EF21: A New, Simpler, Theoretically Better, and Practically Faster Error Feedback

Peter Richtárik [1] Igor Sokolov [1] Ilyas Fatkhulin [1, 2]

[1] KAUST [2] TU Munich

The problem

We are interested in solving the *nonconvex distributed optimization problem*

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

where $x \in \mathbb{R}^d$ represents the parameters of a machine learning model we wish to train, n is the number of workers/nodes/machines, and $f_i(x)$ is the loss of model x on the data stored on node i.

Assumptions

We assume throughout that $f^{\inf} := \inf_{x\in\mathbb{R}^d} f(x) > -\infty$.

Assumption 1 (Lipschitz gradient). *Every f_i has L_i-Lipschitz gradient, i.e.,*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \le L_i\|x-y\| \quad \forall x, y \in \mathbb{R}^d.$$

Assumption 2 (Polyak-Lojasiewicz). *There exists $\mu > 0$ such that $f(x) - f(x^\star) \le \frac{1}{2\mu}\|\nabla f(x)\|^2$ for all $x \in \mathbb{R}^d$, where $x^\star = \arg\min f$.*

Contractive compressors

Contractive compressor: A (possibly randomized) map $\mathcal{C} : \mathbb{R}^d \to \mathbb{R}^d$ is called a *contractive compressor* if there exists a constant $0 < \alpha \le 1$:

$$\mathbb{E}\left[\|\mathcal{C}(x) - x\|^2\right] \le (1-\alpha)\|x\|^2, \qquad \forall x \in \mathbb{R}^d. \tag{2}$$

Example: Top-k (greedy) sparsification operator keeps the k largest entries of x in absolute value, and zeros out the rest. This is a biased contractive compressor with $\alpha = \frac{k}{d}$.

Main goal

Design an efficient **distributed** first-order method that works naturally with **contractive compressors** (which can be biased!) and relies on **standard assumptions** only.

Generic method

Consider the generic first-order method

$$x^{t+1} = x^t - \frac{\gamma}{n}\sum_{i=1}^n g_i^t, \tag{3}$$

where $\gamma > 0$ is a learning rate, and g_i^t is an easy-to-communicate (i.e., compressed) approximation of $\nabla f_i(x^t)$.

How to construct the estimators g_i^t in the generic method?

- Naive idea
- Good but non-implementable idea
- Good and implementable idea

Naive idea

Simply use the compressed gradient

$$g_i^t = \mathcal{C}\left(\nabla f_i(x^t)\right)$$

Advantages: • Conceptually easy

Problems: • $\mathbb{E}\left[\|g_i^t - \nabla f_i(x^t)\|^2\right] \nrightarrow 0$ as $t \to \infty$
• As a result, can diverge for $n > 1$ [1]

Good but non-implementable idea

Assume that we know $\nabla f_i(x^*)$ and use

$$g_i^t = \nabla f_i(x^*) + \mathcal{C}\left(\nabla f_i(x^t) - \nabla f_i(x^*)\right)$$

Advantages: • $\mathbb{E}\left[\|g_i^t - \nabla f_i(x^t)\|^2\right] \to 0$ as $t \to \infty$
• As a result, converges for all $n \ge 1$

Problems: • Not implementable since we do not know $\nabla f(x^*)$

Good and implementable idea

Define recursively a **Markov compressor**:

$$g_i^0 = \mathcal{C}\left(\nabla f_i\left(x^0\right)\right), \quad g_i^{t+1} = g_i^t + \mathcal{C}\left(\nabla f_i\left(x^{t+1}\right) - g_i^t\right)$$

Advantages: • Easy to implement
• $\mathbb{E}\left[\|g_i^t - \nabla f_i(x^t)\|^2\right] \to 0$ as $t \to \infty$
• As a result, converges for all $n \ge 1$
• Fast convergence in theory and practice

EF21 = Generic method + Markov compressor

Algorithm 1: EF21 (Error Feedback version 2021)

