Distributed training of GNN on HPC

Xiangyang Ju

Lawrence Berkeley National Lab

4th IML workshop, 21 October 2020

BERKELEY LAB
LAWRENCE BERKELEY NATIONAL LABORATORY
Physics Division
U.S. DEPARTMENT OF ENERGY

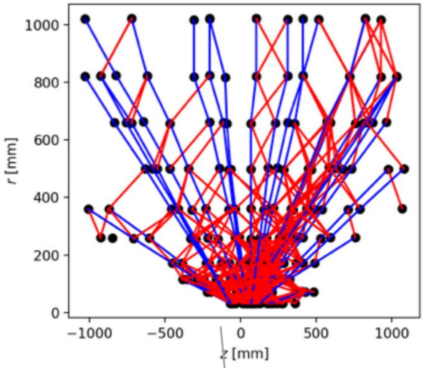
GNN & HEP

- Graph Neural Network (GNN) has seen many interesting applications in the HEP community and the impact of GNN on High Energy Physics can only grow further in the future.

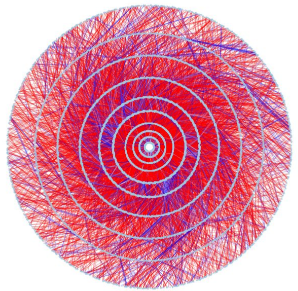
- In the meantime, the colliders and experimental detectors are becoming larger and more sophisticated in order to explore new phasespace.
  - Leading to larger graph sizes
  - Plots in the right are an excellent example

- Graph is irregular, sparse and dynamic in its size. It composites non-trivial computational challenges, including training GNN. To cope with grown graph size, we must explore different computing resources.

- We take the GNN used in [arxiv:2003.11603] for tracking reconstruction as a benchmark model to explore **distributed training in High Performance Computers**

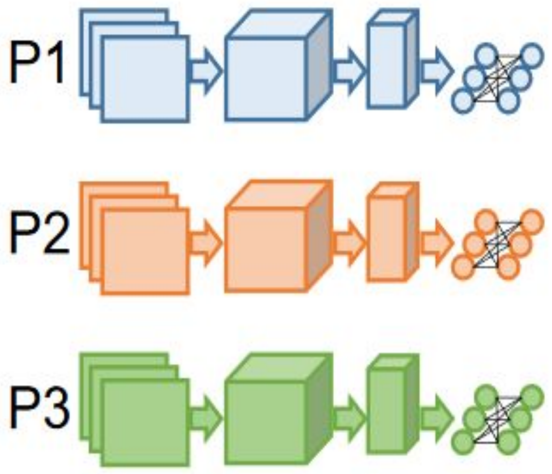


Project to future



Distributed training strategy

Performing data parallel distributed training:
Same model is replicated to different devices (GPUs, TPUs), different data are sent to devices for training, gradients are averaged among devices to update the weights



arxiv:1802.09941

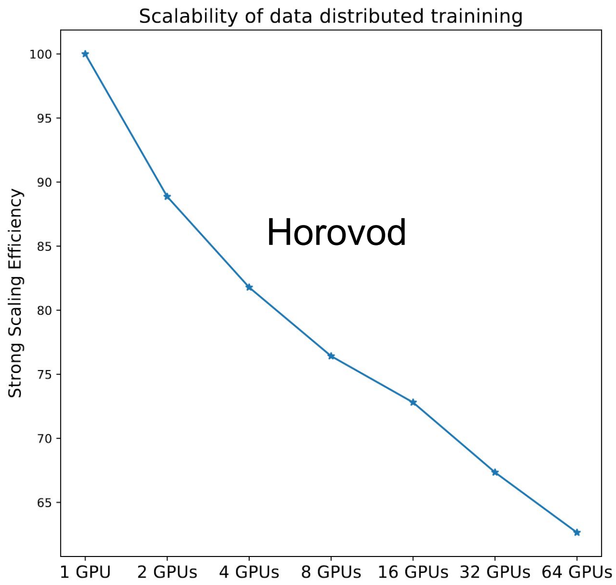
Two implementations:

