

Efficiency Benchmarking of Distributed Machine Learning via Cloud

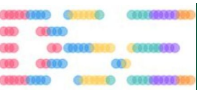
Xiaorui Liu

Computer Science and Engineering

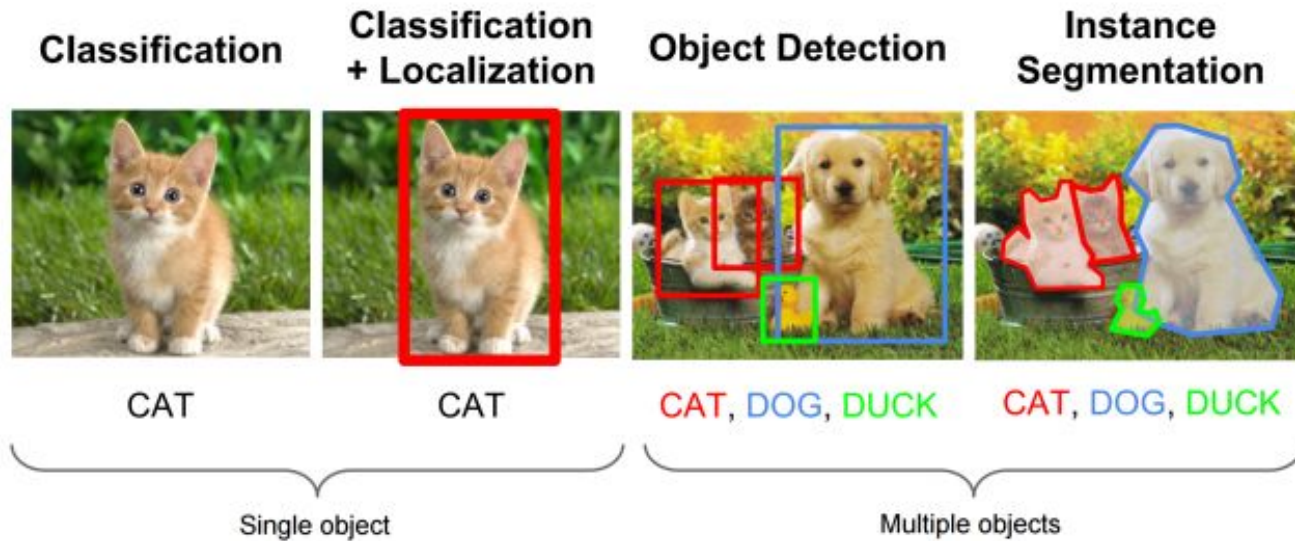
Michigan State University

<http://cse.msu.edu/~xiaorui/>

