Distributed Machine Learning on Summit

Machine Learning for ATLAS and beyond

Abid Malik
Machine Learning Group
Computer Science and Mathematics Dept.

Computational Science Initiative

Amir Farbin

University of Texas (Arlington)

Sergey Panitkin (BNL- ATLAS)

Jean Roch (Caltech)

BROOKHAVEN NATIONAL LABORATORY

U.S. DEPARTMENT OF ENERGY

Motivation

- **Machine learning techniques, especially deep learning, are fast becoming part of High Energy and Nuclear Physics**
- **Simulation code, Data analysis:**
  - **50% performance gain on LHC running**
- **ATLAS and LHC will generate huge data and this can not be handled using single node training mode**
- **Software stack for machine learning that would help port machine learning algorithms across different hardware**

PanDA System and DNNs

- How the PandDA ecosystem can handle DNN applications as part of the HEP workflow?
- How PandDA can facilitate the training process of DNNs especially with respect to improving its performance?
- Distributed Learning Working Group
  - Bi-weekly meetings

Single Model Single Node DNN training

- **Simple Case: One model per accelerator**
- **Hyper-parameter optimization:**
  - **Tuning DNNs' parameters to improve the performance**
  - **$O(n^6)$ depending on what kind of model is being used**

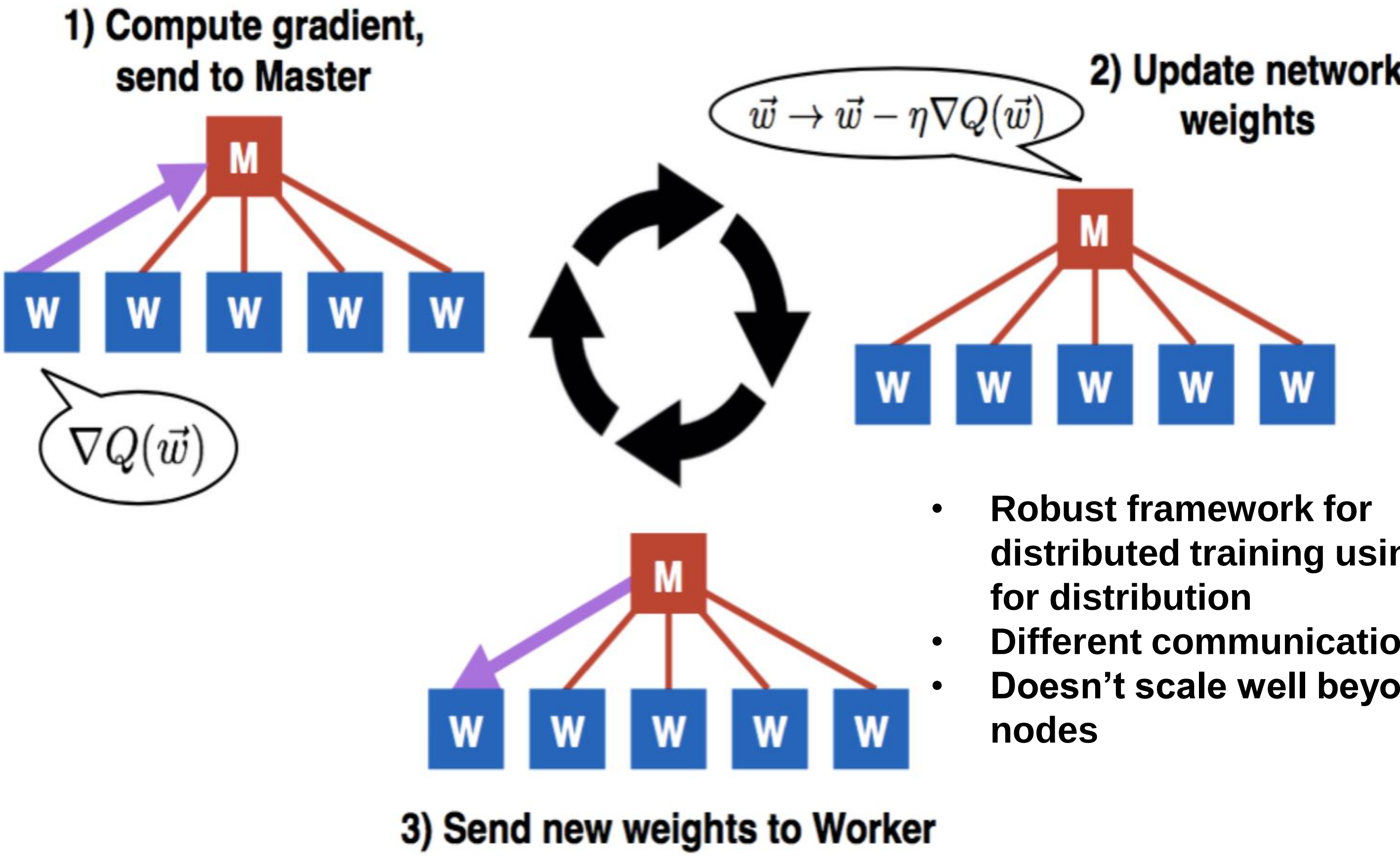
Single Model Single Node DNN training

- CERN has done experiments for hyperparameter optimization
- Using containerized software stack
- Using PanDA to distribute the models with different parameter across different nodes
- Doing linear search to find the best model
- Feedback is good
  - Results are not available
  - PanDA is useful for such kind of problems

Distributed Learning using more than one GPUs

- Need it to manage huge data and reduce training time
- HEP community is already using it:
  - MPI_Learn (from CERN)
  - Horovod (from Uber)
  - LBANN (Livermore Big Artificial Neural Network) Toolkit

MPI_Learn Framework

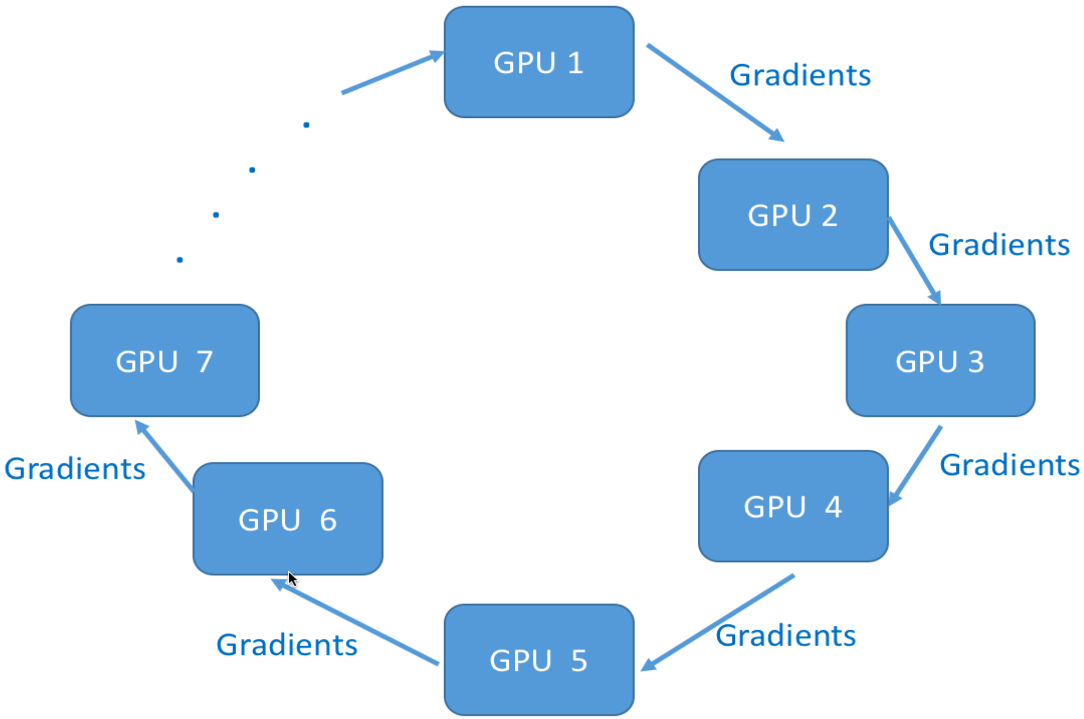


- **Robust framework for distributed training using MPI for distribution**
- **Different communication models**
- **Doesn't scale well beyond 600 nodes**

