Stochastic Recursive Gradient Descent Ascent for Stochastic Nonconvex-Strongly-Concave Minimax Problems

Luo Luo[1] **Haishan Ye**[2] **Zhichao Huang**[1] **Tong Zhang**[1]
[1]Department of Mathematics, The Hong Kong University of Science and Technology
[2]Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong, Shenzhen
luoluo@ust.hk hsye_cs@outlook.com zhuangbx@connect.ust.hk tongzhang@ust.hk

Abstract

We consider nonconvex-concave minimax optimization problems of the form $\min_{\mathbf{x}} \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y})$, where f is strongly-concave in $\mathbf{y}$ but possibly nonconvex in $\mathbf{x}$ and $\mathcal{Y}$ is a convex and compact set. We focus on the stochastic setting, where we can only access an unbiased stochastic gradient estimate of f at each iteration. This formulation includes many machine learning applications as special cases such as robust optimization and adversary training. We are interested in finding an $\mathcal{O}(\varepsilon)$-stationary point of the function $\Phi(\cdot) = \max_{\mathbf{y}\in\mathcal{Y}} f(\cdot,\mathbf{y})$. The most popular algorithm to solve this problem is stochastic gradient decent ascent, which requires $\mathcal{O}(\kappa^3\varepsilon^{-4})$ stochastic gradient evaluations, where κ is the condition number. In this paper, we propose a novel method called Stochastic Recursive gradiEnt Descent Ascent (SREDA), which estimates gradients more efficiently using variance reduction. This method achieves the best known stochastic gradient complexity of $\mathcal{O}(\kappa^3\varepsilon^{-3})$, and its dependency on ε is optimal for this problem.

1 Introduction

This paper considers the following minimax optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^d} \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y}) \triangleq \mathbb{E}\left[F(\mathbf{x},\mathbf{y};\boldsymbol{\xi})\right], \tag{1}$$

where the stochastic component $F(\mathbf{x},\mathbf{y};\boldsymbol{\xi})$, indexed by some random vector $\boldsymbol{\xi}$, is ℓ-gradient Lipschitz on average. This minimax optimization formulation includes many machine learning applications such as regularized empirical risk minimization [42, 53], AUC maximization [40, 49], robust optimization [14, 47], adversarial training [16, 17, 41] and reinforcement learning [13, 44]. Many existing work [8, 9, 12, 13, 18, 28, 34, 35, 42, 46, 49, 51, 53] focused on the convex-concave case of problem (1), where f is convex in $\mathbf{x}$ and concave in $\mathbf{y}$. For such problems, one can establish strong theoretical guarantees.

In this paper, we focus on a more general case of (1), where $f(\mathbf{x},\mathbf{y})$ is μ-strongly-concave in $\mathbf{y}$ but possibly nonconvex in $\mathbf{x}$. This case is referred to as stochastic nonconvex-strongly-concave minimax problems, and it is equivalent to the following problem

$$\min_{\mathbf{x}\in\mathbb{R}^d} \left\{\Phi(\mathbf{x}) \triangleq \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y})\right\}. \tag{2}$$

Formulation (2) contains several interesting examples in machine learning such as robust optimization [14, 47] and adversarial training [17, 41].

Since Φ is possibly nonconvex, it is infeasible to find the global minimum in general. One important task of the minimax problem is finding an approximate stationary point of Φ. A simple way to

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

solve this problem is stochastic gradient descent with max-oracle (SGDmax) [19, 25]. The algorithm includes a nested loop to solve $\max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y})$ and use the solution to run approximate stochastic gradient descent (SGD) on $\mathbf{x}$. Lin et al. [25] showed that we can solve problem (2) by directly extending SGD to stochastic gradient descent ascent (SGDA). The iteration of SGDA is just using gradient descent on $\mathbf{x}$ and gradient descent on $\mathbf{y}$. The complexity of SGDA to find $\mathcal{O}(\varepsilon)$-stationary point of Φ in expectation is $\mathcal{O}\left(\kappa^3\varepsilon^{-4}\right)$ stochastic gradient evaluations, where $\kappa \triangleq \ell/\mu$ is the condition number. SGDA is more efficient than SGDmax whose complexity is $\mathcal{O}\left(\left(\kappa^3\varepsilon^{-4}\right)\log(1/\varepsilon)\right)$.

One insight of SGDA is that the algorithm selects an appropriate ratio of learning rates for $\mathbf{x}$ and $\mathbf{y}$. Concretely, the learning rate for updating $\mathbf{y}$ is $\mathcal{O}(\kappa^2)$ times that of $\mathbf{x}$. Using this idea, it can be shown that the nested loop of SGDmax is unnecessary, and SGDA eliminates the logarithmic term in the complexity result. In addition, Rafique et al. [37] presented some nested-loop algorithms that also achieved $\mathcal{O}\left(\kappa^3\varepsilon^{-4}\right)$ complexity. Recently, Yan et al. [48] proposed Epoch-GDA which considered constraints on both two variables.

Lin et al. [26] proposed a deterministic algorithm called minimax proximal point algorithm (Minimax PPA) to solve nonconvex-strongly-concave minimax problem whose complexity has square root dependence on κ. Barazandeh and Razaviyayn [7], Ostrovskii et al. [33], Thekumparampil et al. [43] also studied the non-convex-concave minimax problems, however, these methods do not cover the stochastic setting in this paper and only work for a special case of problem (2) when the stochastic variable $\boldsymbol{\xi}$ is finitely sampled from $\{\boldsymbol{\xi}_1,\dots,\boldsymbol{\xi}_n\}$ (a.k.a. finite-sum case). That is,

$$f(\mathbf{x},\mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^{n} F(\mathbf{x},\mathbf{y};\boldsymbol{\xi}_i). \tag{3}$$

In this paper, we propose a novel algorithm called Stochastic Recursive gradiEnt Descent Ascent (SREDA) for stochastic nonconvex-strongly-concave minimax problems. Unlike SGDmax and SGDA, which only iterate with current stochastic gradients, our SREDA updates the estimator recursively and reduces its variance.

The variance reduction techniques have been widely used in convex and nonconvex minimization problems [1–4, 11, 15, 20, 22, 23, 30, 31, 36, 38, 39, 45, 52, 54] and convex-concave saddle point problems [9, 12, 13, 28, 35]. However, the nonconvex-strongly-concave minimax problems have two variables $\mathbf{x}$ and $\mathbf{y}$ and their roles in the objective function are quite different. To apply the technique of variance reduction, SREDA employs a concave maximizer with multi-step iteration on $\mathbf{y}$ to simultaneously balance the learning rates, gradient batch sizes and iteration numbers of the two variables. We prove SREDA reduces the number of stochastic gradient evaluations to $\mathcal{O}(\kappa^3\varepsilon^{-3})$, which is the best known upper bound complexity. The result gives optimal dependency on ε since the lower bound of stochastic first order algorithms for general nonconvex optimization is $\mathcal{O}(\varepsilon^{-3})$ [6]. For finite-sum cases, the gradient cost of SREDA is $\mathcal{O}\left(n\log(\kappa/\varepsilon)+\kappa^2 n^{1/2}\varepsilon^{-2}\right)$ when $n \geq \kappa^2$, and $\mathcal{O}\left((\kappa^2+\kappa n)\varepsilon^{-2}\right)$ when $n \leq \kappa^2$. This result is sharper than Minimax PPA [26] in the case of n is larger than κ^2. We summarize the comparison of all algorithms in Table 1.

The paper is organized as follows. In Section 2, we present notations and preliminaries. In Section 3, we review the existing work for stochastic nonconvex-strongly-concave optimization and related techniques. In Section 4, we present the SREDA algorithm and the main theoretical result. In Section 5, we give a brief overview of our convergence analysis. In Section 6, we demonstrate the effectiveness of our methods on robust optimization problem. We conclude this work in Section 7.

2 Notation and Preliminaries

We first introduce the notations and preliminaries used in this paper. For a differentiable function $f(\mathbf{x},\mathbf{y})$, we denote the partial gradient of f with respect to $\mathbf{x}$ and $\mathbf{y}$ at $(\mathbf{x},\mathbf{y})$ as $\nabla_{\mathbf{x}}f(\mathbf{x},\mathbf{y})$ and $\nabla_{\mathbf{y}}f(\mathbf{x},\mathbf{y})$ respectively. We use $\|\cdot\|_2$ to denote the Euclidean norm of vectors. For a finite set $\mathcal{S}$, we denote its cardinality as $|\mathcal{S}|$. We assume that the minimax problem (2) satisfies the following assumptions.

Assumption 1. *The function $\Phi(\cdot)$ is lower bounded, i.e., we have $\Phi^* = \inf_{\mathbf{x}\in\mathbb{R}^d}\Phi(\mathbf{x}) > -\infty$.*

Assumption 2. *The component function F has an average ℓ-Lipschitz gradient, i.e., there exists a constant $\ell > 0$ such that $\mathbb{E}\left\|\nabla F(\mathbf{x},\mathbf{y};\boldsymbol{\xi}) - \nabla F(\mathbf{x}',\mathbf{y}';\boldsymbol{\xi})\right\|_2^2 \leq \ell^2(\|\mathbf{x}-\mathbf{x}'\|_2^2 + \|\mathbf{y}-\mathbf{y}'\|_2^2)$ for any $(\mathbf{x},\mathbf{y})$, $(\mathbf{x}',\mathbf{y}')$ and random vector $\boldsymbol{\xi}$*

Table 1: We present the comparison on stochastic gradient complexities of algorithms to solve stochastic problem (2) and finite-sum problem (3). We use notation $\tilde{\mathcal{O}}(\cdot)$ to hide logarithmic factors. Some baseline algorithms solve problem (3) without considering the finite-sum structure and we regard the cost of full gradient evaluation is $\mathcal{O}(n)$.

Algorithm	Stochastic	Finite-sum	Reference
SGDmax (GDmax)	$\tilde{\mathcal{O}}(\kappa^3\varepsilon^{-4})$	$\tilde{\mathcal{O}}(\kappa^2 n\varepsilon^{-2})$	[19, 25]
PGSMD / PGSVRG	$\mathcal{O}(\kappa^3\varepsilon^{-4})$	$\mathcal{O}(\kappa^2 n\varepsilon^{-2})$	[37]
MGDA / HiBSA	–	$\mathcal{O}(\kappa^4 n\varepsilon^{-2})$	[27, 32]
Minimax PPA	–	$\tilde{\mathcal{O}}(\kappa^{1/2} n\varepsilon^{-2})$	[26]
SGDA (GDA)	$\mathcal{O}(\kappa^3\varepsilon^{-4})$	$\mathcal{O}(\kappa^2 n\varepsilon^{-2})$	[25]
SREDA	$\mathcal{O}(\kappa^3\varepsilon^{-3})$	$\begin{cases}\tilde{\mathcal{O}}\left(n+\kappa^2 n^{1/2}\varepsilon^{-2}\right), & n\geq\kappa^2\\ \mathcal{O}\left((\kappa^2+\kappa n)\varepsilon^{-2}\right), & n\leq\kappa^2\end{cases}$	this paper

Assumption 3. *The component function F is concave in $\mathbf{y}$. That is, for any $\mathbf{x}$, $\mathbf{y}$, $\mathbf{y}'$ and random vector $\boldsymbol{\xi}$, we have $F(\mathbf{x},\mathbf{y};\boldsymbol{\xi})\leq F(\mathbf{x},\mathbf{y}';\boldsymbol{\xi})+\langle\nabla_{\mathbf{y}}F(\mathbf{x},\mathbf{y}';\boldsymbol{\xi}),\mathbf{y}-\mathbf{y}'\rangle$.*

Assumption 4. *The function $f(\mathbf{x},\mathbf{y})$ is μ-strongly-concave in $\mathbf{y}$. That is, there exists a constant $\mu>0$ such that for any $\mathbf{x}$, $\mathbf{y}$ and $\mathbf{y}'$, we have $f(\mathbf{x},\mathbf{y})\leq f(\mathbf{x},\mathbf{y}')+\langle\nabla_{\mathbf{y}}f(\mathbf{x},\mathbf{y}'),\mathbf{y}-\mathbf{y}'\rangle-\frac{\mu}{2}\left\|\mathbf{y}-\mathbf{y}'\right\|_2^2$.*

Assumption 5. *The gradient of each component function $F(\mathbf{x},\mathbf{y};\boldsymbol{\xi})$ has bounded variance. That is, there exists a constant $\sigma>0$ such that $\mathbb{E}\left\|\nabla F(\mathbf{x},\mathbf{y};\boldsymbol{\xi})-\nabla f(\mathbf{x},\mathbf{y})\right\|_2^2\leq\sigma^2<\infty$ for any $\mathbf{x}$, $\mathbf{y}$ and random vector $\boldsymbol{\xi}$.*

Under the assumptions of Lipschitz-gradient and strongly-concavity on f, we can show that $\Phi(\cdot)$ also has Lipschitz-gradient.

Lemma 1 ([25, Lemma 4.3])**.** *Under Assumptions 2 and 4, the function $\Phi(\cdot)=\max_{\mathbf{y}\in\mathcal{Y}}f(\cdot,\mathbf{y})$ has $(\ell+\kappa\ell)$-Lipschitz gradient. Additionally, the function $\mathbf{y}^*(\cdot)=\arg\max_{\mathbf{y}}f(\cdot,\mathbf{y})$ is unique defined and we have $\nabla\Phi(\cdot)=\nabla_{\mathbf{x}}f(\cdot,\mathbf{y}^*(\cdot))$.*

Since Φ is differentiable, we may define ε-stationary point based on its gradient. The goal of this paper is to establish a stochastic gradient algorithm that output an $\mathcal{O}(\varepsilon)$-stationary point in expectation.

Definition 1. *We call $\mathbf{x}$ an $\mathcal{O}(\varepsilon)$-stationary point of Φ if $\left\|\nabla\Phi(\mathbf{x})\right\|_2\leq\mathcal{O}(\varepsilon)$.*

We also need the notations of projection and gradient mapping to address the constraint on $\mathcal{Y}$.

Definition 2. *We define the projection of $\mathbf{y}$ on to convex set $\mathcal{Y}$ by $\Pi_{\mathcal{Y}}(\mathbf{y})=\arg\min_{\mathbf{z}\in\mathcal{Y}}\left\|\mathbf{z}-\mathbf{y}\right\|_2$.*

Definition 3. *We define the gradient mapping of f at $(\mathbf{x}',\mathbf{y}')$ with respect to $\mathbf{y}$ as follows*

$$\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}',\mathbf{y}')=\frac{1}{\lambda}\left(\mathbf{y}'-\Pi_{\mathcal{Y}}\left(\mathbf{y}'+\lambda\nabla_{\mathbf{y}}f(\mathbf{x}',\mathbf{y}')\right)\right),\ \textit{where } \lambda>0.$$

3 Related Work

In this section, we review recent works for solving stochastic nonconvex-strongly-convex minimax problem (2) and introduce variance reduction techniques in stochastic optimization.

3.1 Nonconvex-Strongly-Concave Minimax

We present SGDmax [19, 25] in Algorithm 1. We can realize the max-oracle by stochastic gradient ascent (SGA) with $\mathcal{O}(\kappa^2\varepsilon^{-2}\log(1/\varepsilon))$ stochastic gradient evaluations to achieve sufficient accuracy. Using $S=\mathcal{O}(\kappa\varepsilon^{-2})$ guarantees that the variance of the stochastic gradients is less than $\mathcal{O}(\kappa^{-1}\varepsilon^2)$. It requires $\mathcal{O}(\kappa\varepsilon^{-2})$ iterations with step size $\eta=\mathcal{O}(1/(\kappa\ell))$ to obtain an $\mathcal{O}(\varepsilon)$-stationary point of Φ. The total stochastic gradient evaluation complexity is $\mathcal{O}(\kappa^3\varepsilon^{-4}\log(1/\varepsilon))$. The procedure of SGDA is shown in Algorithm 2.

Since variables $\mathbf{x}$ and $\mathbf{y}$ are not symmetric, we need to select different step sizes for them. In our case, we choose $\eta = \mathcal{O}(1/(\kappa^2\ell))$ and $\lambda = \mathcal{O}(1/\ell)$. This leads to an $\mathcal{O}(\kappa^3\varepsilon^{-4})$ complexity to obtain an $\mathcal{O}(\varepsilon)$-stationary point with $S = \mathcal{O}(\kappa\varepsilon^{-2})$ and $\mathcal{O}(\kappa^2\varepsilon^{-2})$ iterations [25]. Rafique et al. proposed proximally guided stochastic mirror descent and variance reduction (PGSMD / PGSVRG) whose complexity is also $\mathcal{O}(\kappa^3\varepsilon^{-4})$. Both of the above algorithms reveal that the key of solving problem (2) efficiently is to update $\mathbf{y}$ much more frequently than $\mathbf{x}$. The natural intuition is that finding stationary point of a nonconvex function is typically more difficult than finding that of a concave or convex function. SGDmax implements it by updating $\mathbf{y}$ more frequently (SGA in max-oracle) while SGDA iterates $\mathbf{y}$ with a larger step size such that $\lambda/\eta = \mathcal{O}(\kappa^2)$.

3.2 Variance Reduction Techniques

Variance reduction techniques has been widely used in stochastic optimization [2, 4, 15, 22, 23, 30, 31, 36, 38]. One scheme of this type of methods is StochAstic Recursive grAdient algoritHm (SARAH) [30, 31]. Nguyen et al. [30] first proposed it for convex minimization and established a convergence result. For nonconvex optimization, a closely related method is Stochastic Path-Integrated Differential EstimatoR (SPIDER) [15]. The algorithm estimates the gradient recursively together with a normalization rule, which guarantees the approximation error of the gradient is $\mathcal{O}(\varepsilon^2)$ at each step. As a result, it can find $\mathcal{O}(\varepsilon)$-stationary point of the nonconvex objective in $\mathcal{O}(\varepsilon^{-3})$ complexity, which matches the lower bound [6]. This idea can also be extended to nonsmooth cases [36, 45].

It is also possible to employ variance reduction to solve minimax problems. Most of the existing works focused on the convex-concave case. For example, Chavdarova et al. [9], Palaniappan and Bach [35], extend SVRG [20, 52] and SAGA [11] to solving strongly-convex-strongly-concave minimax problem in the finite-sum case, and established a linear convergence. One may also use the Catalyst framework [24, 35] and proximal point iteration [10, 28] to further accelerate when the problem is ill-conditioned. Du and Hu [12], Du et al. [13] pointed out that for some special cases, the strongly-convex and strongly-concave assumptions of linear convergence for minimax problem may not be necessary. Additionally, Zhang and Xiao [50] solved multi-level composite optimization problems by variance reduction, but the oracles in their algorithms are different from our settings.

