Machine Learning: Why Do Simple Algorithms Work So Well?

Chi Jin

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2019-53
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2019/EECS-2019-53.html

May 17, 2019

Machine Learning:
Why Do Simple Algorithms Work So Well?

by

Chi Jin

A dissertation submitted in partial satisfaction of the

requirements for the degree of

Doctor of Philosophy

in

Electrical Engineering and Computer Science

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Michael I. Jordan, Chair
Professor Peter L. Bartlett
Associate Professor Aditya Guntuboyina

Spring 2019

Machine Learning:
Why Do Simple Algorithms Work So Well?

Abstract

Machine Learning:
Why Do Simple Algorithms Work So Well?

by

Chi Jin

Doctor of Philosophy in Electrical Engineering and Computer Science

University of California, Berkeley

Professor Michael I. Jordan, Chair

While state-of-the-art machine learning models are deep, large-scale, sequential and highly nonconvex, the backbone of modern learning algorithms are simple algorithms such as stochastic gradient descent, gradient descent with momentum or Q-learning (in the case of reinforcement learning tasks). A basic question endures—*why do simple algorithms work so well even in these challenging settings?*

To answer above question, this thesis focuses on four concrete and fundamental questions:

1. In nonconvex optimization, can (stochastic) gradient descent or its variants escape saddle points efficiently?

2. Is gradient descent with momentum provably faster than gradient descent in the general nonconvex setting?

3. In nonconvex-nonconcave minmax optimization, what is a proper definition of local optima and is gradient descent ascent game-theoretically meaningful?

4. In reinforcement learning, is Q-learning sample efficient?

This thesis provides the first line of provably positive answers to all above questions. In particular, this thesis will show that although the standard versions of these classical algorithms do not enjoy good theoretical properties in the worst case, simple modifications are sufficient to grant them desirable behaviors, which explain the underlying mechanisms behind their favorable performance in practice.

To Jiaqi, Kelvin and my parents.

Contents

List of Figures

List of Tables

Acknowledgments

I would like to give my foremost thanks to my advisor Michael I. Jordan. Throughout my years at Berkeley, he demonstrated me not only how to do first-class research, but also how to be a respectful academic figure as well as a caring advisor. His pioneering vision over the entire field inspires me to think out of the box, and to attack new challenging problems. His modest viewpoints and encouraging words guide me through the most difficult time in my Ph.D. I am fortunate to be in his group—a highly collaborative environment with diverse interests. It not only gives me a general awareness of the field, but also provides me sufficient freedom to pursue my own interests. I could not have wished for a better advisor.

I am especially grateful to Rong Ge, Praneeth Netrapalli and Sham M. Kakade, who are my wonderful long-term collaborators and have a remarkable influence on my entire Ph.D. path. My journey in nonconvex optimization starts with a long-distance collaboration with Rong, and then a summer internship mentored by Sham at Microsoft Research New England, where both Praneeth and Rong were postdoctoral researchers there. Through collaboration with them, I learned a wide range of ideas, intuitions and powerful technics. This grants me an incredible amount of technical strength for solving challenging problems, which is invaluable for my entire career.

I also want to express my sincere gratitude to Prateek Jain and Sebastian Bubeck, who generously provide me advices and guidance through different stages of my Ph.D. In addition, I would like to thank Zeyuan Allen-Zhu, who contributed significantly to the work in the last part of this thesis. The collaboration experience with him is truly a pleasure. He has demonstrated me how to be a passionate and smart person, but at the same time work tremendously hard.

I would like to thank Peter Bartlett and Adytia Guntuboyina for being on my thesis committee. Peter provided me many helpful advices and feedbacks during my first year. Adytia taught me a lot of useful statistics, and I had a wonderful experience serving as his teaching assistant. I am also grateful to Martin Wainwright and Prasad Raghavendra for being on my Qual Exam committee and providing many helpful discussions.

There are also several peer fellows who have substantial impact on my Ph.D. life. Other than being great collaborators, they are also my personal best friends. Nilesh Tripuraneni taught me a lot about the American tradition, culture and politics, and patiently helped me improve my English writings; Yuchen Zhang provided me various guidance during my first three years; Yuansi Chen and I share many precious memories of teaming up for esports; and Yian Ma convinced me the charm of outdoor activities.

In addition to all people mentioned above, I am fortunate to have many amazing collaborators: Lydia Liu, Darren Lin, Mitchell Stern, Jeff Regier, Nicolas Flammarion, Bin Yu, Simon Du, Jason Lee, Barnabas Poczos, Aarti Singh, Sivaraman Balakrishnan, Aaron Sidford, Cameron Musco, Furong Huang, and Yang Yuan. It was really a great pleasure to work with you all. I must also sincerely thank many other my peer fellows at Berkeley, too many to list here, for making my time at Berkeley some of the best in my life.

I also own credit to Liwei Wang, my undergraduate thesis advisor. As an undergraduate student in physics, it is him who introduced me to machine learning. I am also especially grateful to many members in Liwei's group—Chicheng Zhang, Hongyi Zhang, Ziteng Wang, Kai Fan, Hongyuan You, Songbai Yan, etc. We read through various books and lecture notes together when I had very limited knowledge about machine learning. Without them, I would never have been able to start my graduate study in this area.

Finally, my heartfelt gratitude goes to my family. I am especially grateful to my love and wonderful wife Jiaqi Xiao, for all the sweet time we have been through together, as well as for the all the sacrifice she made for us to live together. We are fortunate to have our lovely son—baby Kelvin Jin. Observing how he learns from the unknown world is a great fun, and often inspires me view learning from different perspectives. I also thank my parents and my parents-in-law for their tremendous support and for taking good care of baby Kelvin.

Chapter 1

Overview

While the empirical performance of large-scale machine learning models has seen rapid progress in recent years, the community also witnessed a growing divergence between what we do in practice and what we understand. On one hand, large-scale, highly nonconvex, deep models are routinely trained using simple algorithms on structured data that is potentially temporally correlated. On the other hand, a basic question remains—*why do simple algorithms work in these challenging settings?* Lacking this understanding makes it hard for practitioners to judge the scope and limits of existing methods. This also makes the design of more powerful algorithms and models difficult. Addressing these issues is vital to sustain the rapid progress of the field.

1.1 Machine Learning and Simple Algorithms

In a high level, modern machine learning tasks can be divided into two categories—*pattern recognition* and *decision making*. In this section, we will overview the major machine learning frameworks for tasks in both categories, and the corresponding simple algorithms to solve them. These algorithms are not only widely used in practice but also frequently reported to achieve state-of-the-art performances.

Pattern Recognition

Pattern Recognition (PR) is the process of automated recognition of underlying patterns by analyzing data using learning algorithms. It dates back to roots in statistics and signal processing in the 1950s and 1960s, and has been the focus of machine learning for the past decades. With the success of deep learning, the field witnessed several major breakthroughs in solving PR tasks especially in image classification (e.g. Krizhevsky, Sutskever, and Hinton, 2012; He et al., 2016), speed recognition (e.g. Hochreiter and Schmidhuber, 1997), etc.

Take image classification as an example. Given a dataset of millions of images and their corresponding labels, the popular modern approach feeds those images into a gigantic deep

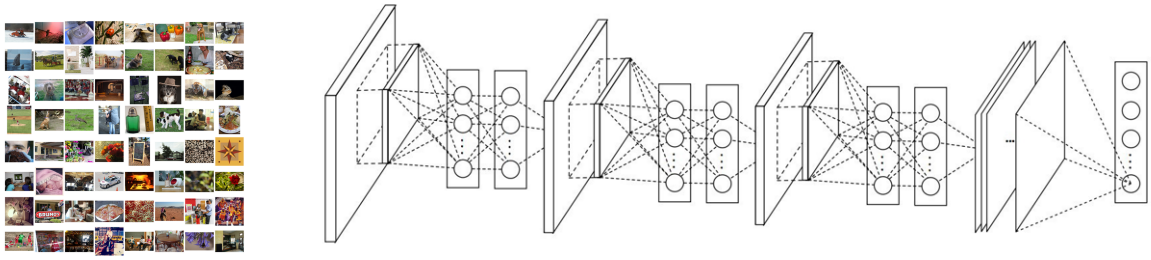

Figure 1.1: Image Classification and Deep Neural Networks

neural network of millions of parameters. Then, we can formulate a loss function which measure the discrepancy between the true labels of the image, and the outputs of the neural network. The process of learning is to find the optimal set of parameters which minimize the loss.

In an abstract level, this is an optimization problem, where we want to find a parameter $\mathbf{x}$, to minimize a target function $f : \mathcal{X} \to \mathbb{R}$:

$$\min_{\mathbf{x}} f(\mathbf{x})$$

Compared to the classical optimization literature, one major difference here is that f need not to be convex, i.e. this is *nonconvex optimization*. In fact, most deep neural network architectures render highly nonconvex objectives. It is NP-hard to find a global optimum of a nonconvex function in general.

One of the most popular algorithms in this setting is Gradient Descent (GD) or its stochastic variant—Stochastic Gradient Descent (SGD). SGD is reported to perform better than several carefully designed adaptive algorithms (Wilson et al., 2017), and to achieve state-of-the-art performance in many applications. One big mystery here is why SGD performs so well in many practical applications despite their objective functions being nonconvex.

Decision Making

Decision making, in addition to possibly identifying the underlying patterns, uses data to make informed decisions that affect the real world. Modern applications in decision making not only involve the standard single-agent one-time decision making, but also involve more sophisticated *multi-agent decision making* and *sequential decision making*.

Multi-agent decision making typically involves multiple agents making decisions to collaborate or compete with each other. Distributed control, multi-agent robotic system and many interdisciplinary applications in economy and machine learning all fall into this category. This thesis will focus on a basic setting in this category—a setting in which two agents compete against each other with a zero-sum reward. This special setting also plays an

important role in several subfields of modern machine learning such as Generative Adversarial Network (GAN) (Goodfellow et al., 2014) and adversarial training (Madry et al., 2017).

A standard formulation for this two-player zero-sum setting is the *minmax optimization* problem where there is a utility function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$. The value of $f(\mathbf{x}, \mathbf{y})$ denotes the gain of the $\mathbf{y}$ player, as well as the loss of the $\mathbf{x}$ player. That is, they are trying to solve the following minmax problem.

$$\min_{\mathbf{x}} \max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$$

Since most models for GAN and adversarial training use neural networks, the function f is typically neither convex in $\mathbf{x}$ nor concave in $\mathbf{y}$, and therefore classical theories for convex-concave functions do not apply here. A basic and popular algorithm in this setting is Gradient Descent Ascent (GDA) which simultaneously performs or alternates between gradient descent on $\mathbf{x}$ and gradient ascent on $\mathbf{y}$. Nonconvex-nonconcave minmax optimization is challenging, and much less understood than nonconvex optimization. Even the basic question of "what GDA is converging to" remains open.

Sequential decision making is a process of making multiple decisions in a sequence. After making each decision, the agent will receive feedback which allows her to better adjust her strategy for later decisions. This type of decision making frequently appears in bidding and advertising, personalized recommendation, games, and robotics.

A common framework to solve these problems is *Reinforcement Learning* (RL), where these problems are usually modeled as a Markov Decision Process (MDP, see Section 5.2). A majority of practical algorithms are variants of two classical algorithms—policy gradients and Q-learning (Watkins, 1989).

One big challenge in RL is the sample efficiency: for difficult tasks, descent machine learning models would already require more than millions of high-quality samples to train. These samples can be either expensive or very time-consuming to collect. A line of recent research tried to design better algorithms that utilize samples more efficiently. However, an even more fundamental question remains: is Q-learning—one of the most classical RL algorithms—sample-efficient?

1.2 Types of Theoretical Guarantees

As machine learning lies in the intersection of computer science and statistics, there are two complexities that are crucial for machine learning algorithms: (1) *iteration complexity* for computational efficiency; (2) *sample complexity* for statistical efficiency.

Iteration complexity or more formally query complexity, is a rigorous way in optimization to measure the computational efficiency of an algorithm. In the case of this paper, we assume there is an gradient oracle such that whenever an algorithm queries a single point

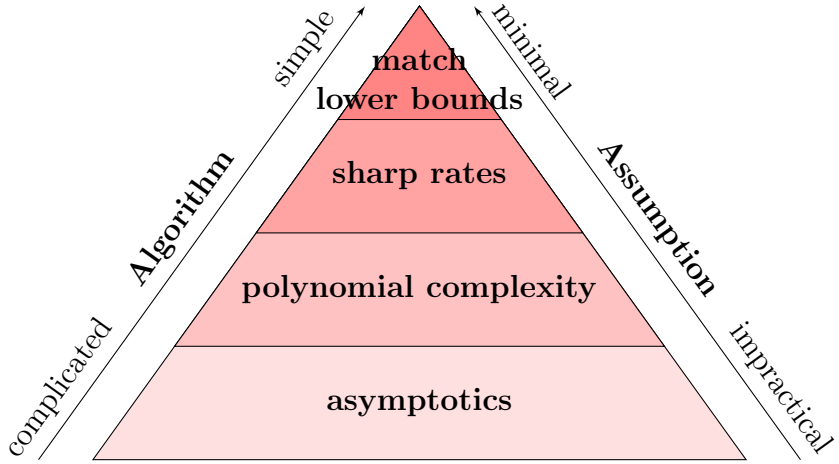

Figure 1.2: Types of theoretical guarantees and their relevance to practice.

$\mathbf{x}$, the oracle will return its gradient $\nabla f(\mathbf{x})$. A standard theoretical guarantee bounds the number of queries needed to find a point of interests.

Sample complexity describes how many samples or data points an algorithm requires to learn well, such as to find a good classifier or a reasonable policy. For a particular algorithm, its sample complexity can be very different from its iteration complexity, as it is possible to collect a small amount of data but perform a computational-intensive subroutine (may even cost exponential time) in analyzing them. However, for many online algorithms such as Q-learning where they collect only one sample per iteration, the sample complexity and the iteration complexity are roughly the same.

Other than the difference in bounding the iteration complexity or the sample complexity, there is also a hierarchy of results based on how strong or how relevant to practice they are (see Figure 1.2).

At the bottom of the pyramid are the asymptotics which are the important first steps in understanding the behavior of the algorithm when the number of samples and runtime go to infinity. However, this type of guarantees falls short of predicting whether the algorithm is useful in practice. For instance, grid search can approximately solve any bounded nonconvex problem in exponential time, and thus owns a desirable asymptotic behavior. However, we know grid search is not a practial algorithm, as we cannot afford the exponential time in practice.

The first step to move beyond the asymptotics is to provide polynomial iteration or sample complexity guarantees. However, in modern machine learning, the ambient dimension of a neural network can be extremely large (on the order of a million). There, quadratic or linear dependence, although both are polynomial, can mean a tremendous difference in practice. This makes a quest for a theory to provide not only polynomial guarantees but also sharp dependence on problem parameters. Finally, at the very top of the pyramid are

the guarantees which match the fundamental limits of the problem that no algorithms can surpass.

Last but not least, when theoretical performances are similar, practitioners usually favor simple and general purposed algorithms over those complicated algorithms that are heavily modified for theoretical proofs. A theory with minimal assumptions would also be more relevant than a theory that makes many impractical assumptions.

This thesis aims to provide the theoretical guarantees which give sharp rates with minimal assumptions for simple algorithms that are widely used in practice.

1.3 Organization

This thesis is centered around four concrete questions in answering the general basic question—in modern machine learning, why do simple algorithms work so well?

We start with nonconvex optimization in Part I, and ask whether (stochastic) gradient descent or its variants can escape saddle points efficiently. Chapter 2 provided the first, sharp (i.e., *almost dimension-free*) guarantee on how fast (stochastic) gradient descent escapes from saddle points; showing saddle points are of little practical concern even in large-scale models. This chapter is based on joint work with Rong Ge, Praneeth Netrapalli, Sham M. Kakade, and Michael I. Jordan (Jin et al., 2017; Jin et al., 2019b). The thesis then proceeds to the next question "Is gradient descent with momentum provably faster than gradient descent in the general nonconvex setting?" Chapter 3 rigorously explained the advantages of adding momentum in nonconvex scenarios. This chapter is based on the joint work with Praneeth Netrapalli and Michael I. Jordan (Jin, Netrapalli, and Jordan, 2017).

Part II studies minmax optimization problem in the nonconvex-nonconcave setting. Unlike nonconvex optimization, very basic questions remain open for nonconvex-nonconcave minmax optimization, including what the proper notion of local optimality is and what the game-theoretical meaning of gradient descent ascent is. Chapter 4 defines a new notion of local optimality, and provides the first full characterization of the stable limit points of gradient descent ascent using this new notion. This chapter is based on a joint work with Praneeth Netrapalli and Michael I. Jordan (Jin, Netrapalli, and Jordan, 2019).

Finally, the central topic of Part III is reinforcement learning. One fundamental question there is whether the basic algorithm Q-learning is sample efficient. It remained unsolved even in the basic scenario with finitely many states and actions. Chapter 5 provides the first positive answer in this scenario. We showed that when paired with properly designed exploration strategies, Q-learning is sample efficient. Our analysis is the first to establish a *near-optimal* regret in the model-free setting. This is based on the joint work with Zeyuan Allen-Zhu, Sebastian Bubeck, and Michael I. Jordan (Jin et al., 2018).

Part I

Nonconvex Optimization

Chapter 2

Escaping Saddle Points by Gradient Descent

Gradient descent (GD) and stochastic gradient descent (SGD) are the most popular workhorses for solving nonconvex optimization problems arising in several fields, most notably in large scale machine learning. Traditional analyses of GD and SGD in this setting show that both algorithms converge to stationary points efficiently. Unfortunately however, they do not rule out convergence to saddle points. On the other hand, for several important machine learning problems, recent works have shown the importance of converging to local minima rather than saddle points. The main contribution of this work is to show that, perturbed versions of GD and SGD escape saddle points and converge to second-order local minima in essentially the same time they take to converge to stationary points, with only extra logarithmic factors.

2.1 Introduction

Nonconvex optimization problems are ubiquitous in several fields of engineering such as control theory (Bertsekas, 1995), signal processing (Oppenheim and Schafer, 1989), machine learning (Bishop, 2006), etc. Gradient descent (GD) (Cauchy, 1847) and its variants such as stochastic gradient descent (SGD) (Robbins and Monro, 1951) are some of the most widely used algorithms for solving these problems in practice. There are two key reasons for the wide usage of GD and SGD in solving these nonconvex problems—(a) *each step* of GD and SGD can usually be implemented in time *linear* in the dimension (thus suitable for solving high dimensional problems) and (b) in many applications, they are observed to converge to *good* solutions in a *few* steps. Contrast this with the fact that solving general nonconvex problems is NP-hard in the worst case, it leaves a basic question—how does GD manage to converge to good solutions efficiently.

Traditional analyses of GD only show its efficient convergence to first-order stationary points (i.e., points where the gradient $\nabla f(\mathbf{x}) = 0$) in general (Nesterov, 1998). First-order stationary points can be local minima, local maxima or even saddle points, where

an enormous number of them are highly suboptimal. However, a recent series of works, some theoretical and some empirical, have uncovered a nice structure in several problems of practical interest that sheds light on this surprising behavior of GD. These works show that even though these nonconvex problems have a large number of *bad* saddle points, all local minima are *good*. More precisely, they show that, for a large class of interesting nonconvex problems, second-order stationarity (i.e., $\nabla f(\mathbf{x}) = 0$ and $\nabla^2 f(\mathbf{x}) \succeq 0$)—a weaker notion of local optimality which only excludes saddle points with strictly negative curvatures—already guarantees (approximate) global optimality: Choromanska et al. (2014) presents such a result for learning multi-layer neural networks, Bandeira, Boumal, and Voroninski (2016) and Mei et al. (2017) for synchronization and MaxCut, Boumal, Voroninski, and Bandeira (2016) for smooth semidefinite programs, Bhojanapalli, Neyshabur, and Srebro (2016) for matrix sensing, Ge, Lee, and Ma (2016) for matrix completion, and Ge, Jin, and Zheng (2017) for robust PCA. This motivates the quest to find second-order stationary points, as a natural second-order surrogates for local minima.

Recent work (Lee et al., 2016) shows that GD, under random initialization or with perturbations, converges to second-order stationary points with probability one. (Ge et al., 2015) further makes this result quantitative by bounding the number of iterations taken by a perturbed version of GD for finding an ϵ-second-order stationary point ($\|\nabla f(\mathbf{x})\| \le \epsilon$ and $\nabla^2 f(\mathbf{x}) \succeq -\sqrt{\epsilon}\mathbf{I}$) by $\mathrm{poly}(d, \epsilon^{-1})$. While these convergence results are inspiring, the number of steps required is still significantly larger than the number of steps for GD to find first-order stationary points, which is $\mathcal{O}(\epsilon^{-2})$ *independent of dimension* d. The additional polynomial dependence on d is particularly undersirable for high dimensional applications for which GD methods are most interesting and useful. This leads to the following question on efficiency:

Can GD and SGD escape saddle points and find second-order stationary point efficiently?

More precisely, we are interested if this efficiency can be competitive to the efficiency of GD and SGD in finding a first-order stationary point, which only takes a dimension-free number of iterations.

This work provides the first provable positive answer to the above question. It shows that, rather surprisingly, with small perturbations, GD and SGD escape saddle points and find second-order stationary points in essentially the same time they take to find first-order stationary points. More concretely, we show that the overheads are only logarithmic factors in the first two cases, and a linear factor in d in the third case:

- Perturbed gradient descent (PGD) finds ϵ-second-order stationary point in $\tilde{\mathcal{O}}(\epsilon^{-2})$ iterations, where $\tilde{\mathcal{O}}(\cdot)$ hides only absolute constants and ploylogarithmic factors. Compared to the $\mathcal{O}(\epsilon^{-2})$ iterations required by GD in finding first-order stationary points (Nesterov, 1998), this involves only additional *polylogarithmic* factors in d.

- In the stochastic setting where stochastic gradients are Lipschitz, perturbed stochastic gradient descent (PSGD) finds ϵ-second-order stationary points in $\tilde{\mathcal{O}}(\epsilon^{-4})$ iterations.

Setting	Algorithm	Iterations	Guarantees
Non-stochastic	GD (Nesterov, 2000)	$\mathcal{O}(\epsilon^{-2})$	first-order stationary point
	PGD	$\tilde{\mathcal{O}}(\epsilon^{-2})$	second-order stationary point
Stochastic	SGD (Ghadimi and Lan, 2013)	$\mathcal{O}(\epsilon^{-4})$	first-order stationary point
	PSGD (*with* Assumption C)	$\tilde{\mathcal{O}}(\epsilon^{-4})$	second-order stationary point
	PSGD (*no* Assumption C)	$\tilde{\mathcal{O}}(d\epsilon^{-4})$	second-order stationary point

Table 2.1: A high level summary of the results of this work and their comparison to prior state of the art for GD and SGD algorithms. This table only highlights the dependences on d and ϵ.

Compared to the $\mathcal{O}(\epsilon^{-4})$ iterations required by SGD in finding first-order stationary points (Ghadimi and Lan, 2013), this again incurs overhead that is only *polylogarithmic* in d.

- When stochastic gradients are not Lipschitz, PSGD finds ϵ-second-order stationary point in $\tilde{\mathcal{O}}(d\epsilon^{-4})$ iterations – this involves only an additional *linear* factor in d.

Related Work

In this section we review the related works which provide convergence guarantees to find second-order stationary points. See also Appendix 2.8 for tables of comparison of our work to existing works.

Non-stochastic settings. Classical algorithms for finding second-order stationary points require access to exact Hessian information, and are thus second-order algorithms. Some of the most well known algorithms here are cubic regularization method (Nesterov and Polyak, 2006) and trust region methods (Curtis, Robinson, and Samadi, 2014), both of which require $\mathcal{O}(\epsilon^{-1.5})$ gradient and Hessian queries. However, owing to the size of Hessian matrices which scales quadratically with respect to dimension, these methods are computational intensive per iteration especially for high dimensional problems. This has motivated researchers to focus on first order methods, which only utilize gradient information and therefore are much cheaper per iteration.

Among first-order algorithms, Carmon et al. (2016) and Agarwal et al. (2017) design double-loop algorithms which require Hessian-vector product oracles, and obtain convergence rates of $\tilde{\mathcal{O}}(\epsilon^{-1.75})$ gradient queries. This line of algorithms is carefully designed for analysis

purposes. They are relatively diffcult to implement, thus less appealing in practice. On the other hand, simple single-loop algorithms, such as gradient descent are still challenging to understand and analyze as we can no longer artificially change the algorithm to control its behavior. Ge et al. (2015) and Levy (2016) studied simple variants of gradient descent, but require poly(d) gradient queries to find second-order stationary points. This work is the first to show that a simple perturbed version of GD escapes saddle points and finds second-order stationary points in $\tilde{\mathcal{O}}(\epsilon^{-2})$ gradient queries, only paying overhead of logarithmic factors compared to the rate of finding first-order stationary point. As a followup work, Jin, Netrapalli, and Jordan (2017) show that a perturbed version of celebrated Nesterov's accelerated gradient descent (Nesterov, 1983) enjoys a faster convergence rate of $\tilde{\mathcal{O}}(\epsilon^{-1.75})$.

Stochastic setting with Lipschitz stochastic gradient. In this setting, the algorithm only has access to stochastic gradients. Most existing works assume that the *stochastic gradients themselves are Lipschitz* (or equivalently that the stochastic functions are gradient-Lipschitz, see Assumption C). Under this assumption, and an additional *Hessian-vector product* oracle, (Allen-Zhu, 2018; Zhou, Xu, and Gu, 2018; Tripuraneni et al., 2018) designed algorithms that have an iteration complexity of $\tilde{\mathcal{O}}(\epsilon^{-3.5})$. (Xu, Rong, and Yang, 2018; Allen-Zhu and Li, 2017) obtain similar results without the requirement for Hessian-vector product oracle. The sharpest rates in this category are by (Fang et al., 2018; Zhou and Gu, 2019), which show that the iteration complexity can be further reduced to $\tilde{\mathcal{O}}(\epsilon^{-3})$. Again, this line of works consists of double-loop algorithms, and are relatively diffcult to implement in practice.

Among single-loop algorithms that are simple variants of SGD, (Ge et al., 2015) provides the first polynomial result showing noisy gradient descent finds second-order stationary points in $d^4\text{poly}(\epsilon^{-1})$ iterations. (Daneshmand et al., 2018) designs a new algorithm CNC-SGD and shows that assuming the variance of stochastic gradient along the escaping direction of saddle points is at least γ for all saddle points, then CNC-SGD finds SOSPs in $\tilde{\mathcal{O}}(\gamma^{-4}\epsilon^{-5})$ iterations. We note that in general, γ scales as $1/d$, which gives complexity $\tilde{\mathcal{O}}(d^4\epsilon^{-5})$. Our work is the first result showing that a simple perturbed version of SGD achieves the convergence rate of $\tilde{\mathcal{O}}(\epsilon^{-4})$, which matches the speed of SGD to find a first-order stationary point up to polylogarithmic factors in dimension. Concurrent to this work, (Fang, Lin, and Zhang, 2019) analyzes SGD with averaging over last few iterates, and obtains a faster convergence rate $\tilde{\mathcal{O}}(\epsilon^{-3.5})$.

Stochastic setting (general). Significantly less amount of prior works provide results in the general setting where stochastic gradients are no longer guaranteed to be Lipschitz. In fact, only the results of Ge et al. (2015) and Daneshmand et al. (2018) apply here, and both of them require at least $\Omega(d^4)$ gradient queries to find second-order stationary points. Our work is the first result in this setting achieving linear dimension dependence.

Other settings. Finally, there are also several recent results in the setting where objective function can be written as a finite sum of individual functions, we refer readers to Reddi et al. (2017) and Allen-Zhu and Li (2017) and the references therein for further reading.

Chapter Organization

In Section 2.2, we review the preliminaries. In Section 2.3, we discuss the landscape of a wide class of nonconvex problems in machine learning, demonstrating how second-order stationarity already ensures approximate global optimality via a simple example. In Section 2.4, we state the algorithms and present our main results for perturbed GD and SGD. In Section 2.6, we present our proof for non-stochastic case (perturbed GD), which illustrates some of our key ideas. The proof for stochastic setting is presented in the appendix. We conclude in Section 2.5, with discussions on several related topics.

2.2 Preliminaries

In this section, we will first introduce our notation, and then present definitions, assumptions and existing results in nonconvex optimization, in both deterministic and stochastic settings.

Notation

We use bold upper-case letters $\mathbf{A}, \mathbf{B}$ to denote matrices and bold lower-case letters $\mathbf{x}, \mathbf{y}$ to denote vectors. For vectors we use $\|\cdot\|$ to denote the ℓ_2-norm, and for matrices we use $\|\cdot\|$ and $\|\cdot\|_{\mathrm{F}}$ to denote spectral (or operator) norm and Frobenius norm respectively. We use $\lambda_{\min}(\cdot)$ to denote the smallest eigenvalue of a matrix. For a function $f : \mathbb{R}^d \to \mathbb{R}$, we use ∇f and $\nabla^2 f$ to denote its gradient and Hessian, and $f^\star$ to denote the global minimum of function f. We use notation $\mathcal{O}(\cdot), \Theta(\cdot), \Omega(\cdot)$ to hide only absolute constants which do not depend on any problem parameter, and notation $\tilde{\mathcal{O}}(\cdot), \tilde{\Theta}(\cdot), \tilde{\Omega}(\cdot)$ to hide only absolute constants and factors that are ploy-logarithmically dependent on all problem parameters.

Nonconvex Optimization and Gradient Descent

In this work, we are interested in solving general unconstrained optimization problems

$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}),$$

where f is a smooth function which can be nonconvex. More concretely, we assume that f has Lipschitz gradients and Lipschitz Hessians, which ensures both gradient and Hessian can not change too rapidly.

Definition 2.2.1. A differentiable function f is **ℓ-gradient Lipschitz** (or ℓ-smooth) if:

$$\|\nabla f(\mathbf{x}_1) - \nabla f(\mathbf{x}_2)\| \le \ell \|\mathbf{x}_1 - \mathbf{x}_2\| \quad \forall\, \mathbf{x}_1, \mathbf{x}_2.$$

Definition 2.2.2. A twice-differentiable function f is **ρ-Hessian Lipschitz** if:

$$\|\nabla^2 f(\mathbf{x}_1) - \nabla^2 f(\mathbf{x}_2)\| \leq \rho \|\mathbf{x}_1 - \mathbf{x}_2\| \quad \forall\, \mathbf{x}_1, \mathbf{x}_2.$$

Assumption A. Function f is ℓ-gradient Lipschitz and ρ-Hessian Lipschitz.

One of the most classical algorithms in optimization is Gradient Descent (GD), whose update takes following form with learning rate η:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \nabla f(\mathbf{x}_t) \tag{2.1}$$

Since finding a global optimum for general nonconvex function is NP-hard, most classical results turn to analyze convergence to a local surrogate—first-order stationary points.

Definition 2.2.3. For differentiable function f, $\mathbf{x}$ is a **(first-order) stationary point** if $\nabla f(\mathbf{x}) = \mathbf{0}$.

Definition 2.2.4. For differentiable function f, $\mathbf{x}$ is a **ϵ-(first-order) stationary point** if $\|\nabla f(\mathbf{x})\| \leq \epsilon$.

It is well known that gradient descent converges to first-order stationary points in a number of iterations that is independent of dimension; this is referred to as "dimension-free optimization" in literature.

Theorem 2.2.5 ((Nesterov, 1998))**.** *For any $\epsilon > 0$, assume function $f(\cdot)$ is ℓ-gradient Lipschitz, and let learning rate $\eta = 1/\ell$. Then, gradient descent Eq.(2.1) will visit ϵ-stationary point at least once in the following number of iterations:*

$$\frac{\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2}$$

Note that in the above results, the last iterate is not guaranteed to be a stationary point. However, it is not hard to figure out which iterate is the stationary point by calculating the norm of gradient at every iteration.

A first-order stationary point can be a local minimum, a local maximum or even a saddle point:

Definition 2.2.6. For differentiable function f, a stationary point $\mathbf{x}$ is a

- **local minimum**, if there exists $\delta > 0$ so that $f(\mathbf{x}) \leq f(\mathbf{y})$ for any $\mathbf{y}$ with $\|\mathbf{y} - \mathbf{x}\| \leq \delta$.
- **local maximum**, if there exists $\delta > 0$ so that $f(\mathbf{x}) \geq f(\mathbf{y})$ for any $\mathbf{y}$ with $\|\mathbf{y} - \mathbf{x}\| \leq \delta$.
- **saddle point**, otherwise.

For minimization problems, both saddle points and local maxima are clearly undesirable, and we abuse nomenclature to call both of them "saddle points" in this work. Unfortunately, distinguishing saddle points versus local minima for smooth functions is still NP-hard in general (Nesterov, 2000). To avoid these hardness results, this work focuses on escaping a subclass of saddle points.

Definition 2.2.7. For twice-differentiable function f, $\mathbf{x}$ is a **strict saddle point** if $\mathbf{x}$ is a stationary point and $\lambda_{\min}(\nabla^2 f(\mathbf{x})) < 0$.

A generic saddle point must satisfy that $\lambda_{\min}(\nabla^2 f(\mathbf{x})) \leq 0$. Being "strict" simply rules out the case where $\lambda_{\min}(\nabla^2 f(\mathbf{x})) = 0$. Equivalently, this defines a more suitable goal: to find those stationary points that are not strict saddle points.

Definition 2.2.8. For twice-differentiable function $f(\cdot)$, $\mathbf{x}$ is a **second-order stationary point** if:

$$\nabla f(\mathbf{x}) = \mathbf{0}, \quad \text{and} \quad \nabla^2 f(\mathbf{x}) \succeq \mathbf{0}.$$

Definition 2.2.9. For a ρ-Hessian Lipschitz function $f(\cdot)$, $\mathbf{x}$ is an **ϵ-second-order stationary point** if:

$$\|\nabla f(\mathbf{x})\| \leq \epsilon \quad \text{and} \quad \nabla^2 f(\mathbf{x}) \succeq -\sqrt{\rho\epsilon} \cdot \mathbf{I}.$$

Definition 2.2.9 is a ϵ-robust version of Definition 2.2.4. Definition 2.2.9 uses the Hessian Lipschitz parameter ρ to help match the units of gradient and Hessian, following the convention of Nesterov and Polyak (2006).

Although second-order stationarity is only a necessary condition for being a local minimum, a line of recent analyses shows that for many popular applications in machine learning, all ϵ-second-order stationary points are approximate global minima, thus finding second-order stationary points is sufficient for solving those problems. See Section 2.3 for more details.

Stochastic Approximation

We also consider the stochastic approximation setting, where we may not access exact $\nabla f(\cdot)$ directly. Instead for any point $\mathbf{x}$, a gradient query will return a stochastic gradient $\mathbf{g}(\mathbf{x}; \theta)$, where θ is a random variable drawn from a distribution $\mathcal{D}$. The key property satisfied by stochastic gradients $\mathbf{g}(\cdot; \cdot)$ is that $\nabla f(\mathbf{x}) = \mathbb{E}_{\theta \sim \mathcal{D}}[\mathbf{g}(\mathbf{x}; \theta)]$, i.e. the expection of stochastic gradient equals true gradient. In short, the update of Stochastic Gradient Descent (SGD) is:

$$\text{Sample } \ \theta_t \sim \mathcal{D}, \qquad \mathbf{x}_{t+1} = \mathbf{x}_t - \eta \nabla \mathbf{g}(\mathbf{x}_t; \theta_t) \tag{2.2}$$

Other than being an unbiased estimator of true gradient, another standard assumption on the stochastic gradients is that their variance is bounded by some number σ^2, i.e.

$$\mathbb{E}_{\theta \sim \mathcal{D}}\left[\|\mathbf{g}(\mathbf{x}, \theta) - \nabla f(\mathbf{x})\|^2\right] \leq \sigma^2$$

When we are interested in high probability bounds, one often makes the stronger assumption on tail distribution of stochasticity.

Assumption B. For any $\mathbf{x} \in \mathbb{R}^d$, stochastic gradient $\mathbf{g}(\mathbf{x};\theta)$ with $\theta \sim \mathcal{D}$ satisfies:

$$\mathbb{E}\mathbf{g}(\mathbf{x};\theta) = \nabla f(\mathbf{x}), \qquad \mathbb{P}\left(\|\mathbf{g}(\mathbf{x};\theta) - \nabla f(\mathbf{x})\| \geq t\right) \leq 2\exp(-t^2/(2\sigma^2)), \qquad \forall t \in \mathbb{R}$$

We note this assumption is more general than the standard notion of sub-Gassuain random vector which assumes $\mathbb{E}\exp(\langle \mathbf{v}, \mathbf{X} - \mathbb{E}\mathbf{X}\rangle) \leq \exp(\sigma^2\|\mathbf{v}\|^2/d)$ for any $\mathbf{v} \in \mathbb{R}^d$. The latter one requires distribution to be "isotropic" while our assumption does not. By Lemma 2.9.2 we know that both bounded random vector, and standard sub-Gaussian random vector are special cases of our assumption.

Again, prior works show that stochastic gradient descent also converges to first-order stationary points in a number of iterations that are independent of dimension.

Theorem 2.2.10 ((Ghadimi and Lan, 2013))**.** *For any $\epsilon, \delta > 0$, assume function f is ℓ-gradient Lipschitz, stochastic gradient $\mathbf{g}$ satisfies Assumption B, and let learning rate $\eta = \tilde{\Theta}(\ell^{-1}(1+\sigma^2/\epsilon^2)^{-1})$. Then, with probability at least $1-\delta$, stochastic gradient descent Eq.(2.2) will visit ϵ-stationary point at least once in the following number of iterations:*

$$\tilde{\mathcal{O}}\left(\frac{\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2}\left(1 + \frac{\sigma^2}{\epsilon^2}\right)\right)$$

2.3 Common Landscape of Nonconvex Applications in Machine Learning

In this section, we illustrate the importance of second-order stationary points—for a wide class of nonconvex applications in machine learning and signal processing, all second-order stationary points are global minima.

These applications include tensor decomposition (Ge et al., 2015), dictionary learning (Sun, Qu, and Wright, 2016b), phase retrieval (Sun, Qu, and Wright, 2016a), synchronization and MaxCut (Bandeira, Boumal, and Voroninski, 2016; Mei et al., 2017), smooth semidefinite programs (Boumal, Voroninski, and Bandeira, 2016), and many problems related to low-rank matrix factorization, such as matrix sensing (Bhojanapalli, Neyshabur, and Srebro, 2016), matrix completion (Ge, Lee, and Ma, 2016) and robust PCA (Ge, Jin, and Zheng, 2017).

The above works show that, by adding appropriate regularization terms, and under mild conditions, there are two geometric properties satisfied by the corresponding objective functions (a) all local minima are global minima. There might be multiple local minima due to permutation, but they are all equally good; (b) all saddle points have at least one direction with strictly negative curvature, thus are strict saddle points. Finally, we observe:

Fact 2.3.1. *If a function f satisfies (a) all local minima are global minima; (b) all saddle points (including local maxima) are strict saddle points, then all second-order stationary points are global minima.*

This implies that the core problem for these nonconvex applications is to find second-order stationary points efficiently. If we can prove that some simple variants of GD and SGD converges to second-order stationary points efficiently, then we immediately establish global convergence results for all the above applications (i.e. convergence from arbitrary initialization), and in fact do so efficiently.

In the rest of this section, we illustrate the above common geometric properties via a simple example of finding top eigenvector. Given a positive semidefinite matrix $\mathbf{M} \in \mathbb{R}^{d\times d}$, consider the following objective:

$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\mathbf{x}^\top - \mathbf{M}\|_{\mathrm{F}}^2, \tag{2.3}$$

Denote the eigenvalues and eigenvectors of $\mathbf{M}$ as $(\lambda_i, \mathbf{v}_i)$ for $i = 1, \ldots, d$, and assume there is a gap between the first and second eigenvalues, i.e. $\lambda_1 > \lambda_2 \geq \lambda_3 \geq \ldots \geq \lambda_d \geq 0$. In this case, the global optimal solutions are $\mathbf{x} = \pm\sqrt{\lambda_1}\mathbf{v}_1$ giving the top eigenvector direction.

However, the objective function (2.3) is nonconvex in $\mathbf{x}$. In order to directly optimize over objective via gradient methods, we need to analyze the global landscape of objective function (2.3). Its gradient and Hessian are of the form:

$$\begin{aligned}\nabla f(\mathbf{x}) =& (\mathbf{x}\mathbf{x}^\top - \mathbf{M})\mathbf{x}\\ \nabla^2 f(\mathbf{x}) =& \|\mathbf{x}\|^2\mathbf{I} + 2\mathbf{x}\mathbf{x}^\top - \mathbf{M}\end{aligned}$$

Therefore, all the stationary points satisfy the equation $\mathbf{M}\mathbf{x} = \|\mathbf{x}\|^2\mathbf{x}$. That is, they are $\mathbf{0}$ and $\pm\sqrt{\lambda_i}\mathbf{v}_i$ for $i = 1, \ldots, d$. We already know $\pm\sqrt{\lambda_1}\mathbf{v}_1$ are global minima, thus are also local minima, they are equivalent up to a sign difference. For all remaining stationary points $\mathbf{x}^\dagger$, we note that their Hessian always has strict negative curvature along $\mathbf{v}_1$ direction, i.e. $\mathbf{v}_1^\top\nabla^2 f(\mathbf{x}^\dagger)\mathbf{v}_1 \leq \lambda_2 - \lambda_1 < 0$. Thus they are strict saddle points. So far, we have proved all the preconditions of Fact 2.3.1, which enables us to conclude:

Proposition 2.3.2. *Assume* $\mathbf{M}$ *is a positive semidefinite matrix with top two eigenvalues* $\lambda_1 > \lambda_2 \geq 0$*, then for the objective Eq.(2.3) of finding the top eigenvector, all second-order stationary points are global optima.*

Further analysis can be done to establish the ϵ-robust version of the Proposition 2.3.2. Informally, it can be shown that under technical conditions, for polynomially small ϵ, all ϵ-second-order stationary point are close to global optima. We refer the readers to (Ge, Jin, and Zheng, 2017) for the formal statement.

To summarize the discussion, in order to solve a wide class of nonconvex problems, it suffices to establish algorithmic results that find ϵ-second-order stationary points efficiently.

2.4 Main Results

In this section, we present our main results on the efficiency of simple variants of GD and SGD to escape saddle points and find second-order stationary points. We first study the

Algorithm 1 Perturbed Gradient Descent (PGD)

Input: $\mathbf{x}_0$, learning rate η, perturbation radius r.
for $t = 0, 1, \ldots,$ **do**
$\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta(\nabla f(\mathbf{x}_t) + \xi_t), \qquad \xi_t \sim \mathcal{N}(\mathbf{0}, (r^2/d)\mathbf{I})$

case where the exact gradients are accessible, and present the result for Perturbed GD. In Section 2.4, we study the stochastic setting, and present the results for Perturbed SGD and its mini-batch version.

