# GradSkip: Communication-Accelerated Local Gradient Methods with Better Computational Complexity


**Artavazd Maranjyan**, Mher Safaryan, Peter Richtárik

Artificial Intelligence Initiative, KAUST

OIST

MLSS Machine Learning Summer Schools

جامعة الملك عبدالله للعلوم والتقنية

King Abdullah University of Science and Technology


## Federated Learning

**Federated Learning** is collaborative machine learning from private data stored across a (large) number of clients/devices (e.g., hospitals, phones).


$$\min_{x\in\mathbb{R}^d} f(x) \overset{\text{def}}{=} \frac{1}{n}\sum_{i=1}^{n} f_i(x)$$

$n$: # devices / machines

$d$: # model parameters / features

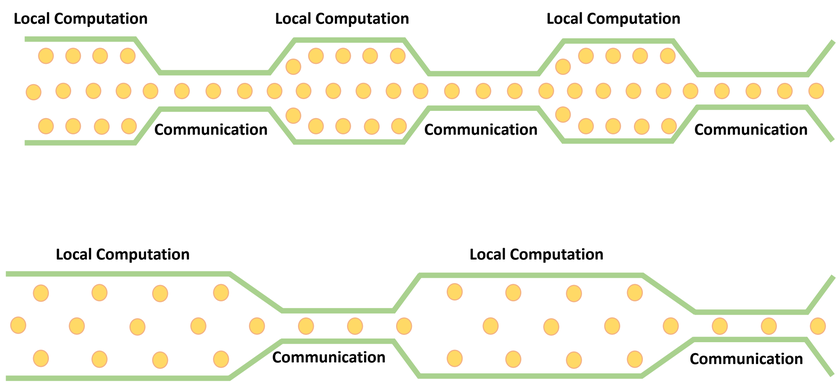
Loss on local data $\mathcal{D}_i$ stored on device $i$

$$f_i(x) = \mathbb{E}_{\xi\sim\mathcal{D}_i} f_{i,\xi}(x)$$

The datasets $\mathcal{D}_1,\dots,\mathcal{D}_n$ can be arbitrarily heterogeneous

## The idea behind Local Training

Communication costs often become a **bottleneck** in federated optimization due to slow and unreliable wireless links between clients and the central server. A simple trick to reduce communication costs is to **infrequently** perform the costly synchronization step, allowing multiple local gradient steps in each communication round.




## Assumptions

All functions $f_i(x)$ are strongly convex with parameter $\mu > 0$ and have Lipschitz continuous gradients with Lipschitz constants $L_i > 0$, i.e., for all $i \in [n]$ and any $x, y \in \mathbb{R}^d$ we have

$$\mu\|x-y\| \le \|\nabla f_i(x) - \nabla f_i(y)\| \le L_i\|x-y\|.$$

## Key theoretical trick

Local Step: $x_{i,t+1} = x_{i,t} - \gamma(\nabla f_i(x_{i,t}) - h_{i,t})$

with probability $1-q_i$ do $\hat{h}_{i,t+1} = h_{i,t}$

with probability $q_i$ do $\hat{h}_{i,t+1} = \nabla f_i(x_{i,t})$

$$x_{i,t+1} = x_{i,t} - \gamma\left(\nabla f_i(x_{i,t}) - \hat{h}_{i,t+1}\right)$$

## The GradSkip Algorithm

**Worker 1**

$f_1$ $L_1$

Receive $x_t$ and $K_t$ from the server

$x_{1,t} = x_t \qquad h_{1,t} = h_{1,t-1} + \text{CD} \to \nabla f_1(x_\star)$

$M_{1,t} \sim \text{Geo}(q_1)$ $\qquad K_{1,t} = \min\{M_{1,t}, K_t\}$

$x_{1,t+1} = x_{1,t} - \gamma\left(\nabla f_1(x_{1,t}) - h_{1,t}\right)$

$x_{1,t+2} = x_{1,t+1} - \gamma\left(\nabla f_1(x_{1,t+1}) - h_{1,t}\right)$

$\vdots$

$x_{1,t+K_{1,t}} = x_{1,t+K_{1,t}-1} - \gamma\left(\nabla f_1(x_{1,t+K_{1,t}-1}) - h_{1,t}\right)$

