Tuesday 04 June 2019

Session 1: Room 15-300: Looking forward by looking back - I (09:00-12:00)

Distributed ML on HPC VLIMANT, Jean-Roch; FARRELL, Steven

- Batch size critical for Data Parallelism. Serial batch size for trackml models is 1 event. Tuning learning rate to avoid instabilities. Critical batch size depends on gradient noise scale (slide 17). We haven't measured any of this for our GNN models
- Cray ML plugin introduces gradient lag (sync but delayed)
- *Joosep: how big our problems are in terms of parameters, inputs, outputs, data size - how much do we need to scale up?*
- Steve: async training (mpilearn) may be a good match for irregular problems like GNNs
- NESAP project on distr ML for GANs (Ben, Jean-Roch)
- Other solutions for HPC workflows (distributed training and/or HPO)
    - Ray: https://rise.cs.berkeley.edu/projects/ray/
    - Ray Tune: https://ray.readthedocs.io/en/latest/tune.html
    - https://ecp-candle.github.io/Candle/html/index.html
    - Cray HPO (no good link yet): https://www.cray.com/blog/how-hyperparameter-optimization-improves-machine-learning-accuracy/

Tracking with neutrino detectors USHER, Tracy

- Metrics: need to get vertex position (including secondaries) and particle ID for all partcles coming out of vertex right. Efficiency more important than purity in spp formation.
-

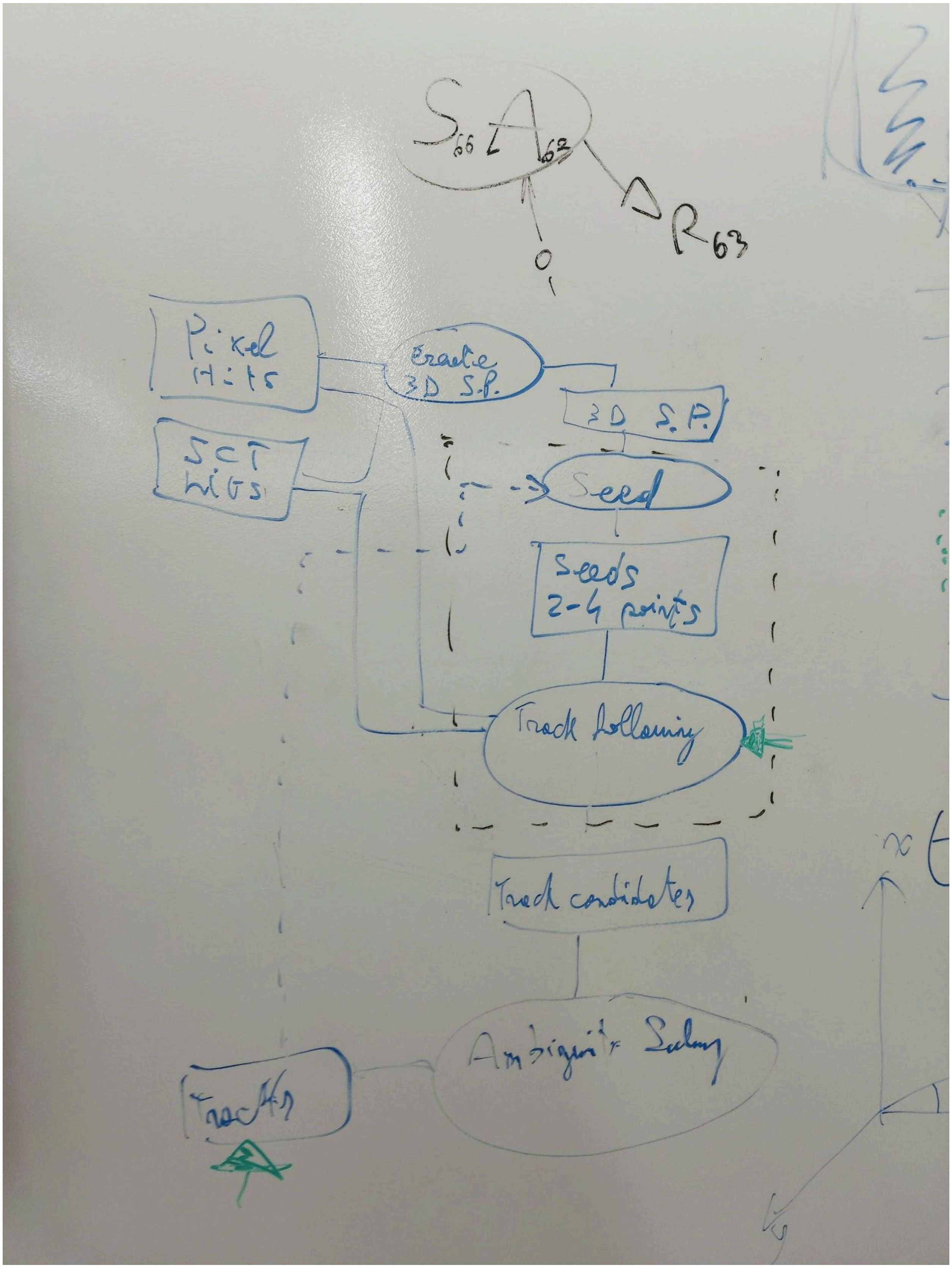
ML algorithms for neutrino detectors TERAO, Kazu; WONGJIRAD, Taritree