When exact gradients are available, GD is the simplest algorithm to run in this setting. However, according to its update rule Eq.(2.1), GD only moves its iterates when gradient is non-zero. That is, GD will natually get stuck at saddle points if initialized there. A simple fix to this problem is to inject certain randomness to the iterates. Therefore, this work considers a perturbed version of gradient descent (Algorithm 1).

At each iteration, Algorithm 1 is almost the same as gradient descent, except it adds a small isotropic random Gaussian perturbation to the gradient. The perturbation ξ_t is sampled from a zero-mean Gaussian with covariance $(r^2/d)\mathbf{I}$ so that $\mathbb{E}\|\xi_t\|^2 = r^2$. We note that Algorithm 1 simplifies the preliminary version in (Jin et al., 2017) which adds perturbation more carefully only when certain conditions hold.

We are now ready to present our main result, which claims that if we pick $r = \tilde{\Theta}(\epsilon)$ in Algorithm 1, PGD will find ϵ-second-order stationary point in a number of iterations that is only polylogarithmic in dimension.

Theorem 2.4.1. *For any $\epsilon, \delta > 0$, assume function $f(\cdot)$ satisfies Assumption A, and we run PGD (Algorithm 1) with parameter $\eta = \tilde{\Theta}(1/\ell), r = \tilde{\Theta}(\epsilon)$. Then, with probability at least $1 - \delta$, PGD will visit $\epsilon-$second-order stationary point at least once in the following number of iterations:*

$$\tilde{\mathcal{O}}\left(\frac{\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2}\right)$$

where $\tilde{\mathcal{O}}, \tilde{\Theta}$ hides poly-logarithmic factors in $d, \ell, \rho, 1/\epsilon, 1/\delta$ and $\Delta_f := f(\mathbf{x}_0) - f^\star$.

Remark 2.4.2 (Output a second-order stationary point)**.** *In order to output an ϵ-second-order stationary point, it can be shown by minorly adjusting the proof that, if we run PGD for double that number of iterations in Theorem 2.4.1, one half of the iterates will be ϵ-second-order stationary points. Then, if we output one iterate uniformly at random, with at least a constant probability, it will be an ϵ-second-order stationary point.*

Remark 2.4.3 (Alternative distributions of perturbations)**.** *We note that the distribution of perturbations is not necessarily Gaussian as in Algorithm 1. The key properties needed for the perturbation distributions are (a) light tail distribution for concentration, (b) at least a small amount of variance in every direction.*

Algorithm 2 Perturbed Stochastic Gradient Descent (PSGD)

Input: $\mathbf{x}_0$, learning rate η, perturbation radius r.
for $t = 0, 1, \ldots,$ **do**
sample $\theta_t \sim \mathcal{D}$
$\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta(\mathbf{g}(\mathbf{x}_t; \theta_t) + \xi_t), \qquad \xi_t \sim \mathcal{N}(\mathbf{0}, (r^2/d)\mathbf{I})$

Comparing Theorem 2.4.1 to classical result Theorem 2.2.5, our result shows that rather surprisingly, perturbed gradient descent finds second-order stationary points in almost the same time as gradient descent finds first-order stationary points, up to only logarithmic factors. Therefore, escaping strict saddle points is a very easy task even in terms of efficiency.

We also note that comparing to Theorem 2.2.5, Theorem 2.4.1 also makes an additional assumption on Hessian Lipschitz, which is essential in separating strict saddle points from second-order stationary points.

Stochastic Setting

Recall in the stochastic approximation setting, exact gradients $\nabla f(\cdot)$ are no longer available, and the algorithms are given stochastic gradients $\mathbf{g}(\cdot; \theta)$ such that $\nabla f(\mathbf{x}) = \mathbb{E}_{\theta \sim \mathcal{D}}[\mathbf{g}(\mathbf{x}; \theta)]$.

In many applications in machine learning, the stochastic gradient $\mathbf{g}$ is often realized as gradient of a stochastic function $\mathbf{g}(\cdot; \theta) = \nabla f(\cdot; \theta)$ where the stochastic function itself can have good smoothness property. That is, the stochastic gradient can be Lipschitz.

Assumption C. For any $\theta \in \mathrm{supp}(\mathcal{D})$, $\mathbf{g}(\cdot; \theta)$ is $\tilde{\ell}$-Lipschitz, i.e.

$$\|\mathbf{g}(\mathbf{x}_1; \theta) - \mathbf{g}(\mathbf{x}_2; \theta)\| \le \tilde{\ell}\|\mathbf{x}_1 - \mathbf{x}_2\| \quad \forall\, \mathbf{x}_1, \mathbf{x}_2.$$

Most prior works heavily rely on this assumption. Intuitively, in the special case of $\mathbf{g}(\cdot; \theta) = \nabla f(\cdot; \theta)$ for some twice-differentiable stochastic function $f(\cdot; \theta)$, Assumption C ensures the spectral norm of Hessian of stochastic function $f(\cdot; \theta)$ to be bounded by $\tilde{\ell}$ for all θ. Therefore, the stochastic Hessian also enjoys good concentration properties, which helps algorithms to find points with second-order characterization. In contrast, when Assumption C no longer holds, the problem of finding second-order stationary points becomes much more challenging without the concentration of stochastic Hessian. For the sake of clean presentation, this work treats the general case where Assumption C does not hold by taking $\tilde{\ell} = +\infty$.

We are now ready to present our main result which guarantees the efficiency of PSGD (Algorithm 2) in finding a second-order stationary point. The parameter choice of Algorithm 2 is given by:

$$\eta = \tilde{\Theta}(\frac{1}{\ell \cdot \mathfrak{N}}), \quad r = \tilde{\Theta}(\epsilon\sqrt{\mathfrak{N}}), \quad \text{where} \quad \mathfrak{N} = 1 + \min\left\{\frac{\sigma^2}{\epsilon^2} + \frac{\tilde{\ell}^2}{\ell\sqrt{\rho\epsilon}},\ \frac{\sigma^2 d}{\epsilon^2}\right\} \tag{2.4}$$

Algorithm 3 Mini-batch Perturbed Stochastic Gradient Descent (Mini-batch PSGD)

Input: $\mathbf{x}_0$, learning rate η, perturbation radius r.
for $t = 0, 1, \ldots,$ **do**
sample $\{\theta_t^{(1)}, \cdots \theta_t^{(m)}\} \sim \mathcal{D}$
$\mathbf{g}_t(\mathbf{x}_t) \leftarrow \sum_{i=1}^m \mathbf{g}(\mathbf{x}_t; \theta_t^{(i)})/m$
$\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta(\mathbf{g}_t(\mathbf{x}_t) + \xi_t), \qquad \xi_t \sim \mathcal{N}(\mathbf{0}, (r^2/d)\mathbf{I})$

Theorem 2.4.4. *For any $\epsilon, \delta > 0$, if function f satisfies Assumption A, and stochastic gradient $\mathbf{g}$ satisfies Assumption B (and C optionally), and we run PSGD (Algorithm 2) with parameter (η, r) chosen as Eq.(2.4). Then, with probability at least $1 - \delta$, PSGD will visit $\epsilon-$second-order stationary point at least once in the following number of iterations:*

$$\tilde{O}\left(\frac{\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2} \cdot \mathfrak{N}\right)$$

We note Remark 2.4.2 on how to output a second-order stationary point and Remark 2.4.3 on alterative distribution of perturbations can also be directly applied to Theorem 2.4.4.

Theorem 2.4.4 summarize the results for both scenarios with or without Assumption C. In case where stochastic gradients are Lipschitz (i.e. Assumption C is valid), for sufficiently small ϵ where $\sigma^2/\epsilon^2 \geq \tilde{\ell}^2/(\ell\sqrt{\rho\epsilon})$, we have $\mathfrak{N} \approx 1 + \sigma^2/\epsilon^2$. Our results then show that perturbed SGD finds second-order stationary points in $\tilde{\mathcal{O}}(\epsilon^{-4})$ iterations, which matches Theorem 2.2.10 up to logarithmic factors.

In the general case where Assumption C does not hold ($\tilde{\ell} = \infty$), we have $\mathfrak{N} = 1 + \sigma^2 d/\epsilon^2$, and Theorem 2.4.4 guarantees that PSGD finds ϵ-second-order stationary point in $\tilde{\mathcal{O}}(d\epsilon^{-4})$ iterations. Comparing to Theorem 2.2.10, this pays an additional factor linear in dimension d.

Finally, Theorem 2.4.4 can be easily extended to the minibatch setting, with parameters chosen as:

$$\eta = \tilde{\Theta}(\frac{1}{\ell \cdot \mathfrak{M}}), \quad r = \tilde{\Theta}(\epsilon\sqrt{\mathfrak{M}}), \quad \text{where} \quad \mathfrak{M} = 1 + \frac{1}{m}\min\left\{\frac{\sigma^2}{\epsilon^2} + \frac{\tilde{\ell}^2}{\ell\sqrt{\rho\epsilon}}, \frac{\sigma^2 d}{\epsilon^2}\right\} \tag{2.5}$$

Theorem 2.4.5 (Mini-batch Version)**.** *For any $\epsilon, \delta, m > 0$, if function f satisfies Assumption A, and stochastic gradient $\mathbf{g}$ satisfies Assumption B, (and C optionally), and we run mini-batch PSGD (Algorithm 3) with parameter (η, r) chosen as Eq.(2.5). Then, with probability at least $1 - \delta$, mini-batch PSGD will visit an $\epsilon-$second-order stationary point at least once in the following number of iterations:*

$$\tilde{O}\left(\frac{\ell(f(\mathbf{x}_0) - f^\star)}{\epsilon^2} \cdot \mathfrak{M}\right)$$

Theorem 2.4.5 says that if the minibatch size m is not too large i.e., $m \leq \mathfrak{N}$, where $\mathfrak{N}$ is defined in Eq.(2.4), then mini-batch PSGD will reduce the number of iterations linearly, while not increasing the total number of stochastic gradient queries.

2.5 Conclusion

In this work, we show simple perturbed versions of GD and SGD escape saddle points and find second-order stationary points in essentially the same time GD and SGD take to find first-order stationary points. The overheads are only logarithmic factors in the non-stochastic setting, and the stochastic setting with Lipschitz stochastic gradient. In the general stochastic setting, the overhead is a linear factor in d.

Combined with previous landscape results on a wide class of nonconvex optimization in machine learning and signal processing, that all second-order stationary points are global optima, our results directly provide efficient guarantees for solving those nonconvex problem via simple local search approaches. We now discuss several possible future directions, and on connections to other fields.

Optimal rates for finding second-order stationary points According to Carmon et al. (2017b), GD achieves the optimal rate for finding stationary point for gradient Lipschitz functions. However, we note the results of this work assume, in addition, Lipschitz Hessian. This additional smooth structure of the function allows for more sophisticated algorithms to exploit it and achieve faster convergence rate. Therefore, GD and SGD or their variants themselves are no longer optimal algorithms.

The main focus of this work is to provide sharp guarantees to variants of the simplest algorithms in optimization—GD and SGD. Optimality is a separate topic worth further investigation. To find second-order stationary points of functions with Lipschitz gradient and Hessian via first-order algorithms, the best known gradient query complexity so far is $\tilde{\mathcal{O}}(\epsilon^{-1.75})$ achieved by Carmon et al. (2016), Agarwal et al. (2017), and Jin, Netrapalli, and Jordan (2017), while the existing lower bound is $\Omega(\epsilon^{-12/7})$ by Carmon et al. (2017c) with only a small gap of $\epsilon^{1/28}$. We also note that the current lower bound is restricted to only deterministic algorithms, thus does not apply to most existing algorithms for escaping saddle points as they are all randomized algorithms. For stochastic setting with Lipschitz stochastic gradient, the best query complexity of stochastic gradients is $\tilde{\mathcal{O}}(\epsilon^{-3})$ achieved by Fang et al. (2018) and Zhou and Gu (2019), while the lower bound remains open. See Appendix 2.8 for more comparison of exisiting works.

Escaping high-order saddle points. In this work, we study escaping strict saddle point and finding second-order stationary point. One can equivalently define n-th order stationary points as points which satisfies the KKT-necessary conditions for being local minima up to n-th order derivatives. It becomes more challenging to find n-th order stationary points as

n increases, since it requires escaping higher-order saddle points. In terms of efficiency, Nesterov (2000) rules out the possibility of efficient algorithms for finding n-th order stationary points for all $n \geq 4$, as the problem in general is NP-hard. Anandkumar and Ge (2016) present a third-order algorithm to find third-order stationary point in polynomial time. It remains open whether simple variants of GD can also find third-order stationary point efficiently. It is unlikely that the overhead will still be small or only logarithmic factors in this case. Another related question is to identify applications where third-order stationarity is needed beyond second-order stationarity to achieve global optimality.

Connection to gradient Langevin dynamics. A closely related algorithm in Bayesian statistics is the Langevin Monte Carlo (LMC) algorithm (Roberts and Tweedie, 1996), which performs the following iterative update:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta(\nabla f(\mathbf{x}_t) + \sqrt{2/(\eta\beta)}\mathbf{w}_t) \quad \text{where} \quad \mathbf{w}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}).$$

Here β is known as the inverse temperature. When learning rate $\eta \to 0$, the distribution of LMC iterates is known to converge to a stationary distribution $\mu(\mathbf{x}) \propto e^{-\beta f(\mathbf{x})}$ (Roberts and Tweedie, 1996).

While the LMC algorithm looks essentially the same to perturbed gradient descent considered in this work, there are two key differences on the standard settings between the two communities:

- **Goal**: While the focus of our work is to find a second order stationary point, the goal of LMC algorithm is to quickly converge to the stationary distribution (i.e., to mix rapidly).
- **Scaling of Noise**: The scaling of perturbation in this work is much smaller than the one considered in the standard LMC literature. Running our algorithm is equivalent to running LMC with temperature $\beta^{-1} \propto d^{-1}$. In this low temperature or small noise regime, the algorithm can no longer mix efficiently for smooth nonconvex function, as it takes $\Omega(e^d)$ steps in the worst case (Bovier et al., 2004). However, with this small amount of noise, the algorithm can still perform local search efficiently, and find a second-order stationary point in a small number of iterations as shown in Theorem 2.4.1.

Finally we note that a recent result (Zhang, Liang, and Charikar, 2017) studied, instead of mixing time, the time LMC takes to hit a second-order stationary point. The runtime is no longer exponential, but is still polynomially dependent on dimension d with large degree.

On the Necessity of Adding Perturbations. In this work, we discuss algorithms adding perturbations to every iteration of GD or SGD to escape saddle points efficiently. As an alternative, one can also simply run GD with random initialization, and try to escape saddle

Algorithm 4 Perturbed Gradient Descent (Variant)

Input: $\mathbf{x}_0$, learning rate η, perturbation radius r, time interval $\mathscr{T}$, tolerance ϵ.
$t_{\text{perturb}} = 0$
for $t = 0, 1, \ldots, T$ **do**
  if $\|\nabla f(\mathbf{x}_t)\| \le \epsilon$ and $t - t_{\text{perturb}} > \mathscr{T}$ **then**
    $\mathbf{x}_t \leftarrow \mathbf{x}_t - \eta\xi_t$, $(\xi_t \sim \text{Uniform}(B_0(r)))$; $\quad t_{\text{perturb}} \leftarrow t$
  $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta\nabla f(\mathbf{x}_t)$

points using only the randomness within the initialzation. Although this alternative algorithm exhibits asymptotic convergence (Lee et al., 2016), it does not yield efficient convergence in general. Du et al. (2017) shows that even with fairly natural random initialization schemes and non-pathological functions, GD with only random initialization can be significantly slowed by saddle points, taking exponential time to escape them.

2.6 Proofs for Non-stochastic Setting

In this section, we present our proof for the iteration complexity of PGD to find a second-order stationary point. While gradients are exact in Algorithm 1, the addition of perturbation in each step makes the algorithms stochastic in nature, and makes the analysis involves many concetrations inequalities and stochastic analysis. In order to illustrate the proof ideas and make the proof transparent, we present instead a proof for the iteration complexity of a variant of PGD (Algorithm 4), which is less stochastic. We leave the formal proof of Theorem 2.4.1 as a direct corollary of Theorem 2.4.4 by setting $\sigma = 0$.

Algorithm 4 adds perturbation only when the norm of gradient at current iterate is small, and the algorithm has not added perturbation in previous $\mathscr{T}$ iterations. Similar guarantees as Theorem 2.4.1 can be shown for this version of PGD as follows:

Theorem 2.6.1. *There is an absolute constant c such that the following holds: for any $\epsilon, \delta > 0$, if f satisfies Assumption A, let $\Delta_f := f(\mathbf{x}_0) - f^\star$ and we run PGD (Variant) (Algorithm 4) with parameters $\eta, r, \mathscr{T}$ chosen as Eq.(2.6) with $\iota = c \cdot \log(d\ell\Delta_f/(\rho\epsilon\delta))$, then with probability at least $1 - \delta$, in the following number of iterations, at least one half of iterations of PGD (Variant) will be $\epsilon-$second order stationary points.*

$$\tilde{O}\left(\frac{\ell\Delta_f}{\epsilon^2}\right),$$

In order to prove this theorem, we first specify our choice of hyperparameter $\eta, r, \mathscr{T}$, and two quantities $\mathscr{F}, \mathscr{S}$ which are frequently used:

$$\eta = \frac{1}{\ell}, \quad r = \frac{\epsilon}{400\iota^3}, \quad \mathscr{T} = \frac{\ell}{\sqrt{\rho\epsilon}} \cdot \iota, \quad \mathscr{F} = \frac{1}{50\iota^3}\sqrt{\frac{\epsilon^3}{\rho}}, \quad \mathscr{S} = \frac{1}{4\iota}\sqrt{\frac{\epsilon}{\rho}} \tag{2.6}$$

Our high-level proof strategy is to prove by contradiction: when the current iterate is not ϵ-second order stationary point, it must either have a large gradient or a strictly negative Hessian, and we prove that in either case, PGD must decrease a large amount of function value in a reasonable number of iterations. Finally since the function value can not decrease more than $f(\mathbf{x}_0) - f^\star$, we know that the total number of iterates that are not ϵ-second order stationary points can not be very large.

First, we show the decreasing speed when gradient is large.

Lemma 2.6.2 (Descent Lemma)**.** *If $f(\cdot)$ satisfies Assumption A and $\eta \le 1/\ell$, then the gradient descent sequence $\{\mathbf{x}_t\}$ satisfies:*

$$f(\mathbf{x}_{t+1}) - f(\mathbf{x}_t) \le -\eta \|\nabla f(\mathbf{x}_t)\|^2/2$$

Proof. According to the ℓ-gradient Lipschitz assumption, we have:

$$\begin{aligned} f(\mathbf{x}_{t+1}) \le & f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \mathbf{x}_{t+1} - \mathbf{x}_t \rangle + \frac{\ell}{2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \\ = & f(\mathbf{x}_t) - \eta \|\nabla f(\mathbf{x}_t)\|^2 + \frac{\eta^2 \ell}{2}\|\nabla f(\mathbf{x}_t)\|^2 \le f(\mathbf{x}_t) - \frac{\eta}{2}\|\nabla f(\mathbf{x}_t)\|^2 \end{aligned}$$

□

Next is our key lemma, which shows if the starting point has strictly negative Hessian, then adding a perturbation and following by gradient descent will decrease a large amount of function value in $\mathscr{T}$ iterations.

Lemma 2.6.3 (Escaping Saddle)**.** *Assume $f(\cdot)$ satisfies Assumption A and $\tilde{\mathbf{x}}$ satisfies $\|\nabla f(\tilde{\mathbf{x}})\| \le \epsilon$ and $\lambda_{\min}(\nabla^2 f(\tilde{\mathbf{x}})) \le -\sqrt{\rho\epsilon}$. Then let $\mathbf{x}_0 = \tilde{\mathbf{x}} + \eta\xi$ ($\xi \sim$ Uniform$(B_0(r))$) and run gradient descent starting from $\mathbf{x}_0$, we have*

$$\mathbb{P}(f(\mathbf{x}_{\mathscr{T}}) - f(\tilde{\mathbf{x}}) \le -\mathscr{F}/2) \ge 1 - \frac{\ell\sqrt{d}}{\sqrt{\rho\epsilon}} \cdot \iota^2 2^{8-\iota},$$

where $\mathbf{x}_{\mathscr{T}}$ is the $\mathscr{T}^{th}$ gradient descent iterate starting from $\mathbf{x}_0$.

In order to prove this, we need to prove two lemmas, and the major simplification over (Jin et al., 2017) comes from the following lemma which says that if function value does not decrease too much over t iterations, then all the iterates $\{\mathbf{x}_\tau\}_{\tau=0}^t$ will remain in a small neighborhood of $\mathbf{x}_0$.

Lemma 2.6.4 (Improve or Localize)**.** *Under the setting of Lemma 2.6.2, for any $t \ge \tau > 0$:*

$$\|\mathbf{x}_\tau - \mathbf{x}_0\| \le \sqrt{2\eta t (f(\mathbf{x}_0) - f(\mathbf{x}_t))}$$

Proof. Recall gradient update $\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \nabla f(\mathbf{x}_t)$, then for any $\tau \le t$:

$$\begin{aligned}\|\mathbf{x}_\tau - \mathbf{x}_0\| \le& \sum_{\tau=1}^{t} \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\| \overset{(1)}{\le} [t \sum_{\tau=1}^{t} \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|^2]^{\frac{1}{2}} \\ =& [\eta^2 t \sum_{\tau=1}^{t} \|\nabla f(\mathbf{x}_{\tau-1})\|^2]^{\frac{1}{2}} \overset{(2)}{\le} \sqrt{2\eta t (f(\mathbf{x}_0) - f(\mathbf{x}_t))}\end{aligned}$$

where step (1) uses Cauchy-Schwartz inequality, and step (2) is due to Lemma 2.6.2. □

Second, we show that the stuck region (where GD will get stuck in a small local neighborhood for at least $\mathscr{T}$ iterations if initialized there) is thin. We show this by tracking any pair of points that differ only in escaping direction, and are at least ω far apart. We show that, out of the two GD sequences that initialized at these two points, at least one sequence is guaranteed to escape the saddle point with high probability, so the stuck region along escaping direction has width at most ω.

Lemma 2.6.5 (Coupling Sequence)**.** *Suppose $f(\cdot)$ satisfies Assumption A and $\tilde{\mathbf{x}}$ satisfies $\lambda_{\min}(\nabla^2 f(\tilde{\mathbf{x}})) \le -\sqrt{\rho\epsilon}$. Let $\{\mathbf{x}_t\}$, $\{\mathbf{x}'_t\}$ be two gradient descent sequences which satisfy: (1) $\max\{\|\mathbf{x}_0 - \tilde{\mathbf{x}}\|, \|\mathbf{x}'_0 - \tilde{\mathbf{x}}\|\} \le \eta r$; (2) $\mathbf{x}_0 - \mathbf{x}'_0 = \eta r_0 \mathbf{e}_1$, where $\mathbf{e}_1$ is the minimum eigenvector direction of $\nabla^2 f(\tilde{\mathbf{x}})$ and $r_0 > \omega := 2^{2-\iota}\ell\mathscr{S}$. Then:*

$$\min\{f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0), f(\mathbf{x}'_{\mathscr{T}}) - f(\mathbf{x}'_0)\} \le -\mathscr{F}.$$

Proof. Assume the contrary, that is $\min\{f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0), f(\mathbf{x}'_{\mathscr{T}}) - f(\mathbf{x}'_0)\} > -\mathscr{F}$. Lemma 2.6.4 implies localization of both sequences around $\tilde{\mathbf{x}}$, that is for any $t \le \mathscr{T}$

$$\begin{aligned}\max\{\|\mathbf{x}_t - \tilde{\mathbf{x}}\|, \|\mathbf{x}'_t - \tilde{\mathbf{x}}\|\} \le& \max\{\|\mathbf{x}_t - \mathbf{x}_0\|, \|\mathbf{x}'_t - \mathbf{x}'_0\|\} + \max\{\|\mathbf{x}_0 - \tilde{\mathbf{x}}\|, \|\mathbf{x}'_0 - \tilde{\mathbf{x}}\|\} \\ \le& \sqrt{2\eta\mathscr{T}\mathscr{F}} + \eta r \le \mathscr{S} \end{aligned} \tag{2.7}$$

where the last step is due to our choice of $\eta, r, \mathscr{T}, \mathscr{F}, \mathscr{S}$ as in Eq.(2.6), and $\ell/\sqrt{\rho\epsilon} \ge 1$.[1] On the other hand, we can write the update equations for the difference $\hat{\mathbf{x}}_t := \mathbf{x}_t - \mathbf{x}'_t$ as:

$$\begin{aligned}\hat{\mathbf{x}}_{t+1} =& \hat{\mathbf{x}}_t - \eta[\nabla f(\mathbf{x}_t) - \nabla f(\mathbf{x}'_t)] = (\mathbf{I} - \eta\mathcal{H})\hat{\mathbf{x}}_t - \eta\Delta_t\hat{\mathbf{x}}_t \\ =& \underbrace{(\mathbf{I} - \eta\mathcal{H})^{t+1}\hat{\mathbf{x}}_0}_{\mathbf{p}(t+1)} - \underbrace{\eta\sum_{\tau=0}^{t}(\mathbf{I} - \eta\mathcal{H})^{t-\tau}\Delta_\tau\hat{\mathbf{x}}_\tau}_{\mathbf{q}(t+1)}\end{aligned}$$

where $\mathcal{H} = \nabla^2 f(\tilde{\mathbf{x}})$ and $\Delta_t = \int_0^1 [\nabla^2 f(\mathbf{x}'_t + \theta(\mathbf{x}_t - \mathbf{x}'_t) - \mathcal{H}]\mathrm{d}\theta$. We note $\mathbf{p}(t)$ is the leading term which is due to initial difference $\hat{x}_0$, and $\mathbf{q}(t)$ is the error term which is the result of

[1] We note that when $\ell/\sqrt{\rho\epsilon} < 1$, ϵ-second-order stationary points are equivalent to ϵ-first-order stationary points due to function f being ℓ-gradient Lipschitz. In this case, the problem of finding ϵ-second-order stationary points becomes very easy.

that function f is not quadratic. Now we use induction to show that the error term is always small compared to the leading term. That is:

$$\|\mathbf{q}(t)\| \le \|\mathbf{p}(t)\|/2, \qquad t \in [\mathscr{T}]$$

The claim is true for the base case $t = 0$ as $\|\mathbf{q}(0)\| = 0 \le \|\hat{\mathbf{x}}_0\|/2 = \|\mathbf{p}(0)\|/2$. Now suppose the induction claim is true till t, we prove it is true for $t+1$. Denote $\lambda_{\min}(\nabla^2 f(\mathbf{x}_0)) = -\gamma$. First, note $\hat{\mathbf{x}}_0$ is in the minimum eigenvector direction of $\nabla^2 f(\mathbf{x}_0)$. Thus for any $\tau \le t$, we have:

$$\|\hat{\mathbf{x}}_\tau\| \le \|\mathbf{p}(\tau)\| + \|\mathbf{q}(\tau)\| \le 2\|\mathbf{p}(\tau)\| = 2\|(\mathbf{I} - \eta\mathcal{H})^\tau \hat{\mathbf{x}}_0\| = 2(1+\eta\gamma)^\tau \eta r_0$$

By Hessian Lipschitz, we have $\|\Delta_t\| \le \rho \max\{\|\mathbf{x}_t - \tilde{\mathbf{x}}\|, \|\mathbf{x}'_t - \tilde{\mathbf{x}}\|\} \le \rho\mathscr{S}$, therefore:

$$\begin{aligned}\|\mathbf{q}(t+1)\| =& \|\eta \sum_{\tau=0}^{t} (\mathbf{I} - \eta\mathcal{H})^{t-\tau} \Delta_\tau \hat{\mathbf{x}}_\tau\| \le \eta\rho\mathscr{S} \sum_{\tau=0}^{t} \|(\mathbf{I} - \eta\mathcal{H})^{t-\tau}\| \|\hat{\mathbf{x}}_\tau\| \\ \le& 2\eta\rho\mathscr{S} \sum_{\tau=0}^{t} (1+\eta\gamma)^t \eta r_0 \le 2\eta\rho\mathscr{S}\mathscr{T}(1+\eta\gamma)^t \eta r_0 \le 2\eta\rho\mathscr{S}\mathscr{T}\|\mathbf{p}(t+1)\|\end{aligned}$$

where the second last inequality used $t + 1 \le \mathscr{T}$. By our choice of hyperparameter as in Eq.(2.6), we have $2\eta\rho\mathscr{S}\mathscr{T} \le 1/2$, which finishes the proof for induction.

Finally, the induction claim implies:

$$\begin{aligned}\max\{\|\mathbf{x}_{\mathscr{T}} - \mathbf{x}_0\|, \|\mathbf{x}'_{\mathscr{T}} - \mathbf{x}_0\|\} \ge& \frac{1}{2}\|\hat{\mathbf{x}}(\mathscr{T})\| \ge \frac{1}{2}[\|\mathbf{p}(\mathscr{T})\| - \|\mathbf{q}(\mathscr{T})\|] \ge \frac{1}{4}[\|\mathbf{p}(\mathscr{T})\| \\ =& \frac{(1+\eta\gamma)^{\mathscr{T}}\eta r_0}{4} \overset{(1)}{\ge} 2^{\iota-2}\eta r_0 > \mathscr{S}\end{aligned}$$

where step (1) uses the fact $(1+x)^{1/x} \ge 2$ for any $x \in (0, 1]$. This contradicts the localization fact Eq.(2.7), which finishes the proof. □

Equipped with Lemma 2.6.4 and Lemma 2.6.5, now we are ready to prove the Lemma 2.6.3.

Proof of Lemma 2.6.3. Recall $\mathbf{x}_0 \sim \text{Uniform}(B_{\tilde{\mathbf{x}}}(\eta r))$. We call $B_{\tilde{\mathbf{x}}}(\eta r)$ the perturbation ball, and define stuck region within it to be the set of points starting from which GD requires more than $\mathscr{T}$ steps to escape:

$$\mathcal{X}_{\text{stuck}} := \{\mathbf{x} \in B_{\tilde{\mathbf{x}}}(\eta r) \mid \{\mathbf{x}_t\} \text{ is GD sequence with } \mathbf{x}_0 = \mathbf{x}, \text{and } f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0) > -\mathscr{F}\}.$$

See Figure 2.1 and Figure 2.2 for illustrations. Although the shape of stuck region can be very complicated, according to Lemma 2.6.5, we know the width of $\mathcal{X}_{\text{stuck}}$ along $\mathbf{e}_1$ direction is at most $\eta\omega$. That is, $\text{Vol}(\mathcal{X}_{\text{stuck}}) \le \text{Vol}(\mathbb{B}_0^{d-1}(\eta r))\eta\omega$. Therefore:

$$\mathbb{P}(\mathbf{x}_0 \in \mathcal{X}_{\text{stuck}}) = \frac{\text{Vol}(\mathcal{X}_{\text{stuck}})}{\text{Vol}(\mathbb{B}_{\tilde{\mathbf{x}}}^d(\eta r))} \le \frac{\eta\omega \times \text{Vol}(\mathbb{B}_0^{d-1}(\eta r))}{\text{Vol}(\mathbb{B}_0^d(\eta r))} = \frac{\omega}{r\sqrt{\pi}} \frac{\Gamma(\frac{d}{2}+1)}{\Gamma(\frac{d}{2}+\frac{1}{2})} \le \frac{\omega}{r} \cdot \sqrt{\frac{d}{\pi}} \le \frac{\ell\sqrt{d}}{\sqrt{\rho\epsilon}} \cdot \iota^2 2^{8-\iota}$$

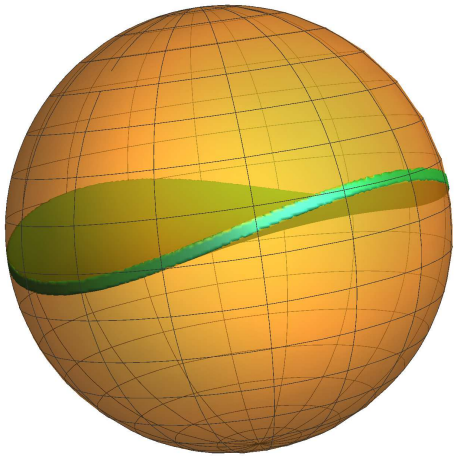

Figure 2.1: Pertubation ball in 3D and "thin pancake" shape stuck region

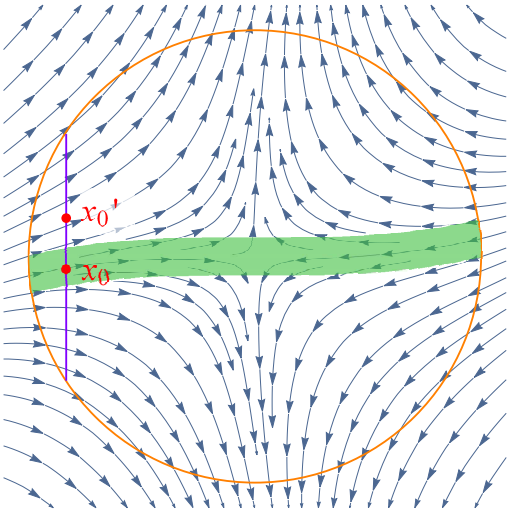

Figure 2.2: Pertubation ball in 2D and "narrow band" stuck region under gradient flow

On the event that $\mathbf{x}_0 \notin \mathcal{X}_{\text{stuck}}$, according to our parameter choice Eq.(2.6), we have:

$$f(\mathbf{x}_{\mathscr{T}}) - f(\tilde{\mathbf{x}}) = [f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0)] + [f(\mathbf{x}_0) - f(\tilde{\mathbf{x}})] \le -\mathscr{F} + \epsilon\eta r + \frac{\ell\eta^2 r^2}{2} \le -\mathscr{F}/2.$$

This finishes the proof. □

With Lemma 2.6.2 and Lemma 2.6.3, it is not hard to finally prove Theorem 2.6.1.

Proof of Theorem 2.6.1. First, we set total iterations T to be:

$$T = 8\max\left\{\frac{(f(x_0) - f^\star)\mathscr{T}}{\mathscr{F}}, \frac{(f(x_0) - f^\star)}{\eta\epsilon^2}\right\} = O\left(\frac{\ell(f(x_0) - f^\star)}{\epsilon^2} \cdot \iota^4\right)$$

Next, we choose $\iota = c \cdot \log(\frac{d\ell\Delta_f}{\rho\epsilon\delta})$ with large enough absolute constant c so that:

$$(T\ell\sqrt{d}/\sqrt{\rho\epsilon}) \cdot \iota^2 2^{8-\iota} \le \delta.$$

Then, we argue with probability $1-\delta$, algorithm 4 will add perturbation at most $T/(4\mathscr{T})$ times. This is because otherwise, we can use Lemma 2.6.3 every time we add perturbation, and:

$$f(\mathbf{x}_T) \le f(\mathbf{x}_0) - T\mathscr{F}/(4\mathscr{T}) < f^\star$$

which can not happen. Finally, excluding those iterations that are within $\mathscr{T}$ steps after adding perturbations, we still have $3T/4$ steps left. They are either large gradient steps

$\|\nabla f(\mathbf{x}_t)\| \geq \epsilon$ or ϵ-second order stationary points. Within them, we know large gradient steps can not be more than $T/4$. Because again otherwise, by Lemma 2.6.2:

$$f(\mathbf{x}_T) \leq f(\mathbf{x}_0) - T\eta\epsilon^2/4 < f^\star$$

which again can not happen. Therefore, we conclude at least $T/2$ iterations must be $\epsilon-$second order stationary points. □

2.7 Proofs for Stochastic Setting

In this section, we provide proofs for our main results—Theorem 2.4.4 and Theorem 2.4.5. Theorem 2.4.1 can be proved as a special case of Theorem 2.4.4 by taking $\sigma = 0$.

Notation

Recall the update equation of Algorithm 2 is $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta(\mathbf{g}(\mathbf{x}_t;\theta_t) + \xi_t)$ where $\xi_t \sim \mathcal{N}(\mathbf{0}, (r^2/d)\mathbf{I})$. Across this section, we denote $\zeta_t := \mathbf{g}(\mathbf{x}_t;\theta_t) - \nabla f(\mathbf{x}_t)$, as the noise part within the stochastic gradient. For simplicity, we also denote $\tilde{\zeta}_t := \zeta_t + \xi_t$, which is the summation of noise in stochastic gradient and the injected perturbation, and $\tilde{\sigma}^2 := \sigma^2 + r^2$. Then the update equation can be rewrite as $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta(\nabla f(\mathbf{x}_t) + \tilde{\zeta}_t)$. We also denote $\mathcal{F}_t = \sigma(\zeta_0, \xi_0, \ldots, \zeta_t, \xi_t)$ be the corresponding filteration up to time step t. We choose parameters in Algorithm 2 as follows:

$$\eta = \frac{1}{\iota^9 \cdot \ell\mathfrak{N}}, \qquad r = \iota \cdot \epsilon\sqrt{\mathfrak{N}}, \qquad \mathscr{T} := \frac{\iota}{\eta\sqrt{\rho\epsilon}}, \qquad \mathscr{F} := \frac{1}{\iota^5}\sqrt{\frac{\epsilon^3}{\rho}}, \qquad \mathscr{S} := \frac{2}{\iota^2}\sqrt{\frac{\epsilon}{\rho}} \tag{2.8}$$

where $\mathfrak{N}$ and log factor ι are defined as:

$$\mathfrak{N} = 1 + \min\left\{\frac{\sigma^2}{\epsilon^2} + \frac{\tilde{\ell}^2}{\ell\sqrt{\rho\epsilon}},\ \frac{\sigma^2 d}{\epsilon^2}\right\}, \qquad \iota = \mu \cdot \log\left(\frac{d\ell\Delta_f\mathfrak{N}}{\rho\epsilon\delta}\right)$$

Here, μ is a sufficiently large absolute constant to be determined later. Also we note c in this sections are absolute constant that does not depend on the choice of μ. The value of c may change from line to line.

Descent Lemma

We first prove that the change in the function value can be always decomposed as the decrease due to the magnitudes of gradients, and the possible increase due to randomness in both stochastic gradients and perturbations.

Lemma 2.7.1 (Descent Lemma)**.** *There exists absolute constant c, under Assumption A, B, for any fixed $t, t_0, \iota > 0$, if $\eta \le 1/\ell$, then with at least $1 - 4e^{-\iota}$ probability, the sequence PSGD(η, r) (Algorithm 2) satisfies: (denote $\tilde{\sigma}^2 = \sigma^2 + r^2$)*

$$f(\mathbf{x}_{t_0+t}) - f(\mathbf{x}_{t_0}) \le -\frac{\eta}{8}\sum_{i=0}^{t-1}\|\nabla f(\mathbf{x}_{t_0+i})\|^2 + c\cdot\eta\tilde{\sigma}^2(\eta\ell t + \iota)$$

Proof. Since Algorithm 2 is Markovian, the operations in each iterations does not depend on time step t. Thus, it suffices to prove Lemma 2.7.1 for special case $t_0 = 0$. Recall the update equation:

$$\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta(\nabla f(\mathbf{x}_t) + \tilde{\zeta}_t)$$

where $\tilde{\zeta}_t = \zeta_t + \xi_t$. By assumption, we know $\zeta_t|\mathcal{F}_{t-1}$ is zero-mean nSG(σ). Also $\xi_t|\mathcal{F}_{t-1}$ comes from $\mathcal{N}(\mathbf{0}, (r^2/d)\mathbf{I})$, and thus by Lemma 2.9.2 is zero-mean nSG$(c\cdot r)$ for some absolute constant c. By Taylor expansion, ℓ-gradient Lipschitz and $\eta \le 1/\ell$, we know:

$$\begin{aligned}
f(\mathbf{x}_{t+1}) \le& f(\mathbf{x}_t) + \langle\nabla f(\mathbf{x}_t), \mathbf{x}_{t+1} - \mathbf{x}_t\rangle + \frac{\ell}{2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \\
\le& f(\mathbf{x}_t) - \eta\langle\nabla f(\mathbf{x}_t), \nabla f(\mathbf{x}_t) + \tilde{\zeta}_t\rangle + \frac{\eta^2\ell}{2}\left[\frac{3}{2}\|\nabla f(\mathbf{x}_t)\|^2 + 3\|\tilde{\zeta}_t\|^2\right] \\
\le& f(\mathbf{x}_t) - \frac{\eta}{4}\|\nabla f(\mathbf{x}_t)\|^2 - \eta\langle\nabla f(\mathbf{x}_t), \tilde{\zeta}_t\rangle + \frac{3}{2}\eta^2\ell\|\tilde{\zeta}_t\|^2
\end{aligned}$$

Summing over the inequality above, we have following:

$$f(\mathbf{x}_t) - f(\mathbf{x}_0) \le -\frac{\eta}{4}\sum_{i=0}^{t-1}\|\nabla f(\mathbf{x}_i)\|^2 - \eta\sum_{i=0}^{t-1}\langle\nabla f(\mathbf{x}_i), \tilde{\zeta}_i\rangle + \frac{3}{2}\eta^2\ell\sum_{i=0}^{t-1}\|\tilde{\zeta}_i\|^2 \tag{2.9}$$

For the second term in RHS, applying Lemma 2.9.8, there exists an absolute constant c, with probability $1 - 2e^{-\iota}$:

$$-\eta\sum_{i=0}^{t-1}\langle\nabla f(\mathbf{x}_i), \tilde{\zeta}_i\rangle \le \frac{\eta}{8}\sum_{i=0}^{t-1}\|\nabla f(\mathbf{x}_i)\|^2 + c\eta\tilde{\sigma}^2\iota$$

For the third term in RHS of Eq.(2.9), applying Lemma 2.9.7, with probability $1 - 2e^{-\iota}$:

$$\frac{3}{2}\eta^2\ell\sum_{i=0}^{t-1}\|\tilde{\zeta}_i\|^2 \le 3\eta^2\ell\sum_{i=0}^{t-1}(\|\zeta_i\|^2 + \|\xi_i\|^2) \le c\eta^2\ell\tilde{\sigma}^2(t+\iota)$$

Substituting both above inequality into Eq.(2.9), and note the fact $\eta \le 1/\ell$, we have with probability $1 - 4e^{-\iota}$:

$$f(\mathbf{x}_t) - f(\mathbf{x}_0) \le -\frac{\eta}{8}\sum_{i=0}^{t-1}\|\nabla f(\mathbf{x}_i)\|^2 + c\eta\tilde{\sigma}^2(\eta\ell t + \iota)$$

This finishes the proof. □

The descent lemma enables as to show following Improve or Localize phenomena for perturbed SGD. That is, with high probability over a small number of iterations, either the function value decrease significantly, or the iterates stay within a small local region.

