Mercury: Efficient On-Device Distributed DNN Training via Stochastic Importance Sampling

Xiao Zeng, Ming Yan, Mi Zhang

SenSys 21

Presented by Huai-an Su

Outline

- Motivation
- Mercury
- Implementation Setup
- Results
- Conclusion

Motivation

- Deep learning model training
- Smart phones, AR/VR headsets, Smart speakers, Robotics, Drones





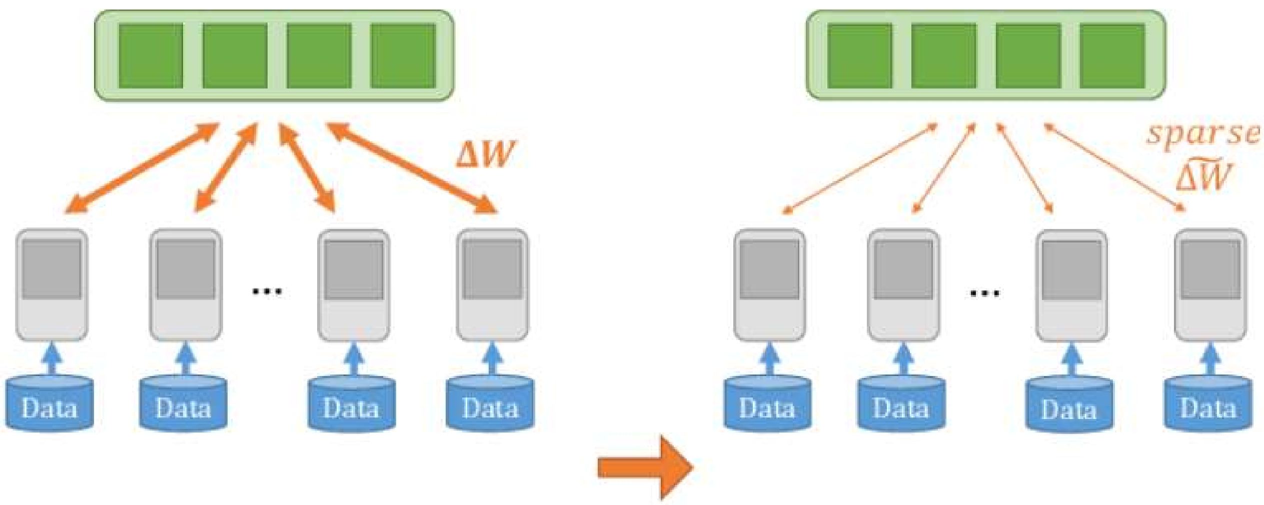
Motivation

- Contribution: On-device distributed Training
- Limitation: significant amount of training time
- Limited bandwidth slows down communication

$$T = E \cdot (T_{cp} + T_{cm}).$$

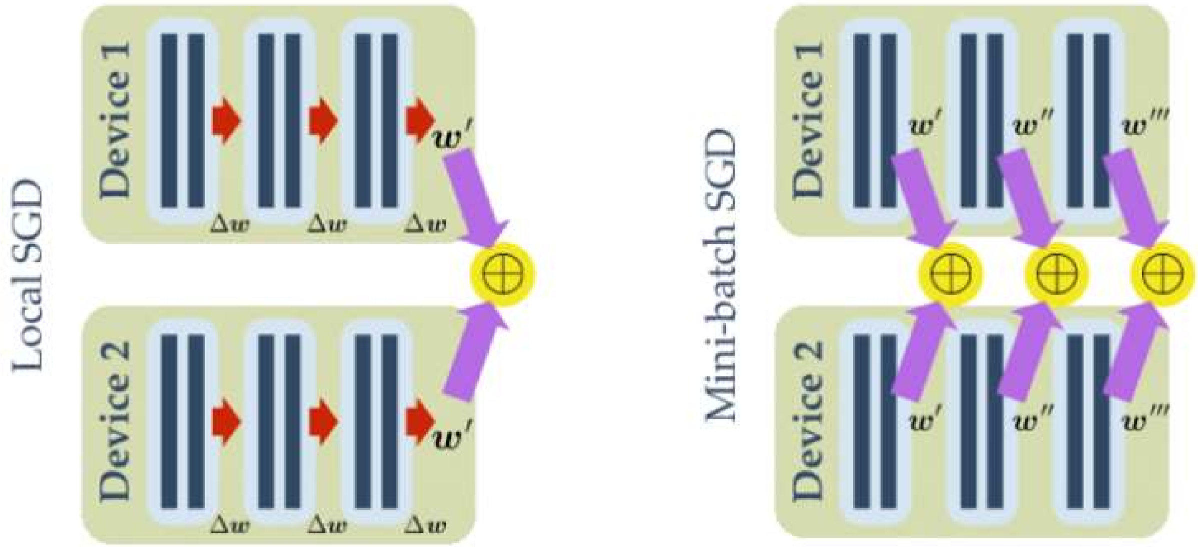
Motivation

- Solution 1: gradient compression
- Quantizing gradients (smaller number of bits)
- Sparsification (selecting important gradients)
- Sacrifices the accuracy of the trained model



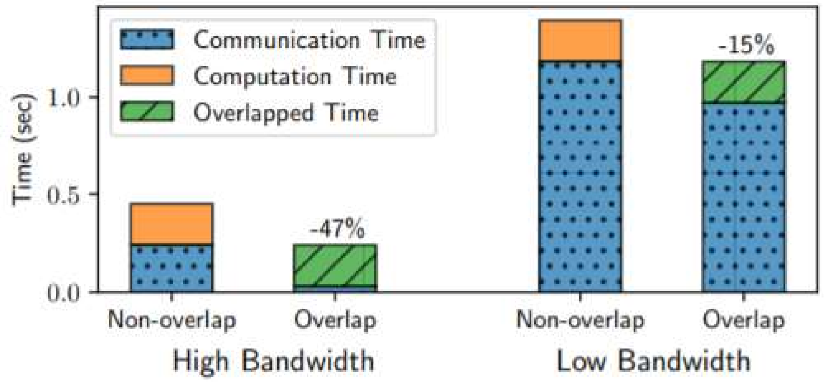
Motivation

- Solution 2: Local SGD
- Clients perform multiple local updates
- May deviate from global optimal model



Motivation

- Solution 3: Overlapping
- Overlap communication with gradient computation
- Can mask out the communication cost
- On-device communication is way higher than gradient computation

