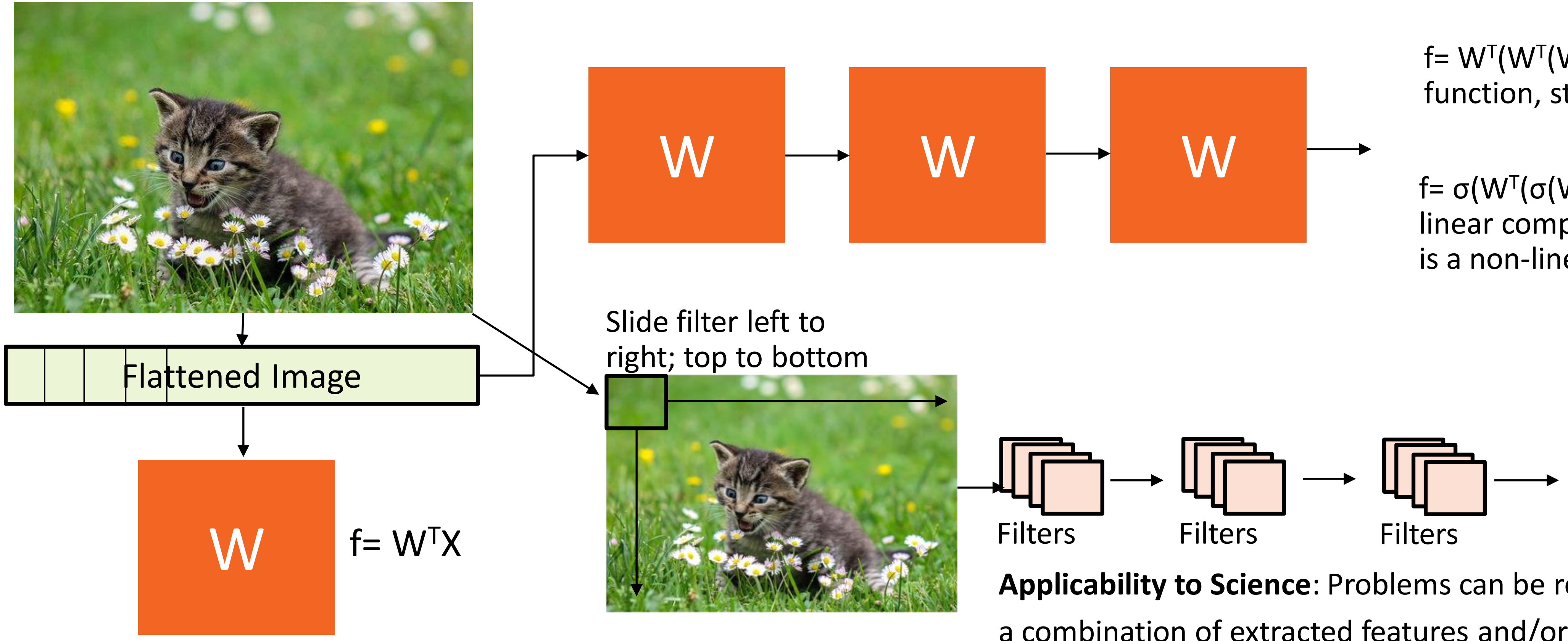
Bridging the Gap Between Deep Learning Algorithms and Systems

ABHINAV VISHNU

AUGUST 28TH, 2019

A QUICK INTRODUCTION TO MACHINE LEARNING/DEEP LEARNING

Input (X)



f= $W^T(W^T(W^TX)))$ – complex function, still a linear function

f= $\sigma(W^T(\sigma(W^T(\sigma(W^TX)))))$ – non-linear complex function where σ is a non-linear function

Predict the correct class

Image classification

Task: Predict the correct class – Image classification

Applicability to Science: Problems can be represented using a combination of extracted features and/or images