Lemma 2.7.2 (Improve or Localize)**.** *Under the same setting of Lemma 2.7.1, with at least* $1 - 8dt \cdot e^{-\iota}$ *probability, the sequence PSGD*(η, r) *(Algorithm 2) satisfies:*

$$\forall \tau \leq t : \|\mathbf{x}_{t_0+\tau} - \mathbf{x}_{t_0}\|^2 \leq c\eta t \cdot [f(\mathbf{x}_{t_0}) - f(\mathbf{x}_{t_0+\tau}) + \eta\tilde{\sigma}^2(\eta\ell t + \iota)]$$

Proof. By similar arguement as in proof of Lemma 2.7.1, it suffices to prove Lemma 2.7.2 in special case $t_0 = 0$. According to Lemma 2.7.1, with probability $1 - 4e^{-\iota}$, for some absolute constant c:

$$\sum_{i=0}^{t-1} \|\nabla f(\mathbf{x}_i)\|^2 \leq \frac{8}{\eta}[f(\mathbf{x}_0) - f(\mathbf{x}_t)] + c\tilde{\sigma}^2(\eta\ell t + \iota)$$

Therefore, for any fixed $\tau \leq t$, with probability $1 - 8d \cdot e^{-\iota}$,:

$$\begin{aligned}
\|\mathbf{x}_\tau - \mathbf{x}_0\|^2 =& \eta^2 \|\sum_{i=0}^{\tau-1} (\nabla f(\mathbf{x}_i) + \tilde{\zeta}_i)\|^2 \leq 2\eta^2 \left[\|\sum_{i=0}^{\tau-1} \nabla f(\mathbf{x}_i)\|^2 + \|\sum_{i=0}^{\tau-1} \tilde{\zeta}_i\|^2 \right] \\
\overset{(1)}{\leq}& 2\eta^2 t \sum_{i=0}^{\tau-1} \|\nabla f(\mathbf{x}_i)\|^2 + c\eta^2\tilde{\sigma}^2 t\iota \leq 2\eta^2 t \sum_{i=0}^{t-1} \|\nabla f(\mathbf{x}_i)\|^2 + c\eta^2\tilde{\sigma}^2 t\iota \\
\leq& c\eta t[f(\mathbf{x}_0) - f(\mathbf{x}_t) + \eta\tilde{\sigma}^2(\eta\ell t + \iota)]
\end{aligned}$$

Where in step (1) we use Cauchy-Schwartz inequality and Lemma 2.9.5. Finally, applying union bound for all $\tau \leq t$, we finishes the proof. □

Escaping Saddle Points

Descent Lemma 2.7.1 shows that large gradients contribute to the fast decrease of the function value. In this subsection, we will show that starting in the vicinity of strict saddle points will also enable PSGD to decrease the function value rapidly. Concretely, this entire subsection will be devoted to prove following lemma:

Lemma 2.7.3 (Escaping Saddle Point)**.** *There exists absolute constant* $c_{\max}$*, under Assumption A, B, for any fixed* $t_0 > 0, \iota > c_{\max}\log(\ell\sqrt{d/(\rho\epsilon)})$*, if* $\eta, r, \mathscr{F}, \mathscr{T}$ *are chosen as in Eq.(2.8), and* $\mathbf{x}_{t_0}$ *satisfies* $\|\nabla f(\mathbf{x}_{t_0})\| \leq \epsilon$ *and* $\lambda_{\min}(\nabla^2 f(\mathbf{x}_{t_0})) \leq -\sqrt{\rho\epsilon}$*, then the sequence PSGD*(η, r) *(Algorithm 2) satisfies:*

$$\begin{aligned}
&\mathbb{P}(f(\mathbf{x}_{t_0+\mathscr{T}}) - f(\mathbf{x}_{t_0}) \leq 0.1\mathscr{F}) \geq 1 - 4e^{-\iota} \qquad \textit{and} \\
&\mathbb{P}(f(\mathbf{x}_{t_0+\mathscr{T}}) - f(\mathbf{x}_{t_0}) \leq -\mathscr{F}) \geq 1/3 - 5d\mathscr{T}^2 \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}
\end{aligned}$$

Since Algorithm 2 is Markovian, the operations in each iterations does not depend on time step t. Thus, it suffices to prove Lemma 2.7.3 for special case $t_0 = 0$. To prove this lemma, we first need to introduce the concept of coupling sequence.

Notation: Across this subsection, we let $\mathcal{H} := \nabla^2 f(\mathbf{x}_0)$, and $\mathbf{e}_1$ be the minimum eigendirection of $\mathcal{H}$, and $\gamma := \lambda_{\min}(\mathcal{H})$. We also let $\mathcal{P}_{-1}$ be the projection to subspace complement to $\mathbf{e}_1$.

To prove the lemma, we introduce an important concept—coupling sequence.

Definition 2.7.4 (Coupling Sequence)**.** Consider two sequences $\{\mathbf{x}_i\}$ and $\{\mathbf{x}'_i\}$ as two seperate runs of PSGD (algorithm 2) both starting from $\mathbf{x}_0$. They are coupled if both sequences share the same randomness $\mathcal{P}_{-1}\xi_\tau$ and θ_τ, while in $\mathbf{e}_1$ direction $\mathbf{e}_1^\top \xi_\tau = -\mathbf{e}_1^\top \xi'_\tau$.

The first thing we can show is that if function values of both sequences do not have sufficient decreases, then both sequences are localized in a small ball around $\mathbf{x}_0$ within $\mathscr{T}$ iterations.

Lemma 2.7.5 (Localization of coupling sequence)**.** *Under the notation of Lemma 2.7.6, then:*

$$\mathbb{P}(\min\{f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0), f(\mathbf{x}'_{\mathscr{T}}) - f(\mathbf{x}_0)\} \le -\mathscr{F}, \quad \text{or} \\ \forall t \le \mathscr{T} : \max\{\|\mathbf{x}_t - \mathbf{x}_0\|^2, \|\mathbf{x}'_t - \mathbf{x}_0\|^2\} \le \mathscr{S}^2) \ge 1 - 16d\mathscr{T} \cdot e^{-\iota}$$

Proof. This lemma follows from applying Lemma 2.7.2 on both sequences and union bound. □

The overall proof strategy for Lemma 2.7.3 is to show localization happens with a very small chance, thus at least one of the sequence must have sufficient descent. In order to prove so, we study the dynamics of the difference of the coupling sequence.

Lemma 2.7.6 (Dynamics of the difference of coupling sequence)**.** *Consider coupling sequence $\{\mathbf{x}_i\}$ and $\{\mathbf{x}'_i\}$ as in Definition 2.7.4 and let $\hat{\mathbf{x}}_t := \mathbf{x}_i - \mathbf{x}'_i$. Then $\hat{\mathbf{x}}_t = -\mathbf{q}_h(t) - \mathbf{q}_{sg}(t) - \mathbf{q}_p(t)$, where:*

$$\mathbf{q}_h(t) := \eta \sum_{\tau=0}^{t-1} (\mathbf{I} - \eta\mathcal{H})^{t-1-\tau} \Delta_\tau \hat{\mathbf{x}}_\tau, \;\; \mathbf{q}_{sg}(t) := \eta \sum_{\tau=0}^{t-1} (\mathbf{I} - \eta\mathcal{H})^{t-1-\tau} \hat{\zeta}_\tau, \;\; \mathbf{q}_p(t) := \eta \sum_{\tau=0}^{t-1} (\mathbf{I} - \eta\mathcal{H})^{t-1-\tau} \hat{\xi}_\tau$$

Here $\Delta_t := \int_0^1 \nabla^2 f(\psi \mathbf{x}_t + (1-\psi)\mathbf{x}'_t)\mathrm{d}\psi - \mathcal{H}$, and $\hat{\zeta}_\tau := \zeta_\tau - \zeta'_\tau$, $\hat{\xi}_\tau := \xi_\tau - \xi'_\tau$.

Proof. Recall $\zeta_i = \mathbf{g}(\mathbf{x}_i; \theta_i) - \nabla f(\mathbf{x}_i)$, thus, we have update formula:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta(\mathbf{g}(\mathbf{x}_t; \theta_t) + \xi_t) = \mathbf{x}_t - \eta(\nabla f(\mathbf{x}_t) + \zeta_t + \xi_t)$$

Taking the difference between $\{\mathbf{x}_i\}$ and $\{\mathbf{x}'_i\}$:

$$\begin{aligned} \hat{\mathbf{x}}_{t+1} =& \mathbf{x}_{t+1} - \mathbf{x}'_{t+1} = \hat{\mathbf{x}}_t - \eta(\nabla f(\mathbf{x}_t) - \nabla f(\mathbf{x}'_t) + \zeta_t - \zeta'_t + (\xi_t - \xi'_t)) \\ =& \hat{\mathbf{x}}_t - \eta[(\mathcal{H} + \Delta_t)\hat{\mathbf{x}}_t + \hat{\zeta}_t + \hat{\xi}_t] = (\mathbf{I} - \eta\mathcal{H})\hat{\mathbf{x}}_t - \eta[\Delta_t \hat{\mathbf{x}}_t + \hat{\zeta}_t + \mathbf{e}_1\mathbf{e}_1^\top \hat{\xi}_t] \\ =& -\eta \sum_{\tau=0} t(I - \eta\mathcal{H})^{t-\tau}(\Delta_\tau \hat{\mathbf{x}}_\tau + \hat{\zeta}_\tau + \hat{\xi}_\tau)) \end{aligned}$$

where $\Delta_t := \int_0^1 \nabla^2 f(\psi \mathbf{x}_t + (1-\psi)\mathbf{x}'_t)\mathrm{d}\psi - \mathcal{H}$. This finishes the proof. □

In high level, we will show with constant probability, $\mathbf{q}_p(t)$ is the dominating term which controls the major behavior of the dynamics, and $\mathbf{q}_h(t)$, $\mathbf{q}_{sg}(t)$ will stay small compared to $\mathbf{q}_p(t)$. To achieve this, we prove following three lemmas.

Lemma 2.7.7. *Denote* $\alpha(t) := \left[\sum_{\tau=0}^{t-1}(1+\eta\gamma)^{2(t-1-\tau)}\right]^{\frac{1}{2}}; \beta(t) := (1+\eta\gamma)^t/\sqrt{2\eta\gamma}$. *If* $\eta\gamma \in [0,1]$, *then (1)* $\alpha(t) \le \beta(t)$ *for any* $t \in \mathbb{N}$; *(2)* $\alpha(t) \ge \beta(t)/\sqrt{3}$ *for* $t \ge \ln(2)/(\eta\gamma)$.

Proof. By summation formula of geometric sequence:

$$\alpha^2(t) := \sum_{\tau=0}^{t-1}(1+\eta\gamma)^{2(t-1-\tau)} = \frac{(1+\eta\gamma)^{2t}-1}{2\eta\gamma+(\eta\gamma)^2}$$

Thus, the claim $\alpha(t) \le \beta(t)$ for any $t \in \mathbb{N}$ immediately follows. On the other hand, note for $t \ge \ln(2)/(\eta\gamma)$, we have $(1+\eta\gamma)^{2t} \ge 2^{2\ln 2} \ge 2$, where the second claim follows by calculations. □

Lemma 2.7.8. *Under the notation of Lemma 2.7.6 and Lemma 2.7.7, let* $-\gamma := \lambda_{\min}(\mathcal{H})$, *then* $\forall t > 0$*:*

$$\mathbb{P}(\|\mathbf{q}_p(t)\| \le \frac{c\beta(t)\eta r}{\sqrt{d}} \cdot \sqrt{\iota}) \ge 1 - 2e^{-\iota}$$
$$\mathbb{P}(\|\mathbf{q}_p(\mathscr{T})\| \ge \frac{\beta(\mathscr{T})\eta r}{10\sqrt{d}}) \ge \frac{2}{3}$$

Proof. Note $\hat{\xi}_\tau$ is one dimensional Gaussian with standard deviation $2r/\sqrt{d}$ along $\mathbf{e}_1$ direction. As a immediate result, $\eta\sum_{\tau=0}^{t}(I-\eta\mathcal{H})^{t-\tau}\hat{\xi}_\tau$ also satisfies one dimensional Gaussian distribution since summation of Gaussian is again Gaussian. Finally note $\mathbf{e}_1$ is an eigendirection of $\mathcal{H}$ with corresponding eigenvalue $-\gamma$, and by Lemma 2.7.7 that $\alpha(t) \le \beta(t)$. Then, the first inequality immediately follows from the standard concentration inequality for Gaussian; the second inequality follows from the fact if $Z \sim \mathcal{N}(0,\sigma^2)$ then $\mathbb{P}(|Z| \le \lambda\sigma) \le 2\lambda/\sqrt{2\pi} \le \lambda$. □

Lemma 2.7.9. *There exists absolute constant* $c_{\max}$, *for any* $\iota \ge c_{\max}$, *under the notation of Lemma 2.7.6 and Lemma 2.7.7, let* $-\gamma := \lambda_{\min}(\mathcal{H})$, *we have:*

$$\mathbb{P}(\min\{f(\mathbf{x}_{\mathscr{I}}) - f(\mathbf{x}_0), f(\mathbf{x}'_{\mathscr{I}}) - f(\mathbf{x}_0)\} \le -\mathscr{F}, \quad \text{or}$$
$$\forall t \le \mathscr{T} : \|\mathbf{q}_h(t) + \mathbf{q}_{sg}(t)\| \le \frac{\beta(t)\eta r}{20\sqrt{d}}) \ge 1 - 10d\mathscr{T}^2 \cdot \log(\frac{\mathscr{S}\sqrt{d}}{\eta r})e^{-\iota}$$

Proof. For simplicity we denote $\mathfrak{E}$ to be the event $\{\forall\tau \le t : \max\{\|\mathbf{x}_\tau - \mathbf{x}_0\|^2, \|\mathbf{x}'_\tau - \mathbf{x}_0\|^2\} \le \mathscr{S}^2\}$. We use induction to prove following claims for any $t \in [0, \mathscr{T}]$:

$$\mathbb{P}(\mathfrak{E} \ \Rightarrow \ \forall\tau \le t : \|\mathbf{q}_h(\tau) + \mathbf{q}_{sg}(\tau)\| \le \frac{\beta(\tau)\eta r}{20\sqrt{d}}) \ge 1 - 10d\mathscr{T}t \cdot \log(\frac{\mathscr{S}\sqrt{d}}{\eta r})e^{-\iota}$$

Then Lemma 2.7.9 directly follow from combining Lemma 2.7.5 and this induction claim.

Clearly for the base case $t = 0$, the claim trivially holds as $\mathbf{q}_{sg}(0) = \mathbf{q}_h(0) = \mathbf{0}$. Suppose the claim holds for t, then by Lemma 2.7.8, with probability at least $1 - 2\mathscr{T}e^{-\iota}$, we have for any $\tau \le t$:

$$\|\hat{\mathbf{x}}_\tau\| \le \eta\|\mathbf{q}_h(\tau) + \mathbf{q}_{sg}(\tau)\| + \eta\|\mathbf{q}_p(\tau)\| \le \frac{c\beta(\tau)\eta r}{\sqrt{d}} \cdot \sqrt{\iota}$$

Then, under the condition $\max\{\|\mathbf{x}_\tau - \mathbf{x}_0\|^2, \|\mathbf{x}'_\tau - \mathbf{x}_0\|^2\} \le \mathscr{S}^2$, by Hessian Lipschitz, we have $\|\Delta_\tau\| = \|\int_0^1 \nabla^2 f(\psi\mathbf{x}_\tau + (1-\psi)\mathbf{x}'_\tau)\mathrm{d}\psi - \mathcal{H}\| \le \rho\max\{\|\mathbf{x}_\tau - \mathbf{x}_0\|, \|\mathbf{x}'_\tau - \mathbf{x}_0\|\} \le \rho\mathscr{S}$. This gives bounds on $\mathbf{q}_h(t+1)$ terms as:

$$\|\mathbf{q}_h(t+1)\| \le \eta\sum_{\tau=0}^{t}(1+\eta\gamma)^{t-\tau}\rho\mathscr{S}\|\hat{\mathbf{x}}_\tau\| \le \eta\rho\mathscr{S}\mathscr{T}\frac{c\beta(t)\eta r}{\sqrt{d}} \le \frac{\beta(t)\eta r}{40\sqrt{d}}$$

where the last step is due to $\eta\rho\mathscr{S}\mathscr{T} = 1/\iota$ by Eq.(2.8). By picking ι larger than absolute constant $40c$, then we have $c\eta\rho\mathscr{S}\mathscr{T} \le 1/40$.

Also, recall $\hat{\zeta}_\tau|\mathcal{F}_{\tau-1}$ is the summation of one $\mathrm{nSG}(\sigma)$ random vector and one $\mathrm{nSG}(c \cdot r)$ random vector, by Lemma 2.9.5, we know with probability at least $1 - 4de^{-\iota}$:

$$\|\mathbf{q}_{sg}(t+1)\| \le c\beta(t+1)\eta\sigma\sqrt{\iota}$$

On the other hand, when assumption C is avaliable, we also have $\hat{\zeta}_\tau|\mathcal{F}_{\tau-1} \sim \mathrm{nSG}(\tilde{\ell}\|\hat{\mathbf{x}}_\tau\|)$, by applying Lemma 2.9.6 with $B = \alpha^2(t)\cdot\eta^2\tilde{\ell}^2\mathscr{S}^2; b = \alpha^2(t)\cdot\eta^2\tilde{\ell}^2\cdot\eta^2 r^2/d$, we know with probability at least $1 - 4d\cdot\log(\mathscr{S}\sqrt{d}/(\eta r))\cdot e^{-\iota}$:

$$\|\mathbf{q}_{sg}(t+1)\| \le c\eta\tilde{\ell}\sqrt{\sum_{\tau=0}^{t}(1+\eta\gamma)^{2(t-\tau)}\cdot\max\{\|\hat{\mathbf{x}}_\tau\|^2, \frac{\eta^2 r^2}{d}\}\iota} \le \eta\tilde{\ell}\sqrt{\mathscr{T}}\cdot\frac{c\beta(t)\eta r}{\sqrt{d}}\cdot\sqrt{\iota} \quad (2.10)$$

Finally, combine both cases, and by our choice of learning rate η, r as in Eq.(2.8) with ι large enough:

$$\|\mathbf{q}_{sg}(t+1)\| \le c\frac{\beta(t)\eta r}{\sqrt{d}}\cdot\min\{\eta\tilde{\ell}\sqrt{\mathscr{T}\iota}, \frac{\sigma\sqrt{d\iota}}{r}\} \le \frac{\beta(t)r}{40\sqrt{d}}$$

and the induction follows by triangular inequality and union bound. □

Now, we are ready to prove the Lemma 2.7.3, which is the focus of this subsection.

Proof of Lemma 2.7.3. We first prove the first claim $\mathbb{P}(f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0) \le 0.1\mathscr{F}) \ge 1 - 4e^{-\iota}$. This is essentially because our choice of learning rate and Lemma 2.7.1, we have with probability $1 - 4e^{-\iota}$:

$$f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0) \le c\eta\tilde{\sigma}^2(\eta\ell\mathscr{T} + \iota) \le 0.1\mathscr{F}$$

where the last step is because of our choice of parameters as Eq.(2.8), we have $c\eta\tilde{\sigma}^2(\eta\ell\mathscr{T} + \iota) \le 2c\mathscr{F}/\iota$ and by picking ι larger than absolute constant $20c$.

For the second claim $\mathbb{P}(f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0) \le -\mathscr{F}) \ge 1/3 - 5d\mathscr{T}^2 \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}$. We consider coupling sequences $\{\mathbf{x}_i\}$ and $\{\mathbf{x}'_i\}$ as defined in Definition 2.7.4. We note Lemma 2.7.8 and Lemma 2.7.9, we know with probability at least $2/3 - 10d\mathscr{T}^2 \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}$, if $\min\{f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0), f(\mathbf{x}'_{\mathscr{T}}) - f(\mathbf{x}_0)\} > -\mathscr{F}$, i.e. both sequences stuck around the saddle point, then we have:

$$\|\mathbf{q}_p(\mathscr{T})\| \ge \frac{\beta(\mathscr{T})\eta r}{10\sqrt{d}}, \quad \|\mathbf{q}_h(\mathscr{T}) + \mathbf{q}_{sg}(\mathscr{T})\| \le \frac{\beta(\mathscr{T})\eta r}{20\sqrt{d}}$$

By Lemma 2.7.6, when $\iota \ge c \cdot \log(\ell\sqrt{d/(\rho\epsilon)})$ with large absolute constant c, we have:

$$\begin{aligned}\max\{\|\mathbf{x}_{\mathscr{T}} - \mathbf{x}_0\|, \|\mathbf{x}'_{\mathscr{T}} - \mathbf{x}_0\|\} \ge& \frac{1}{2}\|\hat{\mathbf{x}}(\mathscr{T})\| \ge \frac{1}{2}[\|\mathbf{q}_p(\mathscr{T})\| - \|\mathbf{q}_h(\mathscr{T}) + \mathbf{q}_{sg}(\mathscr{T})\|] \\ \ge& \frac{\beta(\mathscr{T})\eta r}{40\sqrt{d}} = \frac{(1+\eta\gamma)^{\mathscr{T}}\eta r}{40\sqrt{2\eta\gamma d}} \le \frac{2^{\iota}\eta r}{80\sqrt{\eta\ell d}} > \mathscr{S}\end{aligned}$$

which contradicts with Lemma 2.7.5. Therefore, we can conclude that $\mathbb{P}(\min\{f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0), f(\mathbf{x}'_{\mathscr{T}}) - f(\mathbf{x}_0)\} \le -\mathscr{F}) \ge 2/3 - 10d\mathscr{T}^2 \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}$. We also know the marginal distribution of $\mathbf{x}_{\mathscr{T}}$ and $\mathbf{x}'_{\mathscr{T}}$ is the same, thus they have same probability to escape saddle point. That is:

$$\begin{aligned}\mathbb{P}(f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0) \le -\mathscr{F}) \ge& \frac{1}{2}\mathbb{P}(\min\{f(\mathbf{x}_{\mathscr{T}}) - f(\mathbf{x}_0), f(\mathbf{x}'_{\mathscr{T}}) - f(\mathbf{x}_0)\} \le -\mathscr{F}) \\ \ge& 1/3 - 5d\mathscr{T}^2 \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}\end{aligned}$$

This finishes the proof. □

Proof of Theorem 2.4.4

Lemma 2.7.1 and Lemma 2.7.3 describe the speed of decrease in the function values when either large gradients or strictly negative curvatures are present. Combining them gives the proof for our main theorem.

Proof of Theorem 2.4.4. First, we set total iterations T to be:

$$T = 100\max\left\{\frac{(f(x_0) - f^\star)\mathscr{T}}{\mathscr{F}}, \frac{(f(x_0) - f^\star)}{\eta\epsilon^2}\right\} = O\left(\frac{\ell(f(x_0) - f^\star)}{\epsilon^2} \cdot \mathfrak{N} \cdot \iota^9\right)$$

We will show that the following **two claims** hold simultaneously with $1 - \delta$ probability:

1. at most $T/4$ iterates has large gradient, i.e. $\|\nabla f(\mathbf{x}_t)\| \ge \epsilon$;
2. at most $T/4$ iterates are close to saddle points, i.e. $\|\nabla f(\mathbf{x}_t)\| \le \epsilon$ and $\lambda_{\min}(\nabla^2 f(\mathbf{x}_t)) \le -\sqrt{\rho\epsilon}$.

Therefore, at least $T/2$ iterates are ϵ-second order stationary point. We prove two claims separately.

Claim 1. Suppose within T steps, we have more than $T/4$ iterates that gradient is large (i.e. $\|\nabla f(\mathbf{x}_t)\| \geq \epsilon$). Recall by Lemma 2.7.1 we have with probability $1 - 4e^{-\iota}$:

$$f(\mathbf{x}_T) - f(\mathbf{x}_0) \leq -\frac{\eta}{8}\sum_{i=0}^{T-1}\|\nabla f(\mathbf{x}_i)\|^2 + c\eta\tilde{\sigma}^2(\eta\ell T + \iota) \leq -\eta\left[\frac{T\epsilon^2}{32} - \tilde{\sigma}^2(\eta\ell T + \iota)\right]$$

we note by our choice of η, r, T and picking ι larger than some absolute constant, we have $T\epsilon^2/32 - \tilde{\sigma}^2(\eta\ell T + \iota) \geq T\epsilon^2/64$, and thus $f(\mathbf{x}_T) \leq f(x_0) - T\eta\epsilon^2/64 < f^\star$ which can not be achieved.

Claim 2. We first define the stopping time which are the starting time we can apply Lemma 2.7.3:

$$\begin{aligned} z_1 =& \inf\{\tau \mid \|\nabla f(\mathbf{x}_\tau)\| \leq \epsilon \text{ and } \lambda_{\min}(f(\mathbf{x}_\tau)) \leq -\sqrt{\rho\epsilon}\} \\ z_i =& \inf\{\tau \mid \tau > z_{i-1} + \mathscr{T} \text{ and } \|\nabla f(\mathbf{x}_\tau)\| \leq \epsilon \text{ and } \lambda_{\min}(f(\mathbf{x}_\tau)) \leq -\sqrt{\rho\epsilon}\}, \qquad \forall i > 1 \end{aligned}$$

Clearly, z_i is a stopping time, and is the i-th time in the sequence that we can apply Lemma 2.7.3. We also let M be a stochastic variable where $M = \max\{i | z_i + \mathscr{T} \leq T\}$. Therefore, we can decompose the decrease $f(\mathbf{x}_T) - f(\mathbf{x}_0)$ as follows:

$$\begin{aligned} f(\mathbf{x}_T) - f(\mathbf{x}_0) =& \underbrace{\sum_{i=1}^{M}[f(\mathbf{x}_{z_i+\mathscr{T}}) - f(\mathbf{x}_{z_i})]}_{T_1} \\ & \underbrace{+ [f(\mathbf{x}_T) - f(\mathbf{x}_{z_M})] + [f(\mathbf{x}_{z_1}) - f(\mathbf{x}_0)] + \sum_{i=1}^{M-1}[f(\mathbf{x}_{z_{i+1}}) - f(\mathbf{x}_{z_i+\mathscr{T}})]}_{T_2} \end{aligned}$$

For the first term T_1, by Lemma 2.7.3 and supermartingale concentration inequality, for each fixed $m \leq T$:

$$\mathbb{P}\left(\sum_{i=1}^{m}[f(\mathbf{x}_{z_i+\mathscr{T}}) - f(\mathbf{x}_{z_i})] \leq -(0.9m - c\sqrt{m\cdot\iota})\mathscr{F}\right) \geq 1 - 5d\mathscr{T}^2 T \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}$$

Since random variable $M \leq T/\mathscr{T} \leq T$, by union bound, we know with probability $1 - 5d\mathscr{T}^2T^2 \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}$:

$$T_1 \leq -(0.9M - c\sqrt{M\cdot\iota})\mathscr{F}$$

For the second term, by union bound on Lemma 2.7.1 for all $0 \leq t_1, t_2 \leq T$, with probability $1 - 4T^2e^{-\iota}$:

$$T_2 \leq c\cdot\eta\tilde{\sigma}^2(\eta\ell T + 2M\iota)$$

Algorithm	Iterations	Simplicity
Noisy GD (Ge et al., 2015)	$d^4\text{poly}(\epsilon^{-1})$	single-loop
Normalized GD (Levy, 2016)	$\mathcal{O}(d^3 \cdot \text{poly}(\epsilon^{-1}))$	
PGD (this work)	$\tilde{\mathcal{O}}(\epsilon^{-2})$	
†Perturbed AGD (Jin, Netrapalli, and Jordan, 2017)	$\tilde{\mathcal{O}}(\epsilon^{-1.75})$	
FastCubic (Agarwal et al., 2017)	$\tilde{\mathcal{O}}(\epsilon^{-1.75})$	double-loop
Carmon et al. (2016)	$\tilde{\mathcal{O}}(\epsilon^{-1.75})$	
Carmon and Duchi (2016)	$\tilde{\mathcal{O}}(\epsilon^{-2})$	

Table 2.2: A summary of related work on first-order algorithms to find second-order stationary points in *non-stochastic* setting. This table only highlights the dependences on d and ϵ. † denotes the follow up work.

Therefore, in sum if within T steps, we have more than $T/4$ saddle points, then $M \geq T/4\mathscr{T}$, and with probaility $1 - 10d\mathscr{T}^2T^2 \cdot \log(\mathscr{S}\sqrt{d}/(\eta r))e^{-\iota}$:

$$f(\mathbf{x}_T) - f(\mathbf{x}_0) \leq -(0.9M - c\sqrt{M \cdot \iota})\mathscr{F} + c \cdot \eta\tilde{\sigma}^2(\eta\ell T + 2M\iota) \leq -0.4M\mathscr{F} \leq -0.4T\mathscr{F}/\mathscr{T}$$

This will gives $f(\mathbf{x}_T) \leq f(x_0) - 0.4T\mathscr{F}/\mathscr{T} < f^\star$ which can not be achieved.

Finally, it is not hard to verify, by choose $\iota = c \cdot \log\left(\frac{d\ell\Delta_f\mathfrak{N}}{\rho\epsilon\delta}\right)$ with absolute constant c large enough, we can make both claims hold with probability $1 - \delta$. □

Proof of Theorem 2.4.5

Our proofs for PSGD easily generalize to the mini-batch setting.

Proof of Theorem 2.4.5. The proof is essentially the same as the proof of Theorem 2.4.4. The only difference is that, up to a log factor, mini-batch PSGD reduces variance σ^2 and $\tilde{\ell}^2\|\hat{\mathbf{x}}_\tau\|^2$ in Eq.(2.10) by a factor of m, where m is the size of mini-batch. □

2.8 Tables of Related Work

In Table 2.2 and Table 2.3, we present the full comparison of our results with other related works in both non-stochastic and stochastic settings. See Section 2.1 for the full text descriptions. We note our algorithms are simple variants of standard GD and SGD, which are the simplest among all the algorithms listed in the table.

Algorithm	**Iterations** (*with* Assumption C)	**Iterations** (*no* Assumption C)	**Simplicity**
Noisy GD (Ge et al., 2015)	$d^4\text{poly}(\epsilon^{-1})$	$d^4\text{poly}(\epsilon^{-1})$	single-loop
CNC-SGD (Daneshmand et al., 2018)	$\tilde{\mathcal{O}}(d^4\epsilon^{-5})$	$\tilde{\mathcal{O}}(d^4\epsilon^{-5})$	
PSGD (this work)	$\tilde{\mathcal{O}}(\epsilon^{-4})$	$\tilde{\mathcal{O}}(d\epsilon^{-4})$	
*SGD with averaging (Fang, Lin, and Zhang, 2019)	$\tilde{\mathcal{O}}(\epsilon^{-3.5})$	$\times$	
Natasha 2 (Allen-Zhu, 2018)	$\tilde{\mathcal{O}}(\epsilon^{-3.5})$	$\times$	double-loop
Stochastic Cubic (Tripuraneni et al., 2018)	$\tilde{\mathcal{O}}(\epsilon^{-3.5})$	$\times$	
SPIDER (Fang et al., 2018)	$\tilde{\mathcal{O}}(\epsilon^{-3})$	$\times$	
SRVRC (Zhou and Gu, 2019)	$\tilde{\mathcal{O}}(\epsilon^{-3})$	$\times$	

Table 2.3: A summary of related work on first-order algorithms to find second-order stationary points in *stochastic* setting. This table only highlights the dependences on d and ϵ. * denotes independent work.

2.9 Concentration Inequalities

In this section, we present the concentration inequalities required for this work. Please refer to the technical note (Jin et al., 2019a) for the proofs of Lemmas 2.9.2, 2.9.3, 2.9.5 and 2.9.6.

Recall the definition of norm-subGaussian random vector.

Definition 2.9.1. A random vector $\mathbf{X} \in \mathbb{R}^d$ is *norm-subGaussian* (or nSG(σ)), if there exists σ so that:

$$\mathbb{P}\left(\|\mathbf{X} - \mathbb{E}\mathbf{X}\| \geq t\right) \leq 2e^{-\frac{t^2}{2\sigma^2}}, \qquad \forall t \in \mathbb{R}$$

We first note bounded random vector and subGaussian random vector are two special case of norm-subGaussian random vector.

Lemma 2.9.2. *There exists absolute constant c so that following random vectors are all nSG($c \cdot \sigma$).*

1. *A bounded random vector $\mathbf{X} \in \mathbb{R}^d$ so that $\|\mathbf{X}\| \leq \sigma$.*
2. *A random vector $\mathbf{X} \in \mathbb{R}^d$, where $\mathbf{X} = \xi \mathbf{e}_1$ and random variable $\xi \in \mathbb{R}$ is σ-subGaussian.*

3. A random vector $\mathbf{X} \in \mathbb{R}^d$ that is $(\sigma/\sqrt{d})$-subGaussian.

Second, we have if $\mathbf{X}$ is norm-subGaussian, then its norm square is subExponential, and its component along a single direction is subGaussian.

Lemma 2.9.3. *There is an absolute constant c so that if random vector $\mathbf{X} \in R^d$ is zero-mean nSG(σ), then $\|\mathbf{X}\|^2$ is $c \cdot \sigma^2$-subExponential, and for any fixed unit vector $\mathbf{v} \in \mathbb{S}^{d-1}$, $\langle \mathbf{v}, \mathbf{X} \rangle$ is $c \cdot \sigma$-subGaussian.*

For concentration, we are interested in the properties of norm-subGaussian martingale difference sequences. Concretely, they are sequences satisfying following conditions.

Condition 2.9.4. Let random vectors $\mathbf{X}_1, \ldots, \mathbf{X}_n \in \mathbb{R}^d$, and corresponding filtrations $\mathcal{F}_i = \sigma(\mathbf{X}_1, \ldots, \mathbf{X}_i)$ for $i \in [n]$ satisfy that $\mathbf{X}_i | \mathcal{F}_{i-1}$ is zero-mean nSG(σ_i) with $\sigma_i \in \mathcal{F}_{i-1}$. i.e.,

$$\mathbb{E}[\mathbf{X}_i | \mathcal{F}_{i-1}] = 0, \quad \mathbb{P}\left(\|\mathbf{X}_i\| \geq t | \mathcal{F}_{i-1}\right) \leq 2e^{-\frac{t^2}{2\sigma_i^2}}, \qquad \forall t \in \mathbb{R}, \forall i \in [n].$$

Similar to subGaussian random variables, we can also prove Hoeffding type inequality for norm-subGaussian random vector which is tight up to a $\log(d)$ factor.

Lemma 2.9.5 (Hoeffding type inequality for norm-subGaussian)**.** *There exists an absolute constant c, assume $\mathbf{X}_1, \ldots, \mathbf{X}_n \in \mathbb{R}^d$ satisfy condition 2.9.4 with fixed $\{\sigma_i\}$, then for any $\iota > 0$, with probability at least $1 - 2d \cdot e^{-\iota}$:*

$$\|\sum_{i=1}^n \mathbf{X}_i\| \leq c \cdot \sqrt{\sum_{i=1}^n \sigma_i^2 \cdot \iota}$$

In case of $\{\sigma_i\}$ also being random, we have the following.

Lemma 2.9.6. *There exists an absolute constant c, assume $\mathbf{X}_1, \ldots, \mathbf{X}_n \in \mathbb{R}^d$ satisfy condition 2.9.4, then for any $\iota > 0$, and $B > b > 0$, with probability at least $1 - 2d \log(B/b) \cdot e^{-\iota}$:*

$$\sum_{i=1}^n \sigma_i^2 \geq B \quad \text{or} \quad \|\sum_{i=1}^n \mathbf{X}_i\| \leq c \cdot \sqrt{\max\{\sum_{i=1}^n \sigma_i^2, b\} \cdot \iota}$$

Finally, we can also provide concentration inequalities for the sum of norm square of norm-subGaussian random vectors, and for the sum of inner product of norm-subGaussian random vectors with another set of random vectors.

Lemma 2.9.7. *Assume $\mathbf{X}_1, \ldots, \mathbf{X}_n \in \mathbb{R}^d$ satisfy Condition 2.9.4 with fixed $\sigma_1 = \ldots = \sigma_n = \sigma$, then there exists absolute constant c, for any $\iota > 0$, with probability at least $1 - e^{-\iota}$:*

$$\sum_{i=1}^n \|\mathbf{X}_i\|^2 \leq c \cdot \sigma^2 (n + \iota)$$

Proof. Note there exists an absolute contant c such that $\mathbb{E}[\|\mathbf{X}_i\|^2|\mathcal{F}_{i-1}] \le c\cdot\sigma^2$, and $\|\mathbf{X}_i\|^2|\mathcal{F}_{i-1}$ is $c\cdot\sigma^2$-subExponential. This lemma directly follows from standard Bernstein type concentration inequalities for subExponential random variables. □

Lemma 2.9.8. *There exists absolute constant c, assume $\mathbf{X}_1,\ldots,\mathbf{X}_n \in \mathbb{R}^d$ satisfy Condition 2.9.4 and random vectors $\{\mathbf{u}_i\}$ satisfy $\mathbf{u}_i \in \mathcal{F}_{i-1}$ for all $i \in [n]$, then for any $\iota > 0$, $\lambda > 0$, with probability at least $1 - e^{-\iota}$:*

$$\sum_i \langle \mathbf{u}_i, \mathbf{X}_i\rangle \le c\cdot\lambda\sum_i \|\mathbf{u}_i\|^2\sigma_i^2 + \frac{1}{\lambda}\cdot\iota$$

Proof. For any $i \in [n]$ and fixed $\lambda > 0$, since $\mathbf{u}_i \in \mathcal{F}_{i-1}$, according to Lemma 2.9.3, there exists constant c so that $\langle \mathbf{u}_i, \mathbf{X}_i\rangle|\mathcal{F}_{i-1}$ is $c\cdot\|\mathbf{u}_i\|\sigma_i$-subGaussian. Thus:

$$\mathbb{E}[e^{\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle}|\mathcal{F}_{i-1}] \le e^{c\cdot\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2}$$

Therefore, consider following quantity:

$$\begin{aligned}
\mathbb{E}e^{\sum_{i=1}^t(\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle - c\cdot\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2)} =& \mathbb{E}\left[e^{\sum_{i=1}^{t-1}\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle - c\cdot\sum_{i=1}^t\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2}\cdot\mathbb{E}\left(e^{\lambda\langle \mathbf{u}_t,\mathbf{X}_t\rangle}|\mathcal{F}_{t-1}\right)\right]\\
\le& \mathbb{E}\left[e^{\sum_{i=1}^{t-1}\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle - c\cdot\sum_{i=1}^t\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2}\cdot e^{c\cdot\lambda^2\|\mathbf{u}_t\|^2\sigma_t^2}\right]\\
=& \mathbb{E}e^{\sum_{i=1}^{t-1}(\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle - c\cdot\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2)} \le 1
\end{aligned}$$

Finally, by Markov's inequality, for any $t > 0$:

$$\begin{aligned}
\mathbb{P}\left(\sum_{i=1}^t(\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle - c\cdot\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2) \ge t\right) \le& \mathbb{P}\left(e^{\sum_{i=1}^t(\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle - c\cdot\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2)} \ge e^t\right)\\
\le& e^{-t}\mathbb{E}e^{\sum_{i=1}^t(\lambda\langle \mathbf{u}_i,\mathbf{X}_i\rangle - c\cdot\lambda^2\|\mathbf{u}_i\|^2\sigma_i^2)} \le e^{-t}
\end{aligned}$$

This finishes the proof. □

Chapter 3

Escaping Saddle Points Faster using Momentum

Nesterov's accelerated gradient descent (AGD), an instance of the general family of "momentum methods," provably achieves faster convergence rate than gradient descent (GD) in the convex setting. However, whether these methods are superior to GD in the nonconvex setting remains open. This work studies a simple variant of AGD, and shows that it escapes saddle points and finds a second-order stationary point in $\tilde{O}(1/\epsilon^{7/4})$ iterations, faster than the $\tilde{O}(1/\epsilon^2)$ iterations required by GD. To the best of our knowledge, this is the first Hessian-free algorithm to find a second-order stationary point faster than GD, and also the first single-loop algorithm with a faster rate than GD even in the setting of finding a first-order stationary point. Our analysis is based on two key ideas: (1) the use of a simple Hamiltonian function, inspired by a continuous-time perspective, which AGD monotonically decreases per step even for nonconvex functions, and (2) a novel framework called *improve or localize*, which is useful for tracking the long-term behavior of gradient-based optimization algorithms. We believe that these techniques may deepen our understanding of both acceleration algorithms and nonconvex optimization.

3.1 Introduction

Nonconvex optimization problems are ubiquitous in modern machine learning. While it is NP-hard to find global minima of a nonconvex function in the worst case, in the setting of machine learning it has proved useful to consider a less stringent notion of success, namely that of convergence to a *first-order stationary point* (where $\nabla f(\mathbf{x}) = 0$). Gradient descent (GD), a simple and fundamental optimization algorithm that has proved its value in large-scale machine learning, is known to find an ϵ-first-order stationary point (where $\|\nabla f(\mathbf{x})\| \le \epsilon$) in $O(1/\epsilon^2)$ iterations (Nesterov, 1998), and this rate is sharp (Cartis, Gould, and Toint, 2010). Such results, however, do not seem to address the practical success of gradient descent; first-order stationarity includes local minima, saddle points or even local maxima, and a mere

guarantee of convergence to such points seems unsatisfying. Indeed, architectures such as deep neural networks induce optimization surfaces that can be teeming with such highly suboptimal saddle points (Dauphin et al., 2014). It is important to study to what extent gradient descent avoids such points, particular in the high-dimensional setting in which the directions of escape from saddle points may be few.

This work focuses on convergence to a *second-order stationary point* (where $\nabla f(\mathbf{x}) = 0$ and $\nabla^2 f(\mathbf{x}) \succeq 0$). Second-order stationarity rules out many common types of saddle points (*strict* saddle points where $\lambda_{\min}(\nabla^2 f(\mathbf{x})) < 0$), allowing only local minima and higher-order saddle points. A significant body of recent work, some theoretical and some empirical, shows that for a large class of well-studied machine learning problems, neither higher-order saddle points nor spurious local minima exist. That is, *all second-order stationary points are (approximate) global minima* for these problems. Choromanska et al. (2014) and Kawaguchi (2016) present such a result for learning multi-layer neural networks, Bandeira, Boumal, and Voroninski (2016) and Mei et al. (2017) for synchronization and MaxCut, Boumal, Voroninski, and Bandeira (2016) for smooth semidefinite programs, Bhojanapalli, Neyshabur, and Srebro (2016) for matrix sensing, Ge, Lee, and Ma (2016) for matrix completion, and Ge, Jin, and Zheng (2017) for robust PCA. These results strongly motivate the quest for *efficient algorithms* to find second-order stationary points.

Hessian-based algorithms can explicitly compute curvatures and thereby avoid saddle points (e.g., (Nesterov and Polyak, 2006; Curtis, Robinson, and Samadi, 2014)), but these algorithms are computationally infeasible in the high-dimensional regime. GD, by contrast, is known to get stuck at strict saddle points Nesterov, 1998, Section 1.2.3. Recent work has reconciled this conundrum in favor of GD; (Jin et al., 2017), building on earlier work of (Ge et al., 2015), show that a perturbed version of GD converges to an ϵ-relaxed version of a second-order stationary point (see Definition 2.2.9) in $\tilde{O}(1/\epsilon^2)$ iterations. That is, perturbed GD in fact finds second-order stationary points as fast as standard GD finds first-order stationary point, up to logarithmic factors in dimension.

