Accelerating Gossip SGD with Periodic Global Averaging

Yiming Chen [*1] **Kun Yuan** [*1] **Yingya Zhang** [1] **Pan Pan** [1] **Yinghui Xu** [1] **Wotao Yin** [1]

Abstract

Communication overhead hinders the scalability of large-scale distributed training. Gossip SGD, where each node averages only with its neighbors, is more communication-efficient than the prevalent parallel SGD. However, its convergence rate is reversely proportional to quantity $1-\beta$ which measures the network connectivity. On large and sparse networks where $1-\beta \to 0$, Gossip SGD requires more iterations to converge, which offsets against its communication benefit. This paper introduces Gossip-PGA, which adds Periodic Global Averaging into Gossip SGD. Its transient stage, i.e., the iterations required to reach asymptotic linear speedup stage, improves from $\Omega(\beta^4 n^3/(1-\beta)^4)$ to $\Omega(\beta^4 n^3 H^4)$ for non-convex problems. The influence of network topology in Gossip-PGA can be controlled by the averaging period H. Its transient-stage complexity is also superior to Local SGD which has order $\Omega(n^3 H^4)$. Empirical results of large-scale training on image classification (ResNet50) and language modeling (BERT) validate our theoretical findings.

1. Introduction

The scale of deep learning nowadays calls for efficient large-scale distributed training across multiple computing nodes in the data-center clusters. In distributed optimization, a network of n nodes cooperate to solve the problem

$$\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{i=1}^{n}[f_i(x) := \mathbb{E}_{\boldsymbol{\xi}_i\sim D_i} F_i(x;\boldsymbol{\xi}_i)] \qquad (1)$$

where each component f_i is local and private to node i and the random variable $\boldsymbol{\xi}_i$ denotes the local data that follows distribution D_i. We assume each node i can locally evaluate stochastic gradients $\nabla F_i(x;\boldsymbol{\xi}_i)$ where $\boldsymbol{\xi}_i \sim D_i$, but must communicate to access information from other nodes.

[*]Equal contribution [1]Alibaba Group, Hangzhou, China. Correspondence to: Kun Yuan <kun.yuan@alibaba-inc.com>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021.

Method	Epoch	Acc.%	Time(hrs.)
Parallel SGD	120	76.26	2.22
Gossip SGD (ring)	120	74.86	1.56
Gossip SGD (expo)	120	75.34	1.55
Gossip SGD (ring)	240	75.62	3.02
Gossip SGD (expo)	240	76.18	3.03

Table 1. Top-1 validation accuracy for ImageNet with 256 GPUs connected with the ring or one-peer exponential network. Gossip SGD takes more time to reach the same accuracy as Parallel SGD.

Parallel SGD methods are leading algorithms to solve (1), in which every node processes local training samples independently, and synchronize gradients every iteration either using a central *Parameter Server (PS)* (Li et al., 2014) or the *All-Reduce* communication primitive (Patarasuk & Yuan, 2009). The global synchronization in Parallel SGD either incurs significant bandwidth cost or high latency, which hampers the training scalability.

Many alternative methods have been proposed to reduce communication overhead in distributed training. Gossip SGD, also known as decentralized SGD (Nedic & Ozdaglar, 2009; Chen & Sayed, 2012; Lian et al., 2017; 2018; Assran et al., 2019), recently received lots of attention. This line of work lets each node communicate with (some of) their direct neighbors. In a sparse topology such as one-peer exponential graph (Assran et al., 2019), each node only communicates with *one* neighbor each time. This gossip-style communication is much faster than *PS* and *All-Reduce* but the computed average can be highly inaccurate. Local SGD (Stich, 2019; Yu et al., 2019; Lin et al., 2018) is another line of work that increases the computation-to-communication ratio. Local SGD lets each node to run local gradient descent for multiple rounds and only average their parameters globally once in a while. By communicating less frequently, Local SGD reduces the communication overhead.

The reduced communication in Gossip and Local SGDs comes at a cost: slower convergence rate. While both algorithms are proved to have convergence linear speedup asymptotically, they are sensitive to network topology and synchronization period, respectively. For Gossip SGD, the convergence rate is inversely proportional to $1-\beta$ (β is defined in Remark 1). Since $\beta \to 1$ on the large and sparse network topology which is most valuable for deep training, Gossip SGD will converge very slow and require more iterations than Parallel SGD to achieve a desired solution. This

	Gossip SGD		Gossip-PGA	
	IID	NON-IID	IID	NON-IID (PROPOSED)
SMALL OR DENSE NETWORK (WHEN $\frac{1}{1-\beta} < H$)	$\Omega(\frac{n^3\beta^4}{\boldsymbol{(1-\beta)^2}})$	$\Omega(\frac{n^3\beta^4}{\boldsymbol{(1-\beta)^4}})$	$\Omega(n^3\beta^4\boldsymbol{C_\beta^2})$	$\Omega(\frac{n^3\beta^4\boldsymbol{C_\beta^2}}{\boldsymbol{(1-\beta)^2}})$
LARGE OR SPARSE NETWORK (WHEN $\frac{1}{1-\beta} \geq H$)	$\Omega(\frac{n^3\beta^4}{\boldsymbol{(1-\beta)^2}})$	$\Omega(\frac{n^3\beta^4}{\boldsymbol{(1-\beta)^4}})$	$\Omega(n^3\beta^4\boldsymbol{C_\beta^2})$	$\Omega(n^3\beta^4\boldsymbol{C_\beta^2 H^2})$

Table 2. The lengths of the transient stages of Gossip SGD and Gossip-PGA. Since $C_\beta = \sum_{k=0}^{H-1}\beta^k = (1-\beta^H)/(1-\beta) < \min\{1/(1-\beta), H\}$, Gossip-PGA always has shorter transient stage, more evident on large and sparse networks where $\beta \to 1$.

	Local SGD	Gossip-PGA
IID SCENARIO	$\Omega(n^3\boldsymbol{H^2})$	$\Omega(n^3\boldsymbol{\beta^4 C_\beta^2})$
NON-IID SCENARIO	$\Omega(n^3\boldsymbol{H^4})$	$\Omega(n^3\boldsymbol{\beta^4 C_\beta^2 H^2})$

Table 3. The lengths of the transient stages of Local SGD and Gossip-PGA. Gossip-PGA always has shorter transient stages than Local SGD since $\beta < 1$ and $C_\beta < H$. Such superiority becomes more significant on well-connected networks where $\beta \to 0$.

may nullify its communication efficiency and result in even more training time (see Table 1). Local SGD with a large averaging period meets the same issue.

This paper proposes Gossip-PGA, which adds periodic All-Reduce global averaging into Gossip to accelerate its convergence especially on large and sparse networks. Gossip-PGA also extends Local SGD with fast gossip-style communication after local updates. When the same averaging period H is used, the additional gossip communication in Gossip-PGA endows it with faster convergence than Local SGD.

Challenges. Gossip-PGA can be regarded as a special form of the topology-changing Gossip SGD (Koloskova et al., 2020) and SlowMo (Wang et al., 2019) (in which the base optimizer is set as Gossip SGD, and the momentum coefficient $\beta = 0$). However, its theory and practical performance were not carefully investigated in literature. Unanswered important questions include how much acceleration can PGA bring to Gossip and Local SGDs, in what scenario can PGA benefits most, how to adjust the averaging period effectively, and how Gossip-PGA performs in large-scale deep learning systems. Providing quantitative answers to these questions requires new understanding on the interplay between gossip communication and global averaging period. Simply following existing analysis in (Koloskova et al., 2020) will result in incomplete conclusions, see Remark 5. Also, the analysis in SlowMo (Wang et al., 2019) does not consider heterogeneous data distributions and cannot cover our results.

