Proximal Gradient Algorithm for Nonconvex Problems

March 2018

1 Introduction

2 introduction to the problem

Proximal gradient (PG) methods (Mine and Fukushima, 1981; Nesterov, 2004; Parikh, Boyd, and others, 2014) are a class of powerful optimization tools in artificial intelligence and machine learning. In general, it considers the following nonsmooth optimization problem:

$$\min_{x\in\mathbb{R}^d} f(x)+h(x) \tag{1}$$

where $f(x)$ usually is the loss function such as hinge loss and logistic loss, and $h(x)$ is the nonsmooth structure regularizer such as l_1-norm regularization.

2.1 Background in research

In recent research, Beck and proposed the accelerate PG methods to solve convex problems by using the Nesterov's accelerated technique. After that, Li and Lin (2015) presented a class of accelerated PG methods for non-convex optimization. To solve the big data problems, the incremental or stochastic PG methods (Bertsekas, 2011; Xiaoand Zhang, 2014) were developed for large-scale convex optimization. There has been extensive research when $f(x)$ is convex (see e.g., Teboulle (2009); Nesterov (2013), [Xiao and Zhang, 2014, Defazio et al., 2014, Lan and Zhou, 2015, Allen-Zhu, 2017a]). In particular, if f_i s are strongly-convex, toi solve large-scale problems Xiao and Zhang [2014] proposed the Prox-SVRG algorithm, which achieves a linear convergence rate, based on the well-known variance reduction technique SVRG developed in [Johnson and Zhang, 2013]. In recent years, due to the increasing popularity of deep learning, the non-convex case has attracted significant attention. Correspondingly, Ghadimi, Lan, and Zhang (2016); Reddi et al. (2016) proposed the stochastic PG methods for large-scale nonconvex optimization. Very recently, Zhou et al. [2018] proposed an algorithm with stochastic gradient complexity $\widetilde{O}(\min\{\frac{1}{\epsilon^{3/2}}, \frac{n^{1/2}}{\epsilon}\})$, improving the previous results $O(\frac{1}{\epsilon^{5/3}})$ [Lei et al., 2017] and $O(\frac{n^{2/3}}{\epsilon})$ [Allen-Zhu and Hazan, 2016]. For the more general nonsmooth nonconvex case, the research is still somewhat limited. More

recently, Gu, Huo, and Huang (2018) introduced inexact PG methods for nonconvex nonsmooth optimization.

2.2 Reason to use zeroth–order techniques

However, in many machine learning problems, the explicit expressions of gradients are difficult or infeasible to obtain. However, there exist situations where the first-order gradient information is computationally infeasible, expensive, or impossible, while the zeroth-order functional information can be easily obtained For example, in some complex graphical model inference (Wainwright, Jordan, and others, 2008) and structure prediction problems (Sokolov, Hitschler, and Riezler, 2018), it is difficult to compute the explicit gradients of the objective functions. Even worse, in bandit (Shamir, 2017) and black-box learning (Chen et al., 2017) problems, only the objective function values are available (the explicit gradients cannot be calculated). For example, in online auctions and advertisement selections, only function values are revealed as feedbacks for algorithms [Wibisono et al., 2012]. In stochastic structured predictions, explicit differentiations may be difficult to per- form while the functional evaluations of predicted structures are easily obtained [Sokolov et al., 2016].

Clearly, the above PG methods will fail in dealing with these scenarios. The optimization problem of Eq. (1) in such situations is referred to SZCO. ZO algorithms achieve gradient-free optimization by approximating the full gradient via gradient estimators based on only the function values [8, 9]. The gradient-free (zeroth-order) optimization method (Nesterov and Spokoiny, 2017) is a promising choice to address these problems because it only uses the function values in optimization process. Thus, the gradient-free optimization methods have been increasingly embraced for solving many ma- chine learning problems. Hence, Zeroth-order (gradient-free) optimization is increasingly embraced for solving machine learning problems where explicit expressions of the gradients are difficult or infeasible to obtain. Recent examples have shown zeroth-order (ZO) based generation of prediction-evasive, black-box adversarial attacks on deep neural networks (DNNs) as effective as state-of-the-art white-box attacks, despite leveraging only the inputs and outputs of the targeted DNN [1–3] (Conn, Scheinberg, and Vicente, 2009). Additional classes of applications include network control and management with time-varying constraints and limited computation capacity [4, 5], and parameter inference of black-box systems [6, 7].

The main issue for those accelerated algorithms is that most of their algorithm designs (e.g., (Allen-Zhu, 2017) and (Hien et al., 2017)) involve tracking at least two highly correlated coupling vectors 2 (in the inner loop). This kind of algorithm structure prevents us from deriving efficient (lock-free) asynchronous sparse variants for those algorithms.

2.3 Problem with existing methods

Although many ZO algorithms have recently been developed and analyzed [5, 10–18], they often suffer from the high variances of ZO gradient estimates, and in turn, hampered convergence rates. A useful technique to accelerate the convergence of SZCO is by leveraging variance reduction method.

In addition, these algorithms are mainly designed for convex settings, which limits their applicability in a wide range of (non-convex) machine learning problems.

A key concern in the development of iterative stochastic zeroth-oder algorithms for solving Eq. (1) is the order of the necessary number of functional evaluations in the form of $f(x,\xi)$, which is termed as sample complexity or iteration complexity.

2.4 Compare with other methods

In Table 1, we summarize the convergence rates and the function query complexities of ZO-SVRG and its two variants, which we call ZO-SVRG-Ave and ZO-SVRG-Coord, respectively. For comparison, we also present the results of ZO-SGD [24] and ZO-SVRC [26], where the later updates J coordinates per iteration within an epoch. Table 1 shows that ZO-SGD has the lowest query complexity but has the worst convergence rate. ZO-SVRG-coord yields the best convergence rate in the cost of high query complexity. By contrast, ZO-SVRG (with an appropriate mini-batch size) and ZO-SVRG-Ave could achieve better trade-offs between the convergence rate and the query complexity. RSG [] do not provide the complexity of non-smooth function.

2.5 what we want to do

In this paper, thus, we propose a class of faster gradient-free proximal stochastic methods for solving the nonconvex nonsmooth problem as follows: In this paper, we consider nonsmooth nonconvex finite-sum optimization problems of the form

$$\min_{x\in\mathbb{R}^d} F(x) = f(x) + h(x), \;\; f(x) := \frac{1}{n}\sum_{i=1}^{n} f_i(x) \tag{2}$$

where each $f_i(x)$ is possibly nonconvex and smooth loss function, and $h(x)$ is a convex and nonsmooth regularization term. The generic form (**??**) encompasses many machine learning problems, ranging from generalized linear models to neural networks. This above optimization problem is fundamental to many machine learning problems, ranging from convex optimization such as Lasso, SVM to highly nonconvex problem such as optimizing deep neural networks. We next elaborate on assumptions of problem (1), and provide a background on ZO gradient estimators.

We will study the design and analysis of variance reduced and faster converging ZO optimization methods for (**??**). To reduce the variance accelerate ZO optimization, one can draw motivations from similar ideas in the first-order regime. The stochastic variance reduced gradient (SVRG) is a commonly-used, effective first-order approach to reduce the variance [19–23]. Due to the variance reduction, it improves the convergence rate of stochastic gradient descent (SGD) from $O(1/\sqrt{T})$ to $O(1/T)$, where T is the total number of iterations.

3 Main Challenge

Although SVRG has shown a great promise, applying similar ideas to ZO optimization is not a trivial task. The main challenge arises due to the fact that SVRG relies upon the assumption that a stochastic gradient is an unbiased estimate of the true batch/full gradient, which unfortunately does not hold in the ZO case. Therefore, it is an open question whether the ZO stochastic variance reduced gradient could enable faster convergence of ZO algorithms. In this paper, we attempt to fill the gap between ZO optimization and SVRG.

4 existing works

Until now, there are few zeroth-order stochastic methods for solving the problem (2.5) except a recent attempt proposed in (Ghadimi, Lan, and Zhang, 2016). Specifically, Ghadimi, Lan, and Zhang (2016) have proposed a randomized stochastic projected gradient-free method (RSPGF), i.e., a zeroth-order proximal stochastic gradient method. However, due to the large variance of zeroth-order estimated gradient generated from randomly selecting the sample and the direction of derivative, the RSPGE only has a conver- gence rate $O(\frac{1}{\sqrt{T}})$, which is significantly slower than $O(\frac{1}{T})$, the best convergence rate of the zeroth-order stochastic algorithm. To accelerate the RSPGF algorithm, we use the variance reduction strategies in the first-order methods, i.e., SVRG (Xiao and Zhang, 2014) and SAGA (Defazio, Bach, and Lacoste-Julien, 2014), to reduce the variance of estimated stochastic gradient.

5 Main contributions

Although SVRG and SAGA have shown good performances, applying these strategies to the zeroth-order method is not a trivial task. The main challenge arises due to that both SVRG and SAGA rely on the assumption that a stochastic gradient is an unbiased estimate of the true full gradient. However, it does not hold in the zeroth-order algorithms. We propose and evaluate a novel ZO algorithm for nonconvex stochastic optimization, ZO-SVRG, which integrates SVRG with ZO gradient estimators. We show that compared to SVRG, ZO-SVRG achieves a similar convergence rate that decays linearly with $O(1/T)$ but up to an additional error correction term of order $1/b$, where b is the mini-batch size.

Our work offers a comprehensive study on how ZO gradient estimators affect SVRG on both iteration complexity (i.e., convergence rate) and function query complexity. Compared to the existing ZO algorithms, our methods can strike a balance between iteration complexity and function query complexity. Our main technical contribution lies in the new convergence analysis of ProxSVRG+, which has notable difference from that of ProxSVRG [Reddi et al., 2016b]. We list our results in Table 1–3 and Figure 1–2. Our convergence results are stated in terms of the number of stochastic first-order oracle (SFO) calls and proximal oracle (PO) calls (see Definition 2). We would like to highlight the following results yielded by our new analysis:

In the paper, thus, we will fill this gap between zeroth-order proximal stochastic method and the classic variance reduction approaches (SVRG and SAGA).

Moreover, we provide the theoretical analysis on the convergence properties of both new ZO-ProxSVRG and ZO-ProxSAGA methods. Table 1 shows the specifical convergence rates of the proposed algorithms and other related ones. In particular, our algorithms have faster convergence rate $O(\frac{1}{T})$ than $O(\frac{1}{\sqrt{T}})$ of the RSPGF (Ghadimi, Lan, and Zhang, 2016) (the existing stochastic PG algorithm for solving nonconvex nonsmoothing problems).

The expectational results R-I and R-II have better dependence on d compared to the high probability result R-III.

1) ProxSVRG+ is b (resp. $\sqrt{b}\epsilon n$) times faster than ProxGD in terms of $\#SFO$ when $b \leq n^{2/3}$ (resp. $b \leq 1/\epsilon^{2/3}$), and n/b times faster than ProxGD when $b > n^{2/3}$ (resp. $b > 1/\epsilon^{2/3}$). Note that $\#PO = O(1/\epsilon)$ for both ProxSVRG+ and ProxGD. Obviously, for any super constant b, ProxSVRG+ is strictly better than ProxGD. Hence, we partially answer the open question (i.e. developing stochastic methods with provably better performance than ProxGD with constant minibatch size b) proposed in [Reddi et al., 2016b]. ProxSVRG+ also matches the best result achieved by ProxSVRG at $b = n^{2/3}$, and it is strictly better for smaller b (using less PO calls). See Figure 1 for an overview.

2) Assuming that the variance of the stochastic gradient is bounded (see Assumption 1), i.e. online/stochastic setting, ProvSVRG+ generalizes the best result achieved by SCSG, recently proposed by [Lei et al., 2017] for the smooth nonconvex case, i.e., $h(x) = 0$ in form (1) (see Table 1, the 5th row). ProxSVRG+ is more straightforward than SCSG and yields simpler proof. Our results also match the results of Natasha1.5 proposed by [Allen-Zhu, 2017b] very recently, in terms of $\#SFO$, if there is no additional assumption (see Footnote 2 for details). In terms of $\#PO$, our algorithm outperforms Natasha1.5.

Extensive experimental results and theoretical analysis demonstrate the effectiveness of our algorithms. To demonstrate the flexibility of our approach in managing this trade-off, we conduct an empirical evaluation of our proposed algorithms and other state-of-the-art algorithms on two diverse applications: black-box chemical material classification and generation of universal adversarial perturbations from black-box deep neural network models. Extensive experimental results and theoretical analysis validate the effectiveness of our approaches.

Without a careful treatment, this correction term (e.g., when b is small) could be a critical factor affecting the optimization performance. To mitigate this error term, we propose two accelerated ZO-SVRG variants, utilizing reduced variance gradient estimators. These yield a faster convergence rate towards $O(\sqrt{d}/T)$, the best known iteration complexity bound for ZO stochastic optimization. In this paper, we propose a very straightforward algorithm called ProxSVRG+ to solve the nons- mooth nonconvex problem (1). Depending on the local error bound (LEB) condition, the improvement over existing results is up to a factor of $\frac{1}{\sqrt{d}}$.

We also note that SCSG [Lei et al., 2017] and ProxSVRG [Reddi et al., 2016b] achieved their best convergence results with $b = 1$ and $b = n^{2/3}$ respectively, while ProxSVRG+ achieves the best result with $b = 1/\epsilon^{2/3}$ (see Figure 1), which is a moderate minibatch size (which is not too small for parallelism/vectorization and not too large for better generalization). In our experiments, the best b for ProxSVRG and ProxSVRG+

in the MNIST experiments is 4096 and 256, respectively (see the second row of Figure 4).

3) For the nonconvex functions satisfying Polyak-Łojasiewicz condition [Polyak, 1963], we prove that ProxSVRG+ achieves a global linear convergence rate without restart, while Reddi et al. [2016b] used PL-SVRG to restart ProxSVRG many times to obtain the linear convergence rate. Thus, ProxSVRG+ can automatically switch to the faster linear convergence in some regions. ProxSVRG+ also improves ProxGD and ProxSVRG/SAGA, and generalizes the results of SCSG in this case (see Table 3). Also see the remarks after Theorem 2 for more details. To the best of our knowledge, this is the first paper that leverages the LP condition for improving the convergence of SZCO.

Among them, accelerated methods enjoy improved convergence rates but have complex coupling structures, which makes them hard to be extended to more settings (e.g., sparse and asynchronous) due to the existence of perturbation. In this paper, we introduce a sim- ple stochastic variance reduced algorithm (MiG), which enjoys the best-known convergence rates for both strongly convex and non-strongly con- vex problems. Moreover, we also present its efficient gradient-free variants, and theoretically analyze its convergence rates in these settings.

More accurately, these methods achieve an improved oracle complexity ? versus ? for improved accelerated methods.

We prove that FSVRG achieves linear convergence for strongly convex problems.

We design a new momentum accelerating update rule, and present two selecting schemes of momentum weights for Cases 1 and 2, respectively.

We prove that FSVRG attains linear convergence.

It is also notable that the best upper bound achieved in this paper can be as good as ?. However, we note that our result does not contradict to the lower bound in [Duchi et al., 2015] because either their considered random functions do not necessarily have bounded gradients as assumed in this paper or their considered problem does not satisfy the LEB condition that yields our best result.

6 Related Works

Gradient-free (zeroth-order) methods have been effectively used to solve many machine learning problems, where the explicit gradient is difficult or infeasible to obtain, and have also been widely studied. In ZO algorithms, a full gradient is typically approximated using either a one-point or a two-point gradient estimator, where the former acquires a gradient estimate $\hat{\nabla} f(x)$ by querying f at a single random location close to x [10, 11], and the latter computes a finite difference using two random function queries [12, 12+1]. In this paper, we focus on the two-point gradient estimator since it has a lower variance and thus improves the complexity bounds of ZO algorithms.