On the other hand, GD is known to be suboptimal in the convex case. In a celebrated work, Nesterov (1983) showed that an accelerated version of gradient descent (AGD) finds an ϵ-suboptimal point (see Section 3.2) in $O(1/\sqrt{\epsilon})$ steps, while gradient descent takes $O(1/\epsilon)$ steps. The basic idea of acceleration has been used to design faster algorithms for a range of other convex optimization problems (Beck and Teboulle, 2009; Nesterov, 2012; Lee and Sidford, 2013; Shalev-Shwartz and Zhang, 2014). We will refer to this general family as "momentum-based methods."

Such results have focused on the convex setting. It is open as to whether momentum-based methods yield faster rates in the *nonconvex setting*, specifically when we consider the convergence criterion of second-order stationarity. We are thus led to ask the following question:

Do momentum-based methods yield faster convergence than GD in the presence of saddle points?

Algorithm 5 Nesterov's Accelerated Gradient Descent $(\mathbf{x}_0, \eta, \theta)$

$\mathbf{v}_0 \leftarrow 0$
for $t = 0, 1, \ldots,$ **do**
  $\mathbf{y}_t \leftarrow \mathbf{x}_t + (1-\theta)\mathbf{v}_t$
  $\mathbf{x}_{t+1} \leftarrow \mathbf{y}_t - \eta \nabla f(\mathbf{y}_t)$
  $\mathbf{v}_{t+1} \leftarrow \mathbf{x}_{t+1} - \mathbf{x}_t$

Algorithm 6 Perturbed Accelerated Gradient Descent $(\mathbf{x}_0, \eta, \theta, \gamma, s, r, \mathscr{T})$

1: $\mathbf{v}_0 \leftarrow 0$
2: **for** $t = 0, 1, \ldots,$ **do**
3: **if** $\|\nabla f(\mathbf{x}_t)\| \le \epsilon$ and *no perturbation in last* $\mathscr{T}$ *steps* **then**
4: $\mathbf{x}_t \leftarrow \mathbf{x}_t + \xi_t \qquad \xi_t \sim \mathrm{Unif}\,(\mathbb{B}_0(r))$
5: $\mathbf{y}_t \leftarrow \mathbf{x}_t + (1-\theta)\mathbf{v}_t$
6: $\mathbf{x}_{t+1} \leftarrow \mathbf{y}_t - \eta \nabla f(\mathbf{y}_t)$
7: $\mathbf{v}_{t+1} \leftarrow \mathbf{x}_{t+1} - \mathbf{x}_t$
8: **if** $f(\mathbf{x}_t) \le f(\mathbf{y}_t) + \langle \nabla f(\mathbf{y}_t), \mathbf{x}_t - \mathbf{y}_t \rangle - \frac{\gamma}{2}\|\mathbf{x}_t - \mathbf{y}_t\|^2$ **then**
9: $(\mathbf{x}_{t+1}, \mathbf{v}_{t+1}) \leftarrow$ Negative-Curvature-Exploitation$(\mathbf{x}_t, \mathbf{v}_t, s)$

This work answers this question in the affirmative. We present a simple momentum-based algorithm (PAGD for "perturbed AGD") that finds an ϵ-second order stationary point in $\tilde{O}(1/\epsilon^{7/4})$ iterations, faster than the $\tilde{O}(1/\epsilon^2)$ iterations required by GD. The pseudocode of our algorithm is presented in Algorithm 6.[1] PAGD adds two algorithmic features to AGD (Algorithm 5):

- Perturbation (Lines 3-4): when the gradient is small, we add a small perturbation sampled uniformly from a d-dimensional ball with radius r. The homogeneous nature of this perturbation mitigates our lack of knowledge of the curvature tensor at or near saddle points.
- Negative Curvature Exploitation (NCE, Lines 8-9; pseudocode in Algorithm 7): when the function becomes "too nonconvex" along $\mathbf{y}_t$ to $\mathbf{x}_t$, we reset the momentum and decide whether to exploit negative curvature depending on the magnitude of the current momentum $\mathbf{v}_t$.

We note that both components are straightforward to implement and increase computation by a constant factor. The perturbation idea follows from (Ge et al., 2015) and (Jin et al., 2017), while NCE is inspired by (Carmon et al., 2017a). To the best of our knowledge, PAGD is the first Hessian-free algorithm to find a second-order stationary point in $\tilde{O}(1/\epsilon^{7/4})$ steps. Note also that PAGD is a "single-loop algorithm," meaning that it does not require

[1] See Section 3.3 for values of various parameters.

Guarantees	Oracle	Algorithm	Iterations	Simplicity
First-order Stationary Point	Gradient	GD (Nesterov, 1998)	$O(1/\epsilon^2)$	Single-loop
		AGD (Ghadimi and Lan, 2016)	$O(1/\epsilon^2)$	Single-loop
		(Carmon et al., 2017a)	$\widetilde{\mathcal{O}}(1/\epsilon^{7/4})$	Nested-loop
Second-order Stationary Point	Hessian -vector	Carmon et al. (2016)	$\widetilde{O}(1/\epsilon^{7/4})$	Nested-loop
		Agarwal et al. (2017)	$\widetilde{O}(1/\epsilon^{7/4})$	Nested-loop
	Gradient	Noisy GD (Ge et al., 2015)	$O(\text{poly}(d/\epsilon))$	Single-loop
		Perturbed GD (Jin et al., 2017)	$\widetilde{O}(1/\epsilon^2)$	Single-loop
		Perturbed AGD [This Work]	$\widetilde{O}(1/\epsilon^{7/4})$	Single-loop

Table 3.1: Complexity of finding stationary points. $\widetilde{O}(\cdot)$ ignores polylog factors in d and ϵ.

an inner loop of optimization of a surrogate function. It is the first single-loop algorithm to achieve a $\tilde{O}(1/\epsilon^{7/4})$ rate even in the setting of finding a first-order stationary point.

Related Work

In this section, we review related work from the perspective of both nonconvex optimization and momentum/acceleration. For clarity of presentation, when discussing rates, we focus on the dependence on the accuracy ϵ and the dimension d while assuming all other problem parameters are constant. Table 3.1 presents a comparison of the current work with previous work.

Convergence to first-order stationary points: Traditional analyses in this case assume only Lipschitz gradients (see Definition 2.2.1). (Nesterov, 1998) shows that GD finds an ϵ-first-order stationary point in $O(1/\epsilon^2)$ steps. (Ghadimi and Lan, 2016) guarantee that AGD also converges in $\widetilde{\mathcal{O}}(1/\epsilon^2)$ steps. Under the additional assumption of Lipschitz Hessians (see Definition 2.2.2), Carmon et al. (2017a) develop a new algorithm that converges in $O(1/\epsilon^{7/4})$ steps. Their algorithm is a nested-loop algorithm, where the outer loop adds a proximal term to reduce the nonconvex problem to a convex subproblem. A key novelty in their algorithm is the idea of "negative curvature exploitation," which inspired a similar step in our algorithm. In addition to the qualitative and quantitative differences between (Carmon et al., 2017a) and the current work, as summarized in Table 3.1, we note that while (Carmon et al., 2017a) analyze AGD applied to convex subproblems, we analyze AGD applied directly to nonconvex functions through a novel Hamiltonian framework.

Convergence to second-order stationary points: All results in this setting assume Lipschitz conditions for both the gradient and Hessian. Classical approaches, such as cubic regularization (Nesterov and Polyak, 2006) and trust region algorithms (Curtis, Robinson,

and Samadi, 2014), require access to Hessians, and are known to find ϵ-second-order stationary points in $O(1/\epsilon^{1.5})$ steps. However, the requirement of these algorithms to form the Hessian makes them infeasible for high-dimensional problems. A second set of algorithms utilize only Hessian-vector products instead of the explicit Hessian; in many applications such products can be computed efficiently. Rates of $\widetilde{O}(1/\epsilon^{7/4})$ have been established for such algorithms (Carmon et al., 2016; Agarwal et al., 2017; Royer and Wright, 2017). Finally, in the realm of purely gradient-based algorithms, Ge et al. (2015) present the first polynomial guarantees for a perturbed version of GD, and Jin et al. (2017) sharpen it to $\widetilde{O}(1/\epsilon^2)$. For the special case of quadratic functions, (O'Neill and Wright, 2017) analyze the behavior of AGD around critical points and show that it escapes saddle points faster than GD. We note that the current work is the first achieving a rate of $\widetilde{O}(1/\epsilon^{7/4})$ for general nonconvex functions.

Acceleration: There is also a rich literature that aims to understand momentum methods; e.g., Allen-Zhu and Orecchia (2014) view AGD as a linear coupling of GD and mirror descent, Su, Boyd, and Candes (2016) and Wibisono, Wilson, and Jordan (2016) view AGD as a second-order differential equation, and Bubeck, Lee, and Singh (2015) view AGD from a geometric perspective. Most of this work is tailored to the convex setting, and it is unclear and nontrivial to generalize the results to a nonconvex setting. There are also several works that study AGD with relaxed versions of convexity—see Necoara, Nesterov, and Glineur (2015) and Li and Lin (2017) and references therein for overviews of these results.

Main Techniques

Our results rely on the following three key ideas. To the best of our knowledge, the first two are novel, while the third one was delineated in Jin et al. (2017).

Hamiltonian: A major challenge in analyzing momentum-based algorithms is that the objective function does not decrease monotonically as is the case for GD. To overcome this in the convex setting, several Lyapunov functions have been proposed (Wilson, Recht, and Jordan, 2016). However these Lyapunov functions involve the global minimum $\mathbf{x}^\star$, which cannot be computed by the algorithm, and is thus of limited value in the nonconvex setting. A key technical contribution of this work is the design of a function which is both computable and tracks the progress of AGD. The function takes the form of a Hamiltonian:

$$E_t := f(\mathbf{x}_t) + \frac{1}{2\eta}\|\mathbf{v}_t\|^2; \tag{3.1}$$

i.e., a sum of potential energy and kinetic energy terms. It is monotonically decreasing in the continuous-time setting. This is *not* the case in general in the discrete-time setting, a fact which requires us to incorporate the NCE step.

Improve or localize: Another key technical contribution of this work is in formalizing a simple but powerful framework for analyzing nonconvex optimization algorithms. This framework requires us to show that for a given algorithm, *either the algorithm makes significant progress or the iterates do not move much.* We call this the *improve-or-localize* phe-

nomenon. For instance, when progress is measured by function value, it is easy to show that for GD, with proper choice of learning rate, we have:

$$\frac{1}{2\eta}\sum_{\tau=0}^{t-1}\|\mathbf{x}_{\tau+1}-\mathbf{x}_\tau\|^2 \le f(\mathbf{x}_0)-f(\mathbf{x}_t).$$

For AGD, a similar lemma can be shown by replacing the objective function with the Hamiltonian (see Lemma 3.4.1). Once this phenomenon is established, we can conclude that if an algorithm does not make much progress, it is localized to a small ball, and we can then approximate the objective function by either a linear or a quadratic function (depending on smoothness assumptions) in this small local region. Moreover, an upper bound on $\sum_{\tau=0}^{t-1}\|\mathbf{x}_{\tau+1}-\mathbf{x}_\tau\|^2$ lets us conclude that iterates do not oscillate much in this local region (oscillation is a unique phenomenon of momentum algorithms as can be seen even in the convex setting). This gives us better control of approximation error.

Coupling sequences for escaping saddle points: When an algorithm arrives in the neighborhood of a strict saddle point, where $\lambda_{\min}(\nabla^2 f(\mathbf{x})) < 0$, all we know is that there exists a direction of escape (the direction of the minimum eigenvector of $\nabla^2 f(\mathbf{x})$); denote it by $\mathbf{e}_{\text{esc}}$. To avoid such points, the algorithm randomly perturbs the current iterate uniformly in a small ball, and runs AGD starting from this point $\tilde{\mathbf{x}}_0$. As in (Jin et al., 2017), we can divide this ball into a "stuck region," $\mathcal{X}_{\text{stuck}}$, starting from which AGD does not escape the saddle quickly, and its complement from which AGD escapes quickly. In order to show quick escape from a saddle point, we must show that the volume of $\mathcal{X}_{\text{stuck}}$ is very small compared to that of the ball. Though $\mathcal{X}_{\text{stuck}}$ may be without an analytical form, one can control the rate of escape by studying two AGD sequences that start from two realizations of perturbation, $\tilde{\mathbf{x}}_0$ and $\tilde{\mathbf{x}}_0'$, which are separated along $\mathbf{e}_{\text{esc}}$ by a small distance r_0. In this case, at least one of the sequences escapes the saddle point quickly, which proves that the width of $\mathcal{X}_{\text{stuck}}$ along $\mathbf{e}_{\text{esc}}$ can not be greater than r_0, and hence $\mathcal{X}_{\text{stuck}}$ has small volume.

3.2 Preliminaries

In this section, we will review some well-known results on GD and AGD in the strongly convex setting, and existing results on convergence of GD to second-order stationary points.

Notation

Bold upper-case letters $(\mathbf{A}, \mathbf{B})$ denote matrices and bold lower-case letters $(\mathbf{x}, \mathbf{y})$ denote vectors. For vectors $\|\cdot\|$ denotes the ℓ_2-norm. For matrices, $\|\cdot\|$ denotes the spectral norm and $\lambda_{\min}(\cdot)$ denotes the minimum eigenvalue. For $f : \mathbb{R}^d \to \mathbb{R}$, $\nabla f(\cdot)$ and $\nabla^2 f(\cdot)$ denote its gradient and Hessian respectively, and $f^\star$ denotes its global minimum. We use $O(\cdot), \Theta(\cdot), \Omega(\cdot)$ to hide absolute constants, and $\tilde{O}(\cdot), \tilde{\Theta}(\cdot), \tilde{\Omega}(\cdot)$ to hide absolute constants and polylog factors for all problem parameters.

Algorithm 7 Negative Curvature Exploitation($\mathbf{x}_t, \mathbf{v}_t, s$)

if $\|\mathbf{v}_t\| \geq s$ **then**
  $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t$;
else
  $\delta = s \cdot \mathbf{v}_t / \|\mathbf{v}_t\|$
  $\mathbf{x}_{t+1} \leftarrow \text{argmin}_{\mathbf{x} \in \{\mathbf{x}_t + \delta, \mathbf{x}_t - \delta\}} f(\mathbf{x})$
return $(\mathbf{x}_{t+1}, 0)$

Convex Setting

To minimize a function $f(\cdot)$, GD performs the following sequence of steps:

$$\mathbf{x}_{t+1} = \mathbf{x}_t - \eta \nabla f(\mathbf{x}_t).$$

The suboptimality of GD and the improvement achieved by AGD can be clearly illustrated for the case of smooth and strongly convex functions.

Definition 3.2.1. A twice-differentiable function $f(\cdot)$ is **α-strongly convex** if $\lambda_{\min}(\nabla^2 f(\mathbf{x})) \geq \alpha$, $\forall\, \mathbf{x}$.

Let $f^* := \min_{\mathbf{y}} f(\mathbf{y})$. A point $\mathbf{x}$ is said to be **ϵ-suboptimal** if $f(\mathbf{x}) \leq f^* + \epsilon$. The following theorem gives the convergence rate of GD and AGD for smooth and strongly convex functions.

Theorem 3.2.2 ((Nesterov, 2004))**.** *Assume that the function $f(\cdot)$ is ℓ-gradient Lipschitz and α-strongly convex. Then, for any $\epsilon > 0$, the iteration complexities to find an ϵ-suboptimal point are as follows:*

- *GD with $\eta = 1/\ell$:* $\quad O((\ell/\alpha) \cdot \log((f(\mathbf{x}_0) - f^*)/\epsilon))$
- *AGD (Algorithm 5) with $\eta = 1/\ell$ and $\theta = \sqrt{\alpha/\ell}$:* $\quad O(\sqrt{\ell/\alpha} \cdot \log((f(\mathbf{x}_0) - f^*)/\epsilon))$.

The number of iterations of GD depends linearly on the ratio ℓ/α, which is called the condition number of $f(\cdot)$ since $\alpha \mathbf{I} \preceq \nabla^2 f(\mathbf{x}) \preceq \ell \mathbf{I}$. Clearly $\ell \geq \alpha$ and hence condition number is always at least one. Denoting the condition number by κ, we highlight two important aspects of AGD: (1) the momentum parameter satisfies $\theta = 1/\sqrt{\kappa}$ and (2) AGD improves upon GD by a factor of $\sqrt{\kappa}$.

3.3 Main Result

In this section, we present our algorithm and main result. As mentioned in Section 3.1, the algorithm we propose is essentially AGD with two key differences (see Algorithm 6): perturbation and negative curvature exploitation (NCE). A perturbation is added when the

gradient is small (to escape saddle points), and no more frequently than once in $\mathscr{T}$ steps. The perturbation ξ_t is sampled uniformly from a d-dimensional ball with radius r. The specific choices of gap and uniform distribution are for technical convenience (they are sufficient for our theoretical result but not necessary).

NCE (Algorithm 7) is explicitly designed to guarantee decrease of the Hamiltonian (3.1). When it is triggered, i.e., when

$$f(\mathbf{x}_t) \le f(\mathbf{y}_t) + \langle \nabla f(\mathbf{y}_t), \mathbf{x}_t - \mathbf{y}_t \rangle - \frac{\gamma}{2}\|\mathbf{x}_t - \mathbf{y}_t\|^2 \tag{3.2}$$

the function has a large negative curvature between the current iterates $\mathbf{x}_t$ and $\mathbf{y}_t$. In this case, if the momentum $\mathbf{v}_t$ is small, then $\mathbf{y}_t$ and $\mathbf{x}_t$ are close, so the large negative curvature also carries over to the Hessian at $\mathbf{x}_t$ due to the Lipschitz property. Assaying two points along $\pm(\mathbf{y}_t - \mathbf{x}_t)$ around $\mathbf{x}_t$ gives one point that is negatively aligned with $\nabla f(\mathbf{x}_t)$ and yields a decreasing function value and Hamiltonian. If the momentum $\mathbf{v}_t$ is large, negative curvature can no longer be exploited, but fortunately resetting the momentum to zero kills the second term in (3.1), significantly decreasing the Hamiltonian.

Setting of hyperparameters: Let ϵ be the target accuracy for a second-order stationary point, let ℓ and ρ be gradient/Hessian-Lipschitz parameters, and let c, χ be absolute constant and log factor to be specified later. Let $\kappa := \ell/\sqrt{\rho\epsilon}$, and set

$$\eta = \frac{1}{4\ell}, \quad \theta = \frac{1}{4\sqrt{\kappa}}, \quad \gamma = \frac{\theta^2}{\eta}, \quad s = \frac{\gamma}{4\rho}, \quad \mathscr{T} = \sqrt{\kappa}\cdot\chi c, \quad r = \eta\epsilon\cdot\chi^{-5}c^{-8}. \tag{3.3}$$

The following theorem is the main result of this work.

Theorem 3.3.1. *Assume that the function $f(\cdot)$ is ℓ-smooth and ρ-Hessian Lipschitz. There exists an absolute constant $c_{\max}$ such that for any $\delta > 0$, $\epsilon \le \frac{\ell^2}{\rho}$, $\Delta_f \ge f(\mathbf{x}_0) - f^\star$, if $\chi = \max\{1, \log\frac{d\ell\Delta_f}{\rho\epsilon\delta}\}$, $c \ge c_{\max}$ and such that if we run PAGD (Algorithm 6) with choice of parameters according to (3.3), then with probability at least $1-\delta$, one of the iterates $\mathbf{x}_t$ will be an ϵ-second order stationary point in the following number of iterations:*

$$O\left(\frac{\ell^{1/2}\rho^{1/4}(f(\mathbf{x}_0) - f^*)}{\epsilon^{7/4}}\log^6\left(\frac{d\ell\Delta_f}{\rho\epsilon\delta}\right)\right)$$

Theorem 3.3.1 says that when PAGD is run for the designated number of steps (which is poly-logarithmic in dimension), at least one of the iterates is an ϵ-second-order stationary point. We focus on the case of small ϵ (i.e., $\epsilon \le \ell^2/\rho$) so that the Hessian requirement for the ϵ-second-order stationary point ($\lambda_{\min}(\nabla^2 f(\mathbf{x})) \ge -\sqrt{\rho\epsilon}$) is nontrivial. Note that $\|\nabla^2 f(\mathbf{x})\| \le \ell$ implies $\kappa = \ell/\sqrt{\rho\epsilon}$, which can be viewed as a condition number, akin to that in convex setting. Comparing Theorem 3.3.1 with Theorem 2.6.1, PAGD, with a momentum parameter $\theta = \Theta(1/\sqrt{\kappa})$, achieves $\tilde{\Theta}(\sqrt{\kappa})$ better iteration complexity compared to PGD.

Output ϵ-second order stationary point: Although Theorem 3.3.1 only guarantees that one of the iterates is an ϵ-second order stationary point, it is straightforward to identify

one of them by adding a proper termination condition: once the gradient is small and satisfies the pre-condition to add a perturbation, we can keep track of the point $\mathbf{x}_{t_0}$ prior to adding perturbation, and compare the Hamiltonian at t_0 with the one $\mathscr{T}$ steps after. If the Hamiltonian decreases by $\mathscr{F} = \tilde{\Theta}(\sqrt{\epsilon^3/\rho})$, then the algorithm has made progress, otherwise $\mathbf{x}_{t_0}$ is an ϵ-second-order stationary point according to Lemma 3.4.6. Doing so will add a hyperparameter (threshold $\mathscr{F}$) but does not increase complexity.

3.4 Overview of Analysis

In this section, we will present an overview of the proof of Theorem 3.3.1. Section 3.4 presents the Hamiltonian for AGD and its key property of monotonic decrease. This leads to Section 3.4 where the *improve-or-localize* lemma is stated, as well as the main intuition behind acceleration. Section 3.4 demonstrates how to apply these tools to prove Theorem 3.3.1. Complete details can be found in the appendix.

Hamiltonian

While GD guarantees decrease of function value in every step (even for nonconvex problems), the biggest stumbling block to analyzing AGD is that it is less clear how to keep track of "progress." Known Lyapunov functions for AGD (Wilson, Recht, and Jordan, 2016) are restricted to the convex setting and furthermore are not computable by the algorithm (as they depend on $\mathbf{x}^\star$).

To deepen the understanding of AGD in a nonconvex setting, we inspect it from a dynamical systems perspective, where we fix the ratio $\tilde{\theta} = \theta/\sqrt{\eta}$ to be a constant, while letting $\eta \to 0$. This leads to an ODE which is the continuous limit of AGD (Su, Boyd, and Candes, 2016):

$$\ddot{\mathbf{x}} + \tilde{\theta}\dot{\mathbf{x}} + \nabla f(\mathbf{x}) = 0, \tag{3.4}$$

where $\ddot{\mathbf{x}}$ and $\dot{\mathbf{x}}$ are derivatives with respect to time t. This equation is a second-order dynamical equation with *dissipative forces* $-\tilde{\theta}\dot{\mathbf{x}}$. Integrating both sides, we obtain:

$$f(\mathbf{x}(t_2)) + \frac{1}{2}\dot{\mathbf{x}}(t_2)^2 = f(\mathbf{x}(t_1)) + \frac{1}{2}\dot{\mathbf{x}}(t_1)^2 - \tilde{\theta}\int_{t_1}^{t_2} \dot{\mathbf{x}}(t)^2 \mathrm{d}t. \tag{3.5}$$

Using physical language, $f(\mathbf{x})$ is a *potential energy* while $\dot{\mathbf{x}}^2/2$ is a *kinetic energy*, and the sum is a *Hamiltonian*. The integral shows that the Hamiltonian decreases monotonically with time t, and the decrease is given by the *dissipation* term $\tilde{\theta}\int_{t_1}^{t_2} \dot{\mathbf{x}}(t)^2 \mathrm{d}t$. Note that (3.5) holds regardless of the convexity of $f(\cdot)$. This monotonic decrease of the Hamiltonian can in fact be extended to the discretized version of AGD when the function is convex, or mildly nonconvex:

Lemma 3.4.1 (Hamiltonian decreases monotonically)**.** *Assume that the function $f(\cdot)$ is ℓ-smooth, the learning rate $\eta \le \frac{1}{2\ell}$, and $\theta \in [2\eta\gamma, \frac{1}{2}]$ in AGD (Algorithm 5). Then, for every iteration t where (3.2) does not hold, we have:*

$$f(\mathbf{x}_{t+1}) + \frac{1}{2\eta}\|\mathbf{v}_{t+1}\|^2 \le f(\mathbf{x}_t) + \frac{1}{2\eta}\|\mathbf{v}_t\|^2 - \frac{\theta}{2\eta}\|\mathbf{v}_t\|^2 - \frac{\eta}{4}\|\nabla f(\mathbf{y}_t)\|^2. \tag{3.6}$$

Denote the discrete Hamiltonian as $E_t := f(\mathbf{x}_t) + \frac{1}{2\eta}\|\mathbf{v}_t\|^2$, and note that in AGD, $\mathbf{v}_t = \mathbf{x}_t - \mathbf{x}_{t-1}$. Lemma 3.4.1 tolerates nonconvexity with curvature at most $\gamma = \Theta(\theta/\eta)$. Unfortunately, when the function becomes too nonconvex in certain regions (so that (3.2) holds), the analogy between the continuous and discretized versions breaks and (3.6) no longer holds. In fact, standard AGD can even increase the Hamiltonian in this regime (see Appendix 3.6 for more details). This motivates us to modify the algorithm by adding the NCE step, which addresses this issue. We have the following result:

Lemma 3.4.2. *Assume that $f(\cdot)$ is ℓ-smooth and ρ-Hessian Lipschitz. For every iteration t of Algorithm 6 where (3.2) holds (thus running NCE), we have:*

$$E_{t+1} \le E_t - \min\{\frac{s^2}{2\eta}, \frac{1}{2}(\gamma - 2\rho s)s^2\}.$$

Lemmas 3.4.1 and 3.4.2 jointly assert that the Hamiltonian decreases monotonically in all situations, and are the main tools in the proof of Theorem 3.3.1. They not only give us a way of tracking progress, but also quantitatively measure the amount of progress.

Improve or Localize

One significant challenge in the analysis of gradient-based algorithms for nonconvex optimation is that many phenomena—for instance the accumulation of momentum and the escape from saddle points via perturbation—are multiple-step behaviors; they do not happen in each step. We address this issue by developing a general technique for analyzing the long-term behavior of such algorithms.

In our case, to track the long-term behavior of AGD, one key observation from Lemma 3.4.1 is that the amount of progress actually relates to movement of the iterates, which leads to the following *improve-or-localize* lemma:

Corollary 3.4.3 (Improve or localize)**.** *Under the same setting as in Lemma 3.4.1, if (3.2) does not hold for all steps in $[t, t+T]$, we have:*

$$\sum_{\tau=t+1}^{t+T} \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|^2 \le \frac{2\eta}{\theta}(E_t - E_{t+T}).$$

Corollary 3.4.3 says that the algorithm either makes progress in terms of the Hamiltonian, or the iterates do not move much. In the second case, Corollary 3.4.3 allows us to approximate the dynamics of $\{\mathbf{x}_\tau\}_{\tau=t}^{t+T}$ with a *quadratic approximation* of $f(\cdot)$.

The acceleration phenomenon is rooted in and can be seen clearly for a quadratic, where the function can be decomposed into eigen-directions. Consider an eigen-direction with eigenvalue λ, and linear term g (i.e., in this direction $f(x) = \frac{\lambda}{2}x^2 + gx$). The GD update becomes $x_{\tau+1} = (1-\eta\lambda)x_\tau - \eta g$, with $\mu_{\mathrm{GD}}(\lambda) := 1-\eta\lambda$ determining the rate of GD. The update of AGD is $(x_{\tau+1}, x_\tau) = (x_\tau, x_{\tau-1})\mathbf{A}^\top - (\eta g, 0)$ with matrix $\mathbf{A}$ defined as follows:

$$\mathbf{A} := \begin{pmatrix} (2-\theta)(1-\eta\lambda) & -(1-\theta)(1-\eta\lambda) \\ 1 & 0 \end{pmatrix}.$$

The rate of AGD is determined by largest eigenvalue of matrix $\mathbf{A}$, which is denoted by $\mu_{\mathrm{AGD}}(\lambda)$. Recall the choice of parameter (3.3), and divide the eigen-directions into the following three categories.

- **Strongly convex directions** $\lambda \in [\sqrt{\rho\epsilon}, \ell]$**:** the slowest case is $\lambda = \sqrt{\rho\epsilon}$, where $\mu_{\mathrm{GD}}(\lambda) = 1 - \Theta(1/\kappa)$ while $\mu_{\mathrm{AGD}}(\lambda) = 1 - \Theta(1/\sqrt{\kappa})$, which results in AGD converging faster than GD.

- **Flat directions** $\lambda \in [-\sqrt{\rho\epsilon}, \sqrt{\rho\epsilon}]$**:** the representative case is $\lambda = 0$ where AGD update becomes $x_{\tau+1} - x_\tau = (1-\theta)(x_\tau - x_{\tau-1}) - \eta g$. For $\tau \le 1/\theta$, we have $|x_{t+\tau} - x_t| = \Theta(\tau)$ for GD while $|x_{t+\tau} - x_t| = \Theta(\tau^2)$ for AGD, which results in AGD moving along negative gradient directions faster than GD.

- **Strongly nonconvex directions** $\lambda \in [-\ell, -\sqrt{\rho\epsilon}]$**:** similar to the strongly convex case, the slowest rate is for $\lambda = -\sqrt{\rho\epsilon}$ where $\mu_{\mathrm{GD}}(\lambda) = 1 + \Theta(1/\kappa)$ while $\mu_{\mathrm{AGD}}(\lambda) = 1 + \Theta(1/\sqrt{\kappa})$, which results in AGD escaping saddle point faster than GD.

Finally, the approximation error (from a quadratic) is also under control in this framework. With appropriate choice of T and threshold for $E_t - E_{t+T}$ in Corollary 3.4.3, by the Cauchy-Swartz inequality we can restrict iterates $\{\mathbf{x}_\tau\}_{\tau=t}^{t+T}$ to all lie within a local ball around $\mathbf{x}_t$ with radius $\sqrt{\epsilon/\rho}$, where both the gradient and Hessian of $f(\cdot)$ and its quadratic approximation $\tilde{f}_t(\mathbf{x}) = f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \mathbf{x} - \mathbf{x}_t\rangle + \frac{1}{2}(\mathbf{x}-\mathbf{x}_t)^\top \nabla^2 f(\mathbf{x}_t)(\mathbf{x}-\mathbf{x}_t)$ are close:

Fact 3.4.4. *Assume $f(\cdot)$ is ρ-Hessian Lipschitz, then for all $\mathbf{x}$ so that $\|\mathbf{x} - \mathbf{x}_t\| \le \sqrt{\epsilon/\rho}$, we have $\|\nabla f(\mathbf{x}) - \nabla \tilde{f}_t(\mathbf{x})\| \le \epsilon$ and $\|\nabla^2 f(\mathbf{x}) - \nabla^2 \tilde{f}_t(\mathbf{x})\| = \|\nabla^2 f(\mathbf{x}) - \nabla^2 f(\mathbf{x}_t)\| \le \sqrt{\rho\epsilon}$.*

Main Framework

For simplicity of presentation, recall $\mathscr{T} := \sqrt{\kappa}\cdot\chi c = \tilde{\Theta}(\sqrt{\kappa})$ and denote $\mathscr{F} := \sqrt{\epsilon^3/\rho}\cdot\chi^{-5}c^{-7} = \tilde{\Theta}(\sqrt{\epsilon^3/\rho})$, where c is sufficiently large constant as in Theorem 3.3.1. Our overall proof strategy will be to show the following "average descent claim": *Algorithm 6 decreases the Hamiltonian by $\mathscr{F}$ in every set of $\mathscr{T}$ iterations as long as it does not reach an ϵ-second-order stationary point.* Since the Hamiltonian cannot decrease more than $E_0 - E^\star =$

$f(\mathbf{x}_0) - f^\star$, this immediately shows that it has to reach an ϵ-second-order stationary point in $O((f(\mathbf{x}_0) - f^\star)\mathscr{T}/\mathscr{F})$ steps, proving Theorem 3.3.1.

It can be verified by the choice of parameters (3.3) and Lemma 3.4.1 that whenever (3.2) holds so that NCE is triggered, the Hamiltonian decreases by at least $\mathscr{F}$ in one step. So, if NCE step is performed even once in each round of $\mathscr{T}$ steps, we achieve enough average decrease. The troublesome case is when in some time interval of $\mathscr{T}$ steps starting with $\mathbf{x}_t$, only AGD steps are performed without NCE. If $\mathbf{x}_t$ is not an ϵ-second order stationary point, either the gradient is large or the Hessian has a large negative direction. We prove the average decrease claim by considering these two cases.

Lemma 3.4.5 (Large gradient)**.** *Consider the setting of Theorem 3.3.1. If $\|\nabla f(\mathbf{x}_\tau)\| \geq \epsilon$ for all $\tau \in [t, t + \mathscr{T}]$, then by running Algorithm 6 we have $E_{t+\mathscr{T}} - E_t \leq -\mathscr{F}$.*

Lemma 3.4.6 (Negative curvature)**.** *Consider the setting of Theorem 3.3.1. If $\|\nabla f(\mathbf{x}_t)\| \leq \epsilon$, $\lambda_{\min}(\nabla^2 f(\mathbf{x}_t)) < -\sqrt{\rho\epsilon}$, and perturbation has not been added in iterations $\tau \in [t - \mathscr{T}, t)$, then by running Algorithm 6, we have $E_{t+\mathscr{T}} - E_t \leq -\mathscr{F}$ with high probability.*

We note that an important aspect of these two lemmas is that the Hamiltonian decreases by $\Omega(\mathscr{F})$ in $\mathscr{T} = \tilde{\Theta}(\sqrt{\kappa})$ steps, which is faster compared to PGD which decreases the function value by $\Omega(\mathscr{F})$ in $\mathscr{T}^2 = \tilde{\Theta}(\kappa)$ steps (Jin et al., 2017). That is, the acceleration phenomenon in PAGD happens in both cases. We also stress that under both of these settings, PAGD cannot achieve $\Omega(\mathscr{F}/\mathscr{T})$ decrease in each step—it has to accumulate momentum over time to achieve $\Omega(\mathscr{F}/\mathscr{T})$ amortized decrease.

Large Gradient Scenario

For AGD, gradient and momentum interact, and both play important roles in the dynamics. Fortunately, according to Lemma 3.4.1, the Hamiltonian decreases sufficiently whenever the momentum $\mathbf{v}_t$ is large; so it is sufficient to discuss the case where the momentum is small.

One difficulty in proving Lemma 3.4.5 lies in the difficulty of enforcing the precondition that gradients of all iterates are large even with quadratic approximation. Intuitively we hope that the large initial gradient $\|\nabla f(\mathbf{x}_t)\| \geq \epsilon$ suffices to give a sufficient decrease of the Hamiltonian. Unfortunately, this is not true. Let $\mathbb{S}$ be the subspace of eigenvectors of $\nabla^2 f(\mathbf{x}_t)$ with eigenvalues in $[\sqrt{\rho\epsilon}, \ell]$, consisting of all the strongly convex directions, and let $\mathbb{S}^c$ be the orthogonal subspace. It turns out that the initial gradient component in $\mathbb{S}$ is not very helpful in decreasing the Hamiltonian since AGD rapidly decreases the gradient in these directions. We instead prove Lemma 3.4.5 in two steps.

Lemma 3.4.7. *(informal) If $\mathbf{v}_t$ is small, $\|\nabla f(\mathbf{x}_t)\|$ not too large and $E_{t+\mathscr{T}/2} - E_t \geq -\mathscr{F}$, then for all $\tau \in [t + \mathscr{T}/4, t + \mathscr{T}/2]$ we have $\|\mathcal{P}_{\mathbb{S}} \nabla f(\mathbf{x}_\tau)\| \leq \epsilon/2$.*

Lemma 3.4.8. *(informal) If $\mathbf{v}_t$ is small and $\|\mathcal{P}_{\mathbb{S}^c} \nabla f(\mathbf{x}_t)\| \geq \epsilon/2$, then we have $E_{t+\mathscr{T}/4} - E_t \leq -\mathscr{F}$.*

See the formal versions, Lemma 3.7.5 and Lemma 3.7.6, for more details. We see that if the Hamiltonian does not decrease much (and so is localized in a small ball), the gradient in the strongly convex subspace $\|\mathcal{P}_{\mathbb{S}}\nabla f(\mathbf{x}_\tau)\|$ vanishes in $\mathscr{T}/4$ steps by Lemma 3.4.7. Since the hypothesis of Lemma 3.4.5 guarantees a large gradient for all of the $\mathscr{T}$ steps, this means that $\|\mathcal{P}_{\mathbb{S}^c}\nabla f(\mathbf{x}_t)\|$ is large after $\mathscr{T}/4$ steps, thereby decreasing the Hamiltonian in the next $\mathscr{T}/4$ steps (by Lemma 3.4.8).

Negative Curvature Scenario

In this section, we will show that the volume of the set around a strict saddle point from which AGD does not escape quickly is very small (Lemma 3.4.6). We do this using the coupling mechanism introduced in (Jin et al., 2017), which gives a fine-grained understanding of the geometry around saddle points. More concretely, letting the perturbation radius $r = \tilde{\Theta}(\epsilon/\ell)$ as specified in (3.3), we show the following lemma.

Lemma 3.4.9. *(informal) Suppose $\|\nabla f(\tilde{\mathbf{x}})\| \le \epsilon$ and $\lambda_{\min}(\nabla^2 f(\tilde{\mathbf{x}})) \le -\sqrt{\rho\epsilon}$. Let $\mathbf{x}_0, \mathbf{x}_0'$ be at distance at most r from $\tilde{\mathbf{x}}$, and $\mathbf{x}_0 - \mathbf{x}_0' = r_0\mathbf{e}_1$ where $\mathbf{e}_1$ is the minimum eigen-direction of $\nabla^2 f(\tilde{\mathbf{x}})$ and $r_0 \ge \delta r/\sqrt{d}$. Then for AGD starting at $(\mathbf{x}_0, \mathbf{v})$ and $(\mathbf{x}_0', \mathbf{v})$, we have:*

$$\min\{E_{\mathscr{T}} - \widetilde{E}, E'_{\mathscr{T}} - \widetilde{E}\} \le -\mathscr{F},$$

where $\widetilde{E}, E_{\mathscr{T}}$ and $E'_{\mathscr{T}}$ are the Hamiltonians at $(\tilde{\mathbf{x}}, \mathbf{v}), (\mathbf{x}_{\mathscr{T}}, \mathbf{v}_{\mathscr{T}})$ and $(\mathbf{x}'_{\mathscr{T}}, \mathbf{v}'_{\mathscr{T}})$ respectively.

See the formal version in Lemma 3.7.7. We note δ in above Lemma is a small number characterize the failure probability of the algorithm (as defined in Theorem 3.3.1), and $\mathscr{T}$ has logarithmic dependence on δ according to (3.3). Lemma 3.4.9 says that around any strict saddle, for any two points that are separated along the smallest eigen-direction by at least $\delta r/\sqrt{d}$, PAGD, starting from at least one of those points, decreases the Hamiltonian, and hence escapes the strict saddle. This implies that the width of the region starting from where AGD is stuck has width at most $\delta r/\sqrt{d}$, and thus has small volume.

3.5 Conclusions

In this work, we show that a variant of AGD can escape saddle points faster than GD, demonstrating that momentum techniques can indeed accelerate convergence even for non-convex optimization. Our algorithm finds an ϵ-second order stationary point in $\widetilde{\mathcal{O}}(1/\epsilon^{7/4})$ iterations, faster than the $\widetilde{\mathcal{O}}(1/\epsilon^2)$ iterations taken by GD. This is the first algorithm that is both Hessian-free and single-loop that achieves this rate. Our analysis relies on novel techniques that lead to a better understanding of momentum techniques as well as nonconvex optimization.

The results here also give rise to several questions. The first concerns lower bounds; is the rate of $\widetilde{\mathcal{O}}(1/\epsilon^{7/4})$ that we have established here optimal for gradient-based methods

under the setting of gradient and Hessian-Lipschitz? We believe this upper bound is very likely sharp up to log factors, and developing a tight algorithm-independent lower bound will be necessary to settle this question. The second is whether the negative-curvature-exploitation component of our algorithm is actually necessary for the fast rate. To attempt to answer this question, we may either explore other ways to track the progress of standard AGD (other than the particular Hamiltonian that we have presented here), or consider other discretizations of the ODE (3.4) so that the property (3.5) is preserved even for the most nonconvex region. A final direction for future research is the extension of our results to the finite-sum setting and the stochastic setting.

3.6 Proof of Hamiltonian Lemmas

In this section, we prove Lemma 3.4.1, Lemma 3.4.2 and Corollary 3.4.3, which are presented in Section 3.4 and Section 3.4. In section 3.6 we also give an example where standard AGD with negative curvature exploitation can increase the Hamiltonian.

Recall that we define the Hamiltonian as $E_t := f(\mathbf{x}_t) + \frac{1}{2\eta}\|\mathbf{v}_t\|^2$, where, for AGD, we define $\mathbf{v}_t = \mathbf{x}_t - \mathbf{x}_{t-1}$. The first lemma shows that this Hamiltonian decreases in every step of AGD for mildly nonconvex functions.

