A Unified Framework for Decentralized Composite Optimization

Wei Xiong

Mathematics, The Hong Kong University of Science and Technology

wxiongae@connect.ust.hk

Dec 9, 2022

Outline

Table of Contents

Decentralized Composite Optimization

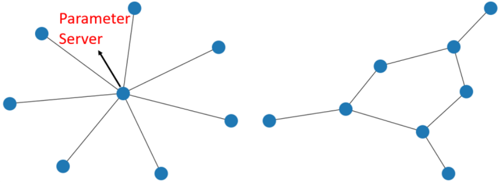
We consider the decentralized composite optimization with m agents:

$$\min_{x \in \mathbb{R}^d} h(x) = f(x) + r(x) := \frac{1}{m} \sum_{i=1}^{m} f_i(x) + r(x) \tag{1}$$

- Each agent has a private local dataset: $f_i(x) := \frac{1}{n} \sum_{j=1}^{n} f_{i,j}(x)$;
- $r(x)$ is a convex regularization and the following operator can be efficiently solved:

$$\textbf{prox}_{\eta,r}(x) = \operatorname*{argmin}_{z \in \mathbb{R}^d} \left(r(z) + \frac{1}{2\eta} \|z - x\|^2 \right),$$

- Communication: each agent can send $O(1)$ d-dimensional vectors to her neighbors.



Decentralized Communication

We adopt the gossip matrix based communication protocol. Let $W \in \mathbb{R}^{m\times m}$ be the gossip matrix and let $\mathbf{x}^{\mathbf{old}} = [x_1^{\text{old}}, \cdots, x_m^{\text{old}}]^\top$, and $\mathbf{x}^{\mathbf{new}} = [x_1^{\text{new}}, \cdots, x_m^{\text{new}}]^\top$,

- In parallel, for each agent i
  - agent i receives x_j^{old} from all neighbors $j \in \mathcal{N}_i$;
  - agent i updates her local variable by a weighted sum of them: $x_i^{\text{new}} = \sum_{j\in\mathcal{N}_i} w_{ij} x_j^{\text{old}}$;
- Mathematically, the communication can be abstracted as

$$\mathbf{x}^{\text{new}} = W\mathbf{x}^{\text{old}};$$

- Assumptions on W
  - $w_{ij} \neq 0$ if agent i and j can exchange information;
  - W is symmetric;
  - $\mathbf{0} \preceq W \preceq I, W\mathbf{1} = \mathbf{1}, \mathrm{null}(I - W) = \mathrm{span}(\mathbf{1})$;
- Mixing rate: $\|W\mathbf{x} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\| \leq \lambda_2(W)\|\mathbf{x} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\|$. Therefore, $\lambda_2(W) \in [0, 1)$ indicates how fast the variables will be averaged through decentralized communications;
- For any network, there exists such a W. We may design the network to achieve a balance between mixing rate and communication burden.

Problem Setting Continued

- Each $f_{i,j} : \mathbb{R}^d \to \mathbb{R}$ is L-smooth and convex:

$$f_{i,j}(y) - f_{i,j}(x) \leq \langle \nabla f_{i,j}(x), y - x \rangle + \frac{L}{2}\|y - x\|^2;$$

- Each $f_{i,j} : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex:

$$f_{i,j}(y) - f_{i,j}(x) \geq \langle \nabla f_{i,j}(x), y - x \rangle + \frac{\mu}{2}\|y - x\|^2.$$

- We denote the condition number $\kappa := \frac{L}{\mu}$ to measure the hardness of the problem;
- Learning objective: let x^* be the global minimizer:

$$\max\left\{\frac{1}{m}\sum_{i=1}^{m}||x_i^t - \bar{x}^t||^2, ||\bar{x}^t - x^*||^2\right\} < \epsilon;$$

- Metric:
  - Computational complexity: the number of evaluations of $\nabla f_{ij}(\cdot)$;
  - Communication complexity: the number of decentralized communications.

