Decentralized Stochastic Optimization and Gossip Algorithms with Compressed Communication

Anastasia Koloskova [*1] **Sebastian U. Stich** [*1] **Martin Jaggi** [1]

Abstract

We consider decentralized stochastic optimization with the objective function (e.g. data samples for machine learning tasks) being distributed over n machines that can only communicate to their neighbors on a fixed communication graph. To address the communication bottleneck, the nodes compress (e.g. quantize or sparsify) their model updates. We cover both unbiased and biased compression operators with quality denoted by $\delta \leq 1$ ($\delta = 1$ meaning no compression).

We (i) propose a novel gossip-based stochastic gradient descent algorithm, CHOCO-SGD, that converges at rate $\mathcal{O}\left(1/(nT) + 1/(T\rho^2\delta)^2\right)$ for strongly convex objectives, where T denotes the number of iterations and ρ the eigengap of the connectivity matrix. We (ii) present a novel gossip algorithm, CHOCO-GOSSIP, for the average consensus problem that converges in time $\mathcal{O}(1/(\rho^2\delta)\log(1/\varepsilon))$ for accuracy $\varepsilon > 0$. This is (up to our knowledge) the first gossip algorithm that supports arbitrary compressed messages for $\delta > 0$ and still exhibits linear convergence. We (iii) show in experiments that both of our algorithms do outperform the respective state-of-the-art baselines and CHOCO-SGD can reduce communication by at least two orders of magnitudes.

1. Introduction

Decentralized machine learning methods are becoming core aspects of many important applications, both in view of scalability to larger datasets and systems, but also from the perspective of data locality, ownership and privacy. We consider decentralized optimization methods that do not rely on a central coordinator (e.g. parameter server) but instead only require on-device computation and local communication with neighboring devices. This covers for instance the classic setting of training machine learning models in large data-centers, but also emerging applications were the computations are executed directly on the consumer devices, which keep their part of the data private at all times.[1]

Formally, we consider optimization problems distributed across n devices or nodes of the form

$$f^\star := \min_{\mathbf{x}\in\mathbb{R}^d} \left[f(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}) \right], \tag{1}$$

where $f_i \colon \mathbb{R}^d \to \mathbb{R}$ for $i \in [n] := \{1, \dots, n\}$ are the objectives defined by the data available locally on each node. We also allow each local objective f_i to have stochastic optimization (or sum) structure, covering the important case of empirical risk minimization in distributed machine learning and deep learning applications.

Decentralized Communication. We model the network *topology* as a graph where edges represent the communication links along which messages (e.g. model updates) can be exchanged. The decentralized setting is motivated by centralized topologies (corresponding to a star graph) often not being possible, and otherwise often posing a significant bottleneck on the central node in terms of communication latency, bandwidth and fault tolerance. Decentralized topologies avoid these bottlenecks and thereby offer hugely improved potential in scalability. For example, while the master node in the centralized setting receives (and sends) in each round messages from all workers, $\Theta(n)$ in total[2], in decentralized topologies the maximal degree of the network is often constant (e.g. ring or torus) or a slowly growing function in n (e.g. scale-free networks).

Decentralized Optimization. For the case of deterministic (full-gradient) optimization, recent seminal theoretical advances show that the network topology only affects higher-order terms of the convergence rate of decentralized optimization algorithms on convex problems (Scaman et al.,

[*]Equal contribution [1]EPFL, Lausanne, Switzerland. Correspondence to: Anastasia Koloskova <anastasia.koloskova@epfl.ch>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019.

[1]Note the optimization process itself (as for instance the computed result) might leak information about the data of other nodes. We do not focus on quantifying notions of privacy in this work.

[2]For better connected topologies sometimes more efficient all-reduce and broadcast implementations are available.

2017; 2018). We prove the first analogue result for the important case of decentralized stochastic gradient descent (SGD), proving convergence at rate $\mathcal{O}(1/(nT))$ (ignoring for now higher order terms) on strongly convex functions where T denotes the number of iterations.

This result is significant since stochastic methods are highly preferred for their efficiency over deterministic gradient methods in machine learning applications. Our algorithm, CHOCO-SGD, is as efficient in terms of iterations as centralized mini-batch SGD (and consequently also achieves a speedup of factor n compared to the serial setting on a single node) but avoids the communication bottleneck that centralized algorithms suffer from.

Communication Compression. In distributed training, model updates (or gradient vectors) have to be exchanged between the worker nodes. To reduce the amount of data that has to be sent, gradient *compression* has become a popular strategy. These ideas have recently been introduced also to the decentralized setting by Tang et al. (2018a). However, their analysis only covers unbiased compression operators with very (unreasonably) high accuracy constraints. Here we propose the first method that supports arbitrary low accuracy and even biased compression operators, such as in (Alistarh et al., 2018; Lin et al., 2018; Stich et al., 2018).

Contributions. Our contributions can be summarized as:

- We show that the proposed CHOCO-SGD converges at rate $\mathcal{O}(1/(nT)+1/(T\rho^2\delta)^2)$, where T denotes the number of iterations, n the number of workers, ρ the eigengap of the gossip (connectivity) matrix and $\delta \leq 1$ the compression quality factor ($\delta = 1$ meaning no compression). Despite ρ and δ affecting the higher order terms, the first term in the rate, $\mathcal{O}(1/(nT))$, is the same as for the centralized baseline with exact communication, achieving the same speedup as centralized mini-batch SGD when the number n of workers grows. This is verified experimentally on the ring topology and by reducing the communication by a factor of 100 ($\delta = \frac{1}{100}$).
- We present the first linearly-converging gossip algorithm with communication compression for the distributed average consensus problem. Our algorithm, CHOCO-GOSSIP, converges at linear rate $\mathcal{O}(1/(\rho^2\delta)\log(1/\varepsilon))$ for accuracy $\varepsilon > 0$, and allows arbitrary communication compression operators (including biased and unbiased ones). In contrast, previous works either exhibited sublinear convergence, or required very high-precision quantization $\delta \approx 1$, or could only show convergence towards a neighborhood of the optimal solution.
- CHOCO-SGD significantly outperforms state-of-the-art methods for decentralized optimization with gradient compression, such as ECD-SGD and DCD-SGD introduced in (Tang et al., 2018a), in all our experiments.

2. Related Work

Stochastic gradient descent (SGD) (Robbins & Monro, 1951; Bottou, 2010) and variants thereof are the standard algorithms for machine learning problems of the form (1), though it is an inherently serial algorithm that does not take the distributed setting into account. Mini-batch SGD (Dekel et al., 2012) is the natural parallelization of SGD for (1) in the centralized setting, i.e. when a master node collects the updates from all worker nodes, and serves a baseline here.

Decentralized Optimization. The study of decentralized optimization algorithms can be tracked back at least to the 1980s (Tsitsiklis, 1984). Decentralized algorithms are sometimes referred to as *gossip* algorithms (Kempe et al., 2003; Xiao & Boyd, 2004; Boyd et al., 2006) as the information is not broadcasted by a central entity, but spreads—similar to gossip—along the edges specified by the communication graph. The most popular algorithms are based on (sub)gradient descent (Nedić & Ozdaglar, 2009; Johansson et al., 2010), alternating direction method of multipliers (ADMM) (Wei & Ozdaglar, 2012; Iutzeler et al., 2013) or dual averaging (Duchi et al., 2012; Nedić et al., 2015). He et al. (2018) address the more specific problem class of generalized linear models.

For the deterministic (non-stochastic) convex version of (1) a recent line of work developed optimal algorithms based on acceleration (Jakovetić et al., 2014; Scaman et al., 2017; 2018; Uribe et al., 2018). Reisizadeh et al. (2018) and Doan et al. (2018) studied quantization in this setting. Reisizadeh et al. (2018) could achieve only sublinear rate for smooth and strongly convex objectives, while (Doan et al., 2018) considered non-smooth objectives and provided sublinear rates, matching optimal rates up to logarithmic factor (Scaman et al., 2018). Rates for the stochastic setting are derived in (Shamir & Srebro, 2014; Rabbat, 2015), under the assumption that the distributions on all nodes are equal. Such an i.i.d. assumption is a strong restriction which prohibits most distributed machine learning applications, for example also federated learning setting (McMahan et al., 2017). Our algorithm CHOCO-SGD overcomes this and is free of i.i.d. assumptions. Also, (Rabbat, 2015) requires multiple communication rounds per stochastic gradient computation and so is not suited for sparse communication, as the required number of communication rounds would increase proportionally to the sparsity. Lan et al. (2018) applied gradient sliding techniques allowing to skip some of the communication rounds. Assran et al. (2019) have studied time-varying networks; (Yu et al., 2019) in the case of parameter servers. Lian et al. (2017); Tang et al. (2018b;a); Assran et al. (2019) consider the non-convex setting with Tang et al. (2018a) also applying gradient quantization techniques to reduce the communication cost. However, their algorithms require very high precision quantization, a constraint we overcome.

Gradient Compression. Instead of transmitting a full dimensional (gradient) vector $\mathbf{g} \in \mathbb{R}^d$, methods with gradient compression transmit a compressed vector $Q(\mathbf{g})$ instead, where $Q\colon \mathbb{R}^d \to \mathbb{R}^d$ is a (random) operator chosen such that $Q(\mathbf{g})$ can be more efficiently represented, for instance by using limited bit representation (*quantization*) or enforcing *sparsity*. A class of very common quantization operators is based on random dithering (Goodall, 1951; Roberts, 1962) that is in addition also unbiased, $\mathbb{E}_\xi\, Q(\mathbf{x}) = \mathbf{x}$, $\forall \mathbf{x} \in \mathbb{R}^d$, see (Alistarh et al., 2017; Wen et al., 2017; Zhang et al., 2017). Much sparser vectors can be obtained by random sparsification techniques that randomly mask the input vectors and only preserve a constant number of coordinates (Wangni et al., 2018; Konecny & Richtárik, 2018; Stich et al., 2018). Techniques that do not directly quantize gradients, but instead maintain additional states are known to perform better in theory and practice (Seide et al., 2014; Lin et al., 2018; Stich et al., 2018), an approach that we pick up here. Our analysis also covers deterministic and biased compression operators, such as in (Alistarh et al., 2018; Stich et al., 2018). We will not further distinguish between sparsification and quantization approaches, and refer to both of them as *compression* operators in the following.

Distributed Average Consensus. The average consensus problem consists in finding the average vector of n local vectors (see (2) below for a formal definition). The problem is an important sub-routine of many decentralized algorithms. Gossip-type algorithms converge linearly for average consensus (Kempe et al., 2003; Xiao & Boyd, 2004; Olfati-Saber & Murray, 2004; Boyd et al., 2006). However, for consensus algorithms with compressed communication it has been remarked that the standard gossip algorithm does not converge to the correct solution (Xiao et al., 2005). The proposed schemes in (Carli et al., 2007; Nedić et al., 2008; Aysal et al., 2008; Carli et al., 2010b; Yuan et al., 2012) do only converge to a neighborhood (whose size depends on the compression accuracy) of the solution.
In order to converge, adaptive schemes (with varying compression accuracy) have been proposed (Carli et al., 2010a; Fang & Li, 2010; Li et al., 2011; Thanou et al., 2013). However, these approaches fall back to full (uncompressed) communication to reach high accuracy. In contrast, our method converges linearly, even for arbitrary compressed communication, without requiring adaptive accuracy. We are not aware of a method in the literature with similar guarantees.

3. Average Consensus with Communication Compression

In this section we present CHOCO-GOSSIP, a novel gossip algorithm for distributed average consensus with compressed communication. The average consensus problem is an important special case of type (1), and formalized as

$$\overline{\mathbf{x}} := \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i\,, \tag{2}$$

for vectors $\mathbf{x}_i \in \mathbb{R}^d$ distributed on n nodes (consider $f_i(\mathbf{x}) = \frac{1}{2}\left\|\mathbf{x}-\mathbf{x}_i\right\|^2$ in (1)). Our proposed algorithm will later serve as a crucial primitive in our optimization algorithm for the general optimization problem (1), but is of independent interest for any average consensus problem with communication constraints.

In Sections 3.1–3.3 below we first review existing schemes that we later consider as baselines for the numerical comparison. The novel algorithm follows in Section 3.4.

3.1. Gossip algorithms

The classic decentralized algorithms for the average consensus problem are *gossip* type algorithms (see e.g. (Xiao & Boyd, 2004)) that generate sequences $\{\mathbf{x}_i^{(t)}\}_{t\geq 0}$ on every node $i \in [n]$ by iterations of the form

$$\mathbf{x}_i^{(t+1)} := \mathbf{x}_i^{(t)} + \gamma \sum_{j=1}^n w_{ij}\Delta_{ij}^{(t)}\,. \tag{3}$$

Here $\gamma \in (0,1]$ denotes a stepsize parameter, $w_{ij} \in [0,1]$ averaging weights and $\Delta_{ij}^{(t)} \in \mathbb{R}^d$ denotes a vector that is sent from node j to node i in iteration t. No communication is required when $w_{ij} = 0$. If we assume symmetry, $w_{ij} = w_{ji}$, the weights naturally define the communication graph $G = ([n], E)$ with edges $\{i,j\} \in E$ if $w_{ij} > 0$ and self-loops $\{i\} \in E$ for $i \in [n]$. The convergence rate of scheme (3) crucially depends on the connectivity matrix $W \in \mathbb{R}^{n\times n}$ of the network defined as $(W)_{ij} = w_{ij}$, also called the interaction or gossip matrix.

Definition 1 (Gossip matrix)**.** *We assume that $W \in [0,1]^{n\times n}$ is a symmetric ($W = W^\top$) doubly stochastic ($W\mathbf{1} = \mathbf{1}, \mathbf{1}^\top W = \mathbf{1}^\top$) matrix with eigenvalues $1 = |\lambda_1(W)| > |\lambda_2(W)| \geq \cdots \geq |\lambda_n(W)|$ and spectral gap*

$$\rho := 1 - |\lambda_2(W)| \in (0,1]\,. \tag{4}$$

It will also be convenient to define

$$\beta := \left\|I - W\right\|_2 \in [0,2]\,. \tag{5}$$

Table 1 depicts values of the spectral gap for important network topologies (with uniform averaging between the nodes). For the special case of uniform averaging on connected graphs it holds $\rho > 0$ (see e.g. (Xiao & Boyd, 2004)).

graph/topology	ρ^{-1}	node degree
ring	$\mathcal{O}(n^2)$	2
2d-torus	$\mathcal{O}(n)$	4
fully connected	$\mathcal{O}(1)$	$n-1$

Table 1. Spectral gap ρ for some important network topologies on n nodes (see e.g. (Aldous & Fill, 2002, p. 169)) for uniformly averaging W, i.e. $w_{ij} = \frac{1}{deg(i)} = \frac{1}{deg(j)}$ for $\{i,j\} \in E$.

3.2. Gossip with Exact Communication

For a fixed gossip matrix W, the classical algorithm analyzed in (Xiao & Boyd, 2004) corresponds to the choice

$$\gamma := 1, \qquad \Delta_{ij}^{(t)} := \mathbf{x}_j^{(t)} - \mathbf{x}_i^{(t)}, \tag{E-G}$$

in (3), with (E-G) standing for *exact gossip*. This scheme can also conveniently be written in matrix notation as

$$X^{(t+1)} := X^{(t)} + \gamma X^{(t)}(W - I)\,, \tag{6}$$

for iterates $X^{(t)} := [\mathbf{x}_1^{(t)}, \dots, \mathbf{x}_n^{(t)}] \in \mathbb{R}^{d\times n}$.

Theorem 1. *Let $\gamma \in (0, 1]$ and ρ be the spectral gap of W. Then the iterates of (E-G) converge linearly to the average $\overline{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^{(0)}$ with the rate*

$$\sum_{i=1}^n \left\|\mathbf{x}_i^{(t)} - \overline{\mathbf{x}}\right\|^2 \le (1-\gamma\rho)^{2t} \sum_{i=1}^n \left\|\mathbf{x}_i^{(0)} - \overline{\mathbf{x}}\right\|^2.$$

For $\gamma = 1$ the result corresponds to (Xiao & Boyd, 2004), here we slightly extend the analysis for arbitrary stepsizes. The short proof shows the elegance of the matrix notation (that we will later also adapt for the proofs that will follow).

