HiDL
High-Performance Deep Learning

MVAPICH
MPI, PGAS and Hybrid MPI+PGAS Library

Hy-Fi: **<u>Hy</u>**brid **<u>Fi</u>**ve-Dimensional Parallel DNN Training on High-Performance GPU Clusters

Presentation at MUG '22

<u>Arpan Jain</u>

Network Based Computing Laboratory (NBCL)

Dept. of Computer Science and Engineering , The Ohio State University

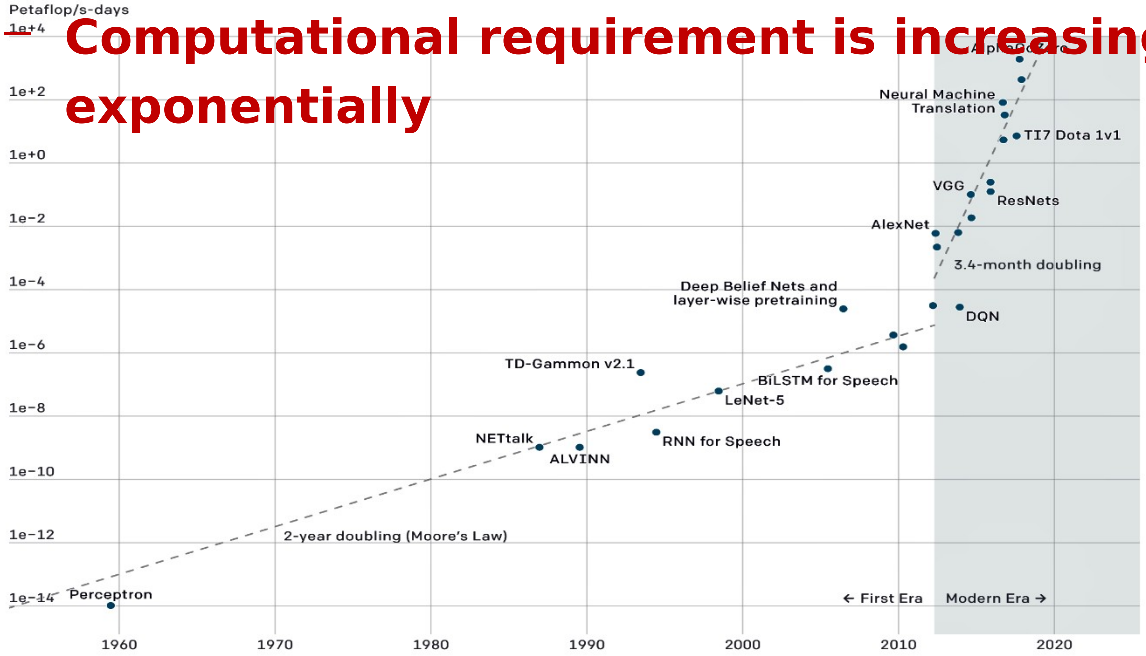
jain.575@osu.edu

Agenda

- **Introduction and Motivation**
- Problems and Challenges
- Key Contributions
- Performance Evaluation
- Conclusion

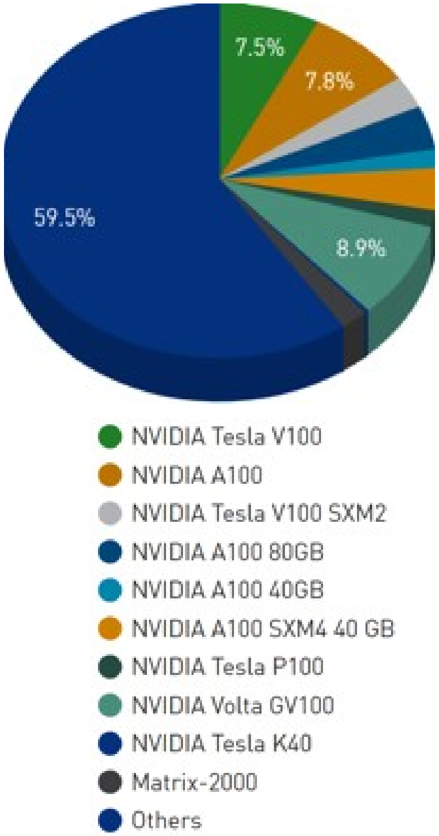
Deep Learning meets Super Computers

- NVIDIA GPUs - major force for accelerating DL workloads
  - **Computational requirement is increasing exponentially**



