Abstract—This article develops a communication-efficient algorithm to solve the stochastic optimization problem defined over a distributed network, aiming at reducing the burdensome communication in applications, such as distributed machine learning. Different from the existing works based on quantization and sparsification, we introduce a communication-censoring technique to reduce the transmissions of variables, which leads to our communication-censored distributed stochastic gradient descent (CSGD) algorithm. Specifically, in CSGD, the latest minibatch stochastic gradient at a worker will be transmitted to the server if and only if it is sufficiently informative. When the latest gradient is not available, the stale one will be reused at the server. To implement this communication-censoring strategy, the batch size is increasing in order to alleviate the effect of stochastic gradient noise. Theoretically, CSGD enjoys the same order of convergence rate as that of SGD but effectively reduces communication. Numerical experiments demonstrate the sizable communication saving of CSGD.

Index Terms—Communication censoring, communication efficiency, distributed optimization, stochastic gradient descent (SGD).

I. INTRODUCTION

Considering a distributed network with one server and \( M \) workers, we aim to design a communication-efficient algorithm to solve the following optimization problem:

\[
\min_{\mathbf{x} \in \mathbb{R}^d} \sum_{m=1}^{M} \mathbb{E}_{\xi_m}[f_m(\mathbf{x}; \xi_m)]
\]

where \( \mathbf{x} \) is the optimization variable, \( \{f_m\}_{m=1}^{M} \) are smooth local objective functions with \( f_m \) being kept at worker \( m \), and \( \{\xi_m\}_{m=1}^{M} \) are independent random variables associated with distributions \( \{\mathcal{D}_m\}_{m=1}^{M} \).

Problem (1) arises in a wide range of science and engineering fields, e.g., in distributed machine learning [1]. For distributed machine learning, there are two major drivers for solving problems in the form of (1): 1) distributed computing resources—for massive and high-dimensional datasets, performing the training processes over multiple workers in parallel is more efficient than relying on a single worker and 2) user privacy concerns—with a massive amount of sensors, nowadays, distributively collected data may contain private information about end users, and thus, keeping the computation at local workers is more privacy-preserving than uploading the data to central servers. However, the communication between the server and the workers is one of the major bottlenecks of distributed machine learning. Indeed, reducing the communication cost is also a common consideration in popular machine learning frameworks, e.g., federated learning [2]–[5].

A. Prior Art

Before discussing our algorithm, we review several existing works for solving (1) in a distributed manner.

Finding the best communication–computation tradeoff has been a long-standing problem in distributed consensus optimization [6], [7] since it is critical to many important engineering problems in signal processing and wireless communications [8]. For the emerging machine learning tasks, communication efficiency has been frequently discussed during the past decade [9]–[11], and it attracts more attention when the notion of federated learning becomes popular [2]–[5]. Many dual-domain methods have been demonstrated as efficient problem solvers [12], [13], which, nonetheless, requires primal-dual loops and own empirical communication-saving performance without theoretical guarantee.

In general, there are three different kinds of strategies to save communication costs. First, the acceleration strategies aim to reach a target solution with less number of iterations and, hence, with less round of communication. Typical approaches include momentum [14], variance reduction [15], and those utilizing the special problem structures [16], to name a few. We refer the readers to a recent survey paper [17] for the acceleration strategies. Second, due to the limited bandwidth in practice, transmitting compressed information, which is called quantization [18]–[21] or sparsification [22], [23], is an effective method to alleviate the communication...
burden. In particular, the quantized version of the stochastic gradient descent (SGD) has been developed [20], [21]. The sparsification methods propose to transmit the most valuable elements, instead of the entire messages [22], [23]. Third, instead of consistently broadcasting the latest information, the cutoff of some “less informative” messages is encouraged, which results in the so-called event-triggered control [24], [25] or communication censoring [26]–[28]. For adaptive communication censoring, the work of [26] extends the original continuous-time setting [24] to discrete-time setting [25] and achieves a sublinear rate of convergence, while the work of [28] shows a linear rate, and its further extension [27] proves both rates of convergence. However, the algorithms in [26]–[28] utilize the primal-dual loops and only rigorously proves both rates of convergence. Nevertheless, the local SGD requires the establishment convergence without showing communication reduction [26]–[28] utilize the primal-dual loops and only rigorously proves both rates of convergence. However, the algorithms achieves a sublinear rate of convergence, while the work continuous-time setting [24] to discrete-time setting [25] and achieves a sublinear rate of convergence, while the work.

Consider the SGD update

\[ x^{k+1} = x^k - \alpha \hat{\nabla} x^k \]

where \( \hat{\nabla} x_k \) is the averaged gradient at every iteration, which is rather expensive in communication.

To maintain the desired properties of SGD and overcome its limitations, we design our communication-censored distributed SGD (CSDG) method, which leverages the communication-censoring strategy. Consider a starting point \( x^{k-1} \) at iteration \( k \). As in SGD, every worker \( m \) samples a batch of i.i.d. stochastic gradients \( \{ \nabla f_m(x^{k-1}; z_m^{k,b}) \}_{b=1}^{B^k} \) with a batch size \( B^k \) and then calculates the sample mean \( \nabla ^k_m := \frac{1}{B^k} \sum_{b=1}^{B^k} \nabla f_m(x^{k-1}; z_m^{k,b}) \). Seeking a desired communication-censoring strategy, we are interested in the distance between the calculated gradient \( \nabla ^k_m \) at worker \( m \) and the latest uploaded gradient \( \hat{\nabla} x^k_m \) with a batch size \( B^k \) back to the server, which aggregates all the means and performs the SGD update with the step size \( \alpha \) as

\[ \bar{x}^k = \bar{x}^{k-1} - \alpha \hat{\nabla} x^k \]

Specifically, in (3), we use the following censoring threshold:

\[ \tau^k := \frac{1}{M^2} \left( \frac{\bar{\omega}}{D} \sum_{d=1}^{D} \| \nabla ^{k-d} \|^2 + \sigma^k \right) \]

where \( \| \nabla ^{k-d} \|_{d=1}^{D} \) are two-norms of recent \( D \) aggregated gradients with \( \nabla ^{k-d} = 0 \) for \( d > 0 \), \( \bar{\omega} \) is a weight representing the confidence of the censoring threshold, and \( \sigma^k \) controls the randomness of the stochastic part that we call control size. The adaptive threshold consists of a scaling factor \( \alpha (1/M^2) \) and the sum of two parts. The first part learns information from the previous \( D \) updates, while the second part helps alleviate the stochastic gradient noise.

Building upon this innovative censoring condition, our main contributions can be summarized as follows.

1) We propose CSGD with dynamic batch size that achieves the same order of convergence rate as the original SGD.
2) CSGD provably saves the total number of uploads to reach the targeted accuracy relative to SGD.
3) We conduct extensive experiments to show the superior performance of the proposed CSGD algorithm.

II. CSGD DEVELOPMENT

In this section, we introduce CSGD and provide some insights behind its threshold design in (5). In CSGD, at the

\[ x^k = x^{k-1} - \alpha \hat{\nabla} x^k \]

Therein, every worker is required to upload the latest locally averaged gradient at every iteration, which is rather expensive in communication.

To maintain the desired properties of SGD and overcome its limitations, we design our communication-censored distributed SGD (CSDG) method, which leverages the communication-censoring strategy. Consider a starting point \( x^{k-1} \) at iteration \( k \). As in SGD, every worker \( m \) samples a batch of i.i.d. stochastic gradients \( \{ \nabla f_m(x^{k-1}; z_m^{k,b}) \}_{b=1}^{B^k} \) with a batch size \( B^k \) and then calculates the sample mean \( \nabla ^k_m := \frac{1}{B^k} \sum_{b=1}^{B^k} \nabla f_m(x^{k-1}; z_m^{k,b}) \). Seeking a desired communication-censoring strategy, we are interested in the

\[ \tau^k := \frac{1}{M^2} \left( \frac{\bar{\omega}}{D} \sum_{d=1}^{D} \| \nabla ^{k-d} \|^2 + \sigma^k \right) \]

where \( \| \nabla ^{k-d} \|_{d=1}^{D} \) are two-norms of recent \( D \) aggregated gradients with \( \nabla ^{k-d} = 0 \) for \( d > 0 \), \( \bar{\omega} \) is a weight representing the confidence of the censoring threshold, and \( \sigma^k \) controls the randomness of the stochastic part that we call control size. The adaptive threshold consists of a scaling factor \( \alpha (1/M^2) \) and the sum of two parts. The first part learns information from the previous \( D \) updates, while the second part helps alleviate the stochastic gradient noise.

Building upon this innovative censoring condition, our main contributions can be summarized as follows.

1) We propose CSGD with dynamic batch size that achieves the same order of convergence rate as the original SGD.
2) CSGD provably saves the total number of uploads to reach the targeted accuracy relative to SGD.
3) We conduct extensive experiments to show the superior performance of the proposed CSGD algorithm.

