An Acceleration of Derivative-Free Proximal Stochastic Gradient Method for Nonconvex Nonsmooth Optimization

March 2018

Abstract

We provide a simpler analysis with better dependence on dimension of problem d.

1 Introduction

1.1 what we want to do

In this paper, we consider nonsmooth nonconvex finite-sum optimization problems of the form

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

where each $f_i(x)$ is possibly nonconvex and smooth loss function, and $h(x)$ is a nonsmooth but convex structure regularizer such as l_1-norm regularization. The generic form (1.1) encompasses many machine learning problems, ranging from generalized linear models to neural networks and from convex optimization such as Lasso, SVM to highly nonconvex problem such as optimizing deep neural networks.

We will study the design and analysis of a class of variance reduced and faster converging stochastic zeroth-order (SZO) optimization methods for (1.1). To reduce the variance accelerate SZO 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 Johnson and Zhang (2013); Reddi et al. (2016a); Nitanda (2016); Allen-Zhu and Yuan (2016); Lei et al. (2017). Due to the variance reduction, it improves the convergence rate of stochastic gradient descent (SGD) complexity from $O(1/\epsilon^2)$ to $O(1/\epsilon)$.

1.2 Background in research

In recent research Beck and Teboulle (2009), the accelerated proximal gradient (PG) methods are proposed 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 nonconvex optimization. To solve the big data problems, the incremental or stochastic PG methods Bertsekas (2011); Xiao and Zhang (2014) were developed for large-scale convex optimization. There has been extensive research when $f(x)$ is convex (see e.g., Nesterov (2013), Xiao and Zhang (2014); Defazio et al. (2014); Lan and Zhou (2017); Allen-Zhu (2017). In particular, if each f_i is strongly-convex, Xiao and Zhang (2014) proposed the Prox-SVRG algorithm for large-scale problems, 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 nonconvex case has attracted significant attention. For a general nonsmooth nonconvex case, the research is still somewhat limited. Correspondingly, Ghadimi and Lan (2016); Reddi et al. (2016b) proposed the stochastic PG methods for large-scale nonconvex optimization. Very recently, Li and Li (2018) proposed an algorithm generalized the results by Reddi et al. (2016b) and improved the stochastic gradient complexity.

1.3 Reason to use zeroth–order techniques

The main issue for these accelerated algorithms is that most of their algorithm designs involve computing first-order information. 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 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 perform while the functional evaluations of predicted structures are easily obtained Sokolov et al. (2016).

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). Clearly, the above PG methods will fail in dealing with these scenarios. 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. The optimization problem of equation (1.1) in such situations is referred to stochastic proximal zeroth-order optimization. These algorithms achieve gradient-free optimization by approximating the full gradient via gradient estimators based on only the function values Brent (2013); Spall (2005). Hence, SZO 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 Papernot et al. (2017); Madry et al. (2017); Chen et al. (2017) Conn et al. (2009). Additional classes of applications include network control and management with time-varying constraints and limited computation capacity Chen and Giannakis (2019); Liu et al. (2017), and parameter inference of black-box systems Fu (2002); Lian et al. (2016).

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.

1.4 Problem with existing methods

Although many SZO algorithms have recently been developed and analyzed Liu et al. (2017); Flaxman et al. (2005); Shamir (2013); Agarwal et al. (2010); Nesterov and Spokoiny (2017); Duchi et al. (2015); Shamir (2017); Dvurechensky et al. (2018); Wang et al. (2017), they often suffer from the high variances of SZO gradient estimates, and in turn, hampered convergence rates. Thus, a useful technique to accelerate the convergence of SZO 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 (nonconvex) machine learning problems.

Until now, there are few zeroth-order stochastic methods for solving the problem (1.1) e.g., Ghadimi and Lan (2016) and Huang et al. (2019). Specifically, Ghadimi and Lan (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 iteration complexity of $O(\frac{d}{\epsilon^2})$ based on two function evaluations, which is significantly slower than $O(\frac{d}{\epsilon})$, the best known convergence rate of the zeroth-order stochastic algorithm. To accelerate the RSPGF algorithm, the zeroth-order variance reduction methods, i.e., ZO-SVRG Liu et al. (2018b) and ZO-ProxSVRG/SAGA Huang et al. (2019) with the iteration complexity of $O(\frac{d}{\epsilon})$, were introduced to reduce the variance of estimated stochastic gradient. A key concern in the development of iterative stochastic zeroth-oder algorithms for solving (1.1) is the order of the necessary number of functional evaluations, namely SZO calls or iteration complexity. Without a careful treatment, the dimension dependent factor in the convergence rate (i.e., d) could be a critical factor affecting the optimization performance. To mitigate this factor, we propose an accelerated ZO proximal variants, utilizing reduced variance gradient estimators. These yield a faster iteration complexity towards $O(\sqrt{d}/\epsilon)$, which is, to our knowledge, the best iteration complexity bound so far achieved for ZO stochastic optimization with nonconvex structure. This indicates an improvement over existing results up to a factor of $\sqrt{d}$.

1.5 Compare with other methods

We list the results of proposed algorithms and other related ones in Table 4 and Figure **??**.
In Table 4, we summarize the convergence rates and the function query complexities of ZO-PSVRG+. For comparison, we also present the results of ZO-SGD Ghadimi and Lan (2013), ZO-SVRC Gu et al. (2016), ZO-SVRG-Coord Liu et al. (2018b) and ZO-ProxSVRG/SAGA Huang et al. (2019). Table 4 shows that RGF has the highest query complexity and nevertheless has the worst convergence rate. ZO-SVRG-coord and ZO-ProxSVRG/SAGA yield the best convergence rate in the cost of high query complexity. By contrast, ZO-PSVRG+ could achieve better trade-offs between the convergence rate and the query complexity.

2 Main Challenge

Although ProxSVRG has shown a great promise, applying similar ideas to ZO optimization is not a trivial task. Zeroth-order method are applicable for problem that the first-order gradient is not available, but, due to the existence of perturbation, gradient estimation has complex coupling structures, which makes them hard to be extended to more settings. The other main challenge is due to the fact that ProxSVRG 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 SZO optimization and ProxSVRG and improve the efficiency of exiting ZO variance reduced methods for problem (1.1).

3 Main contributions

We propose and evaluate a novel SZO algorithm for nonconvex nonsmooth stochastic optimization, ZO-PSVRG+, which integrates ProxSVRG with ZO gradient estimators. We show that compared to ProxSVRG, ZO-PSVRG+ achieves a similar convergence with SZO complexity of $O(\sqrt{d}/\epsilon)$. Our work offers a comprehensive study on how ZO gradient estimators affect ProxSVRG on both iteration complexity (i.e., convergence rate) and function query complexity. Note that our considered problem does not necessarily satisfy bounded gradients assumption in Ghadimi and Lan (2016); Huang et al. (2019). Our main technical contribution lies in the new convergence analysis of ZO-PSVRG+, which has notable difference from that of ZO-SVRG Liu et al. (2018b) and ZO-ProxSVRG Huang et al. (2019). The convergence results are stated in terms of the number of stochastic zeroth-order (SZO) calls and proximal oracle (PO) calls.

In particular, our algorithm has iteration complexity $O(\frac{\sqrt{d}}{\epsilon})$ compared with $O(\frac{d}{\epsilon^2})$ of RGF Ghadimi and Lan (2016) and $O(\frac{d}{\epsilon})$ of ZO-ProxSVRG/SAGA Huang et al. (2019) (the existing variance-reduce SZO proximal algorithm for solving nonconvex nonsmooth problems). In other words, the proposal expectational results have better dependence on d compared to the existing variance-reduced SZO methods and can strike a balance between iteration complexity and function query complexity.

We would like to highlight the following results yielded by our new analysis:

1) ZO-PSVRG+ is $\frac{n\sqrt{b}}{\sqrt{d}}$ (resp. $\frac{\sqrt{b}}{\sqrt{d\epsilon}}$) times faster than RSPGF in terms of the number of SZO calls when $bm \leq 1/\epsilon$ (resp. $bm \leq n$), and $n/m\sqrt{bd}\epsilon$ times faster than ProxGD when $bm > 1/\epsilon$ (resp. $bm > n$). Note that the number of PO calls equal to $O(1/\epsilon)$ and $O(1/\epsilon^2)$ for ZO-PSVRG+ and RSPGF, respectively. Obviously, for any super constant b, ZO-PSVRG+ is strictly better than RSPGF. ZO-PSVRG+ also matches the best result achieved by ZO-ProxSVRG at $b = n^{2/3}$ for $m = \sqrt{b}$, and it is strictly better for smaller b (using less PO calls). See Figure **??** for an overview.

2) Assuming that the variance of the stochastic zeroth-order gradient is bounded (see Assumption **??**), i.e. online/stochastic setting, ZO-PSVRG+ generalizes the best result achieved by ZO-SVRG-Coord, recently proposed by Liu et al. (2018b) for the smooth

nonconvex case, i.e., $h(x) = 0$ in form (1.1) (see Table 4, the 4th row). ZO-PSVRG+ is more straightforward than ZO-SVRG-Coord proposed by Liu et al. (2018b) very recently, and yields simpler proof. Our results also improve the results of ZO-SVRG-Coord in terms of the number of SZO calls, by a factor of $\sqrt{d}$. Hence, we partially answer the open question if the dependence on the dimension d for the convergence analysis proposed in Liu et al. (2018b) is optimal.

We also note that RGF and ZO-ProxSVRG Liu et al. (2018b) achieved their best convergence results with $b = 1$ and $b = n^{2/3}$ respectively, while ZO-PSVRG+ achieves the best result with $b = 1/\epsilon^{2/3}$ (see Figure **??**), 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 ZO-PSVRG+ and ZO-ProxSVRG in the MNIST experiments is 4096 and 256, respectively (see the second row of Figure **??**).

