Advances in Neural Information Processing Systems 37
(NeurIPS 2023)

Global Update Tracking (GUT)

Sai Aparna Aketi, Abolfazl Hashemi, Kaushik Roy

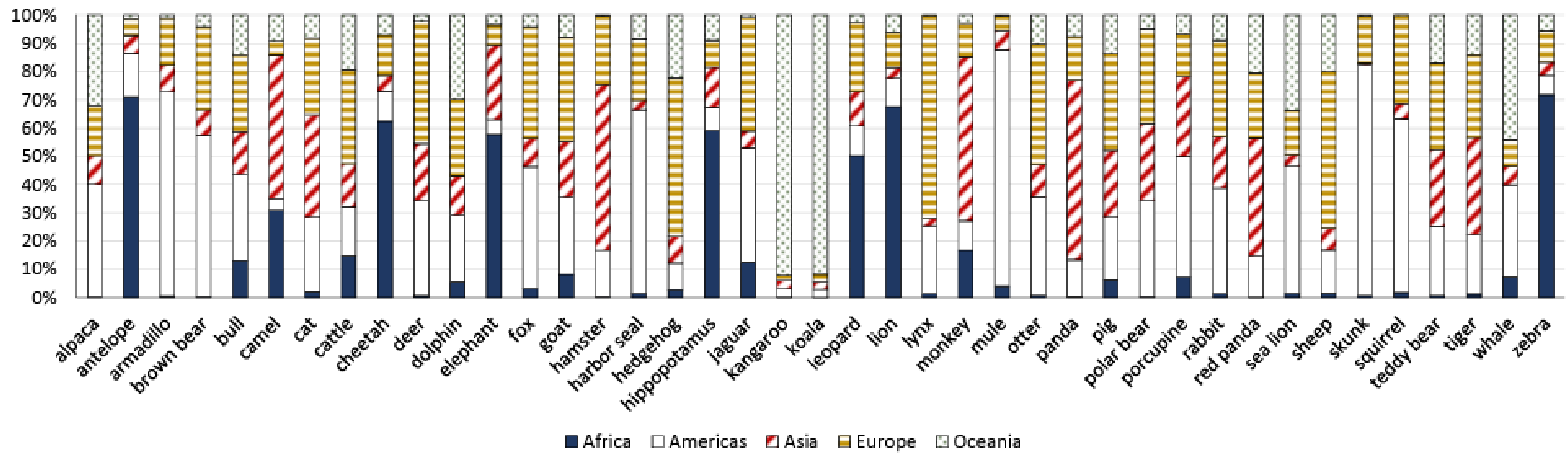
SRC JUMP

PURDUE UNIVERSITY

CoCoSys
CENTER FOR THE
CO-DESIGN OF COGNITIVE SYSTEMS

Problem of non-IID Data

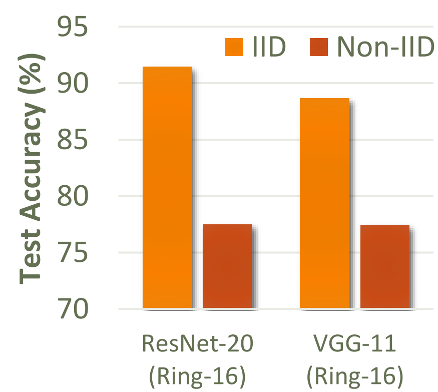
Flickr-Mammal Dataset: The share of images in each continent [1]



Significant skew in data distribution

Traditional decentralized learning algorithms :

- Assume IID data
- Performance degradation with non-IID
- 10-15% drop with CIFAR-10 on 16 nodes ring



[1] HSIEH, KEVIN, ET AL. "THE NON-IID DATA QUAGMIRE OF DECENTRALIZED MACHINE LEARNING." *INTERNATIONAL CONFERENCE ON MACHINE LEARNING*. PMLR, 2020.

Decentralized Learning on Heterogeneous Data

Method	Communication	Memory	Compute
D^2 (Exact Diffusion)	1x	m	Bias estimation
Gradient Tracking	2x	2m	Bias estimation
Cross Gradient Aggregation	2x	nm	Cross gradients computation, QP step
NGM	2x	0	Cross gradients computation, bias estimation
Relay SGD	1x	2m	Relay computation
Quasi Global Momentum	1x	m	-
Momentum Tracking	2x	2m	Bias estimation

Modifies Local SGD step

Modifies gossip

Modifies Momentum

m = model size, n = number of neighbors

Can we achieve the effects of compute efficient gradient tracking (bias correction) without additional communication round?

