S-Caffe: Co-designing MPI Runtimes and Caffe for Scalable Deep Learning on Modern GPU Clusters

CSE 5449
Tu Tran
Outline

1. Motivation and Problem statement
2. Caffe and CUDA-Aware MPI
3. Challenges and Requirements
4. S-Caffe and MPI co-designs
   – Basic CUDA-Aware MPI Design (SC-B)
   – Overlapped Data Propagation (SC-OB)
   – Overlapped Gradient Aggregation (SC-OBR)
   – Hierarchical Reductions (HR)
5. Performance evaluation
Introduction

• The resurgence of deep learning
  – Public availability of large-scale datasets like MNIST, CIFAR, and ImageNet.
  – The presence of modern high-performance and data-parallel hardware, especially GPUs

• Triggering the development and adoption of DL frameworks
  – Ex: Caffe, Torch, Theano, TensorFlow, and Microsoft CNTK.

• Designing better, bigger, and deeper neural networks (DNN) for improving the accuracy of trained models
  – Ex: AlexNet, CaffeNet, GoogLeNet, and VGG.
Motivation and Problem statement

• As DL models getting bigger and complicated with large-scale datasets
  – Training time becomes an issue.
  – The performance of one-node training is saturated and cannot scale-up anymore
  – A need for a distributed training (scale-out)

How to design a DL framework that can both efficiently scale-up (single node) and scale-out (multiple nodes)?
Caffe Architecture

GPU0 (Bcast) → packed_comm_buff → GPU0 (Reduce) → ApplyUpdates → GPU0 (Reduce)

Data Propagation
Forward Backward Pass
Gradient Aggregation

GPU 0
Params
L_1
L_2
. . .
L_n
packed_reduce_buff

GPU 1
Params
L_1
L_2
. . .
L_n
packed_reduce_buff

GPU 2
Params
L_1
L_2
. . .
L_n
packed_reduce_buff

GPU 3
Params
L_1
L_2
. . .
L_n
packed_reduce_buff
CUDA-Aware MPI

Naïve mechanism

CUDA-aware mechanism
Challenges and Requirements

• **Parallel data reading**
  – Caffe uses Data Reader thread to constantly bring data from disk (maintained by LMDB) to memory queues.
  – How to feed the training data efficiently to all the compute units through parallel file system (PFS) such as Lustre?

• **Overlap of Computation and Communication**
  – Each phase blocks until the completion of the earlier phase
  – Data propagation and Forward pass can be overlapped
  – Gradient aggregation and Backward pass can be overlapped
Challenges and Requirements

• The design choices for data parallel approach:
  – Parameter-server:
    • One server will lead to bottleneck.
    • Many servers will take away compute power to do the training.
  – Reduction-tree:
    • A symmetric approach where all GPUs communicate using a tree-like pattern.
    • Each GPU performs the same amount of work in a collective fashion.

• Designing DL-Aware Communication Runtimes
  – MPI communication protocols and runtimes are optimized for small and large messages up to 4 MB for traditional HPC applications
  – However, DL frameworks, with their extensively large message sizes (256 MB), bring forward new requirements.
S-Caffe Architecture

Co-design (SC-OB) → S-Caffe → Middleware (MPI Runtime) → Co-design (SC-OBR)

1. Data Propagation
2. Forward Backward
3. Gradient Aggregation
4. CUDA-Aware NBC
5. Point-to-Point
6. Collectives
7. Chunked Reduce

HPC Platforms:
- Lustre
- InfiniBand
- GPUs
Basic CUDA-Aware MPI Design (SC-B)
Non-blocking Collectives for Maximal Overlap (SC-OB)
Gradient Aggregation for Maximal Overlap
Efficient DL-Aware Hierarchical Reduce (HR)

Typically, two ways to implement a reduce

1) Chunked Chain Algorithm (CC)
   \[0 \rightarrow 1 \rightarrow 2 \rightarrow 4\]

2) Binomial tree Algorithm (Bin)
   ![Binomial tree diagram]

CC performs better than Bin for large message sizes (> 8MB) and when # of processes <= 8
Hierarchical Reduce (160 GPUs)

CB-4:
- Low level: Chain
- Upper level: Binomial
- 4: the size of the chain

Tuned:
Use the best parameters for each algorithm
S-Caffe: CIFAR 10 (64 GPUs)

- CIFAR10 is a good example for medium sized models
  - Model Size is not very big!
  - Achieves good scalability

- S-Caffe: for single node, better or comparable performance

- S-Caffe allows multi-node training! (33x improvement over single-GPU training)
Comparison of S-Caffe, Inspur-Caffe, and CNTK using AlexNet

- AlexNet – the benchmark network for Image recognition
- Large number of parameters ~ 64 Million
- Communication buffer Size ~ 256 MB
S-Caffe: GoogLeNet (160 GPUs)

- GoogLeNet is another popular DL network
- 13 million parameters
- Communication buffer > 50 MB
- S-Caffe provides scale-out to 160 GPUs!!
Thank you for listening