Algorithm 1 SGDmax

```
Input initial point $\mathbf{x}_0$, learning rate $\eta > 0$, batch size $S > 0$, max-oracle accuracy $\zeta$
for $k = 0, \ldots, K$ do
    draw $S$ samples $\{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_S\}$
    find $\mathbf{y}_k$ so that $\mathbb{E}[f(\mathbf{x}_k, \mathbf{y}_k)] \geq \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}_k, \mathbf{y}) - \zeta$
    $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \cdot \frac{1}{S}\sum_{i=1}^{S} \nabla_{\mathbf{x}} F(\mathbf{x}_k, \mathbf{y}_k; \boldsymbol{\xi}_i)$
end for
Output $\hat{\mathbf{x}}$ chosen uniformly at random from $\{\mathbf{x}_i\}_{i=0}^{K}$
```

Algorithm 2 SGDA

```
Input initial point $(\mathbf{x}_0, \mathbf{y}_0)$, learning rates $\eta > 0$ and $\lambda > 0$, batch size $S > 0$
for $k = 0, \ldots, K$ do
    draw $M$ samples $\{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_S\}$
    $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta \cdot \frac{1}{S}\sum_{i=1}^{S} \nabla_{\mathbf{x}} F(\mathbf{x}_k, \mathbf{y}_k; \boldsymbol{\xi}_i)$
    $\mathbf{y}_{k+1} = \Pi_{\mathcal{Y}}\left(\mathbf{y}_k + \lambda \cdot \frac{1}{S}\sum_{i=1}^{S} \nabla_{\mathbf{y}} F(\mathbf{x}_k, \mathbf{y}_k; \boldsymbol{\xi}_i)\right)$
end for
Output $\hat{\mathbf{x}}$ chosen uniformly at random from $\{\mathbf{x}_i\}_{i=0}^{K}$
```

Algorithm 3 SREDA

```
Input initial point $\mathbf{x}_0$, learning rates $\eta_k, \lambda > 0$, batch sizes $S_1, S_2 > 0$; periods $q, m > 0$, number of initial iterations $K_0$
$\mathbf{y}_0 = \text{PiSARAH}\left(-f(\mathbf{x}_k, \cdot),\ K_0\right)$
for $k = 0, \ldots, K-1$ do
    if $\mod(k, q) = 0$
        draw $S_1$ samples $\{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_{S_1}\}$
        $\mathbf{v}_k = \frac{1}{S_1}\sum_{i=1}^{S_1} \nabla_{\mathbf{x}} F(\mathbf{x}_k, \mathbf{y}_k; \boldsymbol{\xi}_i)$
        $\mathbf{u}_k = \frac{1}{S_1}\sum_{i=1}^{S_1} \nabla_{\mathbf{y}} F(\mathbf{x}_k, \mathbf{y}_k; \boldsymbol{\xi}_i)$
    else
        $\mathbf{v}_k = \mathbf{v}'_k$
        $\mathbf{u}_k = \mathbf{u}'_k$
    end if
    $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_k \mathbf{v}_k$
    $(\mathbf{y}_{k+1}, \mathbf{v}'_{k+1}, \mathbf{u}'_{k+1}) = \text{ConcaveMaximizer}\left(k, m, S_2, \mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_k, \mathbf{u}_k, \mathbf{v}_k\right)$
end for
Output $\hat{\mathbf{x}}$ chosen uniformly at random from $\{\mathbf{x}_i\}_{i=0}^{K-1}$
```

4 Algorithms and Main Results

In this section, we propose a novel algorithm for solving problem (2), which we call Stochastic Recursive gradiEnt Descent Ascent (SREDA). We show that the algorithm finds an $\mathcal{O}(\varepsilon)$-stationary point with a complexity of $\mathcal{O}(\kappa^3\varepsilon^{-3})$ stochastic gradient evaluations, and this result may be extended to the finite-sum case (3).

4.1 Stochastic Recursive Gradient Descent Ascent

SREDA uses variance reduction to track the gradient estimator recursively. Because there are two variables $\mathbf{x}$ and $\mathbf{y}$ in our problem (2), it is not efficient to combine SGDA with SPIDER [15] or (inexact) SARAH [30, 31] directly. The algorithm should approximate the gradient of $f(\mathbf{x}_k, \mathbf{y}_k)$ with small error, and keep the value of $f(\mathbf{x}_k, \mathbf{y}_k)$ sufficiently close to $\Phi(\mathbf{x}_k)$. To achieve this, in the proposed method SREDA, we employ a concave maximizer with stochastic variance reduced gradient ascent on $\mathbf{y}$. The details of SREDA and the concave maximizer are presented in Algorithm 3 and Algorithm 4 respectively. In the rest of this section, we show SREDA can find an $\mathcal{O}(\varepsilon)$-stationary point in $\mathcal{O}(\kappa^3\varepsilon^{-3})$ stochastic gradient evaluations.

In the initialization of SREDA, we hope to obtain $\mathbf{y}_0 \approx \arg\max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}_0, \mathbf{y}_0)$ for given $\mathbf{x}_0$ such that $\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_0, \mathbf{y}_0)\right\|_2^2 \le \mathcal{O}(\kappa^{-2}\varepsilon^2)$. We proposed a new algorithm called projected inexact SARAH (PiSARAH) to address it. PiSARAH extends inexact SARAH (iSARAH) [31] to constrained case, which could achieve the desired accuracy of our initialization with a complexity of $\mathcal{O}(\kappa^2\varepsilon^{-2}\log(\kappa/\varepsilon))$. We present the details of PiSARAH in Appendix C.

SREDA estimates the gradient of $f(\mathbf{x}_k, \mathbf{y}_k)$ by $(\mathbf{v}_k, \mathbf{u}_k) \approx (\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k), \nabla_{\mathbf{y}} f(\mathbf{x}_k, \mathbf{y}_k))$. As illustrated in Algorithm 4, we evaluate the gradient of f with a large batch size $S_1 = \mathcal{O}(\kappa^2\varepsilon^{-2})$ at the beginning of each period, and update the gradient estimate recursively in concave maximizer with a smaller batch size $S_2 = \mathcal{O}(\kappa\varepsilon^{-1})$.

For variable $\mathbf{x}_k$, we adopt a normalized stochastic gradient descent with a learning rate for theoretical analysis:

$$\eta_k = \min\left(\frac{\varepsilon}{\ell\left\|\mathbf{v}_k\right\|_2}, \frac{1}{2\ell}\right) \cdot \mathcal{O}(\kappa^{-1}).$$

With this step size, the change of $\mathbf{x}_k$ is not dramatic at each iteration, which leads to accurate gradient estimates. To simplify implementations of the algorithm, we can also use a fixed learning rate in practical.

Algorithm 4 ConcaveMaximizer $(k, m, S_2, \mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_k, \mathbf{u}_k, \mathbf{v}_k)$

```
**Initialize** $\tilde{\mathbf{x}}_{k,-1} = \mathbf{x}_k$, $\tilde{\mathbf{y}}_{k,-1} = \mathbf{y}_k$, $\tilde{\mathbf{x}}_{k,0} = \mathbf{x}_{k+1}$, $\tilde{\mathbf{y}}_{k,0} = \mathbf{y}_k$.
draw $S_2$ samples $\{\boldsymbol{\xi}_1, \dots, \boldsymbol{\xi}_{S_2}\}$
$\tilde{\mathbf{v}}_{k,0} = \mathbf{v}_k + \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{x}} F(\tilde{\mathbf{x}}_{k,0}, \tilde{\mathbf{y}}_{k,0}; \boldsymbol{\xi}_i) - \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{x}} F(\tilde{\mathbf{x}}_{k,-1}, \tilde{\mathbf{y}}_{k,-1}; \boldsymbol{\xi}_i)$
$\tilde{\mathbf{u}}_{k,0} = \mathbf{u}_k + \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{y}} F(\tilde{\mathbf{x}}_{k,0}, \tilde{\mathbf{y}}_{k,0}; \boldsymbol{\xi}_i) - \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{y}} F(\tilde{\mathbf{x}}_{k,-1}, \tilde{\mathbf{y}}_{k,-1}; \boldsymbol{\xi}_i)$
$\tilde{\mathbf{x}}_{k,1} = \tilde{\mathbf{x}}_{k,0}$
$\tilde{\mathbf{y}}_{k,1} = \Pi_{\mathcal{Y}}\left(\tilde{\mathbf{y}}_{k,0} + \lambda \tilde{\mathbf{u}}_{k,0}\right)$
**for** $t = 1, \dots, m+1$ **do**
    draw $S_2$ samples $\{\boldsymbol{\xi}_{t,1}, \dots, \boldsymbol{\xi}_{t,S_2}\}$
    $\tilde{\mathbf{v}}_{k,t} = \tilde{\mathbf{v}}_{k,t-1} + \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{x}} F(\tilde{\mathbf{x}}_{k,t}, \tilde{\mathbf{y}}_{k,t}; \boldsymbol{\xi}_{t,i}) - \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{x}} F(\tilde{\mathbf{x}}_{k,t-1}, \tilde{\mathbf{y}}_{k,t-1}; \boldsymbol{\xi}_{t,i})$
    $\tilde{\mathbf{u}}_{k,t} = \tilde{\mathbf{u}}_{k,t-1} + \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{y}} F(\tilde{\mathbf{x}}_{k,t}, \tilde{\mathbf{y}}_{k,t}; \boldsymbol{\xi}_{t,i}) - \frac{1}{S_2}\sum_{i=1}^{S_2} \nabla_{\mathbf{y}} F(\tilde{\mathbf{x}}_{k,t-1}, \tilde{\mathbf{y}}_{k,t-1}; \boldsymbol{\xi}_{t,i})$
    $\tilde{\mathbf{x}}_{k,t+1} = \tilde{\mathbf{x}}_{k,t}$
    $\tilde{\mathbf{y}}_{k,t+1} = \Pi_{\mathcal{Y}}\left(\tilde{\mathbf{y}}_{k,t} + \lambda \tilde{\mathbf{u}}_{k,t}\right)$
**end for**
**Output** $\tilde{\mathbf{y}}_{k,s_k}$, $\tilde{\mathbf{v}}_{k,s_k}$ and $\tilde{\mathbf{u}}_{k,s_k}$ where $s_k$ is chosen uniformly at random from $\{1, \dots, m\}$
```

For variable $\mathbf{y}_k$, we additionally expect $f(\mathbf{x}_k, \mathbf{y}_k)$ is a good approximation of $\Phi(\mathbf{x}_k)$, which implies the gradient mapping with respect to $\mathbf{y}_k$ should be small enough. We hope to maintain the inequality $\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \mathbf{y}_k)\right\|_2^2 \le \mathcal{O}(\kappa^{-2}\varepsilon^2)$ holds. Hence, we include a multi-step concave maximizer to update $\mathbf{y}$ whose details given in Algorithm 4. This procedure can be regarded as one epoch of PiSARAH. We choose the step size $\lambda = \mathcal{O}(1/\ell)$ for inner iterations, which simultaneously ensure that the gradient mapping with respect to $\mathbf{y}$ is small enough and the change of $\mathbf{y}$ is not dramatic.

4.2 Complexity Analysis

As shown in Algorithm 3, SREDA updates variables with a large batch size per q iterations. We choose $q = \mathcal{O}(\varepsilon^{-1})$ as a balance between the number of large batch evaluations with $S_1 = \mathcal{O}(\kappa^2\varepsilon^{-2})$ samples and the concave maximizer with $\mathcal{O}(\kappa)$ iterations and $S_2 = \mathcal{O}(\kappa\varepsilon^{-1})$ samples.

Based on above parameter setting, we can obtain an approximate stationary point $\hat{\mathbf{x}}$ in expectation such that $\mathbb{E}\left\|\nabla\Phi(\hat{\mathbf{x}})\right\|_2 \le \mathcal{O}(\varepsilon)$ with $K = \mathcal{O}(\kappa\varepsilon^{-2})$ outer iterations. The total number of stochastic gradient evaluations of SREDA comes from the initial run of PiSARAH, large batch gradient evaluation (S_1 samples) and concave maximizer. That is,

$$\mathcal{O}(\kappa^2\varepsilon^{-2}\log(\kappa/\varepsilon)) + \mathcal{O}\left(K/q \cdot S_1\right) + \mathcal{O}\left(K \cdot S_2 \cdot m\right) = \mathcal{O}(\kappa^3\varepsilon^{-3}).$$

Let $\Delta_f = f(\mathbf{x}_0, \mathbf{y}_0) + \frac{134\varepsilon^2}{\kappa\ell} - \Phi^*$, then we formally present the main result in Theorem 1.

Theorem 1. *Under Assumptions 1-5 with the following parameter choices:*

$$\zeta = \kappa^{-2}\varepsilon^2,\ \eta_k = \min\left(\frac{\varepsilon}{5\kappa\ell\left\|\mathbf{v}_k\right\|_2}, \frac{1}{10\kappa\ell}\right),\ \lambda = \frac{1}{8\ell},\ S_1 = \left\lceil \frac{2250}{19}\sigma^2\kappa^{-2}\varepsilon^2 \right\rceil,$$

$$S_2 = \left\lceil \frac{3687}{76}\kappa q \right\rceil,\ q = \left\lceil \varepsilon^{-1} \right\rceil,\ K = \left\lceil \frac{100\kappa\ell\varepsilon^{-2}\Delta_f}{9} \right\rceil \text{ and } m = \lceil 1024\kappa \rceil,$$

Algorithm 3 outputs $\hat{\mathbf{x}}$ such that $\mathbb{E}\left\|\nabla\Phi(\hat{\mathbf{x}})\right\|_2 \le 1504\varepsilon$ with $\mathcal{O}(\kappa^3\varepsilon^{-3})$ stochastic gradient evaluations.

We should point out the complexity shown in Theorem 1 gives optimal dependency on ε. We consider the special case of minimax problem whose objective function has the form

$$f(\mathbf{x}, \mathbf{y}) = g(\mathbf{x}) + h(\mathbf{y})$$

where g is possibly nonconvex and h is strongly-concave, which leads to minimizing on $\mathbf{x}$ and maximizing on $\mathbf{y}$ are independent.

Algorithm 5 SREDA (Finite-sum Case)

```
**Input** initial point $\mathbf{x}_0$, learning rates $\eta_k, \lambda > 0$, batch sizes $S_1, S_2 > 0$; periods $q, m > 0$; number of initial iterations $K_0$
$\mathbf{y}_0 = \text{PSARAH}\left(-f(\mathbf{x}_k, \cdot),\ K_0\right)$
**for** $k = 0, \ldots, K-1$ **do**
    **if** $\mod(k, q) = 0$
        $\mathbf{v}_k = \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)$
        $\mathbf{u}_k = \nabla_{\mathbf{y}} f(\mathbf{x}_k, \mathbf{y}_k)$
    **else**
        $\mathbf{v}_k = \mathbf{v}'_k$
        $\mathbf{u}_k = \mathbf{u}'_k$
    **end if**
    $\mathbf{x}_{k+1} = \mathbf{x}_k - \eta_k \mathbf{v}_k$
    $(\mathbf{y}_{k+1}, \mathbf{v}'_{k+1}, \mathbf{u}'_{k+1}) = \text{ConcaveMaximizer}\left(k, m, S_2, \mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_k, \mathbf{u}_k, \mathbf{v}_k\right)$
**end for**
**Output** $\hat{\mathbf{x}}$ chosen uniformly at random from $\{\mathbf{x}_i\}_{i=0}^{K-1}$
```

Consequently, finding $\mathcal{O}(\varepsilon)$-stationary point of the corresponding $\Phi(\mathbf{x})$ can be reduced to finding $\mathcal{O}(\varepsilon)$-stationary point of nonconvex function $g(\mathbf{x})$, which is based on the stochastic first order-oracle $\nabla_{\mathbf{x}} F(\mathbf{x}, \mathbf{y}; \xi) = \nabla g(\mathbf{x}; \xi)$ (this equality holds for any $\mathbf{y}$ since $\mathbf{x}$ and $\mathbf{y}$ are independent). Hence, the analysis of stochastic nonconvex minimization problem [6] based on $\nabla g(\mathbf{x}; \xi)$ can directly lead to the $\mathcal{O}(\varepsilon^{-3})$ lower bound for our minimax problem. We can prove it by constructing the separate function as $f(\mathbf{x}, \mathbf{y}) = g(\mathbf{x}) + h(\mathbf{y})$ where g is the nonconvex function in Arjevani et al.'s [6] lower bound analysis of stochastic nonconvex minimization, and h is an arbitrary smooth, μ-strongly concave function. It is obvious that the lower bound complexity of finding an $\mathcal{O}(\varepsilon)$-stationary point of Φ is no smaller than that of finding an $\mathcal{O}(\varepsilon)$-stationary point of g, which requires at least $\mathcal{O}(\varepsilon^{-3})$ stochastic gradient evaluations [6].

4.3 Extension to Finite-sum Case

SREDA also works for nonconvex-strongly-concave minimax optimization in the finite-sum case (3) with little modification of Algorithm 3. We just need to replace line 5-7 of Algorithm 3 with the full gradients, and use projected SARAH (PSARAH)[1] to initialization. We present the details in Algorithm 5. The algorithm is more efficient than Minimax PPA [26] when $n \geq \kappa^2$. We state the result formally in Theorem 2.

Theorem 2. *Suppose Assumption 1-4 hold. In the finite-sum case with $n \geq \kappa^2$, we set the parameters*

$$\zeta = \kappa^{-2}\varepsilon^2,\ \eta_k = \min\left(\frac{\varepsilon}{5\kappa\ell \left\|\mathbf{v}_k\right\|_2}, \frac{1}{10\kappa\ell}\right),\ \lambda = \frac{2}{7\ell},\ q = \lceil \kappa^{-1} n^{1/2} \rceil,$$

$$S_2 = \left\lceil \frac{3687}{76}\kappa q \right\rceil,\ K = \left\lceil \frac{100\kappa\ell\varepsilon^{-2}\Delta_f}{9} \right\rceil,\ \textit{and } m = \lceil 1024\kappa \rceil.$$

Algorithm 5 outputs $\hat{\mathbf{x}}$ such that $\mathbb{E}\left\|\nabla\Phi(\hat{\mathbf{x}})\right\|_2 \leq 1504\varepsilon$ with $\mathcal{O}\left(n\log(\kappa/\varepsilon) + \kappa^2 n^{1/2}\varepsilon^{-2}\right)$ stochastic gradient evaluations.