D-PSGD vs Gradient Tracking

D-PSGD

Local Update Step (SGD)

$$x_i^{t+0.5} = x_i^t - \eta g_i^t$$

Communication step

Send and receive $x^{t+0.5}$

Gossip Averaging Step

$$x_i^{t+1} = \sum_{j \in N(i)} w_{ij} x_j^{t+0.5}$$

2x communication

Scalability

Gradient Tracking (GT)

Local Update Step (SGD)

$$x_i^{t+0.5} = x_i^t - \eta y_i^t$$

Communication step

Send and receive $x^{t+0.5}, y^t$

Gossip Averaging Step

$$x_i^{t+1} = \sum_{j \in N(i)} w_{ij} x_j^{t+0.5}$$

Update Tracking Variable

$$y_i^{t+1} = \sum_{j \in N(i)} w_{ij} y_j^t + g_i^{t+1} - g_i^t$$

Step 1: Update Sharing

Idea: Share model updates rather than model parameters by keeping track of neighbors' model parameters

-- Local update

-- Gossip update

D-PSGD Update: $x_i^{t+1} = x_i^t - \eta\left[g_i^t + \frac{1}{\eta}\sum_{j\in N(i)} w_{ij}(x_i^t - x_j^t)\right]$ *Communicate $x's$*

Communicate model updates $\boldsymbol{i.e.,}$ $\boldsymbol{x_j^t - x_j^{t-1}}$

and store neighbors' parameters as $\boldsymbol{\hat{x}_j^{t-1}}$

$$x_i^{t+1} = x_i^t - \eta \underbrace{\left[g_i^t + \frac{1}{\eta}\sum_{j\in N(i)} w_{ij} * (x_i^t - \hat{x}_j^t)\right]}_{\delta_i^t} \text{ and } \hat{x}_j^t = \hat{x}_j^{t-1} - \eta\delta_j^t$$

Communicate $\delta's$

Model update
= local update + gossip update

Copy of neighbors' parameters
($\hat{x}_j^t = x_j^t$)

Memory efficient implementation of this algorithm stores $s_i = \sum_{j\in N(i)} w_{ij}\hat{x}_j$ instead of each neighbors' copy separately

Step 2: Incorporate Tracking

Modified D-PSGD Update: $x_i^{t+1} = x_i^t - \eta \delta_i^t \quad and \quad \delta_i^t = g_i^t + \frac{1}{\eta}\sum_{j \in N(i)} w_{ij} * (x_i^t - \hat{x}_j^t)$

Communicate δ's

Add tracking to variable δ

$$x_i^{t+1} = x_i^t - \eta y_i^t \quad and \quad y_i^t = \sum_{j \in N(i)} w_{ij} * y_j^{t-1} + \delta_i^t - \delta_i^{t-1}$$

Communicate y's

Scaling and reference correction

Global Update Tracking: $x_i^{t+1} = x_i^t - \eta y_i^t$

$$y_i^t = \delta_i^t + \underset{\text{Scaling factor}}{\mu} \left[\sum_{j \in N(i)} w_{ij} * (y_j^{t-1} + \underbrace{\frac{1}{\eta}(x_i^t - \hat{x}_j^t)}_{\text{Reference correction}}) - \delta_i^{t-1} \right]$$

Convergence Guarantees

- Objective: Minimize global loss function $f(x)$ distributed across *n* agents

$$\min_x f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) \quad \text{where } f_i(x) = \mathbb{E}_{d_i \sim D_i}[F_i(x, d_i)]$$

- Assumptions

1. Lipschitz Gradients: The loss function on each agent is L-smooth i.e., $\left|\left|\nabla f_i(y) - \nabla f_i(x)\right|\right| \leq L\, ||y - x||$
2. Bounded Variance: $\mathbb{E}_{d \sim D_i}||\nabla F_i(x, d) - \nabla f_i(x)||^2 \leq \sigma^2$ and $\frac{1}{n}\sum_{i=1}^{n} ||\, \nabla f_i(x) - \nabla f(x)||^2 \leq \zeta^2$
3. Doubly Stochastic Mixing Matrix (W): $\lambda_1 = 1,\ \ max\{|\lambda_2|, |\lambda_n|\} \leq 1 - \rho\ < 1$

