dE/dx & V0 Finder

The last two weeks

Uli Einhaus
Asian Physics & Software Meeting
09.03.2018

HELMHOLTZ
ASSOCIATION

DESY

dE/dx - Compute_dEdxProcessor

- DD4hep geometry issue
- dx calculation strategy
- New processor options:
  - Change truncation values (8%, 30%)
  - Select strategy for dx calculation
  - Generate dE/dx plots for all strategies
  - Weighting each track with sqrt(number of hits) for plots
  - Turn off assining dE/dx to track to only generate plots
- General polishing and documentation
- Intention to upload the new version to github soon

dE/dx – DD4hep geometry issue

- Gear base unit: mm, DD4hep base unit: cm
- Feature: >> TPCdata.rMinReadout / dd4hep::mm ;
- Problem: without specifying the units min radius was at 37 **mm**

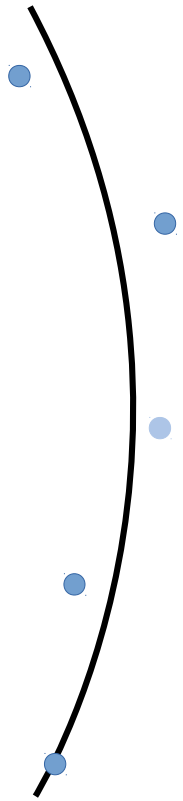
  → Silicon hits were accepted and used for dE/dx calculation

  → usually truncated (lower 8%, upper 30%)

  → only came up when number of hits was low
- Now corrected
- Be aware of DD4hep units!

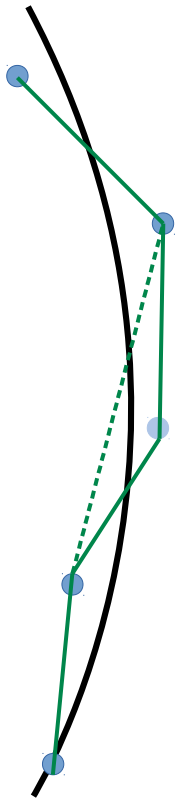
dE/dx – dx calculation strategies

- 1 (so far): use real distance between track hit centers
- 2: use helix path length of projected hits (points on the helix closest to the hit position)
  - Gets rid of hit-to-hit position fluctuation
  - Can be acquired from class MarlinUtil::SimpleHelix
  - Performs worse than strategy 1
- 3: use helix path length over the row height of the hit row
  - Gets rid of missing-hits problem, uses all hits
  - Calculate crossing point of helix with cylinder at upper and lower row edge (hit radius +/- half pad height)
  - Get helix path length between those crossing points
  - Performs similar to strategy 1



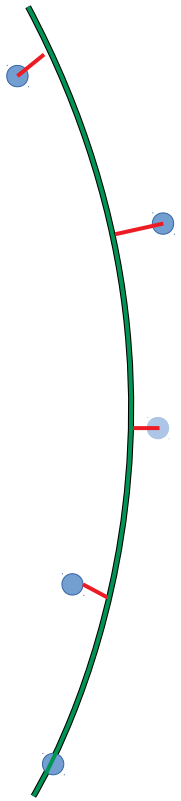
dE/dx – dx calculation strategies

- 1 (so far): use real distance between track hit centers
- 2: use helix path length of projected hits (points on the helix closest to the hit position)
  - Gets rid of hit-to-hit position fluctuation
  - Can be acquired from class MarlinUtil::SimpleHelix
  - Performs worse than strategy 1
- 3: use helix path length over the row height of the hit row
  - Gets rid of missing-hits problem, uses all hits
  - Calculate crossing point of helix with cylinder at upper and lower row edge (hit radius +/- half pad height)
  - Get helix path length between those crossing points
  - Performs similar to strategy 1



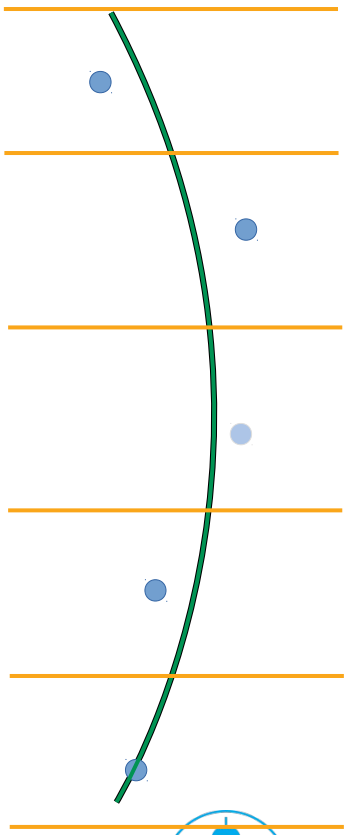
dE/dx – dx calculation strategies

- 1 (so far): use real distance between track hit centers
- 2: use helix path length of projected hits (points on the helix closest to the hit position)
  - Gets rid of hit-to-hit position fluctuation
  - Can be acquired from class MarlinUtil::SimpleHelix
  - Performs worse than strategy 1
- 3: use helix path length over the row height of the hit row
  - Gets rid of missing-hits problem, uses all hits
  - Calculate crossing point of helix with cylinder at upper and lower row edge (hit radius +/- half pad height)
  - Get helix path length between those crossing points
  - Performs similar to strategy 1

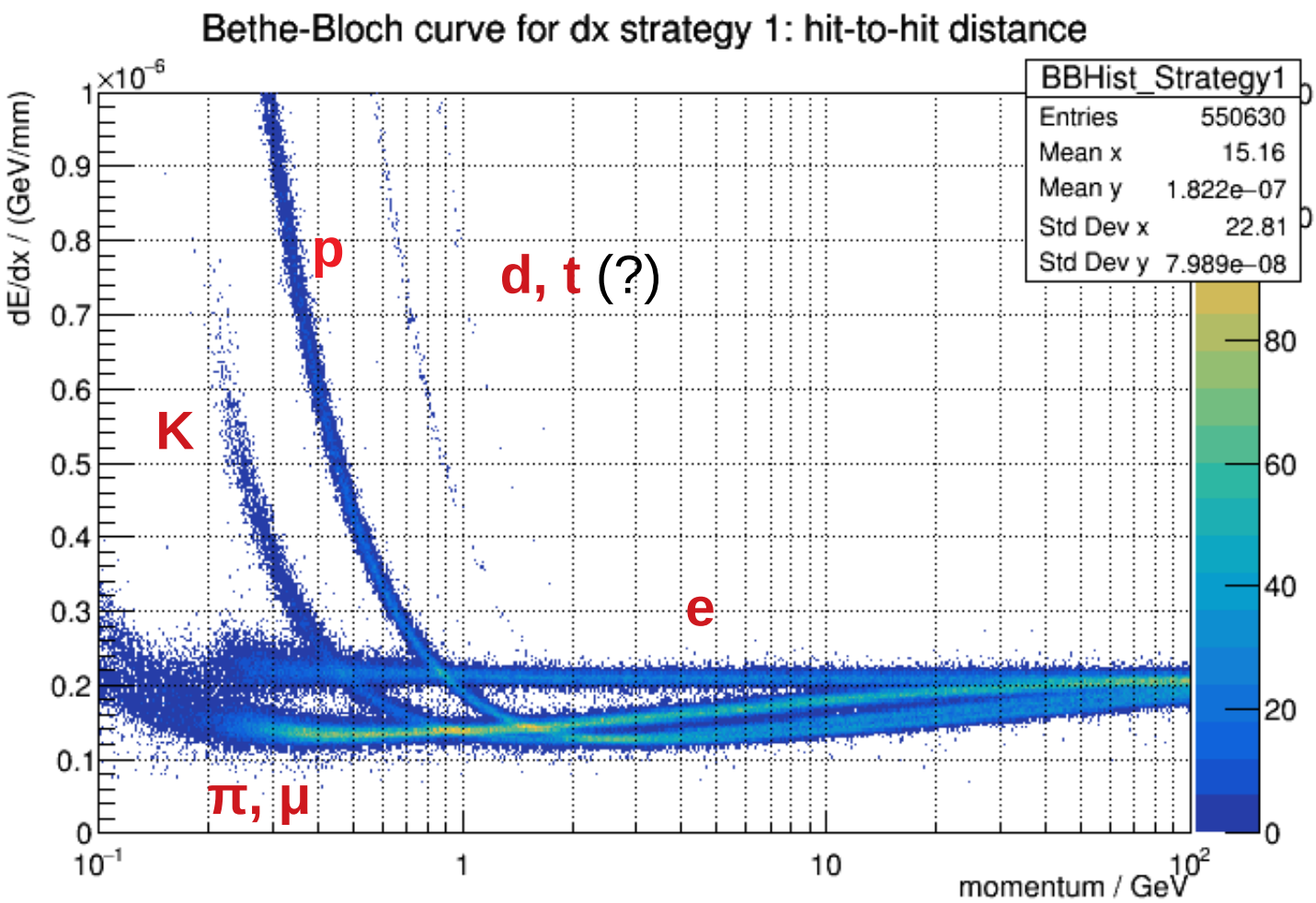


dE/dx – dx calculation strategies

- 1 (so far): use real distance between track hit centers
- 2: use helix path length of projected hits (points on the helix closest to the hit position)
  - Gets rid of hit-to-hit position fluctuation
  - Can be acquired from class MarlinUtil::SimpleHelix
  - Performs worse than strategy 1
- 3: use helix path length over the row height of the hit row
  - Gets rid of missing-hits problem, uses all hits
  - Calculate crossing point of helix with cylinder at upper and lower row edge (hit radius +/- half pad height)
  - Get helix path length between those crossing points
  - Performs similar to strategy 1

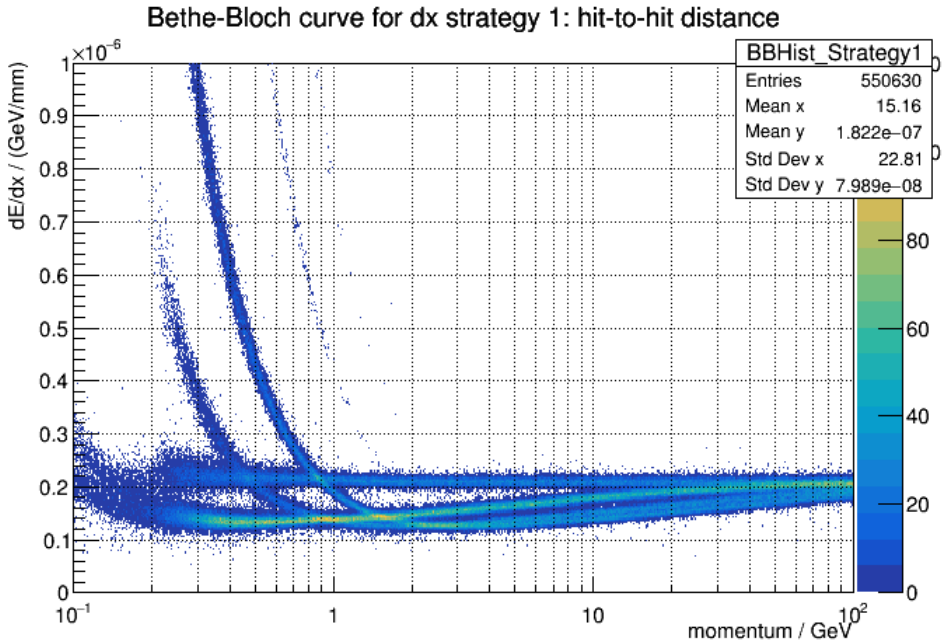


dE/dx – Plots:

