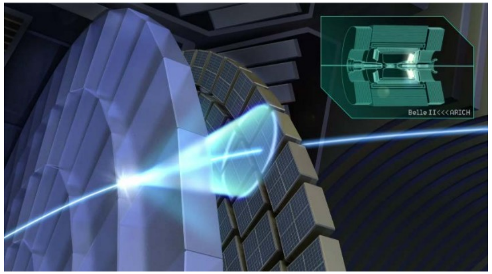
Institut "Jožef Stefan" Ljubljana, Slovenija

University of Ljubljana
Faculty of Mathematics and Physics



Recent advances and trends in pattern recognition and data analysis for RICH detectors

Luka Šantelj,

Jozef Stefan Institute and University of Ljubljana

12th International Workshop on Ring Imaging Cherenkov Detectors

Mainz, 15-19 September 2025

Introduction

→ good PID is essential part in many experiments in particle, astro-particle and nuclear physics

(imagine your favourite plot w/, w/o PID here)

→ good PID can only be achieved by excellent detectors (idea+hardware implementation) plus excellent data reconstruction algorithms

- improving reconstruction algorithms is often easier and cheaper than improving hardware
- it can be done continuously throughout the lifetime of experiment and even after
- in order to exploit the full potential of our experimental apparatuses (have more precise physics results on less data) it is our "duty" to develop efficient data reconstruction algorithms

Outline

- Traditional approaches to RICH reconstruction

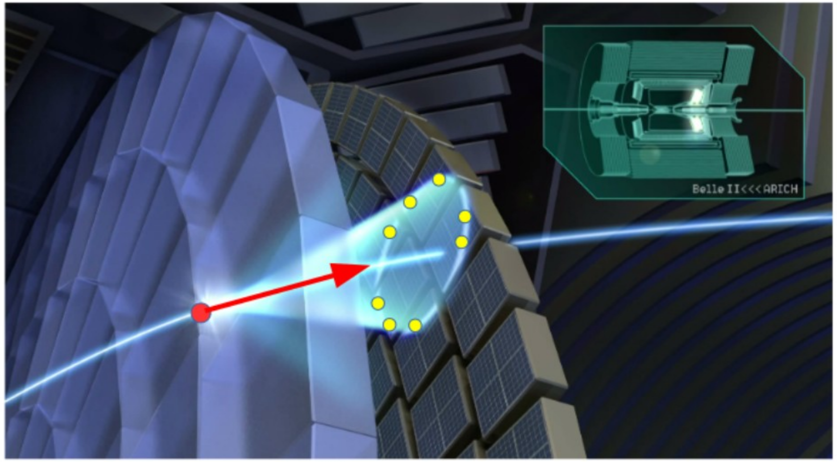
  → likelihoods
  → Hough transform

- Use of Machine Learning methods for PID and RICH reconstruction

  → global PIDs
  → ring image reconstruction
  → use of generative AI

- Conclusions

Likelihood approach



→ commonly employed when track information (position, direction, momentum) in the radiator is available (from other detector sub-systems)

→ we can base PID on comparison of observed pattern of detected photons with the expected one assuming given track parameters and particle type.

→ evaluate likelihoods for given particle type hypothesis (e.g. e, μ, π, K, p)

Likelihood for particle type h

$$\mathcal{L}^h = \prod_i^{pixels} p_i^h \quad \text{with} \quad p_i^h = e^{-n_i^h} (\boxed{n_i^h})^{m_i} / \boxed{m_i}!$$

name of the game is evaluation of n_i^h !

probability of observing the observed number of photons on pixel-i

expected number of photons on pixel-i

observed number of photons on pixel-i

Nucl. Instrum. Meth. A, 238:341, 1985

Likelihood approach

→ in case of “binary” photon detection, i.e. pixel is fired for $m_i \geq 1$

$$p_i^h = e^{-n_i^h} \quad \text{for non-fired pixels}$$

$$p_i^h = 1 - e^{-n_i^h} \quad \text{for fired pixels}$$

→ evaluating n_i^h for all pixels in the detector might be costly. In low occupancy environment, rewriting the likelihood in the following format can be useful

$$\mathcal{L}^h = \prod_i^{pixels} p_i^h \quad \Rightarrow \quad \ln \mathcal{L}^h = -N^h + \sum_i^{hit} [n_i^h + \ln(1 - e^{-n_i^h})]$$

total number of fired pixels (hits) expected to be observed

$$\sum_i^{pixels} n_i^h = N^h$$

→ evaluate n_i^h only for fired pixels, but need a method to estimate N^h

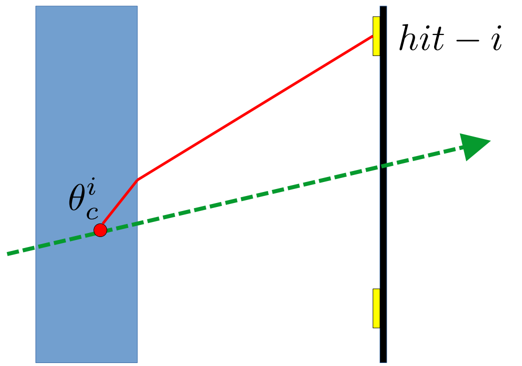
Likelihood approach

Evaluating n_i^h

→ can be done at different levels of sophistication, depending on performance needs, factors limiting the PID performance, computing resources etc.

→ based on detector geometry, track information and hit position, Cherenkov angle $(\theta_c,\phi_c)^i$ for each hit can be reconstructed, assuming photon emission point in the radiator

→ it is a reverse ray-tracing problem, depending on complexity of the geometry and required precision can be solved analytically, semi-analytically (numerically solving analytic relations), or with the use of iterative algorithms



→ finally $n_i^h = \underbrace{n_i^{h,s}(\theta_c^i, \phi_c^i)}_{\text{signal (non-scattered)}} + \underbrace{n_i^{h,b}(\theta_c^i, \phi_c^i)}_{\text{background (scattered, other particles, noise...)}}$

$$n_i^{h,s/b} = \boxed{N^{h,s/b}} \times \boxed{\mathcal{P}^{h,s/b}(\theta_c^i, \phi_c^i)} \times \boxed{d\Omega_i}$$

expected number of "emitted" photons

Cherenkov angle PDF (e.g. Gaussian peak at θ_c^h)

solid angle covered by pixel-i

Likelihood approach

Evaluating N^h

→ can be obtained from expected total number of signal photons to be emitted in the radiator for hypothesis h (scatter and PDE corrected) and average geometrical acceptance for Cherenkov ring.

