Mini-Batch Semi-Stochastic Gradient Descent in the Proximal Setting

The University of Edinburgh · NAIS Centre for Numerical Algorithms & Intelligent Software · Lehigh University · COR@L Computational Optimization Research at Lehigh · ISE

Jakub Konečný   Jie Liu   Peter Richtárik   Martin Takáč

1. Problem Description

The problem is to minimize a sum of two convex functions,

$$\min_{x\in\mathbb{R}^d}\{P(x) := f(x) + R(x)\}, \tag{1}$$

where f is the average of a large number of smooth convex functions $f_i(x)$, i.e.,

$$f(x) = \tfrac{1}{n}\textstyle\sum_{i=1}^n f_i(x).$$

2. Assumptions

Assumption 1. *The regularizer $R:\mathbb{R}^d\to\mathbb{R}\cup\{+\infty\}$ is convex and closed. The functions $f_i:\mathbb{R}^d\to\mathbb{R}$ are differentiable and have **Lipschitz continuous gradients with constant** $L>0$, i.e., $\forall x,y\in\mathbb{R}^d$,*

$$\|\nabla f_i(x) - \nabla f_i(y)\| \le L\|x-y\|, \quad \textit{where } \|\cdot\| \textit{ is L2 norm.}$$

Assumption 2. *P is **strongly convex with parameter** $\mu>0$, i.e., $\forall x,y\in$ dom(P),*

$$P(y) \ge P(x) + \xi^T(y-x) + \frac{\mu}{2}\|y-x\|^2, \quad \forall \xi\in\partial P(x), \tag{2}$$

where $\partial P(x)$ is the subdifferential of P at x.

3. The Algorithm (mS2GD)

Algorithm 1 mS2GD

1: **Input:** m (max # of stochastic steps per epoch); $h>0$ (stepsize); $x_0\in\mathbb{R}^d$ (starting point); minibatch size $b\in\{1,\dots,n\}$
2: **for** $k=0,1,2,\dots$ **do**
3: Compute and store $g_k \leftarrow \nabla f(x_k) = \frac{1}{n}\sum_i \nabla f_i(x_k)$
4: Initialize the inner loop: $y_{k,0}\leftarrow x_k$
5: Let $t_k \leftarrow t\in\{1,2,\dots,m\}$ uniformly at random
6: **for** $t=0$ to t_k-1 **do**
7: Choose mini-batch $A_{kt}\subset\{1,\dots,n\}$ of size b, uniformly at random
8: Compute a stoch. estimate of $\nabla f(y_{k,t})$:
$v_{k,t}\leftarrow g_k + \frac{1}{b}\sum_{i\in A_{kt}}(\nabla f_i(y_{k,t}) - \nabla f_i(x_k))$
9: $y_{k,t+1}\leftarrow \mathrm{prox}_{hR}(y_{k,t} - hv_{k,t})$
10: **end for**
11: Set $x_{k+1}\leftarrow y_{k,t_k}$
12: **end for**

This is a simplified case of our original algorithm. A complete version of the algorithm and convergence result, with known lower bounds of the convexity parameters ν_F, ν_R for F and R respectively, requires a non-uniform distribution for the number of steps per epoch [1].

4. Convergence Result

Theorem 1. *Let Assumptions 1 and 2 be satisfied and let $x_* \overset{def}{=} \arg\min_x P(x)$. In addition, assume that the stepsize satisfies $0<h<\min\{\frac{1}{4L\alpha(b)}, \frac{1}{L}\}$ and that m is sufficiently large so that*

$$\rho \overset{def}{=} \frac{1}{m\eta\mu(1-4\eta L\alpha(\mathbf{b}))} + \frac{4\eta L\alpha(\mathbf{b})(m+1)}{m(1-4\eta L\alpha(\mathbf{b}))} < 1, \tag{3}$$

where $\alpha(\mathbf{b}) = \frac{\mathbf{n-b}}{\mathbf{b(n-1)}}$. Then mS2GD has linear convergence in expectation:

$$\mathbf{E}(P(x_k) - P(x_*)) \le \rho^k(P(x_0) - P(x_*)).$$

The following bound of variance is considered crucial:

$$\mathbf{E}\left[\|v_{k,t} - \nabla F(y_{k,t})\|^2\right] \le 4\alpha(\mathbf{b})L(P(y_{k,t}) - P(x_*) + P(x_k) - P(x_*)).$$

5. Mini-Batch Speedup

Theorem 2. *Fix target $\rho\in(0,1)$ and the mini-batch size b. Let us define*

$$\tilde{h}^b := \sqrt{\left(\frac{1+\rho}{\rho\mu}\right)^2 + \frac{1}{4\mu\alpha(\mathbf{b})L}} - \frac{1+\rho}{\rho\mu}.$$

Then the optimal step size h_*^b and the maximum size of inner loop m_*^b — which minimizes the number of gradient evaluation while keeping sufficient overall decrease — are given as follows:
If $\tilde{h}^b \le \frac{1}{L}$ then $h_*^b = \tilde{h}^b$ and

$$\mathbf{m_*^b} = 8\alpha(\mathbf{b})L\frac{1+\rho+\sqrt{\frac{1}{4\alpha(\mathbf{b})L}\mu\rho^2 + (1+\rho)^2}}{\mu\rho^2}. \tag{4}$$

Otherwise $h_*^b = \frac{1}{L}$ and $\mathbf{m_*^b} = \frac{L/\mu + 4\alpha(\mathbf{b})}{\rho - 4\alpha(\mathbf{b})(1+\rho)}$.

