New Insight into Hybrid Stochastic Gradient Descent: Beyond With-Replacement Sampling and Convexity

Pan Zhou[*] **Xiaotong Yuan**[†] **Jiashi Feng**[*]
[*] Learning & Vision Lab, National University of Singapore, Singapore
[†] B-DAT Lab, Nanjing University of Information Science & Technology, Nanjing, China
pzhou@u.nus.edu xtyuan@nuist.edu.cn elefjia@nus.edu.sg

Abstract

As an incremental-gradient algorithm, the hybrid stochastic gradient descent (HSGD) enjoys merits of both stochastic and full gradient methods for finite-sum problem optimization. However, the existing rate-of-convergence analysis for HSGD is made under with-replacement sampling (WRS) and is restricted to convex problems. It is not clear whether HSGD still carries these advantages under the common practice of without-replacement sampling (WoRS) for non-convex problems. In this paper, we affirmatively answer this open question by showing that under WoRS and for both convex and non-convex problems, it is still possible for HSGD (with constant step-size) to match full gradient descent in rate of convergence, while maintaining comparable sample-size-independent incremental first-order oracle complexity to stochastic gradient descent. For a special class of finite-sum problems with linear prediction models, our convergence results can be further improved in some cases. Extensive numerical results confirm our theoretical affirmation and demonstrate the favorable efficiency of WoRS-based HSGD.

1 Introduction

We consider the following *finite-sum minimization problem*:

$$\min_{\boldsymbol{x}\in\boldsymbol{\mathcal{X}}} f(\boldsymbol{x}) := \frac{1}{n}\sum\nolimits_{i=1}^{n} f_i(\boldsymbol{x}), \tag{1}$$

where each individual $f_i(\boldsymbol{x})$ is ℓ-smooth and the feasible set $\boldsymbol{\mathcal{X}} \subseteq \mathbb{R}^d$ is convex. In the field of machine learning, formulation (1) encapsulates a lot of optimization problems, *e.g.* least square regression and logistic regression. Such a problem can be solved by various algorithms, *e.g.* full gradient descent (FGD) [1], stochastic GD (SGD) [2], hybrid SGD [3], SDCA [4] and SVRG [5].

In this paper, we are particularly interested in Hybrid SGD (HSGD) [3, 6] which is an inexact gradient method that iteratively samples an evolving mini-batch of the terms in (1) for gradient estimation. The iteration of HSGD is given by

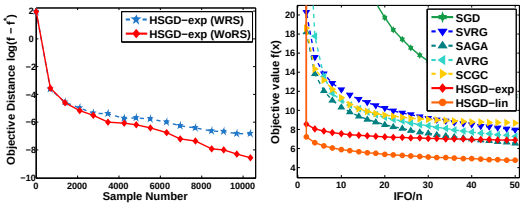
$$\boldsymbol{x}^{k+1} = \Phi_{\boldsymbol{\mathcal{X}}}\left(\boldsymbol{x}^k - \eta_k \boldsymbol{g}^k\right), \text{with } \boldsymbol{g}^k = \frac{1}{s_k}\sum\nolimits_{i_k\in\mathcal{S}_k} \nabla f_{i_k}(\boldsymbol{x}^k),$$

where $\Phi_{\boldsymbol{\mathcal{X}}}(\cdot)$ denotes the Euclidean projection onto $\boldsymbol{\mathcal{X}}$, η_k is the learning rate, and $\mathcal{S}_k$ denotes the set of the s_k selected samples at the k-th iteration. In early iterations, HSGD selects a few samples to compute the full gradient approximately; and along with more iterations, s_k is increased gradually, leading to more accurate full gradient estimation. Such a mechanism allows HSGD to simultaneously enjoy the merits of both SGD and FGD, *i.e.* rapid initial process of SGD and constant learning rate η_k without sacrificing the convergence rate of FGD [6].

Motivation. Though HSGD has been shown, both in theory and practice, to bridge smoothly the gap between full and stochastic gradient descent methods, its rate-of-convergence analysis remains restrictive in several aspects.

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

First, the convergence behavior of HSGD under *without-replacement* sampling (WoRS) is not clear. In the existing analysis [6], the stochastic gradient is assumed to be computed under with-replacement sampling (WRS). But for stochastic optimization, it is a more common practice to use WoRS, *i.e.*, to pass the loss functions $f_i(\boldsymbol{x})$ sequentially, after random shuffling, without re-visiting any of them [7, 8]. This makes significant discrepancy between the theoretical guarantee and practical implementation. As shown in Figure 1 (a), WoRS tends to provide better performance than WRS in actual implementation.



(a) (b)

Figure 1: Comparison of WoRS-based HSGD. (a) WoRS *vs.* WRS in HSGD: optimizing a softmax regression model with a single full pass over the data letter. (b) Comparison among randomized algorithms for optimizing a feedforward neural network with 50 full passes over the data sensorless. HSGD-exp and HSGD-lin respectively denote WoRS based HSGD with exponentially and linearly increasing mini-batch sizes (ref. Section 3.2 and 3.4). See more results in supplement.

Second, the convergence behavior of HSGD for non-convex problems is not clear. Prior convergence guarantees on HSGD are limited to convex problems. Bertsekas [3] established linear convergence of HSGD for least square problems. Friedlander *et al.* [6] proved that HSGD converges linearly for strongly convex problems with exponentially increasing s_k, and sub-linearly for arbitrary convex problems with polynomially increasing s_k. Unfortunately, non-convex convergence guarantee on HSGD is still absent, though highly desirable in machine learning applications and extensively studied in other stochastic algorithms, *e.g.* SVRG [9, 10]. In Figure 1 (b), HSGD has sharper convergence behavior than several state-of-the-art SGD methods in training neural networks.

Third, the Incremental First-order Oracle (IFO) complexity (*i.e.* stochastic gradient computation; see Definition 2) of HSGD is largely left unknown. Although Friedlander *et al.* [6] showed that HSGD maintains steady convergence rates of FGD, its IFO complexity is not explicitly analyzed, making it less clear where HSGD should be positioned w.r.t. existing stochastic gradient approaches in overall computational complexity.

Summary of contributions. In this work, we address the aforementioned three limitations in the existing analysis of HSGD. We analyze the rate-of-convergence of HSGD under WoRS in a wide problem spectrum including strongly convex, non-strongly convex and non-convex problems. Table 1 summarizes our main results on IFO complexity of HSGD (WoRS) and compares them against state-of-the-art WoRS-oriented results for (stochastic) gradient methods. These results are divided into two groups: for general problems and for a special class of problems with linear prediction loss $f_i(\boldsymbol{x}) = h(\boldsymbol{a}_i^\top \boldsymbol{x})$. As shown in the bottom row of Table 1, we contribute several new theoretical insights into HSGD, which are elaborated in the following paragraphs.

The bounds highlighted in *green*: For both general and certain specially structured strongly convex problems, HSGD is $n\times$ faster than FGD. Compared to the results for SAGA and AVRG [12], the IFO complexity of HSGD is independent of the sample size n but more dependent on $1/\epsilon$. This suggests that HSGD will converge faster when n dominates $1/\epsilon$. Finally, compared to the results for SGD in linear prediction problems [11], ours has removed the dependency on the logarithm term $\log(\kappa/\epsilon)$.

The bounds highlighted in *red*: To our best knowledge, for the first time these new results establish guarantees on WoRS-based stochastic approaches for non-strongly convex and non-convex problems.

The bounds highlighted in *blue*: If the loss function $h(\boldsymbol{a}_i^\top \boldsymbol{x})$ in the specially structured problem is strongly convex in terms of $\boldsymbol{a}_i^\top \boldsymbol{x}$ (but $f(\boldsymbol{x})$ may be non-strongly convex), HSGD has $\mathcal{O}(1/\epsilon)$ IFO complexity. The least square regression and logistic regression (with a bounded feasible set) models have such a linear prediction structure.

The bounds highlighted in *brown*: When the specially structured problem is non-strongly convex, HSGD converges to the minimum of problem (1), while SGD can only be shown to converge to a sub-optimum up to some statistical error (see footnote 2 below Table 1).

Related work. Understanding randomized algorithms under WoRS and random reshuffling is gaining considerable attention in recent years. By focusing on least squares problems, Recht *et al.* [13] utilized arithmetic-mean inequality on matrices to show that for randomized algorithms, WoRS is always faster than WRS if the data are randomly generated from a certain distribution. For more general

Table 1: Comparison of IFO complexity for randomized algorithms under WoRS. $\kappa = \ell/\rho$ denotes the condition number of ℓ-smooth and ρ-strong convex cases for problem (1). Best viewed in color.

	General Problem			Specially Structured Problem with $f_i(\boldsymbol{x}) = h(\boldsymbol{a}_i^\top \boldsymbol{x})$		
	Stro. conv.	Non-Stro. conv.	Non-conv.	$f(\cdot)$ is stro. conv.	$f(\cdot)$ is non-stro. conv.	$h(\cdot)$ is stro. conv.
Metric: $\mathbb{E}\|\boldsymbol{x}^a - \boldsymbol{x}^*\|_2^2 \leq \epsilon$ for stro. conv., $\mathbb{E}[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \epsilon$ for non-stro. conv., $\mathbb{E}\|\nabla f(\boldsymbol{x}^a)\|_2^2 \leq \epsilon$ for non-conv.						
FGD [8]	$\mathcal{O}\big(\frac{n\kappa^2}{\epsilon}\big)$	—	—	$\mathcal{O}\big(\frac{n\kappa^2}{\epsilon}\big)$	—	—
SAGA [12]	$\mathcal{O}\big(n\kappa^2 \log\big(\frac{1}{\epsilon}\big)\big)$	—	—	$\mathcal{O}\big(n\kappa^2 \log\big(\frac{1}{\epsilon}\big)\big)$	—	—
AVRG [12]	$\mathcal{O}\big(n\kappa^2 \log\big(\frac{1}{\epsilon}\big)\big)$	—	—	$\mathcal{O}\big(n\kappa^2 \log\big(\frac{1}{\epsilon}\big)\big)$	—	—
HSGD	$\mathcal{O}\big(\frac{\kappa^2}{\epsilon}\big)$	$\mathcal{O}\big(\frac{1}{\epsilon^3}\big)$[1]	$\mathcal{O}\big(\frac{1}{\epsilon^2}\big)$	$\mathcal{O}\big(\frac{\kappa^2}{\epsilon}\big)$	$\mathcal{O}\big(\frac{1}{\epsilon^3}\big)$	$\mathcal{O}\big(\frac{1}{\epsilon}\big)$
Metric: $\mathbb{E}[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \epsilon$ for both stro. and non-stro. conv., $\mathbb{E}\|\nabla f(\boldsymbol{x}^a)\|_2^2 \leq \epsilon$ for non-conv.						
SGD [11]	—	—	—	$\mathcal{O}\big(\frac{\kappa}{\epsilon}\log\big(\frac{\kappa}{\epsilon}\big)\big)$	$\mathcal{O}\big(\frac{1}{\epsilon^2}\big)$[2]	—
HSGD	$\mathcal{O}\big(\frac{\kappa}{\epsilon}\big)$	$\mathcal{O}\big(\frac{1}{\epsilon^3}\big)$[1]	$\mathcal{O}\big(\frac{1}{\epsilon^2}\big)$	$\mathcal{O}\big(\frac{\kappa}{\epsilon}\big)$	$\mathcal{O}\big(\frac{1}{\epsilon^3}\big)$	$\mathcal{O}\big(\frac{1}{\epsilon}\big)$

[1] Our IFO complexity for arbitrary convex cases appears higher than the non-convex ones, as we use sub-optimality metric $\mathbb{E}\,[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \epsilon$ for convex cases while $\mathbb{E}\|\nabla f(\boldsymbol{x}^a)\|_2^2 \leq \epsilon$ for non-convex cases.

[2] Corollary 1 in [11] provides $\mathbb{E}\,[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq R_T/k + 2(12+\sqrt{2}D)/\sqrt{n}$ where D denotes the diameter of the domain $\boldsymbol{\mathcal{X}}$, k is the iteration number and $R_T \sim \mathcal{O}(D\ell/\sqrt{k})$ is the regret bound of SGD for (1). The term $2(12+\sqrt{2}D)/\sqrt{n}$ is a statistical error which is an artifact from the regret analysis approach.

smooth and strongly convex problems, Gürbüzbalaban *et al.* [8] proved that gradient descent based on random reshuffling enjoys $\mathcal{O}(1/k^2)$ rate of convergence after k epoches, as opposed to $\mathcal{O}(1/k)$ under WRS. But this analysis does not explicitly explain why WoRS works well after a few (or even just one) passes over the data. To answer such a central question, by leveraging regret analysis, Shamir *et al.* [11] proved that for a special class of loss functions $f_i(\boldsymbol{x}) = h(\boldsymbol{a}_i^\top \boldsymbol{x})$, SGD and SVRG using WoRS can achieve competitive IFO complexity to their WRS counterparts. More recently, Ying *et al.* [12] proved that for strongly convex problems, both SAGA [14] and their proposed AVRG algorithm achieve linear convergence rate with WoRS. Our work differs from these prior works: 1) For the first time, we provide WoRS based theoretical analysis for HSGD. 2) Our analysis covers non-strongly convex and non-convex cases which are not covered by current WoRS theories.

2 Preliminaries

We first introduce the concepts of strong convexity and Lipschtiz smoothness which are commonly used in analyzing stochastic gradient methods [4, 5, 15, 16, 17, 18].

Definition 1 (Strong convexity and Lipschitz smoothness)**.** *We say a function $g(\boldsymbol{x})$ is ρ-strongly-convex if there exists a positive constant ρ such that $\forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \boldsymbol{\mathcal{X}}$, $g(\boldsymbol{x}_1) \geq g(\boldsymbol{x}_2) + \langle \nabla g(\boldsymbol{x}_2), \boldsymbol{x}_1 - \boldsymbol{x}_2 \rangle + \frac{\rho}{2}\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2^2$. Moreover, we say $g(\boldsymbol{x})$ is ℓ-smooth if there exists a positive constant ℓ such that $\|\nabla g(\boldsymbol{x}_1) - \nabla g(\boldsymbol{x}_2)\|_2 \leq \ell \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2$.*

In all our analysis, we will impose the basic Assumption 1 to bound stochastic gradient variance.

Assumption 1 (Bounded gradient)**.** *For each loss $f_i(\boldsymbol{x})$, the distance between its gradient $\nabla f_i(\boldsymbol{x})$ and the full gradient $\nabla f(\boldsymbol{x})$ is upper bounded as $\max_i \|\nabla f_i(\boldsymbol{x}) - \nabla f(\boldsymbol{x})\|_2 \leq G$.*

If $f_i(\boldsymbol{x})$ is ℓ-smooth and the domain of interest $\boldsymbol{\mathcal{X}}$ is bounded, then the bounded gradient assumption can be naturally implied. We explicitly write out this assumption for the sake of notation simplicity. Following [5, 19, 20], we also employ the incremental first order oracle (IFO) complexity as the computational complexity metric for solving the finite-sum minimization problem (1).