xiaorui@msu.edu



Large-scale Machine Learning



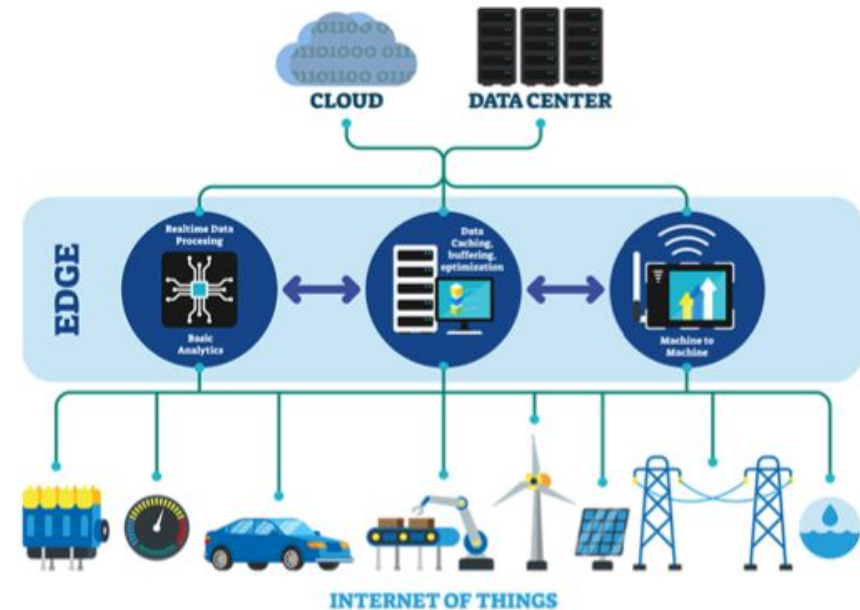
1.2 M
images

Distributed Machine Learning

Limited computation power and memory space in a single machine



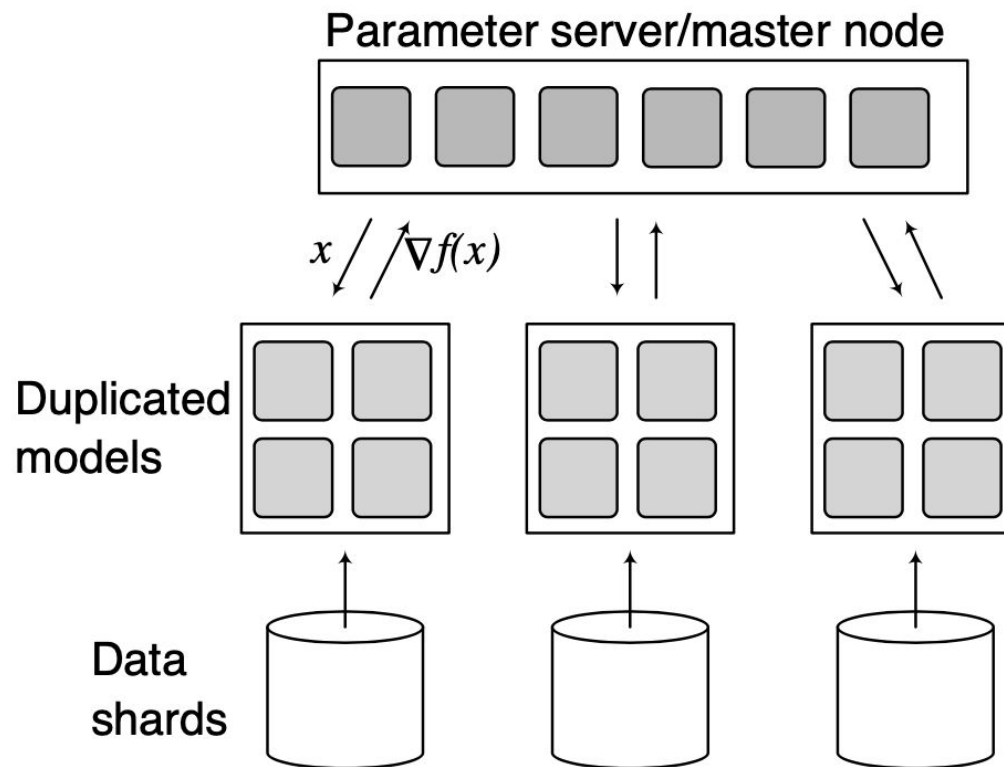