II. CSGD DEVELOPMENT

In this section, we introduce CSGD and provide some insights behind its threshold design in (5). In CSGD, at the
beginning of iteration $k$, the server broadcasts its latest variable and threshold to all workers. With the considerations of data privacy and uploading burden, each worker locally computes an estimate of its gradient with batch size $B^k$ and then decides whether to upload the fresh gradient or not. Specifically, the worker’s upload will be skipped if and only if $\|\nabla m_k - \hat{\nabla}_m^{k-1}\|^2 \leq \tau^k$. When such a communication skipping happens, we say that the worker is censored. At the end of iteration $k$, the server only receives the latest uploaded gradients and updates its variable via (4) and the censoring threshold $\tau^{k+1}$ via (5), using the magnitudes of $D$ recent updates given by 

$$\|\nabla d_{k-d}\|_{d=0}^{-1}.$$ 

We illustrate CSGD in Fig. 1 and summarize it in Algorithm 1.

![Algorithm 1 CSGD](image)

**Algorithm 1 CSGD**

**Require:** $\alpha$, \{${B^k}_{m=1}^{K}$, ${\sigma^k}_{m=1}^{K}$

**Initialize:** $x^0$, $\{\nabla m_0\}_{m=1}^{M}$, $\tau^1$.

1: for iterations $k = 1, 2, \ldots$ do
2: Server broadcasts $x^k$, $\tau^k$.
3: for Worker $m = 1, \ldots, M$ do
4: Sample stochastic gradients $\{\nabla f_m(x^{k-1}, z^k_m, b)\}_{b=1}^{B^k}$.
5: Compute $\nabla m_k = (1/B^k) \sum_{b=1}^{B^k} \nabla f_m(x^{k-1}, z^k_m, b)$.
6: if $\|\nabla m_k - \hat{\nabla}_m^{k-1}\|^2 > \tau^k$ then
7: Worker $m$ uploads $\nabla m_k$ to the server.
8: Set $\hat{\nabla}_m^k = \nabla m_k$ on the server and worker $m$.
9: else
10: Worker $m$ does not upload.
11: Set $\hat{\nabla}_m^k = \hat{\nabla}_m^{k-1}$ on the server and worker $m$.
12: end if
13: end if
14: Server updates the model $x^k$ via (4) and $\tau^{k+1}$ via (5).
15: end for

**A. CSGD Parameters**

If we choose the parameters properly, our proposed framework is general in the sense that it also recovers several existing algorithms. For the deterministic optimization problem, LAG [31] that sets $w < 1$ and $\sigma^k = 0$ in (5) guarantees communication saving compared with the original gradient descent (i.e., $w = 0$, $\sigma^k = 0$). Note that, in the deterministic case, all data are used at every iteration such that the control size $\sigma^k$ designed to handle the randomness is not necessary and the batch size $B^k$ is no longer a hyperparameter. For the stochastic optimization task, setting $w = 0$ and $\sigma^k = 0$ in (5) recovers the SGD with dynamic batch size [34], [35]. We give some interpretations of the parameters in CSGD as follows (see Table I).

1) **Step Size $\alpha$ and the Batch Size $B^k$:** In recent works [34], [35], SGD with constant step size and exponentially increasing batch size has been studied. It achieves the $O(1/k)$ accuracy with $O(\log k)$ iterations and $O(k)$ samples of stochastic gradients. Intuitively speaking, a larger step size $\alpha$ leads to faster convergence but requires a faster increasing rate of batch size (which depends on $\alpha$) to control the bias from the stochastic gradient sampling. Then, in total, the sampling time is in the same order regardless of the magnitude of $\alpha$. Nonetheless, the choice of $\alpha$ cannot be arbitrary; extremely large step size learns from the current stochastic gradient too much and, thus, deteriorates the convergence.

In our analysis, choosing the increasing rate of $B^k$ larger than a lower bound depending on $\alpha$ will result in a convergence rate depending only on $\alpha$, which is consistent with previous SGD works.

2) **Control Size $\sigma^k$:** The term $\sigma^k$ has two implications.

1) It excludes some noisy uploads. When the worker takes the mean of $B^k$ stochastic gradients, the variance of the mean shrinks to $(1/B^k)$ of the variance of one stochastic gradient if the variance exists. Thus, $\sigma^k$ decreasing no faster than $(1/B^k)$ helps the threshold to make an effect in the long term.

2) As a tradeoff, the control size may slow down the convergence. If it decreases at an extremely slow rate, the censoring threshold will be hard to reach, and the server will use the inaccurate stale gradient for a long time before receiving a fresh gradient, which affects the rate of convergence.

In the theoretical analysis, we will theoretically show that, if $\sigma^k$ decreases properly at a rate similar to those of $(1/B^k)$ and the objective, then CSGD converges at a comparable rate to that of SGD but with improved communication efficiency.

3) **Confidence Time $D$ and Confidence Weight $w$:** Those two parameters bound how much historic information that we leverage in CSGD. First, $D$ is regarded as a confidence time.
Once a newly calculated local gradient is uploaded, we are confident that it will be a good approximation of the gradients in the consecutive $D$ iterations from now on. Therefore, we prefer using it to update variables for no less than $D$ times, instead of uploading a fresher gradient. In fact, the communication-saving property proved in Section III is motivated by the intuition that the upload is as sparse as no more than once in $D$ consecutive iterations. Meanwhile, we multiply a weight $\omega < 1$ to the historic gradients, with the consideration of lessening the impact of historic errors.

Theoretically, we specify $\omega = (1/60)$ to simplify the threshold and constrain the step size and the batch size such that any large $D$ is able to work as a confidence time.

### B. Motivation of the Censzoring Threshold $\tau^k$

For brevity, we stack the random variables into $\xi = [\xi_1; \ldots; \xi_M]$ and define $f(x; \xi) = \sum_{m=1}^M f_m(x; \xi_m)$, $F_m(x) = \mathbb{E}_{\xi_m}[f_m(x; \xi_m)]$, and $F(x) = \sum_{m=1}^M F_m(x)$. The following lemma bounds how much the objectives in CSGD and SGD descend after one update.

**Lemma 1 (Objective Descent):** Suppose that the gradient of the objective function $F(x)$ is $L$-Lipschitz continuous; then, for SGD iteration (2), we have, for any $\tilde{\epsilon} > 0$, that

$$
F(\tilde{x}^k) - F(\tilde{x}^{k-1}) \leq -\alpha \left(1 - \frac{\tilde{\epsilon}}{2} - (1 + \tilde{\epsilon}) \frac{L}{2} \alpha\right) \|\nabla F(\tilde{x}^{k-1})\|^2 \\
+ \frac{\alpha}{2\epsilon} \|\nabla F(\tilde{x}^{k-1}) - \hat{\nabla}^k\|^2 := \bar{\Delta}^k(\tilde{x}^{k-1}, \tilde{\epsilon}).
$$

Likewise, for CSGD iteration (4), we have, for any $\epsilon > 0$, that

$$
F(x^k) - F(x^{k-1}) \leq -\alpha \left(1 - \epsilon - (1 + \epsilon) \frac{L}{2} \alpha\right) \|\nabla F(x^{k-1})\|^2 \\
+ \frac{\alpha}{\epsilon} \|\nabla F(x^{k-1}) - \hat{\nabla}^k\|^2 + \alpha M^2 \left(\frac{1}{\epsilon} + (1 + \frac{1}{\epsilon}) \frac{L}{2} \alpha\right) \tau^k := \Delta^k(x^{k-1}, \epsilon).
$$

Recall the confidence time interpretation of the constant $D$ in (5). Ideally, in CSGD, an uploaded gradient will be used for at least $D$ iterations, and thus, the number of communication reduces to at most $(1/D)$ of the uncensored SGD. At the same time, the objective may descend less in CSGD relative to SGD. Nonetheless, conditioned on the same starting point $\tilde{x}^{k-1} = x^{k-1}$ at iteration $k$, if the objective descents of CSGD and SGD satisfy

$$
-\Delta^k(\tilde{x}^{k-1}, \epsilon) \geq \frac{1}{D}
$$

then CSGD still outperforms SGD in terms of communication efficiency. Equivalently, we write (8) as

$$
\tau^k \leq \frac{1}{M^2} \left(\omega \|\nabla F(x^{k-1})\|^2 + c \|\nabla F(x^{k-1}) - \hat{\nabla}^k\|^2\right) \text{ for any } x \in \mathbb{R}^d,
$$

where

$$
\omega = \frac{1 - \tilde{\epsilon} - (1 + \tilde{\epsilon}) \frac{L}{2} \alpha}{\tilde{\epsilon} + (1 + \frac{1}{\tilde{\epsilon}}) \frac{L}{2} \alpha}
$$

and

$$
c = \frac{1}{1 + \frac{L}{2} \alpha} - \frac{1}{\tilde{\epsilon} + (1 + \frac{1}{\tilde{\epsilon}}) \frac{L}{2} \alpha}
$$

are two constants.

Intuitively, larger $\tau^k$ increases the possibility of censoring communication. However, the right-hand side of (9) is not available at the beginning of iteration $k$ since we know neither $\nabla F(x^{k-1})$ nor $\nabla^k$. Instead, we will approximate $\|\nabla F(x^{k-1})\|^2$ using the aggregated gradients in the recent $D$ iterations, that is,

$$
\|\nabla F(x^{k-1})\|^2 \approx \frac{1}{D} \sum_{d=1}^D \|\hat{\nabla}^{k-d}\|^2.
$$

Further controlling $c \|\nabla F(x^{k-1}) - \hat{\nabla}^k\|^2$ by $\sigma^k$, (9) becomes

$$
\tau^k = \frac{1}{M^2} \left(\omega \cdot \frac{1}{D} \sum_{d=1}^D \|\hat{\nabla}^{k-d}\|^2 + \sigma^k\right)
$$

which leads to the CSGD threshold (5).

