AI/ML Particle Identification for the ALERT Detector

CLAS Collaboration Meeting

November 17th, 2026

Uditha Weerasinghe

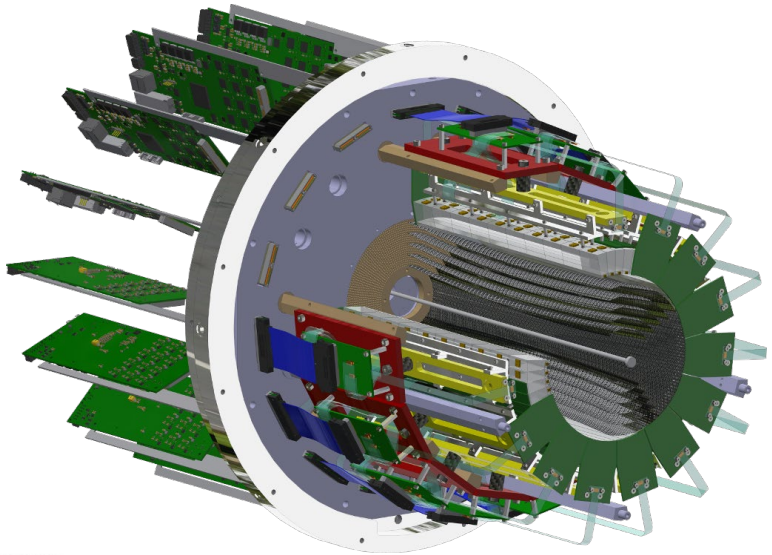
Mississippi State University

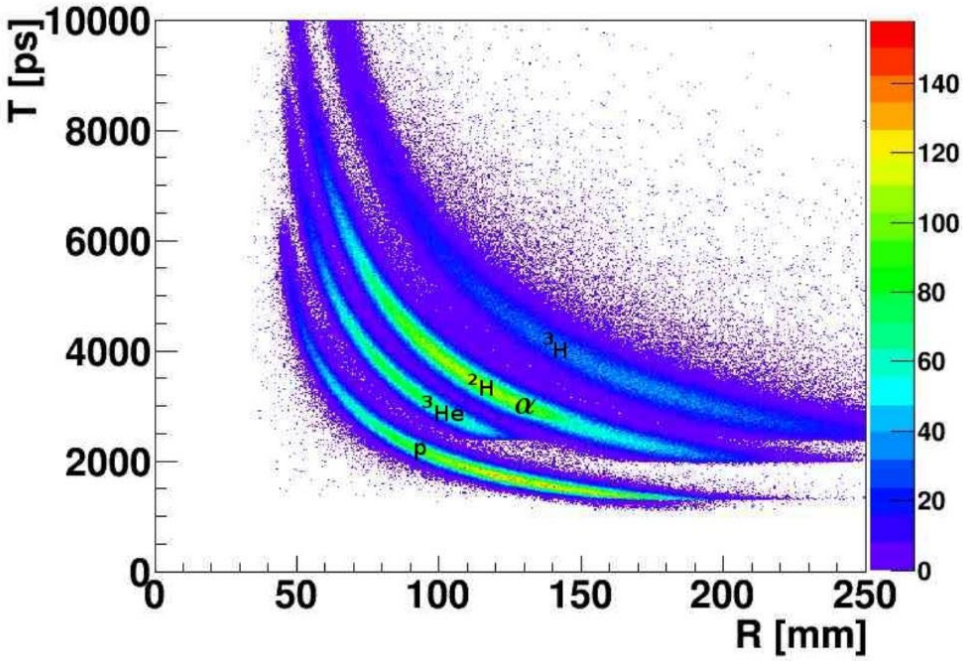
Outline

- Particle Identification in ALERT
- Two-Stage AI PID Framework
- Training Datasets
- Input Features
- Data vs Simulation Comparison
- Machine Learning Models
- Current Results
- Summary and Outlook

Particle Identification in ALERT

- ALERT detects recoil nuclear fragments:
  - ^{1}H, ^{2}H, ^{3}H
  - ^{3}He, ^{4}He
- Proposal assumptions:
  - ALERT Time of Flight (ATOF) resolves most species
  - ALERT Hyperbolic Drift Chamber (AHDC) resolves complementary combinations
- In practice:
  - Significant overlap between H and He isotopes
  - Traditional cut-based PID struggles in key kinematic regions





