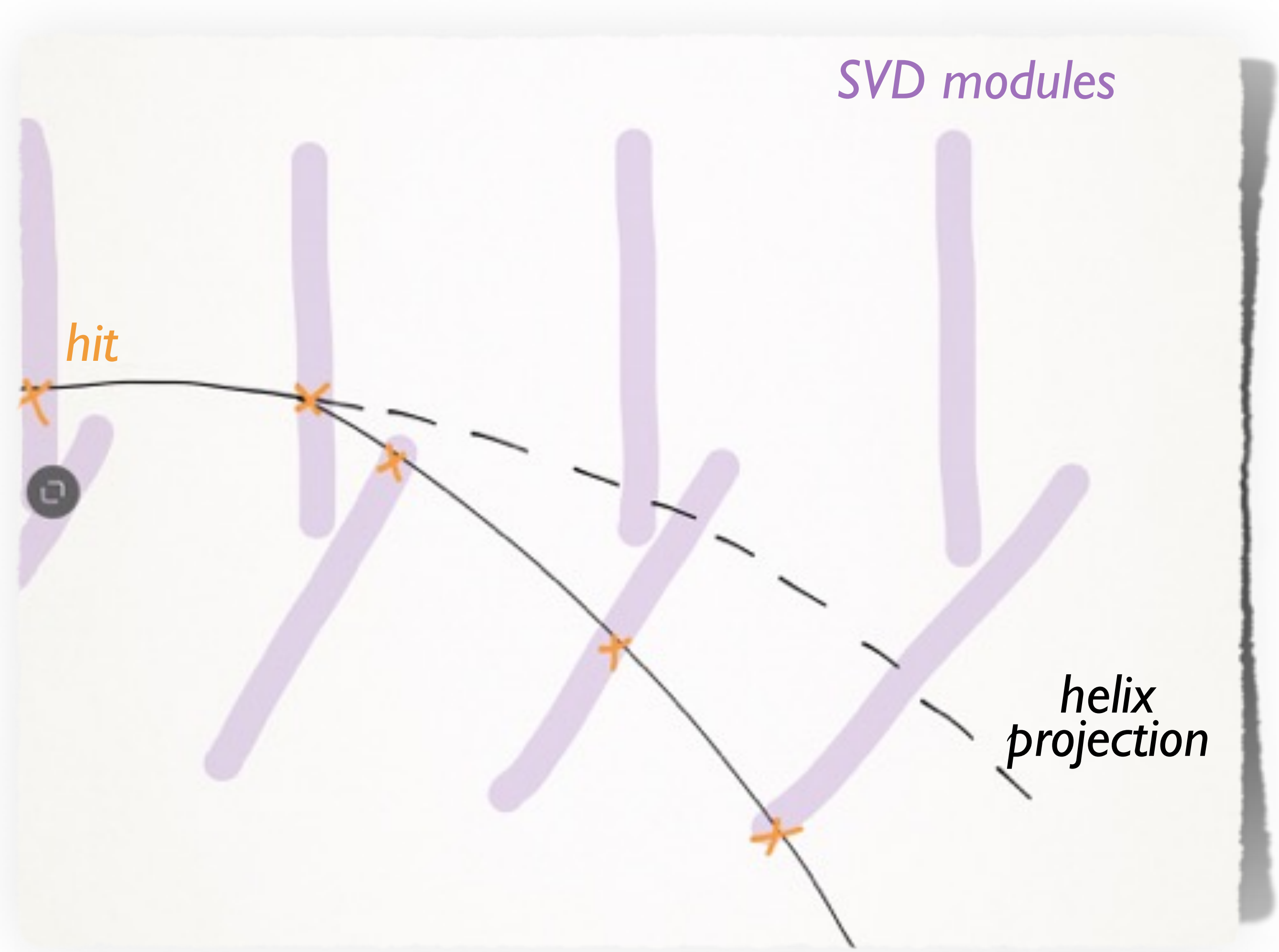
$$\varepsilon_{PR} = \frac{\#\ \text{MCTrackCand with at least one associated TrackCand}}{\#\ \text{MCTrackCand}}$$

TrackFinderMCTruth:

- ✓ factors out geometrical acceptance
- ✓ factors out detector inefficiencies
- ✓ requires a minimum number of hits

BUT it does not handle:

- kinks & large multiple scattering
- tracking volume covering only partially the helix
- hits in both outgoing and ingoing helix arms



TrackFinderMCTruth TrackCands

current PR efficiency definition:

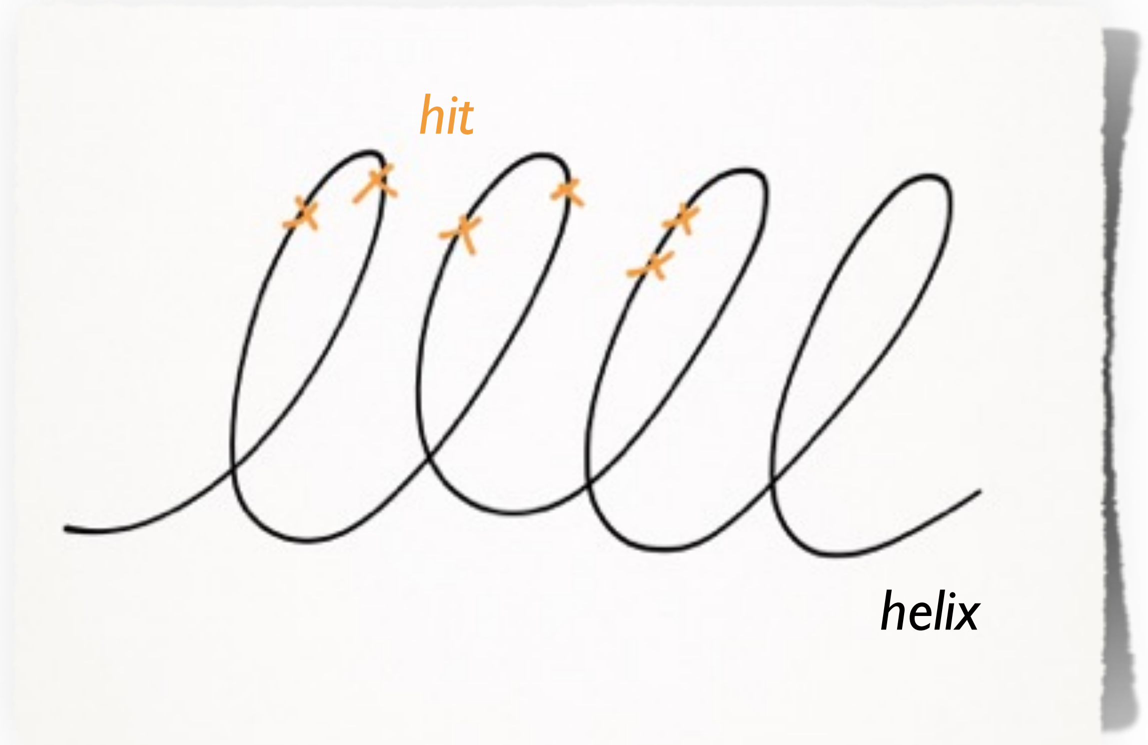
$$\varepsilon_{PR} = \frac{\#\ \text{MCTrackCand with at least one associated TrackCand}}{\#\ \text{MCTrackCand}}$$

TrackFinderMCTruth:

- ✓ factors out geometrical acceptance
- ✓ factors out detector inefficiencies
- ✓ requires a minimum number of hits

BUT it does not handle:

- ◉ kinks & large multiple scattering
- ◉ tracking volume covering only partially the helix
- ◉ hits in both outgoing and ingoing helix arms



TrackFinderMCTruth TrackCands

current PR efficiency definition:

$$\varepsilon_{PR} = \frac{\#\ \text{MCTrackCand with at least one associated TrackCand}}{\#\ \text{MCTrackCand}}$$

TrackFinderMCTruth:

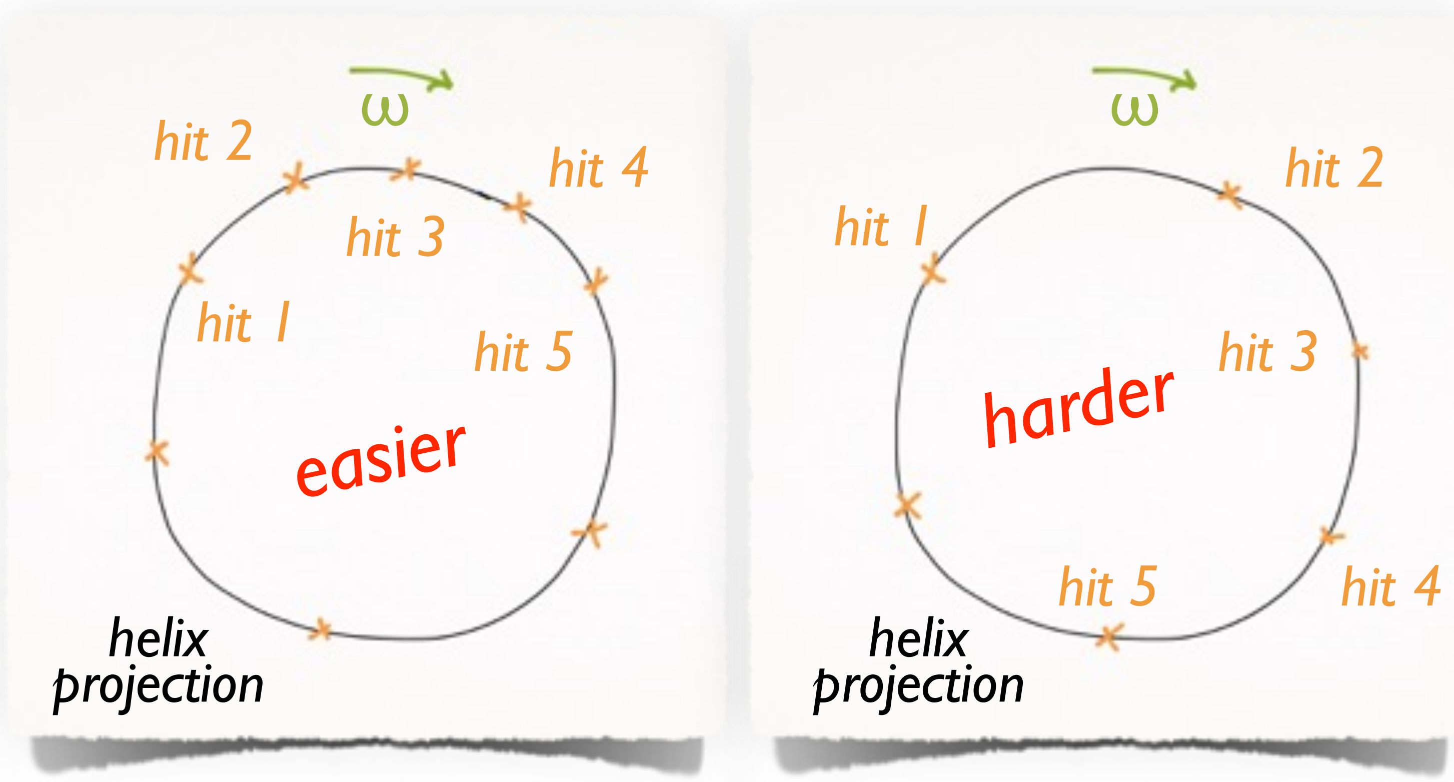
✓ factors out geometrical acceptance

✓ factors out detector inefficiencies

✓ requires a minimum number of hits

BUT it does not handle:

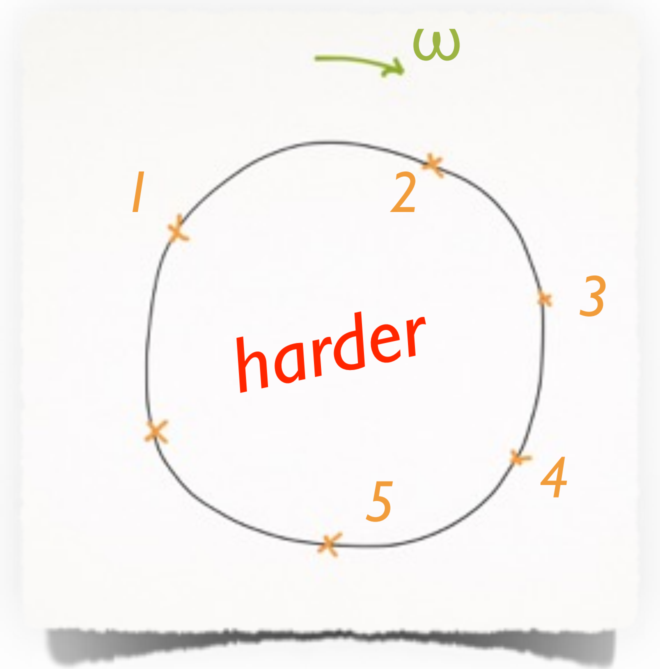
- kinks & large multiple scattering
- tracking volume covering only partially the helix
- hits in both outgoing and ingoing helix arms