### III. THEORETICAL RESULTS

In this section, we study how the introduction of censoring in CSGD affects the convergence and the communication compared to the uncensored SGD. The proofs are given in the Appendixes. Before presenting our theoretical results, we first provide the following sufficient conditions.

**Assumption 1 (Aggregate Function):** The aggregate function $f(x; \xi)$ and its expectation $F(x)$ satisfy the following.

1) **Smoothness:** $\nabla F(x)$ is $L$-Lipschitz continuous.

2) **Bounded Variance:** For any $x \in \mathbb{R}^d$, there exists $G \leq \infty$ such that

$$
\mathbb{E}\|\nabla f(x; \xi) - \nabla F(x)\|^2 \leq G^2.
$$

**Assumption 2 (Local Functions):** Two conditions on the function per worker are given as follows.

1) **Smoothness:** For each $m$, $\nabla F_m(x)$ is $L_m$-Lipschitz continuous.

2) **Bounded Variance:** For any $x \in \mathbb{R}^d$ and $m$, there exists $G_m \leq \infty$ such that

$$
\mathbb{E}\|\nabla f_m(x; \xi_m) - \nabla F_m(x)\|^2 \leq G_m^2.
$$

Notice that Assumption 2 is sufficient for Assumption 1 to hold with $L = \sum_{m=1}^M L_m$, and the independence of $\{\xi_m\}_{m=1}^M$ leads to $G^2 = \sum_{m=1}^M G_m^2$.

**A. Polyak–Łojasiewicz Case**

In the first part, we will assume the Polyak–Łojasiewicz condition [36], which is generally weaker than strong convexity or even convexity.

**Assumption 3 (Polyak–Łojasiewicz Condition):** There exists a constant $\mu > 0$ such that, for any $x$, we have

$$
2\mu (F(x) - F^*) \leq \|\nabla F(x)\|^2
$$

where $F^*$ is the minimum of (1).
Define the Lyapunov function for CSGD as
\[ V^k := F(x^k) - F^* + \sum_{d=1}^{D} \beta_d \| \nabla x^{k-d+1} \|^2 \]
(13)
where \( \{ \beta_d = ((D + 1 - d)/9D)a)^d \}_{d=1}^{D} \) is a set of constant weights. Analogously, the Lyapunov function for uncensored SGD is defined as
\[ \tilde{V}^k := F(x^k) - F^* + \sum_{d=1}^{D} \beta_d \| \tilde{\nabla x}^{k-d+1} \|^2. \]
(14)

The following theorem guarantees the a.s. convergence of CSGD.

**Theorem 1 (a.s. Convergence):** Under Assumptions 1 and 3, if we choose \( w = (1/60), \alpha \leq \min\{(3/2D\mu), (1/3L), (1/(6\sqrt{\max_m L_m MD})\} \). Furthermore, denote \( \rho = (13/\mu a) \), and assume that
\[ B^k \geq B^0(1 - \eta_1)^{-k}, \quad \sigma^k \leq \sigma^0(1 - \eta_2)^k \]
(16)
for some \( \eta_1, \eta_2 > \rho \). Then, conditioned on the same initial point \( x^0 \), we have
\[ E[V^k|x^0] \leq C_{\text{SGD}}(1 - \rho)^k, \quad E[\tilde{V}^k|x^0] \leq C_{\text{SGD}}(1 - \rho)^k \]
(17)
where \( C_{\text{SGD}} = V^0 + ((a(1 - \rho))/3)((10\alpha^0/(\eta_2 - \rho)) + (7G^2/(B^0(\eta_1 - \rho))) \) and \( C_{\text{SGD}} = V^0 + ((7\alpha(1 - \rho)G^2)/(3B^0(\eta_1 - \rho))) \) are two constants.

Theorem 2 implies that, even if CSGD skips some communications, its convergence rate is still in the same order as that of the original SGD. To facilitate the analysis of communication saving, we define \( v \)-iteration complexities and \( v \)-communication complexities as follows.

**Definition 1 (Iteration and Communication Complexities):** The \( v \)-iteration complexities of CSGD and SGD are defined as
\[ K_{\text{CSGD}}(v) = \min\{ k : C_{\text{CSGD}}(1 - \rho)^k \leq v \} \]
(18)
\[ K_{\text{SGD}}(v) = \min\{ k : C_{\text{SGD}}(1 - \rho)^k \leq v \} \]
(19)
respectively. When applying the proposed CSGD algorithm, let \( I_{m,k} \) be the indicator function of worker \( m \) uploading \( \nabla x^k_m \) to the server at time \( k \) (namely, \( I_{m,k} = 1 \) if the upload happens, and \( I_{m,k} = 0 \) otherwise). The \( v \)-communication complexities of CSGD and SGD are defined as
\[ \kappa_{\text{CSGD}}(v) = \sum_{k=1}^{K_{\text{CSGD}}(v)} \sum_{m=1}^{M} I_{m,k} \]
(20)
\[ \kappa_{\text{SGD}}(v) = K_{\text{SGD}}(v)M \]
(21)
respectively.

Briefly speaking, \( v \)-iteration complexity is the number of theoretically required iterations for an algorithm to reach a target accuracy \( v \), while \( v \)-communication complexity is the total number of theoretically required uploads. With these definitions, we will compare the communication complexities of SGD and CSGD in the following theorem.

**Theorem 3 (Communication Saving):** Under Assumptions 2 and 3, set \( w = (1/60), \alpha \leq \min\{(3/2D\mu), (1/3L), (1/(6\sqrt{\max_m L_m MD})\} \). Furthermore, denote \( \rho = (13/\mu a) \), and assume that
\[ B^k \geq B^0(1 - \eta_1)^{-k}, \quad \sigma^k \leq \sigma^0(1 - \eta_2)^k \]
(22)
for some \( \eta_1, \eta_2 > \rho \) and \( B^0 \geq (6M^2(1 - \eta_1) \sum_{m=1}^{M} G_m^2)/((\sigma^0/\eta_2 - \rho)(1 - \eta_2)^2\delta) \), where \( \delta > 0 \) is a given probability. Then, with probability at least \( 1 - \delta \), each worker updates at most once in every \( D \) consecutive iterations. In addition, CSGD will have less communication complexity than SGD with the same step size and batch size, namely, \( \kappa_{\text{CSGD}}(v) < \kappa_{\text{SGD}}(v) \), when
\[ D \geq 2, \quad K_{\text{CSGD}}(v) \geq 4 + \frac{120}{7} \frac{(\eta_1 - \rho)(1 - \eta_1)\rho}{(\eta_2 - \rho)(\eta_1 - \eta_2)(1 - \eta_2)^D} \frac{M^2}{\delta}. \]
(23)

**Remark 1:** Specifically, if we set \( \eta_1 = (1/2) + \rho \) and \( \eta_2 = (1/2) \), then (23) becomes
\[ D \geq 2, \quad K_{\text{CSGD}}(v) \geq 4 + \frac{120}{7} \frac{2^D M^2}{\delta}. \]

Therefore, if \( D \geq 2 \), CSGD saves communication than SGD when the number of iterations is sufficiently large. In addition, from the proof, we observe that, as \( v \to 0 \), approximately CSGD needs at most \( (1/D) \) communication complexity as SGD does (see Remark 4 for more details).

**Remark 2:** The batch-size rule in (22) is commonly used in SGD algorithms with dynamic batch size to guarantee convergence [34], [35]. On the other hand, censoring introduces a geometrically convergent control size \( \sigma^k \), which leads to the same convergence rate as SGD, but provably improves communication efficiency.

In short, Theorem 3 implies that, with high probability, if we properly choose the parameters and run CSGD more than a given number of iterations, then the censoring strategy helps CSGD save communication. Intuitively, a larger \( D \) cuts off more communications, while it slows down the linear rate of convergence since \( \rho = (1/3\mu a) \leq (1/2D) \).

Compared to LAG [31] and LAPG [32], whose objective functions are not stochastic, our convergence results in that Theorem 1 holds in the a.s. sense, and our communication reduction in Theorem 3 is universal. That is to say, with a smaller step size, the heterogeneity characteristic needed to establish communication reduction in LAG and LAPG (see [31, Proposition 1] that depends on the heterogeneity score function) is no longer a prerequisite in our work. Note that, for both CSGD and SGD with dynamic batch size [35], [37, Th. 5.3], the magnitude of step size does not affect the order of the overall number of stochastic gradient calculations to achieve the targeted accuracy. Specifically, since \( \eta_1 \) goes to zero, as shown in Theorem 2, up to iteration \( K \) (the \( v \)-iteration...
complexity of CSGD), the number of needed samples is
\[
\sum_{k=1}^{K} B^k = \sum_{k=1}^{K} B^0 (1 - \eta_1)^{-k} \\
\approx \sum_{k=1}^{K} B^0 (1 - \rho)^{-k} = O((1 - \rho)^{-K}) = O(\nu^{-1})
\]
where the order is regardless of the magnitude of \(\alpha\). Therefore, different from using an optimal (possibly large) step size in existing algorithms, such as LAG and LAPG, it is reasonable to set the step sizes in SGD and CSGD as the same small values, which leads to our universal communication reduction result.