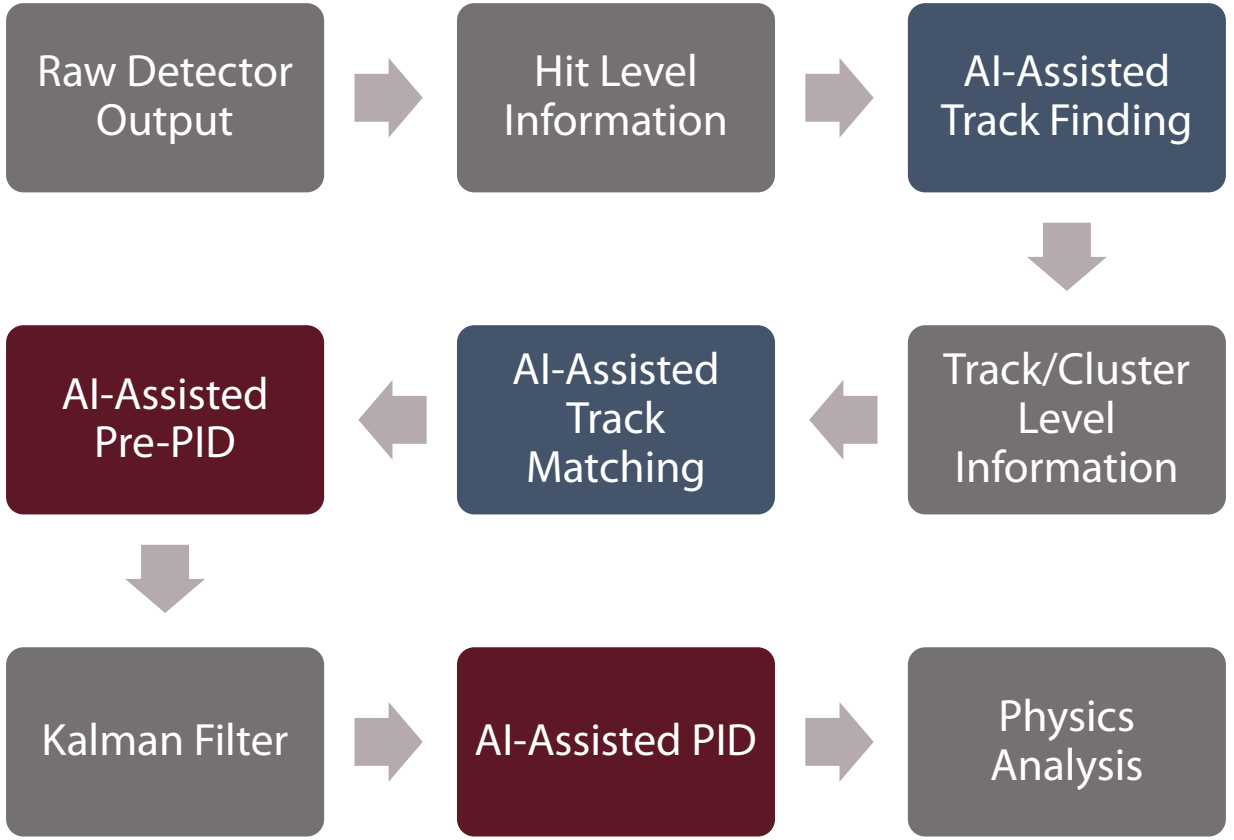
Two-Stage AI PID Framework

Pre-KF PID

- Uses AHDC track + matched ATOF hit when available
- Provides fast probabilistic classification
- Intended to assist Kalman Filter (KF)

Post-KF PID

- Uses KF-refined kinematics
- Final physics-level classification
- Improved accuracy



Training Datasets

- A training dataset of 100M single-particle tracks was prepared
- After running through reconstruction ≈25% of the tracks were reconstructed, with ≈10% containing matched ATOF hits
- For **Pre-KF PID**, all events with a reconstructed track (AHDC::track was present) were selected without considering if an ATOF hit was matched
- For **Post-KF PID**, only events with a Kalman-Filter reconstructed track (AHDC::kftrack was present) and a matching ATOF::hit (ALERT::ai:prediction having a matching pair) were selected
- The two datasets were then individually class-balanced across five particle species[^{1}H, ^{2}H, ^{3}H, ^{3}He, ^{4}He] and split into training, testing, and validation samples

