### Introduction

The training of high-dimensional federated learning models [1] reduces to solving an optimization problem of the form

$$x^* = \arg\min_{x \in \mathbb{R}^d} f(x) = \frac{1}{n} \sum_{i=1}^n f_i(x),$$  \hspace{1cm} (1)

where $n$ is the number of consumer devices (e.g., mobile devices), $d$ is the number of parameters/features of the model, and $f_i : \mathbb{R}^d \rightarrow \mathbb{R}$ is a loss function that depends on the data stored on the $i$th device. The simplest benchmark method used for the solution of problem (1) is Gradient Descent, which proceeds in iterates of the form

$$x_{k+1} = \frac{1}{n} \sum_{i=1}^n f_i'(x_k),$$

More sophisticated methods used in Federated Learning take multiple local steps instead of a single step before averaging the resultant iterates, or compress the iterates before averaging [2], and the latter practice is the starting point of our work.

### Gradient Descent with Compressed Iterates (GDCI)

**Algorithm 1** Gradient Descent with Compressed Iterates

**Input:** Step size $\gamma > 0$, initial vector $x_0$

1. for $t = 0, 1, \ldots$ do
2. Compute a stochastic compression of iterate $x_t$ as $\tilde{C}(x_t)$.
3. Take a gradient descent step from the compressed iterate
   $$x_{t+1} = \tilde{C}(x_t) - \gamma \nabla f(C(x_t)).$$
4. end for

Clearly, if we are to tackle iterative model compression as used in more complex distributed optimization settings we must understand it when used on a single node as in Algorithm 1.

### Assumptions

**Smoothness and Convexity:** The function $f$ is $L$-smooth and $\mu$-strongly convex. That is, there exists $L > \mu > 0$ such that for all $x, y \in \mathbb{R}^d$ we have:

$$f(y) + \nabla f(y) \cdot (x - y) + \frac{\mu}{2} \|x - y\|^2 \leq f(x),$$

and $f(x) < f(y) + \nabla f(y) \cdot (x - y) + \frac{L}{2} \|x - y\|^2$. We define the condition number of $f$ as $\kappa = L/\mu$.

### Convergence of Gradient Descent with Compressed Iterates (GDCI)

**Theorem 1**

For GDCI run with a constant stepsize $\gamma > 0$ such that $\gamma < \frac{1}{L}$ and a compression coefficient $\omega > 0$ that satisfies

$$\omega = \frac{\mu}{L} \leq \frac{1 - 2\gamma L}{2\gamma L^2 + \frac{1}{2} + L - \mu}.$$  

Then,

$$\mathbb{E} \left[ \|x_t - x^*\|^2 \right] \leq (1 - \gamma \mu)^t \|x_0 - x^*\|^2 + \frac{2\omega}{\mu} \left( \frac{4\gamma L^2}{\gamma} + \frac{4}{\gamma} + L - \mu \right) \|x_0 - x^*\|^2.$$  

Using a specific stepsize choice, we can gain more insight into the convergence rate given by Theorem 1.

**Corollary 1**

Choose $\gamma = \frac{1}{L} \kappa$ and suppose that $\omega < \frac{1}{\kappa^2}$, then we have

$$\mathbb{E} \left[ \|x_t - x^*\|^2 \right] \leq \left( 1 - \frac{1}{4\kappa} \right)^t \|x_0 - x^*\|^2 + 2\omega (18\kappa - 1) \|x_0 - x^*\|^2.$$  

This is the same rate as gradient descent, but only to a $O(\kappa^2)$ neighboorhood (in squared distances) of the solution.

### More on the convergence of GDCI

While the analysis shows that GDCI only converges to a neighbourhood, in many cases when only an approximate solution is desired this is acceptable. However, if we desire to set the neighbourhood to be $O(1)$, then we should have $\omega - O(\kappa^{-2})$. While this seems to be a pessimistic bound on the compression level possible, we note that in practice compression is done only intermittently (this could be modelled by an appropriate choice of $C$) or combined with averaging (which naturally reduces the variance associated with quantization).

In practical situations where averaging is not performed, such as the quantization of server-to-client communication, high compression levels do not seem possible without serious deterioration of the accuracy of the solution [2], and our experiments also suggest that this is the case.

### Experimental Results

We experiment with a logistic regression problem on two different datasets. We consider the random sparsification operator, where each coordinate is independently set to zero according to some given probability. To model intermittent quantization experimentally, we apply the quantization operator $C$ with probability 1/10 and keep the iterate as it is with probability 9/10.

![Figure 2](image.png)

Figure 2: GDCI as $\omega$ varies for two different regularized logistic regression problems. Red star indicates $C$ was applied in that iteration.

### References
