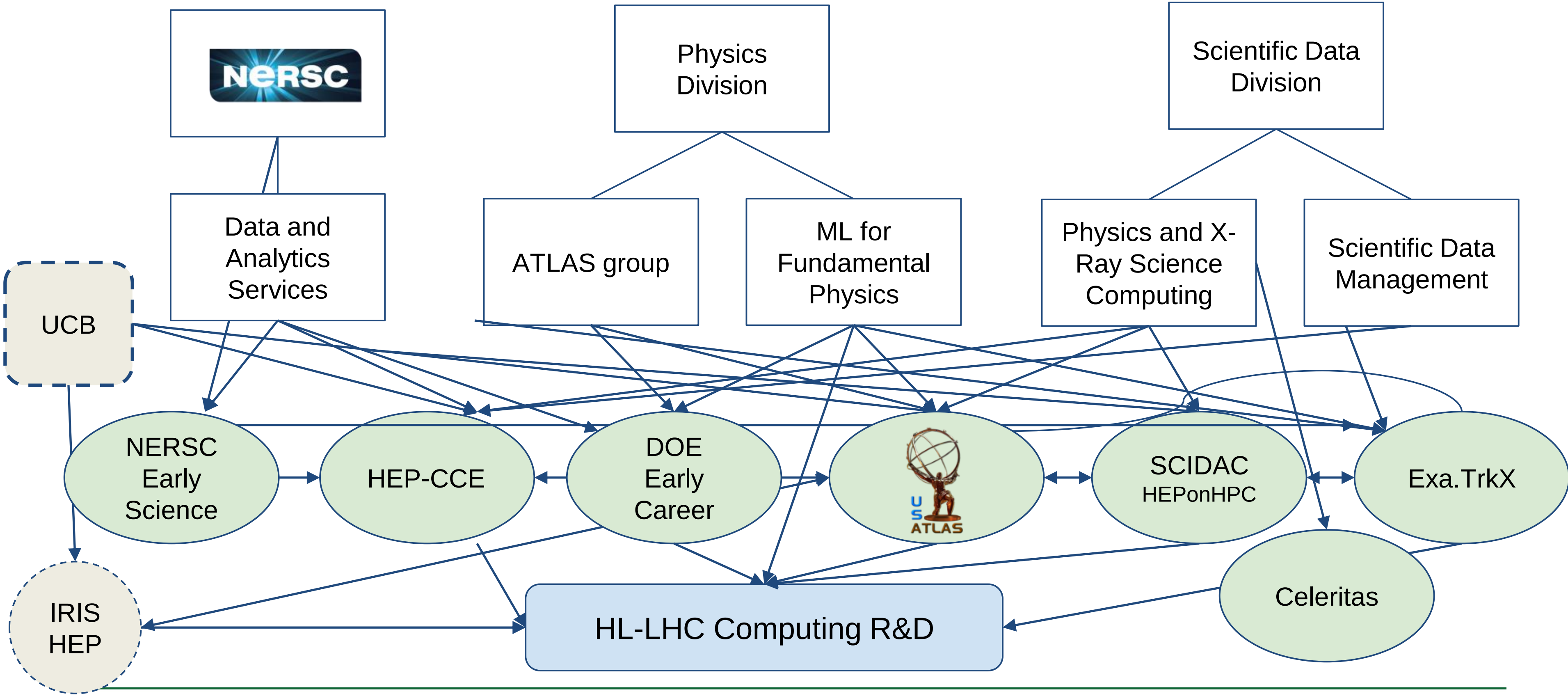
Computational HEP R&D @ LBL

Paolo Calafiura

IRIS-HEP Ecosystem Workshop, Nov 7 2022

HL-LHC Computing R&D at LBL



Vertically Integrated Scheduler for Heterogeneous Distributed Applications

Scientific Achievement

Develop a fine-grained HEP application scheduler that matches available resources to application requirements.

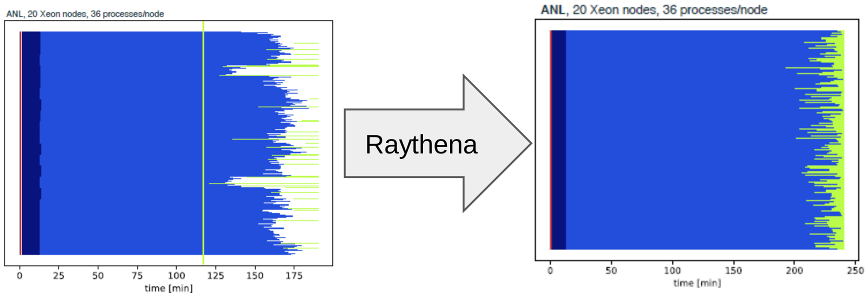
Significance and Impact

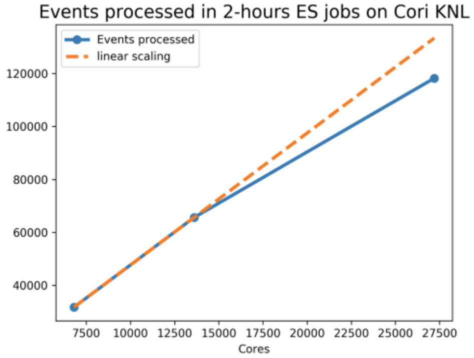
Integrating scheduling from system-level down to individual execution threads will increase efficiency of hybrid CPU/GPU applications on Exascale HPCs.

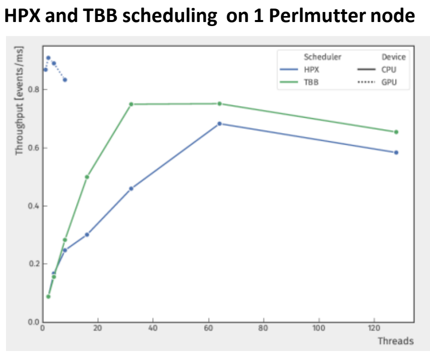
Research Details

- Phase 1 (“Raythena”): Use Ray to parallelize “HEP event loop” applications as a distributed, load-balanced task-farm:
  - **Good scaling up to 27500 cores on NERSC Cori**
- Phase 2: Use HPX and Ray to integrate multi-node parallelism with multi-threaded task scheduler.
- *Phase 3: Support hybrid applications on heterogeneous systems, matching tasks to resources.*

