Compressed Gradient Methods with Hessian-Aided Error Compensation

Sarit Khirirat, Sindri Magnússon, and Mikael Johansson

Abstract—The emergence of big data has caused a dramatic shift in the operating regime for optimization algorithms. The performance bottleneck, which used to be computations, is now often communications. Several gradient compression techniques have been proposed to reduce the communication load at the price of a loss in solution accuracy. Recently, it has been shown how compression errors can be compensated for in the optimization algorithm to improve the solution accuracy. Even though convergence guarantees for error-compensated algorithms have been established, there is very limited theoretical support for quantifying the observed improvements in solution accuracy. In this paper, we show that Hessian-aided error compensation, unlike other existing schemes, avoids accumulation of compression errors on quadratic problems. We also present strong convergence guarantees of Hessian-based error compensation, and demonstrate that similar convergence improvements are attained when only a diagonal Hessian approximation is used.

I. INTRODUCTION

Large-scale and data-intensive problems in machine learning, signal processing, and control are typically solved by parallel/distributed optimization algorithms. These algorithms achieve high performance by splitting the computation load between multiple nodes that cooperatively determine the optimal solution. In the process, much of the algorithm complexity is shifted from the computation to the coordination. This means that the communication can easily become the main bottleneck of the algorithms, making it expensive to exchange full precision information especially when the decision vectors are large and dense. For example, in training state-of-the-art deep neural network models with millions of parameters such as AlexNet, ResNet and LSTM communication can account for up to 80% of overall training time.

To reduce the communication overhead in large-scale optimization much recent literature has focused on algorithms that compress the communicated information. Some successful examples of such compression strategies are sparsification, where some elements of information are set to be zero [5], and quantization, where information is reduced to a low-precision representation [2], [7]. Algorithms that compress information in this manner have been extensively analyzed under both centralized and decentralized architectures, [2], [6], [7], [8], [9]. These algorithms are theoretically shown to converge to approximate optimal solutions with an accuracy that is limited by the compression precision. Even though compression schemes reduce the number of communicated bits in practice, they often lead to significant performance degradation in terms of both solution accuracy and convergence times, [4], [9], [10], [11].

To mitigate these negative effects of information compression on optimization algorithms, several error compensation strategies have been proposed [4], [12], [13], [11]. In essence, error compensation corrects for the accumulation of many consecutive compression errors by keeping a memory of previous errors. Even though very coarse compressors are used, optimization algorithms using error compensation often display the same practical performance as as algorithms using full-precision information, [4], [12]. Motivated by these encouraging experimental observations, several works have studied different optimization algorithms with error compensation, [13], [1], [11], [14], [10], [15], [16]. However, there are not many theoretical studies which validate why error compensation exhibits better convergence guarantees than direct compression. For instance, Wu et al [11] derived better worst-case bound guarantees of error compensation as the iteration goes on for quadratic optimization. Karimireddy et al [10] showed that binary compression may cause optimization algorithms to diverge, already for one-dimensional problems, but that this can be remedied by error compensation. However, we show in this paper (see Section IV-C) that these methods still accumulate errors, even for quadratic problems.

The goal of this paper is develop a better theoretical understanding of error-compensation in compressed gradient methods. Our key results quantify the accuracy gains of error-compensation and prove that Hessian-aided error compensation removes all accumulated errors on strongly convex quadratic problems. The improvements in solution accuracy are particularly dramatic on ill-conditioned problems. We also provide strong theoretical guarantees of error compensation in stochastic gradient descent methods distributed across multiple computing nodes. Numerical experiments confirm the superior performance of Hessian-aided error compensation over existing schemes. In addition, the experiments indicate that error compensation with a diagonal Hessian approximation achieves
similar performance improvements as using the full Hessian.

Notation and definitions. We let \( \mathbb{N}, \mathbb{N}_0, \mathbb{Z}, \) and \( \mathbb{R} \) be the set of natural numbers, the set of natural numbers including zero, the set of integers, and the set of real numbers, respectively. The set \( \{0, 1, \ldots, T\} \) is denoted by \([0, T]\). For \( x \in \mathbb{R}^d \), \( \|x\| \) and \( \|x\|_1 \) are the \( \ell_2 \) norm and the \( \ell_1 \) norm, respectively, and \( \|x\|_+ = \max\{0, x\} \). For a symmetric matrix \( A \in \mathbb{R}^{d \times d} \), we let \( \lambda_1(A), \ldots, \lambda_d(A) \) denote the eigenvalues of \( A \) in an increasing order (including multiplicities), and its spectral norm is defined by \( \|A\| = \max_i |\lambda_i(A)| \). A continuously differentiable function \( f: \mathbb{R}^d \to \mathbb{R} \) is \( \mu \)-strongly convex if there exists a positive constant \( \mu \) such that

\[
f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|y - x\|^2, \quad \forall x, y.
\]

and \( \mu \)-smooth if

\[
\|\nabla f(y) - \nabla f(x)\| \leq \mu \|y - x\|, \quad \forall x, y \in \mathbb{R}^d.
\]

II. MOTIVATION AND PRELIMINARY RESULTS

In this section, we motivate our study of error-compensated gradient methods. We give an overview of distributed optimization algorithms based on communicating gradient information in §[II-A] and describe a general form of gradient compressors, covering most existing ones, in §[II-B]. Later in §[III] we illustrate the limits of directly compressing the gradient, motivating the need for the error-compensated gradient methods studied in this paper.

A. Distributed Optimization

Distributed optimization algorithms have enabled us to solve large-scale and data-intensive problems in a wide range of application areas such as smart grids, wireless networks, and statistical learning. Many distributed optimization algorithms build on gradient methods and can be categorized based on whether they use a) full gradient communication or b) partial gradient communication; see Figure 1. The full gradient algorithms solve problems on the form

\[
\min_{x} \quad f(x),
\]

by the standard gradient descent iterations

\[
x^{k+1} = x^k - \gamma \nabla f(x^k),
\]

communicating the full gradient \( \nabla f(x^k) \) in every iteration. Such a communication pattern usually appears in dual decomposition methods where \( f(\cdot) \) is a dual function associated with some large-scale primal problem; we illustrate this in subsection [II-A]. The partial gradient algorithms are used to solve separable optimization problems on the form

\[
\min_{x} \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x),
\]

by gradient descent

\[
x^{k+1} = x^k - \frac{2}{n} \sum_{i=1}^{n} \nabla f_i(x^k),
\]

and distributing the gradient evaluation on \( n \) nodes, each responsible for evaluating one of the partial gradients \( \nabla f_i(x) \); see §[II-A2]. Clearly, full gradient communication is a special case of partial gradient communication with \( n = 1 \). However, considering the full gradient communication algorithms separately will enable us to get stronger results in that case. We now review these algorithms separately in more detail.

1) Full Gradient Communication: Dual Decomposition: Resource allocation is a class of distributed optimization problems where a group of \( n \) nodes aim to minimize the sum of their local utility function over a set of shared resource constraints. In particular, the nodes collaboratively solve

\[
\min_{q_1, \ldots, q_n} \sum_{i=1}^{n} U_i(q_i)
\]

subject to

\[
q_i \in Q_i, \quad i = 1, \ldots, n
\]

\[
h(q_1, q_2, \ldots, q_n) = 0.
\]

Each node has a utility function \( U_i(q_i) \) depending on its own private resource allocation \( q_i \), constrained by the set \( Q_i \). The decision variables are coupled through the total resource constraint \( h(q_1, q_2, \ldots, q_n) = 0 \), which captures system-wide physical or economical limitations.