3) For the nonconvex functions satisfying Polyak-Łojasiewicz condition Polyak (1963), we prove that ZO-PSVRG+ achieves a global linear convergence rate similar to first-order ProxSVRG. Thus, ZO-PSVRG+ can automatically switch to the faster linear convergence in some regions. This generalizes the results of MD Duchi et al. (2015) and achieves linear convergence compared with sublinear rate of MD, (see Table 4). Also see the remarks after Theorem 40 for more details. To the best of our knowledge, this is the first paper that leverages the PL condition for improving the convergence of SZO. It is also notable that the convergence rate achieved in this case can be as good as first-order ProxSVRG. This method achieves an improved oracle complexity by a factor of $\sqrt{d}$ versus RGF for nonsmooth convex problems.

To demonstrate the flexibility of our approach in managing trade-off between the rate of convergence and the number of SZO calls, 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.

4 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 Flaxman et al. (2005); Shamir (2013), and the latter computes a finite difference using two random function queries Agarwal et al. (2010); Nesterov and Spokoiny (2017). 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.

Nesterov and Spokoiny (2017) Nesterov and Spokoiny (2017) proposed several random gradient-free methods by using Gaussian smoothing technique. Duchi et al. Duchi et al. (2015) proposed a zeroth-order mirror descent algorithm.

Despite the meteoric rise of two-point based ZO algorithms, most of the work is restricted to convex problems Liu et al. (2017); Duchi et al. (2015); Shamir (2017);

Dvurechensky et al. (2018); Wang et al. (2017). For example, a ZO mirror descent algorithm proposed by Duchi et al. (2015) has an exact rate $O(d/\epsilon^2)$, where d is the number of optimization variables. The same rate is obtained by bandit convex optimization Shamir (2017) and ZO online alternating direction method of multipliers Liu et al. (2017) for nonsmooth problems. More recently, Yu et al. (2018); Dvurechensky et al. (2018) presented the accelerated zeroth-order methods for the convex optimization. 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.

The above zeroth-order methods mainly focus on the (strongly) convex problems. Thus, exploring zeroth-order stochastic methods for the nonconvex optimization is indeed desirable. In fact, there exist many nonconvex machine learning tasks, whose explicit gradients are not available, such as the nonconvex black-box learning problems Chen et al. (2017); Liu et al. (2018b). In contrast to the convex setting, nonconvex ZO algorithms are comparatively under-studied except a few recent attempts Lian et al. (2016); Nesterov and Spokoiny (2017); Ghadimi and Lan (2013); Hajinezhad et al. (2017); Gu et al. (2016); Kazemi and Wang (2018). 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. (2018a,b) proposed the zeroth-order stochastic variance reduction gradient (ZO-SVRG) methods. In contrast to convex optimization, in nonconvex setting the stationary condition is used to measure the convergence of stationary points. Recently, for the nonsmooth nonconvex case, Huang et al. (2019) provided two algorithms called ZO-ProxSVRG and ZO-ProxSAGA, which are based on the well-known variance reduction techniques ProxSVRG and ProxSAGA Reddi et al. (2016b). Before that, Ghadimi and Lan (2013) also considered the stochastic case (here we denote it as RSPGF). However, RSPGF requires the batch sizes being a large number (i.e., $\Omega(1/\epsilon)$) or increasing with iterations. Note that RSPGF may reduce to deterministic proximal gradient descent (ZO-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 RSPGF) 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. Hence, it is important to study the convergence in moderate and constant minibatch size regime.

Moreover, to solve the large-scale machine learning problems, some asynchronous parallel stochastic zeroth-order algorithms have been proposed in Gu et al. (2018b); Lian et al. (2016); Gu et al. (2018a). In Lian et al. (2016), an asynchronous ZO stochastic coordinate descent (ZO-SCD) was derived for parallel optimization and achieved the rate of $O(d/\epsilon^2)$. Note that from the perspectives of both computational efficiency and statistical generalization, always computing full-gradient may not be desirable for large-scale machine learning problems. This motivates our study on a more general framework ZO-SVRG under different gradient estimators.

Method	Problem	Stepsize	Convergence rate	SZO complexity
RGFNesterov and Spokoiny (2017)	NS(C)	$O\left(\frac{1}{\sqrt{dT}}\right)$	$O\left(\frac{d^2}{\epsilon^2}\right)$	$O\left(\frac{nd^2}{\epsilon^2}b\right)$
RSPGFGhadimi and Lan (2016)	S(NC)+NS(C)	$O(1)$	$O\left(\frac{d}{\epsilon^2}\right)$	$O\left(\frac{nd}{\epsilon^2}\right)$
MD Duchi et al. (2015)	S(SC)	$\frac{1}{\sqrt{dT}}$	$O\left(\frac{d}{\epsilon^2}\right)$	$O\left(\frac{db}{\epsilon^2}\right)$
ZO-SVRG-Coord Liu et al. (2018b)	S(NC)	$O\left(\frac{1}{d}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O(\frac{nd^2}{\epsilon}+\frac{d^2b}{\epsilon})$
ZO-ProxSVRGGu et al. (2018a)	S(NC)+NS(C)	$O\left(\frac{1}{d}\right)$	$O\left(\frac{d}{\epsilon}\right)$	$O(\frac{nd^2}{\epsilon\sqrt{b}}+\frac{md^2\sqrt{b}}{\epsilon})$
ZO-ProxSAGAGu et al. (2018a)	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)	$O\left(\frac{1}{\sqrt{d}}\right)$	$O\left(\frac{\sqrt{d}}{\epsilon}\right)$	$O\left(\min\{n,\frac{1}{\epsilon}\}\frac{d\sqrt{d}}{\epsilon\sqrt{b}}+\frac{md\sqrt{db}}{\epsilon}\right)$
Ours	S(PL)+NS(C)	$O\left(\frac{1}{\sqrt{d}}\right)$	$O\left(\frac{\sqrt{d}}{\epsilon}\right)$	$O(\min\{n,\frac{1}{\epsilon}\}\frac{d\sqrt{d}}{\sqrt{b}}\log\frac{1}{\epsilon}+md\sqrt{db}\log\frac{1}{\epsilon}$

Table 1: Summary of convergence rate and function query complexity of SZO algorithms. NC: Nonconvex, C: Convex, SC: Strong Convexity, and PL: Polyak-Łojasiewicz Condition.

4.1 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 and Lan (2016) and ZO-ProxSVRG/SAGA Huang et al. (2019). We emphasize that, unlike these methods, we do not assume bounded gradient since it is not the case for many unconstrained optimization problems. In addition, Liu et al. Liu et al. (2018b) have also studied the zeroth-order algorithm for solving the nonconvex nonsmooth problem, which is different from problem (1.1).

4.2 Zeroth-order (ZO) gradient estimators

We next provide a background on ZO gradient estimators. Given an individual cost function f_i, a two-point random stochastic gradient estimator (RandSGE) $\hat{\nabla}_r f_i(x)$ is defined Nesterov and Spokoiny (2017); Gao et al. (2018)

$$\hat{\nabla}_r f_i(x, u_i) = \frac{d(f_i(x+\mu u_i) - f_i(x))}{\mu} u_i, \qquad i \in [n], \tag{2}$$

where recall that d is the number of optimization variables, $\mu > 0$ is a smoothing parameter, and $\{u_i\}$ are i.i.d. random directions drawn from a uniform distribution over a unit sphere Flaxman et al. (2005); Shamir (2017); Gao et al. (2018). In general, RandSGE 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 Lian et al. (2016).

Assume that the function $f(x)$ is L-smooth. $\hat{\nabla} f(x)$ denote the estimated gradient defined by RandSGE. Define $f_\mu = \mathbb{E}_{u\sim N(0,I)}[f(x+\mu u)]$. Then, we have

1) For any $x \in \mathbb{R}^d$, $\nabla f_\mu(x) = \mathbb{E}_u[\hat{\nabla}_r f(x,u)]$.

2) $\left|f_\mu(x) - f(x)\right| \leq \frac{\mu^2 L}{2}$ and $\left\|f_\mu(x) - f(x)\right\| \leq \frac{\beta L d}{2}$ for any $x \in \mathbb{R}^d$.

3) $\mathbb{E}_{u_j}\left\|\hat{\nabla}_r f_j(x,u_j) - \hat{\nabla}_r f_j(y,u_j)\right\|^2 \leq 3dL^2\|x-y\|^2 + \frac{3L^2 d^2 \mu^2}{2}$

To obtain a better estimated gradient, we can use the coordinate gradient estimator (CoordSGE) Gu et al. (2018b,a); Liu et al. (2018b) to estimate the gradients as follows:

$$\hat{\nabla} f_i(x) = \sum_{j=1}^{d} \frac{f_i(x+\mu_j e_j) - f_i(x-\mu_j e_j)}{2\mu_j} e_j, \qquad i \in [n], \tag{3}$$

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. Compared to RandSGE, CoordSGE is deterministic and requires d times more function queries. However, it is evident that it yields an improved convergence rate and iteration complexity Liu et al. (2018b). More details on ZO gradient estimation can be found in Kazemi and Wang (2018).

In this work we only consider CoordSGE and the extension to RandSGE is straightforward.

4.3 Accelerated Proximal Gradient Method

Because proximal gradient method 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 learning 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. (2018b)(Nesterov and Spokoiny, 2017; Liu et al., 2018c) to estimate the gradient only by function values. Based on the estimated gradients (3), we give a zeroth-order proximal gradient descent method, which performs iterations similar to:

$$x_t^s = \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta \hat{\nabla} f(x_{t-1}^s)), \qquad t = 1, 2, \ldots \tag{4}$$

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

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

We also assume that the nonsmooth convex function $h(x)$ in (1.1) is well structured, i.e., the proximal operator (5) on h can be computed efficiently.

5 Accelerated Proximal Gradient Method

In this section, we propose a proximal stochastic gradient algorithm called ZO-PSVRG+, by using VR technique of ProxSVRG in Xiao and Zhang (2014); Reddi et al. (2016b); Li and Li (2018). (similar to nonconvex ZO-ProxSVRG Huang et al. (2019) and ZO-SVRG-Coord Liu et al. (2018b)). The details are described in Algorithm 5. Our

Algorithm 1 ZO-PSVRG+

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

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 b to estimate the gradient. We call B the batch size and b the minibatch size.

