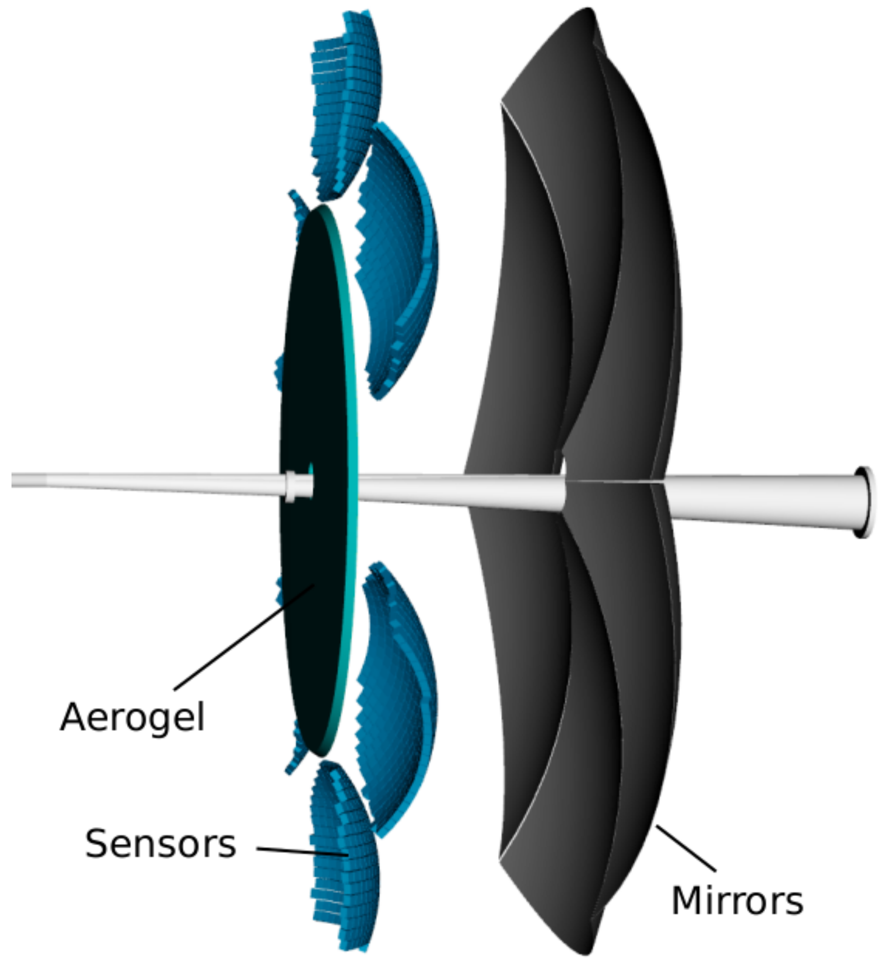
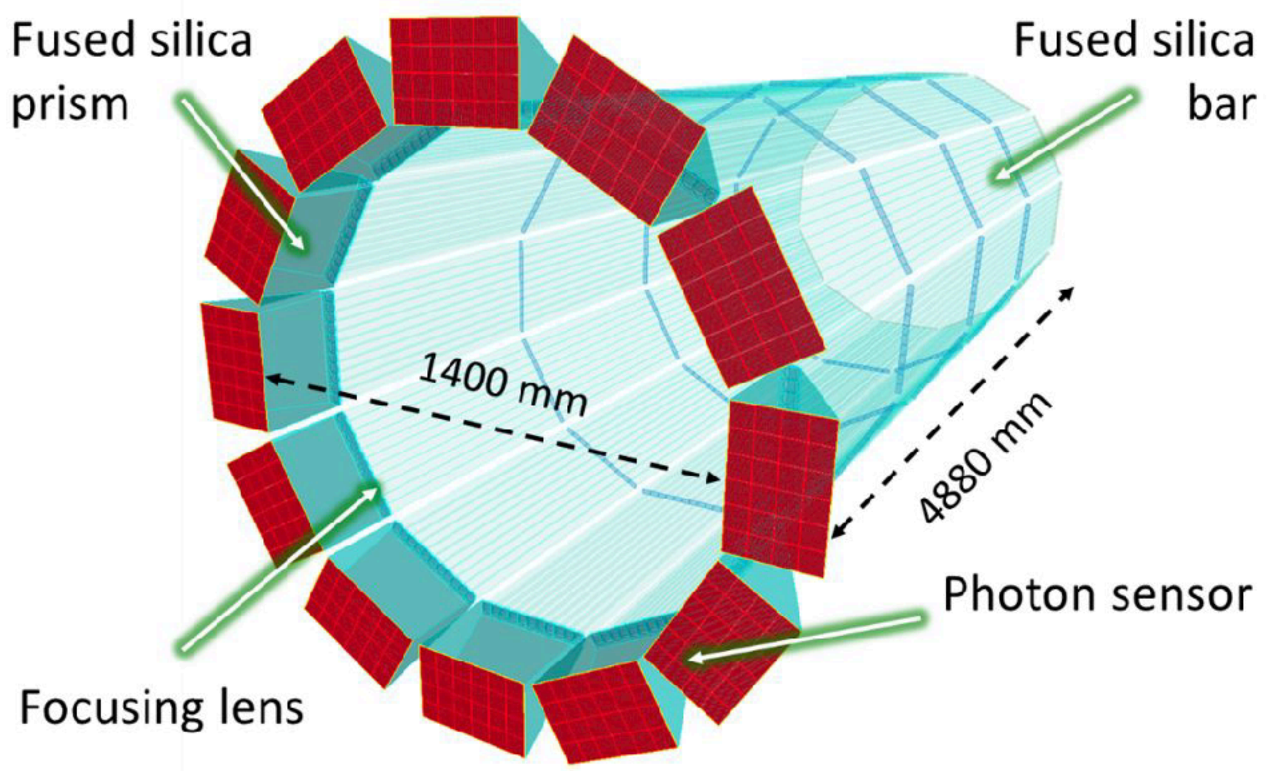
