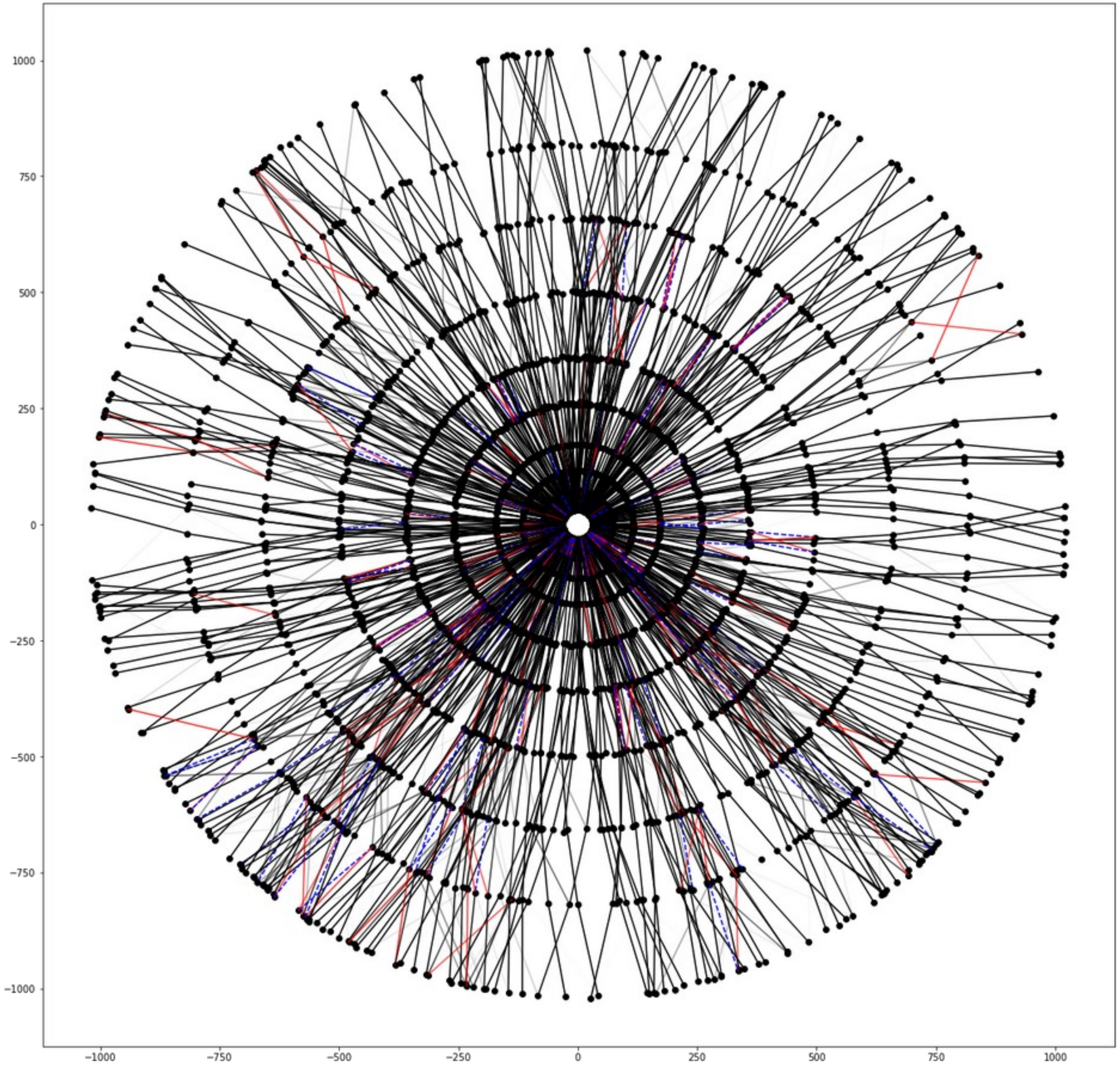
Reconstruction of particle tracks using Deep Neural Networks

