Lower Complexity Bounds for Finite-Sum Optimization: Proximal Incremental Methods and Randomized Algorithms

Chris Junchi Li[◇]

Department of Electrical Engineering and Computer Sciences[◇]
University of California, Berkeley

October 7, 2024

Abstract

In this paper, we investigate the complexity bounds for finite-sum optimization problems, where the objective is the average of multiple component functions. We focus on Proximal Incremental First-Order (PIFO) algorithms, which access gradient and proximal oracles. By introducing a novel framework based on partitioning tridiagonal matrices into groups, we derive tighter lower bounds for convex-concave and nonconvex-strongly-concave minimax problems. Our construction enables a better analysis of loopless methods and demonstrates that proximal oracles offer limited improvement over gradient oracles. Furthermore, we show that our lower bounds nearly match existing upper bounds, revealing the fundamental limitations of randomized PIFO algorithms in large-scale settings.

Keywords: Finite-Sum Optimization, Proximal Incremental First-Order Methods, Convex-Concave Problems, Complexity Bounds, Minimax Optimization

1 Introduction

Finite-sum optimization problems have become central in many machine learning and large-scale optimization tasks. In these problems, the objective function is the average of several component functions, each representing a part of a larger model. As data sets grow, the need for efficient algorithms that scale well with the number of components becomes critical. Classical first-order methods, while effective, often fall short in practical settings where accessing the full gradient is computationally prohibitive. Thus, randomized methods like Proximal Incremental First-Order (PIFO) algorithms, which balance gradient and proximal updates, have gained prominence. However, the theoretical limits of these methods, particularly for minimax optimization problems, are not well understood.

In this work, we aim to fill this gap by providing lower complexity bounds for finite-sum optimization problems using PIFO algorithms. We focus on both convex-concave and nonconvex-strongly-concave settings, extending previous results by incorporating more realistic assumptions such as average smoothness of the component functions. A key aspect of our analysis is a novel construction of hard instances based on partitioning classical tridiagonal matrices into groups, which allows for a more intuitive and tighter lower bound analysis for PIFO algorithms.

Furthermore, we investigate the power of proximal oracles in comparison to gradient oracles. We demonstrate that, in many cases, proximal oracles offer limited advantages, particularly for smooth functions, challenging the assumption that they significantly improve algorithmic performance.

Mathematically, we consider the following optimization problem

$$\min_{\mathbf{x}\in\mathcal{X}}\max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x},\mathbf{y}) \tag{1}$$

where the feasible sets $\mathcal{X} \subseteq \mathbb{R}^{d_x}$ and $\mathcal{Y} \subseteq \mathbb{R}^{d_y}$ are closed and convex. This formulation contains several popular machine learning applications such as matrix games [CJST19, CJST20, IAGM19], regularized empirical risk minimization [ZX17, TZML18], AUC maximization [Joa05, YWL16, SMZ$^+$18], robust optimization [BTEGN09, YXL$^+$19] and reinforcement learning [DCL$^+$17, DSL$^+$18].

A popular approach for solving minimax problems is the first-order algorithm which iterates with gradient and proximal point operation [CP11, CP16, MOP19a, MOP19b, TJNO19, LCL$^+$19]. Along this line, [ZHZ19] and [IAGM19] presented tight lower bounds for solving strongly-convex-strongly-concave minimax problems by first-order algorithms. [OX18] studied a more general case that the objective function is only convex-concave. However, these analyses [OX18, ZHZ19, IAGM19] do not consider the specific finite-sum structure as in Problem (1). They only considered the deterministic first-order algorithms which are based on the full gradient and exact proximal point iteration.

In big data regimes, the number of components n in Problem (1) could be very large and we would like to devise randomized optimization algorithms that avoid accessing the full gradient frequently. For example, [PB16] used stochastic variance reduced gradient (SVRG) algorithms to solve Problem (1). Similar to convex optimization, one can accelerate it by catalyst [LMH18, YZKH20] and proximal point techniques [Def16, LCL$^+$19]. Note that SVRG is a double-loop algorithm, where the full gradient is calculated periodically with a constant interval. There are also some loopless algorithms where a coin flip decides whether to calculate the full gradient at each iteration [AM22, LXZZ21]. Although randomized algorithms are widely used for solving minimax problems, the study of their lower bounds is still open. All of the existing lower bound analysis focuses on convex or nonconvex minimization problems [AB15, WS16, AS16, LZ17, HLOY18, FLLZ18].

This paper considers randomized PIFO algorithms for solving Problem (1), which are formally defined in Definition 10. These algorithms have access to the Proximal Incremental First-order Oracle (PIFO)

$$h_{f_i}^{\mathrm{PIFO}}(\mathbf{x},\mathbf{y},\gamma) \triangleq \big[f_i(\mathbf{x},\mathbf{y}), \nabla_{\mathbf{x}} f_i(\mathbf{x},\mathbf{y}), -\nabla_{\mathbf{y}} f_i(\mathbf{x},\mathbf{y}), \mathrm{prox}_{f_i}^{\gamma}(\mathbf{x},\mathbf{y})\big] \tag{2}$$

where $i \in \{1,\dots,n\}$, $\gamma > 0$, and the proximal operator is defined as

$$\mathrm{prox}_{f_i}^{\gamma}(\mathbf{x},\mathbf{y}) \triangleq \arg\min_{\mathbf{u}\in\mathbb{R}^{d_x},\mathbf{v}\in\mathbb{R}^{d_y}} \max \left\{ f_i(\mathbf{u},\mathbf{v}) + \frac{1}{2\gamma}\|\mathbf{x}-\mathbf{u}\|_2^2 - \frac{1}{2\gamma}\|\mathbf{y}-\mathbf{v}\|_2^2 \right\}$$

Compared to Incremental First-order Oracle (IFO), which is defined as $h_{f_i}^{\mathrm{IFO}}(\mathbf{x},\mathbf{y}) = [f_i(\mathbf{x},\mathbf{y}), \nabla_{\mathbf{x}} f_i(\mathbf{x},\mathbf{y}), -\nabla_{\mathbf{y}} f_i(\mathbf{x},\mathbf{y})]$, PIFO additionally provides the proximal oracle of the component function. To incorporate loopless methods, we also allow PIFO algorithms to access the full gradient infrequently with the interval obeying geometric distributions.

We consider the general setting where $f(\mathbf{x},\mathbf{y})$ is L-smooth and (μ_x,μ_y)-convex-concave, i.e., the function $f(\cdot,\mathbf{y}) - \frac{\mu_x}{2}\|\cdot\|_2^2$ is convex for any $\mathbf{y}\in\mathcal{Y}$ and the function $-f(\mathbf{x},\cdot) - \frac{\mu_y}{2}\|\cdot\|_2^2$ is convex for any $\mathbf{x}\in\mathcal{X}$. When $\mu_x,\mu_y \geq 0$, our goal is to find an ε-suboptimal solution $(\widehat{\mathbf{x}},\widehat{\mathbf{y}})$ to Problem (1) such that the primal-dual gap is less than ε, i.e.,

$$\max_{\mathbf{y}\in\mathcal{Y}} f(\widehat{\mathbf{x}},\mathbf{y}) - \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x},\widehat{\mathbf{y}}) < \varepsilon$$

On the other hand, when $\mu_x < 0, \mu_y > 0$, $f(\mathbf{x}, \mathbf{y})$ is called a nonconvex-strongly-concave function, which has been widely studied in [RLLY18, LJJ20, OLR20, LYHZ20]. In this case, our goal is instead to find an ε-stationary point $\widehat{\mathbf{x}}$ of $\phi_f(\mathbf{x}) \triangleq \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}, \mathbf{y})$, which is defined as

$$\|\nabla\phi_f(\widehat{\mathbf{x}})\|_2 < \varepsilon$$

It is worth noting that by setting the feasible set of $\mathbf{y}$ as a singleton, the minimax problem becomes a minimization problem. Then we can omit the dependence of f on $\mathbf{y}$ and rewrite the function as $f(\mathbf{x})$ with some abuse of notation. When $f(\mathbf{x})$ is convex, our goal is to find an ε-suboptimal solution $\widehat{\mathbf{x}}$ such that $f(\widehat{\mathbf{x}}) - \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) < \varepsilon$, while when $f(\mathbf{x})$ is nonconvex, our goal is to find an ε-stationary point $\widehat{\mathbf{x}}$ such that $\|\nabla f(\widehat{\mathbf{x}})\|_2 < \varepsilon$.

Contributions Our work provides a novel perspective on the complexity bounds for finite-sum optimization problems. The key contributions of this paper are summarized as follows:

(i) We introduce a new framework for analyzing lower complexity bounds in finite-sum optimization by partitioning the classical tridiagonal matrix into n groups, with each group representing a component function. This construction enhances the analysis of both Incremental First-Order (IFO) and Proximal Incremental First-Order (PIFO) algorithms.

(ii) We establish tighter lower complexity bounds for finite-sum minimax problems where the objective function is convex-concave or nonconvex-strongly-concave, and the component functions are L-average smooth. Our bounds nearly match existing upper bounds, differing only by logarithmic factors, as summarized in our results.

(iii) For finite-sum minimization problems, we derive similar lower bounds as previous work, but with a more intuitive framework that requires fewer dimensions to construct hard instances. This provides greater insight into the optimization process.

(iv) Our analysis shows that for smooth functions, proximal oracles do not provide a significant advantage over gradient oracles, particularly in terms of complexity bounds. This reinforces prior observations, indicating that the power of PIFO algorithms is comparable to that of IFO algorithms for many practical cases.

These findings deepen our understanding of the limitations of randomized algorithms in finite-sum optimization problems, particularly when dealing with large-scale datasets. Future research may explore additional algorithmic strategies that further leverage proximal information without increasing complexity.

1.1 Related Work

Lower bounds for finite-sum minimization problems There has been extensive study on this topic. [AB15] established the lower bound $\Omega(n + \sqrt{n(\kappa-1)}\log(1/\varepsilon))$ when each component is L-smooth and their average is μ-strongly convex by a resisting oracle construction, where $\kappa = L/\mu$ is the condition number. However, their lower bound only applies to deterministic algorithms. [LZ17] obtained the lower bound $\Omega((n+\sqrt{n\kappa})\log(1/\varepsilon))$ for randomized incremental gradient methods, but their bound does not apply to multi-loop methods such as SVRG [JZ13] and SARAH [NLST17]. [WS16] provided the lower bound $\Omega(n + \sqrt{n\kappa}\log(1/\varepsilon))$ for any randomized algorithms using gradient and proximal oracles. Moreover, when the objective is only convex, their lower

Table 1. Upper and lower bounds with the assumption that $\{f_i\}_{i=1}^n$ is L-average smooth and f is (μ_x, μ_y)-convex-concave. When $\mu_x \geq 0$ and $\mu_y \geq 0$, the goal is to find an ε-suboptimal solution with $\operatorname{diam}(\mathcal{X}) \leq 2R_x, \operatorname{diam}(\mathcal{Y}) \leq 2R_y$. And when $\mu_x < 0$, the goal is to find an ε-stationary point of the function $\phi_f(\mathbf{x}) \triangleq \max_{\mathbf{y}\in\mathcal{Y}} f(\cdot, \mathbf{y})$ with $\Delta = \phi_f(\mathbf{x}_0) - \min_{\mathbf{x}} \phi_f(\mathbf{x})$ and $\mathcal{X} = \mathbb{R}^{d_x}, \mathcal{Y} = \mathbb{R}^{d_y}$. The condition numbers are defined as $\kappa_x = L/\mu_x$ and $\kappa_y = L/\mu_y$ when $\mu_x, \mu_y > 0$.

Cases	Upper or Lower bounds	References
$\mu_x > 0, \mu_y > 0$	$\widetilde{\mathcal{O}}\left(\sqrt{n\left(\sqrt{n}+\kappa_x\right)\left(\sqrt{n}+\kappa_y\right)}\log(1/\varepsilon)\right)$	[LXZZ21]
	$\Omega\left(\sqrt{n\left(\sqrt{n}+\kappa_x\right)\left(\sqrt{n}+\kappa_y\right)}\log(1/\varepsilon)\right)$	Theorem 1
$\mu_x = 0, \mu_y > 0$	$\widetilde{\mathcal{O}}\Big(\big(n + R_x n^{3/4}\sqrt{\frac{L}{\varepsilon}} + R_x\sqrt{\frac{nL\kappa_y}{\varepsilon}} + n^{3/4}\sqrt{\kappa_y}\big)\log\left(\frac{1}{\varepsilon}\right)\Big)$	[LXZZ21]
	$\Omega\Big(n + R_x n^{3/4}\sqrt{\frac{L}{\varepsilon}} + R_x\sqrt{\frac{nL\kappa_y}{\varepsilon}} + n^{3/4}\sqrt{\kappa_y}\log\left(\frac{1}{\varepsilon}\right)\Big)$	Theorem 2
$\mu_x = 0, \mu_y = 0$	$\widetilde{\mathcal{O}}\Big(\big(n + \frac{\sqrt{n}LR_xR_y}{\varepsilon} + (R_x{+}R_y)n^{3/4}\sqrt{\frac{L}{\varepsilon}}\big)\log\left(\frac{1}{\varepsilon}\right)\Big)$	[LXZZ21]
	$\Omega\Big(n + \frac{\sqrt{n}LR_xR_y}{\varepsilon} + (R_x{+}R_y)n^{3/4}\sqrt{\frac{L}{\varepsilon}}\Big)$	Theorem 3
$\mu_x < 0, \mu_y > 0,$ $\kappa_y = \Omega(n)$	$\widetilde{\mathcal{O}}\left((n + n^{3/4}\sqrt{\kappa_y})\Delta L\varepsilon^{-2}\right)$	[ZYG+21]
	$\Omega\left(n + \sqrt{n\kappa_y}\Delta L\varepsilon^{-2}\right)$	Theorem 4; [ZYG+21]

bound is $\Omega(n + \sqrt{nL/\varepsilon})$. [AS16] established a similar lower bound for the strongly convex case and their bound also applies to stochastic coordinate-descent methods. [HLOY18] improved this bound to $\Omega\big(\frac{n\log(1/\varepsilon)}{(1+\log(n/\kappa))_+}\big)$ when $\kappa = \mathcal{O}(n)$. [ZG19] proved lower bounds $\Omega(n+n^{3/4}\sqrt{\kappa}\log(1/\varepsilon))$ and $\Omega(n + n^{3/4}\sqrt{L/\varepsilon})$ for the strongly convex and convex case respectively under the weaker condition that the class of component function is L-average smooth.

When the objective is nonconvex, [FLLZ18] proved the lower bound $\Omega(L\sqrt{n}/\varepsilon^2)$ for $\varepsilon = \mathcal{O}(\sqrt{L}/n^{1/4})$ under the average smooth condition. [LBZR21] improved the bound to $\Omega(n+L\sqrt{n}/\varepsilon^2)$ for an arbitrary ε. Under a more refined condition that objective is μ-weakly convex (see Definition 4), [ZG19] established the lower bound to $\Omega(1/\varepsilon^2 \min\{n^{3/4}\sqrt{L\mu}, \sqrt{n}L\})$ when ε is sufficiently small. They also provided the lower bound $\Omega(1/\varepsilon^2 \min\{\sqrt{nL\mu}, L\})$ when each component is L-smooth.

Upper bounds for finite-sum minimax problems For Problem (1), if $\mu_x, \mu_y \geq 0$ and each f_i is L-smooth, the best known upper bound is $\mathcal{O}\left((n + \sqrt{n}(\kappa_x + \kappa_y))\log(1/\varepsilon)\right)$ [CJST19, LCL+19]. Furthermore, if each f_i has L-cocoercive gradient, which is a stronger assumption than smoothness, [CGFLJ19] provided an upper bound $\mathcal{O}\left((n + \kappa_x + \kappa_y)\log(1/\varepsilon)\right)$. If $\{f_i\}_{i=1}^n$ is L-average smooth, Accelerated SVRG [PB16] attained the upper bound $\widetilde{\mathcal{O}}\left((n + \sqrt{n}(\kappa_x + \kappa_y))\log(1/\varepsilon)\right)$ and [AM22] obtained the bound $\mathcal{O}\left((n + \sqrt{n}(\kappa_x + \kappa_y))\log(1/\varepsilon)\right)$. Then [LXZZ21] improved this bound to $\widetilde{\mathcal{O}}(\sqrt{n(\sqrt{n}+\kappa_x)(\sqrt{n}+\kappa_y)}\log(1/\varepsilon))$ by catalyst acceleration. The same technique was also employed to derive lower bounds for the convex-strongly-concave case where $\mu_x = 0, \mu_y > 0$ [YZKH20, LXZZ21].

For the convex-concave case ($\mu_x = \mu_y = 0$), [CJST19] established the upper bound $\mathcal{O}(n +$

$\sqrt{n}L/\varepsilon)$ under the smoothness assumption, while [AM22] developed the same upper bound under the average smoothness assumption. [LXZZ21] still used the catalyst acceleration and derived a similar bound.

In terms of the nonconvex-strongly-concave case ($\mu_x < 0, \mu_y > 0$), [LYHZ20] proposed an upper bound $\widetilde{\mathcal{O}}\left(n + \min\{\sqrt{n}\kappa_y^2, \kappa_y^2 + n\kappa_y\}\varepsilon^{-2}\right)$, while [ZYG+21] developed an upper bound $\widetilde{\mathcal{O}}\left((n + n^{3/4}\sqrt{\kappa_y})L\varepsilon^{-2}\right)$. The latter is better when $n = \mathcal{O}(\kappa^4)$. We emphasize that both results are under the average smoothness assumption.