M. Muskinja, et al., EPJ Web of Conferences 245, 05042 (2020)
B. Stanislaus et al., ACAT 2022 presentation




Events processed in 2-hours ES jobs on Cori KNL


HPX and TBB scheduling on 1 Perlmutter node

***Top:** Worker execution state as function of time for a traditional HEP application (left) and for Raythena (right). Each worker state is represented by a different color: Worker initialization (dark blue), execution (blue), finalization (green), and inactive (white).*
***Bottom:** Left: weak-scaling throughput of Raythena task-farm. Right: Single-node MT performance using HPX and Intel TBB. GPU throughput ~constant as CUDA serializes thread execution.*

Enabling the GPU pipeline through Cling/cppyy

PHAX

Scientific Achievement

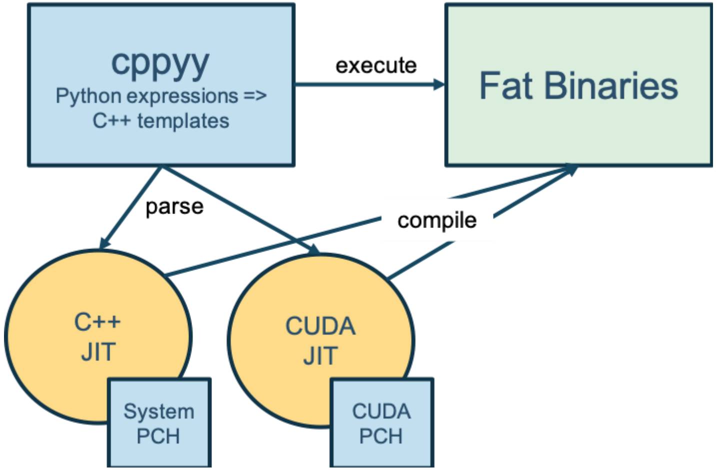
cppyy release 2.3.0 supports just-in-time compilation of CUDA code through Cling (an interactive C++ interpreter based on Clang) from Python.

Significance and Impact

Python and C++ are some of the most used languages in scientific research and GPUs are becoming the most prevalent compute resource in HPC. Access to CUDA from Python through a JIT allows run-time creation of kernels customized to the input data at hand.

Research Details

- C++ template specializations (e.g., from Eigen, Nvidia's CUTLASS, or MatX) can outperform their generic counterparts, but they require data types, sizes, and hardware capabilities to be known at compile time.
- Types and/or sizes depend on the program input data; and Python is fully run-time.
- With cppyy, one can take advantage of C++ templates (and the performance gains from their specialization by Python) at run-time by deferring instantiation to Cling's JIT, which is based on actual data types and sizes used.
- The new pipeline brings this functionality to GPU programming.



Overview of cppyy pipeline: *cppyy weaves together a C++ and a CUDA JIT (both from Cling), producing "fat binaries" with customized precompiled headers supplied to each for performance. The JITs can be programmed in tandem or independently, producing customized kernels from the templated libraries for CPU or GPU as desired.*

Credit: Wim Lavrijsen

Automated Optimization of HEP Event Generators

SCIDAC

Scientific Achievement

Automatically tune High-Energy Physics Event Generators (first-principles simulation of collision events) to experimental data

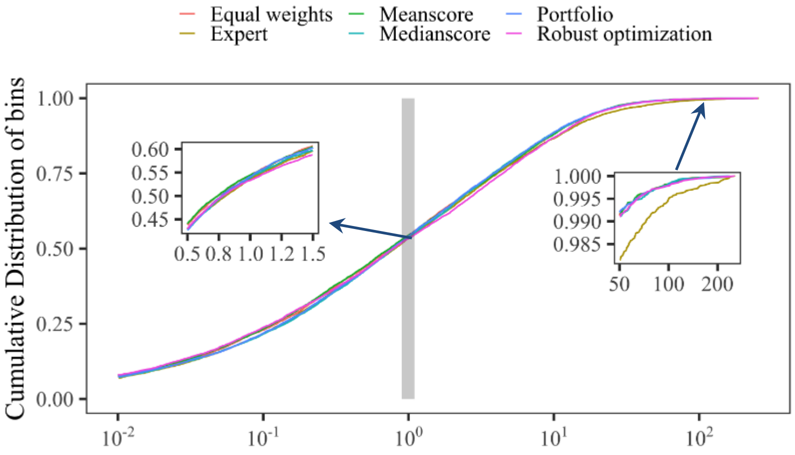
Significance and Impact

Leveraging HPC, auto-tuning tool can build a library of tuned event generators for all HEP measurements.

Research Details

- SciDAC4 HEPonHPC project applies advanced optimization methods and HPC workflows to automatically select and weight most relevant data for a given physics analysis.
- Event Generators auto-tuned by the tool described measured data as well as (sometimes better than) those hand-tuned by experts.
- Rational approximation, better in modeling non-linear effects, replaces polynomial approximation to model the relationship between generator parameters and observables.
- Two advanced optimization methods are used to automatically weight measured data: Robust optimization and bi-level optimization.
- Both optimization techniques yield generator parameters that make generated data agree better with the measured data.

Wenjing Wang, et al, arXiv:2103.05751



χ2, measuring the agreement between the generator and experimental measurements

This plot shows the fraction of number of observables that are with χ^2 smaller than different thresholds. Over 99.0% of observables from the generator auto-tuned with advanced optimization are with $\chi^2 < 50$, while only 98.5% from the expert-tuned generator. The labels, "Meanscore," "Medianscore," and "Portfolio" refer to the method used in the innermost optimization level. "Expert" refers to state-of-the-art hand-tuning of the generator.

Credit: Xiangyang Ju, Wenjing Wang, Juli Mueller

Generator tuning with MC systematic uncertainties

Scientific Achievement

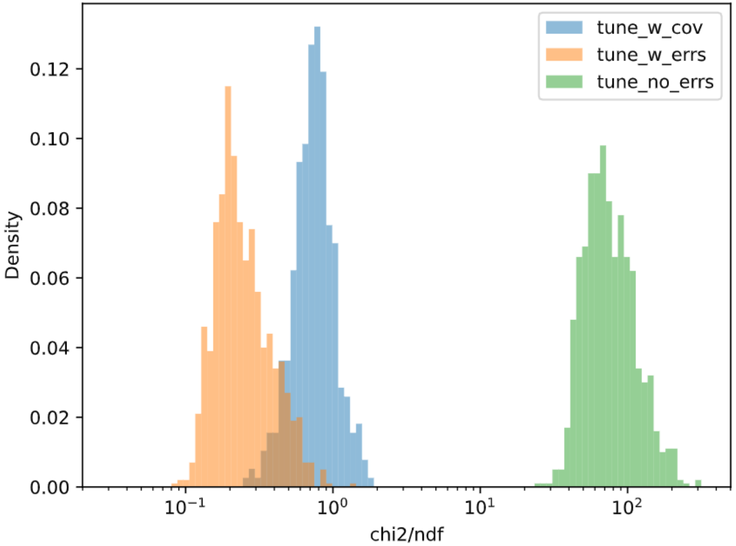
Examined the impact of Monte Carlo (MC) systematic uncertainties on MC Event Generator tuning.