For example, Nesterov and Spokoiny (2017) proposed several random gradient-free methods by using Gaussian smoothing technique. Duchi et al. (2015) proposed a zeroth-order mirror descent algorithm. More recently, Yu et al. (2018); Dvurechensky, Gasnikov, and Gorbunov (2018) presented the accelerated zeroth-order methods for the convex optimization. To solve the nonsmooth problems, the zeroth-order online or

stochastic ADMM methods (Liu et al., 2018b; Gao, Jiang, and Zhang, 2018) have been introduced.

The above zeroth-order methods mainly focus on the (strongly) convex problems. In fact, there exist many non-convex machine learning tasks, whose explicit gradients are not available, such as the nonconvex black-box learning problems (Chen et al., 2017; Liu et al., 2018c). Thus, several recent works have begun to study the zeroth-order stochastic methods for the nonconvex optimization. For example, Ghadimi and Lan (2013) proposed the randomized stochastic gradient-free (RSGF) method, i.e., a zeroth-order stochastic gradient method. To accelerate optimization, more recently, Liu et al. (2018c,a) proposed the zeroth-order stochastic variance reduction gradient (ZO-SVRG) methods. Moreover, to solve the large-scale machine learning problems, some asynchronous parallel stochastic zeroth-order algorithms have been proposed in (Gu, Huo, and Huang, 2016; Lian et al., 2016; Gu et al., 2018).

Despite the meteoric rise of two-point based ZO algorithms, most of the work is restricted to convex problems [5, 14–18]. For example, a ZO mirror descent algorithm proposed by [14] has an exact rate $O(\sqrt{d}/\sqrt{T})$, where d is the number of optimization variables. The same rate is obtained by bandit convex optimization [15] and ZO online alternating direction method of multipliers [5]. Current studies suggested that ZO algorithms typically agree with the iteration complexity of first-order algorithms up to a small-degree polynomial of the problem size d.

In contrast to the convex setting, non-convex ZO algorithms are comparatively under-studied except a few recent attempts [7, 13, 24–26]. Different from convex optimization, the stationary condition is used to measure the convergence of nonconvex methods. In [12+1], the ZO gradient descent (ZO- GD) algorithm was proposed for deterministic nonconvex programming, which yields $O(d/T)$ convergence rate. A stochastic version of ZO-GD (namely, ZO-SGD) studied in [24] achieves the rate of $O(\sqrt{d}/\sqrt{T})$. In [25], a ZO distributed algorithm was developed for multi-agent optimization, leading to $O(1/T + d/q)$ convergence rate. Here q is the number of random directions used to construct a gradient estimate. In [7], an asynchronous ZO stochastic coordinate descent (ZO-SCD) was derived for parallel optimization and achieved the rate of $O(\sqrt{d}/\sqrt{T})$. In [26], a variant of ZO-SCD, known as ZO stochastic variance reduced coordinate (ZO-SVRC) descent, improved the convergence rate from $O(\sqrt{d}/\sqrt{T})$ to $O(d/T)$ under the same parameter setting for the gradient estimation. Although the authors in [26] considered the stochastic variance reduced technique, only a coordinate descent algorithm using a coordinate-wise (deterministic) gradient estimator was studied. This motivates our study on a more general framework ZO-SVRG under different gradient estimators. Recently, for the nonsmooth nonconvex case, Reddi et al. [2016b] provided two algorithms called ProxSVRG and ProxSAGA, which are based on the well-known variance reduction techniques SVRG and SAGA [Johnson and Zhang, 2013, Defazio et al., 2014]. Also, we would like to mention that Aravkin and Davis [2016] considered the case when h can be nonconvex in a more general context of robust optimization. Before that, Ghadimi et al. [2016] analyzed the deterministic proximal gradient method (i.e., computing the full-gradient in every iteration) for nonconvex nonsmooth problems. Here we denote it as ProxGD. Ghadimi et al. [2016] also considered the stochastic case (here we denote it as ProxSGD). However, ProxSGD requires the batch sizes being a large number (i.e., $\Omega(1/\epsilon)$) or

increasing with the iteration number t. Note that ProxSGD may reduce to deterministic ProxGD after some iterations due to the increasing batch sizes. Note that from the perspectives of both computational efficiency and statistical generalization, always computing full-gradient (GD or ProxGD) may not be desirable for large-scale machine learning problems. A reasonable minibatch size is also desirable in practice, since the computation of minibatch stochastic gradients can be implemented in parallel. In fact, practitioners typically use moderate minibatch sizes, often ranging from something like 16 or 32 to a few hundreds (sometimes to a few thousands, see e.g., [Goyal et al., 2017]). 1 Hence, it is important to study the convergence in moderate and constant minibatch size regime.

Reddi et al. [2016b] provided the first non-asymptotic convergence rates for Prox-SVRG with minibatch size at most $O(n^{2/3})$, for the nonsmooth nonconvex problems. However, their convergence bounds (using constant or moderate size minibatches) are worse than the deterministic ProxGD in terms of the number of proximal oracle calls. Note that their algorithms (i.e., ProxSVRG/SAGA) outperform the ProxGD only if they use quite large minibatch size $b = O(n^{2/3})$. Note that in a typical application, the number of training data is $n = 10^6 \sim 10^9$, and $n^{2/3} = 10^4 \sim 10^6$. Hence, $O(n^{2/3})$ is a quite large minibatch size. Finally, they presented an important open problem of developing stochastic methods with provably better performance than ProxGD with constant minibatch size

6.1 Motivation at the end of related works

Although the above zeroth-order stochastic methods can effectively solve the nonconvex optimization, there are few zeroth-order stochastic methods for the nonconvex nonsmooth composite optimization except the RSPGF method presented in (Ghadimi, Lan, and Zhang, 2016). In addition, Liu et al. (2018a) have also studied the zeroth-order algo- rithm for solving the nonconvex nonsmooth problem, which is different from problem (2.5).

7 Accelerated Proximal Gradient Method

Algorithm 1 Nonconvex ProxZOSVRG+

Input: initial point x_0, batch size B, minibatch size b, epoch length m, step size η
Initialize: $\tilde{x}^0 = x_0$
for $s = 1, 2, \ldots, S$ **do**
  $x_0^s = \widetilde{x}^{s-1}$
  $\hat{g}^s = \frac{1}{B}\sum_{j\in I_B} \hat{\nabla} f_j(\widetilde{x}^{s-1})$
  for $t = 1, 2, \ldots, m$ **do**
    $\hat{v}_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s$
    $x_t^s = \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta \hat{v}_{t-1}^s)$
  $\widetilde{x}^s = x_m^s$
Output: $\hat{x}$ chosen uniformly from $\{x_{t-1}^s\}_{t\in[m], s\in[S]}$

Method	Problem	Stepsize	Convergence rate	SZO complexity
ZO-SGD [**?**]	S(NC)	$O\left(\min\{\frac{1}{d}, \frac{1}{\sqrt{dT}}\}\right)$	$O\left(\frac{d}{\epsilon^2}\right)$	$O\left(\frac{nd}{\epsilon^2}b\right)$
RGF[**?**]	NS(C)	$O\left(\frac{1}{\sqrt{dT}}\right)$	$O\left(\frac{d^2}{\epsilon^2}\right)$	$O\left(\frac{nd^2}{\epsilon^2}b\right)$
MD [**?**]	S(SC)	$\frac{1}{\sqrt{dT}}$	$O\left(\frac{d}{\epsilon^2}\right)$	$O\left(\frac{nd}{\epsilon^2}\right)$
ZO-SVRG-Coord [**?**]	S(NC)	$O\left(\frac{1}{d}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O(\frac{nd^2}{\epsilon} + \frac{d^2b}{\epsilon})$
ZO-ProxSVRG[**?**]	S(NC)+NS(C)	$O\left(\frac{1}{d}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O(\frac{nd^2}{\sqrt{b}\epsilon} + \frac{d^2\sqrt{b}}{\epsilon})$
ZO-ProxSAGA[**?**]	S(NC)+NS(C)	$O\left(\frac{1}{d}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O(\frac{nd^2}{\epsilon\sqrt{b}})$
Ours	S(NC)+NS(C)	$O\left(\frac{1}{\sqrt{d}}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O\left(\min\{n, \frac{1}{\epsilon}\}\frac{d\sqrt{d}}{\epsilon\sqrt{b}} + \frac{d\sqrt{db}}{\epsilon}\right)$
Ours	S(PL)+NS(C)	$O\left(\frac{1}{\sqrt{d}}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O(\min\{n, \frac{1}{\epsilon}\}\frac{d\sqrt{d}}{\sqrt{b}}\log\frac{1}{\epsilon} + d\sqrt{db}\log\frac{1}{\epsilon})$
Ours	S(SC)+NS(C)	$O\left(\frac{1}{\sqrt{d}}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O(\min\{n, \frac{1}{\epsilon}\}\frac{d\sqrt{d}}{b^{3/4}}\log\frac{1}{\epsilon} + d\sqrt{d}b^{3/4}\log\frac{1}{\epsilon})$

Table 1: This is the Table

In this section, we propose a proximal stochastic gradient algorithm called Prox-SVRG+, which is very straightforward (similar to nonconvex ProxSVRG [Reddi et al., 2016b] and convex Prox-SVRG [Xiao and Zhang, 2014]). The details are described in Algorithm 1. Our algorithm has two kinds of random procedure. That is, in outer iteration, we compute the gradient include B samples. In inner iteration, we randomly select a mini-batch of samples to estimate the gradient. We call B the batch size and b the minibatch size.

Compared with Prox-SVRG, ProxSVRG [Reddi et al., 2016b] analyzed the non-convex functions while Prox- SVRG [Xiao and Zhang, 2014] only analyzed the convex functions. The major difference of our ProxSVRG+ is that we avoid the computation of the full gradient at the beginning of each epoch, i.e., B may not equal to n (see Line 4 of Algorithm 1) while ProxSVRG and Prox-SVRG used $B = n$. Note that even if we choose $B = n$, our analysis is stronger than ProxSVRG [Reddi et al., 2016b]. Also, our ProxSVRG+ shows that the "stochastically controlled" trick of SCSG [Lei et al., 2017] (i.e., the length of each epoch is a geometrically distributed random variable) is not really necessary for achieving the desired bound. 5 As a result, our straightforward ProxSVRG+ generalizes the result of SCSG to the more general nonsmooth nonconvex case and yields simpler analysis. In this section, we briefly review the zeroth-order proximal stochastic gradient (ZO-ProxSGD) method to solve the problem (2.5). Before that, we first revisit the proximal gradient descent (ProxGD) method (Mine and Fukushima, 1981). ProxGD is an effective method to solve the problem (2.5) via the following iteration:

$$x_t^s = \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta \nabla f(x_{t-1}^s)), \qquad t = 1, 2, \ldots \tag{3}$$

where $\eta > 0$ is a step size, and $\mathrm{Prox}_{\eta h}(\cdot)$ is a proximal operator defined as:

$$\mathrm{Prox}_{\eta h}(x) := \arg\min_{y \in \mathbb{R}^d} \left(h(y) + \frac{1}{2\eta} \|y - x\|^2 \right) \tag{4}$$

As discussed above, because ProxGD needs to compute the gradient at each iteration, it cannot be applied to solve the problems, where the explicit gradient of function f (x) is not available. For example, in the black-box machine learn- ing model, only function values (e.g., prediction results) are available Chen et al. (2017). To avoid computing explicit gradient, we use the zeroth-order gradient estimators (Nesterov and Spokoiny, 2017; Liu et al., 2018c) to estimate the gradient only by function values. We also assume that the nonsmooth convex function $h(x)$ in (**??**) is well structured, i.e., the following proximal operator on h can be computed efficiently:

$$\mathrm{Prox}_{\eta h}(x) := \arg\min_{y\in\mathbb{R}^d}\left(h(y)+\frac{1}{2\eta}\|y-x\|^2\right) \tag{5}$$

7.1 Zeroth-order (ZO) gradient estimators

Given an individual cost function f i (or an arbitrary function under A1 and A2), a two-point random gradient estimator $\hat{\nabla} f_i(x)$ is defined [12+1,16]

$$\hat{\nabla} f_i(x) = \frac{d(f_i(x+\mu u_i)-f_i(x))}{\mu}u_i, \qquad i\in[n], \tag{6}$$

where recall that d is the number of optimization variables, $\mu > 0$ is a smoothing parameter, and $\{ui\}$ are i.i.d. random directions drawn from a uniform distribution over a unit sphere [10, 15, 16]. In general, RandGradEst is a biased approximation to the true gradient $\nabla f_i(x)$, and its bias reduces as μ approaches zero. However, in a practical system, if μ is too small, then the function difference could be dominated by the system noise and fails to represent the function differential [7]. To obtain better estimated gradient, we can use the Coordinate Smoothing Gradient Estimator (CooSGE) (Gu, Huo, and Huang, 2016; Gu et al., 2018; Liu et al., 2018c) to estimate the gradients as follows:

$$\hat{\nabla} f_i(x) = \sum_{j=1}^{d}\frac{f_i(x+\mu_j e_j)-f_i(x-\mu_j e_j)}{2\mu_j}e_j, \qquad i\in[n], \tag{7}$$

where μ_j is a coordinate-wise smoothing parameter, and e_j is a standard basis vector with 1 at its j-th coordinate, and 0 otherwise. Although the CooSGE need more function queries than the GauSGE, it can get better estimated gradient, and even can make the algorithms to obtain a faster convergence rate. Given an individual cost function f i (or an arbitrary function under A1 and A2), a two-point random gradient estimator $\hat{\nabla} f_i(x)$ is defined [12+1,16]

$$\hat{\nabla} f_i(x) = \frac{d(f_i(x+\mu u_i)-f_i(x))}{\mu}u_i, \qquad i\in[n], \tag{8}$$

where recall that d is the number of optimization variables, $\mu > 0$ is a smoothing parameter, and $\{ui\}$ are i.i.d. random directions drawn from a uniform distribution over a unit sphere [10, 15, 16]. In general, RandGradEst is a biased approximation to the true gradient $\nabla f_i(x)$, and its bias reduces as μ approaches zero. However, in a practical

system, if μ is too small, then the function difference could be dominated by the system noise and fails to represent the function differential [7].

In addition to RandGradEst and Avg-RandGradEst, the work [7, 26, 27] considered a coordinate-wise gradient estimator. Here every partial derivative is estimated via the two-point querying scheme under fixed direction vectors,

$$\hat{\nabla} f_i(x) = \sum_{j=1}^{d} \frac{f_i(x+\mu_j e_j) - f_i(x-\mu_j e_j)}{2\mu_j} e_j, \qquad i \in [n], \tag{9}$$

where $\mu_j > 0$ is a coordinate-wise smoothing parameter, and $e_j \in \mathbb{R}^d$ is a standard basis vector with 1 at its jth coordinate and 0s elsewhere. Compared to RandGradEst, CoordGradEst is deterministic and requires d times more function queries. However, as will be evident later, it yields an improved iteration complexity (i.e., convergence rate). More details on ZO gradient estimation can be found in Appendix A.1. In this work we only consider CoordGradEst and extension to RandGradEst si straightforward.

The disadvantage of CoordGradEst is the need of d times more function queries than RandGradEst in gradient estimation.

7.2 Accelerated Proximal Gradient Method

Finally, based on these estimated gradients, we give a zeroth-order proximal gradient descent (ZO-ProxGD) method, which performs the following iteration:

$$x_t^s = \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta \nabla f(x_{t-1}^s)), \qquad t = 1, 2, \ldots \tag{10}$$

where $\hat{\nabla} = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$.

7.3 reason to avoid full gradient calculation

Since ZO-ProxGD needs to estimate full gradient $\hat{\nabla} = \frac{1}{n}\sum_{i=1}^{n} f_i(x)$ when n is large in the problem (2.5), its high cost per iteration is prohibitive. As a result, Ghadimi, Lan, and Zhang (2016) proposed the RSPGF with calculating the gradient on the mini-batch $\mathcal{I}_t$

8 New Methods

In this section, to efficiently solve the large-scale nonconvex nonsmooth problems, we propose a class of faster zeroth-order proximal stochastic methods with the variance reduction (VR) techniques of SVRG and SAGA, respectively.