U.S. DEPARTMENT OF ENERGY
BROOKHAVEN NATIONAL LABORATORY

Horovod Framework

- Established itself as robust framework for distributed learning
- Good scaling beyond 400 nodes
- Uses NCCL2 and cuda-aware MPI

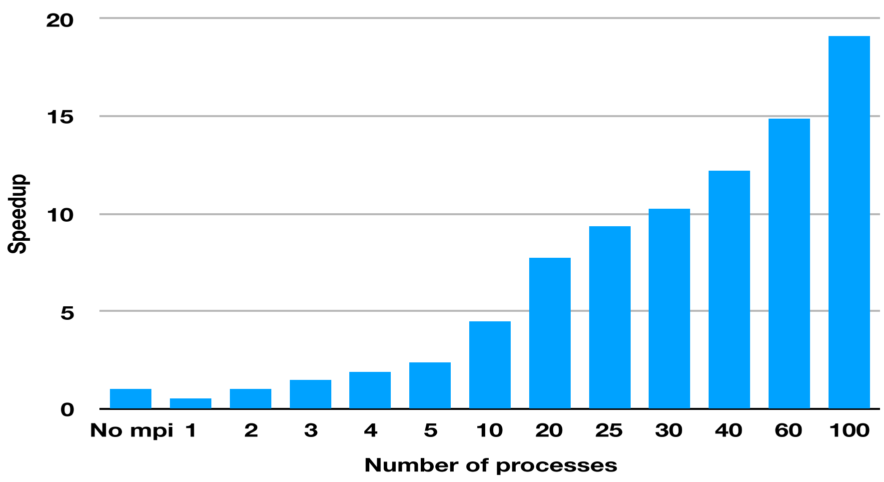


Detailed Performance Analysis of 3D GANs (Generative Adversarial Networks) on Summit

- Using MPI_Learn and Horovod for distributed learning
- Using data from CERN
- Able to run with 1200 GPUs (V100) on Summit

Scalability Results

- MPI_Learn scales linearly
- Used only one epoch for analysis
- Horovod shows better results