Loopless methods Variance-reduced methods for finite-sum minimization problems such as SVRG [JZ13], Katyusha [AZ17a] and SARAH [NLST17] have a double-loop design where the full gradient needs to be calculated periodically. Recently, many researchers aim to study their loopless variants or devise new loopless methods such that whether to access the full gradient depends on a coin toss with a small head probability. Equivalently speaking, the inner loop size obeys the geometric distribution with a small success probability. Such a design facilities theoretical analysis without deteriorating the convergence rates. For example, loopless SVRG (L-SVRG) was first proposed in [HLLJM15] and then further analyzed in [KHR20, QQR21] together with loopless Katyusha (L-Katyusha). Loopless SARAH (L2S) was developed in [LMG20]. Other loopless methods include but are not limited to KatyushaX [AZ18], PAGE [LBZR21] and Anita [Li21]. For finite-sum minimax problems, there are also many loopless methods [LBJM+20, AM22, BGBL22].

The proximal oracle The proximal oracle provides more information than the gradient oracle and has been used in algorithm design [SSZ13, Def16, LZ17, LCL+19]. Compared with catalyst acceleration, employing proximal oracles would neither increase the number of loops nor induce additional parameter tuning. When each component function enjoys a simple form [ZX17, DCL+17, LZ17, CJST19], the proximal operator can be computed efficiently. In terms of the power of proximal oracles, [WS16] have shown that for smooth functions, the gradient oracle is sufficient for the optimal rate. As a comparison, for nonsmooth functions, having access to proximal oracles does reduce the complexity and [WS16] presented optimal methods that improve over those only using gradient oracles.

Organization The remainder of this paper is organized as follows. In Section 2, we introduce essential definitions, necessary notation, and a concentration inequality for geometric distributions. Section 3 presents the definition of PIFO algorithms and discusses their properties. In Section 4, we describe our framework for constructing hard instances and define the optimization complexity for Problem (1). Section 5 provides the lower complexity bounds for finite-sum minimax problems, while Section 6 focuses on minimization problems. Finally, in Section 7, we summarize our results and outline several directions for future research.

2 Preliminaries

In this section, we present some necessary notation and definitions used in our paper and then give a concentration inequality about geometric distributions.

Notation We denote the set $\{1, 2, \ldots, n\}$ by $[n]$. $a_+ \triangleq \max\{a, 0\}$ represent the positive part of a real number. The projection operator is defined as $\mathcal{P}_{\mathcal{X}}(\mathbf{x}) \triangleq \arg\min_{\mathbf{x}' \in \mathcal{X}} \|\mathbf{x}' - \mathbf{x}\|_2$ where $\mathcal{X}$ is a convex set and $\|\cdot\|_2$ is the Euclidean norm. We use $\mathbf{0}$ for all-zero vectors and $\mathbf{e}_i$ for the

unit vector with the i-th element equal to 1 and others equal to 0. Their dimensions will be specified by an additional subscript, if necessary, and otherwise are clear from the context. We use $\mathrm{Geo}(p)$ to denote the geometric distribution with success probability p, i.e., $Y \sim \mathrm{Geo}(p)$ implies $\mathbb{P}[Y = k] = (1-p)^k p$ for $0 < p \leq 1, k \in \{0, 1, 2, \dots\}$. Finally, we use the notation $\mathcal{O}(\cdot), \Omega(\cdot), \Theta(\cdot)$ to hide absolute constants which do not depend on any problem parameter, and notation $\widetilde{\mathcal{O}}(\cdot)$ to hide absolute constants and log factors.

Definition 1. *For a differentiable function $\varphi(\mathbf{x})$ from $\mathcal{X}$ to $\mathbb{R}$ and $L > 0$, φ is said to be L-smooth if its gradient is L-Lipschitz continuous; that is, for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$, we have*

$$\|\nabla\varphi(\mathbf{x}_1) - \nabla\varphi(\mathbf{x}_2)\|_2 \leq L \|\mathbf{x}_1 - \mathbf{x}_2\|_2$$

Definition 2. *For a class of differentiable functions $\{\varphi_i(\mathbf{x}) : \mathcal{X} \to \mathbb{R}\}_{i=1}^n$ and $L > 0$, $\{\varphi_i\}_{i=1}^n$ is said to be L-average smooth if for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$, we have*

$$\frac{1}{n}\sum_{i=1}^n \|\nabla\varphi_i(\mathbf{x}_1) - \nabla\varphi_i(\mathbf{x}_2)\|_2^2 \leq L^2 \|\mathbf{x}_1 - \mathbf{x}_2\|_2^2$$

The assumption of average smoothness is widely used in many finite-sum optimizations [ZXG18, FLLZ18, ZG19, AM22].

Now we discuss the relationship between smoothness and average smoothness. For a class of differentiable functions $\{\varphi_i(\mathbf{x}) : \mathcal{X} \to \mathbb{R}\}_{i=1}^n$ and their average $\overline{\varphi}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^n \varphi_i(\mathbf{x})$, we have the following result

$$\varphi_i \text{ is } L\text{-smooth}, \forall i \Longrightarrow \{\varphi_i\}_{i=1}^n \text{ is } L\text{-average smooth} \Longrightarrow \overline{\varphi} \text{ is } L\text{-smooth}$$

Moreover, suppose that φ_i is L_i-smooth, $\overline{\varphi}$ is L-smooth and $\{\varphi_i\}_{i=1}^n$ is L'-average smooth, we have $L \leq L' \leq \sqrt{\frac{1}{n}\sum_{i=1}^n L_i^2}$ and $L \leq \frac{1}{n}\sum_{i=1}^n L_1$.

However, L and L' can be much smaller than L_i. For example, if $\varphi_i(\mathbf{x}) = \frac{1}{2}(\langle \mathbf{e}_i, \mathbf{x}\rangle)^2$, then we have $L_i = 1$, $L = 1/n$ and $L' = 1/\sqrt{n}$. As a result, it is more restrictive to say that each φ_i is L-smooth than to say that $\{\varphi_i\}_{i=1}^n$ is L-average smooth.

Definition 3. *For a differentiable function $\varphi(\mathbf{x})$ from $\mathcal{X}$ to $\mathbb{R}$, φ is said to be convex if for any $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$, we have*

$$\varphi(\mathbf{x}_2) \geq \varphi(\mathbf{x}_1) + \langle \nabla\varphi(\mathbf{x}_1), \mathbf{x}_2 - \mathbf{x}_1\rangle$$

Definition 4. *For a constant μ, if the function $\widehat{\varphi}(\mathbf{x}) = \varphi(\mathbf{x}) - \frac{\mu}{2}\|\mathbf{x}\|_2^2$ is convex, then φ is said to be μ-strongly convex if $\mu > 0$ and φ is said to be μ-weakly convex if $\mu < 0$.*

One can check that if φ is L-smooth, then it is $(-L)$-weakly-convex.

Definition 5. *For a differentiable function $\varphi(\mathbf{x})$ from $\mathcal{X}$ to $\mathbb{R}$, we call $\widehat{\mathbf{x}}$ an ε-stationary point of φ if*

$$\|\nabla\varphi(\widehat{\mathbf{x}})\|_2 < \varepsilon$$

Definition 6. *For a differentiable function $f(\mathbf{x}, \mathbf{y})$ from $\mathcal{X} \times \mathcal{Y}$ to $\mathbb{R}$, f is said to be convex-concave, if the function $f(\cdot, \mathbf{y})$ is convex for any $\mathbf{y} \in \mathcal{Y}$ and the function $-f(\mathbf{x}, \cdot)$ is convex for any $\mathbf{x} \in \mathcal{X}$. Furthermore, f is said to be (μ_x, μ_y)-convex-concave, if the function $f(\mathbf{x}, \mathbf{y}) - \frac{\mu_x}{2}\|\mathbf{x}\|_2^2 + \frac{\mu_y}{2}\|y\|_2^2$ is convex-concave.*

Definition 7. *We call a minimax optimization problem* $\min_{\mathbf{x}\in\mathcal{X}}\max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y})$ *satisfying the strong duality condition if*

$$\min_{\mathbf{x}\in\mathcal{X}}\max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y}) = \max_{\mathbf{y}\in\mathcal{Y}}\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x},\mathbf{y})$$

By Sion's minimax theorem, if $\varphi(\mathbf{x},\mathbf{y})$ is convex-concave and either $\mathcal{X}$ or $\mathcal{Y}$ is a compact set, then the strong duality condition holds.

Definition 8. *We call* $(\mathbf{x}^*,\mathbf{y}^*)\in\mathcal{X}\times\mathcal{Y}$ *the saddle point of* $f(\mathbf{x},\mathbf{y})$ *if*

$$f(\mathbf{x}^*,\mathbf{y}) \le f(\mathbf{x}^*,\mathbf{y}^*) \le f(\mathbf{x},\mathbf{y}^*)$$

for all $(\mathbf{x},\mathbf{y})\in\mathcal{X}\times\mathcal{Y}$.

Definition 9. *Suppose the strong duality of Problem (1) holds. We call* $(\widehat{\mathbf{x}},\widehat{\mathbf{y}})\in\mathcal{X}\times\mathcal{Y}$ *an* ε*-suboptimal solution to Problem (1) if*

$$\max_{\mathbf{y}\in\mathcal{Y}} f(\widehat{\mathbf{x}},\mathbf{y}) - \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x},\widehat{\mathbf{y}}) < \varepsilon$$

2.1 A Concentration Inequality about Geometric Distributions

In this subsection, we introduce a concentration inequality about geometric distributions.

Lemma 1. *Let* $\{Y_i\}_{i=1}^m$ *be independent random variables, and* Y_i *follows a geometric distribution with success probability* p_i*. Then for* $m\ge 2$*, we have*

$$\mathbb{P}\left[\sum_{i=1}^m Y_i > \frac{m^2}{4(\sum_{i=1}^m p_i)}\right] \ge \frac{1}{9}$$

Lemma 1 implies that at least with a constant probability, the sum of geometric random variables is larger than a constant number, which depends on the number of variables and their success probabilities. Then we can obtain a lower bound of $\mathbb{E}\sum_{i=1}^m Y_i$, which is helpful to the construction in Section 4. The proof is deferred to Appendix A.

3 PIFO Algorithms

In this section, we present our definition of PIFO algorithms. We first discuss previous definitions in Section 3.1 and our formal definition is given in Section 3.2.

3.1 Discussion on Previous Definitions

In this subsection, we discuss the definitions of oracles and algorithms in previous work on the minimization problem $\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^n f_i(\mathbf{x})$. With some abuse of notation, we do not distinguish the oracles for minimization problems from those for minimax problems.

IFO and PIFO The IFO is defined as $h_{f_i}^{\mathrm{IFO}}(\mathbf{x}) \triangleq [f_i(\mathbf{x}), \nabla f_i(\mathbf{x})]$, which takes as input a point $\mathbf{x} \in \mathcal{X}$ and a component function f_i and returns the function value and the gradient of f_i at $\mathbf{x}$. Many lower bounds for minimization optimization are based on this oracle, e.g., [AB15, LZ17, ZG19]. They all consider *linear-span randomized first-order algorithms*[1]. For these algorithms, the current point lies in the linear span of previous points and gradients returned by earlier IFO calls.

[WS16] considers the PIFO which is stronger than IFO and is defined as $h_{f_i}^{\mathrm{PIFO}}(\mathbf{x}, \gamma) \triangleq [f_i(\mathbf{x}), \nabla f_i(\mathbf{x}), \mathrm{prox}_{f_i}^{\gamma}(\mathbf{x})]$ with the proximal operator $\mathrm{prox}_{f_i}^{\gamma}(\mathbf{x}) \triangleq \arg\min_{\mathbf{u}} \left\{ f_i(\mathbf{u}) + \frac{1}{2\gamma} \|\mathbf{x} - \mathbf{u}\|_2^2 \right\}$. When f_i is convex, any $\gamma > 0$ is feasible. Different from IFO, PIFO provides global information about the function. To see this, letting $\gamma \to \infty$ yields the exact minimizer of f_i. Based on PIFO, [WS16] consider the class of *any* randomized algorithms, a more general class than *linear-span randomized first-order algorithms*. We also emphasize that when f_i is nonconvex, γ should be sufficiently small such that $f_i(\mathbf{u}) + \frac{1}{2\gamma} \|\mathbf{x} - \mathbf{u}\|_2^2$ is a convex function of $\mathbf{u}$. Otherwise, it can be pretty hard to calculate $\mathrm{prox}_{f_i}^{\gamma}(\mathbf{x})$. Specially, if f is $(-\mu)$-weakly convex, we need to ensure $0 < \gamma < 1/\mu$.

Sampling of the component function Note that both IFO and PIFO depend on a specific component f_i. Different methods use different ways to choose the index i. Some of them, e.g., SAGA [DBLJ14], RPDG [LZ17], pick i randomly according to some distribution over $[n]$ and the full gradient is calculated only at the initial point. However, much more methods need to calculate the full gradient periodically, either with a deterministic or random interval. For multi-loop methods, e.g., SVRG [JZ13], Katyusha [AZ17a] and Spider [FLLZ18], the interval is predetermined, while for loopless methods, e.g., KatyushaX [AZ18], L2S [LMG20], L-SVRG [KHR20], the interval is a geometric random variable.

The lower bound of [LZ17] requires that the index i_t at iteration t is sampled from a predetermined distribution over $[n]$. Thus their bound does not apply to methods such as SVRG and L-SVRG. [WS16, ZG19] do not specify the way to choose i_t. As a result, their class of algorithms does include those multi-loop or loopless methods.

[AS16] and [HLOY18] consider p-CLI algorithms equipped with the generalized first-order oracle, where the current point and the gradient can be left-multiplied by preconditioning matrices. They do not specify the way to choose i_t, either. Thus their lower bounds apply to all the methods mentioned above. Moreover, their framework can also be equipped with the steepest coordinate descent oracle to incorporate methods such as SDCA [SS16].

3.2 Our Definition

In this subsection, we come back to the minimax problem (1) and formally introduce the definition of PIFO algorithms.

Recall that the PIFO has been defined in (2). For convenience, we also define the First-order Oracle (FO) as $h_f^{\mathrm{FO}}(\mathbf{x}, \mathbf{y}) \triangleq [f(\mathbf{x}, \mathbf{y}), \nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}), -\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})]$, which returns the full gradient information. Since the feasible set of Problem (1) is not necessarily the whole space, the algorithm should also access the projection operators $\mathcal{P}_{\mathcal{X}}$ and $\mathcal{P}_{\mathcal{Y}}$. Then we can define the PIFO algorithms we focus on in our paper.

Definition 10. *Consider a randomized PIFO algorithm $\mathcal{A}$ to solve Problem (1). Denote the point obtained by $\mathcal{A}$ after step t by $(\mathbf{x}_t, \mathbf{y}_t)$, which is generated by the following procedure.*

[1]The formal definition is given in Definition 3.3 in [ZG19]. Although the results of [AB15] do not rely on the linear span assumption, this assumption can be made without loss of generality, as shown in their Appendix A.

(i) Initialize the set $\mathcal{H}$ as $\{(\mathbf{x}_0, \mathbf{y}_0)\}$, the distribution $\mathcal{D}$ over $[n]$, a positive number $q \leq c_0/n$ and set $t = 1$.

(ii) Sample $i_t \sim \mathcal{D}$ and query the oracle $h_{f_i}^{\mathrm{PIFO}}$

(iii) Sample a Bernoulli random variable a_t with expectation equal to q. If $a_t = 1$, query the FO h_f^{FO} and add $(\mathbf{x}_{t-1}, \mathbf{y}_{t-1})$ to $\mathcal{H}$.

(iv) Obtain $(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t)$ following the linear-span protocol

$$\begin{aligned}(\widetilde{\mathbf{x}}_t, \widetilde{\mathbf{y}}_t) \in \operatorname{span}\big\{&(\mathbf{x}_0, \mathbf{y}_0), \ldots, (\mathbf{x}_{t-1}, \mathbf{y}_{t-1}), \operatorname{prox}_{f_{i_j}}^{\gamma_j}(\mathbf{x}_l, \mathbf{y}_l) \text{ for } l < j \leq t,\\ &(\nabla_{\mathbf{x}} f_{i_j}(\mathbf{x}_l, \mathbf{y}_l), \mathbf{0}_{d_y}), (\mathbf{0}_{d_x}, -\nabla_{\mathbf{y}} f_{i_j}(\mathbf{x}_l, \mathbf{y}_l)) \text{ for } l < j \leq t,\\ &(\nabla_{\mathbf{x}} f(\mathbf{u}, \mathbf{v}), \mathbf{0}_{d_y}), (\mathbf{0}_{d_x}, -\nabla_{\mathbf{y}} f(\mathbf{u}, \mathbf{v})) \text{ for } (\mathbf{u}, \mathbf{v}) \in \mathcal{H}\big\}\end{aligned}$$

(v) Projection step: $\mathbf{x}_t = \mathcal{P}_{\mathcal{X}}(\widetilde{\mathbf{x}}_t), \mathbf{y}_t = \mathcal{P}_{\mathcal{Y}}(\widetilde{\mathbf{y}}_t)$.

(vi) Output $(\mathbf{x}_t, \mathbf{y}_t)$, or set $t + 1$ to t and go back to step (ii).

Let $\mathscr{A}$ be the class of all such PIFO algorithms. A PIFO algorithm becomes an IFO algorithm if it queries the IFO at step (ii).

Remark. We remark on some details in our definition of PIFO algorithms.

(i) The random vector sequence $\{(i_t, a_t)\}_{t \geq 1}$ are mutually independent and each i_t is also independent of a_t.

