Momentum Provably Improves Error Feedback!

Ilyas Fatkhullin (ETH Zurich & ETH AI Center) Alexander Tyurin (KAUST) Peter Richtárik (KAUST)

Distributed stochastic optimization

$$\min_{x\in\mathbb{R}^d}\left[f(x) := \frac{1}{n}\sum_{i=1}^n f_i(x)\right], \qquad f_i(x) := \mathbb{E}_{\xi_i\sim\mathcal{D}_i}[f_i(x,\xi_i)]$$

- $f_i(\cdot)$ are nonconvex, • n is the number of nodes,
- $\mathcal{D}_i$ are arbitrary distributions of data.

Goal: find x with $\mathbb{E}[\|\nabla f(x)\|] \le \varepsilon$.

Assumptions

Assumption 1 (Lipschitz gradient). *$f^* := \inf_{x\in\mathbb{R}^d} f(x) > -\infty$, $f(\cdot)$ and all $f_i(\cdot)$ have Lipschitz continuous gradient, i.e., $\forall x, y\in\mathbb{R}^d$*

$$\|\nabla f(x) - \nabla f(y)\| \le L\|x-y\|,$$

$$\|\nabla f_i(x) - \nabla f_i(y)\| \le L_i\|x-y\|, \qquad \widetilde{L}^2 := \frac{1}{n}\sum_{i=1}^n L_i^2.$$

Assumption 2 (Bounded variance). *Unbiased stochastic gradients: $\mathbb{E}[\nabla f_i(x,\xi_i)] = \nabla f_i(x)$. There exists $\sigma > 0$ such that*

$$\mathbb{E}\left[\|\nabla f_i(x,\xi_i) - \nabla f_i(x)\|^2\right] \le \sigma^2.$$

Contractive compressors

A map $\mathcal{C}:\mathbb{R}^d\to\mathbb{R}^d$ is a **contractive compressor** if $\exists\, \alpha\in(0,1]$:

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

Example: Top$K$ (greedy) sparsification keeps the $K \le d$ largest entries of x in absolute value, and zeros out the rest. It is biased and **contractive** with $\alpha \ge \frac{K}{d}$.

Compressed gradient methods

Distributed first-order method

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

where $\gamma > 0$ is the step-size, and g_i^t is an easy-to-communicate (i.e., compressed) approximation of $\nabla f_i(x^t)$. How to construct g_i^t?

1. Naive method: Compressed SGD $\qquad g_i^t = \mathcal{C}(\nabla f_i(x^t,\xi_i^t))$.

Advantages:
- Conceptually easy.

Problem:
- Diverges even for $\sigma = 0$ (if $n > 1$).

2. Error feedback (2014): EF14-SGD [1, 2]

$$e_i^{t+1} = e_i^t + \gamma\nabla f_i(x^t,\xi_i^t) - g_i,$$
$$g_i^{t+1} = \mathcal{C}(e_i^{t+1} + \gamma\nabla f_i(x^{t+1},\xi_i^{t+1})).$$

Advantages:
- Converges (if $\mathcal{D}_i$ are similar) [2].

Problems:
- Not optimal even if $\sigma = 0$.
- Similarity of $\mathcal{D}_i$ is needed for analysis.

3. Modern error feedback (2021): EF21-SGD [3, 4]

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

Advantages:
- Converges if $\sigma = 0$ [3].
- Optimal iteration complexity if $\sigma = 0$.

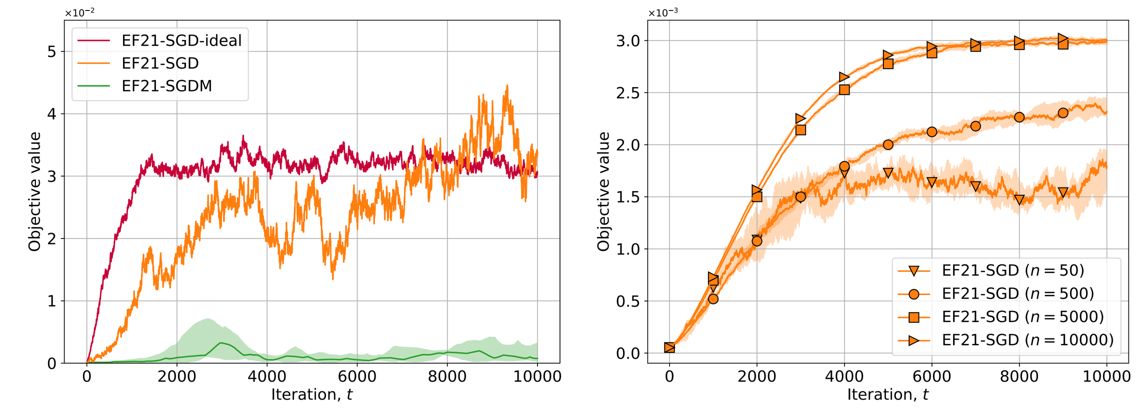
Problems:
- Requires mega-batch: $B = \mathcal{O}(\frac{1}{\alpha\varepsilon^2})$.
- Poor dependence on α.
- No improvement with n.