Proof for $\gamma = 1$. Let $\overline{X} := [\overline{\mathbf{x}}, \dots, \overline{\mathbf{x}}] \in \mathbb{R}^{d\times n}$. Then for $\gamma = 1$ the theorem follows from the observation

$$\begin{aligned}
\left\|X^{(t+1)} - \overline{X}\right\|_F^2 &\overset{(6)}{=} \left\|(X^{(t)} - \overline{X})W\right\|_F^2 \\
&= \left\|(X^{(t)} - \overline{X})(W - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top)\right\|_F^2 \\
&\le \left\|W - \tfrac{1}{n}\mathbf{1}\mathbf{1}^\top\right\|_2^2 \left\|X^{(t)} - \overline{X}\right\|_F^2 \\
&= (1-\rho)^2 \left\|X^{(t)} - \overline{X}\right\|_F^2.
\end{aligned}$$

Here on the second line we used the crucial identity $X^{(t)}(\frac{1}{n}\mathbf{1}\mathbf{1}^\top) = \overline{X}$, i.e. the algorithm preserves the average over all iterations. This can be seen from (6):

$$X^{(t+1)}(\tfrac{1}{n}\mathbf{1}\mathbf{1}^\top) = X^{(t)}W(\tfrac{1}{n}\mathbf{1}\mathbf{1}^\top) = X^{(t)}(\tfrac{1}{n}\mathbf{1}\mathbf{1}^\top) = \overline{X}\,,$$

by Definition 1. The proof for arbitrary γ follows the same lines and is given in the appendix. □

3.3. Gossip with Quantized Communication

In every round of scheme (E-G) a full dimensional vector $\mathbf{g} \in \mathbb{R}^d$ is exchanged between two neighboring nodes for every link on the communication graph (node j sends $\mathbf{g} = \mathbf{x}_j^{(t)}$ to all its neighbors i, $\{i,j\} \in E$). A natural way to reduce the communication is to compress $\mathbf{g}$ before sending it, denoted as $Q(\mathbf{g})$, for a (potentially random) compression $Q\colon \mathbb{R}^d \to \mathbb{R}^d$. Informally, we can think of Q as either a sparsification operator (that enforces sparsity of $Q(\mathbf{g})$) or a quantization operator that reduces the number of bits required to represent $Q(\mathbf{g})$. For instance random rounding to less precise floating point numbers or to integers.

Aysal et al. (2008) propose the quantized gossip (Q1-G),

$$\gamma := 1, \qquad \Delta_{ij}^{(t)} := Q(\mathbf{x}_j^{(t)}) - \mathbf{x}_i^{(t)}\,, \tag{Q1-G}$$

in scheme (3), i.e. to apply the compression operator directly on the message that is send out from node j to node i. However, this algorithm does not preserve the average of the iterates over the iterations, $\frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^{(0)} \neq \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^{(t)}$ for $t \ge 1$, and as a consequence does not converge to the optimal solution $\overline{\mathbf{x}}$ of (2).

An alternative proposal by Carli et al. (2007) alleviates this drawback. The scheme

$$\gamma := 1, \qquad \Delta_{ij}^{(t)} := Q(\mathbf{x}_j^{(t)}) - Q(\mathbf{x}_i^{(t)})\,, \tag{Q2-G}$$

preserves the average of the iterates over the iterations. However, the scheme also fails to converge for arbitrary precision. If $\overline{\mathbf{x}} \neq \mathbf{0}$, the noise introduced by the compression, $\left\|Q(\mathbf{x}_j^{(t)})\right\|$, does not vanish for $t \to \infty$. As a consequence, the iterates oscillate around $\overline{\mathbf{x}}$ when compression error becomes larger than the suboptimality $\left\|\mathbf{x}_i^{(t)} - \overline{\mathbf{x}}\right\|$.

Both these schemes have been theoretically studied in (Carli et al., 2010b) under assumption of unbiasedness, i.e. assuming $\mathbb{E}_Q\, Q(\mathbf{x}) = \mathbf{x}$ for all $\mathbf{x} \in \mathbb{R}^d$. We will later adopt this theoretically understood setting in our experiments.

3.4. Proposed Method for Compressed Communication

We propose the novel compressed gossip scheme CHOCO-GOSSIP that supports a much larger class of compression operators, beyond unbiased quantization as for the schemes above. The algorithm can be summarized as

$$\begin{aligned}
\Delta_{ij}^{(t)} &:= \hat{\mathbf{x}}_j^{(t)} - \hat{\mathbf{x}}_i^{(t)}\,, \\
\hat{\mathbf{x}}_j^{(t+1)} &:= \hat{\mathbf{x}}_j^{(t)} + Q(\mathbf{x}_j^{(t+1)} - \hat{\mathbf{x}}_j^{(t)})\,,
\end{aligned} \tag{CHOCO-G}$$

for a stepsize $\gamma < 1$ depending on the specific compression operator Q (this will be detailed below). Here $\hat{\mathbf{x}}_i^{(t)} \in \mathbb{R}^d$ denote additional variables that are stored by all neighbors j of node i, $\{i,j\} \in E$, as well as on node i itself.

We will show in Theorem 2 below that this scheme (i) preserves the averages of the iterates $\mathbf{x}_i^{(t)}$, $i \in [n]$ over the iterations $t \ge 0$. Moreover, (ii) the noise introduced by the compression operator vanishes as $t \to \infty$. Precisely, we will show that $(\mathbf{x}_i^{(t)}, \hat{\mathbf{x}}_i^{(t)}) \to (\overline{\mathbf{x}}, \overline{\mathbf{x}})$ for $t \to \infty$ for every $i \in [n]$. Consequently, the argument for Q in (CHOCO-G) goes to zero, and the noise introduced by Q can be controlled.

The proposed scheme is summarized in Algorithm 1. Every worker $i \in [n]$ stores and updates its own local variable $\mathbf{x}_i$ as well as the variables $\hat{\mathbf{x}}_j$ for all neighbors (including itself) $j : \{i,j\} \in E$. This seems to require each machine to store $deg(i) + 2$ vectors. This is not necessary and the algorithm

Algorithm 1 CHOCO-GOSSIP

input : Initial values $\mathbf{x}_i^{(0)} \in \mathbb{R}^d$ on each node $i \in [n]$, stepsize γ, communication graph $G = ([n], E)$ and mixing matrix W, initialize $\hat{\mathbf{x}}_i^{(0)} := \mathbf{0}\ \forall i$

1: **for** t **in** $0 \dots T-1$ **do** *in parallel for all workers* $i \in [n]$
2: $\quad \mathbf{x}_i^{(t+1)} := \mathbf{x}_i^{(t)} + \gamma \sum_{j:\{i,j\}\in E} w_{ij}\big(\hat{\mathbf{x}}_j^{(t)} - \hat{\mathbf{x}}_i^{(t)}\big)$
3: $\quad \mathbf{q}_i^{(t)} := Q(\mathbf{x}_i^{(t+1)} - \hat{\mathbf{x}}_i^{(t)})$
4: $\quad$ **for** neighbors $j\colon \{i,j\} \in E$ (including $\{i\} \in E$) **do**
5: $\quad\quad$ Send $\mathbf{q}_i^{(t)}$ and receive $\mathbf{q}_j^{(t)}$
6: $\quad\quad \hat{\mathbf{x}}_j^{(t+1)} := \hat{\mathbf{x}}_j^{(t)} + \mathbf{q}_j^{(t)}$
7: $\quad$ **end for**
8: **end for**

could be re-written in a way that every node stores only *three* vectors: $\mathbf{x}_i$, $\hat{\mathbf{x}}_i$ and $\mathbf{s}_i = \sum_{j:\{i,j\}\in E} w_{ij}\hat{\mathbf{x}}_j$. For simplicity, we omit this modification here and refer to Appendix E for the exact form of the memory-efficient algorithm.

3.5. Convergence Analysis for CHOCO-GOSSIP

We analyze Algorithm 1 under the following general quality notion for the compression operator Q.

Assumption 1 (Compression operator)**.** *We assume that the compression operator $Q\colon \mathbb{R}^d \to \mathbb{R}^d$ satisfies*

$$\mathbb{E}_Q \left\|Q(\mathbf{x}) - \mathbf{x}\right\|^2 \leq (1-\delta)\left\|\mathbf{x}\right\|^2, \quad \forall \mathbf{x} \in \mathbb{R}^d, \tag{7}$$

for a parameter $\delta > 0$. Here $\mathbb{E}_Q$ denotes the expectation over the internal randomness of operator Q.

Example operators that satisfy (7) include

- *sparsification*: Randomly selecting k out of d coordinates (rand_k), or the k coordinates with highest magnitude values (top_k) give $\delta = \frac{k}{d}$ (Stich et al., 2018, Lemma A.1).
- *randomized gossip*: Setting $Q(\mathbf{x}) = \mathbf{x}$ with probability $p \in (0,1]$ and $Q(\mathbf{x}) = \mathbf{0}$ otherwise, gives $\delta = p$.
- *rescaled unbiased estimators*: suppose $\mathbb{E}_Q\, Q(\mathbf{x}) = \mathbf{x}$, $\forall \mathbf{x} \in \mathbb{R}^d$ and $\mathbb{E}_Q \left\|Q(\mathbf{x})\right\|^2 \leq \tau \left\|\mathbf{x}\right\|^2$, then $Q'(\mathbf{x}) := \frac{1}{\tau}Q(\mathbf{x})$ satisfies (7) with $\delta = \frac{1}{\tau}$.
- *random quantization*: For precision (levels) $s \in \mathbb{N}_+$, and $\tau = (1 + \min\{d/s^2, \sqrt{d}/s\})$ the quantization operator

$$\mathrm{qsgd}_s(x) = \frac{\mathrm{sign}(\mathbf{x}) \cdot \|\mathbf{x}\|}{s\tau} \cdot \left\lfloor s\frac{|\mathbf{x}|}{\|\mathbf{x}\|} + \xi \right\rfloor,$$

for random variable $\xi \sim_{\text{u.a.r.}} [0,1]^d$ satisfies (7) with $\delta = \frac{1}{\tau}$ (Alistarh et al., 2017, Lemma 3.1).

Theorem 2. CHOCO-GOSSIP *(Algorithm 1) converges linearly for average consensus:*

$$e_t \leq \left(1 - \frac{\rho^2\delta}{82}\right)^t e_0\,,$$

when using the stepsize $\gamma := \frac{\rho^2\delta}{16\rho + \rho^2 + 4\beta^2 + 2\rho\beta^2 - 8\rho\delta}$, *where δ is the compression factor as in Assumption 1, and* $e_t = \mathbb{E}_Q \sum_{i=1}^n \left(\left\|\mathbf{x}_i^{(t)} - \overline{\mathbf{x}}\right\|^2 + \left\|\mathbf{x}_i^{(t)} - \hat{\mathbf{x}}_i^{(t)}\right\|^2\right)$.

For the proof we refer to the appendix. For the exact communication case $\delta = 1$ we recover the rate from Theorem 1 for stepsize $\gamma < 1$ up to constant factors (which seems to be a small artifact of our proof technique). The theorem shows convergence for arbitrary $\delta > 0$, showing the superiority of scheme (CHOCO-G) over (Q1-G) and (Q2-G).

4. Decentralized Stochastic Optimization

We now leverage our proposed average consensus Alg. 1 to achieve consensus among the compute nodes in a decentralized optimization setting with communication restrictions.

In the decentralized optimization setting (1), not only does every node have a different local objective f_i, but we also allow each f_i to have stochastic optimization (or sum) structure, that is

$$f_i(\mathbf{x}) := \mathbb{E}_{\xi_i \sim \mathcal{D}_i}\, F_i(\mathbf{x}, \xi_i)\ , \tag{8}$$

for a loss function $F_i\colon \mathbb{R}^d \times \Omega \to \mathbb{R}$ and distributions $\mathcal{D}_1, \dots, \mathcal{D}_n$ which can be different on every node. Our framework therefore covers both stochastic optimization (e.g. when all $\mathcal{D}_i$ are identical) and empirical risk minimization when the $\mathcal{D}_i$'s are discrete with disjoint support.

4.1. Proposed Scheme for Decentralized Optimization

Our proposed method CHOCO-SGD—Communication-Compressed Decentralized SGD—is stated in Algorithm 2.

Algorithm 2 CHOCO-SGD

input : Initial values $\mathbf{x}_i^{(0)} \in \mathbb{R}^d$ on each node $i \in [n]$, consensus stepsize γ, SGD stepsizes $\{\eta_t\}_{t\geq 0}$, communication graph $G = ([n], E)$ and mixing matrix W, initialize $\hat{\mathbf{x}}_i^{(0)} := \mathbf{0}\ \forall i$

1: **for** t **in** $0 \dots T-1$ **do** *in parallel for all workers* $i \in [n]$
2: $\quad$ Sample $\xi_i^{(t)}$, compute gradient $\mathbf{g}_i^{(t)} := \nabla F_i(\mathbf{x}_i^{(t)}, \xi_i^{(t)})$
3: $\quad \mathbf{x}_i^{(t+\frac{1}{2})} := \mathbf{x}_i^{(t)} - \eta_t \mathbf{g}_i^{(t)}$
4: $\quad \mathbf{x}_i^{(t+1)} := \mathbf{x}_i^{(t+\frac{1}{2})} + \gamma \sum_{j:\{i,j\}\in E} w_{ij}\big(\hat{\mathbf{x}}_j^{(t)} - \hat{\mathbf{x}}_i^{(t)}\big)$
5: $\quad \mathbf{q}_i^{(t)} := Q(\mathbf{x}_i^{(t+1)} - \hat{\mathbf{x}}_i^{(t)})$
6: $\quad$ **for** neighbors $j\colon \{i,j\} \in E$ (including $\{i\} \in E$) **do**
7: $\quad\quad$ Send $\mathbf{q}_i^{(t)}$ and receive $\mathbf{q}_j^{(t)}$
8: $\quad\quad \hat{\mathbf{x}}_j^{(t+1)} := \mathbf{q}_j^{(t)} + \hat{\mathbf{x}}_j^{(t)}$
9: $\quad$ **end for**
10: **end for**

The algorithm consists of four parts. The stochastic gradient step in line 3, iterate update in step 4, application of the compression operator in step 5, followed by the (CHOCO-G) local communication in lines 6–9.

Remark 3. *As a special case for $\delta = 1$ and consensus stepsize $\gamma = 1$,* CHOCO-SGD *(Algorithm 2) recovers the following standard variant of decentralized SGD with gossip (similar e.g. to (Sirb & Ye, 2016; Lian et al., 2017)), given for illustration in Algorithm 3.*

Algorithm 3 PLAIN DECENTRALIZED SGD

for t **in** $0 \dots T-1$ **do** *in parallel for all workers* $i \in [n]$
  Sample $\xi_i^{(t)}$, compute gradient $\mathbf{g}_i^{(t)} := \nabla F_i(\mathbf{x}_i^{(t)}, \xi_i^{(t)})$
  $\mathbf{x}_i^{(t+\frac{1}{2})} := \mathbf{x}_i^{(t)} - \eta_t \mathbf{g}_i^{(t)}$
  $\mathbf{x}_i^{(t+1)} := \sum_{i=1}^n w_{ij} \mathbf{x}_j^{(t+\frac{1}{2})}$
end for

4.2. Convergence Analysis for CHOCO-SGD

Assumption 2. *We assume that each function $f_i \colon \mathbb{R}^d \to \mathbb{R}$ for $i \in [n]$ is L-smooth and μ-strongly convex and that the variance on each worker is bounded*

$$\mathbb{E}_{\xi_i} \|\nabla F_i(\mathbf{x}, \xi_i) - \nabla f_i(\mathbf{x})\|^2 \leq \sigma_i^2 \,, \quad \forall \mathbf{x} \in \mathbb{R}^d, i \in [n],$$
$$\mathbb{E}_{\xi_i} \|\nabla F_i(\mathbf{x}, \xi_i)\|^2 \leq G^2 \,, \quad \forall \mathbf{x} \in \mathbb{R}^d, i \in [n],$$

where $\mathbb{E}_{\xi_i}[\cdot]$ denotes the expectation over $\xi_i \sim \mathcal{D}_i$. It will be also convenient to denote

$$\overline{\sigma}^2 := \frac{1}{n} \sum_{i=1}^n \sigma_i^2.$$

For the (standard) definitions of smoothness and strong convexity we refer to Appendix A.1. The assumptions above could be relaxed to only hold for $\mathbf{x} \in \{\mathbf{x}_i^{(t)}\}_{t=1}^T$, the set of iterates of Algorithm 2.

