DISS. ETH NO. 29718

TOWARDS NEAR-OPTIMAL AND ADAPTIVE ALGORITHMS IN MINIMAX OPTIMIZATION

A thesis submitted to attain the degree of

DOCTOR OF SCIENCES of ETH ZURICH
(Dr. sc. ETH Zurich)

presented by

JUNCHI YANG
Master of Science, University of Illinois Urbana-Champaign
born on June 13, 1995

accepted on the recommendation of

Prof. Dr. Niao He, examiner
Prof. Dr. Thomas Hofmann, co-examiner
Prof. Dr. Negar Kiyavash, co-examiner
Prof. Dr. Meisam Razaviyayn, co-examiner

2023

JUNCHI YANG

TOWARDS NEAR-OPTIMAL AND ADAPTIVE ALGORITHMS IN MINIMAX OPTIMIZATION

To a world united in peace, equality, happiness, and dignity of individuals.

ABSTRACT

In machine learning, the training process refines models by extracting patterns from vast datasets. This refinement typically hinges on an optimization formulation that minimizes a task-specific loss function over the model's parameters. While numerous efficient first-order optimization methods exist, they may fall short in addressing contemporary machine learning tasks. For instance, Generative Adversarial Networks (GANs) employ a two-player approach: one player generates data emulating the training data distribution, while another discerns between the generated and real data. Also, Adversarial Training trains a model against worst-case scenarios, anticipating perturbations in the training data. Such *two-player* or *worst-case* tasks are typically framed as minimax optimization, where one variable seeks to minimize a loss function, and another aims to maximize it.

Minimax optimization, expressed as $\min_x \max_y f(x, y)$, is pivotal across various domains. While the study on it can be traced back to game theory and variational inequalities, we spotlight three predominant challenges in its application to modern machine learning: asymmetry, non-convexity, and adaptivity.

The initial part of this thesis addresses the asymmetry challenge. In real-world scenarios, the loss function often exhibits asymmetrical convexity/concavity properties for two variables. For instance, it might be nonconvex with respect to x but concave with respect to y. Optimal algorithms for such unbalanced minimax scenarios remain elusive, especially when the objective function adopts a finite-sum form. Current solutions for these unbalanced tasks are intricate, with distinct algorithms tailored to specific settings. We propose a universal "Catalyst" framework, drawing inspiration from proximal point methods. This approach solves a series of regularized problems using balanced-regime algorithms, achieving near-optimal or state-of-the-art complexities in unbalanced settings.

The subsequent part delves into problems that are nonconvex in x and nonconcave in y simultaneously. While some studies highlight the intractability of general nonconvex-nonconcave minimax problems, we argue that discerning unique structures can pave the way for efficient algorithms. For instance, when the objective function satisfies the Polyak-Łojasiewicz inequality for both variables, we demonstrate that the Alternating Gradient Descent Ascent (AGDA) — a single-loop, prevalent algorithm — can pinpoint the global solution. If the inequality holds for just one variable, AGDA and its regularized counterpart can find stationary points.

Lastly, our focus shifts to adaptive methods for nonconvex minimax optimization, aiming to obviate stepsize tuning. We observe that Gradient Descent Ascent, when paired with prevalent adaptive stepsize schemes, still fails to converge without manual tuning. This inconsistency might underpin the unstable training observed in minimax optimization, especially in GANs. We introduce a nested-loop algorithm, combined with AdaGrad, that adaptively balances updates in x and y, ensuring convergence without stepsize tuning.

ZUSAMMENFASSUNG

Im maschinellen Lernen verfeinert der Trainingsprozess Modelle, indem er Muster aus umfangreichen Datensätzen extrahiert. Diese Verfeinerung basiert in der Regel auf einer Optimierungsformulierung, die eine aufgabenbezogene Verlustfunktion über die Parameter des Modells minimiert. Obwohl zahlreiche effiziente Optimierungsmethoden erster Ordnung existieren, können sie bei der Behandlung zeitgenössischer maschineller Lernaufgaben an ihre Grenzen stoßen. Zum Beispiel verwenden Generative Adversarial Networks (GANs) einen Zwei-Spieler-Ansatz: Ein Spieler erzeugt Daten, die die Verteilung der Trainingsdaten nachahmen, während ein anderer zwischen den erzeugten und echten Daten unterscheidet. Auch das Adversarial Training schult ein Modell gegen Worst-Case-Szenarien und erwartet Störungen in den Trainingsdaten. Solche Zwei-Spieler- oder Worst-Case-Aufgaben werden in der Regel als Minimax-Optimierung formuliert, bei der eine Variable versucht, eine Verlustfunktion zu minimieren, und eine andere sie zu maximieren.

Die Minimax-Optimierung, ausgedrückt als $\min_x \max_y f(x, y)$, ist in verschiedenen Bereichen von zentraler Bedeutung. Obwohl die Studie dazu auf die Spieltheorie und Variationsungleichheiten zurückgeführt werden kann, beleuchten wir drei vorherrschende Herausforderungen bei ihrer Anwendung auf modernes maschinelles Lernen: Asymmetrie, Nicht-Konvexität und Adaptivität.

Der Anfangsteil dieser Arbeit befasst sich mit der Herausforderung der Asymmetrie. In realen Szenarien zeigt die Verlustfunktion oft asymmetrische Konvexitäts- und Konkavitätseigenschaften für zwei Variablen. Zum Beispiel könnte sie in Bezug auf x nicht konvex sein, aber in Bezug auf y konkav. Optimale Algorithmen für solche unausgewogenen Minimax-Szenarien sind schwer zu finden, insbesondere wenn die Zielfunktion eine endliche Summenform annimmt. Aktuelle Lösungen für diese unausgewogenen Aufgaben sind komplex und es werden spezifische Algorithmen für bestimmte Einstellungen entwickelt. Wir schlagen ein universelles “Catalyst”-Framework vor, das von proximalen Punktmethoden inspiriert ist. Dieser Ansatz löst eine Reihe von regularisierten Problemen mit Algorithmen aus ausgewogenen Regimen und erreicht nahezu optimale oder Spitzenkomplexitäten in unausgewogenen Einstellungen.

Note: The translation aims to preserve the meaning and context of the original text while adapting it to the German language structure.

Der nachfolgende Teil geht auf Probleme ein, die gleichzeitig in x nicht konvex und in y nicht konkav sind. Während einige Studien die Unlösbarkeit allgemeiner nicht

konvexer-nicht konkaver Minimax-Probleme hervorheben, argumentieren wir, dass das Erkennen einzigartiger Strukturen den Weg für effiziente Algorithmen ebnen kann. Wenn die Zielfunktion zum Beispiel die Polyak-Łojasiewicz-Ungleichung für beide Variablen erfüllt, zeigen wir, dass der Alternating Gradient Descent Ascent (AGDA) — ein weit verbreiteter Einzelschleifen-Algorithmus — die globale Lösung finden kann. Wenn die Ungleichung nur für eine Variable gilt, können AGDA und sein regularisiertes Gegenstück stationäre Punkte finden.

Schließlich konzentrieren wir uns auf adaptive Methoden für nicht konvexe Minimax-Optimierung, um die Schrittwertabstimmung zu vermeiden. Wir stellen fest, dass Gradient Descent Ascent, wenn er mit gängigen adaptiven Schrittwertschemata kombiniert wird, immer noch nicht ohne manuelle Abstimmung konvergiert. Diese Inkonsistenz könnte das instabile Training untermauern, das bei der Minimax-Optimierung beobachtet wurde. Wir führen einen verschachtelten Schleifenalgorithmus ein, der mit AdaGrad kombiniert wird und die Updates in x und y adaptiv ausgleicht, um eine Konvergenz ohne Schrittwertabstimmung zu gewährleisten.

ACKNOWLEDGEMENTS

I wish to convey my deepest appreciation, beyond merely naming individuals.

First and foremost, my gratitude goes to my advisor, Niao He. She introduced me to the fascinating realm of optimization. Throughout my Ph.D. journey, her guidance has been invaluable. She granted me the freedom to explore diverse academic topics and always encouraged my curiosity. Her mentoring style will undoubtedly influence me for years to come.

I extend my thanks to all my committee members. A special mention to Negar Kiyavash, my co-advisor at UIUC, for her collaboration over the years. Meeting Meisam Razaviyayn five years ago at UIUC was pivotal; his work has significantly influenced several chapters of this thesis. I'm also grateful to Thomas Hofmann, my second advisor at ETH Zurich, for his consistent encouragement and invaluable advice.

My heartfelt thanks go out to my wonderful collaborators: Aurelien Lucchi, Amin Karbasi, Cristóbal Guzmán, Xiang Li, Ilyas Fatkhullin, Liang Zhang, Antonio Orvieto, and Siqi Zhang. Their contributions were instrumental in realizing many of the works presented. I also cherish the interactions with the master's thesis students I supervised at ETH: Florian Hübler, Martin Popeski, and Xiang Li. Their insights were truly inspiring.

As I approached graduation and ventured into the job market, the support from Niao and Pragnya Alatur was indispensable, especially in preparing my job talk. I'm also indebted to numerous friends who offered invaluable career advice.

Outside the academic sphere, I've been blessed with unwavering support. I owe a great deal to my parents for their foresight regarding my education and their constant encouragement of my choices. Their sacrifices have afforded me incredible educational opportunities. A special thanks to Esther Tang, who stood by me during my most challenging years at UIUC. Without her, I might have prematurely ended this journey. I'm also grateful to Jinglin Chen for being a wonderful roommate in Champaign. Lastly, my heartfelt gratitude to my girlfriend, Yi Zhang, for her enduring support over the past three years and for her understanding of my often hectic schedule.

Lastly, I'd like to acknowledge the serenity I found in Champaign, especially the sunsets at the Lake of the Woods, and the vibrant life I experienced in Chicago with Yi.

To all those not mentioned, but who have been a part of this journey, thank you.

PUBLICATIONS

The following are publications covered in this thesis (* indicates equal contribution):

- [Yang et al., 2020a] Junchi Yang, Negar Kiyavash, and Niao He. "Global Convergence and Variance Reduction for a Class of Nonconvex-Nonconcave Minimax Problems" Conference on Neural Information Processing Systems (NeurIPS) 2020.
- [Yang et al., 2020b] Junchi Yang, Siqi Zhang, Negar Kiyavash, and Niao He. "A Catalyst Framework for Minimax Optimization" Conference on Neural Information Processing Systems (NeurIPS) 2020.
- [Zhang et al., 2021b] Siqi Zhang*, Junchi Yang*, Cristobal Guzman, Negar Kiyavash and Niao He. "The Complexity of Nonconvex-strongly-concave Minimax Optimization" Conference on Uncertainty in Artificial Intelligence (UAI) 2021.
- [Yang et al., 2022b] Junchi Yang, Antonio Orvieto, Aurelien Lucchi, and Niao He. "Faster Single-loop Algorithms for Minimax Optimization without Strong Concavity" Artificial Intelligence and Statistics Conference (AISTATS) 2022.
- [Yang et al., 2022a] Junchi Yang*, Xiang Li*, and Niao He. "Nest Your Adaptive Algorithm for Parameter-agnostic Nonconvex Minimax Optimization" Conference on Neural Information Processing Systems (NeurIPS) 2022.
- [Yang et al., 2023] Junchi Yang*, Xiang Li*, Ilyas Fatkhullin, and Niao He. "Two Sides of One Coin: the Limits of Untuned SGD and the Power of Adaptive Methods" Conference on Neural Information Processing Systems (NeurIPS) 2023.

The following works are closely related but not covered in this thesis:

- [Li et al., 2023] Xiang Li, Junchi Yang, and Niao He. "TiAda: A Time-scale Adaptive Algorithm For Nonconvex Minimax Optimization" International Conference on Learning Representations (ICLR) 2023.
- [Zhang et al., 2023] Liang Zhang*, Junchi Yang*, Amin Karbasi, Niao He. "Optimal Guarantees for Algorithmic Reproducibility and Gradient Complexity in Convex Optimization" Conference on Neural Information Processing Systems (NeurIPS) 2023.

Contents

ACRONYM

SC-SC Strongly-Convex-Strongly-Concave

SC-C Strongly-Convex-Concave

C-C Convex-Concave

NC-SC Nonconvex-Strongly-Concave

NC-C Nonconvex-Concave

NC-NC Nonconvex-Nonconcave

PL Polyak-Łojasiewicz

GD Gradient Descent

SGD Stochastic Gradient Descent

GDA Gradient Descent Ascent

AGDA Alternating Gradient Descent Ascent

EG Extragradient

OGDA Optimistic Gradient Descent Ascent

SVRG Stochastic Variance Reduced Method

GAN Generative Adversarial Networks

WGAN Wasserstein Generative Adversarial Networks

SAM Sharpness-Aware Optimization

DRO Distributional Robustness Optimization

1

INTRODUCTION

We can only see a short distance ahead, but we can see plenty there that needs to be done.

— Alan Turing

Machine learning, particularly deep learning, has catalyzed breakthroughs across a diverse array of applications, ranging from image recognition and natural language processing to autonomous vehicles and healthcare. Central to these advancements is a fundamental concept: optimization. Many traditional applications can be formulated as a simple optimization problem $\min_x f(x)$, where the goal is to minimize a loss function over the decision variable, often the weights of a neural network. A straightforward first-order algorithm, (stochastic) gradient descent, has proven effective in machine learning. As noted in the book "Deep Learning" [Goodfellow et al., 2016], "Nearly all of deep learning is powered by one very important algorithm: stochastic gradient descent (SGD)." However, with the rapid evolution and emerging trends in machine learning, the sufficiency of such a minimization formulation is being questioned.

Despite the remarkable capabilities of machine learning models, their trustworthiness, particularly in high-stakes real-world scenarios, has been a subject of concern. Trustworthiness in machine learning encompasses a broad set of characteristics, including reliability, interpretability, fairness, and privacy. For instance, a slight, carefully crafted alteration to an input image — undetectable to the human eye — can cause a state-of-the-art image classifier to misclassify the object in the image, illustrating a vulnerability in the model's robustness [Szegedy et al., 2013]. Adversarial Training methods [Goodfellow et al., 2015, Madry et al., 2017] have been introduced to improve robustness by augmenting the training data with adversarial data. This approach can be viewed as a defender seeking the best model under the worst-case scenario when an attacker perturbs the data to degrade the model. Moreover, concerns about fairness have been raised, with biases towards certain population groups found to exist in various machine learning systems [Mehrabi et al., 2021]. To mitigate such issues, a line of work [Agarwal et al., 2018] formulates fairness as a constraint in the optimization process, preventing the model from discriminating against certain groups. Many of these constraints are nontrivial and challenging to handle in the optimization process.

Generative Adversarial Networks (GANs) [Goodfellow et al., 2014] have gained tremendous popularity over the past decade. GANs are naturally formulated as a contest between

a Generator network, which tries to generate realistic data, and a Discriminator network, which tries to distinguish generated data from real data. GANs have found wide applications, including generating artwork, improving astronomical images [Schawinski et al., 2017], and image-to-image translation [Isola et al., 2017]. However, the training of GANs is known to be challenging, often leading to non-convergence and instability.

Another trend in deep learning is the use of over-parametrization. For example, GPT-3 has 175 billion parameters [Brown et al., 2020], and many other domains use over-parametrized neural networks to achieve state-of-the-art results [Tan and Le, 2019]. While over-parametrization simplifies the task of optimization algorithms in finding points with low loss or even global solutions [Ma et al., 2018], there may exist a large set of global minima with different test errors. Therefore, simply minimizing the loss function may not necessarily yield good generalization. Some work [Keskar et al., 2016] connects the geometry of the loss function (such as sharpness) around a solution with its generalization errors. Instead of just finding a solution with a small loss, we would also want to ensure that it satisfies certain geometrical properties [Foret et al., 2021].

From trustworthy machine learning and GANs to optimization in the over-parametrized regime, we realize that many of the modern applications may not be adequately captured by the simple formulation $\min_x f(x)$. This realization motivates us to explore a particular form of optimization that shows promise in addressing these challenges: minimax optimization. This concept will be explored in depth in the following sections of this thesis.

1.1 MINIMAX OPTIMIZATION

Minimax optimization serves as a cornerstone in contemporary machine learning, with its applications extending across a broad spectrum of areas. These include, but are not limited to, Generative Adversarial Networks [Goodfellow et al., 2014, Arjovsky et al., 2017], adversarial learning [Goodfellow et al., 2015, Miller et al., 2020], reinforcement learning [Dai et al., 2017, Modi et al., 2021], sharpness-aware minimization [Foret et al., 2021], domain-adversarial training [Ganin et al., 2016]. A minimax problem can be mathematically expressed as:

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \; f(x, y), \tag{1.1}$$

where $\mathcal{X}$ and $\mathcal{Y}$ are the domains of the decision variables x and y, respectively, and $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is the objective function. In the context of adversarial machine learning, for instance, x could symbolize the parameters of a model, y could denote adversarial perturbations to the input data, and $f(x, y)$ could represent the loss function that the model aims to minimize.

Minimax optimization often manifests in the form of finite-sum or stochastic optimization in machine learning scenarios:

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} f(x,y) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_i(x,y), \qquad \text{(finite-sum form)}$$

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} f(x,y) \triangleq \mathbb{E}_{\xi\sim P}\left[F(x,y;\xi)\right], \qquad \text{(stochastic form)}$$

where P is an unknown distribution. The finite-sum form naturally arises in empirical risk minimization (ERM), where the objective is to find model parameters that minimize the average loss over a given dataset, with each f_i in the sum corresponding to the loss on a single data point. The stochastic form, on the other hand, is a general form that may arise when the data follows a distribution P, and $F(x,y;\xi)$ represents the loss function for a single data point ξ.

Minimax optimization is intrinsically linked to two-player zero-sum games [Von Neumann et al., 2007]. In such a game, Player 1 chooses strategies from set $\mathcal{X}$ and Player 2 from set $\mathcal{Y}$. Given that Player 1 adopts strategy x and Player 2 strategy y, the payoff for Player 1 is $-f(x,y)$ and for Player 2, it's $f(x,y)$. As both players aim to maximize their payoff, Player 1 seeks to minimize f, while Player 2 attempts to maximize it.

In the ensuing sections, we will delve deeper into the applications of minimax optimization across various domains, elucidate definitions for different notions of optimality, and discuss the challenges associated with minimax optimization.

1.2 APPLICATIONS

While we have listed numerous applications of minimax optimization in machine learning, we will delve into three specific ones: Generative Adversarial Networks, Sharpness-Aware Minimization, and fairness in machine learning. Our aim is to elucidate how the minimax formulation manifests and plays a crucial role in these domains.

1.2.1 *Generative Adversarial Networks*

Generative Adversarial Networks (GANs) [Goodfellow et al., 2014] represent a generative modeling approach, designed to create new data instances that closely resemble a given set of training data. GANs comprise two neural networks: the Generator $G_\theta(\cdot)$, parameterized by θ, and the Discriminator $D_\omega(\cdot)$, parameterized by ω. These networks are trained simultaneously through a two-player minimax game.

The Generator network G_θ takes random noise as input and generates samples as output. Its goal is to produce data that is indistinguishable from real data. On the other hand, the Discriminator network D_ω aims to distinguish between real and generated data. Let p_z denote the distribution of the random noise and p_{data} denote the distribution of the real data. When the input x originates from the real data distribution, the Discriminator seeks to maximize $\mathbb{E}_{x\sim p_{\text{data}}}[\log D_\omega(x)]$. When the input z comes from the noise distribution p_z, the Discriminator aims to minimize $\mathbb{E}_{z\sim p_z}[\log(1 - D_\omega(G_\theta(z)))]$, while the Generator seeks to maximize it.

Consequently, the objective function for a standard GAN can be formulated as a minimax optimization problem:

$$\min_\theta \max_\omega L(G_\theta, D_\omega) = \mathbb{E}_{x\sim p_{\text{data}}}[\log D_\omega(x)] + \mathbb{E}_{z\sim p_z}[\log(1 - D_\omega(G_\theta(z)))].$$

The loss function above is closely related to the Jensen-Shannon divergence between two distributions. Let p_g denote the distribution of the generated distribution by the Generator, i.e., $p_g^\theta(x) = \mathbb{P}(G_\theta(z) = x)$ with $z \sim p_z$. The optimal Generator with respect to the Discriminator D_ω that maximizes $L(G_\theta, \cdot)$ can be shown to be $[D^*(G_\theta)](x) = \frac{p_{data}(x)}{p_{data}(x)+p_g^\theta(x)}$. Assuming the Generator network is expressive enough that there exists an optimal parameter $\omega^*(\theta)$ such that $D_{\omega^*(\theta)} = D^*(G_\theta)$, it can be further shown that the loss function is a similar quantity as the Jensen-Shannon divergence between the real data distribution and generated distribution when the discriminator is optimal, i.e., $L(G_\theta, D^*(G_\theta)) = 2D_{JS}(p_{data}\|p_g^\theta) - 2\log 2$, where $D_{JS}(\cdot\|\cdot)$ denotes the Jensen-Shannon divergence between two distributions.

GANs are often found to suffer from training stability issues, such as mode collapse [Goodfellow, 2016]. Mode collapse occurs when the generator produces limited diversity in the samples, often generating very similar or even identical samples. The Wasserstein GAN (WGAN) [Arjovsky et al., 2017] mitigates this issue by using the Wasserstein distance, a metric that provides a more meaningful measure of the difference between the generated and real data distributions. WGAN aims to minimize the Wasserstein Distance between the distribution of the real data and generated data, $W(p_{data}, p_g)$. Although the Wasserstein Distance itself is hard to compute, by Kantorovich-Rubinstein duality, it equals to

$$W\left(p_{data}, p_g\right) = \frac{1}{K} \sup_{\|D\|_L \le K} \mathbb{E}_{x\sim p_{data}}[D(x)] - \mathbb{E}_{x\sim p_g}[D(x)],$$

where $D(\cdot)$ is a function and $\|\cdot\|_L$ denotes the Lipchitz norm. Therefore, WGAN can be formulated as a minimax optimization problem:

$$\min_{\theta} \max_{\omega: D_\omega \in \mathcal{D}} \mathbb{E}_{x \sim p_{\text{data}}}[D_\omega(x)] - \mathbb{E}_{z \sim p_z}[D_\omega(G_\theta(z))].$$

Here, $\mathcal{D}$ is the set of K-Lipschitz functions.

To summarize, GAN and WGAN correspond to different distance metrics, and both of them are formulated as minimax optimization. Other metrics are also considered in the literature, for example, the family of f-divergence [Nowozin et al., 2016] and Sobolev integral probability metric [Mroueh et al., 2017]. While GANs have been highly successful in various applications, their training can be challenging.

1.2.2 *Sharpness-Aware Minimization*

Sharpness-Aware Minimization (SAM) [Foret et al., 2021] is an optimization procedure proposed to enhance the generalization performance. A conceptually similar approach, termed Adversarial Model Perturbation, was independently proposed by Zheng et al. [2021]. In the context of over-parameterized models, multiple solutions might yield the same loss, but their resultant model qualities can differ. Recognizing the correlation between function sharpness at a solution and the generalization bound, SAM's core idea is to concurrently minimize both the loss value and its sharpness. This approach contrasts with traditional methods that prioritize only the reduction of training loss.

SAM is looking for a parameter that maintains low loss in a neighborhood around it. Formally, given a loss function $L(\theta)$, where θ represents the model parameters, SAM aims to solve the following minimax optimization problem:

$$\min_{\theta} L^{SAM}(\theta) + \lambda\|\theta\|^2, \text{ with } L^{SAM}(\theta) = \max_{\delta: \|\delta\| \leq \rho} L(\theta + \delta),$$

where $\lambda > 0$ is a regularization parameter and $\rho > 0$ is the diameter of the neighborhood. This formulation ensures that the selected parameters lie in a neighborhood with uniformly low loss values, and therefore the local sharpness of the loss function is potentially low.

While L^{SAM} above represents the worse-case loss in ℓ_2 ball neighborhood of θ, several variants of sharpness-aware loss have been proposed. For example, Kwon et al. [2021] highlight that the aforementioned loss does not have the scale-invariant property, i.e., $\max_{\|\delta\| \leq \rho} L(\theta + \delta) \neq \max_{\|\delta\| \leq \rho} L(A\theta + \delta)$ even with a scaling operator A such that $L(\theta) = L(A\theta)$. To address this, they first introduce a family of normalization operator $\{T_\theta : \theta \in$

$\mathcal{R}^k\}$ which satisfies $T_{A\theta}^{-1}A = T_\theta^{-1}$ for any invertible operator A with $L(\theta) = L(A\theta)$. They propose to use a different SAM loss:

$$L^{SAM}(\theta) = \max_{\delta:\|T_\theta^{-1}\delta\|\leq\rho} L(\theta+\delta).$$

One potential normalization operator is $T_\theta = \text{diag}\,(|\theta_1|, \ldots, |\theta_k|)$ with $\theta = [\theta_1, \theta_2, \ldots, \theta_k]$. This revised SAM loss ensures scale-invariance, that is, $\max_{\|T_\theta^{-1}\delta\|\leq\rho} L(\theta+\delta) = \max_{\|T_{A\theta}^{-1}\delta\|\leq\rho} L(A\theta+\delta)$. This still leads to a minimax optimization, albeit with a different constraint for the dual variable δ.

In practice, SAM improves model generalization across a variety of tasks, including image classification, fine-tuning, and machine translation. Notably, SAM also exhibits robustness to label noise, a characteristic that aligns it with the performance of existing methods designed to handle noisy data [Foret et al., 2021].

1.2.3 *Machine Learning with Fairness*

In machine learning applications that influence critical societal decisions—like credit lending, resource allocation, and job opportunities—it's imperative to ensure fairness. A model trained solely to minimize loss might inadvertently harbor biases against certain groups. Minimax optimization emerges as a pivotal tool in the realm of fair machine learning. In this section, we will delve into two applications: one where fairness is encapsulated as a constraint and reformulated using the Lagrangian, and another that aims to optimize the worst-case loss across diverse groups.

Agarwal et al. [2018] explored a binary classification scenario. Training samples are represented as (X, A, Y), where X is the feature vector, $A \in \mathcal{A}$ is a protected attribute (e.g., race), and $Y \in \{0, 1\}$ is the label. The objective is to derive a classifier $h : X \to \{0, 1\}$ from the classifier family $\mathcal{H}$, with a classifier's loss denoted by $L(h)$. To instill fairness, a linear constraint, $M\mu(h) \leq c$, is introduced. An example of a fairness criterion fitting this mold is demographic parity, which mandates equal selection rates across groups. This can be expressed as:

$$\mathbb{E}[h(X) \mid A = a] - \mathbb{E}[h(X)] \leq 0, \quad -\mathbb{E}[h(X) \mid A = a] + \mathbb{E}[h(X)] \leq 0.$$

Various fairness criteria can be articulated through the linear constraint $M\mu(h) \leq c$. One example is demographic parity, i.e., $\mathbb{E}[h(X) \mid A = a] = \mathbb{E}[h(X)]$ for all a. For a randomized classifier Q, the overarching optimization problem becomes:

$$\min_{Q \in \Delta(\mathcal{H})} L(Q) \text{ subject to } M\mu(Q) \leq c,$$

which can be further transformed into a minimax optimization using the Lagrangian:

$$\min_{Q \in \Delta(\mathcal{H})} \max_{\lambda \in \mathbb{R}^j_+} L(Q) + \lambda^\intercal (M\mu(Q) - c).$$

This minimax optimization adeptly manages the constraint for the decision variable, sidestepping the complexities of projection.

While the aforementioned method adopts fairness criteria that directly mitigate differences between groups, other approaches strive to minimize the maximum loss across all groups. For a classification task with data (X, A, Y), the group-specific loss, $L_a(h) = \mathbb{E}\left[L(h) \mid A = a\right]$ is defined for $a \in \mathcal{A}$. The objective is to find a randomized classifier Q that minimizes the maximum loss across all groups [Diana et al., 2021]:

$$\min_{Q \in \Delta(\mathcal{H})} \max_{a \in \mathcal{A}} L_a(Q).$$

Martinez et al. [2020] also consider a similar minimax formulation, but restricted their focus to the set of Pareto optimal classifiers, $P_{\mathcal{A},\mathcal{H}} = \{h \in \mathcal{H} : \nexists h' \in \mathcal{H}$ such that $L_a(h') \leq L_a(h) \forall a \in \mathcal{A}$ and $L_{a'}(h') < L_{a'}(h)$ for some $a'\}$, instead of all classifiers in $\Delta(\mathcal{H})$. These classifiers ensure that no other classifier performs equally well across all group-specific losses and is strictly better in one of the groups.

1.3 OPTIMALITY AND EQUILIBRIA

Defining optimality within minimax optimization is not an easy task due to its two-player nature, involving one variable that seeks to minimize and another aiming to maximize. Its notions of optimality are sometimes termed equilibria. We will begin by clarifying our terminology. Subsequently, we will delve into the concept of global solutions, stationary points, and local solutions.

1.3.1 *Settings and Terminology*

Throughout this thesis, we will assume the objection function f in equation (1.1) to be differentiable and ℓ-Lipschitz smooth, and the domain sets $\mathcal{X}$ and $\mathcal{Y}$ to be closed and convex.

Assumption 1 (Lipschitz smoothness). *There exists a positive constant $\ell > 0$ such that*

$$\max\left\{ \left\|\nabla_x f(x_1, y_1) - \nabla_x f(x_2, y_2)\right\|, \left\|\nabla_y f(x_1, y_1) - \nabla_y f(x_2, y_2)\right\| \right\} \leq \ell[\|x_1 - x_2\| + \|y_1 - y_2\|],$$

holds for all $x_1, x_2 \in \mathcal{X}$ and $y_1, y_2 \in \mathcal{Y}$.

We will categorize different minimax optimization settings based on their convexity in x and concavity in y. Below, we provide the definitions of convexity and strong convexity for differentiable functions.

Definition 1 (Convexity). *A differentiable function $g : \mathcal{X} \to \mathbb{R}$ is convex if for all $x_1, x_2 \in \mathcal{X}$, we have*

$$g(x_2) \geq g(x_1) + \nabla g(x_1)^\intercal (x_2 - x_1).$$

Definition 2 (Strong Convexity). *A function $g : \mathcal{X} \to \mathbb{R}$ is μ-strongly convex if for all $x_1, x_2 \in \mathcal{X}$, we have*

$$g(x_2) \geq g(x_1) + \nabla g(x_1)^\intercal (x_2 - x_1) + \frac{\mu}{2}\|x_1 - x_2\|^2.$$

If function $-g$ is convex, then g is concave. Similarly, if function $-g$ is μ-strongly convex, then g is μ strongly concave. We will also introduce the concepts of weak convexity and the Polyak-Łojasiewicz (PL) inequality. Fréchet sub-differential of a function g at x is defined as the set $\partial g(x) = \left\{u \mid \liminf_{x' \to x} g(x') - g(x) - u^\intercal (x' - x) / \|x' - x\| \geq 0\right\}$.

Definition 3 (Weak Convexity). *A function $g : \mathcal{X} \to \mathbb{R}$ is ν-weakly convex[1] if for all $x_1, x_2 \in \mathcal{X}$ and all $u \in \partial g(x_1)$, we have*

$$g(x_2) \geq g(x_1) + u^\top (x_2 - x_1) - \frac{\nu}{2}\|x_1 - x_2\|^2.$$

If a differentiable function g is ℓ-smooth, it is also ℓ-weakly convex. This is because $\nabla g(x)$ is the only element in $\partial g(x)$, and ℓ-smoothness implies $-g(x_2) \leq -g(x_1) - \nabla g(x_1)^\intercal (x_2 - x_1) + \frac{\ell}{2}\|x_1 - x_2\|^2$ for all x_1 and x_2.

1 Some literature use weak convexity to mean the function is convex but not strictly convex.

Definition 4 (Polyak-Łojasiewicz (PL) Inequality)**.** *A function $g : \mathcal{X} \to \mathbb{R}$ satisfies μ-PL inequality if optimal value $g^* = \max_{x \in \mathcal{X}}$ is finite and for all $x \in \mathcal{X}$, we have*

$$\|\nabla g(x)\|^2 \geq 2\mu(g(x) - g^*), \forall x$$

It's worth noting that μ-strong convexity will imply μ-PL inequality. However, a function satisfying the PL inequality can be nonconvex. A simple nonconvex example is $g(x) = x^2 + 3\sin^2 x$. PL inequality is also observed in many nonconvex real-world scenarios, such as deep neural networks [Du et al., 2019], phase retrieval [Sun et al., 2018], and linear quadratic regulator (LQR) control [Fazel et al., 2018].

In this thesis, we will explore various settings of minimax optimization, each characterized by distinct assumptions on the objective function f with respect to x and y. Since we minimize over x and maximize over y, the condition of f with respect to x will be described by strong convex, convex, or nonconvex, while its condition with respect to y will be characterized as strongly concave, concave, or nonconcave. We denote these settings using a dash to connect the assumptions about x and y. The following settings, which will be frequently discussed throughout this thesis, provide a comprehensive overview of the different scenarios we will consider.

(μ_x, μ_y)-Strongly-Convex-Strongly-Concave (SC-SC) Setting: f is μ_x-strongly convex in x and μ_y-strongly concave in y.

μ-Strongly-Convex-Concave (SC-C) Setting: f is μ-strongly convex in x and concave in y.

Convex-Concave (C-C) Setting: f is convex in x and concave in y.

μ-Nonconvex-Strongly-Concave (NC-SC) Setting: f is possibly nonconvex in x and μ-strongly-concave in y.

Nonconvex-Concave (NC-C) Setting: f is nonconvex in x and concave in y.

Nonconvex-Nonconcave (NC-NC) Setting: f is possibly nonconvex in x and nonconcave in y.

1.3.2 *Global Optimality*

In minimax optimization, saddle points and minimax points are two frequently used global solutions. We will begin by defining two functions that will be referenced later in

this thesis. These are termed the primal and dual functions, though some literature may refer to them as envelope functions.

$$\Phi(x) = \max_{y \in \mathcal{Y}} f(x, y), \qquad \text{(Primal Function)}$$

$$\Psi(y) = \min_{x \in \mathcal{X}} f(x, y). \qquad \text{(Dual Function)}$$

Definition 5 (Saddle Point)**.** *A point (x^*, y^*) is a (global) saddle point if, for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$:*

$$f(x^*, y) \leq f(x^*, y^*) \leq f(x, y^*).$$

If, for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$,

$$f(x^*, y) - \epsilon \leq f(x^*, y^*) \leq f(x, y^*) + \epsilon,$$

then (x^*, y^*) is an approximate ϵ-saddle point.

The saddle point definition implies that x^* is the optimal solution for $\min_{x \in \mathcal{X}} f(x, y^*)$ and y^* is the optimal solution for $\max_{y \in \mathcal{Y}} f(x^*, y)$. This corresponds to the Nash equilibrium in a two-player zero-sum game. If the objective function f in Problem (1.1) is convex-concave, and if the domains $\mathcal{X}$ and $\mathcal{Y}$ are closed and convex sets with one of them being bounded, then a saddle point exists. In the convex-concave setting, first-order optimization methods can efficiently find an approximate saddle point in polynomial time [Nemirovski, 2004]. However, outside this setting, a saddle point might not exist, as illustrated by the simple example $\min_{x \in [0,1]} \max_{y \in [0,1]} (x - y)^2$.

Definition 6 (Minimax Point and Maximin Point)**.** *A point (x^*, y^*) is a (global) minimax point if, for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$:*

$$f(x^*, y) \leq f(x^*, y^*) \leq \max_{y' \in \mathcal{Y}} f(x, y').$$

A point (x^*, y^*) is a (global) maximin point if, for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$:

$$\min_{x' \in \mathcal{X}} f(x', y) \leq f(x^*, y^*) \leq f(x, y^*).$$

The minimax point is a mathematically intuitive solution to minimax problems of the form $\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y)$. It suggests that y^* is the optimal solution for $\max_{y \in \mathcal{Y}} f(x^*, y)$ and x^* is the optimal solution for $\min_{x \in \mathcal{X}} \Phi(x)$. This corresponds to the Stackelberg equilibrium in two-player games, where the x player (leader) acts first, and the y player (follower) acts second after observing the leader's move. In general, saddle points, minimax

points, and maximin points can differ significantly. For instance, consider the function $f(x,y) = \frac{x^4}{4} - \frac{x^2}{2} + xy$ [Zhang, 2021a]. The following theorem characterizes their relationship.

Theorem 1.3.1. *For any function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, the point (x^*, y^*) is a saddle point if and only if it is both a minimax point and a maximin point, and if and only if:*

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y) = \max_{y\in\mathcal{Y}} \min_{x\in\mathcal{X}} f(x,y),$$
$$x^* \in \operatorname*{Argmin}_{x\in\mathcal{X}} \Phi(x) \text{ and } y^* \in \operatorname*{Argmax}_{y\in\mathcal{Y}} \Psi(y).$$

In this thesis, when we explore settings within the convex-concave regime, including SC-SC, SC-C, and C-C settings, our objective is to identify ϵ-saddle points.

1.3.3 *Stationarity and Local Optimality*

For minimax problems outside the convex-concave framework, seeking a global solution is intractable. This is because even identifying a global solution for a nonconvex minimization problem is NP-hard [Pardalos and Vavasis, 1991]. Therefore, we introduce notions of stationary and local solutions.

To define an approximate stationary point, we can consider the gradient of the objective function $f(\cdot,\cdot)$ or the primal function $\Phi(\cdot)$. We use $\mathcal{P}_{\mathcal{C}}(\cdot)$ to denote the projection onto a closed convex set $\mathcal{C}$.

Definition 7 (Stationary point of $f(\cdot,\cdot)$)**.** *A point (x^*, y^*) is an (ϵ_1, ϵ_2)-stationary point of a differentiable function $f(\cdot,\cdot)$ if*

$$\ell \left\| x^* - \mathcal{P}_{\mathcal{X}} \left(x^* - \frac{1}{\ell} \nabla_x f(x^*, y^*) \right) \right\| \le \epsilon_1, \quad \ell \left\| y^* - \mathcal{P}_{\mathcal{Y}} \left(y^* + \frac{1}{\ell} \nabla_y f(x^*, y^*) \right) \right\| \le \epsilon_2.$$

For unconstrained problems, the definition simplifies to $\|\nabla_x f(x^*, y^*)\| \le \epsilon_1$ and $\|\nabla_y f(x^*, y^*)\| \le \epsilon_2$. We can also define the stationary point using the primal function Φ. However, Φ is not always differentiable. When f is strongly convex in y, Φ is differentiable and smooth [Lin et al., 2020a]. Otherwise, when $\mathcal{Y}$ is bounded, Φ is only guaranteed to be ℓ-weakly convex (not necessarily differentiable) [Thekumparampil et al., 2019]. We will use different definitions based on whether the function is differentiable.

Definition 8 (Stationary point of differentiable $\Phi(\cdot)$)**.** *When Φ is differentiable, a point x^* is an ϵ-stationary point of Φ if*

$$\ell \left\| x^* - \mathcal{P}_{\mathcal{X}} \left(x^* - \frac{1}{\ell} \nabla \Phi(x^*) \right) \right\| \leq \epsilon.$$

Definition 9 (Stationary point of weakly convex $\Phi(\cdot)$)**.** *For a weakly convex (and potentially non-differentiable) Φ, we first define the Moreau envelope as:*

$$\Phi_\lambda(z) \triangleq \min_{x \in \mathcal{X}} \Phi(x) + \frac{1}{2\lambda} \|x - z\|^2.$$

A point x^* is an ϵ-stationary point of Φ if

$$\left\| \nabla \Phi_{\frac{1}{2\ell}}(x^*) \right\| \leq \epsilon.$$

Definition 9 is meaningful because the condition implies the existence of some z near x^* with a small subgradient. Later in this thesis, we will focus on seeking stationary points as defined in Definitions 7 or 8 for the NC-SC setting, and stationary points as in Definition 9 for the NC-C setting.

Definition 10 (Local Saddle Point)**.** *A point (x^*, y^*) is a (ϵ, δ)-local saddle point if, for all $x \in \mathcal{X}$ with $\|x - x^*\| \leq \delta$ and for all $y \in \mathcal{Y}$ with $\|y - y^*\| \leq \delta$:*

$$f(x^*, y) - \epsilon \leq f(x^*, y^*) \leq f(x, y^*) + \epsilon.$$

While our primary focus is not on finding a local solution, we have provided a definition of a local saddle point above. In nonconvex minimization problems, local minimizers can be identified using first-order methods like stochastic gradient descent [Jin et al., 2017]. However, finding a local saddle point is particularly challenging for NC-NC minimax problems [Daskalakis et al., 2021]. We will discuss more about the difficulty in solving NC-NC problems in the subsequent subchapter.

1.4 KEY CHALLENGES

We will spotlight three predominant challenges in minimax optimization that remain inadequately addressed in current literature: asymmetry, nonconvexity-nonconcavity, and adaptivity. Each challenge encapsulates distinct research questions. Through this thesis, we endeavor to tackle these questions.

1.4.1 *Asymmetry*

In minimax problems, we categorize a setting as "balanced" if it maintains symmetrical assumptions regarding x and y, such as the (μ, μ)-SC-SC and C-C settings. Conversely, a setting is deemed "unbalanced" if it exhibits asymmetrical assumptions, including the (μ_x, μ_y)-SC-SC (with $\mu_x \neq \mu_y$), SC-C, NC-SC, and NC-C settings. For balanced settings, traditional algorithms like the extragradient (EG) can already achieve optimal complexities. Specifically, EG attains a complexity of $\mathcal{O}(\frac{\ell}{\mu}\log\frac{1}{\epsilon})$ in the (μ, μ)-SC-SC setting and $\mathcal{O}(\frac{\ell}{\epsilon})$ in the C-C setting. Both complexities align with the lower bounds, leaving no room for further enhancement. However, for unbalanced settings, these classic algorithms might not be optimal. For instance, in the (μ, μ)-SC-SC setting, EG achieves a complexity of $\mathcal{O}(\frac{\ell}{\min\{\mu_x,\mu_y\}}\log\frac{1}{\epsilon})$, whereas the lower bound stands at $\mathcal{O}(\frac{\ell}{\sqrt{\mu_x\mu_y}}\log\frac{1}{\epsilon})$ [Zhang et al., 2019b]. As we will explore in Chapter 2, certain unbalanced settings still present a gap between the upper and lower bounds. In other unbalanced scenarios, achieving optimal complexities requires intricate algorithms.

In scenarios where the problem presents finite-sum structures, variance-reduction algorithms can be employed. However, crafting such algorithms for unbalanced settings remains a formidable challenge. For the finite-sum (μ, μ)-SC-SC setting, optimal algorithms have already been developed, such as [Alacaoglu and Malitsky, 2022] and [Balamurugan and Bach, 2016]. Yet, in the SC-C setting, a variance-reduction algorithm is notably absent.

Existing algorithms tailored for unbalanced settings tend to be more intricate than those designed for balanced settings. Furthermore, distinct algorithms are often crafted for each unique setting, complicating their practical application. Our aspiration is to unify these algorithms across various settings, for both general and finite-sum structured minimax problems. The challenge lies in designing a universal framework that can accommodate these unbalanced minimax problems and achieve near-optimal guarantees across all settings.

1.4.2 *Nonconvexity-Nonconcavity*

Finding a meaningful solution for nonconvex-nonconcave (NC-NC) minimax optimization is recognized as a challenging task, a stark contrast to minimization optimization where methods such as gradient descent or stochastic gradient descent can effectively locate approximate stationary points or local minimizers. Daskalakis et al. [2021] show that seeking a (ϵ, δ)-local saddle point in a function that is G-Lipschitz and ℓ-smooth is PPAD-complete, with a polynomial-time Turing machine outputing approximate the values for the objective function f and its gradients. Furthermore, when relying on a first-order

oracle that returns the exact gradient, the discovery of a local saddle point necessitates a number of oracle queries that is exponential in at least one of the following parameters: $1/\epsilon$, ℓ, G, or d, where d represents the dimension of the domain. Hsieh et al. [2021] also show that many commonly used algorithms, such as gradient descent ascent (GDA) and extragradient (EG), will converge to a spurious set that does not include stationary points. This highlights the computational complexity and inherent challenges associated with NC-NC minimax optimization.

Given the inherent difficulty of general smooth NC-NC minimax problems, researchers have pivoted towards pinpointing sufficient conditions that guarantee convergence [Grimmer et al., 2020, Lu, 2021, Abernethy et al., 2021]. Notably, Nouiehed et al. [2019] introduced an efficient algorithm tailored for a subset of NC-NC minimax problems where the objective function $-f(x,\cdot)$ satisfies the Polyak-Łojasiewicz (PL) inequality [Polyak, 1963]. We refer to this setting as nonconvex-PL or NC-PL setting. Due to the ubiquity of the PL condition, the NC-PL setting captures many important applications, such as generative adversarial imitation learning of linear quadratic regulators [Cai et al., 2019].

While many of the applications we enumerate are formulated as NC-NC minimax problems, it remains imperative to discern specific structures that can guide us towards meaningful solutions. The true challenge resides in recognizing these structures and designing efficient algorithms to exploit them.

1.4.3 *Adaptivity*

Adaptive gradient methods, such as AdaGrad [Duchi et al., 2011] and Adam [Kingma and Ba, 2015], have emerged as the go-to optimization algorithms in numerous machine learning applications. Their popularity stems from their robustness to hyper-parameter selection and rapid empirical convergence. In minimax optimization, particularly in applications like generative adversarial networks [Goodfellow et al., 2014], these methods have seen widespread adoption. Often, they are integrated with popular minimax optimization algorithms like (stochastic) gradient descent ascent (GDA) as seen in works like [Gulrajani et al., 2017, Mishchenko et al., 2020, Reisizadeh et al., 2020]. Specifically, the two step-sizes, τ_x and τ_y, in GDA are determined adaptively according to some existing adaptive mechanism:

$$x_{t+1} = x_t - \tau_x \nabla_x f(x_t, y_t), \quad y_{t+1} = y_t + \tau_y \nabla_y f(x_t, y_t),$$

A standout benefit of adaptive step size schemes in minimization problems is their ability to converge without prior knowledge of problem-specific parameters, such as the smoothness constant. For intricate models like deep neural networks (DNNs), these

parameters are often elusive. For instance, traditional analysis for gradient descent in ℓ-smooth functions necessitates a step size smaller than $2/\ell$, where ℓ is the smoothness parameter. However, many adaptive schemes, which typically vary step sizes based on accumulated gradient information, can adapt to such parameters, achieving convergence without hyper-parameter tuning [Ward et al., 2020, Xie et al., 2020].

Yet, this parameter-agnostic property remains unproven in minimax optimization outside the convex-concave domain. Within the convex-concave regime, several adaptive algorithms, built upon EG and AdaGrad step sizes, retain this parameter-agnostic feature [Bach and Levy, 2019, Antonakopoulos et al., 2019]. However, when the objective function is nonconvex with respect to one variable, most existing adaptive algorithms necessitate knowledge of problem parameters [Huang and Huang, 2021, Huang et al., 2021, Guo et al., 2021a].

A pressing research question is whether a straightforward combination of GDA with adaptive schemes can yield a parameter-agnostic algorithm in minimax optimization. In Chapter 5, we provide a simple nonconvex-strongly-concave function:

$$f(x,y) = -\frac{1}{2}y^2 + Lxy - \frac{L^2}{2}x^2,$$

where $L > 0$ is a constant. Our findings indicate that directly employing adaptive step sizes, such as AdaGrad, Adam, and AMSGrad, results in non-convergence without hyper-parameter tuning. The challenge ahead is to devise parameter-agnostic adaptive algorithms, as current algorithms for nonconvex minimax optimization often come with multiple hyper-parameters, hampering their practical utility.

1.5 ROADMAP AND CONTRIBUTIONS

This thesis is structured to systematically address each of the three identified challenges in minimax optimization. In Chapter 2, we delve into the challenge posed by imbalance. Chapters 3 and 4 are dedicated to exploring the NC-NC regimes. Chapter 5 focuses on the challenge of adaptivity. To complement our discussions, Chapter 6 provides an analysis of the advantages of adaptive methods for the minimization problem. We summarize the contributions of each chapter as the following.

Chapter 2: A Catalyst Framework for Unbalanced Minimax Ploblems
This chapter is based on two papers [Yang et al., 2020b] and [Zhang et al., 2021b], as well as an unpublished note prior to 2021 . We delve into four significant unbalanced regimes in minimax optimization: (μ_x, μ_y)-SC-SC, SC-C, NC-SC, and NC-C. These regimes are considered in both the general form (1.1) and the finite-sum form. We introduce a Catalyst

framework, which is inspired by proximal point methods and the work of [Lin et al., 2017]. The framework operates by iteratively solving a subproblem: (x_{t+1}, y_{t+1}) is set as an approximate solution to the following:

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y) + \frac{\tau_x}{2}\|x-\bar{x}_t\|^2 - \frac{\tau_y}{2}\|y-z_t\|^2,$$

where $\bar{x}_t$ comes from the previous epoch and z_t is an extrapolated point from the previous y_t. By solving this subproblem using optimal existing algorithms for balanced regimes, such as Extragradient (EG) for the general form or variance-reduction methods [Alacaoglu and Malitsky, 2022] for the finite-sum form, we can achieve near-optimal or state-of-the-art complexity in these unbalanced settings.

Chapter 3: Global Convergence for PL-PL Minimax Problems
This chapter is based on [Yang et al., 2020a]. We focus on the global convergence of a specific class of NC-NC minimax problems. We introduce a class of minimax optimization problems that satisfy the "two-sided Polyak-Łojasiewicz (PL) condition", in which the objective function satisfies PL inequality in both variables, and establish the equivalence between three global convergence notions. We then analyze the convergence behavior of the Alternating Gradient Descent Ascent (AGDA) algorithm for this class of problems, showing that AGDA exhibits linear convergence to the global solution in the deterministic setting and sublinear convergence in the stochastic setting. Furthermore, we extend our analysis to the case where the objective function has a finite-sum structure, demonstrating that a variance reduction method can achieve linear convergence with a better dependence on the number of components.

Chapter 4: Single-Loop Algorithms for Nonconvex-PL Minimax Problems
This chapter is based on [Yang et al., 2022b]. We extend our focus to a broader class of NC-NC minimax problems, where the objective function satisfies the Polyak-Łojasiewicz (PL) inequality with respect to only one variable. We denote the condition number as $\kappa \triangleq \ell/\mu$, where μ is the PL modulus. We show that the Alternating Gradient Descent Ascent (AGDA) algorithm achieves a complexity of $\mathcal{O}(\kappa^2\epsilon^{-2})$ in deterministic settings and $\mathcal{O}(\kappa^4\epsilon^{-4})$ in stochastic settings without minibatch. Notably, this is the first demonstration of an $\mathcal{O}(\epsilon^{-4})$ complexity for GDA-type algorithms without minibatch or additional assumptions, even in the more stringent NC-SC setting. Furthermore, we prove another single-loop algorithm, Smoothed AGDA, achieves a complexity of $\mathcal{O}(\kappa\epsilon^{-2})$ in deterministic settings and $\mathcal{O}(\kappa^2\epsilon^{-4})$ in stochastic settings. These results represent the best-known complexity for single-loop algorithms even under the stronger assumption of NC-SC.

Chapter 5: Parameter-Agnostic Nonconvex Minimax Optimization

This chapter is based on [Yang et al., 2022a]. We delve into the NC-SC setting. We begin by demonstrating, through an example, that a straightforward combination of Gradient Descent Ascent (GDA) with adaptive schemes, a common heuristic, fails to converge without hyper-parameter tuning. To address this, we propose a nested adaptive framework, NeAda, which consists of an inner loop that adaptively maximizes the y variable and an outer loop that adaptively minimizes the x variable. When equipped with AdaGrad [Duchi et al., 2011] step sizes, NeAda achieves an $\mathcal{O}(\epsilon^{-2})$ complexity in deterministic settings and an $\mathcal{O}(\epsilon^{-4})$ complexity in stochastic settings, all without the need for prior knowledge of problem parameters. Notably, we provide one of the first theoretical analyses of AdaGrad without the bounded gradient assumptions in the stochastic setting. Prior to this, this is not established, under the classic noise assumption of bounded variance, even for minimization problems.

Chapter 6: Limit of Untuned SGD and Power of Adaptive Methods
This chapter is based on [Yang et al., 2023]. we investigate the advantages of adaptive methods over Stochastic Gradient Descent (SGD) in nonconvex minimization problems. We first demonstrate that while SGD with any polynomially decreasing step size can achieve an order-optimal convergence rate for minimizing smooth objectives, it is hindered by an unavoidable exponential dependence on the smoothness constant. We then scrutinize three widely-used families of adaptive methods: Normalized SGD, AMSGrad, and AdaGrad. We establish that these methods can circumvent the exponential dependency in deterministic or stochastic settings. However, we also uncover some unexpected limitations of Normalized SGD and AMSGrad in stochastic settings. Specifically, we find that Normalized SGD fails to converge, and AMSGrad, without the bounded stochastic gradient assumption, can converge at an arbitrarily slow rate.

2

A CATALYST FRAMEWORK FOR UNBALANCED MINIMAX PLOBLEMS

In this chapter, we present a versatile framework designed for a range of balanced smooth minimax optimization problems. These problems are characterized by an objective function that exhibits asymmetric convexity and concavity properties with respect to the primal and dual variables. Our framework applies the (accelerated) proximal point method to the associated primal or dual problem of the original minimax problem. This results in a sequence of harmoniously balanced, strongly-convex-strongly-concave subproblems, which are readily addressed using established gradient-based algorithms. This single cohesive framework is apt for all the unbalanced scenarios we delve into, encompassing both the general and finite-sum forms. Despite its simplicity, it gives rise to a suite of algorithms that achieve either near-optimal or state-of-the-art complexities.

2.1 OVERVIEW

We focus on the minimax optimization problem in its general form:

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y), \tag{general form}$$

where the function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is smooth (i.e., gradient Lipschitz), $\mathcal{X}$ is a convex set in $\mathbb{R}^m$, and $\mathcal{Y}$ is a convex set in $\mathbb{R}^n$. In many machine learning applications, f possesses a finite sum structure, where each component is associated with a loss from a single observation, so we are also interested in the following form:

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

A vast array of first-order algorithms for minimax optimization can be found in the literature, ranging from the classical projection method [Sibony, 1970], Korpelevich's extra-gradient method [Korpelevich, 1976], to many recent hybrid or randomized algorithms, e.g., [Monteiro and Svaiter, 2010, He et al., 2015, Kong and Monteiro, 2019]. However, most of these existing works are limited to the following settings (i) the *well-balanced* strongly-convex-strongly-concave setting (e.g., [Tseng, 1995, Mokhtari et al., 2019]), where the x-component and y-component share the same strong-convexity/concavity constant

μ, (ii) the general convex-concave setting (e.g., [Nemirovski, 2004, Nesterov, 2007]), and (iii) the special bilinear convex-concave setting (e.g., [Chambolle and Pock, 2016, Chen et al., 2014]. The lower complexity bounds for these three settings established in [Zhang et al., 2019b], [Nemirovsky and Yudin, 1983], [Ouyang and Xu, 2019], respectively, are already attained by some existing algorithms. For example, extragradient (EG) achieves the optimal $\mathcal{O}(1/\epsilon)$ complexity for smooth convex-concave minimax problems, and the optimal $\mathcal{O}(\kappa \log(1/\epsilon))$ complexity for well-balanced strongly-convex-strongly-concave minimax problems [Zhang et al., 2019b], where κ is the condition number.

Yet, results for configurations outside these settings are sparse. We are particularly interested in the following unbalanced settings: (1) unbalanced strongly-convex-strongly-concave (SC-SC) objectives with strong-convexity constant μ_x different from strong-concavity constant μ_y, (2) strongly-convex-concave (SC-C) objectives, (3) nonconvex-strongly-concave (NC-SC) objectives, (4) nonconvex-concave (NC-C) objectives.

In recent years, there has been a growing interest in addressing problems within unbalanced regimes. For instance, [Thekumparampil et al., 2019] introduced the dual implicit accelerated gradient algorithm (DIAG) specifically for the SC-C setting and its proximal variant (Prox-DIAG) for the NC-C setting; Luo et al. [2020] developed a variance reduction method for the NC-SC setting by integrating SARAH [Nguyen et al., 2017] into minimax optimization. However, these algorithms are considerably more complex than those crafted for balanced regimes, usually incorporating multiple acceleration procedures and necessitating several loops. Moreover, as they are tailored for a specific setting, and it is difficult to extend them to other unbalanced settings.

More recently, [Lin et al., 2020b] pioneered the development of near-optimal algorithms catering to all the aforementioned unbalanced settings. Unfortunately, they still introduce extra logarithmic terms in their complexities relative to the lower bounds in certain settings. Additionally, while they cater to the general form of the problem, the integration of these advanced algorithms with variance-reduction techniques remains ambiguous when the problem adopts a finite-sum form. A notable gap in the current literature is the absence of a dedicated variance reduction approach for the SC-C setting.

This raises the question:

> *Can we simply leverage the rich off-the-shelf methods designed for well-balanced strongly-convex-strongly-concave minimax problems to design a universal framework for all unbalanced settings in both general and finite-sum forms?*

Inspired by the success of the Catalyst framework that uses gradient-based algorithms originally designed for strongly convex minimization problems to minimize convex/nonconvex objectives [Lin et al., 2017, Paquette et al., 2017], we introduce a generic Catalyst framework for minimax optimization. Rooted in an inexact accelerated proximal

point framework, the idea is to repeatedly solve the following auxiliary strongly-convex-strongly-concave problems using an existing method $\mathcal{M}$:

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y) + \tfrac{\tau_x}{2}\|x-\bar{x}_t\|^2 - \tfrac{\tau_y}{2}\|y-z_t\|^2.$$

At first glance, the design of this algorithm might seem intuitive. However, the selection of appropriate proximal parameters τ_x and τ_y, the prox centers $\bar{x}_t, z_t$, and the method $\mathcal{M}$ for solving the auxiliary problems, plays a crucial role and significantly influences the overall complexity. In this chapter, we provide a comprehensive guide on how to make these selections.

With this generic Catalyst framework, we derive a series of compelling results. We encapsulate our findings for the general form of the problem in Table 2.1 and for the finite-sum form in Table 2.2. Notably, Catalyst combined with specific subroutine $\mathcal{M}$ either matches with the lower complexities or improves over the previous known results.

2.1.1 *Related Work.*

Catalyst and Proximal Point Methods. Catalyst was first introduced by [Lin et al., 2015] for minimization of convex and strongly-convex objectives, and it is further generalized to nonconvex objectives by [Paquette et al., 2017]. Catalyst can be considered as a variant of accelerated proximal point algorithm. The acceleration of the proximal point algorithm [Rockafellar, 1976b,a, Güler, 1991] was first discussed in [Güler, 1992]. Several other works, such as [Shalev-Shwartz and Zhang, 2014, He and Yuan, 2012, Salzo and Villa, 2012], also explore inexact accelerated proximal point algorithm under different settings in minimization optimization. Although [Rockafellar, 1976b] discussed proximal point algorithm for monotone operators including minimax optimization, before our work it remains mysterious (i) how to apply the acceleration scheme to minimax optimization and attain primal-dual gap convergence; (2) how to design practical notion of exactness for minimax auxiliary problems.

SC-SC Setting. Classic algorithms like EG and Optimistic Gradient Descent Ascent (OGDA) have demonstrated linear convergence with a complexity of $\mathcal{O}\left(\left(\frac{\ell}{\mu_x}+\frac{\ell}{\mu_y}\right)\log\left(\frac{1}{\epsilon}\right)\right)$ [Mokhtari et al., 2019, Azizian et al., 2019, Tseng, 1995, Gidel et al., 2018]. This complexity is optimal specifically when $\mu_x = \mu_y$. Later, [Lin et al., 2020b] introduced MINIMAX-APPA, which improved the condition number dependency but had a less favorable ϵ dependency. More recently, [Wang and Li, 2020] further refined this to $\mathcal{O}\left(\frac{\ell}{\sqrt{\mu_x\mu_y}}\log\left(\frac{1}{\epsilon}\right)\right)$, aligning with the lower bound by [Zhang et al., 2019b]. Additionally, there are several variance-reduced algorithms for strongly-convex-strongly-concave objectives. For instance, [Balamurugan

Settings	Algorithms/Lower Bound	Oracle Complexity
SC-SC	Extragradient [Tseng, 1995]	$\mathcal{O}\left(\ell/\min\{\mu_x,\mu_y\}\log(1/\epsilon)\right)$
	APPA-ABR [Wang and Li, 2020]	$\tilde{\mathcal{O}}\left(\ell/\sqrt{\mu_x\mu_y}\log(1/\epsilon)\right)$
	MINIMAX-APPA [Lin et al., 2020b]	$\tilde{\mathcal{O}}\left(\ell/\sqrt{\mu_x\mu_y}\log^3(1/\epsilon)\right)$
	Catalyst-EG (this work)	$\tilde{\mathcal{O}}\left(\ell/\sqrt{\mu_x\mu_y}\log(1/\epsilon)\right)$
	Lower bound [Zhang et al., 2019b]	$\Omega\left(\ell/\sqrt{\mu_x\mu_y}\log(1/\epsilon)\right)$
SC-C	DIAG [Thekumparampil et al., 2019]	$\tilde{\mathcal{O}}\left(\ell^{3/2}\mathcal{D}_{\mathcal{Y}}/(\mu\sqrt{\epsilon})\log^2(1/\epsilon)\right)$
	APPA-ABR [Wang and Li, 2020]	$\tilde{\mathcal{O}}\left(\ell\mathcal{D}_{\mathcal{Y}}/\sqrt{\mu\epsilon}\log(1/\epsilon)\right)$
	MINIMAX-APPA [Lin et al., 2020b]	$\tilde{\mathcal{O}}\left(\ell\mathcal{D}_{\mathcal{Y}}/\sqrt{\mu\epsilon}\log^3(1/\epsilon)\right)$
	Catalyst-EG (this work)	$\tilde{\mathcal{O}}\left(\ell\mathcal{D}_{\mathcal{Y}}/\sqrt{\mu\epsilon}\log(1/\epsilon)\right)$
	Lower bound [Ouyang and Xu, 2019]	$\Omega\left(\ell\mathcal{D}_{\mathcal{Y}}/\sqrt{\mu\epsilon}\right)$
NC-SC	GDA/AGDA [Lin et al., 2020a, Yang et al., 2020a]	$\mathcal{O}(\ell^3/\mu^2\epsilon^{-2})$
	MINIMAX-PPA [Lin et al., 2020b]	$\tilde{\mathcal{O}}\left(\ell^{3/2}/\mu^{1/2}\epsilon^{-2}\log^2(1/\epsilon)\right)$
	Catalyst-EG (this work)	$\tilde{\mathcal{O}}\left(\ell^{3/2}/\mu^{1/2}\epsilon^{-2}\right)$
	Lower bound	$\Omega\left(\ell^{3/2}/\mu^{1/2}\epsilon^{-2}\right)$
NC-C	Prox-DIAG [Thekumparampil et al., 2019]	$\tilde{\mathcal{O}}(\ell^2\mathcal{D}_{\mathcal{Y}}\epsilon^{-3}\log^2(1/\epsilon))$
	FNE-search [Ostrovskii et al., 2020]	$\tilde{\mathcal{O}}(\ell^2\mathcal{D}_{\mathcal{Y}}\epsilon^{-3}\log^2(1/\epsilon))$
	Catalyst-EG (this work)	$\tilde{\mathcal{O}}(\ell^2\mathcal{D}_{\mathcal{Y}}\epsilon^{-3}\log(1/\epsilon))$
	Lower bound	?

TABLE 2.1: The table summarizes results for general form minimax problems. The objective function is ℓ-smooth. It presents the oracle complexities to find an ϵ-saddle point for (μ_x,μ_y)-SC-SC and μ-SC-C settings, and an ϵ-stationary point of the primal function for μ-NC-SC and NC-C settings. Here $\mathcal{D}_{\mathcal{Y}}^2$ denotes the diameter of $\mathcal{Y}$. We assume $\epsilon \le \mu\mathcal{D}_{\mathcal{Y}}^2$ in SC-C setting and $\epsilon \le \ell\mathcal{D}_{\mathcal{Y}}^2$ in NC-C setting. $\tilde{\mathcal{O}}$ only hides logarithmic factors in problem parameters, but not ϵ^{-1}.

Settings	Algorithms/Lower Bound	Oracle Complexity
SC-SC	SVRG [Balamurugan and Bach, 2016]	$\mathcal{O}\left(n+\left(\frac{\ell}{\min\{\mu_x,\mu_y\}}\right)^2\log\frac{1}{\epsilon}\right)$
	Acc-SVRG [Balamurugan and Bach, 2016]	$\tilde{\mathcal{O}}\left(n+\frac{\sqrt{n}\ell}{\min\{\mu_x,\mu_y\}}\log\frac{1}{\epsilon}\right)$
	Catalyst-Acc-SVRG (this work)	$\tilde{\mathcal{O}}\left(\left(n+\frac{\sqrt{n}\ell}{\sqrt{\mu_x\mu_y}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}}{\sqrt{\mu_y}}\right)\log\frac{1}{\epsilon}\right)$
	Lower bound [Han et al., 2021]	$\Omega\left(\left(n+\frac{\sqrt{n}\ell}{\sqrt{\mu_x\mu_y}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}}{\sqrt{\mu_y}}\right)\log\frac{1}{\epsilon}\right)$
SC-C	Catalyst-Acc-SVRG (this work)	$\tilde{\mathcal{O}}\left(\left(n+\frac{\sqrt{n}\ell\mathcal{D}_\mathcal{Y}}{\sqrt{\mu\epsilon}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}\mathcal{D}_\mathcal{Y}}{\sqrt{\epsilon}}\right)\log\frac{1}{\epsilon}\right)$
	Lower bound [Han et al., 2021]	$\Omega\left(n+\frac{\sqrt{n}\ell\mathcal{D}_\mathcal{Y}}{\sqrt{\mu\epsilon}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}\mathcal{D}_\mathcal{Y}}{\sqrt{\epsilon}}\right)$
NC-SC	SREDA [Luo et al., 2020]	$\begin{cases}\tilde{\mathcal{O}}\left(n+\frac{\sqrt{n}\ell^3}{\mu_y^2\epsilon^2}\right), & n\geq\frac{\ell^2}{\mu^2}\\ \mathcal{O}\left(\left(\frac{n\ell}{\mu}+\frac{\ell^2}{\mu_y^2}\right)\frac{\ell}{\epsilon^2}\right), & n\leq\frac{\ell^2}{\mu^2}\end{cases}$
	Catalyst-SVRG (this work)	$\tilde{\mathcal{O}}\left(\left(n+\frac{n^{3/4}\sqrt{\ell}}{\sqrt{\mu}}\right)\frac{\ell}{\epsilon^2}\right)$
	Lower bound [Zhang et al., 2021b]	$\Omega\left(n+\frac{\sqrt{n}\ell^{\frac{3}{2}}}{\sqrt{\mu\epsilon^2}}\right)$
NC-C	PG-SVRG [Rafique et al., 2022]	$\tilde{\mathcal{O}}\left(\left(\frac{n}{\epsilon^2}+\epsilon^{-6}\right)\log\frac{1}{\epsilon}\right)$
	Catalyst-SVRG (this work)	$\tilde{\mathcal{O}}\left(\left(\frac{n^{\frac{3}{4}}\ell^2\mathcal{D}_\mathcal{Y}}{\epsilon^3}+\frac{n\ell}{\epsilon^2}\right)\log\frac{1}{\epsilon}\right)$
	Lower bound	?

TABLE 2.2: The table summarizes results for finite-sum form minimax problems. The objective function f is ℓ-averaged-smooth (Assumption 4). It presents the oracle complexities to find an ϵ-saddle point for (μ_x,μ_y)-SC-SC and μ-SC-C settings, and an ϵ-stationary point of the primal function for μ-NC-SC and NC-C settings. Here $\mathcal{D}_\mathcal{Y}^2$ denotes the diameter of $\mathcal{Y}$. $\tilde{\mathcal{O}}$ only hides logarithmic factors in problem parameters, but not ϵ^{-1}. The dependence on problem parameters is not explicit in [Rafique et al., 2022].

and Bach, 2016] adapted SVRG and SAGA for minimax optimization and provided an accelerated variant with a complexity of $\mathcal{O}\left(n+\sqrt{n}\left(\frac{\ell}{\mu_x}+\frac{\ell}{\mu_y}\right)\right)\log\left(\frac{1}{\epsilon}\right)$, achieving an optimal result in balanced settings, as confirmed by [Han et al., 2021]. A similar result was reported by [Alacaoglu and Malitsky, 2022].

SC-C Setting. Several works have achieved a complexity of $\mathcal{O}\left(\frac{\ell}{\sqrt{\mu\epsilon}}\right)$ for problems where the coupling term is linear in both variables [Nesterov, 2005, Chambolle and Pock, 2016, Xie and Shi, 2019], or just in one variable [Hamedani and Aybat, 2018, Juditsky et al., 2011]. For general problems, both [Lin et al., 2020b] and [Wang and Li, 2020] transformed a general SC-C problem into an SC-SC problem by adding a term $\mathcal{O}(\epsilon)\|y\|^2$. While these methods yield complexities that closely align with the lower bound $\Omega\left(\frac{\ell}{\sqrt{\mu\epsilon}}\right)$ up to logarithmic terms, they require the target accuracy ϵ to be predetermined. In contrast, [Thekumparampil et al., 2019] merged accelerated gradient descent with Mirror Prox to achieve near-optimal complexity without the need for a prefixed target accuracy. Notably, before our contribution in [Yang et al., 2020b], no variance reduction method had been proposed for this setting. A subsequent work by [Han et al., 2021] provided a nearly matching lower bound.

NC-SC Setting. Basic algorithms like simultaneous GDA [Lin et al., 2020a] and alternating GDA [Yang et al., 2020a, Boţ and Böhm, 2020, Xu et al., 2020c] have been shown to achieve a complexity of $\mathcal{O}\left(\frac{\ell^3}{\mu^2\epsilon^2}\right)$ when seeking an ϵ-stationary point for the primal function. The MINIMAX-PPA, introduced by [Lin et al., 2020b], employs the proximal point algorithm on the primal function and addresses the auxiliary problems using accelerated gradient ascent. This approach aligns with the lower bound $\Omega\left(\frac{\ell^{3/2}}{\mu^{1/2}\epsilon^2}\right)$ up to logarithmic factors. Furthermore, [Luo et al., 2020] presented a variance-reduced algorithm named SREDA, for stochastic NC-SC problems. However, its complexity does not fully align with the lower bound for finite-sum form problems, particularly in its dependence on the condition number, as highlighted by [Zhang et al., 2021b].

NC-C Setting. The prevailing best complexity for identifying an approximate stationary point of the primal function stands at $\tilde{\mathcal{O}}(\ell^2\epsilon^{-3})$ up to polynomial terms [Lin et al., 2020b, Thekumparampil et al., 2019, Zhao, 2020, Ostrovskii et al., 2020]. Notably, both [Ostrovskii et al., 2020] and [Lin et al., 2020b] suggest analogous algorithms that employ an inexact accelerated method to address auxiliary problems derived from smoothed proximal steps. Several other algorithms, such as those by [Nouiehed et al., 2019, Lu et al., 2020, Kong and Monteiro, 2019], aim to achieve the stationarity of $f(\cdot,\cdot)$. Additionally, [Rafique et al., 2022] introduced a variance-reduced algorithm for finite-sum objectives with a complexity of $\widetilde{\mathcal{O}}(n\epsilon^{-2}+\epsilon^{-6})$. More recently, [Tran-Dinh et al., 2020] unveiled a hybrid variance-reduced algorithm for stochastic nonconvex-concave minimax challenges with a coupling term linear in y, achieving a complexity of $\mathcal{O}(\epsilon^{-5})$.

NOTATIONS. Throughout the chapter, $\|\cdot\|$ stands for the standard ℓ_2-norm. For non-negative functions f and g, we say $f = \mathcal{O}(g)$ if $f(x) \leq cg(x)$ for some $c > 0$. We use $\tilde{\mathcal{O}}$ to hide logarithmic factors of problem parameters and the initial point, but not that of ϵ^{-1}.

2.2 STRONGLY-CONVEX-(STRONGLY)-CONCAVE MINIMAX OPTIMIZATION

In this subchapter, we focus on solving strongly-convex-strongly-concave (SC-SC) and strongly-convex-concave (SC-C) minimax problems and introduce a general Catalyst scheme. We formally make the following assumptions.

Assumption 2 (SC-SC/SC-C). *$f(\cdot, y)$ is μ_x-strongly-convex for any y in $\mathcal{Y}$ with $\mu_x > 0$, i.e.,*

$$f(x_1, y) \geq f(x_2, y) + \nabla_x f(x_2, y)^T (x_1 - x_2) + \frac{\mu_x}{2}\|x_1 - x_2\|^2, \quad \forall x_1, x_2 \in \mathcal{X}.$$

$f(x, \cdot)$ is μ_y-strongly-concave for any x in $\mathcal{X}$ with $\mu_y \geq 0$, i.e.,

$$f(x, y_1) \geq f(x, y_2) + \nabla_y f(x, y_2)^T (y_1 - y_2) + \frac{\mu_y}{2}\|y_1 - y_2\|^2, \quad \forall y_1, y_2 \in \mathcal{Y}.$$

Without loss of generality, we assume $\mu_x \geq \mu_y$. $\mathcal{X}$ and $\mathcal{Y}$ are convex and closed sets, and we further assume $\mathcal{Y}$ to be bounded with diameter $\mathcal{D}_\mathcal{Y} \triangleq \max_{y,y' \in \mathcal{Y}} \|y - y'\|$ when $\mu_y = 0$.

In the definition above we allow the strong concavity module about y to 0. When $\mu_x > 0$ and $\mu_y > 0$, we refer to such f as (μ_x, μ_y)-SC-SC; when $\mu_x > 0$ and $\mu_y = 0$, we refer to such f as μ_x-SC-C. We further assume the objective function f is ℓ-smooth or ℓ-averaged smooth defined as follows.

Assumption 3 (Lipschitz gradient). *There exists a positive constant ℓ such that*

$$\max\{\left\|\nabla_y f\left(x_1, y_1\right) - \nabla_y f\left(x_2, y_2\right)\right\|, \left\|\nabla_x f\left(x_1, y_1\right) - \nabla_x f\left(x_2, y_2\right)\right\|\} \leq \ell[\|x_1 - x_2\| + \|y_1 - y_2\|],$$

holds for all $x_1, x_2 \in \mathcal{X}$ and $y_1, y_2 \in \mathcal{Y}$. We call such f ℓ-smooth (ℓ-S).

Assumption 4 (Averaged Lipschitz gradient). *For $f(x, y) = \frac{1}{n}\sum_{i=1}^n f_i(x, y)$, we assume each f_i is differentiable and there exists a positive constant ℓ such that*

$$\frac{1}{n}\sum_{i=1}^n \|\nabla f_i(x_1, y_1) - \nabla f_i(x_2, y_2)\|^2 \leq \ell^2 \left(\|x_1 - x_2\|^2 + \|y_1 - y_2\|^2\right). \tag{2.1}$$

holds for all $x_1, x_2 \in \mathcal{X}$ and $y_1, y_2 \in \mathcal{Y}$. We call such f ℓ-averaged-smooth (ℓ-AS).

Remark 2.2.1. *If $f(x, y) = \frac{1}{n}\sum_{i=1}^n f_i(x, y)$ is ℓ-averaged-smooth, then it is ℓ-smooth and $f(x, y) + \frac{\tau_x}{2}\|x - \tilde{x}\|^2 - \frac{\tau_y}{2}\|y - \tilde{y}\|^2$ is $\sqrt{2}(\ell + \max\{\tau_x, \tau_y\})$-AS for any $\tilde{x}$ and $\tilde{y}$.*

Now we introduce the definitions of primal and dual functions, and the gap function associated with objective function f.

Definition 11. *For a function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, we define the primal function $\Phi : \mathcal{X} \to \mathbb{R}$ as*

$$\Phi(x) = \max_{y \in \mathcal{Y}} f(x, y),$$

and we define the dual function $\Psi : \mathcal{Y} \to \mathbb{R}$ as

$$\Psi(y) = \min_{x \in \mathcal{X}} f(x, y).$$

The primal-dual gap function $\text{gap}_f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is defined as

$$\text{gap}_f(x, y) \triangleq \max_{y' \in \mathcal{Y}} f(x, y') - \min_{x' \in \mathcal{X}} f(x', y) \leq \epsilon.$$

In the SC-SC and SC-C settings, our goal is to find an ϵ-saddle point (x, y) such that $\text{gap}_f(x, y) \leq \epsilon$. If $\text{gap}_f(x^*, y^*) = 0$, then (x^*, y^*) is a saddle point, i.e., $f(x^*, y) \leq f(x^*, y^*) \leq f(x, y^*)$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Observe that $\text{gap}_f(x, y) = \Phi(x) + \Psi(y)$ in these settings, so a small primal-dual gap at a point (x, y) is equivalent to optimality gaps of both primal and dual functions being small.

2.2.1 *A Catalyst Framework*

We present a generic Catalyst scheme in Algorithm 1. Analogous to its prototype [Lin et al., 2017, Paquette et al., 2017], this scheme consists of three important components: an inexact accelerated proximal point step as the wrapper, a linearly-convergent first-order method $\mathcal{M}$ as the workhorse, as well as carefully chosen parameters and stopping criteria.

INEXACT ACCELERATED PROXIMAL POINT STEP. The main idea is to repeatedly solve a series of regularized problems by adding a quadratic term in y to the original problem:

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \left[\tilde{f}_t(x, y) := f(x, y) - \frac{\tau}{2} \|y - z_t\|^2 \right], \tag{$\star$}$$

where $\tau > 0$ is a regularization parameter (to be specified later) and z_t is the prox-center. The prox-centers $\{z_t\}_t$ are built on extrapolation steps of [Nesterov, 2013]. Noticeably, this step can also be viewed as applying the original Catalyst scheme [Lin et al., 2017] to the dual function $\Psi(y) \triangleq \min_{x \in \mathcal{X}} f(x, y)$. The major distinction is that we do not have access to the closed-form dual function or its gradient, which does not allow us to solve the

Algorithm 1 `Catalyst for SC-C/SC-SC Minimax Optimization`

Input: initial point (x_0, y_0), parameter $\tau > 0$
Initialization: $q = \frac{\mu_y}{\mu_y + \tau}$, $v_0 = y_0$; $\alpha_1 = 1$ if $\mu_y = 0$ and $\alpha_1 = \sqrt{q}$ if $\mu_y > 0$.
for all $t = 1, 2, ..., T$ **do**
  Set $z_t = \eta_t v_{t-1} + (1 - \eta_t) y_{t-1}$ with $\eta_t = \frac{\alpha_t - q}{1 - q}$
  Find an inexact solution (x_t, y_t) to the following problem with Algorithm $\mathcal{M}$

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \left[\tilde{f}_t(x, y) \triangleq f(x, y) - \ell \|y - z_t\|^2 \right] \tag{$\star$}$$

  such that

$$\text{if } \mu_y = 0 : f(x_t, y_t) - \min_{x \in \mathcal{X}} f(x, y_t) \leq \epsilon_t \text{ and } \nabla_y \tilde{f}_t(x_t, y_t)^T (y - y_t) \leq \epsilon_t, \forall y \in \mathcal{Y}; \tag{2.2}$$

$$\text{if } \mu_y > 0 : \text{gap}_{\tilde{f}_t}(x_t, y_t) \leq \epsilon_t. \tag{2.3}$$

  $v_t = y_{t-1} + \frac{1}{\alpha_t}(y_t - y_{t-1})$;
  Choose $\alpha_{t+1} \in [0, 1]$ such that $\alpha_{t+1}^2 = (1 - \alpha_{t+1})\alpha_t^2 + q\alpha_{t+1}$.
end for
Output: if $\mu_y = 0$: $(\bar{x}_T, y_T)$ with $\bar{x}_T = \sum_{t=1}^T \frac{1/\alpha_t}{\sum_{m=1}^T 1/\alpha_m} x_t$; if $\mu_y > 0$: (x_T, y_T).

auxiliary problem ($\star$) with minimization algorithms and causes difficulty in measuring the inexactness during solving auxiliary problems. Moreover, we should guarantee the solution quality in terms of the primal-dual gap instead of just dual optimality.

LINEARLY-CONVERGENT ALGORITHM $\mathcal{M}$. By construction, the series of auxiliary problems ($\star$) are $(\mu_x, \mu_y + \tau)$-SC-SC, and $(\ell + \tau)$-S or $\sqrt{2}(\ell + \tau)$-AS if f is ℓ-S or ℓ-AS, respectively. Consequently, they can be solved at a linear convergence rate by a broad range of first-order algorithms documented in existing literature. Let $\mathcal{M}$ present any algorithm that can solve the $(\mu_x, \mu_y + \tau)$-SC-SC auxiliary problem at a linear convergence rate such that after N iterations:

$$\|x_N - x^*\|^2 + \|y_N - y^*\|^2 \leq \left(1 - \frac{1}{\Lambda_{\mathcal{M},\tau}}\right)^N [\|x_0 - x^*\|^2 + \|y_0 - y^*\|^2], \tag{2.4}$$

if $\mathcal{M}$ is a deterministic algorithm; or taking expectation to the left-hand side above if $\mathcal{M}$ is randomized. The choices for $\mathcal{M}$ include, but are not limited to, extragradient (EG) [Tseng, 1995], optimistic gradient descent ascent (OGDA) [Gidel et al., 2018], SVRG [Palaniappan and Bach, 2016], SPD1-VR [Tan et al., 2018], Point-SAGA [Luo et al., 2019], and variance reduced prox-method [Carmon et al., 2019]. Here $\Lambda_{\mathcal{M},\tau}$ depends on τ and algorithm $\mathcal{M}$. For example, in the case of EG or OGDA, we have $\Lambda_{\mathcal{M},\tau} = \frac{\ell + \tau}{4 \min\{\mu_x, \mu_y + \tau\}}$ [Gidel et al., 2018, Azizian et al., 2019]. When using SVRG or SAGA, assuming f is ℓ-AS, we have

$\Lambda_{\mathcal{M},\tau} \propto n + (\frac{\ell+\tau}{\min\{\mu_x,\mu_y+\tau\}})^2$ [Balamurugan and Bach, 2016]. We will give more choices of $\mathcal{M}$ later.

STOPPING CRITERIA. To ensure the overall convergence in terms of the primal-dual gap, it is important to approximately solve the auxiliary problem ($\star$) with a reasonable degree of accuracy, ensuring the pair (x_t, y_t) converges properly. For a more general approach, we utilize the criterion specified in (2.2) and (2.3) in our scheme. Most existing minimax optimization algorithms can meet this stopping criterion after a sufficient number of iterations. However, it could still be hard to check in practice because $\min_{x\in\mathcal{X}} f(x, y_t)$, $\max_{y\in\mathcal{Y}} \nabla_y \tilde{f}_t(x_t, y_t)^T(y - y_t)$ and $\text{gap}_{\tilde{f}_t}(x_t, y_t)$ are not always computable. One approach is to predetermine the number of iterations for running the algorithm $\mathcal{M}$ based on its complexity in solving ($\star$), although this will vary depending on the specific algorithm used. Alternatively, the following lemma allows us to transform these conditions into ones that are verifiable, with the minor cost of a full gradient evaluation and a projection step.

Lemma 2.2.2. *Consider a function $\tilde{f}(x, y)$ that is (μ_1, μ_2)-SC-SC and has $\tilde{\ell}$-Lipschitz gradient on $\mathcal{X} \times \mathcal{Y}$. Let $z^* = (x^*, y^*)$ be the saddle point. For any point $z = (x, y)$ in $\mathcal{X} \times \mathcal{Y}$, we define $[z]_\beta = ([x]_\beta, [y]_\beta)$ with $\beta > 2\tilde{\ell}$ to be the point after one step of projected gradient descent ascent:*

$$[x]_\beta = \mathcal{P}_\mathcal{X}\left(x - \tfrac{1}{\beta}\nabla_x \tilde{f}(x, y)\right), \quad [y]_\beta = \mathcal{P}_\mathcal{Y}\left(y + \tfrac{1}{\beta}\nabla_y \tilde{f}(x, y)\right),$$

then we have

1. $\text{gap}_{\tilde{f}}([z]_\beta) \le A\|z - z^*\|^2, \quad \nabla\tilde{f}([x]_\beta, [y]_\beta)^T(\bar{y} - [y]_\beta) \le A\|z - z^*\|^2 + 2\beta\mathcal{D}_\mathcal{Y}\|z - z^*\|;$
2. $\|z - z^*\| \le \frac{\beta+\tilde{\ell}}{\tilde{\mu}}\|z - [z]_\beta\|, \quad \|z - [z]_\beta\|^2 \le \frac{2}{(1-\tilde{\ell}/\beta)^3}\|z - z^*\|^2, \quad \frac{\tilde{\mu}}{2}\|z - z^*\| \le \text{gap}_{\tilde{f}}(x, y),$

where $A = \beta + \frac{2\beta\tilde{\ell}^2}{\tilde{\mu}^2} + \frac{4\beta\tilde{\ell}^2}{\tilde{\mu}^2(1-\tilde{\ell}/\beta)^3}$ and $\tilde{\mu} = \min\{\mu_1, \mu_2\}$

Based on this lemma, we can replace (2.2) by the following easy-to-check criterion:

$$\|x - [x]_\beta\|^2 + \|y - [y]_\beta\|^2 \le \min\left\{\frac{\tilde{\mu}^2\epsilon_t}{2A(\beta+\tilde{\ell})^2}, \left(\frac{\tilde{\mu}\epsilon_t}{4\beta\mathcal{D}_\mathcal{Y}(\beta+\tilde{\ell})}\right)^2\right\}, \tag{2.5}$$

and (2.3) by the following:

$$\|x - [x]_\beta\|^2 + \|y - [y]_\beta\|^2 \le \frac{\tilde{\mu}^2\epsilon_t}{A(\beta+\tilde{\ell})^2}. \tag{2.6}$$

Note that many algorithms such as EG or GDA, already compute $([x]_\beta, [y]_\beta)$ with β being the stepsize, so there is no additional computation cost to check criterion (2.5) and (2.6).

CHOICE OF REGULARIZATION PARAMETER. As we can see, the smaller τ is, the auxiliary problem is closer to the original problem. However, smaller τ will give rise to worse conditions of the auxiliary problems, making them harder to solve. We will discuss the dependence of the inner and outer loop complexities on τ and provide a guideline for choosing τ for different $\mathcal{M}$.

As a final remark, while the idea of employing the proximal point algorithm (PPA) and adding regularization in minimax optimization has been explored in the literature, it is crucial to emphasize that these existing approaches differ from our framework in various ways. Each has its unique characteristics and methodologies that set it apart from our proposed scheme. To list a few: [Rockafellar, 1976b, Monteiro and Svaiter, 2010, Lin et al., 2018, Palaniappan and Bach, 2016] considered the inexact PPA for C-C or NC-NC minimax problems by adding quadratic terms in both x and y; [Rafique et al., 2022, Thekumparampil et al., 2019] considered the inexact PPA for NC-C minimax problems, by adding a quadratic term in x; [Lin et al., 2020b] considered the inexact accelerated PPA for SC-SC minimax problems by adding a quadratic term in x. On the other hand, a number of work, e.g., [Kong and Monteiro, 2019, Lin et al., 2020b, Zhao, 2020] also add a quadratic term in y to the minimax optimization when the objective is non-strongly concave about y, but in the form $\mathcal{O}(\epsilon)\|y\|^2$, which is completely different from PPA. Besides these differences, the subroutines used to solve the auxiliary minimax problems and choices of regularization parameters in these work are quite distinct from ours. Lastly, we point out that the proposed framework is closely related to the inexact accelerated augmented Lagrangian method designed for linearly constrained optimization problems [Kang et al., 2015], which can be viewed as a special case by setting $f(x, y)$ as the Lagrangian dual. Nevertheless, the strategies for addressing the auxiliary problems and the theoretical analyses between the two are distinctly separate.

2.2.2 *Convergence Analysis*

In order to derive the total complexity, we first establish the complexity of the outer loop and then combine it with the inner loop complexity from algorithm $\mathcal{M}$. We then discuss the optimal choice of the regularization parameter τ for different settings.