Resource allocation problems arise naturally in wireless networks, data communications, and smart grids. [17], [18], [19]. In data communications we optimize the data flows between \( n \) source-destination pairs through an \( L \)–link communications network by solving the utility minimization problem [7] with

\[
h(q_1, q_2, \ldots, q_n) = \sum_{s \in S_l} q_s - c_t
\]

for \( l \in [1, L] \). Here, \( S_l \) is the set of source-destination pairs that use link \( l \), and \( U_i(\cdot) \) represents the utility of data flow \( i \) to communicate at rate \( q_i \). In electric power systems, where problems on the form of [7] are used to optimize the electricity generation and consumptions of a group of electric devices (e.g., smart meters, household appliances and renewable generators), \( h(q_1, q_2, \ldots, q_n) = 0 \) represents the physics of the grid.
The solution to problems on this form is typically decomposed by considering the dual problem \cite{9, 10, 11, 12, 13, 14}. To illustrate this procedure, we consider the following dual problem which is equivalent to solving \cite{14} (under mild technical conditions \cite{15} chapter 5)

\[
\max_z f(z) := \min_q L(q, x),
\]

In this formulation, \(x\) is the dual variable, \(f(z)\) is the dual objective function, and

\[
L(q, x) = \sum_{i=1}^{n} U_i(q_i) + x^T h(q_1, \ldots, q_n),
\]

is the Lagrangian of Problem \cite{14}. The dual function is concave and the dual gradient (or a dual subgradient) is given by

\[
\nabla f(x) = h(q_1(x), \ldots, q_n(x)), \quad q(x) = \arg \min_q L(q, x).
\]

In many networks the dual gradient is obtained from measurements of the effect of the current decisions. Often, we only get a stochastic version of the gradient denoted by \(g(x, \xi)\) where \(\xi\) is a random variable. If the primal problem has structure, then dual gradient methods can often be used to decompose its solution. For example, in many network applications \(h(q_1(x), \ldots, q_n(x)) = \sum_{i=1}^{n} h_i(q_1(x), \ldots, q_n(x))\). Then, the equivalent dual problem \cite{14} can be solved by the gradient method \cite{14}, leading to the following iteration

\[
q_i^{k+1} = \arg \min_q U_i(q_i^k) + (x^k)^T h_i(q_i^k), \quad i = 1, \ldots, n
\]

\[
x^{k+1} = x^k + \gamma g(q(x; \xi^k))
\]

where \(\gamma\) is a step-size parameter. Notice that the essential step in the algorithm is the communication of the stochastic dual gradient \(g(x^k, \xi^k) \approx h(q_1^{k+1}, \ldots, q_n^{k+1})\) which allows each node \(i\) to update \(q_i^{k+1}\) based on the dual variable \(x^k\). To communicate the gradient it must first be compressed into the finite number of bits. Our results in this paper demonstrate how naïve gradient compression can be improved by an error correction step, leading to significant accuracy improvements.

2) Partial Gradient Communication: Problems on the form of \cite{16} appear, e.g., in machine learning and signal processing where we wish to find optimal estimators based on data from multiple nodes. One important example is empirical risk minimization (ERM) where labelled data is split among \(n\) nodes which collaborate to find the optimal estimate. In particular, if each node \(i \in [1, n]\) has access to its local data with feature vectors \(z_i = (z_{i1}, \ldots, z_{im})\) and labels \(y_i = (y_{i1}, \ldots, y_{im})\) with \(z_{ij} \in \mathbb{R}^d\) and \(y_{ij} \in \mathbb{R}\), then the local objective functions are

\[
f_i(x) = \frac{1}{m} \sum_{j=1}^{m} \ell(x; z_{ij}, y_{ij}) + \frac{\lambda}{2} ||x||^2, \quad i = 1, 2, \ldots, n
\]

where \(\ell(\cdot)\) is some loss function and \(\lambda > 0\) is a regularization parameter. The ERM formulation covers many important machine learning problems. For example, we obtain the least-squares regression problem by letting \(\ell(x; z, y) = (1/2)(y - z^T x)^2\), the logistic regression problem when \(\ell(x; z, y) = \log(1+\exp(-y \cdot z^T x))\), and the support vector machine (SVM) problem if \(\ell(x; z, y) = [1 - y \cdot z^T x]_+\).

When the data set on each node is large, the above optimization problem is typically solved using stochastic gradient decent methods (SGD). In each iteration of distributed SGD, the master node broadcasts a decision variable \(x^k\), while each worker node \(i\) computes a stochastic gradient \(g_i(x^k; \xi_i^k)\) by evaluating its objective function gradient on a random subset of its local data \(D_i\). After the master receives the information from all worker nodes, it can perform the update \(x^{k+1} = x^k - \gamma \frac{1}{n} \sum_{i=1}^{n} g_i(x^k; \xi_i^k)\).

We assume that the stochastic gradient preserves the unbiased-ness and bounded variance assumptions, i.e.

\[
\mathbb{E}_{\xi_i} g_i(x; \xi_i) = \nabla f_i(x), \quad \text{and} \quad \mathbb{E}_{\xi_i} \|g_i(x; \xi_i) - \nabla f_i(x)\|^2 \leq \sigma^2, \quad \forall x \in \mathbb{R}^d.
\]

Notice that unlike \cite{20} Assumption 1 and \cite{21} Assumption 3 this condition only requires similarity between the local gradient and its stochastic oracle but allows for arbitrary differences between the whole data and local data distributions. To save communication bandwidth, worker nodes need to compress stochastic gradients into low-resolution representations. Our results illustrate how the low-resolution gradients can achieve high accuracy solutions by error-compensation. First, we present the compression schemes considered in this paper.

B. Gradient Compression

We consider the following class of gradient compressors.

**Definition 1.** The operator \(Q : \mathbb{R}^d \rightarrow \mathbb{R}^d\) is an \(\epsilon\)-compressor if there exists a positive constant \(\epsilon\) such that

\[
\|Q(v) - v\| \leq \epsilon, \quad \forall v \in \mathbb{R}^d.
\]

Definition 1 only requires bounded magnitude of the compression errors. A small value of \(\epsilon\) corresponds to high accuracy. At the extreme when \(\epsilon = 0\), we have \(Q(v) = v\). An \(\epsilon\)-compressor does not need to be unbiased (in constrast to those considered in \cite{2, 9}) and is allowed to have a quantization error arbitrarily larger than magnitude of the original vector (in constrast to \cite{7} Definition 2.1 and \cite{10} Assumption A). Definition 1 covers most popular compressors in machine learning and signal processing applications, which substantiates the generality of our results later in the paper. One common example is the rounding quantizer, where each element of a full-precision vector \(v_i \in \mathbb{R}\) is rounded to the closest point in a grid with resolution level \(\Delta > 0\)

\[
[Q_{dr}(v)]_i = \text{sign}(v_i) \cdot \Delta \cdot \left\lfloor \frac{|v_i|}{\Delta} + \frac{1}{2} \right\rfloor.
\]

This rounding quantizer is an \(\epsilon\)-compressor with \(\epsilon = d \cdot \Delta^2/4\), \cite{22}, \cite{23}, \cite{29}, \cite{30}. In addition, if gradients are bounded, the sign compressor \cite{1}, the \(K\)-greedy quantizer \cite{6} and the dynamic gradient quantizer \cite{2}, \cite{6} are all \(\epsilon\)- compressors.
III. THE LIMITS OF DIRECT GRADIENT COMPRESSION

To reduce communication overhead in distributed optimization, it is most straightforward to compress the gradients directly. The goal of this section is to illustrate the limits of this approach, which motivates our gradient correction compression algorithms in the next section.

A. Full Gradient Communication and Quadratic Case

A major drawback with direct gradient compression is that it leads to error accumulation. To illustrate why this happens we start by considering convex quadratic objectives

\[ f(x) = \frac{1}{2} x^T H x + b^T x. \]

(14)

Gradient descent using compressed gradients reduces to

\[ x^{k+1} = x^k - \gamma Q(\nabla f(x^k)), \]

(15)

which can be equivalently expressed as

\[ x^{k+1} = (I - \gamma H)x^k - \gamma b + \gamma(\nabla f(x^k) - Q(\nabla f(x^k))). \]

(16)

Hence,

\[ x^{k+1} - x^* = A_\gamma^{k+1}(x^0 - x^*) + \gamma \sum_{j=0}^{k} A_\gamma^{k-j}(\nabla f(x^j) - Q(\nabla f(x^j))). \]

(17)

where \( x^* \) is the optimal solution and the equality follows from the fact that \( Hx^* + b = 0 \). The final term of Equation (17) describes how the compression errors from every iteration accumulate. We show how error compensation helps to remove this accumulation in Section IV. Even though the error accumulates, the compression error will remain bounded if the matrix \( A_\gamma \) is stable (which can be achieved by a sufficiently small step-size), as illustrated in the following theorem.