Theorem 4. *Under Assumption 2, Algorithm 2 with SGD stepsizes $\eta_t := \frac{4}{\mu(a+t)}$ for parameter $a \geq \max\left\{\frac{410}{\rho^2\delta}, 16\kappa\right\}$ for condition number $\kappa = \frac{L}{\mu}$ and consensus stepsize $\gamma := \gamma(\rho, \delta)$ chosen as in Theorem 2, converges with the rate*

$$\mathbb{E}\,\Upsilon^{(T)} = \mathcal{O}\left(\frac{\overline{\sigma}^2}{\mu n T}\right) + \mathcal{O}\left(\frac{\kappa G^2}{\mu \delta^2 \rho^4 T^2}\right) + \mathcal{O}\left(\frac{G^2}{\mu \delta^3 \rho^6 T^3}\right),$$

where $\Upsilon^{(T)} := f(\mathbf{x}_{avg}^{(T)}) - f^\star$ for an averaged iterate $\mathbf{x}_{avg}^{(T)} = \frac{1}{S_T} \sum_{t=0}^{T-1} w_t \overline{\mathbf{x}}^{(t)}$ with weights $w_t = (a+t)^2$, and $S_T = \sum_{t=0}^{T-1} w_t$. As reminder, ρ denotes the eigengap of W, and δ the compression ratio.

For the proof we refer to the appendix. When T and $\overline{\sigma}$ are sufficiently large, the second two terms become negligible compared to $\mathcal{O}(\frac{\overline{\sigma}^2}{\mu n T})$—and we recover the convergence rate of of mini-batch SGD in the centralized setting and with exact communication. This is because topology (parameter ρ) and compression (parameter δ) only affect the higher-order terms in the rate. We also see that we obtain in this setting a $n\times$ speed up compared to the serial implementation of SGD on only one worker.

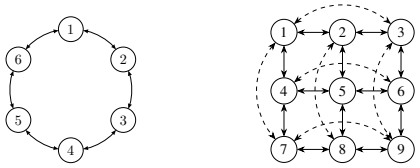

Figure 1. Ring topology (left) and Torus topology (right).

4.3. Distinction to Previous Baselines

Unlike the previous methods DCD-SGD and ECD-SGD from (Tang et al., 2018a), CHOCO-SGD converges under arbitrary high compression. As main difference to those schemes, CHOCO-SGD tries to carefully *compensate* quantization errors, while DCD- and ECD-SGD *ignore* them. In both, DCD- and ECD-SGD, the local variable on each worker is updated using only copies on the neighbors (in DCD copies and local variables are the same), which do not carry information about the true uncompressed values. Errors made previously are lost and cannot be corrected in later iterations. In CHOCO-SGD, the shared copy $\hat{\mathbf{x}}_i$ is in general different from the private copy $\mathbf{x}_i$, allowing to carry on the *true* values to the next iterations, and to compensate for errors made in previous quantization steps.

5. Experiments

We first compare CHOCO-GOSSIP to the gossip baselines from Sec. 5.2 and then compare the CHOCO-SGD to state of the art decentralized stochastic optimization schemes (that also support compressed communication) in Sec. 5.3.

5.1. Shared Experimental Setup

For our experiments we always report the *number of iterations* of the respective scheme, as well as *the number of transmitted bits*. These quantities are independent of systems architectures and network bandwidth.

Datasets. We rely on the *epsilon* (Sonnenburg et al., 2008) and *rcv1* (Lewis et al., 2004) datasets (cf. Table 2).

Compression operators. We use the (rand_k), (top_k) and (qsgd_s) compression operators introduced in Sec. 3.5, with k set to 1% of all coordinates and $s \in \{2^4, 2^8\}$. Details of counting number of bits are presented in Appendix F.2

In contrast to CHOCO-GOSSIP, the earlier schemes (Q1-G) and (Q2-G) were both analyzed for unbiased compression operators (Carli et al., 2010b). In order to reflect this theoretically understood setting we use the rescaled operators $(\frac{d}{k} \cdot \mathrm{rand}_k)$ and $(\tau \cdot \mathrm{qsgd}_s)$ in combination with those.

5.2. Average Consensus

We compare the performance of the gossip schemes (E-G) (exact communication), (Q1-G), (Q2-G) (both with unbi-

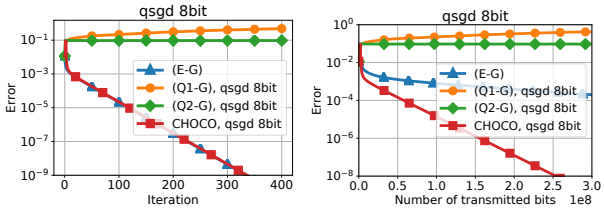

Figure 2. Average consensus on the ring topology with $n = 25$ nodes, $d = 2000$ and (qsgd_{256}) compression.

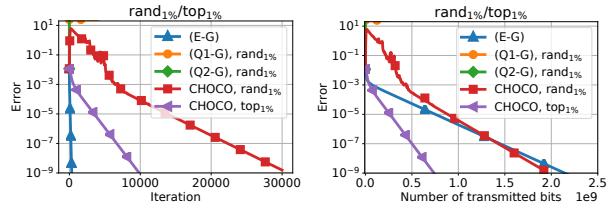

Figure 3. Average consensus on the ring topology with $n = 25$ nodes, $d = 2000$ and $(\mathrm{rand}_{1\%})$ and $(\mathrm{top}_{1\%})$ compression.

ased compression), and our scheme (CHOCO-G) in Figure 2 for the (qsgd_{256}) compression scheme and in Figure 3 for the random $(\mathrm{rand}_{1\%})$ compression scheme. In addition, we also depict the performance of CHOCO-GOSSIP with biased $(\mathrm{top}_{1\%})$ compression. We use ring topology with uniformly averaging mixing matrix W as in Figure 1, left. The stepsizes γ that were used for CHOCO-GOSSIP are listed in Table 3. We consider here the consensus problem (2) with data $(\mathbf{x}_i + 1) \in \mathbb{R}^d$ on the i-machine with $\mathbf{x}_i$ being the i-th vector in the *epsilon* dataset. The shift was added to move the average away from 0, as some of the schemes are biased towards this special output. We depict the errors $\frac{1}{n}\sum_{i=1}^n \left\|\mathbf{x}_i^{(t)} - \overline{\mathbf{x}}\right\|^2$. For more details, plots of the full error e_t and additional experiments we refer to Appendix G.1.

The proposed scheme (CHOCO-G) with 8 bit quantization (qsgd_{256}) converges with the same rate as (E-G) that uses exact communications (Fig. 2, left), while it requires much less data to be transmitted (Fig. 2, right). The schemes (Q1-G) and (Q2-G) do not converge in both settings (Fig. 2, 3, right). (Q1-G) diverges because the quantization error is too large already after the first step.

With sparsified communication $(\mathrm{rand}_{1\%})$, i.e. transmitting only 1% of all the coordinates, the scheme (Q1-G) quickly zeros out all the coordinates (Fig. 3). CHOCO-GOSSIP proves to be more robust and converges. The observed rate matches with the theoretical findings, as we expect the scheme with factor $100\times$ compression to be $100\times$ slower than (E-G) without compression. In terms of total data transmitted, both schemes converge at approximately same speed (Fig. 3, right). We also see that $(\mathrm{rand}_{1\%})$ sparsification can give additional gains and comes out as the most data-efficient method in these experiments.

dataset	m	d	density
epsilon	400000	2000	100%
rcv1	20242	47236	0.15%

Table 2. Size (m, d) and density of the datasets.

experiment	γ
CHOCO, (qsgd_{256})	1
CHOCO, $(\mathrm{rand}_{1\%})$	0.011
CHOCO, $(\mathrm{top}_{1\%})$	0.046

Table 3. Tuned stepsizes γ for averaging in Figs. 2– 3.

	epsilon			*rcv1*		
algorithm	a	b	γ	a	b	γ
PLAIN	0.1	d	-	1	1	-
CHOCO, (qsgd_{16})	0.1	d	0.34	1	1	0.078
CHOCO, $(\mathrm{rand}_{1\%})$	0.1	d	0.01	1	1	0.016
CHOCO, $(\mathrm{top}_{1\%})$	0.1	d	0.04	1	1	0.04
DCD, $(\mathrm{rand}_{1\%})$	10^{-15}	d	-	10^{-10}	d	-
DCD, (qsgd_{16})	0.01	d	-	10^{-10}	d	-
ECD, $(\mathrm{rand}_{1\%})$	10^{-10}	d	-	10^{-10}	d	-
ECD, (qsgd_{16})	10^{-12}	d	-	10^{-10}	d	-

Table 4. SGD learning rates $\eta_t = \frac{ma}{t+b}$ and consensus learning rates γ used in the experiments in Figs. 5–6. The parameters where tuned separately for each algorithm, tuning details can be found in Appendix F.1. The ECD and DCD stepsizes are small because the algorithms were observed to diverge for larger choices.

5.3. Decentralized SGD

We assess the performance of CHOCO-SGD on logistic regression, defined as $\frac{1}{m}\sum_{j=1}^m \log(1 + \exp(-b_j \mathbf{a}_j^\top \mathbf{x})) + \frac{1}{2m}\|\mathbf{x}\|^2$, where $\mathbf{a}_j \in \mathbb{R}^d$ and $b_j \in \{-1, 1\}$ are the data samples and m denotes the number of samples in the dataset. We distribute the m data samples evenly among the n workers and consider two settings: (i) *randomly shuffled*, where datapoints are randomly assigned to workers, and the more difficult (ii) *sorted* setting, where each worker only gets data samples just from one class (with the possible exception of one worker that gets two labels assigned). Moreover, we try to make the setting as difficult as possible, meaning that e.g. on the ring topology the machines with the same label form two connected clusters. We repeat each experiment three times and depict the mean curve and the area corresponding to one standard deviation. We plot suboptimality, i.e. $f(\overline{\mathbf{x}}^{(t)}) - f^\star$ (obtained by the LogisticSGD optimizer from scikit-learn (Pedregosa et al., 2011)) versus number of iterations and the number of transmitted bits between workers, which is proportional to the actual running time if communication is a bottleneck.

Algorithms. As baselines we consider Alg. 3 with exact communication (denoted as 'plain') and the communication efficient state-of-the-art optimization schemes DCD-SGD and ECD-SGD recently proposed in (Tang et al., 2018a) (for unbiased quantization operators) and compare them to CHOCO-SGD. We use decaying stepsize $\eta_t = \frac{ma}{t+b}$ where the parameters a, b are individually tuned for each algorithm and compression scheme, with values given in Table 4. Consensus learning rates γ were tuned on the simpler problem separately from optimization (see appendix F.1 for details, Table 4 for final values).

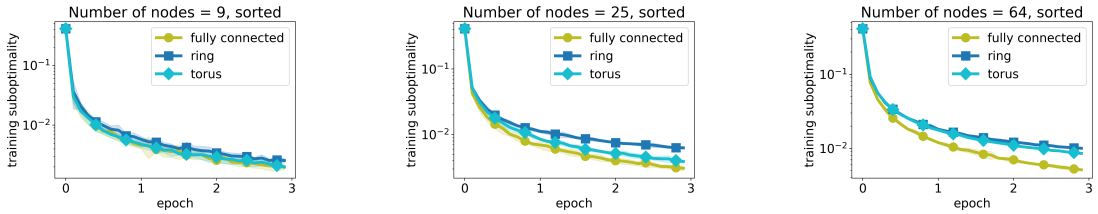

Figure 4. Performance of Algorithm 3 on ring, torus and fully connected topologies for $n \in \{9, 25, 64\}$ nodes. Here we consider the *sorted* setting, whilst the performance for randomly shuffled data is depicted in the Appendix G.

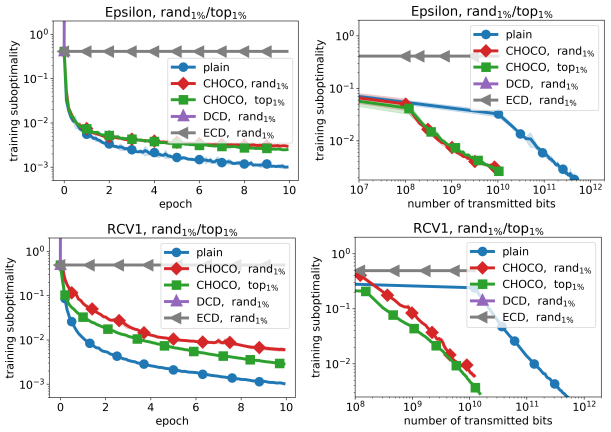

Figure 5. Comparison of Algorithm 3 (plain), ECD-SGD, DCD-SGD and CHOCO-SGD with $(\mathrm{rand}_{1\%})$ sparsification (in addition $(\mathrm{top}_{1\%})$ for CHOCO-SGD), for *epsilon* (top) and $rcv1$ (bottom) in terms of iterations (left) and communication cost (right), $n = 9$.

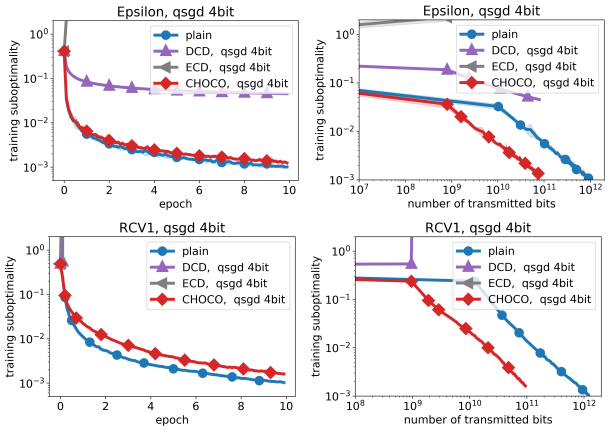

Figure 6. Comparison of Algorithm 3 (plain), ECD-SGD, DCD-SGD and CHOCO-SGD with (qsgd_{16}) quantization, for *epsilon* (top) and $rcv1$ (bottom) in terms of iterations (left) and communication cost (right), on $n = 9$ nodes on a ring topology.

Impact of Topology. In Figure 4 we show the performance of the baseline Algorithm 3 with exact communication on different topologies (ring, torus and fully-connected; Fig. 1) with uniformly averaging mixing matrix W. Note that Algorithm 3 for fully-connected graph corresponds to mini-batch SGD. Increasing the number of workers from $n = 9$ to $n = 25$ and $n = 64$ shows the mild effect of the network topology on the convergence. We observe that the *sorted* setting is more difficult than the *randomly shuffled* setting (see Fig. 11 in the Appendix G), where the convergence behavior remains almost unaffected. In the following we focus on the hardest case, i.e. the ring topology.