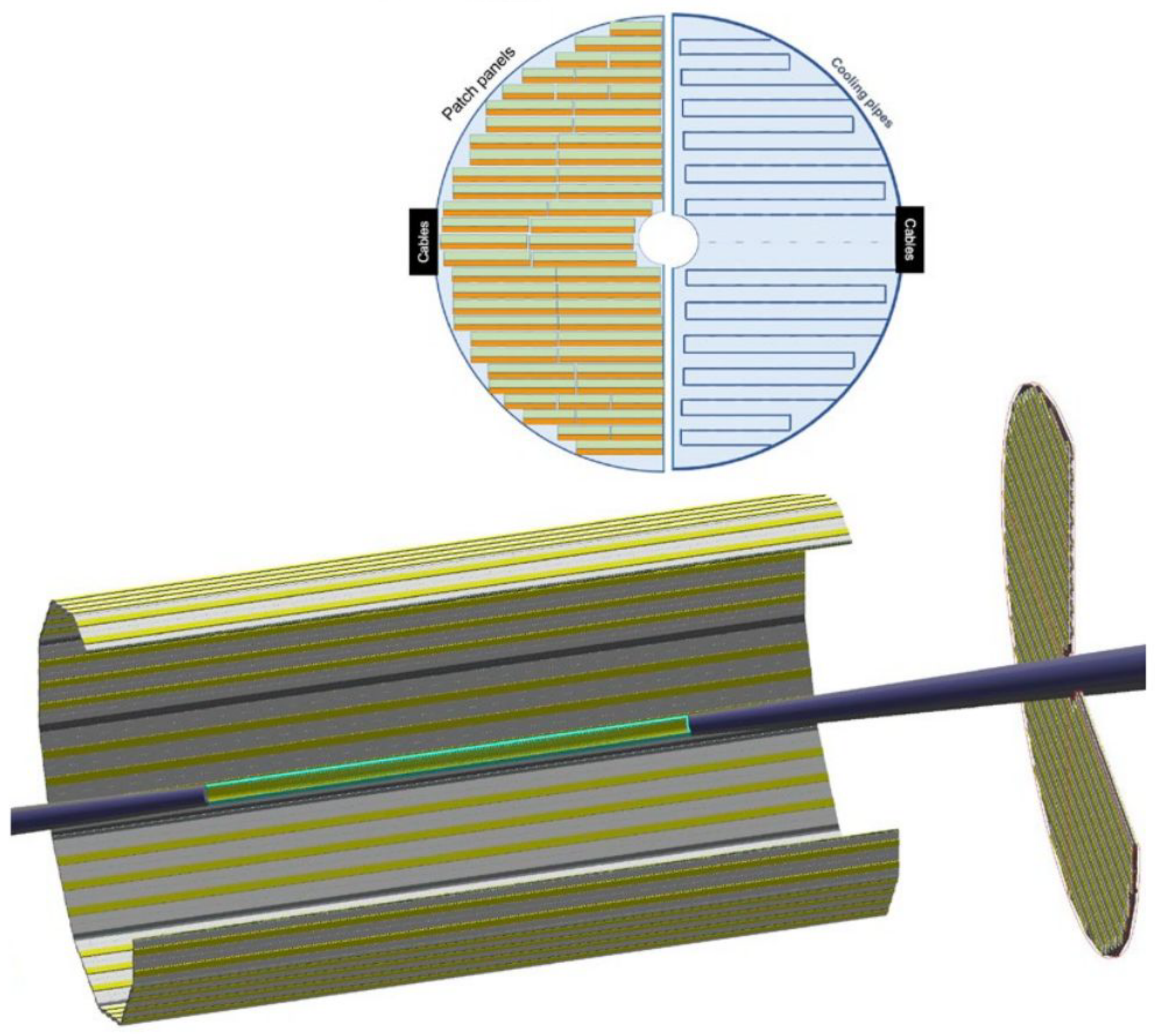
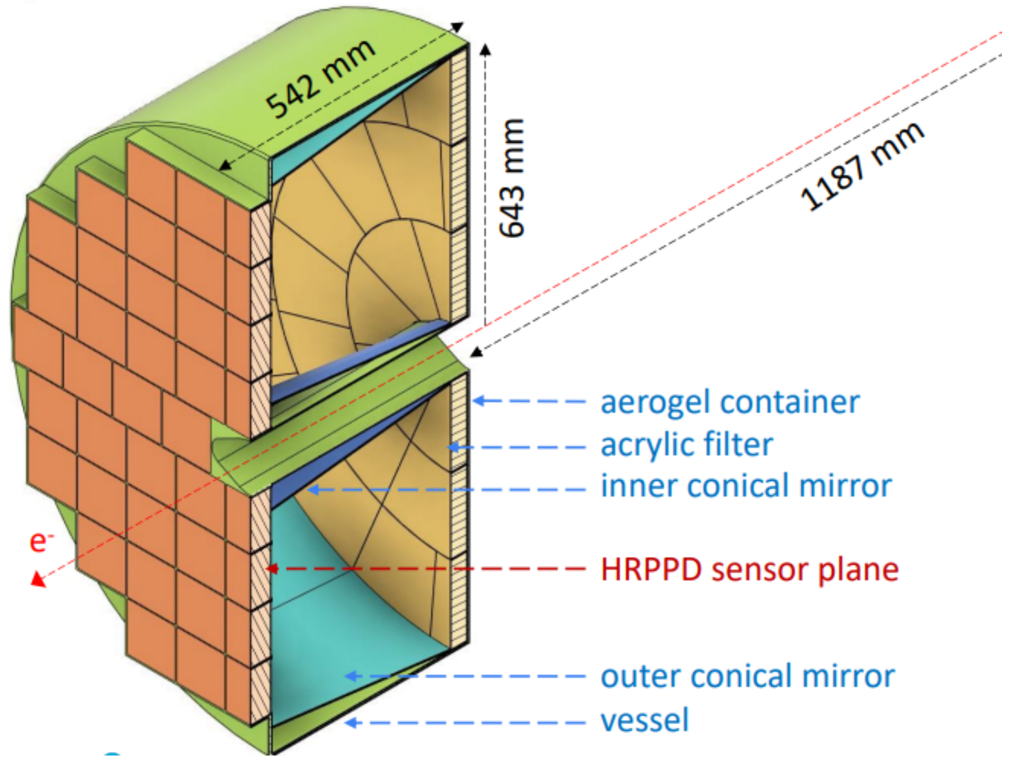
Status of PID Lookup Tables