Definition 2. *An IFO takes an index $i \in [n]$ and a point $\boldsymbol{x} \in \boldsymbol{\mathcal{X}}$, and returns the pair $(f_i(\boldsymbol{x}), \nabla f_i(\boldsymbol{x}))$.*

The IFO complexity can more accurately reflect the overall computational performance of a first-order algorithm, as objective value and gradient evaluation usually dominate the per-iteration complexity.

3 General Analysis for WoRS HSGD

The WoRS-based HSGD algorithm is outlined in Algorithm 1. Here we systematically analyze its convergence performance for strongly/non-strongly convex and non-convex problems. Similar

Algorithm 1 Hybrid SGD under WoRS

Input: Initial point $\boldsymbol{x}^0$, sample index set $\mathcal{S}=\{1,\cdots,n\}$, learning rate $\{\eta_k\}$, mini-batch size $\{s_k\}$.
for $k=0$ **to** $T-1$ **do**
  Select s_k samples $\mathcal{S}_k$ by WoRS from $\mathcal{S}-\bigcup_{i=0}^{k-1}\mathcal{S}_i$.
  Compute the gradient $\boldsymbol{g}^k=\frac{1}{s_k}\sum_{i_k\in\mathcal{S}_k}\nabla f_{i_k}(\boldsymbol{x}^k)$.
  Update $\boldsymbol{x}^{k+1}=\Phi_{\boldsymbol{\mathcal{X}}}\left(\boldsymbol{x}^k-\eta_k\boldsymbol{g}^k\right)$.
end for
Output: $\boldsymbol{x}^a$ sampled uniformly from $\{\boldsymbol{x}^k\}_{k=0}^{T-1}$ for strong convex and linearly structured problems or $\{\boldsymbol{x}^k\}_{k=\lfloor 0.5T\rfloor}^{T-1}$ for non-strongly/non-convex problems.

to [11], we focus our analysis on the scenario where a single pass (or less) over data is of interest, which occurs, *e.g.* in streaming data analysis. According to our empirical study (see, *e.g.*, Figure 3), running Algorithm 1 for a single pass over data can provide satisfactory accuracy in many cases.

3.1 A key lemma

It is well understood that unbiased gradient estimation with gradually vanishing variance is important for accelerating randomized algorithms [5, 14]. This is because the increasingly more accurate estimate of full gradient allows the algorithm to move ahead with more aggressive step-size to decrease the objective value. However, for WoRS implementation, the mini-batch terms selected at each iteration are no longer statistically independent, leading to biased gradient estimate $\boldsymbol{g}^k$, *i.e.*

$$\mathbb{E}[\boldsymbol{g}^k]=\mathbb{E}\Big[\frac{1}{s_k}\sum\nolimits_{i_k\in\mathcal{S}_k}\nabla f_{i_k}(\boldsymbol{x}^k)\Big]\neq\nabla f(\boldsymbol{x}^k).$$

Such a biased estimate $\boldsymbol{g}^k$ brings a challenge to bounding its variance $\mathbb{E}\|\boldsymbol{g}^k-\nabla f(\boldsymbol{x}^k)\|_2^2$ with common techniques such as Bernstein inequality [21] and those existing bounds on $\mathbb{E}\|\boldsymbol{g}^k-\nabla f(\boldsymbol{x}^k)\|_2^2$ under WRS [22]. To tackle this challenge, we formulate the k-th WoRS as a stochastic process consisting of two phases. In the first phase, we sample s_k' data indexed by $\mathcal{S}_k'=\mathcal{S}-\cup_{i=0}^{k-1}\mathcal{S}_i$ after $k-1$ times of WoRS over all the data, where $s_k'=n-\sum_{i=0}^{k-1}s_i$ and $\mathcal{S}=\{1,2,\cdots,n\}$ denotes the index set of all samples. Then, in the second phase, we sample s_k data from the remaining s_k' samples indexed by $\mathcal{S}_k'$ in a without-replacement fashion. Based on such a WoRS process, we can formulate the two sampling phases as martingales and further bound the gradient variance, giving the following lemma which is key to our WoRS-based convergence analysis in following sections.

Lemma 1. *The gradient $\boldsymbol{g}^k$ estimated by WoRS in Algorithm 1 satisfies* $\mathbb{E}\left[\|\boldsymbol{g}^k-\nabla f(\boldsymbol{x}^k)\|_2^2\right]\leq\frac{24G^2}{s_k}$.

See its proof in Appendix A. Lemma 1 shows that the gradient variance $\mathbb{E}[\|\boldsymbol{g}^k-\nabla f(\boldsymbol{x}^k)\|_2^2]$ in Algorithm 1 is controlled by $1/s_k$. This means that by gradually increasing the mini-batch size, HSGD under WoRS can reduce variance, similar to SVRG and SAGA, but without requiring to integrate historical gradients or full gradient of the snapshot point into current gradient estimate.

By applying Bernstein inequality, Friedlander *et al.* [6] showed that $\mathbb{E}[\|\boldsymbol{g}^k-\nabla f(\boldsymbol{x}^k)\|_2^2]=\mathcal{O}\big(\frac{n-s_k}{ns_k}\big)$ if the s_k samples selected at iteration k are different, but are sampled from the *entire* data set. In contrast, our considered WoRS strategy assumes the s_k different samples are drawn from the *remaining* set $\mathcal{S}-\cup_{i=0}^{k-1}\mathcal{S}_i$, and thus needs to take into account the statistical dependence among iterations to bound the stochastic gradient variance.

3.2 Strongly convex functions

We analyze the convergence behavior of both the computed solution $\boldsymbol{x}$ and the objective $f(\boldsymbol{x})$ under the strongly convex setting. Our convergence result on the computed solution is stated in Theorem 1.

Theorem 1. *Suppose $f(\boldsymbol{x})$ is ρ-strongly-convex and each $f_i(\boldsymbol{x})$ is ℓ-smooth. With learning rate $\eta_k=\frac{\rho}{\ell^2}$ and mini-batch size $s_k=\frac{\tau}{\zeta^k}$ where $\zeta=1-\frac{\rho}{18\ell^2}$ and $\tau\geq\frac{G^2}{\|\boldsymbol{x}^0-\boldsymbol{x}^*\|^2}\max\big(\frac{324}{\rho^2},\frac{432}{\ell^2}\big)$, we have*

$$\mathbb{E}\|\boldsymbol{x}^a-\boldsymbol{x}^*\|_2^2\leq\big(1-\frac{\rho^2}{18\ell^2}\big)^T\|\boldsymbol{x}^0-\boldsymbol{x}^*\|_2^2,$$

where $\boldsymbol{x}^a$ is the output solution of Algorithm 1 and T is the number of iterations.

A proof of this result is given in Appendix B.1. From Theorem 1, if mini-batch size is increased at an exponential rate $\frac{1}{1-\gamma}$ with $\gamma = \frac{\rho}{18\ell^2}$, then the objective in HSGD converges linearly at the rate of $\mathcal{O}\left((1-\gamma)^k\right)$ for strongly convex problems. This implies that HSGD enjoys the merits of both SGD and FGD. Specifically, similar to SGD, the per-iteration computation of HSGD is cheap as it is free of computing the full gradient $\nabla f(\boldsymbol{x})$. Meanwhile, it uses a constant learning rate and enjoys the steady convergence rate of FGD. As the condition number $\kappa = \ell/\rho$ is usually large in realistic problems, the exponential rate $\frac{1}{1-\gamma}$ is actually only a little larger than one. This means even a moderate-scale dataset allows a lot of HSGD iterations which decreases the loss sufficiently as shown in Figure 2 and 3. Friedlander *et al.* [6] proved that HSGD has linear convergence rate under WRS. Theorem 1 generalizes the result to WoRS. Then we can derive the IFO complexity of HSGD for strongly-convex problems in the following corollary, for which proof is given in Appendix B.2.

Corollary 1. *Suppose the assumptions in Theorem 1 hold. To achieve $\mathbb{E}\|\boldsymbol{x}^a - \boldsymbol{x}^*\|_2^2 \leq \epsilon$, the IFO complexity of HSGD is $\mathcal{O}\left(\frac{\kappa^2 G^2}{\epsilon}\right)$ where $\kappa = \frac{\ell}{\rho}$ denotes the condition number of the objective $f(\boldsymbol{x})$.*

From Corollary 1, the IFO complexity of HSGD for strongly convex problems is at the order of $\mathcal{O}\left(\frac{\kappa^2}{\epsilon}\right)$, which is independent of the sample size n. So when n dominates $\frac{1}{\epsilon}$, HSGD can be superior to the algorithms with complexity linearly relying on n, such as SVRG and SAGA.

Gürbüzbalaban *et al.* [8] showed that by processing each individual $f_i(\boldsymbol{x})$ with random shuffling at each iteration and adopting a diminishing learning rate $\eta_k = \mathcal{O}\left(\frac{1}{k^\beta}\right)$ with $\beta \in (\frac{1}{2}, 1)$, the IFO complexity of FGD is $\mathcal{O}\left(\kappa^2 \frac{n}{\epsilon}\right)$ for achieving $\mathbb{E}\|\boldsymbol{x}^a - \boldsymbol{x}^*\|_2^2 \leq \epsilon$. So HSGD is n times faster than FGD. This is because at each iteration, unlike FGD requiring to access all data, HSGD only samples a mini-batch for gradient estimation without sacrificing convergence rate. Ying *et al.* [12] proved that under WoRS, both SAGA and AVRG converge linearly and have IFO complexity of $\mathcal{O}\left(n\kappa^2 \log\left(\frac{1}{\epsilon}\right)\right)$. Hence, HSGD will outperform SAGA and AVRG if n dominates $\frac{1}{\epsilon}$, which is usually the case when the data scale is huge while the desired accuracy ϵ is moderately small (*e.g.* 10^{-5}).

Shamir [11] proved that for linearly structured problems, SGD under WoRS has IFO complexity $\mathcal{O}\left(\frac{\kappa}{\epsilon}\log\left(\frac{\kappa}{\epsilon}\right)\right)$ by measuring the objective (see Section 4). Here we can also establish the shaper convergence behavior of the objective. The result is presented in Theorem 2 with proof in Appendix B.3.

Theorem 2. *Assume $f(\boldsymbol{x})$ is ρ-strongly-convex and each $f_i(\boldsymbol{x})$ is ℓ-smooth. With learning rate $\eta_k = \frac{1}{\ell}$ and mini-batch size $s_k = \frac{\tau}{\zeta^k}$ where $\zeta = 1 - \frac{\rho}{2\ell}$ and $\tau \geq \frac{6G^2}{\rho[f(\boldsymbol{x}^0) - f(\boldsymbol{x}^*)]}$, the output $\boldsymbol{x}^a$ of Algorithm 1 satisfies*

$$\mathbb{E}\left[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)\right] \leq \left(1 - \frac{\rho}{2\ell}\right)^T \left(f(\boldsymbol{x}^0) - f(\boldsymbol{x}^*)\right).$$

Moreover, to achieve $\mathbb{E}[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \epsilon$, the IFO complexity of HSGD is $\mathcal{O}\left(\frac{\kappa G^2}{\epsilon}\right)$, where $\kappa = \frac{\ell}{\rho}$.

Theorem 2 shows that HSGD also enjoys linear convergence rate on the objective by using exponentially mini-batch size. But it has lower complexity $\mathcal{O}\left(\frac{\kappa}{\epsilon}\right)$ under the measurement $\mathbb{E}[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \epsilon$ which is in contrast to the complexity $\mathcal{O}\left(\frac{\kappa^2}{\epsilon}\right)$ for achieving $\mathbb{E}\|\boldsymbol{x}^a - \boldsymbol{x}^*\|_2^2 \leq \epsilon$. This is because the objective analysis allows to use more aggressive step-size $\frac{1}{\ell}$, while the analysis on the solution requires smaller learning rate $\frac{\rho}{\ell^2}$. In this way, HSGD with larger step-size converges faster.

3.3 Non-strongly convex functions

We proceed to analyze the convergence performance of HSGD for non-strongly convex problems. Our result for this case is summarized in Theorem 3 with proof in Appendix B.4. To our best knowledge, this is the first convergence guarantee of WoRS-based methods for *non-strongly* convex problems.

Theorem 3. *Suppose $f(\boldsymbol{x})$ is convex and each $f_i(\boldsymbol{x})$ is ℓ-smooth. Assume that $\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2 \leq D$ holds for $\forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \boldsymbol{\mathcal{X}}$. Then with the learning rate $\eta_k = \frac{1}{2\ell}$ and mini-batch size $s_k = (k+1)^2$, we have*

$$\mathbb{E}[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \frac{4\ell D^2 + 24GD}{T} + \frac{48G^2}{\ell T^2},$$

where $\boldsymbol{x}^a$ denotes the output solution of Algorithm 1 and T is the number of iterations.

Theorem 3 shows that if one expands the mini-batch size at $\mathcal{O}\left(k^2\right)$, then the convergence rate of HSGD under WoRS is $\mathcal{O}\left(\frac{1}{T}\right)$. In [11], a sub-linear rate was established for WoRS-based SGD

in a special class of convex problems with $f_i(\boldsymbol{x}) = h_i(\langle \boldsymbol{a}_i, \boldsymbol{x}\rangle)$. A detailed comparison between their result and ours for such a structured formulation will be discussed in Section 4. Under the assumption $\sum_{k=0}^{+\infty} \|\boldsymbol{g}^k - \nabla f(\boldsymbol{x}^k)\|_2 < +\infty$, Friedlander *et al.* [6] showed that WRS-based HSGD outputs $f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*) = \mathcal{O}\left(\frac{1}{T}\right)$. However, such an assumption holds only if HSGD selects at least $\mathcal{O}\left(k^2\right)$ samples at the k-th iteration due to $\mathbb{E}[\|\boldsymbol{g}^k - \nabla f(\boldsymbol{x}^k)\|_2^2] = \mathcal{O}\left(\frac{n-s_k}{ns_k}\right)$. In this way, their result under WRS is the same as ours under WoRS. The following corollary gives the corresponding IFO complexity. A proof of this result is given in Appendix B.5.

Corollary 2. *Suppose the assumptions in Theorem 3 hold. To achieve* $\mathbb{E}[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \epsilon$, *the IFO complexity of HSGD is* $\mathcal{O}\left(\frac{(6GD+\ell D^2)^3}{\epsilon^3}\right)$.

3.4 Non-convex functions

Now we analyze HSGD for general non-convex problems, which to our knowledge has not yet been addressed so far. The result is formally stated in Theorem 4 with proof in Appendix B.6. .