(ii) $\mathcal{H}$ is the set of points where FO is called and simultaneous PIFO queries [FLLZ18, ZXG18, LYHZ20] are allowed. Previous PIFO or FO queries can be reused. At step t, the algorithm has access to $(\nabla_{\mathbf{x}} f_{i_t}(\mathbf{x}_0, \mathbf{y}_0), -\nabla_{\mathbf{y}} f_{i_t}(\mathbf{x}_0, \mathbf{y}_0)), \ldots, (\nabla_{\mathbf{x}} f_{i_t}(\mathbf{x}_{t-1}, \mathbf{y}_{t-1}), -\nabla_{\mathbf{y}} f_{i_t}(\mathbf{x}_{t-1}, \mathbf{y}_{t-1}))$ with shared i_t as well as gradient information obtained at previous steps, i.e., $(\nabla_{\mathbf{x}} f_{i_j}(\mathbf{x}_l, \mathbf{y}_l), -\nabla_{\mathbf{y}} f_{i_j}(\mathbf{x}_l, \mathbf{y}_l))$ for $l < j < t$.

(iii) When f_i is not convex-concave, γ should be chosen such that $f_i(\mathbf{u}, \mathbf{v}) + \frac{1}{2\gamma}\|\mathbf{x} - \mathbf{u}\|_2^2 - \frac{1}{2\gamma}\|\mathbf{y} - \mathbf{v}\|_2^2$ is convex-concave w.r.t. to $(\mathbf{u}, \mathbf{v})$.

(iv) Without loss of generality, we assume that the PIFO algorithm $\mathcal{A}$ starts from $(\mathbf{x}_0, \mathbf{y}_0) = (\mathbf{0}_{d_x}, \mathbf{0}_{d_y})$ to simplify our analysis. Otherwise, we can take $\{\widetilde{f}_i(\mathbf{x}, \mathbf{y}) = f_i(\mathbf{x} + \mathbf{x}_0, \mathbf{y} + \mathbf{y}_0)\}_{i=1}^n$ into consideration.

(v) Let $p_i = \mathbb{P}_{Z \sim \mathcal{D}}[Z = i]$ for $i \in [n]$. The distribution $\mathcal{D}$ can be the uniform distribution or based on the smoothness of the component functions, e.g., $p_i \propto L_i$ [XZ14] or $p_i \propto L_i^2$ [AZ18] for $i \in [n]$, where L_i is the smoothness parameter of f_i. We can assume that $p_1 \leq p_2 \leq \cdots \leq p_n$ by rearranging the component functions $\{f_i\}_{i=1}^n$. Suppose that $p_{s_1} \leq p_{s_2} \leq \cdots \leq p_{s_n}$ where $\{s_i\}_{i=1}^n$ is a permutation of $[n]$. We can consider $\{\widehat{f}_i\}_{i=1}^n$ and categorical distribution $\mathcal{D}'$ such that the algorithm draws $\widehat{f}_i \triangleq f_{s_i}$ with probability p_{s_i} instead.

Recall that by setting $\mathcal{Y}$ as a singleton, we can obtain the definition of IFO and PIFO algorithms for finite-sum minimization problems.

We emphasize that only the proximal operator of the individual component function f_i is allowed. The algorithm is not accessible to the proximal operator of the averaged function f. In practice, each f_i usually depends on a single sample and enjoys a simple form [ZX17, DCL+17, LZ17, CJST19]. Then $\operatorname{prox}_{f_i}^{\gamma}(\mathbf{x}, \mathbf{y})$ is easy to calculate. However, computing $\operatorname{prox}_{f}^{\gamma}(\mathbf{x}, \mathbf{y})$ is as hard as solving the original problem (1). To see this, just let $\gamma \to \infty$.

Methods for minimization problems Clearly, methods such as SAGA [DBLJ14] and PointSAGA [Def16] belong to PIFO algorithms, since these methods only calculate the full gradient at the first iteration. Other methods such as SVRG [JZ13] and Katyusha [AZ17a] have two loops and the full gradient needs to be calculated periodically at the beginning of the outer loop. Although these two-loop methods do not satisfy our definition, their loopless variants do. These loopless variants only have one loop and whether to compute the full gradient depends on a coin toss with a small head probability, i.e., q in Definition 10. [KHR20] have shown that L-SVRG and L-Katyusha enjoy the same theoretical properties as the original methods. With a constant q, these loopless methods can also be viewed as two-loop methods with a random inner-loop size that obeys the geometric distribution with success probability q. Other loopless methods that satisfy our definition include KatyushaX [AZ18], L2S [LMG20], PAGE [LBZR21], Anita [Li21] and so on. For these methods, the order of q is usually $\Theta(1/n)$. And it suffices to set $c_0 = 2$.

Now we consider catalyst accelerated methods. It looks like these methods do not satisfy our definition, since they have two loops and the full gradient needs to be calculated at the beginning of the outer loop. Nevertheless, we can slightly change them without affecting the convergence rate. Firstly, we can replace the algorithm used to solve the inner-loop subproblem, e.g., SVRG, with its loopless variant. Secondly, the complexity of the inner-loop is at least of the order $\Omega(n)$ (all the components need to be sampled at least once). Now we remove the full gradient step at the beginning of the outer loop and do not update the current point until the FO is called. In expectation, we need $\Theta(1/q)$ more steps. Thus, if we choose $q = \Theta(1/n)$, such a change makes no difference to the order of the complexity.

Methods for minimax problems One can check SAGA [PB16] and PointSAGA [Def16] are PIFO algorithms. Although SVRG [PB16] does not satisfy our definition, we believe a loopless variant of it can share the same convergence properties. Existing loopless methods that belong to PIFO algorithms include L-SVRHG [LBJM$^+$20] and L-SVRE[2] [AM22]. Moreover, similar to the analysis above, the catalyst accelerated methods in [LXZZ21, ZYG$^+$21] also satisfy our definition. For these methods, the order of q is still $\Theta(1/n)$ and we can set $c_0 = 2$.

Finally, we emphasize that all the methods analyzed above except PointSAGA [DBLJ14, Def16] are also IFO algorithms. From the results in Table 1 and the analysis in Section 6, we find that IFO algorithms are powerful enough for smooth functions.

4 Framework of Construction

In this section, we introduce the framework of our construction to prove the lower bound for Problem (1). In Section 4.1, we give the definition of the optimization complexity. In Section 4.2, we construct the hard instances used to prove the lower bound and present some fundamental lemmas. Now we first highlight the key idea of our construction.

Key idea To construct the hard instance, we partition the tridiagonal matrix in [Nes13] into n groups and each component function is defined in terms of only one group. Then the hard instance satisfies a variant of *zero-chain* property: starting from the origin, only when a specific component is drawn, can we increase the nonzero elements of the current point by at most 2. And the number of PIFO calls required to draw this component obeys the geometric distribution. Once

[2]The method was renamed by [LXZZ21] and we adopt the new name.

we prove that we cannot obtain any ε-suboptimal solution or ε-stationary point unless we span all the dimensions, the complexity can be lower bounded by the concentration inequality of geometric distributions, i.e., Lemma 1. As a comparison, previous span-based constructions [LZ17, ZG19] partition the variable and the number of nonzero elements of the current point can increase no matter which component is drawn. A more detailed analysis is deferred to Section 6.1.

4.1 Optimization Complexity

Before presenting the definition of the optimization complexity, we first introduce the function class we consider. Define the primal function as $\phi_f(\mathbf{x}) = \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y})$ and the dual function as $\psi_f(\mathbf{y}) = \min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x},\mathbf{y})$.

Function class We develop lower bounds for PIFO algorithms that find a suboptimal solution or near stationary point of Problem (1) in the following sets.

$$\mathcal{F}_{\mathrm{CC}}(R_x,R_y,L,\mu_x,\mu_y) = \Bigg\{ f(\mathbf{x},\mathbf{y}) = \frac{1}{n}\sum_{i=1}^n f_i(\mathbf{x},\mathbf{y}) \;\Big|\; f\colon \mathcal{X}\times\mathcal{Y}\to\mathbb{R},\ \mathrm{diam}(\mathcal{X}) \le 2R_x,,$$
$$\mathrm{diam}(\mathcal{Y}) \le 2R_y,\ \{f_i\}_{i=1}^n \text{ is } L\text{-average smooth}, f \text{ is } (\mu_x,\mu_y)\text{-convex-concave} \Bigg\}$$
$$\mathcal{F}_{\mathrm{NCC}}(\Delta,L,\mu_x,\mu_y) = \Bigg\{ f(\mathbf{x},\mathbf{y}) = \frac{1}{n}\sum_{i=1}^n f_i(\mathbf{x},\mathbf{y}) \;\Big|\; f\colon \mathcal{X}\times\mathcal{Y}\to\mathbb{R},\ \phi(\mathbf{0}) - \inf_{\mathbf{x}\in\mathcal{X}}\phi(\mathbf{x}) \le \Delta,$$
$$\{f_i\}_{i=1}^n \text{ is } L\text{-average smooth}, f \text{ is } (-\mu_x,\mu_y)\text{-convex-concave} \Bigg\}$$

We remark that for the second class, μ_x measures how nonconvex the function is. A natural upper bound of μ_x is L. Moreover, we do not specify the dimensions of the feasible set. That is to say, the two classes include functions defined on $\mathcal{X}\times\mathcal{Y} \subseteq \mathbb{R}^{d_x}\times\mathbb{R}^{d_y}$ with any positive integers d_x and d_y.

Optimization complexity Then we formally define the optimization complexity.

Definition 11. *For a function f, a PIFO algorithm $\mathcal{A}$ and a tolerance $\varepsilon > 0$, the number of queries to PIFO needed by $\mathcal{A}$ to find an ε-suboptimal solution to Problem (1) or an ε-stationary point of $\phi_f(\mathbf{x})$ is defined as*

$$T(\mathcal{A},f,\varepsilon) = \begin{cases} \inf\left\{T\in\mathbb{N} \mid \mathbb{E}\phi_f(\mathbf{x}_{\mathcal{A},T}) - \mathbb{E}\psi_f(\mathbf{y}_{\mathcal{A},T}) < \varepsilon\right\}, & \text{if } f\in\mathcal{F}_{\mathrm{CC}}(R_x,R_y,L,\mu_x,\mu_y), \\ \inf\left\{T\in\mathbb{N} \mid \mathbb{E}\left\|\nabla\phi_f(\mathbf{x}_{\mathcal{A},T})\right\|_2 < \varepsilon\right\}, & \text{if } f\in\mathcal{F}_{\mathrm{NCC}}(\Delta,L,\mu_x,\mu_y), \end{cases}$$

where $(\mathbf{x}_{\mathcal{A},T},\mathbf{y}_{\mathcal{A},T})$ is the point obtained by the algorithm $\mathcal{A}$ at time-step $T-1$. The optimization complexity with respect to the two function classes is defined as [3]

$$\mathfrak{m}^{\mathrm{CC}}(\varepsilon,R_x,R_y,L,\mu_x,\mu_y) \triangleq \inf_{\mathcal{A}\in\mathscr{A}} \sup_{f\in\mathcal{F}_{\mathrm{CC}}(R_x,R_y,L,\mu_x,\mu_y)} T(\mathcal{A},f,\varepsilon).$$
$$\mathfrak{m}^{\mathrm{NCC}}(\varepsilon,\Delta,L,\mu_x,\mu_y) \triangleq \inf_{\mathcal{A}\in\mathscr{A}} \sup_{f\in\mathcal{F}_{\mathrm{NCC}}(\Delta,L,\mu_x,\mu_y)} T(\mathcal{A},f,\varepsilon)$$

[3] Our definition follows from [CDHS17a].

When f is convex-concave, the functions we consider have a bounded feasible set and L-average smooth components. By Sion's minimax theorem, the strong duality condition holds. Then the primal-dual gap is a natural measurement of the optimality[4]. Specially, if f is strongly-convex-strongly-concave, the saddle point $(\mathbf{x}^*, \mathbf{y}^*)$ is unique and the distance to the saddle point is also a measurement of the optimality. And we have $\frac{\mu_x}{2}\|\mathbf{x}-\mathbf{x}^*\|_2^2 + \frac{\mu_y}{2}\|\mathbf{y}-\mathbf{y}^*\|_2^2 \le \phi_f(\mathbf{x}) - \psi_f(\mathbf{y}) \le \frac{L}{2}\|\mathbf{x}-\mathbf{x}^*\|_2^2 + \frac{L}{2}\|\mathbf{y}-\mathbf{y}^*\|_2^2$. The results in Section 5 show that the optimal methods have linear convergence rates in this case. Thus, the complexities w.r.t. the two measurements are equivalent up to log factors. As for the nonconvex-strongly-concave case, we aim to find the stationary point of the primal function and use the norm of the gradient of the primal function as the measurement.

Note that we use the number of PIFO calls to measure the complexity. We claim that the infrequent FO calls do not influence the order of this complexity. At each step, the FO is called with probability $q = \mathcal{O}\left(\frac{1}{n}\right)$. Since the computation cost of each FO call is no larger than that of n PIFO calls, the total cost of PO calls is no larger than the order of the number of PIFO calls in expectation. Thus our definition of complexity is reasonable, due to that we usually ignore the influence of constants.

4.2 The Hard Instances

In this subsection, we construct the (unscaled) hard instances used to prove the lower bound. The constructions for convex-concave case and the nonconvex-strongly-concave case are slightly different and presented in Sections 4.2.1 and 4.2.2 respectively. However, they are both based on the following class of matrices, which is also used in the proof of lower bounds in deterministic minimax optimization [OX18, ZHZ19]:

$$\mathbf{B}(m,\omega,\zeta) = \begin{bmatrix} \omega & & & & \\ 1 & -1 & & & \\ & 1 & -1 & & \\ & & \ddots & \ddots & \\ & & & 1 & -1 \\ & & & & \zeta \end{bmatrix} \in \mathbb{R}^{(m+1)\times m} \tag{3}$$

In fact, $\mathbf{B}(m,\omega,\zeta)^\top \mathbf{B}(m,\omega,\zeta)$ is the widely-used tridiagonal matrix in the analysis of lower bounds for convex optimization [Nes13, LZ17, ZG19].

For convenience, we denote the l-th row of the matrix $\mathbf{B}(m,\omega,\zeta)$ by $\mathbf{b}_{l-1}(m,\omega,\zeta)^\top$. To construct a hard instance for the finite-sum optimization problem, we partition the row vectors of $\mathbf{B}(m,\omega,\zeta)$ according to the index sets $\mathcal{L}_i = \left\{l : 0 \le l \le m, l \equiv i-1 \pmod n\right\}$. The i-th component is constructed in terms of $\{\mathbf{b}_l(m,\omega,\zeta) : l \in \mathcal{L}_i\}$. This way of partition is different from those used in [LZ17] and [ZG19] (a detailed comparison is deferred to Section 6.1). We find that the $\mathbf{b}_l(m,\omega,\zeta)$ have at most two nonzero elements and the vectors whose indices lie in the same index sets are mutually orthogonal, as long as $n \ge 2$.

[4]When f is strongly-convex-strongly-concave, the boundness of the feasible set is not necessary and we can also use the prima-dual gap at the initial point as the parameter to define the function class.

4.2.1 Convex-Concave Case

The hard instance for the convex-concave case is constructed as

$$\min_{\mathbf{x}\in\mathcal{X}}\max_{\mathbf{y}\in\mathcal{Y}} r^{\mathrm{CC}}(\mathbf{x},\mathbf{y};m,\zeta,\mathbf{c}^{\mathrm{CC}}) \triangleq \frac{1}{n}\sum_{i=1}^{n} r_i^{\mathrm{CC}}(\mathbf{x},\mathbf{y};m,\zeta,\mathbf{c}^{\mathrm{CC}}) \tag{4}$$

where $\mathbf{c}^{\mathrm{CC}} = (c_1^{\mathrm{CC}}, c_2^{\mathrm{CC}})$, $\mathcal{X} = \{\mathbf{x}\in\mathbb{R}^m : \|\mathbf{x}\|_2 \le R_x\}, \mathcal{Y} = \{\mathbf{y}\in\mathbb{R}^m : \|\mathbf{y}\|_2 \le R_y\}$ and

$$r_i^{\mathrm{CC}}(\mathbf{x},\mathbf{y};m,\zeta,\mathbf{c}^{\mathrm{CC}}) = \begin{cases} n\sum\limits_{l\in\mathcal{L}_i}\mathbf{y}^\top\mathbf{e}_l\mathbf{b}_l(m,0,\zeta)^\top\mathbf{x} + \frac{c_1^{\mathrm{CC}}}{2}\|\mathbf{x}\|_2^2 - \frac{c_2^{\mathrm{CC}}}{2}\|\mathbf{y}\|_2^2 - n\langle\mathbf{e}_1,\mathbf{x}\rangle, & \text{for } i=1,\\ n\sum\limits_{l\in\mathcal{L}_i}\mathbf{y}^\top\mathbf{e}_l\mathbf{b}_l(m,0,\zeta)^\top\mathbf{x} + \frac{c_1^{\mathrm{CC}}}{2}\|\mathbf{x}\|_2^2 - \frac{c_2^{\mathrm{CC}}}{2}\|\mathbf{y}\|_2^2, & \text{for } i=2,3,\ldots,n \end{cases}$$

Note that $\mathbf{b}_0(m,0,\zeta) = \mathbf{0}$, which implies that this hard instance is based on the last m rows of $\mathbf{B}(m,\omega,\zeta)$. Then we can determine the smoothness and strong convexity coefficients of r_i^{CC} as follows.

