Spider: Near-Optimal Non-Convex Optimization via Stochastic Path Integrated Differential Estimator

Cong Fang [*†] Chris Junchi Li [‡] Zhouchen Lin [*] Tong Zhang [‡]

July 4, 2018 (Initial)
October 17, 2018 (Current)

Abstract

In this paper, we propose a new technique named *Stochastic Path-Integrated Differential EstimatoR* (Spider), which can be used to track many deterministic quantities of interest with significantly reduced computational cost. We apply Spider to two tasks, namely the stochastic first-order and zeroth-order methods. For stochastic first-order method, combining Spider with normalized gradient descent, we propose two new algorithms, namely Spider-SFO and Spider-SFO$^+$, that solve non-convex stochastic optimization problems using stochastic gradients only. We provide sharp error-bound results on their convergence rates. In special, we prove that the Spider-SFO and Spider-SFO$^+$ algorithms achieve a record-breaking gradient computation cost of $\mathcal{O}\left(\min(n^{1/2}\epsilon^{-2}, \epsilon^{-3})\right)$ for finding an ϵ-approximate first-order and $\tilde{\mathcal{O}}\left(\min(n^{1/2}\epsilon^{-2} + \epsilon^{-2.5}, \epsilon^{-3})\right)$ for finding an $(\epsilon, \mathcal{O}(\epsilon^{0.5}))$-approximate second-order stationary point, respectively. In addition, we prove that Spider-SFO nearly matches the algorithmic lower bound for finding approximate first-order stationary points under the gradient Lipschitz assumption in the finite-sum setting. For stochastic zeroth-order method, we prove a cost of $\mathcal{O}(d\min(n^{1/2}\epsilon^{-2}, \epsilon^{-3}))$ which outperforms all existing results.

Contents

[*]Peking University; email: fangcong@pku.edu.cn; zlin@pku.edu.cn
[†]This work was done while Cong Fang was a Research Intern with Tencent AI Lab.
[‡]Tencent AI Lab; email: junchi.li.duke@gmail.com; tongzhang@tongzhang-ml.org

1 Introduction

In this paper, we study the optimization problem

$$\underset{\mathbf{x}\in\mathbb{R}^d}{\text{minimize}} \qquad f(\mathbf{x}) \equiv \mathbb{E}\left[F(\mathbf{x};\boldsymbol{\zeta})\right] \tag{1.1}$$

where the stochastic component $F(\mathbf{x};\boldsymbol{\zeta})$, indexed by some random vector $\boldsymbol{\zeta}$, is smooth and possibly *non-convex*. Non-convex optimization problem of form (1.1) contains many large-scale statistical learning tasks. Optimization methods that solve (1.1) are gaining tremendous popularity due to their favorable computational and statistical efficiencies (Bottou, 2010; Bubeck et al., 2015; Bottou et al., 2018). Typical examples of form (1.1) include principal component analysis, estimation of graphical models, as well as training deep neural networks (Goodfellow et al., 2016). The expectation-minimization structure of stochastic optimization problem (1.1) allows us to perform iterative updates and minimize the objective using its stochastic gradient $\nabla F(\mathbf{x};\boldsymbol{\zeta})$ as an estimator of its deterministic counterpart.

A special case of central interest is when the stochastic vector $\boldsymbol{\zeta}$ is finitely sampled. In such *finite-sum* (or *offline*) case, we denote each component function as $f_i(x)$ and (1.1) can be restated as

$$\underset{\mathbf{x}\in\mathbb{R}^d}{\text{minimize}} \qquad f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}) \tag{1.2}$$

where n is the number of individual functions. Another case is when n is reasonably large or even infinite, running across of the whole dataset is exhaustive or impossible. We refer it as the *online* (or *streaming*) case. For simplicity of notations we will study the optimization problem of form (1.2) in both finite-sum and on-line cases till the rest of this paper.

One important task for non-convex optimization is to search for, given the precision accuracy $\epsilon > 0$, an *ϵ-approximate first-order stationary point* $\mathbf{x} \in \mathbb{R}^d$ or $\|\nabla f(\mathbf{x})\| \leq \epsilon$. In this paper, we aim to propose a new technique, called the *Stochastic Path-Integrated Differential EstimatoR* (SPIDER), which enables us to construct an estimator that tracks a deterministic quantity with significantly lower sampling costs. As the readers will see, the SPIDER technique further allows us to design an algorithm with a faster rate of convergence for non-convex problem (1.2), in which we utilize the idea of *Normalized Gradient Descent* (NGD) (Nesterov, 2004; Hazan et al., 2015). NGD is a variant of Gradient Descent (GD) where the stepsize is picked to be inverse-proportional to the norm of the full gradient. Compared to GD, NGD exemplifies faster convergence, especially in the neighborhood of stationary points (Levy, 2016). However, NGD has been less popular due to its requirement of accessing the full gradient and its norm at each update. In this paper, we estimate and track the gradient and its norm via the SPIDER technique and then hybrid it with NGD. Measured by *gradient cost* which is the total number of computation of stochastic gradients, our proposed SPIDER-SFO algorithm achieves a faster rate of convergence in $\mathcal{O}(\min(n^{1/2}\epsilon^{-2}, \epsilon^{-3}))$ which outperforms the previous best-known results in both finite-sum (Allen-Zhu & Hazan, 2016)(Reddi et al., 2016) and on-line cases (Lei et al., 2017) by a factor of $\mathcal{O}(\min(n^{1/6}, \epsilon^{-0.333}))$.

For the task of finding stationary points for which we already achieved a faster convergence rate via our proposed SPIDER-SFO algorithm, a follow-up question to ask is: *is our proposed* SPIDER-*SFO algorithm optimal for an appropriate class of smooth functions?* In this paper, we provide an *affirmative* answer to this question in the finite-sum case. To be specific, inspired by a counterexample proposed by Carmon et al. (2017b) we are able to prove that the gradient cost upper bound of SPIDER-SFO algorithm matches the *algorithmic lower bound*. To put it differently, the gradient cost of SPIDER-SFO *cannot* be further improved for finding stationary points for some particular non-convex functions.

Nevertheless, it has been shown that for machine learning methods such as deep learning, approximate stationary points that have at least one negative Hessian direction, including saddle points and local maximizers, are often *not* sufficient and need to be avoided or escaped from (Dauphin et al., 2014; Ge et al., 2015). Specifically, under the smoothness condition for $f(\mathbf{x})$ and an additional Hessian-Lipschitz condition for $\nabla^2 f(\mathbf{x})$, we aim to find an *$(\epsilon, O(\epsilon^{0.5}))$-approximate second-order stationary point* which is a point $\mathbf{x} \in \mathbb{R}^d$ satisfying $\|\nabla f(\mathbf{x})\| \leq \epsilon$ and $\lambda_{\min}(\nabla^2 f(\mathbf{x})) \geq -\mathcal{O}(\epsilon^{0.5})$ (Nesterov & Polyak, 2006). As a side result, we propose a variant of our SPIDER-SFO algorithm, named SPIDER-SFO$^+$ (Algorithm 2) for finding an approximate second-order stationary point, based a so-called *Negative-Curvature-Search* method. Under an additional Hessian-Lipschitz assumption, SPIDER-SFO$^+$ achieves an $(\epsilon, \mathcal{O}(\epsilon^{0.5}))$-approximate second-order stationary point at a gradient cost of $\tilde{\mathcal{O}}(\min(n^{1/2}\epsilon^{-2} + \epsilon^{-2.5}, \epsilon^{-3}))$. In the on-line case, this indicates that our SPIDER-SFO algorithm improves upon the best-known gradient cost in the on-line case by a factor of $\tilde{\mathcal{O}}(\epsilon^{-0.25})$ (Allen-Zhu & Li, 2018). For the finite-sum case, the gradient cost of SPIDER is sharper than that of the state-of-the-art NEON+FastCubic/CDHS algorithm in Agarwal et al. (2017); Carmon et al. (2016) by a factor of $\tilde{\mathcal{O}}(n^{1/4}\epsilon^{0.25})$ when $n \geq \epsilon^{-1}$.[1]

1.1 Related Works

In the recent years, there has been a surge of literatures in machine learning community that analyze the convergence property of non-convex optimization algorithms. Limited by space and our knowledge, we have listed all literatures that we believe are mostly related to this work. We refer the readers to the monograph by Jain et al. (2017) and the references therein on recent general and model-specific convergence rate results on non-convex optimization.

First- and Zeroth-Order Optimization and Variance Reduction For the general problem of finding approximate stationary points, under the smoothness condition of $f(\mathbf{x})$, it is known that vanilla Gradient Descent (GD) and Stochastic Gradient Descent (SGD), which can be traced back to Cauchy (1847) and Robbins & Monro (1951) and achieve an ϵ-approximate stationary point with a gradient cost of $\mathcal{O}(\min(n\epsilon^{-2}, \epsilon^{-4}))$ (Nesterov, 2004; Ghadimi & Lan, 2013; Nesterov & Spokoiny, 2011; Ghadimi & Lan, 2013; Shamir, 2017).

Recently, the convergence rate of GD and SGD have been improved by the variance-reduction type of algorithms (Johnson & Zhang, 2013; Schmidt et al., 2017). In special, the finite-sum Stochastic Variance-Reduced Gradient (SVRG) and on-line Stochastically Controlled Stochastic Gradient (SCSG), to the gradient cost of $\tilde{\mathcal{O}}(\min(n^{2/3}\epsilon^{-2}, \epsilon^{-3.333}))$ (Allen-Zhu & Hazan, 2016; Reddi et al., 2016; Lei et al., 2017).

First-order method for finding approximate stationary points Recently, many literature study the problem of how to avoid or escape saddle points and achieve an approximate second-order stationary point at a polynomial gradient cost (Ge et al., 2015; Jin et al., 2017a; Xu et al., 2017; Allen-Zhu & Li, 2018; Hazan et al., 2015; Levy, 2016; Allen-Zhu, 2018; Reddi et al., 2018; Tripuraneni et al., 2018; Jin et al., 2017b; Lee et al., 2016; Agarwal et al., 2017; Carmon et al., 2016; Paquette et al., 2018). Among them, the group of authors Ge et al. (2015); Jin et al. (2017a) proposed the noise-perturbed variants of Gradient Descent (PGD) and Stochastic Gradient Descent (SGD) that escape from all saddle points and achieve an ϵ-approximate second-order stationary point in gradient cost of $\tilde{\mathcal{O}}(\min(n\epsilon^{-2}, poly(d)\epsilon^{-4}))$ stochastic gradients. Levy (2016) proposed the noise-perturbed variant of NGD which yields faster evasion of saddle points than GD.

The breakthrough of gradient cost for finding second-order stationary points were achieved in 2016/2017, when the two recent lines of literatures, namely FastCubic (Agarwal et al., 2017) and CDHS (Carmon et al., 2016) as well as their stochastic versions (Allen-Zhu, 2018; Tripuraneni et al., 2018), achieve a gradient cost of $\tilde{\mathcal{O}}(\min(n\epsilon^{-1.5} + n^{3/4}\epsilon^{-1.75}, \epsilon^{-3.5}))$ which serve as the best-known gradient cost for finding an $(\epsilon, \mathcal{O}(\epsilon^{0.5}))$-approximate second-order stationary point before the

[1] In the finite-sum case, when $n \leq \epsilon^{-1}$ Spider-SFO has a slower rate of $\tilde{\mathcal{O}}(\epsilon^{-2.5})$ than the state-of-art $\tilde{\mathcal{O}}(n^{3/4}\epsilon^{-1.75})$ rate achieved by Neon+FastCubic/CDHS (Allen-Zhu & Li, 2018). Neon+FastCubic/CDHS has exploited appropriate acceleration techniques, which has *not* been considered for Spider.

initial submission of this paper.[2] [3] In particular, Agarwal et al. (2017); Tripuraneni et al. (2018) converted the cubic regularization method for finding second-order stationary points (Nesterov & Polyak, 2006) to stochastic-gradient based and stochastic-Hessian-vector-product-based methods, and Carmon et al. (2016); Allen-Zhu (2018) used a Negative-Curvature Search method to avoid saddle points. See also recent works by Reddi et al. (2018) for related saddle-point-escaping methods that achieve similar rates for finding an approximate second-order stationary point.

Online PCA and the NEON method In late 2017, two groups Xu et al. (2017); Allen-Zhu & Li (2018) proposed a generic saddle-point-escaping method called Neon, a Negative-Curvature-Search method using stochastic gradients. Using such Neon method, one can convert a series of optimization algorithms whose update rules use stochastic gradients and Hessian-vector products (GD, SVRG, FastCubic/CDHS, SGD, SCSG, Natasha2, etc.) to the ones using *only* stochastic gradients without increasing the gradient cost. The idea of Neon was built upon Oja's iteration for principal component estimation (Oja, 1982), and its global convergence rate was proved to be near-optimal (Li et al., 2017; Jain et al., 2016). Allen-Zhu & Li (2017) later extended such analysis to the rank-k case as well as the gap-free case, the latter of which serves as the pillar of the Neon method.

Other concurrent works As the current work is carried out in its final phase, the authors became aware that an idea of resemblance was earlier presented in an algorithm named the *StochAstic Recursive grAdient algoritHm* (SARAH) (Nguyen et al., 2017a,b). Both our Spider-type of algorithms and theirs adopt the recursive stochastic gradient update framework. Nevertheless, our techniques essentially differ from the works Nguyen et al. (2017a,b) in two aspects:

(i) The version of SARAH proposed by Nguyen et al. (2017a,b) can be seen as a variant of gradient descent, while ours hybrids the Spider technique with a stochastic version of NGD.

(ii) Nguyen et al. (2017a,b) adopt a large stepsize setting (in fact their goal was to design a memory-saving variant of SAGA (Defazio et al., 2014)), while our algorithms adopt a small stepsize that is proportional to ϵ;

Soon after the initial submission to NIPS and arXiv release of this paper, we became aware that similar convergence rate results for stochastic first-order method were also achieved independently by the so-called SNVRG algorithm (Zhou et al., 2018b,a).[4]

[2] Allen-Zhu (2018) also obtains a gradient cost of $\tilde{\mathcal{O}}(\epsilon^{-3.25})$ to achieve a (modified and weakened) $(\epsilon, \mathcal{O}(\epsilon^{0.25}))$-approximate second-order stationary point.

[3] Here and in many places afterwards, the gradient cost also includes the number of stochastic Hessian-vector product accesses, which has similar running time with computing per-access stochastic gradient.

[4] To our best knowledge, the work by Zhou et al. (2018b,a) appeared on-line on June 20, 2018 and June 22, 2018, separately. SNVRG (Zhou et al., 2018b) obtains a gradient complexity of $\tilde{\mathcal{O}}(\min(n^{1/2}\epsilon^{-2}, \epsilon^{-3}))$ for finding an approximate first-order stationary point, and achieves $\tilde{\mathcal{O}}(\epsilon^{-3})$ gradient complexity for finding an approximate second-order stationary point (Zhou et al., 2018a) for a wide range of δ. By exploiting the third-order smoothness condition, SNVRG can also achieve an $(\epsilon, \mathcal{O}(\epsilon^{0.5}))$-approximate second-order stationary point in $\tilde{\mathcal{O}}(\epsilon^{-3})$ gradient costs.

1.2 Our Contributions

In this work, we propose the Stochastic Path-Integrated Differential Estimator (Spider) technique, which significantly avoids excessive access of stochastic oracles and reduces the time complexity. Such technique can be potential applied in many stochastic estimation problems.

(i) As a first application of our Spider technique, we propose the Spider-SFO algorithm (Algorithm 1) for finding an approximate first-order stationary point for non-convex stochastic optimization problem (1.2), and prove the optimality of such rate in at least one case. Inspired by recent works Johnson & Zhang (2013); Carmon et al. (2016, 2017b) and independent of Zhou et al. (2018b,a), this is the *first* time that the gradient cost of $\mathcal{O}(\min(n^{1/2}\epsilon^{-2}, \epsilon^{-3}))$ in both upper and lower (finite-sum only) bound for finding first-order stationary points for problem (1.2) were obtained.

(ii) Following Carmon et al. (2016); Allen-Zhu & Li (2018); Xu et al. (2017), we propose Spider-SFO^{+} algorithm (Algorithm 2) for finding an approximate second-order stationary point for non-convex stochastic optimization problem. To best of our knowledge, this is also the *first* time that the gradient cost of $\tilde{\mathcal{O}}(\min(n^{1/2}\epsilon^{-2} + \epsilon^{-2.5}, \epsilon^{-3}))$ achieved with standard assumptions.

(iii) As a second application of our Spider technique, we apply it to zeroth-order optimization for problem (1.2) and achieves individual function accesses of $\mathcal{O}(\min(dn^{1/2}\epsilon^{-2}, d\epsilon^{-3}))$. To best of our knowledge, this is also the *first* time that using Variance Reduction technique (Schmidt et al., 2017; Johnson & Zhang, 2013) to reduce the individual function accesses for non-convex problems to the aforementioned complexity.

(iv) We propose a much simpler analysis for proving convergence to a stationary point. One can flexibly apply our proof techniques to analyze others algorithms, e.g. SGD, SVRG (Johnson & Zhang, 2013), and SAGA (Defazio et al., 2014).

Organization. The rest of this paper is organized as follows. §2 presents the core idea of stochastic path-integrated differential estimator that can track certain quantities with much reduced computational costs. §3 provides the Spider method for stochastic first-order methods and convergence rate theorems of this paper for finding approximate first-order stationary and second-order stationary points, and details a comparison with concurrent works. §4 provides the Spider method for stochastic zeroth-order methods and relevant convergence rate theorems. §5 concludes the paper with future directions. All the detailed proofs are deferred to the appendix in their order of appearance.

Notation. Throughout this paper, we treat the parameters L, Δ, σ, and ρ, to be specified later as global constants. Let $\|\cdot\|$ denote the Euclidean norm of a vector or spectral norm of a square matrix. Denote $p_n = \mathcal{O}(q_n)$ for a sequence of vectors p_n and positive scalars q_n if there is a global constant C such that $|p_n| \le Cq_n$, and $p_n = \tilde{\mathcal{O}}(q_n)$ such C hides a poly-logarithmic factor of the parameters. Denote $p_n = \Omega(q_n)$ if there is a global constant C such that $|p_n| \ge Cq_n$. Let $\lambda_{\min}(\mathbf{A})$

denote the least eigenvalue of a real symmetric matrix $\mathbf{A}$. For fixed $K \ge k \ge 0$, let $\mathbf{x}_{k:K}$ denote the sequence $\{\mathbf{x}^k, \dots, \mathbf{x}^K\}$. Let $[n] = \{1, \dots, n\}$ and S denote the cardinality of a multi-set $\mathcal{S} \subset [n]$ of samples (a generic set that allows elements of multiple instances). For simplicity, we further denote the averaged sub-sampled stochastic estimator $\mathcal{B}_S := (1/S) \sum_{i \in \mathcal{S}} \mathcal{B}_i$ and averaged sub-sampled gradient $\nabla f_S := (1/S) \sum_{i \in \mathcal{S}} \nabla f_i$. Other notations are explained at their first appearance.

2 Stochastic Path-Integrated Differential Estimator: Core Idea

In this section, we present in detail the underlying idea of our Stochastic Path-Integrated Differential Estimator (Spider) technique behind the algorithm design. As the readers will see, such technique significantly avoids excessive access of the stochastic oracle and reduces the complexity, which is of independent interest and has potential applications in many stochastic estimation problems.

Let us consider an arbitrary deterministic vector quantity $Q(\mathbf{x})$. Assume that we observe a sequence $\hat{\mathbf{x}}_{0:K}$, and we want to dynamically track $Q(\hat{\mathbf{x}}^k)$ for $k = 0, 1, \dots, K$. Assume further that we have an initial estimate $\tilde{Q}(\hat{\mathbf{x}}^0) \approx Q(\hat{\mathbf{x}}^0)$, and an unbiased estimate $\boldsymbol{\xi}_k(\hat{\mathbf{x}}_{0:k})$ of $Q(\hat{\mathbf{x}}^k) - Q(\hat{\mathbf{x}}^{k-1})$ such that for each $k = 1, \dots, K$

$$\mathbb{E}\left[\boldsymbol{\xi}_k(\hat{\mathbf{x}}_{0:k}) \mid \hat{\mathbf{x}}_{0:k}\right] = Q(\hat{\mathbf{x}}^k) - Q(\hat{\mathbf{x}}^{k-1}).$$

Then we can integrate (in the discrete sense) the stochastic differential estimate as

$$\tilde{Q}(\hat{\mathbf{x}}_{0:K}) := \tilde{Q}(\hat{\mathbf{x}}^0) + \sum_{k=1}^{K} \boldsymbol{\xi}_k(\hat{\mathbf{x}}_{0:k}). \tag{2.1}$$

We call estimator $\tilde{Q}(\hat{\mathbf{x}}_{0:K})$ the *Stochastic Path-Integrated Differential EstimatoR*, or Spider for brevity. We conclude the following proposition which bounds the error of our estimator $\|\tilde{Q}(\hat{\mathbf{x}}_{0:K}) - Q(\hat{\mathbf{x}}^K)\|$, in terms of both expectation and high probability:

Proposition 1. *We have*

(i) The martingale variance bound has

$$\mathbb{E}\|\tilde{Q}(\hat{\mathbf{x}}_{0:K}) - Q(\hat{\mathbf{x}}^K)\|^2 = \mathbb{E}\|\tilde{Q}(\hat{\mathbf{x}}^0) - Q(\hat{\mathbf{x}}^0)\|^2 + \sum_{k=1}^{K} \mathbb{E}\|\boldsymbol{\xi}_k(\hat{\mathbf{x}}_{0:k}) - (Q(\hat{\mathbf{x}}^k) - Q(\hat{\mathbf{x}}^{k-1}))\|^2. \tag{2.2}$$

(ii) Suppose

$$\|\tilde{Q}(\hat{\mathbf{x}}^0) - Q(\hat{\mathbf{x}}^0)\| \le b_0 \tag{2.3}$$

and for each $k = 1, \dots, K$

$$\|\boldsymbol{\xi}_k(\hat{\mathbf{x}}_{0:k}) - (Q(\hat{\mathbf{x}}^k) - Q(\hat{\mathbf{x}}^{k-1}))\| \le b_k, \tag{2.4}$$

Then for any $\gamma > 0$ and a given $k \in \{1, \ldots, K\}$ we have with probability at least $1 - 4\gamma$

$$\left\| \tilde{Q}(\hat{\mathbf{x}}_{0:k}) - Q(\hat{\mathbf{x}}^k) \right\| \leq 2\sqrt{\sum_{s=0}^{k} b_s^2 \cdot \log \frac{1}{\gamma}}. \tag{2.5}$$

Proposition 1(i) can be easily concluded using the property of square-integrable martingales. To prove the high-probability bound in Proposition 1(ii), we need to apply an Azuma-Hoeffding-type concentration inequality (Pinelis, 1994). See §A in the Appendix for more details.

Now, let $\mathcal{B}$ map any $\mathbf{x} \in \mathbb{R}^d$ to a random estimate $\mathcal{B}_i(\mathbf{x})$ such that, conditioning on the observed sequence $\mathbf{x}_{0:k}$, we have for each $k = 1, \ldots, K$,

$$\mathbb{E}\left[\mathcal{B}_i(\mathbf{x}^k) - \mathcal{B}_i(\mathbf{x}^{k-1}) \mid \mathbf{x}_{0:k}\right] = \mathcal{V}^k - \mathcal{V}^{k-1}. \tag{2.6}$$

At each step k let S_* be a subset that samples $\mathcal{S}_*$ elements in $[n]$ with replacement, and let the stochastic estimator $\mathcal{B}_{S_*} = (1/\mathcal{S}_*) \sum_{i \in S_*} \mathcal{B}_i$ satisfy

$$\mathbb{E}\|\mathcal{B}_i(\mathbf{x}) - \mathcal{B}_i(\mathbf{y})\|^2 \leq L_{\mathcal{B}}^2 \|\mathbf{x} - \mathbf{y}\|^2, \tag{2.7}$$

and $\|\mathbf{x}^k - \mathbf{x}^{k-1}\| \leq \epsilon_1$ for all $k = 1, \ldots, K$. Finally, we set our estimator $\mathcal{V}^k$ of $\mathcal{B}(\mathbf{x}^k)$ as

$$\mathcal{V}^k = \mathcal{B}_{S_*}(\mathbf{x}^k) - \mathcal{B}_{S_*}(\mathbf{x}^{k-1}) + \mathcal{V}^{k-1}.$$

Applying Proposition 1 immediately concludes the following lemma, which gives an error bound of the estimator $\mathcal{V}^k$ in terms of the second moment of $\|\mathcal{V}^k - \mathcal{B}(\mathbf{x}^k)\|$:

Lemma 1. *We have under the condition (2.7) that for all $k = 1, \ldots, K$,*

$$\mathbb{E}\|\mathcal{V}^k - \mathcal{B}(\mathbf{x}^k)\|^2 \leq \frac{k L_{\mathcal{B}}^2 \epsilon_1^2}{\mathcal{S}_*} + \mathbb{E}\|\mathcal{V}^0 - \mathcal{B}(\mathbf{x}^0)\|^2. \tag{2.8}$$

It turns out that one can use Spider to track many quantities of interest, such as stochastic gradient, function values, zero-order estimate gradient, functionals of Hessian matrices, etc. Our proposed Spider-based algorithms in this paper take $\mathcal{B}_i$ as the stochastic gradient ∇f_i and the zeroth-order estimate gradient, separately.

3 SPIDER for Stochastic First-Order Method

In this section, we apply Spider to the task of finding both first-order and second-order stationary points for non-convex stochastic optimization. The main advantage of Spider-SFO lies in using SPIDER to estimate the gradient with a low computation cots. We introduce the basic settings and assumptions in §3.1 and propose the main error-bound theorems for finding approximate first-order and second-order stationary points, separately in §3.2 and §3.3.

3.1 Settings and Assumptions

We first introduce the formal definition of approximate first-order and second-order stationary points, as follows.