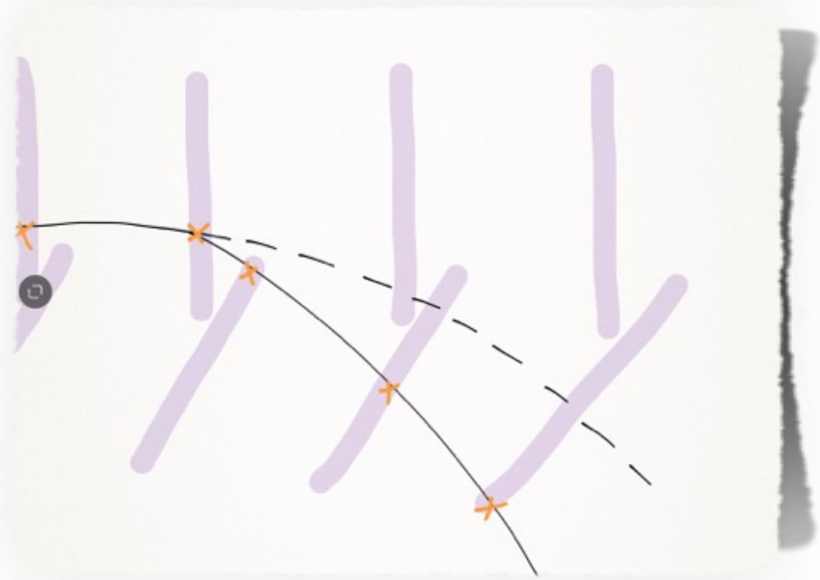


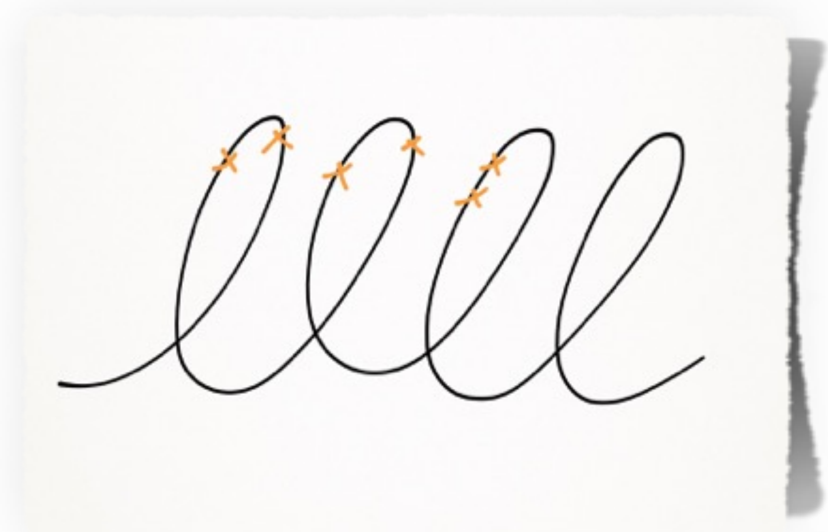
➡ There are other classes of non-**ideal**MCTrackCands

The MCTrackCandClassifier Module

➡ I have implemented a set of criteria to reject the MCTrackCands belonging to the categories shown in the previous slides:





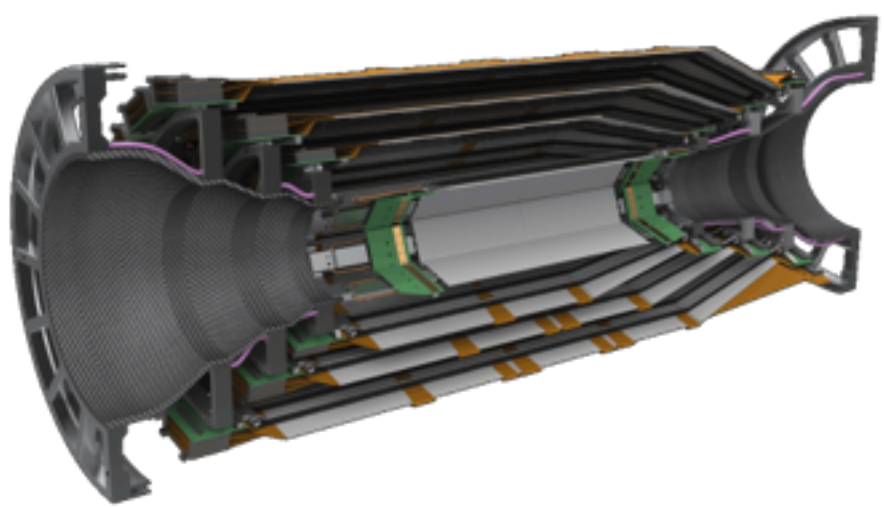


➡ The accepted MCTrackCands are *copied* into the **ideal**MCTrackCand StoreArray.

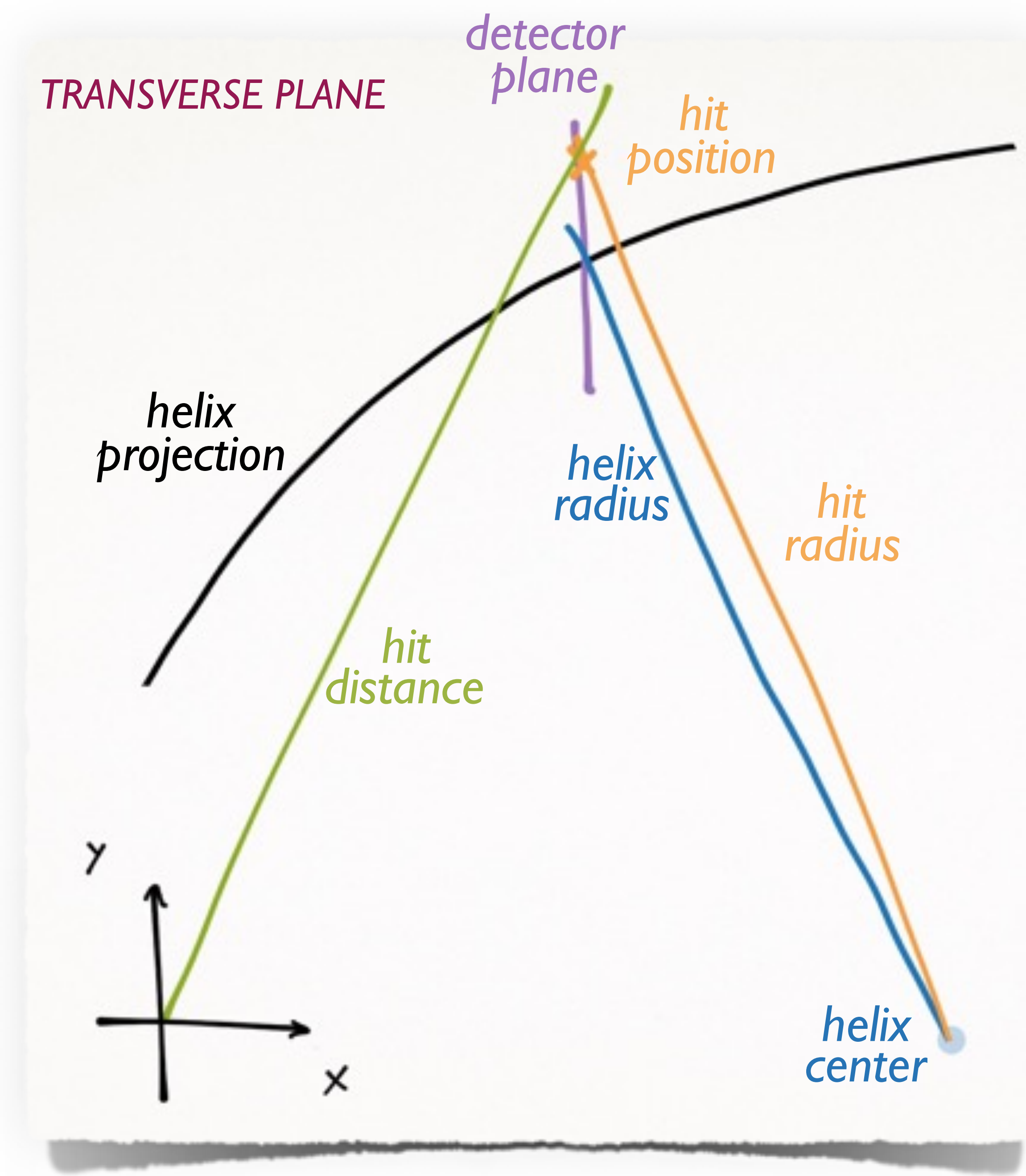
➡ This is a work-in-progress module, further improvements needed on the criteria side and on the module design (wait for slide 22)

The Model

- The geometry of the problem is quite complicated



- limit the numbers of degrees of freedom:
  - work on the transverse plane
  - hit position (distance from helix center and from 0,0)
  - helix radius
- Use as much as useful MCTruth informations as possible
  - MCParticle
  - TrueHit



Classification Sequence

- ➡ TrackFinderMCTruth produces the list of MCTrackCands
  - no cut on track energy (the default EnergyCut = 0.1)
  - use PXDHits and SVDHits, or SVDHits only
  - use clusters
  - minimum number of one-dimensional hits = 5
- ➡ retrieve the MCParticle related to the MCTrackCand
- ➡ Loop on the Clusters related to the MCTrackCand
  - for each Cluster retrieve the corresponding TrueHit:
    - if the TrueHit satisfies the classification criteria → the cluster is accepted
    - otherwise → the cluster is rejected & move to the next MCTrackCand
- ➡ Check if the Cluster is 1D (only *u* or only *v* SVD cluster) or 2D (PXD cluster or both *u* and *v* SVD clusters)
- ➡ If at least 5 1D information are accepted → the MCTrackCand is classified as **ideal**MCTrackCand (with *all* the hits belonging to the original MCTrackCand)

Criteria #1

(isInAnnulus)

- predict the position of the next hit on the expected detector plane, taking into account multiple scattering

$$\theta_{MS} = \frac{13.6\ \mathrm{MeV}}{\beta c p} Z \sqrt{\frac{X}{X_0}}$$

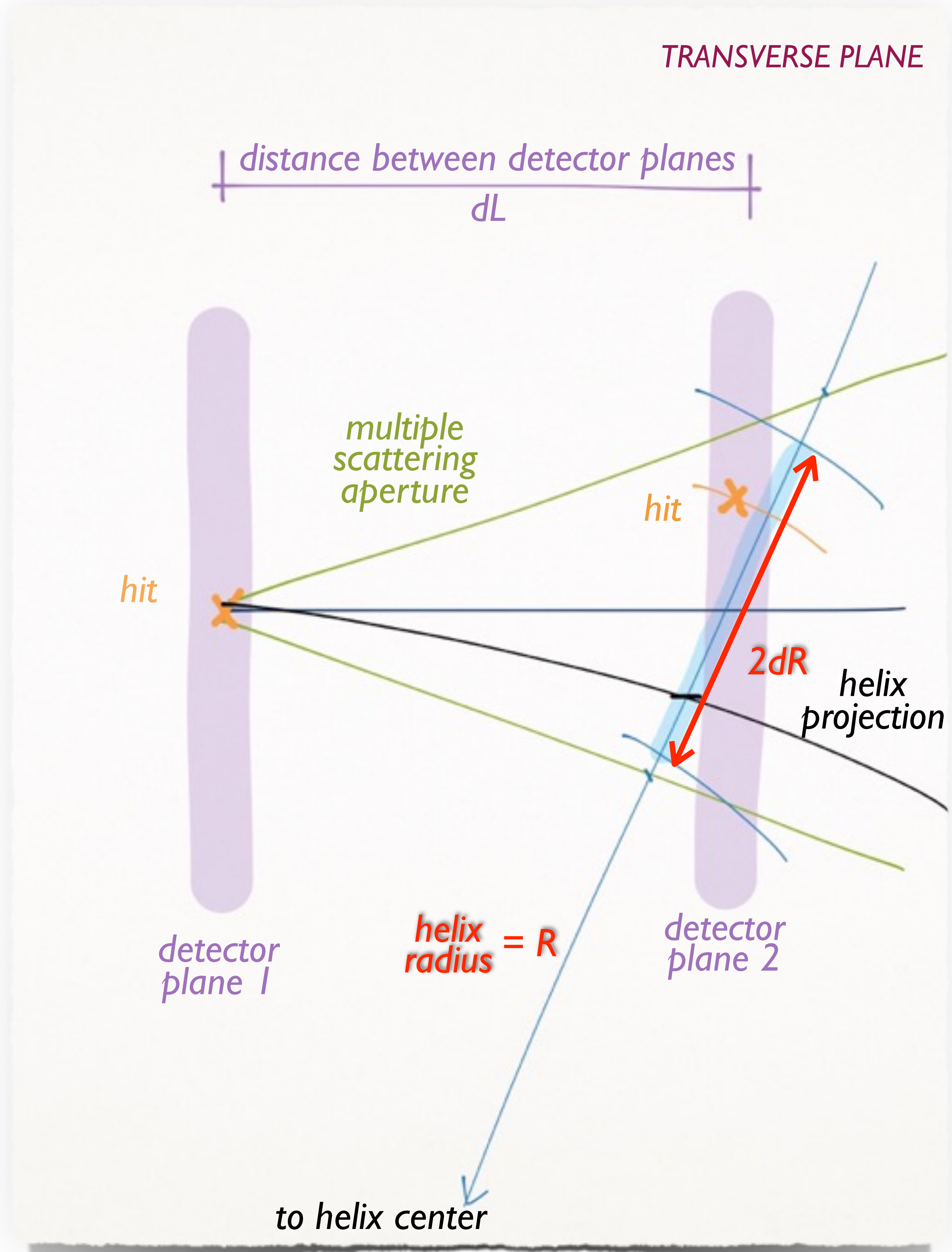
- project the region where the next hit is expected along the helix radius

$$dR = n\, dL\, \theta_{MS}$$

- check if the next-hit distance from the helix center (d) lies in the expected region

$$|d - R| < dR$$

- If the hit satisfies the criteria, check the next criteria (next slide), otherwise move to the next MCTrackCand



