NuGraph3

Hierarchical GNN for Neutrino Physics Event Reconstruction

V Hewes

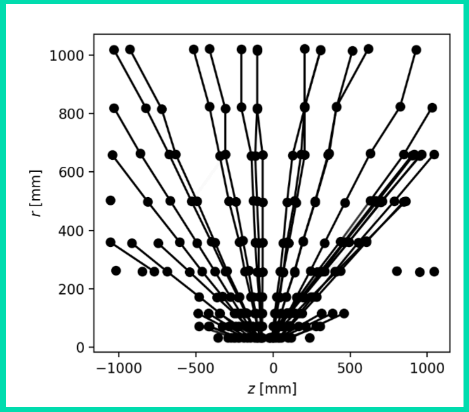
Neutrino Physics and Machine Learning workshop

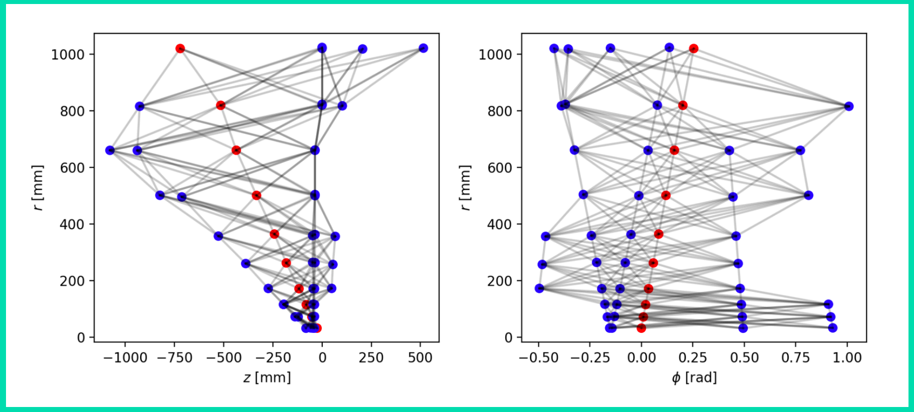
16th June 2026

Introduction

- **NuGraph** is a graph neural network (GNN) for end-to-end event reconstruction in neutrino physics detectors.
- **NuGraph3** is a **hierarchical GNN** that operates on detector hits to perform a full reconstruction including clustering, particle ID, background filtering and more.
- The second and third generation models are generic in design, and can be applied across a range of detector geometries.
- These models are currently being used for reconstruction across several LArTPC experiments including MicroBooNE, ICARUS, SBND, ProtoDUNE and DUNE.
- NuGraph is being utilized across a range of physics interactions, including neutrino interactions, proton decay, test beam events and supernova neutrino events.
- This talk will focus on results from the **MicroBooNE open data release**.

