Application of quantum computing techniques in particle tracking at LHC

DAIYA AKIYAMA\textsuperscript{A}, KOKI ARAKAWA\textsuperscript{A}, WAI YUEN CHAN\textsuperscript{B}, SANMAY GANGULY\textsuperscript{B}, TOSHIAKI KAJI\textsuperscript{B}, JURI MINAMI\textsuperscript{A}, RYU SAWADA\textsuperscript{B}, JUNICHI TANAKA\textsuperscript{B}, KOJI TERASHI\textsuperscript{B}, KOHEI YORITA\textsuperscript{A}

WASEDA UNIVERSITY\textsuperscript{A}, ICEPP, THE UNIVERSITY OF TOKYO\textsuperscript{B}

CHEP2023

2023/5/9
Motivation

- HL-LHC is coming (~2027).
- With larger pile-up ($\langle \mu \rangle \sim 200$) and high readout rate, CPU consumption will dramatically increase.
  - Especially track reconstruction -> New techniques are needed

CERN-LHCC-2022-005

ATLAS Computing Public Result
Introduction: Quantum Annealing

• Quantum Annealing:
  An optimisation process for finding the global minimum of a given function by using quantum fluctuations.

• Quantum Annealer: The machine which is designed to perform the quantum annealing process.
  e.g. D-Wave computers

- Quantum Annealer can only deal with the problem which can be transform to a "QUBO" or "Ising" function.
Quantum Pattern Recognition: Algorithm Overview

- We found the Quantum Annealing could improve the speed of pattern recognition and provide another way to perform the particle tracking. Result published on: [arXiv: 1902.08324](https://arxiv.org/abs/1902.08324)

\[ O(a, b, T) = \sum_{i}^{N} a_i T_i + \sum_{i}^{N} \sum_{j<i}^{N} b_{ij} T_i T_j \]

- \( T_i \): potential triplet
- \( a_i \): Bias weight which has been set to 0.
- \( b_i \): The coupling strength, depending on the relation between \( T_i \) & \( T_j \)
GNN application in Quantum Pattern Recognition
Motivation: Replace QUBO formation by ML techniques.
Graph neural network (GNN) is a suitable choice to deal with it.

Data preparation (I)

- The parameter “Hit ID” parameter is the key for the data extraction.
  - Hits in each Triplet \( T \) contained Hit ID.
  - These IDs are used to form the doublet list and the hit list.
    - The GNN Target, is given by matching the Hit ID from the Hit list per \( T_iT_j \)
    - If the Hit ID from a pair of hits \( (n_i, n_j) \) in the is equal to Hit ID in the doublet list:
      Target (edge score) = 1
    - If it’s not, then Target (edge score) = 0
  - We want to train the GNN to predict the correction combination of edge scores.
  - We would like to preform a edge classification in order to form the \( T_iT_j \) from raw hits.
    - To do that we need to consider all the combinations of edges scores per graph.

In single event:

\[ T_iT_j \text{ from the QA solved QUBO (Hit ID)} \]

"Raw input" from TrackML challenge datasets:
(HitID, x,y,z, detector layers etc.)

For each \( T_iT_j \):

- Doublet list (3 pair of HitID)
- Hit list (HitID, x,y,z, detector layers etc.)

In 1 GNN sample:

- GNN Target
- Trainable parameters

The graph:

- \( n_i \): node \( i \)
- \( e_{ij} \): edge for \( n_i \) to \( n_j \)

\[
O(a,b,T) = \sum_{i=1}^{N} a_i T_i + \sum_{i=1}^{N} \sum_{j<i}^{N} b_{ij} T_i T_j
\]
In order to get all the combinations, we need to search for nearby hits which is close to the original hits in the same detector layer (i.e. same r-coordinate).

- Our dataset have 269550 samples, only 6% of them are Signal samples.
- 40% are used for training; 40% are used for validation; 20% are used for test.
Network architecture

<table>
<thead>
<tr>
<th>Node features ‘x’</th>
<th>Node features ‘en’</th>
<th>Edge features ‘D’</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>r</td>
<td>Δr</td>
</tr>
<tr>
<td>y</td>
<td>Detector layer ID</td>
<td>Δz</td>
</tr>
<tr>
<td>z</td>
<td>Detector volume ID</td>
<td>Curvature $atan2(Δr, Δz)$</td>
</tr>
<tr>
<td></td>
<td>Detector module ID</td>
<td></td>
</tr>
</tbody>
</table>

In each EdgeConv Layers

- From each layer, the EdgeConv parameters $e_x$, $e_{en}$ are passing to the next layer.

