Smoothed Normalization for Efficient Distributed Private Optimization

Egor Shulgin Sarit Khirirat Peter Richtárik

King Abdullah University of Science and Technology (KAUST)

Motivation: Private Learning

- Machine Learning (ML) models can leak sensitive data.
- DP-SGD (2) is the standard method for Differentially Private (DP) optimization [1].
- Clipping introduces bias preventing convergence to the exact solution.
- Existing analyses rely on **unrealistic assumptions** (e.g. bounded gradients), effectively ignoring the clipping bias.

Problem Formulation

Many ML problems can be reformulated as finite-sum (distributed) optimization:

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

where each f_i is L-smooth (privately known to client i) and f is lower-bounded by $f^{\inf} > -\infty$.

Private Optimization

DP-SGD performs **clipping** $\Psi(g) = \min(1, \Phi/\|g\|)g$ of clients' gradients and **adds noise**:

$$x^{k+1} = x^k - \gamma\left(\frac{1}{|\mathcal{S}^k|}\sum_{i\in\mathcal{S}^k}\Psi(\nabla f_i(x^k)) + z^k\right), \tag{2}$$

where $\|\Psi(g)\| \le \Phi$ and z^k is zero-mean Gaussian with variance proportional to *sensitivity* Φ.

Example: DP-SGD Fails to Converge

Consider the **problem** (1) with $n = 2, d = 1$, i.e.

$$f_1(x) = \frac{1}{2}(x-3)^2 \quad \text{and} \quad f_2(x) = \frac{1}{2}(x+3)^2.$$

The **minimum** to this problem is at $x^\star = 0$.

The **failure:** At $x^0 = 2$, the algorithm stalls (even without noise $z^k = 0$).

The **reason:** The normalized gradients cancel each other out, resulting in a zero update:

$$\underbrace{\frac{\nabla f_1(x^0)}{\|\nabla f_1(x^0)\|}}_{-1} + \underbrace{\frac{\nabla f_2(x^0)}{\|\nabla f_2(x^0)\|}}_{+1} = 0$$

Smoothed Normalization

Smoothed normalization [3] is an alternative operator to clipping defined as

$$\mathrm{Norm}_\alpha(g) := \frac{1}{\alpha + \|g\|}g \quad (\alpha \ge 0). \tag{3}$$

This operator ensures

- Sensitivity control for DP ($\|\mathrm{Norm}_\alpha(g)\| \le 1$).
- Contractive property, unlike clipping.
- Easier parameter tuning [3].

Operator	Property
Contractive Compressor	$\|\mathcal{C}(g) - g\| \le \|1-\eta\|\,\|g\|$
Clipping	$\|\mathrm{Clip}_\tau(g) - g\| \le \max(0, \|g\| - \tau)$
Smoothed Normalization	$\|\mathrm{Norm}_\alpha(g) - g\| \le \left\|1 - \frac{1}{\alpha+\|g\|}\right\|\,\|g\|$

Table 1. Smoothed normalization, unlike clipping, satisfies the contractive property similar to contractive compressors.

Our Method: α-NormEC

To alleviate the convergence issue of DP-SGD, we propose α-NormEC, which utilizes

1. **Smoothed normalization** to bound sensitivity of clients' contributions,
2. **Error Compensation** (EC), i.e. EF21 [4], to alleviate the operator-induced bias.

Algorithm Description

Given parameters: $\beta, \gamma, \sigma_{\mathrm{DP}} > 0$ and $x^0, g_i^0 \in \mathbb{R}^d$.