Multiple Scattering Effect

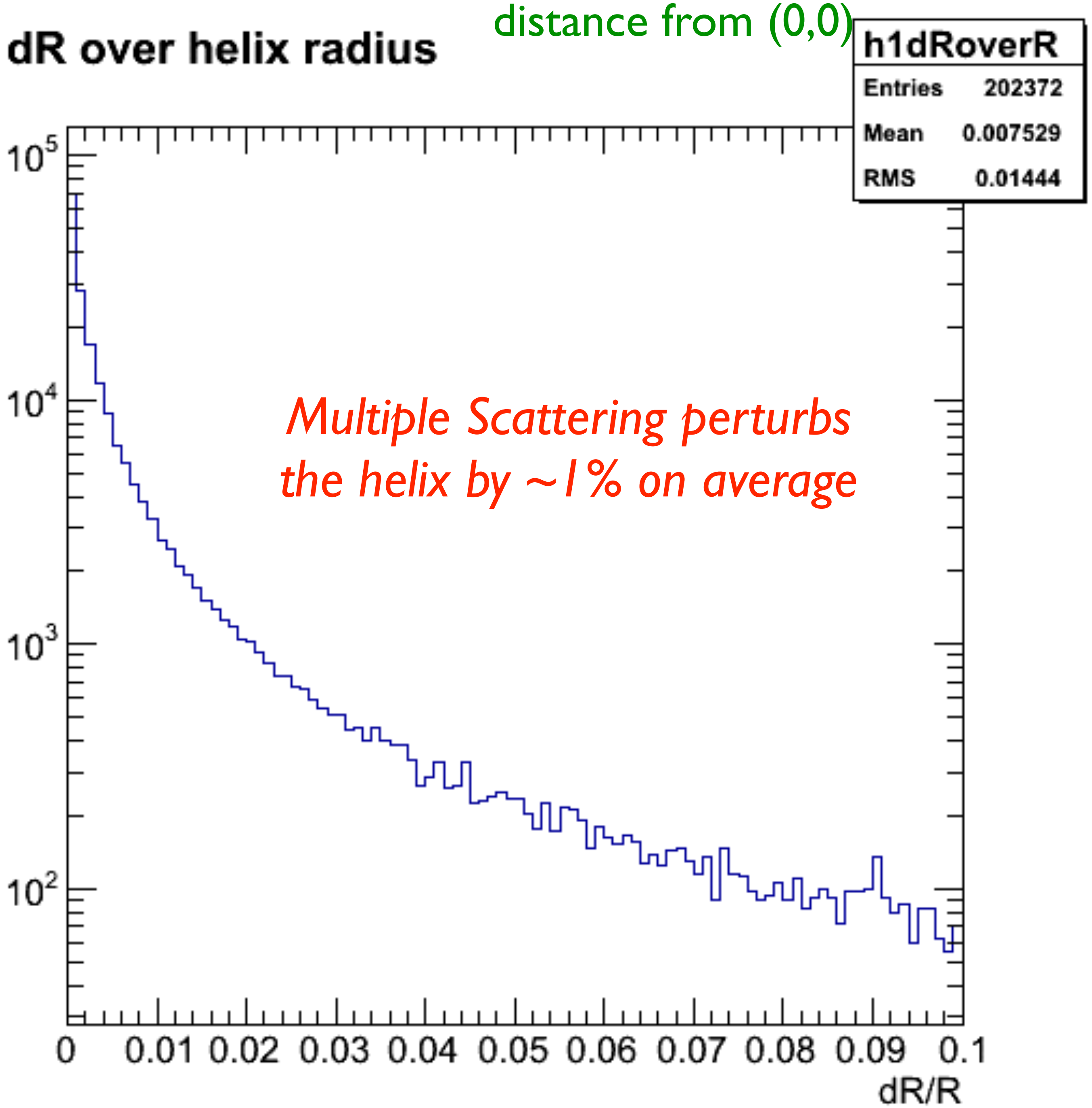
$$dR = n\, dL\, \theta_{MS}$$

- n: 3
- dL: chosen based on hit distance from (0,0)

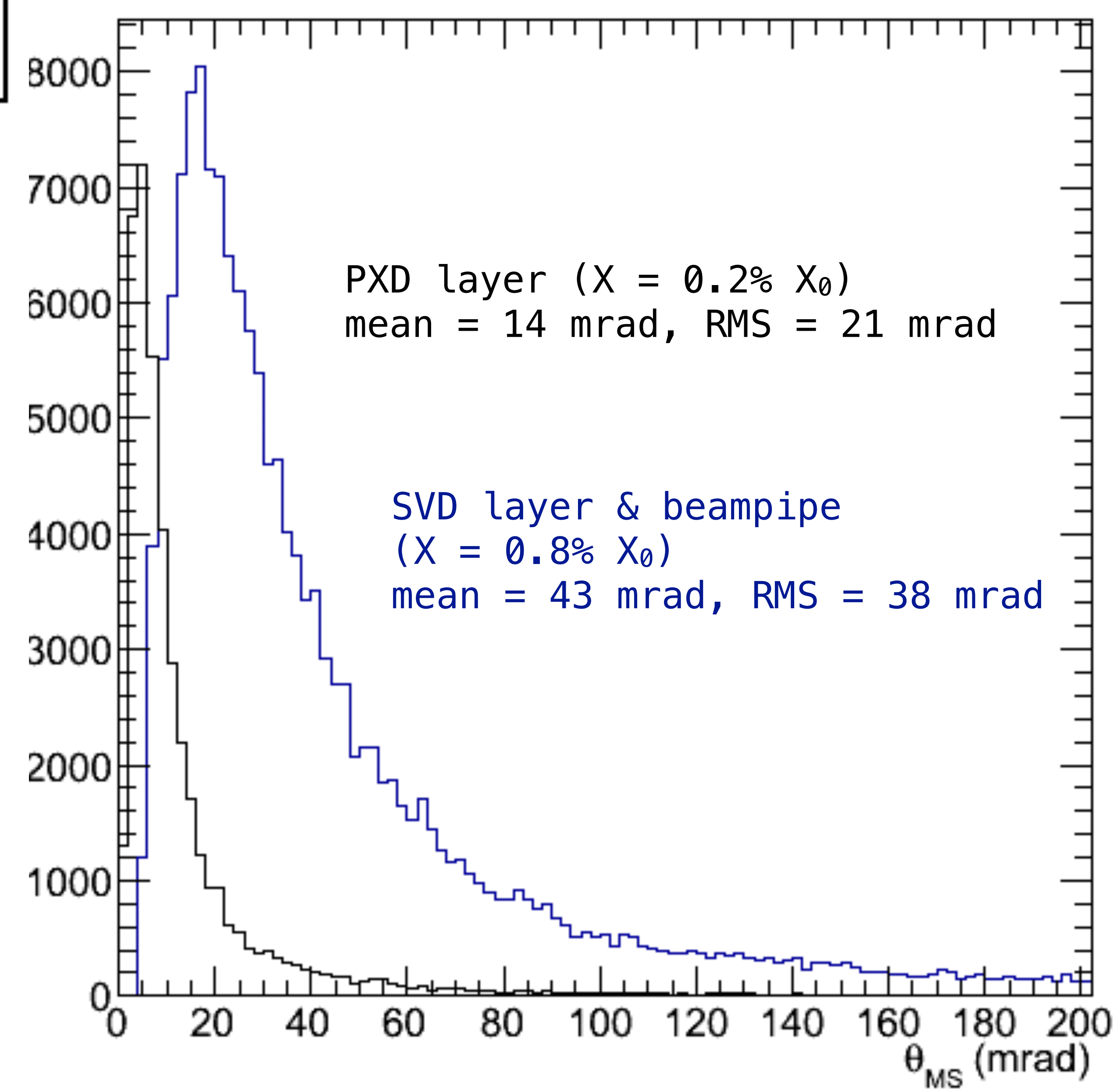
$$\theta_{MS} = \frac{13.6\ \mathrm{MeV}}{\beta c p} Z \sqrt{\frac{X}{X_0}}$$

- $\frac{X}{X_0}$: evaluated using the Enter and Exit Particle positions in the sensor volume (from TrueHit)
- $\beta c p$: p^2/E from the related MCParticle

dR over helix radius

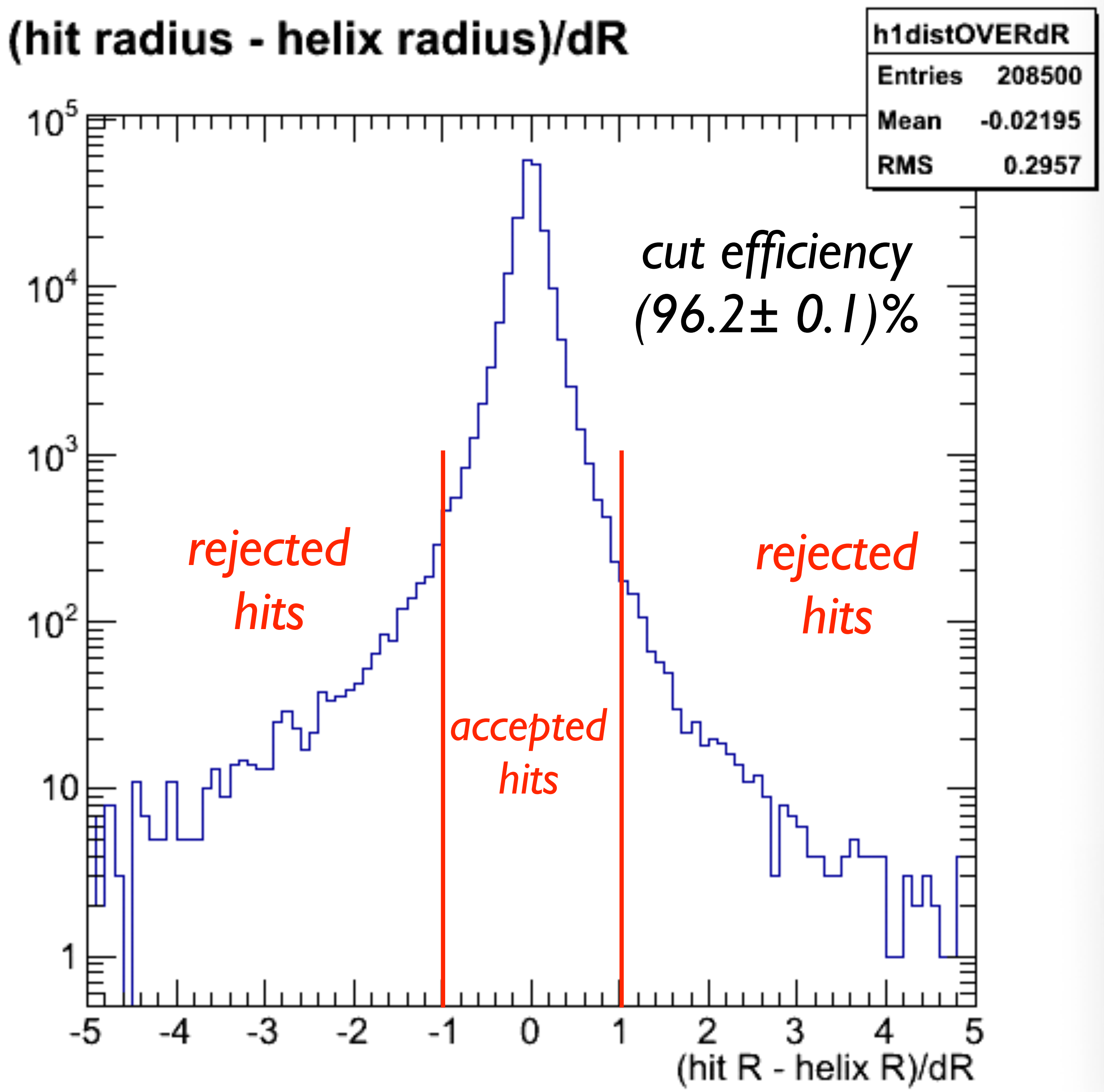


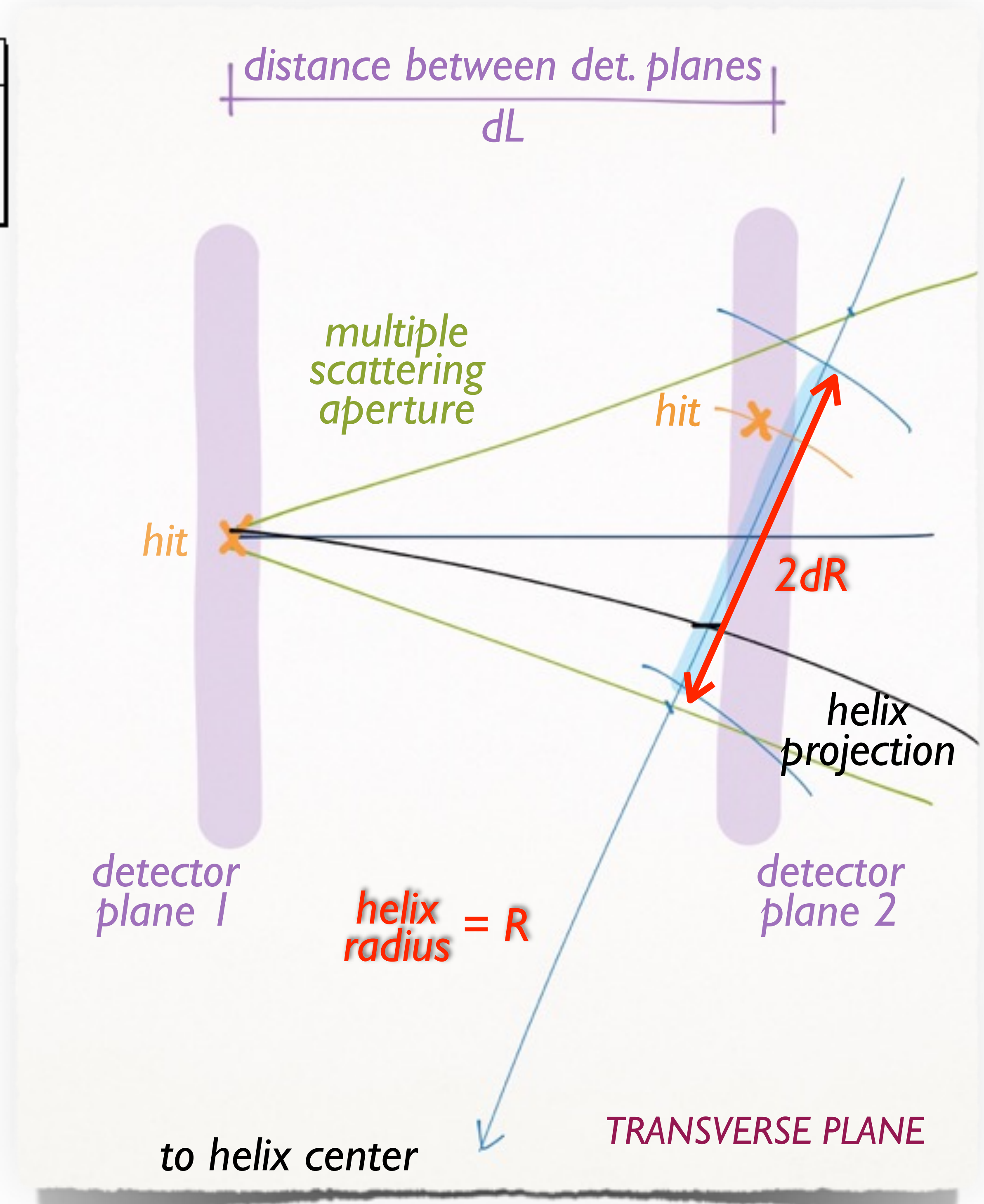
distribution of θ_{MS} for generic Y(4S) events



