Communication Efficient Distributed Training

Ji Liu, Ph.D.
Objective

\[
\min_{\mathbf{x}} \quad f(\mathbf{x}) := \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{a(i) \sim D_i} F(\mathbf{x}; a^{(i)})
\]

All functions are assumed to be L-Lipschitzian

How to reduce communication cost?
Summary

Foundations and Trends® in Databases
Distributed Learning Systems with
First-Order Methods
An Introduction


Ji Liu
University of Rochester and Kuaishou Inc., USA
ji.liu.uwisc@gmail.com

Ce Zhang
ETH Zurich, Switzerland
ce.zhang@inf.ethz.ch

Coming Soon.
Compression
Algorithm

Server

Worker 1

Worker 2

Worker 3

\[ g_i := \nabla F(x; a^{(i)}) \]

(Standard) \( x \leftarrow x - \gamma \bar{g} \)

Exchange 2N full vectors

\[ \bar{g} = \frac{1}{3} (g_1^{(1)} + g_2^{(2)} + g_3^{(3)}) \]

(Single compression)

Exchange N(1+c) full vectors

\[ \bar{g} = \frac{1}{3} (C(g_1^{(1)}) + C(g_2^{(2)}) + C(g_3^{(3)})) \]

(Double compression)

Exchange 2cN full vectors

\( C(\cdot) \) Compression operator (maybe randomized)
Unfortunately

To ensure convergence, it should satisfy $E(C(x)) = x$

Early methods only work for $C(\cdot)$ compression operator

- Randomized quantization (unbiased)
  - Randomized quantization (biased)
- 1bit quantization
- Clipping
- Top-k sparsification

Can we relax it to allow more aggressive or even arbitrary compression?
Double Squeeze: Error Compensated SGD

Worker $i$

$$
\begin{align*}
g^{(i)} &\leftarrow \nabla F(x; \alpha^{(i)}) \\
v^{(i)} &\leftarrow C \left[ g^{(i)} + \delta^{(i)} \right] \\
\delta^{(i)} &\leftarrow \left( g^{(i)} + \delta^{(i)} \right) - v^{(i)} \\
x &\leftarrow x - \gamma \bar{v}
\end{align*}
$$

Master $s$

$$
\begin{align*}
\bar{g} &\leftarrow \frac{1}{n} \sum_{i=1}^{n} v^{(i)} \\
\bar{v} &\leftarrow C \left[ \bar{g} + \bar{\delta} \right] \\
\bar{\delta} &\leftarrow (\bar{g} + \bar{\delta}) - \bar{v}
\end{align*}
$$

Local gradient
Error compensation
Compression error
Pull $\bar{V}$ to update model
Aggregate compressed gradient
Compress the error compensated $g$
Compression error
Intuition

Essential updating rule of DoubleSqueeze (SGD alike)

\[ x_{t+1} = x_t - \gamma g_t + \gamma (\hat{\delta}_t - \hat{\delta}_{t-1}) \]

\[ \bar{g}_t = \frac{1}{n} \sum_{i=1}^{n} g_{ti} \]

\[ \hat{\delta}_t = \frac{1}{n} \sum_{i=1}^{n} \delta_{ti} + \bar{\delta}_t \]

C-SGD (Uncompressed)

\[ x_{t+1} = x_0 - \gamma \sum_{s=0}^{t} g_s \]

Naive Compressed C-SGD

\[ x_{t+1} = x_0 - \gamma \sum_{s=0}^{t} \bar{g}_s + \gamma \sum_{s=0}^{t} \hat{\delta}_s \]

DoubleSqueeze

\[ x_{t+1} = x_0 - \gamma \sum_{s=0}^{t} \bar{g}_s + \gamma \hat{\delta}_t \]

Much smaller
Convergence

Assumption
\[ \mathbb{E}[\|C(x) - x\|^2] \leq \sigma^2 \]

Convergence rates
\[
\frac{1}{T} \sum_{t=1}^{T} \mathbb{E} (\| \nabla f(\bar{x}_t) \|^2) \lesssim \frac{1}{T} + \frac{\sigma}{\sqrt{nT}}
\]

SGD

\[
\frac{1}{T} \sum_{t=1}^{T} \mathbb{E} (\| \nabla f(\bar{x}_t) \|^2) \lesssim \frac{1}{T} + \frac{\sigma}{\sqrt{nT}} + \frac{\sigma'}{\sqrt{T}}
\]

C-SGD (C(.) needs to be unbiased)

\[
\frac{1}{T} \sum_{t=1}^{T} \mathbb{E} (\| \nabla f(\bar{x}_t) \|^2) \lesssim \frac{1}{T} + \frac{\sigma}{\sqrt{nT}} + \left( \frac{\sigma'}{T} \right)^{\frac{2}{3}}
\]