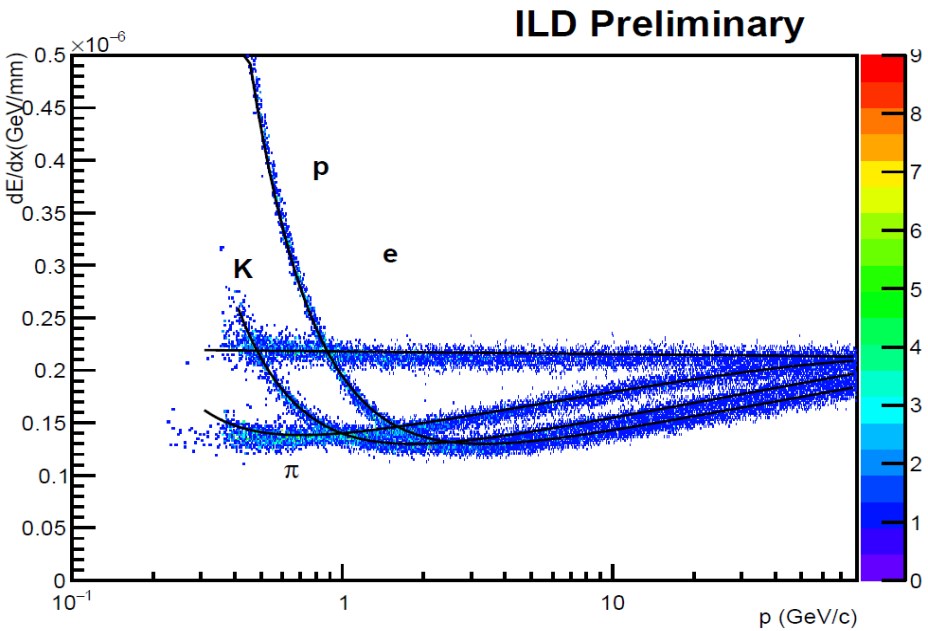

Bethe-Bloch curve for dx strategy 1: hit-to-hit distance
BBHist_Strategy1
Entries 550630
Mean x 15.16
Mean y 1.822e−07
Std Dev x 22.81
Std Dev y 7.989e−08
dE/dx / (GeV/mm)
×10⁻⁶
momentum / GeV
p
d, t (?)
K
e
π, μ

dE/dx – Plots: comparing strategies

- Used single particle random momentum rec-files, 100k events for each PDG (electrons, muons, pions, kaons, protons)



Generated with dE/dx processor

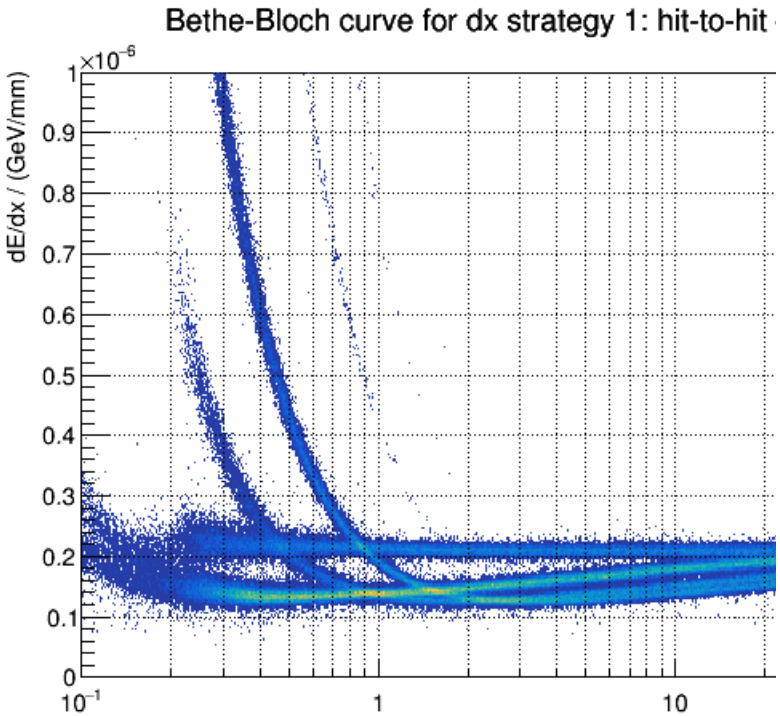


Plot by M. Kurata with standard tools

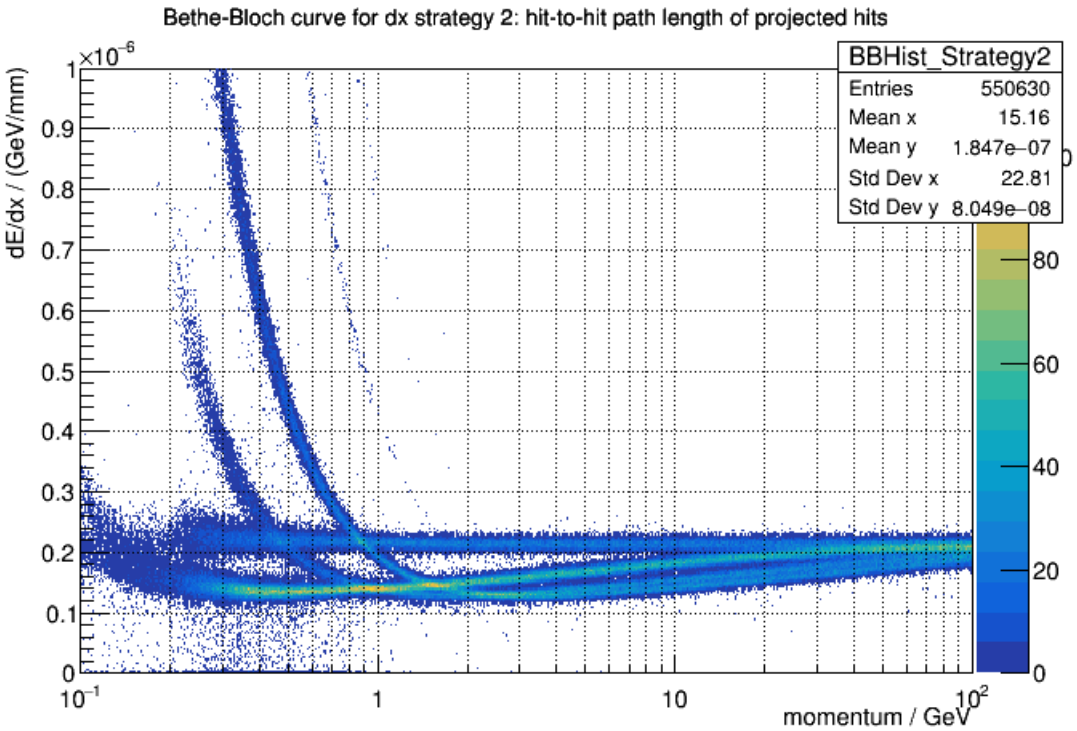
Different binning and reconstruction stages!
To do: particle separation plots

dE/dx – Plots: comparing strategies

- Used single particle random momentum rec-files, 100k events for each PDG (electrons, muons, pions, kaons, protons)