**Worker 2**

$f_2$ $L_2$

Receive $x_t$ and $K_t$ from the server

$x_{2,t} = x_t \qquad h_{2,t} = h_{2,t-1} + \text{CD} \to \nabla f_2(x_\star)$

$M_{2,t} \sim \text{Geo}(q_2)$ $\qquad K_{2,t} = \min\{M_{2,t}, K_t\}$

$x_{2,t+1} = x_{2,t} - \gamma\left(\nabla f_1(x_{2,t}) - h_{2,t}\right)$

$x_{2,t+2} = x_{2,t+1} - \gamma\left(\nabla f_1(x_{2,t+1}) - h_{2,t}\right)$

$\vdots$

$x_{2,t+K_{2,t}} = x_{2,t+K_{2,t}-1} - \gamma\left(\nabla f_2(x_{2,t+K_{2,t}-1}) - h_{2,t}\right)$

**Worker 3**

$f_3$ $L_3$

Receive $x_t$ and $K_t$ from the server

$x_{3,t} = x_t \qquad h_{3,t} = h_{3,t-1} + \text{CD} \to \nabla f_3(x_\star)$

$M_{3,t} \sim \text{Geo}(q_3)$ $\qquad K_{3,t} = \min\{M_{3,t}, K_t\}$

$x_{3,t+1} = x_{3,t} - \gamma\left(\nabla f_1(x_{3,t}) - h_{3,t}\right)$

$x_{3,t+2} = x_{3,t+1} - \gamma\left(\nabla f_1(x_{3,t+1}) - h_{3,t}\right)$

$\vdots$

$x_{3,t+K_{3,t}} = x_{3,t+K_{3,t}-1} - \gamma\left(\nabla f_3(x_{3,t+K_{3,t}-1}) - h_{3,t}\right)$

**Server**

$K_t \sim \text{Geo}(p)$

$$x_{t+K_t} = \frac{1}{3}\sum_{i=1}^{3} x_{i,t+K_t}$$

Broadcast $K_t$ and $x_{t+K_t}$ to the workers

CD = Client Drift

$$\frac{p}{\gamma}\left(x_{i,t+K_t} - \hat{x}_{i,t}\right)$$

## Convergence Theorem

$$\gamma \le \min_i\left\{\frac{1}{L_i}\frac{p^2}{p^2+q_i(1-p^2)}\right\}$$

$$t \ge \max\left\{\frac{1}{\gamma\mu}, \frac{1}{p^2-q_{\min}(1-p^2)}\right\}\log\frac{1}{\varepsilon} \Rightarrow \mathbb{E}\left[\Psi_t\right] \le \varepsilon\Psi_0$$

$t$: # iterations

Lyapunov function:

$$\Psi_t := \sum_{i=1}^{n}\|x_{i,t} - x_\star\|^2 + \frac{\gamma^2}{p^2}\sum_{i=1}^{n}\|h_{i,t} - \nabla f_i(x_\star)\|^2$$

## Optimal Hyperparameters

$$\kappa_i = \frac{L_i}{\mu} \qquad \kappa_{\max} = \frac{L_{\max}}{\mu}$$

$$q_i = \frac{\frac{1}{\kappa_i} - \frac{1}{\kappa_{\max}}}{1 - \frac{1}{\kappa_{\max}}} \qquad p^2 = \frac{1}{\kappa_{\max}}$$

$$\gamma \le \min_i\left\{\frac{1}{L_i}\frac{p^2}{p^2+q_i(1-p^2)}\right\} = \frac{1}{L_{\max}}$$

## Optimal Convergence Rate

$$t \ge \max\left\{\frac{1}{\gamma\mu}, \frac{1}{p^2-q_{\min}(1-p^2)}\right\}\log\frac{1}{\varepsilon} = \kappa_{\max}\log\frac{1}{\varepsilon} \Rightarrow \mathbb{E}\left[\Psi_t\right] \le \varepsilon\Psi_0$$

$q_{\min} = 0$

This is the same rate that ProxSkip achieves, which does not reduce computational complexity. Moreover, this is the optimal convergence rate.