- We show that GUT achieves linear speed up with a convergence rate of $\mathcal{O}\left(\frac{1}{\sqrt{nT}}\right)$

Lemma 1. *Given assumptions 3, we define* $\bar{b}^t = B^t \frac{1}{n}\mathbb{1}\mathbb{1}^T$, *where* $\mathbb{1}$ *is a vector of all ones. For all* t, *we have:* $\bar{b}^t = \mu \bar{b}^{t-1}$.

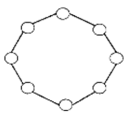
Theorem 1. *(Convergence of GUT algorithm) Given Assumptions 1, 2, and 3 let step size* $\eta \leq \frac{\rho}{7L}$ *and the scaling factor* $\frac{\mu}{1-\mu} \leq \frac{\rho}{42}$. *For all* $T \geq 1$, *we have*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}||\nabla f(\bar{x}^t)||^2 \leq \frac{4}{\eta T}(f(\bar{x}^0) - f^*) + \eta\frac{4L\sigma^2}{n} + \eta^2\frac{1248L^2}{\rho^2}(\zeta^2 + \sigma^2(2-\mu)),$$

where $f(\bar{x}^0) - f^*$ *is the sub-optimality gap,* $\bar{x}$ *is the average/consensus model parameters.*

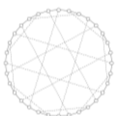
Experimental Setup

Graph Topology



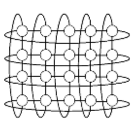
Ring

- 16-40 nodes (2 peers)



Dyck

- 32 agents (3 peers)



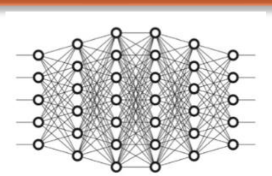
Torus

- 32 agents (4 peers)

Datasets

- CIFAR10
- CIFAR100
- Fashion MNIST
- Imagenette

Models



- ResNet-20
- VGG 11
- LeNet-5
- MobileNet-V2

- All the hyperparameters are synchronized across the nodes
- Stopping criteria: Fixed number of epochs
- The results are averaged over 3 seeds
- Dirichlet Distribution: Smaller the α, larger the heterogeneity in the data distribution