Compared with ZO-SVRG-Coord, ZO-ProxSVRG analyzed the nonconvex nonsmooth functions while ZO-SVRG-Coord only analyzed the smooth functions. The major difference of our ZO-PSVRG+ 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 5 of Algorithm 5) while ZO-SVRG-Coord and ZO-ProxSVRG used $B = n$. Note that even if we choose $B = n$, our analysis is stronger than ZO-SVRG-Coord and ZO-ProxSVRG. As a result, our straightforward ZO-PSVRG+ generalizes these variance-reduced methods to a more general nonsmooth nonconvex zeroth-order case and yields simpler analysis. It has been shown in Johnson and Zhang (2013); Reddi et al. (2016a); Li and Li (2018) that the first-order ProxSVRG achieves the convergence rate $O(1/\epsilon)$, yielding $O(1/\epsilon)$ times less iterations than the ordinary SGD for solving finite sum problems. The key step of corresponding variance-reduced algorithmic framework is to generate an auxiliary sequence $\tilde{x}^{s-1}$ at which the full gradient is used as a reference in building a modified stochastic gradient estimate

$$v_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\nabla f_i(x_{t-1}^s) - \nabla f_i(\tilde{x}^{s-1})\right) + g^s \tag{6}$$

where v_{t-1}^s denotes the gradient estimate at x_{t-1}^s. The key property of (6) is that v_{t-1}^s is an unbiased gradient estimate of $\nabla f(x_{t-1}^s)$. In the ZO setting, the gradient blending (6) is approximated using only function values,

$$\hat{v}_{t-1}^s = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x_{t-1}^s) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s \tag{7}$$

where $\hat{g}^s = \frac{1}{B}\sum_{i\in I_B}\hat{\nabla} f_i(\tilde{x}^{s-1})$ and $\hat{\nabla} f_i$ is a ZO gradient estimate specified by CoordSGE. Replacing (6) with (7) in ProxSVRG leads to a new ZO algorithm, which we call ZO-PSVRG+ (Algorithm 5). We also avoid the computation of the full gradient at the beginning of each epoch, i.e., $B \neq n$. 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 ProxSVRG Reddi et al. (2016b); Li and Li (2018) no longer holds. We highlight that although ZO-PSVRG is similar to ProxSVRG except the use of ZO gradient estimators to estimate batch, mini-batch, as well as blended gradients, this seemingly minor differences yield an essential difficulties in the analysis of ZO-PSVRG+. Thus, adapting the similar ideas of ProxSVRG to zeroth-order algorithm 5 is not a trivial task and a careful analysis of ZO-PSVRG+ 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.

5.1 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 (1.1), its high cost per iteration is prohibitive. As a result, Ghadimi and Lan (2016) proposed the RSPGF with calculating the gradient on the mini-batch $\mathcal{I}_t$. Note that from the perspectives of both computational efficiency and statistical generalization, always computing full-gradient may not be desirable for large-scale machine learning problems.

6 Convergence Analysis

Next, we give some mild assumptions regarding problem (1.1) as follows:

Assumption 6.1. 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

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

Assumption 6.2. For $\forall x \in \mathbb{R}^d$, $\mathbb{E}\left[\left\|\hat{\nabla} f_i(x) - \hat{\nabla} f(x)\right\|^2\right] \le \sigma^2$, where $\sigma > 0$ is a constant and $\hat{\nabla} f_i(x)$ is a CoordSGE gradient estimator of $\nabla f_i(x)$.

Both Assumptions 6.1 and 6.2 are the standard assumptions used in nonconvex optimization. The first assumption is used for the convergence analysis of the zeroth-order algorithms Ghadimi and Lan (2016); Nesterov and Spokoiny (2017); Liu et al. (2018b). The second assumption gives the bounded variance of zeroth-order gradient estimates Lian et al. (2016); Liu et al. (2018a,b); Hajinezhad et al. (2017). We emphasize that, unlike Duchi et al. (2015); Ghadimi and Lan (2013); Huang et al. (2019) we do not assume bounded gradient since it is not the case for many unconstrained optimization problems. Note that assumption 6.2 is milder than the assumption of bounded gradients Liu et al. (2017); Hajinezhad et al. (2017). Although, we are able to analyze more complex problem (1.1) including a non-smooth part and drive faster convergence rates. 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 CoordSGE.

Lemma 6.3. *Using CoordSGE given the mixture estimated gradient* $\hat{v}^s_{t-1} = \frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) + \hat{g}^s$ *with* $\hat{g}^s = \frac{1}{B}\sum_{j\in I_B}\hat{\nabla} f_j(\tilde{x}^{s-1})$*, then the following inequality holds.*

$$\begin{aligned}\mathbb{E}\left[\eta\left\|\nabla f(x^s_{t-1}) - \hat{v}^s_{t-1}\right\|^2\right] \leq & \frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x^s_{t-1} - \tilde{x}^{s-1}\right\|^2\right] \\ & + 2\frac{I\{B<n\}\eta\sigma^2}{B} + \eta\frac{L^2 d^2\mu^2}{2}\end{aligned} \tag{8}$$

Proof. We have

$$\begin{aligned}
&\mathbb{E}\left[\eta\left\|\nabla f(x^s_{t-1}) - \hat{v}^s_{t-1}\right\|^2\right] \\
=&\mathbb{E}\left[\eta\left\|\frac{1}{b}\sum_{i\in I_b}\left(\hat{\nabla} f_i(x^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) - \left(\nabla f(x^s_{t-1}) - \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^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) - \left(\nabla f(x^s_{t-1}) - \frac{1}{B}\sum_{j\in I_B}\hat{\nabla} f_j(\tilde{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^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) - \left(\nabla f(x^s_{t-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right) + \left(\frac{1}{B}\sum_{j\in I_B}\hat{\nabla} f_j(\tilde{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^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) - \left(\nabla f(x^s_{t-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right) + \frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla} f_j(\tilde{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^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) - \left(\hat{\nabla} f(x^s_{t-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right) + \frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla} f_j(\tilde{x}^{s-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right\|^2\right] \\
&+ 2\eta\mathbb{E}\left\|\hat{\nabla} f(x^s_{t-1}) - \nabla f(x^s_{t-1})\right\|^2 \qquad (9) \\
=&2\eta\mathbb{E}\left[\left\|\frac{1}{b}\sum_{i\in I_b}\left(\left(\hat{\nabla} f_i(x^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) - \left(\hat{\nabla} f(x^s_{t-1}) - \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(\tilde{x}^{s-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right\|^2\right] + 2\eta\mathbb{E}\left\|\hat{\nabla} f(x^s_{t-1}) - \nabla f(x^s_{t-1})\right\|^2 \qquad (10) \\
=&\frac{2\eta}{b^2}\mathbb{E}\left[\sum_{i\in I_b}\left\|\left(\left(\hat{\nabla} f_i(x^s_{t-1}) - \hat{\nabla} f_i(\tilde{x}^{s-1})\right) - \left(\hat{\nabla} f(x^s_{t-1}) - \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(\tilde{x}^{s-1}) - \hat{\nabla} f(\tilde{x}^{s-1})\right)\right\|^2\right] + 2\eta\mathbb{E}\left\|\hat{\nabla} f(x^s_{t-1}) - \nabla f(x^s_{t-1})\right\|^2 \qquad (11)
\end{aligned}$$

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

$$+2\eta\mathbb{E}\left\|\hat{\nabla} f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{12}$$

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

$$+2\eta\mathbb{E}\left\|\hat{\nabla} f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{13}$$

$$\leq \frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\tilde{x}^{s-1}\right\|^2\right]+2\frac{I\{B<n\}\eta\sigma^2}{B}+2\eta\mathbb{E}\left\|\hat{\nabla} f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{14}$$

$$\leq \frac{2\eta L^2 d}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\tilde{x}^{s-1}\right\|^2\right]+2\frac{I\{B<n\}\eta\sigma^2}{B}+\eta\frac{L^2d^2\mu^2}{2} \tag{15}$$

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

$$+2\eta\mathbb{E}\left\|\hat{\nabla} f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{16}$$

$$\leq \frac{3\eta L^2d^2\mu^2}{b}+\frac{6\eta L^2}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\tilde{x}^{s-1}\right\|^2\right]+2\eta\mathbb{E}\left[\left\|\frac{1}{B}\sum_{j\in I_B}\left(\hat{\nabla} f_j(\tilde{x}^{s-1})-\hat{\nabla} f(\tilde{x}^{s-1})\right)\right\|^2\right]$$

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

$$\leq \frac{6\eta L^2}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\tilde{x}^{s-1}\right\|^2\right]+2\frac{I\{B<n\}\eta\sigma^2}{B}+2\eta\mathbb{E}\left\|\hat{\nabla} f(x_{t-1}^s)-\nabla f(x_{t-1}^s)\right\|^2 \tag{18}$$

$$+\frac{3\eta L^2d^2\mu^2}{b} \tag{19}$$

$$\leq \frac{6\eta L^2}{b}\mathbb{E}\left[\left\|x_{t-1}^s-\tilde{x}^{s-1}\right\|^2\right]+2\frac{I\{B<n\}\eta\sigma^2}{B}+\eta\frac{7L^2d^2\mu^2}{2} \tag{20}$$

where, recalling that a deterministic gradient estimator is used, the expectations are taking with respect to I_b and I_B. The inequality (9) holds by the Jensen's inequality. (10) and (11) 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). (16) uses the fact that $\mathbb{E}[\|x-\mathbb{E}[x]\|^2]\leq\mathbb{E}[\|x\|^2]$, for any random variable x. (17) holds due to the following inequality

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

$$\begin{aligned}\mathbb{E}\left\|\hat{\nabla} f_i(x_t^s) - \hat{\nabla} f_i(\tilde{x}^s)\right\|^2 &= \mathbb{E}\left\|\hat{\nabla} f_i(x_t^s) - \nabla f_i(x_t^s) + \nabla f_i(x_t^s) - \hat{\nabla} f_i(\tilde{x}^s) + \nabla f_i(\tilde{x}^s) - \nabla f_i(\tilde{x}^s)\right\|^2 \\ &\le 3\mathbb{E}\left\|\hat{\nabla} f_i(x_t^s) - \nabla f_i(x_t^s)\right\|^2 + 3\left\|\hat{\nabla} f_i(\tilde{x}^s) - \nabla f_i(\tilde{x}^s)\right\|^2 \\ &+ 3\|\nabla f_i(x_t^s) - \nabla f_i(\tilde{x}^s)\|^2 \\ &\le \frac{3L^2 d^2 \mu^2}{2} + 3L^2 \|x_t^s - \tilde{x}^s\|^2\end{aligned} \tag{22}$$

where the last inequality used the fact that f_{i,μ_j} is L-smooth. (18) is by Assumption 6.2 and (20) uses Lemma 9.2. The proof is now complete. □