Definition 1. *We call $\mathbf{x} \in \mathbb{R}^d$ an ϵ-approximate first-order stationary point, or simply an FSP, if*

$$\|\nabla f(\mathbf{x})\| \le \epsilon. \tag{3.1}$$

Also, call $\mathbf{x}$ an (ϵ, δ)-approximate second-order stationary point, or simply an SSP, if

$$\|\nabla f(\mathbf{x})\| \le \epsilon, \qquad \lambda_{\min}\left(\nabla^2 f(\mathbf{x})\right) \ge -\delta. \tag{3.2}$$

The definition of an (ϵ, δ)-approximate second-order stationary point generalizes the classical version where $\delta = \sqrt{\rho\epsilon}$, see e.g. Nesterov & Polyak (2006). For our purpose of analysis, we also pose the following additional assumption:

Assumption 1. *We assume the following*

(i) The $\Delta := f(\mathbf{x}^0) - f^* < \infty$ where $f^* = \inf_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x})$ is the global infimum value of $f(\mathbf{x})$;

(ii) The component function $f_i(\mathbf{x})$ has an averaged L-Lipschitz gradient, i.e. for all $\mathbf{x}, \mathbf{y}$,

$$\mathbb{E}\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\|^2 \le L^2\|\mathbf{x} - \mathbf{y}\|^2;$$

(iii) (For on-line case only) the stochastic gradient has a finite variance bounded by $\sigma^2 < \infty$, i.e.

$$\mathbb{E}\,\|\nabla f_i(\mathbf{x}) - \nabla f(\mathbf{x})\|^2 \le \sigma^2.$$

Alternatively, to obtain high-probability results using concentration inequalities, we propose the following more stringent assumptions:

Assumption 2. *We assume that Assumption 1 holds and, in addition,*

(ii') (Optional) each component function $f_i(\mathbf{x})$ has L-Lipschitz continuous gradient, i.e. for all $i, \mathbf{x}, \mathbf{y}$,

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| \le L\|\mathbf{x} - \mathbf{y}\|.$$

Note when f is twice continuously differentiable, Assumption 1 (ii) is equivalent to $\mathbb{E}\|\nabla^2 f_i(\mathbf{x})\|^2 \le L^2$ for all $\mathbf{x}$ and is weaker than the additional Assumption 2 (ii'), since the absolute norm squared bounds the variance for any random vector.

(iii') (For on-line case only) the gradient of each component function $f_i(\mathbf{x})$ has finite bounded variance by $\sigma^2 < \infty$ (with probability 1) , i.e. for all $i, \mathbf{x}$,

$$\|\nabla f_i(\mathbf{x}) - \nabla f(\mathbf{x})\|^2 \le \sigma^2.$$

Algorithm 1 SPIDER-SFO: Input $\mathbf{x}^0$, q, S_1, S_2, n_0, ϵ, and $\tilde{\epsilon}$ (For finding first-order stationary point)

for $k = 0$ to K **do**
  if $\mod(k, q) = 0$ **then**
    Draw S_1 samples (or compute the full gradient for the finite-sum case), let $\mathbf{v}^k = \nabla f_{\mathcal{S}_1}(\mathbf{x}^k)$
  else
    Draw S_2 samples, and let $\mathbf{v}^k = \nabla f_{\mathcal{S}_2}(\mathbf{x}^k) - \nabla f_{\mathcal{S}_2}(\mathbf{x}^{k-1}) + \mathbf{v}^{k-1}$
  end if

OPTION I ◇ for convergence rates in high probability
  if $\|\mathbf{v}^k\| \le 2\tilde{\epsilon}$ **then**
    return $\mathbf{x}^k$
  else
    $\mathbf{x}^{k+1} = \mathbf{x}^k - \eta \cdot (\mathbf{v}^k / \|\mathbf{v}^k\|)$ where $\eta = \frac{\epsilon}{Ln_0}$
  end if

OPTION II ◇ for convergence rates in expectation
  $\mathbf{x}^{k+1} = \mathbf{x}^k - \eta^k \mathbf{v}^k$ where $\eta^k = \min\left(\frac{\epsilon}{Ln_0\|\mathbf{v}^k\|}, \frac{1}{2Ln_0}\right)$

end for

OPTION I: Return $\mathbf{x}^K$ ◇ however, this line is *not* reached with high probability

OPTION II: Return $\tilde{\mathbf{x}}$ chosen uniformly at random from $\{\mathbf{x}^k\}_{k=0}^{K-1}$

Assumption 2 is common in applying concentration laws to obtain high probability result[5].

For the problem of finding an (ϵ, δ)-approximate second-order stationary point, we pose in addition to Assumption 1 the following assumption:

Assumption 3. *We assume that Assumption 2 (including (ii')) holds and, in addition, each component function $f_i(\mathbf{x})$ has ρ-Lipschitz continuous Hessian, i.e. for all $i, \mathbf{x}, \mathbf{y}$,*

$$\|\nabla^2 f_i(\mathbf{x}) - \nabla^2 f_i(\mathbf{y})\| \le \rho \|\mathbf{x} - \mathbf{y}\|.$$

We emphasize that Assumptions 1, 2, and 3 are standard for non-convex stochastic optimization (Agarwal et al., 2017; Carmon et al., 2017b; Jin et al., 2017a; Xu et al., 2017; Allen-Zhu & Li, 2018).

3.2 First-Order Stationary Point

[5]In this paper, we use Azuma-Hoeffding-type concentration inequality to obtain high probability results like Xu et al. (2017); Allen-Zhu & Li (2018). By applying Bernstein inequality, under the Assumption 1, the parameters in the Assumption 2 are allowed to be $\tilde{\Omega}(\epsilon^{-1})$ larger without hurting the convergence rate.

Recall that NGD has iteration update rule

$$\mathbf{x}^{k+1} = \mathbf{x}^k - \eta \frac{\nabla f(\mathbf{x}^k)}{\|\nabla f(\mathbf{x}^k)\|}, \tag{3.3}$$

where η is a constant step size. The NGD update rule (3.3) ensures $\|\mathbf{x}^{k+1} - \mathbf{x}^k\|$ being constantly equal to the stepsize η, and might fastly escape from saddle points and converge to a second-order stationary point (Levy, 2016). We propose Spider-SFO in Algorithm 1, which is like a stochastic variant of NGD with the Spider technique applied, so as to maintain an estimator in each epoch $\nabla f(\mathbf{x}^k)$ at a higher accuracy under limited gradient budgets.

To analyze the convergence rate of Spider-SFO, let us first consider the on-line case for Algorithm 1. We let the input parameters be

$$S_1 = \frac{2\sigma^2}{\epsilon^2}, \qquad S_2 = \frac{2\sigma}{\epsilon n_0}, \qquad \eta = \frac{\epsilon}{L n_0}, \qquad \eta^k = \min\left(\frac{\epsilon}{L n_0 \|\mathbf{v}^k\|}, \frac{1}{2 L n_0}\right), \qquad q = \frac{\sigma n_0}{\epsilon}, \tag{3.4}$$

where $n_0 \in [1, 2\sigma/\epsilon]$ is a free parameter to choose.[6] In this case, $\mathbf{v}^k$ in Line 5 of Algorithm 1 is a Spider for $\nabla f(\mathbf{x}^k)$. To see this, recall $\nabla f_i(\mathbf{x}^{k-1})$ is the stochastic gradient drawn at step k and

$$\mathbb{E}\left[\nabla f_i(\mathbf{x}^k) - \nabla f_i(\mathbf{x}^{k-1}) \mid \mathbf{x}_{0:k}\right] = \nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^{k-1}). \tag{3.5}$$

Plugging in $\mathcal{V}^k = \mathbf{v}^k$ and $\mathcal{B}_i = \nabla f_i$ in Lemma 1 of §2, we can use $\mathbf{v}^k$ in Algorithm 1 as the Spider and conclude the following lemma that is pivotal to our analysis.

Lemma 2. *Set the parameters S_1, S_2, η, and q as in (3.4), and $k_0 = \lfloor k/q \rfloor \cdot q$. Then under the Assumption 1, we have*

$$\mathbb{E}\left[\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \mid \mathbf{x}_{0:k_0}\right] \le \epsilon^2.$$

Here we compute the conditional expectation over the randomness of $x_{(k_0+1):k}$.

Lemma 2 shows that our Spider $\mathbf{v}^k$ of $\nabla f(\mathbf{x})$ maintains an error of $\mathcal{O}(\epsilon)$. Using this lemma, we are ready to present the following results for Stochastic First-Order (SFO) method for finding first-order stationary points of (1.2).

Upper Bound for Finding First-Order Stationary Points, in Expectation

Theorem 1 (First-Order Stationary Point, on-line setting, expectation)**.** *For the on-line case, set the parameters S_1, S_2, η, and q as in (3.4), and $K = \lfloor (4L\Delta n_0)\epsilon^{-2} \rfloor + 1$. Then under the Assumption 1, for Algorithm 1 with OPTION I, after K iteration, we have*

$$\mathbb{E}\left[\|\nabla f(\tilde{\mathbf{x}})\|\right] \le 5\epsilon. \tag{3.6}$$

The gradient cost is bounded by $16L\Delta\sigma \cdot \epsilon^{-3} + 2\sigma^2\epsilon^{-2} + 4\sigma n_0^{-1}\epsilon^{-1}$ for any choice of $n_0 \in [1, 2\sigma/\epsilon]$. Treating Δ, L and σ as positive constants, the stochastic gradient complexity is $\mathcal{O}(\epsilon^{-3})$.

[6] When $n_0 = 1$, the mini-batch size is $2\sigma/\epsilon$, which is the largest mini-batch size that Algorithm 1 allows to choose.

The relatively reduced minibatch size serves as the key ingredient for the superior performance of SPIDER-SFO. For illustrations, let us compare the sampling efficiency among SGD, SCSG and SPIDER-SFO in their special cases. With some involved analysis of these algorithms, we can conclude that to ensure a sufficient function value decrease of $\Omega(\epsilon^2/L)$ at each iteration,

(i) for SGD the choice of mini-batch size is $\mathcal{O}\big(\sigma^2 \cdot \epsilon^{-2}\big)$;

(ii) for SCSG (Lei et al., 2017) and Natasha2 (Allen-Zhu, 2018) the mini-batch size is $\mathcal{O}\big(\sigma \cdot \epsilon^{-1.333}\big)$;

(iii) for our SPIDER-SFO only needs a reduced mini-batch size of $\mathcal{O}\big(\sigma \cdot \epsilon^{-1}\big)$

Turning to the finite-sum case, analogous to the on-line case we let

$$S_2 = \frac{n^{1/2}}{n_0}, \qquad \eta = \frac{\epsilon}{Ln_0}, \qquad \eta^k = \min\left(\frac{\epsilon}{Ln_0\|\mathbf{v}^k\|}, \frac{1}{2Ln_0}\right), \qquad q = n_0 n^{1/2}, \tag{3.7}$$

where $n_0 \in [1, n^{1/2}]$. In this case, one computes the full gradient $\mathbf{v}^k = \nabla f_{S_1}(\mathbf{x}^k)$ in Line 3 of Algorithm 1. We conclude our second upper-bound result:

Theorem 2 (First-Order Stationary Point, finite-sum setting)**.** *In the finite-sum case, set the parameters $\mathcal{S}_2$, η, and q as in (3.7), $K = \lfloor (4L\Delta n_0)\epsilon^{-2} \rfloor + 1$ and let $S_1 = [n]$, i.e. we obtain the full gradient in Line 3. The gradient cost is bounded by $n + 8(L\Delta) \cdot n^{1/2}\epsilon^{-2} + 2n_0^{-1}n^{1/2}$ for any choice of $n_0 \in [1, n^{1/2}]$. Treating Δ, L and σ as positive constants, the stochastic gradient complexity is $\mathcal{O}(n + n^{1/2}\epsilon^{-2})$.*

Lower Bound for Finding First-Order Stationary Points To conclude the optimality of our algorithm we need an algorithmic lower bound result (Carmon et al., 2017b; Woodworth & Srebro, 2016). Consider the finite-sum case and any random algorithm $\mathcal{A}$ that maps functions $f : \mathbb{R}^d \to \mathbb{R}$ to a sequence of iterates in $\mathbb{R}^{d+1}$, with

$$[\mathbf{x}^k; i_k] = \mathcal{A}^{k-1}\big(\boldsymbol{\xi}, \nabla f_{i_0}(\mathbf{x}^0), \nabla f_{i_1}(\mathbf{x}^1), \ldots, \nabla f_{i_{k-1}}(\mathbf{x}^{k-1})\big), \quad k \geq 1, \tag{3.8}$$

where $\mathcal{A}^k$ are measure mapping into $\mathbb{R}^{d+1}$, i_k is the individual function chosen by $\mathcal{A}$ at iteration k, and $\boldsymbol{\xi}$ is uniform random vector from $[0, 1]$. And $[\mathbf{x}^0; i_0] = \mathcal{A}^0(\boldsymbol{\xi})$, where $\mathcal{A}^0$ is a measure mapping. The lower-bound result for solving (1.2) is stated as follows:

Theorem 3 (Lower bound for SFO for the finite-sum setting)**.** *For any $L > 0$, $\Delta > 0$, and $2 \leq n \leq O\left(\Delta^2 L^2 \cdot \epsilon^{-4}\right)$, for any algorithm $\mathcal{A}$ satisfying (3.8), there exists a dimension $d = \tilde{\mathcal{O}}\big(\Delta^2 L^2 \cdot n^2 \epsilon^{-4}\big)$, and a function f satisfies Assumption 1 in the finite-sum case, such that in order to find a point $\tilde{\mathbf{x}}$ for which $\|\nabla f(\tilde{\mathbf{x}})\| \leq \epsilon$, $\mathcal{A}$ must cost at least $\Omega\big(L\Delta \cdot n^{1/2}\epsilon^{-2}\big)$ stochastic gradient accesses.*

Note the condition $n \leq \mathcal{O}(\epsilon^{-4})$ in Theorem 3 ensures that our lower bound $\Omega(n^{1/2}\epsilon^{-2}) = \Omega(n + n^{1/2}\epsilon^{-2})$, and hence our upper bound in Theorem 1 matches the lower bound in Theorem 3 up to a constant factor of relevant parameters, and is hence *near-optimal*. Inspired by Carmon et al. (2017b), our proof of Theorem 3 utilizes a specific counterexample function that requires at least

$\Omega(n^{1/2}\epsilon^{-2})$ stochastic gradient accesses. Note Carmon et al. (2017b) analyzed such counterexample in the deterministic case $n = 1$ and we generalize such analysis to the finite-sum case $n \geq 1$.

Remark 1. *Note by setting $n = \mathcal{O}(\epsilon^{-4})$ the lower bound complexity in Theorem 3 can be as large as $\Omega(\epsilon^{-4})$. We emphasize that this does not violate the $\mathcal{O}(\epsilon^{-3})$ upper bound in the on-line case [Theorem 1], since the counterexample established in the lower bound depends not on the stochastic gradient variance σ^2 specified in Assumption 1(iii), but on the component number n. To obtain the lower bound result for the on-line case with the additional Assumption 1(iii), with more efforts one might be able to construct a second counterexample that requires $\Omega(\epsilon^{-3})$ stochastic gradient accesses with the knowledge of σ instead of n. We leave this as a future work.*

Upper Bound for Finding First-Order Stationary Points, in High-Probability We consider obtaining high-probability results. With Theorem 1 and Theorem 2 in hand, by Markov Inequality, we have $\|\nabla f(\tilde{\mathbf{x}})\| \leq 15\epsilon$ with probability $\frac{2}{3}$. Thus a straightforward way to obtain a high probability result is by adding an additional verification step in the end of Algorithm 1, in which we check whether $\tilde{\mathbf{x}}$ satisfies $\|\nabla f(\tilde{\mathbf{x}})\| \leq 15\epsilon$ (for the on-line case when $\nabla f(\tilde{\mathbf{x}})$ are unaccessible, under Assumption 2 (iii'), we can draw $\tilde{O}(\epsilon^{-2})$ samples to estimate $\|\nabla f(\tilde{\mathbf{x}})\|$ in high accuracy). If not, we can restart Algorithm 1 (at most in $O(\log(1/p))$ times) until it find a desired solution. However, because the above way needs running Algorithm 1 in multiple times, in the following, we show with Assumption 2 (including (2)), original Algorithm 1 obtains a solution with an additional polylogarithmic factor under high probability.

Theorem 4 (First-Order Stationary Point, on-line setting, high probability)**.** *For the on-line case, set the parameters S_1, S_2, η and q in (3.4). Set $\tilde{\epsilon} = 10\epsilon\log\left(\left(4\lfloor 4L\Delta n_0\epsilon^{-2}\rfloor + 12\right)p^{-1}\right) \sim \tilde{\mathcal{O}}(\epsilon)$. Then under the Assumption 2 (including (ii')), with probability at least $1-p$, Algorithm 1 terminates before $K_0 = \lfloor(4L\Delta n_0)\epsilon^{-2}\rfloor + 2$ iterations and outputs an $\mathbf{x}^{\mathcal{K}}$ satisfying*

$$\|\mathbf{v}^{\mathcal{K}}\| \leq 2\tilde{\epsilon} \quad \textit{and} \quad \|\nabla f(\mathbf{x}^{\mathcal{K}})\| \leq 3\tilde{\epsilon}. \tag{3.9}$$

The gradient costs to find a FSP satisfying (3.9) with probability $1-p$ are bounded by $16L\Delta\sigma\cdot\epsilon^{-3} + 2\sigma^2\epsilon^{-2} + 8\sigma n_0^{-1}\epsilon^{-1}$ for any choice of of $n_0 \in [1, 2\sigma/\epsilon]$. Treating Δ, L and σ as constants, the stochastic gradient complexity is $\tilde{\mathcal{O}}(\epsilon^{-3})$.

Theorem 5 (First-Order Stationary Point, finite-sum setting)**.** *In the finite-sum case, set the parameters S_1, S_2, η, and q as (3.7). let $S_1 = [n]$, i.e. we obtain the full gradient in Line 3. Then under the Assumption 2 (including (ii')), with probability at least $1-p$, Algorithm 1 terminates before $K_0 = \lfloor 4L\Delta n_0/\epsilon^2\rfloor + 2$ iterations and outputs an $\mathbf{x}^{\mathcal{K}}$ satisfying*

$$\|\mathbf{v}^{\mathcal{K}}\| \leq 2\tilde{\epsilon} \quad \textit{and} \quad \|\nabla f(\mathbf{x}^{\mathcal{K}})\| \leq 3\tilde{\epsilon}. \tag{3.10}$$

where $\tilde{\epsilon} = 16\epsilon\log\left(\left(4(L\Delta n_0\epsilon^{-2} + 12\right)p^{-1}\right) = \tilde{\mathcal{O}}(\epsilon)$. So the gradient costs to find a FSP satisfying (3.10) with probability $1-p$ are bounded by $n + 8L\Delta n^{1/2}\epsilon^{-2} + (2n_0^{-1})n^{1/2} + 4n_0^{-1}n^{1/2}$ with any choice of $n_0 \in [1, n^{1/2}]$. Treating Δ, L and σ as constants, the stochastic gradient complexity is $\tilde{\mathcal{O}}(n + n^{1/2}\epsilon^{-2})$.

3.3 Second-Order Stationary Point

To find a second-order stationary point with (3.1), we can fuse our Spider-SFO in Algorithm 1 with a Negative-Curvature-Search (NC-Search) iteration that solves the following task: given a point $\mathbf{x} \in \mathbb{R}^d$, decide if $\lambda_{\min}(\nabla^2 f(\mathbf{x})) \geq -\delta$ or find a unit vector $\mathbf{w}_1$ such that $\mathbf{w}_1^\top \nabla^2 f(\mathbf{x}) \mathbf{w}_1 \leq -\delta/2$ (for numerical reasons, one has to leave some room between the two bounds). For the on-line case, NC-Search can be efficiently solved by Oja's algorithm (Oja, 1982; Allen-Zhu, 2018) and also by Neon (Allen-Zhu & Li, 2018; Xu et al., 2017) with the gradient cost of $\tilde{\mathcal{O}}(\delta^{-2})$.[7] When $\mathbf{w}_1$ is found, one can set $\mathbf{w}_2 = \pm(\delta/\rho)\mathbf{w}_1$ where $\pm$ is a random sign. Then under Assumption 3, Taylor's expansion implies that (Allen-Zhu & Li, 2018)

$$f(\mathbf{x} + \mathbf{w}_2) \leq f(\mathbf{x}) + [\nabla f(\mathbf{x})]^\top \mathbf{w}_2 + \frac{1}{2}\mathbf{w}_2^\top [\nabla^2 f(\mathbf{x})] \mathbf{w}_2 + \frac{\rho}{6}\|\mathbf{w}_2\|^3. \tag{3.11}$$

Taking expectation, one has $\mathbb{E} f(\mathbf{x} + \mathbf{w}_2) \leq f(\mathbf{x}) - \delta^3/(2\rho^2) + \delta^3/(6\rho^2) = f(\mathbf{x}) - \delta^3/(3\rho^2)$. This indicates that when we find a direction of negative curvature or Hessian, updating $\mathbf{x} \leftarrow \mathbf{x} + \mathbf{w}_2$ decreases the function value by $\Omega(\delta^3)$ in expectation. Our Spider-SFO algorithm fused with NC-Search is described in the following steps:

Step 1. Run an efficient NC-Search iteration to find an $\mathcal{O}(\delta)$-approximate negative Hessian direction $\mathbf{w}_1$ using stochastic gradients, e.g. Neon2 (Allen-Zhu & Li, 2018).

Step 2. If NC-Search find a $\mathbf{w}_1$, update $\mathbf{x} \leftarrow \mathbf{x} \pm (\delta/\rho)\mathbf{w}_1$ in $\delta/(\rho\eta)$ mini-steps, and simultaneously use Spider $\mathbf{v}^k$ to maintain an estimate of $\nabla f(\mathbf{x})$. Then Goto Step 1.

Step 3. If not, run Spider-SFO for $\delta/(\rho\eta)$ steps directly using the Spider $\mathbf{v}^k$ (without restart) in Step 2. Then Goto Step 1.

Step 4. During Step 3, if we find $\|\mathbf{v}^k\| \leq 2\tilde{\epsilon}$, return $\mathbf{x}^k$.

The formal pseudocode of the algorithm described above, which we refer to as Spider-SFO$^+$, is detailed in Algorithm 2[8]. The core reason that Spider-SFO$^+$ enjoys a highly competitive convergence rate is that, instead of performing a single large step δ/ρ at the approximate direction of negative curvature as in Neon2(Allen-Zhu & Li, 2018), we split such one large step into $\delta/(\rho\eta)$ small, equal-length mini-steps in Step 2, where each mini-step moves the iteration by an η distance. This allows the algorithm to successively maintain the Spider estimate of the current gradient in Step 3 and avoid re-computing the gradient in Step 1.

Our final result on the convergence rate of Algorithm 2 is stated as:

Theorem 6 (Second-Order Stationary Point)**.** *Let Assumptions 3 hold. For the on-line case, set q, S_1, S_2, η in (3.4), $\mathscr{K} = \frac{\delta L n_0}{\rho\epsilon}$ with any choice of $n_0 \in [1, 2\sigma/\epsilon]$, then with probability at*

[7]Recall that the NEgative-curvature-Originated-from-Noise method (or Neon method for short) proposed independently by Allen-Zhu & Li (2018); Xu et al. (2017) is a generic procedure that convert an algorithm that finds an approximate first-order stationary points to the one that finds an approximate second-order stationary point.

[8]In our initial version, Spider-SFO$^+$ first find a FSP and then run NC-search iteration to find a SSP, which also ensures competitive $\tilde{\mathcal{O}}(\epsilon^{-3})$ rate. Our newly Spider-SFO$^+$ are easier to fuse momentum technique when n is small. Please see the discussion later.

Algorithm 2 SPIDER-SFO$^+$: Input $\mathbf{x}^0$, S_1, S_2, n_0, q, η, $\mathscr{K}$, $k=0$, ϵ, $\tilde{\epsilon}$, (For finding a second-order stationary point)

for $j = 0$ **to** J **do**
  Run an efficient NC-search iteration, e.g. NEON2$(f, \mathbf{x}^k, 2\delta, \frac{1}{16J})$ and obtain $\mathbf{w}_1$
  if $\mathbf{w}_1 \neq \perp$ **then**
    $\diamond$ Second-Order Descent:
    Randomly flip a sign, and set $\mathbf{w}_2 = \pm\eta\mathbf{w}_1$ and $j = \delta/(\rho\eta) - 1$
    for k **to** $k + \mathscr{K}$ **do**
      if $\mathrm{mod}(k, q) = 0$ **then**
        Draw $\mathcal{S}_1$ samples, $\mathbf{v}^k = \nabla f_{S_1}(\mathbf{x}^k)$
      else
        Draw $\mathcal{S}_2$ samples, $\mathbf{v}^k = \nabla f_{S_2}(\mathbf{x}^k) - \nabla f_{S_2}(\mathbf{x}^{k-1}) + \mathbf{v}^{k-1}$
      end if
      $\mathbf{x}^{k+1} = \mathbf{x}^k - \mathbf{w}_2$
    end for
  else
    $\diamond$ First-Order Descent:
    for k **to** $k + \mathscr{K}$ **do**
      if $\mathrm{mod}(k, q) = 0$ **then**
        Draw $\mathcal{S}_1$ samples, $\mathbf{v}^k = \nabla f_{S_1}(\mathbf{x}^k)$
      else
        Draw $\mathcal{S}_2$ samples, $\mathbf{v}^k = \nabla f_{S_2}(\mathbf{x}^k) - \nabla f_{S_2}(\mathbf{x}^{k-1}) + \mathbf{v}^{k-1}$
      end if
      if $\|\mathbf{v}^k\| \leq 2\tilde{\epsilon}$ **then**
        return $\mathbf{x}^k$
      end if
      $\mathbf{x}^{k+1} = \mathbf{x}^k - \eta \cdot (\mathbf{v}^k / \|\mathbf{v}^k\|)$
    end for
  end if
end for

least $1/2$[9]*, Algorithm 2 outputs an* $\mathbf{x}^k$ *with* $j \leq J = 4\left\lfloor \max\left(\frac{3\rho^2\Delta}{\delta^3}, \frac{4\Delta\rho}{\delta\epsilon}\right)\right\rfloor + 4$*, and* $k \leq K_0 = \left(4\left\lfloor \max\left(\frac{3\rho^2\Delta}{\delta^3}, \frac{4\Delta\rho}{\delta\epsilon}\right)\right\rfloor + 4\right)\frac{Ln_0\delta}{\rho\epsilon}$ *satisfying*

$$\|\nabla f(\mathbf{x}^k)\| \leq \tilde{\epsilon} \quad \text{and} \quad \lambda_{\min}(\nabla^2 f(\mathbf{x}^k)) \geq -3\delta, \tag{3.12}$$

with $\tilde{\epsilon} = 10\epsilon \log\left(256\left(\left\lfloor \max\left(\frac{3\rho^2\Delta}{\delta^3}, \frac{4\Delta\rho}{\delta\epsilon}\right)\right\rfloor + 1\right)\frac{\delta L n_0}{\rho\epsilon} + 64\right) = \tilde{\mathcal{O}}(\epsilon)$*. The gradient cost to find a Second-Order Stationary Point with probability at least* $1/2$ *is upper bounded by*

$$\tilde{\mathcal{O}}\left(\frac{\Delta L\sigma}{\epsilon^3} + \frac{\Delta\sigma L\rho}{\epsilon^2\delta^2} + \frac{\Delta L^2\rho^2}{\delta^5} + \frac{\Delta L^2\rho}{\epsilon\delta^3} + \frac{\sigma^2}{\epsilon^2} + \frac{L^2}{\delta^2} + \frac{L\sigma\delta}{\rho\epsilon^2}\right).$$

[9] By multiple times (at most in $O(\log(1/p))$ times) of verification and restarting Algorithm 2 , one can also obtain a high-probability result.