Table of Contents

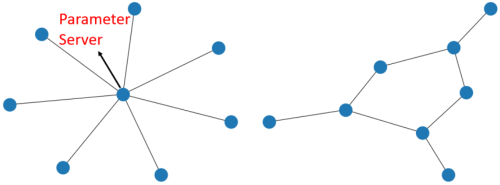
Distributed SGD

We assume that $r(x) = 0$ for simplicity and return to the composite case later.

- A centralized node (parameter sever) aggregates local gradients g_i and perform update:

$$x^{t+1} = x^t - \eta \frac{1}{m} \sum_{i=1}^{m} g_i;$$

- Distributed SGD is essentially the mini-batch SGD;
- The consensus error is zero after one communication: $\frac{1}{m}||\mathbf{x}^t - \mathbf{1}\bar{x}^t||^2 = 0$;
- The convergence error decreases similarly with the (mini-batch) single-agent SGD: $||\bar{x}^t - x^*||^2$.



Decentralized SGD

- Agents update with local gradient and average the variables by decentralized communication:
$$\mathbf{x}_i^{t+1} = (W\mathbf{x}^t)_i - \eta \nabla f_{i,j_i}(\mathbf{x}_i^t),$$
where $j_i \sim \mathrm{Unif}\{1, 2, \cdots, n\}$;
- Convergence rate with a constant learning rate:
$$\limsup_{t\to\infty} \frac{1}{n}\sum_{i=1}^{n} \mathbb{E}\left[\left\|\mathbf{x}_i^t - x^*\right\|_2^2\right] = \mathcal{O}\left(\frac{\eta\sigma^2}{m\mu} + \frac{\eta^2\kappa^2\sigma^2}{1-\lambda_2(W)} + \frac{\eta^2\kappa^2\sum_{i=1}^{m}||\nabla f_i(x^*)||^2}{m(1-\lambda_2(W))^2}\right),$$
where σ^2 is the upper bound of the variances of the local gradient noise;
- The third bias term is from the dissimilarity among the datasets across m agents;
- Moreover, x^* is not a fixed point of the update in expectation since $\nabla f_i(x^*) \neq 0$ in general.

Gradient Tracking (GT) SGD

- Challenge of DSGD: local agents have no access to the global gradient (of $f(x)$);
- Solution: Maintain an estimator s_i^t to approximate $\nabla f(\bar{x}^t)$ by communicating local gradients;
- Update rule:

$$\mathbf{x}_i^{t+1} = (W\mathbf{x}^t)_i - \mathbf{s}_i^t,$$
$$\mathbf{s}_i^{t+1} = (W\mathbf{s}^t)_i + \nabla f_{i,j_i}(\mathbf{x}_i^{t+1}) - \nabla f_{i,j_i}(\mathbf{x}_i^t).$$

- Dynamic tracking: $\mathbb{E}\bar{s}^t = \frac{1}{m}\sum_{i=1}^m \nabla f_i(\mathbf{x}_i^t)$;
- Tracking error: $||\nabla f(\bar{x}^t) - \mathbb{E}[\bar{s}^t]|| \leq \frac{L}{\sqrt{m}}||\mathbf{x}^t - \mathbf{1}\bar{\mathbf{x}}^t||$;
- With decentralized communications, we can show that

$$\forall i \in [m], \quad \mathbf{x}_i^t \to \bar{x}^t \qquad \text{and} \qquad \bar{\mathbf{s}}_i^t \to \bar{s}^t \to \nabla f(\bar{x}^t);$$

- With a well-connected network, the convergence behavior of GT-DSGD is determined only by the step-size sequence and the variance of the local stochastic gradient, which is similar to SGD.

