Rescaled Asynchronous SGD

Optimal Distributed Optimization under Data and System Heterogeneity

Ammar Mahran, Artavazd Maranjyan, Peter Richtárik

KAUST

Problem Setup

Nonconvex optimization problem:

$$\min_{\mathbf{x}\in\mathbb{R}^d} \left\{ F(\mathbf{x}) := \frac{1}{n}\sum_{i=1}^n F_i(\mathbf{x}) \right\}$$

- Workers ($i = 1, \dots, n$) independently compute stochastic gradients to find a near-stationary point of the loss function
- **Data heterogeneity** = worker i has their own local data and expected loss function

$$F_i(\mathbf{x}) := \mathbb{E}_{\xi_i\sim\mathcal{D}_i}\left[f_i(\mathbf{x};\xi_i)\right]$$

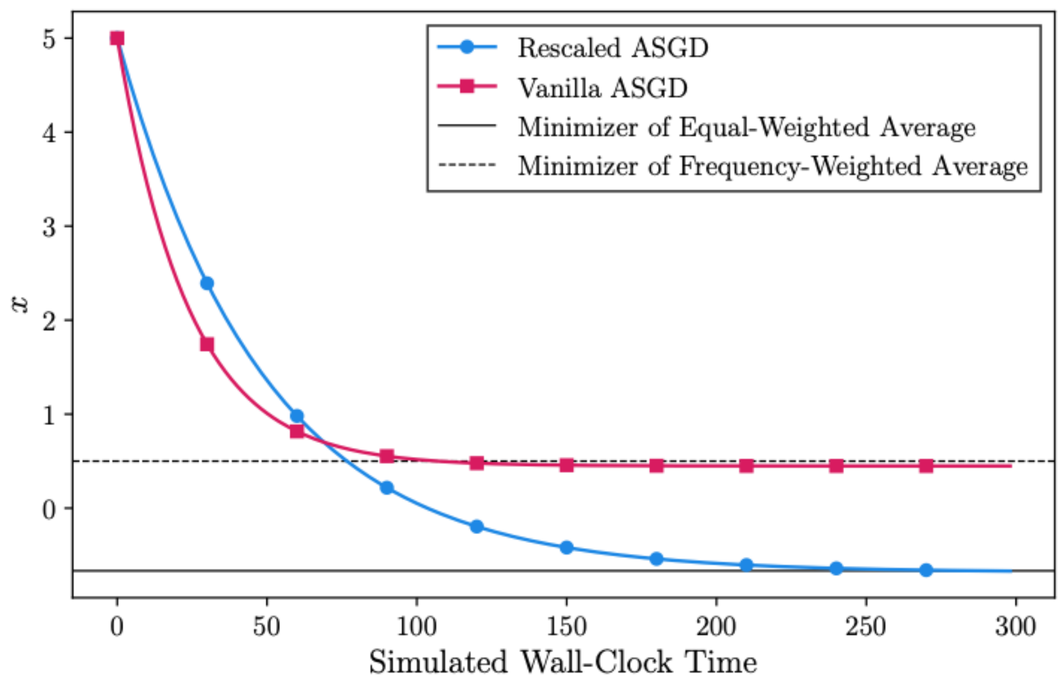
- **System heterogeneity** = worker i needs τ_i seconds to compute a stochastic gradient

Algorithm

***Asynchronous* Stochastic Gradient Descent (SGD)**

Input: initial point $\mathbf{x}_0 \in \mathbb{R}^d$, stepsizes $\gamma_k > 0$
Set $\mathbf{y}_{0,i} = \mathbf{x}_0, \quad \forall i$
▷ $\mathbf{y}_{k,i}$ = worker i's model before update k.
Each worker i begins calculation of $\nabla f_i(\mathbf{y}_{0,i})$
for $k = 0, 1, \dots$ **do**
  Some worker i_k delivers stochastic gradient $\nabla f_{i_k}(\mathbf{y}_{k,i_k})$
  Server updates $\mathbf{x}_{k+1} = \mathbf{x}_k - \gamma_k \nabla f_{i_k}(\mathbf{y}_{k,i_k})$
  ▷ Model update with stale gradient.
  Update worker model:
    $\mathbf{y}_{k+1,i_k} = \mathbf{x}_{k+1}$
    ▷ Update model held by i_k.
    $\mathbf{y}_{k+1,j} = \mathbf{y}_{k,j}, \quad \forall j \neq i_k$
    ▷ Models held by other workers unchanged.
  Worker i_k begins calculating $\nabla f_{i_k}(\mathbf{x}_{k+1})$
end for

- Vanilla ASGD ($\gamma_k = const.$) targets frequency-weighted average = wrong objective
- Idea: **Rescale stepsizes** to ensure aggregate stepsizes are equal across workers, $\gamma_k \propto \tau_{i_k}^{-1}$
- Rescaled ASGD targets correct objective



Theoretical Result

Assumptions	
Unbiased Gradients	$\mathbb{E}_{\xi_i\sim\mathcal{D}_i}[\nabla f_i(\mathbf{x};\xi_i)] = \nabla F_i(\mathbf{x})$
Bounded Variance	$\mathbb{E}_{\xi_i\sim\mathcal{D}_i}[\|\nabla f_i(\mathbf{x};\xi_i) - \nabla F_i(\mathbf{x})\|^2] \le \sigma^2$
Global Smoothness	$\|\nabla F(\mathbf{x}) - \nabla F(\mathbf{y})\| \le L\|\mathbf{x}-\mathbf{y}\|$
Boundedness	$\Delta := F(\mathbf{x}^0) - \inf_{\mathbf{x}} F(\mathbf{x}) < \infty$
Local Smoothness	$\|\nabla F_i(\mathbf{x}) - \nabla F_i(\mathbf{y})\| \le L_{\max}\|\mathbf{x}-\mathbf{y}\|$
Bounded Heterogeneity	$\|\nabla F_i(\mathbf{x})\|^2 \le \zeta^2 + \rho^2\|\nabla F(\mathbf{x})\|^2$