Theorem 4. *Suppose each* $f_i(\boldsymbol{x})$ *is* ℓ*-smooth and for* $\forall \boldsymbol{x}_1, \boldsymbol{x}_2 \in \boldsymbol{\mathcal{X}}$, $\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2 \leq D$. *With learning rate* $\eta_k = \frac{1}{2\ell}$ *and mini-batch size* $s_k = k+1$, *the output* $\boldsymbol{x}^a$ *of Algorithm 1 with* T *iterations satisfies*

$$\mathbb{E}\left[\|\nabla f(\boldsymbol{x}^a)\|_2^2\right] \leq \frac{4\ell^2 D^2 + 35G^2}{T}.$$

Theorem 4 guarantees that for non-convex problems, HSGD exhibits $\mathcal{O}\left(\frac{1}{T}\right)$ rate of convergence by linearly expanding the mini-batch size at each iteration. Here we follow the convention in [9, 10, 22] to adopt the value $\|\nabla f(\boldsymbol{x}^a)\|_2^2$ as a measurement of quality for approximate stationary solutions. Then we drive the IFO complexity of HSGD in the following corollary with proof in Appendix B.7.

Corollary 3. *Suppose the assumptions in Theorem 4 hold. To achieve* $\mathbb{E}\left[\|\nabla f(\boldsymbol{x}^a)\|_2^2\right] \leq \epsilon$, *the IFO complexity of the HSGD in Algorithm 1 is* $\mathcal{O}\left(\frac{(4\ell^2 D^2 + 35G^2)^2}{\epsilon^2}\right)$.

The IFO complexity for non-convex problems looks lower than that for non-strongly convex ones in Corollary 2. This is because we use $\mathbb{E}\left[f(\boldsymbol{x}^a - f(\boldsymbol{x}^*)\right] \leq \epsilon$ as sub-optimality measurement for arbitrary convex problems and $\mathbb{E}\left[\|\nabla f(\boldsymbol{x}^a)\|_2^2\right] \leq \epsilon$ for non-convex problems.

4 Analysis for Linearly Structured Problems

We further consider a special case of problem (1) where each $f_i(\boldsymbol{x})$ has a linear prediction structure:

$$f(\boldsymbol{x}) := \frac{1}{n}\sum\nolimits_{i=1}^{n} f_i(\boldsymbol{x}), \text{ where } f_i(\boldsymbol{x}) = h(\langle \boldsymbol{a}_i, \boldsymbol{x}\rangle). \tag{2}$$

Here $\boldsymbol{a}_i$ denotes the i-th sample vector and $h(\cdot)$ denotes a convex loss function. Such a formulation covers several common problems in machine learning, such as $f_i(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{b}_i - \boldsymbol{a}_i^\top \boldsymbol{x})^2$ for least square regression and $f_i(\boldsymbol{x}) = \log(1 + \exp(-\boldsymbol{b}_i \boldsymbol{a}_i^\top \boldsymbol{x}))$ for logistic regression, where $\boldsymbol{b}_i$ is the real or binary target output. Such a special problem setting has been considered in [11] for analyzing SGD under WoRS. To make a comprehensive comparison, we specify our strongly convex analysis to (2), and improve our non-strongly-convex results when the surrogate loss $h(\cdot)$ is strongly convex.

Strongly convex case. In this case, according to Theorem 2, HSGD converges linearly and its IFO complexity is $\mathcal{O}\left(\frac{\kappa}{\epsilon}\right)$. By comparison, SGD under WoRS in [11] converges at $\mathcal{O}\left(\frac{\kappa}{T}\log\left(\frac{1}{T}\right)\right)$ and has IFO complexity $\mathcal{O}\left(\frac{\kappa}{\epsilon}\log\left(\frac{\kappa}{\epsilon}\right)\right)$, slightly higher than ours due to the presence of the factor $\log\left(\frac{\kappa}{\epsilon}\right)$. Moreover, it is allowed in HSGD to use constant step-size which is required to be shrinking in [11].

On this special problem, other results on general strongly convex problems can also be applied. As discussed in Section 3.2, HSGD is n times faster than FGD [8], and is superior to SAGA [12] and AVRG [12] when n dominates $\frac{1}{\epsilon}$. Shamir [11] showed that SVRG [5] under WoRS has IFO complexity $\mathcal{O}\left(\left(n + \kappa\log\left(\frac{1}{\epsilon}\right)\right)\log\left(\frac{1}{\epsilon}\right)\right)$ in ridge regression with the measurement $\mathbb{E}[f(\boldsymbol{x}) - f(\boldsymbol{x}^*)]$. Comparatively, such an IFO complexity is still higher than HSGD when sample size n is large and the desired accuracy is moderate.

Non-strongly convex case with strongly-convex $h(\cdot)$. When the loss $f(\boldsymbol{x})$ in (2) is non-strongly convex but the surrogate loss $h(\cdot)$ is strongly convex, we show an improved convergence rate in Theorem 5 than that in Theorem 3 for general cases. See proof of Theorem 5 in Appendix C.1.

Theorem 5. *Suppose $f_i(\boldsymbol{x}) = h(\boldsymbol{a}_i^\top \boldsymbol{x})$ is ℓ-smooth and $h(\cdot)$ is α-strongly convex. Let $\sigma(\boldsymbol{A})$ denote the smallest non-zero singular value of the matrix $\boldsymbol{A} = [\boldsymbol{a}_1^\top; \boldsymbol{a}_2^\top; \ldots, \boldsymbol{a}_n^\top]$ and $\mu = \alpha\sigma(\boldsymbol{A})$. If the learning rate $\eta_k = \frac{1}{\ell}$ and mini-batch size $s_k = \frac{\tau}{\zeta^k}$ with $\tau \geq \frac{24G^2}{\mu[f(\boldsymbol{x}^0) - f(\boldsymbol{x}^*)]}$ and $\zeta = 1 - \frac{\mu}{2\ell}$, we have*

$$\mathbb{E}\left[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)\right] \leq \left(1 - \frac{\mu}{2\ell}\right)^T \left(f(\boldsymbol{x}^0) - f(\boldsymbol{x}^*)\right),$$

where $\boldsymbol{x}^a$ denotes the output solution of Algorithm 1 and T is the number of iterations.

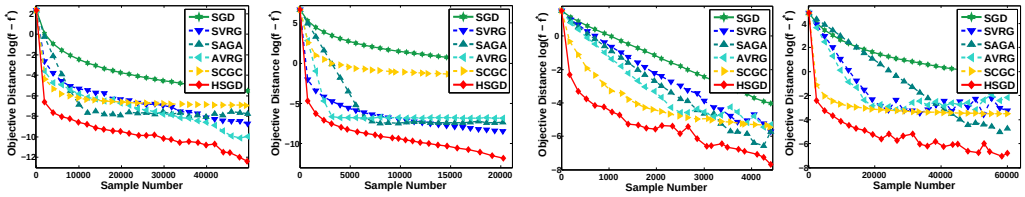
Theorem 5 shows if the function $h(\boldsymbol{a}_i^\top \boldsymbol{x})$ is strongly convex in terms of the linear prediction $\boldsymbol{a}_i^\top \boldsymbol{x}$, by exponentially sampling the data at each iteration, HSGD converges linearly even though $f(\boldsymbol{x})$ might be non-strongly convex. Based on Theorem 5, we further derive the IFO complexity of Algorithm 1 for such a special problem, as summarized in Corollary 4 with proof in Appendix C.2.

Corollary 4. *Suppose the assumptions in Theorem 5 hold. To achieve $\mathbb{E}[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)] \leq \epsilon$ for the special problem, the IFO complexity of the proposed algorithm is $\mathcal{O}\left(\frac{\ell G^2}{\mu^2 \epsilon}\right)$.*

It is interesting to compare Theorem 5 and Corollary 4 with those existing ones for SGD. Particularly, it was shown by Shamir [11] that $\mathbb{E}\left[f(\boldsymbol{x}^a) - f(\boldsymbol{x}^*)\right] \leq R_T/T + 2(12 + \sqrt{2}D)/\sqrt{n}$ for SGD, where R_T is the regret bound of SGD on problem (2), at the order of $\mathcal{O}(D\ell\sqrt{T})$. This gives a convergence rate of $\mathcal{O}(1/\sqrt{T})$ and IFO complexity of $\mathcal{O}(1/\epsilon^2)$. However, there exists an accuracy barrier $\mathcal{O}\left(1/\sqrt{n}\right)$ due to the statistical error term $2(12 + \sqrt{2}D)/\sqrt{n}$ which is the artifact brought by analyzing the regret. In sharp contrast, our result in Theorem 5 guarantees that HSGD converges to the global optimum of problem (2). More importantly, provided that $h(\cdot)$ is strongly convex, HSGD has superior IFO complexity of $\mathcal{O}\left(\frac{1}{\epsilon}\right)$ to the SGD complexity $\mathcal{O}\left(\frac{1}{\epsilon^2}\right)$ given in [11].

5 Experiments

We compare HSGD with several state-of-the-art algorithms, including SGD [2], SVRG [5], SAGA [14], AVRG [12] and SCGC [22], under WoRS for all. We consider two sets of learning tasks. The first contains two convex problems: ℓ_2-regularized logistic regression $\min_{\boldsymbol{x}} \frac{1}{n}\sum_{i=1}^n \left[\log(1 + \exp(-\boldsymbol{b}_i \boldsymbol{a}_i^\top \boldsymbol{x})) + \frac{\lambda}{2}\|\boldsymbol{x}\|_2^2\right]$ and k-classes softmax regression $\min_{\boldsymbol{x}} \frac{1}{n}\sum_{i=1}^n \sum_{j=1}^k \left[\frac{\lambda}{2k}\|\boldsymbol{x}_j\|_2^2 - \mathbf{1}\{\boldsymbol{b}_i = j\} \log \frac{\exp(\boldsymbol{a}_i^\top \boldsymbol{x}_j)}{\sum_{l=1}^k \exp(\boldsymbol{a}_i^\top \boldsymbol{x}_l)}\right]$, where $\boldsymbol{b}_i$ is the target output of $\boldsymbol{a}_i$. The other one is a non-convex problem of training multi-layer neural networks. We run simulations on 10 datasets (see Appendix D). Hyper-parameters of all the algorithms are tuned to best.



(a) Logistic regression on w08 (left) and rcv11 (right). (b) Softmax on satimage (left) and mnist (right).

Figure 2: Single-epoch processing: comparison of randomized algorithms for a single pass over data.

5.1 Convex problems

As the first set of problems are strongly convex, we follow Theorem 2 to exponentially expand the mini-batch size s_k in HSGD with $\tau = 1$. We run FGD until the gradient $\|\nabla f(\boldsymbol{x})\|_2 \leq 10^{-10}$. Then use the output as the optimal value f^* for sub-optimality estimation in Figure 1 (a), 2 and 3.

Single-epoch processing in well-conditioned problems. We first consider the case where the optimization problem is well-conditioned with strong regularization, such that good results can be obtained after only one epoch of data pass. Single-epoch learning is common in online learning. For two problems, we respectively set their regularization parameters to $\lambda = 0.01$ and $\lambda = 0.1$.

Figure 2 summarizes the numerical results. On the simulated well-conditioned tasks most algorithms achieve high accuracy after one epoch, while HSGD (WoRS) converges much faster. This confirms

Corollary 1 that HSGD is cheaper in IFO complexity ($\mathcal{O}\left(\frac{\kappa^2}{\epsilon}\right)$) than other considered variance-reduced algorithms ($\mathcal{O}\left(n\kappa^2 \log\left(\frac{1}{\epsilon}\right)\right)$) when the desired accuracy is moderately low and data size is large.

Multi-epoch processing in ill-conditioned problems. To solve more challenging problems, a method usually needs multiple cycles of data processing to reach high accuracy solution. Thus we develop a practical implementation of HSGD for multiple epochs processing. After visiting all data in one full pass, it continues to increase the mini-batch size, allowing possible with-replacement sampling, until $s_k > n$. After that, HSGD degenerates to standard FGD. But this does not happen in our testing cases, since we set the exponential rate small. To generate more challenging optimization tasks, we reset the regularization parameter λ in softmax regression to smaller value 0.001.

Figure 3 shows that HSGD under WoRS outperforms all compared algorithms. These observations accord with those in Figure 2, implying HSGD has sharper convergence behavior when the sample size n is large and the desired accuracy is moderate. The convergence curves of HSGD also confirm the effectiveness of our practical implementation: continuously decreasing the objective value.

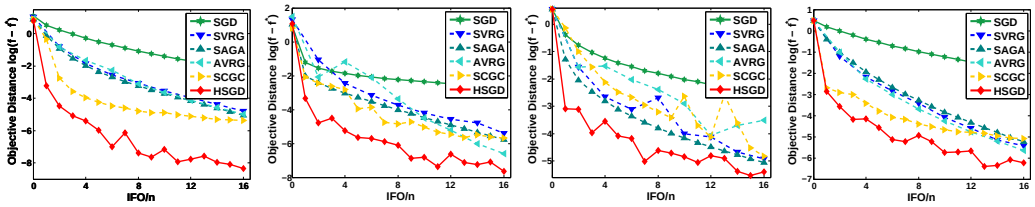
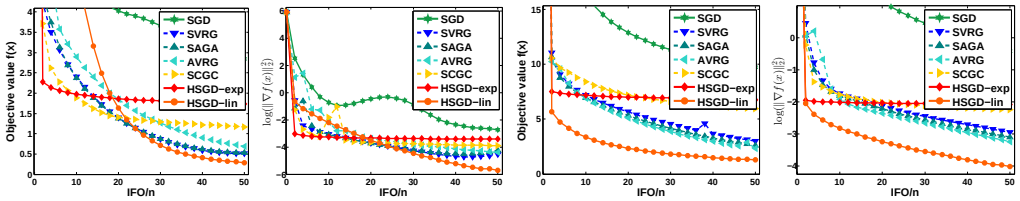


Figure 3: Multi-epoch processing: comparison of randomized algorithms for multiple passes over data (Softmax regression. From left to right: protein, satimage, sensorless and letter).

5.2 Non-convex problems

Here we evaluate HSGD for optimizing a three-layer feedforward neural network with a logistic loss on ijcnn1 and covtype and softmax loss on sensorless (see Figure 1 (b)). Both regularization parameters λ are set to 0.01. The network has an architecture of $d-30-c$, where d and c respectively denote the input and output dimension and 30 is the neuron number in the hidden layer. We test two versions of HSGD, namely HSGD-lin and HSGD-exp, respectively with linearly and exponentially increasing mini-batch size from $s_0 = 1$. We use the same initial points for all algorithms.

From Figure 4, HSGD-exp exhibits similar convergence behavior as above. It decreases the loss very fast. Comparatively, HSGD-lin gives more accurate solutions as it requires smaller batch size without harming convergence rate, consistent with Theorem 4 that advocates linearly increasing batch size. We note HSGD-lin behaves differently in Figure 4 (a) and (b). In Figure 4 (a), it converges relatively slowly at the beginning, while in Figure 4 (b) much faster, because of different data properties.