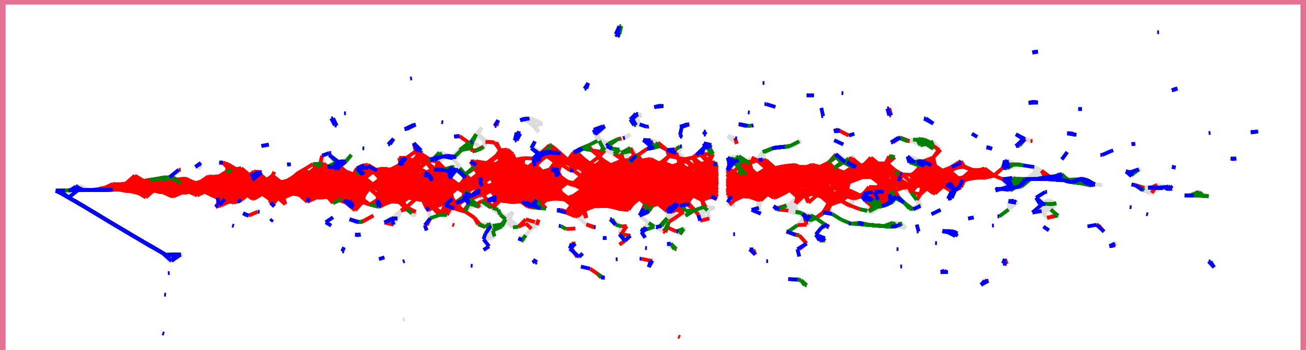
Exa.TrkX and NuGraph



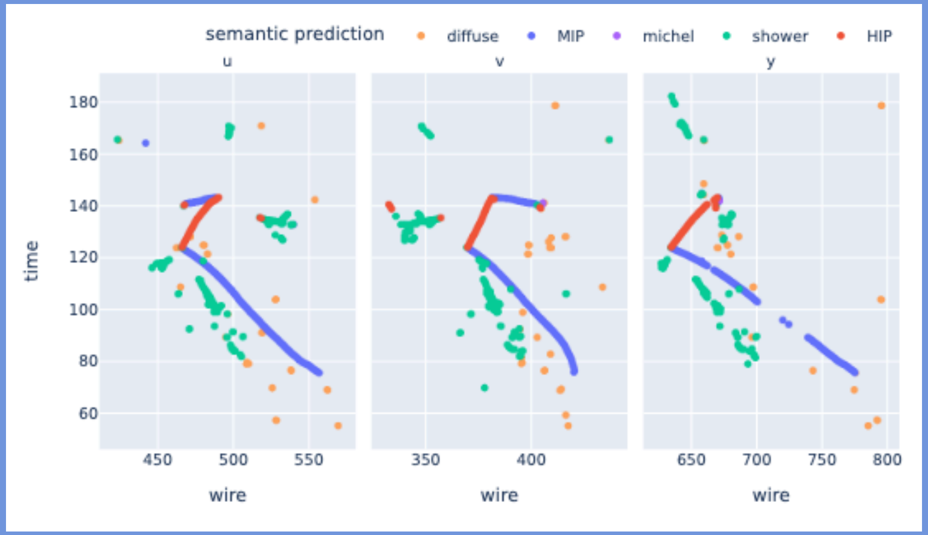


- The HEP.TrkX project developed a prototype GNN for jet reconstruction in the HL-LHC.
- NuGraph was developed within its successor Exa.TrkX, a project to develop GNN-based reconstruction in the HL-LHC and LArTPCs.

- First-generation NuGraph1 architecture adapted HEP.TrkX prototype for LArTPCs.
- Promising performance, but link prediction mechanism was inappropriate for neutrino interactions in LArTPCs.

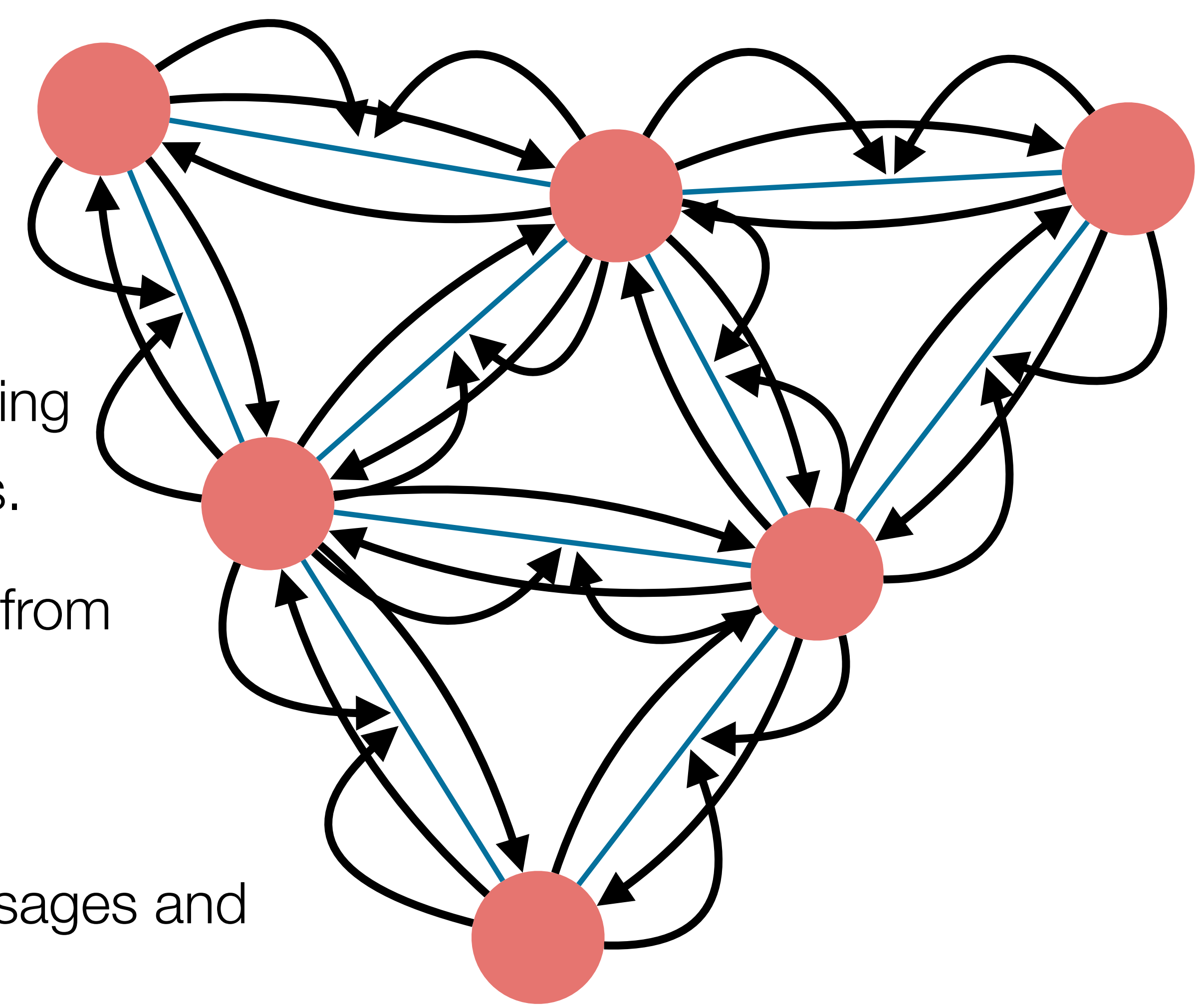


- NuGraph2 is a fully redesigned architecture for use with LArTPCs.
- Performs node classification on detector hits:
  - 98% efficient at removing cosmic background hits.
  - 95% efficient at classifying hits according to particle type.
- For more information see *Phys. Rev. D* **110** (2024) 3, 032008, arXiv:2403.11872.