Theorem (Time Complexity). *Under the assumptions above, the worst-case wall-clock time complexity to find an ε-stationary point of F is*

optimal leading term

$$\mathcal{O}\left(\frac{\Delta L\sigma^2}{n\varepsilon^2}\tau_A + \frac{\Delta L_{\max}\sqrt{\sigma^2+\zeta^2}}{\varepsilon^{1.5}}\frac{\tau_{\max}}{\tau_H}\tau_A + \frac{\Delta L_{\max}\rho}{\varepsilon}\frac{\tau_{\max}}{\tau_H}\tau_{\max} + \frac{\Delta L}{\varepsilon}\tau_{\max}\right),$$

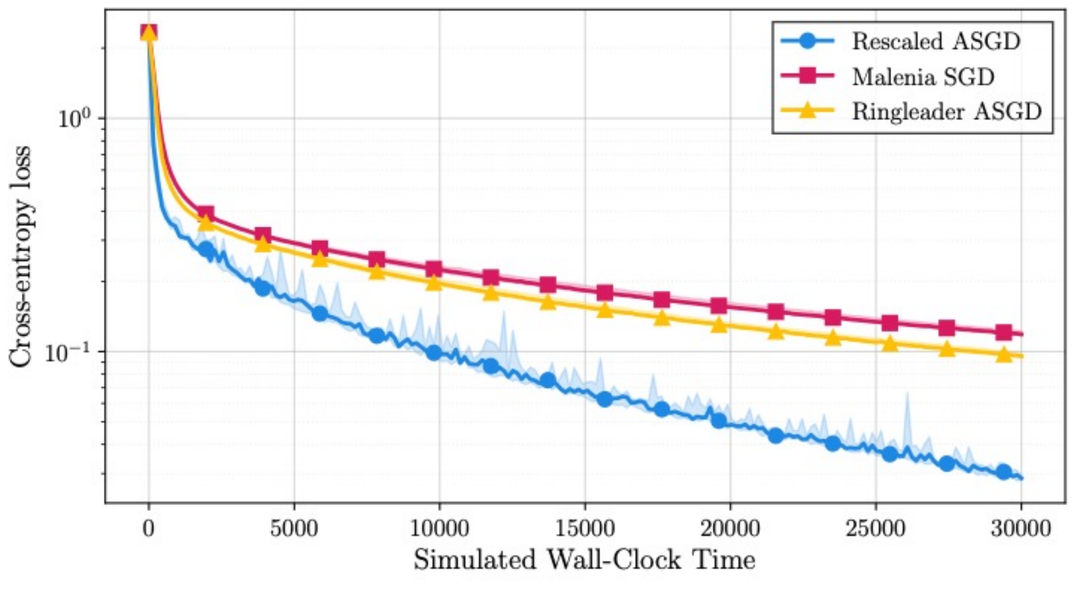
where $\tau_A := \frac{1}{n}\sum_{i=1}^n \tau_i$ is the arithmetic average of workers' computation times.

Comparison

Method (Reference)	Time Complexity (Leading Term)	Asymptotic Optimality	No Idle Workers	No Memory Overhead
Naive Minibatch SGD (Cotter et al., 2011; Dekel et al., 2012)	$\frac{\Delta L\sigma^2}{n\varepsilon^2}\tau_{\max}$	✘	✘	✔
Malenia SGD (Tyurin & Richtárik, 2023)	$\frac{\Delta L\sigma^2}{n\varepsilon^2}\tau_A$	✔	✔	✘
Ringleader ASGD (Maranjyan & Richtárik, 2026)	$\frac{\Delta L\sigma^2}{n\varepsilon^2}\tau_A$	✘	✔	✘
Concurrent ASGD (Koloskova et al., 2022)	$\frac{\Delta L(\sigma^2+\zeta^2)}{n\varepsilon^2}\tau_{\max}$ (†)	✘	✘	✘
Delay-Adaptive ASGD (Mishchenko et al., 2022)	$\frac{\Delta L\sigma^2}{n\varepsilon^2}\tau_A$	✘(†)	✔	✔
Rescaled ASGD **[NEW]**	$\frac{\Delta L\sigma^2}{n\varepsilon^2}\tau_A$	✔	✔	✔

(†) Koloskova et al. (2022); Mishchenko et al. (2022) rely on a stricter assumption on the data heterogeneity, $\|\nabla F_i(\mathbf{x}) - \nabla F(\mathbf{x})\|^2 \le \zeta^2$. In Concurrent ASGD, the constant ζ^2 affects the leading term of the time complexity. In Delay-Adaptive ASGD, the gradient norm can be bounded only up to this constant, therefore not achieving arbitrary ε-stationarity, and thus no asymptotic optimality.

- Asymptotically **optimal time complexity, no idle workers, no memory overhead**
- Competitive performance on MNIST benchmark



SWISS OPTIMIZATION SYMPOSIUM

جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

EPFL

arXiv: 2605.13434