Courtesy: https://openai.com/blog/ai-and-compute/

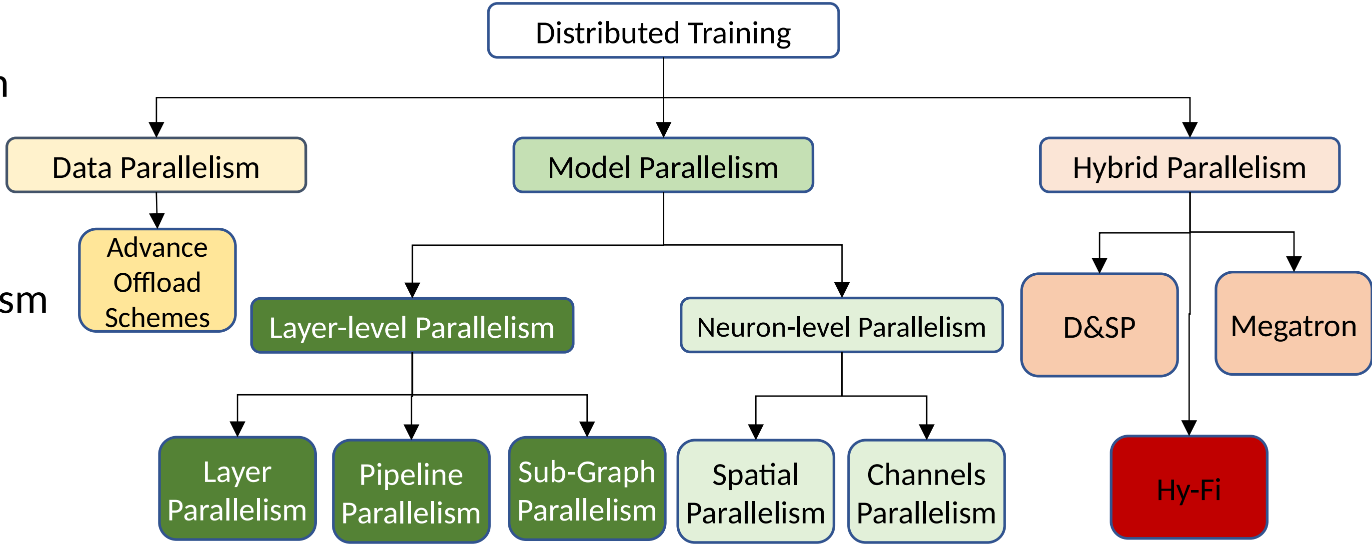
Accelerator/CP Family Performance Share



www.top500.org

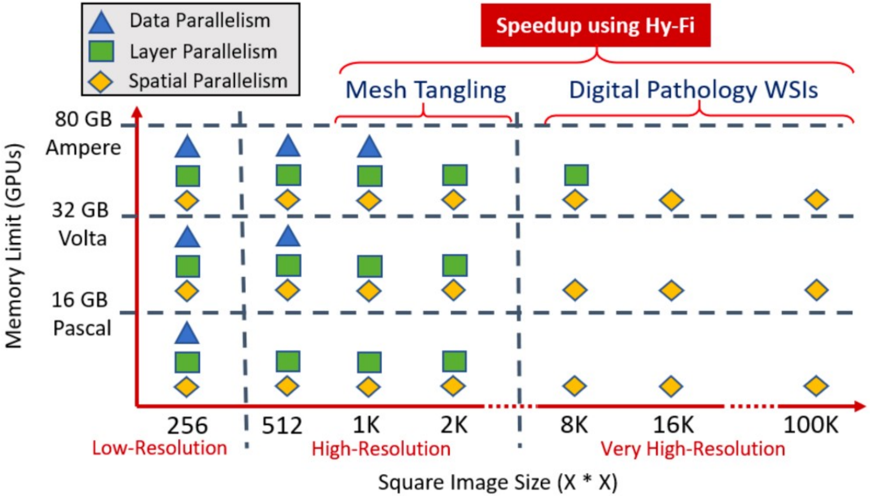
Parallelization Strategies

- Data Parallelism
- Model Parallelism
  - Layer-level Parallelism
    - Layer
    - Pipeline
    - Sub-Graph
  - Neuron-level Parallelism
    - Spatial
    - Channel
- Hybrid Parallelism
  - D&SP
  - Megatron
  - **Hy-Fi**



Why Hybrid Parallelism?

- Data-Parallelism– only for models that fit the memory
- Out-of-core models