Proposition 1. *For $c_1^{\mathrm{CC}}, c_2^{\mathrm{CC}} \ge 0$ and $0 \le \zeta \le \sqrt{2}$, we have that r_i^{CC} is L-smooth and $(c_1^{\mathrm{CC}}, c_2^{\mathrm{CC}})$-convex-concave, and $\{r_i^{\mathrm{CC}}\}_{i=1}^n$ is L'-average smooth, where*

$$L = \sqrt{4n^2 + 2\max\{c_1^{\mathrm{CC}}, c_2^{\mathrm{CC}}\}^2} \quad \text{and} \quad L' = \sqrt{8n + 2\max\{c_1^{\mathrm{CC}}, c_2^{\mathrm{CC}}\}^2}$$

We find if $\max\{c_1^{\mathrm{CC}}, c_2^{\mathrm{CC}}\} = \mathcal{O}(\sqrt{n})$, then $L/L' = \Theta(\sqrt{n})$.
Define the subspaces $\{\mathcal{F}_k\}_{k=0}^m$ as

$$\mathcal{F}_k = \begin{cases} \mathrm{span}\{\mathbf{e}_1,\mathbf{e}_2,\ldots,\mathbf{e}_k\}, & \text{for } 1\le k\le m,\\ \{\mathbf{0}\}, & \text{for } k=0 \end{cases} \tag{5}$$

Now we show that the hard instance satisfies a variant of the *zero-chain* property [CDHS17a].

Lemma 2. *Suppose that $n \ge 2$ and $\mathcal{F}_{-1} = \mathcal{F}_0$. Then for $(\mathbf{x},\mathbf{y}) \in \mathcal{F}_k \times \mathcal{F}_{k-1}$ and $0 \le k < m$, we have that*

$$\begin{pmatrix} \nabla_{\mathbf{x}} r_i^{\mathrm{CC}}(\mathbf{x},\mathbf{y}) \\ -\nabla_{\mathbf{y}} r_i^{\mathrm{CC}}(\mathbf{x},\mathbf{y}) \end{pmatrix}, \mathrm{prox}_{r_i^{\mathrm{CC}}}^{\gamma}(\mathbf{x},\mathbf{y}) \in \begin{cases} \mathcal{F}_{k+1}\times\mathcal{F}_k, & \text{if } i \equiv k+1 \,(\mathrm{mod}\ n),\\ \mathcal{F}_k\times\mathcal{F}_{k-1}, & \text{otherwise} \end{cases}$$

where we omit the parameters of r_i^{CC} to simplify the presentation.

If the current point is $(\mathbf{x},\mathbf{y})$, the information brought by the PIFO call at $(\mathbf{x},\mathbf{y})$ will not increase the nonzero elements of $(\mathbf{x},\mathbf{y})$ unless a specific component function is drawn. Moreover, if such a specific component is drawn, the increase is at most 2. This variant of *zero-chain* property is also different from the conventional *zero-chain* property in finite-sum minimization problems [LZ17, ZG19], where regardless of which component is drawn, the nonzero elements of the current point can increase. Such a difference comes from different ways of partitioning and ensures that our construction requires a lower dimension (see the analysis in Section 6.2). The proofs of Proposition 1 and Lemma 2 are given in Appendix C.1.

When we apply a PIFO algorithm $\mathcal{A}$ to solve Problem (4), Lemma 2 implies that $\mathbf{x}_t = \mathbf{y}_t = \mathbf{0}$ will hold until algorithm $\mathcal{A}$ draws the component f_1 or calls the FO. Then, for any $t < T_1 =$

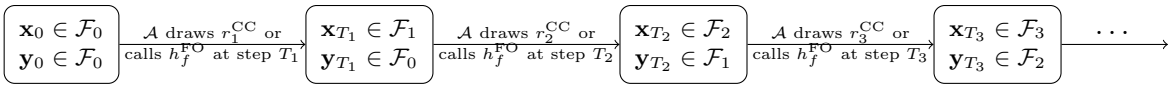

Figure 1: An illustration of the process of solving the Problem (4) with a PIFO algorithm $\mathcal{A}$.

$\min_t\{t : i_t = 1 \text{ or } a_t = 1\}$, we have $\mathbf{x}_t, \mathbf{y}_t \in \mathcal{F}_0$ while $\mathbf{x}_{T_1} \in \mathcal{F}_1$ and $\mathbf{y}_{T_1} \in \mathcal{F}_0$. The value of T_1 can be regarded as the smallest integer such that $\mathbf{x}_{T_1} \in \mathcal{F}_1 \setminus \mathcal{F}_0$ could hold. Similarly, for $T_1 \le t < T_2 = \min_t\{t > T_1 : i_t = 2 \text{ or } a_t = 1\}$ it holds that $\mathbf{x}_t \in \mathcal{F}_1$ and $\mathbf{y}_t \in \mathcal{F}_0$ while we can ensure that $\mathbf{x}_{T_2} \in \mathcal{F}_2$ and $\mathbf{y}_{T_2} \in \mathcal{F}_1$. Figure 1 illustrates this optimization process.

We can define T_k to be the smallest integer such that $\mathbf{x}_{T_k} \in \mathcal{F}_k \backslash \mathcal{F}_{k-1}$ and $\mathbf{y}_{T_k} \in \mathcal{F}_{k-1} \backslash \mathcal{F}_{k-2}$ could hold. The following corollary demonstrates that we can connect T_k to geometrically distributed random variables.

Corollary 1. *Assume we employ a PIFO algorithm $\mathcal{A}$ to solve Problem (4). Let*

$$T_0 = 0, \quad \textit{and} \quad T_k = \min_t\{t : t > T_{k-1}, i_t \equiv k \pmod n \textit{ or } a_t = 1\} \quad \textit{for } k \ge 1 \tag{6}$$

Then we have

$$(\mathbf{x}_t, \mathbf{y}_t) \in \mathcal{F}_{k-1} \times \mathcal{F}_{k-2}, \qquad \textit{for } t < T_k, k \ge 1$$

Moreover, the random variables $\{Y_k\}_{k\ge1}$ such that $Y_k \triangleq T_k - T_{k-1}$ are mutually independent and Y_k follows a geometric distribution with success probability $p_{k'} + q - p_{k'}q$ where $k' \equiv k \pmod n$ and $l \in [n]$.

The basic idea of our analysis is that we guarantee that the ε-suboptimal solution of Problem (4) does not lie in $\mathcal{F}_k \times \mathcal{F}_k$ for $k < m$ and assure that the PIFO algorithm extends the space $\mathrm{span}\{(\mathbf{x}_0, \mathbf{y}_0), (\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_t, \mathbf{y}_t)\}$ slowly with t increasing. By Corollary 1, we know that $\mathrm{span}\{(\mathbf{x}_0, \mathbf{y}_0), (\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_{T_k-1}, \mathbf{y}_{T_k-1})\} \subseteq \mathcal{F}_{k-1} \times \mathcal{F}_{k-1}$. Hence, T_k is the quantity that measures how $\mathrm{span}\{(\mathbf{x}_0, \mathbf{y}_0), (\mathbf{x}_1, \mathbf{y}_1), \ldots, (\mathbf{x}_t, \mathbf{y}_t)\}$ expands. Note that T_k can be written as the sum of geometrically distributed random variables. Recalling Lemma 1, we can obtain how many PIFO calls we need.

Lemma 3. *If M satisfies $1 \le M < m$,*

$$\min_{\substack{\mathbf{x}\in\mathcal{X}\cap\mathcal{F}_M \\ \mathbf{y}\in\mathcal{Y}\cap\mathcal{F}_M}} \left(\max_{\mathbf{v}\in\mathcal{Y}} r^{\mathrm{CC}}(\mathbf{x}, \mathbf{v}) - \min_{\mathbf{u}\in\mathcal{X}} r^{\mathrm{CC}}(\mathbf{u}, \mathbf{y}) \right) \ge 9\varepsilon \tag{7}$$

and $N = \frac{n(M+1)}{4(1+c_0)}$, then we have

$$\min_{t\le N} \mathbb{E} \left(\max_{\mathbf{v}\in\mathcal{Y}} r^{\mathrm{CC}}(\mathbf{x}_t, \mathbf{v}) - \min_{\mathbf{u}\in\mathcal{X}} r^{\mathrm{CC}}(\mathbf{u}, \mathbf{y}_t) \right) \ge \varepsilon$$

Note that rescaling will not influence the *zero-chain* property. Thus Lemma 3 still holds for any rescaled version of r^{CC}. It remains to pick up the parameters carefully, obtain a condition of the form (7) and then estimate the order of N. These steps depend on the specific problem and are deferred to Sections 5.2 to 5.4 and Appendices D.1 to D.3.

The proofs of Corollary 1 and Lemma 3 are given in Appendix C.2.

4.2.2 Nonconvex-Strongly-Concave Case

For the nonconvex-strongly-concave case, the hard instance is constructed as

$$\min_{\mathbf{x}\in\mathbb{R}^m}\max_{\mathbf{y}\in\mathbb{R}^m} r^{\mathrm{NCC}}(\mathbf{x},\mathbf{y};m,\omega,\mathbf{c}^{\mathrm{NCC}}) \triangleq \frac{1}{n}\sum_{i=1}^{n} r_i^{\mathrm{NCC}}(\mathbf{x},\mathbf{y};m,\omega,\mathbf{c}^{\mathrm{NCC}}) \tag{8}$$

where $\mathbf{c}^{\mathrm{NCC}} = (c_1^{\mathrm{NCC}}, c_2^{\mathrm{NCC}}, c_3^{\mathrm{NCC}})$ and

$$r_i^{\mathrm{NCC}}(\mathbf{x},\mathbf{y};m,\omega,\mathbf{c}^{\mathrm{NCC}}) = \begin{cases} n\sum\limits_{l\in\mathcal{L}_i}\mathbf{y}^\top\mathbf{e}_{l+1}\mathbf{b}_l(m,\omega,0)^\top\mathbf{x} - \frac{c_1^{\mathrm{NCC}}}{2}\|\mathbf{y}\|_2^2 + c_2^{\mathrm{NCC}}\sum\limits_{i=1}^{m-1}\Gamma(c_3^{\mathrm{NCC}}x_i) - n\langle\mathbf{e}_1,\mathbf{y}\rangle, & \text{for } i=1,\\ n\sum\limits_{l\in\mathcal{L}_i}\mathbf{y}^\top\mathbf{e}_{l+1}\mathbf{b}_l(m,\omega,0)^\top\mathbf{x} - \frac{c_1^{\mathrm{NCC}}}{2}\|\mathbf{y}\|_2^2 + c_2^{\mathrm{NCC}}\sum\limits_{i=1}^{m-1}\Gamma(c_3^{\mathrm{NCC}}x_i), & \text{for } i=2,3,\ldots,n \end{cases}$$

The nonconvex function $\Gamma:\mathbb{R}\to\mathbb{R}$ is

$$\Gamma(x) \triangleq 120\int_1^x \frac{t^2(t-1)}{1+t^2}dt$$

which was introduced by [CDHS17b]. Since $\mathbf{b}_m(m,\omega,0) = \mathbf{0}_m$, the vector $\mathbf{e}_{m+1}$ will not appear in the definition of r^{NCC}. Thus r^{NCC} is well-defined and only depends on the first m rows of $\mathbf{B}(m,\omega,\zeta)$. We can determine the smoothness and strong convexity coefficients of r_i^{NCC} as follows.

Proposition 2. *For $c_1^{\mathrm{NCC}} \geq 0$, $c_2^{\mathrm{NCC}}, c_3^{\mathrm{NCC}} > 0$ and $0 \leq \omega \leq \sqrt{2}$, we have that r_i^{NCC} is L-smooth and $\big(-45(\sqrt{3}-1)c_2^{\mathrm{NCC}}(c_3^{\mathrm{NCC}})^2, c_1^{\mathrm{NCC}}\big)$-convex-concave, and $\{r_i^{\mathrm{NCC}}\}_{i=1}^n$ is L'-average smooth, where*

$$L = \sqrt{4n^2 + 2(c_1^{\mathrm{NCC}})^2} + 180c_2^{\mathrm{NCC}}(c_3^{\mathrm{NCC}})^2 \quad \text{and} \quad L' = 2\sqrt{4n + (c_1^{\mathrm{NCC}})^2 + 16200(c_2^{\mathrm{NCC}})^2(c_3^{\mathrm{NCC}})^4}$$

We find if $\max\{c_1^{\mathrm{NCC}}, c_2^{\mathrm{NCC}}(c_3^{\mathrm{NCC}})^2\} = \mathcal{O}(\sqrt{n})$, then $L/L' = \Theta(\sqrt{n})$.

The next lemma shows that the r_i^{NCC} share the similar *zero-chain* property as Lemma 2.

Lemma 4. *Suppose that $n \geq 2$, $c_2^{\mathrm{NCC}}, c_3^{\mathrm{NCC}} > 0$ and $\gamma < \frac{\sqrt{2}+1}{60c_2^{\mathrm{NCC}}(c_3^{\mathrm{NCC}})^2}$. If $(\mathbf{x},\mathbf{y}) \in \mathcal{F}_k \times \mathcal{F}_k$ and $0 \leq k < m-1$, we have that*

$$\begin{pmatrix} \nabla_x r_i^{\mathrm{NCC}}(\mathbf{x},\mathbf{y}) \\ -\nabla_y r_i^{\mathrm{NCC}}(\mathbf{x},\mathbf{y}) \end{pmatrix}, \mathrm{prox}_{r_i^{\mathrm{NCC}}}^{\gamma}(\mathbf{x},\mathbf{y}) \in \begin{cases} \mathcal{F}_{k+1}\times\mathcal{F}_{k+1}, & \text{if } i \equiv k+1 \,(\mathrm{mod}\ n),\\ \mathcal{F}_k\times\mathcal{F}_k, & \text{otherwise} \end{cases}$$

where we omit the parameters of r_i^{NCC} to simplify the presentation.

The proofs of Proposition 2 and Lemma 4 are given in Appendix C.3.

It is worth emphasizing that the assumption on γ naturally holds. Recall that the choice of γ should satisfy that $r_i(\mathbf{u},\mathbf{y}) + \frac{1}{2\gamma}\|\mathbf{x}-\mathbf{u}\|_2^2 - \frac{1}{2\gamma}\|\mathbf{y}-\mathbf{v}\|_2^2$ is convex-concave in $(\mathbf{u},\mathbf{v})$. Proposition 2 implies that we must have $\gamma \leq \frac{1}{45(\sqrt{3}-1)c_2^{\mathrm{NCC}}(c_3^{\mathrm{NCC}})^2} \leq \frac{\sqrt{2}+1}{60c_2^{\mathrm{NCC}}(c_3^{\mathrm{NCC}})^2}$.

When we apply a PIFO algorithm to solve Problem (8), the optimization process is similar to the process related to Problem (4). We demonstrate the optimization process in Figure 2 and present a formal statement in Corollary 2.

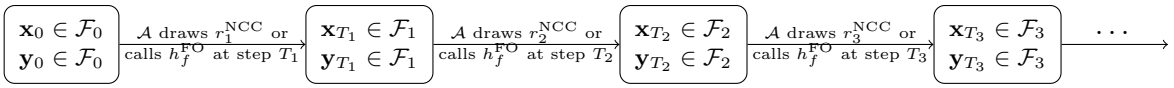

Figure 2: An illustration of the process of solving the Problem (8) with a PIFO algorithm $\mathcal{A}$.

Corollary 2. *Assume we employ a PIFO algorithm $\mathcal{A}$ to solve Problem (8). Let*

$$T_0 = 0, \quad \textit{and} \quad T_k = \min_t\{t : t > T_{k-1}, i_t \equiv k \pmod n \textit{ or } a_t = 1\} \quad \textit{for } k \geq 1$$

Then we have

$$(\mathbf{x}_t, \mathbf{y}_t) \in \mathcal{F}_{k-1} \times \mathcal{F}_{k-1}, \qquad \textit{for } t < T_k, k \geq 1$$

Moreover, the random variables $\{Y_k\}_{k\geq 1}$ such that $Y_k \triangleq T_k - T_{k-1}$ are mutual independent and Y_k follows a geometric distribution with success probability $p_{k'} + q - p_{k'}q$ where $k' \equiv k \pmod n$ and $l \in [n]$.

The proof of Corollary 2 is similar to that of Corollary 1. Furthermore, the prime-dual gap in Lemma 3 can be replaced with the gradient norm of the primal function in the nonconvex-strongly-concave case.

Lemma 5. *Let $\phi_{r^{\mathrm{NCC}}}(\mathbf{x}) \triangleq \max_{\mathbf{y}\in\mathbb{R}^m} r^{\mathrm{NCC}}(\mathbf{x}, \mathbf{y})$. If M satisfies $1 \leq M < m$ and*

$$\min_{\mathbf{x}\in\mathcal{F}_M} \left\|\nabla \phi_{r^{\mathrm{NCC}}}(\mathbf{x})\right\|_2 \geq 9\varepsilon \tag{9}$$

and $N = \frac{n(M+1)}{4(1+c_0)}$, then we have

$$\min_{t\leq N} \mathbb{E}\left\|\nabla \phi_{r^{\mathrm{NCC}}}(\mathbf{x}_t)\right\|_2 \geq \varepsilon$$

Lemma 5 also holds for any rescaled version of r^{NCC}. It remains to pick up the parameters carefully, obtain a condition of the form (9) and then estimate the order of N. The details are deferred to Section 5.5 and Appendix D.4.

