PID Reconstruction

Uli Einhaus

ECFA Focus Group $H \rightarrow s\bar{s}$

16.05.2024

Overview

- Particle ID technology proposals for Future HTE Factories
- New framework CPID for HTE Factories, implementation in ILD
- Focus: kaon ID (in particular for strange tagging)

Lepton ID via calorimeters and trackers

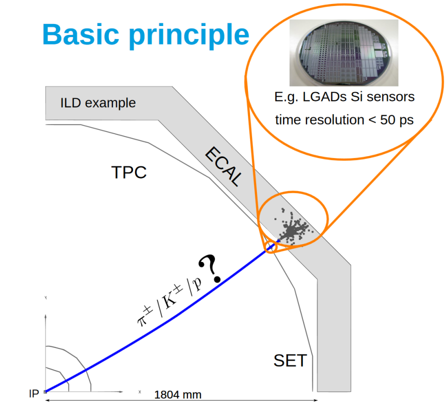
- ECal cluster vs. HCal cluster vs. muon chamber → e/γ vs. hadron vs. μ
- Has track → charged, i.e. e vs. γ and charged vs. neutral hadrons

- Very basic, done in any future detector
- Generally high efficiency/purity, depends mostly on calorimeter
- Can be built in the calorimeter reconstruction, e.g. Pandora Particle Flow

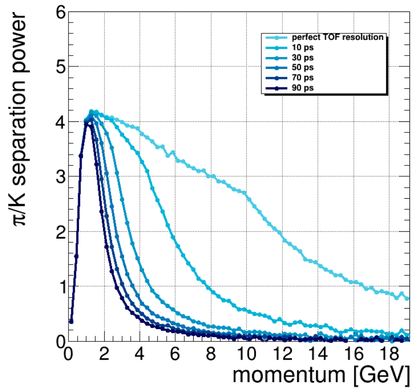
Time of flight

- Recent developments driven by pile-up rejections requirements from LHC allow for ~ 30 ps timing precision → can be used for TOF PID
- π/K separation up to 5 GeV
- Direct dependence on achievable timing resolution, but also track length needs to be known to the same relative precision
- Generally, interaction region is small enough (< 10 ps) to use clock instead of reference timing layer at VTX
- Do timing measurement with longest possible lever arm, i.e. in or immediately before ECal
- Many possible hardware implementations, prefer generic and flexible implementation in simulation, applicable to any HTE factory

Basic principle



B. Dudar

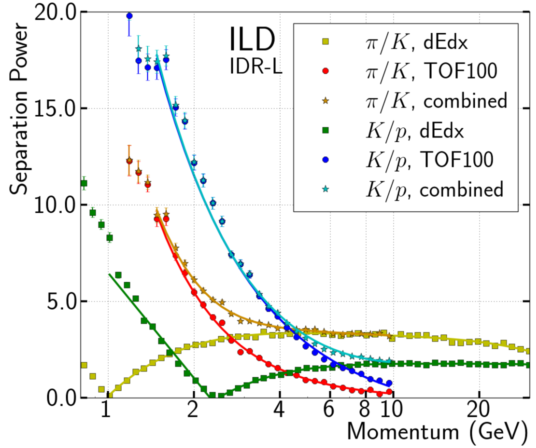


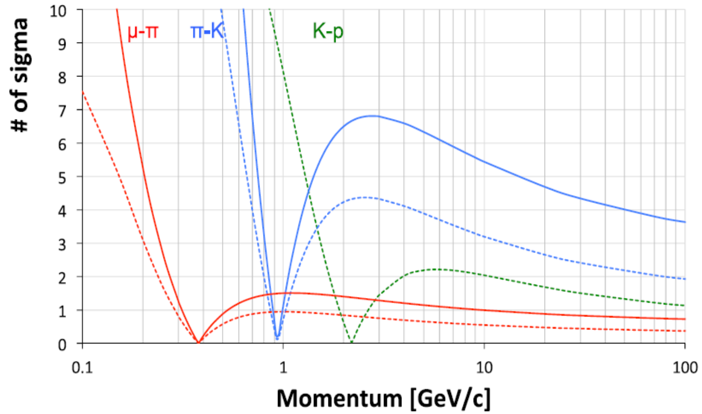
Specific energy loss

- Gaseous tracker allows correlation of specific energy loss and momentum
- Traditionally dE/dx, high granularity (in space or time) allow for dN/dx with up to factor 2 improved performance wrt. dE/dx
- π/K separation up to 20-50 GeV, ‘blind spot’ at 1 GeV

- Proposals:
  - time projection chamber at ILD and CEPC baseline det. with dE/dx, with PixelTPC readout also moderate dN/dx (spatial granularity)
  - drift chamber at IDEA and ALLEGRO with dN/dx (temporal granularity)

[ILD IDR 2020]

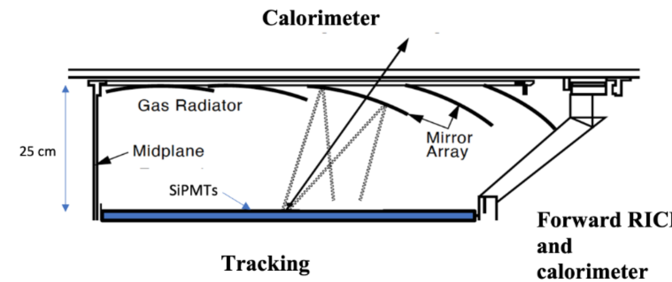




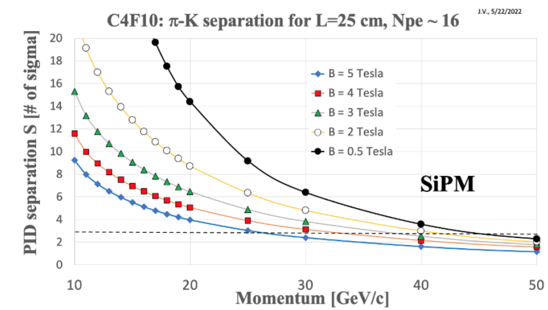
[IDEA, FCC-ee CDR 2019]

Cherenkov ring imaging

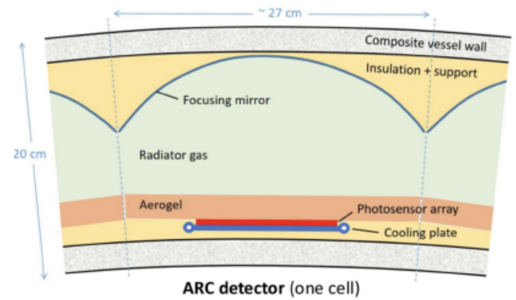
- Also well known technology, needs dedicated subdetector
  → additional material, compact RICH ~ 30 cm depth
- π/K separation up to 50-100 GeV

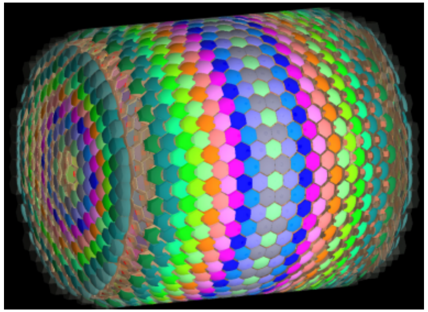


[J. Vav'ra e.a.](#)



- Largely redundant PID performance to sepecific energy loss, so proposals have focused on full-Si detectors
- Proposals:
  - RICH for SiD, single phase
  - ARC for CLD, with dual phase (gas and aerogel) to cover low-momentum range





[B. Francois](#)

PID combination task

- How to best combine these different PID obeservables to get the best combined PID for your tagger or analysis? How to assess the performance?
- Largely independent observables, but strong and different dependences on momentum and angle of incident particle

- Something with machine learning?
  And comparable between detectors concepts?