In the case of $n \leq \kappa^2$, we set the parameters

$$\zeta = \kappa^{-2}\varepsilon^2,\ \eta_k = \min\left(\frac{\varepsilon}{5\kappa\ell \left\|\mathbf{v}_k\right\|_2}, \frac{1}{10\kappa\ell}\right),\ \lambda = \frac{1}{8\ell},\ q = 1,$$

$$S_2 = 1,\ K = \left\lceil \frac{100\kappa\ell\varepsilon^{-2}\Delta_f}{9} \right\rceil,\ \textit{and } m = \lceil 1024\kappa \rceil.$$

[1] PSARAH extends SARAH [30] to constrained case, which requires $\mathcal{O}\left((n+\kappa)\log(\kappa/\varepsilon)\right)$ stochastic gradient evaluation to achieve sufficient accuracy for our initialization. Please see Appendix E.1 for details

Algorithm 5 outputs $\hat{\mathbf{x}}$ such that $\mathbb{E}\left\|\nabla\Phi(\hat{\mathbf{x}})\right\|_2 \leq 1504\varepsilon$ with $\mathcal{O}\left((\kappa^2+\kappa n)\varepsilon^{-2}\right)$ stochastic gradient evaluations.

5 Sketch of Proofs

We present the briefly overview of the proof of Theorem 1. The details are shown in appendix. Different from Lin et al.'s analysis of SGDA [25] which directly considered the value of $\Phi(\mathbf{x}_k)$ and the distance $\|\mathbf{y}_k - \mathbf{y}^*(\mathbf{x}_k)\|_2$, our proof mainly depends on $f(\mathbf{x}_k, \mathbf{y}_k)$ and its gradient. We split the change of objective functions after one iteration on $(\mathbf{x}_k, \mathbf{y}_k)$ into A_k and B_k as follows

$$f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - f(\mathbf{x}_k, \mathbf{y}_k) = \underbrace{f(\mathbf{x}_{k+1}, \mathbf{y}_k) - f(\mathbf{x}_k, \mathbf{y}_k)}_{A_k} + \underbrace{f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - f(\mathbf{x}_{k+1}, \mathbf{y}_k)}_{B_k}, \quad (4)$$

where A_k provides the decrease of function value f and B_k can characterize the difference between $f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})$ and $\Phi(\mathbf{x}_{k+1})$. We can show that $\mathbb{E}[A_k] \leq -\mathcal{O}(\kappa^{-1}\varepsilon)$ and $\mathbb{E}[B_k] \leq \mathcal{O}(\kappa^{-1}\varepsilon^2/\ell)$. By taking the average of (4) over $k = 0, \ldots, K$, we obtain

$$\frac{1}{K}\sum_{k=0}^{K-1} \mathbb{E}\left\|\mathbf{v}_k\right\|_2 \leq \mathcal{O}(\varepsilon).$$

We can also approximate $\mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k)\right\|_2$ by $\mathbb{E}\left\|\mathbf{v}_k\right\|_2$ with $\mathcal{O}(\varepsilon)$ estimate error. Then the output $\hat{\mathbf{x}}$ of Algorithm 3 satisfies $\mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k)\right\|_2 \leq \mathcal{O}(\varepsilon)$. Based on the discussion in Section 4.2, the number of stochastic gradient evaluation is $\mathcal{O}(\kappa^3\varepsilon^{-3})$. We can also use similar idea to prove Theorem 2.

6 Numerical Experiments

We conduct the experiments by using distributionally robust optimization with nonconvex regularized logistic loss [5, 14, 21, 47]. Given dataset $\{(\mathbf{a}_i, b_i)\}_{i=1}^n$ where $\mathbf{a}_i \in \mathbb{R}^d$ is the feature of i-th sample and $b_i \in \{1, -1\}$ the corresponding label, the minimax formulation is:

$$\min_{\mathbf{x}\in\mathbb{R}^d} \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}, \mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^{n}\Big(y_i l_i(\mathbf{x}) - V(\mathbf{y}) + g(\mathbf{x})\Big),$$

$l_i(\mathbf{x}) = \log(1 + \exp(-b_i\mathbf{a}_i^\top\mathbf{x}))$, g is the nonconvex regularizer [5]:

$$g(\mathbf{x}) = \lambda_2 \sum_{i=1}^{d} \frac{\alpha x_i^2}{1+\alpha x_i^2},$$

$V(\mathbf{y}) = \frac{1}{2}\lambda_1 \left\|n\mathbf{y} - \mathbf{1}\right\|_2^2$ and $\mathcal{Y} = \{\mathbf{y} \in \mathbb{R}^n : 0 \leq y_i \leq 1, \sum_{i=1}^n y_i = 1\}$ is a simplex. Following Yan et al. [47], Kohler and Lucchi [21]'s settings, we let $\lambda_1 = 1/n^2$, $\lambda_2 = 10^{-3}$ and $\alpha = 10$ for experiments.

We evaluate compared the performance of SREDA with baseline algorithms GDAmax, GDA, SGDA [25] and Minimax PPA [26] on six real-world data sets "a9a", "w8a", "gisette", "mushrooms", "sido0" and "rcv1", whose details are listed in Table 2. The dataset "sido0" comes from Causality Workbench[2] and the others can be downloaded from LIBSVM repository[3]. Our experiments are conducted on a workstation with Intel Xeon Gold 5120 CPU and 256GB memory. We use MATLAB 2018a to run the code and the operating system is Ubuntu 18.04.4 LTS.

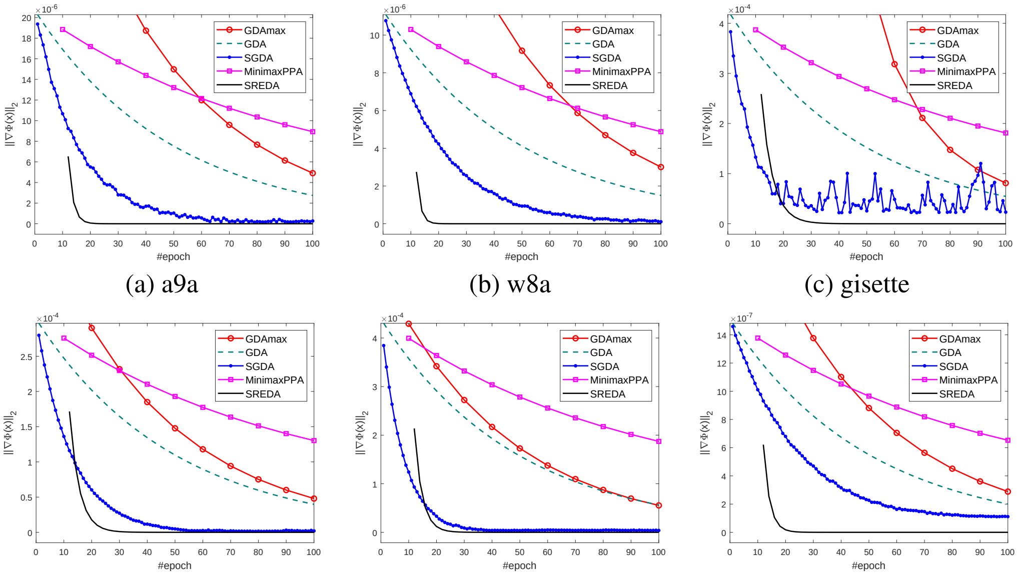
The parameters of the algorithms are chosen as follows: The stepsizes of all algorithms are tuned from $\{10^{-3}, 10^{-2}, 10^{-1}, 1\}$ and we keep the stepsize ratio is $\{10, 10^2, 10^3\}$. For stochastic algorithms SGDA and SREDA, the mini-batch size is set with $\{10, 100, 200\}$. For SREDA, we use the finite-sum version (Algorithm 5 with the first case of Theorem 2) and let $q = m = \lceil n/S_2 \rceil$ heuristically. The initialization of SREDA is based on PSARAH with $K_0 = 5$, $b = 1$ and $m = 20$. For Minimax PPA, we tune the proximal parameter from $\{1, 10, 100\}$ and momentum parameter from $\{0.2, 0.5, 0.7\}$. Each inner loop of Minimax PPA has five times Maximin-AG2 which contains five AGD iterations. The results are shown in Figure 1. It is clear that SREDA converges faster than the baseline algorithms.

[2] `https://www.causality.inf.ethz.ch/challenge.php?page=datasets`

[3] `https://www.csie.ntu.edu.tw/~cjlin/libsvmtools/datasets/`

datasets	n	d
a9a	32,561	123
w8a	49,749	300
gisette	6,000	5,000
mushrooms	8,124	112
sido0	12,678	4,932
rcv1	20,242	47,236

Table 2: Summary of datasets used in our experiments



(a) a9a

(b) w8a

(c) gisette

(d) mushrooms

(e) sido0

(f) rcv1

Figure 1: We demonstrate $\|\nabla\Phi(\mathbf{x})\|_2$ vs. the number of epochs for DRO model on real-world datasets "a9a", "w8a", "gisette", "mushrooms", "sido0" and "rcv1" with SREDA and baseline algorithms.

7 Conclusion

In this paper, we studied stochastic nonconvex-strongly-concave minimax problems. We proposed a novel algorithm called Stochastic Recursive gradiEnt Descent Ascent (SREDA). The algorithm employs variance reduction to solve minimax problems. Based on the appropriate choice of the parameters, we prove SREDA finds an $\mathcal{O}(\varepsilon)$-stationary point of Φ with a stochastic gradient complexity of $\mathcal{O}(\kappa^3\varepsilon^{-3})$. This result is better than state-of-the-art algorithms and optimal in its dependency on ε. We can also apply SREDA to the finite-sum case, and show that it performs well when n is larger than κ^2.

There are still some open problems left. The complexity of SREDA is optimal with respect to ε, but weather it is optimal with respect to κ is unknown. It is also possible to employ SREDA to reduce the complexity of stochastic nonconvex-concave minimax problems without the strongly-concave assumption.

Broader Impact

This paper studied the theory of stochastic minimax optimization. The proposed method SREDA is the first stochastic algorithm which attains the optimal dependency on ε. This observation help us to understand the minimax optimization without convex-concave assumption. It is interesting to apply SREDA to more machine learning applications in future.

Acknowledgments and Disclosure of Funding

The authors would like to thank Min Tao and Jiahao Xie to point out that the first version of this paper on arXiv has a mistake in the original proof of Theorem 1. This work is supported by GRF 16201320 and the project of Shenzhen Research Institute of Big Data (named "Automated Machine Learning").

References

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

[2] Zeyuan Allen-Zhu. Natasha: Faster non-convex stochastic optimization via strongly non-convex parameter. In *ICML*, 2017.

[3] Zeyuan Allen-Zhu. Katyusha X: Practical momentum method for stochastic sum-of-nonconvex optimization. In *ICML*, 2018.

[4] Zeyuan Allen-Zhu and Elad Hazan. Variance reduction for faster non-convex optimization. In *ICML*, 2016.

[5] Anestis Antoniadis, Irène Gijbels, and Mila Nikolova. Penalized likelihood regression for generalized linear models with non-quadratic penalties. *Annals of the Institute of Statistical Mathematics*, 63(3):585–615, 2011.

[6] Yossi Arjevani, Yair Carmon, John C. Duchi, Dylan J. Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *arXiv preprint:1912.02365*, 2019.

[7] Babak Barazandeh and Meisam Razaviyayn. Solving non-convex non-differentiable min-max games using proximal gradient method. In *ICASSP*, 2020.

[8] Antonin Chambolle and Thomas Pock. A first-order primal-dual algorithm for convex problems with applications to imaging. *Journal of mathematical imaging and vision*, 40(1):120–145, 2011.

[9] Tatjana Chavdarova, Gauthier Gidel, François Fleuret, and Simon Lacoste-Julien. Reducing noise in GAN training with variance reduced extragradient. In *NeurIPS*, 2019.

[10] Aaron Defazio. A simple practical accelerated method for finite sums. In *NIPS*, 2016.

[11] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien. SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives. In *NIPS*, 2014.

[12] Simon S. Du and Wei Hu. Linear convergence of the primal-dual gradient method for convex-concave saddle point problems without strong convexity. In *AISTATS*, 2018.

[13] Simon S. Du, Jianshu Chen, Lihong Li, Lin Xiao, and Dengyong Zhou. Stochastic variance reduction methods for policy evaluation. In *ICML*, 2017.

[14] John C. Duchi and Hongseok Namkoong. Variance-based regularization with convex objectives. *Journal of Machine Learning Research*, 20(68):1–55, 2019.

[15] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. SPIDER: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *NeurIPS*, 2018.

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, 2014.

[17] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint:1412.6572*, 2014.

[18] Yu-Guan Hsieh, Franck Iutzeler, Jérôme Malick, and Panayotis Mertikopoulos. On the convergence of single-call stochastic extra-gradient methods. In *NeurIPS*, 2019.

[19] Chi Jin, Praneeth Netrapalli, and Michael I. Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? In *ICML*, 2020.

[20] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS*, 2013.

[21] Jonas Moritz Kohler and Aurelien Lucchi. Sub-sampled cubic regularization for non-convex optimization. In *ICML*, 2017.

[22] Lihua Lei, Cheng Ju, Jianbo Chen, and Michael I. Jordan. Non-convex finite-sum optimization via SCSG methods. In *NIPS*, 2017.

[23] Zhize Li and Jian Li. A simple proximal stochastic gradient method for nonsmooth nonconvex optimization. In *NeurIPS*, 2018.

[24] Hongzhou Lin, Julien Mairal, and Zaid Harchaoui. Catalyst acceleration for first-order convex optimization: from theory to practice. *Journal of Machine Learning Research*, 18(212):1–54, 2018.

[25] Tianyi Lin, Chi Jin, and Michael I. Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *ICML*, 2020.

[26] Tianyi Lin, Chi Jin, and Michael I. Jordan. Near-optimal algorithms for minimax optimization. In *COLT*, 2020.

[27] Songtao Lu, Ioannis Tsaknakis, Mingyi Hong, and Yongxin Chen. Hybrid block successive approximation for one-sided non-convex min-max problems: algorithms and applications. *arXiv preprint:1902.08294*, 2019.

[28] Luo Luo, Cheng Chen, Yujun Li, Guangzeng Xie, and Zhihua Zhang. A stochastic proximal point algorithm for saddle-point problems. *arXiv preprint:1909.06946*, 2019.

[29] Yurii Nesterov. *Lectures on convex optimization*, volume 137. Springer, 2018.

[30] Lam M. Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. SARAH: A novel method for machine learning problems using stochastic recursive gradient. In *ICML*, 2017.

[31] Lam M. Nguyen, Katya Scheinberg, and Martin Takáč. Inexact SARAH algorithm for stochastic optimization. *arXiv preprint:1811.10105*, 2018.

[32] Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D. Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. In *NeurIPS*, 2019.

[33] Dmitrii M. Ostrovskii, Andrew Lowy, and Meisam Razaviyayn. Efficient search of first-order Nash equilibria in nonconvex-concave smooth min-max problems. *arXiv preprint:2002.07919*, 2020.

[34] Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *arXiv preprint:1808.02901*, 2018.

[35] Balamurugan Palaniappan and Francis Bach. Stochastic variance reduction methods for saddle-point problems. In *NIPS*, 2016.

[36] Nhan H. Pham, Lam M. Nguyen, Dzung T. Phan, and Quoc Tran-Dinh. ProxSARAH: An efficient algorithmic framework for stochastic composite nonconvex optimization. *arXiv preprint:1902.05679*, 2019.

[37] Hassan Rafique, Mingrui Liu, Qihang Lin, and Tianbao Yang. Non-convex min-max optimization: Provable algorithms and applications in machine learning. *arXiv preprint:1810.02060*, 2018.

[38] Sashank J. Reddi, Ahmed Hefny, Suvrit Sra, Barnabas Poczos, and Alex Smola. Stochastic variance reduction for nonconvex optimization. In *ICML*, 2016.

[39] Mark Schmidt, Nicolas Le Roux, and Francis Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2):83–112, 2017.

[40] Zebang Shen, Aryan Mokhtari, Tengfei Zhou, Peilin Zhao, and Hui Qian. Towards more efficient stochastic decentralized learning: Faster convergence and sparse communication. In *ICML*, 2018.

[41] Aman Sinha, Hongseok Namkoong, and John Duchi. Certifying some distributional robustness with principled adversarial training. In *ICLR*, 2018.

[42] Conghui Tan, Tong Zhang, Shiqian Ma, and Ji Liu. Stochastic primal-dual method for empirical risk minimization with O(1) per-iteration complexity. In *NeurIPS*, 2018.

[43] Kiran K. Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Efficient algorithms for smooth minimax optimization. In *NeurIPS*, 2019.

[44] Hoi-To Wai, Zhuoran Yang, Zhaoran Wang, and Mingyi Hong. Multi-agent reinforcement learning via double averaging primal-dual optimization. In *NeurIPS*, 2018.

[45] Zhe Wang, Kaiyi Ji, Yi Zhou, Yingbin Liang, and Vahid Tarokh. SpiderBoost and momentum: Faster variance reduction algorithms. In *NeurIPS*, 2019.

[46] Guangzeng Xie, Luo Luo, Yijiang Lian, and Zhihua Zhang. Lower complexity bounds for finite-sum convex-concave minimax optimization problems. In *ICML*, 2020.

[47] Yan Yan, Yi Xu, Qihang Lin, Lijun Zhang, and Tianbao Yang. Stochastic primal-dual algorithms with faster convergence than $O(1/\sqrt{t})$ for problems without bilinear structure. *arXiv preprint arXiv:1904.10112*, 2019.

[48] Yan Yan, Yi Xu, Qihang Lin, Wei Liu, and Tianbao Yang. Sharp analysis of epoch stochastic gradient descent ascent methods for min-max optimization. In *NeurIPS*, 2020.

[49] Yiming Ying, Longyin Wen, and Siwei Lyu. Stochastic online AUC maximization. In *NIPS*, 2016.

[50] Junyu Zhang and Lin Xiao. Multi-level composite stochastic optimization via nested variance reduction. *arXiv preprint:1908.11468*, 2019.

[51] Junyu Zhang, Mingyi Hong, and Shuzhong Zhang. On lower iteration complexity bounds for the saddle point problems. *arXiv preprint:1912.07481*, 2019.

[52] Lijun Zhang, Mehrdad Mahdavi, and Rong Jin. Linear convergence with condition number independent access of full gradients. In *NIPS*, 2013.

[53] Yuchen Zhang and Lin Xiao. Stochastic primal-dual coordinate method for regularized empirical risk minimization. *The Journal of Machine Learning Research*, 18(1):2939–2980, 2017.

[54] Dongruo Zhou, Pan Xu, and Quanquan Gu. Stochastic nested variance reduction for nonconvex optimization. In *NeurIPS*, 2018.