High Performance Computing



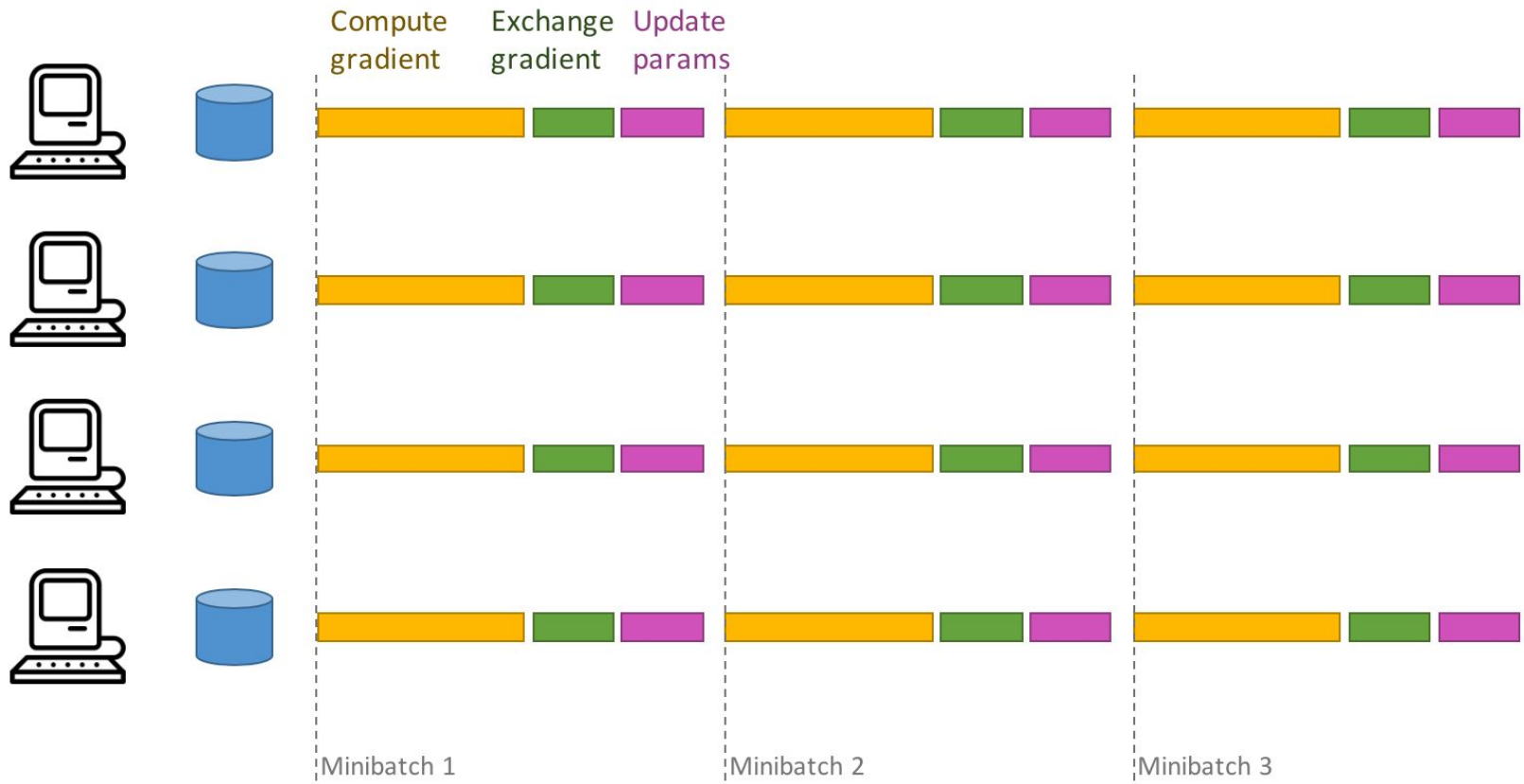
Edge Computing

Parallel SGD

$$\mathbf{x}^* := \arg \min_{\mathbf{x} \in \mathbb{R}^d} \left[f(\mathbf{x}) := \frac{1}{n} \sum_{i=1}^n f_i(\mathbf{x}) \right]$$