Double squeeze

EC-SGD

better
Experiments

ResNet-18. CIFAR-10. 8 workers

Iteration (epoch) is consistent with SGD

Running time in each iteration is faster
Decentralization
Centralized communication
(fully exchanged)

\[ O(N \cdot \alpha + NB \cdot \beta) \]

Decentralized communication
(partially exchanged)

\[ O(\alpha + B \cdot \beta) \]

- \( \alpha \): latency per message
- \( \beta \): transfer time per byte
- \( N \): # workers
- \( B \): # bytes of the message

**How does the decentralized approach compare to the centralized approach?**
Objective

$$\min_x f(x) := \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}_{a^{(i)} \sim D_i} F(x; a^{(i)}) =: f_i(x)$$
Centralized-SGD:

\[ \mathbf{x} \leftarrow \mathbf{x} - \gamma \frac{1}{n} \sum_{i=1}^{n} g(\mathbf{x}; a^{(i)}) \]

- shared model
- local sample
Centralized-SGD:

\[ \mathbf{x} \leftarrow \mathbf{x} - \gamma \frac{1}{n} \sum_{i=1}^{n} g(\mathbf{x}; \mathbf{a}^{(i)}) \]

Decentralized-SGD:

\[ \mathbf{x}^{(2)} \leftarrow \frac{1}{3} \sum_{i=1,2,3} \left( \mathbf{x}^{(i)} - \gamma g^{(i)} \right) \]
Decentralized SGD

\[
\begin{bmatrix}
    x^{(1)} \\
    x^{(2)} \\
    \ldots \\
    x^{(n)}
\end{bmatrix}
\leftarrow W
\begin{bmatrix}
    x^{(1)} \\
    x^{(2)} \\
    \ldots \\
    x^{(n)}
\end{bmatrix}
- \gamma
\begin{bmatrix}
    g(x^{(1)}; a^{(1)}) \\
    g(x^{(2)}; a^{(2)}) \\
    \ldots \\
    g(x^{(n)}; a^{(n)})
\end{bmatrix}
\]

weight matrix: symmetric, doubly stochastic
\((W1 = 1, W^T1 = 1, \text{nonnegative, } W = W^T)\)

ring network

\[
W = \begin{pmatrix}
    1/3 & 1/3 & \ldots & 1/3 \\
    1/3 & 1/3 & 1/3 & \ldots \\
    \ldots & \ldots & \ldots & \ldots \\
    1/3 & 1/3 & 1/3 & 1/3
\end{pmatrix}
\]
Assumptions
- Lipschitzian All $f_i(\cdot)$ are with $L$-Lipschitzian gradient
- Bounded variance

$$\mathbb{E}_{a \sim D_i} \left\| \nabla F(x; a) - \nabla f_i(x) \right\|^2 \leq \sigma_i^2, \forall i, \forall x$$

$$\left\| \nabla f_i(x) - \nabla f(x) \right\|^2 \leq \zeta_i^2, \forall i, \forall x$$

data variance **within** each worker

data variance **among** workers
Assumptions
- Lipschitzian All $f_i(\cdot)$ are with $L$-Lipschitzian gradient
- Bounded variance

$$
\mathbb{E}_{a \sim D_i} \| \nabla F(x; a) - \nabla f_i(x) \|^2 \leq \sigma^2, \forall i, \forall x
$$

$$
\| \nabla f_i(x) - \nabla f(x) \|^2 \leq \zeta^2, \forall i, \forall x
$$

- Spectral gap

$$
\rho := \max_{j \geq 2} |\lambda_j(W)|
$$

Measure how fast the information can spread across the network
Fully connected network

\[ W = \frac{11^\top}{N} \]
\[ \rho = 0 \]

Ring network

\[ W = \begin{pmatrix} 1/3 & 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 & 1/3 \\ \vdots & \vdots & \vdots & \vdots \\ 1/3 & 1/3 & 1/3 & 1/3 \end{pmatrix} \]
\[ \rho \approx \left(1 - \frac{16\pi^2}{3N^2}\right) \]

Disconnected network

\[ W = \begin{pmatrix} D & 0 \\ 0^\top & 1 \end{pmatrix} \]
\[ \rho = 1 \]
**Theorem [DSGD]** Choose the learning rate approximately. When $T$ is sufficiently large, we have

\[
\frac{1}{T} \sum_{t=1}^{T} \mathbb{E} \left( \| \nabla f(x_t) \|^2 \right) + \frac{1}{T} \sigma + \frac{\zeta \rho}{T(1 - \rho)} \frac{2}{3}
\]

Average of local models

Convergence rate of CSGD

Cost of using decentralized communication (minor)
Centralized includes PS and AllReduce!