9 ZO-ProxSVRG

In the subsection, we propose the zeroth-order proximal SVRG (ZO-ProxSVRG) method by using VR technique of SVRG in (Xiao and Zhang, 2014; Reddi et al., 2016). It has

been shown in [19, 20] that the first-order SVRG achieves the convergence rate $O(1/T)$, yielding $O(\sqrt{T})$ less iterations than the ordinary SGD for solving finite sum problems. The corresponding algorithmic framework is described in Algorithm 1, where we use a mixture stochastic gradient $\hat{v}_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s$. The key step of SVRG 3 (Algorithm 1) is to generate an auxiliary sequence $\hat{x}$ at which the full gradient is used as a reference in building a modified stochastic gradient estimate

$$v_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s \tag{11}$$

where v_{t-1}^s denotes the gradient estimate at x. The key property of (11) is that g is an unbiased gradient estimate of $\nabla f(x)$. The gradient blending (2) is also motivated by a variance reduced technique known as control variate [28–30]. In the ZO setting, the gradient blending (2) is approximated using only function values,

$$\hat{v}_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s \tag{12}$$

where $\hat{g}^s = \sum_{i\in I_B}\hat{\nabla} f_i(\tilde{x}^{s-1})$ and $\hat{\nabla} f_i$ is a ZO gradient estimate specified by RandGradEst, AvgRandGradEst or CoordGradEst. Note that, $\mathbb{E}_{I_b}[\hat{v}_{t-1}^s] = \hat{\nabla} f(x_{t-1}^s) \neq \nabla f(x_{t-1}^s)$, i.e., this stochastic gradient is a biased estimate of the true full gradient. That is, the unbiased assumption on gradient estimates used in SVRG no longer holds. We highlight that although ZO-SVRG is similar to SVRG except the use of ZO gradient estimators to estimate batch, mini-batch, as well as blended gradients, this seemingly minor difference yields an essential difficulty in the analysis of ZO-SVRG. Although the SVRG has shown a great promise, it relies upon the assumption that the stochastic gradient is an unbiased estimate of the true full gradient. Thus, adapting the similar ideas of SVRG to zeroth-order optimization is not a trivial task. Thus, a careful analysis of ZO-SVRG is much needed. To address this issue, we analyze the upper bound for the variance of the estimated gradient $\hat{v}_t^s$, and choose the appropriate step size η and smoothing parameter μ to control this variance, which will be in detail discussed in the below theorems.

Replacing (2) with (3) in SVRG (Algorithm **??**) leads to a new ZO algorithm, which we call ZO-SVRG (Algorithm 2).

10 Convergence Analysis

First, we give some mild assumptions regarding problem (2.5) as follows:

Assumption 10.1. For $\forall i \in 1, 2, \ldots, n$, gradient of the function f_i is Lipschitz continuous with a Lipschitz constant $L > 0$, such that

$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|, \ \forall x, y \in \mathbb{R}^d.$$

Assumption 10.2. For $\forall x$, $\mathbb{E}\left[\left\|\hat{\nabla} f_i(x) - \hat{\nabla} f(x)\right\|^2\right] \leq \sigma^2$, where $\sigma > 0$ is a constant and $\hat{\nabla} f_i(x)$ is a CoodSGE gradient estimator of $\nabla f_i(x)$.

Both A1 and A2 are the standard assumptions used in nonconvex optimization literature [7, 13, 23– 26]. The first assumption is used for the convergence analysis of the zeroth-order algorithms (Ghadimi, Lan, and Zhang, 2016; Nesterov and Spokoiny, 2017; Liu et al., 2018c).

The second assumption gives the bounded variance of zeroth-order gradient estimates and are used in first-order optimization literature (Lian et al., 2016; Liu et al., 2018c,a), and due to that we need to analyze more complex problem (**??**) including a non-smooth part. Note that assumption **??** is milder than the assumption of bounded gradients [5, 25]. Such an assumption is necessary if one wants the convergence result to be independent of n. We start by deriving an upper bounds for the variance of estimated gradient $\hat{v}_{t-1}^s$ based on the CoodSGE.

Lemma 10.3. *Using CoodSGE given the mixture estimated gradient* $\hat{v}_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s$ *with* $\hat{g}^s = \frac{1}{B}\sum_{j\in I_B}\hat{\nabla} f_j(\widetilde{x}^{s-1})$*, then the following inequality holds.*

$$\begin{aligned}\mathbb{E}\left[\eta\left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\|^2\right] \leq & \frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2\right] \\ & + 2\frac{I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2 d^2\mu^2}{2}\end{aligned} \tag{13}$$

Proof. We have

$$\begin{aligned}
&\mathbb{E}\left[\eta\left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\|^2\right] \\
=&\mathbb{E}\left[\eta\left\|\frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\widetilde{x}^{s-1})\right) - \left(\nabla f(x_{t-1}^s) - \hat{g}^s\right)\right\|^2\right] \\
=&\mathbb{E}\left[\eta\left\|\frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\widetilde{x}^{s-1})\right) - \left(\nabla f(x_{t-1}^s) - \frac{1}{B}\sum_{j\in I_B}\hat{\nabla} f_j(\widetilde{x}^{s-1})\right)\right\|^2\right] \\
=&\mathbb{E}\left[\eta\left\|\frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\widetilde{x}^{s-1})\right) - \left(\nabla f(x_{t-1}^s) - \hat{\nabla} f(\tilde{x}^{s-1})\right) + \left(\frac{1}{B}\sum_{j\in I_B}\hat{\nabla} f_j(\widetilde{x}^{s-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right\|^2\right] \\
=&\eta\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i\in I_b}\left(\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\widetilde{x}^{s-1})\right) - \left(\nabla f(x_{t-1}^s) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right) + \frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla} f_j(\widetilde{x}^{s-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right\|^2\right] \\
\leq&2\eta\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i\in I_b}\left(\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\widetilde{x}^{s-1})\right) - \left(\hat{\nabla} f(x_{t-1}^s) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right) + \frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla} f_j(\widetilde{x}^{s-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right\|^2\right] \\
&+ 2\eta\mathbb{E}\left\|\hat{\nabla} f(x_{t-1}^s) - \nabla f(x_{t-1}^s)\right\|^2 \qquad (14) \\
=&2\eta\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i\in I_b}\left(\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\widetilde{x}^{s-1})\right) - \left(\hat{\nabla} f(x_{t-1}^s) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right)\right\|^2\right]
\end{aligned}$$

$$+2\eta\mathbb{E}\left[\left\|\frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla}f_j(\widetilde{x}^{s-1})-\hat{\nabla}f(\tilde{x}^{s-1})\right)\right\|^2\right]+2\eta\mathbb{E}\left\|\hat{\nabla}f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{15}$$

$$=\frac{2\eta}{b^2}\mathbb{E}\left[\sum_{i\in I_b}\left\|\left(\left(\hat{\nabla}f_i(x_{t-1}^s)-\hat{\nabla}f_i(\widetilde{x}^{s-1})\right)-\left(\hat{\nabla}f(x_{t-1}^s)-\hat{\nabla}f(\tilde{x}^{s-1})\right)\right)\right\|^2\right]$$

$$+2\eta\mathbb{E}\left[\left\|\frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla}f_j(\widetilde{x}^{s-1})-\hat{\nabla}f(\tilde{x}^{s-1})\right)\right\|^2\right]+2\eta\mathbb{E}\left\|\hat{\nabla}f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{16}$$

$$\leq\frac{2\eta}{b^2}\mathbb{E}\left[\sum_{i\in I_b}\left\|\hat{\nabla}f_i(x_{t-1}^s)-\hat{\nabla}f_i(\widetilde{x}^{s-1})\right\|^2\right]+2\eta\mathbb{E}\left[\left\|\frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla}f_j(\widetilde{x}^{s-1})-\hat{\nabla}f(\tilde{x}^{s-1})\right)\right\|^2\right]$$

$$+2\eta\mathbb{E}\left\|\hat{\nabla}f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{17}$$

$$\leq\frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2\right]+2\eta\mathbb{E}\left[\left\|\frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla}f_j(\widetilde{x}^{s-1})-\hat{\nabla}f(\tilde{x}^{s-1})\right)\right\|^2\right]$$

$$+2\eta\mathbb{E}\left\|\hat{\nabla}f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{18}$$

$$\leq\frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2\right]+2\frac{I\{B<n\}\eta\sigma^2}{B}+2\eta\mathbb{E}\left\|\hat{\nabla}f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{19}$$

$$\leq\frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2\right]+2\frac{I\{B<n\}\eta\sigma^2}{B}+\eta\frac{L^2d^2\mu^2}{2} \tag{20}$$

where, recalling that a deterministic gradient estimator is used, the expectations are taking with respect to I_b and I_B. The inequality (14) holds by the Jensen's inequality. (15) and (16) are based on $\mathbb{E}[\|x_1+x_2+\ldots+x_k\|^2]=\sum_{i=1}^k\mathbb{E}[\|x_i\|^2]$ if $x_1,x_2,\ldots,x_k$ are independent and of mean zero (note that I_b and I_B are also independent). (17) uses the fact that $\mathbb{E}[\|x-\mathbb{E}[x]\|^2]\leq\mathbb{E}[\|x\|^2]$, for any random variable x. (18) holds due to the following inequality

$$\begin{aligned}\mathbb{E}\left\|\hat{\nabla}f_i(x_t^s)-\hat{\nabla}f_i(\tilde{x}^s)\right\|^2&=\mathbb{E}\left\|\sum_{j=1}^d\frac{f_{i,\mu_j}(x_t^s)}{\partial x_j}e_j-\frac{f_{i,\mu_j}(\tilde{x}^s)}{\partial x_j}e_j\right\|^2\\&\leq d\sum_{j=1}^d\mathbb{E}\left\|\frac{f_{i,\mu_j}(x_t^s)}{\partial x_j}-\frac{f_{i,\mu_j}(\tilde{x}^s)}{\partial x_j}\right\|^2\\&\leq L^2d\sum_{j=1}^d\mathbb{E}\left\|x_{t,j}^s-\tilde{x}_j^s\right\|^2=L^2d\left\|x_t^s-\tilde{x}^s\right\|^2\end{aligned} \tag{21}$$

where the last inequality used the fact that f_{i,μ_j} is L-smooth. (19) is by Assumption 10.2 and (20) uses Lemma 13.2. The proof is now complete. □

Lemma 1 shows that variance of $\hat{v}_{t-1}^s$ has an upper bound. As the number of iterations increases, based on convergence analysis both x_{t-1}^s and $\widetilde{x}^{s-1}$ will approach

the same stationary point x^*, then the variance of stochastic gradient decreases, but does not vanishes, due to using the zeroth-order estimated gradient and variance with respect to the full gradient.

In the sequel we frequently use the following inequality

$$\|x-z\|^2 \leq (1+\frac{1}{\beta})\|x-y\|^2 + (1+\beta)\|y-z\|^2, \forall \beta > 0 \tag{22}$$

10.1 Gradient Mapping

For convex problems, one typically uses the optimality gap $F(x) - F(x^*)$ as the convergence criterion (see e.g., [Nesterov, 2004]). But for general nonconvex problems, one typically uses the gradient norm as the convergence criterion. E.g., for smooth nonconvex problems (i.e., $h(x) = 0$), Ghadimi and Lan [2013], Reddi et al. [2016a] and Lei et al. [2017] used $\|\nabla F(x)\|^2$ (i.e., $\|\nabla f(x)\|^2$) to measure the convergence results. In order to analyze the convergence results for nonsmooth nonconvex problems, we need to define the gradient mapping as follows (as in[Ghadimi et al., 2016, Reddi et al., 2016b]):

$$g_\eta = \frac{1}{\eta}(x - \text{Prox}_{\eta,h}(x - \eta\nabla f(x))) \tag{23}$$

Note that if $h(x)$ is a constant function (in particular, zero), this gradient mapping reduces to the ordinary gradient: $g_\eta = \nabla F(x) = \nabla f(x)$. In this paper, we use the gradient mapping g_η as the convergence criterion (same as [Ghadimi et al., 2016, Reddi et al., 2016b])(Parikh, Boyd, and others, 2014). For the nonconvex problems, if g_η, the point x is a critical point (Parikh, Boyd, and others, 2014). Thus, we can use the following definition as the convergence metric.

10.2 Convergence

In Theorem 10.4, we focus on the effect of CoodSGE on the convergence rate of ZO-PSVRG++ and give some remarks.

Theorem 10.4. *Suppose A1 and A2 hold, and the coordinate gradient estimator CoodSGE is used. The output $\hat{x}$ of Algorithm 7 satisfies*

$$\mathbb{E}[\left\|\mathcal{G}_\eta(\hat{x})\right\|^2] \leq \frac{6(F(x_0) - F(x^*))}{\eta S m} + \frac{I\{B < n\}12\sigma^2}{B} + 3L^2 d^2 \mu^2 \tag{24}$$

where $\eta = \min\{\frac{1}{8L}, \frac{\sqrt{b}}{6mL\sqrt{d}}\}$ denotes the step size and x^* denotes the optimal value of problem 2.5.

Proof. Now, we apply Lemma 13.3 to prove Theorem 10.4. Let $x_t^s = \text{Prox}_{\eta h}(x_{t-1}^s - \eta\hat{v}_{t-1}^s)$ and $\overline{x}_t^s = \text{Prox}_{\eta h}(x_{t-1}^s - \eta\nabla f(x_{t-1}^s))$. By letting $x^+ = x_t^s$, $x = x_{t-1}^s$, $v = \hat{v}_{t-1}^s$ and $z = \overline{x}_t^s$ in (100), we have

$$F(x_t^s) \leq F(\overline{x}_t^s) + \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle - \frac{1}{\eta}\left\langle x_t^s - x_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle$$

$$+\frac{L}{2}\left\|x_t^s - x_{t-1}^s\right\|^2 + \frac{L}{2}\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2. \tag{25}$$

Besides, by letting $x^+ = \overline{x}_t^s$, $x = x_{t-1}^s$, $v = \nabla f(x_{t-1}^s)$ and $z = x = x_{t-1}^s$ in (100), we have

$$\begin{aligned} F(\overline{x}_t^s) \leq & F(x_{t-1}^s) - \frac{1}{\eta}\left\langle \overline{x}_t^s - x_{t-1}^s, \overline{x}_t^s - x_{t-1}^s \right\rangle + \frac{L}{2}\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 \\ = & \Phi(x_{t-1}^s) - (\frac{1}{\eta} - \frac{L}{2})\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2. \end{aligned} \tag{26}$$

Combining (25) and (26) we have

$$\begin{aligned} F(x_t^s) \leq & F(x_{t-1}^s) + \frac{L}{2}\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 + \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \\ & - \frac{1}{\eta}\left\langle x_t^s - x_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \\ = & F(x_{t-1}^s) + \frac{L}{2}\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 + \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \\ & - \frac{1}{2\eta}\left(\left\|x_t^s - x_{t-1}^s\right\|^2 + \|x_t^s - \overline{x}_t^s\|^2 - \left\|\overline{x}_t^s - x_{t-1}^s\right\|^2\right) \\ = & F(x_{t-1}^s) - (\frac{1}{2\eta} - \frac{L}{2})\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{2\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 + \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \\ & - \frac{1}{2\eta}\|x_t^s - \overline{x}_t^s\|^2 \\ \leq & F(x_{t-1}^s) - (\frac{1}{2\eta} - \frac{L}{2})\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{2\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 + \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \\ & - \frac{1}{8\eta}\left\|x_t^s - x_{t-1}^s\right\|^2 + \frac{1}{6\eta}\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 \qquad (27) \\ = & F(x_{t-1}^s) - (\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{3\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 + \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \\ \leq & F(x_{t-1}^s) - (\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{3\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 + \eta\left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\|^2 \qquad (28) \end{aligned}$$

where the second inequality uses (22) with $\beta = 3$ and the last inequality holds due to the Lemma 13.4.