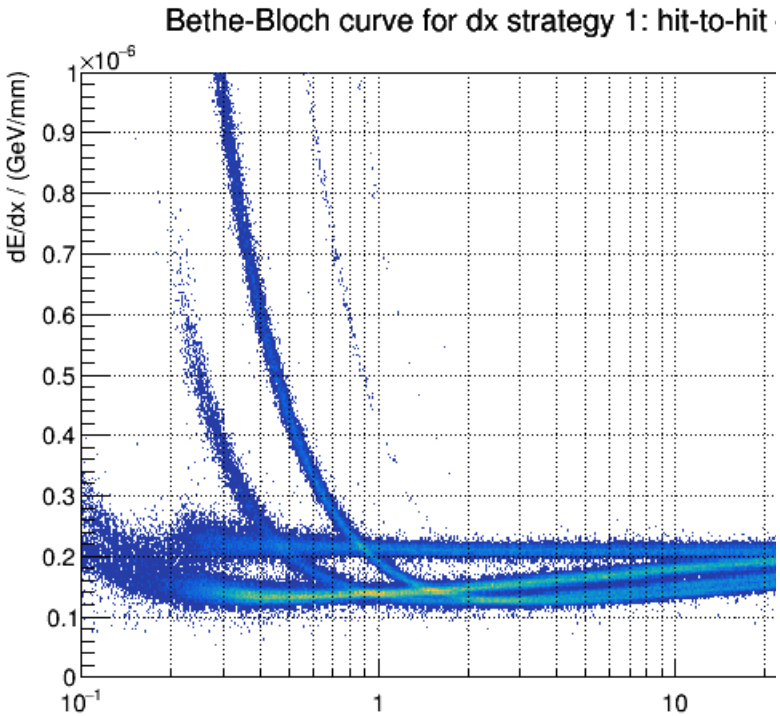
Strategy 1



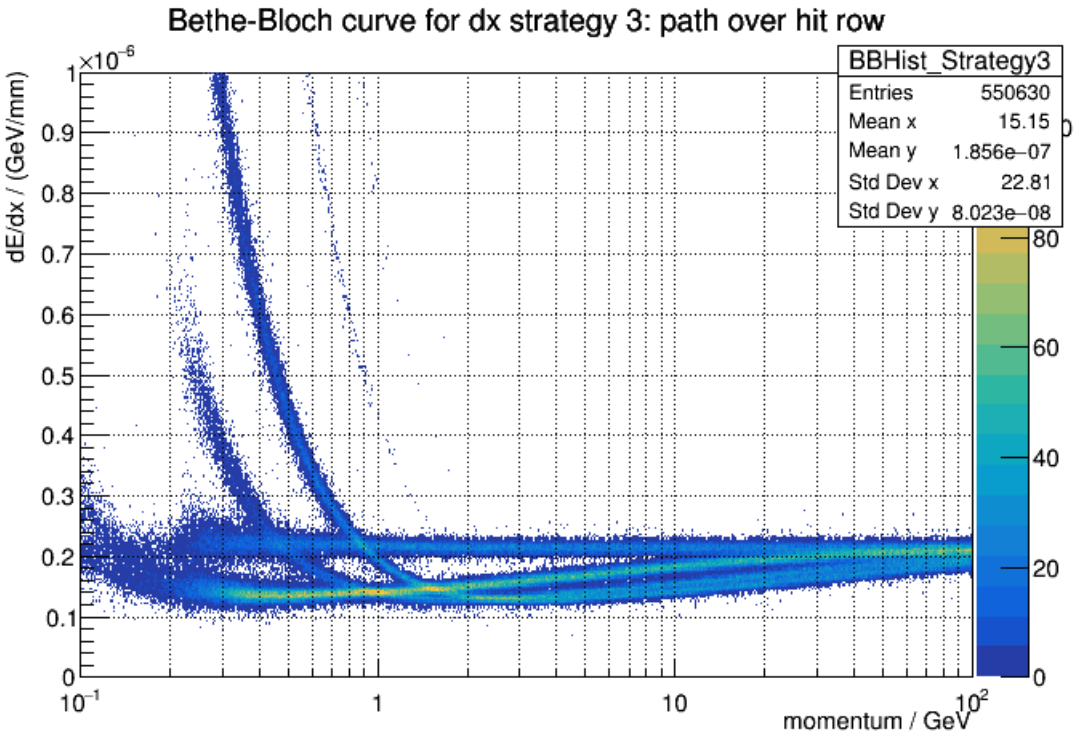
Strategy 2

dE/dx – Plots: comparing strategies

- Used single particle random momentum rec-files, 100k events each PDG

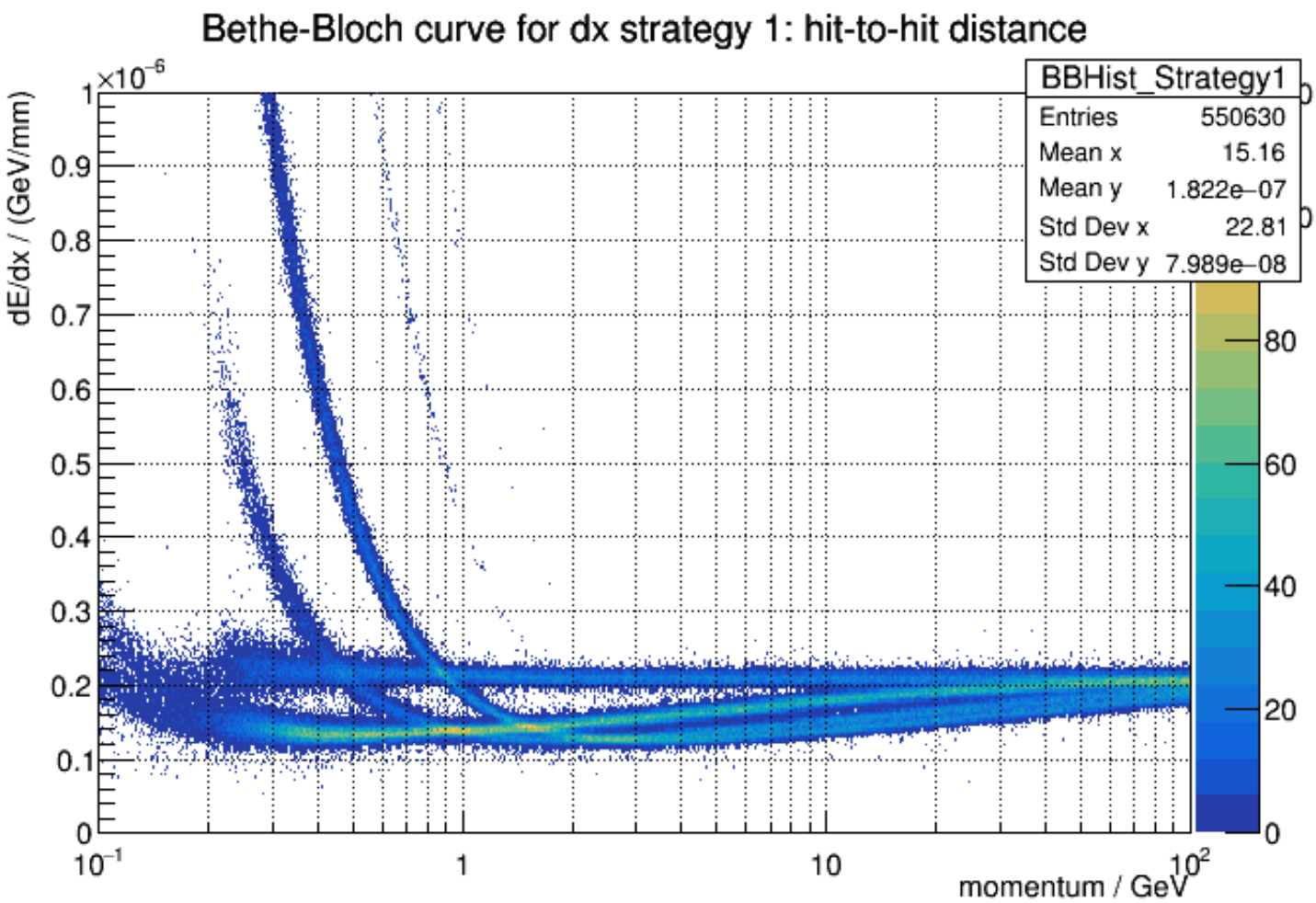


Strategy 1



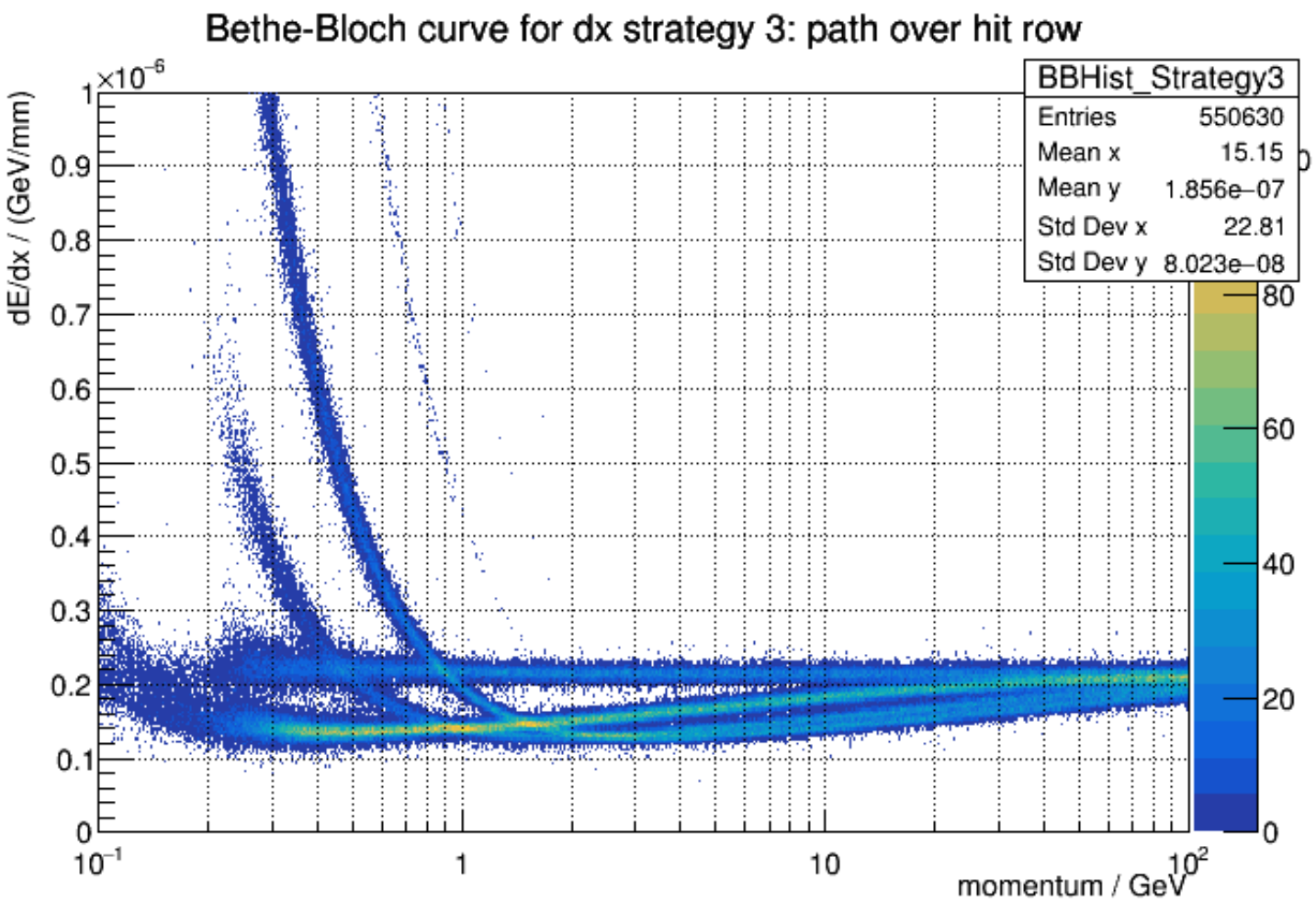
Strategy 3

dE/dx – Plots: comparing strategies



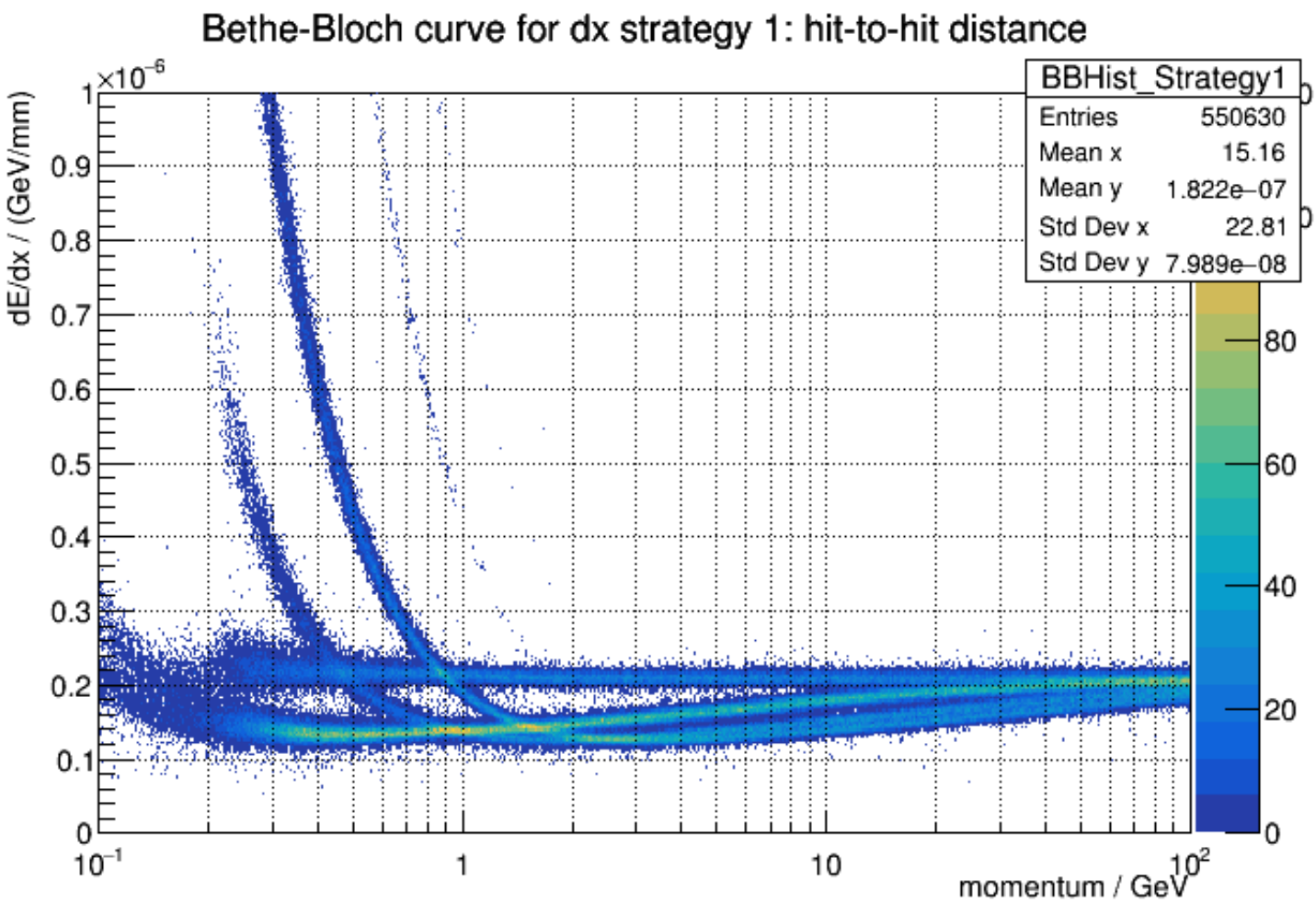
Strategy 1

dE/dx – Plots: comparing strategies



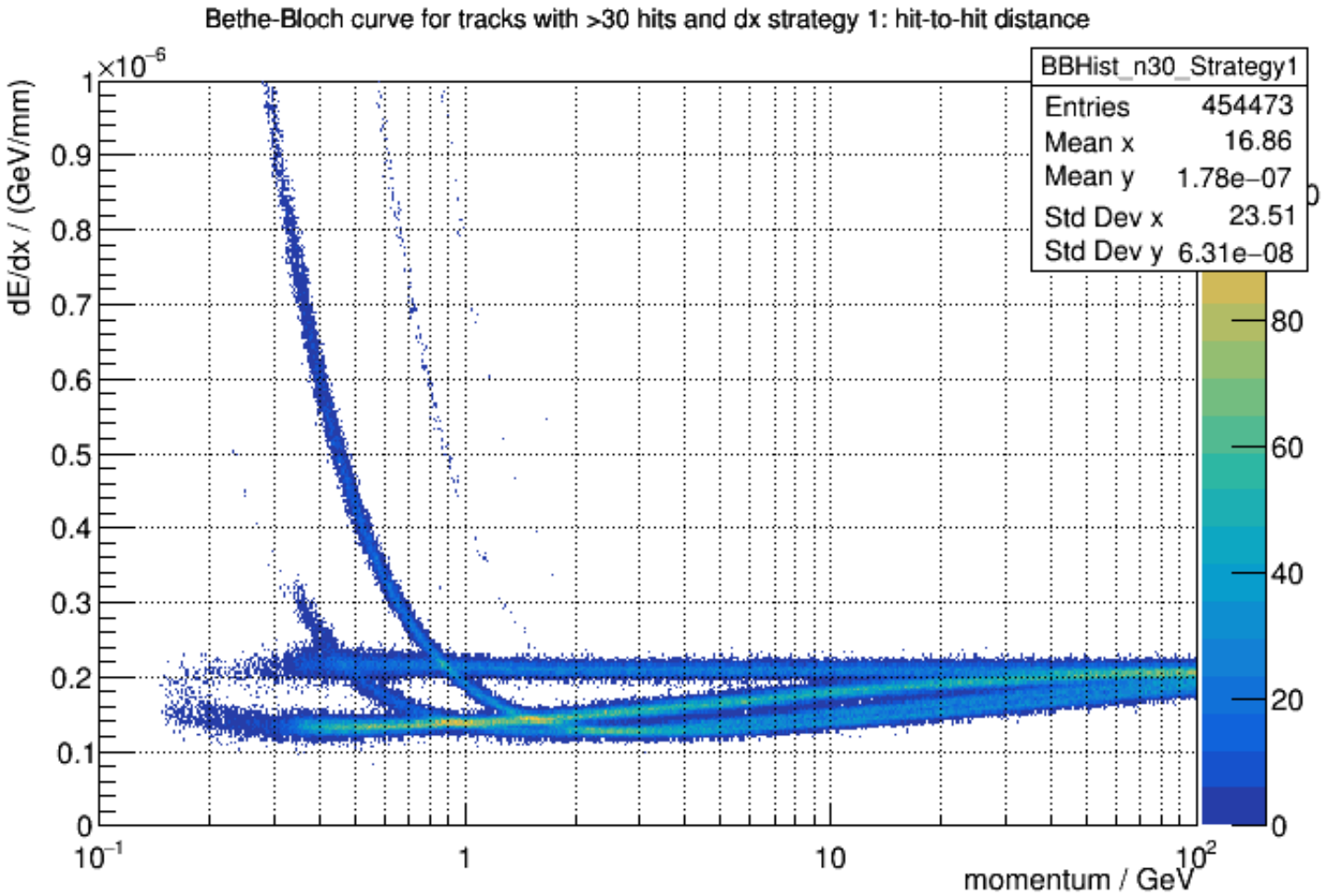
Strategy 3

dE/dx – Plots: comparing cut >=30 hits



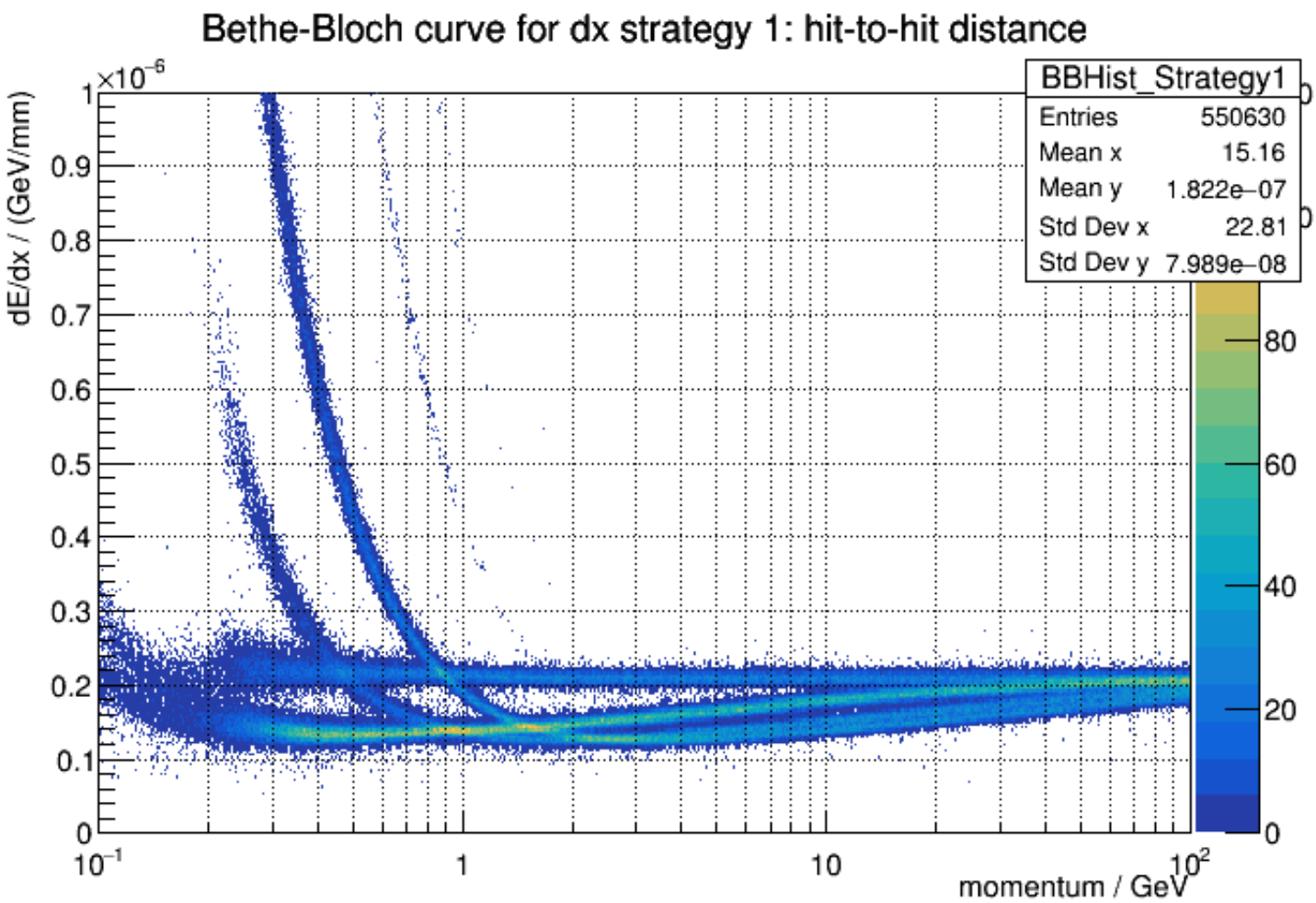
All tracks