Thomas Ullrich,
Analysis TDR Kick Off
February 5, 2024









BROOKHAVEN NATIONAL LABORATORY

Reconstruction of PID Detectors

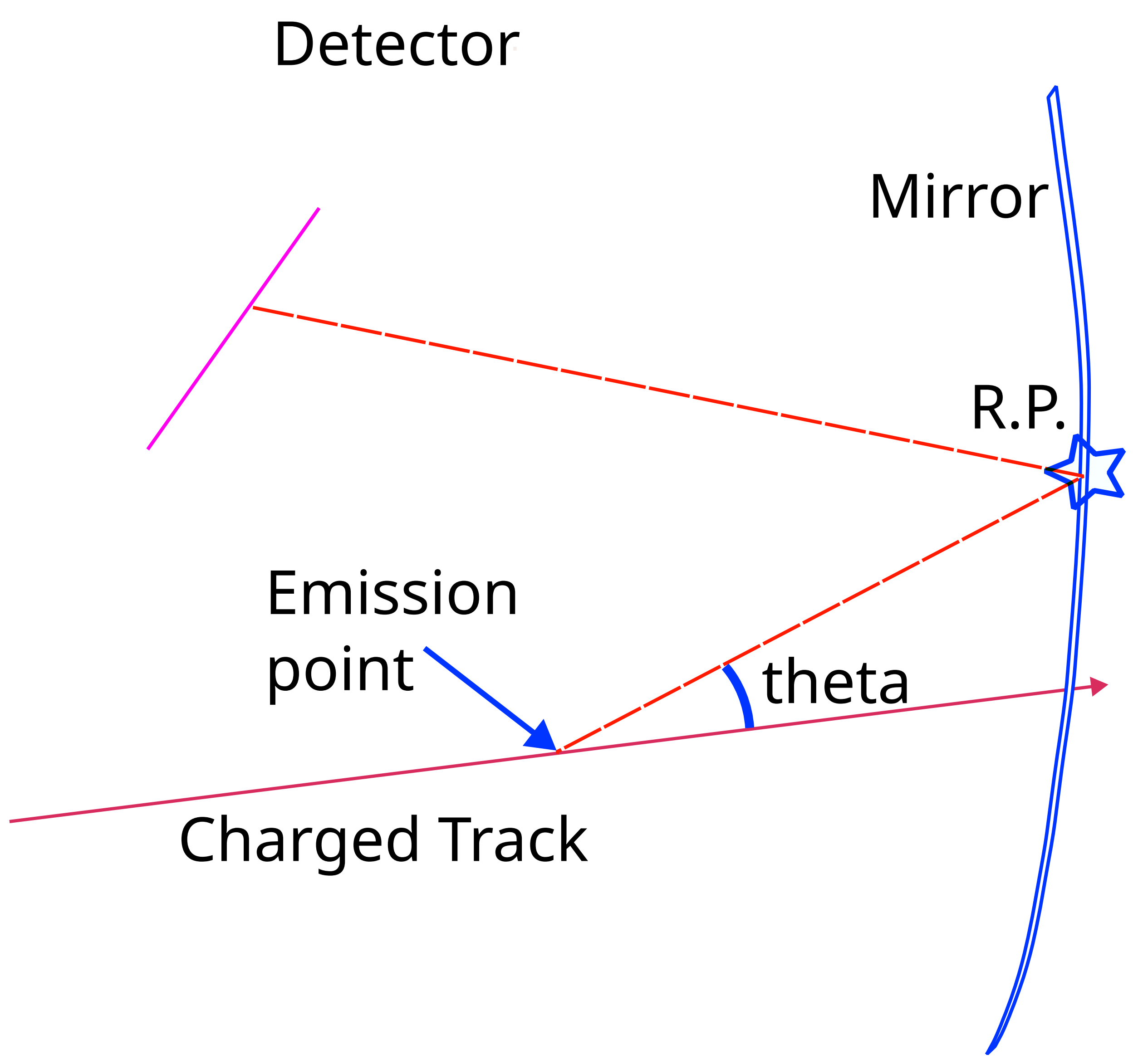
- pfRICH
  - stand alone
  - porting to DD4HEP and EICrecon in progress
- dRICH
  - stand alone
  - DD4HEP and EICrecon in place and being tested
- hpDIRC
  - stand alone
- ToF
  - mix with external reconstruction (stand alone)

Note: stand alone sims and reconstruction on high standard and reliable, many verified and tested in beam tests

Status: dRICH Reconstruction

Inverse Ray Tracing (IRT):

- currently with some simplifications such as parametric surfaces
- Provides reliable values for angular resolution
- Sufficient for single particle characterization w/o noise
- capable of 3σ pi/K separation slightly above than 50 GeV/c in the forward region
- Aerogel provides 3σ separation above the K threshold in gas providing substantial overlap.
- Current version IRT v1.0 incapable to perform complicated noise handling
- Priorities: Improve and fix the reconstruction limitations, and start looking into more complicated scenarios



- Used in several RICH detectors; e.g. HERMES, COMPASS.
- Iterative solution to estimate mirror impinging point.
- W.R.T a fixed star (beam direction, mirror centre), given knowledge of detection and emission point, Cherenkov angle can be measured.