Comparison to Baselines. Figures 5 and 6 depict the performance of the algorithms on the ring topology with $n = 9$ nodes for *sorted* data of the *epsilon* and $rcv1$ datasets. CHOCO-SGD performs almost as good as the exact Alg. 3, but uses $100\times$ less communication with $(\mathrm{rand}_{1\%})$ sparsification (Fig. 5, right) and approximately $13\times$ less communication for (qsgd_{16}) quantization. The $(\mathrm{top}_{1\%})$ variant performs slightly better than $(\mathrm{rand}_{1\%})$ sparsification.

CHOCO-SGD consistently outperforms DCD-SGD in all settings. We also observed that DCD-SGD starts to perform better for larger number of levels s in the (qsgd_s) in the quantification operator (increasing communication cost). This is consistent with the reporting in (Tang et al., 2018a) that assumed high precision quantization. As a surprise to us, ECD-SGD, which was proposed in (Tang et al., 2018a) as the preferred alternative over DCD-SGD for less precise quantization operators, always performs worse than DCD-SGD, and often diverges.

Figures for *randomly shuffled* data can be found in the Appendix G. In that case CHOCO-SGD performs exactly as well as the exact Algorithm 3 in all situations.

Conclusion. The experiments verify our theoretical findings: CHOCO-GOSSIP is the first linearly convergent gossip algorithm with quantized communication and CHOCO-SGD consistently outperforms the baselines for decentralized optimization, reaching almost the same performance as exact communication, while significantly reducing communication cost. In view of the striking popularity of SGD as opposed to full-gradient methods for deep-learning, the application of CHOCO-SGD to decentralized deep learning—an instance of problem (1)—is a promising direction. We leave the analysis of CHOCO-SGD on non-convex function for future work. We believe that most of techniques presented here should carry over to the smooth non-convex setting as well.

Acknowledgements

We acknowledge funding from SNSF grant 200021_175796, as well as a Google Focused Research Award.

References

Aldous, D. and Fill, J. A. Reversible markov chains and random walks on graphs, 2002. Monograph, recompiled 2014.

Alistarh, D., Grubic, D., Li, J., Tomioka, R., and Vojnovic, M. QSGD: Communication-efficient SGD via gradient quantization and encoding. *NIPS - Advances in Neural Information Processing Systems 30*, pp. 1709–1720. 2017.

Alistarh, D., Hoefler, T., Johansson, M., Konstantinov, N., Khirirat, S., and Renggli, C. The convergence of sparsified gradient methods. *NeurIPS - Advances in Neural Information Processing Systems 31*, pp. 5977–5987. 2018.

Assran, M., Loizou, N., Ballas, N., and Rabbat, M. Stochastic Gradient Push for Distributed Deep Learning. *ICML*, 2019.

Aysal, T. C., Coates, M. J., and Rabbat, M. G. Distributed average consensus with dithered quantization. *IEEE Transactions on Signal Processing*, 56(10):4905–4918, 2008.

Bottou, L. Large-scale machine learning with stochastic gradient descent. In Lechevallier, Y. and Saporta, G. (eds.), *Proceedings of COMPSTAT'2010*, pp. 177–186.

Boyd, S., Ghosh, A., Prabhakar, B., and Shah, D. Randomized gossip algorithms. *IEEE/ACM Trans. Netw.*, 14(SI):2508–2530, 2006.

Carli, R., Fagnani, F., Frasca, P., Taylor, T., and Zampieri, S. Average consensus on networks with transmission noise or quantization. In *2007 European Control Conference (ECC)*, pp. 1852–1857, 2007.

Carli, R., Bullo, F., and Zampieri, S. Quantized average consensus via dynamic coding/decoding schemes. *International Journal of Robust and Nonlinear Control*, 20:156–175, 2010a.

Carli, R., Frasca, P., Fagnani, F., and Zampieri, S. Gossip consensus algorithms via quantized communication. *Automatica*, 46: 70–80, 2010b.

Dekel, O., Gilad-Bachrach, R., Shamir, O., and Xiao, L. Optimal distributed online prediction using mini-batches. *J. Mach. Learn. Res.*, 13(1):165–202, 2012.

Doan, T. T., Theja Maguluri, S., and Romberg, J. Accelerating the Convergence Rates of Distributed Subgradient Methods with Adaptive Quantization. *arXiv e-prints*, art. arXiv:1810.13245, 2018.

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, 2012.

Fang, J. and Li, H. Distributed estimation of gauss - markov random fields with one-bit quantized data. *IEEE Signal Processing Letters*, 17(5):449–452, 2010.

Goodall, W. M. Television by pulse code modulation. *The Bell System Technical Journal*, 30(1):33–49, 1951.

He, L., Bian, A., and Jaggi, M. Cola: Decentralized linear learning. In *Advances in Neural Information Processing Systems 31*, pp. 4541–4551. 2018.

Iutzeler, F., Bianchi, P., Ciblat, P., and Hachem, W. Asynchronous distributed optimization using a randomized alternating direction method of multipliers. In *Proceedings of the 52nd IEEE Conference on Decision and Control, CDC 2013, Firenze, Italy*, pp. 3671–3676. IEEE, 2013.

Jakovetić, D., Xavier, J., and Moura, J. M. F. Fast distributed gradient methods. *IEEE Transactions on Automatic Control*, 59 (5):1131–1146, 2014.

Johansson, B., Rabi, M., and Johansson, M. A randomized incremental subgradient method for distributed optimization in networked systems. *SIAM Journal on Optimization*, 20(3): 1157–1170, 2010.

Kempe, D., Dobra, A., and Gehrke, J. Gossip-based computation of aggregate information. In *Proceedings of the 44th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '03, pp. 482–, Washington, DC, USA, 2003. IEEE Computer Society.

Konecny, J. and Richtárik, P. Randomized Distributed Mean Estimation: Accuracy vs. Communication. *Frontiers in Applied Mathematics and Statistics*, 4:1502, 2018.

Lan, G., Lee, S., and Zhou, Y. Communication-efficient algorithms for decentralized and stochastic optimization. *Mathematical Programming*, 2018.

Lewis, D. D., Yang, Y., Rose, T. G., and Li, F. Rcv1: A new benchmark collection for text categorization research. *J. Mach. Learn. Res.*, 5:361–397, 2004.

Li, T., Fu, M., Xie, L., and Zhang, J. Distributed consensus with limited communication data rate. *IEEE Transactions on Automatic Control*, 56(2):279–292, 2011.

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 *NIPS - Advances in Neural Information Processing Systems 30*, pp. 5330–5340. 2017.

Lin, Y., Han, S., Mao, H., Wang, Y., and Dally, B. Deep gradient compression: Reducing the communication bandwidth for distributed training. In *ICLR 2018 - International Conference on Learning Representations*, 2018.

McMahan, B., Moore, E., Ramage, D., Hampson, S., and Arcas, B. A. y. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *AISTATS 2017 - Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, pp. 1273–1282, 2017.

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

Nedić, A., Olshevsky, A., Ozdaglar, A., and Tsitsiklis, J. N. Distributed subgradient methods and quantization effects. In *Proceedings of the 47th IEEE Conference on Decision and Control, CDC 2008*, pp. 4177–4184, 2008.

Nedić, A., Lee, S., and Raginsky, M. Decentralized online optimization with global objectives and local communication. In *2015 American Control Conference (ACC)*, pp. 4497–4503, 2015.

Olfati-Saber, R. and Murray, R. M. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Rabbat, M. Multi-agent mirror descent for decentralized stochastic optimization. In *2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 517–520, 2015.

Rakhlin, A., Shamir, O., and Sridharan, K. Making gradient descent optimal for strongly convex stochastic optimization. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pp. 1571–1578, USA, 2012. Omnipress.

Reisizadeh, A., Mokhtari, A., Hassani, S. H., and Pedarsani, R. Quantized decentralized consensus optimization. *CoRR*, abs/1806.11536, 2018.

Robbins, H. and Monro, S. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951.

Roberts, L. Picture coding using pseudo-random noise. *IRE Transactions on Information Theory*, 8(2):145–154, February 1962. ISSN 0096-1000. doi: 10.1109/TIT.1962.1057702.

Scaman, K., Bach, F., Bubeck, S., Lee, Y. T., and Massoulié, L. Optimal algorithms for smooth and strongly convex distributed optimization in networks. In *ICML - Proceedings of the 34th 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 31*, pp. 2745–2754. 2018.

Seide, F., Fu, H., Droppo, J., Li, G., and Yu, D. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. In Li, H., Meng, H. M., Ma, B., Chng, E., and Xie, L. (eds.), *INTERSPEECH*, pp. 1058–1062. ISCA, 2014.

Shamir, O. and Srebro, N. Distributed stochastic optimization and learning. *2014 52nd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 850–857, 2014.

Sirb, B. and Ye, X. Consensus optimization with delayed and stochastic gradients on decentralized networks. In *2016 IEEE International Conference on Big Data (Big Data)*, pp. 76–85, 2016.

Sonnenburg, S., Franc, V., Yom-Tov, E., and Sebag, M. Pascal large scale learning challenge. *25th International Conference on Machine Learning (ICML2008) Workshop. J. Mach. Learn. Res*, 10:1937–1953, 01 2008.

Stich, S. U. Local SGD Converges Fast and Communicates Little. *arXiv e-prints*, art. arXiv:1805.09767, May 2018.

Stich, S. U., Cordonnier, J.-B., and Jaggi, M. Sparsified SGD with memory. In *NeurIPS - Advances in Neural Information Processing Systems 31*, pp. 4452–4463. 2018.

Tang, H., Gan, S., Zhang, C., Zhang, T., and Liu, J. Communication compression for decentralized training. In *Advances in Neural Information Processing Systems 31*, pp. 7663–7673. 2018a.

Tang, H., Lian, X., Yan, M., Zhang, C., and Liu, J. d^2: Decentralized training over decentralized data. In *ICML - Proceedings of the 35th International Conference on Machine Learning*, pp. 4848–4856, 2018b.

Thanou, D., Kokiopoulou, E., Pu, Y., and Frossard, P. Distributed average consensus with quantization refinement. *IEEE Transactions on Signal Processing*, 61(1):194–205, 2013.

Tsitsiklis, J. N. *Problems in decentralized decision making and computation*. PhD thesis, Massachusetts Institute of Technology, 1984.

Uribe, C. A., Lee, S., and Gasnikov, A. A Dual Approach for Optimal Algorithms in Distributed Optimization over Networks. *arXiv*, September 2018.

Wangni, J., Wang, J., Liu, J., and Zhang, T. Gradient sparsification for communication-efficient distributed optimization. In *NeurIPS - Advances in Neural Information Processing Systems 31*, pp. 1306–1316. 2018.

Wei, E. and Ozdaglar, A. Distributed alternating direction method of multipliers. In *2012 IEEE 51st IEEE Conference on Decision and Control (CDC)*, pp. 5445–5450, 2012.

Wen, W., Xu, C., Yan, F., Wu, C., Wang, Y., Chen, Y., and Li, H. Terngrad: Ternary gradients to reduce communication in distributed deep learning. In *NIPS - Advances in Neural Information Processing Systems 30*, pp. 1509–1519. 2017.

Xiao, L. and Boyd, S. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

Xiao, L., Boyd, S., and Lall, S. A scheme for robust distributed sensor fusion based on average consensus. In *IPSN 2005. Fourth International Symposium on Information Processing in Sensor Networks, 2005.*, pp. 63–70, 2005.

Yu, C., Tang, H., Renggli, C., Kassing, S., Singla, A., Alistarh, D., Zhang, C., and Liu, J. Distributed Learning over Unreliable Networks. *ICML 2019*, 2019.

Yuan, D., Xu, S., Zhao, H., and Rong, L. Distributed dual averaging method for multi-agent optimization with quantized communication. *Systems & Control Letters*, 61(11):1053 – 1061, 2012.

Zhang, H., Li, J., Kara, K., Alistarh, D., Liu, J., and Zhang, C. ZipML: Training linear models with end-to-end low precision, and a little bit of deep learning. In *ICML - Proceedings of the 34th International Conference on Machine Learning*, pp. 4035–4043, 2017.

A. Basic Identities and Inequalities

A.1. Smooth and Strongly Convex Functions

Definition 2. *A differentiable function $f\colon \mathbb{R}^d \to \mathbb{R}$ is L-smooth for parameter $L \geq 0$ if*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2} \|\mathbf{y} - \mathbf{x}\|^2 , \qquad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d . \tag{9}$$

Definition 3. *A differentiable function $f\colon \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex for parameter $\mu \geq 0$ if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2} \|\mathbf{y} - \mathbf{x}\|^2 , \qquad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^d . \tag{10}$$

Remark 5. *If f is L-smooth with minimizer $\mathbf{x}^\star$ s.t $\nabla f(\mathbf{x}^\star) = \mathbf{0}$, then*

$$\|\nabla f(\mathbf{x})\|^2 = \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{x}^\star)\|^2 \leq 2L \left(f(\mathbf{x}) - f(\mathbf{x}^\star)\right) . \tag{11}$$

A.2. Vector and Matrix Inequalities

Remark 6. *For $A \in \mathbb{R}^{d \times n}$, $B \in \mathbb{R}^{n \times n}$*

$$\|AB\|_F \leq \|A\|_F \|B\|_2 . \tag{12}$$

Remark 7. *For arbitrary set of n vectors $\{\mathbf{a}_i\}_{i=1}^n$, $\mathbf{a}_i \in \mathbb{R}^d$*

$$\left\| \sum_{i=1}^n \mathbf{a}_i \right\|^2 \leq n \sum_{i=1}^n \|\mathbf{a}_i\|^2 . \tag{13}$$

Remark 8. *For given two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$*

$$2 \langle \mathbf{a}, \mathbf{b} \rangle \leq \gamma \|\mathbf{a}\|^2 + \gamma^{-1} \|\mathbf{b}\|^2 , \qquad \forall \gamma > 0 . \tag{14}$$

Remark 9. *For given two vectors $\mathbf{a}, \mathbf{b} \in \mathbb{R}^d$*

$$\|\mathbf{a} + \mathbf{b}\|^2 \leq (1 + \alpha) \|\mathbf{a}\|^2 + (1 + \alpha^{-1}) \|\mathbf{b}\|^2 , \qquad \forall \alpha > 0 . \tag{15}$$

This inequality also holds for the sum of two matrices $A, B \in \mathbb{R}^{n \times d}$ in Frobenius norm.

A.3. Implications of the bounded gradient and bounded variance assumption

Remark 10. *If $F_i : \mathbb{R}^d \times \Omega \to \mathbb{R}, i = 1, \ldots, n$ are convex functions with $\mathbb{E}_\xi \|\nabla F_i(\mathbf{x}, \xi)\|^2 \leq G^2$, $\partial F(X, \xi) = [\nabla F_1(\mathbf{x}, \xi_1), \ldots, \nabla F_n(\mathbf{x}, \xi_n)]$*

$$\mathbb{E}_{\xi_1, \ldots, \xi_n} \|\partial F(X, \xi)\|_F^2 \leq nG^2 , \qquad \forall X .$$

Remark 11 (Mini-batch variance)**.** *If for functions f_i, F_i defined in (8) $\mathbb{E}_\xi \|\nabla F_i(\mathbf{x}, \xi) - \nabla f_i(\mathbf{x})\|^2 \leq \sigma_i^2, i \in [n]$, then*

$$\mathbb{E}_{\xi_1^{(t)}, \ldots, \xi_n^{(t)}} \left\| \frac{1}{n} \sum_{j=1}^n \left(\nabla f_j(\mathbf{x}_j^{(t)}) - \nabla F_j(\mathbf{x}_j^{(t)}, \xi_j^{(t)}) \right) \right\|^2 \leq \frac{\overline{\sigma}^2}{n},$$

where $\overline{\sigma}^2 = \frac{\sum_{i=1}^n \sigma_i^2}{n}$.