-
- LArFlow find transformation that matches track image in Y plane with U and V plane
- Downsampling in time x6 to fit in memory
    - Sparse convolutions will fix that.
- Cropping planes with overlap
- LArFlow goal is to match hits in source and target views and build spp out of matching hits.

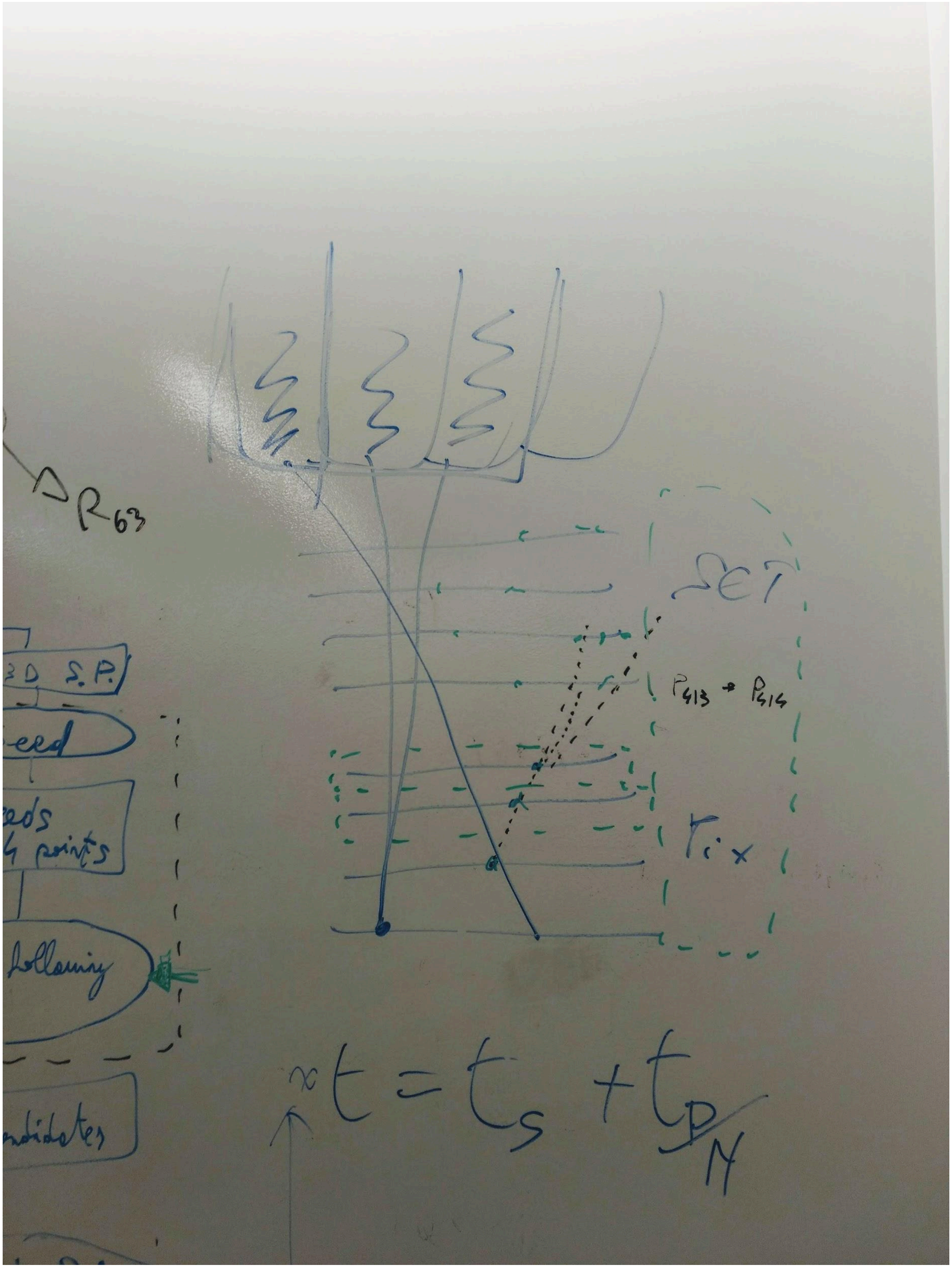
- Formulated as regression problem, with the matching done by some post-processing. Could instead treat it as pairwise classification, and can represent as graph
- Kazu
- Targeting Icarus in this project: 6x microBoone
- DUNE Near detector. Pixel, Optical and TPC measurements can be combined to reject pileup
- Goals of ML algorithm. Cluster 3d points by particle, and by interaction vertex
- U-ResNet PPN Classify each 3d point by particle type + other features
- Horovod sync GD
- Distributed training useful to speed up dev cycles (4hrs→ 0.5hrs) significant
- How do you define precision and efficiency for your end-to-end algorithm and piece-by-piece