Attention message-passing

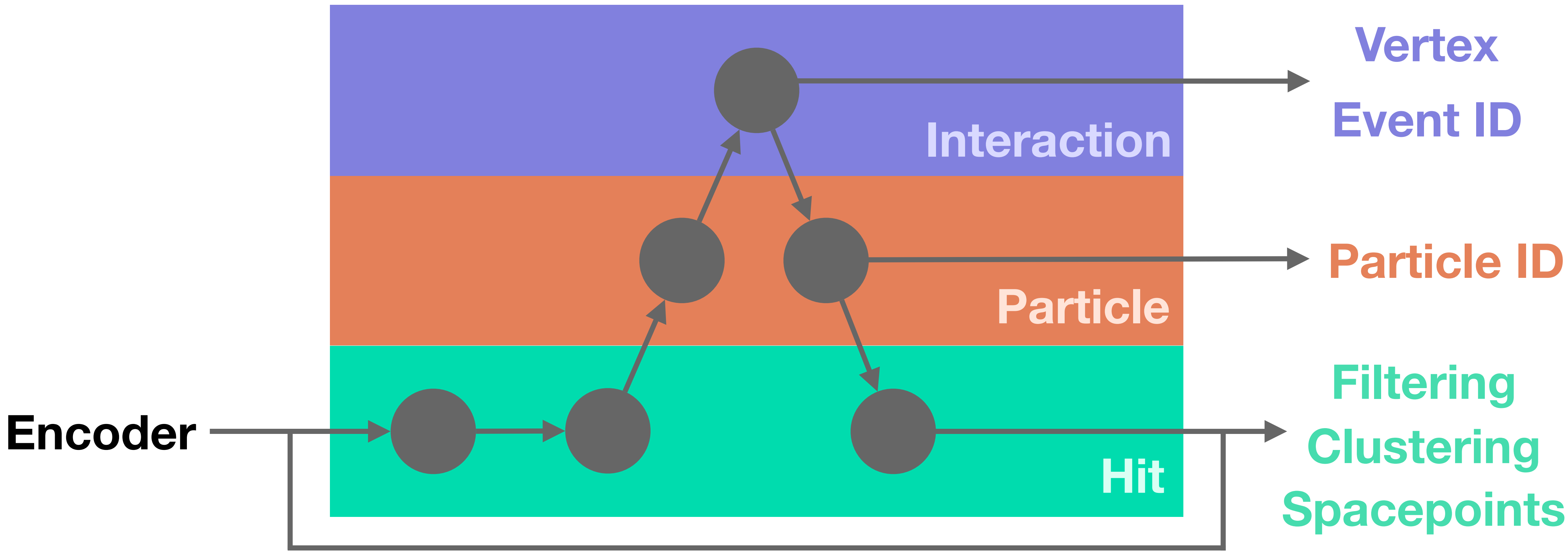
- Describe detector hits as a **graph** represented by **nodes** and **edges**.
- Use **hit charge integral**, **RMS** and **position** to form node features.
- Use **Delaunay triangulation** to form connections between nodes.
- Encode hit features into **learned embedding** h_i^0.
- Form **edge attention weights** a_{ij} by concatenating and transforming source and target node features.
- **Pass weighted messages** across graph edges from source nodes and aggregate into target nodes: $h_i^1 = A_j(a_{ij}h_j^0)$.
- **Repeat this procedure iteratively** to pass messages and refine node embedding h_i^n.