EF21 and stochastic gradients

Assume we know $\nabla f_i(x^{t+1})$. Replace g_i^t with $\nabla f_i(x^{t+1})$.

EF21-SGD-ideal: $g_i^{t+1} = \nabla f_i(x^{t+1}) + \mathcal{C}\left(\nabla f_i(x^{t+1},\xi_i^{t+1}) - \nabla f_i(x^{t+1})\right)$.

Divergence of EF21-SGD **and** EF21-SGD-ideal on $f(x) = \frac{1}{2}\|x\|^2$.
No improvement when increasing n (right).



Divergence of EF21-SGD-ideal

Theorem 1. EF21-SGD-ideal may diverge for any $n \ge 1$:

$$\mathbb{E}\left[\|\nabla f(x^t)\|^2\right] \ge \frac{1}{60}\min\left\{\sigma^2, \|\nabla f(x^0)\|^2\right\}.$$

Conceptual fix

Apply damping of the noise with $\eta\in(0,1)$. EF21-SGDM-ideal:

$$v_i^{t+1} = \nabla f_i(x^{t+1}) + \eta(\nabla f_i(x^{t+1},\xi_i^{t+1}) - \nabla f_i(x^{t+1})),$$
$$g_i^{t+1} = \nabla f_i(x^{t+1}) + \mathcal{C}\left(v_i^{t+1} - \nabla f_i(x^{t+1})\right).$$

Advantages:
- Fast convergence for small η.
- If $\eta = 0$, method recovers GD.

Problems:
- Not implementable without $\nabla f(x^{t+1})$.

Implementable method

Replace $\nabla f_i(x^{t+1})$ with v_i^t and $\nabla f_i(x^{t+1})$ with g_i^t. EF21-SGDM:

$$v_i^{t+1} = v_i^t + \eta(\nabla f_i(x^{t+1},\xi_i^{t+1}) - v_i^t),$$
$$g_i^{t+1} = g_i^t + \mathcal{C}\left(v_i^{t+1} - g_i^t\right).$$

Advantages:
- Easy to implement.
- Works with stochastic gradients.
- Fast convergence in theory and practice.

Algorithm 1: EF21-SGDM
Error Feedback 2021 Enhanced with Polyak Momentum

Input: $x^0, v_i^0, g_i^0\in\mathbb{R}^d$; $\gamma > 0$; $\eta\in(0,1]$, initial batch size B_{init}
for $t = 0, 1, \ldots, T-1$ **do**
  Master computes $x^{t+1} = x^t - \gamma g^t$ and broadcasts x^{t+1} to all nodes
  for *all nodes* $i = 1, \ldots, n$ *in parallel* **do**
    Compute momentum estimator
    $$v_i^{t+1} = (1-\eta)v_i^t + \eta\nabla f_i(x^{t+1},\xi_i^{t+1})$$
    Compress $c_i^t = \mathcal{C}(v_i^{t+1} - g_i^t)$ and send c_i^t to the master
    Update local state $g_i^{t+1} = g_i^t + \mathcal{C}(v_i^{t+1} - g_i^t)$
  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

Summary of complexity results

Method	Communication complexity with Top$K$	Asymptotic sample complexity	No batch?	No extra assump.?
EF14-SGD [2]	$\frac{KG}{\alpha\varepsilon^3}$	$\frac{\sigma^2}{n\varepsilon^4}$	✔	✗(a)
NEOLITHIC	$\frac{K}{\alpha\varepsilon^2}\log\left(\frac{G}{\varepsilon}\right)$	$\frac{\sigma^2}{n\varepsilon^4}$	✗	✗(b)
EF21-SGD [4]	$\frac{K}{\alpha\varepsilon^2}$	$\frac{\sigma^2}{\alpha^3\varepsilon^4}$	✗	✔
BEER	$\frac{K}{\alpha\varepsilon^2}$	$\frac{\sigma^2}{\alpha^2\varepsilon^4}$	✗	✔
EF21-SGDM, EF21-SGD2M	$\frac{K}{\alpha\varepsilon^2}$	$\frac{\sigma^2}{n\varepsilon^4}$	✔	✔

(a) Extra assumption (BG): $\mathbb{E}\left[\|\nabla f_i(x,\xi_i)\|^2\right] \le G^2$. (b) Extra assumption (BGS) $\frac{1}{n}\sum_{i=1}^n\|\nabla f_i(x) - \nabla f(x)\|^2 \le G^2$.