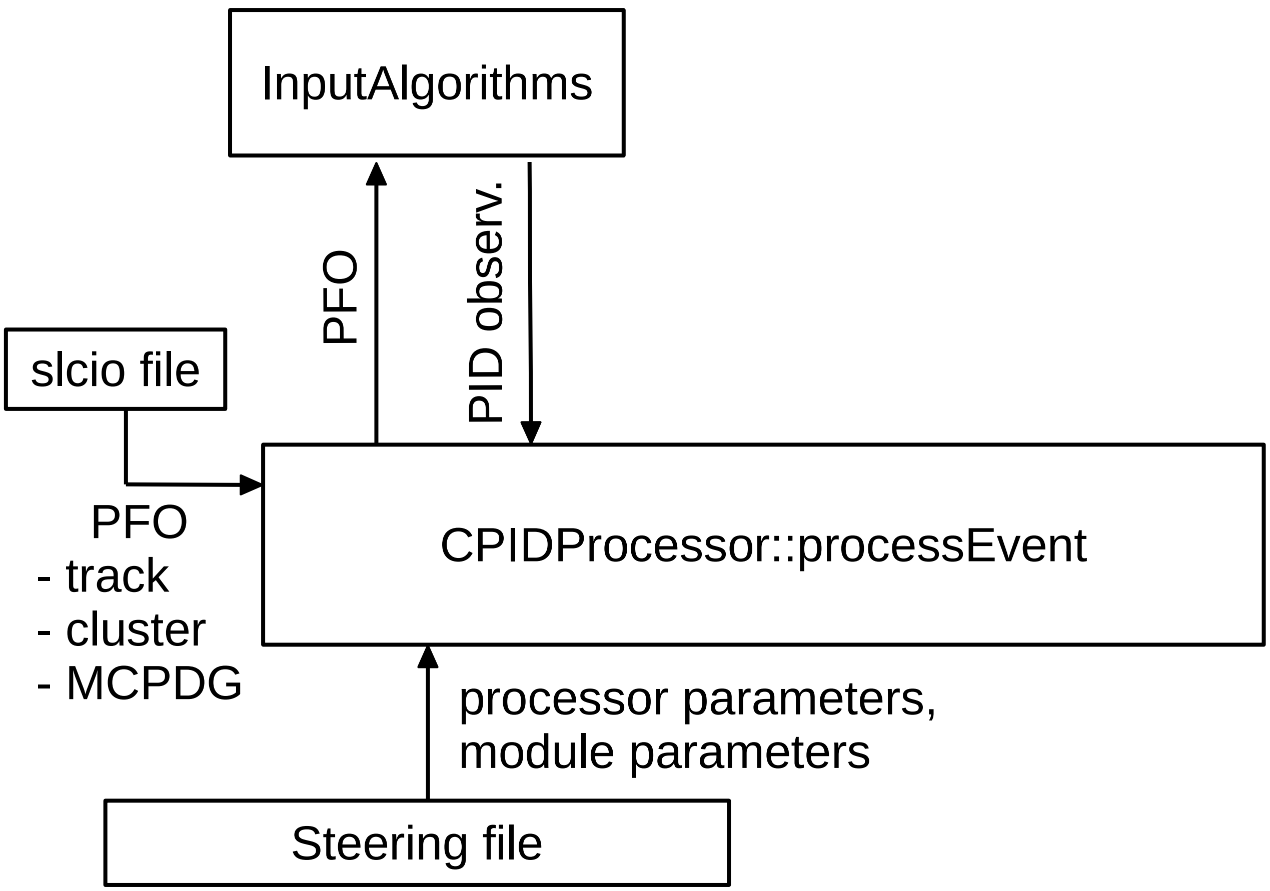
CPID

- Comprehensive Particle Identification (CPID) framework
- Target: provide platform for future collider detectors to evaluate PID
- Approach: central book-keeping, modules for PID observables as well as combination e.g. via ML models (training & inference)
- Use only reconstructed particles, i.e. Particle Flow Objects (PFOs)
- Currently Marlin processor using LCIO, usable in Gaudi via MarlinWrapper, goal is to have native implementation in EDM4HEP
- CPID is part of MarlinReco in the latest iLCSoft release