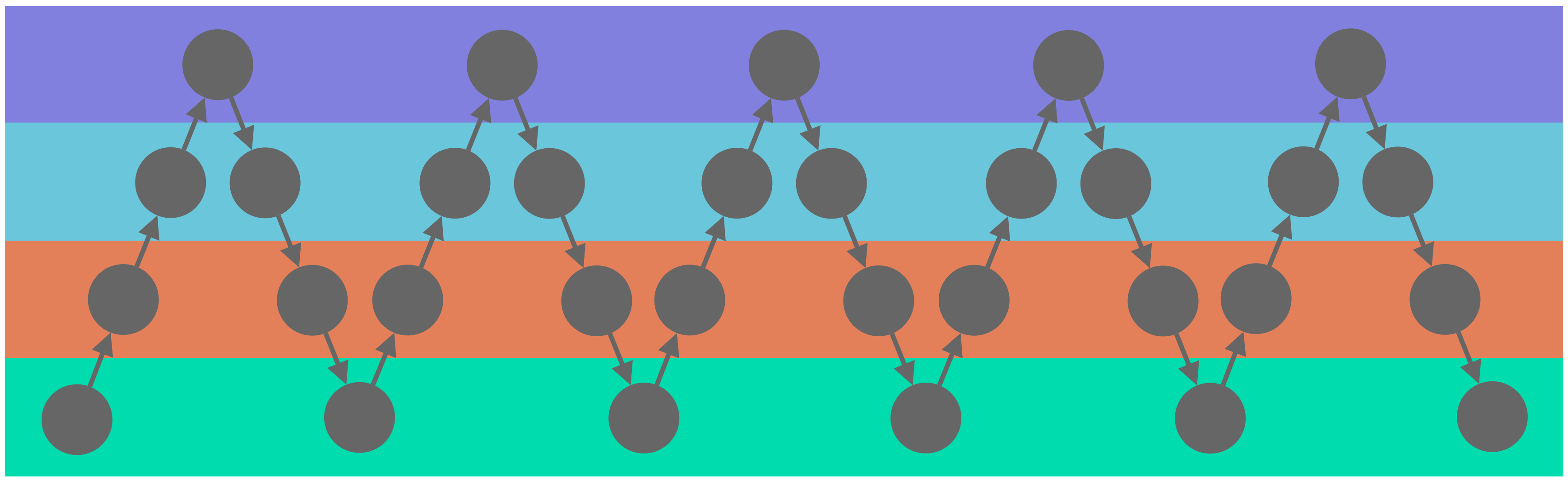
NuGraph3

- Third-generation architecture adopts a **hierarchical graph network** approach.
  - Construct a **heterogeneous graph** with nodes representing different levels of information such as hits, particles and interactions.

- This approach is still in active development, so results shown here are work in progress!



Sawtooth mechanism



- Refer to this hierarchical architecture as the **sawtooth mechanism**, due to the iterative triangular structure.
- Under this construction, **all hierarchical levels converge towards a consistent final state simultaneously**.
    - Information learned at the upper hierarchical levels (ie. particle, event) can be used to refine the lower hierarchical levels (ie. hit).
    - **Leverage correlations between levels**: predicting a hit is muon-like, predicting a particle instance is a muon, predicting an interaction is a CC ν_μ event.
- Able to reproduce performance of NuGraph2 model for semantic, filter & vertex decoders trained simultaneously.

End-to-End Architecture

- NuGraph3 is a single architecture that operates directly on low-level detector outputs.
- When trained, NuGraph3 learns a feature embedding at each level of the hierarchy.
- Decoders operate on these embeddings to predict output quantities.
- NuGraph3 only needs to be trained once for each dataset, as multiple decoders can be trained simultaneously.
- Training multiple decoders is not only possible but preferred, as the model can leverage correlations between tasks to learn faster.

Interaction decoders

Event
Classify interaction type (eg. ν flavour)