**Theorem 1.** Consider the optimization problem over the objective function \( f \) where \( H \) is positive definite and let \( \mu \) and \( L \) be the smallest and largest eigenvalues of \( H \), respectively. Then, the iterates \( \{x^k\}_{k \in \mathbb{N}} \) generated by (15) satisfy

\[ \|x^k - x^*\| \leq \rho^k \|x^0 - x^*\| + \frac{1}{\mu} \epsilon, \]

where

\[ \rho = \begin{cases} 1 - 1/\kappa & \text{if } \gamma = 1/L \\ 1 - 2/(\kappa + 1) & \text{if } \gamma = 2/(\mu + L) \end{cases}, \]

and \( \kappa = L/\mu \) is the condition number of \( H \).

**Proof.** See Appendix [B](#appendix-b).

B. Partial Gradient Communication

We now study direct gradient compression in the partial gradient communication architecture. We focus on the distributed compressed stochastic gradient descent algorithm (D-CSGD)

\[ x^{k+1} = x^k - \gamma \frac{1}{n} \sum_{i=1}^{n} Q(g_i(x^k; \xi^k_i)), \]

(18)

where each \( g_i(x; \xi) \) is a partial stochastic gradient sent by worker node \( i \) to the central node. We have the following convergence result analogous to Theorem 1.

**Theorem 2.** Consider an optimization problem \( f \) where each \( f_i(\cdot) \) is \( L \)-smooth, and the iterates \( \{x^k\}_{k \in \mathbb{N}} \) generated by (18) under the assumption that the underlying partial stochastic gradients \( g_i(x^k, \xi^k_i) \) satisfies the unbiased and bounded variance assumptions in Equation (11) and (12). Assume that \( Q(\cdot) \) is the \( \epsilon \)-compressor and \( \gamma < 1/(3L) \).

a) (non-convex problems) Then,

\[ \min_{\epsilon \in [0,k]} \mathbb{E}\|\nabla f(x^\epsilon)\|^2 \leq \frac{1}{k+1} \left( \frac{1}{\gamma^2} \frac{1}{1 - 3L\gamma} (f(x^0) - f(x^*)) + \frac{1 + 3L\gamma}{1 - 3L\gamma} \right) \]

b) (strongly-convex problems) If \( f \) is also \( \mu \)-strongly convex, then

\[ \mathbb{E} (f(x^\epsilon) - f(x^*)) \leq \frac{1}{k+1} + \frac{1}{2} \left( \frac{1}{1 - 3L\gamma} \|x^0 - x^*\|^2 + \frac{1 + 3L\gamma}{2} \frac{1}{1 - 3L\gamma} \epsilon^2 \right) \]

where \( x^\epsilon = \sum_{i=0}^{k} x^i / (k+1) \).

**Proof.** See Appendix [C](#appendix-c).

Theorem 2 establishes a sub-linear convergence of D-CSGD toward the optimum with a residual error depending on the stochastic gradient noise \( \sigma \), compression \( \epsilon \), problem parameters \( \mu, L \) and the step-size \( \gamma \). In particular, the residual error consists of two terms. The first term comes from the stochastic gradient noise \( \sigma^2 \) and decreases in proportion to the step-size. The second term arises from the precision of the compression \( \epsilon \), and cannot diminish towards zero no matter how small we choose the step-size. In fact, it can be bounded by noting that

\[ \frac{1 + 3L\gamma}{1 - 3L\gamma} > 1 \quad \text{and} \quad \frac{1}{2} \frac{1 + 3L\gamma}{2} \frac{1}{1 - 3L\gamma} > \frac{1}{2\mu}. \]
for all $\gamma \in (0, 1/(3L))$. This means that the upper bound in Equation (19) cannot become smaller than $\epsilon^2$ and the upper bound in Equation (20) cannot become smaller than $\epsilon^2/(2\mu)$.

C. Limits of Direct Compression: Lower Bound

We now show that the bounds derived above are tight. 

Example 1. Consider the scalar optimization problem

$$\min_x \frac{\mu}{2} x^2,$$

and the iterates generated by the CGD algorithm

$$x^{k+1} = x^k - \gamma Q(f'(x^k)) = x^k - \gamma Q(x^k),$$

(21)

where $Q(\cdot)$ is the $\epsilon$-compression (see Definition 1)

$$Q(z) = \begin{cases} z - \epsilon \frac{z}{\|z\|} & \text{if } z \neq 0 \\ \epsilon & \text{otherwise}. \end{cases}$$

If $\gamma \in (0, 1/\mu]$ and $|x^0| > \epsilon$ then for all $k \in \mathbb{N}$ we have

$$|x^{k+1} - x^*| = |x^{k+1}| = |x^k - \gamma Q(f(x^k))|$$

$$(1 - \mu\gamma)|x^k| + \gamma \epsilon$$

$$=(1 - \mu\gamma)^k + \gamma \epsilon$$

$$=(1 - \mu\gamma)^k + \frac{1}{\mu\gamma} - (1 - \mu\gamma)^{k+1} + \epsilon/\mu$$

$$\geq \epsilon/\mu,$$

where we have used that $x^* = 0$. In addition,

$$f(x^k) - f(x^*) = \frac{\mu}{2} \sum_{i=0}^k |x^i|^2$$

$$\geq \frac{1}{2\mu} \epsilon^2,$$

where $x^k = \sum_{i=0}^k x^i/(k+1)$.

The above example shows that the $\epsilon$-compressor cannot achieve accuracy better than $\epsilon/\mu$ and $\epsilon^2/(2\mu)$ in terms of $\|x^k - x^*\|^2$ and $f(x^k) - f(x^*)$, respectively. These lower bounds match the upper bound in Theorem 1 and the upper bound in Theorem 2 if the step-size is sufficiently small. However, in this paper we show the surprising fact that an arbitrarily good solution accuracy can be obtained with $\epsilon$-compressor and any $\epsilon > 0$ if we include a simple correction step in the optimization algorithms.

IV. ERROR COMPENSATED GRADIENT COMPRESSION

In this section we illustrate how we can avoid the accumulation of compression errors in gradient-based optimization. In subsection IV-A we introduce our error compensation mechanism and illustrate its powers on quadratic problems. In subsection IV-B we provide a more general error-compensation algorithm and derive associated convergence results. In subsection IV-C we compare our algorithm to existing work.

A. Error Compensation: Algorithm and Illustrative Example

To introduce the error compensation algorithm and show how it avoids the accumulation of compression errors, we again consider the quadratic problem

$$f(x) = \frac{1}{2} x^T H x + b^T x.$$  

The basic idea of the error compensation scheme is to compute the compression error in each iteration and compensate for it in the next search direction. For quadratic problem and full gradient descent, the iterations can be written as

$$x^{k+1} = x^k - \gamma (\nabla f(x^k) + A_\gamma e^k)$$

$$e^{k+1} = \nabla f(x^k) + A_\gamma e^k - Q(\nabla f(x^k) + A_\gamma e^k).$$

(22)

with $e^0 = 0$ and $A_\gamma = I - \gamma H$. This algorithm is similar to the direct gradient compression in Equation (19). However, the main difference is that we have introduced the memory term $e^k$ in the gradient update. The term $e^k$ is essentially the compression error, the difference between the compression input and output. To see how the error correction is helpful, consider the gradient error

$$e^k = \nabla f(x^k) - Q(\nabla f(x^k) + A_\gamma e^k).$$

The compression error can then be re-written as

$$e^{k+1} = e^k + A_\gamma e^k,$$

which reduces to

$$e^k = \sum_{j=0}^{k-1} A_\gamma^{k-1-j} e^j.$$

With this in mind, we can re-write the algorithm step as

$$x^{k+1} = A_\gamma x^k - \gamma b + \gamma c^k$$

and establish that

$$x^{k+1} - x^* = A_\gamma^{k+1} (x^0 - x^*) + \gamma \sum_{i=0}^{k} A_\gamma^{k-i} e^i$$

$$= A_\gamma^{k+1} (x^0 - x^*) + \gamma e^{k+1}.$$  

Notice that here the residual error depends only on the latest compression error $e^{k+1}$, instead of the accumulation of previous compression errors as in Equation (17). In particular, $\|e^{k+1}\| \leq \epsilon$ if $Q(\cdot)$ is an $\epsilon$-compressor and we do not accumulate compression errors. This means that we can recover high solution accuracy given proper step-size tuning. We illustrate this in the following theorem.