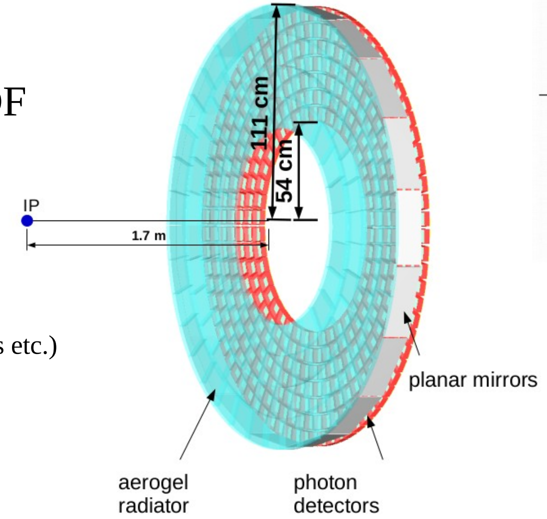
→ or better, ray-tracing "toy simulation" is used to obtain N^h on track-by-track basis.

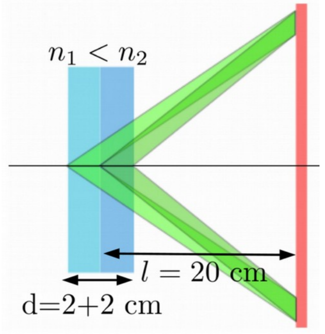
Use case – ARICH @ Belle II

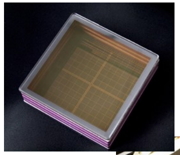
→ instead of evaluating n_i^h in the Cherenkov angle space the PDF is projected from it onto the photo-detector plane

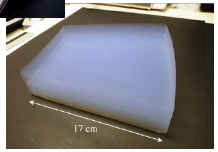
→ this allows for easier inclusion of effects that cannot be easily described in Cherenkov space

(geometrical acceptance effects, inefficiencies of detectors, internal reflections in photon detectors etc.)









Likelihood approach

→ implementation of PDF projection to photo-detector plane

Expected Cherenkov angle!

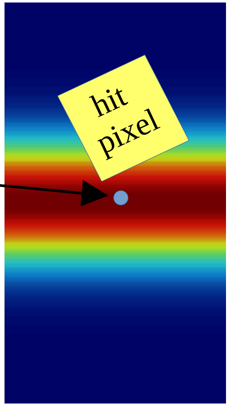
- first (θ_c^i, ϕ_c^i) are reconstructed, followed by propagation of "toy" photon with $(\boldsymbol{\theta_c^h}, \boldsymbol{\phi_c^i})$ from the assumed emission point in the aerogel to the photo-detector plane

- around the "toy" photon impact point assume Gaussian profile in the radial direction of the expected ring and flat in the azimuthal direction in the x-y space of detector plane

- width of the Gaussian profile is calculated on the track-by-track basis, based on the track path length in aerogel, distance between emission point and photon hit and track parameters uncert.

- 2D integral of Gaussian profile over the pixel surface is performed to obtain $n_i^{h,s}$
  (as pixel size is not small compared to Gaussian width)

toy photon
@ (θ_c^h, ϕ_c^i)



- nominally background is added in the form $n_i^{h,b} = const. + \mathcal{P}(\theta_c^i; N^h, \text{winHit})$

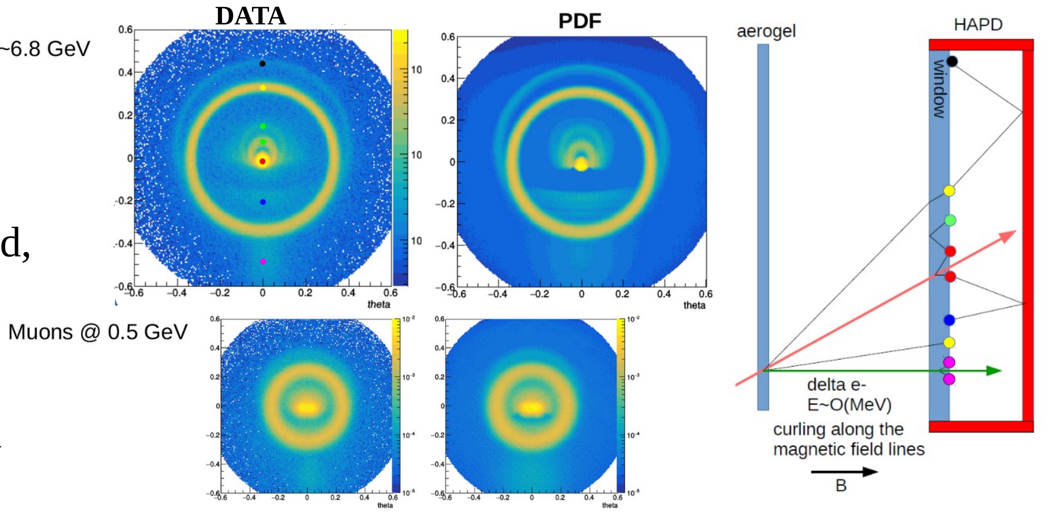
account for scattered signal photons

particle hit in photon-detector window

Likelihood approach

→ in the past few years effort was made to include several additional ring image features to PDF
(cherenkov photons from window, photon reflections…)

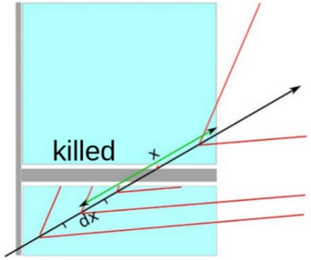
→ only marginal improvements in performance observed, so as default we maintain using original PDF



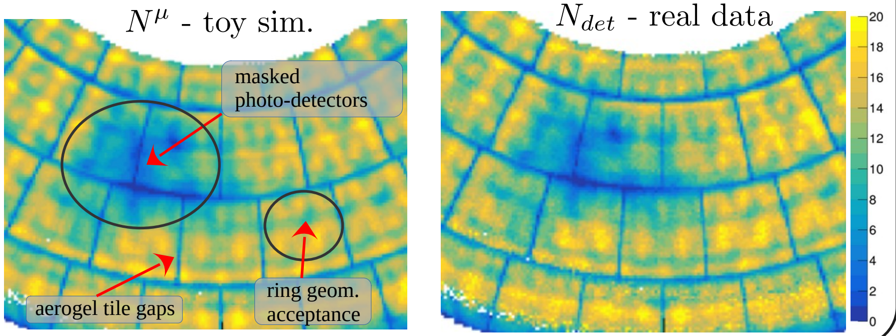
→ total number of expected photons N^h is obtained by "toy simulation" of photon emission and propagation

- 20 times expected emitted photons are propagated at θ_c^h from 10 points along the track path in aerogel to the photo-detector plane
(includes: photon loss in gaps between aerogel tiles, pixel-by-pixel Q.E., run based pixel masking)