- Final output, $e_s^N$ is the edge score coming from the last EdgeConv layer.
• We use Cross Entropy Loss as our Loss function.
• The impact of batch size is studies and batch size = 8 has been used.
• Underfitting observed.
**Edge score distributions**

- “Predict: TP”: edges have target edge score = 1
- “Predict: FP”: edges have target edge score = 0
ROC curves are shown with different batch size.
- Smaller batch size gives higher AUC.
- The best cases here is given by batch size = 8, with AUC = 0.92
• Sum of edge scores per graph is shown.

• In target, only score = 0, 2, 4, 6 are allowed since $e_{ij} = e_{ji}$ is always true.

• The predicted score, on the other hand, have also score = 1, 3, 5; this is because in some cases, score of $e_{ij} \neq e_{ji}$. This is possible as we didn’t add $e_{ij} = e_{ji}$ as one of the input features of our network.

• However, this is a non-physical result. Since the edges, $e_{ij} = e_{ji}$ have to be always true as this is a physical doublet.
Graphs with scores > 3 are selected

\(-1 \leq b_{ij} \leq -0.2\) is given by the previous study.

We are expecting similar distribution to the original QUBO.

More investigation is needed for the distribution for GNN-generated QUBO in -0.85 to -0.25.
Application for the ATLAS dataset
Application for the ATLAS dataset

- We verified if track finding by annealing machines works in a realistic environment with the ATLAS dataset.
- Detector-hit information is taken from the dataset processed by the ATLAS software (https://cds.cern.ch/record/2767187), while the annealing tracking is done by other standalone software.
- **This study has been performed independently from the previous GNN study.**

- We used Fixstars Amplify Annealing Engine (AE) which was an annealing machine developed by Fixstars.
  - Perform simulated annealing using GPU (NVIDIA A100)
  - 262k bits, fully connected

- This time, we used “doublets” for bits.
  - When two doublets have close curvature in X-Y plane or close angle in R-Z plane, we give them low energy.
  - Doublets and double-pairs were selected before a QUBO building to reduce the size of QUBO.

\[
H(a, b, D) = \sum_{i}^{N} a_i D_i - \sum_{i}^{N} \sum_{j<i}^{N} S_{ij} D_i D_j - \sum_{i}^{N} \sum_{j<i}^{N} W_{ij} D_i D_j + \sum_{i}^{N} \sum_{j<i}^{N} \zeta_{ij} D_i D_j
\]

- \(D_i\) : Potential doublet
- \(a_i\) : Bias weight which is depending on \(N_{holes}\) in a doublet.
- \(S_{ij}, W_{ij}\) : The coupling strength, depending on the \(\Delta \left(\frac{1}{\Delta R}\right), \Delta \theta\) between \(D_i\) and \(D_j\).
- \(\zeta_{ij}\) : The coupling strength, we give constant (\(\zeta_{ij}=5\)).
Event display

- Reconstructed results with 200 muons/event MC sample generated with the ATLAS software.
  - $0.5 \text{ GeV} < p_T < 10 \text{ GeV}$ in $1/p_T$ flat distribution, $|\eta| < 1.0$

Track findings by annealing machines work successfully in a realistic environment.

**ATLAS Simulation Preliminary**
Proof-of-concept
GPU-Based Annealer Reconstruction
MC 200 muons w/o pile-up

- Reconstructed true tracks
- Unreconstructed true tracks
- Fake tracks

**ATLAS Simulation Preliminary**
Proof-of-concept
GPU-Based Annealer Reconstruction
MC 200 muons w/o pile-up

- Reconstructed true tracks
- Unreconstructed true tracks
- Fake tracks

**Layout of the ATLAS inner tracker**
Results with real data

- We applied this algorithm to real ATLAS data taken by non-physics random triggers.
- The efficiency is calculated w.r.t. the ATLAS offline tracks. The matching to the offline tracks is performed if reconstructed tracks with annealing machines share more than 50% of hits with the offline tracks.
- The annealing time was compared with MC sample (10 pions/event with pile-up 20).

- Our algorithm also works successfully with real ATLAS data.
- It is a good starting point to further explore the method.
Conclusions

- 2 type of studies are performed with the annealing tracking algorithm:
  - GNN application in Quantum Pattern Recognition
    - In order to make use of the GNN in the pre-processing stage of this algorithm, we performed an edge-classification using a bi-directional graph made by simplified sample which only contain 4 hits.
    - The result indicated that it is possible but there are rooms to improve.
  - Application for the ATLAS dataset
    - Another study shows that the algorithm can also deal with real data collected from the ATLAS detector using low pileup (and no pileup) samples, in both MC and data.
    - Result looks promising and further development is under consideration.