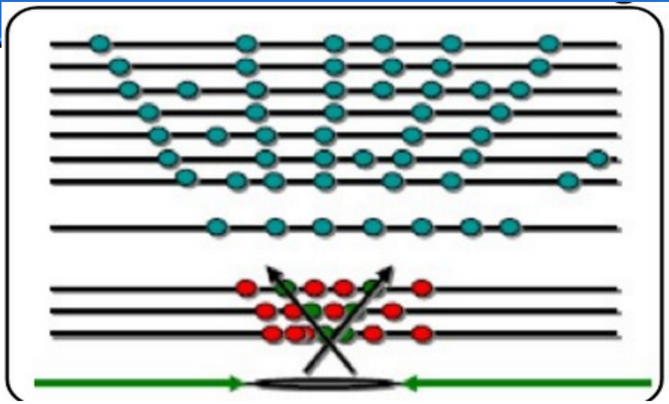


Marcin Wolter

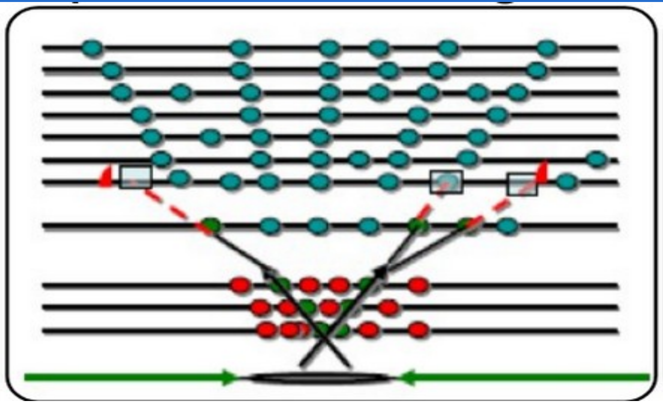
IFJ PAN

11 July 2022

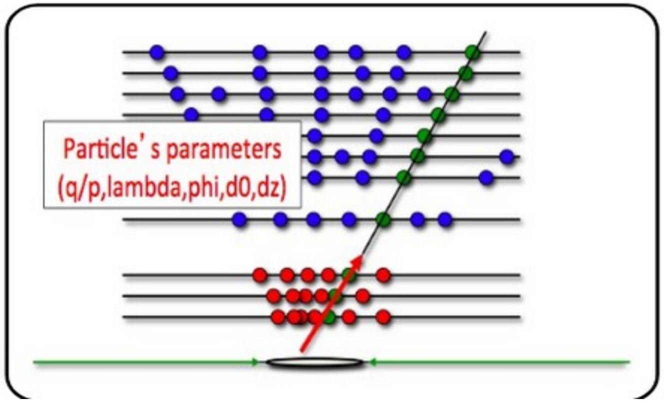
Track reconstruction



Seeding



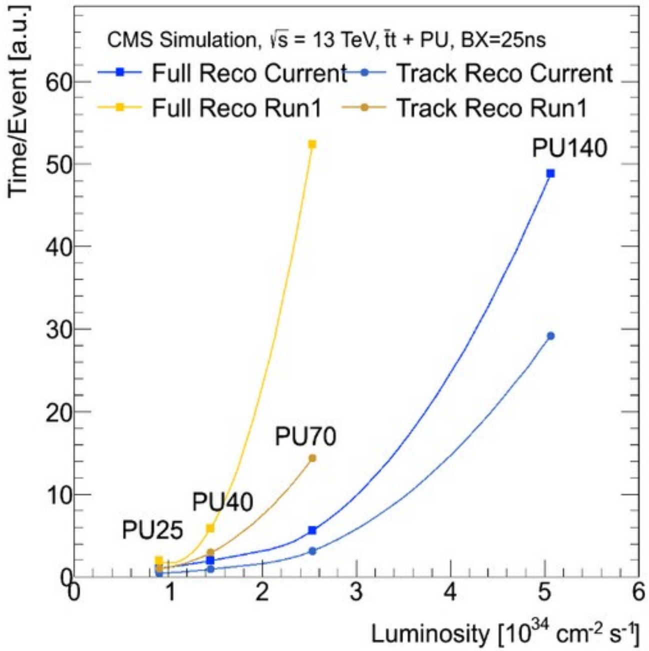
Track Building



Track Fitting

Usually this method works fine, is robust and efficient!

- **Problem:**
  - The time needed to process one event grows quickly (worse than quadratic) with luminosity (number of collisions).
  - Huge part of CPU consumption by the track finding.