5 Lower Complexity Bounds for the Minimax Problems

In this section, we focus on the minimax problem (1), which is restated as follows.

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}, \mathbf{y}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}, \mathbf{y})$$

We assume that the function class $\{f_i(\mathbf{x}, \mathbf{y})\}_{i=1}^n$ is L-average smooth, and the feasible sets $\mathcal{X}$ and $\mathcal{Y}$ are closed and convex. In addition, $f(\mathbf{x}, \mathbf{y})$ is convex in $\mathbf{x}$ and concave in $\mathbf{y}$ or $f(\mathbf{x}, \mathbf{y})$ is nonconvex in $\mathbf{x}$ and strongly-concave in $\mathbf{y}$. The lower bound results are shown in Section 5.1. The detailed constructions for different cases are shown in Sections 5.2 to 5.5. Finally, in Section 5.6, we consider the more constrained case where each f_i is L-smooth and briefly introduce the results.

5.1 Main Results

Recall that the comparison of the upper and lower bounds is already shown in Table 1. In this subsection, we present the formal statements of our lower bounds and give some interpretation. We emphasize that the methods in [LXZZ21, ZYG+21] are just IFO algorithms from the analysis in Section 3.2, which implies PIFO oracles are not much more powerful than IFO oracles.

We start with the case where the objective function f is μ_x-strongly-convex in $\mathbf{x}$ and μ_y-strongly-concave in $\mathbf{y}$. Define the condition numbers $\kappa_x \triangleq L/\mu_x$ and $\kappa_y \triangleq L/\mu_y$. Without loss of generality, we assume $\mu_x \leq \mu_y$. According to the relationship between κ_x, κ_y and n, we can classify the problem into three cases: (a) f is extremely ill-conditioned w.r.t. both $\mathbf{x}$ and $\mathbf{y}$, i.e., $\kappa_x, \kappa_y = \Omega(\sqrt{n})$; (b) f is only extremely ill-conditioned w.r.t. $\mathbf{x}$, i.e., $\kappa_x = \Omega(\sqrt{n}), \kappa_y = \mathcal{O}(\sqrt{n})$; (c) f is relatively well-conditioned w.r.t. both $\mathbf{x}$ and $\mathbf{y}$, i.e., $\kappa_x, \kappa_y = \mathcal{O}(\sqrt{n})$. For the three cases, we can prove different lower bounds as follows.

Theorem 1. *Let $n \geq 4$ be a positive integer and $L, \mu_x, \mu_y, R_x, R_y, \varepsilon$ be positive parameters. Assume additionally that $\kappa_x \geq \kappa_y \geq 2$ and $\varepsilon \leq \min\left\{\frac{n\mu_x R_x^2}{800\kappa_x\kappa_y}, \frac{\mu_x R_x^2}{720}, \frac{\mu_y R_y^2}{800}\right\}$. Then we have*

$$\mathfrak{m}^{\mathrm{CC}}(\varepsilon, R_x, R_y, L, \mu_x, \mu_y) = \begin{cases} \Omega\left(\left(n+\sqrt{\kappa_x\kappa_y n}\right)\log\left(1/\varepsilon\right)\right), & \text{for } \kappa_x, \kappa_y = \Omega(\sqrt{n}), \\ \Omega\left(\left(n+n^{3/4}\sqrt{\kappa_x}\right)\log\left(1/\varepsilon\right)\right), & \text{for } \kappa_x = \Omega(\sqrt{n}), \kappa_y = \mathcal{O}(\sqrt{n}), \\ \Omega\left(n\right), & \text{for } \kappa_x, \kappa_y = \mathcal{O}(\sqrt{n}) \end{cases}$$

We mainly focus on the first two cases where at least one condition number is of the order $\Omega(\sqrt{n})$. Then the lower bound can be summarized as $\Omega(\sqrt{n(\sqrt{n}+\kappa_x)(\sqrt{n}+\kappa_y)}\log(1/\varepsilon))$, as shown in Table 1.

Some works focus on the balanced case $\kappa_x = \kappa_y$. For example, the upper bound of Accelerated SVRG/SAGA [PB16] is $\mathcal{O}\left(\left(n + \frac{\sqrt{n}L}{\min\{\mu_x,\mu_y\}}\right)\log(1/\varepsilon)\right)$. L-SVRE [AM22] also achieves the same upper bound[5]. At least for the balanced case, their upper bounds nearly match our lower bound. However, for the unbalanced case, there still exists a gap. [LXZZ21] focus on the unbalanced case. They employ the catalyst technique to accelate L-SVRE and propose the method AL-SVRE, which achieves the upper bound $\widetilde{\mathcal{O}}(\sqrt{n(\sqrt{n}+\kappa_x)(\sqrt{n}+\kappa_y)}\log(1/\varepsilon))$. This bound nearly matches our lower bound for the unbalanced case up to log factors.

Then we consider the lower bound when the objective function is not strongly-convex in $\mathbf{x}$, i.e., $\mu_x = 0$. In this case, only the condition number w.r.t. $\mathbf{y}$ is well-defined. According to the relationship between κ_y and $\sqrt{n}$, we can also split the problem into two cases: (a) f is extremely ill-conditioned w.r.t. $\mathbf{y}$, i.e., $\kappa_y = \Omega(\sqrt{n})$; (b) f is relatively well-conditioned w.r.t. $\mathbf{y}$, i.e., $\kappa_y = \mathcal{O}(\sqrt{n})$. We can prove the lower bounds as follows.

Theorem 2. *Let $n \geq 4$ be a positive integer and $L, \mu_y, R_x, R_y, \varepsilon$ be positive parameters. Assume additionally that $\kappa_y \geq 2$ and $\varepsilon \leq \min\left\{\frac{LR_x^2}{4}, \frac{\mu_y R_y^2}{36}\right\}$. Then we have*

$$\mathfrak{m}^{\mathrm{CC}}(\varepsilon, R_x, R_y, L, 0, \mu_y) = \begin{cases} \Omega\Big(n+R_x n^{3/4}\sqrt{\frac{L}{\varepsilon}}+R_x\sqrt{\frac{nL\kappa_y}{\varepsilon}}+n^{3/4}\sqrt{\kappa_y}\log(\frac{1}{\varepsilon})\Big), & \text{for } \kappa_y{=}\Omega(\sqrt{n}), \\ \Omega\Big(n+R_x n^{3/4}\sqrt{\frac{L}{\varepsilon}}+R_x\sqrt{\frac{nL\kappa_y}{\varepsilon}}\Big), & \text{for } \kappa_y{=}\mathcal{O}(\sqrt{n}) \end{cases}$$

[5]The setting in Section 4.3 of [AM22] is slightly different from ours here. However, the proof of their result can be adapted to strongly-convex-strongly-concave cases.

For both cases, the leading term w.r.t. ε is of the order $\Omega(\sqrt{1/\varepsilon})$ and the only difference between the two bounds is the term $\Omega(n^{3/4}\sqrt{\kappa_y}\log(\frac{1}{\varepsilon}))$, which is usually much smaller than the $\Omega(\sqrt{1/\varepsilon})$ term, especially when ε is small. The upper bound of AL-SVRE [LXZZ21] for this case is $\mathcal{O}\Big(\big(n + R_x n^{3/4}\sqrt{\frac{L}{\varepsilon}} + R_x\sqrt{\frac{nL\kappa_y}{\varepsilon}} + n^{3/4}\sqrt{\kappa_y}\big)\log(\frac{1}{\varepsilon})\Big)$, which nearly matches our lower bound up to log factors.

For the general convex-concave case where $\mu_x = \mu_y = 0$, we have the following lower bound.

Theorem 3. *Let $n \geq 2$ be a positive integer and L, R_x, R_y, ε be positive parameters. Assume additionally that $\varepsilon \leq \frac{L}{4}\min\{R_x^2, R_y^2\}$. Then we have*

$$\mathfrak{m}^{\mathrm{CC}}(\varepsilon, R_x, R_y, L, 0, 0) = \Omega\left(n + \frac{\sqrt{n}LR_xR_y}{\varepsilon} + (R_x + R_y)n^{3/4}\sqrt{\frac{L}{\varepsilon}}\right)$$

The leading term w.r.t. ε is of the order $\Omega(1/\varepsilon)$. If $\varepsilon = \mathcal{O}\left(\frac{LR_x^2R_y^2}{\sqrt{n}(R_x+R_y)^2}\right)$, our lower bound is $\Omega\left(n + \frac{\sqrt{n}LR_xR_y}{\varepsilon}\right)$, which matches the upper bound $\mathcal{O}\left(n + \frac{\sqrt{n}L(R_x^2+R_y^2)}{\varepsilon}\right)$ of [AM22] in terms of n, L and ε. The upper bound of AL-SVRE [LXZZ21] for this case is $\mathcal{O}\Big(\big(n + \frac{\sqrt{n}LR_xR_y}{\varepsilon} + (R_x + R_y)n^{3/4}\sqrt{\frac{L}{\varepsilon}}\big)\log(\frac{1}{\varepsilon})\Big)$, which nearly matches our lower bound up to log factors.

Finally, we give the lower bound when the objective function is nonconvex in $\mathbf{x}$ but strongly-concave in $\mathbf{y}$.

Theorem 4. *Let $n \geq 2$ be a positive integer and $L, \mu_x, \mu_y, R_x, R_y, \varepsilon$ be positive parameters. Assume additionally that $\varepsilon^2 \leq \frac{\Delta L^2\alpha}{435456n\mu_y}$, where $\alpha = \min\left\{1, \frac{128(\sqrt{3}+1)n\mu_x\mu_y}{45L^2}, \frac{32n\mu_y}{135L}\right\}$. Then we have*

$$\mathfrak{m}^{\mathrm{NCC}}(\varepsilon, \Delta, L, \mu_x, \mu_y) = \Omega\left(n + \frac{\Delta L^2\sqrt{\alpha}}{\mu_y\varepsilon^2}\right)$$

For $\kappa_y = L/\mu_y \geq 32n/135$, we have

$$\Omega\left(n + \frac{\Delta L^2\sqrt{\alpha}}{\mu_y\varepsilon^2}\right) = \Omega\left(n + \frac{\Delta L\sqrt{n}}{\varepsilon^2}\min\left\{\sqrt{\kappa_y}, \sqrt{\frac{\mu_x}{\mu_y}}\right\}\right)$$

We mainly focus on the ill-conditioned setting $\kappa_y = \Omega(n)$, where the lower bound has a more concise expression. Recall that μ_x measures the nonconvexity of function. When L is fixed, we must have $\mu_x \leq L$. If we are uninterested in the dependence of the lower bound on μ_x, then we can consider the largest function class $\mathcal{F}_{\mathrm{NCC}}(\Delta, L, L, \mu_y)$. which corresponds to the complexity $\mathfrak{m}^{\mathrm{NCC}}(\varepsilon, \Delta, L, L, \mu_y) = \Omega\left(n + \frac{\Delta L\sqrt{n\kappa_y}}{\varepsilon^2}\right)$[6], as shown in Table 1.

As for the upper bound, [LYHZ20] propose the method SREDA and establish the upper bound $\mathcal{O}\left(n\log(\kappa_y/\varepsilon) + L\kappa_y^2\sqrt{n}\varepsilon^{-2}\right)$ for $n \geq \kappa_y^2$ and $\mathcal{O}\left((\kappa_y^2 + \kappa_y n)L\varepsilon^{-2}\right)$ for $n < \kappa_y^2$. [ZYG$^+$21] propose Catalyst-SVRG/SAGA and obtain the upper bound $\widetilde{\mathcal{O}}\left((n + n^{3/4}\sqrt{\kappa_y})\Delta L\varepsilon^{-2}\right)$. When $n \leq \kappa^4$, the upper bound of [ZYG$^+$21] is better; otherwise, the upper bound of [LYHZ20] is better. Since we focus on the ill-conditioned setting, the upper and lower bounds nearly match in terms of κ_y. And there is still a $n^{1/4}$ gap in terms of n.

[6] A concurrent work by [ZYG$^+$21] obtains a similar lower bound.

5.2 Construction for the Strongly-Convex-Strongly-Concave Case

In this subsection, we give the exact forms of the hard instance when the objective function is strongly-convex in $\mathbf{x}$ and strongly-concave in $\mathbf{y}$. We still assume $\mu_x \leq \mu_y$. Then we have $\kappa_y \leq \kappa_x$. This means that the max part has a smaller condition number and is easier to solve. According to the magnitude of κ_x and κ_y, the construction can be divided into three cases.

Case 1: $\kappa_x, \kappa_y = \Omega(\sqrt{n})$**.** When both condition numbers are no smaller than $\Theta(\sqrt{n})$, the analysis depends on the following construction.

Definition 12. *For fixed $L, \mu_x, \mu_y, R_x, R_y$ and n such that $\mu_x \leq \mu_y$, $\kappa_x \geq \kappa_y \geq 2$ we define $f_{\mathrm{SCSC},i} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ as follows*

$$f_{\mathrm{SCSC},i}(\mathbf{x}, \mathbf{y}) = \lambda\, r_i^{\mathrm{CC}}\left(\mathbf{x}/\beta, \mathbf{y}/\beta; m, \sqrt{\frac{2}{\alpha+1}}, \mathbf{c}^{\mathrm{SCSC}}\right), \text{ for } 1 \leq i \leq n$$

where

$$\alpha = \sqrt{\frac{(\kappa_y - 2/\kappa_y)\,\kappa_x}{2n} + 1}, \quad \mathbf{c}^{\mathrm{SCSC}} = \left(\frac{2\kappa_y}{\kappa_x}\sqrt{\frac{2n}{\kappa_y^2 - 2}}, 2\sqrt{\frac{2n}{\kappa_y^2 - 2}}\right),$$

$$\beta = \min\left\{2R_x\sqrt{\frac{2\alpha n}{\kappa_x^2(1 - 2/\kappa_y^2)}}, \frac{4R_x}{\alpha+1}\sqrt{\frac{\alpha n}{\kappa_x^2(1 - 2/\kappa_y^2)}}, \frac{\sqrt{2\alpha}R_y}{\alpha - 1}\right\} \text{ and } \lambda = \frac{\beta^2}{2}\sqrt{\frac{L^2 - 2\mu_y^2}{2n}}$$

Consider the minimax problem

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} f_{\mathrm{SCSC}}(\mathbf{x}, \mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_{\mathrm{SCSC},i}(\mathbf{x}, \mathbf{y}) \tag{10}$$

where $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^m : \|\mathbf{x}\|_2 \leq R_x\}$ and $\mathcal{Y} = \{\mathbf{y} \in \mathbb{R}^m : \|\mathbf{y}\|_2 \leq R_y\}$. Define $\phi_{\mathrm{SCSC}}(\mathbf{x}) = \max_{\mathbf{y}\in\mathcal{Y}} f_{\mathrm{SCSC}}(\mathbf{x}, \mathbf{y})$ and $\psi_{\mathrm{SCSC}}(\mathbf{y}) = \min_{\mathbf{x}\in\mathcal{X}} f_{\mathrm{SCSC}}(\mathbf{x}, \mathbf{y})$.

One can check that f_{SCSC} belongs to $\mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, \mu_x, \mu_y)$ and satisfies a condition of the form (7) (please see Proposition 4 in Appendix D.1). Then we can prove the lower bound of the complexity for finding ε-suboptimal point of Problem 10) by PIFO algorithms.

Theorem 5. *Consider the minimax problem 10) and $\varepsilon > 0$. Let $\alpha = \sqrt{\frac{(\kappa_y - 2/\kappa_y)\kappa_x}{2n} + 1}$. Suppose that*

$$n \geq 2,\ \kappa_x \geq \kappa_y \geq \sqrt{2n+2},\ \varepsilon \leq \frac{1}{800}\min\left\{\frac{n\mu_x R_x^2}{\kappa_x\kappa_y}, \mu_y R_y^2\right\},$$

$$\text{and } m = \left\lfloor \frac{\alpha}{4}\log\left(\frac{\max\left\{\mu_x R_x^2, \mu_y R_y^2\right\}}{9\varepsilon}\right)\right\rfloor + 1$$

In order to find $(\widehat{\mathbf{x}}, \widehat{\mathbf{y}}) \in \mathcal{X} \times \mathcal{Y}$ such that $\mathbb{E}\phi_{\mathrm{SCSC}}(\widehat{\mathbf{x}}) - \mathbb{E}\psi_{\mathrm{SCSC}}(\widehat{\mathbf{y}}) < \varepsilon$, PIFO algorithm $\mathcal{A}$ needs at least N queries, where

$$N = \Omega\left(\left(n + \sqrt{n\kappa_x\kappa_y}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$$

The proof of Theorem 5 is deferred to Appendix D.1.

Case 2: $\kappa_x = \Omega(\sqrt{n})$, $\kappa_y = \mathcal{O}(\sqrt{n})$. When only κ_y is no smaller than $\Theta(\sqrt{n})$, the lower bound is characterized by the following theorem.