(a) ijcnn1 (b) covtype

Figure 4: Non-convex results: comparison of randomized algorithms on forward neural networks.

6 Conclusion

We analyzed the rate-of-convergence of HSGD under WoRS for strongly/non-strongly convex and non-convex problems. We proved under WoRS, HSGD with constant step-size can match FG descent in convergence rate, while maintaining comparable sample-size-independent IFO complexity to SGD. Compared to the variance-reduced SGD methods such as SVRG and SAGA, HSGD has lower cost in cases of large sample number and and moderately low required accuracy. Numerical results confirmed our theoretical results.

References

[1] M. A. Cauchy. Méthode générale pour la résolution des systèmes d'équations simultanées. *Comptesrendus des séances de l'Académie des sciences de Paris*, 25:536–538, 1847. 1

[2] H. Robbins and S. Monro. A stochastic approximation method. *The Annals of Mathematical Statistics*, 22(3):400–407, 1951. 1, 7

[3] D. P. Bertsekas. A new class of incremental gradient methods for least squares problems. *SIAM Journal on Optimization*, 7(4):913–926, 1997. 1, 2

[4] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *J. of Machine Learning Research*, 14(Feb):567–599, 2013. 1, 3

[5] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Proc. Conf. Neutral Information Processing Systems*, pages 315–323, 2013. 1, 3, 4, 6, 7

[6] M. P. Friedlander and M. Schmidt. Hybrid deterministic-stochastic methods for data fitting. *SIAM Journal on Scientific Computing*, 34(3):A1380–A1405, 2012. 1, 2, 4, 5, 6

[7] L. Bottou. Curiously fast convergence of some stochastic gradient descent algorithms. In *Proc. Symposium on Learning and Data Science, Paris*, 2009. 2

[8] M. Gürbüzbalaban, A. Ozdaglar, and P. Parrilo. Why random reshuffling beats stochastic gradient descent. *arXiv preprint arXiv:1510.08560*, 2015. 2, 3, 5, 6

[9] S. Reddi, A. Hefny, S. Sra, B. Poczos, and A. Smola. Stochastic variance reduction for nonconvex optimization. In *Proc. Int'l Conf. Machine Learning*, pages 314–323, 2016. 2, 6

[10] Z. Allen-Zhu and E. Hazan. Variance reduction for faster non-convex optimization. In *Proc. Int'l Conf. Machine Learning*, pages 699–707, 2016. 2, 6

[11] O. Shamir. Without-replacement sampling for stochastic gradient methods. In *Proc. Conf. Neutral Information Processing Systems*, pages 46–54, 2016. 2, 3, 4, 5, 6, 7

[12] B. Ying, K. Yuan, and A. H. Sayed. Convergence of variance-reduced stochastic learning under random reshuffling. *arXiv preprint arXiv:1708.01383*, 2017. 2, 3, 5, 6, 7

[13] B. Recht and C. Ré. Toward a noncommutative arithmetic-geometric mean inequality: conjectures, case-studies, and consequences. In *Conf. on Learning Theory*, pages 1–11, 2012. 2

[14] A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Proc. Conf. Neutral Information Processing Systems*, pages 1646–1654, 2014. 3, 4, 7

[15] N. L. Roux, M. Schmidt, and F. R. Bach. A stochastic gradient method with an exponential convergence rate for finite training sets. In *Proc. Conf. Neutral Information Processing Systems*, pages 2663–2671, 2012. 3

[16] A. Defazio, J. Domke, and T. S. Caetano. Finito: A faster, permutable incremental gradient method for big data problems. In *Proc. Int'l Conf. Machine Learning*, pages 1125–1133, 2014. 3

[17] H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Proc. Conf. Neutral Information Processing Systems*, pages 3384–3392, 2015. 3

[18] Z. Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. In *ACM SIGACT Symposium on Theory of Computing*, pages 1200–1205, 2017. 3

[19] Y. Zhang and L. Xiao. Stochastic primal-dual coordinate method for regularized empirical risk minimization. In *Proc. Int'l Conf. Machine Learning*, pages 353–361, 2015. 3

[20] Q. Lin, Z. Lu, and L. Xiao. An accelerated proximal coordinate gradient method. In *Proc. Conf. Neutral Information Processing Systems*, pages 3059–3067, 2014. 3

[21] J. M. Kohler and A. Lucchi. Sub-sampled cubic regularization for non-convex optimization. In *Proc. Int'l Conf. Machine Learning*, 2017. 4

[22] L. Lei and M. Jordan. Less than a single pass: Stochastically controlled stochastic gradient. In *Artificial Intelligence and Statistics*, pages 148–156, 2017. 4, 6, 7

New Insight into Hybrid Stochastic Gradient Descent: Beyond With-Replacement Sampling and Convexity (Supplementary File)

Pan Zhou* **Xiaotong Yuan†** **Jiashi Feng***
* Learning & Vision Lab, National University of Singapore, Singapore
† B-DAT Lab, Nanjing University of Information Science & Technology, Nanjing, China
pzhou@u.nus.edu xtyuan@nuist.edu.cn elefjia@nus.edu.sg

Abstract

This supplementary document contains the technical proofs of convergence results and some additional numerical results of the NIPS'18 submission entitled "New Insight into Hybrid Stochastic Gradient Descent: Beyond With-Replacement Sampling and Convexity". It is structured as follows. The proof of our key technical lemma, Lemma 1, is presented in Appendix A, followed by the proofs of main results in Appendices B and C for Section 3 and Section 4, respectively. Some detailed descriptions of data and algorithm along with more numerical results are provided in Appendix D.

A Proof of Lemma 1

Before proving Lemma 1 in the manuscript, we first give a useful lemma as stated in Lemma 2.

Lemma 2. *Assume that $\boldsymbol{a}_1, \ldots, \boldsymbol{a}_n$ denote the feature vectors of the n samples and let $\{\sigma_{(1)}, \ldots, \sigma_{(n)}\}$ be a permutation over $\{1, \ldots, n\}$ chosen uniformly at random. Let $\bar{\mathcal{S}}_k = \{\sigma_{(1)}, \ldots, \sigma_{(k)}\}$ and $\widetilde{\mathcal{S}}_k = \{\sigma_{(k+1)}, \ldots, \sigma_{(n)}\}$. For brevity, we further define*

$$\widetilde{\boldsymbol{z}}_k = \frac{1}{n-k} \sum_{i_k \in \widetilde{\mathcal{S}}_k} (\nabla f_{i_k}(\boldsymbol{x}) - \mu) \qquad \text{and} \qquad \widetilde{\boldsymbol{z}}_0 = \boldsymbol{0}$$
$$\bar{\boldsymbol{z}}_k = \frac{1}{k} \sum_{i_k \in \bar{\mathcal{S}}_k} (\nabla f_{i_k}(\boldsymbol{x}) - \mu) \qquad \text{and} \qquad \bar{\boldsymbol{z}}_0 = \boldsymbol{0},$$

where $\mu = \nabla f(\boldsymbol{x})$. Then we have

$$\mathbb{E}\left[\|\widetilde{\boldsymbol{z}}_k\|_2^2\right] \leq \frac{4G^2}{n-k}\left[1 - \frac{(n-k)^2 - k}{n(n-k)}\right], \qquad \mathbb{E}\left[\|\widetilde{\boldsymbol{z}}_k\|_2\right] \leq \frac{2G}{\sqrt{n-k}}\sqrt{1 - \frac{(n-k)^2 - k}{n(n-k)}},$$
$$\mathbb{E}\left[\|\bar{\boldsymbol{z}}_k\|_2^2\right] \leq \frac{4G^2}{k}\left[1 - \frac{k-1}{n}\right], \qquad \mathbb{E}\left[\|\bar{\boldsymbol{z}}_k\|_2\right] \leq \frac{2G}{\sqrt{k}}\sqrt{1 - \frac{k-1}{n}}.$$

Proof. Since $\mu = \frac{1}{n}\sum_{i_k=1}^{n} \nabla f_{i_k}(\boldsymbol{x})$, we can establish

$$\widetilde{\boldsymbol{z}}_k = \frac{1}{n-k}\left[-(n-k)\mu + n\mu - \sum_{i_k \in \bar{\mathcal{S}}_k} \nabla f_{i_k}(\boldsymbol{x})\right] = -\frac{1}{n-k}\sum_{i_k \in \bar{\mathcal{S}}_k} (\nabla f_{i_k}(\boldsymbol{x}) - \mu). \tag{3}$$

32nd Conference on Neural Information Processing Systems (NIPS 2018), Montréal, Canada.

On the other hand, we have

$$\begin{aligned}\mathbb{E}\left[\nabla f_{\sigma_{(k)}}(\boldsymbol{x})|\widetilde{\boldsymbol{z}}_{k-1},\ldots,\widetilde{\boldsymbol{z}}_0\right]=&\frac{1}{n-k+1}\sum_{i=k}^{n}\nabla_{\sigma_{(i)}}f(\boldsymbol{x})=\frac{1}{n-k+1}\left(n\mu-\sum_{i=1}^{k-1}\nabla f_{\sigma_{(i)}}(\boldsymbol{x})\right)\\=&\mu-\frac{1}{n-k+1}\sum_{i=1}^{k-1}\left(\nabla f_{\sigma_{(i)}}(\boldsymbol{x})-\mu\right).\end{aligned}\tag{4}$$

So we can obtain the following relation between $\mathbb{E}[\widetilde{\boldsymbol{z}}_k]$ and $\widetilde{\boldsymbol{z}}_{k-1}$:

$$\begin{aligned}&\mathbb{E}\left[\widetilde{\boldsymbol{z}}_k|\widetilde{\boldsymbol{z}}_{k-1},\ldots,\widetilde{\boldsymbol{z}}_0\right]\\=&-\frac{1}{n-k}\sum_{i=1}^{k-1}\left(\nabla f_{\sigma_{(i)}}(\boldsymbol{x})-\mu\right)-\frac{1}{n-k}\left(\mathbb{E}\left[\nabla f_{\sigma_{(k)}}(\boldsymbol{x})|\widetilde{\boldsymbol{z}}_{k-1},\ldots,\widetilde{\boldsymbol{z}}_0\right]-\mu\right)\\\overset{①}{=}&-\frac{1}{n-k}\sum_{i=1}^{k-1}\left(\nabla f_{\sigma_{(i)}}(\boldsymbol{x})-\mu\right)-\frac{1}{n-k}\left[\mu-\frac{1}{n-k+1}\sum_{i=1}^{k-1}\left(\nabla f_{\sigma_{(i)}}(\boldsymbol{x})-\mu\right)-\mu\right]\\=&-\frac{1}{n-k+1}\sum_{i=1}^{k-1}\left(\nabla f_{\sigma_{(i)}}(\boldsymbol{x})-\mu\right)\\\overset{②}{=}&\widetilde{\boldsymbol{z}}_{k-1},\end{aligned}$$

where ① holds since we plug Eqn. (4) and ② holds due to Eqn. (3). This means that the sequence $\widetilde{\boldsymbol{z}}_k$ is actually a martingale. Meanwhile we have

$$\widetilde{\boldsymbol{z}}_k=\frac{n-k+1}{n-k}\widetilde{\boldsymbol{z}}_{k-1}+\frac{1}{n-k}\left[\nabla f_{\sigma_{(k)}}(\boldsymbol{x})-\mu\right]=\widetilde{\boldsymbol{z}}_{k-1}+\frac{1}{n-k}\left[\nabla f_{\sigma_{(k)}}(\boldsymbol{x})-\mu+\widetilde{\boldsymbol{z}}_{k-1}\right].$$

Then we can further bound

$$\begin{aligned}\mathbb{E}\left[\|\widetilde{\boldsymbol{z}}_k\|^2|\widetilde{\boldsymbol{z}}_{k-1},\ldots,\widetilde{\boldsymbol{z}}_0\right]=&\mathbb{E}[\|\widetilde{\boldsymbol{z}}_k-\widetilde{\boldsymbol{z}}_{k-1}+\widetilde{\boldsymbol{z}}_{k-1}\|^2]\\=&\mathbb{E}[\|\widetilde{\boldsymbol{z}}_k-\widetilde{\boldsymbol{z}}_{k-1}\|^2+2\langle\widetilde{\boldsymbol{z}}_k-\widetilde{\boldsymbol{z}}_{k-1},\widetilde{\boldsymbol{z}}_{k-1}\rangle+\|\widetilde{\boldsymbol{z}}_{k-1}\|^2]\\=&\mathbb{E}\left[\frac{1}{(n-k)^2}\|\nabla f_{\sigma_{(k)}}(\boldsymbol{x})-\mu+\widetilde{\boldsymbol{z}}_{k-1}\|^2+\|\widetilde{\boldsymbol{z}}_{k-1}\|^2\right]\\\overset{①}{\leq}&\frac{4G^2}{(n-k)^2}+\|\widetilde{\boldsymbol{z}}_{k-1}\|^2,\end{aligned}\tag{5}$$

where ① holds since we have $\|\nabla f_{\sigma_{(k)}}(\boldsymbol{x})-\mu+\widetilde{\boldsymbol{z}}_{k-1}\|_2\leq 2(\|\nabla f_{\sigma_{(k)}}(\boldsymbol{x})-\mu\|^2+\|\widetilde{\boldsymbol{z}}_{k-1}\|^2)\leq 4G^2$, where $G=\max_i\|\nabla f_i(\boldsymbol{x})-\mu\|_2$. Conditioned on all the random process and sum Eqn. (5) together, we obtain

$$\begin{aligned}\mathbb{E}\left[\|\widetilde{\boldsymbol{z}}_k\|^2\right]\leq&4G^2\sum_{i=1}^{k}\frac{1}{(n-i)^2}+\mathbb{E}\|\widetilde{\boldsymbol{z}}_0\|^2\leq 4G^2\sum_{i=1}^{k}\frac{1}{(n-i)^2}\overset{①}{\leq}\frac{4G^2}{(n-k)^2}+\frac{4(k-1)G^2}{n(n-k)}\\=&\frac{4G^2}{n-k}\left(1-\frac{(n-k)^2-k}{n(n-k)}\right),\end{aligned}$$

where ① holds since for $1\leq k\leq n$, we have $\sum_{i=k+1}^{n}\frac{1}{i^2}\leq\frac{n-k}{k(n+1)}$. Since the function $\sqrt{\cdot}$ is concave function, we can use Jensen's inequality to obtain

$$\mathbb{E}\left[\|\widetilde{\boldsymbol{z}}_k\|\right]\leq\sqrt{\mathbb{E}\left[\|\widetilde{\boldsymbol{z}}_k\|^2\right]}\leq\frac{2G}{\sqrt{n-k}}\sqrt{1-\frac{(n-k)^2-k}{n(n-k)}}.$$

In a similar way, we can prove that $\hat{\boldsymbol{z}}_k=\frac{k}{n-k}\bar{\boldsymbol{z}}_k$ is a martingale sequence and

$$\hat{\boldsymbol{z}}_k=\hat{\boldsymbol{z}}_{k-1}+\frac{1}{n-k}\left[\nabla f_{\sigma_{(k)}}(\boldsymbol{x})-\mu+\hat{\boldsymbol{z}}_{k-1}\right].$$

Therefore, we can bound

$$\mathbb{E}\left[\|\hat{\boldsymbol{z}}_k\|^2\right] \leq \frac{4G^2}{(n-k)^2} + \mathbb{E}\|\hat{\boldsymbol{z}}_{k-1}\|^2 \leq 4G^2 \sum_{i=1}^{k} \frac{1}{(n-i)^2}.$$

So by using $\hat{\boldsymbol{z}}_k = \frac{k}{n-k}\bar{\boldsymbol{z}}_k$, it follows

$$\begin{aligned}\mathbb{E}\left[\|\bar{\boldsymbol{z}}_k\|^2\right] \leq & \frac{4G^2}{k^2} \sum_{i=1}^{k} \frac{(n-k)^2}{(n-i)^2} \leq \frac{4G^2}{k^2}\left(1 + \sum_{i=n-k+1}^{n-1} \frac{(n-k)^2}{i^2}\right) \overset{①}{\leq} \frac{4G^2}{k^2}\left(1 + (n-k)^2 \frac{k-1}{n(n-k)}\right) \\ \leq & \frac{4G^2}{k^2}\left(1 + k - 1 - \frac{k(k-1)}{n}\right) \leq \frac{4G^2}{k}\left(1 - \frac{k-1}{n}\right).\end{aligned}$$

Therefore, by Jensen's inequality we have

$$\mathbb{E}\left[\|\bar{\boldsymbol{z}}_k\|\right] \leq \sqrt{\mathbb{E}\left[\|\bar{\boldsymbol{z}}_k\|^2\right]} \leq \frac{2G}{\sqrt{k}}\sqrt{1 - \frac{k-1}{n}}.$$

The proof is completed. □

Now we use Lemma 2 to prove the following lemma.

