A Variance Controlled Stochastic Method with Biased Estimation for Faster Non-convex Optimization

Jia Bi[1] and Steve R. Gunn[2]

University of Southampton, Southampton, UK[1,2]
[1]J.Bi@soton.ac.uk
[2]srg@soton.ac.uk

Abstract. This paper proposes a new novelty optimization method Variance Controlled Stochastic Gradient (VCSG) to improve the performance of the stochastic variance reduced gradient (SVRG) algorithm. To avoid over-reducing the variance of gradient by SVRG, a hyper-parameter λ is introduced in VCSG that is able to control the reduced variance of SVRG. Theory shows that the optimization method can converge by using an unbiased gradient estimator, but in practice, biased gradient estimation can allow more efficient convergence to the vicinity since an unbiased approach is computationally more expensive. λ also has the effect of balancing the trade-off between unbiased and biased estimations. Secondly, to minimize the number of full gradient calculations in SVRG, a variance-bounded batch is introduced to reduce the number of gradient calculations required in each iteration. For smooth non-convex functions, the proposed algorithm converges to an approximate first-order stationary point (i.e. $\mathbb{E}\|\nabla f(x)\|^2 \leq \epsilon$) within $\mathcal{O}(min\{1/\epsilon^{3/2}, n^{1/4}/\epsilon\})$ number of stochastic gradient evaluations, which improves the leading gradient complexity of stochastic gradient-based method SCSG ($\mathcal{O}(min\{1/\epsilon^{5/3}, n^{2/3}/\epsilon\})$ [20]. It is shown theoretically and experimentally that VCSG can be deployed to improve convergence.

Keywords: Non-convex Optimization · Deep learning · Computational Complexity.

1 Introduction

We study smooth non-convex optimization problems which is shown in Eq.1,

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

where neither f nor each component $f_i(x)(i \in [n])$ are necessary convex; possibly non-convex and Lipschitz ($\mathcal{L}-smooth$) [33, 30]. We use $\mathcal{F}_n$ to denote all $f_i(x)$ functions of the form in Eq. 1, and optimize such functions using an Incremental First-Order (IFO) oracle and a Stochastic First-Order (SFO) Oracle, which are defined in Definition 1 and 2 respectively.

Definition 1. *[1] For $f \in \mathcal{F}_n$, an IFO takes an index $i \in [n]$ and a point $x \in \mathbb{R}^d$, and returns the pair $(f_i(x), \nabla f_i(x))$.*

Definition 2. *[28] For a function $F(x) = \mathbb{E}_y f(x, y)$ where $y \sim P$, a SFO returns the stochastic gradient $G(x_k, y_k) = \nabla_x f(x_k, y_k)$ where y_k is a sample drawn i.i.d. from P in the k_{th} call.*

Non-convex optimization is required for many statistical learning tasks ranging from generalized linear models to deep neural networks [23, 20].

As finding the global minima is NP-hard problem, the standard target of non-convex optimisation with provably guarantee is to estimate *approximate local minima* [2, 4]. After analysing this goal, there are rich literature translating this goal into what a fast *heuristic* algorithms for finding global minimum. For example, many earlier works have focused on the asymptotic performance of algorithms [15, 7, 34] and non-asymptotic complexity bounds have emerged [20]. To our knowledge, the first non-asymptotic convergence for stochastic gradient descent (SGD) was proposed by [16] with $\mathcal{O}(1/\epsilon^2)$. Full batch gradient descent (GD) is known to ensure convergence with $\mathcal{O}(n/\epsilon)$. Compared with SGD, GD's rate has better dependence on ϵ but worse dependence on n due to the requirement of computing a full gradient. Variance reduced (VR) methods based on SGD, e.g. Stochastic Variance Reduced Gradient (SVRG) [17], SAGA [13] have been shown to achieve better dependence on n than GD on non-convex problems with $\mathcal{O}(n+(n^{2/3}/\epsilon))$ [29, 30]. However, compared with SGD, the rate of VR based methods still have worse dependence on ϵ unless $\epsilon \ll n^{-2/3}$. Recently, [20] proposed a method called SCSG combining the benefits of SGD and SVRG, which is the first algorithm that achieves a better rate than SGD and is no worse than SVRG with $\mathcal{O}(1/\epsilon^{5/3} \wedge n^{2/3}/\epsilon)$. SNVRG proposed by [35] uses nested variance reduction to reduce the result of SCSG to $\mathcal{O}(log(\epsilon^{-1})(1/\epsilon^{3/2}) \wedge log(n)(n^{1/2}/\epsilon))$ that outperforms both SGD, GD and SVRG. Further SPIDER [14] proposes their both lower and upper bound as $\mathcal{O}(1/\epsilon^{3/2} \wedge n^{1/2}/\epsilon)$. Recently, [5] provide the lower bound of ϵ-based convergence rate as $\mathcal{O}(1/\epsilon^{3/2})$ which is same with the ϵ-related upper bound of SPIDER. As a result, the ϵ-related convergence rate $\mathcal{O}(1/\epsilon^{3/2})$ is likely to be the best currently. To the best of our knowledge, SPIDER is a leading result of gradient complexity for smooth non-convex optimization by using averaged L-Lipschitz gradients. Their work motivates the research question about whether an algorithm based on SGD or VR-based methods can further reduce the rate of SPIDER when it depends on ϵ in the regime of modest target accuracy or depends on n in the regime of high target accuracy, respectively.

However, for SGD and VR-based stochastic algorithms, there still exists three challenges. Firstly, stochastic based optimization algorithm do not require a full gradient computation. As a result, SCSG, SNVRG, SPIDER reduce full batch-size from $\mathcal{O}(n)$ to its subset $\mathcal{O}(B)$ where $1 \leq B < n$, which can significantly reduce the computational cost. However, it is challenging to appropriately scale the subset of samples in each stage of optimization to accelerate the convergence and achieve the same accuracy with full samples. Secondly, the variance of SGD is reduced by VR methods since the gradient of SGD is often too noisy to converge. However, VR schemes reduce the ability to escape local minima in later iterations due to a diminishing variance [8]. The challenge of SGD and VR methods is, therefore, to control the variance of gradients. Lastly, there exists a trade-off between biased/unbiased estimation in VR-based algorithms. SVRG is an unbiased estimation that can guarantee to converge but is not efficient to be used in real-world applications. Biased estimation can give a greater upper bound

of the mean squared error (MSE) loss function [22], and many works have proposed asymptotically biased optimization with biased gradient estimators to converge to the vicinity of minima, which is an economical alternative to an unbiased version [10–12, 9]. These methods provide a good insight into the biased gradient search. However, they hold under restrictive conditions, which are very hard to verify for complicated stochastic gradient algorithms. Thus, the last challenge is how to balance the unbiased and biased estimator in different stages of the non-convex optimization process.

To address these three challenges in order to accelerate the convergence of non-convex optimization, we propose our method Variance Controlled Stochastic Gradient(VCSG) to control the reduced variance of gradients, scale the subset of full batch samples and choose the biased or unbiased estimator in each iteration. Table 1 compares the five methods' theoretical convergence rates, which shows that VCSG has the fastest convergence rate among the methods. Here, we did not compare our result to SNVRG and SPIDER since both of their results are under averaged Lipschitz assumption, which is not same with our problem domain. We then show empirically that VCSG has faster rates of convergence than SGD, SVRG and SCSG.

Table 1. Comparison of results on SFO Definition 2 and IFO calls Definition 1 of gradient methods for smooth non-convex problems. The best upper bound of SFO in VCSG is still the lower bound that is proven by [5]. The upper bound of IFO in VCSG is better than other methods that use both full or subset of batch samples.

Algorithms	SFO/IFO calls on Non-convex	Batch size B	Learning rate η
GD [26]	$\mathcal{O}(n/\epsilon)$	n	$\mathcal{O}(L^{-1})$
SGD [16]	$\mathcal{O}(1/\epsilon^2)$	n	$\mathcal{O}(L^{-1})$
SVRG [30, 3]	$\mathcal{O}(n + (n^{2/3}/\epsilon))$	n	$\mathcal{O}(L^{-1}n^{-2/3})$
SCSG [20]	$\mathcal{O}(1/\epsilon^{5/3} \wedge n^{2/3}/\epsilon)$	$B(B < n)$	$\mathcal{O}(L^{-1}(n^{-2/3} \wedge \epsilon^{4/3}))$
VCSG	$\mathcal{O}(\mathbf{1}/\epsilon^{\mathbf{3/2}} \wedge \mathbf{n^{1/4}}/\epsilon)$	$B(B < n)$	$\mathcal{O}(L^{-1} \wedge L^{-1}B^{-1/2})$

We summarize and list our main contributions:

- We provide an new method VCSG, a well-balanced VR method for SGD to achieve a competitive convergence rate. We provide a theoretical analysis of VCSG on non-convex problems, which might be the first analysis about controlled variance reduction that can achieve comparable or faster convergence than gradient-based optimization.
- VCSG provides an appropriate sample size in each iteration by the controlled variance reduction, which can significantly save computational cost.
- VCSG balances the trade-off in biased and unbiased estimation, which provides a fast convergence rate.
- We also evaluate VCSG on three different datasets with three deep learning models. It is shown that our method in practice can achieve better performance than other leading results.

2 Preliminaries

We use $\|\cdot\|$ to denote the Euclidean norm for brevity throughout the paper. For our analysis, the background that are required to introduce definitions for L-smooth and ϵ-accuracy which now are defined in Definition 3 and Definition 4 respectively.

Definition 3. *Assume the individual functions f_i in Eq.1 are $\mathcal{L}$-smooth if there is a constant L such that*

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

for some $L < \infty$ and for all $i \in \{1, ..., n\}$.

We analyze convergence rates for Eq.1 and apply $\|\nabla f\|^2 \leq \epsilon$ convergence criterion by [25], which the concept of *ϵ-accurate* is defined in Definition 4. Moreover, the minimum IFO/SFO in Definition 1 and 2 to reach an $\epsilon-$accurate solution is denoted by $C_{comp}(\epsilon)$, and its complexity bound is denoted by $\mathbb{E}C_{comp}(\epsilon)$.

Definition 4. *A point x is called ϵ-accurate if $\|\nabla f(x)\|^2 \leq \epsilon$. An iterative stochastic algorithm can achieve ϵ-accuracy within t iterations if $\mathbb{E}[\|\nabla f(x^t)\|^2] \leq \epsilon$, where the expectation is over the algorithm.*

We follow part of the work in SCSG. Based on their algorithm settings, we recall that a random variable N has a geometric distribution $N \sim Geom(\gamma)$ if N is supported on the non-negative integrate, which their elementary calculation has been shown as $\mathbb{E}_{N\sim Geom}(\gamma) = \gamma/(1-\gamma)$. For brevity, we also write $\nabla f_{\mathcal{I}}(x) = (1/|\mathcal{I}|)\sum_{i\in\mathcal{I}} \nabla f_i(x)$. Note that calculating $\nabla f_{\mathcal{I}}(x)$ incurs $|\mathcal{I}|$ units of computational cost. The minimum IFO complexity to reach an *ϵ-accurate* solution is denoted by $\mathcal{C}_{comp}(\epsilon)$.

To formulate our complexity bound, we define:

$$f^* = \inf_x f(x) \quad \text{and} \quad \triangle_f = f(\tilde{x}_0) - f^* > 0, \tag{2}$$

Further, an upper bound on the variance of the stochastic gradients can be defined as:

$$\mathcal{S}^* = \sup_x \frac{1}{n}\sum_{i=1}^{n} \|\nabla f_i(x) - \nabla f(x)\|^2. \tag{3}$$

3 Variance controlled SVRG with a combined unbiased/biased estimation

To resolve the first challenge of SG-based optimization, we provide an adjustable schedule of batch size $B < n$, which scales the sample size for optimization. For the second challenge of controlling reduced variance, one method [8] balanced the gradient of SVRG in terms of the stochastic element and its variance to allow the algorithm to choose appropriate behaviors of gradient from stochastic, through reduced variance, to batch gradient descent by introducing a hyper-parameter λ. Based on this method, we focus on analysing the variance controller λ in our case. Towards the last challenge associated with the trade-off between biased and unbiased estimators, we analyze the nature

of biased and unbiased estimators in different stages of the non-convex optimization and propose a method that combines the benefits of both biased and unbiased estimator to achieve a fast convergence rate. Firstly, we show a generic form of the batched SVRG in Alg 1, which is proposed by [20]. We modified Alg 1 that scale the gradient by multiplying 0.5 to maintain same scale range with our algorithm in later section. Compared with the SVRG algorithm, the batched SVRG algorithm has a mini-batch procedure in the inner loop and outputs a random sample that instead of an average of the iterates. As seen in the pseudo-code, the batched SVRG method consists of multiple epochs, the batch-size B_j is randomly chosen from the whole samples n in j-th epoch and work with mini-batch b_j to generate the total number of updates for inner k-th epoch by a geometric distribution with mean equal to the batch size. Finally it outputs a random sample from $\{\tilde{x}_j\}_{j=1}^T$. This is a standard way also proposed by [24], which can save additional overhead by calculating the minimum value of output as $\arg\min_{j\leq T} \|\nabla f(\tilde{x}_j)\|$.

Algorithm 1: Batching SVRG

input : Number of epochs T, initial iterate $\tilde{x}_0$, step-size $(\eta_j)_{j=1}^T$, batch size $(B_j)_{j=1}^T$, mini-batch sizes $(b_j)_{j=1}^T$.

for $j = 1$ **to** T **do**
  Uniformly sample a batch $\mathcal{I}_j$ from total number of training samples n as $\mathcal{I}_j \subset \{1, ..., n\}$ with $|\mathcal{I}_j| = B_j$;
  $g_j \leftarrow \nabla f_{\mathcal{I}_j}(\tilde{x}_{j-1})$;
  $x_0^{(j)} \leftarrow \tilde{x}_{j-1}$;
  Generate $\mathcal{N}_j \sim \text{Geom}(B_j/(B_j + b_j))$;
  for $k = 1$ **to** $\mathcal{N}_j$ **do**
    Randomly select $\tilde{\mathcal{I}}_{k-1} \subset \{1, ..., n\}$ with $|\tilde{\mathcal{I}}_{k-1}| = b_j$;
    $v_{k-1}^{(j)} \leftarrow 0.5 \cdot (\nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_{k-1}^{(j)}) - \nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_0^{(j)}) + g_j)$;
    $x_k^{(j)} \leftarrow x_{k-1}^{(j)} - \eta_j v_{k-1}^{(j)}$;
  end
  $\tilde{x}_j \leftarrow x_{\mathcal{N}_j}^{(j)}$;
end

output: Sample $\tilde{x}_T^*$ from $\{\tilde{x}_j\}_{j=1}^T$ with $P(\tilde{x}_T^* = \tilde{x}_j) \propto \eta_j B_j / b_j$.