Analogously for the finite-sum case, under the same setting of Theorem 2, set q, S_1, S_2, η in (3.7), $\mathscr{K} = \frac{\delta L n_0}{\rho\epsilon}$, $\tilde{\epsilon} = 16\epsilon\log\left(256\left(\left\lfloor\max\left(\frac{3\rho^2\Delta}{\delta^3}, \frac{4\Delta\rho}{\delta\epsilon}\right)\right\rfloor + 1\right)\frac{\delta L n_0}{\rho\epsilon} + 64\right) = \tilde{\mathcal{O}}(\epsilon)$, with probability $1/2$, Algorithm 2 outputs an $\mathbf{x}^k$ satisfying (3.12) in $j \le J$ and $k \le K_0$ with gradients cost of

$$\tilde{\mathcal{O}}\left(\frac{\Delta L n^{1/2}}{\epsilon^2} + \frac{\Delta\rho L n^{1/2}}{\epsilon\delta^2} + \frac{\Delta L^2\rho^2}{\delta^5} + \frac{\Delta L^2\rho}{\epsilon\delta^3} + n + \frac{L^2}{\delta^2} + \frac{L n^{1/2}\delta}{\rho\epsilon}\right).$$

Corollary 7. *Treating Δ, L, σ, and ρ as positive constants, with high probability the gradient cost for finding an (ϵ,δ)-approximate second-order stationary point is $\tilde{\mathcal{O}}(\epsilon^{-3} + \delta^{-2}\epsilon^{-2} + \delta^{-5})$ for the on-line case and $\tilde{\mathcal{O}}(n^{1/2}\epsilon^{-2} + n^{1/2}\delta^{-2}\epsilon^{-1} + \delta^{-3}\epsilon^{-1} + \delta^{-5} + n)$ for the finite-sum case, respectively. When $\delta = \mathcal{O}(\epsilon^{0.5})$, the gradient cost is $\mathcal{O}(\min(n^{1/2}\epsilon^{-2} + \epsilon^{-2.5}, \epsilon^{-3}))$.*

Notice that one may directly apply an on-line variant of the Neon method to the Spider-SFO Algorithm 1 which alternately does Second-Order Descent (but not maintaining Spider) and First-Order Descent (Running a new Spider-SFO). Simple analysis suggests that the Neon+ Spider-SFO algorithm achieves a gradient cost of $\tilde{\mathcal{O}}\big(\epsilon^{-3} + \epsilon^{-2}\delta^{-3} + \delta^{-5}\big)$ for the on-line case and $\tilde{\mathcal{O}}\big(n^{1/2}\epsilon^{-2} + n^{1/2}\epsilon^{-1}\delta^{-3} + \delta^{-5}\big)$ for the finite-sum case (Allen-Zhu & Li, 2018; Xu et al., 2017). We discuss the differences in detail.

- The dominate term in the gradient cost of Neon+ Spider-SFO is the so-called *coupling term* in the regime of interest: $\epsilon^{-2}\delta^{-3}$ for the on-line case and $n^{1/2}\epsilon^{-1}\delta^{-3}$ for the finite-sum case, separately. Due to this term, most convergence rate results in concurrent works for the on-line case such as Reddi et al. (2018); Tripuraneni et al. (2018); Xu et al. (2017); Allen-Zhu & Li (2018); Zhou et al. (2018a) have gradient costs that cannot break the $\mathcal{O}(\epsilon^{-3.5})$ barrier when δ is chosen to be $\mathcal{O}(\epsilon^{0.5})$. Observe that we always need to run a new Spider-SFO which at least costs $\mathcal{O}\big(\min(\epsilon^{-2}, n)\big)$ stochastic gradient accesses.

- Our analysis sharpens the seemingly non-improvable coupling term by modifying the single large Neon step to many mini-steps. Such modification enables us to maintain the Spider estimates and obtain a coupling term $\mathcal{O}\left(\min(n, \epsilon^{-2})\delta^{-2}\right)$ of Spider-SFO$^+$, which improves upon the Neon coupling term $\mathcal{O}\left(\min(n, \epsilon^{-2})\delta^{-3}\right)$ by a factor of δ.

- For the finite-sum case, Spider-SFO$^+$ enjoys a convergence rate that is faster than existing methods only in the regime $n = \Omega(\epsilon^{-1})$ [Table 1]. For the case of $n = \mathcal{O}(\epsilon^{-1})$, using Spider to track the gradient in the Neon procedure can be more costly than applying appropriate acceleration techniques (Agarwal et al., 2017; Carmon et al., 2016).[10] Beacause it is well-known that momentum technique (Nesterov, 1983) provably ensures faster convergence rates when n is sufficient small (Shalev-Shwartz & Zhang, 2016). One can also apply momentum technique to solve the sub-problem in Step 1 and 3 like Carmon et al. (2016); Allen-Zhu & Li (2018) when $n \le \mathcal{O}(\epsilon^{-1})$, and thus can achieve the state-of-the-art gradient cost of

$$\tilde{\mathcal{O}}\left(\min\left(n\epsilon^{-1.5} + n^{3/4}\epsilon^{-1.75}, n^{1/2}\epsilon^{-2} + n^{1/2}\epsilon^{-1}\delta^{-2}\right) + \min\left(n + n^{3/4}\delta^{-0.5}, \delta^{-2}\right)\delta^{-3}\right),$$

[10] Spider-SFO$^+$ enjoys a faster rate than Neon+Spider-SFO where computing the "full" gradient dominates the gradient cost, namely $\delta = \mathcal{O}(1)$ in the on-line case and $\delta = \mathcal{O}(n^{1/2}\epsilon)$ for the finite-sum case.

	Algorithm		Online	Finite-Sum
First-order Stationary Point	GD / SGD	(Nesterov, 2004)	ϵ^{-4}	$n\epsilon^{-2}$
	SVRG / SCSG	(Allen-Zhu & Hazan, 2016) (Reddi et al., 2016) (Lei et al., 2017)	$\epsilon^{-3.333}$	$n + n^{2/3}\epsilon^{-2}$
	Spider-SFO	(this work)	ϵ^{-3}	$n + n^{1/2}\epsilon^{-2}$ $^{\Delta}$
First-order Stationary Point (Hessian-Lipschitz Required)	Perturbed GD / SGD	(Ge et al., 2015) (Jin et al., 2017a)	$poly(d)\epsilon^{-4}$	$n\epsilon^{-2}$
	Neon+GD / Neon+SGD	(Xu et al., 2017) (Allen-Zhu & Li, 2018)	ϵ^{-4}	$n\epsilon^{-2}$
	AGD	(Jin et al., 2017b)	N/A	$n\epsilon^{-1.75}$
	Neon+SVRG / Neon+SCSG	(Allen-Zhu & Hazan, 2016) (Reddi et al., 2016) (Lei et al., 2017)	$\epsilon^{-3.5}$ $(\epsilon^{-3.333})$	$n\epsilon^{-1.5} + n^{2/3}\epsilon^{-2}$
	Neon+FastCubic/CDHS	(Agarwal et al., 2017) (Carmon et al., 2016) (Tripuraneni et al., 2018)	$\epsilon^{-3.5}$	$n\epsilon^{-1.5} + n^{3/4}\epsilon^{-1.75}$
	Neon+Natasha2	(Allen-Zhu, 2018) (Xu et al., 2017) (Allen-Zhu & Li, 2018)	$\epsilon^{-3.5}$ $(\epsilon^{-3.25})$	$n\epsilon^{-1.5} + n^{2/3}\epsilon^{-2}$
	Spider-SFO^{+}	(this work)	ϵ^{-3}	$n^{1/2}\epsilon^{-2}$ $^{\Theta}$

Table 1: Comparable results on the gradient cost for nonconvex optimization algorithms that use only individual (or stochastic) gradients. Note that the gradient cost hides a poly-logarithmic factors of d, n, ϵ. For clarity and brevity purposes, we record for most algorithms the gradient cost for finding an $(\epsilon, \mathcal{O}(\epsilon^{0.5}))$-approximate second-order stationary point. For some algorithms we added in a bracket underneath the best gradient cost for finding an $(\epsilon, \mathcal{O}(\epsilon^{\alpha}))$-approximate second-order stationary point among $\alpha \in (0, 1]$, for the fairness of comparison.

$^{\Delta}$: we provide lower bound for this gradient cost entry.

$^{\Theta}$: this entry is for $n \geq \Omega(\epsilon^{-1})$ only, in which case Spider-SFO^{+} outperforms Neon+FastCubic/CDHS.

in all scenarios.

3.4 Comparison with Concurrent Works

This subsection compares our Spider algorithms with concurrent works. In special, we detail our main result for applying Spider to first-order methods in the list below:

(i) For the problem of finding an ϵ-approximate first-order stationary point, under Assumption 1 our results indicate a gradient cost of $\mathcal{O}(\min(\epsilon^{-3}, n^{1/2}\epsilon^{-2}))$ which supersedes the best-known convergence rate results for stochastic optimization problem (1.2) [Theorems 1 and 2]. Before this work, the best-known result is $\mathcal{O}\left(\min(\epsilon^{-3.333}, n^{2/3}\epsilon^{-2})\right)$, achieved by Allen-Zhu & Hazan (2016); Reddi et al. (2016) in the finite-sum case and Lei et al. (2017) in the on-line case, separately. Moreover, such a gradient cost achieves the algorithmic lower bound for the finite-sum setting [Theorem 3].

(ii) For the problem of finding (ϵ, δ)-approximate second-order stationary point x, under both

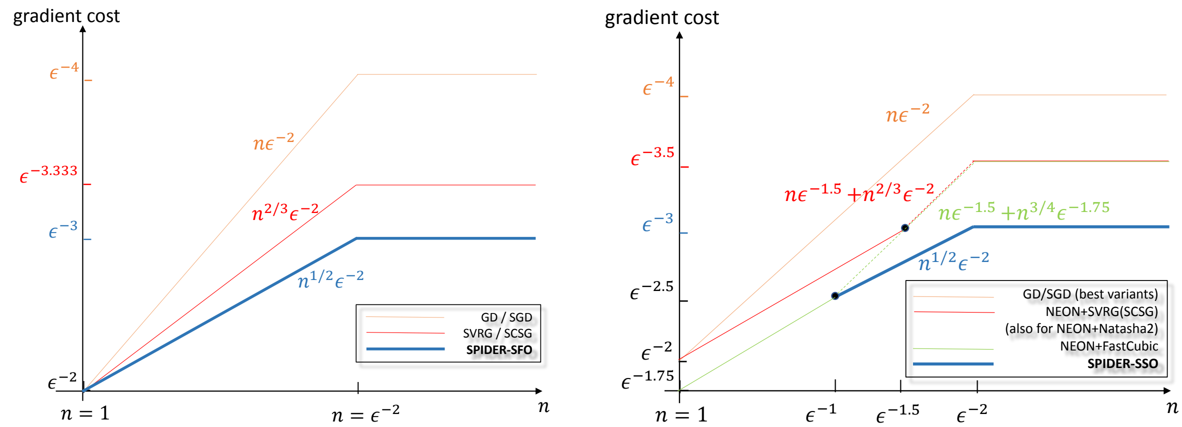

Figure 1: Left panel: gradient cost comparison for finding an ϵ-approximate first-order stationary point. Right panel: gradient cost comparison for finding an $(\epsilon, \mathcal{O}(\epsilon^{0.5}))$-approximate second-order stationary points (note we assume Hessian Lipschitz condition). Both axes are on the logarithmic scale of ϵ^{-1}.

> Assumptions 1 and 3, the gradient cost is $\tilde{\mathcal{O}}(\epsilon^{-3} + \epsilon^{-2}\delta^{-2} + \delta^{-5})$ in the on-line case and $\tilde{\mathcal{O}}(n^{1/2}\epsilon^{-2} + n^{1/2}\epsilon^{-1}\delta^{-2} + \epsilon^{-1}\delta^{-3} + \delta^{-5} + n)$ in the finite-sum case [Theorem 6]. In the classical definition of second-order stationary point where $\delta = \mathcal{O}(\epsilon^{0.5})$, such gradient cost is simply $\mathcal{O}(\epsilon^{-3})$ in the on-line case. In comparison, to the best of our knowledge the best-known results only achieve a gradient cost of $\mathcal{O}(\epsilon^{-3.5})$ under similar assumptions (Reddi et al., 2018; Tripuraneni et al., 2018; Allen-Zhu, 2018; Allen-Zhu & Li, 2018; Zhou et al., 2018a).

We summarize the comparison with concurrent works that solve (1.2) under similar assumptions in Table 1. In addition, we provide Figure 1 which draws the gradient cost against the magnitude of n for both an approximate stationary point.[11] For simplicity, we leave out the complexities of the algorithms that has Hessian-vector product access and only record algorithms that use stochastic gradients only.[12] Specifically, the yellow-boxed complexity $\mathcal{O}(n\epsilon^{-1.5} + n^{3/4}\epsilon^{-1.75})$ in Table 1, which was achieved by Neon+FastCubic/CDHS (Allen-Zhu & Li, 2018; Jin et al., 2017b) for finding an approximate second-order stationary point in the finite-sum case using momentum technique, are the only results that have *not* been outperformed by our Spider-SFO$^+$ algorithm in certain parameter regimes ($n \le \mathcal{O}(\epsilon^{-1})$ in this case).

4 SPIDER for Stochastic Zeroth-Order Method

For SZO algorithms, (2.3) can be solved only from the Incremental Zeroth-Order Oracle (IZO)(Nesterov & Spokoiny, 2011), which is defined as:

[11] One of the results not included in this table is Carmon et al. (2017a), which finds an ϵ-approximate first-order stationary point in $\mathcal{O}(n\epsilon^{-1.75})$ gradient evaluations. However, their result relies on a more stringent Hessian-Lipschitz condition, in which case a second-order stationary point can be found in similar gradient cost (Jin et al., 2017b).

[12] Due to the Neon method (Xu et al., 2017; Allen-Zhu & Li, 2018), nearly all existing Hessian-vector product based algorithms in stochastic optimization can be converted to ones that use stochastic gradients only.

Algorithm 3 SPIDER-SZO: Input $\mathbf{x}^0$, S_1, S_2, q, u, ϵ (For finding first-order stationary point)

for $k = 0$ to K **do**
  if mod $(k, q) = 0$ **then**
    Draw $S_1' = S_1/d$ training samples, for each dimension $j \in [d]$, compute ($\diamond$ with $2S_1$ total IZO costs)

$$v_j^k = \frac{1}{S_1'} \sum_{i \in \mathcal{S}_1'} \frac{f_i(\mathbf{x}^k + \mu \mathbf{e}_j) - f_i(\mathbf{x}^k)}{\mu}$$

    where $\mathbf{e}_j$ denotes the vector with j-th natural unit basis vector.
  else
    Draw S_2 sample pairs $(i, \mathbf{u})$, where $i \in [n]$ and $\mathbf{u} \sim N(\mathbf{0}, \mathbf{I}_d)$ with i and μ being independent.
    Update

$$\mathbf{v}^k = \frac{1}{S_2} \sum_{(i,\mathbf{u}) \in \mathcal{S}_2} \left(\frac{f_i(\mathbf{x}^k + \mu \mathbf{u}) - f_i(\mathbf{x}^k)}{\mu} \mathbf{u} - \frac{f_i(\mathbf{x}^{k-1} + \mu \mathbf{u}) - f_i(\mathbf{x}^{k-1})}{\mu} \mathbf{u} \right) + \mathbf{v}^{k-1}$$

  end if
  $\mathbf{x}^{k+1} = \mathbf{x}^k - \eta^k \mathbf{v}^k$ where $\eta^k = \min\left(\frac{\epsilon}{Ln_0\|\mathbf{v}^k\|}, \frac{1}{2Ln_0}\right)$ $\diamond$ for convergence rates in expectation
end for
Return $\tilde{\mathbf{x}}$ chosen uniformly at random from $\{\mathbf{x}^k\}_{k=0}^{K-1}$

Definition 2. *An IZO takes an index $i \in [n]$ and a point $\mathbf{x} \in \mathbb{R}^d$, and returns the $f_i(\mathbf{x})$.*

We use Assumption 2 (including (ii')) for convergence analysis which is standard for SZO(Nesterov & Spokoiny, 2011; Ghadimi & Lan, 2013) algorithms. Because the true gradient are not allowed to obtain for SZO. Most works (Nesterov & Spokoiny, 2011; Ghadimi & Lan, 2013; Shamir, 2017) use the gradient of a smoothed version of the objective function through a two-point feedback in a stochastic setting. Following (Nesterov & Spokoiny, 2011), we consider the typical Gaussian distribution in the convolution to smooth the function. Define

$$\hat{f}(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}}} \int f(\mathbf{x} + \mu \mathbf{u}) e^{-\frac{1}{2}\|\mathbf{u}\|^2} d\mathbf{u} = \mathbb{E}_{\mathbf{u}}[f(\mathbf{x} + \mu \mathbf{u})], \tag{4.1}$$

where $\mathbf{x} \in \mathbb{R}^d$. From (Nesterov & Spokoiny, 2011), the following properties holds :

(i) The gradient of $\hat{f}$ satisfies:

$$\nabla \hat{f}(\mathbf{x}) = \frac{1}{(2\pi)^{\frac{d}{2}}} \int \frac{f(\mathbf{x} + \mu \mathbf{u}) - f(\mathbf{x})}{\mu} \mathbf{u} e^{-\frac{1}{2}\|\mathbf{u}\|^2} d\mathbf{u}. \tag{4.2}$$

(ii) For any $\mathbf{x} \in \mathbb{R}^d$, $f(\mathbf{x})$ has Lipschitz continuous gradients, we have

$$\|\nabla \hat{f}(\mathbf{x}) - \nabla f(\mathbf{x})\| \leq \frac{\mu}{2} L (d+3)^{\frac{3}{2}}. \tag{4.3}$$

(iii) For any $\mathbf{x} \in \mathbb{R}^d$, $f(\mathbf{x})$ has Lipschitz continuous gradients, we have

$$\mathbb{E}_{\mathbf{u}}\left[\frac{1}{\mu^2}\left(f(\mathbf{x}+\mu\mathbf{u})-f(\mathbf{x})\right)^2\|\mathbf{u}\|^2\right] \leq \frac{\mu^2}{2}L^2(d+6)^3 + 2(d+4)\|\nabla f(\mathbf{x})\|^2. \tag{4.4}$$

From the (1), suppose $\mathbf{u} \sim N(\mathbf{0}, \mathbf{I}_d)$, and $i \in [n]$, with $\mathbf{u}$ and i being independent, we have

$$\begin{aligned} \mathbb{E}_{i,\mathbf{u}}\frac{f_i(\mathbf{x}^k+\mu\mathbf{u}_i)-f_i(\mathbf{x}^k)}{\mu}\mathbf{u} &= \frac{1}{(2\pi)^{\frac{d}{2}}}\mathbb{E}_i\left(\int \frac{f_i(\mathbf{x}+\mu\mathbf{u})-f_i(\mathbf{x})}{\mu}\mathbf{u}e^{-\frac{1}{2}\|\mathbf{u}\|^2}d\mathbf{u}\right) \\ = \frac{1}{(2\pi)^{\frac{d}{2}}}\left(\int \frac{f(\mathbf{x}+\mu\mathbf{u})-f(\mathbf{x})}{\mu}\mathbf{u}e^{-\frac{1}{2}\|\mathbf{u}\|^2}d\mathbf{u}\right) &= \nabla\hat{f}(\mathbf{x}^k). \end{aligned} \tag{4.5}$$

Also

$$\mathbb{E}_{i,\mathbf{u}}\left[\frac{f_i(\mathbf{x}^k+\mu\mathbf{u})-f_i(\mathbf{x}^k)}{\mu}\mathbf{u} - \left(\frac{f_i(\mathbf{x}^{k-1}+\mu\mathbf{u})-f_i(\mathbf{x}^{k-1})}{\mu}\mathbf{u}\right)\right] = \nabla\hat{f}(\mathbf{x}^k) - \nabla\hat{f}(\mathbf{x}^{k-1}). \tag{4.6}$$

For non-convex case, the best known result is $\mathcal{O}(d\epsilon^{-4})$ from Ghadimi & Lan (2013). We has not found a work that applying Variance Reduction technique to significantly reduce the complexity of IZO. This might because that even in finite-sum case, the full gradient is not available (with noise). In this paper, we give a stronger results by Spider technique, directly reducing the IZO from $\mathcal{O}(d\epsilon^{-4})$ to $\mathcal{O}(\min(dn^{1/2}\epsilon^{-2}, d\epsilon^{-3}))$.

From (4.6), we can integrate the two-point feed-back to track $\nabla\hat{f}(\mathbf{x})$. The algorithm is shown in Algorithm 3. Then the following lemma shows that $\mathbf{v}^k$ is a high accurate estimator of $\|\nabla\hat{f}(\mathbf{x}^k)\|$:

Lemma 3. *Under the Assumption 2, suppose i is random number of the function index, ($i \in [n]$) and $\mathbf{u}$ is a standard Gaussian random vector, i.e. $\mathbf{u} \sim N(\mathbf{0}, \mathbf{I}_d)$, we have*

$$\mathbb{E}_{i,\mathbf{u}}\left\|\left[\frac{f_i(\mathbf{x}+\mu\mathbf{u})-f_i(\mathbf{x})}{\mu}\mathbf{u} - \left(\frac{f_i(\mathbf{y}+\mu\mathbf{u})-f_i(\mathbf{y})}{\mu}\mathbf{u}\right)\right]\right\|^2 \leq 2(d+4)L^2\|\mathbf{x}-\mathbf{y}\|^2 + 2\mu^2(d+6)^3L^2 \tag{4.7}$$

From (4.3), by setting a smaller μ, the smoothed gradient $\nabla\hat{f}(\mathbf{x})$ approximates $\nabla f(\mathbf{x})$, which ensures sufficient function descent in each iteration. For simpleness, we only give expectation result, shown in Theorem 8.

Theorem 8. *Under the Assumption 2 (including (ii')). For infinite case, set $\mu = \min\left(\frac{\epsilon}{2\sqrt{6}L\sqrt{d}}, \frac{\epsilon}{\sqrt{6}n_0L(d+6)^{3/2}}\right)$, $S_1 = \frac{96d\sigma^2}{\epsilon^2}$, $S_2 = \frac{30(2d+9)\sigma}{\epsilon n_0}$, $q = \frac{5n_0\sigma}{\epsilon}$, where $n_0 \in [1, \frac{30(2d+9)\sigma}{\epsilon}]$. In the finite-sum case, set the parameters $S_2 = \frac{(2d+9)n^{1/2}}{n_0}$, and $q = \frac{n_0n^{1/2}}{6}$, let $S_1/d = [n]$, i.e. $v_j^k = f(\mathbf{x}^k+\mu\mathbf{e}_j) - f(\mathbf{x}^k)/\mu$ with $j \in [d]$, where $n_0 \in [1, \frac{n^{1/2}}{6}]$. Then with $\eta^k = \min(\frac{1}{2Ln_0}, \frac{\epsilon}{Ln_0\|\mathbf{v}^k\|})$, $K = \lfloor(4L\Delta n_0)\epsilon^{-2}\rfloor + 1$, for Algorithm 3 we have*

$$\mathbb{E}\left[\|\nabla f(\tilde{\mathbf{x}})\|\right] \leq 6\epsilon. \tag{4.8}$$

The IZO calls are $\mathcal{O}\left(d\min(n^{1/2}\epsilon^{-2}, \epsilon^{-3})\right)$.

5 Summary and Future Directions

We propose in this work the Spider method for non-convex optimization. Our Spider-type algorithms for first-order and zeroth-order optimization have update rules that are reasonably simple and achieve excellent convergence properties. However, there are still some important questions left. For example, the lower bound results for finding a second-order stationary point are *not* complete. Specially, it is *not* yet clear if our $\tilde{\mathcal{O}}(\epsilon^{-3})$ for the on-line case and $\tilde{\mathcal{O}}(n^{1/2}\epsilon^{-2})$ for the finite-sum case gradient cost upper bound for finding a second-order stationary point (when $n \geq \Omega(\epsilon^{-1})$) is *optimal* or the gradient cost can be further improved, assuming both Lipschitz gradient and Lipschitz Hessian. We leave this as a future research direction.

