
Weiyang (Frank) Wang, MIT CSAIL

Moein Khazraee, Zhizhen Zhong, Manya Ghobadi, Zhihao Jia, Dheevatsa Mudigere, Ying Zhang, Anthony Kewitsch
The growth of large DNN models creates demands for efficient distributed DNN training systems.
• Fat-Trees provide uniform bandwidth and latency between server pairs

• Ideal when the workload is unpredictable and consists mostly of short transfers

• Fat-Tree networks are not the best network topology for DNN training!
Network is becoming a bottleneck of DNN training

- Fat-Tree based DNN training infrastructures are facing a network bottleneck
  - Network Bottleneck: the amount of time spent on communication only
Previous work on distributed DNN training optimization does not consider physical topology

- Compression and encoding
  - Qsgd [NeurIPS ’17]

- Asynchronous transmit
  - DC-ASGD [PMLR ’17]

- Collective communication
  - BytePS [OSDI ’20]

- Schedulers
  - Themis [NSDI ’20]

- Parallelization strategy
  - FlexFlow [MLSys ’19]

- Hyper parameters
  - ASHA [MLSys ’20]

- Network topology
  - ?
Reconfiguring physical network topology
Reconfiguring physical network topology

Topology A

Topology B

Topology C
DNNs training traffic has different properties
DNNs training traffic has different properties

• Key observations:
  1. Traffic patterns are predictable, and do not change across training iterations
DNNs training traffic has different properties

- Key observations:
  1. Traffic patterns are predictable, and do not change across training iterations
  2. Traffic patterns are model-dependent
TopoOpt

The first system to leverage reconfigurable network, to co-optimize network topology and parallelization strategy for distributed training

TopoOpt achieves 3.4x faster training time for DNN training
Co-optimization challenge: Huge search space for optimal DNN training

- The configuration space is huge!

Search space explodes!

Missing potential solutions!

DNN Parallelization Strategy

Network Topology & Communication
Alternating optimization framework to co-optimize DNN parallelization strategy and network topology

**Strategy Optimization**
- Parallelization Strategy Search

**Topology Optimization**
- Traffic Demand Extraction
- TopologyFinder Algorithm

Parallelization strategy

Topology and routing
Alternating optimization framework to co-optimize DNN parallelization strategy and network topology

What algorithm should we use to find the topology in this framework?
Characteristics of DNN training traffic

- Model Parallel Transfers
- AllReduce Transfers
Challenge: finding a good network topology for both AllReduce and Model-Parallel transfers

- Degree (d) = 3, unidirectional

8 hops!
Meeting the requirements of both AllReduce and Model-Parallel transfers

- Degree \((d) = 3\), unidirectional

<table>
<thead>
<tr>
<th>Transfer Type</th>
<th>Characteristics</th>
<th>Network Requirement</th>
</tr>
</thead>
<tbody>
<tr>
<td>AllReduce Transfers</td>
<td>Large, Sparse</td>
<td>Ample Bandwidth</td>
</tr>
<tr>
<td>Model Parallel Transfers</td>
<td>Small, Dense</td>
<td>Low hop-count</td>
</tr>
</tbody>
</table>
Key idea: **mutate the traffic matrix**

AllReduce transfers are **mutable**. Model-Parallel transfers are not mutable.
Leverage the mutability of AllReduce transfers to achieve high bandwidth for AllReduce & low hop-count for Model-Parallel!
Key technique: Regular permutations

- $n$ total accelerator, each with degree $d$

Regular permutations - every server connects to another one with a fixed distance $\delta$

Irregular permutations

$O(n!)$ different permutations
Key technique: Regular permutations

- $n$ total accelerator, each with degree $d$

- The possible set of $\delta$ are the positive integers less than $n$, such that $\gcd(\delta, n) = 1$
  - $\rightarrow O(n)$ search space!

- Among all possible $\delta$ distances, choose a set of them within the degree to minimize the cluster diameter

- The technique of permuting labels works for other AllReduce algorithms as well

TopoOpt bounds the cluster diameter to $O(d \cdot \sqrt[4]{n})$
TopoOpt uses optical switches
TopoOpt uses optical switches

- Fully functional 12-node, degree 4 testbed integrated with NCCL
TopoOpt uses optical switches

- Fully functional 12-node, degree 4 testbed integrated with NCCL
Evaluation

• We evaluate TopoOpt with large scale simulation and a small-scale prototype
• Artifact code can be found at [http://TopoOpt.csail.mit.edu](http://TopoOpt.csail.mit.edu)
Simulation – tail completion time

- Running several jobs together on a 432 node, $d = 8$, 100Gbps TopoOpt system, compared to several other options

TopoOpt achieves up to $3.4x$ faster 99%-tile latency compared to cost-equivalent Fat-trees
TopoOpt: the first system to co-optimize DNN training with demand-aware network topology

Leverages the mutability of DNN training traffic to search and construct the best topology

Achieves up to 3.4x faster 99%-ile training iteration time compared to cost equivalent Fat-trees