→ Structure, module overview, PID performance, calibration for ILD MC production

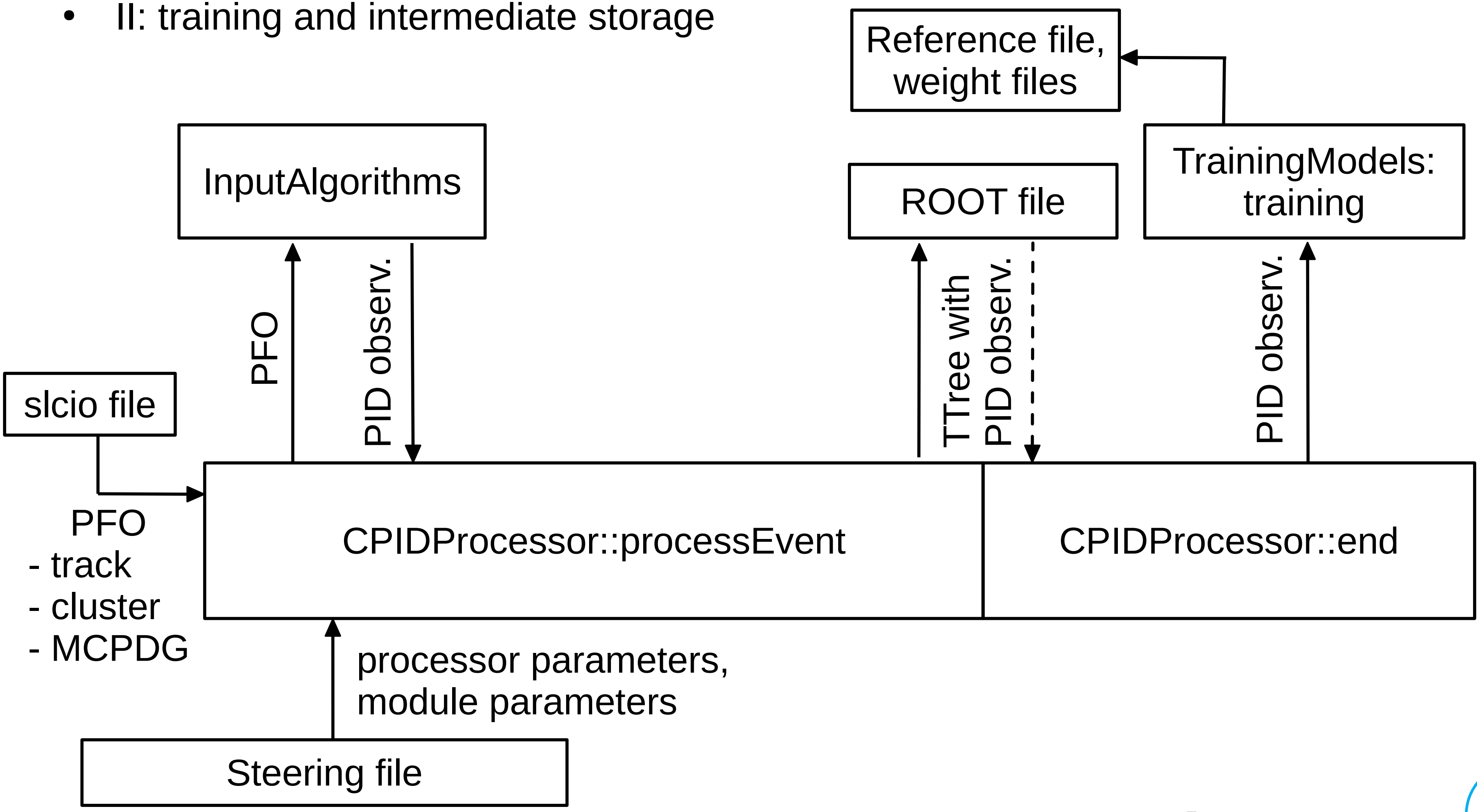
Structure of the CPID workflow

- I: set up and observable extraction



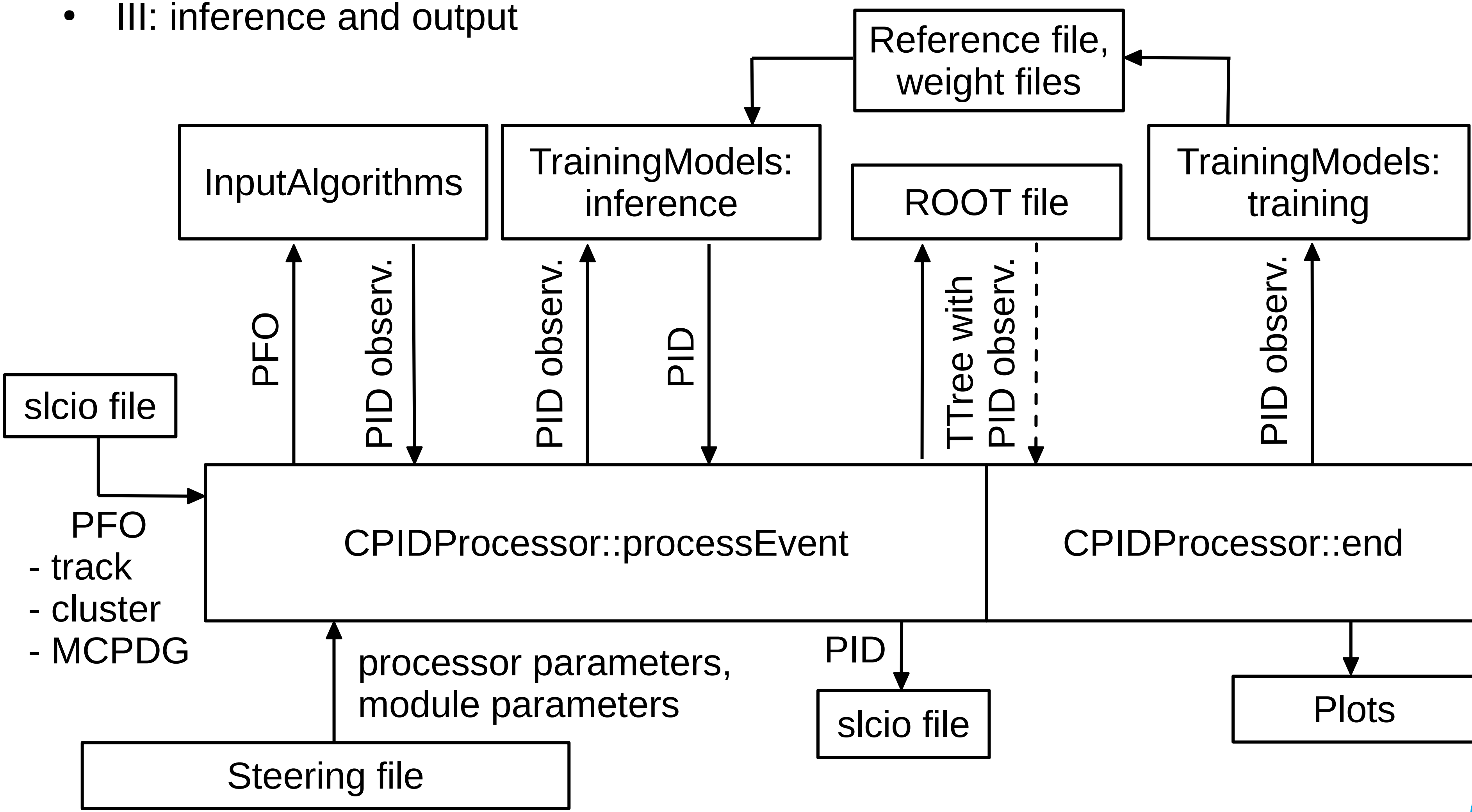
Structure of the CPID workflow

- II: training and intermediate storage



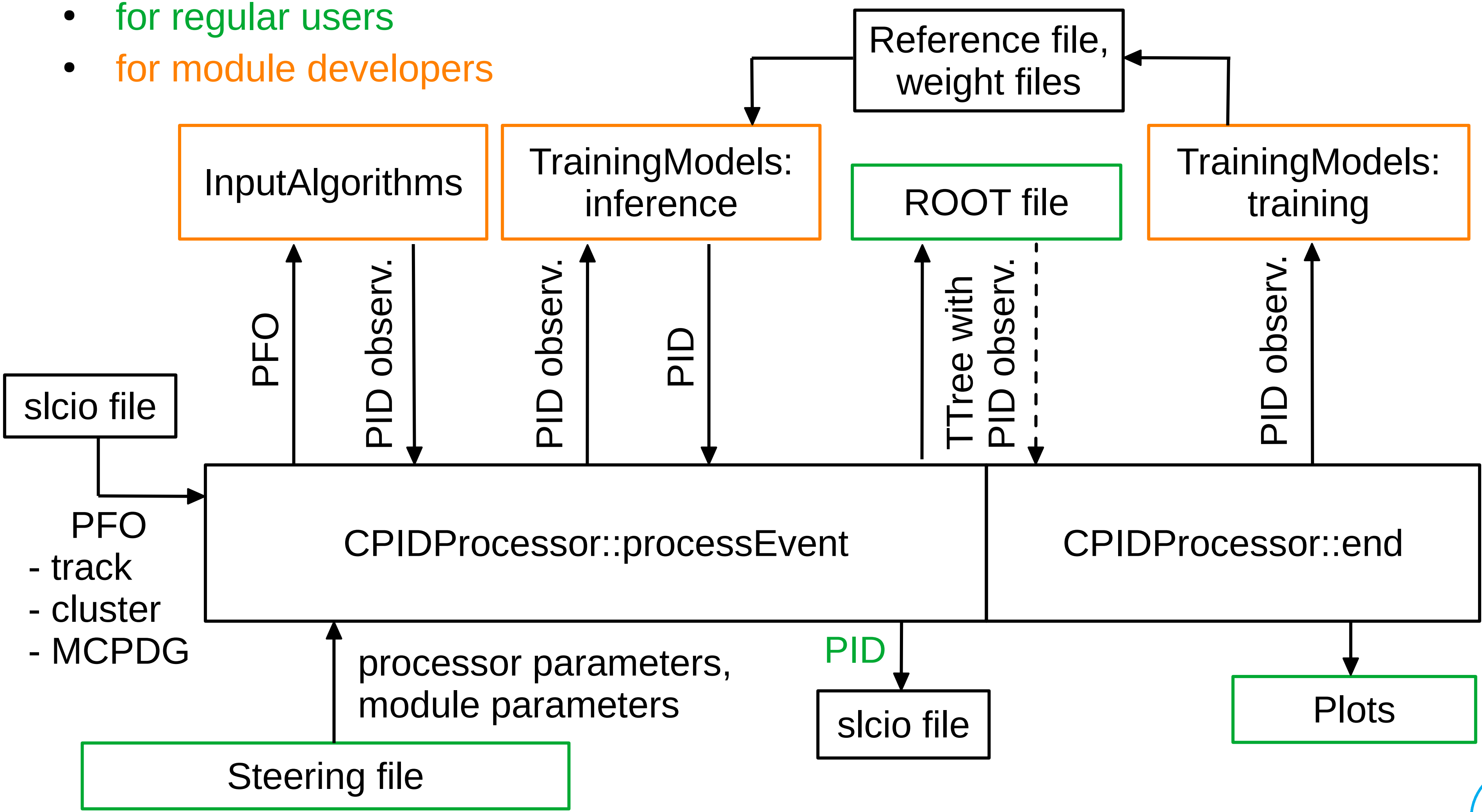
Structure of the CPID workflow

- III: inference and output



Structure of the CPID workflow

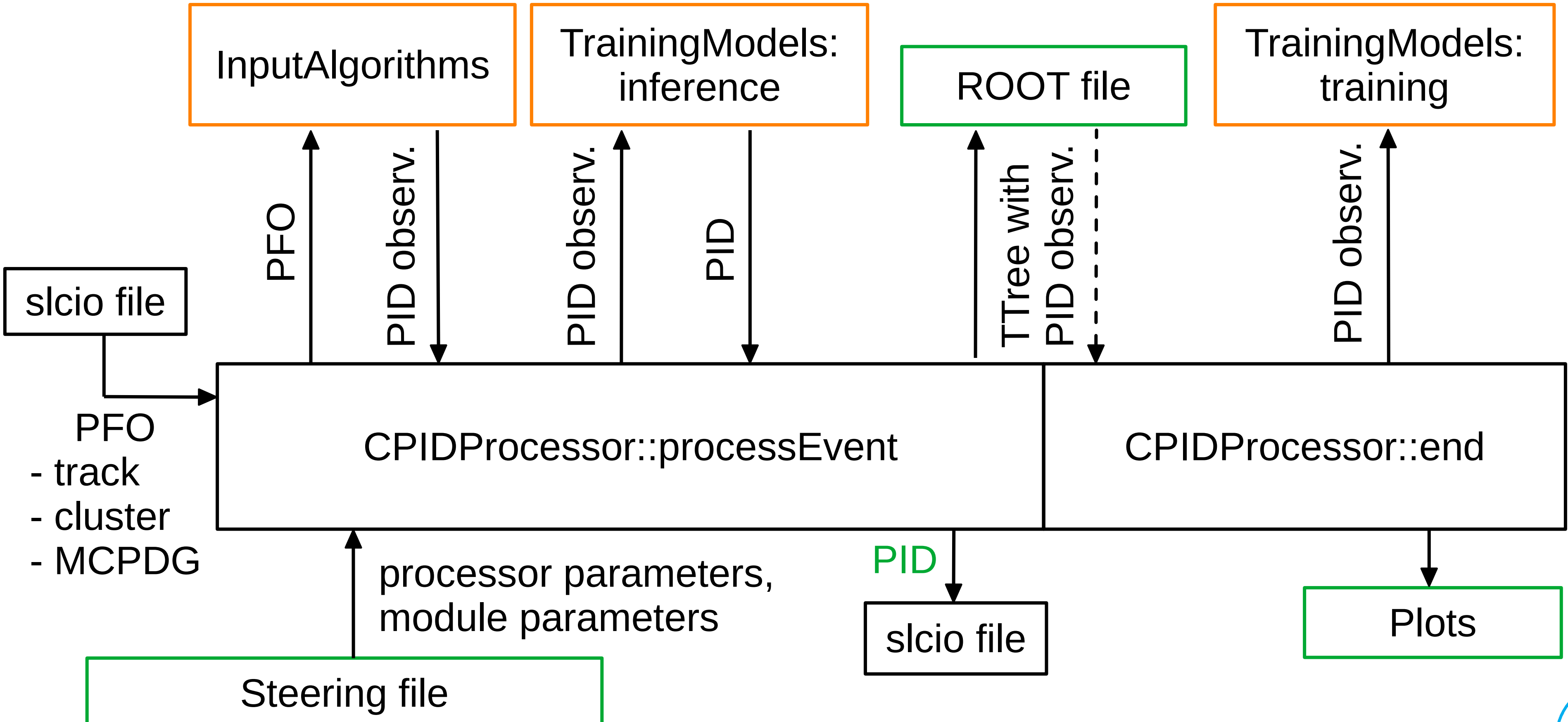
- for regular users
- for module developers



Structure of the CPID workflow

- for regular users
- for module developers