1.1. Main Results

This paper proves that Gossip-PGA converges at

$$O\Big(\underbrace{\frac{\sigma}{\sqrt{nT}}}_{\text{SGD rate}} + \underbrace{\frac{C_\beta^{\frac{1}{3}}\beta^{\frac{2}{3}}(\sigma^{\frac{2}{3}} + D_\beta^{\frac{1}{3}}b^{\frac{2}{3}})}{T^{\frac{2}{3}}} + \frac{\beta D_\beta}{T}}_{\text{Extra overhead}}\Big) \quad (2)$$

for both smooth *convex* and *non-convex* functions f_i (the metrics used for both scenarios can be referred to Theorems 1 and 2), where n is the network size, T is the total number of iterations, σ^2 denotes gradient noise, b^2 gauges data heterogeneity, $\beta \in (0,1)$ measures how well the network is connected, H is the global averaging period, and we define $C_\beta = \sum_{k=0}^{H-1}\beta^k$ and $D_\beta = \min\{H, 1/(1-\beta)\}$.

Linear speedup. When T is sufficiently large, the first term $1/\sqrt{nT}$ dominates (2). This also applies to Parallel, Local, and Gossip SGDs. Gossip-PGA and these algorithms all require $T = \Omega(1/(n\epsilon^2))$ iterations to reach a desired accuracy ϵ, which is inversely proportional to n. We say an algorithm is in its linear-speedup stage at Tth iteration if, for this T, the term involving nT is dominating the rate.

Transient stage. Transient stage is referred to those iterations before an algorithm reaches its linear-speedup stage, that is iterations $1, \ldots, T$ where T is relatively small so non-nT terms (i.e., the extra overhead terms in (2)) still dominate the rate. We take Gossip-PGA in the non-iid scenario ($b^{2/3} \geq \sigma$) as example. To reach linear speedup, T has to satisfy $T^{\frac{2}{3}}/(C_\beta^{\frac{1}{3}}\beta^{\frac{2}{3}}D_\beta^{1/3}) \geq n^{\frac{1}{2}}T^{\frac{1}{2}}$, i.e., $T \geq n^3\beta^4C_\beta^2D_\beta^2$. So, the transient stage has $\Omega(n^3\beta^4C_\beta^2D_\beta^2)$ iterations. **Transient stage is an important metric to measure the scalability of distributed algorithms.**

Shorter transient stage than Gossip SGD. The transient stage comparison between Gossip SGD and Gossip-PGA is shown in Table 2. Since $C_\beta = (1-\beta^H)/(1-\beta) < \min\{H, 1/(1-\beta)\}$, we conclude Gossip-PGA *always* has a shorter transient stage than Gossip SGD for any β and H. Moreover, the superiority of Gossip-PGA becomes evident when the network is large and sparse, i.e., $1-\beta \to 0$. In this case, the transient stage of Gossip SGD can grow dramatically (see the second line in Table 2) while Gossip-PGA is controlled by the global period H because $C_\beta < H$. As a result, Gossip-PGA improves the transient stage of Gossip-SGD from $O(n^3/(1-\beta)^4)$ (or $O(n^3/(1-\beta)^2$ in the iid scenario) to $O(n^3)$ when $\beta \to 1$.

Shorter transient stage than Local SGD. The transient stage comparison between Local SGD and Gossip-PGA is shown in Table 3. Using $C_\beta < H$, we find Gossip-PGA is *always* endowed with a shorter transient stage than Local SGD. Moreover, when the network is well-connected such that $\beta \to 0$, it holds that $C_\beta \to 1$. Gossip-PGA will have a significantly shorter transient stage than Local SGD.

1.2. Contributions

- We establish the convergence rate of Gossip-PGA for both smooth convex and non-convex problems. Our results clarify how gossip communication and periodic global averaging collaborate to improve the transient stage of Gossip and Local SGDs. We also established shorter wall-clock training *times* of Gossip-PGA.
- We propose Gossip-AGA, which has adaptive global averaging periods. Gossip-AGA automatically adjusts H and has convergence guarantees.
- We conduct various experiments (convex logistic regression and large-scale deep learning tasks) to validate all established theoretical results. In particular, the proposed Gossip-PGA/AGA achieves a similar convergence speed to parallel SGD in *iterations*, but provides $1.3 \sim 1.9\times$ *runtime* speed-up. The introduced global averaging steps in Gossip-PGA/AGA remedy the accuracy degradation in Gossip SGD and Local SGD.

2. Related Work

Decentralized optimization algorithms can be tracked back to (Tsitsiklis et al., 1986). After that, decentralized optimization has been intensively studied in signal processing and control community. Decentralized gradient descent (DGD) (Nedic & Ozdaglar, 2009), diffusion (Chen & Sayed, 2012) and dual averaging (Duchi et al., 2011) are among the first decentralized algorithms that target on general optimization problems. However, these algorithms suffer from a bias caused by data heterogeneity (Yuan et al., 2016). Various primal-dual algorithms are proposed to overcome this issue, and they are based on alternating direction method of multipliers (ADMM) (Shi et al., 2014), explicit bias-correction (Shi et al., 2015; Yuan et al., 2019; Li et al., 2019c), gradient tracking (Xu et al., 2015; Di Lorenzo & Scutari, 2016; Nedic et al., 2017; Qu & Li, 2018), coordinate-descent methods (He et al., 2018), and dual acceleration (Scaman et al., 2017; 2018; Uribe et al., 2020).

In the context of machine learning, decentralized SGD, also known as Gossip SGD, have gained a lot of attention recently. (Lian et al., 2017) first proves Gossip SGD can reach the same linear speedup as vanilla parallel SGD. After that, (Assran et al., 2019) comes out to extend Gossip SGD to directed topology. A recent work (Koloskova et al., 2020) proposes a unified framework to analyze algorithms with changing topology and local updates. While it covers Gossip-PGA as a special form, the theoretical and practical benefits of periodic global averaging were not studied therein. The data heterogeneity issue suffered in Gossip SGD is discussed and addressed in (Tang et al., 2018; Yuan et al., 2020; Lu et al., 2019; Xin et al., 2020). Gossip SGD is also extended to asynchronous scenarios in (Lian et al., 2018; Luo et al., 2020).

Local SGD can be traced back to (Zinkevich et al., 2010) which proposed a one-shot averaging. More frequent averaging strategy is proposed in (Zhang et al., 2016), and the convergence property of Local SGD is established in (Yu et al., 2019; Stich, 2019; Bayoumi et al., 2020). Local SGD is also widely-used in federated learning (McMahan et al., 2017; Li et al., 2019a).

Another closely related work (Wang et al., 2019) proposes a slow momentum (SlowMo) framework, where each node, similar to the Gossip-PGA algorithm proposed in this paper, periodically synchronizes across the network and performs a momentum update. The analysis in SlowMo cannot cover the convergence results in this paper due to its data-homogeneous setting. In addition, we will clarify some new questions such as how much acceleration can PGA bring to Gossip and Local SGDs, and how to adjust the averaging period effectively.

Various techniques can be integrated to Gossip SGD to improve its communication efficiency. This paper does not consider quantization (Alistarh et al., 2017; Bernstein et al., 2018), gradient compression (Tang et al., 2019; Koloskova et al., 2019b;a) and lazy communication (Chen et al., 2018; Liu et al., 2019), but these orthogonal techniques can be added to our methods.

3. Gossip SGD with Periodic Global Average

Assume all computing nodes are connected over a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V} = \{1, 2, \cdots, n\}$ denote the node index and $\mathcal{E}$ denote the communication links between all nodes. Similar to existing decentralized algorithms (Nedic & Ozdaglar, 2009; Chen & Sayed, 2012; Lian et al., 2017; Assran et al., 2019), information exchange in the gossip step is only allowed to occur between connected neighbors. To characterize the decentralized communication, we let $W \in \mathbb{R}^{n\times n}$ be a doubly stochastic matrix, i.e., $W \geq 0$, $W\mathbb{1}_n = \mathbb{1}_n$ and $\mathbb{1}_n^T W = \mathbb{1}_n^T$. The (i,j)-th element w_{ij} is the weight to scale information flowing from node j to node i. If nodes i and j are not neighbors then $w_{ij} = 0$, and if they are neighbors or identical then the weight $w_{ij} > 0$. Furthermore, we define $\mathcal{N}_i$ as the set of neighbors of node i which also includes node i itself.