Lemma 3. *Let $\boldsymbol{g}^k$ be the gradient estimate in Algorithm 1 by WoRS. We have $\mathbb{E}\left[\|\boldsymbol{g}^k - \nabla f(\boldsymbol{x}^k)\|_2^2\right] \leq c_k$, where*

$$c_k = \frac{8G^2}{n-b_k}\left[1 - \frac{(n-b_k)^2 - b_k}{n(n-b_k)}\right] + \frac{8G^2}{s_k}\left[1 - \frac{s_k - 1}{n-b_k}\right],$$

and $b_k = \sum_{i=0}^{k-1} s_i$.

Proof. Firstly, we introduce the following sequence of random variables $\boldsymbol{z}_k$:

$$\boldsymbol{z}_k = \frac{1}{s'_k} \sum_{i_k \in \mathcal{S}'_k} \left(\nabla f_{i_k}(\boldsymbol{x}^k) - \nabla f(\boldsymbol{x}^k)\right) \quad \text{and} \quad \boldsymbol{z}_0 = \boldsymbol{0},$$

where $\mathcal{S}'_k = \mathcal{S} - \bigcup_{i=0}^{k-1} \mathcal{S}_i$ and $s'_k = n - \sum_{i=0}^{k-1} s_i$ respectively denote the indexes and number of remaining samples after the $(k-1)$-th without-replacement sampling in which $\mathcal{S} = \{1, 2, \cdots, n\}$. So actually $\boldsymbol{z}_k$ is actually equivalent to $\widetilde{\boldsymbol{z}}_{b_k}$ where $b_k = \sum_{i=1}^{k-1} s_i$ due to the definition of $\widetilde{\boldsymbol{z}}_{b_k}$ in Lemma 2:

$$\widetilde{\boldsymbol{z}}_{b_k} = \frac{1}{n-b_k} \sum_{i_k \in \widetilde{\mathcal{S}}_k} \left(\nabla f_{i_k}(\boldsymbol{x}) - \nabla f(\boldsymbol{x})\right) \qquad \text{and} \qquad \widetilde{\boldsymbol{z}}_0 = \boldsymbol{0},$$

where $\widetilde{\mathcal{S}}_k = \mathcal{S} - \bigcup_{i=1}^{k-1} \mathcal{S}_i$. This is because that both $\boldsymbol{z}_k$ and $\widetilde{\boldsymbol{z}}_{b_k}$ actually measure the gradient variance of the data points indexed by $\mathcal{S}'_k = \mathcal{S} - \bigcup_{i=0}^{k-1} \mathcal{S}_i$ which is sampled by WoRS. The only difference is that in the sequence $\boldsymbol{z}_k$, we sample the data $\mathcal{S}'_k = \mathcal{S} - \bigcup_{i=0}^{k-1} \mathcal{S}_i$ by removing mini-batch $\mathcal{S}_k$ at the k-th iteration, while in $\widetilde{\boldsymbol{z}}_{b_k}$ in Lemma 2, we sample the data $\widetilde{\mathcal{S}}_k = \mathcal{S} - \bigcup_{i=1}^{k-1} \mathcal{S}_i$ by removing one data in one sampling operation under WoRS. Since both sequences use without-replacement sampling, they have the same gradient variance when the sampled data have the same number. So we can use the bound of $\widetilde{\boldsymbol{z}}_{b_k}$ to bound $\boldsymbol{z}_k$. Thus, by Lemma 2, we can obtain that $\widetilde{\boldsymbol{z}}_{b_k}$ is a martingale (namely, $\mathbb{E}[\widetilde{\boldsymbol{z}}_k | \widetilde{\boldsymbol{z}}_{k-1}, \ldots, \widetilde{\boldsymbol{z}}_0] = \widetilde{\boldsymbol{z}}_{k-1}$) and its norm can be bounded as

$$\begin{aligned}\mathbb{E}\left[\|\boldsymbol{z}_k\|_2 | \boldsymbol{z}_{k-1}, \ldots, \boldsymbol{z}_0\right] &= \mathbb{E}[\|\widetilde{\boldsymbol{z}}_k\|_2 | \widetilde{\boldsymbol{z}}_{k-1}, \ldots, \widetilde{\boldsymbol{z}}_0] \leq \frac{2G}{\sqrt{n-b_k}}\sqrt{1 - \frac{(n-b_k)^2 - b_k}{n(n-b_k)}}, \\ \mathbb{E}\left[\|\boldsymbol{z}_k\|_2^2 | \boldsymbol{z}_{k-1}, \ldots, \boldsymbol{z}_0\right] &= \mathbb{E}[\|\widetilde{\boldsymbol{z}}_k\|_2^2 | \widetilde{\boldsymbol{z}}_{k-1}, \ldots, \widetilde{\boldsymbol{z}}_0] \leq \frac{4G^2}{n-b_k}\left[1 - \frac{(n-b_k)^2 - b_k}{n(n-b_k)}\right].\end{aligned} \tag{6}$$

On the other hand, we define a sequence of $\bar{\boldsymbol{z}}_i$ for the process of without-replacement sampling a subset $\widehat{\mathcal{S}}_i$ of size $\widehat{s}_i$ from $\mathcal{S}'_k$ of size s'_k:

$$\bar{\boldsymbol{z}}_i = \frac{1}{\widehat{s}_i} \sum_{i_k \in \widehat{\mathcal{S}}_i} \nabla f_{i_k}(\boldsymbol{x}^k) - \bar{\boldsymbol{\mu}}_k \quad \text{and} \quad \bar{\boldsymbol{z}}_0 = \mathbf{0},$$

where $\widehat{s}_i$ actually equals to s_k. Then we can use the result in Lemma 2 on $\bar{\boldsymbol{z}}_i$ to bound its norm:

$$\mathbb{E}\left[\|\bar{\boldsymbol{z}}_i\|_2 | \bar{\boldsymbol{z}}_{i-1}, \ldots, \bar{\boldsymbol{z}}_0\right] \leq \frac{2G}{\sqrt{\widehat{s}_i}} \sqrt{1 - \frac{\widehat{s}_i - 1}{s'_k}} \quad \text{and} \quad \mathbb{E}\left[\|\bar{\boldsymbol{z}}_i\|_2^2 | \bar{\boldsymbol{z}}_{i-1}, \ldots, \bar{\boldsymbol{z}}_0\right] \leq \frac{4G^2}{\widehat{s}_i} \left[1 - \frac{\widehat{s}_i - 1}{s'_k}\right]. \tag{7}$$

Finally, we combine these two bounds together to obtain our final results. We can formulate the k-th without-replacement sampling as a random process, including two phases. In the first phase, we view the remaining samples after the first $k-1$ without-replacement sampling as a without-replacement sampling. In this case, we obtain s'_k samples indexed by $\mathcal{S}'_k = \mathcal{S} - \bigcup_{i=1}^{k-1} \mathcal{S}_i$. This sampling step corresponds to the martingale $\boldsymbol{z}_i$. Then, in the second phase, we sample s_k data from the remaining s'_k samples indexed by $\mathcal{S}'_k$, which corresponds to the martingale sequence $\bar{\boldsymbol{z}}_i$. Define $\bar{\mu} = \frac{1}{s'_k} \sum_{i_k \in \mathcal{S}'_k} \nabla f_{i_k}(\boldsymbol{x}^k)$. Then we can bound

$$\begin{aligned}
\mathbb{E}[\|\boldsymbol{g}^k - \nabla f(\boldsymbol{x}^k)\|_2^2] \leq & 2\mathbb{E}[\|\bar{\mu} - \nabla f(\boldsymbol{x}^k)\|_2^2 + \|\boldsymbol{g}^k - \bar{\mu}\|_2^2] \\
= & 2\mathbb{E}[\|\boldsymbol{z}_k\|_2^2 | \boldsymbol{z}_{k-1}, \ldots, \boldsymbol{z}_0] + \mathbb{E}[\|\bar{\boldsymbol{z}}_{s_k}\|_2^2 | \bar{\boldsymbol{z}}_{s_k - 1}, \ldots, \bar{\boldsymbol{z}}_0; \boldsymbol{z}_{k-1}, \ldots, \boldsymbol{z}_0] \\
\leq & \frac{8G^2}{n - b_k} \left[1 - \frac{(n - b_k)^2 - b_k}{n(n - b_k)}\right] + \frac{8G^2}{s_k} \left[1 - \frac{s_k - 1}{n - b_k}\right].
\end{aligned}$$

This completes the proof. □

We are now in the position to prove Lemma 1.

Proof of Lemma 1. Since we have $n \geq \sum_{i=0}^{k} s_i$ and s_k is monotone increasing, it follows $n - \sum_{i=0}^{k-1} s_i \geq s_k \geq s_{k-1}$. So in Lemma 3, we have

$$1 - \frac{(n - \sum_{i=0}^{k-1} s_i)^2 - \sum_{i=0}^{k-1} s_i}{n(n - \sum_{i=0}^{k-1} s_i)} \leq 1 + \frac{\sum_{i=0}^{k-1} s_i}{n(n - \sum_{i=0}^{k-1} s_i)} \leq 1 + \frac{1}{n - \sum_{i=0}^{k-1} s_i} \leq 2. \tag{8}$$

Therefore, plugging this into Lemma 3, we can further obtain

$$\mathbb{E}\|\boldsymbol{g}^k - \mu\|_2^2 \leq \frac{24G^2}{s_k}.$$

This proves the desired bound in the lemma. □

B Proofs of Results in Section 3

For brevity, here we use f^k and f^* to denote $f(\boldsymbol{x}^k)$ and $f(\boldsymbol{x}^*)$, respectively.

B.1 Proof of Theorem 1

Before we prove Theorem 1, we first give a useful corollary derived from Lemma 1.

Corollary 1. *Let the sub-sampled gradient $\boldsymbol{g}_k$ be defined in Algorithm 1 without replacement and $s_{k+1} \geq s_k (k \geq 0)$. Then we have*

$$\mathbb{E}[\|\boldsymbol{g}^k - \nabla f(\boldsymbol{x}^k)\|_2] \leq \frac{4G}{\sqrt{s_k}}. \tag{9}$$

where $G = \max_i \|\nabla f_i - \mu\|_2$.

Proof. From Eqn. (8) in proof of Lemma 1 we have

$$1-\frac{(n-\sum_{i=0}^{k-1}s_i)^2-\sum_{i=0}^{k-1}s_i}{n(n-\sum_{i=0}^{k-1}s_i)}\leq 2.$$

Therefore, by using Eqn. (6) and (7)we can further obtain

$$\mathbb{E}\|\boldsymbol{g}^k-\nabla f(\boldsymbol{x}^k)\|_2\leq\frac{(2\sqrt{2}+1)G}{\sqrt{s_k}}\leq\frac{4G}{\sqrt{s_k}}.$$

The proof is completed. □

Now we begin to prove Theorem 1. For brevity, here we use f^k and f^* to denote $f(\boldsymbol{x}^k)$ and $f(\boldsymbol{x}^*)$, respectively.

Proof. Now we begin to prove the linear convergence of HSGD. We firstly give an useful inequality:

$$\begin{aligned}\mathbb{E}\langle\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k,\nabla f^k-\boldsymbol{g}^k\rangle=&\mathbb{E}\langle\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k(\nabla f^k-\nabla f^*),\nabla f^k-\boldsymbol{g}^k\rangle\\=&\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k(\nabla f^k-\nabla f^*)\|\cdot\|\nabla f^k-\boldsymbol{g}^k\|\\=&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*\|+\eta_k\|\nabla f^k-\nabla f^*\|\right)\|\nabla f^k-\boldsymbol{g}^k\|\\\overset{①}{\leq}&\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*\|+\eta_k\ell\|\boldsymbol{x}^k-\boldsymbol{x}^*\|\right)\frac{4G}{\sqrt{s_k}}\\\leq&\frac{4G}{\sqrt{s_k}}\left(1+\eta_k\ell\right)\|\boldsymbol{x}^k-\boldsymbol{x}^*\|\end{aligned}\tag{10}$$

where ① holds since $f(\boldsymbol{x})$ is ℓ-smooth and we can bound $\mathbb{E}\|\nabla f^k-\boldsymbol{g}^k\|$ by using Corollary 1.