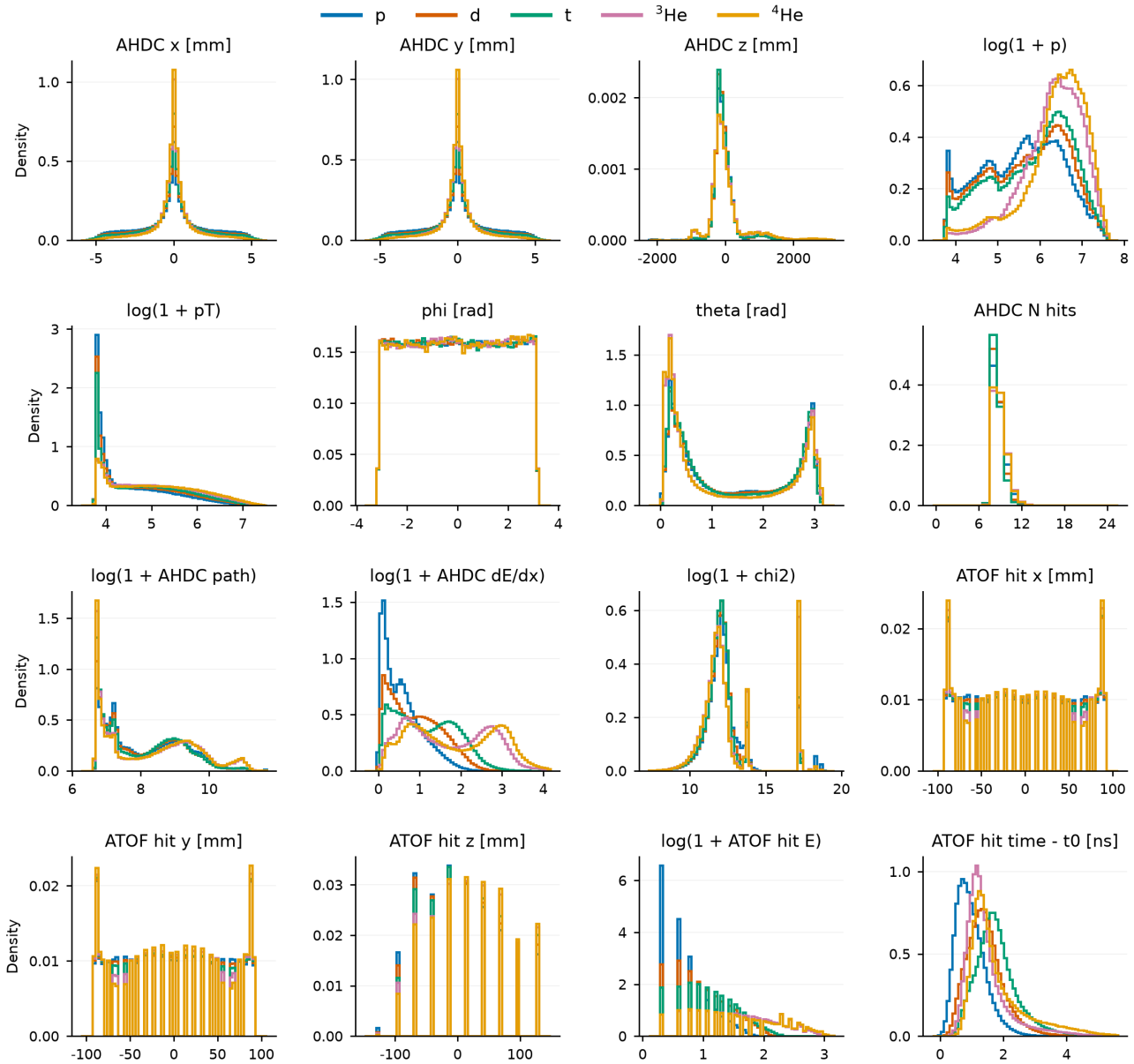
Input Features – Pre-KF PID model

AHDC::track (11 features)

- x, y, z
- p_x, p_y, p_z
- n_{hits}, ~~chi2~~
- sum(adc)
- dE/dx

ATOF::hits (5 features) [where available]

- time
- x, y, z
- ~~energy~~



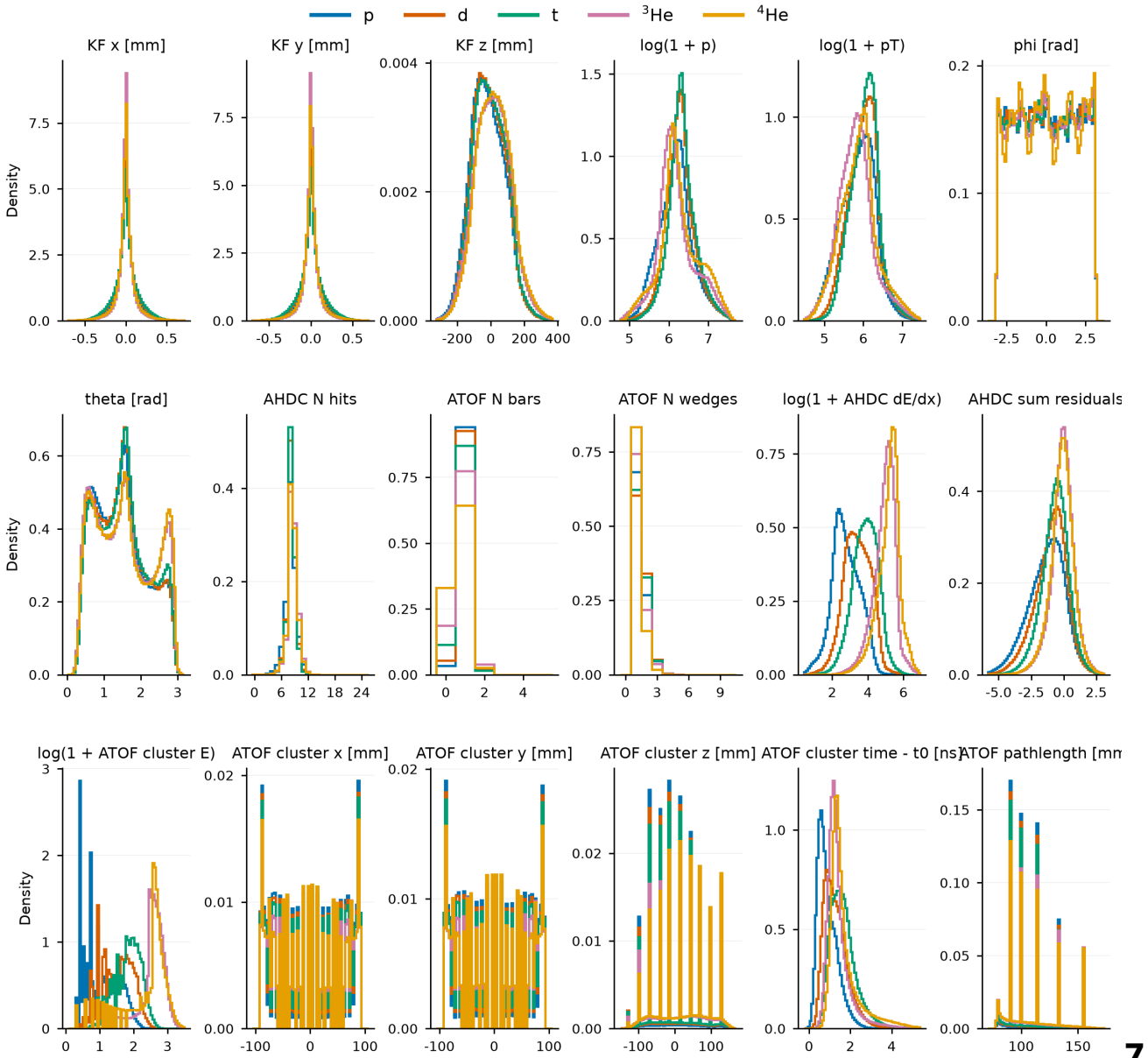
Input Features – Post-KF PID model

AHDC::kftrack (10 features)