Supplementary Materials

This supplementary materials are organized as follows. Appendix A provide several technique lemmas for later analysis. Appendix B give some properties for our concave maximizer. Then, Appendix C proposes projected inexact SARAH (PiSARAH), which generalizes SARAH to constrained optimization and we use it for the initialization of SREDA. Appendix D presents the proof of our main results Theorem 1 and we extend it to prove finite-sum case Theorem 2 in Appendix E.

A Technical Tools

We first present some useful inequalities in convex optimization, martingale variance bound and gradient mapping.

Lemma 2 ([29, Theorem 2.1.5 and 2.1.12])**.** *Suppose $g(\cdot)$ is μ-strongly convex and has ℓ-Lipschitz gradient. Let $\mathbf{w}^*$ be the minimizer of g. Then for any $\mathbf{w}$ and $\mathbf{w}'$, we have the following inequalities*

$$\langle \nabla g(\mathbf{w}) - \nabla g(\mathbf{w}'), \mathbf{w} - \mathbf{w}' \rangle \geq \frac{1}{\ell} \left\| \nabla g(\mathbf{w}) - \nabla g(\mathbf{w}') \right\|_2^2, \tag{5}$$

$$\langle \nabla g(\mathbf{w}) - \nabla g(\mathbf{w}'), \mathbf{w} - \mathbf{w}' \rangle \geq \frac{\mu\ell}{\mu+\ell} \left\| \mathbf{w} - \mathbf{w}' \right\|_2^2 + \frac{1}{\mu+\ell} \left\| \nabla g(\mathbf{w}) - \nabla g(\mathbf{w}') \right\|_2^2, \tag{6}$$

Lemma 3 ([15, Lemma 1])**.** *Let $\mathcal{V}_k$ be estimator of $\mathcal{B}(\mathbf{z}_k)$ as*

$$\mathcal{V}_k = \mathcal{B}_{\mathcal{S}_*}(\mathbf{z}_k) - \mathcal{B}_{\mathcal{S}_*}(\mathbf{z}_{k-1}) + \mathcal{V}_{k-1},$$

where $\mathcal{B}_{\mathcal{S}_*} = \frac{1}{|\mathcal{S}_*|} \sum_{\mathcal{B}_i \in \mathcal{S}_*} \mathcal{B}_i$ satisfies

$$\mathbb{E}\left[\mathcal{B}_i(\mathbf{z}_k) - \mathcal{B}_i(\mathbf{z}_{k-1}) \mid \mathbf{z}_0, \dots, \mathbf{z}_{k-1}\right] = \mathbb{E}\left[\mathcal{V}_k - \mathcal{V}_{k-1} \mid \mathbf{z}_0, \dots, \mathbf{z}_{k-1}\right],$$

and $\mathcal{B}_i$ is L-Lipschitz continuous for any $\mathcal{B}_i \in \mathcal{S}_*$. Then for all $k = 1, \dots, K$, we have

$$\mathbb{E} \left\| \mathcal{V}_k - \mathcal{B}(\mathbf{z}_k) \mid \mathbf{z}_0, \dots, \mathbf{z}_{k-1} \right\|_2^2 \leq \left\| \mathcal{V}_{k-1} - \mathcal{B}(\mathbf{z}_{k-1}) \right\|_2^2 + \frac{\ell^2}{|\mathcal{S}_*|} \mathbb{E}\left[\left\| \mathbf{z}_k - \mathbf{z}_{k-1} \right\|_2^2 \mid \mathbf{z}_0, \dots, \mathbf{z}_{k-1} \right].$$

Lemma 4 ([29, Corollary 2.2.3])**.** *Given convex and compact set $\mathcal{C} \subseteq \mathbb{R}^d$ and any $\mathbf{w}, \mathbf{w}' \in \mathbb{R}^d$, we have $\left\| \Pi_{\mathcal{C}}(\mathbf{w}) - \Pi_{\mathcal{C}}(\mathbf{w}') \right\|_2 \leq \left\| \mathbf{w} - \mathbf{w}' \right\|_2$.*

Lemma 5 ([29, Corollary 2.2.4])**.** *Let g has ℓ-Lipschitz gradient and μ-strongly convex, and $\mathcal{C}$ is a convex set. Denote $\mathcal{G}_\gamma$ be the gradient mapping such that*

$$\mathcal{G}_\gamma(\mathbf{w}) = \frac{\mathbf{w} - \Pi_{\mathcal{C}}(\mathbf{w} - \gamma \nabla g(\mathbf{w}))}{\gamma},$$

where $\gamma \leq 1/\ell$. Let $\mathbf{w}^* = \arg\min_{\mathbf{w} \in \mathcal{C}} g(\mathbf{w})$, then we have

$$\langle \mathcal{G}_\gamma(\mathbf{w}), \mathbf{w} - \mathbf{w}^* \rangle^2 \geq \frac{\mu}{2} \left\| \mathbf{w} - \mathbf{w}^* \right\|_2 + \frac{1}{2\ell} \left\| \mathcal{G}_\gamma(\mathbf{w}) \right\|_2^2$$

Corollary 1. *Under assumptions of Lemma 5, we have $\frac{\mu}{2} \left\| \mathbf{w} - \mathbf{w}^* \right\|_2 \leq \left\| \mathcal{G}_\gamma(\mathbf{w}) \right\|_2$.*

Proof. We have

$$\begin{aligned} \frac{\mu}{2} \left\| \mathbf{w} - \mathbf{w}^* \right\|_2^2 \leq & \langle \mathcal{G}_\gamma(\mathbf{w}), \mathbf{w} - \mathbf{w}^* \rangle - \frac{1}{2\ell} \left\| \mathcal{G}_\gamma(\mathbf{w}) \right\|_2^2 \\ \leq & \langle \mathcal{G}_\gamma(\mathbf{w}), \mathbf{w} - \mathbf{w}^* \rangle \\ \leq & \left\| \mathcal{G}_\gamma(\mathbf{w}) \right\|_2 \left\| \mathbf{w} - \mathbf{w}^* \right\|_2, \end{aligned}$$

where the first inequality use Lemma 5 and the last one is based on Cauchy-Schwarz inequality. Then we obtain the desired result. □

B Some Results of Concave Maximizer

In this section, we present some results of concave maximizer Algorithm 4. The analysis of SREDA is based on the following two auxiliary quantities:

$$\Delta_k = \mathbb{E}\left[\|\mathbf{v}_k - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\|_2^2 + \|\mathbf{u}_k - \nabla_{\mathbf{y}} f(\mathbf{x}_k, \mathbf{y}_k)\|_2^2\right] \text{ and } \delta_k = \mathbb{E}\,\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \mathbf{y}_k)\|_2^2 .$$

The main target is to prove both Δ_k and δ_k can be bounded by $\mathcal{O}(\kappa^{-2}\varepsilon^2)$.

Using the notations of Algorithm 3 and 4, we denote $\tilde{\mathbf{y}}_k^* = \arg\min_{\mathbf{y}\in\mathcal{Y}} g_k(\mathbf{y})$ and $\hat{\mathbf{u}}_{k,t} = -\tilde{\mathbf{u}}_{k,t}$. It is obvious that $g_k(\cdot)$ is μ-strongly convex and has ℓ-Lipschitz gradient. The update rule of $\mathbf{x}_k$ in Algorithm 3 means for any $k \geq 0$, we have

$$\|\mathbf{x}_{k+1} - \mathbf{x}_k\|_2^2 \leq \varepsilon_{\mathbf{x}}^2,$$

where $\varepsilon_{\mathbf{x}}^2$ is defined as $\frac{1}{25}\kappa^{-2}\varepsilon^2$. We also denote the gradient mapping with respect to $\mathbf{y}$ as

$$\tilde{\mathcal{G}}_{\lambda,k}(\mathbf{y}) \triangleq \frac{\mathbf{y} - \Pi_{\mathcal{Y}}(\mathbf{y} - \lambda \nabla g_k(\mathbf{y}))}{\lambda} = \mathcal{G}_{\lambda}(\mathbf{x}_{k+1}, \mathbf{y}).$$

We first introduce some lemmas for our iteration and gradient mapping.

Lemma 6 (Lemma 1 of [23])**.** *Let* $\mathbf{y}^+ := \Pi_{\mathcal{Y}}(\mathbf{y} - \lambda\mathbf{u})$*, then for all* $\mathbf{z}$*, we have*

$$g_k(\mathbf{y}^+) \leq g_k(\mathbf{z}) + \langle \nabla g_k(\mathbf{y}) - \mathbf{u}, \mathbf{y}^+ - \mathbf{z}\rangle - \frac{\langle \mathbf{y}^+ - \mathbf{y}, \mathbf{y}^+ - \mathbf{z}\rangle}{\lambda} + \frac{\ell}{2}\left\|\mathbf{y}^+ - \mathbf{y}\right\|_2^2 + \frac{\ell}{2}\left\|\mathbf{z} - \mathbf{y}\right\|_2^2 .$$

Lemma 7 (Lemma 2 of [23])**.** *Let* $\mathbf{y}^+ := \Pi_{\mathcal{Y}}(\mathbf{y} - \lambda\mathbf{u})$ *and* $\overline{\mathbf{y}_{k,t}} := \Pi_{\mathcal{Y}}\big(\mathbf{y} - \lambda\nabla g_k(\mathbf{y})\big)$*, then we have* $\langle \nabla g_k(\mathbf{y}) - \mathbf{u}, \mathbf{y}^+ - \overline{\mathbf{y}_{k,t}}\rangle \leq \lambda\left\|\nabla g_k(\mathbf{y}) - \mathbf{u}\right\|_2^2$.

Lemma 8. *For Algorithm 3 and any* $\mathbf{y}$*, we have* $\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \mathbf{y}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \mathbf{y})\|_2^2 \leq \ell^2\varepsilon_{\mathbf{x}}^2$.

Proof. Using Lemma 4 and smoothness of f, we have

$$\begin{aligned}
&\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \mathbf{y}_k) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \mathbf{y}_k)\|_2^2 \\
=&\left\|\frac{\mathbf{y}_k - \Pi_{\mathcal{Y}}(\mathbf{y}_k - \lambda\nabla_{\mathbf{y}} f(\mathbf{x}_{k+1}, \mathbf{y}_k))}{\lambda} - \frac{\mathbf{y}_k - \Pi_{\mathcal{Y}}(\mathbf{y}_k - \lambda\nabla_{\mathbf{y}} f(\mathbf{x}_k, \mathbf{y}_k))}{\lambda}\right\|_2^2 \\
\leq&\frac{1}{\lambda^2}\left\|\Pi_{\mathcal{Y}}(\mathbf{y}_k - \lambda\nabla_{\mathbf{y}} f(\mathbf{x}_{k+1}, \mathbf{y}_k)) - \Pi_{\mathcal{Y}}(\mathbf{y}_k - \lambda\nabla_{\mathbf{y}} f(\mathbf{x}_k, \mathbf{y}_k))\right\|_2^2 \\
\leq&\left\|\nabla_{\mathbf{y}} f(\mathbf{x}_{k+1}, \mathbf{y}_k) - \nabla_{\mathbf{y}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2^2 \leq \ell^2\varepsilon_{\mathbf{x}}^2.
\end{aligned}$$

□

Lemma 9. *Let* $\mathbf{y}^+ = \Pi_{\mathcal{Y}}(\mathbf{y} - \lambda\mathbf{u})$ *and* $\lambda < 1/\ell$*, then we have*

$$g_k(\mathbf{y}^+) - g_k(\mathbf{y}^*) \leq \frac{1}{\mu}\left(\|\tilde{\mathcal{G}}_{\lambda,k}(\mathbf{y})\|_2^2 + \|\nabla g_k(\mathbf{y}) - \mathbf{u}\|_2^2\right)$$

for any $\mathbf{y}^* \in \mathcal{Y}$.

Proof. Let $Q(\mathbf{z}) = f(\mathbf{y}) + \langle \mathbf{u}, \mathbf{z} - \mathbf{y}\rangle + \frac{1}{2\lambda}\left\|\mathbf{z} - \mathbf{y}\right\|_2^2 + r(\mathbf{z})$. We have $\mathbf{y}^+ = \arg\min_{\mathbf{z}} Q(\mathbf{z})$ and

$$\langle \mathbf{u} + \lambda^{-1}(\mathbf{y}^+ - \mathbf{y}) + \boldsymbol{\xi}, \mathbf{y}^+ - \mathbf{y}^*\rangle = \langle \nabla Q(\mathbf{y}^+), \mathbf{y}^* - \mathbf{y}^+\rangle \geq 0,$$

for any $\mathbf{y}^* \in \mathcal{Y}$. Then

$$\begin{aligned}
&\langle \nabla g_k(\mathbf{y}^+), \mathbf{y}^* - \mathbf{y}^+\rangle \\
=&\langle \nabla g_k(\mathbf{y}^+) - \nabla g_k(\mathbf{y}), \mathbf{y}^* - \mathbf{y}^+\rangle + \langle \nabla g_k(\mathbf{y}), \mathbf{y}^* - \mathbf{y}^+\rangle \\
=&\langle \nabla g_k(\mathbf{y}^+) - \nabla g_k(\mathbf{y}), \mathbf{y}^* - \mathbf{y}^+\rangle + \langle \nabla g_k(\mathbf{y}) - \mathbf{u} + \lambda^{-1}(\mathbf{y} - \mathbf{y}^+), \mathbf{y}^* - \mathbf{y}^+\rangle \\
=&\langle \nabla \tilde{g}_k(\mathbf{y}^+) - \nabla \tilde{g}_k(\mathbf{y}), \mathbf{y}^* - \mathbf{y}^+\rangle + \langle \nabla g_k(\mathbf{y}) - \mathbf{u}, \mathbf{y}^* - \mathbf{y}^+\rangle \\
\geq&-\max(\ell, \lambda^{-1})\left\|\mathbf{y}^+ - \mathbf{y}\right\|_2\left\|\mathbf{y}^* - \mathbf{y}^+\right\|_2 + \langle \nabla g_k(\mathbf{y}) - \mathbf{u}, \mathbf{y}^* - \mathbf{y}^+\rangle
\end{aligned}$$

$$\geq -\max(\ell,\lambda^{-1})\left\|\mathbf{y}^{+}-\mathbf{y}\right\|_2\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2-\|\nabla g_k(\mathbf{y})-\mathbf{u}\|_2\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2,$$

where $\tilde{g}_k(\mathbf{y})=g_k(\mathbf{y})-\frac{1}{2\lambda}\|\mathbf{y}\|_2^2$. The first inequality is due to $\tilde{g}_k$ is at most $\max(\ell,\lambda^{-1})$-smooth, that is

$$-\lambda^{-1}\mathbf{I}\preceq(\mu-\lambda^{-1})\mathbf{I}\preceq\nabla^2\tilde{g}_k(\mathbf{y})\preceq(\ell-\lambda^{-1})\mathbf{I}\preceq\ell\mathbf{I}.$$

Consequently, we have

$$\begin{aligned}
&-\max(\ell,\lambda^{-1})\left\|\mathbf{y}^{+}-\mathbf{y}\right\|_2\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2\\
\leq&\langle\nabla g_k(\mathbf{y}^{+}),\mathbf{y}^{*}-\mathbf{y}^{+}\rangle+\|\nabla g_k(\mathbf{y})-\mathbf{u}\|_2\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2\\
\leq&g_k(\mathbf{y}^{*})-g_k(\mathbf{y}^{+})-\frac{\mu}{2}\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2^2+\|\nabla g_k(\mathbf{y})-\mathbf{u}\|_2\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2
\end{aligned}$$

which implies

$$\begin{aligned}
&g_k(\mathbf{y}^{*})-g_k(\mathbf{y}^{+})\\
\geq&\frac{\mu}{2}\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2^2-\max(\ell,\lambda^{-1})\left\|\mathbf{y}^{+}-\mathbf{y}\right\|_2\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2-\|\nabla g_k(\mathbf{y})-\mathbf{u}\|_2\left\|\mathbf{y}^{*}-\mathbf{y}^{+}\right\|_2\\
\geq&\inf_{\mathbf{y}}\left\{\frac{\mu}{2}\left\|\mathbf{z}-\mathbf{y}^{+}\right\|_2^2-\left(\max(\ell,\lambda^{-1})\left\|\mathbf{y}^{+}-\mathbf{y}\right\|_2+\|\nabla g_k(\mathbf{y})-\mathbf{u}\|_2\right)\left\|\mathbf{z}-\mathbf{y}^{+}\right\|_2\right\}\\
=&-\frac{1}{2\mu}\left(\max(\ell,\lambda^{-1})\left\|\mathbf{y}^{+}-\mathbf{y}\right\|_2+\|\nabla g_k(\mathbf{y})-\mathbf{u}\|_2\right)^2\\
\geq&-\frac{1}{\mu}\left(\max(\ell,\lambda^{-1})^2\left\|\mathbf{y}^{+}-\mathbf{y}\right\|_2^2+\|\nabla g_k(\mathbf{y})-\mathbf{u}\|_2^2\right).
\end{aligned}$$

Considering that $\eta\leq 1/\ell$, we can obtain the desired result by rearranging above inequality. □

We can now present the key lemma for the concave maximizer, which upper bounds the magnitude of gradient mapping after one epoch iterations on $\mathbf{y}$.