  - Deeper model Better accuracy but more memory required!
  - Real-world applications have very high-resolution images
- Model parallelism can work for out-of-core models!
  - Performance is questionable!
  - Layer-parallelism is not enough



Agenda

- Introduction and Motivation
- **Problems and Challenges**
- Key Contributions
- Performance Evaluation
- Conclusion

Research Challenges

Challenge-1: Halo Exchange in PyTorch

Challenge-2: Exploitation of different parallelism dimension

Challenge-3: Scaling Integrated Hybrid Training Solutions

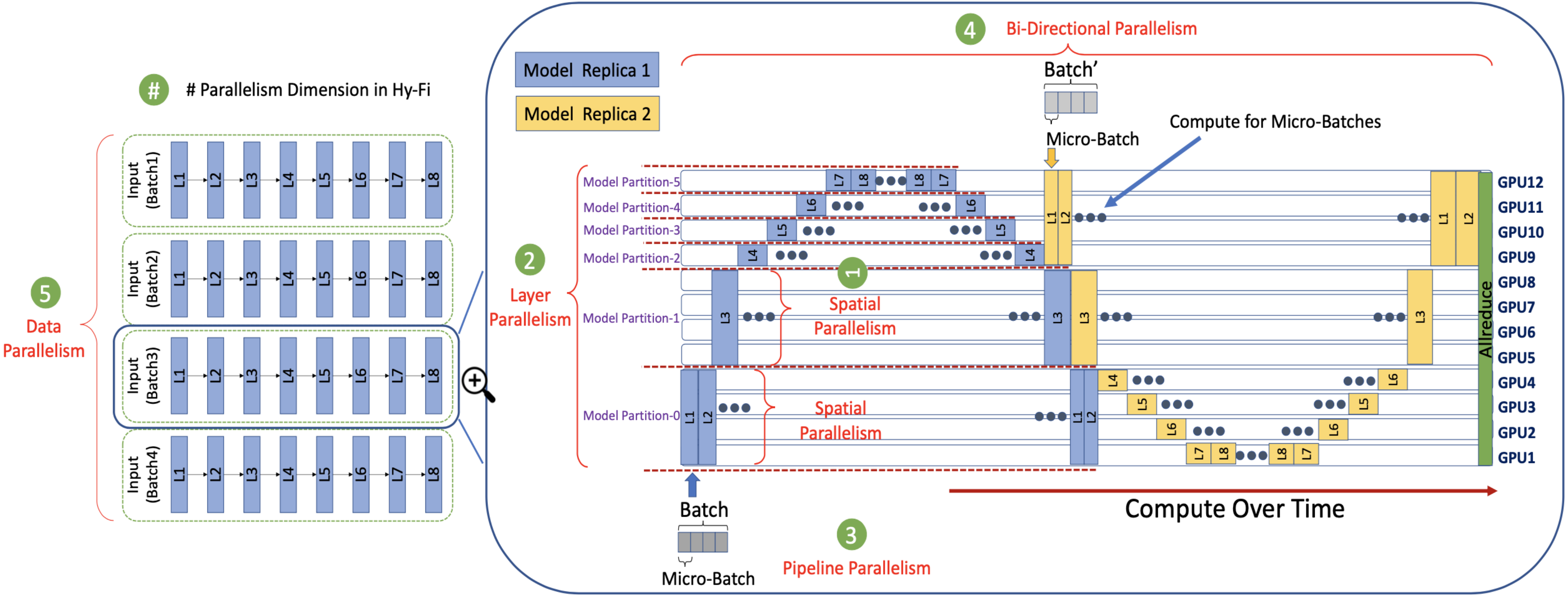
Meet Hy-Fi!