Then we give the recurrence relation between $\mathbb{E}\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^*\|^2$ and $\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2$ as follows:

$$\begin{aligned}&\mathbb{E}\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^*\|^2\\=&\mathbb{E}\|\Phi_{\boldsymbol{\mathcal{X}}}(\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\boldsymbol{g}^k)\|^2\\=&\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k-\eta_k(\nabla f^k-\boldsymbol{g}^k)\|^2\\=&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k\|^2+\eta_k^2\|\nabla f^k-\boldsymbol{g}^k\|^2\ -2\eta_k\langle\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k,\nabla f^k-\boldsymbol{g}^k\rangle\right)\\=&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2-2\eta_k\langle\boldsymbol{x}^k-\boldsymbol{x}^*,\nabla f^k-\nabla f^*\rangle\ +\eta_k^2\|\nabla f^k-\nabla f^*\|^2\right)+\eta_k^2\mathbb{E}\|\nabla f^k-\boldsymbol{g}^k\|^2\\&-2\eta_k\mathbb{E}\langle\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k,\nabla f^k-\boldsymbol{g}^k\rangle\\\overset{①}{\leq}&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2-2\eta_k\rho\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2+\eta_k^2\ell^2\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2\right)+\eta_k^2\mathbb{E}\|\nabla f^k-\boldsymbol{g}^k\|^2\\&-2\eta_k\mathbb{E}\langle\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k,\nabla f^k-\boldsymbol{g}^k\rangle\\\overset{②}{\leq}&(1-2\eta_k\rho+\eta_k^2\ell^2)\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2+\eta_k^2\frac{24G^2}{s_k}+\frac{8G\eta_k}{\sqrt{s_k}}\left(1+\eta_k\ell\right)\|\boldsymbol{x}^k-\boldsymbol{x}^*\|,\end{aligned}$$

where ① holds because we use the ℓ-smooth property of $f(\boldsymbol{x})$, and for a strong convex function $f(\boldsymbol{x})$, we have the monotonicity of ∇f:

$$\langle\boldsymbol{x}^k-\boldsymbol{x}^*,\nabla f^k-\nabla f^*\rangle\geq\rho\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2.$$

② holds due to Lemma 1 and Eqn. (10).

Here we set $\eta_k=\frac{\rho}{\ell^2}$ and $s_k=\tau(1/\zeta)^k$, where $\zeta\in(0,1)$. Then consider $\ell\geq\rho$, it yields

$$\mathbb{E}\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^*\|^2\leq\left(1-\frac{\rho^2}{\ell^2}\right)\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2+\frac{8\rho G}{\ell^2\sqrt{\tau}}\left(1+\frac{\rho}{\ell}\right)\zeta^{k/2}\|\boldsymbol{x}^k-\boldsymbol{x}^*\|+\frac{24\rho^2G^2}{\tau\ell^4}\zeta^k.$$

For brevity, let $\alpha=1-\frac{\rho^2}{\ell^2}$, $\beta=\frac{8\rho G}{\ell^2\sqrt{\tau}}\left(1+\frac{\rho}{\ell}\right)$ and $\gamma=\frac{24\rho^2G^2}{\tau\ell^4}$. Thus, we have

$$\mathbb{E}\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^*\|^2=\alpha\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2+\beta\zeta^{k/2}\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*\|+\gamma\zeta^k.$$

We further assume that τ is large enough such that

$$\gamma = \frac{24\rho^2 G^2}{\tau \ell^4} \leq \delta \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2, \tag{11}$$

where δ is a positive constant and will be discussed later. Now we use mathematical induction to prove

$$\mathbb{E}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 \leq \theta^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2 \tag{12}$$

where $\theta < 1$ is a constant and will be given below.

Obviously, when $k = 0$, Eqn. (12) holds. Now assume that for all $t \leq k$, Eqn. (12) holds. Then for $t = k + 1$, we have

$$\begin{aligned}
\mathbb{E}\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^*\|^2 \leq & \alpha \mathbb{E}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 + \beta \zeta^{k/2} \mathbb{E}\|\boldsymbol{x}^k - \boldsymbol{x}^*\| + \gamma \zeta^k \\
\leq & \alpha \theta^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2 + \beta \zeta^{k/2} \theta^{k/2} \mathbb{E}\|\boldsymbol{x}^0 - \boldsymbol{x}^*\| + \gamma \zeta^k \\
\overset{①}{\leq} & \left(\alpha + \frac{\beta}{\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|} + \delta \right) \theta^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2 \\
\overset{②}{\leq} & \theta^{k+1} \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2,
\end{aligned}$$

where ① and ② hold since we let

$$\theta \geq \max(\zeta, \alpha + \frac{\beta}{\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|} + \delta). \tag{13}$$

This means that if Eqn. (13), then Eqn. (12) always holds. So the conclusion holds.

Now we discuss the values of θ, ζ and τ such that Eqn. (12) is satisfied. We just set $\delta = \frac{24\rho^2 G^2}{\tau \ell^4 \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2}$, $\tau = \max\left(\frac{324 G^2}{\rho^2 \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2}, \frac{432 G^2}{\ell^2 \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2} \right)$ and $\theta = \zeta = 1 - \frac{\rho^2}{18\ell^2}$, which gives

$$\begin{aligned}
\theta \geq & \alpha + \frac{\beta}{\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|} + \delta \\
= & 1 - \frac{\rho^2}{\ell^2} + \frac{8\rho G}{\ell^2 \sqrt{\tau}} \left(1 + \frac{\rho}{\ell} \right) \frac{1}{\|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2} + \frac{24\rho^2 G^2}{\tau \ell^4 \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2} \\
\geq & 1 - \frac{\rho^2}{\ell^2} + \frac{8\rho^2}{9\ell^2} + \frac{\rho^2}{18\ell^2} = 1 - \frac{\rho^2}{18\ell^2}.
\end{aligned}$$

In this case, all the conditions, including Eqn. (11) and (13). So we can see that the values of θ, ζ and τ are proper. Therefore, we have

$$\mathbb{E}\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 \leq \left(1 - \frac{\rho^2}{18\ell^2} \right)^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2.$$

The proof if completed. □

B.2 Proof of Corollary 1

Proof. To achieve ϵ-accurate solution, *i.e.*

$$\|\boldsymbol{x}^k - \boldsymbol{x}^*\|^2 \leq \theta^k \|\boldsymbol{x}^0 - \boldsymbol{x}^*\|^2 \leq \epsilon,$$

where $\theta = 1 - \frac{\rho^2}{18\ell^2}$, we have

$$k^* \geq \log_{1/\theta} \left(\frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\epsilon} \right).$$

Therefore, the IFO complexity is

$$\begin{aligned}
\tau \left[1 + \frac{1}{\zeta} + \cdots + \frac{1}{\zeta^{k^*-1}} \right] = & \tau \frac{(1/\zeta)^{\log_{1/\theta}\left(\frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\epsilon} \right)} - 1}{1/\zeta - 1} = \frac{\tau}{1/\zeta - 1} \left[\frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\epsilon} - 1 \right] \\
\leq & \frac{\tau}{1/\zeta - 1} \left[\frac{\|\boldsymbol{x}_0 - \boldsymbol{x}^*\|^2}{\epsilon} \right] = \mathcal{O}\left(\frac{\ell^2 G^2}{\rho^2 \epsilon} \right).
\end{aligned}$$

This means that we have the IFO complexity $\mathcal{O}\left(\frac{\ell^2 G^2}{\rho^2 \epsilon} \right)$. The proof is completed. □

B.3 Proof of Theorem 2

Proof. Now we begin to prove the linear convergence of WoRS-based HSGD. Firstly, by smooth property, we have

$$\begin{aligned}
\mathbb{E}f^{k+1} \leq & \mathbb{E}\left[f^k + \langle \nabla f^k, \boldsymbol{x}^{k+1} - \boldsymbol{x}^k\rangle + \frac{\ell}{2}\|\boldsymbol{x}^{k+1} - \boldsymbol{x}^k\|^2\right] \\
\overset{①}{=} & \mathbb{E}\left[f^k - \eta_k\langle \nabla f^k, \boldsymbol{g}^k - \nabla f^k + \nabla f^k\rangle + \frac{\ell}{2}\|\Phi_{\boldsymbol{\mathcal{X}}}\left(\boldsymbol{x}^k - \eta_k \boldsymbol{g}^k\right) - \boldsymbol{x}^k\|^2\right] \\
= & \mathbb{E}\left[f^k - \eta_k\langle \nabla f^k, \boldsymbol{g}^k - \nabla f^k + \nabla f^k\rangle + \frac{\ell}{2}\|\Phi_{\boldsymbol{\mathcal{X}}}\left(\eta_k \boldsymbol{g}^k\right)\|^2\right] \\
\leq & \mathbb{E}\left[f^k - \eta_k\langle \nabla f^k, \boldsymbol{g}^k - \nabla f^k + \nabla f^k\rangle + \frac{\ell\eta_k^2}{2}\|\boldsymbol{g}^k - \nabla f^k + \nabla f^k\|^2\right] \\
= & \mathbb{E}\left[f^k - \eta_k(1-\eta_k\ell)\langle \nabla f^k, \boldsymbol{g}^k - \nabla f^k\rangle + \frac{\ell\eta_k^2}{2}\|\boldsymbol{g}^k - \nabla f^k\|^2 - \eta_k\left(1 - \frac{\ell\eta_k}{2}\right)\|\nabla f^k\|^2\right],
\end{aligned}$$

where ① holds due to $\boldsymbol{x}^k \in \boldsymbol{\mathcal{X}}$. Here we set $\eta_k = \frac{1}{\ell}$ and plug it into the above inequality:

$$\mathbb{E}f^{k+1} \leq \mathbb{E}\left[f^k + \frac{1}{2\ell}\|\boldsymbol{g}^k - \nabla f^k\|^2 - \frac{1}{2\ell}\|\nabla f^k\|^2\right] \overset{①}{\leq} \mathbb{E}\left[f^k + \frac{12G^2}{\ell s_k} - \frac{1}{2\ell}\|\nabla f^k\|^2\right], \tag{14}$$

where ① holds since we can bound $\mathbb{E}\|\nabla f^k - \boldsymbol{g}^k\|_2^2$ by using Lemma 1.

On the other hand, $f(\boldsymbol{x})$ is a strongly convex function. Namely, we have

$$f(\boldsymbol{y}) \geq f(\boldsymbol{x}) + \nabla f(\boldsymbol{x})^T(\boldsymbol{y} - \boldsymbol{x}) + \frac{\rho}{2}\|\boldsymbol{y} - \boldsymbol{x}\|^2.$$

Then by minimizing $\boldsymbol{y}$ on both sides, it yields

$$\frac{1}{2\rho}\|\nabla f(\boldsymbol{x})\|^2 \geq f(\boldsymbol{x}) - f(\boldsymbol{x}^*). \tag{15}$$

We plug Eqn. (15) into Eqn. (14) and obtain

$$\mathbb{E}(f^{k+1} - f^*) \leq \left(1 - \frac{\rho}{\ell}\right)(f^k - f^*) + \frac{12G^2}{\ell s_k}.$$

Here we set $s_k = \tau(1/\zeta)^k$, where $\zeta \in (0,1)$. For brevity, let $\alpha = 1 - \frac{\rho}{\ell}$ and $\gamma = \frac{12G^2}{\tau\ell}$. It yields

$$\mathbb{E}(f^{k+1} - f^*) \leq \alpha(f^k - f^*) + \gamma\zeta^k.$$

We further assume that τ is large enough such that

$$\gamma = \frac{12G^2}{\tau\ell} \leq \delta(f^0 - f^*), \tag{16}$$

where δ is a positive constant and will be discussed later. Now we use mathematical induction to prove

$$f^k - f^* \leq \theta^k(f^0 - f^*),\ (\forall k), \tag{17}$$

where $\theta < 1$ is a constant and will be given below.

Obviously, when $k = 0$, Eqn. (17) holds. Now assume that for all $t \leq k$, Eqn. (17) holds. Then for $t = k+1$, we have

$$\begin{aligned}
\mathbb{E}(f^{k+1} - f^*) \leq & \alpha\mathbb{E}(f^k - f^*) + \gamma\zeta^k \leq \alpha\theta^k(f^0 - f^*) + \gamma\zeta^k \\
\overset{①}{\leq} & (\alpha + \delta)\,\theta^k(f^0 - f^*) \overset{②}{\leq} \theta^{k+1}(f^0 - f^*),
\end{aligned}$$

where ① and ② hold since we let

$$\theta \geq \max(\zeta, \alpha + \delta). \tag{18}$$

This means that if Eqn. (18) holds, then Eqn. (17) always holds. So the conclusion holds.

Now we discuss the values of θ, ζ and τ such that Eqn. (18) is satisfied. We just set $\delta = \frac{12G^2}{\tau\ell(f^0-f^*)}$, $\tau \geq \frac{6G^2}{\rho(f^0-f^*)}$ and $\theta = \zeta = 1 - \frac{\rho}{2\ell}$, giving

$$\theta \geq \alpha + \delta \geq 1 - \frac{\rho}{\ell} + \frac{\rho}{2\ell} = 1 - \frac{\rho}{2\ell}.$$

In this case, all the conditions hold, including Eqn. (16) and (18). So we can see that the values of θ, ζ and τ are proper. Therefore, we have

$$\mathbb{E}(f^k - f^*) \leq \left(1 - \frac{\rho}{2\ell}\right)^k (f^0 - f^*).$$

Then we derive the IFO complexity. To achieve ϵ-accurate solution, *i.e.*

$$\mathbb{E}(f^k - f^*) \leq \theta^k (f^0 - f^*) \leq \epsilon,$$

where $\theta = 1 - \frac{\rho}{2\ell}$, we have

$$k^* \geq \log_{1/\theta}\left(\frac{f^0 - f^*}{\epsilon}\right).$$

Therefore, the IFO complexity is

$$\begin{aligned}\tau\left[1 + \frac{1}{\zeta} + \cdots + \frac{1}{\zeta^{k^*-1}}\right] =& \tau\frac{(1/\zeta)^{\log_{1/\theta}\left(\frac{f^0-f^*}{\epsilon}\right)} - 1}{1/\zeta - 1} = \frac{\tau}{1/\zeta - 1}\left[\frac{f^0 - f^*}{\epsilon} - 1\right] \\ \leq& \frac{\tau}{1/\zeta - 1}\left[\frac{f^0 - f^*}{\epsilon}\right] \leq \mathcal{O}\left(\frac{\kappa G^2}{\epsilon}\right),\end{aligned}$$

where $\kappa = \ell/\rho$. This means that we have the IFO complexity $\mathcal{O}\left(\frac{\kappa G^2}{\epsilon}\right)$. The proof is completed.

The proof is completed.

□

B.4 Proof of Theorem 3

For brevity, here we use f^k and f^* to denote $f(\boldsymbol{x}^k)$ and $f(\boldsymbol{x}^*)$, respectively.

