IRIS-HEP Project Proposal

Augmenting Line-Segment Tracking with Graph Neural Network

Mentor: Philip Chang

Povilas Hubert Pugzlys

Introduction

The increase of the pile-up in the upcoming HL-LHC will present a challenge to event reconstruction for the CMS experiment. The single largest contribution to the total reconstruction time comes from charged-particle tracking. Without algorithm innovation, the projected charged-particle reconstruction timing is projected to exponentially increase [1]. This increase in timing in combination with the fact that the computational performance of single thread processors is plateauing, CMS Collaboration estimates that without algorithmic innovation the computing resource requirement will hit a factor 2 to 5 over the projected computing capabilities [2]. This can seriously hinder physicists to publish timely scientific results.

Line-Segment Tracking

This motivates a new approach in tracking to develop a new algorithm that are parallel in nature to alleviate problems of combinatorics, and also can leverage industry advancements in parallel computing such as the GPUs. In light of this, Line-Segment Tracking project started.

Line-Segment Tracking (LST) project leverages the CMS outer-tracker's doublet modules to build mini-doublets (a pair of hits in each layer of the doublet layer) in parallel, and subsequently build line-segments via connecting consistent pair of mini-doublets across different logical layers of the tracker, all done on high-performance GPUs. Eventually, the line-segments are linked together iteratively to form a long chain of line-segments to produce list of track candidates.

The parallel nature of the LST algorithm allows the algorithm to naturally lends itself for GPU usage. The project has produced on-par performance with the existing tracking alternatives, and have been integrated to central CMS Software as a step towards production [3].

Graph Neural Network for Line-Segment Tracking

As LST algorithm creates line-segments and links them to create track candidates, a graph representation of hits and linking between them is naturally obtained. In other words, LST can also be thought of as a fast graph producing algorithm. The project will take the graph data and develop GNN models that classify linkings. We plan to integrate the GNN model to the LST algorithm to augment its capability to produce

high-quality track candidates at a shorter time while keeping the same or better tracking performance. Also, a solution for a "one-shot" linking of long chains of line-segments in one algorithm instead of through iteration will also be studied.

Estimated Timeline

Week 1 & 2: Understanding the preliminary LST GNN workflow for Line Segment classification

Week 3: Creating example of running the Line Segment classification inference on C++ environment with TorchScript

Week 4: Integrating the inference with LST's CUDA code to run the inference on GNN

Week 5: Validating the implementation in the LST framework

Week 6: Performing optimization of utilizing the GNN inferences to measure performance gain in the efficiency metric of LST framework (i.e. efficiency, fake rate, and duplicate rate)

Week 7: Perform large scale hyperparameter optimization to find best resulting model architectecture

Week 8: Perform research and development of extending the ability to classify Triplets, and beyond, with the Line Graph transformation approach, which would enable "one-shot" inference. Explore learning from and collaborating with other GNN tracking researchers at CMS on different approaches of connecting objects, such as object condensation approach [4].

Week 9: Wrap up the project, document and summarize the findings to allow for next steps

Reference

[1] Giuseppe Benedetto Cerati for the CMS Collaboration, *Tracking and vertexing algorithms at high pileup*. Tech. rep. CERN, 2014. https://cds.cern.ch/record/1966040

[2] CMS Offline Software and Computing, *CMS Phase-2 Computing Model: Update Document*, Tech. rep. CERN-CMS-NOTE-2022-008, 2022. https://cds.cern.ch/record/2815292

[3] Philip Chang et al. *Segment Linking: A Highly Parallelizable Track Reconstruction Algorithm for HL-LHC*. J. Phys. Conf. Ser., 2375(1):012005, 2022.

[4] G. DeZoort, K. Lieret, *Object Condensation Tracking* https://indico.jlab.org/event/459/contributions/11741/