- x, y, z
- p_x, p_y, p_z
- n_{hits}, dE/dx
- ~~p_{drift}~~

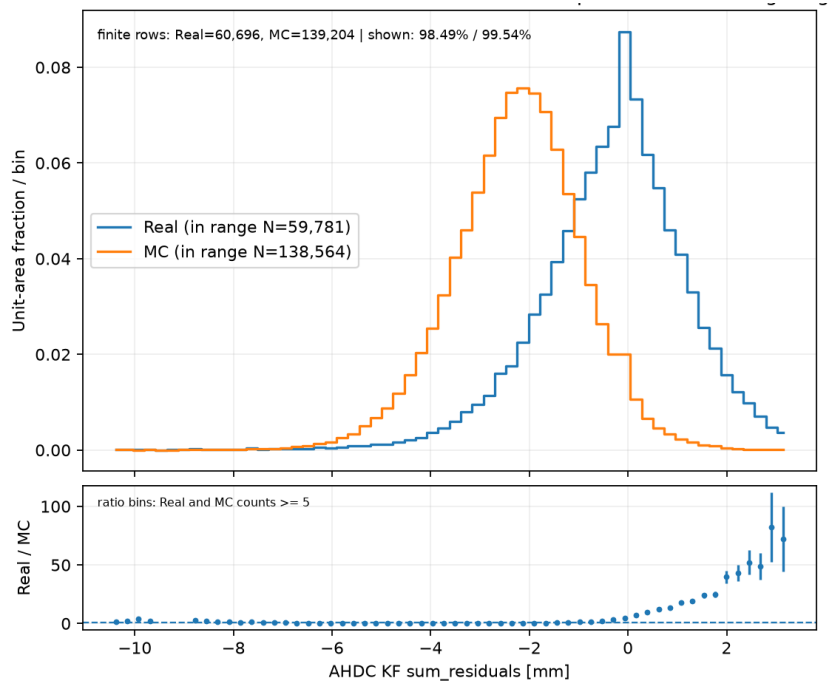
ATOF::clusters (8 features)

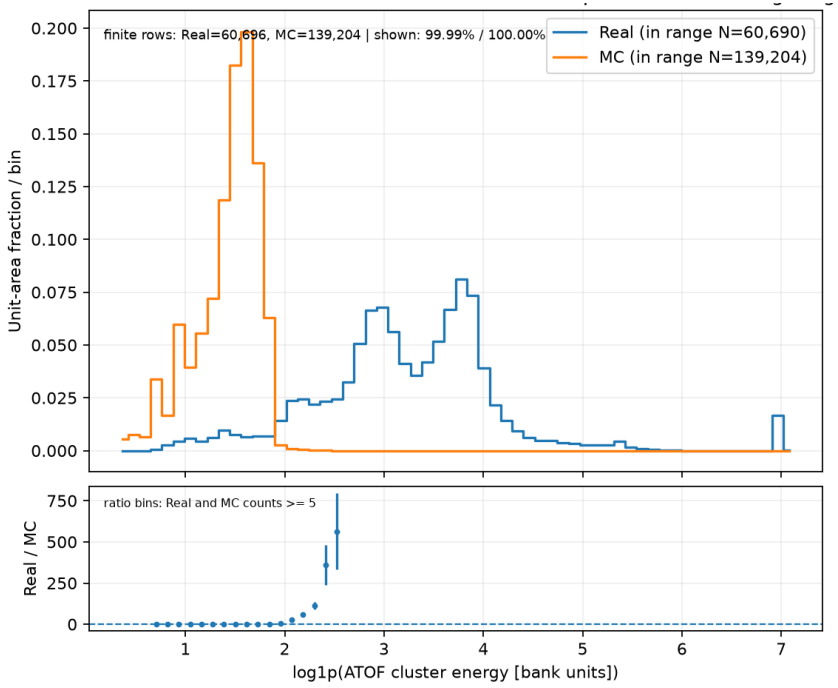
- n_bar, n_wedge
- time
- x, y, z
- ~~energy~~
- pathlength