Lemma 6.3 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 $\tilde{x}^{s-1}$ 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

$$\|x - z\|^2 \le (1 + \frac{1}{\beta})\,\|x - y\|^2 + (1 + \beta)\,\|y - z\|^2, \forall \beta > 0 \tag{23}$$

6.1 Gradient Mapping

For convex problems, one typically uses the optimality gap $F(x) - F(x^*)$ as the convergence criterion. 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); Lei et al. (2017); Liu et al. (2018b) 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 and Lan (2016); Reddi et al. (2016b); Huang et al. (2019)):

$$g_\eta(x) = \frac{1}{\eta}(x - \mathrm{Prox}_{\eta,h}(x - \eta \nabla f(x))) \tag{24}$$

Note that if $h(x)$ is a constant function (in particular, zero), this gradient mapping reduces to the ordinary gradient: $g_\eta(x) = \nabla F(x) = \nabla f(x)$. In this paper, we use the gradient mapping $g_\eta(x)$ as the convergence criterion (same as Ghadimi and Lan (2016); Reddi et al. (2016b); Parikh et al. (2014)). For the nonconvex problems, if $g_\eta(x) = 0$, the point x is a critical point (Parikh, Boyd, and others, 2014). Thus, we can use the following definition as the convergence metric.

Definition 6.4. Solution x is called ϵ-accurate, if $\mathbb{E}\left\|g_\eta(x)\right\|^2 \leq \epsilon$, for some $\eta > 0$.

6.2 Convergence

In Theorem 6.5, we focus on the effect of CoordSGE on the convergence rate of ZO-PSVRG+ and give some remarks.

Theorem 6.5. *Suppose Assumptions 6.1 and 6.2 hold, and the coordinate gradient estimator CoordSGE is used. The output $\hat{x}$ of Algorithm 5 satisfies*

$$\mathbb{E}[\left\|g_\eta(\hat{x})\right\|^2] \leq \frac{6(F(x_0) - F(x^*))}{\eta S m} + \frac{I\{B < n\}12\sigma^2}{B} + 3L^2 d^2 \mu^2 \tag{25}$$

where $\eta = \min\{\frac{1}{8L}, \frac{\sqrt{b}}{6mL\sqrt{d}}\}$ denotes the step size and x^* denotes the optimal value of problem 1.1.

Proof. Now, we apply Lemma 9.3 to prove Theorem 6.5. Let $x_t^s = \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta \hat{v}_{t-1}^s)$ and $\overline{x}_t^s = \mathrm{Prox}_{\eta h}(x_{t-1}^s - \eta \nabla f(x_{t-1}^s))$. By letting $x^+ = x_t^s$, $x = x_{t-1}^s$, $v = \hat{v}_{t-1}^s$ and $z = \overline{x}_t^s$ in (59), we have

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

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 (59), we have

$$\begin{aligned} F(\overline{x}_t^s) \leq& F(x_{t-1}^s) - \frac{1}{\eta}\left\langle \overline{x}_t^s - x_{t-1}^s, \overline{x}_t^s - x_{t-1}^s \right\rangle + \frac{L}{2}\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2 \\ =& F(x_{t-1}^s) - (\frac{1}{\eta} - \frac{L}{2})\left\|\overline{x}_t^s - x_{t-1}^s\right\|^2. \end{aligned} \tag{27}$$

Combining (26) and (27) we have

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

$$
\begin{aligned}
=&F(x^s_{t-1}) - (\frac{1}{2\eta} - \frac{L}{2}) \left\| x^s_t - x^s_{t-1} \right\|^2 - \left(\frac{1}{2\eta} - L\right) \left\| \overline{x}^s_t - x^s_{t-1} \right\|^2 + \left\langle \nabla f(x^s_{t-1}) - \hat{v}^s_{t-1}, x^s_t - \overline{x}^s_t \right\rangle \\
&- \frac{1}{2\eta} \| x^s_t - \overline{x}^s_t \|^2 \\
\leq& F(x^s_{t-1}) - (\frac{1}{2\eta} - \frac{L}{2}) \left\| x^s_t - x^s_{t-1} \right\|^2 - \left(\frac{1}{2\eta} - L\right) \left\| \overline{x}^s_t - x^s_{t-1} \right\|^2 + \left\langle \nabla f(x^s_{t-1}) - \hat{v}^s_{t-1}, x^s_t - \overline{x}^s_t \right\rangle \\
&- \frac{1}{8\eta} \left\| x^s_t - x^s_{t-1} \right\|^2 + \frac{1}{6\eta} \left\| \overline{x}^s_t - x^s_{t-1} \right\|^2 \qquad (28) \\
=&F(x^s_{t-1}) - (\frac{5}{8\eta} - \frac{L}{2}) \left\| x^s_t - x^s_{t-1} \right\|^2 - \left(\frac{1}{3\eta} - L\right) \left\| \overline{x}^s_t - x^s_{t-1} \right\|^2 + \left\langle \nabla f(x^s_{t-1}) - \hat{v}^s_{t-1}, x^s_t - \overline{x}^s_t \right\rangle \\
\leq& F(x^s_{t-1}) - (\frac{5}{8\eta} - \frac{L}{2}) \left\| x^s_t - x^s_{t-1} \right\|^2 - \left(\frac{1}{3\eta} - L\right) \left\| \overline{x}^s_t - x^s_{t-1} \right\|^2 + \eta \left\| \nabla f(x^s_{t-1}) - \hat{v}^s_{t-1} \right\|^2 \\
&\qquad (29)
\end{aligned}
$$

where the second inequality uses (23) with $\beta = 3$ and the last inequality holds due to the Lemma 9.4.

Note that $x^s_t = \text{Prox}_{\eta h}(x^s_{t-1} - \eta \hat{v}^s_{t-1})$ is the iterated form in our algorithm. By taking the expectation with respect to all random variables in (29) we obtain

$$
\mathbb{E}[F(x^s_t)] \leq \mathbb{E}\left[F(x^s_{t-1}) - (\frac{5}{8\eta} - \frac{L}{2}) \left\| x^s_t - x^s_{t-1} \right\|^2 - \left(\frac{1}{3\eta} - L\right) \left\| \overline{x}^s_t - x^s_{t-1} \right\|^2 + \eta \left\| \nabla f(x^s_{t-1}) - \hat{v}^s_{t-1} \right\|^2 \right] \qquad (30)
$$

In (30), we further bound $\eta \left\| \nabla f(x^s_{t-1}) - \hat{v}^s_{t-1} \right\|^2$ using Lemma 6.3 to obtain

$$
\begin{aligned}
&\mathbb{E}[F(x^s_t)] \\
\leq& \mathbb{E}\left[F(x^s_{t-1}) - (\frac{5}{8\eta} - \frac{L}{2}) \left\| x^s_t - x^s_{t-1} \right\|^2 - \left(\frac{1}{3\eta} - L\right) \left\| \overline{x}^s_t - x^s_{t-1} \right\|^2 \right] \\
&+ \frac{2\eta L^2 d}{b} \mathbb{E} \left\| x^s_{t-1} - \tilde{x}^{s-1} \right\|^2 + \frac{2I\{B < n\}\eta\sigma^2}{B} + \eta \frac{L^2 d^2 \mu^2}{2} \\
=& \mathbb{E}\left[F(x^s_{t-1}) - (\frac{5}{8\eta} - \frac{L}{2}) \left\| x^s_t - x^s_{t-1} \right\|^2 - \left(\frac{\eta}{3} - L\eta^2\right) \left\| g_\eta(x^s_{t-1}) \right\|^2 \right] \\
&+ \frac{2\eta L^2 d}{b} \mathbb{E} \left\| x^s_{t-1} - \tilde{x}^{s-1} \right\|^2 + \frac{2I\{B < n\}\eta\sigma^2}{B} + \eta \frac{L^2 d^2 \mu^2}{2} \qquad (31) \\
\leq& \mathbb{E}\left[F(x^s_{t-1}) - \frac{1}{2t}(\frac{5}{8\eta} - \frac{L}{2}) \left\| x^s_t - \tilde{x}^{s-1} \right\|^2 - \left(\frac{\eta}{3} - L\eta^2\right) \left\| g_\eta(x^s_{t-1}) \right\|^2 \right] \\
&+ (\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta} - \frac{L}{2})) \mathbb{E} \left\| x^s_{t-1} - \tilde{x}^{s-1} \right\|^2 + \frac{2I\{B < n\}\eta\sigma^2}{B} + \eta \frac{L^2 d^2 \mu^2}{2} \qquad (32)
\end{aligned}
$$

where recalling $\overline{x}^s_t := \text{Prox}_{\eta h}(x^s_{t-1} - \eta \nabla f(x^s_{t-1}))$, (31) is based on the definition of gradient mapping $g_\eta(x^s_{t-1})$. (32) uses (23) by choosing $\beta = 2t - 1$.