For the two cases of unbiased and biased estimations for the batched SVRG, we provide two upper bounds on their convergence for their gradients in the following two sub-sections. Meanwhile, two corresponding lower bounds of batch size are provided for each case when their dependency is sample size n. Unlike the specific parameter settings in SCSG, we use more general schedules (including learning rate η_j and mini-batch size b_j), aiming to estimate the best schedules in each stage of optimization for both unbiased and biased estimators, which avoids ad hoc choosing parameters. Proof details are presented in the appendix.

3.1 Weighted unbiased estimator analysis

In the first case, we introduce a hyper-parameter λ that is applied in a weighted unbiased version of the batched SVRG and is shown in Alg 2. Since our method based on SVRG, the λ should be within the range $0 < \lambda < 1$ in unbiased and biased cases.

Algorithm 2: Batching SVRG with weighted unbiased estimator

1 Replace line number 8 in Alg. 1 with the following line:
$v_{k-1}^{(j)} \leftarrow (1-\lambda)\nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_{k-1}^{(j)}) - \lambda\left(\nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_0^{(j)}) - g_j\right)$;

We now analyse the upper bound of expectation of gradients in a single epoch. Under our settings, we can achieve the upper bound for one-epoch analysis which is shown in Theorem 2.

Theorem 1. *Let* $\eta_j L = \gamma(\frac{b_j}{B_j})^{\alpha}$ $(0 \le \alpha \le 1)$ *and* $B_j \ge b_j \ge B_j^{\beta}$ $(0 \le \beta \le 1)$ *for all* j. *Suppose* $0 < \gamma \le \frac{1}{3}$, *then under Definition 3, the output* $\tilde{x}_j$ *of Alg 2 we have*

$$\mathbb{E}\|\nabla f(\tilde{x}_j)\|^2 \le \frac{2L}{\gamma\theta} \cdot (\frac{b_j}{B_j})^{1-\alpha}\mathbb{E}\left(f(\tilde{x}_{j-1}) - f(\tilde{x_j})\right) + \frac{2\lambda^4 I(B_j < n)\mathcal{S}^*}{\theta B_j^{1-2\alpha}},$$

where $I(B_j < n) \ge \frac{n - B_j}{(n-1)B_j}$, $\mathcal{S}$ *is defined in Eq.3,* $\lambda = \frac{1}{2}$ *and* $\theta = 2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1})(1-\lambda)^2 - 1.29(1-\lambda)^2$.

When only assuming smoothness, Over all epochs T, the output $\tilde{x}_T^*$ that is randomly selected from $(\tilde{x}_j)_{j=1}^T$. Thus, Theorem1 can be telescoped for over all epochs in the following theorem.

Theorem 2. *Under all assumptions of Theorem 1,*

$$\mathbb{E}\|\nabla f(\tilde{x}_j)\|^2 \le \frac{(\frac{2L}{\gamma})\triangle_f}{\theta\sum_{j=1}^T b_j^{\alpha-1}B_j^{1-\alpha}} + \frac{2\lambda^4 I(B_j < n)\mathcal{S}^*}{\theta\sum_{j=1}^T B_j^{1-2\alpha}},$$

where $\triangle_f$ *is defined in Eq.2.*

3.2 Biased estimator analysis

In this sub-section we theoretically analyze the performance of the biased estimator, which is shown in Alg 3.

Algorithm 3: Batching SVRG with biased estimator

1 Replace the line number 8 in Alg 1 with the following line:
$v_{k-1}^{(j)} \leftarrow (1-\lambda)\left(\nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_{k-1}^{(j)}) - \nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_0^{(j)})\right) + \lambda g_j$;

Applying the same schedule of η_j and b_j that are used in the unbiased case, we can achieve the results on both one-epoch and all-epoch for this case, which are shown in Theorem 3 and Theorem 4 respectively.

Theorem 3. *let* $\eta_j L = \gamma(\frac{b_j}{B_j})^{\alpha}$ $(0 \leq \alpha \leq 1)$. *Suppose* $0 < \gamma \leq \frac{1}{3}$ *and* $B_j \geq b_j \geq B_j^{\beta}$ $(0 \leq \beta \leq 1)$ *for all* j, *then under Definition 3, the output* $\tilde{x}_j$ *of Alg 3 we have,*

$$\mathbb{E}\|\nabla f(\tilde{x}_j)\|^2 \leq \frac{2L}{\gamma\Theta} \cdot (\frac{b_j}{B_j})^{1-\alpha}\mathbb{E}\left(f(\tilde{x}_{j-1}) - f(\tilde{x_j})\right) + \frac{2(1-\lambda)^2 I(B_j < n)\mathcal{S}^*}{\Theta B_j^{1-2\alpha}},$$

where $I(B_j < n) \geq \frac{n - B_j}{(n-1)B_j}$, $\mathcal{S}$ *is defined in Eq.3,* $0 < \lambda < 1$ *and* $\Theta = 2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1} - 4LB_j^{2\alpha-2})(1-\lambda)^2 - 1.16(1-\lambda)^2$.

Theorem 4. *Under all assumptions of Theorem 3,*

$$\mathbb{E}\|\nabla f(\tilde{x}_j)\|^2 \leq \frac{(\frac{2L}{\gamma})\triangle_f}{\Theta\sum_{j=1}^{T} b_j^{\alpha-1}B_j^{1-\alpha}} + \frac{2(1-\lambda)^2 I(B_j < n)\mathcal{S}^*}{\Theta\sum_{j=1}^{T} B_j^{1-2\alpha}},$$

where $I(B_j < n) \geq \frac{n - B_j}{(n-1)B_j}$, $0 < \lambda < 1$ *and* $\Theta = 2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1} - 4LB_j^{2\alpha-2})(1-\lambda)^2 - 1.16(1-\lambda)^2$.

3.3 Convergence analysis for smooth non-convex optimization

Starting to consider from a constant batch/mini-batch size $1 \leq B_j \equiv B \leq n$ for some $1 < B \leq n$, $b_j = B_j^{\beta} \equiv B^{\beta}$ $(0 \leq \beta \leq 1)$, we can achieve the computational complexity of output from Theorem 2 and 4 that is given as

$$\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 = \mathcal{O}\left(\frac{L\triangle_f}{TB^{1+\alpha\beta-\alpha-\beta}} + \frac{\mathcal{S}^*}{B^{1-2\alpha}}\right), \tag{4}$$

which covers two extreme cases of complexity bounds since the batch-size B_j has two different dependencies.

Dependence on ϵ. If $b_j = B^{\beta}$ when $\beta = 1$ and $1 < B_j \equiv B < n$, the second term of Eq.4 can be made $\mathcal{O}(\epsilon)$ by setting $B_j^{1-2\alpha} = B = \mathcal{O}\left(\frac{\mathcal{S}^*}{\epsilon}\right)$, where incurs $\alpha = 0$.And $T(\epsilon) = \left(\frac{L\triangle_f}{\epsilon}\right)$ resulting in the complexity bound is given as $\mathbb{E}C_{comp}(\epsilon) = \mathcal{O}\left(\frac{L\triangle_f B}{\epsilon}\right) = \mathcal{O}\left(\frac{L\triangle_f \mathcal{S}^*}{\epsilon^2}\right)$, which obtains the same with the rate of SGD as shown in Table 1.

Dependence on n. If $b_j = 1$ when $\beta = 0$ and $B_j = n$, Eq.4 can be further alternative as $\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 = \mathcal{O}\left(\frac{L\triangle_f}{Tn^{1-\alpha}} + \frac{\mathcal{S}^*}{n^{1-2\alpha}}\right)$. When $\alpha \leq \frac{1}{2}$, $T(\epsilon)$ can be made as $\mathcal{O}\left(1 + \frac{L\triangle_f}{\epsilon n^{1/2}}\right)$, which yields the complexity bound become as

$$\mathbb{E}C_{comp}(\epsilon) = \mathcal{O}\left(n + \frac{n^{\frac{1}{2}}L\triangle_f}{\epsilon}\right). \tag{5}$$

This upper bound of rate can guarantee to be better than SCSG, as shown in Table 1.

However, both of the above settings are two sub-optimal cases since their extreme setting either the parameter mini-batch size b_j is too large or batch size B_j is too large. We now discuss the batch-size schedules depending on the above two dependencies.

3.4 Scaling Batch samples

For the case of batch size B_j depending on ϵ, $B_j = \mathcal{O}\left(\frac{\mathcal{S}^*}{\epsilon}\right)$, $b_j \neq 1$, and learning rate $\eta_j = \frac{\gamma}{L}(\frac{1}{B_j})^{\alpha(1-\beta)}$ where $0 \leq \alpha \leq \frac{1}{2}$. To determine the optimal value of b_j in this case, we compared to the extreme case when $b_j = 1$ and $B_j = n$ that the optimal schedule of learning rate $\eta_j = \frac{\gamma}{L}(\frac{1}{B_j})^{\frac{2}{3}}$ is provided by [30, 29, 3, 20]. Correspondingly in our general form of learning rate, they specified $\alpha = \frac{2}{3}$ and $\beta = 0$. Thus, the learning rate η_j has a range which is shown as $\frac{\gamma}{L} \geq \frac{\gamma}{L}(\frac{1}{B_j})^{\frac{2}{3}(1-\beta)} \geq \frac{\gamma}{L}(\frac{1}{B_j})^{\frac{1}{2}}$. As a result, we can estimate the range of β as $0 \leq \beta \leq 1/4$. Consequently, $\beta = 1/4$ and $\alpha = 0$ are the optimal values in this case.

After determined the three schedules including B_j, η_j and b_j, we can estimate the optimal value of λ^*. For the first case that $B_j = \mathcal{O}\left(\frac{\mathcal{S}^*}{\epsilon}\right)$, $b_j = B_j^{\frac{1}{4}}$, $\eta_j = \frac{1}{3L}$, Eq.4 is specified as

$$\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 = \mathcal{O}\left(\frac{L\triangle_f}{T}\left(\frac{\epsilon}{\mathcal{S}^*}\right)^{\frac{3}{4}} + \epsilon\right). \tag{6}$$

Since in this case batch size depends on ϵ, we more focus on the second term in Eq. 6. As a result, we optimize the second term of $\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2$ from both Theorem 2 and 4 in order to achieve lowest upper bound. After comparison the upper bounds in both Theorem. 2 and 4, we choose the optimal value of $\lambda^* = \dfrac{1}{2}$ with the unbiased estimation case, which can provide the lowest upper bound of gradient resulting faster convergence.

For the case of batch size B_j depending on n, we now analyse the lower bound of batch size B_j in both unbiased and biased estimations. When applying unbiased estimator, for a single epoch, j, we define the weighted unbiased variance as $e_j = \lambda\left(\nabla f_{\mathcal{I}_j}(\tilde{x}_{j-1}) - \nabla f(\tilde{x}_{j-1})\right)$. Thus, the gradients in Alg 2 can be updated within the j-th epoch as $\mathbb{E}_{\tilde{\mathcal{I}}_k} v_k^{(j)} = (1-\lambda)\nabla f(x_k^{(j)}) + e_j$, which reveals the key difference between the batched SVRG and the variance controlled batched SVRG on both unbiased/ biased estimators. Most of the novelty in our analysis lies in dealing with the extra term e_j. Since we achieve a lower bound of batch-size by bounding the term e_j, we provide the bound of the term e_j as $\mathbb{E}_{\mathcal{I}_j}\|e_j\|^2 \leq \lambda^2\frac{n-B_j}{nB_j}\mathcal{K}^2 \leq \lambda^2\frac{n-B_j}{nB_j}\frac{n}{\sqrt{n-1}}\mathcal{S}^* \leq \sigma\rho^{2j}$, where the first inequation follows [18, 6] the variance of the norms of gradients $\mathcal{K}^2 \geq \frac{1}{n-1}\sum_{i=1}^n[\|\nabla f_i(\tilde{x}_{j-1})\|^2 - \|\nabla f(\tilde{x}_{j-1})\|^2]$, the second inequation follows the Samuelson inequality [27] that $\mathcal{K}^2 \leq \frac{n}{\sqrt{n-1}}\mathcal{S}^*$ where $\mathcal{S}^*$ is shown in Eq. 3, and in the last inequation, there is an upper bound of variance where $\sigma \geq 0$ is a constant for some $\rho < 1$. Thus B_j in unbiased case can be bounded as,

$$B_j \geq \frac{n\mathcal{S}^*}{\mathcal{S}^* + \lambda^2 n^{\frac{1}{2}}\sigma\rho^{2j}}. \tag{7}$$

For batch size in biased case, we use the same approach adopted in the unbiased version. For a single epoch, j, we define the biased variance as $e_j = \lambda \nabla f_{\mathcal{I}_j}(\tilde{x}_{j-1}) - (1-\lambda)\nabla f(\tilde{x}_{j-1})$. And we achieve the lower bound of batch-size, which is shown in the following.

$$B_j \geq \begin{cases} \dfrac{n\mathcal{S}^*}{\mathcal{S}^* + (1-\lambda)^2 n^{\frac{1}{2}} \sigma \rho^{2j}}, & \text{if } 0 < \lambda < \dfrac{\sqrt{2}}{2}. \\ \dfrac{n\mathcal{S}^*}{\mathcal{S}^* + (3\lambda^2 - 2\lambda)^2 n^{\frac{1}{2}} \sigma \rho^{2j}}, & \text{if } \dfrac{\sqrt{2}}{2} < \lambda < 1. \end{cases} \tag{8}$$

To estimate the optimal value of λ^* in this case that batch size depending on n, we specified lower bound of batch size B_j which has two versions of biased and unbiased estimations, $b_j = 1$ and $\eta_j = \dfrac{1}{3L}(\dfrac{1}{B_j})^{\frac{1}{2}}$ when optimal value $\alpha = \dfrac{1}{2}$. Thus Eq.4 can be specified as

$$\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 = \mathcal{O}\left(\frac{L\triangle_f}{TB_j^{\frac{1}{2}}} + \mathcal{S}^*\right). \tag{9}$$

Due to this case that batch size depending on n, we more focus on the first term in Eq. 9. Thus we optimise the first term in the upper bound of $\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2$ in both Theorem 2 and 4. After comparison of upper bounds both in unbiased and biased cases, we determine $\lambda^* = 5/8$ with biased estimation that obtain the lowest upper bound.