Theorem 2.2.3 (Outer-loop complexity for SC-SC objectives)**.** *Suppose function f satisfies Assumptions 2 with $\mu_y > 0$ and Assumption 3. If we choose $\epsilon_t = \frac{\sqrt{2}}{4}(1-\rho)^t \operatorname{gap}_f(x_0, y_0)$ with $\rho < \sqrt{q}$, the output (x_T, y_T) from Algorithm 1 satisfies*

$$\|x_T - x^*\|^2 + \|y_T - y^*\|^2 \leq \left[\frac{48\ell^2}{\mu_x^2 \mu_y (\sqrt{q} - p)^2} + \frac{\sqrt{2}}{\mu_x}\right](1-\rho)^T \operatorname{gap}_f(x_0, y_0),$$

where $q \triangleq \frac{\mu_y}{\mu_y+\tau}$ *as defined in Algorithm 1.*

Remark 2.2.4. *In practice, if we choose* $\rho = 0.9\sqrt{q} = 0.9\sqrt{\frac{\mu_y}{\mu_y+\tau}}$*, Theorem 2.2.3 implies that Algorithm 1 outputs a point* (x_T, y_T) *such that* $\|x_T - x^*\|^2 + \|y_T - y^*\|^2 \leq \epsilon$ *within* $\tilde{\mathcal{O}}\left(\sqrt{\frac{\tau+\mu_y}{\mu_y}}\log(\frac{1}{\epsilon})\right)$ *iterations. By Lemma 2.2.2, we can find a point with* ϵ*-primal-dual gap with the same order of complexity by performing a projected gradient descent ascent step.*

Theorem 2.2.5 (Outer-loop complexity for SC-C objectives)**.** *Suppose function* f *satisfies Assumptions 2 with* $\mu_y = 0$ *and Assumption 3. The output* $(\bar{x}_T, y_T)$ *from Algorithm 1 satisfies*

$$\text{gap}_f(\bar{x}_T, y_T) \leq \alpha_T^2\left[\frac{\tau}{2}\mathcal{D}_{\mathcal{Y}}^2 + 2\sum_{t=1}^{T}\frac{1}{\alpha_t^2}\epsilon_t\right]. \tag{2.7}$$

If we further choose $\epsilon_t = \frac{\tau(\rho-1)\mathcal{D}_{\mathcal{Y}}^2\alpha_t^2}{4(t+1)^\rho}$ *with* $\rho > 1$*, then*

$$\text{gap}_f(\bar{x}_T, y_T) \leq \alpha_T^2\tau\mathcal{D}_{\mathcal{Y}}^2. \tag{2.8}$$

Remark 2.2.6. *The above result still holds true without requiring strong convexity in* x*. In addition, the regularization parameter* τ *can be any positive value, so Algorithm 1 is quite flexible. Because* $2/(t+2)^2 \leq \alpha_t^2 \leq 4/(t+1)^2$ *[Paquette et al., 2017], Theorem 2.2.5 implies that the algorithm finds a point with* ϵ*-primal-dual gap within* $\mathcal{O}(\sqrt{\tau/\epsilon}\mathcal{D}_{\mathcal{Y}} + 1)$ *outer-loop iterations. Notice that the outer-loop complexity decreases as* τ *decreases.*

We now delve into the complexity of the inner loop. By design, the auxiliary problem $(\star)$ is $(\mu_x, \mu_y+\tau)$-SC-SC and can be solved by many existing first-order algorithms at a linear convergence rate. For ease of reference, we denote the optimal solution to the auxiliary problem $(\star)$ as (x_t^*, y_t^*). We first introduce a straightforward warm start for the auxiliary problems: the previous iterate, (x_{t-1}, y_{t-1}), serves as the initial point for $\mathcal{M}$. We show that the distance between this starting point and (x_t^*, y_t^*) is relatively small or bounded. Subsequently, we outline the complexity of the inner loop.

Lemma 2.2.7 (Warm start for SC-SC objectives)**.** *Suppose function* f *satisfies Assumptions 2 with* $\mu_y > 0$ *and Assumption 3 and we run Algorithm 1 with* ϵ_t *specified in Theorem 2.2.3. If we set the initial point of the auxiliary problem* $(\star)$ *at iteration* t *to be* (x_{t-1}, y_{t-1})*, then we have*

$$\|x_{t-1} - x_t^*\|^2 + \|y_{t-1} - y_t^*\|^2 \leq C_t\epsilon_t,$$

where $C_1 = \frac{16\sqrt{2}\ell^2}{\mu_x^2\min\{\mu_x,\mu_y\}} + \frac{16\sqrt{2}\ell^2+4\sqrt{2}\mu_x^2}{(2\tau+\mu_y)\mu_x^2}$ *and* $C_t = \left(\frac{4}{\mu_x} + \frac{4}{\mu_y+\tau}\right)\frac{1}{1-\rho} + \frac{2304\sqrt{2}\tau^2\left(\ell^2+\mu_x^2\right)}{\mu_y\mu_x^2(\mu_y+\tau)^2(\sqrt{q}-p)}\frac{1}{(1-\rho)^2}$ *for* $t > 1$.

Lemma 2.2.8 (Warm start for SC-C objectives). *Suppose function f satisfies Assumptions 2 with $\mu_y = 0$ and Assumption 3 and we run Algorithm 1 with ϵ_t specified in Theorem 2.2.5. If we set the initial point of the auxiliary problem $(\star)$ at iteration t to be (x_{t-1}, y_{t-1}), then we have*

$$\|x_{t-1} - x_t^*\|^2 + \|y_{t-1} - y_t^*\|^2 \le D_t,$$

where $D_1 = 2\|x_0 - x^*\|^2 + \left(\frac{2\ell^2}{\mu_x^2} + 1\right)\mathcal{D}_{\mathcal{Y}}^2$ and $D_t = \left[\tau(\rho-1)\left(\frac{1}{\mu_x} + \frac{1}{\mu_y+\tau}\right) + 2\left(\frac{\ell^2}{\mu_x^2} + 1\right)\right]\mathcal{D}_{\mathcal{Y}}^2$ for $t > 1$.

Corollary 2.2.9 (Inner-loop complexity for SC-SC objectives). *Under the same assumptions as Lemma 2.2.7, suppose we apply a linearly convergent algorithm $\mathcal{M}$ described by (2.4) to solve the auxiliary problem $(\star)$ with the initial point specified in Lemma 2.2.7. The number of iterations (expected number of iterations if $\mathcal{M}$ is stochastic) for $\mathcal{M}$ to find a point satisfying (2.6) is $N_t = \mathcal{O}\left(\Lambda_{\mathcal{M},\tau}\log\left(\frac{\max\{1,\ell,\tau\}}{\min\{1,\mu_x,\mu_y\}}\right)\right)$.*

Corollary 2.2.10 (Inner-loop complexity for SC-C objectives). *Under the same assumptions as Lemma 2.2.8, suppose we apply a linearly convergent algorithm $\mathcal{M}$ described by (2.4) to solve the auxiliary problem $(\star)$ with the initial point specified in Lemma 2.2.8. The number of iterations (expected number of iterations if $\mathcal{M}$ is stochastic) for $\mathcal{M}$ to find a point satisfying (2.5) is $N_t = \mathcal{O}\left(\Lambda_{\mathcal{M},\tau}\log\left(\frac{\max\{1,\ell,\tau\}(\mathcal{D}_{\mathcal{Y}}+\|x_0-x^*\|)t}{\min\{1,\mu_x,\tau\}}\right)\right)$.*

In practice, choosing a good initial point to warm start algorithm $\mathcal{M}$ can be helpful in accelerating the convergence. Without the warm start strategy, one would require $\mathcal{X}$ to be bounded and $N_t = \tilde{\mathcal{O}}\left(\Lambda_{\mathcal{M},\tau}\log(\frac{\mathcal{D}_{\mathcal{X}}+\mathcal{D}_{\mathcal{Y}}}{\epsilon_t})\right)$. The above corollaries show that in theory, using a simple warm start strategy helps to remove the assumption on boundedness of $\mathcal{X}$ and when $\mu_y > 0$ the inner-loop complexity does not increase with t.

As we can see, the choice of τ plays a crucial role since it affects both inner-loop and outer-loop complexities. Combining the above two results immediately leads to the total complexities:

Corollary 2.2.11 (Total complexity for SC-SC objectives). *Suppose function f satisfies Assumption 2 with $\mu_y > 0$ and Assumption 3, and the auxiliary problems are solved by a linearly convergent algorithm $\mathcal{M}$ to satisfy the stopping criterion (2.3) with accuracy ϵ_t as specified in Theorem 2.2.3. For Algorithm 1 to find an ϵ-saddle point, the total number of gradient evaluations (expected number if $\mathcal{M}$ is stochastic) is*

$$\begin{aligned}&\mathcal{O}\left(\Lambda_{\mathcal{M},\tau}\sqrt{\frac{\mu_y+\tau}{\mu_y}}\log\left(\frac{\max\{1,\ell,\tau\}}{\min\{1,\mu_x,\mu_y\}}\right)\log\left(\frac{\max\{1,\ell\}\,\mathrm{gap}_f(x_0,y_0)}{\min\{1,\mu_x,\mu_y\}}\cdot\frac{1}{\epsilon}\right)\right)\\ &= \tilde{\mathcal{O}}\left(\Lambda_{\mathcal{M},\tau}\sqrt{\frac{\mu_y+\tau}{\mu_y}}\log\left(\frac{1}{\epsilon}\right)\right).\end{aligned}$$

Corollary 2.2.12 (Total complexity for SC-C objectives)**.** *Suppose function f satisfies Assumption 2 with $\mu_y = 0$ and Assumption 3, and the auxiliary problems are solved by a linearly convergent algorithm $\mathcal{M}$ to satisfy the stopping criterion (2.2) or (2.5) with accuracy ϵ_t as specified in Theorem 2.2.5. For Algorithm 1 to find an ϵ-saddle point, the total number of gradient evaluations (expected number if $\mathcal{M}$ is stochastic) is*

$$\begin{aligned}&\mathcal{O}\left(\Lambda_{\mathcal{M},\tau}\left(\sqrt{\tau/\epsilon}\mathcal{D}_{\mathcal{Y}}+1\right)\log\left(\frac{\max\{1,\ell,\tau\}(\mathcal{D}_{\mathcal{Y}}+\|x_0-x^*\|)}{\min\{1,\mu_x,\tau\}}\cdot\frac{1}{\epsilon}\right)\right)\\&=\tilde{\mathcal{O}}\left(\Lambda_{\mathcal{M},\tau}\left(\sqrt{\tau/\epsilon}\mathcal{D}_{\mathcal{Y}}+1\right)\log\left(\frac{1}{\epsilon}\right)\right).\end{aligned}$$

For any given linearly-convergent method, denoted as $\mathcal{M}$, and any selected regularization parameter τ, the oracle complexity for SC-SC objectives stands at $\mathcal{O}\left(\log\frac{1}{\epsilon}\right)$. This is already optimal in terms of ϵ as per [Ouyang and Xu, 2019]. Meanwhile, the oracle complexity for SC-C objectives is $\mathcal{O}\left(\mathcal{D}_{\mathcal{Y}}/\sqrt{\epsilon}\log(\mathcal{D}_{\mathcal{Y}}/\epsilon)\right)$. This is optimal in both ϵ and $\mathcal{D}_{\mathcal{Y}}$, up a logarithmic factor, as indicated by [Ouyang and Xu, 2019]. The dependence on the condition number will be determined by the term $\Lambda_{\mathcal{M},\tau}\sqrt{\tau}$, which we analyze in detail below for specific algorithms.

2.2.3 *Specific Algorithms and Complexities*

In order to minimize the total complexity, we should choose the regularization parameter $\tau > 0$ that minimizes $\Lambda_{\mathcal{M},\tau}\sqrt{\mu_y+\tau}$ when $\mu_y > 0$, and minimizes $\Lambda_{\mathcal{M},\tau}\sqrt{\tau}$ when $\mu_y = 0$. Below we derive the choice of the optimal τ for different algorithms $\mathcal{M}$ and present the corresponding total complexity. Table 2.3 and Table 2.4 summarize the results for SC-SC and SC-CC minimax optimization, respectively.

DETERMINISTIC FIRST-ORDER ALGORITHMS. When employing GDA as $\mathcal{M}$ to solve the auxiliary problem, the value of $\Lambda_{\mathcal{M},\tau}$ is given by $\Lambda_{\mathcal{M},\tau} = \left(\frac{\ell+\tau}{2\min\{\mu_x,\mu_y+\tau\}}\right)^2$ [Facchinei and Pang, 2007]. If we use EG or OGDA as $\mathcal{M}$, then $\Lambda_{\mathcal{M},\tau} = \frac{\ell+\tau}{4\min\{\mu_x,\mu_y+\tau\}}$ [Gidel et al., 2018, Azizian et al., 2019]. Minimizing $\Lambda_{\mathcal{M},\tau}\sqrt{\tau}$ for both cases yields that the optimal choice for τ is $\mu_x - \mu_y$. Specifically, when using EG or OGDA, the total complexity of finding an ϵ-saddle point ($\epsilon \le \mu_x\mathcal{D}_{\mathcal{Y}}^2$ when $\mu_y = 0$) is

$$\begin{cases}\tilde{\mathcal{O}}\left(\frac{\ell\cdot\mathcal{D}_{\mathcal{Y}}}{\sqrt{\mu_x\epsilon}}\log\frac{1}{\epsilon}\right), & \text{when } \mu_y = 0;\\ \tilde{\mathcal{O}}\left(\frac{\ell}{\sqrt{\mu_x\mu_y}}\log\frac{1}{\epsilon}\right), & \text{when } \mu_y > 0.\end{cases} \tag{2.9}$$

Remark 2.2.13. *The complexity for the SC-SC setting matches the lower complexity bound for this class of problem [Zhang et al., 2019b], up to only a logarithmic factor in problem parameters. Lin et al. [2020b] and Wang and Li [2020] also achieve $\tilde{\mathcal{O}}\left(\frac{\ell}{\sqrt{\mu_x\mu_y}}\right)$ dependency on the condition number, but has higher order poly-logarithmic terms in ϵ or problem parameters.*

Remark 2.2.14. *The complexity for SC-C setting matches the lower complexity bound for this class of problems [Ouyang and Xu, 2019] in ϵ, ℓ_x, μ and $\mathcal{D}_\mathcal{Y}$, up to a logarithmic factor. In addition, it improves over [Thekumparampil et al., 2019] on the dependency in condition number and improves over [Wang and Li, 2020], which has higher order poly-logarithmic factor in ℓ, μ_x and $\mathcal{D}_\mathcal{Y}$.*

A key observation is that by setting $\tau = \mu_x - \mu_y$, the auxiliary problem $(\star)$ becomes (μ_x, μ_x)-SC-SC, and it is known that simple EG or OGDA achieves the optimal complexity for solving this class of well-balanced SC-SC problems [Zhang et al., 2019b]. Subproblems in [Thekumparampil et al., 2019, Lin et al., 2020b] are harder to solve because of ill-balanced condition numbers, thus requiring complicated subroutines.

Besides the complexity improvement, our algorithm is significantly simpler and easier to implement than the current state-of-the-arts. Under SC-SC setting, Minimax-APPA in [Lin et al., 2020b] and APPA-ABR in [Wang and Li, 2020] are triple-loop algorithms which stack several acceleration schemes together. Under SC-C setting, they add a smoothing term in y to induce a SC-SC auxiliary problem. The DIAG algorithm in [Thekumparampil et al., 2019] applies Nesterov's accelerated gradient ascent to the dual function and an additional two-loop algorithm to solve their subproblems. In contrast, our algorithm only requires two loops for either setting, does not require to prefix accuracy ϵ, and has fewer tuning parameters.

STOCHASTIC VARIANCE-REDUCED ALGORITHMS. We now turn our attention to minimax problems with a finite-sum structure, represented as $\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}} f(x,y) \triangleq \frac{1}{n}\sum_{i=1}^n f_i(x,y)$. Correspondingly, the SC-SC auxiliary problem $(\star)$ can also be readily written in a finite-sum form. When f is ℓ-AS, this auxiliary problem can be solved by a number of linearly-convergent variance-reduced algorithms, such as SVRG, accelerated SVRG (Acc-SVRG) [Balamurugan and Bach, 2016][1] [2], and EG with variance reduction (VR-EG) [Alacaoglu and Malitsky, 2022].

When using SVRG or SAGA as $\mathcal{M}$, we have $\Lambda_{\mathcal{M},\tau} \propto n + \left(\frac{\ell+\tau}{\min\{\mu_x,\mu_y+\tau\}}\right)^2$ [Balamurugan and Bach, 2016]. If Acc-SVRG or VR-EG is employed, $\Lambda_{\mathcal{M},\tau} \propto n + \frac{\sqrt{n}(\ell+\tau)}{\min\{\mu_x,\tau\}}$ [Balamurugan

1 Although Balamurugan and Bach [2016] assumes individual smoothness, their analysis can be extended to the averaged smoothness.

2 Algorithms in [Balamurugan and Bach, 2016] requires computing the proximal operator of an (μ,μ)-SC-SC function. For any (μ,μ)-SC-SC function in the form of $\sum_i f_i(x,y)$, we can rewrite it as $\sum_i[f_i(x,y) - \frac{\mu}{2n}||x||^2 + \frac{\mu}{2n}||y||^2] + \frac{\mu}{2}(||x||^2 - ||y||^2)$, where the first term is convex-concave, and the second term is (μ,μ)-SC-SC and admits a simple proximal operator.

$\mathcal{M}$	$\Lambda_{\mathcal{M},\tau} \propto$	Choice for τ	Total Complexity of Catalyst
GDA	$\left(\frac{\ell+\tau}{\min\{\mu_x,\mu_y+\tau\}}\right)^2$	$\mu_x - \mu_y$	$\tilde{\mathcal{O}}\left(\frac{n\ell^2}{\sqrt{\mu_x^3\mu_y}}\log\frac{1}{\epsilon}\right)$
EG/OGDA	$\frac{\ell+\tau}{\min\{\mu_x,\mu_y+\tau\}}$	$\mu_x - \mu_y$	$\tilde{\mathcal{O}}\left(\frac{n\ell}{\sqrt{\mu_x\mu_y}}\log\frac{1}{\epsilon}\right)$
SVRG/SAGA	$n+\left(\frac{\ell+\tau}{\min\{\mu_x,\mu_y+\tau\}}\right)^2$	$\mu_x-\mu_y$, if $\ell \geq \mu_x\sqrt{n}$ $\frac{\ell}{\sqrt{n}}-\mu_y$, if $\mu_y\sqrt{n}\leq\ell\leq\mu_x\sqrt{n}$ 0, if $\ell\leq\mu_y\sqrt{n}$	$\tilde{\mathcal{O}}\left(\left(n+\frac{\ell^2}{\sqrt{\mu_x^3\mu_y}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}}{\sqrt{\mu_y}}\right)\log\frac{1}{\epsilon}\right)$
Acc-SVRG/ VR-EG	$n+\frac{\sqrt{n}(\ell+\tau)}{\min\{\mu_x,\mu_y+\tau\}}$	$\mu_x-\mu_y$, if $\ell \geq \mu_x\sqrt{n}$ $\frac{\ell}{\sqrt{n}}-\mu_y$, if $\mu_y\sqrt{n}\leq\ell\leq\mu_x\sqrt{n}$ 0, if $\ell\leq\mu_y\sqrt{n}$	$\tilde{\mathcal{O}}\left(\left(n+\frac{\sqrt{n}\ell}{\sqrt{\mu_x\mu_y}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}}{\sqrt{\mu_y}}\right)\log\frac{1}{\epsilon}\right)$

TABLE 2.3: The table summarizes the optimal choice of regularization parameter τ and total complexity of the proposed Catalyst framework for finite-sum SC-SC minimax optimization with $f(x,y)=\frac{1}{n}\sum_{i=1}^n f_i(x,y)$, when combined with different methods $\mathcal{M}$.

and Bach, 2016, Alacaoglu and Malitsky, 2022]. Specifically, for Acc-SVRG or VR-EG, the optimal τ is (proportional to) $\mu_x-\mu_y$ if $\ell\geq\mu_x\sqrt{n}$, $\frac{\ell}{\sqrt{n}}-\mu_y$ if $\mu_y\sqrt{n}\leq\ell\leq\mu_x\sqrt{n}$, and 0 otherwise. Therefore, the total complexity for the case $\mu_y>0$ is

$$\begin{cases}\tilde{\mathcal{O}}\left(\frac{\sqrt{n}\ell}{\sqrt{\mu_x\mu_y}}\log\frac{1}{\epsilon}\right), & \text{if } \ell\geq\mu_x\sqrt{n};\\ \tilde{\mathcal{O}}\left(\frac{n^{\frac{3}{4}}\sqrt{\ell}}{\sqrt{\mu_y}}\log\frac{1}{\epsilon}\right), & \text{if } \mu_y\sqrt{n}\leq\ell\leq\mu_x\sqrt{n};\\ \tilde{\mathcal{O}}\left(n\log\frac{1}{\epsilon}\right), & \text{otherwise.}\end{cases} \tag{2.10}$$

The total complexity for the case $\mu_y=0$ is

$$\begin{cases}\tilde{\mathcal{O}}\left(\left(n+\frac{\sqrt{n}\ell\mathcal{D}_y}{\sqrt{\mu_x\epsilon}}\right)\log\frac{1}{\epsilon}\right), & \text{if } \ell\geq\mu_x n;\\ \tilde{\mathcal{O}}\left(\left(n+\frac{n^{\frac{3}{4}}\sqrt{\ell}\mathcal{D}_y}{\sqrt{\epsilon}}\right)\log\frac{1}{\epsilon}\right), & \text{otherwise.}\end{cases} \tag{2.11}$$

Remark 2.2.15. *In the SC-SC setting, our complexity, as presented in (2.10), aligns with the recently established lower complexity bound in [Han et al., 2021], with differences only in logarithmic factors related to μ_x, μ_y, and ℓ. For the SC-C setting, the complexity in (2.11) also nearly matches*

$\mathcal{M}$	$\Lambda_{\mathcal{M},\tau} \propto$	Choice for τ	Total Complexity of Catalyst
GDA	$\left(\frac{\ell+\tau}{\min\{\mu_x,\tau\}}\right)^2$	μ_x	$\tilde{\mathcal{O}}\left(\frac{n\ell^2\mathcal{D}_\mathcal{Y}}{\sqrt{\mu_x^3\epsilon}}\log\frac{1}{\epsilon}\right)$
EG/OGDA	$\frac{\ell+\tau}{\min\{\mu_x,\tau\}}$	μ_x	$\tilde{\mathcal{O}}\left(\frac{n\ell\mathcal{D}_\mathcal{Y}}{\sqrt{\mu_x\epsilon}}\log\frac{1}{\epsilon}\right)$
SVRG/SAGA	$n+\left(\frac{\ell+\tau}{\min\{\mu_x,\tau\}}\right)^2$	μ_x, if $\ell \geq \mu_x\sqrt{n}$ $\frac{\ell}{\sqrt{n}}$, if $\ell < \mu_x\sqrt{n}$	$\tilde{\mathcal{O}}\left(\left(n+\frac{\ell^2\mathcal{D}_\mathcal{Y}}{\sqrt{\mu_x^3\epsilon}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}\mathcal{D}_\mathcal{Y}}{\sqrt{\epsilon}}\right)\log\frac{1}{\epsilon}\right)$
Acc-SVRG/ VR-EG	$n+\frac{\sqrt{n}(\ell+\tau)}{\min\{\mu_x,\tau\}}$	μ_x, if $\ell \geq \mu_x\sqrt{n}$ $\frac{\ell}{\sqrt{n}}$, if $\ell < \mu_x\sqrt{n}$	$\tilde{\mathcal{O}}\left(\left(n+\frac{\sqrt{n}\ell\mathcal{D}_\mathcal{Y}}{\sqrt{\mu_x\epsilon}}+\frac{n^{\frac{3}{4}}\ell^{\frac{1}{2}}\mathcal{D}_\mathcal{Y}}{\sqrt{\epsilon}}\right)\log\frac{1}{\epsilon}\right)$

TABLE 2.4: The table summarizes the optimal choice of regularization parameter τ and total complexity of the proposed Catalyst framework for finite-sum SC-C minimax optimization with $f(x,y)=\frac{1}{n}\sum_{i=1}^n f_i(x,y)$, when combined with different methods $\mathcal{M}$.

with the lower complexity bound in [Han et al., 2021], up to logarithmic factors with problem parameters and ϵ. Notably, these complexities improve over Acc-SVRG and batch Catalyst-EG.

2.3 NONCONVEX-(STRONGLY)-CONCAVE MINIMAX OPTIMIZATION

We now shift our focus to nonconvex-strongly-concave (NC-SC) and nonconvex-concave (NC-C) minimax problems. We continue to assume that f has ℓ-Lipschitz gradients, as stated in Assumption 3.

Assumption 5. *$f(x,\cdot)$ is μ-strongly-concave for any x in $\mathcal{X}$ with $\mu \geq 0$, i.e.,*

$$f(x,y_1) \geq f(x,y_2) + \nabla_y f(x,y_2)^T(y_1-y_2) + \frac{\mu_y}{2}\|y_1-y_2\|^2, \quad \forall y_1,y_2 \in \mathcal{Y}.$$

$\mathcal{X}$ and $\mathcal{Y}$ are convex and closed sets, and we further assume $\mathcal{Y}$ be bounded with diameter $\mathcal{D}_\mathcal{Y}$ when $\mu_y = 0$.

When the strong concavity modulus $\mu > 0$, we refer to the setting as μ-NC-SC; when $\mu = 0$, we refer to the setting as NC-C. Given the objective function is nonconvex about x, finding a global solution will be intractable. Our goal is to identify an approximate stationary solution for the primal function $\Phi(x)$. As per [Lin et al., 2020a], in the NC-SC setting, Φ is differentiable and L-smooth with $L = \frac{2\ell^2}{\mu}$. In the NC-C setting, however, Φ is ℓ-weakly convex and might not be differentiable [Thekumparampil et al., 2019]. Consequently, we adopt distinct stationarity concepts for these two settings.

Definition 12 (Stationary point in NC-SC setting)**.** *For a differentiable Φ, a point x^* is an ϵ-stationary point of Φ if*

$$\ell \left\| x^* - \mathcal{P}_{\mathcal{X}} \left(x^* - \frac{1}{\ell} \nabla \Phi(x^*) \right) \right\| \leq \epsilon.$$

Definition 13 (Stationary point in NC-C setting)**.** *For a weakly convex (and potentially non-differentiable) Φ, we first define the Moreau envelope as:*

$$\Phi_\lambda(z) \triangleq \min_{x \in \mathcal{X}} \Phi(x) + \frac{1}{2\lambda} \|x - z\|^2.$$

A point x^* is an ϵ-stationary point of Φ if $\left\| \nabla \Phi_{\frac{1}{2\ell}}(x^*) \right\| \leq \epsilon$.

2.3.1 *A Catalyst Framework*

Our nonconvex Catalyst framework is described in Algorithm 2. This can be considered as applying the proximal point algorithm to the primal function $\Phi(x) \triangleq \max_{y \in \mathcal{Y}} f(x, y)$, leading to a new minimax subproblem ($\star\star$), which is (ℓ, μ)-SC-SC after we add the regularzation in x. This is in the same spirit as [Rafique et al., 2022, Thekumparampil et al., 2019, Lin et al., 2020b]. The main difference lies in that we use Algorithm 1 to solve this unbalanced subproblems. The algorithm is easier to implement than others, because the overall algorithm, i.e., Algorithm 2 equipped with Algorithm 1 to solve the subproblem, can be considered as a two-time-scale inexact proximal point algorithm, which repeatedly solves a series of problem with changing prox centers $\tilde{x}_t$ and z_t,

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y) + \ell \|x - \tilde{x}_t\|^2 + \tfrac{\tau}{2} \|y - z_t\|^2, \tag{2.12}$$

by some existing algorithm $\mathcal{M}$.

2.3.2 *Convergence Analysis*

We begin by presenting the convergence analysis for the outer loop of Algorithm 2. The subproblem ($\star\star$) is strongly-convex-(strongly)-concave, so we can utilize the results from Chapter 2.2 to determine the inner-loop complexity.

Algorithm 2 `Catalyst for NC-C/NC-SC Minimax Optimization`

1: Input: initial point (x_0, y_0)
2: **for all** $t = 0, 1, ..., T$ **do**
3: Find an inexact solution (x_{t+1}, y_{t+1}) to the following problem by Algorithm 1 from the initial point (x_t, y_t)

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} \left[\hat{f}_t(x,y) \triangleq f(x,y) + \ell\|x - x_t\|^2\right] \tag{$\star\star$}$$

such that

$$\text{if } \mu_y = 0 : \text{gap}_{\hat{f}_t}(x_{t+1}, y_{t+1}) \le \hat{\epsilon}_t; \tag{2.13}$$

$$\text{if } \mu_y > 0 : \text{gap}_{\hat{f}_t}(x_{t+1}, y_{t+1}) \le \alpha_t(\|x_t - \hat{x}_t\|^2 + \|y_t - \hat{y}_t\|^2). \tag{2.14}$$

4: **end for**
5: Output: $\hat{x}_T$, which is uniformly sampled from $x_0, ..., x_{T-1}$ if $\mu_y = 0$, and from $x_1, ..., x_T$ if $\mu_y > 0$.

Theorem 2.3.1 (Outer-loop complexity for NC-SC objectives). *Suppose f satisfies Assumption 5 with $\mu_y > 0$ and Assumption 3. If we choose $\alpha_t = \frac{\mu^4}{28\ell^3}$ for $t > 0$ and $\alpha_0 = \frac{\mu^4}{32\ell^4 \max\{1,\ell\}}$, the output from Algorithm 2 satisfies*

$$\frac{1}{T}\sum_{t=1}^{T} \ell^2 \left\|\hat{x}_T - \mathcal{P}_{\mathcal{X}}\left(\hat{x}_T - \frac{1}{\ell}\nabla\Phi(\hat{x}_T)\right)\right\|^2 \le \frac{275\ell}{5T}\Delta + \frac{35\ell}{5T}D_y^0, \tag{2.15}$$

where $\Delta = \Phi(x_0) - \min_{x\in\mathcal{X}}\Phi(x)$, $D_y^0 = \|y_0 - y^*(x_0)\|^2$ and $y^*(x_0) = \text{argmax}_{y\in\mathcal{Y}} f(x_0, y)$.

Theorem 2.3.2 (Outer-loop complexity for NC-C objectives). *Suppose f satisfies Assumption 5 with $\mu_y = 0$ and Assumption 3. The output from Algorithm 2 satisfies*

$$\mathbb{E}\left\|\nabla\Phi_{\frac{1}{2\ell}}(\hat{x}_T)\right\|^2 \le \frac{8\ell}{T}\left[\Delta + \sum_{t=0}^{T-1}\hat{\epsilon}_t\right],$$

where $\Delta = \Phi(x_0) - \min_{x\in\mathcal{X}}\Phi(x)$. If $T = 16\ell\Delta\epsilon^{-2}$ and $\hat{\epsilon}_t \le \frac{\epsilon^2}{8\ell}$, then $\mathbb{E}\left[\|\nabla\Phi_{\frac{1}{2\ell}}(\hat{x}_T)\|\right] \le \epsilon$.

Remark 2.3.3. *Above we choose $\hat{\epsilon}_t = \mathcal{O}(\epsilon^2)$, which requires to fix the target accuracy. We can also choose $\hat{\epsilon}_t$ to be decreasing, i.e. $\hat{\epsilon}_t = \frac{\Delta}{t+1}$, which leads to $\mathbb{E}\left[\|\nabla\Phi_{1/2\ell}(\hat{x}_T)\|^2\right] \le \frac{8\ell(1+2\log T)\Delta}{T}$. Compared with the constant $\hat{\epsilon}_t$, it has an additional logarithmic term in T.*

Corollary 2.2.11 and 2.2.12 will capture the complexity of solving auxiliary problems ($\star\star$) with Algorithm 1. However, we should first specify how far the initial point (x_t, y_t) we pick for the subproblem is from the optimal solution of ($\star\star$) compared to the target accuracy we want.

Lemma 2.3.4 (Warm start for NC-SC objectives). *Under the assumptions in Theorem 2.3.1, if we can find a point* (x_{t+1}, y_{t+1}) *such that* $\operatorname{gap}_{\hat{f}_t}(x_{t+1}, y_{t+1}) \le \frac{\alpha_t}{A} \operatorname{gap}_{\hat{f}_t}([z_t]_\beta)$*, where* $z_t = (x_t, y_t)$*,* $\beta > 6\ell$*, and* A *and* $[z_t]_\beta$ *are defined as in Lemma 2.2.2, then it also satisfies the stopping criterion (2.14).*

Lemma 2.3.5 (Warm start for NC-C objectives). *Under the assumptions in Theorem 2.3.2, suppose we run Algorithm 2 with* $\hat{\epsilon}_t = \frac{\epsilon^2}{8\ell}$*. As we set the initial point of the algorithm for solving subproblem (⋆⋆) as* (x_t, y_t)*, then for all* $t < T = 16\ell\Delta\epsilon^{-2}$*, we have* $\|x_t - \hat{x}_t^*\| \le \sqrt{\frac{6\Delta}{\ell}}$*, where* $(\hat{x}_t^*, \hat{y}_t^*)$ *is the saddle point of* $\hat{f}_t$*.*

2.3.3 *Specific Algorithms and Complexities*

Theorem 2.3.2 and 2.3.1 imply that the outer-loop complexity is $\tilde{\mathcal{O}}\left(\ell\Delta\epsilon^{-2}\right)$. In the following corollaries, we specify the choices of τ and $\mathcal{M}$ for solving subproblems and the total complexity.

DETERMINISTIC FIRST-ORDER ALGORITHMS. Since $(\star\star)$ is (ℓ, μ_y)-SC-SC, by our discussion in Section 2.2.3, the best choice for τ is $\ell - \mu_y$ no matter we choose GDA, EG or OGDA as $\mathcal{M}$ in Algorithm 1. Then Algorithm 2 finds an ϵ-stationary point ($\epsilon \le \ell\mathcal{D}_y^2$ when $\mu_y = 0$) with the total number of gradient evaluations of

$$\begin{cases} \tilde{\mathcal{O}}\left(\dfrac{\ell^2\mathcal{D}_y\Delta}{\epsilon^3}\log\left(\dfrac{1}{\epsilon}\right)\right), & \text{when } \mu = 0; \\ \tilde{\mathcal{O}}\left(\dfrac{\ell^{\frac{3}{2}}(\Delta + D_y^0)}{\sqrt{\mu}\epsilon^2}\right), & \text{when } \mu > 0. \end{cases}$$

Remark 2.3.6. *The above complexity for the NC-SC setting matches the lower bound in [Zhang et al., 2021b, Han et al., 2021], up to a logarithmic factor in* L *and* κ *(*$\kappa \triangleq \frac{\ell}{\mu}$*). It improves over Minimax-PPA [Lin et al., 2020b] by* $\log^2(1/\epsilon)$*, GDA [Lin et al., 2020a] by* $\kappa^{\frac{3}{2}}$ *and therefore achieves the best of two worlds in terms of dependency on* κ *and* ϵ*. In addition, our Catalyst-EG/OGDA algorithm does not require the bounded domain assumption on* y*, unlike [Lin et al., 2020b].*

Remark 2.3.7. *The complexity for the NC-C setting matches with the current state-of-the-art complexity for nonconvex-concave minimization [Lin et al., 2020b, Thekumparampil et al., 2019, Zhao, 2020, Ostrovskii et al., 2020] with improvement in logarithmic factors. Note that our algorithm is much simpler than the existing algorithms, e.g., Prox-DIAG [Thekumparampil et al., 2019] requires a four-loop procedure.*

STOCHASTIC VARIANCE-REDUCED ALGORITHMS. We now consider finite-sum-structure minimax problems, $f(x,y) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_i(x,y)$ and assume f is ℓ-AS. Since the ratio between smoothness constant and strong-convexity constant in auxiliary problem $(\star\star)$ is $\Theta(1)$, according to Section 2.2.3, the best choice for τ_y is proportional to $\max\left\{\frac{\ell}{\sqrt{n}} - \mu_y, 0\right\}$ when we choose SVRG or SAGA as $\mathcal{M}$. In particular, the total complexity is

$$\begin{cases} \tilde{\mathcal{O}}\left(\left(\frac{n^{\frac{3}{4}}\ell^2\mathcal{D}_y\Delta}{\epsilon^3} + \frac{n\ell\Delta}{\epsilon^2}\right)\log\left(\frac{1}{\epsilon}\right)\right), & \text{when } \mu_y = 0; \\ \tilde{\mathcal{O}}\left(\left(n + \frac{n^{3/4}\ell^{1/2}}{\mu_y^{1/2}}\right)\frac{\ell(\Delta + G_y^0)}{\epsilon^2}\right), & \text{when } \mu_y > 0. \end{cases}$$

Remark 2.3.8. *In the NC-SC setting, according to the lower bound established in [Zhang et al., 2021b], the dependency on κ in the above upper bound is nearly tight, up to logarithmic factors. Recall that SREDA [Luo et al., 2020] achieves the complexity of $\tilde{O}\left(\kappa^2\sqrt{n}\epsilon^{-2} + n + (n+\kappa)\log(\kappa)\right)$ for $n \geq \kappa^2$ and $O\left(\left(\kappa^2 + \kappa n\right)\epsilon^{-2}\right)$ for $n \leq \kappa^2$. Hence, our Catalyst-SVRG/SAGA algorithm attains better complexity in the regime $n \leq \kappa^4$. Particularly, in the critical regime $\kappa = \Omega(\sqrt{n})$ arising in statistical learning [Shalev-Shwartz and Ben-David, 2014], our algorithm performs strictly better.*

Remark 2.3.9. *Variance reduced algorithms are still under-explored for NC-C setting. PG-SVRG proposed in [Rafique et al., 2022] provides a complexity of $\tilde{\mathcal{O}}\left(n\epsilon^{-2} + \epsilon^{-6}\right)$, which has a much worse dependence on ϵ and n.*

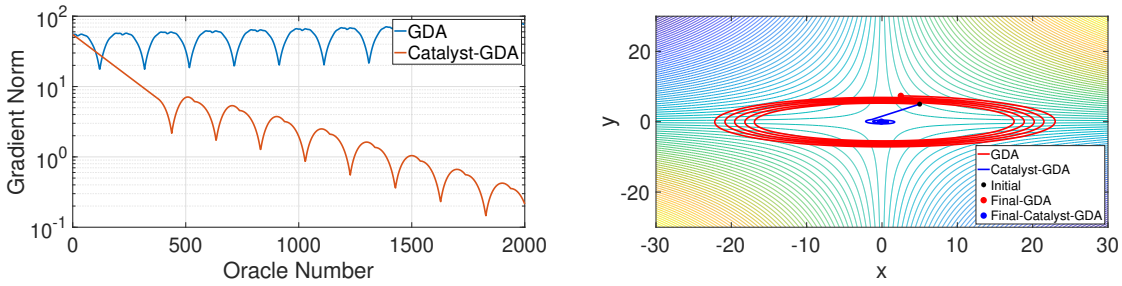
2.4 NUMERICAL EXPERIMENTS

In this subchapter, we carry out experiments across a range of applications to demonstrate the enhanced performance of Catalyst in diverse settings. Our experiments encompass a communication challenge involving an adversary, a binary classification task, and distributionally robust logistic regression. Our primary emphasis will be on comparing the performance of algorithms before and after the integration of Catalyst acceleration.

2.4.1 *2-D Nonconvex-Concave Example*

We design a straightforward nonconvex-concave example, drawing inspiration from an example presented in [Abernethy et al., 2019]. The smooth function is defined as:

$$F(x) = \begin{cases} -\frac{1}{2}\cos x - \frac{1}{2}x - \frac{1}{4}\pi & \text{for } x \leq -\frac{\pi}{2}, \\ -\cos x & \text{for } -\frac{\pi}{2} < x \leq \frac{\pi}{2}, \\ -\frac{1}{2}\cos x + \frac{1}{2}x - \frac{1}{4}\pi & \text{for } x > \frac{\pi}{2}. \end{cases}$$



(a) Gradient mapping (b) Trajectory

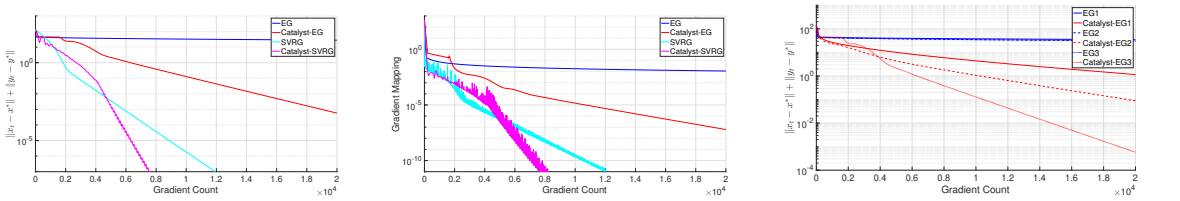
FIGURE 2.1: Comparison of GDA and Catalyst-GDA on the 2-dimensional example.

We apply both GDA and Catalyst-GDA to a minimax problem defined as: $\min_x \max_y f(x,y) = F(x) + 10xy$. The function f has a single saddle point and stationary point located at $(0,0)$. Given it is 2-smooth, we select $\tau = 1$ for Catalyst-GDA. Figure 2.1 contrasts the performances of GDA and Catalyst-GDA, both utilizing a stepsize of 0.01 for x and y, based on the gradient norm and trajectory. Notably, while GDA struggles to converge, Catalyst-GDA converges without any issues.

2.4.2 *Experiments on Simulated Datasets.*

We explore a wireless communication problem as presented in [Boyd et al., 2004]. Given n communication channels, each with signal power $p \in \mathbb{R}^n$ and noise power $\sigma \in \mathbb{R}^n$, the capacity of the i-th channel is proportional to $\log(1 + \beta_i p_i/(\sigma_i^0 + \sigma_i))$, where both $\beta_i > 0$ and σ_i^0 are predefined constants. Our objective is to optimize the channel capacity in the face of noise chosen by an adversary, as discussed in [Garnaev and Trappe, 2009]. This scenario can be modeled as a minimax problem:

$$\min_p \max_\sigma \; f(p,\sigma) := -\sum_{i=1}^n \log\left(1 + \frac{\beta_i p_i}{\sigma_i^0 + \sigma_i}\right) + \frac{\lambda}{2}\|p\|^2 - \frac{\nu}{2}\|\sigma\|^2, \tag{2.16}$$



(a) Distance to limit point (b) Gradient mapping (c) Comparison of EG and Catalyst-EG

FIGURE 2.2: SC-SC experiment on power allocation with same stepsizes for EG and Catalyst-EG

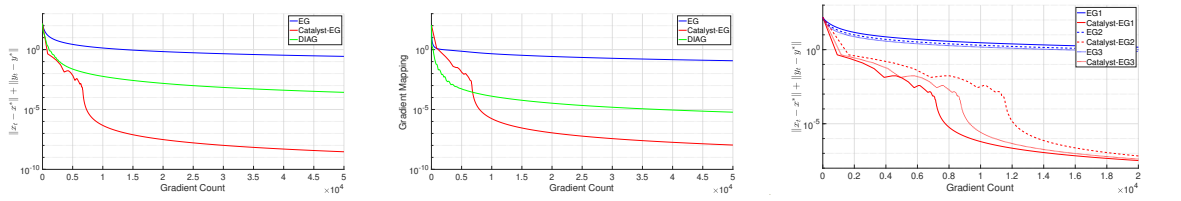
$$\text{subject to } \mathbf{1}^\top \sigma = N, p \geq 0, \sigma \geq 0,$$

where $N > 0$ represents the total noise power constraint, and λ and ν serve as regularization parameters.

SC-SC SETTING. For our experiments, we set $\beta = \mathbf{1}, \lambda = 3, \nu = 0.0001$, and uniformly sample $\sigma^0 \in \mathbb{R}^{1000}$ from $[0, 100]^{1000}$. The problem defined by (2.16) is strongly convex with respect to p and strongly concave in σ. Our primary focus is on comparing the performance of EG, Catalyst-EG, SVRG, and Catalyst-SVRG to understand their respective behaviors in SC-SC scenarios. For both EG and its Catalyst-enhanced version, we employ the same stepsizes for the primal and dual variables. Within the Catalyst framework, we employ the following as the stopping criterion for the subproblem: $\|x_t - \mathcal{P}_{\mathcal{X}}(x_t - \beta \nabla_x f(x_t, y_t))\|/\beta + \|y_t - \mathcal{P}_{\mathcal{Y}}(y_t + \beta \nabla_y f(x_t, y_t))\|/\beta$. The subroutine accuracy, ϵ^t, is controlled as $\max\{c\left(1 - 0.9\sqrt{q}\right)^t, \tilde{\epsilon}\}$, where c is a constant, $\tilde{\epsilon}$ is a predetermined threshold, and q is is set to be $q = \lambda/\nu$ as specified in Algorithm 1.

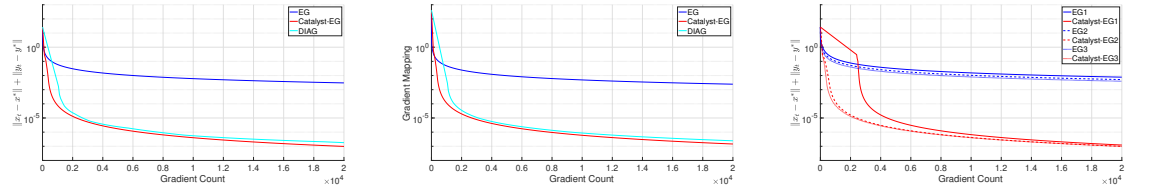
Figures 2.2(a) and 2.2(b) present results based on two error metrics with optimally-tuned stepsizes: (a) distance to the limit point, represented as $\|p_t - p^*\| + \|\sigma_t - \sigma^*\|$, and (b) the norm of gradient mapping, given by $\|\nabla_p f(p_t, \sigma_t))\| + \|\sigma_t - \mathcal{P}_{\Sigma}(\sigma_t + \beta \nabla_\sigma f(p_t, \sigma_t))\|/\beta$. To highlight the acceleration effects of Catalyst, Figure 2.2(c) compares the convergence results of EG and the Catalyst-EG subroutine, both using the same stepsizes ranging from 0.05 to 0.2. Our observations indicate that SVRG achieves convergence significantly faster than EG, and the Catalyst framework notably boosts the performance of both algorithms.

SC-C SETTING. For our experiments, we set $\beta = \mathbf{1}, \lambda = 3, \nu = 0$, and uniformly sample $\sigma^0 \in \mathbb{R}^{500}$ from the interval $[0, 10]^{500}$. The minimax problem, as defined in (2.16),is strongly-convex-concave. We compare the performance of EG with averaged iterates, Catalyst-EG, and DIAG. While EG with averaged iterates boasts a complexity of $\mathcal{O}(1/\epsilon)$ in the convex-concave setting [Nemirovski, 2004], both Catalyst-EG and DIAG are tailored for SC-C minimax optimization. Specifically, Catalyst-EG has a complexity of $\tilde{\mathcal{O}}(\ell/\sqrt{\mu\epsilon})$, whereas



(a) Distance to limit point (b) Gradient mapping (c) Comparison of EG and Catalyst-EG

FIGURE 2.3: SC-C experiment on power allocation with same stepsizes for EG and Catalyst-EG



(a) Distance to limit point (b) Gradient mapping (c) Comparison of EG and Catalyst-EG

FIGURE 2.4: SC-C experiment on distributionally robust logistic regression

DIAG's complexity is $\tilde{\mathcal{O}}\left(\ell^{\frac{3}{2}}/(\mu\sqrt{\epsilon})\right)$. Within the Catalyst framework, the subroutine accuracy, ϵ^t, is controlled as $\max\{c/t^8, \tilde{\epsilon}\}$, where c is a constant and $\tilde{\epsilon}$ is a predetermined threshold. In contrast, DIAG lacks a straightforward stopping criterion for its subroutine. For DIAG's subroutine, we employ the stopping criterion: $\|x_k - x_{k-1}\|^2 + \|y_k - y_{k-1}\|^2$, where k denotes the subroutine iterations.

Figures 2.3(a) and 2.3(b) showcase a comparison of these algorithms using optimally-tuned stepsizes, focusing on metrics like distance to the limit point and gradient mapping. Another figure contrasts the performances of EG and Catalyst-EG across three stepsizes: 1, 1.5, and 2. Our observations confirm that Catalyst-EG not only accelerates EG but also outperforms DIAG in performance.

2.4.3 *Distributionally Robust Logistic Regression*

We consider the distributionally robust logistic regression problem as presented in [Namkoong and Duchi, 2016]. This problem aims to minimize the worst-case loss over an ambiguity set centered around the empirical distribution, leading to a minimax formulation:

$$\min_{\theta} \max_{p \in \Delta_n} \sum_{i=1}^{n} -p_i \phi \left(y_i \log\left(\hat{y}\left(X_i\right)\right) + \left(1 - y_i\right) \log\left(1 - \hat{y}\left(X_i\right)\right)\right) \tag{2.17}$$

$$\text{subject to } \|p - \mathbf{1}/n\| \leq \rho,$$

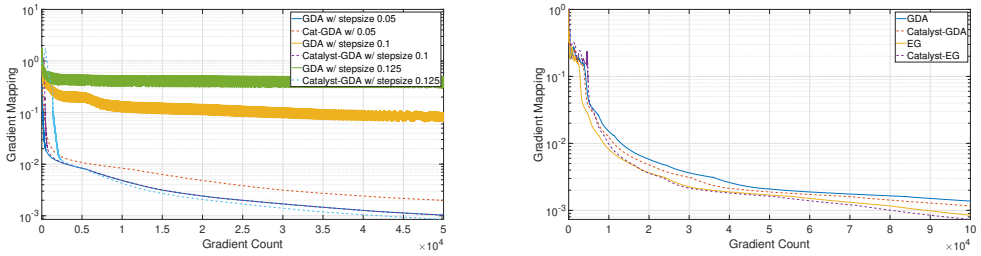
where θ represents the parameters of the classifier $\hat{y}(\cdot)$, $\phi : \mathbb{R} \to \mathbb{R}$ is a specified function, and (y, X) denotes the classification dataset.

SC-C SETTING. In this setting, we define $\hat{y}(x) = \frac{e^{\theta^\top x}}{1+e^{\theta^\top x}}$ and choose $\phi(z) = z$. The problem in (2.17) can then be reformulated as an SC-C minimax problem with L_2 regularization:

$$\min_{\theta} \max_{p \in \Delta_n} \sum_{i=1}^{n} p_i \log(1 + \exp(-y_i \theta^\top X_i)) + \frac{\lambda}{2}\|\theta\|^2 \text{ subject to } \|p - \mathbf{1}/n\| \leq \rho,$$

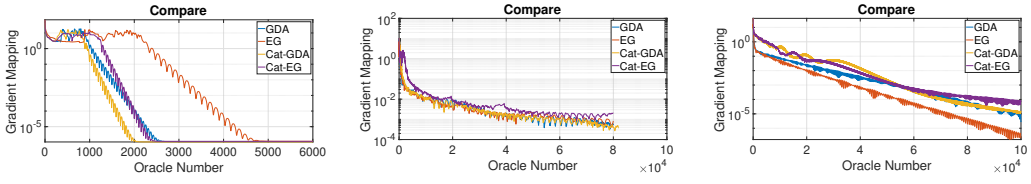
where λ denotes the regularization parameter.

For our experiments, we utilize the Wisconsin breast cancer dataset [Dua and Graff, 2017], which comprises 30 attributes and 569 samples. We allocate 80% of the data for training. Our comparative analysis includes EG, Catalyst-EG, and DIAG, with implementations consistent with those in Chapter 2.4.2. Figures 2.4(a) and 2.4(b) present the convergence behaviors of these algorithms using optimally-tuned stepsizes. In Figure 2.4(c), we compare the performances of EG and Catalyst-EG across three stepsizes: 0.2, 0.3, and 0.4. Notably, Catalyst-EG and DIAG exhibit comparable performances, both significantly outpacing EG.



(a) stepsize comparison (b) performance comparison

FIGURE 2.5: NC-C experiments on DRO on Breast Cancer Dataset with same stepsizes



(a) Toxic Dataset (b) breast cancer Dataset (c) Madelon Dataset

FIGURE 2.6: NC-C experiments on DRO with different stepsize

NC-C SETTING. For this setting, we define $\hat{y}(x) = \frac{e^{\theta^\top x}}{1+e^{\theta^\top x}}$ and set $\phi(z) = 2\log\left(1+\frac{z}{2}\right)$. We transition the constraint on p into a smoothed L_1 regularization relative to $p - \mathbf{1}/n$:

$$R_a(p-\mathbf{1}/n) = \sum_{i=1}^{d} \frac{1}{a}\left(\log\left(1+e^{a(p_i-1/n)}\right) + \log\left(1+e^{-a(p_i-1/n)}\right)\right),$$

with the regularization parameter λ designated as 0.01. This leads us to the following minimax formulation:

$$\min_{\theta}\max_{p\in\Delta_n} \sum_{i=1}^{n} p_i\phi\left[\log(1+\exp(-y_i\theta^\top X_i))\right] + \lambda R_a(p-\mathbf{1}/n). \tag{2.18}$$

Our comparison encompasses four algorithms: GDA, EG, Catalyst-GDA, and Catalyst-EG. We employ the mushrooms dataset from LIBSVM [Chang and Lin, 2011], drawing a random subset of 2000 samples for training, with each sample comprising 112 features. It's noteworthy that there is an absence of established theoretical outcomes for the vanilla EG in this context, while GDA boasts a complexity of $\mathcal{O}(\epsilon^{-6})$ [Lin et al., 2020a].

In Figure 2.6, we report the gradient mapping norm: $\|\nabla_\theta f(\theta_t, p_t))\| + \|p_t - \mathcal{P}_P(p_t + \beta\nabla_p f(\theta_t, p_t))\|/\beta$ in relation to the count of gradient evaluations, where P represents the feasible set for p. Observationally, both EG and GDA experience a marked enhancement in performance when integrated with the Catalyst framework.

2.5 APPENDIX

2.5.1 *Notations and Useful Lemmas*

Before we present the theorem and converge, we adopt the following notations.

- $\Psi(y) = \min_{x\in\mathcal{X}} f(x,y)$: the dual function;
- $\Phi(x) = \max_{y\in\mathcal{Y}} f(x,y)$: the primal function;
- $x^*(y) = \operatorname{argmin}_{x\in\mathcal{X}} f(x,y)$: the optimal x w.r.t y;
- $y^*(x) = \operatorname{argmax}_{y\in\mathcal{Y}} f(x,y)$: the optimal y w.r.t x;
- $\tilde{f}_t(x,y) = f(x,y) - \ell\|y - z_t\|^2$: the auxiliary problem $(\star)$ at iteration t;
- $\Psi_t(y) = \min_{x\in\mathcal{X}} f(x,y) - \frac{\tau}{2}\|y - z_t\|^2 = \Psi(y) - \frac{\tau}{2}\|y - z_t\|^2$: the dual function of the auxiliary problem $(\star)$;
- (x_t^*, y_t^*): the saddle point the auxiliary problem $(\star)$ at iteration t.

Lemma 2.5.1 (Lemma B.2 [Lin et al., 2020b])**.** *Consider a minimax problem* $\min_{\mathcal{X}} \max_{\mathcal{Y}} f(x,y)$. *Assume* $f(\cdot,y)$ *is* μ_x*-strongly convex for* $\forall y \in \mathcal{Y}$ *and* $f(x,\cdot)$ *is* μ_y*-strongly concave for* $\forall x \in \mathcal{X}$ *and* f *is* ℓ*-Lipschitz smooth. Then we have*

a) $y^*(x) = \operatorname{argmax}_{y\in\mathcal{Y}} f(x,y)$ *is* $\frac{\ell}{\mu_y}$*-Lipschitz;*

b) $\Phi(x) = \max_{y\in\mathcal{Y}} f(x,y)$ *is* $\frac{2\ell^2}{\mu_y}$*-Lipschitiz smooth and* μ_x*-strongly convex with* $\nabla\Phi(x) = \nabla_x f(x, y^*(x))$*;*

c) $x^*(y) = \operatorname{argmin}_{x\in\mathcal{X}} f(x,y)$ *is* $\frac{\ell}{\mu_x}$*-Lipschitz;*

d) $\Psi(y) = \min_{x\in\mathcal{X}} f(x,y)$ *is* $\frac{2\ell^2}{\mu_x}$*-Lipschitiz smooth and* μ_y*-strongly concave with* $\nabla\Psi(y) = \nabla_y f(x^*(y), y)$.

2.5.2 *Proofs for Chapter 2.2*

A. Proof for Chapter 2.2.1

Proof of Lemma 2.2.2

Proof. We construct a "ghost" point:

$$x_1 = \mathcal{P}_{\mathcal{X}}\left(x - \frac{1}{\beta}\nabla_x \tilde{f}([x]_\beta, [y]_\beta)\right), \quad y_1 = \mathcal{P}_{\mathcal{Y}}\left(y + \frac{1}{\beta}\nabla_y \tilde{f}([x]_\beta, [y]_\beta)\right).$$

From (x, y) to (x_1, y_1) is just one step of extra-gradient with stepsize $\frac{1}{\beta}$. According to [Nemirovski, 2004] or Section 4.5 of [Bubeck, 2017], we have, $\forall \bar{x} \in \mathcal{X}, \bar{y} \in \mathcal{Y}$,

$$\begin{aligned} &\nabla_x \tilde{f}([x]_\beta, [y]_\beta)^T([x]_\beta - \bar{x}) - \nabla_y \tilde{f}([x]_\beta, [y]_\beta)^T([y]_\beta - \bar{y}) \\ &\leq \frac{\beta}{2}[(\|x - \bar{x}\|^2 + \|y - \bar{y}\|^2) - (\|x_1 - \bar{x}\|^2 + \|y_1 - \bar{y}\|^2)]. \end{aligned} \tag{2.19}$$

1. Because $\tilde{f}$ is convex in x and concave in y, we have

$$\begin{aligned} &\mathrm{gap}_{\tilde{f}}([z]_\beta) \\ &= \tilde{f}([x]_\beta, [y]_\beta) - \min_{x \in \mathcal{X}} \tilde{f}(x, [y]_\beta) + \max_{y \in \mathcal{Y}} \tilde{f}([x]_\beta, y) - \tilde{f}([x]_\beta, [y]_\beta) \\ &\leq \nabla_x \tilde{f}([x]_\beta, [y]_\beta)^T([x]_\beta - x^*([y]_\beta)) - \nabla_y \tilde{f}([x]_\beta, [y]_\beta)^T([y]_\beta - y^*([x]_\beta)) \\ &\leq \frac{\beta}{2}\left[(\|x - x^*([y]_\beta)\|^2 + \|y - y^*([x]_\beta)\|^2) - (\|x_1 - x^*([y]_\beta)\|^2 + \|y_1 - y^*([x]_\beta)\|^2)\right] \\ &\leq \beta[\|x - x^*\|^2 + \|x^* - x^*([y]_\beta)\|^2 + \|y - y^*\|^2 + \|y^* - y^*([x]_\beta)\|^2] \qquad (2.20) \\ &\leq \beta[\|x - x^*\|^2 + \|y - y^*\|^2] + \frac{\beta\tilde{\ell}^2}{\tilde{\mu}^2}[\|[x]_\beta - x^*\|^2 + \|[y]_\beta - y^*\|^2] \\ &\leq \left(\beta + \frac{2\beta\tilde{\ell}^2}{\tilde{\mu}^2}\right)[\|x - x^*\|^2 + \|y - y^*\|^2] + \frac{2\beta\tilde{\ell}^2}{\tilde{\mu}^2}[\|[x]_\beta - x\|^2 + \|[y]_\beta - y\|^2], \qquad (2.21) \end{aligned}$$

where in the second inequality we apply (2.19), in the third and last inequalities we use Young's inequality, and in the fourth inequality we use $\|x^* - x^*([y]_\beta)\| = \|x^*(y^*) - x^*([y]_\beta)\| \leq \frac{\tilde{\ell}}{\tilde{\mu}}\|[y]_\beta - y^*\|$ (and similarly for $\|y^* - y^*([x]_\beta)\|$, by Lemma 2.5.1). From Lemma 3.1 and Proposition 3.2 in [Tseng, 1995], we have

$$\begin{aligned} \|[x]_\beta - x\|^2 + \|[y]_\beta - y\|^2 &\leq \frac{1}{(1 - \tilde{\ell}/\beta)^2}[\|x - x_1\|^2 + \|y - y_1\|^2] \\ &\leq \frac{2}{(1 - \tilde{\ell}/\beta)^3}[\|x - x^*\|^2 + \|y - y^*\|^2]. \end{aligned} \tag{2.22}$$

Combining with (2.21), we have

$$\mathrm{gap}_{\tilde{f}}([z]_\beta) \leq \left(\beta + \frac{2\beta\tilde{\ell}^2}{\tilde{\mu}^2} + \frac{4\beta\tilde{\ell}^2}{\tilde{\mu}^2(1 - \tilde{\ell}/\beta)^3}\right)[\|x - x^*\|^2 + \|y - y^*\|^2]. \tag{2.23}$$

Then again from (2.19), for any $\bar{y} \in \mathcal{Y}$, we have

$$
\begin{aligned}
&\nabla_x \tilde{f}([x]_\beta, [y]_\beta)^T([x]_\beta - x^*([y]_\beta)) - \nabla_y \tilde{f}([x]_\beta, [y]_\beta)^T([y]_\beta - \bar{y}) \\
&\leq \frac{\beta}{2}[(\|x - x^*([y]_\beta)\|^2 + \|y - \bar{y}\|^2) - (\|x_1 - x^*([y]_\beta)\|^2 + \|y_1 - \bar{y}\|^2)] \\
&\leq \frac{\beta}{2}\|x - x^*([y]_\beta)\|^2 + \frac{\beta}{2}[\|y - \bar{y}\|^2 - \|y_1 - \bar{y}\|^2] \\
&\leq \frac{\beta}{2}\|x - x^*([y]_\beta)\|^2 + \frac{\beta}{2}\|y - y_1\|\|y - \bar{y} + y_1 - \bar{y}\| \\
&\leq \left(\beta + \frac{2\beta\tilde{\ell}^2}{\tilde{\mu}^2} + \frac{4\beta\tilde{\ell}^2}{\tilde{\mu}^2(1 - \tilde{\ell}/\beta)^3}\right)[\|x - x^*\|^2 + \|y - y^*\|^2] + \beta\mathcal{D}_{\mathcal{Y}}[\|y - y^*\| + \|y_1 - y^*\|] \\
&\leq \left(\beta + \frac{2\beta\tilde{\ell}^2}{\tilde{\mu}^2} + \frac{4\beta\tilde{\ell}^2}{\tilde{\mu}^2(1 - \tilde{\ell}/\beta)^3}\right)[\|x - x^*\|^2 + \|y - y^*\|^2] + 2\beta\mathcal{D}_{\mathcal{Y}}[\|x - x^*\| + \|y - y^*\|],
\end{aligned}
$$

where in the fourth inequality, we bound $\|x - x^*([y]_\beta)\|^2$ the same way as we did from (2.20) to (2.22), and in the last inequality we use $\|z - z^*\| \leq \|z_1 - z^*\|$ (Proposition 3.2 in [Tseng, 1995]). By noting that

$$
\nabla_x \tilde{f}([x]_\beta, [y]_\beta)^T([x]_\beta - x^*([y]_\beta)) \geq 0,
$$

we reach our conclusion.

2. Theorem 3.1 of [Pang, 1987] shows the relationship between $\|x - x^*\| + \|y - y^*\|$ and $\|x - [x]_\beta\| + \|y - [y]_\beta\|$ in the case $\beta = 1$. The proof can be extended to the following general case:

$$
\|x - x^*\| + \|y - y^*\| \leq \frac{\beta + \tilde{\ell}}{\tilde{\mu}}\left[\|x - [x]_\beta\| + \|y - [y]_\beta\|\right].
$$

The second inequality we want to show is just equation (2.22). To show the third inequality, since $\Phi(x)$ is μ_x strongly-convex and differentiable, we have

$$
\Phi(x) \geq \Phi(x^*) + \langle \nabla\Phi(x^*), x - x^* \rangle + \frac{\mu_x}{2}\|x - x^*\|^2 \geq \Phi(x^*) + \frac{\mu_x}{2}\|x - x^*\|^2.
$$

Similarly, because $\Psi(y)$ is μ_y strongly-concave and differentiable, we have $\Psi(y^*) - \Psi(y) \geq \frac{\mu_y}{2}\|y - y^*\|^2$. □

B. Outer-loop complexity

Proof of Theorem 2.2.5

Proof. Because $\tilde{f}(x_t, y) \triangleq f(x_t, y) - \frac{\tau}{2}\|y - z_t\|^2$ is τ-strongly-concave in y, we have, $\forall y \in \mathcal{Y}$,

$$f(x_t, y_t) - \frac{\tau}{2}\|y_t - z_t\|^2 - [f(x_t, y) - \frac{\tau}{2}\|y - z_t\|^2] \geq \frac{1}{2}\tau\|y - y_t\|^2 + \nabla_y \tilde{f}(x_t, y_t)^T(y_t - y).$$

With stopping criterion of the auxiliary problem (2.2), we have

$$f(x_t, y_t) - f(x_t, y) \geq \frac{1}{2}\tau\|y - y_t\|^2 + \frac{\tau}{2}\|y_t - z_t\|^2 - \frac{\tau}{2}\|y - z_t\|^2 - \epsilon^t. \tag{2.24}$$

Choose $y = \alpha_t\tilde{y} + (1 - \alpha_t)y_{t-1}$ in (2.24), where $\tilde{y}$ is an arbitrary vector in $\mathcal{Y}$, then

$$\begin{aligned} f(x_t, \tilde{y}) - f(x_t, y_t) \leq (1 - \alpha_t)[f(x_t, \tilde{y}) - f(x_t, y_{t-1})] - \frac{\tau}{2}\alpha_t^2(\|v_t - \tilde{y}\|^2 - \|v_{t-1} - \tilde{y}\|^2) \\ - \frac{\tau}{2}\|y_t - z_t\|^2 + \epsilon^t. \end{aligned} \tag{2.25}$$

Note that

$$\begin{aligned} & f(x_t, \tilde{y}) - f(x_t, y_{t-1}) \\ &= f(x_{t-1}, \tilde{y}) - f(x_{t-1}, y_{t-1}) + f(x_{t-1}, y_{t-1}) - f(x_t, y_{t-1}) + f(x_t, \tilde{y}) - f(x_{t-1}, \tilde{y}) \\ &\leq f(x_{t-1}, \tilde{y}) - f(x_{t-1}, y_{t-1}) + f(x_t, \tilde{y}) - f(x_{t-1}, \tilde{y}) + \epsilon^{t-1}, \end{aligned} \tag{2.26}$$

where the inequality follows because $f(x_t, y_t) - \min_{x \in \mathcal{X}} f(x, y_t) \leq \epsilon^t$. Plugging this back to (2.25) and rearranging,

$$\begin{aligned} & \frac{1}{\alpha_t^2}[f(x_t, \tilde{y}) - f(x_t, y_t)] + \frac{\tau}{2}\|v_t - \tilde{y}\|^2 \\ \leq & \frac{1 - \alpha_t}{\alpha_t^2}[f(x_{t-1}, \tilde{y}) - f(x_{t-1}, y_{t-1})] + \frac{\tau}{2}\|v_{t-1} - \tilde{y}\|^2 + \\ & \frac{1 - \alpha_t}{\alpha_t^2}[f(x_t, \tilde{y}) - f(x_{t-1}, \tilde{y})] + \frac{1 - \alpha_t}{\alpha_t^2}\epsilon^{t-1} + \frac{1}{\alpha_t^2}\epsilon^t. \end{aligned} \tag{2.27}$$

Using the update rule for sequence $\{\alpha\}_t$, for $t > 1$ we have

$$\begin{aligned} & \frac{1}{\alpha_t^2}[f(x_t, \tilde{y}) - f(x_t, y_t)] + \frac{\tau}{2}\|v_t - \tilde{y}\|^2 \\ \leq & \frac{1}{\alpha_{t-1}^2}[f(x_{t-1}, \tilde{y}) - f(x_{t-1}, y_{t-1})] + \frac{\tau}{2}\|v_{t-1} - \tilde{y}\|^2 + \\ & \frac{1}{\alpha_{t-1}^2}[f(x_t, \tilde{y}) - f(x_{t-1}, \tilde{y})] + \frac{1}{\alpha_{t-1}^2}\epsilon^{t-1} + \frac{1}{\alpha_t^2}\epsilon^t. \end{aligned} \tag{2.28}$$

Iterating this inequality results in

$$\begin{aligned}
&\frac{1}{\alpha_t^2}[f(x_t,\tilde{y})-f(x_t,y_t)]+\frac{\tau}{2}\|v_t-\tilde{y}\|^2\\
\le&\frac{1}{\alpha_1^2}[f(x_1,\tilde{y})-f(x_1,y_1)]+\frac{\tau}{2}\|v_1-\tilde{y}\|^2+\\
&\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}[f(x_t,\tilde{y})-f(x_{t-1},\tilde{y})]+\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}\epsilon^{t-1}+\sum_{t=2}^{T}\frac{1}{\alpha_t^2}\epsilon^t\\
=&f(x_1,\tilde{y})-f(x_1,y_1)+\frac{\tau}{2}\|v_1-\tilde{y}\|^2+\\
&\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}[f(x_t,\tilde{y})-f(x_{t-1},\tilde{y})]+\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}\epsilon^{t-1}+\sum_{t=2}^{T}\frac{1}{\alpha_t^2}\epsilon^t, \qquad (2.29)
\end{aligned}$$

where we use $\alpha_1=1$. Applying (2.27) with $t=1$ (note $\alpha_1=1$), we have

$$f(x_1,\tilde{y})-f(x_1,y_1)+\frac{\tau}{2}\|v_1-\tilde{y}\|^2\le\frac{\tau}{2}\|y_0-\tilde{y}\|^2+\epsilon^1. \qquad (2.30)$$

Combining with (2.29),

$$\begin{aligned}
&\frac{1}{\alpha_T^2}[f(x_T,\tilde{y})-f(x_T,y_T)]+\frac{\tau}{2}\|v_T-\tilde{y}\|^2\\
\le&\frac{\tau}{2}\|y_0-\tilde{y}\|^2+\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}[f(x_t,\tilde{y})-f(x_{t-1},\tilde{y})]+\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}\epsilon^{t-1}+\sum_{t=1}^{T}\frac{1}{\alpha_t^2}\epsilon^t\\
\le&\frac{\tau}{2}\|y_0-\tilde{y}\|^2+\frac{1}{\alpha_{T-1}^2}f(x_T,\tilde{y})-\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}}f(x_{t-1},\tilde{y})+\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}\epsilon^{t-1}+\sum_{t=1}^{T}\frac{1}{\alpha_t^2}\epsilon^t,
\end{aligned}$$

where in the last inequality we use $\frac{1}{\alpha_t^2}-\frac{1}{\alpha_{t-1}^2}=\frac{1}{\alpha_t}$. Rearranging,

$$\begin{aligned}
&\frac{\tau}{2}\|y_0-\tilde{y}\|^2+\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}^2}\epsilon^{t-1}+\sum_{t=1}^{T}\frac{1}{\alpha_t^2}\epsilon^t\\
\ge&\frac{1}{\alpha_T^2}[f(x_T,\tilde{y})-f(x_T,y_T)]+\frac{\tau}{2}\|v_T-\tilde{y}\|^2-\frac{1}{\alpha_{T-1}^2}f(x_T,\tilde{y})+\sum_{t=2}^{T}\frac{1}{\alpha_{t-1}}f(x_{t-1},\tilde{y})\\
\ge&\sum_{t=1}^{T}\frac{1}{\alpha_t}f(x_t,\tilde{y})-\frac{1}{\alpha_T^2}f(x_T,y_T)\\
\ge&\sum_{m=1}^{T}\frac{1}{\alpha_m}f\left(\sum_{t=1}^{T}\frac{1/\alpha_t}{\sum_{k=1}^{T}1/\alpha_k}x_t,\tilde{y}\right)-\frac{1}{\alpha_T^2}f(x_T,y_T)\\
\ge&\sum_{m=1}^{T}\frac{1}{\alpha_m}f\left(\sum_{t=1}^{T}\frac{1/\alpha_t}{\sum_{k=1}^{T}1/\alpha_k}x_t,\tilde{y}\right)-\frac{1}{\alpha_T^2}f(\tilde{x},y_T)-\frac{1}{\alpha_T^2}\epsilon^T,\quad\forall\tilde{x}\in\mathcal{X},
\end{aligned}$$

where in the third inequality we use the convexity of $f(\cdot, \tilde{y})$, and in the last inequality we use $f(x_t, y_t) - \min_{x\in\mathcal{X}} f(x, y_t) \leq \epsilon^t$. Note that

$$\sum_{m=1}^{t} \frac{1}{\alpha_m} = \frac{1}{\alpha_1} + \left(\frac{1}{\alpha_2^2} - \frac{1}{\alpha_1^2}\right) + \left(\frac{1}{a_3^2} - \frac{1}{\alpha_2^2}\right) + ... + \left(\frac{1}{\alpha_t^2} - \frac{1}{\alpha_{t-1}^2}\right) = \frac{1}{\alpha_t^2}. \tag{2.31}$$

Therefore

$$f(\bar{x}_T, \tilde{y}) - f(\tilde{x}, y_T) \leq a_T^2 \left[\frac{\tau}{2}\|y_0 - \tilde{y}\|^2 + 2\sum_{t=1}^{T} \frac{1}{\alpha_t^2}\epsilon^t\right], \quad \forall \tilde{x} \in \mathcal{X}, \tilde{y} \in \mathcal{Y}, \tag{2.32}$$

which directly implies

$$\text{gap}_f(\bar{x}_T, y_T) \leq \alpha_T^2 \left[\frac{\tau}{2}\mathcal{D}_{\mathcal{Y}}^2 + 2\sum_{t=1}^{T} \frac{1}{\alpha_t^2}\epsilon^t\right]. \tag{2.33}$$

By choosing $\epsilon^t = \frac{\tau(\rho-1)\mathcal{D}_{\mathcal{Y}}^2\alpha_t^2}{4(t+1)^\rho}$ with $\rho > 1$,

$$\sum_{t=1}^{T} \frac{1}{\alpha_t^2}\epsilon^t = \frac{\tau\mathcal{D}_{\mathcal{Y}}^2(\rho-1)}{4}\sum_{t=1}^{T} \frac{1}{(t+1)^\rho} \leq \frac{\tau\mathcal{D}_{\mathcal{Y}}^2(\rho-1)}{4}\int_1^\infty \frac{1}{t^\rho}\partial t = \frac{\mathcal{D}_{\mathcal{Y}}^2\tau}{4}, \tag{2.34}$$

therefore,

$$\text{gap}_f(\bar{x}_T, y_T) \leq \alpha_T^2\tau\mathcal{D}_{\mathcal{Y}}^2. \tag{2.35}$$

□

Before we prove Theorem 2.2.5, we present a lemma from [Lin et al., 2017]. Algorithm 1 can be considered as applying Catalyst for strongly-convex minimization in [Lin et al., 2017] to the function $-\Psi(y) = -\min_{x\in\mathcal{X}} f(x, y)$. The following lemma captures the convergence of Catalyst framework in minimization, which we present in Algorithm 3.

Lemma 2.5.2 ([Lin et al., 2017]). *Assume function h is μ-strongly convex. Define $A_t = \prod_{i=1}^{t}(1-\alpha_i)$, $\eta_t = \frac{\alpha_t - q}{1-q}$ and a sequence $\{v_t\}_t$ with $v_0 = x_0$ and $v_t = x_{t-1} + \frac{1}{\alpha_t}(x_t - x_{t-1})$ for $t > 1$. We construct a potential function: $S_t = h(x_t) - h(x^*) + \frac{\eta_{t+1}\alpha_{t+1}\tau}{2(1-\alpha_{t+1})}\|x^* - v_t\|^2$, where x^* is the optimal solution. After running Algorithm 3 for T iterations, we have*

$$\frac{1}{A_T}S_T \leq \left(\sqrt{S_0} + 2\sum_{t=1}^{T}\sqrt{\frac{\epsilon^t}{A_t}}\right)^2. \tag{2.37}$$

Proof of Theorem 2.2.5

Algorithm 3 `Catalyst for Strongly-Convex Minimization`

1: Input: function h, initial point x_0, strong-convexity constant μ, parameter $\tau > 0$
2: Initialization: $q = \frac{\mu}{\mu+\tau}$, $z_1 = x_0$, $\alpha_1 = \sqrt{q}$.
3: **for all** $t = 1, 2, ..., T$ **do**
4: Find an inexact solution x_t to the following problem with algorithm $\mathcal{M}$

$$\min_{x \in \mathcal{X}} \tilde{h}_t(x) := \left[h(x) + \frac{\tau}{2}\|x - z_t\|^2 \right]$$

such that

$$\tilde{h}_t(x_t) - \min_{x \in \mathcal{X}} \tilde{h}_t(x) \leq \epsilon^t. \tag{2.36}$$

5: Choose $\alpha_{t+1} \in [0, 1]$ such that $\alpha_{t+1}^2 = (1 - \alpha_{t+1})\alpha_t^2 + q\alpha_{t+1}$.
6: $z_{t+1} = x_t + \beta_t(x_t - x_{t-1})$ where $\beta_t = \frac{\alpha_t(1-\alpha_t)}{\alpha_t^2 + \alpha_{t+1}}$.
7: **end for**
8: Output: x_T.

Proof. First, we will see that sequences $\{z_t\}_t$ in Algorithm 1 are built in the same way as in Algorithm 3. Note that by the definition of z_t,

$$\begin{aligned} z_t = \eta_t v_{t-1} + (1 - \eta_t) y_{t-1} &= \eta_t \left[y_{t-2} + \frac{1}{\alpha_{t-1}}(y_{t-1} - y_{t-2}) \right] + (1 - \eta_t) y_{t-1} \\ &= y_{t-1} + \eta_t \left(\frac{1}{\alpha_{t-1}} - 1 \right) (y_{t-1} - y_{t-2}). \end{aligned} \tag{2.38}$$

Furthermore,

$$\begin{aligned} \eta_t \left(\frac{1}{\alpha_{t-1}} - 1 \right) &= \frac{\alpha_{t-1} - q}{1 - q} \left(\frac{1}{\alpha_{t-1}} - 1 \right) = \frac{(1 - \alpha_t)\alpha_{t-1}^2}{\alpha_t(1 - q)} \cdot \frac{1 - \alpha_{t-1}}{\alpha_{t-1}} = \frac{(1 - \alpha_t)\alpha_{t-1}^2(1 - \alpha_{t-1})}{\alpha_t - \alpha_t q} \\ &= \frac{(1 - \alpha_t)\alpha_{t-1}^2(1 - \alpha_{t-1})}{\alpha_t - \alpha_t^2 + (1 - \alpha_t)\alpha_{t-1}^2} = \frac{\alpha_{t-1}(1 - \alpha_{t-1})}{\alpha_{t-1}^2 + \alpha_t} =: \beta_{t-1}, \end{aligned} \tag{2.39}$$

where in the second and fourth equality we use the update rule of $\{\alpha_t\}_t$.

The dual function of f is $\Psi(y) = \min_{x \in \mathcal{X}} f(x, y)$. Define $\Psi_t(y) = \min_{x \in \mathcal{X}} f(x, y) - \frac{\tau}{2}\|y - z_t\|^2 = \Psi(y) - \frac{\tau}{2}\|y - z_t\|^2$, and $y_t^* = \arg\min_{y \in \mathcal{Y}} \Psi_t(y)$. The auxiliary problem $(\star)$ can be considered as $\max_{y \in \mathcal{Y}} \Psi_t(y)$. The stopping criterion (2.3) implies that $\max_{y \in \mathcal{Y}} \Psi_t(y) - \Psi_t(y_t) \leq \epsilon^t$. Therefore Algorithm 1 can be considered as applying Algorithm 3 to $-\Psi(y)$

and Lemma 2.5.2 can guarantee the convergence of the dual function. Define $S_t = \Psi(y^*) - \Psi(y_t) + \frac{\eta_{t+1}\alpha_{t+1}\tau}{2(1-\alpha_{t+1})}\|y^* - v_t\|^2$, and Lemma 2.5.2 gives rise to

$$\frac{1}{A_T}S_T \le \left(\sqrt{S_0} + 2\sum_{t=1}^{T}\sqrt{\frac{\epsilon^t}{A_t}}\right)^2. \tag{2.40}$$

When $\alpha_1 = \sqrt{q}$, it is easy to check that $\alpha_t = \sqrt{q}$, $A_t = (1-\sqrt{q})^t$ and

$$\frac{\eta_1\alpha_1\tau}{2(1-\alpha_1)} = \frac{\sqrt{q}-q}{1-q}\frac{\sqrt{q}\tau}{2(1-\sqrt{q})} = \frac{\sqrt{q}-q}{\tau/(\mu_y+\tau)}\frac{\sqrt{q}\tau}{2(1-\sqrt{q})} = \frac{q(\mu_y+\tau)}{2} = \frac{\mu_y}{2}.$$

Therefore $S_0 = \Psi(y^*) - \Psi(y_0) + \frac{\mu_y}{2}\|y^* - y_0\|^2 \le 2(\Psi(y^*) - \Psi(y_0))$. Then with $\epsilon^t = \frac{\sqrt{2}}{4}(1-\rho)^t \operatorname{gap}_f(x_0, y_0)$, we have

$$\begin{aligned}
&\text{Right-hand side of (2.40)}\\
&\le \left(\sqrt{2(\Psi(y^*) - \Psi(y_0))} + \sum_{t=1}^{T}\sqrt{2\left(\frac{1-\rho}{1-\sqrt{q}}\right)^t \operatorname{gap}_f(x_0, y_0)}\right)^2\\
&\le 2\left(1 + \sum_{t=1}^{T}\left(\sqrt{\frac{1-\rho}{1-\sqrt{q}}}\right)^t\right)^2 \operatorname{gap}_f(x_0, y_0)\\
&\le 2\left(\frac{\left(\sqrt{\frac{1-\rho}{1-\sqrt{q}}}\right)^{T+1}}{\sqrt{\frac{1-\rho}{1-\sqrt{q}}} - 1}\right)^2 \operatorname{gap}_f(x_0, y_0) \le 2\left(\frac{\sqrt{\frac{1-\rho}{1-\sqrt{q}}}}{\sqrt{\frac{1-\rho}{1-\sqrt{q}}} - 1}\right)^2 \left(\frac{1-\rho}{1-\sqrt{q}}\right)^T \operatorname{gap}_f(x_0, y_0).
\end{aligned}$$

Plugging back into (2.40),

$$\begin{aligned}
S_T &\le 2\left(\frac{1}{\sqrt{1-\rho} - \sqrt{1-\sqrt{q}}}\right)^2 (1-\rho)^{T+1} \operatorname{gap}_f(x_0, y_0)\\
&\le \frac{8}{(\sqrt{q}-\rho)^2}(1-\rho)^{T+1} \operatorname{gap}_f(x_0, y_0),
\end{aligned} \tag{2.41}$$

where the second inequality is due to $\sqrt{1-x} + \frac{x}{2}$ is decreasing in $[0, 1]$. Note that

$$\begin{aligned}
\|x_T - x^*\|^2 &\le 2\|x_T - x^*(y_T)\|^2 + 2\|x^*(y_T) - x^*(y^*)\|^2\\
&\le \frac{4}{\mu_x}[f(x_T, y_T) - f(x^*(y_T), y_T)] + 2\left(\frac{\ell}{\mu_x}\right)^2 \|y_T - y^*\|^2\\
&\le \frac{4}{\mu_x}\epsilon^T + 2\left(\frac{\ell}{\mu_x}\right)^2 \|y_T - y^*\|^2.
\end{aligned} \tag{2.42}$$

where in the second inequality we use Lemma 2.5.1. Then,

$$\|x_T - x^*\|^2 + \|y_T - y^*\|^2 \leq \left[2\left(\frac{\ell}{\mu_x}\right)^2 + 1\right]\|y_T - y^*\|^2 + \frac{4}{\mu_x}\epsilon^T. \tag{2.43}$$

Because $\|y_T - y^*\|^2 \leq \frac{2}{\mu_y}[\Psi(y^*) - \Psi(y_T)] \leq \frac{2}{\mu_y}S_T$, we finish the proof by plugging in (2.41) and definition of ϵ^t and get

$$\|x_T - x^*\|^2 + \|y_T - y^*\|^2 \leq \left\{\left[2\left(\frac{\ell}{\mu_x}\right)^2 + 1\right]\frac{16(1-\rho)}{\mu_y(\sqrt{q}-\rho)^2} + \frac{\sqrt{2}}{\mu_x}\right\}(1-\rho)^T \operatorname{gap}_f(x_0, y_0).$$

□

C. Inner-loop complexity

Proof of Lemma 2.2.8

Proof. We split the proof into case $t = 1$ and case t > 1.
Case $t = 1$**:** Note that $z_1 = y_0$ and therefore the auxiliary problem at iteration 1 is

$$\min_{x\in\mathcal{X}}\max_{y\in\mathcal{Y}}\left[\tilde{f}_1(x,y) \triangleq f(x,y) - \frac{\tau}{2}\|y - y_0\|^2\right]. \tag{2.44}$$

As $x_1^* = \operatorname{argmin}_{x\in\mathcal{X}}\tilde{f}_1(x, y_1^*) = \operatorname{argmin}_{x\in\mathcal{X}} f(x, y_1^*)$ and $x^* = \operatorname{argmin}_{x\in\mathcal{X}} f(x, y^*)$, by Lemma 2.5.1, we have $\|x^* - x_1^*\| \leq \frac{\ell}{\mu_x}\|y^* - y_1^*\|$. Then we further have

$$\begin{aligned}\|x_0 - x_1^*\|^2 + \|y_0 - y_1^*\|^2 &\leq 2\|x_0 - x^*\|^2 + 2\|x^* - x_1^*\|^2 + \|y_0 - y_1^*\|^2 \\ &\leq 2\|x_0 - x^*\|^2 + \frac{2\ell^2}{\mu_x^2}\|y^* - y_1^*\|^2 + \|y_0 - y_1^*\|^2 \\ &\leq 2\|x_0 - x^*\|^2 + \left(\frac{2\ell^2}{\mu_x^2} + 1\right)\mathcal{D}_{\mathcal{Y}}^2.\end{aligned} \tag{2.45}$$

Case $t > 1$**:** Since $f(\cdot, y)$ is μ-strongly convex, we have

$$\|x^*(y_{t-1}^*) - x^*(y_t^*)\|^2 \leq \left(\frac{\ell}{\mu_x}\right)^2\|y_t^* - y_{t-1}^*\|^2. \tag{2.46}$$

Define $\Phi_t(x) = \max_{y\in\mathcal{Y}} f(x,y) - \frac{\tau}{2}\|y - z_t\|^2$ to be the primal function of the the auxiliary problem. Because it is μ_x-strongly convex,

$$\|x_{t-1} - x^*(y_{t-1}^*)\|^2 \leq \frac{2}{\mu_x}[g_t(x_{t-1}) - g_t^*] \leq \frac{2\epsilon^{t-1}}{\mu_x}. \tag{2.47}$$

We further have

$$\begin{aligned}\|x_{t-1}-x_t^*\|^2 &\le 2\|x_{t-1}-x^*(y_{t-1}^*)\|^2+2\|x^*(y_{t-1}^*)-x^*(y_t^*)\|^2\\ &\le \frac{4\epsilon^{t-1}}{\mu_x}+2\left(\frac{\ell}{\mu_x}\right)^2\|y_t^*-y_{t-1}^*\|^2. \end{aligned}\tag{2.48}$$

Define $\Psi_t(y)=\min_{x\in\mathcal{X}} f(x,y)-\frac{\tau}{2}\|y-z_t\|^2$ to be the dual function of the the auxiliary problem, which is $(\mu_y+\tau)$-strongly concave. Define $\Psi_t^*=\max_{y\in\mathcal{Y}}\Psi_t(y)$. We have

$$\begin{aligned}\|\tilde{y}_0-y_t^*\|^2=2\|y_{t-1}-y_{t-1}^*\|^2+2\|y_{t-1}^*-y_t^*\|^2 &\le \frac{4}{\mu_y+\tau}[\Psi_t^*-\Psi_t(y_{t-1})]+2\|y_{t-1}^*-y_t^*\|^2\\ &\le \frac{4\epsilon^{t-1}}{\mu_y+\tau}+2\|y_{t-1}^*-y_t^*\|^2. \end{aligned}\tag{2.49}$$

Combining with (2.48), we have

$$\|\tilde{x}_0-x_t^*\|^2+\|\tilde{y}_0-y_t^*\|^2\le\left(\frac{4}{\mu_x}+\frac{4}{\mu_y+\tau}\right)\epsilon^{t-1}+\left[2\left(\frac{\ell}{\mu_x}\right)^2+2\right]\|y_{t-1}^*-y_t^*\|^2.\tag{2.50}$$

We finish the proof by noting that $\|y_t^*-y_{t-1}^*\|\le\mathcal{D}_\mathcal{Y}$ and $\epsilon^t\le\frac{\tau(\rho-1)\mathcal{D}_\mathcal{Y}^2}{4},\forall t$.