Vertex
Predict 3D vertex position

Particle decoders

Semantic
Classify particle type

Hit decoders

Semantic
Classify particle type

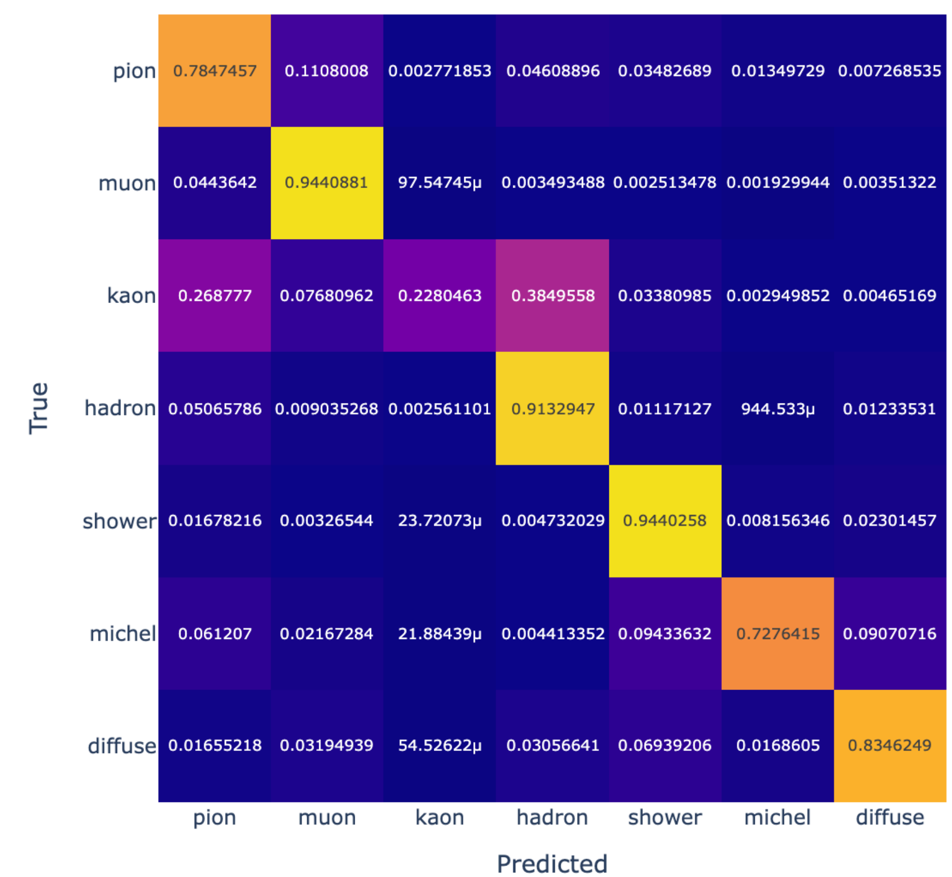
Filter
Remove background hits

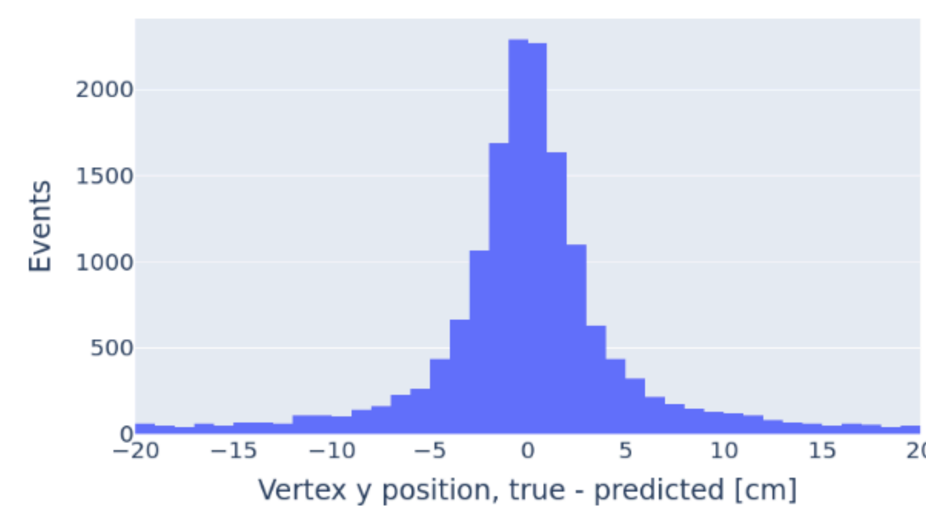
Instance
Cluster hits into particles

Position
Predict 3D position for 2D hit

NuGraph3 performance

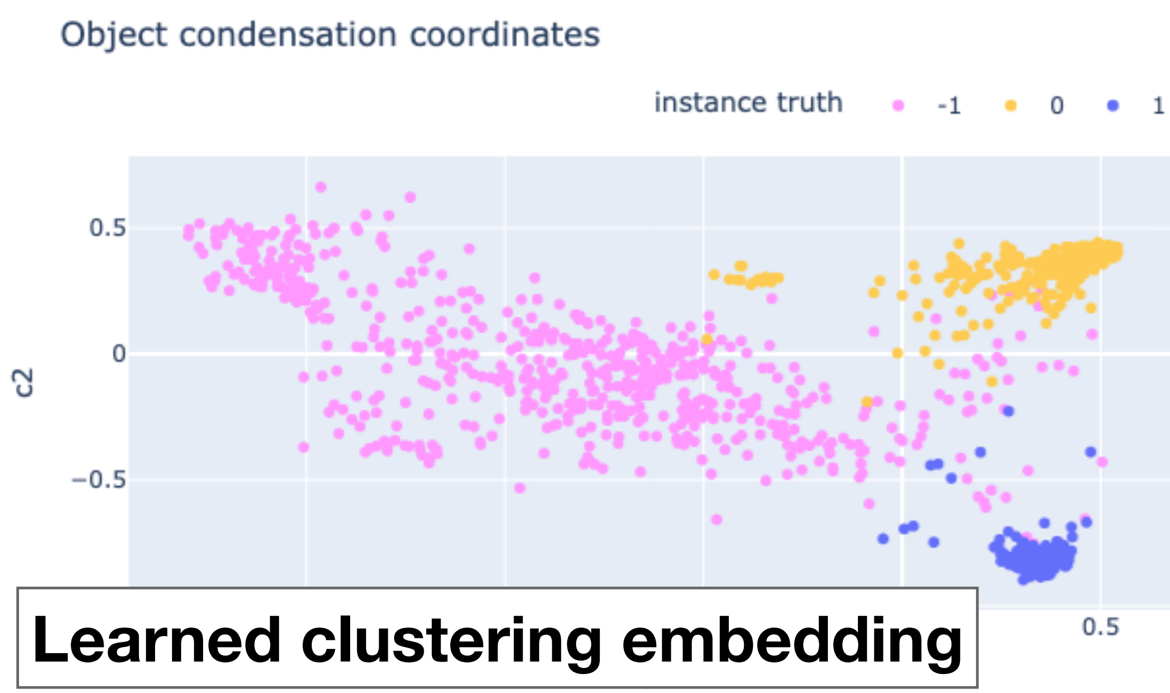
- **98% efficiency** for removing background hits.
- **Expanded semantic labelling**:
  - HIP + MIP ➝ muon, pion, kaon, hadron.
  - **90% efficiency** for semantically labelling hits.
- **Predict event-level quantities**:
  - **Classify entire event** with semantic label.
    - eg. neutrino flavour, proton decay, etc.
  - Reconstruct **3D vertex position**.
    - Perform 2D ➝ 3D coordinate transformation, identify vertex inside event.
    - Achieve **~5cm spatial resolution** in MicroBooNE, but hope to improve on this.
- **Reconstruct spacepoints**:
  - Predict 3D spatial position for each 2D detector hit.