Agenda

- Introduction and Motivation
- Problems and Challenges
- **Key Contributions**
- Performance Evaluation
- Conclusion

Proposed Hybrid Five-Dimensional Parallelism (Hy-Fi)

- Integrates spatial, layer, pipeline, bi-directional, and data parallelism



Strategies to Optimize Halo Exchange

- Halo Exchange
  - Needed to compute convolution and pooling operations
- Proposed Halo-D2 and explored different image distribution strategies

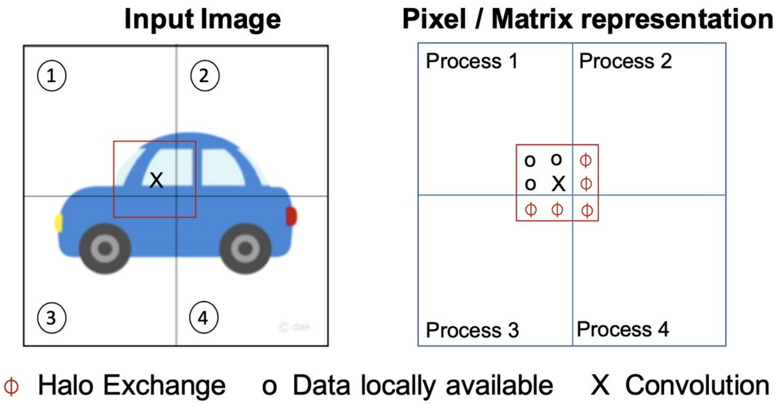
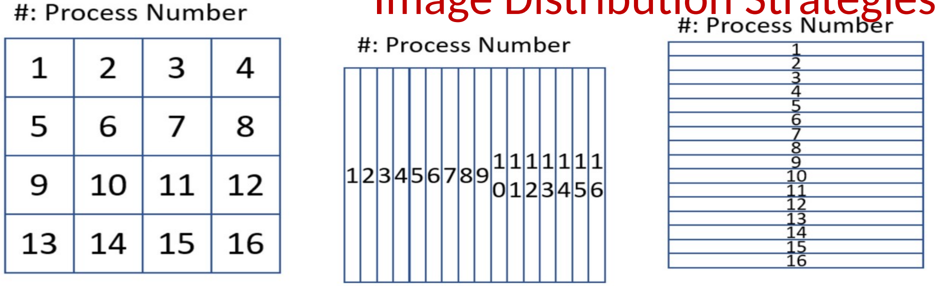
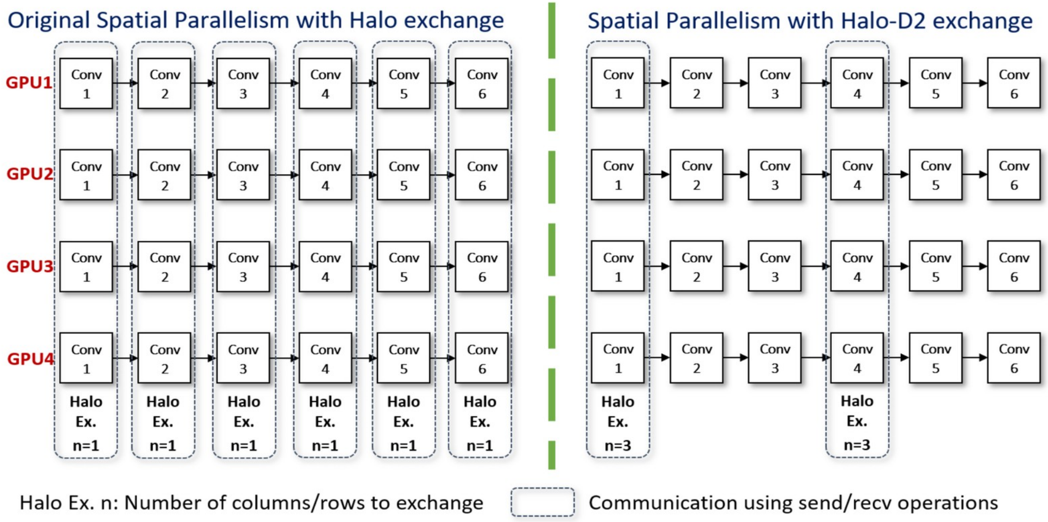