Input: $x^0 \in \mathbb{R}^d$; $\gamma > 0$; $g_i^0 = \mathcal{C}(\nabla f_i(x^0))$ for $i = 1, \dots, n$ (known by nodes and the master); $g^0 = \frac{1}{n}\sum_{i=1}^n g_i^0$ (known by master)
for $t = 0, 1, \dots, T-1$ **do**
  Master computes $x^{t+1} = x^t - \gamma g^t$ and broadcasts x^{t+1} to all nodes
  for *all nodes $i = 1, \dots, n$ in parallel* **do**
    Compress $c_i^t = \mathcal{C}(\nabla f_i(x^{t+1}) - g_i^t)$ and send c_i^t to the master
    Update local state $g_i^{t+1} = g_i^t + \mathcal{C}(\nabla f_i(x^{t+1}) - g_i^t)$
  end
  Master computes $g^{t+1} = \frac{1}{n}\sum_{i=1}^n g_i^{t+1}$ via $g^{t+1} = g^t + \frac{1}{n}\sum_{i=1}^n c_i^t$
end

Relationship between EF and EF21

Algorithm 2: EF (Error Feedback version 2014) [2]

Input: $x^0 \in \mathbb{R}^d$; $\gamma > 0$; $e^0 = 0$, $w^0 = \mathcal{C}\left(\gamma\nabla f(x^0)\right)$
for $t = 0, 1, 2, \dots, T-1$ **do**
  $x^{t+1} = x^t - \gamma w^t$
  $e^{t+1} = e^t + \gamma\nabla f\left(x^t\right) - w^t$
  $w^{t+1} = \mathcal{C}\left(e^{t+1} + \gamma\nabla f\left(x^{t+1}\right)\right)$
end

Algorithm 3: EF21 (Single node)

Input: $x^0 \in \mathbb{R}^d$; $\gamma > 0$; $g^0 = \mathcal{C}(\nabla f(x^0))$
for $t = 0, 1, 2, \dots, T-1$ **do**
  $x^{t+1} = x^t - \gamma g^t$
  $g^{t+1} = g^t + \mathcal{C}(\nabla f(x^{t+1}) - g^t)$
end

Restricted equivalence of EF and EF21

Theorem 1. *Assume that $\mathcal{C}$ is **deterministic**, **positively homogeneous** and **additive**. Then EF (Algorithm 2) and EF21 (Algorithm 3) produce the same sequences of iterates $\{x^t\}_{t\ge0}$. The same holds for distributed versions of the methods.*

Remark 1. *The conditions of Theorem 1 are not met for popular compressors used in practice. For example, Top-k compressor is deterministic and positively homogeneous, but **is not additive**.*

EF21+: an improved variant

Idea: Use $\mathcal{C}$ or the Markov compressor, whichever is better.

Compute gradient compressed by the **contractive compressor**

$$b_i^{t+1} = \mathcal{C}(\nabla f_i(x^{t+1})).$$

Compute gradient compressed by the **Markov compressor**

$$m_i^{t+1} = g_i^t + \mathcal{C}(\nabla f_i(x^{t+1}) - g_i^t).$$

Compute distortions:

$$B_i^{t+1} = \left\|b_i^{t+1} - \nabla f_i(x^{t+1})\right\|^2, \quad M_i^{t+1} = \left\|m_i^{t+1} - \nabla f_i(x^{t+1})\right\|^2.$$

Define local gradient estimator as the best of the two:

$$g_i^{t+1} = \begin{cases} m_i^{t+1} & \text{if} \quad M_i^{t+1} \le B_i^{t+1} \\ b_i^{t+1} & \text{if} \quad M_i^{t+1} > B_i^{t+1}. \end{cases}$$

Convergence theory

Define

$$G^t := \frac{1}{n}\sum_{i=1}^n \left\|g_i^t - \nabla f_i(x^t)\right\|^2, \quad \widetilde{L} := \frac{1}{n}\sum_{i=1}^n L_i^2,$$

$$\theta := 1 - \sqrt{1-\alpha}, \quad \beta := \frac{1-\alpha}{1-\sqrt{1-\alpha}}.$$

EF21 for general non-convex functions

Theorem 2. *Let Assumption 1 hold, and let the stepsize in Algorithm 1 be set as*

$$0 < \gamma \le \left(L + \widetilde{L}\sqrt{\frac{\beta}{\theta}}\right)^{-1}. \tag{4}$$