Remark 3: For simplicity, in the proof, we set the constants in Lemma 1 as \(\bar{\varepsilon} = \varepsilon = (1/2)\). Keeping \(\varepsilon = \bar{\varepsilon}\) gives the same linear rate of convergence. Yet, with different values, we may achieve better results, but it is not the main focus here.

B. Nonconvex Case

While CSGD can achieve a linear convergence rate under the Polyak–Łojasiewicz condition, many important learning problems, such as deep neural network training, do not satisfy such a condition. Without Assumption 3, we establish more general results that also work for a large family of nonconvex functions.

Theorem 4 (Nonconvex Case): Under Assumption 2 and the same parameter settings in Theorem 3, then, with probability at least \(1 - \delta\), we have
\[
\min_{1 \leq k \leq K} \| \nabla F(x^k) \|^2 = o\left(\frac{1}{K}\right), \quad \min_{1 \leq k \leq K} \| \hat{\nabla}^k \|^2 = o\left(\frac{1}{K}\right)
\]
and each worker uploads to the server at most once in \(D\) consecutive iterations. As a consequence, CSGD will have less communication complexity than SGD with the same step size and batch size, as long as
\[
D \geq 10, \quad K_{CSGD} \geq 247.
\]

Here, for CSGD and SGD, we evaluate \(v\)-iteration complexities by \(\min_{1 \leq k \leq K} \| \nabla F(x^k) \|^2 \) and \(\min_{1 \leq k \leq K} \| \nabla F(\hat{x}^k) \|^2 \), respectively, and consider their corresponding \(v\)-communication complexities.

IV. NUMERICAL EXPERIMENTS

To demonstrate the merits of our proposed CSGD, especially the two-part design of the censoring threshold, we conduct experiments on four different problems: least squares on a synthetic dataset, softmax regression on the MNIST dataset [38], logistic regression on the Covertype dataset [39], and deep neural network training on the CIFAR-10 dataset [40]. The source codes are written in Python 3.7.4 and available at https://github.com/Weiyu-USTC/CSGD. The experiment of deep neural network training is conducted on a computer with GeForce RTX 2080 Ti GPU, while all other experiments are conducted on a computer with Intel i5 CPU @ 2.3 GHz. We simulate one server and ten workers. To benchmark CSGD, we consider the following approaches.

1) CSGD: Our proposed method with update (4).
2) LAG-S: Directly applying the LAG [31] censoring condition to the stochastic problem, which can be viewed as CSGD with zero control size.
3) SGD: Update (2), which can be viewed as CSGD with censoring threshold 0.

In practice, when the batch size is larger than the number of samples (denoted as \(B\)), a worker can get all the samples; thus, there is no more need for stochastic sampling and averaging. Therefore, in the experiments, unless otherwise specified, the batch size and censoring threshold are calculated via
\[
\begin{align*}
B^k &= \min\left\{ \left[ B^0 (1 - \eta_1)^{-k} \right], \bar{B} \right\} \\
\nu^k &= \frac{w}{M^2 D \alpha^2} \sum_{d=1}^{D} \| x^k - d - x^k - d - 1 \|^2 + \frac{\sigma^0}{M^2} (1 - \eta_2)^k
\end{align*}
\]
where \(B^0, \eta_1, \) and \(\eta_2\) are set the same for all four algorithms, while \(w\) and \(\sigma^0\) are parameters depending on the method that we use. Specifically, \(w\) in CSGD and LAG-S is set as \((1/60)\) according to our theoretical analysis, while \(w = 0\) in SGD and local SGD. The initial control size \(\sigma^0\) is manually tuned to give proper performance in the first few iterations of CSGD and is \(0\) in LAG-S, SGD, and local SGD. In all the experiments, we choose \(D = 10\) since it works for both Polyak–Łojasiewicz and nonconvex cases in the theorems.

We tune the parameters by the following principles. First, choose the step size \(\alpha\) and the batch-size parameters (i.e., \(B^0\) and \(\eta_1\)) that work well for SGD; then, keep them the same in the others. Second, tune the control-size parameters (i.e., \(\sigma^0\) and \(\eta_2\)) to reach a considerable communication saving with the tolerable difference in the convergence with respect to the number of iterations.

A. Least Squares

We first test on the least squares problem, given by
\[
f_m(x; \zeta_m) = \frac{1}{2} \| (\zeta_m^{(1)})^T (x - \bar{x}) + \zeta_m^{(2)} \|^2, \quad x \in \mathbb{R}^{10}. \quad (28)
\]
Therein, entries of \(\zeta_m^{(1)} \in \mathbb{R}^{10}, m \neq 1\), are randomly chosen from the standard Gaussian, while \(\zeta_m^{(1)}\) adds an additional all-1 vector to the standard Gaussian so that the data are heterogeneous. Entries of \(x\) are uniformly sampled from \([-2, 2]\), and \(\zeta_m^{(2)} \in \mathbb{R}\) is a Gaussian noise with distribution \(N(0, 0.01 I^2)\). All values are generated independently. The parameters are set as \(\alpha = 0.2, B^0 = 1, (1 - \eta_1)^{-1} = 1.2, \bar{B} = 100, 000, \sigma_0 = 5,\) and \(1 - \eta_2 = 0.85\), which guarantees that the condition \(\eta_1 > \eta_2\) in Theorem 3 is satisfied. We warm-start all the methods from the starting point generated from 30-step SGD with all-zero initialization. From Fig. 2, we observe that CSGD significantly saves the communication comparing to the other methods.

We also use an intuitive explanation in Fig. 3 to showcase the effectiveness of CSGD on censoring gradient uploads. One blue stick refers to one gradient upload for the corresponding worker at that iteration. The first few iterations in this experiment adjust the initial variable to the point where newly calculated gradients become less informative,
and after that, communication events happen sparsely. Note that the uploads in Fig. 3 are not as sparse as our theorems suggest—no more than one communication event happens in $D$ consecutive iterations since we set $B_0 = 1$ instead of a sufficiently large number. Nevertheless, the design of the batch size and the control size sparsifies the communication and results in the significant reduction of communication cost, as shown in Fig. 2. Besides, a well-designed control size also plays an important role; without the control term, the curve of LAG-S highly overlaps with that of SGD, while CSGD outperforms the other two methods with the consideration of communication efficiency.

B. Softmax Regression

Second, we conduct experiments on the MNIST dataset [38]. The 60,000 training samples are randomly and evenly assigned to the workers. The parameters are set as $\alpha = 0.1$, $B^0 = 1$, $(1 - \eta_1)^{-1} = 1.01$, $\bar{B} = 1,000$, $\sigma^0 = 0.7$, and $1 - \eta_2 = 0.991$. From Fig. 4, the reduction of communication cost in CSGD can be easily observed, with a slightly slower convergence with respect to the number of iterations, which is consistent with the performance in the previous experiment.
It has been rigorously established that our proposed CSGD method achieves the same order of convergence rate as the uncensored SGD, while CSGD guarantees fewer number of gradient uploads if a sufficient number of historic variables are utilized. Numerical tests demonstrated the communication-saving merit of CSGD.

**V. CONCLUSION AND DISCUSSION**

We focused on the problem of communication-efficient distributed machine learning in this article. Targeting higher communication efficiency, we developed a new stochastic distributed optimization algorithm abbreviated as CSGD. By introducing a communication-censoring protocol, CSGD significantly reduces the number of gradient uploads, while it only sacrifices slightly the needed number of iterations.

**APPENDIX A**

**Supporting Lemma**

Lemma 2 (Quasi-Martingale Convergence): Let \((X_n)_{n \geq 0}\) be a nonnegative sequence adapted to a filtration \((\mathcal{F}_n)_{n \geq 0}\). Denote \(D_n = X_n - X_{n-1}\) for any \(n \geq 1\). If

\[
\sum_{n=1}^{\infty} \mathbb{E}[D_n I_{\{D_n | \mathcal{F}_{n-1} \geq 0\}}] = \sum_{n=1}^{\infty} \mathbb{E}[^{\max\mathbb{E}[D_n | \mathcal{F}_{n-1}], 0}] < \infty
\]

where \(I_{\{D_n | \mathcal{F}_{n-1} \geq 0\}}\) is the indicator function of the event \([D_n | \mathcal{F}_{n-1} > 0]\), then there exists a random variable \(X_\infty\) such that, when \(n \to \infty\)

\[
X_n \overset{\text{a.s.}}{\to} X_\infty \geq 0.
\]

**Proof:** Step 1 (Decomposition): We claim that \((X_n)_{n \geq 0}\) can be decomposed into the sum of a submartingale and a supermartingale. The construction is described as follows.

Let \(Y_{n-1} := \mathbb{E}[D_n | \mathcal{F}_{n-1}]\); then, \(I_{Y_{n-1} > 0} \in \mathcal{F}_{n-1}\). Letting \(D_n^+ := D_n I_{Y_{n-1} > 0}\), we have that

\[
\mathbb{E}[D_n^+ | \mathcal{F}_{n-1}] = Y_{n-1} I_{Y_{n-1} > 0} \geq 0.
\]

Similarly, we let \(D_n^- := D_n I_{Y_{n-1} \leq 0}\). Therefore, if we set

\[
X_n^+ = X_{n-1}^+ + D_n^+ \quad X_0^+ = 0
\]

and

\[
X_n^- = X_{n-1}^- + D_n^- \quad X_0^- = X_0
\]

then \(X_n^+, X_n^-\) are submartingale and supermartingale with respect to \(\mathcal{F}_n\), respectively, and \(X_n = X_n^+ + X_n^-\).