Acknowledgement The authors would like to thank NIPS Reviewer 1 to point out a mistake in the original proof of Theorem 1 and thank Zeyuan Allen-Zhu and Quanquan Gu for relevant discussions and pointing out references Zhou et al. (2018b,a), also Jianqiao Wangni for pointing out references Nguyen et al. (2017a,b), and Zebang Shen, Ruoyu Sun, Haishan Ye, Pan Zhou for very helpful discussions and comments. Zhouchen Lin is supported by National Basic Research Program of China (973 Program) (grant no. 2015CB352502), National Natural Science Foundation (NSF) of China (grant nos. 61625301 and 61731018), and Microsoft Research Asia.

References

Agarwal, N., Allen-Zhu, Z., Bullins, B., Hazan, E., & Ma, T. (2017). Finding approximate local minima faster than gradient descent. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing* (pp. 1195–1199).: ACM.

Allen-Zhu, Z. (2018). Natasha 2: Faster non-convex optimization than sgd. In *Advances in Neural Information Processing Systems*.

Allen-Zhu, Z. & Hazan, E. (2016). Variance reduction for faster non-convex optimization. In *International Conference on Machine Learning* (pp. 699–707).

Allen-Zhu, Z. & Li, Y. (2017). First effcient convergence for streaming k-PCA: a global, gap-free, and near-optimal rate. *The 58th Annual Symposium on Foundations of Computer Science*.

Allen-Zhu, Z. & Li, Y. (2018). Neon2: Finding local minima via first-order oracles. In *Advances in Neural Information Processing Systems*.

Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010* (pp. 177–186). Springer.

Bottou, L., Curtis, F. E., & Nocedal, J. (2018). Optimization methods for large-scale machine learning. *SIAM Review*, 60(2), 223–311.

Bubeck, S. et al. (2015). Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4), 231–357.

Carmon, Y., Duchi, J. C., Hinder, O., & Sidford, A. (2016). Accelerated methods for non-convex optimization. *To appear in SIAM Journal on Optimization, accepted*.

Carmon, Y., Duchi, J. C., Hinder, O., & Sidford, A. (2017a). "Convex Until Proven Guilty": Dimension-free acceleration of gradient descent on non-convex functions. In *International Conference on Machine Learning* (pp. 654–663).

Carmon, Y., Duchi, J. C., Hinder, O., & Sidford, A. (2017b). Lower bounds for finding stationary points i. *arXiv preprint arXiv:1710.11606*.

Cauchy, A. (1847). Méthode générale pour la résolution des systemes déquations simultanées. *Comptes Rendus de l'Academie des Science*, 25, 536–538.

Dauphin, Y. N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., & Bengio, Y. (2014). Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In *Advances in Neural Information Processing Systems* (pp. 2933–2941).

Defazio, A., Bach, F., & Lacoste-Julien, S. (2014). SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances in Neural Information Processing Systems* (pp. 1646–1654).

Durrett, R. (2010). *Probability: Theory and Examples (4th edition)*. Cambridge University Press.

Ge, R., Huang, F., Jin, C., & Yuan, Y. (2015). Escaping from saddle points – online stochastic gradient for tensor decomposition. In *Proceedings of The 28th Conference on Learning Theory* (pp. 797–842).

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

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Hazan, E., Levy, K., & Shalev-Shwartz, S. (2015). Beyond convexity: Stochastic quasi-convex optimization. In *Advances in Neural Information Processing Systems* (pp. 1594–1602).

Jain, P., Jin, C., Kakade, S. M., Netrapalli, P., & Sidford, A. (2016). Matching matrix Bernstein and near-optimal finite sample guarantees for Oja's algorithm. In *Proceedings of The 29th Conference on Learning Theory* (pp. 1147–1164).

Jain, P., Kar, P., et al. (2017). Non-convex optimization for machine learning. *Foundations and Trends® in Machine Learning*, 10(3-4), 142–336.

Jin, C., Ge, R., Netrapalli, P., Kakade, S. M., & Jordan, M. I. (2017a). How to escape saddle points efficiently. In *International Conference on Machine Learning* (pp. 1724–1732).

Jin, C., Netrapalli, P., & Jordan, M. I. (2017b). Accelerated gradient descent escapes saddle points faster than gradient descent. *arXiv preprint arXiv:1711.10456*.

Johnson, R. & Zhang, T. (2013). Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems* (pp. 315–323).

Kallenberg, O. & Sztencel, R. (1991). Some dimension-free features of vector-valued martingales. *Probability Theory and Related Fields*, 88(2), 215–247.

Lee, J. D., Simchowitz, M., Jordan, M. I., & Recht, B. (2016). Gradient descent only converges to minimizers. In *Proceedings of The 29th Conference on Learning Theory* (pp. 1246–1257).

Lei, L., Ju, C., Chen, J., & Jordan, M. I. (2017). Non-convex finite-sum optimization via scsg methods. In *Advances in Neural Information Processing Systems* (pp. 2345–2355).

Levy, K. Y. (2016). The power of normalization: Faster evasion of saddle points. *arXiv preprint arXiv:1611.04831*.

Li, C. J., Wang, M., Liu, H., & Zhang, T. (2017). Near-optimal stochastic approximation for online principal component estimation. *Mathematical Programming, Series B, Special Issue on Optimization Models and Algorithms for Data Science*.

Nesterov, Y. (1983). A method for unconstrained convex minimization problem with the rate of convergence o (1/kˆ 2). In *Doklady AN USSR*, volume 269 (pp. 543–547).

Nesterov, Y. (2004). *Introductory lectures on convex optimization: A basic course*, volume 87. Springer.

Nesterov, Y. & Polyak, B. T. (2006). Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1), 177–205.

Nesterov, Y. & Spokoiny, V. (2011). *Random gradient-free minimization of convex functions*. Technical report, Université catholique de Louvain, Center for Operations Research and Econometrics (CORE).

Nguyen, L. M., Liu, J., Scheinberg, K., & Takáč, M. (2017a). SARAH: A novel method for machine learning problems using stochastic recursive gradient. In D. Precup & Y. W. Teh (Eds.), *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research* (pp. 2613–2621). International Convention Centre, Sydney, Australia: PMLR.

Nguyen, L. M., Liu, J., Scheinberg, K., & Takáč, M. (2017b). Stochastic recursive gradient algorithm for nonconvex optimization. *arXiv preprint arXiv:1705.07261*.

Oja, E. (1982). Simplified neuron model as a principal component analyzer. *Journal of mathematical biology*, 15(3), 267–273.

Paquette, C., Lin, H., Drusvyatskiy, D., Mairal, J., & Harchaoui, Z. (2018). Catalyst for gradient-based nonconvex optimization. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics* (pp. 613–622).

Pinelis, I. (1994). Optimum bounds for the distributions of martingales in banach spaces. *The Annals of Probability*, (pp. 1679–1706).

Reddi, S., Zaheer, M., Sra, S., Poczos, B., Bach, F., Salakhutdinov, R., & Smola, A. (2018). A generic approach for escaping saddle points. In A. Storkey & F. Perez-Cruz (Eds.), *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research* (pp. 1233–1242). Playa Blanca, Lanzarote, Canary Islands: PMLR.

Reddi, S. J., Hefny, A., Sra, S., Poczos, B., & Smola, A. (2016). Stochastic variance reduction for nonconvex optimization. In *International conference on machine learning* (pp. 314–323).

Robbins, H. & Monro, S. (1951). A stochastic approximation method. *The annals of mathematical statistics*, (pp. 400–407).

Schmidt, M., Le Roux, N., & Bach, F. (2017). Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2), 83–112.

Shalev-Shwartz, S. & Zhang, T. (2016). Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, 155(1), 105–145.

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

Tripuraneni, N., Stern, M., Jin, C., Regier, J., & Jordan, M. I. (2018). Stochastic cubic regularization for fast nonconvex optimization. In *Advances in Neural Information Processing Systems*.

Vershynin, R. (2010). Introduction to the non-asymptotic analysis of random matrices. *arXiv preprint arXiv:1011.3027*.

Woodworth, B. & Srebro, N. (2017). Lower bound for randomized first order convex optimization. *arXiv preprint arXiv:1709.03594*.

Woodworth, B. E. & Srebro, N. (2016). Tight complexity bounds for optimizing composite objectives. In *Advances in Neural Information Processing Systems* (pp. 3639–3647).

Xu, Y., Jin, R., & Yang, T. (2017). First-order stochastic algorithms for escaping from saddle points in almost linear time. *arXiv preprint arXiv:1711.01944*.

Zhang, T. (2005). Learning bounds for kernel regression using effective data dimensionality. *Neural Computation*, 17(9), 2077–2098.

Zhou, D., Xu, P., & Gu, Q. (2018a). Finding local minima via stochastic nested variance reduction. *arXiv preprint arXiv:1806.08782*.

Zhou, D., Xu, P., & Gu, Q. (2018b). Stochastic nested variance reduction for nonconvex optimization. *arXiv preprint arXiv:1806.07811*.

A Vector-Martingale Concentration Inequality

In this and next section, we sometimes denote for brevity that $\mathbb{E}_k[\cdot] = \mathbb{E}[\cdot \mid x_{0:k}]$, the expectation operator conditional on $x_{0:k}$, for an arbitrary $k \geq 0$.

Concentration Inequality for Vector-valued Martingales We apply a result by Pinelis (1994) and conclude Proposition 2 which is an Azuma-Hoeffding-type concentration inequality. See also Kallenberg & Sztencel (1991), Lemma 4.4 in Zhang (2005) or Theorem 2.1 in Zhang (2005) and the references therein.

Proposition 2 (Theorem 3.5 in Pinelis (1994))**.** *Let $\boldsymbol{\epsilon}_{1:K} \in \mathbb{R}^d$ be a vector-valued martingale difference sequence with respect to $\boldsymbol{\mathcal{F}}_k$, i.e., for each $k = 1, \ldots, K$, $\mathbb{E}[\boldsymbol{\epsilon}_k \mid \boldsymbol{\mathcal{F}}_{k-1}] = 0$ and $\|\epsilon_k\|^2 \leq B_k^2$. We have*

$$\mathbb{P}\left(\left\|\sum_{k=1}^{K} \boldsymbol{\epsilon}_k\right\| \geq \lambda\right) \leq 4\exp\left(-\frac{\lambda^2}{4\sum_{k=1}^{K} B_k^2}\right), \tag{A.1}$$

where λ is an arbitrary real positive number.

Proposition 2 is not a straightforward derivation of one-dimensional Azuma's inequality. The key observation of Proposition 2 is that, the bound on the right hand of (A.1) is *dimension-free* (note the Euclidean norm version of $\mathbb{R}^d$ is $(2,1)$-smooth). Such dimension-free feature could be found as early as in Kallenberg & Sztencel (1991), uses the so-called *dimension reduction lemma* for Hilbert space which is inspired from its continuum version proved in Kallenberg & Sztencel (1991). Now, we are ready to prove Proposition 1.

A.1 Proof of Proposition 1

Proof of Proposition 1. It is straightforward to verify from the definition of $\tilde{Q}$ in (2.1) that

$$\tilde{Q}(\hat{\mathbf{x}}_{0:K}) - Q(\hat{\mathbf{x}}^K) = \tilde{Q}(\hat{\mathbf{x}}^0) - Q(\hat{\mathbf{x}}^0) + \sum_{k=1}^{K} \boldsymbol{\xi}_k(\hat{\mathbf{x}}_{0:k}) - (Q(\hat{\mathbf{x}}^k) - Q(\hat{\mathbf{x}}^{k-1}))$$

is a martingale, and hence (2.2) follows from the property of L^2 martingales (Durrett, 2010). □

A.2 Proof of Lemma 1

Proof of Lemma 1. For any $k > 0$, we have from Proposition 1 (by applying $\tilde{Q} = \mathcal{V}$)

$$\mathbb{E}_k\|\mathcal{V}^k - \mathcal{B}(\mathbf{x}^k)\|^2 = \mathbb{E}_k\|\mathcal{B}_{S_*}(\mathbf{x}^k) - \mathcal{B}(\mathbf{x}^k) - \mathcal{B}_{S_*}(\mathbf{x}^{k-1}) + \mathcal{B}(\mathbf{x}^{k-1})\|^2 + \|\mathcal{V}^{k-1} - \mathcal{B}(\mathbf{x}^{k-1})\|^2. \tag{A.2}$$

Then

$$
\begin{aligned}
& \mathbb{E}_k\|\mathcal{B}_{S_*}(\mathbf{x}^k)-\mathcal{B}(\mathbf{x}^k)-\mathcal{B}_{S_*}(\mathbf{x}^{k-1})+\mathcal{B}(\mathbf{x}^{k-1})\|^2 \\
\overset{a}{=}\ & \frac{1}{\mathcal{S}_*}\mathbb{E}\|\mathcal{B}_i(\mathbf{x}^k)-\mathcal{B}(\mathbf{x}^k)-\mathcal{B}_i(\mathbf{x}^{k-1})+\mathcal{B}(\mathbf{x}^{k-1})\|^2 \\
\overset{b}{\leq}\ & \frac{1}{\mathcal{S}_*}\mathbb{E}\|\mathcal{B}_i(\mathbf{x}^k)-\mathcal{B}_i(\mathbf{x}^{k-1})\|^2 \\
\overset{(2.7)}{\leq}\ & \frac{1}{\mathcal{S}_*}L_{\mathcal{B}}^2\mathbb{E}\|\mathbf{x}^k-\mathbf{x}^{k-1}\|^2 \leq \frac{L_{\mathcal{B}}^2\epsilon_1^2}{\mathcal{S}_*},
\end{aligned}
\tag{A.3}
$$

where in $\overset{a}{=}$ and $\overset{b}{\leq}$, we use Eq (2.6), and S_* are random sampled from $[n]$ with replacement. Combining (A.2) and (A.3), we have

$$
\mathbb{E}_k\|\mathcal{V}^k-\mathcal{B}(\mathbf{x}^k)\|^2 \leq \frac{L_{\mathcal{B}}^2\epsilon_1^2}{\mathcal{S}_*}+\|\mathcal{V}^{k-1}-\mathcal{B}(\mathbf{x}^{k-1})\|^2. \tag{A.4}
$$

Telescoping the above display for $k'=k-1,\dots,0$ and using the iterated law of expectation, we have

$$
\mathbb{E}\|\mathcal{V}^k-\mathcal{B}(\mathbf{x}^k)\|^2 \leq \frac{kL_{\mathcal{B}}^2\epsilon_1^2}{\mathcal{S}_*}+\mathbb{E}\|\mathcal{V}^0-\mathcal{B}(\mathbf{x}^0)\|^2. \tag{A.5}
$$

□

B Deferred Proofs

B.1 Proof of Lemma 2

Proof of Lemma 2. For $k=k_0$, we have

$$
\begin{aligned}
& \mathbb{E}_{k_0}\|\mathbf{v}^{k_0}-\nabla f(\mathbf{x}^{k_0})\|^2 \\
=\ & \mathbb{E}_{k_0}\|\nabla f_{S_1}(\mathbf{x}^{k_0})-\nabla f(\mathbf{x}^{k_0})\|^2 \leq \frac{\sigma^2}{S_1}=\frac{\epsilon^2}{2}.
\end{aligned}
\tag{B.1}
$$

From Line 14 of Algorithm 1 we have for all $k\geq 0$,

$$
\|\mathbf{x}^{k+1}-\mathbf{x}^k\| = \min\left(\frac{\epsilon}{Ln_0\|\mathbf{v}^k\|},\frac{1}{2Ln_0}\right)\|\mathbf{v}^k\| \leq \frac{\epsilon}{Ln_0}. \tag{B.2}
$$

Applying Lemma 1 with $\epsilon_1=\epsilon/(Ln_0)$, $S_2=2\sigma/(\epsilon n_0)$, $K=k-k_0\leq q=\sigma n_0/\epsilon$, we have

$$
\mathbb{E}_{k_0}\|\mathbf{v}^k-\nabla f(\mathbf{x}^k)\|^2 \leq \frac{\sigma n_0 L^2}{\epsilon}\cdot\frac{\epsilon^2}{L^2n_0^2}\cdot\frac{\epsilon n_0}{2\sigma}+\mathbb{E}_{k_0}\|\mathbf{v}^{k_0}-\nabla f(\mathbf{x}^{k_0})\|^2 \overset{(B.1)}{=} \epsilon^2, \tag{B.3}
$$

completing the proof. □

B.2 Proof of Expectation Results for FSP

The rest of this section devotes to the proofs of Theorems 1, 2. To prepare for them, we first conclude via standard analysis the following

Lemma 4. *Under the Assumption 1, setting $k_0 = \lfloor k/q \rfloor \cdot q$, we have*

$$\mathbb{E}_{k_0}\left[f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k)\right] \le -\frac{\epsilon}{4Ln_0}\mathbb{E}_{k_0}\left\|\mathbf{v}^k\right\| + \frac{3\epsilon^2}{4n_0L}. \tag{B.4}$$

Proof of Lemma 4. From Assumption 1 (ii), we have

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 = \|\mathbb{E}_i\left(\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\right)\|^2 \le \mathbb{E}_i\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\|^2 \le L^2\|\mathbf{x}-\mathbf{y}\|^2. \tag{B.5}$$

So $f(\mathbf{x})$ has L-Lipschitz continuous gradient, then

$$\begin{aligned} f(\mathbf{x}^{k+1}) &\le f(\mathbf{x}^k) + \left\langle \nabla f(\mathbf{x}^k), \mathbf{x}^{k+1} - \mathbf{x}^k \right\rangle + \frac{L}{2}\left\|\mathbf{x}^{k+1} - \mathbf{x}^k\right\|^2 \\ &= f(\mathbf{x}^k) - \eta^k\left\langle \nabla f(\mathbf{x}^k), \mathbf{v}^k \right\rangle + \frac{L(\eta^k)^2}{2}\left\|\mathbf{v}^k\right\|^2 \\ &= f(\mathbf{x}^k) - \eta^k\left(1 - \frac{\eta^k L}{2}\right)\left\|\mathbf{v}^k\right\|^2 - \eta^k\left\langle \nabla f(\mathbf{x}^k) - \mathbf{v}^k, \mathbf{v}^k \right\rangle \\ &\overset{a}{\le} f(\mathbf{x}^k) - \eta^k\left(\frac{1}{2} - \frac{\eta^k L}{2}\right)\left\|\mathbf{v}^k\right\|^2 + \frac{\eta^k}{2}\left\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\right\|^2, \end{aligned} \tag{B.6}$$

where in $\overset{a}{\le}$, we applied Cauchy-Schwarz inequality. Since $\eta^k = \min\left(\frac{\epsilon}{Ln_0\|\mathbf{v}^k\|}, \frac{1}{2Ln_0}\right) \le \frac{1}{2Ln_0} \le \frac{1}{2L}$, we have

$$\eta^k\left(\frac{1}{2} - \frac{\eta^k L}{2}\right)\left\|\mathbf{v}^k\right\|^2 \ge \frac{1}{4}\eta^k\left\|\mathbf{v}^k\right\|^2 = \frac{\epsilon^2}{8n_0L}\min\left(2\left\|\frac{\mathbf{v}^k}{\epsilon}\right\|, \left\|\frac{\mathbf{v}^k}{\epsilon}\right\|^2\right) \overset{a}{\ge} \frac{\epsilon\|\mathbf{v}^k\| - 2\epsilon^2}{4n_0L}, \tag{B.7}$$

where in $\overset{a}{\ge}$, we use $V(x) = \min\left(|x|, \frac{x^2}{2}\right) \ge |x| - 2$ for all x. Hence

$$\begin{aligned} f(\mathbf{x}^{k+1}) &\le f(\mathbf{x}^k) - \frac{\epsilon\|\mathbf{v}^k\|}{4Ln_0} + \frac{\epsilon^2}{2n_0L} + \frac{\eta^k}{2}\left\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\right\|^2 \\ &\overset{\eta^k \le \frac{1}{2Ln_0}}{\le} f(\mathbf{x}^k) - \frac{\epsilon\|\mathbf{v}^k\|}{4Ln_0} + \frac{\epsilon^2}{2n_0L} + \frac{1}{4Ln_0}\left\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\right\|^2. \end{aligned} \tag{B.8}$$

Taking expectation on the above display and using Lemma 2, we have

$$\mathbb{E}_{k_0}f(\mathbf{x}^{k+1}) - \mathbb{E}_{k_0}f(\mathbf{x}^k) \le -\frac{\epsilon}{4Ln_0}\mathbb{E}_{k_0}\left\|\mathbf{v}^k\right\| + \frac{3\epsilon^2}{4Ln_0}. \tag{B.9}$$

□

The proof is done via the following lemma:

Lemma 5. *Under Assumption 1, for all $k \geq 0$, we have*

$$\mathbb{E}\|\nabla f(\mathbf{x}^k)\| \leq \mathbb{E}\|\mathbf{v}^k\| + \epsilon. \tag{B.10}$$

Proof. By taking the total expectation in Lemma 2, we have

$$\mathbb{E}\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \epsilon^2. \tag{B.11}$$

Then by Jensen's inequality

$$\left(\mathbb{E}\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|\right)^2 \leq \mathbb{E}\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \epsilon^2.$$

So using triangle inequality

$$\begin{aligned}\mathbb{E}\|\nabla f(\mathbf{x}^k)\| &= \mathbb{E}\|\mathbf{v}^k - (\mathbf{v}^k - \nabla f(\mathbf{x}^k))\| \\ &\leq \mathbb{E}\|\mathbf{v}^k\| + \mathbb{E}\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\| \leq \mathbb{E}\|\mathbf{v}^k\| + \epsilon. \end{aligned} \tag{B.12}$$

This completes our proof. □

Now, we are ready to prove Theorem 1.

Proof of Theorem 1. Taking full expectation on Lemma 4, and telescoping the results from $k = 0$ to $K - 1$, we have

$$\frac{\epsilon}{4Ln_0}\sum_{k=0}^{K-1}\mathbb{E}\|\mathbf{v}^k\| \leq f(\mathbf{x}^0) - \mathbb{E}f(\mathbf{x}^K) + \frac{3K\epsilon^2}{4Ln_0} \overset{\mathbb{E}f(\mathbf{x}^K)\geq f^*}{\leq} \Delta + \frac{3K\epsilon^2}{4Ln_0}. \tag{B.13}$$

Diving $\frac{4Ln_0}{\epsilon}K$ both sides of (B.13), and using $K = \lfloor \frac{4L\Delta n_0}{\epsilon^2} \rfloor + 1 \geq \frac{4L\Delta n_0}{\epsilon^2}$, we have

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\mathbf{v}^k\| \leq \Delta \cdot \frac{4Ln_0}{\epsilon}\frac{1}{K} + 3\epsilon \leq 4\epsilon. \tag{B.14}$$

Then from the choose of $\tilde{\mathbf{x}}$, we have

$$\mathbb{E}\|\nabla f(\tilde{\mathbf{x}})\| = \frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\nabla f(\mathbf{x}^k)\| \overset{(B.10)}{\leq} \frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\mathbf{v}^k\| + \epsilon \overset{(B.14)}{\leq} 5\epsilon. \tag{B.15}$$

To compute the gradient cost, note in each q iterations we access for one time S_1 stochastic gradients and for q times of S_2 stochastic gradients, and hence the cost is

$$\begin{aligned}\left\lceil K \cdot \frac{1}{q} \right\rceil S_1 + KS_2 &\overset{S_1=qS_2}{\leq} 2K \cdot S_2 + S_1 \\ &\leq 2\left(\frac{4Ln_0\Delta}{\epsilon^2}\right)\frac{2\sigma}{\epsilon n_0} + \frac{2\sigma^2}{\epsilon^2} + 2S_2 \\ &= \frac{16L\sigma\Delta}{\epsilon^3} + \frac{2\sigma^2}{\epsilon^2} + \frac{4\sigma}{n_0\epsilon}. \end{aligned} \tag{B.16}$$

This concludes a gradient cost of $16L\Delta\sigma\epsilon^{-3} + 2\sigma^2\epsilon^{-2} + 4\sigma n_0^{-1}\epsilon^{-1}$. $\square$

Proof of Theorem 2. For Lemma 2, we have

$$\mathbb{E}_{k_0}\|\mathbf{v}^{k_0} - \nabla f(\mathbf{x}^{k_0})\|^2 = \mathbb{E}_{k_0}\|\nabla f(\mathbf{x}^{k_0}) - \nabla f(\mathbf{x}^{k_0})\|^2 = 0. \tag{B.17}$$

With the above display, applying Lemma 1 with $\epsilon_1 = \frac{\epsilon}{Ln_0}$, and $S_2 = \frac{n^{1/2}}{\epsilon n_0}$, $K = k - k_0 \leq q = n_0 n^{1/2}$, we have

$$\mathbb{E}_{k_0}\|\mathbf{v}^{k_0} - \nabla f(\mathbf{x}^{k_0})\|^2 \leq n_0 n^{1/2} L^2 \cdot \frac{\epsilon^2}{L^2 n_0^2} \cdot \frac{\epsilon n_0}{n^{1/2}} + \mathbb{E}_{k_0}\|\mathbf{v}^{k_0} - \nabla f(\mathbf{x}^{k_0})\|^2 \overset{(B.1)}{=} \epsilon^2. \tag{B.18}$$

So Lemma 2 holds. Then from the same technique of on-line case, we can obtain (B.2) and (5), and (B.15). The gradient cost analysis is computed as:

$$\begin{aligned}\left\lceil K \cdot \frac{1}{q} \right\rceil S_1 + KS_2 &\overset{S_1=qS_2}{\leq} 2K + S_1 \\ &\leq 2\left(\frac{4Ln_0\Delta}{\epsilon^2}\right)\frac{n^{1/2}}{n_0} + n + 2S_2 \\ &= \frac{8(L\Delta)\cdot n^{1/2}}{\epsilon^2} + n + \frac{2n^{1/2}}{n_0}. \end{aligned} \tag{B.19}$$

This concludes a gradient cost of $n + 8(L\Delta)\cdot n^{1/2}\epsilon^{-2} + 2n_0^{-1}n^{1/2}$. $\square$

B.3 Proof of High Probability Results for FSP

Set $\mathcal{K}$ be the time when Algorithm 1 stops. We have $\mathcal{K} = 0$ if $\|\mathbf{v}^0\| < 2\epsilon$, and $\mathcal{K} = \inf\{k \geq 0 : \|\mathbf{v}^k\| < 2\epsilon\} + 1$ if $\|\mathbf{v}^0\| \geq 2\epsilon$. It is a random stopping time. Let $K_0 = \lfloor 4L\Delta n_0\epsilon^{-2}\rfloor + 2$. We have the following lemma:

Lemma 6. *Set the parameters S_1, S_2, η, and q as in Theorem 4. Then under the Assumption 2, for fixed K_0, define the event:*

$$\boldsymbol{\mathcal{H}}_{K_0} = \left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \epsilon \cdot \tilde{\epsilon}, \quad \forall k \leq \min(\mathcal{K}, K_0)\right).$$

we have $\boldsymbol{\mathcal{H}}_{K_0}$ occurs with probability at least $1-p$.