Convergence theory

Denote $\delta_t := f(x^t) - f^*$, $v^t := \frac{1}{n}\sum_{i=1}^n v_i^t$. The Lyapunov function is Λ_t:

$$\delta_t + \frac{\gamma}{\alpha n}\sum_{i=1}^n\|g_i^t - v_i^t\|^2 + \frac{\gamma\eta}{\alpha^2 n}\sum_{i=1}^n\|v_i^t - \nabla f_i(x^t)\|^2 + \frac{\gamma}{\eta}\|v^t - \nabla f(x^t)\|^2.$$

EF21-SGDM

Theorem 2. *Under Assumptions 1, 2 and small enough step-size we have for* EF21-SGDM

$$\mathbb{E}\left[\|\nabla f(\hat{x}^T)\|^2\right] = \mathcal{O}\left(\frac{\Lambda_0}{\gamma T} + \frac{\eta^3\sigma^2}{\alpha^2} + \frac{\eta^2\sigma^2}{\alpha} + \frac{\eta\sigma^2}{n}\right).$$

Choosing appropriate γ, η and B_{init}, we have for the RHS

$$\mathcal{O}\left(\frac{\widetilde{L}\delta_0}{\alpha T} + \left(\frac{L\delta_0\sigma^{2/3}}{\alpha^{2/3}T}\right)^{3/4} + \left(\frac{L\delta_0\sigma}{\sqrt{\alpha}T}\right)^{2/3} + \left(\frac{L\delta_0\sigma^2}{nT}\right)^{1/2}\right).$$

The sample complexity:

$$\#grad = T = \mathcal{O}\left(\frac{\widetilde{L}}{\alpha\varepsilon^2} + \frac{L\sigma^{2/3}}{\alpha^{2/3}\varepsilon^{8/3}} + \frac{L\sigma}{\alpha^{1/2}\varepsilon^3} + \frac{L\sigma^2}{n\varepsilon^4}\right).$$

Advantages:
- Better than EF14-SGD: $\#grad = \mathcal{O}\left(\frac{G}{\alpha\varepsilon^3} + \frac{\sigma^2}{n\varepsilon^4}\right)$.
- Better than EF21-SGD: $\#grad = \mathcal{O}\left(\frac{1}{\alpha\varepsilon^2} + \frac{\sigma^2}{\alpha^3\varepsilon^4}\right)$.
- Weaker assumptions than for EF14-SGD, and batch-free!
- Optimal communication complexity when used with $B \ge 1$.
- Asymptotically optimal sample complexity ($\varepsilon\to 0$).

Further improvement with double momentum!

Use momentum **twice** before compression. EF21-SGD2M:

$$v_i^{t+1} = (1-\eta)v_i^t + \eta\nabla f_i(x^{t+1},\xi_i^{t+1}),$$
$$u_i^{t+1} = (1-\eta)u_i^t + \eta v_i^{t+1},$$
$$g_i^{t+1} = g_i^t + \mathcal{C}(u_i^{t+1} - g_i^t).$$

EF21-SGD2M

Theorem 3. The sample complexity of EF21-SGD2M:

$$T = \mathcal{O}\left(\frac{\widetilde{L}}{\alpha\varepsilon^2} + \frac{L\sigma^{2/3}}{\alpha^{2/3}\varepsilon^{8/3}} + \frac{L\sigma^2}{n\varepsilon^4}\right).$$