Nucl. Instrum. Meth. A, 876:104–107, 2017
Nucl. Instrum. Meth. A, 952:161800, 2020



Expected / detected number of photons vs. track position
(in e+e−→ $\mu^+\mu^-$)



Likelihood approach

Global likelihoods

Nucl. Instrum. Meth. A, 433:257–261, 1999

→ in cases of non-negligible probability for Cherenkov rings from multiple particles to overlap improved PID performance can be achieved by the use of global likelihood

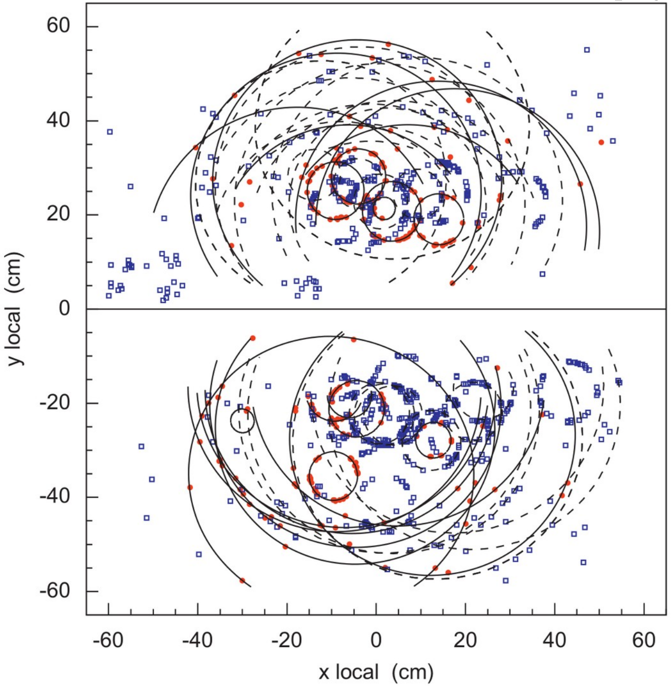
→ here likelihood is not evaluated on track-by-track basis, but for collection of all tracks in an event

→ using similar methods as described $n_i^{\{h\}}$ is calculated summing up contributions from all tracks given their assumed id. hypothesis (here {h} denotes a set of hypotheses for all tracks in an event)

→ set of identities {h} that maximizes $\mathcal{L}^{\{h\}}$ gives the most likely hypothesis for each particle in event

→ in cases where other backgrounds dominate the ring image (not cross-feed from neighboring tracks; e.g. detector noise, photons from secondaries) usage of global likelihood might become unpractical (→ local likelihood with effective background description)

LHCb RICH 1 – simulated event display



Likelihood approach

Use case – LHCb

→ evaluating $\mathcal{L}^{\{h\}}$ for all possible combinations of particle hypothesis in non feasible (~50 tracks | 5 hypotheses)

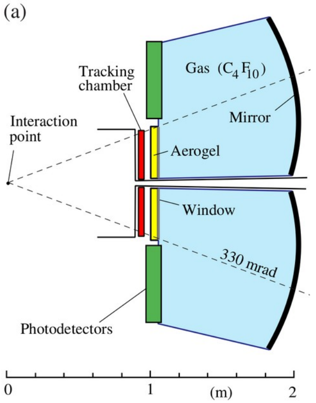
→ start with assuming all particles are pions (most abundant), track-by-track change mass hypothesis and fix it to most likely (max $\mathcal{L}$), iterate until no increase in $\mathcal{L}$ is found.

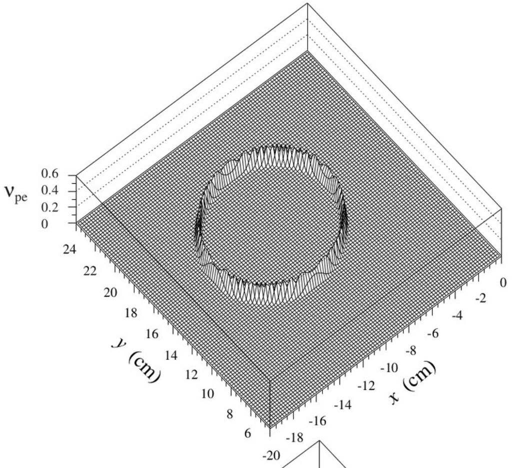
→ typically $2(N_{\mathrm{track}})^2$ likelihood evaluations are needed to obtain max. $\mathcal{L}$, which is manageable.

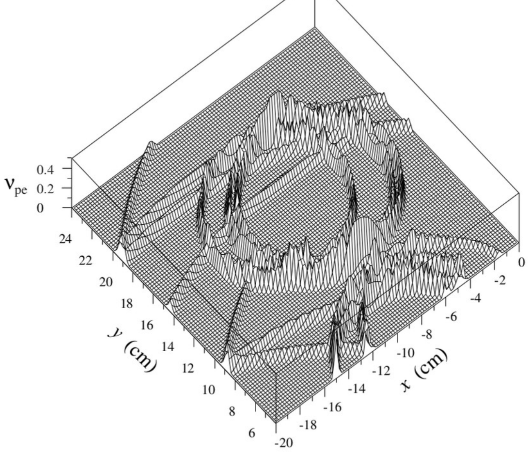
→ particle ID estimators for each track are then Obtained as

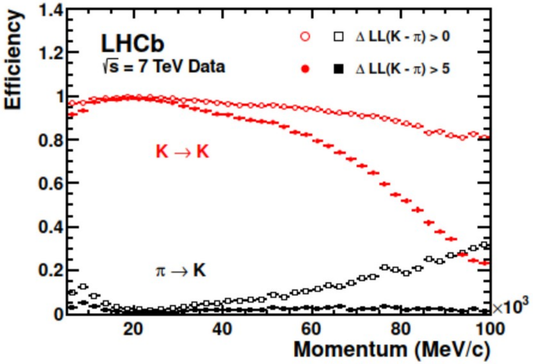
e.g. for i-th track, K/pi discrimination:

$$\ln \mathcal{L}^{\{h\}^{max} \Leftarrow K_i} - \ln \mathcal{L}^{\{h\}^{max} \Leftarrow \pi_i}$$









Nucl. Instrum. Meth. A, 433:257–261, 1999

Hough transform

→ well established technique for pattern recognition: allows to reconstruct family of given shapes (e.g. circles, ellipses) from discrete data points.