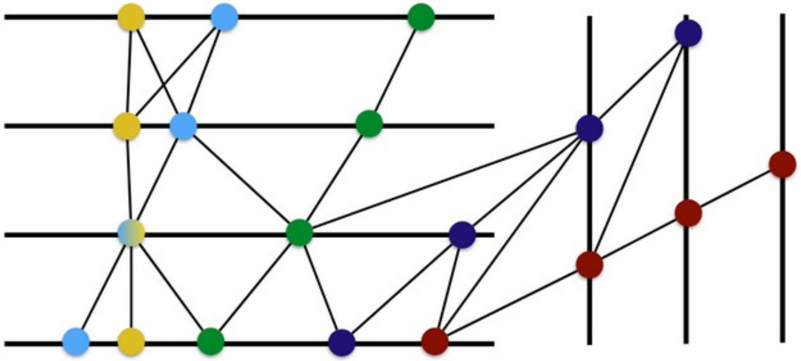


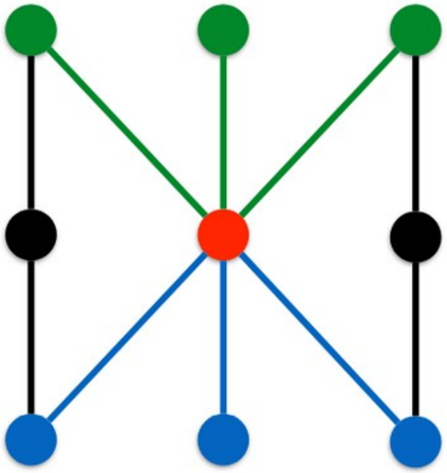
Solution: use Neural Network!

Graph Neural Network

Structure our data as a graph of connected hits

- What kinds of models can we apply to this representation?
- Traditional architectures clearly don't work
- but there's a growing sub-field of ML called Geometric (Graph) Deep Learning





Two main components operate on the graph locally:

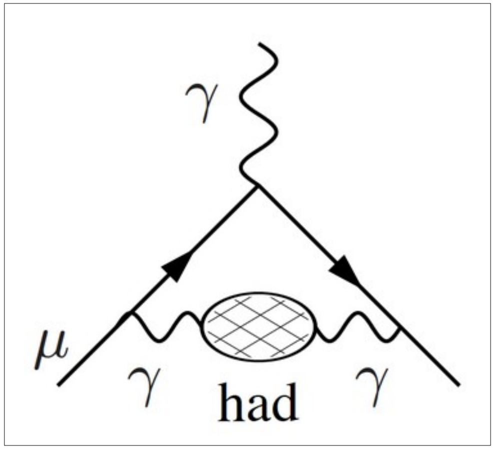
- Edge network uses the node features to compute edge weights
- Node network aggregates forward and backward node features with the edge weights and updates node features