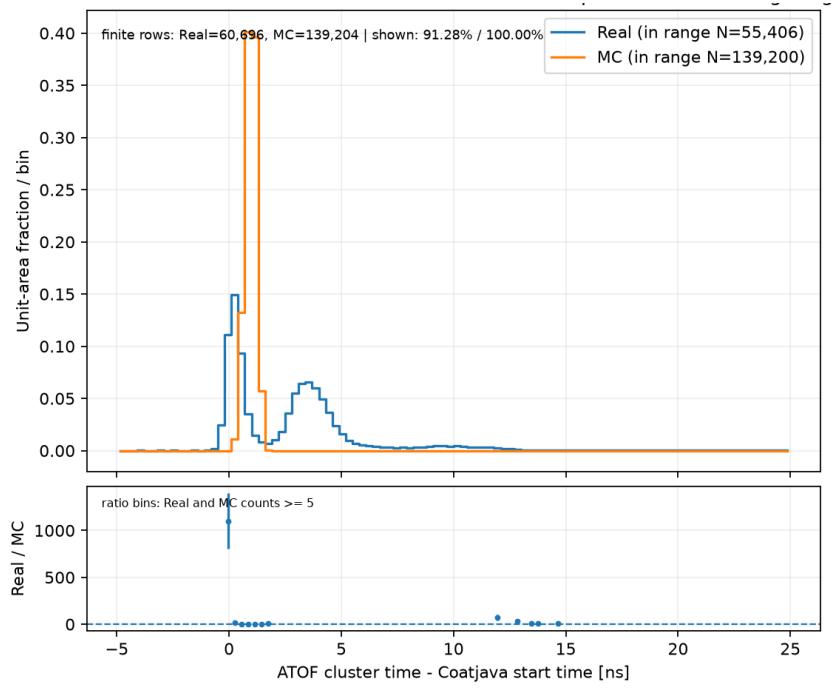


Data vs Simulation Comparison

- Most features agreed well between simulation and real data
- Three variables disagreed:
  - AHDC sum of residuals – removed
  - ATOF energy deposited – temporarily removed
  - ATOF time – added an offset to match real data