The Gossip-PGA algorithm is listed in Algorithm 1. In the gossip step, every node i collects information from all its *connected* neighbors. For global average step, nodes synchronize their model parameters using the efficient All-Reduce primitives. When $H \to \infty$, Gossip-PGA will reduce to standard Gossip SGD; when $W = \frac{1}{n}\mathbb{1}\mathbb{1}_n$, Gossip-

Algorithm 1 Gossip-PGA

Require: Initialize learning rate $\gamma > 0$, weight matrix W, global averaging period H, and let each $\boldsymbol{x}_i^{(0)}$ to be equivalent to each other.

for $k = 0, 1, 2, ..., T-1$, *every node* i **do**
  Sample $\boldsymbol{\xi}_i^{(k+1)}$, update $\boldsymbol{g}_i^{(k)} = \nabla F_i(\boldsymbol{x}_i^{(k)}; \boldsymbol{\xi}_i^{(k+1)})$
  $\boldsymbol{x}_i^{(k+\frac{1}{2})} = \boldsymbol{x}_i^{(k)} - \gamma \boldsymbol{g}_i^{(k)}$ ▷ Local SGD update
  if $\mathrm{mod}(k+1, H) = 0$ **then**
    $\boldsymbol{x}_i^{(k+1)} = \frac{1}{n}\sum_{j=1}^n \boldsymbol{x}_j^{(k+\frac{1}{2})}$ ▷ global average
  else
    $\boldsymbol{x}_i^{(k+1)} = \sum_{j \in \mathcal{N}_i} w_{ij} \boldsymbol{x}_j^{(k+\frac{1}{2})}$ ▷ one gossip step

PGA will reduce to vanilla parallel SGD; when $W = I$, Gossip-PGA will reduce to Local SGD.

All-Reduce v.s. multiple Gossips. In a computing cluster with n nodes, global averaging is typically conducted in an efficient Ring All-Reduce manner, rather than via multiple gossip steps as in (Berahas et al., 2018). The communication time comparison between a single gossip and Ring All-Reduce step is listed in Appendix H. In the one-peer exponential network, the exact global average can be achieved via $\ln(n)$ gossip communications, which generally takes more wall-clock time than a single Ring All-Reduce operation. Therefore, we recommend exploiting All-Reduce to conduct global averaging in Gossip-PGA.

Data-center v.s. wireless network. This paper considers deep training within high-performance data-center clusters, in which all GPUs are connected with high-bandwidth channels and the network topology can be fully controlled. Under such setting, the periodic global averaging conducted with Ring All-Reduce has tolerable communication cost, see Appendix H. For scenarios where global averaging is extremely expensive to conduct such as in wireless sensor network, the global averaging can be approximated via multiple gossip steps, or may not be recommended.

3.1. Assumptions and analysis highlights

We now establish convergence rates for Gossip-PGA on smooth convex and non-convex problems. For all our theoretical results we make the following standard assumptions.

Assumption 1 (L-SMOOTHNESS). *Each local cost function $f_i(x)$ is differentiable, and there exists a constant L such that for each $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^d$:*

$$\|\nabla f_i(\boldsymbol{x}) - \nabla f_i(\boldsymbol{y})\| \le L\|\boldsymbol{x} - \boldsymbol{y}\|. \tag{3}$$

Assumption 2 (GRADIENT NOISE). *Recall $\boldsymbol{g}_i^{(k)}$ is the stochastic gradient noise defined in line 2 of Algorithm 1. It is assumed that for any k and i that*

$$\mathbb{E}[\boldsymbol{g}_i^{(k)} - \nabla f_i(\boldsymbol{x}_i^{(k)}) | \mathcal{F}^{(k-1)}] = 0, \tag{4}$$

$$\mathbb{E}[\|\boldsymbol{g}_i^{(k)} - \nabla f_i(\boldsymbol{x}_i^{(k)})\|^2 | \mathcal{F}^{(k-1)}] \le \sigma^2 \tag{5}$$

for some constant $\sigma^2 > 0$. Moreover, we assume $\boldsymbol{\xi}_i^{(k)}$ is independent of each other for any k and i. Filtration is defined as $\mathcal{F}^{(k)} = \{\{\boldsymbol{x}_i^{(k)}\}_{i=1}^n, \{\boldsymbol{\xi}_i^{(k)}\}_{i=1}^n, \cdots, \{\boldsymbol{x}_i^{(0)}\}_{i=1}^n, \{\boldsymbol{\xi}_i^{(0)}\}_{i=1}^n\}$

Assumption 3 (WEIGHTING MATRIX). *The network is strongly connected and the weight matrix W satisfies $W\mathbb{1}_n = \mathbb{1}_n$, $\mathbb{1}_n^T W = \mathbb{1}_n^T$, $\mathrm{null}(I - W) = \mathrm{span}(\mathbb{1}_n)$. We also assume $\|W - \frac{1}{n}\mathbb{1}\mathbb{1}^T\|_2 \le \beta$ for some $\beta \in (0, 1)$.*

Remark 1. *Quantity $\beta \in (0, 1)$ indicates how well the topology is connected. Smaller β indicates better-connected network while larger β implies worse-connected topology.*

Analysis highlights. To derive the influence of periodic global averaging, we have to exploit all useful algorithm structures to achieve its superiority. These structures are:

- $\boldsymbol{x}_i^{(k)} = \bar{\boldsymbol{x}}^{(k)}$ when $\mod(k, H) = 0$. This structure relieves the influence of network topology;
- Gossip communications within each period also contribute to consensus among nodes. This structure is crucial to establish superiority to Local SGD;
- When network is large and sparse, i.e., $H < \frac{1}{1-\beta}$, the global averaging is more critical to drive consensus. This structure is crucial to establish superiority to Gossip SGD when $H < \frac{1}{1-\beta}$.
- When network is small or dense, i.e., $H > \frac{1}{1-\beta}$, gossip communication is more critical to drive consensus. This structure is crucial to establish superiority to Gossip SGD when $H > \frac{1}{1-\beta}$.

Ignoring any of the above structures in the analysis will result in incomplete conclusions on comparison among Gossip-PGA, Gossip SGD and Local SGD.

3.2. Convergence analysis: convex scenario

Assumption 4 (CONVEXITY). *Each $f_i(x)$ is convex.*

Definition 1 (DATA HETEROGENEITY). *When each $f_i(x)$ is convex, we let $b^2 = \frac{1}{n}\sum_{i=1}^n \|\nabla f_i(x^\star)\|^2$ denote the data heterogeneity.*

When each local data follows the same distribution, it holds that $f_i(x) = f(x)\ \forall i$ and hence $\nabla f_i(x^\star) = \nabla f(x^\star) = 0$ which also implies $b^2 = 0$. With Assumption 4, we let $x^\star$ be one of the global solutions to problem (1).