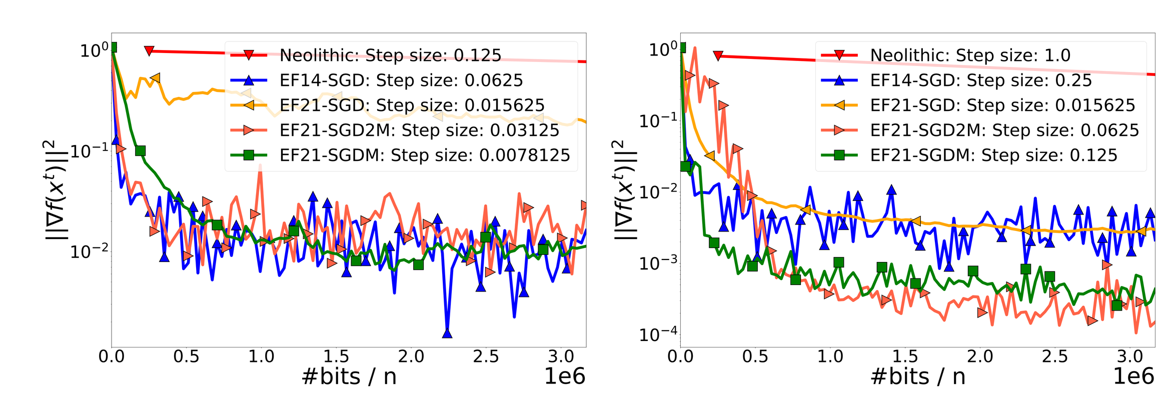
Experiments

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

$$f_i(x_1,\ldots,x_c) = -\frac{1}{m}\sum_{j=1}^m\log\left(\frac{\exp(a_{ij}^\top x_{y_{ij}})}{\sum_{y=1}^c\exp(a_{ij}^\top x_y)}\right) + \lambda\sum_{y=1}^c\sum_{k=1}^l\frac{[x_y]_k^2}{1+[x_y]_k^2},$$

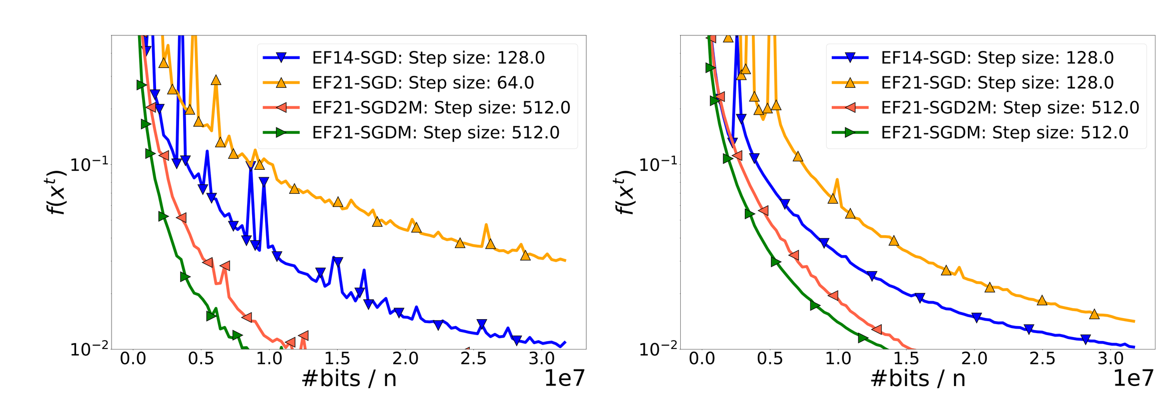
where $\lambda > 0$, $x_1,\ldots,x_c\in\mathbb{R}^l$, $[\cdot]_k$ is an indexing operation of a vector, $c \ge 2$ is the number of classes, l is the number of features, m is the size of a dataset, $a_{ij}\in\mathbb{R}^l$ and $y_{ij}\in\{1,\ldots,c\}$ are features and labels.

Experiment 1: increasing batch-size. Dataset: `MNIST`.



$B = 1$ $\qquad$ $B = 128$

Experiment 2: improving with n. Dataset: `real-sim`.



$n = 10$ $\qquad$ $n = 100$

Stepsizes were fine-tuned in all experiments.

References

[1] F. Seide, H. Fu, J. Droppo, G. Li, D. Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs. Interspeach 2014.

[2] A. Koloskova, T. Lin, S. Stich, and M. Jaggi. Decentralized deep learning with arbitrary communication compression. ICLR 2020.

[3] P. Richtárik, I. Sokolov, I. Fatkhullin. EF21: A New, Simpler, Theoretically Better, and Practically Faster Error Feedback. NeurIPS 2021.

[4] I. Fatkhullin, I. Sokolov, E. Gorbunov, Z. Li, P. Richtárik. EF21 with Bells & Whistles: Practical Algorithmic Extensions of Modern Error Feedback. arXiv:2110.03294. 2021.