Lemma 10. *For Algorithm 4 with $\lambda=\frac{1}{8\ell}$, we have*

$$\begin{aligned}
&\mathbb{E}\|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,s_k})\|_2^2\\
\leq&\frac{64\ell}{m\mu}\mathbb{E}\|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,0})\|_2^2+\left(\frac{64\ell}{m\mu}+8\right)\mathbb{E}\left\|\nabla g_k(\tilde{\mathbf{y}}_{k,0})-\tilde{\mathbf{u}}_{k,0}\right\|_2^2+8\ell^2\mathbb{E}\left\|\tilde{\mathbf{y}}_{k,1}-\tilde{\mathbf{y}}_{k,0}\right\|_2^2.
\end{aligned}$$

Proof. We define $\overline{\mathbf{y}_{k,t}}:=\Pi_{\mathcal{Y}}\big(\tilde{\mathbf{y}}_{k,t-1}-\lambda\nabla g_k(\tilde{\mathbf{y}}_{k,t-1})\big)$. The procedure of Algorithm 4 means $\tilde{\mathbf{y}}_{k,t}:=\Pi_{\mathcal{Y}}(\tilde{\mathbf{y}}_{k,t-1}-\lambda\hat{\mathbf{u}}_{k,t-1})$. Using Lemma 6 by letting $\mathbf{y}^{+}=\tilde{\mathbf{y}}_{k,t},\mathbf{y}=\tilde{\mathbf{y}}_{k,t-1},\mathbf{u}=\hat{\mathbf{u}}_{k,t-1}$ and $\mathbf{z}=\overline{\mathbf{y}_{k,t}}$, we have

$$\begin{aligned}
g_k(\tilde{\mathbf{y}}_{k,t})\leq&g_k(\overline{\mathbf{y}_{k,t}})+\langle\nabla g_k(\tilde{\mathbf{y}}_{k,t-1})-\tilde{\mathbf{u}}_{k,t-1},\tilde{\mathbf{y}}_{k,t}-\overline{\mathbf{y}_{k,t}}\rangle\\
&-\frac{1}{\lambda}\langle\tilde{\mathbf{y}}_{k,t}-\tilde{\mathbf{y}}_{k,t-1},\tilde{\mathbf{y}}_{k,t}-\overline{\mathbf{y}_{k,t}}\rangle+\frac{\ell}{2}\left\|\tilde{\mathbf{y}}_{k,t}-\tilde{\mathbf{y}}_{k,t-1}\right\|_2^2+\frac{\ell}{2}\left\|\overline{\mathbf{y}_{k,t}}-\tilde{\mathbf{y}}_{k,t-1}\right\|_2^2.
\end{aligned}\tag{7}$$

Using Lemma 6 again by letting $\mathbf{y}^{+}=\overline{\mathbf{y}_{k,t}},\mathbf{y}=\tilde{\mathbf{y}}_{k,t-1},\mathbf{u}=\nabla g_k(\tilde{\mathbf{y}}_{k,t-1})$ and $\mathbf{z}=\mathbf{y}=\tilde{\mathbf{y}}_{k,t-1}$, we have

$$\begin{aligned}
g_k(\overline{\mathbf{y}_{k,t}})\leq&g_k(\tilde{\mathbf{y}}_{k,t-1})-\frac{1}{\lambda}\langle\overline{\mathbf{y}_{k,t}}-\tilde{\mathbf{y}}_{k,t-1},\overline{\mathbf{y}_{k,t}}-\tilde{\mathbf{y}}_{k,t-1}\rangle+\frac{\ell}{2}\left\|\overline{\mathbf{y}_{k,t}}-\tilde{\mathbf{y}}_{k,t-1}\right\|_2^2\\
=&g_k(\mathbf{y}_{t-1})-\left(\frac{1}{\lambda}-\frac{\ell}{2}\right)\left\|\overline{\mathbf{y}_{k,t}}-\tilde{\mathbf{y}}_{k,t-1}\right\|_2^2.
\end{aligned}\tag{8}$$

Sum over inequalities (7) and (8), we have

$$\begin{aligned}
&g_k(\tilde{\mathbf{y}}_{k,t})\\
\leq&g_k(\tilde{\mathbf{y}}_{k,t-1})+\frac{\ell}{2}\left\|\tilde{\mathbf{y}}_{k,t}-\tilde{\mathbf{y}}_{k,t-1}\right\|_2^2-\left(\frac{1}{\lambda}-\ell\right)\left\|\overline{\mathbf{y}_{k,t}}-\tilde{\mathbf{y}}_{k,t-1}\right\|_2^2\\
&+\langle\nabla g_k(\tilde{\mathbf{y}}_{k,t-1})-\hat{\mathbf{u}}_{k,t-1},\tilde{\mathbf{y}}_{k,t}-\overline{\mathbf{y}_{k,t}}\rangle-\frac{1}{\lambda}\langle\tilde{\mathbf{y}}_{k,t}-\tilde{\mathbf{y}}_{k,t-1},\tilde{\mathbf{y}}_{k,t}-\overline{\mathbf{y}_{k,t}}\rangle^2
\end{aligned}$$

$$
\begin{aligned}
&= g_k(\tilde{\mathbf{y}}_{k,t-1}) + \frac{\ell}{2}\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 - \left(\frac{1}{\lambda} - \ell\right)\left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 \\
&\quad + \langle \nabla g_k(\tilde{\mathbf{y}}_{k,t-1}) - \hat{\mathbf{u}}_{k,t-1}, \tilde{\mathbf{y}}_{k,t} - \overline{\mathbf{y}_{k,t}} \rangle \\
&\quad - \frac{1}{2\lambda}\left(\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 + \left\|\tilde{\mathbf{y}}_{k,t} - \overline{\mathbf{y}_{k,t}}\right\|_2^2 - \left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2\right) \\
&= g_k(\tilde{\mathbf{y}}_{k,t-1}) - \left(\frac{1}{2\lambda} - \frac{\ell}{2}\right)\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 - \left(\frac{1}{2\lambda} - \ell\right)\left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 \\
&\quad + \langle \nabla g_k(\tilde{\mathbf{y}}_{k,t-1}) - \hat{\mathbf{u}}_{k,t-1}, \tilde{\mathbf{y}}_{k,t} - \overline{\mathbf{y}_{k,t}} \rangle - \frac{1}{2\lambda}\left\|\tilde{\mathbf{y}}_{k,t} - \overline{\mathbf{y}_{k,t}}\right\|_2^2 \\
&\le g_k(\tilde{\mathbf{y}}_{k,t-1}) - \left(\frac{1}{2\lambda} - \frac{\ell}{2}\right)\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 - \left(\frac{1}{2\lambda} - \ell\right)\left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 \\
&\quad + \langle \nabla g_k(\tilde{\mathbf{y}}_{k,t-1}) - \hat{\mathbf{u}}_{k,t-1}, \tilde{\mathbf{y}}_{k,t} - \overline{\mathbf{y}_{k,t}} \rangle - \frac{1}{8\lambda}\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 + \frac{1}{6\lambda}\left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 \\
&= g_k(\tilde{\mathbf{y}}_{k,t-1}) - \left(\frac{5}{8\lambda} - \frac{\ell}{2}\right)\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 - \left(\frac{1}{3\lambda} - \ell\right)\left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 \\
&\quad + \langle \nabla g_k(\tilde{\mathbf{y}}_{k,t-1}) - \hat{\mathbf{u}}_{k,t-1}, \tilde{\mathbf{y}}_{k,t} - \overline{\mathbf{y}_{k,t}} \rangle \\
&\le g_k(\tilde{\mathbf{y}}_{k,t-1}) - \left(\frac{5}{8\lambda} - \frac{\ell}{2}\right)\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 - \left(\frac{1}{3\lambda} - \ell\right)\left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 \\
&\quad + \lambda\left\|\nabla g_k(\tilde{\mathbf{y}}_{k,t-1}) - \hat{\mathbf{u}}_{k,t-1}\right\|_2^2,
\end{aligned}
$$

where the second inequality uses Young's inequality as follows

$$
\left\|\tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 \le \left(1 + \alpha^{-1}\right)\left\|\overline{\mathbf{y}_{k,t}} - \tilde{\mathbf{y}}_{k,t-1}\right\|_2^2 + (1+\alpha)\left\|\tilde{\mathbf{y}}_{k,t} - \overline{\mathbf{y}_{k,t}}\right\|_2^2 \quad \text{with } \alpha = 3,
$$

and the last inequality holds due to Lemma 7. Take the expectation on above result, we have

$$
\begin{aligned}
&\mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,t+1})] \\
\le& \mathbb{E}\Bigg[g_k(\tilde{\mathbf{y}}_{k,t}) - \left(\frac{1}{2\lambda} - \frac{\ell}{2}\right)\left\|\tilde{\mathbf{y}}_{k,t+1} - \tilde{\mathbf{y}}_{k,t}\right\|_2^2 - \left(\frac{1}{3\lambda} - \ell\right)\left\|\overline{\mathbf{y}_{k,t+1}} - \tilde{\mathbf{y}}_{k,t}\right\|_2^2 \\
&\qquad + \lambda\left\|\nabla g_k(\tilde{\mathbf{y}}_{k,t}) - \hat{\mathbf{u}}_{k,t}\right\|_2^2\Bigg] \\
\le& \mathbb{E}\Bigg[g_k(\tilde{\mathbf{y}}_{k,t}) - \left(\frac{1}{2\lambda} - \frac{\ell}{2}\right)\left\|\tilde{\mathbf{y}}_{k,t+1} - \tilde{\mathbf{y}}_{k,t}\right\|_2^2 - \left(\frac{1}{3\lambda} - \ell\right)\left\|\overline{\mathbf{y}_{k,t+1}} - \tilde{\mathbf{y}}_{k,t}\right\|_2^2 \\
&\qquad + \lambda\left(\left\|\nabla g_k(\tilde{\mathbf{y}}_{k,0}) - \tilde{\mathbf{u}}_{k,0}\right\|_2^2 + \frac{\ell^2}{S_2}\sum_{i=0}^{t-1}\left\|\tilde{\mathbf{y}}_{k,i+1} - \tilde{\mathbf{y}}_{k,i}\right\|_2^2\right)\Bigg],
\end{aligned} \tag{9}
$$

where the second inequality is based on Lemma 3. Summing over (9) with t from 1 to m and relax the upper bound of i to m, we obtain

$$
\begin{aligned}
&\mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,m})] \\
\le& \mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,1})] - \sum_{i=1}^{m}\left(\frac{1}{2\lambda} - \frac{\ell}{2} - \frac{\lambda\ell^2 m}{S_2}\right)\mathbb{E}\left\|\tilde{\mathbf{y}}_{k,i+1} - \tilde{\mathbf{y}}_{k,i}\right\|_2^2 \\
&- \left(\frac{1}{3\lambda} - \ell\right)\sum_{i=1}^{m}\mathbb{E}\left\|\overline{\mathbf{y}_{k,i+1}} - \tilde{\mathbf{y}}_{k,i}\right\|_2^2 + m\lambda\left\|\nabla g_k(\tilde{\mathbf{y}}_{k,0}) - \hat{\mathbf{u}}_{k,0}\right\|_2^2 + \frac{\lambda\ell^2 m}{S_2}\mathbb{E}\left\|\tilde{\mathbf{y}}_{k,1} - \tilde{\mathbf{y}}_{k,0}\right\|_2^2
\end{aligned}
$$

Consider that $\lambda = \frac{1}{8\ell}$, we further obtain that

$$
\begin{aligned}
&\mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,m})] \\
\le& \mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,1})] - 3\ell\sum_{i=1}^{m}\mathbb{E}\left\|\tilde{\mathbf{y}}_{k,i+1} - \tilde{\mathbf{y}}_{k,i}\right\|_2^2 - \ell\lambda^2\sum_{i=1}^{m}\mathbb{E}\|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,i})\|_2^2
\end{aligned}
$$

$$
\begin{aligned}
&+ m\lambda \left\| \nabla g_k(\tilde{\mathbf{y}}_{k,1}) - \hat{\mathbf{u}}_{k,0} \right\|_2^2 + \frac{\lambda \ell^2 m}{S_2} \mathbb{E} \left\| \tilde{\mathbf{y}}_{k,1} - \tilde{\mathbf{y}}_{k,0} \right\|_2^2 \\
\leq & \mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,1})] - \ell \lambda^2 \sum_{i=1}^m \mathbb{E} \|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,i})\|_2^2 + m\lambda \left\| \nabla g_k(\tilde{\mathbf{y}}_{k,1}) - \hat{\mathbf{u}}_{k,0} \right\|_2^2 + \frac{\lambda \ell^2 m}{S_2} \mathbb{E} \left\| \tilde{\mathbf{y}}_{k,1} - \tilde{\mathbf{y}}_{k,0} \right\|_2^2
\end{aligned}
$$

Let $\tilde{\mathbf{y}}_k^* = \arg\min_{\mathbf{y} \in \mathcal{Y}} g_k(\mathbf{y})$, then the above inequality further implies that

$$
\sum_{i=1}^m \mathbb{E} \|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,i})\|_2^2 \leq 64\ell \left(\mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,1}) - g_k(\tilde{\mathbf{y}}_k^*)] \right) + 8m \left\| \nabla g_k(\tilde{\mathbf{y}}_{k,0}) - \hat{\mathbf{u}}_{k,0} \right\|_2^2 .
$$

Since $\mathbf{y}_{k+1} = \tilde{\mathbf{y}}_{k,s_k}$ and s_k is sampled from $\{1, \dots, m\}$, we have

$$
\begin{aligned}
& \mathbb{E} \|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,s_k})\|_2^2 \\
= & \frac{1}{m} \sum_{i=1}^m \mathbb{E} \|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,i})\|_2^2 \\
\leq & \frac{64\ell}{m} \left(\mathbb{E}[g_k(\tilde{\mathbf{y}}_{k,1}) - g_k(\tilde{\mathbf{y}}_k^*)] \right) + 8\mathbb{E} \left\| \nabla g_k(\tilde{\mathbf{y}}_{k,0}) - \tilde{\mathbf{u}}_{k,0} \right\|_2^2 + \frac{8\ell^2}{S_2} \mathbb{E} \left\| \tilde{\mathbf{y}}_{k,1} - \tilde{\mathbf{y}}_{k,0} \right\|_2^2 \\
\leq & \frac{64\ell}{m\mu} \left(\mathbb{E} \|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,0})\|_2^2 + \left\| \nabla g_k(\tilde{\mathbf{y}}_{k,0}) - \tilde{\mathbf{u}}_{k,0} \right\|_2^2 \right) + 8\mathbb{E} \left\| \nabla g_k(\tilde{\mathbf{y}}_{k,0}) - \tilde{\mathbf{u}}_{k,0} \right\|_2^2 + 8\ell^2 \mathbb{E} \left\| \tilde{\mathbf{y}}_{k,1} - \tilde{\mathbf{y}}_{k,0} \right\|_2^2 \\
= & \frac{64\ell}{m\mu} \mathbb{E} \|\tilde{\mathcal{G}}_{\lambda,k}(\tilde{\mathbf{y}}_{k,0})\|_2^2 + \left(\frac{64\ell}{m\mu} + 8 \right) \mathbb{E} \left\| \nabla g_k(\tilde{\mathbf{y}}_{k,0}) - \tilde{\mathbf{u}}_{k,0} \right\|_2^2 + 8\ell^2 \mathbb{E} \left\| \tilde{\mathbf{y}}_{k,1} - \tilde{\mathbf{y}}_{k,0} \right\|_2^2 ,
\end{aligned}
$$

where the first inequality is based on Lemma 9 and the second one is due to assumption $S_2 \geq m$. □

We bound the progress of $\tilde{\mathbf{y}}_{k,t}$ by the following lemmas.

Lemma 11. *For Algorithm 4 with $\lambda \leq 2/(\mu + \ell)$, we have*

$$
\mathbb{E} \left\| \tilde{\mathbf{y}}_{k,t+1} - \tilde{\mathbf{y}}_{k,t} \right\|_2^2 \leq \left(1 - \frac{2\lambda\mu\ell}{\mu + \ell} \right) \left\| \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1} \right\|_2^2 \quad \textit{for any} \;\; t \geq 1.
$$

Proof. Using the notations of Algorithm 4, we define $g_{k,t}(\mathbf{y}) = -\frac{1}{S_2} \sum_{i=1}^{S_2} \nabla_{\mathbf{y}} F(\mathbf{x}_{k+1}, \mathbf{y}; \boldsymbol{\xi}_{t,i})$. Then, we have

$$
\begin{aligned}
& \mathbb{E} \left\| \tilde{\mathbf{y}}_{k,t+1} - \tilde{\mathbf{y}}_{k,t} \right\|_2^2 \\
= & \mathbb{E} \left\| \Pi_{\mathcal{Y}}(\tilde{\mathbf{y}}_{k,t} - \lambda \hat{\mathbf{u}}_{k,t}) - \Pi_{\mathcal{Y}}(\tilde{\mathbf{y}}_{k,t-1} - \lambda \hat{\mathbf{u}}_{k,t-1}) \right\|_2^2 \\
\leq & \mathbb{E} \left\| (\tilde{\mathbf{y}}_{k,t} - \lambda \hat{\mathbf{u}}_{k,t}) - (\tilde{\mathbf{y}}_{k,t-1} - \lambda \hat{\mathbf{u}}_{k,t-1}) \right\|_2^2 \\
= & \left\| \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1} \right\|_2^2 - 2\lambda \mathbb{E} \langle \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}, \hat{\mathbf{u}}_{k,t} - \hat{\mathbf{u}}_{k,t-1} \rangle + \lambda^2 \mathbb{E} \left\| \hat{\mathbf{u}}_{k,t} - \hat{\mathbf{u}}_{k,t-1} \right\|_2^2 \\
= & \left\| \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1} \right\|_2^2 - 2\lambda \mathbb{E} \langle \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1}, \nabla g_{k,t-1}(\tilde{\mathbf{y}}_{k,t}) - \nabla g_{k,t-1}(\tilde{\mathbf{y}}_{k,t-1}) \rangle \\
& + \lambda^2 \mathbb{E} \left\| \nabla g_{k,t-1}(\tilde{\mathbf{y}}_{k,t}) - \nabla g_{k,t-1}(\tilde{\mathbf{y}}_{k,t-1}) \right\|_2^2 \\
\leq & \left\| \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1} \right\|_2^2 - \frac{2\lambda\mu\ell}{\mu + \ell} \mathbb{E} \left\| \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1} \right\|_2^2 \\
& + \left(\lambda^2 - \frac{2\lambda}{\mu + \ell} \right) \mathbb{E} \left\| \nabla g_{k,t-1}(\tilde{\mathbf{y}}_{k,t}) - \nabla g_{k,t-1}(\tilde{\mathbf{y}}_{k,t-1}) \right\|_2^2 \\
\leq & \left(1 - \frac{2\lambda\mu\ell}{\mu + \ell} \right) \left\| \tilde{\mathbf{y}}_{k,t} - \tilde{\mathbf{y}}_{k,t-1} \right\|_2^2 ,
\end{aligned}
$$

where the first inequality is based on Lemma 4, the second one comes from inequality (6) of Lemma 2 and the last one is due to $\lambda < 2/(\mu + \ell)$. □

Note that our algorithm estimate $\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k,0})$ by $\frac{1}{\lambda}(\tilde{\mathbf{y}}_{k,0} - \tilde{\mathbf{y}}_{k,1})$, whose norm can be bounded as follows:

Lemma 12. *For Algorithm 4, we have*

$$
\left\| \frac{\tilde{\mathbf{y}}_{k,0} - \tilde{\mathbf{y}}_{k,1}}{\lambda} \right\|_2^2 \leq 3(\Delta_k + 2\ell^2 \varepsilon_{\mathbf{x}}^2 + \delta_k).
$$