GT Variance Reduction (VR)

The convergence error of SGD cannot shrink exponentially:

- $\nabla f_{i,j_i}(x)$ is an unbiased estimator of $\nabla f_i(x)$;
- The variance requires a decreasing sequence of learning rate;

Solution: each agent i maintains a variance-reduction estimator of $\nabla f_i(x)$;

- Let $\mathbf{w}_i^t$ be the most recent iterate at which $\nabla f_i(\cdot)$ is evaluated;
- Agent i replaces $\nabla f_{i,j_i}(\mathbf{x}_i^t)$ with SVRG-style gradient estimator:

$$\mathbf{v}_i^t = \nabla f_{i,j_i}(\mathbf{x}_i^t) - \nabla f_{i,j_i}(\mathbf{w}_i^t) + \nabla f_i(\mathbf{w}_i^t),$$

- Gradient tracking framework to mix local estimators:

$$\mathbf{x}_i^{t+1} = (W\mathbf{x}^t)_i - \mathbf{s}_i^t,$$
$$\mathbf{s}_i^{t+1} = (W\mathbf{s}^t)_i + \mathbf{v}_i^{t+1} - \mathbf{v}_i^t.$$

- Update $\mathbf{w}_i^{t+1} := \mathbf{x}_i^t$ with probability $1/n$;
- Convergence rate compared to SVRG:

$$\mathcal{O}\left((n + \frac{\kappa^2 \log \kappa}{(1-\lambda_2(W))^2}) \log \frac{1}{\epsilon}\right) \qquad \text{v.s.} \qquad \mathcal{O}\left((n+\kappa)\log\frac{1}{\epsilon}\right)$$

Multi-consensus GT-VR

- Challenge: the mixing rate may not match the convergence rate;
- Observation: mixing rate can be improved by involving K communication rounds:

$$\|W^K\mathbf{x} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\| \le \lambda_2(W)^K\|\mathbf{x} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\|$$

  - $K = \infty$: return to the distributed setting with $\|W^\infty\mathbf{x} - \frac{1}{m}\mathbf{1}\mathbf{1}^\top\mathbf{x}\| = 0$;
  - An appropriate K to "improve" the mixing rate and to match the convergence rate;
- Multi-consensus + Gradient Tracking + Variance Reduction (PMGT-VR):

Algorithm 1 PMGT-VR Framework

```
1: Input: x_i^0 = x_j^0 for 1 ≤ i, j, ≤ m, v^{-1} = s^{-1} = ∇F(x^0), η, and K
2: for t = 0, ..., T do
3:     Update the local stochastic gradient estimators v^t;
4:     Update the local gradient trackers as s^t = W^K (s^{t-1} + v^t - v^{t-1}).
5:     Update: x^{t+1} = W^K (x^t - η s^t);
6: end for
7: Output: x^{T+1}.
```

Fast Mixing

- One advantage of multi-consensus is that the K communications can be naturally accelerated;
- By using Chebyshev acceleration or FastMix subroutine, the communication rounds for one iteration is improved:

$$(\log \kappa + \log n) \cdot \frac{1}{(1-\lambda_2(W))} \rightarrow (\log \kappa + \log n) \cdot \frac{1}{\sqrt{1-\lambda_2(W)}};$$

- Trade-off between a fast mixing rate $\lambda_2(W) \approx 1 - \frac{1}{\log_2(m)}$ and the communication burden $\log m$ (the maximum degree of the node).

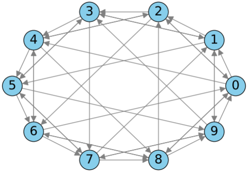


Figure: An exponential graph. [3]

Main Result

Theorem

Let $K = \frac{1}{\sqrt{1-\lambda_2(W)}} \log \frac{1}{\rho}$ *where* ρ *satisfies* $\rho \leq \frac{1}{41} \min\left(\frac{1}{24\kappa}, \frac{1}{4n}\right)$, *and let step-size* $\eta = 1/(12L)$. *Then, it holds that*

$$\mathbb{E}\left[||\bar{x}^t - x^*||\right] \leq \max\left(1 - \frac{1}{24\kappa}, 1 - \frac{1}{4n}\right)^t \left(V^0 + \left\|\mathbf{z}^0\right\|\right)$$

$$\mathbb{E}\left[\frac{1}{m}\left\|\mathbf{x}^t - \mathbf{1}\bar{x}^t\right\|^2\right] \leq \max\left(1 - \frac{1}{24\kappa}, 1 - \frac{1}{4n}\right)^t \cdot \left(V^0 + \left\|\mathbf{z}^0\right\|\right).$$