Theorem 3. Consider the quadratic optimization problem with objective function (14) where $H$ is positive definite, and let $\mu$ and $L$ be the smallest and largest eigenvalues of $H$, respectively. Then, the iterates $\{x^k\}_{k \in \mathbb{N}}$ generated by (22) with $A^k = I - \gamma H$ and $e^0 = 0$ satisfy

$$\|x^k - x^*\| \leq \rho^k \|x^0 - x^*\| + \gamma \epsilon,$$
where
\[ \rho = \begin{cases} 1 - 1/\kappa & \text{if } \gamma = 1/L \\ 1 - 2/(\kappa + 1) & \text{if } \gamma = 2/(\mu + L) \end{cases}, \]
and \( \kappa = L/\mu. \)

**Proof.** See Appendix D. \( \square \)

Theorem 3 implies that error-compensated gradient descent has linear convergence rate and can attain arbitrarily high solution accuracy by decreasing the step-size. Comparing with Theorem 1 we note that error compensation attains lower residual error than direct compression if we insist on maintaining the same convergence rate. In particular, error compensation in Equation (22) with \( \gamma = 1/L \) and \( \gamma = 2/(\mu + L) \) reduces compression error \( \kappa \) and \((\kappa + 1)/2 \), respectively. Hence, the benefit is especially pronounced for ill-conditioned problems [1]. Finally, Figure 2 shows that our worst-case solution accuracy by decreasing the step-size. Comparing with Theorem 3 implies that error-compensated gradient descent and add a local error compensation term \( e_i \) before applying the \( \epsilon \)-compressor. The master node waits for all compressed \( \bar{x}_k \) to be compressed and add a local error compensation term \( e_i \) before applying the \( \epsilon \)-compressor. The master node waits for all compressed \( \bar{x}_k \) to be compressed.

![Figure 2](image)

**Figure 2:** The performance of CGD (15) and EC-CGD (22) with their theoretical bounds presented in Theorems 1 and 3 for least-squares problems over synthetic data sets with 40,000 data points and 1,000 problem variables. Here, we set the step-size \( \gamma = 1/L \) and the initial point \( x^0 = 0 \).

We next generalize these results to stochastic gradient methods under partial gradient communication architectures.

### B. Partial Gradient Communication

For optimization with partial gradient communication, the natural generalization of error-compensated gradient algorithms consist of the following steps: at each iteration in parallel, worker nodes compute their local stochastic gradients \( g_i(x; \xi_i) \) and add a local error compensation term \( e_i \) before applying the \( \epsilon \)-compressor. The master node waits for all compressed gradients and updates the decision vector by

\[ x^{k+1} = x^k - \gamma \frac{1}{n} \sum_{i=1}^{n} Q(g_i(x^k; \xi_i^k) + A_i e_i^k), \]

while each worker \( i \) updates its memory \( e_i \) according to

\[ e_i^{k+1} = g_i(x^k; \xi_i^k) + A_i e_i^k - Q(g_i(x^k; \xi_i^k) + A_i e_i^k). \]

Similarly as in the previous subsection, we define

\[ A_i^k = I - \gamma H_i^k \]

where \( H_i^k \) is either a deterministic or stochastic version of the Hessian \( \nabla^2 f_i(x^k) \). In this paper, we define the stochastic Hessian in analogous way as the stochastic gradient as follows:

\[ E[H_i^k] = \nabla^2 f_i(x^k), \quad \text{and} \]

\[ E[H_i^k - \nabla^2 f_i(x^k)] \leq \sigma_H^2. \]

Notice that \( H_i^k \) is a local information of worker \( i \). In real implementations, each worker can form the stochastic Hessian and the stochastic gradient independently at random. This algorithm has similar convergence properties as the error compensation for the quadratic problems studied above.

**Theorem 4.** Consider an optimization problem (5) where each \( f_i(\cdot) \) is \( L \)-smooth, and the iterates \( \{x^k\}_{k \in \mathbb{N}} \) generated by (23) with \( A_i \) defined by Equation (25), under the assumptions of stochastic gradients \( g_i(x; \xi_i^k) \) in Equation (11) and (12), and stochastic Hessians \( H_i^k \) in Equation (26) and (27). Assume that \( Q(\cdot) \) is an \( \epsilon \)-compressor and that \( e_i^k = 0 \) for all \( i \in [1, n] \).

a) (non-convex problems) If \( \gamma < 1/(3L) \), then

\[ \min_{x \in [0, k]} E[\nabla f(x)] \leq \frac{1}{k + 1} \cdot \frac{1}{2} \cdot \frac{1}{3L} \cdot \gamma \cdot f(x_0) - \frac{1}{f(x^*)} + \frac{1}{2} \cdot \frac{1}{3L} \cdot \gamma \cdot \sigma^2 + \frac{1}{4L} \cdot \gamma \cdot \epsilon^2, \]

where \( \alpha_2 = L^2 + (2 + 6L) \gamma (\sigma_H^2 + L^2). \)

b) (strongly-convex problems) If \( f \) is also \( \mu \)-strongly convex, and \( \gamma < (1 - \beta)/(3L) \) with \( 0 < \beta < 1 \), then

\[ E(f(\bar{x}^k) - f(x^*)) \leq \frac{1}{k + 1} \cdot \frac{1}{2} \cdot \frac{1}{3L} \cdot \gamma \cdot (\bar{x}^0 - x^*)^2 + \frac{1}{2} \cdot \frac{1}{3L} \cdot \gamma \cdot \sigma^2 + \frac{1}{2} \cdot \frac{1}{3L} \cdot \gamma \cdot \epsilon^2, \]

where \( \alpha_1 = \mu + L/\beta + (4/\mu + 6\gamma) (\sigma_H^2 + L^2) \) and \( \bar{x}^k = \sum_{l=0}^{k} x^l/(k + 1). \)

**Proof.** See Appendix E. \( \square \)

The theorem establishes the convergence of our error-compensation method at the rate \( O(1/k) \) toward the optimum with a residual error. Like Theorem 3 for direct gradient compression, the residual error consists of two terms. The first residual term depends on the stochastic gradient noise \( \sigma^2 \) and the second term depends on the precision of the compression \( \epsilon \). The first term can be made arbitrary small by decreasing the step-size \( \gamma > 0 \), similarly as in Theorem 3. However, unlike in Theorem 3 here we can make the second residual term arbitrarily small by decreasing \( \gamma > 0 \). In particular, for a fixed \( \epsilon > 0 \), the second residual term goes to zero at the rate \( O(\gamma^2) \). This means that in absence of gradient noise (\( \sigma^2 = 0 \)) we can get an arbitrarily high solution accuracy even when the compression resolution \( \epsilon \) is fixed.
C. Comparison With Other Error Compensation Schemes

We now compare our algorithm and results with the other recent works on error compensation, \cite{13, 11, 14, 10, 15, 16}. The error compensation in all of the previous papers keeps in memory the sum (or weighted sum) of all previous compression errors. We can write all these error compensation schemes in the form of Equation (23) and (24) (Section IV-B) by setting $A^k = I$ (or $A^k = \alpha I$ for some $\alpha \in (0, 1)$). If we perform the same convergence analysis on this error compensation as we did for the centralized algorithm for quadratic problems in Section IV-A then we have

$$x^{k+1} - x^* = A^k \gamma (x^k - x^*) + \gamma e^{k+1} - \gamma^2 \sum_{l=0}^{k} A^{k-l} \gamma H e^l,$$

where $A_\gamma = I - \gamma H$. The final term shows that these error compensation schemes do not remove the accumulated quantization errors, even though they have been shown to outperform direct compression. However, our error compensation does remove all of the accumulated error, as shown in Section IV-A. This example shows why Hessian-based error compensation can be more effective than the existing schemes.

We validate the superior performance of Hessian-based error compensation over existing schemes in Section V. To reduce computing and memory requirements, we propose a Hessian approximation (using only the diagonal elements of the Hessian). Error compensation with this approximation is shown to have comparable performance to using the full Hessian.