□

Proof of Lemma 2.2.7

Proof. We split the proof into case $t=1$ and case t > 1.
Case $t=1$**:** Following (2.45) in the proof of Lemma 2.2.8, we have

$$\begin{aligned}\|x_0-x_1^*\|^2+\|y_0-y_1^*\|^2 \le& 2\|x_0-x^*\|^2+\frac{2\ell^2}{\mu_x^2}\|y^*-y_1^*\|^2+\|y_0-y_1^*\|^2\\ \le& 2\|x_0-x^*\|^2+\frac{4\ell^2}{\mu_x^2}\|y_0-y^*\|^2+\left(\frac{4\ell^2}{\mu_x^2}+1\right)\|y_0-y_1^*\|^2\\ \le& \frac{8\ell^2}{\mu_x^2\min\{\mu_x,\mu_y\}}\,\mathrm{gap}_f(x_0,y_0)+\left(\frac{4\ell^2}{\mu_x^2}+1\right)\|y_0-y_1^*\|^2, \end{aligned}\tag{2.51}$$

where in the last inequaltiy we use the strong convexity of $\Phi(\cdot)$ and strong concavity of $\Psi(\cdot)$. It remains to bound $\|y_0-y_1^*\|$. Since $\Psi(y)-\frac{\tau}{2}\|y-y_0\|^2$ is $(\mu_y+\tau)$ strongly-concave about y, we have

$$\begin{aligned}\left(\Psi(y_1^*)-\frac{\tau}{2}\|y_1^*-y_0\|^2\right)-\frac{\tau+\mu_y}{2}\|y_1^*-y_0\|^2 &\ge \Psi(y_0)=\Psi^*-[\Psi^*-\Psi(y_0)]\\ &\ge \Psi(y_1^*)-[\Psi^*-\Psi(y_0)], \end{aligned}$$

and it further implies

$$\left(\tau + \frac{\mu_y}{2}\right) \|y_1^* - y_0\|^2 \leq \Psi^* - \Psi(y_0) \leq \text{gap}_f(x_0, y_0). \tag{2.52}$$

Plugging back into (2.51), we have

$$\begin{aligned}\|x_0 - x_1^*\|^2 + \|y_0 - y_1^*\|^2 &\leq \left[\frac{8\ell^2}{\mu_x^2 \min\{\mu_x, \mu_y\}} + \frac{8\ell^2 + 2\mu_x^2}{(2\tau + \mu_y)\mu_x^2}\right] \text{gap}_f(x_0, y_0) \\ &\leq \left[\frac{16\sqrt{2}\ell^2}{\mu_x^2 \min\{\mu_x, \mu_y\}} + \frac{16\sqrt{2}\ell^2 + 4\sqrt{2}\mu_x^2}{(2\tau + \mu_y)\mu_x^2}\right] \frac{1}{1-\rho}\epsilon^1.\end{aligned}$$

Case $t > 1$: (2.50) in the proof of Lemma 2.2.8 still holds. Now we want to bound $\|y_{t-1}^* - y_t^*\|$. By optimality condition, we have for $\forall y \in \mathcal{Y}$,

$$(y - y_t^*)^\top \nabla \Psi_t(y_t^*) \leq 0, \quad (y - y_{t-1}^*)^\top \nabla \Psi_{t-1}(y_{t-1}^*) \leq 0. \tag{2.53}$$

Choose y in the first inequality to be y_{t-1}^*, y in the second inequality to be y_t^*, and sum them together, we have

$$(y_t^* - y_{t-1}^*)^\top (\nabla \Psi_{t-1}(y_{t-1}^*) - \nabla \Psi_t(y_t^*)) \leq 0. \tag{2.54}$$

Using $\nabla \Psi_t(y) = \nabla_y f(x^*(y), y) - \tau(y - z_t)$, we have

$$(y_t^* - y_{t-1}^*)^\top (\nabla_y f(x^*(y_{t-1}^*), y_{t-1}^*) - \tau(y_{t-1}^* - z_{t-1}) - \nabla_y f(x^*(y_t^*), y_t^*) + \tau(y_t^* - z_t)) \leq 0. \tag{2.55}$$

By strong concavity of $\Psi(y) = \max_{x \in \mathcal{X}} f(x, y)$, we have

$$(y_t^* - y_{t-1}^*)^\top (\nabla \Psi(y_t^*) - \nabla \Psi(y_{t-1}^*)) \leq -\mu_y \|y_t^* - y_{t-1}^*\|^2. \tag{2.56}$$

Adding to (2.55), we have

$$(y_t^* - y_{t-1}^*)^\top [\tau(y_t^* - z_t) - \tau(y_{t-1}^* - z_{t-1})] \leq -\mu_y \|y_t^* - y_{t-1}^*\|^2 \tag{2.57}$$

Rearranging,

$$\frac{\tau}{\mu_y + \tau}(y_t^* - y_{t-1}^*)^\top (z_{t-1} - z_t) \geq \|y_t^* - y_{t-1}^*\|^2. \tag{2.58}$$

Further with $(y_t^* - y_{t-1}^*)^\top (z_{t-1} - z_t) \leq \|y_t^* - y_{t-1}^*\| \|z_{t-1} - z_t\|$, we have

$$\|y_t^* - y_{t-1}^*\| \leq \frac{\tau}{\mu_y + \tau} \|z_{t-1} - z_t\|. \tag{2.59}$$

From (2.38) and (2.39), we have for $t > 2$

$$\begin{aligned}\|z_t - z_{t-1}\| &= \|y_{t-1} + \beta_{t-1}(y_{t-1} - y_{t-2}) - y_{t-2} - \beta_{t-2}(y_{t-2} - y_{t-3})\| \\ &\leq (1 + \beta_{t-1})\|y_{t-1} - y_{t-2}\| + \beta_{t-2}\|y_{t-2} - y_{t-3}\| \\ &\leq 2\|y_{t-1} - y_{t-2}\| + \|y_{t-2} - y_{t-3}\| \\ &\leq 6\max\{\|y_{t-1} - y^*\|, \|y_{t-2} - y^*\|, \|y_{t-3} - y^*\|\}\end{aligned}$$

where in the second inequality we use $\beta_t \leq 1, \forall t$ (shown in the proof of Proposition 12 in [Lin et al., 2017]). Note that

$$\begin{aligned}\|z_t - z_{t-1}\|^2 &\leq 36\max\{\|y_{t-1} - y^*\|^2, \|y_{t-2} - y^*\|^2, \|y_{t-3} - y^*\|^2\} \\ &\leq \frac{72}{\mu_y}\max\{\Psi(y_{t-1}) - \Psi^*, \Psi(y_{t-2}) - \Psi^*, \Psi(y_{t-3}) - \Psi^*\} \\ &\leq \frac{72}{\mu_y}\max\{S_{t-1}, S_{t-2}, S_{t-3}\},\end{aligned}$$

where in the second inequality we use strongly concavity of h and in the last we use $\Psi(y_t) - \Psi^* \leq S_t$. Combining with (2.59) and (2.50), we have

$$\begin{aligned}&\|\tilde{x}_0 - x_t^*\|^2 + \|\tilde{y}_0 - y_t^*\|^2 \\ &\leq \left(\frac{4}{\mu_x} + \frac{4}{\mu_y + \tau}\right)\epsilon^{t-1} + \frac{144\tau^2}{(\mu_y + \tau)^2\mu_y}\left[\left(\frac{\ell}{\mu_x}\right)^2 + 1\right]\max\{S_{t-1}, S_{t-2}, S_{t-3}\}.\end{aligned}$$

Plugging in $S_t \leq \frac{8}{(\sqrt{q}-\rho)^2}(1-\rho)^{t+1}\,\mathrm{gap}_f(x_0, y_0)$ and definition of ϵ^t, we have

$$\|\tilde{x}_0 - x_t^*\|^2 + \|\tilde{y}_0 - y_t^*\|^2 \leq \left\{\left(\frac{4}{\mu_x} + \frac{4}{\mu_y + \tau}\right)\frac{1}{1-\rho} + \frac{2304\sqrt{2}\tau^2\left(\frac{\ell^2}{\mu_x^2} + 1\right)}{\mu_y(\mu_y + \tau)^2(\sqrt{q} - p)}\frac{1}{(1-\rho)^2}\right\}\epsilon^t. \tag{2.60}$$

It left to discuss the case $t = 2$. Similarly, we have

$$\|z_2 - z_1\| = \|y_1 + \beta_1(y_1 - y_0) - y_0\| = (1 + \beta_1)\|y_1 - y_0\| \leq 4\max\{\|y_1 - y^*\|, \|y_0 - y^*\|\}$$

Then

$$\begin{aligned}\|z_2 - z_1\|^2 \leq 16\max\{\|y_1 - y^*\|^2, \|y_0 - y^*\|^2\} &\leq \frac{32}{\mu_y}\max\{\Psi(y_1) - \Psi^*, \Psi(y_0) - \Psi^*\} \\ &\leq \frac{32}{\mu_y}\max\{S_1, \mathrm{gap}_f(x_0, y_0)\},\end{aligned}$$

Combining with (2.59) and (2.50), we have

$$\begin{aligned}&\|\tilde{x}_0 - x_2^*\|^2 + \|\tilde{y}_0 - y_2^*\|^2 \\ &\leq \left(\frac{4}{\mu_x} + \frac{4}{\mu_y + \tau}\right)\epsilon^1 + \frac{64\tau^2}{(\mu_y+\tau)^2\mu_y}\left[\left(\frac{\ell}{\mu_x}\right)^2 + 1\right]\max\{S_1, \mathrm{gap}_f(x_0, y_0)\}.\end{aligned}$$

Plugging in $S_1 \leq \frac{8}{(\sqrt{q}-\rho)^2}(1-\rho)^2\,\mathrm{gap}_f(x_0, y_0)$ and definition of ϵ^2, we have

$$\|\tilde{x}_0 - x_2^*\|^2 + \|\tilde{y}_0 - y_2^*\|^2 \leq \left\{\left(\frac{4}{\mu_x} + \frac{4}{\mu_y+\tau}\right)\frac{1}{1-\rho} + \frac{1024\sqrt{2}\tau^2\left(\frac{\ell^2}{\mu_x^2}+1\right)}{\mu_y(\mu_y+\tau)^2(\sqrt{q}-p)}\right\}\epsilon^2. \quad (2.61)$$

□

Proof of Corollary 2.2.10

Proof. We seperate into deterministic and stochastic settings.

DETERMINISTIC SETTING. We apply a deterministic algorithm $\mathcal{M}$ to solve the auxiliary problem and $\mathcal{M}$ has a linear rate described by (2.4). Denote $\tilde{\ell} = \ell + \tau$ as gradient lipschtiz constant of the auxiliary problem, and $\tilde{\mu} = \max\{\mu_x, \tau\}$. By Lemma 2.2.2, after K iterations of algorithm $\mathcal{M}$,

$$\|x_K - [x_K]_\beta\|^2 + \|y_K - [y_K]_\beta\|^2 \leq \frac{2}{(1-\tilde{\ell}/\beta)^3}[\|x_K - x^*\|^2 + \|y_K - y^*\|^2] \quad (2.62)$$

$$\leq \frac{2}{(1-\tilde{\ell}/\beta)^3}\left(1 - \frac{1}{\Lambda_{\mathcal{M},\tau}}\right)^K [\|\tilde{x}_0 - x^*\|^2 + \|\tilde{y}_0 - y^*\|^2]. \quad (2.63)$$

Let $\tilde{\epsilon}^{(t)} = \min\left\{\frac{\tilde{\mu}^2\epsilon^t}{2A(\beta+\tilde{\ell})^2}, \left(\frac{\tilde{\mu}\epsilon^t}{4\beta\mathcal{D}_\mathcal{Y}(\beta+\tilde{\ell})}\right)^2\right\}$. Choosing

$$K = \Lambda_{\mathcal{M},\tau}\log\frac{(1-\tilde{\ell}/\beta)^3(\|\tilde{x}_0 - x^*\|^2 + \|\tilde{y}_0 - y^*\|^2)}{2\tilde{\epsilon}^{(t)}} \leq \Lambda_{\mathcal{M},\tau}\log\frac{(1-\tilde{\ell}/\beta)^3 D}{2\tilde{\epsilon}^{(t)}}$$

where $D = D_1$ if $t = 1$ and $D = D_2$ if $t > 1$ as specified in Lemma 2.2.8, then we have $\|x_K - [x_K]_\beta\|^2 + \|y_K - [y_K]_\beta\|^2 \leq \tilde{\epsilon}^{(t)}$.

STOCHASTIC SETTING. We apply a stochastic algorithm $\mathcal{M}$ to solve the auxiliary problem and $\mathcal{M}$ has a linear rate described by (2.4). With the same reasoning as in deterministic setting and applying Appendix B.4 of [Lin et al., 2017], we have

$$K(\epsilon) \leq \Lambda_{\mathcal{M},\tau} \log \frac{(1-\tilde{\ell}/\beta)^3(\|\tilde{x}_0 - x^*\|^2 + \|\tilde{y}_0 - y^*\|^2)}{2\Lambda_{\mathcal{M},\tau}\tilde{\epsilon}^{(t)}} + 1,$$

and the conclusion follows. □

Proof of Corollary 2.2.9

Proof. We discuss the deterministic setting, since the stochastic setting follows in the same way as in the proof of Corollary 2.2.10.

DETERMINISTIC SETTING. Denote $\tilde{\ell} = \ell + \tau$ as gradient lipschtiz constant of the auxiliary problem, and $\tilde{\mu} = \max\{\mu_x, \mu_y + \tau\}$. Then (2.63) still holds. Let $\tilde{\epsilon}^{(t)} = \frac{\tilde{\mu}^2\epsilon^t}{A(\beta+\tilde{\ell})^2}$. Choosing

$$\begin{aligned} K &= \Lambda_{\mathcal{M},\tau} \log \frac{(1-\tilde{\ell}/\beta)^3(\|\tilde{x}_0 - x^*\|^2 + \|\tilde{y}_0 - y^*\|^2)}{2\tilde{\epsilon}^{(t)}} \\ &\leq \Lambda_{\mathcal{M},\tau} \log \frac{C(1-\tilde{\ell}/\beta)^3\epsilon^t}{2\tilde{\epsilon}^{(t)}} = \Lambda_{\mathcal{M},\tau} \log \frac{C(1-\tilde{\ell}/\beta)^3(\beta+\tilde{\ell})^2 A}{2\tilde{\mu}}, \end{aligned}$$

where $C = C_1$ if $t = 1$ and $C = C_2$ if $t > 1$ as specified in Lemma 2.2.7, then we have $\|x_K - [x_K]_\beta\|^2 + \|y_K - [y_K]_\beta\|^2 \leq \tilde{\epsilon}^{(t)}$.

□

D. Total Complexity

Proof of Corollary 2.2.11

Proof. By Theorem 2.2.3 and Lemma 2.2.2, Algorithm 1 finds a point (x, y) such that $\|x - x^*\|^2 + \|y - y^*\|^2 \leq \epsilon$ after $T = \mathcal{O}\left(\sqrt{\frac{\mu_y+\tau}{\mu_y}} \log\left(\frac{\max\{1,\ell\}\,\mathrm{gap}_f(x_0,y_0)}{\min\{1,\mu_x,\mu_y\}} \cdot \frac{1}{\epsilon}\right)\right)$ outer-loop iterations. By Corollary 2.2.9, it takes at most

$$K = \mathcal{O}\left(\Lambda_{\mathcal{M},\tau} \log\left(\frac{\max\{1,\ell,\tau\}}{\min\{1,\mu_x,\mu_y\}}\right)\right)$$

gradient oracle calls for $\mathcal{M}$ to solve the auxiliary problem. The total complexity is $K \cdot T$.

□

Proof of Corollary 2.2.12

Proof. Because $2/(t+2)^2 \le \alpha_t^2 \le 4/(t+1)^2$, by Theorem 2.2.5, Algorithm 1 finds ϵ-saddle point after $T = \mathcal{O}\left(\sqrt{\tau/\epsilon}\cdot\mathcal{D}_{\mathcal{Y}}+1\right)$ outer-loop iterations. Note that the accuracy we want for solving auxiliary problem $(\star)$ is, $\forall t \in [T]$,

$$\begin{aligned}\epsilon^t &= \frac{\tau(\rho-1)\mathcal{D}_{\mathcal{Y}}^2\alpha_t^2}{4(t+1)^\rho} \ge \frac{\tau(\rho-1)\mathcal{D}_{\mathcal{Y}}^2}{(t+1)^\rho(t+2)^2} \ge \frac{\tau(\rho-1)\mathcal{D}_{\mathcal{Y}}^2}{(T+1)^\rho(T+2)^2}\\ &= \Omega(\tau\rho\mu^{-1-\rho/2}\mathcal{D}_{\mathcal{Y}}^{-\rho}\epsilon^{1+\rho/2}),\end{aligned}$$

By Corollary 2.2.10, it takes at most

$$K = \mathcal{O}\left(\Lambda_{\mathcal{M},\tau}\log\left(\frac{\max\{1,\ell,\tau\}\mathcal{D}_{\mathcal{Y}}\|x_0-x^*\|}{\min\{1,\mu_x,\tau\}}\cdot\frac{1}{\epsilon}\right)\right)$$

gradient oracle calls for $\mathcal{M}$ to solve the auxiliary problem. The total complexity is $K\cdot T$. □

2.5.3 *Proofs for Chapter 2.3*

A. Outer-loop Complexity

Proof of Theorem 2.3.2

Proof. First we define ϕ as the extended-value function of Φ: $\phi(x) = \Phi(x)$ if $x \in \mathcal{X}$ and $\phi(x) = \infty$ if $x \notin \mathcal{X}$. Note that $\Phi(x) = \max_{y\in\mathcal{Y}} f(x,y)$ is ℓ-weakly convex [Lemma 3, [Thekumparampil et al., 2019]]. It directly follows from the definition of ϕ that ϕ is also ℓ-weakly convex. We define $\Phi_\tau(z;x) = \Phi(z) + \frac{1}{2\tau}\|z-x\|^2$. Define the proximal point of x by

$$\operatorname{prox}_{\tau\phi}(x) = \operatorname*{argmin}_z\left\{\phi(z)+\frac{1}{2\tau}\|z-x\|^2\right\} = \operatorname*{argmin}_{z\in\mathcal{X}}\Phi_\tau(z;x).$$

With $\tau = 2\ell$, By Lemma 4.3 in [Drusvyatskiy and Paquette, 2019],

$$\begin{aligned}\left\|\nabla\phi_{\frac{1}{2\ell}}(x_t)\right\|^2 &= 4\ell^2\|x_t - \operatorname{prox}_{\phi/2\ell}(x_t)\|^2\\ &\le 8\ell[\Phi_{1/2\ell}(x_t;x_t) - \Phi_{1/2\ell}(\operatorname{prox}_{\phi/1/2\ell}(x_t);x_t)]\\ &\le 8\ell[\Phi_{1/2\ell}(x_t;x_t) - \Phi_{1/2\ell}(x_{t+1};x_t) + \hat{\epsilon}]\\ &= 8\ell\left\{\Phi(x_t) - \left[\Phi(x_{t+1}) + \ell\|x_{t+1}-x_t\|^2\right] + \hat{\epsilon}\right\}\\ &\le 8\ell[\Phi(x_t) - \Phi(x_{t+1}) + \hat{\epsilon}],\end{aligned} \tag{2.64}$$

where in the first inequality we use ℓ-strong convexity of $\Phi_{1/2\ell}(\cdot; x_t)$, and the second inequality follows from $\Phi_{1/2\ell}(x_{t+1}; x_t) \leq \min_{x\in\mathcal{X}} \Phi_{1/2\ell}(x; x_t) + \hat{\epsilon}$. Summing from 0 to $T-1$, we get

$$\frac{1}{T}\sum_{t=0}^{T-1} \left\|\nabla \phi_{\frac{1}{2\ell}}(x_t)\right\|^2 \leq 8\ell \left[\frac{\Phi(x_0) - \Phi(x_T)}{T} + \bar{\epsilon}\right] \leq \frac{2\tau_x^2}{\tau_x - \ell}\left[\frac{\Delta}{T} + \hat{\epsilon}\right].$$

We finish the proof by noting $\nabla \phi_{\frac{1}{2\ell}}(x) = \nabla \Phi_{\frac{1}{2\ell}}(x)$ for $x \in \mathcal{X}$.

□

Now we will provide the outer-loop complexity for the NC-SC setting. We denote $(\hat{x}_t, \hat{y}_t)$ as the optimal solution to the auxiliary problem $(\star\star)$ at t-th iteration. It is easy to observe that $\hat{x}_t = \text{prox}_{\Phi/2\ell}(x_t)$. Define $\hat{\Phi}_t(x) = \max_y f(x, y) + \ell\|x - x_t\|^2$.

In the following theorem, we first show the convergence of the Moreau envelope $\|\nabla \Phi_{1/2\ell}(x)\|^2$.

Theorem 2.5.3. *Under the same assumption as Theorem 2.3.1, if we pick $\alpha_t = \frac{\mu^4}{28\ell^3}$ for $t > 0$ and $\alpha_0 = \frac{\mu^4}{32\ell^4 \max\{1,\ell\}}$, then iterates from Algorithm 2 satisfy*

$$\sum_{t=0}^{T-1} \|\nabla \Phi_{1/2\ell}(x_t)\|^2 \leq \frac{87L}{5}\Delta_0 + \frac{7\ell}{5} D_y^0, \tag{2.65}$$

where $D_y^0 = \|y_0 - y_*(x_0)\|^2$ and $\Delta_0 = \Phi(x^0) - \min_x \Phi(x)$.

Proof. In this proof, we denote $b_{t+1} = \text{gap}_{\hat{f}_t}(x_{t+1}, y_{t+1})$. According to the last proof and inequality (4.40),

$$\|\nabla \Phi_{1/2\ell}(x_t)\|^2 \leq 8\ell[\Phi(x_t) - \Phi(x_{t+1}) + b_{t+1}]. \tag{2.66}$$

Then, for $t \geq 1$

$$\begin{aligned}
\|y_t - \hat{y}_t\|^2 &\leq 2\|y_t - \hat{y}_{t-1}\|^2 + 2\|y^*(\hat{x}_{t-1}) - y^*(\hat{x}_t)\|^2 \\
&\leq 2\|y_t - \hat{y}_{t-1}\|^2 + 2\left(\frac{\ell}{\mu}\right)^2 \|\hat{x}_t - \hat{x}_{t-1}\|^2 \\
&\leq 2\|y_t - \hat{y}_{t-1}\|^2 + 4\left(\frac{\ell}{\mu}\right)^2 \|\hat{x}_t - x_t\|^2 + 4\left(\frac{\ell}{\mu}\right)^2 \|x_t - \hat{x}_{t-1}\|^2 \\
&\leq \frac{8\ell^2}{\mu^3} b_t + 4\left(\frac{\ell}{\mu}\right)^2 \|\hat{x}_t - x_t\|^2,
\end{aligned}$$

where we use Lemma 2.5.1 in the second inequality, and (ℓ, μ)-SC-SC of $\tilde{f}_{t-1}(x, y)$ and Lemma 2.2.2 in the last inequality. Therefore,

$$\|x_t - \hat{x}_t\|^2 + \|y_t - \hat{y}_t\|^2 \leq \frac{8\ell^2}{\mu^3} b_t + \left(\frac{4\ell^2}{\mu^2} + 1\right) \|\hat{x}_t - x_t\|^2. \tag{2.67}$$

By our inner-loop stopping criterion and $\|\nabla \Phi_{1/2\ell}(x_t)\|^2 = 4\ell^2 \|x_t - \hat{x}_t\|^2$, for $t \geq 1$

$$b_{t+1} \leq \alpha_t \left[\|x_t - \hat{x}_t\|^2 + \|y_t - \hat{y}_t\|^2\right] \leq \frac{8\ell^2 \alpha_t}{\mu^3} b_t + \alpha_t \left(\frac{1}{\mu^2} + \frac{1}{4\ell^2}\right) \|\nabla \Phi_{1/2\ell}(x_t)\|^2.$$

Define $\theta = \frac{2}{7}$ and $w = \frac{5\mu^2}{112\ell^3}$. It is easy to verify that as $\alpha_t = \frac{\mu^4}{28\ell^3}$, then $\frac{8\ell\alpha_t}{\mu^2} \leq \theta$ and $\alpha_t \left(\frac{1}{\mu^2} + \frac{1}{4\ell^2}\right) \leq w$. We conclude the following recursive bound

$$b_{t+1} \leq \theta b_t + w \|\nabla \Phi_{1/2\ell}(x_t)\|^2. \tag{2.68}$$

For $t = 0$,

$$\|y_0 - \hat{y}_0\|^2 \leq 2\|y_0 - y^*(x_0)\|^2 + 2\|\hat{y}_0 - y^*(x_0)\|^2 \leq 2\|y_0 - y^*(x_0)\|^2 + 2\left(\frac{\ell}{\mu}\right)^2 \|x_0 - \hat{x}_0\|^2. \tag{2.69}$$

Because $\Phi(x) + \ell\|x - x_0\|^2$ is ℓ-strongly convex, we have

$$\left(\Phi(\hat{x}_0) + \ell\|\hat{x}_0 - x_0\|^2\right) + \frac{\ell}{2}\|\hat{x}_0 - x_0\|^2 \leq \Phi(x_0) = \Phi^* + (\Phi(x_0) - \Phi^*) \leq \Phi(\hat{x}_0) + (\Phi(x_0) - \Phi^*).$$

This implies $\|\hat{x}_0 - x_0\|^2 \leq \frac{\ell}{2}(\Phi(x_0) - \Phi^*)$. Combining with (2.69)

$$\|y_0 - \hat{y}_0\|^2 + \|x_0 - \hat{x}_0\|^2 \leq \left(\frac{\ell^3}{\mu^2} + \frac{\ell}{2}\right)(\Phi(x_0) - \Phi^*) + 2\|y_0 - y^*(x_0)\|^2.$$

Hence, by the stopping criterion,

$$b_1 \leq \alpha_0 \left(\frac{\ell^3}{\mu^2} + \frac{\ell}{2}\right)(\Phi(x_0) - \Phi^*) + 2\alpha_0 \|y_0 - y^*(x_0)\|^2.$$

Define $\theta_0 = \frac{\mu^2}{16\ell^2}$. With $\alpha_0 = \frac{\mu^4}{32\ell^4 \max\{1,\ell\}}$, $\alpha_0 \left(\frac{\ell^3}{\mu^2} + \frac{\ell}{2}\right) \leq \theta_0$ and $2\alpha_0 \leq \theta_0$. Then we can write

$$b_1 \leq \theta_0(\Phi(x_0) - \Phi^*) + \theta_0 \|y_0 - y^*(x_0)\|^2.$$

Unravelling (2.68), we have for $t \geq 1$

$$\begin{aligned} b_{t+1} &\leq \theta^t b_1 + w \sum_{k=1}^{t} \theta^{t-k} \|\nabla \Phi_{1/2\ell}(x_k)\|^2 \\ &\leq \theta^t \theta_0 (\Phi(x_0) - \Phi^*) + \theta^t \theta_0 \|y_0 - y^*(x_0)\|^2 + w \sum_{k=1}^{t} \theta^{t-k} \|\nabla \Phi_{1/2\ell}(x_k)\|^2. \end{aligned}$$

Summing from $t = 0$ to $T - 1$,

$$\begin{aligned} \sum_{t=0}^{T-1} b_{t+1} &= \sum_{t=1}^{T-1} b_t + b_1 \\ &\leq \theta_0 \sum_{t=0}^{T-1} \theta^t [\Phi(x_0) - \Phi^*] + \theta_0 \sum_{t=0}^{T-1} \theta^t \|y_0 - y^*(x_0)\|^2 + w \sum_{t=1}^{T-1} \sum_{k=1}^{t} \theta^{t-k} \|\nabla \Phi_{1/2\ell}(x_k)\|^2 \\ &\leq \theta_0 \sum_{t=0}^{T-1} \theta^t [\Phi(x_0) - \Phi^*] + \theta_0 \sum_{t=0}^{T-1} \theta^t \|y_0 - y^*(x_0)\|^2 + w \sum_{t=1}^{T-1} \frac{1}{1-\theta} \|\nabla \Phi_{1/2\ell}(x_t)\|^2, \end{aligned} \tag{2.70}$$

where we use $\sum_{t=1}^{T-1} \sum_{k=1}^{t} \theta^{t-k} \|\nabla \Phi_{1/2\ell}(x_k)\|^2 = \sum_{k=1}^{T-1} \sum_{t=k}^{T} \theta^{t-k} \|\nabla \Phi_{1/2\ell}(x_k)\|^2 \leq \sum_{k=1}^{T-1} \frac{1}{1-\theta} \|\nabla \Phi_{1/2\ell}(x_k)\|^2$. Now, by telescoping (2.66),

$$\frac{1}{8\ell} \sum_{t=0}^{T-1} \|\nabla \Phi_{1/2\ell}(x_t)\|^2 \leq \Phi(x_0) - \Phi^* + \sum_{t=0}^{T-1} b_{t+1}.$$

Plugging (2.70) in,

$$\begin{aligned} &\frac{1}{8\ell} \sum_{t=0}^{T-1} \|\nabla \Phi_{1/2\ell}(x_t)\|^2 - w \sum_{t=1}^{T-1} \frac{1}{1-\theta} \|\nabla \Phi_{1/2\ell}(x_t)\|^2 \\ &\leq \left(1 + \frac{\theta_0}{1-\theta}\right) [\Phi(x_0) - \Phi^*] + \frac{\theta_0}{1-\theta} \|y_0 - y^*(x_0)\|^2. \end{aligned}$$

Plugging in $w \leq \frac{5}{112\ell}$, $\frac{1}{1-\theta} = \frac{7}{5}$ and $\theta_0 \leq \frac{1}{16}$

$$\frac{1}{16\ell} \sum_{t=0}^{T-1} \|\nabla \Phi_{1/2\ell}(x_t)\|^2 \leq \frac{87}{80} [\Phi(x_0) - \Phi^*] + \frac{7}{80} \|y_0 - y^*(x_0)\|^2.$$

□

Proof of Theorem 2.3.1

Proof. We still use $b_{t+1} = \text{gap}_{\hat{f}_t}(x_{t+1}, y_{t+1})$ as in the proof of Theorem 2.5.3. Since $\hat{x}_t$ is the optimal solution to $\min_{x\in\mathcal{X}} \hat{\Phi}_t(x)$, $\hat{\Phi}_t$ is differentiable and $\mathcal{X}$ is convex, we have

$$\left\| \hat{x}_t - \mathcal{P}_{\mathcal{X}} \left(\hat{x}_t - \frac{\nabla\Phi(\hat{x}_t) + 2\ell(\hat{x}_t - x_t)}{\ell} \right) \right\| = 0. \tag{2.71}$$

Therefore,

$$\begin{aligned}
&\left\| x_{t+1} - \mathcal{P}_{\mathcal{X}} \left(x_{t+1} - \frac{1}{\ell}\nabla\Phi(x_{t+1}) \right) \right\| \\
&= \left\| x_{t+1} - \mathcal{P}_{\mathcal{X}} \left(x_{t+1} - \frac{1}{\ell}\nabla\Phi(x_{t+1}) \right) \right\| - \left\| \hat{x}_t - \mathcal{P}_{\mathcal{X}} \left(\hat{x}_t - \frac{\nabla\Phi(\hat{x}_t) + 2\ell(\hat{x}_t - x_t)}{\ell} \right) \right\| \\
&\leq 2\|x_{t+1} - \hat{x}_t\| + 2\|x_t - \hat{x}_t\| + \frac{\|\nabla\Phi(\hat{x}_t) - \nabla\Phi(x_{t+1})\|}{\ell} \\
&\leq 2\left(1 + \frac{\ell}{\mu}\right)\|x_{t+1} - \hat{x}_t\| + 2\|x_t - \hat{x}_t\| \\
&\leq \frac{4\ell}{\mu}\|x_{t+1} - \hat{x}_t\| + 2\|x_t - \hat{x}_t\|,
\end{aligned} \tag{2.72}$$

where in the second inequality we use Lemma 2.5.1. Further with Lemma 2.2.2 and Lemma 4.3 in [Drusvyatskiy and Paquette, 2019],

$$\ell^2 \left\| x_{t+1} - \mathcal{P}_{\mathcal{X}} \left(x_{t+1} - \frac{1}{\ell}\nabla\Phi(x_{t+1}) \right) \right\|^2 \leq \frac{32\ell^3}{\mu^2} b_{t+1} + 2\|\nabla\Phi_{1/2\ell}(x_t)\|^2.$$

Summing from $t = 0$ to $T - 1$, we have

$$\ell^2 \sum_{t=0}^{T-1} \left\| x_{t+1} - \mathcal{P}_{\mathcal{X}} \left(x_{t+1} - \frac{1}{\ell}\nabla\Phi(x_{t+1}) \right) \right\|^2 \leq \frac{32\ell^3}{\mu^2} \sum_{t=0}^{T-1} b_{t+1} + 2\sum_{t=0}^{T-1} \|\nabla\Phi_{1/2\ell}(x_t)\|^2. \tag{2.73}$$

Applying (2.70), we have

$$\begin{aligned}
\frac{32\ell^3}{\mu^2} \sum_{t=0}^{T-1} b_{t+1} \leq &\frac{32\ell^3\theta_0}{\mu^2} \sum_{t=0}^{T-1} \theta^t[\Phi(x_0) - \Phi^*] + \frac{32\ell^3\theta_0}{\mu^2} \sum_{t=0}^{T-1} \theta^t\|y^0 - y^*(x_0)\|^2 + \\
&\frac{32\ell^3 w}{\mu^2} \sum_{t=1}^{T-1} \frac{1}{1-\theta}\|\nabla\Phi_{1/2\ell}(x_t)\|^2.
\end{aligned}$$

Plugging in $\theta_0 = \frac{\mu^2}{16\ell^2}$, $\theta = \frac{2}{7}$ and $w = \frac{5\mu^2}{112\ell^3}$,

$$\frac{32\ell^3}{\mu^2} \sum_{t=0}^{T-1} b_{t+1} \leq \frac{14\ell}{5}[\Phi(x_0) - \Phi^*] + \frac{14\ell}{5}\|y_0 - y^*(x^0)\|^2 + 2\sum_{t=1}^{T-1} \|\nabla\Phi_{1/2\ell}(x_t)\|^2.$$

Plugging back into (2.73),

$$\ell^2 \sum_{t=0}^{T-1} \left\| x_{t+1} - \mathcal{P}_{\mathcal{X}} \left(x_{t+1} - \frac{1}{\ell} \nabla \Phi(x_{t+1}) \right) \right\|^2$$
$$\leq \frac{14\ell}{5} [\Phi(x_0) - \Phi^*] + \frac{14\ell}{5} \|y_0 - y^*(x_0)\|^2 + 3 \sum_{t=0}^{T-1} \|\nabla \Phi_{1/2\ell}(x_t)\|^2.$$

Applying Theorem 2.5.3,

$$\ell^2 \sum_{t=1}^{T} \left\| x_t - \mathcal{P}_{\mathcal{X}} \left(x_t - \frac{1}{\ell} \nabla \Phi(x_t) \right) \right\|^2 \leq \frac{275\ell}{5T} [\Phi(x_0) - \Phi^*] + \frac{35\ell}{5T} \|y_0 - y^*(x_0)\|^2.$$

□

B. Warm Start

Proof of Lemma 2.3.4

Proof. The subproblem is (ℓ, μ)-SC-SC and 3ℓ-smooth. By Lemma 2.2.2, with $\beta > 6\ell$,

$$\frac{\alpha_t}{A} \operatorname{gap}_{\hat{f}_t}([z_t]_\beta]) \leq \alpha_t \|z_t - \hat{z}_t\|^2.$$

□

Proof of Lemma 2.3.5

Proof. From equation (2.64) in the proof of Theorem 2.3.2, we have

$$\|x_t - x_t^*\|^2 \leq \frac{2}{\ell} [\Phi(x_t) - \Phi(x_{t+1}) + \hat{\epsilon}_t].$$

Summing from 0 to $T-1$, we get

$$\sum_{t=0}^{T-1} \|x_t - x_t^*\|^2 \leq \frac{2\Delta}{\ell} + \frac{2}{\ell} \sum_{t=0}^{T-1} \hat{\epsilon}_t \leq \frac{6\Delta}{\ell},$$

where in the second inequality, we use $T = \frac{16\ell\Delta}{\epsilon^2}$ and $\hat{\epsilon}_t = \frac{\epsilon^2}{8\ell}$. Therefore, for $\forall t \leq \frac{16\ell\Delta}{\epsilon^2} - 1$, we have $\|x_t - x_t^*\| \leq \sqrt{\frac{6\Delta}{\ell}}$. □

3

GLOBAL CONVERGENCE FOR PL-PL MINIMAX PROBLEMS

Simple algorithms such as the gradient descent ascent (GDA) are the common practice for solving these nonconvex games and receive lots of empirical success. Yet, it is known that these vanilla GDA algorithms with constant step size can potentially diverge even in the convex setting. In this chapter, we show that for a subclass of nonconvex-nonconcave objectives satisfying a so-called two-sided Polyak-Łojasiewicz inequality, the alternating gradient descent ascent (AGDA) algorithm converges globally at a linear rate and the stochastic AGDA achieves a sublinear rate. We further develop a variance-reduced algorithm that attains a provably faster rate than AGDA when the problem has a finite-sum structure.

3.1 OVERVIEW

We consider unconstrained minimax optimization problems of the forms

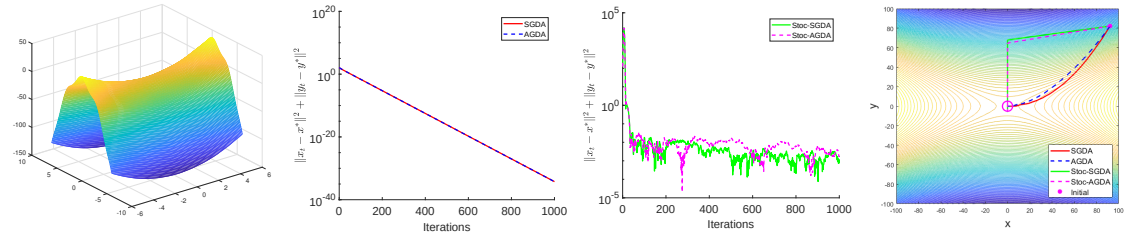
$$\min_{x\in\mathbb{R}^{d_1}} \max_{y\in\mathbb{R}^{d_2}} f(x,y) \triangleq \mathbb{E}_{\xi\sim P}[F(x,y;\xi)], \tag{3.1}$$

and

$$\min_{x\in\mathbb{R}^{d_1}} \max_{y\in\mathbb{R}^{d_2}} f(x,y) \triangleq \frac{1}{n}\sum_{i=1}^{n} f_i(x,y), \tag{3.2}$$

where ξ is a random vector, and $f(x,y)$ is a possibly nonconvex-nonconcave function.

The most frequently used methods for solving minimax problems are the gradient descent ascent (GDA) algorithms (or their stochastic variants), with either simultaneous or alternating updates of the primal-dual variables, referred to as SGDA and AGDA, respectively. While these algorithms have received much empirical success, especially in adversarial training, it is known that GDA algorithms with constant stepsizes could fail to converge even for the bilinear games [Gidel et al., 2019, Mescheder et al., 2018]; when they do converge, the stable limit point may not be a local Nash equilibrium [Daskalakis et al., 2018, Mazumdar and Ratliff, 2018]. On the other hand, GDA algorithms can converge linearly to the saddle point for strongly-convex-strongly-concave functions [Facchinei and Pang, 2007]. Moreover, for many simple nonconvex-nonconcave objective functions, such as, $f(x,y) = x^2 + 3\sin^2 x \sin^2 y - 4y^2 - 10\sin^2 y$, we observe that GDA algorithms with constant stepsizes converge to the global saddle point (see Figure 1). These facts naturally



(a) Surface plot of $f(x, y)$ (b) Deterministic GDA (c) Stochastic GDA (d) Trajectories

FIGURE 3.1: (a) Surface plot of the nonconvex-nonconcave function $f(x, y) = x^2 + 3\sin^2 x \sin^2 y - 4y^2 - 10\sin^2 y$; (b) Convergence of SGDA and AGDA; (c) Convergence of stochastic SGDA and stochastic AGDA; (d) Trajectories of four algorithms

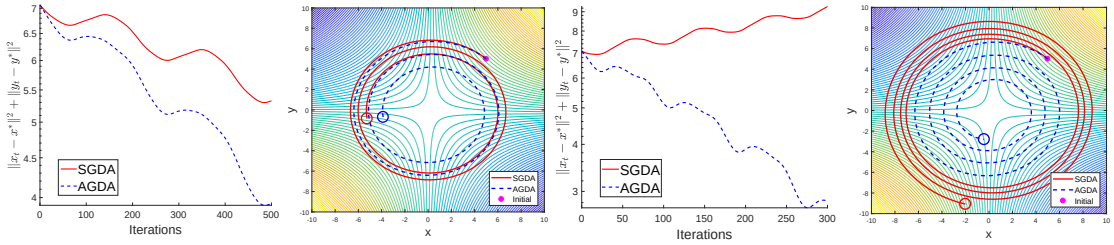
raise a question: *Is there a general condition under which GDA algorithms converge to the global optima?*

Furthermore, the use of variance reduction techniques has played a prominent role in improving the convergence over stochastic or batch algorithms for both convex and nonconvex minimization problems [Johnson and Zhang, 2013, Reddi et al., 2016a,b, Xiao and Zhang, 2014]. However, when it comes to the minimax problems, there are limited results, except under convex-concave setting [Palaniappan and Bach, 2016, Du and Hu, 2019]. This leads to another open question: *Can we improve GDA algorithms for nonconvex-nonconcave minimax problems?*

3.1.1 *Contributions*

In this chapter, we address these two questions and specifically focus on the alternating gradient descent ascent, namely AGDA. AGDA is widely used for training GANs and other minimax problems in practice; see e.g., [Liu and Tuzel, 2016, Metz et al., 2016]. Yet there is a lack of discussion on the convergence of AGDA for general minimax problems in the literature, even for the favorable strongly-convex-strongly-concave setting. Our main contributions are summarized as follows.

TWO-SIDED PL CONDITION. First, we identity a general condition that relaxes the convex-concavity requirement of the objective function while still guaranteeing global convergence of AGDA and stochastic AGDA (Stoc-AGDA). We call this the two-sided PL condition, which requires that both players' utility functions satisfy Polyak-Łojasiewicz (PL) inequality [Polyak, 1963]. The two-sided PL condition is very general and is satisfied by many important classes of functions: (a) all strongly-convex-strongly-concave functions; (b) all PL-strongly-concave function (discussed in [Guo et al., 2020]) and (c) many nonconvex-nonconcave objectives. Such conditions also hold true for various applications, including



(a) $\tau = 0.01$ (b) $\tau = 0.01$ (c) $\tau = 0.025$ (d) $\tau = 0.025$

FIGURE 3.2: Consider $f(x, y) = \log(1 + e^x) + 3xy - \log(1 + e^y)$: (a) Convergence of AGDA and SGDA with the stepsize $\tau = 0.01$; (b) Trajectories of two algorithms with $\tau = 0.01$; (c) Convergence of AGDA and SGDA with stepsize $\tau = 0.025$; (d) Trajectories of two algorithms with $\tau = 0.025$;

robust least square, generative adversarial imitation learning for linear quadratic regulator (LQR) dynamics [Cai et al., 2019], zero-sum linear quadratic game [Zhang et al., 2019c], and potentially many others in adversarial learning [Du et al., 2019], robust phase retrieval [Sun et al., 2018, Zhou et al., 2016], robust control [Fazel et al., 2018], and etc. We first investigate the landscape of objectives under the two-sided PL condition. In particular, we show that three notions of optimality: saddle point, minimax point, and stationary point are equivalent.

GLOBAL CONVERGENCE OF AGDA. We show that under the two-sided PL condition, AGDA with proper constant stepsizes converges globally to a saddle point at a linear rate of $\mathcal{O}(1 - \kappa^{-3})^t$, while Stoc-AGDA with proper diminishing stepsizes converges to a saddle point at a sublinear rate of $\mathcal{O}(\kappa^5/t)$, where κ is the underlying condition number. To the best of our knowledge, this is the first result on the global convergence of a class of nonconvex-nonconvex problems. In contrast, most previous work deals with nonconvex-concave problems and obtains convergence to stationary points. On the other hand, because all strongly-convex-strongly-concave and PL-strongly-concave functions naturally satisfy the two-sided PL condition, our analysis fills the theoretical gap with the first convergence results of AGDA under these settings.

VARIANCE REDUCED ALGORITHM. For minimax problems with the finite-sum structure, we introduce a variance-reduced AGDA algorithm (VR-AGDA) that leverages the idea of stochastic variance reduced gradient (SVRG) [Johnson and Zhang, 2013, Reddi et al., 2016a] with the alternating updates. We prove that VR-AGDA achieves the complexity of $\mathcal{O}\left((n + n^{2/3}\kappa^3)\log(1/\epsilon)\right)$, which improves over the $\mathcal{O}\left(n\kappa^3 \log\frac{1}{\epsilon}\right)$ complexity of AGDA and the $\mathcal{O}\left(\kappa^5/\epsilon\right)$ complexity of Stoc-AGDA when applied to finite-sum minimax problems. Our numerical experiments further demonstrate that VR-AGDA performs significantly

better than AGDA and Stoc-AGDA, especially for problems with large condition numbers. To our best knowledge, this is the first work to provide a variance-reduced algorithm and theoretical guarantees in the nonconvex-nonconcave regime of minimax optimization. In contrast, most previous variance-reduced algorithms require full or partial strong convexity and only apply to simultaneous updates.

3.1.2 *Related work*

NONCONVEX MINIMAX PROBLEMS. There has been a recent surge in research on solving minimax optimization beyond the convex-concave regime [Sinha et al., 2017, Chen et al., 2017, Qian et al., 2019, Thekumparampil et al., 2019, Lin et al., 2018, Nouiehed et al., 2019, Abernethy et al., 2019, Lin et al., 2020b, Barazandeh and Razaviyayn, 2020, Ostrovskii et al., 2020], but they differ from our work from various perspectives. Most of these work focus on the nonconvex-concave regime and aim for convergence to stationary points of minimax problems [Chen et al., 2017, Sinha et al., 2017, Lin et al., 2020a, Thekumparampil et al., 2019]. Algorithms in these work require solving the inner maximization or some subproblems with high accuracy, which are different from AGDA. Lin et al. [2018] proposed an inexact proximal point method to find an ϵ-stationary point for a class of weakly-convex-weakly-concave minimax problems. Their convergence result relies on assuming the existence of a solution to the corresponding Minty variational inequality, which is hard to verify. Abernethy et al. [2019] showed the linear convergence of a second-order iterative algorithm, called Hamiltonian gradient descent, for a subclass of "sufficiently bilinear" functions. Very recently, Xu et al. [2020c] and Boţ and Böhm [2020] anslyze AGDA in nonconvex-(strongly-)concave setting. There is also a line of work in understanding the dynamics in minimax games [Mazumdar et al., 2020, Fiez et al., 2019, Fiez and Ratliff, 2020, Fiez et al., 2020, Daskalakis and Panageas, 2018, Hsieh et al., 2021].

VARIANCE-REDUCED MINIMAX OPTIMIZATION. Palaniappan and Bach [2016], Luo et al. [2019], Chavdarova et al. [2019] provided linear-convergent algorithms for strongly-convex-strongly-concave objectives, based on simultaneous updates. Du and Hu [2019] extended the result to convex-strongly-concave objectives with full-rank coupling bilinear term. In contrast, we are dealing with a much broader class of objectives that are possibly nonconvex-nonconcave. We point out that Luo et al. [2020] and Xu et al. [2020a] recently introduced variance-reduced algorithms for finding the stationary point of nonconvex-strongly-concave problems, which is again different from our setting.

3.2 GLOBAL OPTIMA AND TWO-SIDED PL CONDITION

We assume that the function $f(x, y)$ in (4.1) is continuously differentiable and has Lipschitz gradient. Here $\|\cdot\|$ is used to denote the Euclidean norm.

Assumption 6 (Lipschitz gradient)**.** *There exists a positive constant $l > 0$ such that*

$$\max\{\left\|\nabla_y f(x_1, y_1) - \nabla_y f(x_2, y_2)\right\|, \|\nabla_x f(x_1, y_1) - \nabla_x f(x_2, y_2)\|\} \leq l[\|x_1 - x_2\| + \|y_1 - y_2\|],$$

holds for all $x_1, x_2 \in \mathbb{R}^{d_1}, y_1, y_2 \in \mathbb{R}^{d_2}$.

We now define three notions of optimality for minimax problems. The most direct notion of optimality is the global minimax point, at which x^* is an optimal solution to the function $g(x) := \max_y f(x, y)$ and y^* is an optimal solution to $\max_y f(x^*, y)$. In the two-player zero-sum game, the notion of saddle point is also widely used [Von Neumann et al., 2007, Nash, 1953]. For a saddle point (x^*, y^*), x^* is an optimal solution to $\min_x f(x, y^*)$ and y^* is an optimal solution to $\max_y f(x^*, y)$.

Definition 14 (Global optima)**.**

1. (x^*, y^*) *is a global minimax point, if for any* $(x, y) : f(x^*, y) \leq f(x^*, y^*) \leq \max_{y'} f(x, y')$.
2. (x^*, y^*) *is a saddle point, if for any* $(x, y) : f(x^*, y) \leq f(x^*, y^*) \leq f(x, y^*)$.
3. (x^*, y^*) *is a stationary point, if* : $\nabla_x f(x^*, y^*) = \nabla_y f(x^*, y^*) = 0$.

For general nonconvex-nonconcave minimax problems, these three notions of optimality are not necessarily equivalent. A stationary point may not be a saddle point or a global minimax point; a global minimax point may not be a saddle point or a stationary point. Note that for minimax problems, a saddle point or a global minimax point may not always exist. However, since our goal in this chapter is to find global optima, in the remainder of the chapter, we assume that a saddle point always exists.

Assumption 7 (Existence of saddle point)**.** *The objective function f has at least one saddle point. We also assume that for any fixed y, $\min_{x \in \mathbb{R}^{d_1}} f(x, y)$ has a nonempty solution set and a optimal value, and for any fixed x, $\max_{y \in \mathbb{R}^{d_2}} f(x, y)$ has a nonempty solution set and a finite optimal value.*

For unconstrained minimization problems: $\min_{x \in \mathbb{R}^n} f(x)$, Polyak [1963] proposed Polyak-Łojasiewicz (PL) condition, which is sufficient to show global linear convergence for gradient descent without assuming convexity. Specifically, a function $f(\cdot)$ satisfies PL condition if it has a nonempty solution set and a finite optimal value f^*, and there exists

some $\mu > 0$ such that $\|\nabla f(x)\|^2 \geq 2\mu(f(x) - f^*), \forall x$. As discussed in Karimi et al. [2016], PL condition is weaker, or not stronger, than other well-known conditions that guarantee linear convergence for gradient descent, such as error bounds (EB) [Luo and Tseng, 1993], weak strong convexity (WSC) [Necoara et al., 2018] and restricted secant inequality (RSI) [Zhang and Yin, 2013].

We introduce a straightforward generalization of the PL condition to the minimax problem: function $f(x, y)$ satisfies the PL condition with constant μ_1 with respect to x, and -f satisfies PL condition with constant μ_2 with respect to y. We formally state this in the following definition.

Definition 15 (Two-sided PL condition)**.** *A continuously differentiable function $f(x, y)$ satisfies the two-sided PL condition if there exist constants $\mu_1, \mu_2 > 0$ such that: $\forall x, y$,*

$$\|\nabla_x f(x, y)\|^2 \geq 2\mu_1[f(x, y) - \min_x f(x, y)], \quad \|\nabla_y f(x, y)\|^2 \geq 2\mu_2[\max_y f(x, y) - f(x, y)].$$

The two-sided PL condition does not imply convexity-concavity, and it is a much weaker condition than strong-convexity-strong-concavity. In Lemma 3.2.1, we show that three notions of optimality are equivalent under the two-sided PL condition. Note that they may not be unique.

Lemma 3.2.1. *If the objective function $f(x, y)$ satisfies the two-sided PL condition, then the following holds true:*

$$\textit{(saddle point)} \Leftrightarrow \textit{(global minimax)} \Leftrightarrow \textit{(stationary point)}.$$

Below we give some examples that satisfy this condition.

Example 1. *The nonconvex-nonconcave function in the introduction, $f(x, y) = x^2 + 3\sin^2 x \sin^2 y - 4y^2 - 10\sin^2 y$ satisfies the two-sided PL condition with $\mu_1 = 1/16, \mu_2 = 1/11$ (see Appendix 3.6.1).*

Example 2. *$f(x, y) = F(Ax, By)$, where $F(\cdot, \cdot)$ is strongly-convex-strongly-concave and A and B are arbitrary matrices, satisfies the two-sided PL condition.*

Example 3. *The generative adversarial imitation learning for LQR can be formulated as $\min_K \max_\theta m(K, \theta)$, where m is strongly-concave in terms of θ and satisfies PL condition in terms of K (see [Cai et al., 2019] for more details), thus satisfying the two-sided PL condition.*

Example 4. *In a zero-sum linear quadratic (LQ) game, the system dynamics are characterized by $x_{t+1} = Ax_t + Bu_t + Cv_t$, where x_t is the system state and u_t, v_t are the control inputs from two-players. After parameterizing the policies of two players by $u_t = -Kx_t$ and $v_t = -Lx_t$, the value*

function is $C(K,L) = \mathbb{E}_{x_0\sim\mathcal{D}}\left\{\sum_{t=0}^{\infty}\left[x_t^\top Q x_t + (Kx_t)^\top R^u (Kx_t) - (Lx_t)^\top R^v (Lx_t)\right]\right\}$*, where* $\mathcal{D}$ *is the distribution of the initial state* x_0 *(see [Zhang et al., 2019c] for more details). Player 1 (player 2) wants to minimize (maximize)* $C(K,L)$*, and the game is formulated as* $\min_K \max_L C(K,L)$*. Fixing* L *(or* K*),* $C(\cdot,L)$ *(or* $-C(K,\cdot)$*) becomes an objective of an LQR problem, and therefore satisfies the PL condition [Fazel et al., 2018] when* $\operatorname{argmin}_K C(K,L)$ *and* $\operatorname{argmax}_L C(K,L)$ *are well-defined.*

The two-sided PL condition includes rich classes of functions, including: (a) all strongly-convex-strongly-concave functions; (b) some convex-concave functions (e.g., Example 2); (c) some nonconvex-strongly-concave functions (e.g., Example 3); (d) some nonconvex-nonconcave functions (e.g., Example 1 and 4). Under the two-sided PL condition, the function $g(x) := \max_y f(x,y)$ satisfies PL condition with μ_1 (see Appendix 3.6.1). Moreover, it holds that g is also L-smooth with $L := l + l^2/\mu_2$ [Nouiehed et al., 2019]. Finally, we denote $\mu = \min(\mu_1,\mu_2)$ and $\kappa = \frac{l}{\mu}$, which represents the condition number of the problem.

3.3 GLOBAL CONVERGENCE OF AGDA AND STOC-AGDA

In this subchapter, we establish the convergence rate of the stochastic alternating gradient descent ascent (Stoc-AGDA) algorithm, which we present in Algorithm 4, under the two-sided PL condition. Stoc-AGDA updates variables x and y sequentially using stochastic gradient descent/ascent steps. Here we make standard assumptions about stochastic gradients $G_x(x,y,\xi)$ and $G_y(x,y,\xi)$.

Assumption 8 (Bounded variance)**.** *$G_x(x,y,\xi)$ and $G_y(x,y,\xi)$ are unbiased stochastic estimators of $\nabla_x f(x,y)$ and $\nabla_y f(x,y)$ and have variances bounded by $\sigma^2 > 0$.*

Algorithm 4 Stoc-AGDA

1: Input: (x_0,y_0), stepsizes $\{\tau_1^t\}_t > 0, \{\tau_2^t\}_t > 0$
2: **for all** $t = 0,1,2,\ldots$ **do**
3: Draw two i.i.d. samples $\xi_{t1},\xi_{t2} \sim P(\xi)$
4: $x_{t+1} \leftarrow x_t - \tau_1^t G_x(x_t,y_t,\xi_{t1})$
5: $y_{t+1} \leftarrow y_t + \tau_2^t G_y(x_{t+1},y_t,\xi_{t2})$
6: **end for**

Note that Stoc-AGDA with constant stepsizes (i.e., $\tau_1^t = \tau_1$ and $\tau_2^t = \tau_2$) and noiseless stochastic gradient (i.e., $\sigma^2 = 0$) reduces to AGDA:

$$\begin{aligned} x_{t+1} &= x_t - \tau_1 \nabla_x f(x_t,y_t), \\ y_{t+1} &= y_t + \tau_2 \nabla_y f(x_{t+1},y_t). \end{aligned} \tag{3.3}$$

We measure the inaccuracy of (x_t, y_t) through the potential function

$$P_t := a_t + \lambda \cdot b_t, \tag{3.4}$$

where $a_t = \mathbb{E}[g(x_t) - g^*], b_t = \mathbb{E}[g(x_t) - f(x_t, y_t)]$ and the balance parameter $\lambda > 0$ will be specified later in the theorems. Recall that $g(x) := \max_y f(x, y)$ and $g^* = \min_x g(x)$. This metric is driven by the definition of minimax point, because $g(x) - g^*$ and $g(x) - f(x, y)$ are non-negative for any (x, y), and both equal to 0 if and only if (x, y) is a minimax point.

STOC-AGDA WITH CONSTANT STEPSIZES We first consider Stoc-AGDA with constant stepsizes. We show that $\{(x_t, y_t)\}_t$ will converge linearly to a neighbourhood of the optimal set.

Theorem 3.3.1. *Suppose Assumptions 10, 7, 12 hold and $f(x, y)$ satisfies the two-sided PL condition with μ_1 and μ_2. Define $P_t := a_t + \frac{1}{10} b_t$. If we run Algorithm 4 with $\tau_2^t = \tau_2 \leq \frac{1}{l}$ and $\tau_1^t = \tau_1 \leq \frac{\mu_2^2 \tau_2}{18 l^2}$,*

$$P_t \leq (1 - \frac{1}{2}\mu_1 \tau_1)^t P_0 + \delta, \tag{3.5}$$

where $\delta = \frac{(1-\mu_2\tau_2)(L+l)\tau_1^2 + l\tau_2^2 + 10L\tau_1^2}{10\mu_1\tau_1}\sigma^2$.

Remark 3.3.2. *In the theorem above, we choose τ_1 smaller than τ_2, $\tau_1/\tau_2 \leq \mu_2^2/(18l^2)$, because our potential function is not symmetric about x and y. Another reason is because we want y_t to approach $y^*(x_t) \in \arg\max_y f(x_t, y)$ faster so that $\nabla_x f(x_t, y_t)$ is a better approximation for $\nabla g(x_t)$ ($\nabla g(x) = \nabla_x f(x, y^*(x))$, see Nouiehed et al. [2019]). Indeed, it is common to use different learning rates for x and y in GDA algorithms for nonconvex minimax problems; see e.g., Jin et al. [2020] and Lin et al. [2020a]. Note that the ratio between these two learning rates is quite crucial here. We also observe empirically when the same learning rate is used, even if small, the algorithm may not converge to saddle points.*

Remark 3.3.3. *When $t \to \infty$, $P_t \to \delta$. If $\tau_1 \to 0$ and $\tau_2^2/\tau_1 \to 0$, the error term δ will go to 0. When using smaller stepsizes, the algorithm reaches a smaller neighbour of the saddle point yet at the cost of a slower rate, as the contraction factor also deteriorates.*

LINEAR CONVERGENCE OF AGDA Setting $\sigma^2 = 0$, it follows immediately from the previous theorem that AGDA converges linearly under the two-sided PL condition. Moreover, we have the following:

Theorem 3.3.4. *Suppose Assumptions 10, 7 hold and $f(x,y)$ satisfies the two-sided PL condition with μ_1 and μ_2. Define $P_t := a_t + \frac{1}{10}b_t$. If we run AGDA with $\tau_1 = \frac{\mu_2^2}{18l^3}$ and $\tau_2 = \frac{1}{l}$, then*

$$P_t \leq \left(1 - \frac{\mu_1\mu_2^2}{36l^3}\right)^t P_0. \tag{3.6}$$

Furthermore, $\{(x_t, y_t)\}_t$ converges to some saddle point (x^*, y^*), and

$$\|x_t - x^*\|^2 + \|y_t - y^*\|^2 \leq \alpha \left(1 - \frac{\mu_1\mu_2^2}{36l^3}\right)^t P_0, \tag{3.7}$$

where α is a constant depending on μ_1, μ_2 and l.

The above theorem implies that the limit point of $\{(x_t, y_t)\}_t$ is a saddle point and the distance to the saddle point decreases in the order of $\mathcal{O}\left((1-\kappa^{-3})^t\right)$. Note that in the special case when the objective is strongly-convex-strongly-concave, it is known that SGDA (GDA with simultaneous updates) achieves an $\mathcal{O}(\kappa^2 \log(1/\epsilon))$ iteration complexity (see, e.g., Facchinei and Pang [2007]) and this can be further improved to match the lower complexity bound $\mathcal{O}(\kappa \log(1/\epsilon))$ [Zhang et al., 2019b] by extragradient methods [Korpelevich, 1976] or Nesterov's dual extrapolation [Nesterov and Scrimali, 2006]. However, these results heavily rely on the strong monotonicity of the corresponding variational inequality, which does not apply here. Since the general two-sided PL condition contains a much broader class of functions, we may not expect to achieve the same dependency on κ.

STOC-AGDA WITH DIMINISHING STEPSIZES While Stoc-AGDA with constant stepsizes only converges linearly to a neighbourhood of the saddle point, Stoc-AGDA with diminishing stepsizes converges to the saddle point but at a sublinear rate $\mathcal{O}(1/t)$.

Theorem 3.3.5. *Suppose Assumptions 10, 7, 12 hold and $f(x,y)$ satisfies the two-sided PL condition with μ_1 and μ_2. Define $P_t = a_t + \frac{1}{10}b_t$. If we run algorithm 4 with stepsizes $\tau_1^t = \frac{\beta}{\gamma+t}$ and $\tau_2^t = \frac{18l^2\beta}{\mu_2^2(\gamma+t)}$ for some $\beta > 2/\mu_1$ and $\gamma > 0$ such that $\tau_1^1 \leq \min\{1/L, \mu_2^2/18l^2\}$, then we have*

$$P_t \leq \frac{\nu}{\gamma+t}, \quad \text{where } \nu := \max\left\{\gamma P_0, \frac{\left[(L+l)\beta^2 + 18^2 l^5\beta^2/\mu_2^4 + 10L\beta^2\right]\sigma^2}{10\mu_1\beta - 20}\right\}. \tag{3.8}$$

Remark 3.3.6. *Note the rate is affected by ν, and the first term in the definition of ν is controlled by the initial point. In practice, we can find a good initial point by running Stoc-AGDA with constant stepsizes so that only the second term in the definition of ν matters. Then by choosing $\beta = 3/\mu_1$, we have $\nu = \mathcal{O}\left(\frac{l^5\sigma^2}{\mu_1^2\mu_2^4}\right)$. Thus, the convergence rate of Stoc-AGDA is $\mathcal{O}\left(\frac{\kappa^5\sigma^2}{\mu t}\right)$.*

Algorithm 5 VR-AGDA

input: $(\tilde{x}_0, \tilde{y}_0)$, stepsizes τ_1, τ_2, iteration numbers N, T
for all $k = 0, 1, 2, ...$ **do**
  for all $t = 0, 1, 2, ...T-1$ **do**
    $x_{t,0} = \tilde{x}_t, \quad y_{t,0} = \tilde{y}_t,$
    compute $\nabla_x f(\tilde{x}_t, \tilde{y}_t) = \frac{1}{n}\sum_{i=1}^n \nabla_x f_i(\tilde{x}_t, \tilde{y}_t)$ and $\nabla_y f(\tilde{x}_t, \tilde{y}_t) = \frac{1}{n}\sum_{i=1}^n \nabla_y f_i(\tilde{x}_t, \tilde{y}_t)$
    for all $j = 0$ to $N-1$ **do**
      sample i.i.d. indices i_j^1, i_j^2 uniformly from $[n]$
      $x_{t,j+1} = x_{t,j} - \tau_1[\nabla_x f_{i_j^1}(x_{t,j}, y_{t,j}) - \nabla_x f_{i_j^1}(\tilde{x}_t, \tilde{y}_t) + \nabla_x f(\tilde{x}_t, \tilde{y}_t)]$
      $y_{t,j+1} = y_{t,j} + \tau_2[\nabla_y f_{i_j^2}(x_{t,j+1}, y_{t,j}) - \nabla_y f_{i_j^2}(\tilde{x}_t, \tilde{y}_t) + \nabla_y f(\tilde{x}_t, \tilde{y}_t)]$
    end for
    $\tilde{x}_{t+1} = x_{t,N}, \quad \tilde{y}_{t+1} = y_{t,N}$
  end for
  choose (x^k, y^k) from $\{\{(x_{t,j}, y_{t,j})\}_{j=0}^{N-1}\}_{t=0}^{T-1}$ uniformly at random
  $\tilde{x}_0 = x^k, \quad \tilde{y}_0 = y^k$
end for

3.4 STOCHASTIC VARIANCE-REDUCED ALGORITHM

In this subchapter, we study the minimax problem with the finite-sum structure: $\min_x \max_y f(x,y) = \frac{1}{n}\sum_{i=1}^n f_i(x,y)$, which arises ubiquitously in machine learning. We are especially interested in the case when n is large. We assume the overall objective function $f(x,y)$ satisfies the two-sided PL condition with μ_1 and μ_2, but do not assume each f_i to satisfy the two-sided PL condition. Instead of Assumption 10, we assume each component f_i has Lipschitz gradients.

Assumption 9. *Each f_i has l-Lipschitz gradients.*

If we run AGDA with full gradients to solve the finite-sum minimax problem, the total complexity for finding an ϵ-optimal solution is $\mathcal{O}(n\kappa^3 \log(1/\epsilon))$ by Theorem 3.3.4. Despite the linear convergence, the per-iteration cost is high and the complexity can be huge when the number of components n and condition number κ are large. Instead, if we run Stoc-AGDA, this leads to the total complexity $\mathcal{O}\left(\frac{\kappa^5\sigma^2}{\mu_2\epsilon}\right)$ by Remark 3.3.6, which has worse dependence on ϵ.

Motivated by the recent success of stochastic variance reduced gradient (SVRG) technique [Johnson and Zhang, 2013, Reddi et al., 2016a, Palaniappan and Bach, 2016], we introduce the VR-AGDA algorithm (presented in Algorithm 5), that combines AGDA with SVRG so that the linear convergence is preserved while improving the dependency on n and κ. VR-AGDA can be viewed as the applying SVRG to AGDA with restarting: at every epoch k, we restart the SVRG subroutine by initializing it with (x^k, y^k), which is randomly

selected from previous SVRG subroutine. This is partly inspired by the GD-SVRG algorithm for minimizing PL functions [Reddi et al., 2016a]. Notice when $T = 1$, VR-AGDA reduces to a double-loop algorithm which is similar to the SVRG for saddle point problems proposed by Palaniappan and Bach [2016], except for several notable differences: (i) we are using the alternating updates rather than simultaneous updates, (ii) as a result, we require to sample two independent indices rather than one at each iteration, and (iii) most importantly, we are dealing with possibly nonconvex-nonconcave objectives that satisfy the two-sided PL condition. The following two theorems capture the convergence of VR-AGDA under different hyper-parameter setups.

Theorem 3.4.1. *Suppose Assumptions 7 and 9 hold and $f(x, y)$ satisfies the two-sided PL condition with μ_1 and μ_2. Define $P_k = a^k + \frac{1}{20}b^k$, where $a^k = \mathbb{E}[g(x^k) - g^*]$ and $b^k = \mathbb{E}[g(x^k) - f(x^k, y^k)]$. If we run VR-AGDA with $\tau_1 = \beta/(28\kappa^8 l)$, $\tau_2 = \beta/(l\kappa^6)$, $N = \lfloor \alpha\beta^{-2/3}\kappa^9(2 + 4\beta^{1/2}\kappa^{-3})^{-1} \rfloor$ and $T = 1$, where α, β are constants irrelevant to l, n, μ_1, μ_2, then $P_{k+1} \leq \frac{1}{2}P_k$. This implies complexity of*

$$\mathcal{O}\big((n + \kappa^9)\log(1/\epsilon)\big)$$

total for VR-AGDA to achieve an ϵ-optimal solution.

Theorem 3.4.2. *Under the same assumptions as Theorem 3.4.1 , if we run VR-AGDA with $\tau_1 = \beta/(28\kappa^2 l n^{2/3})$, $\tau_2 = \beta/(l n^{2/3})$, $N = \lfloor \alpha\beta^{-2/3} n(2 + 4\beta^{1/2} n^{-1/3})^{-1} \rfloor$, and $T = \lceil \kappa^3 n^{-1/3} \rceil$, where α, β are constants irrelevant to l, n, μ_1, μ_2, then $P_{k+1} \leq \frac{1}{2}P_k$. This implies complexity of*

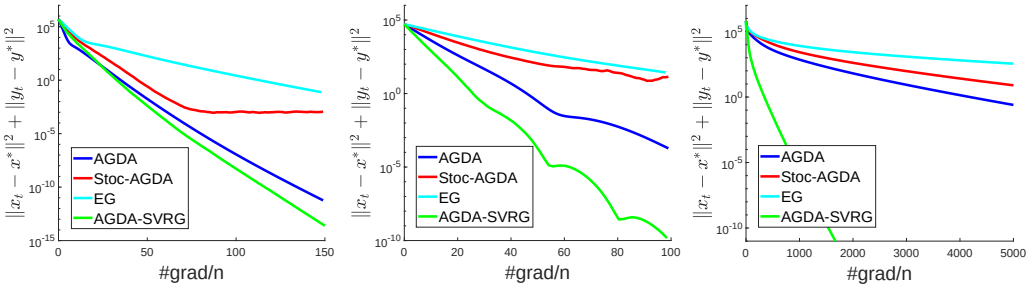
$$\mathcal{O}\big((n + n^{2/3}\kappa^3)\log(1/\epsilon)\big)$$

for VR-AGDA to achieve an ϵ-optimal solution.

Remark 3.4.3. *Theorems 3.4.1 and 3.4.2 are different in their choices of stepsizes and iteration numbers, which gives rise to different complexities. VR-AGDA with the second setting has a lower complexity than the first setting in the regime $n \leq \kappa^9$, but the first setting allows for a simpler double-loop algorithm with $T = 1$. The two theorems imply that VR-AGDA always improves over AGDA. To the best of our knowledge, this is also the first theoretical analysis of variance-reduced algorithms with alternating updating rules for minimax optimization.*

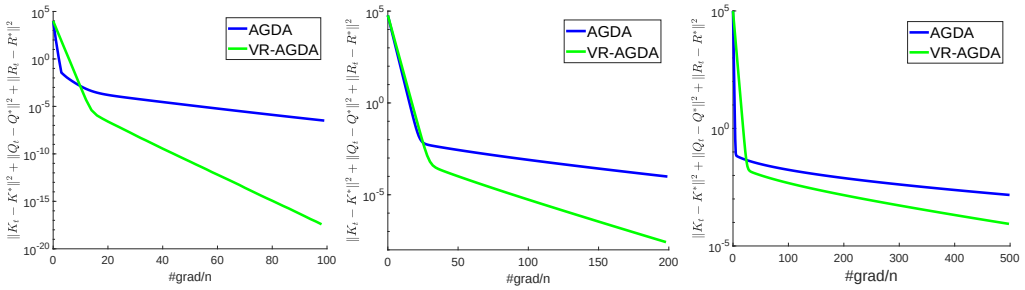
3.5 EXPERIMENTS

We present experiments on two applications: robust least square and imitation learning for LQR. We mainly focus on the comparison between AGDA, Stoc-AGDA, and VR-AGDA, which are the only algorithms with known theoretical guarantees. Because of their simplicity, only few hyperparameters are induced and are tuned based on grid search.



(a) Dataset 1 (b) Dataset 2 (c) Dataset 3

FIGURE 3.3: GDA, Stoc-AGDA and SVRG-AGDA for robust least square.



(a) $d = 3, k = 2$ (b) $d = 20, k = 10$ (c) $d = 30, k = 20$

FIGURE 3.4: AGDA and VR-AGDA on generative adversarial learning for LQR

3.5.1 *Robust Least Square*

We consider the least square problems with the coefficient matrix $A \in \mathbb{R}^{n\times m}$ and noisy vector $y_0 \in \mathbb{R}^n$ subject to bounded deterministic perturbation δ. Robust least square (RLS) minimizes the worst-case residual, and can be formulated as [El Ghaoui and Lebret, 1997]: $\min_x \max_{\delta:\|\delta\|\leq\rho} \|Ax - y\|^2$, where $\delta = y_0 - y$. We consider RLS with soft constraint:

$$\min_x \max_y F(x, y) := \|Ax - y\|_M^2 - \lambda\|y - y_0\|_M^2, \tag{3.9}$$

where we adopt the general M-(semi-)norm in: $\|x\|_M^2 = x^T Mx$ and M is positive semi-definite. $F(x, y)$ satisfies the two-sided PL condition when $\lambda > 1$, because it can be written as the composition of a strongly-convex-strongly-concave function and an affine function (Example 2). However, $F(x, y)$ is not strongly convex about x, and when M is not full-rank, it is not strongly concave about y.

Datasets. We use three datasets in the experiments, and two of them are generated in the same way as in Du and Hu [2019]. We generate the first dataset with $n = 1000$ and $m = 500$ by sampling rows of A from a Gaussian $\mathcal{N}(0, I_n)$ distribution and setting $y_0 = Ax^* + \epsilon$

with x^* from Gaussian $\mathcal{N}(0,1)$ and ϵ from Gaussian $\mathcal{N}(0,0.01)$. We set $M = I_n$ and $\lambda = 3$. The second dataset is the rescaled aquatic toxicity dataset by Cassotti et al. [2014], which uses 8 molecular descriptors of 546 chemicals to predict quantitative acute aquatic toxicity towards Daphnia Magna. We use $M = I$ and $\lambda = 2$ for this dataset. The third dataset is generated with $A \in \mathbb{R}^{1000\times 500}$ from Gaussian $\mathcal{N}(0,\Sigma)$ where $\Sigma_{i,j} = 2^{-|i-j|/10}$, M being rank-deficit with positive eigenvalues sampled from $[0.2, 1.8]$ and $\lambda = 1.5$. These three datasets represent cases with low, median, and high condition numbers, respectively.

Evaluation. We compare four algorithms: AGDA, Stoc-AGDA, VR-AGDA and extra-gradient (EG) with fine-tuned stepsizes. For Stoc-AGDA, we choose constant stepsizes to form a fair comparison with the other two. We report the potential function value, i.e., P_t described in our theorems, and distance to the limit point $\|(x_t, y_t) - (x^*, y^*)\|^2$. These errors are plotted against the number of gradient evaluations normalized by n (i.e., number of full gradients). Results are reported in Figure 3.3. We observe that VR-AGDA and AGDA both exhibit linear convergence, and the speedup of VR-AGDA is fairly significant when the condition number is large, whereas Stoc-AGDA progresses fast at the beginning and stagnates later on. These numerical results clearly validate our theoretical findings. EG performs poorly in this example.

3.5.2 *Generative Adversarial Imitation Learning for LQR*

The optimal control problem for LQR can be formulated as [Fazel et al., 2018]:

$$\underset{\pi_t}{\text{minimize}} \quad \mathbb{E}_{x_0\sim\mathcal{D}} \sum_{t=0}^{\infty} x_t^\top Q x_t + u_t^\top R u_t \quad \text{such that} \quad x_{t+1} = Ax_t + Bu_t, u_t = \pi_t(x_t),$$

where $x_t \in \mathbb{R}^d$ is a state, $u_t \in \mathbb{R}^k$ is a control, $\mathcal{D}$ is the distribution of initial state x_0, and π_t is a policy. It is known that the optimal policy is linear: $u_t = -K^* x_t$, where $K^* \in \mathbb{R}^{k\times d}$. If we parametrize the policy in the linear form, $u_t = -Kx_t$, the problem can be written as: $\min_K C(K; Q, R) := \mathbb{E}_{x_0\sim\mathcal{D}}\left[\sum_{t=0}^{\infty}\left(x_t^\top Q x_t + (Kx_t)^\top R(Kx_t)\right)\right]$ where the trajectory is induced by LQR dynamics and policy K. In generative adversarial imitation learning for LQR, the trajectories induced by an expert policy K_E are observed and part of the goal is to learn the cost function parameters Q and R from the expert. This can be formulated as a minimax problem [Cai et al., 2019]:

$$\min_K \max_{(Q,R)\in\Theta} \Big\{ m(K, Q, R) := C(K; Q, R) - C(K_E; Q, R) - \Phi(Q, R) \Big\},$$

where $\Theta = \{(Q,R) : \alpha_Q I \preceq Q \preceq \beta_Q I, \alpha_R I \preceq R \preceq \beta_R I\}$ and Φ is a strongly-convex regularizer. We sample n initial points $x_0^{(1)}, x_0^{(2)}, ..., x_0^{(n)}$ from $\mathcal{D}$ and approximate $C(K;Q,R)$ by sample average $C_n(K;Q,R) := \frac{1}{n}\sum_{i=1}^{n}\left[\sum_{t=0}^{\infty}\left(x_t^\top Q x_t + u_t^\top R u_t\right)\right]_{x_0=x_0^{(i)}}$. We then consider:

$$\min_K \max_{(Q,R)\in\Theta} \{m_n(K,Q,R) := C_n(K;Q,R) - C_n(K_E;Q,R) - \Phi(Q,R)\}. \tag{3.10}$$

Note that m_n satisfies the PL condition in terms of K [Fazel et al., 2018], and m_n is strongly-concave in terms of (Q,R), so the function satisfies the two-sided PL condition.

In our experiment, we use $\Phi(Q,R) = \lambda(\|Q-\bar{Q}\|^2 + \|R-\bar{R}\|^2)$ for some $\bar{Q},\bar{R}$ and $\lambda = 1$. We generate a dataset with different n and k: (1) $d = 3, k = 2$; (2) $d = 20, k = 10$; (3) $d = 30, k = 20$. The initial distribution $\mathcal{D}$ is $\mathcal{N}(0, I_d)$ and we sample $n = 100$ initial points. The exact gradients can be computed based on the compact forms established in [Fazel et al., 2018, Cai et al., 2019]. We compare AGDA and VR-AGDA under fine-tuned stepsizes, and track their errors in terms of $\|K_t - K^*\|^2 + \|Q_t - Q^*\|_F^2 + \|R_t - R^*\|_F^2$. The result is reported in Figure 3.4, which again indicates that VR-AGDA significantly outperforms AGDA.

3.6 APPENDIX

3.6.1 *Proofs for Chapter 3.2*

We first present several key lemmas.

Lemma 3.6.1 (Karimi et al. [2016])**.** *If $f(\cdot)$ is l-smooth and it satisfies PL with constant μ, then it also satisfies error bound (EB) condition with μ, i.e.*

$$\|\nabla f(x)\| \geq \mu \|x_p - x\|, \forall x,$$

where x_p is the projection of x onto the optimal set, also it satisfies quadratic growth (QG) condition with μ, i.e.

$$f(x) - f^* \geq \frac{\mu}{2}\|x_p - x\|^2, \forall x.$$

Conversely, if $f(\cdot)$ is l-smooth and it satisfies EB with constant μ, then it satisfies PL with constant μ/l.

From the above lemma, we easily derive that $l \geq \mu$.

Lemma 3.6.2 (Nouiehed et al. [2019])**.** *In the minimax problem, when $-f(x,\cdot)$ satisfies PL condition with constant μ_2 for any x and f satisfies Assumption 10, then the function $g(x) := \max_y f(x,y)$ is L-smooth with $L := l + l^2/\mu_2$ and $\nabla g(x) = \nabla_x f(x, y^*(x))$ for any $y^*(x) \in \arg\max_y f(x,y)$.*

Lemma 3.6.3. *In the minimax problem 4.1, when the objective function f satisfies Assumption 10 (Lipschitz gradient) and the two-sided PL condition with constant μ_1 and μ_2, then function $g(x) := \max_y f(x,y)$ satisfies the PL condition with μ_1.*

Proof. From Lemma 4.6.3,

$$\|\nabla g(x)\|^2 = \|\nabla_x f(x, y^*(x))\|^2.$$

Since $f(\cdot, y)$ satisfies PL condition with constant μ_1, we get

$$\|\nabla g(x)\|^2 \geq 2\mu_1[f(x, y^*(x)) - \min_{x'} f(x', y^*(x))]. \tag{3.11}$$

Also,

$$f(x', y^*(x)) \leq \max_y f(x', y) \Longrightarrow \min_{x'} f(x', y^*(x)) \leq \min_{x'} \max_y f(x', y) = g^*. \tag{3.12}$$

Combining equation (3.11) and (3.12), we obtain,

$$\|\nabla g(x)\|^2 \geq 2\mu_1(g(x) - g^*).$$

□

The following lemma states that stochastic gradient descent converges linearly to the neighbourhood of the optimal set under PL condition. The proof is based on [Karimi et al., 2016].

Lemma 3.6.4. *Consider the optimization problem* $\min_x f(x) = \mathbb{E}[F(x;\xi)]$*, where* f *is* l*-smooth and satisfies PL condition with constant* μ*. Using the stochastic gradient descent with stepsize* $\tau \leq 1/l$*,*

$$x_{t+1} = x_t - \tau G(x_t, \xi_t),$$

where

$$\mathbb{E}[G(x,\xi) - \nabla f(x)] = 0, \qquad \mathbb{E}[\|G(x,\xi) - \nabla f(x)\|^2] \leq \sigma^2,$$

then we have

$$\mathbb{E}[f(x_{t+1}) - f^*] \leq (1 - \mu\tau)\mathbb{E}[f(x_t) - f^*] + \frac{l\tau^2}{2}\sigma^2.$$

Proof. By smoothness of f we have

$$\begin{aligned} f(x_{t+1}) - f^* &\leq f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{l}{2}\|x_{t+1} - x\|^2 - f^* \\ &= f(x_t) - \tau\langle \nabla f(x_t), G(x_t, \xi_t)\rangle + \frac{l\tau^2}{2}\|G(x_t,\xi_t)\|^2 - f^*. \end{aligned}$$

Taking expectations of both sides, we get

$$\begin{aligned} \mathbb{E}[f(x_{t+1}) - f^*] &\leq \mathbb{E}[f(x_t) - f^*] - \tau\mathbb{E}[\|\nabla f(x_t)\|^2] + \frac{l\tau^2}{2}\mathbb{E}[\|G(x_t,\xi_t)\|^2] \\ &= \mathbb{E}[f(x_t) - f^*] - \tau\mathbb{E}[\|\nabla f(x_t)\|^2] + \frac{l\tau^2}{2}\mathbb{E}[\|\nabla f(x_t)\|^2] \\ &\quad + \frac{l\tau^2}{2}\mathbb{E}[\|\nabla f(x_t) - G(x_t,\xi_t)\|^2] \\ &\leq \mathbb{E}[f(x_t) - f^*] - \frac{\tau}{2}\mathbb{E}[\|\nabla f(x_t)\|^2] + \frac{l\tau^2}{2}\sigma^2 \\ &\leq (1 - \mu\tau)\mathbb{E}[f(x_t) - f^*] + \frac{l\tau^2}{2}\sigma^2, \end{aligned}$$

where in the equality we use $\mathbb{E}[G(x_t,\xi_t)] = \nabla f(x_t)$, in the second inequality we use $\tau \leq 1/l$, and we use PL condition in the last inequality. □

Proof for Lemma 3.2.1

Proof. • (stationary point) $\Longrightarrow$ (saddle point): From the definition of PL condition, if (x^*, y^*) is a stationary point,

$$\max_y f(x^*, y) - f(x^*, y^*) \leq \frac{1}{2\mu_2}\left\|\nabla_y f(x^*, y^*)\right\|^2 = 0,$$

$$f(x^*, y^*) - \min_x f(x, y^*) \leq \frac{1}{2\mu_1} \|\nabla_x f(x^*, y^*)\|^2 = 0,$$

so $\max_y f(x^*, y) = f(x^*, y^*) = \min_x f(x, y^*)$, and therefore $f(x^*, y^*)$ is a saddle point.

- (saddle point) $\Longrightarrow$ (global minimax point): Follow from definitions.
- (global minimax point) $\Longrightarrow$ (stationary point): If (x^*, y^*) is a global minimax point, then by definition,

  $$y^* \in \arg\max_y f(x^*, y^*), x^* \in \arg\min_x g(x),$$

  Then by first order necessary condition, we have,

  $$\nabla_y f(x^*, y^*) = 0, \nabla g(x^*) = 0,$$

  Further with Lemma 4.6.3,

  $$\nabla g(x^*) = \nabla_x f(x^*, y^*) = 0$$

  Thus, (x^*, y^*) is a stationary point.

□

Proposition 1. *The function*

$$f(x, y) = x^2 + 3\sin^2 x \sin^2 y - 4y^2 - 10\sin^2 y,$$

satisfies the two-sided PL condition with $\mu_1 = 1/16, \mu_2 = 1/14$.

Proof. It is not hard to derive that $\arg\min_x f(x, y) = 0, \forall y$, and $\arg\max_y f(x, y) = 0, \forall x$, i.e. $x^*(y) = y^*(x) = 0, \forall x, y$. Therefore, $(0, 0)$ is the only saddle point. Then compute the gradients:

$$\nabla_x f(x, y) = 2x + 3\sin^2(y)\sin(2x),$$
$$\nabla_y f(x, y) = -8y + 3\sin^2(x)\sin(2y) - 10\sin(2y).$$

and

$$|\nabla_x^2 f(x, y)| = |2 + 6\sin^2(y)\cos(2x)| \leq 8,$$
$$|\nabla_y^2 f(x, y)| = |-8 + 6\sin^2(x)\cos(2y) - 20\cos(2y)| \leq 28.$$

so $f(\cdot, y)$ is L_1-smooth with $L_1 = 8$ for any x and $f(x, \cdot)$ is L_2-smooth with $L_2 = 28$ for any y. Then note that:

$$\frac{|\nabla_x f(x, y)|}{|x - x^*(y)|} = \frac{|\nabla_x f(x, y)|}{|x|} = \frac{|2x + 3\sin^2(y)\sin(2x)|}{|x|} \geq \frac{1}{2},$$
$$\frac{|\nabla_y f(x, y)|}{|y - y^*(x)|} = \frac{|\nabla_y f(x, y)|}{|y|} = \frac{|-8y + 3\sin^2(x)\sin(2y) - 10\sin(2y)|}{|y|} \geq 2.$$

So $f(\cdot, y)$ satisfies EB with $\mu_{EB1} = 1/2$, and -$f(x, \cdot)$ satisfies EB with $\mu_{EB2} = 2$. By Lemma 4.6.2, we have $f(\cdot, y)$ satisfies PL with constant $\mu_1 = 1/16$ and -$f(x, \cdot)$ satisfies PL with constant $\mu_1 = 1/14$.

□

3.6.2 *Proofs for Chapter 3.3*

Before we step into proofs for Theorem 3.3.1, 3.3.4 and 3.3.5, we first present a contraction theorem for each iteration.

Theorem 3.6.5. *Assume Assumption 10, 7, 12 hold and $f(x, y)$ satisfies the two-sided PL condition with μ_1 and μ_2. Define $a_t = \mathbb{E}[g(x_t) - g^*]$ and $b_t = \mathbb{E}[g(x_t) - f(x_t, y_t)]$. If we run one iteration of Algorithm 4 with $\tau_1^t = \tau_1 \leq 1/L$ (L is specified in Lemma 4.6.3) and $\tau_2^t = \tau_2 \leq 1/l$, then*

$$a_{t+1} + \lambda b_{t+1} \leq \max\{k_1, k_2\}(a_t + \lambda b_t) + \lambda(1 - \mu_2\tau_2)\frac{L + l}{2}\tau_1^2\sigma^2 + \frac{l}{2}\lambda\tau_2^2\sigma^2 + \frac{L}{2}\tau_1^2\sigma^2,$$

where

$$k_1 := 1 - \mu_1\big[\tau_1 + \lambda(1 - \mu_2\tau_2)\tau_1 - \lambda(1 + \beta)(1 - \mu_2\tau_2)(2\tau_1 + l\tau_1^2)\big], \tag{3.13}$$

$$k_2 := 1 - \mu_2\tau_2 + \frac{l^2\tau_1}{\mu_2\lambda} + (1 - \mu_2\tau_2)\frac{l^2}{\mu_2}\tau_1 + (1 + \frac{1}{\beta})(1 - \mu_2\tau_2)\frac{l^2}{\mu_2}(2\tau_1 + l\tau_1^2), \tag{3.14}$$

and $\lambda, \beta > 0$ such that $k_1 \leq 1$.