Note that $x_t^s = \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta \hat{v}_{t-1}^s)$ is the iterated from in our algorithm. By taking the expectation with respect to all random variables in (28) we obtain

$$\mathbb{E}[F(x_t^s)] \leq \mathbb{E}\left[F(x_{t-1}^s) - (\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{3\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 + \eta\left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\|^2\right] \tag{29}$$

In (29), we further bound $\eta\left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\|^2$ using Lemma 10.3 to obtain

$\mathbb{E}[F(x_t^s)]$

$$
\begin{aligned}
&\leq \mathbb{E}\left[F(x_{t-1}^s) - (\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{1}{3\eta} - L\right)\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2\right] \\
&\quad + \frac{2\eta L^2 d}{b}\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 + \frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2 d^2\mu^2}{2} \\
&= \mathbb{E}\left[F(x_{t-1}^s) - (\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - x_{t-1}^s\right\|^2 - \left(\frac{\eta}{3} - L\eta^2\right)\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&\quad + \frac{2\eta L^2 d}{b}\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 + \frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2 d^2\mu^2}{2} \qquad (30) \\
&\leq \mathbb{E}\left[F(x_{t-1}^s) - \frac{1}{2t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 - \left(\frac{\eta}{3} - L\eta^2\right)\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&\quad + (\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 + \frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2 d^2\mu^2}{2} \qquad (31)
\end{aligned}
$$

where recalling $\overline{x}_t^s := \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta\nabla f(x_{t-1}^s))$, (30) is based on the definition of gradient mapping $\mathcal{G}_\eta(x_{t-1}^s)$. (31) uses (22) by choosing $\beta = 2t-1$.

Taking a telescopic sum for $t = 1, 2, \ldots, m$ in epoch s from (31) and recalling that $x_m^s = \tilde{x}^s$ and $x_0^s = \tilde{x}^{s-1}$, we obtain

$$
\begin{aligned}
&\mathbb{E}[F(\tilde{x}^s)] \\
&\leq \mathbb{E}\left[F(\tilde{x}^{s-1}) - \sum_{t=1}^m \frac{1}{2t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 - \left(\frac{\eta}{3} - L\eta^2\right)\sum_{t=1}^m\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&\quad + \sum_{t=1}^m (\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 \\
&\quad + \sum_{t=1}^m \frac{2I\{B<n\}\eta\sigma^2}{B} + \sum_{t=1}^m \eta\frac{L^2 d^2\mu^2}{2} \\
&\leq \mathbb{E}\left[F(\tilde{x}^{s-1}) - \sum_{t=1}^{m-1} \frac{1}{2t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 - \left(\frac{\eta}{3} - L\eta^2\right)\sum_{t=1}^m\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&\quad + \sum_{t=2}^m (\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 \\
&\quad + \sum_{t=1}^m \frac{2I\{B<n\}\eta\sigma^2}{B} + \sum_{t=1}^m \eta\frac{L^2 d^2\mu^2}{2} \qquad (32) \\
&= \mathbb{E}\left[F(\tilde{x}^{s-1}) - \left(\frac{\eta}{3} - L\eta^2\right)\sum_{t=1}^m\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&\quad - \sum_{t=1}^{m-1}\left((\frac{1}{2t} - \frac{1}{2t+1})(\frac{5}{8\eta} - \frac{L}{2}) - \frac{2\eta L^2 d}{b}\right)\mathbb{E}\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 \\
&\quad + \sum_{t=1}^m \frac{2I\{B<n\}\eta\sigma^2}{B} + \sum_{t=1}^m \eta\frac{L^2 d^2\mu^2}{2}
\end{aligned}
$$

$$
\begin{aligned}
&\leq \mathbb{E}\left[F(\tilde{x}^{s-1}) - \left(\frac{\eta}{3} - L\eta^2\right)\sum_{t=1}^{m}\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&\quad - \sum_{t=1}^{m-1}\left(\frac{1}{6t^2}\left(\frac{5}{8\eta} - \frac{L}{2}\right) - \frac{2\eta L^2 d}{b}\right)\mathbb{E}\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 \\
&\quad + \sum_{t=1}^{m}\frac{2I\{B<n\}\eta\sigma^2}{B} + \sum_{t=1}^{m}\eta\frac{L^2d^2\mu^2}{2} \\
&\leq \mathbb{E}\left[F(\tilde{x}^{s-1}) - \frac{\eta}{6}\sum_{t=1}^{m}\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] + \sum_{t=1}^{m}\frac{2I\{B<n\}\eta\sigma^2}{B} + \sum_{t=1}^{m}\eta\frac{L^2d^2\mu^2}{2} \qquad (33)
\end{aligned}
$$

where (32) holds since norm is always non-negative and $x_0^s = \tilde{x}^{s-1}$. In (33) we have used the fact that $\left(\frac{1}{6t^2}\left(\frac{5}{8\eta} - \frac{L}{2}\right) - \frac{2\eta L^2 d}{b}\right) \geq 0$ for all $1 \leq t \leq m$ and $\frac{\eta}{5} \leq \frac{\eta}{3} - L\eta^2$ since $\eta = \min\{\frac{1}{8L}, \frac{\sqrt{b}}{6mL\sqrt{d}}\}$. Telescoping the sum for $s = 1, 2, \ldots, S$ in (33), we obtain

$$
\begin{aligned}
&0 \leq \mathbb{E}[F(\tilde{x}^S) - F(x^*)] \\
&\leq \mathbb{E}\left[F(\tilde{x}^0) - F(x^*) - \sum_{s=1}^{S}\sum_{t=1}^{m}\frac{\eta}{6}\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2 + \sum_{s=1}^{S}\sum_{t=1}^{m}\left(\frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2d^2\mu^2}{2}\right)\right]
\end{aligned}
$$

Thus, we have

$$
\mathbb{E}[\left\|\mathcal{G}_\eta(\hat{x})\right\|^2] \leq \frac{6\left(F(x_0) - F(x^*)\right)}{\eta S m} + \frac{I\{B<n\}12\sigma^2}{B} + 3L^2d^2\mu^2 \qquad (34)
$$

where (34) holds since we choose $\hat{x}$ uniformly randomly from $\{x_{t-1}^s\}_{t\in[m], s\in[S]}$. □

The proof for Theorem 10.4 is notably different from that of ProxSVRG [] as they used a Lyapunov function to show that the accumulated gradient mapping decreases with epoch s. In our proof, we directly show that $F(x^s)$ decreases by using a different analysis. This is made possible by tightening the inequalities using Young's inequality and Lemma 10.3 which yields a much simpler analysis for our ZO-PSVRG+ compared with ZO-SVRG-Coord, ZO-ProxSVRG and ZO-ProxSAGA. Also, our convergence result holds for any minibatch size and any epoch size m unlike ZO-SVRG-Coord which holds true only for specific values of m with an involved parameter setting. We also avoid the computation of the full gradient at the beginning of each epoch, i.e., $B \neq n$. (24) shows that a large batch size B indeed reduces the variance of estimated full gradient and improves the convergence of ZO-PSVRG+.

Compared to the convergence rate of SVRG as given in [20, Theorem 2], Theorem 1 exhibits two additional errors $\frac{I\{B<n\}\sigma^2}{B}$ and $O(L^2d^2\mu^2)$ due to the use of SZO gradient estimates and $B < n$ in full gradient estimation, respectively. The error due to $B < n$ is eliminated only when $B = n$. Roughly speaking, if we choose the smoothing parameter μ reasonably small, and the batch size B reasonably large, then the error (**??**) would reduce, leading to non-dominant effect on the convergence rate of ZO-PSVRG+.

If $B = n$, ZO-PSVRG+ reduces to ZO-ProxSVRG since Step 7 of Algorithm 2 becomes $\frac{1}{B}\sum_{i\in I_B}\nabla f_i(\widetilde{x}_{t-1}^s) = \nabla f(\widetilde{x}_{t-1}^s)$. Note that the stepsize η is involved, relying

on the epoch length m, the minibatch size b , and the number of optimization variables d.

In order to acquire explicit dependence on these parameters and to explore deeper insights of convergence, with the aid of Theorem 10.4, Corollary 10.5 provides the convergence rate of ZO-PSVRG+ in terms precision at the solution $\hat{x}$ and simplifies (24) for a specific parameter setting, as formalized below.

Corollary 10.5. *We set the batch size $B = \min\{12\sigma^2/\epsilon, n\}$ and the smoothing parameter $\mu \leq \frac{\sqrt{\epsilon}}{3\sqrt{dL}}$. Suppose $\hat{x}$ returned by Algorithm 7 is an ϵ- accurate solution for problem (2.5). Recalling that CoodSGE require $O(d)$ function queries, the number of SZO calls is at most*

$$d(SB + Smb) = 6d\left(\Phi(x_0) - \Phi(x^*)\right)\left(\frac{B}{\epsilon\eta m} + \frac{b}{\epsilon\eta}\right) = O\left(\frac{Bd}{\epsilon\eta m} + \frac{bd}{\epsilon\eta}\right). \tag{35}$$

and the number of PO calls is equal to $T = Sm = \frac{6(\Phi(x_0)-\Phi(x^*))}{\epsilon\eta} = O\left(\frac{1}{\epsilon\eta}\right)$. In particular, by setting $m = \sqrt{b}$ and $\eta = \frac{1}{6L\sqrt{d}}$, the number of ZO calls is at most

$$36dL(\Phi(x_0) - \Phi(x^*))\left(\frac{B\sqrt{d}}{\epsilon\sqrt{b}} + \frac{b\sqrt{d}}{\epsilon}\right) = O\left(s_n\frac{d\sqrt{d}}{\epsilon\sqrt{b}} + \frac{bd\sqrt{d}}{\epsilon}\right). \tag{36}$$

where $s_n = \min\{n, \frac{1}{\epsilon}\}$. The number of PO calls is equal to $T = Sm = S\sqrt{b} = \frac{6\sqrt{d}(\Phi(x_0)-\Phi(x^*))}{\epsilon} = O\left(\frac{\sqrt{d}}{\epsilon}\right)$.

Proof. Using Theorem 10.4 we have $\eta = \min\{\frac{1}{8L}, \frac{\sqrt{b}}{6mL\sqrt{d}}\}$

$$\mathbb{E}[\left\|\mathcal{G}_\eta(\hat{x})\right\|^2] \leq \frac{6\left(\Phi(x_0) - \Phi(x^*)\right)}{\eta Sm} + \frac{I\{B < n\}12\sigma^2}{B} + 3L^2d^2\mu^2 = 3\epsilon \tag{37}$$

Now we obtain the total number of iterations $T = Sm = S\sqrt{b} = \frac{36L(\Phi(x_0)-\Phi(x^*))}{\epsilon}$ $T = Sm = \frac{6(\Phi(x_0)-\Phi(x^*))}{\epsilon\eta}$. Since $\mu \leq \frac{\sqrt{\epsilon}}{3\sqrt{dL}}$, if $B = n$, the second term in the bound (37) is 0 and the proof is finished as the number of SFO call equals to $Sn + Smb = 36L\left(\Phi(x_0) - \Phi(x^*)\right)\left(\frac{n}{\epsilon\sqrt{b}} + \frac{b}{\epsilon}\right)$ $Sn + Smb = 6\left(\Phi(x_0) - \Phi(x^*)\right)\left(\frac{n}{\epsilon\eta m} + \frac{b}{\epsilon\eta}\right)$. If $B < n$ the number of SZO calls equal to $d(SB + Smb) = 6d\left(\Phi(x_0) - \Phi(x^*)\right)\left(\frac{B}{\epsilon\eta m} + \frac{b}{\epsilon\eta}\right)$(note that $\frac{I\{B<n\}12\sigma^2}{B} \leq \epsilon$ since $B \geq 12\sigma^2/\epsilon$). The second part of corollary is obtained by setting $m = \sqrt{b}$ in the first part. □

Roughly speaking, Corollary 10.5 shows that if we choose the smoothing parameter μ reasonably small and the batch size B sufficiently large, then the error induced by these terms would reduce, leading to non-dominant effect on the convergence rate of ZO-PSVRG+. The error term inherited by batch size is eliminated only when $B = n$ (i.e., $I\{B < n\} = 0$). In this case, ZO-PSVRG+ reduces to ZO-ProxSVRG since Step 7 of Algorithm 7 becomes $\hat{g}^s = \hat{\nabla} f(\widetilde{x}^{s-1})$.

If the smoothing parameter and batch-size are selected appropriately, then we obtain the error term $O(\sqrt{d}/T)$, with is better than the convergence rate of competitor ZO methods (Table 6) by factor of $\frac{1}{\sqrt{d}}$. Moreover, ZO-PSVRG+ uses much less SZO oracle which is indicated in Table 6.

It is worth mentioning that the condition on the value of step size in Theorem 10.4 is less restrictive than several SZO algorithms. For example, ZO-SVRG-Coord required $\eta = O(\frac{1}{d})$ which is smaller by a factor of $\sqrt{d}$ than ours.

On the other hand, the condition on the value of smoothing parameter μ in Corollary 10.5 is more restrictive than several SZO algorithms. For instance, ZO-ProxSVRG required $\mu = O(\frac{1}{\sqrt{d}})$ with a stepsize η which scales by $\frac{1}{d}$.

It is noted from equation (36), if $b\sqrt{b} = O(B)$, the SZO complexity is increased by a factor $\sqrt{b}$, which is smaller than the size of the mini-batch. However, the corresponding complexity of RGF and RSG will be increased by multiplying a factor of b (see Table 6), so our algorithm has a better dependency to the mini-batch size in this special case.

Our work and reference [] show that a large batch B for $B \neq n$ indeed reduces the error inherited by variance and improves the convergence of ZO optimization methods.

11 Convergence Under PL Condition

In this section, we provide the global linear convergence rate for nonconvex functions under the Polyak-Łojasiewicz (PL) condition [Polyak, 1963].

In this section, we provide the global linear convergence rate for nonconvex functions under the Polyak-Łojasiewicz (PL) condition [Polyak, 1963]. The original form of PL condition is

$$\exists \lambda > 0, \text{such that} \|\nabla f(x)\|^2 \geq 2\lambda(f(x) - f^*), \forall x, \tag{38}$$

where f^* denotes the (global) optimal function value. It is worth noting that f satisfies PL condition when f is μ-strongly convex. This condition specifies how fast the objective function grows in a local neighborhood of optimal solutions. We show the iteration complexity of ZO-SVRG (Algorithm 2) im improved by applying PL condition .

In particular, we propose a generic convergence frame- work for accelerating existing SZCO algorithms in various settings by leveraging the local error bound condition. This is accomplished by a novel synthesis of existing SZCO algorithms.

Due to the nonsmooth term $h(x)$ in problem (**??**), we use the gradient mapping to define a more general form of PL condition as follows

$$\exists \lambda > 0, \text{such that} \left\|G_\eta(x)\right\|^2 \geq 2\lambda(\Phi(x) - \Phi^*), \forall x. \tag{39}$$

Recall that if $h(x)$ is a constant function, the gradient mapping reduces to $G_\eta(x) = \nabla f(x)$. Note that the LEB condition has been studied thoroughly in [Yang and Lin, 2015; Bolte et al., 2015; Xu et al., 2017]. It is satisfied for a broad family of problems. For ex- ample, when $f(x)$ is continuous and semi-algebraic (or subanalytic), the LEB condition holds on any compact set [Bolte et al., 2015]. Below, we consider several

instances of problems that satisfy the LEB condition. More interesting exam- ples in machine learning can be found in [Yang and Lin, 2015; Xu et al., 2017].