Dynamic loading of modules means module developers don’t need to touch the actual processor (analogous to Marlin processors and actual Marlin)



Module library

- PID observables modules
  - Pandora lepton ID
  - LeptonID (L. Reichenbach)
  - TOF
  - dE/dx
  - dN/dx

- Combination modules
  - BDT signal/background
  - BDT multiclass

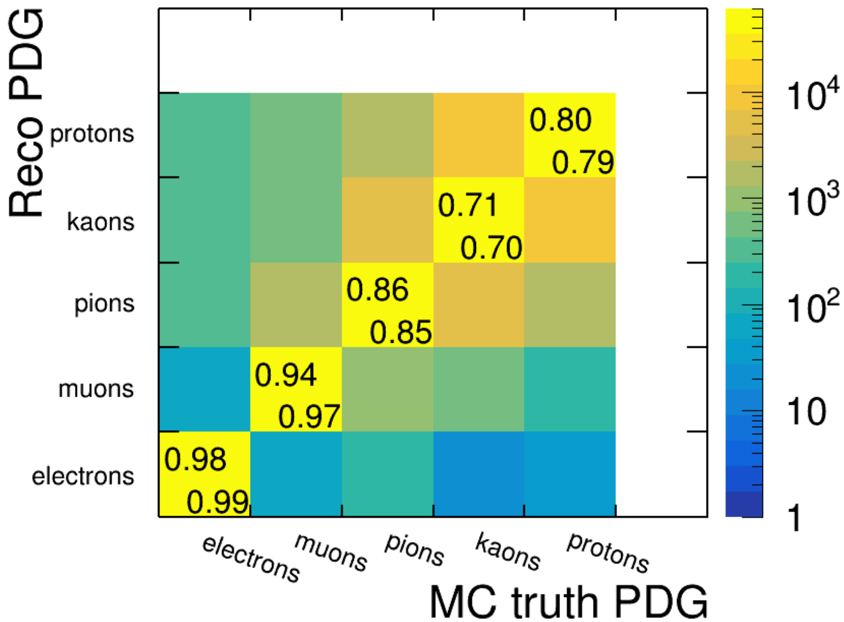
- Based on ROOT TMVA, so any other TMVAs can easily be added

- Missing: Cherenkov

- ‘Generic level’ modules (e.g. dN/dx based on IDEA Delphes) can work on their own, while ‘detailed level’ modules (e.g. dE/dx based on ILD full sim&reco) require separate algorithms to be run in the full-reconstruction chain