If $\mathbf{m_*^b} \le \mathbf{m_*^1/b}$, then we can reach the **same** accuracy with **fewer** gradient evaluations. Equation (4) shows that as long as the condition $\tilde{h}^b\le\frac{1}{L}$ is satisfied, m_*^b is decreasing at a rate roughly faster than $1/b$. Hence, we can attain the same accuracy with less work, compared to the case when $b=1$.

6. Numerical Experiments

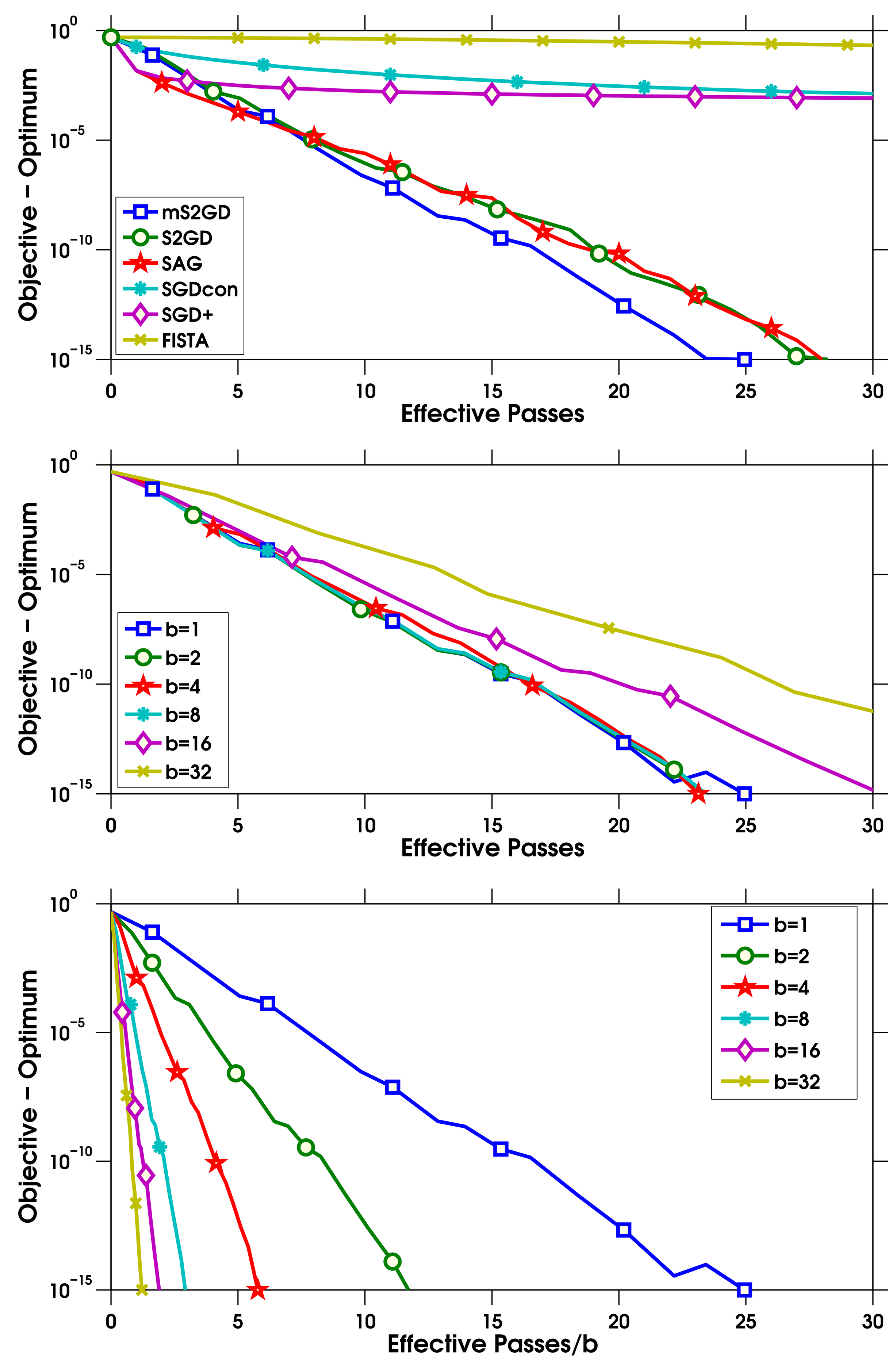

Figure 1: rcv1 dataset, logistic regression, $R(x) = \frac{1}{2n}\|x\|^2$.

TOP: Comparison between mS2GD and the other relevant algorithms implies its competitiveness. SGDcon is by using constant step-size in hindsight and SGD+ is the one with adaptive step-size $h = h_0/(k+1)$, where k is the number of effective passes.

MIDDLE: We compare mS2GD algorithm for different mini-batch sizes with the best parameters m and h for each batch size.

BOTTOM: We present the ideal speedup by parallelism — that would happen if we could always efficiently evaluate the b gradients in parallel, thus being b times faster.

7. References

[1] Jakub Konečný, Jie Liu, Peter Richtárik and Martin Takáč: mS2GD: Mini-Batch Semi-Stochastic Gradient Descent in the Proximal Setting, OPT 2014 @NIPS.
[2] Jakub Konečný and Peter Richtárik: Semi-Stochastic Gradient Descent Methods, arXiv 1312.1666, 2013.
[3] Lin Xiao and Tong Zhang: A Proximal Stochastic Gradient Method with Progressive Variance Reduction, arXiv 1403.4699, 2014.