Consequently, we can achieve the greater complexity bound of Eq.4 for both biased/unbiased estimations via replacing full sample size n by the batched sample size B_j in Eq.5, which is shown in Eq.10.

$$\mathbb{E}C_{comp}(\epsilon) = \mathcal{O}\left(B_j + B_j^{\frac{1}{2}} \cdot \frac{L\triangle_f}{\epsilon}\right). \tag{10}$$

3.5 Best of two worlds

We have seen in the previous section that the variance controlled SVRG combines the benefits of both SVRG and SGD. We now show these benefits can be made more pronounced by λ^* with best combinations between B_j and b_j in different stages of optimization. We introduce our algorithm *VCSG* shown in Alg 4.

Following Alg 4, we can achieve a general result for VCSG in the following theorem.

Theorem 5. *Suppose* $\gamma \leq \dfrac{1}{3}$. *Let* $B_j = \min\left\{\frac{\mathcal{S}^*}{\epsilon}, \frac{n\mathcal{S}^*}{\mathcal{S}^* + 0.14 \cdot n^{\frac{1}{2}}\sigma\rho^{2j}}\right\}$, *under Definition 3 and Theorem2 and 4, the output* $\tilde{x}_T^*$ *in Alg 4 satisfies one of two bounds.*

1. *If* $B_j = \dfrac{\mathcal{S}^*}{\epsilon}$,$b_j = B_j^{\frac{1}{4}}$, $\eta_j = \dfrac{\gamma}{L}$, $\lambda^* = \dfrac{1}{2}$, $\theta \approx 0.51$ *with an unbiased estimator,*

$$\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 \leq \frac{\frac{4L}{\gamma}\triangle_f}{\sum_{j=1}^T B_j^{\frac{3}{4}}} + \frac{0.24(I(B_j < n)\mathcal{S}^*}{B_j},$$

.

Algorithm 4: (Mini-Batch)VCSG

input : Same input parameters with Alg 1, initial batch size $B_1 = n$ and $b_1 = n^{1/4}$.

for $j = 1$ **to** T **do**
  Uniformly sample a batch $\mathcal{I}_j$ from total number of training samples n as $\mathcal{I}_j \subset \{1, ..., n\}$ with $|\mathcal{I}_j| = B_j$;
  $B_j \leftarrow \left\{ \frac{12\mathcal{S}_j^*}{\epsilon} \wedge \frac{n\mathcal{S}_j^*}{\mathcal{S}_j^* + 0.14 \cdot n^{\frac{1}{2}}\sigma\rho^{2j}} \right\}$ where $\sigma \geq 0, \rho < 1$;
  $g_j \leftarrow \nabla f_{\mathcal{I}_j}(\tilde{x}_{j-1})$;
  $\tilde{x}_0^{(j)} \leftarrow \tilde{x}_{j-1}$;
  Generate $\mathcal{N}_j \sim$ Geom$(B_j/(B_j + b_j))$;
  for $k = 1$**to** $\mathcal{N}_j$ **do**
    if $B_j = \mathcal{S}_j^*/\epsilon$ **then**
      $b_j = B_j^{\frac{1}{4}}$; $\eta_j = \frac{1}{3L}$;
      Randomly select $\mathcal{I}_{k-1} \subset \{1, ..., n\}$ with $|\tilde{\mathcal{I}}_{k-1}| = b_j$;
      $v_{k-1}^{(j)} = \frac{1}{2} \cdot \left(\nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_{k-1}^{(j)}) - \nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_0^{(j)}) + g_j \right)$;
    else if $B_j = B_0$ *or* $B_j = \frac{n\mathcal{S}_j^*}{\mathcal{S}_j^* + 0.14 \cdot n^{\frac{1}{2}}\sigma\rho^{2j}}$ **then**
      $b_j = 1$; $\eta_j = \frac{1}{3L}(\frac{1}{B_j})^{\frac{1}{2}}$;
      Randomly select $\mathcal{I}_{k-1} \subset \{1, ..., n\}$ with $|\tilde{\mathcal{I}}_{k-1}| = b_j$;
      $v_{k-1}^{(j)} = \frac{3}{8} \cdot (\nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_{k-1}^{(j)}) - \nabla f_{\tilde{\mathcal{I}}_{k-1}}(x_0^{(j)})) + \frac{5}{8} \cdot g_j$;
    $x_k^{(j)} \leftarrow x_{k-1}^{(j)} - \eta_j v_{k-1}^{(j)}$;
    $\mathcal{S}_k^{(j)} \leftarrow \| \nabla f_{\tilde{\mathcal{I}}_{k-1}}(\tilde{x}_0^{(j)}) - g_j \|^2$;
  end
  $\tilde{x}_j \leftarrow x_{\mathcal{N}_j}^{(j)}, \mathcal{S}_j^* \leftarrow \mathcal{S}_{\mathcal{N}_j}^{(j)}$;
end

output: Sample $\tilde{x}_T^*$ from $(\tilde{x}_j)_{j=1}^T$ with $P(\tilde{x}_T^* = \tilde{x}_j) \propto \eta_j B_j / b_j$

2. *If* $B_j = \frac{n\mathcal{S}^*}{\mathcal{S}^* + 0.14 \cdot n^{\frac{1}{2}}\sigma\rho^{2j}}$, $b_j = 1$, $\eta_j = \frac{\gamma}{L}(\frac{1}{B_j})^{\frac{1}{2}}$, $\lambda^* = \frac{5}{8}$, $\Theta \approx 0.59$ *with a biased estimator,*

$$\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 < \frac{\frac{3.4L}{\gamma}\triangle_f}{\sum_{j=1}^T B_j^{\frac{1}{2}}} + 0.48\mathcal{S}^*.$$

Now we discuss how parameters, including λ, step-size, batch-size, and mini-batch size, work together to control the variance of gradients from stochastic to batch and balance the trade-off between bias/unbiased estimation in batched optimization. Firstly, in very early iterations B_j might choose its first term due to the low variance. In this condition, the small λ with relatively large learning rate may help gradients being more stochastic to search more region of problem space, and also can help points escape from

bad local minima. During increasing variance, the first term of B_j would be increased as well, resulting B_j will choose its second term. In the second case, both relatively large λ, small learning rate and the biased estimator work together that can reduce variance to fast converge into a small region of space. In case of the variance that is reduced too small in the second case, B_j will turn to be its first term. We regard this whole process as **Coarse-to-Fine** dynamic searching methods.

To calculate the computational complexity of VCSG, we bring the schedule of batch size B_j into Eq. 10, which is shown in Corollary 1.

Corollary 1. *Under parameters setting in Theorem 5,* $B_j \equiv B = \{\frac{\mathcal{S}^*}{\epsilon} \wedge \frac{n\mathcal{S}^*}{\mathcal{S}^* + 0.14 \cdot n^{(1/2)}\sigma\rho^{2j}}\}$ *then it holds that*

$$\mathbb{E}_{comp}(\epsilon) = \mathcal{O}\left(B + \frac{L\triangle_f}{\epsilon} \cdot B^{\frac{1}{2}}\right).$$

$B = \{\frac{1}{\epsilon} \wedge n^{\frac{1}{2}}\}$ *since assume that* $L\triangle_f, \mathcal{S}^*, \sigma\rho^{2j} = \mathcal{O}(1)$. *Thus, the above bound can be simplified to*

$$\mathbb{E}_{comp}(\epsilon) = \mathcal{O}\left((\frac{1}{\epsilon} \wedge n^{\frac{1}{2}}) + \frac{1}{\epsilon} \cdot (\frac{1}{\epsilon} \wedge n^{\frac{1}{2}})^{\frac{1}{2}}\right) = \mathcal{O}\left(\frac{1}{\epsilon^{\frac{3}{2}}} \wedge \frac{n^{\frac{1}{4}}}{\epsilon}\right).$$

4 Application

To experimentally verify our theoretical results and insights, we evaluate VCSG compared with SVRG, SGD, and SCSG on three common DL topologies, including LeNet (LeNet-300-100 which has two fully connected layers as hidden layers with 300 and 100 neurons respectively, and LeNet-5 which has two convolutional layers and two fully connected layers) and VGG-16 [32] using three datasets including MNIST, CIFAR-10 and tiny ImageNet. Tiny ImageNet contains 200 classes for training each with 500 images and the test set contains 10,000 images. Each image is re-sized to 64×64 pixels [31].We initialize $B_j = B_0 = n$, correspondingly $b_j = b_0 = n^{\frac{1}{4}}$, $\eta_0 = 1/(3Ln^{\frac{1}{2}})$, and $\lambda = \frac{5}{8}$ via using biased estimator in the first epoch. Meanwhile, we choose a scaled SGD as our baseline by multiplying 0.5 with stochastic gradients, and applied decayed learning rate $\eta_j = \eta_0/(j)$ on SGD. For SVRG, we set-up $\lambda = 0.5$ with fixed learning rate $\eta_j = 1/(3Ln^{\frac{1}{2}})$ in Alg 1. The reason we choose SCSG is that our algorithm is inspired from SCSG which is a leading batched SVRG.

Fig.1 compares the performance of four methods, including SGD, SVRG, SCSG, and VCSG, via test log error, training log loss, and training time usage. It has two baselines in all sub-figures, including the performance of SVRG and SGD. The performance of SCSG test error and training loss is smaller than SGD on MNIST and CIFAR-10 data sets, consistent with the experimental results shown in [20]. However, in the ImageNet data set, which is a relatively larger scale application than the previous two data sets, the performance of SCSG becomes worse than SVRG and SGD, which showed weak robustness in our experiments. By contrast, VCSG shown as the green colour in all three datasets, has the lowest test error and training loss among all methods. In the ImageNet

data set, both the test error of VCSG is initially higher than SVRG and SGD, but VCSG can reduce the test error and loss dramatically after around 75 epochs. One possible explanation is that the algorithm changes the batch size to the first term resulting in an escape from a local minima by increasing the variance to find a better solution. The right-hand column of Fig.1 presents the time usage, and it can be seen that SVRG and SGD are similar, having higher training time than the other two methods in all three data sets. In Fig.2, we use a more visualized format to show the time usage in Fig.1.

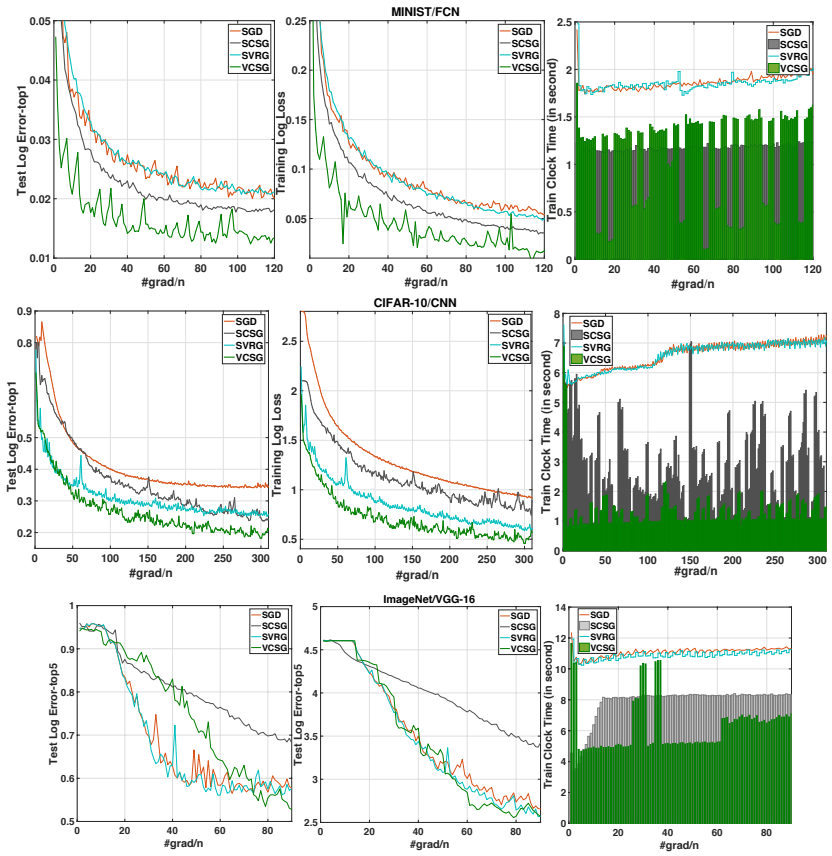

Fig. 1. Comparison of rates of convergence in four approaches, including SGD, SVRG, SCSG, and VCSG via test error, training loss and time consumption. Comparatively, we can see that VCSG can converge fastest during all iterations on MINIST and CIFAR-10 data sets. Even though VCSG on the ImageNet data set is slightly slower converging than the other three methods in the beginning, it can significantly decrease after several epochs when the batch-size becomes stable.

We can see in three sub-figures VCSG can achieve the lowest test error over a shorter time. To achieve the 0.025 top-1 test error in the MNIST data set, VCSG only takes 16 seconds around $2\times$ faster than SCSG, $3\times$ faster than SVRG, and $4\times$ faster than SGD.

In CIFAR-10 to achieve 0.3 top-1 test error, VCSG is around $6\times$ faster than SVRG, $4\times$ faster than SCSG and $13\times$ faster than SGD. In the ImageNet data set, to achieve 0.55 top-5 test error, VCSG can be faster than other methods by up to $5\times$.