Typical performance plot

- Confusion matrix of reconstructed vs. MC PID for the 5 detector-stable charged particles (electrons, muons, pions, kaons, protons)
- Numbers on diagonal are efficiency/purity of that element, i.e. correctly identified PID
- Note: colour is log scale
- Use single particle samples with identical numbers of particles per species, flat in log(p) and isotropic
- Split momentum range of 1 - 100 GeV into 12 momentum bin with separate multiclass BDT each (to ease momentum dependence of PID observables)
- CPID Output: BDT score for each species hypothesis, for plot put in bin with highest score

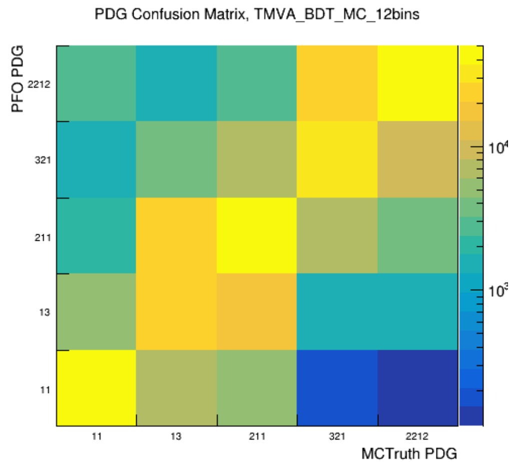


Modules used for ILD Std Reco

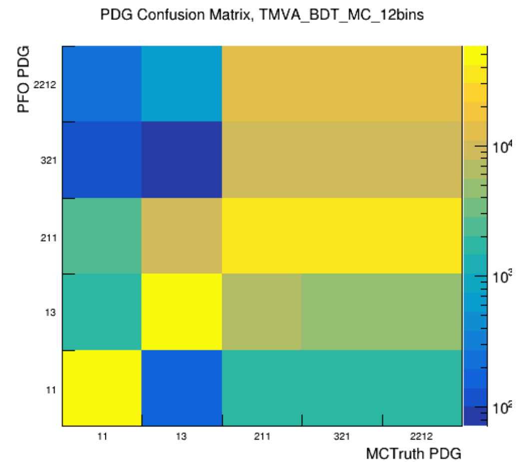
- Run CPID training on ILD to provide default weight files for current ~best PID
- This config is intended for the 250 GeV MC production of 2020
- To be run on distilled data (DST) files
- Use available observables:
  - Pandora PID output based on cluster info
  - LeptonID, largely re-assessment of cluster info + some dE/dx, some improvements over Pandora PID
  - Time of flight, 100 ps timing resolution for each of 10 first hits of a PFO in the ECal; unfortunately not ‘latest and greatest’ track length reco
  - dE/dx, resolution of about 4.5%

Individual observable performances

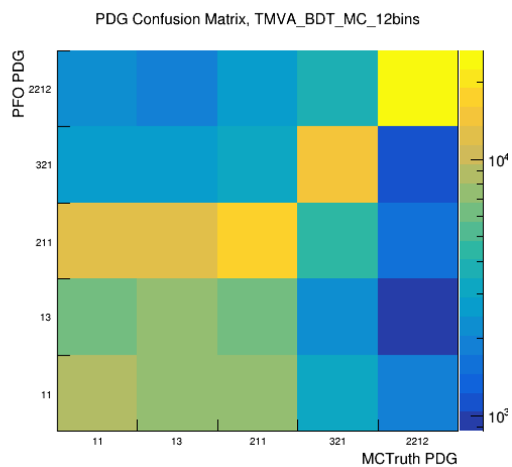
dE/dx



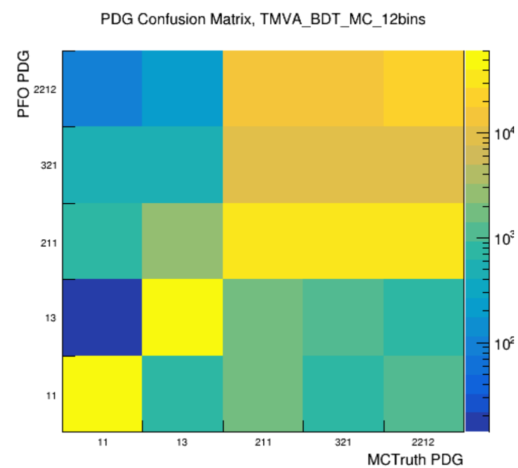
Pandora



TOF
(only from
1 - 10 GeV)

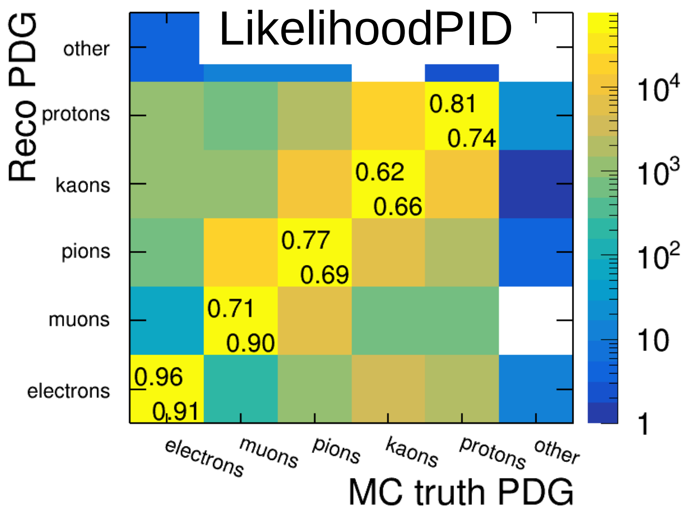


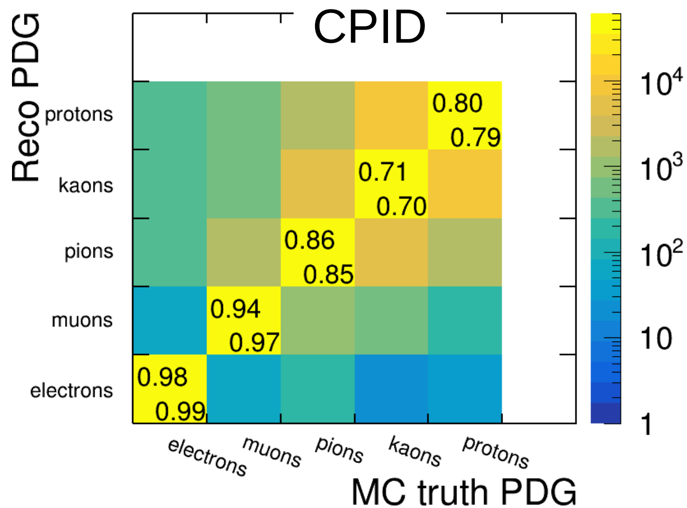
LeptonID



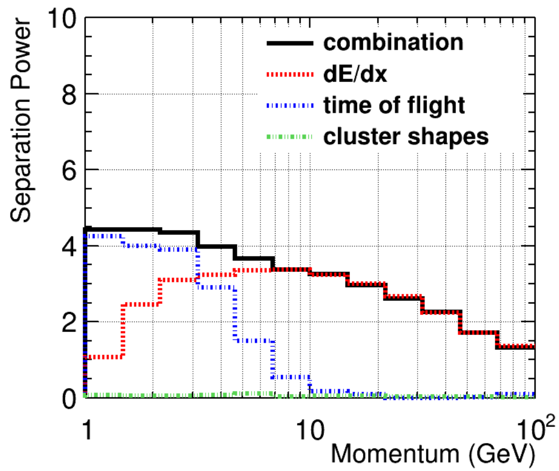
CPID calibration for 250 GeV

- Result below shows that it does everywhere better than the current LikelihoodPID, in particular thanks to the additional TOF
- Combination of several observables follows expected behaviour
- Check if training is universal or sample-specific
  → train on single particles, 2f-Z-hadronic, 4f-WW-hadronic, infer full cross-over
- Results on next slide: matrix of confusion matrices