We want to point out that [] used the following form of PL condition

$$\exists \lambda > 0, \text{such that} D_h(x,\alpha) \geq 2\lambda(\Phi(x) - \Phi^*),\ \forall x. \tag{40}$$

where $D_h(x,\alpha) := -2\alpha \min_y \{\langle \nabla f(x), y - x\rangle + \frac{\alpha}{2}\|y-x\|^2 + h(y) - h(x)\}$. Our PL condition is arguably more natural. We next study the effect of the coordinate-wise gradient estimator (CooGrad) on the convergence rate of ZO-PSVRG++ for functions with PL condition, as formalized in Theorem 11.1.

Similar to Theorem 1, we provide the convergence result of ProxSVRG+ (Algorithm 1) under PL-condition in the following Theorem 2. Note that under PL condition (i.e. (7) holds), ProxSVRG+ can directly use the final iteration $\widetilde{x}^S$ as the output point instead of the randomly chosen one $\hat{x}$. Similar to [Reddi et al., 2016b], we assume the condition number $L/\mu > n$ for simplicity.

Theorem 11.1. *Suppose A1 and A2 hold, and CooSGD is used in Algorithm 7 with step size* $\eta \leq \min\{\frac{1}{8L}, \frac{2\sqrt{\gamma b}}{10mL\sqrt{d}}\}$ *where* $\gamma = 1 - \frac{2\lambda\eta}{3}m - \frac{\lambda\eta}{3} > 0$. *Then*

$$\mathbb{E}[\Phi(\widetilde{x}^S) - \Phi^*] \leq \left(1 - \frac{\lambda\eta}{3}\right)^{Sm} \mathbb{E}[\Phi(\widetilde{x}^0) - \Phi^*] + \frac{6I\{B<n\}\sigma^2}{\lambda B} + \frac{3L^2d^2\mu^2}{2\lambda} \tag{41}$$

where x^* *is same as Theorem 10.4.*

Proof. We start by recalling inequality (30) from the proof of Theorem 10.4, i.e.,

$$\begin{aligned}
&\mathbb{E}[\Phi(x_t^s)] \\
\leq &\mathbb{E}\left[\Phi(x_{t-1}^s) - \frac{1}{2t}\left(\frac{5}{8\eta} - \frac{L}{2}\right)\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 - \left(\frac{\eta}{3} - L\eta^2\right)\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&+ \left(\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}\left(\frac{5}{8\eta} - \frac{L}{2}\right)\right)\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 + \frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2d^2\mu^2}{2} \\
\leq &\mathbb{E}\left[\Phi(x_{t-1}^s) - \frac{1}{2t}\left(\frac{5}{8\eta} - \frac{L}{2}\right)\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 - \frac{\eta}{6}\left\|\mathcal{G}_\eta(x_{t-1}^s)\right\|^2\right] \\
&+ \left(\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}\left(\frac{5}{8\eta} - \frac{L}{2}\right)\right)\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 + \frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2d^2\mu^2}{2}
\end{aligned} \tag{42}$$

where in (42) inequality we applied $\eta L \leq \frac{1}{6}$. Moreover, substituting PL inequality, i.e.,

$$\left\|G_\eta(x)\right\|^2 \geq 2\lambda(\Phi(x) - \Phi^*) \tag{43}$$

into (42), we obtain

$$\mathbb{E}[\Phi(x_t^s)]$$

$$\leq \mathbb{E}\left[\Phi(x_{t-1}^s) - \frac{1}{2t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 - \lambda\frac{\eta}{3}(\Phi(x_{t-1}^s) - \Phi^*)\right]$$
$$+ (\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 + \frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2d^2\mu^2}{2} \quad (44)$$

Thus, we have

$$\mathbb{E}[\Phi(x_t^s)]$$
$$\leq \mathbb{E}\left[(1-\lambda\frac{\eta}{3})(\Phi(x_{t-1}^s) - \Phi^*) - \frac{1}{2t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2\right]$$
$$+ (\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 + \frac{2I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2d^2\mu^2}{2} \quad (45)$$

Let $\alpha := 1 - \lambda\frac{\eta}{3}$ and $\Psi_t^s := \frac{\mathbb{E}[\Phi(x_t^s) - \Phi^*]}{\alpha^t}$. Combining these definitions with (45), we have

$$\Psi_t^s$$
$$\leq \Psi_{t-1}^s - \frac{1}{\alpha^t}\mathbb{E}\left[\frac{1}{2t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \widetilde{x}^{s-1}\right\|^2 - (\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2\right]$$
$$+ \frac{1}{\alpha^t}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1}{\alpha^t}\eta\frac{L^2d^2\mu^2}{2} \quad (46)$$

Similar to the proof of Theorem 10.4, summing (46) for $t = 1, 2, \ldots, m$ in epoch s and recalling that $x_m^s = \tilde{x}^s$ and $x_0^s = \tilde{x}^{s-1}$, we have

$$\mathbb{E}[\Phi(\tilde{x}^s) - \Phi^*]$$
$$\leq \alpha^m\mathbb{E}\left[(\Phi(\tilde{x}^{s-1}) - \Phi^*)\right] + \alpha^m\sum_{t=1}^m\frac{1}{\alpha^t}\frac{2I\{B<n\}\eta\sigma^2}{B} + \alpha^m\sum_{t=1}^m\frac{1}{\alpha^t}\eta\frac{L^2d^2\mu^2}{2}$$
$$-\alpha^m\mathbb{E}\left[\sum_{t=1}^m\frac{1}{2t\alpha^t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \tilde{x}^{s-1}\right\|^2 - \sum_{t=1}^m\frac{1}{\alpha^t}(\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\left\|x_{t-1}^s - \tilde{x}^{s-1}\right\|^2\right]$$
$$\leq \alpha^m\mathbb{E}\left[(\Phi(\tilde{x}^{s-1}) - \Phi^*)\right] + \frac{1-\alpha^m}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1-\alpha^m}{1-\alpha}\frac{\eta L^2d^2\mu^2}{2}$$
$$-\alpha^m\mathbb{E}\left[\sum_{t=1}^m\frac{1}{2t\alpha^t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \tilde{x}^{s-1}\right\|^2 - \sum_{t=1}^m\frac{1}{\alpha^t}(\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\left\|x_{t-1}^s - \tilde{x}^{s-1}\right\|^2\right]$$
$$\leq \alpha^m\mathbb{E}\left[(\Phi(\tilde{x}^{s-1}) - \Phi^*)\right] + \frac{1-\alpha^m}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1-\alpha^m}{1-\alpha}\frac{\eta L^2d^2\mu^2}{2}$$
$$-\alpha^m\mathbb{E}\left[\sum_{t=1}^{m-1}\frac{1}{2t\alpha^t}(\frac{5}{8\eta} - \frac{L}{2})\left\|x_t^s - \tilde{x}^{s-1}\right\|^2\right]$$
$$+\alpha^m\mathbb{E}\left[\sum_{t=2}^m\frac{1}{\alpha^t}(\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2}))\left\|x_{t-1}^s - \tilde{x}^{s-1}\right\|^2\right] \quad (47)$$
$$\leq \alpha^m\mathbb{E}\left[(\Phi(\tilde{x}^{s-1}) - \Phi^*)\right] + \frac{1-\alpha^m}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1-\alpha^m}{1-\alpha}\frac{\eta L^2d^2\mu^2}{2}$$

$$-\alpha^m \mathbb{E}\left[\sum_{t=1}^{m-1} \frac{1}{\alpha^{t+1}}\left((\frac{\alpha}{2t}-\frac{1}{2t+1})(\frac{5}{8\eta}-\frac{L}{2})-\frac{2\eta L^2 d}{b}\right)\left\|x_t^s-\tilde{x}^{s-1}\right\|^2\right]$$
$$\leq \alpha^m \mathbb{E}\left[(\Phi(\tilde{x}^{s-1})-\Phi^*)\right]+\frac{1-\alpha^m}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B}+\frac{1-\alpha^m}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2} \tag{48}$$

where (47) since $\|.\|^2$ always is non-negative and $x_0^s=\tilde{x}^{s-1}$. (48) holds since it is sufficient to show $(\frac{\alpha}{2t}-\frac{1}{2t+1})(\frac{5}{8\eta}-\frac{L}{2})-\frac{2\eta L^2 d}{b}\geq 0$, for all $t=1,2,\ldots,m$. We should have

$$\begin{aligned}(\frac{\alpha}{2t}-\frac{1}{2t+1})(\frac{5}{8\eta}-\frac{L}{2}) &\geq \frac{2\eta L^2 d}{b}\\ (\frac{1-2\beta t-\beta}{6t^2})(\frac{5}{8\eta}-\frac{L}{2}) &\geq \frac{2\eta L^2 d}{b}\\ (\frac{1-2\beta m-\beta}{6m^2})(\frac{5}{8\eta}-\frac{L}{2}) &\geq \frac{2\eta L^2 d}{b}\\ (\frac{5\gamma}{48m^2}) &\geq \frac{2\eta^2 L^2 d}{b}+\frac{L\eta\gamma}{12m^2}\end{aligned} \tag{49}$$

with $L\eta\leq\frac{1}{8}$, we have

$$\begin{aligned}(\frac{4\gamma}{100m^2}) &\geq \frac{\eta^2 L^2 d}{b}\\ \eta &\leq \frac{2\sqrt{\gamma b}}{10mL\sqrt{d}}\end{aligned} \tag{50}$$

It is easy to see that this inequality holds since $\eta\leq\min\{\frac{1}{8L},\frac{2\sqrt{\gamma b}}{10mL\sqrt{d}}\}$, where $\gamma=1-2\beta m-\beta>0$. Similarly, let $\tilde{\alpha}=\alpha^m$ and $\tilde{\Psi}^s=\frac{\mathbb{E}[\Phi(\tilde{x}^s)-\Phi^*]}{\tilde{\alpha}^s}$. Substituting these definitions into (48), we have

$$\widetilde{\Psi}^s\leq\widetilde{\Psi}^{s-1}+\frac{1}{\tilde{\alpha}^s}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{I\{B<n\}\eta\sigma^2}{B}+\frac{1}{\tilde{\alpha}^s}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{Ld\mu^2}{12} \tag{51}$$

Taking a telescopic sum from ((51) for all epochs $1\leq s\leq S$, we obtain

$$\begin{aligned}\mathbb{E}[\Phi(\widetilde{x}^S)-\Phi^*] &\leq \widetilde{\alpha}^S\mathbb{E}[\Phi(\widetilde{x}^0)-\Phi^*]+\widetilde{\alpha}^S\sum_{s=1}^{S}\frac{1}{\widetilde{\alpha}^s}\frac{1-\widetilde{\alpha}}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B}+\widetilde{\alpha}^S\sum_{s=1}^{S}\frac{1}{\widetilde{\alpha}^s}\frac{1-\widetilde{\alpha}}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}\\ &= \alpha^{Sm}\mathbb{E}[\Phi(\widetilde{x}^0)-\Phi^*]+\frac{1-\widetilde{\alpha}^S}{1-\widetilde{\alpha}}\frac{1-\widetilde{\alpha}}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B}+\frac{1-\widetilde{\alpha}^S}{1-\widetilde{\alpha}}\frac{1-\widetilde{\alpha}}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}\\ &\leq \alpha^{Sm}\mathbb{E}[\Phi(\widetilde{x}^0)-\Phi^*]+\frac{1}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B}+\frac{1}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}\\ &= \left(1-\frac{\lambda\eta}{3}\right)^{Sm}\mathbb{E}[\Phi(\widetilde{x}^0)-\Phi^*]+\frac{6I\{B<n\}\sigma^2}{\lambda B}+\frac{3L^2 d^2\mu^2}{2\lambda} \qquad (52)\\ &= \left(1-\frac{\lambda\eta}{3}\right)^{Sm}\mathbb{E}[\Phi(\widetilde{x}^0)-\Phi^*]+\frac{6I\{B<n\}\sigma^2}{\lambda B}+\frac{3L^2 d^2\mu^2}{2\lambda}=3\epsilon \qquad (53)\end{aligned}$$

where in (52) we recall that $\alpha = 1 - \frac{\lambda\eta}{3}$, and (53) uses $\eta \le \frac{1}{6Ld}$. □

Proposition 2 shows that compared to CoordGradEst, RandGradEst and Avg-RandGradEst involve an additional error term within a factor ?, respectively. Such an error is introduced by the second-order moment of gradient estimators using random direction samples [12+1, 14], and it decreases as the number of direction samples q increases. On the other hand, all gradient estimators have a common error bounded by $O(\mu^2 L^2 d^2)$, where let $\mu_l = \mu$ for $l \in [d]$ in CoordGradEst. If μ is specified as in (9), then we obtain the error term $O(d/T)$, consistent with the convergence rate of ZO-SVRG in Corollary 1.

By comparing with Theorem **??**, it can be seen from (**??**) that the use of PL condition amplifies the error $O(\frac{I\{B<n\}\sigma^2}{B} + \frac{L^2 d^2 \mu^2}{\lambda}$ through multiple $1/\lambda$. And the error induced by these terms ceases to be significantly improved for this term as $\lambda >> 1$.

Corollary 11.2. *Suppose the final iteration point $\tilde{x}^S$ in Algorithm **??** satisfies $\mathbb{E}[\Phi(\tilde{x}^S) - \Phi^*] \le \epsilon$ under PL condition. Under Assumption **??** and **??**, we let batch size $B = \min\{\frac{6\sigma^2}{\lambda\epsilon}, n\}$ and the smoothing parameter $\mu \le \frac{\epsilon}{2Ld\lambda}$. The number of ZO calls is bounded by*

$$d(SB + Smb) = O(\frac{s_n d}{\lambda\eta m}\log\frac{1}{\epsilon} + \frac{bd}{\lambda\eta}\log\frac{1}{\epsilon})$$

where $s_n = \min\{n, \frac{1}{\lambda\epsilon}\}$. The number of PO calls equals to the total number of iterations T which is bounded by

$$T = Sm = O(\frac{1}{\lambda\eta}\log\frac{1}{\epsilon})$$

3) In particular, given the setting $m = \sqrt{b}$ and $\eta = \frac{\sqrt{\gamma}}{5L\sqrt{d}}$, the number of ZO calls simplifies to $d(SB + Smb) = O(\frac{Bd\sqrt{d}}{\lambda\sqrt{\gamma}m}\log\frac{1}{\epsilon} + \frac{bd\sqrt{d}}{\lambda\sqrt{\gamma}}\log\frac{1}{\epsilon}$.

Proof. From Theorem 11.1, we have

$$\mathbb{E}[\Phi(\tilde{x}^S) - \Phi^*] \le \left(1 - \frac{\lambda\eta}{3}\right)^{Sm}\mathbb{E}[\Phi(\tilde{x}^0) - \Phi^*] + \frac{6I\{B<n\}\sigma^2}{\lambda B} + \frac{3L^2 d^2\mu^2}{2\lambda} = 3\epsilon \quad (54)$$

which gives the total number of iterations $T = Sm = S\sqrt{b} = O(\frac{1}{\lambda}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{1}{\lambda\eta}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{\sqrt{d}}{\lambda\sqrt{\gamma}}\log\frac{1}{\epsilon})$. The number of PO calls equals to $T = Sm = O(\frac{1}{\lambda}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{1}{\lambda\eta}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{\sqrt{d}}{\lambda\sqrt{\gamma}}\log\frac{1}{\epsilon})$. The number of SFO calls equals to $Sn + Smb = O(\frac{n}{\lambda\sqrt{b}}\log\frac{1}{\epsilon} + \frac{b}{\lambda}\log\frac{1}{\epsilon})$ if $B = n$, or equals to $Sn + Smb = O(\frac{B}{\lambda\sqrt{b}}\log\frac{1}{\epsilon} + \frac{b}{\lambda}\log\frac{1}{\epsilon})$ $SB + Smb = O(\frac{B}{\lambda\eta m}\log\frac{1}{\epsilon} + \frac{b}{\lambda\eta}\log\frac{1}{\epsilon})$ $SB + Smb = O(\frac{B\sqrt{d}}{\lambda\sqrt{\gamma}m}\log\frac{1}{\epsilon} + \frac{b\sqrt{d}}{\lambda\sqrt{\gamma}}\log\frac{1}{\epsilon})$if $B < n$(note that $\frac{I\{B<n\}3\sigma^2}{\lambda B} \le \epsilon$ since $B \ge 6\sigma^2/\lambda\epsilon$). $\mu \le \frac{\sqrt{2\lambda\epsilon}}{Ld}$ □

Corollary 11.2 shows that the use of PL condition improves the dominant convergence rate, where the error of order $O(d/\epsilon)$ in Corollary 10.5 improves to $O(\log(d/\epsilon))$.