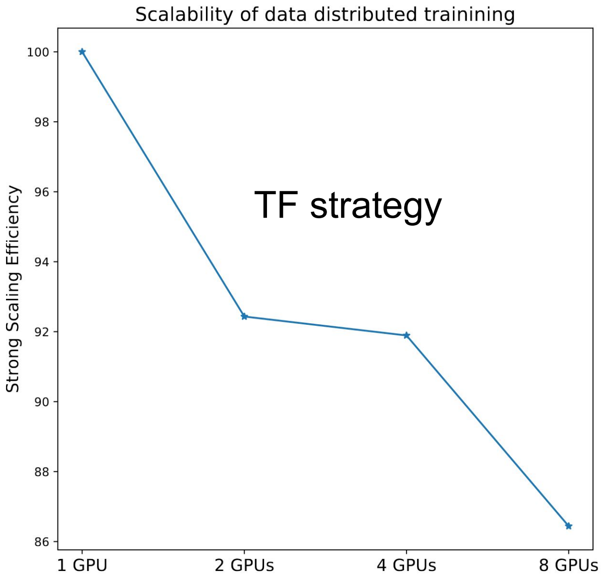
1. Horovod [link]
2. Distributed strategy in TensorFlow [link]

GPU: Nvidia V100, 14 TFLOPS in fp32, 16 GB memory.

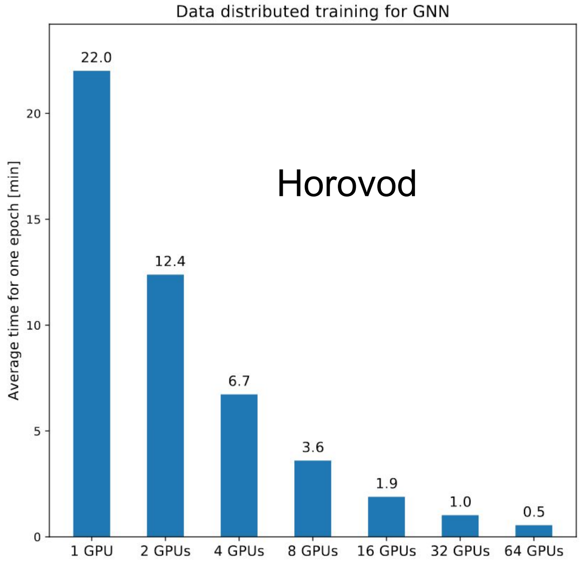
Scalability

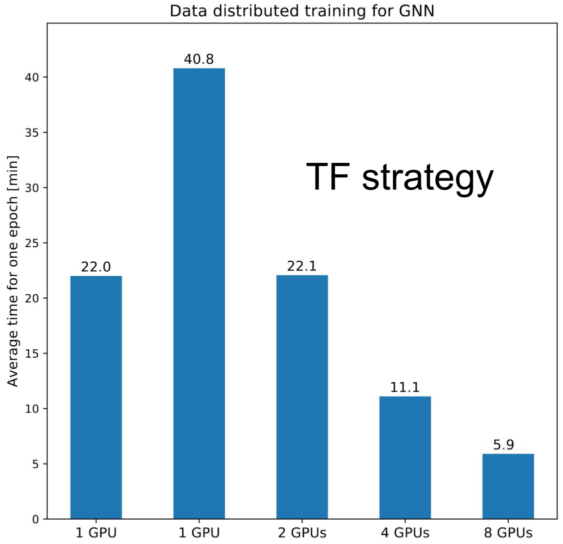
Strong scaling efficiency = $\frac{t_1}{N \times t_N} \times 100\%$ where *N* is number of devices





Time for each epoch





We measured many other metrics: precision, energy cost per epoch, dollar per epoch, ….
We also compared GPU with TPUs. A public doc is in preparation.
Stay tuned.