Fix $T \ge 1$ and let $\hat{x}^T$ be chosen from the iterates $x^0, x^1, \dots, x^{T-1}$ uniformly at random. Then

$$\mathbb{E}\left[\left\|\nabla f(\hat{x}^T)\right\|^2\right] \le \frac{2\left(f(x^0) - f^{\inf}\right)}{\gamma T} + \frac{\mathbb{E}\left[G^0\right]}{\theta T}. \tag{5}$$

This is the first $O(1/T)$ rate for error feedback. Best previous rate was $O(1/T^{2/3})$, and under strong/unreasonable assumptions.

EF21 for PL functions

Theorem 3. *Let Assumptions 1 and 2 hold, and let the stepsize in Algorithm 1 be set as*

$$0 < \gamma \le \min\left\{\left(L + \widetilde{L}\sqrt{\frac{2\beta}{\theta}}\right)^{-1}, \frac{\theta}{2\mu}\right\}. \tag{6}$$

Let $\Psi^t := f(x^t) - f(x^\star) + \frac{\gamma}{\theta}G^t$. Then for any $T \ge 0$, we have

$$\mathbb{E}\left[\Psi^T\right] \le (1-\gamma\mu)^T\mathbb{E}\left[\Psi^0\right]. \tag{7}$$

This is the first linear rate for error feedback in the distributed setting $n > 1$ without strong assumptions (such as reliance on over-parameterized regime).

EF21+ for general non-convex & PL functions

Theorem 4. *If additionally, the compressor $\mathcal{C}$ is deterministic, then Theorems 2 and 3 hold for EF21+ as well.*

Summary of complexity results

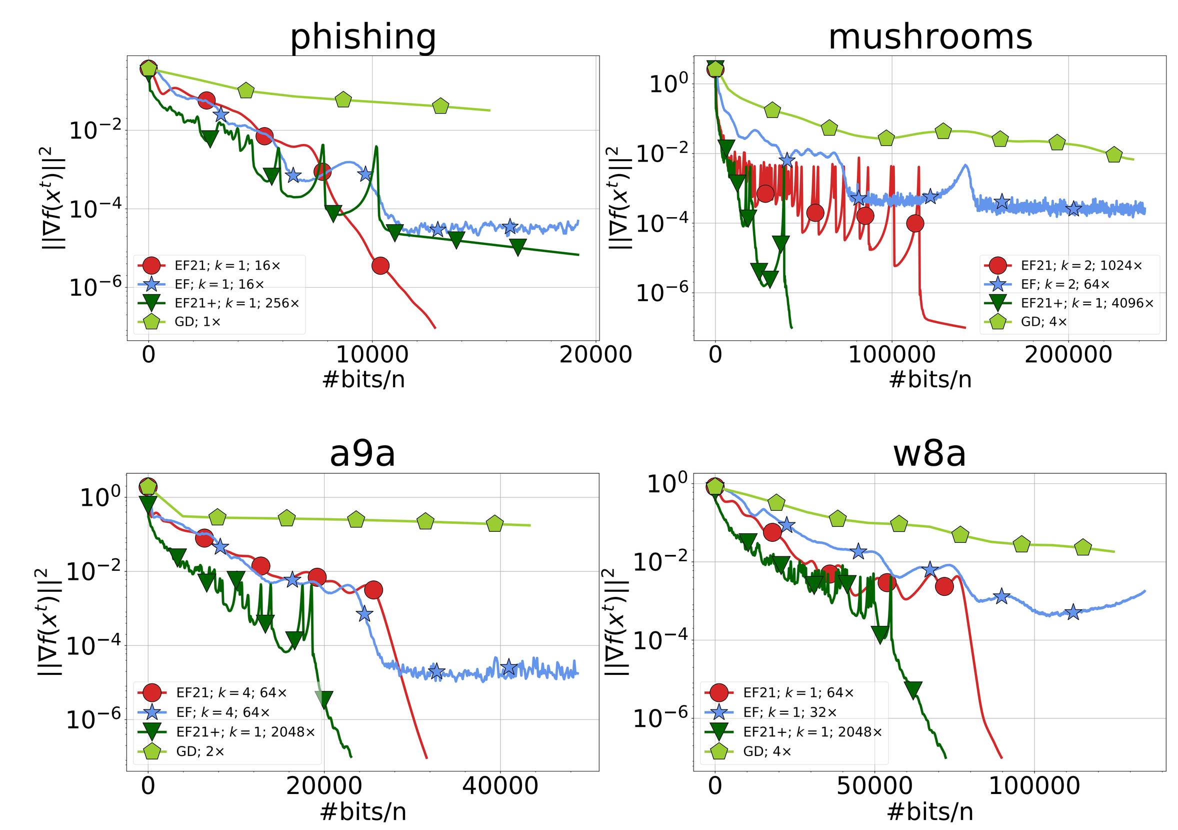
Assumptions	Complexity	Theorem
1	$\mathbb{E}\left[\left\|\nabla f(\hat{x}^T)\right\|^2\right] \le \frac{2(f(x^0)-f^{\inf})}{\gamma T} + \frac{\mathbb{E}[G^0]}{\theta T}$	2
1, 2	$\mathbb{E}\left[\Psi^T\right] \le (1-\gamma\mu)^T\mathbb{E}\left[\Psi^0\right]$	3