Significance and Impact

Our treatment of MC systematic errors leads to more accurate estimates of the generator parameters uncertainties

Research Details

- Builds on "Automated Optimization of HEP Event Generators" project
- For the first time we evaluate the impact MC systematic uncertainties on the tuned parameters in a principled way, rather than through educated guesses.
- Preliminary studies on toy data show that the χ2/NDF distribution shows our setup produces a realistic estimation of the uncertainties.
- Next, we will apply this novel method to real data and tune generator parameters for the LHC experiments



χ2/NDF distribution for different MC generator tunings.

Our method, tuning with covariance matrix (i.e. including MC uncertainty), yields the χ2/NDF distribution closer to 1 as predicted by Wilks' theorem. Therefore, one can easily obtain 68% confidence interval. NDF is the number of degree of freedom.

Differentiable simulation of hadronic interactions

SCIDAC

Scientific Achievement

Prototyped deep generative models to simulate hadronic interactions

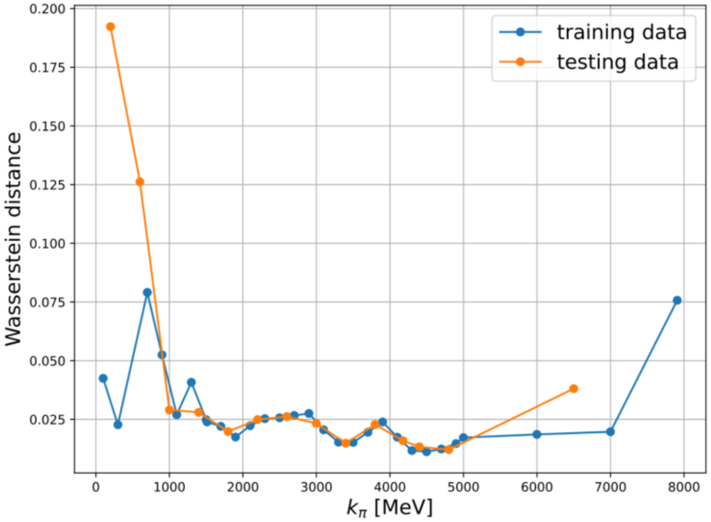
Significance and Impact

Our generative models could replace ad-hoc hadronic simulation modules in Geant4.

A step towards differentiable detector simulation.

Research Details

- Simulating hadronic interactions is the essential component in simulation hadronic calorimeters at the LHC
- Current hadronic simulation in Geant4 uses a group of parametric models, each simulating a specific range of hadron energies for specific hadron flavor types
- We developed Generative Adversarial Network (GAN) and Normalizing Flow (NF) models to simulate hadronic interactions
- We find NF performs better than GAN for low energy regions
- We simulated pions on a hydrogen target; will expand to a wider range of interactions
- After that we will integrate the model into Geant4 and compare its performance with G4 simulation



Differences between Normalizing Flow-generated events and Geant4 simulated events for different pion energies.

Wasserstein distance is the figure of merit. The smaller the distance, the better the agreement is. Note that the testing was done with a pion energy not used in the training. Better results are obtained for pion energies above 1 GeV.

A Fair Universe: Unbiased Data Benchmark Ecosystem for Physics

AI 4 HEP

Scientific Goals

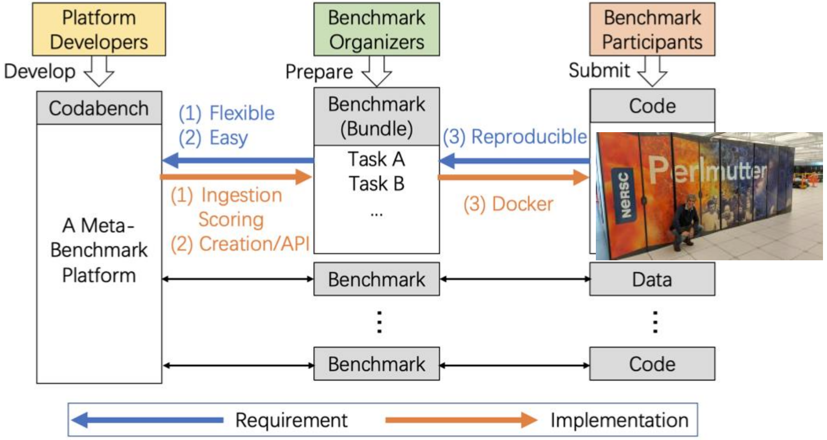
Provide a **large-compute-scale AI ecosystem** for sharing datasets, training large models, fine-tuning those models, and hosting challenges and benchmarks.
Host challenges and benchmarks focussed on discovering and minimizing the effects of systematic uncertainties.

Significance and Impact

Provide a platform for large-scale AI experimentation and development of systematic-uncertainty-aware AI models.

Research Details

- Recently funded three year comp-hep project
- Constructing datasets and tasks for challenged and *long-lived* benchmarks systematic uncertainty aware AI techniques in particle physics and cosmology
- Building HPC-enabled AI benchmark platform to host new models and be able to leverage NERSC resources to apply new AI algorithms on existing and new datasets



Overview of the core proposed platform (based on Codabench). Codabench is designed to support diverse benchmarks. Each benchmark is implemented by a benchmark bundle that contains one or more tasks (wrapping around datasets). This project will exploit and extend Codabench's new features; interface it to NERSC HPC capabilities and tackle the problem of systematics in physics from various angles

U.S. DEPARTMENT OF ENERGY | Office of Science
NERSC
CHA LEARN
UNIVERSITY of WASHINGTON
BERKELEY LAB

Unbinned Unfolding of Detector Effects

Scientific Goals

Introduced an unfolding method that uses machine learning to capitalize on all available information.

Significance and Impact

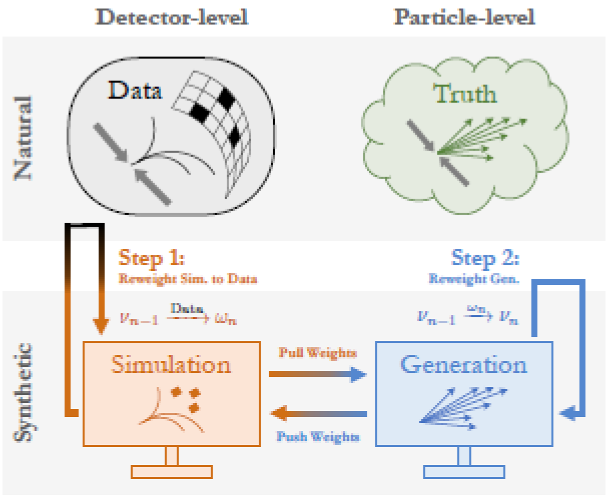
Reduce errors and remove potential biases on collider data that must be corrected for detector effects (“unfolded”).