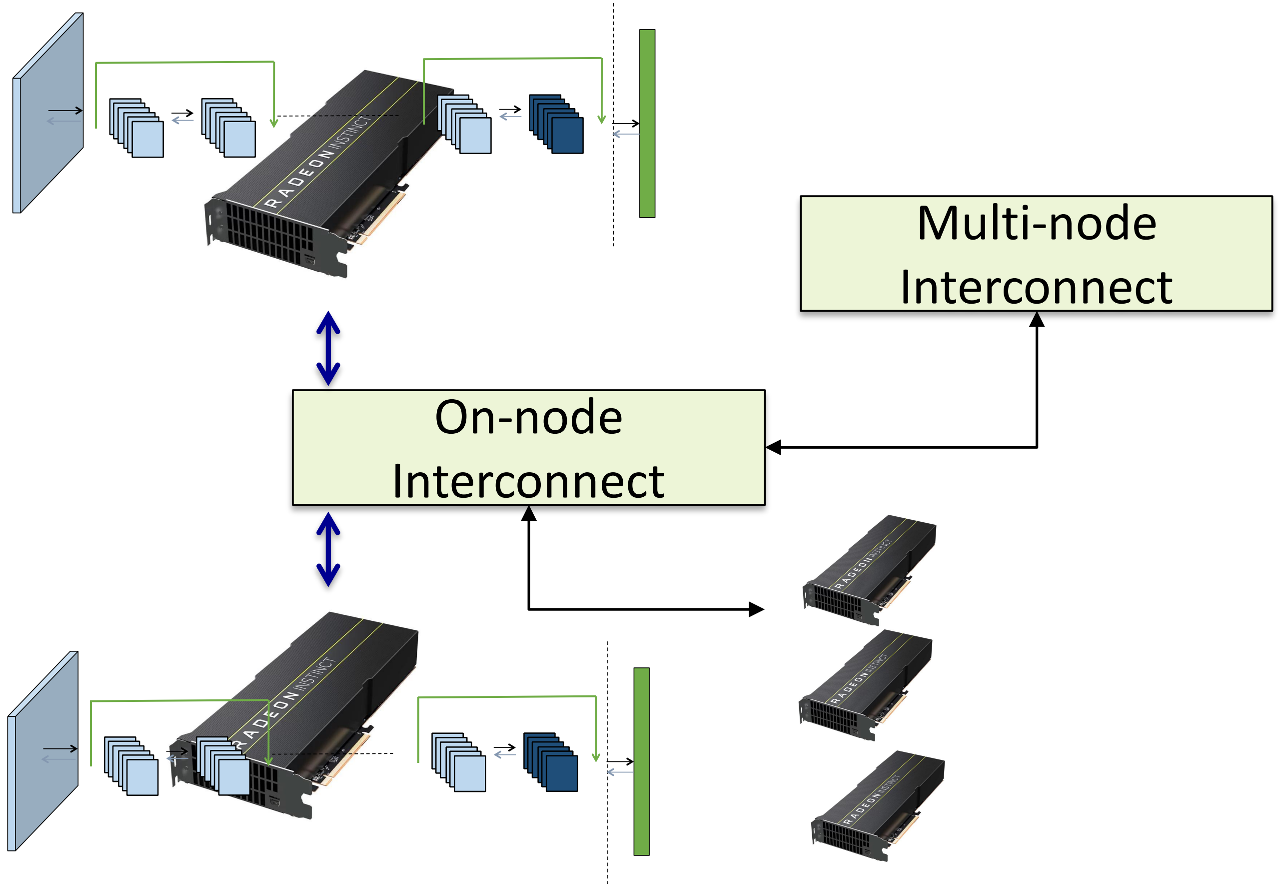
Data requirements: Complex representations (such as images) require large amount of labeled data

Compute requirements: Models with complex representations have large compute requirements requiring HPC systems