Proof. This follows from

$$\mathbb{E} \left\| \frac{1}{n} \sum_{j=1}^n Y_j \right\|^2 = \frac{1}{n^2} \left(\sum_{j=1}^n \mathbb{E} \|Y_j\|^2 + \sum_{i \neq j} \mathbb{E} \langle Y_i, Y_j \rangle \right) = \frac{1}{n^2} \sum_{j=1}^n \mathbb{E} \|Y_j\|^2 \leq \frac{1}{n^2} \sum_{j=1}^n \sigma_j^2 = \frac{\overline{\sigma}^2}{n}$$

for $Y_j = f_j(\mathbf{x}_j^{(t)}) - \nabla F_j(\mathbf{x}_j^{(t)}, \xi_j^{(t)})$. Expectation of scalar product is equal to zero because ξ_i is independent of ξ_j since $i \neq j$. □

B. Consensus in Matrix notation

In the proofs in the next section we will use the matrix notation, as already introduced in the main text. We define

$$X^{(t)} := \left[\mathbf{x}_1^{(t)}, \dots, \mathbf{x}_n^{(t)}\right] \in \mathbb{R}^{d\times n}, \quad Q^{(t)} := \left[\mathbf{q}_1^{(t)}, \dots, \mathbf{q}_n^{(t)}\right] \in \mathbb{R}^{d\times n}, \quad \hat{X}^{(t)} := \left[\hat{\mathbf{x}}_1^{(t)}, \dots, \hat{\mathbf{x}}_n^{(t)}\right] \in \mathbb{R}^{d\times n}. \tag{16}$$

Then using matrix notation we can rewrite Algorithm 1 as

Algorithm 1 CHOCO-GOSSIP IN MATRIX NOTATION

input : $X^{(0)}$, γ, W.
1: Initialize: $\hat{X}^{(0)} = 0$
2: **for** t **in** $0 \dots T-1$ **do**
3: $\quad X^{(t+1)} = X^{(t)} + \gamma \hat{X}^{(t)} (W - I)$
4: $\quad Q^{(t)} = Q(X^{(t+1)} - \hat{X}^{(t)})$
5: $\quad \hat{X}^{(t+1)} = \hat{X}^{(t)} + Q^{(t)}$
6: **end for**

Remark 12. *Note that since every worker i for each neighbor $j : \{i,j\} \in E$ stores $\hat{\mathbf{x}}_j$, the proper notation for $\hat{\mathbf{x}}$ would be to use $\hat{\mathbf{x}}_{ij}$ instead. We simplified it using the property that if $\hat{\mathbf{x}}_{ij}^{(0)} = \hat{\mathbf{x}}_{kj}^{(0)}$, $\forall i,k : \{i,j\} \in E$ and $\{k,j\} \in E$, then they are equal at all timesteps $\hat{\mathbf{x}}_{ij}^{(t)} = \hat{\mathbf{x}}_{kj}^{(t)}$, $\forall t \geq 0$.*

Remark 13. *The results of Theorem 4 and 19 also hold for arbitrary initialized $\hat{X}^{(0)}$ with the constraint that $\forall j$ all the neighbors of the node j initialized with the same $\hat{\mathbf{x}}_i$, i.e. using extended notation $\hat{\mathbf{x}}_{ij}^{(0)} = \hat{\mathbf{x}}_{kj}^{(0)}$, $\forall i,k : \{i,j\} \in E$ and $\{k,j\} \in E$.*

B.1. Useful Facts

Remark 14. *Let $X^{(t)} = \left[\mathbf{x}_1^{(t)}, \dots, \mathbf{x}_n^{(t)}\right] \in \mathbb{R}^{d\times n}$ and $\overline{X}^{(t)} = \left[\overline{\mathbf{x}}^{(t)}, \dots, \overline{\mathbf{x}}^{(t)}\right] \in \mathbb{R}^{d\times n}$, for $\overline{\mathbf{x}}^{(t)} = \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^{(t)}$, then because W is doubly stochastic*

$$\overline{X}^{(t)} = X^{(t)} \frac{1}{n}\mathbf{1}\mathbf{1}^\top, \qquad \overline{X}^{(t)} W = \overline{X}^{(t)}. \tag{17}$$

Remark 15. *The average $\overline{X}^{(t)} = \left[\overline{\mathbf{x}}^{(t)}, \dots, \overline{\mathbf{x}}^{(t)}\right] \in \mathbb{R}^{d\times n}$ during iterates of the Algorithm 1 is preserved, i.e.*

$$\overline{X}^{(t)} = \overline{X}^{(0)}, \quad \forall t, \tag{18}$$

where $\overline{X}^{(t)} = \left[\overline{\mathbf{x}}^{(t)} \dots, \overline{\mathbf{x}}^{(t)}\right] \in \mathbb{R}^{d\times n}$.

Proof.

$$\overline{X}^{(t+1)} = \overline{X}^{(t)} + \gamma \hat{X}^{(t)} (W - I) \frac{\mathbf{1}\mathbf{1}^\top}{n} = \overline{X}^{(t)},$$

because $W \frac{\mathbf{1}\mathbf{1}^\top}{n} = \frac{\mathbf{1}\mathbf{1}^\top}{n}$ since W is doubly stochastic. □

Lemma 16. *For W satisfying Definition 1, i.e. W is symmetric doubly stochastic matrix with second largest eigenvalue $1 - \rho = |\lambda_2(W)| < 1$*

$$\left\| W^k - \frac{1}{n}\mathbf{1}\mathbf{1}^\top \right\|_2 \leq (1-\rho)^k. \tag{19}$$

Proof. Let $U\Lambda U^\top$ be SVD-decomposition of W, then $W^k = U\Lambda^k U^\top$ Because of the stochastic property of W its first eigenvector is $u_1 = \frac{1}{\sqrt{n}}\mathbf{1}$.

$$U \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \dots & & & \\ 0 & 0 & \dots & 0 \end{pmatrix} U^\top = u_1 u_1^\top = \frac{1}{n}\mathbf{1}\mathbf{1}^\top$$

Hence,

$$\left\|W^k - \frac{1}{n}\mathbf{1}\mathbf{1}^\top\right\|_2 = \left\|U\Lambda^k U^\top - U \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \dots & & & \\ 0 & 0 & \dots & 0 \end{pmatrix} U^\top\right\|_2 = \left\|\Lambda^k - \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \dots & & & \\ 0 & 0 & \dots & 0 \end{pmatrix}\right\|_2 = (1-\rho)^k. \qquad \square$$

C. Proof of Theorem 2—Convergence of CHOCO-GOSSIP

Lemma 17. *Let $X^{(t)}, \hat{X}^{(t)} \in \mathbb{R}^{d\times n}$, $\overline{X} = [\overline{\mathbf{x}}, \dots, \overline{\mathbf{x}}]$ for average $\overline{\mathbf{x}} = \frac{1}{n}X^{(t)}\mathbf{1} \in \mathbb{R}^d$ and let $X^{(t+1)} = X^{(t)} + \gamma\hat{X}^{(t)}(W - I) \in \mathbb{R}^{d\times n}$ be defined as in Algorithm 1 with stepsize $\gamma \geq 0$ and mixing matrix $W \in \mathbb{R}^{n\times n}$ as in Definition 1. Then*

$$\left\|X^{(t+1)} - \overline{X}\right\|_F^2 \leq (1-\rho\gamma)^2(1+\alpha_1)\left\|X^{(t)} - \overline{X}\right\|_F^2 + \gamma^2(1+\alpha_1^{-1})\beta^2\left\|\hat{X}^{(t)} - X^{(t)}\right\|_F^2, \ \forall\alpha_1 > 0\,.$$

Here $\alpha_1 > 0$ is a parameter whose value will be chosen later, $\rho = 1 - |\lambda_2(W)|$ and $\beta = \max_i\{1-\lambda_i(W)\}$ as defined above.

Proof. By the definition of $X^{(t+1)}$ and the observation $\overline{X}(W-I) = 0$ from Remark 14, we can write

$$\begin{aligned}
\left\|X^{(t+1)} - \overline{X}\right\|_F^2 &= \left\|X^{(t)} - \overline{X} + \gamma\hat{X}^{(t)}(W-I)\right\|_F^2 \\
&= \left\|X^{(t)} - \overline{X} + \gamma\left(X^{(t)} - \overline{X}\right)(W-I) + \gamma\left(\hat{X}^{(t)} - X^{(t)}\right)(W-I)\right\|_F^2 \\
&= \left\|\left(X^{(t)} - \overline{X}\right)((1-\gamma)I + \gamma W) + \gamma\left(\hat{X}^{(t)} - X^{(t)}\right)(W-I)\right\|_F^2 \\
&\overset{(15)}{\leq} (1+\alpha_1)\left\|\left(X^{(t)} - \overline{X}\right)((1-\gamma)I + \gamma W)\right\|_F^2 + (1+\alpha_1^{-1})\left\|\gamma\left(\hat{X}^{(t)} - X^{(t)}\right)(W-I)\right\|_F^2 \\
&\overset{(12)}{\leq} (1+\alpha_1)\left\|\left(X^{(t)} - \overline{X}\right)((1-\gamma)I + \gamma W)\right\|_F^2 + (1+\alpha_1^{-1})\gamma^2\left\|W-I\right\|_2^2 \cdot \left\|\hat{X}^{(t)} - X^{(t)}\right\|_F^2.
\end{aligned}$$

Let's estimate the first term

$$\begin{aligned}
\left\|\left(X^{(t)} - \overline{X}\right)((1-\gamma)I + \gamma W)\right\|_F &\leq (1-\gamma)\left\|X^{(t)} - \overline{X}\right\|_F + \gamma\left\|\left(X^{(t)} - \overline{X}\right)W\right\|_F \\
&\overset{(17)}{=} (1-\gamma)\left\|X^{(t)} - \overline{X}\right\|_F + \gamma\left\|\left(X^{(t)} - \overline{X}\right)(W - \mathbf{1}\mathbf{1}^\top/n)\right\|_F \\
&\overset{(19),(12)}{\leq} (1-\gamma\rho)\left\|X^{(t)} - \overline{X}\right\|_F
\end{aligned}$$

where we used $(X^{(t)} - \overline{X})\mathbf{1}\mathbf{1}^\top/n = 0$, by definition of $\overline{X}$, in the second line. Putting this together gives us the statement of the lemma. $\square$

Lemma 18. *Let $X^{(t)}, \hat{X}^{(t)} \in \mathbb{R}^{d\times n}$, $\overline{X} = [\overline{\mathbf{x}}, \dots, \overline{\mathbf{x}}]$ for average $\overline{\mathbf{x}} = \frac{1}{n}X^{(t)}\mathbf{1} \in \mathbb{R}^d$ and let $X^{(t+1)} \in \mathbb{R}^{d\times n}$ and $\hat{X}^{(t+1)} \in \mathbb{R}^{d\times n}$ be defined as in Algorithm 1 with stepsize $\gamma \geq 0$, mixing matrix $W \in \mathbb{R}^{n\times n}$ as in Definition 1 and quantization as in Assumption 1. Then*

$$\begin{aligned}
\mathbb{E}_Q\left\|X^{(t+1)} - \hat{X}^{(t+1)}\right\|_F^2 &\leq (1-\delta)(1+\gamma\beta)^2(1+\alpha_2)\left\|X^{(t)} - \hat{X}^{(t)}\right\|_F^2 \\
&\quad + (1-\delta)\gamma^2\beta^2(1+\alpha_2^{-1})\left\|X^{(t)} - \overline{X}\right\|_F^2, \qquad \forall\alpha_2 > 0\,.
\end{aligned}$$

Here $\alpha_2 > 0$ is a parameter whose value will be chosen later, $\beta = \max_i\{1-\lambda_i(W)\}$ as defined above and compression ratio $\delta > 0$.

Proof. By the definition of $X^{(t+1)}$ and $\hat{X}^{(t+1)}$ we can write

$$\begin{aligned}
\mathbb{E}_Q \left\|X^{(t+1)} - \hat{X}^{(t+1)}\right\|_F^2 &= \mathbb{E}_Q \left\|X^{(t+1)} - \hat{X}^{(t)} - Q(X^{(t+1)} - \hat{X}^{(t)})\right\|_F^2 \overset{(7)}{\leq} (1-\delta)\left\|X^{(t+1)} - \hat{X}^{(t)}\right\|_F^2 \\
&= (1-\delta)\left\|X^{(t)} + \gamma\hat{X}^{(t)}(W-I) - \hat{X}^{(t)}\right\|_F^2 \\
&\overset{(17)}{=} (1-\delta)\left\|\left(X^{(t)} - \hat{X}^{(t)}\right)((1+\gamma)I - \gamma W) + \gamma(W-I)\left(X^{(t)} - \overline{X}\right)\right\|_F^2 \\
&\overset{(15)}{\leq} (1-\delta)(1+\alpha_2)\left\|\left(X^{(t)} - \hat{X}^{(t)}\right)((1+\gamma)I - \gamma W)\right\|_F^2 \\
&\qquad + (1-\delta)(1+\alpha_2^{-1})\left\|\gamma(W-I)\left(X^{(t)} - \overline{X}\right)\right\|_F^2 \\
&\overset{(12)}{\leq} (1-\delta)(1+\gamma\beta)^2(1+\alpha_2)\left\|X^{(t)} - \hat{X}^{(t)}\right\|_F^2 + (1-\delta)\gamma^2\beta^2(1+\alpha_2^{-1})\left\|X^{(t)} - \overline{X}\right\|_F^2 ,
\end{aligned}$$

where we used $\|I + \gamma(I-W)\|_2 = 1 + \gamma\|I-W\|_2 = 1+\gamma\beta$ because eigenvalues of $\gamma(I-W)$ are positive. □

Proof of Theorem 2. As observed in Remark 14 the averages of the iterates is preserved, i.e. $\overline{X} \equiv X^{(t)}\frac{1}{n}\mathbf{1}\mathbf{1}^\top$ for all $t \geq 0$. By applying the Lemmas 17 and 18 from above we obtain

$$\mathbb{E}_Q\, e_{t+1} \leq \eta_1(\gamma)\left\|X^{(t)} - \overline{X}\right\|_F^2 + \xi_1(\gamma)\left\|\hat{X}^{(t)} - X^{(t)}\right\|_F^2 \leq \max\{\eta_1(\gamma), \xi_1(\gamma)\}\cdot e_t\,,$$

where

$$\begin{aligned}
\eta_1(\gamma) &:= (1-\rho\gamma)^2(1+\alpha_1) + (1-\delta)\gamma^2\beta^2(1+\alpha_2^{-1})\,, \\
\xi_1(\gamma) &:= \gamma^2\beta^2(1+\alpha_1^{-1}) + (1-\delta)(1+\gamma\beta)^2(1+\alpha_2)\,.
\end{aligned}$$

Now, we need to choose the parameters α_1, α_2 and stepsize γ such as to minimize the factor $\max\{\eta_1(\gamma), \xi_1(\gamma)\}$. Whilst the optimal parameter settings can for instance be obtained using specialized optimization software, we here proceed by showing that for the (suboptimal) choice

$$\begin{aligned}
\alpha_1 &:= \frac{\gamma\rho}{2}, \\
\alpha_2 &:= \frac{\delta}{2} \\
\gamma^\star &:= \frac{\rho\delta}{16\rho + \rho^2 + 4\beta^2 + 2\rho\beta^2 - 8\rho\delta}
\end{aligned} \tag{20}$$

it holds

$$\max\{\eta_1(\gamma^\star), \xi_1(\gamma^\star)\} \leq 1 - \frac{\rho^2\delta}{2(16\rho + \rho^2 + 4\beta^2 + 2\rho\beta^2 - 8\rho\delta)}\,. \tag{21}$$

The claim of the theorem then follows by observing

$$1 - \frac{\rho^2\delta}{2(16\rho + \rho^2 + 4\beta^2 + 2\rho\beta^2 - 8\rho\delta)} \leq 1 - \frac{\rho^2\delta}{82}\,, \tag{22}$$

using the crude estimates $0 \leq \rho \leq 1$, $\beta \leq 2$, $\delta \geq 0$.