Step 2 (Martingale Convergence): From [42, Th. 5.2.8], if \(U_n\) is a submartingale with \(\sup \mathbb{E}[\max U_n, 0] < \infty\), then, as \(n \to \infty\), \(U_n\) converges a.s. to an absolutely integrable limit \(U\). For here, notice that

\[
\mathbb{E}[\max\{X_n^+, 0\}] = \mathbb{E}[\mathbb{E}[\max\{X_n^+, 0\} | \mathcal{F}_{n-1}]]
\]

\[
= \mathbb{E}[\mathbb{E}[\max\{X_n^+, X_{n-1}^+, 0\} | \mathcal{F}_{n-1}]]
\]

\[
\leq \mathbb{E}[\max\{X_n^+, 0\} + \max(Y_{n-1}, 0) | \mathcal{F}_{n-1}]
\]

\[
= \mathbb{E}[\max\{X_n^+, 0\} + \max(\mathbb{E}[D_n | \mathcal{F}_{n-1}], 0)]
\]

which iteratively derives

\[
\mathbb{E}[\max\{X_n^+, 0\}]
\]

\[
\leq \mathbb{E}[\max\{X_0^+, 0\} + \sum_{n=1}^{\infty} \mathbb{E}[\max\{D_n | \mathcal{F}_{n-1}, 0\}] < \infty.
\]

Using the cited martingale convergence theorem, we have \(X_n^+ \overset{\text{a.s.}}{\to} X_\infty\).

On the other hand, the submartingale \(-X_n^-\) is no more than \(X_n^0\) since \(X_n^+ + X_n^- = X_n \geq 0\). Then, we have \(\sup \mathbb{E}[\max\{-X_n, 0\}] = \sup \mathbb{E}[\max\{X_n^+, 0\}] < \infty\). Again, the martingale convergence theorem shows that \(X_n^- \overset{\text{a.s.}}{\to} X_\infty\).
Summing up those two sequences yields the desired convergence. Furthermore, the nonnegativeness of $X_n$ provides that $X_\infty \geq 0$.

\section*{Appendix B}
\textbf{Proof of Lemma 1}

We first prove the CSGD part, and then, the SGD part can be obtained with slight modifications. Notice that

$$\|\nabla^k\|^2 = \left\| \sum_{m=1}^{M} \nabla_m^k \right\|^2 \leq (1 + \epsilon) \left\| \sum_{m=1}^{M} \nabla_m^k \right\|^2 + \left(1 + \frac{1}{\epsilon}\right) \left\| \sum_{m=1}^{M} (\nabla_m^k - \nabla_m^k) \right\|^2 \leq (1 + \epsilon) \left\| \sum_{m=1}^{M} \nabla_m^k \right\|^2 + \left(1 + \frac{1}{\epsilon}\right) M \sum_{m=1}^{M} \tau^k$$

which where the first inequality comes from $\|y + z\|^2 = \|y\|^2 + \|z\|^2 + 2\langle y, z \rangle \leq (1 + \epsilon)\|y\|^2 + (1 + (1/\epsilon))\|z\|^2$ for any $\epsilon > 0$. Second, the inequality $(y, z) \leq (\epsilon/4)\|y\|^2 + (1/\epsilon)\|z\|^2$ gives that

$$\langle \nabla F(x^{(k-1)}), \nabla F(x^{(k-1)}) - \hat{\nabla}^k \rangle$$

$$= \langle \nabla F(x^{(k-1)}), \nabla F(x^{(k-1)}) - \hat{\nabla}^k + \nabla^k - \hat{\nabla}^k \rangle$$

$$= \langle \nabla F(x^{(k-1)}), \nabla F(x^{(k-1)}) - \hat{\nabla}^k \rangle = \frac{\epsilon}{4} \|\nabla F(x^{(k-1)})\|^2 + \frac{1}{\epsilon} \|\nabla F(x^{(k-1)}) - \nabla^k\|^2$$

$$+ \frac{\epsilon}{4} \|\nabla F(x^{(k-1)})\|^2 + \frac{1}{\epsilon} \|\nabla F(x^{(k-1)}) - \hat{\nabla}^k\|^2$$

$$= \frac{\epsilon}{2} \|\nabla F(x^{(k-1)})\|^2 + \frac{1}{\epsilon} \|\nabla F(x^{(k-1)}) - \nabla^k\|^2 + \frac{1}{\epsilon} \|\nabla F(x^{(k-1)}) - \hat{\nabla}^k\|^2 + \frac{1}{\epsilon} \|\nabla F(x^{(k-1)}) - \hat{\nabla}^k\|^2$$

(30)

From (4) and the Lipschitz continuity of $\nabla F$, we have

$$F(x^{(k-1)}) - F(x^{(k-1)} - \alpha \hat{\nabla}^k) = F(x^{(k-1)} - \alpha \hat{\nabla}^k) - F(x^{(k-1)})$$

$$\leq -\alpha \langle \nabla F(x^{(k-1)}), \hat{\nabla}^k \rangle + \frac{L}{2} \alpha^2 \|\hat{\nabla}^k\|^2$$

$$= -\alpha \langle \nabla F(x^{(k-1)}), \nabla F(x^{(k-1)} - \hat{\nabla}^k) \rangle$$

$$+ \frac{L}{2} \alpha^2 \|\hat{\nabla}^k\|^2$$

$$\leq -\alpha \left(1 - \frac{\epsilon}{2} - \frac{(1 + \epsilon) L}{2} \right) \|\nabla F(x^{(k-1)})\|^2$$

$$+ \frac{\alpha M^2}{\epsilon} \left(1 + \frac{1 + \epsilon}{\epsilon} \right) \|\nabla F(x^{(k-1)} - \hat{\nabla}^k\|^2$$

$$+ \frac{\alpha M^2}{\epsilon} \|\nabla F(x^{(k-1)} - \hat{\nabla}^k\|^2$$

where the last inequality uses (29) and (30).

For SGD with update rule (2), (6) can be derived by replacing (30) with

$$\langle \nabla F(x^{(k-1)}) \rangle$$

$$\nabla F(x^{(k-1)}) - \hat{\nabla}^k \rangle$$

$$\leq \frac{\epsilon}{2} \|\nabla F(x^{(k-1)})\|^2 + \frac{1}{2\epsilon} \|\nabla F(x^{(k-1)}) - \hat{\nabla}^k\|^2$$

\section*{Appendix C}

\textbf{Proofs of Theorems 1 and 2}

\textbf{Lemma 3 (Lyapunov Descent):} For $k \geq 1$, denote $x_{k-1} = \sigma(\{x^j, j \leq k - 1\})$ and $\hat{x}_{k-1} = \sigma(\{\hat{x}^j, j \leq k - 1\})$ as the natural sum fields generated by the random vectors before iteration $k - 1$ in CSGD and SGD, respectively. Under Assumptions 1 and 3, if we choose $\beta_d = ((D + 1 - d)/9D)d$, $w = (1/60)$ and the step size $\alpha = \min((32/2D\mu), (1/3L))$, then the Lyapunov functions satisfy

$$\mathbb{E}[V^k - V^{k-1} | x_{k-1}] \leq -\rho V^{k-1} + R^k$$

$$\mathbb{E}[\hat{V}^k - \hat{V}^{k-1} | x_{k-1}] \leq -\rho \hat{V}^{k-1} + \hat{R}^k$$

where $\rho = (1/3)\mu$, $R^k = (10\alpha \sigma^k/3) + (7\alpha G^2/3B^k)$, and $\hat{R}^k = (7\alpha G^2/3B^k)$.

\textbf{Proof:} By convention, define $\beta_{D+1} = 0$. From the definition of $V^k$ in (13) and the inequality in (6)

$$V^k - V^{k-1} = F(x^k) - F(x^{k-1}) + \beta_1 \|\hat{\nabla}^k\|^2 - \sum_{d=1}^{D} (\beta_d - \beta_{d+1}) \|\hat{\nabla}^{k-d}\|^2$$

$$\leq \Delta^k + \beta_1 \|\hat{\nabla}^k\|^2 - \sum_{d=1}^{D} (\beta_d - \beta_{d+1}) \|\hat{\nabla}^{k-d}\|^2.$$ (33)

Plugging in the definition of $\Delta^k$, and taking conditional expectation on $x_{k-1}$ and $\epsilon = \bar{\epsilon} = (1/2)$, hereafter, we have

$$\mathbb{E}[\Delta^k | x_{k-1}] = -\frac{3}{4} a (1 - \bar{\epsilon}) \|\nabla F(x^{(k-1)})\|^2 + \alpha M^2 \left(2 + \frac{3}{2} L a\right) \bar{\epsilon} \xrightarrow{\frac{1}{B^k}} G^2$$

(34)

since $La \leq (1/3)$ and

$$\mathbb{E}[\|\nabla F(x^{(k-1)}) - \hat{\nabla}^k\|^2$$

$$= \frac{1}{(B^k)^2} \mathbb{E} \left[ \sum_{n=1}^{B^k} \|\nabla F(x^{(k-1)}) - \hat{\nabla}^k(x^{(k-1)})\|^2 \right]$$

$$\leq \frac{1}{B^k} G^2.$$ (35)

Furthermore, (35) and $\|\hat{\nabla}^k\|^2 \leq 3(\|\hat{\nabla}^k - \nabla^k\|^2 + \|\nabla^k - \nabla F(x^{(k-1)})\|^2 + \|\nabla F(x^{(k-1)})\|^2)$ give that

$$\mathbb{E}[\|\hat{\nabla}^k\|^2 | x_{k-1}] \leq 3 \left( M^2 \bar{\epsilon} t + \frac{G^2}{B^k} + \|\nabla F(x^{(k-1)})\|^2 \right).$$

Thus, conditioned on $x_{k-1}$, (33) becomes

$$\mathbb{E}[V^k - V^{k-1} | x_{k-1}]$$

$$\leq -\frac{1}{2} a \|\nabla F(x^{(k-1)})\|^2 + 3\alpha M^2 \bar{\epsilon} t + \frac{2a G^2}{B^k}$$

$$+ 3\beta_1 \left( M^2 \bar{\epsilon} t + \frac{G^2}{B^k} + \|\nabla F(x^{(k-1)})\|^2 \right)$$

$$- \sum_{d=1}^{D} (\beta_d - \beta_{d+1}) \|\hat{\nabla}^{k-d}\|^2.$$
which goes to $-\infty$ as $k \to \infty$. Therefore, when $k$ is sufficiently large, $E[V_k | \mathcal{F}_k] \leq 0$ on a set $A$ with positive probability, which is a contradiction. In summary, $V_k \not\geq 0$.