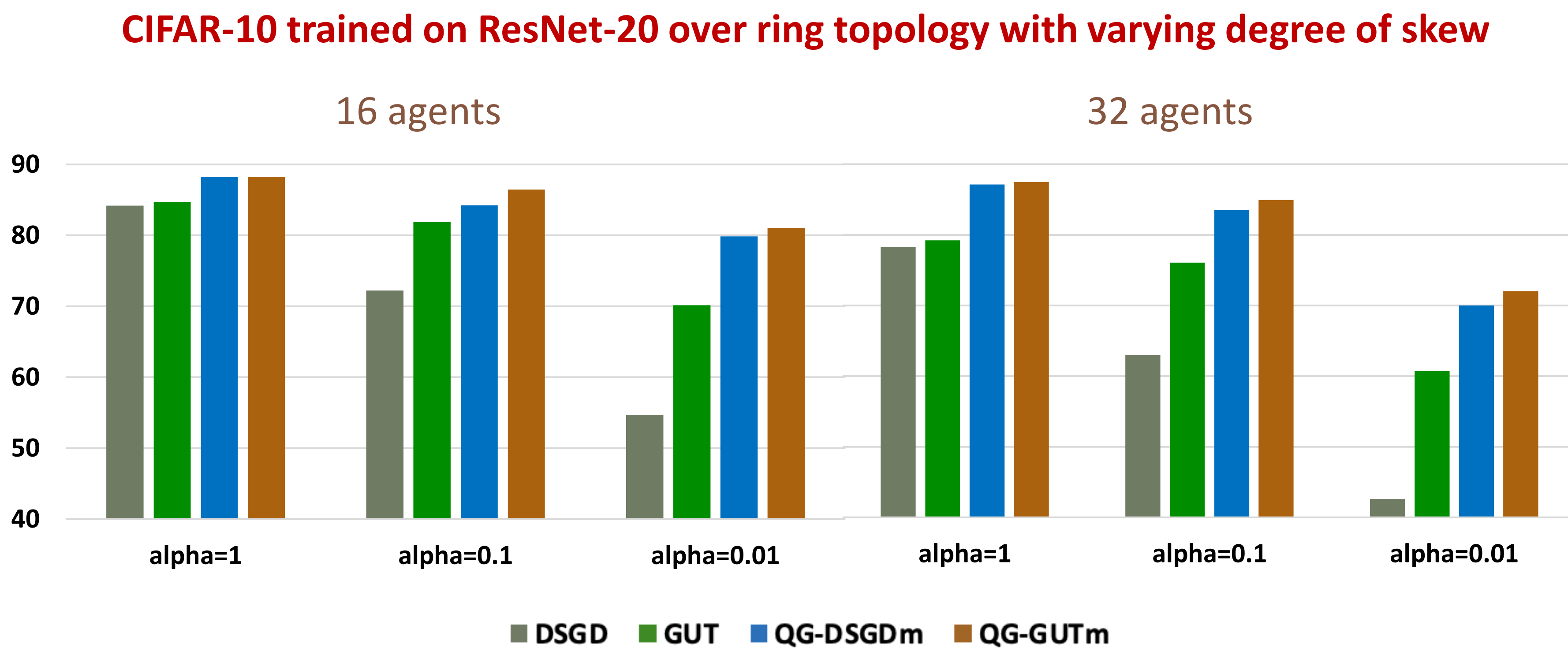
Comparison with existing techniques

Decentralized Learning Algorithms

1x Communication	D-PSGD: Assumes IID distributions	→ Baseline for GUT
	Relay-SGD: Works on spanning trees	
	D^2: Not compatible for all graphs	
	QGM: Uses quasi-global momentum	Can be used in synergy
	NGM_{mv} : Compute Heavy	
	Global Update Tracking (this work)	

D-PSGD + QGM: QG-DSGDm is compared with GUT+QGM: QG-GUTm

Results: CIFAR-10



1.2% average improvement over QG-DSGDm

Results: Various Datasets and Graph Topologies

Generalizability:

- Compare QG-GUTm with QG-DSGDm
- Various graph topologies: 1.5% improvement on an average
- Various datasets: 2.5% improvement on an average

Analysis of CIFAR-10 trained on ResNet-20 over various graph topologies

Method	Dyck Graph (32 agents)		Torus (32 agents)	
	$\alpha = 0.1$	$\alpha = 0.01$	$\alpha = 0.1$	$\alpha = 0.01$
QG-DSGDm	86.49 ± 0.81	81.32 ± 1.50	86.88 ± 0.30	85.20 ± 0.56
QG-GUTm (ours)	$\mathbf{86.93} \pm 0.53$	$\mathbf{84.80} \pm 0.47$	$\mathbf{87.75} \pm 0.42$	$\mathbf{86.20} \pm 0.82$

Analysis of various datasets trained over ring topology with 16 agents

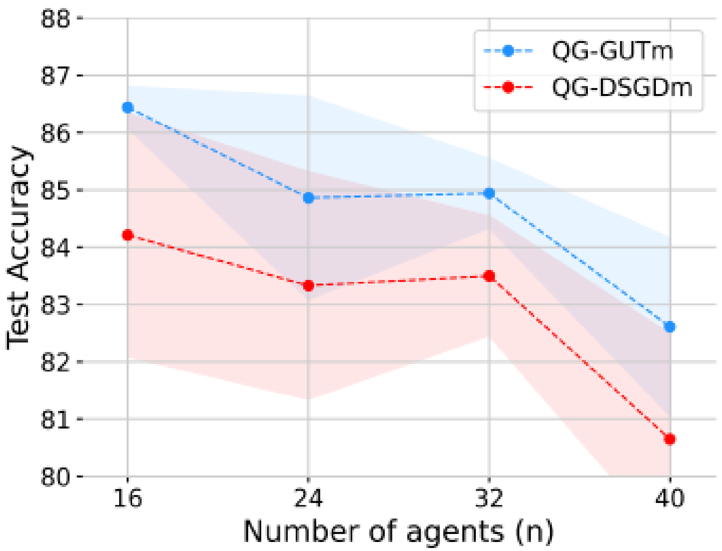
Method	Fashion MNIST (LeNet-5)		CIFAR-100 (ResNet-20)		Imagenette (MobileNet-V2)	
	$\alpha = 0.1$	$\alpha = 0.01$	$\alpha = 0.1$	$\alpha = 0.01$	$\alpha = 0.1$	$\alpha = 0.01$
QG-DSGDm	89.94 ± 0.44	83.43 ± 0.94	53.19 ± 1.68	44.17 ± 3.64	63.60 ± 4.50	39.49 ± 4.57
QG-GUTm	$\mathbf{90.11} \pm 0.02$	$\mathbf{84.60} \pm 1.00$	$\mathbf{53.40} \pm 1.23$	$\mathbf{50.45} \pm 1.30$	$\mathbf{66.52} \pm 3.68$	$\mathbf{43.85} \pm 8.24$