Proof of Lemma 6. Because when $k \geq \mathcal{K}$, the algorithm has already stopped. So if $\mathcal{K} \leq k \leq K_0$, we can define a virtual update as $\mathbf{x}^{k+1} = \mathbf{x}^k$, and $\mathbf{v}^k$ is still generated by Line 3 and Line 5 in Algorithm 1.

Then let the event $\tilde{\boldsymbol{\mathcal{H}}}_k = \left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \epsilon \cdot \tilde{\epsilon}\right)$, with $0 \leq k \leq K_0$. We want to prove that for any k with $0 \leq k \leq K_0$, $\tilde{\boldsymbol{\mathcal{H}}}_k$ occurs with probability at least $1 - p/(K_0+1)$. If so, using the fact that

$$\boldsymbol{\mathcal{H}}_{K_0} \supseteq \left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \epsilon \cdot \tilde{\epsilon}, \quad \forall k \leq K_0\right) = \bigcap_{k=0}^{K_0} (\tilde{\boldsymbol{\mathcal{H}}}_k),$$

we have

$$\mathbb{P}(\boldsymbol{\mathcal{H}}_{K_0}) \geq \mathbb{P}\left(\bigcap_{k=0}^{K_0} (\tilde{\boldsymbol{\mathcal{H}}}_k)\right) = \mathbb{P}\left(\left(\bigcup_{k=0}^{K_0} (\tilde{\boldsymbol{\mathcal{H}}}_k)^c\right)^c\right) \geq 1 - \sum_{k=0}^{K_0} \mathbb{P}(\tilde{\boldsymbol{\mathcal{H}}}_k^c) = 1 - p.$$

We prove that $\tilde{\boldsymbol{\mathcal{H}}}_k$ occurs with probability $1 - p/(K_0+1)$ for any k with $0 \leq k \leq K_0$.

Let $\boldsymbol{\xi}^k$ with $k \geq 0$ denote the randomness in maintaining Spider $\mathbf{v}^k$ at iteration k. And $\boldsymbol{\mathcal{F}}^k = \sigma\{\boldsymbol{\xi}^0, \cdots \boldsymbol{\xi}^k\}$, where $\sigma\{\cdot\}$ denotes the sigma field. We know that $\mathbf{x}^k$ and $\mathbf{v}^{k-1}$ are measurable on $\boldsymbol{\mathcal{F}}^{k-1}$.

Then given $\boldsymbol{\mathcal{F}}^{k-1}$, if $k = \lfloor k/q \rfloor q$, we set

$$\boldsymbol{\epsilon}_{k,i} = \frac{1}{S_1}\left(\nabla f_{\mathcal{S}_1(i)}(\mathbf{x}^k) - \nabla f(\mathbf{x}^k)\right)$$

where i is the index with $\mathcal{S}_1(i)$ denoting the i-th random component function selected at iteration k and $1 \leq i \leq S_1$. We have

$$\mathbb{E}[\boldsymbol{\epsilon}_{k,i} | \boldsymbol{\mathcal{F}}^{k-1}] = 0, \quad \|\boldsymbol{\epsilon}_{k,i}\| \overset{Assum. 2(iii')}{\leq} \frac{\sigma}{S_1}.$$

Then from Proposition 2, we have

$$\begin{aligned}
\mathbb{P}\left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \geq \epsilon \cdot \tilde{\epsilon} \mid \boldsymbol{\mathcal{F}}^{k-1}\right) &= \mathbb{P}\left(\left\|\sum_{i=1}^{S_1} \boldsymbol{\epsilon}_{k,i}\right\|^2 \geq \epsilon \cdot \tilde{\epsilon} \mid \boldsymbol{\mathcal{F}}^{k-1}\right) \\
\leq \quad 4\exp\left(-\frac{\epsilon \cdot \tilde{\epsilon}}{4 S_1 \frac{\sigma^2}{S_1^2}}\right) &\overset{S_1 = \frac{2\sigma^2}{\epsilon^2},\ \tilde{\epsilon} = 10\epsilon \log(4(K_0+1)/p)}{\leq} \frac{p}{K_0+1}.
\end{aligned} \tag{B.20}$$

So $\mathbb{P}\left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \geq \epsilon \cdot \tilde{\epsilon}\right) \leq \frac{p}{K_0+1}$.

When $k \neq \lfloor k/q \rfloor q$, set $k_0 = \lfloor k/q \rfloor q$, and

$$\boldsymbol{\epsilon}_{j,i} = \frac{1}{S_2}\left(\nabla f_{\mathcal{S}_2(i)}(\mathbf{x}^j) - \nabla f_{\mathcal{S}_2(i)}(\mathbf{x}^{j-1}) - \nabla f(\mathbf{x}^j) + \nabla f(\mathbf{x}^{j-1})\right)$$

where i is the index with $\mathcal{S}_2(i)$ denoting the i-th random component function selected at iteration

k, $1 \le i \le S_2$ and $k_0 \le j \le k$. We have

$$\mathbb{E}[\boldsymbol{\epsilon}_{j,i}|\boldsymbol{\mathcal{F}}^{j-1}] = 0.$$

For any $\mathbf{x}$ and $\mathbf{y}$, we have

$$\begin{aligned} \|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| &= \left\| \frac{1}{n} \sum_{i=1}^{n} (\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})) \right\| \\ \le \quad \frac{1}{n} \sum_{i=1}^{n} \|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\| &\overset{Assum.2\ (ii')}{\le} L\|\mathbf{x} - \mathbf{y}\|, \end{aligned} \tag{B.21}$$

So $f(\mathbf{x})$ also have L-Lipschitz continuous gradient.

Then from the update rule if $k < \mathcal{K}$, we have $\|\mathbf{x}^{k+1} - \mathbf{x}^k\| = \|\eta(\mathbf{v}^k/\|\mathbf{v}^k\|)\| = \eta = \frac{\epsilon}{Ln_0}$, if $k \ge \mathcal{K}$, we have $\|\mathbf{x}^{k+1} - \mathbf{x}^k\| = 0 \le \frac{\epsilon}{Ln_0}$. We have

$$\begin{aligned} & \|\epsilon_{j,i}\| \\ \le \quad & \frac{1}{S_2} \left(\|\nabla f_i(\mathbf{x}^j) - \nabla f_i(\mathbf{x}^{j-1})\| + \|\nabla f(\mathbf{x}^j) - \nabla f(\mathbf{x}^{j-1})\| \right) \\ \overset{(B.21),\ Assum.2\ (ii')}{\le} \quad & \frac{2L}{S_2} \|\mathbf{x}^j - \mathbf{x}^{j-1}\| \le \frac{2\epsilon}{S_2 n_0}, \end{aligned} \tag{B.22}$$

for all $k_0 < j \le k$ and $1 \le i \le S_2$. On the other hand, we have

$$\begin{aligned} & \|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\| \\ = \quad & \|\nabla f_{S_2}(\mathbf{x}^k) - \nabla f_{S_2}(\mathbf{x}^{k-1}) - \nabla f(\mathbf{x}^k) + \nabla f(\mathbf{x}^{k-1}) + (\mathbf{v}^{k-1} - \nabla f(\mathbf{x}^{k-1}))\| \\ = \quad & \left\| \sum_{j=k_0+1}^{k} \Big(\nabla f_{S_2}(\mathbf{x}^k) - \nabla f_{S_2}(\mathbf{x}^{k-1}) - \nabla f(\mathbf{x}^k) + \nabla f(\mathbf{x}^{k-1}) \Big) + \nabla f_{S_1}(\mathbf{x}^{k_0}) - \nabla f(\mathbf{x}^{k_0}) \right\| \\ = \quad & \left\| \sum_{j=k_0+1}^{k} \sum_{i=1}^{S_1} \boldsymbol{\epsilon}_{j,i} + \sum_{i=1}^{S_2} \boldsymbol{\epsilon}_{k_0,i} \right\|. \end{aligned} \tag{B.23}$$

Plugging (B.22) and (B.23) together, and using Proposition 2, we have

$$\begin{aligned}
& \mathbb{P}\left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \geq \epsilon \cdot \tilde{\epsilon} \mid \boldsymbol{\mathcal{F}}^{k_0-1}\right) \\
\leq \quad & 4\exp\left(-\frac{\epsilon \cdot \tilde{\epsilon}}{4S_1\frac{\sigma^2}{S_1^2} + 4S_2(k-k_0)\frac{4\epsilon^2}{S_2^2 n_0^2}}\right) \\
\leq \quad & 4\exp\left(-\frac{\epsilon \cdot \tilde{\epsilon}}{4S_1\frac{\sigma^2}{S_1^2} + 4S_2 q\frac{4\epsilon^2}{S_2^2 n_0^2}}\right) \\
\overset{a}{=} \quad & 4\exp\left(-\frac{\epsilon^2 10\log(4(K_0+1)/p)}{4\sigma^2\frac{\epsilon^2}{2\sigma^2} + \frac{4\epsilon n_0}{2\sigma}\frac{\sigma n_0}{\epsilon}\frac{4\epsilon^2}{n_0^2}}\right) \leq \frac{p}{K_0+1},
\end{aligned} \tag{B.24}$$

where in $\overset{a}{=}$, we use $S_1 = \frac{2\sigma^2}{\epsilon^2}$, $S_2 = \frac{2\sigma}{\epsilon n_0}$, and $q = \frac{\sigma n_0}{\epsilon}$. So $\mathbb{P}\left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \geq \epsilon \cdot \tilde{\epsilon}\right) \leq \frac{p}{K_0+1}$, which completes the proof.

$\square$

Lemma 7. *Under Assumption 2, we have that on $\boldsymbol{\mathcal{H}}_{K_0} \cap (\mathcal{K} > K_0)$, for all $0 \leq k \leq K_0$,*

$$f(\mathbf{x}^{k+1}) - f(\mathbf{x}^k) \leq -\frac{\epsilon \cdot \tilde{\epsilon}}{4Ln_0}. \tag{B.25}$$

and hence

$$f(\mathbf{x}^{K_0+1}) - f(\mathbf{x}^0) \leq -\frac{\epsilon \cdot \tilde{\epsilon}}{4Ln_0} \cdot (K_0).$$

Proof of Lemma 7. Let $\eta^k := \eta / \|\mathbf{v}^k\|$. Since f has L-Lipschitz continuous gradient from (B.21), we have

$$f(\mathbf{x}^{k+1}) \overset{(B.6)}{\leq} f(\mathbf{x}^k) - \eta^k\left(\frac{1}{2} - \frac{\eta^k L}{2}\right)\left\|\mathbf{v}^k\right\|^2 + \frac{\eta^k}{2}\left\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\right\|^2. \tag{B.26}$$

Because we are on the event $\boldsymbol{\mathcal{H}}_{K_0} \cap (\mathcal{K} > K_0)$, so $\mathcal{K} - 1 \geq K_0$, then for all $0 \leq k \leq K_0$, we have $\|\mathbf{v}^k\| \geq 2\epsilon$, thus

$$\eta^k = \frac{\epsilon}{Ln_0}\frac{1}{\|\mathbf{v}^k\|} \overset{\|\mathbf{v}^k\| \geq 2\tilde{\epsilon} \geq 2\epsilon}{\leq} \frac{1}{2Ln_0} \leq \frac{1}{2L},$$

we have

$$\eta^k\left(\frac{1}{2} - \frac{\eta^k L}{2}\right)\left\|\mathbf{v}^k\right\|^2 \geq \frac{1}{4} \cdot \frac{\epsilon}{Ln_0\|\mathbf{v}^k\|}\|\mathbf{v}^k\|^2 \overset{\|\mathbf{v}^k\| \geq 2\tilde{\epsilon}}{\geq} \frac{\epsilon \cdot \tilde{\epsilon}}{2Ln_0}, \tag{B.27}$$

and for $\boldsymbol{\mathcal{H}}_{K_0}$ happens, we also have

$$\frac{\eta^k}{2}\left\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\right\|^2 \overset{\eta^k \leq \frac{1}{2Ln_0}}{\leq} \frac{\epsilon \cdot \tilde{\epsilon}}{4Ln_0}.$$

Hence

$$f(\mathbf{x}^{k+1}) \le f(\mathbf{x}^k) - \frac{\epsilon \cdot \tilde{\epsilon}}{2Ln_0} + \frac{\eta^k}{2} \left\| \mathbf{v}^k - \nabla f(\mathbf{x}^k) \right\|^2 \le f(\mathbf{x}^k) - \frac{\epsilon \cdot \tilde{\epsilon}}{4Ln_0}, \tag{B.28}$$

By telescoping (B.28) from 0 to K_0, we have

$$f(\mathbf{x}^{K_0+1}) - f(\mathbf{x}^0) \le -\frac{\epsilon \cdot \tilde{\epsilon}}{4Ln_0} \cdot (K_0).$$

$\square$

Now, we are ready to prove Theorem 4.

Proof of Theorem 4. We only want to prove $(\mathcal{K} \le K_0) \supseteq \boldsymbol{\mathcal{H}}_{K_0}$, so if $\boldsymbol{\mathcal{H}}_{K_0}$ occurs, we have $\mathcal{K} \le K_0$, and $\|\mathbf{v}^{\mathcal{K}}\| \le 2\tilde{\epsilon}$. Because $\|\mathbf{v}^{\mathcal{K}} - \nabla f(\mathbf{x}^{\mathcal{K}})\| \le \sqrt{\epsilon \cdot \tilde{\epsilon}} \le \tilde{\epsilon}$ occurs in $\boldsymbol{\mathcal{H}}_{K_0}$, so $\|\nabla f(\mathbf{x}^{\mathcal{K}})\| \le 3\tilde{\epsilon}$.

If $(\mathcal{K} > K_0)$ and $\boldsymbol{\mathcal{H}}_{K_0}$ occur, plugging in $K_0 = \lfloor \frac{4L\Delta n_0}{\epsilon^2} \rfloor + 2 \ge \frac{4L\Delta n_0}{\epsilon^2} + 1 \ge \frac{4L\Delta n_0}{\epsilon \cdot \tilde{\epsilon}} + 1$, then from Lemma 7 at each iteration the function value descends by at least $\epsilon \cdot \tilde{\epsilon}/(4Ln_0)$, We thus have

$$-\Delta \le f^* - f(\mathbf{x}^0) \le f(\mathbf{x}^{K_0}) - f(\mathbf{x}^0) \le -\left(\Delta + \frac{\epsilon \cdot \tilde{\epsilon}}{4Ln_0}\right),$$

contradicting the fact that $-\Delta > -\left(\Delta + \frac{\epsilon \cdot \tilde{\epsilon}}{4Ln_0}\right)$. This indicates $(\mathcal{K} \le K_0) \supseteq \boldsymbol{\mathcal{H}}_{K_0}$. From Lemma 6, with probability $1-p$, $\boldsymbol{\mathcal{H}}_{K_0}$ occurs, and then $\|\mathbf{v}^{\mathcal{K}}\| \le 2\tilde{\epsilon}$ and $\|\nabla f(\mathbf{x}^{\mathcal{K}})\| \le 3\tilde{\epsilon}$.

Then gradient cost can be bounded by the same way in Theorem 2 as:

$$\begin{aligned}
\left\lceil K_0 \cdot \frac{1}{q} \right\rceil S_1 + K_0 S_2 \quad &\overset{S_1 = qS_2}{\le} \quad 2K_0 \cdot S_2 + S_1 \\
&\le \quad 2\left(\frac{\Delta}{\epsilon^2/(4Ln_0)}\right) \cdot S_2 + S_1 + 4S_2 \\
&\le \quad 2\left(\frac{4Ln_0\Delta}{\epsilon^2}\right)\frac{2\sigma}{\epsilon n_0} + \frac{2\sigma^2}{\epsilon^2} + \frac{8\sigma}{\epsilon n_0} \\
&= \quad \frac{16L\sigma\Delta}{\epsilon^3} + \frac{2\sigma^2}{\epsilon^2} + \frac{8\sigma}{\epsilon n_0}.
\end{aligned} \tag{B.29}$$

$\square$

Proof of Theorem 5. We first verify that $\tilde{\boldsymbol{\mathcal{H}}}_k = \left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \le \epsilon \cdot \tilde{\epsilon}\right)$ with $0 \le k \le K_0$ occurs with probability $1 - \delta/(K_0+1)$ for any k.

When $k = \lfloor k/q \rfloor q$, we have $\mathbf{v}^k = \nabla f(\mathbf{x}^k)$.

When $k \ne \lfloor k/q \rfloor q$, set $k_0 = \lfloor k/q \rfloor q$, and

$$\epsilon_{j,i} = \frac{1}{S_2}\left(\nabla f_{\mathcal{S}_2(i)}(\mathbf{x}^j) - f_{\mathcal{S}_2(i)}(\mathbf{x}^{j-1}) - \nabla f(\mathbf{x}^j) + \nabla f(\mathbf{x}^{j-1})\right)$$

where i is the index with $\mathcal{S}_2(i)$ denoting the i-th random component function selected at iteration

k, from (B.22), we have

$$\mathbb{E}\left[\boldsymbol{\epsilon}_{j,i}|\boldsymbol{\mathcal{F}}^{j-1}\right]=0, \quad \|\boldsymbol{\epsilon}_{j,i}\|\le\frac{2\epsilon}{S_2n_0},$$

for all $k_0<j\le k$ and $1\le i\le S_2$. On the other hand

$$\begin{aligned}
&\left\|\mathbf{v}^k-\nabla f(\mathbf{x}^k)\right\|\\
=&\left\|\nabla f_{\mathcal{S}_2}(\mathbf{x}^k)-\nabla f_{\mathcal{S}_2}(\mathbf{x}^{k-1})-\nabla f(\mathbf{x}^k)+\nabla f(\mathbf{x}^{k-1})+(\mathbf{v}^{k-1}-\nabla f(\mathbf{x}^{k-1})\right\|\\
=&\left\|\sum_{j=k_0+1}^{k}\left(\nabla f_{\mathcal{S}_2}(\mathbf{x}^k)-\nabla f_{\mathcal{S}_2}(\mathbf{x}^{k-1})-\nabla f(\mathbf{x}^k)+\nabla f(\mathbf{x}^{k-1})\right)+\nabla f_{\mathcal{S}_1}(\mathbf{x}^{k_0})-\nabla f(\mathbf{x}^{k_0})\right\|\\
=&\left\|\sum_{j=k_0+1}^{k}\sum_{i=1}^{S_1}\boldsymbol{\epsilon}_{j,i}\right\|. \qquad\qquad (B.30)
\end{aligned}$$

Then from Proposition 2, we have

$$\begin{aligned}
&\mathbb{P}\left(\|\mathbf{v}^k-\nabla f(\mathbf{x}^k)\|^2\ge\epsilon\cdot\tilde{\epsilon}\mid\boldsymbol{\mathcal{F}}^{k_0-1}\right)\\
\le\ & 4\exp\left(-\frac{\epsilon\cdot\tilde{\epsilon}}{4S_2(k-k_0)\frac{4\epsilon^2}{S_2^2n_0^2}}\right)\\
\le\ & 4\exp\left(-\frac{\epsilon\cdot\tilde{\epsilon}}{4S_2q\frac{4\epsilon^2}{S_2^2n_0^2}}\right)\\
\overset{a}{=}\ & 4\exp\left(-\frac{\epsilon^2 16\log(4(K_0+1)/p)}{4n_0n^{1/2}\frac{n_0}{n^{1/2}}\frac{4\epsilon^2}{n_0^2}}\right)\le\frac{p}{K_0+1}, \qquad\qquad (B.31)
\end{aligned}$$

where in $\overset{a}{=}$, we use $S_2=n^{1/2}/n_0$, and $q=n_0n^{1/2}$. So $\mathbb{P}\left(\|\mathbf{v}^k-\nabla f(\mathbf{x}^k)\|^2\ge\epsilon\cdot\tilde{\epsilon}\right)\le\frac{p}{K_0+1}$, which completes the proof.

Thus Lemma 2 holds. Then using the same technique of Lemma 7 and Theorem 4, we have $(\mathcal{K}\le K_0)\supseteq\boldsymbol{\mathcal{H}}_{K_0}$. With probability at least $1-p$, $\boldsymbol{\mathcal{H}}_{K_0}$ occurs, and $\|\mathbf{v}^{\mathcal{K}}\|\le2\tilde{\epsilon}$ and $\|\nabla f(\mathbf{x}^{\mathcal{K}})\|\le3\tilde{\epsilon}$.

$$\begin{aligned}
\left\lceil K_0\cdot\frac{1}{q}\right\rceil S_1+K_0S_2 &\overset{S_1=qS_2}{\le} 2K_0\cdot S_2+S_1\\
&\le 2\frac{\Delta}{\epsilon^2/(4Ln_0)}\cdot S_2+S_1+4S_2\\
&= 2\left(\frac{4Ln_0\Delta}{\epsilon^2}\right)\frac{n^{1/2}}{n_0}+n+4n_0^{-1}n^{1/2}\\
&= \frac{8(L\Delta)\cdot n^{1/2}}{\epsilon^2}+n+4n_0^{-1}n^{1/2}. \qquad\qquad (B.32)
\end{aligned}$$

□

B.4 Proof of Theorem 6 for SSP

We first restate the NEON result in Allen-Zhu & Li (2018) for NC-search in the following Theorem:

Theorem 9 (Theorem 1 in Allen-Zhu & Li (2018), NEON2 (on-line))**.** *Under the Assumption 2 (including (ii')), for every point $\mathbf{x}_0 \in \mathcal{R}^d$, for every $\delta \in (0, L]$, and every $p \in (0,1)$, the NEON2 (NC search) output*

$$\mathbf{w} = \text{NEON2}(f, \mathbf{x}_0, \delta, p)$$

satisfies that, with probability at least $1-p$:

1. *if $\mathbf{w} = \perp$, then $\nabla f^2(\mathbf{x}_0) \succeq -\delta I$.*
2. *if $\mathbf{w} \neq \perp$, then $\|\mathbf{w}\|_2 = 1$, and $\mathbf{w}^{\mathrm{T}} \nabla^2 f(\mathbf{x}_0) \mathbf{w} \le \frac{\delta}{2}$.*

Moreover, the total number of stochastic gradient evaluations are $O\left(\log^2((d/p)) L^2 \delta^{-2}\right)$.

One can refer to Allen-Zhu & Li (2018) for more details.

Now we prove Theorem 6:

From Algorithm 2, we can find that all the randomness in iteration k come from 3 parts: 1) maintaining SPIDER $\mathbf{v}^k$ (Line 7-11 and 17-21); 2) to conducting NC-search in Line 2 (if $\mathrm{mod}(k, \mathscr{K}) = 0$); 3) choosing a random direction to update $\mathbf{x}^k$ in Line 5 (if Algorithm 2 performs first-order updates). We denote the randomness from the three parts as $\boldsymbol{\xi}_k^1$, $\boldsymbol{\xi}_k^2$, $\boldsymbol{\xi}_k^3$, respectively. Let $\boldsymbol{\mathcal{F}}^k$ be the filtration involving the full information of $\mathbf{x}_{0:k}, \mathbf{v}_{0:k}$, i.e $\boldsymbol{\mathcal{F}}^k = \sigma\left\{\boldsymbol{\xi}_{0:k}^1, \boldsymbol{\xi}_{0:k}^2, \boldsymbol{\xi}_{0:k-1}^3\right\}$. So the randomness in iteration k given $\boldsymbol{\mathcal{F}}^k$ only comes from $\boldsymbol{\xi}_k^3$ (choosing a random direction in Line 5).

Let the random index $\mathcal{I}_k = 1$, if Algorithm 2 plans to perform the first-order update, $\mathcal{I}_k = 2$, if it plans to perform the second-order update, we know that $\mathcal{I}_k$ is measurable on $\boldsymbol{\mathcal{F}}^{\lfloor k/\mathscr{K} \rfloor \mathscr{K}}$ and also on $\boldsymbol{\mathcal{F}}^k$. Because the algorithm shall be stopped if it finds $\mathbf{v}^k \ge 2\tilde{\epsilon}$ when it plans to do first-order descent, we can define a virtual update as $\mathbf{x}^{k+1} = \mathbf{x}^k$ in Line 12 and 25, with others unchanged if the algorithm has stopped. Let $\boldsymbol{\mathcal{H}}_1^k$ denotes the event that algorithm has not stopped before k, i.e.

$$\boldsymbol{\mathcal{H}}_1^k = \bigcap_{i=0}^{k} \left(\left(\|\mathbf{v}^k\| \ge 2\tilde{\epsilon} \cap \mathcal{I}_k = 1 \right) \bigcup \mathcal{I}_k = 2 \right),$$

we have $\boldsymbol{\mathcal{H}}_1^k \in \boldsymbol{\mathcal{F}}^k$, and $\boldsymbol{\mathcal{H}}_1^1 \supseteq \boldsymbol{\mathcal{H}}_1^2 \supseteq \cdots \supseteq \boldsymbol{\mathcal{H}}_1^k$. Let $\boldsymbol{\mathcal{H}}_2^{\mathscr{K} j}$ denotes the event that the NC-search in iteration $\mathscr{K} j$ runs successfully. And $\boldsymbol{\mathcal{H}}_3^k$ denotes the event that

$$\boldsymbol{\mathcal{H}}_3^k = \left(\bigcap_{i=0}^{k} \left(\|\mathbf{v}^i - \nabla f(\mathbf{x}^i)\|^2 \le \epsilon \cdot \tilde{\epsilon} \right) \right) \bigcap \left(\bigcap_{j=0}^{\lfloor k/\mathscr{K} \rfloor} \boldsymbol{\mathcal{H}}_2^{j \cdot \mathscr{K}} \right).$$

We know that $\boldsymbol{\mathcal{H}}_3^k \in \boldsymbol{\mathcal{F}}^k$, and $\boldsymbol{\mathcal{H}}_3^1 \supseteq \boldsymbol{\mathcal{H}}_3^2 \supseteq \cdots \supseteq \boldsymbol{\mathcal{H}}_3^k$. And if $\boldsymbol{\mathcal{H}}_3^k$ happens, all NC-search before iteration k run successfully and $\|\mathbf{v}^i - \nabla f(\mathbf{x}^i)\|^2 \le \epsilon \cdot \tilde{\epsilon}$ for all $0 \le i \le k$.