Experiments

Logistic regression problem with a **non-convex** regularizer,

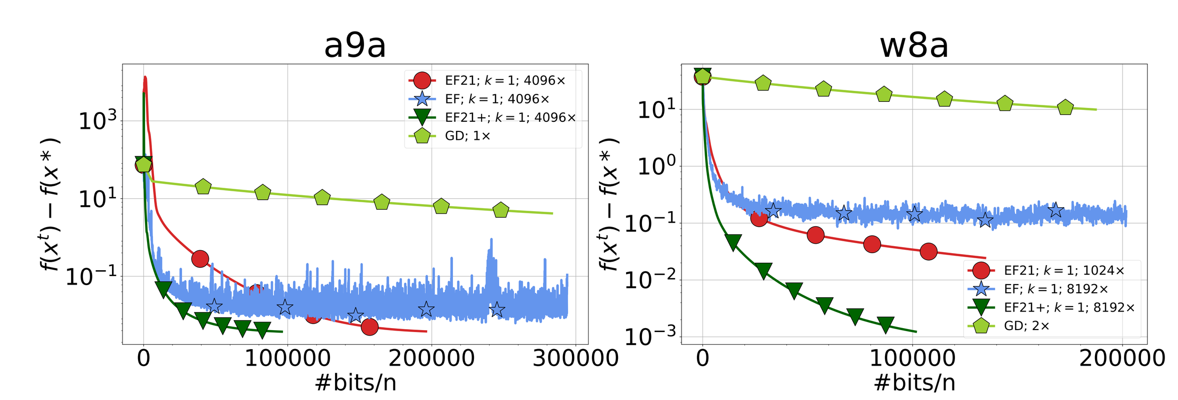
$$f(x) = \frac{1}{n}\sum_{i=1}^n \log\left(1 + \exp\left(-y_i a_i^\top x\right)\right) + \lambda\sum_{j=1}^d \frac{x_j^2}{1+x_j^2}, \tag{8}$$

where $a_i \in \mathbb{R}^d$, $y_i \in \{-1, 1\}$ are the training data, and $\lambda > 0$ is the regularization parameter.



Least squares problem (satisfies PL condition with $\mu = \sigma_{min}^2(A)$)

$$f(x) = \frac{1}{n}\sum_{i=1}^n (a_i^\top x - y_i)^2. \tag{9}$$



By $1\times, 2\times, 4\times$ (and so on) it is indicated that the stepsize was set to a multiple of the largest stepsize predicted by our theory. $k = 1$ means that Top-1 compressor was used in the experiment. Both stepsizes and k were fine-tuned in the experiments above.

References

[1] A. Beznosikov, S. Horváth, P. Richtárik, and M. Safaryan. On biased compression for distributed learning. arXiv:2002.12410, 2020.

[2] S. U. Stich and S. P. Karimireddy. The error-feedback framework: Better rates for SGD with delayed gradients and compressed communication. arXiv:1909.05350, 2019.