→ can be used to reconstruct rings without external track information, largely unaffected by topological gaps and high backgrounds

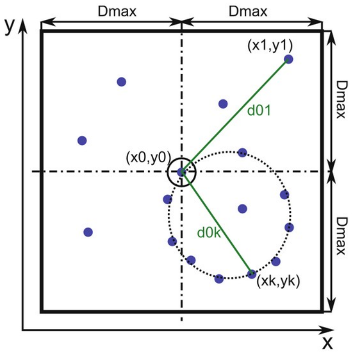
→ basic idea (circle search): each hit (x,y) in 2D plane is mapped to a 3D cone surface in the circle parameter space H(x_0, y_0, r), since $f(x,y,x_0,y_0,r) = (x-x_0)^2 + (y-y_0)^2 - r^2 = 0$

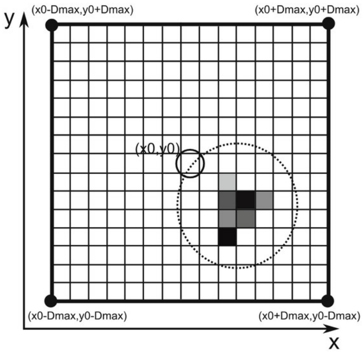
→ intersection point of N cones gives circle (x_0, y_0, r) containing the N associated hits on the detector (finding rings → finding intersection of hyper-surfaces in parameter space)

→ in practice one can discretize the parameter space and for each (x,y) increment content of cells for which $|f(x,y,x_0,y_0,r)| < T$

→ or alternatively histogram circle candidate parameters of all possible hit triplets

→ cells with content above some threshold considered as rings





Hough transform

Use case – ALICE HMPID J. Phys. G: Nucl. Part. Phys. 32 (2006) 1295

→ track information is available but very high background environment (O(10%) occ.)

→ reconstruct Cherenkov angle for each hit based on track information
(parameter space reduces to 1D – only Cherenkov angle)

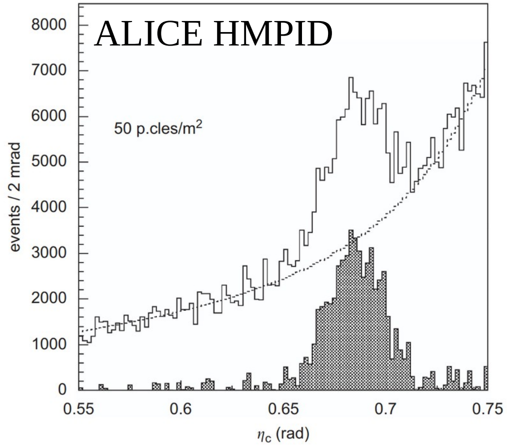
→ histogram entries are weighted to effectively subtract background

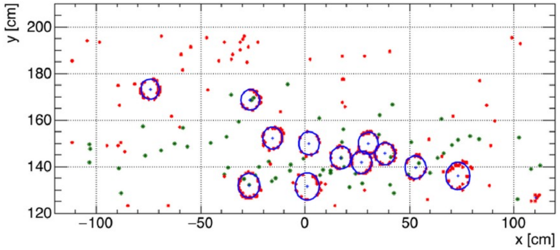
→ finally "sliding-window" is used to determine mean angle

Other use cases

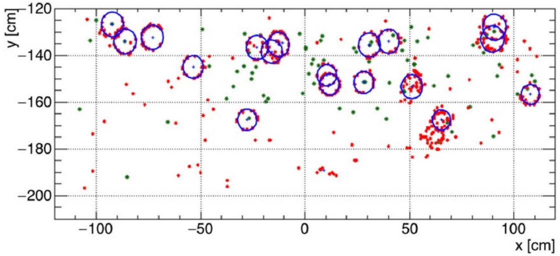
→ variety of Hough transform based algorithms successfully employed at different experiments (mostly for track-less reconstruction)

→ often analysis is done in multiple steps where Hough transform is used for ring search, and rings are then re-fitted using other methods
(e.g. CBM, water Cherenkov detectors)





CBM experiment @ FAIR - simulation



What can Machine Learning methods do for us?

- global PID @ experiments
- pattern recognition / ring reconstruction
- fast simulation and reconstruction (generative AI)
- detector design optimization
  (e.g. E. Cisbani et al 2020 JINST 15 P05009)
- calibration methods
  (e.g. C. Fanelli 2020 JINST 15 C02012)

Global particle identification @ experiments

→ traditionally PID is performed using global PID likelihood which is obtained as $\log \mathcal{L}_h = \sum_d \log \mathcal{L}_{h,d}$

sum over sub-detectors

→ **in idealistic case this is end of story**: assuming same particle crosses all sub-detectors and each sub-detector likelihood correctly describes probabilistic processes involved in the generation of detector response upon passage of a particle of a given type

→ in reality: - sub-detector likelihoods are always imperfect/incomplete

- non-trivial correlations between them might exist (e.g. via track information in Cherenkov likelihoods and dE/dx)

→ therefore whole can be more than a sum of "broken" parts!

→ combining sub-detector likelihoods with ML methods has been employed at several experiments in recent years and proved to bring notable performance gains

→ adding additional input observables other than sub-system likelihood usually further enhances performance

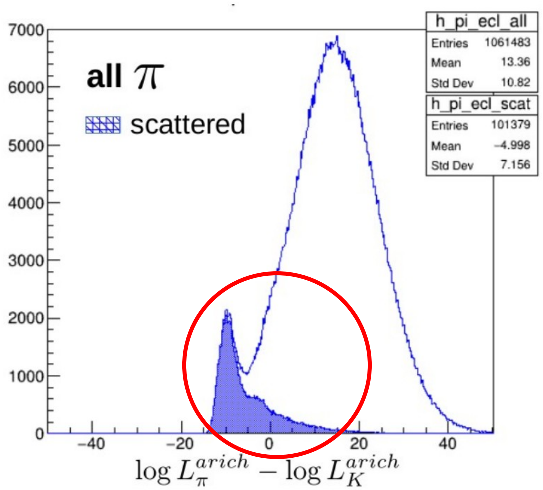
Global particle identification @ experiments

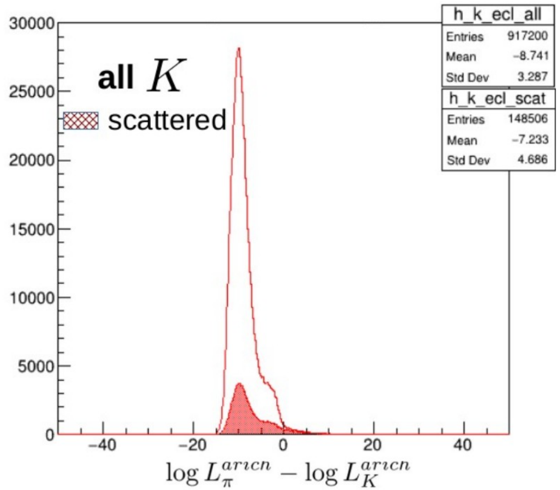
→ case example: **global PID @ Belle II**