Theorem 6. *For any $L, \mu_x, \mu_y, n, R_x, R_y, \varepsilon$ such that $n \geq 4$,*

$$n \geq 4,\ \kappa_x \geq \sqrt{2n+2} \geq \kappa_y \geq 2,\ \varepsilon \leq \frac{1}{720}\mu_x R_x^2,\ \widetilde{L} = \sqrt{n(L^2-\mu_x^2)/2 - \mu_x^2},$$
$$\text{and } m = \left\lfloor \frac{1}{4}\left(\sqrt{\frac{2(\widetilde{L}/\mu_x - 1)}{n}} + 1\right)\log\left(\frac{\mu_x R_x^2}{9\varepsilon}\right)\right\rfloor + 1$$

there exist n functions $\{f_i : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}\}_{i=1}^n$ such that $f = \frac{1}{n}\sum_{i=1}^n f_i \in \mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, \mu_x, \mu_y)$. Let $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^m : \|\mathbf{x}\|_2 \leq R_x\}$ and $\mathcal{Y} = \{\mathbf{y} \in \mathbb{R}^m : \|\mathbf{y}\|_2 \leq R_y\}$. In order to find $(\widehat{\mathbf{x}}, \widehat{\mathbf{y}}) \in \mathcal{X} \times \mathcal{Y}$ such that $\mathbb{E}\max_{\mathbf{y}\in\mathcal{Y}} f(\widehat{\mathbf{x}}, \mathbf{y}) - \mathbb{E}\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}, \widehat{\mathbf{y}}) < \varepsilon$, PIFO algorithm $\mathcal{A}$ needs at least N queries, where $N = \Omega\left(\left(n + n^{3/4}\sqrt{\kappa_x}\right)\log\left(\frac{1}{\varepsilon}\right)\right)$.

We find that κ_y does not appear in the lower bound. In fact, since κ_y is relatively small, the max part is easier to solve than the min part and the min part becomes the main obstacle. To construct the hard instance, it suffices to consider the separable function of the form $f(\mathbf{x}, \mathbf{y}) = f_x(\mathbf{x}) - f_y(\mathbf{y})$ where f_x is the hard instance used for finite-sum minimization problems and $f_y(\mathbf{y}) = \frac{\mu_y}{2}\|\mathbf{y}\|_2^2$. For the details, see Appendix D.1.

Case 3: $\kappa_x, \kappa_y = \mathcal{O}(\sqrt{n})$. When both the condition numbers are relatively small, the lower bound is $\Omega(n)$, which means that the number of component functions becomes the main obstacle.

Lemma 6. *For any $L, \mu_x, \mu_y, n, R_x, R_y, \varepsilon$ such that $n \geq 2$, $L \geq \mu_x$, $L \geq \mu_y$ and $\varepsilon \leq \frac{1}{4}LR_x^2$, there exist n functions $\{f_i : \mathbb{R} \times \mathbb{R} \to \mathbb{R}\}_{i=1}^n$ such that $f = \frac{1}{n}\sum_{i=1}^n f_i \in \mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, \mu_x, \mu_y)$. Let $\mathcal{X} = \{x \in \mathbb{R} : |x| \leq R_x\}$ and $\mathcal{Y} = \{y \in \mathbb{R} : |y| \leq R_y\}$. In order to find $(\widehat{x}, \widehat{y}) \in \mathcal{X} \times \mathcal{Y}$ such that $\mathbb{E}\max_{y\in\mathcal{Y}} f(\widehat{x}, y) - \mathbb{E}\min_{x\in\mathcal{X}} f(x, \widehat{y}) < \varepsilon$, PIFO algorithm $\mathcal{A}$ needs at least $N = \Omega(n)$ queries.*

This bound is trivial in some sense, since we usually need to compute the full gradient at least once, whose complexity is of the order $\Omega(n)$. The proof is also deferred to Appendix D.1. Combining Theorems 5, 6 and Lemma 6, we can obtain Theorem 1.

5.3 Construction for the Convex-Strongly-Concave Case

In this subsection, we construct the hard instance when f is convex in $\mathbf{x}$ and strongly-concave in $\mathbf{y}$. The condition number κ_y is still well-defined. Our analysis is based on the following functions.

Definition 13. *For fixed L, μ_y, n, R_x, R_y such that $\kappa_y \geq 2$, we define $f_{\mathrm{CSC},i} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ as follows*

$$f_{\mathrm{CSC},i}(\mathbf{x}, \mathbf{y}) = \lambda\, r_i^{\mathrm{CC}}\left(\mathbf{x}/\beta, \mathbf{y}/\beta; m, 1, \mathbf{c}^{\mathrm{CSC}}\right)$$

where

$$\mathbf{c}^{\mathrm{CSC}} = \left(0, 2\sqrt{\frac{2n}{\kappa_y^2 - 2}}\right),\ \beta = \min\left\{\frac{R_x\sqrt{\frac{\kappa_y^2-2}{2n}}}{2(m+1)^{3/2}}, \frac{R_y}{\sqrt{m}}\right\}\ \text{and}\ \lambda = \frac{\beta^2}{2}\sqrt{\frac{L^2 - 2\mu_y^2}{2n}}$$

Consider the minimax problem

$$\min_{\mathbf{x}\in\mathcal{X}}\max_{\mathbf{y}\in\mathcal{Y}} f_{\mathrm{CSC}}(\mathbf{x},\mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_{\mathrm{CSC},i}(\mathbf{x},\mathbf{y}) \tag{11}$$

where $\mathcal{X} = \{\mathbf{x}\in\mathbb{R}^m : \|\mathbf{x}\|_2 \le R_x\}$ *and* $\mathcal{Y} = \{\mathbf{y}\in\mathbb{R}^m : \|\mathbf{y}\|_2 \le R_y\}$. *Define* $\phi_{\mathrm{CSC}}(\mathbf{x}) = \max_{\mathbf{y}\in\mathcal{Y}} f_{\mathrm{CSC}}(\mathbf{x},\mathbf{y})$ *and* $\psi_{\mathrm{CSC}}(\mathbf{y}) = \min_{\mathbf{x}\in\mathcal{X}} f_{\mathrm{CSC}}(\mathbf{x},\mathbf{y})$.

One can check that f_{CSC} belongs to $\mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, 0, \mu_y)$ and satisfies a condition of the form (7) (please see Proposition 5 in Appendix D.2). Then we can prove the lower bound of the complexity for finding ε-suboptimal point of Problem (11) by PIFO algorithms.

Theorem 7. *Consider the minimax problem (11) and* $\varepsilon > 0$. *Suppose that*

$$n \ge 2,\ \kappa_y \ge 2,\ \varepsilon \le \min\left\{\frac{L^2R_x^2}{5184\,n\mu_y}, \frac{\mu_y R_y^2}{36}\right\} \ \text{ and } \ m = \left\lfloor \frac{R_x}{6}\sqrt{\frac{L^2-2\mu_y^2}{2n\mu_y\varepsilon}}\right\rfloor - 2$$

In order to find $(\widehat{\mathbf{x}},\widehat{\mathbf{y}}) \in \mathcal{X}\times\mathcal{Y}$ *such that* $\mathbb{E}\phi_{\mathrm{CSC}}(\widehat{\mathbf{x}}) - \mathbb{E}\psi_{\mathrm{CSC}}(\widehat{\mathbf{y}}) < \varepsilon$, *PIFO algorithm* $\mathcal{A}$ *needs at least* N *queries, where* $N = \Omega\left(n + R_x\sqrt{nL\kappa_y/\varepsilon}\right)$.

When κ_y is small, the second term of N is also small. In fact, when $\kappa_y = \mathcal{O}(\sqrt{n})$, we can provide a better lower bound as follows.

Theorem 8. *For any* $L, \mu_y, n, R_x, R_y, \varepsilon$ *such that* $n \ge 2$, $L \ge \mu_y$, $\varepsilon \le \frac{\sqrt{2}R_x^2 L}{768\sqrt{n}}$ *and* $m = \left\lfloor \frac{\sqrt[4]{18}}{12} R_x n^{-1/4}\sqrt{\frac{L}{\varepsilon}}\right\rfloor - 1$, *there exist* n *functions* $\{f_i : \mathbb{R}^m\times\mathbb{R}^m\to\mathbb{R}\}_{i=1}^n$ *such that* $f = \frac{1}{n}\sum_{i=1}^n f_i \in \mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, 0, \mu_y)$ *Let* $\mathcal{X} = \{\mathbf{x}\in\mathbb{R}^m : \|\mathbf{x}\|_2 \le R_x\}$ *and* $\mathcal{Y} = \{\mathbf{y}\in\mathbb{R}^m : \|\mathbf{y}\|_2 \le R_y\}$. *In order to find* $(\widehat{\mathbf{x}},\widehat{\mathbf{y}}) \in \mathcal{X}\times\mathcal{Y}$ *such that* $\mathbb{E}\max_{\mathbf{y}\in\mathcal{Y}} f(\widehat{\mathbf{x}},\mathbf{y}) - \mathbb{E}\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x},\widehat{\mathbf{y}}) < \varepsilon$, *PIFO algorithm* $\mathcal{A}$ *needs at least* $N = \Omega\left(n + R_x n^{3/4}\sqrt{L/\varepsilon}\right)$ *queries.*

The construction of Theorem 8 is similar to that of Theorem 6. We still consider the separable function $f(\mathbf{x},\mathbf{y}) = f_x(\mathbf{x}) - f_y(\mathbf{y})$ where f_x is the hard instance used for finite-sum minimization problems and $f_y(\mathbf{y}) = \frac{\mu_y}{2}\|\mathbf{y}\|_2^2$. The proofs of Theorems 7 and 8 are deferred to Appendix D.2.

Now we give the proof of Theorem 2.

Proof of Theorem 2. By Lemma 6, we have the lower bound $\Omega(n)$ if $\varepsilon \le LR_x^2/4$. Note that if $\varepsilon \ge \frac{L^2R_x^2}{5184n\mu_y}$, $\Omega(n) = \Omega\left(n + R_x\sqrt{\frac{nL\kappa_y}{\varepsilon}}\right)$. And if $\varepsilon \ge \frac{\sqrt{2}R_x^2L}{768\sqrt{n}}$, $\Omega(n) = \Omega\left(n + R_x n^{3/4}\sqrt{\frac{L}{\varepsilon}}\right)$. Then for $\varepsilon \le \min\left\{\frac{LR_x^2}{4}, \frac{\mu_y R_y^2}{36}\right\}$, we have $\mathfrak{m}^{\mathrm{CC}}(\varepsilon, R_x, R_y, L, 0, \mu_y) = \Omega\left(n + R_x\sqrt{\frac{nL}{\varepsilon}} + \frac{R_xL}{\sqrt{\mu_y\varepsilon}}\right)$. It remains to add the term $\Omega(n^{3/4}\sqrt{\kappa_y}\log(\frac{1}{\varepsilon}))$ for $\kappa_y = \Omega(\sqrt{n})$.

Now we construct $\{H_{\mathrm{CSC},i}\}_{i=1}^n, H_{\mathrm{CSC}} : \mathbb{R}^m\times\mathbb{R}^m\to\mathbb{R}$ as follows.

$$H_{\mathrm{CSC},i}(\mathbf{x},\mathbf{y}) = \frac{L}{2}\|\mathbf{x}\|_2^2 - g_{\mathrm{SC},i}(\mathbf{y}),$$
$$H_{\mathrm{CSC}}(\mathbf{x},\mathbf{y}) = \frac{1}{n}\sum_{i=1}^n H_{\mathrm{CSC},i}(\mathbf{x},\mathbf{y}) = \frac{L}{2}\|\mathbf{x}\|_2^2 - g_{\mathrm{SC}}(\mathbf{y})$$

where $g_{\mathrm{SC}}(\mathbf{y})$ is μ_y-convex and $\{g_{\mathrm{SC},i}(\mathbf{y})\}_{i=1}^n$ is L-average smooth. It is easy to check $H_{\mathrm{CSC}} \in \mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, 0, \mu_y)$,

$$\min_{\mathbf{x}\in\mathcal{X}} H_{\mathrm{CSC}}(\mathbf{x},\mathbf{y}) = -g_{\mathrm{SC}}(\mathbf{y}) \quad \text{and} \quad \max_{\mathbf{y}\in\mathcal{Y}} H_{\mathrm{CSC}}(\mathbf{x},\mathbf{y}) = \frac{1}{2}\|\mathbf{x}\|_2^2 - \min_{\mathbf{y}\in\mathcal{Y}} g_{\mathrm{SC}}(\mathbf{y})$$

It follows that for any $(\widehat{\mathbf{x}}, \widehat{\mathbf{y}}) \in \mathcal{X} \times \mathcal{Y}$, we have

$$\max_{\mathbf{y}\in\mathcal{Y}} H_{\mathrm{CSC}}(\widehat{\mathbf{x}},\mathbf{y}) - \min_{\mathbf{x}\in\mathcal{X}} H_{\mathrm{CSC}}(\mathbf{x},\widehat{\mathbf{y}}) \geq g_{\mathrm{SC}}(\widehat{\mathbf{y}}) - \min_{\mathbf{y}\in\mathcal{Y}} g_{\mathrm{SC}}(\mathbf{y})$$

By Theorem 17, for $\varepsilon \leq LR_y^2/4$ and $\kappa_y = \Omega(\sqrt{n})$, we have $\mathfrak{m}_\varepsilon^{\mathrm{CC}}(R_x, R_y, L, 0, \mu_y) = n^{3/4}\sqrt{\kappa_y}\log\left(\frac{1}{\varepsilon}\right)$. This completes the proof. □

5.4 Construction for the Convex-Concave Case

For the general convex-concave case, the hard instance is constructed as follows.

Definition 14. *For fixed L, n, R_x, R_y such that $n \geq 2$, we define $f_{\mathrm{CC},i} : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}$ as follows*

$$f_{\mathrm{CC},i}(\mathbf{x},\mathbf{y}) = \lambda\, r_i^{\mathrm{CC}}\left(\mathbf{x}/\beta, \mathbf{y}/\beta; m, 1, \mathbf{0}\right)$$

where $\lambda = \frac{LR_y^2}{m\sqrt{8n}}$ and $\beta = \frac{R_y}{\sqrt{m}}$. Consider the minimax problem

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} f_{\mathrm{CC}}(\mathbf{x},\mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^n f_{\mathrm{CC},i}(\mathbf{x},\mathbf{y}) \tag{12}$$

where $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^m : \|\mathbf{x}\|_2 \leq R_x\}$ and $\mathcal{Y} = \{\mathbf{y} \in \mathbb{R}^m : \|\mathbf{y}\|_2 \leq R_y\}$. Define $\phi_{\mathrm{CC}}(\mathbf{x}) = \max_{\mathbf{y}\in\mathcal{Y}} f_{\mathrm{CC}}(\mathbf{x},\mathbf{y})$ and $\psi_{\mathrm{CC}}(\mathbf{y}) = \min_{\mathbf{x}\in\mathcal{X}} f_{\mathrm{CC}}(\mathbf{x},\mathbf{y})$.

One can check that f_{CC} belongs to $\mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, 0, 0)$ and satisfies a condition of the form (7) (please see Proposition 6 in Appendix D.3). Then, we can obtain a PIFO lower bound complexity for the general finite-sum convex-concave minimax problem.

Theorem 9. *Consider minimax problem (12) and $\varepsilon > 0$. Suppose that*

$$n \geq 2,\ \varepsilon \leq \frac{LR_xR_y}{72\sqrt{n}},\ \text{and}\ \ m = \left\lfloor \frac{LR_xR_y}{18\varepsilon\sqrt{n}} \right\rfloor - 1$$

In order to find $(\widehat{\mathbf{x}}, \widehat{\mathbf{y}}) \in \mathcal{X} \times \mathcal{Y}$ such that $\mathbb{E}\phi_{\mathrm{CC}}(\widehat{\mathbf{x}}) - \mathbb{E}\psi_{\mathrm{CC}}(\widehat{\mathbf{y}}) < \varepsilon$, PIFO algorithm $\mathcal{A}$ needs at least $N = \Omega\left(n + \sqrt{n}LR_xR_y/\varepsilon\right)$ queries.

Note that Theorem 7 requires the condition $\varepsilon \leq \mathcal{O}(L/\sqrt{n})$ to obtain the desired lower bound. For large ε, we can apply the following lemma.

Lemma 7. *For any positive $L, n, R_x, R_y, \varepsilon$ such that $n \geq 2$ and $\varepsilon \leq \frac{1}{4}LR_xR_y$ there exist n functions $\{f_i : \mathbb{R}\times\mathbb{R} \to \mathbb{R}\}_{i=1}^n$ such that $f = \frac{1}{n}\sum_{i=1}^n f_i \in \mathcal{F}_{\mathrm{CC}}(R_x, R_y, L, 0, 0)$. Let $\mathcal{X} = \{x \in \mathbb{R} : |x| \leq R_x\}$ and $\mathcal{Y} = \{y \in \mathbb{R} : |y| \leq R_y\}$. In order to find $(\widehat{x}, \widehat{y}) \in \mathcal{X} \times \mathcal{Y}$ such that $\mathbb{E}\max_{y\in\mathcal{Y}} f(\widehat{x}, y) - \mathbb{E}\min_{x\in\mathcal{X}} f(x, \widehat{y}) < \varepsilon$, PIFO algorithm $\mathcal{A}$ needs at least $N = \Omega(n)$ queries.*

This Lemma is similar to Lemma 6. The proofs of Theorem 9 and Lemma 7 are deferred to Appendix D.3.

Now we can give the proof of Theorem 3.