Ablation Study

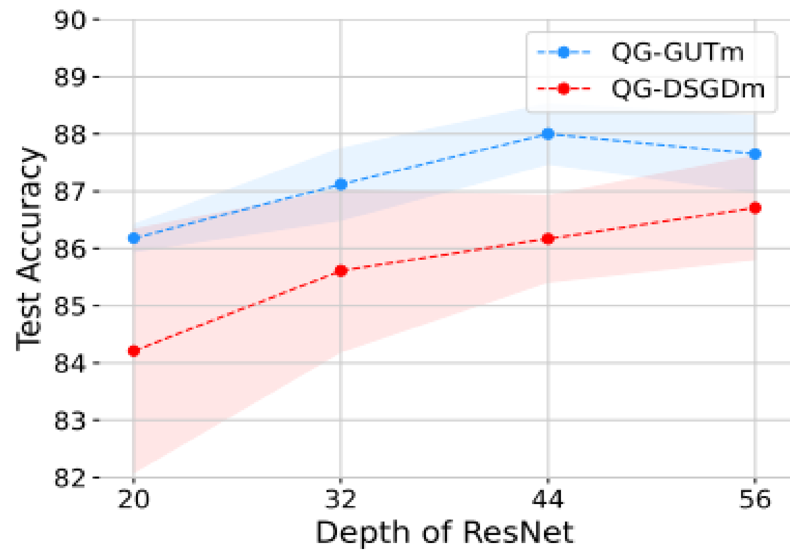
Scalability:

- Compare QG-GUTm with QG-DSGDm
- Number of agents: 1.7% improvement on an average
- Depth of ResNet: 1.4% improvement on an average

CIFAR-10 dataset trained on ResNet architecture over ring topology



$\alpha = 0.1$, ResNet-20



$\alpha = 0.1, n = 16$

Analysis of overheads

Overheads comparison

- No communication overhead
- $\mathcal{O}(1)$ memory overhead in terms of model
- Minimal compute overhead – less than 2% for compact models
- Memory and compute overheads are independent of graph type and size