dE/dx – Plots: comparing cut >=30 hits



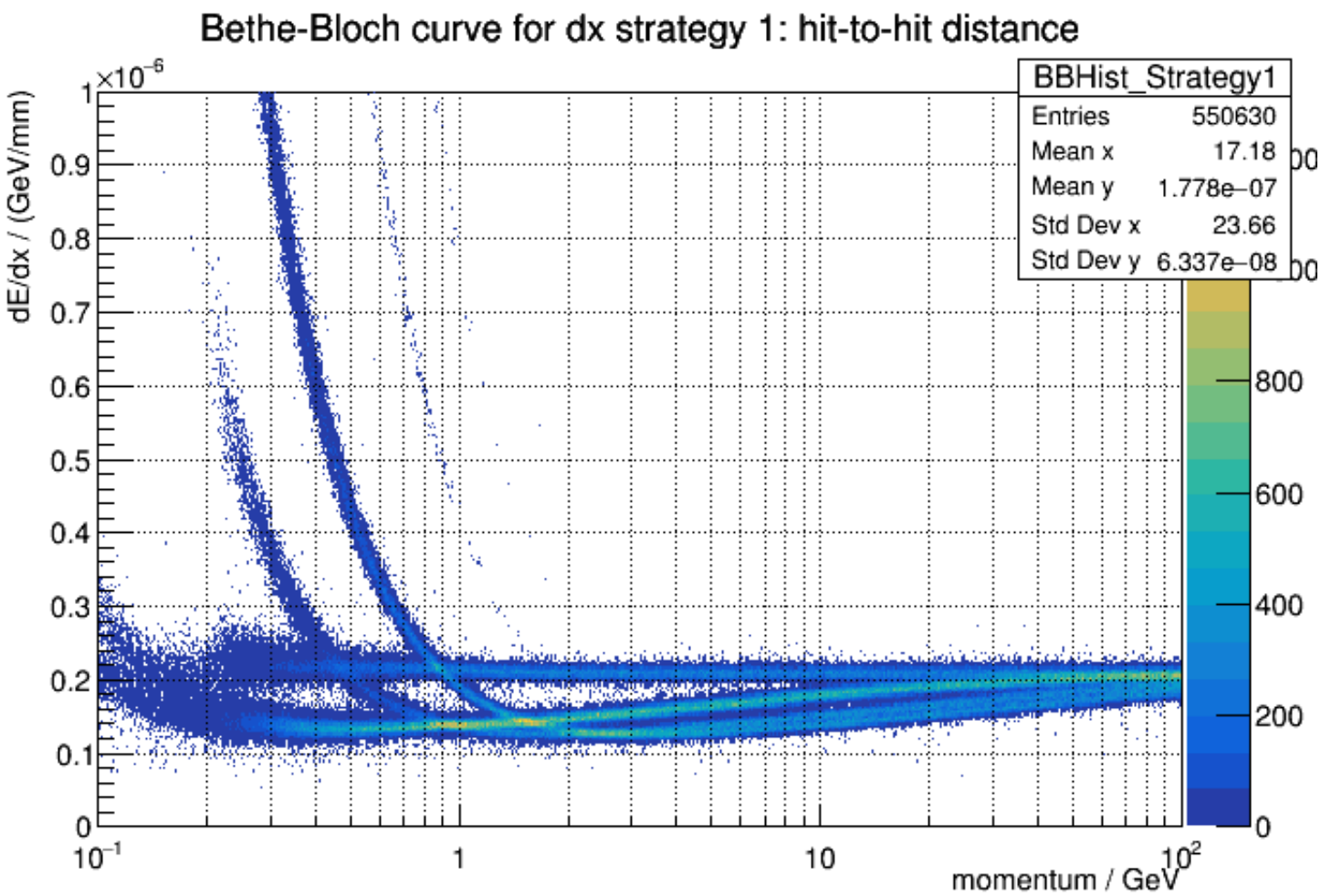
Only tracks with >= 30 hits

dE/dx – Plots: comparing weighting



Without weighting

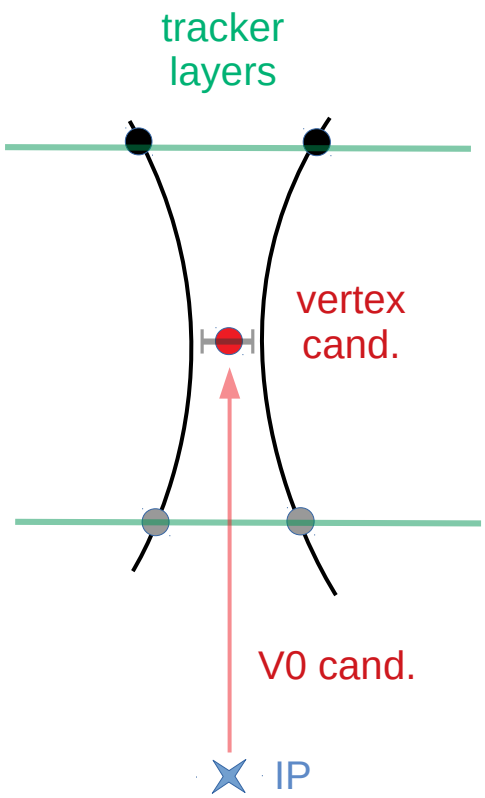
dE/dx – Plots: comparing weighting



With sqrt(nHits) weighting

V0Finder

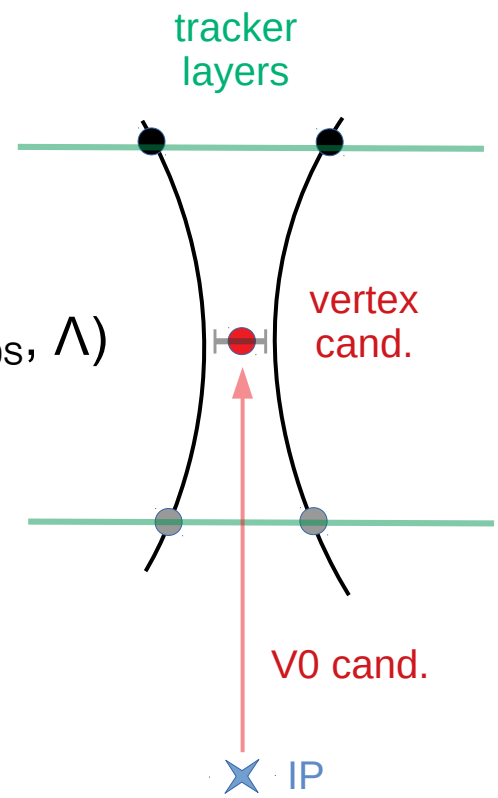
- Part of Marlin TrackingReco
- Works on reconstructed tracks, cut-based, no fitting
- Takes all pairs of reconstructed tracks, calculates their point of closest approach (vertex candidate) and combines the 4-vectors to V0 candidates using corresponding rest mass hypotheses