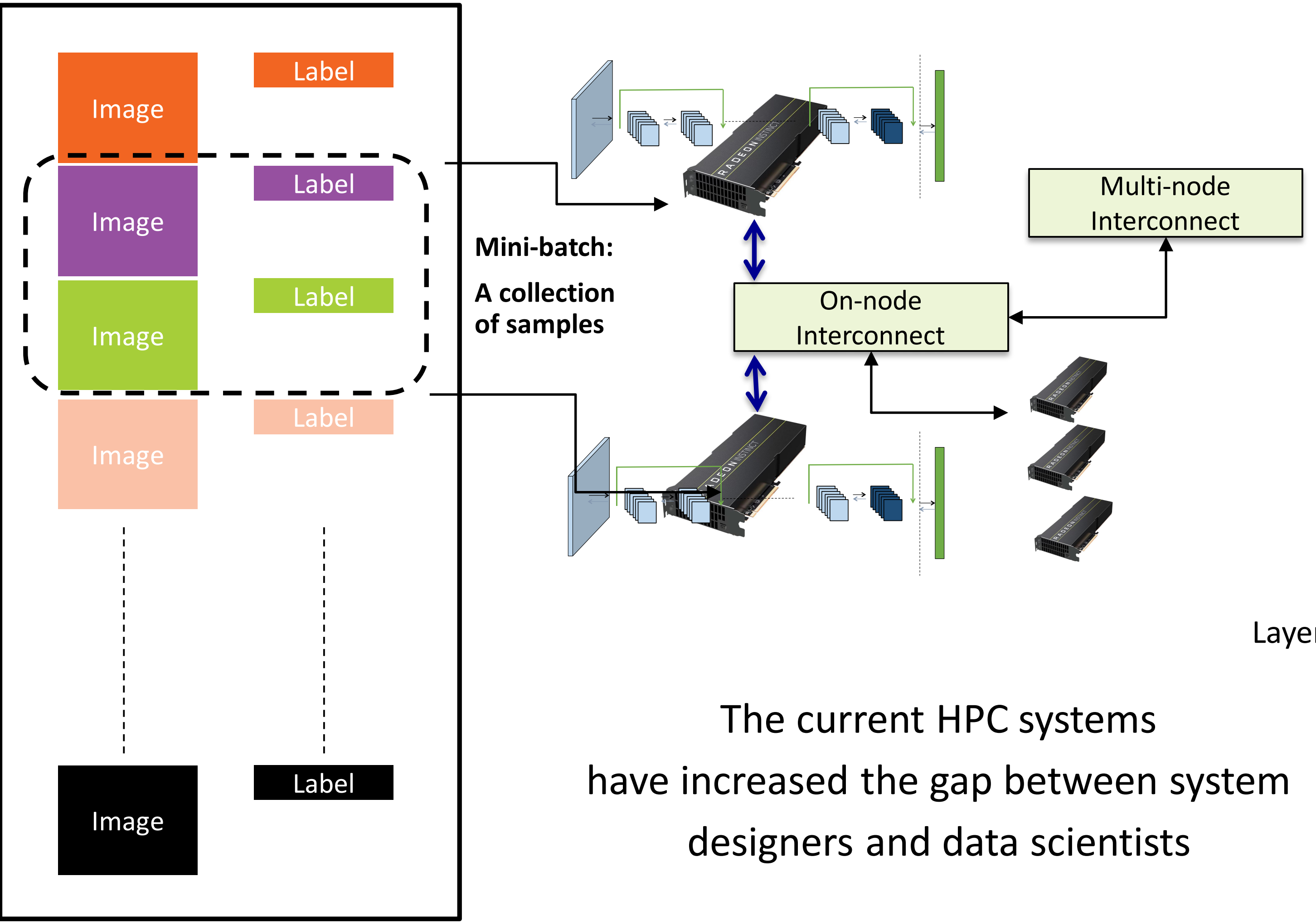
TRENDS IN LABELED DATA AND MODEL LEARNING (TRAINING) TIME

Name	Number of Images	Model Learning (Training Time)/Device
ImageNet	1.1M	~1 day
Tencent-ML	18M	Few Weeks
JFT-Google	300M	Not reported
Facebook	3.5B	Not reported