At iteration k

1. Each client $i \in [1, n]$
   - Computes normalized **difference**
   $$\Delta_i^k = \mathrm{Norm}_\alpha(\nabla f_i(x^k) - g_i^k), \tag{4}$$
   - Updates local error **memory**
   $$g_i^{k+1} = g_i^k + \beta\Delta_i^k, \tag{5}$$
   - Transmits **privatized** update
   $$\hat{\Delta}_i^k = \Delta_i^k + z_i^k. \tag{6}$$
2. The server updates the next iterate
$$x^{k+1} = x^k - \gamma\hat{g}^{k+1}/\left\|\hat{g}^{k+1}\right\|, \tag{7}$$
where
$$\hat{g}^{k+1} = \hat{g}^k + \frac{\beta}{n}\sum_{i=1}^{n}\hat{\Delta}_i^k. \tag{8}$$

Convergence Guarantees

Consider problem (1), then for parameters β, γ

$$\frac{\beta}{\alpha + R} < 1, \quad \text{and} \quad \gamma \le \frac{\beta R}{\alpha + R}\frac{1}{L},$$

the iterates of α-NormEC satisfy:

$$\min_{k\le K}\mathbb{E}\left\|\nabla f(x^k)\right\| \le \underbrace{\frac{f(x^0) - f^{\inf}}{\gamma(K+1)} + \frac{\gamma L}{2}}_{\text{Standard convergence}} + 2R + 2\underbrace{\sqrt{\frac{\beta^2(K+1)\sigma_{\mathrm{DP}}^2}{n}}}_{\text{DP noise cost}},$$

where $R = \max_i\left\|\nabla f_i(x^0) - g_i^0\right\|$ is the initialization error, and σ_{DP}^2 is the DP noise variance.

Our Contributions

- This is the **first provable convergence guarantee** for a distributed DP method that explicitly handles the operator-induced bias **without restrictive assumptions**.
- Unlike Clip21 [2], α-NormEC achieves convergence in the presence of DP noise.
- In the non-private case ($\sigma_{\mathrm{DP}} = 0$), α-NormEC obtains the $\mathcal{O}(1/\sqrt{K})$ rate for non-convex problems, which is faster than Clip21.

References

[1] M. Abadi, A. Chu, I. Goodfellow, H. B. McMahan, I. Mironov, K. Talwar, and L. Zhang. Deep learning with differential privacy. *ACM SIGSAC*, 2016.

[2] S. Khirirat, E. Gorbunov, S. Horváth, R. Islamov, F. Karray, and P. Richtárik. Clip21: Error feedback for gradient clipping. *arXiv:2305.18929*, 2023.

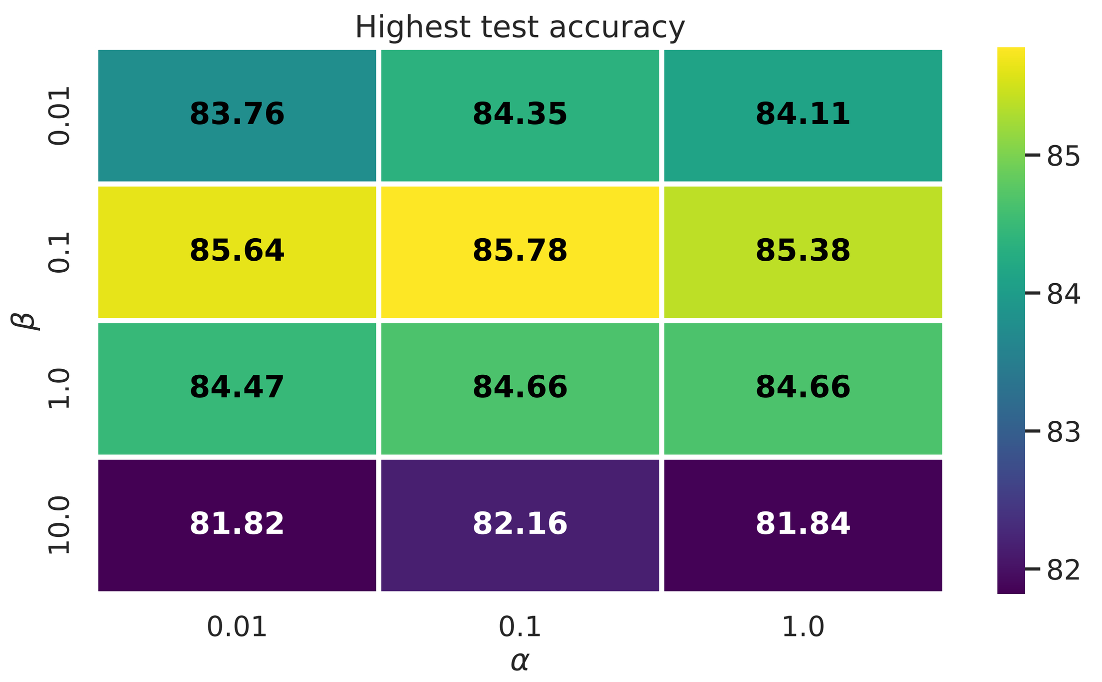
[3] Z. Bu, Y.X. Wang, S. Zha, G. Karypis. Automatic clipping: Differentially private deep learning made easier and stronger. *NeurIPS*, 2023.