Proof. The fact $\|\mathbf{a}+\mathbf{b}+\mathbf{c}\|_2^2 \leq 3\left(\|\mathbf{a}\|_2^2+\|\mathbf{b}\|_2^2+\|\mathbf{c}\|_2^2\right)$ means

$$\begin{aligned}
&\left\|\frac{\tilde{\mathbf{y}}_{k,0}-\tilde{\mathbf{y}}_{k,1}}{\lambda}\right\|_2^2 \\
=&\left\|\frac{\tilde{\mathbf{y}}_{k,0}-\tilde{\mathbf{y}}_{k,1}}{\lambda}-\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1},\mathbf{y}_k)+\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1},\mathbf{y}_k)-\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k,\mathbf{y}_k)+\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k,\mathbf{y}_k)\right\|_2^2 \\
\leq&3\left\|\frac{\tilde{\mathbf{y}}_{k,0}-\tilde{\mathbf{y}}_{k,1}}{\lambda}-\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1},\mathbf{y}_k)\right\|_2^2+3\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1},\mathbf{y}_k)-\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k,\mathbf{y}_k)\right\|_2^2+3\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k,\mathbf{y}_k)\right\|_2^2.
\end{aligned}$$

We use Lemma 4 and Lemma 8 to bound the first and the second term respectively, that is

$$\begin{aligned}
&\left\|\frac{\tilde{\mathbf{y}}_{k,0}-\tilde{\mathbf{y}}_{k,1}}{\lambda}-\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1},\mathbf{y}_k)\right\|_2^2 \\
=&\left\|\frac{1}{\lambda}(\mathbf{y}_k-\Pi_{\mathcal{Y}}(\mathbf{y}_k-\lambda\tilde{\mathbf{u}}_{k,0}))-\frac{1}{\lambda}(\mathbf{y}_k-\Pi_{\mathcal{Y}}(\mathbf{y}_k-\lambda\nabla_{\mathbf{y}}f(\mathbf{x}_{k+1},\mathbf{y}_k)))\right\|_2^2 \\
=&\frac{1}{\lambda^2}\left\|\Pi_{\mathcal{Y}}(\mathbf{y}_k-\lambda\tilde{\mathbf{u}}_{k,0})-\Pi_{\mathcal{Y}}(\mathbf{y}_k-\lambda\nabla_{\mathbf{y}}f(\mathbf{x}_{k+1},\mathbf{y}_k))\right\|_2^2 \\
\leq&\left\|\tilde{\mathbf{u}}_{k,0}-\nabla_{\mathbf{y}}f(\mathbf{x}_{k+1},\mathbf{y}_k)\right\|_2^2=\widetilde{\Delta}_{k,0},
\end{aligned}$$

and

$$\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1},\mathbf{y}_k)-\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k,\mathbf{y}_k)\right\|_2^2 \leq \ell^2\varepsilon_{\mathbf{x}}^2.$$

The third term is $\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k,\mathbf{y}_k)\|_2^2=\delta_k$ because of the definition. Hence, we have

$$\begin{aligned}
\left\|\frac{\tilde{\mathbf{y}}_{k,0}-\tilde{\mathbf{y}}_{k,1}}{\lambda}\right\|_2^2 \leq&3(\widetilde{\Delta}_{k,0}+\ell^2\varepsilon_{\mathbf{x}}^2+\delta_k) \\
\leq&3(\Delta_k+\frac{\ell^2\varepsilon_{\mathbf{x}}^2}{S_2}+\ell^2\varepsilon_{\mathbf{x}}^2+\delta_k) \\
\leq&3(\Delta_k+2\ell^2\varepsilon_{\mathbf{x}}^2+\delta_k).
\end{aligned}$$

□

Then we can establish the recursive relationship of Δ_k and δ_k.

Lemma 13. *For Algorithm 3 and 4 with $\lambda=\frac{1}{8\ell}$, for any $k=k_0+1,k_0+2,\ldots,k_0+q-1$, we have*

$$\begin{aligned}
\Delta_k &\leq \Delta_{k_0}+\frac{3}{64S_2(1-\alpha)}\sum_{i=k_0}^{k-1}\left(\Delta_i+\delta_i+2\ell^2\varepsilon_{\mathbf{x}}^2\right)+\frac{(k-k_0)\ell^2\varepsilon_{\mathbf{x}}^2}{S_2}, \\
\delta_{k+1} &\leq \left(\frac{128\ell}{m\mu}+\frac{3}{8}\right)\delta_k+\left(\frac{64\ell}{m\mu}+\frac{67}{8}\right)\Delta_k+\left(\frac{192\ell}{m\mu}+\frac{35}{4}\right)\ell^2\varepsilon_{\mathbf{x}}^2,
\end{aligned}$$

where

$$\alpha=1-\frac{2\lambda\mu\ell}{\mu+\ell}.$$

Proof. We define

$$\widetilde{\Delta}_{k,t}=\mathbb{E}\left(\|\tilde{\mathbf{v}}_{k,t}-\nabla_{\mathbf{x}}f(\tilde{\mathbf{x}}_{k,t},\tilde{\mathbf{y}}_{k,t})\|_2^2+\|\tilde{\mathbf{u}}_{k,t}-\nabla_{\mathbf{y}}f(\tilde{\mathbf{x}}_{k,t},\tilde{\mathbf{y}}_{k,t})\|_2^2\right),$$

then we have

$$\begin{aligned}
&\widetilde{\Delta}_{k,0} \\
=&\mathbb{E}\left(\|\tilde{\mathbf{v}}_{k,0}-\nabla_{\mathbf{x}}f(\tilde{\mathbf{x}}_{k,0},\tilde{\mathbf{y}}_{k,0})\|_2^2+\|\tilde{\mathbf{u}}_{k,0}-\nabla_{\mathbf{y}}f(\tilde{\mathbf{x}}_{k,0},\tilde{\mathbf{y}}_{k,0})\|_2^2\right) \\
\leq&\mathbb{E}\left(\|\mathbf{v}_k-\nabla_{\mathbf{x}}f(\mathbf{x}_k,\mathbf{y}_k)\|_2^2+\|\mathbf{u}_k-\nabla_{\mathbf{y}}f(\mathbf{x}_k,\mathbf{y}_k)\|_2^2\right)+\frac{\ell^2}{S_2}\mathbb{E}\left(\|\tilde{\mathbf{x}}_{k,0}-\mathbf{x}_k\|_2^2+\|\tilde{\mathbf{y}}_{k,0}-\mathbf{y}_k\|_2^2\right) \\
=&\Delta_k+\frac{\ell^2}{S_2}\mathbb{E}\left(\|\mathbf{x}_{k+1}-\mathbf{x}_k\|_2^2+\|\mathbf{y}_k-\mathbf{y}_k\|_2^2\right) \\
\leq&\Delta_k+\frac{\ell^2\varepsilon_{\mathbf{x}}^2}{S_2},
\end{aligned} \tag{10}$$

where the first inequality comes from Lemma 3 by letting $\mathcal{B}(\cdot) = \nabla f(\cdot)$ and $\mathcal{V}_k = (\mathbf{v}_k, \mathbf{u}_k)$.

Now for any $k \geq 1$, we have

$$\begin{aligned}
\Delta_k =&\mathbb{E}\left(\|\mathbf{v}_k - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\|_2^2 + \|\mathbf{u}_k - \nabla_{\mathbf{y}} f(\mathbf{x}_k, \mathbf{y}_k)\|_2^2\right) \\
=&\widetilde{\Delta}_{k-1, s_{k-1}+1} \\
\leq&\mathbb{E}\left(\|\tilde{\mathbf{v}}_{k-1,0} - \nabla_{\mathbf{y}} f(\tilde{\mathbf{x}}_{k-1,0}, \tilde{\mathbf{y}}_{k-1,0})\|_2^2 + \|\tilde{\mathbf{u}}_{k-1,0} - \nabla_{\mathbf{y}} f(\tilde{\mathbf{x}}_{k-1,0}, \tilde{\mathbf{y}}_{k-1,0})\|_2^2\right) \\
&+ \frac{\ell^2}{S_2} \sum_{t=0}^{s_{k-1}} \left(\|\tilde{\mathbf{x}}_{k-1,t+1} - \tilde{\mathbf{x}}_{k-1,t}\|_2^2 + \|\tilde{\mathbf{y}}_{k-1,t+1} - \tilde{\mathbf{y}}_{k-1,t}\|_2^2\right) \\
=&\widetilde{\Delta}_{k-1,0} + \frac{\ell^2}{S_2} \sum_{t=0}^{s_{k-1}} \|\tilde{\mathbf{y}}_{k-1,t+1} - \tilde{\mathbf{y}}_{k-1,t}\|_2^2 \\
\leq&\widetilde{\Delta}_{k-1,0} + \frac{\ell^2}{S_2} \sum_{t=0}^{s_k - 1} \alpha^t \|\tilde{\mathbf{y}}_{k-1,1} - \tilde{\mathbf{y}}_{k-1,0}\|_2^2 \\
\leq&\widetilde{\Delta}_{k-1,0} + \frac{\ell^2 \lambda^2}{S_2(1-\alpha)} \left\|\frac{\tilde{\mathbf{y}}_{k-1,1} - \tilde{\mathbf{y}}_{k-1,0}}{\lambda}\right\|_2^2,
\end{aligned}$$

where the first inequality is due to Lemma 3; the second inequality comes from Lemma 11 and the third one is due to basic property of geometric sequence.

Combining above results and Lemma 12, we have

$$\begin{aligned}
\Delta_k \leq&\widetilde{\Delta}_{k-1,0} + \frac{\ell^2 \lambda^2}{S_2(1-\alpha)} \left\|\frac{\tilde{\mathbf{y}}_{k-1,1} - \tilde{\mathbf{y}}_{k-1,0}}{\lambda}\right\|_2^2 \\
\leq&\Delta_{k-1} + \frac{\ell^2 \varepsilon_{\mathbf{x}}^2}{S_2} + \frac{3\ell^2 \lambda^2}{S_2(1-\alpha)} \left(\Delta_{k-1} + \delta_{k-1} + 2\ell^2 \varepsilon_{\mathbf{x}}^2\right).
\end{aligned}$$

Summing over the above inequality from k_0 to k, we can prove the first part of this theorem as follows

$$\begin{aligned}
\Delta_k \leq&\Delta_{k-1} + \frac{3\ell^2 \lambda^2}{S_2(1-\alpha)} \left(\Delta_{k-1} + \delta_{k-1} + 2\ell^2 \varepsilon_{\mathbf{x}}^2\right) + \frac{\ell^2 \varepsilon_{\mathbf{x}}^2}{S_2} \\
\leq&\Delta_{k_0} + \frac{3}{64 S_2(1-\alpha)} \sum_{i=0}^{k-1} \left(\Delta_i + \delta_i + 2\ell^2 \varepsilon_{\mathbf{x}}^2\right) + \frac{(k-k_0)\ell^2 \varepsilon_{\mathbf{x}}^2}{S_2}.
\end{aligned}$$

Recall that $\mathbf{y}_{k+1} = \tilde{\mathbf{y}}_{k, s_k}$ and the definition of g_k, we achieve the second part of this lemma:

$$\begin{aligned}
&\delta_{k+1} \\
=&\mathbb{E}\,\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})\|_2^2 \\
\leq&\frac{64\ell}{m\mu}\mathbb{E}\,\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \mathbf{y}_k)\|_2^2 + \left(8 + \frac{64\ell}{m\mu}\right)\mathbb{E}\,\|\nabla f(\tilde{\mathbf{x}}_{k,0}, \tilde{\mathbf{y}}_{k,0}) - \tilde{\mathbf{u}}_{k,0}\|_2^2 + 8\ell^2 \mathbb{E}\,\|\tilde{\mathbf{y}}_{k,1} - \tilde{\mathbf{y}}_{k,0}\|_2^2 \\
\leq&\frac{128\ell}{m\mu}\mathbb{E}\left(\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \mathbf{y}_k) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \mathbf{y}_k)\|_2^2 + \|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \mathbf{y}_k)\|_2^2\right) \\
&+ \left(8 + \frac{64\ell}{m\mu}\right)\widetilde{\Delta}_{k,0} + \frac{3}{8}(\Delta_k + \delta_k + 2\ell^2 \varepsilon_{\mathbf{x}}^2) \\
\leq&\frac{128\ell}{m\mu}\left(\ell^2 \varepsilon_{\mathbf{x}}^2 + \delta_k\right) + \left(8 + \frac{64\ell}{m\mu}\right)\widetilde{\Delta}_{k,0} + \frac{3}{8}(\Delta_k + \delta_k + 2\ell^2 \varepsilon_{\mathbf{x}}^2) \\
\leq&\frac{128\ell}{m\mu}\left(\ell^2 \varepsilon_{\mathbf{x}}^2 + \delta_k\right) + \left(8 + \frac{64\ell}{m\mu}\right)\left(\Delta_k + \frac{\ell^2 \varepsilon_{\mathbf{x}}^2}{S_2}\right) + \frac{3}{8}(\Delta_k + \delta_k + 2\ell^2 \varepsilon_{\mathbf{x}}^2) \\
=&\left(\frac{128\ell}{m\mu} + \frac{3}{8}\right)\delta_k + \left(\frac{64\ell}{m\mu} + \frac{67}{8}\right)\Delta_k + \left(\frac{192\ell}{m\mu} + \frac{35}{4}\right)\ell^2 \varepsilon_{\mathbf{x}}^2,
\end{aligned}$$

where the first inequality is according to Lemma 10, the second inequality is based on Young's inequality and Lemma 12, the third inequality is due to Lemma 8 and the other steps are based on definitions. □

Now we can provide the upper bound of Δ_k and δ_k.

Corollary 2. *For Algorithm 3 and Algorithm 4 with*

$$\eta_k = \min\left(\frac{\varepsilon}{5\kappa\ell\left\|\mathbf{v}_k\right\|_2}, \frac{1}{10\kappa\ell}\right), \lambda = \frac{1}{8\ell}, m = \lceil 1024\kappa\rceil, q = \left\lceil \varepsilon^{-1}\right\rceil,$$
$$S_1 = \left\lceil \frac{2250}{19}\sigma^2\kappa^{-2}\varepsilon^2\right\rceil, S_2 = \left\lceil \frac{3687}{76}\kappa q\right\rceil, \ \textit{and}\ \ \delta_0 \leq \kappa^{-2}\varepsilon^2,$$

Then we have $\Delta_k \leq \frac{19}{1125}\kappa^{-2}\varepsilon^2$ and $\delta_k \leq \kappa^{-2}\varepsilon^2$ for all $k \geq 0$.

Proof. Firstly, we let k_0 be the number of round satisfies $\mathrm{mod}(k_0, q) = 0$ in Algorithm 3 and α be the one defined in Lemma 10, such that

$$\alpha = 1 - \frac{2\lambda\mu\ell}{\mu+\ell} \leq 1 - \frac{1}{8\kappa}. \tag{11}$$

The choice of S_1 indicates we have

$$\Delta_{k_0} < \frac{19}{2250}\kappa^{-2}\varepsilon^2 \tag{12}$$

for all k_0. Then we prove the statement by induction.

Induction base: The choice of S_1 and Assumption 5 means

$$\Delta_{k_0} \leq \frac{19}{2250}\kappa^{-2}\varepsilon^2 < \frac{19}{1125}\kappa^{-2}\varepsilon^2.$$

Combing the assumptions of δ_0, we obtain the induction base.

Induction step: For any $k \geq 1$, we suppose $\Delta_{k'} \leq \frac{19}{1125}\kappa^{-2}\varepsilon^2$ and $\delta_{k'} \leq \kappa^{-2}\varepsilon^2$ holds for all $k' < k$. Let k_0' be the largest integer such that $\mathrm{mod}(k_0', q) = 0$ and $k_0' \leq k$. Using Lemma 10, and inequalities (11) and (12), we have

$$\begin{aligned}\Delta_k \leq& \Delta_{k_0'} + \frac{3}{64S_2(1-\alpha)}\sum_{i=k_0}^{k-1}\left(\Delta_i + \delta_i + 2\ell^2\varepsilon_{\mathbf{x}}^2\right) + \frac{(k-k_0)\ell^2\varepsilon_{\mathbf{x}}^2}{S_2} \\ \leq& \Delta_{k_0'} + \frac{3q}{64S_2(1-\alpha)}\left(\frac{19}{1125}\kappa^{-2}\varepsilon^2 + \kappa^{-2}\varepsilon^2 + \frac{2}{25}\kappa^{-2}\varepsilon^2\right) + \frac{q}{25S_2}\kappa^{-2}\varepsilon^2 \\ \leq& \frac{19}{1125}\kappa^{-2}\varepsilon^2\end{aligned} \tag{13}$$

and

$$\begin{aligned}\delta_k &\leq \left(\frac{128\ell}{m\mu} + \frac{3}{8}\right)\delta_{k-1} + \left(\frac{67}{8} + \frac{64\ell}{m\mu}\right)\Delta_{k-1} + \left(\frac{35}{4} + \frac{192\ell}{m\mu}\right)\ell^2\varepsilon_{\mathbf{x}}^2 \\ &\leq \left(\frac{1}{8} + \frac{3}{8}\right)\delta_{k-1} + \left(\frac{67}{8} + \frac{1}{16}\right)\Delta_{k-1} + \left(\frac{35}{4} + \frac{3}{16}\right)\cdot\frac{\kappa^{-2}\varepsilon^2}{25} \leq \kappa^{-2}\varepsilon^2.\end{aligned} \tag{14}$$

□

C Initialization via Projected Inexact SARAH

The initialization of SREDA (line 2 of Algorithm 3) can be regarded as solving a stochastic constrained convex minimization (concave maximization) problem. Hence, we consider the following formulation

$$\min_{\mathbf{w}\in\mathcal{C}} g(\mathbf{w}) \triangleq \mathbb{E}[G(\mathbf{w};\boldsymbol{\xi})], \tag{15}$$

where $\mathcal{C} \subseteq \mathbb{R}^d$ is a compact and convex set and $\boldsymbol{\xi}$ is a random variable.

We suppose the optimal solution is $\mathbf{w}^* = \arg\min_{\mathbf{w}\in\mathcal{C}} g(\mathbf{w})$, the condition number is $\kappa = \mu/\ell$ and the following assumptions hold.