Proof. From Eqn. (10) in Appendix B.1, we have

$$\mathbb{E}\langle \boldsymbol{x}^k - \boldsymbol{x}^* - \eta_k \nabla f^k, \nabla f^k - \boldsymbol{g}^k\rangle \leq \frac{4G}{\sqrt{s_k}}(1 + \eta_k\ell)\left\|\boldsymbol{x}^k - \boldsymbol{x}^*\right\|.$$

For arbitrary $\boldsymbol{x}_1 \in \boldsymbol{\mathcal{X}}$ and $\boldsymbol{x}_2 \in \boldsymbol{\mathcal{X}}$ that satisfy $\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2 \leq D$, we can bound $\mathbb{E}\langle \boldsymbol{x}^k - \boldsymbol{x}^* - \eta_k \nabla f^k, \nabla f^k - \boldsymbol{g}^k\rangle$ as follows:

$$\mathbb{E}\langle \boldsymbol{x}^k - \boldsymbol{x}^* - \eta_k \nabla f^k, \nabla f^k - \boldsymbol{g}^k\rangle \overset{①}{\leq} \frac{4G}{\sqrt{s_k}}(1 + \eta_k\ell)\left\|\boldsymbol{x}^k - \boldsymbol{x}^*\right\| \leq \frac{4(1 + \eta_k\ell)GD}{\sqrt{s_k}}. \tag{19}$$

Then we utilize Eqn. (19) to further give the relationship between $\mathbb{E}\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^*\|^2$ and $\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2$:

$$
\begin{aligned}
&\mathbb{E}\|\boldsymbol{x}^{k+1}-\boldsymbol{x}^*\|^2\\
=&\mathbb{E}\|\Phi_{\boldsymbol{\mathcal{X}}}\left(\boldsymbol{x}^k-\eta_k\boldsymbol{g}^k\right)-\boldsymbol{x}^*\|^2\\
\overset{①}{\leq}&\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k-\eta_k(\nabla f^k-\boldsymbol{g}^k)\|^2\\
=&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k\|^2+\eta_k^2\|\nabla f^k-\boldsymbol{g}^k\|^2-2\eta_k\langle\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k,\nabla f^k-\boldsymbol{g}^k\rangle\right)\\
\overset{②}{\leq}&\mathbb{E}\|\boldsymbol{x}^k-\boldsymbol{x}^*-\eta_k\nabla f^k\|^2+\frac{8\eta_k(1+\eta_k\ell)GD}{\sqrt{s_k}}+\frac{24\eta_k^2G^2}{s_k}\\
=&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2-2\eta_k\langle\boldsymbol{x}^k-\boldsymbol{x}^*,\nabla f^k\rangle+\eta_k^2\|\nabla f^k\|^2\right)+\frac{8\eta_k(1+\eta_k\ell)GD}{\sqrt{s_k}}+\frac{24\eta_k^2G^2}{s_k}\\
\overset{③}{\leq}&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2+2\eta_k(f^*-f^k)+2\ell\eta_k^2(f^k-f^*)\right)+\frac{8\eta_k(1+\eta_k\ell)GD}{\sqrt{s_k}}+\frac{24\eta_k^2G^2}{s_k}\\
=&\mathbb{E}\left(\|\boldsymbol{x}^k-\boldsymbol{x}^*\|^2-2\eta_k(1-\ell\eta_k)(f^k-f^*)\right)+\frac{8\eta_k(1+\eta_k\ell)GD}{\sqrt{s_k}}+\frac{24\eta_k^2G^2}{s_k},
\end{aligned}
\tag{20}
$$

where ① holds due to $\boldsymbol{x}^*\in\boldsymbol{\mathcal{X}}$. ② holds since we use Corollary 1 and Eqn. (19), and ③ holds due to the convexity of $f(\boldsymbol{x})$:

$$
f^*-f^k\geq-\langle\nabla f^k,\boldsymbol{x}^k-\boldsymbol{x}^*\rangle,
$$

and the ℓ-smooth property of $f(\boldsymbol{x})$:

$$
f^*\leq\inf_{\boldsymbol{y}}\left(f(\boldsymbol{x})+\langle\nabla f(\boldsymbol{x}),\boldsymbol{y}-\boldsymbol{x}\rangle+\frac{\ell}{2}\|\boldsymbol{y}-\boldsymbol{x}\|^2\right)=f(\boldsymbol{x})-\frac{1}{2\ell}\|\nabla f(\boldsymbol{x})\|^2,
$$

where we set $\boldsymbol{y}=\boldsymbol{x}-\nabla f(\boldsymbol{x})/\ell$.

Next we sum up Eqn. (20) from $k=\theta T$ to $T-1$ and obtain

$$
\sum_{k=\theta T}^{T-1}2\eta_k(1-\ell\eta_k)\mathbb{E}(f^k-f^*)\leq\|\boldsymbol{x}^{\theta T}-\boldsymbol{x}^*\|^2-\|\boldsymbol{x}^T-\boldsymbol{x}^*\|^2+\sum_{k=\theta T}^{T-1}\left[\frac{8\eta_k(1+\eta_k\ell)GD}{\sqrt{s_k}}+\frac{24\eta_k^2G^2}{s_k}\right].
$$

Here we set $\eta_k=\frac{1}{2\ell}$. Then it yields

$$
\begin{aligned}
&\frac{1}{(1-\theta)T}\sum_{k=\theta T}^{T-1}\mathbb{E}(f^k-f^*)\\
\leq&\frac{2\ell}{(1-\theta)T}\left(\|\boldsymbol{x}^{\theta T}-\boldsymbol{x}^*\|^2-\|\boldsymbol{x}^T-\boldsymbol{x}^*\|^2\right)+\frac{1}{(1-\theta)T}\sum_{k=\theta T}^{T-1}\left(\frac{12GD}{\sqrt{s_k}}+\frac{12G^2}{\ell s_k}\right).
\end{aligned}
\tag{21}
$$

Then, we further set $s_k=(k+1)^2$. We have

$$
\sum_{k=\theta T}^{T-1}\frac{1}{\sqrt{s_k}}=\sum_{k=\theta T}^{T-1}\frac{1}{k+1}\leq\int_{\theta T}^{T-1}\frac{1}{x}dx=\log(x)\,|_{\theta T}^{T-1}\leq\log\left(\frac{1}{\theta}\right)
$$

and

$$
\sum_{k=\theta T}^{T-1}\frac{1}{s_k}=\sum_{k=\theta T}^{T-1}\frac{1}{(k+1)^2}\leq\sum_{k=\theta T}^{T-1}\left(\frac{1}{k}-\frac{1}{k+1}\right)\leq\frac{1}{\theta T}.
$$

Finally, we submit the above inequalities into Eqn. (21) and set $\theta=\frac{1}{2}$:

$$
\begin{aligned}
\mathbb{E}(f(\boldsymbol{x}^a)-f(\boldsymbol{x}^*))=&\frac{1}{(1-\theta)T}\sum_{k=\theta T}^{T-1}\mathbb{E}(f^k-f^*)\\
\leq&\frac{4\ell}{T}\|\boldsymbol{x}^{\theta T}-\boldsymbol{x}^*\|^2+\frac{24GD}{T}+\frac{48G^2}{\ell T^2}\leq\frac{4\ell D^2+24GD}{T}+\frac{48G^2}{\ell T^2}.
\end{aligned}
$$

The proof is completed. □

B.5 Proof of Corollary 2

Proof. From Theorem 3, we know that the convergence rate is decided by $\mathcal{O}((6GD+\ell D^2)/T)$. In order to achieve ϵ accuracy, we need $T \geq \mathcal{O}(\frac{6GD+\ell D^2}{\epsilon})$. So the IFO complexity of the algorithm is

$$\mathcal{O}(1^2+2^2+\cdots+T^2) = \mathcal{O}\left(\frac{(6GD+\ell D^2)^3}{\epsilon^3}\right).$$

The proof is completed. □

B.6 Proof of Theorem 4

For brevity, here we use f^k and f^* to denote $f(\boldsymbol{x}^k)$ and $f(\boldsymbol{x}^*)$, respectively.

Proof. From Eqn. (14) in Sec. B.3, by setting $\eta_k = \frac{1}{\ell}$ we have

$$\mathbb{E}f^{k+1} \leq \mathbb{E}\left[f^k + \frac{12G^2}{\ell s_k} - \frac{1}{2\ell}\|\nabla f^k\|^2\right]. \tag{22}$$

We set $s_k = k+1$ and sum up Eqn. (22) from $k=\theta T$ to $T-1$:

$$\begin{aligned}\frac{1}{(1-\theta)T}\sum_{k=\theta T}^{T-1}\mathbb{E}\|\nabla f^k\|^2 \leq& \frac{2\ell}{(1-\theta)T}(f^{\theta T}-f^T) + \frac{24G^2}{(1-\theta)T}\sum_{k=\theta T}^{T-1}\frac{1}{s_k}\\ \overset{①}{\leq}& \frac{2\ell}{(1-\theta)T}(f^{\theta T}-f^T) + \frac{24G^2}{(1-\theta)T}\log\left(\frac{1}{\theta}\right),\end{aligned}$$

where ① holds since we have

$$\sum_{k=\theta_1 T+1}^{\theta_2 T}\frac{1}{s_k} \leq \int_{\theta_1 T}^{\theta_2 T-1}\frac{1}{x}dx = \log(x)\Big|_{\theta_1 T}^{\theta_2 T-1} \leq \log\left(\frac{\theta_2}{\theta_1}\right).$$

Suppose we are given arbitrary $\boldsymbol{x}_1 \in \boldsymbol{\mathcal{X}}$ and $\boldsymbol{x}_2 \in \boldsymbol{\mathcal{X}}$ that satisfy $\|\boldsymbol{x}_1-\boldsymbol{x}_2\|_2 \leq D$, and $f(\boldsymbol{x})$ is ℓ-smooth. We have

$$f^{\theta T}-f^T = (f^{\theta T}-f^*)-(f^T-f^*) \leq \frac{\ell}{2}\|\boldsymbol{x}^{\theta T}-\boldsymbol{x}^*\|_2^2 + \frac{\ell}{2}\|\boldsymbol{x}^T-\boldsymbol{x}^*\|_2^2 \leq \ell D^2.$$

By setting $\theta = 1/2$, we can further establish

$$\mathbb{E}\|\nabla f(\boldsymbol{x}^a)\|^2 = \frac{1}{0.5T}\sum_{k=0.5T+1}^{T}\mathbb{E}\|\nabla f^k\|^2 \leq \frac{4\ell^2D^2+35G^2}{T}.$$

The proof if completed. □

B.7 Proof of Corollary 3

Proof. From Theorem 4, we know

$$\mathbb{E}\|\nabla f(\boldsymbol{x}^a)\|^2 \leq \frac{4\ell^2D^2+35G^2}{T}.$$

In this case, we can further achieve $\mathbb{E}\|\nabla f(\boldsymbol{x}^a)\|^2 \leq \epsilon$. We need $T \geq \frac{4\ell^2D^2+35G^2}{\epsilon}$. So the IFO complexity is

$$\mathcal{O}\left(\frac{\left(4\ell^2D^2+35G^2\right)^2}{\epsilon^2}\right).$$

The proof is completed. □

C Proofs of Results in Section 4

C.1 Proof of Theorem 5

Before proving Theorem 5, we first give a useful lemma stated in Lemma 4.

Lemma 4. *[1] For the convex function $f(\boldsymbol{x}) = g(\boldsymbol{A}\boldsymbol{x})$, if the function $g(\cdot)$ is α-strongly convex and $\boldsymbol{\mathcal{X}}$ is a compact set, then $f(\boldsymbol{x})$ satisfies Polyak-Łojasiewicz (PL) inequality:*

$$\mu(f(\boldsymbol{x}) - f(\boldsymbol{x}^*)) \leq \frac{1}{2}\|\nabla f(\boldsymbol{x})\|_2^2,$$

where $\mu = \alpha\sigma(\boldsymbol{A})$ in which α is a universal constant and $\sigma(\boldsymbol{A})$ denotes the smallest non-zero singular value of the matrix $\boldsymbol{A}$.

Now we are to prove Theorem 5. For brevity, here we use f^k and f^* to denote $f(\boldsymbol{x}^k)$ and $f(\boldsymbol{x}^*)$, respectively.

Proof. From Eqn. (14) in Sec. B.3, by setting $\eta_k = \frac{1}{\ell}$ we have

$$\mathbb{E}f^{k+1} \leq \mathbb{E}\left[f^k + \frac{12G^2}{\ell s_k} - \frac{1}{2\ell}\|\nabla f^k\|^2\right]. \tag{23}$$

Then since each individual function $f_i(\boldsymbol{x})$ is of form $f_i(\boldsymbol{x}) = h(\langle \boldsymbol{a}_i, \boldsymbol{x}\rangle)$, we can formulate $f(\boldsymbol{x}) = g(\boldsymbol{A}\boldsymbol{x})$, where $\boldsymbol{A} = [\boldsymbol{a}_1^T; \boldsymbol{a}_2^T; \cdots, \boldsymbol{a}_n^T]$ (namely, each row denotes a datum vector). Since $h(\cdot)$ is stongly convex, then by Lemma 4 we know $g'(\boldsymbol{x}) = g(\boldsymbol{A}\boldsymbol{x})$ satisfies the Polyak-Łojasiewicz (PL) inequality:

$$\mu(g'(\boldsymbol{x}) - g'(\boldsymbol{x}^*)) \leq \frac{1}{2}\|\nabla g'(\boldsymbol{x})\|_2^2,$$

where $\mu = \alpha\sigma(\boldsymbol{A})$ in which $\sigma(\boldsymbol{A})$ denotes the smallest non-zero singular value of the matrix $\boldsymbol{A}$. It can be easily verified that $\mu = \alpha\sigma(\boldsymbol{A}) \leq \ell$. Note that the most commonly used optimization losses, namely least square and logistic regression, satisfy such a PL inequality [1]. Thus, by substituting the above PL inequality into Eqn. (23), it yields

$$\mathbb{E}f^{k+1} \leq \mathbb{E}\left[f^k + \frac{12G^2}{\ell s_k} - \frac{\mu}{2\ell}(f^k - f^*)\right],$$

which is actually equivalent to

$$\mathbb{E}[f^{k+1} - f^*] \leq \left(1 - \frac{\mu}{\ell}\right)\mathbb{E}[f^k - f^*] + \frac{12G^2}{\ell s_k}.$$

Then we set $s_k = \tau(1/\zeta)^k$, where $\zeta \in (0, 1)$. Then by considering $\ell \geq \rho$, it yields

$$\mathbb{E}[f^{k+1} - f^*] \leq \left(1 - \frac{\mu}{\ell}\right)\mathbb{E}[f^k - f^*] + \frac{12G^2}{\ell\tau}\zeta^k.$$

For brevity, let $\alpha = 1 - \frac{\mu}{\ell}$ and $\gamma = \frac{12G^2}{\tau\ell}$. Thus, we have

$$\mathbb{E}[f^{k+1} - f^*] \leq \alpha[f^k - f^*] + \gamma\zeta^k.$$

We further assume that τ is large enough such that

$$\gamma = \frac{12G^2}{\tau\ell} \leq \delta(f^0 - f^*), \tag{24}$$

where δ is a positive constant and will be discussed later. Now we use mathematical induction to prove

$$\mathbb{E}(f^k - f^*) \leq \theta^k\left(f^0 - f^*\right), \tag{25}$$

where $\theta < 1$ is a constant and will be given below.