→ in the FWD endcap ~10% of particles with extrapolated track in the ARICH do not pass through it / or strongly displaced (material scattering / decay in flight)

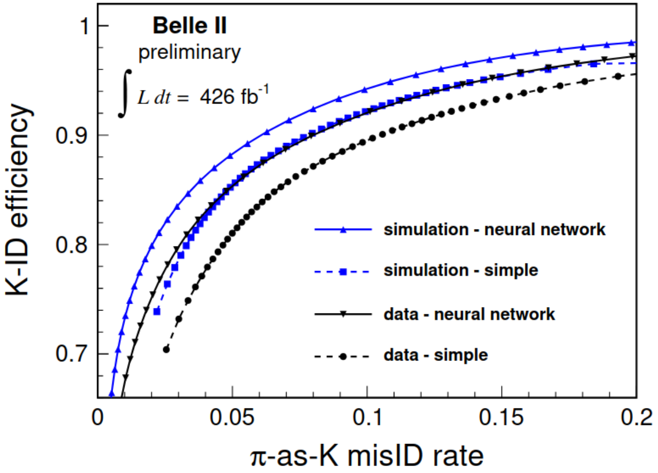
→ most problematic for pions at momenta below K threshold
(since scatter/decay probability is not considered in ARICH likelihood → strongly misidentified pions in global PID!)

→ new neural net based classifier uses 36 likelihoods $\mathcal{L}_d^h$ and track parameters (p, θ, ϕ, q)





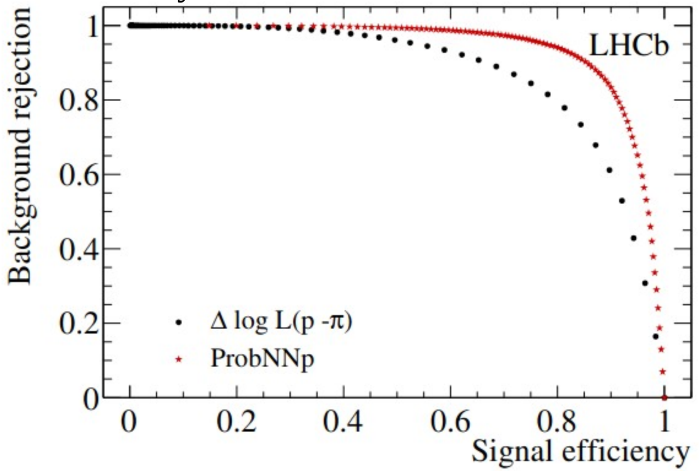
arXiv:2506.04355



Global particle identification @ experiments

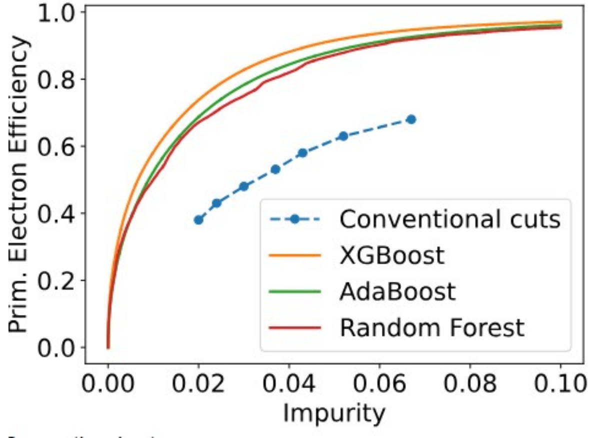
→ similar approaches used at other experiments (LHCb, ALICE, CBM, AMS-02)

J. Phys.: Conf. Ser. 1085 042038

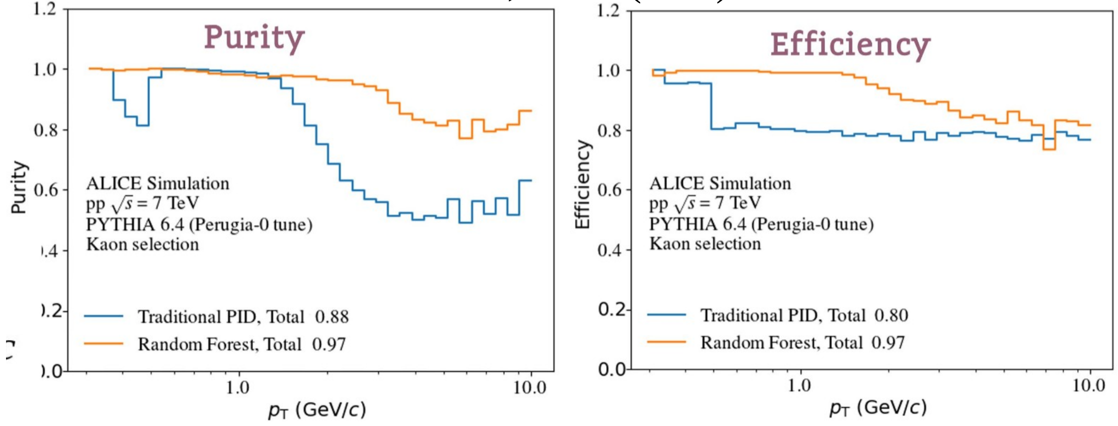


See poster from M. Beyer:
CBM RICH ring reconstruction using machine learning

CBM electron ID / simulation



EPJ Web of Conferences 295, 09029 (2024)



Machine learning in Cherenkov ring reconstruction

→ in recent years large progress in image classification algorithms: particularly **convolutional neural networks** (CNNs) have proven to excel in image and pattern recognition tasks

→ PID is essentially classification problem, often with 2D detector images!

→ in its most “radical” form this approach abandons event reconstruction altogether and feeds “raw” images into NN

→ at least at exploratory level several tries can be found in literature

→ there are several challenges in using such approaches (understanding PID systematics, variable experimental conditions etc.)