Blueprint of a candidate HPC system for scaling Deep Learning (DL) algorithms

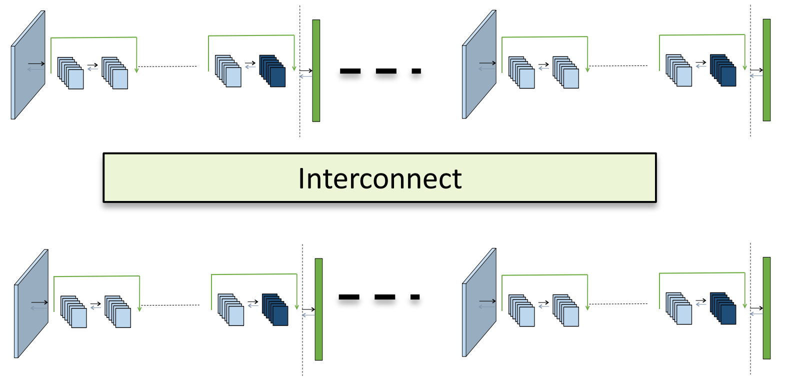


TRAINING DEEP LEARNING ALGORITHMS ON HPC SYSTEMS

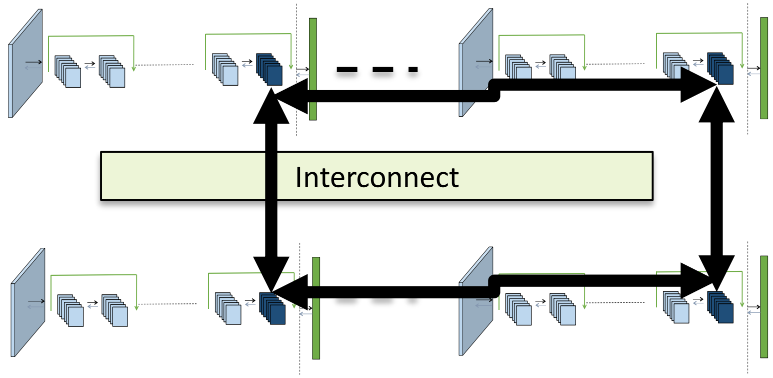


Training Set

No communication during Feedforward(error calculation) step



Layer-wise All-to-all reduction during Back-propagation (model learning) step



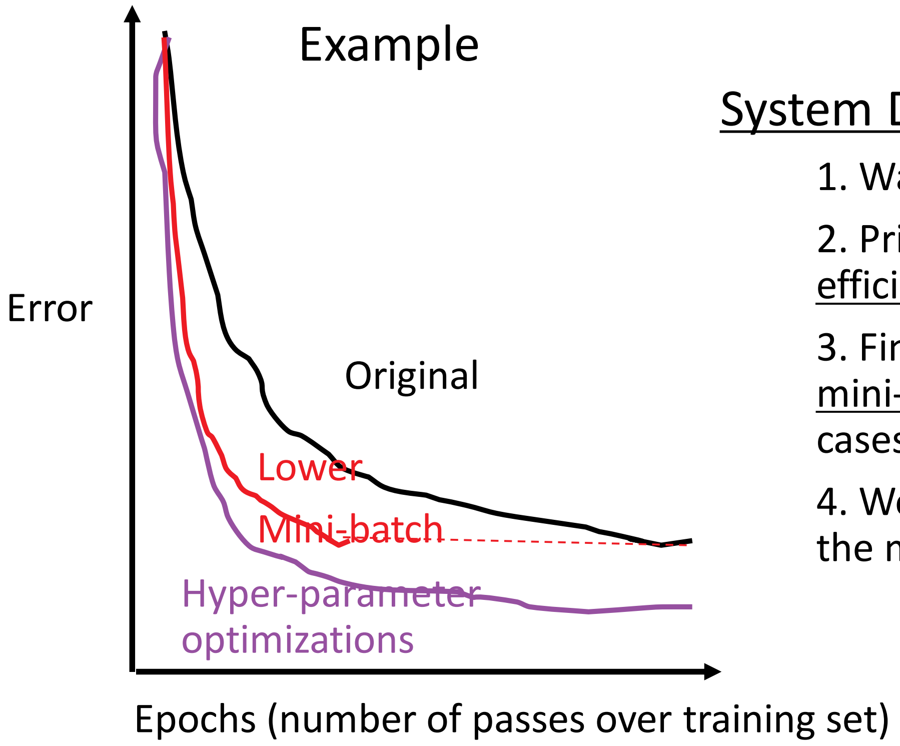
Popular Ring algorithm in DL algorithms

The current HPC systems
have increased the gap between system
designers and data scientists