Taking a telescopic sum for $t = 1, 2, \ldots, m$ in epoch s from (32) and recalling that $x^s_m = \tilde{x}^s$ and $x^s_0 = \tilde{x}^{s-1}$, we obtain

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

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

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

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

$$
\leq \mathbb{E}\left[F(\tilde{x}^{s-1}) - \frac{\eta}{6} \sum_{t=1}^{m} \left\|g_\eta(x_{t-1}^s)\right\|^2\right] + \sum_{t=1}^{m} \frac{2I\{B < n\}\eta\sigma^2}{B} + \sum_{t=1}^{m} \eta \frac{L^2 d^2 \mu^2}{2} \tag{34}
$$

where (33) holds since norm is always non-negative and $x_0^s = \tilde{x}^{s-1}$. In (34) we have used the fact that $(\frac{1}{6t^2}(\frac{5}{8\eta} - \frac{L}{2}) - \frac{2\eta L^2 d}{b}) \geq 0$ for all $1 \leq t \leq m$ and $\frac{\eta}{6} \leq \frac{\eta}{3} - L\eta^2$ since $\eta = \min\{\frac{1}{8L}, \frac{\sqrt{b}}{6mL\sqrt{d}}\}$. Telescoping the sum for $s = 1, 2, \ldots, S$ in (34), we obtain

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

Thus, we have

$$\mathbb{E}[\left\|g_\eta(\hat{x})\right\|^2] \le \frac{6\left(F(x_0)-F(x^*)\right)}{\eta Sm} + \frac{I\{B<n\}12\sigma^2}{B} + 3L^2d^2\mu^2 \tag{35}$$

where (35) holds since we choose $\hat{x}$ uniformly randomly from $\{x_{t-1}^s\}_{t\in[m],s\in[S]}$. □

The proof for Theorem 6.5 is notably different from that of ZO-SVRG-Coord and ZO-ProxSVRG/SAGA 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 6.3 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., $B \neq n$. (25) 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 Reddi et al. (2016b), Theorem 6.5 exhibits two additional errors $\frac{I\{B<n\}\sigma^2}{B}$ and $O(L^2d^2\mu^2)$ due to batch gradient estimation $B<n$ and the use of SZO gradient estimates, 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 (25) 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 5 becomes $\frac{1}{B}\sum_{i\in I_B}\nabla f_i(\tilde{x}_{t-1}^s) = \nabla f(\tilde{x}_{t-1}^s)$. 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 6.5, Corollary 6.6 provides the convergence rate of ZO-PSVRG+ in terms of precision at the solution $\hat{x}$ and simplifies (25) for a specific parameter setting, as formalized below.

Corollary 6.6. *We set the batch size $B=\min\{12\sigma^2/\epsilon, n\}$ and the smoothing parameter $\mu \le \frac{\sqrt{\epsilon}}{3\sqrt{dL}}$. Suppose $\hat{x}$ returned by Algorithm 5 is an ϵ-accurate solution for problem (1.1). Recalling that CoordSGE require $O(d)$ function queries, the number of SZO calls is at most*

$$d(SB+Smb) = 6d\left(F(x_0)-F(x^*)\right)\left(\frac{B}{\epsilon\eta m} + \frac{b}{\epsilon\eta}\right) = O\left(\frac{Bd}{\epsilon\eta m} + \frac{bd}{\epsilon\eta}\right). \tag{36}$$

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

$$36dL(F(x_0)-F(x^*))\left(\frac{B\sqrt{d}}{\epsilon\sqrt{b}} + \frac{b\sqrt{d}}{\epsilon}\right) = O\left(s_n\frac{d\sqrt{d}}{\epsilon\sqrt{b}} + \frac{bd\sqrt{d}}{\epsilon}\right). \tag{37}$$

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

Proof. Using Theorem 6.5, we have $\eta = \min\{\frac{1}{8L}, \frac{\sqrt{b}}{6mL\sqrt{d}}\}$

$$\mathbb{E}[\left\|g_\eta(\hat{x})\right\|^2] \leq \frac{6\,(F(x_0)-F(x^*))}{\eta Sm} + \frac{I\{B<n\}12\sigma^2}{B} + 3L^2d^2\mu^2 = 3\epsilon \tag{38}$$

Now we obtain the total number of iterations $T = Sm = \frac{6(F(x_0)-F(x^*))}{\epsilon\eta}$. Since $\mu \leq \frac{\sqrt{\epsilon}}{3\sqrt{d}L}$, and for $B = n$, the second term in the bound (38) is 0, the proof is finished as the number of SFO call equals to $Sn + Smb = 6\,(F(x_0)-F(x^*))\,(\frac{n}{\epsilon\eta m} + \frac{b}{\epsilon\eta})$. If $B < n$ the number of SZO calls equal to $d(SB + Smb) = 6d\,(F(x_0)-F(x^*))\,(\frac{B}{\epsilon\eta m} + \frac{b}{\epsilon\eta})$ by noting that $\frac{I\{B<n\}12\sigma^2}{B} \leq \epsilon$ due to $B \geq 12\sigma^2/\epsilon$. The second part of corollary is obtained by setting $m = \sqrt{b}$ in the first part. □

Roughly speaking, Corollary 6.6 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 5 of Algorithm 5 becomes $\hat{g}^s = \hat{\nabla} f(\tilde{x}^{s-1})$.

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 SZO methods by factor of $\frac{1}{\sqrt{d}}$. Moreover, ZO-PSVRG+ uses much less SZO oracle which is detailed in Table 4.

It is worth mentioning that the condition on the value of step size in Theorem 6.5 is less restrictive than several SZO algorithms in Table 4. 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 6.6 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 (37) that the SZO complexity is increased by a factor $\sqrt{b}$, which is smaller than the size of the minibatch. However, the corresponding complexity of RGF is increased by multiplying a factor of b (see Table 4), so our algorithm has a better dependency to the mini-batch size in this special case.

Further, 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 SZO optimization methods.

7 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). The original form of PL condition is

$$\exists\lambda > 0, \text{such that } \|\nabla f(x)\|^2 \geq 2\lambda(f(x) - f^*),\ \forall x, \tag{39}$$

where f^* denotes the (global) optimal function value. It is worth noting that f satisfies PL condition when f is λ-strongly convex. This condition specifies how fast the objective function grows in a local neighborhood of optimal solutions. In particular, we propose a generic convergence framework for accelerating existing SZO algorithms for functions satisfying PL settings by leveraging variance reduced methods. This is accomplished by a novel synthesis of existing SZO algorithms. We show the iteration complexity of ZO-PSVRG+ (Algorithm 5) is improved by applying PL condition .

Due to the nonsmooth term $h(x)$ in problem (1.1), we use the gradient mapping to define a more general form of PL condition as follows

$$\exists\lambda > 0, \text{such that } \left\|g_\eta(x)\right\|^2 \geq 2\lambda(F(x) - F^*),\ \forall x. \tag{40}$$

Recall that if $h(x)$ is a constant function, the gradient mapping reduces to $g_\eta(x) = \nabla f(x)$. Note that the PL condition has been studied thoroughly in Karimi et al. (2016). The authors show that PL condition is weaker than broad family of conditions. For example, when $f(x)$ is convex, quadratic growth condition holds Luo and Tseng (1993); Anitescu (2000).

The PL condition (40) is arguably natural and considered in several existing works Li and Li (2018). Similar to Theorem 6.5, we provide the convergence result of ZO-PSVRG+ (Algorithm 5) under PL-condition in the following Theorem 7.1. Note that under PL condition (i.e. (40) holds), ZO-PSVRG+ can directly use the final iteration $\tilde{x}^S$ as the output point instead of the randomly chosen one $\hat{x}$.

Theorem 7.1. *Let Assumptions 6.1 and 6.2 hold, and CoordSGE is used in Algorithm 5 with step size* $\eta \leq \min\{\frac{1}{8L}, \frac{\sqrt{\gamma b}}{5mL\sqrt{d}}\}$ *where* $\gamma = 1 - \frac{2\lambda\eta}{3}m - \frac{\lambda\eta}{3} > 0$. *Then*

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

where x^* is same as Theorem 6.5.

Proof. We start by recalling inequality (31) from the proof of Theorem 6.5, i.e.,

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

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

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

into (42), we obtain

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

Thus, we have

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

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

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

Similar to the proof of Theorem 6.5, 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

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

$$\leq \alpha^m \mathbb{E}\left[(F(\tilde{x}^{s-1}) - F^*)\right] + \frac{1-\alpha^m}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1-\alpha^m}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}$$
$$-\alpha^m\mathbb{E}\left[\sum_{t=1}^{m-1}\frac{1}{2t\alpha^t}(\frac{5}{8\eta}-\frac{L}{2})\left\|x_t^s - \tilde{x}^{s-1}\right\|^2\right]$$
$$+\alpha^m\mathbb{E}\left[\sum_{t=2}^{m}\frac{1}{\alpha^t}(\frac{2\eta L^2 d}{b} + \frac{1}{2t-1}(\frac{5}{8\eta}-\frac{L}{2}))\left\|x_{t-1}^s - \tilde{x}^{s-1}\right\|^2\right] \tag{47}$$
$$\leq \alpha^m \mathbb{E}\left[(F(\tilde{x}^{s-1}) - F^*)\right] + \frac{1-\alpha^m}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1-\alpha^m}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}$$
$$-\alpha^m\mathbb{E}\left[\sum_{t=1}^{m-1}\frac{1}{\alpha^{t+1}}\left((\frac{\alpha}{2t}-\frac{1}{2t+1})(\frac{5}{8\eta}-\frac{L}{2})-\frac{2\eta L^2 d}{b})\right)\left\|x_t^s - \tilde{x}^{s-1}\right\|^2\right]$$
$$\leq \alpha^m \mathbb{E}\left[(F(\tilde{x}^{s-1}) - F^*)\right] + \frac{1-\alpha^m}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1-\alpha^m}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2} \tag{48}$$

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 $t = 1, 2, \ldots, m$. It is easy to see that this inequality holds since $\eta \leq \min\{\frac{1}{8L}, \frac{\sqrt{\gamma b}}{5mL\sqrt{d}}\}$, where $\gamma = 1 - 2\beta m - \beta > 0$. Similarly, let $\tilde{\alpha} = \alpha^m$ and $\tilde{\Psi}^s = \frac{\mathbb{E}[F(\tilde{x}^s) - F^*]}{\tilde{\alpha}^s}$. Substituting these definitions into (48), we have

$$\tilde{\Psi}^s \leq \tilde{\Psi}^{s-1} + \frac{1}{\tilde{\alpha}^s}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{I\{B<n\}\eta\sigma^2}{B} + \frac{1}{\tilde{\alpha}^s}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{Ld\mu^2}{12} \tag{49}$$