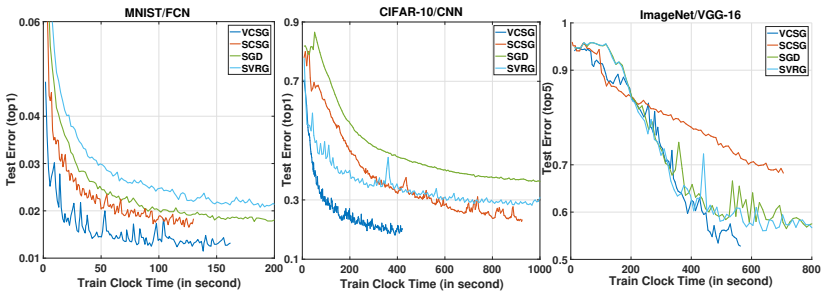

Fig. 2. Visualization of test error of four approaches, including SGD, SVRG and SCSG and VCSG against time consumption.

5 Discussion

In this paper, we proposed a VR-based optimization $VCSG$ for non-convex problems. We theoretically determined that a hyper-parameter λ in each iteration can control the reduced variance of SVRG and balance the trade-off between a biased and an unbiased estimator. Meanwhile, an adjustable batch bounded by controlled reduced variance can work with λ, step size, and mini-batch to choose an appropriate estimator to converge faster to a stationary point on non-convex problems. Moreover, to verify our theoretical results, our experiments use three datasets on three DL models to present the performance of VCSG via test error/loss and elapsed time and compare these with other leading results. Both theoretical and experimental results show that VCSG can efficiently accelerate convergence. We believe that our algorithm is worthy of further study for non-convex optimization, particularly in deep neural networks training in large-scale applications.

A Technique lemmas

The first two lemmas we will use in our theorems are from Lemma A.1 and Lemma A.2 in [20].

Lemma 1. *Let $x_1, ..., x_M \in \mathbb{R}^d$ be an arbitrary population of N vectors with*

$$\sum_{j=1}^{M} x_j = 0.$$

Further let $\mathcal{J}$ be a uniform random subset of $\{1, ...M\}$ with size m. Then

$$\mathbb{E}\|\frac{1}{m}\sum_{j\in\mathcal{J}} x_j\|^2 = \frac{M-m}{(M-1)m}\cdot\frac{1}{M}\sum_{j=1}^{M}\|x_j\|^2 \leq \frac{I(m<M)}{m}\cdot\frac{1}{M}\sum_{j=1}^{M}\|x_j\|^2.$$

The geometric random variable N_j has the key properties below.

Lemma 2. *Let $N \sim Geom(\gamma)$ for some $B > 0$. Then for any sequence $D_0,\ D_1, ...,\ D_N$ with $\mathbb{E}|D_N| < \infty$,*

$$\mathbb{E}(D_N - D_{N+1}) = (\frac{1}{\gamma} - 1)(D_0 - \mathbb{E}D_N).$$

B One-Epoch Analysis

B.1 Unbiased Estimator Version

Our algorithm is based on the SVRG method, thus the hyper-parameter λ should be within the range as $0 < \lambda < 1$ in both unbiased and biased cases. We provide all useful lemmas we will applied in our proof of theorems at first, and provide a proof sketch for guidance. We start by bounding the gradient $\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2$ in Lemma 3 and the variance $\mathbb{E}_{\mathcal{I}_j} \parallel e_j \parallel^2$ in Lemma 4.

Lemma 3. *Under Definition 3,*

$$\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 \leq \frac{L^2}{4b_j} \parallel x_k^{(j)} - x_0^{(j)} \parallel + 2(1-\lambda)^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + 2\lambda^2 \parallel e_j \parallel^2 .$$

Proof. Using the fact that for a random variable Z $\mathbb{E} \parallel Z \parallel^2 = \|Z - \mathbb{E}Z \|^2 + \| \mathbb{E}Z \|^2$, we have

$$\begin{aligned}
\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 &= \mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} - \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k^{(j)} \parallel^2 + \parallel \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k^{(j)} \parallel^2 \\
&= \mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel (1-\lambda)\nabla f_{\tilde{\mathcal{I}}_k}(x_k^{(j)}) - \lambda\nabla f_{\tilde{\mathcal{I}}_k}(x_0^{(j)}) - ((1-\lambda)\nabla f(x_k^{(j)}) - \lambda\nabla f x_0^{(j)}) \parallel^2 \\
&\quad + \parallel (1-\lambda)\nabla f(x_k^{(j)}) + \lambda e_j \parallel^2 \\
&\leq \mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel (1-\lambda)\nabla f_{\tilde{\mathcal{I}}_k}(x_k^{(j)}) - \lambda\nabla f_{\tilde{\mathcal{I}}_k}(x_0^{(j)}) - ((1-\lambda)\nabla f(x_k^{(j)}) - \lambda\nabla f x_0^{(j)}) \parallel^2 \\
&\quad + 2 \parallel (1-\lambda)\nabla f(x_k^{(j)}) \parallel^2 + 2 \parallel \lambda e_j \parallel^2 .
\end{aligned} \tag{11}$$

By Lemma 1,

$$\begin{aligned}
&\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel (1-\lambda)\nabla f_{\tilde{\mathcal{I}}_k}(x_k^{(j)}) - \lambda\nabla f_{\tilde{\mathcal{I}}_k}(x_0^{(j)}) - ((1-\lambda)\nabla f(x_k^{(j)}) - \lambda\nabla f x_0^{(j)}) \parallel^2 \\
&\leq \frac{1}{b_j}\cdot\frac{1}{n}\sum_{i=1}^{n} \parallel (1-\lambda)\nabla f_i(x_k^{(j)}) - \lambda\nabla f_i(x_0^{(j)}) - ((1-\lambda)\nabla f(x_k^{(j)}) - \lambda\nabla f(x_0^{(j)})) \parallel^2 \\
&= \frac{1}{b_j}\cdot(\frac{1}{n}\sum_{i=1}^{n} \parallel (1-\lambda)\nabla f_i(x_k^{(j)}) - \lambda\nabla f_i(x_0^{(j)}) \parallel^2 - \parallel ((1-\lambda)\nabla f(x_k^{(j)}) - \lambda\nabla f(x_0^{(j)})) \parallel^2) \\
&\leq \frac{1}{b_j}\cdot\frac{1}{n}\sum_{i=1}^{n} \parallel (1-\lambda)\nabla f_i(x_k^{(j)}) - \lambda\nabla f_i(x_0^{(j)}) \parallel^2 \\
&\leq \frac{1}{b_j}\cdot\frac{1}{4n}\sum_{i=1}^{n} \parallel \nabla f_i(x_k^{(j)}) - \nabla f_i(x_0^{(j)}) \parallel^2 \\
&\leq \frac{1}{b_j}\cdot\frac{L^2}{4} \parallel x_k^{(j)} - x_0^{(j)} \parallel^2
\end{aligned} \tag{12}$$

where the last line can be achieved by Definition3 only if $\lambda = \frac{1}{2}$.

Thus the bound of the gradient can be alternatively written as,

$$\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 \leq \frac{L^2}{4b_j} \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 + 2(1-\lambda)^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + 2\lambda^2 \parallel e_j \parallel^2 . \tag{13}$$

Lemma 4.

$$\mathbb{E}_{\mathcal{I}_j} \parallel e_j \parallel^2 \leq \lambda^2 \frac{I(B_j < n)}{B_j}\cdot\mathcal{S}^*.$$

Proof. Based on Lemma 3 and the observation that $\tilde{x}_{j-1}$ *is independent of* $\mathcal{I}_j$*, the bound of variance* e_j *can be expressed as*

$$\begin{aligned}
\mathbb{E}_{\mathcal{I}_j} \parallel e_j \parallel^2 &= \frac{n-B_j}{(n-1)B_j}\cdot\frac{\lambda^2}{n}\sum_{i=1}^{n} \parallel \nabla f_i(\tilde{x}_{j-1}) - \nabla f(\tilde{x}_{j-1}) \parallel^2 \\
&\leq \lambda^2\frac{n-B_j}{(n-1)B_j}\cdot\mathcal{S}^* \leq \lambda^2\frac{I(B_j<n)}{B_j}\mathcal{S}^*
\end{aligned} \tag{14}$$

where the upper bound of the variance of the stochastic gradients $\mathcal{S}^* = \frac{1}{n}\sum_{i=1}^{n} \parallel \nabla f_i(\tilde{x}_{j-1}) - \nabla f(\tilde{x}_{j-1}) \parallel^2$.

Lemma 5. *Suppose* $\eta_j L < 1$*, then under Definition 3,*

$$\begin{aligned}
&(1-\lambda)\eta_j(1-(1-\lambda)L\eta_j)B_j\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 + \lambda\eta_j B_j\mathbb{E} < e_j, \nabla f(\tilde{x}_j) > \\
&\leq b_j\mathbb{E}(f(\tilde{x}_{j-1}) - f(\tilde{x}_j)) + \frac{\eta_j^2 B_j L^3}{2b_j}\mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2 + \lambda^2 L\eta_j^2 B_j\mathbb{E} \parallel e_j \parallel^2 .
\end{aligned}$$

where $\mathbb{E}$ *denotes the expectation with respect to all randomness.*

Proof. By Definition 3, we have

$$\begin{aligned}
&\mathbb{E}_{\tilde{\mathcal{I}}_k}[f(x_{k+1}^{(j)})] \leq f(x_k^{(j)}) - \eta_j < \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k, \nabla f(x_k^{(j)}) > + \frac{L\eta_j^2}{2} \mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k \parallel^2 \\
&= f(x_k^{(j)}) - \eta_j < ((1-\lambda)\nabla f(x_k^{(j)}) + \lambda e_j), \nabla f(x)_k^{(j)}) > + \frac{L\eta_j^2}{2} \mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k \parallel^2 \\
&\leq f(x_k^{(j)}) - \eta_j(1-\lambda) \parallel \nabla f(x_k^{(j)}) \parallel^2 - \eta_j < \lambda e_j, \nabla f(x_k^{(j)}) > + \frac{L^3\eta_j^2}{2b_j} \parallel (1-\lambda)x_k^{(j)} - \lambda x_0^{(j)} \parallel^2 \\
&+ L\eta_j^2(1-\lambda)^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + L\eta_j^2\lambda^2 \parallel e_j \parallel^2 \\
&= f(x_k^{(j)}) - (\eta_j(1-\lambda) - L\eta_j^2(1-\lambda)^2) \parallel \nabla f(x_k^{(j)}) \parallel^2 - \lambda\eta_j < e_j, \nabla f(x_k^{(j)}) > \\
&+ \frac{L^3\eta_j^2}{2b_j} \parallel (1-\lambda)x_k^{(j)} - \lambda x_0^{(j)} \parallel^2 + L\eta_j^2\lambda^2 \parallel e_j \parallel^2 \\
&\leq f(x_k^{(j)}) - (\eta_j(1-\lambda) - L\eta_j^2(1-\lambda)^2) \parallel \nabla f(x_k^{(j)}) \parallel^2 - \lambda\eta_j < e_j, \nabla f(x_k^{(j)}) > \\
&+ \frac{L^3\eta_j^2}{2b_j} \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 + L\eta_j^2\lambda^2 \parallel e_j \parallel^2
\end{aligned} \tag{15}$$

Let $\mathbb{E}_j$ denote the expectation $\tilde{\mathcal{I}}_0, \tilde{\mathcal{I}}_1$,..., given $\tilde{\mathcal{N}}_j$ since $\tilde{\mathcal{N}}_j$ is independent of them and let k=$\mathcal{N}_j$ in Inq. 15. As $\tilde{\mathcal{I}}_{k+1}, \tilde{\mathcal{I}}_{k+2}$,... are independent of $x_k^{(j)}$ and taking the expectation with respect to $\mathcal{N}_j$ and using Fubini's theorem, Inq. 15 implies that

$$\begin{aligned}
&\eta_j(1-\lambda)(1-(1-\lambda)L\eta_j)\mathbb{E}_{\mathcal{N}_j}\mathbb{E}_j[\parallel \nabla f(x_{\mathcal{N}_j}^{(j)}) \parallel^2] + \lambda\eta_j\mathbb{E}_{\mathcal{N}_j}\mathbb{E}_j < e_j, \nabla f(x_{\mathcal{N}_j}^{(j)}) > \\
&\leq \mathbb{E}_{\mathcal{N}_j}(\mathbb{E}_j[f(x_{\mathcal{N}_j}^{(j)})] - \mathbb{E}_j[f(x_{\mathcal{N}_{j+1}}^{(j)})]) + \frac{L^3\eta_j^2}{2b_j}\mathbb{E}_{\mathcal{N}_j}\mathbb{E}_j\mathbb{E}[\parallel (1-\lambda)x_{\mathcal{N}_j}^{(j)} - \lambda x_0^{(j)} \parallel^2] + L\lambda^2\eta_j^2 \parallel e_j \parallel^2 \\
&= \frac{b_j}{B_j}(f(x_0^{(j)}) - \mathbb{E}_j\mathbb{E}_{\mathcal{N}_j}[f_{\mathcal{N}_j}^{(j)}]) + \frac{L^3\eta_j^2}{2b_j}\mathbb{E}_j\mathbb{E}_{\mathcal{N}_j}[\parallel (1-\lambda)x_{\mathcal{N}_j}^{(j)} - \lambda x_0^{(j)} \parallel^2] + L\lambda^2\eta_j^2 \parallel e_j \parallel^2
\end{aligned} \tag{16}$$

where the last equation in Inq. 16 follows from Lemma 2. The lemma substitutes $x_{\mathcal{N}_j}^{(j)}(x_0^j)$ by $\tilde{x}_j(\tilde{x}_{j-1})$.