[4] P. Richtárik, I. Sokolov, I. Fatkhullin. EF21: A new, simpler, theoretically better, and practically faster error feedback. *NeurIPS*, 2021.

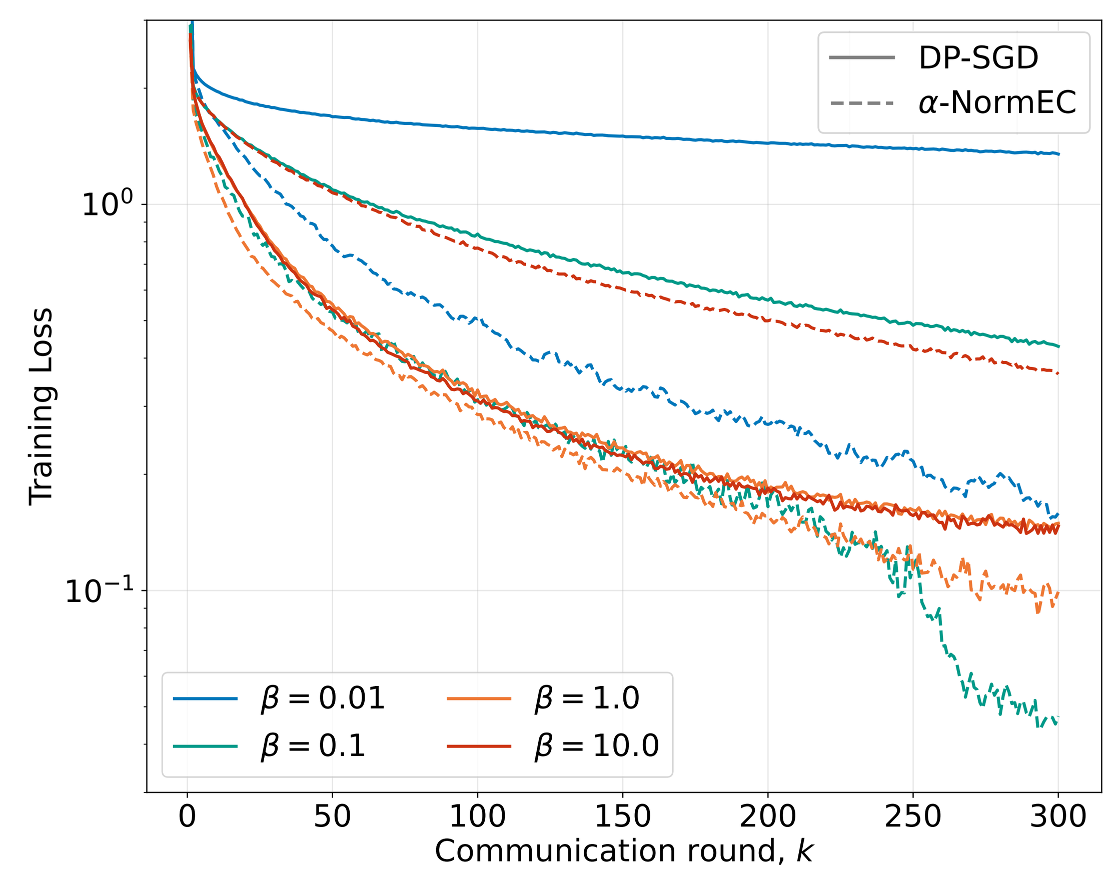
Experimental Results

We ran α-NormEC, DP-SGD, and Clip21 for training a ResNet20 model on a CIFAR-10 dataset.