Lemma 3.6.1 (Hamiltonian decreases monotonically)**.** *Assume that the function $f(\cdot)$ is ℓ-smooth and set the learning rate to be $\eta \le \frac{1}{2\ell}$, $\theta \in [2\eta\gamma, \frac{1}{2}]$ in AGD (Algorithm 5). Then, for every iteration t where (3.2) does not hold, we have:*

$$E_{t+1} \le E_t - \frac{\theta}{2\eta}\|\mathbf{v}_t\|^2 - \frac{\eta}{4}\|\nabla f(\mathbf{y}_t)\|^2.$$

Proof. Recall that the update equation of accelerated gradient descent has following form:

$$\begin{aligned}\mathbf{x}_{t+1} &\leftarrow \mathbf{y}_t - \eta\nabla f(\mathbf{y}_t)\\ \mathbf{y}_{t+1} &\leftarrow \mathbf{x}_{t+1} + (1-\theta)(\mathbf{x}_{t+1} - \mathbf{x}_t).\end{aligned}$$

By smoothness, with $\eta \le \frac{1}{2\ell}$:

$$f(\mathbf{x}_{t+1}) \le f(\mathbf{y}_t) - \eta\|\nabla f(\mathbf{y}_t)\|^2 + \frac{\ell\eta^2}{2}\|\nabla f(\mathbf{y}_t)\|^2 \le f(\mathbf{y}_t) - \frac{3\eta}{4}\|\nabla f(\mathbf{y}_t)\|^2, \tag{3.7}$$

assuming that the precondition (3.2) does not hold:

$$f(\mathbf{x}_t) \ge f(\mathbf{y}_t) + \langle\nabla f(\mathbf{y}_t), \mathbf{x}_t - \mathbf{y}_t\rangle - \frac{\gamma}{2}\|\mathbf{y}_t - \mathbf{x}_t\|^2, \tag{3.8}$$

and given the following update equation:

$$\begin{aligned}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 =& \|\mathbf{y}_t - \mathbf{x}_t - \eta\nabla f(\mathbf{y}_t)\|^2\\ =& \left[(1-\theta)^2\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2 - 2\eta\langle\nabla f(\mathbf{y}_t), \mathbf{y}_t - \mathbf{x}_t\rangle + \eta^2\|\nabla f(\mathbf{y}_t)\|^2\right],\end{aligned} \tag{3.9}$$

we have:

$$\begin{aligned}
f(\mathbf{x}_{t+1}) + \frac{1}{2\eta}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \leq & f(\mathbf{x}_t) + \langle \nabla f(\mathbf{y}_t), \mathbf{y}_t - \mathbf{x}_t \rangle - \frac{3\eta}{4}\|\nabla f(\mathbf{y}_t)\|^2 \\
& + \frac{1+\eta\gamma}{2\eta}(1-\theta)^2\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2 - \langle \nabla f(\mathbf{y}_t), \mathbf{y}_t - \mathbf{x}_t \rangle + \frac{\eta}{2}\|\nabla f(\mathbf{y}_t)\|^2 \\
\leq & f(\mathbf{x}_t) + \frac{1}{2\eta}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2 - \frac{2\theta - \theta^2 - \eta\gamma(1-\theta)^2}{2\eta}\|\mathbf{v}_t\|^2 - \frac{\eta}{4}\|\nabla f(\mathbf{y}_t)\|^2 \\
\leq & f(\mathbf{x}_t) + \frac{1}{2\eta}\|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2 - \frac{\theta}{2\eta}\|\mathbf{v}_t\|^2 - \frac{\eta}{4}\|\nabla f(\mathbf{y}_t)\|^2.
\end{aligned}$$

The last inequality uses the fact that $\theta \in [2\eta\gamma, \frac{1}{2}]$ so that $\theta^2 \leq \frac{\theta}{2}$ and $\eta\gamma \leq \frac{\theta}{2}$. We substitute in the definition of $\mathbf{v}_t$ and E_t to finish the proof. □

We see from this proof that (3.8) relies on approximate convexity of $f(\cdot)$, which explains why in all existing proofs, the convexity between $\mathbf{x}_t$ and $\mathbf{y}_t$ is so important. A perhaps surprising fact to note is that the above proof can in fact go through even with mild non-convexity (captured in line 8 of Algorithm 6). Thus, high nonconvexity is the problematic situation. To overcome this, we need to slightly modify AGD so that the Hamiltonian is decreasing. This is formalized in the following lemma.

Lemma 3.6.1. *Assume that $f(\cdot)$ is ℓ-smooth and ρ-Hessian Lipschitz. For every iteration t of Algorithm 6 where (3.2) holds (thus running NCE), we have:*

$$E_{t+1} \leq E_t - \min\{\frac{s^2}{2\eta}, \frac{1}{2}(\gamma - 2\rho s)s^2\}.$$

Proof. When we perform an NCE step, we know that (3.2) holds. In the first case ($\|\mathbf{v}_t\| \geq s$), we set $\mathbf{x}_{t+1} = \mathbf{x}_t$ and set the momentum $\mathbf{v}_{t+1}$ to zero, which gives:

$$E_{t+1} = f(\mathbf{x}_{t+1}) = f(\mathbf{x}_t) = E_t - \frac{1}{2\eta}\|\mathbf{v}_t\|^2 \leq E_t - \frac{s^2}{2\eta}.$$

In the second case ($\|\mathbf{v}_t\| \leq s$), expanding in a Taylor series with Lagrange remainder, we have:

$$f(\mathbf{x}_t) = f(\mathbf{y}_t) + \langle \nabla f(\mathbf{y}_t), \mathbf{x}_t - \mathbf{y}_t \rangle + \frac{1}{2}(\mathbf{x}_t - \mathbf{y}_t)^\top \nabla^2 f(\zeta_t)(\mathbf{x}_t - \mathbf{y}_t),$$

where $\zeta_t = \phi\mathbf{x}_t + (1-\phi)\mathbf{y}_t$ and $\phi \in [0,1]$. Due to the certificate (3.2) we have

$$\frac{1}{2}(\mathbf{x}_t - \mathbf{y}_t)^\top \nabla^2 f(\zeta_t)(\mathbf{x}_t - \mathbf{y}_t) \leq -\frac{\gamma}{2}\|\mathbf{x}_t - \mathbf{y}_t\|^2.$$

On the other hand, clearly $\min\{\langle \nabla f(\mathbf{x}_t), \delta \rangle, \langle \nabla f(\mathbf{x}_t), -\delta \rangle\} \leq 0$. WLOG, suppose $\langle \nabla f(\mathbf{x}_t), \delta \rangle \leq 0$, then, by definition of $\mathbf{x}_{t+1}$, we have:

$$f(\mathbf{x}_{t+1}) \leq f(\mathbf{x}_t + \delta) = f(\mathbf{x}_t) + \langle \nabla f(\mathbf{x}_t), \delta \rangle + \frac{1}{2}\delta^\top \nabla^2 f(\zeta_t')\delta \leq f(\mathbf{x}_t) + \frac{1}{2}\delta^\top \nabla^2 f(\zeta_t')\delta,$$

where $\zeta_t' = \mathbf{x}_t + \phi'\delta$ and $\phi' \in [0,1]$. Since $\|\zeta_t - \zeta_t'\| \le 2s$, δ also lines up with $\mathbf{y}_t - \mathbf{x}_t$:

$$\delta^\top \nabla^2 f(\zeta_t')\delta \le \delta^\top \nabla^2 f(\zeta_t)\delta + \|\nabla^2 f(\zeta_t') - \nabla^2 f(\zeta_t)\|\|\delta\|^2 \le -\gamma\|\delta\|^2 + 2\rho s\|\delta\|^2.$$

Therefore, this gives

$$E_{t+1} = f(\mathbf{x}_{t+1}) \le f(\mathbf{x}_t) - \frac{1}{2}(\gamma - \rho s)s^2 \le E_t - \frac{1}{2}(\gamma - 2\rho s)s^2,$$

which finishes the proof. □

The Hamiltonian decrease has an important consequence: if the Hamiltonian does not decrease much, then all the iterates are localized in a small ball around the starting point. Moreover, the iterates do not oscillate much in this ball. We called this the improve-or-localize phenomenon.

Corollary 3.6.1 (Improve or localize)**.** *Under the same setting as in Lemma 3.4.1, if (3.2) does not hold for all steps in $[t, t+T]$, we have:*

$$\sum_{\tau=t+1}^{t+T} \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|^2 \le \frac{2\eta}{\theta}(E_t - E_{t+T}).$$

Proof. The proof follows immediately from telescoping the argument of Lemma 3.4.1. □

AGD can increase the Hamiltonian under nonconvexity

In the previous section, we proved Lemma 3.4.1 which requires $\theta \ge 2\eta\gamma$, that is, $\gamma \le \theta/(2\eta)$. In this section, we show Lemma 3.4.1 is almost tight in the sense that when $\gamma \ge 4\theta/\eta$ in (3.2), we have:

$$f(\mathbf{x}_t) \le f(\mathbf{y}_t) + \langle \nabla f(\mathbf{y}_t), \mathbf{x}_t - \mathbf{y}_t\rangle - \frac{\gamma}{2}\|\mathbf{x}_t - \mathbf{y}_t\|^2.$$

Monotonic decrease of the Hamiltonian may no longer hold, indeed, AGD can increase the Hamiltonian for those steps.

Consider a simple one-dimensional example, $f(x) = -\frac{1}{2}\gamma x^2$, where (3.2) always holds. Define the initial condition $x_0 = -1, v_0 = 1/(1-\theta)$. By update equation in Algorithm 5, the next iterate will be $x_1 = y_0 = 0$, and $v_1 = x_1 - x_0 = 1$. By the definition of Hamiltonian, we have

$$\begin{aligned} E_0 =& f(x_0) + \frac{1}{2\eta}|v_0|^2 = -\frac{\gamma}{2} + \frac{1}{2\eta(1-\theta)^2} \\ E_1 =& f(x_1) + \frac{1}{2\eta}|v_1|^2 = \frac{1}{2\eta}, \end{aligned}$$

since $\theta \le 1/4$. It is not hard to verify that whenever $\gamma \ge 4\theta/\eta$, we will have $E_1 \ge E_0$; that is, the Hamiltonian increases in this step.

This fact implies that when we pick a large learning rate η and small momentum parameter θ (both are essential for acceleration), standard AGD does not decrease the Hamiltonian in a very nonconvex region. We need another mechanism such as NCE to fix the monotonically decreasing property.

3.7 Proof of Main Result

In this section, we set up the machinery needed to prove our main result, Theorem 3.3.1. We first present the generic setup, then, as in Section 3.4, we split the proof into two cases, one where gradient is large and the other where the Hessian has negative curvature. In the end, we put everything together and prove Theorem 3.3.1.

To simplify the proof, we introduce some notation for this section, and state a convention regarding absolute constants. Recall the choice of parameters in Eq.(3.3):

$$\eta = \frac{1}{4\ell}, \quad \theta = \frac{1}{4\sqrt{\kappa}}, \quad \gamma = \frac{\theta^2}{\eta} = \frac{\sqrt{\rho\epsilon}}{4}, \quad s = \frac{\gamma}{4\rho} = \frac{1}{16}\sqrt{\frac{\epsilon}{\rho}}, \quad r = \eta\epsilon \cdot \chi^{-5}c^{-8},$$

where $\kappa = \frac{\ell}{\sqrt{\rho\epsilon}}, \chi = \max\{1, \log\frac{d\ell\Delta_f}{\rho\epsilon\delta}\}$, and c is a sufficiently large constant as stated in the precondition of Theorem 3.3.1. Throughout this section, we also always denote

$$\mathscr{T} := \sqrt{\kappa} \cdot \chi c, \quad \mathscr{F} := \sqrt{\frac{\epsilon^3}{\rho}} \cdot \chi^{-5}c^{-7}, \quad \mathscr{S} := \sqrt{\frac{2\eta\mathscr{T}\mathscr{F}}{\theta}} = \sqrt{\frac{2\epsilon}{\rho}} \cdot \chi^{-2}c^{-3}, \quad \mathscr{M} := \frac{\epsilon\sqrt{\kappa}}{\ell}c^{-1},$$

which represent the special units for time, the Hamiltonian, the parameter space and the momentum. All the lemmas in this section hold when the constant c is picked to be sufficiently large. To avoid ambiguity, throughout this section $O(\cdot), \Omega(\cdot), \Theta(\cdot)$ notation **only hides an absolute constant which is independent of the choice of sufficiently large constant** c, which is defined in the precondition of Theorem 3.3.1. That is, we will always make c dependence explicit in $O(\cdot), \Omega(\cdot), \Theta(\cdot)$ notation. Therefore, for a quantity like $O(c^{-1})$, we can always pick c large enough so that it cancels out the absolute constant in the $O(\cdot)$ notation, and make $O(c^{-1})$ smaller than any fixed required constant.

Common setup

Our general strategy in the proof is to show that if none of the iterates $\mathbf{x}_t$ is a SOSP, then in all $\mathscr{T}$ steps, the Hamiltonian always decreases by at least $\mathscr{F}$. This gives an average decrease of $\mathscr{F}/\mathscr{T}$. In this section, we establish some facts which will be used throughout the entire proof, including the decrease of the Hamiltonian in NCE step, the update of AGD in matrix form, and upper bounds on approximation error for a local quadratic approximation.

The first lemma shows if negative curvature exploitation is used, then in a single step, the Hamiltonian will decrease by $\mathscr{F}$.

Lemma 3.7.1. *Under the same setting as Theorem 3.3.1, for every iteration t of Algorithm 6 where (3.2) holds (thus running NCE), we have:*

$$E_{t+1} - E_t \le -2\mathscr{F}.$$

Proof. It is also easy to check that the precondition of Lemma 3.4.2 holds, and by the particular choice of parameters in Theorem 3.3.1, we have:

$$\min\{\frac{s^2}{2\eta}, \frac{1}{2}(\gamma - 2\rho s)s^2\} \ge \Omega(\mathscr{F}c^7) \ge 2\mathscr{F},$$

where the last inequality is by picking c in Theorem 3.3.1 large enough, which finishes the proof. □

Therefore, whenever NCE is called, the decrease of the Hamiltonian is already sufficient. We thus only need to focus on AGD steps. The next lemma derives a general expression for $\mathbf{x}_t$ after an AGD update, which is very useful in multiple-step analysis. The general form is expressed with respect to a reference point $\mathbf{0}$, which can be any arbitrary point (in many cases we choose it to be $\mathbf{x}_0$).

Lemma 3.7.2. *Let $\mathbf{0}$ be an origin (which can be fixed at an arbitrary point). Let $\mathcal{H} = \nabla^2 f(\mathbf{0})$. Then an AGD (Algorithm 5) update can be written as:*

$$\begin{pmatrix} \mathbf{x}_{t+1} \\ \mathbf{x}_t \end{pmatrix} = \mathbf{A}^t \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_0 \end{pmatrix} - \eta \sum_{\tau=1}^{t} \mathbf{A}^{t-\tau} \begin{pmatrix} \nabla f(\mathbf{0}) + \delta_\tau \\ 0 \end{pmatrix}, \tag{3.10}$$

where $\delta_\tau = \nabla f(\mathbf{y}_\tau) - \nabla f(\mathbf{0}) - \mathcal{H}\mathbf{y}_\tau$, and

$$\mathbf{A} = \begin{pmatrix} (2-\theta)(\mathbf{I} - \eta\mathcal{H}) & -(1-\theta)(\mathbf{I} - \eta\mathcal{H}) \\ \mathbf{I} & 0 \end{pmatrix}.$$

Proof. Substituting for $(\mathbf{y}_t, \mathbf{v}_t)$ in Algorithm 5, we have a recursive equation for $\mathbf{x}_t$:

$$\mathbf{x}_{t+1} = (2-\theta)\mathbf{x}_t - (1-\theta)\mathbf{x}_{t-1} - \eta\nabla f((2-\theta)\mathbf{x}_t - (1-\theta)\mathbf{x}_{t-1}). \tag{3.11}$$

By definition of δ_τ, we also have:

$$\nabla f(\mathbf{y}_\tau) = \nabla f(\mathbf{0}) + \mathcal{H}\mathbf{y}_\tau + \delta_\tau.$$

Therefore, in matrix form, we have:

$$\begin{aligned} \begin{pmatrix} \mathbf{x}_{t+1} \\ \mathbf{x}_t \end{pmatrix} =& \begin{pmatrix} (2-\theta)(\mathbf{I} - \eta\mathcal{H}) & -(1-\theta)(\mathbf{I} - \eta\mathcal{H}) \\ \mathbf{I} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{x}_t \\ \mathbf{x}_{t-1} \end{pmatrix} - \eta \begin{pmatrix} \nabla f(0) + \delta_t \\ 0 \end{pmatrix} \\ =& \mathbf{A}^t \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_0 \end{pmatrix} - \eta \sum_{\tau=1}^{t} \mathbf{A}^{t-\tau} \begin{pmatrix} \nabla f(0) + \delta_\tau \\ 0 \end{pmatrix}, \end{aligned}$$

which finishes the proof. □

Clearly $\mathbf{A}$ in Lemma 3.7.2 is a $2d \times 2d$ matrix, and if we expand $\mathbf{A}$ according to the eigenvector directions of $\begin{pmatrix} \mathcal{H} & 0 \\ 0 & \mathcal{H} \end{pmatrix}$, $\mathbf{A}$ can be reorganized as a block-diagonal matrix consisting of d 2×2 matrices. Let the jth eigenvalue of $\mathcal{H}$ be denoted λ_j, and denote $\mathbf{A}_j$ as the jth 2×2 matrix with corresponding eigendirections:

$$\mathbf{A}_j = \begin{pmatrix} (2-\theta)(1-\eta\lambda_j) & -(1-\theta)(1-\eta\lambda_j) \\ 1 & 0 \end{pmatrix}. \tag{3.12}$$

We note that the choice of reference point $\mathbf{0}$ is mainly to simplify mathmatical expressions involving $\mathbf{x}_t - \mathbf{0}$.

Lemma 3.7.2 can be viewed as update from a quadratic expansion around origin $\mathbf{0}$, and δ_τ is the approximation error which marks the difference between true function and its quadratic approximation. The next lemma shows that when sequence $\mathbf{x}_0, \cdots, \mathbf{x}_t$ are all close to $\mathbf{0}$, then the approximation error is under control:

Proposition 3.7.3. *Using the notation of Lemma 3.7.2, if for any $\tau \le t$, we have $\|\mathbf{x}_\tau\| \le R$, then for any $\tau \le t$, we also have*

1. $\|\delta_\tau\| \le O(\rho R^2)$;
2. $\|\delta_\tau - \delta_{\tau-1}\| \le O(\rho R)(\|\mathbf{x}_t - \mathbf{x}_{\tau-1}\| + \|\mathbf{x}_{\tau-1} - \mathbf{x}_{\tau-2}\|)$;
3. $\sum_{\tau=1}^t \|\delta_\tau - \delta_{\tau-1}\|^2 \le O(\rho^2 R^2) \sum_{\tau=1}^t \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|^2$.

Proof. Let $\Delta_\tau = \int_0^1 (\nabla^2 f(\phi \mathbf{y}_\tau) - \mathcal{H}) \mathrm{d}\phi$. The first inequality is true because $\delta_\tau = \Delta_\tau \mathbf{y}_\tau$, thus:

$$\begin{aligned}\|\delta_\tau\| =& \|\Delta_\tau \mathbf{y}_\tau\| \le \|\Delta_\tau\| \|\mathbf{y}_\tau\| = \|\int_0^1 (\nabla^2 f(\phi \mathbf{y}_\tau) - \mathcal{H}) \mathrm{d}\phi \| \|\mathbf{y}_\tau\| \\ \le& \int_0^1 \|(\nabla^2 f(\phi \mathbf{y}_\tau) - \mathcal{H})\| \mathrm{d}\phi \cdot \|\mathbf{y}_\tau\| \le \rho \|\mathbf{y}_\tau\|^2 \le \rho \|(2-\theta)\mathbf{x}_\tau - (1-\theta)\mathbf{x}_{\tau-1}\|^2 \le O(\rho R^2).\end{aligned}$$

For the second inequality, we have:

$$\delta_\tau - \delta_{\tau-1} = \nabla f(\mathbf{y}_\tau) - \nabla f(\mathbf{y}_{\tau-1}) - \mathcal{H}(\mathbf{y}_\tau - \mathbf{y}_{\tau-1}) = \Delta'_\tau (\mathbf{y}_\tau - \mathbf{y}_{\tau-1}),$$

where $\Delta'_\tau = \int_0^1 (\nabla^2 f(\mathbf{y}_{\tau-1} + \phi(\mathbf{y}_\tau - \mathbf{y}_{\tau-1})) - \mathcal{H}) \mathrm{d}\phi$. As in the proof of the first inequality, we have:

$$\begin{aligned}\|\delta_\tau - \delta_{\tau-1}\| \le& \|\Delta'_\tau\| \|\mathbf{y}_\tau - \mathbf{y}_{\tau-1}\| = \|\int_0^1 (\nabla^2 f(\mathbf{y}_{\tau-1} + \phi(\mathbf{y}_\tau - \mathbf{y}_{\tau-1})) - \mathcal{H}) \mathrm{d}\phi \| \|\mathbf{y}_\tau - \mathbf{y}_{\tau-1}\| \\ \le& \rho \max\{\|\mathbf{y}_\tau\|, \|\mathbf{y}_{\tau-1}\|\} \|\mathbf{y}_\tau - \mathbf{y}_{\tau-1}\| \le O(\rho R)(\|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\| + \|\mathbf{x}_{\tau-1} - \mathbf{x}_{\tau-2}\|).\end{aligned}$$

Finally, since $(\|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\| + \|\mathbf{x}_{\tau-1} - \mathbf{x}_{\tau-2}\|)^2 \le 2(\|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|^2 + \|\mathbf{x}_{\tau-1} - \mathbf{x}_{\tau-2}\|^2)$, the third inequality is immediately implied by the second inequality. □

Proof for large-gradient scenario

We prove Lemma 3.4.5 in this subsection. Throughout this subsection, we let $\mathbb{S}$ be the subspace with eigenvalues in $(\theta^2/[\eta(2-\theta)^2], \ell]$, and let $\mathbb{S}^c$ be the complementary subspace. Also let $\mathcal{P}_{\mathbb{S}}$ and $\mathcal{P}_{\mathbb{S}^c}$ be the corresponding projections. We note $\theta^2/[\eta(2-\theta)^2] = \Theta(\sqrt{\rho\epsilon})$, and this particular choice lies at the boundary between the real eigenvalues and complex eigenvalues of the matrix $\mathbf{A}_j$, as shown in Lemma 3.8.3.

The first lemma shows that if momentum or gradient is very large, then the Hamiltonian already has sufficient decrease on average.

Lemma 3.7.4. *Under the setting of Theorem 3.3.1, if $\|\mathbf{v}_t\| \geq \mathscr{M}$ or $\|\nabla f(\mathbf{x}_t)\| \geq 2\ell\mathscr{M}$, and at time step t only AGD is used without NCE or perturbation, then:*

$$E_{t+1} - E_t \leq -4\mathscr{F}/\mathscr{T}.$$

Proof. When $\|\mathbf{v}_t\| \geq \frac{\epsilon\sqrt{\kappa}}{10\ell}$, by Lemma 3.4.1, we have:

$$E_{t+1} - E_t \leq -\frac{\theta}{2\eta}\|\mathbf{v}_t\|^2 \leq -\Omega\left(\frac{\ell}{\sqrt{\kappa}}\frac{\epsilon^2\kappa}{\ell^2}c^{-2}\right) = -\Omega\left(\frac{\epsilon^2\sqrt{\kappa}}{2\ell}c^{-2}\right) \leq -\Omega(\frac{\mathscr{F}}{\mathscr{T}}c^6) \leq -\frac{4\mathscr{F}}{\mathscr{T}}.$$

The last step is by picking c to be a large enough constant. When $\|\mathbf{v}_t\| \leq \mathscr{M}$ but $\|\nabla f(\mathbf{x}_t)\| \geq 2\ell\mathscr{M}$, by the gradient Lipschitz assumption, we have:

$$\|\nabla f(\mathbf{y}_t)\| \geq \|\nabla f(\mathbf{x}_t)\| - (1-\theta)\ell\|\mathbf{v}_t\| \geq \ell\mathscr{M}.$$

Similarly, by Lemma 3.4.1, we have:

$$E_{t+1} - E_t \leq -\frac{\eta}{4}\|\nabla f(\mathbf{y}_t)\|^2 \leq -\Omega(\frac{\epsilon^2\kappa}{\ell}c^{-2}) \leq -\Omega(\frac{\mathscr{F}}{\mathscr{T}}c^6) \leq -\frac{4\mathscr{F}}{\mathscr{T}}.$$

Again the last step is by picking c to be a large enough constant, which finishes the proof. □

Next, we show that if the initial momentum is small, but the initial gradient on the non-convex subspace $\mathbb{S}^c$ is large enough, then within $O(\mathscr{T})$ steps, the Hamiltonian will decrease by at least $\mathscr{F}$.

Lemma 3.7.5 (Formal Version of Lemma 3.4.8)**.** *Under the setting of Theorem 3.3.1, if $\|\mathcal{P}_{\mathbb{S}^c}\nabla f(\mathbf{x}_0)\| \geq \frac{\epsilon}{2}$, $\|\mathbf{v}_0\| \leq \mathscr{M}$, $\mathbf{v}_0^\top[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]\mathbf{v}_0 \leq 2\sqrt{\rho\epsilon}\mathscr{M}^2$, and for $t \in [0, \mathscr{T}/4]$ only AGD steps are used without NCE or perturbation, then:*

$$E_{\mathscr{T}/4} - E_0 \leq -\mathscr{F}.$$

Proof. The high-level plan is a proof by contradiction. We first assume that the energy doesn't decrease very much; that is, $E_{\mathscr{T}/4} - E_0 \geq -\mathscr{F}$ for a small enough constant μ. By Corollary 3.4.3 and the Cauchy-Swartz inequality, this immediately implies that for all

$t \le \mathscr{T}$, we have $\|\mathbf{x}_t - \mathbf{x}_0\| \le \sqrt{2\eta\mathscr{T}\mathscr{F}/(4\theta)} = \mathscr{S}/2$. In the rest of the proof we will show that this leads to a contradiction.

Given initial $\mathbf{x}_0$ and $\mathbf{v}_0$, we define $\mathbf{x}_{-1} = \mathbf{x}_0 - \mathbf{v}_0$. Without loss of generality, set $\mathbf{x}_0$ as the origin $\mathbf{0}$. Using the notation and results of Lemma 3.7.2, we have the following update equation:

$$\begin{pmatrix} \mathbf{x}_t \\ \mathbf{x}_{t-1} \end{pmatrix} = \mathbf{A}^t \begin{pmatrix} 0 \\ -\mathbf{v}_0 \end{pmatrix} - \eta \sum_{\tau=0}^{t-1} \mathbf{A}^{t-1-\tau} \begin{pmatrix} \nabla f(0) + \delta_\tau \\ 0 \end{pmatrix}.$$

Consider the j-th eigen-direction of $\mathcal{H} = \nabla^2 f(\mathbf{0})$, recall the definition of the 2×2 block matrix $\mathbf{A}_j$ as in (3.12), and denote

$$(a_t^{(j)}, \ -b_t^{(j)}) = \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}_j^t.$$

Then we have for the j-th eigen-direction:

$$\begin{aligned} x_t^{(j)} =& b_t^{(j)} v_0^{(j)} - \eta \sum_{\tau=0}^{t-1} a_{t-1-\tau}^{(j)} (\nabla f(0)^{(j)} + \delta_\tau^{(j)}) \\ =& -\eta \left[\sum_{\tau=0}^{t-1} a_\tau^{(j)} \right] \left(\nabla f(0)^{(j)} + \sum_{\tau=0}^{t-1} p_\tau^{(j)} \delta_\tau^{(j)} + q_t^{(j)} v_0^{(j)} \right), \end{aligned}$$

where

$$p_\tau^{(j)} = \frac{a_{t-1-\tau}^{(j)}}{\sum_{\tau=0}^{t-1} a_\tau^{(j)}} \quad \text{and} \quad q_t^{(j)} = -\frac{b_t^{(j)}}{\eta \sum_{\tau=0}^{t-1} a_\tau^{(j)}}.$$

Clearly $\sum_{\tau=0}^{t-1} p_\tau^{(j)} = 1$. For $j \in \mathbb{S}^c$, by Lemma 3.8.7, we know $\sum_{\tau=0}^{t-1} a_\tau^{(j)} \ge \Omega(\frac{1}{\theta^2})$. We can thus further write the above equation as:

$$x_t^{(j)} = -\eta \left[\sum_{\tau=0}^{t-1} a_\tau^{(j)} \right] \left(\nabla f(0)^{(j)} + \tilde{\delta}^{(j)} + \tilde{v}^{(j)} \right),$$

where $\tilde{\delta}^{(j)} = \sum_{\tau=0}^{t-1} p_\tau^{(j)} \delta_\tau^{(j)}$ and $\tilde{v}^{(j)} = q_t^{(j)} v_0^{(j)}$, coming from the Hessian Lipschitz assumption and the initial momentum respectively. For the remaining part, we would like to bound $\|\mathcal{P}_{\mathbb{S}^c} \tilde{\delta}\|$ and $\|\mathcal{P}_{\mathbb{S}^c} \tilde{\mathbf{v}}\|$, and show that both of them are small compared to $\|\mathcal{P}_{\mathbb{S}^c} \nabla f(\mathbf{x}_0)\|$.

First, for the $\|\mathcal{P}_{\mathbb{S}^c} \tilde{\delta}\|$ term, we know by definition of the subspace $\mathbb{S}^c$, and given that both eigenvalues of $\mathbf{A}_j$ are real and positive according to Lemma 3.8.3, such that $p_\tau^{(j)}$ is positive

by Lemma 3.8.1, we have for any $j \in \mathbb{S}^c$:

$$|\tilde{\delta}^{(j)}| = |\sum_{\tau=0}^{t-1} p_\tau^{(j)} \delta_\tau^{(j)}| \leq \sum_{\tau=0}^{t-1} p_\tau^{(j)} (|\delta_0^{(j)}| + |\delta_\tau^{(j)} - \delta_0^{(j)}|)$$
$$\leq \left[\sum_{\tau=0}^{t-1} p_\tau^{(j)}\right] \left(|\delta_0^{(j)}| + \sum_{\tau=1}^{t-1} |\delta_\tau^{(j)} - \delta_{\tau-1}^{(j)}|\right) \leq |\delta_0^{(j)}| + \sum_{\tau=1}^{t-1} |\delta_\tau^{(j)} - \delta_{\tau-1}^{(j)}|.$$

By the Cauchy-Swartz inequality, this gives:

$$\|\mathcal{P}_{\mathbb{S}^c}\tilde{\delta}\|^2 = \sum_{j\in\mathbb{S}^c} |\tilde{\delta}^{(j)}|^2 \leq \sum_{j\in\mathbb{S}^c} (|\delta_0^{(j)}| + \sum_{\tau=1}^{t-1} |\delta_\tau^{(j)} - \delta_{\tau-1}^{(j)}|)^2 \leq 2\left[\sum_{j\in\mathbb{S}^c} |\delta_0^{(j)}|^2 + \sum_{j\in\mathbb{S}^c} (\sum_{\tau=1}^{t-1} |\delta_\tau^{(j)} - \delta_{\tau-1}^{(j)}|)^2\right]$$
$$\leq 2\left[\sum_{j\in\mathbb{S}^c} |\delta_0^{(j)}|^2 + t\sum_{j\in\mathbb{S}^c}\sum_{\tau=1}^{t-1} |\delta_\tau^{(j)} - \delta_{\tau-1}^{(j)}|^2\right] \leq 2\|\delta_0\|^2 + 2t\sum_{\tau=1}^{t-1} \|\delta_\tau - \delta_{\tau-1}\|^2.$$

Recall that for $t \leq \mathscr{T}$, we have $\|\mathbf{x}_t\| \leq \mathscr{S}/2$. By Proposition 3.7.3, we know: $\|\delta_0\| \leq O(\rho\mathscr{S}^2)$, and by Corollary 3.4.3 and Proposition 3.7.3:

$$t\sum_{\tau=1}^{t-1} \|\delta_\tau - \delta_{\tau-1}\|^2 \leq O(\rho^2\mathscr{S}^2) t\sum_{\tau=1}^{t-1} \|\mathbf{x}_\tau - \mathbf{x}_{\tau-1}\|^2 \leq O(\rho^2\mathscr{S}^4).$$

This gives $\|\mathcal{P}_{\mathbb{S}^c}\tilde{\delta}\| \leq O(\rho\mathscr{S}^2) \leq O(\epsilon \cdot c^{-6}) \leq \epsilon/10$.

Next we consider the $\|\mathcal{P}_{\mathbb{S}^c}\tilde{\mathbf{v}}\|$ term. By Lemma 3.8.7, we have

$$-\eta q_t^{(j)} = \frac{b_t}{\sum_{\tau=0}^{t-1} a_\tau} \leq O(1)\max\{\theta, \sqrt{\eta|\lambda_j|}\}.$$

This gives:

$$\|\mathcal{P}_{\mathbb{S}^c}\tilde{\mathbf{v}}\|^2 = \sum_{j\in\mathbb{S}^c} [q_t^{(j)} v_0^{(j)}]^2 \leq O(1)\sum_{j\in\mathbb{S}^c} \frac{\max\{\eta|\lambda_j|, \theta^2\}}{\eta^2} [v_0^{(j)}]^2. \tag{3.13}$$

Recall that we have assumed by way of contradiction that $E_{\mathscr{T}/4} - E_0 \leq -\mathscr{F}$. By the precondition that NCE is not used at $t = 0$, due to the certificate (3.2), we have:

$$\frac{1}{2}\mathbf{v}_0^\top \nabla^2 f(\zeta_0)\mathbf{v}_0 \geq -\frac{\gamma}{2}\|\mathbf{v}_0\|^2 = -\frac{\sqrt{\rho\epsilon}}{8}\|\mathbf{v}_0\|^2,$$

where $\zeta_0 = \phi\mathbf{x}_0 + (1-\phi)\mathbf{y}_0$ and $\phi \in [0,1]$. Noting that we fix $\mathbf{x}_0$ as the origin $\mathbf{0}$, by the Hessian Lipschitz property, it is easy to show that $\|\nabla^2 f(\zeta_0) - \mathcal{H}\| \leq \rho\|\mathbf{y}_0\| \leq \rho\|\mathbf{v}_0\| \leq \rho\mathscr{M} \leq \sqrt{\rho\epsilon}$. This gives:

$$\mathbf{v}_0\mathcal{H}\mathbf{v}_0 \geq -2\sqrt{\rho\epsilon}\|\mathbf{v}_0\|^2.$$

Again letting λ_j denote the eigenvalues of $\mathcal{H}$, rearranging the above sum give:

$$\begin{aligned}\sum_{j:\lambda_j\le 0}|\lambda_j|[v_0^{(j)}]^2 \le& O(\sqrt{\rho\epsilon})\|\mathbf{v}_0\|^2+\sum_{j:\lambda_j>0}\lambda_j[v_0^{(j)}]^2\\ \le& O(\sqrt{\rho\epsilon})\|\mathbf{v}_0\|^2+\sum_{j:\lambda_j>\theta^2/\eta(2-\theta)^2}\lambda_j[v_0^{(j)}]^2\le O(\sqrt{\rho\epsilon})\|\mathbf{v}_0\|^2+\mathbf{v}_0^\top[\mathcal{P}_{\mathbb{S}}^\top\mathcal{H}\mathcal{P}_{\mathbb{S}}]\mathbf{v}_0.\end{aligned}$$

The second inequality uses the fact that $\theta^2/\eta(2-\theta)^2 \le O(\sqrt{\rho\epsilon})$. Substituting into (3.13) gives:

$$\|\mathcal{P}_{\mathbb{S}^c}\tilde{\mathbf{v}}\|^2 \le O(\frac{1}{\eta})\left[\sqrt{\rho\epsilon}\|\mathbf{v}_0\|^2+\mathbf{v}_0^\top[\mathcal{P}_{\mathbb{S}}^\top\mathcal{H}\mathcal{P}_{\mathbb{S}}]\mathbf{v}_0\right]\le O(\ell\sqrt{\rho\epsilon}\mathscr{M}^2)=O(\epsilon^2c^{-2})\le\epsilon^2/100.$$

Finally, putting all pieces together, we have:

$$\begin{aligned}\|\mathbf{x}_t\| \ge&\|\mathcal{P}_{\mathbb{S}^c}\mathbf{x}_t\|\ge\eta\left[\min_{j\in\mathbb{S}^c}\sum_{\tau=0}^{t-1}a_\tau^{(j)}\right]\|\mathcal{P}_{\mathbb{S}^c}(\nabla f(0)+\tilde{\delta}+\tilde{\mathbf{v}})\|\\ \ge&\Omega(\frac{\eta}{\theta^2})\left[\|\mathcal{P}_{\mathbb{S}^c}\nabla f(0)\|-\|\mathcal{P}_{\mathbb{S}^c}\tilde{\delta}\|-\|\mathcal{P}_{\mathbb{S}^c}\tilde{\mathbf{v}})\|\right]\ge\Omega(\frac{\eta\epsilon}{\theta^2})\ge\Omega(\mathscr{S}c^3)\ge\mathscr{S}\end{aligned}$$

which contradicts the fact $\|\mathbf{x}_t\|$ that remains inside the ball around $\mathbf{0}$ with radius $\mathscr{S}/2$. □

The next lemma shows that if the initial momentum and gradient are reasonably small, and the Hamitonian does not have sufficient decrease over the next $\mathscr{T}$ iterations, then both the gradient and momentum of the strongly convex component $\mathbb{S}$ will vanish in $\mathscr{T}/4$ iterations.

Lemma 3.7.6 (Formal Version of Lemma 3.4.7)**.** *Under the setting of Theorem 3.3.1, suppose $\|\mathbf{v}_0\|\le\mathscr{M}$ and $\|\nabla f(\mathbf{x}_0)\|\le 2\ell\mathscr{M}$, $E_{\mathscr{T}/2}-E_0\ge-\mathscr{F}$, and for $t\in[0,\mathscr{T}/2]$ only AGD steps are used, without NCE or perturbation. Then $\forall\, t\in[\mathscr{T}/4,\mathscr{T}/2]$:*

$$\|\mathcal{P}_{\mathbb{S}}\nabla f(\mathbf{x}_t)\|\le\frac{\epsilon}{2}\quad\textit{and}\quad\mathbf{v}_t^\top[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]\mathbf{v}_t\le\sqrt{\rho\epsilon}\mathscr{M}^2.$$

Proof. Since $E_{\mathscr{T}}-E_0\ge-\mathscr{F}$, by Corollary 3.4.3 and the Cauchy-Swartz inequality, we see that for all $t\le\mathscr{T}$ we have $\|\mathbf{x}_t-\mathbf{x}_0\|\le\sqrt{2\eta\mathscr{T}\mathscr{F}/\theta}=\mathscr{S}$.