Assumption 6. *The component function G has an average ℓ-Lipschitz gradient, i.e., there exists a constant $\ell > 0$ such that for any $\mathbf{w}$, $\mathbf{w}'$ and random vector $\boldsymbol{\xi}$, we have*

$$\mathbb{E}\left\|\nabla G(\mathbf{w};\boldsymbol{\xi}) - \nabla F(\mathbf{w}';\boldsymbol{\xi})\right\|_2^2 \leq \ell^2 \left\|\mathbf{w} - \mathbf{w}'\right\|_2^2.$$

Assumption 7. *The component function G is convex. That is, for any $\mathbf{w}$, $\mathbf{w}'$ and random vector $\boldsymbol{\xi}$, we have*

$$G(\mathbf{w};\boldsymbol{\xi}) \geq G(\mathbf{w}';\boldsymbol{\xi}) + \langle \nabla G(\mathbf{w}';\boldsymbol{\xi}), \mathbf{w} - \mathbf{w}' \rangle.$$

Assumption 8. *The function $g(\mathbf{w})$ is μ-strongly-convex. That is, there exists a constant $\mu > 0$ such that for any $\mathbf{w}$ and $\mathbf{w}'$, we have*

$$g(\mathbf{w}) \geq g(\mathbf{w}') + \langle \nabla g(\mathbf{w}'), \mathbf{w} - \mathbf{w}' \rangle + \frac{\mu}{2}\left\|\mathbf{w} - \mathbf{w}'\right\|_2^2.$$

Assumption 9. *The gradient of each component function $G(\mathbf{w};\boldsymbol{\xi})$ has bounded variance. That is, there exists a constant $\sigma > 0$ such that for and $\mathbf{w}$ and random vector $\boldsymbol{\xi}$, we have*

$$\mathbb{E}\left\|\nabla G(\mathbf{w};\boldsymbol{\xi}) - \nabla g(\mathbf{w})\right\|_2^2 \leq \sigma^2 < \infty.$$

We propose projected inexact SARAH (PiSARAH) to solve problem (15), whose detailed procedure is presented in Algorithm 6.

Algorithm 6 PiSARAH $(g(\cdot), K_0)$

```
Input $\mathbf{w}_0 \in \mathcal{C}$, learning rate $\gamma > 0$, inner loop size $m$, batch sizes $b_1$
for $k = 0, \dots, K_0 - 1$ do
    draw $b_1$ samples $\{\boldsymbol{\xi}_1, \cdots, \boldsymbol{\xi}_b\}$
    $\tilde{\mathbf{w}}_{k,0} = \mathbf{w}_k$
    $\tilde{\mathbf{v}}_{k,0} = \frac{1}{b_1}\sum_{i=1}^{b_1} \nabla G(\tilde{\mathbf{w}}_{k,0};\boldsymbol{\xi}_i)$
    $\tilde{\mathbf{w}}_{k,1} = \Pi_{\mathcal{C}}\left(\tilde{\mathbf{w}}_{k,0} - \gamma\tilde{\mathbf{v}}_{k,0}\right)$
    for $t = 1, \dots, m - 1$ do
        draw sample $\boldsymbol{\xi}_t$
        $\tilde{\mathbf{v}}_{k,t} = \tilde{\mathbf{v}}_{k,t-1} + \nabla G(\tilde{\mathbf{w}}_{k,t};\boldsymbol{\xi}_t) - \nabla G(\tilde{\mathbf{w}}_{k,t-1};\boldsymbol{\xi}_t)$
        $\tilde{\mathbf{w}}_{k,t+1} = \Pi_{\mathcal{C}}(\tilde{\mathbf{w}}_{k,t} - \gamma\tilde{\mathbf{v}}_{k,t})$
    end for
    $\mathbf{w}_{k+1} = \tilde{\mathbf{w}}_{k,s_k}$, where $s_k$ is uniformly sampled from $\{1, \dots, m\}$
end for
Output: $\mathbf{w}_{K_0}$
```

We are interested in the convergence behavior of the gradient mapping, that is

$$\mathcal{G}_\gamma(\mathbf{w}) \triangleq \frac{\mathbf{w} - \Pi_{\mathcal{C}}(\mathbf{w} - \lambda\nabla g(\mathbf{w}))}{\lambda}.$$

The remain of this section provide the convergence analysis of PiSARAH.

Note that each epoch of PiSARAH can be regarded as using ConcaveMaximizer (Algorithm 4) on $-g(\cdot)$. Hence, we can follow the analysis of Lemma 10 to achieve the result as follows.

Corollary 3. *For Algorithm 6 with $\gamma = \frac{1}{8\ell}$, we have*

$$\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{k+1})\|_2^2 \leq \left(\frac{64\ell}{m\mu} + \frac{1}{4}\right)\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_k)\|_2^2 + \left(\frac{64\ell}{m\mu} + \frac{33}{4}\right)\mathbb{E}\left\|\nabla g(\tilde{\mathbf{w}}_{k,0}) - \tilde{\mathbf{v}}_{k,0}\right\|_2^2.$$

Proof. Using Lemma 10 in the view of $g_k(\cdot) = g(\cdot)$, we have

$$\begin{aligned}&\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{k+1})\|_2^2\\ \leq&\frac{64\ell}{m\mu}\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_k)\|_2^2 + \left(\frac{64\ell}{m\mu} + 8\right)\mathbb{E}\left\|\nabla g(\tilde{\mathbf{w}}_{k,0}) - \tilde{\mathbf{v}}_{k,0}\right\|_2^2 + 8\ell^2\mathbb{E}\left\|\tilde{\mathbf{w}}_{k,1} - \tilde{\mathbf{w}}_{k,0}\right\|_2^2.\end{aligned} \tag{16}$$

The last term of (16) can be bounded by

$$\begin{aligned}&\left\|\frac{\tilde{\mathbf{w}}_{k,0}-\tilde{\mathbf{w}}_{k,1}}{\gamma}\right\|_2^2\\ \leq&2\left\|\frac{\tilde{\mathbf{w}}_{k,0}-\tilde{\mathbf{w}}_{k,1}}{\gamma}-\mathcal{G}_\gamma(\mathbf{w}_{k,0})\right\|_2^2+2\left\|\mathcal{G}_\gamma(\mathbf{w}_{k,0})\right\|_2^2\\ =&2\left\|\frac{\tilde{\mathbf{w}}_{k,0}-\Pi_{\mathcal{C}}(\mathbf{w}_{k,0}-\lambda\tilde{\mathbf{v}}_{k,0})}{\gamma}-\frac{\mathbf{w}_{k,0}-\Pi_{\mathcal{C}}(\mathbf{w}_{k,0}-\lambda\nabla g(\mathbf{w}_{k,0}))}{\gamma}\right\|_2^2+2\left\|\mathcal{G}_\gamma(\mathbf{w}_{k,0})\right\|_2^2\\ =&2\left\|\tilde{\mathbf{v}}_{k,0}-\nabla g(\mathbf{w}_{k,0})\right\|_2^2+2\left\|\mathcal{G}_\gamma(\mathbf{w}_{k,0})\right\|_2^2,\end{aligned}$$

which implies

$$8\ell^2\mathbb{E}\left\|\tilde{\mathbf{w}}_{k,1}-\tilde{\mathbf{w}}_{k,0}\right\|_2^2\leq\frac{1}{4}\mathbb{E}\left[\left\|\tilde{\mathbf{v}}_{k,0}-\nabla g(\mathbf{w}_{k,0})\right\|_2^2+\left\|\mathcal{G}_\gamma(\mathbf{w}_{k,0})\right\|_2^2\right]. \quad (17)$$

We finish the proof by combining (16) and (17). □

Then we provide the main result in this section to show the convergence of the gradient mapping.

Theorem 3. *For Algorithm 6 with*

$$K_0=\left\lceil\frac{\log\left(2\zeta^{-1}\|\mathcal{G}_\gamma(\mathbf{w}_0)\|_2^2\right)}{\log 2}\right\rceil, m=\lceil 256\kappa\rceil, \lambda=\frac{1}{8\ell} \text{ and } b_1=\left\lceil 34\sigma^2\zeta^{-1}\right\rceil.$$

Then we have $\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2\leq\zeta$.

Proof. Using Corollary 3 with $m=\lceil 256\kappa\rceil$ and $b_1=\left\lceil 34\sigma^2\zeta^{-1}\right\rceil$ we have

$$\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{k+1})\|_2^2\leq\frac{1}{2}\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_k)\|_2^2+\frac{\zeta}{4},$$

which implies

$$\begin{aligned}&\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2-\frac{\zeta}{2}\\ \leq&\frac{1}{2}\left(\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0-1})\|_2^2-\frac{\zeta}{2}\right)\\ \leq&\frac{1}{2^{K_0}}\left(\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_0)\|_2^2-\frac{\zeta}{2}\right)\\ \leq&\frac{1}{2^{K_0}}\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_0)\|_2^2\leq\frac{\zeta}{2}.\end{aligned}$$

Hence, we have $\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2\leq\zeta$. □

The result of Corollary 3 indicate that we hope the PiSARAH as initialization to make the gradient mapping is no larger than $\mathcal{O}(\kappa^{-2}\varepsilon^2)$. The following statement shows we can implement it within $\mathcal{O}(\kappa^2\varepsilon^{-2}\log(\kappa/\varepsilon))$ stochastic gradient evaluations.

Corollary 4. *Under assumptions of Theorem 3, we can obtain* $\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2\leq\kappa^{-2}\varepsilon^2$ *with* $\mathcal{O}(\kappa^2\varepsilon^{-2}\log(\kappa/\varepsilon))$ *stochastic gradient evaluations.*

Proof. Using Theorem 3 with $\zeta=\kappa^{-2}\varepsilon^2$, we have $\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2\leq\kappa^{-2}\varepsilon^2$. The total number of stochastic gradient evaluation is

$$\begin{aligned}&K_0\cdot(b_1+m)\\ =&\left\lceil\frac{\log\left(2\kappa^2\varepsilon^{-2}\|\mathcal{G}_\gamma(\mathbf{w}_0)\|_2^2\right)}{\log 2}\right\rceil\cdot\left(\left\lceil 34\sigma^2\kappa^2\varepsilon^{-2}\right\rceil+\lceil 256\kappa\rceil\right)\\ =&\mathcal{O}(\kappa^2\varepsilon^{-2}\log(\kappa/\varepsilon))\end{aligned}$$

□

D The Proof of Theorem 1

Our proof mainly depends on $f(\mathbf{x}_k, \mathbf{y}_k)$ and its gradient mapping with respect to $\mathbf{y}$, which is different from Lin et al.'s [25] analysis that directly considered the value of $\Phi(\mathbf{x}_k)$ and the distance $\|\mathbf{y}_k - \mathbf{y}^*(\mathbf{x}_k)\|_2$. We split the change of objective functions after one iteration on $(\mathbf{x}_k, \mathbf{y}_k)$ into A_k and B_k as follows

$$f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - f(\mathbf{x}_k, \mathbf{y}_k) = \underbrace{f(\mathbf{x}_{k+1}, \mathbf{y}_k) - f(\mathbf{x}_k, \mathbf{y}_k)}_{A_k} + \underbrace{f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - f(\mathbf{x}_{k+1}, \mathbf{y}_k)}_{B_k},$$

where A_k provides the decrease of function value f and B_k can characterize the difference between $f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1})$ and $\Phi(\mathbf{x}_{k+1})$. We want to show $\mathbb{E}[A_k] \le -\mathcal{O}\left(\kappa^{-1}\varepsilon\right)$ and $\mathbb{E}[B_k] \le \mathcal{O}\left((\kappa\ell)^{-1}\varepsilon^2\right)$. Connecting the upper bound of A_k and B_k, we can bound the average of $\mathbb{E}\left\|\mathbf{v}_k\right\|_2^2$ and use it to prove the upper bound of $\mathbb{E}[\nabla\Phi(\hat{\mathbf{x}})]$ we desired.

We provide two lemmas for preparing the proof of our main results, Theorem 1. The first lemma is to upper bound B_k.

Lemma 14. *Under assumptions of Theorem 1, we have* $\mathbb{E}[B_k] \le \dfrac{134\varepsilon^2}{\kappa\ell}$ *for any* $k \ge 1$.

Proof. Note that our algorithm means $\mathbf{y}_k = \tilde{\mathbf{y}}_{k-1,s_{k-1}}$, where s_{k-1} is sampled from $\{1, \dots, m\}$. Using Lemma 8 by letting $\mathbf{y}^+ = \mathbf{y}_k = \tilde{\mathbf{y}}_{k-1,s_{k-1}}$, $\mathbf{y} = \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}$ and $\mathbf{u} = \tilde{\mathbf{u}}_{k-1,s_{k-1}-1} = -\hat{\mathbf{u}}_{k-1,s_{k-1}-1}$, then we have

$$\begin{aligned}
&\mathbb{E}[f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - f(\mathbf{x}_{k+1}, \mathbf{y}_k)] \\
=&\mathbb{E}[f(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}) - f(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}})] \\
\le&\frac{1}{\mu}\mathbb{E}\left[\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1})\right\|_2^2 + \left\|\nabla_{\mathbf{y}} f(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \tilde{\mathbf{u}}_{k-1,s_{k-1}-1}\right\|_2^2\right].
\end{aligned} \tag{18}$$

We first bound the first term of (18):

$$\begin{aligned}
&\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1})\right\|_2^2 \\
\le&2\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \mathbf{y}_k)\right\|_2^2 + 2\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2 \\
=&2\mathbb{E}\left[\|\mathcal{G}_{\lambda}(\mathbf{x}_{k+1}, \mathbf{y}_k) - \mathcal{G}_{\lambda}(\mathbf{x}_k, \mathbf{y}_k)\|_2 + \|\mathcal{G}_{\lambda}(\mathbf{x}_k, \mathbf{y}_k)\|_2^2\right] \\
&+ 2\mathbb{E}\left\|\mathcal{G}_{\lambda}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2 \\
\le&2(\ell^2\varepsilon_{\mathbf{x}}^2 + \delta_k) + 2\mathbb{E}\left\|\mathcal{G}_{\lambda}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2,
\end{aligned}$$

where the inequalities are based on Young's inequality and Lemma 8.

Using similar ideas, we can also prove

$$\begin{aligned}
&\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2 \\
\le&3\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1})\right\|_2^2 \\
&+ 3\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1}, \tilde{\mathbf{y}}_{k-1,s_{k-1}}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2 \\
&+ 3\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2 \\
\le&6(\ell^2\varepsilon_{\mathbf{x}}^2 + \delta_k) + 3\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2,
\end{aligned}$$

and

$$\begin{aligned}
&\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}-1}) - \mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \tilde{\mathbf{y}}_{k-1,s_{k-1}})\right\|_2^2 \\
=&\mathbb{E}\Bigg\|\frac{\tilde{\mathbf{y}}_{k-1,s_{k-1}-1} - \Pi_{\mathcal{Y}}(\tilde{\mathbf{y}}_{k-1,s_{k-1}-1} - \lambda\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}))}{\lambda} \\
&\qquad - \frac{\tilde{\mathbf{y}}_{k-1,s_{k-1}} - \Pi_{\mathcal{Y}}(\tilde{\mathbf{y}}_{k-1,s_{k-1}} - \lambda\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}}))}{\lambda}\Bigg\|_2^2
\end{aligned}$$

$$
\begin{aligned}
&\leq 2\mathbb{E}\left\|\frac{\Pi_{\mathcal{Y}}(\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}-\lambda\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}))}{\lambda}-\frac{\Pi_{\mathcal{Y}}(\tilde{\mathbf{y}}_{k-1,s_{k-1}}-\lambda\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}}))}{\lambda}\right\|_2^2\\
&\quad+2\mathbb{E}\left\|\frac{\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}-\tilde{\mathbf{y}}_{k-1,s_{k-1}}}{\lambda}\right\|_2^2\\
&\leq 2\mathbb{E}\left\|\frac{(\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}-\lambda\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}))}{\lambda}-\frac{(\tilde{\mathbf{y}}_{k-1,s_{k-1}}-\lambda\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}}))}{\lambda}\right\|_2^2\\
&\quad+2\mathbb{E}\left\|\frac{\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}-\tilde{\mathbf{y}}_{k-1,s_{k-1}}}{\lambda}\right\|_2^2\\
&\leq 6\mathbb{E}\left\|\frac{\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}-\tilde{\mathbf{y}}_{k-1,s_{k-1}}}{\lambda}\right\|_2^2+2\mathbb{E}\left\|\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}))-\nabla g_k(\tilde{\mathbf{y}}_{k-1,s_{k-1}}))\right\|_2^2\\
&\leq (6+2\ell^2\lambda^2)\mathbb{E}\left\|\frac{\tilde{\mathbf{y}}_{k-1,s_{k-1}-1}-\tilde{\mathbf{y}}_{k-1,s_{k-1}}}{\lambda}\right\|_2^2\\
&\leq (6+2\ell^2\lambda^2)\mathbb{E}\left\|\frac{\tilde{\mathbf{y}}_{k-1,1}-\tilde{\mathbf{y}}_{k-1,0}}{\lambda}\right\|_2^2.
\end{aligned}
$$

Combining all above results, we have

$$
\begin{aligned}
&\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_{k+1},\tilde{\mathbf{y}}_{k-1,s_{k-1}-1})\right\|_2^2\\
\leq&14(\ell^2\varepsilon_{\mathbf{x}}^2+\delta_k)+6(6+2\ell^2\lambda^2)\mathbb{E}\left\|\frac{\tilde{\mathbf{y}}_{k-1,1}-\tilde{\mathbf{y}}_{k-1,0}}{\lambda}\right\|_2^2\\
\leq&14(\ell^2\varepsilon_{\mathbf{x}}^2+\delta_k)+18(6+2\ell^2\lambda^2)(\Delta_k+2\ell^2\varepsilon_{\mathbf{x}}^2+\delta_k)\\
\leq&14\left(\frac{1}{25}\kappa^{-2}\varepsilon^2+\kappa^{-2}\varepsilon^2\right)+\frac{1737}{16}\left(\frac{19}{1125}\kappa^{-2}\varepsilon^2+\frac{2}{25}\kappa^{-2}\varepsilon^2+\kappa^{-2}\varepsilon^2\right)\\
\leq&\left(\frac{364}{25}+\frac{1737}{16}\cdot\frac{1234}{1125}\right)\kappa^{-2}\varepsilon^2\\
=&\frac{133641}{1000}\kappa^{-2}\varepsilon^2
\end{aligned}
\tag{19}
$$

where the second inequality is based on Lemma 12 and the third inequality is due to Corollary 3.

We bound the second term of (18) as follows:

$$
\begin{aligned}
&\mathbb{E}\left\|\nabla_{\mathbf{y}}f(\mathbf{x}_k,\tilde{\mathbf{y}}_{k-1,s_{k-1}-1})-\tilde{\mathbf{u}}_{k-1,s_{k-1}-1}\right\|_2^2\\
\leq&\mathbb{E}\left\|\nabla_{\mathbf{y}}f(\mathbf{x}_k,\tilde{\mathbf{y}}_{k-1,s_{k-1}})-\tilde{\mathbf{u}}_{k-1,s_{k-1}}\right\|_2^2\\
=&\mathbb{E}\left\|\nabla_{\mathbf{y}}f(\mathbf{x}_k,\mathbf{y}_k)-\mathbf{u}_k\right\|_2^2\\
\leq&\Delta_k\leq\frac{19}{1125}\kappa^{-2}\varepsilon^2,
\end{aligned}
\tag{20}
$$

where the first inequality is based on Lemma 3 and the last one is due to Corollary 3.