Memory and compute overhead incurred per agent during training

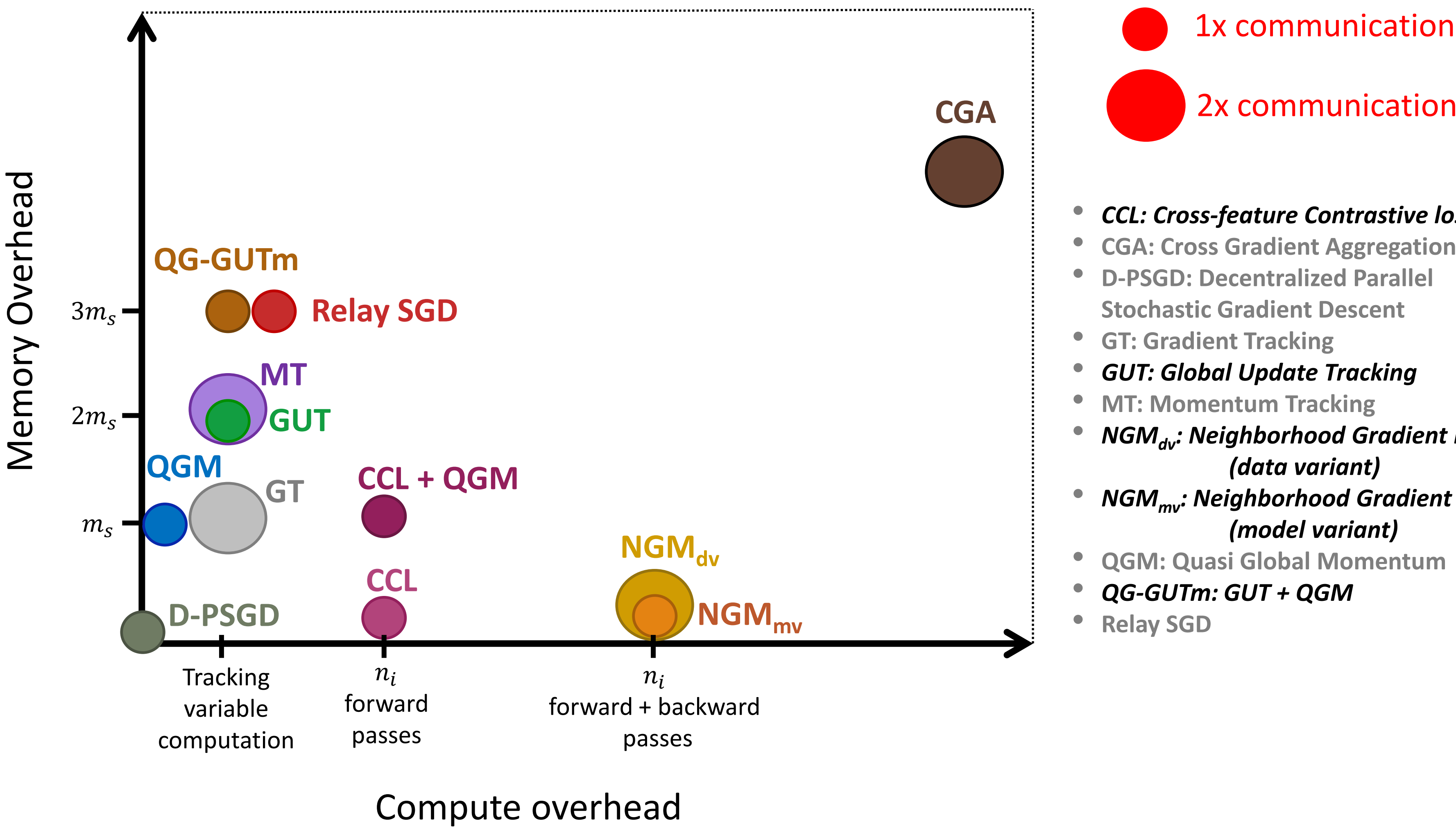
Dataset	Model	Memory Overhead	Compute Overhead
Fashion MNIST	LeNet-5	0.099	0.275
CIFAR-10	ResNet-20	0.016	0.021
CIFAR-10	VGG-11	0.138	0.149
CIFAR-100	ResNet-20	0.016	0.022
Imagenette	MobileNet-V2	0.005	0.021

Summary

Proposed Global Update Tracking

- ✓ Generate a proxy to gradient tracking variable utilizing shared model updates of the neighborhood
- ✓ No communication overhead
- ✓ $\mathcal{O}(1)$ memory overhead
- ✓ Exhaustive experiments show the efficiency, scalability and generalizability of the proposed method
- ✓ Performance improvement of 1-6% on non-IID data over the current SoTA
- ✓ Theoretically show that GUT has same convergence rate as the state-of-the-art decentralized methods.

Conclusion



- ***CCL: Cross-feature Contrastive loss***
- CGA: Cross Gradient Aggregation
- D-PSGD: Decentralized Parallel Stochastic Gradient Descent
- GT: Gradient Tracking
- ***GUT: Global Update Tracking***
- MT: Momentum Tracking
- ***NGM$_{dv}$: Neighborhood Gradient Mean (data variant)***
- ***NGM$_{mv}$: Neighborhood Gradient Mean (model variant)***
- QGM: Quasi Global Momentum
- ***QG-GUTm: GUT + QGM***
- Relay SGD

Thank You!