Criteria #1 at work

$$|d - R| < dR$$





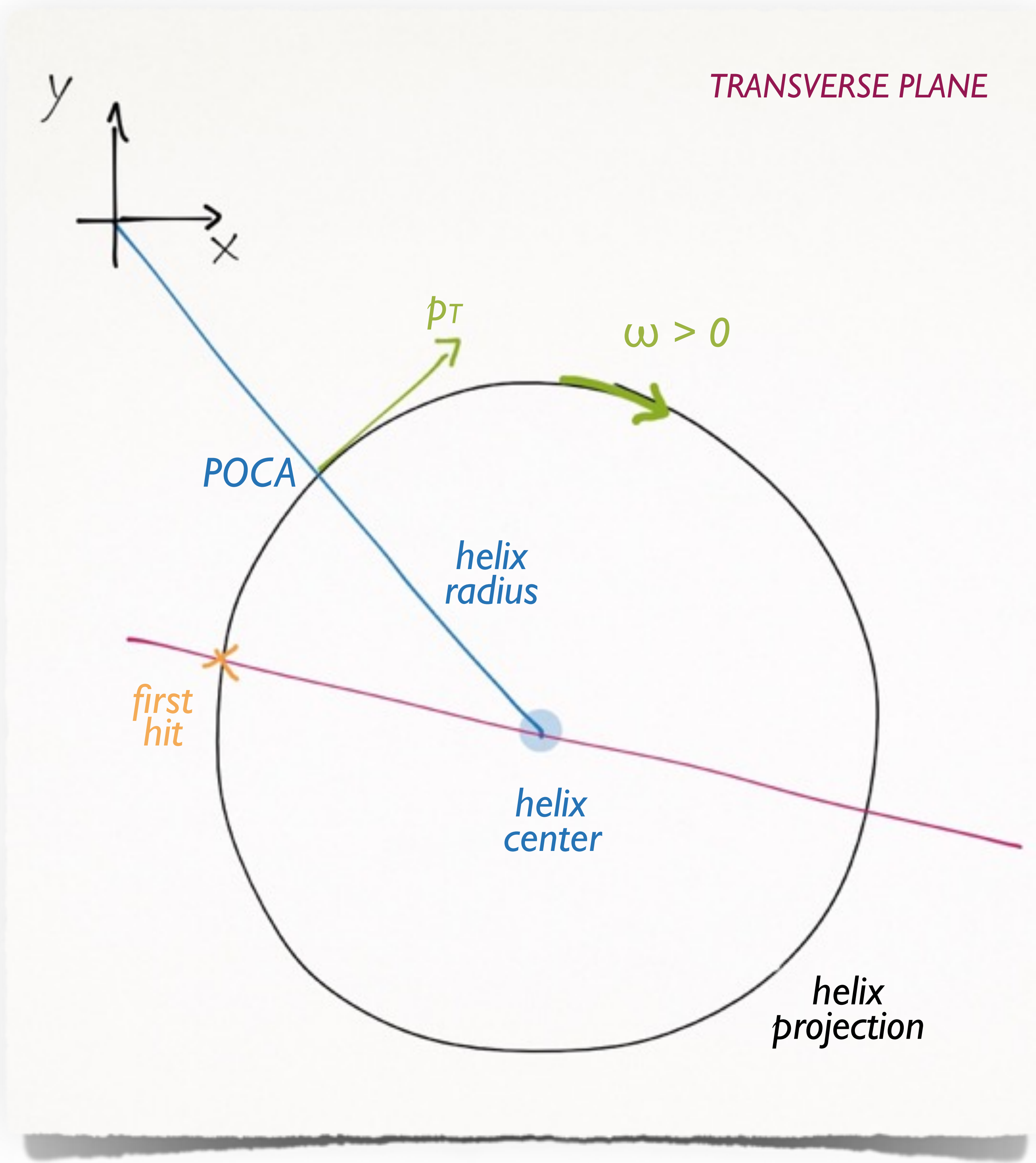
➡ asymmetric distribution, higher left tail since no energy loss taken into account: the correction is symmetric while the physical effect (energy loss) is not.

can be taken into account in the future

Criteria #2

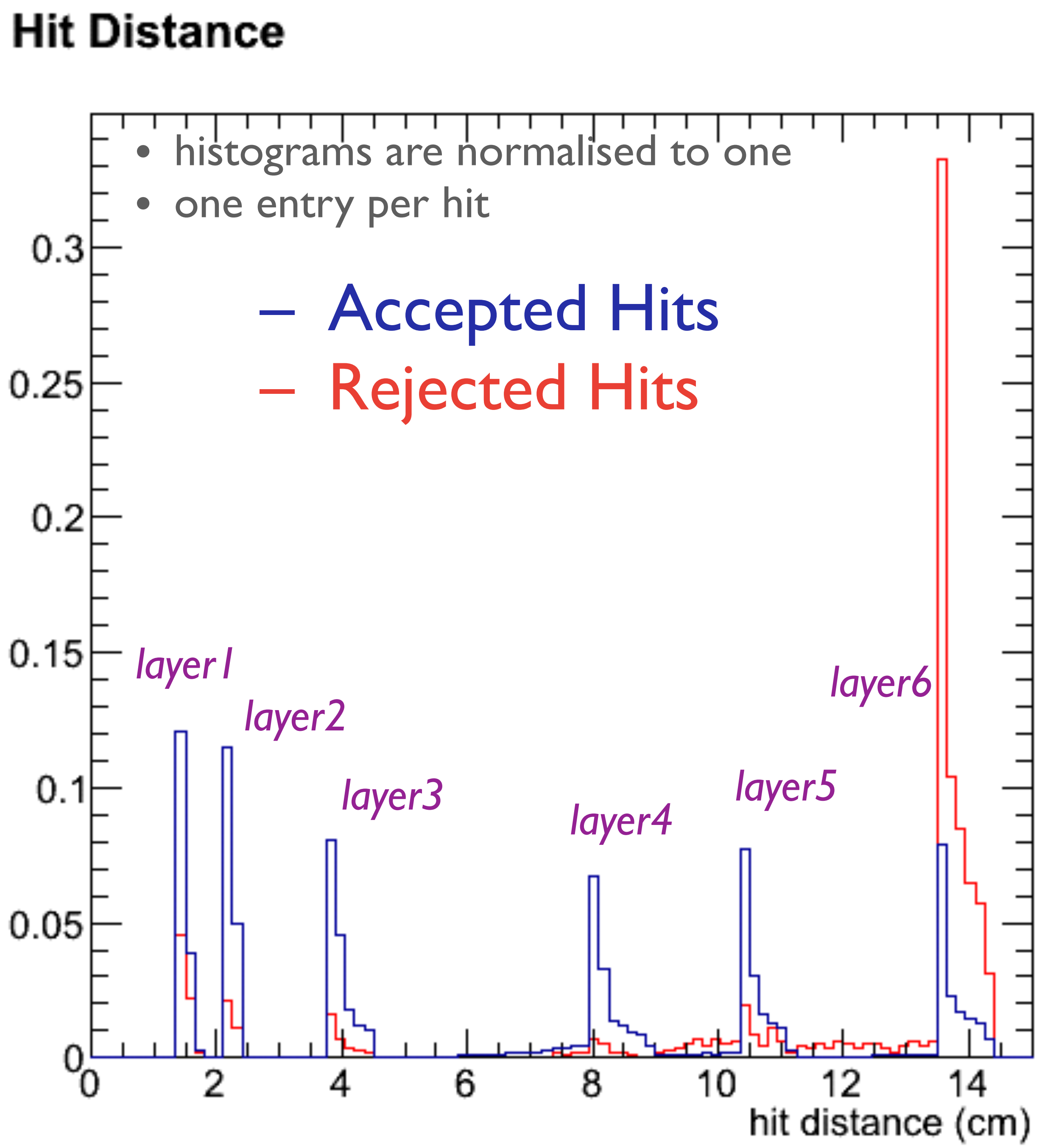
(isInSemiplane)

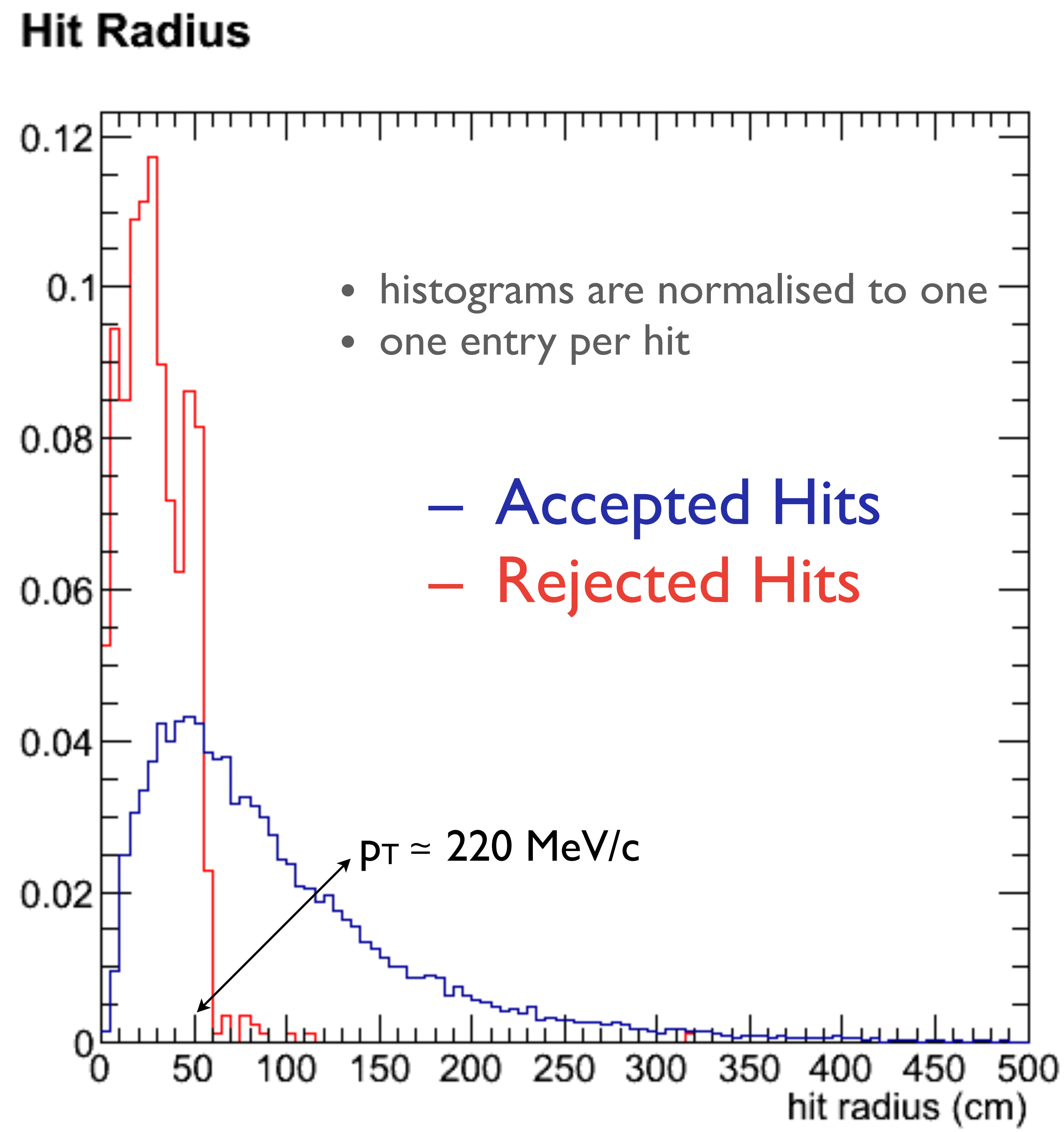
- ➡ divide the transverse plane into 2 regions given the first hit and the p_T at POCA
- ➡ given the curvature ω and the relative position between the first hit and the helix center, predict the semi-plane where the next hit is expected to be
- ➡ if the next hit belongs to the expected semi-plane, accept the hit and move to the next one (checking first criteria #1), otherwise move to the next MCTrackCand



Criteria #2 at work

cut efficiency
(98.8± 0.1)%





➡ asymmetric distribution, higher left tail since no energy loss taken into account: the correction is symmetric while the physical effect (energy loss) is not.