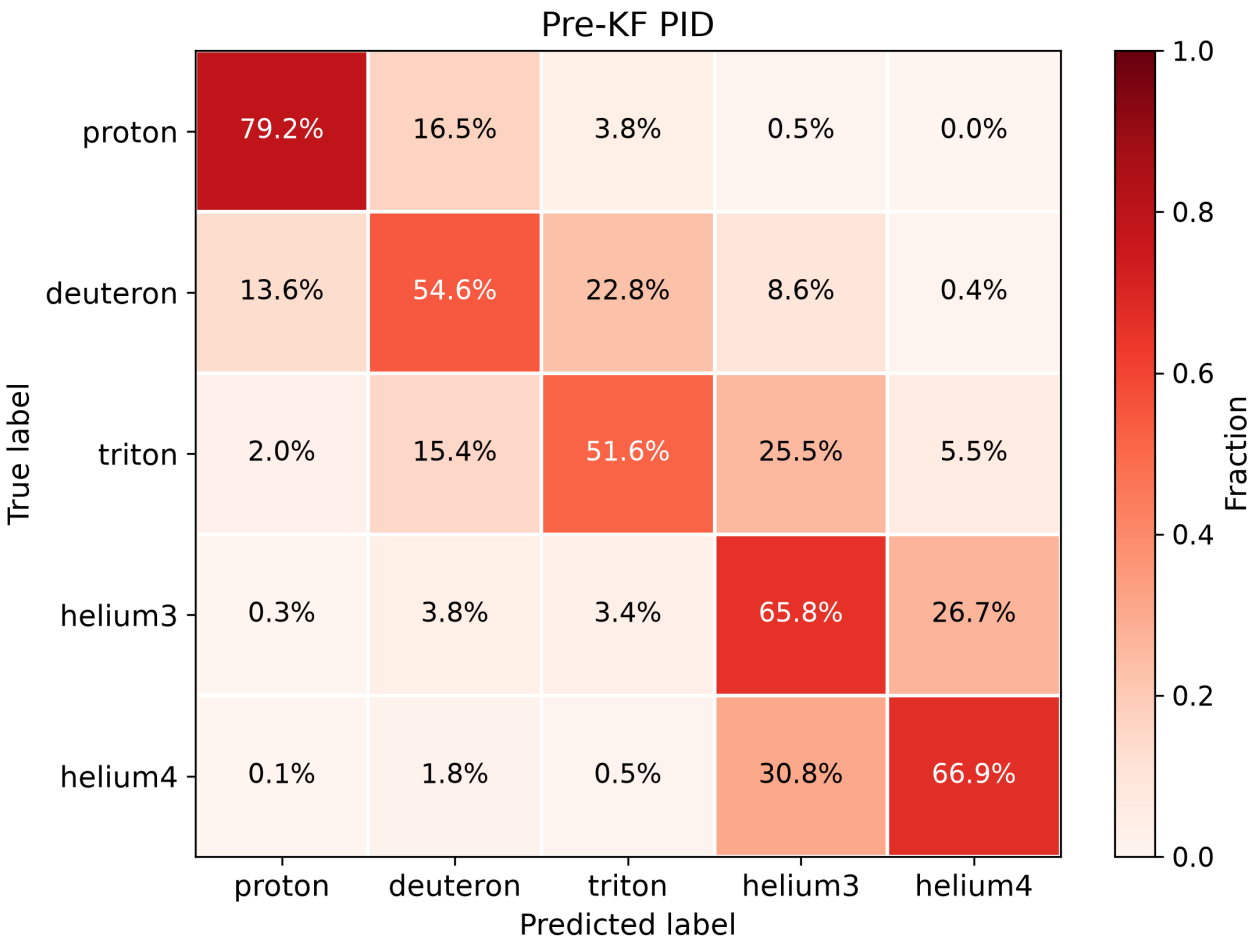
Machine Learning Models

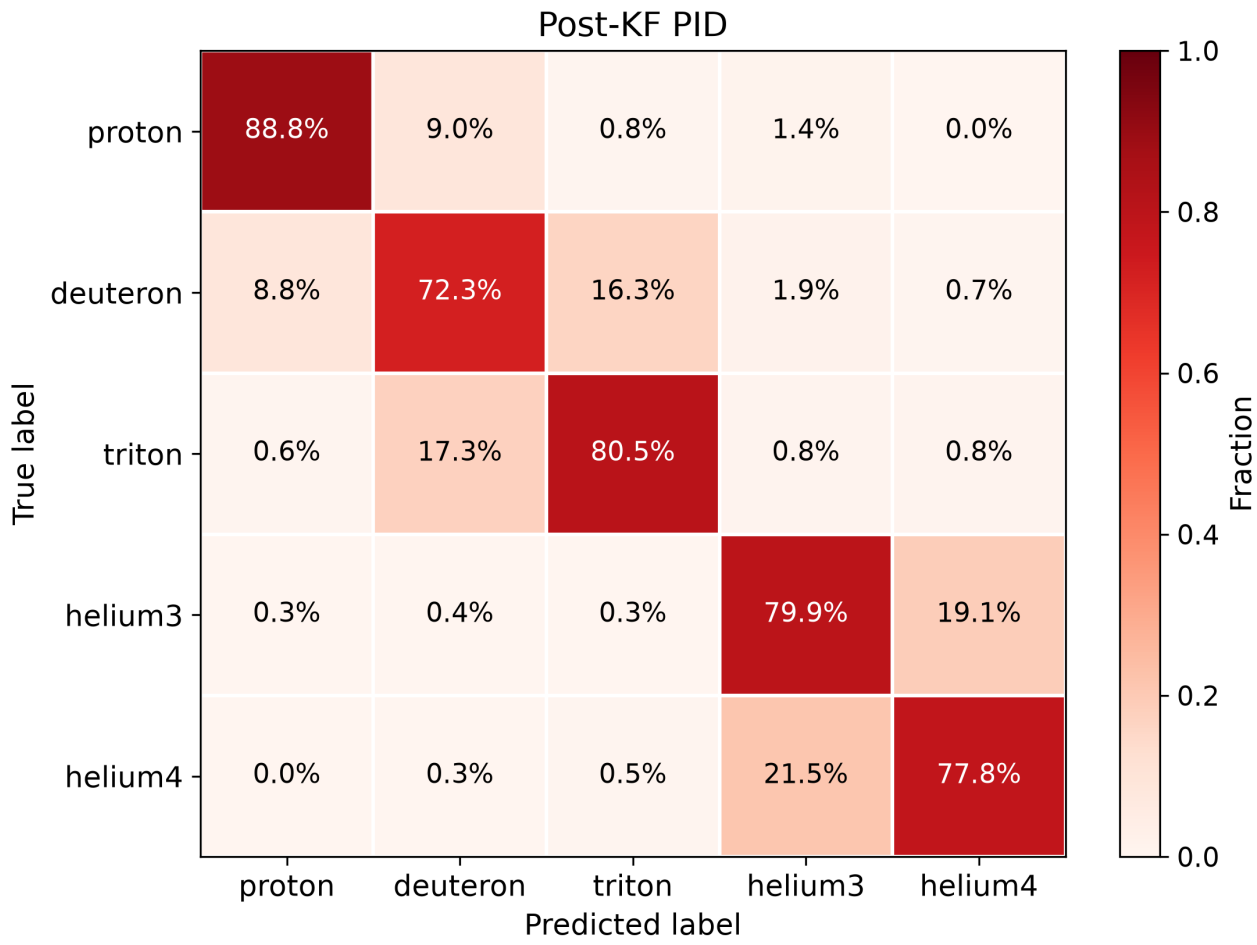
- After testing several models, two Multi-Layer Perceptron (MLP) models were used for pre- and post-KF PID
- Hyperparameter optimization with Optuna was used to tune each model fairly and identify the best-performing configuration for validation accuracy

Hyper Parameter	Pre-KF PID	Post-KF PID
No. of hidden layers	4	4
Hidden Layer sizes	[128, 32, 32, 128]	[129, 170, 189, 186]
Batch size	512	2048
Dropout	0.00959	0.00063
Learning Rate	0.0020	0.0016
Weight Decay	0.00021	0.00012

Current Results

- The currently trained models are able to reasonably predict PID for the two levels of reconstruction





Summary and Outlook

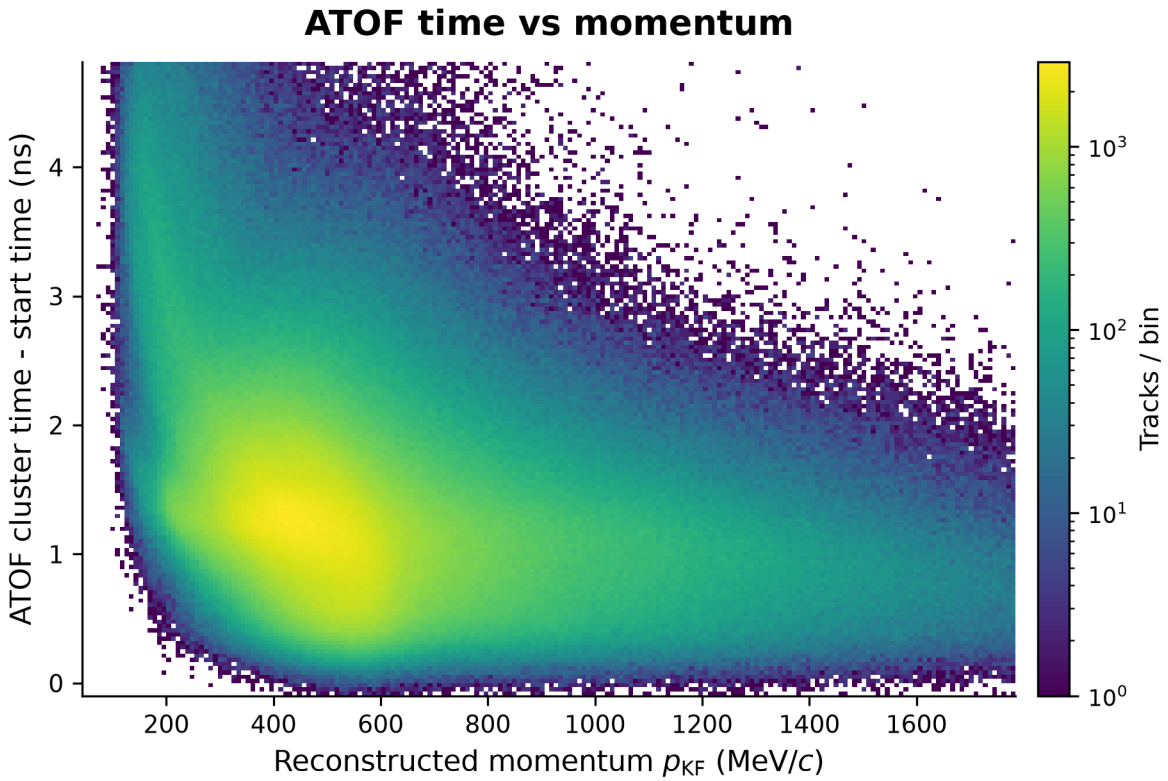
- Two-stage AI PID framework was developed to identify nuclear recoil fragments in ALERT
- The pre-PID model shows ≈70% accuracy, while the post-KF PID model has accuracy > 80%
- Both models are implemented in coatjava
- Real-data performance has room for improvement
- Future improvements are in development