	Gossip SGD (Koloskova et al., 2020)	Gossip-PGA
Rates (general form)	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\beta^{\frac{2}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}\boldsymbol{(1-\beta)^{\frac{1}{3}}}}+\frac{\beta^{\frac{2}{3}}b^{\frac{2}{3}}}{T^{\frac{2}{3}}\boldsymbol{(1-\beta)^{\frac{2}{3}}}}+\frac{\beta}{\boldsymbol{(1-\beta)}T}\Big)$	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\boldsymbol{C_\beta^{\frac{1}{3}}}\beta^{\frac{2}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\boldsymbol{C_\beta^{\frac{1}{3}}D_\beta^{\frac{1}{3}}}\beta^{\frac{2}{3}}b^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\beta\boldsymbol{D_\beta}}{T}\Big)$
Rates (when $\frac{1}{1-\beta}<H$)	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\beta^{\frac{2}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}\boldsymbol{(1-\beta)^{\frac{1}{3}}}}+\frac{\beta^{\frac{2}{3}}b^{\frac{2}{3}}}{T^{\frac{2}{3}}\boldsymbol{(1-\beta)^{\frac{2}{3}}}}+\frac{\beta}{(1-\beta)T}\Big)$	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\boldsymbol{C_\beta^{\frac{1}{3}}}\beta^{\frac{2}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\boldsymbol{C_\beta^{\frac{1}{3}}}\beta^{\frac{2}{3}}b^{\frac{2}{3}}}{\boldsymbol{(1-\beta)^{\frac{1}{3}}}T^{\frac{2}{3}}}+\frac{\beta}{(1-\beta)T}\Big)$
Rates (when $\frac{1}{1-\beta}\geq H$)	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\beta^{\frac{2}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}\boldsymbol{(1-\beta)^{\frac{1}{3}}}}+\frac{\beta^{\frac{2}{3}}b^{\frac{2}{3}}}{T^{\frac{2}{3}}\boldsymbol{(1-\beta)^{\frac{2}{3}}}}+\frac{\beta}{(1-\beta)T}\Big)$	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\boldsymbol{C_\beta^{\frac{1}{3}}}\beta^{\frac{2}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\boldsymbol{C_\beta^{\frac{1}{3}}H^{\frac{1}{3}}}\beta^{\frac{2}{3}}b^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\beta\boldsymbol{H}}{T}\Big)$

Table 4. Convergence rate comparison between Gossip SGD and Gossip-PGA for smooth convex/non-convex problems. We use notation b^2 to indicate the data heterogeneity for both convex and non-convex scenarios.

Theorem 1. *Under Assumptions 1–4, if γ is chosen as*

$$\gamma=\min\Big\{\frac{1}{12\beta LD_\beta},\Big(\frac{r_0}{r_1(T+1)}\Big)^{\frac{1}{2}},\Big(\frac{r_0}{r_2(T+1)}\Big)^{\frac{1}{3}}\Big\} \quad (6)$$

with constants $r_0=2\mathbb{E}\|\bar{\boldsymbol{x}}^{(0)}-x^\star\|^2, r_1=2\sigma^2/n$, and $r_2=6L\beta^2C_\beta\sigma^2+18L\beta^2C_\beta D_\beta$, it holds for any T that

$$\mathbb{E}f(\hat{\boldsymbol{x}}^{(T)})-f(x^\star)$$
$$=O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{C_\beta^{\frac{1}{3}}\beta^{\frac{2}{3}}(\sigma^{\frac{2}{3}}+D_\beta^{\frac{1}{3}}b^{\frac{2}{3}})}{T^{\frac{2}{3}}}+\frac{\beta D_\beta}{T}\Big) \quad (7)$$

where $\bar{\boldsymbol{x}}^{(k)}=\frac{1}{n}\sum_{i=1}^n\boldsymbol{x}_i^{(k)}$, $\hat{\boldsymbol{x}}^{(T)}=\frac{1}{T+1}\sum_{k=0}^T\bar{\boldsymbol{x}}^{(k)}$, $C_\beta=\sum_{k=0}^{H-1}\beta^k$ and $D_\beta=\min\{H,1/(1-\beta)\}$. (Proof is in Appendix B.)

Remark 2. *When $\beta\to 0$, i.e., the network tends to be fully connected, Gossip-PGA will converge at rate $O(\sigma/\sqrt{nT})$, which recovers the rate of parallel SGD.*

Remark 3. *When $\beta\to 1$, i.e., the information exchange via gossip communication is inefficient, it holds that $C_\beta\to H$ and $D_\beta=\min\{H,1/(1-\beta)\}=H$. Substituting them to (7) will recover the rate of Local SGD, see Table 6.*

Remark 4. *When $H\to\infty$, i.e., the networked agents tend not to conduct global synchronization, it holds that $C_\beta\to 1/(1-\beta)$ and $D_\beta=\frac{1}{1-\beta}$. Substituting these values to (7) will recover the rate of Gossip SGD, see Table 4.*

3.3. Convergence analysis: non-convex scenario

We first introduce an assumption about data heterogeneity specifically for non-convex problems:

Assumption 5 (Data heterogeneity)**.** *There exists constant $\hat{b}>0$ such that $\frac{1}{n}\sum_{i=1}^n\|\nabla f_i(\boldsymbol{x})-\nabla f(\boldsymbol{x})\|^2\leq\hat{b}^2$ for any $x\in\mathbb{R}^d$. If local data follows the same distribution, it holds that $\hat{b}=0$.*

Theorem 2. *Under Assumptions 1–3 and 5, if γ satisfies the condition (6) (replace b^2 with $\hat{b}^2$ and use $r_0=4\mathbb{E}f(\bar{\boldsymbol{x}}^{(0)})$), it holds for any $T>0$ that*

$$\frac{1}{T+1}\sum_{k=0}^T\mathbb{E}\|\nabla f(\bar{\boldsymbol{x}}^{(k)})\|^2$$

	Gossip SGD	Gossip-PGA
Transient iter.	$O(n^7)$	$O(n^5)$
Single comm.	$O(\theta d+\alpha)$	$O(\theta d+\sqrt{n}\alpha)$
Transient time	$O(n^7\theta d+n^7\alpha)$	$O(n^5\theta d+n^{5.5}\alpha)$

Table 5. Transient time comparison between non-iid Gossip SGD and Gossip-PGA over the specific grid ($1-\beta=O(1/n)$) topology. We choose $H=\sqrt{n}$ as the period in Gossip-PGA.

$$=O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{C_\beta^{\frac{1}{3}}\beta^{\frac{2}{3}}(\sigma^{\frac{2}{3}}+D_\beta^{\frac{1}{3}}b^{\frac{2}{3}})}{T^{\frac{2}{3}}}+\frac{\beta D_\beta}{T}\Big) \quad (8)$$

where $\bar{\boldsymbol{x}}^{(k)}=\frac{1}{n}\sum_{i=1}^n\boldsymbol{x}_i^{(k)}$. (Proof is in Appendix C.)

3.4. Comparison with Gossip SGD

To better illustrate how periodic global averaging helps relieve the affects of network topology in Gossip SGD, we list convergence rates of Gossip SGD and Gossip-PGA for smooth convex or non-convex problems in Table 4. The first line is the general rate expression for both algorithms. In the second line we let $D_\beta=\min\{H,1/(1-\beta)\}=1/(1-\beta)$ for Gossip-PGA, and in the third line we let $D_\beta=H$. According to this table, we derive the transient stages of Gossip SGD and Gossip-PGA for each scenarios (i.e., large/small network, iid/non-iid data distributions) in Table 2 (see the derivation detail in Appendix D). As we have explained in Main Results subsection in the introduction, it is observed from Tables 2 and 4 that: (i) Gossip-PGA always converges faster (or has shorter transient stages) than Gossip SGD for any β and H value. (ii) Such superiority gets evident for large and sparse networks where $\beta\to 1$.

Remark 5. *The convergence analysis in topology-changing Gossip SGD (Koloskova et al., 2020) covers Gossip-PGA. By letting $p=1$ and $\tau=H$ in Theorem 2 of (Koloskova et al., 2020), it is derived that Gossip-PGA has a transient stage on the order of $\Omega(n^3H^4)$ for non-convex non-iid scenario. Such transient stage cannot quantify the superiority to Gossip and Local SGDs. In fact, it may even show PGA can do harm to Gossip SGD when $H>\frac{1}{1-\beta}$, which is counter-intuitive. This is because (Koloskova et al., 2020) is for the general time-varying topology. It does not utilize*

	RATES
L-SGD	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\boldsymbol{H}^{\frac{1}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\boldsymbol{H}^{\frac{2}{3}}b^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{H}{T}\Big)$
G-PGA	$O\Big(\frac{\sigma}{\sqrt{nT}}+\frac{\boldsymbol{C}_{\beta}^{\frac{1}{3}}\boldsymbol{\beta}^{\frac{2}{3}}\sigma^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\boldsymbol{C}_{\beta}^{\frac{1}{3}}\boldsymbol{H}^{\frac{1}{3}}\boldsymbol{\beta}^{\frac{2}{3}}\boldsymbol{b}^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\beta H}{T}\Big)$

Table 6. Convergence rate comparison between Local SGD (L-SGD) and Gossip-PGA (G-PGA) over smooth convex/non-convex problems. The rate for Local SGD is from (Koloskova et al., 2020; Yu et al., 2019; Li et al., 2019b).

the structures listed in Sec. 3.1.