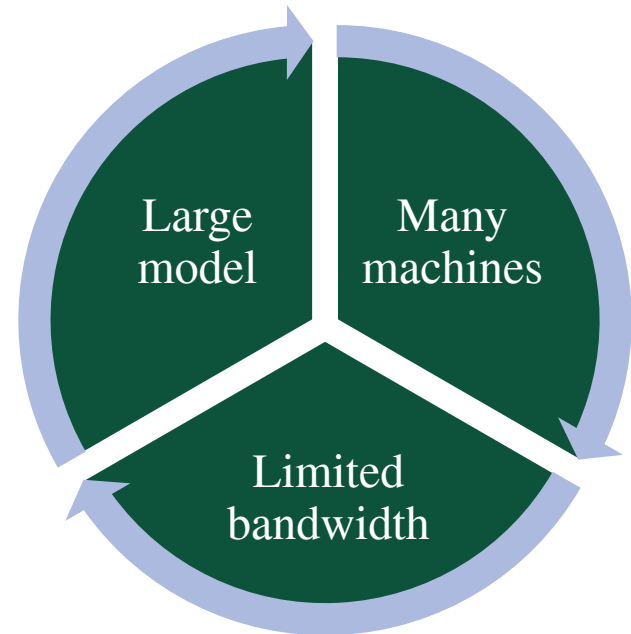
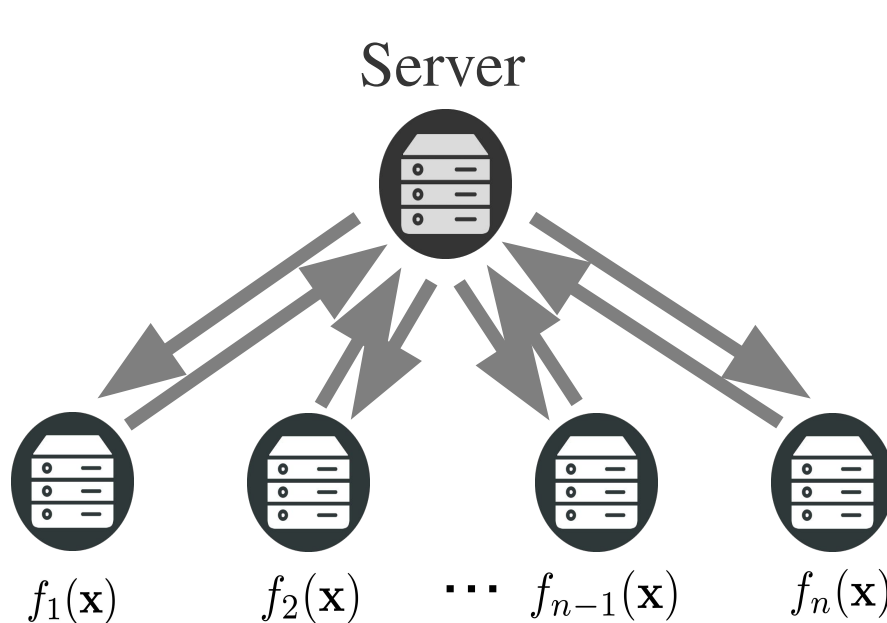
Parallel SGD



Time cost = local computation + **communication** + model update

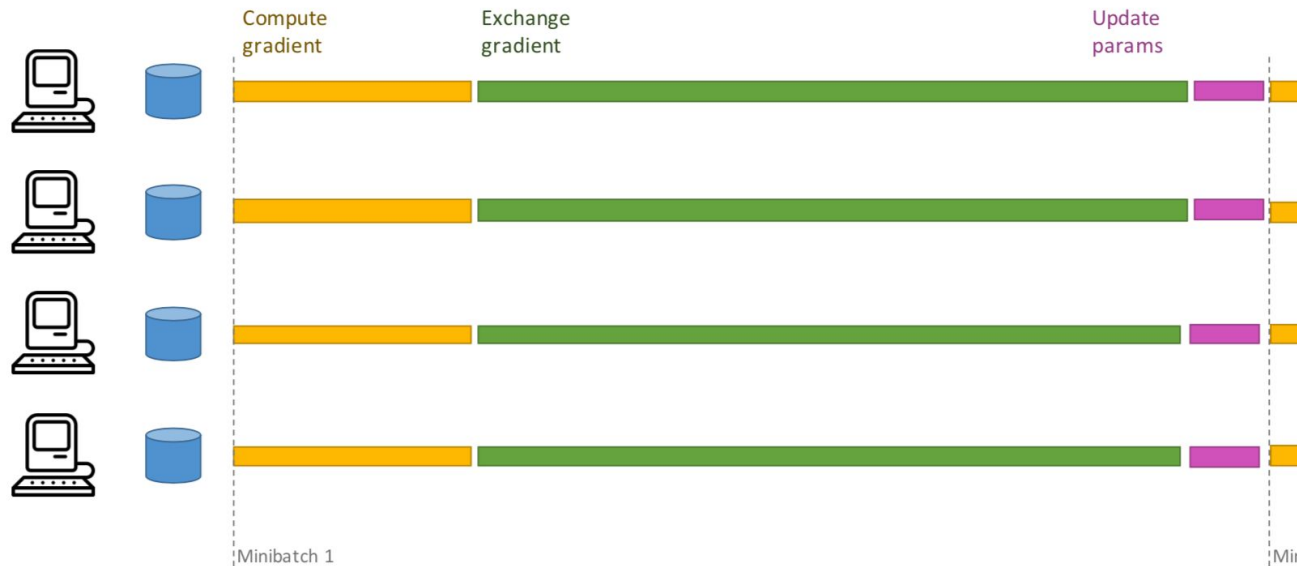
Communication Bottleneck

$$\text{Communication time} = \text{latency} + \frac{\text{model size}}{\text{network bandwidth}}$$