Lemma 8. *With the setting of Theorem 6, and under the Assumption 3, we have*

$$\mathbb{P}\left(\boldsymbol{\mathcal{H}}_3^{K_0}\right) \geq \frac{7}{8}. \tag{B.33}$$

Proof. Let event $\tilde{\boldsymbol{\mathcal{H}}}^k = \left(\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \epsilon \cdot \tilde{\epsilon}\right)$, with $0 \leq k \leq K_0$. Once we prove that $\tilde{\boldsymbol{\mathcal{H}}}^k$ occurs with probability at least $1 - \frac{1}{16(K_0+1)}$, we have $\mathbb{P}\left(\bigcap_{i=0}^{K_0}\left(\|\mathbf{v}^i - \nabla f(\mathbf{x}^i)\|^2 \leq \epsilon \cdot \tilde{\epsilon}\right)\right) \geq 1 - \frac{1}{16}$. On the other hand, from Theorem 9, we know each time the NC-search conducts successfully at probability $1 - \frac{1}{16J}$, so $\mathbb{P}(\boldsymbol{\mathcal{H}}_2^{K_0}) \geq 1 - \frac{1}{16}$. Combining the above results, we obtain $\mathbb{P}\left(\boldsymbol{\mathcal{H}}_3^{K_0}\right) \geq \frac{7}{8}$.

To prove $\mathbb{P}(\tilde{\boldsymbol{\mathcal{H}}}^k) \geq 1 - \frac{1}{16(K_0+1)}$, consider the filtration $\boldsymbol{\mathcal{F}}_2^k = \sigma\left\{\boldsymbol{\xi}_{0:k-1}^1, \boldsymbol{\xi}_{0:k}^2, \cdots, \boldsymbol{\xi}_{0:k-1}^3\right\}$, which involves the full information of $\mathbf{x}_{0:k}$. We know $\mathbf{x}^k$ is measurable on $\boldsymbol{\mathcal{F}}_2^k$. Given $\boldsymbol{\mathcal{F}}_2^k$, we have

$$\mathbb{E}_i\left[\nabla f_i(\mathbf{x}^k) - \nabla f(\mathbf{x}^k) \mid \boldsymbol{\mathcal{F}}_2^k\right] = \mathbf{0}$$

when $\mathrm{mod}(k,p) = 0$. For $\mathrm{mod}(k,p) \neq 0$, we have

$$\mathbb{E}_i\left[\nabla f_i(\mathbf{x}^k) - \nabla f_i(\mathbf{x}^{k-1}) - \left(\nabla f(\mathbf{x}^k) - \nabla f(\mathbf{x}^{k-1})\right) \mid \boldsymbol{\mathcal{F}}_2^k\right] = \mathbf{0}.$$

Because $\mathbf{x}^k$ is generated by one of the three ways:

1. Algorithm 2 performs First-order descent, we have $\|\mathbf{x}^k - \mathbf{x}^{k-1}\| = \|\eta(\mathbf{v}^{k-1}/\|\mathbf{v}^{k-1}\|)\| = \eta = \frac{\epsilon}{Ln_0}$.
2. Algorithm 2 performs Second-order descent, we have $\|\mathbf{x}^k - \mathbf{x}^{k-1}\| = \eta = \frac{\epsilon}{Ln_0}$.
3. Algorithm 2 has already stopped. $\|\mathbf{x}^k - \mathbf{x}^{k-1}\| = 0 \leq \frac{\epsilon}{Ln_0}$.

So $\mathbf{v}^k - \nabla f(\mathbf{x}^k)$ is martingale, and the second moment of its difference is bounded by $\frac{\epsilon}{Ln_0}$. We can find that the parameters S_1, S_2, η, with $\tilde{\epsilon} = 16\epsilon\log(64(K_0+1))$ for on-line case and $\tilde{\epsilon} = 10\epsilon\log(64(K_0+1))$ for off-line case are set as the same in Lemma 6 with $p = \frac{1}{16}$. Thus using the same technique of Lemma 6, we can obtain $\mathbb{P}(\tilde{\boldsymbol{\mathcal{H}}}^k) \geq 1 - \frac{1}{16(K_0+1)}$ for all $0 \leq k \leq K_0$. $\square$

Let $\boldsymbol{\mathcal{H}}_4^k = \boldsymbol{\mathcal{H}}_1^k \cap \boldsymbol{\mathcal{H}}_3^k$. We show that Theorem 6 is essentially to measure the probability of the event that $\left(\boldsymbol{\mathcal{H}}_1^{K_0}\right)^c \bigcap \boldsymbol{\mathcal{H}}_3^{K_0}$.

Lemma 9. *If $\left(\boldsymbol{\mathcal{H}}_1^{K_0}\right)^c \bigcap \boldsymbol{\mathcal{H}}_3^{K_0}$ happens, Algorithm 2 outputs an $\mathbf{x}^k$ satisfying (3.12) before K_0 iterations.*

Proof. Because $\left(\boldsymbol{\mathcal{H}}_1^{K_0}\right)^c$ happens, we know that Algorithm 2 has already stopped before K_0 and output $\mathbf{x}^k$ with $\|\mathbf{v}^k\| \leq 2\tilde{\epsilon}$. For $\boldsymbol{\mathcal{H}}_3^{K_0}$ happens, we have $\|\nabla f(\mathbf{x}^k) - \mathbf{v}^k\| \leq \sqrt{\epsilon \cdot \tilde{\epsilon}} \leq \tilde{\epsilon}$. So $\|\nabla f(\mathbf{x}^k)\| \leq 3\tilde{\epsilon}$. Set $k_0 = \lfloor k/\mathscr{K}\rfloor\mathscr{K}$. Since the NC-search conducts successfully, from Theorem 9, we have $\lambda_{\min}\left(\nabla f^2(\mathbf{x}^{k_0})\right) \geq -2\delta I$. From Assumption 2, we have

$$\left\|\nabla f^2(\mathbf{x}) - \nabla f^2(\mathbf{y})\right\|_2 \leq \left\|\frac{1}{n}\sum_{i=1}^n \nabla f_i^2(\mathbf{x}) - \nabla f_i^2(\mathbf{y})\right\|_2 \leq \frac{1}{n}\sum_{i=1}^n \left\|\nabla f_i^2(\mathbf{x}) - \nabla f_i^2(\mathbf{y})\right\|_2 \leq \rho\|\mathbf{x} - \mathbf{y}\|. \tag{B.34}$$

So $f(\cdot)$ has ρ-Lipschitz Hessian. We have

$$
\begin{aligned}
& \left\|\nabla f^2(\mathbf{x}^k) - \nabla f^2(\mathbf{x}^{k_0})\right\|_2 \\
\leq\ & \left\|\sum_{i=k_0}^{k-1} \left(\nabla f^2(\mathbf{x}^{i+1}) - \nabla f^2(\mathbf{x}^i)\right)\right\|_2 \\
\leq\ & \sum_{i=k_0}^{k-1} \rho \left\|\mathbf{x}^{i+1} - \mathbf{x}^i\right\|_2 \leq \mathscr{K} \frac{\rho\epsilon}{Ln_0} \overset{\mathscr{K}=\frac{\delta L n_0}{\rho\epsilon}}{=} \delta.
\end{aligned} \tag{B.35}
$$

Thus $\lambda_{\min}(\nabla f^2(\mathbf{x}^k)) \geq -3\delta I$. □

Now, we are ready to prove Theorem 6.

Proof of Theorem 6. For all iteration $\mathcal{K}$ with $\mathrm{mod}(\mathcal{K}, \mathscr{K}) = 0$, given $\boldsymbol{\mathcal{F}}^{\mathcal{K}}$, we consider the case when $\mathcal{I}_{\mathcal{K}} = 2$ and $\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}$ happens. Because $f(\cdot)$ has ρ-Lipschitz Hessian, we have

$$
\begin{aligned}
& f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right) \\
\leq\ & f\left(\mathbf{x}^{\mathcal{K}}\right) + \left[\nabla f\left(\mathbf{x}^{\mathcal{K}}\right)\right]^{\mathrm{T}}\left[\mathbf{x}^{\mathcal{K}+\mathscr{K}} - \mathbf{x}^{\mathcal{K}}\right] + \frac{1}{2}\left[\mathbf{x}^{\mathcal{K}+\mathscr{K}} - \mathbf{x}^{\mathcal{K}}\right]^{\mathrm{T}}\left[\nabla^2 f\left(\mathbf{x}^{\mathcal{K}}\right)\right]\left[\mathbf{x}^{\mathcal{K}+\mathscr{K}} - \mathbf{x}^{\mathcal{K}}\right] \\
& + \frac{\rho}{6}\left\|\left[\mathbf{x}^{\mathcal{K}+\mathscr{K}} - \mathbf{x}^{\mathcal{K}}\right]\right\|^3.
\end{aligned} \tag{B.36}
$$

Because $\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}$ happens, and $\mathcal{I}_{\mathcal{K}} = 2$, we have $\mathbf{w}_1^{\mathrm{T}}\left[\nabla f^2(\mathbf{x}^{\mathcal{K}})\right]\mathbf{w}_1 \leq -\delta$, and by taking expectation on the random number of the sign, we have

$$
\mathbb{E}\left[\left[\nabla f\left(\mathbf{x}^{\mathcal{K}}\right)\right]^{\mathrm{T}}\left[\mathbf{x}^{\mathcal{K}+\mathscr{K}} - \mathbf{x}^{\mathcal{K}}\right] \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right] = 0,
$$

thus

$$
\mathbb{E}\left[f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right) \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right] \leq f\left(\mathbf{x}^{\mathcal{K}}\right) - \frac{\delta^3}{2\rho^2} + \frac{\delta^3}{6\rho^2} = f\left(\mathbf{x}^{\mathcal{K}}\right) - \frac{\delta^3}{3\rho^2}. \tag{B.37}
$$

Furthermore, by analyzing the difference of $\left(f\left(\mathbf{x}^{\mathcal{K}}\right) - f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}}$, where $\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}}$ is the indication function for the event $\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}$, we have

$$
\begin{aligned}
& \mathbb{E}\left[\left(f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right) - f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}} \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}} - \left(f\left(\mathbf{x}^{\mathcal{K}}\right) - f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}} \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right] \\
=\ & \mathbb{E}\left[\left(f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right) - f^*\right)\left(\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}} - \mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}}\right) \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right] + \mathbb{E}\left[\left(f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right) - f\left(\mathbf{x}^{\mathcal{K}}\right)\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}} \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right] \\
\overset{a}{\leq}\ & \mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^{\mathcal{K}} \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)\mathbb{E}\left[f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right) - f\left(\mathbf{x}^{\mathcal{K}}\right) \mid \boldsymbol{\mathcal{H}}_4^{\mathcal{K}} \cap \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right] \\
\overset{(B.37)}{\leq}\ & -\mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^{\mathcal{K}} \mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)\frac{\delta^3}{3\rho^2},
\end{aligned} \tag{B.38}
$$

where in $\overset{a}{\leq}$, we use that $\boldsymbol{\mathcal{H}}_4^{\mathcal{K}} \supseteq \boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}$, so $\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}} - \mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}} \leq 0$ and $f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right) - f^* \geq 0$, then

$\mathbb{E}\left[\left(f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right)-f^*\right)\left(\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}}-\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}}\right)\mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\right]\leq 0$.

On the other hand, given $\boldsymbol{\mathcal{F}}^{\mathcal{K}}$, we consider the case when $\mathcal{I}_{\mathcal{K}}=1$ and $\boldsymbol{\mathcal{H}}_4^{\mathcal{K}}$ happens, then for any k satisfying $\mathcal{K}\leq k<\mathcal{K}+\mathscr{K}$, we know $\mathcal{I}_k=1$.

Given $\boldsymbol{\mathcal{F}}^k$ with $\mathcal{K}\leq k<\mathcal{K}+\mathscr{K}$, then from (B.26) we have

$$f\left(\mathbf{x}^{k+1}\right)\leq f\left(\mathbf{x}^k\right)-\eta^k\left(\frac{1}{2}-\frac{\eta^kL}{2}\right)\left\|\mathbf{v}^k\right\|^2+\frac{\eta^k}{2}\left\|\mathbf{v}^k-\nabla f\left(\mathbf{x}^k\right)\right\|^2, \tag{B.39}$$

with $\eta^k=\eta/\left\|\mathbf{v}^k\right\|$. Also $\boldsymbol{\mathcal{H}}_4^K$ is measurable on $\boldsymbol{\mathcal{F}}^k$, and if $\boldsymbol{\mathcal{H}}_4^K$ happens, we have $\left\|\mathbf{v}^k\right\|\geq 2\tilde{\epsilon}$, and $\left\|\mathbf{v}^k-\nabla f\left(\mathbf{x}^k\right)\right\|^2\leq\epsilon\cdot\tilde{\epsilon}$, then from (B.27) and (B.28), we have

$$f\left(\mathbf{x}^{k+1}\right)\leq f\left(\mathbf{x}^k\right)-\frac{\epsilon\cdot\tilde{\epsilon}}{4Ln_0}. \tag{B.40}$$

Taking expectation up to $\boldsymbol{\mathcal{F}}^{\mathcal{K}}$, we have

$$\mathbb{E}\left[f\left(\mathbf{x}^{k+1}\right)-f\left(\mathbf{x}^k\right)\mid \boldsymbol{\mathcal{F}}^{\mathcal{K}}\cap\boldsymbol{\mathcal{H}}_4^k\right]\leq-\frac{\epsilon\cdot\tilde{\epsilon}}{4Ln_0}. \tag{B.41}$$

By analyzing the difference of $\left(f\left(\mathbf{x}^k\right)-f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^k}$, we have

$$\begin{aligned}
& \mathbb{E}\left[\left(f\left(\mathbf{x}^{k+1}\right)-f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{k+1}}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}-\left(f\left(\mathbf{x}^k\right)-f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^k}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right]\\
=\ & \mathbb{E}\left[\left(f\left(\mathbf{x}^{k+1}\right)-f^*\right)\left(\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{k+1}}-\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^k}\right)\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right]+\mathbb{E}\left[\left(f\left(\mathbf{x}^{k+1}\right)-f\left(\mathbf{x}^k\right)\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^k}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right]\\
\overset{a}{\leq}\ & \mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^k\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)\mathbb{E}\left[f\left(\mathbf{x}^{k+1}\right)-f\left(\mathbf{x}^k\right)\mid\boldsymbol{\mathcal{H}}_4^k\cap\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right]\\
\leq\ & -\mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^k\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)\frac{\epsilon\cdot\tilde{\epsilon}}{4Ln_0},
\end{aligned} \tag{B.42}$$

where in $\overset{a}{\leq}$, we use $\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{k+1}}-\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^k}\leq 0$ and $f\left(\mathbf{x}^{k+1}\right)-f^*\geq 0$.

By telescoping (B.42) with k from $\mathcal{K}$ to $\mathcal{K}+\mathscr{K}-1$, we have

$$\begin{aligned}
& \mathbb{E}\left[\left(f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right)-f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}-\left(f\left(\mathbf{x}^{\mathcal{K}}\right)-f^*\right)\mathbb{1}_{\boldsymbol{\mathcal{H}}_4^k}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right]\\
\leq\ & -\frac{\epsilon\cdot\tilde{\epsilon}}{4Ln_0}\sum_{i=\mathcal{K}}^{\mathcal{K}+\mathscr{K}}\mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^i\mid\boldsymbol{\mathcal{F}}^k\right)\\
\overset{a}{\leq}\ & -\mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)\frac{\mathscr{K}\epsilon^2}{4Ln_0}\overset{\mathscr{K}=\frac{\delta Ln_0}{\rho\epsilon}}{=}-\mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)\frac{\delta\epsilon}{4\rho}.
\end{aligned} \tag{B.43}$$

where in $\overset{a}{\leq}$, we use $\boldsymbol{\mathcal{H}}_4^i\supseteq\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}$ with $\mathcal{K}\leq i\leq\mathcal{K}+\mathscr{K}$, and then $\mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^i\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)\geq\mathbb{P}\left(\boldsymbol{\mathcal{H}}_4^{\mathcal{K}+\mathscr{K}}\mid\boldsymbol{\mathcal{F}}^{\mathcal{K}}\right)$.

Combining (B.38) and (B.43), using $\mathbb{P}\left(\mathcal{H}_4^{\mathcal{K}} \mid \mathcal{F}^k\right) \geq \mathbb{P}\left(\mathcal{H}_4^{\mathcal{K}+\mathscr{K}} \mid \mathcal{F}^{\mathcal{K}}\right)$, we have

$$\begin{aligned}
&\mathbb{E}\left[\left(f\left(\mathbf{x}^{\mathcal{K}+\mathscr{K}}\right)-f^*\right)\mathbb{1}_{\mathcal{H}_4^{\mathcal{K}+\mathscr{K}}} \mid \mathcal{F}^{\mathcal{K}} - \left(f\left(\mathbf{x}^{\mathcal{K}}\right)-f^*\right)\mathbb{1}_{\mathcal{H}_4^k} \mid \mathcal{F}^{\mathcal{K}}\right] \\
\leq\; & -\mathbb{P}\left(\mathcal{H}_4^{\mathcal{K}+\mathscr{K}} \mid \mathcal{F}^{\mathcal{K}}\right)\min\left(\frac{\delta\tilde{\epsilon}}{4\rho}, \frac{\delta^3}{3\rho^2}\right).
\end{aligned} \tag{B.44}$$

By taking full expectation on (B.44), and telescoping the results with $\mathcal{K}=0,\mathscr{K},\cdots,(J-1)\mathscr{K}$, and using $\mathbb{P}\left(\mathcal{H}_4^{\mathscr{K}j}\right) \leq \mathbb{P}\left(\mathcal{H}_4^{J\mathscr{K}}\right)$ with $j=1,\cdots,J$, we have

$$\begin{aligned}
&\mathbb{E}\left[\left(f\left(\mathbf{x}^{J\mathscr{K}}\right)-f^*\right)\mathbb{1}_{\mathcal{H}_4^{J\mathscr{K}}} - \left(f\left(\mathbf{x}^0\right)-f^*\right)\mathbb{1}_{\mathcal{H}_4^0}\right] \\
\leq\; & -\mathbb{P}\left(\mathcal{H}_4^{J\mathscr{K}}\right)\min\left(\frac{\delta\tilde{\epsilon}}{4\rho}, \frac{\delta^3}{3\rho^2}\right)J.
\end{aligned} \tag{B.45}$$

Substituting the inequalities $f\left(\mathbf{x}^{J\mathscr{K}}\right)-f^* \geq 0$, $f\left(\mathbf{x}^0\right)-f^* \leq \Delta$, and $J = 4\max\left(\left\lfloor \frac{3\rho^2\Delta}{\delta^3}, \frac{4\Delta\rho}{\delta\epsilon}\right\rfloor\right)+4 \geq \frac{4\Delta}{\min\left(\frac{\delta\tilde{\epsilon}}{4\rho}, \frac{\delta^3}{3\rho^2}\right)}$ into (B.45), we have

$$\mathbb{P}\left(\mathcal{H}_4^{K_0}\right) \leq \frac{1}{4}. \tag{B.46}$$

Then by union bound, we have

$$\mathbb{P}\left(\mathcal{H}_1^{K_0}\right) = \mathbb{P}\left(\mathcal{H}_1^{K_0}\bigcap\mathcal{H}_3^{K_0}\right) + \mathbb{P}\left(\mathcal{H}_1^{K_0}\bigcap\left(\mathcal{H}_3^{K_0}\right)^c\right) \leq \mathbb{P}\left(\mathcal{H}_4^{K_0}\right) + \mathbb{P}\left(\left(\mathcal{H}_3^{K_0}\right)^c\right) \overset{\text{Lemma 8}}{\leq} \frac{1}{4}+\frac{1}{8}. \tag{B.47}$$

Then our proof is completed by obtaining

$$\mathbb{P}\left(\left(\mathcal{H}_1^{K_0}\right)^c\bigcap\mathcal{H}_3^{K_0}\right) = 1-\mathbb{P}\left(\left(\mathcal{H}_1^{K_0}\right)\bigcup\left(\mathcal{H}_3^{K_0}\right)^c\right) \geq 1-\mathbb{P}\left(\left(\mathcal{H}_1^{K_0}\right)\right)-\mathbb{P}\left(\left(\mathcal{H}_3^{K_0}\right)^c\right) \overset{\text{Lemma 8}}{\geq} \frac{1}{2}. \tag{B.48}$$

From Lemma 9, with probability $1/2$, the algorithm shall be terminated before $\mathscr{K}J$ iterations, and output a $\mathbf{x}^k$ satisfying (3.12).

The total stochastic gradient complexity consists of two parts: the Spider maintenance cost and NC-Search cost. We estimate them as follows:

1. With probability $1/2$, the algorithm ends in at most K_0 iterations, thus the number of stochastic gradient accesses to maintain Spider can be bounded by

$$\begin{aligned}
\lfloor K_0/q\rfloor qS_1 + S_2 &\overset{S_1=qS_2}{\leq} 2K_0S_2+S_1 \\
&\leq \left(4\max\left\lfloor\frac{3\rho^2\Delta}{\delta^3}, \frac{4\Delta\rho}{\epsilon\delta}\right\rfloor+4\right)(2S_2\mathscr{K})+S_1.
\end{aligned} \tag{B.49}$$

2. With probability $1/2$, the algorithm ends in K_0 iterations, thus there are at most J times of NC search. From Theorem 9, suppose the NC-search costs $\tilde{C}L^2\delta^{-2}$, where $\tilde{C}$ hides a

polylogarithmic factor of d. The stochastic gradient access for NC-Search is less than:

$$J L^2 \tilde{C} \delta^{-2} = \left(4 \max\left\lfloor \frac{3\rho^2\Delta}{\delta^3}, \frac{4\Delta\rho}{\epsilon\delta} \right\rfloor + 4\right) \tilde{C} L^2 \delta^{-2}, \tag{B.50}$$

By summing (B.49) and (B.50), using $\max\lfloor a, b \rfloor \leq a + b$ with $a \geq 0$ and $b \geq 0$, we have that the total stochastic gradient complexity can be bounded:

$$4\left(\frac{3\rho^2\Delta}{\delta^3} + \frac{4\Delta\rho}{\epsilon\delta} + 2\right)\left(2S_2\mathscr{K} + \tilde{C}L^2\delta^{-2}\right) + S_1.$$

For the on-line case, plugging into $\mathscr{K} = \frac{\delta L n_0}{\rho\epsilon}$, $S_1 = \frac{2\sigma}{\epsilon^2}$, and $S_2 = \frac{2\sigma}{n_0\epsilon}$, the stochastic gradient complexity can be bounded:

$$\frac{64\Delta L\sigma}{\epsilon^3} + \frac{48\Delta\sigma L\rho}{\epsilon^2\delta^2} + \frac{12\tilde{C}\Delta L^2\rho^2}{\delta^5} + \frac{16\tilde{C}\Delta L^2\rho}{\epsilon\delta^3} + \frac{2\sigma^2}{\epsilon^2} + \frac{8\tilde{C}L^2}{\delta^2} + \frac{32L\sigma\delta}{\rho\epsilon^2}.$$

For the off-line case, plugging into $S_2 = \frac{n^{1/2}}{n_0}$, we obtain the stochastic gradient complexity is bounded by:

$$\frac{32\Delta L n^{1/2}}{\epsilon^2} + \frac{12\Delta\rho L n^{1/2}}{\epsilon\delta^2} + \frac{12\tilde{C}\Delta L^2\rho^2}{\delta^5} + \frac{16\tilde{C}\Delta L^2\rho}{\epsilon\delta^3} + n + \frac{8\tilde{C}L^2}{\delta^2} + \frac{16Ln^{1/2}\delta}{\rho\epsilon}.$$

