Rotationally Equivariant Graph Neural Networks

Sofia Graziano

The problem

- The LHC is a unique particle accelerator complex that collides protons at unprecedented energies
    - In 2025, the LHC will be receiving an upgrade in which it will collect data of increasing complexity and at an increasing rate
- The overarching goal is to accurately track the trajectory of particles in charged particle collisions within the HL-LHC detectors

The problem

- The analysis pipelines of the proton collisions rely on the reconstruction of the 3D trajectories of the particles within the detector
    - This is currently solved by combinatorial optimization methods, but CPU time to reconstruct the trajectories from the measurements is expected to increase faster than the projected computing resources
- Therefore, the goal of the research project is to utilize rotational equivariances within the data to construct a GNN that would increase accuracy and decrease CPU time in preparation for the HL-LHC addition

TrackML

- We decided to use the TrackML dataset because of its relevance and usefulness to the project
    - For each event, it contains 3D hit position and truth information about the particles that generated them
    - The data set is very large: 10,000 events, a billion points, one hundred million tracks

TrackML

- Meant to simulate measured particle hits similar to what is expected for an HL-LHC experiment
  - Based on large surface all-silicon detectors with a cylinder-like geometry in the central regions and a disk-like geometry in the forward regions
- The coordinate system is a right-handed Cartesian coordinate system (x, y, z) with the global z axis defined along the beam direction*
  - *This is the axis of symmetry of the cylinders of disks composing the detector
  - In order to measure the particle momentum, tracking detectors are embedded in a strong magnetic field
- When a charged particle moves through a constant magnetic field, it follows a helical trajectory
  - The magnetic field is usually aligned with the beam direction such that the particle is bent in the (x - y) plane

TrackML

- Comprises multiple independent events, where each event contains simulated measurements (3D points) of particles generated in a collision between proton bunches at the LHC
- The training dataset contains the recorded hits, their ground truth counterpart and their association to particles, and the initial parameters of those particles
- Each event can have up to four associated files that contain hits, hit cells, particles, and the ground truth association between them

Graph Neural Networks

- How does this configuration lend itself to a graph?
  - Hits are nodes and pairs of hits on adjacent layers as edges
    - If the edges correspond to true track segments, the edges are labeled as 1
    - If the edges don't correspond to true track segments, the edges are labeled as 0
- Three stages: graph construction, edge classification, and track building

A visual representation of track building

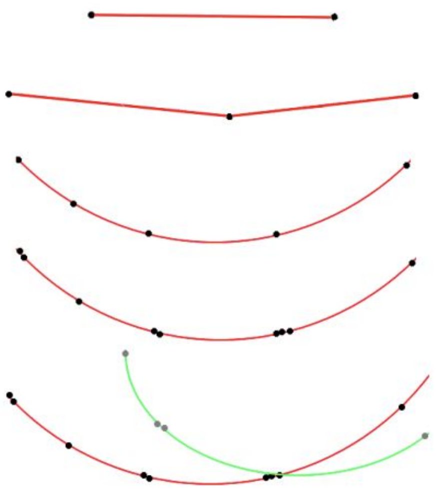
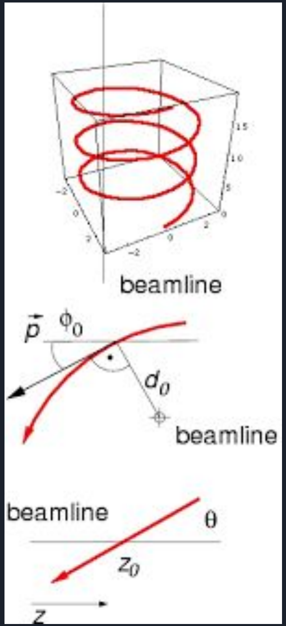


Figure 18: Schematic representation of the steps of the challenge winner algorithm. From top to bottom : pair finding, extension to triplet, extension to tracks, addition of hits from overlapping modules, and final track disambiguation.

Where rotation comes into play

- The hits created by each particle together form tracks. The tracks are slightly distorted helices with axes parallel to the z-axis.
- Instead of using a typical GNN, we can use this new model to learn graph neural networks equivariant to rotations, translations, reflections, and permutations
    - Does not require computationally expensive higher-order representations in intermediate layers
    - Still achieves competitive or better performance in comparison to existing methods
    - Model is easily scaled to higher-dimensional spaces



What we were able to do

- A lot of this project was focused on figuring out what would be possible
  - Given that this is such a novel approach to charged particle tracking, there was not a lot of existing literature on the topic
    - Conducted a literature review
- Additionally, there was a big learning curve with GNNs
  - Once we were able to get me up to speed with the content, it took a lot of technical effort to get the dataset and existing IN running
  - By the time that this happened, the fellowship was nearly over

Looking forward

- Had we had more time, it would be wise to
  - Train the GNN to be rotationally equivariant while
    - Reducing the time it takes to construct the graph
    - Increasing the accuracy
- This is extremely important because each event is generated with 200 pileup interactions on average
  - Need to be able to sort through these efficiently and accurately
- Exploiting rotational equivariance would cut down on time tremendously, as existing algorithms take anywhere from 10 minutes to 1 day per event

Special thanks

- Thank you to my mentors, Savannah Thais and Daniel Murnane for everything!