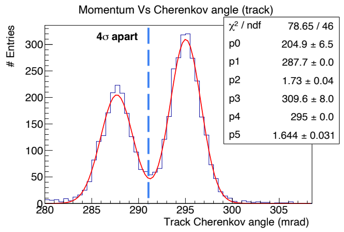
Machinery and basic assumptions

https://github.com/eic/irt/tree/pfrich/delphes

- Each mass hypothesis is associated with a 1D Gaussian peak in a measurement space
  - Time of Flight, track-level Cherenkov angle
- Smearing matrices produced separately for π/K/p and e/π cases
- Smearing between any two mass hypotheses is defined as an overlap in Gaussian tails
  - Eventually reduces to an erf() calculation (math needs to be verified)
- Photo-statistics ignored for Cherenkov detectors
  - Effectively this means MC truth hit-to-track association (a handful of hits is considered as a valid case)
  - Production threshold can be set by hand though, for each mass hypothesis separately



- Delphes-like .tcl (ETOF example) and a custom .root format output
- Assume 100% efficiency
  - Can likely produce the same tables with a number of fixed efficiency values
  - Applying variable efficiencies on the fly should be possible if .root output is used

Time of Flight in ATHENA

- 1D quantity: time of flight between a primary vertex and a detection point
- Calculable analytically under few basic assumptions / inputs:
  - Constant magnetic field
  - Location of the detection surface (plane in the endcaps, cylinder in the barrel)
  - Mass hypothesis
  - Momentum
  - Expected detector timing resolution
  - t_0 uncertainty
  - Momentum resolution, path length resolution

- Example script
- Output: a .tcl file in a Delphes format **-> Fast simulation (Delphes itself)**

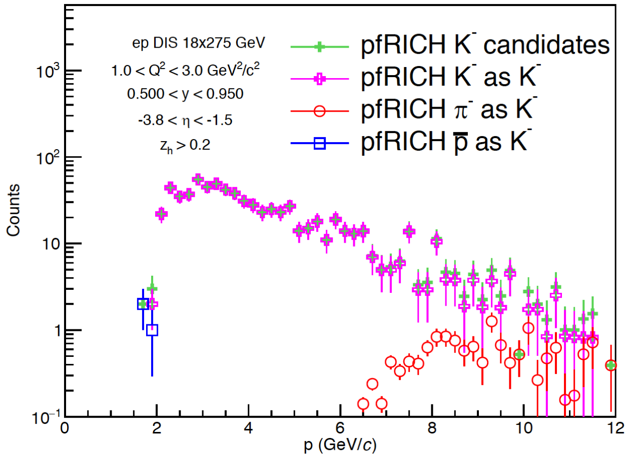
DIRC in ATHENA

- 1D quantity: a Cherenkov angle in quartz
- ROOT file with the resolutions as provides by DIRC group (example)
- The rest is calculable analytically under few basic assumptions / inputs:
  - Constant magnetic field (defines the momentum cutoff)
  - Installation radius (defines the momentum cutoff)
  - Mass hypothesis
  - Momentum
  - Tracker angular resolution @ the installation radius (parameterized using ATHENA GEANT simulation)

- Example script
- Output: a .tcl file in a Delphes format **-> Fast simulation (Delphes itself)**

ePIC pfRICH in March 2023 review

- 1D quantity: a Cherenkov angle in aerogel
- Track-level resolutions are taken from a gaussian fit to a standalone simulation output plots
- The rest is calculable analytically under few basic assumptions / inputs:
  - Average refractive index of the aerogel
  - Installation radius
  - Mass hypothesis
  - Momentum
  - Tracker angular resolution taken from ePIC full simulations



- Example script
- Output: a .root file (a C++ factory dump) **-> Imported in a PYTHIA MC data analysis**
- Example import script

See also slide 17-20 here (ePIC GDI meeting: pfRICH software validation)

Implementation options for ePIC

- Replicate something like this (or write something different) from scratch in EICrecon environment
  - Native fit to the EIC software, workforce, time, potentially yet another reinvention of the wheel

- Append existing code by a podio output format
  - A good match to EIC software, workforce, time

- Use Delphes .tcl output as is
  - Existing format (can be cooked by hand if needed), import code in EICrecon is needed

A hint: https://github.com/delphes/delphes/blob/master/examples/DelphesBrowser.C - L20

- Use .root output as is
  - Works "out of the box", "low coding standard", dependency on IRT repo, python / Go / C# users out of luck