□

B.5 Proof for SZO

Proof of Lemma 3. We have that

$$\begin{aligned}
& \mathbb{E}_{i,\mathbf{u}} \left\| \left[\frac{f_i(\mathbf{x}+\mu\mathbf{u}) - f_i(\mathbf{x})}{\mu}\mathbf{u} - \left(\frac{f_i(\mathbf{y}+\mu\mathbf{u}) - f_i(\mathbf{y})}{\mu}\mathbf{u} \right) \right] \right\|^2 \\
= \; & \mathbb{E}_{i,\mathbf{u}} \Bigg\| \langle \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}), \mathbf{u} \rangle \, \mathbf{u} + \frac{f_i(\mathbf{x}+\mu\mathbf{u}) - f_i(\mathbf{x}) - \langle \nabla f_i(\mathbf{x}), \mu\mathbf{u}\rangle}{\mu}\mathbf{u} \\
& - \left(\frac{f_i(\mathbf{y}+\mu\mathbf{u}) - f_i(\mathbf{y}) - \langle \nabla f_i(\mathbf{y}), \mu\mathbf{u}\rangle}{\mu}\mathbf{u} \right) \Bigg\|^2 \\
\leq \; & 2\mathbb{E}_{i,\mathbf{u}} \left\| \langle \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}), \mathbf{u} \rangle \, \mathbf{u} \right\|^2 \\
& + 2\mathbb{E}_{i,\mathbf{u}} \left\| \frac{f_i(\mathbf{x}+\mu\mathbf{u}) - f_i(\mathbf{x}) - \langle \nabla f_i(\mathbf{x}), \mu\mathbf{u}\rangle}{\mu}\mathbf{u} - \left(\frac{f_i(\mathbf{y}+\mu\mathbf{u}) - f_i(\mathbf{y}) - \langle \nabla f_i(\mathbf{y}), \mu\mathbf{u}\rangle}{\mu}\mathbf{u} \right) \right\|^2 \\
\leq \; & 2\mathbb{E}_{i,\mathbf{u}} \left\| \langle \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}), \mathbf{u} \rangle \, \mathbf{u} \right\|^2 + 4\mathbb{E}_{i,\mathbf{u}} \left(\left| \frac{f_i(\mathbf{x}+\mu\mathbf{u}) - f_i(\mathbf{x}) - \langle \nabla f_i(\mathbf{x}), \mu\mathbf{u}\rangle}{\mu} \right|^2 \|\mathbf{u}\|^2 \right) \\
& + 4\mathbb{E}_{i,\mathbf{u}} \left(\left| \frac{f_i(\mathbf{y}+\mu\mathbf{u}) - f_i(\mathbf{y}) - \langle \nabla f_i(\mathbf{y}), \mu\mathbf{u}\rangle}{\mu} \right|^2 \|\mathbf{u}\|^2 \right) \\
\overset{a}{\leq} \; & 2\mathbb{E}_{i,\mathbf{u}} \left\| \langle \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}), \mathbf{u} \rangle \, \mathbf{u} \right\|^2 + 8\frac{\mu^2 L^2}{4}\mathbb{E}_{\mathbf{u}}\|\mathbf{u}\|^6 \\
\overset{b}{\leq} \; & 2(d+4)\mathbb{E}_i \left\| \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}) \right\|^2 + 2\mu^2 L^2 \mathbb{E}_{\mathbf{u}}\|\mathbf{u}\|^6 \\
\leq \; & 2(d+4)\mathbb{E}_i \left\| \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}) \right\|^2 + 2\mu^2 L^2 \mathbb{E}_{\mathbf{u}}\|\mathbf{u}\|^6 \\
\overset{c}{\leq} \; & 2(d+4)\mathbb{E}_i \left\| \nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y}) \right\|^2 + 2\mu^2 (d+6)^3 L^2 \\
\leq \; & 2(d+4)L^2\|\mathbf{x}-\mathbf{y}\|^2 + 2\mu^2(d+6)^3L^2,
\end{aligned} \tag{B.51}$$

where in $\overset{a}{\leq}$ we use

$$|f_i(\mathbf{a}) - f_i(\mathbf{b}) - \langle \nabla f_i(\mathbf{a}), \mathbf{a}-\mathbf{b}\rangle\,| \leq \frac{L}{2}\|\mathbf{a}-\mathbf{b}\|^2,$$

because f_i has L-Lipschitz continuous gradient ((6) in (Nesterov & Spokoiny, 2011)); $\overset{b}{\leq}$, we use

$$\mathbb{E}_{\mathbf{u}}\|\, \langle \mathbf{a}, \mathbf{u}\rangle \, \mathbf{u}\|^2 \leq (d+4)\|\mathbf{a}\|^2;$$

obtained from the same technique of (33) in (Nesterov & Spokoiny, 2011); in $\overset{c}{=}$, $\mathbb{E}_{\mathbf{u}}\|\mathbf{u}\|^6 \leq (d+6)^3$ in (17) of (Nesterov & Spokoiny, 2011). □

Lemma 10. *Under the Assumption 2 (including (ii')), if $\lfloor k/q \rfloor q = k$, given $\mathbf{x}^k$, we have*

$$\mathbb{E}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|^2 \leq \frac{\epsilon^2}{4}. \tag{B.52}$$

Proof. Let $\mathbb{E}_k$ denote that the expectation is taken on the random number at iteration k given the full information of $\mathbf{x}_{0:k}$. Denote $\nabla_j f(\mathbf{x})$ as the value in the j-th coordinate of $\nabla f(\mathbf{x})$, we have that

$$
\begin{aligned}
& \mathbb{E}_k\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \\
= \;& \mathbb{E}_k \sum_{j\in[d]} \left|\mathbf{v}_j^k - \nabla_j f(\mathbf{x}^k)\right|^2 \\
= \;& \sum_{j\in[d]} \mathbb{E}_k \left|\frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'} \frac{f_i(\mathbf{x}^k+\mu\mathbf{e}_j) - f_i(\mathbf{x}^k)}{\mu} - \nabla_j f(\mathbf{x}^k)\right|^2 \\
\leq \;& 2\sum_{j\in[d]} \mathbb{E}_k \left|\frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'} \frac{f_i(\mathbf{x}^k+\mu\mathbf{e}_j) - f_i(\mathbf{x}^k)}{\mu} - \frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'}\nabla_j f_i(\mathbf{x}^k)\right|^2 + 2\sum_{j\in[d]} \mathbb{E}_k \left|\frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'}\nabla_j f_i(\mathbf{x}^k) - \nabla_j f(\mathbf{x}^k)\right|^2 \\
\overset{a}{\leq} \;& \frac{2}{S_1'}\sum_{j\in[d]} \mathbb{E}_k \left|\frac{f_i(\mathbf{x}^k+\mu\mathbf{e}_j) - f_i(\mathbf{x}^k)}{\mu} - \nabla_j f_i(\mathbf{x}^k)\right|^2 + 2\sum_{j\in[d]} \mathbb{E}_k \left|\frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'}\nabla_j f_i(\mathbf{x}^k) - \nabla_j f(\mathbf{x}^k)\right|^2, \qquad \text{(B.53)}
\end{aligned}
$$

where in $\overset{a}{\leq}$, we use $|a_1 + a_2 + \cdots + a_s|^2 \leq s|a_1|^2 + s|a_2|^2 + \cdots + s|a_s|^2$.

For the first term in the right hand of (B.53), because $f_i(\mathbf{x})$ has L-Lipschitz continuous gradient, we have

$$
\begin{aligned}
& \left|\frac{f_i(\mathbf{x}^k+\mu\mathbf{e}_j) - f_i(\mathbf{x}^k)}{\mu} - \nabla_j f_i(\mathbf{x}^k)\right| \\
= \;& \frac{1}{\mu}\left|f_i(\mathbf{x}^k+\mu\mathbf{e}_j) - f_i(\mathbf{x}^k) - \left\langle \nabla f_i(\mathbf{x}^k), \mu\mathbf{e}_j\right\rangle\right| \leq \frac{1}{\mu}\frac{L}{2}\|\mu\mathbf{e}_j\|^2 = \frac{L\mu}{2}. \qquad \text{(B.54)}
\end{aligned}
$$

Thus we have

$$
\begin{aligned}
& \mathbb{E}_k\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \qquad \text{(B.55)} \\
\leq \;& \frac{dL^2\mu^2}{2} + 2\sum_{j\in[d]} \mathbb{E}_k \left|\frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'}\nabla_j f_i(\mathbf{x}^k) - \nabla_j f(\mathbf{x}^k)\right|^2 \\
= \;& \frac{dL^2\mu^2}{2} + 2\mathbb{E}_k \left\|\frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'}\nabla f_i(\mathbf{x}^k) - \nabla f(\mathbf{x}^k)\right\|^2 \qquad \text{(B.56)}
\end{aligned}
$$

For the on-line case, due to $\mu \leq \frac{\epsilon}{2\sqrt{6}L\sqrt{d}}$, and $S_1 = \frac{96d\sigma^2}{\epsilon^2}$, we have

$$
\mathbb{E}_k\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \frac{dL^2\epsilon^2}{48L^2 d} + \frac{2}{S_1'}\sigma^2 \leq \frac{\epsilon^2}{24}. \qquad \text{(B.57)}
$$

In finite-sum case, we have $\mathbb{E}_k\left\|\frac{1}{S_1'}\sum_{i\in\mathcal{S}_1'}\nabla f_i(\mathbf{x}^k) - \nabla f(\mathbf{x}^k)\right\|^2 = 0$, so $\mathbb{E}_k\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq \frac{\epsilon^2}{24}$.

Also from (4.3), and $\mu \le \frac{\epsilon}{\sqrt{6}n_0 L(d+6)^{3/2}}$, we have

$$\left\|\nabla f(\mathbf{x}^k) - \nabla \hat{f}(\mathbf{x}^k)\right\|^2 \le \frac{\mu^2 L^2 (d+3)^3}{4} \le \frac{\epsilon^2}{6L^2(d+6)^3}\frac{L^2(d+3)^3}{4} \le \frac{\epsilon^2}{24}. \tag{B.58}$$

We have

$$\mathbb{E}_k\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|^2 \le 2\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 + 2\|\nabla f(\mathbf{x}^k) - \nabla \hat{f}(\mathbf{x}^k)\|^2 \le \frac{\epsilon^2}{6}. \tag{B.59}$$

□

Lemma 11. *From the setting of Theorem 8, and under the Assumption 2 (including (ii')), for $k_0 = \lfloor k/q \rfloor \cdot q$, we have $\mathbb{E}_{k_0}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\| \le \epsilon^2$.*

Proof. For $k = k_0$, from Lemma 10, we obtain the result. When $k \ge k_0$, from Lemma 3, we have that

$$\begin{aligned}
&\mathbb{E}_{\mathcal{S}_2}\left\|\frac{1}{S_2}\sum_{(i,\mathbf{u})\in\mathcal{S}_2}\left(\frac{f_i(\mathbf{x}^k+\mu\mathbf{u}^k)-f_i(\mathbf{x}^k)}{\mu}\mathbf{u} - \frac{f_i(\mathbf{x}^{k-1}+\mu\mathbf{u}^{k-1})-f_i(\mathbf{x}^{k-1})}{\mu}\mathbf{u}\right) - (\hat{f}(\mathbf{x}^k)-\hat{f}(\mathbf{x}^{k-1}))\right\|^2 \\
= \quad & \frac{1}{S_2}\mathbb{E}_{i,\mathbf{u}}\left\|\left(\frac{f_i(\mathbf{x}^k+\mu\mathbf{u}^k)-f_i(\mathbf{x}^k)}{\mu}\mathbf{u} - \frac{f_i(\mathbf{x}^{k-1}+\mu\mathbf{u}^{k-1})-f_i(\mathbf{x}^{k-1})}{\mu}\mathbf{u}\right) - (\hat{f}(\mathbf{x}^k)-\hat{f}(\mathbf{x}^{k-1}))\right\|^2 \\
\overset{(4.7)}{\le} \quad & \frac{1}{S_2}\left(2(d+4)L^2\|\mathbf{x}^k-\mathbf{x}^{k-1}\|^2 + 2\mu^2(d+6)^3L^2\right) \\
\le \quad & \frac{1}{S_2}\left(2(d+4)L^2\|\eta^k\mathbf{v}^k\|^2 + 2\mu^2(d+6)^3L^2\right) \\
\overset{\eta^k\le\frac{\epsilon}{Ln_0\|\mathbf{v}^k\|}}{\le} \quad & \frac{1}{S_2}\left(2(d+4)L^2\frac{\epsilon^2}{L^2n_0^2} + 2(d+6)^3L^2\frac{\epsilon^2}{6n_0^2L^2(d+6)^3}\right) \\
= \quad & \frac{1}{S_2}\left((2d+9)\frac{\epsilon^2}{n_0^2}\right).
\end{aligned} \tag{B.60}$$

Using Proposition 1, for on-line case, we have

$$\mathbb{E}_{k_0}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|^2 \overset{S_2=\frac{30(2d+9)\sigma}{\epsilon n_0}}{\le} \frac{\epsilon^2}{6} + \sum_{j=k_0}^{k}\frac{\epsilon^3}{30n_0\sigma} \overset{q=\frac{5n_0\sigma}{\epsilon}}{\le} \frac{\epsilon^2}{3}. \tag{B.61}$$

for finite-sum case, we have

$$\mathbb{E}_{k_0}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|^2 \overset{S_2=\frac{(2d+9)n^{1/2}}{n_0}}{\le} \frac{\epsilon^2}{6} + \sum_{j=k_0}^{k}\frac{\epsilon^2}{n_0 n^{1/2}} \overset{q=\frac{n_0 n^{1/2}}{6}}{\le} \frac{\epsilon^2}{3}. \tag{B.62}$$

□

Proof of Theorem 8. By taking full expectation on Lemma 11, we have

$$\mathbb{E}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|^2 \leq \frac{\epsilon^2}{3}. \tag{B.63}$$

Thus

$$\mathbb{E}_k\|\mathbf{v}^k - \nabla f(\mathbf{x}^k)\|^2 \leq 2\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|^2 + 2\|\nabla f(\mathbf{x}^k) - \nabla \hat{f}(\mathbf{x}^k)\|^2 \overset{(B.58)}{\leq} \epsilon^2. \tag{B.64}$$

By using Lemma 4, (B.13), and (B.14), we have

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\mathbf{v}^k\| \leq \Delta \cdot \frac{4Ln_0}{\epsilon}\frac{1}{K} + 3\epsilon \leq 4\epsilon. \tag{B.65}$$

One the other hand, by Jensen's inequality, we have

$$(\mathbb{E}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|)^2 = \mathbb{E}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\|^2 - \mathbb{E}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k) - \mathbb{E}(\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k))\|^2 \leq \epsilon^2.$$

So

$$\begin{aligned}
& \mathbb{E}\|\nabla f(\mathbf{x}^k)\| \\
= \;& \mathbb{E}\|\mathbf{v}^k - (\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)) + \nabla \hat{f}(\mathbf{x}^k) - \nabla f(\mathbf{x}^k)\| \\
\leq \;& \mathbb{E}\|\mathbf{v}^k\| + \mathbb{E}\|\mathbf{v}^k - \nabla \hat{f}(\mathbf{x}^k)\| + \mathbb{E}\|\nabla \hat{f}(\mathbf{x}^k) - \nabla f(\mathbf{x}^k)\| \\
\overset{(B.58)}{\leq} \;& \mathbb{E}\|\mathbf{v}^k\| + \epsilon + \frac{\epsilon}{2\sqrt{6}} \leq \mathbb{E}\|\mathbf{v}^k\| + 2\epsilon.
\end{aligned} \tag{B.66}$$

We have

$$\mathbb{E}\|\nabla f(\tilde{\mathbf{x}})\| = \frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\nabla f(\mathbf{x}^k)\| \overset{(B.66)}{\leq} \frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\mathbf{v}^k\| + 2\epsilon \overset{(B.65)}{\leq} 6\epsilon. \tag{B.67}$$

□

B.6 Proof of Theorem 3 for Lower Bound

Our proof is a direct extension of Carmon et al. (2017b). Before we drill into the proof of Theorem 3, we first introduce the hard instance $\tilde{f}_K$ with $K \geq 1$ constructed by Carmon et al. (2017b).

$$\hat{f}_K(\mathbf{x}) := -\Psi(1)\Phi(x_1) + \sum_{i=2}^{K}\left[\Psi(-x_{i-1})\Phi(-x_i) - \Psi(x_{i-1})\Phi(x_i)\right], \tag{B.68}$$

where the component functions are

$$\Psi(x) := \begin{cases} 0 & x \le \frac{1}{2} \\ \exp\left(1 - \frac{1}{(2x-1)^2}\right) & x > \frac{1}{2} \end{cases} \tag{B.69}$$

and

$$\Phi(x) := \sqrt{e}\int_{-\infty}^{x} e^{-\frac{t^2}{2}}, \tag{B.70}$$

where x_i denote the value of i-th coordinate of $\mathbf{x}$, with $i \in [d]$. $\hat{f}_K(\mathbf{x})$ constructed by Carmon et al. (2017b) is a zero-chain function, that is for every $i \in [d]$, $\nabla_i f(\mathbf{x}) = 0$ whenever $x_{i-1} = x_i = x_{i+1}$. So any deterministic algorithm can only recover "one" dimension in each iteration (Carmon et al., 2017b). In addition, it satisfies that : If $|x_i| \le 1$ for any $i \le K$,

$$\left\|\nabla \hat{f}_K(\mathbf{x})\right\| \ge 1. \tag{B.71}$$

Then to handle random algorithms, Carmon et al. (2017b) further consider the following extensions:

$$\tilde{f}_{K,\mathbf{B}^K}(\mathbf{x}) = \hat{f}_K\left((\mathbf{B}^K)^{\mathrm{T}}\rho(\mathbf{x})\right) + \frac{1}{10}\|\mathbf{x}\|^2 = \hat{f}_K\left(\left\langle \mathbf{b}^{(1)}, \rho(\mathbf{x})\right\rangle, \ldots, \left\langle \mathbf{b}^{(K)}, \rho(\mathbf{x})\right\rangle\right) + \frac{1}{10}\|\mathbf{x}\|^2, \tag{B.72}$$

where $\rho(\mathbf{x}) = \frac{\mathbf{x}}{\sqrt{1+\|\mathbf{x}\|^2/R^2}}$ and $R = 230\sqrt{K}$, $\mathbf{B}^K$ is chosen uniformly at random from the space of orthogonal matrices $\mathcal{O}(d, K) = \{\mathbf{D} \in \mathbb{R}^{d\times K} | \mathbf{D}^\top\mathbf{D} = I_K\ \}$. The function $\tilde{f}_{K,\mathbf{B}}(\mathbf{x})$ satisfies the following:

(i)

$$\tilde{f}_{K,\mathbf{B}^K}(\mathbf{0}) - \inf_{\mathbf{x}} \tilde{f}_{K,\mathbf{B}^K}(\mathbf{x}) \le 12K. \tag{B.73}$$

(ii) $\tilde{f}_{K,\mathbf{B}^K}(\mathbf{x})$ has constant l (independent of K and d) Lipschitz continuous gradient.

(iii) if $d \ge 52{\cdot}230^2K^2\log(\frac{2K^2}{p})$, for any algorithm $\mathcal{A}$ solving (1.2) with $n = 1$, and $f(\mathbf{x}) = \tilde{f}_{K,\mathbf{B}^K}(\mathbf{x})$, then with probability $1 - p$,

$$\left\|\nabla \tilde{f}_{K,\mathbf{B}^K}(\mathbf{x}^k)\right\| \ge \frac{1}{2}, \quad \text{for every } k \le K. \tag{B.74}$$

The above properties found by Carmon et al. (2017b) is very technical. One can refer to Carmon et al. (2017b) for more details.

Proof of Theorem 3. Our lower bound theorem proof is as follows. The proof mirrors Theorem 2 in Carmon et al. (2017b) by further taking the number of individual function n into account. Set

$$f_i(\mathbf{x}) := \frac{ln^{1/2}\epsilon^2}{L}\tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b) = \frac{ln^{1/2}\epsilon^2}{L}\left(\hat{f}_K\left((\mathbf{B}_i^K)^{\mathrm{T}}\rho(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b)\right) + \frac{1}{10}\left\|\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b\right\|^2\right), \tag{B.75}$$

and

$$f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}). \tag{B.76}$$

where $\mathbf{B}^{nK} = [\mathbf{B}_1^K, \ldots, \mathbf{B}_n^K]$ is chosen uniformly at random from the space of orthogonal matrices $\mathcal{O}(d, K) = \{\mathbf{D} \in \mathbb{R}^{(d/n)\times(nK)} | \mathbf{D}^\top\mathbf{D} = I_{(nK)}\ \}$, with each $\mathbf{B}_i^K \in \{\mathbf{D} \in \mathbb{R}^{(d/n)\times(K)} | \mathbf{D}^\top\mathbf{D} = I_{(K)}\ \}$, $i \in [n]$, $\mathbf{C} = [\mathbf{C}_1, \ldots, \mathbf{C}_n]$ is an arbitrary orthogonal matrices $\mathcal{O}(d, K) = \{\mathbf{D} \in \mathbb{R}^{d\times d} | \mathbf{D}^\top\mathbf{D} = I_d\ \}$, with each $\mathbf{C}_i^K \in \{\mathbf{D} \in \mathbb{R}^{(d)\times(d/n)} | \mathbf{D}^\top\mathbf{D} = I_{(d/n)}\ \}$, $i \in [n]$. $K = \frac{\Delta L}{12ln^{1/2}\epsilon^2}$, with $n \le \frac{144\Delta^2L^2}{l^2\epsilon^4}$ (to ensure $K \ge 1$), $b = \frac{l\epsilon}{L}$, and $R = \sqrt{230K}$. We first verify that $f(\mathbf{x})$ satisfies Assumption 1 (i). For Assumption 1 (i), from (B.73), we have

$$f(\mathbf{0}) - \inf_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) \le \frac{1}{n}\sum_{i=1}^{n}(f_i(\mathbf{0}) - \inf_{\mathbf{x}\in\mathbb{R}^d} f_i(\mathbf{x})) \le \frac{ln^{1/2}\epsilon^2}{L}12K = \frac{ln^{1/2}\epsilon^2}{L}\frac{12\Delta L}{12ln^{1/2}\epsilon^2} = \Delta^{13}.$$

For Assumption 1(ii), for any i, using the $\tilde{f}_{K,\mathbf{B}_i^K}$ has l-Lipschitz continuous gradient, we have

$$\left\|\nabla\tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b) - \nabla\tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{y}/b)\right\|^2 \le l^2\left\|\mathbf{C}_i^{\mathrm{T}}(\mathbf{x}-\mathbf{y})/b\right\|^2, \tag{B.77}$$

Because $\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\|^2 = \left\|\frac{ln^{1/2}\epsilon^2}{Lb}\mathbf{C}_i\left(\nabla\tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b) - \nabla\tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{y}/b)\right)\right\|^2$, and using $\mathbf{C}_i^\top\mathbf{C}_i = I_{d/n}$, we have

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\|^2 \le \left(\frac{ln^{1/2}\epsilon^2}{L}\right)^2\frac{l^2}{b^4}\left\|\mathbf{C}_i^{\mathrm{T}}(\mathbf{x}-\mathbf{y})\right\|^2 = nL^2\left\|\mathbf{C}_i^{\mathrm{T}}(\mathbf{x}-\mathbf{y})\right\|^2, \tag{B.78}$$

where we use $b = \frac{l\epsilon}{L}$. Summing $i = 1, \ldots, n$ and using each $\mathbf{C}_i$ are orthogonal matrix, we have

$$\mathbb{E}\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{y})\|^2 \le L^2\|\mathbf{x}-\mathbf{y}\|^2. \tag{B.79}$$

Then with

$$d \ge 2\max(9n^3K^2, 12n^2KR^2)\log\left(\frac{2n^3K^2}{p}\right) + n^2K \sim \mathcal{O}\left(\frac{n^2\Delta^2L^2}{\epsilon^4}\log\left(\frac{n^2\Delta^2L^2}{\epsilon^4 p}\right)\right),$$

from Lemma 2 of Carmon et al. (2017b) (or similarly Lemma 7 of Woodworth & Srebro (2016) and Theorem 3 of Woodworth & Srebro (2017), also refer to Lemma 12 in the end of the paper), with probability at least $1-p$, after $T = \frac{nK}{2}$ iterations (at the end of iteration $T-1$), for all I_i^{T-1} with

[13] If $\mathbf{x}^0 \ne \mathbf{0}$, we can simply translate the counter example as $f'(\mathbf{x}) = f(\mathbf{x}-\mathbf{x}^0)$, then $f'(\mathbf{x}^0) - \inf_{\mathbf{x}\in\mathbb{R}^d} f'(\mathbf{x}) \le \Delta$.

$i \in [d]$, if $I_i^{T-1} < K$, then for any $j_i \in \{I_i^{T-1}+1, \ldots, K\}$, we have $\left\langle \mathbf{b}_{i,j_i}, \rho(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b) \right\rangle \leq \frac{1}{2}$, where I_i^{T-1} denotes that the algorithm $\mathcal{A}$ has called individual function i with I_i^{T-1} times ($\sum_{i=1}^n I_i^{T-1} = T$) at the end of iteration $T-1$, and $\mathbf{b}_{i,j}$ denotes the j-th column of $\mathbf{B}_i^K$. However, from (B.74), if $\left\langle \mathbf{b}_{i,j_i}, \rho(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b) \right\rangle \leq \frac{1}{2}$, we will have $\|\nabla \tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}/b)\| \geq \frac{1}{2}$. So f_i can be solved only after K times calling it.

From the above analysis, for any algorithm $\mathcal{A}$, after running $T = \frac{nK}{2} = \frac{\Delta L n^{1/2}}{24 l \epsilon^2}$ iterations, at least $\frac{n}{2}$ functions cannot be solved (the worst case is when $\mathcal{A}$ exactly solves $\frac{n}{2}$ functions), so

$$\begin{aligned} \left\|\nabla f(\mathbf{x}^{nK/2})\right\|^2 &= \frac{1}{n^2}\left\|\sum_{i \text{ not solved}} \frac{l n^{1/2}\epsilon^2}{Lb}\mathbf{C}_i \nabla \tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}^{nK/2}/b)\right\|^2 \\ &\overset{a}{=} \frac{1}{n^2}\sum_{i \text{ not solved}} \left\| n^{1/2}\epsilon \nabla \tilde{f}_{K,\mathbf{B}_i^K}(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}^{nK/2}/b)\right\|^2 \overset{(B.74)}{\geq} \frac{\epsilon^2}{8}, \end{aligned} \tag{B.80}$$

where in $\overset{a}{=}$, we use $\mathbf{C}_i^{\top}\mathbf{C}_j = \mathbf{0}_{d/n}$, when $i \neq j$, and $\mathbf{C}_i^{\top}\mathbf{C}_i = I_{d/n}$. □

Lemma 12. *Let $\{\mathbf{x}\}_{0:T}$ with $T = \frac{nK}{2}$ is informed by a certain algorithm in the form (3.8). Then when $d \geq 2\max(9n^3K^2, 12n^3KR^2)\log(\frac{2n^2K^2}{p}) + n^2K$, with probability $1-p$, at each iteration $0 \leq t \leq T$, $\mathbf{x}^t$ can only recover one coordinate.*

Proof. The proof is essentially same to Carmon et al. (2017b) and Woodworth & Srebro (2017). We give a proof here. Before the poof, we give the following definitions:

1. Let i^t denotes that at iteration t, the algorithm choses the i^t-th individual function.

2. Let I_i^t denotes the total times that individual function with index i has been called before iteration k. We have $I_i^0 = 0$ with $i \in [n]$, $i \neq i^t$, and $I_{i^0}^0 = 1$. And for $t \geq 1$,

$$I_i^t = \begin{cases} I_i^{t-1} + 1, & i = i_t. \\ I_i^{t-1}, & \text{otherwise}. \end{cases} \tag{B.81}$$

3. Let $\mathbf{y}_i^t = \rho(\mathbf{C}_i^{\mathrm{T}}\mathbf{x}^t) = \frac{\mathbf{C}_i^{\mathrm{T}}\mathbf{x}^t}{\sqrt{R^2 + \|\mathbf{C}_i^{\mathrm{T}}\mathbf{x}^t\|^2}}$ with $i \in [n]$. We have $\mathbf{y}_i^t \in \mathbb{R}^{d/n}$ and $\|\mathbf{y}_i^t\| \leq R$.

