A Simple Proximal Stochastic Gradient Method for Nonsmooth Nonconvex Optimization

Zhize Li, Jian Li

IIIS, Tsinghua University

https://zhizeli.github.io/

Dec 6th, NeurIPS 2018

Problem Definition

Machine learning problems, such as image classification or voice recognition, are usually modeled as a (nonconvex) optimization problem:

$$\min_{\theta} L(\theta).$$

Goal: find a good enough solution (parameters) $\hat{\theta}$, e.g., $\|\nabla L(\hat{\theta})\|^2 \leq \epsilon$

Problem Definition

We consider the more general **nonsmooth nonconvex** case:

$$\min_x \Phi(x) := f(x) + h(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) + h(x),$$

Where $f(x)$ and all $f_i(x)$ are possibly nonconvex (loss on data samples), and $h(x)$ is nonsmooth but convex (e.g., l_1 regularizer $\|x\|_1$ or indicator function $I_C(x)$ for some convex set C).

Problem Definition

We consider the more general **nonsmooth nonconvex** case:

$$\min_x \Phi(x) := f(x) + h(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x) + h(x),$$

Where $f(x)$ and all $f_i(x)$ are possibly nonconvex (loss on data samples), and $h(x)$ is nonsmooth but convex (e.g., l_1 regularizer $\|x\|_1$ or indicator function $I_C(x)$ for some convex set C).

Benefit of $h(x)$: try to deal with the nonsmooth and constrained problems.

Our Results

We propose a simple **ProxSVRG+** algorithm, which **recovers/improves** several previous results (e.g., ProxGD, ProxSVRG/SAGA, SCSG).

Our Results

We propose a simple **ProxSVRG+** algorithm, which **recovers/improves** several previous results (e.g., ProxGD, ProxSVRG/SAGA, SCSG).

Benefits: simpler algorithm, simpler analysis, better theoretical results,

Our Results

We propose a simple **ProxSVRG+** algorithm, which **recovers/improves** several previous results (e.g., ProxGD, ProxSVRG/SAGA, SCSG).

Benefits: simpler algorithm, simpler analysis, better theoretical results, more attractive in practice (prefers moderate minibatch size, auto-adapt to local curvature, i.e., auto-switch to faster linear convergence $O(\cdot \log 1/\epsilon)$ in that regions although the objective function is generally nonconvex).