Given initial $\mathbf{x}_0$ and $\mathbf{v}_0$, we define $\mathbf{x}_{-1}=\mathbf{x}_0-\mathbf{v}_0$. Without loss of generality, setting $\mathbf{x}_0$ as the origin $\mathbf{0}$, by the notation and results of Lemma 3.7.2, we have the update equation:

$$\begin{pmatrix}\mathbf{x}_t\\ \mathbf{x}_{t-1}\end{pmatrix}=\mathbf{A}^t\begin{pmatrix}0\\ -\mathbf{v}_0\end{pmatrix}-\eta\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\nabla f(0)+\delta_\tau\\ 0\end{pmatrix}.\tag{3.14}$$

First we prove the upper bound on the gradient: $\forall\, t \in [\mathscr{T}/4, \mathscr{T}]$, we have $\|\mathcal{P}_{\mathbb{S}}\nabla f(\mathbf{x}_t)\| \le \frac{\epsilon}{2}$. Let $\Delta_t = \int_0^1 (\nabla^2 f(\phi \mathbf{x}_t) - \mathcal{H})\mathrm{d}\phi$. According to (3.14), we have:

$$\begin{aligned}\nabla f(\mathbf{x}_t) =& \nabla f(0) + (\mathcal{H} + \Delta_t)\mathbf{x}_t \\ =& \underbrace{\left(\mathbf{I} - \eta\mathcal{H}\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\mathbf{I}\\ 0\end{pmatrix}\right)\nabla f(0)}_{\mathbf{g}_1} + \underbrace{\mathcal{H}\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}0\\ -\mathbf{v}_0\end{pmatrix}}_{\mathbf{g}_2} \\ & \underbrace{-\eta\mathcal{H}\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\delta_t\\ 0\end{pmatrix}}_{\mathbf{g}_3} + \underbrace{\Delta_t\mathbf{x}_t}_{\mathbf{g}_4}.\end{aligned}$$

We will upper bound four terms $\mathbf{g}_1, \mathbf{g}_2, \mathbf{g}_3, \mathbf{g}_4$ separately. Clearly, for the last term $\mathbf{g}_4$, we have:

$$\|\mathbf{g}_4\| \le \rho\|\mathbf{x}_t\|^2 \le O(\rho\mathscr{S}^2) = O(\epsilon c^{-6}) \le \epsilon/8.$$

Next, we show that the first two terms $\mathbf{g}_1, \mathbf{g}_2$ become very small for $t \in [\mathscr{T}/4, \mathscr{T}]$. Consider coordinate $j \in \mathbb{S}$ and the 2×2 block matrix $\mathbf{A}_j$. By Lemma 3.8.2 we have:

$$1 - \eta\lambda_j\begin{pmatrix}1 & 0\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}_j^{t-1-\tau}\begin{pmatrix}1\\ 0\end{pmatrix} = \begin{pmatrix}1 & 0\end{pmatrix}\mathbf{A}_j^t\begin{pmatrix}1\\ 1\end{pmatrix}.$$

Denote:

$$(a_t^{(j)}, \ -b_t^{(j)}) = \begin{pmatrix}1 & 0\end{pmatrix}\mathbf{A}_j^t.$$

By Lemma 3.8.9, we know:

$$\max_{j\in\mathbb{S}}\left\{|a_t^{(j)}|, |b_t^{(j)}|\right\} \le (t+1)(1-\theta)^{\frac{t}{2}}.$$

This immediately gives when $t \ge \mathscr{T}/4 = \Omega(\frac{c}{\theta}\log\frac{1}{\theta})$ for c sufficiently large:

$$\begin{aligned}\|\mathcal{P}_{\mathbb{S}}\mathbf{g}_1\|^2 =& \sum_{j\in\mathbb{S}}|(a_t^{(j)} - b_t^{(j)})\nabla f(0)^{(j)}|^2 \le (t+1)^2(1-\theta)^t\|\nabla f(0)\|^2 \le \epsilon^2/64 \\ \|\mathcal{P}_{\mathbb{S}}\mathbf{g}_2\|^2 =& \sum_{j\in\mathbb{S}}|\lambda_j b_t^{(j)}\mathbf{v}_0^{(j)}|^2 \le \ell^2(t+1)^2(1-\theta)^t\|\mathbf{v}_0\|^2 \le \epsilon^2/64.\end{aligned}$$

Finally, for $\mathbf{g}_3$, by Lemma 3.8.11, for all $j \in \mathbb{S}$, we have

$$|\mathbf{g}_3^{(j)}| = \left|\eta\lambda_j\sum_{\tau=0}^{t-1}a_\tau^{(j)}\delta_{t-1-\tau}\right| \le |\delta_{t-1}^{(j)}| + \sum_{\tau=1}^{t-1}|\delta_\tau^{(j)} - \delta_{\tau-1}^{(j)}|.$$

By Proposition 3.7.3, this gives:

$$\|\mathcal{P}_{\mathbb{S}}\mathbf{g}_3\|^2 \le 2\|\delta_{t-1}\|^2 + 2t\sum_{\tau=1}^{t-1}\|\delta_\tau - \delta_{\tau-1}\|^2 \le O(\rho^2\mathscr{S}^4) \le O(\epsilon^2\cdot c^{-12}) \le \epsilon^2/64.$$

In sum, this gives for any fixed $t \in [\mathscr{T}/4, \mathscr{T}]$:

$$\|\mathcal{P}_{\mathbb{S}}\nabla f(\mathbf{x}_t)\| \le \|\mathcal{P}_{\mathbb{S}}\mathbf{g}_1\| + \|\mathcal{P}_{\mathbb{S}}\mathbf{g}_2\| + \|\mathcal{P}_{\mathbb{S}}\mathbf{g}_3\| + \|\mathbf{g}_4\| \le \frac{\epsilon}{2}.$$

We now provide a similar argument to prove the upper bound for the momentum. That is, $\forall\, t \in [\mathscr{T}/4, \mathscr{T}]$, we show $\mathbf{v}_t^\top[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]\mathbf{v}_t \le \sqrt{\rho\epsilon}\mathscr{M}^2$. According to (3.14), we have:

$$\mathbf{v}_t = \begin{pmatrix}1 & -1\end{pmatrix}\begin{pmatrix}\mathbf{x}_t\\ \mathbf{x}_{t-1}\end{pmatrix} = \underbrace{\begin{pmatrix}1 & -1\end{pmatrix}\mathbf{A}^t\begin{pmatrix}0\\ -\mathbf{v}_0\end{pmatrix}}_{\boldsymbol{m}_1} - \underbrace{\eta\begin{pmatrix}1 & -1\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\nabla f(0)\\ 0\end{pmatrix}}_{\boldsymbol{m}_2}$$
$$\underbrace{-\eta\begin{pmatrix}1 & -1\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix}}_{\boldsymbol{m}_3}.$$

Consider the j-th eigendirection, so that $j \in \mathbb{S}$, and recall the 2×2 block matrix $\mathbf{A}_j$. Denoting

$$(a_t^{(j)},\ -b_t^{(j)}) = \begin{pmatrix}1 & 0\end{pmatrix}\mathbf{A}_j^t,$$

by Lemma 3.8.1 and 3.8.9, we have for $t \ge \mathscr{T}/4 = \Omega(\frac{c}{\theta}\log\frac{1}{\theta})$ with c sufficiently large:

$$\|[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]^{\frac{1}{2}}\boldsymbol{m}_1\|^2 = \sum_{j\in\mathbb{S}}|\lambda_j^{\frac{1}{2}}(b_t^{(j)} - b_{t-1}^{(j)})\mathbf{v}_0^{(j)}|^2 \le \ell(t+1)^2(1-\theta)^t\|\mathbf{v}_0\|^2 \le O(\frac{\epsilon^2}{\ell}c^{-3}) \le \frac{1}{3}\sqrt{\rho\epsilon}\mathscr{M}^2.$$

On the other hand, by Lemma 3.8.2, we have:

$$\left|\eta\lambda_j\begin{pmatrix}1 & -1\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}_j^{t-1-\tau}\begin{pmatrix}1\\ 0\end{pmatrix}\right| = \left|\eta\lambda_j\begin{pmatrix}1 & 0\end{pmatrix}\sum_{\tau=0}^{t-1}(\mathbf{A}_j^{t-1-\tau} - \mathbf{A}_j^{t-2-\tau})\begin{pmatrix}1\\ 0\end{pmatrix}\right| = \left|\begin{pmatrix}1 & 0\end{pmatrix}(\mathbf{A}_j^t - \mathbf{A}_j^{t-1})\begin{pmatrix}1\\ 1\end{pmatrix}\right|.$$

This gives, for $t \ge \mathscr{T}/4 = \Omega(\frac{c}{\theta}\log\frac{1}{\theta})$, and for c sufficiently large:

$$\begin{aligned}\|[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]^{\frac{1}{2}}\boldsymbol{m}_2\|^2 =& \sum_{j\in\mathbb{S}}|\lambda_j^{-\frac{1}{2}}(a_t^{(j)} - a_{t-1}^{(j)} - b_t^{(j)} + b_{t-1}^{(j)})\nabla f(0)^{(j)}|^2\\ \le& O(\frac{1}{\sqrt{\rho\epsilon}})(t+1)^2(1-\theta)^t\|\nabla f(0)\|^2 \le O(\frac{\epsilon^2}{\ell}c^{-3}) \le \frac{1}{3}\sqrt{\rho\epsilon}\mathscr{M}^2.\end{aligned}$$

Finally, for any $j \in \mathbb{S}$, by Lemma 3.8.11, we have:

$$|(\mathcal{H}^{\frac{1}{2}}\boldsymbol{m}_3)^{(j)}| = |\eta\lambda_j^{\frac{1}{2}}\sum_{\tau=0}^{t-1}(a_\tau - a_{\tau-1})\delta_{t-1-\tau}| \le \sqrt{\eta}\left[\sum|\delta_{t-1}^{(j)}| + \sum_{\tau=1}^{t-1}|\delta_\tau^{(j)} - \delta_{\tau-1}^{(j)}|\right].$$

Again by Proposition 3.7.3:

$$\|[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]^{\frac{1}{2}}\boldsymbol{m}_3\|^2 = \eta\left[2\|\delta_{t-1}\|^2 + 2t\sum_{\tau=1}^{t-1}\|\delta_\tau - \delta_{\tau-1}\|^2\right] \le O(\eta\rho^2\mathscr{S}^4) \le O(\frac{\epsilon^2}{\ell}c^{-6}) \le \frac{1}{3}\sqrt{\rho\epsilon}\mathscr{M}^2.$$

Putting everything together, we have:

$$\begin{aligned}\mathbf{v}_t^\top[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]\mathbf{v}_t \le& \|[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]^{\frac{1}{2}}\boldsymbol{m}_1\|^2 + \|[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]^{\frac{1}{2}}\boldsymbol{m}_2\|^2 \\ &+ \|[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_0)\mathcal{P}_{\mathbb{S}}]^{\frac{1}{2}}\boldsymbol{m}_3\|^2 \le \sqrt{\rho\epsilon}\mathscr{M}^2.\end{aligned}$$

This finishes the proof. □

Finally, we are ready to prove the main lemma of this subsection (Lemma 3.4.5), which claims that if gradients in $\mathscr{T}$ iterations are always large, then the Hamiltonian will decrease sufficiently within a small number of steps.

Lemma 3.7.7 (Large gradient)**.** *Consider the setting of Theorem 3.3.1. If $\|\nabla f(\mathbf{x}_\tau)\| \ge \epsilon$ for all $\tau \in [0, \mathscr{T}]$, then by running Algorithm 6 we have $E_{\mathscr{T}} - E_0 \le -\mathscr{F}$.*

Proof. Since $\|\nabla f(\mathbf{x}_\tau)\| \ge \epsilon$ for all $\tau \in [0, \mathscr{T}]$, according to Algorithm 6, the precondition to add perturbation never holds, so Algorithm will not add any perturbation in these $\mathscr{T}$ iterations.

Next, suppose there is at least one iteration where NCE is used. Then by Lemma 3.7.1, we know that that step alone gives $\mathscr{F}$ decrease in the Hamiltonian. According to Lemma 3.4.1 and Lemma 3.7.1 we know that without perturbation, the Hamiltonian decreases monotonically in the remaining steps. This means whenever at least one NCE step is performed, Lemma 3.4.5 immediately holds.

For the remainder of the proof, we can restrict the discussion to the case where NCE is never performed in steps $\tau \in [0, \mathscr{T}]$. Letting

$$\tau_1 = \arg\min_{t\in[0,\mathscr{T}]}\{t \,|\, \|\mathbf{v}_t\| \le \mathscr{M} \text{ and } \|\nabla f(\mathbf{x}_t)\| \le 2\ell\mathscr{M}\},$$

we know in case $\tau_1 \ge \frac{\mathscr{T}}{4}$, that Lemma 3.7.4 ensures $E_{\mathscr{T}} - E_0 \le E_{\frac{\mathscr{T}}{4}} - E_0 \le -\mathscr{F}$. Thus, we only need to discuss the case $\tau_1 \le \frac{\mathscr{T}}{4}$. Again, if $E_{\tau_1+\mathscr{T}/2} - E_{\tau_1} \le -\mathscr{F}$, Lemma 3.4.5 immediately holds. For the remaining case, $E_{\tau_1+\mathscr{T}/2} - E_{\tau_1} \le -\mathscr{F}$, we apply Lemma 3.7.6 starting at τ_1, and obtain

$$\|\mathcal{P}_{\mathbb{S}}\nabla f(\mathbf{x}_t)\| \le \frac{\epsilon}{2} \text{ and } \mathbf{v}_t^\top[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_{\tau_1})\mathcal{P}_{\mathbb{S}}]\mathbf{v}_t \le \sqrt{\rho\epsilon}\mathscr{M}^2. \qquad \forall t \in [\tau_1 + \frac{\mathscr{T}}{4}, \tau_1 + \frac{\mathscr{T}}{2}].$$

Letting:

$$\tau_2 = \arg\min_{t\in[\tau_1+\frac{\mathscr{T}}{4},\mathscr{T}]} \{t \,|\|\mathbf{v}_t\| \le \mathscr{M}\},$$

by Lemma 3.7.4 we again know we only need to discuss the case where $\tau_2 \le \tau_1 + \frac{\mathscr{T}}{2}$; otherwise, we already guarantee sufficient decrease in the Hamiltonian. Then, we clearly have $\|\mathcal{P}_{\mathbb{S}}\nabla f(\mathbf{x}_{\tau_2})\| \le \frac{\epsilon}{2}$, also by the precondition of Lemma 3.4.5, we know $\|\nabla f(\mathbf{x}_{\tau_2})\| \ge \epsilon$, thus $\|\mathcal{P}_{\mathbb{S}^c}\nabla f(\mathbf{x}_{\tau_2})\| \ge \frac{\epsilon}{2}$. On the other hand, since if the Hamiltonian does not decrease enough, $E_{\tau_2} - E_0 \ge -\mathscr{F}$, by Lemma 3.4.3, we have $\|\mathbf{x}_{\tau_1} - \mathbf{x}_{\tau_2}\| \le 2\mathscr{S}$, by the Hessian Lipschitz property, which gives:

$$\mathbf{v}_{\tau_2}^\top[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_{\tau_2})\mathcal{P}_{\mathbb{S}}]\mathbf{v}_{\tau_2} \le \mathbf{v}_{\tau_2}^\top[\mathcal{P}_{\mathbb{S}}^\top\nabla^2 f(\mathbf{x}_{\tau_1})\mathcal{P}_{\mathbb{S}}]\mathbf{v}_{\tau_2} + \|\nabla^2 f(\mathbf{x}_{\tau_1}) - \nabla^2 f(\mathbf{x}_{\tau_2})\|\|\mathbf{v}_{\tau_2}\|^2 \le 2\sqrt{\rho\epsilon}\mathscr{M}^2.$$

Now $\mathbf{x}_{\tau_2}$ satisfies all the preconditions of Lemma 3.7.5, and by applying Lemma 3.7.5 we finish the proof. □

Proof for negative-curvature scenario

We prove Lemma 3.4.6 in this section. We consider two trajectories, starting at $\mathbf{x}_0$ and $\mathbf{x}'_0$, with $\mathbf{v}_0 = \mathbf{v}'_0$, where $\mathbf{w}_0 = \mathbf{x}_0 - \mathbf{x}'_0 = r_0\mathbf{e}_1$, where $\mathbf{e}_1$ is the minimum eigenvector direction of $\mathcal{H}$, and where r_0 is not too small. We show that at least one of the trajectories will escape saddle points efficiently.

Lemma 3.7.7 (Formal Version of Lemma 3.4.9)**.** *Under the same setting as Theorem 3.3.1, suppose $\|\nabla f(\tilde{\mathbf{x}})\| \le \epsilon$ and $\lambda_{\min}(\nabla^2 f(\tilde{\mathbf{x}})) \le -\sqrt{\rho\epsilon}$. Let $\mathbf{x}_0$ and $\mathbf{x}'_0$ be at distance at most r from $\tilde{\mathbf{x}}$. Let $\mathbf{x}_0 - \mathbf{x}'_0 = r_0 \cdot \mathbf{e}_1$ and let $\mathbf{v}_0 = \mathbf{v}'_0 = \tilde{\mathbf{v}}$ where $\mathbf{e}_1$ is the minimum eigen-direction of $\nabla^2 f(\tilde{\mathbf{x}})$. Let $r_0 \ge \frac{\delta\mathscr{F}}{2\Delta_f}\cdot\frac{r}{\sqrt{d}}$. Then, running AGD starting at $(\mathbf{x}_0, \mathbf{v}_0)$ and $(\mathbf{x}'_0, \mathbf{v}'_0)$ respectively, we have:*

$$\min\{E_{\mathscr{T}} - \widetilde{E}, E'_{\mathscr{T}} - \widetilde{E}\} \le -\mathscr{F},$$

where $\widetilde{E}, E_{\mathscr{T}}$ and $E'_{\mathscr{T}}$ are the Hamiltonians at $(\tilde{\mathbf{x}}, \tilde{\mathbf{v}}), (\mathbf{x}_{\mathscr{T}}, \mathbf{v}_{\mathscr{T}})$ and $(\mathbf{x}'_{\mathscr{T}}, \mathbf{v}'_{\mathscr{T}})$ respectively.

Proof. Assume none of the two sequences decrease the Hamiltonian fast enough; that is,

$$\min\{E_{\mathscr{T}} - E_0, E'_{\mathscr{T}} - E'_0\} \ge -2\mathscr{F},$$

where E_0 and E'_0 are the Hamiltonians at $(\mathbf{x}_0, \mathbf{v}_0)$ and $(\mathbf{x}'_0, \mathbf{v}'_0)$. Then, by Corollary 3.4.3 and the Cauchy-Swartz inequality, we have for any $t \le \mathscr{T}$:

$$\max\{\|\mathbf{x}_t - \tilde{\mathbf{x}}\|, \|\mathbf{x}'_t - \tilde{\mathbf{x}}\|\} \le r + \max\{\|\mathbf{x}_t - \mathbf{x}_0\|, \|\mathbf{x}'_t - \mathbf{x}'_0\|\} \le r + \sqrt{4\eta\mathscr{T}\mathscr{F}/\theta} \le 2\mathscr{S}.$$

Fix the origin $\mathbf{0}$ at $\tilde{\mathbf{x}}$ and let $\mathcal{H}$ be the Hessian at $\tilde{\mathbf{x}}$. Recall that the update equation of AGD (Algorithm 5) can be re-written as:

$$\mathbf{x}_{t+1} = (2-\theta)\mathbf{x}_t - (1-\theta)\mathbf{x}_{t-1} - \eta\nabla f((2-\theta)\mathbf{x}_t - (1-\theta)\mathbf{x}_{t-1})$$

Taking the difference of two AGD sequences starting from $\mathbf{x}_0, \mathbf{x}_0'$, and let $\mathbf{w}_t = \mathbf{x}_t - \mathbf{x}_t'$, we have:

$$\begin{aligned}\mathbf{w}_{t+1} =&(2-\theta)\mathbf{w}_t - (1-\theta)\mathbf{w}_{t-1} - \eta\nabla f(\mathbf{y}_t) + \eta\nabla f(\mathbf{y}_t')\\ =&(2-\theta)(I - \eta\mathcal{H} - \eta\Delta_t)\mathbf{w}_t - (1-\theta)(I-\eta\mathcal{H}-\eta\Delta_t)\mathbf{w}_{t-1},\end{aligned}$$

where $\Delta_t = \int_0^1 (\nabla^2 f(\phi\mathbf{y}_t + (1-\phi)\mathbf{y}_t') - \mathcal{H})\mathrm{d}\phi$. In the last step, we used

$$\nabla f(\mathbf{y}_t) - \nabla f(\mathbf{y}_t') = (\mathcal{H}+\Delta_t)(\mathbf{y}_t - \mathbf{y}_t') = (\mathcal{H}+\Delta_t)[(2-\theta)\mathbf{w}_t - (1-\theta)\mathbf{w}_{t-1}].$$

We thus obtain the update of the $\mathbf{w}_t$ sequence in matrix form:

$$\begin{aligned}\begin{pmatrix}\mathbf{w}_{t+1}\\ \mathbf{w}_t\end{pmatrix} =& \begin{pmatrix}(2-\theta)(\mathbf{I}-\eta\mathcal{H}) & -(1-\theta)(\mathbf{I}-\eta\mathcal{H})\\ \mathbf{I} & 0\end{pmatrix}\begin{pmatrix}\mathbf{w}_t\\ \mathbf{w}_{t-1}\end{pmatrix}\\ &- \eta\begin{pmatrix}(2-\theta)\Delta_t\mathbf{w}_t - (1-\theta)\Delta_t\mathbf{w}_{t-1}\\ 0\end{pmatrix}\\ =&\mathbf{A}\begin{pmatrix}\mathbf{w}_t\\ \mathbf{w}_{t-1}\end{pmatrix} - \eta\begin{pmatrix}\delta_t\\ 0\end{pmatrix} = \mathbf{A}^{t+1}\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_{-1}\end{pmatrix} - \eta\sum_{\tau=0}^{t}\mathbf{A}^{t-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix},\end{aligned} \tag{3.15}$$

where $\delta_t = (2-\theta)\Delta_t\mathbf{w}_t - (1-\theta)\Delta_t\mathbf{w}_{t-1}$. Since $\mathbf{v}_0 = \mathbf{v}_0'$, we have $\mathbf{w}_{-1} = \mathbf{w}_0$, and $\|\Delta_t\| \le \rho\max\{\|\mathbf{x}_t - \tilde{\mathbf{x}}\|, \|\mathbf{x}_t' - \tilde{\mathbf{x}}\|\} \le 2\rho\mathscr{S}$, as well as $\|\delta_\tau\| \le 6\rho\mathscr{S}(\|\mathbf{w}_\tau\| + \|\mathbf{w}_{\tau-1}\|)$. According to (3.15):

$$\mathbf{w}_t = \begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix} - \eta\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix}.$$

Intuitively, we want to say that the first term dominates. Technically, we will set up an induction based on the following fact:

$$\|\eta\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix}\| \le \frac{1}{2}\|\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\|.$$

It is easy to check the base case holds for $t = 0$. Then, assume that for all time steps less than or equal to t, the induction assumption hold. We have:

$$\begin{aligned}\|\mathbf{w}_t\| \le&\|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\| + \|\eta\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix}\|\\ \le&2\|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\|,\end{aligned}$$

which gives:

$$\begin{aligned}\|\delta_t\| \leq & O(\rho\mathscr{S})(\|\mathbf{w}_t\| + \|\mathbf{w}_{t-1}\|) \leq O(\rho\mathscr{S}) \left[\|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\| + \|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^{t-1}\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\|\right] \\ \leq & O(\rho\mathscr{S})\|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\|,\end{aligned}$$

where in the last inequality, we used Lemma 3.8.15 for monotonicity in t.

To prove that the induction assumption holds for $t+1$ we compute:

$$\begin{aligned}\|\eta\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\sum_{\tau=0}^{t}\mathbf{A}^{t-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix}\| \leq & \eta\sum_{\tau=0}^{t}\|\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\mathbf{A}^{t-\tau}\begin{pmatrix}\mathbf{I}\\ 0\end{pmatrix}\|\|\delta_\tau\| \\ \leq & O(\eta\rho\mathscr{S})\sum_{\tau=0}^{t}\|\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\mathbf{A}^{t-\tau}\begin{pmatrix}\mathbf{I}\\ 0\end{pmatrix}\|\|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^{\tau}\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\|. \quad (3.16)\end{aligned}$$

By the precondition we have $\lambda_{\min}(\mathcal{H}) \leq -\sqrt{\rho\epsilon}$. Without loss of generality, assume that the minimum eigenvector direction of $\mathcal{H}$ is along he first coordinate $\mathbf{e}_1$, and denote the corresponding 2×2 matrix as $\mathbf{A}_1$ (as in the convention of (3.12). Let:

$$(a_t^{(1)}, \ -b_t^{(1)}) = \begin{pmatrix}1 & 0\end{pmatrix}\mathbf{A}_1^t.$$

We then see that (1) $\mathbf{w}_0$ is along the $\mathbf{e}_1$ direction, and (2) according to Lemma 3.8.14, the matrix $\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\mathbf{A}^{t-\tau}\begin{pmatrix}\mathbf{I}\\ 0\end{pmatrix}$ is a diagonal matrix, where the spectral norm is achieved along the first coordinate which corresponds to the eigenvalue $\lambda_{\min}(\mathcal{H})$. Therefore, using Equation (3.16), we have:

$$\begin{aligned}\|\eta\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\sum_{\tau=0}^{t}\mathbf{A}^{t-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix}\| \leq & O(\eta\rho\mathscr{S})\sum_{\tau=0}^{t}a_{t-\tau}^{(1)}(a_\tau^{(1)} - b_\tau^{(1)})\|\mathbf{w}_0\| \\ \leq & O(\eta\rho\mathscr{S})\sum_{\tau=0}^{t}[\frac{2}{\theta} + (t+1)]|a_{t+1}^{(1)} - b_{t+1}^{(1)}|\|\mathbf{w}_0\| \\ \leq & O(\eta\rho\mathscr{S}\mathscr{T}^2)\|\begin{pmatrix}\mathbf{I}, 0\end{pmatrix}\mathbf{A}^{t+1}\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\|,\end{aligned}$$

where, in the second to last step, we used Lemma 3.8.13, and in the last step we used $1/\theta \le \mathscr{T}$. Finally, $O(\eta\rho\mathscr{S}\mathscr{T}^2) \le O(c^{-1}) \le 1/2$ by choosing a sufficiently large constant c. Therefore, we have proved the induction, which gives us:

$$\|\mathbf{w}_t\| = \|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\| - \|\eta\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\sum_{\tau=0}^{t-1}\mathbf{A}^{t-1-\tau}\begin{pmatrix}\delta_\tau\\ 0\end{pmatrix}\| \ge \frac{1}{2}\|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\|.$$

Noting that $\lambda_{\min}(\mathcal{H}) \le -\sqrt{\rho\epsilon}$, by applying Lemma 3.8.15 we have

$$\frac{1}{2}\|\begin{pmatrix}\mathbf{I} & 0\end{pmatrix}\mathbf{A}^t\begin{pmatrix}\mathbf{w}_0\\ \mathbf{w}_0\end{pmatrix}\| \ge \frac{\theta}{4}(1+\Omega(\theta))^t r_0,$$

which grows exponentially. Therefore, for $r_0 \ge \frac{\delta\mathscr{F}}{2\Delta_f}\cdot\frac{r}{\sqrt{d}}$, and $\mathscr{T} = \Omega(\frac{1}{\theta}\cdot\chi c)$ where $\chi = \max\{1, \log\frac{d\ell\Delta_f}{\rho\epsilon\delta}\}$, where the constant c is sufficiently large, we have

$$\|\mathbf{x}_{\mathscr{T}} - \mathbf{x}'_{\mathscr{T}}\| = \|\mathbf{w}_{\mathscr{T}}\| \ge \frac{\theta}{4}(1+\Omega(\theta))^{\mathscr{T}} r_0 \ge 4\mathscr{S},$$

which contradicts the fact that:

$$\forall t \le \mathscr{T}, \max\{\|\mathbf{x}_t - \tilde{\mathbf{x}}\|, \|\mathbf{x}'_t - \tilde{\mathbf{x}}\|\} \le O(\mathscr{S}).$$

This means our assumption is wrong, and we can therefore conclude:

$$\min\{E_{\mathscr{T}} - E_0, E'_{\mathscr{T}} - E'_0\} \le -2\mathscr{F}.$$

On the other hand, by the precondition on $\tilde{x}$ and the gradient Lipschitz property, we have:

$$\max\{E_0 - \tilde{E}, E'_0 - \tilde{E}\} \le \epsilon r + \frac{\ell r^2}{2} \le \mathscr{F},$$

where the last step is due to our choice of $r = \eta\epsilon\cdot\chi^{-5}c^{-8}$ in (3.3). Combining these two facts:

$$\min\{E_{\mathscr{T}} - \tilde{E}, E'_{\mathscr{T}} - \tilde{E}\} \le \min\{E_{\mathscr{T}} - E_0, E'_{\mathscr{T}} - E'_0\} + \max\{E_0 - \tilde{E}, E'_0 - \tilde{E}\} \le -\mathscr{F},$$

which finishes the proof. □

We are now ready to prove the main lemma in this subsection, which states with that random perturbation, PAGD will escape saddle points efficiently with high probability.

Lemma 3.7.8 (Negative curvature)**.** *Consider the setting of Theorem 3.3.1. If $\|\nabla f(\mathbf{x}_0)\| \le \epsilon$, $\lambda_{\min}(\nabla^2 f(\mathbf{x}_0)) < -\sqrt{\rho\epsilon}$, and a perturbation has not been added in iterations $\tau \in [-\mathscr{T}, 0)$, then, by running Algorithm 6, we have $E_{\mathscr{T}} - E_0 \le -\mathscr{F}$ with probability at least $1 - \frac{\delta\mathscr{F}}{2\Delta_f}$.*

Proof. Since a perturbation has not been added in iterations $\tau \in [-\mathscr{T}, 0)$, according to PAGD (Algorithm 6), we add perturbation at $t = 0$, the Hamiltonian will increase by at most:

$$\Delta E \leq \epsilon r + \frac{\ell r^2}{2} \leq \mathscr{F},$$

where the last step is due to our choice of $r = \eta\epsilon \cdot \chi^{-5} c^{-8}$ in (3.3) with constant c sufficiently large. Again by Algorithm 6, a perturbation will never be added in the remaining iterations, and by Lemma 3.4.1 and Lemma 3.7.1 we know the Hamiltonian always decreases for the remaining steps. Therefore, if at least one NCE step is performed in iteration $\tau \in [0, \mathscr{T}]$, by Lemma 3.7.1 we will decrease $2\mathscr{F}$ in that NCE step, and at most increase by $\mathscr{F}$ due to the perturbation. This immediately gives $E_{\mathscr{T}} - E_0 \leq -\mathscr{F}$.

Therefore, we only need to focus on the case where NCE is never used in iterations $\tau \in [0, \mathscr{T}]$. Let $\mathbb{B}_{\mathbf{x}_0}(r)$ denote the ball with radius r around $\mathbf{x}_0$. According to algorithm 6, we know the iterate after adding perturbation to $\mathbf{x}_0$ is uniformly sampled from the ball $\mathbb{B}_{\mathbf{x}_0}(r)$. Let $\mathcal{X}_{\text{stuck}} \subset \mathbb{B}_{\mathbf{x}_0}(r)$ be the region where AGD is stuck (does not decrease the Hamiltonian $\mathscr{F}$ in $\mathscr{T}$ steps). Formally, for any point $\mathbf{x} \in \mathcal{X}_{\text{stuck}}$, let $\mathbf{x}_1, \cdots, \mathbf{x}_{\mathscr{T}}$ be the AGD sequence starting at $(\mathbf{x}, \mathbf{v}_0)$, then $E_{\mathscr{T}} - E_0 \geq -\mathscr{F}$. By Lemma 3.7.7, $\mathcal{X}_{\text{stuck}}$ can have at most width $r_0 = \frac{\delta\mathscr{F}}{2\Delta_f} \cdot \frac{r}{\sqrt{d}}$ along the minimum eigenvalue direction. Therefore,

$$\frac{\text{Vol}(\mathcal{X}_{\text{stuck}})}{\text{Vol}(\mathbb{B}^{(d)}_{\mathbf{x}_0}(r))} \leq \frac{r_0 \times \text{Vol}(\mathbb{B}^{(d-1)}_0(r))}{\text{Vol}(\mathbb{B}^{(d)}_0(r))} = \frac{r_0}{r\sqrt{\pi}} \frac{\Gamma(\frac{d}{2}+1)}{\Gamma(\frac{d}{2}+\frac{1}{2})} \leq \frac{r_0}{r\sqrt{\pi}} \cdot \sqrt{\frac{d}{2}+\frac{1}{2}} \leq \frac{\delta\mathscr{F}}{2\Delta_f}.$$

Thus, with probability at least $1 - \frac{\delta\mathscr{F}}{\Delta_f}$, the perturbation will end up outside of $\mathcal{X}_{\text{stuck}}$, which give $E_{\mathscr{T}} - E_0 \leq -\mathscr{F}$. This finishes the proof.

□

Proof of Theorem 3.3.1

Our main result is now easily obtained from Lemma 3.4.5 and Lemma 3.4.6.

Proof of Theorem 3.3.1. Suppose we never encounter any ϵ-second-order stationary point. Consider the set $\mathfrak{T} = \{\tau | \tau \in [0, \mathscr{T}] \text{ and } \|\nabla f(\mathbf{x}_\tau)\| \leq \epsilon\}$, and two cases: (1) $\mathfrak{T} = \varnothing$, in which case we know all gradients are large and by Lemma 3.4.5 we have $E_{\mathscr{T}} - E_0 \leq -\mathscr{F}$; (2) $\mathfrak{T} \neq \varnothing$. In this case, define $\tau' = \min \mathfrak{T}$; i.e., the earliest iteration where the gradient is small. Since by assumption, $\mathbf{x}'_\tau$ is not an ϵ-second-order stationary point, this gives $\nabla^2 f(\mathbf{x}_{\tau'}) \leq -\sqrt{\rho\epsilon}$, and by Lemma 3.4.6, we can conclude $E_{\tau'+\mathscr{T}} - E_0 \leq E_{\tau'+\mathscr{T}} - E_{\tau'} \leq -\mathscr{F}$. Clearly $\tau' + \mathscr{T} \leq 2\mathscr{T}$. That is, in either case, we will decrease the Hamiltonian by $\mathscr{F}$ in at most $2\mathscr{T}$ steps.

Then, for the the first case, we can repeat this argument starting at iteration $\mathscr{T}$, and for the second case, we can repeat the argument starting at iteration $\tau' + \mathscr{T}$. Therefore, we will continue to obtain a decrease of the Hamiltonian by an average of $\mathscr{F}/(2\mathscr{T})$ per step. Since the function f is lower bounded, we know the Hamiltonian can not decrease

beyond $E_0 - E^\star = f(\mathbf{x}_0) - f^\star$, which means that in $\frac{2(f(\mathbf{x}_0)-f^\star)\mathscr{T}}{\mathscr{F}}$ steps, we must encounter an ϵ-second-order stationary point at least once.

Finally, in $\frac{2(f(\mathbf{x}_0)-f^\star)\mathscr{T}}{\mathscr{F}}$ steps, we will call Lemma 3.4.6 at most $\frac{2\Delta_f}{\mathscr{F}}$ times, and since Lemma 3.4.6 holds with probability $1 - \frac{\delta\mathscr{F}}{2\Delta_f}$, by a union bound, we know that the argument above is true with probability at least:

$$1 - \frac{\delta\mathscr{F}}{2\Delta_f} \cdot \frac{2\Delta_f}{\mathscr{F}} = 1 - \delta,$$

which finishes the proof. □

3.8 Auxiliary Lemma

In this section, we present some auxiliary lemmas which are used in proving Lemma 3.7.5, Lemma 3.7.6 and Lemma 3.7.7. These deal with the large-gradient scenario (nonconvex component), the large-gradient scenario (strongly convex component), and the negative curvature scenario, respectively.

The first two lemmas establish some facts about powers of the structured matrices arising in AGD.

Lemma 3.8.1. *Let the 2×2 matrix $\mathbf{A}$ have following form, for arbitrary $a, b \in \mathbb{R}$:*

$$\mathbf{A} = \begin{pmatrix} a & b \\ 1 & 0 \end{pmatrix}.$$

Letting μ_1, μ_2 denote the two eigenvalues of $\mathbf{A}$ (can be repeated or complex eigenvalues), then, for any $t \in \mathbb{N}$:

$$\begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t = \begin{pmatrix} \sum_{i=0}^{t} \mu_1^i \mu_2^{t-i}, & -\mu_1\mu_2 \sum_{i=0}^{t-1} \mu_1^i \mu_2^{t-1-i} \end{pmatrix}$$
$$\begin{pmatrix} 0 & 1 \end{pmatrix} \mathbf{A}^t = \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^{t-1}.$$

Proof. When the eigenvalues μ_1 and μ_2 are distinct, the matrix $\mathbf{A}$ can be rewritten as $\begin{pmatrix} \mu_1 + \mu_2 & -\mu_1\mu_2 \\ 1 & 0 \end{pmatrix}$, and it is easy to check that the two eigenvectors have the form $\begin{pmatrix} \mu_1 \\ 1 \end{pmatrix}$ and $\begin{pmatrix} \mu_2 \\ 1 \end{pmatrix}$. Therefore, we can write the eigen-decomposition as:

$$\mathbf{A} = \frac{1}{\mu_1 - \mu_2} \begin{pmatrix} \mu_1 & \mu_2 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \mu_1 & 0 \\ 0 & \mu_2 \end{pmatrix} \begin{pmatrix} 1 & -\mu_2 \\ -1 & \mu_1 \end{pmatrix},$$

and the tth power has the general form:

$$\mathbf{A}^t = \frac{1}{\mu_1 - \mu_2} \begin{pmatrix} \mu_1 & \mu_2 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \mu_1^t & 0 \\ 0 & \mu_2^t \end{pmatrix} \begin{pmatrix} 1 & -\mu_2 \\ -1 & \mu_1 \end{pmatrix}$$

When there are two repeated eigenvalue μ_1, the matrix $\begin{pmatrix} a & b \\ 1 & 0 \end{pmatrix}$ can be rewritten as $\begin{pmatrix} 2\mu_1 & -\mu_1^2 \\ 1 & 0 \end{pmatrix}$. It is easy to check that $\mathbf{A}$ has the following Jordan normal form:

$$\mathbf{A} = - \begin{pmatrix} \mu_1 & \mu_1 + 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \mu_1 & 1 \\ 0 & \mu_1 \end{pmatrix} \begin{pmatrix} 1 & -(\mu_1 + 1) \\ -1 & \mu_1 \end{pmatrix},$$

which yields:

$$\mathbf{A}^t = - \begin{pmatrix} \mu_1 & \mu_1 + 1 \\ 1 & 1 \end{pmatrix} \begin{pmatrix} \mu_1^t & t\mu_1^{t-1} \\ 0 & \mu_1^t \end{pmatrix} \begin{pmatrix} 1 & -(\mu_1 + 1) \\ -1 & \mu_1 \end{pmatrix}.$$

The remainder of the proof follows from simple linear algebra calculations for both cases. □

Lemma 3.8.2. *Under the same setting as Lemma 3.8.1, for any $t \in \mathbb{N}$:*

$$(\mu_1 - 1)(\mu_2 - 1) \begin{pmatrix} 1 & 0 \end{pmatrix} \sum_{\tau=0}^{t-1} \mathbf{A}^\tau \begin{pmatrix} 1 \\ 0 \end{pmatrix} = 1 - \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t \begin{pmatrix} 1 \\ 1 \end{pmatrix}.$$

Proof. When μ_1 and μ_2 are distinct, we have:

$$\begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t = \left(\frac{\mu_1^{t+1} - \mu_2^{t+1}}{\mu_1 - \mu_2}, \quad -\frac{\mu_1\mu_2(\mu_1^t - \mu_2^t)}{\mu_1 - \mu_2} \right).$$

When μ_1, μ_2 are repeated, we have:

$$\begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t = \left((t+1)\mu_1^t, \quad -t\mu_1^{t+1} \right).$$

The remainder of the proof follows from Lemma 3.8.4 and linear algebra. □

The next lemma tells us when the eigenvalues of the AGD matrix are real and when they are complex.

Lemma 3.8.3. *Let $\theta \in (0, \frac{1}{4}]$, $\mathbf{x} \in [-\frac{1}{4}, \frac{1}{4}]$ and define the 2×2 matrix $\mathbf{A}$ as follows:*

$$\mathbf{A} = \begin{pmatrix} (2-\theta)(1-x) & -(1-\theta)(1-x) \\ 1 & 0 \end{pmatrix}$$

Then the two eigenvalues μ_1 and μ_2 of $\mathbf{A}$ are solutions of the following equation:

$$\mu^2 - (2-\theta)(1-x)\mu + (1-\theta)(1-x) = 0.$$

Moreover, when $x \in [-\frac{1}{4}, \frac{\theta^2}{(2-\theta)^2}]$, μ_1 and μ_2 are real numbers, and when $x \in (\frac{\theta^2}{(2-\theta)^2}, \frac{1}{4}]$, μ_1 and μ_2 are conjugate complex numbers.

Proof. An eigenvalue μ of the matrix $\mathbf{A}$ must satisfy the following equation:

$$\det(\mathbf{A} - \mu\mathbf{I}) = \mu^2 - (2-\theta)(1-x)\mu + (1-\theta)(1-x) = 0.$$

The discriminant is equal to

$$\begin{aligned} \Delta =& (2-\theta)^2(1-x)^2 - 4(1-\theta)(1-x) \\ =& (1-x)(\theta^2 - (2-\theta^2)x). \end{aligned}$$

Then μ_1 and μ_2 are real if and only if $\Delta \geq 0$, which finishes the proof. □

Finally, we need a simple lemma for geometric sums.

Lemma 3.8.4. *For any $\lambda > 0$ and fixed t, we have:*

$$\sum_{\tau=0}^{t-1} (\tau+1)\lambda^\tau = \frac{1-\lambda^t}{(1-\lambda)^2} - \frac{t\lambda^t}{1-\lambda}.$$

Proof. Consider the truncated geometric series:

$$\sum_{\tau=0}^{t-1} \lambda^\tau = \frac{1-\lambda^t}{1-\lambda}.$$

Taking derivatives, we have:

$$\sum_{\tau=0}^{t-1} (\tau+1)\lambda^\tau = \frac{\mathrm{d}}{\mathrm{d}\lambda} \sum_{\tau=0}^{t-1} \lambda^{\tau+1} = \frac{\mathrm{d}}{\mathrm{d}\lambda} \left[\lambda \cdot \frac{1-\lambda^t}{1-\lambda} \right] = \frac{1-\lambda^t}{(1-\lambda)^2} - \frac{t\lambda^t}{1-\lambda}.$$

□

Large-gradient scenario (nonconvex component)

All the lemmas in this section are concerned with the behavior of the AGD matrix for eigendirections of the Hessian with eigenvalues being negative or small and positive, as used in proving Lemma 3.7.5. The following lemma bounds the smallest eigenvalue of the AGD matrix for those directions.

Lemma 3.8.5. *Under the same setting as Lemma 3.8.3, and for $x \in [-\frac{1}{4}, \frac{\theta^2}{(2-\theta)^2}]$, where $\mu_1 \geq \mu_2$, we have:*

$$\mu_2 \leq 1 - \frac{1}{2}\max\{\theta, \sqrt{|x|}\}.$$

Proof. The eigenvalues satisfy:

$$\det(\mathbf{A} - \mu\mathbf{I}) = \mu^2 - (2-\theta)(1-x)\mu + (1-\theta)(1-x) = 0.$$

Let $\mu = 1 + u$. We have

$$\begin{aligned} & (1+u)^2 - (2-\theta)(1-x)(1+u) + (1-\theta)(1-x) = 0 \\ \Rightarrow \quad & u^2 + ((1-x)\theta + 2x)u + x = 0. \end{aligned}$$

Let $f(u) = u^2 + \theta u + 2xu - x\theta u + x$. To prove $\mu_2(\mathbf{A}) \leq 1 - \frac{\sqrt{|x|}}{2}$ when $x \in [-\frac{1}{4}, -\theta^2]$, we only need to verify $f(-\frac{\sqrt{|x|}}{2}) \leq 0$:

$$\begin{aligned} f(-\frac{\sqrt{|x|}}{2}) =& \frac{|x|}{4} - \frac{\theta\sqrt{|x|}}{2} + |x|\sqrt{|x|} - \frac{|x|\sqrt{|x|}\theta}{2} - |x| \\ \leq& |x|\sqrt{|x|}(1 - \frac{\theta}{2}) - \frac{3|x|}{4} \leq 0 \end{aligned}$$

The last inequality follows because $|x| \leq \frac{1}{4}$ by assumption.

For $x \in [-\theta^2, 0]$, we have:

$$f(-\frac{\theta}{2}) = \frac{\theta^2}{4} - \frac{\theta^2}{2} - x\theta + \frac{x\theta^2}{2} + x = -\frac{\theta^2}{4} + x(1-\theta) + \frac{x\theta^2}{2} \leq 0.$$

On the other hand, when $x \in [0, \theta^2/(2-\theta)^2]$, both eigenvalues are still real, and the midpoint of the two roots is:

$$\frac{u_1 + u_2}{2} = -\frac{(1-x)\theta + 2x}{2} = -\frac{\theta + (2-\theta)x}{2} \leq -\frac{\theta}{2}.$$

Combining the two cases, we have shown that when $x \in [-\theta^2, \theta^2/(2-\theta)^2]$ we have $\mu_2(\mathbf{A}) \leq 1 - \frac{\theta}{2}$.

In summary, we have proved that

$$\mu_2(\mathbf{A}) \leq \begin{cases} 1 - \frac{\sqrt{|x|}}{2}, & x \in [-\frac{1}{4}, -\theta^2] \\ 1 - \frac{\theta}{2}. & x \in [-\theta^2, \theta^2/(2-\theta)^2], \end{cases}$$

which finishes the proof. □

In the same setting as above, the following lemma bounds the largest eigenvalue.

Lemma 3.8.6. *Under the same setting as Lemma 3.8.3, and with $x \in [-\frac{1}{4}, \frac{\theta^2}{(2-\theta)^2}]$, and letting $\mu_1 \geq \mu_2$, we have:*

$$\mu_1 \leq 1 + 2\min\{\frac{|x|}{\theta}, \sqrt{|x|}\}.$$

Proof. By Lemma 3.8.3 and Vieta's formula we have:

$$(\mu_1 - 1)(\mu_2 - 1) = \mu_1\mu_2 - (\mu_1 + \mu_2) + 1 = x.$$

An application of Lemma 3.8.5 finishes the proof. □

The following lemma establishes some properties of the powers of the AGD matrix.

Lemma 3.8.7. *Consider the same setting as Lemma 3.8.3, and let $x \in [-\frac{1}{4}, \frac{\theta^2}{(2-\theta)^2}]$. Denote:*

$$(a_t, \ -b_t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t.$$

Then, for any $t \geq \frac{2}{\theta} + 1$, we have:

$$\begin{aligned} \sum_{\tau=0}^{t-1} a_\tau \geq& \Omega(\frac{1}{\theta^2}) \\ \frac{1}{b_t}\left(\sum_{\tau=0}^{t-1} a_\tau\right) \geq& \Omega(1)\min\left\{\frac{1}{\theta}, \frac{1}{\sqrt{|x|}}\right\}. \end{aligned}$$

Proof. We prove the two inequalities seperately.
First Inequality: By Lemma 3.8.1:

$$\begin{aligned} \sum_{\tau=0}^{t}\begin{pmatrix} 1 & 0 \end{pmatrix}\mathbf{A}^\tau\begin{pmatrix} 1 \\ 0 \end{pmatrix} =& \sum_{\tau=0}^{t}\sum_{i=0}^{\tau}\mu_1^{\tau-i}\mu_2^i = \sum_{\tau=0}^{t}(\mu_1\mu_2)^{\frac{\tau}{2}}\sum_{i=0}^{\tau}(\frac{\mu_1}{\mu_2})^{\frac{\tau}{2}-i} \\ \geq& \sum_{\tau=0}^{t}[(1-\theta)(1-x)]^{\frac{\tau}{2}}\cdot\frac{\tau}{2} \end{aligned}$$

The last inequality holds because in $\sum_{i=0}^{\tau}(\frac{\mu_1}{\mu_2})^{\frac{\tau}{2}-i}$ at least $\frac{\tau}{2}$ terms are greater than one. Finally, since $x \leq \theta^2/(2-\theta)^2 \leq \theta^2 \leq \theta$, we have $1 - x \geq 1 - \theta$, thus:

$$\begin{aligned} \sum_{\tau=0}^{t}[(1-\theta)(1-x)]^{\frac{\tau}{2}}\cdot\frac{\tau}{2} \geq& \sum_{\tau=0}^{t}(1-\theta)^\tau\cdot\frac{\tau}{2} \geq \sum_{\tau=0}^{1/\theta}(1-\theta)^\tau\cdot\frac{\tau}{2} \\ \geq& (1-\theta)^{\frac{1}{\theta}}\sum_{\tau=0}^{1/\theta}\frac{\tau}{2} \geq \Omega(\frac{1}{\theta^2}), \end{aligned}$$

which finishes the proof.