THE MIS-MATCH BETWEEN SYSTEM DESIGNERS AND DATA SCIENTISTS

Data Scientist

1. Wants small mini-batch size
2. Primary objective is faster convergence
3. Defines an application model for system architect to optimize with generalizability to other problems
4. Work on algorithmic (hyper-parameter) optimizations to improve accuracy

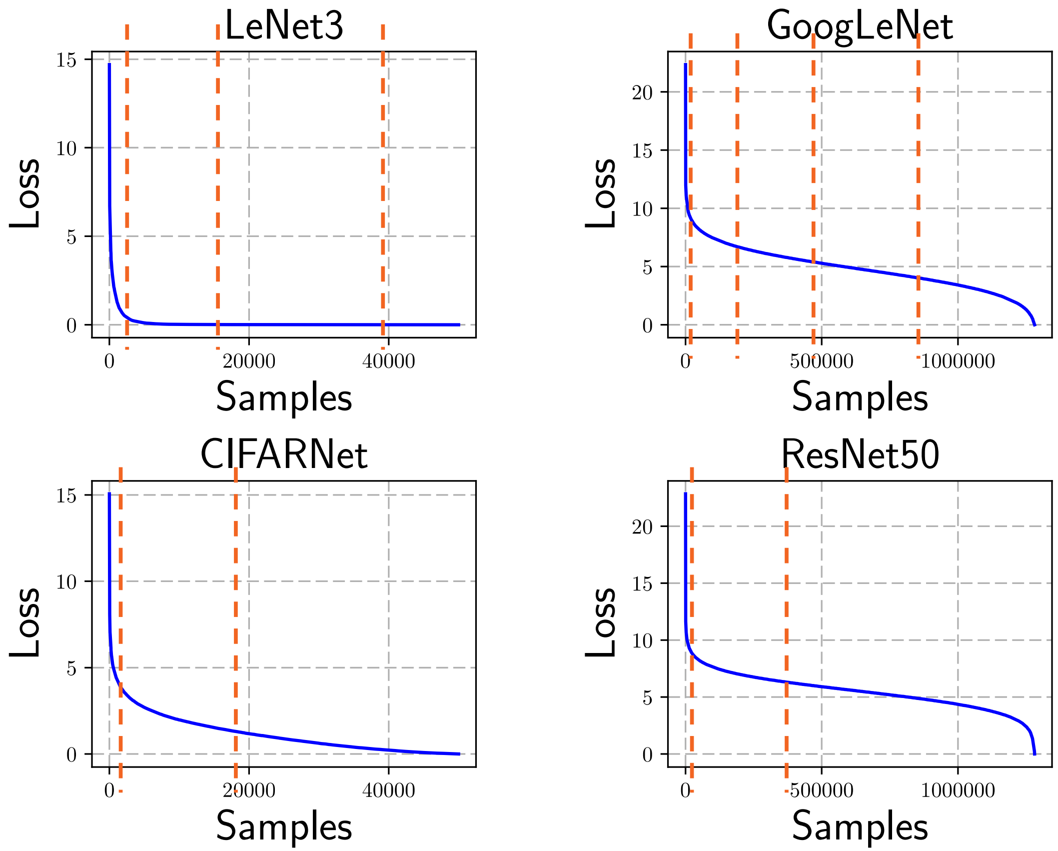


System Designer

1. Wants large mini-batch size
2. Primary objective is higher compute efficiency given convergence constraints
3. Find the inter-play between maximum mini-batch size and accuracy for the use-cases; generalizability is not the focus
4. Work on holistic system design to enable the maximum mini-batch size

Machine Learning models and system architecture/software needs to be co-designed to help bridge the gap between data scientists and system designer