Transient stage in runtime. Table 2 compares transient stages between Gossip-PGA and Gossip SGD in *iterations*. But what people really care about in practice is *runtime*. Since both Gossip SGD and Gossip-PGA have the same computational overhead per iteration, we will focus on communication time spent in the transient stage.

Given the bandwidth in a computing cluster with size n, we let α denote the point-to-point latency in the network, and θ denote the communication time cost to transmit a scalar variable. Since variable x in problem (1) has dimension d, it will take θd time to transmit x between two nodes. Under this setting, the All-Reduce global averaging step will take $2\theta d + n\alpha = O(\theta d + n\alpha)$ time (see section 2.5 in (Ben-Nun & Hoefler, 2019)). The gossip-style communication time varies with different network topologies. For the commonly-used ring or grid topology, it takes $|\mathcal{N}_i|\theta d + \alpha = O(\theta d + \alpha)$ for one gossip communication, where $|\mathcal{N}_i|$ is the neighborhood size of node i, and $|\mathcal{N}_i| = 3$ for the ring and 5 for the grid. As to Gossip-PGA, if we amortize the periodic All-Reduce cost into each communication, it will have $|\mathcal{N}_i|\theta d + \alpha + (2\theta d + n\alpha)/H = O(\theta d + \sqrt{n}\alpha)$ when we set $H = \sqrt{n}$. With the formula Total time = transient stage (in iteration) × comm. per iter. We calculate and compare the transient time between non-iid Gossip-PGA and Gossip-SGD (over the grid topology) in Table 5. Other comparisons for iid scenario or the ring topology can be found in Appendix D. It is observed in all tables that Gossip-PGA has shorter transient time.

3.5. Comparison with Local SGD

The convergence rates of Gossip-PGA and Local SGD are listed in Table 6, from which we derive the transient stages of them in Table 3 (details are in Appendix D). As we have explained in the introduction, it is observed from Tables 3 and 6 that (i) Gossip-PGA always converges faster (or has shorter transient stages) than Local SGD for any β and H value, and (ii) Such superiority gets more evident for well-connected network where $\beta \to 0$.

As to the wall-clock transient time of Local SGD, if we amortize the periodic All-Reduce cost into each local update, it will take $(2\theta d + n\alpha)/H = O(\theta d/H + n\alpha/H)$ communication time per iteration. Using the transient iteration derived in Table 3, the total transient time for Local SGD (non-iid scenario) will be $O(n^3H^3(\theta d + n\alpha))$. Comparing it with the total transient time $O(n^3HC_\beta^2\beta^4(H\theta d + n\alpha))$ for Gossip-PGA, we find Gossip-PGA always has shorter transient runtime for a large $H > \beta^4 C_\beta^2$.

Remark 6. *While we discuss in detail that the transient time of Gossip-PGA is shorter than Gossip and Local SGDs, it is worth noting that the communication time during the linear speedup stage (i.e., after the transient stage) also contributes to the total training time. In this stage, Gossip-PGA is less efficient due to its periodic global averaging. However, we illustrate that Gossip-PGA is always endowed with shorter total training time than Gossip and Local SGDs with extensive deep learning experiments in Sec. 5.*

4. Gossip SGD with Adaptive Global Average

Gossip-PGA suffers from the burden of tuning H by hand. A small H will incur more communication overhead while a large value can slow down the convergence. We further propose Gossip-AGA, an adaptive extension of Gossip-PGA.

Intuition. A small consensus variance $\sum_{i=1}^n \mathbb{E}\|\boldsymbol{x}_i - \bar{\boldsymbol{x}}\|^2$ would accelerate Gossip-PGA. To see that, if $\sum_{i=1}^n \mathbb{E}\|\boldsymbol{x}_i - \bar{\boldsymbol{x}}\|^2 = 0$ for each iteration, then Gossip-PGA reduces to parallel SGD and can reach its fastest convergence. Recall from Lemma 8 in the appendix that the averaged consensus $\frac{1}{T+1}\sum_{k=0}^T \mathbb{E}\|\mathbf{x}^{(k)} - \bar{\mathbf{x}}^{(k)}\|^2$ is bounded by $\frac{d_1\gamma^2}{T+1}\sum_{k=0}^T \mathbb{E}\|\nabla f(\bar{\boldsymbol{x}}^{(k)})\|^2 + d_2\gamma^2$ where d_1 and d_2 are constants. It is observed that the initial consensus variance (when T is small) can be significant due to large γ and $\mathbb{E}\|\nabla f(\bar{\boldsymbol{x}}^{(k)})\|^2$. In the later stage when T is sufficiently large, both the diminishing step-size γ and gradient $\mathbb{E}\|\nabla f(\bar{\boldsymbol{x}}^{(k)})\|^2$ go to 0 and hence leading to a small consensus variance naturally. With these observations, it is intuitive to take global synchronizations more frequently in initial stages to reduce the overall consensus variance.

Convergence. We denote $H^{(\ell)}$ as the duration of the ℓ-th period. The following corollary establishes convergence for Gossip-PGA with any time-varying but finite global averaging period sequence $\{H^{(\ell)}\}$:

Corollary 1. *Suppose Assumptions 1–3 and 5 hold and the time-varying period $H^{(\ell)}$ is upper bounded by $H_{\max} = \max_{\ell\ge 0}\{H^{(\ell)}\}$. If γ satisfies the condition in Theorem 1 with $H = H_{\max}$, then Gossip-AGA converges at rate (8) in which H is replaced by $H_{\max}$. (Proof is in Appendix E.)*

Adaptive Strategy. This subsection will propose an adaptive strategy that is inspired by (Wang & Joshi, 2019). If we recover the influence of the initial value $F_0 = \mathbb{E}f(\bar{\boldsymbol{x}}^{(0)})$ on convergence rate (8), Gossip-PGA for non-convex problems will converge at

$$O\Big(\frac{\sigma F_0^{\frac{1}{2}}}{\sqrt{nT}}+\frac{H^{\frac{1}{3}}\beta^{\frac{2}{3}}\sigma^{\frac{2}{3}}F_0^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{H^{\frac{2}{3}}\beta^{\frac{2}{3}}\hat{b}^{\frac{2}{3}}F_0^{\frac{2}{3}}}{T^{\frac{2}{3}}}+\frac{\beta D_\beta F_0}{T}\Big).$$

For a fixed T, a period $H=\sigma^{\frac{3}{2}}T^{\frac{1}{4}}/(\beta\hat{b}F_0^{\frac{1}{4}}n^{\frac{3}{4}})$ will guarantee the linear speedup. Therefore, the initial period $H^{(0)}$ can be chosen as $H^{(0)}=d_1/[\mathbb{E}f(\bar{\boldsymbol{x}}^{(0)})]^{\frac{1}{4}}$ for some constant d_1. Similarly, for the ℓ-th period, workers can be viewed as restarting training at a new initial point $\bar{\boldsymbol{x}}^{(T_{\ell-1})}$ where $T_{\ell-1}=H^{(0)}+\cdots+H^{(\ell-1)}$. As a result, the ℓ-th period $H^{(\ell)}$ can be chosen as $H^{(\ell)}=d_1/[\mathbb{E}f(\bar{\boldsymbol{x}}^{(T_{\ell-1})})]^{\frac{1}{4}}$. With such choice of $H^{(0)}$ and $H^{(\ell)}$, it is not difficult to have

$$H^{(\ell)}=\Big(\frac{\mathbb{E}f(\bar{\boldsymbol{x}}^{(0)})}{\mathbb{E}f(\bar{\boldsymbol{x}}^{(T_{\ell-1})})}\Big)^{\frac{1}{4}}H^{(0)}. \tag{9}$$

Since $\mathbb{E}f(\bar{\boldsymbol{x}}^{(k)})$ will decrease as k increases, (9) will generate an increasing sequence of period $H^{(\ell)}$. We list Gossip-AGA as Algorithm 2 in Appendix G and elaborate on implementation details there.