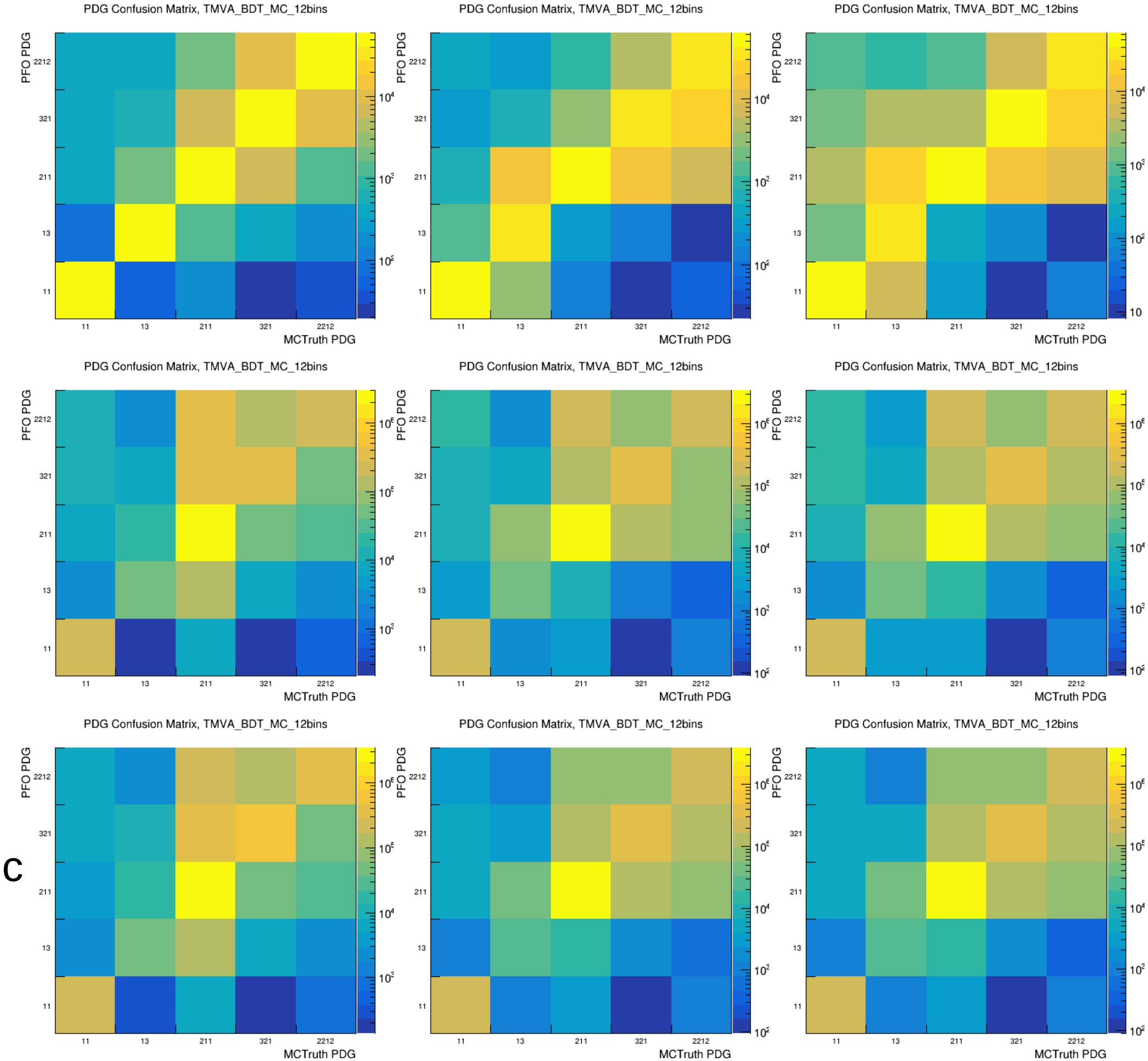
π/K separation



training on: single particles 2f-Z-hadronic 4f-WW-hadronic

inference to:

- single particles
- 2f-Z-hadronic
- 4f-WW-hadronic



CPID calibration result

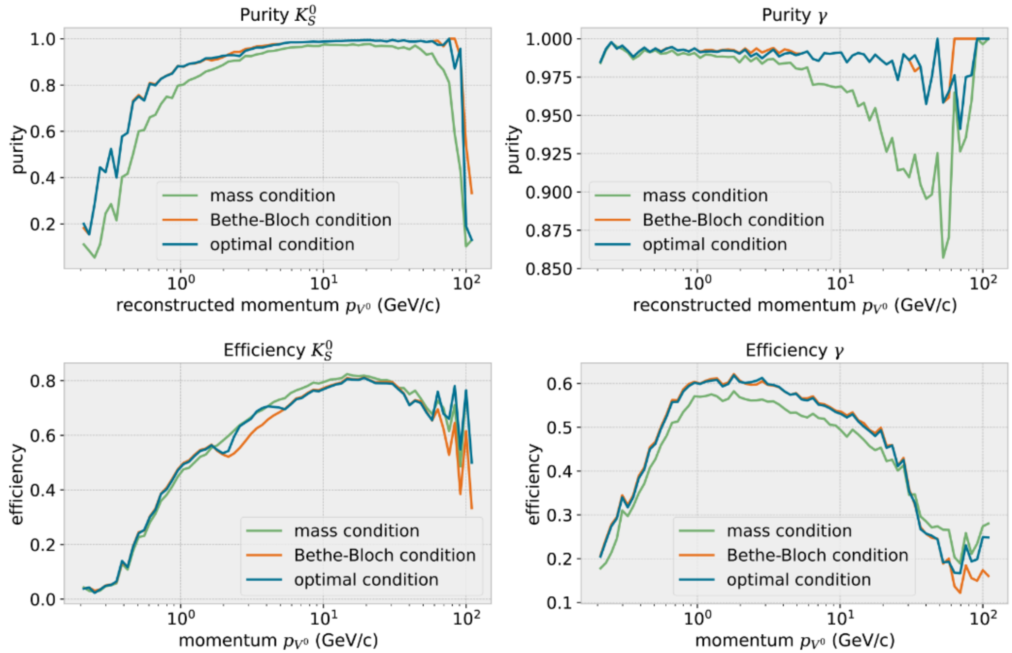
- Check if training is universal or sample-specific
  → train on single particles, 2f-Z-hadronic, 4f-WW-hadronic, infer full cross-over