By contrast with Corollary 1, it can be seen from (12) that the use of Avg-RandGradEst reduces the error ? in (10) through multiple (q) direction samples. And the convergence rate ceases to be significantly improved as ?. Our empirical results show that a moderate choice of q can significantly speed up the convergence of ZO-SVRG.

Compared to the aforementioned ZO algorithms [5, 14, 24], the convergence performance of ZO-SVRG in (**??**) has an improved (linear rather than sub-linear) dependence on $1/T$. However, it suffers an additional error of order $O(1/b)$ inherited from ? in (5), which is also a consequence of the last error term in (4). We recall from the definition of δ_n in Proposition 1 that if $b < n$ or samples in the mini-batch are chosen independently from $[n]$, then ?.

Remark 11.3. Compared to the convergence rate of SVRG as given in Theorem **??**, Theorem **??** exhibits additional parameter γ for parameter selection due to the use of PL condition. If we assume the condition number $\lambda/L \leq \sqrt{n}$ and choose $m = n^{1/2}$ and $\rho \leq \frac{1}{2}$, then the definition of γ yields

$$\begin{aligned}\gamma &= 1 - \frac{2\lambda\eta}{3}m - \frac{\lambda\eta}{3} \\ &\geq 1 - \frac{2\lambda\rho}{3L}m - \frac{\lambda\rho}{3L} \\ &\geq 1 - \frac{2\rho}{3\sqrt{n}}m - \frac{\rho}{3\sqrt{n}} \\ &\geq 1 - \rho \geq \frac{1}{2} \end{aligned} \tag{55}$$

According to Theorem 11.1, equation (55) implies $\eta \leq \min\{\frac{1}{8L}, \frac{\sqrt{\gamma b}}{10mL\sqrt{d}}\}$. Hence, choosing $b = n$ leads to the constant step size $\eta = \frac{1}{10L\sqrt{d}}$.

We show that ProxSVRG+ directly obtains a global linear convergence rate without restart by a nontrivial proof. Note that Reddi et al. [2016b] used PL-SVRG/SAGA to restart ProxSVRG/SAGA $O(log(1/\epsilon))$ times to obtain the linear convergence rate under PL condition. Moreover, similar to Table 2, if we choose $b = 1$ or n for ProxSVRG+, then its convergence result is $O(\log \frac{1}{\epsilon})$, which is the same as ProxGD [Karimi et al., 2016]. If we choose $b = n$ for ProxSVRG+, then the convergence result is $O()$, the same as the best result achieved by ProxSVRG/SAGA [Reddi et al., 2016b]. If we choose $b =$ for ProxSVRG+, then its convergence result is $O(\log \frac{1}{\epsilon})$ which generalizes the best result of SCSG [Lei et al., 2017] to the more general nonsmooth nonconvex case and is better than ProxGD and ProxSVRG/SAGA. Also note that our ProxSVRG+ uses much less proximal oracle calls than ProxSVRG/SAGA if $b < n^{2/3}$.

12 Strongly Convex with Momentum Acceleration

In the subsection, we propose the zeroth-order proximal SAGA (ZO-ProxSAGA) method via using VR technique of SAGA in (Defazio, Bach, and Lacoste-Julien, 2014; Reddi et al., 2016).

The corresponding algorithmic description is given in Algorithm **??**, where we use a mixture stochatic gradient $\hat{v}_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s$. Similarly, $\mathbb{E}_{I_b}[\hat{v}_{t-1}^s] = \hat{\nabla} f(x_{t-1}^s) \neq \nabla f(x_{t-1}^s)$, i.e., this stochastic gradient is a biased estimate of the true full gradient. Note that in Algorithm **??**, $\widetilde{x}^s$ =? which is computer in the step ?, to avoid unnecessary calculations. Next, we give the upper bounds for the variance of stochastic gradient $\hat{v}_{t-1}^s$ based on the CooSGE.

In this section, we improve the efficiency of ZO-PSVRG++ (Algorithm **??**) by using momentum acceleration. In Theorem 12, we show the effect of CooGrad on the convergence rate of ZO-SVRG. Similar to Theorem **??**, we analyze the convergence based on optimality gap. Next, based on the above lemma, we study the convergence

Algorithm 2 ZO-PROXSVRG for convex Optimization

Input: initial point x_0, batch size B, minibatch size b, epoch length m, step size η
Initialize: $\tilde{x}^0 = x_0$
for $s = 1, 2, \ldots, S$ **do**
  $x_0^s = z_0^s = \widetilde{x}^{s-1}$
  $\hat{g}^s = \frac{1}{B}\sum_{j\in I_B}\hat{\nabla} f_j(\widetilde{x}^{s-1})$
  for $t = 1, 2, \ldots, m$ **do**
    $\hat{v}_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s$
    $z_t^s = \mathrm{Prox}_{\frac{\eta}{\theta}h}(z_{t-1}^s - \frac{\eta}{\theta}\hat{v}_{t-1}^s)$
    $x_t^s = \theta z_t^s + (1-\theta)\widetilde{x}^{s-1}$
  $\tilde{x}^s = \frac{1}{m}\sum_{j=1}^m x_j^s$
Output: $\tilde{x}^S$

property of the ZO-ProxSVRG-CooSGE.

In this section, we introduce a simple accelerated stochastic algorithm (MiG) for strongly convex problems. More recently, researchers have proposed accelerated stochastic variance reduced methods for Problem (1), which include Acc-Prox-SVRG (Nitanda, 2014), APCG (Lin et al., 2014), Catalyst (Lin et al., 2015), SPDC (Zhang Xiao, 2015), point-SAGA (Defazio, 2016), and Katyusha (Allen-Zhu, 2017). For strongly convex problems, both Acc-Prox-SVRG (Nitanda, 2014) and Catalyst (Lin et al., 2015) make good use of the "Nesterov's momentum" in (Nesterov, 2004) and attain the corresponding oracle complexities $O((n + b\sqrt{\kappa})\log(1/\epsilon))$.

In this section we make the following assumption:

Assumption 12.1. In problem (**??**), function f is λ-strongly convex.

As we can see in Algorithm 1, y is a convex combination of x and $\widetilde{x}^s$ with the parameter θ. So for implementation, we do not need to keep track of y in the whole inner loop. For the purpose of giving a clean proof, we mark y with iteration number j.

We choose to use the following update for $\widetilde{x}^s$ in contrast to use the last iterate and use it as the initial vector for new epoch.

Next we give the convergence rate of MiG in terms of oracle complexity as follows

In this part, we consider the case of Problem (1) when f is ν-strongly convex.

Similar to existing stochastic variance reduced methods such as SVRG [1] and Prox-SVRG [35], we design a simple fast stochastic variance reduction algorithm with

momentum acceleration for solving smooth objective functions, as outlined in Algorithm 1.

Inspired by the momentum acceleration trick for accelerating first-order optimization methods [17, 20, 1], we design the following update rule

In addition, FSVRG only has one momentum weight θ, compared with the existing methods and . therefore, FSVRG is much simpler than existing accelerated methods [1, 8]. If $b = n$ (i.e., the batch setting), we have ?, and the second term on the right-hand side of (19) diminishes. In other words, FSVRG degenerates to the accelerated deterministic method.

Theorem 12.2. *Suppose A1 and A2 hold, and CooSGD is used in Algorithm 12 and define a positive sequence $\{c_t\}$ as follows:*

$$c_{t-1} = c_t(1+\beta) + \frac{Ld}{bC_\eta} + \frac{L^2 d\theta}{b\mu}$$

Let $c_m = 0$, $C_\eta = \frac{1-\eta L}{2\eta L}$, $\eta = \frac{\rho}{L}$ where $\rho \le \min\{\frac{1}{2}, \frac{b}{8dm^2(4+m)}\}$ $\rho = \min\{\frac{1}{2}, \frac{\sqrt{b}}{6m\sqrt{d(4+m)}}\}$. Then we have

$$\begin{aligned}\mathbb{E}[F(\widetilde{x}^S) - F(x^*)] \le (1-\gamma)^{Sm}\mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \frac{1}{\gamma}\left(\frac{Ld^2\mu^2}{4C_\eta} + \frac{L^2 d^2\mu^2\theta}{4\lambda}\right) \\ + \frac{1}{\gamma}\left((\frac{1}{C_\eta L} + \frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}\right)\end{aligned} \tag{56}$$

where $\gamma = \theta - \frac{\theta^2}{\lambda\eta m}$ and x^* is same as Theorem 10.4.

Proof. We start by applying Lipschitz continuous nature of the gradient of function f

$$\begin{aligned} f(x_t^s) &\le f(x_{t-1}^s) + \left\langle \nabla f(x_{t-1}^s), x_t^s - x_{t-1}^s \right\rangle + \frac{L}{2}\left\|x_t^s - x_{t-1}^s\right\|^2 \\ &= f(x_{t-1}^s) + \left\langle \nabla f(x_{t-1}^s), x_t^s - x_{t-1}^s \right\rangle \\ &\quad + \frac{1}{2\eta}\left\|x_t^s - x_{t-1}^s\right\|^2 - C_\eta L\left\|x_t^s - x_{t-1}^s\right\|^2 \qquad (57) \\ &= f(x_{t-1}^s) + \left\langle \hat{v}_{t-1}, x_t^s - x_{t-1}^s \right\rangle \\ &\quad + \frac{1}{2\eta}\left\|x_t^s - x_{t-1}^s\right\|^2 + \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}, x_t^s - x_{t-1}^s \right\rangle \\ &\quad - C_\eta L\left\|x_t^s - x_{t-1}^s\right\| \qquad (58) \end{aligned}$$

where is based on definition of $C_\eta = \frac{1-\eta L}{2\eta L}$. In (58), we further bound the following

$$\begin{aligned} &\left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}, x_t^s - x_{t-1}^s \right\rangle \\ \le &\mathbb{E}\left[\frac{1}{2C_\eta L}\left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}\right\|^2 + \frac{C_\eta L}{2}\left\|x_t^s - x_{t-1}^s\right\|^2\right] \end{aligned} \tag{59}$$

Substituting the inequality (59) in (58) and taking expectation, we have

$$\begin{aligned}
&\mathbb{E}\left[F(x_t^s) - f(x_{t-1}^s)\right] \\
&\leq \mathbb{E}\left[h(x_t^s) + \left\langle \hat{v}_{t-1}, x_t^s - x_{t-1}^s \right\rangle + (\frac{1}{2\eta} - \frac{C_\eta L}{2}) \left\| x_t^s - x_{t-1}^s \right\|^2\right] \\
&\quad + \mathbb{E}\left[\frac{1}{2C_\eta L} \left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1} \right\|^2\right] \\
&\leq \mathbb{E}\left[\left\langle \theta \hat{v}_{t-1}, z_t^s - z_{t-1}^s \right\rangle\right] + \frac{\theta^2}{2\eta} \left\| z_t^s - z_{t-1}^s \right\|^2 - \frac{C_\eta L}{2} \left\| x_t^s - x_{t-1}^s \right\|^2 \\
&\quad + \mathbb{E}\left[\theta h(z_t^s) + (1-\theta) h(\widetilde{x}^{s-1})\right] + \frac{1}{2C_\eta L} \mathbb{E}\left[\left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1} \right\|^2\right] \qquad (60)
\end{aligned}$$

where in (60) we use the update $x_t^s = \theta z_t^s + (1-\theta)\widetilde{x}^{s-1}$ and convexity of h. Using Three Point Property **??**, with $\phi(x) = h(x) + \left\langle \hat{v}_{t-1}, x - z_{t-1}^s \right\rangle$, $D_l(x,z) = \frac{\theta}{2\eta}\|x-z\|^2$, $z^+ = z_t^s, z = z_{t-1}^s, x^+ = x^*$, it follows

$$\begin{aligned}
&h(z_t^s) + \left\langle \hat{v}_{t-1}, z_t^s - z_{t-1}^s \right\rangle + \frac{\theta}{2\eta} \left\| z_t^s - z_{t-1}^s \right\|^2 \\
&\leq h(x^*) + \left\langle \hat{v}_{t-1}, x^* - z_{t-1}^s \right\rangle \\
&\quad + \frac{\theta}{2\eta}(\left\| z_{t-1}^s - x^* \right\|^2 - \|z_t^s - x^*\|^2). \qquad (61)
\end{aligned}$$

Moreover, substituting (61) into (60), we have

$$\begin{aligned}
&\mathbb{E}\left[F(x_t^s) - f(x_{t-1}^s)\right] \\
&\leq \mathbb{E}\left[\left\langle \theta(\hat{v}_{t-1} - \nabla f(x_{t-1}^s) + \nabla f(x_{t-1}^s)), x^* - z_{t-1}^s \right\rangle + \frac{\theta^2}{2\eta}(\left\| z_{t-1}^s - x^* \right\|^2 - \|z_t^s - x^*\|^2)\right] \\
&\quad + \mathbb{E}\left[\theta h(x^*) + (1-\theta) h(\widetilde{x}^{s-1})\right] \\
&\quad + \frac{1}{2C_\eta L} \mathbb{E}\left[\left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1} \right\|^2\right] - \frac{C_\eta L}{2} \left\| x_t^s - x_{t-1}^s \right\|^2 \qquad (62) \\
&= \mathbb{E}\left[\frac{\theta^2}{2\eta}(\left\| z_{t-1}^s - x^* \right\|^2 - \|z_t^s - x^*\|^2) + \theta h(x^*)\right] \\
&\quad + \mathbb{E}\left[\left\langle \nabla f(x_{t-1}^s), \theta x^* + (1-\theta)\widetilde{x}^{s-1} - x_{t-1}^s \right\rangle\right] \\
&\quad + \frac{1}{2C_\eta L} \mathbb{E}\left[\left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1} \right\|^2\right] - \frac{C_\eta L}{2} \left\| x_t^s - x_{t-1}^s \right\|^2 \\
&\quad + (1-\theta)\mathbb{E}[h(\widetilde{x}^{s-1})] + \mathbb{E}\left[\left\langle \theta(\hat{v}_{t-1} - \nabla f(x_{t-1}^s)), x^* - z_{t-1}^s \right\rangle\right] \qquad (63)
\end{aligned}$$

The equality (63) is obtained by $x_{t-1}^s = \theta z_{t-1}^s + (1-\theta)\widetilde{x}^{s-1}$ and rearranging terms.