Proof of Theorem 3. Note that for $\varepsilon \geq \frac{LR_xR_y}{72\sqrt{n}}$, we have $\Omega\left(n + \frac{\sqrt{n}LR_xR_y}{\varepsilon}\right) = \Omega(n)$. Combining Theorem 9 and Lemma 8, we obtain the lower bound $\Omega\left(n + \frac{\sqrt{n}LR_xR_y}{\varepsilon}\right)$ for $\varepsilon \leq LR_xR_y/4$. On the other hand, G_{CSC} defined in the proof of Theorem 8 and H_{SCSC} defined in the proof of Lemma 6 are also convex-concave and $\varepsilon \geq \frac{\sqrt{2}R_x^2L}{768\sqrt{n}}$ implies $\Omega\left(n + R_xn^{3/4}\sqrt{\frac{L}{\varepsilon}}\right) = \Omega(n)$. Thus, we have the lower bound $\Omega\left(n + R_xn^{3/4}\sqrt{\frac{L}{\varepsilon}}\right)$ for $\varepsilon \leq LR_x^2/4$. It is also worth noting that if $f(\mathbf{x}, \mathbf{y})$ is convex in $\mathbf{x}$ and concave in $\mathbf{y}$, then $-f(\mathbf{x}, \mathbf{y})$ is convex in $\mathbf{y}$ and concave in $\mathbf{x}$. This implies the symmetry of $\mathbf{x}$ and $\mathbf{y}$. Thus, we can also obtain the lower bound $\Omega\left(n + R_yn^{3/4}\sqrt{\frac{L}{\varepsilon}}\right)$ for $\varepsilon \leq LR_y^2/4$. In summary, for $\varepsilon \leq \frac{LR_xR_y}{4}$, the lower bound is $\Omega\left(n + \frac{LR_xR_y}{\varepsilon} + (R_x + R_y)n^{3/4}\sqrt{\frac{L}{\varepsilon}}\right)$. □

5.5 Construction for the Nonconvex-Strongly-Concave Case

In this subsection, we consider the finite-sum minimax problem where the objective function is strongly-concave in $\mathbf{y}$ but nonconvex in $\mathbf{x}$. The analysis is based on the following construction.

Definition 15. *For fixed $L, \mu_x, \mu_y, \Delta, n$, we define $f_{\mathrm{NCSC},i} : \mathbb{R}^{m+1} \times \mathbb{R}^{m+1} \to \mathbb{R}$ as follows*

$$f_{\mathrm{NCSC},i}(\mathbf{x}, \mathbf{y}) = \lambda\, r_i^{\mathrm{NCC}}\left(\mathbf{x}/\beta, \mathbf{y}/\beta; m+1, \sqrt[4]{\alpha}, \mathbf{c}^{\mathrm{NCSC}}\right), \text{ for } 1 \leq i \leq n$$

where

$$\alpha = \min\left\{1, \frac{32n\mu_y}{135L}, \frac{128(\sqrt{3}+1)n\mu_x\mu_y}{45L^2}\right\},\ \mathbf{c}^{\mathrm{NCSC}} = \left(\frac{16\sqrt{n}\mu_y}{L}, \frac{\sqrt{\alpha}L}{16\sqrt{n}\mu_y}, \sqrt[4]{\alpha}\right),$$
$$\lambda = \frac{5308416n^{3/2}\mu_y^2\varepsilon^2}{L^3\alpha},\ \beta = 4\sqrt{\lambda\sqrt{n}/L} \ \text{ and } \ m = \left\lfloor \frac{\Delta L^2\sqrt{\alpha}}{3483648n\varepsilon^2\mu_y} \right\rfloor$$

Define $\phi_{\mathrm{NCSC}}(\mathbf{x}) = \max_{\mathbf{y}\in\mathbb{R}^{m+1}} f_{\mathrm{NCSC}}(\mathbf{x}, \mathbf{y})$. Consider the minimax problem

$$\min_{\mathbf{x}\in\mathbb{R}^{m+1}} \max_{\mathbf{y}\in\mathbb{R}^{m+1}} f_{\mathrm{NCSC}}(\mathbf{x}, \mathbf{y}) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_{\mathrm{NCSC},i}(\mathbf{x}, \mathbf{y}) \tag{13}$$

One can check that f_{NCSC} belongs to $\mathcal{F}_{\mathrm{NCC}}(\Delta, L, \mu_x, \mu_y)$ and satisfies a condition of the form (9) (please see Proposition 7 in Appendix D.4). With Proposition 7, we can give the proof of Theorem 4.

Proof of Theorem 4. Combining Lemma 5 and the third property of Proposition 7, for $N = \frac{nm}{4(1+c_0)}$, we have $\min_{t\leq N} \mathbb{E}\left\|\nabla\phi_{\mathrm{NCSC}}(\mathbf{x}_t)\right\|_2 \geq \varepsilon$. Thus, in order to find $(\widehat{\mathbf{x}}, \widehat{\mathbf{y}})$ such that $\mathbb{E}\left\|\nabla\phi_{\mathrm{NCSC}}(\widehat{\mathbf{x}})\right\|_2 < \varepsilon$, $\mathcal{A}$ needs at least N PIFO queries, where $N = \frac{nm}{4(1+c_0)} = \Omega\left(\frac{\Delta L^2\sqrt{\alpha}}{\varepsilon^2\mu_y}\right)$. Since $\varepsilon^2 \leq \frac{\Delta L^2\alpha}{6767296n\mu_y}$ and $\alpha \leq 1$, we have $\Omega\left(\frac{\Delta L^2\sqrt{\alpha}}{\varepsilon^2\mu_y}\right) = \Omega\left(n + \frac{\Delta L^2\sqrt{\alpha}}{\varepsilon^2\mu_y}\right)$. □

5.6 Smooth Cases

In this subsection, we focus on the more constrained function classes where each component f_i is L-smooth. The results are summarized in Table 2. We defer the definitions of the function class and optimization complexity and the formal statements of our lower bounds to Appendix D.5.

Table 2. Upper and lower bounds with the assumption that f_i is L-smooth and f is (μ_x, μ_y)-convex-concave. The condition numbers are defined as $\kappa_x = L/\mu_x$ and $\kappa_y = L/\mu_y$ when $\mu_x, \mu_y > 0$. The definitions of R_x, R_y and Δ are given in Table 1.

Cases	Upper or Lower Bounds	References
$\mu_x > 0, \mu_y > 0$	$\widetilde{\mathcal{O}}\left(\left(n + \frac{\sqrt{n}L}{\min\{\mu_x,\mu_y\}}\right)\log(1/\varepsilon)\right)$	[CJST19]; [LCL$^+$19]
	$\Omega\left(\sqrt{(n+\kappa_x)(n+\kappa_y)}\log(1/\varepsilon)\right)$	Theorem 10
$\mu_x = 0, \mu_y > 0$	$\Omega\left(n + R_x\sqrt{\frac{nL}{\varepsilon}} + R_x\sqrt{\frac{L\kappa_y}{\varepsilon}} + \sqrt{n\kappa_y}\log\left(\frac{1}{\varepsilon}\right)\right)$	Theorem 11
$\mu_x = 0, \mu_y = 0$	$\widetilde{\mathcal{O}}\left(n + \frac{\sqrt{n}L(R_x^2+R_y^2)}{\varepsilon}\right)$	[CJST19]
	$\Omega\left(n + \frac{LR_xR_y}{\varepsilon} + (R_x+R_y)\sqrt{\frac{nL}{\varepsilon}}\right)$	Theorem 12
$\mu_x < 0, \mu_y > 0$ $\kappa_y = \Omega(\sqrt{n})$	$\Omega\left(n + \frac{\Delta L\sqrt{\kappa_y}}{\varepsilon^2}\right)$	Theorem 13

In Table 2, we only present the upper bounds of some methods designed for the smoothness case[7]. Methods designed for the average smoothness functions also apply here and thus the upper bounds in Table 1 are still valid. However, there exists some gap in all cases.

Compared to the lower bounds in Table 1, the lower bounds in Table 2 have the same dependence on $L, \kappa_x, \kappa_y, \varepsilon$, but with a weaker dependence on n. Specially. if we replace L, κ_x and κ_y in Table 2 by $\sqrt{n}L$, $\sqrt{n}\kappa_x$ and $\sqrt{n}\kappa_y$ respectively[8], we can obtain the lower bounds in Table 1. This is due to the way of partitioning the matrix $\mathbf{B}(m, \omega, \zeta)$ in Section 4.2. Intuitively, we partition the Hessian matrix of the coupling term between $\mathbf{x}$ and $\mathbf{y}$ and each component only gets a low-rank part. Propositions 1 and 2 have shown the $\sqrt{n}$ gap between the smoothness and average smoothness parameters as long as the non-coupling term is not too large.

Convex-concave cases We speculate that when f is convex-concave, the lower bounds in Table 2 are the best ones our framework can obtain, because the corresponding lower bounds under the average smoothness assumption have been nearly matched by existing upper bounds. To further improve the lower bounds, one may have to resort to new constructions.

As for the upper bounds, we notice that most work only uses the average smoothness condition. We guess that the smoothness property of each component function needs to be better employed, because the upper and lower bounds for convex minimization problems under the two smoothness conditions nearly match (see Tables 3 and 4),

Nonconvex-strongly-concave case When f is nonconvex-strongly-concave, there exists a gap between the upper and lower bounds under both smoothness and average smoothness assumptions. Since the nonconvexity poses more difficulty to the problem, it remains an open problem whether

[7] Although the method in [CJST19] has two loops and does not satisfy our definition, we list it here for a better comparison.

[8] For the nonconvex-strongly-concave case, we just need to replace L by $\sqrt{n}L$.

the upper bounds, the lower bounds, or both can be further tightened.

6 Lower Complexity Bounds for the Minimization Problems

In this section, we focus on the minimization problem

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} f_i(\mathbf{x}) \tag{14}$$

where each individual component $f_i(\mathbf{x})$ is L-smooth or the function class $\{f_i(\mathbf{x})\}_{i=1}^n$ is L-average smooth, the feasible set $\mathcal{X}$ is closed and convex such that $\mathcal{X} \subseteq \mathbb{R}^d$. We show that we can obtain similar lower bounds as those in [WS16, HLOY18, ZG19].

Recall that Problem (1) becomes Problem (14) if we set $\mathcal{Y}$ as a singleton. Then the definitions of function classes and optimization complexity come directly from their counterparts in Sections 4.1. The details are deferred to Appendix E.1.

In Section 6.1, we construct the hard instances for Problem (14). In Section 6.2, we summarize our results and compare them with previous work.

6.1 The Hard Instances

In this subsection, we present the construction of hard instances for Problem (14) and compare our construction with some related work.

The construction is also based on the class of matrices $\mathbf{B}(m,\omega,\zeta)$ define in Equation (3). We still use $\mathbf{b}_{l-1}(m,\omega,\zeta)^\top$ to denote the l-th row of $\mathbf{B}(m,\omega,\zeta)$ and defined the index sets $\mathcal{L}_1,\ldots,\mathcal{L}_n$ as $\mathcal{L}_i = \big\{l : 0 \le l \le m, l \equiv i-1 \pmod n\big\}$. Then the hard instance is constructed as

$$\min_{\mathbf{x}\in\mathcal{X}} r(\mathbf{x};m,\omega,\zeta,\mathbf{c}) \triangleq \frac{1}{n}\sum_{i=1}^{n} r_i(\mathbf{x};m,\omega,\zeta,\mathbf{c}) \tag{15}$$

where $\mathbf{c} = (c_1,c_2,c_3)$, $\mathcal{X} = \{\mathbf{x}\in\mathbb{R}^m : \|\mathbf{x}\|_2 \le R_x\}$ or $\mathbb{R}^m$, and

$$r_i(\mathbf{x};m,\omega,\zeta,\mathbf{c}) = \begin{cases} \frac{n}{2}\sum\limits_{l\in\mathcal{L}_i}\left\|\mathbf{b}_l(m,\omega,\zeta)^\top\mathbf{x}\right\|_2^2 + \frac{c_1}{2}\|\mathbf{x}\|_2^2 + c_2\sum\limits_{i=1}^{m-1}\Gamma(x_i) - c_3 n\,\langle\mathbf{e}_1,\mathbf{x}\rangle, & \text{for } i=1,\\ \frac{n}{2}\sum\limits_{l\in\mathcal{L}_i}\left\|\mathbf{b}_l(m,\omega,\zeta)^\top\mathbf{x}\right\|_2^2 + \frac{c_1}{2}\|\mathbf{x}\|_2^2 + c_2\sum\limits_{i=1}^{m-1}\Gamma(x_i), & \text{for } i=2,3,\ldots,n \end{cases}$$

The nonconvex function $\Gamma:\mathbb{R}\to\mathbb{R}$ is $\Gamma(x) \triangleq 120\int_1^x \frac{t^2(t-1)}{1+t^2}dt$. We can determine the smoothness and strong convexity parameters of r_i similar to Propositions 1 and 2. The details are deferred to Proposition 9 in Appendix E.2.

One can check that $r(\mathbf{x};m,\omega,\zeta,\mathbf{c}) = \frac{1}{2}\mathbf{x}^\top\mathbf{A}(m,\omega,\zeta)\,\mathbf{x} + \frac{c_1}{2}\|\mathbf{x}\|_2^2 + c_2\sum_{i=1}^{m-1}\Gamma(\mathbf{x}_i) - c_3\,\langle\mathbf{e}_1,\mathbf{x}\rangle$, where

$$\mathbf{A}(m,\omega,\zeta) \triangleq \mathbf{B}(m,\omega,\zeta)^\top\mathbf{B}(m,\omega,\zeta) = \begin{bmatrix} \omega^2+1 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & \zeta^2+1 \end{bmatrix}$$

The matrix $\mathbf{A}(m,\omega,\zeta)$ is widely-used in the analysis of lower bounds for convex optimization [Nes13, AB15, LZ17, CDHS17a, ZG19].

Now we compare our construction with [LZ17] and [ZG19]. In our construction, we partition the row vectors of $\mathbf{B}(m,\omega,\zeta)$ into n parts and each component function is defined in terms of only one part. All the component functions share the same $\mathbf{x}$. However, in [LZ17], different component functions share the same form except that they are based on different subvectors of the high-dimensional $\mathbf{x}$. Intuitively speaking, we partition the Hessian matrix while [LZ17] partition the variable. The construction of [ZG19] is more complex than [LZ17] but the basic idea is the same.

Recall the subspaces $\{\mathcal{F}_k\}_{k=0}^m$ defined in (5). The next lemma shows that the hard instance also satisfies a variant of the *zero-chain* property.

Lemma 8. *Suppose that $n \geq 2$, $c_1 \geq 0$ and $\mathbf{x} \in \mathcal{F}_k$, $0 \leq k < m$. If (i) (convex case) $c_2 = 0$ and $\omega = 0$, or (ii) (nonconvex case) $c_1 = 0$, $c_2 > 0$, $\zeta = 0$ and $\gamma < \frac{\sqrt{2}+1}{60c_2}$, we have*

$$\nabla r_i(\mathbf{x}),\, \mathrm{prox}_{r_i}^{\gamma}(\mathbf{x}) \in \begin{cases} \mathcal{F}_{k+1}, & \text{if } i \equiv k+1 (\mathrm{mod}\ n), \\ \mathcal{F}_k, & \text{otherwise} \end{cases}$$

We omit the parameters of r_i to simplify the presentation.

The proof of Lemma 8 are given in Appendix E.7.

We emphasize that the assumption on γ naturally holds. Recall that the choice of γ should satisfy that $r_i(\mathbf{u}) + \frac{1}{2\gamma}\|\mathbf{x}-\mathbf{u}\|_2^2$ is a convex function of $\mathbf{u}$ for a fixed $\mathbf{x}$. Proposition 9 implies that we must have $\gamma \leq \frac{1}{45(\sqrt{3}-1)c_2} \leq \frac{\sqrt{2}+1}{60c_2}$.

In short, if $\mathbf{x} \in \mathcal{F}_k$, then there exists only one $i \in \{1,\ldots,n\}$ such that $h_{f_i}^{\mathrm{PIFO}}$ could provide additional information in $\mathcal{F}_{k+1}$. This property is the main difference between the constructions in [LZ17, ZG19] and ours. In [LZ17, ZG19], no matter which component is drawn, the number of the nonzero elements of the current point can increase. Such a difference results from the different ways of partitioning. As a consequence, their hard instances need to be constructed in a space with a higher dimension than ours. Moreover, our construction also works for PIFO oracles while the constructions of [LZ17] and [ZG19] only apply to IFO oracles.

With Lemma 8, we can obtain how many PIFO calls we need as what we did in Section 4.2. The details are deferred to Appendix E.2.

6.2 Results

In this subsection, we present our lower bounds in Tables 3 and 4, and compare them with previous upper and lower bounds. It is worth emphasizing that we are not trying to list all the upper bounds, just to provide a few algorithms that could match our lower bounds. The formal statements of our lower bounds are deferred to Appendix E.3.

Smooth cases Table 3 shows the upper and lower bounds when each f_i is L-smooth[9]. For the strongly-convex and convex cases, the upper bounds and lower bounds nearly match up to log factors, while for the nonconvex case, there is still a $\sqrt{n}$ gap. Specially, when $\kappa = \Omega(n)$, the lower bound is $\Omega(n + \Delta\sqrt{n|\mu|L}/\varepsilon^2)$ and has been achieved by [LZ17] up to log factors. When $\kappa = \mathcal{O}(n)$, the lower bound is $\Omega(n + \Delta/\varepsilon^2)$, while the upper bound by [LMG20] is $\Omega(n + \sqrt{n}\Delta/\varepsilon^2)$. From the

[9]The lower bound of [HLOY18] for $\kappa = \Omega(n)$ uses the lower bound in [WS16].

Table 3. The upper and lower bounds with the assumption that f_i is L-smooth and f is μ-strongly convex, convex or μ-weakly convex. $\kappa = L/\mu$ for $\mu > 0$. The definitions of R, Δ and optimization complexity are given in Appendix E.1.