V. Numerical Results

In this section, we validate the stronger convergence performance of Hessian-aided error compensation than the existing variant in the literature. We also highlight that error compensation with the diagonal Hessian approximation is almost as competitive as the full Hessian. In particular, we evaluated these error compensation schemes on centralized SGD and distributed gradient descent over the linear least-squares SVM classification, which is the minimization problem \cite{5} with each component function on the form

$$f_i(x) = \sum_{j=1}^{m} \ell(x; z_i^j, y_i^j), \quad \text{for} \quad i = 1, \ldots, n.$$ 

Here, $(z_i^1, y_i^1), \ldots, (z_i^m, y_i^m)$ are its associated data samples with feature vectors $z_i^j \in \mathbb{R}^d$ and associated class labels $y_i^j \in \{-1, 1\}$. Throughout all the simulations, we normalized each data sample by its Euclidean norm, and set the initial point $x_0 = 0$. We denoted EC-Vr.1 as the existing error compensation scheme in the literature, EC-Hessian as the Hessian-aided error compensation, and EC-diag-Hessian as the error compensation with the diagonal Hessian approximation. Thus, D-EC-CSGD with EC-Vr.1, EC-Hessian, and EC-diag-Hessian is governed by the iteration according to (23) with $A^k = I$, $A^k = I - \gamma H^k$ and $A^k = I - \gamma \text{diag}(H^k)$, respectively. Here, $\text{diag}(H^k)$ is the diagonal matrix storing only the diagonal elements of the Hessian information $H^k_i$ which is associated with the stochastic gradient $g_i(x^k; \xi^k_i)$.

Consider the deterministic SGD with naive compression cannot reach solution accuracy lower than a certain threshold, and expect-edly has worse performance when the quantization resolution is high (the compression is coarse). Despite slightly worse performance for too coarse compression, error compensation guarantees better convergence guarantees.

Consider the binary (sign) compressor. From Figures 4 and 5 and Table I, almost all variants of error compensation guarantee higher solution accuracy when compressed gradient algorithms run for a sufficiently large number of iterations. In particular, EC-Hessian outperforms other error compensation variants in terms of high convergence speed and low residual error guarantees for centralized SGD and distributed GD. In addition, EC-diag-Hessian has almost the same performance as EC-Hessian.

Table I: Performance comparisons between error compensation schemes on different datasets in terms of the number of epochs to reach a certain sub-optimality $E\{f(x^k) - f^*\}/(f(x^0) - f^*)$. The experiment settings are the same as in Figure 4.

| Dataset       | mushrooms | a9a | ws8a |
|---------------|-----------|-----|------|
|               | sub-optimality | $10^{-4}$ | $10^{-2}$ | $10^{-2}$ |
| EC-Hessian    | 43 epochs | 7 epochs | 0.8 epochs |
| EC-Diag-Hessian | 47 epochs | 7.1 epochs | 0.8 epochs |
| EC-Vr.1       | 42 epochs | 22.6 epochs | 11.3 epochs |

Figure 3: Comparisons of D-CSGD and D-EC-CSGD with one worker node using different compesation schemes for the least-squares SVM classification problems over a3a from \cite{31} when the deterministic rounding quantizer is applied. We set the step-size $\gamma = 0.1/L$, the mini-batch size $b = |D|/20$, where $|D|$ is the total number of data samples.
VI. CONCLUSION

In this paper, we provided a theoretical support for error-compensation in compressed gradient methods. In particular, we showed that optimization methods with Hessian-aided error compensation can, unlike existing schemes, avoid all accumulated compression errors on quadratic problems and provide accuracy gains on ill-conditioned problems. We also provided strong convergence guarantees of Hessian-aided error-compensation for centralized and decentralized stochastic gradient methods on both convex and nonconvex problems. The superior performance of Hessian-based compensation compared to other error-compensation methods was illustrated numerically on classification problems using large benchmark data-sets in machine learning. Our experiments showed that error-compensation with diagonal Hessian approximation can achieve comparable performance as the full Hessian while saving the computational costs.
REFERENCES

[1] S. Khirirat, S. Magnusson, and M. Johansson, “Convergence bounds for compressed gradient methods with memory error compensation,” in ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2019, pp. 2857–2861.

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

[3] Y. Lin, S. Han, H. Mao, Y. Wang, and W. J. Dally, “Deep gradient compression: Reducing the communication bandwidth for distributed training,” arXiv preprint arXiv:1712.01887, 2017.

[4] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, “1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs,” in Fifteenth Annual Conference of the International Speech Communication Association, 2014.

[5] J. Wangni, J. Wang, J. Liu, and T. Zhang, “Gradient sparsification for communication-efficient distributed optimization,” arXiv preprint arXiv:1710.09854, 2017.

[6] Sarit Khirirat, Mikael Johansson, and Dan Alistarh, “Gradient compression for communication-limited convex optimization,” in 2018 IEEE Conference on Decision and Control (CDC), Dec 2018, pp. 166–171.

[7] S. Magnusson, C. Enyioha, N. Li, C. Fischione, and V. Tarokh, “Convergence of limited communications gradient methods,” IEEE Transactions on Automatic Control, 2017.

[8] S. Magnusson, H. Shokri-Ghadikolaei, and N. Li, “On maintaining linear convergence of distributed learning and optimization under limited communication,” arXiv preprint arXiv:1902.11163, 2019.

[9] S. Khirirat, H. R. Feyzmahdavian, and M. Johansson, “Distributed resource allocation using one-way communication with applications to power networks,” in 2016 Annual Conference on Information Science and Systems (CISS). IEEE, 2016, pp. 631–636.

[10] S. Boyd and L. Vandenberghe, Convex optimization. Cambridge university press, 2004.

[11] X. Lian, Y. Huang, Y. Li, and J. Liu, “Asynchronous parallel stochastic gradient for nonconvex optimization,” in Advances in Neural Information Processing Systems, 2015, pp. 2737–2745.

[12] H. R. Feyzmahdavian, A. Aytekin, and M. Johansson, “An asynchronous mini-batch algorithm for regularized stochastic optimization,” IEEE Transactions on Automatic Control, vol. 61, no. 12, pp. 3740–3754, 2016.

[13] M. G. Rabitat and R. D. Nowak, “Quantized incremental algorithms for distributed optimization,” IEEE Journal on Selected Areas in Communications, vol. 23, no. 4, pp. 798–808, 2005.

[14] S. Zhu, M. Hong, and B. Chen, “Quantized consensus adam for multi-agent distributed optimization,” in Acoustics, Speech and Signal Processing (ICASSP), 2016 IEEE International Conference on. IEEE, 2016, pp. 4134–4138.

[15] H. Li, S. De, Z. Xu, C. Studer, H. Samet, and T. Goldstein, “Training quantized nets: A deeper understanding,” in Advances in Neural Information Processing Systems, 2017, pp. 5811–5821.

[16] C. De Sa, M. Lesczynski, J. Zhang, A. Marzoev, C. R. Aberger, K. Olukotun, and C. Ré, “High-accuracy low-precision training,” arXiv preprint arXiv:1803.03383, 2018.

[17] C.-C. Chang and C.-J. Lin, “Libsvm: a library for support vector machines,” ACM transactions on intelligent systems and technology (TIST), vol. 2, no. 3, p. 27, 2011.

[18] Y. Nesterov, Introductory lectures on convex optimization: A basic course. Springer Science & Business Media, 2013, vol. 87.

APPENDIX A

REVIEW OF USEFUL LEMMAS

This section states lemmas which are instrumental in our convergence analysis.