Second Inequality: Without loss of generality, assume $\mu_1 \geq \mu_2$. Again by Lemma 3.8.1:

$$\begin{aligned}\frac{\sum_{\tau=0}^{t-1} a_\tau}{b_t} =& \frac{\sum_{\tau=0}^{t-1}\sum_{i=0}^{\tau}\mu_1^i\mu_2^{\tau-i}}{\mu_1\mu_2\sum_{i=0}^{t-1}\mu_1^i\mu_2^{t-1-i}} = \frac{1}{\mu_1\mu_2}\sum_{\tau=0}^{t-1}\frac{\sum_{i=0}^{\tau}\mu_1^i\mu_2^{\tau-i}}{\sum_{i=0}^{t-1}\mu_1^i\mu_2^{t-1-i}} \\ \geq& \frac{1}{\mu_1\mu_2}\sum_{\tau=(t-1)/2}^{t-1}\frac{\sum_{i=0}^{\tau}\mu_1^i\mu_2^{\tau-i}}{\sum_{i=0}^{t-1}\mu_1^i\mu_2^{t-1-i}} \geq \frac{1}{\mu_1\mu_2}\sum_{\tau=(t-1)/2}^{t-1}\frac{1}{2\mu_1^{t-1-\tau}} \\ =& \frac{1}{2\mu_1\mu_2}\left[1+\frac{1}{\mu_1}+\cdots+\frac{1}{\mu_1^{(t-1)/2}}\right] \geq \frac{1}{2\mu_1\mu_2}\left[1+\frac{1}{\mu_1}+\cdots+\frac{1}{\mu_1^{1/\theta}}\right].\end{aligned}$$

The second-to-last inequality holds because it is easy to check

$$2\mu_1^{t-1-\tau}\sum_{i=0}^{\tau}\mu_1^i\mu_2^{\tau-i} \geq \sum_{i=0}^{t-1}\mu_1^i\mu_2^{t-1-i},$$

for any $\tau \geq (t-1)/2$. Finally, by Lemma 3.8.6, we have

$$\mu_1 \leq 1 + 2\min\{\frac{|x|}{\theta}, \sqrt{|x|}\}.$$

Since $\mu_1 = \Theta(1)$, $\mu_2 = \Theta(1)$, we have that when $|x| \leq \theta^2$,

$$\frac{\sum_{\tau=0}^{t-1} a_\tau}{b_t} \geq \Omega(1)\left[1+\frac{1}{\mu_1}+\cdots+\frac{1}{\mu_1^{1/\theta}}\right] \geq \Omega(1)\cdot\frac{1}{\theta}\cdot\frac{1}{(1+\theta)^{\frac{1}{\theta}}} \geq \Omega(\frac{1}{\theta}).$$

When $|x| > \theta^2$, we have:

$$\frac{\sum_{\tau=0}^{t-1} a_\tau}{b_t} \geq \Omega(1)\left[1+\frac{1}{\mu_1}+\cdots+\frac{1}{\mu_1^{1/\theta}}\right] = \Omega(1)\frac{1-\frac{1}{\mu_1^{1/\theta+1}}}{1-\frac{1}{\mu_1}} = \Omega(\frac{1}{\mu_1-1}) = \Omega(\frac{1}{\sqrt{|x|}}).$$

Combining the two cases finishes the proof. □

Large-gradient scenario (strongly convex component)

All the lemmas in this section are concerned with the behavior of the AGD matrix for eigendirections of the Hessian with eigenvalues being large and positive, as used in proving Lemma 3.7.6. The following lemma gives eigenvalues of the AGD matrix for those directions.

Lemma 3.8.8. *Under the same setting as Lemma 3.8.3, and with $x \in (\frac{\theta^2}{(2-\theta)^2}, \frac{1}{4}]$, we have $\mu_1 = re^{i\phi}$ and $\mu_2 = re^{-i\phi}$, where:*

$$r = \sqrt{(1-\theta)(1-x)}, \qquad \sin\phi = \sqrt{((2-\theta)^2x-\theta^2)(1-x)}/2r.$$

Proof. By Lemma 3.8.3, we know that μ_1 and μ_2 are two solutions of

$$\mu^2 - (2-\theta)(1-x)\mu + (1-\theta)(1-x) = 0.$$

This gives $r^2 = \mu_1\mu_2 = (1-\theta)(1-x)$. On the other hand, discriminant is equal to

$$\begin{aligned}\Delta =&(2-\theta)^2(1-x)^2 - 4(1-\theta)(1-x)\\ =&(1-x)(\theta^2 - (2-\theta^2)x).\end{aligned}$$

Since $\mathrm{Im}(\mu_1) = r\sin\phi = \frac{\sqrt{-\Delta}}{2}$, the proof is finished. □

Under the same setting as above, the following lemma delineates some properties of powers of the AGD matrix.

Lemma 3.8.9. *Under the same setting as in Lemma 3.8.3, and with $x \in (\frac{\theta^2}{(2-\theta)^2}, \frac{1}{4}]$, denote:*

$$(a_t, \ \ -b_t) = \begin{pmatrix}1 & 0\end{pmatrix}\mathbf{A}^t.$$

Then, for any $t \geq 0$, we have:

$$\max\{|a_t|,\ |b_t|\} \leq (t+1)(1-\theta)^{\frac{t}{2}}.$$

Proof. By Lemma 3.8.1 and Lemma 3.8.8, using $|\cdot|$ to denote the magnitude of a complex number, we have:

$$\begin{aligned}|a_t| =& \left|\sum_{i=0}^{t}\mu_1^i\mu_2^{t-i}\right| \leq \sum_{i=0}^{t}|\mu_1^i\mu_2^{t-i}| = (t+1)r^t \leq (t+1)(1-\theta)^{\frac{t}{2}}\\ |b_t| =& \left|\mu_1\mu_2\sum_{i=0}^{t-1}\mu_1^i\mu_2^{t-1-i}\right| \leq \sum_{i=0}^{t-1}|\mu_1^{i+1}\mu_2^{t-i}| \leq tr^{t+1} \leq t(1-\theta)^{\frac{t+1}{2}}.\end{aligned}$$

Reorganizing these two equations finishes the proof. □

The following is a technical lemma which is useful in bounding the change in the Hessian by the amount of oscillation in the iterates.

Lemma 3.8.10. *Under the same setting as Lemma 3.8.8, for any $T \geq 0$, any sequence $\{\epsilon_t\}$, and any $\varphi_0 \in [0, 2\pi]$:*

$$\sum_{t=0}^{T} r^t\sin(\phi t + \varphi_0)\epsilon_t \leq O(\frac{1}{\sin\phi})\left(|\epsilon_0| + \sum_{t=1}^{T}|\epsilon_t - \epsilon_{t-1}|\right).$$

Proof. Let $\tau = \lfloor 2\pi/\phi \rfloor$ be the approximate period, and $J = \lfloor T/\tau \rfloor$ be the number of periods that exist within time T. Then, we can group the summation by each period:

$$\begin{aligned}
\sum_{t=0}^{T} r^t \sin(\phi t)\epsilon_t =& \sum_{j=0}^{J} \left[\sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t \sin(\phi t + \varphi_0)\epsilon_t \right] \\
=& \sum_{j=0}^{J} \left[\sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t \sin(\phi t + \varphi_0)[\epsilon_{j\tau} + (\epsilon_t - \epsilon_{j\tau})] \right] \\
\le& \underbrace{\sum_{j=0}^{J} \left[\sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t \sin(\phi t + \varphi_0) \right] \epsilon_{j\tau}}_{\text{Term 1}} + \underbrace{\sum_{j=0}^{J} \left[\sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t |\epsilon_t - \epsilon_{j\tau}| \right]}_{\text{Term 2}}.
\end{aligned}$$

We prove the lemma by bounding the first term and the second term on the right-hand-side of this equation separately.

Term 2: Since $r \le 1$, it is not hard to see:

$$\begin{aligned}
\text{Term 2} =& \sum_{j=0}^{J} \left[\sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t |\epsilon_t - \epsilon_{j\tau}| \right] \\
\le& \sum_{j=0}^{J} \left[\sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t \right] \left[\sum_{t=j\tau+1}^{\min\{(j+1)\tau-1,T\}} |\epsilon_t - \epsilon_{t-1}| \right] \\
\le& \tau \sum_{j=0}^{J} \left[\sum_{t=j\tau+1}^{\min\{(j+1)\tau-1,T\}} |\epsilon_t - \epsilon_{t-1}| \right] \le \tau \sum_{t=1}^{T} |\epsilon_t - \epsilon_{t-1}|.
\end{aligned}$$

Term 1: We first study the inner-loop factor, $\sum_{t=j\tau}^{(j+1)\tau-1} r^t \sin(\phi t)$. Letting $\psi = 2\pi - \tau\phi$ be the offset for each approximate period, we have that for any $j < J$:

$$\begin{aligned}
\left| \sum_{t=j\tau}^{(j+1)\tau-1} r^t \sin(\phi t + \varphi_0) \right| =& \left| \mathrm{Im} \left[\sum_{t=0}^{\tau-1} r^{j\tau+t} e^{i\cdot[\phi(j\tau+t)+\varphi_0]} \right] \right| \\
\le& r^{j\tau} \| \sum_{t=0}^{\tau-1} r^t e^{i\cdot\phi t} \| \le r^{j\tau} \| \frac{1 - r^\tau e^{i\cdot(2\pi-\psi)}}{1 - re^{i\cdot\phi}} \| \\
=& r^{j\tau} \sqrt{\frac{(1 - r^\tau \cos\psi)^2 + (r^\tau \sin\psi)^2}{(1 - r\cos\phi)^2 + (r\sin\phi)^2}}.
\end{aligned}$$

Combined with the fact that for all $y \in [0,1]$ we have $e^{-3y} \le 1-y \le e^{-y}$, we obtain the following:

$$1 - r^\tau = 1 - [(1-\theta)(1-x)]^{\frac{\tau}{2}} = 1 - e^{-\Theta((\theta+x)\tau)} = \Theta((\theta+x)\tau) = \Theta\left(\frac{(\theta+x)}{\phi}\right) \tag{3.17}$$

Also, for any $a, b \in [0,1]$, we have $(1-ab)^2 \le (1-\min\{a,b\})^2 \le (1-a^2)^2 + (1-b^2)^2$, and by definition of τ, we immediately have $\psi \le \phi$. This yields:

$$\begin{aligned}\frac{(1-r^\tau\cos\psi)^2 + (r^\tau\sin\psi)^2}{(1-r\cos\phi)^2 + (r\sin\phi)^2} \le& \frac{2(1-r^{2\tau})^2 + 2(1-\cos^2\psi)^2 + (r^\tau\sin\psi)^2}{(r\sin\phi)^2} \\ \le& O\left(\frac{1}{\sin^2\phi}\right)\left[\frac{(\theta+x)^2}{\phi^2} + \sin^4\phi + \sin^2\phi\right] \le O\left(\frac{(\theta+x)^2}{\sin^4\phi}\right)\end{aligned}$$

The second last inequality used the fact that $r = \Theta(1)$ (although note r^τ is not $\Theta(1)$). The last inequality is true since by Lemma 3.8.8, we know $(\theta+x)/\sin^2\phi \ge \Omega(1)$. This gives:

$$\left|\sum_{t=j\tau}^{(j+1)\tau-1} r^t \sin(\phi t + \varphi_0)\right| \le r^{j\tau} \cdot \frac{\theta+x}{\sin^2\phi},$$

and therefore, we can now bound the first term:

$$\begin{aligned}\text{Term } 1 =& \sum_{j=0}^{J} \sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t \sin(\phi t+\varphi_0)\epsilon_{j\tau} = \sum_{j=0}^{J}\left[\sum_{t=j\tau}^{\min\{(j+1)\tau-1,T\}} r^t\sin(\phi t+\varphi_0)\right](\epsilon_0 + \epsilon_{j\tau} - \epsilon_0) \\ \le& O(1)\sum_{j=0}^{J-1}\left[r^{j\tau}\frac{\theta+x}{\sin^2\phi}\right](|\epsilon_0| + |\epsilon_{j\tau}-\epsilon_0|) + \sum_{t=J\tau}^{T}(|\epsilon_0| + |\epsilon_{J\tau}-\epsilon_0|) \\ \le& O(1)\left[\frac{1}{1-r^\tau}\frac{\theta+x}{\sin^2\phi} + \tau\right]\cdot\left[|\epsilon_0| + \sum_{t=1}^{T}|\epsilon_t - \epsilon_{t-1}|\right] \le \left[O(\frac{1}{\sin\phi}) + \tau\right]\cdot\left[|\epsilon_0| + \sum_{t=1}^{T}|\epsilon_t-\epsilon_{t-1}|\right].\end{aligned}$$

The second-to-last inequality used Eq.(3.17). In conclusion, since $\tau \le \frac{2\pi}{\phi} \le \frac{2\pi}{\sin\phi}$, we have:

$$\begin{aligned}\sum_{t=0}^{T} r^t\sin(\phi t+\varphi_0)\epsilon_t \le& \text{Term } 1 + \text{Term } 2 \le \left[O(\frac{1}{\sin\phi}) + 2\tau\right]\cdot\left[|\epsilon_0| + \sum_{t=1}^{T}|\epsilon_t-\epsilon_{t-1}|\right] \\ \le& O\left(\frac{1}{\sin\phi}\right)\left[|\epsilon_0| + \sum_{t=1}^{T}|\epsilon_t-\epsilon_{t-1}|\right].\end{aligned}$$

□

The following lemma combines the previous two lemmas to bound the approximation error in the quadratic.

Lemma 3.8.11. *Under the same setting as Lemma 3.8.3, and with $x \in (\frac{\theta^2}{(2-\theta)^2}, \frac{1}{4}]$, denote:*

$$(a_t, \ -b_t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t.$$

Then, for any sequence $\{\epsilon_\tau\}$, any $t \geq \Omega(\frac{1}{\theta})$, we have:

$$\sum_{\tau=0}^{t-1} a_\tau \epsilon_\tau \leq O(\frac{1}{x}) \left(|\epsilon_0| + \sum_{\tau=1}^{t-1} |\epsilon_\tau - \epsilon_{\tau-1}| \right)$$
$$\sum_{\tau=0}^{t-1} (a_\tau - a_{\tau-1}) \epsilon_\tau \leq O(\frac{1}{\sqrt{x}}) \left(|\epsilon_0| + \sum_{\tau=1}^{t-1} |\epsilon_\tau - \epsilon_{\tau-1}| \right).$$

Proof. We prove the two inequalities separately.
First Inequality: Since $x \in (\frac{\theta^2}{(2-\theta)^2}, \frac{1}{4}]$, we further split the analysis into two cases:
Case $x \in (\frac{\theta^2}{(2-\theta)^2}, \frac{2\theta^2}{(2-\theta)^2}]$: By Lemma 3.8.1, we can expand dthe left-hand-side as:

$$\sum_{\tau=0}^{t-1} a_\tau \epsilon_\tau \leq \sum_{\tau=0}^{t-1} |a_\tau| (|\epsilon_0| + |\epsilon_\tau - \epsilon_0|) \leq \left[\sum_{\tau=0}^{t-1} |a_\tau| \right] \left(|\epsilon_0| + \sum_{\tau=1}^{t-1} |\epsilon_\tau - \epsilon_{\tau-1}| \right).$$

Noting that in this case $x = \Theta(\theta^2)$, by Lemma 3.8.9 and Lemma 3.8.4, we have for $t \geq O(1/\theta)$:

$$\sum_{\tau=0}^{t-1} |a_\tau| \leq \sum_{\tau=0}^{t-1} (\tau+1)(1-\theta)^{\frac{\tau}{2}} \leq O(\frac{1}{\theta^2}) = O(\frac{1}{x}).$$

Case $x \in (\frac{2\theta^2}{(2-\theta)^2}, \frac{1}{4}]$: Again, we expand the left-hand-side as:

$$\sum_{\tau=0}^{t-1} a_\tau \epsilon_\tau = \sum_{\tau=0}^{t-1} \frac{\mu_1^{\tau+1} - \mu_2^{\tau+1}}{\mu_1 - \mu_2} \epsilon_\tau = \sum_{\tau=0}^{t-1} \frac{r^{\tau+1} \sin[(\tau+1)\phi]}{r \sin[\phi]} \epsilon_\tau.$$

Noting in this case that $x = \Theta(\sin^2 \phi)$ by Lemma 3.8.8, then by Lemma 3.8.10 we have:

$$\sum_{\tau=0}^{t-1} a_\tau \epsilon_\tau \leq O(\frac{1}{\sin^2 \phi}) \left(|\epsilon_0| + \sum_{\tau=1}^{t-1} |\epsilon_\tau - \epsilon_{\tau-1}| \right) \leq O(\frac{1}{x}) \left(|\epsilon_0| + \sum_{\tau=1}^{t-1} |\epsilon_\tau - \epsilon_{\tau-1}| \right).$$

Second Inequality: Using Lemma 3.8.1, we know:

$$\begin{aligned} a_\tau - a_{\tau-1} =& \frac{(\mu_1^{\tau+1} - \mu_2^{\tau+1}) - (\mu_1^\tau - \mu_2^\tau)}{\mu_1 - \mu_2} \\ =& \frac{r^{\tau+1} \sin[(\tau+1)\phi] - r^\tau \sin[\tau\phi]}{r \sin[\phi]} \\ =& \frac{r^\tau \sin[\tau\phi](r\cos\phi - 1) + r^{\tau+1} \cos[\tau\phi] \sin\phi}{r \sin[\phi]} \\ =& \frac{r\cos\phi - 1}{r \sin\phi} \cdot r^\tau \sin[\tau\phi] + r^\tau \cos[\tau\phi], \end{aligned}$$

where we note $r = \Theta(1)$ and the coefficient of the first term is upper bounded by the following:

$$\left|\frac{r\cos\phi - 1}{r\sin\phi}\right| \leq \frac{(1-\cos^2\phi)+(1-r^2)}{r\sin\phi} \leq O\left(\frac{\theta+x}{\sin\phi}\right).$$

As in the proof of the first inequality, we split the analysis into two cases:
Case $x \in (\frac{\theta^2}{(2-\theta)^2}, \frac{2\theta^2}{(2-\theta)^2}]$**:** Again, we use

$$\sum_{\tau=0}^{t-1}(a_\tau - a_{\tau-1})\epsilon_\tau \leq \sum_{\tau=0}^{t-1}|a_\tau - a_{\tau-1}|(|\epsilon_0| + |\epsilon_\tau - \epsilon_0|) \leq \left[\sum_{\tau=0}^{t-1}|a_\tau - a_{\tau-1}|\right]\left(|\epsilon_0| + \sum_{\tau=1}^{t-1}|\epsilon_\tau - \epsilon_{\tau-1}|\right).$$

Noting $x = \Theta(\theta^2)$, again by Lemma 3.8.4 and $|\frac{\sin\tau\phi}{\sin\phi}| \leq \tau$, we have:

$$\left[\sum_{\tau=0}^{t-1}|a_\tau - a_{\tau-1}|\right] \leq O(\theta+x)\sum_{\tau=0}^{t-1}\tau(1-\theta)^{\frac{\tau}{2}} + \sum_{\tau=0}^{t-1}(1-\theta)^{\frac{\tau}{2}} \leq O(\frac{1}{\theta}) = O(\frac{1}{\sqrt{x}}).$$

Case $x \in (\frac{2\theta^2}{(2-\theta)^2}, \frac{1}{4}]$**:** From the above derivation, we have:

$$\sum_{\tau=0}^{t-1}(a_\tau - a_{\tau-1})\epsilon_\tau = \frac{r\cos\phi - 1}{r\sin\phi}\sum_{\tau=0}^{t-1}r^\tau\sin[\tau\phi]\epsilon_\tau + \sum_{\tau=0}^{t-1}r^\tau\cos[\tau\phi]\epsilon_\tau.$$

According to Lemma 3.8.8, in this case $x = \Theta(\sin^2\phi)$, $r = \Theta(1)$ and since $\Omega(\theta^2) \leq x \leq O(1)$, we have:

$$\left|\frac{r\cos\phi - 1}{r\sin\phi}\right| \leq O\left(\frac{\theta+x}{\sin\phi}\right) \leq O\left(\frac{\theta+x}{\sqrt{x}}\right) \leq O(1).$$

Combined with Lemma 3.8.10, this gives:

$$\sum_{\tau=0}^{t-1}(a_\tau - a_{\tau-1})\epsilon_\tau \leq O(\frac{1}{\sin\phi})\left(|\epsilon_0| + \sum_{\tau=1}^{t-1}|\epsilon_\tau - \epsilon_{\tau-1}|\right) \leq O(\frac{1}{\sqrt{x}})\left(|\epsilon_0| + \sum_{\tau=1}^{t-1}|\epsilon_\tau - \epsilon_{\tau-1}|\right).$$

Putting all the pieces together finishes the proof. □

Negative-curvature scenario

In this section, we will prove the auxiliary lemmas required for proving Lemma 3.7.7.

The first lemma lower bounds the largest eigenvalue of the AGD matrix for eigen-directions whose eigenvalues are negative.

Lemma 3.8.12. *Under the same setting as Lemma 3.8.3, and with $x \in [-\frac{1}{4}, 0]$, and $\mu_1 \geq \mu_2$, we have:*

$$\mu_1 \geq 1 + \frac{1}{2}\min\{\frac{|x|}{\theta}, \sqrt{|x|}\}.$$

Proof. The eigenvalues satisfy:

$$\det(\mathbf{A}-\mu\mathbf{I}) = \mu^2 - (2-\theta)(1-x)\mu + (1-\theta)(1-x) = 0.$$

Let $\mu = 1+u$. We have

$$\begin{aligned} & (1+u)^2 - (2-\theta)(1-x)(1+u) + (1-\theta)(1-x) = 0 \\ \Rightarrow \quad & u^2 + ((1-x)\theta + 2x)u + x = 0. \end{aligned}$$

Let $f(u) = u^2 + \theta u + 2xu - x\theta u + x$. To prove $\mu_1(\mathbf{A}) \geq 1 + \frac{\sqrt{|x|}}{2}$ when $x \in [-\frac{1}{4}, -\theta^2]$, we only need to verify $f(\frac{\sqrt{|x|}}{2}) \leq 0$:

$$\begin{aligned} f(\frac{\sqrt{|x|}}{2}) =& \frac{|x|}{4} + \frac{\theta\sqrt{|x|}}{2} - |x|\sqrt{|x|} + \frac{|x|\sqrt{|x|}\theta}{2} - |x| \\ \leq& \frac{\theta\sqrt{|x|}}{2} - \frac{3|x|}{4} - |x|\sqrt{|x|}(1-\frac{\theta}{2}) \leq 0 \end{aligned}$$

The last inequality holds because $\theta \leq \sqrt{|x|}$ in this case.

For $x \in [-\theta^2, 0]$, we have:

$$f(\frac{|x|}{2\theta}) = \frac{|x|^2}{4\theta^2} + \frac{|x|}{2} - \frac{|x|^2}{\theta} + \frac{|x|^2}{2} - |x| = \frac{|x|^2}{4\theta^2} - \frac{|x|}{2} - |x|^2(\frac{1}{\theta} - \frac{1}{2}) \leq 0,$$

where the last inequality is due to $\theta^2 \geq |x|$.

In summary, we have proved

$$\mu_1(\mathbf{A}) \geq \begin{cases} 1 + \frac{\sqrt{|x|}}{2}, & x \in [-\frac{1}{4}, -\theta^2] \\ 1 + \frac{|x|}{2\theta}. & x \in [-\theta^2, 0], \end{cases}$$

which finishes the proof. □

The next lemma is a technical lemma on large powers.

Lemma 3.8.13. *Under the same setting as Lemma 3.8.3, and with $x \in [-\frac{1}{4}, 0]$, denote*

$$(a_t, \ -b_t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t.$$

Then, for any $0 \leq \tau \leq t$, we have

$$|a_{t-\tau}^{(1)}||a_\tau^{(1)} - b_\tau^{(1)}| \leq [\frac{2}{\theta} + (t+1)]|a_{t+1}^{(1)} - b_{t+1}^{(1)}|.$$

Proof. Let μ_1 and μ_2 be the two eigenvalues of the matrix $\mathbf{A}$, where $\mu_1 \geq \mu_2$. Since $x \in [-\frac{1}{4}, 0]$, according to Lemma 3.8.3 and Lemma 3.8.5, we have $0 \leq \mu_2 \leq 1 - \frac{\theta}{2} \leq 1 \leq \mu_1$, and thus expanding both sides using Lemma 3.8.1 yields:

$$\begin{aligned}
\text{LHS} =& \left[\sum_{i=0}^{t-\tau} \mu_1^{t-\tau-i}\mu_2^i\right]\left[(1-\mu_2)\left(\sum_{i=0}^{\tau-1}\mu_1^{\tau-i}\mu_2^i\right)+\mu_2^\tau\right] \\
=& \left[\sum_{i=0}^{t-\tau} \mu_1^{t-\tau-i}\mu_2^i\right](1-\mu_2)\left(\sum_{i=0}^{\tau-1}\mu_1^{\tau-i}\mu_2^i\right)+\left[\sum_{i=0}^{t-\tau} \mu_1^{t-\tau-i}\mu_2^i\right]\mu_2^\tau \\
\leq& (t-\tau+1)\mu_1^{t-\tau}(1-\mu_2)\left(\sum_{i=0}^{\tau-1}\mu_1^{\tau-i}\mu_2^i\right)+\left[\sum_{i=0}^{t-\tau} \mu_1^{t-\tau-i}\mu_2^i\right] \\
\leq& (t+1)(1-\mu_2)\left(\sum_{i=0}^{\tau-1}\mu_1^{t+1-i}\mu_2^i\right)+\frac{2}{\theta}(1-\mu_2)\left[\sum_{i=0}^{t-\tau} \mu_1^{t+1-i}\mu_2^i\right] \\
\leq& [\frac{2}{\theta}+(t+1)]\left[(1-\mu_2)\sum_{i=0}^{t}\mu_1^{t+1-i}\mu_2^i+\mu_2^{t+1}\right] = \text{RHS},
\end{aligned}$$

which finishes the proof. □

The following lemma gives properties of the $(1,1)$ element of large powers of the AGD matrix.

Lemma 3.8.14. *Let the 2×2 matrix $\mathbf{A}(x)$ be defined as follows and let $x \in [-\frac{1}{4}, 0]$ and $\theta \in (0, \frac{1}{4}]$.*

$$\mathbf{A}(x) = \begin{pmatrix} (2-\theta)(1-x) & -(1-\theta)(1-x) \\ 1 & 0 \end{pmatrix}.$$

For any fixed $t > 0$, letting $g(x) = \left|\begin{pmatrix}1 & 0\end{pmatrix}[\mathbf{A}(x)]^t\begin{pmatrix}1\\0\end{pmatrix}\right|$, then we have:

1. *$g(x)$ is a monotonically decreasing function for $x \in [-1, \theta^2/(2-\theta)^2]$.*
2. *For any $x \in [\theta^2/(2-\theta)^2, 1]$, we have $g(x) \leq g(\theta^2/(2-\theta)^2)$.*

Proof. For $x \in [-1, \theta^2/(2-\theta)^2]$, we know that $\mathbf{A}(x)$ has two real eigenvalues $\mu_1(x)$ and $\mu_2(x)$, Without loss of generality, we can assume $\mu_1(x) \geq \mu_2(x)$. By Lemma 3.8.1, we know:

$$g(x) = \left|\begin{pmatrix}1 & 0\end{pmatrix}[\mathbf{A}(x)]^t\begin{pmatrix}1\\0\end{pmatrix}\right| = \sum_{i=0}^{t}[\mu_1(x)]^i[\mu_2(x)]^{t-i} = [\mu_1(x)\mu_2(x)]^{\frac{t}{2}}\sum_{i=0}^{t}\left[\frac{\mu_1(x)}{\mu_2(x)}\right]^{\frac{t}{2}-i}.$$

By Lemma 3.8.3 and Vieta's formulas, we know that $[\mu_1(x)\mu_2(x)]^{\frac{t}{2}} = [(1-\theta)(1-x)]^{\frac{t}{2}}$ is monotonically decreasing in x. On the other hand, we have that:

$$\frac{\mu_1(x)}{\mu_2(x)} + \frac{\mu_2(x)}{\mu_1(x)} + 2 = \frac{[\mu_1(x)+\mu_2(x)]^2}{\mu_1(x)\mu_2(x)} = \frac{(2-\theta)^2(1-x)}{1-\theta}$$

is monotonically decreasing in x, implying that $\sum_{i=0}^{t}\left[\frac{\mu_1(x)}{\mu_2(x)}\right]^{\frac{t}{2}-i}$ is monotonically decreasing in x. Since both terms are positive, this implies the product is also monotonically decreasing in x, which finishes the proof of the first part.

For $x \in [\theta^2/(2-\theta)^2, 1]$, the two eigenvalues $\mu_1(x)$ and $\mu_2(x)$ are conjugate, and we have:

$$[\mu_1(x)\mu_2(x)]^{\frac{t}{2}} = [(1-\theta)(1-x)]^{\frac{t}{2}} \le [\mu_1(\theta^2/(2-\theta)^2)\mu_2(\theta^2/(2-\theta)^2)]^{\frac{t}{2}}$$

which yields:

$$\sum_{i=0}^{t}\left[\frac{\mu_1(x)}{\mu_2(x)}\right]^{\frac{t}{2}-i} \le \|\sum_{i=0}^{t}\left[\frac{\mu_1(x)}{\mu_2(x)}\right]^{\frac{t}{2}-i}\| \le \sum_{i=0}^{t}\|\frac{\mu_1(x)}{\mu_2(x)}\|^{\frac{t}{2}-i} = t+1 = \sum_{i=0}^{t}\left[\frac{\mu_1(\theta^2/(2-\theta)^2)}{\mu_2(\theta^2/(2-\theta)^2)}\right]^{\frac{t}{2}-i},$$

and this finishes the proof of the second part. □

The following lemma gives properties of the sum of the first row of large powers of the AGD matrix.

Lemma 3.8.15. *Under the same setting as Lemma 3.8.3, and with $x \in [-\frac{1}{4}, 0]$, denote*

$$(a_t, \ -b_t) = \begin{pmatrix} 1 & 0 \end{pmatrix} \mathbf{A}^t.$$

Then we have

$$|a_{t+1} - b_{t+1}| \ge |a_t - b_t|$$

and

$$|a_t - b_t| \ge \frac{\theta}{2}\left(1 + \frac{1}{2}\min\{\frac{|x|}{\theta}, \sqrt{|x|}\}\right)^t.$$

Proof. Since $x < 0$, we know that $\mathbf{A}$ has two distinct real eigenvalues. Let μ_1 and μ_2 be the two eigenvalues of $\mathbf{A}$. For the first inequality, by Lemma 3.8.1, we only need to prove:

$$\mu_1^{t+1} - \mu_2^{t+1} - \mu_1\mu_2(\mu_1^t - \mu_2^t) \ge \mu_1^t - \mu_2^t - \mu_1\mu_2(\mu_1^{t-1} - \mu_2^{t-1}).$$

Taking the difference of the LHS and RHS, we have:

$$\begin{aligned}
&\mu_1^{t+1} - \mu_2^{t+1} - \mu_1\mu_2(\mu_1^t - \mu_2^t) - (\mu_1^t - \mu_2^t) + \mu_1\mu_2(\mu_1^{t-1} - \mu_2^{t-1}) \\
=&\mu_1^t(\mu_1 - \mu_1\mu_2 - 1 + \mu_2) - \mu_2^t(\mu_2 - \mu_1\mu_2 - 1 + \mu_1) \\
=&(\mu_1^t - \mu_2^t)(\mu_1 - 1)(1 - \mu_2).
\end{aligned}$$

According to Lemma 3.8.3 and Lemma 3.8.5, $\mu_1 \geq 1 \geq \mu_2 \geq 0$, which finishes the proof of the first claim.

For the second inequality, again by Lemma 3.8.1, since both μ_1 and μ_2 are positive, we have:

$$a_t - b_t = \sum_{i=0}^{t} \mu_1^i \mu_2^{t-i} - \mu_1\mu_2 \sum_{i=0}^{t-1} \mu_1^i \mu_2^{t-1-i} \geq (1-\mu_2) \sum_{i=0}^{t} \mu_1^i \mu_2^{t-i} \geq (1-\mu_2)\mu_1^t.$$

By Lemma 3.8.5 we have $1 - \mu_2 \geq \frac{\theta}{2}$, By Lemma 3.8.12 we know $\mu_1 \geq 1 + \frac{1}{2}\min\{\frac{|x|}{\theta}, \sqrt{|x|}\}$. Combining these facts finishes the proof. □

Part II

Minmax Optimization

Chapter 4

On Stable Limit Points of Gradient Descent Ascent

Minmax optimization, especially in its general nonconvex-nonconcave formulation, has found extensive applications in modern machine learning frameworks such as generative adversarial networks (GAN), adversarial training and multi-agent reinforcement learning. Gradient-based algorithms, in particular gradient descent ascent (GDA), are widely used in practice to solve these problems. Despite the practical popularity of GDA, however, its theoretical behavior has been considered highly undesirable. Indeed, apart from possiblity of non-convergence, recent results (Daskalakis and Panageas, 2018; Mazumdar and Ratliff, 2018; Adolphs et al., 2018) show that even when GDA converges, its stable limit points can be points that are not local Nash equilibria, thus not game-theoretically meaningful.

In this work, we initiate a discussion on the proper optimality measures for minmax optimization, and introduce a new notion of local optimality—*local minmax*—as a more suitable alternative to the notion of local Nash equilibrium. We establish favorable properties of local minmax points, and show, most importantly, that as the ratio of the ascent step size to the descent step size goes to infinity, stable limit points of GDA are exactly local minmax points up to some degenerate points, demonstrating that all stable limit points of GDA have a game-theoretic meaning for minmax problems.

4.1 Introduction

Minmax optimization refers to problems of the form $\min_{\mathbf{x}} \max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$. Such problems arise in a number of fields, including mathematics, biology, social science, and particularly economics (Myerson, 2013). Due to the wide range of applications of these problems and their rich mathematical structure, they have been studied for several decades in the setting of zero-sum games. In the last few years, minmax optimization has also found significant applications in machine learning, in settings such as generative adversarial networks (GAN) (Goodfellow et al., 2014), adversarial training (Madry et al., 2017) and multi-agent

reinforcement learning (Omidshafiei et al., 2017). In practice, these minmax problems are often solved using gradient based algorithms, especially gradient descent ascent (GDA), an algorithm that alternates between a gradient descent step for $\mathbf{x}$ and some number of gradient ascent steps for $\mathbf{y}$.

Such gradient-based algorithms have been well studied for convex-concave games, where $f(\cdot, \cdot)$ is a convex function of $\mathbf{x}$ for any fixed $\mathbf{y}$ and a concave function of $\mathbf{y}$ for any fixed $\mathbf{x}$. In this case, it can be shown that the average of iterates of GDA converges to a Nash equilibrium; i.e., a point $(\mathbf{x}^*, \mathbf{y}^*)$ such that $f(\mathbf{x}^*, \mathbf{y}) \leq f(\mathbf{x}^*, \mathbf{y}^*) \leq f(\mathbf{x}, \mathbf{y}^*)$ for every $\mathbf{x}$ and $\mathbf{y}$ (Bubeck, 2015; Hazan, 2016). In the convex-concave setting, it turns out that Nash equilibria and global optima are equivalent: $(\mathbf{x}^*, \mathbf{y}^*)$ is a Nash equilibrium if and only if $f(\mathbf{x}^*, \mathbf{y}^*) = \min_{\mathbf{x}} \max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$. Most of the minmax problems arising in modern machine learning applications do not, however, have this simple convex-concave structure.

Given the widespread usage of GDA in practice, it is natural to ask about its properties when applied to general nonconvex-nonconcave settings. It turns out that this question is extremely challenging—GDA dynamics do not monotonically decrease any known potential function and GDA may not converge in general (Daskalakis et al., 2017). Worse still, even when GDA converges, recent results suggest that it has some undesirable properties. Specifically, (Daskalakis and Panageas, 2018), (Mazumdar and Ratliff, 2018), and (Adolphs et al., 2018) show that some of the stable limit points of GDA may not be Nash equilibria. This suggests that they may have nothing to do with the minmax problem being solved. This raises the following question:

Is GDA an appropriate algorithm for solving general minmax problems?

This work provides a positive theoretical answer to this queestion in the general nonconvex-nonconcave setting. Critical to our perspective is a new notion of local optimality—*local minmax*, which we propose as a more useful alternative than local Nash equilibrium for a range of problems. We show that, as the ratio of the ascent step size to the descent step size goes to infinity, the stable limit points of GDA are identical to local minmax points up to some degenerate points. Therefore, almost all stable limit points of GDA are game-theoretically meaningful for minmax problems.

Our contributions

The main contributions of the work are as follows:

- We initiate a discussion on the proper optimality measures for minmax optimization, distinguishing among pure strategy Nash equilibria, global minmax points and mixed strategy Nash equilibria. We show that the latter two are well-defined and of practical relevance. We further show a reduction from the problem of finding mixed strategy Nash equilibira to the problem of finding global minmax points for Lipschitz games, demonstrating the central importance of finding global minmax points.

- We define a new notion of local optimality—*local minmax*—as a natural local surrogate for global minmaxity. We explain its relation to local Nash equilibria and global minmax points, and we establish its first- and second-order characterizations. It is worth noting that minmax optimization exhibits unique properties compared to nonconvex optimization in that global minmax points can be neither local minmax nor stationary (see Proposition 4.4.2).

- We analyze the asymptotic behavior of GDA, and show that as the ratio of the ascent step size to the descent step size goes to infinity, stable limit points of GDA are exactly local minmax points up to some degenerate points, demonstrating that almost all stable limit points of GDA have a game-theoretic meaning for minmax problems.

- We also consider the minmax problem with an approximate oracle for the maximization over $\mathbf{y}$. We show that gradient descent with inner maximization (over $\mathbf{y}$) finds a point that is close to an approximate stationary point of $\phi(\mathbf{x}) := \max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$.

Chapter organization In Section 4.1, we review additional related work. Section 4.2 presents preliminaries. In Section 4.3, we discuss the right objective for general nonconvex-nonconcave minmax optimization. Section 4.4 presents our main results on a new notion of local optimality, the limit points of GDA and gradient descent with a maximization oracle. We conclude in Section 4.5. Due to space constraints, all proofs are presented in the appendix.

Related Work

GDA dynamics: There have been several lines of work studying GDA dynamics for minmax optimization. Cherukuri, Gharesifard, and Cortes (2017) investigate GDA dynamics under some strong conditions and show that it converges locally to Nash equilibria. Heusel et al. (2017) and Nagarajan and Kolter (2017) similarly impose strong assumptions in the setting of the training of GANs and show that under these conditions Nash equilibria are stable fixed points of GDA. Gidel et al. (2018) investigate the effect of simultaneous versus alternating gradient updates as well as the effect of momentum on the convergence in bilinear games. The most closely related analyses to ours are Mazumdar and Ratliff (2018) and Daskalakis and Panageas (2018). While Daskalakis and Panageas (2018) study minmax optimization (or zero-sum games), Mazumdar and Ratliff (2018) studies a much more general setting of non-zero-sum games and multi-player games. Both of these works show that the stable limit points of GDA are not necessarily Nash equilibria. Adolphs et al. (2018) and Mazumdar, Jordan, and Sastry (2019) propose Hessian-based algorithms whose stable fixed points are exactly Nash equilibria. We note that all the works in this setting use Nash equilibrium as the notion of goodness.

General minmax optimization in machine learning: There have also been several other recent works on minmax optimization that study algorithms other than GDA. Rafique

et al. (2018) consider nonconvex but concave minmax problems where for any $\mathbf{x}$, $f(\mathbf{x}, \cdot)$ is a concave function. In this case, they analyze an algorithm combining approximate maximization over $\mathbf{y}$ and a proximal gradient method for $\mathbf{x}$ to show convergence to stationary points. Lin et al. (2018) consider a special case of the nonconvex-nonconcave minmax problem, where the function $f(\cdot, \cdot)$ satisfies a variational inequality. In this setting, they consider a proximal algorithm that requires the solving of certain strong variational inequality problems in each step and show its convergence to stationary points. Hsieh, Liu, and Cevher (2018) propose proximal methods that asymptotically converge to a *mixed* Nash equilibrium; i.e., a distribution rather than a point.

No regret dynamics for minmax optimization: Online learning/no regret dynamics have also been used to design algorithms for minmax optimization. All of these results require, however, access to oracles which solve the minimization and maximization problems separately, keeping the other variable fixed and outputting a mixed Nash equilibrium (see, e.g., Feige, Mansour, and Schapire, 2015; Chen et al., 2017; Grnarova et al., 2017; Gonen and Hazan, 2018). Finding the global minmax point even with access to these oracles is NP hard (Chen et al., 2017).

Nonconvex optimization: Gradient-based methods are also widely used for solving nonconvex optimization problems in practice. There has been a significant amount of recent work on understanding simple gradient-based algorithms such as gradient descent in this setting. Since finding global minima is already NP hard, many works focus on obtaining convergence to second-order stationary points. Lee et al. (2016) and Panageas and Piliouras (2016) show that gradient descent converges to only these points with probability one. Ge et al. (2015) and Jin et al. (2017) show that with a small amount of randomness gradient descent also converges to second-order stationary points and give nonasymptotic rates of convergence.

4.2 Preliminaries

In this section, we will first introduce our notation, and then present definitions and results for minmax optimization, zero-sum games, and general game-theoretic dynamics that are relevant to our work.