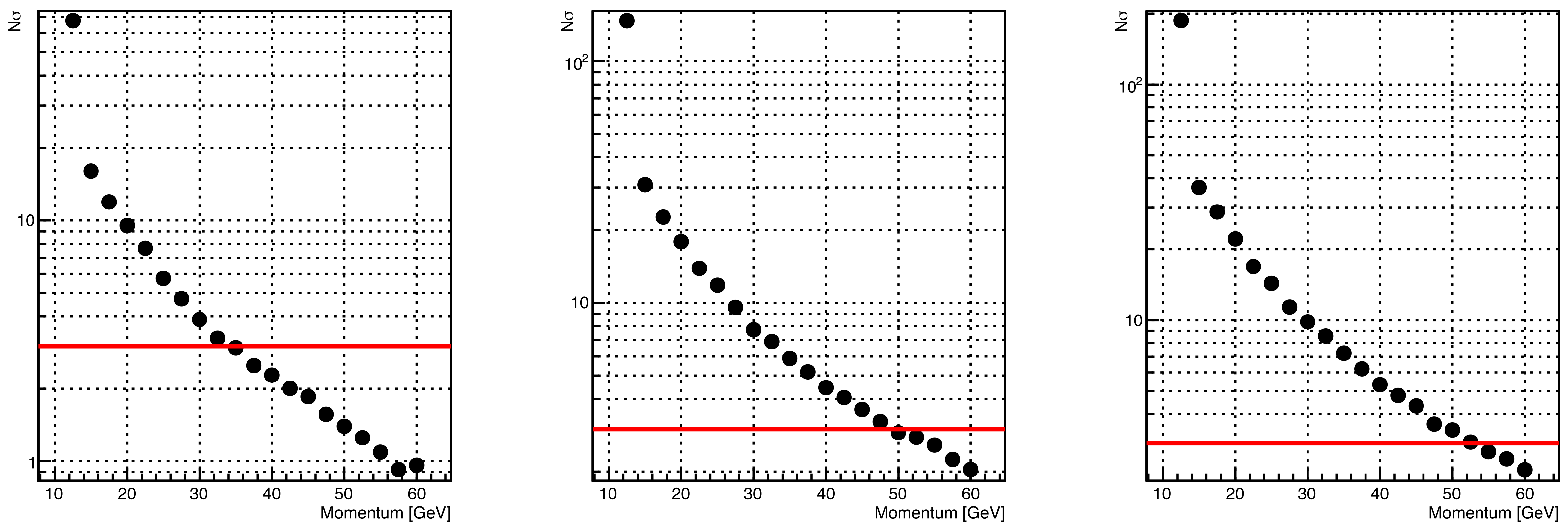


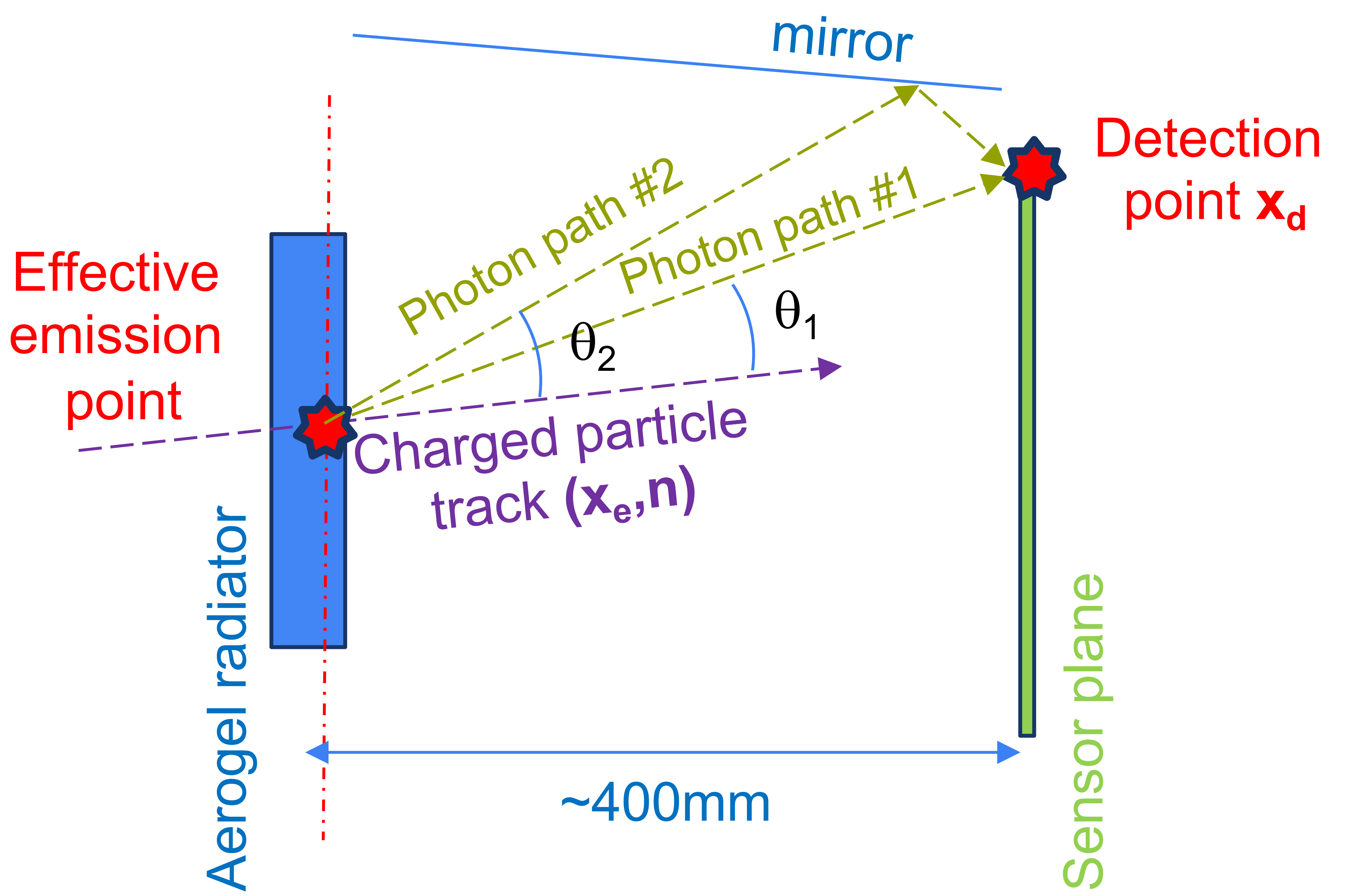
Figure: Seperation power(Aerogel to be redone); eta (1.3-2.0); (2.0-2.5);(2.5-3.5)

Full chain working in DD4Hep and EICRecon

Status: pfRICH Reconstruction

- Existing codes are algorithmic, χ^2 based
- Developed in a standalone GEANT4 environment
- Porting to dd4hep is in the geometry description stage
- Started with ideal case then add reality
  - Emission point uncertainty (aerogel thickness)
  - Detection point uncertainty (sensors have finite resolution)
  - Chromatic effects ($n(\lambda)$)
  - Refraction on optical media boundaries
- IRT Algorithm
- Noise and overlapping rings studied
- pfRICH + HRPPD (ToF δt ~ 50 ps)
  - Timing is used in both hit-to-track association for a given mass hypothesis, and in the χ^2 ansatz