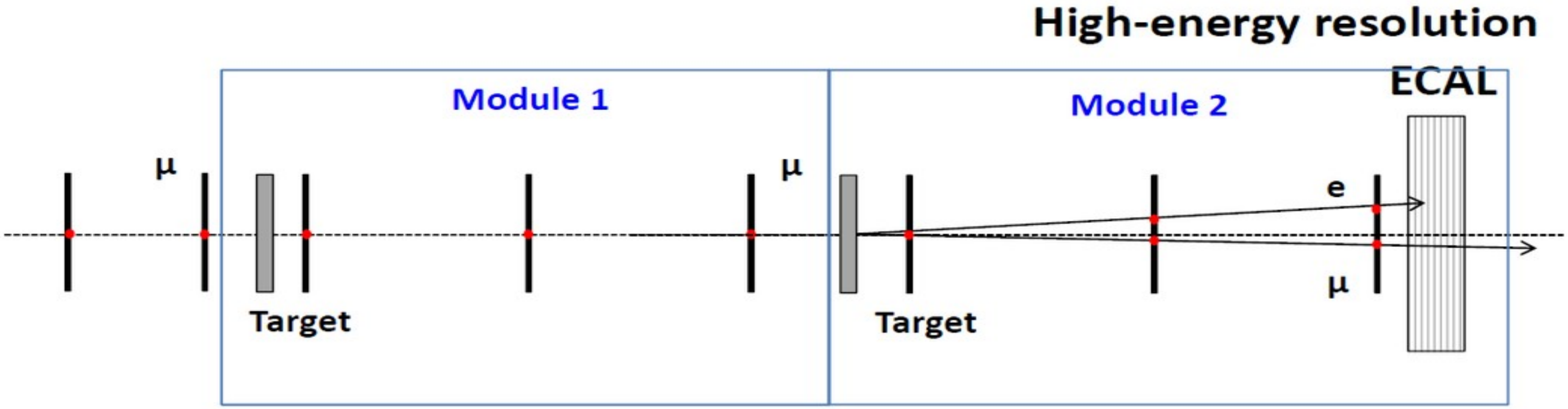
Task – GNN for MuonE experiment

- MuonE – future CERN experiment dedicated to measure a hadronic correction to the anomalous muon magnetic moment.

- **TASK: apply GNN to the simulated test beam data:**
  - No magnetic field – straight tracks
  - Few detector layers
  - Layers detecting X and Y position with so called stereo layers rotated by 45 degrees

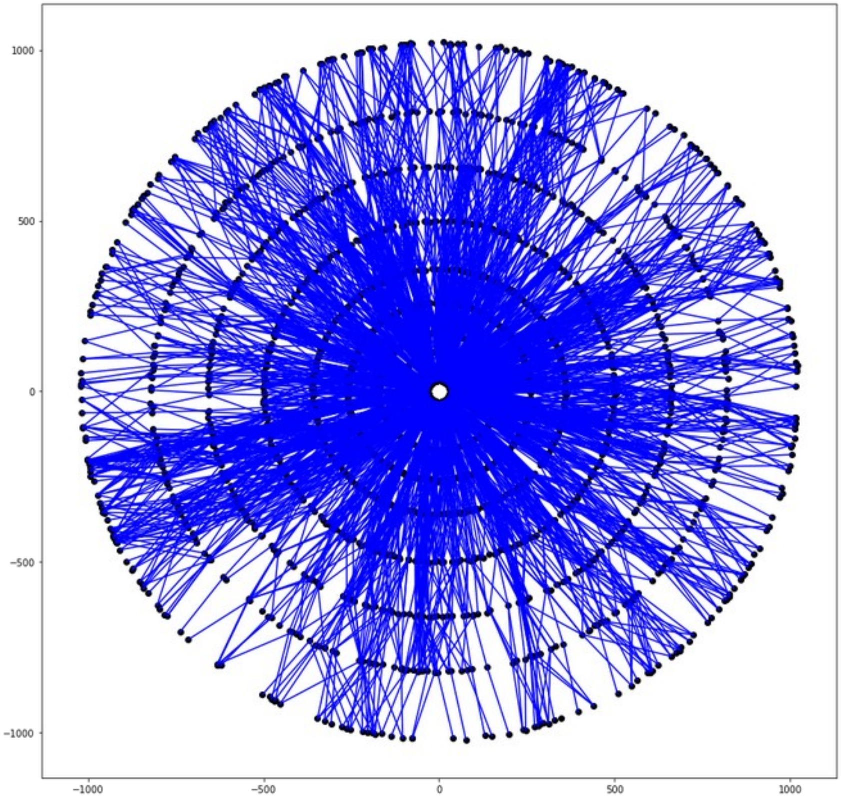
hadronic vacuum polarization





Information about Graph Neural Networks

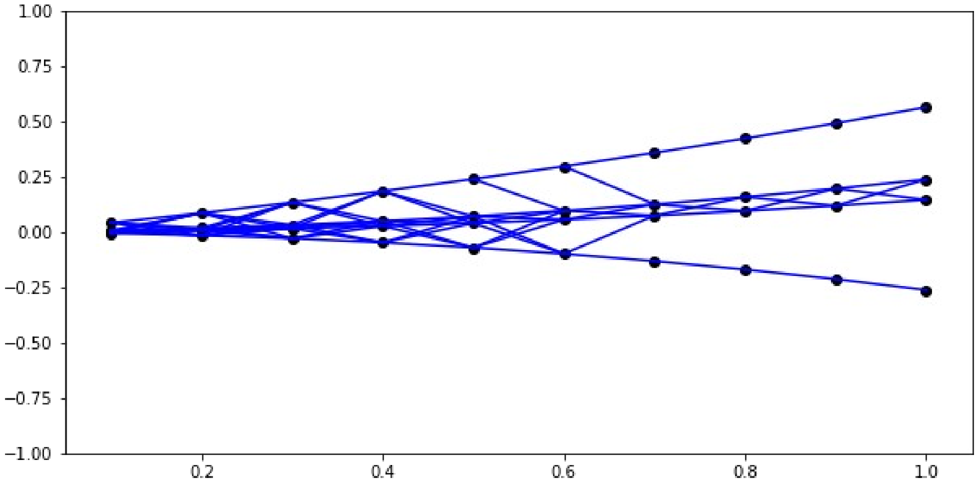
- Great page: http://geometricdeeplearning.com/
- Nice lecture: https://ucbrise.github.io/cs294-ai-sys-sp19/assets/lectures/lec03/gnn.pdf
- Presentation about GNN tracking:
- https://indico.cern.ch/event/658267/contributions/2881175/

