Motivation
Distributed Deep Learning (DL)
Deep learning training over large volumes of data using private and public resources for efficiency, cost, and privacy.

Network Bottleneck
• WANS: Low bandwidth (Mbps), Highly Heterogeneous
• LANs: High bandwidth (Gbps), Relatively Uniform
• DL Training: Parameters (gradients) are shared across machines every iterations until the DL model is converged

Approaches
A. Centralized Distributed DL training with important gradient exchange
• Reducing the size of data sent across networks
• No solution to handle heterogeneous WAN network bandwidth
• No study about computing power heterogeneity

B. Decentralized Distributed DL training with partial gradient exchange
• Target environment is uniform LANs
• No study about WAN constrains and computing power heterogeneity

Approaches A and B are not suitable for heterogeneous WAN environments

Implementation and Preliminary Results
Implemented a distributed DL algorithm in TensorFlow
• Decentralized DL with partial gradient exchange
• Open sourced (github): https://github.com/mesh-umn/TF.AKO
• Project home: https://www-users.cs.umn.edu/~chandra/tfako/home.html

Testbed
• 5 local servers with 6 CPU cores, 45G memory, 1Gbps ethernet per server
• Emulated network bandwidth
  - Uniform LAN: All links are 800 Mbps
  - Uniform WAN: All links are 40 Mbps
  - Heterogeneous WAN: Links are mixed with 40/20/10 Mbps

DL model & Dataset
• 2 convolution and 2 fully-connected layers, model size is 128 Mb
• CIFAR10 (32X32 color image: 50,000 training and 10,000 test images)

Computation Power Heterogeneity
• Network Bandwidth Constrains
  - Model accuracy drop or convergence failure in asynchronous DL trainings
  - Inefficient resource use in synchronous DL trainings

Effect of various network environments
Decentralized DL with partial parameter exchange outperforms in WAN environments ➔ Faster time-to-converge

Relationship between accuracy and batch size
There is a nonlinear relationship between processing time and batch size, and a range of batch size which results in similar accuracies

Proposed Idea
Accumulated priority gradient exchange to dynamically and automatically adjust:
• Network bandwidth
• Model convergence trend
• Monetary cost

This work is supported in part by NSF grant III-1422802