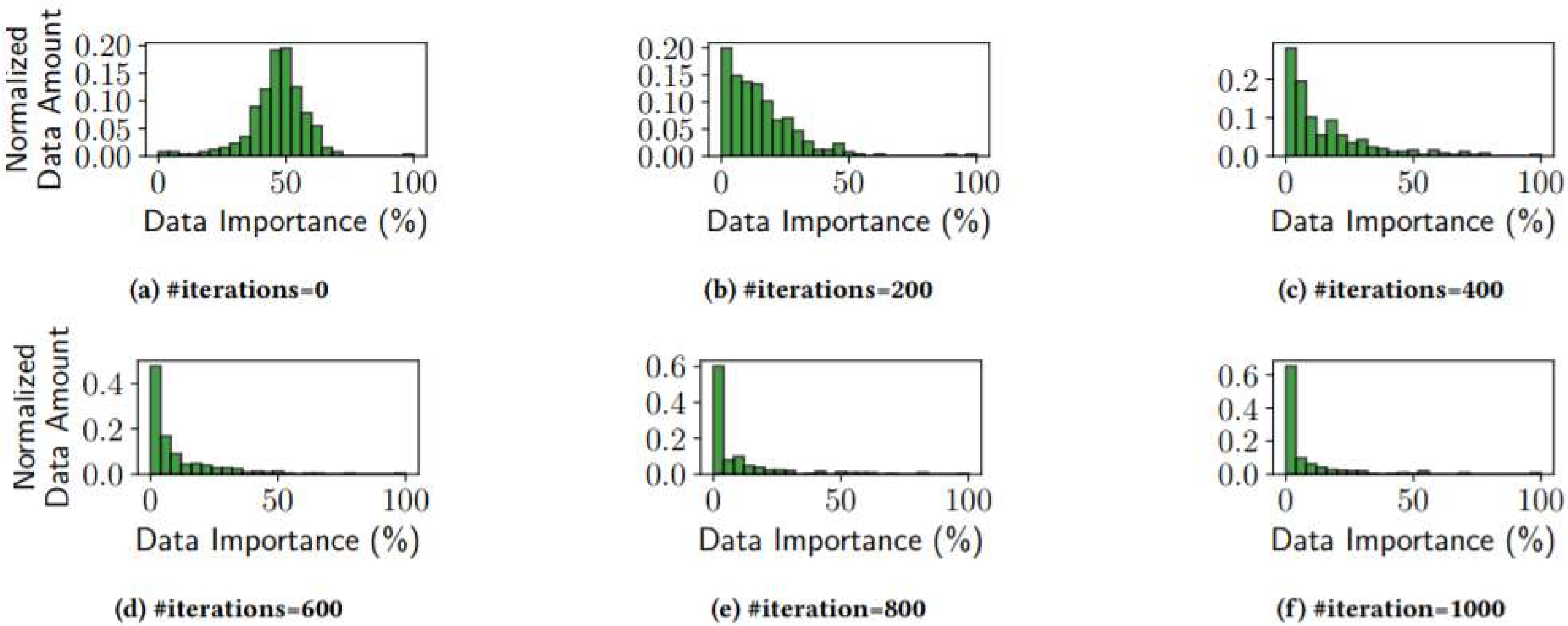


Mercury

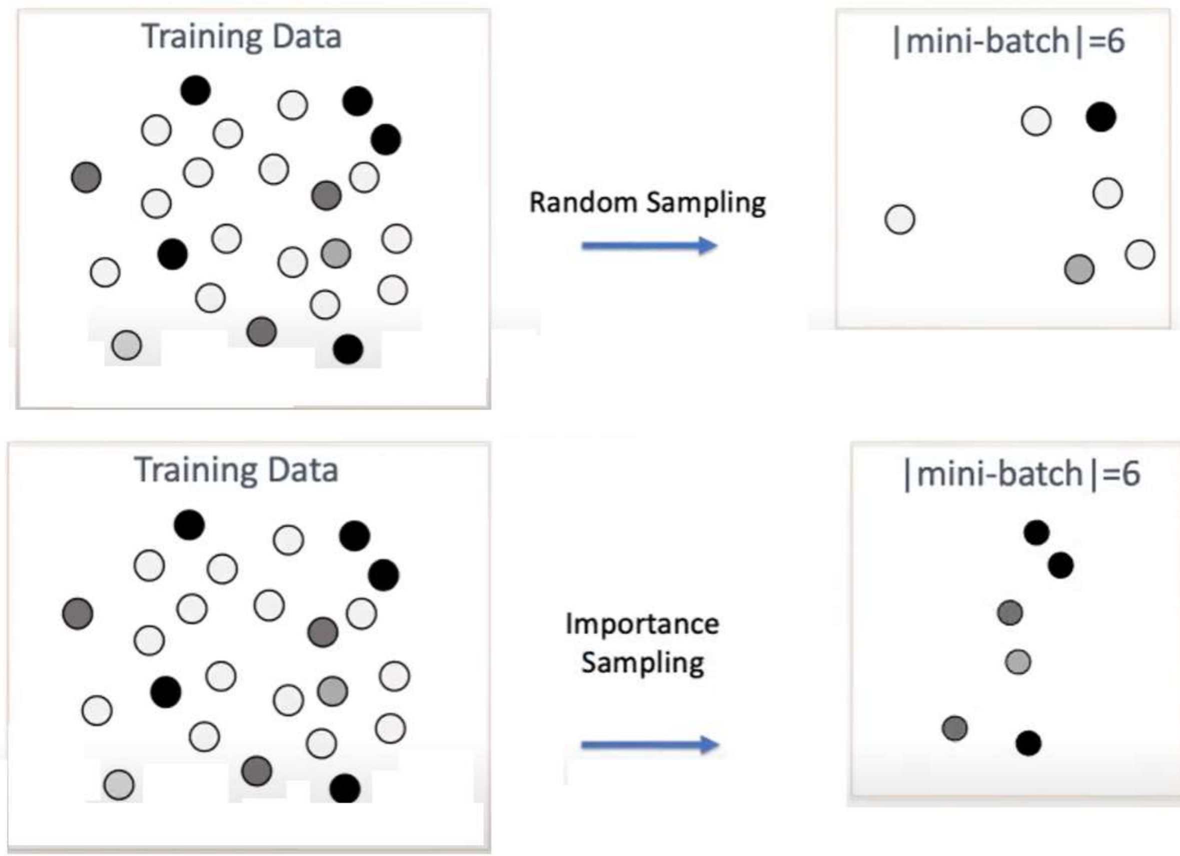
- Improve training efficiency
- Reduce the iterations for convergence
- Key concept: importance sampling

Mercury

- Data importance distribution differs as the iteration goes up

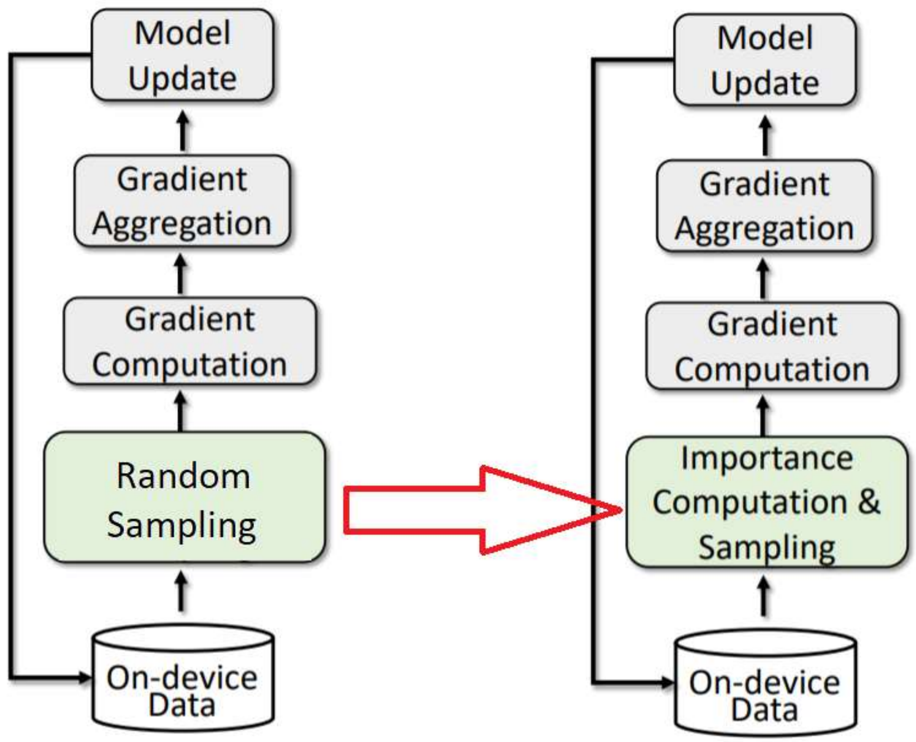


Mercury


Training Data
Random Sampling
|mini-batch|=6
Training Data
Importance Sampling
|mini-batch|=6