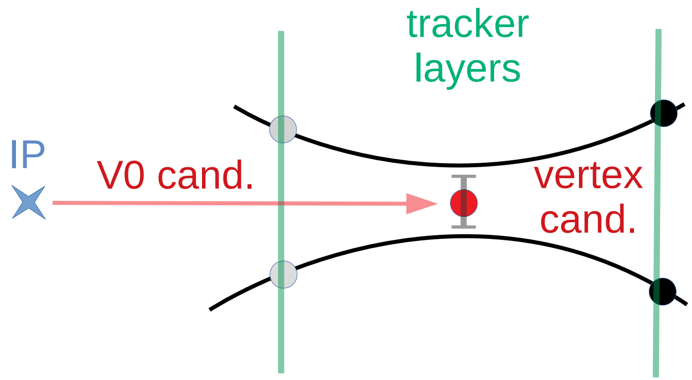
- Conclusion: training is sufficiently independent from sample to provide a default set of weights
- Pull request contains a set trained on single particles as default and one trained on 2f-Z-hadronic as alternative
- One can always run a dedicated training on a specific sample

- CPID has been added to the high-high-level reconstruction chain (MiniDST; together with isolated lepton finding, vertexing, flavour tagging, etc.) in ILD, intended to produce a full (?) MiniDST-level sample of the 2020 MC production

V^0s

- K^0_S and Lambda are strange hadrons, which often decay in the tracker and can be identified via their decay products + invariant mass reconstruction
- Decent performance in ILD, recently improved by using dE/dx for the decay products (reduced gamma background in reconstructed K^0s by factor 13)





S. Aumiller

Conclusion

- Particle identification is by now considered for the majority of Future HTE Factory detectors, several technological options and various combinations exist
- Challenge: combination and comparability
- Modular CPID framework developed in ilcsoft/key4HEP available, applied to ILD an calibrated to existing large-scale MC production

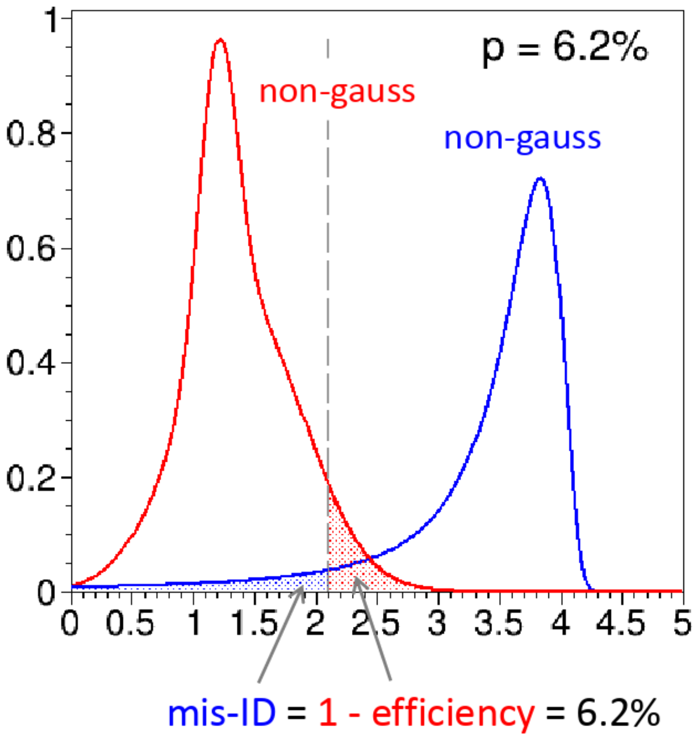
Thank you!

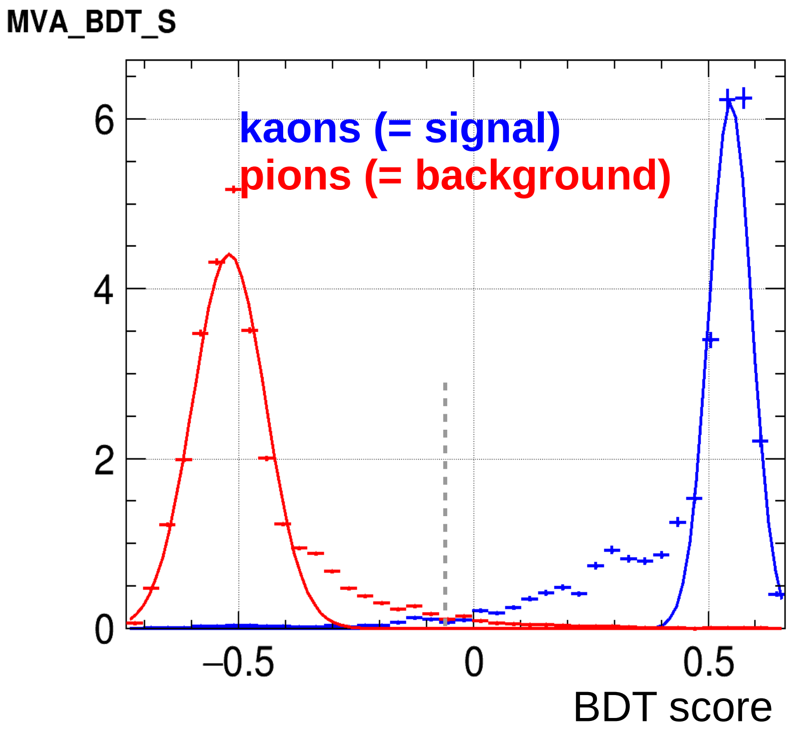
-

Backup

p-value Assessment

- Find cut with mis-ID = 1 - efficiency = p-value → find Gaussian quantile → compute Z = 2 · quantile of standard Gauss