We now proceed to show that (21) holds. Observe that for α_1, α_2 as in (20),

$$\begin{aligned}
\eta_1(\gamma) &\leq (1-\gamma\rho)(1-\gamma\rho)\left(1 + \frac{\gamma\rho}{2}\right) + \gamma^2\beta^2(1-\delta)\left(1 + \frac{2}{\delta}\right) \\
&\leq \left(1 - \frac{\gamma\rho}{2}\right)^2 + \frac{2}{\delta}\gamma^2\beta^2 =: \eta_2(\gamma)\,,
\end{aligned}$$

where we used the inequality $(1-x)(1+\frac{x}{2}) \leq (1-\frac{x}{2})$ and $(1-\delta)(1+2/\delta) \leq \frac{2}{\delta}$ for $\delta > 0$. The quadratic function $\eta_2(\gamma)$ is minimized for $\gamma' = \frac{2\rho\delta}{8\beta^2+\rho^2\delta}$ with value $\eta_2(\gamma') = \frac{8\beta^2}{8\beta^2+\rho^2\delta} < 1$. Thus by Jensen's inequality

$$\eta_2(\lambda\gamma') \leq (1-\lambda)\eta_2(0) + \lambda\eta_2(\gamma') = 1 - \lambda\frac{\rho^2\delta}{8\beta^2+\rho^2\delta} \tag{23}$$

for $0 \leq \lambda \leq 1$, and especially for the choice $\lambda' = \frac{8\beta^2+\rho^2\delta}{2(16\rho+\rho^2+4\beta^2+2\rho\beta^2-8\rho\delta)}$ we have

$$\eta_1(\gamma^\star) \leq \eta_2(\lambda'\gamma') \overset{(23)}{\leq} 1 - \frac{\rho^2\delta}{2(16\rho+\rho^2+4\beta^2+2\rho\beta^2-8\rho\delta)}\,, \tag{24}$$

as $\gamma^\star = \lambda'\gamma'$. Now we proceed to estimate $\xi_1(\gamma^\star)$. Observe

$$\xi_1(\gamma) \leq \gamma^2\beta^2\left(1+\frac{2}{\gamma\rho}\right) + (1+\gamma\beta)^2(1-\delta)\left(1+\frac{\delta}{2}\right) \leq \gamma^2\beta^2\left(1+\frac{2}{\gamma\rho}\right) + (1+\gamma\beta)^2\left(1-\frac{\delta}{2}\right)\,, \tag{25}$$

again from $(1-x)(1+\frac{x}{2}) \leq (1-\frac{x}{2})$ for $x > 0$. As $\beta \leq 2$ we can estimate $(1+\gamma\beta)^2 \leq 1+8\gamma$ for any $0 \leq \gamma \leq 1$. Furthermore $\gamma^2 \leq \gamma$ for $0 \leq \gamma \leq 1$. Thus

$$\xi_1(\gamma^\star) \leq \beta^2\left(\gamma^\star + \frac{2\gamma^\star}{\rho}\right) + \left(1-\frac{\delta}{2}\right)(1+8\gamma^\star) = 1 - \frac{\rho^2\delta}{2(16\rho+\rho^2+4\beta^2+2\rho\beta^2-8\rho\delta)}\,, \tag{26}$$

as a quick calculation shows. □

D. Proof of Theorem 4—Convergence of CHOCO-SGD

Recall, that $\{\mathbf{x}_i^{(t)}\}_{t=0}^T$ denote the iterates of Algorithm 2 on worker $i \in [n]$. We define

$$\overline{\mathbf{x}}^{(t)} := \frac{1}{n}\sum_{i=1}^n \mathbf{x}_i^{(t)}\,, \tag{27}$$

the average over all workers. Note that this quantity is not available to the workers at any given time, but it will be conveniently to use for the proofs. In this section we use both vector and matrix notation whenever it is more convenient, and define

$$\begin{aligned} X^{(t)} &:= \left[\mathbf{x}_1^{(t)}, \dots, \mathbf{x}_n^{(t)}\right] \in \mathbb{R}^{d\times n}, \qquad \overline{X}^{(t)} := \left[\overline{\mathbf{x}}^{(t)}, \dots, \overline{\mathbf{x}}^{(t)}\right] \in \mathbb{R}^{d\times n}, \\ \partial F(X^{(t)}, \xi^{(t)}) &:= \left[\nabla F_1(\mathbf{x}_1^{(t)}, \xi_1^{(t)}), \dots, \nabla F_n(\mathbf{x}_n^{(t)}, \xi_n^{(t)})\right] \in \mathbb{R}^{d\times n}. \end{aligned} \tag{28}$$

Instead of proving Theorem 4 directly, we prove a slightly more general statement in this section. Algorithm 2 relies on the (compressed) consensus Algorithm 1. However, we can also show convergence of Algorithm 2 for more general average consensus schemes. In Algorithm 4 below, the function $h : \mathbb{R}^{d\times n} \times \mathbb{R}^{d\times n} \to \mathbb{R}^{d\times n} \times \mathbb{R}^{d\times n}$ denotes a *blackbox averaging scheme*. Note that h could be random.

Algorithm 4 DECENTRALIZED SGD WITH ARBITRARY AVERAGE CONSENSUS SCHEME

input : $X^{(0)}$, stepsizes $\{\eta_t\}_{t=0}^{T-1}$, averaging function $h : \mathbb{R}^{d\times n} \times \mathbb{R}^{d\times n} \to \mathbb{R}^{d\times n} \times \mathbb{R}^{d\times n}$

1: *In parallel (task for worker $i, i \in [n]$)*
2: **for** t **in** $0 \dots T-1$ **do**
3: $\quad X^{(t+\frac{1}{2})} = X^{(t)} - \eta_t \partial F_i(X^{(t)}, \xi^{(t)})$ ▷ stochastic gradient updates
4: $\quad (X^{(t+1)}, Y^{(t+1)}) = h(X^{(t+\frac{1}{2})}, Y^{(t)})$ ▷ blackbox averaging/gossip
5: **end for**

In this work we in particular focus on two choices of h, the average consensus operator $h(X^{(t)}, Y^{(t)}) \mapsto (X^{(t+1)}, Y^{(t+1)})$:

- Setting $X^{(t+1)} = X^{(t)}W$ and $Y^{(t+1)} = X^{(t+1)}$ corresponds to standard (exact) averaging with mixing matrix W, as in algorithm (E-G).

- Setting $X^{(t+1)} = X^{(t)} + \gamma Y^{(t)}(W - I)$ and $Y^{(t+1)} = Y^{(t)} + Q(X^{(t+1)} - Y^{(t)})$ for $Y^{(t)} = \hat{X}^{(t)}$, we get the compressed consensus algorithm (CHOCO-G), leading to Algorithm 2, as introduced in the main text.

Assumption 3. *For an averaging scheme $h\colon \mathbb{R}^{d\times n}\times\mathbb{R}^{d\times n}\to\mathbb{R}^{d\times n}\times\mathbb{R}^{d\times n}$ let $(X^+, Y^+) := h(X, Y)$ for $X, Y \in \mathbb{R}^{d\times n}$. Assume that h preserves the average of the first iterate over all iterations:*

$$X^+\frac{\mathbf{1}\mathbf{1}^\top}{n} = X\frac{\mathbf{1}\mathbf{1}^\top}{n}\,, \qquad\qquad \forall X, Y \in \mathbb{R}^{d\times n}\,,$$

and that it converges with linear rate for a parameter $0 < p \leq 1$

$$\mathbb{E}_h\ \Psi(X^+, Y^+) \leq (1-p)\Psi(X, Y)\,, \qquad\qquad \forall X, Y \in \mathbb{R}^{d\times n}\,,$$

and Laypunov function $\Psi(X, Y) := \|X - \overline{X}\|_F^2 + \|X - Y\|_F^2$ with $\overline{X} := \frac{1}{n}X\mathbf{1}\mathbf{1}^\top$, where $\mathbb{E}_h$ denotes the expectation over internal randomness of averaging scheme h.

This assumption holds for exact averaging as in (E-G) with parameter $p = \gamma\rho$ (as shown in Theorem 1). For the proposed compressed consensus algorithm (CHOCO-G) the assumption holds for parameter $p = \frac{\delta\rho^2}{82}$ (as show in Theorem 2). Here δ denotes the compression ratio and ρ the eigengap of mixing matrx W. We can now state the more general Theorem (which generalizes Theorem 4):

Theorem 19. *Under Assumption 3 for $p > 0$, Algorithm 4 with stepsize $\eta_t = \frac{4}{\mu(a+t)}$, for parameter $a \geq \max\left\{\frac{5}{p}, 16\kappa\right\}$, $\kappa = \frac{L}{\mu}$ converges at the rate*

$$f(\mathbf{x}_{avg}^{(T)}) - f^\star \leq \frac{\mu a^3}{8S_T}\left\|\overline{\mathbf{x}}^{(0)} - \mathbf{x}^\star\right\|^2 + \frac{4T(T+2a)}{\mu S_T}\frac{\overline{\sigma}^2}{n} + \frac{64T}{\mu^2 S_T}(2L+\mu)\frac{40}{p^2}G^2,$$

where $\mathbf{x}_{avg}^{(T)} = \frac{1}{S_T}\sum_{t=0}^{T-1} w_t\overline{\mathbf{x}}^{(t)}$ for weights $w_t = (a+t)^2$, and $S_T = \sum_{t=0}^{T-1} w_t \geq \frac{1}{3}T^3$.

Proof of Theorem 4. The proof follows from Theorem 19 using the consensus averaging Algorithm 1 (giving $p = \frac{\rho^2\delta}{82}$ by Theorem 2) and the inequality $\mathbb{E}\ \mu\ \|\mathbf{x}_0 - \mathbf{x}^\star\| \leq 2G$ derived in (Rakhlin et al., 2012, Lemma 2) to upper bound the first term. □

D.1. Proof of Theorem 19

The proof below uses techniques from both (Stich et al., 2018) and (Stich, 2018).

Lemma 20. *The averages $\overline{\mathbf{x}}^{(t)}$ of the iterates of the Algorithm 4 satisfy the following*

$$\mathbb{E}_{\xi_1^{(t)},\dots,\xi_n^{(t)}}\ \|\overline{\mathbf{x}}^{(t+1)} - \mathbf{x}^\star\|^2 \leq \left(1 - \frac{\eta_t\mu}{2}\right)\left\|\overline{\mathbf{x}}^{(t)} - \mathbf{x}^\star\right\|^2 + \frac{\eta_t^2\overline{\sigma}^2}{n} - 2\eta_t\left(1 - 2L\eta_t\right)\left(f(\overline{\mathbf{x}}^{(t)}) - f^\star\right) + \\ + \eta_t\frac{2\eta_t L^2 + L + \mu}{n}\sum_{i=1}^n\left\|\overline{\mathbf{x}}^{(t)} - \mathbf{x}_i^{(t)}\right\|^2,$$

where $\overline{\sigma}^2 = \frac{1}{n}\sum_{i=1}^n\sigma_i^2$.

Proof. Because the blackbox averaging function h preserves the average (Assumption 3), we have

$$\begin{aligned}
\left\|\overline{\mathbf{x}}^{(t+1)}-\mathbf{x}^\star\right\|^2 &= \left\|\overline{\mathbf{x}}^{(t)}-\frac{\eta_t}{n}\sum_{j=1}^n\nabla F_j(\mathbf{x}_j^{(t)},\xi_j^{(t)})-\mathbf{x}^\star\right\|^2\\
&= \left\|\overline{\mathbf{x}}^{(t)}-\mathbf{x}^\star-\frac{\eta_t}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})+\frac{\eta_t}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})-\frac{\eta_t}{n}\sum_{j=1}^n\nabla F_j(\mathbf{x}_j^{(t)},\xi_j^{(t)})\right\|^2 =\\
&= \left\|\overline{\mathbf{x}}^{(t)}-\mathbf{x}^\star-\frac{\eta_t}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})\right\|^2+\eta_t^2\left\|\frac{1}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})-\frac{1}{n}\sum_{j=1}^n\nabla F_j(\mathbf{x}_j^{(t)},\xi_j^{(t)})\right\|^2+\\
&\quad+\frac{2\eta_t}{n}\left\langle\overline{\mathbf{x}}^{(t)}-\mathbf{x}^\star-\frac{\eta_t}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)}),\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})-\sum_{j=1}^n\nabla F_j(\mathbf{x}_j^{(t)},\xi_j^{(t)})\right\rangle.
\end{aligned}$$

The last term is zero in expectation, as $\mathbb{E}_{\xi_i^{(t)}}\nabla F_i(\mathbf{x}_i^{(t)},\xi_i^{(t)})=\nabla f_i(\mathbf{x}_i^{(t)})$. The second term is less than $\frac{\eta_t^2\overline{\sigma}^2}{n}$ (Remark 11). The first term can be written as:

$$\left\|\overline{\mathbf{x}}^{(t)}-\mathbf{x}^\star-\frac{\eta_t}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})\right\|^2=\left\|\overline{\mathbf{x}}^{(t)}-\mathbf{x}^\star\right\|^2+\eta_t^2\underbrace{\left\|\frac{1}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})\right\|^2}_{=:T_1}\underbrace{-2\eta_t\left\langle\overline{\mathbf{x}}^{(t)}-\mathbf{x}^\star,\frac{1}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}_i^{(t)})\right\rangle}_{=:T_2}.$$

We can estimate

$$\begin{aligned}
T_1 &= \left\|\frac{1}{n}\sum_{i=1}^n(\nabla f_i(\mathbf{x}_i^{(t)})-\nabla f_i(\overline{\mathbf{x}}^{(t)})+\nabla f_i(\overline{\mathbf{x}}^{(t)})-\nabla f_i(\mathbf{x}^\star))\right\|^2\\
&\overset{(13)}{\leq}\frac{2}{n}\sum_{i=1}^n\left\|\nabla f_i(\mathbf{x}_i^{(t)})-\nabla f_i(\overline{\mathbf{x}}^{(t)})\right\|^2+2\left\|\frac{1}{n}\sum_{i=1}^n\nabla f_i(\overline{\mathbf{x}}^{(t)})-\frac{1}{n}\sum_{i=1}^n\nabla f_i(\mathbf{x}^\star)\right\|^2\\
&\overset{(9),(11)}{\leq}\frac{2L^2}{n}\sum_{i=1}^n\left\|\mathbf{x}_i^{(t)}-\overline{\mathbf{x}}^{(t)}\right\|^2+\frac{4L}{n}\sum_{i=1}^n\left(f_i(\overline{\mathbf{x}}^{(t)})-f_i(\mathbf{x}^\star)\right)\\
&=\frac{2L^2}{n}\sum_{i=1}^n\left\|\mathbf{x}_i^{(t)}-\overline{\mathbf{x}}^{(t)}\right\|^2+4L\left(f(\overline{\mathbf{x}}^{(t)})-f^\star\right).
\end{aligned}$$

And for the remaining T_2 term:

$$\begin{aligned}
-\frac{1}{\eta_t}T_2 &= -\frac{2}{n}\sum_{i=1}^n\left[\left\langle\overline{\mathbf{x}}^{(t)}-\mathbf{x}_i^{(t)},\nabla f_i(\mathbf{x}_i^{(t)})\right\rangle+\left\langle\mathbf{x}_i^{(t)}-\mathbf{x}^\star,\nabla f_i(\mathbf{x}_i^{(t)})\right\rangle\right]\\
&\overset{(9),(10)}{\leq}-\frac{2}{n}\sum_{i=1}^n\left[f_i(\overline{\mathbf{x}}^{(t)})-f_i(\mathbf{x}_i^{(t)})-\frac{L}{2}\left\|\overline{\mathbf{x}}^{(t)}-\mathbf{x}_i^{(t)}\right\|^2+f_i(\mathbf{x}_i^{(t)})-f_i(\mathbf{x}^\star)+\frac{\mu}{2}\left\|\mathbf{x}_i^{(t)}-\mathbf{x}^\star\right\|^2\right]\\
&\overset{(13)}{\leq}-2\left(f(\overline{\mathbf{x}}^{(t)})-f(\mathbf{x}^\star)\right)+\frac{L+\mu}{n}\sum_{i=1}^n\left\|\overline{\mathbf{x}}^{(t)}-\mathbf{x}_i^{(t)}\right\|^2-\frac{\mu}{2}\left\|\overline{\mathbf{x}}^{(t)}-\mathbf{x}^\star\right\|^2.
\end{aligned}$$

Putting everything together we are getting statement of the lemma. □

Lemma 21. *The iterates $\{X^{(t)}\}_{t\geq 0}$ of Algorithm 4 with stepsizes $\eta_t=\frac{b}{t+a}$, for parameters $a\geq\frac{5}{p}$, $b>0$ satisfy*

$$\left\|X^{(t+1)}-\overline{X}^{(t+1)}\right\|_F^2\leq 40\eta_t^2\frac{1}{p^2}nG^2.$$

Here $0<p\leq 1$ denotes the a convergence rate of the blackbox averaging algorithm as in Assumption 3.