Step 2 (Convergence Rates of $V^k$ and $V^j$): By conditioning on $\mathcal{F}_{k-1}$ first and then conditioning on $\mathcal{F}_0$, Lemma 3 gives an important inequality

$$E[V_k | \mathcal{F}_0] \leq (1 - \rho)E[V_{k-1} | \mathcal{F}_0] + R_k$$

which holds for all $k \geq 1$. Iteratively using (39) yields

$$E[V_k | \mathcal{F}_0] \leq (1 - \rho)^k V_0 + \frac{k}{\rho} \left( 1 - (1 - \rho)^k \right)$$

Similar result holds for SGD that

$$E[V_k | \mathcal{F}_0] \leq (1 - \rho)^k V_0 + \frac{k}{\rho} \left( 1 - (1 - \rho)^k \right)$$

Equations (40) and (41) are exactly (17).

\section*{Appendix D
Proof of Theorem 3}

\textbf{Proof:} From the Markov inequality and (35)

$$E \left[ \sum_{m=1}^{\infty} \left( \| V_{m-1} - V_m \|_{\mathbb{B}^k} \right)^2 > t^k \right] \leq \frac{1}{t^k} E \left[ \sum_{m=1}^{\infty} \| V_{m-1} - V_m \|_{\mathbb{B}^k}^2 \right] \leq \frac{1}{t^k} G_m^2$$

We mainly focus on sample paths where

$$\| V_{m-1} - V_m \|_{\mathbb{B}^k}^2 \leq t^k$$

holds for all $k$ and $m$ with $t^k = (\sigma^{k+D}/6M^2)$. Since

$$\sum_{k=1}^{K} \sum_{m=1}^{\infty} \frac{1}{t^k} G_m^2 \leq \sum_{k=1}^{K} \frac{1}{t^k} G_m^2 \leq \frac{6M^2 \sum_{m=1}^{\infty} G_m^2}{\sigma^2} \leq \delta$$

such a sample path appears with probability at least $1 - \delta$. Suppose that, at iteration $k$, when the worker $m$ decides to upload its latest gradient $\nabla \mathbf{m}$, the most recent iteration that it did communicate with the server is $k - d'$; that is, $\hat{\nabla}_{m-1} = \nabla_{m-k+d'}$. If $1 \leq d' \leq D$, we next prove that it contradicts with the censoring threshold. On the one hand, $\| \nabla_{m-k} - \nabla_{m-k+d'} \|^2 > t^k$ since communication happens at iteration $k$. On the other hand, we have

$$\| \nabla_{m-k} - \nabla_{m-k+d'} \|^2 \leq 3 \left( \| \nabla_{m-k} - \nabla_m \|_{\mathbb{B}^k}^2 + \| \nabla_{m-k} - \nabla_m \|_{\mathbb{B}^k}^2 - \| \nabla_{m-k} - \nabla_{m-k+d'} \|_{\mathbb{B}^k}^2 \right) \leq \frac{3}{2M^2} + \frac{\sigma^2}{2M^2}$$

Iteratively using this fact, we obtain

$$E[V_k - V_{k|\mathcal{F}_0}] \leq -\frac{\alpha}{\rho} \sum_{j=k_0+1}^{k} \alpha + \sum_{j=k_0+1}^{k} R_k$$

where $R_k$ is summable since both $\alpha$ and $(1/B^k)$ are summable. Therefore, from Lemma 2, there exists a random variable $V_x$ such that $V_x \not\geq 0$ on some set $A$ in the probability space with $P(A) > 0$.

To conclude that $V_x \not\geq 0$, we assume $V_x \to V_x \geq e > 0$ on some set $A$ in the probability space with $P(A) > 0$.

For any $a \in A$, there exists an integer $k_0 = k_0(a)$ such that $V_{k-1} \geq (e/2)$ for all $k > k_0$. Then, from Lemma 3

$$E[V_k - V_{k|\mathcal{F}_0}] \leq -\mu V_{k-1} + R_k$$

Iteratively using this fact, we obtain

$$E[V_k - V_{k|\mathcal{F}_0}] \leq -\frac{\alpha}{\rho} \sum_{j=k_0+1}^{k} \alpha + \sum_{j=k_0+1}^{k} R_k$$

such a sample path appears with probability at least $1 - \delta$. Suppose that, at iteration $k$, when the worker $m$ decides to upload its latest gradient $\nabla_m$, the most recent iteration that it did communicate with the server is $k - d'$; that is, $\hat{\nabla}_{m-1} = \nabla_{m-k+d'}$. If $1 \leq d' \leq D$, we next prove that it contradicts with the censoring threshold. On the one hand, $\| \nabla_{m-k} - \nabla_{m-k+d'} \|^2 > t^k$ since communication happens at iteration $k$. On the other hand, we have

$$\| \nabla_{m-k} - \nabla_{m-k+d'} \|^2 \leq 3 \left( \| \nabla_{m-k} - \nabla_m \|_{\mathbb{B}^k}^2 + \| \nabla_{m-k} - \nabla_m \|_{\mathbb{B}^k}^2 - \| \nabla_{m-k} - \nabla_{m-k+d'} \|_{\mathbb{B}^k}^2 \right) \leq \frac{3}{2M^2} + \frac{\sigma^2}{2M^2}$$

Iteratively using this fact, we obtain

$$E[V_k - V_{k|\mathcal{F}_0}] \leq -\frac{\alpha}{\rho} \sum_{j=k_0+1}^{k} \alpha + \sum_{j=k_0+1}^{k} R_k$$

such a sample path appears with probability at least $1 - \delta$. Since all $\alpha$ and $(1/B^k)$ are summable.

Therefore, from Lemma 2, there exists a random variable $V_x$ such that $V_x \not\geq 0$ on some set $A$ in the probability space with $P(A) > 0$.

To conclude that $V_x \not\geq 0$, we assume $V_x \to V_x \geq e > 0$ on some set $A$ in the probability space with $P(A) > 0$.

For any $a \in A$, there exists an integer $k_0 = k_0(a)$ such that $V_{k-1} \geq (e/2)$ for all $k > k_0$. Then, from Lemma 3

$$E[V_k - V_{k|\mathcal{F}_0}] \leq -\frac{\alpha}{\rho} \sum_{j=k_0+1}^{k} \alpha + \sum_{j=k_0+1}^{k} R_k$$

Iteratively using this fact, we obtain

$$E[V_k - V_{k|\mathcal{F}_0}] \leq -\frac{\alpha}{\rho} \sum_{j=k_0+1}^{k} \alpha + \sum_{j=k_0+1}^{k} R_k$$


where $(a)$ comes from $\sigma^{K+D} \leq \sigma^{k-d+D} \leq \sigma^k$ and the Lipschitz continuity of $\nabla F_m$ in Assumption 2, and $(b)$ comes from $\alpha \leq \frac{1}{(6\sqrt{S}L_mMD)}$. The contradiction results in the conclusion that at most one communication happens in $D$ consecutive iterations for every worker. Consequently, the number of communications is at most $\lceil (K/D) \rceil$ after $K$ iterations for worker $m$. Thus, we have

\[
\kappa_{\text{CSGD}}(v) \leq \left[ \frac{K_{\text{CSGD}}(v)}{D} \right] M. \tag{45}
\]

To reach the accuracy $v$, from

\[
\kappa_{\text{SGD}}(1-\rho)^{K_{\text{SGD}}(v)} \leq v \leq \kappa_{\text{SGD}}(1-\rho)^{K_{\text{CSGD}}(v)-1}
\]

the iteration complexity of SGD is lower bounded as

\[
K_{\text{SGD}}(v) > K_{\text{CSGD}}(v) - 1 + (\log(1-\rho))^{-1} \log \left( \frac{C_{\text{CSGD}}}{C_{\text{SGD}}} \right).
\]

Then, in order to have less communication complexity for CSGD, a sufficient condition is that

\[
\left[ \frac{K_{\text{CSGD}}(v)}{D} \right] \leq K_{\text{CSGD}}(v) - 1 + \frac{\log \left( \frac{C_{\text{CSGD}}}{C_{\text{SGD}}} \right)}{\log(1-\rho)}. \tag{46}
\]