By connecting inequalities (18), (19) and (20), we have

$$
\mathbb{E}[B_k]\leq\frac{1}{\mu}\cdot\frac{1202921}{9000}\kappa^{-2}\varepsilon^2\leq\frac{134\varepsilon^2}{\kappa\ell}.
$$

□

Then we show the estimate error of approximating $\nabla\Phi(\mathbf{x}_k)$ by $\mathbf{v}_k$.

Lemma 15. *Under assumptions of Theorem 1, we have*

$$
\mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k)\right\|_2\leq\mathbb{E}\left\|\mathbf{v}_k\right\|_2+\frac{15}{7}\varepsilon.
$$

Proof. Consider that we have defined $\mathbf{y}^*(\mathbf{x}) = \arg\max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}, \mathbf{y})$, then we have

$$\begin{aligned}
&\mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k) - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2^2 \\
=&\mathbb{E}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}^*(\mathbf{x}_k)) - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2^2 \\
\leq&\ell^2\mathbb{E}\left\|\mathbf{y}^*(\mathbf{x}_k) - \mathbf{y}_k\right\|_2^2 \\
\leq&\frac{4\ell^2}{\mu^2}\mathbb{E}\left\|\mathcal{G}_{\lambda,\mathbf{y}}(\mathbf{x}_k, \mathbf{y}_k)\right\|_2^2 \\
=&\frac{4\ell^2}{\mu^2}\delta_k \leq 4\varepsilon^2,
\end{aligned}$$

where the first equality is based on Lemma 1, the second inequality comes from Corollary 1 and the last inequality is due to Lemma 10.

By using Jensen's inequality, we have

$$\left(\mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k) - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2\right)^2 \leq \mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k) - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2^2 \leq 4\varepsilon^2,$$

which means

$$\begin{aligned}
\mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k)\right\|_2 =&\mathbb{E}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k) - (\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k) - \nabla\Phi(\mathbf{x}_k))\right\|_2 \\
\leq&\mathbb{E}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2 + \mathbb{E}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k) - \nabla\Phi(\mathbf{x}_k)\right\|_2 \\
\leq&\mathbb{E}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2 + 2\varepsilon.
\end{aligned} \tag{21}$$

Similarly, we can use Jensen's inequality and Lemma 10 to prove

$$\left(\mathbb{E}\left\|\mathbf{v}_k - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2\right)^2 \leq \mathbb{E}\left\|\mathbf{v}_k - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2^2 \leq \Delta_k \leq \frac{19}{1125}\kappa^{-2}\varepsilon^2,$$

and

$$\begin{aligned}
\mathbb{E}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2 =&\mathbb{E}\left\|\mathbf{v}_k - (\mathbf{v}_k - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k))\right\|_2 \\
\leq&\mathbb{E}\left\|\mathbf{v}_k\right\|_2 + \mathbb{E}\left\|\mathbf{v}_k - \nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2 \\
\leq&\mathbb{E}\left\|\mathbf{v}_k\right\|_2 + \sqrt{\frac{19}{1125}}\kappa^{-1}\varepsilon \\
\leq&\mathbb{E}\left\|\mathbf{v}_k\right\|_2 + \frac{1}{7}\varepsilon.
\end{aligned} \tag{22}$$

By combining the inequalities (21) and (22), we obtain

$$\begin{aligned}
\mathbb{E}\left\|\nabla\Phi(\mathbf{x}_k)\right\|_2 \leq&\mathbb{E}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k)\right\|_2 + 2\varepsilon \\
\leq&\mathbb{E}\left\|\mathbf{v}_k\right\|_2 + \frac{1}{7}\varepsilon + 2\varepsilon \\
\leq&\mathbb{E}\left\|\mathbf{v}_k\right\|_2 + \frac{15}{7}\varepsilon.
\end{aligned}$$

□

Now we can present the proof of Theorem 1.

Proof. Based on the update of $\mathbf{x}_k$ in Algorithm 3, we have

$$\begin{aligned}
A_k \leq& -\eta_k\langle\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k), \mathbf{v}_k\rangle + \frac{\ell\eta_k^2}{2}\left\|\mathbf{v}_k\right\|_2^2 \\
\leq&\frac{\eta_k}{2}\left\|\nabla_{\mathbf{x}} f(\mathbf{x}_k, \mathbf{y}_k) - \mathbf{v}_k\right\|_2^2 - \left(\frac{\eta_k}{2} - \frac{\ell\eta_k^2}{2}\right)\left\|\mathbf{v}_k\right\|_2^2,
\end{aligned} \tag{23}$$

where the first inequality is due to the average smoothness of f, and second comes from the Cauchy-Schwartz inequality.

The choice of step size η_k implies that

$$\begin{aligned}\left(\frac{\eta_k}{2}-\frac{\ell\eta_k^2}{2}\right)\|\mathbf{v}_k\|_2^2 \geq & \frac{9\varepsilon^2}{100\kappa\ell}\min\left(\frac{\|\mathbf{v}_k\|_2}{\varepsilon},\frac{\|\mathbf{v}_k\|_2^2}{2\varepsilon^2}\right)\\ \geq & \frac{9\varepsilon^2}{100\kappa\ell}\left(\frac{\|\mathbf{v}_k\|_2}{\varepsilon}-2\right)\\ = & \frac{9}{100\kappa\ell}\left(\varepsilon\|\mathbf{v}_k\|_2-2\varepsilon^2\right),\end{aligned} \tag{24}$$

where the first inequality is based on $\kappa \geq 1$ and the definition of η_k; the second one uses the fact that $\min(|x|, \frac{x^2}{2}) \geq |x| - 2$ holds for all x.

By combining inequalities (23), (24) and taking expectation, we obtain the upper bound of $\mathbb{E}[A_k]$:

$$\begin{aligned}\mathbb{E}[A_k] \leq & \frac{1}{20\kappa\ell}\mathbb{E}\|\nabla_{\mathbf{x}}f(\mathbf{x}_k,\mathbf{y}_k)-\mathbf{v}_k\|_2^2-\frac{9}{100\kappa\ell}\left(\varepsilon\mathbb{E}\|\mathbf{v}_k\|_2-2\varepsilon^2\right)\\ \leq & \frac{1}{20\kappa\ell}\Delta_k-\frac{9}{100\kappa\ell}\left(\varepsilon\mathbb{E}\|\mathbf{v}_k\|_2-2\varepsilon^2\right).\end{aligned} \tag{25}$$

The definition of Φ^* and Assumption 1 implies

$$\Phi^* - f(\mathbf{x}_K,\mathbf{y}_K) \leq f(\mathbf{x}_K,\mathbf{y}^*(\mathbf{x}_K)) - f(\mathbf{x}_K,\mathbf{y}_K) \leq \frac{134\varepsilon^2}{\kappa\ell}. \tag{26}$$

where the second inequality can be shown by following the proof of Lemma 14[4].

By combining inequalities (25), (26), Lemma 14 and Corollary 3; and taking the average over $k = 0, \dots, K-1$, we obtain

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}[f(\mathbf{x}_{k+1},\mathbf{y}_{k+1})-f(\mathbf{x}_k,\mathbf{y}_k)] \leq \frac{1}{K}\sum_{k=0}^{K-1}\left(\frac{1}{20\kappa\ell}\Delta_k-\frac{9}{100\kappa\ell}\left(\varepsilon\mathbb{E}\|\mathbf{v}_k\|_2-2\varepsilon^2\right)+\frac{134\varepsilon^2}{\kappa\ell}\right).$$

Consequently, we have

$$\begin{aligned}&\frac{9\varepsilon}{100\kappa\ell}\cdot\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\mathbf{v}_k\|_2\\ \leq & \frac{1}{K}\sum_{k=0}^{K-1}\left(\frac{1}{20\kappa\ell}\Delta_k+\frac{9\varepsilon^2}{50\kappa\ell}+\frac{134\varepsilon^2}{\kappa\ell}\right)+\frac{1}{K}\Big(f(\mathbf{x}_0,\mathbf{y}_0)-\mathbb{E}[f(\mathbf{x}_K,\mathbf{y}_K)]\Big)\\ \leq & \frac{135\varepsilon^2}{\kappa\ell}+\frac{1}{K}\Big(f(\mathbf{x}_0,\mathbf{y}_0)+\frac{134\varepsilon^2}{\kappa\ell}-\Phi^*\Big),\end{aligned}$$

where the second inequality uses Corollary 3 to bound Δ_k.

Rearranging above result, we achieve

$$\frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\mathbf{v}_k\|_2 \leq 1500\varepsilon + \frac{100\kappa\ell}{9K\varepsilon}\Big(\mathbb{E}[f(\mathbf{x}_0,\mathbf{y}_0)]+\frac{134\varepsilon^2}{\kappa\ell}-\Phi^*\Big) = 1500\varepsilon+\frac{100\kappa\ell\Delta_f}{9K\varepsilon}. \tag{27}$$

According to $K = \lceil 100\kappa\ell\varepsilon^{-2}\Delta_f/9\rceil$ and inequality (27), we have

$$\mathbb{E}\|\nabla\Phi(\hat{\mathbf{x}})\|_2 = \frac{1}{K}\sum_{k=0}^{K-1}\mathbb{E}\|\nabla\Phi(\mathbf{x}_k)\|_2 \leq \frac{1}{K}\sum_{k=0}^{K-1}\left(\mathbb{E}\|\mathbf{v}_k\|_2+\frac{15}{7}\varepsilon\right) = 1504\varepsilon.$$

□

E The proof of Theorem 2

In the finite-sum case, we use the full gradient to replace the large batch sample size in stochastic case. Similar to previous section, we extend SARAH [30] to constrained case as the initialization of $\mathbf{y}_0$. We can prove Theorem 2 with minor modifications on the analysis of Theorem 1.

[4] Note that the proof of Lemma 14 is based on inequality (18) whose left-hand side can be replaced by $\mathbb{E}[f(\mathbf{x}_{k+1},\mathbf{y}^*) - f(\mathbf{x}_{k+1},\mathbf{y})]$ for any $\mathbf{y}^* \in \mathcal{Y}$ because of Lemma 9. Hence, we can directly obtain the second inequality of (26) by letting $k = K-1$ and $\mathbf{y}^* = \mathbf{y}^*(\mathbf{x}_K)$.

Algorithm 7 PSARAH $(g(\cdot),\ K_0)$

```
Input $\mathbf{w}_0 \in \mathcal{C}$, learning rate $\gamma > 0$, inner loop size $m$
for $k = 1, \ldots, K_0$ do
    $\tilde{\mathbf{w}}_{k,0} = \mathbf{w}_{s-1}$
    $\tilde{\mathbf{v}}_{k,0} = \nabla g(\tilde{\mathbf{w}}_{k,0})$
    $\tilde{\mathbf{w}}_{k,1} = \Pi_{\mathcal{C}}\left(\tilde{\mathbf{w}}_{k,0} - \gamma \tilde{\mathbf{v}}_{k,0}\right)$
    for $t = 1, \ldots, m-1$ do
        draw sample $\boldsymbol{\xi}_t$
        $\tilde{\mathbf{v}}_{k,t} = \tilde{\mathbf{v}}_{k,t-1} + \nabla G(\tilde{\mathbf{w}}_{k,t}; \boldsymbol{\xi}_t) - \nabla G(\tilde{\mathbf{w}}_{k,t-1}; \boldsymbol{\xi}_t)$
        $\tilde{\mathbf{w}}_{k,t+1} = \Pi_{\mathcal{C}}(\tilde{\mathbf{w}}_{k,t} - \gamma \tilde{\mathbf{v}}_{k,t})$
    end for
    $\mathbf{w}_{k+1} = \tilde{\mathbf{w}}_{k,s_k}$, where $s_k$ is uniformly sampled from $\{1, \ldots, m\}$
end for
Output: $\mathbf{w}_{K_0}$
```

E.1 Initialization by Projected SARAH

We present the detailed procedure of projected SARAH (PSARAH) in Algorithm 7, which is used to initialize $\mathbf{y}_0$ in SREDA for problem (3) (line 2 of Algorithm 5). The algorithm considers the following convex optimization problem

$$\min_{\mathbf{w} \in \mathcal{C}} g(\mathbf{w}) \triangleq \frac{1}{n} \sum_{i=1}^{n} G_i(\mathbf{w}; \boldsymbol{\xi}_i), \tag{28}$$

where H is average ℓ-Lipschitz gradient and convex, h is μ-strongly convex, and $\boldsymbol{\xi}_i$ is a random vector. We have the following convergence result by using SARAH to solve problem (28).

Theorem 4. *For Algorithm 6 with*

$$K_0 = \left\lceil \frac{\log\left(\|\mathcal{G}_\gamma(\mathbf{w}_0)\|_2^2\right)}{\log 2} \right\rceil, m = \lceil 256\kappa \rceil \quad \textit{and} \quad \lambda = \frac{1}{8\ell}.$$

Proof. By following the proof of Corollary 3 with $\tilde{\mathbf{v}}_{k,0} = \nabla g(\tilde{\mathbf{w}}_{k,0})$, we have

$$\begin{aligned}
\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{k+1})\|^2 \leq & \frac{64\ell}{m\mu} \mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_k)\|_2^2 + 8\ell^2 \mathbb{E}\left\|\tilde{\mathbf{w}}_{k,1} - \tilde{\mathbf{w}}_{k,0}\right\|_2^2 \\
\leq & \left(\frac{64\ell}{m\mu} + \frac{1}{4}\right) \mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_k)\|_2^2 \\
\leq & \frac{1}{2} \mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_k)\|_2^2.
\end{aligned}$$

Hence, we have $\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2 \leq \zeta$. □

Similar to stochastic case, we directly obtain the following result.

Corollary 5. *Under assumptions of Theorem 4, we can obtain $\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2 \leq \kappa^{-2}\varepsilon^2$ with $\mathcal{O}\left((n+\kappa)\log(\kappa/\varepsilon)\right)$ stochastic gradient evaluations.*

Proof. Using Theorem 4 with $\zeta = \kappa^{-2}\varepsilon^2$, we have $\mathbb{E}\|\mathcal{G}_\gamma(\mathbf{w}_{K_0})\|_2^2 \leq \kappa^{-2}\varepsilon^2$. The total number of stochastic gradient evaluation is

$$\begin{aligned}
& K_0 \cdot (n + m) \\
= & \left\lceil \frac{\log\left(\kappa^2\varepsilon^{-2}\|\mathcal{G}_\gamma(\mathbf{w}_0)\|_2^2\right)}{\log 2} \right\rceil \cdot (n + \lceil 256\kappa \rceil) \\
= & \mathcal{O}\left((n+\kappa)\log(\kappa/\varepsilon)\right).
\end{aligned}$$

□

E.2 The case of $n \geq \kappa^2$

We set the parameters

$$\zeta = \kappa^{-2}\varepsilon^2,\ \eta_k = \min\left(\frac{\varepsilon}{5\kappa\ell\left\|\mathbf{v}_k\right\|_2}, \frac{1}{10\kappa\ell}\right),\ \lambda = \frac{1}{8\ell},\ q = \lceil\kappa^{-1}n^{1/2}\rceil,$$
$$S_2 = \left\lceil\frac{3687}{76}\kappa q\right\rceil,\ K = \left\lceil\frac{100\kappa\ell\varepsilon^{-2}\Delta_f}{9}\right\rceil,\ \text{and } m = \lceil 1024\kappa\rceil.$$

Then the quantity Δ_{k_0} is zero for any k_0 with $\bmod\,(k_0, q) = 0$. We can follow all analysis of Theorem 1. Note that the different values of q and Δ_{k_0} do not affect the proof of Lemma 13. Therefore we still obtain $\mathbb{E}\left\|\nabla\Phi(\hat{\mathbf{x}})\right\|_2 \leq 1504\varepsilon$ by the parameters setting above. The total complexity of stochastic gradient evaluation is

$$\begin{aligned}
&\mathcal{O}((n+\kappa)\log(\kappa/\varepsilon)) \ + \ \mathcal{O}\left(\frac{K}{q}\cdot n\right) \ + \ \mathcal{O}\left(K \cdot S_2 \cdot m\right) \\
=&\mathcal{O}((n+\kappa)\log(\kappa/\varepsilon)) \ + \ \mathcal{O}\left(\frac{\kappa\varepsilon^{-2}}{\kappa^{-1}n^{1/2}}\cdot n\right) \ + \ \mathcal{O}\left(\kappa\varepsilon^{-2}\cdot n^{1/2}\cdot\kappa\right) \\
=&\mathcal{O}\left(n\log(\kappa/\varepsilon) + \kappa^2 n^{1/2}\varepsilon^{-2}\right).
\end{aligned}$$

E.3 The case of $n \leq \kappa^2$

We set the parameters

$$\zeta = \kappa^{-2}\varepsilon^2,\ \eta_k = \min\left(\frac{\varepsilon}{5\kappa\ell\left\|\mathbf{v}_k\right\|_2}, \frac{1}{10\kappa\ell}\right),\ \lambda = \frac{2}{8\ell},\ q = 1,$$
$$S_2 = 1,\ K = \left\lceil\frac{100\kappa\ell\varepsilon^{-2}\Delta_f}{9}\right\rceil,\ \text{and } m = \lceil 1024\kappa\rceil.$$

The procedure of the algorithm means $\Delta_k = 0$ holds for all k since $q = 1$. Everything is identical to Theorem 1 until Lemma 13. Then we revisit the derivation of Corollary 3. Since we have $\Delta_k = 0$, the inequalities (13) and (14) will be tighter. Hence the all original bounds still hold. The remains could still follow the proof of Theorem 1 and we finally obtain $\mathbb{E}\left\|\nabla\Phi(\hat{\mathbf{x}})\right\|_2 \leq 1504\varepsilon$.

The total complexity of stochastic gradient evaluation is

$$\begin{aligned}
&\mathcal{O}((n+\kappa)\log(\kappa/\varepsilon)) \ + \ \mathcal{O}\left(\frac{K}{q}\cdot n\right) \ + \ \mathcal{O}\left(K \cdot S_2 \cdot m\right) \\
=&\mathcal{O}((n+\kappa)\log(\kappa/\varepsilon)) \ + \ \mathcal{O}\left(\frac{\kappa\varepsilon^{-2}}{1}\cdot n\right) \ + \ \mathcal{O}\left(\kappa\varepsilon^{-2}\cdot 1\cdot\kappa\right) \\
=&\mathcal{O}\left((\kappa^2 + \kappa n)\varepsilon^{-2}\right).
\end{aligned}$$