Taking a telescopic sum from (49) for all epochs $1 \leq s \leq S$, we obtain

$$\mathbb{E}[F(\tilde{x}^S) - F^*] \leq \tilde{\alpha}^S\mathbb{E}[F(\tilde{x}^0) - F^*] + \tilde{\alpha}^S\sum_{s=1}^{S}\frac{1}{\tilde{\alpha}^s}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \tilde{\alpha}^S\sum_{s=1}^{S}\frac{1}{\tilde{\alpha}^s}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}$$
$$= \alpha^{Sm}\mathbb{E}[F(\tilde{x}^0) - F^*] + \frac{1-\tilde{\alpha}^S}{1-\tilde{\alpha}}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1-\tilde{\alpha}^S}{1-\tilde{\alpha}}\frac{1-\tilde{\alpha}}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}$$
$$\leq \alpha^{Sm}\mathbb{E}[F(\tilde{x}^0) - F^*] + \frac{1}{1-\alpha}\frac{2I\{B<n\}\eta\sigma^2}{B} + \frac{1}{1-\alpha}\frac{\eta L^2 d^2\mu^2}{2}$$
$$= \left(1 - \frac{\lambda\eta}{3}\right)^{Sm}\mathbb{E}[F(\tilde{x}^0) - F^*] + \frac{6I\{B<n\}\sigma^2}{\lambda B} + \frac{3L^2 d^2\mu^2}{2\lambda} \tag{50}$$

where in (50) we recall that $\alpha = 1 - \frac{\lambda\eta}{3}$. □

Theorem 7.1 shows that if the batch size and smoothing parameter are appropriately chosen, ZO-PSVRG+ has a dominant linearly decaying convergence rate.

Further, by comparing with Theorem 6.5, it is evident from (41) that the common error term $\frac{6I\{B<n\}\sigma^2}{B} + \frac{3L^2d^2\mu^2}{2}$ is amplified through multiple $1/\lambda$. Thus, the error induced by these terms ceases to be significantly improved if $\lambda >> 1$. We next study the number of oracle calls in ZO-PSVRG+ under PL condition to obtain an ϵ-accurate solution, as formalized in Corollary 7.2.

Corollary 7.2. *Suppose the final iteration point $\tilde{x}^S$ in Algorithm 5 satisfies $\mathbb{E}[F(\tilde{x}^S) - F^*] \leq \epsilon$ under PL condition. Under Assumptions 6.1 and 6.2, we let batch size $B = \min\{\frac{6\sigma^2}{\lambda\epsilon}, n\}$ and the smoothing parameter $\mu \leq \frac{\epsilon}{2Ld\lambda}$. The number of SZO 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

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

In particular, given the setting $m = \sqrt{b}$ and $\eta = \frac{\sqrt{\gamma}}{5L\sqrt{d}}$, the number of SZO 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 7.1, we have

$$\mathbb{E}[F(\tilde{x}^S) - F^*] \leq \left(1 - \frac{\lambda\eta}{3}\right)^{Sm}\mathbb{E}[F(\tilde{x}^0) - F^*] + \frac{6I\{B < n\}\sigma^2}{\lambda B} + \frac{3L^2d^2\mu^2}{2\lambda} = 3\epsilon \quad (51)$$

which gives the total number of iterations $T = Sm = O(\frac{1}{\lambda\eta}\log\frac{1}{\epsilon})$ and equals to the number of PO calls. Since $\mu \leq \frac{\sqrt{2\lambda\epsilon}}{Ld}$, we have $\frac{3L^2d^2\mu^2}{2\lambda} \leq \epsilon$. The number of SZO calls equals to $d(SB+Smb) = O(\frac{Bd}{\lambda\eta m}\log\frac{1}{\epsilon} + \frac{bd}{\lambda\eta}\log\frac{1}{\epsilon})$. Note that if $B < n$ then $\frac{6I\{B<n\}\sigma^2}{\lambda B} \leq \epsilon$ since $B \geq 6\sigma^2/\lambda\epsilon$. With $m = \sqrt{b}$ and $\eta = \frac{\sqrt{\gamma}}{5L\sqrt{d}}$, the number of PO calls equals to $T = Sm = O(\frac{B\sqrt{d}}{\lambda\sqrt{\gamma}}\log\frac{1}{\epsilon})$ and the number of SZO calls equals to $d(SB + Smb) = O(\frac{Bd\sqrt{d}}{\lambda\sqrt{\gamma}m}\log\frac{1}{\epsilon} + \frac{bd\sqrt{d}}{\lambda\sqrt{\gamma}}\log\frac{1}{\epsilon})$. □

Corollary 7.2 shows that the use of PL condition improves the dominant convergence rate, where the error of order $O(d/\epsilon)$ in Corollary 6.6 improves to $O(\sqrt{d}\log(1/\epsilon))$, resulting in a significant speed up. Compared to the aforementioned ZO algorithms Duchi et al. (2015); Nesterov and Spokoiny (2017); Liu et al. (2018b), the convergence performance of ProxSVRG+ under PL condition has a global linear rather than sublinear convergence rate and therefore uses much less ZO oracle calls.

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 $b < n^{2/3}$.

Remark 7.3. Compared to the convergence rate of ZO-PSVRG+ as given in Theorem 6.5, Theorem 7.1 exhibits additional parameter γ for parameter selection due to the use of PL condition. If we assume the condition number $\lambda/L \leq \frac{1}{n^{1/3}}$ and choose $m = n^{1/3}$ and $\rho \leq \frac{1}{2}$, then the definition of γ yields

$$\begin{aligned}\gamma &= 1 - \frac{2\lambda\eta}{3}m - \frac{\lambda\eta}{3} \\ &= 1 - \frac{2\lambda\rho}{3L}m - \frac{\lambda\rho}{3L} \\ &\geq 1 - \frac{2\rho}{3n^{1/3}}m - \frac{\rho}{3n^{1/3}} \\ &\geq 1 - \rho \geq \frac{1}{2} \end{aligned} \tag{52}$$

According to Theorem 7.1, equation (52) implies $\eta \leq \min\{\frac{1}{8L}, \frac{\sqrt{b}}{5\sqrt{2}mL\sqrt{d}}\}$. Hence, choosing $b = n^{2/3}$ leads to the constant step size $\eta = \frac{1}{10L\sqrt{d}}$. Note that the assumption $\lambda/L \leq \frac{1}{n^{1/3}}$ is milder than the assumption $\lambda/L < \frac{1}{\sqrt{n}}$ in Reddi et al. (2016b) for condition number.

8 Experimental results

We present our empirical results in this section. We evaluate the following variance reduction algorithms for our experiments: 1) ZO-ProxSVRG **?**, 2) ZO-ProxSVRG**?** and 3) ZO-ProxSGD on two applications: black-box binary classification and ad- versarial attacks on black-box deep neural networks (DNNs). Note that the ZO-ProxSGD is obtained by combining RSPGF and CoordSGE (3) for gradient estimation. It is shown in Huang et al. (2019) that the stochastic gradiet method using CoordSGE shows better performance than counterparts based on RandSGE.

8.1 Black-Box Binary Classification

For the first set of our experiments, we study logistic regression loss function with L_1 and L_2 regularization to learn the black-box binary classification problem. The problem can be formulated as optimization problem (1.1) with $f_i(x) = \log(1 + e^{-y_i z_i^T x})$, $h(x) = \lambda_1 \|x\|_1 + \frac{\lambda_2}{2}\|x\|^2$, where $z_i \in \mathbb{R}^d$ and y_i is the corresponding label for each i. The L_1 regularization and L_2 regularization weights λ_1 and λ_2 are set 10^{-4} in all the experiments. The learning rates are tuned for competitive algorithms in our experiments, and the results shown in this section are based on the best learning rate for each algorithm we achieved.

We run our experiments on datasets from LIBSVM website[1], as shown in Table 2. The epoch size m is chosen as 50 in all our experiments and we fix the minibatch size to b =?. In ZO-PSVRG+, we set step sizes η and μ to satisfy our assumptions in lemmas and theorems with $\eta \sim O(\frac{1}{\sqrt{d}})$.

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

Table 2: Summary of training datasets.

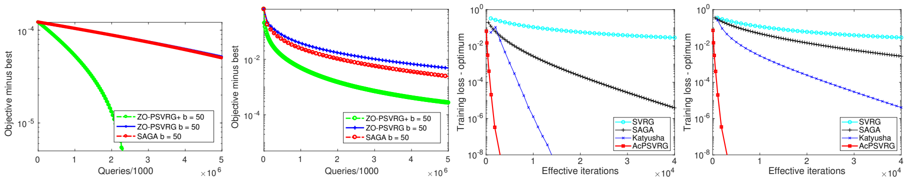
Datasets	Data	Features	Non-zeros
ijcnn	49990	22	455,000
a9a	32561	123	7,521,450
w8a	64,700	300	2
mnist	60000	784	50,233,657

In Fig. **??**, we present the training loss against the number of epochs (i.e., iterations divided by the epoch length $m = 50$) and function queries. Since the computation complexity of each epoch of these algorithms is different, in Figure 2 (bottom) we directly plot the objective value versus the number of zeroth order queries of these algorithms. Results in Figure (**??**) compare the performance of ZO-PSVRG+ with the variants of variance reduction stochastic gradient descent described earlier in this section. Fig. 2-(a) presents the convergence trajectories of ZO algorithms as functions of the number of epochs, where ZO-SVRG is evaluated under different mini-batch sizes $b \in \{1, 10, 40\}$. We observe that the convergence error of ZO-PSVRG+ decreases as b increases, and for a small mini-batch size $b \leq 10$, ZO-PSVRG+ likely converges to a neighborhood of a critical point as shown by Corollary **??**. We also note that our proposed algorithms ZO-PSVRG+($b = 40$) have faster convergence speeds (i.e., less iteration complexity) than the existing variance reduced algorithms ZO-ProxSVRG and ZO-ProxSAGA. It is seen that the performance of ZO-PSVRG+ outperforms other variants of SGD methods in all cases. Figures **??** and **??** show that both objective values and test losses of the proposed methods faster decrease than the ZO-ProxSGD method, as the time increases. In particular ZO-PSVRG+ shows better performances than both ZO-ProxSVRG and ZO-ProxSAGA using the CoordSGE . From these results, we find that the ZO-PSVRG+ shows better performances. Moreover, these results also demonstrate that both the ZO-ProxSVRG and ZO-ProxSAGA using the CoordSGE have a relatively faster convergence rate than the counterparts using the GauSGE. It is observed that ZO-PSVRG+ always exhibits better convergence than other ZO-ProxSGD variants. In particular, compared to ZO-ProxSVRG, as an alternative ZO method, ZO-PSVRG+ shows significantly better convergence. As seen in the figure, ZO-PSVRG+ outperforms variance-reduced ZO-ProxSGD algorithms in all the cases. In particular compared to ZO-ProxSVRG and ZO-ProxSAGA, as ZO-PSVRG+ uses only a batch of size $B < n$ for calculation of full gradient, it has lower complexity per iteration and so shows better convergence speed. Particularly, the use of $B = \min\{n, \frac{1}{\epsilon}\}$ in ZO-PSVRG+ significantly accelerates ZO-PSVRG+ since the ZO-queries of order $O(n)$ is reduced to $O(\min\{n, \frac{1}{\epsilon}\})$ (see Table **??**), leading to a non-dominant factor $O(I_{\{B<n\}}/B)$ in the convergence rate of ZO-PSVRG+.