Centralized

Decentralized

Training Loss

Epochs

DECENTRALIZED METHOD
Ring Topology

100 GPUs
ResNet
CIFAR10
Decentralized algorithms outperform centralized algorithms for networks with low bandwidth and high latency.
Decentralized-SGD achieves the same convergence rate as Centralized-PSGD.

When the network is with high latency, decentralized communication can outperform its centralized counterpart.
Compression + Decentralization
Naïve compression does not work

Can we further reduce the communication cost?

Naïve compression for D-SGD

\[ x_{t+1}^{(i)} = \sum_j W_{ij} C \left( x_{t}^{(j)} \right) - \gamma \nabla F(x_{t}^{(i)} ; a^{(i)}) \]
DCD-SGD

Store a copy of its neighbors’ models

\[ \hat{x}_{t+1}^{(i)} = \sum_j W_{ij} x_t^{(j)} - \gamma \nabla F(x_t^{(i)}; a^{(i)}) \]

\[ x_{t+1}^{(i)} = \hat{x}_t^{(i)} + C \left( \hat{x}_t^{(i)} - x_t^{(i)} \right) \]

Compress the difference and send to its neighbors

\[ \mathbb{E} \left[ \| C(x) - x \|^2 \right] \leq \alpha^2 \]

\[ \mathbb{E}(C(x)) = x \]

Consistent with D-SGD

\[ \frac{1}{T} \sum_{t=1}^{T} \mathbb{E} \left( \| \nabla f(x_t) \|^2 \right) \leq \frac{1}{T} + \frac{\sigma(1 + \alpha^2)}{\sqrt{nT}} + \frac{\zeta_2^3 (1 + \alpha^2)}{T^{2/3}} \]

[NIPS 2018]
Experiments

(b) Time vs Training Loss
Bandwidth = 1.4Gbps,
Latency = 0.13ms

(c) Time vs Training Loss
Bandwidth = 1.4Gbps,
Latency = 20ms

(d) Time vs Training Loss
Bandwidth = 5Mbps,
Latency = 20ms
Limitation of DCD-SGD

Two Issues of DCD-SGD:

Require $E(C(x)) = x$

Diverges when using 4-bit compression in most cases

Can we fix it by using error compression strategy?
$X_{t+1} = (X_t - \gamma G_t)W$

$= X_t - \gamma G_t + (X_t - \gamma G_t)(W - I)$

Share this with Error Compensation
One More Thing: DeepSqueeze

DCD-SGD + Error Compensation

Local:

\[ \mathbf{v}_{t+1}^{(i)} = \mathcal{C} \left( \mathbf{x}_t^{(i)} - \gamma \mathbf{g}_t^{(i)} + \delta_t^{(i)} \right) \]

\[ \delta_{t+1}^{(i)} = \mathbf{v}_{t+1}^{(i)} - \left( \mathbf{x}_t^{(i)} - \gamma \mathbf{g}_t^{(i)} + \delta_t^{(i)} \right) \]

Communicate:

\[ \mathbf{x}_{t+1}^{(i)} = \mathbf{x}_t^{(i)} - \gamma \mathbf{g}_t^{(i)} + \eta \sum_{j \in \mathcal{N}_i} (W_{ij} - I_{ij}) \mathbf{v}_{t+1}^{(j)} \]

Control the compression error explicitly
DeepSqueeze V.S. DCD-PSGD

**DCD-SGD**

\[
\sup_x \frac{\mathbb{E}\|C(x) - x\|^2}{\|x\|^2} \leq \alpha^2
\]

\[
\mathbb{E}(C(x)) = x
\]

**DeepSqueeze**

\[
\sup_x \frac{\mathbb{E}\|C(x) - x\|^2}{\|x\|^2} \leq \alpha^2
\]

**Compression can be biased**

Fails for 4-bit compression

\[
O\left(\frac{1}{T} + \frac{\sigma(1 + \alpha^2)}{\sqrt{nT}} + \frac{\zeta^2(1 + \alpha^2)}{T^{2/3}}\right)
\]

Robust to 2-bit compression

\[
O\left(\frac{1}{T} + \frac{\sigma\left(1 + \frac{\alpha^2\sqrt{n}}{\sqrt{T}}\right)}{\sqrt{nT}} + \frac{\zeta^2(1 + \alpha^2)}{T^{2/3}}\right)
\]
Experiments

DeepSqueeze (save 16x communication cost)

DSGD