Lemma 1. For \( x_i \in \mathbb{R}^d \) and a natural number \( N \),
\[
\left\| \sum_{i=1}^N x_i \right\|^2 \leq N \sum_{i=1}^N \|x_i\|^2.
\]

Lemma 2. For \( x, y \in \mathbb{R}^d \) and a positive scalar \( \theta \),
\[
\|x + y\|^2 \leq (1 + \theta)\|x\|^2 + (1 + 1/\theta)\|y\|^2.
\]

Lemma 3. For \( x, y \in \mathbb{R}^d \) and a positive scalar \( \theta \),
\[
2\langle x, y \rangle \leq \theta \|x\|^2 + (1/\theta)\|y\|^2.
\]

Lemma 4. Assume that \( f \) is convex and \( L \)-smooth, and the optimum is denoted by \( x^* \). Then,
\[
\|\nabla f(x)\|^2 \leq 2L(f(x) - f(x^*)), \quad \text{for } x \in \mathbb{R}^d.
\]

APPENDIX B

PROOF OF THEOREM [1]

The algorithm in Equation (15) can be written as
\[
x^{k+1} = x^k - \gamma (\nabla f(x^k) + e^k),
\]
where \( e^k = Q(\nabla f(x^k)) - \nabla f(x^k) \). Using that \( \nabla f(x^*) = Hx^* - b \) we have
\[
x^{k+1} - x^* = (I - \gamma H)(x^k - x^*) - \gamma e^k,
\]
or equivalently
\[ x^k - x^* = (I - \gamma H)^k(x^0 - x^*) - \gamma \sum_{i=0}^{k-1} (I - \gamma H)^{k-1-i} \epsilon^i. \] (29)

By the triangle inequality and the fact that for a symmetric matrix \( I - \gamma H \) we have
\[ \| (I - \gamma H)x \| \leq \rho \| x \| \quad \text{for all } x \in \mathbb{R}^d. \]
where \( \rho = \max_{i \in [1,d]} |\lambda_i(I - \gamma H)| = \max_{i \in [1,d]} |1 - \gamma \lambda_i| \)
we have
\[ \|x^k - x^*\| \leq \rho^k \|x^0 - x^*\| + \gamma \sum_{i=0}^{k-1} \rho^{k-1-i} \epsilon. \]

In particular, when \( \gamma = 1/L \) then \( \rho = 1 - 1/\kappa \) meaning that
\[ \|x^k - x^*\| \leq (1 - 1/\kappa)^k \|x^0 - x^*\| + \frac{1}{\mu} \sum_{i=0}^{k-1} (1 - 1/\kappa)^{k-1-i} \epsilon, \]
where \( \kappa = L/\mu. \) Since \( 1 - 1/\kappa \in (0,1) \) we have
\[ \sum_{i=0}^{k-1} (1 - 1/\kappa)^{k-1-i} \leq \sum_{i=0}^{\infty} (1 - 1/\kappa)^i = \kappa, \]
which implies
\[ \|x^k - x^*\| \leq (1 - 1/\kappa)^k \|x^0 - x^*\| + \frac{1}{\mu} \epsilon. \]

Similarly, when \( \gamma = 2/(\mu + L) \) then \( \rho = 1 - 2/(\kappa + 1) \) and
\[ \|x^k - x^*\| \leq (1 - 2/(\kappa + 1))^k \|x^0 - x^*\| + \frac{2}{\mu + L} \sum_{i=0}^{k-1} (1 - 2/(\kappa + 1))^{k-1-i} \epsilon. \]
Since \( 1 - 2/(\kappa + 1) \in (0,1) \) we have
\[ \sum_{i=0}^{k-1} (1 - 2/(\kappa + 1))^{k-1-i} \leq \sum_{i=0}^{\infty} (1 - 2/(\kappa + 1))^i = (\kappa + 1)/2, \]
This means that
\[ \|x^k - x^*\| \leq \left( \frac{\kappa - 1}{\kappa + 1} \right)^k \|x^0 - x^*\| + \frac{1}{\mu} \epsilon. \]

**APPENDIX C**

**PROOF OF THEOREM 2**

We can write the algorithm in Equation (18) equivalently as
\[ x^{k+1} = x^k - \gamma \left( \nabla f(x^k) + \eta^k + e^k \right), \] (30)

where
\[ \eta^k = \frac{1}{n} \sum_{i=1}^{n} [g_i(x^k; \xi_i^k) - \nabla f_i(x^k)], \quad \text{and} \]
\[ e^k = \frac{1}{n} \sum_{i=1}^{n} [Q \left( g_i(x^k; \xi_i^k) \right) - g_i(x^k; \xi_i^k)]. \]

By Lemma 1 the bounded gradient assumption, and the definition of the \( \epsilon \)-compressor we have
\[ \mathbb{E}\|\eta^k\|^2 \leq \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}\|g_i(x^k; \xi_i^k) - \nabla f_i(x^k)\|^2 \leq \sigma^2, \]
\[ \text{and} \]
\[ \|e^k\|^2 \leq \frac{1}{n} \sum_{i=1}^{n} \|Q(g_i(x^k; \xi_i^k)) - g_i(x^k; \xi_i^k)\|^2 \leq \epsilon^2. \] (32)

**A. Proof of Theorem 2(a)**

By the Lipschitz smoothness assumption on \( f(\cdot) \) and Equation (30) we have
\[ f(x^{k+1}) \leq f(x^k) - \gamma \langle \nabla f(x^k), \nabla f(x^k) + \eta^k + e^k \rangle + \frac{L\gamma^2}{2} \|\nabla f(x^k) + \eta^k + e^k\|^2. \]

Due to the unbiased property of the stochastic gradient (i.e. \( \mathbb{E}\eta^k = 0 \)), taking the expectation and applying Lemma 1 and Equation (31) and (32) yields
\[ \mathbb{E}f(x^{k+1}) \leq \mathbb{E}f(x^k) - \left( \gamma - \frac{3L\gamma^2}{2} \right) \mathbb{E}\|\nabla f(x^k)\|^2 + \gamma \mathbb{E}\langle \nabla f(x^k), -e^k \rangle + \frac{3L\gamma^2}{2} (\sigma^2 + \epsilon^2). \]

Next, applying Lemma 2 with \( x = \nabla f(x^k), \) \( y = -e^k \) and \( \theta = 1 \) into the main result yields
\[ \mathbb{E}\|\nabla f(x^k)\|^2 \leq \frac{2}{\gamma (1 - 3L\gamma)} \left( \mathbb{E}f(x^k) - \mathbb{E}f(x^{k+1}) + T \right). \]

Using the fact that
\[ \min_{t \in [0,k]} \mathbb{E}\|\nabla f(x^t)\|^2 \leq \frac{1}{k+1} \sum_{t=0}^{k} \mathbb{E}\|\nabla f(x^t)\|^2 \]
and the cancelations of telescopic series we get
\[ \min_{t \in [0,k]} \mathbb{E}\|\nabla f(x^t)\|^2 \leq \frac{1}{k+1} \gamma \left( \frac{1}{1 - 3L\gamma} - \frac{1}{1 - 3L\gamma} \right) + \frac{2}{\gamma (1 - 3L\gamma)} T. \]

We can now conclude the proof by noting that \( f(x^*) \leq f(x) \) for all \( x \in \mathbb{R}^d \)
B. Proof of Theorem 2)

From the definition of the Euclidean norm and Equation (30),
\[
\begin{align*}
\|x^{k+1} - x^*\|^2 &= \|x^k - x^*\|^2 \\
&\quad - 2\gamma(\nabla f(x^k) + \eta k + e^k, x^k - x^*) \\
&\quad + \gamma^2\|\nabla f(x^k) + \eta k + e^k\|^2.
\end{align*}
\]
(33)

By taking the expected value on both sides and using the unbiasedness of the stochastic gradient, i.e., that
\[
\mathbb{E}[\eta_k] = \frac{1}{n} \sum_{i=1}^{n} \mathbb{E}[g_i(x_k; \xi^k) - \nabla f_i(x^k)] = 0,
\]
and Lemma I and Equation (31) and (32) to get the bound
\[
\|\nabla f(x^k) + \eta k + e^k\|^2 \leq 3\mathbb{E}\|\nabla f(x^k)\|^2 + 3(\sigma^2 + \epsilon^2)
\]
we have
\[
\begin{align*}
\mathbb{E}[x^{k+1} - x^*]^2 &\leq \mathbb{E}[x^k - x^*]^2 \\
&\quad - 2\gamma\mathbb{E}(\nabla f(x^k) + e^k, x^k - x^*) \\
&\quad + 3\gamma^2\mathbb{E}\|\nabla f(x^k)\|^2 + 3\sigma^2 + \epsilon^2).
\end{align*}
\]