Non-Private Setting

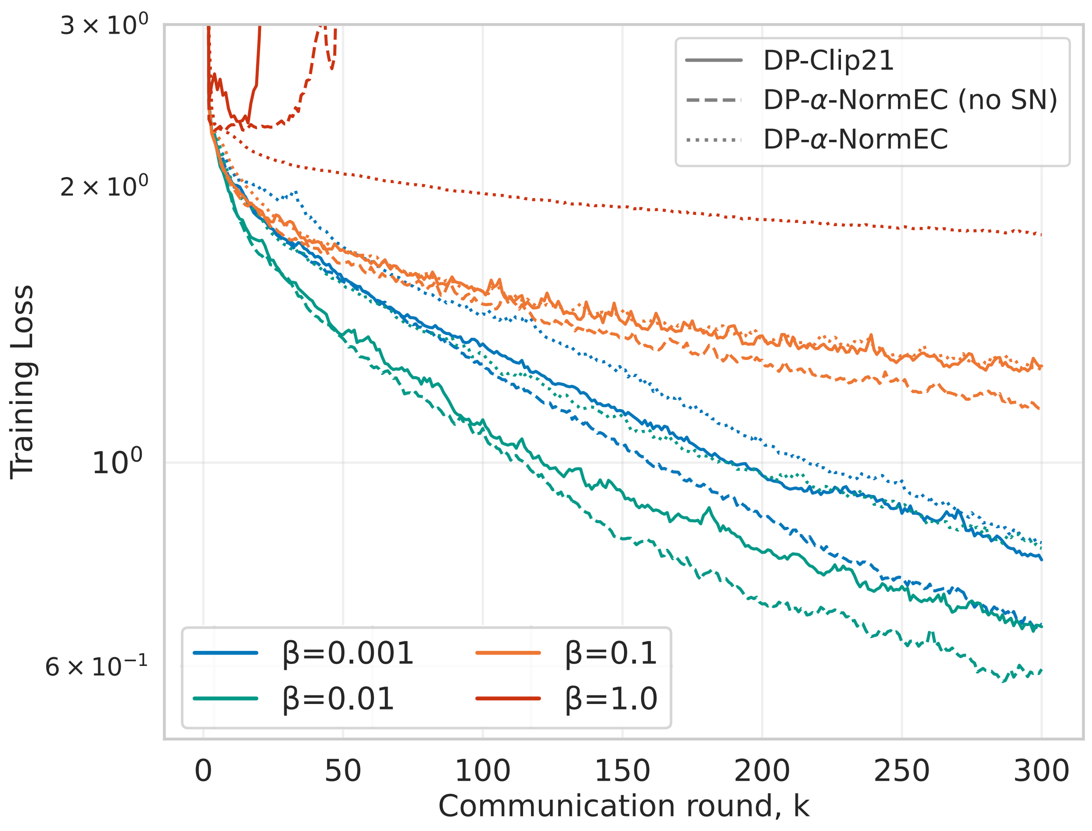


α-NormEC is robust to hyperparameters, remaining stable across β values and insensitive to the normalization parameter α.



Error Compensation significantly improves convergence allowing α-NormEC to outperform DP-SGD across various β values.

Private Setting



DP-α-NormEC outperforms DP-Clip21, across different β values. Server-side normalization **(SN) provides stability** for high noise level ($\beta = 1$).

Conclusion and Future Work

α-NormEC is the first distributed private optimization method with **convergence guarantees**. In practice it **outperforms** existing **competitors** across varying hyper-parameters.

Promising future directions are to

- Extend to partial client participation settings.
- Use stochastic gradients at the clients.
- Adapt to complex federated learning protocols.