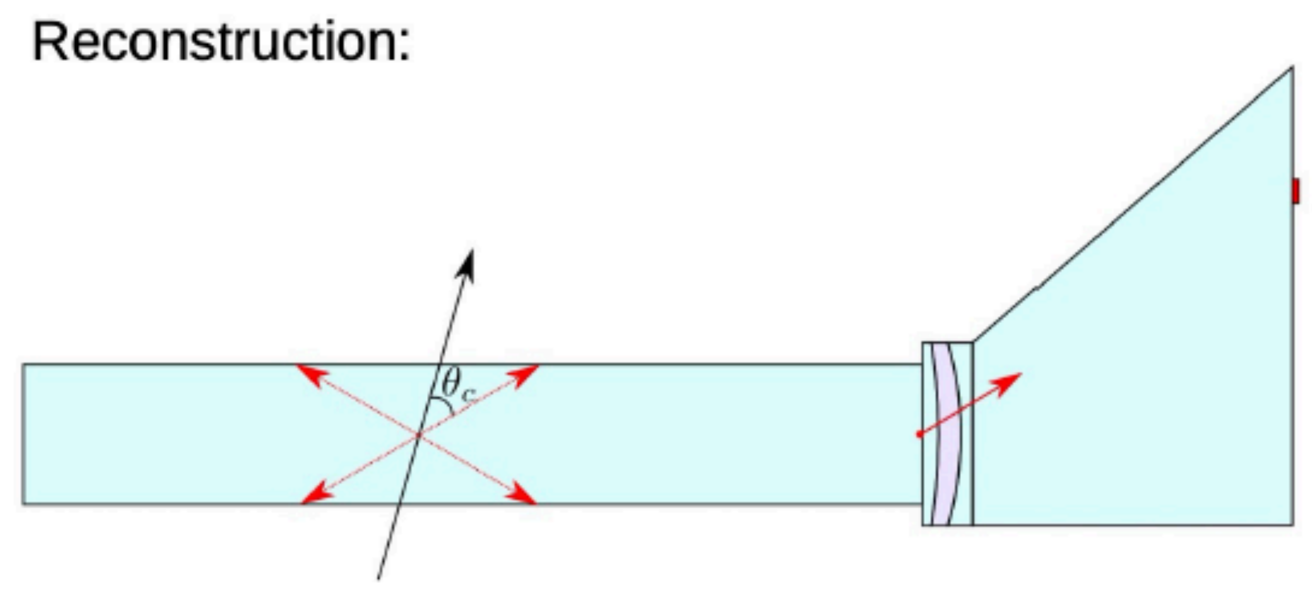
$$\chi^2_H = \frac{[\theta_H(p,n) - \theta_c]^2}{\sigma^2_\theta}$$ for a given PID hypothesis

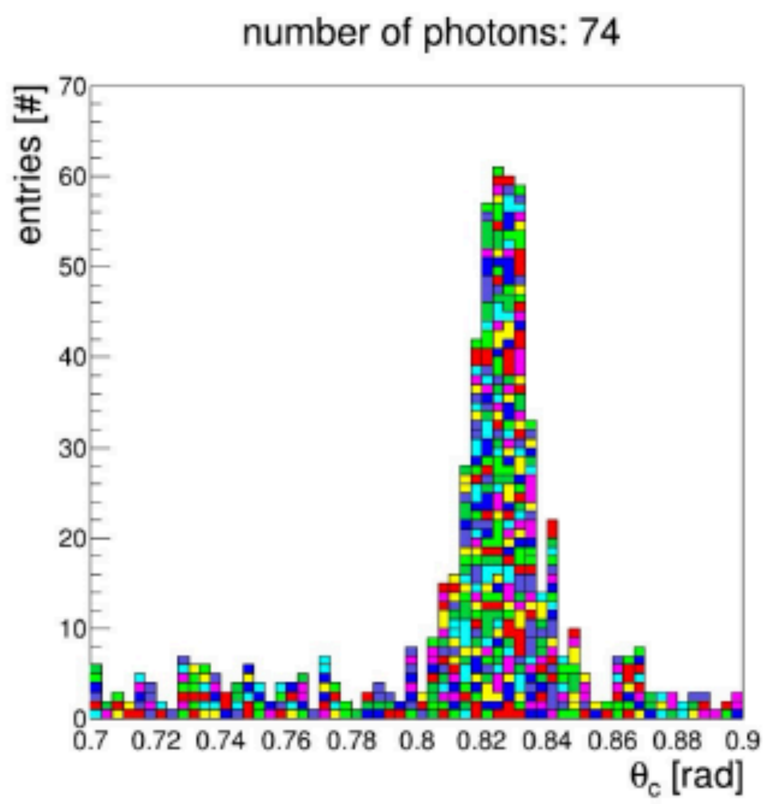


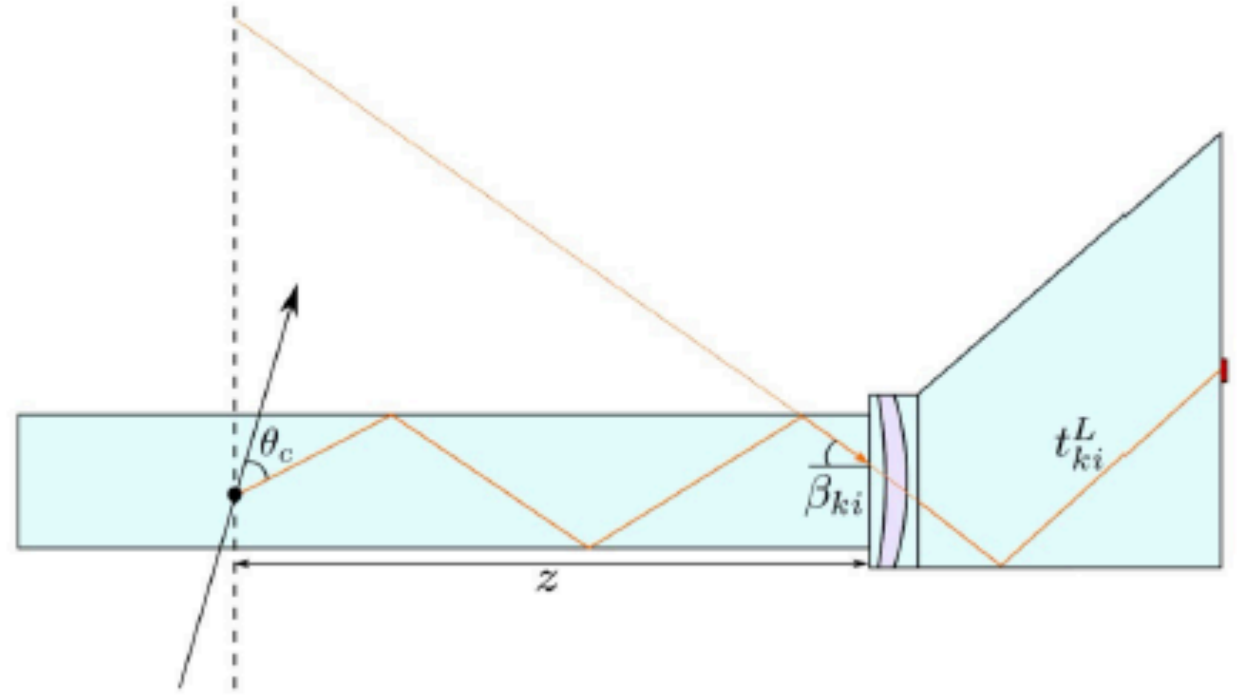
$$\chi^2_H = \sum_{k=1}^{nhits} \frac{[\theta_H(p,n) - \theta^k_c]^2}{\sigma^2_\theta} + \sum_{k=1}^{nhits} \frac{[t_H(p) - (t^k_c - t_0)]^2}{\sigma^2_t}$$