Theoretical Results

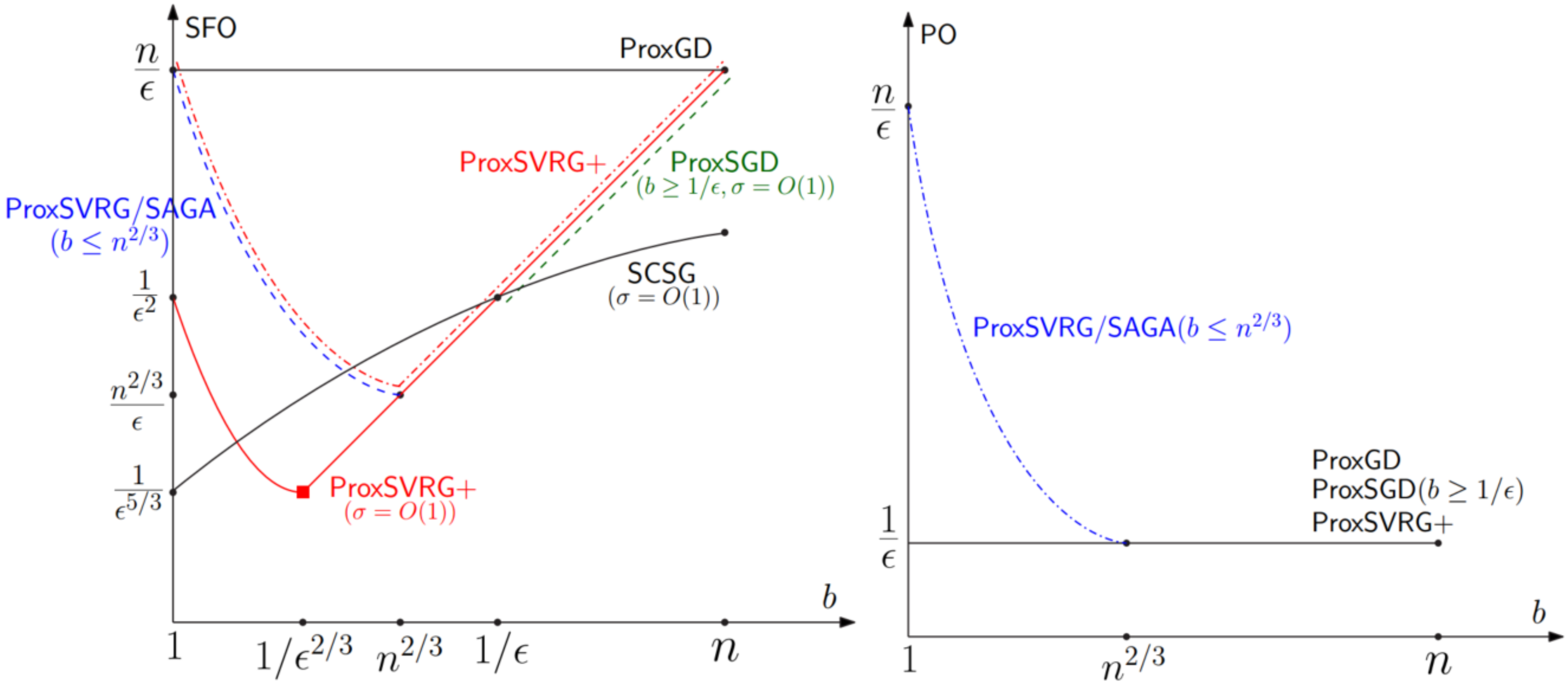

Figure 1: Stochastic first-order oracle (SFO) and proximal oracle (PO) complexity wrt. minibatch size b

Our ProxSVRG+ prefers moderate minibatch size (red box) which is not too small for parallelism or vectorization and not too large for better generalization,