Fig. 2-(b) presents the training loss against the number of function queries. For the same experiment, Table **??** shows the number of iterations and the testing error of algorithms studied in Fig. **??**-(b) using 7.3×10^6 function queries. We observe that the performance of ZO-ProxSVRG degrades due to the need of large number of function queries to construct coordinate-wise gradient estimates. By contrast, ZO-PSVRG+

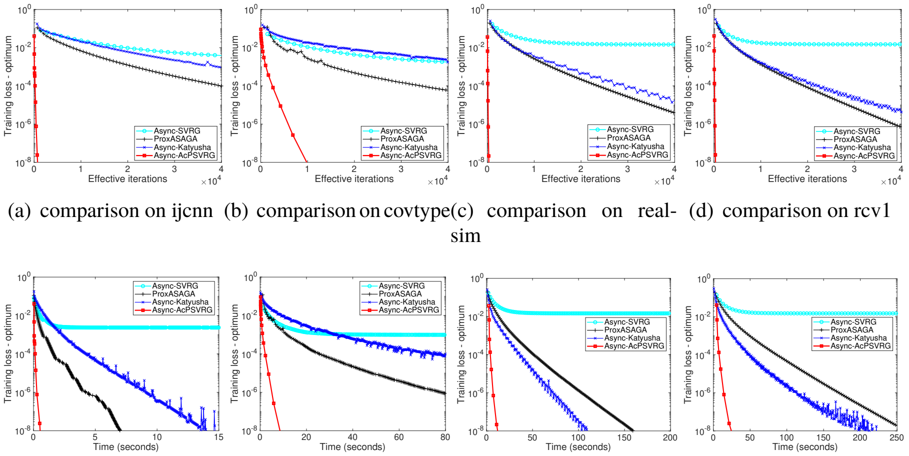
yields better training results, while ZO-ProxSGD consumes an extremely large number of iterations (14600 epochs). As a result, ZO-PSVRG+ ($b = 40$) achieve better tradeoffs between the iteration and the function query complexity.

Since the ZO-ProxSAGA has less function query complexity than the ZO-ProxSVRG, it shows the better performances than the ZO-ProxSVRG. For example, the ZO-ProxSVRG- CooSGE needs $O(ndS + bdT)$ function queries, while ZO- SAGA-CoordSGE needs $O(bdT)$ function queries.



(a) comparison on ijcnn (b) comparison on covtype (c) comparison on real-sim (d) comparison on rcv1

Figure 1: Convergence performance of SVRG, SAGA, Katyusha, and AcPSVRG with single worker.



(a) comparison on ijcnn (b) comparison on covtype (c) comparison on real-sim (d) comparison on rcv1

(e) comparison on ijcnn (f) comparison on covtype (g) comparison on real-sim (h) comparison on rcv1

Figure 2: Training loss residual $f(x) - f(x^*)$ versus iteration (top) and time (bottom) plot of Async-SVRG, ProxASAGA, Async-Katyusha, and Async-AcPSVRG.

8.2 Adversarial Attacks on Black-Box DNNs

In image classification, adversarial examples refer to carefully crafted perturbations such that, when added to the natural images, are visually imperceptible but will lead the target model to misclassify. In the setting of "zeroth order" attacks [2, 3, 35], the model parameters are hidden and acquiring its gradient is inadmissible. Only the model evaluations are accessible. We can then regard the task of generating a universal adversarial perturbation (to n natural images) as an ZO optimization problem of the form (1). More exactly, we use the zeroth-order algorithms to find an universal adversarial perturbation $x \in \mathbb{R}^d$ that could fool the samples $\{a_i \in \mathbb{R}^d, l_i \in \mathbb{N}\}_{i=1}^n$, which can be specified as the following elastic-net attacks to black-box DNNs problem:

$$\min_{x \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} \max\{F_{l_i}(a_i + x) - \max_{j \neq l_i} F_j(a_i + x), 0\} + \lambda_1 \|x\|_1 + \lambda_2 \|x\|^2 \tag{53}$$

where λ_1 and λ_2 are nonnegative parameters to balance attack success rate, distortion and sparsity. Here $F(a) = [F_1(a), \ldots, F_K(a)] \in [0,1]^K$ represents the final layer output of neural network, which is the probabilities of K classes.

We use a well-trained DNN 7 on the MNIST handwritten digit classification task as the target black- box model, which achieves 99.4% test accuracy on natural examples. Two ZO optimization methods, ZO-SGD and ZO-SVRG-Ave, are performed in our experiment. Note that ZO-SVRG-Ave reduces to ZO-SVRG when $q = 1$. We choose $n = 10$ images from the same class, and set the same parameters $b = 5$ and constant step size $30/d$ for both ZO methods, where $d = 28 \times 28$ is the image dimension. For ZO-SVRG-Ave, we set $m = 10$ and vary the number of random direction samples $q \in \{10, 20, 30\}$.

In addition, we set $\lambda_1 = 10^{-3}$ and $\lambda_2 = 1$ in the experiment.

Figure **??** shows that both objective values and black-box attack losses (i.e. the first part of the problem (**??**)) of the proposed algorithms faster decrease than the RSPGF method, as the number of iteration increases. Here, we add ZO-ProxSGD-CooSGE method for comparison, which is obtained by combining the ZO-ProxSGD method with CooSGE. Interestingly, ZO-PSVRG+ shows better performance than both ZO-ProxSVRG and ZO-ProxSAGA. Although having a relatively good performance in generating the adversarial samples, ZO-ProxSGD still shows worse performance than both the ZO-ProxSVRG and ZO-ProxSAGA, due to not using the VR technique.

In Fig. **??**, we show the black-box attack loss (against the number of epochs) as well as the least l_2 distortion of the successful (universal) adversarial perturbations. To reach the same attack loss (e.g., 7 in our example), ZO-SVRG-Ave requires roughly $30 \times (q = 10)$, $77 \times (q = 20)$ and $380 \times (q = 30)$ more function evaluations than ZO-SGD. The sharp drop of attack loss in each method could be caused by the hinge-like loss as part of the total loss function, which turns to 0 only if the attack becomes successful. Compared to ZO-ProxSGD, ZO-PSVRG+ offers a faster convergence to a more accurate solution, and its convergence trajectory is more stable as q becomes larger (due to the reduced variance of Avg-RandGradEst). In addition, ZO-PSVRG+ improves the l_2 distortion of adversarial examples compared to ZO-ProxSGD (e.g., 30% improvement when $q = 30$).

9 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 9.1 (Three-Point Property)**.** *Let $F(\cdot)$ be a convex function, and let $D_l(\cdot,\cdot)$ be the Bregman distance for $l(\cdot)$. For a given vector z, let*

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

Then

$$F(x) + D_l(x,z) \geq F(z^+) + D_l(x^+,z) + D_l(x,z^+) \qquad for\ all\ x \in \mathbb{R}^n \tag{54}$$

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

Lemma 9.2. *Assume that the function $f(x)$ is L-smooth. Let $\hat{\nabla} f(x)$ denote the estimated gradient defined by CoordSGE. 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*

$$\hat{\nabla} f(x) = \sum_{j=1}^{d} \frac{\partial f_{\mu_j}}{\partial x_j} e_j \tag{55}$$

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

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

$$\left| f_{\mu_j}(x) - f(x) \right| \leq \frac{L\mu_j^2}{2} \tag{56}$$

$$\left| \frac{\partial f_{\mu_j}(x)}{\partial x_j} \right| \leq \frac{L\mu_j^2}{2} \tag{57}$$

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

$$\left\| \hat{\nabla} f(x) - \nabla f(x) \right\|^2 \leq \frac{L^2 d^2 \mu^2}{4} \tag{58}$$

Lemma 9.3. *Let $x^+ = Prox_{\eta h}(x - \eta v)$, then the following inequality holds:*

$$F(x^+) \leq F(z) + \langle \nabla f(x) - v, x^+ - z\rangle - \frac{1}{\eta}\langle x^+ - x, x^+ - z\rangle + \frac{L}{2}\|x^+ - x\|^2 + \frac{L}{2}\|z - x\|^2, \forall z \in \mathbb{R}^d. \tag{59}$$

Proof. First, we recall the proximal operator

$$\text{Prox}_{\eta h}(x - \eta v) := \text{arg} \min_{y\in\mathbb{R}^d} \left(h(y) + \frac{1}{2\eta}\|y - x\|^2 + \langle v, y\rangle \right) \tag{60}$$

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

$$\begin{aligned} h(x^+) &\leq h(z) + \langle p, x^+ - z\rangle \\ &= h(z) - \left\langle v + \frac{1}{\eta}(x^+ - x), x^+ - z \right\rangle \end{aligned} \tag{61}$$

where $p \in \partial h(x^+)$ such that $p + \frac{1}{\eta}(x^+ - x) + v = 0$ according to the optimality condition of (60), and (61) due to the convexity of h.

$$f(x^+) \leq f(x) + \langle \nabla f(x), x^+ - x \rangle + \frac{L}{2} \|x^+ - x\|^2 \tag{62}$$

$$-f(z) \leq -f(x) + \langle -\nabla f(x), z - x \rangle + \frac{L}{2} \|z - x\|^2 \tag{63}$$

where (62) holds since $f(x)$ has L-Lipschitz continuous gradient, and (63) holds since $-f(x)$ has the same L-Lipschitz continuous gradient as $f(x)$.