Communication bottleneck $\Rightarrow$ Poor efficiency & scalability

Project Goal: Efficiency Benchmarking

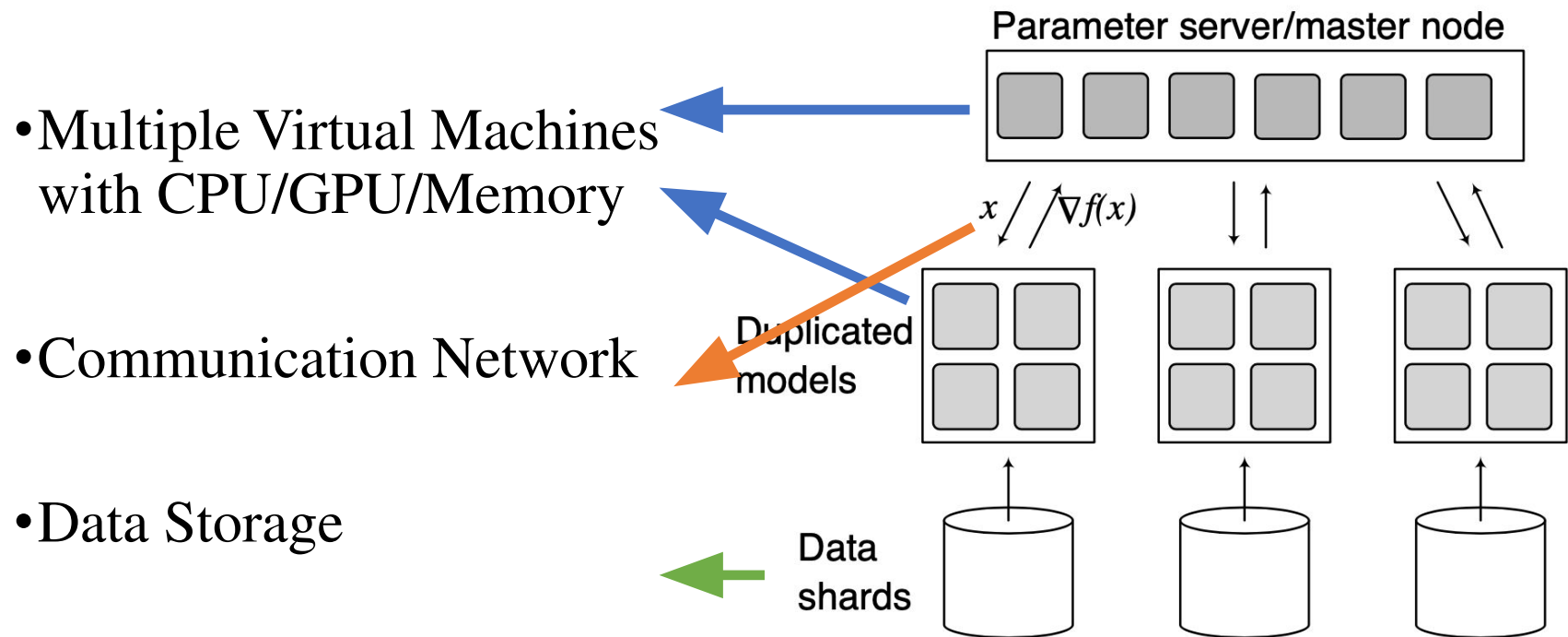


Measure the time cost decomposition and speedup under different **hardware and problem configurations** such as

- *Number of machines or GPUs*
- *Machine learning model size*
- *Network communication bandwidth*

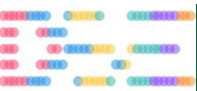
Cloud Computing

Advantages: resource elasticity and flexible configuration



Cloud Service

- Azure Machine Learning Service: workspace, compute instance, compute cluster, and Blob data storage
- Azure CLI Interface, Azure Machine Learning SDK for Python
- Virtual Machines:
 - 16 x NC6 instances (GPU K80 + CPU + Memory + Ethernet network)
 - 8 x NC24r instances (4 x GPU K80 + CPU + Memory + Infiniband network)
- VM images with PyTorch and MPI installed