→ benefit is speed! Once NN is trained its application is very fast compared to traditional event reconstruction methods

Machine learning in Cherenkov ring reconstruction

Example: **CNN @ LHCb RICH**

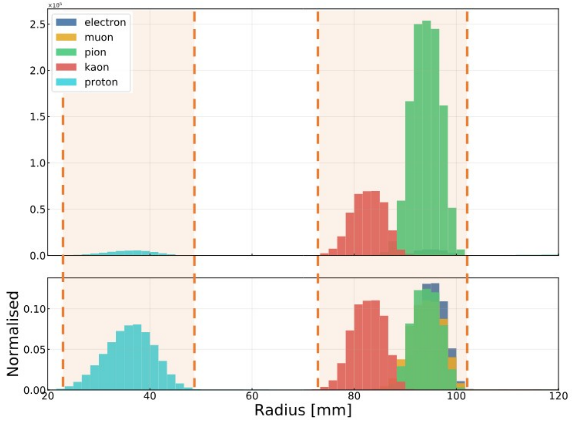
→ around each extrapolated track center radial search of hits is done in selected radius range

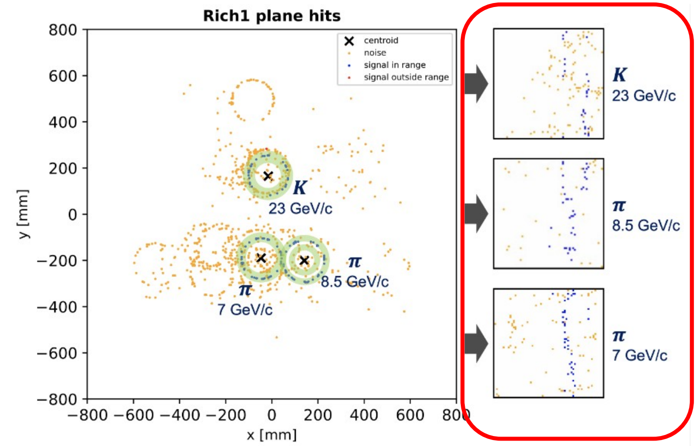
→ hits in this range are polar transformed to obtain 64x64 pixel images (radius, ϕ)

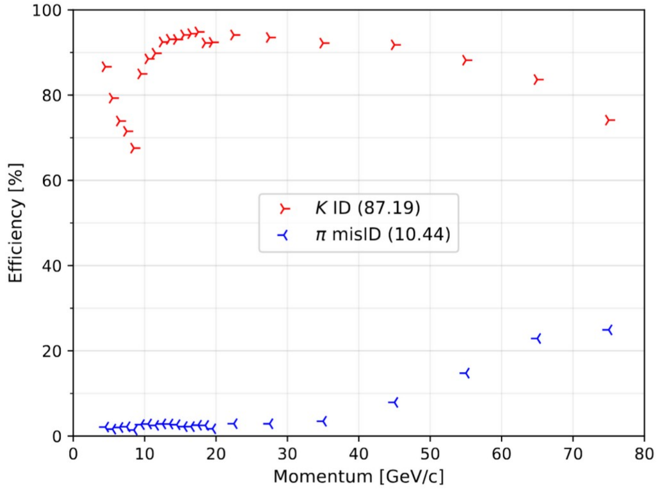
→ CNN is trained on images in 1 GeV/c wide momentum bins

→ up to 50 GeV/c performance close to traditional reconstruction is achieved, but notably worse at momenta above

J. Phys.: Conf. Ser. 2438 012076, 2023







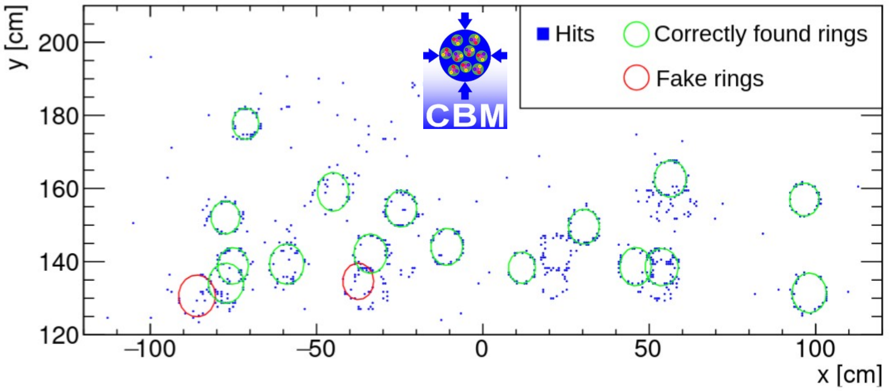
Machine learning in Cherenkov ring reconstruction

Example: **ML @ CBM**

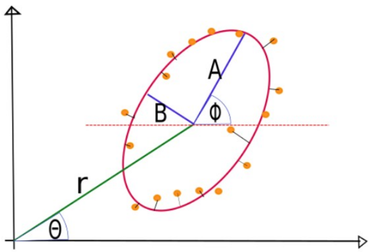
→ rings are first reconstructed using Hough transform method

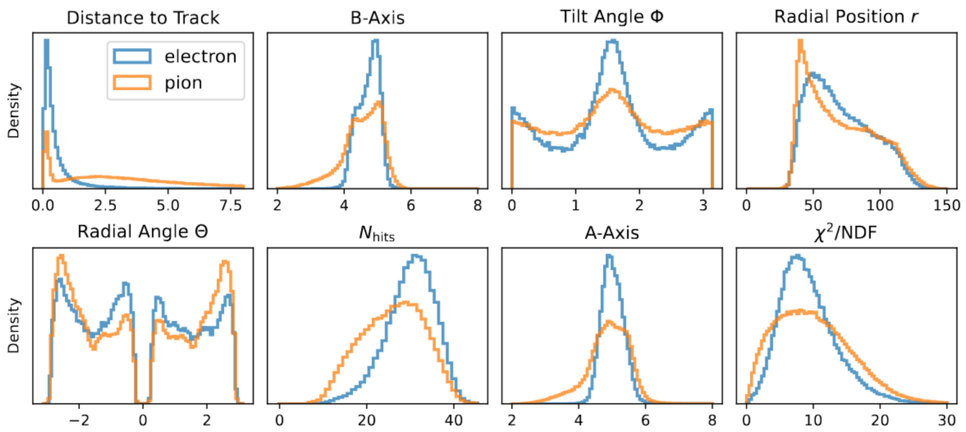
→ several man-crafted observables are then used as input to ML

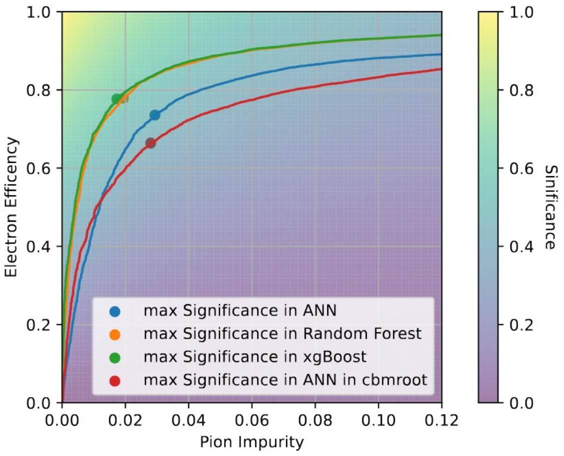
→ BDT based algorithms (xgBoost & Random Forest) result have notably better performance than their originally used NN



CBM RICH single-event-display of upper photo camera @ 11 AGeV Au+Au mbias







From H. Schiller thesis: Application of Machine Learning to Particle Identification for Dielectron Analysis in CBM, 2022