5. Experimental Results

In this section, we first examine how the transient stage differs for Gossip-PGA, Gossip and Local SGDs on networks with different topology and size on convex logistic regression. Next, we systematically evaluate the aforementioned methods on two typical large-scale deep learning tasks: image classification (over 256 GPUs) and language modeling (over 64 GPUs). See Appendix F for implementation details.

5.1. Logistic Regression

We consider a distributed logistic regression problem with $f_i(x)=\frac{1}{M}\sum_{m=1}^{M}\ln[1+\exp(-y_{i,m}h_{i,m})^Tx]$, where $\{h_{i,m},y_{i,m}\}_{m=1}^{M}$ are local data samples at agent i with $h_{i,m}\in\mathbb{R}^d$ being the feature vector and $y_{i,m}\in\{+1,-1\}$ being the corresponding label. Each $h_{i,m}$ is generated from the normal distribution $\mathcal{N}(0;10I_d)$. To generate $y_{i,m}$, we first generate an auxiliary random vector $x_i^\star\in\mathbb{R}^d$ with each entry following $\mathcal{N}(0,1)$. Next, we generate $y_{i,m}$ from a uniform distribution $\mathcal{U}(0,1)$. If $y_{i,m}\le 1/[1+\exp(-h_{i,m}^Tx_i^\star)]$ then $y_{i,m}$ is set as $+1$; otherwise $y_{i,m}$ is set as -1. We let $x_i^\star=x^\star\ \forall i$ to generate data for iid scenario and $x_i^\star\neq x_j^\star\ \forall i,j$ for non-iid scenario. Each $x_i^\star$ is normalized.

Figure 1 compares how Gossip-PGA performs against parallel and Gossip SGD over the ring topology and non-iid data distribution. The network sizes are set as $n=20,50,100$ which results in $\beta=0.967,0.995,0.998$. We set $d=10$ and $M=8000$. H is set as 16 in Gossip-PGA. The stepsize γ is initialized as 0.2 and gets decreased by half for every 1000 iterations. We repeat all simulations 50 times and illustrate the mean of all trials with solid curve and standard deviation with shaded area. It is observed that both Gossip SGD and Gossip-PGA will asymptotically converge at the same rate as parallel SGD (i.e., the linear speedup stage), albeit with different transient stages. Gossip-PGA always has shorter transient stages than Gossip SGD, and such superiority gets more evident when network size increases (recall that $1-\beta=O(1/n^2)$). For experiments on different topologies such as grid and exponential graph, on iid data distribution, and comparison with Local SGD, see Appendix F. All experiments are consistent with the theoretical transient stage comparisons in Tables 2 and 3.

Method	Acc.%	Hrs	Epochs/Hrs to 76%.
Parallel SGD	76.26	2.22	94 / 1.74
Local SGD	74.20	1.05	N.A.
Local SGD ×3	75.41	3	N.A.
Gossip SGD	75.34	1.55	N.A.
Gossip SGD ×2	76.18	3	198/2.55
OSGP	75.04	1.32	N.A.
OSGP ×2	76.07	2.59	212/2.28
Gossip-PGA	76.28	1.66	109/1.50
Gossip-AGA	76.25	1.57	**91/1.20**

Table 7. Comparison of Top-1 validation accuracy (Column 2) and wall-clock training time (Column 3) on different methods after finishing all epochs. We also report the epochs and training time required to reach 76% accuracy (Column 4). "N.A." implies that the target accuracy is not reached when all epochs are completed.

5.2. Image Classification

The ImageNet-1k (Deng et al., 2009) [1] dataset consists of 1,281,167 training images and 50,000 validation images in 1000 classes. We train ResNet-50 (He et al., 2016) model (∼25.5M parameters) following the training protocol of (Goyal et al., 2017). We train total 120 epochs. The learning rate is warmed up in the first 5 epochs and is decayed by a factor of 10 at 30, 60 and 90 epochs. We set the period to 6 for both Local SGD and Gossip-PGA. In Gossip-AGA, the period is set to 4 initially and changed adaptively afterwards, roughly 9% iterations conduct global averaging.

Table 7 shows the top-1 validation accuracy and wall-clock training time of aforementioned methods. It is observed both Gossip-PGA and Gossip-AGA can reach comparable accuracy with parallel SGD after all 120 epochs but with roughly 1.3x $\sim$ 1.4x training time speed-up. On the other hand, while local and Gossip SGD completes all 120 epochs faster than Gossip-PGA/AGA and parallel SGD, they suffer from a 2.06% and 0.92% accuracy degradation separately. Moreover, both algorithms cannot reach the 76% top-1 accuracy within 120 epochs. We also compare with OSGP (Assran et al., 2019), which adding overlapping on the Gossip

[1] The usage of ImageNet dataset in this paper is for non-commercial research purposes only.

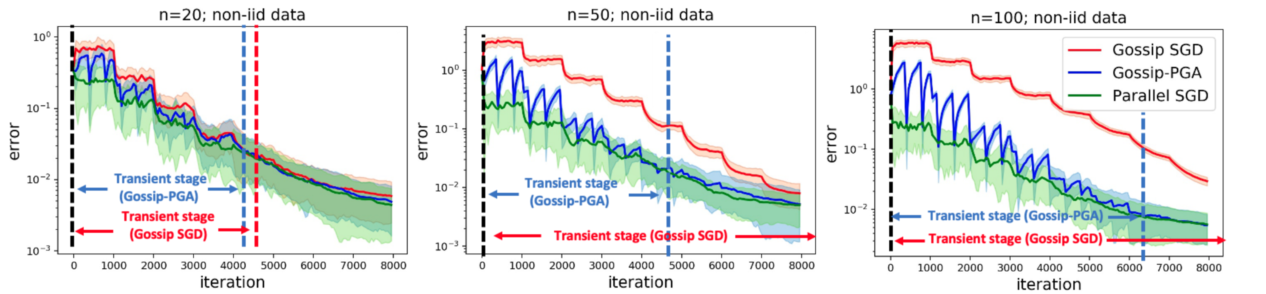

Figure 1. Convergence comparison between Gossip-PGA, Gossip and parallel SGDs on the logistic regression problem over ring topology. The transient stage is determined by counting iterations before an algorithm exactly matches the convergence curve of Parallel SGD. Note that the transient stage for Gossip SGD in the middle and right sub-figures is beyond the plotting canvas.

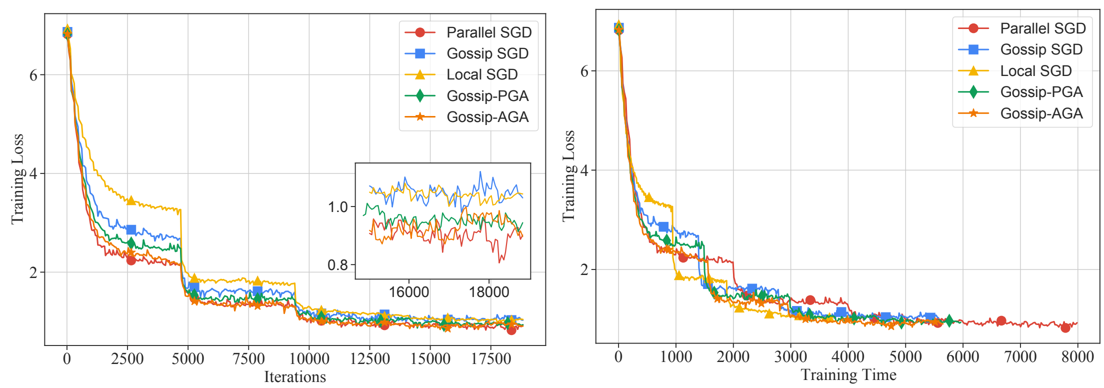

Figure 2. Convergence results on the ImageNet in terms of iteration and runtime. More results are in Appendix F.3.