Obviously, when $k = 0$, Eqn. (25) holds. Now assume that for all $t \leq k$, Eqn. (25) holds. Then for $t = k + 1$, we have

$$\begin{aligned}\mathbb{E}(f^{k+1} - f^*) \leq & \alpha \mathbb{E}(f^k - f^*) + \beta\zeta^k \\ \leq & \alpha\theta^k \left(f^0 - f^*\right) + \beta\zeta^k \\ \overset{①}{\leq} & \left(\alpha + \delta\right)\theta^k \left(f^0 - f^*\right) \\ \overset{②}{\leq} & \theta^{k+1} \left(f^0 - f^*\right),\end{aligned}$$

where ① and ② hold since we let

$$\theta \geq \max(\zeta, \alpha + \delta). \tag{26}$$

This means that if Eqn. (26) holds, then Eqn. (25) always holds. So the conclusion holds.

Now we discuss the values of θ, ζ and τ to make Eqn. (26) satisfied. We just set $\delta = \frac{\mu}{2\ell}$, $\tau \geq \frac{24G^2}{\mu(f^0 - f^*)}$ and $\theta = \zeta = 1 - \frac{\mu}{2\ell}$, giving

$$\theta \geq \alpha + \delta = 1 - \frac{\mu}{2\ell}.$$

In this case, all the conditions hold, including Eqn. (24) and (26). So we can see that the values of θ, ζ and τ are proper. Therefore, we have

$$\mathbb{E}(f^k - f^*) \leq \left(1 - \frac{\mu}{2\ell}\right)^k (f^0 - f^*).$$

The proof if completed. □

C.2 Proof of Corollary 4

Proof. To achieve ϵ-accurate solution, *i.e.*

$$\mathbb{E}(f^k - f^*) \leq \theta^k (f^0 - f^*) \leq \epsilon,$$

where $\theta = 1 - \frac{\mu}{2\ell}$, we have

$$k^* \geq \log_{1/\theta}\left(\frac{f^0 - f^*}{\epsilon}\right).$$

Therefore, the IFO complexity is

$$\begin{aligned}\tau\left[1 + \frac{1}{\zeta} + \cdots + \frac{1}{\zeta^{k^*-1}}\right] = & \tau\frac{(1/\zeta)^{\log_{1/\theta}\left(\frac{f^0 - f^*}{\epsilon}\right)} - 1}{1/\zeta - 1} = \frac{\tau}{1/\zeta - 1}\left[\frac{f^0 - f^*}{\epsilon} - 1\right] \\ \leq & \frac{\tau}{1/\zeta - 1}\left[\frac{f^0 - f^*}{\epsilon}\right] \leq \mathcal{O}\left(\frac{48\ell G^2}{\mu^2\epsilon}\right).\end{aligned}$$

The proof is completed. □

D Additional Experimental Results

D.1 Descriptions of Testing Datasets and Compared Algorithms

We first briefly introduce the ten testing datasets in the manuscript. Among them, nine datasets are provided in the LibSVM website[1], including ijcnn1, a9a, w8a, covtype, rcv11, protein, satimage, sensorless and letter. We also evaluate our algorithms on the mnist[2] dataset, which is a very commonly used handwritting recognition dataset. Their detailed information is summarized in Table 2. We can observe that these datasets are different from each other in feature dimension, training samples, and class numbers, *etc*.

Now we briefly introduce the compared algorithms in the manuscript, including SVRG [2], SAGA [3], AVRG [4] and SCGC [5]. Since SGD is well known, here we do not introduce it.

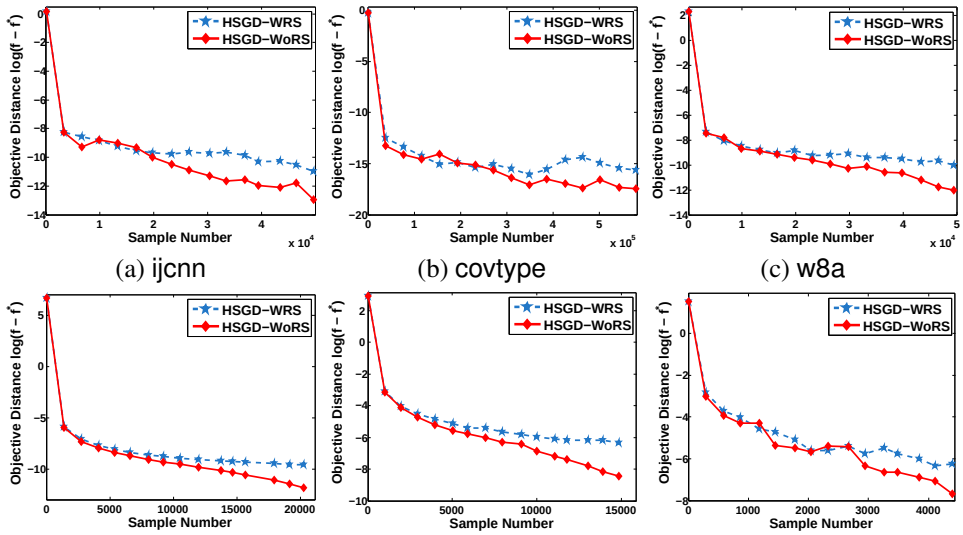
[1] https://www.csie.ntu.edu.tw/ cjlin/libsvmtools/datasets/

[2] http://yann.lecun.com/exdb/mnist/

Table 2: Descriptions of the ten testing datasets.

	#class	#sample	#feature		#class	#sample	#feature
ijcnn1	2	49,990	22	protein	3	14,895	357
a9a	2	32,561	123	satimage	6	4,435	36
w8a	2	49,749	300	sensorless	7	2,310	19
covtype	2	581,012	54	letter	26	10,500	16
rcv11	2	20,242	47,236	mnist	10	60,000	784

- SVRG: It is a variance-reduced variant of SGD. At the k-th epoch, it firstly computes the full gradient $\nabla f(\tilde{\boldsymbol{x}})$ at a snapshot point $\tilde{\boldsymbol{x}}$. Typically, the snapshot point $\tilde{\boldsymbol{x}}$ is set to the final output $\boldsymbol{x}^{k-1}$ of the previous epoch. Then it updates the variables as $\boldsymbol{x}_t^k = \boldsymbol{x}_{t-1}^k - \eta_t \left(f_{i_t}(\boldsymbol{x}_{t-1}^k) - f_{i_t}(\tilde{\boldsymbol{x}}) + \nabla f(\tilde{\boldsymbol{x}})\right)$ where i_t is the sampled index at the t-th iteration in the k-th epoch. The iteration number T in each epoch is usually set to the sample number n and the final output of the k-th epoch is usually the final computed solution $\boldsymbol{x}_T^k$ in this epoch in implementation.
- SAGA: It needs a table to store the gradient of historical computed variables. Specifically, let the initial point denoted by $\boldsymbol{x}^0$ and the known gradient $\nabla f_i(\phi_i^0)$ $(i = 1, \cdots, n)$ where $\phi_i^0 = \boldsymbol{x}^0$. Then at the k iteration, it picks an index j at random. Then it sets $\phi_j^k = \boldsymbol{x}^{k-1}$ and stores $\nabla f_j(\phi_j^k)$ in the table. All other entries in the table remain unchanged. Finally, it updates $\boldsymbol{x}^k$ as $\boldsymbol{x}^k = \boldsymbol{x}^{k-1} - \eta_k \left(f_j(\phi_j^k) - f_j(\phi_j^{k-1}) + \frac{1}{n}\sum_{i=1}^n f_i(\phi_i^{k-1})\right)$. SAGA is also a variance-reduced method.
- AVRG: It uses the historical gradient to estimate full gradient of the snapshot point in SVRG. Namely, at each epoch, it sums up the gradient $f_{i_t}(\boldsymbol{x}_{t-1}^k)$ and uses its average as the estimation of $f(\tilde{\boldsymbol{x}})$ in next epoch. Such a strategy can reduce computational complexity.
- SCGC: It has similar updating process as SVRG. Namely, it also takes the output in the previous epoch as the current snapshot point. But at each epoch, it only samples a subset $\mathcal{S}_k$ of data and uses the average gradient $g(\tilde{\boldsymbol{x}})$ of the samples in $\mathcal{S}_k$ at the snapshot point $\tilde{\boldsymbol{x}}$ to estimate the full gradient at $\tilde{\boldsymbol{x}}$. During the iteration, it updates $\boldsymbol{x}_t^k$ as $\boldsymbol{x}_t^k = \boldsymbol{x}_{t-1}^k - \eta_t \left(f_{i_t}(\boldsymbol{x}_{t-1}^k) - f_{i_t}(\tilde{\boldsymbol{x}}) + g(\tilde{\boldsymbol{x}})\right)$ where i_t is the sampled index from $\mathcal{S}_k$ in the t-th iteration. Typically, the size of S_k gradually increases along with more iterations. Note that to date there has been no work analyzing the convergence performance of SCGC under WoRS.



(a) ijcnn (b) covtype (c) w8a

(d) rcv11 (e) protein (f) satimage

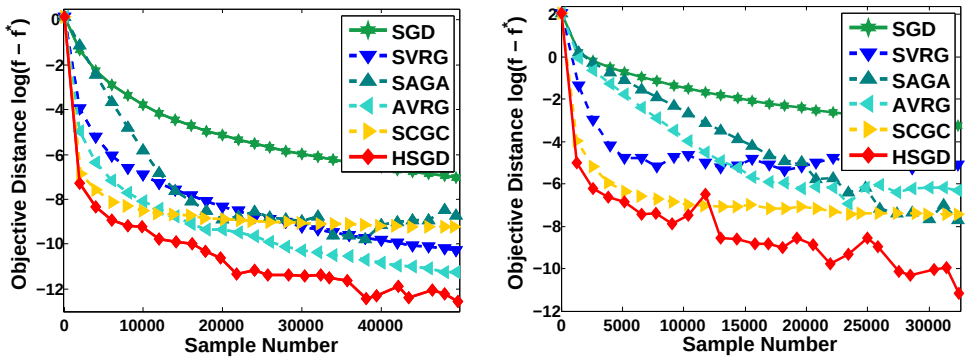
Figure 5: WoRS *vs.* WRS in HSGD. We test logistic regression (regularization parameter $\lambda = 0.01$) on ijcnn, covtype, w8a and rcv11, and evaluate softmax regression (regularization parameter $\lambda = 0.1$) on protein and satimage.

D.2 Comparison between WoRS and WRS in HSGD

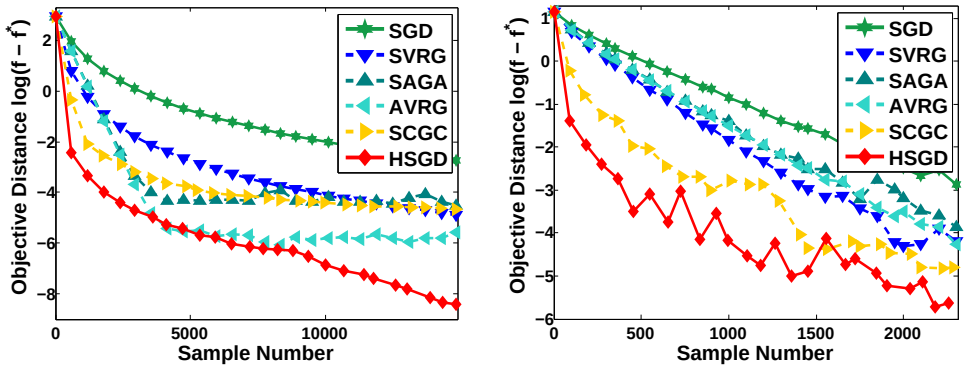
Then we present more experimental results to compare WoRS and WRS. Since the ℓ_2-regularized logistic and multi-class softmax regression problems are strongly convex, we follow Theorem 2 to exponentially expand the mini-batch size s_k in HSGD and set $\tau = 1$. From the comparison in Figure 5, we can find that WoRS strategy often outperforms WRS in the anaphasis of going through data for one pass, while at the beginning of the iteration, their performance is mostly the same. This is because at the beginning, only a few samples are selected and it is highly probable for WRS to select different samples, which is almost the same as WoRS. Thus, their performance in the early phase is very similar. In contrast, as the iteration proceeds, more samples are required. It is likely that WRS selects repeated samples which provide redundant descending information (gradient). By comparison, WoRS has no such weakness as it uses different samples. So it can utilize all samples more effectively and runs faster.

D.3 More Experiments on A Single Pass over Data

Finally, we give more experimental results on a single pass over data. Following the setting in Section 5.1 in manuscript, here we test the considered algorithms on logistic and multi-class softmax regression with regularization parameters $\lambda = 0.01$ and $\lambda = 0.1$, respectively. Similarly, since the ℓ_2-regularized logistic and multi-class softmax regression problems are strongly convex, we follow Theorem 2 to exponentially expand the mini-batch size s_k in HSGD and set $\tau = 1$. Figure 6 summarizes the numerical results in this setting. One can observe that on these well-conditioned tasks, most algorithms still achieve high accuracy after one pass over data, while HSGD (WoRS) also converges significantly faster than the other algorithms. These observations are consistent with the results in Figure 2 in the manuscript. All these results demonstrate the high efficiency of HSGD and also confirm the theoretical implication of Corollary 1 that HSGD is cheaper in IFO complexity ($\mathcal{O}\left(\frac{\kappa^2}{\epsilon}\right)$) than other considered variance-reduced algorithms ($\mathcal{O}\left(n\kappa^2 \log\left(\frac{1}{\epsilon}\right)\right)$) when the desired accuracy is moderately small and data scale is large.



(a) Logistic regression on ijcnn (left) and A09 (right).



(b) Softmax on protein (left) and sensorless (right).

Figure 6: Single-epoch processing: comparison of randomized algorithms for a single pass over data.

References

[1] H. Karimi, J. Nutini, and M. Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *ECML/KDD*, pages 795–811. Springer, 2016. 11

[2] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Proc. Conf. Neutral Information Processing Systems*, pages 315–323, 2013. 12

[3] A. Defazio, F. Bach, and S. Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Proc. Conf. Neutral Information Processing Systems*, pages 1646–1654, 2014. 12

[4] B. Ying, K. Yuan, and A. H. Sayed. Convergence of variance-reduced stochastic learning under random reshuffling. *arXiv preprint arXiv:1708.01383*, 2017. 12

[5] L. Lei and M. Jordan. Less than a single pass: Stochastically controlled stochastic gradient. In *Artificial Intelligence and Statistics*, pages 148–156, 2017. 12