Image Distribution Strategies

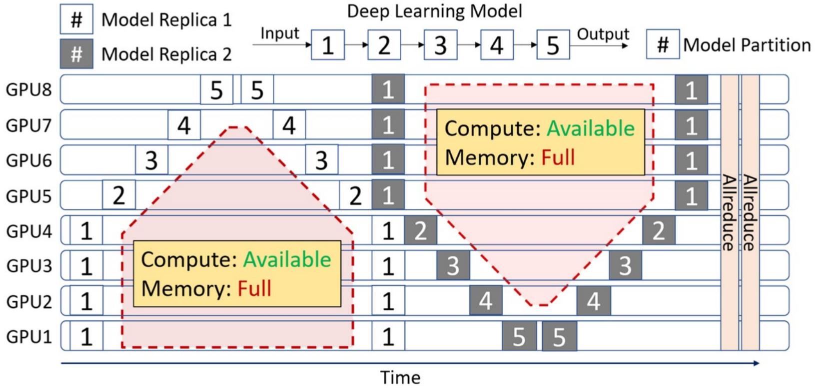


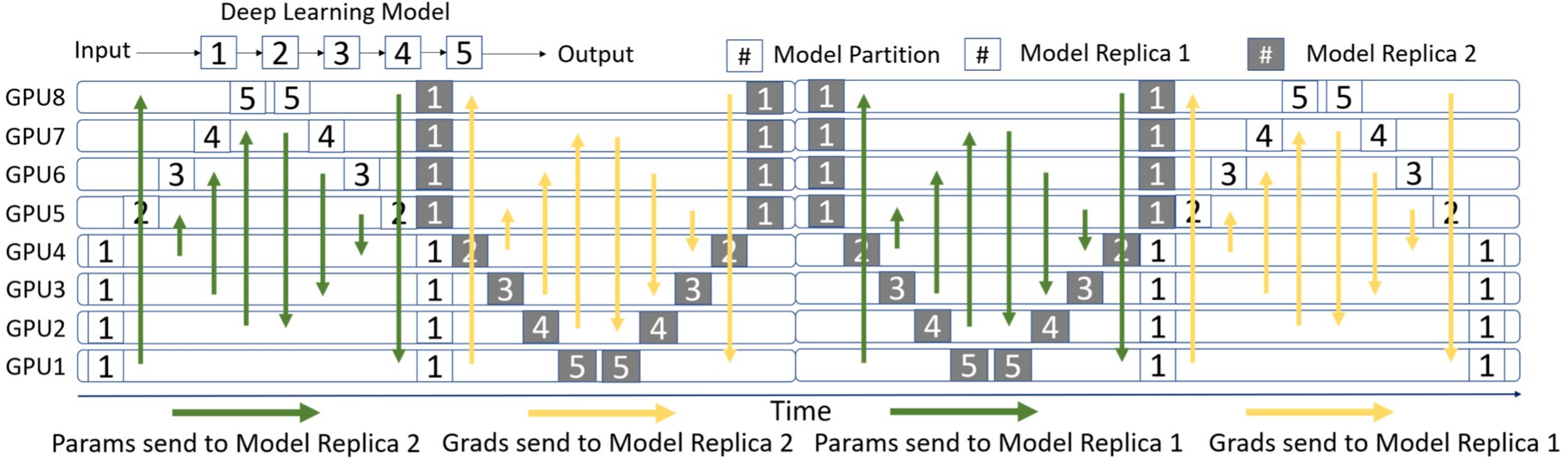


Proposed Communication OptimizationHalo-D2

GEMS-MAST vs GEMS-MASTER

- Bi-directional Parallelism
  - Hard to overlap Allreduce operation ->
- Proposed communication optimization





Agenda

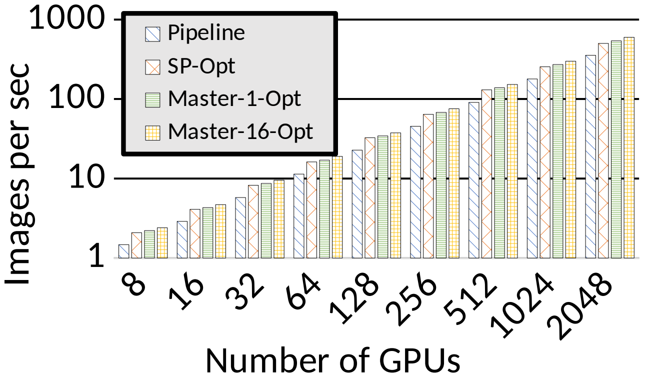
- Introduction and Motivation
- Problems and Challenges
- Key Contributions
- **Performance Evaluation**
- Conclusion

Evaluation Setup

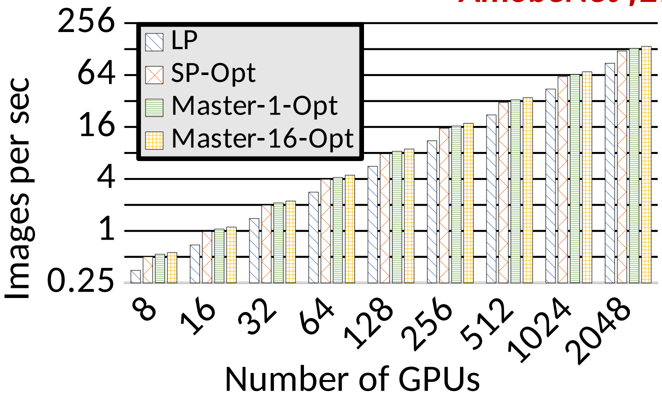
- System
  - Lassen at Lawrence Livermore National Laboratory (LLNL)
    - POWER9 processor
    - 4 NVIDIA Volta V100 GPUs per node
- Interconnect
  - X Bus to connect two NUMA Nodes
  - NVLink is used to connect GPU-GPU and GPU-Processor
  - Infiband EDR
- PyTorch, MVAPICH2-GDR 2.3.5
- We use and modify model definitions for ResNet(s) from *keras.applications* and AmoebaNet model from *TorchGpipe*

Accelerating Out-of-core Training at Scale

- Approach
  - LP: Layer Parallelism
  - Pipeline: Pipeline Parallelism
  - SP: Spatial Parallelism
  - Master: Hy-Fi
  - Opt: with Communication optimization in Hy-Fi
- Setup
  - Image Size: 2048 X 2048
  - 2048 NVIDIA V100 GPUs
- Speedup
  - Up to **2.67X** over Layer Parallelism (LP)
  - Near-linear scaling (**94.5%**) on 2,048 GPUs