Applying Equation (1) with \(x = x^k\) and \(y = x^*\) and using Lemma [4] with \(x = x^k\) we have
\[
\begin{align*}
\mathbb{E}[x^{k+1} - x^*]^2 &\leq (1 - \mu \gamma)\mathbb{E}[x^k - x^*]^2 \\
&\quad - 2\gamma(1 - 3L\gamma)\mathbb{E}[f(x^k) - f(x^*)] \\
&\quad + 2\gamma\mathbb{E}(e^k, x^* - x^k) + 3\mu \gamma(\sigma^2 + \epsilon^2).
\end{align*}
\]

From Lemma 3 with \(\theta = \mu \gamma\) and Equation (32), we have
\[
2\gamma(e^k, x^* - x^k) \leq \mu \gamma\|x^k - x^*\|^2 + \epsilon^2/\mu,
\]
which yields
\[
\begin{align*}
\mathbb{E}[x^{k+1} - x^*]^2 &\leq \mathbb{E}[x^k - x^*]^2 \\
&\quad - 2\gamma(1 - 3L\gamma)\mathbb{E}[f(x^k) - f(x^*)] + T,
\end{align*}
\]
where \(T = \gamma(1/\mu + 3\gamma^2)\sigma^2 + 3\gamma^2\epsilon^2\). By rearranging the terms and recalling that \(\gamma < 1/3L\) we get
\[
\begin{align*}
\mathbb{E}(f(x^k) - f(x^*)) &\leq \frac{1}{2\gamma} \frac{1}{1 - 3L\gamma} \left(\mathbb{E}[x^k - x^*]^2 - \mathbb{E}[x^{k+1} - x^*]^2 + T\right).
\end{align*}
\]

Define \(\tilde{x}^k = \sum_{i=0}^{k} x^i/(k + 1)\). By the convexity of \(f()\) and from the cancelations of the telescopic series we have
\[
\begin{align*}
\mathbb{E}(f(\tilde{x}^k) - f(x^*)) &\leq \frac{1}{k + 1} \sum_{i=0}^{k} \mathbb{E}(f(x^i) - f(x^*)) \\
&\leq \frac{1}{k + 1} \frac{1}{2\gamma} \frac{1}{1 - 3L\gamma} \|x^0 - x^*\|^2 \\
&\quad + \frac{1}{2\gamma} \frac{1}{1 - 3L\gamma} T.
\end{align*}
\]

Hence, the proof is complete.

APPENDIX D

PROOF OF THEOREM 3

We can write the algorithm in Equation (23) equivalently as
\[
x^{k+1} = x^k - \gamma(\nabla f(x^k) - e^k),
\]
where
\[
e^k = \nabla f(x^k) - Q(\nabla f(x^k) + A\gamma e^k), \quad \text{and}
\]
e^k+1 = e^k + A\gamma e^k
and \(A\gamma = I - \gamma H\). Following similar line of arguments as in the proof of Theorem 1 we obtain
\[
x^k - x^* = A\gamma^k x^0 - x^* + \gamma k \sum_{i=0}^{k-1} A\gamma^{k-1-i} e^i.
\]
By using that \(e^k = \sum_{i=0}^{k-1} A\gamma^{k-1-i} e^i\) and \(e^0 = 0\) we get that
\[
x^k - x^* = A\gamma^k x^0 - x^* + \gamma e^k.
\]
Since \(A\gamma\) is symmetric, by the triangle inequality and the fact that \(\|e^k\| \leq \epsilon\) (since \(e^k\) is the compression error) we have
\[
\|x^k - x^*\| \leq \rho \|x^0 - x^*\| + \gamma \epsilon,
\]
where \(\rho = \max_{i \in [1, d]} |1 - \gamma \lambda_i|\). Now following similar arguments as used in the proof of Theorem 1 if \(\gamma = 1/L\) then \(\rho = 1 - 1/\kappa\). Since \(1 - 1/\kappa \in (0, 1)\) we have
\[
\|x^k - x^*\| \leq \left(1 - \frac{1}{\kappa}\right)^k \|x^0 - x^*\| + \frac{1}{L} \epsilon.
\]
If \(\gamma = 2/(\mu + L)\) then \(\rho = 1 - 2/(\kappa + 1)\). Since \(1 - 2/(\kappa + 1) \in (0, 1)\) we have
\[
\|x^k - x^*\| \leq \left(\frac{\kappa - 1}{\kappa + 1}\right)^k \|x^0 - x^*\| + \frac{2}{\mu + L} \epsilon.
\]

APPENDIX E

PROOF OF THEOREM 4

We can write the algorithm in Equation (23) equivalently as
\[
\tilde{x}^{k+1} = \tilde{x}^k - \gamma \left(\nabla f(x^k) + \eta^k\right) - \gamma \frac{1}{n} \sum_{i=1}^{n} (A^k_i - I)e_i^k,
\]
(34)

where
\[
\tilde{x}^k = x^k - \gamma \frac{1}{n} \sum_{i=1}^{n} e_i^k,
\]
\(\eta^k = \frac{1}{n} \sum_{i=1}^{n} (\nabla g_i(x^k; \xi^k) - \nabla f_i(x^k))\).

By Lemma 1 the bounded gradient assumption and by the definition of \(e\)-compressor, it can be proved that
\[
\|\eta^k\|^2 \leq \sigma^2, \quad \text{and}
\]
\[
\|x^k - \tilde{x}^k\|^2 \leq \gamma^2 \sum_{i=1}^{n} \|e_i^k\|^2/n \leq \gamma^2 \epsilon^2.
\]
A. Proof of Theorem 4.4

Before deriving the main result we prove two lemmas that are need in our analysis.

**Lemma 5.** Assume that $\|e_i^k\| \leq \epsilon$, and that the Hessian $H_i^k$ satisfies the unbiased and bounded variance assumptions described in Equation (26) and (27). If $\nabla^2 f_i(x) \preceq LI$ for $x \in \mathbb{R}^d$, then

$$E \left\| \frac{1}{n} \sum_{i=1}^{n} H_i^k e_i^k \right\|^2 \leq 2 \gamma^2 (\sigma_H^2 + L^2) \epsilon^2, \quad \text{for } k \in \mathbb{N}. \quad (37)$$

**Proof.** By Lemma 1 we have

$$E \left\| \frac{1}{n} \sum_{i=1}^{n} H_i^k e_i^k \right\|^2 \leq 2 \gamma^2 \frac{1}{n} \sum_{i=1}^{n} E \| [H_i^k - \nabla^2 f_i(x)] e_i^k \|^2 + 2 \gamma^2 \frac{1}{n} \sum_{i=1}^{n} E \| \nabla^2 f_i(x) e_i^k \|^2.$$  

Since $H_i^k - \nabla^2 f_i(x)$ is symmetric, using Equation (27), and the fact that $\nabla^2 f_i(x) \preceq LI$ and that $\|e_i^k\| \leq \epsilon$ yields

$$E \left\| \frac{1}{n} \sum_{i=1}^{n} H_i^k e_i^k \right\|^2 \leq 2 \gamma^2 (\sigma_H^2 + L^2) \epsilon^2. \quad \square$$

**Lemma 6.** If $f(\cdot)$ is strongly convex, then for $\theta_1 > 0$

$$-\langle \nabla f(x^k), \tilde{x}^k - x^* \rangle \leq - (f(x^k) - f(x^*)) - \frac{\mu}{4} \| \tilde{x}^k - x^* \|^2 \quad \frac{1}{2} \left( \mu + \frac{1}{\theta_1} \right) \| \tilde{x}^k - x^* \|^2 \quad \frac{\theta_1}{2} \| \nabla f(x^k) \|^2. \quad (38)$$

**Proof.** By using the strong convexity inequality in Equation 1 with $x = x^k$ and $y = x^*$ we have

$$-\langle \nabla f(x^k), \tilde{x}^k - x^* \rangle \leq - (f(x^k) - f(x^*)) - \frac{\mu}{4} \| \tilde{x}^k - x^* \|^2.$$

Using the fact that $\|x+y\|^2 \leq 2\|x\|^2 + 2\|y\|^2$ with $x = x^k - x^*$ and $y = \tilde{x}^k - x^k$, we have

$$-\| x^k - x^* \|^2 \leq - \frac{1}{2} \| \tilde{x}^k - x^* \|^2 + \| x^k - \tilde{x}^k \|^2.$$  