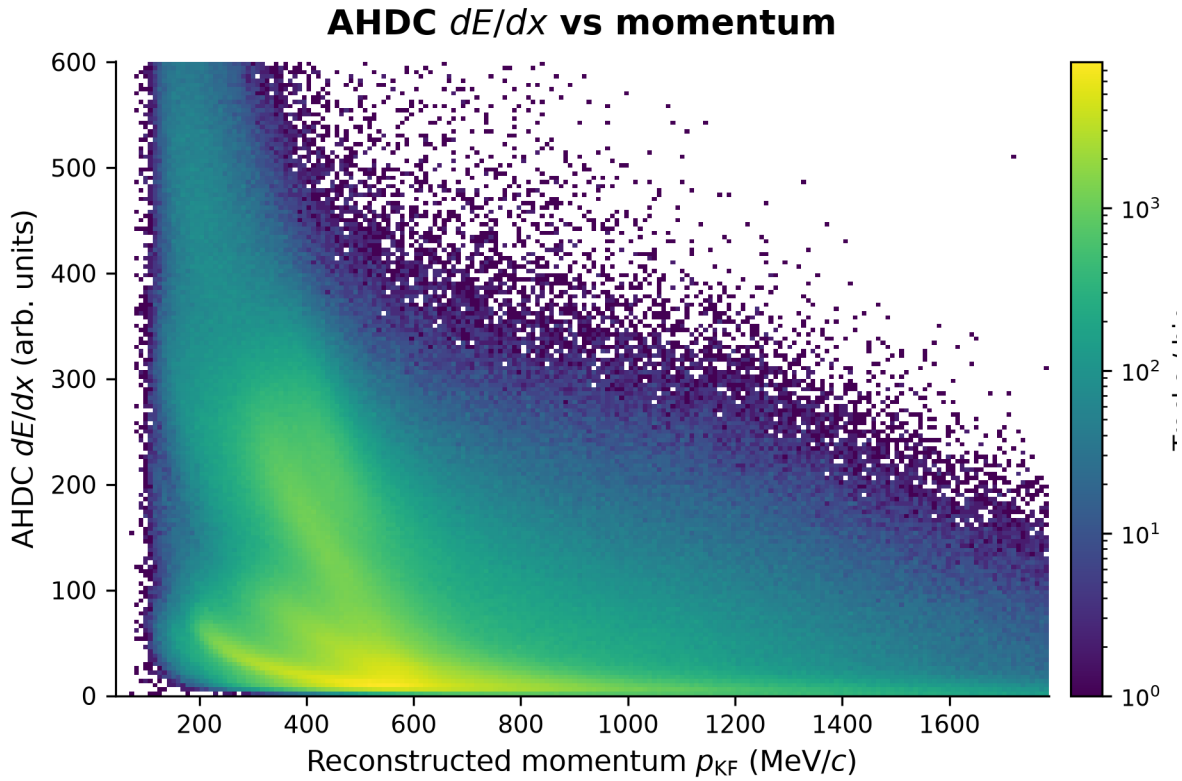
Thank You!