Criteria #3

(isInFirstLap)

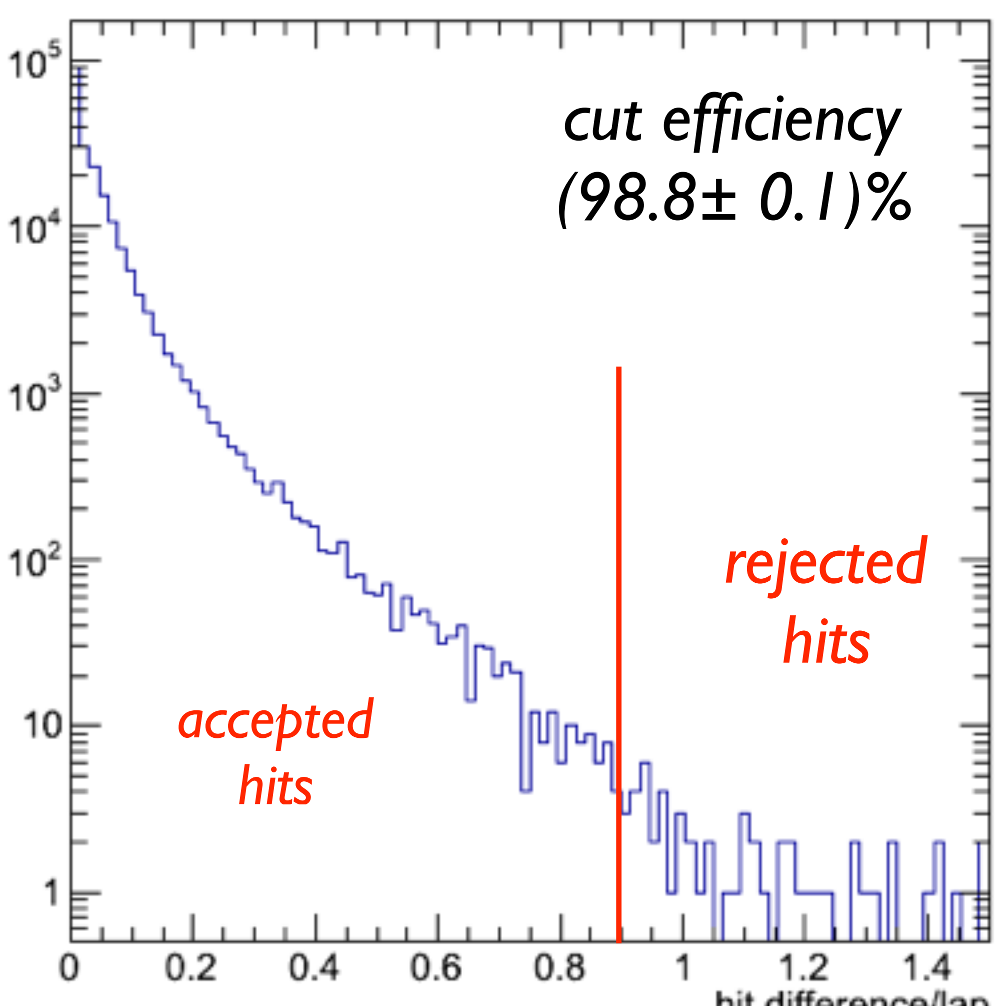
➡ evaluate the time need to complete a lap in the transverse plane:

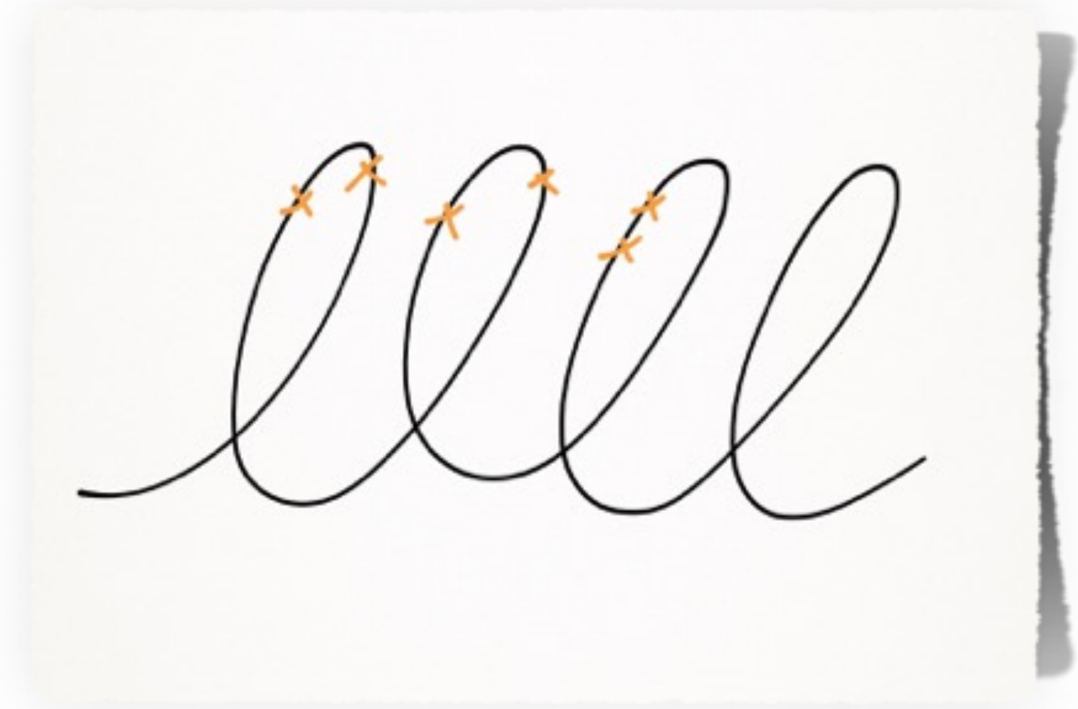
$$t_{hit} - t_{firstHit} < f\, t_{lap}$$