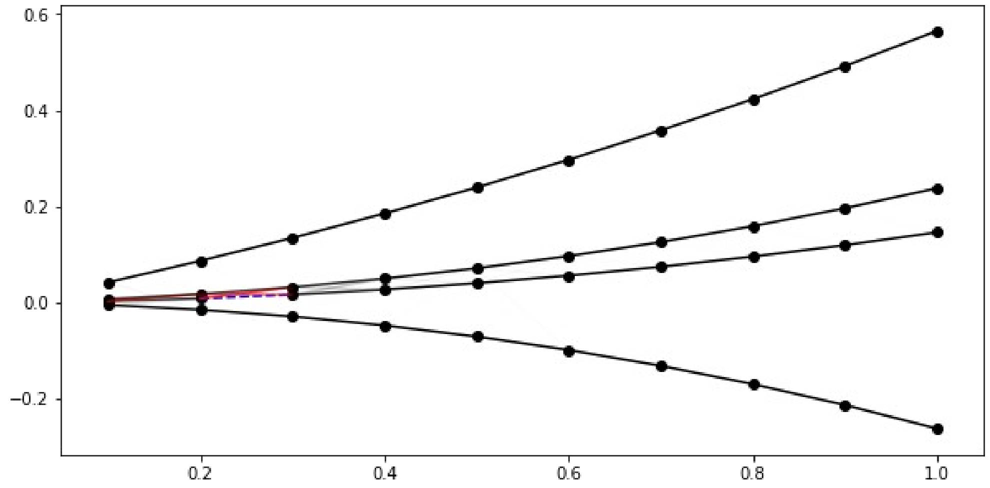


See the code of a toy example on github

- It is pretty compact:
- https://github.com/marcinwolter/Tracking_student2022/blob/master/GNN_Tutorial_Colab.ipynb
- Plus input data: https://drive.google.com/open?id=1NNZadxZcrxkm0NJv3CgeNGwBnsnByioV

And you can run it on Google Colab: https://colab.research.google.com





Repository

- There is a starting repository for you:

  https://github.com/marcinwolter/Tracking_student2022

- It contains:
  - GNN_Tutorial_Colab.ipynb - tutorial code for GNN tracking

    Data for GNN_Tutorial_Colab: https://drive.google.com/drive/folders/1NNZadxZcrxkm0NJv3CgeNGwBnsnByioV?usp=sharing
  - GNN_MuonE_v1.ipynb - starting code I have prepared for students in 2020

    Data: https://drive.google.com/drive/folders/1PpdNJHw9KYpY6kczmeGsFLRfaeCq65gK?usp=sharing
  - GNN_MuonE_v1_Robust_mc-recon_comparison.ipynb - program for tracking in 2D created by students in 2020

Your task

- Read the tutorials (we can do it together)
- Read the presentation of summer students from 2022:
  - https://github.com/marcinwolter/Tracking_student2020/blob/master/PPSS_Presentation.pdf
  - Try to understand the code GNN_MuonE_v1.ipynb
  - Think how to make pattern recognition
- Students in 2020 created the code working in 2D doing:
  - Pattern recognition using GNN
  - Robust fit to the hits to find the track parameters

Your task

- There are so called stereo layers tilted by 45 deg – they allow reconstruction in 3D
- Think, whether we can do reconstruction using three types of layers at once:
  - X layers
  - Y layers
  - Stereo layers