POTENTIAL SOLUTION: ADAPTIVE MINI-BATCHING

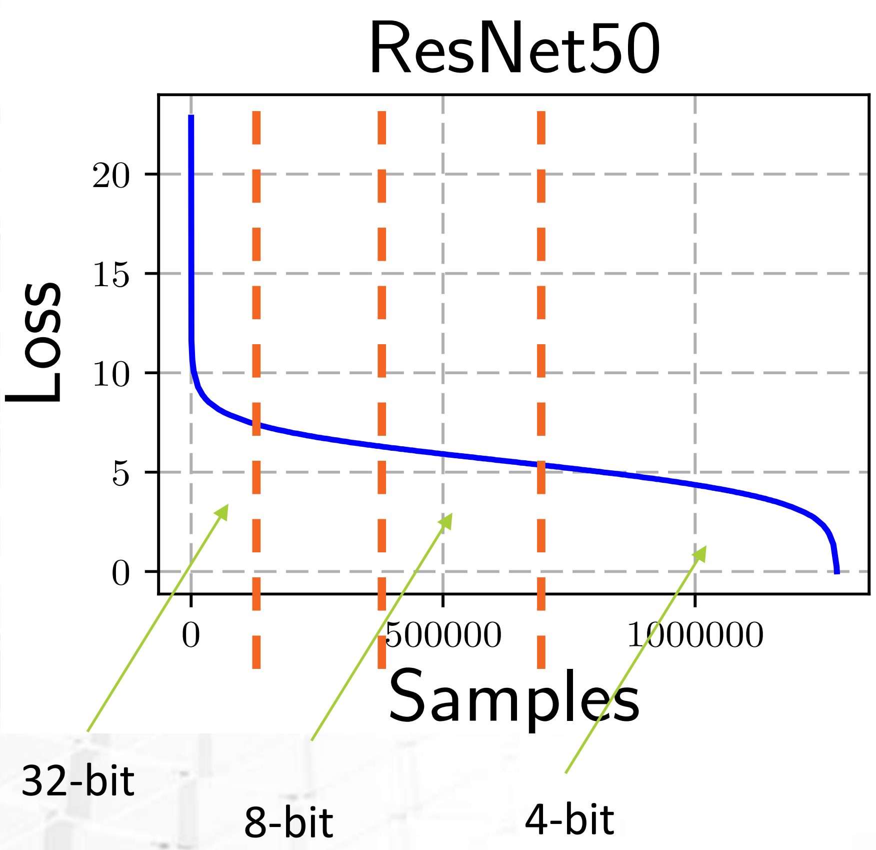


Three datasets, four networks

- The error magnitude is computed by adding the error from samples in the training dataset
- However, only a handful of samples contribute to the error
  - For example consider ResNet50 (after one epoch) on the adjacent figure
- Few samples have very high error, most samples have low error
  - The error curve becomes flatter with epochs
  - <u>Low error samples contribute less to model learning</u>
- A combination of large and small mini-batches may be created by epoch-wise analysis of the error/loss. An example is shown on the left
- Communication overhead is also reduced with adaptive mini-batching

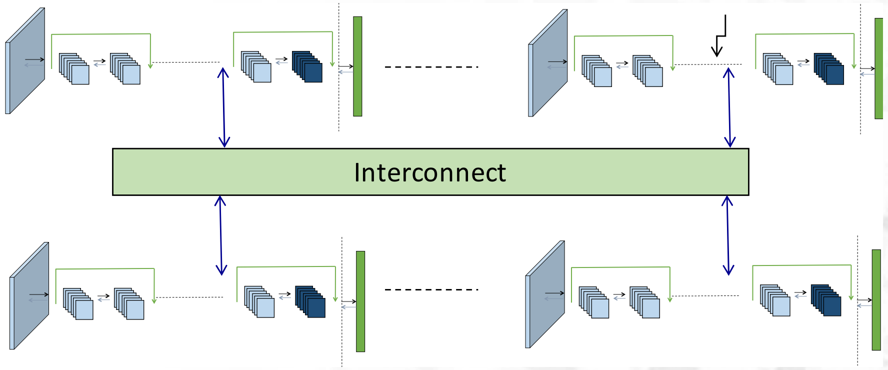
ACCELERATION USING ADAPTIVE PRECISION

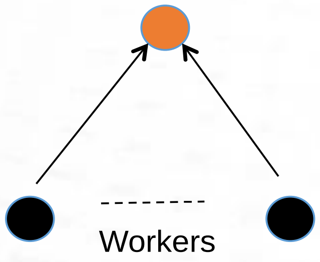
- Split the samples in multiple buckets of different precision
- The buckets may be defined by sorting the samples using non-increasing error
  - Flatter loss implies lower number of bits may be enough to encode the weight updates in that bucket
  - Loss becomes flatter with increasing epochs
- Reset the precision if validation loss increases
  - Reduce the precision adaptively after the reset
  - Self-corrects the problems due to aggressive reduction in precision