Lemma 6. *Suppose $\eta_j^2L^2B_j < b_j^2$, then under Definition 3,*

$$\begin{aligned}
&(b_j - \frac{\eta_j^2L^2B_j}{4b_j})\mathbb{E}[\parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2] + 2\lambda\eta_jB_j\mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) > \\
&\leq -2\eta_j(1-\lambda)B_j\mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) > + 2(1-\lambda)^2\eta_j^2B_j\mathbb{E}[\parallel \nabla f(\tilde{x}_j) \parallel^2] \\
&+ 2\lambda^2\eta_j^2B_j\mathbb{E}[\parallel e_j \parallel^2]
\end{aligned}$$

Proof. Since $x_{k+1}^{(j)} = x_k^{(j)} - \eta_j v_k^{(j)}$, *we have*

$$\begin{aligned}
&\mathbb{E}_{\tilde{\mathcal{I}}_k}[\| x_{k+1}^{(j)} - x_0^{(j)} \|^2] \\
&= \| x_k^{(j)} - x_0^{(j)} \|^2 - 2\eta_j < \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k^{(j)}, (x_k^{(j)} - x_0^{(j)}) > + \eta_j^2 \mathbb{E}_{\tilde{\mathcal{I}}_k} \| v_k^{(j)} \|^2 \\
&= \| x_k^{(j)} - x_0^{(j)} \|^2 - 2(1-\lambda)\eta_j < \nabla f(x_k^{(j)}), (x_k^{(j)} - x_0^{(j)}) > - 2\lambda\eta_j < e_j, (x_k^{(j)} - x_0^{(j)}) > + \eta_j^2 \mathbb{E}_{\tilde{\mathcal{I}}_k} \| v_k^{(j)} \|^2 \\
&\leq (1 + \frac{\eta_j^2 L^2}{4b_j}) \| x_k^{(j)} - x_0^{(j)} \|^2 - 2\eta_j(1-\lambda) < \nabla f(x_k^{(j)}), x_k^{(j)} - x_0^{(j)} > \\
&- 2\lambda\eta_j < e_j, (x_k^{(j)} - x_0^{(j)}) > + 2(1-\lambda)^2 \eta_j^2 \| \nabla f(x_k^{(j)}) \|^2 + 2\lambda^2 \eta_j^2 \| e_j \|^2 .
\end{aligned} \tag{17}$$

where the last inequality follows from Lemma 3. Using the same notation $\mathbb{E}_j$ *from Eq 7 we have*

$$\begin{aligned}
&2\eta_j(1-\lambda)\mathbb{E}_j < \nabla f(x_k^{(j)}), (x_k^{(j)} - x_0^{(j)}) > + 2\lambda\eta_j \mathbb{E}_j < e_j, (x_k^{(j)} - x_0^{(j)}) > \\
&\leq (1 + \frac{\eta_j^2 L^2}{4b_j}) \mathbb{E}_j \| x_k^{(j)} - x_0^{(j)} \|^2 - \mathbb{E}_j \| x_{k+1}^{(j)} - x_0^{(j)} \|^2 + 2(1-\lambda)^2 \eta_j^2 \| \nabla f(x_k^{(j)}) \|^2 + 2\lambda\eta_j^2 \| e_j \|^2
\end{aligned} \tag{18}$$

Let $k = N_j$, *and using Fubini's theorem, we have,*

$$\begin{aligned}
&2(1-\lambda)\eta_j \mathbb{E}_{N_j} \mathbb{E}_j < \nabla f(x_{N_j}^{(j)}), (x_{N_j}^{(j)} - x_0^{(j)}) > + 2\lambda\eta_j \mathbb{E}_{N_j} \mathbb{E}_j < e_j, (x_{N_j}^{(j)} - x_0^{(j)}) > \\
&\leq (1 + \frac{\eta_j L^2}{4b_j}) \mathbb{E}_{N_j} \mathbb{E}_j \| x_{N_j}^{(j)} - x_0^{(j)} \|^2 - \mathbb{E}_{N_j} \mathbb{E}_j \| x_{N_j+1}^{(j)} - x_0^{(j)} \|^2 \\
&+ 2(1-\lambda)^2 \eta_j^2 \mathbb{E}_{N_j} \| \nabla f(x_{N_j}^{(j)}) \|^2 + 2\lambda^2 \eta_j^2 \| e_j \|^2 \\
&= (-\frac{b_j}{B_j} + \frac{\eta_j^2 L^2}{4b_j}) \mathbb{E}_{N_j} \mathbb{E}_j \| x_{N_j}^{(j)} - x_0^{(j)} \|^2 + 2(1-\lambda)^2 \eta_j^2 \mathbb{E}_{N_j} \| \nabla f(x_{N_j}^{(j)}) \|^2 + 2\lambda^2 \eta_j^2 \| e_j \|^2 .
\end{aligned} \tag{19}$$

The lemma is then proved by substituting $x_{N_j}^{(j)}(x_0^{(j)})$ *by* $\tilde{x}_j(\tilde{x}_{j-1})$.

Lemma 7.

$$b_j \mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) > = -\eta_j(1-\lambda) B_j \mathbb{E} < e_j, \nabla f(\tilde{x}_j) > - \lambda^2 \eta_j B_j \mathbb{E} \| e_j \|^2$$

Proof. Let $M_k^{(j)} = < e_j, (x_k^{(j)} - x_0^{(j)}) >$, *then we have*

$$\mathbb{E}_{N_j} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) > = \mathbb{E}_{N_j} M_{N_j}^{(j)}. \tag{20}$$

Since N_j *is independent of* $(x_0^{(j)}, e_j)$, *it has*

$$\mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) > = \mathbb{E} M_{N_j}^{(j)}. \tag{21}$$

Also $M_0^{(j)} = 0$, *then we have*

$$\begin{aligned}
&\mathbb{E}_{\tilde{\mathcal{I}}_k}(M_{k+1}^{(j)} - M_k^{(j)}) \\
&= \mathbb{E}_{\tilde{\mathcal{I}}_k} < e_j, (x_{k+1}^{(j)} - x_k^{(j)}) > \\
&= -\eta_j < e_j, \mathbb{E}_{\tilde{\mathcal{I}}_k}[v_k^{(j)}] > .
\end{aligned} \tag{22}$$

Using the same notation $\mathbb{E}_j$ in Lemma 5 and Lemma 6, we have

$$\mathbb{E}_j(M_{k+1}^{(j)} - M_k^{(j)}) = -\eta_j(1-\lambda) < e_j, \mathbb{E}_j \nabla f(x_k^{(j)}) > -\lambda^2 \eta_j \parallel e_j \parallel^2 . \tag{23}$$

Let $k = N_j$ in Eq.23. Using Fubini's theorem and Lemma 4, we have,

$$\frac{b_j}{B_j}\mathbb{E}_{N_j} M_{N_j}^{(j)} = -\eta_j(1-\lambda) < e_j, \mathbb{E}_{N_j}\mathbb{E}_j \nabla f(x_k^{(j)}) > -\eta_j \parallel e_j \parallel^2 . \tag{24}$$

The lemma is then proved by substituting $x_{N_j}^{(j)}(x_0^{(j)})$ by $\tilde{x}_j(\tilde{x}_{j-1})$.

Proof of Theorem 1

Let $\eta_j L = \gamma(\frac{b_j}{B_j})^\alpha$ $(0 \le \alpha \le 1)$ and $B_j \ge b_j \ge B_j^\beta$ $(0 \le \beta \le 1)$ for all j. Suppose $0 < \gamma \le \frac{1}{3}$, then under Definition 3, the output $\tilde{x}_j$ of Alg 2 we have

$$\mathbb{E}\|\nabla f(\tilde{x}_j)\|^2 \le \frac{2L}{\gamma\theta} \cdot (\frac{b_j}{B_j})^{1-\alpha}\mathbb{E}\left(f(\tilde{x}_{j-1}) - f(\tilde{x_j})\right) + \frac{2\lambda^4 I(B_j < n)\mathcal{S}^*}{\theta B_j^{1-2\alpha}},$$

where $I(B_j < n) \ge \frac{n - B_j}{(n-1)B_j}$, $\mathcal{S}$ is defined in Eq.3, $\lambda = \frac{1}{2}$ and $\theta = 2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1})(1-\lambda)^2 - 1.29(1-\lambda)^2$.

> *Proof Sketch*: Combine two equations in Lemma 5 and Lemma 6, we can achieve a upper bound of unbiased version gradient in single epoch. And further use Lemma 4, the final result of Theorem 2 can be achieved.

Proof. Multiplying Lemma 3 by 2 and Lemma 6 by $\frac{b_j}{\eta_j B_j}$ and summing them, then we have,

$$\begin{aligned}
&2\eta_j B_j(1-\lambda)(1-(1-\lambda)L\eta_j - \frac{(1-\lambda)b_j}{B_j})\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 + \frac{b_j^3 - \eta_j^2 L^2 b_j B_j - \eta_j^3 L^3 B_j^2}{8 b_j \eta_j B_j}\mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2 \\
&+ 2\lambda\eta_j B_j \mathbb{E} < e_j, \nabla f(\tilde{x}_j) > + 2\lambda b_j \mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) > \\
&= 2\eta_j B_j(1-\lambda)(1-(1-\lambda)L\eta_j - \frac{(1-\lambda)b_j}{B_j})\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \\
&+ \frac{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2}{8 b_j \eta_j B_j}\mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2 - 2\frac{\lambda^3}{(1-\lambda)}\eta_j B_j \mathbb{E} \parallel e_j \parallel^2 \text{ (Lemma 7)} \\
&\le -2(1-\lambda)b_j \mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) > + 2b_j \mathbb{E}(f(\tilde{x}_{j-1}) - f(\tilde{x}_j)) + (2\lambda^2 L\eta_j^2 B_j + 2\lambda^2 \eta_j b_j)\mathbb{E} \parallel e_j \parallel^2
\end{aligned} \tag{25}$$

Using the fact that $2 < q, p > \leq \beta \parallel q \parallel^2 + \frac{1}{\beta} \parallel p \parallel^2$ for any $\beta > 0$, $-2(1-\lambda)b_j\mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) >$ in Inq. 25 can be bounded as

$$\begin{aligned} & -2(1-\lambda)b_j\mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) > \\ & \leq (1-\lambda)(\frac{(1-\lambda)b_j\eta_j B_j}{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2} b_j^2 \mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \\ & + \frac{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2}{8(1-\lambda)b_j\eta_j B_j} \mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2) \end{aligned} \tag{26}$$

Then Inq. 25 can be expressed as

$$\begin{aligned} & \frac{\eta_j B_j}{b_j}(2(1-\lambda) - 2(1-\lambda)^2 L\eta_j - 2(1-\lambda)^2 \frac{b_j}{B_j} - \frac{(1-\lambda)^2 b_j^3}{b_j^3 - 8(1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2}) \\ & \mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \\ & \leq 2\mathbb{E}(f(\tilde{x}_{j-1}) - f(\tilde{x}_j)) + \frac{2\eta_j B_j \lambda^2}{b_j}(\frac{\lambda^2}{(1-\lambda)} + \eta_j L + \frac{b_j}{B_j})\mathbb{E} \parallel e_j \parallel^2 . \end{aligned} \tag{27}$$

Since $\eta_j L = \gamma(\frac{b_j}{B_j})^\alpha$, $b_j \geq 1$ and $B_j \geq b_j \geq B_j^\beta$ where $\alpha > 0$ and $\beta \geq 0$ by Eq. 7, and $\lambda = \frac{1}{2}, \gamma = \frac{1}{3}$, one part in left hand side of above inequality can be simplified and positive as following:

$$\begin{aligned} & b_j^3 - 8(1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2 \\ & = b_j^3(1 - 8(1-\lambda)^2\gamma^2 \frac{b_j^{2\alpha-2}}{B_j^{2\alpha-1}} - (1-\lambda)^2\gamma^3 \frac{b_j^{3\alpha-3}}{B_j^{3\alpha-2}}) \\ & \geq b_j^3(1 - 8(1-\lambda)^2\gamma^2 B_j^{-1} - (1-\lambda)^2\gamma^3 B_j^{-1}) \geq 0.77 b_j^3 \end{aligned} \tag{28}$$

By Eq.28, the left side of Inq. 27 can be simplified since the factor of geometry distribution $\gamma \geq 0$ as

$$\begin{aligned} & \frac{\eta_j B_j}{b_j}(2(1-\lambda) - 2(1-\lambda)^2 L\eta_j - 2(1-\lambda)^2 \frac{b_j}{B_j} - \frac{(1-\lambda)^2 b_j^3}{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2}) \\ & \mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \\ & \geq \frac{\gamma}{L} B_j^{\alpha\beta-\alpha-\beta+1} \left(2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2\frac{b_j}{B_j})(1-\lambda)^2 - 1.29(1-\lambda)^2\right) \mathbb{E}||\nabla f(\tilde{x}_j)||^2 \\ & \geq \frac{\gamma}{L} B_j^{\alpha\beta-\alpha-\beta+1} \left(2(1-\lambda) - (2\gamma+2)B_j^{-1}(1-\lambda)^2 - 1.29(1-\lambda)^2\right) \mathbb{E}||\nabla f(\tilde{x}_j)||^2 \end{aligned} \tag{29}$$

Then Eq.27 can be simplified by Eq.29 as

$$\begin{aligned}\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 &\leq \frac{2\mathbb{E}[f(\tilde{x}_{j-1}) - f(\tilde{x}_j)] + 2\frac{\gamma}{L}B_j^{\alpha\beta-\alpha-\beta+1}\lambda^2(\frac{\lambda^2}{(1-\lambda)} + B_j^{\alpha\beta-\alpha}\gamma + B_j^{\beta-\alpha}L)\mathbb{E}||e_j||^2}{\frac{\gamma}{L}B_j^{\alpha\beta-\alpha-\beta+1}\left(2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1})(1-\lambda)^2 - 1.29(1-\lambda)^2\right)} \\ &\leq \frac{\overbrace{2\mathbb{E}(f(\tilde{x}_{j-1} - f(\tilde{x}_j))}^{\text{positive by Lemma 2}} + \overbrace{2\frac{\gamma}{L}\lambda^2 B_j^{\alpha\beta-\alpha-\beta+1}B_j^{2\alpha}\mathbb{E} \parallel e_j \parallel^2}^{\text{positive}}}{\frac{\gamma}{L}B_j^{\alpha\beta-\alpha-\beta+1}\left(2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1})(1-\lambda)^2 - 1.29(1-\lambda)^2\right)},\end{aligned} \tag{30}$$

Then, using Lemma 4, Inq. 30 can be rewritten as

$$\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \leq \frac{2\mathbb{E}(f(\tilde{x}_{j-1} - f(\tilde{x}_j)) + 2\frac{\gamma}{L}\lambda^4 B_j^{\alpha\beta+\alpha-\beta}I(B_j < n)\mathcal{S}^*}{\frac{\gamma}{L}B_j^{\alpha\beta-\alpha-\beta+1}\left(2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1})(1-\lambda)^2 - 1.29(1-\lambda)^2\right)}. \tag{31}$$

Since the learning rate $\eta \leq \frac{1}{3L}$ was determined by [21, 19] that $\gamma \leq \frac{1}{3}$ which guarantees the convergence in non-convex case. Thus $\gamma \leq \frac{1}{3}$ as a upper bound is considered in our biased and unbiased cases.