Azure ML Studio

Microsoft Azure Machine Learning Studio

Michigan State University > mlworkspace > Notebooks

Notebooks

Files Samples

Users

xiaorui

cifar-10

PY job.py

PY job_large.py

PY models.py

PY train.py

PY train_large.py

PY util.py

PY job_large.py

```
1 # description: train CNN model on CIFAR-10 dataset with distributed PyTorch
2
3 # imports
4 import os
5 import urllib
6 import tarfile
7 from pathlib import Path
8 from azureml.core import Workspace
9 from azureml.core import ScriptRunConfig, Experiment, Environment, Dataset
10 from azureml.core.runconfig import PyTorchConfiguration
11
12 # get workspace
13 ws = Workspace.from_config()
14
15 # get root of git repo
16 prefix = Path(__file__).parent
17
18 # training script
19 # source_dir = str(prefix.joinpath("src"))
20 source_dir = str(prefix.joinpath("./"))
21 script_name = "train_large.py"
22
23 node_num = 1
24 model_name = 'AlexNet'
25
26
27 # azure ml settings
28
29 # compute_name = "gpu-K80-2"
30 # compute_name = "xiaorui1"
31 # compute_name = "xiaorui-cluster"
```

Azure ML Studio

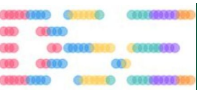
Compute

Compute instances **Compute clusters** Inference clusters Attached computes

+ New ↺ Refresh 🗑 Delete 📊 Edit columns ↶ Reset view | 📑 View quota

🔍 Search

Name	State	Size	Location	Created on ↓	Active runs	Idle nodes
xiaorui-cluster-NC24	✔ Succeeded (0 nodes)	STANDARD_NC24	northcentralus	May 5, 2022 11:13 PM	0	0
xiaorui-cluster-NC24r	✔ Succeeded (0 nodes)	STANDARD_NC24R	northcentralus	May 5, 2022 10:55 PM	0	0
xiaorui-cluster	✔ Succeeded (0 nodes)	STANDARD_NC6	northcentralus	May 5, 2022 4:14 PM	0	0