Session 2: Room 59-4101: Looking forward by looking back - II (13:00-17:30)

HL-LHC Tracking CERATI, Giuseppe; PAGAN-GRISO, Simone



Pixel Hits
SCT Hits
Create 3D S.P.
3D S.P.
Seed
Seeds 2-4 points
Track following
Track candidates
Ambiguity Solving
Tracks



3D S.P.
eed
eeds
4 points
candidates
SET
P413 + P414
t = ts + tD/N

- Track following done of a simplified “tracking geometry” with O(100) volumes.
- Track following unlike seeding uses raw measurements (hits).
- Ambiguity solving different ATLAS/CMS
    - CMS approach iterative. ~9 iterations. ATLAS ~1 pass. Harder Ambig solving
- Most of ATLAS tracking inherently iterative, hence serial
-

Recent HEP.TrkX Results JU, Xiangyang

- New feature: edge selection with 90 NNs that take hits global coordinates, sensor cluster widths, and sensor thickness as inputs, and classify a doublet as good or bad.

Wed Jun 5

HEP.TrkX to Exa.TrkX, Paolo

- Should convert table to calendar dates
- Match deliverables with NERSC plans
- D2 Exa. TrkX for LArTPC … primarily DUNE ND interaction clustering as a target goal
- D3 Exa. TrkX for LArTPC … consider gap, pixel size, and readout deadtime (current concerns on detector designs for DUNE ND)
- Define inputs, reference solution and metrics to compare different results
- Datasets: Initially trackML & “LARML”, but need to go to experimental datasets by 2021
- Make sure open data from experiments is available for the tracking challenge for 2021
- Metrics: either integrated evaluation in experimental software or upfront work to develop a metrics toolbox on TrackML

Next steps following HEP.TrkX Results

- Doublets to triplets
- Missing hits
- NERSC Big data center contribution:
    - Sparse matrices, pytorch_geometric
    - Cray load balancing
- Training time Xiangyang 1 hour, two epochs gives good result Volta, 7K events, 100K samples (16 sub-events)