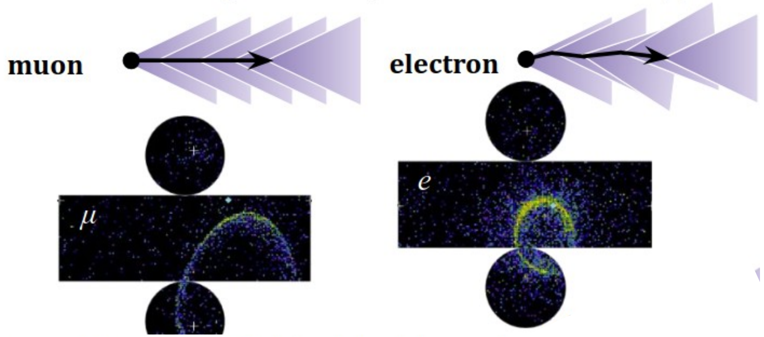
Machine learning in Cherenkov ring reconstruction

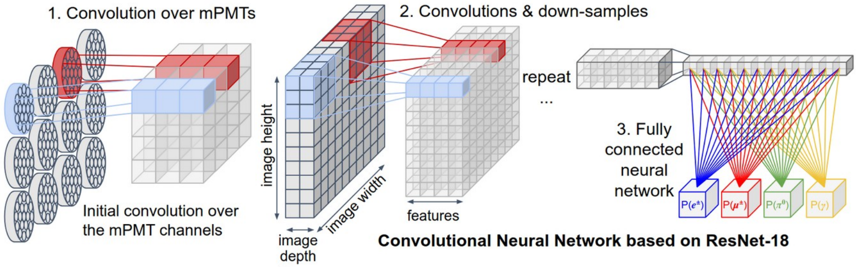
Example: **CNN for water Cherenkov detectors**

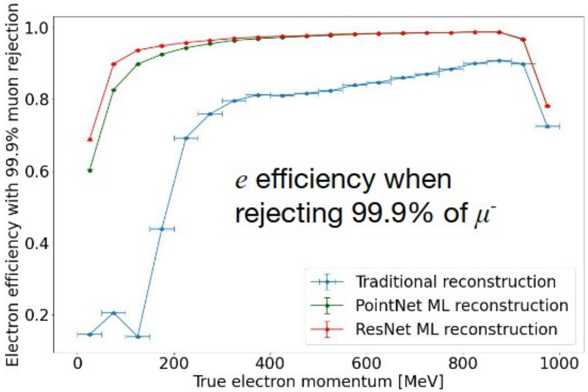
→ computational time of traditional Maximum-Likelihood reconstruction is becoming a limiting factor

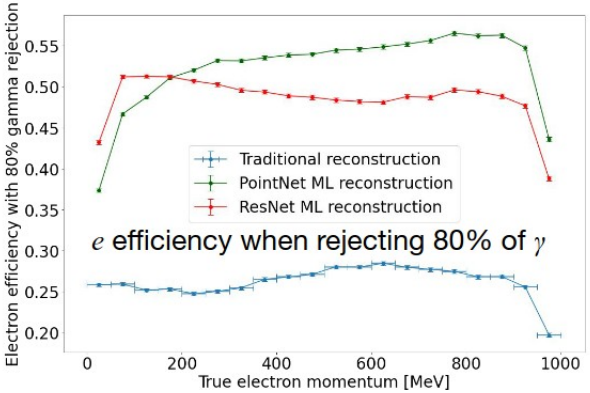
→ CNNs once trained are very fast
- at Hyper-K: 1 event reconstruction ~1 minute on CPU
- CNN on GPU can process O(100k) per minute

→ in addition PID performance greatly improved
(e.g. e/γ discrimination not possible w/ traditional reconstruction methods)









Phys. Sci. Forum 2023, 8(1), 63;
https://doi.org/10.3390/psf2023008063

Generative AI

→ generative AI is nowadays everywhere

→ it can generate also useful things and is entering also particle and nuclear physics experiments in recent years

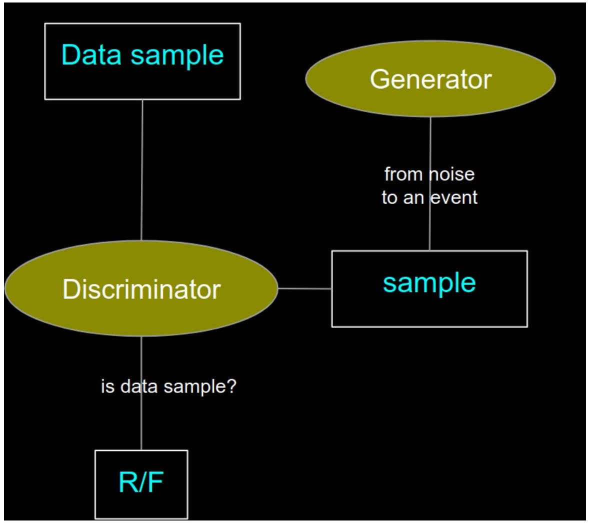
→ the use can be two-fold: - (very) fast simulations (instead of using Geant4 higher level observables are generated)
- classification (e.g. generate PDFs for given track parameters)

→ typically so-called Generative Adversarial Network (GAN) architecture is used

→ it is 2 NN game: one model maps noise to images (generator), the other (discriminator) classifies the images if real or fake.

→ networks are trained in turns, with the goal highest confusion rate for the discriminator. Generator effectively learns parameters of the probability distribution that underlies our data, then we can sample data points from this distribution to obtain new data points