Azure ML Studio

Experiments

All experiments All runs

 Refresh  Archive experiment  Edit columns  Reset view

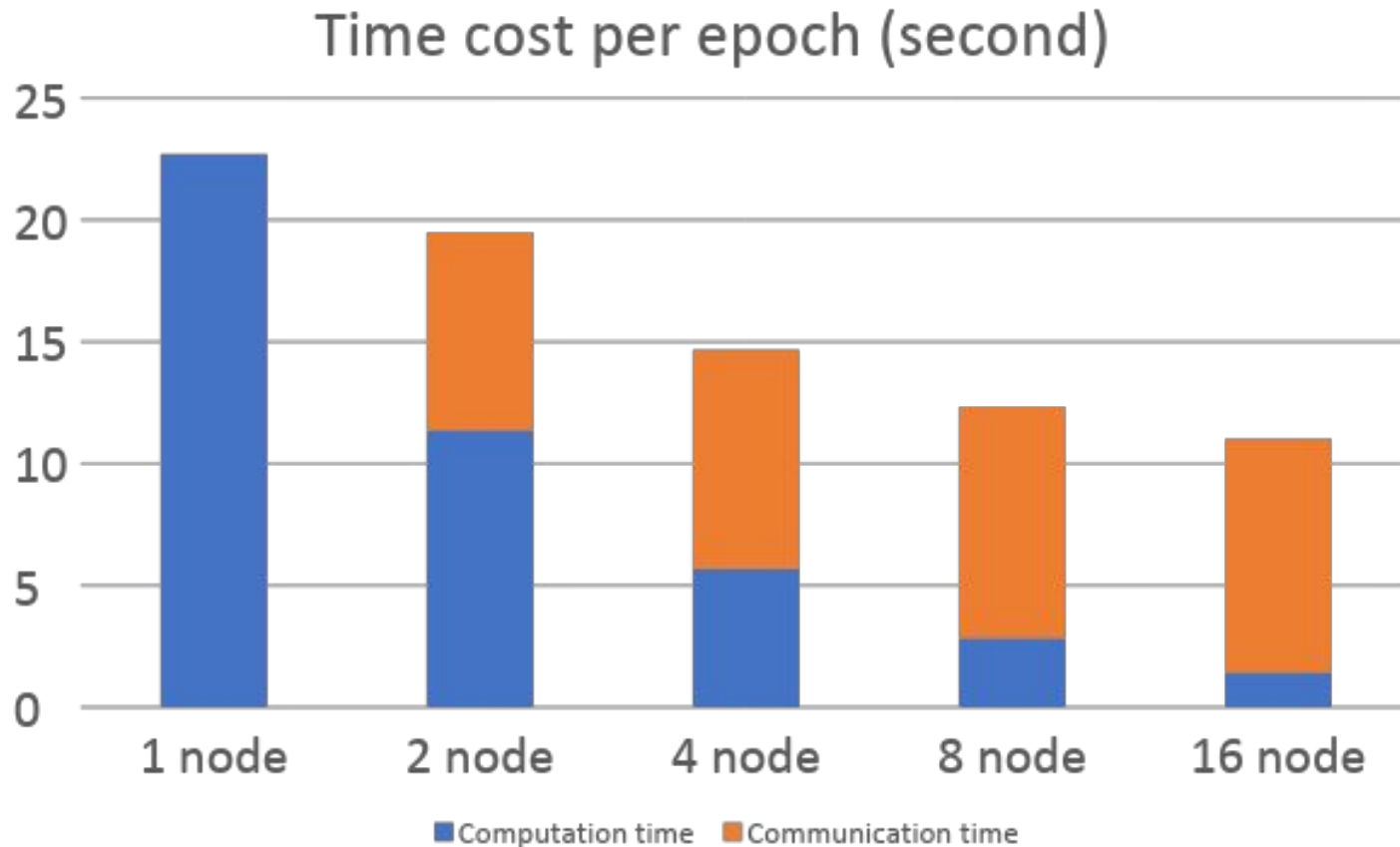
 Search

Showing 1-14 experiments

Experiment	☆ Latest run	Last submitted ↓
pytorch-cifar10-NC24R-AlexNet-nodcount-1	affable_kitten_1yhlg6fh	May 5, 2022 11:16 PM
pytorch-cifar10-NC24R-AlexNet-nodcount-2	happy_wheel_3rcg5jff	May 5, 2022 11:13 PM
pytorch-cifar10-distributed-AlexNet-nodcount-4	bold_farm_257k9k56	May 5, 2022 10:44 PM
pytorch-cifar10-distributed-AlexNet-nodcount-8	hungry_pen_9lsxkqy8	May 5, 2022 10:33 PM
pytorch-cifar10-distributed-AlexNet-nodcount-16	dynamic_camera_p5jp2jtg	May 5, 2022 10:10 PM
pytorch-cifar10-distributed-AlexNet-nodcount-2	ivory_pasta_x88mrrr7	May 5, 2022 8:15 PM
pytorch-cifar10-distributed-AlexNet-nodcount-1	gentle_steelpan_srbv8q0t	May 5, 2022 8:06 PM
pytorch-cifar10-distributed-example	frosty_piano_dcbm0znh	May 5, 2022 5:26 PM
pytorch-cifar10-distributed-bigmodel-nodcount-1	mango_grape_s7t01s2w	May 5, 2022 5:26 PM
pytorch-cifar10-distributed-nodcount-16	sincere_night_2ylvry15	May 5, 2022 5:14 PM
pytorch-cifar10-distributed-nodcount-8	green_loquat_gdvxx9p1	May 5, 2022 5:05 PM
pytorch-cifar10-distributed-nodcount-4	olive_eagle_bbct86y9	May 5, 2022 4:57 PM
pytorch-cifar10-distributed-nodcount-2	careful_cat_bvfxh6z7	May 5, 2022 4:44 PM
pytorch-cifar10-distributed-nodcount-1	elated_gold_7b2gppl2	May 5, 2022 4:18 PM