Additionally, we have the following,

$$\left\langle \nabla f(x_{t-1}^s), \theta x^* + (1-\theta)\widetilde{x}^{s-1} - x_{t-1}^s \right\rangle$$

$$
\begin{aligned}
&= \left\langle \nabla f(x_{t-1}^s), \theta x^* - \theta z_{t-1}^s \right\rangle \\
&\leq \theta f(x^*) - \theta f(z_{t-1}^s) - \frac{\lambda\theta}{2} \left\| x^* - z_{t-1}^s \right\|^2 \qquad (64) \\
&\leq \theta f(x^*) + (1-\theta) f(\widetilde{x}^{s-1}) - f(x_{t-1}^s) - \frac{\lambda\theta}{2} \left\| x^* - z_{t-1}^s \right\|^2 \qquad (65)
\end{aligned}
$$

where in (64) and (65) we used the strong convexity of f. The last term at the right hand side (RHS) of (63) yields

$$
\begin{aligned}
&\mathbb{E}\left[\left\langle \theta(\hat{v}_{t-1} - \nabla f(x_{t-1}^s)), x^* - z_{t-1}^s \right\rangle \right] \\
&\quad \leq \frac{\theta}{2\lambda} \left\| \hat{v}_{t-1} - \nabla f(x_{t-1}^s) \right\|^2 + \frac{\lambda\theta}{2} \left\| x^* - z_{t-1}^s \right\|^2 \qquad (66)
\end{aligned}
$$

Substituting (65) and (66) into (63)

$$
\begin{aligned}
\mathbb{E}[F(x_t^s)] \leq & \theta F(x^*) + (1-\theta) F(\widetilde{x}^{s-1}) \\
& + \frac{\theta^2}{2\eta} \mathbb{E}[\left\| z_{t-1}^s - x^* \right\|^2 - \| z_t^s - x^* \|^2] \\
& + (\frac{1}{2C_\eta L} + \frac{\theta}{2\lambda}) \mathbb{E}\left[\left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1} \right\|^2 \right] - \frac{C_\eta L}{2} \left\| x_t^s - x_{t-1}^s \right\|^2 . \qquad (67)
\end{aligned}
$$

Hence, we have the following

$$
\begin{aligned}
\mathbb{E}[F(x_t^s) - F(x^*)] \leq & (1-\theta)[F(\widetilde{x}^{s-1}) - F(x^*)] \\
& + \frac{\theta^2}{2\eta} \mathbb{E}[\left\| z_{t-1}^s - x^* \right\|^2 - \| z_t^s - x^* \|^2] \\
& + (\frac{1}{2C_\eta L} + \frac{\theta}{2\lambda}) \mathbb{E}\left[\left\| \nabla f(x_{t-1}^s) - \hat{v}_t \right\|^2 \right] - \frac{C_\eta L}{2} \left\| x_t^s - x_{t-1}^s \right\|^2 . \qquad (68)
\end{aligned}
$$

By telescoping the sum of (68) for $t = 1, 2, \ldots, m$ we obtain

$$
\begin{aligned}
\mathbb{E}[\sum_{t=1}^{m} F(x_t^s) - F(x^*)] \leq & \, m(1-\theta)[F(\widetilde{x}^{s-1}) - F(x^*)] \\
& + \frac{\theta^2}{2\eta} \mathbb{E}[\left\| z_0^s - x^* \right\|^2 - \| z_m^s - x^* \|^2] \\
& + (\frac{1}{2C_\eta L} + \frac{\theta}{2\lambda}) \sum_{t=1}^{m} \mathbb{E}\left[\left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1} \right\|^2 \right] - \frac{C_\eta L}{2} \sum_{t=1}^{m} \left\| x_t^s - x_{t-1}^s \right\|^2 . \qquad (69)
\end{aligned}
$$

We define $\psi_t^s = F(x_t^s) - F(x^*)$. We have

$$
\sum_{t=1}^{m} \mathbb{E}[\psi_t^s] \leq m(1-\theta)[F(\widetilde{x}^{s-1}) - F(x^*)]
$$

$$
\begin{aligned}
&+\frac{\theta^2}{2\eta}\mathbb{E}[\left\|z_0^s-x^*\right\|^2-\|z_m^s-x^*\|^2]\\
&+(\frac{1}{2C_\eta L}+\frac{\theta}{2\lambda})\sum_{t=1}^m\mathbb{E}\left[\left\|\nabla f(x_{t-1}^s)-\hat{v}_{t-1}\right\|^2\right]-\frac{C_\eta L}{2}\sum_{t=1}^m\left\|x_t^s-x_{t-1}^s\right\|^2\\
\le& m(1-\theta)[F(\widetilde{x}^{s-1})-F(x^*)]\\
&+\frac{\theta^2}{2\eta}\mathbb{E}[\left\|z_0^s-x^*\right\|^2-\|z_m^s-x^*\|^2]\\
&+(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})\sum_{t=1}^m\mathbb{E}\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2+m\frac{Ld^2\mu^2}{4C_\eta}+m\frac{L^2d^2\mu^2\theta}{4\lambda}\\
&+m(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}-\frac{C_\eta L}{2}\sum_{t=1}^m\left\|x_t^s-x_{t-1}^s\right\|^2 \qquad (70)
\end{aligned}
$$

$$
\frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2\right]+2\frac{I\{B<n\}\eta\sigma^2}{B}+\eta\frac{L^2d^2\mu^2}{2} \qquad (71)
$$

where in (70), we used Lemma 10.3. Define the following Lyapunov function:

$$
R_t^s=\mathbb{E}\left[\psi_t^s+c_t\left\|x_t^s-\tilde{x}^s\right\|^2\right], \qquad (72)
$$

where $\{c_t\}$ are non-negative coefficients which are defined recursively later. Using (22), we have

$$
\left\|x_t^s-\widetilde{x}^{s-1}\right\|^2\le(1+\frac{1}{\beta})\left\|x_t^s-x_{t-1}^s\right\|^2+(1+\beta)\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2 \qquad (73)
$$

where $\beta>0$. Then from (70) we have

$$
\begin{aligned}
&\sum_{t=1}^m\mathbb{E}[R_t^s]=\sum_{t=1}^m\mathbb{E}\left[\psi_t^s+c_t\left\|x_t^s-\tilde{x}^{s-1}\right\|^2\right]\\
&\le\sum_{t=1}^m\mathbb{E}\left[\psi_t^s+c_t(1+\frac{1}{\beta})\left\|x_t^s-x_{t-1}^s\right\|^2+c_t(1+\beta)\left\|x_{t-1}^s-\tilde{x}^{s-1}\right\|^2\right] \qquad (74)\\
&\le m(1-\theta)[F(\widetilde{x}^{s-1})-F(x^*)]\\
&\quad+\frac{\theta^2}{2\eta}\mathbb{E}[\left\|z_0^s-x^*\right\|^2-\|z_m^s-x^*\|^2]\\
&\quad+\sum_{t=1}^m(c_t(1+\beta)+\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})\mathbb{E}\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2\\
&\quad+\sum_{t=1}^m(c_t(1+\frac{1}{\beta})-\frac{C_\eta L}{2})\left\|x_t^s-x_{t-1}^s\right\|^2\\
&\quad+m\frac{Ld^2\mu^2}{4C_\eta}+m\frac{L^2d^2\mu^2\theta}{4\lambda}+m(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B} \qquad (75)
\end{aligned}
$$

$$
\begin{aligned}
=&m(1-\theta)[F(\widetilde{x}^{s-1})-F(x^*)]\\
&+\frac{\theta^2}{2\eta}\mathbb{E}[\left\|z_0^s-x^*\right\|^2-\|z_m^s-x^*\|^2]\\
&+\sum_{t=1}^{m}c_{t-1}\mathbb{E}\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2\\
&+\sum_{t=1}^{m}(c_t(1+\frac{1}{\beta})-\frac{C_\eta L}{2})\left\|x_t^s-x_{t-1}^s\right\|^2\\
&+m\frac{Ld^2\mu^2}{4C_\eta}+m\frac{L^2d^2\mu^2\theta}{4\lambda}+m(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}
\end{aligned}
\tag{76}
$$

where in (76) we define $c_{t-1}=c_t(1+\beta)+\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda}$ and $c_m=0$. Recursing on t and setting $\beta=\frac{1}{m}$, we have

$$
\begin{aligned}
c_{t-1}&=c_t(1+\beta)+\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\mu}\\
c_{t-1}&=(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})\frac{(1+\beta)^{m-t}-1}{\beta}\\
&\le m(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})((1+\frac{1}{m})^m-1) \qquad (77)\\
&\le m(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})(e-1)\\
&\le 2m(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda}) \qquad (78)
\end{aligned}
$$

where the inequality (77) holds since $(1+1/a)^a\le\lim_{a\to\infty}(1+1/a)^a=e$. Equation (78) implies that

$$
\begin{aligned}
c_t(1+\frac{1}{\beta})&\le 2m(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})(1+m)\\
&\le 4m^2(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})
\end{aligned}
\tag{79}
$$

Moreover, recall $C_\eta=\frac{1-\eta L}{2\eta L}=\frac{1-\rho}{2\rho}$, if we set $\rho\le\min\{\frac{1}{2},\frac{b}{8dm^2(4+m)}\}$, it is easy to verify that We need to pick $C_\eta=\frac{1-\eta L}{2\eta L}=\frac{1-\rho}{2\rho}$ such that we have

$$
\begin{aligned}
4m^2(\frac{Ld}{bC_\eta}+\frac{L^2d\theta}{b\lambda})&\le\frac{C_\eta L}{2} \qquad (80)\\
4m^2(\frac{d}{bC_\eta}+\frac{Ld\theta}{b\lambda})&\le\frac{C_\eta}{2}
\end{aligned}
$$

by choosing $\rho \le \frac{1}{2}$ and $4\rho + \frac{\rho m}{\sqrt{2}} \le \frac{b}{8dm^2}$ $\rho \le \frac{b}{8dm^2(4+m)}$ (79) holds.

$$C_\eta = \frac{1-\eta L}{2\eta L} = \frac{1-\rho}{2\rho}$$

We assume $C_\eta \ge 1$ and thus we should have $\rho \le \frac{1}{2}$. Hence, it is sufficient

$$\begin{aligned}
(\frac{1}{C_\eta} + \frac{L\theta}{\lambda}) &\le \frac{b}{8dm^2} \\
(\frac{2\rho}{1-\rho} + \frac{L\theta}{\lambda}) &\le \frac{b}{8dm^2} \\
4\rho + \frac{\rho m}{\sqrt{2}} &\le \frac{b}{8dm^2}
\end{aligned} \tag{81}$$

Hence, it is sufficient

$$\begin{aligned}
4m^2(\frac{d}{bC_\eta} + \frac{Ld\theta}{b\lambda}) &\le \frac{C_\eta}{2} \\
(\frac{1}{C_\eta} + \frac{L\theta}{\lambda}) &\le \frac{b}{32dm^2\rho} \\
(\frac{2\rho^2}{1-\rho} + \frac{L\theta\rho}{\lambda}) &\le \frac{b}{32dm^2} \\
4\rho^2 + \frac{\rho^2 m}{\sqrt{2}} &\le \frac{b}{32dm^2} \\
\rho^2 &\le \frac{b}{32dm^2(4+m)} \\
\rho &\le \frac{\sqrt{b}}{6m\sqrt{d(4+m)}}
\end{aligned} \tag{82}$$

Combining (79) and (80), we obtain from (76)

$$\begin{aligned}
\sum_{t=1}^{m} \mathbb{E}[R_t^s] &\le m(1-\theta)[F(\widetilde{x}^{s-1}) - F(x^*)] \\
&+ \frac{\theta^2}{2\eta}\mathbb{E}[\left\|z_0^s - x^*\right\|^2 - \|z_m^s - x^*\|^2] \\
&+ \sum_{t=1}^{m} c_{t-1}\mathbb{E}\left\|x_{t-1}^s - \widetilde{x}^{s-1}\right\|^2 \\
&+ m\frac{Ld^2\mu^2}{4C_\eta} + m\frac{L^2d^2\mu^2\theta}{4\lambda} + m(\frac{1}{C_\eta L} + \frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}
\end{aligned} \tag{83}$$

Therefore, we have

$$
\begin{aligned}
&\sum_{t=1}^{m}\mathbb{E}[R_t^s]-\sum_{t=1}^{m-1}c_{t-1}E\left\|x_{t-1}^s-\widetilde{x}^{s-1}\right\|^2\\
&=\sum_{t=1}^{m}\mathbb{E}[\psi_t^s]+c_m\left\|x_m^s-\widetilde{x}^{s-1}\right\|^2-c_0\left\|x_0^s-\widetilde{x}^{s-1}\right\|^2\\
\leq& m(1-\theta)[F(\widetilde{x}^{s-1})-F(x^*)]\\
&+\frac{\theta^2}{2\eta}\mathbb{E}[\left\|z_0^s-x^*\right\|^2-\|z_m^s-x^*\|^2]\\
&+m\frac{Ld^2\mu^2}{4C_\eta}+m\frac{L^2d^2\mu^2\theta}{4\lambda}+m(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}
\end{aligned}
\tag{84}
$$

Recall convexity of function F and the definition $\widetilde{x}^s=\frac{1}{m}\sum_{t=0}^{m-1}x_{t+1}^s$ $\widetilde{x}^s=\frac{1}{m}\sum_{t=1}^{m}x_t^s$, we have

$$
F(\widetilde{x}^s)=F(\frac{1}{m}\sum_{t=1}^{m}x_t^s)\leq\frac{1}{m}\sum_{t=1}^{m}F(x_t^s)
\tag{85}
$$

Using $x_0^s=\widetilde{x}^{s-1}$ and $c_m=0$ in (84), we obtain

$$
\begin{aligned}
[F(\widetilde{x}^s)-F(x^*)]\leq&(1-\theta)[F(\widetilde{x}^{s-1})-F(x^*)]\\
&+\frac{\theta^2}{2\eta m}\mathbb{E}[\left\|z_0^s-x^*\right\|^2-\|z_m^s-x^*\|^2]\\
&+\frac{Ld^2\mu^2}{4C_\eta}+\frac{L^2d^2\mu^2\theta}{4\lambda}+(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}\\
\leq&(1-\theta+\frac{\theta^2}{\lambda\eta m})[F(\widetilde{x}^{s-1})-F(x^*)]\\
&+\frac{Ld^2\mu^2}{4C_\eta}+\frac{L^2d^2\mu^2\theta}{4\lambda}+(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}
\end{aligned}
\tag{86}
$$

$$
\begin{aligned}
=&\tilde{\alpha}[F(\widetilde{x}^{s-1})-F(x^*)]+\frac{Ld^2\mu^2}{4C_\eta}+\frac{L^2d^2\mu^2\theta}{4\lambda}\\
&+(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}
\end{aligned}
\tag{87}
$$

where in (86) we used the strong convexity of F and in (87) we define $\tilde{\alpha}=(1-\gamma)$ where $\gamma=\theta-\frac{\theta^2}{\lambda\eta m}$. The rest of the proof essentially follow along the lines of 11.1 under a different parameter setting. Based on the definition of $\widetilde{\Psi}^s:=\frac{\mathbb{E}[F(\tilde{x}^s)-F(x^*)]}{\alpha^s}$, we can simplify (87) as

$$
\widetilde{\Psi}^s\leq\widetilde{\Psi}^{s-1}+\frac{Ld^2\mu^2}{4C_\eta}+\frac{L^2d^2\mu^2\theta}{4\lambda}+(\frac{1}{C_\eta L}+\frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}
\tag{88}
$$

Now, we sum up (87) for all epochs $s = 1, 2, \ldots, S$ and telescoping to obtain

$$\begin{aligned}
\mathbb{E}[F(\widetilde{x}^S) - F(x^*)] \leq\ & \widetilde{\alpha}^S \mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \widetilde{\alpha}^S \sum_{s=1}^{S} \frac{1}{\widetilde{\alpha}^s} \left(\frac{Ld^2\mu^2}{4C_\eta} + \frac{L^2 d^2 \mu^2 \theta}{4\lambda} \right) \\
& + \widetilde{\alpha}^S \sum_{s=1}^{S} \frac{1}{\widetilde{\alpha}^s} \left(2(\frac{1}{2C_\eta L} + \frac{\theta}{2\lambda}) \frac{I\{B < n\}\sigma^2}{B} \right) \\
=& \widetilde{\alpha}^S \mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \frac{1 - \widetilde{\alpha}^S}{1 - \widetilde{\alpha}} \left(\frac{Ld^2\mu^2}{4C_\eta} + \frac{L^2 d^2 \mu^2 \theta}{4\lambda} \right) \\
& + \frac{1 - \widetilde{\alpha}^S}{1 - \widetilde{\alpha}} \left((\frac{1}{C_\eta L} + \frac{\theta}{\lambda}) \frac{I\{B < n\}\sigma^2}{B} \right) \\
\leq& \alpha^{Sm} \mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \frac{1}{1 - \alpha} \left(\frac{Ld^2\mu^2}{4C_\eta} + \frac{L^2 d^2 \mu^2 \theta}{4\lambda} \right) \\
& + \frac{1}{1 - \alpha} \left((\frac{1}{C_\eta L} + \frac{\theta}{\lambda}) \frac{I\{B < n\}\sigma^2}{B} \right) \qquad (89) \\
=& (1 - \gamma)^{Sm} \mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \frac{1}{\gamma} \left(\frac{Ld^2\mu^2}{4C_\eta} + \frac{L^2 d^2 \mu^2 \theta}{4\lambda} \right) \\
& + \frac{1}{\gamma} \left((\frac{1}{C_\eta L} + \frac{\theta}{\lambda}) \frac{I\{B < n\}\sigma^2}{B} \right) \qquad (90)
\end{aligned}$$

where (89) uses $\widetilde{\alpha} \leq \alpha$ and (90) holds since $\alpha = 1 - \gamma$. The proof is now complete. □

As it is observed in previous theorems, all gradient have a common error bounded induced by ZO-estimator and batch size. If μ and B are selected appropriately as we elaborated in Corollary **??**, then we obtain the error term $O(d \log \frac{1}{\epsilon})$ which exhibits a sub-linear convergence rate.