Since $\left[ (K_{\text{CSGD}}(v)/D) \right] \leq (K_{\text{CSGD}}(v)/2) + 1$ for $D \geq 2$, (46) holds with

\[
K_{\text{CSGD}}(v) \geq 4 - 2 \frac{\log \left( \frac{C_{\text{CSGD}}}{C_{\text{SGD}}} \right)}{\log(1-\rho)}. \tag{47}
\]

Letting $V^0 = 0$ gives the upper bound of the ratio

\[
\frac{C_{\text{CSGD}}}{C_{\text{SGD}}} \leq 1 + \frac{60}{7} \frac{(\eta_1 - \rho)(1-\eta_1)}{(\eta_2 - \rho)(\eta_1 - \eta_2)(1-\eta_2)} \frac{M^2}{\sigma^k}. \tag{49}
\]

Using the inequalities

\[
\log(1+x) \leq x \text{ and } \log \left( \frac{1}{1-\rho} \right) \leq \frac{\rho}{1-\rho}
\]

the sufficient condition for (47) to hold is that

\[
K_{\text{CSGD}}(v) \geq 4 + \frac{120}{7} \frac{(\eta_1 - \rho)(1-\eta_1)}{(\eta_2 - \rho)(\eta_1 - \eta_2)(1-\eta_2)} \frac{M^2}{\sigma^k}. \tag{50}
\]

This completes the proof. \(\square\)

**Remark 4:** From (46) in the above proof, we observe that the $v$-communication complexities satisfy

\[
\frac{\kappa_{\text{CSGD}}(v)}{\kappa_{\text{SGD}}(v)} \leq \left[ \frac{K_{\text{CSGD}}(v)}{D} \right] M - 1 + \frac{\log \left( \frac{C_{\text{CSGD}}}{C_{\text{SGD}}} \right)}{\log(1-\rho)}.
\]

As $v \to 0$, approximately CSGD only requires at most $1/D$ communication complexity as SGD does.

**APPENDIX E**

**PROOF OF THEOREM 4**

**Proof:**

*Step 1 (a.s. Convergence (Not Necessarily Converging to Zero)):* Without Assumption 3, it is unable to derive Lemma 3, but (36) in its proof still holds, from which we know that

\[
\mathbb{E}[V^k - V^{k-1}|F_{k-1}] \leq R^k \tag{48}
\]

where $R^k$ is summable. Thus, applying Lemma 2 gives the a.s. convergence that $V^k \xrightarrow{a.s.} V^\infty \geq 0$. Hereafter, we focus on the case

\[
\begin{align*}
V^k &\to V^\infty \geq 0 \\
\|\nabla m_k - \nabla F_m(x^{k-1})\|_2^2 &\leq \frac{\sigma^{k+D}}{6D^2} \forall k, m
\end{align*} \tag{49}
\]

which happens with probability at least $1 - \delta$, following the same calculation as in (43).

*Step 2 (Bounds of the Lyapunov Differences):* Without Assumption 3, inequality similar to (36) still holds. The only difference comes from using

\[
\|\nabla k - \nabla F(x^{k-1})\|^2 \leq M^2 \sum_{m=1}^{M} \|\nabla m_k - \nabla F_m(x^{k-1})\|^2 \leq \frac{\sigma^{k+D}}{6}
\]

instead of taking expectations that

\[
\mathbb{E}[\nabla k - \nabla F(x^{k-1})]^2 \leq \frac{G^2}{B^k}.
\]

Thus, replacing $(G^2/B^k)$ in (36) by $(\sigma^{k+D}/6)$ yields

\[
V^k - V^{k-1} \leq -\frac{1}{2} \alpha - 3\beta \left\| \nabla F(x^{k-1}) \right\|^2 - \frac{\alpha}{18D} \sum_{d=1}^{D} \left( \beta_d - \beta_{d+1} - 3(\alpha + \beta_1) \frac{\eta_1}{\eta_2} \right) \left\| \hat{\nabla}^{k-d} \right\|^2 + 3(\alpha + \beta_1) \sigma^k + (2\alpha + 3\beta_1) \frac{\sigma^{k+D}}{6} - \frac{1}{18D} \alpha \left\| \nabla F(x^{k-1}) \right\|^2 - \frac{10}{18} \alpha \sigma^k + \frac{7}{18} \alpha \sigma^{k+D} \tag{50}
\]

where $S^k = (10/3)\alpha \sigma^k + (7/18)\alpha \sigma^{k+D}$ is summable with $\sum_{k=1}^{\infty} S^k \leq (67/18)\alpha \sigma^0 \sum_{k=1}^{\infty} (1 - \eta_2)^k = (67\alpha (1 - \eta_2) \sigma^0) / (18 \eta_2)$. Then, summing (50) from $k = 1$ to $k = K$ gives

\[
\sum_{k=1}^{K} \left\| \nabla F(x^{k-1}) \right\|^2 \leq \frac{\alpha}{18D} \sum_{k=1}^{K} \sum_{d=1}^{D} \left\| \hat{\nabla}^{k-d} \right\|^2 + 67\alpha (1 - \eta_2) \sigma^0 \frac{18 \eta_2}{\alpha} \tag{51}
\]

which implies that

\[
\sum_{k=1}^{K} \left\| \nabla F(x^{k-1}) \right\|^2 \leq \frac{\alpha}{N_{\text{CSGD}}}, \tag{51}
\]
\[ \sum_{k=1}^{K} \| \nabla \hat{f}^k \|^2 \leq \frac{18}{\kappa} N_{\text{CSGD}} \]  
with \( N_{\text{CSGD}} = V^0 + \left((67\alpha(1-\eta_2)\sigma^0)/18\eta_2\right) \). Then, (25) can be derived by finding contradiction if assuming it does not hold.

More generally, for any summable sequence \( \sum a_k < \infty \), if there exists \( e > 0 \) such that \( \min_{K_i < k \leq K_{i+1}} a_k \geq (e/K_i) \) for increasing integers \( \{K_i, i = 1, \ldots, \infty\} \), then \( \sum a_k \geq \min_{K_i < k \leq K_{i+1}} a_k = \infty \) is contradictory to the assumption of summable sequence. Therefore, for any \( e > 0 \), \( \min_{1 \leq k \leq K} a_k < (e/K) \) holds except for finite choices of \( K \), which is equivalent to say \( \min_{1 \leq k \leq K} a_k = o(1/K) \).

Step 3 (Communication Saving): From (51), we have
\[ \min_{0 \leq k \leq K-1} \| \nabla F(x^k) \|^2 \leq \frac{6\eta}{K} N_{\text{CSGD}}. \]
Following the above three steps, SGD analogously satisfies
\[ \min_{0 \leq k \leq K-1} \| \nabla F(x^k) \|^2 \leq \frac{6\eta}{K} N_{\text{SGD}} \]
with \( N_{\text{SGD}} = V^0 + \left((7\alpha(1-\eta_2)\sigma^0)/18\eta_2\right) \).

In CSGD, (45) still holds in this nonconvex case since the observation (44) can also be derived as we focus on the situation
\[ \| \nabla_m - \nabla F(x^{k-1}) \|^2 \leq \frac{\sigma^{k-1} + D}{6M^2} \]
which is the same as in Theorem 3. On the other hand, notice that the iteration complexities in the nonconvex case satisfy
\[ \frac{6N_{\text{SGD}}}{\kappa K_{\text{SGD}}(v)} \leq v \leq \frac{6N_{\text{CSGD}}}{\kappa K_{\text{CSGD}}(v) - 1}. \]
Thus, we have that
\[ K_{\text{SGD}}(v) > \frac{N_{\text{SGD}}}{N_{\text{CSGD}}}(K_{\text{CSGD}}(v) - 1). \]
Then, in order to have less communication complexity, it is sufficient to derive
\[ \left[ \frac{K_{\text{CSGD}}(v)}{D} \right] \leq \frac{N_{\text{SGD}}}{N_{\text{CSGD}}}(K_{\text{CSGD}}(v) - 1). \]  
(53)
Since \( \left[ (K_{\text{CSGD}}(v)/D) \right] < (K_{\text{CSGD}}(v)/D) + 1 \), it suffices to have
\[ \frac{1}{D} < \frac{N_{\text{SGD}}}{N_{\text{CSGD}}}, \quad K_{\text{CSGD}}(v) \geq 1 + \frac{1}{\frac{N_{\text{SGD}}}{N_{\text{CSGD}}}} = \frac{1}{\frac{N_{\text{SGD}}}{N_{\text{CSGD}}}} + 1. \]
Note that
\[ \frac{N_{\text{SGD}}}{N_{\text{CSGD}}} = \frac{V^0 + \frac{7a(1-\eta_2)\sigma^0}{18\eta_2}}{V^0 + \frac{67a(1-\eta_2)\sigma^0}{18\eta_2}} \geq \frac{7}{67}. \]
Then, CSGD saves communication if
\[ D \geq \frac{67}{7} = 10, \quad K_{\text{CSGD}}(v) \geq 1 + \left[ \frac{1 + \frac{7}{10}}{\frac{7}{67} - 1} \right] = 247 \]
which completes the proof. □

Remark 5: From (53) in the above proof and following the same argument as in Remark 4, we observe that the ratio of \( v \)-communication complexities satisfies
\[ \frac{K_{\text{CSGD}}(v)}{K_{\text{SGD}}(v)} \leq \frac{N_{\text{SGD}}}{N_{\text{CSGD}}}(K_{\text{CSGD}}(v) - 1) \]
which is asymptotically \( (N_{\text{SGD}} / D N_{\text{CSGD}}) < 1 \) as \( v \rightarrow 0 \).