Prototype to Production

- Efficiency benchmarks
    - muon (99.5% efficiency above 10 GeV)
    - “Pion” 80-90% efficiency. Inefficiency from hadronic interactions

- Electrons

Metrics (MC)

- separate to Ptype
- efficiency → absolute, → algo

$$\frac{N_{track}(\text{matched})}{N_{sim}(\text{cut})}$$

- fake rate

$$\frac{N_{track}(\text{NOT matched})}{N_{track}}$$

- duplicate rate

$$\frac{N_{track}(\text{multiple matches})}{N_{track}}$$

- matching → hit fraction > X, → parameters

$$\frac{\#\text{hits from state}}{\#\text{raw}\ (\#\text{sim})}$$

- pulls $\frac{\Delta X}{E(X)}$
- resolution σ_x

Met…

- Top 4…
- Exclu…
- Bottom…
  + M…
  -
  -
  -
- ?
  + …
  + D…
  + A…

Metrics (data)

- Tag & Probe
- Exclusive hadr. decays

Z

Direct
In-situ

- Bottom-up approach
  + Material estim.
    - Eng.
    - in-situ.
      + hadr. int.
      + γ conv.
    - Design.
- Robustness
  + B field
  + Detector efficiency
  + Alignment

Metrics (MC)	Metrics (data)
Absolute efficiency (compared to truth) Algo efficiency (assuming upstream I am perfectly efficient By particle type	In-situ (data driven) ● Tag & probe ○ Use known resonance (z→ mumu) ● Exclusive hadron decays (compare BRs with measurements from precision experiments)

Fake rate (1- purity)	Bottom-up approach: check goodness of simulation
	Need material estimation
	● From engineering drawing
	● Data-driven (in-situ)
	    ○ Hadronic interactions
	    ○ Gamma conversions
	Can go very detailed e.g. when checking how MC reproduces clusters in pixel detector coming from multiple adjacent tracks

Duplicate rate (truth particles matching multiple tracks) Matching defined ● as #reco hits matching (track following) or ● as track parameter matching (used for track fitting & selection)	
Pulls (dx/Err(x) for track parameters	
Resolution (sigma X)	

Algorithm flexibility: tune algorithm behavior to different use cases. E.g. some CMS iterations are specific to one physics channel or data set (low lumi, cosmic, no b-field) . Moving to production requires the ability to tune algorithm quickly.

Robustness against variability (b-field, missing layers, channels, alignment).

Next Steps for Intensity Frontier

Icarus SBN or Proto DUNE	DUNE ND	
Approval policy	Open data unknown publication policy	
3D point matching -Giuseppe Collaboration with Cincinnati postdoc active.	Detector design is still moving target (CDR end of August). Kazu is one of those controlling simulation.	
3D clustering - Kazu ● Interaction ● Particle	Can we handle pileup and do interaction clustering.	
	Goal is to have Particle clustering and interaction clustering prototype by end of summer.	This would impact detector design directly

General issue with publishing results using experiment data

Open questions

What role can NNLO play compared or combined with e.g. naked Horovod for our project

Can matchability/flow be applied to rho/Z and rho/phi views @ colliders?

Match optical+pixel+wire information for Near detector

Sparse CNN / GraphNet applicability
Ambiguity solving as a classifier?

Maria tensor decomposition on TPU. Can this be applicable to pattern recognition? mkFit Kalman Filter as matrix multiplication

Next six months

Caltech JR, Joosep + three summer students. Standalone classic tracking algorithm to run on trackml. Possibly in collaboration with Andi Salzburger/aCTS
CA seeding → Build GNN → use GNN weights to reduce KF combinatorics

GNN with Helix transport

Question: how to compare edge selection from Alexander Gaussian Processes with Xiangyang 90 NNs

LBL
Steve will continue with distributed learning

Xiangyang
Interested in trying GNN on ATLAS data. Technical issues aside, the problem is how to share results among ourselves

Conferences
Giuseppe PASC 19 Tracking on GPUs will mention ExaTrks
School @ Institute Pascal Oct 14
IPDPS paper submission October.