Proof. Because g is L-smooth by Lemma 4.6.3, we have

$$\begin{aligned} g(x_{t+1}) - g^* &\leq g(x_t) - g^* + \langle \nabla g(x_t), x_{t+1} - x_t \rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2 \\ &= g(x_t) - g^* - \tau_1\langle \nabla g(x_t), G_x(x_t, y_t, \xi_{t1}) \rangle + \frac{L}{2}\tau_1^2\|G_x(x_t, y_t, \xi_{t1})\|^2. \end{aligned}$$

Taking expectations of both sides and using Assumption 12, we get

$$\begin{aligned} &\mathbb{E}[g(x_{t+1}) - g^*] \\ &\leq \mathbb{E}[g(x_t) - g^*] - \tau_1\mathbb{E}[\langle \nabla g(x_t), \nabla_x f(x_t, y_t) \rangle] + \frac{L}{2}\tau_1^2\mathbb{E}[\|G_x(x_t, y_t, \xi_{t1})\|^2] \\ &\leq \mathbb{E}[g(x_t) - g^*] - \tau_1\mathbb{E}[\langle \nabla g(x_t), \nabla_x f(x_t, y_t) \rangle] + \frac{L}{2}\tau_1^2\mathbb{E}[\|\nabla_x f(x_t, y_t)\|^2] + \frac{L}{2}\tau_1^2\sigma^2 \\ &\leq \mathbb{E}[g(x_t) - g^*] - \tau_1\mathbb{E}[\langle \nabla g(x_t), \nabla_x f(x_t, y_t) \rangle] + \frac{\tau_1}{2}\mathbb{E}[\|\nabla_x f(x_t, y_t)\|^2] + \frac{L}{2}\tau_1^2\sigma^2 \\ &\leq \mathbb{E}[g(x_t) - g^*] - \frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_t)\|^2 + \frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 + \frac{L}{2}\tau_1^2\sigma^2, \end{aligned} \tag{3.15}$$

where in the second inequality we use Assumption 12, and in the third inequality we use $\tau_1 \leq 1/L$. Because $-f(x_{t+1}, y)$ is l-smooth and μ_1-PL, by Lemma 3.6.4, when $\tau_1 \leq 1/l$ we have

$$\mathbb{E}[g(x_{t+1}) - f(x_{t+1}, y_{t+1})]$$

$$\leq (1-\mu_2\tau_2)\mathbb{E}[g(x_{t+1}) - f(x_{t+1}, y_t)] + \frac{l}{2}\tau_2^2\sigma^2$$
$$\leq (1-\mu_2\tau_2)\mathbb{E}[g(x_t) - f(x_t, y_t) + f(x_t, y_t) - f(x_{t+1}, y_t) + g(x_{t+1}) - g(x_t)] + \frac{l}{2}\tau_2^2\sigma^2. \tag{3.16}$$

Because of the Lipschitz continuity of the gradient, we can bound $f(x_t, y_t) - f(x_{t+1}, y_t)$ as

$$f(x_t, y_t) - f(x_{t+1}, y_t) \leq -\langle \nabla_x f(x_t, y_t), x_{t+1} - x_t\rangle + \frac{l}{2}\|x_{t+1} - x_t\|^2$$
$$\leq \tau_1\langle \nabla_x f(x_t, y_t), G_x(x_t, y_t, \xi_{t1})\rangle + \frac{l}{2}\tau_1^2\|G_x(x_t, y_t, \xi_{t1})\|^2.$$

Taking expectations of both sides and using Assumption 12,

$$\mathbb{E}[f(x_t, y_t) - f(x_{t+1}, y_t)] \leq (\tau_1 + \frac{l}{2}\tau_1^2)\mathbb{E}\|\nabla_x f(x_t, y_t)\|^2 + \frac{l}{2}\tau_1^2\sigma^2. \tag{3.17}$$

Also from (3.15) ,

$$\mathbb{E}[g(x_{t+1}) - g(x_t)] \leq -\frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_t)\|^2 + \frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 + \frac{L}{2}\tau_1^2\sigma^2. \tag{3.18}$$

Combining (3.16), (3.17) and (3.18),

$$\mathbb{E}[g(x_{t+1}) - f(x_{t+1}, y_{t+1})]$$
$$\leq (1-\mu_2\tau_2)\mathbb{E}[g(x_t) - f(x_t, y_t)] + (1-\mu_2\tau_2)(\tau_1 + \frac{l}{2}\tau_1^2)\mathbb{E}\|\nabla_x f(x_t, y_t)\|^2 -$$
$$(1-\mu_2\tau_2)\frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_t)\|^2 + (1-\mu_2\tau_2)\frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 +$$
$$(1-\mu_2\tau_2)\frac{L+l}{2}\tau_1^2\sigma^2 + \frac{l}{2}\tau_2^2\sigma^2. \tag{3.19}$$

Combining (3.15) and (3.19), we have $\forall \lambda > 0$,

$$a_{t+1} + \lambda b_{t+1}$$
$$\leq a_t - \left[\frac{\tau_1}{2} + \lambda(1-\mu_2\tau_1)\frac{\tau_1}{2}\right]\mathbb{E}\|\nabla g(x_t)\|^2 + \lambda(1-\mu_2\tau_2)b_t +$$
$$\left[\frac{\tau_1}{2} + \lambda(1-\mu_2\tau_2)\frac{\tau_1}{2}\right]\mathbb{E}\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 + \lambda(1-\mu_2\tau_2)\left(\tau_1 + \frac{l}{2}\tau_1^2\right)\mathbb{E}\|\nabla_x f(x_t, y_t)\|^2 +$$
$$\lambda(1-\mu_2\tau_2)\frac{L+l}{2}\tau_1^2\sigma^2 + \frac{l}{2}\lambda\tau_2^2\sigma^2 + \frac{L}{2}\tau_1^2\sigma^2$$
$$\leq a_t - \left[\frac{\tau_1}{2} + \lambda(1-\mu_2\tau_1)\frac{\tau_1}{2} - \lambda(1+\beta)(1-\mu_2\tau_2)\left(\tau_1 + \frac{l}{2}\tau_1^2\right)\right]\mathbb{E}\|\nabla g(x_t)\|^2 + \lambda(1-\mu_2\tau_2)b_t +$$
$$\left[\frac{\tau_1}{2} + \lambda(1-\mu_2\tau_2)\frac{\tau_1}{2} + \lambda\left(1+\frac{1}{\beta}\right)(1-\mu_2\tau_2)\left(\tau_1 + \frac{l}{2}\tau_1^2\right)\right]\mathbb{E}\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 +$$
$$\lambda(1-\mu_2\tau_2)\frac{L+l}{2}\tau_1^2\sigma^2 + \frac{l}{2}\lambda\tau_2^2\sigma^2 + \frac{L}{2}\tau_1^2\sigma^2, \tag{3.20}$$

where in the second inequality we use Young's Inequality and $\beta > 0$. Now it suffices to bound $\nabla\|g(x_t)\|^2$ and $\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2$ by a_t and b_t. With Lemma 4.6.3, we have:

$$\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 = \|\nabla_x f(x_t, y_t) - \nabla_x f(x_t, y^*(x_t))\|^2 \leq l^2\|y^*(x_t) - y_t\|^2, \tag{3.21}$$

for any $y^*(x_t) \in \arg\max_y f(x_t, y)$. Now we fix $y^*(x_t)$ to be the projection of y_t on the the set $\arg\max_y f(x_t, y)$. Because $-f(\boldsymbol{x_t}, \cdot)$ satisfies PL condition with μ_2, and Lemma 4.6.2 therefore indicates it also satisfies quadratic growth condition with μ_2, i.e.

$$\|y^*(x_t) - y_t\|^2 \leq \frac{2}{\mu_2}[g(x_t) - f(x_t, y_t)], \tag{3.22}$$

along with (3.21), we get

$$\|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 \leq \frac{2l^2}{\mu_2}[g(x_t) - f(x_t, y_t)]. \tag{3.23}$$

Because g satisfies PL condition with μ_1 by Lemma 3.6.3,

$$\|\nabla g(x_t)\|^2 \geq 2\mu_1[g(x_t) - g^*]. \tag{3.24}$$

Plugging (3.23) and (3.24) into (3.20), we can get

$$\begin{aligned} a_{t+1} + \lambda b_{t+1} \leq & \Big\{1 - \mu_1\big[\tau_1 + \lambda(1 - \mu_2\tau_2)\tau_1 - \lambda(1+\beta)(1-\mu_2\tau_2)(2\tau_1 + l\tau_1^2)\big]\Big\}a_t + \\ & \lambda\Big\{1 - \mu_2\tau_2 + \frac{l^2\tau_1}{\mu_2\lambda} + (1-\mu_2\tau_2)\frac{l^2}{\mu_2}\tau_1 + (1+\frac{1}{\beta})(1-\mu_2\tau_2)\frac{l^2}{\mu_2}(2\tau_1 + l\tau_1^2)\Big\}b_t + \\ & \lambda(1-\mu_2\tau_2)\frac{L+l}{2}\tau_1^2\sigma^2 + \frac{l}{2}\lambda\tau_2^2\sigma^2 + \frac{L}{2}\tau_1^2\sigma^2. \end{aligned} \tag{3.25}$$

□

PROOF OF THEOREM 3.3.1

Proof. In the setting of Theorem 1, $\tau_1^t = \tau_1$ and $\tau_2^t = \tau_2, \forall t$. By Thoerem 3.6.5, we only need to choose τ_1, τ_2, λ and β to let $k_1, k_2 < 1$. Here we first choose $\beta = 1$ and $\lambda = 1/10$. Then

$$\begin{aligned} k_1 &= 1 - \mu_1\big[\tau_1 + \lambda(1-\mu_2\tau_2)\tau_1 - \lambda(1+\beta)(1-\mu_2\tau_2)(2\tau_1 + l\tau_1^2)\big] \\ &\leq 1 - \mu_1\big\{\tau_1 - \lambda(1-\mu_2\tau_2)\tau_1[(1+\beta)(2+l\tau_1) - 1]\big\} \leq 1 - \frac{1}{2}\tau_1\mu_1, \end{aligned} \tag{3.26}$$

where in the last inequality we just plug in β and λ and use $l\tau_1 \leq 1$. Also,

$$\begin{aligned} k_2 &= 1 - \mu_2\tau_2 + \frac{l^2\tau_1}{\mu_2\lambda} + (1-\mu_2\tau_2)\frac{l^2}{\mu_2}\tau_1 + (1+\frac{1}{\beta})(1-\mu_2\tau_2)\frac{l^2}{\mu_2}(2\tau_1 + l\tau_1^2) \\ &\leq 1 - \frac{l^2\tau_1}{\mu_2}\left\{\frac{\mu_2^2\tau_2}{\tau_1 l^2} - \frac{1}{\lambda} - (1-\mu_2\tau_2)\left[1 + \left(1 + \frac{1}{\beta}\right)(2 + l\tau_1)\right]\right\} \\ &\leq 1 - \frac{l^2\tau_1}{\mu_2}, \end{aligned} \tag{3.27}$$

where in the last inequality we plug in β and λ and we use $\frac{\mu_2^2\tau_2}{\tau_1 l^2} \leq 18$ by our choice of τ_1. Note that $\frac{1}{2}\tau_1\mu_1 < \frac{l^2\tau_1}{\mu_2}$, because $\left(\frac{1}{2}\tau_1\mu_1\right) / \left(\frac{l^2\tau_1}{\mu_2}\right) = \frac{\mu_1\mu_2}{2l^2} < 1$. Define $P_t := a_t + \frac{1}{10}b_t$. By Theorem 3.6.5,

$$P_{t+1} \leq \left(1 - \frac{1}{2}\tau_1\mu_1\right)P_t + \frac{(1-\mu_2\tau_2)(L+l)\tau_1^2}{20}\sigma^2 + \frac{l\tau_2^2}{20}\sigma^2 + \frac{L\tau_1^2}{2}\sigma^2.$$

With some simple computation,

$$P_t \leq (1 - \frac{1}{2}\mu_1\tau_1)^t P_0 + \frac{(1-\mu_2\tau_2)(L+l)\tau_1^2 + l\tau_2^2 + 10L\tau_1^2}{10\mu_1\tau_1}\sigma^2.$$

We verify that $\tau_1 \leq 1/L$ by noting: $\tau_1 \leq \frac{\mu_2^2\tau_2}{18l^2} \leq \frac{\mu_2^2}{18l^3} \leq \frac{\mu_2}{2l^2}$ and $L = l + \frac{l^2}{\mu_2} \leq \frac{2l^2}{\mu_2}$. □

PROOF OF THEOREM 3.3.4

Proof. The first part of Theorem 3.3.4 is a direct corollary of Theorem 3.3.1 by setting $\sigma = 0$. We show the second part by noting that

$$\|x_{t+1} - x_t\|^2 = \tau_1^2 \|\nabla_x f(x_t, y_t)\|^2, \text{and } \|y_{t+1} - y_t\|^2 = \tau_2^2 \|\nabla_y f(x_{t+1}, y_t)\|^2. \tag{3.28}$$

Also,

$$\begin{aligned}
\|\nabla_y f(x_{t+1}, y_t)\|^2 &\leq \|\nabla_y f(x_t, y_t)\|^2 + \|\nabla_y f(x_{t+1}, y_t) - \nabla_y f(x_t, y_t)\|^2 \\
&\leq \|\nabla_y f(x_t, y_t) - \nabla_y f(x_t, y^*(x_t))\|^2 + l^2\|x_{t+1} - x_t\|^2 \\
&\leq l^2\|y_t - y^*(x_t)\|^2 + l^2\|x_{t+1} - x_t\|^2 \\
&\leq \frac{2l^2}{\mu_2} b_t + l^2\|x_{t+1} - x_t\|^2 = \frac{2l^2}{\mu_2} b_t + l^2\tau_1^2\|\nabla_x f(x_t, y_t)\|^2,
\end{aligned} \tag{3.29}$$

where in the second inequality $y^*(x_t)$ is the projection of y_t on the set $\arg\max_y f(x_t, y)$ and $\nabla_y f(x_t, y^*(x_t)) = 0$, in the third inequality we use the Lipschitz continuity of gradient, and in the last inequality we use quadratic growth condition. Also,

$$\begin{aligned}
\|\nabla_x f(x_t, y_t)\|^2 &\leq \|\nabla g(x_t)\|^2 + \|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 \\
&= \|\nabla g(x_t) - \nabla g(x^*)\|^2 + \|\nabla_x f(x_t, y_t) - \nabla g(x_t)\|^2 \\
&\leq L^2\|x_t - x^*\|^2 + l^2\|y^*(x_t) - y_t\|^2 \\
&\leq \frac{2L^2}{\mu_1} a_t + \frac{2l^2}{\mu_2} b_t,
\end{aligned} \tag{3.30}$$

where in the first equality x^* is the projection of x_t on the set $\arg\min_x g(x)$ and $\nabla g(x^*) = 0$, in the second inequality $y^*(x_t)$ is the projection of y_t on the set $\arg\max_y f(x_t, y)$ and $\nabla g(x_t) = \nabla_x f(x_t, y_t)$, and in the last inequality we use quadratic growth condition. Therefore with (3.29) and (3.30),

$$\begin{aligned}
\|x_t - x_{t+1}\|^2 + \|y_t - y_{t+1}\|^2 &\leq \tau_1^2 \|\nabla_x f(x_t, y_t)\|^2 + \tau_2^2 \|\nabla_y f(x_{t+1}, y_t)\|^2 \\
&\leq (1 + \tau_2^2 l^2)\tau_1^2\|\nabla_x f(x_t, y_t)\|^2 + \frac{2l^2}{\mu_2}\tau_2^2 b_t \\
&\leq \frac{2(1+\tau_2^2 l^2)\tau_1^2 L^2}{\mu_1} a_t + \frac{2(1+\tau_2^2 l^2)\tau_1^2 l^2 + 2l^2\tau_2^2}{\mu_2} b_t \\
&\leq \left[\frac{2(1+\tau_2^2 l^2)\tau_1^2 L^2}{\mu_1} + \frac{20(1+\tau_2^2 l^2)\tau_1^2 l^2 + 20l^2\tau_2^2}{\mu_2}\right] P_0 c^t,
\end{aligned}$$

where $c = 1 - \frac{\mu_1\mu_2^2}{36l^3}$. Letting $\alpha_1 = \left[\frac{2(1+\tau_2^2l^2)\tau_1^2L^2}{\mu_1} + \frac{20(1+\tau_2^2l^2)\tau_1^2l^2+20l^2\tau_2^2}{\mu_2}\right] P_0$, we have

$$\|x_{t+1} - x_t\| + \|y_{t+1} - y_t\| \leq \sqrt{2\alpha_1}c^{t/2}.$$

For $n \geq t$,

$$\|x_n - x_t\| + \|y_n - y_t\| \leq \sum_{i=t}^{n-1} \|x_{i+1} - x_i\| + \|y_{i+1} - y_i\| \leq \sqrt{2\alpha_1}\sum_{i=t}^{\infty} c^{i/2} \leq \frac{\sqrt{2\alpha_1}c^{t/2}}{1-\sqrt{c}},$$

so $\{(x_t, y_t)\}_t$ converges and by first part of this theorem the limit (x^*, y^*) must be a saddle point. Thus we have

$$\|x_t - x^*\|^2 + \|y_t - y^*\|^2 \leq \frac{2\alpha_1}{(1-\sqrt{c})^2}c^t = \alpha c^t P_0,$$

with $\alpha = 2\left[\frac{2(1+\tau_2^2l^2)\tau_1^2L^2}{\mu_1} + \frac{20(1+\tau_2^2l^2)\tau_1^2l^2+20l^2\tau_2^2}{\mu_2}\right]/(1-\sqrt{c})^2$. □

PROOF OF THEOREM 3.3.5

Proof. First note that since $\tau_1^t \leq \mu_2^2/18l^2$, $\tau_2^t = \frac{18l^2\beta}{\mu_2^2(\gamma+t)} = \frac{18l^2\tau_1^t}{\mu_2^2} \leq \frac{1}{l}$. Similar to the proof of Theorem 3.3.1, by choosing $\beta = 1$ and $\lambda = 1/10$ in the Theorem 3.6.5, we have $\min\{k_1, k_2\} = \frac{1}{2}\mu_1\tau_1^t$. We prove the theorem by induction. When t = 1, it is naturally satisfied by the definition of ν. We assume that $P_t \leq \frac{\nu}{\gamma+t}$. Then by Theorem 3.6.5,

$$\begin{aligned}
P_{t+1} &\leq \left(1 - \frac{1}{2}\mu_1\tau_1\right)P_t + \lambda(1-\mu_2\tau_2^t)\frac{L+l}{2}(\tau_1^t)^2\sigma^2 + \frac{l}{2}\lambda(\tau_2^t)^2\sigma^2 + \frac{L}{2}(\tau_1^t)^2\sigma^2 \\
&\leq \frac{\gamma+t-\frac{1}{2}\mu_1\beta}{\gamma+t}\frac{\nu}{\gamma+t} + \left[\frac{(L+l)\beta^2}{20(\gamma+t)^2} + \frac{18^2l^5\beta^2}{20\mu_2^4(\gamma+t)^2} + \frac{L\beta^2}{2(\gamma+t)^2}\right]\sigma^2 \\
&\leq \frac{\gamma+t-1}{(\gamma+t)^2}\nu - \frac{\frac{1}{2}\mu_1\beta-1}{(\gamma+t)^2}\nu + \left[\frac{(L+l)\beta^2}{20(\gamma+t)^2} + \frac{18^2l^5\beta^2}{20\mu_2^4(\gamma+t)^2} + \frac{L\beta^2}{2(\gamma+t)^2}\right]\sigma^2 \qquad (3.31) \\
&\leq \frac{\nu}{\gamma+t+1},
\end{aligned}$$

where in the second inequality we plug in τ_1^t and τ_2^t, and in the last inequality we use $(\gamma+t+1)(\gamma+t-1) \leq (\gamma+t)^2$ and the fact that sum of last two terms in (3.31) is no greater than 0 by our choice of ν. □

3.6.3 *Proofs for Chapter 3.4*

PROOF OF THEOREM 3.4.1

Proof. Because the proof is long, we break the proof into three parts for the convenience of understanding the intuition behind it.

Part 1.

Consider in one outer loop k. Define $a_{t,j} = \mathbb{E}[g(x_{t,j}) - g^*]$, $b_{t,j} = \mathbb{E}[g(x_{t,j}) - f(x_{t,j}, y_{t,j})]$, $\tilde{a}_t = \mathbb{E}[g(\tilde{x}_t) - g^*]$ and $\tilde{b}_t = \mathbb{E}[g(\tilde{x}_t) - f(\tilde{x}_t, \tilde{y}_t)]$. We omit the subscript t for now and denote the stochastic gradients as

$$G_x(x_j, y_j) = \nabla_x f_{i_j}(x_j, y_j) - \nabla_x f_{i_j}(\tilde{x}, \tilde{y}) + \nabla_x f(\tilde{x}, \tilde{y}),$$
$$G_y(x_j, y_{j+1}) = \nabla_y f_{i_j}(x_{j+1}, y_j) - \nabla_y f_{i_j}(\tilde{x}, \tilde{y}) + \nabla_y f(\tilde{x}, \tilde{y}).$$

Note that these are unbiased stochastic gradients. Similar to the proof of Theorem 3.6.5 (replace σ^2 in (3.15)), with $\tau_1 \leq 1/L$, we have

$$a_{j+1} \leq a_j - \frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_j)\|^2 + \frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_j, y_j) - \nabla g(x_j)\|^2 + \frac{L}{2}\tau_1^2\mathbb{E}\|G_x(x_j, y_j) - \nabla_x f(x_j, y_j)\|^2. \tag{3.32}$$

By Lemma 3.6.4, with $\tau_2 \leq 1/l$,

$$b_{j+1} \leq \mathbb{E}[g(x_{j+1}) - f(x_{j+1}, y_j)] - \frac{\tau_2}{2}\mathbb{E}\|\nabla_y f(x_{j+1}, y_j)\|^2 + \frac{l}{2}\tau_2^2\mathbb{E}\|G_y(x_{j+1}, y_j) - \nabla_y f(x_{j+1}, y_j)\|^2. \tag{3.33}$$

Furthermore, we bound the distance to the $\tilde{x} = x_0$ as

$$\begin{aligned}
\mathbb{E}\|x_{j+1} - \tilde{x}\|^2 &= \mathbb{E}\|x_j - \tau_1 G_x(x_j, y_j) - \tilde{x}\|^2 \\
&= \mathbb{E}\|x_j - \tilde{x}\|^2 + 2\mathbb{E}\langle x_j - \tilde{x}, \tau_1 \nabla_x f(x_j, y_j)\rangle + \tau_1^2\mathbb{E}\|\nabla_x f(x_j, y_j)\|^2 + \\
&\quad \tau_1^2\mathbb{E}\|G_x(x_j, y_j) - \nabla_x f(x_j, y_j)\|^2 \\
&\leq (1 + \tau_1\beta_1)\mathbb{E}\|x_j - \tilde{x}\|^2 + \left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right)\mathbb{E}\|\nabla_x f(x_j, y_j)\|^2 + \\
&\quad \tau_1^2\mathbb{E}\|G_x(x_j, y_j) - \nabla_x f(x_j, y_j)\|^2,
\end{aligned} \tag{3.34}$$

where in the last inequality we use Young's inequality to the inner product, and $\beta_1 > 0$ is a constant which we will determine later. Similarly,

$$\begin{aligned}
\mathbb{E}\|y_{j+1} - \tilde{y}\|^2 \leq (1 + \tau_2\beta_2)\mathbb{E}\|y_j - \tilde{y}\|^2 + \left(\tau_2^2 + \frac{\tau_2}{\beta_2}\right)\mathbb{E}\|\nabla_y f(x_{j+1}, y_j)\|^2 + \\
\tau_2^2\mathbb{E}\|G_y(x_{j+1}, y_j) - \nabla_y f(x_{j+1}, y_j)\|^2,
\end{aligned} \tag{3.35}$$

where in the last inequality we use Young's inequality to the inner product and $\beta_2 > 0$ is a constant. We construct a potential function

$$R_j = a_j + \lambda b_j + c_j\|x_j - \tilde{x}\|^2 + d_j\|y_j - \tilde{y}\|^2, \tag{3.36}$$

and we will determine λ, c_j and d_j later. Combining (3.32), (3.33) and (3.35),

$$\begin{aligned}
R_{j+1} \leq\ & a_j - \frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_j)\|^2 + \frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_j, y_j) - \nabla g(x_j)\|^2 + \frac{L}{2}\tau_1^2\mathbb{E}\|G_x(x_j, y_j) - \nabla_x f(x_j, y_j)\|^2 + \\
& \lambda\mathbb{E}[g(x_{j+1}) - f(x_{j+1}, y_j)] - \frac{\lambda\tau_2}{2}\mathbb{E}\|\nabla_y f(x_{j+1}, y_j)\|^2 + \\
& c_{j+1}\mathbb{E}\|x_{j+1} - \tilde{x}\|^2 + \left(d_{j+1} + \frac{\lambda l}{2}\right)\tau_2^2\mathbb{E}\|G_y(x_{j+1}, y_j) - \nabla_y f(x_{j+1}, y_j)\|^2 +
\end{aligned}$$

$$d_{j+1}(1+\tau_2\beta_2)\mathbb{E}\|y_j-\tilde{y}\|^2+d_{j+1}\left(\tau_2^2+\frac{\tau_2}{\beta_2}\right)\mathbb{E}\|\nabla_y f(x_{j+1},y_j)\|^2 \tag{3.37}$$

Then we bound the variance of the stochastic gradients,

$$\begin{aligned}
&\mathbb{E}\|G_y(x_{j+1},y_j)-\nabla_y f(x_{j+1},y_j)\|^2\\
&=\mathbb{E}\|\nabla_y f_{i_j}(x_{j+1},y_j)-\nabla_y f_{i_j}(\tilde{x},\tilde{y})+\nabla_y f(\tilde{x},\tilde{y})-\nabla_y f(x_{j+1},y_j)\|^2\\
&\le \mathbb{E}\|\nabla_y f_{i_j}(x_{j+1},y_j)-\nabla_y f_{i_j}(\tilde{x},\tilde{y})\|^2\le l^2\mathbb{E}\|x_{j+1}-\tilde{x}\|^2+l^2\mathbb{E}\|y_j-\tilde{y}\|^2,
\end{aligned} \tag{3.38}$$

where in the first inequality we apply $\mathbb{E}[\nabla_y f_{i_j}(x_{j+1},y_j)-\nabla_y f_{i_j}(\tilde{x},\tilde{y})]=\nabla_y f(x_{j+1},y_j)-\nabla_y f(\tilde{x},\tilde{y})$. Similarly,

$$\mathbb{E}\|G_x(x_j,y_j)-\nabla_x f(x_j,y_j)\|^2\le l^2\mathbb{E}\|x_j-\tilde{x}\|^2+l^2\mathbb{E}\|y_j-\tilde{y}\|^2. \tag{3.39}$$

Plugging (3.38) into (3.37),

$$\begin{aligned}
R_{j+1}\le\ & a_j-\frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_j)\|^2+\frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_j,y_j)-\nabla g(x_j)\|^2+\frac{L}{2}\tau_1^2\mathbb{E}\|G_x(x_j,y_j)-\nabla_x f(x_j,y_j)\|^2+\\
&\lambda\mathbb{E}[g(x_{j+1})-f(x_{j+1},y_j)]-\frac{\lambda\tau_2}{2}\mathbb{E}\|\nabla_y f(x_{j+1},y_j)\|^2+\\
&\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\mathbb{E}\|x_{j+1}-\tilde{x}\|^2+\\
&\left[d_{j+1}(1+\tau_2\beta_2)+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\mathbb{E}\|y_j-\tilde{y}\|^2+d_{j+1}\left(\tau_2^2+\frac{\tau_2}{\beta_2}\right)\mathbb{E}\|\nabla_y f(x_{j+1},y_j)\|^2.
\end{aligned} \tag{3.40}$$

Then we plug in (3.34) and rearrange,

$$\begin{aligned}
R_{j+1}\le\ & a_j-\frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_j)\|^2+\\
&\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\left(\tau_1^2+\frac{\tau_1}{\beta_1}\right)\mathbb{E}\|\nabla_x f(x_j,y_j)\|^2+\frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_j,y_j)-\nabla g(x_j)\|^2+\\
&\lambda\mathbb{E}[g(x_{j+1})-f(x_{j+1},y_j)]-\left[\frac{\lambda\tau_2}{2}-d_{j+1}\left(\tau_2^2+\frac{\tau_2}{\beta_2}\right)\right]\mathbb{E}\|\nabla_y f(x_{j+1},y_j)\|^2+\\
&\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right](1+\tau_1\beta_1)\mathbb{E}\|x_j-\tilde{x}\|^2+\left[d_{j+1}(1+\tau_2\beta_2)+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\mathbb{E}\|y_j-\tilde{y}\|^2+\\
&\left[\frac{L}{2}+c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\tau_1^2\mathbb{E}\|G_x(x_j,y_j)-\nabla_x f(x_j,y_j)\|^2.
\end{aligned} \tag{3.41}$$

Consider the third line. Using PL condition $\|\nabla_y f(x_{j+1},y_j)\|^2\ge 2\mu_2[g(x_{j+1})-f(x_{j+1},y_j)]$ and assuming $\lambda\ge d_{j+1}(\tau_2+1/\beta_2)$, which we will justify later by our choices of d_{j+1} and β_2, we have

$$\begin{aligned}
&\text{the third line}\\
&\le\lambda\left[1-\tau_2\mu_2+\frac{\lambda}{2}d_{j+1}\left(\tau_2^2+\frac{\tau_2}{\beta_2}\right)\mu_2\right]\mathbb{E}[g(x_{j+1})-f(x_{j+1},y_j)]\\
&\le\lambda\left[1-\tau_2\mu_2+\frac{\lambda}{2}d_{j+1}\left(\tau_2^2+\frac{\tau_2}{\beta_2}\right)\mu_2\right]\left\{b_j+\mathbb{E}\big(f(x_j,y_j)-f(x_{j+1},y_j)\big)+(a_{j+1}-a_j)\right\}
\end{aligned}$$

$$\leq \lambda\left[1-\tau_2\mu_2+\frac{\lambda}{2}d_{j+1}\left(\tau_2^2+\frac{\tau_2}{\beta_2}\right)\mu_2\right]\Big\{b_j+\left(\tau_1+\frac{l}{2}\tau_1^2\right)\mathbb{E}\|\nabla_x f(x_j,y_j)\|^2+$$
$$\frac{l}{2}\tau_1^2\mathbb{E}\|G_x(x_j,y_j)-\nabla_x f(x_j,y_j\|^2-\frac{\tau_1}{2}\mathbb{E}\|\nabla g(x_j)\|^2+$$
$$\frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_j,y_j)-\nabla g(x_j)\|^2+\frac{L}{2}\tau_1^2\mathbb{E}\|G_x(x_j,y_j)-\nabla_x f(x_j,y_j)\|^2\Big\},$$

where in the last inequality we use (3.32) and (3.17). Now we plug this into R_{j+1},

$$R_{j+1}\leq a_j-\frac{\tau_1}{2}(1+\lambda\zeta)\mathbb{E}\|\nabla g(x_j)\|^2+$$
$$\left\{\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\left(\tau_1^2+\frac{\tau_1}{\beta_1}\right)+\lambda\zeta\left(\tau_1+\frac{l}{2}\tau_1^2\right)\right\}\mathbb{E}\|\nabla_x f(x_j,y_j)\|^2+$$
$$\frac{\tau_1}{2}(1+\lambda\zeta)\mathbb{E}\|\nabla_x f(x_j,y_j)-\nabla g(x_j)\|^2+\lambda\zeta b_j+$$
$$\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right](1+\tau_1\beta_1)\mathbb{E}\|x_j-\tilde{x}\|^2+\left[d_{j+1}(1+\tau_2\beta_2)+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\mathbb{E}\|y_j-\tilde{y}\|^2+$$
$$\left[\frac{L}{2}+c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2+\lambda\zeta\frac{L+l}{2}\right]\tau_1^2\mathbb{E}\|G_x(x_j,y_j)-\nabla_x f(x_j,y_j)\|^2, \tag{3.42}$$

where we define $\zeta=1-\tau_2\mu_2+\frac{\lambda}{2}d_{j+1}\left(\tau_2^2+\frac{\tau_2}{\beta_2}\right)\mu_2$ and $\psi=1-\zeta$. With $\|\nabla_x f(x_j,y_j)\|^2\leq 2\|\nabla g(x_j)\|^2+2\|\nabla g(x_j)-\nabla_x f(x_j,y_j)\|^2$, we have

$$R_{j+1}\leq$$
$$a_j-\left\{\frac{\tau_1}{2}(1+\lambda\zeta)-2\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\left(\tau_1^2+\frac{\tau_1}{\beta_1}\right)-2\lambda\zeta\left(\tau_1+\frac{l}{2}\tau_1^2\right)\right\}\mathbb{E}\|\nabla g(x_j)\|^2+$$
$$\lambda\zeta b_j+\left\{\frac{\tau_1}{2}(1+\lambda\zeta)+2\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\left(\tau_1^2+\frac{\tau_1}{\beta_1}\right)-2\lambda\zeta\left(\tau_1+\frac{l}{2}\tau_1^2\right)\right\}\mathbb{E}\|\nabla_x f(x_j,y_j)-\nabla g(x_j)\|^2+$$
$$\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right](1+\tau_1\beta_1)\mathbb{E}\|x_j-\tilde{x}\|^2+\left[d_{j+1}(1+\tau_2\beta_2)+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\mathbb{E}\|y_j-\tilde{y}\|^2+$$
$$\left[\frac{L}{2}+c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2+\lambda\zeta\frac{L+l}{2}\right]\tau_1^2\mathbb{E}\|G_x(x_j,y_j)-\nabla_x f(x_j,y_j)\|^2. \tag{3.43}$$

Then plugging in (3.23), (3.24) and (3.39), we get

$$R_{j+1}\leq$$
$$a_j-\left\{\tau_1(1+\lambda\zeta)-4\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\left(\tau_1^2+\frac{\tau_1}{\beta_1}\right)-4\lambda\zeta\left(\tau_1+\frac{l}{2}\tau_1^2\right)\right\}\mu_1 a_j+$$
$$\lambda b_j-\lambda\frac{1}{\lambda}\left\{\lambda\psi-\frac{l^2\tau_1}{\mu_2}(1+\lambda\zeta)-\frac{4l^2}{\mu_2}\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]\left(\tau_1^2+\frac{\tau_1}{\beta_1}\right)-\frac{4l^2}{\mu_2}\lambda\zeta\left(\tau_1+\frac{l}{2}\tau_1^2\right)\right\}b_j+$$
$$\left\{\left[c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right](1+\tau_1\beta_1)+\left[\frac{L}{2}+c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2+\lambda\zeta\frac{L+l}{2}\right]\tau_1^2l^2\right\}\mathbb{E}\|x_j-\tilde{x}\|^2+$$
$$\left\{\left[d_{j+1}(1+\tau_2\beta_2)+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2\right]+\left[\frac{L}{2}+c_{j+1}+\left(d_{j+1}+\frac{\lambda l}{2}\right)l^2\tau_2^2+\lambda\zeta\frac{L+l}{2}\right]\tau_1^2l^2\right\}\mathbb{E}\|y_j-\tilde{y}\|^2. \tag{3.44}$$

Now we are ready to define sequences $\{c_j\}_j$ and $\{d_j\}_j$. Let $c_N = d_N = 0$, and

$$c_j = \left[c_{j+1} + \left(d_{j+1} + \frac{\lambda l}{2}\right) l^2\tau_2^2\right](1+\tau_1\beta_1) + \left[\frac{L}{2} + c_{j+1} + \left(d_{j+1} + \frac{\lambda l}{2}\right) l^2\tau_2^2 + \lambda\zeta\frac{L+l}{2}\right]\tau_1^2 l^2,$$
$$d_j = \left[d_{j+1}(1+\tau_2\beta_2) + \left(d_{j+1} + \frac{\lambda l}{2}\right) l^2\tau_2^2\right] + \left[\frac{L}{2} + c_{j+1} + \left(d_{j+1} + \frac{\lambda l}{2}\right) l^2\tau_2^2 + \lambda\zeta\frac{L+l}{2}\right]\tau_1^2 l^2.$$

We further define

$$m_j^1 \triangleq \tau_1(1+\lambda\zeta) - 4\left[c_{j+1} + \left(d_{j+1} + \frac{\lambda l}{2}\right) l^2\tau_2^2\right]\left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right) - 4\lambda\zeta\left(\tau_1 + \frac{l}{2}\tau_1^2\right), \tag{3.45}$$
$$m_j^2 \triangleq \frac{1}{\lambda}\Bigg\{\lambda\psi - \frac{l^2\tau_1}{\mu_2}(1+\lambda\zeta) - \frac{4l^2}{\mu_2}\left[c_{j+1} + \left(d_{j+1} + \frac{\lambda l}{2}\right) l^2\tau_2^2\right]\left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right) - \frac{4l^2}{\mu_2}\lambda\zeta\left(\tau_1 + \frac{l}{2}\tau_1^2\right)\Bigg\}. \tag{3.46}$$

Then we can write (3.44) as

$$R_{j+1} \le R_j - m_j^1 a_j - \lambda m_j^2 b_j \tag{3.47}$$

Now we bring back the subscript t. Summing the equation from 0 to $N-1$,

$$\sum_{j=0}^{N-1} a_{t,j} + \lambda b_{t,j} \le \frac{R_0 - R_N}{N\gamma} = \frac{a_{t,0} + \lambda b_{t,0} - a_{t,N} - \lambda b_{t,N}}{N\gamma} = \frac{\tilde{a}_t + \lambda\tilde{b}_t - \tilde{a}_{t+1} - \lambda\tilde{b}_{t+1}}{N\gamma}, \tag{3.48}$$

where $\gamma := \min_j\{m_j^1, m_j^2\}$, and the first equality is due to $c_N = d_N = 0$ and $(x_{t,0}, y_{t,0}) = (\tilde{x}_t, \tilde{y}_t)$. Summing t from o to $T-1$, we get

$$\frac{1}{NT}\sum_{t=0}^{T-1}\sum_{j=0}^{N-1} a_{t,j} + \lambda b_{t,j} \le \frac{\tilde{a}_0 + \lambda\tilde{b}_0}{NT\gamma} = \frac{a^k + \lambda b^k}{NT\gamma}. \tag{3.49}$$

The left hand side is exactly $a^{k+1} + \lambda b^{k+1}$, because (x_k, y_k) is sampled uniformly from $\{\{(x_{t,j}, y_{t,j})\}_{j=0}^{N-1}\}_{t=0}^{T-1}$.

Part 2.

It suffices to choose proper τ_1, τ_2, N and T such that $NT\gamma > 1$. Driven by the proof, we choose

$$\tau_1 = \frac{k_1}{\kappa^2 l}, \quad \beta_1 = k_2\kappa^2 l, \quad \tau_2 = \frac{k_3}{l}, \quad \beta_2 = lk_4.$$

We will choose k_1, k_2, k_3 and k_4 later and we let $k_1, k_2, k_3, k_4 \le 1$. Plug back to c_j and d_j, we have

$$\begin{aligned} c_j &= \left(1 + k_1k_2 + \frac{k_1^2}{\kappa^4}\right)c_{j+1} + \left[k_3^2(1+k_1k_2) + \frac{k_1^2k_3^2}{\kappa^4} + (L+l)\frac{k_1^2}{\kappa^4}\left(\frac{k_3^2}{l^2} + \frac{k_3}{l^2k_4}\right)\mu_2\right]d_{j+1} + \\ &\quad \frac{\lambda}{2}lk_3^2(1+k_1k_2) + \frac{L}{2\kappa^4}k_1^2 + \frac{\lambda}{2\kappa^4}lk_1^2k_3^2 + \frac{\lambda}{2\kappa^4}(L+l)k_1^2(1-k_3k_4) \\ &\le \left(1 + k_1k_2 + \frac{k_1^2}{\kappa^4}\right)c_{j+1} + \left(3k_3^2 + 3\frac{1}{\kappa^3}k_1^2\right)d_{j+1} + 2\lambda lk_3^2 + (1+2\lambda)\frac{l}{\kappa^3}k_1^2, \end{aligned} \tag{3.50}$$

where in the last inequality we assume $k_3^2 + \frac{k_3}{k_4} \leq 1$.

$$\begin{aligned} d_j &= \frac{k_1^2}{\kappa^4} c_{j+1} + \left[1 + k_3 k_4 + k_3^2 + (L+l)\frac{k_1^2}{\kappa^4}\left(\frac{k_3^2}{l^2} + \frac{k_3}{l^2 k_4}\right)\mu_2 + \frac{1}{\kappa^4} k_1^2 k_3^2\right] d_{j+1} + \\ &\quad \frac{\lambda}{2} l k_3^2 + \frac{L}{2\kappa^4} k_1^2 + \frac{\lambda}{2\kappa^4} l k_1^2 k_3^2 + \frac{\lambda}{2\kappa^4}(L+l) k_1^2 (1 - k_3 k_4) \\ &\leq \frac{k_1^2}{\kappa^4} c_{j+1} + \left(1 + k_3 k_4 + 2k_3^2 + \frac{3}{\kappa^3} k_1^2\right) d_{j+1} + \lambda l k_3^2 + (1+2\lambda)\frac{l}{\kappa^3} k_1^2. \end{aligned} \tag{3.51}$$

We define $e_j = \max\{c_j, d_j\}$. Then combining (3.50) and (3.51), we easily get

$$e_j \leq \left(1 + k_1 k_2 + k_3 k_4 + 3k_3^2 + \frac{4}{\kappa^3} k_1^2\right) e_{j+1} + 2\lambda l k_3^2 + (1+2\lambda)\frac{l}{\kappa^3} k_1^2.$$

As $e_N = 0$, we have

$$e_0 \leq \left[2\lambda l k_3^2 + (1+2\lambda)\frac{l}{\kappa^3} k_1^2\right] \frac{\left(1 + k_1 k_2 + k_3 k_4 + 3k_3^2 + \frac{4}{\kappa^3} k_1^2\right)^N - 1}{k_1 k_2 + k_3 k_4 + 3k_3^2 + \frac{4}{\kappa^3} k_1^2}, \tag{3.52}$$

and note that $e_j > e_{j+1}$ so $e_j \leq e_0, \forall j$. Then we want to lower bound γ. Rearrange (3.45),

$$\begin{aligned} m_j^1 &= \mu_1 \Bigg\{ \tau_1(1 + \lambda - \lambda\tau_2\mu_2) - 2\lambda l^3 \tau_2^2 \left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right) - 4\lambda \left(\tau_1 + \frac{l}{2}\tau_1^2\right)(1 - \tau_2\mu_2) - \\ &\quad \left[-2\tau_1\left(\tau_2^2 + \frac{\tau_2}{\beta_2}\right)\mu_2 + 4\left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right) l^2 \tau_2^2 + 8\left(\tau_1 + \frac{l}{2}\tau_1^2\right)\left(\tau_2^2 + \frac{\tau_2}{\beta_2}\right)\mu_2\right] d_{j+1} - \\ &\quad 4\left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right) c_{j+1} \Bigg\} \\ &\geq \frac{1}{2}\tau_1\mu_1 - \left[\frac{4}{\kappa^4} k_3^2 \left(k_1^2 + \frac{k_1}{k_2}\right) + \frac{10\mu_2}{\kappa^2 l} k_1 \left(k_3^2 + \frac{k_3}{k_4}\right)\right] \frac{\mu_1}{l^2} d_{j+1} - \frac{4}{\kappa^4}\left(k_1^2 + \frac{k_1}{k_2}\right)\frac{\mu_1}{l^2} c_{j+1}, \end{aligned} \tag{3.53}$$

where in the inequality, we use $\lambda = 1/20$ and assume that $\frac{1}{\kappa^2} k_3^2 (k_1 + \frac{1}{k_2}) \leq 10$. Rearranging (3.46),

$$\begin{aligned} m_j^2 &= \tau_2\mu_2 - \frac{l^2\tau_1}{\mu_2}\left(\frac{1}{\lambda} + 1 - \tau_2\mu_2\right) - \frac{2l^5}{\mu_2}\left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right)\tau_2^2 - \frac{4l^2}{\mu_2}\left(\tau_1 + \frac{l}{2}\tau_1^2\right)(1 - \tau_2\mu_2) - \\ &\quad \left[\frac{2}{\lambda}\left(\tau_2^2 + \frac{\tau_2}{\beta_2}\right)\mu_2 + \frac{2}{\lambda} l^2 \tau_1 \left(\tau_2^2 + \frac{\tau_2}{\beta_2}\right) + \frac{4}{\lambda}\frac{l^4}{\mu_2}\tau_2^2\left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right) + \frac{8l^2}{\lambda\mu_2}\left(\tau_1 + \frac{l}{2}\tau_1^2\right)\left(\tau_2^2 + \frac{\tau_2}{\beta_2}\right)\mu_2\right] d_{j+1} - \\ &\quad \frac{4}{\lambda}\frac{l^2}{\mu_2}\left(\tau_1^2 + \frac{\tau_1}{\beta_1}\right) c_{j+1} \\ &\geq \frac{l^2\tau_1}{2\min\{\mu_1,\mu_2\}} - \left[200\left(k_3^2 + \frac{k_3}{k_4}\right) + \frac{80}{\kappa^2}\left(k_1^2 + \frac{k_1}{k_2}\right)\right]\frac{\mu_2}{l^2} d_{j+1} - \frac{80}{\kappa^2}\left(k_1^2 + \frac{k_1}{k_2}\right)\frac{\mu_2}{l^2} c_{j+1}, \end{aligned} \tag{3.54}$$

where in the inequality we use $\lambda = 1/20$ and assume $k_1 \leq k_3/28$ and $\frac{1}{\kappa^2} k_3^2 \left(k_1 + \frac{1}{k_2}\right) \leq 1/4$. Note that $\frac{1}{2}\tau_1\mu_1 = \frac{\mu_1}{2\kappa^2 l} k_1$ and $\frac{l^2\tau_1}{2\min\{\mu_1,\mu_2\}} = \frac{l}{2\kappa^2\min\{\mu_1,\mu_2\}} k_1$. Then we have

$$m_j^1 \geq \frac{1}{\kappa^3}\left\{\frac{1}{2}k_1 - \left[\frac{4}{\kappa^2}k_3^2\left(k_1^2+\frac{k_1}{k_2}\right)+\frac{10\mu_2}{l}k_1\left(k_3^2+\frac{k_3}{k_4}\right)\right]\frac{d_{j+1}}{l} - \frac{4}{\kappa^2}\left(k_1^2+\frac{k_1}{k_2}\right)\frac{c_{j+1}}{l}\right\}, \quad (3.55)$$

$$m_j^2 \geq \frac{1}{\kappa}\left\{\frac{1}{2}k_1 - \left[\frac{80}{\kappa^2}\left(k_1^2+\frac{k_1}{k_2}\right)+200\left(k_3^2+\frac{k_3}{k_4}\right)\right]\frac{d_{j+1}}{l} - \frac{80}{\kappa^2}\left(k_1^2+\frac{k_1}{k_2}\right)\frac{c_{j+1}}{l}\right\}. \quad (3.56)$$

Letting $k_1/k_2 = k_3/k_4$ and $k_1 = \frac{1}{28}k_3$, we have

$$\gamma \geq \frac{1}{\kappa^3}\left\{\frac{1}{56}k_3 - 360\left(k_3^2+\frac{k_3}{k_4}\right)\frac{e_0}{l}\right\}, \quad (3.57)$$

where we use $c_j, d_j \leq e_0, \forall j$. By plugging in $k_1 = k_3/28$ and $\lambda = 1/20$ into (3.52), we have

$$e_0 \leq l\frac{(1+2k_3k_4+4k_3^2)^N - 1}{k_4/k_3+3}. \quad (3.58)$$

Plugging this into (3.57), we have

$$\gamma \geq \frac{1}{\kappa^3}\left[\frac{k_3}{56} - 360\frac{(1+2k_3k_4+4k_3^2)^N-1}{k_4/k_3+3}\left(k_3^2+k_3/k_4\right)\right]. \quad (3.59)$$

We choose $k_4 = k_3^{1/2}$, then

$$NT\gamma \geq \frac{1}{\kappa^3}\left[\frac{k_3}{56} - 360\left((1+2k_3^{3/2}+4k_3^2)^N-1\right)\left(\frac{k_3^2+k_3^{1/2}}{k_3^{-1/2}+3}\right)\right]NT. \quad (3.60)$$

Part 3

We choose $T = 1$, $k_3 = \beta\kappa^{-6}$ and $N = \alpha(2k_3^{3/2}+4k_3^2)^{-1} \geq \frac{\alpha}{2}k_2^{-3/2}$, where α, β is irrelevant to n, l, μ_1, μ_2. Then since $(1+2k_3^{3/2}+4k_3^2)^N \leq e^\alpha$, after plugging in N and k_3, we have

$$NT\gamma \geq \frac{1}{\kappa^3}\left[\frac{k_3}{56} - 360(e^\alpha-1)(2k_3)\right]\frac{\alpha}{2}k_2^{-3/2} \geq \frac{1}{2}\left[\frac{1}{56} - 2\times 360(e^\alpha-1)\right]\alpha\beta^{-1/2}. \quad (3.61)$$

Therefore, for choosing α small enough and β small enough, we have $NT\gamma \geq 2$. Now it remains to verify several assumptions we made in the proof. The first is $\frac{k_3}{k_4}+k_3^2 \leq 1$. Since $\frac{k_3}{k_4}+k_3^2 = k_3^{1/2}+k_3^2$, this assumption easily holds when $\beta \leq 1/4$. The second assumption we want to verify is $\frac{1}{\kappa^2}k_3^2\left(k_1+\frac{1}{k_2}\right) \leq 1/4$. Note that

$$\frac{1}{\kappa^2}k_3^2\left(k_1+\frac{1}{k_2}\right) = \frac{1}{\kappa^2}k_3^2\left(k_1+\frac{k_3}{k_4k_1}\right) = \frac{1}{\kappa^2}k_3^2\left(\frac{1}{28}k_3+28k_3^{-1/2}\right).$$

So this assumption can also be easily satisfied when β is small. The last assumption we need to verify is $\lambda \geq d_{j+1}\left(\tau_2+\frac{1}{\beta_2}\right)$. Because $d_{j+1} \leq e_0$ and (3.58),

$$\begin{aligned} d_{j+1}\left(\tau_2+\frac{1}{\beta_2}\right) &\leq l\frac{(1+2k_3k_4+4k_3^2)^N-1}{k_4/k_3+3}\left(\frac{k_3}{l}+\frac{1}{k_4l}\right) \\ &\leq \left((1+2k_3k_4+4k_3^2)^N-1\right)\left(\frac{k_3^2+k_3^{1/2}}{k_3^{-1/2}+3}\right) \\ &\leq 2(e^\alpha-1)k_3. \end{aligned}$$

So this assumption holds when α and β are small.

□

PROOF OF THEOREM 3.4.2

Proof. We start from Part 3 of the proof of Theorem 3.4.1. We now choose $k_3 = \beta n^{-2/3}$, $N = \alpha(2k_3^{3/2} + 4k_3^2)^{-1}$, and $T = \kappa^3 n^{-1/3}$ then

$$NT\gamma \geq \frac{1}{2}\left[\frac{1}{56} - 2 \times 360(e^{\alpha} - 1)\right]\alpha\beta^{-1/2} \tag{3.62}$$

Therefore, for choosing α small enough and β small enough, we have $NT\gamma \geq 2$. Note that when $\kappa^3 n^{-1/3} \leq 1$, we choose $T = 1$ and the complexity is therefore $\tilde{\mathcal{O}}(n)$. Other assumptions can be easily verified by the same way as in the proof of Theorem 3.4.1. □

4

SINGLE-LOOP ALGORITHMS FOR NONCONVEX-PL MINIMAX PROBLEMS

This chapter establishes new convergence results for two alternative single-loop algorithms – *alternating GDA* and *smoothed GDA* – under the mild assumption that the objective satisfies the Polyak-Łojasiewicz (PL) condition about one variable. We prove that, to find an ϵ-stationary point, (i) alternating GDA and its stochastic variant (without mini batch) respectively require $O(\kappa^2\epsilon^{-2})$ and $O(\kappa^4\epsilon^{-4})$ iterations, while (ii) smoothed GDA and its stochastic variant (without mini batch) respectively require $O(\kappa\epsilon^{-2})$ and $O(\kappa^2\epsilon^{-4})$ iterations. The latter greatly improves over the vanilla GDA and gives the hitherto best known complexity results among single-loop algorithms under similar settings.

4.1 OVERVIEW

In this chapter, we consider finding stationary points for the smooth minimax optimization problems:

$$\min_{x\in\mathbb{R}^{d_1}} \max_{y\in\mathbb{R}^{d_2}} f(x,y) \triangleq \mathbb{E}[F(x,y;\xi)], \tag{4.1}$$

where ξ is a random vector and $f(x,y)$ is nonconvex in x and possibly nonconcave in y.

Due to its simplicity and single-loop nature, gradient descent ascent (GDA) and its stochastic variants, have become the *de facto* algorithms for training GANs and many other applications in practice. Their theoretical properties have also been extensively studied in recent literature [Lei et al., 2020, Nagarajan and Kolter, 2017, Heusel et al., 2017, Mescheder et al., 2017, 2018].

Lin et al. [2020a] derived a complexity analysis for simultaneous GDA (with simultaneous updates for x and y) and for stochastic GDA (hereafter Stoc-GDA) for finding stationary points when the objective is concave in y. In particular, they show that GDA requires $O(\epsilon^{-6})$ iterations and Stoc-GDA without mini-batch requires $O(\epsilon^{-8})$ samples to achieve an ϵ-approximate stationary point. When the objective is strongly concave in y, the iteration complexity of GDA can be significantly improved to $O(\kappa^2\epsilon^{-2})$ while the sample complexity for Stoc-GDA reduces to $O(\kappa^3\epsilon^{-4})$ with a large batch of size $O(\epsilon^{-2})$ or $O(\kappa^3\epsilon^{-5})$ without the batch, i.e., using a single sample to construct the gradient estimator. Here κ is the underlying condition number defined as l/μ with l being Lipschitz smooth-

Algorithms	**Complexity** $\|\nabla\Phi(x)\| \le \epsilon$	**Complexity** $\|\nabla f(x,y)\| \le \epsilon$	**Loops**	**Additional assumptions**
GDA [Lin et al., 2020a]	$O(\kappa^2\Delta l\epsilon^{-2})$	$O(\kappa^2\Delta l\epsilon^{-2})^\star$	1	strong concavity in y
Catalyst-EG [Zhang et al., 2021b]	$O(\sqrt{\kappa}\Delta l\epsilon^{-2})$	$O(\sqrt{\kappa}\Delta l\epsilon^{-2})^\star$	3	strong concavity in y
Multi-GDA [Nouiehed et al., 2019]	$\tilde{O}(\kappa^3\Delta l\epsilon^{-2})^\star$	$\tilde{O}(\kappa^2\Delta l\epsilon^{-2})$	2	
Catalyst-AGDA [Appendix 4.6.4]	$O(\kappa\Delta l\epsilon^{-2})$	$O(\kappa\Delta l\epsilon^{-2})$	2	
AGDA	$O(\kappa^2\Delta l\epsilon^{-2})^\diamond$	$O(\kappa^2\Delta l\epsilon^{-2})$	1	
Smoothed-AGDA	$O(\kappa\Delta l\epsilon^{-2})$	$O(\kappa\Delta l\epsilon^{-2})$	1	

TABLE 4.1: Oracle complexities for deterministic NC-PL problems. Here $\tilde{O}(\cdot)$ hides poly-logarithmic factors. l: Lipschitz smoothness parameter; μ: PL parameter, κ: condition number $\frac{l}{\mu}$; Δ: initial gap of the primal function. We measure the stationarity by $\|\nabla\Phi(x)\|$ with $\Phi(x) = \max_y f(x,y)$ and $\|\nabla f(x,y)\|$. Here * means the complexity is derived by translating from one stationary measure to the other (see Proposition 2). $\diamond$ it recovers the same complexity for AGDA as Appendix D in [Yang et al., 2020a]

ness parameter and μ strong concavity parameter. However, the following question is still unsettled:

> *Can stochastic GDA-type algorithms achieve the better sample complexity of $O(\epsilon^{-4})$ without a large batch size?*

Besides the dependence on ϵ, the condition number also plays a crucial role in the convergence rate. There is a long line of research aiming to reduce such a dependency, see e.g. [Lin et al., 2020b, Zhang et al., 2021b] for some recent results for minimax optimization. These algorithms are typically more complicated as they rely on multiple loops, and are equipped with several acceleration mechanisms. Single-loop algorithms are far more favorable in practice because of their simplicity in implementation. Recently, several single-loop variants of GDA have been proposed, including Alternating Gradient Projection (AGP) [Xu et al., 2020c] and Smoothed-AGDA [Zhang et al., 2020a]. Unfortunately, most of them fail to provide faster convergence in terms of the condition number and they only consider the deterministic setting. The following question is therefore still unanswered:

> *Is it possible to improve the dependence on the condition number without resorting to multi-loop procedures?*

In short, there is an urgent need to obtain *faster convergence in terms of both the target accuracy ϵ and the condition number κ with single-loop algorithms*. This is even more challenging

Algorithms	**Complexity** $\|\nabla\Phi(x)\| \leq \epsilon$	**Complexity** $\|\nabla f(x,y)\| \leq \epsilon$	**Batch size**	**Additional assumptions**
Stoc-GDA[Lin et al., 2020a]	$O(\kappa^3\Delta l\epsilon^{-4})$	$O(\kappa^3\Delta l\epsilon^{-4})^\star$	$O(\epsilon^{-2})$	strong concavity in y
Stoc-GDA[Lin et al., 2020a]	$O(\kappa^3\Delta l\epsilon^{-5})$	$O(\kappa^3\Delta l\epsilon^{-5})^\star$	$O(1)$	strong concavity in y
PDSM[Guo et al., 2021b]	$O(\kappa^3\Delta l\epsilon^{-4})$	$O(\kappa^3\Delta l\epsilon^{-4})^\star$	$O(1)$	strong concavity in y
ALSET[Chen et al., 2021b]	$O(\kappa^3\Delta l\epsilon^{-4})$	$O(\kappa^3\Delta l\epsilon^{-4})^\star$	$O(1)$	strong concavity in y, Lipschitz$^\triangledown$
Stoc-AGDA	$O(\kappa^4\Delta l\epsilon^{-4})$	$O(\kappa^4\Delta l\epsilon^{-4})$	$O(1)$	
Stoc-Smoothed-AGDA	$O(\kappa^2\Delta l\epsilon^{-4})$	$O(\kappa^2\Delta l\epsilon^{-4})$	$O(1)$	

TABLE 4.2: Sample complexities for stochastic NC-PL problems when the target accuracy ϵ is small, i.e. $\epsilon \leq \tilde{O}(\sqrt{\Delta l/\kappa^3})$. We measure the stationarity by $\|\nabla\Phi(x)\|$ with $\Phi(x) = \max_y f(x,y)$ and $\|\nabla f(x,y)\|$. Here * means the complexity is derived by translating from one stationary measure to the other (see Proposition 2). $^\triangledown$ It assumes the function f is Lipschitz continuous about x and its Hessian is Lipschitz continuous.

when the objective is not strongly-concave about y. In this chapter, we investigate two viable single-loop algorithms to address this question: (i) *alternating GDA* (hereafter AGDA and Stoc-AGDA for their stochastic variance) and (ii) *Smoothed-AGDA*. Importantly, AGDA, with sequential updates between x and y, is one of the most popular algorithms used in practice and has an edge over GDA in several settings [Zhang et al., 2021a]. Smoothed-AGDA, first introduced by [Zhang et al., 2020a], utilizes a regularization term to stabilize the performance of GDA when the objective is convex in y. We show that these two algorithms can satisfy our need to achieve faster convergence under milder assumptions.

We are interested in analyzing their theoretical behaviors under the general *NC-PL setting*, namely, the objective is nonconvex in x and satisfies the Polyak-Łojasiewicz (PL) condition in y [Polyak, 1963]. This is a milder assumption than strong concavity and does not even require the objective to be concave in y. Such an assumption has been shown to hold in linear quadratic regulators [Fazel et al., 2018], as well as overparametrized neural networks [Liu et al., 2020a]. This setting has driven a lot of the recent progress in the quest for understanding deep neural networks [Lee et al., 2017, Jacot et al., 2018], and it therefore appears as an ideal candidate to deepen our understanding of the convergence properties of minimax optimization.

4.1.1 *Contributions*

In this chapter, we study the convergence of AGDA and Smoothed-AGDA in the NC-PL setting. Our goal is to find an approximate stationary point for the objective function $f(\cdot,\cdot)$ and its primal function $\Phi(\cdot) \triangleq \max_y f(\cdot, y)$. For each algorithm, we present a *unified* analysis for the deterministic setting, when we have access to exact gradients of (4.1), and the stochastic setting, when we have access to noisy gradients. We denote the smoothness parameter by l, PL parameter by μ, condition number by $\kappa \triangleq \frac{l}{\mu}$ and initial primal function gap $\Phi(x) - \inf_x \Phi(x)$ by Δ.

DETERMINISTIC SETTING. We first show that the output from AGDA is an ϵ-stationary point for both the objective function f and primal function Φ after $O(\kappa^2 \Delta l \epsilon^{-2})$ iterations, which recovers the result of primal function stationary convergence in [Yang et al., 2020a] based on a different analysis. The complexity is optimal in ϵ, since $\Omega(\epsilon^{-2})$ is the lower bound for smooth optimization problems [Carmon et al., 2020]. We further show that Smoothed-AGDA has $O(\kappa \Delta l \epsilon^{-2})$ complexity in finding an ϵ-stationary point of f. We can translate this point to an ϵ-stationary point of Φ after an additional negligible $\tilde{O}(\kappa)$ oracle complexity. This result improves the complexities of existing single-loop algorithms that require the more restrictive assumption of strong-concavity in y (we refer to this class of function as NC-SC). A comparison of our results to existing complexity bounds is summarized in Table 6.1.

STOCHASTIC SETTING. We show that Stoc-AGDA achieves a sample complexity of $O(\kappa^4 \Delta l \epsilon^{-4})$ for both notions of stationary measures, without having to rely on the $O(\epsilon^{-2})$ batch size and Hessian Lipschitz assumption used in prior work. This is the first convergence result for stochastic NC-PL minimax optimization and is also optimal in terms of the dependency to ϵ. We further show that the stochastic Smoothed-AGDA (Stoc-Smoothed-AGDA) algorithm achieves the $O(\kappa^2 \Delta l \epsilon^{-4})$ sample complexity in finding an ϵ stationary point of f or Φ for small ϵ. This result improves upon the state-of-the-art complexity $O(\kappa^3 \Delta l \epsilon^{-4})$ for NC-SC problems, which is a subclass of the NC-PL family. We refer the reader to Table 4.2 for a comparison.

4.1.2 *Related Work*

PL CONDITIONS IN MINIMAX OPTIMIZATION. In the deterministic NC-PL setting, Yang et al. [2020a] and Nouiehed et al. [2019] show that AGDA and its multi-step variant, which applies multiple updates in y after one update of x, can find an approximate

stationary point within $O(\kappa^2\epsilon^{-2})$ and $\tilde{O}(\kappa^2\epsilon^{-2})$ iterations, respectively. Recently, Fiez et al. [2021] showed that GDA converges asymptotically to a differential Stackelberg equilibrium and establish a local convergence rate of $O(\epsilon^{-2})$ for deterministic problems. In comparison, our work establishes non-asymptotic convergence to an ϵ-stationary point regardless of the starting point in both deterministic and stochastic settings, and we also focus on reducing the dependence to the condition number. Xie et al. [2021] consider NC-PL problems in the federated learning setting, showing $O(\epsilon^{-3})$ communication complexity when each client's objective is Lipschitz smooth.

NC-SC MINIMAX OPTIMIZATION. NC-SC problems are a subclass of NC-PL family. In the deterministic setting, GDA-type algorithms has been shown to have $O(\kappa^2\epsilon^{-2})$ iteration complexity [Lin et al., 2020a, Xu et al., 2020c, Boţ and Böhm, 2020, Lu et al., 2020]. Later, Lin et al. [2020b] and Zhang et al. [2021b] improve this to $\tilde{O}(\sqrt{\kappa}\epsilon^{-2})$ by utilizing a proximal point method and Nesterov acceleration, and Zhang et al. [2021b] and Han et al. [2021] develop a tight lower complexity bound of $\Omega(\sqrt{\kappa}\epsilon^{-2})$. Yan et al. [2020] introduce Epoch-GDA for weakly-convex-strongly-concave problems. Comparatively, there are less studies in the stochastic setting. Recently, Chen et al. [2021b] extend their analysis from bilevel optimization to minimax optimization and show $O(\kappa^3\epsilon^{-4})$ sample complexity for an algorithm called ALSET without the $O(\epsilon^{-2})$ batch size required in [Lin et al., 2020a]. ALSET reduces to AGDA in minimax optimization when it only does one step of y update in the inner loop. Guo et al. [2021b] utilize stochastic moving-average estimator to nonconvex optimization and their algorithm PDSM achieves the same complexity for NC-SC minimax problems. We also refer the reader to the increasing body of bilevel optimization literature; e.g. [Guo and Yang, 2021, Ji et al., 2020, Hong et al., 2020, Chen et al., 2021a, Zhang, 2021b]. Also, Luo et al. [2020], Huang and Huang [2021] and Tran-Dinh et al. [2020] explore variance-reduced algorithms in this setting under the averaged smoothness assumption. Concurrently, Fiez et al. [2021] prove perturbed GDA converges to ϵ–local minimax equilibria with complexities of $\tilde{O}(\epsilon^{-4})$ and $\tilde{O}(\epsilon^{-2})$ in stochastic and deterministic problems, respectively, under additional second-order conditions. Notably, Li et al. [2021] develop the lower complexity bound of $\Omega\left(\sqrt{\kappa}\epsilon^{-2} + \kappa^{1/3}\epsilon^{-4}\right)$ for the stochastic setting. Other than first-order algorithms, there are a few explorations of zero-order methods [Xu et al., 2021, Huang et al., 2020, Xu et al., 2020b, Wang et al., 2020b, Liu et al., 2020e, Anagnostidis et al., 2021] and second-order methods [Luo and Chen, 2021, Chen and Zhou, 2021]. All the results above hold in the NC-SC regime, while the PL condition is significantly weaker than strong-concavity as it lies in the nonconvex regime.

4.2 PRELIMINARIES

NOTATIONS. Throughout the chapter, we let $\|\cdot\| = \sqrt{\langle\cdot,\cdot\rangle}$ denote the ℓ_2 (Euclidean) norm and $\langle\cdot,\cdot\rangle$ denote the inner product. For non-negative functions $f(x)$ and $g(x)$, we write $f = O(g)$ if $f(x) \leq cg(x)$ for some $c > 0$, and $f = \tilde{O}(g)$ to omit poly-logarithmic terms. We define the primal-dual gap of a function $f(\cdot,\cdot)$ at a point $(\hat{x},\hat{y})$ as $\text{gap}_f(\hat{x},\hat{y}) \triangleq \max_{y\in\mathbb{R}^{d_2}} f(\hat{x},y) - \min_{x\in\mathbb{R}^{d_1}} f(x,\hat{y})$.

We are interested in minimax problems of the form:

$$\min_{x\in\mathbb{R}^{d_1}} \max_{y\in\mathbb{R}^{d_2}} f(x,y) \triangleq \mathbb{E}[F(x,y;\xi)], \tag{4.2}$$

where ξ is a random vector with support Ξ, and f is possibly nonconvex-nonconcave. We now present the main setting considered in this paper.

Assumption 10 (Lipschitz Smooth)**.** *The function f is differentiable and there exists a positive constant l such that*

$$\|\nabla_x f(x_1,y_1) - \nabla_x f(x_2,y_2)\| \leq l[\|x_1 - x_2\| + \|y_1 - y_2\|],$$
$$\|\nabla_y f(x_1,y_1) - \nabla_y f(x_2,y_2)\| \leq l[\|x_1 - x_2\| + \|y_1 - y_2\|],$$

holds for all $x_1, x_2 \in \mathbb{R}^{d_1}, y_1, y_2 \in \mathbb{R}^{d_2}$.

Assumption 11 (PL Condition in y)**.** *For any fixed x, $\max_{y\in\mathbb{R}^{d_2}} f(x,y)$ has a nonempty solution set and a finite optimal value. There exists $\mu > 0$ such that: $\|\nabla_y f(x,y)\|^2 \geq 2\mu[\max_y f(x,y) - f(x,y)], \forall x, y$.*

The PL condition was originally introduced in [Polyak, 1963] who showed that it guarantees global convergence of gradient descent at a linear rate. This condition is shown in [Karimi et al., 2016] to be weaker than strong convexity as well as other conditions under which gradient descent converges linearly. The PL condition has also drawn much attention recently as it was shown to hold for various non-convex applications of interest in machine learning [Fazel et al., 2018, Cai et al., 2019], including problems related to deep neural networks [Du et al., 2019, Liu et al., 2020a]. In this work, we assume that the objective function f in (4.2) is Lipschitz smooth and satisfies the PL condition about the dual variable y, i.e. Assumption 10 and 11, which is the same setting as in [Nouiehed et al., 2019] and [Yang et al., 2020b] (Appendix D). However, to the best of our knowledge, stochastic algorithms have not yet been studied under such a setting.

From now on, we will define $\Phi(x) \triangleq \max_y f(x,y)$ as the primal function and $\kappa \triangleq \frac{l}{\mu}$ as the condition number. We will assume that $\Phi(\cdot)$ is lower bounded by a finite Φ^*. According to [Nouiehed et al., 2019], $\Phi(\cdot)$ is $2\kappa l$-lipschitz smooth with Assumption 10 and 11. There

are two popular and natural notions of stationarity for minimax optimization in the form of (4.2): one is measured with ∇f and the other is measured with $\nabla\Phi$. We give the formal definitions below.

Definition 16 (Stationarity Measures)**.**

a) $(\hat{x},\hat{y})$ *is an* (ϵ_1,ϵ_2)*-stationary point of a differentiable function* $f(\cdot,\cdot)$ *if* $\|\nabla_x f(\hat{x},\hat{y})\| \le \epsilon_1$ *and* $\|\nabla_y f(\hat{x},\hat{y})\| \le \epsilon_2$*. If* $(\hat{x},\hat{y})$ *is an* (ϵ,ϵ)*-stationary point, we call it* ϵ*-stationary point for simplicity.*

b) $\hat{x}$ *is an* ϵ*-stationary point of a differentiable function* $\Phi(\cdot)$ *if* $\|\nabla\Phi(\hat{x})\| \le \epsilon$*.*

These two notions can be translated to each other by the following proposition.

Proposition 2 (Translation between Stationarity Measures)**.**

a) Under Assumptions 10 and 11, if $\hat{x}$ *is an* ϵ*-stationary point of* Φ *and* $\|\nabla_y f(\hat{x},\tilde{y})\| \le \epsilon'$*, then we can find another* $\hat{y}$ *by maximizing* $f(\hat{x},\cdot)$ *from the initial point* $\tilde{y}$ *with (stochastic) gradient ascent such that* $(\hat{x},\hat{y})$ *is an* $O(\epsilon)$*-stationary point of* f*, which requires* $O\left(\kappa\log\left(\frac{\kappa\epsilon'}{\epsilon}\right)\right)$ *gradients or* $\tilde{O}\left(\kappa+\kappa^3\sigma^2\epsilon^{-2}\right)$ *stochastic gradients.*

b) Under Assumptions 10 and 11, if $(\tilde{x},\tilde{y})$ *is an* $(\epsilon,\epsilon/\sqrt{\kappa})$*-stationary point of* f*, then we can find an* $O(\epsilon)$*-stationary point of* Φ *by approximately solving* $\min_x\max_y f(x,y)+l\|x-\tilde{x}\|^2$ *from the initial point* $(\tilde{x},\tilde{y})$ *with (stochastic) AGDA, which requires* $O(\kappa\log(\kappa))$ *gradients or* $\tilde{O}\left(\kappa+\kappa^5\sigma^2\epsilon^{-2}\right)$ *stochastic gradients.*

Remark 4.2.1. *The proposition implies that we can convert an* ϵ*-stationary point of* Φ *to an* ϵ*-stationary point of* f *and an* $(\epsilon,\epsilon/\sqrt{\kappa})$*-stationary point of* f *to an* ϵ*-stationary point of* Φ*, at a low cost in* $1/\epsilon$ *dependency compared to the complexity of finding the stationary point of either notion. Therefore, we consider the stationarity of* Φ *a slightly stronger notion than the other. Lin et al. [2020a] establish the similar conversion under the NC-SC setting, but it requires an* (ϵ/κ)*-stationary point of* f *to find an* ϵ*-stationary point of* Φ*. Later we will use this proposition to establish the stationary convergence for some algorithms.*

Finally, we assume to have access to unbiased stochastic gradients of f with bounded variance.

Assumption 12 (Stochastic Gradients)**.** $G_x(x,y,\xi)$ *and* $G_y(x,y,\xi)$ *are unbiased stochastic estimators of* $\nabla_x f(x,y)$ *and* $\nabla_y f(x,y)$ *and have variances bounded by* $\sigma^2>0$*.*

Algorithm 6 Stoc-AGDA

1: Input: (x_0, y_0), step sizes $\tau_1 > 0, \tau_2 > 0$
2: **for all** $t = 0, 1, 2, ..., T-1$ **do**
3: Draw two i.i.d. samples ξ_1^t, ξ_2^t
4: $x_{t+1} \leftarrow x_t - \tau_1 G_x(x_t, y_t, \xi_1^t)$
5: $y_{t+1} \leftarrow y_t + \tau_2 G_y(x_{t+1}, y_t, \xi_2^t)$
6: **end for**
7: Output: choose $(\hat{x}, \hat{y})$ uniformly from $\{(x_t, y_t)\}_{t=0}^{T-1}$

Stochastic alternating gradient descent ascent (Stoc-AGDA) presented in Algorithm 6 sequentially updates primal and dual variables with simple stochastic gradient descent/ascent. In each iteration, only two samples are drawn to evaluate stochastic gradients. Here τ_1 and τ_2 denote the stepsize of x and y, respectively, and they can be very different.

Theorem 4.3.1. *Under Assumptions 10, 11 and 12, if we apply Stoc-AGDA with stepsizes* $\tau_1 = \min\left\{\frac{\sqrt{\Delta}}{4\sigma\kappa^2\sqrt{Tl}}, \frac{1}{68l\kappa^2}\right\}$ *and* $\tau_2 = \min\left\{\frac{17\sqrt{\Delta}}{\sigma\sqrt{Tl}}, \frac{1}{l}\right\}$*, then we have*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla\Phi(x_t)\|^2 \leq \frac{1088l\kappa^2}{T}\Delta + \frac{136l\kappa^2}{T}a_0 + \frac{8\kappa^2\sqrt{l}a_0}{\sqrt{\Delta T}}\sigma + \frac{1232\kappa^2\sqrt{l\Delta}}{\sqrt{T}}\sigma,$$

where $\Delta = \Phi(x_0) - \Phi^*$ *and* $a_0 := \Phi(x_0) - f(x_0, y_0)$*. This implies a sample complexity of* $O\left(\frac{l\kappa^2\Delta}{\epsilon^2} + \frac{l\kappa^4\Delta\sigma^2}{\epsilon^4}\right)$ *to find an* ϵ*-stationary point of* Φ*.*

We can either use Proposition 2 to translate to the other notion with extra computations or show that Stoc-AGDA directly outputs an ϵ-stationary point of f with the same sample complexity.

Corollary 4.3.2. *Under the same setting as Theorem 4.3.1, the output* $(\hat{x}, \hat{y})$ *from Stoc-AGDA satisfies* $\mathbb{E}\|\nabla_x f(\hat{x}, \hat{y})\| \leq \epsilon$ *and* $\mathbb{E}\|\nabla_y f(\hat{x}, \hat{y})\| \leq \epsilon$ *after* $O\left(\frac{l\kappa^2\Delta}{\epsilon^2} + \frac{l\kappa^4\Delta\sigma^2}{\epsilon^4}\right)$ *iterations, which implies the same sample complexity as Theorem 4.3.1.*

Remark 4.3.3. *The dependency on* $a_0 = \Phi(x_0) - f(x_0, y_0)$ *can be improved by initializing* y_0 *with gradient ascent or stochastic gradient ascent to maximize the function* $f(x_0, \cdot)$ *satisfying the PL condition, which has exponential convergence in the deterministic setting and* $O(\frac{1}{T})$ *sublinear rate in the stochastic setting [Karimi et al., 2016].*

Remark 4.3.4. *The complexity above has different dependency as a function of* ϵ *and* κ *for the terms with and without the variance term* σ*. When* $\sigma = 0$*, iterations the output from AGDA after* $O\left(l\kappa^2\Delta\epsilon^{-2}\right)$ *will be an* ϵ*-stationary point of both* f *and* Φ*. It recovers the same complexity result in [Yang et al., 2020b] for the primal function stationary convergence. Nouiehed et al. [2019] show the same complexity for multi-GDA based on the stationary measure of* f*, which implies*

$O(l\kappa^3\Delta\epsilon^{-2})$ complexity for the stationary convergence of Φ by Proposition 2. See Table 6.1 for more comparisons.

Remark 4.3.5. *When $\sigma > 0$, we establish the brand-new sample complexity of $O(l\kappa^4\Delta\epsilon^{-4})$ for Stoc-AGDA. It is the first analysis of stochastic algorithms for NC-PL minimax problems. The dependency on ϵ is optimal, because the lower complexity bound of $\Omega(\epsilon^{-4})$ for stochastic nonconvex optimization [Arjevani et al., 2022] still holds when considering $f(x,y) = F(x)$ for some nonconvex function $F(x)$. Even under the strictly stronger assumption of imposing strong-concavity in y, to the best of our knowledge, it is the first time that vanilla stochastic GDA-type algorithm is showed to achieve $O(\epsilon^{-4})$ sample complexity without either increasing batch size as in [Lin et al., 2020a] or Lipschitz continuity of $f(\cdot,y)$ and its Hessian as in [Chen et al., 2021b]. In [Lin et al., 2020a], they show a worse complexity of $O(\epsilon^{-5})$ for GDA with $O(1)$ batch size. We refer the reader to Table 4.2.*

Remark 4.3.6. *We point out that under our weaker assumption, the dependency on the condition number κ is slightly worse than that in [Lin et al., 2020a, Chen et al., 2021b]. If only $O(1)$ samples are available in each iteration, Stoc-GDA only achieves $O(\epsilon^{-5})$ sample complexity [Lin et al., 2020a]. On the other hand, the analysis in [Chen et al., 2021a] is not applicable here. It uses a potential function $V_t = \Phi(x_t) + O(\mu)\|y_t - y^*(x_t)\|^2]$, where $y^*(x_t) = \operatorname{argmax}_y f(x,y)$. To show a descent lemma for $\mathbb{E}[V_t]$, it shows the Lipschitz smoothness of $y^*(\cdot)$, which heavily depends on Lipschtiz continuity of f and its hessian, while under PL condition $y^*(x)$ might not be unique and we no longer make additional Lipschitz assumptions. Instead, we present an analysis based on the potential function $V_t = \Phi(x_t) + O(1)[\Phi(x_t) - f(x_t,y_t)]$ (see Appendix 4.6.2).*

4.4 STOCHASTIC SMOOTHED AGDA

Algorithm 7 Stochastic Smoothed-AGDA

1: Input: (x_0, y_0, z_0), step sizes $\tau_1 > 0, \tau_2 > 0$
2: **for all** $t = 0, 1, 2, ..., T-1$ **do**
3: Draw two i.i.d. samples ξ_1^t, ξ_2^t
4: $x_{t+1} = x_t - \tau_1[G_x(x_t, y_t, \xi_1^t) + p(x_t - z_t)]$
5: $y_{t+1} = y_t + \tau_2 G_y(x_{t+1}, y_t, \xi_2^t)$
6: $z_{t+1} = z_t + \beta(x_{t+1} - z_t)$
7: **end for**
8: Output: choose $(\hat{x}, \hat{y})$ uniformly from $\{(x_t, y_t)\}_{t=0}^{T-1}$

Stochastic Smoothed-AGDA presented in Algorithm 7 is closely related to proximal point method (PPM) on the primal function $\Phi(\cdot)$. In each iteration, we consider solving an auxiliary problem: $\min_x \Phi(x) + \frac{p}{2}\|x - z_t\|^2$, which is equivalent to:

$$\min_x \max_y \hat{f}(x, y; z_t) \triangleq f(x, y) + \frac{p}{2}\|x - z_t\|^2,$$

where z_t is called a proximal center to be defined later. Recently, proximal type algorithms including Catalyst have been shown to efficiently accelerate minimax optimization [Lin et al., 2020b, Yang et al., 2020b, Zhang et al., 2021b, Luo et al., 2021]. While these algorithms require multiple loops to solve the auxiliary problem to some high accuracy[1], Stoc-Smoothed-AGDA only applies one step of Stoc-AGDA to solve it from the point (x_t, y_t) as in step 4 and 5. Step 6 in Algorithm 7 with some $\beta \in (0, 1)$ guarantees that the proximal point z_t in the auxiliary problem is not too far from the previous one z_{t-1}. Smoothed-AGDA was first introduced by Zhang et al. [2020a] in the deterministic nonconvex-concave minimax optimization. To the best of our knowledge, its convergence has not been discussed in either the stochastic or the NC-PL setting.

Stoc-Smoothed-AGDA still maintains the single-loop structure and use only $O(1)$ samples in each iteration. If we choose $\beta = 1$ or $p = 0$, it reduces to Stoc-AGDA. Later in the analysis, we choose $p = 2l$ so that the auxiliary problem is l-strongly convex in x. We will see in the next theorem that this quadratic regularization term enables Smoothed-AGDA to take larger stepsizes for x compared to AGDA. In Smoothed-AGDA, the ratio between stepsize of x and y is $\Theta(1)$[2], while this ratio is $\Theta(1/\kappa^2)$ in AGDA.

Theorem 4.4.1. *Under Assumptions 10, 11 and 12, if we apply Algorithm 7 with* $\tau_1 = \min\left\{\frac{\sqrt{\Delta}}{2\sigma\sqrt{Tl}}, \frac{1}{3l}\right\}$, $\tau_2 = \min\left\{\frac{\sqrt{\Delta}}{96\sigma\sqrt{Tl}}, \frac{1}{144l}\right\}$, $p = 2l$ *and* $\beta = \frac{\tau_2\mu}{1600}$, *then*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\left\{ \|\nabla_x f(x_t, y_t)\|^2 + \kappa \left\|\nabla_y f(x_t, y_t)\right\|^2 \right\} \leq \frac{c_0 l\kappa}{T}[\Delta + b_0] + \frac{c_1\kappa\sqrt{l}b_0}{\sqrt{\Delta T}}\sigma + \frac{c_2\kappa\sqrt{l\Delta}}{\sqrt{T}}\sigma,$$

where $\Delta = \Phi(z_0) - \Phi^*$ *and* $b_0 = 2\,\mathrm{gap}_{\hat{f}(\cdot,\cdot;z_0)}(x_0, y_0)$ *is the primal-dual gap of the first auxiliary function at the initial point, and* c_0, c_1 *and* c_2 *are* $O(1)$ *constants. This implies the sample complexity of* $O\left(\frac{l\kappa\Delta}{\epsilon^2} + \frac{l\kappa^2\Delta\sigma^2}{\epsilon^4}\right)$ *to find an* $(\epsilon, \epsilon/\sqrt{\kappa})$*-stationary point of* f.

Remark 4.4.2. *In the theorem above,* b_0 *measures the optimality of* (x_0, y_0) *in the first auxiliary problem:* $\min_x \max_y f(x, y) + l\|x - z_0\|^2$, *which is l-strongly convex about x and* μ*-PL about y. Therefore, the dependency on* b_0 *can be reduced if we initialize* (x_0, y_0) *by approximately solving the first auxiliary problem with (Stochastic) AGDA, which converges exponentially in the deterministic setting and sublinearly at* $O(1/T)$ *rate in the stochastic setting for strongly-convex-PL minimax optimization [Yang et al., 2020a].*

1 In Appendix 4.6.4, we present a two-loop Catalyst algorithm combined with AGDA (Catalyst-AGDA) that achieves the same complexity as Algorithm 7 in the deterministic setting.

2 In Appendix 4.6.4, we show Catalyst-AGDA takes the stepsizes of the same order in the deterministic setting.

By Proposition 2, we can convert the output from Stoc-Smoothed-AGDA to an $O(\epsilon)$-stationary point of Φ.

Corollary 4.4.3. *From the output $(\hat{x}, \hat{y})$ of stochastic Smoothed-AGDA, we can apply (stochastic) AGDA to find an $O(\epsilon)$-stationary point of Φ by approximately solving $\min_x \max_y f(x,y) + l\|x - \hat{x}\|^2$. The total complexity is $O\left(\frac{l\kappa\Delta}{\epsilon^2}\right)$ in the deterministic setting and $\tilde{O}\left(\frac{l\kappa\Delta}{\epsilon^2} + \frac{l\kappa^2\Delta\sigma^2}{\epsilon^4} + \frac{\kappa^5\sigma^2}{\epsilon^2}\right)$ in the stochastic setting.*

Remark 4.4.4. *In the deterministic setting, the translation cost is $\kappa\log(\kappa)$, which is dominated by the complexity of finding $(\epsilon, \epsilon/\sqrt{\kappa})$-stationary point of f in Theorem 4.4.1. In the stochastic setting, the extra translation cost $\tilde{O}\left(\frac{\kappa^5\sigma^2}{\epsilon^2}\right)$ is low in the dependency of $\frac{1}{\epsilon}$ but larger in terms of the condition number. In practice, the inverse of the target accuracy is usually large. We leave the question of reducing translation cost and whether Stocastic Smoothed-AGDA can directly output an approximate stationary point of Φ to future research.*

Remark 4.4.5. *The term without variance σ has better dependency on ϵ and κ than the term with σ. In the deterministic setting, Smoothed-AGDA achieves the complexity of $O(l\kappa\Delta\epsilon^{-2})$, which improves over AGDA [Yang et al., 2020a] and Multi-AGDA [Nouiehed et al., 2019] with either notion of stationarity. Notably, this complexity under our weaker assumptions is better than that of other single-loop algorithms under a stronger assumption of strong-concavity in y (see Table 4.2). Recently, Zhang et al. [2021b] provide a tight lower bound of $O(l\sqrt{\kappa}\Delta\epsilon^{-2})$ for deterministic NC-SC minimax optimization. However, we do not expect the same complexity can be achieved under weaker assumptions.*

Remark 4.4.6. *In the stochastic setting, we show Stoc-Smoothed-AGDA achieves a sample complexity of $O(l\kappa^2\Delta\epsilon^{-4})$ for finding an ϵ-stationary point of f. To find an ϵ-stationary point of Φ, it bears an additional complexity of $O(\kappa^5\sigma^2\epsilon^{-2})$, which is negligible as long as ϵ is asymptotically small, i.e. when $\epsilon \leq \tilde{O}(\sqrt{\Delta/l\kappa^3})$. This sample complexity improves over $O(l\kappa^4\Delta\epsilon^{-4})$ sample complexity of Stoc-AGDA in NC-PL setting, and even $O(l\kappa^3\Delta\epsilon^{-4})$ complexity of Stoc-GDA [Lin et al., 2020a] and ALSET [Chen et al., 2021b] in NC-SC setting. Moreover, this sample complexity improvement comes without any large batch size, additional Lipschitz assumptions, or multi-loop structure. Very recently, Li et al. [2021] develop the lower complexity bound of $\Omega\left(\sqrt{\kappa}\epsilon^{-2} + \kappa^{1/3}\epsilon^{-4}\right)$ in NC-SC setting, but there is no matching upper bound yet.*

4.5 EXPERIMENTS

We illustrate the effectiveness of stochastic AGDA (Algorithm 6) and stochastic Smoothed-AGDA (Algorithm 7) for solving NC-PL min-max problems. In particular, we show that

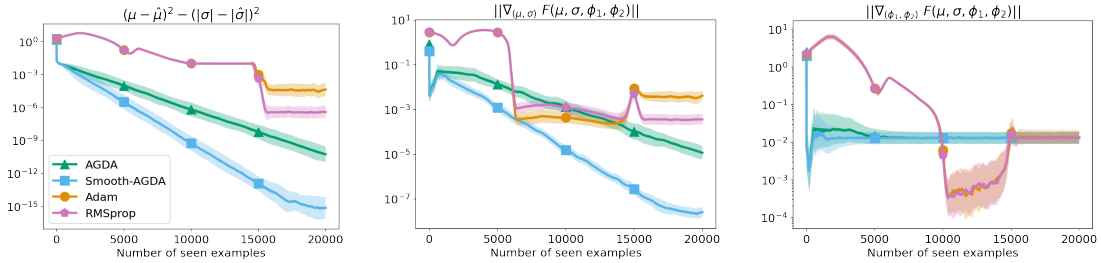


FIGURE 4.1: Training of a toy regularized WGAN with linear generator. Shown is the evolution of the *stochastic* gradients norm and the distance to the optimum. All methods are tuned at best for a minibatch size of 100, and each experiment is repeated 5 times (1 std shown). For Adam and RMSprop, we tuned over 4 learning rates $(1e-4, 5e-4, 1e-3, 5e-3)$ and 2 momentum parameters $0.5, 0.9$. The optimal configuration is obtained for a stepsize of $5e-4$ and momentum 0.5. For stochastic AGDA we considered each combination of $\tau_1, \tau_2 \in \{1e-2, 5e-2, 1e-1, 5e-1, 1\}$. The optimal configuration was found to be $\tau_1 = 5e-1, \tau_2 = 1$. For stochastic Smoothed-AGDA we use $\beta = 0.9$, $p = 10$ and tuned it to best: $\tau_1 = 5e-1, \tau_2 = 5e-1$.

the smoothed version of stochastic AGDA can compete with state-of-the-art deep learning optimizers [3].

TOY WGAN WITH LINEAR GENERATOR. We consider the same setting as [Loizou et al., 2020], i.e. using a Wasserstein GAN [Arjovsky et al., 2017] to approximate a one-dimensional Gaussian distribution. In particular, we have a dataset of real data x^{real} and latent variable z from a normal distribution with mean 0 and variance 1. The generator is defined as $G_{\mu,\sigma}(z) = \mu + \sigma z$ and the discriminator (a.k.a the critic) as $D_\phi(x) = \phi_1 x + \phi_2 x^2$, where x is either real data or fake data from the generator. The true data is generated from $\hat{\mu} = 0, \hat{\sigma} = 0.1$. The problem can be written in the form of:

$$\min_{\mu,\sigma} \max_{\phi_1,\phi_2} f(\mu,\sigma,\phi_1,\phi_2) \triangleq \mathbb{E}_{(x^{real},z)\sim\mathcal{D}}\ D_\phi(x^{real}) - D_\phi(G_{\mu,\sigma}(z)) - \lambda\|\phi\|^2,$$

where $\mathcal{D}$ is the distribution for the real data and latent variable, and the regularization $\lambda\|\phi\|^2$ with $\lambda = 0.001$ makes the problem strongly concave. This problem is non-convex in σ: indeed since z is symmetric around zero, both σ and $-\sigma$ are solutions. We fixed the batch size to 100 and tuned each algorithm at best (see plots in the appendix). Each experiment is repeated 3 times. In Figure 4.1 we provide evidence of the superiority of Stoc-Smoothed-AGDA over Stoc-AGDA, Adam [Kingma and Ba, 2014] and RMSprop [Tieleman et al., 2012]. As the reader can notice, Stoc-Smoothed-AGDA is competitive with fine-tuned popular adaptive methods, and provides a significant speedup over AGDA with carefully tuned learning rates, which verifies our theoretical results.

3 Code available at `https://github.com/aorvieto/NCPL.git`

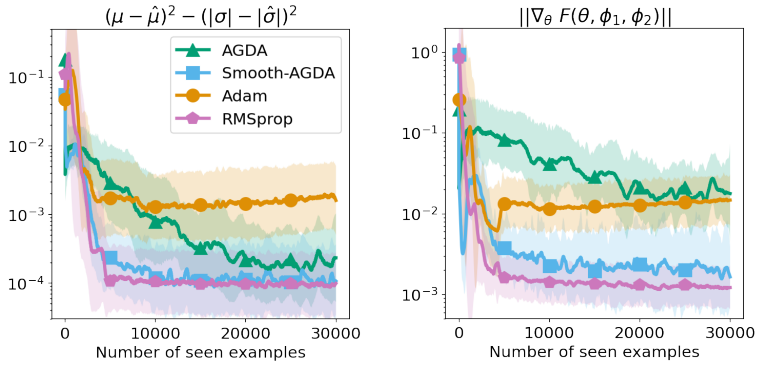


FIGURE 4.2: ReLU Network generator for a regularized WGAN (same settings as for Figure 4.1). Each algorithm is tuned to yield best performance, with a procedure similar to the one in Figure 4.1. The gradient with respect to the discriminator evolves very similarly to the last example, with fast convergence to a non-zero value.