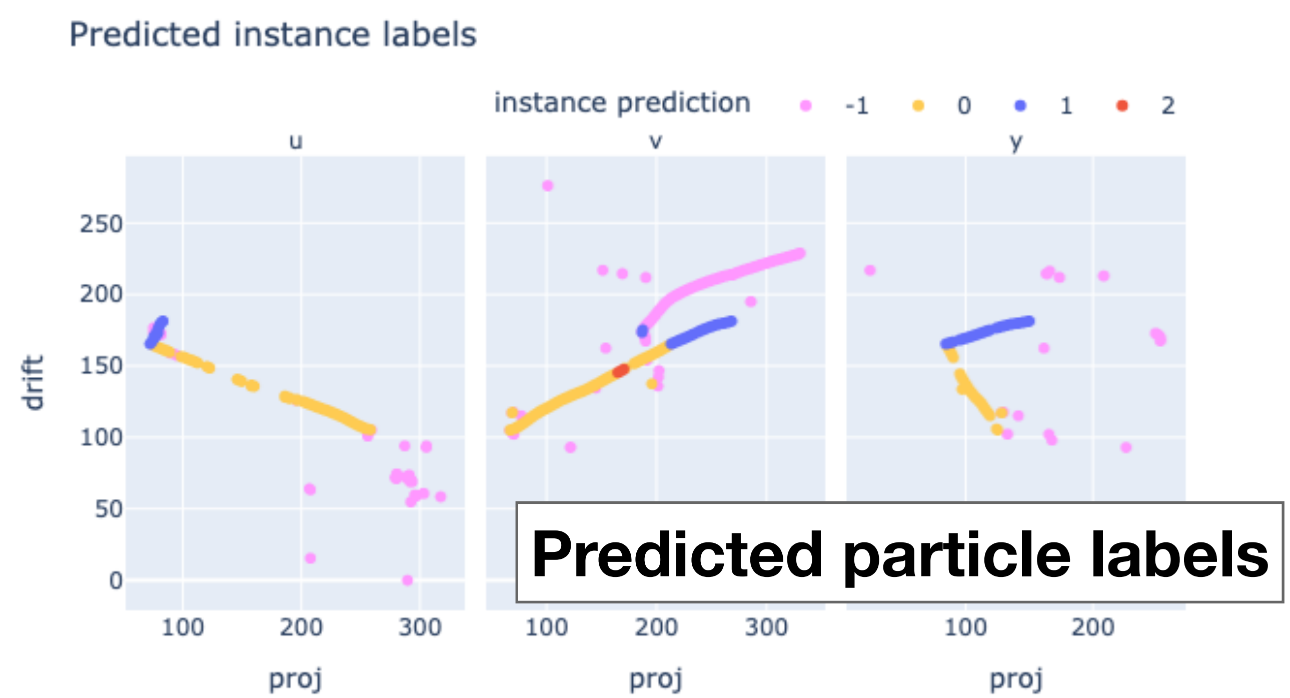




Clustering

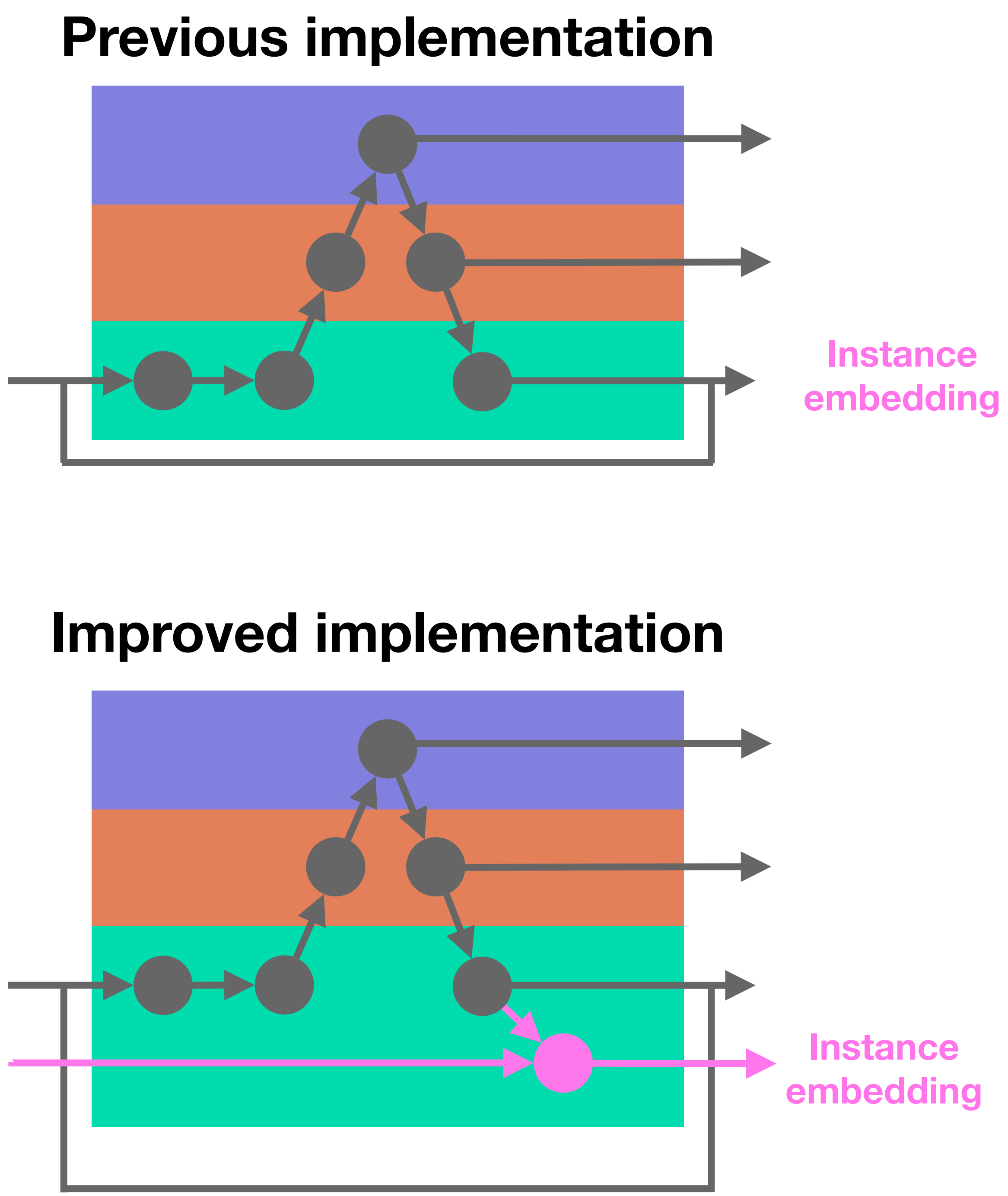
- Utilize **object condensation** (arXiv:2002.03605) to project hits into **learned clustering space**.
- Utilize clustering algorithm (eg. DBSCAN) to **materialize clustering space**.
- Use materialized instances to form **particle nodes**.