Cut defaults

(radius = xy-distance from IP)

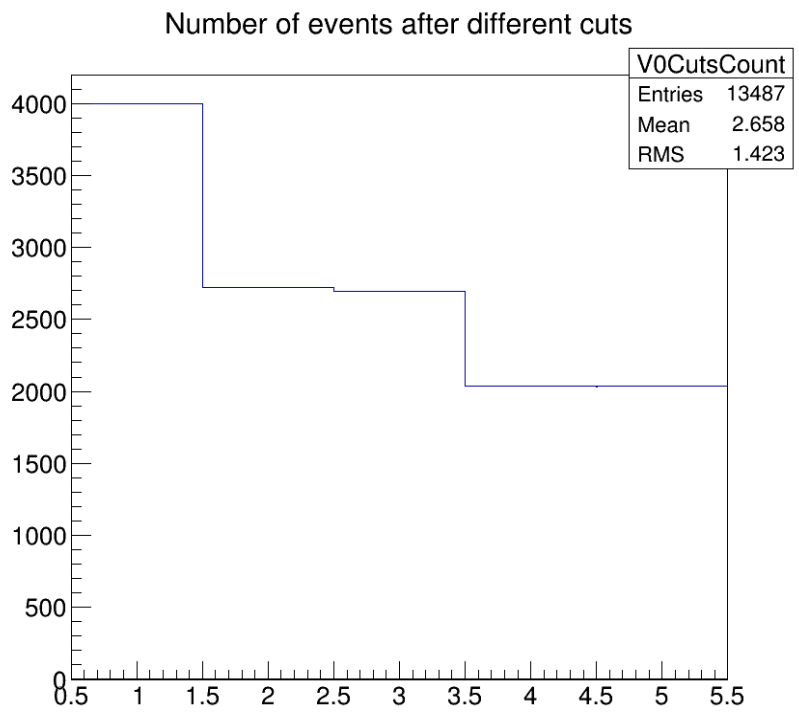
- radius of innermost hit of tracks > 0.7 V0 radius
- distance of reconstructed tracks < 1.5 mm
- reconstructed mass within 10 MeV (γ) or 20 MeV (K_{0S}, Λ)
- V0 radius > {cut radius}
  - 10 mm (γ) - beam pipe at ~ 15 mm
  - 30 mm (K_{0S}) - τ·c = 27 mm
  - 50 mm (Λ) - τ·c = 78 mm



Procedure

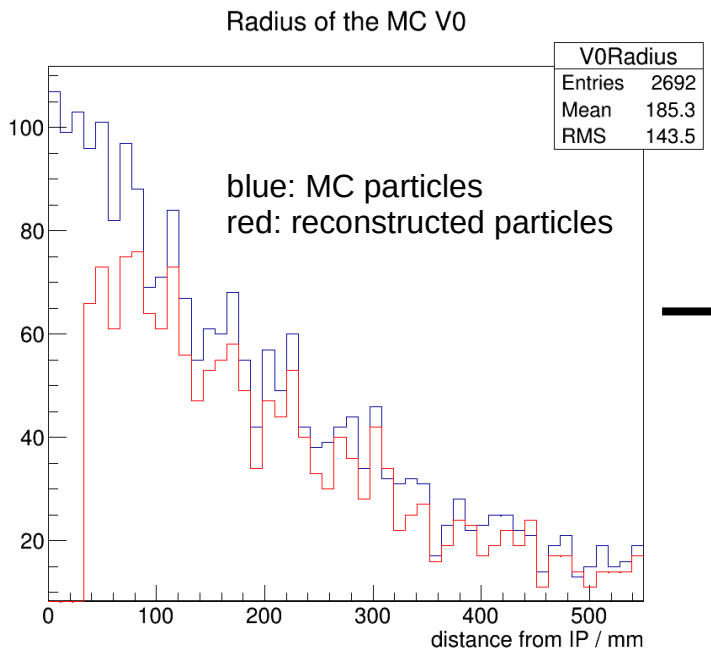
- Used ddsim particle gun to simulate 4000 kaons and lambdas
- p = [1,2,3,4,5,6,7,8,9,10] GeV; RCut = [5,10,20,30,50] mm
- Efficiency: correctly reconstructed / more than 2 tracks found

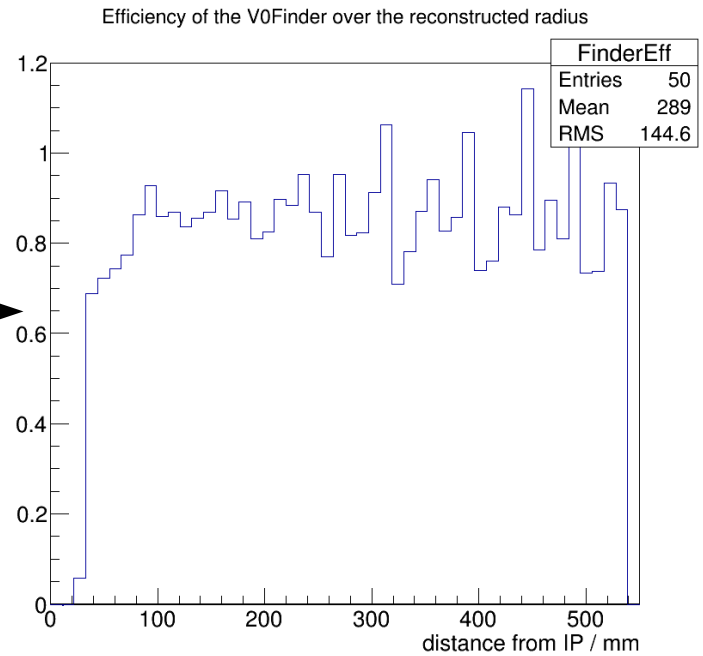
1: found MC V0
2: V0 decayed in charged particles
3: number of reco. tracks > 1
4: at least 1 correct V0 found
5: + only one reco. V0



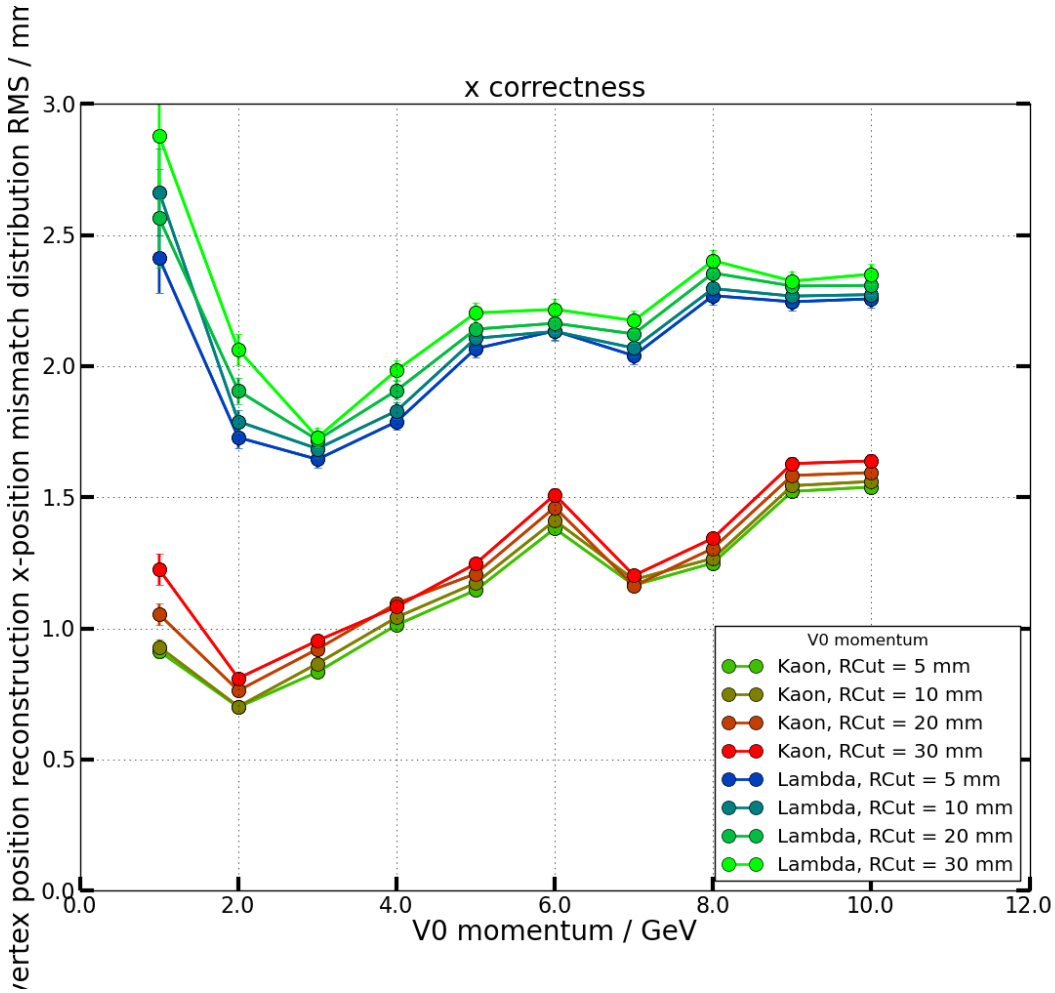
Efficiency plots

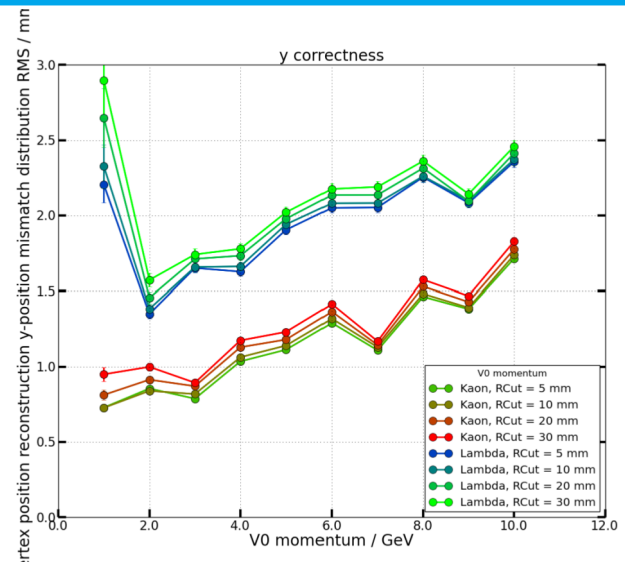
- Efficiency: correctly reconstructed / more than 2 tracks found
- Here: efficiency over distance from IP
- Also: total efficiency (cut 4 / cut 3)