Theoretical Results

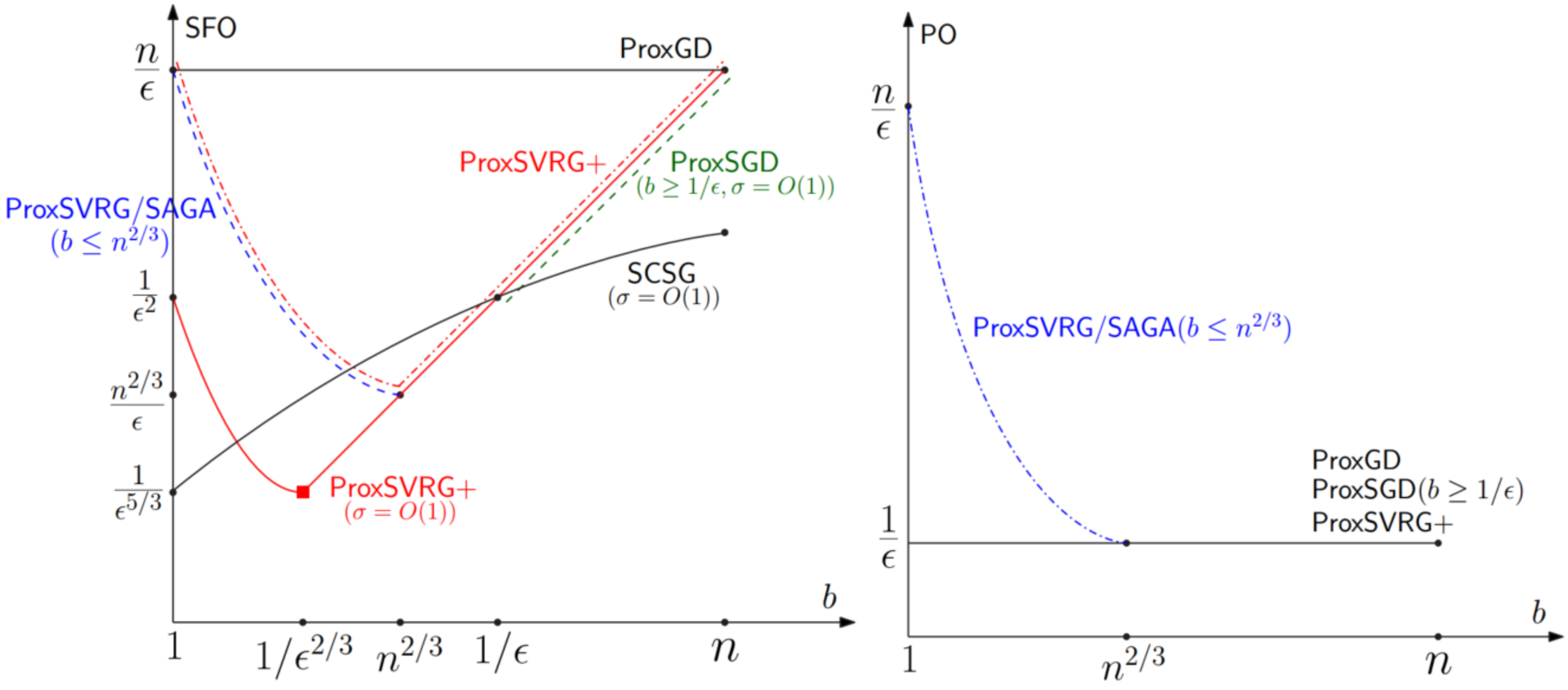

Figure 1: Stochastic first-order oracle (SFO) and proximal oracle (PO) complexity wrt. minibatch size b

Our ProxSVRG+ prefers moderate minibatch size (red box) which is not too small for parallelism or vectorization and not too large for better generalization,

and uses less PO calls than ProxSVRG.