Mercury

- Framework



Mercury

- Challenge 1: importance sampling incurs computation cost

$$Speedup = \frac{E \cdot (T_{cp} + T_{cm})}{E_{is} \cdot (T_{cp} + T_{cm} + T_{is})} = \frac{1}{\frac{E_{is}}{E} \cdot (1 + \frac{T_{is}}{T_{cp}+T_{cm}})}$$

E: iteration of standard SGD

E_{is}: iteration of importance sampling

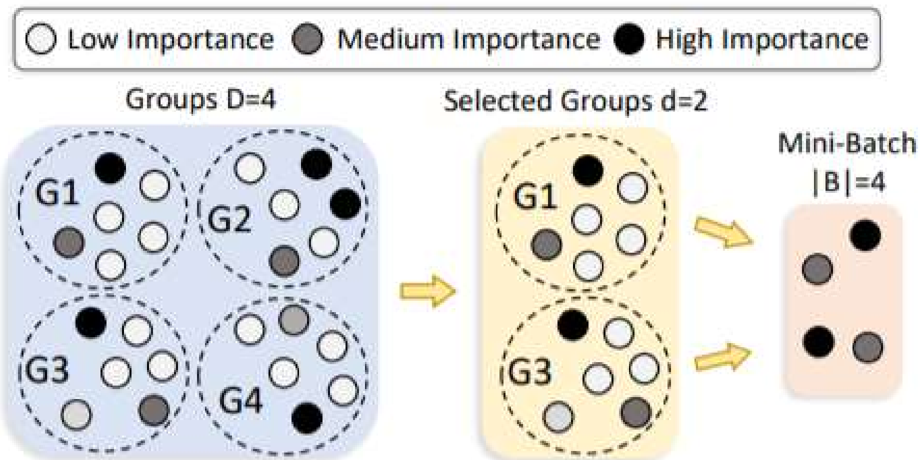
T_{cp}: computation time

T_{cm}: communication time

T_{is}: importance sampling computation time

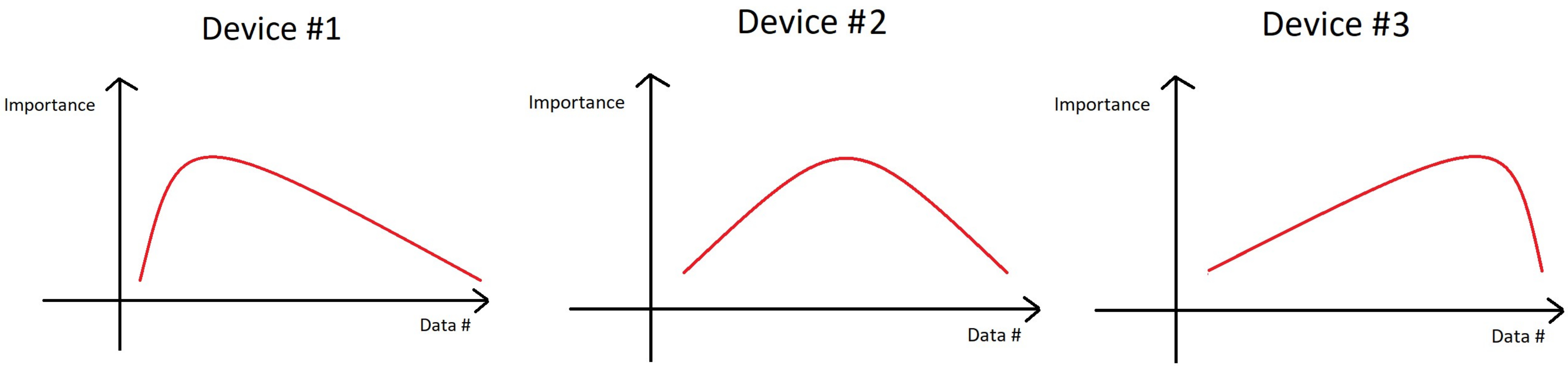
Mercury

- Solution 1: Group-wise Importance Sampling
- Divide training data into groups
- Only update importance distribution for one group
- Reuse cached distribution from other groups



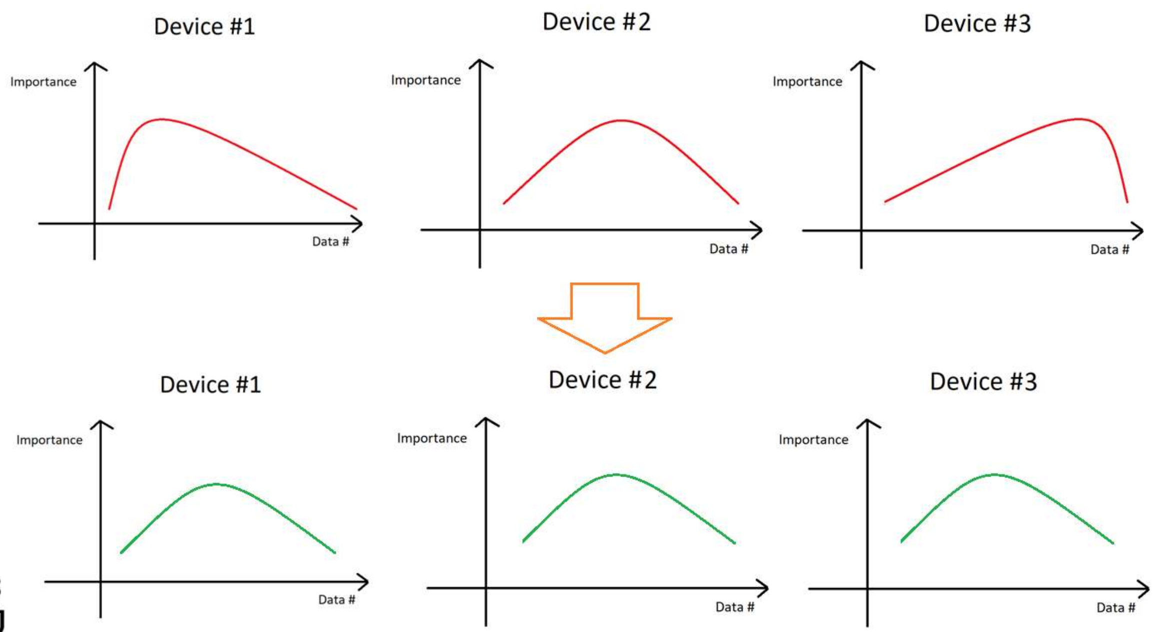
Mercury

- Challenge 2: Data importance is imbalanced
- A device may learn global trivial samples



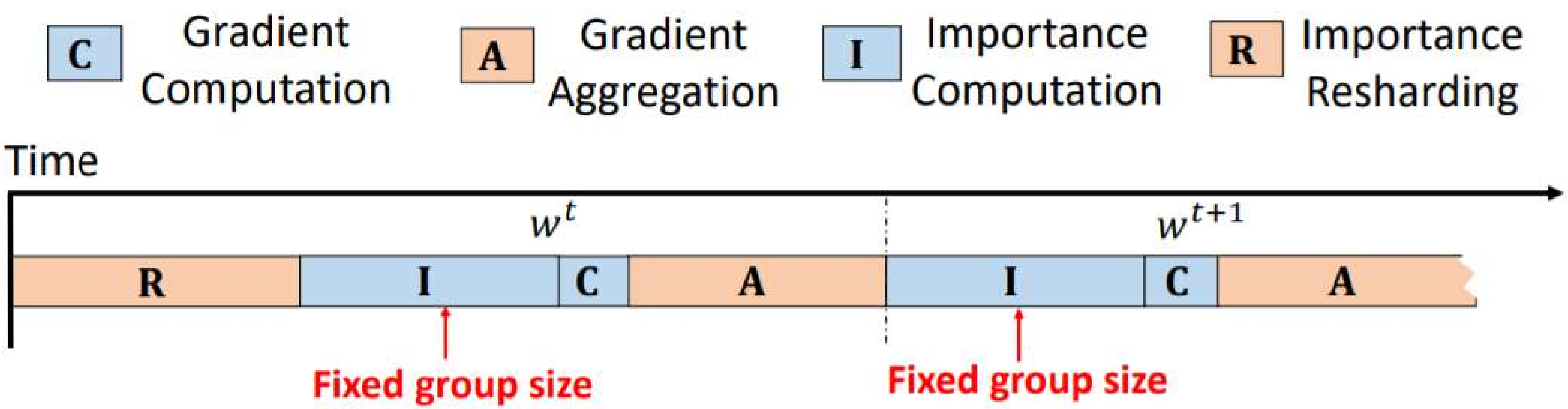
Mercury

- Solution 2: Importance-aware Data Resharding
- Redistribute samples among workers
- Select non-trivial samples to shuffle