REFERENCES

[1] J. Dean et al., “Large scale distributed deep networks,” in Proc. Adv. Neural Process. Syst., Lake Tahoe, NV, USA, 2012, pp. 1223–1231.

[2] J. Konečný, H. B. McMahan, X. F. Yu, P. Richtárik, A. T. Suresh, and D. Bacon, “Federated learning: Strategies for improving communication efficiency,” Oct. 2016, arXiv:1610.05492. [Online]. Available: https://arxiv.org/abs/1610.05492

[3] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, “Communication-efficient learning of deep networks from decentralized data,” in Proc. Int. Conf. Artif. Intell. Statist., vol. 54, Apr. 2017, pp. 1273–1282.

[4] V. Smith, C.-K. Chiang, M. Sanjabi, and S. A. Talwalkar, “Federated multi-task learning,” in Proc. Adv. Neural Process. Syst., Long Beach, CA, USA, Dec. 2017, pp. 4427–4437.

[5] S. Caldas, J. Konečný, H. B. McMahan, and A. Talwalkar, “Expanding the reach of federated learning by reducing client resource requirements,” Dec. 2018, arXiv:1812.07210. [Online]. Available: http://arxiv.org/abs/1812.07210

[6] A. Nedic, A. Olshesky, and M. G. Rabbat, “Network topology and communication-computation tradeoffs in decentralized optimization,” Proc. IEEE, vol. 106, no. 5, pp. 953–976, May 2018.

[7] A. S. Berahas, R. Bollapragada, N. S. Keskar, and E. Wei, “Balancing communication and computation in distributed optimization,” IEEE Trans. Autom. Control, vol. 64, no. 8, pp. 3141–3155, Aug. 2019.

[8] B. G. Giannakis, Q. Ling, G. Mateos, D. I. Schizas, and H. Zhu, “Decentralized learning for wireless communications and networking,” in Splitting Methods in Communication and Imaging, Science and Engineering, New York, NY, USA: Springer, 2016.

[9] Y. Zhang, J. C. Duchi, and M. J. Wainwright, “Communication-efficient algorithms for statistical optimization,” J. Mach. Learn. Res., vol. 14, no. 11, pp. 3321–3363, Nov. 2013.

[10] M. Li, D. G. Andersen, A. J. Smola, and K. Yu, “Communication-efficient distributed machine learning with the parameter server,” in Proc. Adv. Neural Inf. Process. Syst. (NIPS), Montreal, QC, Canada, Dec. 2014, pp. 19–27.

[11] M. I. Jordan, J. D. Lee, and Y. Yang, “Communication-efficient distributed statistical inference,” J. Amer. Stat. Assoc., vol. 114, no. 526, pp. 668–681, Apr. 2019.

[12] T. Yang, “Trading computation for communication: Distributed stochastic dual coordinate ascent,” in Proc. Adv. Neural Inf. Process. Syst., Lake Tahoe, NV, USA, Dec. 2013, pp. 629–637.

[13] M. Jaggi et al., “Communication-efficient distributed dual coordinate ascent,” in Proc. Adv. Neural Process. Syst., Montreal, QC, Canada, Dec. 2014, pp. 3068–3076.

[14] Y. Nesterov, “On an approach to the construction of optimal methods for minimizing smooth convex functions,” Mat. Metody Resheniya Ekon. Zadach, vol. 24, no. 3, pp. 509–517, 1988.

[15] R. Johnson and T. Zhang, “Accelerating stochastic gradient descent using predictive variance reduction,” in Proc. 26th Int. Conf. Neural Inf. Process. Syst., Lake Tahoe, NV, USA, Dec. 2013, pp. 315–323.

[16] A. Elgabli, J. Park, A. S. Bedi, M. Bennis, and V. Aggarwal, “GADMM: Fast and communication efficient framework for distributed machine learning,” J. Mach. Learn. Res., vol. 21, no. 7b, pp. 1–39, 2020.

[17] H. Li, C. Fang, and Z. Lin, “Accelerated first-order optimization algorithms for machine learning,” Proc. IEEE, vol. 108, no. 11, pp. 2067–2082, Nov. 2020.

[18] H. Tang, C. Zhang, S. Gan, T. Zhang, and J. Liu, “Decentralization meets quantization,” Mar. 2018, arXiv:1803.06443. [Online]. Available: http://arxiv.org/abs/1803.06443

[19] M. Rao, S. Rini, and A. Goldsmith, “Distributed convex optimization with limited communications,” Oct. 2018, arXiv:1810.12457. [Online]. Available: http://arxiv.org/abs/1810.12457
H. Karimi, J. Li, R. Tomioika, and M. Vojnovic, “QSGD: Communication-efficient SGD via gradient quantization and encoding,” in Proc. Adv. Neural Process. Syst., Long Beach, CA, USA, Dec. 2017, pp. 1709–1720.

J. Bernstein, Y.-X. Wang, K. Azizzadenesheli, and A. Anandkumar, “SignSGD: Compressed optimisation for non-convex problems,” in Proc. Int. Conf. Mach. Learn., Stockholm, Sweden, Jul. 2018, pp. 559–568.

S. U. Stich, J.-B. Cordonnier, and M. Jaggi, “Sparsified SGD with memory,” in Proc. Adv. Neural Process. Syst., Montreal, QC, Canada, Dec. 2018, pp. 4447–4458.

T. Aji and K. Heafield, “Sparse communication for distributed gradient descent,” in Proc. Conf. Empirical Methods Natural Lang. Process., Copenhagen, Denmark, Sep. 2017, pp. 440–445.

G. Lan, S. Lee, and Y. Zhou, “Communication-efficient algorithms for decentralized and stochastic optimization,” Jan. 2017, arXiv:1701.03961. [Online]. Available: http://arxiv.org/abs/1701.03961

Y. Li, Y. Liu, Z. Tian, and Q. Ling, “COLA: Communication-censored linearized ADMM for decentralized consensus optimization,” in Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP), Brighton, U.K., May 2019, pp. 5237–5241.

Y. Liu, W. Xu, G. Wu, Z. Tian, and Q. Ling, “Communication-censored ADMM for decentralized consensus optimization,” IEEE Trans. Signal Process., vol. 67, no. 10, pp. 2565–2579, May 2019.

M. A. Zinkevich, M. Weimer, A. Smola, and L. Li, “Parallelized stochastic gradient descent,” in Proc. Adv. Neural Process. Syst., Vancouver, BC, Canada, Dec. 2010, pp. 2595–2603.

B. Woodward et al., “Is local SGD better than minibatch SGD? 2 Feb. 2020, arXiv:2002.07839. [Online]. Available: https://arxiv.org/abs/2002.07839

T. Chen, G. Giannakis, T. Sun, and W. Yin, “LAG: Lazily aggregated gradient for communication-efficient distributed learning,” in Proc. Adv. Neural Process. Syst., Montreal, QC, Canada, Dec. 2018, pp. 5050–5060.

T. Chen, K. Zhang, B. G. Giannakis, and T. Basar, “Communication-efficient distributed reinforcement learning,” Dec. 2018, arXiv:1812.03239. [Online]. Available: https://arxiv.org/abs/1812.03239

L. Bottou, “Large-scale machine learning with stochastic gradient descent,” in Proceedings of COMPSTAT, Y. L. G. Saporta, Ed. Heidelberg, Germany: Physica-Verlag HD, 2010, pp. 177–186.

L. Bottou, F. E. Curtis, and J. Nocedal, “Optimization methods for large-scale machine learning,” SIAM Rev., vol. 60, no. 2, pp. 223–311, Jan. 2018.

H. Yu and R. Jin, “On the computation and communication complexity of parallel SGD with dynamic batch sizes for stochastic non-convex optimization,” May 2019, arXiv:1905.04346. [Online]. Available: http://arxiv.org/abs/1905.04346

H. Karimi, J. Nutini, and M. Schmidt, “Linear convergence of gradient and proximal-gradient methods under the Polyak-Lojasiewicz condition,” in Proc. Eur. Conf. Mach. Learn. Knowl. Discovery Databases, Riva del Garda, Italy, 2016, pp. 795–811.

L. Bottou, F. E. Curtis, and J. Nocedal, “Optimization methods for large-scale machine learning,” Jun. 2016, arXiv:1606.04838. [Online]. Available: http://arxiv.org/abs/1606.04838

Y. LeCun, C. Cortes, and C. J. Burges. (1998). The MNIST Dataset. [Online]. Available: http://yann.lecun.com/exdb/mnist

D. Dua and C. Graff, “UCI machine learning repository,” School Inf. Comput. Sci., Univ. California Irvine, Irvine, CA, USA, 2017. [Online]. Available: http://archive.ics.uci.edu/ml

A. Krizhevsky, “Learning multiple layers of features from tiny images,” Univ. Toronto, vol. 1, no. 4, p. 7, May 2012.

K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR), Jun. 2016, pp. 770–778.

R. Durrett, Probability: Theory and Examples, 4th ed. New York, NY, USA: Cambridge Univ. Press, 2010.