SGD. We find OSGP $\times 2$, while faster than Gossip SGD$\times 2$, still needs more time than Gossip-PGA to achieve 76% accuracy. To further illustrate how much time it will take local and Gossip SGD to reach the target accuracy, we run another Local SGD and Gossip SGD experiments with extended epochs (i.e., Gossip SGD $\times 2$ trains total 240 epochs and the learning rate is decayed at 60, 120, and 180 epoch. Local SGD $\times 3$ trains total 360 epochs and the learning rate is decayed at 90, 180, and 270 epochs). It is observed that Gossip-SGD $\times 2$ can reach the target with notably more time expense than Gossip-PGA/AGA and parallel SGD, and Local SGD $\times 3$ still cannot reach the 76% accuracy. All these observations validate that periodic global averaging can accelerate Gossip SGD significantly.

Figure 2 shows the iteration-wise and runtime-wise convergence in terms of training loss. In the left figure, it is observed Gossip-PGA/AGA converges faster (in iteration) and more accurate than local and Gossip SGD, which is consistent with our theory. In the right figure, it is observed that Gossip-PGA/AGA is the fastest method (in time) that can reach the same training loss as parallel SGD.

Compare with SlowMo. Gossip-PGA is an instance of SlowMo, in which the base optimizer is set as Gossip SGD, slow momentum $\beta = 0$, and slow learning rate $\alpha = 1$. We made experiments to compare Gossip-PGA with SlowMo. It is observed the additional slow momentum update helps SlowMo with large H but degrades it when H is small. This observation is consistent with Fig. 3(a) in (Wang et al., 2019). This observation implies that the slow momentum update may not always be beneficial in SlowMo.

Period	Gossip-PGA	SlowMo
$H = 6$	**76.28**	75.23
$H = 48$	75.66	**75.81**

Table 8. Comparison of Top-1 validation accuracy with SlowMo with different periods.

Ring Topology. While the convergence property of Gossip-PGA is established over the *static* network topology, we utilize the dynamic one-peer exponential topology in the above deep experiments because it usually achieves better accuracy. To illustrate the derived theoretical results, we make an additional experiment, over the static ring topology, to compare Gossip-PGA with Gossip SGD in Table 9. It is observed that Gossip-PGA can achieve better accuracy than Gossip SGD after running the same epochs, which coincides with our analysis that Gossip-PGA has faster convergence.

Scalability. We establish in Theorem 2 that Gossip-PGA can achieve linear speedup in the non-convex setting. To

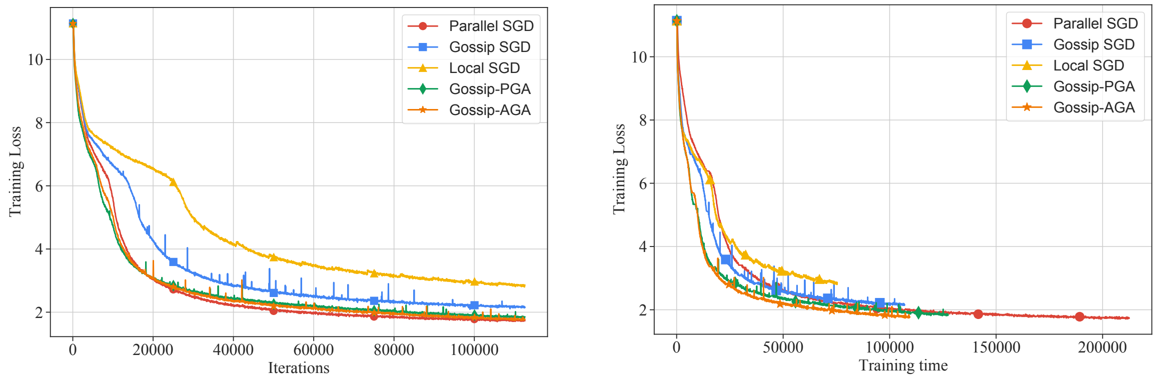

Figure 3. Convergence results of BERT on the language modeling task in terms of iteration and runtime.

Method	Epoch	Acc%	Time(Hrs.)
Gossip SGD	120	74.86	1.56
Gossip PGA	120	75.94	1.68

Table 9. Comparison of Top-1 validation accuracy on Gossip-PGA and Gossip SGD with ring topology.

validate it, we conduct a scaling experiment and list the result in Table 10. Figures represent the final accuracy and hours to finish training. It is observed that Gossip-PGA can achieve a roughly linear speedup in training time without notably performance degradation.

Method	4 nodes	8 nodes	16 nodes	32 nodes
Parallel SGD	76.3/11.6	76.4/6.3	76.3/3.7	76.2/2.2
Gossip SGD	76.3/11.1	76.4/5.7	75.9/2.8	75.0/1.5
Gossip PGA	76.4/11.2	76.7/5.9	76.3/3.0	76.2/1.6

Table 10. Scaling effects on different methods with different numbers of nodes. Figures represent the final accuracy and hours to complete training.

5.3. Language Modeling

BERT (Devlin et al., 2018) is a widely used pre-training language representation model for NLP tasks. We train a BERT-Large model (∼330M parameters) on the Wikipedia and BookCorpus datasets. We set the period to 6 for both Local SGD and Gossip-PGA. In Gossip-AGA, the period is set to 4 initially and changed adaptively afterwards, roughly 9.6% iterations conduct global averaging.

Method	Final Loss	Runtime (hrs)
Parallel SGD	1.75	59.02
Local SGD	2.85	20.93
Local SGD ×3	1.88	60
Gossip SGD	2.17	29.7
Gossip SGD ×2	1.81	59.7
Gossip-PGA	1.82	35.4
Gossip-AGA	1.77	30.4

Table 11. Comparison of training loss and training time of BERT training on different algorithms after completing all training steps.

Table 11 shows the final training loss and training runtime of the aforementioned methods. Gossip-AGA can reach comparable training loss with parallel SGD, but with roughly 1.94 x training time speed-up. Gossip SGD and Local SGD cannot reach training loss that below 1.8 even if they are trained over 60 hours (see Local SGD ×3 and Gossip SGD ×2.) Figure 3 shows the iteration-wise and runtime-wise convergence w.r.t training loss of the aforementioned methods. The left plot shows Gossip-PGA/AGA has almost the same convergence as Gossip SGD in iterations; the right plot shows that Gossip-AGA is the fastest method in training time that can reach the same accuracy as parallel SGD.

6. Conclusion

We introduce Gossip-PGA/AGA to mitigate the slow convergence rate of Gossip SGD in distributed training. Theoretically, we prove the convergence improvement in smooth convex and non-convex problem. Empirically, experimental results of large-scale training validate our theories.

References

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. Qsgd: Communication-efficient sgd via gradient quantization and encoding. In *Advances in Neural Information Processing Systems*, pp. 1709–1720, 2017.

Assran, M., Loizou, N., Ballas, N., and Rabbat, M. Stochastic gradient push for distributed deep learning. In *International Conference on Machine Learning (ICML)*, pp. 344–353, 2019.

Bayoumi, A. K. R., Mishchenko, K., and Richtarik, P. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pp. 4519–4529, 2020.

Ben-Nun, T. and Hoefler, T. Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. *ACM Computing Surveys (CSUR)*, 52(4):1–43, 2019.

Berahas, A. S., Bollapragada, R., Keskar, N. S., and Wei, E. Balancing communication and computation in distributed optimization. *IEEE Transactions on Automatic Control*, 64(8):3141–3155, 2018.

Bernstein, J., Zhao, J., Azizzadenesheli, K., and Anandkumar, A. signsgd with majority vote is communication efficient and fault tolerant. *arXiv preprint arXiv:1810.05291*, 2018.