f → 0.9

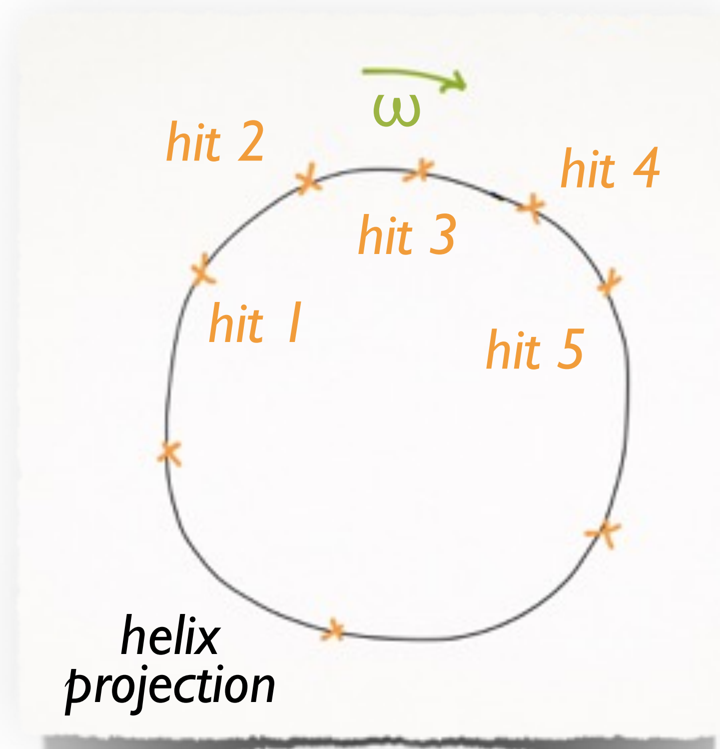
$$t_{lap} = \frac{2\pi E}{B_z c}$$

Hit Time Difference over Lap Time



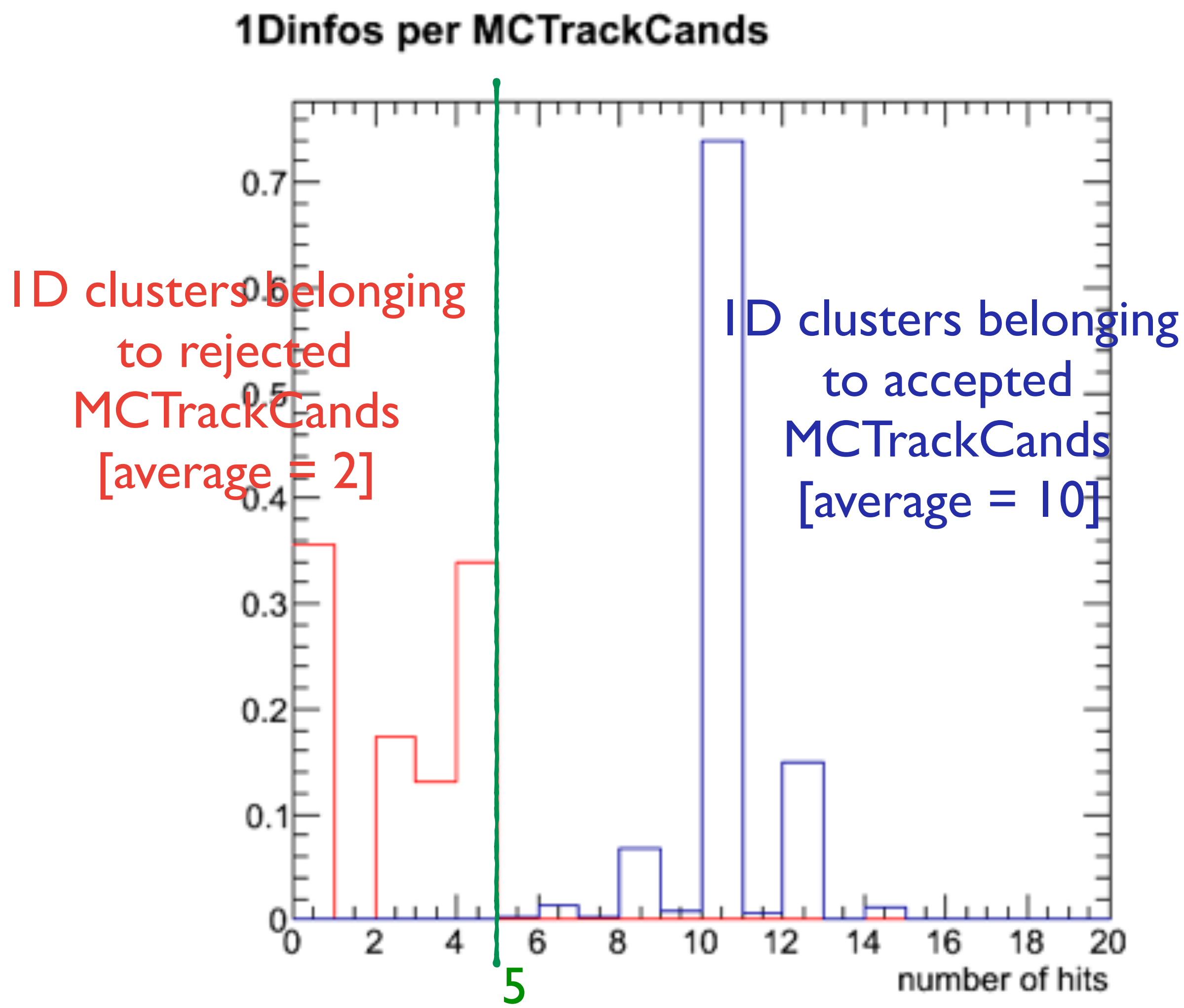


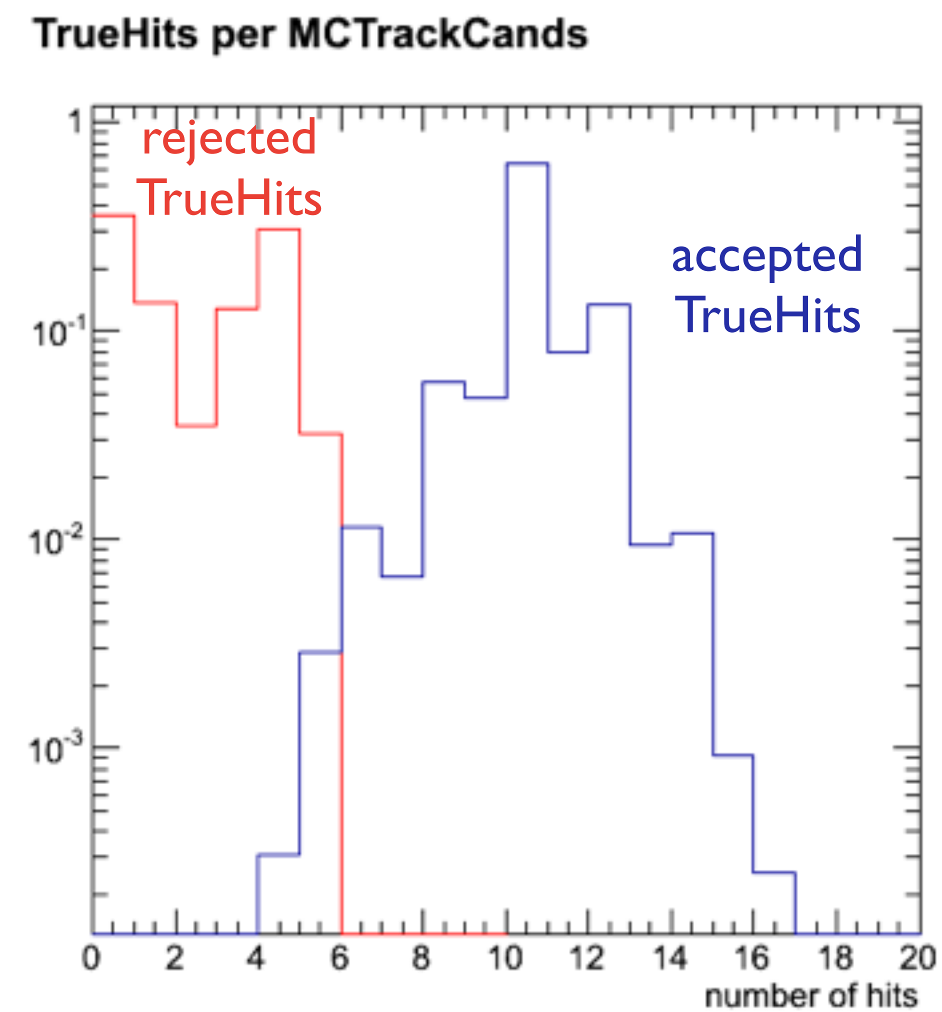
can look like this in transverse plane



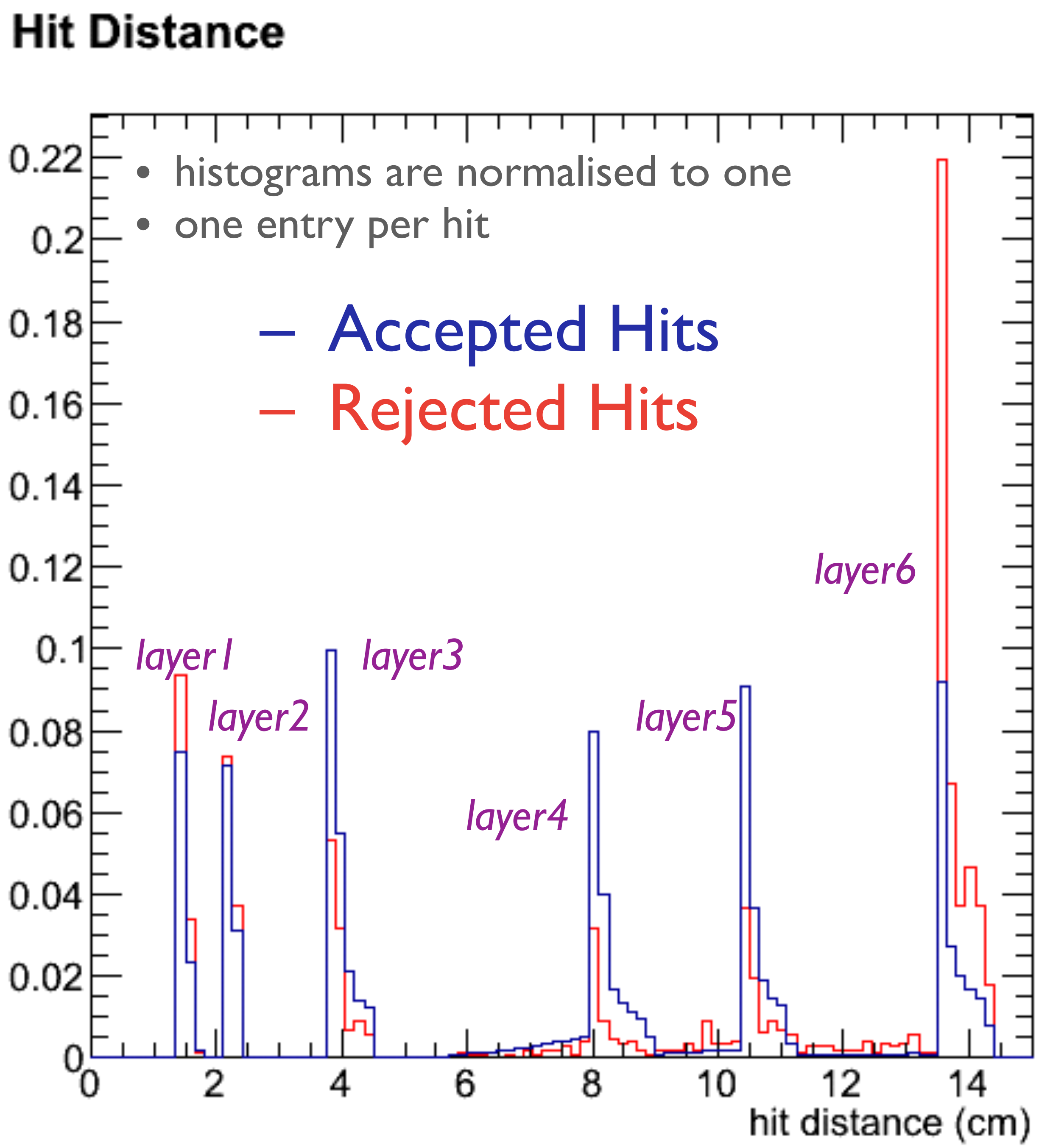
MCTrackCandClassifier at Work

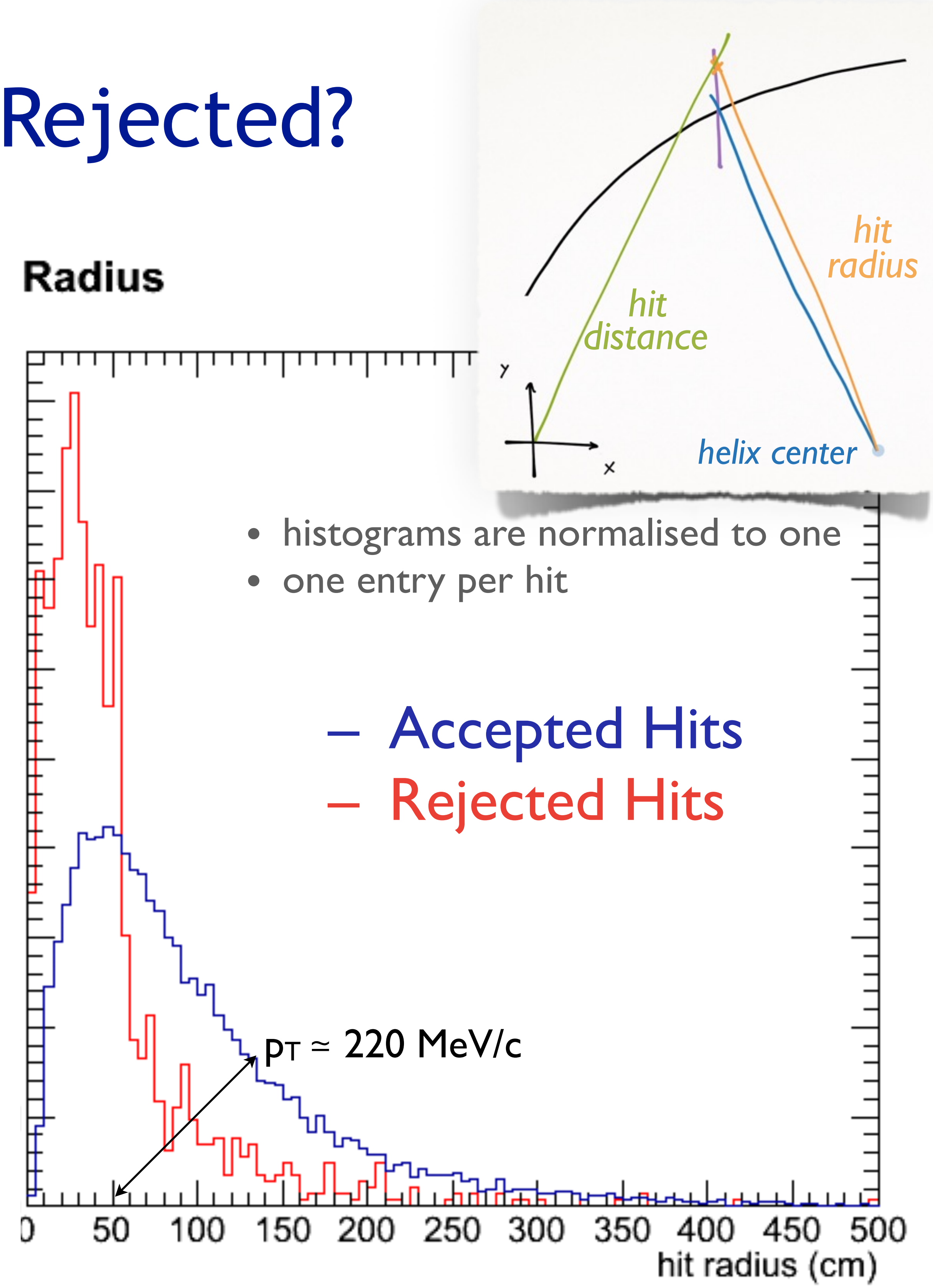
- 2k Υ(4S) generic decays, *Belle II* geometry
- 20k MCTrackCands (PXD&SVD TrueHits, use of clusters, # of 1-D hits > 5, no energy cut)
- **fraction of MCTrackCands classified as ideal = (96.2±0.1)%**





What Hits are Rejected?

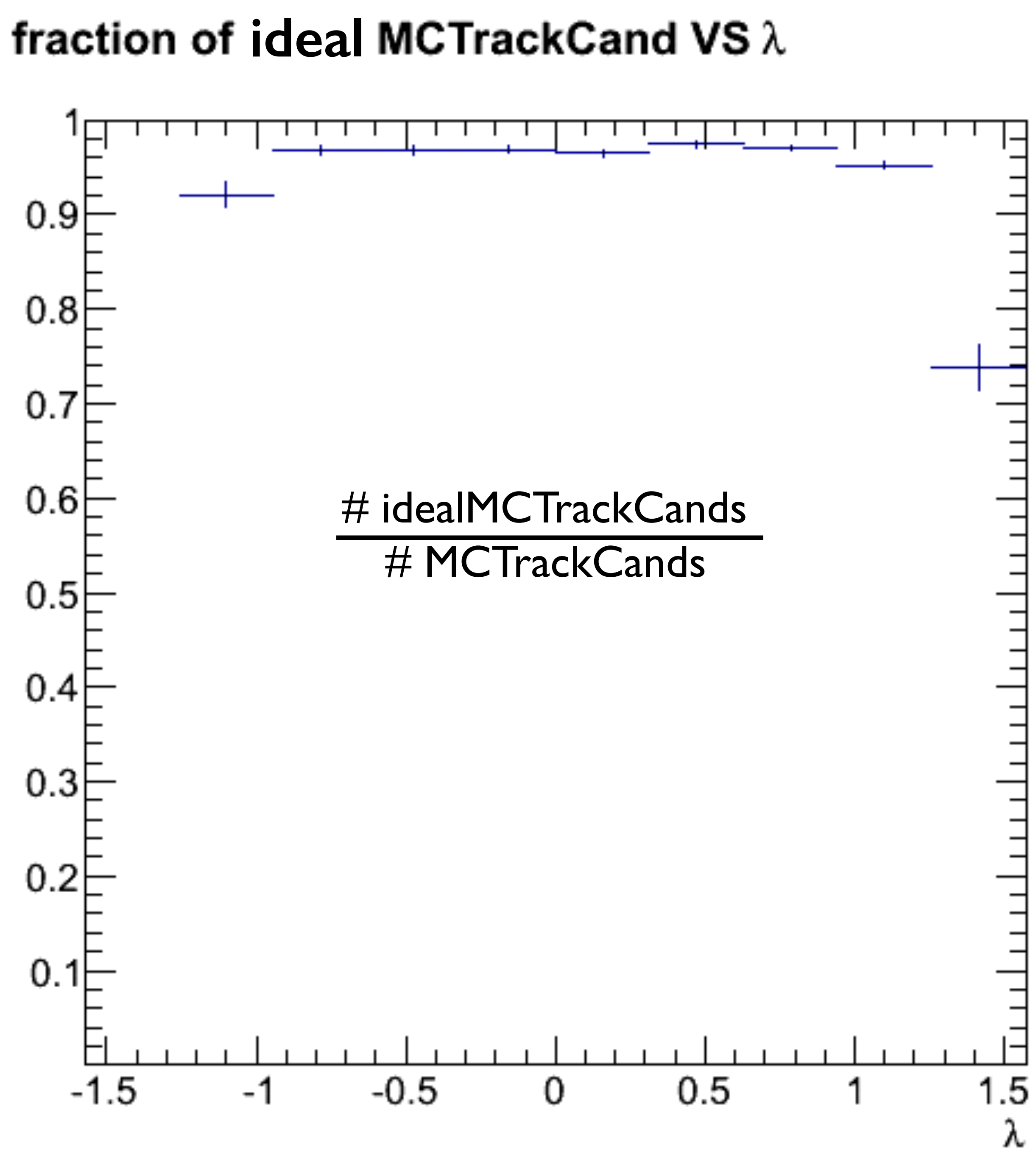




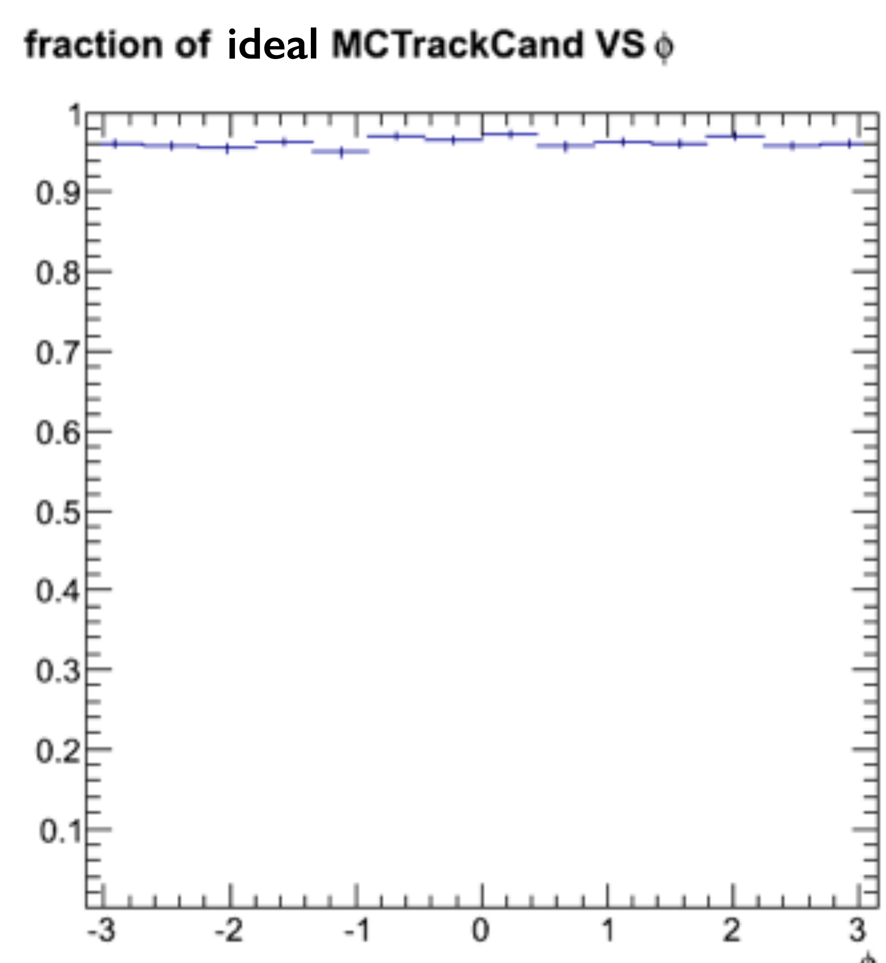
➡ Most of the reject hits are in Layer6, and a smaller fraction in the wedge part of the SVD

