IBM Evaluation of power architectures for machine learning

Daniel Hugo Cámpora Pérez
dcampora@cern.ch

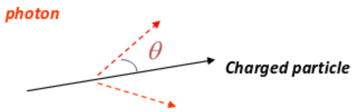
January 23rd, 2019

Universidad de Sevilla
CERN

The RICH reconstruction

Problem formulation

Cherenkov radiation principle



$\cos(\theta) = 1/(n\,\beta)$ where n = Refractive Index = $c/c_M = n(E_{ph})$

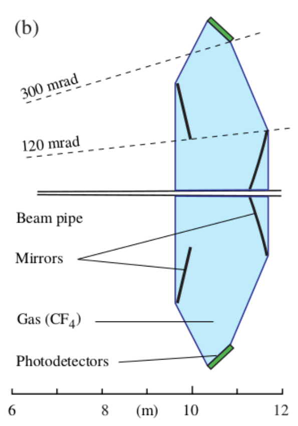
$$\beta = v/c = p/E = p/(p^2 + m^2)^{0.5} = 1/(1+(m/p)^2)^{0.5}$$

β = velocity of the charged particle in units of speed of light (c) vacuum
p, E, m = momentum, Energy, mass of the charged particle.
C_M = Speed of light in the Medium (Phase velocity),
E_{ph} = Photon Energy, λ=Photon Wavelngth.

- Theory of Cherenkov Radiation: Classical Electrodynamics by J.D.Jackson (Section 13.5)

RICH detectors in LHCb

In LHCb, we have two RICH detectors. Below is a schematic XZ view of RICH1:



Analytical solution

The analytical solution consists in *creating photons*, ie. associations of detected pixels in the HPDs / MaPMTs with their originating track segments through the Rich detector.

Once this association is found, a likelihood minimisation algorithm is run in order to find the most likely candidate for each particle.

- Photon creation, heavily involving ray tracing
- Likelihood minimisation

Analytical solution - Ray tracing

The creation of the photons involves a ray tracing algorithm from each segment to the candidate pixels. In turn, this means solving the quartic equation:

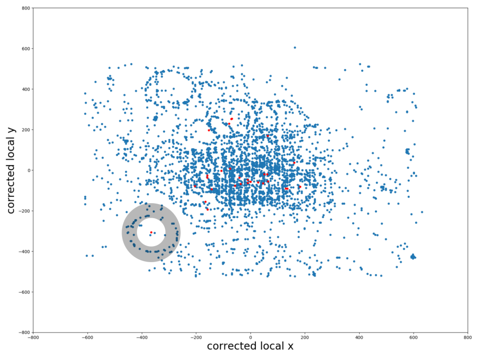
$$4e^2d^2\sin^4\beta \;-\; 4e^2d_yR\sin^3\beta + \left(d_y^2R^2 + (e+d_x)^2R^2 - 4e^2d^2\right)\sin^2\beta$$
$$+\; 2ed_y(e-d_x)R\sin\beta + (e^2-R^2)d_y^2 \;=\; 0\,. \qquad (3)$$

This can be solved, using for example a routine in the CERN library [6], and gives four solutions for $\sin\beta$, two complex and two real. Of the real solutions one is the "backward" reflection (that would exist if the mirror were a complete sphere, shown as M′ in Fig. 3); the other is the desired solution, and can be selected from knowledge of the RICH detector geometry. The value of $\cos\beta$ can then be extracted using Eq. 2, and the coordinates of the reflection point M determined.

Bringing ML into the fray

Alternatively, we are studying whether it is possible to transform the problem into a classification problem from an image into a particle ID (one of *pion*, *muon*, *electron*, *kaon*, *proton*, *deuteron*):

- Each track is extrapolated onto the detector plane (red dots)
- A *corona* shape is fed onto a Convolutional Neural Network to identify the particle



Motivation

If we observe the surrounding area of a single track extrapolation and convert it to polars, we end up with the figures below:

Figure 1: Left: Pion. Right: Electron.

Exploring possibilities

We are exploring adding several features to the system:

- Position of the track (x, y)
- Momentum of the track
- Identify hits in several coronas

with various CNN designs:

```
[CONV2D 32] x2
POOL
[CONV2D 64] x2
POOL
[CONV2D 128] x2
POOL
FC 512
FC 128
FC 6 (output)

Trainable params: 4M
```

Ongoing research

We are mainly interested in distinguishing heavy particles from light particles, where early results report about 90% identification efficiency. This is promising, but still a work in progress.

Predictions (%tot)	heavy	light	Efficiency (%)
heavy	18.090	7.482	70.742
light	2.613	71.815	96.489
Purity (%)	87.377	90.565	

Predictions (%tot)	deuteron	electron	kaon	muon	pion	proton	Efficiency (%)
deuteron	0.000	0.000	0.013	0.000	0.047	0.008	0.000
electron	0.000	0.003	0.080	0.000	1.942	0.015	0.163
kaon	0.000	0.000	10.822	0.000	5.717	0.262	64.415
muon	0.000	0.000	0.020	0.000	0.495	0.002	0.000
pion	0.000	0.000	1.210	0.000	70.348	0.313	97.880
proton	0.000	0.000	1.970	0.000	3.580	3.153	36.231
Purity (%)	0.000	100.000	76.668	0.000	85.656	84.014	
ID eff (%)	K->K,Pr,D : 65.972	pi->e,m,pi : 97.880					
MisID eff (%)	K->e,m,pi : 34.028	pi->K,Pr,D : 2.120					