Chen, J. and Sayed, A. H. Diffusion adaptation strategies for distributed optimization and learning over networks. *IEEE Transactions on Signal Processing*, 60(8):4289–4305, 2012.

Chen, T., Giannakis, G., Sun, T., and Yin, W. LAG: Lazily aggregated gradient for communication-efficient distributed learning. In *Advances in Neural Information Processing Systems*, pp. 5050–5060, 2018.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 248–255. Ieee, 2009.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Di Lorenzo, P. and Scutari, G. Next: In-network nonconvex optimization. *IEEE Transactions on Signal and Information Processing over Networks*, 2(2):120–136, 2016.

Duchi, J. C., Agarwal, A., and Wainwright, M. J. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3):592–606, 2011.

Goyal, P., Dollár, P., Girshick, R., Noordhuis, P., Wesolowski, L., Kyrola, A., Tulloch, A., Jia, Y., and He, K. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

He, L., Bian, A., and Jaggi, M. Cola: Decentralized linear learning. In *Advances in Neural Information Processing Systems*, pp. 4536–4546, 2018.

Koloskova, A., Lin, T., Stich, S. U., and Jaggi, M. Decentralized deep learning with arbitrary communication compression. In *International Conference on Learning Representations*, 2019a.

Koloskova, A., Stich, S., and Jaggi, M. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *International Conference on Machine Learning*, pp. 3478–3487, 2019b.

Koloskova, A., Loizou, N., Boreiri, S., Jaggi, M., and Stich, S. U. A unified theory of decentralized sgd with changing topology and local updates. In *International Conference on Machine Learning (ICML)*, pp. 1–12, 2020.

Li, M., Andersen, D. G., Park, J. W., Smola, A. J., Ahmed, A., Josifovski, V., Long, J., Shekita, E. J., and Su, B.-Y. Scaling distributed machine learning with the parameter server. In *11th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 14)*, pp. 583–598, 2014.

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2019a.

Li, X., Yang, W., Wang, S., and Zhang, Z. Communication efficient decentralized training with multiple local updates. *arXiv preprint arXiv:1910.09126*, 2019b.

Li, Z., Shi, W., and Yan, M. A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates. *IEEE Transactions on Signal Processing*, July 2019c. early acces. Also available on arXiv:1704.07807.

Lian, X., Zhang, C., Zhang, H., Hsieh, C.-J., Zhang, W., and Liu, J. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. In *Advances in Neural Information Processing Systems*, pp. 5330–5340, 2017.

Lian, X., Zhang, W., Zhang, C., and Liu, J. Asynchronous decentralized parallel stochastic gradient descent. In *International Conference on Machine Learning*, pp. 3043–3052, 2018.

Lin, T., Stich, S. U., Patel, K. K., and Jaggi, M. Don't use large mini-batches, use local sgd. *arXiv preprint arXiv:1808.07217*, 2018.

Liu, Y., Xu, W., Wu, G., Tian, Z., and Ling, Q. Communication-censored admm for decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 67(10):2565–2579, 2019.

Lu, S., Zhang, X., Sun, H., and Hong, M. Gnsd: A gradient-tracking based nonconvex stochastic algorithm for decentralized optimization. In *2019 IEEE Data Science Workshop (DSW)*, pp. 315–321. IEEE, 2019.

Luo, Q., He, J., Zhuo, Y., and Qian, X. Prague: High-performance heterogeneity-aware asynchronous decentralized training. In *Proceedings of the Twenty-Fifth International Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 401–416, 2020.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pp. 1273–1282. PMLR, 2017.

Nedic, A. and Ozdaglar, A. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

Nedic, A., Olshevsky, A., and Shi, W. Achieving geometric convergence for distributed optimization over time-varying graphs. *SIAM Journal on Optimization*, 27(4): 2597–2633, 2017.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 8024–8035, 2019.

Patarasuk, P. and Yuan, X. Bandwidth optimal all-reduce algorithms for clusters of workstations. *Journal of Parallel and Distributed Computing*, 69(2):117–124, 2009.

Qu, G. and Li, N. Harnessing smoothness to accelerate distributed optimization. *IEEE Transactions on Control of Network Systems*, 5(3):1245–1260, 2018.

Scaman, K., Bach, F., Bubeck, S., Lee, Y. T., and Massoulié, L. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *International Conference on Machine Learning*, pp. 3027–3036, 2017.

Scaman, K., Bach, F., Bubeck, S., Massoulié, L., and Lee, Y. T. Optimal algorithms for non-smooth distributed optimization in networks. In *Advances in Neural Information Processing Systems*, pp. 2740–2749, 2018.

Shi, W., Ling, Q., Yuan, K., Wu, G., and Yin, W. On the linear convergence of the admm in decentralized consensus optimization. *IEEE Transactions on Signal Processing*, 62(7):1750–1761, 2014.

Shi, W., Ling, Q., Wu, G., and Yin, W. EXTRA: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

Stich, S. U. Local sgd converges fast and communicates little. In *International Conference on Learning Representations (ICLR)*, 2019.

Tang, H., Lian, X., Yan, M., Zhang, C., and Liu, J. d^2: Decentralized training over decentralized data. In *International Conference on Machine Learning*, pp. 4848–4856, 2018.

Tang, H., Yu, C., Lian, X., Zhang, T., and Liu, J. Doublesqueeze: Parallel stochastic gradient descent with double-pass error-compensated compression. In *International Conference on Machine Learning*, pp. 6155–6165. PMLR, 2019.

Tsitsiklis, J., Bertsekas, D., and Athans, M. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE transactions on automatic control*, 31(9):803–812, 1986.

Uribe, C. A., Lee, S., Gasnikov, A., and Nedić, A. A dual approach for optimal algorithms in distributed optimization over networks. *Optimization Methods and Software*, pp. 1–40, 2020.

Wang, J. and Joshi, G. Adaptive communication strategies to achieve the best error-runtime trade-off in local-update sgd. In *Systems and Machine Learning (SysML) Conference*, 2019.

Wang, J., Tantia, V., Ballas, N., and Rabbat, M. SlowMo: Improving communication-efficient distributed sgd with slow momentum. *arXiv preprint arXiv:1910.00643*, 2019.

Xin, R., Khan, U. A., and Kar, S. An improved convergence analysis for decentralized online stochastic non-convex optimization. *arXiv preprint arXiv:2008.04195*, 2020.

Xu, J., Zhu, S., Soh, Y. C., and Xie, L. Augmented distributed gradient methods for multi-agent optimization under uncoordinated constant stepsizes. In *IEEE Conference on Decision and Control (CDC)*, pp. 2055–2060, Osaka, Japan, 2015.

You, Y., Li, J., Reddi, S., Hseu, J., Kumar, S., Bhojanapalli, S., Song, X., Demmel, J., Keutzer, K., and Hsieh, C.-J. Large batch optimization for deep learning: Training bert in 76 minutes. In *International Conference on Learning Representations*, 2019.

Yu, H., Yang, S., and Zhu, S. Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 5693–5700, 2019.

Yuan, K., Ling, Q., and Yin, W. On the convergence of decentralized gradient descent. *SIAM Journal on Optimization*, 26(3):1835–1854, 2016.

Yuan, K., Ying, B., Zhao, X., and Sayed, A. H. Exact dffusion for distributed optimization and learning – Part I: Algorithm development. *IEEE Transactions on Signal Processing*, 67(3):708 – 723, 2019.

Yuan, K., Alghunaim, S. A., Ying, B., and Sayed, A. H. On the influence of bias-correction on distributed stochastic optimization. *IEEE Transactions on Signal Processing*, 2020.

Zhang, J., De Sa, C., Mitliagkas, I., and Ré, C. Parallel sgd: When does averaging help? *arXiv preprint arXiv:1606.07365*, 2016.

Zinkevich, M., Weimer, M., Li, L., and Smola, A. J. Parallelized stochastic gradient descent. In *Advances in neural information processing systems*, pp. 2595–2603, 2010.