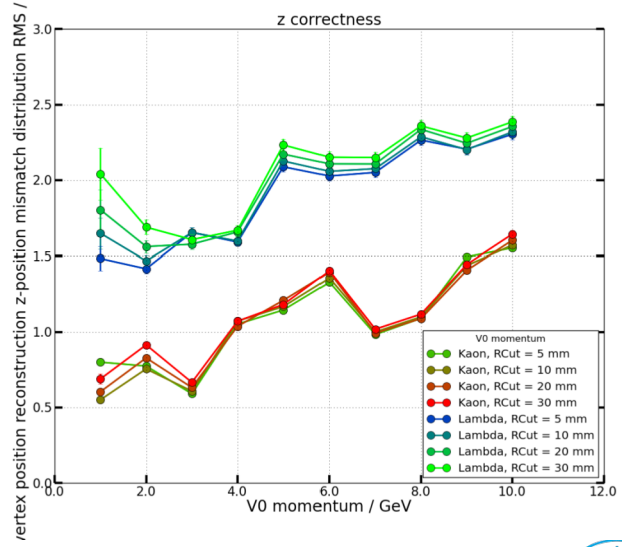




RMS of the position difference between MC and Reco

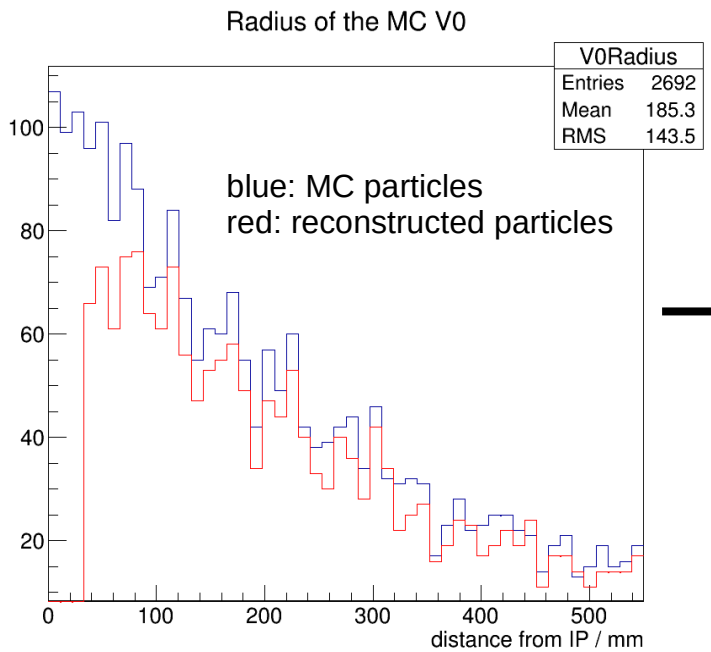


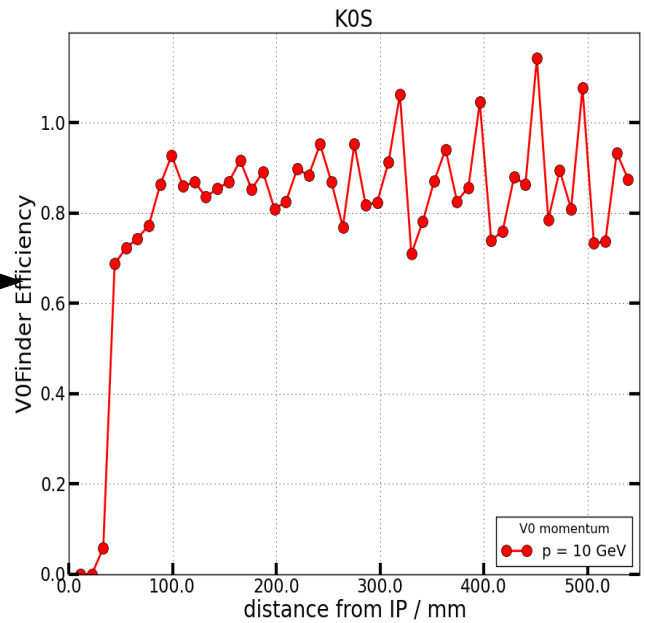




Efficiency plots

- Efficiency: correctly reconstructed / more than 2 tracks found
- Here: efficiency over distance from IP
- Also: total efficiency (cut 4 / cut 3)