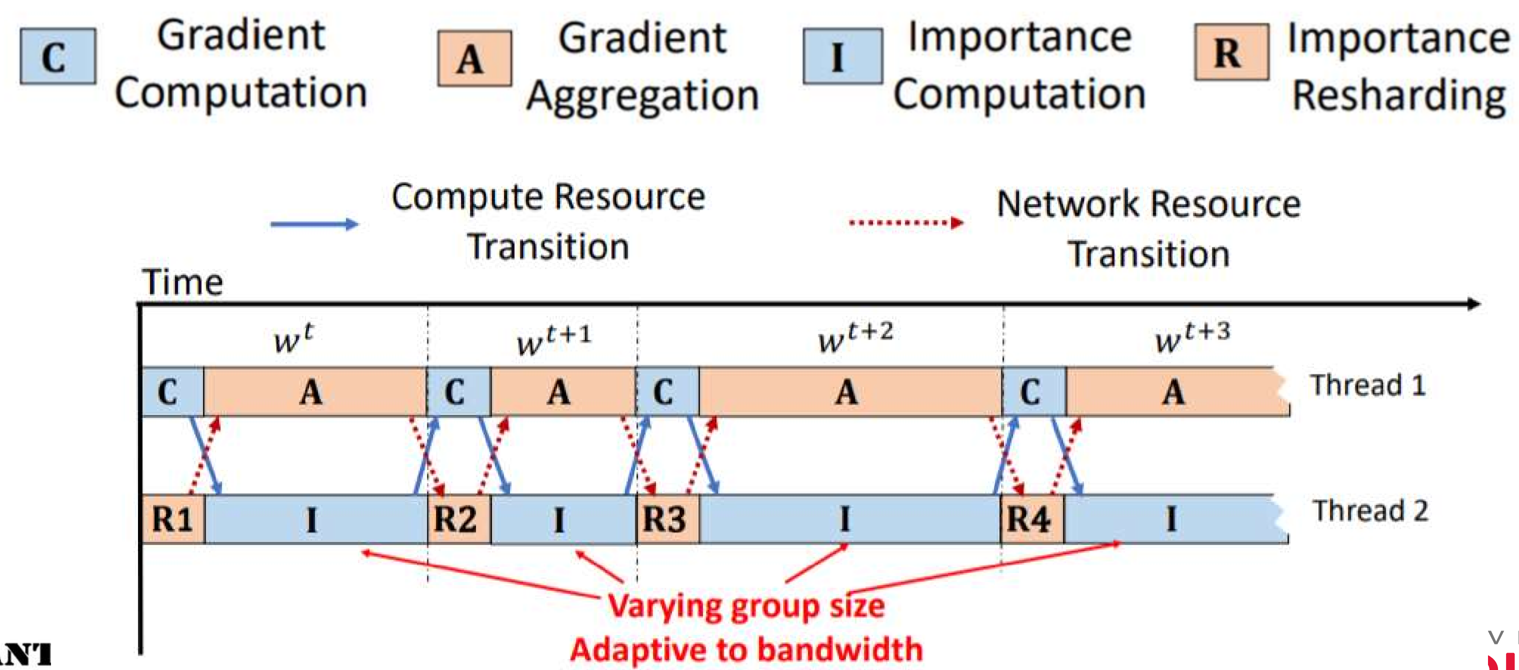
Mercury

- Challenge 3: Sequential implementation is inefficient



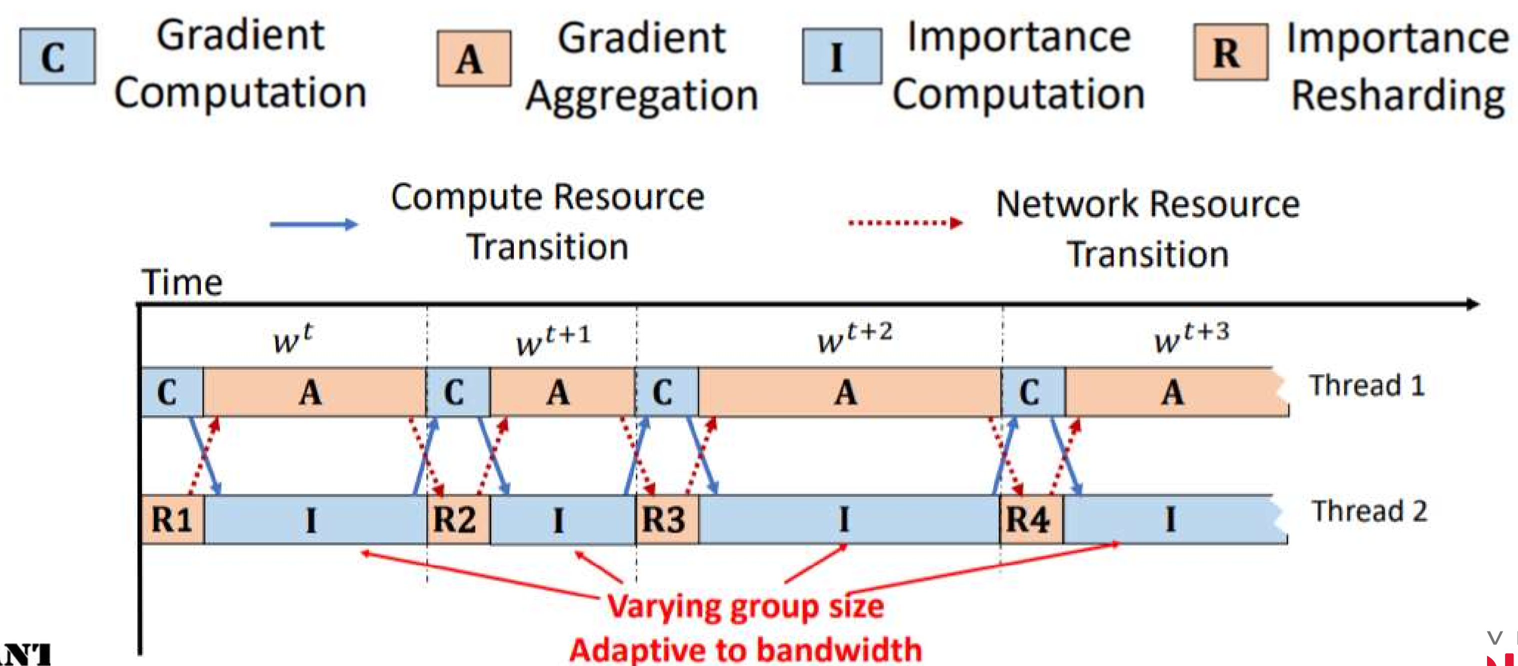
Mercury

- Solution 3: Bandwidth-adaptive computation-communication (BACC) scheduler