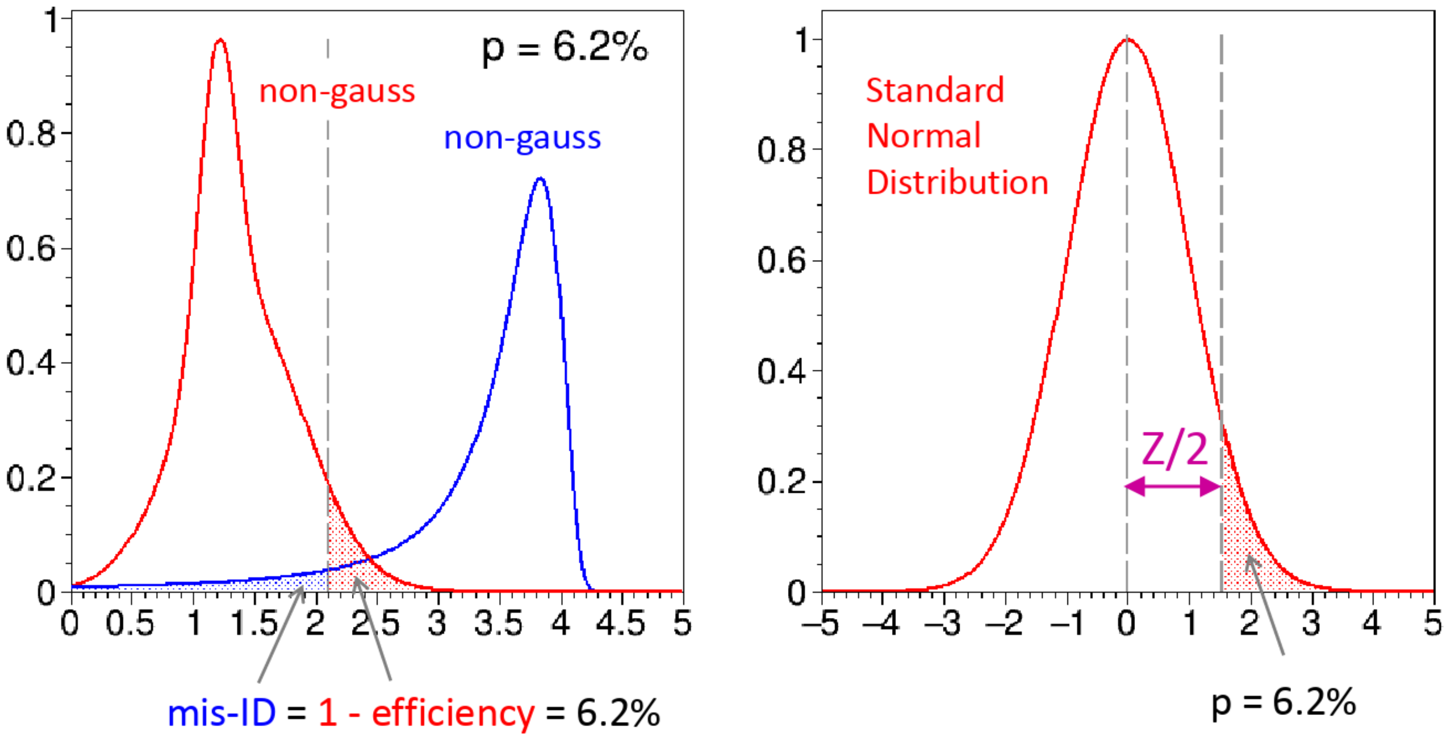




K. Götzen: https://indico.gsi.de/event/7080/contributions/31950/attachments/22952/28789/pid_kgoetzen_separationpower.pdf

p-value Assessment

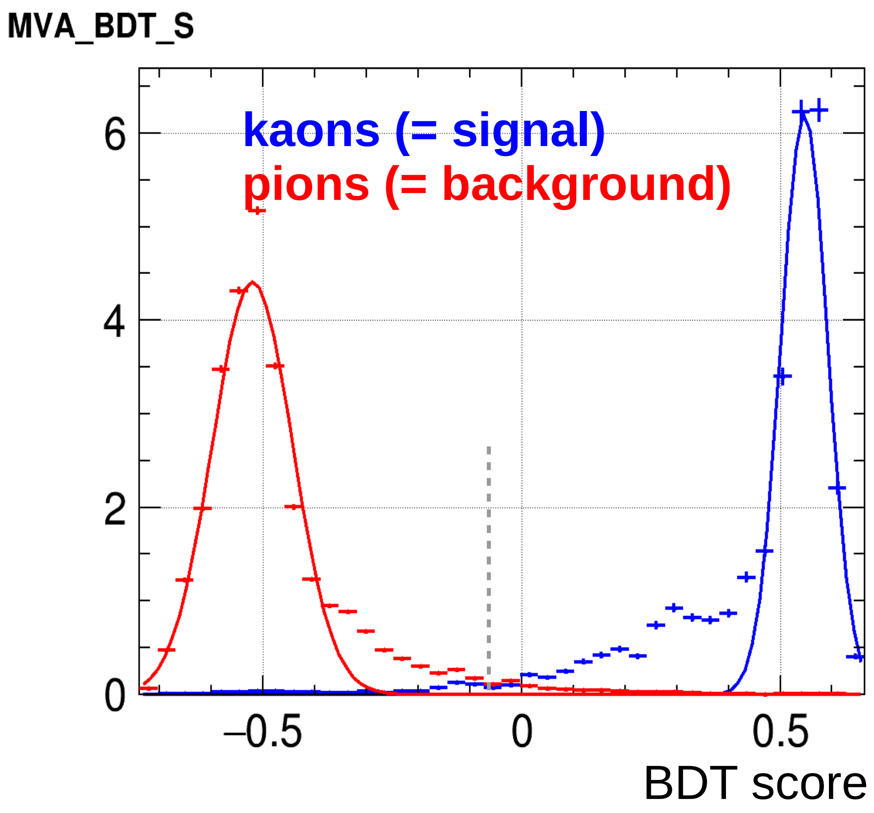
- Find cut with mis-ID = 1 - efficiency = p-value → find Gaussian quantile → compute $Z = 2 \cdot$ quantile of standard Gauss

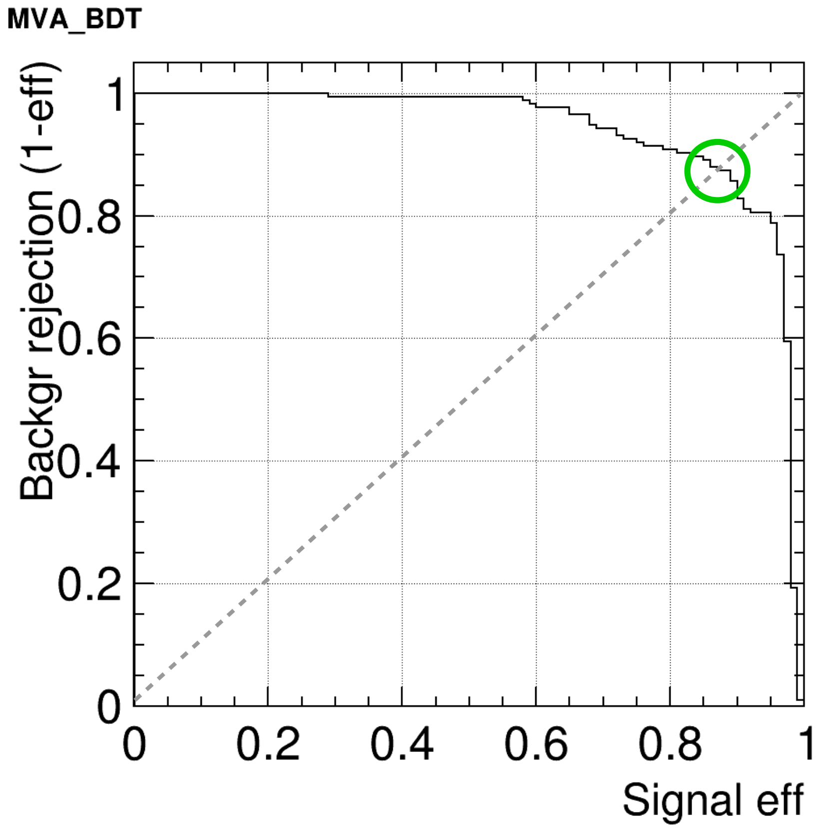


K. Götzen: https://indico.gsi.de/event/7080/contributions/31950/attachments/22952/28789/pid_kgoetzen_separationpower.pdf

p-value Assessment

- ‘Central tail split’ of BDT score is equivalent to crossing point of ROC curve with x=y line





How to access CPID output?

- CPID output is stored as PID info in the PFOs
- The PID algorithm name is the name of the CPID training model chosen in the Marlin steering file, in the example case: **TMVA_BDT_MC_12bins**
- The best PDG, i.e. the one with the highest BDT score, is returned by

```
_PIDMethod = “TMVA_BDT_MC_12bins”;
PIDHan = new PIDHandler(PFO_collection);
PDG = PIDHan->getParticleID(PFO,PIDHan->getAlgorithmID(_PIDMethod)).getPDG();
```

- The individual BDT scores are stored as parameters in the PID info, with the names constructed from the PDG numbers defined as signal PDGs extended by “-ness”, so: **11-ness, 13-ness, 211-ness, 321-ness** and **2212-ness**
- To return the BDT score for the electron hypothesis:

```
Para = PIDHan->getParticleID(PFO,PIDHan->getAlgorithmID(_PIDMethod)).getParameters();
score = Para[PIDHan->getParameterIndex(PIDHan->getAlgorithmID(_PIDMethod),”11-ness”)];
```