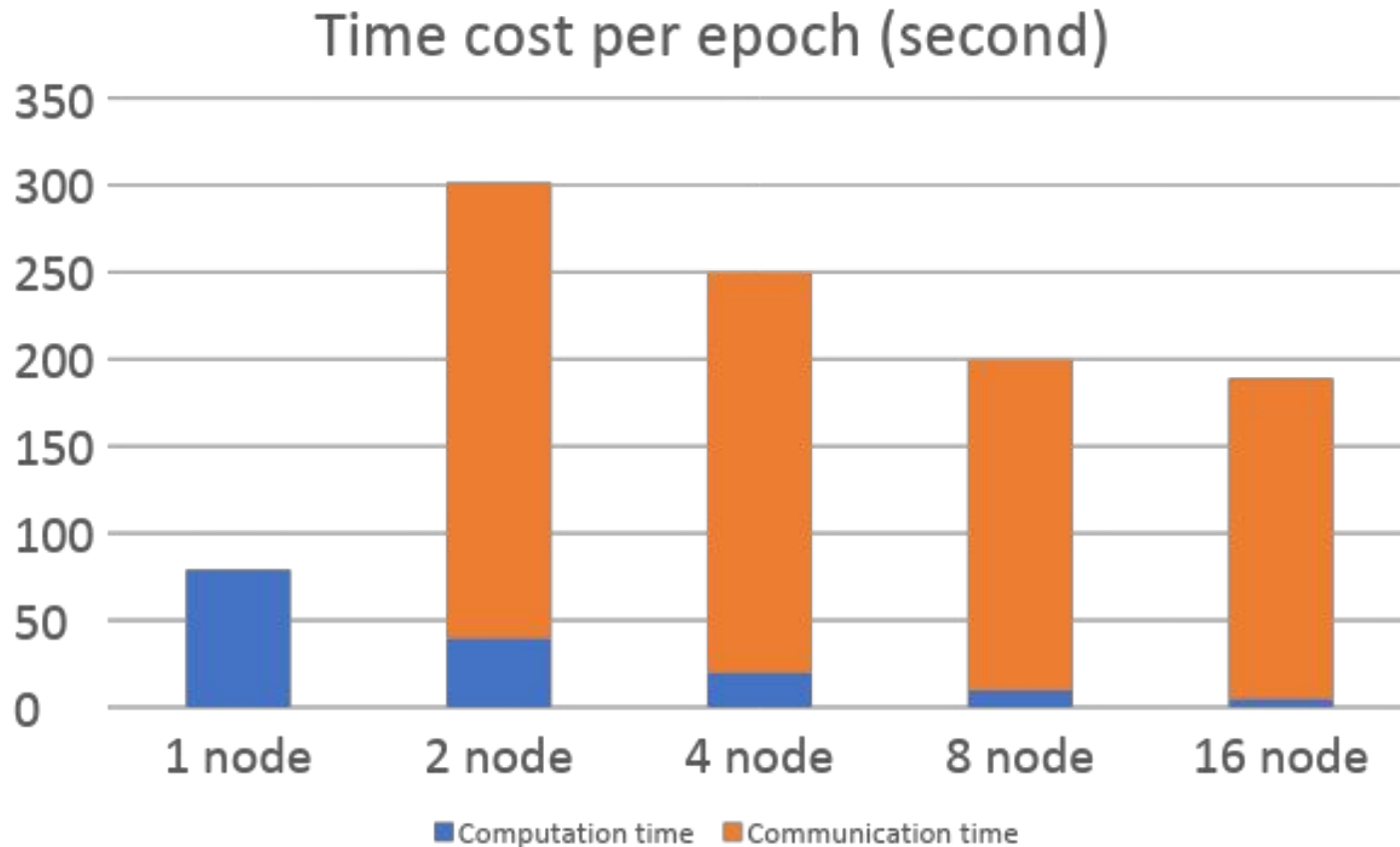
Measurement

Small model: LeNet (<1MB) under Ethernet network



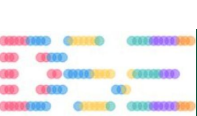
Measurement

Larger model: AlexNet (233 MB) under Ethernet network



Conclusion

- *Cloud computing* provides a versatile platform to benchmark the efficiency of distributed ML (comprehensive documentation)
- *Small models*: distributed ML benefits from parallel computing but the speedup is way worse than linear speedup
- *Large models*: distributed ML might not benefit from parallel computing but just waste computation power
- Faster communication network (such as *Infiniband*) is needed
- *Communication-efficient algorithms* are needed



Future Research: Communication-Efficient ML