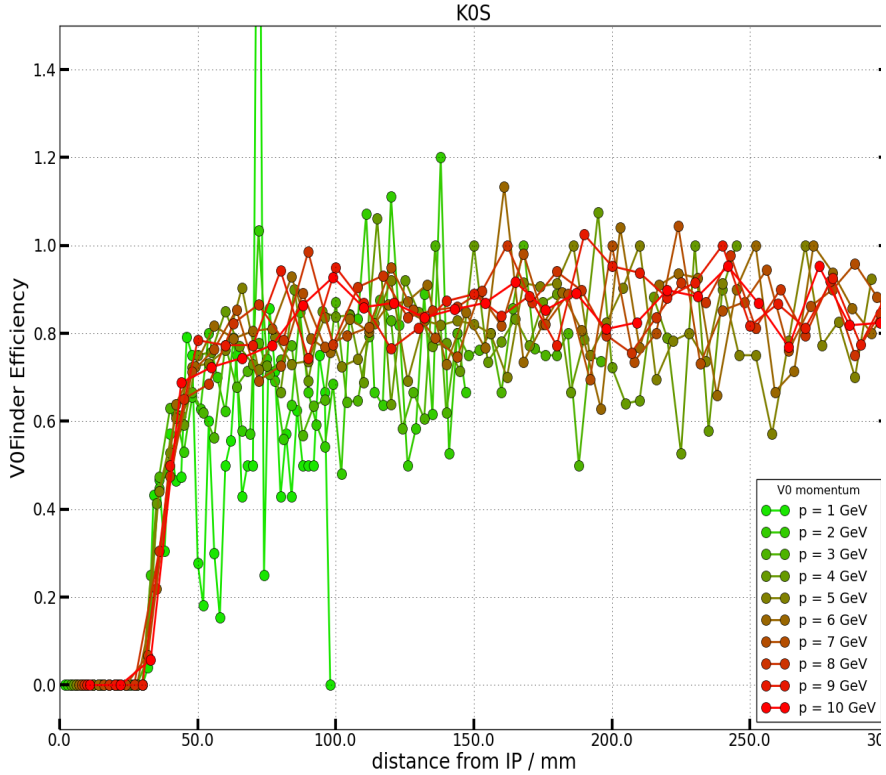




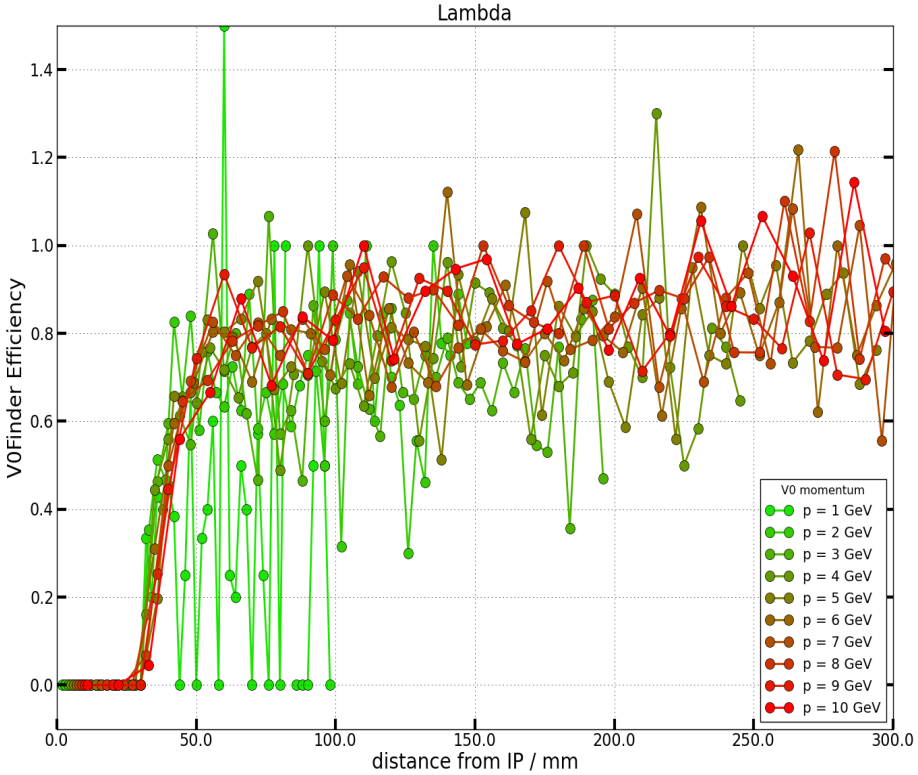
Efficiency over distance from IP

Plots for RCut = 30 mm

Kaon

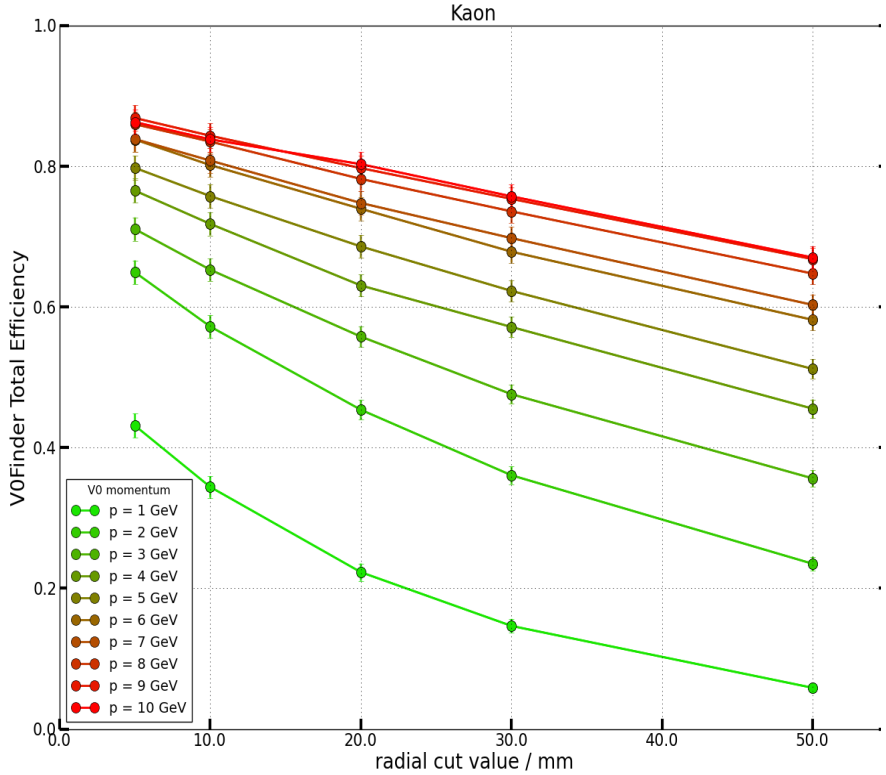


Lambda

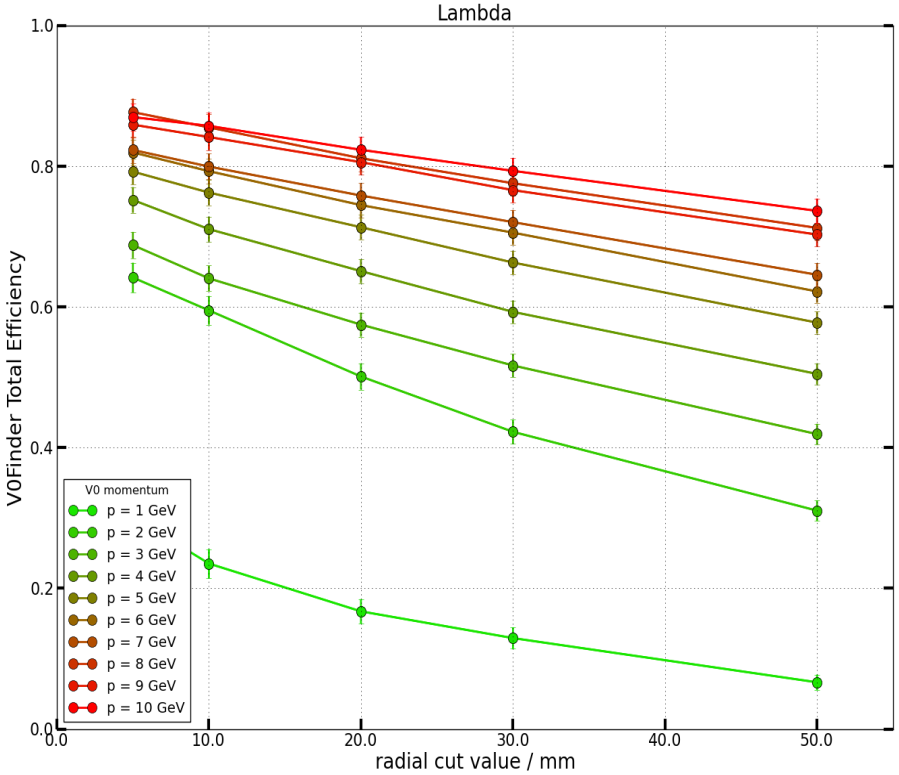


Total efficiency, over RCut

Kaon

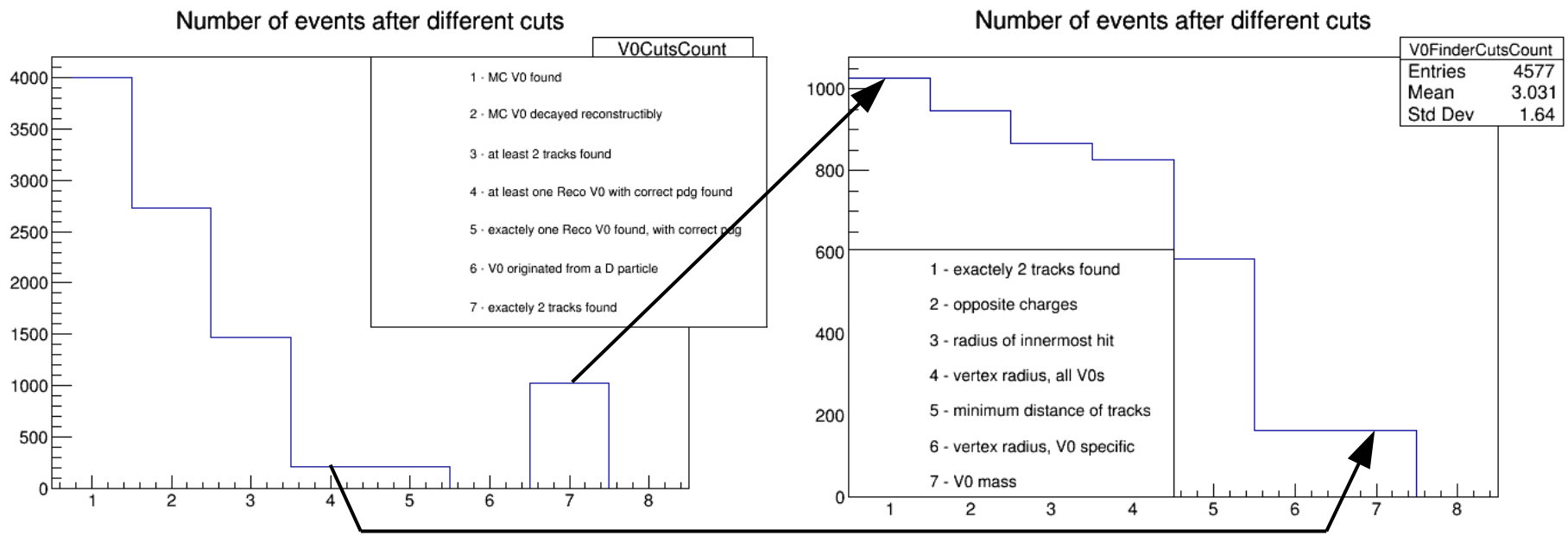


Lambda

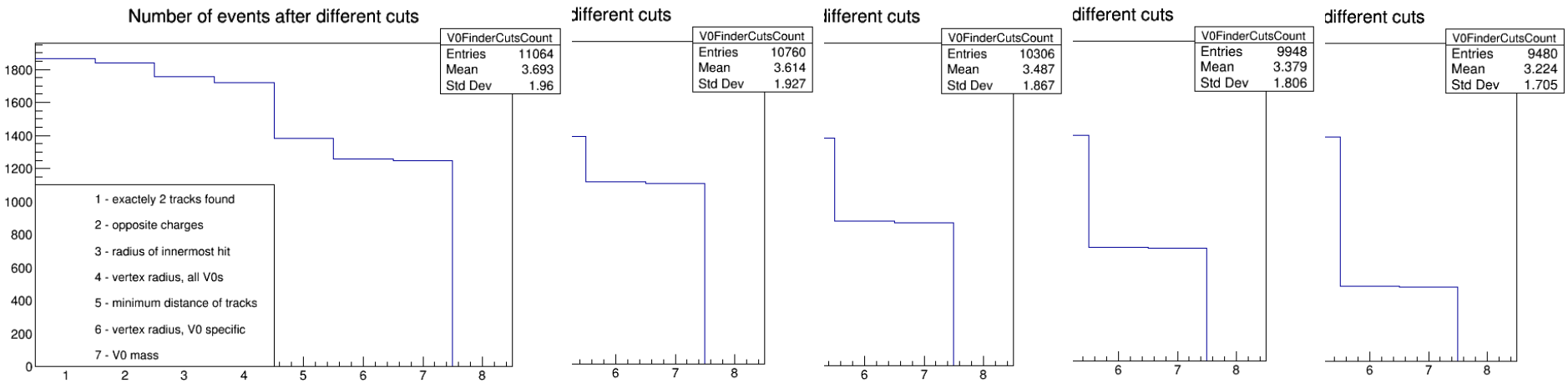


New Plot

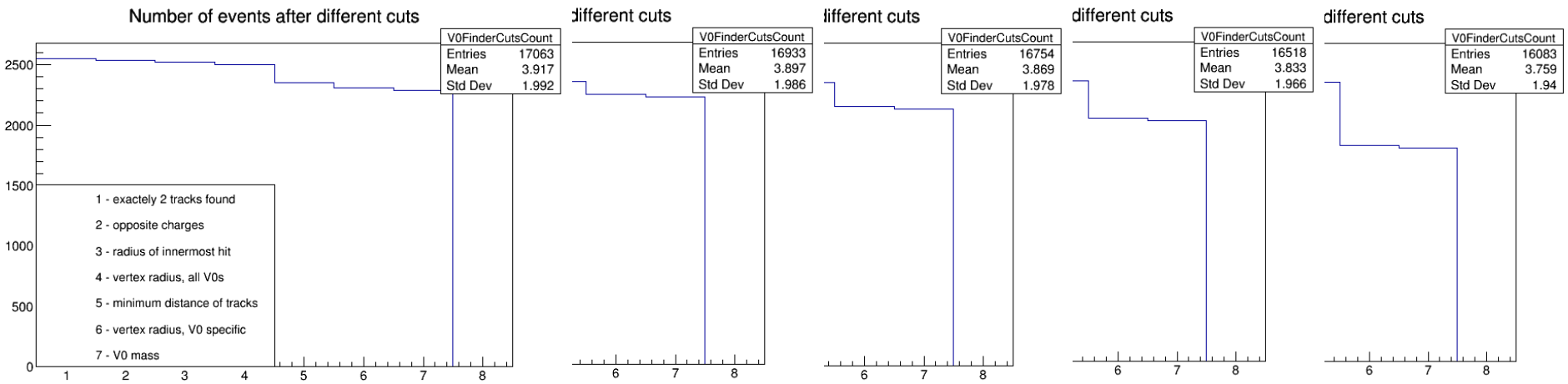
- Old: cut flow in several steps before and including the V0Finder
- New: cut flow along each step in the V0Finder
- Example: Kaons, p = 1 GeV, rCut = 30 mm