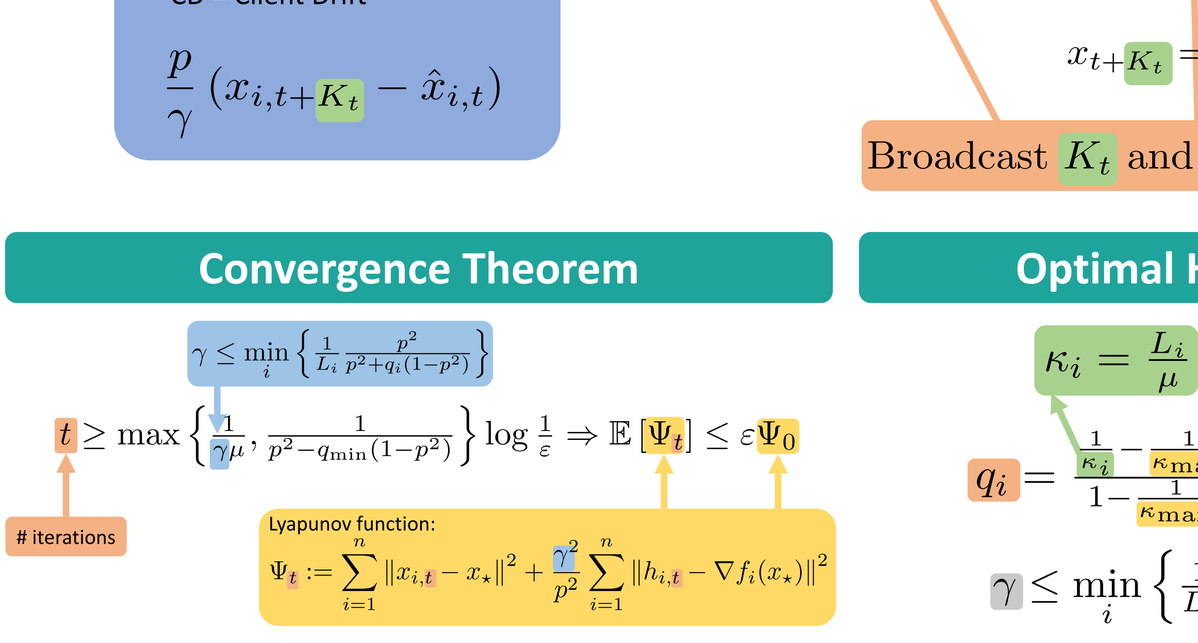
## Experiments

**L2-regularized logistic regression:**

$$f(x) = \frac{1}{n}\sum_{i=1}^{n}\log\left(1+\exp\left(-b_i a_i^\top x\right)\right) + \frac{\lambda}{2}\|x\|^2$$

$$b_i \in \{-1,+1\}, \lambda = 0.1,$$

$$\mathbf{A}_i = \mathbf{U}_i\mathbf{L}_i\mathbf{V}_i \in \mathbb{R}^{200,300}, \sigma_{max}(\mathbf{A}_i) = L_i$$

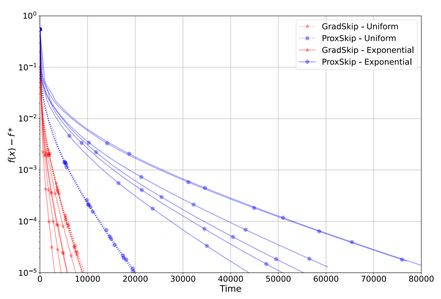



To minimize time complexity, we choose $q_i$ the following way

$$\frac{\mathbb{E}\left[T_i\right]}{1-q_i(1-p)} = \frac{\mathbb{E}\left[T_{min}\right]}{p},$$

where $T_i$ is the time for client $i$ to compute one stochastic gradient.

In the below experiment, we choose $T_i \sim \text{Uniform}(0,1)$ and $T_i \sim \text{Exp}(0,1)$.




## Computational Complexity

The expected number of local steps between two communications for the client $i$ is

$$\frac{\kappa_i(1+\sqrt{\kappa_{\max}})}{\kappa_i+\sqrt{\kappa_{\max}}} \le \min\left\{\kappa_i, \sqrt{\kappa_{\max}}\right\}$$

## Takeaways

- GradSkip is the first Local Training algorithm with SOTA accelerated communication complexity and reduced computational complexity.
- GradSkip can be arbitrarily better than ProxSkip in terms of computational complexity.
- The GradSkip technique can possibly be applied to other distributed or federated learning algorithms.

Paper

Get in touch