Proof. Using linear convergence of the blackbox averaging algorithm as given in Assumption 3 we can write for $\Xi := \mathbb{E}\left\|X^{(t+1)} - \overline{X}^{(t+1)}\right\|_F^2 + \mathbb{E}\left\|X^{(t+1)} - Y^{(t+1)}\right\|_F^2$,

$$
\begin{aligned}
\Xi &\leq (1-p)\mathbb{E}\left\|\overline{X}^{(t+\frac{1}{2})} - X^{(t+\frac{1}{2})}\right\|_F^2 + (1-p)\mathbb{E}\left\|Y^{(t)} - X^{(t+\frac{1}{2})}\right\|_F^2 \\
&= (1-p)\mathbb{E}\left\|\overline{X}^{(t)} - X^{(t)} + \eta_t \partial F(X^{(t)}, \xi^{(t)})\left(\frac{\mathbf{1}\mathbf{1}^\top}{n} - I\right)\right\|_F^2 \\
&\quad + (1-p)\mathbb{E}\left\|Y^{(t)} - X^{(t)} + \eta_t \partial F(X^{(t)}, \xi^{(t)})\right\|_F^2 \\
&\overset{(15)}{\leq} (1-p)(1+\alpha_3^{-1})\mathbb{E}\left(\left\|\overline{X}^{(t)} - X^{(t)}\right\|_F^2 + \left\|Y^{(t)} - X^{(t)}\right\|_F^2\right) \\
&\quad + (1-p)(1+\alpha_3)\eta_t^2\mathbb{E}\left(\left\|\partial F(X^{(t)}, \xi^{(t)})\left(\frac{\mathbf{1}\mathbf{1}^\top}{n} - I\right)\right\|_F^2 + \left\|\partial F(X^{(t)}, \xi^{(t)})\right\|_F^2\right) \\
&\leq (1-p)\left((1+\alpha_3^{-1})\mathbb{E}\left(\left\|\overline{X}^{(t)} - X^{(t)}\right\|_F^2 + \left\|Y^{(t)} - X^{(t)}\right\|_F^2\right) + 2n(1+\alpha_3)\eta_t^2 G^2\right) \\
&\overset{\alpha_3=\frac{2}{p}}{\leq} \left(1 - \frac{p}{2}\right)\mathbb{E}\left(\left\|\overline{X}^{(t)} - X^{(t)}\right\|_F^2 + \left\|Y^{(t)} - X^{(t)}\right\|_F^2\right) + \frac{4n}{p}\eta_t^2 G^2\,.
\end{aligned}
$$

The statement now follows from Lemma 22 and the inequality

$$
\mathbb{E}\left\|X^{(t+1)} - \overline{X}^{(t+1)}\right\|_F^2 \leq \Xi := \mathbb{E}\left\|X^{(t+1)} - Y^{(t+1)}\right\|_F^2 + \mathbb{E}\left\|X^{(t+1)} - \overline{X}^{(t+1)}\right\|_F^2\,. \qquad \square
$$

Lemma 22. *Let $\{r_t\}_{t\geq 0}$ denote a sequence of positive real values satisfying $r_0 = 0$ and*

$$
r_{t+1} \leq \left(1 - \frac{p}{2}\right) r_t + \frac{2}{p}\eta_t^2 A\,, \qquad \forall t \geq 0\,,
$$

for a parameter $p > 0$, stepsize $\eta_t = \frac{b}{t+a}$, for parameters $a \geq \frac{5}{p}$ and with arbitrary $b > 0$. Then r_t is bounded as

$$
r_t \leq 20\eta_t^2 \frac{1}{p^2} A\,, \qquad \forall t \geq 0\,.
$$

Proof. We will proceed the proof by induction. For $t = 0$ the statement is true by assumption on $r_0 = 0$. Suppose that for timestep t the statement is also true, then for timestep $t+1$

$$
r_{t+1} \leq \left(1 - \frac{p}{2}\right) r_t + \frac{2}{p}\eta_t^2 A \leq \left(1 - \frac{p}{2}\right) 20\eta_t^2 \frac{1}{p^2} A + \frac{2}{p}\eta_t^2 A = A\eta_t^2 \frac{1}{p^2}\left(-8p + 20\right)\,.
$$

Now we show $\eta_t^2\left(-8p + 20\right) \leq 20\eta_{t+1}^2$ which proves the claim. By assumption $p \geq \frac{5}{a}$, hence

$$
\eta_t^2\left(-8p + 20\right) \leq 20\eta_t^2\left(1 - \frac{2}{a}\right) \leq 20\eta_{t+1}^2\,,
$$

where the second inequality follows from

$$
\begin{aligned}
(a+t+1)^2\left(1 - \frac{2}{a}\right) &= (a+t)^2 + 2(a+t) + 1 - \left(2\frac{(a+t)^2}{a} + 4\frac{(a+t)}{a} + \frac{2}{a}\right) \\
&\leq (a+t)^2 + 2(a+t) + 1 - (2(a+t) + 4) \leq (a+t)^2\,.
\end{aligned}
\qquad \square
$$

Lemma 23 (Stich (2018))**.** *Let $\{a_t\}_{t\geq 0}$, $a_t \geq 0$, $\{e_t\}_{t\geq 0}$, $e_t \geq 0$ be sequences satisfying*

$$
a_{t+1} \leq (1 - \mu\eta_t)a_t - \eta_t e_t A + \eta_t^2 B + \eta_t^3 C\,,
$$

for stepsizes $\eta_t = \frac{4}{\mu(a+t)}$ *and constants* $A > 0, B, C \geq 0$, $\mu > 0$, $a > 1$. *Then*

$$\frac{A}{S_T}\sum_{t=0}^{T-1} w_t e_t \leq \frac{\mu a^3}{4S_T} a_0 + \frac{2T(T+2a)}{\mu S_T} B + \frac{16T}{\mu^2 S_T} C\,,$$

for $w_t = (a+t)^2$ *and* $S_T := \sum_{t=0}^{T-1} w_t = \frac{T}{6}(2T^2 + 6aT - 3T + 6a^2 - 6a + 1) \geq \frac{1}{3}T^3$.

Proof of Theorem 19. Substituting the result of Lemma 21 into the bound provided in Lemma 20 (here we use $a \geq \frac{5}{p}$) we get that

$$\mathbb{E}\left\|\overline{\mathbf{x}}^{(t+1)} - \mathbf{x}^\star\right\|^2 \leq \left(1 - \frac{\eta_t \mu}{2}\right)\mathbb{E}\left\|\overline{\mathbf{x}}^{(t)} - \mathbf{x}^\star\right\|^2 - 2\eta_t\left(1 - 2L\eta_t\right)e_t + \eta_t^2 \frac{\overline{\sigma}^2}{n}(2\eta_t L^2 + L + \mu)40\eta_t^3 \frac{1}{p^2}G^2,$$

For $\eta_t \leq \frac{1}{4L}$ (this holds, as $a \leq 16\kappa$) it holds $2L\eta_t - 1 \leq -\frac{1}{2}$ and $(2\eta_t L^2 + L + \mu) < (2L + \mu)$, hence

$$\mathbb{E}\left\|\overline{\mathbf{x}}^{(t+1)} - \mathbf{x}^\star\right\|^2 \leq \left(1 - \frac{\eta_t \mu}{2}\right)\mathbb{E}\left\|\overline{\mathbf{x}}^{(t)} - \mathbf{x}^\star\right\|^2 + \frac{\eta_t^2 \overline{\sigma}^2}{n} - \eta_t e_t + (2L + \mu)40\eta_t^3 \frac{1}{p^2}G^2\,.$$

From Lemma 23 we get

$$\frac{1}{S_T}\sum_{t=0}^{T-1} w_t e_t \leq \frac{\mu a^3}{8S_T}\left\|\overline{\mathbf{x}}^{(0)} - \mathbf{x}^\star\right\|^2 + \frac{4T(T+2a)}{\mu S_T}\frac{\overline{\sigma}^2}{n} + \frac{64T}{\mu^2 S_T}(2L + \mu)40\frac{1}{p^2}G^2,$$

for weights $w_t = (a+t)^2$ and $S_T := \sum_{t=0}^{T-1} w_t = \frac{T}{6}(2T^2 + 6aT - 3T + 6a^2 - 6a + 1) \geq \frac{1}{3}T^3$, where p is convergence rate of the averaging scheme. The theorem statement follows from convexity of f. □

E. Efficient Implementation of the Algorithms

In this section we present memory-efficient implementations of CHOCO-GOSSIP and CHOCO-SGD algorithms, which require each node to store only three vectors: $\mathbf{x}$, $\hat{\mathbf{x}}_i$ and $\mathbf{s}_i = \sum_{i=1}^n w_{ij}\hat{\mathbf{x}}_j$.

Algorithm 5 Memory-efficient CHOCO-GOSSIP

input : Initial values $\mathbf{x}_i^{(0)} \in \mathbb{R}^d$ on each node $i \in [n]$, stepsize γ, communication graph $G = ([n], E)$ and mixing matrix W, initialize $\hat{\mathbf{x}}_i^{(0)} := \mathbf{0}$, $\mathbf{s}_i^{(0)} = 0, \forall i$

1: **for** t **in** $0 \dots T-1$ **do** *in parallel for all workers* $i \in [n]$
2: $\quad \mathbf{x}_i^{(t+1)} := \mathbf{x}_i^{(t)} + \gamma\left(\mathbf{s}_i^{(t)} - \hat{\mathbf{x}}_i^{(t)}\right)$
3: $\quad \mathbf{q}_i^{(t)} := Q(\mathbf{x}_i^{(t+1)} - \hat{\mathbf{x}}_i^{(t)})$
4: $\quad$ **for** neighbors j : $\{i,j\} \in E$ (including $\{i\} \in E$) **do**
5: $\quad\quad$ Send $\mathbf{q}_i^{(t)}$ and receive $\mathbf{q}_j^{(t)}$
6: $\quad$ **end for**
7: $\quad \hat{\mathbf{x}}_i^{(t+1)} = \hat{\mathbf{x}}_i^{(t)} + \mathbf{q}_i^{(t)}$
8: $\quad \mathbf{s}_i^{(t+1)} := \mathbf{s}_i^{(t)} + \sum_{i=1}^n w_{ij}\mathbf{q}_j^{(t)}$
9: **end for**

Algorithm 6 Memory-efficient CHOCO-SGD

input : Initial values $\mathbf{x}_i^{(0)} \in \mathbb{R}^d$ on each node $i \in [n]$, consensus stepsize γ, communication graph $G = ([n], E)$ and mixing matrix W, initialize $\hat{\mathbf{x}}_i^{(0)} := \mathbf{0}\ \forall i$

for t **in** $0 \dots T-1$ **do** *in parallel for all workers* $i \in [n]$
  Sample $\xi_i^{(t)}$, compute gradient $\mathbf{g}_i^{(t)} := \nabla F_i(\mathbf{x}_i^{(t)}, \xi_i^{(t)})$
  $\mathbf{x}_i^{(t+\frac{1}{2})} := \mathbf{x}_i^{(t)} - \eta_t \mathbf{g}_i^{(t)}$
  $\mathbf{x}_i^{(t+1)} := \mathbf{x}_i^{(t+\frac{1}{2})} + \gamma \left(\mathbf{s}_i^{(t)} - \hat{\mathbf{x}}_i^{(t)} \right)$
  $\mathbf{q}_i^{(t)} := Q(\mathbf{x}_i^{(t+1)} - \hat{\mathbf{x}}_i^{(t)})$
  for neighbors $j \colon \{i,j\} \in E$ (including $\{i\} \in E$) **do**
    Send $\mathbf{q}_i^{(t)}$ and receive $\mathbf{q}_j^{(t)}$
  end for
  $\hat{\mathbf{x}}_j^{(t+1)} := \mathbf{q}_i^{(t)} + \hat{\mathbf{x}}_i^{(t)}$
  $\mathbf{s}_i^{(t+1)} := \mathbf{s}_i^{(t)} + \sum_{i=1}^n w_{ij} \mathbf{q}_j^{(t)}$
end for

F. Refined Experiments Details

Our code is open-source and publicly available at `github.com/epfml/ChocoSGD`.

F.1. Parameters Search Details of SGD Experiments

For each optimization problem, we first tuned γ on a separate average consensus problem with the same configuration (topology, number of nodes, quantization, dimension). Parameters a, b where later tuned separately for each algorithm by running the algorithm for 10 epochs. To find a and b we performed grid search independently for each algorithm and each quantization function. For values of a we used logarithmic grid of powers of 10. We searched values of b in the set $\{1, 0.1d, d, 10d, 100d\}$.

F.2. Calculation of Number of Transmitted Bits

For (qsgd_{256}) compression we assume that at every iteration 9 bits are transmitted instead of 64 (1 bit for sign and 8 bits for quantization level). Analogously, for (qsgd_{16}) 5 bits are transmitted. For $(\mathrm{top}_{1\%})$ for every chosen coordinate we need to transmit the index of this coordinate and the value. That's why we compute the number of transmitted bits as $64 + \log_2(d)$ for every transmitted coordinate. For $(\mathrm{random}_{1\%})$ we assume that neighbours have access to random seed, so indexes are not needed to be transmitted.

G. Additional Experiments

G.1. Average Consensus

G.1.1. ORIGINAL VECTORS OF *epsilon* DATASET

We compare CHOCO-GOSSIP with baselines on original vectors of *epsilon* dataset on Figures 7, 8. In this case the solution (i.e. the average of all vectors) is close to zero, which is advantageous for all the baselines and for CHOCO-GOSSIP as the initial value of $\hat{\mathbf{x}}_i$ is zero.

The proposed scheme (CHOCO-G) with 8 bit quantization (qsgd_{256}) converges with the same rate as (E-G) that uses exact communications (Fig. 7, left), while it requires much less data to be transmitted (Fig. 7, right). Even in this setting schemes (Q1-G) and (Q2-G) can do not converge and reach only accuracies of 10^{-4} –10^{-5}. The scheme (Q1-G) even starts to diverge, because the quantization error becomes larger than the optimization error.