Enable future public and archival collider data analyses

Research Details

- Traditional unfolding methods correct detector effects O(1) observable at a time and use discretized distributions (histograms)
- **OmniFold** iteratively unfolds an entire dataset using all of the available information. Works for arbitrarily high-dimensional data, and naturally incorporates information from the full phase space.
- New observables can be measured long after the unfolding is carried out
- OmniFold requires significant GPU resources and new ways to publish and share code and datasets

→ Fair Universe Project



OmniFold reweights synthetic detector-level events (“Simulation") to match experimental data (“Data"). The reweighted synthetic events, now evaluated at particle-level (“Generation"), are further reweighted to estimate the true particle-level information (“Truth").

PhysRevLett.124.182001

Anomaly Detection for New Physics

ML Physics

Scientific Goals

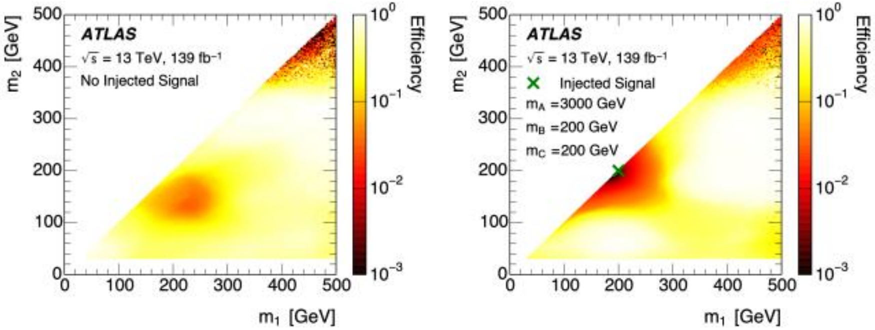
Develop a fully data-driven machine-learning-enhanced anomaly detection methodology.

Significance and Impact

Enhance the sensitivity to a wide variety of hypothetical particles without specifying all of their properties ahead of time.

Research Details

- Developed **CWoLa**, a weakly supervised NN classifier trained on unlabeled data samples
- Combined with a bump-hunt anomaly detection for dijet resonance searches with increased sensitivity.
- This workflow is computationally. The prototype analysis (see right) required training 20k NNs. Higher dimensional analysis will require comparable networks and will need GPU (see Perlmutter).
- DOE Early Career project, in collaboration with NERSC, focused (among other things) on scaling this workflow



The neural network output in one dijet mass bin. As a two-dimensional function, the output can be readily visualised as an image, where the intensity corresponds to the efficiency of the network output in the dijet mass bin. The left plot has no signal injected and the right plot shows the output when a hypothetical particle at 3 TeV that decays into two other particles at 200 GeV is added to the data.

PhysRevLett.125.131801

Point Cloud Deep Learning Methods for Pion Reconstruction in the ATLAS Experiment

Scientific Goals

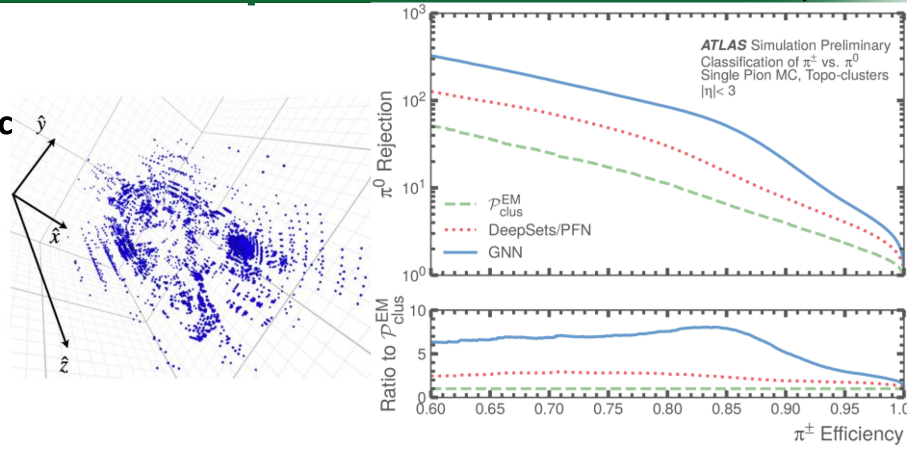
ML4Pions studies a variety of machine learning methods designed for the reconstruction and calibration of hadronic final states

Significance and Impact

Outperformed significantly ATLAS baseline pion classification and reconstruction methods.

Research Details

- Developed ML models for π 0 vs. π ± classification and pion energy regression.
- Used information from both calorimeter clusters and, in the case of energy regression, particle tracks.
- Transformer, Deep Sets, and Graph Neural Network architectures are used to process calorimeter clusters and particle tracks as point clouds, or a collection of data points representing a three-dimensional object in space.
- Complementary and synergistic to the *Exa.TrkX* approach to particle tracking



Left: *A dijet collision event rendered as a 3-dimensional point cloud of calorimeter cells, as seen from two orientations.*

Right: *Comparison of topo-cluster classification performance of all methods for |η| < 3. Performance is measured as π0 topo-cluster rejection (defined as the inverse of π0 selection efficiency) versus π± topo-cluster efficiency, where higher rejection indicates better classification performance for the same selection efficiency. The baseline method performance is shown by the green dashed line.*

ATL-PHYS-PUB-2022-040

In Summary

- **Long-term collaboration across three divisions (NERSC, SciData, Physics)**
  - Always seeking participation from other labs and universities
- **Our R&D program builds on decades of M&O experience in BaBar, CDF/D0, ATLAS, Daya Bay, LZ, ROOT, etc.**
  - Trying to anchor our projects to real needs experiments
- **Contributing to many of the HL-LHC hot-button topics**
  - Distributed heterogeneous computing at scale
  - New pattern recognition algorithms for tracks and jets
  - New analysis methods
  - Core software infrastructure, including for ML workflows