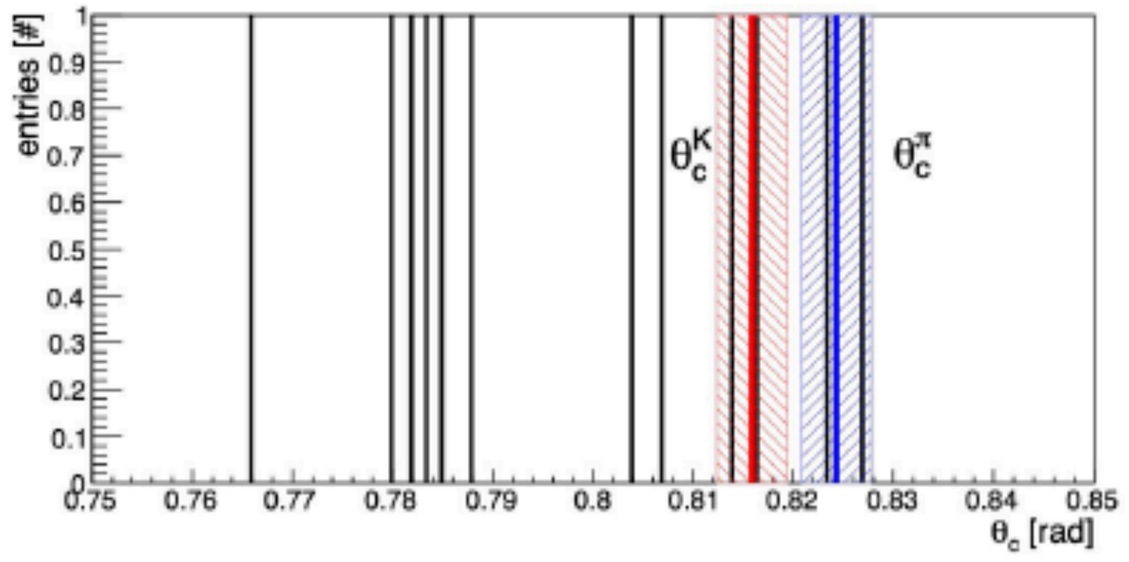
hpDIRC Reconstruction

- Reconstruction and PID methods:
  - Geometrical (BABAR-like), robust and fast method based on LUT, delivers Cherenkov angle per particle and Single Photon Resolution (useful for calibration and in prototype tests), does not depend on precise time measurement
    - Pixel position + bar location define photon direction at bar end, stored in LUT, combined with particle track to calculate Θ_C.
    - Path pixel – bar not unique, combinatorial background in Θ_C requires careful treatment.
    - Arrival time information is used to resolve ambiguities
  - Time Imaging (Belle II TOP-like), uses Probability Density Functions (analytical or simulation-based), makes optimum use of precision of position and time information
    - from data: best PID, requires a large amount of data in whole angular and momentum acceptance
    - simulated: full Geant4 simulation of every possible particle type direction and momentum
    - DIRCs appear to be remarkably robust detectors in presence of background (see https://indico.bnl.gov/event/20473/contributions/85266/attachments/51918/88782/20240111_hpDIRC_pidBG.pptx)