With sparsified communication $(\mathrm{rand}_{1\%})$, i.e. transmitting only 1% of all the coordinates, the scheme (Q1-G) quickly zeros out all the coordinates and just because the average close to zero, it reaches accuracy of 10^{-3}. (Q2-G) diverges because quantization error is too large already from the first step (Fig. 8). CHOCO-GOSSIP proves to be more robust and converges. The observed rate matches with the theoretical findings, as we expect the scheme with factor $100\times$ compression to be $100\times$

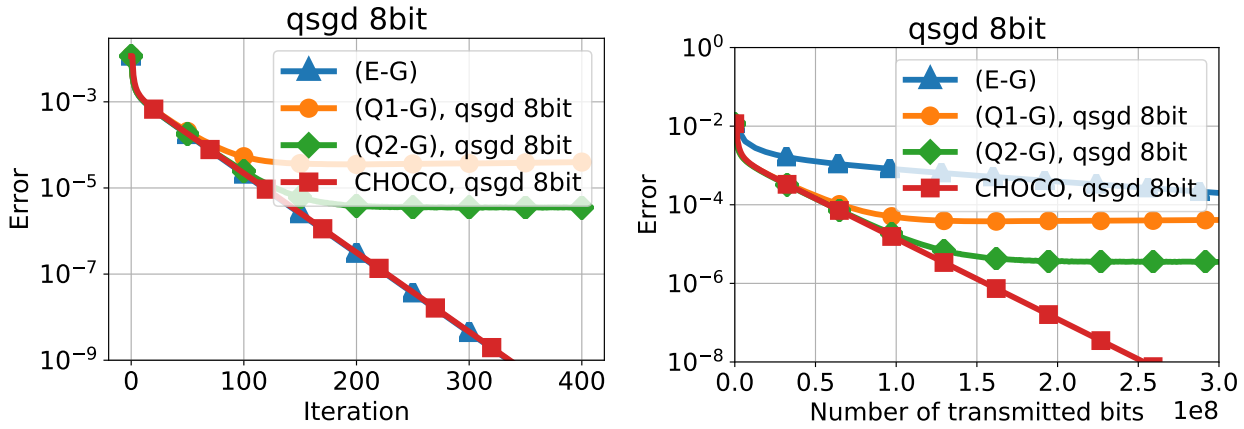

Figure 7. Average consensus on the ring topology with $n = 25$ nodes, $d = 2000$ coordinates and (qsgd_{256}) compression

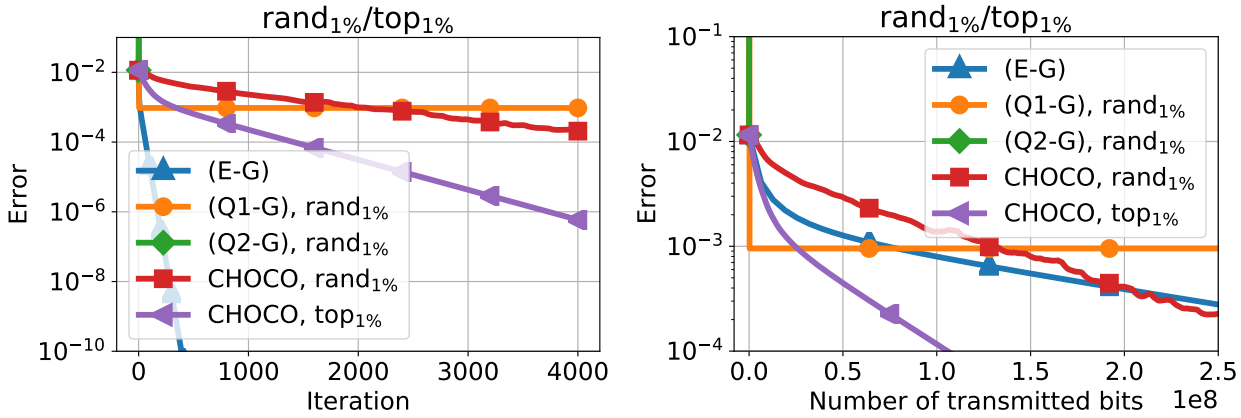

Figure 8. Average consensus on the ring topology with $n = 25$ nodes, $d = 2000$ coordinates and $(\mathrm{rand}_{1\%})$ and $(\mathrm{top}_{1\%})$ compression

slower than (E-G) without compression. In terms of total data transmitted, both schemes converge at the same speed (Fig. 8, right).

G.1.2. Additional Experiments for Shifted Vectors

On Figures 2, 3 CHOCO-GOSSIP in the beginning diverges because initial error $\left\|\mathbf{x}_i^{(t)} - \hat{\mathbf{x}}_i^{(t)}\right\|^2$ is large and is not depicted there. In this section we plot additionally the error $e_t = \mathbb{E}_Q \sum_{i=1}^n \left(\left\|\mathbf{x}_i^{(t)} - \overline{\mathbf{x}}\right\|^2 + \left\|\mathbf{x}_i^{(t)} - \hat{\mathbf{x}}_i^{(t)}\right\|^2\right)$ for CHOCO-GOSSIP to show that our theory holds and CHOCO-GOSSIP converges linearly in terms of e_t from the first step.

G.2. Decentralized SGD

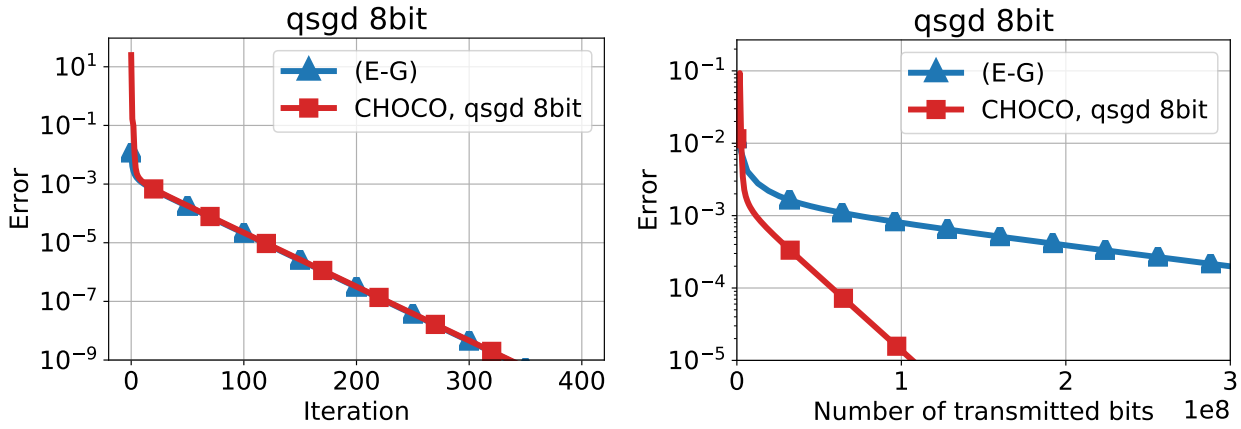

Figure 9. Full errors e_t for average consensus on the ring topology with $n = 25$ nodes, $d = 2000$ coordinates and (qsgd_{256}) compression

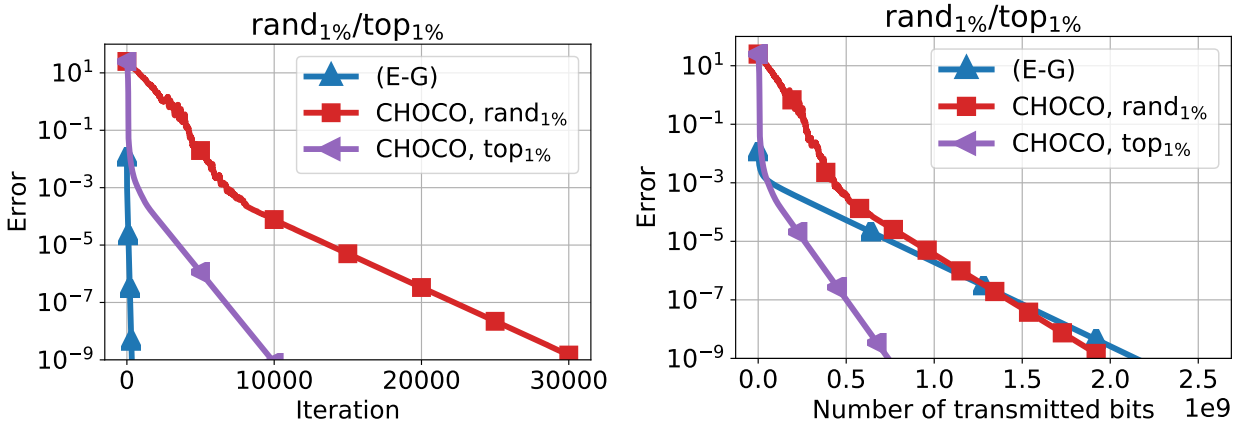

Figure 10. Full errors e_t for average consensus on the ring topology with $n = 25$ nodes, $d = 2000$ coordinates and $(\mathrm{rand}_{1\%})$ and $(\mathrm{top}_{1\%})$ compression

	Epsilon			RCV1		
experiment	a	τ	γ	a	τ	γ
PLAIN	0.1	d	-	1	1	-
CHOCO, (qsgd_{16})	0.1	d	0.34	1	1	0.078
CHOCO, ($\text{rand}_{1\%}$)	0.1	d	0.01	1	$0.1d$	0.016
CHOCO, ($\text{top}_{1\%}$)	0.1	d	0.04	1	1	0.04
DCD, ($\text{rand}_{1\%}$)	10^{-15}	d	-	10^{-15}	d	-
DCD, (qsgd_{16})	0.01	d	-	10^{-15}	d	-
ECD, ($\text{rand}_{1\%}$)	10^{-6}	d	-	10^{-4}	$10d$	-
ECD, (qsgd_{16})	10^{-6}	d	-	10^{-15}	d	-

Table 5. Values for initial learning rate and consensus learning rate used in SGD experiments Fig. 5, 6. Parameter γ found separately by tuning average consensus with the same configuration (topology, number of nodes, quantization, dimension). Parameters a, τ found by tuning. ECD, DCD stepsizes are small because it diverge for larger choices.

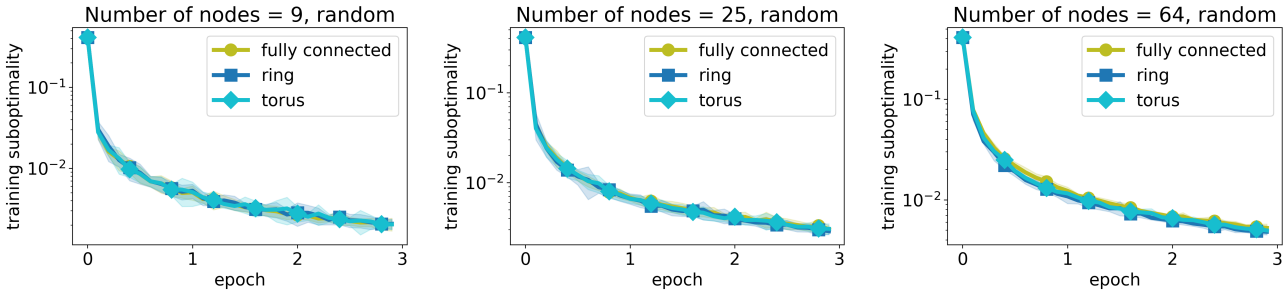

Figure 11. Performance of Algorithm 3 on ring, torus and fully connected topologies for $n \in \{9, 25, 64\}$ nodes. *Randomly shuffled* data between workers

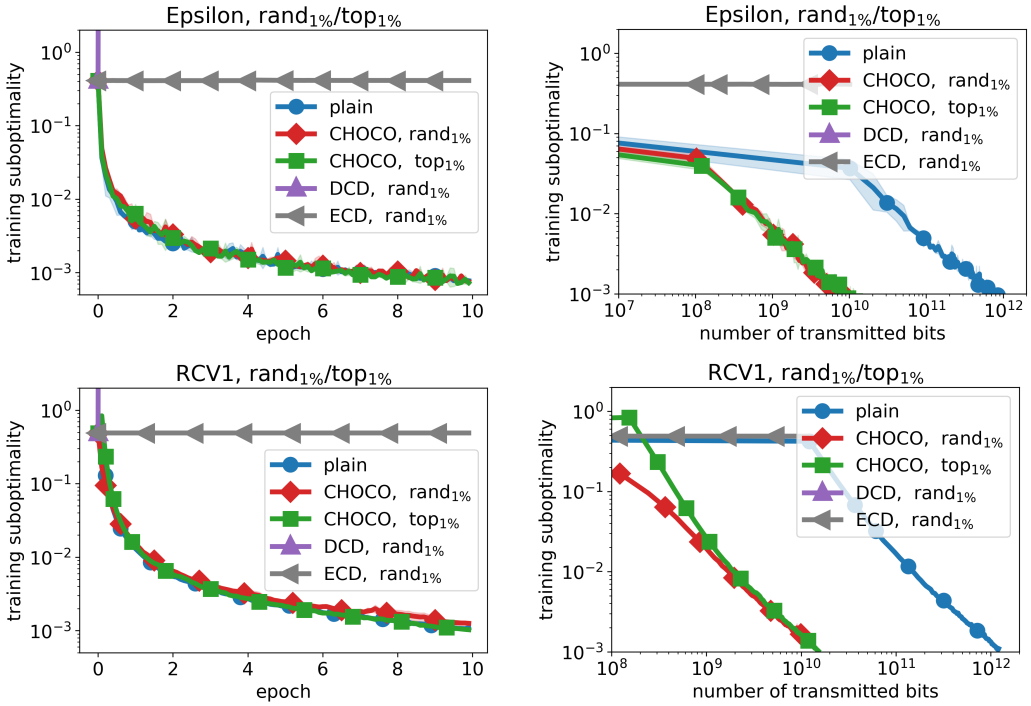

Figure 12. Comparison of Algorithm 3 (plain), ECD-SGD, DCD-SGD and CHOCO-SGD with ($\text{rand}_{1\%}$) sparsification (in addition ($\text{top}_{1\%}$) for CHOCO-SGD), for *epsilon* (top) and *rcv1* (bottom) in terms of iterations (left) and communication cost (right). *Randomly shuffled* data between workers.

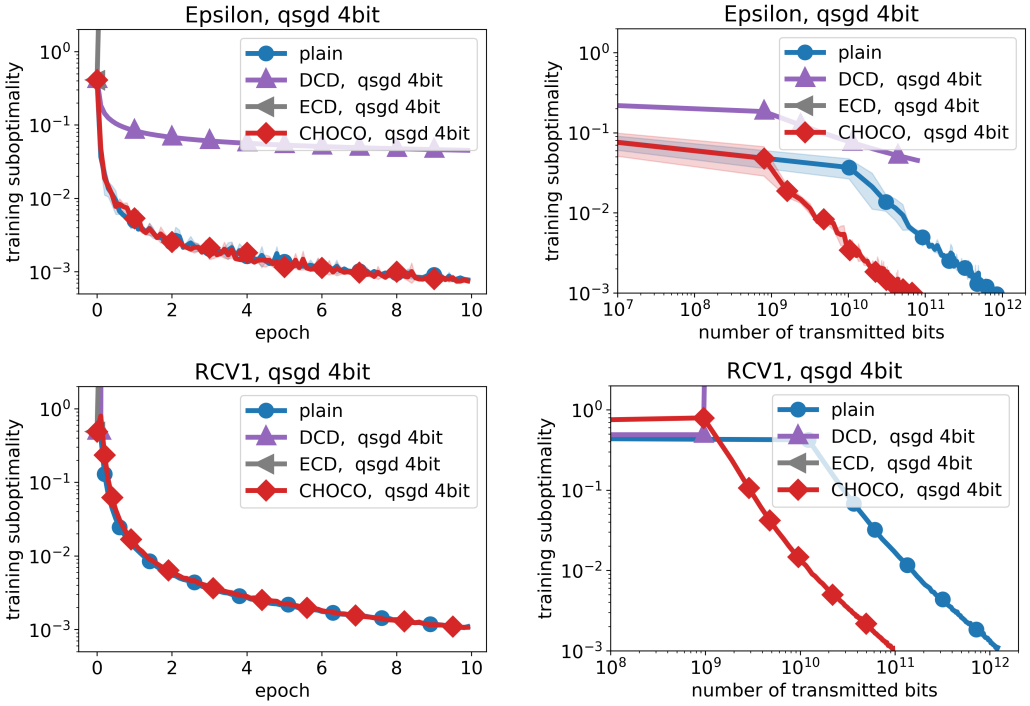

Figure 13. Comparison of Algorithm 3 (plain), ECD-SGD, DCD-SGD and CHOCO-SGD with (qsgd_{16}) quantization, for $epsilon$ (top) and $rcv1$ (bottom) in terms of iterations (left) and communication cost (right). *Randomly shuffled* data between workers.