Thanks

Wahid Bhimji

Julien Esseiva

Wim Lavrijsen

Xiangyang Ju

Ben Nachman

Beojan Stanislaus

Vakho Tsulaia

…

All omissions and misrepresentations are mine only

Exa TrkX

The Exa.TrkX Project

Paolo Calafiura for the Exa.TrkX project

IRIS-HEP Ecosystem Workshop, Nov 7 2022

U.S. DEPARTMENT OF ENERGY | Office of Science

Caltech Fermilab BERKELEY LAB

Geometric Deep Learning for HEP Particle Tracking

Scientific Achievement

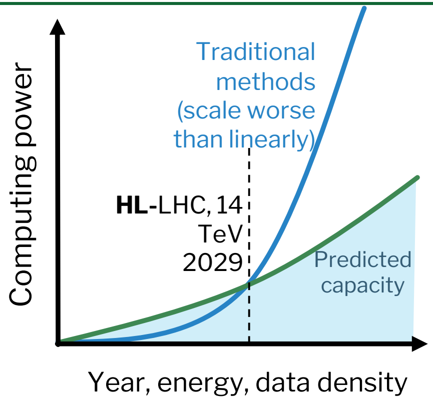
Developed a deep learning pipeline to measure particle trajectories in High-Energy Physics (HEP) detectors.

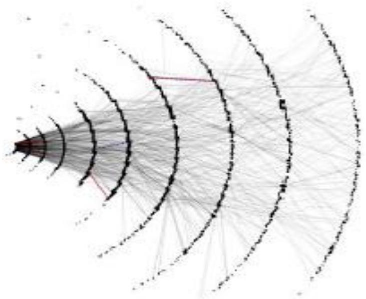
Significance and Impact

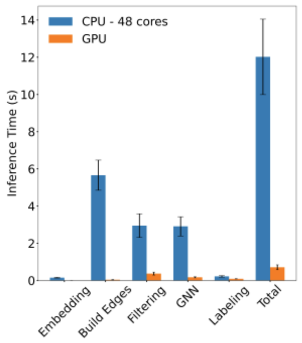
Unlike traditional algorithms, this pipeline scales linearly with data density which could increase the discovery potential of future HEP experiments.

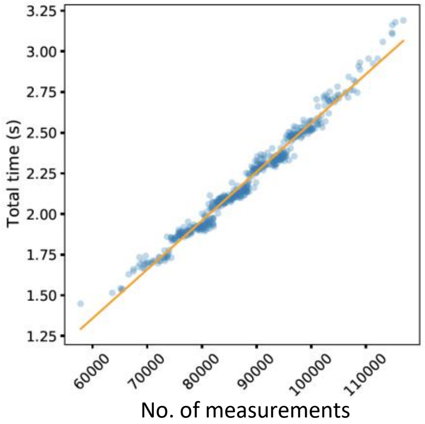
Research Details

- As particle accelerators become more powerful, detectors become more complex with increasingly dense measurements: O(10M) particles/second, O(100M) measurements/second.
- Traditional algorithms generate all possible trajectories. Attaching measurements-as-they-go results in a combinatorial explosion.
- The DOE Exa.TrkX project pioneered the application of Geometric Deep Learning methods, specifically graph neural networks, to capture and regularize relationships between measurements.
- Clustering with the learned metric is crucial to control graph density.
- Our optimized pipeline runs end-to-end on the NVIDIA V100 GPU with a 20X speed-up wrto using a 48-core Xeon 8268s Cascade Lake CPU.









Top left: *HEP computational challenge.* ***Top right****: graph network output for .125% of the detector (misclassified edges in red).* ***Bottom left****: inference time on GPU vs CPU.* ***Bottom right****: inference time vs detector density*

X. Ju, D. Murnane, P. Calafiura, et al — Eur. Phys. J. C 81, 876 (2021)
A. Lazar, X. Ju, D. Murnane, et al — arXiv:2202.06929