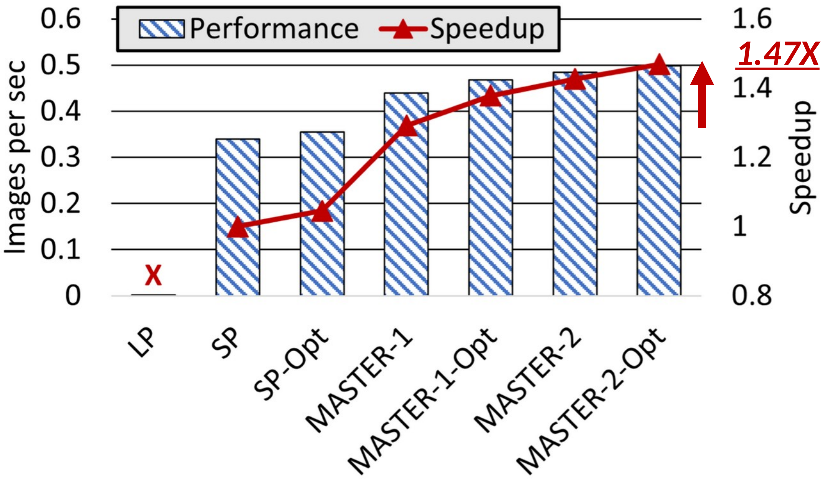
AmobeNet-f214



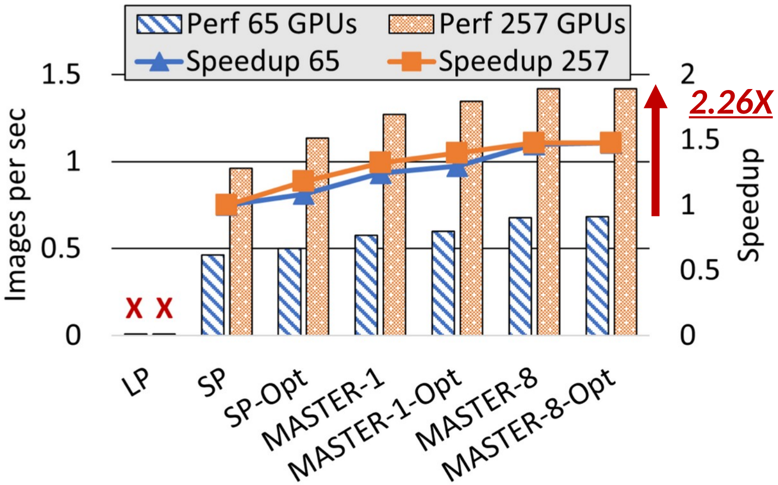
AmobeNet-f416

Enabling Training on Very High-Resolution Images

- Enabled training on 8,192 X 8,192 and 16,384 X 16,384 images sizes
- Speedup over basic spatial parallelism
  - 8,192 X 8,192 Images: 1.476X and 2.26X (Strong Scaling)
  - 16,384 X 16,384 Images: 1.47X



AmobeNet-f214 on 16,384 X 16,384 images



AmobeNet-f214 on 8,192 X 8,192 images

Agenda

- Introduction and Motivation
- Problems and Challenges
- Key Contributions
- Performance Evaluation
- **Conclusion**

Conclusion

- Proposed and Designed Hy-Fi
  - Integrated five different parallelization strategies
    - Spatial, Layer, Pipeline, Bi-directional, and Data
  - Communication optimizations to improve speedup
  - PyTorch and MPI for flexibility and scalability
- Performance Evaluation on large systems
  - Up to 2.67X speedup for out-of-core DNNS
  - Scaled Hy-Fi to 2,048 V100 GPUs on LLNL Lassen
  - Achieved 94.5% scaling efficiency with GEMS-Hybrid
- Future Work
  - Use Hy-Fi to train out-of-core DNNs on larger image tiles for digital pathology
  - Add more parallelization strategies

Thank You!

jain.575@osu.edu

Network-Based Computing Laboratory
http://nowlab.cse.ohio-state.edu/

High Performance Deep Learning
http://hidl.cse.ohio-state.edu/

The High-Performance Deep Learning Project
http://hidl.cse.ohio-state.edu/

The High-Performance MPI/PGAS Project
http://mvapich.cse.ohio-state.edu/