This lemma is proved by adding (61), (62), (63), and recalling $F(x) = f(x) + h(x)$. □

Lemma 9.4. *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*

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

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

$$h(x_t^s) \leq h(\overline{x}_t^s) - \left\langle \hat{v}_{t-1}^s + \frac{1}{\eta}(x_t^s - x_{t-1}^s), x_t^s - \overline{x}_t^s \right\rangle \tag{64}$$

$$h(\overline{x}_t^s) \leq h(x_t^s) - \left\langle \nabla f(x_{t-1}^s) + \frac{1}{\eta}(\overline{x}_t^s - x_{t-1}^s), \overline{x}_t^s - x_t^s \right\rangle \tag{65}$$

where (64) and (65) hold due to (61). Adding (64) and (65), we have

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

$$\frac{1}{\eta} \|x_t^s - \overline{x}_t^s\|^2 \leq \left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s \right\| \|x_t^s - \overline{x}_t^s\| \tag{66}$$

$$\|x_t^s - \overline{x}_t^s\| \leq \eta \left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s \right\| \tag{67}$$

where (66) uses Cauchy-Schwarz inequality. Now this lemma is proved using Cauchy-Schwarz inequality and (67), i.e., $\left\langle \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s), x_t^s - \overline{x}_t^s \right\rangle \leq \left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s \right\| \left\| x_t^s - \overline{x}_t^s \right\| \leq \eta \left\| \nabla f(x_{t-1}^s) - \hat{v}_{t-1}^s \right\|^2$. □

References

Alekh Agarwal, Ofer Dekel, and Lin Xiao. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *COLT*, pages 28–40. Citeseer, 2010.

Zeyuan Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. *The Journal of Machine Learning Research*, 18(1):8194–8244, 2017.

Zeyuan Allen-Zhu and Yang Yuan. Improved svrg for non-strongly-convex or sum-of-non-convex objectives. In *International conference on machine learning*, pages 1080–1089, 2016.

Mihai Anitescu. Degenerate nonlinear programming with a quadratic growth condition. *SIAM Journal on Optimization*, 10(4):1116–1135, 2000.

Amir Beck and Marc Teboulle. A fast iterative shrinkage-thresholding algorithm for linear inverse problems. *SIAM journal on imaging sciences*, 2(1):183–202, 2009.

Dimitri P Bertsekas. Incremental proximal methods for large scale convex optimization. *Mathematical programming*, 129(2):163, 2011.

Richard P Brent. *Algorithms for minimization without derivatives*. Courier Corporation, 2013.

Pin-Yu Chen, Huan Zhang, Yash Sharma, Jinfeng Yi, and Cho-Jui Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pages 15–26. ACM, 2017.

Tianyi Chen and Georgios B Giannakis. Bandit convex optimization for scalable and dynamic iot management. *IEEE Internet of Things Journal*, 6(1):1276–1286, 2019.

Andrew R Conn, Katya Scheinberg, and Luis N Vicente. *Introduction to derivative-free optimization*, volume 8. Siam, 2009.

Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in neural information processing systems*, pages 1646–1654, 2014.

John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Pavel Dvurechensky, Alexander Gasnikov, and Eduard Gorbunov. An accelerated method for derivative-free smooth stochastic convex optimization. *arXiv preprint arXiv:1802.09022*, 2018.

Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 385–394. Society for Industrial and Applied Mathematics, 2005.

Michael C Fu. Optimization for simulation: Theory vs. practice. *INFORMS Journal on Computing*, 14(3):192–215, 2002.

Xiang Gao, Bo Jiang, and Shuzhong Zhang. On the information-adaptive variants of the admm: an iteration complexity perspective. *Journal of Scientific Computing*, 76 (1):327–363, 2018.

Saeed Ghadimi and Guanghui Lan. Stochastic first-and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

Saeed Ghadimi and Guanghui Lan. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2):59–99, 2016.

Bin Gu, Zhouyuan Huo, and Heng Huang. Zeroth-order asynchronous doubly stochastic algorithm with variance reduction. *arXiv preprint arXiv:1612.01425*, 2016.

Bin Gu, Zhouyuan Huo, Cheng Deng, and Heng Huang. Faster derivative-free stochastic algorithm for shared memory machines. In *International Conference on Machine Learning*, pages 1807–1816, 2018a.

Bin Gu, De Wang, Zhouyuan Huo, and Heng Huang. Inexact proximal gradient methods for non-convex and non-smooth optimization. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018b.

Davood Hajinezhad, Mingyi Hong, and Alfredo Garcia. Zeroth order nonconvex multi-agent optimization over networks. *arXiv preprint arXiv:1710.09997*, 2017.

Feihu Huang, Bin Gu, Zhouyuan Huo, Songcan Chen, and Heng Huang. Faster gradient-free proximal stochastic methods for nonconvex nonsmooth optimization. *arXiv preprint arXiv:1902.06158*, 2019.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the polyak-łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 795–811. Springer, 2016.

Ehsan Kazemi and Liqiang Wang. A proximal zeroth-order algorithm for nonconvex nonsmooth problems. In *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 64–71. IEEE, 2018.

Guanghui Lan and Yi Zhou. An optimal randomized incremental gradient method. *Mathematical programming*, pages 1–49, 2017.

Lihua Lei, Cheng Ju, Jianbo Chen, and Michael I Jordan. Non-convex finite-sum optimization via scsg methods. In *Advances in Neural Information Processing Systems*, pages 2348–2358, 2017.

Huan Li and Zhouchen Lin. Accelerated proximal gradient methods for nonconvex programming. In *Advances in neural information processing systems*, pages 379–387, 2015.

Zhize Li and Jian Li. A simple proximal stochastic gradient method for nonsmooth nonconvex optimization. In *Advances in Neural Information Processing Systems*, pages 5564–5574, 2018.

Xiangru Lian, Huan Zhang, Cho-Jui Hsieh, Yijun Huang, and Ji Liu. A comprehensive linear speedup analysis for asynchronous stochastic parallel optimization from zeroth-order to first-order. In *Advances in Neural Information Processing Systems*, pages 3054–3062, 2016.

Liu Liu, Minhao Cheng, Cho-Jui Hsieh, and Dacheng Tao. Stochastic zeroth-order optimization via variance reduction method. *arXiv preprint arXiv:1805.11811*, 2018a.

Sijia Liu, Jie Chen, Pin-Yu Chen, and Alfred O Hero. Zeroth-order online alternating direction method of multipliers: Convergence analysis and applications. *arXiv preprint arXiv:1710.07804*, 2017.

Sijia Liu, Bhavya Kailkhura, Pin-Yu Chen, Paishun Ting, Shiyu Chang, and Lisa Amini. Zeroth-order stochastic variance reduction for nonconvex optimization. In *Advances in Neural Information Processing Systems*, pages 3727–3737, 2018b.

Zhi-Quan Luo and Paul Tseng. Error bounds and convergence analysis of feasible descent methods: a general approach. *Annals of Operations Research*, 46(1):157–178, 1993.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Yu Nesterov. Gradient methods for minimizing composite functions. *Mathematical Programming*, 140(1):125–161, 2013.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Atsushi Nitanda. Accelerated stochastic gradient descent for minimizing finite sums. In *Artificial Intelligence and Statistics*, pages 195–203, 2016.

Nicolas Papernot, Patrick McDaniel, Ian Goodfellow, Somesh Jha, Z Berkay Celik, and Ananthram Swami. Practical black-box attacks against machine learning. In *Proceedings of the 2017 ACM on Asia conference on computer and communications security*, pages 506–519. ACM, 2017.

Neal Parikh, Stephen Boyd, et al. Proximal algorithms. *Foundations and Trends® in Optimization*, 1(3):127–239, 2014.

Boris Teodorovich Polyak. Gradient methods for minimizing functionals. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki*, 3(4):643–653, 1963.

Sashank J Reddi, Ahmed Hefny, Suvrit Sra, Barnabas Poczos, and Alex Smola. Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning*, pages 314–323, 2016a.

Sashank J Reddi, Suvrit Sra, Barnabás Póczos, and Alexander J Smola. Proximal stochastic methods for nonsmooth nonconvex finite-sum optimization. In *Advances in Neural Information Processing Systems*, pages 1145–1153, 2016b.

Ohad Shamir. On the complexity of bandit and derivative-free stochastic convex optimization. In *Conference on Learning Theory*, pages 3–24, 2013.

Ohad Shamir. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18(52):1–11, 2017.

Artem Sokolov, Julia Kreutzer, Stefan Riezler, and Christopher Lo. Stochastic structured prediction under bandit feedback. In *Advances in Neural Information Processing Systems*, pages 1489–1497, 2016.

Artem Sokolov, Julian Hitschler, and Stefan Riezler. Sparse stochastic zeroth-order optimization with an application to bandit structured prediction. *arXiv preprint arXiv:1806.04458*, 2018.

James C Spall. *Introduction to stochastic search and optimization: estimation, simulation, and control*, volume 65. John Wiley & Sons, 2005.

Martin J Wainwright, Michael I Jordan, et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2): 1–305, 2008.

Yining Wang, Simon Du, Sivaraman Balakrishnan, and Aarti Singh. Stochastic zeroth-order optimization in high dimensions. *arXiv preprint arXiv:1710.10551*, 2017.

Andre Wibisono, Martin J Wainwright, Michael I Jordan, and John C Duchi. Finite sample convergence rates of zero-order stochastic optimization methods. In *Advances in Neural Information Processing Systems*, pages 1439–1447, 2012.

Lin Xiao and Tong Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

Xiaotian Yu, Irwin King, Michael R Lyu, and Tianbao Yang. A generic approach for accelerating stochastic zeroth-order convex optimization. In *IJCAI*, pages 3040–3046, 2018.