B.2 Biased Estimator Version

We still provide all useful lemmas we will applied in our proof of theorems at first, and provide a proof sketch for guidance. For the biased estimation version, we still start by bounding the gradient $\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2$ in Lemma 8 and the variance $\mathbb{E}_{\mathcal{I}_j} \parallel e_j \parallel^2$ in Lemma 9.

Lemma 8. *Under Definition 3,*

$$\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 \leq \frac{(1-\lambda)^2L^2}{b_j} \parallel x_k^{(j)} - x_0^{(j)} \parallel + 2(1-\lambda)^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + 2 \parallel e_j \parallel^2 .$$

Proof. Using the fact that for a random variable Z $\mathbb{E} \parallel Z \parallel^2 = \parallel Z - \mathbb{E}Z \parallel^2 + \parallel \mathbb{E}Z \parallel^2$, we have

$$\begin{aligned}&\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 = \mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} - \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k^{(j)} \parallel^2 + \parallel \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k^{(j)} \parallel^2 \\ &= \mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel (1-\lambda)(\nabla f_{\tilde{\mathcal{I}}_k}(x_k^{(j)}) - \nabla f_{\tilde{\mathcal{I}}_k}(x_0^{(j)})) - (1-\lambda)(\nabla f(x_k^{(j)}) - \nabla f x_0^{(j)}) \parallel^2 \\ &+ \parallel (1-\lambda)\nabla f(x_k^{(j)}) + e_j \parallel^2 \\ &\leq (1-\lambda)^2\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel \nabla f_{\tilde{\mathcal{I}}_k}(x_k^{(j)}) - \nabla f_{\tilde{\mathcal{I}}_k}(x_0^{(j)}) - (\nabla f(x_k^{(j)}) - \nabla f x_0^{(j)}) \parallel^2 \\ &+ 2(1-\lambda)^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + 2 \parallel e_j \parallel^2 .\end{aligned} \tag{32}$$

By Lemma 1, the first part of inequality in Eq.32 can be rewritten as,

$$\begin{aligned}
&(1-\lambda)^2\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel \nabla f_{\tilde{\mathcal{I}}_k}(x_k^{(j)}) - \nabla f_{\tilde{\mathcal{I}}_k}(x_0^{(j)}) - (\nabla f(x_k^{(j)}) - \nabla f x_0^{(j)}) \parallel^2 \\
&\leq \frac{(1-\lambda)^2}{b_j} \cdot \frac{1}{n}\sum_{i=1}^{n} \parallel \nabla f_i(x_k^{(j)}) - \nabla f_i(x_0^{(j)}) - (\nabla f(x_k^{(j)}) - \nabla f(x_0^{(j)})) \parallel^2 \\
&= \frac{(1-\lambda)^2}{b_j} \cdot (\frac{1}{n}\sum_{i=1}^{n} \parallel \nabla f_i(x_k^{(j)}) - \nabla f_i(x_0^{(j)}) \parallel^2 - \parallel (\nabla f(x_k^{(j)}) - \nabla f(x_0^{(j)})) \parallel^2) \\
&\leq \frac{(1-\lambda)^2}{b_j} \cdot \frac{1}{n}\sum_{i=1}^{n} \parallel \nabla f_i(x_k^{(j)}) - \nabla f_i(x_0^{(j)}) \parallel^2 \\
&\leq \frac{(1-\lambda)^2}{b_j} \cdot L^2 \parallel x_k^{(j)} - x_0^{(j)} \parallel^2
\end{aligned} \tag{33}$$

where the last line is based on Definition 3, then the bound of the gradient can be written as,

$$\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 \leq \frac{(1-\lambda)^2 L^2}{b_j} \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 + 2(1-\lambda)^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + 2 \parallel e_j \parallel^2 . \tag{34}$$

Lemma 9.

$$\begin{aligned}
\mathbb{E}_{\mathcal{I}_j} \parallel e_j \parallel^2 &\leq (1-\lambda)^2 \frac{I(B_j < n)}{B_j} \mathcal{S}^* + (1-2\lambda)^2 \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2 \\
&= \mathbb{E}_{\mathcal{I}_j} \parallel \tilde{e_j} \parallel^2 + (1-2\lambda)^2 \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2
\end{aligned}$$

where $(1-\lambda)^2 \frac{I(B_j < n)}{B_j} \mathcal{S}^* = \mathbb{E}_{\mathcal{I}_j} \parallel \tilde{e_j} \parallel^2$ *and* $0 < \lambda < 1$.

Proof. Based on Lemma 1 and the observation that $\tilde{x}_{j-1}$ is independent of

$$
\begin{aligned}
\mathbb{E}_{\mathcal{I}_j} \parallel e_j \parallel^2 &= \frac{n-B_j}{(n-1)B_j} \cdot \frac{1}{n}\sum_{i=1}^{n} \parallel (1-\lambda)\nabla f_i(\tilde{x}_{j-1}) - \lambda \nabla f(\tilde{x}_{j-1}) \parallel^2 \\
&= \frac{n-B_j}{(n-1)B_j} \mathbb{E}_{\mathcal{I}_j} \parallel (1-\lambda)\nabla f_i(\tilde{x}_{j-1}) - \lambda \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})] \parallel^2 \\
&= \frac{n-B_j}{(n-1)B_j} \mathbb{E}_{\mathcal{I}_j} \left[(1-\lambda)^2 \nabla f_i(\tilde{x}_{j-1})^2 - (2\lambda - 3\lambda^2)\mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2\right] \\
&= \frac{n-B_j}{(n-1)B_j} \left[\underbrace{(1-\lambda)^2 \mathbb{E}_{\mathcal{I}_j}\left[\nabla f_i(\tilde{x}_{j-1})^2 - \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2\right]}_{Unbiased} + \underbrace{(1-2\lambda)^2 \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2}_{Extra/term} \right] \\
&= \frac{n-B_j}{(n-1)B_j} \cdot \left((1-\lambda)^2 \frac{1}{n}\sum_{i=1}^{n} \parallel \nabla f_i(\tilde{x}_{j-1}) - \nabla f(\tilde{x}_{j-1}) \parallel^2 + (1-2\lambda)^2 \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2 \right) \\
&\leq (1-\lambda)^2 \frac{n-B_j}{(n-1)B_j} \cdot \mathcal{S}^* + \frac{n-B_j}{(n-1)B_j}(1-2\lambda)^2 \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2 \\
&\leq (1-\lambda)^2 \frac{I(B_j < n)}{B_j} \mathcal{S}^* + (1-2\lambda)^2 \mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]^2,
\end{aligned}
\tag{35}
$$

where the upper bound of the variance of the stochastic gradients $\mathcal{S}^* = \frac{1}{n}\sum_{i=1}^{n} \parallel \nabla f_i(\tilde{x}_{j-1}) - \nabla f(\tilde{x}_{j-1}) \parallel^2$.In above function, as $\nabla f(\tilde{x}_{j-1})$ is the expectation value of $\nabla f_i(\tilde{x}_{j-1})$, we use $\mathbb{E}_{\mathcal{I}_j}[\nabla f_i(\tilde{x}_{j-1})]$ to alternative $\nabla f(\tilde{x}_{j-1})$ for easily understanding later proof. Meanwhile, We can achieve the third equation in above function since the fact that $\mathbb{E}[(1-\lambda)Z - \lambda\mathbb{E}[Z]]^2 = (1-\lambda)^2\mathbb{E}[Z^2] - (2\lambda - 3\lambda^2)\mathbb{E}[Z]^2 = \mathbb{E}[(1-\lambda)^2 Z^2 - (2\lambda - 3\lambda^2)\mathbb{E}[Z]^2]$.

Lemma 10. *Suppose $\eta_j L < 1$, then under Definition 3,*

$$
\begin{aligned}
&(1-\lambda)(1-(1-\lambda)L\eta_j)\eta_j B_j \mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 + \eta_j B_j \mathbb{E} < e_j, \nabla f(\tilde{x}_j) > \\
&\leq b_j \mathbb{E}(f(\tilde{x}_{j-1}) - f(\tilde{x}_j)) + \frac{(1-\lambda)^2 \eta_j^2 B_j L^3}{2b_j} \mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2 + L\eta_j^2 B_j \mathbb{E} \parallel e_j \parallel^2 .
\end{aligned}
$$

where $\mathbb{E}$ denotes the expectation with respect to all randomness.

Proof. By Definition 3, we have

$$\begin{aligned}
&\mathbb{E}_{\tilde{\mathcal{I}}_k}[f(x_{k+1}^{(j)})] \leq f(x_k^{(j)}) - \eta_j < \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k, \nabla f(x_k^{(j)}) > + \frac{L\eta_j^2}{2}\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k \parallel^2 \\
&= f(x_k^{(j)}) - \eta_j < ((1-\lambda)\nabla f(x_k^{(j)}) + e_j), \nabla f(x)_k^{(j)}) > + \frac{L\eta_j^2}{2}\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k \parallel^2 \\
&\leq f(x_k^{(j)}) - \eta_j(1-\lambda) \parallel \nabla f(x_k^{(j)}) \parallel^2 - \eta_j < e_j, \nabla f(x_k^{(j)}) > \\
&+ \frac{L^3\eta_j^2(1-\lambda)^2}{2b_j} \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 + L\eta_j^2(1-\lambda)^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + L\eta_j^2 \parallel e_j \parallel^2 \\
&= f(x_k^{(j)}) - (\eta_j(1-\lambda) - L\eta_j^2(1-\lambda)^2) \parallel \nabla f(x_k^{(j)}) \parallel^2 \\
&- \eta_j < e_j, \nabla f(x_k^{(j)}) > + \frac{L^3\eta_j^2(1-\lambda)^2}{2b_j} \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 + L\eta_j^2 \parallel e_j \parallel^2
\end{aligned} \tag{36}$$

Let $\mathbb{E}_j$ denote the expectation $\tilde{\mathcal{I}}_0, \tilde{\mathcal{I}}_1$,..., given $\tilde{\mathcal{N}_j}$ since $\tilde{\mathcal{N}_j}$ is independent of them and let k=$\mathcal{N}_j$ in Inq 36. As $\tilde{\mathcal{I}}_{k+1}, \tilde{\mathcal{I}}_{k+2}$,... are independent of $x_k^{(j)}$ and taking the expectation with respect to $\mathcal{N}_j$ and using Fubini's theorem, Inq. 36 implies that

$$\begin{aligned}
&\eta_j(1-\lambda)(1-(1-\lambda)L\eta_j)\mathbb{E}_{\mathcal{N}_j}\mathbb{E}_j[\parallel \nabla f(x_{\mathcal{N}_j}^{(j)}) \parallel^2] + \eta_j\mathbb{E}_{\mathcal{N}_j}\mathbb{E}_j < e_j, \nabla f(x_{\mathcal{N}_j}^{(j)}) > \\
&\leq \mathbb{E}_{\mathcal{N}_j}(\mathbb{E}_j[f(x_{\mathcal{N}_j}^{(j)})] - \mathbb{E}_j[f(x_{\mathcal{N}_{j+1}}^{(j)})]) + \frac{L^3\eta_j^2(1-\lambda)^2}{2b_j}\mathbb{E}_{\mathcal{N}_j}\mathbb{E}_j\mathbb{E}[\parallel x_{\mathcal{N}_j}^{(j)} - x_0^{(j)} \parallel^2] + L\eta_j^2 \parallel e_j \parallel^2 \\
&= \frac{b_j}{B_j}(f(x_0^{(j)}) - \mathbb{E}_j\mathbb{E}_{\mathcal{N}_j}[f_{\mathcal{N}_j}^{(j)}]) + \frac{L^3\eta_j^2(1-\lambda)^2}{2b_j}\mathbb{E}_j\mathbb{E}_{\mathcal{N}_j}[\parallel x_{\mathcal{N}_j}^{(j)} - x_0^{(j)} \parallel^2] + L\eta_j^2 \parallel e_j \parallel^2
\end{aligned} \tag{37}$$

where the last equation in Inq. 37 follows from Lemma 2. The lemma substitutes $x_{\mathcal{N}_j}^{(j)}(x_0^j)$ by $\tilde{x}_j(\tilde{x}_{j-1})$.