Methods	Problem	Complexity of computation	Complexity of communication
GT-SVRG [3]	f	$\mathcal{O}\left(\left(n + \frac{\kappa^2 \log \kappa}{(1-\lambda_2(W))^2}\right) \log \frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\left(n + \frac{\kappa^2 \log \kappa}{(1-\lambda_2(W))^2}\right) \log \frac{1}{\epsilon}\right)$
NIDS [2, 4]	$f + r$	$\mathcal{O}\left(n\left(\kappa + \frac{1}{(1-\lambda_2(W))}\right) \log \frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\left(\kappa + \frac{1}{(1-\lambda_2(W))}\right) \log \frac{1}{\epsilon}\right)$
Our methods	$f + r$	$\mathcal{O}\left((n + \kappa) \log \frac{1}{\epsilon}\right)$	$\mathcal{O}\left(\frac{(n \log n + \kappa \log \kappa)}{\sqrt{1-\lambda_2(W)}} \log \frac{1}{\epsilon}\right)$

Table: Complexity comparisons between PMGT-VR algorithms and existing works for strongly convex problem.

Proof Sketch: Relate Error Terms

We consider the following error terms.

- Consensus error: $\mathbf{z}^t = [\frac{1}{m}||\mathbf{x}^t - \mathbf{1}\bar{x}^t||^2, \frac{\eta^2}{m}||\mathbf{s}^t - \mathbf{1}\bar{s}^t||^2]^\top$;
- Gradient learning error: $\Delta^t = \frac{1}{mn}\sum_{i,j=1}^{m,n} \|\nabla f_{i,j}(\mathbf{w}_i^t) - \nabla f_{i,j}(x^*)\|^2$;
- Convergence error: $||\bar{x}^t - x^*||$;

We have the following derivation:

- Gradient tracking: $||\nabla f(\bar{x}^t) - \mathbb{E}[\bar{s}^t]|| \leq \frac{L}{\sqrt{m}}||\mathbf{x}^t - \mathbf{1}\bar{x}^t||$;
- Decentralized communication: $||\mathbf{x}^K - \mathbf{1}\bar{x}|| \leq \rho||\mathbf{x}^0 - \mathbf{1}\bar{x}||, \bar{x} = \frac{1}{m}\mathbf{1}^\top\mathbf{x}^K$ with $\rho = (1 - \sqrt{1 - \lambda_2(W)})^K$;
- Update rule.

These component together lead to a inequality system:

$$\mathbb{E}\left[\mathbf{z}^{t+1}\right] \leq 2\rho^2 \cdot \Bigg(\begin{bmatrix} 4, & 4 \\ 8(8\rho^2+1)L^2\eta^2, & 64\rho^2\eta^2L^2+1 \end{bmatrix} \cdot \mathbf{z}^t$$
$$+ \eta^2 \begin{bmatrix} 0 \\ 8L^2(||\bar{x}^{t+1} - x^*||^2 + ||\bar{x}^t - x^*||^2) + 4(\Delta^{t+1} + \Delta^t) \end{bmatrix} \Bigg).$$

Approximate the Centralized Algorithm

- We can directly set a sufficiently large K to get a small enough ρ;

$$\mathbb{E}\left[\mathbf{z}^{t+1}\right] \leq 2\rho^2 \cdot \Bigg(\begin{bmatrix} 4, & 4 \\ 8(8\rho^2+1)L^2\eta^2, & 64\rho^2\eta^2L^2+1 \end{bmatrix} \cdot \mathbf{z}^t + \eta^2 \begin{bmatrix} 0 \\ 8L^2(||\bar{x}^{t+1} - x^*||^2 + ||\bar{x}^t - x^*||^2) + 4(\Delta^{t+1} + \Delta^t) \end{bmatrix} \Bigg).$$

- Then, $\bar{x}^t$ behaves as a centralized one and can be analyzed by standard framework for SGD-type algorithm [1];
- On the contrary, the previous work carefully designed the system so that there exists a feasible solution of hyper-parameters, which may be sub-optimal (e.g. $\eta = \mathcal{O}\left(\frac{\mu(1-\lambda_2^2(W))}{L^2}\right)$ for GT-SVRG).