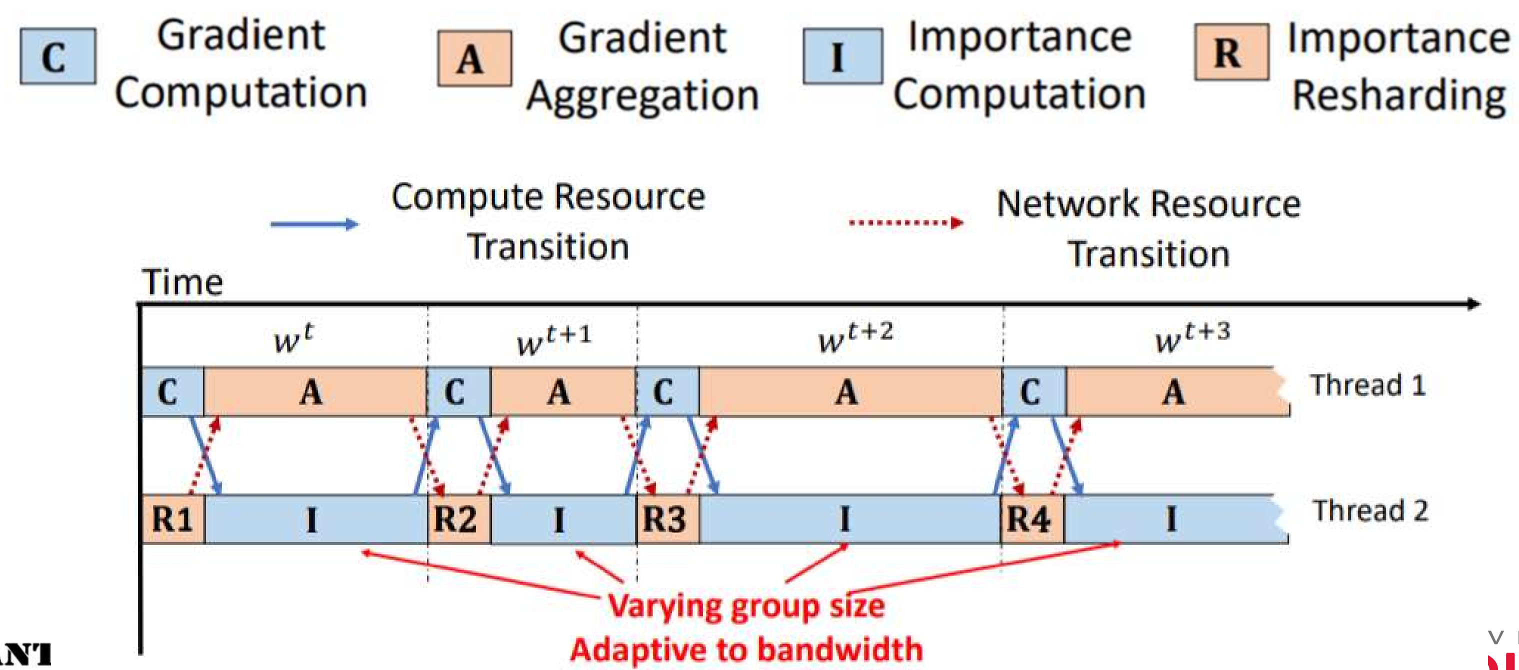
Mercury

- To fully overlap I with A: Adopt varying group sizes



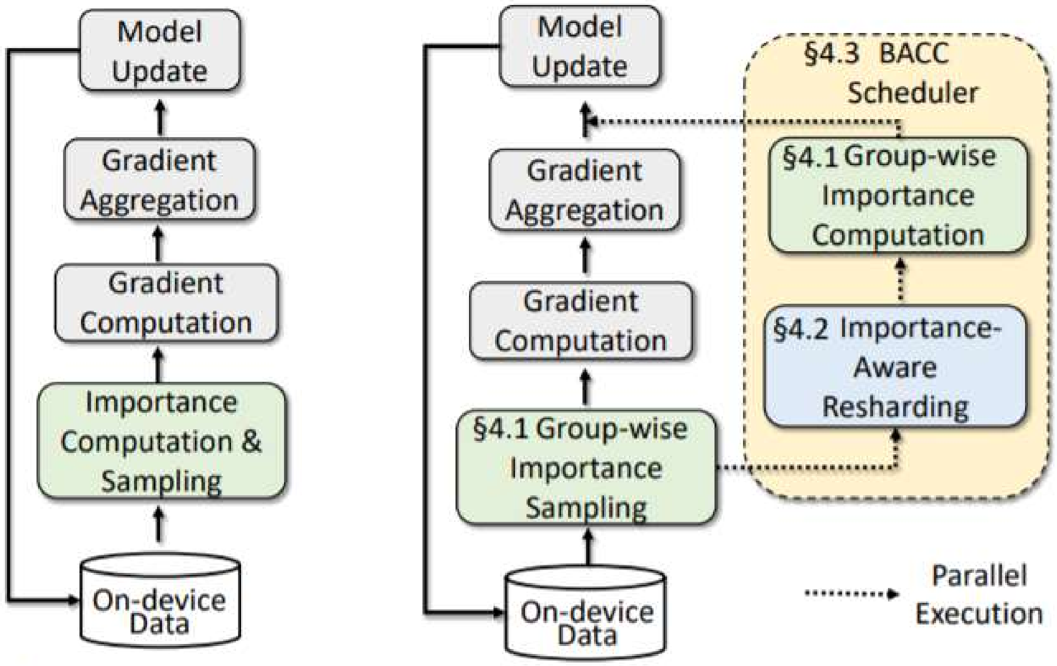
Mercury

- To fully overlap R with C: Resharding pauses when aggregation begins and resumes when it ends