Lemma 11. *Suppose $\eta_j^2L^2B_j < b_j^2$, then under Definition lsmooth1,*

$$\begin{aligned}
&(b_j - \frac{(1-\lambda)^2\eta_j^2L^2B_j}{b_j})\mathbb{E}[\parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2] + 2\eta_jB_j\mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) > \\
&\leq -2(1-\lambda)\eta_jB_j\mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) > + 2(1-\lambda)^2\eta_j^2B_j\mathbb{E}[\parallel \nabla f(\tilde{x}_j) \parallel^2] + 2\eta_j^2B_j\mathbb{E}[\parallel e_j \parallel^2]
\end{aligned}$$

Proof. Since $x_{k+1}^{(j)} = x_k^{(j)} - \eta_j v_k^{(j)}$, we have

$$\begin{aligned}
&\mathbb{E}_{\tilde{\mathcal{I}}_k}[\parallel x_{k+1}^{(j)} - x_0^{(j)} \parallel^2] \\
&= \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 - 2\eta_j < \mathbb{E}_{\tilde{\mathcal{I}}_k} v_k^{(j)}, (x_k^{(j)} - x_0^{(j)}) > + \eta_j^2\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 \\
&= \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 - 2\eta_j(1-\lambda) < \nabla f(x_k^{(j)}), (x_k^{(j)} - x_0^{(j)}) > - 2\eta_j < e_j, (x_k^{(j)} - x_0^{(j)}) > + \eta_j^2\mathbb{E}_{\tilde{\mathcal{I}}_k} \parallel v_k^{(j)} \parallel^2 \\
&\leq (1 + \frac{(1-\lambda)^2\eta_j^2L^2}{b_j}) \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 - 2\eta_j(1-\lambda) < \nabla f(x_k^{(j)}), x_k^{(j)} - x_0^{(j)} > - 2\eta_j < e_j, (x_k^{(j)} - x_0^{(j)}) > \\
&+ 2(1-\lambda)^2\eta_j^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + 2\eta_j^2 \parallel e_j \parallel^2 .
\end{aligned} \tag{38}$$

where the last inequality is based on Lemma 8. Using the same notation $\mathbb{E}_j$ in Eq. 7 we have

$$2\eta_j(1-\lambda)\mathbb{E}_j < \nabla f(x_k^{(j)}), (x_k^{(j)} - x_0^{(j)}) > +2\eta_j\mathbb{E}_j < e_j, (x_k^{(j)} - x_0^{(j)}) >$$
$$\leq (1 + \frac{(1-\lambda)^2\eta_j^2 L^2}{b_j})\mathbb{E}_j \parallel x_k^{(j)} - x_0^{(j)} \parallel^2 - \mathbb{E}_j \parallel x_{k+1}^{(j)} - x_0^{(j)} \parallel^2 + 2(1-\lambda)^2\eta_j^2 \parallel \nabla f(x_k^{(j)}) \parallel^2 + 2\eta_j^2 \parallel e_j \parallel^2 \tag{39}$$

Let $k = N_j$, and using Fubini's theorem, we have,

$$2\eta_j(1-\lambda)\mathbb{E}_{N_j}\mathbb{E}_j < \nabla f(x_{N_j}^{(j)}), (x_{N_j}^{(j)} - x_0^{(j)}) > +2\eta_j\mathbb{E}_{N_j}\mathbb{E}_j < e_j, (x_{N_j}^{(j)} - x_0^{(j)}) >$$
$$\leq (1 + \frac{(1-\lambda)^2\eta_j L^2}{b_j})\mathbb{E}_{N_j}\mathbb{E}_j \parallel x_{N_j}^{(j)} - x_0^{(j)} \parallel^2 - \mathbb{E}_{N_j}\mathbb{E}_j \parallel x_{N_j+1}^{(j)} - x_0^{(j)} \parallel^2$$
$$+ 2(1-\lambda)^2\eta_j^2\mathbb{E}_{N_j} \parallel \nabla f(x_{N_j}^{(j)}) \parallel^2 + 2\eta_j^2 \parallel e_j \parallel^2$$
$$= (-\frac{b_j}{B_j} + \frac{(1-\lambda)^2\eta_j^2 L^2}{b_j})\mathbb{E}_{N_j}\mathbb{E}_j \parallel x_{N_j}^{(j)} - x_0^{(j)} \parallel^2 + 2(1-\lambda)^2\eta_j^2\mathbb{E}_{N_j} \parallel \nabla f(x_{N_j}^{(j)}) \parallel^2 + 2\eta_j^2 \parallel e_j \parallel^2 . \tag{40}$$

The lemma is then proved by substituting $x_{N_j}^{(j)}(x_0^{(j)})$ by $\tilde{x}_j(\tilde{x}_{j-1})$.

Lemma 12.

$$b_j\mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) >= -\eta_j(1-\lambda)B_j\mathbb{E} < e_j, \nabla f(\tilde{x}_j) > -\eta_j B_j\mathbb{E} \parallel e_j \parallel^2$$

Proof. Let $M_k^{(j)} =< e_j, (x_k^{(j)} - x_0^{(j)}) >$, then we have

$$\mathbb{E}_{N_j} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) >= \mathbb{E}_{N_j} M_{N_j}^{(j)}.$$

Since N_j is independent of $(x_0^{(j)}, e_j)$, it has

$$\mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) >= \mathbb{E} M_{N_j}^{(j)}. \tag{41}$$

Also $M_0^{(j)} = 0$, then we have

$$\begin{aligned} &\mathbb{E}_{\tilde{\mathcal{I}}_k}(M_{k+1}^{(j)} - M_k^{(j)}) \\ &= \mathbb{E}_{\tilde{\mathcal{I}}_k} < e_j, (x_{k+1}^{(j)} - x_k^{(j)}) >= -\eta_j < e_j, \mathbb{E}_{\tilde{\mathcal{I}}_k}[v_k^{(j)}] > \\ &= -\eta_j(1-\lambda) < e_j, \nabla f(x_k^{(j)}) > -\eta_j \parallel e_j \parallel^2 . \end{aligned} \tag{42}$$

Using the same notation $\mathbb{E}_j$ in Eq. 7, we have

$$\mathbb{E}_j(M_{k+1}^{(j)} - M_k^{(j)}) = -\eta_j(1-\lambda) < e_j, \mathbb{E}_j\nabla f(x_k^{(j)}) > -\eta_j \parallel e_j \parallel^2 . \tag{43}$$

Let $k = N_j$ in Eq.43. Using Fubini's theorem and Lemma 2, we have,

$$\frac{b_j}{B_j}\mathbb{E}_{N_j} M_{N_j}^{(j)} = -\eta_j(1-\lambda) < e_j, \mathbb{E}_{N_j}\mathbb{E}_j\nabla f(x_k^{(j)}) > -\eta_j \parallel e_j \parallel^2 . \tag{44}$$

The lemma is then proved by substituting $x_{N_j}^{(j)}(x_0^{(j)})$ by $\tilde{x}_j(\tilde{x}_{j-1})$.

Proof of Theorem 3

let $\eta_j L = \gamma(\frac{b_j}{B_j})^\alpha$ $(0 \leq \alpha \leq 1)$. Suppose $0 < \gamma \leq \frac{1}{3}$ and $B_j \geq b_j \geq B_j^\beta$ $(0 \leq \beta \leq 1)$ for all j, then under Definition 3, the output $\tilde{x}_j$ of Alg 3 we have,

$$\mathbb{E}\|\nabla f(\tilde{x}_j)\|^2 \leq \frac{2L}{\gamma\Theta} \cdot (\frac{b_j}{B_j})^{1-\alpha}\mathbb{E}\left(f(\tilde{x}_{j-1}) - f(\tilde{x_j})\right) + \frac{2(1-\lambda)^2 I(B_j < n)\mathcal{S}^*}{\Theta B_j^{1-2\alpha}},$$

where $I(B_j < n) \geq \frac{n - B_j}{(n-1)B_j}$, $\mathcal{S}$ is defined in Eq.3, $0 < \lambda < 1$ and $\Theta = 2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1} - 4LB_j^{2\alpha-2})(1-\lambda)^2 - 1.16(1-\lambda)^2$.

Proof Sketch: Combine two equations in Lemma 10 and Lemma 11, we can achieve a upper bound of biased version gradient in single epoch. And further use Lemma 9, the final result of Theorem 4 can be achieved.

Proof. Multiplying Eq.10 by 2 and Eq.11 by $\frac{b_j}{\eta_j B_j}$ and summing them, then we have,

$$\begin{aligned}
&2\eta_j B_j(1-\lambda)(1-(1-\lambda)L\eta_j - \frac{(1-\lambda)b_j}{B_j})\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \\
&+ \frac{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2}{b_j\eta_j B_j}\mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2 \\
&+ 2\eta_j B_j \mathbb{E} < e_j, \nabla f(\tilde{x}_j) > + 2b_j \mathbb{E} < e_j, (\tilde{x}_j - \tilde{x}_{j-1}) > \\
&= 2\eta_j B_j(1-\lambda)(1-(1-\lambda)L\eta_j - \frac{(1-\lambda)b_j}{B_j} + \frac{(2\lambda-1)^2}{2\eta_j B_j(1-\lambda)})\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \\
&+ \frac{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2}{b_j\eta_j B_j}\mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2 - 2\eta_j B_j \mathbb{E} \parallel \tilde{e_j} \parallel^2 \text{ (Lemma 12)} \\
&\leq -2(1-\lambda)b_j \mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) > + 2b_j \mathbb{E}(f(\tilde{x}_{j-1}) - f(\tilde{x}_j)) + (2L\eta_j^2 B_j + 2\eta_j b_j)\mathbb{E} \parallel \tilde{e_j} \parallel^2
\end{aligned} \tag{45}$$

Using the fact that $2 < q, p > \leq \beta \parallel q \parallel^2 + \frac{1}{\beta} \parallel p \parallel^2$ for any $\beta > 0$, $-2b_j\mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) >$ in Inq. 45 can be bounded as

$$\begin{aligned}
&- 2(1-\lambda)b_j \mathbb{E} < \nabla f(\tilde{x}_j), (\tilde{x}_j - \tilde{x}_{j-1}) > \\
&\leq (1-\lambda)(\frac{(1-\lambda)b_j\eta_j B_j}{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2} b_j^2 \mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \\
&+ \frac{b_j^3 - (1-\lambda)^2\eta_j^2 L^2 b_j B_j - (1-\lambda)^2\eta_j^3 L^3 B_j^2}{(1-\lambda)b_j\eta_j B_j}\mathbb{E} \parallel \tilde{x}_j - \tilde{x}_{j-1} \parallel^2)
\end{aligned} \tag{46}$$

Then Inq. 45 can be rewritten as

$$\begin{aligned}
&\frac{\eta_j B_j}{b_j}(2(1-\lambda)-2(1-\lambda)^2 L\eta_j-2(1-\lambda)^2\frac{b_j}{B_j}+\frac{(2\lambda-1)^2}{\eta_j B_j}\\
&-\frac{(1-\lambda)^2 b_j^3}{b_j^3-(1-\lambda)^2\eta_j^2L^2b_jB_j-(1-\lambda)^2\eta_j^3L^3B_j^2})\mathbb{E}\parallel\nabla f(\tilde{x}_j)\parallel^2 \qquad (47)\\
&\leq 2\mathbb{E}(f(\tilde{x}_{j-1})-f(\tilde{x}_j))+\frac{2\eta_jB_j}{b_j}(1+\eta_jL+\frac{b_j}{B_j})\mathbb{E}\parallel\tilde{e_j}\parallel^2.
\end{aligned}$$

Since $\eta_j L=\gamma(\frac{b_j}{B_j})^\alpha$, $b_j\geq 1$ and $B_j\geq b_j\geq B_j^\beta$ where $0<\alpha\leq 1, 0\leq\beta\leq 1$, we have

$$\begin{aligned}
&b_j^3-(1-\lambda)^2\eta_j^2L^2b_jB_j-(1-\lambda)^2\eta_j^3L^3B_j^2\\
&=b_j^3(1-(1-\lambda)^2\gamma^2\frac{b_j^{2\alpha-2}}{B_j^{2\alpha-1}}-(1-\lambda)^2\gamma^3\frac{b_j^{3\alpha-3}}{B_j^{3\alpha-2}}) \qquad (48)\\
&=b_j^3(1-(1-\lambda)^2\gamma^2B_j^{-1}-(1-\lambda)^2\gamma^3B_j^{-1})\geq 0.86b_j^3
\end{aligned}$$

By Eq. 48, the left side of Inq. 47 can be simplified as

$$\begin{aligned}
&\frac{\eta_jB_j}{b_j}(2(1-\lambda)-2(1-\lambda)^2L\eta_j-2(1-\lambda)^2\frac{b_j}{B_j}+\frac{(2\lambda-1)^2}{\eta_jB_j}-\frac{(1-\lambda)^2b_j^3}{b_j^3-\eta_j^2L^2b_jB_j-\eta_j^3L^3B_j^2})\mathbb{E}\parallel\nabla f(\tilde{x}_j)\parallel^2\\
&=\frac{\gamma}{L}B_j^{1-\alpha+\alpha\beta-\beta}\left(2(1-\lambda)-(2\gamma B_j^{\alpha\beta-\alpha}+2B_j^{\beta-1})(1-\lambda)^2+\frac{(2\lambda-1)^2}{\frac{\gamma}{L}B_j^{2\alpha-2}}-1.16(1-\lambda)^2\right)\mathbb{E}\parallel\nabla f(\tilde{x}_j)\parallel^2\\
&\geq\frac{\gamma}{L}B_j^{\alpha\beta-\alpha-\beta+1}\left(2(1-\lambda)-(2\gamma B_j^{-1}+2B_j^{-1}-4)(1-\lambda)^2-1.16(1-\lambda)^2\right)\mathbb{E}\parallel\nabla f(\tilde{x}_j)\parallel^2. \qquad (49)
\end{aligned}$$

Eq.49 is positive when $0\leq\gamma\leq 2.42B_j-1$ and $B_j\geq 1$. Moreover, [21, 19] determined the learning rate $\eta=\frac{\gamma}{L}\frac{b_j}{B_j}\leq\frac{1}{3L}$ that $\gamma\leq\frac{1}{3}$ which can guarantees the convergence in non-convex case. In our case, γ should satisfy the range $0\leq\gamma\leq\frac{1}{3}\leq 2.42B_j-1$, thus $\gamma\leq\frac{1}{3}$.