Experiments: Comparison with Existing Methods

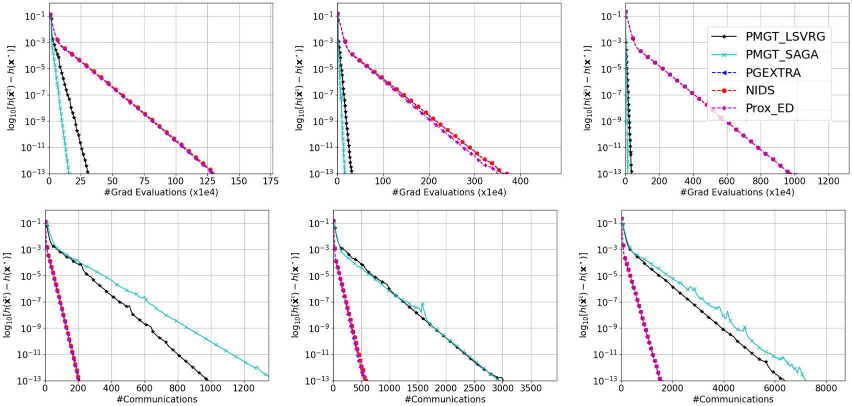


Figure: Performance comparison with $n = 6400$ and $\sigma_i = n \times 10^{-7}$ for all agents. From the left to the right, the network becomes less-connected (slow mixing rate).

Experiments with Different K

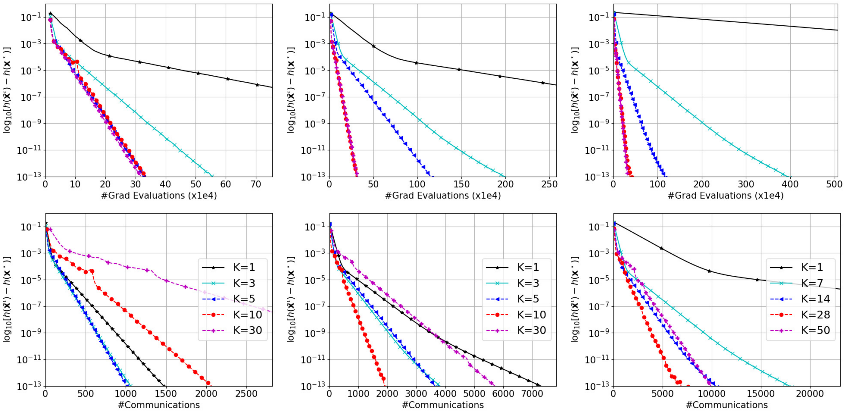


Figure: Performance comparison for PMGT-LSVRG under different consensus steps K with $n = 6400$ and $\sigma_i = n \times 10^{-7}$. From the left to the right, the network becomes less-connected (slow mixing rate).

Thank you for Listening!

Paper : Haishan Ye*, Wei Xiong*, and Tong Zhang, "PMGT-VR: A decentralized proximal-gradient algorithmic framework with variance reduction".

[1] Eduard A. Gorbunov, Filip Hanzely, and Peter Richtárik. A unified theory of SGD: variance reduction, sampling, quantization and coordinate descent. In *The 23rd International Conference on Artificial Intelligence and Statistics, AISTATS 2020, 26-28 August 2020, Online [Palermo, Sicily, Italy]*, volume 108 of *Proceedings of Machine Learning Research*, pages 680–690. PMLR, 2020.

[2] Zhi Li, Wei Shi, and Ming Yan. A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates. *IEEE Trans. Signal Process.*, 67(17):4494–4506, 2019.

[3] Ran Xin, Usman A Khan, and Soummya Kar. Variance-reduced decentralized stochastic optimization with accelerated convergence. *IEEE Transactions on Signal Processing*, 68:6255–6271, 2020.

[4] Jinming Xu, Ye Tian, Ying Sun, and Gesualdo Scutari. Distributed algorithms for composite optimization: Unified and tight convergence analysis. *CoRR*, abs/2002.11534, 2020.