➡ Most of the rejected hits belong to low transverse momentum tracks (R < 70 cm ↔ p_T < 300 MeV/c)

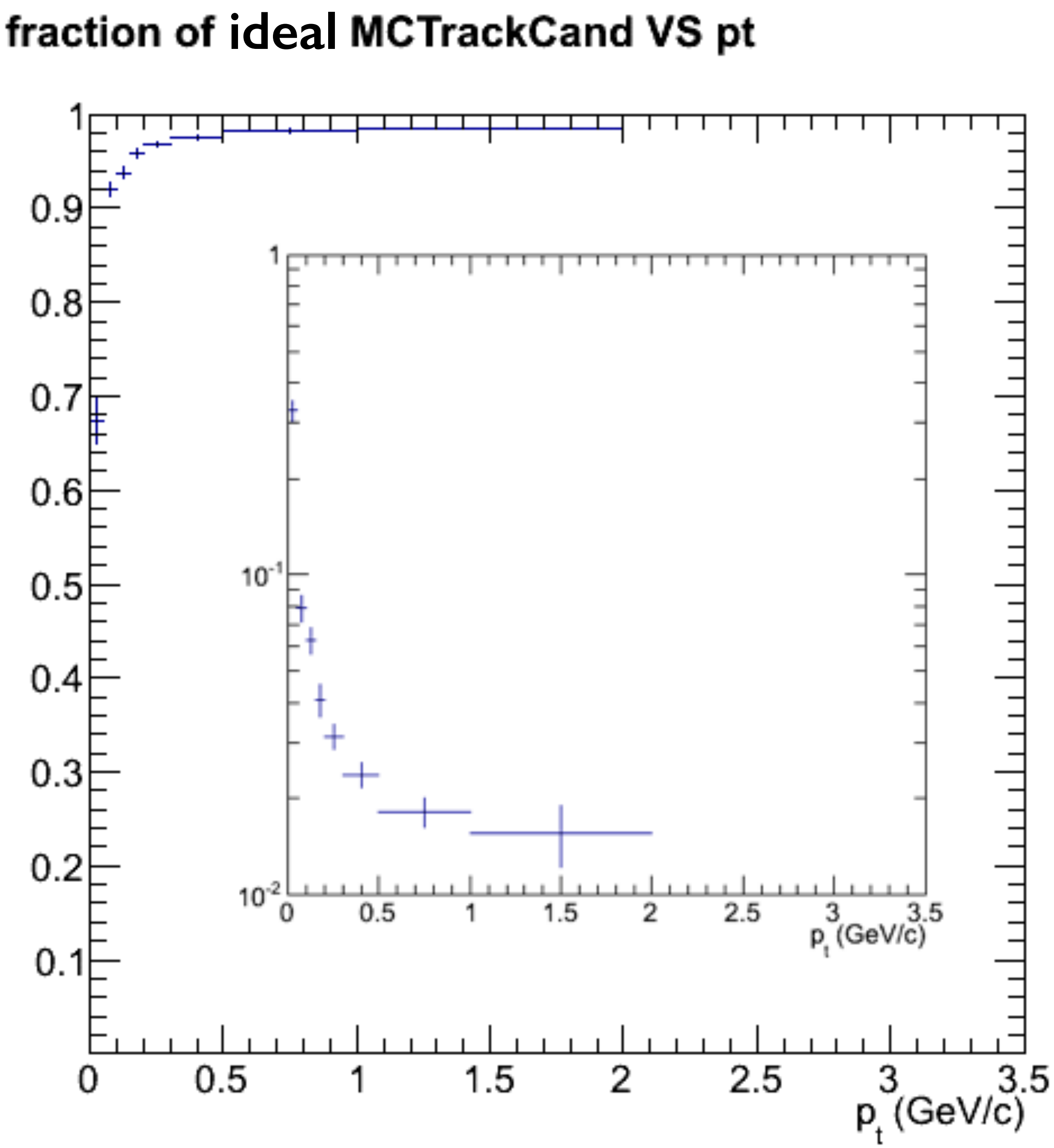
idealMCTrackCand Acceptance



- 75% of MCTrackCands in the forward direction are classified as ideal
    - selection criteria are not designed to be applied to hits in the *wedge* part of the SVD
- 98% of the MCTrackCands in the central region are classified as ideal
- the fraction of ideal MCTrackCands is independent of ϕ, as expected



idealMCTrackCands VS transverse momentum

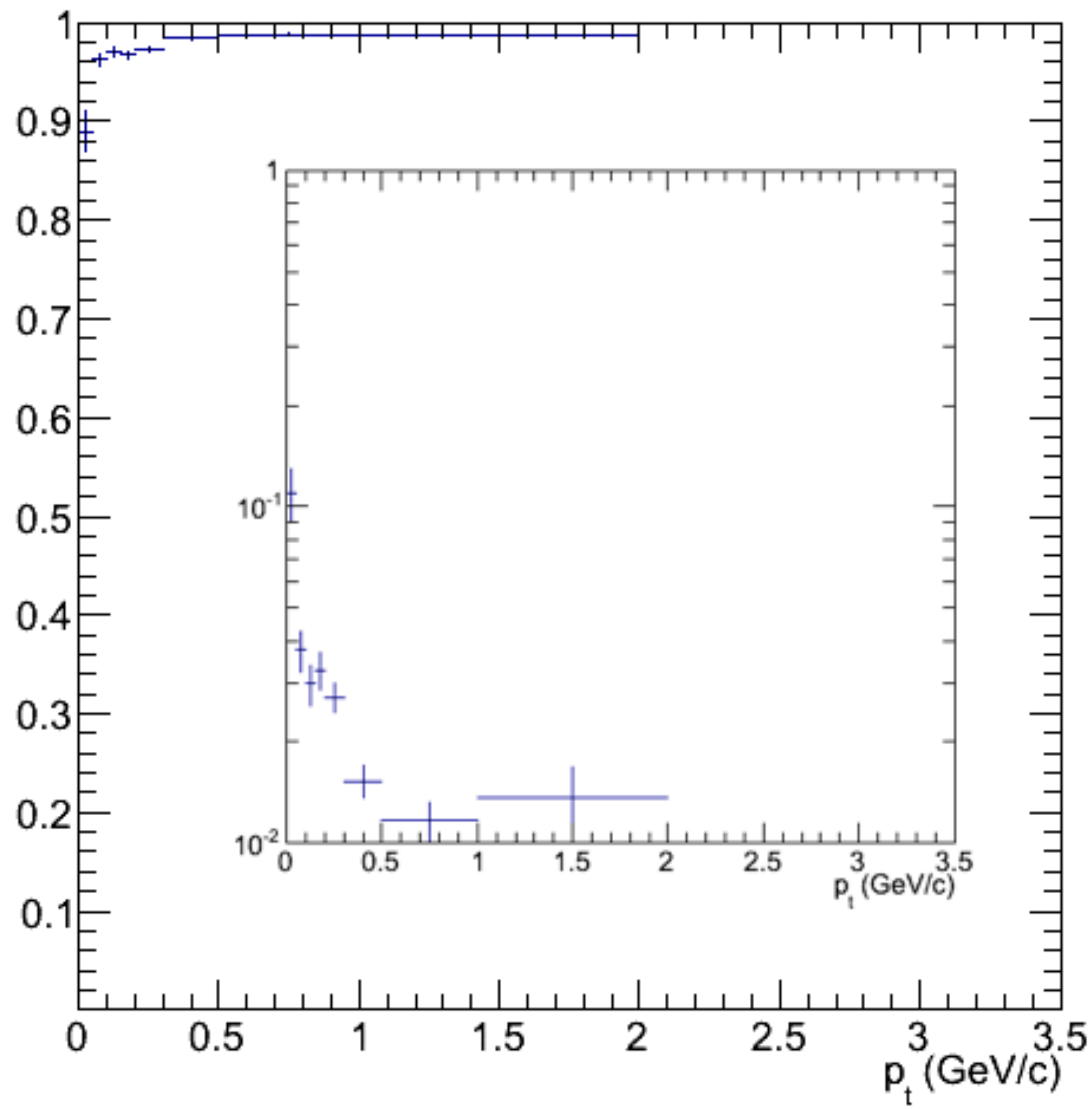


- ➡ **the fraction of MCTrackCands classified as ideal = (96.2±0.1)%**
- ➡ 68% of tracks with p_T < 100 MeV/c are classified as ideal
- ➡ fraction of idealMCTrackCands jumps to 92% for 100 MeV/c < p_T < 200 MeV/c
- ➡ p_T > 1GeV/c around 1% of tracks are not classified as ideal

Remove PXD from Reconstruction

➡ Remove the PXD from reconstruction (TrackFinderMCTruth) and repeat the study

fraction of ideal MCTrackCand VS pt



➡ **the fraction of MCTrackCands classified as ideal = (97.8±0.1)%**

➡ 88% of tracks with p_T < 100 MeV/c are classified as ideal

➡ fraction of idealMCTrackCands jumps to 96% for 100 MeV/c < p_T < 200 MeV/c

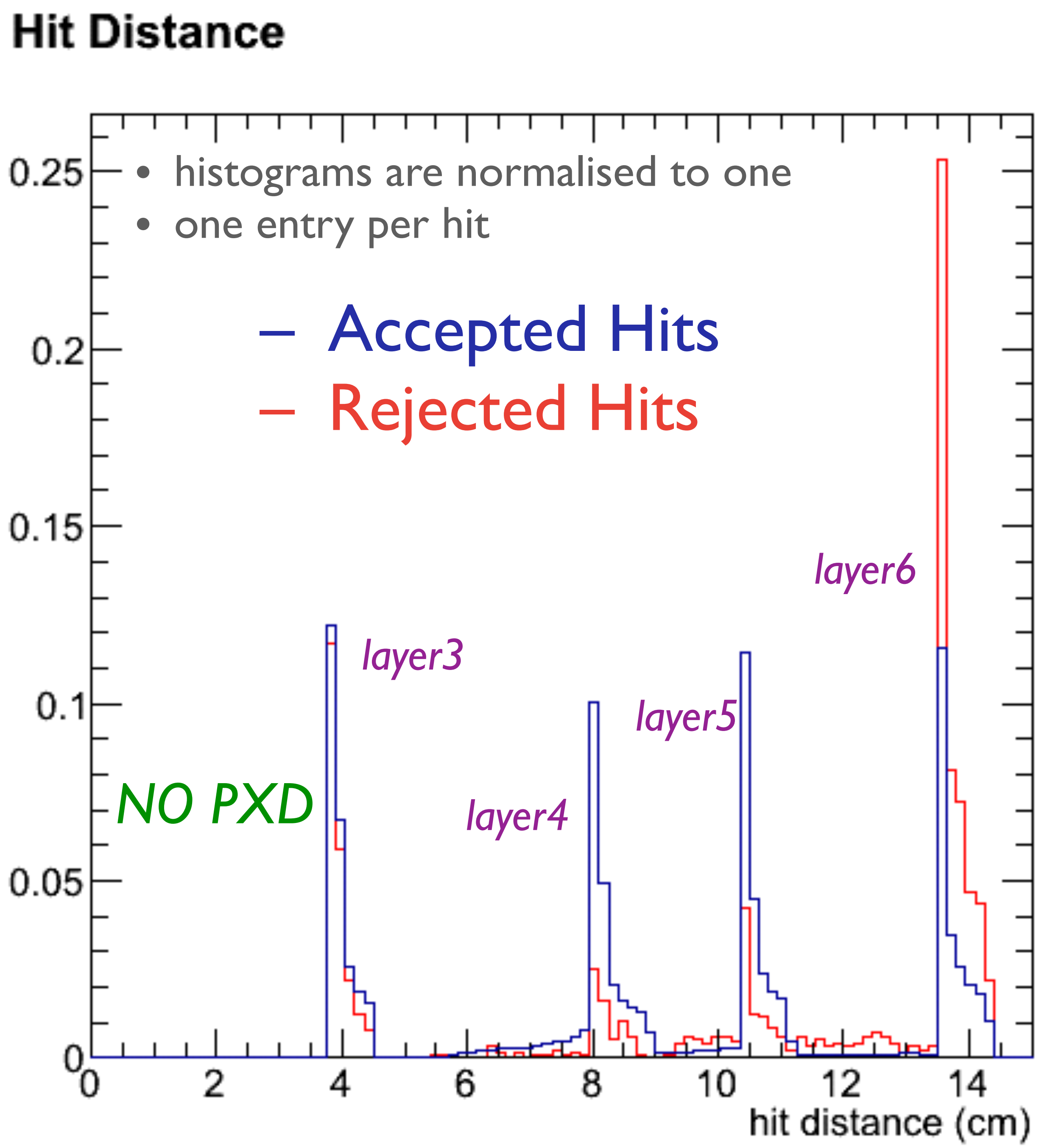
➡ p_T > 1GeV/c around 1% of tracks are not classified as ideal