U.S. DEPARTMENT OF ENERGY | Office of Science
Caltech Fermilab BERKELEY LAB

An Open Collaboration

ML4Pions

acts

TrackML

NuML

NVIDIA

Hewlett Packard Labs

NERSC

a3d3 Accelerated AI Algorithms for Data-Driven Discovery

Princeton

SPRACE

L2IT

UChicago

Krakow

Cincinnati

Youngstown

Weizmann

CalTech, FNAL, LBNL

Tsing Hua

UUIC

SLAC

UWashington

UW Madison

Northwestern

Uppsala

ECP EXASCALE COMPUTING PROJECT

ExaLearn

FastML Lab

panda

ATLAS EXPERIMENT

CMS

Fermilab

LArTPC

MUonE

μ g-2

An Open Collaboration

Exa.TrkX

Possible collaboration

TrackM

prerequisit

eNuML

a(ts

NVIDIA

Use case

Hewlett Packard Labs

NERSC

Know-how Resources

a3d3 Accelerated AI Algorithms for Data-Driven Discovery

Fully integrated

FPGA optimization s

FastML Lab

Princeton

L2IT

UChicago

Youngstown

Cincinnati

Krakow

Weizmann

CalTech, FNAL, LBNL

Tsing Hua

UUIC

SLAC

UWashington

Northwestern

UW Madison

Uppsala

ECP EXASCALE COMPUTING PROJECT

ExaLearn

panda

Published study

ATLAS EXPERIMENT

Part of release. Offline and TDAQ studies

CMS

Related projects

Fermilab

(proto)DUNE studies

Exploring

μ g-2

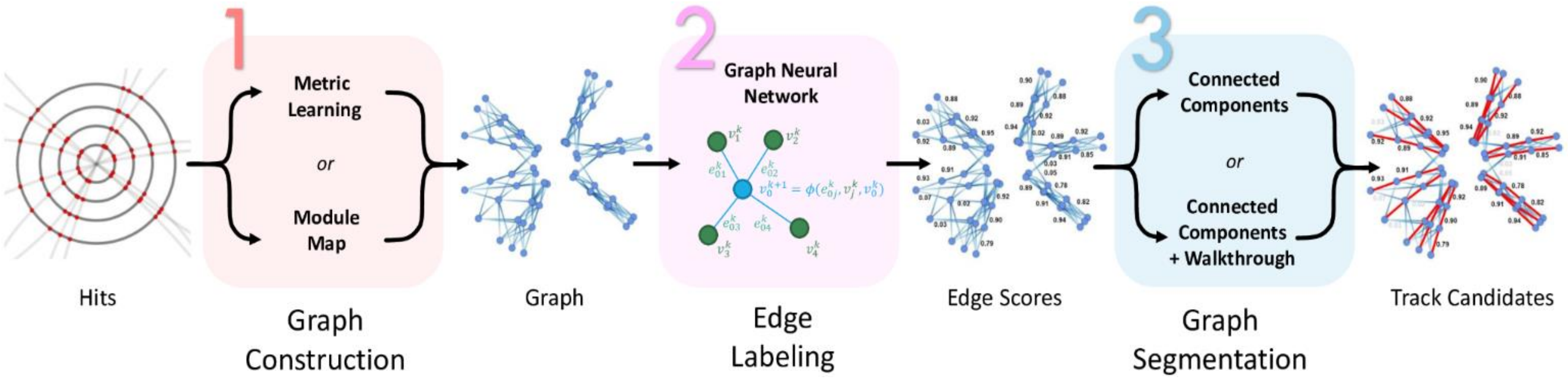
ENERGY Office of Science

Caltech

Fermilab

BERKELEY LAB

A Flexible Pipeline of Composable Modules



Hits

1 Metric Learning *or* Module Map

Graph Construction

Graph

2 Graph Neural Network

Edge Labeling

Edge Scores

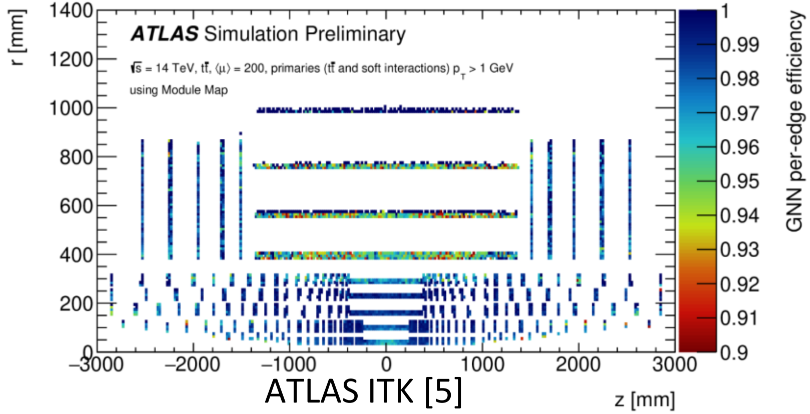
3 Connected Components *or* Connected Components + Walkthrough

Graph Segmentation

Track Candidates

PyTorch geometric

Applicable to Multiple Pattern Recognition Problems



ATLAS ITK [5]

PANDA Straw Tubes [4]

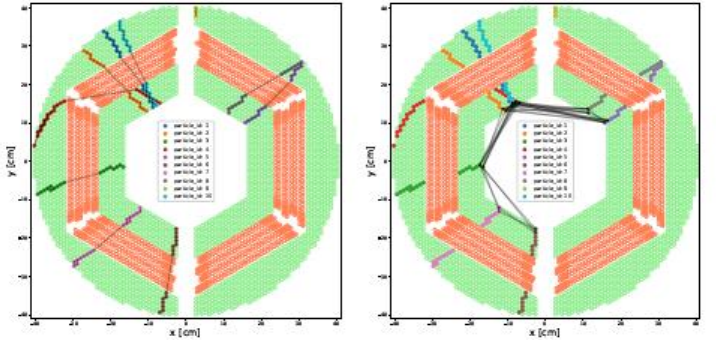
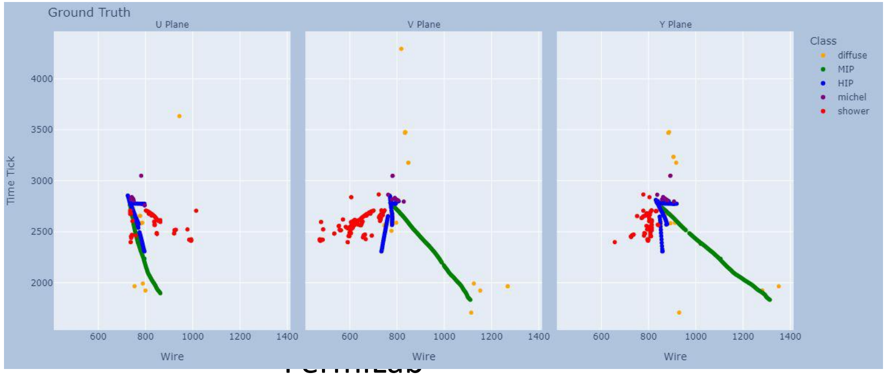
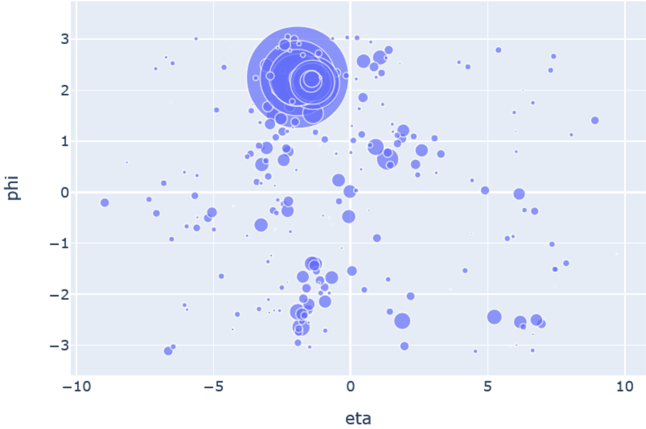


Figure 4: Graph representation of an event: *(left)* True Graph, *(right)* Input Graph

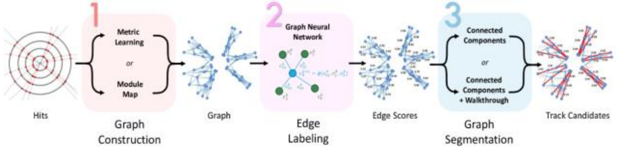


Fermilab LArTPC [8]