Clustering performance

- NuGraph3's instance decoder clusters 2D hits **across views**.
- Hits from different views are projected into a consistent latent clustering space.
- NuGraph3 clustering implicitly performs matching of clusters between 2D views.
- Recently restructured the model's core loop to predict and refine the clustering space from the start, instead of generating it in a decoder.
- NuGraph3 clustering with object condensation achieves **Adjusted Rand Score of 0.76** for the MicroBooNE open data release.
- Currently working to refine this implementation, and generalize the approach to more complicated topologies such as high-energy ν_τ interactions in the DUNE far detector.



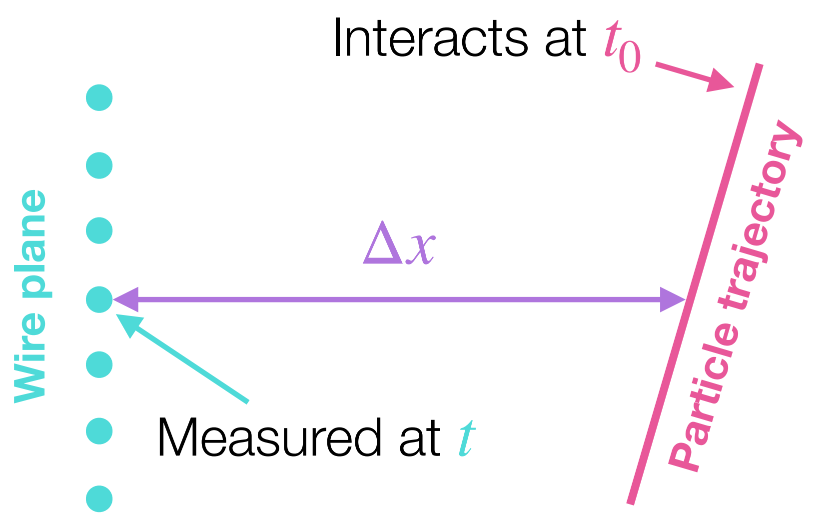
Materializing particle instances

- Object condensation embedding is materialized using the **DBSCAN** algorithm.
- Building the intermediate level of the hierarchy requires **materializing particle nodes during the model's forward pass**, rather than as a post-processing step.
- Utilized the **RAPIDS GPU implementation of DBSCAN**, but handling GPU memory consistently with PyTorch tensors requires overriding PyTorch's default memory manager with the RAPIDS memory manager, RMM.
  - This led to significant slowdown in training and inference.
- Reverted to CPU implementation of DBSCAN from `sklearn`, and moderate training time by **only materializing particle nodes during inference**.