Mercury

- Framework



(a) Rudimentary Importance Sampling-based Framework

(b) Mercury

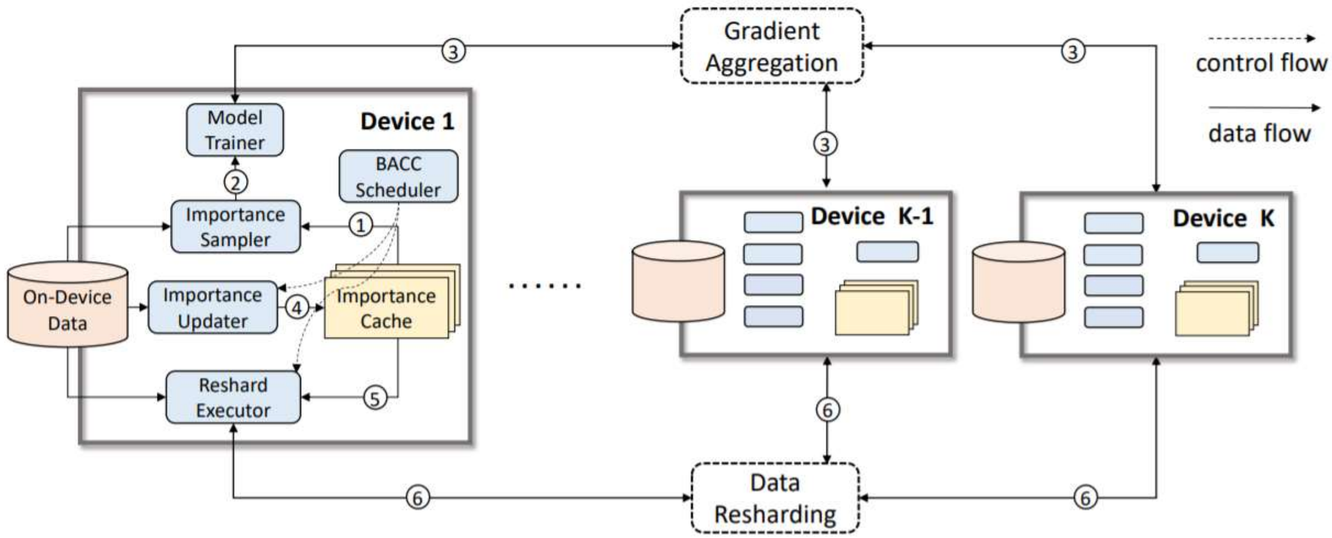
ANTS LAB

UNIVERSITY of HOUSTON
CULLEN COLLEGE of ENGINEERING

Mercury

- System architecture

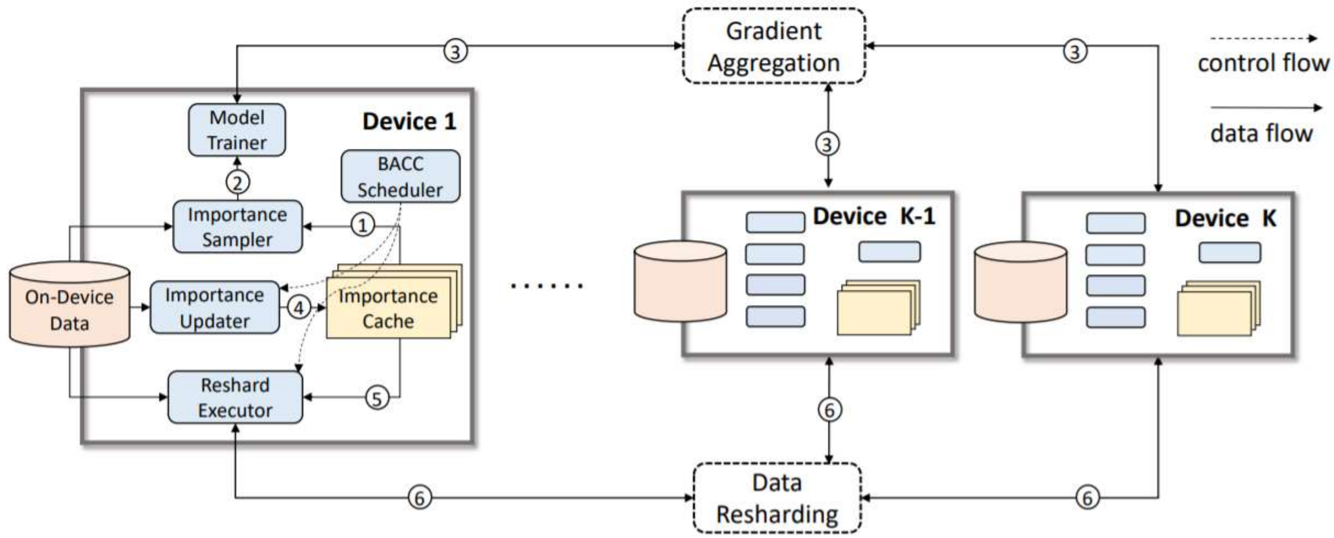
1) Construct mini-batch from on-device data in importance cache



Mercury

- System architecture

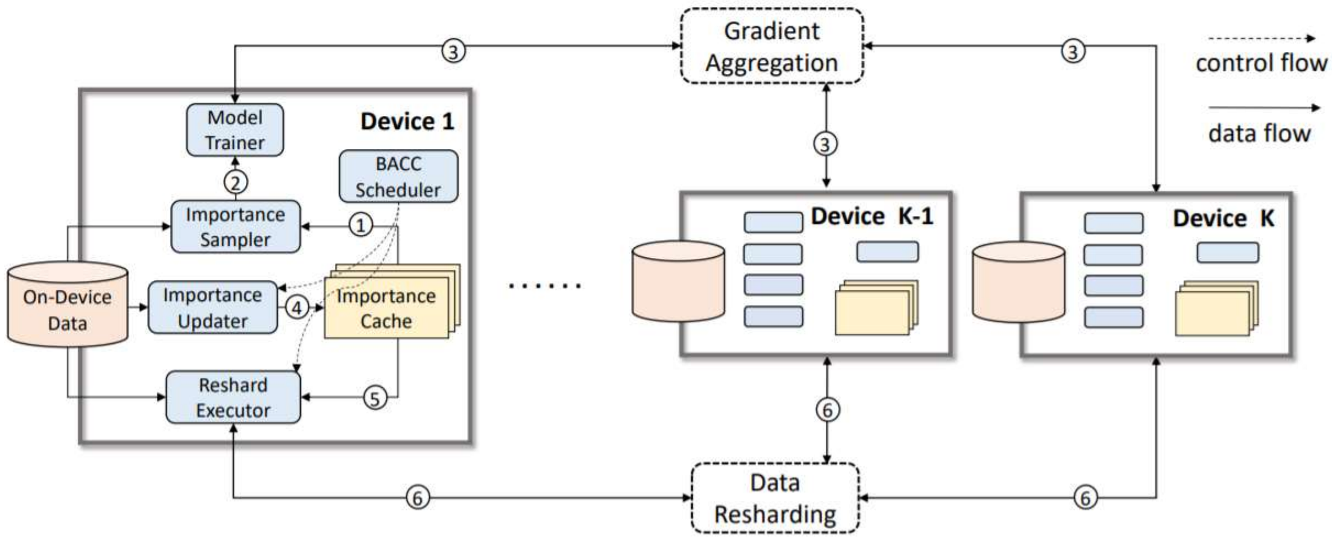
2) Mini-batch is fed to model trainer to compute local gradient