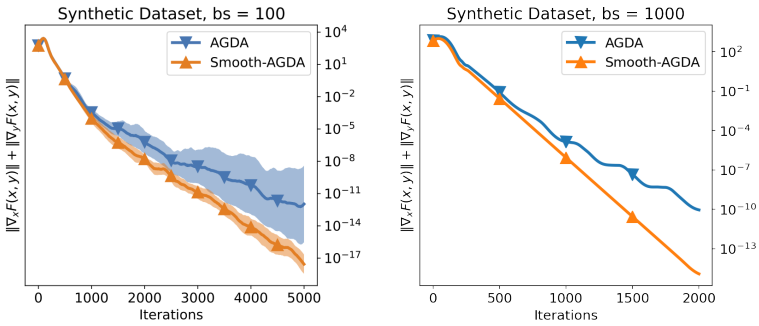


FIGURE 4.3: Robust non-linear regression on a synthetic Gaussian Dataset. Using $\tau_1 = 5e-4, \tau_2 = 5$ for both AGDA and Smoothed-AGDA, we notice a performance improvement for the latter using $\beta = 0.5, p = 10$.

TOY WGAN WITH NEURAL GENERATOR. Inspired by [Lei et al., 2020], we consider a regularized WGAN with a neural network as the generator. For ease of comparison, we leave all the problem settings identical to the last paragraph, and only change the generator $G_{\mu,\sigma}$ to G_θ, where θ are the parameters of a small neural network (one hidden layer with five neurons and ReLU activations). After careful tuning for each algorithm, we observe from Figure 4.2 that Stoc-Smoothed-AGDA still performs significantly better than vanilla Stoc-AGDA and Adam in this setting. The adaptiveness (without momentum) of RMSprop is able to yield slightly better results. This is not surprising, as adaptive methods are the de facto optimizers of choice in generative adversarial nets. Hence, a clear direction of future research is to combine adaptiveness and Smoothed-AGDA.

ROBUST NON-LINEAR REGRESSION. The experiments above suggest that Smoothed-AGDA accelerates convergence of AGDA. We found that this holds true also outside the

WGAN setting: in this last paragraph, we show how this accelerated behavior in a few robust regression problems. We first consider a synthetic dataset of 1000 datapoints z in 500 dimensions, sampled from a Gaussian distribution with mean zero and variance 1. The target values y_0 are sampled according to a random noisy linear model. We consider fitting this synthetic dataset with a two-hidden-layer ReLU network (256 units in the first layer, 64 in the second): $\text{net}_x(z)$ with x being the parameter. For the robustness part, we proceed in the standard way (see e.g.[Adolphs et al., 2019]) and add the concave objective $-\frac{\lambda}{2}\|y - y_0\|^2$ to the loss:

$$F(x, y) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}\|\text{net}_x(z) - y\|^2 - \frac{\lambda}{2}\|y - y_0\|^2,$$

where we chose $\lambda = 1$. In this experiement, we compare the performance of AGDA and Smoothed-AGDA under the same stepsize τ_1, τ_2. From Figure 4.3, we observe that Smoothed-AGDA has much faster convergence than AGDA both in the stochastic and deterministic setting (i.e. with full batch).

4.6 APPENDIX

4.6.1 *Useful Lemmas*

Lemma 4.6.1 (Lemma B.2 [Lin et al., 2020b]). *Assume $f(\cdot, y)$ is μ_x-strongly convex for $\forall y \in \mathbb{R}^{d_2}$ and $f(x, \cdot)$ is μ_y-strongly concave for $\forall x \in \mathbb{R}^{d_1}$ (we will later refer to this as (μ_x, μ_y)-SC-SC)) and f is l-Lipschitz smooth. Then we have*

a) $y^*(x) = \operatorname{argmax}_{y \in \mathbb{R}^{d_2}} f(x, y)$ *is* $\frac{l}{\mu_y}$*-Lipschitz;*

b) $\Phi(x) = \max_{y \in \mathbb{R}^{d_2}} f(x, y)$ *is* $\frac{2l^2}{\mu_y}$*-Lipschitz smooth and* μ_x*-strongly convex with* $\nabla \Phi(x) = \nabla_x f(x, y^*(x))$*;*

c) $x^*(y) = \operatorname{argmin}_{x \in \mathbb{R}^{d_1}} f(x, y)$ *is* $\frac{l}{\mu_x}$*-Lipschitz;*

d) $\Psi(y) = \min_{x \in \mathbb{R}^{d_1}} f(x, y)$ *is* $\frac{2l^2}{\mu_x}$*-Lipschitz smooth and* μ_y*-strongly concave with* $\nabla \Psi(y) = \nabla_y f(x^*(y), y)$*.*

Lemma 4.6.2 (Karimi et al. [2016]). *If $f(\cdot)$ is l-smooth and it satisfies PL condition with constant μ, i.e.*

$$\|\nabla f(x)\|^2 \geq 2\mu[f(x) - \min_x f(x)], \forall x,$$

then it also satisfies error bound (EB) condition with μ, i.e.

$$\|\nabla f(x)\| \geq \mu \|x_p - x\|, \forall x,$$

where x_p is the projection of x onto the optimal set, and it satisfies quadratic growth (QG) condition with μ, i.e.

$$f(x) - \min_x f(x) \geq \frac{\mu}{2} \|x_p - x\|^2, \forall x.$$

Lemma 4.6.3 (Nouiehed et al. [2019]). *Under Assumption 10 and 11, define $\Phi(x) = \max_y f(x, y)$ then*

a) for any x_1, x_2, and $y^*(x_1) \in \operatorname{Argmax}_y f(x_1, y)$, there exists some $y^*(x_2) \in \operatorname{Argmax}_y f(x_2, y)$ such that

$$\|y_1^* - y_2^*\| \leq \frac{l}{2\mu} \|x_1 - x_2\|.$$

b) $\Phi(\cdot)$ is L-smooth with $L := l + \frac{l\kappa}{2}$ with $\kappa = \frac{l}{\mu}$ and $\nabla \Phi(x) = \nabla_x f(x, y^*(x))$ for any $y^*(x) \in \operatorname{Argmax}_y f(x, y)$.

Now we present a Theorem adopted from [Yang et al., 2020a]. Under the two-sided PL condition, it captures the convergence of AGDA with dual updated first[4]:

$$\begin{aligned} y^{k+1} &= y^k + \tau_2 \nabla_y f(x^k, y^k), \\ x^{k+1} &= x^k - \tau_1 \nabla_x f(x^k, y^{k+1}). \end{aligned} \tag{4.3}$$

Theorem 4.6.4 (Yang et al. [2020a])**.** *Consider a minimax optimization problem under Assumption 12:*

$$\min_{x \in \mathbb{R}^{d_1}} \max_{y \in \mathbb{R}^{d_2}} f(x, y) \triangleq \mathbb{E}[F(x, y; \xi)].$$

Suppose the function f is l-smooth, $f(\cdot, y)$ satisfies the PL condition with constant μ_1 and $-f(x, \cdot)$ satisfies the PL condition with constant μ_2 for any x and y. Define

$$P_k = \mathbb{E}[\Psi^* - \Psi(y_t)] + \frac{1}{10}\mathbb{E}[f(x^k, y^k) - \Psi(x^k)]$$

with $\Psi(y) = \min_x f(x, y)$ and $\Psi^* = \max_y \Psi(y)$. If we run Stoc-AGDA (with update rule (4.3)) with stepsizes $\tau_1 \leq \frac{1}{l}$ and $\tau_2 \leq \frac{\mu_1^2 \tau_1}{18 l^2}$, then

$$P_k \leq \left(1 - \frac{\mu_2 \tau_2}{2}\right)^k P_0 + \frac{23 l^2 \tau_2^2 / \mu_1 + l \tau_1^2}{10 \mu_2 \tau_2} \sigma^2. \tag{4.4}$$

In the deterministic setting, e.g. $\sigma = 0$, if we run AGDA with stepsizes $\tau_1 = \frac{1}{l}$ and $\tau_2 = \frac{\mu_1^2}{18 l^3}$ then

$$P_k \leq \left(1 - \frac{\mu_1^2 \mu_2}{36 l^3}\right)^k P_0. \tag{4.5}$$

Definition 17 (Moreau Envelope)**.** *The Moreau envelope of a function Φ with a parameter $\lambda > 0$ is:*

$$\Phi_\lambda(x) = \min_{z \in \mathbb{R}^{d_1}} \Phi(z) + \frac{1}{2\lambda}\|z - x\|^2.$$

The proximal point of x is defined as: $\text{prox}_{\lambda\Phi}(x) = \text{argmin}_{z \in \mathbb{R}^{d_1}} \left\{\Phi(z) + \frac{1}{2\lambda}\|z - x\|^2\right\}$. The gradients of Φ and and Φ_λ are closely related by the following well-known lemma; see e.g. [Drusvyatskiy and Paquette, 2019].

Lemma 4.6.5. *When F is differentiable and ℓ-Lipschitz smooth, for $\lambda \in (0, 1/\ell)$ we have $\nabla\Phi(prox_{\lambda F}(x)) = \nabla\Phi_\lambda(x) = \lambda^{-1}(x - prox_{\lambda\Phi}(x))$.*

4 The update is equivalent to applying AGDA with primal variable update first to $\min_y \max_x -f(x, y)$, so its convergence is a direct result from [Yang et al., 2020a]. We believe a similar convergence rate to Theorem 4.6.4 holds for AGDA with x update first. But for simplicity, here we consider update (4.3) without additional derivation.

PROOF OF PROPOSITION 2

Proof. We will prove Part (a) and (b) separately.
Part (a): If we can find $\hat{y}$ such that $\max_y f(\hat{x}, y) - f(\hat{x}, \hat{y}) \leq \frac{\epsilon^2}{l\kappa}$, then as $\|\nabla_y f(\hat{x}, y^*(\hat{x}))\| = 0$,

$$\begin{aligned}\|\nabla_y f(\hat{x}, \hat{y})\| &= \|\nabla_y f(\hat{x}, \hat{y}) - \nabla_y f(\hat{x}, y^*(\hat{x}))\| \\ &\leq l\|\hat{y} - y^*(\hat{x})\| \leq l\sqrt{\frac{2}{\mu}[\max_y f(\hat{x}, y) - f(\hat{x}, \hat{y})]} \leq \sqrt{2}\epsilon,\end{aligned}$$

where in the first inequality we fix $y^*(x)$ to the projection from $\hat{y}$ to $\text{Argmax}_y f(\hat{x}, y)$, in the second inequality we use Lipschitz smoothness, and in the third inequality we use PL condition and Lemma 4.6.2. Also,

$$\begin{aligned}\|\nabla_x f(\hat{x}, \hat{y})\| &\leq \|\nabla_x f(\hat{x}, y^*(\hat{x})\| + \|\nabla_x f(\hat{x}, \hat{y}) - \nabla_x f(\hat{x}, y^*(\hat{x}))\| \\ &\leq \|\nabla\Phi(\hat{x})\| + l\|\hat{y} - y^*(\hat{x})\| \\ &\leq \|\nabla\Phi(\hat{x})\| + l\sqrt{\frac{2}{\mu}[\max_y f(\hat{x}, y) - f(\hat{x}, \hat{y})]} \leq (1 + \sqrt{2})\epsilon,\end{aligned}$$

where in the second inequality we use Lemma 4.6.3. Therefore, our goal is to find $\hat{y}$ such that $\max_y f(\hat{x}, y) - f(\hat{x}, \hat{y}) \leq \frac{\epsilon^2}{l\kappa}$ by applying (stochastic) gradient ascent to $f(\hat{x}, \cdot)$ from initial point $\tilde{y}$.

Deterministic case: Since $\|\nabla_y f(\hat{x}, \tilde{y})\| \leq \epsilon'$, we have $\max_y f(\hat{x}, y) - f(\hat{x}, \tilde{y}) \leq \frac{\epsilon'^2}{2\mu}$ by PL condition. Let y^k denote k-th iterates of gradient ascent from initial point $\tilde{y}$ with stepsize $\frac{1}{l}$. Then by [Karimi et al., 2016]

$$\max_y f(\hat{x}, y) - f(\hat{x}, y^k) \leq \left(1 - \frac{1}{\kappa}\right)^k \left[\max_y f(\hat{x}, y) - f(\hat{x}, \tilde{y})\right].$$

So after $O\left(\kappa \log\left(\frac{\kappa\epsilon'}{\epsilon}\right)\right)$, we can find the point we want.

Stochastic Case: Let y^k denote k-th iterates of stochastic gradient ascent from initial point $\tilde{y}$ with stepsize $\tau \leq \frac{1}{l}$. Then by Lemma A.4 in [Yang et al., 2020b]

$$\mathbb{E}\left[\max_y f(\hat{x}, y)) - f(\hat{x}, y^{k+1})\right] \leq (1 - \mu\tau)\mathbb{E}\left[\max_y f(\hat{x}, y)) - f(\hat{x}, y^k)\right] + \frac{l\tau^2}{2}\sigma^2,$$

which implies

$$\mathbb{E}\left[\max_y f(\hat{x}, y)) - f(\hat{x}, y^k)\right] \leq (1 - \mu\tau)^k\mathbb{E}\left[\max_y f(\hat{x}, y)) - f(\hat{x}, \tilde{y})\right] + \frac{\kappa\tau}{2}\sigma^2.$$

So with $\tau = \min\left\{\frac{1}{l}, \Theta\left(\frac{\epsilon^2}{l\kappa^2\sigma^2}\right)\right\}$, we can find the point we want with a complexity of $O\left(\kappa \log\left(\frac{\kappa\epsilon'}{\epsilon}\right) + \kappa^3\sigma^2 \log\left(\frac{\kappa\epsilon'}{\epsilon}\right)\epsilon^{-2}\right)$.

Part (b): We first look at $\Phi_{1/2l}(\tilde{x}) = \min_z \Phi(z) + l\|z - \tilde{x}\|^2$. Then by Lemma 4.3 in [Drusvyatskiy and Paquette, 2019],

$$\begin{aligned}
&\|\nabla\Phi_{1/2l}(\tilde{x})\|^2 \\
&= 4l^2\|\tilde{x} - \mathrm{prox}_{\Phi/2l}(\tilde{x})\|^2 \\
&\le 8l[\Phi(\tilde{x}) - \Phi(\mathrm{prox}_{\Phi/2l}(\tilde{x})) - l\|\mathrm{prox}_{\Phi/2l}(\tilde{x}) - \tilde{x}\|^2] \\
&= 8l\left[\Phi(\tilde{x}) - f(\tilde{x},\tilde{y}) + f(\tilde{x},\tilde{y}) - f(\mathrm{prox}_{\Phi/2l}(\tilde{x}),\tilde{y}) + f(\mathrm{prox}_{\Phi/2l}(\tilde{x}),\tilde{y}) - \Phi(\mathrm{prox}_{\Phi/2l}(\tilde{x}))\right. \\
&\qquad \left. -l\|\mathrm{prox}_{\Phi/2l}(\tilde{x}) - \tilde{x}\|^2\right] \\
&\le 8l\left[\frac{1}{2\mu}\|\nabla_y f(\tilde{x},\tilde{y})\|^2 + f(\tilde{x},\tilde{y}) - f(\mathrm{prox}_{\Phi/2l}(\tilde{x}),\tilde{y}) - l\|\mathrm{prox}_{\Phi/2l}(\tilde{x}) - \tilde{x}\|^2\right] \qquad (4.6)
\end{aligned}$$

where in the first inequality we use the l-strong-convexity in x of $\Phi(x) + l\|x - \tilde{x}\|^2$, in the second inequality we use $\Phi(\tilde{x}) - f(\tilde{x},\tilde{y}) \le \frac{1}{2\mu}\|\nabla_y f(\tilde{x},\tilde{y})\|^2$ by PL condition, and $f(\mathrm{prox}_{\Phi/2l}(\tilde{x}),\tilde{y}) - \Phi(\mathrm{prox}_{\Phi/2l}(\tilde{x})) \le 0$. Note that by defining $\hat{f}(x,y) = f(x,y) + l\|x - \tilde{x}\|^2$, we have

$$\begin{aligned}
&f(\tilde{x},\tilde{y}) - f(\mathrm{prox}_{\Phi/2l}(\tilde{x}),\tilde{y}) - l\|\mathrm{prox}_{\Phi/2l}(\tilde{x}) - \tilde{x}\|^2 \\
&= \hat{f}(\tilde{x},\tilde{y}) - \hat{f}(\mathrm{prox}_{\Phi/2l}(\tilde{x}),\tilde{y}) \\
&\le \langle\nabla_x f(\tilde{x},\tilde{y}), x - \mathrm{prox}_{\Phi/2l}(\tilde{x})\rangle - \frac{l}{2}\|x - \mathrm{prox}_{\Phi/2l}(\tilde{x})\|^2 \\
&\le \frac{1}{2l}\|\nabla_x \hat{f}(\tilde{x},\tilde{y})\|^2 + \frac{l}{2}\|x - \mathrm{prox}_{\Phi/2l}(\tilde{x})\|^2 - \frac{l}{2}\|x - \mathrm{prox}_{\Phi/2l}(\tilde{x})\|^2 \\
&\le \frac{1}{2l}\|\nabla_x \hat{f}(\tilde{x},\tilde{y})\|^2 = \frac{1}{2l}\|\nabla_x f(\tilde{x},\tilde{y})\|^2,
\end{aligned}$$

where in the second inequality we use l-strong-convexity in x of $\hat{f}(x,y)$. Plugging into (4.6),

$$\|\nabla\Phi_{1/2l}(\tilde{x})\|^2 = 4l^2\|\tilde{x} - \mathrm{prox}_{\Phi/2l}(\tilde{x})\|^2 \le 4\kappa\|\nabla_y f(\tilde{x},\tilde{y})\|^2 + 4\|\nabla_x f(\tilde{x},\tilde{y})\|^2 \le 8\epsilon. \qquad (4.7)$$

If we can find $\hat{x}$ such that $\|\mathrm{prox}_{\Phi/2l}(\tilde{x}) - \hat{x}\| \le \frac{\epsilon}{\kappa l}$, then

$$\begin{aligned}
\|\nabla\Phi(\hat{x})\| &\le \|\nabla\Phi(\mathrm{prox}_{\Phi/2l}(\tilde{x}))\| + \|\nabla\Phi(\hat{x}) - \nabla\Phi(\mathrm{prox}_{\Phi/2l}(\tilde{x}))\| \\
&\le \|\nabla\Phi_{1/2l}(\tilde{x})\| + 2\kappa l\|\mathrm{prox}_{\Phi/2l}(\tilde{x}) - \hat{x}\| \le (2\sqrt{2} + 2)\epsilon,
\end{aligned}$$

where in the second inequality we use Lemma 4.6.3 and Lemma 4.6.5. Note that $\mathrm{prox}_{\Phi/2l}(\tilde{x})$ is the solution to $\min_x \Phi(x) + l\|x - \tilde{x}\|^2$, which is equivalent to

$$\min_x \max_y f(x,y) + l\|x - \tilde{x}\|^2. \qquad (4.8)$$

This minimax problem is l-strongly convex about x, μ-PL about y and $3l$-smooth. Therefore, we can use (stochastic) alternating gradient descent ascent (AGDA) to find $\hat{x}$ such that $\|\mathrm{prox}_{\Phi/2l}(\tilde{x}) - \hat{x}\| \le \frac{\epsilon}{\kappa l}$ from initial point $(\tilde{x},\tilde{y})$.

Deterministic case: Let (x^k, y^k) denote k-th iterates of AGDA with y updated first from initial point $(\tilde{x}, \tilde{y})$ on function (4.8). Define $\hat{\Phi}(x) = \max_y \hat{f}(x,y) = \max_y f(x,y) + l\|x - \tilde{x}\|^2$, $\hat{\Psi}(y) = \min_x \hat{f}(x,y) = \min_x f(x,y) + l\|x - \tilde{x}\|^2$ and $\hat{\Psi}^* = \max_y \hat{\Psi}(y)$. We also denote $x^* = \operatorname{argmin}_x \hat{\Phi}(x) = \operatorname{prox}_{\Phi/2l}(\tilde{x})$. Then we define $P_k = \hat{\Psi}^* - \hat{\Psi}(y^k) + \frac{1}{10}\left[\hat{f}(x^k, y^k) - \hat{\Psi}(y^k)\right]$. Note that

$$\begin{aligned} P_0 = \hat{\Psi}^* - \hat{\Psi}(\tilde{y}) + \frac{1}{10}\left[\hat{f}(\tilde{x},\tilde{y}) - \hat{\Psi}(\tilde{y})\right] &\le \hat{\Psi}^* - \hat{\Psi}(\tilde{y}) + \frac{1}{20l}\|\nabla_x \hat{f}(\tilde{x},\tilde{y})\|^2 \\ &\le \hat{\Psi}^* - \hat{\Psi}(\tilde{y}) + \frac{\epsilon^2}{20l}. \end{aligned} \tag{4.9}$$

Also, we note that

$$\begin{aligned} \hat{\Psi}^* - \hat{\Psi}(\tilde{y}) &= \max_y \min_x \hat{f}(x,y) - \min_x \hat{f}(x,\tilde{y}) \\ &= \max_y \min_x \hat{f}(x,y) - \max_y \hat{f}(\tilde{x},y) + \max_y \hat{f}(\tilde{x},y) - \hat{f}(\tilde{x},\tilde{y}) + \hat{f}(\tilde{x},\tilde{y}) - \min_x \hat{f}(x,\tilde{y}) \\ &\le \frac{1}{2\mu}\|\nabla_y \hat{f}(\tilde{x},\tilde{y})\|^2 + \frac{1}{2l}\|\nabla_x \hat{f}(\tilde{x},\tilde{y})\|^2 = \frac{1}{2\mu}\|\nabla_y f(\tilde{x},\tilde{y})\|^2 + \frac{1}{2l}\|\nabla_x f(\tilde{x},\tilde{y})\|^2 \le \frac{\epsilon^2}{l}, \end{aligned}$$

where in the first inequality we use $\max_y \min_x \hat{f}(x,y) \le \max_y \hat{f}(\tilde{x},y)$, l-strong-convexity of $\hat{f}(\cdot,\tilde{y})$ and μ-PL of $\hat{f}(\tilde{x},\cdot)$. Combined with (4.9) we have

$$P_0 \le \frac{2\epsilon^2}{l}.$$

Then we note that

$$\begin{aligned} \|x^k - x^*\|^2 &\le 2\|x^k - x^*(y^k)\|^2 + 2\|x^*(y^k) - x^*\|^2 \le \frac{4}{l}[\hat{f}(x^k,y^k) - \hat{\Psi}(y^k)] + 18\|y^k - y^*\|^2 \\ &\le \frac{4}{l}[\hat{f}(x^k,y^k) - \hat{\Psi}(y^k)] + \frac{18}{\mu}[\hat{\Psi}(y^k) - \hat{\Psi}^*] \le \frac{40}{\mu}P_k, \end{aligned}$$

where in the second inequality we use l-strong-convexity of $\hat{f}(\cdot,y^k)$ and Lemma 4.6.1, in the third inequality we use μ-PL of $\hat{\Psi}(\cdot)$ (see e.g. [Yang et al., 2020a]). Because $\hat{f}(x,y)$ is l-strongly convex about x, μ-PL about y and $3l$-smooth, it satifies the two-sided PL condition in [Yang et al., 2020a] and it can be solved by AGDA. By Theorem 4.6.4, if we choose $\tau_1 = \frac{1}{3l}$ and $\tau_2 = \frac{l^2}{18(3l)^3} = \frac{1}{486l}$, we have

$$P_k \le (1 - \frac{1}{972\kappa})^k P_0,$$

Therefore,

$$\|x^k - x^*\|^2 \le \frac{40}{\mu}P_k \le \frac{40}{\mu}\left(1 - \frac{1}{972\kappa}\right)^k P_0 \le \frac{80\epsilon^2}{\mu l}\left(1 - \frac{1}{972\kappa}\right)^k.$$

So after $O(\kappa \log \kappa)$ iterations we have $\|x^k - x^*\|^2 \le \frac{\epsilon^2}{\kappa^2 l^2}$.

Stochastic case: By Theorem 4.6.4, if we choose $\tau_1 \le \frac{1}{3l}$ and $\tau_2 = \frac{l^2\tau_1}{18(3l)^2} = \frac{\tau_1}{162}$, we have

$$P_k \le \left(1 - \frac{\mu\tau_2}{2}\right)^k P_0 + O(\kappa\tau_2\sigma^2).$$

With $\tau_2 = \min\left\{\frac{1}{486l}, \Theta\left(\frac{\epsilon^2}{\kappa^4 l \sigma^2}\right)\right\}$ and $\tau_1 = 162\tau_2$, we have $\|x^k - x^*\|^2 \leq \frac{\epsilon^2}{\kappa^2 l^2}$ after $O\left(\kappa\log(\kappa) + \kappa^5\sigma^2\log(\kappa)\epsilon^{-2}\right)$ iterations.

□

4.6.2 *Proofs for Stochastic AGDA*

PROOF OF THEOREM 4.3.1

Proof. Because Φ is L-smooth with $L = l + \frac{l\kappa}{2}$ by Lemma 4.6.3, we have the following by Lemma A.4 in [Yang et al., 2020a]

$$\begin{aligned}\Phi(x_{t+1}) &\leq \Phi(x_t) + \langle\nabla\Phi(x_t), x_{t+1} - x_t\rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2 \\ &= \Phi(x_t) - \tau_1\langle\nabla\Phi(x_t), G_x(x_t, y_t, \xi_{t1})\rangle + \frac{L}{2}\tau_1^2\|G_x(x_t, y_t, \xi_1^t)\|^2.\end{aligned}$$

Taking expectations of both sides and using Assumption 12, we get

$$\begin{aligned}\mathbb{E}[\Phi(x_{t+1})] &\leq \mathbb{E}[\Phi(x_t)] - \tau_1\mathbb{E}[\langle\nabla\Phi(x_t), \nabla_x f(x_t, y_t)\rangle] + \frac{L}{2}\tau_1^2\mathbb{E}[\|G_x(x_t, y_t, \xi_1^t)\|^2] \\ &\leq \mathbb{E}[\Phi(x_t)] - \tau_1\mathbb{E}[\langle\nabla\Phi(x_t), \nabla_x f(x_t, y_t)\rangle] + \frac{L}{2}\tau_1^2\mathbb{E}[\|\nabla_x f(x_t, y_t)\|^2] + \frac{L}{2}\tau_1^2\sigma^2 \\ &\leq \mathbb{E}[\Phi(x_t)] - \tau_1\mathbb{E}[\langle\nabla\Phi(x_t), \nabla_x f(x_t, y_t)\rangle] + \frac{\tau_1}{2}\mathbb{E}[\|\nabla_x f(x_t, y_t)\|^2] + \frac{L}{2}\tau_1^2\sigma^2 \\ &\leq \mathbb{E}[\Phi(x_t)] - \frac{\tau_1}{2}\mathbb{E}\|\nabla\Phi(x_t)\|^2 + \frac{\tau_1}{2}\mathbb{E}\|\nabla_x f(x_t, y_t) - \nabla\Phi(x_t)\|^2 + \frac{L}{2}\tau_1^2\sigma^2,\end{aligned} \tag{4.10}$$

where in the second inequality we use Assumption 12, and in the third inequality we use $\tau_1 \leq 1/L$. By smoothness of $f(x, \cdot)$, we have

$$\begin{aligned}f(x_{t+1}, y_{t+1}) &\geq f(x_{t+1}, y_t) + \langle\nabla_y f(x_{t+1}, y_t), y_{t+1} - y_t\rangle - \frac{l}{2}\|y_{t+1} - y_t\|^2 \\ &\geq f(x_{t+1}, y_t) + \tau_2\langle\nabla_y f(x_{t+1}, y_t), G_y(x_{t+1}, y_t, \xi_2^t)\rangle - \frac{l\tau_2^2}{2}\|G_y(x_{t+1}, y_t, \xi_2^t)\|^2.\end{aligned}$$

Taking expectation, as $\tau_2 \leq \frac{1}{l}$

$$\begin{aligned}\mathbb{E}f(x_{t+1}, y_{t+1}) - \mathbb{E}f(x_{t+1}, y_t) &\geq \tau_2\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - \frac{l\tau_2^2}{2}\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - \frac{l\tau_2^2}{2}\sigma^2 \\ &\geq \frac{\tau_2}{2}\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - \frac{l\tau_2^2}{2}\sigma^2.\end{aligned} \tag{4.11}$$

By smoothness of $f(\cdot, y)$, we have

$$\begin{aligned}f(x_{t+1}, y_t) &\geq f(x_t, y_t) + \langle\nabla_x f(x_t, y_t), x_{t+1} - x_t\rangle - \frac{l}{2}\|x_{t+1} - x_t\|^2 \\ &\geq f(x_t, y_t) - \tau_1\langle\nabla_x f(x_t, y_t), G_x f(x_t, y_t, \xi_1^t)\rangle - \frac{l\tau_1^2}{2}\|G_x(x_t, y_t, \xi_1^t)\|^2.\end{aligned}$$

Taking expectation, as $\tau_1 \leq \frac{1}{l}$

$$\begin{aligned}\mathbb{E}f(x_{t+1},y_t)-\mathbb{E}f(x_t,y_t) &\geq -\tau_1\mathbb{E}\|\nabla_x f(x_t,y_t)\|-\frac{l\tau_1^2}{2}\mathbb{E}\|\nabla_x f(x_t,y_t)\|-\frac{l\tau_1^2}{2}\sigma^2\\ &\geq -\frac{3\tau_1}{2}\mathbb{E}\|\nabla_x f(x_t,y_t)\|^2-\frac{l\tau_1^2}{2}\sigma^2. \end{aligned} \tag{4.12}$$

Therefore, summing (4.12) and (4.11) together

$$\begin{aligned}&\mathbb{E}f(x_{t+1},y_{t+1})-\mathbb{E}f(x_t,y_t)\\ &\geq \frac{\tau_2}{2}\mathbb{E}\|\nabla_y f(x_{t+1},y_t)\|^2-\frac{3\tau_1}{2}\mathbb{E}\|\nabla_x f(x_t,y_t)\|^2-\frac{l\tau_1^2}{2}\sigma^2-\frac{l\tau_2^2}{2}\sigma^2. \end{aligned} \tag{4.13}$$

Now we consider the following potential function, for some $\alpha > 0$ which we will pick later

$$V_t \triangleq V(x_t,y_t) \triangleq \Phi(x_t)+\alpha[\Phi(x_t)-f(x_t,y_t)] = (1+\alpha)\Phi(x_t)-\alpha f(x_t,y_t).$$

Then by combining (4.10) and (4.13) we have

$$\begin{aligned}&\mathbb{E}V_t-\mathbb{E}V_{t+1}\\ &\geq \frac{\tau_1}{2}(1+\alpha)\mathbb{E}\left\|\nabla\Phi\left(x_t\right)\right\|^2-\frac{\tau_1}{2}(1+\alpha)\mathbb{E}\left\|\nabla_x f\left(x_t,y_t\right)-\nabla\Phi\left(x_t\right)\right\|^2+\frac{\tau_2\alpha}{2}\mathbb{E}\|\nabla_y f(x_{t+1},y_t)\|^2-\\ &\quad\frac{3\tau_1\alpha}{2}\mathbb{E}\|\nabla_x f(x_t,y_t)\|^2-\left[\frac{L(1+\alpha)}{2}\tau_1^2+\frac{l\tau_2^2\alpha}{2}+\frac{l\tau_1^2\alpha}{2}\right]\sigma^2\\ &\geq \left[\frac{\tau_1}{2}(1+\alpha)-3\tau_1\alpha\right]\mathbb{E}\left\|\nabla\Phi\left(x_t\right)\right\|^2-\left[\frac{\tau_1}{2}(1+\alpha)+3\tau_1\alpha\right]\mathbb{E}\left\|\nabla_x f\left(x_t,y_t\right)-\nabla\Phi\left(x_t\right)\right\|^2+\\ &\quad\frac{\tau_2\alpha}{4}\mathbb{E}\|\nabla_y f(x_t,y_t)\|^2-\frac{\tau_2\alpha}{2}\mathbb{E}\|\nabla_y f(x_{t+1},y_t)-\nabla_y f(x_t,y_t)\|^2-\left[\frac{L(1+\alpha)}{2}\tau_1^2+\frac{l\tau_2^2\alpha}{2}+\frac{l\tau_1^2\alpha}{2}\right]\sigma^2\\ &\geq \left[\frac{\tau_1}{2}(1+\alpha)-3\tau_1\alpha\right]\mathbb{E}\left\|\nabla\Phi\left(x_t\right)\right\|^2-\left[\frac{\tau_1 I}{2}(1+\alpha)+3\tau_1\alpha\right]\mathbb{E}\left\|\nabla_x f\left(x_t,y_t\right)-\nabla\Phi\left(x_t\right)\right\|^2+\\ &\quad\frac{\tau_2\alpha}{4}\mathbb{E}\|\nabla_y f(x_t,y_t)\|^2-\frac{\tau_2\alpha}{2}l^2\mathbb{E}\|x_{t+1}-x_t\|^2-\left[\frac{L(1+\alpha)}{2}\tau_1^2+\frac{l\tau_2^2\alpha}{2}+\frac{l\tau_1^2\alpha}{2}\right]\sigma^2\\ &\geq \left[\frac{\tau_1}{2}(1+\alpha)-3\tau_1\alpha\right]\mathbb{E}\left\|\nabla\Phi\left(x_t\right)\right\|^2-\left[\frac{\tau_1}{2}(1+\alpha)+3\tau_1\alpha\right]\mathbb{E}\left\|\nabla_x f\left(x_t,y_t\right)-\nabla\Phi\left(x_t\right)\right\|^2+\\ &\quad\frac{\tau_2\alpha}{4}\mathbb{E}\|\nabla_y f(x_t,y_t)\|^2-\frac{\tau_2\alpha}{2}l^2\tau_1^2\mathbb{E}\|\nabla_x f(x_t,y_t)\|^2-\left[\frac{L(1+\alpha)}{2}\tau_1^2+\frac{l\tau_2^2\alpha}{2}+\frac{l\tau_1^2\alpha}{2}+\frac{\tau_2}{2}\alpha l^2\tau_1^2\right]\sigma^2\\ &\geq \left[\frac{\tau_1}{2}(1+\alpha)-3\tau_1\alpha-\tau_2\alpha l^2\tau_1^2\right]\mathbb{E}\left\|\nabla\Phi\left(x_t\right)\right\|^2-\\ &\quad\left[\frac{\tau_1}{2}(1+\alpha)+3\tau_1\alpha+\tau_2\alpha l^2\tau_1^2\right]\mathbb{E}\left\|\nabla_x f\left(x_t,y_t\right)-\nabla\Phi\left(x_t\right)\right\|^2+\\ &\quad\frac{\tau_2\alpha}{4}\mathbb{E}\|\nabla_y f(x_t,y_t)\|^2-\left[\frac{L(1+\alpha)}{2}\tau_1^2+\frac{l\tau_2^2\alpha}{2}+\frac{l\tau_1^2\alpha}{2}+\frac{\tau_2}{2}\alpha l^2\tau_1^2\right]\sigma^2, \end{aligned} \tag{4.14}$$

where in the first inequality we use $\|a+b\|^2 \leq 2\|a\|^2+2\|b\|^2$ and $\|a\|^2 \geq \|b\|^2/2 - \|a-b\|^2$, in the second inequality we use smoothness, and in the last inequality we use

$\|a+b\|^2 \le \|a\|^2 + \|b\|^2$. Note that by smoothness and PL condition, fixing $y^*(x_t)$ to be the projection of y_t to the set $\underset{y}{\operatorname{Argmin}} f(x_t, y)$,

$$\|\nabla_x f(x_t, y_t) - \nabla\Phi(x_t)\|^2 \le l^2\|y_t - y^*(x_t)\|^2 \le \kappa^2\|\nabla_y f(x_t, y_t)\|^2.$$

Plugging it into (4.14), we get

$$\begin{aligned}\mathbb{E}V_t - \mathbb{E}V_{t+1} \ge &\left[\frac{\tau_1}{2}(1+\alpha) - 3\tau_1\alpha - \tau_2\alpha l^2\tau_1^2\right]\mathbb{E}\,\|\nabla\Phi(x_t)\|^2 + \\ &\left[\frac{\tau_2\alpha}{4} - \frac{\tau_1}{2}(1+\alpha)\kappa^2 - 3\tau_1\alpha\kappa^2 - \tau_2\alpha l^2\tau_1^2\kappa^2\right]\mathbb{E}\|\nabla_y f(x_t, y_t)\|^2 - \\ &\left[\frac{L(1+\alpha)}{2}\tau_1^2 + \frac{l\tau_2^2\alpha}{2} + \frac{l\tau_1^2\alpha}{2} + \frac{\tau_2}{2}\alpha l^2\tau_1^2\right]\sigma^2. \end{aligned} \tag{4.15}$$

Then we note that when $\alpha = \frac{1}{8}$, $\tau_1 \le \frac{1}{l}$ and $\tau_2 \le \frac{1}{l}$,

$$\frac{\tau_1}{2}(1+\alpha) - 3\tau_1\alpha - \tau_2\alpha l^2\tau_1^2 \ge \frac{\tau_1}{16}.$$

Furthermore, when $\tau_1 \le \frac{\tau_2}{68\kappa^2}$, then

$$\frac{\tau_2\alpha}{4} - \frac{\tau_1}{2}(1+\alpha)\kappa^2 - 3\tau_1\alpha\kappa^2 - \tau_2\alpha l^2\tau_1^2\kappa^2 \ge \frac{1}{64}\tau_2 \ge \frac{17}{16}\kappa^2\tau_1.$$

Also, as $\alpha = \frac{1}{8}$, $\tau_2 \le \frac{1}{l}$ and $\tau_1 = \frac{\tau_2}{68\kappa^2}$

$$\frac{L(1+\alpha)}{2}\tau_1^2 + \frac{l\tau_2^2\alpha}{2} + \frac{l\tau_1^2\alpha}{2} + \frac{\tau_2}{2}\alpha l^2\tau_1^2 \le 292\kappa^4 l\tau_1^2.$$

Therefore,

$$\mathbb{E}V_t - \mathbb{E}V_{t+1} \ge \frac{\tau_1}{16}\mathbb{E}\,\|\nabla\Phi(x_t)\|^2 + \frac{17}{16}\kappa^2\tau_1\mathbb{E}\|\nabla_y f(x_t, y_t)\|^2 - 292\kappa^4 l\tau_1^2\sigma^2. \tag{4.16}$$

Telescoping and rearraging, with $a_0 \triangleq \Phi(x_0) - f(x_0, y_0)$,

$$\begin{aligned}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\,\|\nabla\Phi(x_t)\|^2 &\le \frac{16}{\tau_1 T}[V_0 - \min_{x,y} V(x,y)] + 4762\kappa^4 l\tau_1\sigma^2 \\ &\le \frac{16}{\tau_1 T}[\Phi(x_0) - \Phi^*] + \frac{2}{\tau_1 T}a_0 + 4672\kappa^4 l\tau_1\sigma^2,\end{aligned}$$

where in the second inequality we note that since for any x we can find y such that $\Phi(x) = f(x, y)$,

$$\begin{aligned}V_0 - \min_{x,y} V(x,y) &= \Phi(x_0) + \alpha[\Phi(x_0) - f(x_0, y_0)] - \min_{x,y}\{\Phi(x) + \alpha[\Phi(x) - f(x,y)]\} \\ &= \Phi(x_0) - \Phi^* + \alpha[\Phi(x_0) - f(x_0, y_0)].\end{aligned}$$

Picking $\tau_1 = \min\left\{\frac{\sqrt{\Phi(x_0)-\Phi^*}}{4\sigma\kappa^2\sqrt{Tl}}, \frac{1}{68l\kappa^2}\right\}$,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\,\|\nabla\Phi(x_t)\|^2$$

$$\leq \max\left\{\frac{4\sigma\kappa^2\sqrt{Tl}}{\sqrt{\Phi(x_0)-\Phi^*}}, 68l\kappa^2\right\}\frac{16}{T}[\Phi(x_0)-\Phi^*] + \max\left\{\frac{4\sigma\kappa^2\sqrt{Tl}}{\sqrt{\Phi(x_0)-\Phi^*}}, 68l\kappa^2\right\}\frac{2}{T}a_0 + \frac{\sqrt{\Phi(x_0)-\Phi^*}}{4\sigma\kappa^2\sqrt{Tl}}4672\kappa^4 l\sigma^2$$

$$\leq \frac{1088l\kappa^2}{T}[\Phi(x_0)-\Phi^*] + \frac{136l\kappa^2}{T}a_0 + \frac{8\kappa^2\sqrt{l}a_0}{\sqrt{[\Phi(x_0)-\Phi^*]T}}\sigma + \frac{1232\kappa^2\sqrt{l[\Phi(x_0)-\Phi^*]}}{\sqrt{T}}\sigma.$$

Here we can pick $\tau_2 = \min\left\{\frac{17\sqrt{\Phi(x_0)-\Phi^*}}{\sigma\sqrt{Tl}}, \frac{1}{l}\right\}$.

□

PROOF OF COROLLARY 4.3.2

Proof. Similar to the proof of part (a) in Proposition 2, fixing $y^*(x_t)$ to be the projection of x_t to $\text{Argmax}_y f(x_t, y)$, we have

$$\begin{aligned}\|\nabla_x f(x_t, y_t)\|^2 &\leq 2\|\nabla_x f(x_t, y^*(x_t))\|^2 + 2\|\nabla_x f(x_t, y_t) - \nabla_x f(x_t, y^*(x_t))\|^2 \\ &\leq 2\|\nabla\Phi(x_t)\|^2 + 2l^2\|y_t - y^*(x_t)\|^2 \\ &\leq 2\|\nabla\Phi(x_t)\| + 2\kappa^2\|\nabla_y f(x_t, y_t)\|^2,\end{aligned}$$

where in the first inequality we use Lemma 4.6.3 and in the last inequality we use Lemma 4.6.2. Plugging into (4.16),

$$\mathbb{E}V_t - \mathbb{E}V_{t+1} \geq \frac{\tau_1}{32}\mathbb{E}\left\|\nabla\Phi\left(x_t\right)\right\|^2 + \kappa^2\tau_1\mathbb{E}\|\nabla_y f(x_t, y_t)\|^2 - 292\kappa^4 l\tau_1^2\sigma^2.$$

By the same reasoning as the proof of Theorem 4.3.1 (after equation (4.16)), with the same stepsizes, we can show

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left\|\nabla_x f\left(x_t, y_t\right)\right\|^2 + 32\kappa^2\mathbb{E}\left\|\nabla_y f\left(x_t, y_t\right)\right\|^2$$

$$\leq \frac{d_0 l\kappa^2}{T}[\Phi(x_0)-\Phi^*] + \frac{d_1 l\kappa^2}{T}a_0 + \frac{d_2\kappa^2\sqrt{l}a_0}{\sqrt{[\Phi(x_0)-\Phi^*]T}}\sigma + \frac{d_3\kappa^2\sqrt{l[\Phi(x_0)-\Phi^*]}}{\sqrt{T}}\sigma,$$

where d_0, d_1, d_2 and d_3 are $O(1)$ constants.

□

4.6.3 *Proofs for Stochastic Smoothed AGDA*

Before we present the theorems and proofs, we adopt the following notations.

- $\hat{f}(x, y; z) = f(x, y) + \frac{p}{2}\|x - z\|^2$: the auxiliary function;
- $\Psi(y; z) = \min_x \hat{f}(x, y; z)$: the dual function of the auxiliary problem;

- $\Phi(x;z) = \max_y \hat{f}(x,y;z)$: the primal function of the auxiliary problem;
- $P(z) = \min_x \max_y \hat{f}(x,y;z)$: the optimal value for the auxiliary function fixing z;
- $x^*(y,z) = \operatorname{argmin}_x \hat{f}(x,y;z)$: the optimal x w.r.t y and z in the auxiliary function;
- $x^*(z) = \operatorname{argmin}_x \Phi(x;z)$: the optimal x w.r.t z in the auxiliary function when y is already optimal w.r.t x;
- $Y^*(z) = \operatorname{Argmax}_y \Psi(y;z)$: the optimal set of y w.r.t z when x is optimal to y;
- $y^+(z) = y + \tau_2 \nabla_y \hat{f}(x^*(y,z),y;z)$: y after one step of gradient ascent in y with the gradient of the dual function;
- $x^+(y,z) = x - \tau_1 \nabla_x \hat{f}(x,y;z)$: x after one step of gradient descent with gradient at current point;
- $\hat{G}_x(x,y,\xi;z) = G_x(x,y,\xi) + p(x-z)$: the stochastic gradient for regularized auxiliary function.

Lemma 4.6.6. *We have the following inequalities as $p > l$*

$$\begin{aligned}
&\|x^*(y,z) - x^*(y,z')\| \le \gamma_1 \|z - z'\|,\\
&\|x^*(z) - x^*(z') \le \gamma_1 \|z - z'\|,\\
&\|x^*(y,z) - x^*(y',z)\| \le \gamma_2 \|y - y'\|,\\
&\mathbb{E}\|x_{t+1} - x^*(y_t,z_t)\|^2 \le \gamma_3^2 \tau_1^2 \mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + 2\tau_1^2\sigma^2,
\end{aligned}$$

where $\gamma_1 = \frac{p}{-l+p}$, $\gamma_2 = \frac{l+p}{-l+p}$ and $\gamma_3^2 = \frac{2}{\tau_1^2(-l+p)^2} + 2$.

Proof. The first and second inequality is the same as Proposition B.4 in [Zhang et al., 2020a]. The third inequality is a direct result of Lemma 4.6.1. Now we show the last inequality.

$$\begin{aligned}
\|x_{t+1} - x^*(y_t,z_t)\|^2 \le& 2\|x_t - x^*(y_t,z_t)\|^2 + 2\|x_{t+1} - x_t\|^2\\
\le& \frac{2}{(-l+p)^2}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + 2\tau_1^2 \|\hat{G}_x(x_t,y_t,\xi_1^t;z_t)\|^2.
\end{aligned}$$

where the second inequality use $(-l+p)$-strong convexity of $\hat{f}(\cdot,y_t;z_t)$. Taking expectation

$$\begin{aligned}
\mathbb{E}\|x_{t+1} - x^*(y_t,z_t)\|^2 \le& \frac{2}{(-l+p)^2}\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + 2\tau_1^2 \mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + 2\tau_1^2\sigma^2\\
\le& 2\left[\frac{1}{(-l+p)^2} + \tau_1^2\right]\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + 2\tau_1^2\sigma^2.
\end{aligned}$$

□

Lemma 4.6.7. *The following inequality holds*

$$\|x^*(z) - x^*(y^+(z),z)\|^2 \le \frac{1}{(p-l)\mu}\left(1 + \tau_2 l + \frac{\tau_2 l(p+l)}{p-l}\right)^2 \|\nabla_y \hat{f}(x^*(y,z),y;z)\|^2. \quad (4.17)$$

Proof. By the $(p-l)$-strong convexity of $\Phi(\cdot;z)$, we have

$$\begin{aligned}
&\|x^*(z) - x^*(y^+(z),z)\|^2 \\
&\leq \frac{2}{p-l}\left[\Phi(x^*(y^+(z),z);z) - \Phi(x^*(z);z)\right] \\
&\leq \frac{2}{p-l}\left[\Phi(x^*(y^+(z),z);z) - \hat{f}(x^*(y^+(z),z),y^+(z);z) + \hat{f}(x^*(y^+(z),z),y^+(z);z) - \Phi(x^*(z);z)\right] \\
&\leq \frac{1}{(p-l)\mu}\|\nabla_y \hat{f}(x^*(y^+(z),z),y^+(z);z)\|^2,
\end{aligned}$$

where in the last inequality we use μ-PL of $\hat{f}(x,\cdot;z)$ and $\hat{f}(x^*(y^+(z),z),y^+(z);z) \leq \Phi(x^*(z);z)$. Then

$$\begin{aligned}
&\|\nabla_y \hat{f}(x^*(y^+(z),z),y^+(z);z)\| \\
&\leq \|\nabla_y \hat{f}(x^*(y,z),y;z)\| + \|\nabla_y \hat{f}(x^*(y,z),y;z) - \nabla_y \hat{f}(x^*(y^+(z),z),y^+(z);z)\| \\
&\leq \|\nabla_y \hat{f}(x^*(y,z),y;z)\| + l\|x^*(y,z) - x^*(y^+(z),z)\| + l\|y - y^+(z)\| \\
&\leq \left(1 + \frac{\tau_2 l(p+l)}{p-l} + \tau_2 l\right)\|\nabla_y \hat{f}(x^*(y,z),y;z)\|,
\end{aligned}$$

where in the last inequality we use Lemma 4.6.6 and $\|y - y^+(z)\| = \tau_2\|\nabla_y \hat{f}(x^*(y,z),y;z)\|$. We reach our conclusion by combining with the previous inequality.

□

PROOF OF THEOREM 4.4.1

Proof. We separate our proof into several parts: we first present three descent lemmas, then we show the descent property for a potential function, later we discuss the relation between our stationary measure and the potential function, and last we put things together.

PRIMAL DESCENT: By the $(p+l)$-smoothness of $\hat{f}(\cdot,y_t;z_t)$,

$$\begin{aligned}
\hat{f}(x_{t+1},y_t;z_t) &\leq \hat{f}(x_t,y_t;z_t) + \langle \nabla_x \hat{f}(x_t,y_t;z_t), x_{t+1} - x_t\rangle + \frac{p+l}{2}\|x_{t+1} - x_t\|^2 \\
&= \hat{f}(x_t,y_t;z_t) - \tau_1\langle \nabla_x \hat{f}(x_t,y_t;z_t), \hat{G}_x(x_t,y_t,\xi_1^t;z_t)\rangle + \frac{p+l}{2}\tau_1^2\|\hat{G}_x(x_t,y_t,\xi_1^t;z_t)\|^2,
\end{aligned}$$

We can easily verify that $\mathbb{E}\hat{G}_x(x_t,y_t,\xi_1^t;z_t) = \nabla_x \hat{f}(x_t,y_t;z_t)$ and $\mathbb{E}\|\hat{G}_x(x_t,y_t,\xi_1^t;z_t) - \mathbb{E}\hat{G}_x(x_t,y_t,\xi_1^t;z_t)\|^2 = \mathbb{E}\|G_x(x_t,y_t,\xi_1^t) - \nabla_x f(x_t,y_t)\|^2 \leq \sigma^2$. Taking expectations of both sides, we have

$$\begin{aligned}
\mathbb{E}\hat{f}(x_{t+1},y_t;z_t) \leq \mathbb{E}\hat{f}(x_t,y_t;z_t) - \tau_1\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + \\
\frac{p+l}{2}\tau_1^2\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + \frac{p+l}{2}\tau_1^2\sigma^2.
\end{aligned}$$

As $\tau_1 \leq \frac{1}{p+l}$,

$$\mathbb{E}\hat{f}(x_t, y_t; z_t) - \mathbb{E}\hat{f}(x_{t+1}, y_t; z_t) \geq \frac{\tau_1}{2}\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 - \frac{p+l}{2}\tau_1^2\sigma^2. \tag{4.18}$$

Also, because $\hat{f}(x_{t+1}, \cdot; z_t)$ is smooth,

$$\begin{aligned}
&\hat{f}(x_{t+1}, y_t; z_t) - \hat{f}(x_{t+1}, y_{t+1}; z_t) \\
&\geq \langle \nabla_y \hat{f}(x_{t+1}, y_t; z_t), y_t - y_{t+1}\rangle - \frac{l}{2}\|y_t - y_{t+1}\|^2 \\
&= -\tau_2\langle \nabla_y \hat{f}(x_{t+1}, y_t; z_t), G_y(x_{t+1}, y_t, \xi_2^t)\rangle - \frac{l}{2}\tau_2^2\|G_y(x_{t+1}, y_t, \xi_2^t)\|^2.
\end{aligned}$$

Taking expectations of both sides,

$$\begin{aligned}
&\mathbb{E}\hat{f}(x_{t+1}, y_t; z_t) - \mathbb{E}\hat{f}(x_{t+1}, y_{t+1}; z_t) \\
&\geq -\tau_2\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - \frac{l}{2}\tau_2^2\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - \frac{l}{2}\tau_2^2\sigma^2 \\
&= -\left(1 + \frac{l\tau_2}{2}\right)\tau_2\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - \frac{l}{2}\tau_2^2\sigma^2.
\end{aligned} \tag{4.19}$$

Furthermore, by definition of $\hat{f}$ and z_{t+1}, as $0 < \beta < 1$,

$$\begin{aligned}
&\hat{f}(x_{t+1}, y_{t+1}; z_t) - \hat{f}(x_{t+1}, y_{t+1}; z_{t+1}) \\
&= \frac{p}{2}[\|x_{t+1} - z_t\|^2 - \|x_{t+1} - z_{t+1}\|^2] = \frac{p}{2}\left[\frac{1}{\beta^2}\|(z_{t+1} - z_t)\|^2 - \|(1-\beta)(x_{t+1} - z_t)\|^2\right] \\
&= \frac{p}{2}\left[\frac{1}{\beta^2}\|z_{t+1} - z_t\|^2 - \frac{(1-\beta)^2}{\beta^2}\|z_{t+1} - z_t\|^2\right] \geq \frac{p}{2\beta}\|z_t - z_{t+1}\|^2.
\end{aligned} \tag{4.20}$$

Combining (4.18), (4.19) and (4.20),

$$\begin{aligned}
\mathbb{E}\hat{f}(x_t, y_t; z_t) - \mathbb{E}\hat{f}(x_{t+1}, y_t; z_t) \geq \frac{\tau_1}{2}\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 - \left(1 + \frac{l\tau_2}{2}\right)\tau_2\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 + \\
\frac{p}{2\beta}\mathbb{E}\|z_t - z_{t+1}\|^2 - \frac{l}{2}\tau_2^2\sigma^2 - \frac{p+l}{2}\tau_1^2\sigma^2.
\end{aligned} \tag{4.21}$$

DUAL DESCENT: Since the dual function $\Psi(y; z)$ is L_Ψ smooth with $L_\Psi = l + l\gamma_2$ by Lemma B.3 in [Zhang et al., 2020a],

$$\begin{aligned}
\Psi(y_{t+1}; z_t) - \Psi(y_t; z_t) &\geq \langle \nabla_y \Psi(y_t; z_t), y_{t+1} - y_t\rangle - \frac{L_\Psi}{2}\|y_{t+1} - y_t\|^2 \\
&= \langle \nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t), y_{t+1} - y_t\rangle - \frac{L_\Psi}{2}\|y_{t+1} - y_t\|^2.
\end{aligned}$$

Taking expectation,

$$\begin{aligned}
\mathbb{E}\Psi(y_{t+1}; z_t) - \mathbb{E}\Psi(y_t; z_t) \geq \tau_2\mathbb{E}\langle \nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t), \nabla_y f(x_{t+1}, y_t)\rangle - \\
\frac{L_\Psi}{2}\tau_2^2\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - \frac{L_\Psi}{2}\tau_2^2\sigma^2.
\end{aligned} \tag{4.22}$$

Also,

$$\begin{aligned}\Psi(y_{t+1};z_{t+1})-\Psi(y_{t+1};z_t) =&\hat{f}(x^*(x_{t+1},z_{t+1}),y_{t+1};z_{t+1})-\hat{f}(x^*(y_{t+1},z_t),y_{t+1};z_t)\\ &\geq \hat{f}(x^*(x_{t+1},z_{t+1}),y_{t+1};z_{t+1})-\hat{f}(x^*(y_{t+1},z_{t+1}),y_{t+1};z_t)\\ &=\frac{p}{2}\left[\|z_{t+1}-x^*(y_{t+1},z_{t+1})\|^2-\|z_t-x^*(y_{t+1},z_{t+1})\|^2\right]\\ &=\frac{p}{2}(z_{t+1}-z_t)^\top[z_{t+1}+z_t-2x^*(y_{t+1},z_{t+1})]. \end{aligned}\tag{4.23}$$

Combining with (4.22), we have

$$\begin{aligned}&\mathbb{E}\Psi(y_{t+1};z_{t+1})-\mathbb{E}\Psi(y_t;z_t)\\ \geq\ &\tau_2\mathbb{E}\langle\nabla_y\hat{f}(x^*(y_t,z_t),y_t;z_t),\nabla_y f(x_{t+1},y_t)\rangle-\frac{L_\Psi}{2}\tau_2^2\mathbb{E}\|\nabla_y f(x_{t+1},y_t)\|^2+\\ &\frac{p}{2}\mathbb{E}(z_{t+1}-z_t)^\top[z_{t+1}+z_t-2x^*(y_{t+1},z_{t+1})]-\frac{L_\Psi}{2}\tau_2^2\sigma^2. \end{aligned}\tag{4.24}$$

PROXIMAL DESCENT: for all $y^*(z_{t+1})\in Y^*(z_{t+1})$ and $y^*(z_t)\in Y^*(z_t)$,

$$\begin{aligned}P(z_{t+1})-P(z_t)&=\Psi(y^*(z_{t+1});z_{t+1})-\Psi(y^*(z_t);z_t)\\ &\leq\Psi(y^*(z_{t+1});z_{t+1})-\Psi(y^*(z_{t+1});z_t)\\ &=\hat{f}(x^*(y^*(z_{t+1}),z_{t+1}),y^*(z_{t+1});z_{t+1})-\hat{f}(x^*(y^*(z_{t+1}),z_t),y^*(z_{t+1});z_t)\\ &\leq\hat{f}(x^*(y^*(z_{t+1}),z_t),y^*(z_{t+1});z_{t+1})-\hat{f}(x^*(y^*(z_{t+1}),z_t),y^*(z_{t+1});z_t)\\ &=\frac{p}{2}(z_{t+1}-z_t)^\top[z_{t+1}-z_t-2x^*(y^*(z_{t+1}),z_t)]. \end{aligned}\tag{4.25}$$

POTENTIAL FUNCTION We use the potential function $V_t=V(x_t,y_t,z_t)=\hat{f}(x_t,y_t;z_t)-2\Psi(y_t;z_t)+2P(z_t)$. By three descent steps above, we have

$$\begin{aligned}&\mathbb{E}V_t-\mathbb{E}V_{t+1}\\ \geq\ &\frac{\tau_1}{2}\mathbb{E}\|\nabla_x\hat{f}(x_t,y_t;z_t)\|^2-\left(1+\frac{l\tau_2}{2}\right)\tau_2\mathbb{E}\|\nabla_y f(x_{t+1},y_t)\|^2+\frac{p}{2\beta}\mathbb{E}\|z_t-z_{t+1}\|^2+\\ &2\tau_2\mathbb{E}\langle\nabla_y\hat{f}(x^*(y_t,z_t),y_t;z_t),\nabla_y f(x_{t+1},y_t)\rangle-L_\Psi\tau_2^2\mathbb{E}\|\nabla_y f(x_{t+1},y_t)\|^2+\\ &p\mathbb{E}(z_{t+1}-z_t)^\top[z_{t+1}+z_t-2x^*(y_{t+1},z_{t+1})]-p\mathbb{E}(z_{t+1}-z_t)^\top[z_{t+1}-z_t-2x^*(y^*(z_{t+1}),z_t)]-\\ &\frac{l}{2}\tau_2^2\sigma^2-\frac{p+l}{2}\tau_1^2\sigma^2-L_\Psi\tau_2^2\sigma^2\\ \geq\ &\frac{\tau_1}{2}\mathbb{E}\|\nabla_x\hat{f}(x_t,y_t;z_t)\|^2+\left(1-\frac{l\tau_2}{2}-L_\Psi\tau_2\right)\tau_2\mathbb{E}\|\nabla_y f(x_{t+1},y_t)\|^2+\frac{p}{2\beta}\mathbb{E}\|z_t-z_{t+1}\|^2+\\ &2\tau_2\mathbb{E}\langle\nabla_y\hat{f}(x^*(y_t,z_t),y_t;z_t)-\nabla_y f(x_{t+1},y_t),\nabla_y f(x_{t+1},y_t)\rangle+\\ &p\mathbb{E}(z_{t+1}-z_t)^\top[2x^*(y^*(z_{t+1}),z_t)-2x^*(y_{t+1},z_{t+1})]-\frac{l}{2}\tau_2^2\sigma^2-\frac{p+l}{2}\tau_1^2\sigma^2-L_\Psi\tau_2^2\sigma^2 \end{aligned}$$

$$\geq \frac{\tau_1}{2}\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + \frac{\tau_2}{2}\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 + \frac{p}{2\beta}\mathbb{E}\|z_t - z_{t+1}\|^2 +$$
$$2\tau_2\mathbb{E}\langle \nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t) - \nabla_y f(x_{t+1}, y_t), \nabla_y f(x_{t+1}, y_t)\rangle +$$
$$2p\mathbb{E}(z_{t+1} - z_t)^\top [x^*(y^*(z_{t+1}), z_t) - x^*(y_{t+1}, z_{t+1})] - \frac{l}{2}\tau_2^2\sigma^2 - \frac{p+l}{2}\tau_1^2\sigma^2 - L_\Psi\tau_2^2\sigma^2, \tag{4.26}$$

where in the last inequality we use $1 - \frac{l\tau_2}{2} - L_\Psi\tau_2 \geq \frac{1}{2}$ since $L_\Psi = 4l$ by our choice of τ_2 and p. Now we denote $A = 2\tau_2\langle \nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t) - \nabla_y f(x_{t+1}, y_t), \nabla_y f(x_{t+1}, y_t)\rangle$ and $B = 2p(z_{t+1} - z_t)^\top [x^*(y^*(z_{t+1}), z_t) - x^*(y_{t+1}, z_{t+1})]$. We note that

$$\begin{aligned} B &= 2p(z_{t+1} - z_t)^\top [x^*(y^*(z_{t+1}), z_t) - x^*(y^*(z_{t+1}), z_{t+1})] + \\ &\quad 2p(z_{t+1} - z_t)^\top [x^*(y^*(z_{t+1}), z_{t+1}) - x^*(y_{t+1}, z_{t+1})] \\ &\geq -2p\gamma_1\|z_{t+1} - z_t\|^2 + 2p(z_{t+1} - z_t)^\top [x^*(y^*(z_{t+1}), z_{t+1}) - x^*(y_{t+1}, z_{t+1})] \\ &\geq -\left(2p\gamma_1 + \frac{p}{6\beta}\right)\|z_{t+1} - z_t\|^2 - 6p\beta\|x^*(y^*(z_{t+1}), z_{t+1}) - x^*(y_{t+1}, z_{t+1})\|^2, \end{aligned} \tag{4.27}$$

where we use 4.6.6 in the first inequality. Also,

$$\begin{aligned} A &\geq -2\tau_2\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t) - \nabla_y f(x_{t+1}, y_t)\|\|\nabla_y f(x_{t+1}, y_t)\| \\ &\geq -2\tau_2 l\|x_{t+1} - x^*(y_t, z_t)\|\|\nabla_y f(x_{t+1}, y_t)\| \\ &\geq -\tau_2^2 l\nu\|\nabla_y f(x_{t+1}, y_t)\|^2 - l\nu^{-1}\|x_{t+1} - x^*(y_t, z_t)\|^2, \end{aligned} \tag{4.28}$$

where in the second inequality we use $\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t) = \nabla_y f(x^*(y_t, z_t), y_t)$ and in the third inequality $\nu > 0$ and we will choose it later. Taking expectation and applying Lemma 4.6.6

$$\mathbb{E}A \geq -\tau_2^2 l\nu\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 - l\tau_1^2\nu^{-1}\gamma_3^2\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 - 2l\nu^{-1}\tau_1^2\sigma^2. \tag{4.29}$$

Plugging (4.29) and (4.27) into (4.26),

$$\begin{aligned} &\mathbb{E}V_t - \mathbb{E}V_{t+1} \\ &\geq \left(\frac{\tau_1}{2} - l\tau_1^2\nu^{-1}\gamma_3^2\right)\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + \left(\frac{\tau_2}{2} - \tau_2^2 l\nu\right)\mathbb{E}\|\nabla_y f(x_{t+1}, y_t)\|^2 + \\ &\quad \left(\frac{p}{2\beta} - 2p\gamma_1 - \frac{p}{6\beta}\right)\mathbb{E}\|z_t - z_{t+1}\|^2 - 6p\beta\mathbb{E}\|x^*(y^*(z_{t+1}), z_{t+1}) - x^*(y_{t+1}, z_{t+1})\|^2 - \\ &\quad \left(\frac{p+l}{2} + 2l\nu^{-1}\right)\tau_1^2\sigma^2 - \left(\frac{l}{2} + L_\Psi\right)\tau_2^2\sigma^2, \end{aligned} \tag{4.30}$$

We rewrite $\|\nabla_y f(x_{t+1}, y_t)\|^2$ as:

$$\begin{aligned} \|\nabla_y f(x_{t+1}, y_t)\|^2 &= \|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t) + \nabla_y f(x_{t+1}, y_t) - \nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2 \\ &\geq \|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2/2 - \|\nabla_y f(x_{t+1}, y_t) - \nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2 \\ &\geq \|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2/2 - l^2\|x_{t+1} - x^*(y_t, z_t)\|^2. \end{aligned} \tag{4.31}$$

Taking expectation and applying Lemma 4.6.6

$$\mathbb{E}\|\nabla_y f(x_{t+1},y_t)\|^2 \geq \mathbb{E}\|\nabla_y \hat{f}(x^*(y_t,z_t),y_t;z_t)\|^2/2- \\ l^2\gamma_3^2\tau_1^2\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 - 2l^2\tau_1^2\sigma^2. \tag{4.32}$$

Note that $x^*(y^*(z_{t+1}),z_{t+1}) = x^*(z_{t+1})$. We rewrite $\|x^*(y^*(z_{t+1}),z_{t+1}) - x^*(y_{t+1},z_{t+1})\|^2$ as the following

$$\begin{aligned}
&\|x^*(z_{t+1}) - x^*(y_{t+1},z_{t+1})\|^2 \\
&\leq 4\|x^*(z_{t+1}) - x^*(z_t)\|^2 + 4\|x^*(z_t) - x^*(y_t^+(z_t),z_t)\|^2 + \\
&\quad 4\|x^*(y_t^+(z_t),z_t) - x^*(y_{t+1},z_t)\|^2 + 4\|x^*(y_{t+1},z_t) - x^*(y_{t+1},z_{t+1})\|^2 \\
&\leq 4\gamma_1^2\|z_{t+1}-z_t\|^2 + 4\|x^*(z_t) - x^*(y_t^+(z_t),z_t)\|^2 + 4\gamma_2^2\|y_t^+(z_t) - y_{t+1}\|^2 + 4\gamma_1^2\|z_t - z_{t+1}\|^2 \\
&\leq 4\|x^*(z_t) - x^*(y_t^+(z_t),z_t)\|^2 + 8\gamma_2^2\tau_2^2\|\nabla_y \hat{f}(x^*(y_t),z_t),y_t;z_t) - \nabla_y f(x_{t+1},y_t)\|^2 + \\
&\quad 8\gamma_2^2\tau_2^2\|\nabla_y f(x_{t+1},y_t) - G_y(x_{t+1},y_t,\xi_2^t)\|^2 + 8\gamma_1^2\|z_t - z_{t+1}\|^2 \\
&\leq 4\|x^*(z_t) - x^*(y_t^+(z_t),z_t)\|^2 + 8\gamma_2^2\tau_2^2 l^2\|x^*(y_t) - x_{t+1}\|^2 + \\
&\quad 8\gamma_2^2\tau_2^2\|\nabla_y f(x_{t+1},y_t) - G_y(x_{t+1},y_t,\xi_2^t)\|^2 + 8\gamma_1^2\|z_t - z_{t+1}\|^2,
\end{aligned}$$

where in the second and last inequality we use Lemma 4.6.6, and in the third inequality we use the definition of $y_t^+(z_t)$. Taking expectation and applying Lemma 4.6.6

$$\begin{aligned}
&\mathbb{E}\|x^*(z_{t+1}) - x^*(y_{t+1},z_{t+1})\|^2 \\
&\leq 8\gamma_1^2\mathbb{E}\|z_t - z_{t+1}\|^2 + 4\mathbb{E}\|x^*(z_t) - x^*(y_t^+(z_t),z_t)\|^2 + \\
&\quad 8\gamma_2^2\tau_2^2 l^2\gamma_3^2\tau_1^2\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + 16\gamma_2^2\tau_2^2 l^2\tau_1^2\sigma^2 + 8\gamma_2^2\tau_2^2\sigma^2.
\end{aligned} \tag{4.33}$$

Plugging (4.33) and (4.32) into (4.30), we have

$$\begin{aligned}
&\mathbb{E}V_t - \mathbb{E}V_{t+1} \\
&\geq \left[\frac{\tau_1}{2} - l\tau_1^2\nu^{-1}\gamma_3^2 - \left(\frac{\tau_2}{2} - \tau_2^2 l\nu\right) l^2\gamma_3^2\tau_1^2 - 48p\beta\gamma_2^2\tau_2^2 l^2\gamma_3^2\tau_1^2\right]\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 - \\
&\quad 24p\beta\mathbb{E}\|x^*(z_t) - x^*(y_t^+(z_t),z_t)\|^2 + \left(\frac{\tau_2}{4} - \frac{\tau_2^2 l\nu}{2}\right)\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t,z_t),y_t;z_t)\|^2 + \\
&\quad \left[\frac{p}{2\beta} - 2p\gamma_1 - \frac{p}{6\beta} - 48p\beta\gamma_1^2\right]\mathbb{E}\|z_t - z_{t+1}\|^2 - \\
&\quad \left[\frac{p+l}{2} + 2l\nu^{-1} + 96p\beta\gamma_2^2\tau_2^2 l^2 + 2l^2\left(\frac{\tau_2}{2} - \tau_2^2 l\nu\right)\right]\tau_1^2\sigma^2 - \left[\frac{l}{2} + L_\Psi + 48p\beta\gamma_2^2\right]\tau_2^2\sigma^2 \\
&\geq \frac{\tau_1}{4}\mathbb{E}\|\nabla_x \hat{f}(x_t,y_t;z_t)\|^2 + \frac{\tau_2}{8}\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t,z_t),y_t;z_t)\|^2 + \frac{p}{4\beta}\mathbb{E}\|z_t - z_{t+1}\|^2 - \\
&\quad 24p\beta\mathbb{E}\|x^*(z_t) - x^*(y_t^+(z_t),z_t)\|^2 - 2l\tau_1^2\sigma^2 - 5l\tau_2^2\sigma^2.
\end{aligned} \tag{4.34}$$

The last inequality above holds because by our choice of τ_1, τ_2, p and β, we have $\gamma_1 = 2$, $\gamma_2 = 3$ and $\gamma_3 = \frac{2}{\tau_1^2 l^2} + 2$, and therefore when we choose $\nu = \frac{1}{4l\tau_2} = \frac{12}{l\tau_1}$, we have $\frac{\tau_2}{4} - \frac{\tau_2^2 l\nu}{2} = \frac{\tau_2}{8}$ and

$$\begin{aligned}
&l\tau_1^2\nu^{-1}\gamma_3^2 + \left(\frac{\tau_2}{2} - \tau_2^2 l\nu\right) l^2\gamma_3^2\tau_1^2 + 48p\beta\gamma_2^2\tau_2^2 l^2\gamma_3^2\tau_1^2 \\
&= \left[\nu^{-1}(l\tau_1\gamma_3^2) - \frac{1}{\tau_1}\frac{\tau_2}{4}(l^2\tau_1^2\gamma_3^2) + 486l\beta\frac{\tau_2^2}{\tau_1}(l^2\tau_1^2\gamma_3^2)\right]\tau_1 \\
&\le \left[2\nu^{-1}\left(\frac{1}{\tau_1 l} + \tau_1 l\right) + \frac{1}{96}\left(1 + \tau_1^2 l^2\right) + \frac{486\times 2}{48\times 1600} l\mu\tau_2^2\left(1 + \tau_1^2 l^2\right)\right]\tau_1 \\
&\le \left[\frac{20}{9\nu}\frac{1}{\tau_1 l} + \frac{1}{96}\left(1 + \frac{1}{9}\right) + \frac{486\times 2}{48\times 1600}\left(1 + \frac{1}{9}\right) l\mu\tau_2^2\right]\tau_1 \le \frac{\tau_1}{4},
\end{aligned}$$

and

$$\frac{p+l}{2} + 2l\nu^{-1} + 96p\beta\gamma_2^2\tau_2^2 l^2 + 2l^2\left(\frac{\tau_2}{2} - \tau_2^2 l\nu\right) \le \left[\frac{3}{2} + \frac{\tau_1 l}{12} + \frac{96\times 2\times 9}{1600} l^2\mu\tau_2^3 + \frac{\tau_2 l}{2}\right] l \le 2l,$$

and

$$\frac{l}{2} + L_\Psi + 48p\beta\gamma_2^2 \le \left[\frac{1}{2} + 4 + 48\times 2\times 4\times 9\beta\right] l \le 5l,$$

and

$$\frac{p}{2\beta} - 2p\gamma_1 - \frac{p}{6\beta} - 48p\beta\gamma_1^2 \ge \left[\frac{1}{3} - 4\beta - 192\beta^2\right]\frac{p}{\beta} \ge \frac{p}{4\beta}.$$

STATIONARY MEASURE: First we note that

$$\begin{aligned}
\|\nabla_x f(x_t, y_t)\| &\le \|\nabla_x \hat{f}(x_t, y_t; z_t)\| + p\|x_t - z_t\| \\
&\le \|\nabla_x \hat{f}(x_t, y_t; z_t)\| + p\|x_t - x_{t+1}\| + p\|x_{t+1} - z_t\| \\
&\le \|\nabla_x \hat{f}(x_t, y_t; z_t)\| + p\tau_1\|\hat{G}_x(x_t, y_t, \xi_1^t; z_t)\| + p\|x_{t+1} - z_t\|.
\end{aligned}$$

Taking square and expectation

$$\begin{aligned}
&\mathbb{E}\|\nabla_x f(x_t, y_t)\|^2 \\
&\le 6\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 6p^2\tau_1^2\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 6p^2\mathbb{E}\|x_{t+1} - z_t\|^2 + 6p^2\tau_1^2\sigma^2 \\
&= 6(1 + p^2\tau_1^2)\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 6p^2\mathbb{E}\|x_{t+1} - z_t\|^2 + 6p^2\tau_1^2\sigma^2. \qquad (4.35)
\end{aligned}$$

Also,

$$\begin{aligned}
&\|\nabla_y f(x_t, y_t)\| \\
&\le \|\nabla_y f(x_{t+1}, y_t)\| + \|\nabla_y f(x_t, y_t) - \nabla_y f(x_{t+1}, y_t)\| \\
&\le \|\nabla_y f(x_{t+1}, y_t)\| + l\|x_{t+1} - x_t\| \\
&\le l\tau_1\|\hat{G}_x(x_t, y_t, \xi_1^t; z_t)\| + \|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\| + \|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t) - \nabla_y f(x_{t+1}, y_t)\| \\
&\le l\tau_1\|\hat{G}_x(x_t, y_t, \xi_1^t; z_t)\| + \|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\| + l\|x_{t+1} - x^*(y_t, z_t)\|.
\end{aligned}$$

Taking square and expectation, and applying Lemma 4.6.6

$$\begin{aligned}
&\mathbb{E}\|\nabla_y f(x_t, y_t)\|^2 \\
\leq\ & 6l^2\tau_1^2\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 6l^2\tau_1^2\sigma^2 + 6\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2 + \\
& 6l^2\gamma_3^2\tau_1^2\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 12l^2\tau_1^2\sigma^2 \\
\leq\ & 6l^2\tau_1^2(1+\gamma_3^2)\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 6\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2 + 18l^2\tau_1^2\sigma^2. \qquad (4.36)
\end{aligned}$$

Combining with (4.35),

$$\begin{aligned}
&\mathbb{E}\|\nabla_x f(x_t, y_t)\|^2 + \kappa\mathbb{E}\|\nabla_y f(x_t, y_t)\|^2 \\
\leq\ & 6(1 + p^2\tau_1^2 + \kappa l^2\tau_1^2 + \kappa l^2\gamma_3^2\tau_1^2)\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 6\kappa\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2 + \\
& 6p^2\mathbb{E}\|x_{t+1} - z_t\|^2 + (6p^2 + 18\kappa l^2)\tau_1^2\sigma^2 \\
\leq\ & 24\kappa\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + 6\kappa\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2 v + 6p^2\mathbb{E}\|x_{t+1} - z_t\|^2 + 42\kappa l^2\tau_1^2\sigma^2, \qquad (4.37)
\end{aligned}$$

where in the last inequality we use $6p^2 + 18\kappa l^2 = 24l^2 + 18\kappa l^2 \leq 42\kappa l^2$ and

$$\begin{aligned}
1 + p^2\tau_1^2 + kl^2\tau_1^2 + \kappa l^2\gamma_3^2\tau_1^2 =& 1 + 4l^2\tau_1^2 + \kappa l^2\tau_1^2 + 2\kappa(1 + \tau_1^2 l^2) \\
\leq& \frac{13}{9} + 2\kappa + 3\kappa l^2\tau_1^2 \leq 4\kappa.
\end{aligned}$$

PUTTING PIECES TOGETHER: From Lemma 4.6.7,

$$\begin{aligned}
24p\beta\|x^*(z) - x^*(y^+(z), z)\|^2 \leq& \frac{24p\beta}{(p-l)\mu}\left(1 + \tau_2 l + \frac{\tau_2 l(p+l)}{p-l}\right)^2 \|\nabla_y \hat{f}(x^*(y, z), y; z)\|^2 \\
\leq& \frac{1}{16}\tau_2\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2,
\end{aligned}$$

where in the second inequality we use

$$\frac{24p\beta}{(p-l)\mu}\left(1 + \tau_2 l + \frac{\tau_2 l(p+l)}{p-l}\right)^2 = \frac{48\beta}{\mu}(1 + \tau_2 l + 3\tau_2 l)^2 \leq \frac{96\beta}{\mu} \leq \frac{1}{16}\tau_2.$$

Plugging into (4.34),

$$\begin{aligned}
\mathbb{E}V_t - \mathbb{E}V_{t+1} \geq\ & \frac{\tau_1}{4}\mathbb{E}\|\nabla_x \hat{f}(x_t, y_t; z_t)\|^2 + \frac{\tau_2}{16}\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\|^2 + \\
& \frac{p\beta}{4}\mathbb{E}\|z_t - x_{t+1}\|^2 - 2l\tau_1^2\sigma^2 - 5l\tau_2^2\sigma^2.
\end{aligned}$$

Plugging into (4.37),

$$\begin{aligned}
&\mathbb{E}\|\nabla_x f(x_t, y_t)\|^2 + \kappa\mathbb{E}\|\nabla_y f(x_t, y_t)\|^2 \\
\leq\ & 24\kappa\mathbb{E}\|\nabla_x \hat{f}_x(x_t, y_t; z_t)\|^2 + 6\kappa\mathbb{E}\|\nabla_y \hat{f}(x^*(y_t, z_t), y_t; z_t)\| + 6p^2\mathbb{E}\|x_{t+1} - z_t\|^2 + 42\kappa l^2\tau_1^2\sigma^2 \\
\leq\ & \max\left\{\frac{96\kappa}{\tau_1}, \frac{96\kappa}{\tau_2}, \frac{24p}{\beta}\right\}\left[\mathbb{E}V_t - \mathbb{E}V_{t+1} + 2l\tau_1^2\sigma^2 + 5l\tau_2^2\sigma^2\right] + 42\kappa l^2\tau_1^2\sigma^2 \\
\leq\ & \frac{O(1)\kappa}{\tau_2}[\mathbb{E}V_t - \mathbb{E}V_{t+1}] + \frac{O(1)\kappa l\tau_1^2}{\tau_2}\sigma^2 + O(1)\kappa l\tau_2\sigma^2 + O(1)\kappa l^2\tau_1^2\sigma^2
\end{aligned}$$

$$\leq \frac{O(1)\kappa}{\tau_1}[\mathbb{E}V_t - \mathbb{E}V_{t+1}] + O(1)\kappa l\tau_1\sigma^2 + O(1)\kappa l^2\tau_1^2\sigma^2$$
$$\leq \frac{O(1)\kappa}{\tau_1}[\mathbb{E}V_t - \mathbb{E}V_{t+1}] + O(1)\kappa l\tau_1\sigma^2, \tag{4.38}$$

where in the second and fourth inequality we use $\tau_1 = 48\tau_2$ and $p/\beta = 3200\kappa/\tau_2$. Telescoping,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla_x f(x_t,y_t)\|^2 + \kappa\mathbb{E}\|\nabla_y f(x_t,y_t)\|^2 \leq \frac{O(1)\kappa}{T\tau_1}[V_0 - \min_{x,y,z} V(x,y,z)] + O(1)\kappa l\tau_1\sigma^2.$$

Since for any z we can find x, y such that $(\hat{f}(x,y;z) - \Psi(y;z)) + (P(z) - \Psi(y;z)) = 0$,

$$\begin{aligned} & V_0 - \min_{x,y,z} V(x,y,z) \\ = {} & P(z_0) + (\hat{f}(x_0,y_0;z_0) - \Psi(y_0;z_0)) + (P(z_0) - \Psi(y_0;z_0)) - \\ & \min_{x,y,z}[P(z) + (\hat{f}(x,y;z) - \Psi(y;z)) + (P(z) - \Psi(y;z))] \\ \leq {} & (P(z_0) - \min_z P(z)) + (\hat{f}(x_0,y_0;z_0) - \Psi(y_0;z_0)) + (P(z_0) - h(y_0;z_0)). \end{aligned}$$

Note that for any z

$$P(z) = \min_x \max_y f(x,y) + l\|x-z\|^2 = \min_x \Phi(x) + l\|x-z\|^2 = \Phi_{1/2l}(z) \leq \Phi(z),$$

and $P(z) = \Phi_{1/2l}(z)$ also implies $\min_z P(z) = \min_x \Phi(x)$. Hence

$$V_0 - \min_{x,y,z} V(x,y,z) \leq (\Phi(z_0) - \min_x \Phi(x)) + (\hat{f}(x_0,y_0;z_0) - \Psi(y_0;z_0)) + (P(z_0) - \Psi(y_0;z_0)). \tag{4.39}$$

With $b = (\hat{f}(x_0,y_0;z_0) - \Psi(y_0;z_0)) + (P(z_0) - \Psi(y_0;z_0))$, we write

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla_x f(x_t,y_t)\|^2 + \kappa\mathbb{E}\|\nabla_y f(x_t,y_t)\|^2 \leq \frac{O(1)\kappa}{T\tau_1}[\Delta + b] + O(1)\kappa l\tau_1\sigma^2.$$

with $\Delta = \Phi(z_0) - \Phi^*$. Picking $\tau_1 = \min\left\{\frac{\sqrt{\Phi(x_0)-\Phi^*}}{2\sigma\sqrt{Tl}}, \frac{1}{3l}\right\}$,

$$\begin{aligned} & \frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla_x f(x_t,y_t)\|^2 + \kappa\mathbb{E}\|\nabla_y f(x_t,y_t)\|^2 \\ \leq {} & \max\left\{\frac{2\sigma\sqrt{Tl}}{\sqrt{\Delta}}, 3l\right\}\frac{O(1)\kappa}{T}[\Phi(z_0) - \Phi^* + b] + \frac{O(1)\sqrt{\Delta}}{2\sigma\sqrt{Tl}}\cdot\kappa l\tau_1\sigma^2 \\ \leq {} & \frac{O(1)\kappa}{T}[\Delta + b] + \frac{O(1)\kappa\sqrt{l}b}{\sqrt{\Delta T}}\sigma + \frac{O(1)\kappa\sqrt{l\Delta}}{\sqrt{T}}\sigma. \end{aligned}$$

We reach our conclusion by noting that $b \leq 2\,\mathrm{gap}_{\hat{f}(\cdot,\cdot;z_0)}(x_t,y_t)$.

□

4.6.4 *Catalyst-AGDA*

Algorithm 8 Catalyst-AGDA

Input: (x_0, y_0), step sizes $\tau_1 > 0, \tau_2 > 0$.
for all $t = 0, 1, 2, ..., T-1$ **do**
  Let $k = 0$ and $x_0^0 = x_0$.
  repeat
    $y_{k+1}^t = y_k^t + \tau_2 \nabla_y f(x_k^t, y_k^t)$
    $x_{k+1}^t = x_k^t - \tau_1 [\nabla_x f(x_k^t, y_{k+1}^t) + 2l(x_k^t - x_0^t)]$
    $k = k + 1$
  until $\text{gap}_{\hat{f}_t}(x_k^t, y_k^t) \le \beta \, \text{gap}_{\hat{f}_t}(x_0^t, y_0^t)$ where $\hat{f}_t(x, y) \triangleq f(x, y) + l\|x - x_0^t\|^2$
  $x_0^{t+1} = x_{k+1}^t, \quad y_0^{t+1} = y_{k+1}^t$
end for
Output: $\tilde{x}_T$, which is uniformly sampled from $x_0^1, ..., x_0^T$

Now we present a new algorithm, called Catalyst-AGDA, in Algorithm 8. It iteratively solves an augmented auxiliary problem similar to Smoothed-AGDA:

$$\hat{f}_t(x, y) \triangleq f(x, y) + l\|x - x_0^t\|^2,$$

by AGDA with y update first[5]. The stopping criterion for the inner-loop is

$$\text{gap}_{\hat{f}_t}(x_k^t, y_k^t) \le \beta \, \text{gap}_{\hat{f}_t}(x_0^t, y_0^t),$$

and we will specify β later. For Catalyst-AGDA, we only consider the deterministic case, in which we have the exact gradient of $f(\cdot, \cdot)$.

In this subchapter, we use (x^t, y^t) as a shorthand for (x_0^t, y_0^t). We denote $(\hat{x}^t, \hat{y}^t)$ with $\hat{y}^t \in \hat{Y}^t$ as the optimal solution to the auxiliary problem at t-th iteration: $\min_{x \in \mathbb{R}^{d_1}} \max_{y \in \mathbb{R}^{d_2}} \left[\hat{f}_t(x, y) \triangleq f(x, y) + l\|x - x^t\|^2\right]$. Define $\hat{\Phi}_t(x) = \max_y f(x, y) + l\|x - x^t\|^2$. We use $Y^*(x)$ to denote the set $\text{Argmax}_y f(x, y)$. In the following lemma, we show the convergence of the Moreau envelop $\|\nabla \Phi_{1/2l}(x)\|^2$ when we choose β appropriately in the stopping criterion of the AGDA subroutine.

Lemma 4.6.8. *Under Assumptions 10 and 11, define $\Delta = \Phi(x_0) - \Phi^*$, if we apply Catalyst-AGDA with $\beta = \frac{\mu^2}{4l^2}$ in the stopping criterion of the inner-loop, then we have*

$$\sum_{t=0}^{T-1} \|\nabla \Phi_{1/2l}(x^t)\|^2 \le \frac{35l}{2}\Delta + 3la_0,$$

where $a_0 := \Phi(x_0) - f(x_0, y_0)$.

5 We believe that updating x first in the subroutine will lead to the same convergence property. For simplicity, we update y first so that we can directly apply Theorem 4.6.4.