Inductive bias

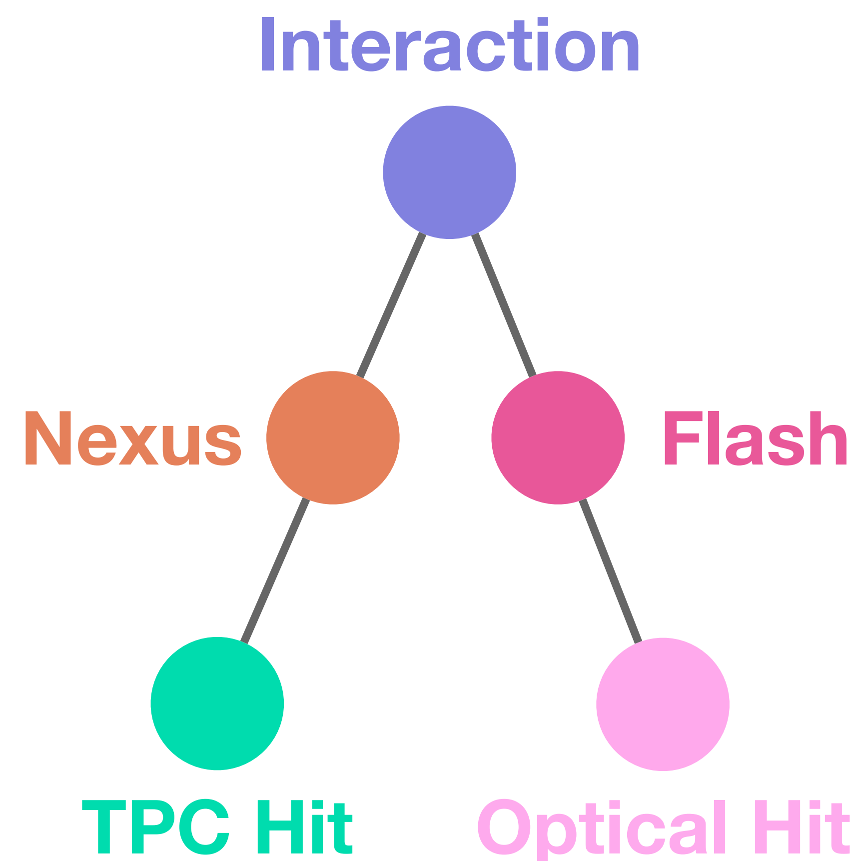
- One of the major design philosophies in developing NuGraph3 is to **maximize inductive bias**.
- ML-based reconstruction techniques often expect the neural network to relearn information that we as physicists already know.
- Work to build our knowledge of detector geometry and physics into the model, isolating specific pattern recognition tasks and simplifying the model.
- For example:
  - Predicting the neutrino vertex position from detector hits requires an implicit 2D-to-3D coordinate transformation.
  - Teach the model to perform this coordinate transformation explicitly for each hit, so the vertex decoder doesn't need to do it implicitly.
  - Predict only missing coordinates, leveraging our knowledge of detector geometry wherever possible.
  - Utilize optical system to help break degeneracies.

Incorporating optical system



$$t = t_0 + v_d \Delta x$$

drift velocity



- Currently working with undergraduate students at the University of Chicago Data Science Institute to **incorporate optical information into NuGraph**.
- TPC optical system provides t_0 in situations where it is not provided by trigger system.
- Incorporate new node types into graph representing **individual optical hits** and **flashes**, which aggregate individual optical hits that are coincident in time.
- Prompt light flashes need to be **matched to associated TPC activity**.

NuGraph HDF5 file formats

- NuGraph utilizes an **HDF5 workflow** that summarizes low-level data products from the Art event record into HDF5 tables.
- This file format is general enough to be used in many different ML applications.
- NuGraph's HDF5 file format was used as the standard for the **MicroBooNE open data release**.
- The pynuml python package offers algorithms to efficiently interface with this file structure in parallel with MPI, and to generate particle truth labels.
- An overhauled and improved version of the HDF5Maker module is currently being merged into larrecodnn, in order to serve as a common data format for DUNE.
- This improved file format utilizes a common structure for storing charge and light information, enabling ML applications that combine both types of information.

University of CINCINNATI

Data product tables and columns

Charge system

LDep table
run, subrun, event, oh_id, g4_id, energy, position_x, position_y, position_z

OpHit table
run, subrun, event, oh_id, flash_id, channel, time, area, width, amplitude

Flash table
run, subrun, event, flash_id, pe, position_y, position_z, position_t, width_y, width_z, width_t

Event table
run, subrun, event

Neutrino table
run, subrun, event, nu_id, is_cc, pdg_code, lepton_energy, vertex_x, vertex_y, vertex_z, vertex_t, momentum_x, momentum_y, momentum_z, momentum_e

GENIE truth

Particle table
run, subrun, event, g4_id, nu_id, pdg_code, parent_id, momentum_x, momentum_y, momentum_z, momentum_e, start_x, start_y, start_z, start_t, end_x, end_y, end_z, end_t, start_process, end_process

Geant4 truth

Optical system

EDep table
run, subrun, event, wh_id, g4_id, energy, position_x, position_y, position_z

WireHit table
run, subrun, event, wh_id, integral, rms, tpc_id, plane, wire, time, view, proj, drift

Spacepoint table
run, subrun, event, sp_id, position_x, position_y, position_z, wh_id_1, wh_id_2, wh_id_3

Summary

- **NuGraph3** is a **hierarchical GNN architecture** for reconstructing particle interactions in neutrino physics detectors.
- NuGraph3 achieves:
  - **98% efficiency** in filtering background.
  - **90% efficiency** in labelling particle type.
  - **0.76 Adjusted Rand Score** in clustering hits into particles.
  - **~5cm resolution** in placing neutrino interaction vertex.
- Work to improve NuGraph3 further is ongoing, with:
  - Further refinements to **particle clustering**.
  - Incorporation of the **optical system**.
  - A refined and overhauled **HDF5 workflow**.