Next we give the convergence rate of MiG in terms of oracle complexity as follows

Corollary 12.3. *The constant $\alpha = 1 - \gamma$ is minimized by choosing $\theta = \frac{m\eta\lambda}{\sqrt{2}}$. Suppose we set $b = m^2$ $b = m^2$, $\rho = \frac{1}{8d(4+m)}$ $\rho = \frac{1}{6\sqrt{d(4+m)}}$ and $\theta = \frac{m\eta\lambda}{\sqrt{2}}$. Then we have the following convergence result:*

$$\begin{aligned}
\mathbb{E}[F(\widetilde{x}^S) - F(x^*)] \leq\ & (1 - \frac{m\eta\lambda}{2})^{Sm} \mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \frac{2}{m\eta\lambda} \left(\frac{Ld^2\mu^2}{4\frac{1-\eta L}{2\eta L}} + \frac{L^2 d^2 \mu^2 m\eta\lambda}{4\lambda} \right) \\
& + \frac{2}{m\eta\lambda} \left((\frac{1}{L\frac{1-\eta L}{2\eta L}} + \frac{m\eta\lambda}{\lambda}) \frac{I\{B < n\}\sigma^2}{B} \right) \qquad (91) \\
\mathbb{E}[F(\widetilde{x}^S) - F(x^*)] \leq\ & (1 - \frac{m\eta\lambda}{2})^{Sm} \mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \left(\frac{2L^2 d^2 \mu^2}{m\lambda} + \frac{L^2 d^2 \mu^2}{2\lambda} \right) \\
& + \left((\frac{8}{m\lambda} + \frac{2}{\lambda}) \frac{I\{B < n\}\sigma^2}{B} \right) \qquad (92)
\end{aligned}$$

$$\mathbb{E}[F(\widetilde{x}^S) - F(x^*)] \le (1-\gamma)^{Sm}\mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \frac{1}{\gamma}\left(\frac{Ld^2\mu^2}{4C_\eta} + \frac{L^2d^2\mu\theta}{4}\right)$$
$$+ \frac{1}{\gamma}\left((\frac{1}{C_\eta L} + \frac{\theta}{\mu})\frac{I\{B<n\}\sigma^2}{B}\right) \tag{93}$$

In order to acquire explicit dependence on these parameters and to explore deeper insights of convergence, we simplify (56) for a specific parameter setting, as formalized below.

Corollary 12.4. *Let step size $\eta = \frac{1}{6L}$ and b denote the minibatch size. Then the final iteration point $\tilde{x}^S$ in Algorithm* **??** *satisfies $\mathbb{E}[\Phi(\tilde{x}^S) - \Phi^*] \le \epsilon$ under PL condition. We distinguish the following two cases:*

1) Under Assumption 1, we let batch size $B = \min\{(\eta + \frac{\theta}{\lambda})\frac{\sigma^2}{\gamma\epsilon}, n\}$ and $\mu \le \frac{\sqrt{\gamma\epsilon}}{\sqrt{\frac{Ld^2}{4C_\eta} + \frac{L^2d^2\theta}{4\lambda}}}$.

The number of SFO calls is bounded by

$$O\left((n \wedge \frac{1}{\lambda\epsilon})\frac{1}{\lambda\sqrt{b}}\log\frac{1}{\epsilon} + \frac{b}{\lambda}\log\frac{1}{\epsilon}\right).SB + Smb = O(\frac{B}{\gamma m}\log\frac{1}{\epsilon} + \frac{b}{\gamma}\log\frac{1}{\epsilon})$$

where $\wedge$ denotes the minimum. The number of PO calls equals to the total number of iterations T which is bounded by

$$O\left(\frac{1}{\lambda}\log\frac{1}{\epsilon}\right).T = Sm = O(\frac{1}{\gamma}\log\frac{1}{\epsilon})$$

2) Given the setting of Corollary 12.4, we let batch size $B = \min\{\frac{I\{B<n\}\sigma^2}{\lambda\epsilon}, n\}$ and smoothing parameter $\mu \le \frac{\sqrt{\lambda\epsilon}}{6Ld}$. The number of SFO calls simplifies to $SB + Smb = O(\frac{B\sqrt{d}}{b^{3/4}\lambda}\log\frac{1}{\epsilon} + \frac{b^{3/4}\sqrt{d}}{\lambda}\log\frac{1}{\epsilon})$

Proof. From Theorem 11.1, we have

$$\mathbb{E}[F(\widetilde{x}^S) - F(x^*)] \le (1-\gamma)^{Sm}\mathbb{E}[F(\widetilde{x}^0) - F(x^*)] + \frac{1}{\gamma}\left(\frac{Ld^2\mu^2}{4C_\eta} + \frac{L^2d^2\mu^2\theta}{4\lambda}\right)$$
$$+ \frac{1}{\gamma}\left((\frac{1}{C_\eta L} + \frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{B}\right) = 3\epsilon \tag{94}$$

which gives the total number of iterations $T = Sm = S\sqrt{b} = O(\frac{1}{\lambda}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{1}{\gamma}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{1}{m\eta\lambda}\log\frac{1}{\epsilon})$. The number of PO calls equals to $T = Sm = O(\frac{1}{\lambda}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{1}{\gamma}\log\frac{1}{\epsilon})$ $T = Sm = O(\frac{1}{m\eta\lambda}\log\frac{1}{\epsilon})$. The number of SFO calls equals to $Sn + Smb = O(\frac{n}{\lambda\sqrt{b}}\log\frac{1}{\epsilon} + \frac{b}{\lambda}\log\frac{1}{\epsilon})$ if $B = n$, or equals to $Sn + Smb = O(\frac{B}{\lambda\sqrt{b}}\log\frac{1}{\epsilon} + \frac{b}{\lambda}\log\frac{1}{\epsilon})$ $SB + Smb = O(\frac{B}{\gamma m}\log\frac{1}{\epsilon} + \frac{b}{\gamma}\log\frac{1}{\epsilon})$ $SB + Smb = O(\frac{B}{m^2\eta\lambda}\log\frac{1}{\epsilon} + \frac{b}{m\eta\lambda}\log\frac{1}{\epsilon})$if $B < n$(note that $\frac{I\{B<n\}3\sigma^2}{\lambda B} \le \epsilon$ since $B \ge 6\sigma^2/\lambda\epsilon$ $\frac{1}{\gamma}\left((\frac{1}{C_\eta L} + \frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{\epsilon}\right) = \frac{1}{\gamma}\left((\eta + \frac{\theta}{\lambda})\frac{I\{B<n\}\sigma^2}{\epsilon}\right)$ $= \left((\frac{1}{m\lambda} + \frac{1}{\lambda})\frac{I\{B<n\}\sigma^2}{\epsilon}\right)$). $\mu \le \frac{\sqrt{\gamma\epsilon}}{\sqrt{\frac{Ld^2}{4C_\eta} + \frac{L^2d^2\theta}{4\lambda}}}\mu \le \frac{\sqrt{m\eta\lambda\epsilon}}{\sqrt{\frac{Ld^2}{4C_\eta} + \frac{L^2d^2\theta}{4\lambda}}}\mu \le \frac{\sqrt{m\eta\lambda\epsilon}}{\sqrt{\frac{Ld^2}{4\frac{1-\eta L}{2\eta L}} + \frac{L^2d^2\frac{m\eta\lambda}{\sqrt{2}}}{4\lambda}}} =$

$\frac{\sqrt{m\eta\lambda\epsilon}}{\sqrt{\frac{Ld^2\eta L}{2(1-\eta L)} + \frac{L^2d^2m\eta}{4\sqrt{2}}}} = \frac{\sqrt{m\eta\lambda\epsilon}}{\sqrt{Ld^2\eta L + \frac{L^2d^2m\eta}{4\sqrt{2}}}} = \frac{\sqrt{m\lambda\epsilon}}{\sqrt{L^2d^2 + L^2d^2m}} = \frac{\sqrt{\lambda\epsilon}}{3Ld}$ □

For $B \leq$ the SZO complexity of our proposed method is $O()$ This result is similar to SCSG [] if the dimension d is not large enough. Furthermore, in our algorithm, we set B as a value that can be less than n rather than a value which is fixed and equals to n. In ZO-SVRG (Algorithm 2), the total number of gradient evaluations is given by $nS + bT$, where $T = mS$. Therefore, by fixing the number of iterations T, the function query complexity of ZO-SVRG using the studied estimators is then given by $O(d(nS + bT))$, respectively.

This result implies that in the strongly convex setting, MiG enjoys the best-known oracle complexity for stochastic first-order algorithms.

13 Appendix

In this section, we provide the detailed proofs of the above lemmas and theorems. First, we give some useful properties of the CooSGE and the GauSGE, respectively.

Lemma 13.1 (Three-Point Property)**.** *Let $\phi(\cdot)$ be a convex function, and let $D_l(\cdot,\cdot)$ be the Bregman distance for $l(\cdot)$. For a given vector z, let*

$$z^+ = arg \min_{x \in \mathbb{R}^d} \{\phi(x) + D_l(x, z)\}.$$

Then

$$\phi(x) + D_l(x, z) \geq \phi(z^+) + D_l(x^+, z) + D_l(x, z^+) \qquad for\ all\ x \in \mathbb{R}^n \tag{95}$$

with equality holding in the case when $\phi(\cdot)$ is a linear function and $l(\cdot)$ is a quadratic function.

Lemma 13.2. *Assume that the function $f(x)$ is L-smooth. Let $\hat{\nabla} f(x)$ denote the estimated gradient defined by* **CooSGE***. Define $f_{\mu_j} = \mathbb{E}_{u \sim U[\mu_j, \mu_j]} f(x + ue_j)$, where $U[-\mu_j, \mu_j]$ denotes the uniform distribution at the interval $[\mu_j, \mu_j]$. Then we have 1) f_{μ_j} is L-smooth, and*

$$\hat{\nabla} f(x) = \sum_{j=1}^{d} \frac{\partial f_{\mu_j}}{\partial x_j} e_j \tag{96}$$

where $\partial f / \partial x_j$ denotes the partial derivative with respect to jth coordinate.

2) For $j \in [d]$,

$$\left| f_{\mu_j}(x) - f(x) \right| \leq \frac{L\mu_j^2}{2} \tag{97}$$

$$\left| \frac{\partial f_{\mu_j}(x)}{\partial x_j} \right| \leq \frac{L\mu_j^2}{2} \tag{98}$$

3) If $\mu = \mu_j$ for $j \in [d]$, then

$$\left\| \hat{\nabla} f(x) - \nabla f(x) \right\|^2 \leq \frac{L^2 d^2 \mu^2}{4} \tag{99}$$

Lemma 13.3. *Let $x^+ = Prox_{\eta h}(x - \eta v)$, then the following inequality holds:*

$$\Phi(x^+) \le \Phi(z) + \langle \nabla f(x) - v, x^+ - z\rangle - \frac{1}{\eta}\langle x^+ - x, x^+ - z\rangle + \frac{L}{2}\|x^+ - x\|^2 + \frac{L}{2}\|z - x\|^2, \forall z \in \mathbb{R}^d. \tag{100}$$

Proof. First, we recall the proximal operator

$$\mathrm{Prox}_{\eta h}(x - \eta v) := \arg\min_{y \in \mathbb{R}^d} \left(h(y) + \frac{1}{2\eta}\|y - x\|^2 + \langle v, y\rangle \right) \tag{101}$$

For the nonsmooth function $h(x)$, we have

$$\begin{aligned} h(x^+) &\le h(z) + \langle p, x^+ - z\rangle \\ &= h(z) - \left\langle v + \frac{1}{\eta}(x^+ - x), x^+ - z \right\rangle \end{aligned} \tag{102}$$

where $p \in \partial h(x^+)$ such that $p + \frac{1}{\eta}(x^+ - x) + v = 0$ according to the optimality condition of (101), and (102) due to the convexity of h.

$$f(x^+) \le f(x) + \langle \nabla f(x), x^+ - x\rangle + \frac{L}{2}\|x^+ - x\|^2 \tag{103}$$

$$-f(z) \le -f(x) + \langle -\nabla f(x), z - x\rangle + \frac{L}{2}\|z - x\|^2 \tag{104}$$

where (103) holds since $f(x)$ has L-Lipschitz continuous gradient, and (104) holds since $-f(x)$ has the same L-Lipschitz continuous gradient as $f(x)$.

This lemma is proved by adding (102), (103), (104), and recalling $\Phi(x) = f(x) + h(x)$. □

Lemma 13.4. *Let $x_t^s = Prox_{\eta h}(x_{t-1}^s - \eta \hat{v}_{t-1}^s)$ and $\overline{x}_t^s = Prox_{\eta h}(x_{t-1}^s - \eta \nabla f(x_{t-1}^s))$. Then the following inequality holds*

$$\left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \le \eta \left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s \right\|^2$$

Proof. First, we obtain $\left\|x_t^s - \overline{x}_t^s\right\|$ and $\left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\|$ as follows

$$h(x_t^s) \le h(\overline{x}_t^s) - \left\langle \hat{v}_{t-1}^s + \frac{1}{\eta}(x_t^s - x_{t-1}^s), x_t^s - \overline{x}_t^s \right\rangle \tag{105}$$

$$h(\overline{x}_t^s) \le h(x_t^s) - \left\langle \nabla f(x_{t-1}^s) + \frac{1}{\eta}(\overline{x}_t^s - x_{t-1}^s), \overline{x}_t^s - x_t^s \right\rangle \tag{106}$$

where (105) and (106) hold due to (102). Adding (105) and (106), we have

$$\begin{aligned} \frac{1}{\eta}\langle x_t^s - \overline{x}_t^s, x_t^s - \overline{x}_t^s\rangle &\le \left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s), x_t^s - \overline{x}_t^s \right\rangle \\ \frac{1}{\eta}\|x_t^s - \overline{x}_t^s\|^2 &\le \left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s) \right\| \|x_t^s - \overline{x}_t^s\| \end{aligned} \tag{107}$$

$$\|x_t^s - \overline{x}_t^s\| \leq \eta \left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\| \tag{108}$$

where (107) uses Cauchy-Schwarz inequality. Now this lemma is proved using Cauchy-Schwarz inequality and (108), i.e., $\left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s, x_t^s - \overline{x}_t^s \right\rangle \leq \left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\| \left\|x_t^s - \overline{x}_t^s\right\| \leq \eta \left\|\nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s\right\|^2$. □