Proof. Define $g_{t+1} = \text{gap}_{\hat{f}_t}(x^{t+1}, y^{t+1})$. It is easy to observe that $\hat{x}^t = \text{prox}_{\Phi/2l}(x^t)$. Define $\hat{\Phi}_t(x) = \max_y f(x,y) + l\|x - x^t\|^2$. By Lemma 4.3 in [Drusvyatskiy and Paquette, 2019],

$$\begin{aligned}\|\nabla\Phi_{1/2l}(x^t)\|^2 = 4l^2\|x^t - \hat{x}^t\|^2 &\le 8l[\hat{\Phi}_t(x^t) - \hat{\Phi}_t(\text{prox}_{\Phi/2l}(x^t))] \\ &\le 8l[\hat{\Phi}_t(x^t) - \hat{\Phi}_t(x^{t+1}) + b_{t+1}] \\ &= 8l\left\{\Phi(x^t) - \left[\Phi(x^{t+1}) + l\|x^{t+1} - x^t\|^2\right] + b_{t+1}\right\} \\ &\le 8l[\Phi(x^t) - \Phi(x^{t+1}) + g_{t+1}], \end{aligned} \tag{4.40}$$

where in the first inequality we use l-strongly convexity of $\hat{\Phi}_t$. Because $\hat{f}$ is $3l$-smooth, l-strongly convex in x and μ-PL in y, its primal and dual function are $18l\kappa$ and $18l$ smooth, respectively, by Lemma 4.6.3. Then we have

$$\begin{aligned}\text{gap}_{\hat{f}_t}(x^t, y^t) &= \max_y \hat{f}_t(x^t, y) - \min_x \max_y \hat{f}_t(x,y) + \min_x \max_y \hat{f}_t(x,y) - \min_x \hat{f}_t(x, y_t) \\ &\le 9l\kappa\|x^t - \hat{x}^t\|^2 + 9l\|y^t - \hat{y}^t\|^2, \end{aligned} \tag{4.41}$$

for all $\hat{y}^t \in \hat{Y}^t$. For $t \ge 1$, by fixing $\hat{y}^{t-1}$ to be the projection of y^t to $\hat{Y}^{t-1}$, there exists $\hat{y}^t \in \hat{Y}^t$ so that

$$\begin{aligned}\|y^t - \hat{y}^t\|^2 \le& 2\|y^t - \hat{y}^{t-1}\|^2 + 2\|y^*(\hat{x}^{t-1}) - y^*(\hat{x}^t)\|^2 \\ &\le 2\|y^t - \hat{y}^{t-1}\|^2 + 2\left(\frac{l}{\mu}\right)^2\|\hat{x}^t - \hat{x}^{t-1}\|^2 \\ &\le 2\|y^t - \hat{y}^{t-1}\|^2 + 4\left(\frac{l}{\mu}\right)^2\|\hat{x}^t - x^t\|^2 + 4\left(\frac{l}{\mu}\right)^2\|x^t - \hat{x}^{t-1}\|^2 \\ &\le \frac{8l}{\mu^2}g_t + 4\left(\frac{l}{\mu}\right)^2\|\hat{x}^t - x^t\|^2, \end{aligned}$$

where we use Lemma 4.6.3 in the second inequality, and strong-convexity and PL condition in the last inequality. By our stopping criterion and $\|\nabla\Phi_{1/2l}(x^t)\|^2 = 4l^2\|x^t - \hat{x}^t\|^2$, for $t \ge 1$

$$g_{t+1} \le \beta\,\text{gap}_{\hat{f}_t}(x^t, y^t) \le 9l\kappa\beta\|x^t - \hat{x}^t\|^2 + 9l\beta\|y^t - \hat{y}^t\|^2 \le 72\kappa^2\beta g_t + \frac{12\kappa^2\beta}{l}\|\nabla\Phi_{1/2l}(x^t)\|^2. \tag{4.42}$$

For $t = 0$, by fixing $y^*(x^0)$ to be the projection of y^0 to $Y^*(x^0)$,

$$\|y^0 - \hat{y}^0\|^2 \le 2\|y^0 - y^*(x^0)\|^2 + 2\|\hat{y}^0 - y^*(x^0)\|^2 \le \frac{4}{\mu}a_0 + 2\kappa^2\|x^0 - \hat{x}^0\|^2. \tag{4.43}$$

Because $\Phi(x) + l\|x - x^0\|^2$ is l-strongly convex, we have

$$\left(\Phi(\hat{x}^0) + l\|\hat{x}^0 - x^0\|^2\right) + \frac{l}{2}\|\hat{x}^0 - x^0\|^2 \le \Phi(x^0) = \Phi^* + (\Phi(x^0) - \Phi^*) \le \Phi(\hat{x}^0) + (\Phi(x^0) - \Phi^*).$$

This implies $\|\hat{x}^0 - x^0\|^2 \le \frac{2}{3l}(\Phi(x^0) - \Phi^*)$. Hence, by the stopping criterion,

$$g_1 \le \beta\,\text{gap}_{\hat{f}_0}(x^0, y^0) \le 9l\kappa\beta\|x^0 - \hat{x}^0\|^2 + 9l\beta\|y^0 - \hat{y}^0\|^2 \le 18\kappa^2\beta\Delta + 36\kappa\beta a_0. \tag{4.44}$$

Recursing (4.42) and (4.44), we have for $t \geq 1$

$$g_{t+1} \leq (72\kappa^2\beta)^t g_1 + \frac{12\kappa^2\beta}{l}\sum_{k=1}^{t}(72\kappa^2\beta)^{t-k}\|\nabla\Phi_{1/2l}(x_k)\|^2$$
$$\leq 18\kappa^2\beta(72\kappa^2\beta)^t\Delta + 36\kappa\beta(72\kappa^2\beta)^t a_0 + \frac{12\kappa^2\beta}{l}\sum_{k=1}^{t}(72\kappa^2\beta)^{t-k}\|\nabla\Phi_{1/2l}(x_k)\|^2.$$

Summing from $t = 0$ to $T-1$,

$$\sum_{t=0}^{T-1} g_{t+1} = \sum_{t=1}^{T-1} g_t + g_1$$
$$\leq 18\kappa^2\beta\sum_{t=0}^{T-1}(72\kappa^2\beta)^t\Delta + 36\kappa\beta\sum_{t=0}^{T-1}(72\kappa^2\beta)^t a_0 + \frac{12\kappa^2\beta}{l}\sum_{t=1}^{T-1}\sum_{k=1}^{t}(72\kappa^2\beta)^{t-k}\|\nabla\Phi_{1/2l}(x_k)\|^2$$
$$\leq \frac{18\kappa^2\beta}{1-72\kappa^2\beta}\Delta + \frac{36\kappa\beta}{1-72\kappa^2\beta}a_0 + \frac{12\kappa^2\beta}{l(1-72\kappa^2\beta)}\sum_{t=1}^{T-1}\|\nabla\Phi_{1/2l}(x^t)\|^2, \tag{4.45}$$

where in the last inequality we use

$$\sum_{t=1}^{T-1}\sum_{k=1}^{t}(72\kappa^2\beta)^{t-k}\|\nabla\Phi_{1/2l}(x_k)\|^2 = \sum_{k=1}^{T-1}\sum_{t=k}^{T}(72\kappa^2\beta)^{t-k}\|\nabla\Phi_{1/2l}(x_k)\|^2$$
$$\leq \sum_{k=1}^{T-1}\frac{1}{1-(72\kappa^2\beta)}\|\nabla\Phi_{1/2l}(x_k)\|^2.$$

Now, by telescoping (4.40),

$$\frac{1}{8l}\sum_{t=0}^{T-1}\|\nabla\Phi_{1/2l}(x^t)\|^2 \leq \Phi(x^0) - \Phi^* + \sum_{t=0}^{T-1} g_{t+1}.$$

Plugging (4.45) in,

$$\left(\frac{1}{8l} - \frac{12\kappa^2\beta}{l(1-72\kappa^2\beta)}\right)\sum_{t=0}^{T-1}\|\nabla\Phi_{1/2l}(x^t)\|^2 \leq \left(1 + \frac{18\kappa^2\beta}{1-72\kappa^2\beta}\right)\Delta + \frac{36\kappa\beta}{1-72\kappa^2\beta}a_0. \tag{4.46}$$

With $\beta = \frac{1}{264\kappa^4}$, we have $\frac{\kappa^2\beta}{1-72\kappa^2\beta} \leq \frac{1}{192\kappa^2}$. Therefore,

$$\sum_{t=0}^{T-1}\|\nabla\Phi_{1/2l}(x^t)\|^2 \leq \frac{35l}{2}\Delta + 3la_0.$$

□

Theorem 4.6.9. *Under Assumptions 10 and 11, if we apply Catalyst-AGDA with $\beta = \frac{1}{264\kappa^4}$ in the stopping criterion of the inner-loop, then the output from Algorithm 8 satisfies*

$$\sum_{t=1}^{T}\|\nabla\Phi(x_0^t)\|^2 \leq \frac{1}{T}\sum_{t=1}^{T}\|\nabla\Phi(x^{t+1})\|^2 \leq \frac{19l}{T}\Delta + \frac{6l}{T}a_0 \tag{4.47}$$

which implies the outer-loop complexity of $O(l\Delta\epsilon^{-2})$. Furthermore, if we choose $\tau_1 = \frac{1}{3l}$ and $\tau_2 = \frac{1}{486l}$, it takes $K = O(\kappa\log(\kappa))$ inner-loop iterations to satisfy the stopping criterion. Therefore, the total complexity is $O(\kappa l\Delta\epsilon^{-2}\log\kappa)$.

Proof. We separate the proof into two parts: 1) outer-loop complexity 2) inner-loop convergence rate.

Outer-loop: We still denote $g_{t+1} = \text{gap}_{\hat{f}_t}(x^{t+1}, y^{t+1})$. First, note that

$$\begin{aligned}\|\nabla\Phi(x^{t+1})\|^2 &\le 2\|\nabla\Phi(x^{t+1}) - \nabla\Phi(\hat{x}^t)\|^2 + 2\|\nabla\Phi(\hat{x}^t)\|^2 \\ &\le 2\left(\frac{2l^2}{\mu}\right)\|x^{t+1} - \hat{x}^t\|^2 + 2\|\nabla\Phi_{1/2l}(x^t)\|^2 \\ &\le \frac{16l^3}{\mu^2} g_{t+1} + 2\|\nabla\Phi_{1/2l}(x^t)\|^2. \end{aligned} \tag{4.48}$$

where in the second inequality we use Lemma 4.6.1 and Lemma 4.3 in [Drusvyatskiy and Paquette, 2019]. Summing from $t = 0$ to $T-1$, we have

$$\sum_{t=0}^{T-1}\|\nabla\Phi(x^{t+1})\|^2 \le \frac{16l^3}{\mu^2}\sum_{t=0}^{T-1} g_{t+1} + 2\sum_{t=0}^{T-1}\|\nabla\Phi_{1/2l}(x^t)\|^2. \tag{4.49}$$

Applying (4.45), we have

$$\begin{aligned}\sum_{t=0}^{T-1}\|\nabla\Phi(x^{t+1})\|^2 \le \left[\frac{16l^3}{\mu^2}\cdot\frac{12\kappa^2\beta}{l(1-72\kappa^2\beta)} + 2\right]\sum_{t=1}^{T-1}\|\nabla\Phi_{1/2l}(x^t)\|^2 + \\ \frac{16l^3}{\mu^2}\cdot\frac{18\kappa^2\beta}{1-72\kappa^2\beta}\Delta + \frac{16l^3}{\mu^2}\cdot\frac{36\kappa\beta}{1-72\kappa^2\beta}a_0,\end{aligned}$$

With $\beta = \frac{1}{264\kappa^4}$, we have

$$\sum_{t=0}^{T-1}\|\nabla\Phi(x^{t+1})\|^2 \le 3\sum_{t=1}^{T-1}\|\nabla\Phi_{1/2l}(x^t)\|^2 + \frac{3l}{2}\Delta + 3la_0.$$

Applying Lemma 4.6.8,

$$\frac{1}{T}\sum_{t=1}^{T}\|\nabla\Phi(x^{t+1})\|^2 \le \frac{19l}{T}\Delta + \frac{6l}{T}a_0.$$

Inner-loop: The objective of auxiliary problem $\min_x\max_y \hat{f}_t(x,y) \triangleq f(x,y) + l\|x - x_0^t\|^2$ is $3l$-smooth and (l,μ)-SC-PL. We denote the dual function of the auxiliary problem by $\hat{\Psi}^t(y) = \min_x \hat{f}_t(x,y)$. We also define

$$P_k^t \triangleq \left[\max_y \hat{\Psi}^t(y) - \hat{\Psi}^t(y_k^t)\right] + \frac{1}{10}\left[\hat{f}_t(x_k^t, y_k^t) - \hat{\Psi}^t(y_k^t)\right].$$

By Theorem 4.6.4, AGDA with stepsizes $\tau_1 = \frac{1}{3l}$ and $\tau_2 = \frac{l^2}{18(3l)^3} = \frac{1}{486l}$ satisfies

$$P_k^t \le \left(1 - \frac{\mu}{972l}\right)^k P_0^t.$$

We denote $x_*^t(y) = \text{argmin}_x \hat{f}_t(x,y)$. We note that

$$\begin{aligned}\|x_k^t - \hat{x}^t\|^2 =& 2\|x_k^t - x_*^t(y_k^t)\|^2 + 2\|x_*^t(y_k^t) - \hat{x}^t\|^2 \\ =& 2\|x_k^t - x_*^t(y_k^t)\|^2 + 2\|x_*^t(y_k^t) - x_*^t(\hat{y}^t)\|^2\end{aligned}$$

$$\begin{aligned}
&\leq \frac{4}{l}\left[\hat{f}_t(x_k^t, y_k^t) - \hat{\Psi}^t(y_k^t)\right] + 2\left(\frac{3l}{\mu}\right)^2 \|y_k^t - \hat{y}^t\|^2 \\
&\leq \frac{4}{l}\left[\hat{f}_t(x_k^t, y_k^t) - \hat{\Psi}^t(y_k^t)\right] + \frac{36l^2}{\mu^3}[\hat{\Psi}^t(\hat{y}^t) - \hat{\Psi}^t(y_k^t)] \\
&\leq \left(\frac{40}{l} + \frac{36l^2}{\mu^3}\right)\left(1 - \frac{\mu}{972l}\right)^k P_0^t, \qquad (4.50)
\end{aligned}$$

where in the first inequality we use l-strong convexity of $\hat{f}_t(\cdot, y_k^t)$ and Lemma 4.6.1, and in the second inequality we use μ-PL of $\hat{\Psi}^t$ and Lemma 4.6.2. Since $\hat{\Phi}^t$ is smooth by Lemma 4.6.3,

$$\hat{\Phi}^t(x_k^t) - \hat{\Phi}^t(\hat{x}^t) \leq \frac{2(3l)^2}{2\mu}\|x_k^t - \hat{x}^t\|^2 \leq \frac{9l^2}{\mu}\left(\frac{40}{l} + \frac{36l^2}{\mu^3}\right)\left(1 - \frac{\mu}{972l}\right)^k P_0^t. \qquad (4.51)$$

Therefore,

$$\begin{aligned}
\mathrm{gap}_{\hat{f}_t}(x_k^t, y_k^t) = \hat{\Phi}^t(x_k^t) - \hat{\Phi}^t(\hat{x}^t) + \hat{\Psi}^t(\hat{y}^t) - \hat{\Psi}^t(y_k^t) &\leq \left[\frac{9l^2}{\mu}\left(\frac{40}{l} + \frac{36l^2}{\mu^3}\right) + 1\right]\left(1 - \frac{\mu}{972l}\right)^k P_0^t \\
&\leq 754\kappa^4\left(1 - \frac{1}{972\kappa}\right)^k \mathrm{gap}_{\hat{f}_t}(x_0^t, y_0^t).
\end{aligned}$$

where in the last inequality we note that $P_0^t \leq \frac{11}{10}\,\mathrm{gap}_{\hat{f}_t}(x_0^t, y_0^t)$. So after $K = O(\kappa \log(\kappa))$ iterations of AGDA, the stopping criterion $\mathrm{gap}_{\hat{f}_t}(x_k^t, y_k^t) \leq \beta\,\mathrm{gap}_{\hat{f}_t}(x_0^t, y_0^t)$ can be satisfied. □

Remark 4.6.10. *The theorem above implies that Catalyst-AGDA can achieve the complexity of $\tilde{O}(\kappa l \Delta \epsilon^{-2})$ in the deterministic setting, which is comparable to the complexity of Smoothed-AGDA up to a logarithmic term in κ.*

5

PARAMETER-AGNOSTIC NONCONVEX MINIMAX OPTIMIZATION

Adaptive algorithms like AdaGrad and AMSGrad are successful in nonconvex optimization owing to their *parameter-agnostic* ability – requiring no a priori knowledge about problem-specific parameters nor tuning of learning rates. However, when it comes to nonconvex minimax optimization, direct extensions of such adaptive optimizers without proper *time-scale separation* may fail to work in practice. We provide such an example proving that the simple combination of Gradient Descent Ascent (GDA) with adaptive stepsizes can diverge if the primal-dual stepsize ratio is not carefully chosen; hence, a fortiori, such adaptive extensions are not parameter-agnostic. To address the issue, we formally introduce a Nested Adaptive framework, NeAda for short, that carries an inner loop for adaptively maximizing the dual variable with controllable stopping criteria and an outer loop for adaptively minimizing the primal variable. Such a mechanism can be equipped with off-the-shelf adaptive optimizers and automatically balance the progress in the primal and dual variables. Theoretically, for nonconvex-strongly-concave minimax problems, we show that NeAda with AdaGrad stepsizes can achieve the near-optimal $\widetilde{O}(\epsilon^{-2})$ and $\widetilde{O}(\epsilon^{-4})$ gradient complexities respectively in the deterministic and stochastic settings, *without* prior information on the problem's smoothness and strong concavity parameters. To the best of our knowledge, this is the first algorithm that simultaneously achieves near-optimal convergence rates and parameter-agnostic adaptation in the nonconvex minimax setting.

5.1 OVERVIEW

Adaptive gradient methods, whose stepsizes and search directions are adjusted based on past gradients, have received phenomenal popularity and are proven successful in a variety of large-scale machine learning applications. Prominent examples include AdaGrad [Duchi et al., 2011], RMSProp [Hinton et al., 2012], AdaDelta [Zeiler, 2012], Adam [Kingma and Ba, 2015], and AMSGrad [Reddi et al., 2019], just to name a few. Their empirical success is especially pronounced for nonconvex optimization such as training deep neural networks. Besides improved performance, being *parameter-agnostic* is another important trait of adaptive methods. Unlike (stochastic) gradient descent, adaptive methods often do not require a priori knowledge about problem-specific parameters (such as Lipschitz

constants, smoothness, etc.).[1] On the theoretical front, some adaptive methods can achieve nearly the same convergence guarantees as (stochastic) gradient descent [Duchi et al., 2011, Ward et al., 2019, Reddi et al., 2019].

Recently, adaptive methods have sprung up for minimax optimization:

$$\min_{x\in\mathbb{R}^d}\max_{y\in\mathcal{Y}} f(x,y) \triangleq \mathbb{E}[F(x,y;\xi)], \tag{5.1}$$

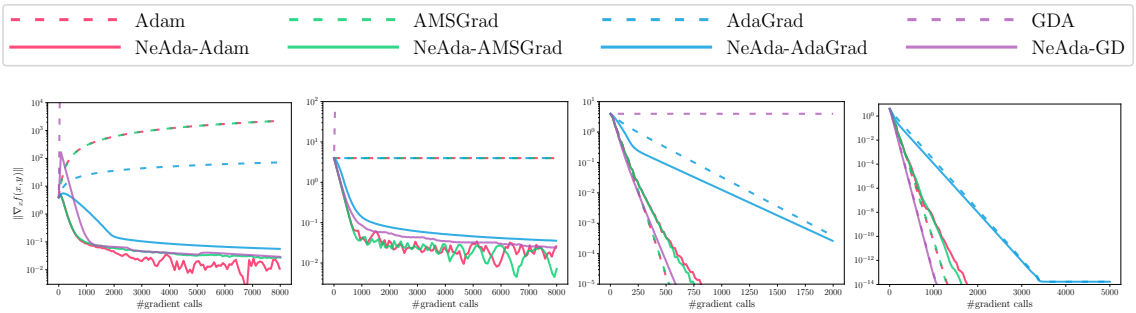
where f is l-Lipschitz smooth jointly in x and y, $\mathcal{Y}$ is closed and convex, and ξ is a random vector. A common practice is to simply combine adaptive stepsizes with popular minimax optimization algorithms such as Gradient Descent Ascent (GDA), extragradient method (EG) and the like; see e.g., [Gidel et al., 2018, Gulrajani et al., 2017, Goodfellow, 2016]. It is worth noting that these methods are reported successful in some applications yet at other times can suffer from training instability. In recent years, theoretical behaviors of such adaptive methods are extensively studied for convex-concave minimax optimization; see e.g., [Bach and Levy, 2019, Antonakopoulos et al., 2019, Antonakopoulos, 2021, Ene and Nguyen, 2020, Stonyakin et al., 2018, Gasnikov et al., 2019, Malitsky, 2020, Diakonikolas, 2020]. However, for minimax optimization in the important nonconvex regime, little theory related to adaptive methods is known.

Unlike the convex-concave setting, a key challenge for nonconvex minimax optimization lies in the necessity of a *problem-specific time-scale separation* of the learning rates between the min-player and max-player when GDA or EG methods are applied, as proven in [Yang et al., 2022b, Lin et al., 2020a, Sebbouh et al., 2022, Boţ and Böhm, 2020]. This makes the design of adaptive methods fundamentally different from and more challenging than nonconvex minimization. Several recent attempts [Guo et al., 2021a, Huang and Huang, 2021, Huang et al., 2021] studied adaptive methods for nonconvex-strongly-concave minimax problems; yet, they all require explicit knowledge of the problems' smoothness and strong concavity parameters to maintain a stepsize ratio proportional to the condition number. Such a requirement evidently undermines the parameter-agnostic trait of adaptive methods. This raises a two interesting questions: (1) *Without a problem-dependent stepsize ratio, does simple combination of GDA and adaptive stepsizes still converge?* (2) *Can we design an adaptive algorithm for nonconvex minimax optimization that is truly parameter-agnostic and provably convergent?*

In this chapter, we address these questions and make the following key contributions:

- We investigate two generic frameworks for adaptive minimax optimization: one is a simple (non-nested) adaptive framework, which performs one step of update of x and y simultaneously with adaptive gradients; the other is Nested Adaptive (NeAda) framework, which performs multiple updates of y after one update of x, each with

1 For distinction, we use "parameter-agnostic" to describe algorithms that do not ask for problem-specific parameters in setting their stepsizes or hyperparameters; we refer to "adaptive algorithms" as methods whose stepsizes are based on the previously observed gradients.



(a) $r = 1$ (b) $r = 2$ (c) $r = 4$ (d) $r = 8$

FIGURE 5.1: Comparison between the two families of non-nested and nested adaptive methods on function $f(x, y) = -\frac{1}{2}y^2 + 2xy - 2x^2$ with deterministic gradient oracles. $r = \eta^y / \eta^x$ is a pre-fixed learning rate ratio.

adaptive gradients. Both frameworks allow flexible choices of adaptive mechanisms such as Adam, AMSGrad and AdaGrad. We provide an example proving that the simple adaptive framework can fail to converge without setting an appropriate stepsize ratio; this applies to any of the adaptive mechanisms mentioned above, even in the noiseless setting. In contrast, the NeAda framework is less sensitive to the stepsize ratio, as numerically illustrated in Figure 5.1.

- We provide the convergence analysis for a representative of NeAda that uses AdaGrad stepsizes for x and a convergent adaptive optimizer for y, in terms of nonconvex-strongly-concave minimax problems. Notably, the convergence of this general scheme does not require to know any problem parameters and does not assume the bounded gradients. We demonstrate that NeAda is able to achieve $\widetilde{O}(\epsilon^{-2})$ oracle complexity for the deterministic setting and $\widetilde{O}(\epsilon^{-4})$ for the stochastic setting to converge to ϵ-stationary point, matching best known bounds. To the best of our knowledge, this seems to be the first adaptive framework for nonconvex minimax optimization that is provably convergent and parameter-agnostic.

- We further make two complementary contributions, which can be of independent interest. First, we propose a general AdaGrad-type stepsize for strongly-convex problems without knowing the strong convexity parameters, and derive a convergence rate comparable to SGD. It can serve as a subroutine for NeAda. Second, we provide a high probability convergence result for the primal variable of NeAda under a subGaussian assumption.

- Finally, we numerically validate the robustness of the NeAda framework on several test functions compared to the non-nested adaptive framework, and demonstrate the

effectiveness of the NeAda framework on distributionally robust optimization task with a real dataset.

5.1.1 *Related work*

ADAPTIVE ALGORITHMS. Duchi et al. [2011] introduce AdaGrad for convex online learning and achieve $O(\sqrt{T})$ regrets. Li and Orabona [2019] and Ward et al. [2019] show an $\widetilde{O}(\epsilon^{-4})$ complexity for AdaGrad in the nonconvex stochastic optimization. There are an extensive number of works on AdaGrad-type methods; to list a few, [Levy et al., 2018, Antonakopoulos and Mertikopoulos, 2021, Kavis et al., 2019, Orabona and Pál, 2018]. Another family of algorithms uses more aggressive stepsizes of exponential moving average of the past gradients, such as Adam [Kingma and Ba, 2015] and RMSProp [Hinton et al., 2012]. Reddi et al. [2019] point out the non-convergence of Adam and provide a remedy with non-increasing stepsizes. There is a surge in the study of Adam-type algorithms due to their popularity in the deep neural network training [Zaheer et al., 2018, Chen et al., 2019, Liu et al., 2020c]. Some work provides the convergence results for adaptive methods in the strongly-convex optimization [Wang et al., 2020a, Levy, 2017, Mukkamala and Hein, 2017]. Line search and stochastic line search are another effective strategy that can detect the objective's curvature and have received much attention [Vaswani et al., 2019, 2021, 2020]. Notably, many adaptive algorithms are parameter-agnostic [Duchi et al., 2011, Reddi et al., 2019, Ward et al., 2019].

ADAPTIVE ALGORITHMS IN MINIMAX OPTIMIZATION. There exist many adaptive and parameter-agnostic methods designed for convex-concave minimax optimization as a special case of monotone variational inequality [Bach and Levy, 2019, Antonakopoulos et al., 2019, Antonakopoulos, 2021, Ene and Nguyen, 2020, Stonyakin et al., 2018, Gasnikov et al., 2019, Malitsky, 2020, Diakonikolas, 2020]. Most of them combine extragradient method, mirror prox [Nemirovski, 2004] or the like, with AdaGrad mechanism. Liu et al. [2019] and Dou and Li [2021] relax the convexity-concavity assumption to the regime where Minty variational inequality (MVI) has a solution. In these settings, time-scale separation of learning rates is not required even for non-adaptive algorithms. For nonconvex-strongly-concave problems, Huang and Huang [2021], Huang et al. [2021], Guo et al. [2021a] propose adaptive methods, which set the learning rates based on knowledge about smoothness and strong-concavity modulus and the bounds for adaptive stepsizes.

5.2 NON-NESTED AND NESTED ADAPTIVE METHODS

In this subchapter, we investigate two generic frameworks that can incorporate most existing adaptive methods into minimax optimization. We remark that many variants encapsulated in these two families are already widely used in practice, such as training of GAN [Goodfellow, 2016], distributionally robust optimization [Sinha et al., 2018], etc. These two frameworks, coined as non-nested and nested adaptive methods, can be viewed as adaptive counterparts of GDA and GDmax. We aim to illustrate the difference between these two adaptive families, even though GDA and GDmax are often considered "twins".

NON-NESTED ADAPTIVE METHODS. In Algorithm 9, non-nested methods update the primal and dual variables in a symmetric way. Weighted gradients m_t^x and m_t^y are the moving average of the past stochastic gradients with the momentum parameters β^x and β^y. The effective stepsizes of x and y are $\eta^x/\sqrt{v_t^x}$ and $\eta^y/\sqrt{v_t^y}$, where the division is taken coordinate-wise. We refer to η^x and η^y as learning rates, and v_t^x, v_t^y are some average of squared-past gradients through function ψ. Many popular choices of adaptive stepsizes are captured in this framework, see also [Reddi et al., 2019]:

$$\text{(GDA)} \quad \beta = 0; \;\; \psi\left(v_0, \{g_i^2\}_{i=0}^t\right) = 1, \quad \text{(AdaGrad)} \quad \beta = 0; \;\; \psi\left(v_0, \{g_i^2\}_{i=0}^t\right) = v_0 + \sum_{i=0}^t g_i^2,$$

$$\text{(Adam)} \quad \psi\left(v_0, \{g_i^2\}_{i=0}^t\right) = \gamma^{t+1} v_0 + (1-\gamma)\sum_{i=0}^t \gamma^{t-i} g_i^2,$$

$$\text{(AMSGrad)} \quad \psi\left(v_0, \{g_i^2\}_{i=0}^t\right) = \max_{m=0,\ldots,t} \gamma^{m+1} v_0 + (1-\gamma)\sum_{i=0}^m \gamma^{m-i} g_i^2.$$

NESTED ADAPTIVE (NEADA) METHODS. NeAda, presented in Algorithm 10, has a nesting inner loop to maximize y until some stopping criterion is reached (see details in Chapter 5.3). Instead of using a fixed number of inner iterations or a fixed target accuracy as in GDmax [Lin et al., 2020a, Nouiehed et al., 2019], NeAda gradually increases the accuracy of the inner loop as the outer loop proceeds to make it fully adaptive.

We refer to the ratio between two learning rates, i.e. η^y/η^x, as the two-time-scale. The current analysis of GDA in nonconvex-strongly-concave setting requires two-time-scale to be proportional with the condition number $\kappa = l/\mu$, where l and μ are Lipschitz smoothness and strongly-concavity modulus [Lin et al., 2020a, Yang et al., 2022b]. We provide an example showing that the problem-dependent two-time-scale is *necessary* for GDA and most non-nested methods even in the deterministic setting.

Algorithm 9 Non-nested Adaptive Method

Input: x_0 and y_0
for $t = 0, 1, 2, \ldots$ **do**
  sample ξ_t and let
  $g_t^x = \nabla_x F(x_t, y_t; \xi_t)$ and
  $g_t^y = \nabla_y F(x_t, y_t; \xi_t)$
  `// update the first moment`
  $m_{t+1}^x = \beta^x m_t^x + (1-\beta^x) g_t^x$ and
  $m_{t+1}^y = \beta^y m_t^y + (1-\beta^y) g_t^y$
  `// update the second moment`
  $v_{t+1}^x = \psi\left(v_0^x, \{(g_i^x)^2\}_{i=0}^t\right)$ and
  $v_{t+1}^y = \psi\left(v_0^y, \{(g_i^y)^2\}_{i=0}^t\right)$
  `// update variables`
  $x_{t+1} = x_t - \frac{\eta^x}{\sqrt{v_{t+1}^x}} m_{t+1}^x$ and
  $y_{t+1} = y_t + \frac{\eta^y}{\sqrt{v_{t+1}^y}} m_{t+1}^y$
end for

Algorithm 10 Nested Adaptive (NeAda) Method

Input: x_0 and y_0^0
for $t = 0, 1, 2, \ldots$ **do**
  for $k = 0, 1, 2, \ldots$ until a stopping criterion is satisfied **do**
    sample $\hat{\xi}_t^k$ and $g_{t,k}^y = \nabla_y F(x_t, y_t^k; \hat{\xi}_t^k)$
    $m_{t,k+1}^y = \beta^y m_{t,k}^y + (1-\beta^y) g_{t,k}^y$
    $v_{t,k+1}^y = \psi^y\left(v_{t,0}^y, \{(g_{t,i}^y)^2\}_{i=0}^k\right)$
    $y_t^{k+1} = y_t^k + \frac{\eta^y}{\sqrt{v_{t,k+1}^y}} m_{t,k+1}^y$
  end for
  $v_{t+1,0}^y = v_{t,k+1}^y$ and $m_{t+1,0}^y = m_{t,k+1}^y$
  sample ξ_t and $g_t^x = \nabla_x F(x_t, y_t^{k+1}; \xi_t)$
  $m_{t+1}^x = \beta^x m_t^x + (1-\beta^x) g_t^x$
  $v_{t+1}^x = \psi^x\left(v_0^x, \{(g_i^x)^2\}_{i=0}^t\right)$
  $x_{t+1} = x_t - \frac{\eta^x}{\sqrt{v_{t+1}^x}} m_{t+1}^x$
end for

Lemma 5.2.1. *Consider the function $f(x,y) = -\frac{1}{2}y^2 + Lxy - \frac{L^2}{2}x^2$ in the deterministic setting. Let $r\eta^x = \eta^y$. (1) GDA will not converge to the stationary point when $r \le L^2$:*

$$\nabla_x f(x_T, y_T) = \nabla_x f(x_0, y_0) \prod_{t=0}^{T-1} \left[1 + \eta^x (L^2 - r)\right].$$

(2) Assume the averaging function ψ^x and ψ^y are the same, and satisfy that for any τ, if $v_t^x = \tau v_t^y$ and $(g_t^x)^2 = \tau (g_t^y)^2$ then $v_{t+1}^x = \tau v_{t+1}^y$. With $\beta^x = \beta^y$, $v_0^x = v_0^y = 0$ and $m_0^x = m_0^y = 0$ (which are commonly used in practice), non-nested adaptive method will not converge when $r \le L$:

$$\nabla_x f(x_T, y_T) \ge \nabla_x f(x_0, y_0) \prod_{t=0}^{T-1} \left[1 + \frac{L\eta^x}{\sqrt{v_t^x}} (1-\beta^x)(L-r)\right].$$

When $r = L$, $\nabla_x f(x_t, y_t) = \nabla_x f(x_0, y_0)$ for all t.

Remark 5.2.2. *Most popular adaptive stepsizes we mentioned before, such as Adam, AMSGrad and AdaGrad, have averaging functions satisfying the assumption in the lemma. Any point on the line $y = Lx$ is a stationary point for the above function, and the distance from a point to this line is proportional to its gradient norm, so the divergence in gradient norm will also implies that of iterates. In the proof, we will also show that the averaged or best iterate will still diverge under the same condition. The lemma implies that for any given time-scale r, there exists a problem*

for which the non-nested algorithm does not converge to the stationary point, so they are not parameter-agnostic.

We compare non-nested and nested methods combined with different stepsizes schemes: Adam, AMSGrad, AdaGrad and fixed stepsize, on the function: $-\frac{1}{2}y^2 + 2xy - 2x^2$. In the experiments of this subchapter, we halt the inner loop when the (stochastic) gradient about y is smaller than $1/t$ or the number iteration is greater than t. We observe from Figure 5.1 that the thresholds for the non-convergence of non-nested methods ($r = 2$ for adaptive methods and $r = 4$ for GDA) are exactly as predicted by the lemma. Although the adaptive methods admit a smaller two-time-scale threshold than GDA in this example, it is not a universal phenomenon from our experiments in Chapter 5.4. Interestingly, nested adaptive methods are robust to different two-time-scales and always have the trend to converge to the stationary point.

5.3 CONVERGENCE ANALYSIS OF NEADA-ADAGRAD

In this subchapter, we reveal the secret behind the robust performance of NeAda by providing the convergence guarantee for a representative member of the family. For the sake of simplicity and clarity, we mainly focus on NeAda with AdaGrad. The Adam-type mechanism can suffer from non-convergence already for nonconvex minimization despite its good performance in practice. Our result also sheds light on the analysis of other more sophisticated members such as AMSGrad in the family.

NEADA-ADAGRAD: Presented in Algorithm 11, NeAda-AdaGrad adopts the scalar AdaGrad scheme for the x-update in the outer loop and uses mini-batch in the stochastic setting. For the inner loop for maximizing y, we run some adaptive algorithm for maximizing y until some easily checkable stopping criterion is satisfied. We suggest two criteria here: at t-th outer loop: (I) the squared gradient mapping norm about y is smaller than $1/(t+1)$ in the deterministic setting, (II) the number of inner loop iterations reaches $t+1$ in the stochastic setting.

For the purpose of theoretical analysis, we mainly focus on the minimax problem of the form (5.1) under the nonconvex-strongly-concave (NC-SC) setting[2], formally stated in the following assumptions.

Assumption 13 (Lipschitz smoothness)**.** *There exists a positive constant $l > 0$ such that*

$$\max\left\{ \left\|\nabla_x f(x_1, y_1) - \nabla_x f(x_2, y_2)\right\|, \left\|\nabla_y f(x_1, y_1) - \nabla_y f(x_2, y_2)\right\| \right\} \leq l[\|x_1 - x_2\| + \|y_1 - y_2\|],$$

2 Note that for other nonconvex minimax optimization beyond the NC-SC setting, even the convergence of non-adaptive gradient methods has not been fully understood.

Algorithm 11 NeAda-AdaGrad

1: Input: (x_0, y_{-1}), $v_0 > 0$, $\eta > 0$.
2: **for** $t = 0, 1, 2, ..., T-1$ **do**
3: from y_{t-1} run an adaptive algorithm $\mathcal{A}$ for maximizing $f(x_t, \cdot)$ to obtain y_t
(a) stopping criterion I (deterministic): stop when $\|y_t - \text{Proj}_{\mathcal{Y}}(y_t + \nabla_y f(x_t, y_t))\|^2 \leq \frac{1}{t+1}$
(b) stopping criterion II (stochastic): stop after $t+1$ inner loop iterations.
4: $v_{t+1} = v_t + \left\| \frac{1}{M} \sum_{i=1}^{M} \nabla_x F(x_t, y_t; \xi_t^i) \right\|^2$ where $\{\xi_t^i\}_{i=1}^M$ are i.i.d samples
5: $x_{t+1} = x_t - \frac{\eta}{\sqrt{v_{t+1}}} \left(\frac{1}{M} \sum_{i=1}^{M} \nabla_x F(x_t, y_t; \xi_t^i) \right)$
6: **end for**

holds for all $x_1, x_2 \in \mathbb{R}^d, y_1, y_2 \in \mathcal{Y}$.

Assumption 14 (Strong-concavity in y)**.** *There exists* $\mu > 0$ *such that:* $f(x, y_1) \geq f(x, y_2) + \langle \nabla_y f(x, y_1), y_1 - y_2 \rangle + \frac{\mu}{2} \|y_1 - y_2\|^2, \forall x \in \mathbb{R}^d, y_1, y_2 \in \mathcal{Y}$.

For simplicity of notation, define $\kappa = l/\mu$ as the condition number, $\Phi(x) = \max_{y \in \mathcal{Y}} f(x, y)$ as the primal function, and $y^*(x) = \text{argmax}_{\mathcal{Y}} f(x, y)$ as the optimal y w.r.t x. Since the objective is nonconvex about x, we aim at finding an ϵ-stationary point (x_t, y_t) such that $\mathbb{E}\|\nabla_x f(x_t, y_t)\| \leq \epsilon$ and $\mathbb{E}\|y_t - y^*(x_t)\| \leq \epsilon$, where the expectation is taken over the randomness in the algorithm.

5.3.1 *Convergence in Deterministic and Stochastic settings*

Assumption 15 (Stochastic gradients)**.** $\nabla_x F(x, y; \xi)$ *and* $\nabla_y F(x, y; \xi)$ *are unbiased stochastic estimators of* $\nabla_x f(x, y)$ *and* $\nabla_y f(x, y)$ *and have variances bounded by* $\sigma^2 \geq 0$.

We assume the unbiased stochastic gradients have the variance σ^2, and the problem reduces to the deterministic setting when $\sigma = 0$. Now we provide a general analysis of the convergence for any adaptive optimizer used in the inner loop.

Theorem 5.3.1. *Define the expected cumulative suboptimality of inner loops as* $\mathcal{E} = \mathbb{E}\left[\sum_{t=0}^{T-1} \frac{l^2 \|y_t - y^*(x_t)\|^2}{2\sqrt{v_0}}\right]$. *Under Assumptions 13, 14 and 15, the output from Algorithm 11 satisfies*

$$\mathbb{E}\left[\sqrt{\frac{1}{T}\sum_{t=0}^{T-1} \|\nabla_x f(x_t, y_t)\|^2}\right] \leq \frac{2(A+\mathcal{E})}{\sqrt{T}} + \frac{v_0^{\frac{1}{4}}\sqrt{A+\mathcal{E}}}{\sqrt{T}} + \frac{2\sqrt{(A+\mathcal{E})\sigma}}{(MT)^{\frac{1}{4}}},$$

where $A = \frac{2\Delta}{\eta} + \left(\frac{4\sigma}{\sqrt{M}} + 2\kappa l\eta\right)\left[1 + 2\log\left(\text{Poly}\left(T, \mathcal{E}, \frac{\Delta}{\eta}, \frac{\sigma}{\sqrt{M}}, \kappa l\eta, v_0, \frac{1}{v_0}\right)\right)\right]$.

Remark 5.3.2. *The general analysis is built upon milder assumptions than existing work on AdaGrad in nonconvex optimization, not requiring either bounded gradient in [Ward et al., 2019] or prior knowledge about the smoothness modulus in [Li and Orabona, 2019]. This theorem implies the algorithm attains convergence for the nonconvex variable x with any constant $\eta > 0$ and $v_0 > 0$ that does not depend on any problem parameter, so it is parameter-agnostic.*

Remark 5.3.3. *Another benefit of this analysis is that the variance σ appears in the leading term $T^{-\frac{1}{4}}$, which means the convergence rate can interpolate between the deterministic and stochastic settings. It implies a complexity of $\widetilde{O}(\epsilon^{-2})$ in the deterministic setting and $\widetilde{O}(\epsilon^{-4})$ in the stochastic setting for the primal variable as long as the accumulated suboptimality for the inner-loops $\mathcal{E}$ is $\widetilde{\mathcal{O}}(1)$, regardless of the batch size M. However, M can control the number of outer loops and there affect the sample complexity for the dual variable.*

In the next two theorems, we derive the total complexities, in the deterministic and stochastic settings, of finding ϵ-stationary point by controlling the cumulative suboptimality $\mathcal{E}$ in Theorem 5.3.1 for subroutine $\mathcal{A}$ with specific convergence rate. In fact, we can also use any off-the-shelf adaptive optimizer for solving the inner maximization problem up to the desired accuracy. Note that (stochastic) GDmax fixes each inner-loop's accuracy or steps to be related with μ, ℓ and ϵ so that $\mathcal{E}$ can be easily bounded [Lin et al., 2020a, Nouiehed et al., 2019]. In contrast, since we do not have access to the problem parameters and ϵ, Algorithm 11 gradually increases the inner-loop accuracy. In the proof of the following theorems, we will show that with our proposed stopping criteria and desired subroutines, $\mathcal{E}$ is bounded by $\mathcal{O}(\log T)$.

Theorem 5.3.4 (deterministic)**.** *Suppose we have a linearly-convergent subroutine $\mathcal{A}$ for maximizing any strongly concave function $h(\cdot)$:*

$$\|y^k - y^*\|^2 \leq a_1(1-a_2)^k\|y^0 - y^*\|^2$$

where y^k is k-th iterate, y^* is the optimal solution, and $a_1 > 0$ and $0 < a_2 < 1$ are constants that can depend on the parameters of h. Under the same setting as Theorem 5.3.1 with $\sigma = 0$, for Algorithm 11 with $M = 1$ and a subroutine $\mathcal{A}$ under stopping criterion I, there exists $t^* \leq \widetilde{O}\left(\epsilon^{-2}\right)$ such that (x_{t^*}, y_{t^*}) is an ϵ-stationary point. Therefore, the total gradient complexity is $\widetilde{O}\left(\epsilon^{-2}\right)$.

Remark 5.3.5. *This complexity is optimal in ϵ up to logarithmic term [Zhang et al., 2021b], similar to GDA [Lin et al., 2020a]. Note that many adaptive and parameter-agnostic algorithms can achieve the linear rate when solving smooth and strongly concave maximization problems; to list a few, gradient ascent with backtracking line-search [Vaswani et al., 2019], SC-AdaNGD [Levy, 2017] and polyak stepsize [Hazan and Kakade, 2019, Loizou et al., 2021, Orvieto et al., 2022]* [3].

3 Levy [2017] needs to know the diameter of $\mathcal{Y}$. Hazan and Kakade [2019], Loizou et al. [2021], Orvieto et al. [2022] use polyak stepsize which requires knowledge of the minimum or lower bound of the function value.

Here we can also pick more general subproblem accuracy in criterion I that only needs to scale with $1/t$.

Theorem 5.3.6 (stochastic)**.** *Suppose we have a sub-linearly-convergent subroutine $\mathcal{A}$ for maximizing any strongly concave function $h(\cdot)$: after $K = k + 1$ iterations*

$$\mathbb{E}\|y^K - y^*\|^2 \leq \frac{b_1\|y^0 - y^*\|^2 + b_2}{k},$$

where y^k is k-th iterate, y^* is the optimal solution, and $b_1, b_2 > 0$ are constants that can depend on the parameters of h. Under the same setting as Theorem 5.3.1, for Algorithm 11 with $M = \epsilon^{-2}$ and subroutine $\mathcal{A}$ under the stopping criterion II, there exists $t^* \leq \widetilde{O}\left(\epsilon^{-2}\right)$ such that (x_{t^*}, y_{t^*}) is an ϵ-stationary point. Therefore, the total stochastic gradient complexity is $\widetilde{O}\left(\epsilon^{-4}\right)$.

Remark 5.3.7. *This $\widetilde{O}\left(\epsilon^{-4}\right)$ complexity is nearly optimal in the dependence of ϵ for stochastic NC-SC problems [Li et al., 2021]. Here we set $M = \epsilon^{-2}$ for the simplicity of exposition, and a similar result also holds for gradually increasing M. The sublinear rate specified above for solving the stochastic strongly convex subproblem can be achieved by several existing parameter-agnostic algorithms under some additional assumptions, such as* FreeRexMomentum *[Cutkosky and Boahen, 2017] and Coin-Betting [Cutkosky and Orabona, 2018][4]. Parameter-free SGD [Carmon and Hinder, 2022] is partially parameter-agnostic that only requires the stochastic gradient bound rather than the strongly-convexity parameter. Mukkamala and Hein [2017] and Wang et al. [2020a] introduce the variants of AdaGrad, RMSProp and Adam for strongly-convex online learning, but they need to know both gradient bounds and strongly-convexity parameter for setting stepsizes. We will show in the next subchapter that AdaGrad with a slower decaying rate is parameter-agnostic. We note that the analysis of this theorem is not the simple gluing of the outer loop and inner loop complexity, but requires more sophisticated control of the cumulative suboptimality $\mathcal{E}$.*

5.3.2 *Generalized AdaGrad for Strongly-Convex Subproblem*

We now introduce the generalized AdaGrad for minimizing strongly convex objectives, which can serve as an adaptive subroutine for Algorithm 11, without requiring knowledge on the strongly convex parameter. We analyze it for the more general online convex optimization setting: at each round t, the learner updates its decision x_t, then it suffers a loss $f_t(x_t)$ and receives the sub-gradient of f_t. The generalized AdaGrad, described in Algorithm 12, keeps the cumulative gradient norm v_t and takes the stepsize η/v_t^α with

AdaGrad achieves the linear rate if the learning rate is smaller than $O(1/l)$, and $O(1/k)$ rate otherwise [Xie et al., 2020].

4 FreeRexMomentum [Cutkosky and Boahen, 2017] and Coin-Betting [Cutkosky and Orabona, 2018] can achieves $\mathcal{O}(\log k/k)$ convergence rate when the stochastic gradient is bounded in $\mathcal{Y}$. If the subroutine has additional logarithmic dependence, it suffices to run the subroutine for $t\log^2(t)$ times using criterion II (see Appendix 5.5.2).

a decaying rate $\alpha \in (0, 1]$. When $\alpha = 1/2$, it reduces to the scalar version of the original AdaGrad [Duchi et al., 2011]; when $\alpha = 1$, it reduced to the scalar version of SC-AdaGrad [Mukkamala and Hein, 2017].

Algorithm 12 Generalized AdaGrad for Strongly-convex Online Learning

1: Input: x_0, $v_0 > 0$ and $0 < \alpha \leq 1$.
2: **for** $t = 0, 1, 2, \ldots$ **do**
3: receive $g_t \in \partial f_t(x_t)$
4: $v_{t+1} = v_t + \|g_t\|^2$
5: $x_{t+1} = \mathcal{P}_{\mathcal{X}}\left(x_t - \frac{\eta}{v_{t+1}^{\alpha}} g_t\right)$
6: **end for**

Theorem 5.3.8. *Consider Algorithm 12 for online convex optimization and assume that (i) f_t is continuous and μ-strongly convex, (ii) $\mathcal{X}$ is convex and compact with diameter $\mathcal{D}$; (iii) $\|g_t\| \leq G$ for every t. Then for $0 < \alpha < 1$ with any $\eta > 0$, the regret of Algorithm 12 satisfies:*

$$\max_{x \in \mathcal{X}} \sum_{t=0}^{T-1} (f_t(x_t) - f_t(x)) \leq c_\alpha + d_\alpha \left(v_0 + \sum_{t=1}^{T-1} \|g_t\|^2\right)^{1-\alpha},$$

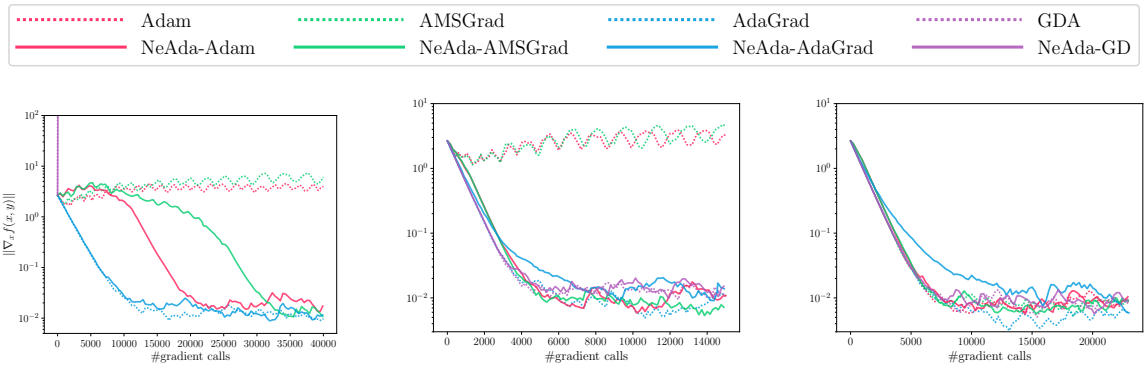
and for $\alpha = 1$ with $\eta \geq \frac{G^2}{2\mu}$,

$$\max_{x \in \mathcal{X}} \sum_{t=0}^{T-1} (f_t(x_t) - f_t(x)) \leq c_\alpha + d_\alpha \log\left(v_0 + \sum_{t=1}^{T-1} \|g_t\|^2\right),$$

where c_α and d_α are constants depending on the problem parameters, α and η.

The theorem implies a logarithmic regret for the case $\alpha = 1$, but the stepsize needs knowledge about problem's parameters μ and G; similar results are shown for SC-AdaGrad [Mukkamala and Hein, 2017] and SAdam [Wang et al., 2020a]. When $\alpha < 1$, the algorithm becomes parameter-agnostic and attains an $O(T^{1-\alpha})$ regret. Such parameter-agnostic phenomenon for smaller decaying rates is also observed for SGD in stochastic optimization [Fontaine et al., 2021]. Proving the regret bound for the generalized AdaGrad with $\alpha < 1$ in the online setting is challenging, since the adversarial g_t can lead to a "sudden" change in the stepsize. In the proof, we bound the possible number of times such "sudden" change could happen.

To the best of our knowledge, this is the first regret bound for adaptive methods with general decaying rates in the strongly convex setting. By online-to-batch conversion [Kakade and Tewari, 2008], it can be converted to $O(T^{-\alpha})$ rate in the strongly convex stochastic optimization. Xie et al. [2020] prove the $O(1/T)$ convergence rate, or a linear convergence rate when the smoothness parameter is known, for AdaGrad with $\alpha = 1/2$ in this setting,



(a) $r = 0.01$ (b) $r = 0.03$ (c) $r = 0.05$

FIGURE 5.2: Comparison between the two families of non-nested and nested adaptive methods on McCromick function with stochastic gradient oracles. $\sigma = 0.01$, $\eta^y = 0.01$ and $r = \eta^y/\eta^x$.

but under a strong assumption — Restricted Uniform Inequality of Gradients (RUIG) — that requires the loss function with respect to each sample ξ to satisfy the error bound condition with some probability.

5.4 EXPERIMENTS

To evaluate the performance of NeAda, we conducted experiments on simple test functions and a real-world application of distributional robustness optimization (DRO). In all cases, we compare NeAda with the non-nested adaptive methods using the same adaptive schemes. For notational simplicity, in all figure legends, we label the non-nested methods with the names of the adaptive mechanisms used. We observe from all our experiments that: 1) while non-nested adaptive methods can diverge without the proper two-time-scale, NeAda with adaptive subroutine always converges; 2) when the non-nested method converges, NeAda can achieve comparable or even better performance.

5.4.1 *Test functions*

In Chapter 5.2, we have compared NeAda with non-nested methods on a quadratic function in Figure 5.1 and the observations match Theorem 5.2.1. Now we consider a more complicated function that is composed of McCormick function in x, a bilinear term, and a quadratic term in y,

$$f(x,y) = \sin(x_1 + x_2) + (x_1 - x_2)^2 - \frac{3}{2}x_1 + \frac{5}{2}x_2 + 1 + x_1y_1 + x_2y_2 - \frac{1}{2}(y_1^2 + y_2^2),$$

For this function, we compare the adaptive frameworks in the stochastic setting with Gaussian noise. As demonstrated in Figure 5.2, non-nested methods are sensitive to the selection of the two-time-scale. When the learning rate ratio is too small, e.g., $\eta^y/\eta^x = 0.01$, non-nested Adam, AMSGrad and GDA all fail to converge. We observe that GDA converges when the ratio reaches 0.03, while non-nested Adam and AMSGrad still diverge until 0.05. Although non-nested adaptive methods require a smaller ratio than GDA in Lemma 5.2.1, this example illustrates that adaptive algorithms sometimes can be more sensitive to the time separation. In comparison, NeAda with adaptive subroutine always converges regardless of the learning rate ratio.

5.4.2 *Distributional robustness optimization*

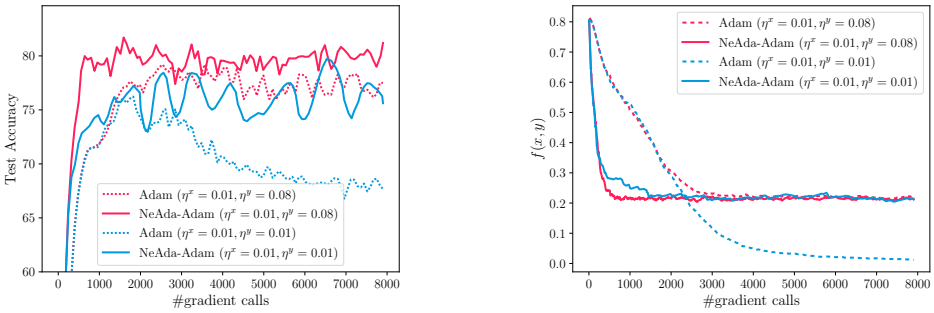
To justify the effectiveness of NeAda on real-world applications, we carried out experiments on distributionally robust optimization [Sinha et al., 2018], where the primal variable is the model weights to be learned by minimizing the empirical loss while the dual variable is the adversarial perturbed inputs. The dual variable problem targets finding perturbations that maximize the empirical loss but not far away from the original inputs. Formally, for model weights x and adversarial samples y, we have:

$$\min_x \max_{y=[y_1,\dots,y_n]} f(x,y), \quad \text{where} \quad f(x,y) := \frac{1}{n}\sum_{i=1}^{n} f_i(x,y_i) - \gamma\|y_i - v_i\|^2,$$

where n is the total number of training samples, v_i is the i-th original input and f_i is the loss function for the i-th sample. γ is a trade-off parameter between the empirical loss and the magnitude of perturbations. When γ is large enough, this problem is nonconvex-strongly-concave, and following the same setting as [Sinha et al., 2018, Sebbouh et al., 2022], we set $\gamma = 1.3$. For NeAda, we use both stopping criterion I with stochastic gradient and criterion II in our experiments. For the results, we report the training loss and the test accuracy on adversarial samples generated from fast gradient sign method (FGSM) [Goodfellow et al., 2015]. FGSM can be formulated as

$$x_{\text{adv}} = x + \epsilon \cdot \text{sign}\left(\nabla_x f(x)\right),$$

where ϵ is the noise level. To get reasonable test accuracy, NeAda with Adam as subroutine is compared with Adam with fixed 15 inner loop iterations, which is consistent with the choice of inner loop steps in [Sinha et al., 2018], and such choice obtains much better test accuracy than the completely non-nested Adam. Our experiments include a synthetic dataset and MNIST [LeCun, 1998] with code modified from [Lv, 2019].



(a) Test Accuracy (b) Training Loss

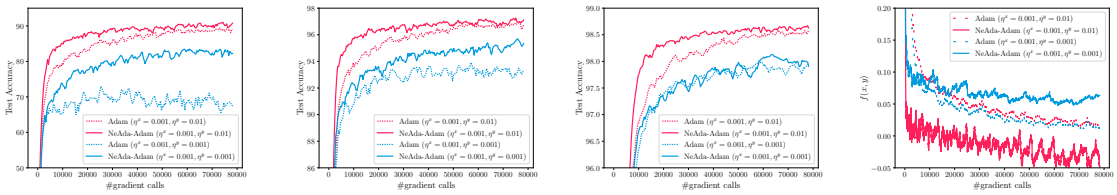
FIGURE 5.3: Experimental results of distributional robustness optimization task on a synthetic dataset.

RESULTS ON SYNTHETIC DATASET. We use the same data generation process as in [Sinha et al., 2018]. The inputs are 2-dimensional i.i.d. random Guassian vectors, i.e., $x_i \sim \mathcal{N}(0, I_2)$, where I_2 is the 2×2 identity matrix. The corresponding y_i is defined as $y_i = \text{sign}(\|x_i\|_2 - \sqrt{2})$. Data points with norm in range $(\sqrt{2}/1.3, 1.3\sqrt{2})$ are removed to make the classification margin wide. 10000 training and 4000 test data points are generated for our experiments. The model we use is a three-layer MLP with ELU activations.

As shown in Figure 5.3(a), when the learning rates are set to different scales, i.e., $\eta^x = 0.01, \eta^y = 0.08$ (red curves in the figure), both methods achieve reasonable test errors. In this case, NeAda has higher test accuracy and reaches such accuracy faster than Adam. If we change the learning rates to the same scale, i.e., $\eta^x = 0.01, \eta^y = 0.01$ (blue curves in the figure), NeAda retains good accuracy while Adam drops to an unsatisfactory performance. This demonstrates the adaptivity and less-sensitivity to learning rates of NeAda. In addition, Figure 5.3(b) illustrates the convergence speeds on the loss function, and NeAda (solid lines) always decreases the loss faster than Adam. Note that Adam with the same learning rates converges to a lower loss but suffers from overfitting, as shown in Figure 5.3(a) that its test accuracy is only about 68%.

RESULTS ON MNIST DATASET. For MNIST, we use a convolutional neural network with three convolutional layers and one final fully-connected layer. Following each convolutional layer, ELU activation and batch normalization are used.

We compare NeAda with Adam under three different noise levels and the accuracy is shown in Figures 5.4(a) to 5.4(c). Under all noise levels, NeAda outperforms Adam with the same learning rates. When we have proper time-scale separation (the red curves), both methods achieve good test accuracy, and NeAda achieves higher accuracy and converges



(a) $\epsilon = 0.1$ (b) $\epsilon = 0.05$ (c) $\epsilon = 0.02$ (d) loss

FIGURE 5.4: Results of distributional robustness optimization task on MNIST. ϵ is the noise level.

faster. After we change to the same learning rates for the primal and dual variables (the blue curves), the accuracy drop of NeAda is slighter compared to Adam, especially when $\epsilon = 0.1$. As for the training loss shown in Figure 5.4(d), NeAda (the solid curves) is always faster at the beginning. We also observed that with proper time-scale separation, NeAda reaches a lower loss.

5.5 APPENDIX

5.5.1 *Helper Lemmas and Proofs for Chapter 5.2*

A. Helper Lemmas

Lemma 5.5.1 (Lemma 4.3 in [Lin et al., 2020a] and Lemma A.5 in [Nouiehed et al., 2019])**.** *Under Assumption 13 and 14, define* $\Phi(x) = \max_{y\in\mathcal{Y}} f(x,y)$. *Define* $y^*(x) = \operatorname{argmax}_{y\in\mathcal{Y}} f(x,y)$. *Then* $y^*(\cdot)$ *is* κ*-Lipschitz with* $\kappa = \frac{l}{\mu}$, $\Phi(\cdot)$ *is L-smooth with* $L := l + l\kappa$ *and* $\nabla\Phi(x) = \nabla_x f\left(x, y^*(x)\right)$.

Lemma 5.5.2. *Let* $x_1, ..., x_T$ *be a sequence of non-negative real numbers,* $x_1 > 0$ *and* $0 < \alpha < 1$. *Then we have*

$$\left(\sum_{t=1}^{T} x_t\right)^{1-\alpha} \leq \sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{t} x_k\right)^{\alpha}} \leq \frac{1}{1-\alpha}\left(\sum_{t=1}^{T} x_t\right)^{1-\alpha}.$$

When $\alpha = 1$, *we have*

$$\sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{t} x_k\right)^{\alpha}} \leq 1 + \log\left(\frac{\sum_{t=1}^{t} x_t}{x_1}\right).$$

Remark 5.5.3. *The case* $\alpha = 1/2$ *has been noticed in [Auer et al., 2002], and the upper bound in the case* $\alpha = 1$ *has already been noticed in [Ward et al., 2019]. Here we we extend it to* $0 < \alpha \leq 1$.

Proof. For the first inequality, we have

$$\sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{t} x_k\right)^{\alpha}} \geq \sum_{t=1}^{T} \frac{x_t}{\left(\sum_{k=1}^{T} x_k\right)^{\alpha}} = \frac{\sum_{t=1}^{T} x_t}{\left(\sum_{t=1}^{T} x_t\right)^{\alpha}} = \left(\sum_{t=1}^{T} x_t\right)^{1-\alpha}.$$

For the second inequality, we follow a similar procedure as in the proof of Lemma 3.5 of [Auer et al., 2002]. First consider the case $0 < \alpha < 1$. By Bernoulli's inequality, as $y \leq 1$ and $0 < \alpha < 1$, we have $1 - (1-\alpha)y \geq (1-y)^{1-\alpha}$. Denoting $S_t = \sum_{k=1}^{t} x_k$ and $S_0 = 0$, by replacing y with x_t/S_t, we have

$$(1-\alpha)\frac{x_t}{S_t} \leq 1 - \left(1 - \frac{x_t}{S_t}\right)^{1-\alpha}.$$

Multiplying both sides by $S_t^{1-\alpha}$, then we have

$$(1-\alpha)\frac{x_t}{S_t^{\alpha}} \leq S_t^{1-\alpha} - S_{t-1}^{1-\alpha}.$$

Summing over the inequalities for $t = 1, ..., T$ gives us the desired result. For $\alpha = 1$, it is proved by [Ward et al., 2019]. □

Proposition 3. *If* $x^2 \leq (a_1 + a_2 x)\left(a_3 + a_4 \log(a_5 a_1 + a_5 a_2 x)\right)$ *with* $x, a_1, a_2, a_3, a_4, a_5 \geq 0$ *and* $a_2 > 0$, *then*

$$x \leq \frac{a_1}{a_2} + 16 a_2^3 a_4^2 a_5 + 3 a_2^2 a_3^2$$

Proof. The proof is similar to Lemma 6 in [Li and Orabona, 2019]. If $a_2 x < a_1$, we have $x \leq a_1/a_2$. Assume $a_2 x \geq a_1$, then

$$x^2 \leq 2 a_2 x (a_3 + a_4 \log(2 a_5 a_2 x)) \leq 2 a_2 x \left(a_3 + a_4 \sqrt{2 a_5 a_2 x}\right),$$

which implies

$$x \leq 2 a_2 a_3 + 2 a_2 a_4 \sqrt{2 a_5 a_2 x} \quad \Longrightarrow \quad x^2 \leq 8 a_2^2 a_3^2 + 16 a_2^3 a_4^2 a_5 x.$$

Solving this, we get

$$x \leq 8 a_2^3 a_4^2 a_5 + \sqrt{64 a_2^6 a_4^4 a_5^2 + 8 a_2^4 a_3^4} \leq 16 a_2^3 a_4^2 a_5 + 3 a_2^2 a_3^2.$$

□

Proposition 4. *Assume* $x_t, a_t, b_t > 0$, *for* $t = 0, 1, 2, \ldots$, *and* $x_{t+1} \leq a_t x_t + b_t$, *then we have*

$$x_T \leq \left(\prod_{t=0}^{T-1} a_t\right) x_0 + \sum_{t=0}^{T-2} \left(\prod_{i=t+1}^{T-1} a_i\right) b_t + b_{T-1}, \quad T \geq 2$$

Proof. Let's prove it by induction. It is obvious that this inequality holds for $T = 2$:

$$x_2 = a_1 x_1 + b_1 = a_1 a_0 x_0 + a_1 b_0 + b_1.$$

Assume this inequality holds for T, then

$$\begin{aligned} x_{T+1} \leq & a_T \left[\left(\prod_{t=0}^{T-1} a_t\right) x_0 + \sum_{t=0}^{T-2} \left(\prod_{i=t+1}^{T-1} a_i\right) b_t + b_{T-1}\right] + b_T \\ = & \left(\prod_{t=0}^{T} a_t\right) x_0 + \sum_{t=0}^{T-1} \left(\prod_{i=t+1}^{T} a_i\right) b_t + b_T. \end{aligned}$$

□

Lemma 5.5.4. *Assume* $x_t > 0$, *for* $t = 0, 1, 2, \ldots$, *and* $x_{t+1} = a_1 x_t/(t+1) + a_2/(t+1)$, *then we have*

$$\sum_{t=0}^{T} x_t \leq a_2(1 + \log T) + a_2 e^{a_1} + x_0 e^{a_1}.$$

Proof. By Proposition 4, we have

$$\begin{aligned} \sum_{t=0}^{T} x_t \leq & x_0 + (a_1 x_0 + a_2) + \sum_{t=2}^{T} \left[\left(\prod_{i=0}^{t-1} \frac{a_1}{i+1}\right) x_0 + \sum_{i=0}^{t-2} \left(\prod_{j=i+1}^{t-1} \frac{a_1}{j+1}\right) \frac{a_2}{i+1} + \frac{a_2}{t}\right] \\ = & x_0 + x_0 \sum_{t=1}^{T} \prod_{i=0}^{t-1} \frac{a_1}{i+1} + \sum_{t=2}^{T} \left[\sum_{i=0}^{t-2} \left(\prod_{j=i+1}^{t-1} \frac{a_1}{j+1}\right) \frac{a_2}{i+1}\right] + \sum_{t=1}^{T} \frac{a_2}{t}. \end{aligned} \tag{5.2}$$

We note that $\sum_{t=1}^{T} \frac{a_2}{t} \le a_2(1+\log T)$ and

$$x_0 \sum_{t=1}^{T} \prod_{i=0}^{t-1} \frac{a_1}{i+1} = x_0 \sum_{t=1}^{T} \frac{a_1^t}{t!} \le x_0 e^{a_1},$$

where the last inequality can be derived from Taylor expansion of exponential function. Then it remains to bound the third term on the right hand side of (5.2). We can upper bound it by noticing

$$\begin{aligned}
\sum_{t=2}^{T} \left[\sum_{i=0}^{t-2} \left(\prod_{j=i+1}^{t-1} \frac{a_1}{j+1} \right) \frac{a_2}{i+1} \right] =& a_2 \sum_{t=1}^{T-1} \sum_{i=1}^{T-t} \left(\prod_{j=i}^{i+t-1} \frac{a_1}{j+1} \right) \frac{1}{i} \\
=& a_2 \sum_{t=1}^{T-1} a_1^t \sum_{i=1}^{T-t} \prod_{j=i}^{i+t} \frac{1}{j} \\
=& a_2 \sum_{t=1}^{T-1} a_1^t \sum_{i=1}^{T-t} \frac{1}{t} \left(\prod_{j=i}^{i+t-1} \frac{1}{j} - \prod_{j=i+1}^{i+t} \frac{1}{j} \right) \\
=& a_2 \sum_{t=1}^{T-1} \frac{a_1^t}{t} \left(\prod_{j=1}^{t} \frac{1}{j} - \prod_{j=T-t+1}^{T} \frac{1}{j} \right) \le a_2 \sum_{t=1}^{T-1} \frac{a_1^t}{t \cdot (t!)} \le a_2 e^{a_1},
\end{aligned}$$

where in the third equality we use $\frac{1}{t}\left(\prod_{j=i}^{i+t-1} \frac{1}{j} - \prod_{j=i+1}^{i+t} \frac{1}{j}\right) = \prod_{j=i}^{i+t} \frac{1}{j}$, the last inequality can be derived from Taylor expansion of exponential function; and to see the first equality, the left hand side is the sum of the following

$$\begin{array}{cccc}
a_2 \times \frac{a_1}{2} & & & \\
a_2 \times \frac{a_1}{2} \times \frac{a_1}{3} & \frac{a_2}{2} \times \frac{a_1}{3} & & \\
a_2 \times \frac{a_1}{2} \times \frac{a_1}{3} \times \frac{a_1}{4} & \frac{a_2}{2} \times \frac{a_1}{3} \times \frac{a_1}{4} & \frac{a_2}{3} \times \frac{a_1}{4} & \\
\vdots & \vdots & & \ddots \\
a_2 \times \frac{a_1}{2} \times \cdots \times \frac{a_1}{T-1} & \frac{a_2}{2} \times \frac{a_1}{3} \times \cdots \times \frac{a_1}{T-1} & & \frac{a_2}{T-2} \times \frac{a_1}{T-1} \\
a_2 \times \frac{a_1}{2} \times \cdots \times \frac{a_1}{T} & \frac{a_2}{2} \times \frac{a_1}{3} \times \cdots \times \frac{a_1}{T} & & \frac{a_2}{T-2} \times \frac{a_1}{T-1} \times \frac{a_1}{T} \quad \frac{a_2}{T-1} \times \frac{a_1}{T},
\end{array}$$

and on the right hand side we sum them by each diagonal.

□

B. Proofs for Chapter 5.2

Proof for Lemma 5.2.1. Note that $\nabla_x f(x,y) = -L^2 x + Ly$ and $\nabla_y f(x,y) = Lx - y$. Then we have

$$\begin{aligned}
\nabla_x f(x_{t+1}, y_{t+1}) &= -L^2 x_{t+1} + L y_{t+1} \\
&= -L^2 \left[x_t - \frac{\eta^x}{\sqrt{v_t^x}} m_t^x \right] + L \left[y_t + \frac{r\eta^x}{\sqrt{v_t^y}} m_t^y \right]
\end{aligned}$$

$$= -L^2 x_t + L y_t + \frac{L^2 \eta^x}{\sqrt{v_t^x}} m_t^x + \frac{L r \eta^x}{\sqrt{v_t^y}} m_t^y.$$

GDA. With $v_t^x = v_t^y = 1$, $m_t^x = -L^2 x_t + L y_t$ and $m_t^y = L x_t - y_t$,

$$\begin{aligned}\nabla_x f(x_{t+1}, y_{t+1}) &= -L^2 x_t + L y_t + L^2 \eta^x (-L^2 x_t + L y_t) + L r \eta^x (L x_t - y_t) \\ &= (-L^2 x_t + L y_t)(1 + L^2 \eta^x - r \eta^x) \\ &= (1 + L^2 \eta^x - r \eta^x) \nabla_x f(x_t, y_t).\end{aligned}$$

ADAPTIVE METHODS. Note that $(g_t^x)^2 = L^2 (g_t^y)^2$, so by our assumption, $v_t^x = L^2 v_t^y$ for all t. Also, with $\beta^x = \beta^y$, we have

$$\begin{aligned}m_t^x + r m_t^y &= \beta^x m_{t-1}^x + (1 - \beta^x)(-L^2 x_t + L y_t) + r \beta^x m_{t-1}^y + r(1 - \beta^x)(L x_t - y_t) \\ &= \beta^x (m_{t-1}^x + r m_{t-1}^y) + \left(1 - \frac{r}{L}\right)(1 - \beta^x) \nabla_x f(x_t, y_t).\end{aligned}$$

Recursing this with

$$\nabla_x f(x_{t+1}, y_{t+1}) = \nabla_x f(x_t, y_t) + \frac{L \eta^x}{\sqrt{v_t^y}} (m_t^x + r m_t^y), \text{ and } \quad m_0^x = m_0^y = 0,$$

when $r \leq L$ we have

$$\nabla_x f(x_T, y_T) \geq \nabla_x f(x_0, y_0) \prod_{t=0}^{T-1} \left[1 + \frac{L \eta^x}{\sqrt{v_t^x}} (1 - \beta^x)(L - r)\right].$$

AVERAGED AND BEST ITERATE. We note that the distance from a point (x, y) to the line $y = Lx$, the set of stationary point, is $\frac{|Lx - y|}{\sqrt{L^2 + 1}}$ that is proportional to $|\nabla_x f(x, y)|$. Therefore, the iterate converges to the set of stationary point if and only if the gradient about x converges to 0. This also explains the best iterate will not converge to the set of stationary point for GDA with $r \leq L^2$ and for adaptive methods with $r \leq L$. Average iterate will not converge under the same condition by observing that if an iterate (x_t, y_t) is on the one side of the line $y = Lx$, the next iterate (x_{t+1}, y_{t+1}) will stay on the same side. Without loss of generality, assume (x_t, y_t) is on the right of the line $y = Lx$, i.e., $y_t < L x_t$. By the update of GDA,

$$x_{t+1} = x_t + \eta^x (L^2 x_t - L y_t), \quad y_{t+1} = y_t + r \eta^x (L x_t - y_t),$$

we have $y_{t+1} < L x_{t+1}$ as $r \leq L^2$. For adaptive methods, by the recursion of m_t^x and m_t^y, if $y_s < L x_s$ for all $s \leq t$, we have $-m_t^x > L m_t^y$. The update of adaptive methods can be written as:

$$x_{t+1} = x_t + \frac{\eta^x}{L \sqrt{v_t^y}} (-m_t^x), \quad y_{t+1} = y_t + \frac{r \eta^x}{\sqrt{v_t^y}} m_t^y.$$

Then $y_{t+1} < Lx_{t+1}$ as $r \leq L^2$. Now we conclude that the iterate will always stay on the one side of line $y = Lx$.

□

5.5.2 *Proofs for Chapter 5.3*

A. Proofs for NeAda-AdaGrad

PROOFS FOR THEOREM 5.3.1

Proof. Part of the proof is motivated by [Ward et al., 2019]. By the smoothness of Φ from Theorem 5.5.1, we have

$$\begin{aligned}\Phi(x_{t+1}) &\leq \Phi(x_t) + \langle \nabla\Phi(x_t), x_{t+1} - x_t\rangle + \kappa l\|x_{t+1} - x_t\|^2 \\ &= \Phi(x_t) - \left\langle \nabla\Phi(x_t), \frac{\eta}{\sqrt{v_{t+1}}}\left(\frac{1}{M}\sum_i \nabla_x f(x_t, y_t; \xi_t^i)\right)\right\rangle + \frac{\kappa l\eta^2}{v_{t+1}}\left\|\frac{1}{M}\sum_i \nabla_x f(x_t, y_t; \xi_t^i)\right\|^2.\end{aligned}$$

Note that

$$\mathbb{E}_{\xi_t}\left[\frac{\left\langle \nabla\Phi(x_t), \nabla_x f(x_t, y_t) - \frac{1}{M}\sum_i \nabla_x f(x_t, y_t; \xi_t^i)\right\rangle}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}\right] = 0.$$

Therefore,

$$\begin{aligned}&\mathbb{E}_{\xi_t}\left[\frac{\Phi(x_{t+1}) - \Phi(x_t)}{\eta}\right] \\ &\leq \mathbb{E}_{\xi_t}\left[\left(\frac{1}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} - \frac{1}{\sqrt{v_{t+1}}}\right)\left\langle \nabla\Phi(x_t), \frac{1}{M}\sum_i \nabla_x f(x_t, y_t; \xi_t^i)\right\rangle\right] - \\ &\quad \frac{\langle \nabla\Phi(x_t), \nabla_x f(x_t, y_t)\rangle}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} + \kappa l\eta\,\mathbb{E}_{\xi_t}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t, y_t; \xi_t^i)\right\|^2}{v_{t+1}}\right]. \qquad (5.3)\end{aligned}$$

Now we want to bound the first term on the right hand side and let's denote it as K. First we note that

$$\begin{aligned}&\left\|\frac{1}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} - \frac{1}{\sqrt{v_{t+1}}}\right\| \\ &\leq \left\|\frac{\sqrt{v_{t+1}} - \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}{\sqrt{v_{t+1}}\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}\right\| \\ &= \left\|\frac{\left(\sqrt{v_{t+1}} - \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}\right)\left(\sqrt{v_{t+1}} + \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}\right)}{\sqrt{v_{t+1}}\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}\left(\sqrt{v_{t+1}} + \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}\right)}\right\|\end{aligned}$$

$$= \left\| \frac{\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|^2 - \|\nabla_x f(x_t, y_t)\|^2 - \sigma^2/M}{\sqrt{v_{t+1}} \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M} \left(\sqrt{v_{t+1}} + \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M} \right)} \right\|$$

$$= \left\| \frac{\left(\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\| - \|\nabla_x f(x_t, y_t)\| \right) \left(\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\| + \|\nabla_x f(x_t, y_t)\| \right) - \sigma^2/M}{\sqrt{v_{t+1}} \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M} \left(\sqrt{v_{t+1}} + \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M} \right)} \right\|$$

$$\leq \max \left\{ \frac{\left| \left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\| - \|\nabla_x f(x_t, y_t)\| \right|}{\sqrt{v_{t+1}} \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}, \frac{\sigma/\sqrt{M}}{\sqrt{v_{t+1}} \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} \right\},$$

where in the second equality we use the definition of v_t. Therefore we have

$$K \leq \max \left\{ \mathbb{E}_{\xi_t} \left[\frac{\left| \left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\| - \|\nabla_x f(x_t, y_t)\| \right| \|\nabla \Phi(x_t)\| \left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|^2}{\sqrt{v_{t+1}} \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} \right], \right.$$
$$\left. \mathbb{E}_{\xi_t} \left[\frac{\frac{\sigma}{\sqrt{M}} \|\nabla \Phi(x_t)\| \left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|^2}{\sqrt{v_{t+1}} \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} \right] \right\}. \tag{5.4}$$

By Young's inequality $ab \leq \frac{1}{4\lambda} a^2 + \lambda b^2$ with $\lambda = \frac{\sigma^2/M}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}$, $a = \frac{\left| \left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\| - \|\nabla_x f(x_t, y_t)\| \right| \|\nabla \Phi(x_t)\|}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}$ and $b = \frac{\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|}{\sqrt{v_{t+1}}}$, the first term on the right hand side of (5.4) can be upper bounded by

$$\mathbb{E}_{\xi_t} \left[\frac{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}{4\sigma^2/M} \left(\frac{\left| \left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\| - \|\nabla_x f(x_t, y_t)\| \right| \|\nabla \Phi(x_t)\|}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} \right)^2 \right] +$$
$$\mathbb{E}_{\xi_t} \left[\frac{\sigma^2/M}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} \left(\frac{\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|}{\sqrt{v_{t+1}}} \right)^2 \right]$$
$$\leq \frac{\|\nabla \Phi(x_t)\|^2 \mathbb{E}_{\xi_t} \left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) - \nabla_x f(x_t, y_t) \right\|^2}{\frac{4\sigma^2}{M} \sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} + \frac{\sigma}{\sqrt{M}} \mathbb{E}_{\xi_t} \left[\frac{\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|^2}{v_{t+1}} \right]$$
$$\leq \frac{\|\nabla \Phi(x_t)\|^2}{4\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} + \frac{\sigma}{\sqrt{M}} \mathbb{E}_{\xi_t} \left[\frac{\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|^2}{v_{t+1}} \right].$$

Similarly, by Young's Inequality with $\lambda = \frac{\sigma^2/M}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}$, $a = \frac{\frac{\sigma}{\sqrt{M}} \|\nabla \Phi(x_t)\|}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}$ and $b = \frac{\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|}{\sqrt{v_{t+1}}}$, the second term on the right hand side of (5.4) can be upper bounded by

$$\mathbb{E}_{\xi_t} \left[\frac{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}}{4\sigma^2/M} \left(\frac{\frac{\sigma}{\sqrt{M}} \|\nabla \Phi(x_t)\|}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} \right)^2 \right] +$$
$$\mathbb{E}_{\xi_t} \left[\frac{1}{\sqrt{v_t + \|\nabla_x f(x_t, y_t)\|^2 + \sigma^2/M}} \left(\frac{\left\| \frac{1}{M} \sum_i \nabla_x f(x_t, y_t; \xi_t^i) \right\|}{\sqrt{v_{t+1}}} \right)^2 \right]$$

$$\leq \frac{\|\nabla\Phi(x_t)\|^2}{4\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}+\frac{\sigma}{\sqrt{M}}\mathbb{E}_{\xi_t}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_{t+1}}\right].$$

Therefore,

$$K \leq \frac{\|\nabla\Phi(x_t)\|^2}{4\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}+\frac{\sigma}{\sqrt{M}}\mathbb{E}_{\xi_t}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_{t+1}}\right].$$

Plugging this into (5.3),

$$\begin{aligned}
&\mathbb{E}_{\xi_t}\left[\frac{\Phi(x_{t+1})-\Phi(x_t)}{\eta}\right]\\
\leq& \frac{\|\nabla\Phi(x_t)\|^2}{4\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}+\frac{\sigma}{\sqrt{M}}\mathbb{E}_{\xi_t}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_{t+1}}\right]-\\
&\frac{\langle\nabla\Phi(x_t),\nabla_x f(x_t,y_t)\rangle}{\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}+\kappa l\eta\mathbb{E}_{\xi_t}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_{t+1}}\right]\\
\leq& \left(\frac{\sigma}{\sqrt{M}}+\kappa l\eta\right)\mathbb{E}_{\xi_t}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_{t+1}}\right]-\frac{\|\nabla_x f(x_t,y_t)\|^2}{2\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}+\\
&\frac{\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2}{2\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}},
\end{aligned}\tag{5.5}$$

where in the second inequality we use $\|a\|^2/4-\langle a,b\rangle\leq -\|b\|^2/2+\|a-b\|^2/2$. Applying the total law of probability,

$$\begin{aligned}
&\frac{1}{2}\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{\|\nabla_x f(x_t,y_t)\|^2}{\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}\right]\\
\leq& \frac{\Phi(x_0)-\min_x\Phi(x)}{\eta}+\left(\frac{\sigma}{\sqrt{M}}+\kappa l\eta\right)\mathbb{E}\sum_{t=0}^{T-1}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_{t+1}}\right]+\\
&\mathbb{E}\sum_{t=0}^{T-1}\frac{\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2}{2\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}.
\end{aligned}\tag{5.6}$$

Denote

$$\begin{aligned}
&Z\triangleq\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2,\quad C\triangleq\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{\|\nabla_x f(x_t,y_t)\|^2}{\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}\right],\\
&D\triangleq\mathbb{E}\sum_{t=0}^{T-1}\left[\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_{t+1}}\right],\quad Q\triangleq\mathbb{E}\sum_{t=0}^{T-1}\frac{\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2}{2\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}.
\end{aligned}$$

By Theorem 5.5.2 with $\alpha=1$,

$$\begin{aligned}
D\leq&\mathbb{E}\left[1+\log\left(1+\sum_{t=0}^{T-1}\frac{\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)\right\|^2}{v_0}\right)\right]\\
\leq&1+\mathbb{E}\left[\log\left(1+\frac{\sum_{t=0}^{T-1}\left\|f(x_t,y_t;\xi_t^i)\right\|^2+\sum_{t=0}^{T-1}\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)-\nabla_x f(x_t,y_t)\right\|^2}{v_0}\right)\right]
\end{aligned}$$

$$
\begin{aligned}
&\leq 1+2\mathbb{E}\left[\log\left(1+\frac{Z+\sum_{t=0}^{T-1}\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)-\nabla_x f(x_t,y_t)\right\|^2}{v_0}\right)^{1/2}\right]\\
&\leq 1+2\mathbb{E}\left[\log\left(1+\frac{\sqrt{Z}}{\sqrt{v_0}}+\frac{\sqrt{\sum_{t=0}^{T-1}\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)-\nabla_x f(x_t,y_t)\right\|^2}}{\sqrt{v_0}}\right)\right]\\
&\leq 1+2\log\left(1+\frac{\mathbb{E}[\sqrt{Z}]}{\sqrt{v_0}}+\frac{\mathbb{E}\left[\sqrt{\sum_{t=0}^{T-1}\left\|\frac{1}{M}\sum_i \nabla_x f(x_t,y_t;\xi_t^i)-\nabla_x f(x_t,y_t)\right\|^2}\right]}{\sqrt{v_0}}\right)\\
&\leq 1+2\log\left(1+\frac{\mathbb{E}[\sqrt{Z}]}{\sqrt{v_0}}+\frac{\sqrt{\sum_{t=0}^{T-1}\sigma^2/M}}{\sqrt{v_0}}\right)\leq 1+2\log\left(1+\frac{\mathbb{E}[\sqrt{Z}]}{\sqrt{v_0}}+\frac{\sqrt{T}\sigma}{\sqrt{v_0 M}}\right),
\end{aligned}
$$

where in the fourth inequality we use $(a+b)^{1/2}\leq a^{1/2}+b^{1/2}$ with $a,b\geq 0$, the fifth and sixth inequalities are from Jensen's inequality. Also, by l-smoothness of f,

$$
Q=\mathbb{E}\sum_{t=0}^{T-1}\frac{\|\nabla_x f(x_t,y_t)-\nabla\Phi(x_t)\|^2}{2\sqrt{v_t+\|\nabla_x f(x_t,y_t)\|^2+\sigma^2/M}}\leq \mathbb{E}\left[\sum_{t=0}^{T-1}\frac{l^2\|y_t-y^*(x_t)\|^2}{2\sqrt{v_0}}\right]\triangleq\mathcal{E}. \tag{5.7}
$$

Also,

$$
\begin{aligned}
C&\geq\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{\|\nabla_x f(x_t,y_t)\|^2}{\sqrt{v_0+\sum_{k=0}^{T-2}\left\|\frac{1}{M}\sum_i \nabla_x f(x_k,y_k;\xi_k^i)\right\|^2+\sum_{j=0}^{T-1}\|\nabla_x f(x_j,y_j)\|^2+\sigma^2/M}}\right]\\
&\geq\sum_{t=0}^{T-1}\mathbb{E}\left[\frac{\|\nabla_x f(x_t,y_t)\|^2}{\sqrt{v_0+3\sum_{j=0}^{T-1}\|\nabla_x f(x_j,y_j)\|^2+2\sum_{k=0}^{T-1}\|\nabla_x f(x_k,y_k)-\frac{1}{M}\sum_i \nabla_x f(x_k,y_k;\xi_k^i)\|^2+\sigma^2/M}}\right]\\
&\geq\mathbb{E}\left[\frac{Z}{\sqrt{v_0+3Z+2\sum_{k=0}^{T-1}\|\nabla_x f(x_k,y_k)-\frac{1}{M}\sum_i \nabla_x f(x_k,y_k;\xi_k^i)\|^2+\sigma^2/M}}\right]\\
&\geq\frac{\left(\mathbb{E}[\sqrt{Z}]\right)^2}{\mathbb{E}\left[\sqrt{v_0+3Z+2\sum_{k=0}^{T-1}\|\nabla_x f(x_k,y_k)-\frac{1}{M}\sum_i \nabla_x f(x_k,y_k;\xi_k^i)\|^2+\sigma^2/M}\right]}\\
&\geq\frac{\left(\mathbb{E}[\sqrt{Z}]\right)^2}{\sqrt{v_0}+3\mathbb{E}[\sqrt{Z}]+\sigma/\sqrt{M}+2\sqrt{\sum_{t=1}^{T-1}\sigma^2/M}}\geq\frac{\left(\mathbb{E}[\sqrt{Z}]\right)^2}{\sqrt{v_0}+3\mathbb{E}[\sqrt{Z}]+2\sigma\sqrt{T}/\sqrt{M}},
\end{aligned}
$$

where in the fourth inequality we use Holder's inequality, i.e. $\mathbb{E}[X^2]\geq\frac{(\mathbb{E}[XY])^2}{\mathbb{E}[Y^2]}$ with

$X=\left(\frac{Z}{\sqrt{v_0+3Z+2\sum_{k=0}^{T-1}\|\nabla_x f(x_k,y_k)-\frac{1}{M}\sum_i \nabla_x f(x_k,y_k;\xi_k^i)\|^2+\sigma^2/M}}\right)^{1/2}$ and

$Y=\left(v_0+3Z+2\sum_{k=0}^{T-1}\|\nabla_x f(x_k,y_k)-\frac{1}{M}\sum_i \nabla_x f(x_k,y_k;\xi_k^i)\|^2+\sigma^2/M\right)^{1/4}$, and in the fifth

inequality we use $(a+b)^{1/2} \le a^{1/2} + b^{1/2}$ and Jensen's inequality. Plugging the bounds for C, D and Q into (5.6),

$$\begin{aligned}
&\frac{\left(\mathbb{E}[\sqrt{Z}]\right)^2}{\sqrt{v_0} + 3\mathbb{E}[\sqrt{Z}] + 2\sigma\sqrt{T}/\sqrt{M}} \\
&\le \frac{2\left(\Phi(x_0) - \min_x \Phi(x)\right)}{\eta} + \left(\frac{4\sigma}{\sqrt{M}} + 2\kappa l\eta\right)\left[1 + 2\log\left(1 + \frac{\mathbb{E}[\sqrt{Z}]}{\sqrt{v_0}} + \frac{\sigma\sqrt{T}}{\sqrt{v_0}\sqrt{M}}\right)\right] + \mathcal{E}.
\end{aligned} \tag{5.8}$$

Now we want to solve for $\mathbb{E}[\sqrt{Z}]$. Denote $\Delta = \Phi(x_0) - \min_x \Phi(x)$. By Proposition 3, we have

$$\mathbb{E}[\sqrt{Z}] \le \frac{\sqrt{v_0}}{3} + \frac{432\Delta^2}{\eta^2} + \frac{2\sigma\sqrt{T}}{3\sqrt{M}} + 432\left(1 + \frac{32}{\sqrt{v_0}}\right)\left(\kappa^2 l^2 \eta^2 + \frac{4\sigma^2}{M}\right) + 108\mathcal{E}^2.$$

We plug this loose upper bound into the logarithmic term on the right hand side of (5.8) and denote the right hand side as $A + \mathcal{E}$. Then we solve the inequality

$$\frac{\left(\mathbb{E}[\sqrt{Z}]\right)^2}{\sqrt{v_0} + 2\mathbb{E}[\sqrt{Z}] + 2\sigma\sqrt{T}/\sqrt{M}} \le A + \mathcal{E},$$

which gives rise to

$$\mathbb{E}[\sqrt{Z}] \le 2(A + \mathcal{E}) + \left(v_0^{\frac{1}{4}} + 2\sigma^{\frac{1}{2}} T^{\frac{1}{4}} M^{-\frac{1}{4}}\right)\sqrt{A + \mathcal{E}}. \tag{5.9}$$

Note that

$$A = \frac{2\Delta}{\eta} + \left(\frac{4\sigma}{\sqrt{M}} + 2\kappa l\eta\right)\left[1 + 2\log\left(\mathrm{Poly}\left(T, \mathcal{E}, \frac{\Delta}{\eta}, \frac{\sigma}{\sqrt{M}}, \kappa l\eta, v_0, \frac{1}{v_0}\right)\right)\right].$$

□

PROOF FOR THEOREM 5.3.4 Now we state Theorem 5.3.4 in a more detailed way.

Theorem 5.5.5 (deterministic). *Suppose we have a linearly-convergent subroutine $\mathcal{A}$ for maximizing any strongly concave function $h(\cdot)$:*

$$\|y^k - y^*\|^2 \le a_1(1 - a_2)^k \|y^0 - y^*\|^2$$

where y^k is k-th iterate, y^* is the optimal solution, and $a_1 > 0$ and $0 < a_2 < 1$ are constants that can depend on the parameters of h.

Under the same setting as Theorem 5.3.1 with $\sigma = 0$, for Algorithm 11 with subroutine $\mathcal{A}$ under criterion I: $\|y_t - Proj_{\mathcal{Y}}(y_t + \nabla_y f(x_t, y_t))\|^2 \le \frac{1}{t+1}$, and $M = 1$, there exists $t^* \le \widetilde{O}\left(\epsilon^{-2}\right)$ such that (x_{t^*}, y_{t^*}) is an ϵ-stationary point. Therefore, the total gradient complexity is $\widetilde{O}\left(\epsilon^{-2}\right)$.