Mercury

- System architecture

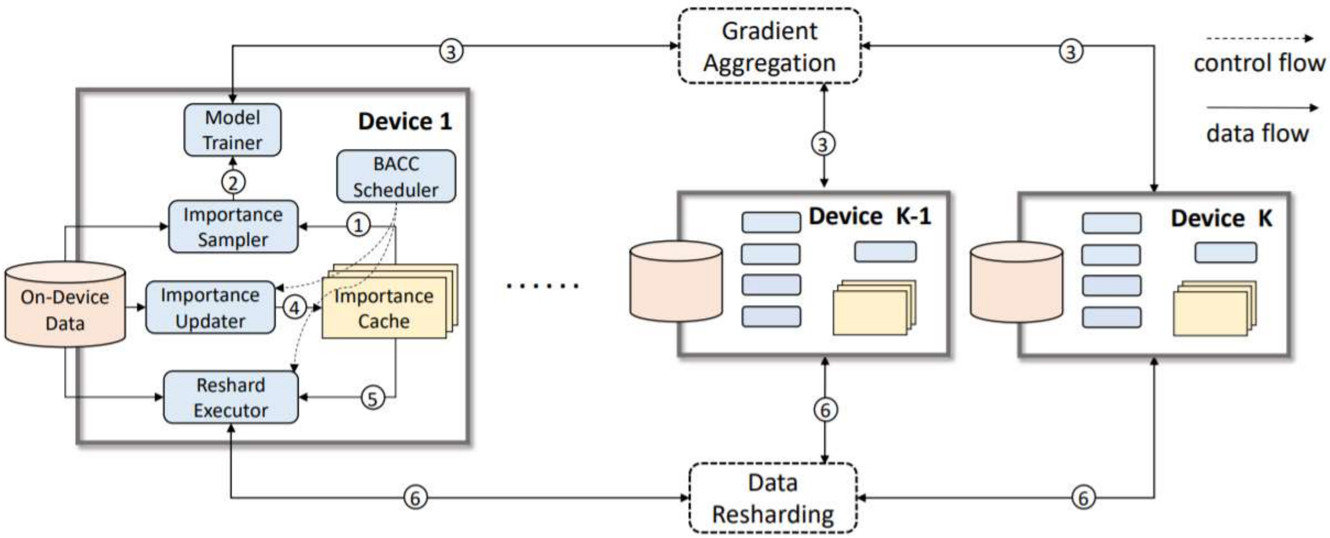
3) Gradients aggregated, model updated



Mercury

- System architecture

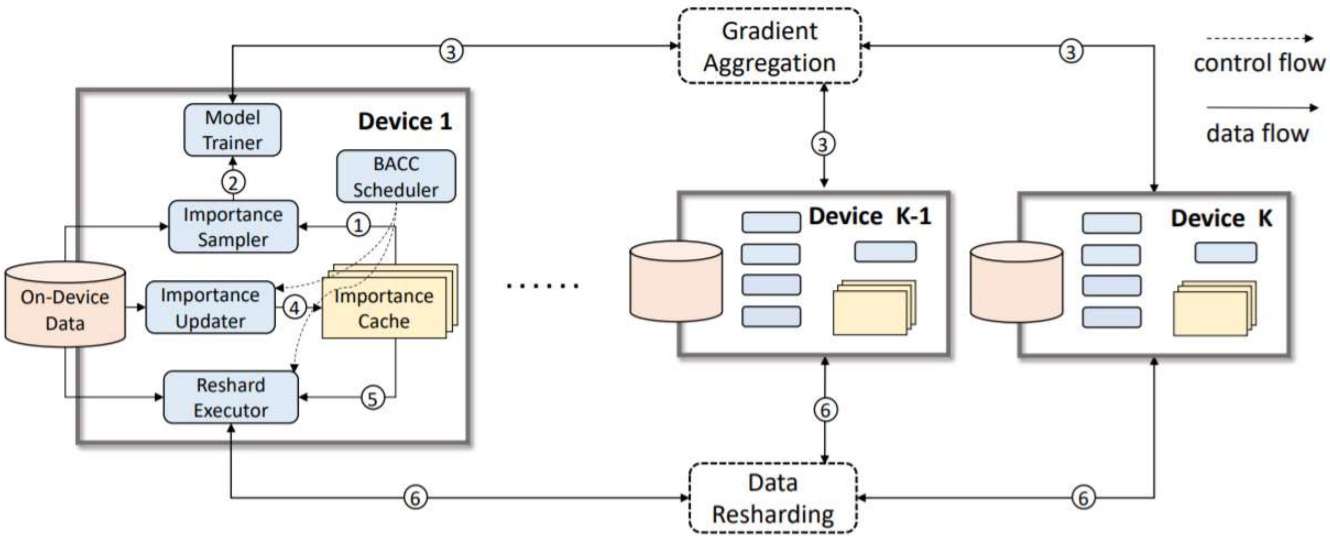
4) Re-compute the data importance and update the Importance Cache



Mercury

- System architecture

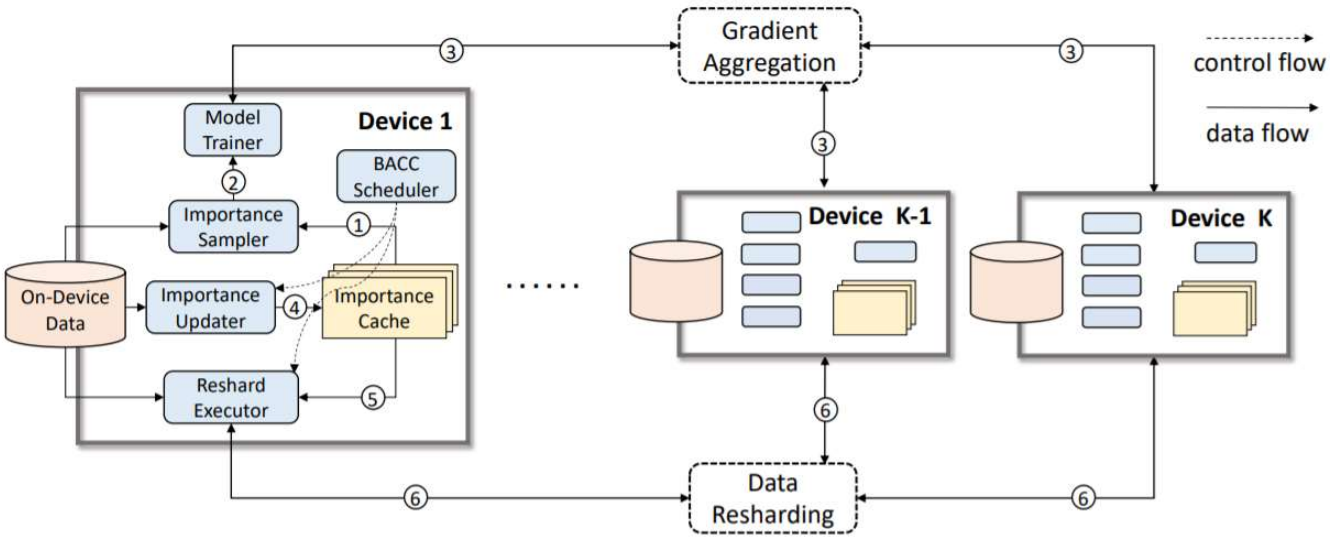
5) Identify important data samples from Importance Cache



Mercury

- System architecture

6) Communicate with other devices to perform importance-aware data resharding



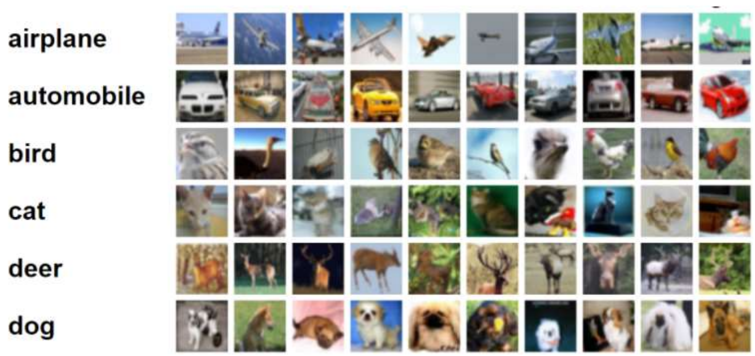
ANTS LAB

UNIVERSITY of HOUSTON
CULLEN COLLEGE of ENGINEERING

Implementation Setup

Applications & models

- Image Classification (ResNet)
- Speech Recognition (VGG)
- Next Language Processing (LSTM)




Implementation Setup

Datasets:

- Image Classification

  Cifar10, Cifar100, SVHN, Aerial Image Dataset
- Speech Recognition

  Tensorflow Speech Command Dataset
- Next Language Processing

  AG News Corpus

Implementation Setup

Devices:

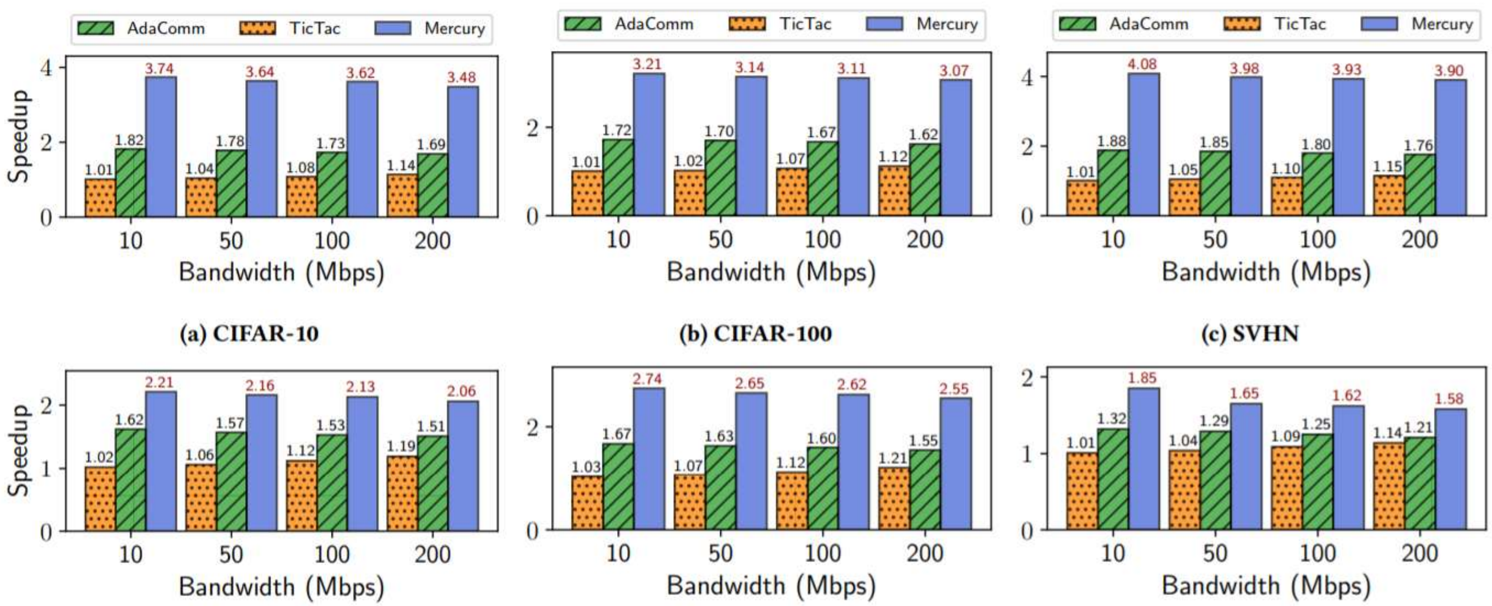
- 12 NVIDIA Jetson TX1
- Wifi routers to connect all TX1

Baselines:

- TicTac (overlapping communication & computation)
- Adacomm (Local SGD)

Results

- End-to-end performance



(a) CIFAR-10 (b) CIFAR-100 (c) SVHN

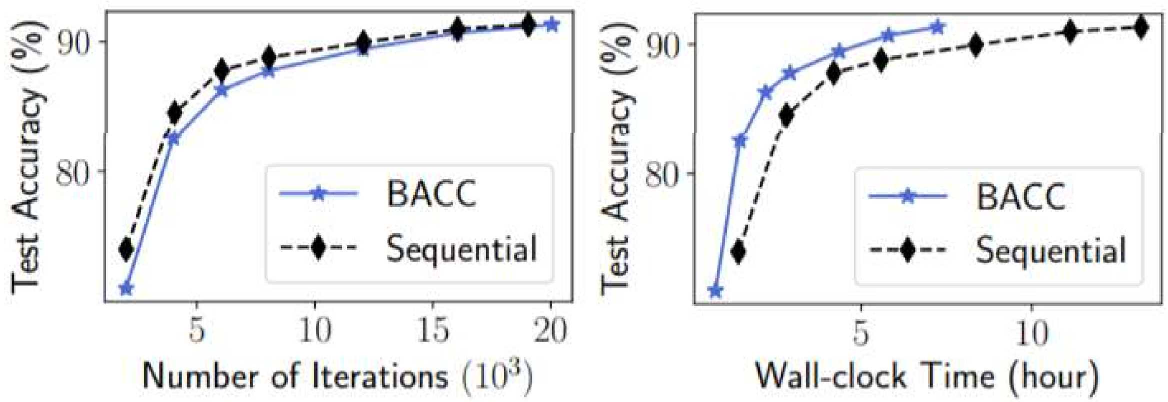
(d) AID (e) Tensorflow Speech Command (f) AGNews

ANTS LAB

UNIVERSITY of HOUSTON
CULLEN COLLEGE of ENGINEERING

Results

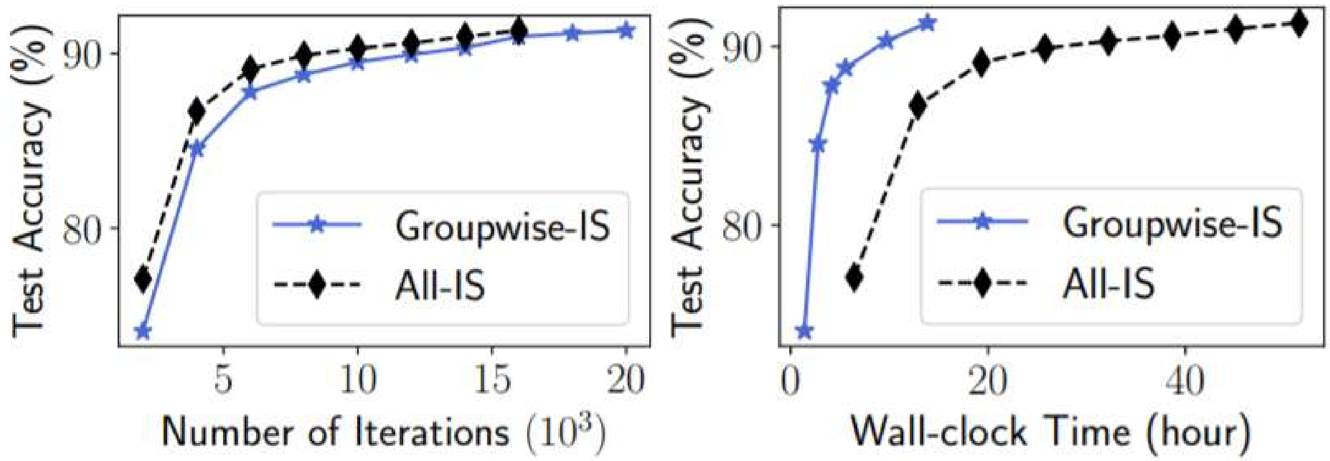
- Sequential implementation & BACC scheduler



(a) **(b)**

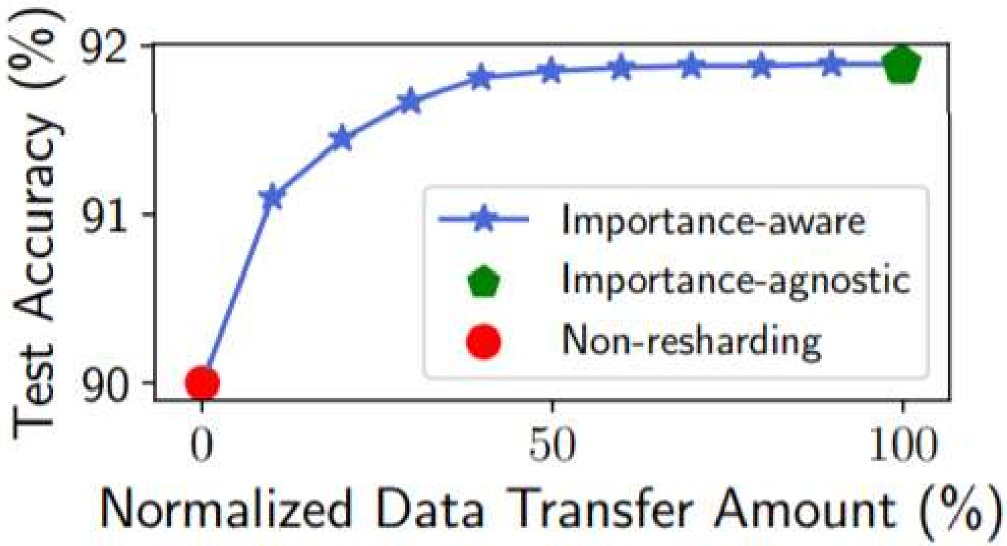
Results

- Group importance & All-inclusive



Results

- Importance-Aware & Importance-Agnostic



Conclusion

- Mercury enables efficient training
- Mercury doesn't damage accuracy too much
- Mercury addresses challenges using

  1) Group-wise importance sampling

  2) Importance-aware resharding

  3) BACC scheduler


THANK YOU
UNIVERSITY of HOUSTON | ENGINEERING