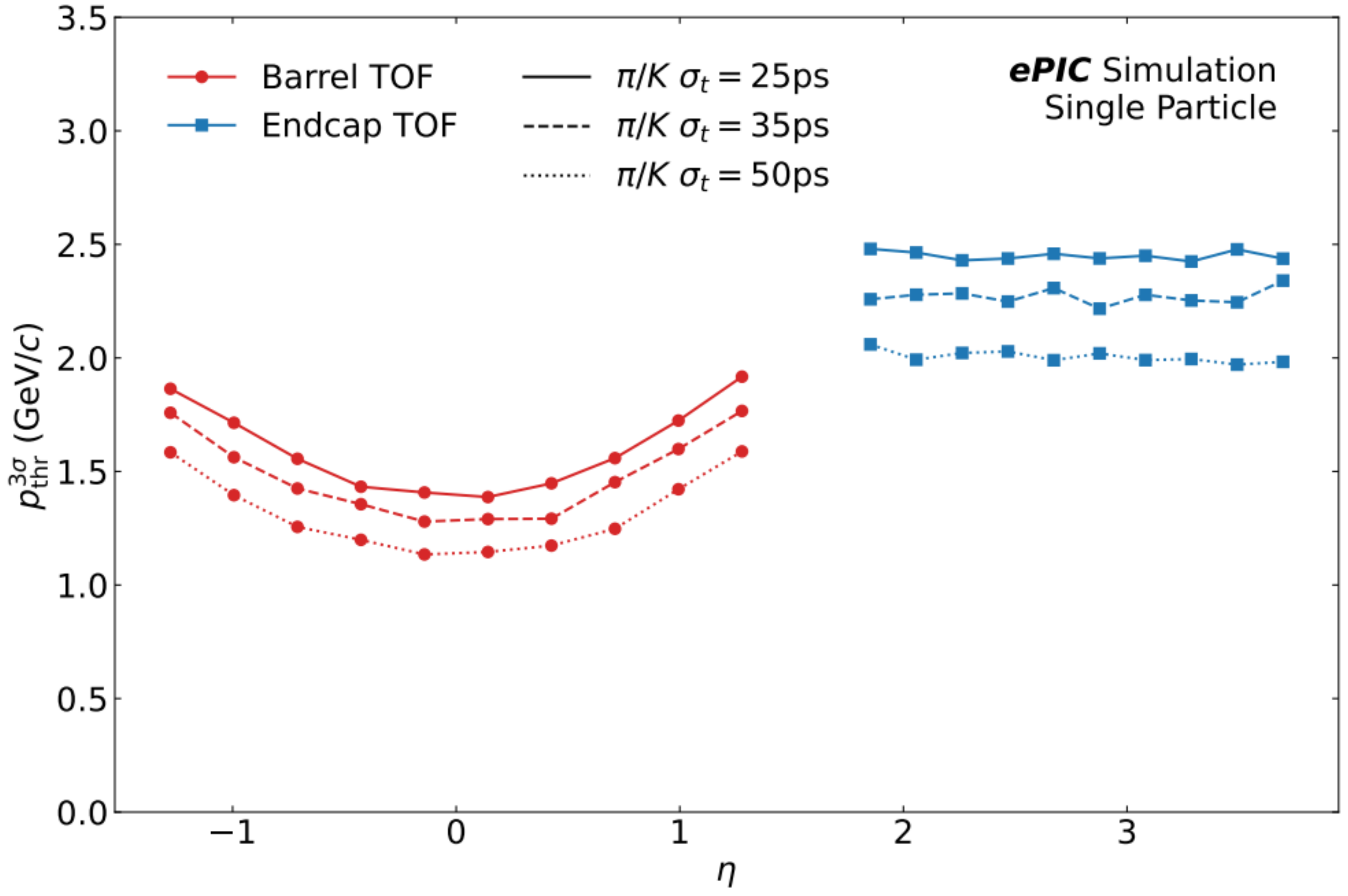




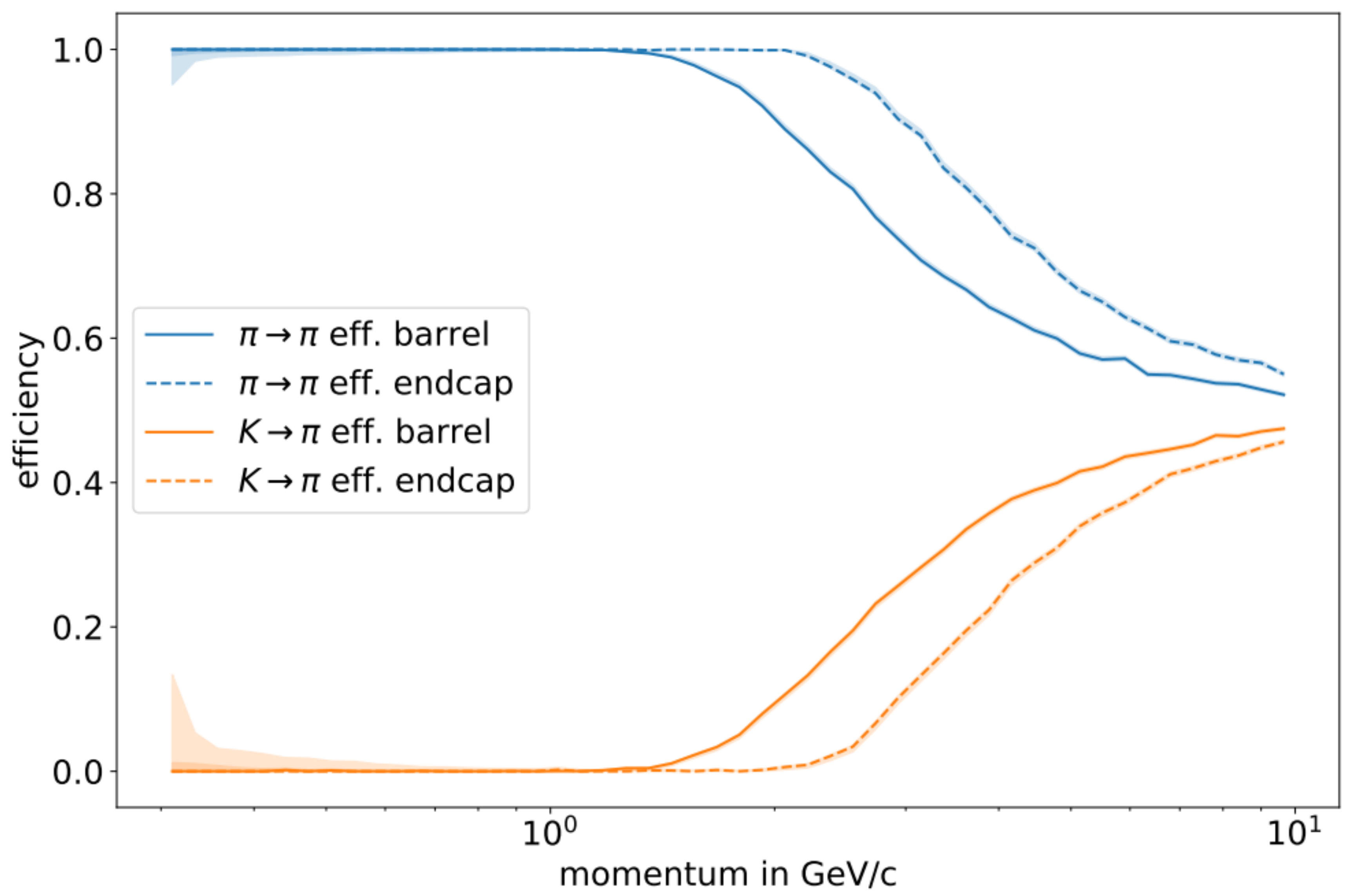
ToF Reconstruction

Oskar Hartbrich (ORNL)

- Overall simple
  - Need reconstructed momentum, track length and hit time
  - From momentum and track length calculate expected hit times of various particle species
  - Comparison of expected hit time to measured hit time yields weights/likelihoods for each hypothesis
- Devil in the Details:
  - No showstoppers but iterative improvements to be made
  - Need correctly modeled time distributions in simulation (digitization)
  - Need correctly modeled time distributions in reconstruction
  - Need correct assignment of TOF hits to track
- Currently using full eicsoft simulation software + custom plugin to write out relevant TOF hits, then do external reconstruction in python



TOF pion efficiency/mis-iD at p_π>0.5



TDR Preparation

- There is no fully worked out PID information in ePIC reconstruction software
- We need physics performance plot for TDR on a few month time scale including PID
- Consensus in PID groups is that validated, debugged, realistic implementation in EICrecon not happening on this time scale

- Note also that there is a certain skepticism about the quality of the track predictor into PID detector match (angular resolution). This is not as worked out as it could be
- PID groups are nervous that PID performance look worse than it actually is simply because of current shortcomings elsewhere.