Proof. For convenience, we denote $G_y(x, y) = \|y - \mathrm{Proj}_{\mathcal{Y}}(y + \nabla_y f(x, y))\|$ as the gradient mapping about y at (x, y). From Theorem 3.1 in [Pang, 1987] and Lemma 10.10 in [Beck,

2017], we have $\frac{\mu}{l+1}\|y-y^*(x)\| \le \|G_y(x,y)\| \le (2+l)\|y-y^*(x)\|$. With criterion I, $\mathcal{E}$ can be bounded as the following

$$\mathcal{E} \le \mathbb{E}\left[\sum_{t=0}^{T-1} \frac{l^2(l+1)^2\|G_y f(x_t,y_t)\|^2}{2\mu^2\sqrt{v_0}}\right] \le \frac{\kappa^2(l+1)^2}{2\sqrt{v_0}}\sum_{t=0}^{T-1}\frac{1}{t+1} \le \frac{\kappa^2(l+1)^2(1+\log T)}{2\sqrt{v_0}},$$

where in the first inequality we use the strong concavity. By setting $\sigma = 0$ and $M = 1$ in Theorem 5.3.1, we have

$$\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2 \le \frac{4(A+\mathcal{E})^2}{T} + \frac{\sqrt{v_0}(A+\mathcal{E})}{T},$$

where $A+\mathcal{E} = \widetilde{\mathscr{O}}\left(\frac{\Phi(x_0)-\min_x\Phi(x)}{\eta} + 2\sigma + \kappa l\eta + \frac{\kappa^2(l+1)^2}{\sqrt{v_0}}\right)$. We use $\mathscr{O}(\cdot)$ to include the problem parameters in $O(\cdot)$, and similarly $\widetilde{\mathscr{O}}(\cdot)$ ignores the logarithmic terms. Second, we need to compute the inner-loop complexity. At $(t+1)$-th inner loop, we need to bound the initial distance from y_t to the optimal y w.r.t x_{t+1}.

$$\begin{aligned}\|y_t - y^*(x_{t+1})\|^2 &\le 2\|y_t - y^*(x_t)\|^2 + 2\|y^*(x_t) - y^*(x_{t+1})\|^2\\ &\le \frac{2(l+1)^2}{\mu^2}\|G_y f(x_t,y_t)\|^2 + 2\kappa^2\|x_t - x_{t+1}\|^2\\ &\le \frac{2(l+1)^2}{\mu^2}\cdot\frac{1}{t+1} + \frac{2\kappa^2\eta^2}{v_{t+1}}\|\nabla_x f(x_t,y_t)\|^2 \le \frac{2(l+1)^2}{\mu^2} + 2\kappa^2\eta^2,\end{aligned}$$

where in the second inequality we use Theorem 5.5.1, and in the third we use x_{t+1} update rule. Therefore subroutine $\mathcal{A}$ takes $O\left(\frac{1}{a_2}\log(1/t)\right)$ iterations to find y_{t+1} such that $\|G_y(x_{t+1},y_{t+1})\|^2 \le (2+l)^2\|y_{t+1}-y^*(x_{t+1})\|^2 \le \frac{1}{t+2}$. Then we note that

$$\begin{aligned}\sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2 + \|y_t - y^*(x_t)\|^2 &\le \sum_{t=0}^{T-1}\|\nabla_x f(x_t,y_t)\|^2 + \frac{(l+1)^2}{\mu^2}\|G_y f(x_t,y_t)\|^2\\ &\le 4(A+\mathcal{E})^2 + \sqrt{v_0}(A+\mathcal{E}) + \frac{(l+1)^2}{\mu^2}(1+\log T).\end{aligned}$$

So there exists $t \le \widetilde{\mathscr{O}}\left(\left((A+\mathcal{E})^2 + \sqrt{v_0}(A+\mathcal{E}) + (\kappa^2+1/\mu^2)\right)\epsilon^2\right)$ such that $\|\nabla_x f(x_t,y_t)\| \le \epsilon$ and $\|y_t - y^*(x_t)\| \le \epsilon$. Therefore the total complexity is $\widetilde{\mathscr{O}}\left(\left(\frac{(A+\mathcal{E})^2}{a_2} + \frac{\sqrt{v_0}(A+\mathcal{E})}{a_2} + \frac{(l+1)^2}{\mu^2 a_2}\right)\epsilon^{-2}\right)$ with $A+\mathcal{E} = \widetilde{\mathscr{O}}\left(\frac{\Phi(x_0)-\min_x\Phi(x)}{\eta} + 2\sigma + \kappa l\eta + \frac{\kappa^2(l+1)^2}{\sqrt{v_0}}\right)$.

□

Remark 5.5.6. *As long as we use the stopping criterion $\|y_t - Proj_{\mathcal{Y}}(y_t + \nabla_y f(x_t,y_t))\|^2 \le \frac{1}{t+1}$, the exact same oracle complexity as above can be attained for the primal variable, regardless of the subroutine choice. The convergence rate of the subroutine (not necessarily linear rate) will only affect the oracle complexity of the dual variable.*

PROOF FOR THEOREM 5.3.6 Now we state Theorem 5.3.6 in a more detailed way. Here we consider more general subroutines with $\widetilde{O}(1/k)$ convergence rate. When the subroutine

has the convergence rate $O(1/k)$ without additional logarithmic terms, it reduces to the setting of Theorem 5.3.6. The proof of the theorem relies on Theorem 5.5.4.

Theorem 5.5.7 (stochastic)**.** *Suppose we have a sub-linearly-convergent subroutine $\mathcal{A}$ for maximizing any strongly concave function $h(\cdot)$: after $K = k\log^p(k) + 1$ iterations*

$$\mathbb{E}\|y^K - y^*\|^2 \leq \frac{b_1\|y^0 - y^*\|^2 + b_2}{k},$$

where y^k is k-th iterate, y^* is the optimal solution, $p \in \mathbb{N}$ is an arbitrary non-negative integer and $b_1, b_2 > 0$ are constants that can depend on the parameters of h.

Under the same setting as Theorem 5.3.1, for Algorithm 11 with $M = \epsilon^{-2}$ and subroutine $\mathcal{A}$ under the stopping criterion: at t-th inner loop the subroutine stops after $t\log^p(t) + 1$ steps, there exists $t^* \leq \widetilde{O}\left(\epsilon^{-2}\right)$ such that (x_{t^*}, y_{t^*}) is an ϵ-stationary point. Therefore, the total stochastic gradient complexity is $\widetilde{O}\left(\epsilon^{-4}\right)$.

Proof. First we note that

$$\|y_t - y^*(x_{t+1})\|^2 \leq 2\|y_t - y^*(x_t)\|^2 + 2\|y^*(x_t) - y^*(x_{t+1})\|^2 \leq 2\|y_t - y^*(x_t)\|^2 + 2\kappa^2\eta^2.$$

By the convergence guarantee of subroutine $\mathcal{A}$, after $t\log^p(t) + 1$ inner loop steps, it outputs

$$\mathbb{E}\|y_{t+1} - y^*(x_{t+1})\|^2 = \frac{b_1\|y_t - y^*(x_{t+1})\|^2 + b_2}{t} \leq \frac{2b_1\|y_t - y^*(x_t)\|^2 + 2\kappa^2\eta^2 b_1 + b_2}{t}. \tag{5.10}$$

Taking expectation of both sides and by Theorem 5.5.4, we have

$$\mathbb{E}\sum_{t=0}^{T}\|y_t - y^*(x_t)\|^2 \leq b_3(1 + \log T) + b_3 e^{2b_1} + X_0 e^{2b_1}, \tag{5.11}$$

with $b_3 = 2\kappa^2\eta^2 b_1 + b_2$ and X_0 denotes $\|y_0 - y^*(x_0)\|^2$. Then

$$\mathcal{E} = \frac{l^2}{2\sqrt{v_0}}\mathbb{E}\sum_{t=0}^{T-1}\|y_t - y^*(x_t)\|^2 \leq \frac{l^2}{2\sqrt{v_0}}\left[b_3(1 + \log T) + b_3 e^{2b_1} + X_0 e^{2b_1}\right].$$

By setting $M = \epsilon^{-2}$ in Theorem 5.3.1, we have

$$\mathbb{E}\left[\sqrt{\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla_x f(x_t, y_t)\|^2}\right] \leq \frac{2(A + \mathcal{E})}{\sqrt{T}} + \frac{v_0^{\frac{1}{4}}\sqrt{A + \mathcal{E}}}{\sqrt{T}} + \frac{2\sqrt{(A + \mathcal{E})\sigma\epsilon}}{T^{\frac{1}{4}}},$$

where $A = \widetilde{\mathcal{O}}\left(\frac{\Phi(x_0) - \min_x \Phi(x)}{\eta} + \left(\frac{2\sigma}{\sqrt{M}} + \kappa l\eta\right)(1 + b_1)\right)$. Therefore,

$$\begin{aligned}
&\mathbb{E}\left[\sqrt{\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla_x f(x_t, y_t)\|^2}\right] + \sqrt{\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\|y_t - y^*(x_t)\|^2\right]} \\
&\leq \frac{2(A + \mathcal{E})}{\sqrt{T}} + \frac{v_0^{\frac{1}{4}}\sqrt{A + \mathcal{E}}}{\sqrt{T}} + \frac{2\sqrt{(A + \mathcal{E})\sigma\epsilon}}{T^{\frac{1}{4}}} + \frac{\sqrt{b_3(1 + \log T) + b_3 e^{2b_1} + X_0 e^{2b_1}}}{\sqrt{T}}.
\end{aligned}$$

By setting the right-hand side to ϵ, we need

$$T = \widetilde{\mathscr{O}}\left(\left((A+\mathcal{E})^2 + \sqrt{v_0}(A+\mathcal{E})(1+\sigma) + b_3 + (b_3+X_0)e^{2b_1}\right)\epsilon^{-2}\right)$$

outer-loop iterations. Since $M = \epsilon^{-2}$, the sample complexity for x is $T\epsilon^{-2} = \widetilde{O}(\epsilon^{-4})$. Since the inner loop iteration is at most $T\log^p T + 1$, the sample complexity for y is $T^2\log^p T + T = \widetilde{O}(\epsilon^{-4})$.

□

Remark 5.5.8. *The same sample complexity for the primal variable can be attained as above, as long as (5.10) holds. The choice for the subroutine will affect the number of samples needed to achieve (5.10), and therefore the sample complexity for the dual variable. Although the complexity above includes an exponential term in b_1, we note that $b_1 = 0$ in many subroutines for strongly-convex objectives [Cutkosky and Boahen, 2017, Rakhlin et al., 2012, Lacoste-Julien et al., 2012].*

B. Proofs for Generalized AdaGrad

PROOF OF THEOREM 5.3.8

Proof. We separate the proof into three parts.

PART 1. From the update of Algorithm 12, we have for any $x \in \mathcal{X}$

$$\|x_{t+1} - x\|^2 = \left\|x_t - \frac{\eta}{v_{t+1}^\alpha}g_t - x\right\|^2 = \|x_t - x\|^2 + \frac{\eta^2}{v_{t+1}^{2\alpha}}\|g_t\|^2 - \frac{2\eta}{v_{t+1}^\alpha}\langle g_t, x_t - x\rangle.$$

Multiple each side by v_{t+1}^α,

$$v_{t+1}^\alpha\|x_{t+1} - x\|^2 = v_{t+1}^\alpha\|x_t - x\|^2 + \frac{\eta^2}{v_{t+1}^\alpha}\|g_t\|^2 - 2\eta\langle g_t, x_t - x\rangle.$$

By strong convexity,

$$f_t(x_t) - f_t(x) \le \langle g_t, x_t - x\rangle - \frac{\mu}{2}\|x_t - x\|^2.$$

Plug it into the previous inequality,

$$v_{t+1}^\alpha\|x_{t+1} - x\|^2 \le v_{t+1}^\alpha\|x_t - x\|^2 + \frac{\eta^2}{v_{t+1}^\alpha}\|g_t\|^2 - 2\eta[f_t(x_t) - f_t(x^*)] - \eta\mu\|x_t - x\|^2.$$

Telescope from $t = 0$ to $T-1$,

$$2\eta\sum_{t=0}^{T-1}[f_t(x_t) - f_t(x)] \le v_1^\alpha\|x_0 - x\|^2 - v_T^\alpha\|x_T - x\|^2 - \sum_{t=1}^{T-1}\left[v_t^\alpha - v_{t+1}^\alpha + \eta\mu\right]\|x_t - x\|^2 + \sum_{t=0}^{T-1}\frac{\eta^2}{v_{t+1}^\alpha}\|g_t\|^2. \tag{5.12}$$

PART II. In the part, we focus on the second term on the right hand side of the previous inequality. For convenience, we denote

$$B_t = v_{t+1}^\alpha - v_t^\alpha - \eta\mu.$$

Denote set $S = \{t : B_t > 0\}$. We will first bound the number of t for which the coefficient B_t is positive, i.e., $|S|$, for the case $0 < \alpha < 1$. We note that

$$\begin{aligned} B_t &= (v_t + \|g_t\|^2)^\alpha - v_t^\alpha - \eta\mu = v_t^\alpha \left[\left(\frac{v_t + \|g_t\|^2}{v_t} \right)^\alpha - 1 \right] - \eta\mu \\ &\le v_t^\alpha \left(1 + \alpha \frac{\|g_t\|^2}{v_t} - 1 \right) - \eta\mu = \frac{\alpha \|g_t\|^2}{v_t^{1-\alpha}} - \eta\mu, \end{aligned} \tag{5.13}$$

where in the inequality we apply Bernoulli's inequality, i.e., $(1+x)^r \le 1 + rx$ with $0 \le r \le 1$ and $x \ge -1$. If B_t is positive, it leads to

$$B_t > 0 \quad \iff \quad \|g_t\|^2 > \frac{\eta\mu}{\alpha} v_t^{1-\alpha} \tag{5.14}$$

$$\implies \quad \|g_t\|^2 > \frac{\eta\mu}{\alpha} v_0^{1-\alpha} \tag{5.15}$$

This means $\|g_t\|$ is not small once we observe $B_t > 0$. Since $\|g_t\|^2 \le G^2$, if the right hand side of (5.14) is larger or equal to G^2, then B_t can not be positive, i.e.

$$\frac{\eta\mu}{\alpha} v_t^{1-\alpha} \ge G^2 \quad \iff \quad v_t \ge \left(\frac{\alpha G^2}{\eta\mu} \right)^{\frac{1}{1-\alpha}}.$$

On the other hand, because $v_{t+1} = v_t + \|g_t\|^2$, (5.15) implies that once we observe $B_t > 0$, v_t will increase by at least $\frac{\eta\mu}{\alpha} v_0^{1-\alpha}$. Therefore, it can be positive for only finite times, i.e.,

$$|S| \le \frac{\left(\frac{\alpha G^2}{\eta\mu} \right)^{\frac{1}{1-\alpha}}}{\frac{\eta\mu}{\alpha} v_0^{1-\alpha}} = \frac{\alpha (\alpha G^2)^{\frac{1}{1-\alpha}}}{(\eta\mu)^{\frac{2-\alpha}{1-\alpha}} v_0^{1-\alpha}}. \tag{5.16}$$

Even when B_t is positive, its value is bounded above from (5.13),

$$B_t \le \frac{\alpha \|g_t\|^2}{v_t^{1-\alpha}} - \eta\mu \le \frac{\alpha G^2}{v_0^{1-\alpha}}. \tag{5.17}$$

Now it is left to discuss the case $\alpha = 1$. When $\alpha = 1$,

$$B_t = -v_t + v_{t+1} - \eta\mu \le \|g_t\|^2 - \eta\mu \le G^2 - \eta\mu.$$

Therefore, when $\eta \ge \frac{G^2}{\mu}$, we have $B_t \le 0$ for all t.

PART III. In this part we wrap up everything for two cases: i) $0 < \alpha \le 1$; ii) $\alpha = 1$. From equation (5.12),

$$2\eta \sum_{t=0}^{T-1} [f_t(x_t) - f_t(x)] \le v_1^\alpha \mathcal{D}^2 + \sum_{t \in S} B_t \mathcal{D}^2 + \eta^2 \sum_{t=0}^{T-1} \frac{1}{v_{t+1}^\alpha} \|g_t\|^2 \tag{5.18}$$

CASE $0 < \alpha \leq 1$. By Lemma 5.5.2, (5.16) and (5.17),

$$2\eta \sum_{t=0}^{T-1} [f_t(x_t) - f_t(x)] \leq v_1^{\alpha} \mathcal{D}^2 + \sum_{t \in S} B_t \mathcal{D}^2 + \frac{\eta^2}{1-\alpha} v_{t+1}^{1-\alpha}$$
$$\leq (v_0 + G^2)^{\alpha} \mathcal{D}^2 + \frac{\alpha (\alpha G^2)^{\frac{2-\alpha}{1-\alpha}}}{(\eta\mu)^{\frac{2-\alpha}{1-\alpha}} v_0^{2-2\alpha}} + \frac{\eta^2}{1-\alpha} v_{t+1}^{1-\alpha}.$$

CASE $\alpha = 1$. We have $B_t \leq 0$ for all t as $\eta \geq \frac{G^2}{\mu}$. Then by Lemma 5.5.2,

$$2\eta \sum_{t=0}^{T-1} [f_t(x_t) - f_t(x)] \leq (v_0 + G^2)\mathcal{D}^2 + \eta^2 \log \left(\frac{\sum_{t=0}^{T-1} \|g_t\|^2}{v_0} \right).$$

□

Remark 5.5.9. *We note that the regret bounds contain a constant term $\mu^{-\frac{1}{1-\alpha}}$, which increases exponentially as α approaches 1. However, such term is common even in the convergence result of SGD with a non-adaptive stepsize $\frac{\eta}{t^{\alpha}}$ in strongly-convex stochastic optimization; e.g., Theorem 1 in [Moulines and Bach, 2011] and Theorem 31 in [Fontaine et al., 2021] both contain a term that will not diminish before $\Theta\left(\mu^{-\frac{1}{1-\alpha}}\right)$ iterations.*

6

LIMIT OF UNTUNED SGD AND POWER OF ADAPTIVE METHODS

The classical analysis of Stochastic Gradient Descent (SGD) with polynomially decaying stepsize $\eta_t = \eta/\sqrt{t}$ relies on well-tuned η depending on problem parameters such as Lipschitz smoothness constant, which is often unknown in practice. In this work, we prove that SGD with arbitrary $\eta > 0$, referred to as *untuned SGD*, still attains an order-optimal convergence rate $\widetilde{\mathcal{O}}(T^{-1/4})$ in terms of gradient norm for minimizing smooth objectives. Unfortunately, it comes at the expense of a catastrophic exponential dependence on the smoothness constant, which we show is unavoidable for this scheme even in the noiseless setting. We then examine three families of adaptive methods — Normalized SGD (NSGD), AMSGrad, and AdaGrad — unveiling their power in preventing such exponential dependency in the absence of information about the smoothness parameter and boundedness of stochastic gradients. Our results provide theoretical justification for the advantage of adaptive methods over untuned SGD in alleviating the issue with large gradients.

6.1 OVERVIEW

In this chapter, we study the stochastic optimization problem of the form:

$$\min_{x\in\mathbb{R}^d} f(x) = \mathbb{E}_{\xi\sim P}\left[F(x;\xi)\right],$$

where P is an unknown probability distribution, and $f:\mathbb{R}^d\to\mathbb{R}$ is an ℓ-Lipschitz smooth function and can be non-convex. In the context of machine learning, ξ may represent an individual training sample from the data distribution P, and x denotes the weights of the model.

Stochastic Gradient Descent (SGD), originated from the seminal work [Robbins and Monro, 1951], performs the following update iteratively:

$$x_{t+1} = x_t - \eta_t \nabla F(x_t;\xi_t),$$

where $\eta_t > 0$ is some stepsize and $\nabla F(x_t;\xi_t)$ is an unbiased stochastic gradient. SGD has shown remarkable empirical success in many modern machine learning applications, e.g., [Bengio, 2009, Sutton and Barto, 2018]. Its efficiency is usually attributed to its cheap per iteration cost and the ability to operate in an online fashion, making it suitable for large-scale problems. However, empirical evidence also reveals undesirable behaviors of SGD, often related to challenges in selecting appropriate stepsizes. In particular, a

Algorithms	Upper bound; deterministic	Lower bound; deterministic	Upper bound; stochastic	Lower bound; stochastic
SGD (Eq. 6.1) $\eta_t = \frac{\eta}{\sqrt{t+1}}$	$\widetilde{\mathcal{O}}\left((4e)^{2(\eta\ell)^2}\epsilon^{-4}\right)$ [Thm. 6.3.1, 6.5.1]	$\Omega\left((8e)^{\eta^2\ell^2/8}\epsilon^{-4}\right)$ [Thm. 6.3.3]	$\widetilde{\mathcal{O}}\left((4e)^{2(\eta\ell)^2}\epsilon^{-4}\right)$ [Thm. 6.3.1, 6.5.1]	$\Omega\left((8e)^{\eta^2\ell^2/8}\epsilon^{-4}\right)$ [Thm. 6.3.3]
NSGD (Alg. 16) $\eta_t = \frac{\gamma}{\sqrt{t+1}\|g(x_t,\xi_t)\|}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-2}\right)$ [Cutkosky and Mehta, 2020] & [Prop. 5]	$\Omega\left(\epsilon^{-2}\right)$ [Carmon et al., 2020]	N/A due to lower bound	Nonconvergent [Thm. 6.4.1]
NSGD-M (Alg. 13) $\eta_t = \frac{\gamma}{(t+1)^\alpha\|g_t\|}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-2}\right)$, $\alpha = 1/2$ [Cutkosky and Mehta, 2020] & [Prop. 5]	$\Omega\left(\epsilon^{-2}\right)$ [Carmon et al., 2020]	$\widetilde{\mathcal{O}}\left(\epsilon^{-4}\right)$, $\alpha = 3/4$ [Cutkosky and Mehta, 2020] & [Prop. 13]	$\Omega\left(\epsilon^{-4}\right)$ [Arjevani et al., 2022]
AMSGrad-norm (Alg. 14) $\eta_t = \frac{\gamma}{\sqrt{(t+1)\hat{v}_{t+1}^2}}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-4}\right)$ [Thm. 6.4.3, 6.5.6]	$\Omega\left(\epsilon^{-4}\right)$ [Thm. 6.5.9]	N/A due to lower bound	$\Omega\left(\epsilon^{-\frac{2}{1-\zeta}}\right)$ $\forall\zeta \in (0.5, 1)$ [Thm. 6.4.2]
AdaGrad-norm (Alg. 15) $\eta_t = \frac{\eta}{\sqrt{v_0^2+\sum_{k=0}^t\|g(x_k,\xi_k)\|^2}}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-2}\right)$ [Yang et al., 2022a] & [Prop. 6.4.4]	$\Omega\left(\epsilon^{-2}\right)$ [Carmon et al., 2020]	$\widetilde{\mathcal{O}}\left(\epsilon^{-4}\right)$ [Yang et al., 2022a] & [Prop. 6.4.4]	$\Omega\left(\epsilon^{-4}\right)$ [Arjevani et al., 2022]

TABLE 6.1: Complexities of finding an ϵ-stationary point for SGD, NSGD [Nesterov, 1984], NSGD-M [Cutkosky and Mehta, 2020], AMSGrad-norm (norm version of AMSGrad [Reddi et al., 2019]), and AdaGrad-norm [Streeter and McMahan, 2010]. We only assume f is ℓ-smooth, and unbiased stochastic gradients have bounded variance σ^2. Hyper-parameters (e.g., γ and η) are untuned. Here, $\widetilde{\mathcal{O}}$ and Ω hide polynomial terms in problem parameters and hyper-parameters. The bounds are with respect to specific algorithms and stepsizes, and lower bounds for general first-order methods still hold [Carmon et al., 2020, Arjevani et al., 2022]. We denote the effective stepsize at iteration t as η_t.

number of works report the *gradient explosion* effect [Bengio et al., 1994, Pascanu et al., 2013, Goodfellow et al., 2016] during the initial phase of training, which may eventually lead to divergence or prohibitively slow convergence. The phenomenon is also observed in our experiments (see Figure 6.1(b)) when the stepsize is poorly chosen. Unfortunately, this phenomenon is not well understood from a theoretical point of view. The classical analysis of SGD in the smooth non-convex case [Ghadimi and Lan, 2013], prescribes to select a non-increasing sequence of stepsizes $\{\eta_t\}_{t\geq 1}$ with $\eta_1 < 2/\ell$. In particular, the choice $\eta_t = 1/(\ell\sqrt{t})$, guarantees[1] to find a point x with $\mathbb{E}\left[\|\nabla f(x)\|\right] \leq \epsilon$ after $\mathcal{O}\left(\epsilon^{-4}\right)$ stochastic gradient calls, which is also known to be unimprovable in the smooth non-convex setting unless additional assumptions are made [Arjevani et al., 2022, Drori and Shamir, 2020].

However, the bound on the smoothness parameter ℓ is usually not readily available for practitioners, and the limited computing power usually refrains them from exhaustive tuning to find the best stepsize. It is therefore important to provide theoretical understanding for SGD with an arbitrary stepsize (which we refer to as *untuned SGD*) that is agnostic to the problem parameter. The following intriguing question remains elusive in the stochastic optimization literature:

1 Given access to unbiased stochastic gradient oracle with bounded variance.

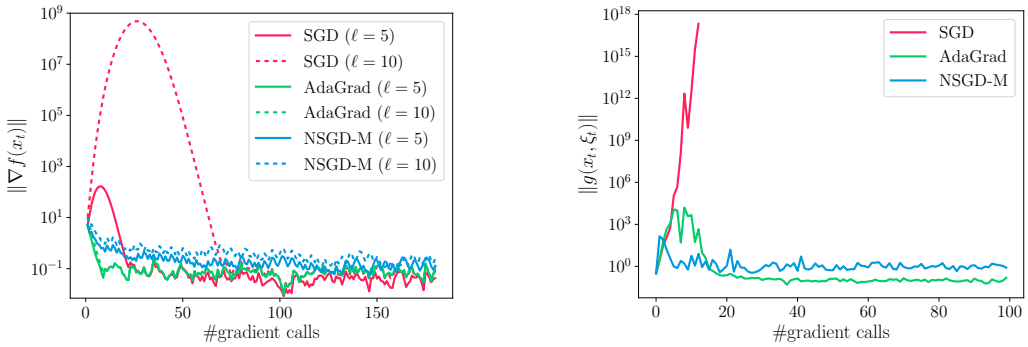
How does untuned SGD with decaying stepsize $\eta_t = \eta/\sqrt{t}$ perform when η is independent of the smoothness parameter? How to explain the undesirably large gradients encountered in training with SGD?

Recently, there has been a surge of interest in adaptive gradient methods such as Adam [Kingma and Ba, 2015], RMSProp [Hinton et al., 2012], AdaDelta [Zeiler, 2012], AMSGrad [Reddi et al., 2019], AdaGrad [Duchi et al., 2011], Normalized SGD [Hazan et al., 2015] and many others. These methods automatically adjust their stepsizes based on past stochastic gradients rather than using pre-defined iteration-based schedules. Empirically, they are observed to converge faster than SGD and mitigate the issue of gradient explosion across a range of problems, even without explicit knowledge of problem-specific parameters [Kingma and Ba, 2015, Liu et al., 2020b, Pascanu et al., 2013]. Figure 6.1(a) provides a basic illustration of performance differences between SGD with $\eta_t = 1/\sqrt{t}$ stepsizes and adaptive schemes such as AdaGrad and Normalized SGD with momentum (NSGD-M) [Cutkosky and Mehta, 2020]. Notably, when the initial stepsize is too large (compared to $1/\ell$ value), SGD reaches the region with *large gradients*, while adaptive methods do not suffer from such effect. However, the theoretical benefits of adaptive methods over SGD remain unclear. A large number of existing analyses of adaptive methods assume bounded gradients, or even stochastic gradients, precluding not only a fair comparison with SGD whose convergence does not need bounded gradient but also the possibility to explain their benefit when facing gradient explosions. While recent developments show that AdaGrad-type methods [Faw et al., 2022, Yang et al., 2022a] can attain $\widetilde{\mathcal{O}}\left(\epsilon^{-4}\right)$ sample complexity under the same standard assumptions as for SGD analysis, there still lacks a good explanation for the huge performance gap observed in practice despite SGD with well-tuned stepsizes theoretically achieving the lower complexity bound. We will address the following open question:

Can we justify the theoretical benefits of adaptive methods over untuned SGD for smooth non-convex problems without assuming bounded gradients?

Consequently, this work is based on the premise of not assuming bounded gradients and hyper-parameters being independent of problem parameters. The main contributions are as follows:

- We show that untuned SGD with diminishing stepsizes $\eta_t = \eta/\sqrt{t}$ finds an ϵ-stationary point of an ℓ-smooth function within $\widetilde{\mathcal{O}}((\ell^2 + \sigma^4\eta^4\ell^4)(4e)^{2\eta^2\ell^2}\epsilon^{-4})$ iterations for any $\eta > 0$. Here σ^2 corresponds to the variance of the stochastic gradient. Although this classical algorithm converges and has the optimal dependence on ϵ, we further show that the disastrous exponential term in $\eta^2\ell^2$ is not avoidable even when the algorithm has access to exact gradients. This explains its proneness to gradient explosion when



(a) quadratic function (b) neural network

FIGURE 6.1: Comparison of SGD, AdaGrad, and NSGD-M on a quadratic function $f(x) = \ell x^2/2$ and a neural network. SGD employs a diminishing stepsize of $\eta/\sqrt{t}$, while the stepsizes for AdaGrad and NSGD-M are specified in Propositions 6 and 6.4.4, respectively. In the left figure, we set $\eta = 1$ for all methods and test with two different values of ℓ. In the right figure, we train a 3-layer neural network on the MNIST dataset [LeCun, 1998] using cross-entropy loss and set $\eta = 10$.

the problem parameter is unknown. Previous analyses fail to capture this exponential term, since they assume η is well-tuned to be $\Theta(1/\ell)$.

- AMSGrad, proposed to fix the nonconvergence of Adam, is not yet well-understood, with previous analyses depending on *bounded stochastic gradients*. We show that AMSGrad (norm version) is free from exponential constants in the deterministic setting without tuning, in stark contrast with SGD. Surprisingly, in the stochastic setting when the stochastic gradients are unbounded, we show that AMSGrad may converge at an arbitrarily slow polynomial rate. To the best of our knowledge, these are the first results of AMSGrad without assuming bounded gradients.

- To further illuminate the advantages of adaptive methods, we re-examine the results for Normalized Gradient Descent (NGD), Normalized SGD with momentum (NSGD-M) from [Cutkosky and Mehta, 2020] and AdaGrad-norm from [Yang et al., 2022a], considering stepsize independent of the problem parameters similar to untuned SGD. They all achieve near-optimal complexities while shredding off the exponential factor. As a side result, we provide a strong non-convergence result of NSGD without momentum under any bounded stepsizes, which might be of independent interest.

Our findings contribute a fresh understanding of the performance gap between SGD and adaptive methods. Albeit with a near-optimal rate, untuned SGD is vulnerable to gradient explosion and slow convergence due to a large exponential constant in its complexity, which can be circumvented by several adaptive methods. To the best of our knowledge, this substantial difference is unformed in the previous literature, because the majority of analyses for SGD and adaptive methods turn to either well-tuned stepsize based on

problem parameters or the assumption of bounded gradients. Part of our results are summarized in Table 6.1 and full results for a broader range of stepsizes can be found in Table 6.2 in the appendix.

6.1.1 *Related Work*

SGD in nonconvex optimization. Stochastic approximation methods and SGD in particular have a long history of development [Robbins and Monro, 1951, Kiefer and Wolfowitz, 1952, Blum, 1954, Chung, 1954, Nemirovski and Yudin, 1983, Polyak and Juditsky, 1992]. When the objective is ℓ-smooth and the gradient noise has bounded variance σ^2, Ghadimi and Lan [2013] and Bottou et al. [2018] prove that if stepsize $\eta_t = \eta/\sqrt{T}$, where $\eta = \eta(\ell, \sigma^2)$ and T is the total iteration budget, then SGD can find an ϵ-stationary point within $\mathcal{O}(\ell\sigma^2\epsilon^{-4})$ iterations. Similar complexity (up to a logarithmic term) can also be achieved by decaying stepsizes $\eta/\sqrt{t}$ [Ghadimi and Lan, 2013, Drori and Shamir, 2020, Wang et al., 2021]. This result was later shown to be optimal for first-order methods under these assumptions [Arjevani et al., 2022]. Several works consider various relaxations of the stochastic oracle model with bounded variance, for instance, biased oracle [Ajalloeian and Stich, 2021] or expected smoothness [Khaled and Richtárik, 2020]. However, these results also heavily rely on sufficiently small η, e.g., $\eta \leq 1/\ell$, and the convergence behavior in the large η regime is rarely discussed. Remarkably, Lei et al. [2019] characterize the convergence of SGD under individual smoothness and unbiased function values. They consider Robbins-Monro stepsize schemes, which includes η/t^α when $\alpha > 1/2$, and derive $\mathcal{O}(\epsilon^{\frac{-2}{\alpha-1}})$ sample complexity including an exponential dependence on the smoothness parameter. Unlike [Lei et al., 2019], we focus on the standard assumptions and derive better dependency in smoothness constant when $\alpha > 1/2$. Importantly, we further justify that the exponential constants are unavoidable with a lower bound.

Adaptive methods. We focus on methods directly using gradients to adjust stepsize, rather than other strategies like backtracking line search [Armijo, 1966]. Normalized Gradient Descent (NGD) was introduced by [Nesterov, 1984] for quasi-convex functions. Hazan et al. [2015] apply NGD and NSGD with minibatch to the class of locally-quasi-convex functions. Later, Cutkosky and Mehta [2020] and Zhao et al. [2021] prove NSGD with momentum or minibatch can find an ϵ-stationary point in smooth nonconvex optimization with sample complexity $\mathcal{O}(\epsilon^{-4})$. AdaGrad was introduced in the online convex optimization [Duchi et al., 2011, McMahan and Streeter, 2010]. In nonconvex optimization, AdaGrad and its scalar version, AdaGrad-norm [Streeter and McMahan, 2010], achieve competitive convergence rates with SGD [Ward et al., 2020, Li and Orabona, 2019, Kavis et al., 2022a, Li and Orabona, 2020]. RMSProp [Hinton et al., 2012] and Adam [Kingma

and Ba, 2015] use the moving average of past gradients, but may suffer from divergence without hyper-parameter tuning [Reddi et al., 2019]. Recently, it was shown in the finite-sum setting that they converge to a neighborhood, whose size shrinks to 0 by tuning hyper-parameters [Shi and Li, 2021, Zhang et al., 2022]. However, most of the results on AdaGrad and Adam-type algorithms assume both Lipschitz and bounded gradients [Zhou et al., 2018, Chen et al., 2019, Défossez et al., 2020, Ward et al., 2020, Zou et al., 2019]. Very recently, Faw et al. [2022] and Yang et al. [2022a] independently show that AdaGrad-norm converges without assuming bounded gradients and without the need for tuning, attaining a sample complexity of $\widetilde{\mathcal{O}}(\epsilon^{-4})$.

SGD v.s. adaptive methods. Despite similar complexities, adaptive methods typically converge faster than SGD in practice [Brown et al., 2020, Liu et al., 2020d] and are widely used to prevent large gradients [Pascanu et al., 2013, Ginsburg et al., 2019]. Various attempts have been made to theoretically explain these differences. Some suggest that the advantage of adaptive algorithms is their ability to achieve order-optimal rates without knowledge of problem parameters such as smoothness and noise variance [Ward et al., 2020, Levy et al., 2021, Kavis et al., 2019]. Other studies investigate the faster escape from saddle points by adaptive methods [Levy, 2016, Murray et al., 2019, Xie et al., 2022]. The importance of coordinate-wise normalization in Adam has also been highlighted [Balles and Hennig, 2018, Kunstner et al., 2023]. Furthermore, the influence of heavy-tail noise on the performance of adaptive methods is studied [Zhang et al., 2020b]. However, most previous works do not provide an explanation for the faster convergence of adaptive methods in terms of sample complexity. Notably, Zhang et al. [2019a] and Wang et al. [2022] explain the benefits of gradient clipping and Adam under a relaxed smoothness assumption, a setting where SGD with non-adaptive stepsizes may not converge. In contrast, we analyze SGD and several adaptive methods under standard smoothness and noise assumptions, distinguishing it from the recent work of Wang et al. [2022] that focuses on one variant of Adam for finite-sum problems with individual relaxed smoothness and random shuffling.

6.2 PROBLEM SETTING

Throughout this chapter, we focus on minimizing an ℓ-smooth function $f : \mathbb{R}^d \to \mathbb{R}$. We have access to a stochastic gradient oracle that returns $g(x;\xi)$ at any point x, and we make the following standard assumptions in nonconvex optimization.

Assumption 16 (smoothness)**.** *Function $f(x)$ is ℓ-smooth with $\ell > 0$, that is, $\|\nabla f(x_1) - \nabla f(x_2)\| \le \ell\|x_1 - x_2\|$ for any x_1 and $x_2 \in \mathbb{R}^d$.*

Assumption 17 (stochastic gradients). *The stochastic gradient $g(x;\xi)$ is unbiased and has a bounded variance, that is, $\mathbb{E}_\xi[g(x;\xi)] = \nabla f(x)$ and $\mathbb{E}_\xi\left[\|g(x;\xi) - f(x)\|^2\right] \leq \sigma^2$ for any $x \in \mathbb{R}^d$.*

We present the general scheme of SGD with initial point x_0 and a stepsize sequence $\{\eta_t\}_{t=0}^{\infty}$:

$$x_{t+1} = x_t - \eta_t\, g(x_t;\xi_t). \tag{6.1}$$

Some commonly used stepsizes include polynomially and geometrically decaying stepsize, constant stepsize, cosine stepsize, etc. When the stepsize depends on the instantaneous or past gradients, i.e., $\{g(x;\xi_k)\}_{k\leq t}$, we call it adaptive stepsize, namely Normalized SGD [Hazan et al., 2015], AdaGrad [Duchi et al., 2011], Adam [Kingma and Ba, 2015], AMSGrad [Reddi et al., 2019], etc. In some adaptive methods, momentum is also considered, replacing $g(x_t;\xi_t)$ in (6.1) with a moving average m_{t+1} of the past stochastic gradients (see Chapter 6.4 for more details). To set the stage for our analysis, we assume that $f(x_0) - \min_{x\in\mathbb{R}^d} f(x) \leq \Delta$, where Δ represents the initial gap. Given that the function class of interest is nonconvex, we aim to find an ϵ-stationary point x with $\mathbb{E}[\|\nabla f(x)\|] \leq \epsilon$.

6.3 CONVERGENCE OF UNTUNED SGD

In this subchapter, we focus on SGD with the decaying stepsize:

$$\eta_t = \frac{\eta}{\sqrt{t+1}},$$

where $\eta > 0$ is the initial stepsize. Most convergent analysis requires $\eta < 2/\ell$ [Ghadimi and Lan, 2013, Bottou et al., 2018] so that there is "sufficient decrease" in function value after each update, and if η is carefully chosen, it can achieve the near-optimal complexity of $\widetilde{\mathcal{O}}(\ell\epsilon^{-4}\sigma^2)$ [Arjevani et al., 2022]. Nevertheless, as the smoothness parameter is usually unknown, providing guarantees with optimal η or assuming η to be problem-dependent does not give enough insights into practical training with SGD. Hence we are interested in its convergence behavior in both small and large initial stepsize regimes, i.e., $\eta \leq 1/\ell$ and $\eta > 1/\ell$.

Theorem 6.3.1. *Under Assumptions 16 and 17, if we run SGD with stepsize $\eta_t = \frac{\eta}{\sqrt{t+1}}$, where $\eta > 0$,*

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\|^2 \leq \begin{cases} 2A\eta^{-1}T^{-\frac{1}{2}}, & \text{when } \eta \leq 1/\ell, \\ 4\sqrt{2}\ell A(4e)^{\tau}(\pi T)^{-\frac{1}{2}}, & \text{when } \eta > 1/\ell, \end{cases}$$

where $\tau = \lceil \eta^2\ell^2 - 1\rceil$ and $A = \left(\Delta + \frac{\ell\sigma^2\eta^2}{2}(1+\log T)\right)$.

This theorem implies that when the initial stepsize $\eta > 1/\ell$, SGD still converges with a sample complexity of $\widetilde{\mathcal{O}}((\ell^2 + \sigma^4\eta^4\ell^4)(4e)^{2\eta^2\ell^2}\epsilon^{-4})$. Although the dependency in the target accuracy ϵ is near-optimal, it includes a disastrous exponential term in $\eta^2\ell^2$. This is due to polynomially decaying stepsizes: in the first stage before $\tau = \lceil(\eta\ell)^2 - 1\rceil$ iterations, the function value and gradients may keep increasing in expectation until reaching an exponential term in $\eta^2\ell^2$, which is in stark contrast with adaptive methods that we will see in Chapter 6.5.3; in the second stage after $t \geq \tau$, the stepsize is small enough to decrease the function value in expectation at a rate of $1/\sqrt{T}$ up to a small term in σ.

If we pick an arbitrary $\eta = \Theta(1)$, untuned SGD may induce large gradients growing exponentially in ℓ in the first stage, which matches our observation in Figure 6.1. On the other hand, deriving the dependence in hyper-parameter η is essential for assessing the effort required in its tuning: SGD with η that is $c > 1$ times larger than the optimally tuned one can have an $\exp(\text{poly}(c))$ times larger gradient norm in the convergence guarantee. To the best of our knowledge, there is limited study for non-asymptotic analysis of untuned SGD under the same assumptions. Moulines and Bach [2011] study untuned SGD under individual smoothness and convexity assumptions, i.e., $g(x;\xi)$ is Lipschitz continuous and $F(x;\xi)$ is convex almost surely. They show an $\mathcal{O}(1/T^{1/3})$ rate, which is suboptimal in the convex case. Later, Fontaine et al. [2021] provide $\mathcal{O}(1/T^{1/2})$ convergence rate for untuned SGD in the convex setting yet without an explicit dependency in ℓ and η.

Remark 6.3.2. *We focus on the stepsize of the order of $1/\sqrt{t}$, because it is known for SGD to achieve the best dependency in ϵ for nonconvex optimization [Drori and Shamir, 2020] and easier to compare with adaptive stepsizes. We also present the convergence results for more general polynomially decaying stepsizes, i.e., $\eta_t = \frac{\eta}{(t+1)^\alpha}$ with $0 < \alpha < 1$, in Theorem 6.5.1 of the appendix. There exists a trade-off between convergence speed $\mathcal{O}(1/T^{\frac{1-\alpha}{2}})$ and the exponential term in $(\eta\ell)^{1/\alpha}$ for $\alpha \in [1/2, 1)$. Intuitively, larger α leads to a shorter time in adapting to $1/\ell$ stepsize but a slower convergence rate. We do not consider constant stepsize, i.e., $\alpha = 0$, because it is well known to diverge even in the deterministic setting if the stepsize is agnostic to the problem parameter [Nesterov, 2013, Ahn et al., 2022].*

The question arises as to whether the exponential term is necessary. In the following, we provide a lower bound for SGD under this choice of stepsize.

Theorem 6.3.3. *Fixing $T \geq 1, \eta > 0, \ell > 0$ and $\Delta > 0$ that $\eta\ell \geq 5$, there exists a ℓ-smooth function $f : \mathbb{R} \to \mathbb{R}$ and an initial point x_0 with $f(x_0) - f^* \leq \Delta$ such that if we run Gradient Descent with stepsize $\eta_t = \frac{\eta}{\sqrt{t+1}}$, then for $t \leq t_0 = \lfloor \eta^2\ell^2/16 - 1\rfloor$,*

$$|\nabla f(x_t)| \geq \sqrt{\frac{2\ell\Delta}{3\sqrt{t}}}\,(8e)^{t/2} \text{ and } |\nabla f(x_{t_0})| \geq \sqrt{\frac{8\Delta}{3\eta}}\,(8e)^{\eta^2\ell^2/32-4};$$

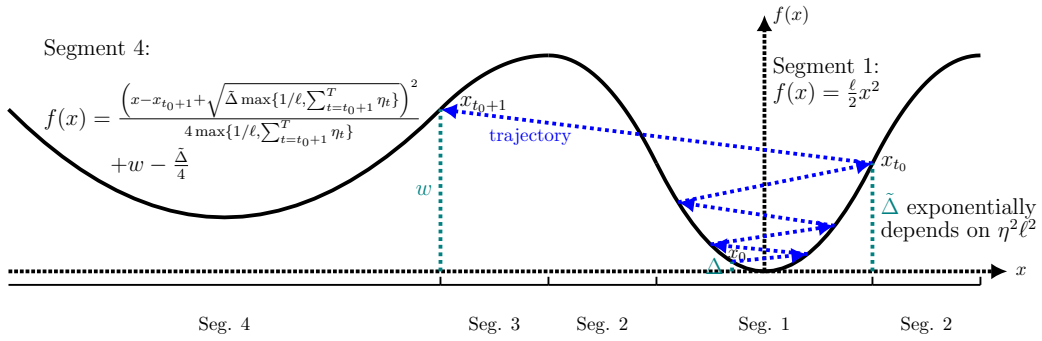


FIGURE 6.2: Demonstration of the constructed function for proving the lower bound.

if $T > t_0$, *then for* $t_0 < t \le T$,

$$|\nabla f(x_t)| \ge \frac{1}{4}\sqrt{\tilde{\Delta}}\min\left\{\ell^{1/2}, (2\eta)^{-1/2}T^{-1/4}\right\}, \text{ where } \tilde{\Delta} \ge \frac{4}{3\eta\ell}(8e)^{\eta^2\ell^2/16-2}\Delta.$$

This theorem suggests that Gradient Descent with decaying stepsize $\eta/\sqrt{t+1}$ needs at least $\Omega(\eta^{-4}\ell^{-2}(8e)^{\eta^2\ell^2/8}\epsilon^{-4})$ iterations to find an ϵ-stationary point in the large initial stepsize regime. Therefore, it justifies that an exponential term in $\eta^2\ell^2$ multiplied by $1/\sqrt{T}$ is not avoidable even in the deterministic setting. Note that our result is limited to untuned (S)GD with the particular stepsize scheme. It is worth pointing out that the existing lower bounds for first-order methods [Arjevani et al., 2022] and SGD [Drori and Shamir, 2020] do not contain any exponential terms.

We illustrate our hard instance for Theorem 6.3.3 in Figure 6.2, which is one-dimensional. The algorithm starts from a valley of the function $f(x) = \ell x^2/2$, i.e., Segment 1. Because of the large initial stepsize and steep slope, in the first t_0 iterations, Gradient Descent increases the function value as large as $\tilde{\Delta} = \Omega\left((8e)^{\eta^2\ell^2/16}\Delta\right)$. Then the iterate x_{t_0+1} jumps to the top of a very flat valley, i.e., Segment 4, so that Gradient Descent decreases the gradient as slowly as $\Omega(T^{-1/4})$.

Why do not we assume gradients to be bounded? The assumption on bounded gradients is not satisfied even for the simple function $f(x) = \ell x^2/2$. When training neural networks, gradient explosion is often observed [Pascanu et al., 2013, Schmidhuber, 2015], which directly suggests that this assumption is not satisfied or only satisfied with a numerically large constant. In Proposition 7 in the appendix, we also provide a simple proof for the convergence under the additional assumption of bounded gradient, i.e., $\|\nabla f(x)\| \le G$ for all x, attaining a sample complexity of $\widetilde{\mathcal{O}}(\eta^2\ell^2G^4\sigma^2\epsilon^{-4})$ without any information about problem parameters. However, compared with Theorem 6.3.1 and 6.3.3, constant G hides the exponential term. In Figure 6.1, we observe that the gradient bound along the trajectory of non-adaptive stepsize can be much larger than that of adaptive stepsize even if starting from the same initial point, so assuming bounded gradient will obscure the difference between them.

6.4 POWER OF ADAPTIVE METHODS

In this subchapter, we focus on the convergence behaviors of adaptive methods, which adjust their stepsizes based on the observed gradients. In particular, when arriving at a point with a large gradient, adaptive methods automatically decrease their stepsizes to counter the effect of possible gradient increase; to list a few, Normalized SGD [Hazan et al., 2015], AdaGrad [Duchi et al., 2011], Adam [Kingma and Ba, 2015]. Since the analysis for adaptive methods is usually on a case-by-case basis, we will examine three examples – Normalized SGD, AMSGrad-norm, and AdaGrad-norm – to establish a universal observation that they avoid exponential dependency in ℓ without tuning. Although many existing analyses rely on bounded gradients (and function values) or information on problem parameters, we will abandon such assumptions as noted in the previous subchapter. We focus on the norm instead of the coordinate-wise version of adaptive methods, which means each coordinate adopts the same stepsize, because the norm version is usually dimension-independent in the complexity, and is also widely used in both theory and practice [Zhang, 2018, Ling et al., 2022, Li and Orabona, 2019, Leevy and Khoshgoftaar, 2020, Palfinger, 2022, Kavis et al., 2022b].

6.4.1 *Family of Normalized SGD*

Normalized (Stochastic) Gradient Descent [Nesterov, 1984, Hazan et al., 2015], referred to as NGD and NSGD, is one of the simplest adaptive methods. It takes the stepsize in (6.1) to be normalized by the norm of the current (stochastic) gradient:

$$\eta_t = \frac{\gamma_t}{\|g(x_t;\xi_t)\|},$$

where $\{\gamma_t\}_{t\geq 0}$ is a sequence of positive learning rate. Cutkosky and Mehta [2020] and Zhao et al. [2021] show that NSGD with $\gamma_t = \gamma/\sqrt{T}$ can find an $\mathcal{O}(1/\sqrt{T}+\sigma)$-stationary point. In order to compare fairly with untuned SGD with decaying stepsize, we present a modification with decaying $\gamma_t = \gamma/\sqrt{t+1}$ in NSGD.

Proposition 5. *Under Assumption 16 and 17, if we run NSGD with* $\gamma_t = \frac{\gamma}{\sqrt{t+1}}$, *then for any* $\gamma > 0$,

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\| \leq 3\left(\frac{\Delta}{\gamma}+\ell\gamma\log(T)\right)T^{-1/2}+24\sigma.$$

NGD. In the deterministic setting, by Proposition 5, NGD converges to an ϵ-stationary point with a complexity of $\widetilde{\mathcal{O}}((\gamma^{-2}+\gamma^2\ell^2)\epsilon^{-2})$ for any $\gamma > 0$, which importantly does not include any exponential term. Thus, even if the initial stepsize is not small enough, it does not result in a catastrophic gradient explosion.

NSGD. In the stochastic setting, Proposition 5 implies that NSGD can find an ϵ-stationary point only when the noise variance is small enough, i.e., $\sigma \leq \mathcal{O}(\epsilon)$. This is not the consequence of a loose analysis. Hazan et al. [2015] show that NSGD with constant $\gamma_t \equiv \gamma$ does not converge when the mini-batch size is smaller than $\Theta(\epsilon^{-1})$ for a non-smooth convex function. Here we provide a non-convergence result in the gradient norm with a smooth objective for all uniformly bounded stepsizes. The intuition behind this is illustrated in Figure 6.3 in the appendix, where $\mathbb{E}_\xi\, g(x;\xi)/\|g(x;\xi)\|$ can easily vanish or be in the opposite direction of $\nabla f(x)$ under some noises.

Theorem 6.4.1. *Fixing $\ell > 0$, $\sigma > 0$, $\epsilon > 0$, $\Delta > 0$ and stepsize sequence $\{\gamma_t\}_{t=0}^{\infty}$ with $\gamma_t \leq \gamma_{\max}$ that $\epsilon^2 < \min\{\sigma^2, 2\ell\Delta, 2\Delta(\sigma-\epsilon)/\gamma_{\max}\}$, there exists an ℓ-smooth convex function f, initial point x_0 with $f(x_0) - \min_x f(x) \leq \Delta$ and zero-mean noises with σ^2 variance such that the output from NSGD satisfies $\mathbb{E}\|\nabla f(x_t)\| \geq \epsilon$ for all t.*

This theorem implies that fixing function class (ℓ, Δ, σ) and any sequence $\{\gamma_t\}_t$ uniformly upper bounded by $\gamma_{\max}$, NSGD cannot converge to an arbitrarily small ϵ. Specifically, the expected gradient norm will always stay larger than $\min\{\sigma, \sqrt{2\ell\Delta}, \gamma_{\max}^{-1}(-\Delta + \sqrt{\Delta^2 + 2\Delta\gamma_{\max}\sigma})\}$. Most $\{\gamma_t\}_t$ used in practice is upper bounded, e.g., constant or decreasing sequences. The condition $\epsilon^2 < 2\ell\Delta$ is necessary by noting that $\|\nabla f(x_0)\|^2 \leq 2\ell[f(x_0) - \min_x f(x)] \leq 2\ell\Delta$. Considering $\gamma_t = 1/\sqrt{t+1}$, when $\Delta \geq \sigma$ and $\sqrt{2\ell\Delta} \geq \sigma$, it matches with Proposition 5 where NSGD can only converge to a $\Theta(\sigma)$-stationary point. Since Sign-SGD and NSGD coincide in one-dimensional objectives, our non-convergent example also applies to Sign-SGD. It sheds light on why increasing batch size improves Normalized and Sign-SGD [Zhao et al., 2021, Kunstner et al., 2023]. However, they are generally different in higher dimensions, as Karimireddy et al. [2019] show that sign-SGD may not converge even with full-batch.

NSGD with momentum. While NSGD may not always converge, Cutkosky and Mehta [2020] introduced NSGD with momentum (NSGD-M) presented in Algorithm 13 with constant $\gamma_t \equiv \gamma$. We provide the following modification with diminishing γ_t that eliminates the need to specify the total number of runs beforehand.

Proposition 6. *Under Assumptions 16 and 17, if we run NSGD-M with $\alpha_t = \frac{\sqrt{2}}{\sqrt{t+2}}$ and $\gamma_t = \frac{\gamma}{(t+1)^{3/4}}$, then for any $\gamma > 0$,*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla f(x_t)\| \leq C\left(\frac{\Delta}{\gamma} + (\sigma + \ell\gamma)\log(T)\right) T^{-\frac{1}{4}},$$

where $C > 0$ is a numerical constant.

It implies that NSGD-M attains a complexity of $\widetilde{\mathcal{O}}((\gamma^{-4} + \gamma^4\ell^4)\epsilon^{-4})$ for any $\gamma > 0$. Compared with Theorem 6.3.1 and 6.3.3, NSGD-M not only achieves near-optimal

Algorithm 13 NSGD-M

Input: initial point x_0, stepsize sequence $\{\gamma_t\}$, momentum sequence $\{\alpha_t\}$, and initial momentum g_0.
for $t = 0, 1, 2, \ldots$ **do**
  $x_{t+1} = x_t - \frac{\gamma_t}{\|g_t\|} g_t$
  sample ξ_{t+1}
  $g_{t+1} = (1 - \alpha_t) g_t + \alpha_t g(x_{t+1}; \xi_{t+1})$
end for

Algorithm 14 AMSGrad-norm

Input: initial point x_0, momentum parameters $0 \leq \beta_1 < 1$ and $0 \leq \beta_2 \leq 1$, stepsize sequence $\{\gamma_t\}$ and initial momentum m_0 and $v_0 > 0$.
$\hat{v}_0 = v_0$
for $t = 0, 1, 2, \ldots$ **do**
  sample ξ_t
  $m_{t+1} = \beta_1 m_t + (1 - \beta_1) g(x_t; \xi_t)$
  $v_{t+1}^2 = \beta_2 v_t^2 + (1 - \beta_2) \|g(x_t; \xi_t)\|^2$
  $\hat{v}_{t+1}^2 = \max\{\hat{v}_t^2, v_{t+1}^2\}$
  $x_{t+1} = x_t - \frac{\gamma_t}{\sqrt{\hat{v}_{t+1}^2}} m_{t+1}$
end for

dependency in the target accuracy ϵ, but also shreds the exponential term when the hyper-parameter is agnostic to smoothness constant.

6.4.2 *AMSGrad-norm*

AMSGrad was introduced by Reddi et al. [2019] to fix the possible non-convergence issue of Adam. Notably, current analyses of AMSGrad in the stochastic setting show a convergence rate of $\widetilde{\mathcal{O}}(1/T^{1/4})$, but they rely on the assumption of *bounded stochastic gradients* [Chen et al., 2019, Zhou et al., 2018], which is much stronger than assumptions used for SGD analysis. Here, we examine the simpler norm version of AMSGrad, presented in Algorithm 14. We prove that without assuming bounded stochastic gradients, AMSGrad-norm with default $\gamma_t = \gamma/\sqrt{t+1}$ may converge at an arbitrarily slow polynomial rate. In fact, this holds even if the true gradients are bounded. We believe this result is of independent interest.

Theorem 6.4.2. *For any $\ell > 0$, $\Delta > 0$, $\sigma > 0$ and $T > 1$, there exists a ℓ-smooth function $f : \mathbb{R}^2 \to \mathbb{R}^2$, x_0 with $f(x_0) - \inf_x f(x) \leq \Delta$ and noise distribution P with variance upper bounded by σ^2, such that if we run AMSGrad-norm with $0 \leq \beta_1 \leq 1$, $0 \leq \beta_2 < 1$ and $\gamma_t = \frac{\gamma}{\sqrt{t+1}}$, we have with probability $\frac{1}{2}$, it holds that*

$$\min_{t \in \{0,1,\ldots,T-1\}} \|\nabla f(x_t)\| \geq \sqrt{\frac{\Delta}{16 \max\left\{1/\ell, \frac{\gamma\sqrt{2\Gamma\left(1-\frac{\zeta}{2}\right)}}{\sigma\left(e\left(\frac{1}{\zeta}-1\right)\right)^{\frac{\zeta}{2}}(1-\zeta)\sqrt{1-\beta_2}} \left(T^{1-\zeta} - \zeta\right)\right\}}}$$

for any $\frac{1}{2} < \zeta < 1$, where $\Gamma(\cdot)$ denotes the Gamma function.

The intuition behind this theroem is that since AMSGrad utilizes the maximum norm of past stochastic gradients with momentum in the denominator of stepsizes, some noise distributions enable this maximum norm to increase polynomially, making the stepsizes too small. However, we can still explore its benefit in the deterministic setting. Whether it converges without assuming bounded gradients, to the best of our knowledge, is unknown. Here, for simplicity, we consider AMSGrad-norm without momentum, i.e., $\beta_1 = \beta_2 = 0$.

Theorem 6.4.3. *Under Assumption 16, if we run AMSGrad-norm with* $\gamma_t = \frac{\gamma}{\sqrt{t+1}}$, $v_0 > 0$ *and* $\beta_1 = \beta_2 = 0$ *in the deterministic setting, then for any* $\gamma > 0$ *and* $0 < \alpha < 1$,

$$\frac{1}{T}\sum_{t=0}^{T-1} \|\nabla f(x_t)\| \leq \begin{cases} T^{-\frac{1}{4}}\sqrt{2\Delta \max\{v_0, \sqrt{2\ell\Delta}\}\gamma^{-1}}, & \text{when } v_0 < \gamma\ell, \\ T^{-\frac{1}{2}}\gamma^2\ell^2 v_0^{-2} + T^{-\frac{1}{4}}\sqrt{2\gamma(M+\Delta)\max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\}}, & \text{when } v_0 \geq \gamma\ell, \end{cases}$$

where $M = \ell\gamma^2\left(1 + \log\left(\frac{\ell\gamma}{v_0}\right)\right)$.

The theorem implies that AMSGrad-norm achieves a complexity of $\widetilde{\mathcal{O}}((\ell^4\gamma^4 + \ell^2 + \ell^3\gamma^2 + \ell\gamma^{-2})\epsilon^{-4})$ with the default $\gamma_t = \Theta(t^{-1/2})$ [Reddi et al., 2019, Chen et al., 2019, Guo et al., 2021a]. Compared with untuned Gradient Descent, it gets rid of the exponential dependency. In the proof, we show that before the first iteration τ when stepsize η_t reduces to $1/\ell$, the accumulated gradient norms $\sum_{t=0}^{\tau-1} \|\nabla f(x_t)\|^2$ are upper bounded polynomially, which is in striking contrast with SGD in Theorem 6.3.3. We further provide theoretical guarantees for more general schemes $\frac{\gamma}{(t+1)^\alpha}$ with $0 < \alpha < 1$ in Theorem 6.5.6 in the appendix. We also derive matching lower bounds in Theorem 6.5.9 for any $0 < \alpha < 1$, and justify that AMSGrad may fail to converge with constant $\gamma_t \equiv \gamma$ (i.e., $\alpha = 0$) if the problem parameter is unknown.

6.4.3 *AdaGrad-norm*

AdaGrad chooses its stepsize to be inversely proportional to the element-wise accumulated past gradients [Duchi et al., 2011, McMahan and Streeter, 2010]. Its norm-version, AdaGrad-norm (presented in Algorithm 15) [Streeter and McMahan, 2010, Ward et al., 2020], picks stepsize in (6.1) to be

$$\eta_t = \frac{\eta}{\sqrt{v_0^2 + \sum_{k=0}^{t} \|g(x_k;\xi_k)\|^2}},$$

where $v_0 > 0$. Very recently, AdaGrad is proven to converge in nonconvex optimization without the assumption on bounded gradients or tuning η [Faw et al., 2022, Yang et al., 2022a]. Because NeAda-AdaGrad (Algorithm 11) reduces to AdaGrad-norm for minimization problems, the following result is a direct corollary of Theorem 5.3.1.

Algorithm 15 AdaGrad-norm

Input: initial point x_0, $v_0 > 0$ and $\eta > 0$
for $t = 0, 1, 2, \ldots$ **do**
  sample ξ_t
  $v_{t+1}^2 = v_t^2 + \|g(x_k; \xi_k)\|^2$
  $x_{t+1} = x_t - \frac{\eta}{\sqrt{v_{t+1}^2}} g(x_t; \xi_t)$
end for

Corollary 6.4.4. *Under Assumptions 16 and 17, if we run AdaGrad-norm, then for any $\eta > 0$ and $v_0 > 0$,*

$$\frac{1}{T}\sum_{t=0}^{T-1} \mathbb{E}\|\nabla f(x_t)\| \leq \frac{2A}{\sqrt{T}} + \frac{\sqrt{v_0 A}}{\sqrt{T}} + \frac{2\sqrt{A\sigma}}{T^{\frac{1}{4}}},$$

where $A = \widetilde{\mathcal{O}}\left(\frac{\Delta}{\eta} + \sigma + \ell\eta\right)$.

Proof. Define a function $\widetilde{f} : \mathbb{R}^d \times \mathbb{R} \to \mathbb{R}$ such that $\widetilde{f}(x, y) = f(x) - \frac{\ell}{2}y^2$. Since the $\widetilde{f}$ is ℓ-smooth and ℓ-strongly concave about y, the condition number is defined to be $\kappa = 1$. Applying AdaGrad-norm to f is equivalent to applying NeAda-AdaGrad (Algorithm 11) to $\widetilde{f}$ with $y_t \equiv 0$. For every x, we know $y^*(x) \triangleq \operatorname{argmax}_y \widetilde{f}(x, y) = 0$. Then $\mathcal{E} \triangleq \sum_{t=0}^{T-1} \frac{\ell^2 \|y_t - y^*(x_t)\|^2}{2v_0} = 0$. Plugging in $\kappa = 1$, $\mathcal{E} = 0$ and batchsize $M = 1$ to Theorem 5.3.1, we reach the conclusion.

□

The above result implies a complexity of $\widetilde{\mathcal{O}}\left((\eta^{-2} + \sigma^2 + \eta^2\ell^2)\left(\epsilon^{-2} + \sigma^2\epsilon^{-4}\right)\right)$. Notably, if η can be chosen to be $1/\sqrt{\ell}$, it achieves the optimal complexity in both ℓ and ϵ up to logarithmic terms like well-tuned SGD [Arjevani et al., 2022]. Even if η is agnostic to ℓ, AdaGrad-norm does not suffer from the exponential term present in untuned SGD. One of the intuitions in the deterministic setting, similar to the AMSGrad-norm, is that the accumulated squared gradient norm before the first iteration with stepsize smaller than $1/\ell$ will be upper bounded by a polynomial term (see Theorem 3.2 in [Li, 2022]). Another benefit of AdaGrad over other methods is to achieve optimal convergence rates simultaneously in deterministic and stochastic settings with the same hyper-parameters. This is sometimes referred to as “noise adaptivity", which is out of the scope of this paper.

6.5 APPENDIX

6.5.1 *Results Summary*

Algorithms	Upper bound; deterministic	Lower bound; deterministic	Upper bound; stochastic	Lower bound; stochastic
SGD (Eq. 6.1) $\eta_t = \frac{\eta}{(t+1)^\alpha}$	$\widetilde{\mathcal{O}}\left((4e^{2\alpha})^{\frac{(\eta\ell)^{1/\alpha}}{(1-\alpha)\wedge\alpha}}\epsilon^{\frac{-2}{(1-\alpha)\wedge\alpha}}\right)$ $\alpha\in(0,1)$ [Thm. 6.3.1, 6.5.1]	$\Omega\left((8e)^{\eta^2\ell^2/8}\epsilon^{-4}\right)$ $\alpha=1/2$ [Thm. 6.3.3]	$\widetilde{\mathcal{O}}\left((4e^{2\alpha})^{\frac{(\eta\ell)^{1/\alpha}}{(1-\alpha)\wedge\alpha}}\epsilon^{\frac{-2}{(1-\alpha)\wedge\alpha}}\right)$ $\alpha\in(0,1)$ [Thm. 6.3.1, 6.5.1]	$\Omega\left((8e)^{\eta^2\ell^2/8}\epsilon^{-4}\right)$ $\alpha=1/2$ [Thm. 6.3.3]
NSGD (Alg. 16) $\eta_t = \frac{\gamma_t}{\|g(x_t;\xi_t)\|}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-2}\right)$, $\gamma_t=\frac{\gamma}{\sqrt{t+1}}$ [Cutkosky and Mehta, 2020] & [Prop. 5]	$\Omega\left(\epsilon^{-2}\right)$ [Carmon et al., 2020]	N/A due to lower bound	Nonconvergent $\forall$ bounded $\{\gamma_t\}$ [Thm. 6.4.1]
NSGD-M (Alg. 13) $\eta_t = \frac{\gamma}{(t+1)^\alpha\|g_t\|}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-2}\right)$, $\alpha=1/2$ [Cutkosky and Mehta, 2020] & [Prop. 5]	$\Omega\left(\epsilon^{-2}\right)$ [Carmon et al., 2020]	$\widetilde{\mathcal{O}}\left(\epsilon^{-4}\right)$, $\alpha=3/4$ [Cutkosky and Mehta, 2020] & [Prop. 13]	$\Omega\left(\epsilon^{-4}\right)$ [Arjevani et al., 2022]
AMSGrad-norm (Alg. 14) $\eta_t = \frac{\gamma}{(t+1)^\alpha\sqrt{\hat{v}_{t+1}^2}}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-2/(1-\alpha)}\right)$, $\alpha\in(0,1)$ [Thm. 6.4.3, 6.5.6]	$\Omega\left(\epsilon^{-2/(1-\alpha)}\right)$, $\alpha\in(0,1)$ Nonconvergent, $\alpha=0$ [Thm. 6.5.9]	N/A due to lower bound	$\Omega\left(\epsilon^{-2/(1-\zeta)}\right)$, $\alpha=1/2$ $\forall\zeta\in(0.5,1)$ [Thm. 6.4.2]
AdaGrad-norm (Alg. 15) $\eta_t = \frac{\eta}{\sqrt{v_0^2+\sum_{k=0}^t\|g(x_k;\xi_k)\|^2}}$	$\widetilde{\mathcal{O}}\left(\epsilon^{-2}\right)$ [Yang et al., 2022a] & [Prop. 6.4.4]	$\Omega\left(\epsilon^{-2}\right)$ [Carmon et al., 2020]	$\widetilde{\mathcal{O}}\left(\epsilon^{-4}\right)$ [Yang et al., 2022a] & [Prop. 6.4.4]	$\Omega\left(\epsilon^{-4}\right)$ [Arjevani et al., 2022]

TABLE 6.2: Comparisons of complexities to find an ϵ-stationary point, i.e., $\mathbb{E}\|\nabla f(x)\|\le\epsilon$, between SGD, NSGD, NSGD-M, AMSGrad-norm and AdaGrad-norm. We only assume f is ℓ-smooth, and unbiased stochastic gradients have bounded variance σ^2. Hyper-parameters (e.g., γ and η) are arbitrary and untuned. In this table, $\widetilde{\mathcal{O}}$ and Ω hide polynomial terms in problem parameters and hyper-parameters, and $\widetilde{\mathcal{O}}$ also hides all logarithmic terms. We use η_t to denote the effective stepsize at iteration t.

In this work, we study stochastic gradient methods for minimizing smooth functions in the parameter-agnostic regime. Firstly, we show SGD with polynomially decaying stepsize $1/\sqrt{t}$ is able to converge with the order-optimal rate, with and without bounded gradients (Proposition 7 and Theorem 6.3.1). Its limitation lies in an unavoidable exponential term in ℓ^2 when we do not assume bounded gradients (Theorem 6.3.3). We demonstrate that several existing adaptive methods do not suffer from the exponential dependency, such as NGD, AdaGrad, AMSGrad-norm in the deterministic setting (Proposition 5 and Theorem 6.4.3), and NSGD-M, AdaGrad in the stochastic setting (Proposition 6 and Proposition 6.4.4). However, it does not mean adaptive methods are always better than SGD. We provide a non-convergence result for NSGD (Theorem 6.4.1) and a slow convergence result for AMSGrad-norm (Thoerem 6.4.2) in the stochastic case. We believe our results shed light on explaining commonly observed large gradients during training and provide a better

theoretical understanding of the convergence behaviors of adaptive methods in the regime with unbounded stochastic gradients.

6.5.2 *Proofs for SGD in Chapter 6.3*

A. Upper Bounds for SGD

We provide an extended theorem of Theorem 6.3.1 and include more general decaying stepsizes $\eta_t = \eta/(t+1)^\alpha$ with $0 < \alpha < 1$.

Theorem 6.5.1. *Under Assumptions 16 and 17, if we run SGD with stepsize $\eta_t = \eta/(t+1)^\alpha$ where $\eta > 0$ and $1/2 \le \alpha < 1$, then with $\eta \le 1/\ell$,*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(x_t)\|^2\right] \le \begin{cases} \frac{2}{\eta\sqrt{T}}\left(\Delta + \frac{\ell\sigma^2\eta^2}{2}(1+\log T)\right), \text{ when } \alpha = 1/2, \\ \frac{2}{\eta T^{1-\alpha}}\left(\Delta + \frac{\ell\sigma^2\eta^2}{2\left(1-2^{1-2\alpha}\right)}\right), \text{ when } 1/2 < \alpha < 1, \\ \frac{2}{\eta T^{\alpha}}\left(\frac{\Delta}{T^{1-2\alpha}} + \frac{\ell\sigma^2\eta^2}{2\left(1-2\alpha\right)}\right), \text{ when } 0 < \alpha < 1/2; \end{cases}$$

with $\eta > 1/\ell$,

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(x_t)\|^2\right] \le \begin{cases} \frac{\sqrt{2}\,(4e)^\tau}{\eta\sqrt{\pi\tau T}}\left[1+\ell\eta\left(1+2\sqrt{\tau}\right)\right]\left(\Delta + \frac{\ell\sigma^2\eta^2}{2}(1+\log T)\right), \\ \qquad \text{when } \alpha = 1/2, \\ \frac{2\,(4e^{2\alpha})^\tau}{\eta(2\pi\tau)^\alpha\, T^{1-\alpha}}\left[1+\ell\eta\left(1+\frac{\tau^{1-\alpha}}{1-\alpha}\right)\right]\left(\Delta + \frac{\ell\sigma^2\eta^2}{2\left(1-2^{1-2\alpha}\right)}\right), \\ \qquad \text{when } 1/2 < \alpha < 1, \\ \frac{2\,(4e^{2\alpha})^\tau}{\eta(2\pi\tau)^\alpha\, T^{\alpha}}\left[1+\ell\eta\left(1+\frac{\tau^{1-\alpha}}{1-\alpha}\right)\right]\left(\frac{\Delta}{T^{1-2\alpha}} + \frac{\ell\sigma^2\eta^2}{2\left(1-2\alpha\right)}\right), \\ \qquad \text{when } 0 < \alpha < 1/2, \end{cases}$$

where $\tau = \lceil (\eta\ell)^{1/\alpha} - 1 \rceil$.

Proof. By ℓ-smoothness of $f(\cdot)$,

$$\begin{aligned} f(x_{t+1}) &\le f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t\rangle + \frac{\ell}{2}\|x_{t+1} - x_t\|^2 \\ &= f(x_t) - \eta_t\langle \nabla f(x_t), g(x_t;\xi_t)\rangle + \frac{\ell\eta_t^2}{2}\|g(x_t;\xi_t)\|^2 \end{aligned}$$

Taking expectation,

$$\mathbb{E}f(x_{t+1}) \le \mathbb{E}f(x_t) - \eta_t\mathbb{E}\|\nabla f(x_t)\|^2 + \frac{\ell\eta_t^2}{2}\mathbb{E}\|\nabla f(x_t)\|^2 + \frac{\ell\eta_t^2}{2}\sigma^2$$

$$\leq \mathbb{E}f(x_t) - \left(\eta_t - \frac{\ell\eta_t^2}{2}\right)\mathbb{E}\|\nabla f(x_t)\|^2 + \frac{\ell\eta_t^2}{2}\sigma^2 \tag{6.2}$$

We note that $\eta_t - \frac{\ell\eta_t^2}{2} \geq \frac{\eta_t}{2}$ when $\eta_t \leq \frac{1}{\ell}$, i.e., $t \geq (\eta\ell)^{1/\alpha} - 1$. Define $\tau = \lceil (\eta\ell)^{1/\alpha} - 1 \rceil$. Therefore, for all $t < \tau$,

$$\mathbb{E}f(x_{t+1}) \leq \mathbb{E}f(x_t) + \frac{\ell\eta_t^2}{2}\mathbb{E}\|\nabla f(x_t)\|^2 + \frac{\ell\eta_t^2}{2}\sigma^2. \tag{6.3}$$

For all $t \geq \tau$, we have

$$\mathbb{E}f(x_{t+1}) \leq \mathbb{E}f(x_t) - \frac{\eta_t}{2}\mathbb{E}\|\nabla f(x_t)\|^2 + \frac{\ell\eta_t^2}{2}\sigma^2. \tag{6.4}$$

Summing from $t = \tau$ to $T-1$, we have

$$\sum_{t=\tau}^{T-1}\frac{\eta_t}{2}\mathbb{E}\|\nabla f(x_t)\|^2 \leq \mathbb{E}f(x_\tau) - \mathbb{E}f(x_T) + \sum_{t=\tau}^{T-1}\frac{\ell\eta_t^2}{2}\sigma^2 \tag{6.5}$$

Now we want to bound $\mathbb{E}f(x_\tau) - f(x_T) \leq \mathbb{E}f(x_\tau) - f^*$, where $f^* \triangleq \min_{x\in\mathbb{R}^d} f(x)$. From (6.3),

$$\begin{aligned}\mathbb{E}f(x_{t+1}) - f^* &\leq \mathbb{E}f(x_t) - f^* + \frac{\ell\eta_t^2}{2}\mathbb{E}\|\nabla f(x_t)\|^2 + \frac{\ell\eta_t^2}{2}\sigma^2 \\ &\leq (1+\ell^2\eta_t^2)[\mathbb{E}f(x_t) - f^*] + \frac{\ell\eta_t^2}{2}\sigma^2,\end{aligned}$$

where in the second inequality we use $\|\nabla f(x)\|^2 \leq 2\ell[f(x) - f^*]$. When $\tau = 0$, $f(x_\tau) - f(x_T) \leq \Delta$; when $\tau \geq 1$, recursing the inequality above, for $j \leq \tau$,

$$\begin{aligned}\mathbb{E}f(x_j) - f^* &\leq \Delta\left(\prod_{t=0}^{j-1} 1+\ell^2\eta_t^2\right) + \sum_{k=0}^{j-2}\left(\prod_{t=k+1}^{j-1} 1+\ell^2\eta_t^2\right)\frac{\ell\eta_k^2}{2}\sigma^2 + \frac{\ell\eta_{j-1}^2}{2}\sigma^2 \\ &\leq \left(\prod_{t=0}^{j-1} 1+\ell^2\eta_t^2\right)\left(\Delta + \sum_{t=0}^{j-1}\frac{\ell\eta_t^2}{2}\sigma^2\right) \\ &\leq \left(\prod_{t=0}^{\tau-1} 1+\ell^2\eta_t^2\right)\left(\Delta + \sum_{t=0}^{\tau-1}\frac{\ell\eta_t^2}{2}\sigma^2\right).\end{aligned} \tag{6.6}$$

Also, with $\|\nabla f(x)\|^2 \leq 2\ell[f(x) - f^*]$, if $\tau \geq 1$,

$$\begin{aligned}\sum_{t=0}^{\tau-1}\frac{\eta_t}{2}\mathbb{E}\|\nabla f(x_t)\|^2 &\leq \sum_{t=0}^{\tau-1}\eta_t\ell\mathbb{E}\left(f(x_t) - f^*\right) \\ &\leq \ell\left(\sum_{t=0}^{\tau-1}\eta_t\right)\left(\prod_{t=0}^{\tau-1} 1+\ell^2\eta_t^2\right)\left(\Delta + \sum_{t=0}^{\tau-1}\frac{\ell\eta_t^2}{2}\sigma^2\right),\end{aligned}$$

where in the second inequality we use (6.6) . Combining with (6.5) and (6.6), if $\tau \geq 1$

$$\sum_{t=0}^{T-1}\frac{\eta_t}{2}\mathbb{E}\|\nabla f(x_t)\|^2 \leq \left(\prod_{t=0}^{\tau-1} 1+\ell^2\eta_t^2\right)\left(\Delta + \sum_{t=0}^{\tau-1}\frac{\ell\eta_t^2}{2}\sigma^2\right) + \sum_{t=0}^{T-1}\frac{\ell\eta_t^2}{2}\sigma^2$$

$$+\ell\left(\sum_{t=0}^{\tau-1}\eta_t\right)\left(\prod_{t=0}^{\tau-1}1+\ell^2\eta_t^2\right)\left(\Delta+\sum_{t=0}^{\tau-1}\frac{\ell\eta_t^2}{2}\sigma^2\right).$$

We note that

$$\begin{aligned}\prod_{t=0}^{\tau-1}(1+\ell^2\eta_t^2)&=\prod_{t=0}^{\tau-1}\left(1+\frac{\ell^2\eta^2}{(t+1)^{2\alpha}}\right)=\frac{\prod_{t=0}^{\tau-1}\left(\ell^2\eta^2+(t+1)^{2\alpha}\right)}{(\tau!)^{2\alpha}}\leq\frac{\left(\ell^2\eta^2+\tau^{2\alpha}\right)^\tau}{(\tau!)^{2\alpha}}\\&\leq\frac{\left(2\ell^2\eta^2\right)^\tau}{(\tau!)^{2\alpha}}\leq\frac{(2\ell^2\eta^2)^\tau}{\left[\sqrt{2\pi\tau}\left(\frac{\tau}{e}\right)^\tau\exp\left(\frac{1}{12\tau+1}\right)\right]^{2\alpha}}\\&\leq\frac{1}{(2\pi\tau)^\alpha}\left(\frac{2\ell^2\eta^2e^{2\alpha}}{\tau^{2\alpha}}\right)^\tau\leq\frac{1}{(2\pi\tau)^\alpha}\left(4e^{2\alpha}\right)^\tau,\end{aligned}$$

where in the third inequality we use Stirling's approximation. Therefore,

$$\sum_{t=0}^{T-1}\frac{\eta_t}{2}\mathbb{E}\|\nabla f(x_t)\|^2\leq\frac{1}{(2\pi\tau)^\alpha}\left(4e^{2\alpha}\right)^\tau\left[1+\ell\left(\sum_{t=0}^{\tau-1}\eta_t\right)\right]\left(\Delta+\sum_{t=0}^{T-1}\frac{\ell\eta_t^2}{2}\sigma^2\right).$$

Plugging in $\eta_t=\eta/(t+1)^\alpha$, when $\alpha=1/2$,

$$\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\|^2\leq\frac{\sqrt{2T}}{\eta\sqrt{\pi\tau}}(4e)^\tau\left[1+\ell\eta\left(1+2\sqrt{\tau}\right)\right]\left(\Delta+\frac{\ell\sigma^2\eta^2}{2}(1+\log T)\right);$$

when $1/2<\alpha<1$,

$$\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\|^2\leq\frac{2\ T^\alpha}{\eta(2\pi\tau)^\alpha}(4e^{2\alpha})^\tau\left[1+\ell\eta\left(1+\frac{\tau^{1-\alpha}}{1-\alpha}\right)\right]\left(\Delta+\frac{\ell\sigma^2\eta^2}{2\left(1-2^{1-2\alpha}\right)}\right).$$

when $0<\alpha<1/2$,

$$\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\|^2\leq\frac{2\ T^\alpha}{\eta(2\pi\tau)^\alpha}(4e^{2\alpha})^\tau\left[1+\ell\eta\left(1+\frac{\tau^{1-\alpha}}{1-\alpha}\right)\right]\left(\Delta+\frac{\ell\sigma^2\eta^2T^{1-2\alpha}}{2\left(1-2\alpha\right)}\right);$$

If $\tau=0$, from (6.5),

$$\sum_{t=0}^{T-1}\frac{\eta_t}{2}\mathbb{E}\|\nabla f(x_t)\|^2\leq\Delta+\sum_{t=0}^{T-1}\frac{\ell\eta_t^2}{2}\sigma^2,$$

Plugging in η_t, when $\alpha=1/2$,

$$\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\|^2\leq\frac{2\sqrt{T}}{\eta}\left(\Delta+\frac{\ell\sigma^2\eta^2}{2}(1+\log T)\right);$$

when $1/2<\alpha<1$,

$$\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\|^2\leq\frac{2T^\alpha}{\eta}\left(\Delta+\frac{\ell\sigma^2\eta^2}{2\left(1-2^{1-2\alpha}\right)}\right).$$

when $0<\alpha<1/2$,

$$\sum_{t=0}^{T-1}\mathbb{E}\|\nabla f(x_t)\|^2\leq\frac{2T^\alpha}{\eta}\left(\Delta+\frac{\ell\sigma^2\eta^2T^{1-2\alpha}}{2\left(1-2\alpha\right)}\right).$$

□

Remark 6.5.2. *When we run SGD with stepsize $\eta_t = \eta/(t+1)^\alpha$, where $1/2 < \alpha < 1$, Theorem 6.5.1 implies a complexity of $\mathcal{O}\left((4e^{2\alpha})^{\frac{(\eta\ell)^{1/\alpha}}{1-\alpha}}(\eta\ell)^{\frac{1}{\alpha(1-\alpha)}}\cdot\epsilon^{\frac{-2}{1-\alpha}}\right)$ in the large initial stepsize regime $\eta > 1/\ell$. Compared with the case $\alpha = 1/2$, when α is larger, the convergence rate in T is slower, but it also comes with a smaller exponent, i.e., $(\eta\ell)^{1/\alpha}$. This is because $\alpha = 1/2$ leads to the best convergence rate in T [Drori and Shamir, 2020], while the faster decaying stepsize $\alpha > 1/2$ will reach the desirable stepsize $1/\ell$ earlier so that it accumulates less gradient norms before τ. For $0 < \alpha < 1/2$, however, it comes with both a larger exponent and a slower convergence rate.*

Proposition 7 (with bounded gradient)**.** *Under Assumption 16, 17 and additionally assuming that the gradient norm is upper bounded by G, i.e., $\|\nabla f(x)\| \le G$ for all $x \in \mathbb{R}^d$, if we run SGD with stepsize $\eta_t = \eta/\sqrt{t+1}$ with $\eta > 0$, then*

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T-1}\|\nabla f(x_t)\|^2\right] \le \frac{1}{\sqrt{T}}\left(\frac{\Delta}{\eta} + \frac{\ell\eta\left(G^2+\sigma^2\right)}{2}\log T\right).$$

Proof. By the smoothness of $f(\cdot)$, we have

$$f(x_{t+1}) \le f(x_t) - \eta_t\langle\nabla F(x_t;\xi_t), \nabla f(x_t)\rangle + \frac{\ell\eta_t^2}{2}\|\nabla F(x_t;\xi_t)\|^2.$$

Taking expectation and summing from $t = 0$ to $T-1$,

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=0}^{T-1}\eta_t\|\nabla f(x_t)\|^2\right] &\le f(x_0) - f(x_T) + \frac{\ell}{2}\sum_{t=0}^{T-1}\eta_t^2\mathbb{E}\left[\|\nabla F(x_t;\xi_t)\|^2\right] \\
&\le \Delta + \frac{\ell}{2}\sum_{t=0}^{T-1}\eta_t^2\mathbb{E}\left[\|\nabla F(x_t;\xi_t)\|^2\right] \\
&\le \Delta + \frac{\ell}{2}\sum_{t=0}^{T-1}\eta_t^2\left(\|\nabla f(x_t)\|^2 + \mathbb{E}\left[\|\nabla F(x_t;\xi_t) - f(x_t)\|^2\right]\right) \\
&\le \Delta + \frac{\ell}{2}\sum_{t=0}^{T-1}\eta_t^2\left(G^2+\sigma^2\right).
\end{aligned}$$

Let $\eta_t = \eta/\sqrt{t+1}$,

$$\begin{aligned}
\frac{\eta}{\sqrt{T}}\mathbb{E}\left[\sum_{t=0}^{T-1}\|\nabla f(x_t)\|^2\right] \le \mathbb{E}\left[\sum_{t=0}^{T-1}\frac{\eta}{\sqrt{t+1}}\|\nabla f(x_t)\|^2\right] &\le \Delta + \frac{\ell\left(G^2+\sigma^2\right)}{2}\sum_{t=0}^{T-1}\frac{\eta^2}{t+1}, \\
&\le \Delta + \frac{\ell\eta^2\left(G^2+\sigma^2\right)}{2}\log T.
\end{aligned}$$

□

B. Lower Bound for SGD

PROOF FOR THEOREM 6.3.3

Proof. We construct the hard instance with 4 segments of quadratic functions. The function is symmetric about $x = 0$, and we will define it on $x \le 0$ as below. We illustrate it in Figure 6.2.

Segment 1. We define $f(x) = \frac{\ell}{2}x^2$. We pick x_0 such that $f(x_0) - f^* = \Delta$, i.e., $x_0 = \sqrt{\frac{2\Delta}{\ell}}$. We define t_0 to be the first iteration that $\eta_{t_0} = \frac{\eta}{\sqrt{t_0+1}} \ge \frac{4}{\ell}$, i.e., $t_0 = \left\lfloor \frac{\eta^2\ell^2}{16} - 1 \right\rfloor$. With the update rule $x_{t+1} = x_t - \frac{\eta\ell}{\sqrt{t+1}}x_t = \left(1 - \frac{\eta\ell}{\sqrt{t+1}}\right)x_t$, we have for $t \le t_0$

$$\begin{aligned}|x_t|^2 &= \left[\prod_{k=1}^{t}\left(\frac{\eta\ell}{\sqrt{k}} - 1\right)\right]^2 |x_0|^2 \ge \prod_{k=1}^{t}\frac{\eta^2\ell^2}{2k}|x_0|^2 \\ &= \frac{(\eta^2\ell^2/2)^t}{(t)!}|x_0|^2 > \frac{(\eta^2\ell^2/2)^t}{\sqrt{2\pi t}(t/e)^t e^{1/12t}}|x_0|^2 \ge \frac{1}{3\sqrt{t}}(8e)^t|x_0|^2,\end{aligned}$$

where in the inequality we use $\left(\frac{\eta\ell}{\sqrt{k}} - 1\right)^2 \ge \frac{\eta^2\ell^2}{2k}$ with $k \le t_0$, in the second inequality we use Stirling's approximation, and in the last inequality we use $t \le \eta^2\ell^2/16$. We note that

$$|x_{t_0}|^2 \ge \frac{1}{3\sqrt{t_0}}(8e)^{t_0}|x_0|^2 \ge \frac{4}{3\eta\ell}(8e)^{\eta^2\ell^2/16-2}|x_0|^2.$$

Without loss of generality, we assume $x_{t_0} > 0$. Segment 1 is define on the domain $\{x : |x| \le x_{t_0}\}$.

Segment 2. This segment is the mirror of Segment 1. On domain $\{x : x_{t_0} \le x \le 2x_{t_0}\}$, we define $f(x) = -\frac{\ell}{2}(x - 2x_{t_0})^2 + \ell x_{t_0}^2$.