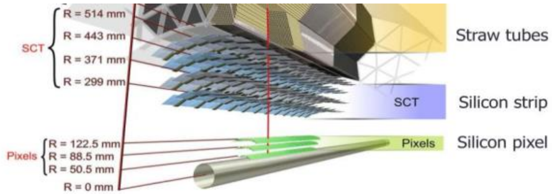
Jet Clustering
2008.06064

Towards a Tracking Supermodel

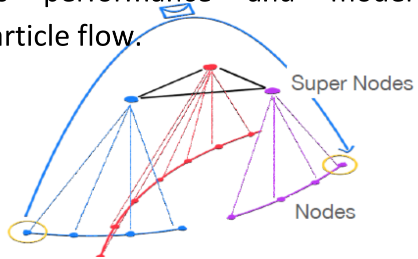


Re-engineered pipeline to improve modularity and usability across experiments

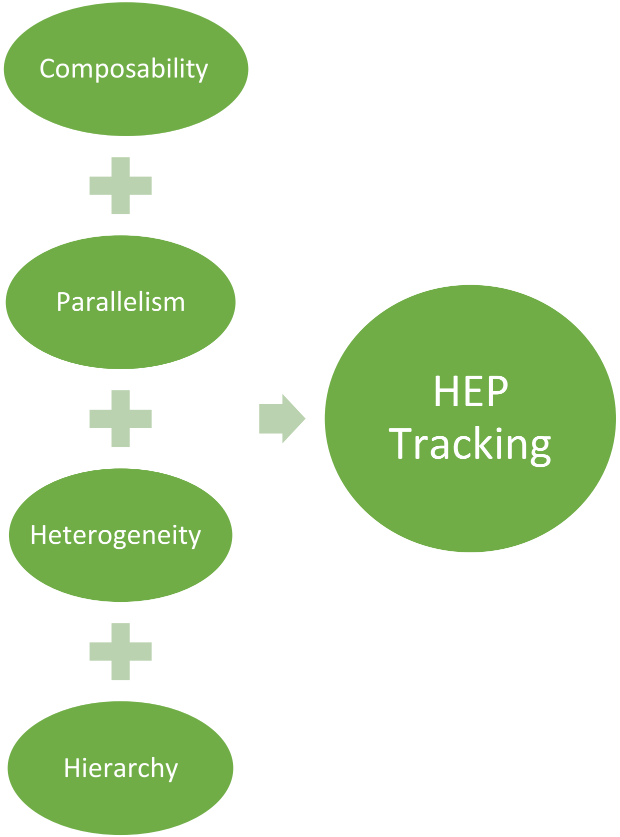
Memory limits our GNN performance. Running on multiple GPU would open new possibilities. We are investigating **graph data parallelism**, an [active research area](). Our graphs change structure with every event, making the problem even more challenging.



Investigating **heterogeneous GNNs** that combine information from multiple detectors [9]. Should improve physics performance and model generalization. Particle flow.



We can recover "difficult" tracks (e.g., tracks with a missing spacepoint) using **hierarchical GNNs** [10]. Next, will scale these models up to full HL-LHC simulations. Again, potential for multi-scale pattern recognition.



Composability + Parallelism + Heterogeneity + Hierarchy → HEP Tracking

Exa.TrkX References

[1]Farrell, S., Calafiura, P., et al. . Novel deep learning methods for track reconstruction. (2018). *arXiv*. https://doi.org/10.48550/arXiv.1810.06111

[2] Ju, X., Murnane, D., et al. Performance of a geometric deep learning pipeline for HL-LHC particle tracking. Eur. Phys. J. C 81, 876 (2021). https://doi.org/10.1140/epjc/s10052-021-09675-8

[3] Hewes, J., Aurisano, A., et al. Graph Neural Network for Object Reconstruction in Liquid Argon Time Projection Chambers. EPJ Web of Conferences 251, 03054 (2021). https://doi.org/10.1051/epjconf/202125103054

[4] Akram, A., & Ju, X. Track Reconstruction using Geometric Deep Learning in the Straw Tube Tracker (STT) at the PANDA Experiment. (2022) arXiv. https://doi.org/10.48550/arXiv.2208.12178

[5] Caillou, S., Calafiura, P. et al. ATLAS ITk Track Reconstruction with a GNN-based pipeline. (2022). ATL-ITK-PROC-2022-006. https://cds.cern.ch/record/2815578

[6] Lazar, A., Ju, X., et al. Accelerating the Inference of the Exa.TrkX Pipeline. (2022). *arXiv*. https://doi.org/10.48550/arXiv.2202.06929

[7] Wang, C., et al. Reconstruction of Large Radius Tracks with the Exa.TrkX pipeline. (2022). *arXiv*. https://doi.org/10.48550/arXiv.2203.08800

[8] Gumpula, K., et al., Graph Neural Network for Three Dimensional Object Reconstruction in Liquid Argon Time Projection Chambers. (2022) https://indico.cern.ch/event/1103637/contributions/4821839

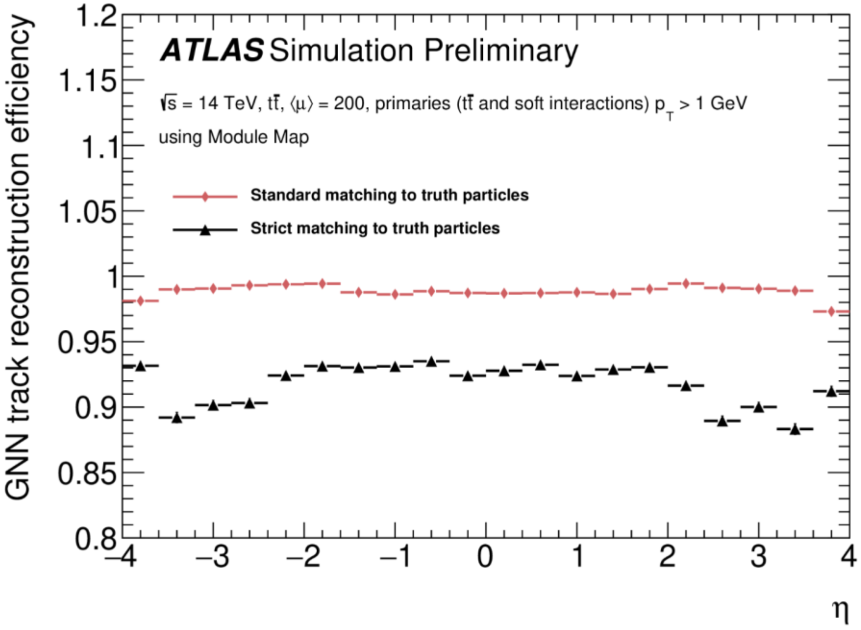
[9] Acharya, N., Liu, E., Lucas, A., Lazar, A. Optimizing the Exa.TrkX Inference Pipeline for Manycore CPUs. (2022). Presented at the Connecting the Dots 2022 workshop. https://indico.cern.ch/event/1103637/contributions/4821918