REDUCING COMMUNICATION CARDINALITY

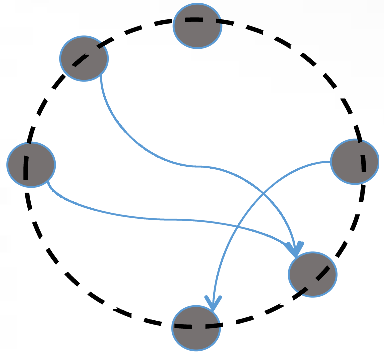
Communication complexity of the ring algorithm is linear,
but the achievable bandwidth is lower with decreasing chunk sizes





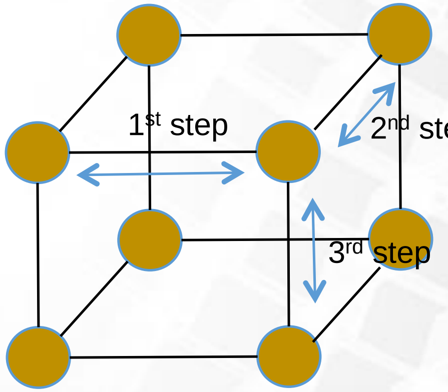
Parameter-Server
Based gossip

(a)



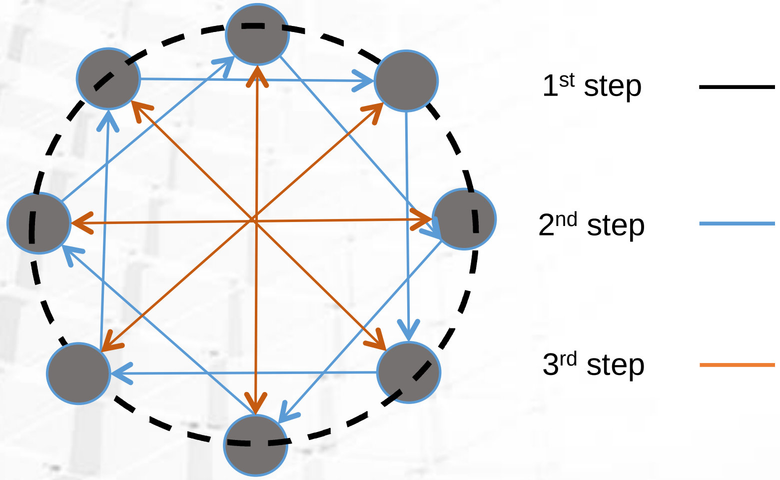
Random Gossip

(b)



(a)

Hypercube-based Gossip



(b)

Dissemination-based Gossip

CONCLUSIONS

- Deep Learning (DL) algorithms are becoming popular as they leverage complex representations (such as raw input with images) in addition to extracted features
- HPC systems play an important role in reducing the time-to-solution for DL algorithms
- There is a widening gap in primary metrics of concern between a data scientist and a system designer
- We proposed approaches to bridge the gap by using adaptive mini-batching
  - For high error samples, use small mini-batches
  - For low error samples, use large mini-batches under the memory and compute constraints of the system
  - Proposed adaptive precision (high precision for high error samples) that matches well with the compute capabilities of today's systems
  - Proposed solution for addressing the limitations of all-to-all reduction by using reduced communication cardinality
- We hope to work with the scientific community to enhance these solutions and present results through publications and open source software

THANKS FOR LISTENING!!

QUESTIONS?

RADEON TECHNOLOGIES GROUP

Contact: **Abhinav Vishnu,**
Abhinav.Vishnu@amd.com

DISCLAIMER & ATTRIBUTION

ATTRIBUTION