4. Set $\boldsymbol{\mathcal{V}}_i^t$ be the set that $\left(\bigcup_{i=1}^n \left\{\mathbf{b}_{i,1}, \cdots \mathbf{b}_{i,\min(K,I_i^t)}\right\}\right) \bigcup \{\mathbf{y}_i^0, \mathbf{y}_i^1, \cdots, \mathbf{y}_i^t\}$, where $\mathbf{b}_{i,j}$ denotes the j-th column of $\mathbf{B}_i^K$.

5. Set $\boldsymbol{\mathcal{U}}_i^t$ be the set of $\left\{\mathbf{b}_{i,\min(K,I_i^{t-1}+1)}, \cdots, \mathbf{b}_{i,K}\right\}$ with $i \in [n]$. $\boldsymbol{\mathcal{U}}^t = \bigcup_{i=1}^n \boldsymbol{\mathcal{U}}_i^t$. And set $\tilde{\boldsymbol{\mathcal{U}}}_i^t = \left\{\mathbf{b}_{i,\min(K,1)}, \cdots, \mathbf{b}_{i,\min(K,I_i^{t-1})}\right\}$. $\tilde{\boldsymbol{\mathcal{U}}}^t = \bigcup_{i=1}^n \tilde{\boldsymbol{\mathcal{U}}}_i^t$.

6. Let $\boldsymbol{\mathcal{P}}_i^t \in \mathcal{R}^{(d/n)\times(d/n)}$ denote the projection operator to the span of $\mathbf{u} \in \boldsymbol{\mathcal{V}}_i^t$. And let $\boldsymbol{\mathcal{P}}_i^{t\perp}$ denote its orthogonal complement.

Because $\mathcal{A}^t$ performs measurable mapping, the above terms are all measurable on $\boldsymbol{\xi}$ and $\mathbf{B}^{nK}$, where $\boldsymbol{\xi}$ is the random vector in $\mathcal{A}$. It is clear that if for all $0 \le t \le T$ and $i \in [n]$, we have

$$\left|\left\langle \mathbf{u}, \mathbf{y}_i^t \right\rangle\right| < \frac{1}{2}, \quad \text{for all } \ \mathbf{u} \in \boldsymbol{\mathcal{U}}_i^t. \tag{B.82}$$

then at each iteration, we can only recover one index, which is our destination. To prove that (B.82) holds with probability at least $1-p$, we consider a more hard event $\boldsymbol{\mathcal{G}}^t$ as

$$\boldsymbol{\mathcal{G}}^t = \left\{ \left|\left\langle \mathbf{u}, \boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t \right\rangle\right| \le a\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\| \mid \mathbf{u} \in \boldsymbol{\mathcal{U}}^t \ (\text{not } \boldsymbol{\mathcal{U}}_i^t), \ i \in [n] \right\}, \quad t \ge 1, \tag{B.83}$$

with $a = \min\left(\frac{1}{3(T+1)}, \frac{1}{2(1+\sqrt{3T})R}\right)$. And $G^{\le t} = \bigcap_{j=0}^t \boldsymbol{\mathcal{G}}^j$.

We first show that if $\boldsymbol{\mathcal{G}}^{\le T}$ happens, then (B.82) holds for all $0 \le t \le T$. For $0 \le t \le T$, and $i \in [n]$, if $\boldsymbol{\mathcal{U}}_i^t = \varnothing$, (B.82) is right; otherwise for any $\mathbf{u} \in \boldsymbol{\mathcal{U}}_i^t$, we have

$$\begin{aligned}
& \left|\left\langle \mathbf{u}, \mathbf{y}_i^t \right\rangle\right| \\
\le & \left|\left\langle \mathbf{u}, \boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t \right\rangle\right| + \left|\left\langle \mathbf{u}, \boldsymbol{\mathcal{P}}_i^{(t-1)}\mathbf{y}_i^t \right\rangle\right| \\
\le & a\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\| + \left|\left\langle \mathbf{u}, \boldsymbol{\mathcal{P}}_i^{t-1}\mathbf{y}_i^t \right\rangle\right| \le aR + R\left\|\boldsymbol{\mathcal{P}}_i^{t-1}\mathbf{u}\right\|,
\end{aligned} \tag{B.84}$$

where in the last inequality, we use $\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\| \le \|\mathbf{y}_i^{(t-1)}\| \le R$.

If $t=0$, we have $\boldsymbol{\mathcal{P}}_i^{t-1} = \mathbf{0}_{d/n \times d/n}$, then $\left\|\boldsymbol{\mathcal{P}}_i^{t-1}\mathbf{u}\right\| = 0$, so (B.82) holds. When $t \ge 1$, suppose at $t-1$, $\boldsymbol{\mathcal{G}}^{\le t}$ happens then (B.82) holds for all 0 to $t-1$. Then we need to prove that $\|\boldsymbol{\mathcal{P}}_i^{t-1}\mathbf{u}\| \le b = \sqrt{3T}a$ with $\mathbf{u} \in \boldsymbol{\mathcal{U}}_i^t$ and $i \in [n]$. Instead, we prove a stronger results: $\|\boldsymbol{\mathcal{P}}_i^{t-1}\mathbf{u}\| \le b = \sqrt{3T}a$ with all $\mathbf{u} \in \boldsymbol{\mathcal{U}}^t$ and $i \in [n]$. Again, When $t=0$, we have $\|\boldsymbol{\mathcal{P}}_i^{t-1}\mathbf{u}\| = 0$, so it is right, when $t \ge 1$, by Graham-Schmidt procedure on $\mathbf{y}_i^0, \mathbf{b}_{i_0,\min(I_{i0}^0,K)}, \cdots, \mathbf{y}_i^{t-1}, \mathbf{b}_{i_{t-1},\min(I_{it-1}^{t-1},K)}$, we have

$$\left\|\boldsymbol{\mathcal{P}}_i^{t-1}\mathbf{u}\right\|^2 = \sum_{z=0}^{t-1}\left|\left\langle \frac{\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z}{\|\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z\|}, \mathbf{u} \right\rangle\right|^2 + \sum_{z=0,\ I_{iz}^z \le K}^{t-1}\left|\left\langle \frac{\hat{\boldsymbol{\mathcal{P}}}_i^{(z-1)\perp}\mathbf{b}_{i_z,I_{iz}^z}}{\|\hat{\boldsymbol{\mathcal{P}}}_i^{(z-1)\perp}\mathbf{b}_{i_z,I_{iz}^z}\|}, \mathbf{u} \right\rangle\right|^2, \tag{B.85}$$

where

$$\hat{\boldsymbol{\mathcal{P}}}_i^{(z-1)} = \boldsymbol{\mathcal{P}}_i^{(z-1)} + \frac{\left(\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z\right)\left(\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z\right)^{\mathrm{T}}}{\left\|\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z\right\|^2}.$$

Using $\mathbf{b}_{i_z,I_{iz}^z} \perp \mathbf{u}$ for all $\mathbf{u} \in \boldsymbol{\mathcal{U}}^t$, we have

$$\begin{aligned}
& \left|\left\langle \hat{\boldsymbol{\mathcal{P}}}_i^{(z-1)\perp}\mathbf{b}_{i_z,I_{iz}^z}, \mathbf{u} \right\rangle\right| \\
= & \left|0 - \left\langle \hat{\boldsymbol{\mathcal{P}}}_i^{(z-1)}\mathbf{b}_{i_z,I_{iz}^z}, \mathbf{u} \right\rangle\right| \\
\le & \left|\left\langle \boldsymbol{\mathcal{P}}_i^{(z-1)}\mathbf{b}_{i_z,I_{iz}^z}, \mathbf{u} \right\rangle\right| + \left|\left\langle \frac{\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z}{\|\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z\|}, \mathbf{b}_{i_z,I_{iz}^z} \right\rangle\left\langle \frac{\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z}{\|\boldsymbol{\mathcal{P}}_i^{(z-1)\perp}\mathbf{y}_i^z\|}, \mathbf{u} \right\rangle\right|.
\end{aligned} \tag{B.86}$$

For the first term in the right hand of (B.86), by induction, we have

$$\left|\left\langle \mathcal{P}_i^{(z-1)}\mathbf{b}_{i_z,I_{iz}^z}, \mathbf{u}\right\rangle\right| = \left|\left\langle \mathcal{P}_i^{(z-1)}\mathbf{b}_{i_z,I_{iz}^z}, \mathcal{P}_i^{(z-1)}\mathbf{u}\right\rangle\right| \le b^2. \tag{B.87}$$

For the second term in the right hand of (B.86), by assumption (B.83), we have

$$\left|\left\langle \frac{\mathcal{P}_i^{(z-1)\perp}\mathbf{y}_i^z}{\|\mathcal{P}_i^{(z-1)\perp}\mathbf{y}_i^z\|}, \mathbf{b}_{i_z,I_{iz}^z}\right\rangle \left\langle \frac{\mathcal{P}_i^{(z-1)\perp}\mathbf{y}_i^z}{\|\mathcal{P}_i^{(z-1)\perp}\mathbf{y}_i^z\|}, \mathbf{u}\right\rangle\right| \le a^2. \tag{B.88}$$

Also, we have

$$\begin{aligned}
& \left\|\hat{\mathcal{P}}_i^{(z-1)\perp}\mathbf{b}_{i_z,I_{iz}^z}\right\|^2 \qquad\qquad (B.89)\\
= \; & \|\mathbf{b}_{i_z,I_{iz}^z}\|^2 - \left\|\hat{\mathcal{P}}_i^{(z-1)}\mathbf{b}_{i_z,I_{iz}^z}\right\|^2\\
= \; & \|\mathbf{b}_{i_z,I_{iz}^z}\|^2 - \left\|\mathcal{P}_i^{(z-1)}\mathbf{b}_{i_z,I_{iz}^z}\right\|^2 - \left|\left\langle \frac{\mathcal{P}_i^{(z-1)\perp}\mathbf{y}_i^z}{\|\mathcal{P}_i^{(z-1)\perp}\mathbf{y}_i^z\|}, \mathbf{b}_{i_z,I_{iz}^z}\right\rangle\right|^2\\
\ge \; & 1 - b^2 - a^2.
\end{aligned}$$

Substituting (B.86) and (B.89) into (B.85), for all $\mathbf{u} \in \mathcal{U}^t$, we have

$$\begin{aligned}
\left\|\mathcal{P}_i^{t-1}\mathbf{u}\right\|^2 \quad & \le \quad ta^2 + t\frac{(a^2+b^2)^2}{1-(a^2+b^2)}\\
& \overset{a^2+b^2\le(3T+1)a^2\le a}{\le} \quad Ta^2 + T\frac{a^2}{1-a} \overset{a\le 1/2}{\le} 3Ta^2 = b^2.
\end{aligned} \tag{B.90}$$

Thus for (B.84), $t \ge 1$, because $\mathbf{u} \in \mathcal{U}_i^t \subseteq \mathcal{U}^t$, we have

$$\left|\langle \mathbf{u}, \mathbf{y}_i^t\rangle\right| \le (a+b)R \overset{a\le\frac{1}{2(1+\sqrt{3T})R}}{\le} \le \frac{1}{2}. \tag{B.91}$$

This shows that if $\mathcal{G}^{\le T}$ happens, (B.82) holds for all $0 \le t \le T$. Then we prove that $\mathbb{P}(\mathcal{G}^{\le T}) \ge 1-p$. We have

$$\mathbb{P}\left((\mathcal{G}^{\le T})^c\right) \quad = \quad \sum_{t=0}^{T}\mathbb{P}\left((\mathcal{G}^{\le t})^c \mid \mathcal{G}^{<t}\right). \tag{B.92}$$

We give the following definition:

1. Denote $\hat{i}^t$ be the sequence of $i_{0:t-1}$. Let $\hat{\mathcal{S}}^t$ be the set that contains all possible ways of $\hat{i}^t$ ($|\hat{\mathcal{S}}^t| \le n^t$).
2. Let $\tilde{\mathbf{U}}_{\hat{i}^t}^j = [\mathbf{b}_{j,1}, \cdots, \mathbf{b}_{j,\min(K,I_j^{t-1})}]$ with $j \in [n]$, and $\tilde{\mathbf{U}}_{\hat{i}^t} = [\tilde{\mathbf{U}}_{\hat{i}^t}^1, \cdots, \tilde{\mathbf{U}}_{\hat{i}^t}^n]$. $\tilde{\mathbf{U}}_{\hat{i}^t}$ is analogous to $\tilde{\mathcal{U}}^t$, but is a matrix.

3. Let $\mathbf{U}_{\hat{i}^t}^j = [\mathbf{b}_{j,\min(K,I_j^t)}; \cdots; \mathbf{b}_{j,K}]$ with $j \in [n]$, and $\mathbf{U}_{\hat{i}^t} = [\mathbf{U}_{\hat{i}^t}^1, \cdots, \mathbf{U}_{\hat{i}^t}^n]$. $\mathbf{U}_{\hat{i}^t}$ is analogous to $\boldsymbol{\mathcal{U}}^t$, but is a matrix. Let $\bar{\mathbf{U}} = [\tilde{\mathbf{U}}_{\hat{i}^t}, \mathbf{U}_{\hat{i}^t}]$.

We have that

$$
\begin{aligned}
&\mathbb{P}\left((\boldsymbol{\mathcal{G}}^{\leq t})^c \mid \boldsymbol{\mathcal{G}}^{<t}\right) \\
=&\sum_{\hat{i}_0^t \in \hat{\mathcal{S}}^t} \mathbb{E}_{\boldsymbol{\xi}, \mathbf{U}_{\hat{i}_0^t}}\left(\mathbb{P}\left((\boldsymbol{\mathcal{G}}^{\leq t})^c \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi}, \mathbf{U}_{\hat{i}_0^t}\right) \mathbb{P}\left(\hat{i}^t = \hat{i}_0^t \mid \boldsymbol{\mathcal{G}}^{<t}, \boldsymbol{\xi}, \mathbf{U}_{\hat{i}_0^t}\right)\right).
\end{aligned} \tag{B.93}
$$

For $\sum_{\hat{i}_0^t \in \hat{\mathcal{S}}^t} \mathbb{E}_{\boldsymbol{\xi}, \mathbf{U}_{\hat{i}_0^t}} \mathbb{P}\left(\hat{i}^t = \hat{i}_0^t \mid \boldsymbol{\mathcal{G}}^{<t}, \boldsymbol{\xi}, \mathbf{U}_{\hat{i}_0^t}\right) = \sum_{\hat{i}_0^t \in \hat{\mathcal{S}}^t} \mathbb{P}\left(\hat{i}^t = \hat{i}_0^t \mid \boldsymbol{\mathcal{G}}^{<t}\right) = 1$, in the rest, we show that the probability $\mathbb{P}\left((\boldsymbol{\mathcal{G}}^{\leq t})^c \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0,\right)$ for all $\xi_0, \tilde{\mathbf{U}}_0$ is small. By union bound, we have

$$
\begin{aligned}
&\mathbb{P}\left((\boldsymbol{\mathcal{G}}^{\leq t})^c \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0\right) \\
\leq&\sum_{i=1}^n \sum_{\mathbf{u} \in \boldsymbol{\mathcal{U}}^t} \mathbb{P}\left(\left\langle \mathbf{u}, \boldsymbol{\mathcal{P}}_i^{(t-1)\perp} \mathbf{y}_i^t \right\rangle \geq a \|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp} \mathbf{y}_i^t\| \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0\right).
\end{aligned} \tag{B.94}
$$

Note that $\hat{i}_0^t$ is a constant. Because given $\boldsymbol{\xi}$ and $\tilde{\mathbf{U}}_{\hat{i}_0^t}$, under $G^{\leq t}$, both $\boldsymbol{\mathcal{P}}_i^{(t-1)}$ and $\mathbf{y}_i^t$ are known. We prove

$$
\mathbb{P}\left(\mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0 \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0\right) = \mathbb{P}\left(\mathbf{U}_{\hat{i}_0^t} = \mathbf{Z}_i \mathbf{U}_0 \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0\right) \tag{B.95}
$$

where $\mathbf{Z}_i \in \mathbb{R}^{d/n \times d/n}$, $\mathbf{Z}_i^{\mathrm{T}} \mathbf{Z}_i = \mathbf{I}_d$, and $\mathbf{Z}_i \mathbf{u} = \mathbf{u} = \mathbf{Z}_i^{\mathrm{T}} \mathbf{u}$ for all $\mathbf{u} \in \boldsymbol{\mathcal{V}}_i^{t-1}$. In this way, $\frac{\boldsymbol{\mathcal{P}}_i^{(t-1)\perp} \mathbf{u}}{\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp} \mathbf{u}\|}$ has uniformed distribution on the unit space. To prove it, we have

$$
\begin{aligned}
&\mathbb{P}\left(\mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0 \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0\right) \\
=&\frac{\mathbb{P}(\mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0, \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0)}{\mathbb{P}(\boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0)} \\
=&\frac{\mathbb{P}(\boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t \mid \boldsymbol{\xi} = \boldsymbol{\xi}_0, \mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0) p(\boldsymbol{\xi} = \boldsymbol{\xi}_0, \mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0)}{\mathbb{P}(\boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}})},
\end{aligned} \tag{B.96}
$$

And

$$
\begin{aligned}
&\mathbb{P}\left(\mathbf{U}_{\hat{i}_0^t} = \mathbf{Z}_i \mathbf{U}_0 \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0} = \tilde{\mathbf{U}}_0\right) \\
=&\frac{\mathbb{P}(\boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t \mid \boldsymbol{\xi} = \boldsymbol{\xi}_0, \mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \mathbf{Z}_i \tilde{\mathbf{U}}_0) p(\boldsymbol{\xi} = \boldsymbol{\xi}_0, \mathbf{U}_{\hat{i}_0^t} = \mathbf{Z}_i \mathbf{U}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0)}{\mathbb{P}(\boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0)}
\end{aligned} \tag{B.97}
$$

For $\boldsymbol{\xi}$ and $\bar{\mathbf{U}}$ are independent. And $p(\bar{\mathbf{U}}) = p(\mathbf{Z}_i \bar{\mathbf{U}})$, we have $p(\boldsymbol{\xi} = \boldsymbol{\xi}_0, \mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0) = p(\boldsymbol{\xi} = \boldsymbol{\xi}_0, \mathbf{U}_{\hat{i}_0^t} = \mathbf{Z}_i \mathbf{U}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0)$. Then we prove that if $\boldsymbol{\mathcal{G}}^{<t}$ and $\hat{i}^t = \hat{i}_0^t$ happens under $\mathbf{U}_{\hat{i}_0^t} =$

$\mathbf{U}_0, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0$, if and only if $\boldsymbol{\mathcal{G}}^{<t}$ and $\hat{i}^t = \hat{i}_0^t$ happen under $\mathbf{U}_{\hat{i}_0^t} = \mathbf{Z}_i\mathbf{U}_0, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0$.

Suppose at iteration $l-1$ with $l \leq t$, we have the result. At iteration l, suppose $\boldsymbol{\mathcal{G}}^{<l}$ and $\hat{i}^l = \hat{i}_0^l$ happen, given $\mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0$. Let $\mathbf{x}'$ and $(\hat{i}')^j$ are generated by $\boldsymbol{\xi} = \boldsymbol{\xi}_0, \mathbf{U}_{\hat{i}_0^t} = \mathbf{Z}_i\mathbf{U}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0$. Because $\boldsymbol{\mathcal{G}}^{<l}$ happens, thus at each iteration, we can only recover one index until $l-1$. Then $(\mathbf{x}')^j = \mathbf{x}^j$ and $(\hat{i}')^j = \hat{i}^j$. with $j \leq l$. By induction, we only need to prove that $\boldsymbol{\mathcal{G}}^{l-1'}$ will happen. Let $\mathbf{u} \in \boldsymbol{\mathcal{U}}^{l-1}$, and $i \in [n]$, we have

$$\left|\left\langle \mathbf{Z}_i\mathbf{u}, \frac{\boldsymbol{\mathcal{P}}_i^{(l-2)\perp}\mathbf{y}_i^{l-1}}{\|\boldsymbol{\mathcal{P}}_i^{(l-2)\perp}\mathbf{y}_i^{l-1}\|}\right\rangle\right| = \left|\left\langle \mathbf{u}, \mathbf{Z}_i^{\mathrm{T}}\frac{\boldsymbol{\mathcal{P}}_i^{(l-2)\perp}\mathbf{y}_i^{l-1}}{\|\boldsymbol{\mathcal{P}}_i^{(l-2)\perp}\mathbf{y}_i^{l-1}\|}\right\rangle\right| \overset{a}{=} \left|\left\langle \mathbf{u}, \frac{\boldsymbol{\mathcal{P}}_i^{(l-2)\perp}\mathbf{y}_i^{l-1}}{\|\boldsymbol{\mathcal{P}}_i^{(l-2)\perp}\mathbf{y}_i^{l-1}\|}\right\rangle\right|, \tag{B.98}$$

where in $\overset{a}{=}$, we use $\boldsymbol{\mathcal{P}}_i^{(l-2)\perp}\mathbf{y}_i^{l-1}$ is in the span of $\boldsymbol{\mathcal{V}}_i^l \subseteq \boldsymbol{\mathcal{V}}_i^{t-1}$. This shows that if $\boldsymbol{\mathcal{G}}^{<t}$ and $\hat{i}^t = \hat{i}_0^t$ happen under $\mathbf{U}_{\hat{i}_0^t} = \mathbf{U}_0, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0$, then $\boldsymbol{\mathcal{G}}^{<t}$ and $\hat{i}^t = \hat{i}^t$ happen under $\mathbf{U}_{\hat{i}_0^t} = \mathbf{Z}_i\mathbf{U}_0, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0$. In the same way, we can prove the necessity. Thus for any $\mathbf{u} \in \mathbf{U}^t$, if $\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\| \neq 0$ (otherwise, $\left|\left\langle \mathbf{u}, \boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\right\rangle\right| \leq a\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\|$ holds), we have

$$\begin{aligned}
& \mathbb{P}\left(\left\langle \mathbf{u}, \frac{\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t}{\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\|}\right\rangle \geq a \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0\right) \\
\overset{a}{\leq} \;& \mathbb{P}\left(\left\langle \frac{\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{u}}{\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{u}\|}, \frac{\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t}{\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\|}\right\rangle \geq a \mid \boldsymbol{\mathcal{G}}^{<t}, \hat{i}^t = \hat{i}_0^t, \boldsymbol{\xi} = \boldsymbol{\xi}_0, \tilde{\mathbf{U}}_{\hat{i}_0^t} = \tilde{\mathbf{U}}_0\right) \\
\overset{b}{\leq} \;& 2e^{\frac{-a^2(d/n-2T)}{2}},
\end{aligned} \tag{B.99}$$

where in $\overset{a}{\leq}$, we use $\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{u}\| \leq 1$; and in $\overset{b}{\leq}$, we use $\frac{\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t}{\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{y}_i^t\|}$ is a known unit vector and $\frac{\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{u}}{\|\boldsymbol{\mathcal{P}}_i^{(t-1)\perp}\mathbf{u}\|}$ has uniformed distribution on the unit space. Then by union bound, we have $\mathbb{P}\left(\left(\boldsymbol{\mathcal{G}}^{\leq t}\right)^c \mid \boldsymbol{\mathcal{G}}^{<t}\right) \leq 2(n^2K)e^{\frac{-a^2(d/n-2T)}{2}}$. Thus

$$\begin{aligned}
\mathbb{P}\left(\left(\boldsymbol{\mathcal{G}}^{\leq T}\right)^c\right) \quad & \leq \quad 2(T+1)n^2K\exp\left(\frac{-a^2(d/n-2T)}{2}\right) \\
& \overset{T=\frac{nK}{2}}{\leq} \quad 2(nK)(n^2K)\exp\left(\frac{-a^2(d/n-2T)}{2}\right).
\end{aligned} \tag{B.100}$$

Then by setting

$$
\begin{aligned}
d/n &\geq 2\max(9n^2K^2, 12nKR^2)\log(\frac{2n^3K^2}{p}) + nK \\
&\geq 2\max(9(T+1)^2, 2(2\sqrt{3T})^2R^2)\log(\frac{2n^3K^2}{p}) + 2T \\
&\geq 2\max(9(T+1)^2, 2(1+\sqrt{3T})^2R^2)\log(\frac{2n^3K^2}{p}) + 2T \\
&\geq \frac{2}{a^2}\log(\frac{2n^3K^2}{p}) + 2T,
\end{aligned}
\tag{B.101}
$$

we have $\mathbb{P}\left(\left(\boldsymbol{\mathcal{G}}^{\leq T}\right)^c\right) \leq p$. This ends proof.

$\square$