Solution: LUT

- Only way out is to handle the sims externally and create lookup tables for the TDR analyses
- Tables to be used after redo as “after burner”.
- Devil is in the details
  - Misidentification causes cross-talk
  - Users could chose tighter PID cuts to obtain cleaner sample jeopardizing statistics
- February 2: meeting on LUT format
  - Rosi/Sal

PID lookup tables – ACs perspective

Rosi Reed (Lehigh) - Salvatore Fazio (Calabria)

- Topic Discussed with our PWG conveners at the Analysis coord. meeting on January 26th
  - Realistic PID simulation crucial for TDR-physics-chapter & related physics paper
  - We communicated the Challenging TDR timeline + expectation of having full PID reconstruction in our common software being unrealistic in the short term
  - PWG conveners agree on the **contingency plan of using lookup tables**
- Format of lookup tables has been discussed (purity vs efficiency vs …)
  - The most realistic format to mimic detector effects on PID would be a matrix showing the **probability for a particle to be reconstructed into something else:**
    - (*i*_generated_ID, *j*_reconstructed_ID, prob_*ij*) - describing probability of particle *i* to be reconstructed as particle *j*
- Timelines are important
  - Lookup tables would be ready by next software campaign?
  - Timeline for a full PID reconstruction? Especially relevant if final TDR realigns with a new CD3 timeline
- Dedicated discussion on Feb 5th - Analysis TDR Kick off Workshop: https://indico.bnl.gov/event/21775/
  - We **need a speaker from this community** to drive a detailed discussion

Solution: LUT

- Only way out is to handle the sims externally and create lookup tables for the TDR analyses
- Tables to be used after redo as “after burner”.
- Devil is in the details
  - Misidentification causes cross-talk
  - Users could chose tighter PID cuts to obtain cleaner sample jeopardizing statistics
- February 2: meeting on LUT format
  - Rosi/Sal
  - Markus

Hadron Identification

- **Representing PID information**:
  - EDM4hep data model has an improved way of representing PID assumptions:
    - There are *N* PID assumptions (which carry their own weight) that link to a `ReconstructedParticle`.
    - This information should be used in physics analyses.

This is interface between reconstruction and physics analysis.

  - EDM4hep data model and EICrecon support algorithmic flow where we combine responses from various PID systems:
    - As a starting point, we should aim for a reconstruction for each PID system.
- We should ideally populate the PID information through the use of reconstruction algorithms.
- As an alternative, we can use in EICrecon parametrizations based on standalone reconstruction or estimates.
- One possible approach is the unfolding method, as utilized by HERMES:

However, there are other methods, for example, the use of ML.

  - Parametrize the relationship between the identified hadron types to the true hadron types:

$$\mathscr{P} = \begin{pmatrix} P_\pi(\pi) & P_\pi(K) & P_\pi(p) \\ P_K(\pi) & P_K(K) & P_K(p) \\ P_p(\pi) & P_p(K) & P_p(p) \\ P_X(\pi) & P_X(K) & P_X(p) \end{pmatrix}$$

The $\mathcal{P}$-matrix depends on the momentum and the event topology of the hadron tracks.

The elements of the inverse $\mathcal{P}$-matrix can be interpreted as event weights which relate the identified hadron types to the true hadron types.

  - Pion weights, kaon weights and proton weights are assigned to each identified hadron track according to the inverse $\mathcal{P}$-matrix .
  - **Required steps**:
    - Use a parametrization for reconstruction of the identified hadron track. As an initial step, we can use use true hadron types.
    - Use a parametrization for the inverse $\mathcal{P}$-matrices. As an initial step, we can use use only 1 and 0 as event weights.
    - The parametrizations could be determined either in standalone reconstruction code or based on estimates.

In a nutshell: Populate the PID information using parametrizations based on standalone reconstruction or estimates in EICrecon.

Solution: LUT

- Only way out is to handle the sims externally and create lookup tables for the TDR analyses
- Tables to be used after redo as “after burner”.
- Devil is in the details
  - Misidentification causes cross-talk
  - Users could chose tighter PID cuts to obtain cleaner sample jeopardizing statistics
- February 2: meeting on LUT format
  - Rosi/Sal
  - Markus
  - Alexander

More distant future

- IRT 2.0 debugging will mostly proceed in a standalone code for the next 3-4 months
- This standalone code will be used to produce detector performance plots for pre-TDR
- Delphes-like tables (smearing matrices) can be produced a la ATHENA
  - These codes exist for dRICH, pfRICH, hpDIRC and part of TOF (funny enough, all this stuff still in IRT repository) …
  - … though effort to resurrect them and adjust to ePIC detector needs may be non-negligible
- Porting to EICrecon environment will continue as a low priority task
- Timelines will strongly depend on how quickly BNL NPPS can hire a person to work on this

```
void delphes_dirc( void )
{
  //printf("%f\n", (1.0 - erf(1.75/(sqrt(2.)*1.0)))/2);

  auto dirc = new DelphesConfigDIRC("barrelDIRC");
  dirc->UsePtMode();

  // Define particle mass hypotheses in ascending mass order; yes, there is no
  // reason to overcomplicate things;
#ifdef _E_PI_SEPARATION_MODE_
  dirc->AddMassHypothesis(-11);
#endif
  dirc->AddMassHypothesis("pi+");
#ifndef _E_PI_SEPARATION_MODE_
  dirc->AddMassHypothesis("K+");
  dirc->AddMassHypothesis("proton");
#endif

  // "d(theta) ~ a/p + b" parameterization as taken from Wenqing's 11/23/21 slides,
  // roughly averaged over eta; 12/05/2022: still use the same, for the time being;
  dirc->SetTrackerAngularResolution(0.9, 0.1);

  // eta and momentum range and binning;
  dirc->SetEtaRange      (-1.55, 1.79,  10);
  // Do not mind to use Pt rather than 1/Pt bins; [GeV/c];
#ifdef _E_PI_SEPARATION_MODE_
  dirc->SetMomentumRange( 0.44, 3.00,  10);
  //dirc->SetMomentumRange( 1.20, 1.21,   1);
#else
  dirc->SetMomentumRange( 0.44,10.00,  10);
#endif

  // Installation radius in [mm]; constant magnetic field in [T];
  dirc->SetInstallationRadius  (729.6);
  dirc->SetMagneticField       ( 1.700);

  dirc->SetParameterizationMap("./ctr_map_p1_0.95.root");

  dirc->DoSigmaCalculations();
  // This is again some generic stuff;
  dirc->WriteTcl(false);
  {
    auto fout = new TFile("barrelDIRC.root", "RECREATE");
    dirc->Write();
    fout->Close();
  }
  exit(0);
} // delphes_dirc()
```

Planning

- PID DSCLs will make decision on best approach this week
- LUT available by end of month for testing