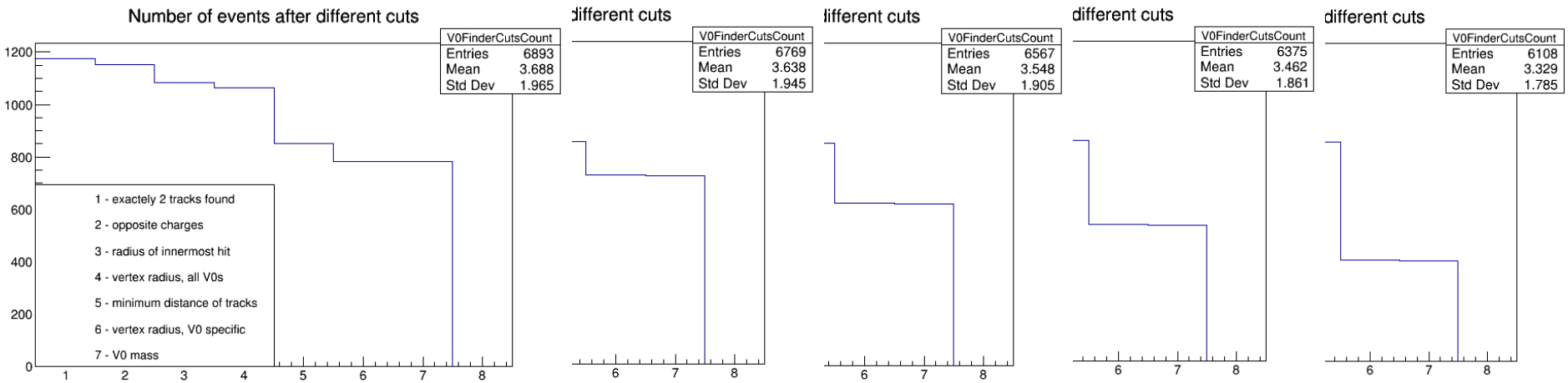
Cut flow for Kaons – 2 GeV and 10 GeV



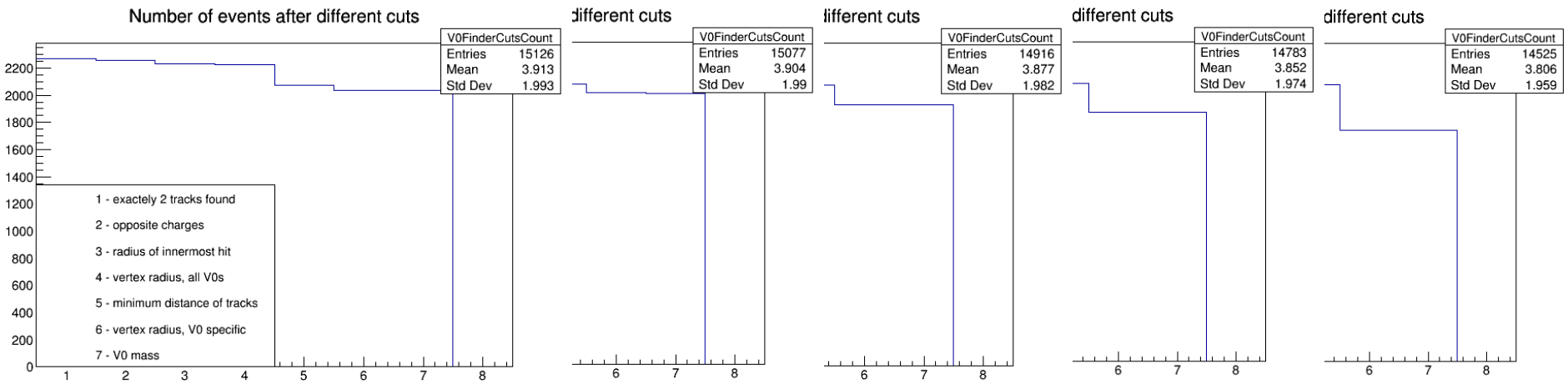
rCut = 5 mm 10 mm 20 mm 30 mm 50 mm



Cut flow for Lambdas – 2 GeV and 10 GeV



rCut = 5 mm 10 mm 20 mm 30 mm 50 mm

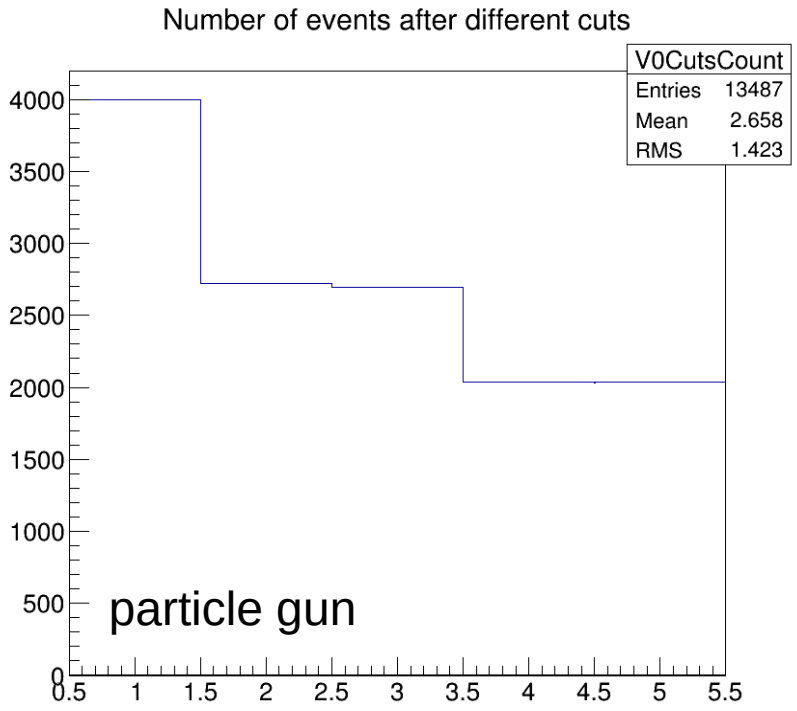


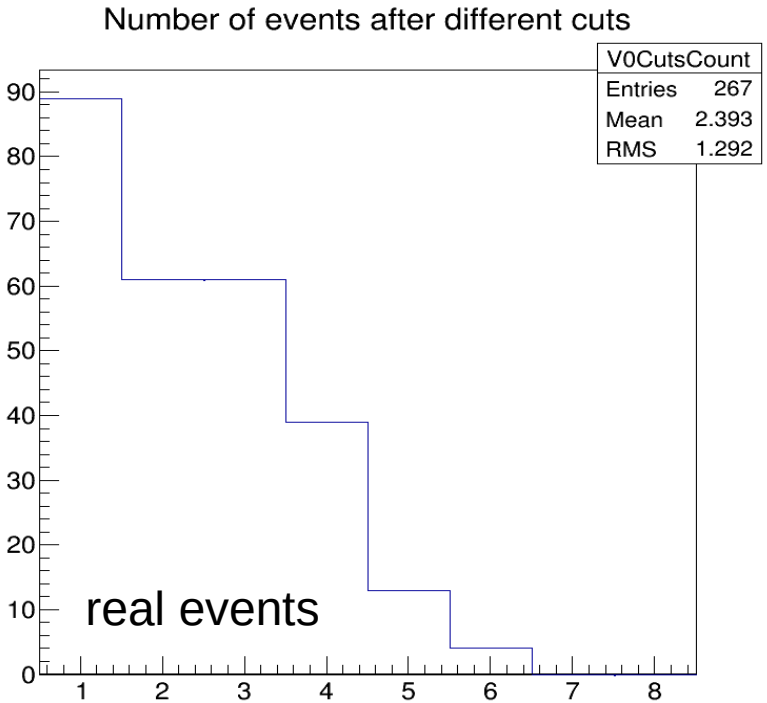
Backup:
Sw&Ana Premeeting slides

D0Finder

- Used full simulation of HWW*, 100 events
- Target: look at D0 decays
- Do V0s change?

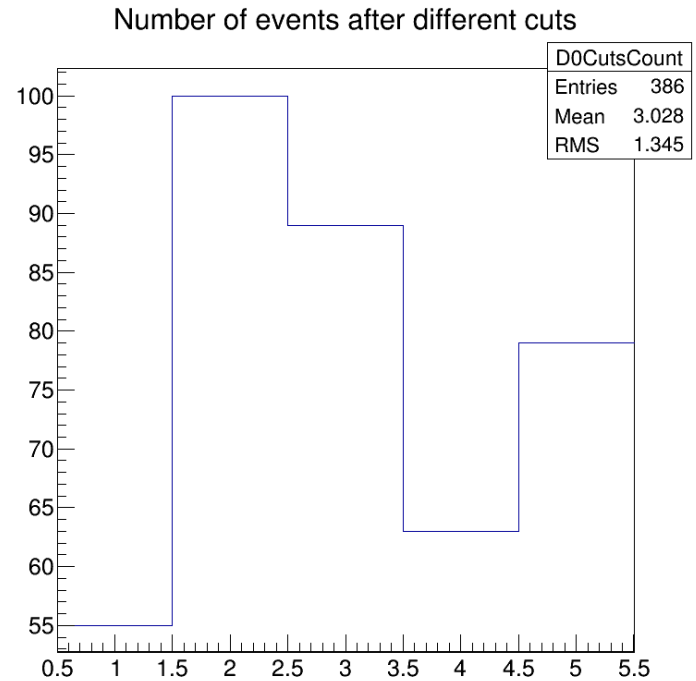
1: found MC V0
2: V0 decayed in charged particles
3: number of reco. tracks > 1
4: at least 1 correct V0 found
5: + only one reco. V0
6: originated from a D0

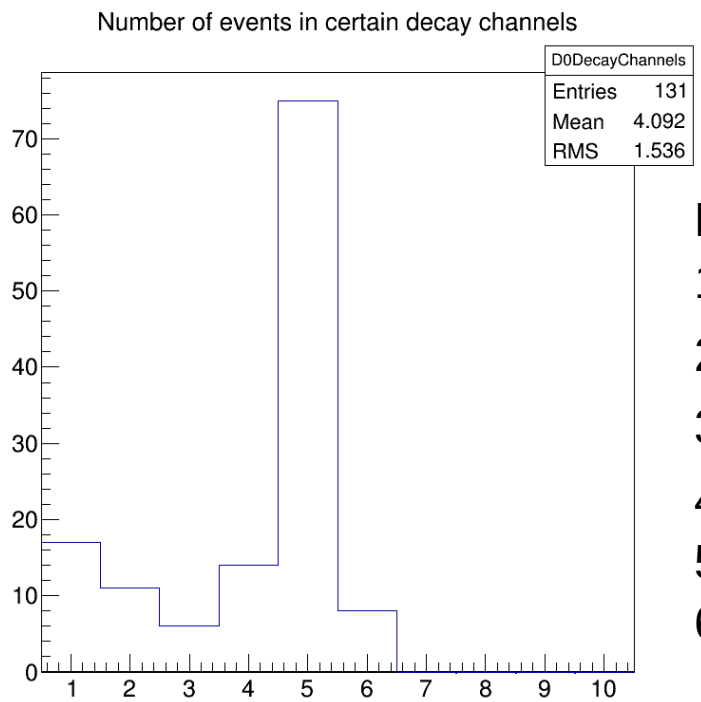




D0 properties - MCParticles only

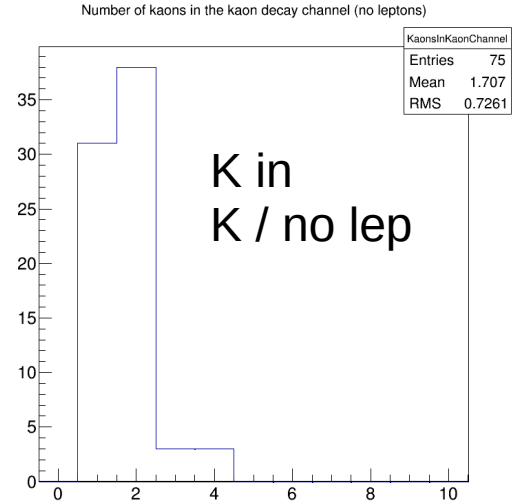
D0 events cuts:
1: D0 found in event
2: number of D0 found
3: decayed into >= 1 Kaon
4: decayed into >= 1 charged Kaon
5: more than 1 parent found



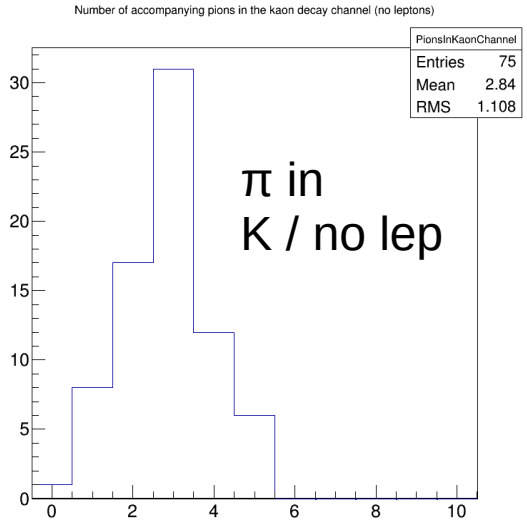


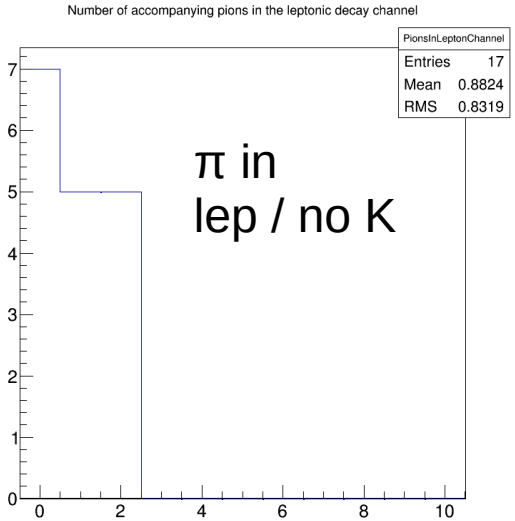
D0 decay channels:
1: leptonic ($e+\nu_e$ or $\mu+\nu_\mu$)
2: $e+\nu_e$
3: $\mu+\nu_\mu$
4: leptonic + Kaon
5: Kaon, no leptons
6: no Kaons, no leptons

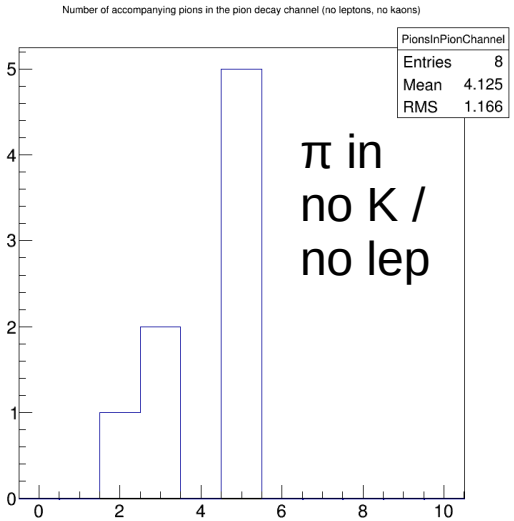
Secondary particles in D0 decay channels



- Significant multiplicities in Kaons and Pions







D0 path length

- Path = abs (endpoint – vertex) for MCParticles
- Mean: 1.84 mm