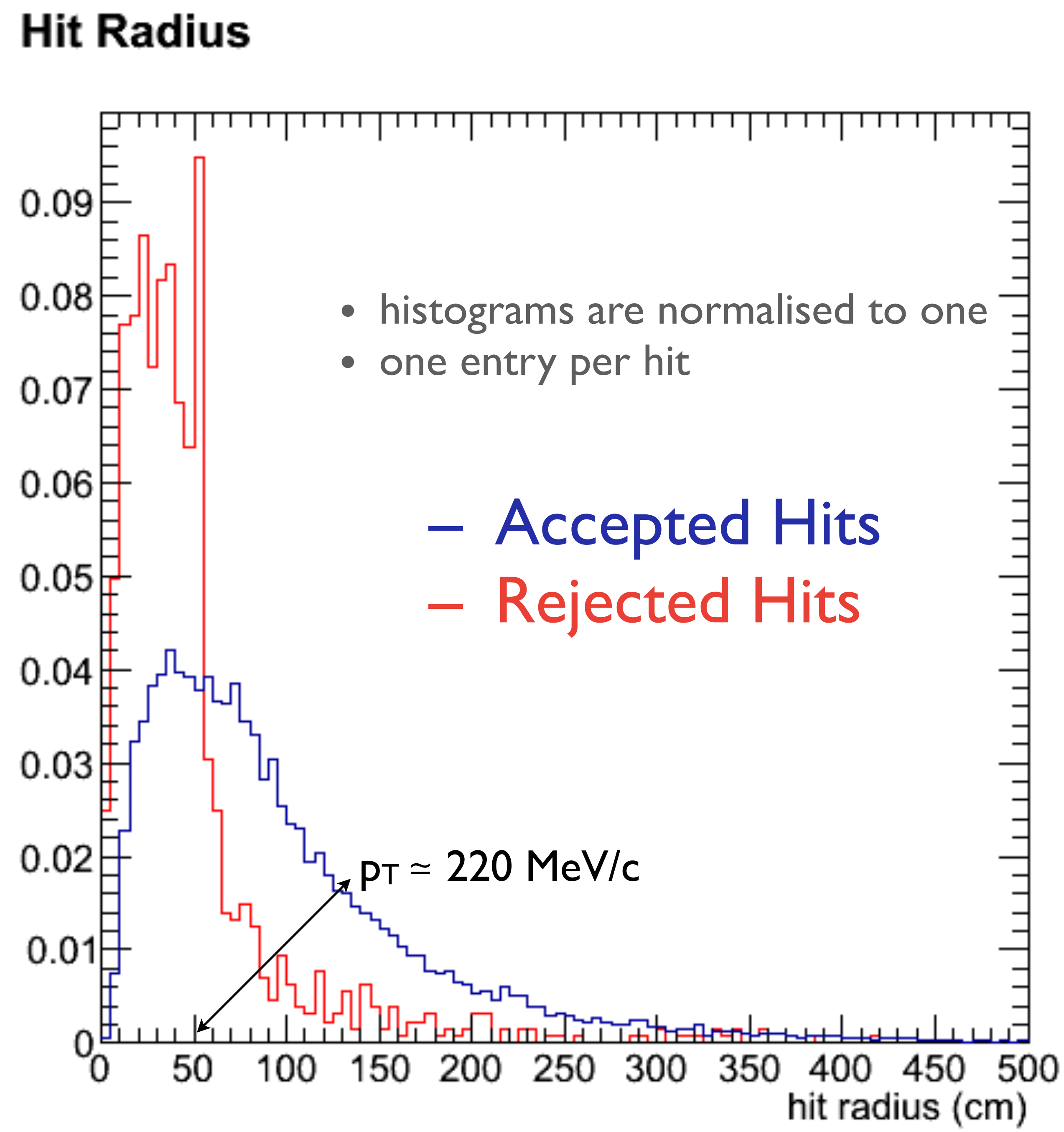
★ the fraction of ideal MCTrackCands is higher than the case with PXD in reconstruction

★ the reason is because some non-ideal MCTrackCands built with the PXD in reconstruction chain are not there anymore (no enough number of 1D infos)

What Hits are Rejected?

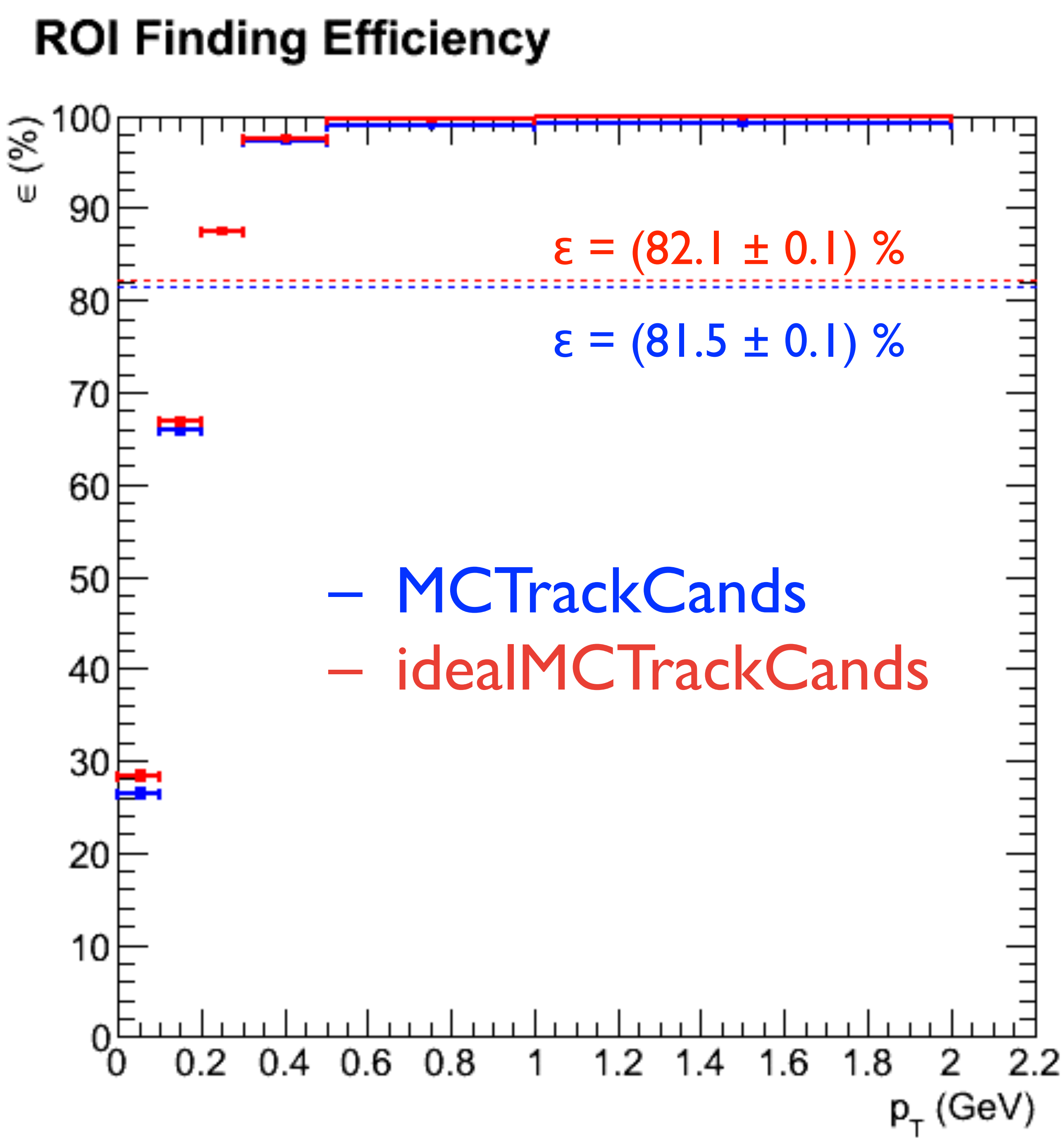
no significant difference in these distributions w/o PXD





- ➡ Most of the reject hits are in Layer6, and a smaller fraction in the wedge part of the SVD
- ➡ Most of the rejected hits belong to low transverse momentum tracks (R < 70 cm ↔ p_T < 300 MeV/c)

What About ROI Finding?



$$\varepsilon = \frac{\text{\# PXDDigits inside a ROI}}{\text{total \# PXDDigits of TrackCand}}$$

➡ ROI Finding efficiency is very slightly improved using the idealMCTrackCands as input

➡ Inefficiency due to failed fits with both MCTrackCands and idealMCTrackCands

★ reasons of the very small increase of efficiency may be:

1. idealMCTrackCand contains *all* the MCTrackCand hits also the hits that have not passed the criteria (can happen in the case the first 5 1D infos pass the criteria)
2. the fitter part is not well configured in the PXDDataReduction Module

Future Improvements

1. add a new criteria: if there is a hit in the wedge part, the hit is not accepted
2. remove the hits from the first one not satisfying any of the criteria
3. correct the computation of θ_{MS} using the incident momentum (from TrueHit)
4. can take into account the (small) energy loss
   - same approach as MS: allow for a "small" energy loss and recompute the helix radius and position
5. classify the non-**ideal**MCTrackCands (**fine**MCTrackCands, **nasty**MCTrackCands)
   - need a (small) redesign of the module, e.g. disentangle the performance analyzer from the core classifier (split the module in 2)
   - what are the parameters that we want to keep configurable by the user?
6. use SelectSubset in order to preserve the relations between TrackCands and other objects
   - can't be used at the moment because the genfit::TrackCand inherits from a TObject and nit from a RelationObject

Conclusions

➡ The first working version of the MCTrackCandClassifier module is on svn.

➡ A few more modifications are needed to actually test the VXDTrackFinder, ROIFinding, …

➡ Next Steps:

in February

- implement 1. and 2. from previous slide (do not accept hits in the wedge part, remove hits not passing the criteria from the idealMCTrackCands)
- test ROIFinding & VXDTrackFinder (Jakob)
- implement the other improvements

➡ PXDDataReduction module need a redesign to improve the fitting part

- why fit twice (PXDDataReduction & GenFitter) on the HLT?

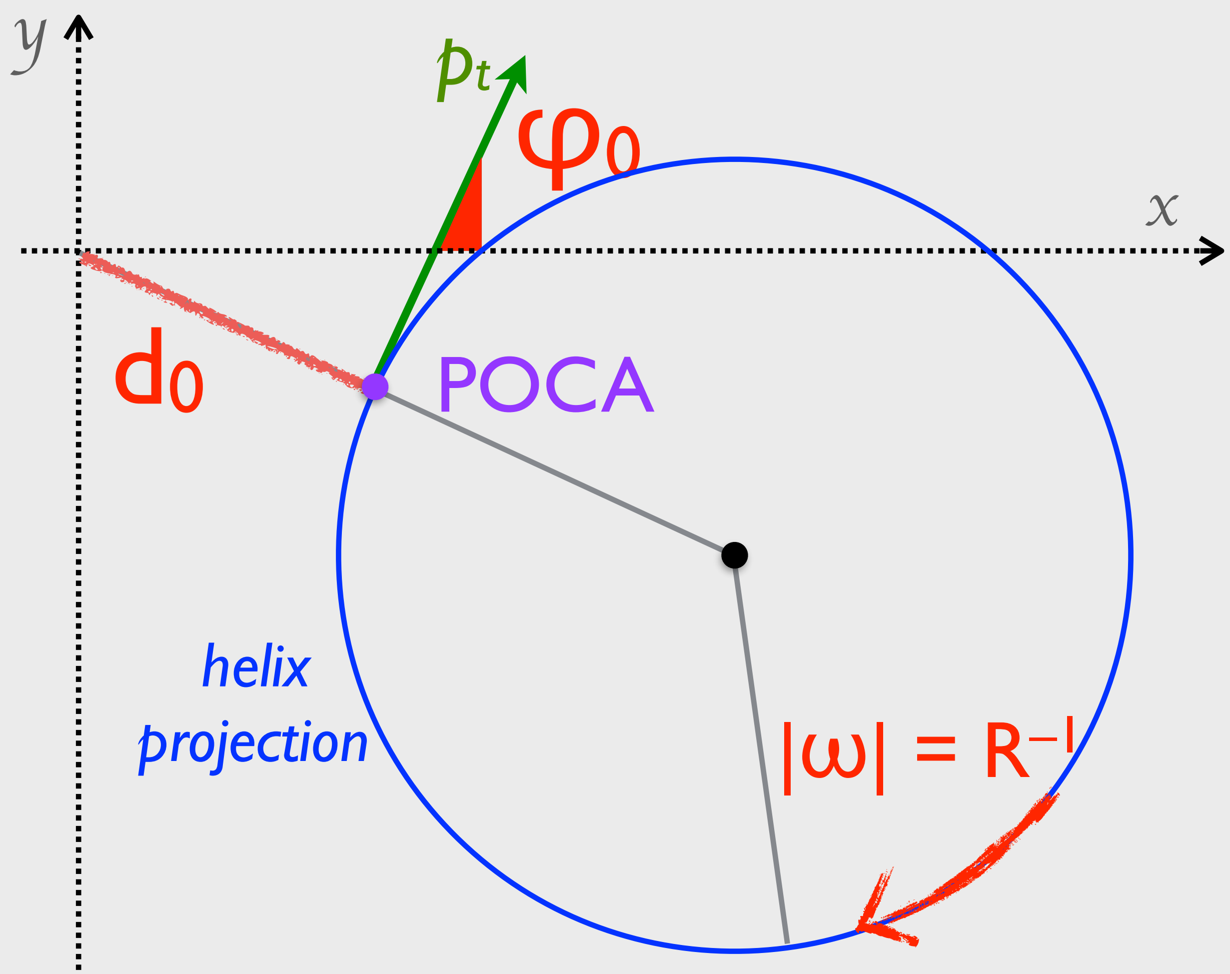
for the next beam test

Thank You!

backup slides

Track Parameterisation

TRANSVERSE PLANE



- POCA = Point Of Closest Approach
- d_0 is the 2d signed distance of the POCA from the z axis, the sign depends on the angular momentum of the track (>0 in the fig.)
- φ_0 is the angle between p_t and the x axis at the POCA, $\varphi_0 \in [-\pi, \pi]$
- the sign of ω, the curvature, is the same as the charge of the track (>0 in the fig.)

LONGITUDINAL VIEW

- $\tan\lambda$ is the ratio of p_z and p_t, $\lambda \in [-\pi, \pi]$
- z_0 is the signed distance of the POCA from the transverse plane