Segment 3. We note that

$$x_{t_0+1} = x_{t_0} - \eta_{t_0}\ell x_{t_0} = \left(1 - \frac{\eta\ell}{\sqrt{t_0+1}}\right)x_{t_0} \le -3x_{t_0},$$

where the inequality is from the definition of t_0, and

$$\tilde{\Delta} \triangleq \frac{\ell x_{t_0}^2}{2} \ge \frac{2}{3\eta}(8e)^{\eta^2\ell^2/16-2}|x_0|^2 = \frac{4}{3\eta\ell}(8e)^{\eta^2\ell^2/16-2}\Delta.$$

We construct a quadratic function such that: it passes $(-2x_{t_0}, \ell x_{t_0}^2)$ with gradient 0; the gradient at $x = x_{t_0+1}$ is $\frac{\sqrt{\tilde{\Delta}}}{2\sqrt{\max\{1/\ell, \sum_{t=t_0+1}^{T-1}\eta_t\}}}$. This quadratic function is uniquely defined to be

$$f(x) = -\frac{\sqrt{\tilde{\Delta}}\,(x + 2x_{t_0})^2}{4(-2x_{t_0} - x_{t_0+1})\sqrt{\max\{1/\ell, \sum_{t=t_0+1}^{T-1}\eta_t\}}} + \ell x_{t_0}^2.$$

It can be verified that this function is ℓ-smooth: as $x_{t_0+1} \le -3x_{t_0}$,

$$\frac{\sqrt{\frac{1}{2}\ell x_{t_0}^2}}{2(-2x_{t_0} - x_{t_0+1})} \le \sqrt{\ell} \iff \frac{\sqrt{\tilde{\Delta}}}{2(-2x_{t_0} - x_{t_0+1})\sqrt{\ell}} \le \ell$$

$$\implies \frac{\sqrt{\tilde{\Delta}}}{2(-2x_{t_0}-x_{t_0+1})\sqrt{\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}} \le \ell.$$

The function is defined on the domain $\{x : x_{t_0+1} \le x \le -2x_{t_0}\}$.

Segment 4. For convenience, we define $w = f(x_{t_0+1})$. We can verify that $w \ge \tilde{\Delta}$: as $\frac{1}{\sqrt{t_0}} < \frac{4}{\eta\ell}$,

$$-2x_{t_0} - \left(1 - \frac{\eta\ell}{\sqrt{t_0+1}}\right) x_{t_0} \le 4\sqrt{\frac{x_{t_0}}{2}} \iff \frac{-2x_{t_0}-x_{t_0+1}}{4} \le \sqrt{\frac{x_{t_0}}{2}}$$

$$\implies \frac{-2x_{t_0}-x_{t_0+1}}{4\sqrt{\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}} \le \sqrt{\frac{1}{2}\ell x_{t_0}^2} \iff \frac{\sqrt{\tilde{\Delta}}\,(x_{t_0+1}+2x_{t_0})^2}{4(-2x_{t_0}-x_{t_0+1})\sqrt{\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}} \le \tilde{\Delta}.$$

So we conclude $w \ge \tilde{\Delta}$. Now we construct a quadratic function similar to that in Proposition 1 of [Drori and Shamir, 2020]: it passes (x_{t_0+1}, w) with gradient $\frac{\sqrt{\tilde{\Delta}}}{2\sqrt{\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}}$; the minimum is at $x = x_{t_0+1} - \sqrt{\tilde{\Delta}\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}$. This quadratic function is defined to be

$$f(x) = \frac{\left(x - x_{t_0+1} + \sqrt{\tilde{\Delta}\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}\right)^2}{4\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}} + w - \frac{\tilde{\Delta}}{4}$$

on the domain $\{x : x \le x_{t_0+1}\}$. It is obvious that $f(x) \ge 0$ and is ℓ-smooth. Following the same reasoning of Proposition 1 in [Drori and Shamir, 2020], also presented as Lemma 6.5.3 in the appendix for completeness, we can conclude for all $t : t_0 + 1 \le t \le T$,

$$|\nabla f(x_t)| \ge \frac{\sqrt{\tilde{\Delta}}}{4\sqrt{\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}} \ge \frac{1}{4}\sqrt{\tilde{\Delta}}\min\left\{\sqrt{\ell}, (2\eta)^{-1/2}T^{-1/4}\right\},$$

where in the second inequality we use $\sum_{t=t_0+1}^{T-1}\eta_t = \sum_{t=t_0+1}^{T-1}\frac{\eta}{\sqrt{t+1}} \le 2\eta T^{1/2}$.

□

The following lemma is used in the proof of Theorem 6.3.3. It is a straightforward modification of Proposition 1 in [Drori and Shamir, 2020]. We present it here for completeness.

Lemma 6.5.3. *Under the same setting and notations as the proof of Theorem 6.3.3, if we run gradient descent with stepsize $\{\eta_t\}_{t=t_0+1}^{T-1}$ starting from point x_{t_0+1} on function*

$$f(x) = \frac{\left(x - x_{t_0+1} + \sqrt{\tilde{\Delta}\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}\right)^2}{4\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}} + w - \frac{\tilde{\Delta}}{4},$$

then for all $t : t_0 + 1 \le t \le T$,

$$|\nabla f(x_t)| \ge \frac{\sqrt{\tilde{\Delta}}}{4\sqrt{\max\{1/\ell,\sum_{t=t_0+1}^{T-1}\eta_t\}}}.$$

Proof. From the update of gradient descent, we have

$$x_{t+1} = x_t - \eta_t \cdot \frac{x_t - x_{t_0+1} + \sqrt{\tilde{\Delta} \max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}}}{2\max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}},$$

which leads to

$$\begin{aligned}
&x_{t+1} - x_{t_0+1} + \sqrt{\tilde{\Delta} \max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}} \\
&= \left(1 - \frac{\eta_t}{2\max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}}\right) \left(x_t - x_{t_0+1} + \sqrt{\tilde{\Delta} \max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}}\right).
\end{aligned}$$

Recursing this, for $j \le T$

$$\begin{aligned}
&x_j - x_{t_0+1} + \sqrt{\tilde{\Delta} \max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}} \\
&= \prod_{k=t_0+1}^{j-1} \left(1 - \frac{\eta_k}{2\max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}}\right) \left(x_{t_0+1} - x_{t_0+1} + \sqrt{\tilde{\Delta} \max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}}\right) \\
&\ge \exp\left(\log\frac{1}{2} \cdot \sum_{k=t_0+1}^{j-1} \frac{\eta_k}{2\max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}}\right) \sqrt{\tilde{\Delta} \max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}} \\
&\ge \frac{1}{2}\sqrt{\tilde{\Delta} \max\{1/\ell, \sum_{t=t_0+1}^{T-1} \eta_t\}},
\end{aligned}$$

where in the second inequality, we use $1 - z/2 \ge \exp(\log\frac{1}{2} \cdot z)$ for $0 \le z \le 1$. This directly implies what we want to prove by computing $\nabla f(x_j)$.

□

6.5.3 *Proofs for NSGD Family in Chapter 6.4*

Algorithm 16 Normalized Stochastic Gradient Descent (NSGD)

1: **Input:** initial point x_0
2: **for** $t = 0, 1, 2, \ldots$ **do**
3: sample ξ_t and set learning rate γ_t
4: $x_{t+1} = x_t - \frac{\gamma_t}{\|g(x_t;\xi_t)\|} g(x_t;\xi_t)$
5: **end for**

(a) example 1 (b) example 2

FIGURE 6.3: The expected update of NSGD can vanish (example 1) or be in the opposite direction (example 2) of the true gradient. The solid black arrow represents the true gradient and the dashed arrows are the possible stochastic gradients (with equal possibilities). The solid blue arrow is the expected direction of NSGD update.

PROOF FOR THEOREM 6.4.1

Proof. Let us pick $f(x) = \frac{L}{2}x^2$ with $\frac{\epsilon^2}{2\Delta} < L \leq \ell$ and $L < \frac{\sigma-\epsilon}{\gamma_{\max}}$. Then we pick x_0 such that $\frac{\epsilon}{L} < x_0 < \sqrt{\frac{2\Delta}{L}}$, which implies that $\|\nabla f(x_0)\| > \epsilon$ and $f(x_0) - \min_x f \leq \Delta$. Now we define $D = \{x : -w \leq x \leq w\}$ with $\frac{\epsilon}{L} + \gamma_{\max} < w < \frac{\sigma}{L}$. For $x \in D$, we have $\|\nabla f(x)\| \leq \sigma$ and we construct the noisy gradients: with $\delta > 1$

$$g(x;\xi) = (1+\delta)\nabla f(x) \text{ w.p. } \frac{1}{2}, \text{ and } g(x;\xi) = (1-\delta)\nabla f(x) \text{ w.p. } \frac{1}{2}.$$

It is obvious that $\nabla f(x) = \mathbb{E}[g(x;\xi)]$ and the variance at this point $\mathbb{E}\|\nabla f(x) - g(x;\xi)\|^2 = \delta^2\|\nabla f(x)\|^2 \leq \sigma^2$ with δ sufficiently close to 1. With the update rule, we note that $x_{t+1} = x_t - \gamma_t$ w.p. $1/2$ and $x_{t+1} = x_t + \gamma_t$ w.p. $1/2$, and therefore

$$\mathbb{E}_{\xi_t}\left[\|\nabla f(x_{t+1})|x_t \in D\|\right] = \frac{1}{2}[L\|x_t - \gamma_t\| + L\|x_t + \gamma_t\|] \geq L\|x_t\| = \|\nabla f(x_t)\|.$$

For $x \notin D$, we have $\|x\| > \epsilon/L + \gamma_{\max}$, and we assume there is no noise in the gradients. Therefore, if $x_t \notin D$, we know that after one step of update $\|x_{t+1}\| > \epsilon/L$, which implies $\|\nabla f(x_{t+1})\| > \epsilon$. Combining two cases that $x_t \in D$ and $x_t \notin D$, we know that $\mathbb{E}\|\nabla f(x_t)\| > \epsilon$ for all t.

□

PROOF OF PROPOSITION 5

Proof. Denote $e_t = g(x_t;\xi_t) - \nabla f(x_t)$. By Lemma 2 in [Cutkosky and Mehta, 2020],

$$f(x_{t+1}) - f(x_t) \leq -\frac{\gamma_t}{3}\|\nabla f(x_t)\| + \frac{8\gamma_t}{3}\|e_t\| + \frac{\ell\gamma_t^2}{2}.$$

Telescoping from $t = 0$ to $T - 1$,

$$\frac{\gamma}{3T^{1/2}} \sum_{t=0}^{T-1} \|\nabla f(x_t)\| \leq \frac{1}{3} \sum_{t=0}^{T-1} \gamma_t \|\nabla f(x_t)\| \leq \Delta + \frac{8}{3} \sum_{t=0}^{T-1} \gamma_t \|e_t\| + \sum_{t=0}^{T-1} \frac{\ell \gamma_t^2}{2},$$

Taking expectation, rearranging and using $\mathbb{E}\left[\|e_t\|\right] \leq \left(\mathbb{E}\left[\|e_t\|^2\right]\right)^{1/2} \leq \sigma$, we derive

$$\begin{aligned}\sum_{t=0}^{T-1} \mathbb{E}\left[\|\nabla f(x_t)\|\right] &\leq 3T^{1/2} \left[\frac{\Delta}{\gamma} + \frac{8\sigma}{3} \sum_{t=0}^{T-1} \frac{1}{(t+1)^{1/2}} + \frac{\ell\gamma}{2} \sum_{t=0}^{T-1} \frac{1}{t+1}\right] \\ &\leq 3T^{1/2} \left[\frac{\Delta}{\gamma} + 8\sigma T^{1/2} + \ell\gamma \log(T)\right].\end{aligned}$$

□

PROOF FOR PROPOSITION 6

Proof. We define $\hat{e}_t = g_t - \nabla f(x_t)$. By Lemma 2 in [Cutkosky and Mehta, 2020], for any $\gamma_t > 0$

$$f(x_{t+1}) - f(x_t) \leq -\frac{\gamma_t}{3} \|\nabla f(x_t)\| + \frac{8\gamma_t}{3} \|\hat{e}_t\| + \frac{\ell \gamma_t^2}{2}. \tag{6.7}$$

Telescoping from $t = 0$ to $T - 1$,

$$\frac{\gamma}{3T^{3/4}} \sum_{t=0}^{T-1} \|\nabla f(x_t)\| \leq \frac{1}{3} \sum_{t=0}^{T-1} \gamma_t \|\nabla f(x_t)\| \leq \Delta + \frac{8}{3} \sum_{t=0}^{T-1} \gamma_t \|\hat{e}_t\| + \sum_{t=0}^{T-1} \frac{\ell \gamma_t^2}{2},$$

By taking expectation on both sides, rearranging and controlling the variance term using Lemma 6.5.4, we derive

$$\begin{aligned}\sum_{t=0}^{T-1} \mathbb{E}\left[\|\nabla f(x_t)\|\right] &\leq 3T^{3/4} \left[\frac{\Delta}{\gamma} + \frac{8}{3\gamma} \sum_{t=0}^{T-1} \gamma_t \mathbb{E}\left[\|\hat{e}_t\|\right] + \frac{\ell\gamma}{2} \sum_{t=0}^{T-1} (t+1)^{-3/2}\right] \\ &\leq 3T^{3/4} \left[\frac{\Delta}{\gamma} + \frac{8}{3} \left(C_1\sigma + C_2\ell\gamma\right) \log(T) + \frac{2\ell\gamma}{T^{1/2}}\right] \\ &\leq CT^{3/4} \left[\frac{\Delta}{\gamma} + (\sigma + \ell\gamma) \log(T)\right].\end{aligned}$$

□

Lemma 6.5.4. *Under the setting of Theorem 6, there exist numerical constants $C_1, C_2 > 0$ such that for all $t \geq 1$,*

$$\mathbb{E}\left[\|\hat{e}_t\|\right] \leq C_1 \sigma \alpha_t^{1/2} + C_2 \ell \gamma_t \alpha_t^{-1},$$

$$\sum_{t=0}^{T-1} \gamma_t \mathbb{E}\left[\|\hat{e}_t\|\right] \leq \left(C_1 \sigma\gamma + C_2 \ell \gamma^2\right) \log(T),$$

where $\hat{e}_t = g_t - \nabla f(x_t)$.

Proof. Define $e_t = g(x_t;\xi_t) - \nabla f(x_t)$, $S_t = \nabla f(x_t) - \nabla f(x_{t+1})$. Then

$$\begin{aligned}\hat{e}_{t+1} &= g_{t+1} - \nabla f(x_{t+1}) \\ &= (1-\alpha_t)g_t + \alpha_t g(x_{t+1};\xi_{t+1}) - \nabla f(x_{t+1}) \\ &= (1-\alpha_t)\hat{e}_t + \alpha_t e_{t+1} + (1-\alpha_t)S_t.\end{aligned}$$

Unrolling the recursion from $t = T-1$ to $t = 0$, we have

$$\hat{e}_T = \left(\prod_{t=0}^{T-1}(1-\alpha_t)\right)\hat{e}_0 + \sum_{t=0}^{T-1}\alpha_t e_{t+1}\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau) + \sum_{t=0}^{T-1}(1-\alpha_t)S_t\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau). \quad (6.8)$$

Define the σ-field $\mathcal{F}_t := \sigma(\{x_0, \xi_0, \ldots, \xi_{t-1}\})$. Notice that for any $t_2 > t_1 \geq 0$ we have

$$\mathbb{E}\left[\langle e_{t_1}, e_{t_2}\rangle\right] = \mathbb{E}\left[\mathbb{E}\left[\langle e_{t_1}, e_{t_2}\rangle|\mathcal{F}_{t_2}\right]\right] = \mathbb{E}\left[\langle e_{t_1}, \mathbb{E}\left[e_{t_2}|\mathcal{F}_{t_2}\right]\rangle\right] = 0. \quad (6.9)$$

Then taking norm, applying expectation on both sides of (6.8) and using $\mathbb{E}\left[\|\hat{e}_0\|\right] \leq \sigma$, we have

$$\begin{aligned}\mathbb{E}\left[\|\hat{e}_T\|\right] &\leq \left(\prod_{t=0}^{T-1}(1-\alpha_t)\right)\sigma + \mathbb{E}\left[\left\|\sum_{t=0}^{T-1}\alpha_t e_{t+1}\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)\right\|\right] \\ &\quad + \mathbb{E}\left[\left\|\sum_{t=0}^{T-1}(1-\alpha_t)S_t\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)\right\|\right] \\ &\leq \left(\prod_{t=0}^{T-1}(1-\alpha_t)\right)\sigma + \left(\mathbb{E}\left[\left\|\sum_{t=0}^{T-1}\alpha_t e_{t+1}\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)\right\|^2\right]\right)^{1/2} \\ &\quad + \sum_{t=0}^{T-1}(1-\alpha_t)\mathbb{E}\left[\|S_t\|\right]\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau) \\ &\leq \left(\prod_{t=0}^{T-1}(1-\alpha_t)\right)\sigma + \left(\sum_{t=0}^{T-1}\alpha_t^2\mathbb{E}\left[\|e_{t+1}\|^2\right]\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)^2\right)^{1/2} \\ &\quad + \ell\sum_{t=0}^{T-1}(1-\alpha_t)\gamma_t\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau) \\ &\leq \left(\prod_{t=0}^{T-1}(1-\alpha_t)\right)\sigma + \left(\sum_{t=0}^{T-1}\alpha_t^2\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)\right)^{1/2}\sigma + \left(\sum_{t=0}^{T-1}\gamma_t\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)\right)\ell,\end{aligned}$$

where the first inequality holds by Jensen's inequality applied to $x \mapsto x^2$, the second inequality follows by (6.9) and the bound $\|S_t\| \leq \ell\|x_{t+1} - x_t\| = \ell\gamma_t$. The last step is due to bounded variance $\mathbb{E}\left[\|\hat{e}_0\|\right] \leq \sigma$ and $\alpha_t \leq 1$.

By the choice of momentum sequence, we have $\alpha_0 = 1$ and the first term is zero. By Lemma 6.5.5, there exist numerical constants $C_1, C_2 > 0$ such that

$$\left(\sum_{t=0}^{T-1}\alpha_t^2\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)\right)^{1/2} \leq C_1\alpha_T^{1/2}, \qquad \left(\sum_{t=0}^{T-1}\gamma_t\prod_{\tau=t+1}^{T-1}(1-\alpha_\tau)\right) \leq C_2\gamma_T\alpha_T^{-1}.$$

Therefore, for all $T \geq 1$, we have

$$\mathbb{E}\left[\|\hat{e}_T\|\right] \leq C_1 \sigma \alpha_T^{1/2} + C_2 \ell \gamma_T \alpha_T^{-1}.$$

$$\begin{aligned}
\sum_{t=0}^{T-1} \gamma_t \mathbb{E}\left[\|\hat{e}_t\|\right] &\leq C_1 \sigma \sum_{t=0}^{T-1} \gamma_t \alpha_t^{1/2} + C_2 \ell \sum_{t=0}^{T-1} \gamma_t^2 \alpha_t^{-1} \\
&\leq C_1 \sigma \gamma \sum_{t=0}^{T-1} (t+1)^{-3/4} (t+1)^{-1/4} + C_2 \ell \gamma^2 \sum_{t=0}^{T-1} (t+1)^{-3/2} (t+1)^{1/2} \\
&\leq \left(C_1 \sigma \gamma + C_2 \ell \gamma^2\right) \log(T).
\end{aligned}$$

□

Lemma 6.5.5 (Lemma 15 in [Fatkhullin et al., 2023])**.** *Let $q \in [0,1)$, $p \geq 0$, $\gamma_0 > 0$ and let $\eta_t = \left(\frac{2}{t+2}\right)^q$, $\gamma_t = \gamma_0 \left(\frac{1}{t+1}\right)^p$ for every integer t. Then for any integers t and $T \geq 1$, it holds*

$$\sum_{t=0}^{T-1} \gamma_t \prod_{\tau=t+1}^{T-1} (1 - \eta_\tau) \leq C \gamma_t \eta_T^{-1},$$

where $C := 2^{p-q}(1-q)^{-1} t_0 \exp\left(2^q (1-q) t_0^{1-q}\right) + 2^{2p+1-q}(1-q)^{-2}$ and

$t_0 := \max\left\{\left(\frac{p}{(1-q)2^q}\right)^{\frac{1}{1-q}}, 2\left(\frac{p-q}{(1-q)^2}\right)\right\}^{\frac{1}{1-q}}$.

6.5.4 *Proofs for AMSGrad-Norm in Chapter 6.4*

The following is an extended version of Theorem 6.4.3 including $\gamma_t = \frac{\gamma}{(t+1)^\alpha}$ with $0 < \alpha < 1$.

Theorem 6.5.6. *Under Assumption 16, if we run AMSGrad-norm with $\gamma_t = \frac{\gamma}{(t+1)^\alpha}$, $v_0 > 0$ and $\beta_1 = \beta_2 = 0$ in the deterministic setting, then for any $\gamma > 0$ and $0 < \alpha < 1$, if $v_0 < \gamma\ell$*

$$\frac{1}{T} \sum_{t=0}^{T-1} \|\nabla f(x_t)\|^2 \leq \frac{2\Delta}{\gamma T^{1-\alpha}} \max\{v_0, \sqrt{2\ell\Delta}\},$$

if $v_0 \geq \gamma\ell$

$$\frac{1}{T} \sum_{t=0}^{T-1} \|\nabla f(x_t)\|^2 \leq \frac{\left(\frac{\ell\gamma}{v_0}\right)^{\frac{1}{\alpha}} \gamma^2 \ell^2}{T} + \frac{2(M+\Delta)}{\gamma T^{1-\alpha}} \max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\},$$

where

$$M = \begin{cases} \ell\gamma^2 \left(1 + \log\left(\frac{\ell\gamma}{v_0}\right)\right), \text{ when } \alpha = 1/2, \\ \frac{\ell\gamma^2}{2\left(1-2^{1-2\alpha}\right)}, \text{ when } 1/2 < \alpha < 1, \\ \frac{\gamma(\ell\gamma)^{\frac{1}{\alpha}-1}}{2(1-2\alpha)v_0^{\frac{1}{\alpha}-2}}, \text{ when } 0 < \alpha < 1/2. \end{cases}$$

Proof. The effective stepsize of AMSGrad-norm contains a maximum over all gradient norms in the denominator. As it is desirable to find a lower bound for the effective stepsize, we begin by bounding the gradient norms.

Let τ be the first iteration where the effective stepsize is less or equal to $1/\ell$, i.e., $\eta_{\tau-1} > 1/\ell$ and $\eta_\tau \leq 1/\ell$. First, we assume $\tau \geq 1$, i.e., $v_0 < \gamma\ell$. The time stamp τ itself is naturally bounded by

$$\eta_{\tau-1} = \frac{\gamma}{\tau^\alpha v_\tau} > \frac{1}{\ell} \quad \Longrightarrow \quad \tau < \left(\frac{\ell\gamma}{v_\tau}\right)^{\frac{1}{\alpha}} \leq \left(\frac{\ell\gamma}{v_0}\right)^{\frac{1}{\alpha}}.$$

We have

$$\sum_{t=0}^{\tau-1} \|\nabla f(x_t)\|^2 \leq \tau\gamma^2\ell^2 \leq \left(\frac{\ell\gamma}{v_0}\right)^{\frac{1}{\alpha}} \gamma^2\ell^2. \tag{6.10}$$

By ℓ-smoothness of $f(\cdot)$,

$$\begin{aligned} f(x_{t+1}) &\leq f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{\ell}{2}\|x_{t+1} - x_t\|^2 \\ &= f(x_t) - \eta_t \|\nabla f(x_t)\|^2 + \frac{\ell\eta_t^2}{2}\|\nabla f(x_t)\|^2 \\ &\leq f(x_t) + \frac{\ell\eta_t^2}{2}\|\nabla f(x_t)\|^2. \end{aligned} \tag{6.11}$$

Therefore,

$$\begin{aligned} f(x_\tau) - f(x_0) &\leq \frac{\ell}{2}\sum_{t=0}^{\tau-1} \eta_t^2 \|\nabla f(x_t)\|^2 = \frac{\ell}{2}\sum_{t=0}^{\tau-1} \frac{\gamma_t^2}{v_{t+1}^2}\|\nabla f(x_t)\|^2 \leq \frac{\ell}{2}\sum_{t=0}^{\tau-1}\gamma_t^2 \\ &\leq \begin{cases} \frac{\ell\gamma^2}{2}(1+\log\tau), \text{ when } \alpha = 1/2, \\ \frac{\ell\gamma^2}{2\left(1-2^{1-2\alpha}\right)}, \text{ when } 1/2 < \alpha < 1, \\ \frac{\ell\gamma^2\tau^{1-2\alpha}}{2(1-2\alpha)}, \text{ when } 0 < \alpha < 1/2. \end{cases} \end{aligned}$$

We denote the right-hand side as M. Also from (6.11) and definition of τ, we know that $f(x_t) \leq f(x_\tau)$ for $t \geq \tau$ and therefore, for all $t \geq \tau$,

$$f(x_t) - f^* = f(x_\tau) - f(x_0) + f(x_0) - f^* \leq M + \Delta,$$

which implies

$$\|\nabla f(x_t)\|^2 \le 2\ell(f(x_t) - f^*) \le 2\ell(M + \Delta).$$

Therefore, we can bound for all $t \ge 0$,

$$v_t \le \max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\}.$$

For $t \ge \tau$, by (6.11)

$$f(x_{t+1}) - f(x_t) \le -\frac{\eta_t}{2}\|\nabla f(x_t)\|^2.$$

By telescoping from $t = \tau$ to $T-1$, we get

$$\begin{aligned}
2\left(f(x_\tau) - f(x_T)\right) &\ge \sum_{t=\tau}^{T-1} \eta_t \|\nabla f(x_t)\|^2 \\
&= \sum_{t=\tau}^{T-1} \frac{\gamma}{(t+1)^\alpha v_{t+1}} \|\nabla f(x_t)\|^2 \\
&\ge \sum_{t=\tau}^{T-1} \frac{\gamma}{T^\alpha v_{t+1}} \|\nabla f(x_t)\|^2 \\
&\ge \sum_{t=\tau}^{T-1} \frac{\gamma}{T^\alpha \max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\}} \|\nabla f(x_t)\|^2.
\end{aligned}$$

Then we have

$$\begin{aligned}
\sum_{t=\tau}^{T-1} \|\nabla f(x_t)\|^2 &\le \frac{2}{\gamma}\left(f(x_\tau) - f(x_T)\right) T^\alpha \max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\} \\
&\le \frac{2}{\gamma}\left(f(x_\tau) - f(x^*)\right) T^\alpha \max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\} \\
&\le \frac{2(M+\Delta)}{\gamma} T^\alpha \max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\}.
\end{aligned}$$

Combining with (6.10), we obtain

$$\sum_{t=0}^{T-1} \|\nabla f(x_t)\|^2 \le \left(\frac{\ell\gamma}{v_0}\right)^{\frac{1}{\alpha}} \gamma^2\ell^2 + \frac{2(M+\Delta)T^\alpha}{\gamma} \max\{\gamma\ell, \sqrt{2\ell(M+\Delta)}\}.$$

When $\tau = 0$, we have

$$2\left(f(x_0) - f(x_T)\right) \ge \sum_{t=0}^{T-1} \frac{\gamma}{T^\alpha v_{t+1}} \|\nabla f(x_t)\|^2 \ge \sum_{t=\tau}^{T-1} \frac{\gamma}{T^\alpha \max\{v_0, \sqrt{2\ell\Delta}\}} \|\nabla f(x_t)\|^2,$$

which implies

$$\sum_{t=0}^{T-1} \|\nabla f(x_t)\|^2 \le \frac{2\Delta T^\alpha}{\gamma} \max\{v_0, \sqrt{2\ell\Delta}\}.$$

Remark 6.5.7. *For any $0 < \alpha < 1$, if we compare simplified AMSGrad with $\gamma_t = \frac{\gamma}{(t+1)^\alpha}$ to SGD with $\eta_t = \frac{\eta}{(t+1)^\alpha}$ in the deterministic case (setting $\sigma = 0$ in Theorem 6.5.1), we observe that they*

achieve the same convergence rate. However, the complexity of simplified AMSGrad only includes polynomial term in γ and ℓ, while that of SGD includes an exponential term in $(\eta\ell)^{1/\alpha}$.

□

In the following, we will first provide the lower bounds for scalar version of AMSGrad (referred to as AMSGrad-norm) with each $\alpha \in (0,1)$ and discuss why it may fail with $\alpha = 0$ when problem parameters are unknown, which means that it can not achieve the optimal complexity $\mathcal{O}(\epsilon^{-2})$ in the deterministic setting. Second, we show that it also fails to achieve the optimal convergence rate in the stochastic setting when stochastic gradients are unbounded. To make the results more general, we consider the standard scalar AMSGrad with momentum hyper-parameters β_1 and β_2, which is presented in Algorithm 14.

Before proceeding to our results, we present a lemma which is handy for conducing lower bounds for SGD-like algorithms with momentum (see Algorithm 17). As long as an upper bound is known for stepsize η_t, we can derive a lower bound similar to Proposition 1 in [Drori and Shamir, 2020].

Algorithm 17 General SGD with Momentum

1: **Input:** initial point x_0, momentum parameters $0 \leq \beta_1 < 1$ and initial moment m_0.
2: **for** $t = 0, 1, 2, \ldots$ **do**
3: sample ξ_t
4: $m_{t+1} = \beta_1 m_t + (1-\beta_1) g(x_t; \xi_t)$
5: obtain stepsize $\eta_t > 0$
6: $x_{t+1} = x_t - \eta_t m_{t+1}$
7: **end for**

Lemma 6.5.8. *For any $\ell > 0$, $\Delta > 0$ and $T > 1$, there exists a ℓ-smooth function $f : \mathbb{R} \to \mathbb{R}$, and x_0 with $f(x_0) - \inf_x f(x) \leq \Delta$, such that if we run Algorithm 17 with deterministic gradients and $\eta_t \leq \tilde{\eta}_t$ for $t = 0, 1, 2, \ldots, T-1$, then we have*

$$\min_{t \in \{0,1,\ldots,T-1\}} |\nabla f(x_t)| \geq \sqrt{\frac{\Delta}{16 \max\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\}}}.$$

Proof. We construct a quadratic function similar to Proposition 1 in [Drori and Shamir, 2020]. The following function is considered:

$$f(x) = \frac{x^2}{4 \max\left\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\right\}}.$$

Without loss of generality, we assume the initial moment m_0 is non-positive, and we set the initial point x_0 as

$$x_0 = \sqrt{\Delta \max\left\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\right\}}.$$

Otherwise if the initial moment is set to be positive, then we let x_0 be negative and follow the same reasoning.

Since x_0 is positive, the first gradient direction would be positive, i.e., $\nabla f(x_0) > 0$. Let τ be the first iteration such that $m_\tau > 0$. By the update rule and definition of τ, it is obvious that $x_t \geq x_0$ for $t \leq \tau - 1$. If $T \leq \tau$, it trivially holds that $\nabla f(x_t) \geq \nabla f(x_0)$ for all $0 \leq t \leq T-1$. Otherwise, we have $m_\tau = \beta_1 m_{\tau-1} + (1-\beta_1)\nabla f(x_{\tau-1}) \leq (1-\beta_1)\nabla f(x_{\tau-1})$. That is to say, the gradient estimation m_τ used in the τ-th step has the correct direction but its magnitude is no larger than the actual gradient. Starting from the τ-th iteration, x_t will monotonically move left towards the solution. Note that since our stepsize is small enough, i.e.,

$$\eta_t \leq \tilde{\eta}_t < 2\max\left\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\right\},$$

the updates will remain positive, i.e., $x_t > 0$ for $t \geq \tau$. By the update rule, we note that $x_{t+1} \leq x_t$ for $t \geq \tau$, and therefore $\nabla f(x_{t+1}) < \nabla f(x_t)$. We can conclude that for any $t \geq \tau$, we have $m_t \leq \nabla f(x_{\tau-1})$. Then for $t \geq \tau - 1$ we have

$$\begin{aligned}
x_t &= x_{\tau-1} - \sum_{k=\tau-1}^{t-1} \eta_t m_{t+1} \\
&\geq x_{\tau-1} - \sum_{k=\tau-1}^{t-1} \tilde{\eta}_t \nabla f(x_{\tau-1}) \\
&= x_{\tau-1} - \sum_{k=\tau-1}^{t-1} \frac{\tilde{\eta}_t}{2\max\left\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\right\}} x_{\tau-1} \\
&\geq \frac{1}{2} x_{\tau-1} \\
&\geq \frac{1}{2} x_0.
\end{aligned}$$

Then we conclude by

$$|\nabla f(x_t)| = \frac{x_t}{2\max\left\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\right\}} \geq \frac{x_0}{4\max\left\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\right\}} = \sqrt{\frac{\Delta}{16\max\{1/\ell, \sum_{t=0}^{T-1} \tilde{\eta}_t\}}}.$$

□

Now we proceed to provide the lower bound for deterministic case.

Theorem 6.5.9. *For any $\ell > 0$, $\Delta > 0$ and $T > 1$, there exists a ℓ-smooth function $f : \mathbb{R} \to \mathbb{R}$ and x_0 with $f(x_0) - \inf_x f(x) \leq \Delta$, such that if we run Algorithm 14 with deterministic gradients, $0 < v_0 \leq \frac{\ell\gamma}{2}$, and $\gamma_t = \frac{\gamma}{(t+1)^\alpha}$ with $\gamma \leq \frac{4\Delta}{v_0}$, we have (1) if $0 < \alpha < 1$, for any $0 \leq \beta_1 < 1$ and $0 \leq \beta_2 \leq 1$, we have*

$$\min_{t\in\{0,1,\ldots,T-1\}} |\nabla f(x_t)| \geq \sqrt{\frac{\Delta}{16\max\{1/\ell, \frac{\gamma}{(1-\alpha)v_0}T^{1-\alpha}\}}},$$

and (2) if $\alpha = 0$, for $\beta_1 = 0$ and any $0 \leq \beta_2 \leq 1$, we have

$$\min_{t\in\{0,1,\ldots,T-1\}} |\nabla f(x_t)| \geq v_0.$$

Remark 6.5.10. *From the theorem, we can conclude that the optimal convergence rate $\frac{1}{\sqrt{T}}$ for $\|\nabla f(x_t)\|$ is infeasible for AMSGrad with polynomially decreasing stepsize. When $\alpha = 0$, a similar result can be obtained for the case $\beta_1 \geq 0$, $\beta_2 = 0$ and small enough v_0.*

Proof. For $\alpha > 0$, we have

$$\eta_t = \frac{\gamma}{(t+1)^\alpha\sqrt{\hat{v}_{t+1}^2}} \leq \frac{\gamma}{(t+1)^\alpha v_0}.$$

Let $\tilde{\eta}_t = \frac{\gamma}{(t+1)^\alpha v_0}$ and then we have

$$\sum_{t=0}^{T-1}\tilde{\eta}_t = \sum_{t=0}^{T-1}\frac{\gamma}{(t+1)^\alpha v_0} \leq \frac{\gamma}{(1-\alpha)v_0}T^{1-\alpha}.$$

Applying Lemma 6.5.8 directly gives us the desired result.

For $\alpha = 0$, we consider function

$$f(x) = \frac{v_0}{\gamma}x^2.$$

Note that since $v_0 \leq \frac{\ell\gamma}{2}$, the function is ℓ-smooth. Let

$$x_0 = \frac{\gamma}{2},$$

which satisfies the condition that $f(x_0) \leq \Delta$. Then after one update

$$v_1^2 = \beta_2 v_0^2 + (1-\beta_2)\|\nabla f(x_0)\|^2 = v_0^2$$
$$x_1 = x_0 - \frac{\gamma}{\sqrt{v_1^2}}\nabla f(x_0) = -\frac{\gamma}{2} = -x_0.$$

If we continue this calculation, we find that the iterates will oscillate between $\frac{\gamma}{2}$ and $-\frac{\gamma}{2}$ forever, which finishes the proof. □

PROOF FOR THEOREM 6.4.2

Proof. We consider a two-dimensional function $f : \mathbb{R}^2 \to \mathbb{R}^2$, for $x = (x^1, x^2)^\top \in \mathbb{R}^2$,

$$f(x) = F(x^1),$$

where its function value only depends on the first dimension and we will define $F : \mathbb{R} \to \mathbb{R}$ later. The gradient at x is $\nabla f(x) = (\nabla F(x^1), 0)^\top$. We add the noise only to the second dimension, i.e., $g(x; \xi) = (\nabla F(x^1), \xi)$. For any $t \geq 0$, the probability density function of the noise as

$$p_{\xi_t}(x) = \begin{cases} \frac{1}{s\zeta} \left(\frac{x}{s}\right)^{-1-\frac{2}{\zeta}} e^{-\left(\frac{x}{s}\right)^{-\frac{2}{\zeta}}}, & x \geq 0; \\ \frac{1}{s\zeta} \left(\frac{-x}{s}\right)^{-1-\frac{2}{\zeta}} e^{-\left(\frac{-x}{s}\right)^{-\frac{2}{\zeta}}}, & x < 0, \end{cases}$$

where $s = \frac{\sigma}{\sqrt{\Gamma\left(1-\frac{\zeta}{2}\right)}}$. Note that the distribution is symmetric and $\mathbb{E}\left[\xi_t\right] = 0$. Also, we note that $|\xi_t|$ follows the Fréchet distribution [De Gusmao et al., 2011] with cumulative distribution function

$$\Pr(|\xi_t| \leq x) = e^{-\left(\frac{x}{s}\right)^{-\frac{2}{\zeta}}},$$

and

$$\begin{aligned} \operatorname{Var}\left[\xi_t\right] &= \mathbb{E}\left[|\xi_t|^2\right] - \left(\mathbb{E}\left[\xi_t\right]\right)^2 \\ &= s^2\Gamma\left(1-\frac{\zeta}{2}\right) - \left(\mathbb{E}\left[\xi_t\right]\right)^2 \\ &\leq s^2\Gamma\left(1-\frac{\zeta}{2}\right) \\ &\leq \sigma^2, \end{aligned}$$

where we used the exact second moment for Fréchet distribution.

Next, we will show that $\tilde{\xi}_t := \max_{0\leq k\leq t}\{|\xi_k|\} \geq \Omega\left(\frac{1}{(t+1)^{\zeta-1/2}}\right)$ with probability $\frac{1}{2}$. We know that $\tilde{\xi}_t$ also follows Fréchet distribution with CDF

$$\Pr(\tilde{\xi}_t \leq x) = \exp\left(-\left(\frac{x}{s\cdot(t+1)^{\frac{\zeta}{2}}}\right)^{-\frac{2}{\zeta}}\right).$$

Then for constant $C > 0$,

$$\begin{aligned} \Pr(\tilde{\xi}_t \leq C\cdot(t+1)^{\zeta-\frac{1}{2}}) &= \exp\left(-\left(\frac{C\cdot(t+1)^{\zeta-\frac{1}{2}}}{s\cdot(t+1)^{\frac{\zeta}{2}}}\right)^{-\frac{2}{\zeta}}\right) \\ &= \exp\left(-\left(\frac{C}{s}\right)^{\frac{2}{\zeta}}(t+1)^{\frac{1}{\zeta}-1}\right) \\ &\leq \frac{1}{4(t+1)^2}, \end{aligned}$$

where the last inequality is by selecting $C = \frac{s\left(e\left(\frac{1}{\zeta}-1\right)\right)^{\frac{\zeta}{2}}}{\sqrt{2}}$ and using $\exp\left(-\frac{2^{m+1}}{em}\cdot t^m\right) \leq \frac{1}{4t^2}$ for any $t > 0$ and $0 < m < 1$. Then using union bound, we have

$$\Pr(\tilde{\xi}_t > C\cdot(t+1)^{\zeta-\frac{1}{2}} \quad \text{for} \quad 0 \leq t \leq T-1) \geq 1 - \sum_{t=0}^{T-1} \frac{1}{4(t+1)^2} \geq \frac{1}{2}.$$

Now we have shown that with some probability, the noise is large enough. We can use this property to provide an upper bound $\tilde{\eta}_t$ for the stepsize as follow

$$\begin{aligned}
\eta_t &= \frac{\gamma}{\sqrt{t+1}\sqrt{\tilde{v}_{t+1}}} \\
&= \frac{\gamma}{\sqrt{t+1}\sqrt{\max_{0\leq k\leq t}\{\beta_2 v_k + (1-\beta_2)\|g(x_k;\xi_k)\|^2\}}} \\
&\leq \frac{\gamma}{\sqrt{t+1}\sqrt{\max_{0\leq k\leq t}\{(1-\beta_2)\|g(x_k;\xi_k)\|^2\}}} \\
&\leq \frac{\gamma}{\sqrt{t+1}\sqrt{\max_{0\leq k\leq t}\{(1-\beta_2)\|\xi_k\|^2\}}} \\
&= \frac{\gamma}{\sqrt{t+1}\sqrt{(1-\beta_2)\tilde{\xi}_t}} \\
&\leq \frac{\gamma}{C(t+1)^{\zeta}\sqrt{(1-\beta_2)}} \triangleq \tilde{\eta}_t,
\end{aligned}$$

This implies

$$\sum_{t=0}^{T-1}\tilde{\eta}_t \leq \frac{\gamma}{(1-\zeta)C\sqrt{(1-\beta_2)}}\left(T^{1-\zeta}-\zeta\right).$$

We observe that the update with AMSGrad-norm in function f corresponds to applying general SGD with momentum (Algorithm 17) to function F with stepsize η_t. Therefore, we can pick a hard instance F according to Lemma 6.5.8, and by noting that $\|\nabla f(x)\| = |\nabla F(x^1)|$ we reach our conclusion.

□

Remark 6.5.11. *As we see above, the function F in the proof is constructed by Lemma 6.5.8. We note that even assuming the gradients of f to be bounded, i.e., $\|\nabla f(x)\| \leq K$ for all x, will not prevent the slow convergence in Theorem 6.4.2. This is because in the proof of Lemma 6.5.8 all iterates stay between $[0, x_{\tau-1}]$ (e.g., $\tau = 1$ if $m_0 = 0$), so we can construct any Lipschitz function outside of this segment.*

7

SUMMARY AND FUTURE DIRECTIONS

Good questions outrank easy answers.

— Paul Samuelson

In this chapter, we summarize the key contributions of this work and explore possible future research directions.

This thesis provides a comprehensive exploration of challenges and advancements in minimax optimization, particularly focusing on imbalance, NC-NC problems, and adaptivity. In the first chapter, a Catalyst framework is introduced, inspired by proximal point methods, to address unbalanced regimes in minimax optimization. The following two chapters shed light on the NC-NC regimes, where particular emphasis is given to the behavior of the Alternating Gradient Descent Ascent (AGDA) algorithm, its convergence under different scenarios. The fourth chapter shifts focus to the NC-SC setting, introducing NeAda, a novel nested adaptive framework designed to make conventional Gradient Descent Ascent combined tuning-free with adaptive schemes. In the closing chapter, the advantages of adaptive methods, particularly in relation to Stochastic Gradient Descent, are analyzed in the setting when problem parameters are unknown. The thesis, as a whole, offers insightful strategies and analyses that push the boundaries of current understanding in minimax optimization and adaptive methods.

7.1 FUTURE DIRECTIONS

7.1.1 *Unbalanced Minimax Problems*

In Chapter 2, we introduce a universal framework for tackling unbalanced minimax problems, aiming to bridge the gap between lower and upper bounds in these scenarios. However, several intriguing questions remain unanswered. Firstly, in the context of the NC-SC setting, it would be compelling to explore if the complexity's dependence on n for finite-sum NC-SC minimax optimization can be further refined. Secondly, the absence of lower bounds for the NC-C setting leaves us uncertain about the potential for enhancing the current state-of-the-art upper bounds. Lastly, devising a universally near-optimal single-loop algorithm that encompasses all these unbalanced scenarios presents an interesting challenge.

7.1.2 *Nonconvex-Nonconcave Minimax Optimization*

While Daskalakis et al. [2021] highlight the challenges associated with identifying or finding certain types of stationary and local solutions for smooth and Lipschitz nonconvex-nonconcave minimax problems, it remains an open question whether there are alternative meaningful notions that can be achieved more efficiently. In Chapters 3 and 4, we introduce efficient algorithms aimed at finding global solutions or stationary points for two specific subclasses of nonconvex-nonconcave problems: the two-sided PL and NC-PL problems. Another intriguing avenue of exploration is determining the lower complexity bounds for these two categories.

7.1.3 *Adaptive Methods for Minimax Optimization*

In Chapter 5, the proposed algorithm, NeAda, is analyzed when the function is strongly-convex w.r.t. the dual variable. For practical applications, such as Generative Adversarial Networks [Goodfellow et al., 2014], it might be overly optimistic to assume that the dual variable exhibits such desirable properties. A potential direction for future research would be to move beyond this assumption to the NC-C setting or nonconcave structures such as PL condition. Furthermore, eliminating the assumption of bounded (stochastic) gradients for the dual variable is another important future direction. As highlighted in Chapter 6, adaptive methods possess a strict advantage over SGD in the absence of this assumption. Achieving this for NC-SC problems with our NeAda algorithm framework would hinge on the development of parameter-agnostic stochastic algorithms for strongly concave maximization problem — a known open challenge [Orvieto et al., 2022].

7.1.4 *Adaptive Methods for Problems*

In Chapter 6, we analyze the benefit of adaptive methods over SGD when the stepsize is independent of problem parameters and the objective function is not necessarily Lipschitz. There are several potential future extensions. Firstly, it is interesting to understand whether similar benefits of adaptive methods persist for the high probability convergence guarantees and extend to other adaptive optimizers. Secondly, we emphasize the significance of eliminating the assumption of bounded gradients for more adaptive algorithms. Such an assumption can hide the dependence on ℓ and obscure the advantage over SGD. Thirdly, based on our negative results concerning AMSGrad-norm, further exploration of the convergence properties of AMSGrad and its variants becomes interesting. This exploration could involve scenarios where true function gradients are unbounded, but additional

assumptions can be made regarding the noise distribution. Lastly, understanding the impact of adaptive algorithms on the optimization of possibly non-smooth nonconvex objectives, which frequently arise in the training of modern machine learning models, is another intriguing avenue for future research.

BIBLIOGRAPHY

Jacob Abernethy, Kevin A Lai, and Andre Wibisono. Last-iterate convergence rates for min-max optimization. *arXiv preprint arXiv:1906.02027*, 2019.

Jacob Abernethy, Kevin A Lai, and Andre Wibisono. Last-iterate convergence rates for min-max optimization: Convergence of hamiltonian gradient descent and consensus optimization. In *Algorithmic Learning Theory*, pages 3–47. PMLR, 2021.

Leonard Adolphs, Hadi Daneshmand, Aurelien Lucchi, and Thomas Hofmann. Local saddle point optimization: A curvature exploitation approach. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 486–495. PMLR, 2019.

Alekh Agarwal, Alina Beygelzimer, Miroslav Dudík, John Langford, and Hanna Wallach. A reductions approach to fair classification. In *International conference on machine learning*, pages 60–69. PMLR, 2018.

Kwangjun Ahn, Jingzhao Zhang, and Suvrit Sra. Understanding the unstable convergence of gradient descent. *arXiv preprint arXiv:2204.01050*, 2022.

Ahmad Ajalloeian and Sebastian U. Stich. On the convergence of sgd with biased gradients. *arXiv preprint arXiv:2008.00051*, 2021.

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

Sotirios-Konstantinos Anagnostidis, Aurelien Lucchi, and Youssef Diouane. Direct-search for a class of stochastic min-max problems. In *International Conference on Artificial Intelligence and Statistics*, pages 3772–3780. PMLR, 2021.

K Antonakopoulos. Adaptive extra-gradient methods for min-max optimization and games. In *ICLR*, volume 3, page 7, 2021.

Kimon Antonakopoulos and Panayotis Mertikopoulos. Adaptive first-order methods revisited: Convex minimization without lipschitz requirements. *NeurIPS*, 34, 2021.

Kimon Antonakopoulos, Veronica Belmega, and Panayotis Mertikopoulos. An adaptive mirror-prox method for variational inequalities with singular operators. *NeurIPS*, 32, 2019.

Yossi Arjevani, Yair Carmon, John C Duchi, Dylan J Foster, Nathan Srebro, and Blake Woodworth. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, pages 1–50, 2022.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223. PMLR, 2017.

Larry Armijo. Minimization of functions having lipschitz continuous first partial derivatives. *Pacific Journal of mathematics*, 16(1):1–3, 1966.

Peter Auer, Nicolo Cesa-Bianchi, and Claudio Gentile. Adaptive and self-confident on-line learning algorithms. *Journal of Computer and System Sciences*, 64(1):48–75, 2002.

Waïss Azizian, Ioannis Mitliagkas, Simon Lacoste-Julien, and Gauthier Gidel. A tight and unified analysis of extragradient for a whole spectrum of differentiable games. *arXiv preprint arXiv:1906.05945*, 2019.

Francis Bach and Kfir Y Levy. A universal algorithm for variational inequalities adaptive to smoothness and noise. In *COLT*, pages 164–194. PMLR, 2019.

P Balamurugan and Francis Bach. Stochastic variance reduction methods for saddle-point problems. In *Proceedings of the 30th International Conference on Neural Information Processing Systems*, pages 1416–1424, 2016.

Lukas Balles and Philipp Hennig. Dissecting adam: The sign, magnitude and variance of stochastic gradients. In *International Conference on Machine Learning*, pages 404–413. PMLR, 2018.

Babak Barazandeh and Meisam Razaviyayn. Solving non-convex non-differentiable min-max games using proximal gradient method. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3162–3166. IEEE, 2020.

Amir Beck. *First-order methods in optimization*. SIAM, 2017.

Y. Bengio, P. Simard, and P. Frasconi. Learning long-term dependencies with gradient descent is difficult. *IEEE Transactions on Neural Networks*, 5(2):157–166, 1994.

Yoshua Bengio. Learning deep architectures for AI. *Foundations and Trends in Machine Learning*, 2(1):1–127, 2009.

Julius R Blum. Multidimensional stochastic approximation methods. *The Annals of Mathematical Statistics*, pages 737–744, 1954.

Radu Ioan Boţ and Axel Böhm. Alternating proximal-gradient steps for (stochastic) nonconvex-concave minimax problems. *arXiv preprint arXiv:2007.13605*, 2020.

Léon Bottou, Frank E Curtis, and Jorge Nocedal. Optimization methods for large-scale machine learning. *Siam Review*, 60(2):223–311, 2018.

Stephen Boyd, Stephen P Boyd, and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Sébastien Bubeck. Convex optimization: Algorithms and complexity. *Foundations and Trends in Machine Learning*, 8, 2017.

Qi Cai, Mingyi Hong, Yongxin Chen, and Zhaoran Wang. On the global convergence of imitation learning: A case for linear quadratic regulator. *arXiv preprint arXiv:1901.03674*, 2019.

Yair Carmon and Oliver Hinder. Making sgd parameter-free. *arXiv preprint arXiv:2205.02160*, 2022.

Yair Carmon, Yujia Jin, Aaron Sidford, and Kevin Tian. Variance reduction for matrix games. *Advances in Neural Information Processing Systems*, 32, 2019.

Yair Carmon, John C Duchi, Oliver Hinder, and Aaron Sidford. Lower bounds for finding stationary points i. *Mathematical Programming*, 184(1):71–120, 2020.

Matteo Cassotti, Davide Ballabio, Viviana Consonni, Andrea Mauri, Igor V Tetko, and Roberto Todeschini. Prediction of acute aquatic toxicity toward daphnia magna by using the ga-k nn method. *Alternatives to Laboratory Animals*, 42(1):31–41, 2014.

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.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: A library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):1–27, 2011.

Tatjana Chavdarova, Gauthier Gidel, François Fleuret, and Simon Lacoste-Julien. Reducing noise in gan training with variance reduced extragradient. In *Advances in Neural Information Processing Systems*, pages 391–401, 2019.

Robert S Chen, Brendan Lucier, Yaron Singer, and Vasilis Syrgkanis. Robust optimization for non-convex objectives. In *Advances in Neural Information Processing Systems*, pages 4705–4714, 2017.

Tianyi Chen, Yuejiao Sun, and Wotao Yin. A single-timescale stochastic bilevel optimization method. *arXiv preprint arXiv:2102.04671*, 2021a.

Tianyi Chen, Yuejiao Sun, and Wotao Yin. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021b.

Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. On the convergence of a class of adam-type algorithms for non-convex optimization. In *International Conference on Learning Representations*. International Conference on Learning Representations, ICLR, 2019.

Yunmei Chen, Guanghui Lan, and Yuyuan Ouyang. Optimal primal-dual methods for a class of saddle point problems. *SIAM Journal on Optimization*, 24(4):1779–1814, 2014.

Ziyi Chen and Yi Zhou. Escaping saddle points in nonconvex minimax optimization via cubic-regularized gradient descent-ascent. *arXiv preprint arXiv:2110.07098*, 2021.

Kai Lai Chung. On a stochastic approximation method. *The Annals of Mathematical Statistics*, pages 463–483, 1954.

Ashok Cutkosky and Kwabena A Boahen. Stochastic and adversarial online learning without hyperparameters. In *NeurIPS*, volume 30, 2017.

Ashok Cutkosky and Harsh Mehta. Momentum improves normalized sgd. In *International conference on machine learning*, pages 2260–2268. PMLR, 2020.

Ashok Cutkosky and Francesco Orabona. Black-box reductions for parameter-free online learning in banach spaces. In *COLT*, pages 1493–1529. PMLR, 2018.

Bo Dai, Niao He, Yunpeng Pan, Byron Boots, and Le Song. Learning from conditional distributions via dual embeddings. In *Artificial Intelligence and Statistics*, pages 1458–1467, 2017.

Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. *arXiv preprint arXiv:1807.03907*, 2018.

Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training gans with optimism. 2018.

Constantinos Daskalakis, Stratis Skoulakis, and Manolis Zampetakis. The complexity of constrained min-max optimization. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 1466–1478, 2021.

Felipe RS De Gusmao, Edwin MM Ortega, and Gauss M Cordeiro. The generalized inverse weibull distribution. *Statistical Papers*, 52:591–619, 2011.

Alexandre Défossez, Léon Bottou, Francis Bach, and Nicolas Usunier. A simple convergence proof of adam and adagrad. *arXiv preprint arXiv:2003.02395*, 2020.

Jelena Diakonikolas. Halpern iteration for near-optimal and parameter-free monotone inclusion and strong solutions to variational inequalities. In *Conference on Learning Theory*, pages 1428–1451. PMLR, 2020.

Emily Diana, Wesley Gill, Michael Kearns, Krishnaram Kenthapadi, and Aaron Roth. Minimax group fairness: Algorithms and experiments. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 66–76, 2021.

Zehao Dou and Yuanzhi Li. On the one-sided convergence of adam-type algorithms in non-convex non-concave min-max optimization. *arXiv preprint arXiv:2109.14213*, 2021.

Yoel Drori and Ohad Shamir. The complexity of finding stationary points with stochastic gradient descent. In *International Conference on Machine Learning*, pages 2658–2667. PMLR, 2020.

Dmitriy Drusvyatskiy and Courtney Paquette. Efficiency of minimizing compositions of convex functions and smooth maps. *Mathematical Programming*, 178(1-2):503–558, 2019.

Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685, 2019.

Simon S Du and Wei Hu. Linear convergence of the primal-dual gradient method for convex-concave saddle point problems without strong convexity. pages 196–205, 2019.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of machine learning research*, 12(7), 2011.

Laurent El Ghaoui and Hervé Lebret. Robust solutions to least-squares problems with uncertain data. *SIAM Journal on matrix analysis and applications*, 18(4):1035–1064, 1997.

Alina Ene and Huy L Nguyen. Adaptive and universal algorithms for variational inequalities with optimal convergence. *arXiv preprint arXiv:2010.07799*, 2020.

Francisco Facchinei and Jong-Shi Pang. *Finite-dimensional variational inequalities and complementarity problems*. Springer Science & Business Media, 2007.

Ilyas Fatkhullin, Anas Barakat, Anastasia Kireeva, and Niao He. Stochastic policy gradient methods: Improved sample complexity for fisher-non-degenerate policies. *arXiv preprint arXiv:2302.01734*, 2023.

Matthew Faw, Isidoros Tziotis, Constantine Caramanis, Aryan Mokhtari, Sanjay Shakkottai, and Rachel Ward. The power of adaptivity in sgd: Self-tuning step sizes with unbounded gradients and affine variance. In *Proceedings of Thirty Fifth Conference on Learning Theory*, volume 178 of *Proceedings of Machine Learning Research*, pages 313–355. PMLR, Jul 2022.

Maryam Fazel, Rong Ge, Sham Kakade, and Mehran Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *International Conference on Machine Learning*, pages 1467–1476, 2018.

Tanner Fiez and Lillian Ratliff. Gradient descent-ascent provably converges to strict local minmax equilibria with a finite timescale separation. *arXiv preprint arXiv:2009.14820*, 2020.

Tanner Fiez, Benjamin Chasnov, and Lillian J Ratliff. Convergence of learning dynamics in stackelberg games. *arXiv preprint arXiv:1906.01217*, 2019.

Tanner Fiez, Benjamin Chasnov, and Lillian Ratliff. Implicit learning dynamics in stackelberg games: Equilibria characterization, convergence analysis, and empirical study. In *International Conference on Machine Learning (ICML)*, 2020.

Tanner Fiez, Lillian J Ratliff, Eric Mazumdar, Evan Faulkner, and Adhyyan Narang. Global convergence to local minmax equilibrium in classes of nonconvex zero-sum games. 2021.

Xavier Fontaine, Valentin De Bortoli, and Alain Durmus. Convergence rates and approximation results for sgd and its continuous-time counterpart. In *COLT*, pages 1965–2058. PMLR, 2021.

Pierre Foret, Ariel Kleiner, Hossein Mobahi, and Behnam Neyshabur. Sharpness-aware minimization for efficiently improving generalization. In *ICLR*, 2021.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

Andrey Garnaev and Wade Trappe. An eavesdropping game with sinr as an objective function. In *International Conference on Security and Privacy in Communication Systems*, pages 142–162. Springer, 2009.

Alexander Vladimirovich Gasnikov, PE Dvurechensky, Fedor Sergeevich Stonyakin, and Aleksandr Aleksandrovich Titov. An adaptive proximal method for variational inequalities. *Computational Mathematics and Mathematical Physics*, 59(5):836–841, 2019.

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

Gauthier Gidel, Hugo Berard, Gaëtan Vignoud, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial networks. *arXiv preprint arXiv:1802.10551*, 2018.

Gauthier Gidel, Reyhane Askari Hemmat, Mohammad Pezeshki, Rémi Le Priol, Gabriel Huang, Simon Lacoste-Julien, and Ioannis Mitliagkas. Negative momentum for improved game dynamics. pages 1802–1811, 2019.

Boris Ginsburg, Patrice Castonguay, Oleksii Hrinchuk, Oleksii Kuchaiev, Vitaly Lavrukhin, Ryan Leary, Jason Li, Huyen Nguyen, Yang Zhang, and Jonathan M Cohen. Stochastic gradient methods with layer-wise adaptive moments for training of deep networks. *arXiv preprint arXiv:1905.11286*, 2019.

Ian Goodfellow. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. 2015.

Benjamin Grimmer, Haihao Lu, Pratik Worah, and Vahab Mirrokni. The landscape of the proximal point method for nonconvex-nonconcave minimax optimization. *arXiv preprint arXiv:2006.08667*, 2020.

Osman Güler. On the convergence of the proximal point algorithm for convex minimization. *SIAM Journal on Control and Optimization*, 29(2):403–419, 1991.

Osman Güler. New proximal point algorithms for convex minimization. *SIAM Journal on Optimization*, 2(4):649–664, 1992.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. *NeurIPS*, 30, 2017.

Zhishuai Guo and Tianbao Yang. Randomized stochastic variance-reduced methods for stochastic bilevel optimization. *arXiv preprint arXiv:2105.02266*, 2021.

Zhishuai Guo, Zhuoning Yuan, Yan Yan, and Tianbao Yang. Fast objective and duality gap convergence for non-convex strongly-concave min-max problems. *arXiv preprint arXiv:2006.06889*, 2020.

Zhishuai Guo, Yi Xu, Wotao Yin, Rong Jin, and Tianbao Yang. A novel convergence analysis for algorithms of the adam family and beyond. *arXiv preprint arXiv:2104.14840*, 2021a.

Zhishuai Guo, Yi Xu, Wotao Yin, Rong Jin, and Tianbao Yang. On stochastic moving-average estimators for non-convex optimization. *arXiv preprint arXiv:2104.14840*, 2021b.

Erfan Yazdandoost Hamedani and Necdet Serhat Aybat. A primal-dual algorithm for general convex-concave saddle point problems. *arXiv preprint arXiv:1803.01401*, 2018.

Yuze Han, Guangzeng Xie, and Zhihua Zhang. Lower complexity bounds of finite-sum optimization problems: The results and construction. *arXiv preprint arXiv:2103.08280*, 2021.

Elad Hazan and Sham Kakade. Revisiting the polyak step size. *arXiv preprint arXiv:1905.00313*, 2019.

Elad Hazan, Kfir Levy, and Shai Shalev-Shwartz. Beyond convexity: Stochastic quasi-convex optimization. *Advances in neural information processing systems*, 28, 2015.

Bingsheng He and Xiaoming Yuan. An accelerated inexact proximal point algorithm for convex minimization. *Journal of Optimization Theory and Applications*, 154(2):536–548, 2012.

Niao He, Anatoli Juditsky, and Arkadi Nemirovski. Mirror prox algorithm for multi-term composite minimization and semi-separable problems. *Computational Optimization and Applications*, 61(2):275–319, 2015.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in Neural Information Srocessing Systems*, 30, 2017.

Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Neural networks for machine learning lecture 6a overview of mini-batch gradient descent. 2012.

Mingyi Hong, Hoi-To Wai, Zhaoran Wang, and Zhuoran Yang. A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *arXiv preprint arXiv:2007.05170*, 2020.

Ya-Ping Hsieh, Panayotis Mertikopoulos, and Volkan Cevher. The limits of min-max optimization algorithms: Convergence to spurious non-critical sets. pages 4337–4348, 2021.

Feihu Huang and Heng Huang. Adagda: Faster adaptive gradient descent ascent methods for minimax optimization. *arXiv preprint arXiv:2106.16101*, 2021.

Feihu Huang, Shangqian Gao, Jian Pei, and Heng Huang. Accelerated zeroth-order momentum methods from mini to minimax optimization. *arXiv e-prints*, pages arXiv–2008, 2020.

Feihu Huang, Xidong Wu, and Heng Huang. Efficient mirror descent ascent methods for nonsmooth minimax problems. *NeurIPS*, 34, 2021.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017.

Arthur Jacot, Franck Gabriel, and Clément Hongler. Neural tangent kernel: Convergence and generalization in neural networks. *arXiv preprint arXiv:1806.07572*, 2018.

Kaiyi Ji, Junjie Yang, and Yingbin Liang. Bilevel optimization: Nonasymptotic analysis and faster algorithms. *arXiv preprint arXiv:2010.07962*, 2020.

Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International conference on machine learning*, pages 1724–1732. PMLR, 2017.

Chi Jin, Praneeth Netrapalli, and Michael Jordan. What is local optimality in nonconvex-nonconcave minimax optimization? pages 4880–4889, 2020.

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

Anatoli Juditsky, Arkadi Nemirovski, et al. First order methods for nonsmooth convex large-scale optimization, i: general purpose methods. *Optimization for Machine Learning*, pages 121–148, 2011.

Sham M Kakade and Ambuj Tewari. On the generalization ability of online strongly convex programming algorithms. *NeurIPS*, 21, 2008.

Myeongmin Kang, Myungjoo Kang, and Miyoun Jung. Inexact accelerated augmented lagrangian methods. *Computational Optimization and Applications*, 62(2):373–404, 2015.

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

Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. In *International Conference on Machine Learning*, pages 3252–3261. PMLR, 2019.

Ali Kavis, Kfir Y Levy, Francis Bach, and Volkan Cevher. Unixgrad: A universal, adaptive algorithm with optimal guarantees for constrained optimization. *NeurIPS*, 32, 2019.

Ali Kavis, Kfir Yehuda Levy, and Volkan Cevher. High probability bounds for a class of nonconvex algorithms with adagrad stepsize. In *ICLR*, 2022a.

Ali Kavis, Stratis Skoulakis, Kimon Antonakopoulos, Leello Tadesse Dadi, and Volkan Cevher. Adaptive stochastic variance reduction for non-convex finite-sum minimization. *arXiv preprint arXiv:2211.01851*, 2022b.

Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

Ahmed Khaled and Peter Richtárik. Better theory for sgd in the nonconvex world. *arXiv preprint arXiv:2002.03329*, 2020.

J. Kiefer and J. Wolfowitz. Stochastic Estimation of the Maximum of a Regression Function. *The Annals of Mathematical Statistics*, 23(3):462 – 466, 1952.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Weiwei Kong and Renato DC Monteiro. An accelerated inexact proximal point method for solving nonconvex-concave min-max problems. *arXiv preprint arXiv:1905.13433*, 2019.

GM Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

Frederik Kunstner, Jacques Chen, Jonathan Wilder Lavington, and Mark Schmidt. Heavy-tailed noise does not explain the gap between SGD and Adam, but sign descent might. In *International Conference on Learning Representations*, 2023.

Jungmin Kwon, Jeongseop Kim, Hyunseo Park, and In Kwon Choi. Asam: Adaptive sharpness-aware minimization for scale-invariant learning of deep neural networks. In *International Conference on Machine Learning*, pages 5905–5914. PMLR, 2021.

Simon Lacoste-Julien, Mark Schmidt, and Francis Bach. A simpler approach to obtaining an o (1/t) convergence rate for the projected stochastic subgradient method. *arXiv preprint arXiv:1212.2002*, 2012.

Yann LeCun. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*, 1998.

Jaehoon Lee, Yasaman Bahri, Roman Novak, Samuel S Schoenholz, Jeffrey Pennington, and Jascha Sohl-Dickstein. Deep neural networks as gaussian processes. *arXiv preprint arXiv:1711.00165*, 2017.

Joffrey L Leevy and Taghi M Khoshgoftaar. A survey and analysis of intrusion detection models based on cse-cic-ids2018 big data. *Journal of Big Data*, 7(1):1–19, 2020.

Qi Lei, Jason Lee, Alex Dimakis, and Constantinos Daskalakis. Sgd learns one-layer networks in wgans. In *International Conference on Machine Learning*, pages 5799–5808. PMLR, 2020.

Yunwen Lei, Ting Hu, Guiying Li, and Ke Tang. Stochastic gradient descent for nonconvex learning without bounded gradient assumptions. *IEEE transactions on neural networks and learning systems*, 31(10):4394–4400, 2019.

Kfir Levy. Online to offline conversions, universality and adaptive minibatch sizes. *NeurIPS*, 30, 2017.

Kfir Levy, Ali Kavis, and Volkan Cevher. Storm+: Fully adaptive sgd with recursive momentum for nonconvex optimization. *NeurIPS*, 34:20571–20582, 2021.

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

Kfir Y Levy, Alp Yurtsever, and Volkan Cevher. Online adaptive methods, universality and acceleration. *NeurIPS*, 31, 2018.

Haochuan Li, Yi Tian, Jingzhao Zhang, and Ali Jadbabaie. Complexity lower bounds for nonconvex-strongly-concave min-max optimization. In *Thirty-Fifth Conference on Neural Information Processing Systems*, 2021.

Xiang Li. Adaptive methods for parameter-agnostic nonconvex minimax optimization. Master's thesis, ETH Zurich, Department of Computer Science, 2022.

Xiang Li, Yang Junchi, and Niao He. Tiada: A time-scale adaptive algorithm for nonconvex minimax optimization. In *The Eleventh International Conference on Learning Representations*, 2023.

Xiaoyu Li and Francesco Orabona. On the convergence of stochastic gradient descent with adaptive stepsizes. In *AISTATS*, pages 983–992. PMLR, 2019.

Xiaoyu Li and Francesco Orabona. A high probability analysis of adaptive sgd with momentum. *arXiv preprint arXiv:2007.14294*, 2020.

Hongzhou Lin, Julien Mairal, and Zaid Harchaoui. A universal catalyst for first-order optimization. In *Advances in neural information processing systems*, pages 3384–3392, 2015.

Hongzhou Lin, Julien Mairal, and Zaid Harchaoui. Catalyst acceleration for first-order convex optimization: from theory to practice. *The Journal of Machine Learning Research*, 18 (1):7854–7907, 2017.

Qihang Lin, Mingrui Liu, Hassan Rafique, and Tianbao Yang. Solving weakly-convex-weakly-concave saddle-point problems as successive strongly monotone variational inequalities. *arXiv preprint arXiv:1810.10207*, 2018.

Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. pages 6083–6093, 2020a.

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

Selena Ling, Nicholas Sharp, and Alec Jacobson. Vectoradam for rotation equivariant geometry optimization. *arXiv preprint arXiv:2205.13599*, 2022.

Chaoyue Liu, Libin Zhu, and Mikhail Belkin. Loss landscapes and optimization in over-parameterized non-linear systems and neural networks. *arXiv preprint arXiv:2003.00307*, 2020a.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. In *International Conference on Learning Representations*, 2020b.

Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. On the variance of the adaptive learning rate and beyond. In *ICLR*, 2020c.

Liyuan Liu, Xiaodong Liu, Jianfeng Gao, Weizhu Chen, and Jiawei Han. Understanding the difficulty of training transformers. *arXiv preprint arXiv:2004.08249*, 2020d.

Ming-Yu Liu and Oncel Tuzel. Coupled generative adversarial networks. In *Advances in neural information processing systems*, pages 469–477, 2016.

Mingrui Liu, Youssef Mroueh, Jerret Ross, Wei Zhang, Xiaodong Cui, Payel Das, and Tianbao Yang. Towards better understanding of adaptive gradient algorithms in generative adversarial nets. In *International Conference on Learning Representations*, 2019.

Sijia Liu, Songtao Lu, Xiangyi Chen, Yao Feng, Kaidi Xu, Abdullah Al-Dujaili, Mingyi Hong, and Una-May O'Reilly. Min-max optimization without gradients: Convergence and applications to black-box evasion and poisoning attacks. In *International Conference on Machine Learning*, pages 6282–6293. PMLR, 2020e.

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.

Nicolas Loizou, Hugo Berard, Gauthier Gidel, Ioannis Mitliagkas, and Simon Lacoste-Julien. Stochastic gradient descent-ascent and consensus optimization for smooth games: Convergence analysis under expected co-coercivity. *Advances in Neural Information Processing Systems*, 34, 2021.

Haihao Lu. An $o(s^r)$-resolution ode framework for understanding discrete-time algorithms and applications to the linear convergence of minimax problems. *Mathematical Programming*, pages 1–52, 2021.

Songtao Lu, Ioannis Tsaknakis, Mingyi Hong, and Yongxin Chen. Hybrid block successive approximation for one-sided non-convex min-max problems: algorithms and applications. *IEEE Transactions on Signal Processing*, 2020.

Luo Luo and Cheng Chen. Finding second-order stationary point for nonconvex-strongly-concave minimax problem. *arXiv preprint arXiv:2110.04814*, 2021.

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

Luo Luo, Haishan Ye, and Tong Zhang. Stochastic recursive gradient descent ascent for stochastic nonconvex-strongly-concave minimax problems. *arXiv preprint arXiv:2001.03724*, 2020.

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.

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

Louis Lv. Reproducing "certifying some distributional robustness with principled adversarial training". `https://github.com/Louis-udm/Reproducing-certifiable-distributional-robustness`, 2019.

Siyuan Ma, Raef Bassily, and Mikhail Belkin. The power of interpolation: Understanding the effectiveness of sgd in modern over-parametrized learning. In *International Conference on Machine Learning*, pages 3325–3334. PMLR, 2018.

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

Yura Malitsky. Golden ratio algorithms for variational inequalities. *Mathematical Programming*, 184(1):383–410, 2020.

Natalia Martinez, Martin Bertran, and Guillermo Sapiro. Minimax pareto fairness: A multi objective perspective. In *International Conference on Machine Learning*, pages 6755–6764. PMLR, 2020.

Eric Mazumdar and Lillian J Ratliff. On the convergence of gradient-based learning in continuous games. *ArXiv e-prints*, 2018.

Eric Mazumdar, Lillian J Ratliff, and S Shankar Sastry. On gradient-based learning in continuous games. *SIAM Journal on Mathematics of Data Science*, 2(1):103–131, 2020.

H Brendan McMahan and Matthew Streeter. Adaptive bound optimization for online convex optimization. *arXiv preprint arXiv:1002.4908*, 2010.

Ninareh Mehrabi, Fred Morstatter, Nripsuta Saxena, Kristina Lerman, and Aram Galstyan. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54 (6):1–35, 2021.

Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 1823–1833, 2017.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. Which training methods for gans do actually converge? In *International Conference on Machine Learning*, pages 3481–3490, 2018.

Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. Unrolled generative adversarial networks. *arXiv preprint arXiv:1611.02163*, 2016.

David J Miller, Zhen Xiang, and George Kesidis. Adversarial learning targeting deep neural network classification: A comprehensive review of defenses against attacks. *Proceedings of the IEEE*, 108(3):402–433, 2020.

Konstantin Mishchenko, Dmitry Kovalev, Egor Shulgin, Peter Richtárik, and Yura Malitsky. Revisiting stochastic extragradient. In *International Conference on Artificial Intelligence and Statistics*, pages 4573–4582. PMLR, 2020.

Aditya Modi, Jinglin Chen, Akshay Krishnamurthy, Nan Jiang, and Alekh Agarwal. Model-free representation learning and exploration in low-rank mdps. *arXiv preprint arXiv:2102.07035*, 2021.

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 arXiv:1901.08511*, 2019.

Renato DC Monteiro and Benar Fux Svaiter. On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean. *SIAM Journal on Optimization*, 20(6):2755–2787, 2010.

Eric Moulines and Francis Bach. Non-asymptotic analysis of stochastic approximation algorithms for machine learning. *NeurIPS*, 24, 2011.

Youssef Mroueh, Chun-Liang Li, Tom Sercu, Anant Raj, and Yu Cheng. Sobolev gan. *arXiv preprint arXiv:1711.04894*, 2017.

Mahesh Chandra Mukkamala and Matthias Hein. Variants of rmsprop and adagrad with logarithmic regret bounds. In *ICML*, pages 2545–2553. PMLR, 2017.

Ryan Murray, Brian Swenson, and Soummya Kar. Revisiting normalized gradient descent: Fast evasion of saddle points. *IEEE Transactions on Automatic Control*, 64(11):4818–4824, 2019.

Vaishnavh Nagarajan and J Zico Kolter. Gradient descent gan optimization is locally stable. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pages 5591–5600, 2017.

Hongseok Namkoong and John C Duchi. Stochastic gradient methods for distributionally robust optimization with f-divergences. In *Advances in neural information processing systems*, pages 2208–2216, 2016.

John Nash. Two-person cooperative games. *Econometrica: Journal of the Econometric Society*, pages 128–140, 1953.

Ion Necoara, Yu Nesterov, and Francois Glineur. Linear convergence of first order methods for non-strongly convex optimization. *Mathematical Programming*, pages 1–39, 2018.

A. S. Nemirovski and D. B. Yudin. *Problem Complexity and Method Efficiency in Optimization*. John Wiley UK/USA, 1983.

Arkadi Nemirovski. Prox-method with rate of convergence o (1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

Arkadi Nemirovsky and David Yudin. Problem complexity and method efficiency in optimization. 1983.

Yu Nesterov. Smooth minimization of non-smooth functions. *Mathematical programming*, 103(1):127–152, 2005.

Yurii Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109(2-3):319–344, 2007.

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

Yurii Nesterov and Laura Scrimali. Solving strongly monotone variational and quasi-variational inequalities. *Available at SSRN 970903*, 2006.

Yurii E Nesterov. Minimization methods for nonsmooth convex and quasiconvex functions. *Matekon*, 29(3):519–531, 1984.

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.

Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. In *Advances in Neural Information Processing Systems*, pages 14905–14916, 2019.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. *Advances in neural information processing systems*, 29, 2016.

Francesco Orabona and Dávid Pál. Scale-free online learning. *Theoretical Computer Science*, 716:50–69, 2018.

Antonio Orvieto, Simon Lacoste-Julien, and Nicolas Loizou. Dynamics of sgd with stochastic polyak stepsizes: Truly adaptive variants and convergence to exact solution. *arXiv preprint arXiv:2205.04583*, 2022.

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.

Yuyuan Ouyang and Yangyang Xu. Lower complexity bounds of first-order methods for convex-concave bilinear saddle-point problems. *Mathematical Programming*, pages 1–35, 2019.

Balamurugan Palaniappan and Francis Bach. Stochastic variance reduction methods for saddle-point problems. In *Advances in Neural Information Processing Systems*, pages 1416–1424, 2016.

Werner Palfinger. Continuous remeshing for inverse rendering. *Computer Animation and Virtual Worlds*, 33(5):e2101, 2022.

Jong-Shi Pang. A posteriori error bounds for the linearly-constrained variational inequality problem. *Mathematics of Operations Research*, 12(3):474–484, 1987.

Courtney Paquette, Hongzhou Lin, Dmitriy Drusvyatskiy, Julien Mairal, and Zaid Harchaoui. Catalyst acceleration for gradient-based non-convex optimization. *arXiv preprint arXiv:1703.10993*, 2017.

Panos M Pardalos and Stephen A Vavasis. Quadratic programming with one negative eigenvalue is np-hard. *Journal of Global optimization*, 1(1):15–22, 1991.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pages 1310–1318. PMLR, 2013.

Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM journal on control and optimization*, 30(4):838–855, 1992.

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

Qi Qian, Shenghuo Zhu, Jiasheng Tang, Rong Jin, Baigui Sun, and Hao Li. Robust optimization over multiple domains. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4739–4746, 2019.

Hassan Rafique, Mingrui Liu, Qihang Lin, and Tianbao Yang. Weakly-convex–concave min–max optimization: provable algorithms and applications in machine learning. *Optimization Methods and Software*, 37(3):1087–1121, 2022.

Alexander Rakhlin, Ohad Shamir, and Karthik Sridharan. Making gradient descent optimal for strongly convex stochastic optimization. pages 1571–1578, 2012.

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

Sashank J Reddi, Suvrit Sra, Barnabás Póczos, and Alex Smola. Fast incremental method for smooth nonconvex optimization. In *2016 IEEE 55th Conference on Decision and Control (CDC)*, pages 1971–1977. IEEE, 2016b.

Sashank J Reddi, Satyen Kale, and Sanjiv Kumar. On the convergence of adam and beyond. 2019.

Amirhossein Reisizadeh, Farzan Farnia, Ramtin Pedarsani, and Ali Jadbabaie. Robust federated learning: The case of affine distribution shifts. *Advances in Neural Information Processing Systems*, 33:21554–21565, 2020.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

R Tyrrell Rockafellar. Augmented lagrangians and applications of the proximal point algorithm in convex programming. *Mathematics of operations research*, 1(2):97–116, 1976a.

R Tyrrell Rockafellar. Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898, 1976b.

Saverio Salzo and Silvia Villa. Inexact and accelerated proximal point algorithms. *Journal of Convex analysis*, 19(4):1167–1192, 2012.

Kevin Schawinski, Ce Zhang, Hantian Zhang, Lucas Fowler, and Gokula Krishnan Santhanam. Generative adversarial networks recover features in astrophysical images of galaxies beyond the deconvolution limit. *Monthly Notices of the Royal Astronomical Society: Letters*, 467(1):L110–L114, 2017.

Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61: 85–117, 2015.

Othmane Sebbouh, Marco Cuturi, and Gabriel Peyré. Randomized stochastic gradient descent ascent. In *AISTATS*, pages 2941–2969. PMLR, 2022.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Shai Shalev-Shwartz and Tong Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. In *International conference on machine learning*, pages 64–72, 2014.

Naichen Shi and Dawei Li. Rmsprop converges with proper hyperparameter. In *International conference on learning representation*, 2021.

M. Sibony. Méthodes itératives pour les équations et iné equations aux dérivées partielles non linéaires de type monotone. *Calcolo*, 7(1-2):65–183, 1970.

Aman Sinha, Hongseok Namkoong, and John Duchi. Certifiable distributional robustness with principled adversarial training. *stat*, 1050:29, 2017.

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

Fedor Stonyakin, Alexander Gasnikov, Pavel Dvurechensky, Mohammad Alkousa, and Alexander Titov. Generalized mirror prox for monotone variational inequalities: Universality and inexact oracle. *arXiv preprint arXiv:1806.05140*, 2018.

Matthew Streeter and H Brendan McMahan. Less regret via online conditioning. *arXiv preprint arXiv:1002.4862*, 2010.

Ju Sun, Qing Qu, and John Wright. A geometric analysis of phase retrieval. *Foundations of Computational Mathematics*, 18(5):1131–1198, 2018.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Conghui Tan, Tong Zhang, Shiqian Ma, and Ji Liu. Stochastic primal-dual method for empirical risk minimization with $o(1)$ per-iteration complexity. In *Advances in Neural Information Processing Systems*, pages 8366–8375, 2018.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

Kiran K Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Efficient algorithms for smooth minimax optimization. In *Advances in Neural Information Processing Systems*, pages 12659–12670, 2019.

Tijmen Tieleman, Geoffrey Hinton, et al. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning*, 4(2):26–31, 2012.

Quoc Tran-Dinh, Deyi Liu, and Lam M Nguyen. Hybrid variance-reduced sgd algorithms for nonconvex-concave minimax problems. *arXiv preprint arXiv:2006.15266*, 2020.

Paul Tseng. On linear convergence of iterative methods for the variational inequality problem. *Journal of Computational and Applied Mathematics*, 60(1-2):237–252, 1995.

Sharan Vaswani, Aaron Mishkin, Issam Laradji, Mark Schmidt, Gauthier Gidel, and Simon Lacoste-Julien. Painless stochastic gradient: Interpolation, line-search, and convergence rates. *NeurIPS*, 32, 2019.

Sharan Vaswani, Frederik Kunstner, Issam H. Laradji, Si Yi Meng, Mark W. Schmidt, and Simon Lacoste-Julien. Adaptive gradient methods converge faster with over-parameterization (and you can do a line-search). *ArXiv*, abs/2006.06835, 2020.

Sharan Vaswani, Benjamin Dubois-Taine, and Reza Babanezhad. Towards noise-adaptive, problem-adaptive stochastic gradient descent. *arXiv preprint arXiv:2110.11442*, 2021.

John Von Neumann, Oskar Morgenstern, and Harold William Kuhn. *Theory of games and economic behavior (commemorative edition)*. Princeton university press, 2007.

Bohan Wang, Yushun Zhang, Huishuai Zhang, Qi Meng, Zhi-Ming Ma, Tie-Yan Liu, and Wei Chen. Provable adaptivity in adam. *arXiv preprint arXiv:2208.09900*, 2022.

Guanghui Wang, Shiyin Lu, Quan Cheng, Wei-wei Tu, and Lijun Zhang. Sadam: A variant of adam for strongly convex functions. In *ICLR*, 2020a.

Xiaoyu Wang, Sindri Magnússon, and Mikael Johansson. On the convergence of step decay step-size for stochastic optimization. *Advances in Neural Information Processing Systems*, 34:14226–14238, 2021.

Yuanhao Wang and Jian Li. Improved algorithms for convex-concave minimax optimization. *arXiv preprint arXiv:2006.06359*, 2020.

Zhongruo Wang, Krishnakumar Balasubramanian, Shiqian Ma, and Meisam Razaviyayn. Zeroth-order algorithms for nonconvex minimax problems with improved complexities. *arXiv preprint arXiv:2001.07819*, 2020b.

Rachel Ward, Xiaoxia Wu, and Leon Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. In *ICML*, pages 6677–6686. PMLR, 2019.

Rachel Ward, Xiaoxia Wu, and Leon Bottou. Adagrad stepsizes: Sharp convergence over nonconvex landscapes. *The Journal of Machine Learning Research*, 21(1):9047–9076, 2020.

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

Jiahao Xie, Chao Zhang, Yunsong Zhang, Zebang Shen, and Hui Qian. A federated learning framework for nonconvex-pl minimax problems. *arXiv preprint arXiv:2105.14216*, 2021.

Yuege Xie, Xiaoxia Wu, and Rachel Ward. Linear convergence of adaptive stochastic gradient descent. In *AISTATS*, pages 1475–1485. PMLR, 2020.

Zeke Xie, Xinrui Wang, Huishuai Zhang, Issei Sato, and Masashi Sugiyama. Adaptive inertia: Disentangling the effects of adaptive learning rate and momentum. In *International Conference on Machine Learning*, pages 24430–24459. PMLR, 2022.

Zhipeng Xie and Jianwen Shi. Accelerated primal dual method for a class of saddle point problem with strongly convex component. *arXiv preprint arXiv:1906.07691*, 2019.

Tengyu Xu, Zhe Wang, Yingbin Liang, and H Vincent Poor. Enhanced first and zeroth order variance reduced algorithms for min-max optimization. *arXiv preprint arXiv:2006.09361*, 2020a.

Tengyu Xu, Zhe Wang, Yingbin Liang, and H Vincent Poor. Gradient free minimax optimization: Variance reduction and faster convergence. *arXiv preprint arXiv:2006.09361*, 2020b.

Zi Xu, Huiling Zhang, Yang Xu, and Guanghui Lan. A unified single-loop alternating gradient projection algorithm for nonconvex-concave and convex-nonconcave minimax problems. *arXiv preprint arXiv:2006.02032*, 2020c.

Zi Xu, Jingjing Shen, Ziqi Wang, and Yuhong Dai. Zeroth-order alternating randomized gradient projection algorithms for general nonconvex-concave minimax problems. *arXiv preprint arXiv:2108.00473*, 2021.

Yan Yan, Yi Xu, Qihang Lin, Wei Liu, and Tianbao Yang. Optimal epoch stochastic gradient descent ascent methods for min-max optimization. *Advances in Neural Information Processing Systems*, 33, 2020.

Junchi Yang, Negar Kiyavash, and Niao He. Global convergence and variance-reduced optimization for a class of nonconvex-nonconcave minimax problems. *Advances in Neural Information Processing Systems*, 2020a.

Junchi Yang, Siqi Zhang, Negar Kiyavash, and Niao He. A catalyst framework for minimax optimization. *Advances in Neural Information Processing Systems*, 33, 2020b.

Junchi Yang, Xiang Li, and Niao He. Nest your adaptive algorithm for parameter-agnostic nonconvex minimax optimization. *Advances in Neural Information Processing Systems*, 2022a.

Junchi Yang, Antonio Orvieto, Aurelien Lucchi, and Niao He. Faster single-loop algorithms for minimax optimization without strong concavity. In *AISTATS*, pages 5485–5517. PMLR, 2022b.

Junchi Yang, Xiang Li, Ilyas Fatkhullin, and Niao He. Two sides of one coin: the limits of untuned sgd and the power of adaptive methods. *Advances in Neural Information Processing Systems*, 2023.

Manzil Zaheer, Sashank Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. Adaptive methods for nonconvex optimization. *NeurIPS*, 31, 2018.

Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

Guodong Zhang, Yuanhao Wang, Laurent Lessard, and Roger Grosse. Don't fix what ain't broke: Near-optimal local convergence of alternating gradient descent-ascent for minimax optimization. *arXiv preprint arXiv:2102.09468*, 2021a.

Guojun Zhang. Understanding minimax optimization in modern machine learning. 2021a.

Hui Zhang and Wotao Yin. Gradient methods for convex minimization: better rates under weaker conditions. *arXiv preprint arXiv:1303.4645*, 2013.

Jiawei Zhang, Peijun Xiao, Ruoyu Sun, and Zhi-Quan Luo. A single-loop smoothed gradient descent-ascent algorithm for nonconvex-concave min-max problems. *arXiv preprint arXiv:2010.15768*, 2020a.

Jingzhao Zhang, Tianxing He, Suvrit Sra, and Ali Jadbabaie. Why gradient clipping accelerates training: A theoretical justification for adaptivity. *arXiv preprint arXiv:1905.11881*, 2019a.

Jingzhao Zhang, Sai Praneeth Karimireddy, Andreas Veit, Seungyeon Kim, Sashank Reddi, Sanjiv Kumar, and Suvrit Sra. Why are adaptive methods good for attention models? *Advances in Neural Information Processing Systems*, 33:15383–15393, 2020b.

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

Kaiqing Zhang, Zhuoran Yang, and Tamer Basar. Policy optimization provably converges to nash equilibria in zero-sum linear quadratic games. In *Advances in Neural Information Processing Systems*, pages 11602–11614, 2019c.

Liang Zhang. Variance reduction for non-convex stochastic optimization: General analysis and new applications. Master's thesis, ETH Zurich, 2021b.

Liang Zhang, Yang Junchi, Amin Karbasi, and Niao He. Optimal guarantees for algorithmic reproducibility and gradient complexity in convex optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023.

Siqi Zhang, Junchi Yang, Cristóbal Guzmán, Negar Kiyavash, and Niao He. The complexity of nonconvex-strongly-concave minimax optimization. *arXiv preprint arXiv:2103.15888*, 2021b.

Yushun Zhang, Congliang Chen, Naichen Shi, Ruoyu Sun, and Zhi-Quan Luo. Adam can converge without any modification on update rules. *Advances in neural information processing systems*, 2022.

Zijun Zhang. Improved adam optimizer for deep neural networks. In *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*, pages 1–2. Ieee, 2018.

Renbo Zhao. A primal dual smoothing framework for max-structured nonconvex optimization. *arXiv preprint arXiv:2003.04375*, 2020.

Shen-Yi Zhao, Yin-Peng Xie, and Wu-Jun Li. On the convergence and improvement of stochastic normalized gradient descent. *Science China Information Sciences*, 64(3):1–13, 2021.

Yaowei Zheng, Richong Zhang, and Yongyi Mao. Regularizing neural networks via adversarial model perturbation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8156–8165, 2021.

Dongruo Zhou, Jinghui Chen, Yuan Cao, Yiqi Tang, Ziyan Yang, and Quanquan Gu. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671*, 2018.

Yi Zhou, Huishuai Zhang, and Yingbin Liang. Geometrical properties and accelerated gradient solvers of non-convex phase retrieval. In *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 331–335. IEEE, 2016.

Fangyu Zou, Li Shen, Zequn Jie, Weizhong Zhang, and Wei Liu. A sufficient condition for convergences of adam and rmsprop. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11127–11135, 2019.

CURRICULUM VITAE

PERSONAL DATA

Name	Junchi Yang
Date of Birth	June 13, 1995
Place of Birth	Fuzhou, China
Citizen of	China

EDUCATION

2021 – 2023	ETH Zurich Zurich Switzerland
2017 – 2021	University of Illinois Urbana-Champaign Champaign, USA *Final degree:* M.S. in Industrial Engineering
2013 – 2017	University of California, Los Angeles Los Angeles, USA *Final degree:* B.S. in Applied Mathematics and B.A. in Economics