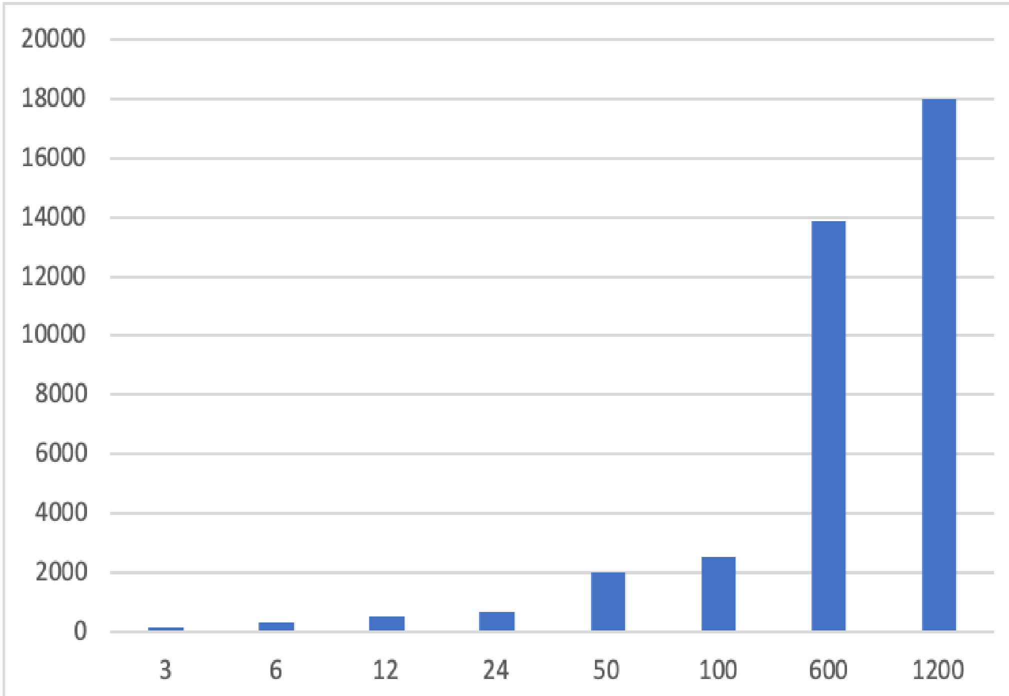


Scalability of MPI_Learn on Summit

Running 3D GANs on Summit MPI_Learn

- Through put performance analysis
- The communication cost becomes very high after 600 GPUs
- Need to optimize the communication model
- NCCL and CUDA-AWARE MPI is the first step

Time in seconds



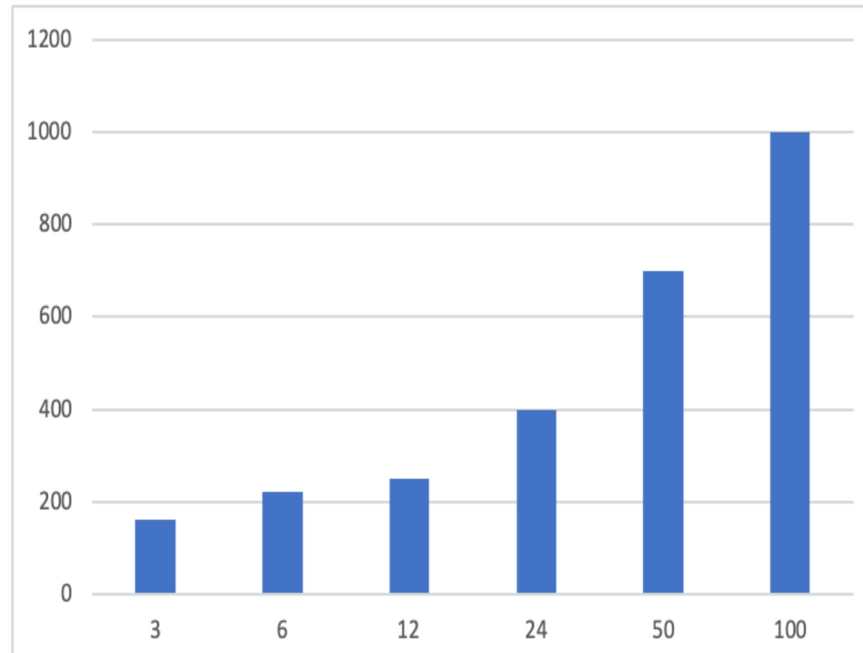
Number of processors

MPI_Learn performance

Running 3D GANs on Summit using Horovod

Time in seconds

- Through put performance analysis
- Better performance than MPI_Learn
- NCCL and CUDA-AWARE MPI



Number of processors

Horovod performance

Detailed Performance Analysis of 3D GANs on Summit

- Inter node and intra node performance
- Computational and communication cost
- Resource utilization
- GANs computational and communication characteristics
- Scalability
- Accuracy and throughput performance
- Variables that control the search space for hyper-parameter optimization

Detailed Performance Analysis

- Horovod is giving better results in terms of through put
- Accuracy needs to be looked into
  - Checkpointing
  - Not enough data
- Score-p Vampire toolkits for more detailed analysis
- Manual instrumentation of the code

Work in progress

- Debugging the code
  - Frameworks are breaking
- Check pointing for 3D GANs
  - For accuracy analysis
- More data for scalability
- Manual instrumentation of python code

Possible next step

- Advanced GANs architecture from HEP (CERN)
- Real data to test the scalability and accuracy
- Running and simple Hyperparameter optimization for 3D GANs using PanDA
    - Linear scanning for the best model

Summary

- Distributed learning is important for applying ML techniques in HEP
- We need to look into the distributed frameworks for distributed learning
- Hyper-parameter optimization is an important problem
- The PanDA ecosystem can play an important part in distributed learning
- Detailed performance analysis of Distributed 3D GANS
    - Interest from the other groups (NERSC ML group, MLPerf HPC group)

Thanks

- Questions!