Backup Slides

Non-AI PID methods

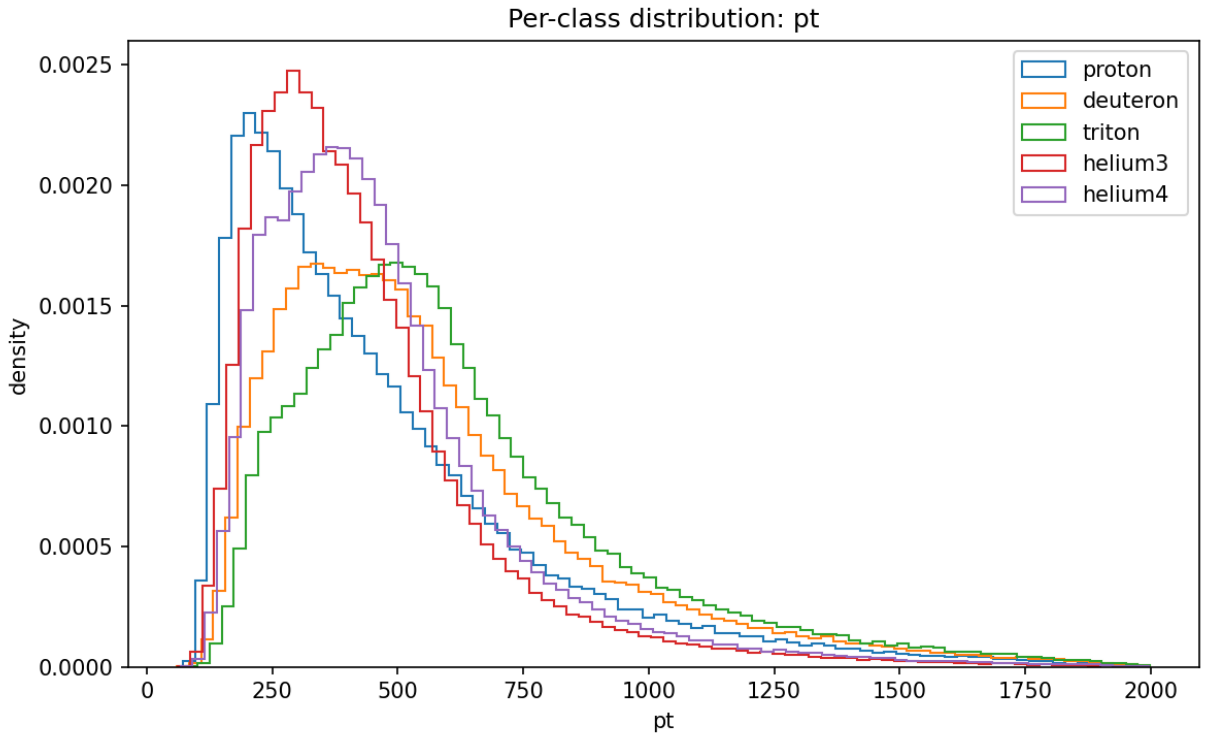
- Traditional PID methods are not currently implemented
- Current reconstructed resolution does not allow for traditional PID methods

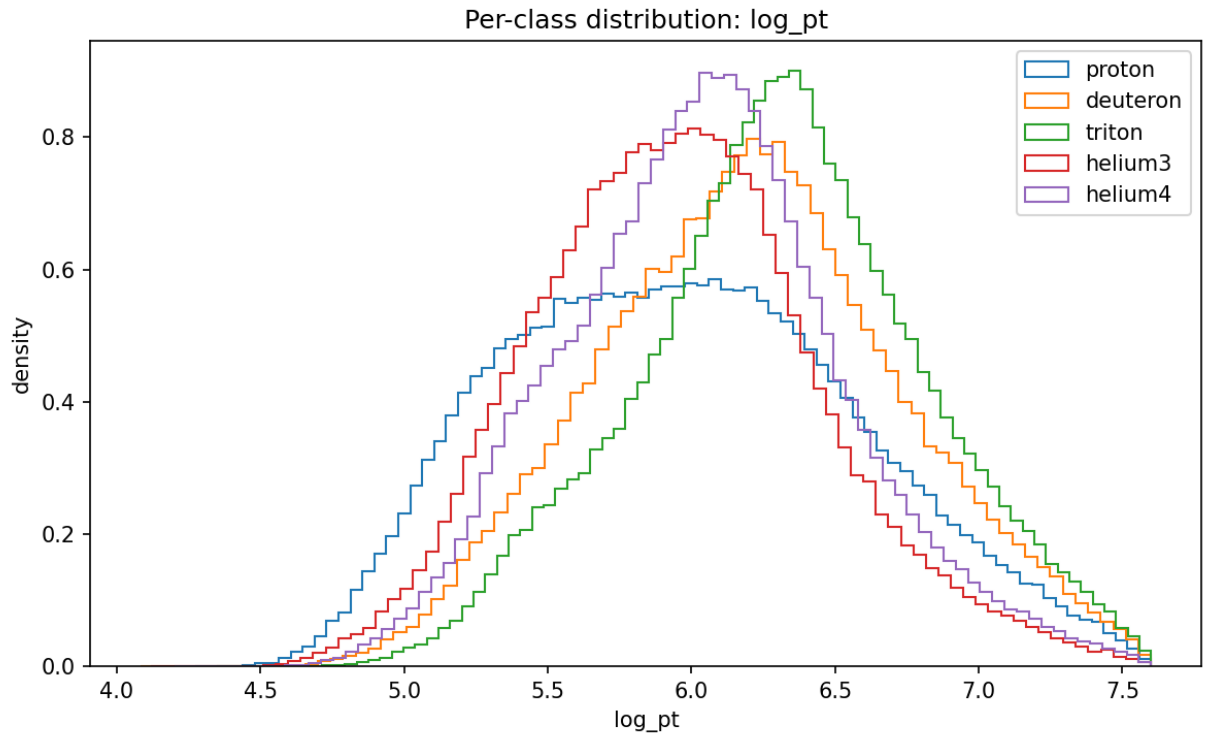




Feature Engineering

- Some features have long tails, which causes the normalization to be skewed.
- Easiest fix is to take the log scale





Real Data Validation

Methodology

- Reconstruct W for events with good electrons
- Select events with an `ALERT::ai:pid` prediction
- Compare W distributions for each predicted particle species

Observations & Interpretation

- Real-data performance is worse than simulation expectations, suggesting a data–simulation mismatch
- Pre-KF PID performs better than post-KF PID on real data
- ATOF-related features are a leading suspect, although differences in KF track inputs may also contribute