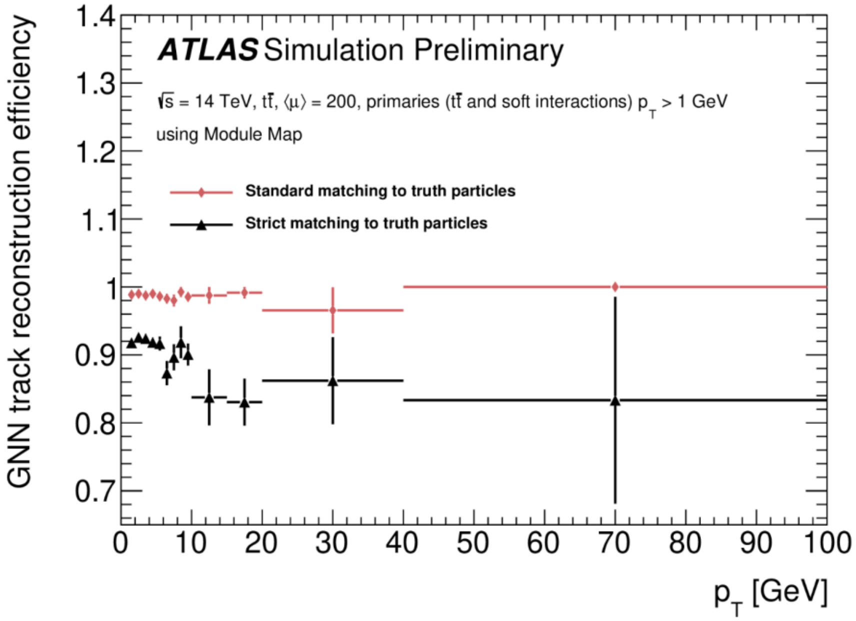
[10] Liu, R., Murnane, D., et al. Hierarchical Graph Neural Networks for Particle Reconstruction. (2022). Presented at the ACAT 2022 conference. https://indico.cern.ch/event/1106990/contributions/4996236/

[11] Murnane, D., Caillou, S.,. Heterogeneous GNN for tracking. (2022). Presented at the Princeton Mini-workshop on Graph Neural Networks for Tracking. https://indico.cern.ch/event/1128328/contributions/4900744

https://exatrkx.github.io

ATLAS ITk Performance Plots

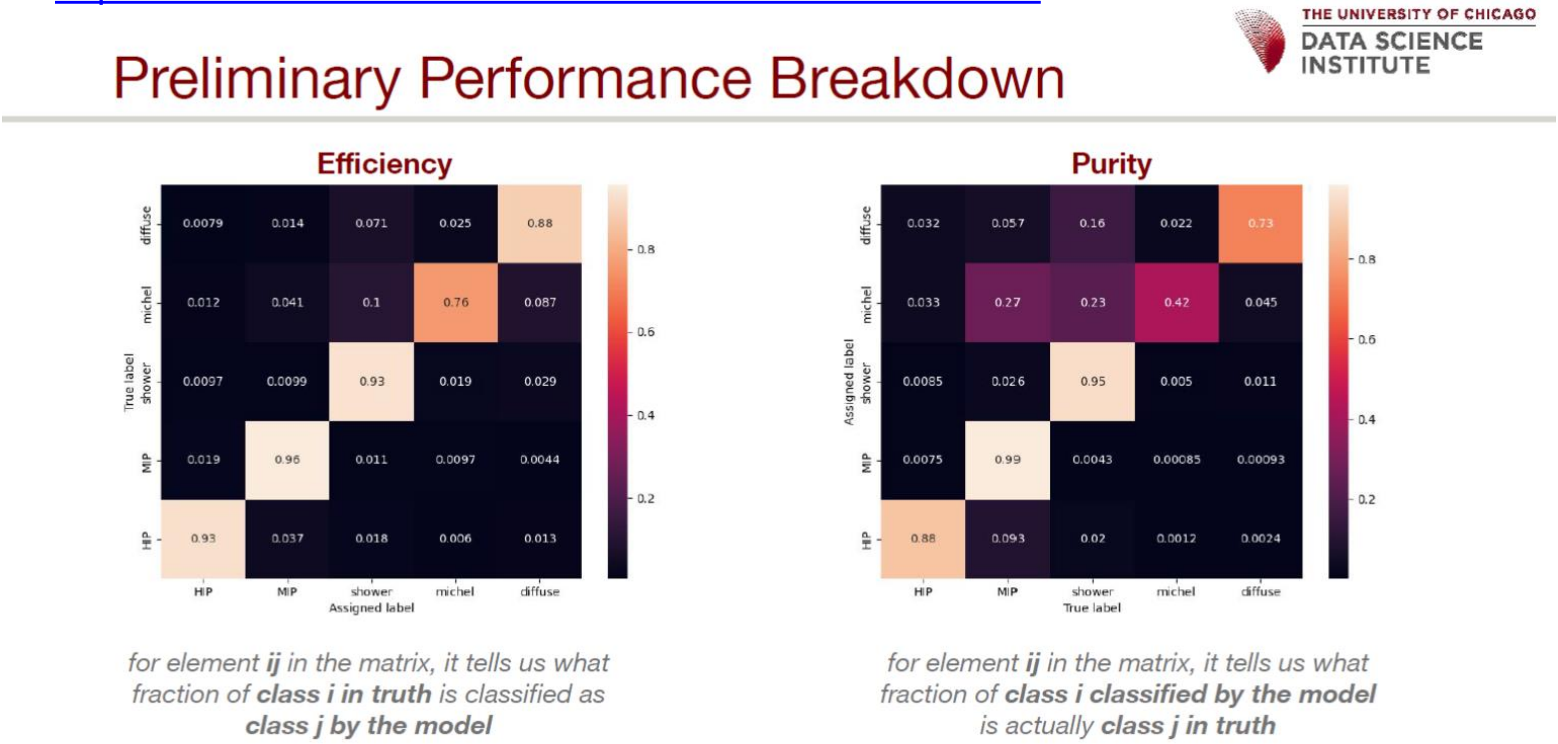




https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/PLOTS/IDTR-2022-01/

LArTPC Performance Plots

https://indico.cern.ch/event/1103637/contributions/4821839/



THE UNIVERSITY OF CHICAGO
DATA SCIENCE INSTITUTE

Preliminary Performance Breakdown

Efficiency

True label \ Assigned label	HIP	MIP	shower	michel	diffuse
diffuse	0.0079	0.014	0.071	0.025	0.88
michel	0.012	0.041	0.1	0.76	0.087
shower	0.0097	0.0099	0.93	0.019	0.029
MIP	0.019	0.96	0.011	0.0097	0.0044
HIP	0.93	0.037	0.018	0.006	0.013

*for element **ij** in the matrix, it tells us what fraction of **class i in truth** is classified as **class j by the model***

Purity

Assigned label \ True label	HIP	MIP	shower	michel	diffuse
diffuse	0.032	0.057	0.16	0.022	0.73
michel	0.033	0.27	0.23	0.42	0.045
shower	0.0085	0.026	0.95	0.005	0.011
MIP	0.0075	0.99	0.0043	0.00085	0.00093
HIP	0.88	0.093	0.02	0.0012	0.0024

*for element **ij** in the matrix, it tells us what fraction of **class i classified by the model** is actually **class j in truth***