Cases	Upper or Lower Bounds	References
$\mu > 0$	$\mathcal{O}\left((n+\sqrt{\kappa n})\log(1/\varepsilon)\right)$	[Def16, Li21]
	$\mathcal{O}\left(n + \frac{n\log(1/\varepsilon)}{1+(\log(n/\kappa))_+}\right), \quad \kappa = \mathcal{O}(n)$	[HLOY18]
	$\begin{cases} \Omega\left((n+\sqrt{\kappa n})\log(1/\varepsilon)\right), & \kappa = \Omega(n), \\ \Omega\left(n + \frac{n\log(1/\varepsilon)}{1+(\log(n/\kappa))_+}\right), & \kappa = \mathcal{O}(n) \end{cases}$	[HLOY18]; Theorem 14
$\mu = 0$	$\widetilde{\mathcal{O}}\left(n + R\sqrt{nL/\varepsilon}\right)$	[Li21]
	$\Omega\left(n+R\sqrt{nL/\varepsilon}\right)$	[WS16]; Theorem 15
$\mu < 0$	$\widetilde{\mathcal{O}}\left(n + \frac{\Delta}{\varepsilon^2}\min\{\sqrt{n}L, n\lvert\mu\rvert + \sqrt{n\lvert\mu\rvert L}\}\right)$	[LY19, LMG20]
	$\Omega\left(n + \frac{\Delta}{\varepsilon^2}\min\{L, \sqrt{n\lvert\mu\rvert L}\}\right)$	[ZG19]; Theorem 16

analysis in Section 3.2, the algorithms in [Def16, HLOY18, Li21, LY19, LMG20] all belong to PIFO algorithms. In fact, except the one in [Def16], others are also IFO algorithms.

As for the lower bounds, [HLOY18] consider the class of p-CLI oblivious algorithms introduced in [AS16]. For these algorithms, we can left-multiply the gradient by a preconditioning matrix. Thus, the linear-span assumption can be violated. However, proximal operators are still not taken into account. [WS16] prove the lower bounds for arbitrary randomized algorithms with access to PIFO oracles. Although smaller than that in [WS16], our class of algorithms is large enough to include many near-optimal algorithms. Moreover, our construction is simpler than [WS16]. As a result, such a construction can not only provide more intuition about the optimization process, but also requires fewer dimensions to construct the hard instances. Specially, for the convex case, our construction only requires the dimension to be $\mathcal{O}\left(1 + R\sqrt{L/(n\varepsilon)}\right)$ (see Appendix E.5), which is much smaller than $\mathcal{O}\left(\frac{L^2R^4}{\varepsilon^2}\log\left(\frac{nLR^2}{\varepsilon}\right)\right)$ in [WS16].

[ZG19] only consider the class of IFO algorithms, which is only a subset of PIFO algorithms. Moreover, our construction still requires fewer dimensions. For the nonconvex case, our construction only requires the dimension to be $\mathcal{O}\left(1 + \frac{\Delta}{\varepsilon^2}\min\{L/n, \sqrt{\mu L/n}\}\right)$ (see Appendix E.6), which is much smaller than $\mathcal{O}\left(\frac{\Delta}{\varepsilon^2}\min\{L, \sqrt{n\mu L}\}\right)$ in [ZG19].

Average smooth cases For the average smooth cases, the upper and lower bounds nearly match up to log factors for all three cases. Specially, for the nonconvex case, when $\kappa = \Omega(\sqrt{n})$, the lower bound is $\Omega(n + \Delta n^{3/4}\sqrt{|\mu|L}/\varepsilon^2)$ and has been achieved by repeatedSVRG in [AAZB$^+$17, CDHS18, AZ17b][10] up to log factors. When $\kappa = \mathcal{O}(\sqrt{n})$, the lower bound is $\Omega(n + \Delta L\sqrt{n}/\varepsilon^2)$ and has been achieved by [LBZR21]. One can check that the algorithms in [AZ18, LBZR21] are both IFO

[10] This method was implicitly proposed in [AAZB$^+$17, CDHS18] and formally named as repeatedSVRG in [AZ17b].

Table 4. The upper and lower bounds with the assumption that $\{f_i\}_{i=1}^n$ is L-average smooth and f is μ-strongly convex, convex or μ-weakly convex. $\kappa = L/\mu$ for $\mu > 0$. The definitions of R, Δ and optimization complexity are given in Appendix E.1.

Cases	Upper or Lower Bounds	References
$\mu > 0$, $\kappa = \Omega(\sqrt{n})$	$\mathcal{O}\left(\left(n+n^{3/4}\sqrt{\kappa}\right)\log\left(1/\varepsilon\right)\right)$	[AZ18]
	$\Omega\left(\left(n+n^{3/4}\sqrt{\kappa}\right)\log\left(1/\varepsilon\right)\right)$	[ZG19]; Theorem 17
$\mu = 0$	$\mathcal{O}\left(n\log(1/\varepsilon) + Rn^{3/4}\sqrt{L/\varepsilon}\right)$	[AZ18]
	$\Omega\left(n+Rn^{3/4}\sqrt{L/\varepsilon}\right)$	[ZG19]; Theorem 18
$\mu < 0$	$\widetilde{\mathcal{O}}\left(n + \frac{\Delta}{\varepsilon^2}\min\{\sqrt{n}L, n^{3/4}\sqrt{\lvert\mu\rvert L}\}\right)$	[AZ17b, LBZR21]
	$\Omega\left(n + \frac{\Delta}{\varepsilon^2}\min\{\sqrt{n}L, n^{3/4}\sqrt{\lvert\mu\rvert L}\}\right)$	[ZG19]; Theorem 19

algorithms. The method repeatedSVRG in [AZ17b] can also be modified into IFO algorithms[11]. As for the lower bounds, our results have the same orders as those in [ZG19] and can apply to PIFO algorithms. And our constructions also require fewer dimensions than [ZG19]. The details are deferred to Appendix E.3.

IFO and PIFO algorithms From the above analysis, we find that PIFO oracles are no more powerful than IFO oracles in terms of the complexity for smooth functions. The PIFO lower bounds have been nearly matched by many IFO algorithms. This is consistent with the observation in [WS16]. From the results in Table 1, this phenomenon also appears in finite-sum minimax problems under the average smoothness assumption. As a comparison, [WS16] shows that for Lipschitz but nonsmooth functions, having access to proximal oracles does reduce the complexity.

7 Conclusion

In this paper, we explored the lower complexity bounds for finite-sum optimization problems, where the objective is the average of n individual component functions. We introduced Proximal Incremental First-Order (PIFO) algorithms, which access both gradient and proximal oracles for each component function, and allow for infrequent full gradient updates, thus accommodating loopless methods.

A key contribution of this work is the novel construction of hard instances, where we partition the classical tridiagonal matrix into n groups. This new framework facilitates the analysis of both IFO and PIFO algorithms, offering deeper insights into the optimization process while reducing the dimensionality required in comparison to prior approaches.

We derived new lower complexity bounds for finite-sum minimax problems when the objective function is convex-concave or nonconvex-strongly-concave, with L-average smoothness of the component functions. Most of these bounds are nearly matched by existing upper bounds up to

[11] Similar to the analysis for catalyst accelerated methods in Section 3.2.

logarithmic factors. For finite-sum minimization problems, we obtained similar lower bounds to prior work under both smoothness and average smoothness assumptions. Importantly, our analysis reveals that proximal oracles do not significantly outperform gradient oracles for smooth functions.

Looking forward, several directions remain open for further research:

- There are gaps between the upper and lower bounds for nonconvex-strongly-concave and L-smooth cases, suggesting opportunities for designing faster algorithms or tightening the lower bounds.
- Extending our construction framework to handle nonconvex-concave problems would provide valuable insights into more general optimization settings.
- The definition of PIFO algorithms could be broadened to include more complex sampling strategies, such as non-stationary distributions or sampling without replacement, and to explore methods that break the linear-span protocol.

References

[AAZB+17] Naman Agarwal, Zeyuan Allen-Zhu, Brian Bullins, Elad Hazan, and Tengyu Ma. Finding approximate local minima faster than gradient descent. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1195–1199, 2017.

[AB15] Alekh Agarwal and Leon Bottou. A lower bound for the optimization of finite sums. In *ICML*, 2015.

[AM22] Ahmet Alacaoglu and Yura Malitsky. Stochastic variance reduction for variational inequality methods. In *Conference on Learning Theory*, pages 778–816. PMLR, 2022.

[AS16] Yossi Arjevani and Ohad Shamir. Dimension-free iteration complexity of finite sum optimization problems. *Advances in Neural Information Processing Systems*, 29, 2016.

[AZ17a] Zeyuan Allen-Zhu. Katyusha: The first direct acceleration of stochastic gradient methods. *Journal of Machine Learning Research*, 18(1):8194–8244, 2017.

[AZ17b] Zeyuan Allen-Zhu. Natasha: Faster non-convex stochastic optimization via strongly non-convex parameter. In *ICML*, 2017.

[AZ18] Zeyuan Allen-Zhu. Katyusha X: Practical momentum method for stochastic sum-of-nonconvex optimization. In *ICML*, 2018.

[BGBL22] Aleksandr Beznosikov, Eduard Gorbunov, Hugo Berard, and Nicolas Loizou. Stochastic gradient descent-ascent: Unified theory and new efficient methods. *arXiv preprint arXiv:2202.07262*, 2022.

[BTEGN09] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization*, volume 28. Princeton University Press, 2009.

[CDHS17a] Yair Carmon, John C. Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points I. *arXiv preprint:1710.11606*, 2017.

[CDHS17b] Yair Carmon, John C. Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points II: first-order methods. *arXiv preprint:1711.00841*, 2017.

[CDHS18] Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Accelerated methods for nonconvex optimization. *SIAM Journal on Optimization*, 28(2):1751–1772, 2018.

[CGFLJ19] Tatjana Chavdarova, Gauthier Gidel, François Fleuret, and Simon Lacoste-Julien. Reducing noise in GAN training with variance reduced extragradient. In *NIPS*, 2019.

[CJST19] Yair Carmon, Yujia Jin, Aaron Sidford, and Kevin Tian. Variance reduction for matrix games. In *Advances in Neural Information Processing Systems*, pages 11381–11392, 2019.

[CJST20] Yair Carmon, Yujia Jin, Aaron Sidford, and Kevin Tian. Coordinate methods for matrix games. *FOCS*, 2020.

[CP11] 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.

[CP16] Antonin Chambolle and Thomas Pock. On the ergodic convergence rates of a first-order primal–dual algorithm. *Mathematical Programming*, 159(1-2):253–287, 2016.

[DBLJ14] 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.

[DCL+17] Simon S. Du, Jianshu Chen, Lihong Li, Lin Xiao, and Dengyong Zhou. Stochastic variance reduction methods for policy evaluation. In *ICML*, 2017.

[Def16] Aaron Defazio. A simple practical accelerated method for finite sums. In *NIPS*, 2016.

[DSL+18] Bo Dai, Albert Shaw, Lihong Li, Lin Xiao, Niao He, Zhen Liu, Jianshu Chen, and Le Song. SBEED: Convergent reinforcement learning with nonlinear function approximation. In *ICML*, 2018.

[FLLZ18] Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. SPIDER: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *NIPS*, 2018.

[HLLJM15] Thomas Hofmann, Aurelien Lucchi, Simon Lacoste-Julien, and Brian McWilliams. Variance reduced stochastic gradient descent with neighbors. *Advances in Neural Information Processing Systems*, 28, 2015.

[HLOY18] Robert Hannah, Yanli Liu, Daniel O'Connor, and Wotao Yin. Breaking the span assumption yields fast finite-sum minimization. In *Advances in Neural Information Processing Systems*, pages 2312–2321, 2018.

[IAGM19] Adam Ibrahim, Waïss Azizian, Gauthier Gidel, and Ioannis Mitliagkas. Linear lower bounds and conditioning of differentiable games. *arXiv preprint arXiv:1906.07300*, 2019.

[Joa05] Thorsten Joachims. A support vector method for multivariate performance measures. In *Proceedings of the 22nd international conference on Machine learning*, pages 377–384, 2005.

[JZ13] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS*, 2013.

[KHR20] Dmitry Kovalev, Samuel Horváth, and Peter Richtárik. Don't jump through hoops and remove those loops: Svrg and katyusha are better without the outer loop. In *Algorithmic Learning Theory*, pages 451–467. PMLR, 2020.

[LBJM+20] Nicolas Loizou, Hugo Berard, Alexia Jolicoeur-Martineau, Pascal Vincent, Simon Lacoste-Julien, and Ioannis Mitliagkas. Stochastic hamiltonian gradient methods for smooth games. In *International Conference on Machine Learning*, pages 6370–6381. PMLR, 2020.

[LBZR21] Zhize Li, Hongyan Bao, Xiangliang Zhang, and Peter Richtárik. Page: A simple and optimal probabilistic gradient estimator for nonconvex optimization. In *International Conference on Machine Learning*, pages 6286–6295. PMLR, 2021.

[LCL+19] 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.

[Li21] Zhize Li. Anita: An optimal loopless accelerated variance-reduced gradient method. *arXiv preprint arXiv:2103.11333*, 2021.

[LJJ20] Tianyi Lin, Chi Jin, and Michael I Jordan. Near-optimal algorithms for minimax optimization. In *Conference on Learning Theory*, pages 2738–2779. PMLR, 2020.

[LMG20] Bingcong Li, Meng Ma, and Georgios B Giannakis. On the convergence of sarah and beyond. In *International Conference on Artificial Intelligence and Statistics*, pages 223–233. PMLR, 2020.

[LMH18] 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.

[LXZZ21] Luo Luo, Guangzeng Xie, Tong Zhang, and Zhihua Zhang. Near optimal stochastic algorithms for finite-sum unbalanced convex-concave minimax optimization. *arXiv preprint arXiv:2106.01761*, 2021.

[LY19] Guanghui Lan and Yu Yang. Accelerated stochastic algorithms for nonconvex finite-sum and multiblock optimization. *SIAM Journal on Optimization*, 29(4):2753–2784, 2019.

[LYHZ20] Luo Luo, Haishan Ye, Zhichao Huang, and Tong Zhang. Stochastic recursive gradient descent ascent for stochastic nonconvex-strongly-concave minimax problems. *Advances in Neural Information Processing Systems*, 33:20566–20577, 2020.

[LZ17] Guanghui Lan and Yi Zhou. An optimal randomized incremental gradient method. *Mathematical programming*, pages 1–49, 2017.

[MOP19a] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. Proximal point approximations achieving a convergence rate of $O(1/k)$ for smooth convex-concave saddle point problems: Optimistic gradient and extra-gradient methods. *arXiv preprint:1906.01115*, 2019.

[MOP19b] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach. *arXiv preprint:1901.08511*, 2019.

[Nes13] Yurii Nesterov. *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media, 2013.

[NLST17] Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International Conference on Machine Learning*, pages 2613–2621. PMLR, 2017.

[OLR20] Dmitrii M Ostrovskii, Andrew Lowy, and Meisam Razaviyayn. Efficient search of first-order nash equilibria in nonconvex-concave smooth min-max problems. *arXiv preprint arXiv:2002.07919*, 2020.

[OX18] Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *arXiv preprint:1808.02901*, 2018.

[PB16] Balamurugan Palaniappan and Francis Bach. Stochastic variance reduction methods for saddle-point problems. In *NIPS*, 2016.

[QQR21] Xun Qian, Zheng Qu, and Peter Richtárik. L-svrg and l-katyusha with arbitrary sampling. 2021.

[RLLY18] Hassan Rafique, Mingrui Liu, Qihang Lin, and Tianbao Yang. Non-convex min-max optimization: Provable algorithms and applications in machine learning. *arXiv preprint arXiv:1810.02060*, 2018.

[SMZ+18] 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.

[SS16] Shai Shalev-Shwartz. Sdca without duality, regularization, and individual convexity. In *International Conference on Machine Learning*, pages 747–754. PMLR, 2016.

[SSZ13] Shai Shalev-Shwartz and Tong Zhang. Stochastic dual coordinate ascent methods for regularized loss minimization. *Journal of Machine Learning Research*, 14(1), 2013.

[TJNO19] Kiran K. Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Efficient algorithms for smooth minimax optimization. In *NIPS*, 2019.

[TZML18] Conghui Tan, Tong Zhang, Shiqian Ma, and Ji Liu. Stochastic primal-dual method for empirical risk minimization with O(1) per-iteration complexity. In *NIPS*, 2018.

[WS16] Blake Woodworth and Nathan Srebro. Tight complexity bounds for optimizing composite objectives. In *NIPS*, 2016.

[XZ14] Lin Xiao and Tong Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.

[YWL16] Yiming Ying, Longyin Wen, and Siwei Lyu. Stochastic online AUC maximization. In *NIPS*, 2016.

[YXL+19] 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.

[YZKH20] Junchi Yang, Siqi Zhang, Negar Kiyavash, and Niao He. A catalyst framework for minimax optimization. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020.

[ZG19] Dongruo Zhou and Quanquan Gu. Lower bounds for smooth nonconvex finite-sum optimization. In *ICML*, 2019.

[ZHZ19] Junyu Zhang, Mingyi Hong, and Shuzhong Zhang. On lower iteration complexity bounds for the saddle point problems. *arXiv preprint:1912.07481*, 2019.

[ZX17] 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.

[ZXG18] Dongruo Zhou, Pan Xu, and Quanquan Gu. Stochastic nested variance reduction for nonconvex optimization. *Advances in Neural Information Processing Systems*, 31:3921–3932, 2018.

[ZYG+21] Siqi Zhang, Junchi Yang, Cristóbal Guzmán, Negar Kiyavash, and Niao He. The complexity of nonconvex-strongly-concave minimax optimization. In *Uncertainty in Artificial Intelligence*, pages 482–492. PMLR, 2021.