Theoretical Results

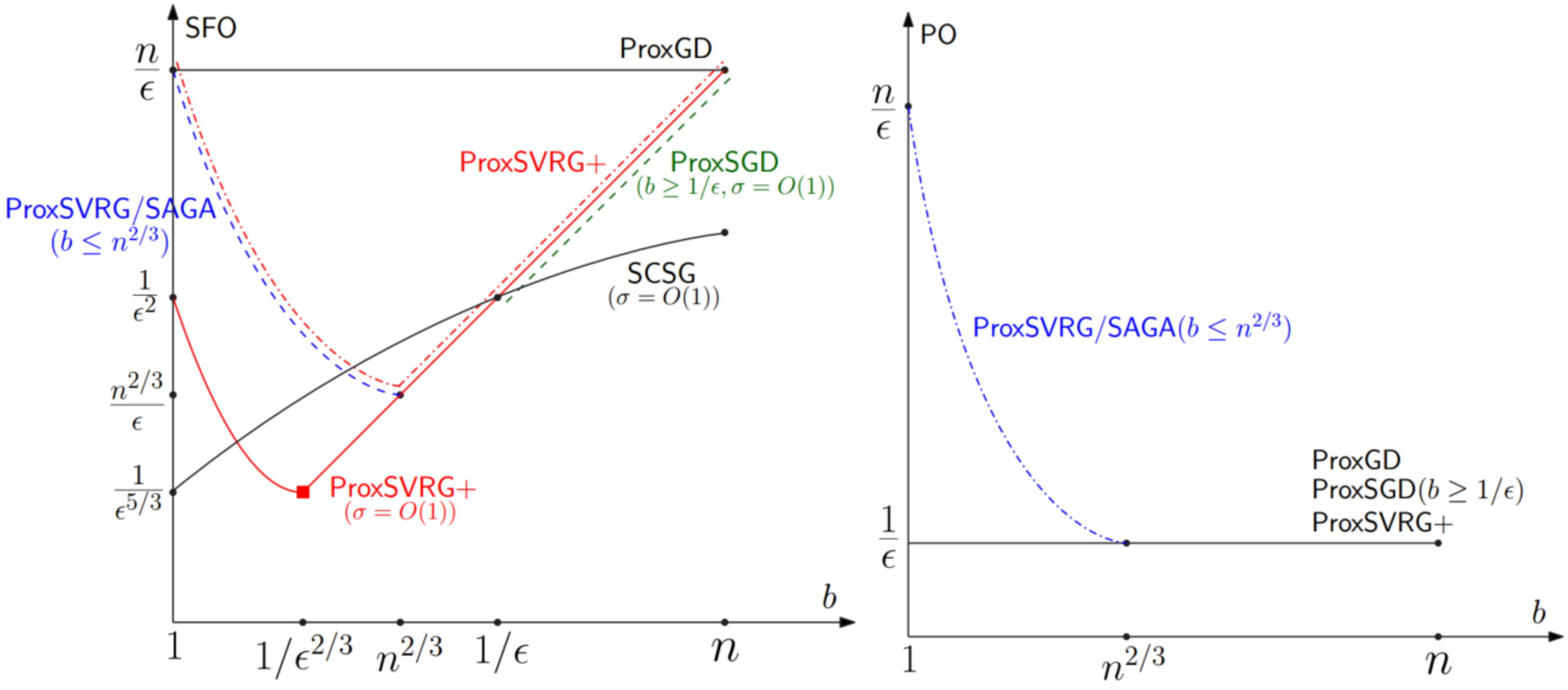

Figure 1: Stochastic first-order oracle (SFO) and proximal oracle (PO) complexity wrt. minibatch size b

Our ProxSVRG+ prefers moderate minibatch size (red box) which is not too small for parallelism or vectorization and not too large for better generalization,

and uses less PO calls than ProxSVRG.

Recently, [Zhou et al., 2018] and [Fang et al., 2018] improve the SFO to $O(n^{1/2}/\epsilon)$ in the smooth setting.

Experimental Results

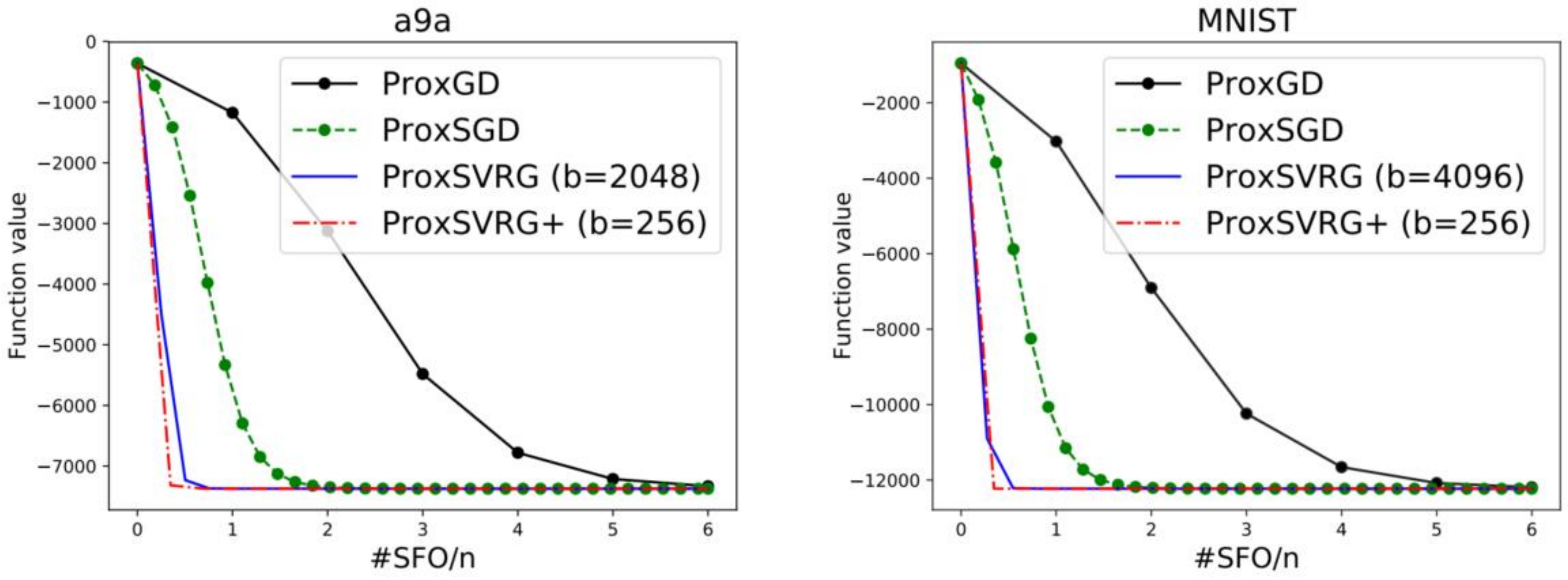

Figure 2: Performance of different algorithms under best minibatch size b

Our ProxSVRG+ prefers much smaller minibatch size than ProxSVRG [Reddi et al., 2016], and performs much better than ProxGD and ProxSGD [Ghadimi et al., 2016].

Thanks!

Our Poster:
5:00-7:00 PM
Room 210 #5