Combining these inequalities yields

$$-\langle \nabla f(x^k), x^k - x^* \rangle \leq -(f(x^k) - f(x^*)) \quad - \frac{\mu}{4} \| \tilde{x}^k - x^* \|^2 + \frac{\mu}{2} \| x^k - \tilde{x}^k \|^2. \quad (39)$$

Next, by Lemma 3

$$-\langle \nabla f(x^k), \tilde{x}^k - x^k \rangle \leq \frac{1}{\theta_1} \| x^k - \tilde{x}^k \|^2 + \frac{\theta_1}{2} \| \nabla f(x^k) \|^2,$$

for $\theta_1 > 0$. Summing Equation (39) and (40) completes the proof. \square

By using the $L$-smoothness of $f(\cdot)$ and Equation (44) with $A_i^k$ defined by Equation (25) we have

$$f(x^{k+1}) \leq f(\tilde{x}^k) - \gamma (\nabla f(\tilde{x}^k), \nabla f(x^k) + \eta_k) + \frac{\gamma}{2} \left\| \nabla f(x^k) + \eta_k - \gamma x_k \right\|^2 + \frac{L \gamma^2}{2} \left\| \nabla f(x^k) + \eta_k - \gamma x_k \right\|^2. \quad (42)$$

By the unbiased property of the stochastic gradient in Equation (11), and by applying Lemma 3 with $\theta = 1$ and Lemma 1 we get

$$E f(x^{k+1}) \leq E f(\tilde{x}^k) - \gamma E \langle \nabla f(\tilde{x}^k), \nabla f(x^k) \rangle + \left( \frac{\gamma}{2} + \frac{3L \gamma^2}{2} \right) E \| \nabla f(x^k) \|^2 + \frac{3L \gamma^2}{2} E \| \eta_k \|^2$$

$$+ \left( \frac{\gamma}{2} + \frac{3L \gamma^2}{2} \right) E \left\| \frac{1}{n} \sum_{i=1}^{n} H_i^k e_i^k \right\|^2 \quad (43)$$

Since each $f_i(\cdot)$ is $L$-smooth, $\nabla^2 f_i(x) \preceq LI$ for $x \in \mathbb{R}^d$. Applying the bounds in Equation (35) and (37) yields

$$E f(x^{k+1}) \leq E f(\tilde{x}^k) - \left( \frac{\gamma}{2} - \frac{3L \gamma^2}{2} \right) E \| \nabla f(x^k) \|^2 + \frac{\gamma}{2} E \| \nabla f(x^k) - \nabla f(x^k) \|^2 + T,$$

where $T = 3L \gamma^2 \sigma_H^2 / 2 + (1 + 3L \gamma) (\sigma_H^2 + L^2) \gamma^2 \epsilon^2$. Using that

$$-2(x, y) = - \| x \|^2 - \| y \|^2 + \| x - y \|^2 \quad \text{for all } x, y \in \mathbb{R}^d,$$

we have

$$E f(x^{k+1}) \leq E f(\tilde{x}^k) - \left( \frac{\gamma}{2} - \frac{3L \gamma^2}{2} \right) E \| \nabla f(x^k) \|^2 + \frac{\gamma}{2} E \| \nabla f(x^k) - \nabla f(x^k) \|^2 + T.$$  

By the Lipschitz continuity assumption of $\nabla f(\cdot)$, and by (36),

$$E f(x^{k+1}) \leq E f(\tilde{x}^k) - \left( \frac{\gamma}{2} - \frac{3L \gamma^2}{2} \right) E \| \nabla f(x^k) \|^2 + \bar{T},$$

where $\bar{T} = T + L^2 (\gamma^3 / 2) \epsilon^2$. By rearranging the terms and recalling that $\gamma < 1 / (3L)$ we get

$$E \| \nabla f(x^k) \|^2 \leq \frac{2}{\gamma (1 - 3L \gamma)} (E f(\tilde{x}^k) - E f(x^{k+1}) + \bar{T}). \quad (46)$$

Since min$_{i \in [0:k]} E \| \nabla f(x^k) \|^2 \leq \sum_{i=0}^{k} E \| \nabla f(x^k) \|^2 / (k + 1)$, we have

$$\min_{i \in [0:k]} E \| \nabla f(x^k) \|^2 \leq \frac{1}{k + 1} \frac{1}{\gamma (1 - 3L \gamma)} (E f(\tilde{x}^0) - E f(x^{k+1}) + \bar{T}).$$

By the fact that $e^0 = 0$ (i.e. $\tilde{x}^0 = x^0$), that $f(x) \geq f(x^*)$ for $x \in \mathbb{R}^d$ we complete the proof.
B. Proof of Theorem 4-b)

From Equation (34) with $A_k^i$ defined by Equation (25) we have
\[
\|\tilde{x}^{k+1} - x^*\|^2 = \|\tilde{x}^k - x^*\|^2 - 2\gamma(\nabla f(x^k) + \eta^k, \tilde{x}^k - x^*) + 2\gamma \left( \gamma \frac{1}{n} \sum_{i=1}^n H_i^k e_i, \tilde{x}^k - x^* \right) + \gamma^2 \left\| \nabla f(x^k) + \eta^k - \gamma \frac{1}{n} \sum_{i=1}^n H_i^k e_i \right\|^2.
\]

By the unbiasedness of the stochastic gradient described in Equation (11), by Lemma 1, by Lemma 3 with $\theta = \mu/2$ and by the bound in Equation (35) we have
\[
E\|\tilde{x}^{k+1} - x^*\|^2 \leq \left( 1 + \frac{\mu^2}{2} \right) E\|\tilde{x}^k - x^*\|^2 - 2\gamma E(\nabla f(x^k), \tilde{x}^k - x^*) + \left( \frac{2\gamma}{\mu} + 3\gamma^2 \right) E\left\| \gamma \frac{1}{n} \sum_{i=1}^n H_i^k e_i \right\|^2 + 3\gamma^2 E\|\nabla f(x^k)\|^2 + 3\gamma^2 \sigma^2.
\]

Since each $f_i(\cdot)$ is $L$-smooth, $\nabla^2 f_i(x) \leq LI$ for $x \in \mathbb{R}^d$ so we can apply Lemma 6. From Equation (36) in Lemma 5 and Equation (35) in Lemma 6 with $\theta_1 = \beta/L$ we have
\[
E\|\tilde{x}^{k+1} - x^*\|^2 \leq E\|\tilde{x}^k - x^*\|^2 - 2\gamma E(f(x^k) - f(x^*)) + \left( \frac{\beta \gamma}{L} + 3\gamma^2 \right) E\|\nabla f(x^k)\|^2 + \bar{T},
\]

where
\[
\bar{T} = \left( \mu + \frac{L}{\beta} + \left( \frac{4}{\mu} + 6\gamma \right) (\sigma_H^2 + L^2) \right) \gamma^3 e^2 + 3\gamma^2 \sigma^2.
\]

By Lemma 4 we have
\[
E\|\tilde{x}^{k+1} - x^*\|^2 \leq E\|\tilde{x}^k - x^*\|^2 - 2\alpha \gamma E(f(x^k) - f(x^*)) + \bar{T},
\]

where $\alpha = 1 - \beta - 3\gamma L$. By recalling that $\gamma < (1 - \beta)/(3L)$ and $\beta \in (0, 1)$ then
\[
E(f(x^k) - f(x^*)) \leq \frac{1}{2\alpha \gamma} (E\|\tilde{x}^k - x^*\|^2 - E\|\tilde{x}^{k+1} - x^*\|^2 + \bar{T}).
\]

Define $\tilde{x}^k = \sum_{i=0}^k x^i/(k + 1)$. By the convexity of $f(\cdot)$ and the cancelations in the telescopic series we have
\[
E(f(\tilde{x}^k) - f(x^*)) \leq \frac{1}{k + 1} \sum_{i=0}^k E(f(x^i) - f(x^*)) \leq \frac{1}{k + 1} \frac{1}{2\alpha \gamma} E\|\tilde{x}^0 - x^*\|^2 + \frac{1}{2\alpha \gamma} \bar{T}.
\]

By the fact that $e^0 = 0$ (i.e. $\tilde{x}^0 = x^0$), the proof is complete.