https://arxiv.org/abs/1406.2661



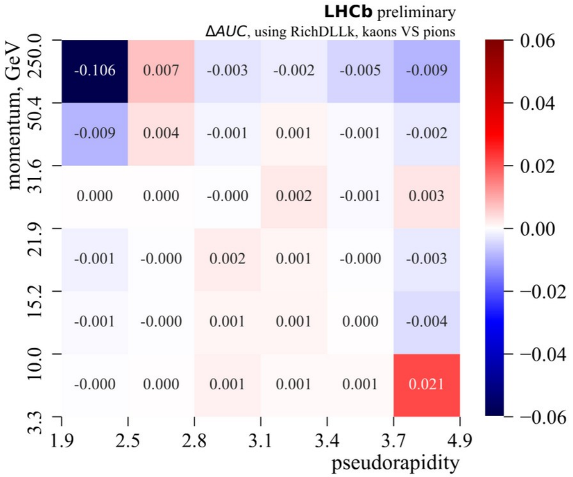
Generative AI

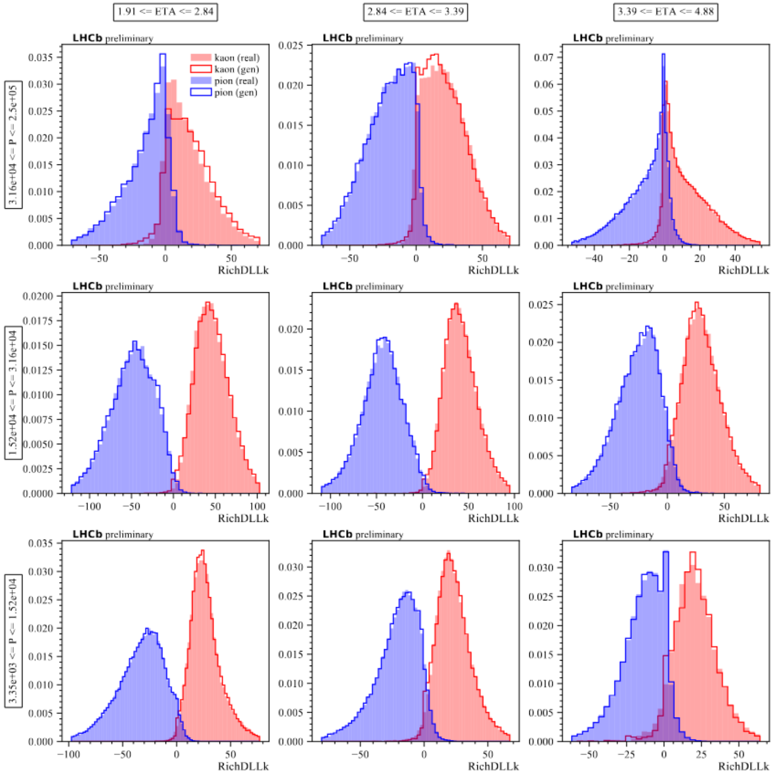
Example: **GANs @ LHCb** J. Phys.: Conf. Ser. 1525 012097, 2020

→ goal is to generate PID likelihoods for tracks in an event, based on particle type, momenta, polar angle, and total number of tracks in an event.

→ real data from control samples is used for training (w/ backgrounds sPlot subtracted)

→ obtained differences in AUC between real and generated data of the order 0.1-1%.





Generative AI

Example: **Deep(er)RICH**

→ extensive work by C. Fanelli et al on GAN models for DIRC-like devices

→ general framework with existing adaptations for GlueX DIRC and hpDIRC at EIC

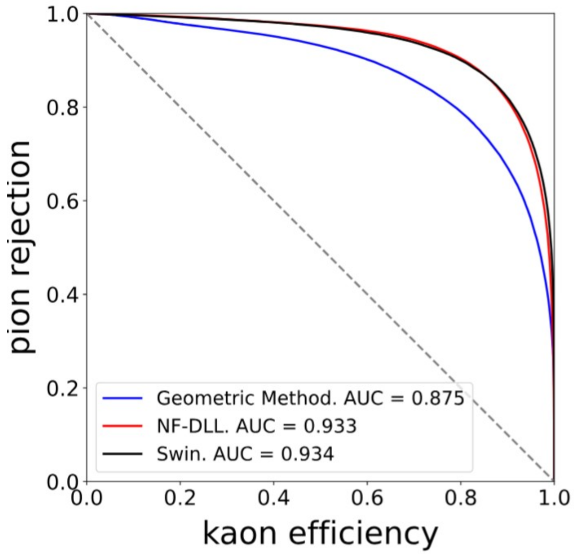
→ used for PID and fast simulation

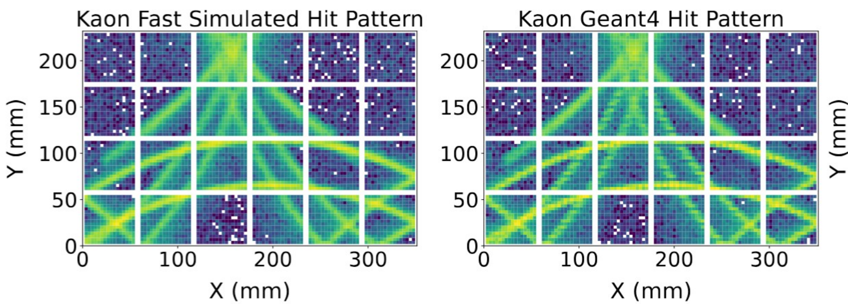
→ GPU deployment:
 - avg. PID time 9us
 - fast sim: 0.5us per hit

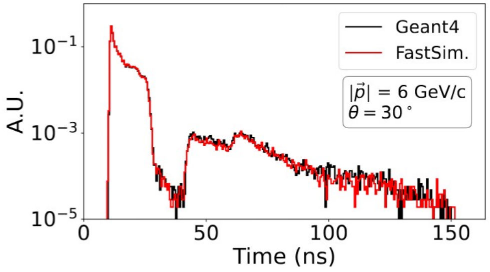
https://arxiv.org/abs/2504.19042

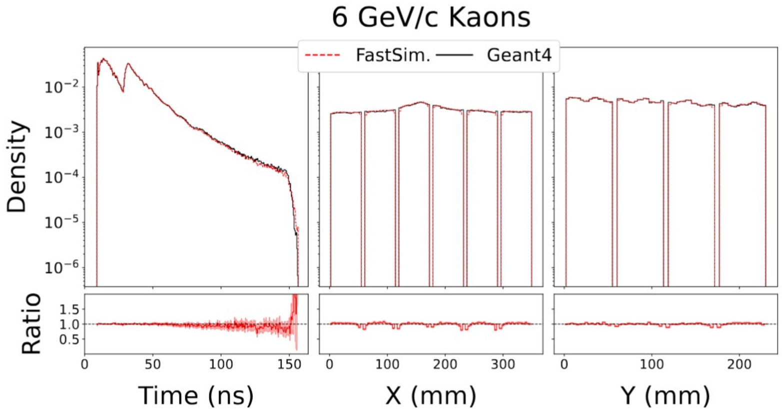
Mach. Learn.: Sci. Technol. 6 015028, 2025

Mach. Learn.: Sci. Technol. 1 (2020) 015010









Conclusions

→ basics of two “traditional” methods used for reconstruction of RICH data that are adopted across a wide range of experiments were presented

- likelihood method
- Hough transform

→ with fast development of Machine Learning techniques these are becoming more widely used also in our field

Some personal remarks:

→ we indeed should explore advantages of these new tools
→ but we have better methods for fitting a Gaussian than CNNs are
→ they need to be used smartly, with caution, and for purposes that they are good at

→ if nothing else, better performance of ML based method than your traditional method tells you there still is something to be learned and that you should work harder on your likelihood!

Thank you for the attention

Backup

Cherenkov angle distribution in Belle II ARICH- Data / MC comparison