Then Eq.47 can be simplified by Eq.49 as

$$\begin{aligned}
\mathbb{E}\parallel\nabla f(\tilde{x}_j)\parallel^2&\leq\frac{2\mathbb{E}[f(\tilde{x}_{j-1})-f(\tilde{x}_j)]+2\frac{\gamma}{L}B_j^{\alpha\beta-\alpha-\beta+1}(1+B_j^{\alpha\beta-\alpha}\gamma+B_j^{b-a}L)\mathbb{E}\parallel e_j\parallel^2}{\frac{\gamma}{L}B_j^{1-\alpha+\alpha\beta-\beta}\left(2(1-\lambda)-(2\gamma B_j^{\alpha\beta-\alpha}+2B_j^{\beta-1}-4LB_j^{2\alpha-2})(1-\lambda)^2-1.16(1-\lambda)^2\right)}\\
&\leq\frac{\overbrace{2\mathbb{E}[f(\tilde{x}_{j-1})-f(\tilde{x}_j)]}^{\text{positive by Lemma 2}}+\overbrace{2\frac{\gamma}{L}B_j^{\alpha\beta-\alpha-\beta+1}B_j^{2a}\mathbb{E}\parallel e_j\parallel^2}^{\text{positive}}}{\frac{\gamma}{L}B_j^{1-\alpha+\alpha\beta-\beta}\left(2(1-\lambda)-(2\gamma B_j^{\alpha\beta-\alpha}+2B_j^{\beta-1}-4LB_j^{2\alpha-2})(1-\lambda)^2-1.16(1-\lambda)^2\right)}. \qquad (50)
\end{aligned}$$

Then, using Lemma 9, Inq. 50 can be expressed as

$$\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \leq \frac{2\mathbb{E}[f(\tilde{x}_{j-1}) - f(\tilde{x}_j)] + 2(1-\lambda)^2 \frac{\gamma}{L} B_j^{\alpha\beta+\alpha-\beta} I(B_j < n)\mathcal{S}^*}{\frac{\gamma}{L} B_j^{1-\alpha+\alpha\beta-\beta} \left(2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1} - 4LB_j^{2\alpha-2})(1-\lambda)^2 - 1.16(1-\lambda)^2\right)}$$
$$= \frac{(\frac{2L}{\gamma})(\frac{b_j}{B_j})^{1-\alpha}\mathbb{E}(f(\tilde{x}_{j-1}) - f(\tilde{x}_j)) + 2(1-\lambda)^2 \frac{I(B_j<n))}{B_j^{1-2\alpha}}\mathcal{S}^*}{2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1} - 4LB_j^{2\alpha-2})(1-\lambda)^2 - 1.16(1-\lambda)^2} \tag{51}$$

Proof of Theorem 5

Suppose $\gamma \leq \frac{1}{3}$. Let $B_j = \min\left\{\frac{\mathcal{S}^*}{\epsilon}, \frac{n\mathcal{S}^*}{\mathcal{S}^* + 0.14 \cdot n^{\frac{1}{2}}\sigma\rho^{2j}}\right\}$, under Definition 3 and Theorem2 and 4, the output $\tilde{x}_T^*$ in Alg 4 satisfies one of two bounds.

1. If $B_j = \frac{\mathcal{S}^*}{\epsilon}$,$b_j = B_j^{\frac{1}{4}}$, $\eta_j = \frac{\gamma}{L}$, $\lambda^* = \frac{1}{2}$, $\theta \approx 0.51$ with an unbiased estimator,

$$\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 \leq \frac{\frac{4L}{\gamma}\triangle_f}{\sum_{j=1}^T B_j^{\frac{3}{4}}} + \frac{0.24(I(B_j < n)\mathcal{S}^*}{B_j},$$

.

2. If $B_j = \frac{n\mathcal{S}^*}{\mathcal{S}^* + 0.14 \cdot n^{\frac{1}{2}}\sigma\rho^{2j}}$, $b_j = 1$, $\eta_j = \frac{\gamma}{L}(\frac{1}{B_j})^{\frac{1}{2}}$, $\lambda^* = \frac{5}{8}$, $\Theta \approx 0.59$ with a biased estimator,

$$\mathbb{E}\|\nabla f(\tilde{x}_T^*)\|^2 < \frac{\frac{3.4L}{\gamma}\triangle_f}{\sum_{j=1}^T B_j^{\frac{1}{2}}} + 0.48\mathcal{S}^*.$$

Proof. Since $\tilde{x}_T^*$ is a random element from $(\tilde{x}_j)_{j=1}^T$ with

$$P(\tilde{x}_T^* = \tilde{x}_j) \propto \frac{\eta_j B_j}{b_j} \propto (\frac{B_j}{b_j})^\alpha, \tag{52}$$

Inq. 31 and 51 will be re-scaled as Inq. 53 and 54 respectively.

– For the unbiased estimator (Alg. 2), the upper bound is shown as,

$$\mathbb{E} \parallel \nabla f(\tilde{x}_T^*) \parallel^2 \leq \frac{(\frac{2L}{\gamma})\triangle_f}{\theta\sum_{j=1}^T b_j^{\alpha-1}B_j^{1-\alpha}} + \frac{2\lambda^4 I(B_j < n))\mathcal{S}^*}{\theta B_j^{1-2\alpha}}, \tag{53}$$

where $\theta = 2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1})(1-\lambda)^2 - 1.16\lambda^2$.

– For the biased estimator (Alg. 3), the upper bound is shown as,

$$\mathbb{E} \parallel \nabla f(\tilde{x}_j) \parallel^2 \leq \frac{(\frac{2L}{\gamma})\triangle_f}{\Theta \sum_{j=1}^{T} b_j^{\alpha-1} B_j^{1-\alpha}} + \frac{(1-\lambda)^2 I(B_j < n))\mathcal{S}^*}{\Theta B_j^{1-2\alpha}}, \tag{54}$$

where $\Theta = 2(1-\lambda) - (2\gamma B_j^{\alpha\beta-\alpha} + 2B_j^{\beta-1} - 4LB_j^{2\alpha-2})(1-\lambda)^2 - 1.16(1-\lambda)^2$.

After achieved the result in above, and specified parameters, we can obtain result of Theorem 5.

References

1. Agarwal, A., Bottou, L.: A lower bound for the optimization of finite sums. In: Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015. pp. 78–86. JMLR Workshop and Conference Proceedings, France (2015), http://leon.bottou.org/papers/agarwal-bottou-2015
2. Agarwal, N., Allen-Zhu, Z., Bullins, B., Hazan, E., Ma, T.: Finding approximate local minima faster than gradient descent. In: STOC 2017 - Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing. pp. 1195–1199. Association for Computing Machinery (Jun 2017). https://doi.org/10.1145/3055399.3055464
3. Allen-Zhu, Z., Hazan, E.: Variance reduction for faster non-convex optimization. In: Balcan, M., Weinberger, K. (eds.) 33rd International Conference on Machine Learning, ICML 2016. pp. 1093–1101. International Machine Learning Society (IMLS) (Jan 2016)
4. Allen-Zhu, Z.: Natasha 2: Faster non-convex optimization than sgd. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems. vol. 31, pp. 2675–2686. Curran Associates, Inc. (2018), https://proceedings.neurips.cc/paper/2018/file/79a49b3e3762632813f9e35f4ba53d6c-Paper.pdf
5. Arjevani, Y., Carmon, Y., Duchi, J.C., Foster, D.J., Srebro, N., Woodworth, B.E.: Lower bounds for non-convex stochastic optimization. CoRR **abs/1912.02365** (2019), http://arxiv.org/abs/1912.02365
6. Babanezhad, R., Ahmed, M.O., Virani, A., Schmidt, M.W., Konecný, J., Sallinen, S.: Stop wasting my gradients: Practical SVRG. CoRR **abs/1511.01942** (2015), http://arxiv.org/abs/1511.01942
7. Bertsekas, D.: A new class of incremental gradient methods for least squares problems. SIAM Journal on Optimization **7**(4), 913–926 (1997). https://doi.org/10.1137/S1052623495287022, https://doi.org/10.1137/S1052623495287022
8. Bi, J., Gunn, S.R.: A stochastic gradient method with biased estimation for faster nonconvex optimization. In: Nayak, A.C., Sharma, A. (eds.) PRICAI 2019: Trends in Artificial Intelligence. pp. 337–349. Springer International Publishing, Cham (2019)
9. Chen, H., Gao, A.: Robustness analysis for stochastic approximation algorithms. Stochastics and Stochastic Reports **26**(1), 3–20 (1989). https://doi.org/10.1080/17442508908833545, https://doi.org/10.1080/17442508908833545
10. Chen, H., Guo, L., Gao, A.: Convergence and robustness of the robbins-monro algorithm truncated at randomly varying bounds. Stochastic Processes and their Applications **27**, 217 – 231 (1987). https://doi.org/https://doi.org/10.1016/0304-4149(87)90039-1, http://www.sciencedirect.com/science/article/pii/0304414987900391

11. Chen, J., Luss, R.: Stochastic gradient descent with biased but consistent gradient estimators. CoRR **abs/1807.11880** (2018), http://arxiv.org/abs/1807.11880
12. Chen, J., Ma, T., Xiao, C.: Fastgcn: Fast learning with graph convolutional networks via importance sampling. CoRR **abs/1801.10247** (2018), http://arxiv.org/abs/1801.10247
13. Defazio, A., Bach, F.R., Lacoste-Julien, S.: SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. CoRR **abs/1407.0202** (2014), http://arxiv.org/abs/1407.0202
14. Fang, C., Li, C.J., Lin, Z., Zhang, T.: Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 31, pp. 689–699. Curran Associates, Inc. (2018)
15. Gaivoronski, A.A.: Convergence properties of backpropagation for neural nets via theory of stochastic gradient methods. part 1. Optimization Methods and Software **4**(2), 117–134 (1994). https://doi.org/10.1080/10556789408805582, https://doi.org/10.1080/10556789408805582
16. Ghadimi, S., Lan, G.: Accelerated gradient methods for nonconvex nonlinear and stochastic programming. Math. Program. **156**(1-2), 59–99 (2016). https://doi.org/10.1007/s10107-015-0871-8, https://doi.org/10.1007/s10107-015-0871-8
17. Johnson, R., Zhang, T.: Accelerating stochastic gradient descent using predictive variance reduction. In: Burges, C.J.C., Bottou, L., Welling, M., Ghahramani, Z., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 26, pp. 315–323. Curran Associates, Inc. (2013)
18. L. Lohr, S.: Sampling: Design and analysis. Technometrics **42** (05 2000). https://doi.org/10.2307/1271491
19. Lei, L., Jordan, M.: Less than a Single Pass: Stochastically Controlled Stochastic Gradient. In: Singh, A., Zhu, J. (eds.) Proceedings of the 20th International Conference on Artificial Intelligence and Statistics. Proceedings of Machine Learning Research, vol. 54, pp. 148–156. PMLR, Fort Lauderdale, FL, USA (20–22 Apr 2017), http://proceedings.mlr.press/v54/lei17a.html
20. Lei, L., Ju, C., Chen, J., Jordan, M.I.: Non-convex finite-sum optimization via scsg methods. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) Advances in Neural Information Processing Systems 30, pp. 2348–2358. Curran Associates, Inc. (2017), http://papers.nips.cc/paper/6829-non-convex-finite-sum-optimization-via-scsg-methods.pdf
21. Lei, L., Ju, C., Chen, J., Jordan, M.I.: Non-convex finite-sum optimization via SCSG methods. In: NIPS. pp. 2345–2355 (2017)
22. Liang, P., Bach, F.R., Bouchard, G., Jordan, M.I.: Asymptotically optimal regularization in smooth parametric models. In: Advances in Neural Information Processing Systems 22: 23rd Annual Conference on Neural Information Processing Systems 2009. Proceedings of a meeting held 7-10 December 2009, Vancouver, British Columbia, Canada. pp. 1132–1140 (2009), http://papers.nips.cc/paper/3693-asymptotically-optimal-regularization-in-smooth-parametric-models
23. McCullagh, P., Nelder, J.A.: Generalized Linear Models. Chapman & Hall / CRC, London (1989)
24. Nemirovski, A., Juditsky, A.B., Lan, G., Shapiro, A.: Robust stochastic approximation approach to stochastic programming. SIAM Journal on Optimization **19**(4), 1574–1609 (2009), http://dblp.uni-trier.de/db/journals/siamjo/siamjo19.html#NemirovskiJLS09
25. Nesterov, Y.: Introductory lectures on convex optimization: A basic course. In: Comput. Program. (01 2003)
26. Nesterov, Y.: Introductory Lectures on Convex Optimization: A Basic Course. Springer Publishing Company, Incorporated, 1 edn. (2014)

27. Niezgoda, M.: Laguerre–samuelson type inequalities. Linear Algebra and its Applications **422**(2), 574 – 581 (2007). https://doi.org/https://doi.org/10.1016/j.laa.2006.11.016, http://www.sciencedirect.com/science/article/pii/S0024379506005180
28. Qu, C., Li, Y., Xu, H.: Non-convex conditional gradient sliding. In: Dy, J., Krause, A. (eds.) Proceedings of the 35th International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 80, pp. 4208–4217. PMLR, Stockholmsmässan, Stockholm Sweden (10–15 Jul 2018), http://proceedings.mlr.press/v80/qu18a.html
29. Reddi, S.J., Sra, S., Póczos, B., Smola, A.: Fast incremental method for smooth nonconvex optimization. In: 2016 IEEE 55th Conference on Decision and Control (CDC). pp. 1971–1977 (Dec 2016). https://doi.org/10.1109/CDC.2016.7798553
30. Reddi, S.J., Hefny, A., Sra, S., Poczos, B., Smola, A.: Stochastic variance reduction for nonconvex optimization. In: Balcan, M.F., Weinberger, K.Q. (eds.) Proceedings of The 33rd International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 48, pp. 314–323. PMLR, New York, New York, USA (20–22 Jun 2016), http://proceedings.mlr.press/v48/reddi16.html
31. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV) **115**(3), 211–252 (2015). https://doi.org/10.1007/s11263-015-0816-y
32. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR **abs/1409.1556** (2014), http://arxiv.org/abs/1409.1556
33. Strongin, R.G., Sergeyev, Y.D.: Global Optimization with Non-Convex Constraints - Sequential and Parallel Algorithms (Nonconvex Optimization and Its Applications Volume 45) (Nonconvex Optimization and Its Applications). Springer-Verlag, Berlin, Heidelberg (2000)
34. Tseng, P.: An incremental gradient(-projection) method with momentum term and adaptive stepsize rule. SIAM Journal on Optimization **8**(2), 506–531 (1998). https://doi.org/10.1137/S1052623495294797, https://doi.org/10.1137/S1052623495294797
35. Zhou, D., Xu, P., Gu, Q.: Stochastic nested variance reduced gradient descent for nonconvex optimization. In: Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., Garnett, R. (eds.) Advances in Neural Information Processing Systems 31, pp. 3921–3932. Curran Associates, Inc. (2018)