Notation

We use bold upper-case letters $\mathbf{A}, \mathbf{B}$ to denote matrices and bold lower-case letters $\mathbf{x}, \mathbf{y}$ to denote vectors. For vectors we use $\|\cdot\|$ to denote the ℓ_2-norm, and for matrices we use $\|\cdot\|$ and $\rho(\cdot)$ to denote spectral (or operator) norm and spectral radius (largest absolute value of eigenvalues) respectively. Note that these two are in general different for asymmetric matrices. For a function $f : \mathbb{R}^d \to \mathbb{R}$, we use ∇f and $\nabla^2 f$ to denote its gradient and Hessian. For functions of two vector arguments, $f : \mathbb{R}^{d_1} \times \mathbb{R}^{d_2} \to \mathbb{R}$, we use $\nabla_{\mathbf{x}} f$, $\nabla_{\mathbf{y}} f$ and $\nabla^2_{\mathbf{xx}} f$, $\nabla^2_{\mathbf{xy}} f$, $\nabla^2_{\mathbf{yy}} f$ to denote its partial gradient and partial Hessian. We also use $O(\cdot)$ and

$o(\cdot)$ notation as follows: $f(\delta) = O(\delta)$ means $\limsup_{\delta \to 0} |f(\delta)/\delta| \leq C$ for some large absolute constant C, and $g(\delta) = o(\delta)$ means $\lim_{\delta \to 0} |g(\delta)/\delta| = 0$. For complex numbers, we use $\Re(\cdot)$ to denote its real part, and $|\cdot|$ to denote its modulus. We also use $\mathcal{P}(\cdot)$, operating over a set, to denote the collection of all probability measures over the set.

Minmax optimization and zero-sum games

In this work, we consider general minmax optimization problems. Given a function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, where $\mathcal{X} \subset \mathbb{R}^{d_1}$ and $\mathcal{Y} \subset \mathbb{R}^{d_2}$, the objective is to solve:

$$\min_{\mathbf{x} \in \mathcal{X}} \max_{\mathbf{y} \in \mathcal{Y}} f(\mathbf{x}, \mathbf{y}). \tag{4.1}$$

While classical theory mostly studied the convex-concave case where $f(\cdot, \mathbf{y})$ is convex for any fixed $\mathbf{y}$ and $f(\mathbf{x}, \cdot)$ is concave for any fixed $\mathbf{x}$, this work considers the general case, where both $f(\mathbf{x}, \cdot)$ and $f(\cdot, \mathbf{y})$ can be nonconvex and nonconcave. Optimality in this setting is defined as follows:

Definition 4.2.1. $(\mathbf{x}^\star, \mathbf{y}^\star)$ is a **global minmax point**, if for any $(\mathbf{x}, \mathbf{y})$ in $\mathcal{X} \times \mathcal{Y}$ we have:

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

The minmax problem (4.1) has been extensively studied in the game theory literature under the name of "zero-sum game." Here, two players play a competitive game with the first player playing $\mathbf{x} \in \mathcal{X}$, and then the second player playing $\mathbf{y} \in \mathcal{Y}$. $f(\mathbf{x}, \mathbf{y})$ is the payoff function which represents the value lost by the first player (which is in turn gained by the second player). In this setting the standard notion of equilibrium is the following:

Definition 4.2.2. $(\mathbf{x}^\star, \mathbf{y}^\star)$ is a **(pure strategy) Nash equilibrium** of f, if for any $(\mathbf{x}, \mathbf{y})$ in $\mathcal{X} \times \mathcal{Y}$:

$$f(\mathbf{x}^\star, \mathbf{y}) \leq f(\mathbf{x}^\star, \mathbf{y}^\star) \leq f(\mathbf{x}, \mathbf{y}^\star).$$

Pure strategy Nash equilibria play an essential role in convex-concave games since for those games, pure strategy Nash equilibria always exist, and are also global minmax points (Bubeck, 2015).

When we move to the nonconvex-nonconcave setting, these nice properties of pure strategy Nash equilibria no longer hold. Moreover, the problem of finding global solutions in this setting is NP hard in general. Therefore, previous work has consider local alternatives see, e.g., Mazumdar and Ratliff, 2018; Daskalakis and Panageas, 2018:

Definition 4.2.3. $(\mathbf{x}^\star, \mathbf{y}^\star)$ is a **local (pure strategy) Nash equilirium** of f, if there exists $\delta > 0$ such that for any $(\mathbf{x}, \mathbf{y})$ satisfying $\|\mathbf{x} - \mathbf{x}^\star\| \leq \delta$ and $\|\mathbf{y} - \mathbf{y}^\star\| \leq \delta$ we have:

$$f(\mathbf{x}^\star, \mathbf{y}) \leq f(\mathbf{x}^\star, \mathbf{y}^\star) \leq f(\mathbf{x}, \mathbf{y}^\star).$$

We can characterize local pure strategy Nash equilibria via first-order and second-order conditions.

Proposition 4.2.4 (First-order Necessary Condition)**.** *Assuming f is differentiable, any local Nash equilibrium satisfies $\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) = 0$ and $\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) = 0$.*

Proposition 4.2.5 (Second-order Necessary Condition)**.** *Assuming f is twice-differentiable, any local Nash equilibrium satisfies $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}) \preceq \mathbf{0}$, and $\nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y}) \succeq \mathbf{0}$.*

Proposition 4.2.6 (Second-order Sufficient Condition)**.** *Assuming f is twice-differentiable, any stationary point (i.e., $\nabla f = 0$) satisfying the following condition is a local Nash equilibrium:*

$$\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}) \prec \mathbf{0}, \text{ and } \nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y}) \succ \mathbf{0}. \tag{4.2}$$

*We also call a stationary point satisfying (4.2) a **strict local Nash equilibrium**.*

In contrast to pure strategies where each player plays a single action, game theorists have also considered mixed strategies where each player is allowed to play a randomized action sampled from a probability measure $\mu \in \mathcal{P}(\mathcal{X})$ or $\nu \in \mathcal{P}(\mathcal{Y})$. Then, the payoff function becomes an expected value $\mathbb{E}_{\mathbf{x}\sim\mu,\mathbf{y}\sim\nu} f(\mathbf{x}, \mathbf{y})$. This corresponds to the scenario where the second player knows the strategy (distribution) of the first player, but does not know the random action he plays. In this setting we define mixed strategy Nash equilibria:

Definition 4.2.7. A probability measure $(\mu^\star, \nu^\star)$ is a **mixed strategy Nash equilibrium** of f, if for any measure (μ, ν) in $\mathcal{P}(\mathcal{X}) \times \mathcal{P}(\mathcal{Y})$, we have

$$\mathbb{E}_{\mathbf{x}\sim\mu^\star,\mathbf{y}\sim\nu} f(\mathbf{x}, \mathbf{y}) \leq \mathbb{E}_{\mathbf{x}\sim\mu^\star,\mathbf{y}\sim\nu^\star} f(\mathbf{x}, \mathbf{y}) \leq \mathbb{E}_{\mathbf{x}\sim\mu,\mathbf{y}\sim\nu^\star} f(\mathbf{x}, \mathbf{y}).$$

Dynamical systems

One of the most popular algorithms for solving minmax problems is Gradient Descent Ascent (GDA). We outline the algorithm in Algorithm 8, with updates written in a general form $\mathbf{z}_{t+1} = \mathbf{w}(\mathbf{z}_t)$, where $\mathbf{w} : \mathbb{R}^d \to \mathbb{R}^d$ is a vector function. One notable distinction from standard gradient descent is that $\mathbf{w}(\cdot)$ may not be a gradient field (i.e., the gradient of a scalar function $\phi(\cdot)$), and so the Jacobian matrix $\mathbf{J} := \partial\mathbf{w}/\partial\mathbf{z}$ may be asymmetric. This results in the possibility of the dynamics $\mathbf{z}_{t+1} = \mathbf{w}(\mathbf{z}_t)$ converging to a limit cycle instead of a single point. Nevertheless, we can still define fixed points and stability for general dynamics.

Definition 4.2.8. $\mathbf{z}^\star$ is a **fixed point** if $\mathbf{z}^\star = \mathbf{w}(\mathbf{z}^\star)$.

Definition 4.2.9 (Linear Stability)**.** For a differentiable dynamical system $\mathbf{w}$, a fixed point $\mathbf{z}^\star$ is a **linearly stable point** of $\mathbf{w}$ if its Jacobian matrix $\mathbf{J}(\mathbf{z}^\star) := (\partial\mathbf{w}/\partial\mathbf{z})(\mathbf{z}^\star)$ has spectral radius $\rho(\mathbf{J}(\mathbf{z}^\star)) \leq 1$. We also say that a fixed point $\mathbf{z}^\star$ is a **strict linearly stable point** if $\rho(\mathbf{J}(\mathbf{z}^\star)) < 1$ and a **strict linearly unstable point** if $\rho(\mathbf{J}(\mathbf{z}^\star)) > 1$.

Intuitively, linear stability captures whether under the dynamics $\mathbf{z}_{t+1} = \mathbf{w}(\mathbf{z}_t)$ a flow that starts at point that is infinitesimally close to $\mathbf{z}^\star$ will remain in a small neighborhood around $\mathbf{z}^\star$.

4.3 What is the Right Objective?

We have introduced three notions of optimality in minmax games: global minmax points (Definition 4.2.1), pure strategy Nash equilibria (Definition 4.2.2) and mixed strategy Nash equilibria (Definition 4.2.7). For convex-concave games, these three notions are essentially identical. However, for nonconvex-nonconcave games, they are all different in general. So, what is the right objective to pursue in this general setting?

Pure strategy Nash equilibrium First, we note that pure strategy Nash equilibria may not exist in nonconvex-nonconcave settings.

Proposition 4.3.1. *There exists a twice-differentiable function f, where pure strategy Nash equilibria (either local or global) do not exist.*

Proof. Consider a two-dimensional function $f(x, y) = \sin(x + y)$. We have $\nabla f(x, y) = (\cos(x + y), \cos(x + y))$. Assuming (x, y) is a local pure strategy Nash equilibrium, by Proposition 4.2.4 it must also be a stationary point; that is, $x + y = (k + 1/2)\pi$ for $k \in \mathbb{Z}$. It is easy to verify, for odd k, $\nabla^2_{xx} f(x, y) = \nabla^2_{yy} f(x, y) = 1 > 0$; for even k, $\nabla^2_{xx} f(x, y) = \nabla^2_{yy} f(x, y) = -1 < 0$. By Proposition 4.2.5, none of the stationary points is a local pure strategy Nash equilibrium. □

Apart from the existence issue, the property that $\mathbf{x}^\star$ is optimal for $f(\cdot, \mathbf{y}^\star)$ is not meaningful in applications such as adversarial training, which translates to the property that the classifier needs to be optimal with respect to a fixed corruption.

Global minmax point On the other hand, global minmax points, a simple but less mentioned notion of optimality, always exist.

Proposition 4.3.2. *Assume that function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is continuous, and assume that $\mathcal{X} \subset \mathbb{R}^{d_1}$, $\mathcal{Y} \subset \mathbb{R}^{d_2}$ are compact. Then the global minmax point (Definition 4.2.1) of f always exists.*

Proposition 4.3.2 is a simple consequence of the extreme-value theorem. Compared to pure strategy Nash equilibria, the notion of global minmax is typically important in the setting where our goal is to find the best $\mathbf{x}^\star$ subject to adversarial perturbation of $\mathbf{y}$ rather than an $\mathbf{x}$ which is optimal for a fixed $\mathbf{y}^\star$. Indeed, both GANs and adversarial training actually fall in this category, where our primary goal is to find the best generator subject to an adversarial discriminator, and to find the best robust classifier subject to adversarial corruption.

Mixed strategy Nash equilibrium Finally, when each agent is allowed to play a random action according to some distribution, such as in the setting of multi-agent reinforcement learning, mixed strategy Nash equilibria are a valid notion of optimality. The existence of mixed strategy Nash equilibrium can be traced back to von (Neumann, 1928). Here we cite a generalized version for continuous games.

Proposition 4.3.3 ((Glicksberg, 1952)). *Assume that the function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is continuous and that $\mathcal{X} \subset \mathbb{R}^{d_1}$, $\mathcal{Y} \subset \mathbb{R}^{d_2}$ are compact. Then*

$$\min_{\mu \in \mathcal{P}(\mathcal{X})} \max_{\nu \in \mathcal{P}(\mathcal{Y})} \mathbb{E}_{(\mu,\nu)} f(\mathbf{x}, \mathbf{y}) = \max_{\nu \in \mathcal{P}(\mathcal{Y})} \min_{\mu \in \mathcal{P}(\mathcal{X})} \mathbb{E}_{(\mu,\nu)} f(\mathbf{x}, \mathbf{y}).$$

Let $\mu^\star$ be the minimum for the minmax problem, and let $\nu^\star$ be the maximum for the maxmin problem. Then $(\mu^\star, \nu^\star)$ is a mixed strategy Nash equilibrium.

In conclusion, both global minmax points and mixed strategy Nash equilibria are well-defined objectives, and of practical interest. For a specific application, which notion is more suitable depends on whether randomized actions are allowed or of interest.

Reduction from mixed strategy nash equilibria to minmax points

We concluded in the last section that both global minmax points and mixed strategy Nash equilibria (or mixed strategies, for short) are of practical interest. However, finding mixed strategy equilibria requires optimizing over a space of probability measures, which is infinite dimensional, making the problem computational infeasible in general. In this section, we show instead how to find approximate mixed strategy Nash equilibria for Lipschitz games. We show that it is sufficient to find a global minmax point of a problem with polynomially large dimension.

Definition 4.3.4. Let $(\mu^\star, \nu^\star)$ be a mixed strategy Nash equilibrium. A probability measure $(\mu^\dagger, \nu^\dagger)$ is an **ϵ-approximate mixed strategy Nash equilibrium** if:

$$\begin{aligned} \forall \nu' \in \mathcal{P}(\mathcal{Y}), \quad & \mathbb{E}_{(\mu^\dagger, \nu')} f(\mathbf{x}, \mathbf{y}) \leq \mathbb{E}_{(\mu^\star, \nu^\star)} f(\mathbf{x}, \mathbf{y}) + \epsilon \\ \forall \mu' \in \mathcal{P}(\mathcal{Y}), \quad & \mathbb{E}_{(\mu', \nu^\dagger)} f(\mathbf{x}, \mathbf{y}) \geq \mathbb{E}_{(\mu^\star, \nu^\star)} f(\mathbf{x}, \mathbf{y}) - \epsilon. \end{aligned}$$

Theorem 4.3.5. *Assume that function f is L-Lipschitz, and the diameters of $\mathcal{X}$ and $\mathcal{Y}$ are at most D. Let $(\mu^\star, \nu^\star)$ be a mixed strategy Nash equilibrium. Then there exists an absolute constant c, for any $\epsilon > 0$, such that if $N \geq c \cdot d_2 (LD/\epsilon)^2 \log(LD/\epsilon)$, we have:*

$$\min_{(\mathbf{x}_1, \ldots, \mathbf{x}_N) \in \mathcal{X}^N} \max_{\mathbf{y} \in \mathcal{Y}} \frac{1}{N} \sum_{i=1}^{N} f(\mathbf{x}_i, \mathbf{y}) \leq \mathbb{E}_{(\mu^\star, \nu^\star)} f(\mathbf{x}, \mathbf{y}) + \epsilon.$$

Intuitively, Theorem 4.3.5 holds because function f is Lipschitz, $\mathcal{Y}$ is a bounded domain, and thus we can establish uniform convergence of the expectation of $f(\cdot, \mathbf{y})$ to its average over N samples for all $\mathbf{y} \in \mathcal{Y}$ simultaneously. A similar argument was made in (Arora et al., 2017).

Theorem 4.3.5 implies that in order to find a approximate mixed strategy Nash equilibrium, we can solve a large minmax problem with objective $F(\mathbf{X}, \mathbf{y}) := \sum_{i=1}^{N} f(\mathbf{x}_i, \mathbf{y})/N$. The global minmax solution $\mathbf{X}^\star = (\mathbf{x}_1^\star, \ldots, \mathbf{x}_n^\star)$ gives a empirical distribution $\hat{\mu}^\star = \sum_{i=1}^{N} \delta(\mathbf{x} - \mathbf{x}_i^\star)/N$, where $\delta(\cdot)$ is the Dirac delta function. By symmetry, we can also solve the maxmin problem to find $\hat{\nu}^\star$. Since optimal pure strategies are always as good as optimal mixed strategies for the second player, we know $(\hat{\mu}^\star, \hat{\nu}^\star)$ is an ϵ-approximate mixed strategy Nash equilibrium. That is, approximate mixed strategy Nash can be found by finding two global minmax points.

4.4 Main Results

In the previous section, we concluded that the central question in minmax optimization is to find a global minmax point. However, the problem of finding global minmax points is in general NP hard. In this section, we present our main results, suggesting possible ways of circumventing this NP-hardness challenge. In Section 4.4, we develop a new notion of local surrogacy for global minmax points which we refer to as *local minmax points*, and we study their properties. In Section 4.4, we establish relations between stable fixed points of GDA and local minmax points. In Section 4.4, we study the behavior of gradient descent with an approximate maximization oracle for $\mathbf{y}$ and show that it converges to approximately stationary points of $\max_{\mathbf{y}} f(\cdot, \mathbf{y})$.

Local minmax points

While most previous work (Daskalakis and Panageas, 2018; Mazumdar and Ratliff, 2018) has focused on local Nash equilibria (Definition 4.2.3), which are local surrogates for pure strategy Nash equilibria, we propose a new notion—*local minmax*—as a natural local surrogate for global minmaxity. To the best of our knowledge, this notion has not been considered before.

Definition 4.4.1. A point $(\mathbf{x}^\star, \mathbf{y}^\star)$ is said to be a **local minmax point** of f, if there exists $\delta_0 > 0$ and a continuous function h satisfying $h(\delta) \to 0$ as $\delta \to 0$, such that for any $\delta \leq \delta_0$, and any $(\mathbf{x}, \mathbf{y})$ satisfying $\|\mathbf{x} - \mathbf{x}^\star\| \leq \delta$ and $\|\mathbf{y} - \mathbf{y}^\star\| \leq \delta$, we have

$$f(\mathbf{x}^\star, \mathbf{y}) \leq f(\mathbf{x}^\star, \mathbf{y}^\star) \leq \max_{\mathbf{y}' : \|\mathbf{y}' - \mathbf{y}^\star\| \leq h(\delta)} f(\mathbf{x}, \mathbf{y}'). \tag{4.3}$$

A notion of local maxmin point can be defined similarly. Local minmax points are different from local Nash equilibria since local minmax points only require $\mathbf{x}^\star$ to be the minimum of a local max function $\max_{\mathbf{y}' : \|\mathbf{y}' - \mathbf{y}^\star\| \leq h(\delta)} f(\cdot, \mathbf{y}')$, while local Nash equilibria require $\mathbf{x}^*$ to be the local minimum after fixing $\mathbf{y}^*$ (see Figure 4.1). The local radius $h(\delta)$ over which

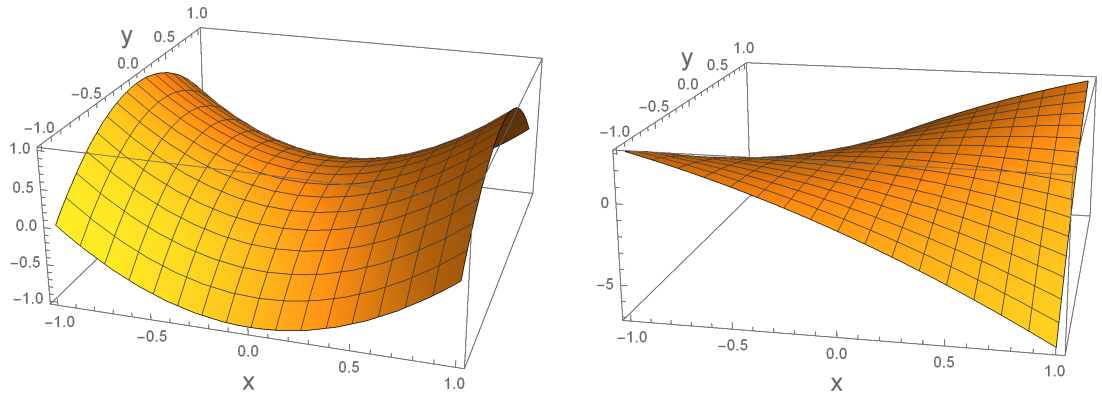

Figure 4.1: **Left:** $f(x, y) = x^2 - y^2$ where $(0, 0)$ is both local Nash and local minmax. **Right:** $f(x, y) = -x^2 + 5xy - y^2$ where $(0, 0)$ is not local Nash but local minmax with $h(\delta) = \delta$.

the maximum is taken needs to decrease to zero as δ approaches zero. We note that our definition does not control the relative rate at which $h(\delta)$ and δ go to zero; indeed, it is allowed that $\lim_{\delta \to 0} h(\delta)/\delta = \infty$.

We would like to highlight an interesting fact: in minmax optimization, global minmax can be neither local minmax nor stationary points (and thus not local Nash equilibria). This is in contrast to the well-known fact in nonconvex optimization where global minima are always local minima.

Proposition 4.4.2. *The global minmax point can be neither local minmax nor a stationary point.*

See Figure 4.2 for an illustration and Appendix 4.7 for the proof. The proposition is a natural consequece of the definitions where global minmax points are obtained as a minimum of a *global* maximum function while local minmax points are the minimum of a *local* maximum function. This also illustrates that minmax optimization is a challenging task, and worthy of independent study, beyond nonconvex optimization.

Nevertheless, global minmax points can be guaranteed to be local minmax if the problem has some structure. For instance, this is true when f is strongly-concave in $\mathbf{y}$, or more generally when f satisfies the following properties that have been established to hold in several popular machine learning problems (Ge, Jin, and Zheng, 2017; Boumal, Voroninski, and Bandeira, 2016):

Theorem 4.4.3. *Assume that f is twice differentiable, and for any fixed $\mathbf{x}$, the function $f(\mathbf{x}, \cdot)$ is strongly concave in the neighborhood of local maxima and satisfies the assumption that all local maxima are global maxima. Then the global minmax point of $f(\cdot, \cdot)$ is also a local minmax point.*

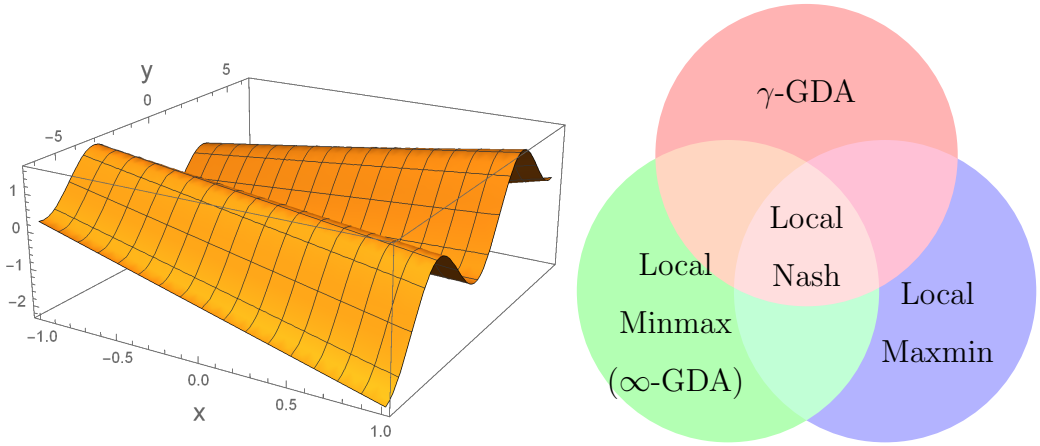

Figure 4.2: **Left:** $f(x, y) = 0.2xy - cos(y)$, the global minmax points $(0, -\pi)$ and $(0, \pi)$ are not stationary. **Right:** The relations among local Nash equilibria, local minmax points, local maxmin points and linearly stable points of γ-GDA, and ∞-GDA (up to degenerate points).

We consider local minmax as a more suitable notion of local optimality than local Nash equilibrium for minmax optimization. First, local minmaxity is a strictly relaxed notion of local Nash equilibrium, and it alleviates the non-existence issue for local Nash equilibria.

Proposition 4.4.4. *Any local pure strategy Nash equilibrium is a local minmax point.*

Second, local minmax points enjoy simple first-order and second-order characterizations.

Proposition 4.4.5 (First-order Necessary Condition)**.** *Assuming that f is continuously differentiable, then any local minmax point $(\mathbf{x}, \mathbf{y})$ satisfies $\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) = 0$ and $\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) = 0$.*

Proposition 4.4.6 (Second-order Necessary Condition)**.** *Assuming that f is twice differentiable, then $(\mathbf{x}, \mathbf{y})$ is a local minmax point implies that $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}) \preceq \mathbf{0}$, and for any $\mathbf{v}$ satisfying $\nabla^2_{\mathbf{yx}} f(\mathbf{x}, \mathbf{y}) \cdot \mathbf{v} \in \text{column\_span}(\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}))$ that $\mathbf{v}^\top [\nabla^2_{\mathbf{xx}} f - \nabla^2_{\mathbf{xy}} f (\nabla^2_{\mathbf{yy}} f)^\dagger \nabla^2_{\mathbf{yx}} f](\mathbf{x}, \mathbf{y}) \cdot \mathbf{v} \geq \mathbf{0}$. (Here † denotes Moore-Penrose inverse.)*

Proposition 4.4.7 (Second-order Sufficient Condition)**.** *Assume that f is twice differentiable. Any stationary point $(\mathbf{x}, \mathbf{y})$ satisfying $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}) \prec \mathbf{0}$ and*

$$[\nabla^2_{\mathbf{xx}} f - \nabla^2_{\mathbf{xy}} f (\nabla^2_{\mathbf{yy}} f)^{-1} \nabla^2_{\mathbf{yx}} f](\mathbf{x}, \mathbf{y}) \succ \mathbf{0} \tag{4.4}$$

is a local minmax point. We call stationary points satisfying (4.4) strict local minmax points.

Algorithm 8 Gradient Descent Ascent (γ-GDA)

Input: $(\mathbf{x}_0, \mathbf{y}_0)$, step size η, ratio γ.
  for $t = 0, 1, \ldots,$ **do**
    $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - (\eta/\gamma)\nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t)$.
    $\mathbf{y}_{t+1} \leftarrow \mathbf{y}_t + \eta \nabla_{\mathbf{y}} f(\mathbf{x}_t, \mathbf{y}_t)$.

We note that if $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$ is non-degenerate, then the second-order necessary condition (Proposition 4.4.6) becomes $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}) \prec \mathbf{0}$ and $[\nabla^2_{\mathbf{xx}} f - \nabla^2_{\mathbf{xy}} f (\nabla^2_{\mathbf{yy}} f)^{-1} \nabla^2_{\mathbf{yx}} f](\mathbf{x}, \mathbf{y}) \succeq \mathbf{0}$, which is identical to the sufficient condition Eq.(4.4) up to an equals sign.

Comparing Eq. (4.4) to the second-order sufficient condition for local Nash equilibrium in Eq. (4.2), we see that local minmax requires the Shur complement to be positive definite instead of requiring $\nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y})$ to be positive definite. Contrary to local Nash equilibria, this characterization of local minmax not only takes into account the interaction term $\nabla^2_{\mathbf{xy}} f$ between $\mathbf{x}$ and $\mathbf{y}$, but also reflects the order of minmax vs maxmin.

Limit points of gradient descent ascent

In this section, we consider the asymptotic behavior of Gradient Descent Ascent (GDA). As shown in the pseudo-code in Algorithm 8, GDA simultaneously performs gradient descent on $\mathbf{x}$ and gradient ascent on $\mathbf{y}$. We consider the general form where the step size for $\mathbf{x}$ can be different from the step size for $\mathbf{y}$ by a ratio γ, and denoted this algorithm by γ-GDA. When the step size η is small, this is essentially equivalent to gradient descent with multiple steps of gradient ascent where γ indicates how many gradient ascent steps are performed for one gradient descent step.

To study the limiting behavior, we primarily focus on linearly stable points of γ-GDA, since with random initialization, γ-GDA will almost surely escape strict linearly unstable points.

Theorem 4.4.8 ((Daskalakis and Panageas, 2018))**.** *For any $\gamma > 1$, assuming the function f is ℓ-gradient Lipschitz, and the step size $\eta \leq 1/\ell$, then the set of initial points $\mathbf{x}_0$ so that γ-GDA converges to its strict linear unstable point is of Lebesgue measure zero.*

We further simplifiy the problem by considering the limiting case where the step size $\eta \to 0$, which corresponds to γ-GDA flow

$$\frac{\mathrm{d}\mathbf{x}}{\mathrm{d}t} = -\frac{1}{\gamma} \nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) \qquad \frac{\mathrm{d}\mathbf{y}}{\mathrm{d}t} = \nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}).$$

The strict linearly stable points of the γ-GDA flow have a very simple second-order characterization.

Proposition 4.4.9. $(\mathbf{x}, \mathbf{y})$ *is a strict linearly stable point of* γ*-GDA if and only if for all the eigenvalues* $\{\lambda_i\}$ *of following Jacobian matrix,*

$$\mathbf{J}_\gamma = \begin{pmatrix} -(1/\gamma)\nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y}) & -(1/\gamma)\nabla^2_{\mathbf{xy}} f(\mathbf{x}, \mathbf{y}) \\ \nabla^2_{\mathbf{yx}} f(\mathbf{x}, \mathbf{y}) & \nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}), \end{pmatrix}$$

their real part $\Re(\lambda_i) < 0$ *for any* i*.*

In the remainder of this section, we assume that f is a twice-differentiable function, and we use $\mathit{Local\_Nash}$ to represent the set of strict local Nash equilibria, $\mathit{Local\_Minmax}$ for the set of strict local minmax points, $\mathit{Local\_Maxmin}$ for the set of strict local maxmin points, and $\gamma-\mathcal{GDA}$ for the set of strict linearly stable points of the γ-GDA flow. Our goal is to understand the relationships between these sets. Daskalakis and Panageas (2018) and Mazumdar and Ratliff (2018) provided a relation between $\mathit{Local\_Nash}$ and $1-\mathcal{GDA}$ which can be generalized to $\gamma-\mathcal{GDA}$ as follows.

Proposition 4.4.10 ((Daskalakis and Panageas, 2018))**.** *For any fixed* γ*, for any twice-differentiable* f*,* $\mathit{Local\_Nash} \subset \gamma-\mathcal{GDA}$*, but there exist twice-differentiable* f *such that* $\gamma-\mathcal{GDA} \not\subset \mathit{Local\_Nash}$*.*

That is, if γ-GDA converges, it may converge to points not in $\mathit{Local\_Nash}$. This raises a basic question as to what those additional stable limit points of $\gamma-$GDA are. Are they meaningful? This work answers this question through the lens of $\mathit{Local\_Minmax}$. Although for fixed γ, the set $\gamma-\mathcal{GDA}$ does not have a simple relation with $\mathit{Local\_Minmax}$, it turns out that an important relationship arises when γ goes to ∞. To describe the limit behavior of the set $\gamma-\mathcal{GDA}$ when $\gamma \to \infty$ we define two set-theoretic limits:

$$\overline{\infty-\mathcal{GDA}} := \limsup_{\gamma\to\infty} \gamma-\mathcal{GDA} = \cap_{\gamma_0>0} \cup_{\gamma>\gamma_0} \gamma-\mathcal{GDA}$$
$$\underline{\infty-\mathcal{GDA}} := \liminf_{\gamma\to\infty} \gamma-\mathcal{GDA} = \cup_{\gamma_0>0} \cap_{\gamma>\gamma_0} \gamma-\mathcal{GDA}.$$

The relations between $\gamma-\mathcal{GDA}$ and $\mathit{Local\_Minmax}$ are given as follows:

Proposition 4.4.11. *For any fixed* γ*, there exists twice-differentiable* f *such that* $\mathit{Local\_Minmax} \not\subset \gamma-\mathcal{GDA}$*; there also exists twice-differentiable* f *such that* $\gamma-\mathcal{GDA} \not\subset \mathit{Local\_Minmax} \cup \mathit{Local\_Maxmin}$*.*

Theorem 4.4.12 (Main Theorem)**.** *For any twice-differentiable* f*,* $\mathit{Local\_Minmax} \subset \underline{\infty-\mathcal{GDA}} \subset \overline{\infty-\mathcal{GDA}} \subset \mathit{Local\_Minmax} \cup \{(\mathbf{x}, \mathbf{y}) | (\mathbf{x}, \mathbf{y})$ *is stationary and* $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$ *is degenerate*$\}$*.*

That is, $\infty-\mathcal{GDA} = \mathit{Local\_Minmax}$ up to some degenerate points. Intuitively, when γ is large, γ-GDA can move a long distance in $\mathbf{y}$ while only making very small changes in $\mathbf{x}$. As $\gamma \to \infty$, γ-GDA can approximately find the local maximum of $f(\mathbf{x} + \delta_{\mathbf{x}}, \cdot)$, subject to any small change in $\delta_{\mathbf{x}}$; therefore, stable limit points are indeed local minmax.

Algorithm 9 Gradient Descent with Max-oracle

Input: $\mathbf{x}_0$, step size η.
  for $t = 0, 1, \ldots, T$ **do**
    find $\mathbf{y}_t$ so that $f(\mathbf{x}_t, \mathbf{y}_t) \geq \max_{\mathbf{y}} f(\mathbf{x}_t, \mathbf{y}) - \epsilon$.
    $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t - \eta \nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t)$.
  Pick t uniformly at random from $\{0, \cdots, T\}$.
  return $\bar{\mathbf{x}} \leftarrow \mathbf{x}_t$.

Algorithmically, one can view $\infty - \mathcal{GDA}$ as a set that discribes the strict linear stable limit points for GDA with γ very slowly increasing with respect to t, and eventually going to ∞. To the best of our knowledge, this is the first result showing that all stable limit points of GDA are meaningful and locally optimal up to some degenerate points.

Gradient descent with max-oracle

In this section, we consider solving the minmax problem when we have access to an oracle for approximate inner maximization; i.e., for any $\mathbf{x}$, we have access to an oracle that outputs a $\widehat{\mathbf{y}}$ such that $f(\mathbf{x}, \widehat{\mathbf{y}}) \geq \max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) - \epsilon$. A natural algorithm to consider in this setting is to alternate between gradient descent on $\mathbf{x}$ and a (approximate) maximization step on $\mathbf{y}$. The pseudocode is presented in Algorithm 9.

It can be shown that Algorithm 9 indeed converges (in contrast with GDA which can converge to limit cycles). Moreover, the limit points of Algorithm 9 satisfy a nice property—they turn out to be approximately stationary points of $\phi(\mathbf{x}) := \max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$. For a smooth function, "approximately stationary point" means that the norm of gradient is small. However, even when $f(\cdot, \cdot)$ is smooth (up to whatever order), $\phi(\cdot)$ as defined above need not be differentiable. The norm of subgradient can be a discontinuous function which is an undesirable measure for closeness to stationarity. Fortunately, however, and ℓ-gradient Lipschitz of $f(\cdot, \cdot)$ imply that $\phi(\cdot)$ is ℓ-*weakly convex* (Rafique et al., 2018); i.e., $\phi(\mathbf{x}) + (\ell/2)\|\mathbf{x}\|^2$ is convex. In such settings, the approximate stationarity of $\phi(\cdot)$ can be measured by the norm of gradient of its Moreau envelope $\phi_\lambda(\cdot)$.

$$\phi_\lambda(\mathbf{x}) := \min_{\mathbf{x}'} \phi(\mathbf{x}') + \frac{1}{2\lambda}\|\mathbf{x} - \mathbf{x}'\|^2. \tag{4.5}$$

Here $\lambda < 1/\ell$. The Moreau envelope satisfies the following two important properties if $\lambda < 1/\ell$. Let $\hat{\mathbf{x}} = \operatorname{argmin}_{\mathbf{x}'} \phi(\mathbf{x}') + (1/2\lambda)\|\mathbf{x} - \mathbf{x}'\|^2$, then:

$$\|\hat{\mathbf{x}} - \mathbf{x}\| = \lambda \|\nabla \phi_\lambda(\mathbf{x})\|, \quad \text{and} \quad \min_{\mathbf{g} \in \partial \phi(\hat{\mathbf{x}})} \|\mathbf{g}\| \leq \|\nabla \phi_\lambda(\mathbf{x})\|,$$

where ∂ denotes the subdifferential of a weakly convex function. A proof of this fact can be found in (Rockafellar, 2015). Therefore, $\|\nabla \phi_\lambda(\mathbf{x})\|$ being small means that $\mathbf{x}$ is close to a point $\hat{\mathbf{x}}$ that is approximately stationary. We now present the convergence guarantee for Algorithm 9.

Theorem 4.4.13. *Suppose f is ℓ-smooth and L-Lipschitz and define $\phi(\cdot) := \max_{\mathbf{y}} f(\cdot, \mathbf{y})$. Then the output $\bar{\mathbf{x}}$ of GD with Max-oracle (Algorithm 9) with step size $\eta = \gamma/\sqrt{T+1}$ will satisfy*

$$\mathbb{E}\left[\|\nabla \phi_{1/2\ell}(\bar{\mathbf{x}})\|^2\right] \leq 2 \cdot \frac{\left(\phi_{1/2\ell}(\mathbf{x}_0) - \min \phi(\mathbf{x})\right) + \ell L^2 \gamma^2}{\gamma\sqrt{T+1}} + 4\ell\epsilon,$$

where $\phi_{1/2\ell}$ is the Moreau envelope (4.5) of ϕ.

The proof of Theorem 4.4.13 is similar to the convergence analysis for nonsmooth weakly-convex functions (Davis and Drusvyatskiy, 2018), except here the max-oracle has error ϵ. Theorem 4.4.13 claims, other than an additive error $4\ell\epsilon$ as a result of the oracle solving the maximum approximately, that the remaining term decreases at a rate of $1/\sqrt{T}$.

4.5 Conclusion

In this work, we consider general nonconvex-nonconcave minmax optimization. While gradient descent ascent (GDA) is widely used in practice for such problems, previous results suggest that GDA has undesirable limiting behavior, questioning GDA's relevance for this problem. We formulate a new notion of local optimum for minmax problems, which we refer to as *local minmax*, and show that it is more suitable for many learning problems than standard notions such as local Nash equilibrium. We establish that as the ratio of the ascent step size to the descent step size in GDA goes to infinity, all strict stable limit points are equivalent to local minmax points except for degenerate points. This yields a game-theoretic meaning for all stable limit points of GDA. We also consider the minmax problem when we have access to an approximate inner maximization. In this setting, we analyze gradient descent with maximization and show that it finds a point close to an approximate stationary point.

4.6 Proofs for Reduction from Mixed Strategy Nash to Minmax Points

In this section we prove Theorem 4.3.5 in Section 4.3.

Theorem 4.3.5. *Assume that function f is L-Lipschitz, and the diameters of $\mathcal{X}$ and $\mathcal{Y}$ are at most D. Let $(\mu^\star, \nu^\star)$ be a mixed strategy Nash equilibrium. Then there exists an absolute constant c, for any $\epsilon > 0$, such that if $N \geq c \cdot d_2(LD/\epsilon)^2 \log(LD/\epsilon)$, we have:*

$$\min_{(\mathbf{x}_1,\ldots,\mathbf{x}_N)\in\mathcal{X}^N} \max_{\mathbf{y}\in\mathcal{Y}} \frac{1}{N}\sum_{i=1}^{N} f(\mathbf{x}_i, \mathbf{y}) \leq \mathbb{E}_{(\mu^\star,\nu^\star)} f(\mathbf{x}, \mathbf{y}) + \epsilon.$$

Proof. Note that WLOG, the second player can always play pure strategy. That is,

$$\min_{\mu\in\mathcal{P}(\mathcal{X})}\max_{\nu\in\mathcal{P}(\mathcal{Y})}\mathbb{E}_{\mathbf{x}\sim\mu,\mathbf{y}\sim\nu}f(\mathbf{x},\mathbf{y})=\min_{\mu\in\mathcal{P}(\mathcal{X})}\max_{\mathbf{y}\in\mathcal{Y}}\mathbb{E}_{\mathbf{x}\sim\mu}f(\mathbf{x},\mathbf{y})$$

Therefore, we only need to solve the problem of RHS. Suppose the minimum over $\mathcal{P}(\mathcal{X})$ is achieved at $\mu^\star$. First, sample $(\mathbf{x}_1,\ldots,\mathbf{x}_N)$ i.i.d from $\mu^\star$, and note $\max_{\mathbf{x}_1,\mathbf{x}_2\in\mathcal{X}}|f(\mathbf{x}_1,\mathbf{y})-f(\mathbf{x}_2,\mathbf{y})|\le LD$ for any fixed $\mathbf{y}$. Therefore by Hoeffding inequality, for any fixed $\mathbf{y}$:

$$\mathbb{P}\left(\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y})-\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y})\ge t\right)\le e^{-\frac{Nt^2}{(LD)^2}}$$

Let $\bar{\mathcal{Y}}$ be a minimal $\epsilon/(2L)$-covering over $\mathcal{Y}$. We know the covering number $|\bar{\mathcal{Y}}|\le(2DL/\epsilon)^d$. Thus by union bound:

$$\mathbb{P}\left(\forall\mathbf{y}\in\bar{\mathcal{Y}},\ \frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y})-\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y})\ge t\right)\le e^{d\log\frac{2DL}{\epsilon}-\frac{Nt^2}{(LD)^2}}$$

Pick $t=\epsilon/2$ and $N\ge c\cdot d(LD/\epsilon)^2\log(LD/\epsilon)$ for some large absolute constant c, we have:

$$\mathbb{P}\left(\forall\mathbf{y}\in\bar{\mathcal{Y}},\ \frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y})-\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y})\ge\frac{\epsilon}{2}\right)\le\frac{1}{2}$$

Let $\mathbf{y}^\star=\arg\max_{\mathbf{y}}\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y})$, by definition of covering, we can always find a $\mathbf{y}'\in\bar{\mathcal{Y}}$ so that $\|\mathbf{y}^\star-\mathbf{y}'\|\le\epsilon/(4L)$. Thus, with probability at least $1/2$:

$$\begin{aligned}&\max_{\mathbf{y}}\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y})-\max_{\mathbf{y}}\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y})=\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y}^\star)-\max_{\mathbf{y}}\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y})\\ \le&\left[\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y}^\star)-\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y}')\right]+\left[\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y}')-\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y}')\right]\\ &+\left[\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y}')-\max_{\mathbf{y}}\mathbb{E}_{\mathbf{x}\sim\mu^\star}f(\mathbf{x},\mathbf{y})\right]\le\epsilon/2+\epsilon/2+0\le\epsilon\end{aligned}$$

That is, with probability at least $1/2$:

$$\max_{\mathbf{y}\in\mathcal{Y}}\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y})\le\min_{\mu\in\mathcal{P}(\mathcal{X})}\max_{\mathbf{y}\in\mathcal{Y}}\mathbb{E}_{\mathbf{x}\sim\mu}f(\mathbf{x},\mathbf{y})+\epsilon$$

This implies:

$$\min_{(\mathbf{x}_1,\ldots,\mathbf{x}_N)\in\mathcal{X}^N}\max_{\mathbf{y}\in\mathcal{Y}}\frac{1}{N}\sum_{i=1}^N f(\mathbf{x}_i,\mathbf{y})\le\min_{\mu\in\mathcal{P}(\mathcal{X})}\max_{\mathbf{y}\in\mathcal{Y}}\mathbb{E}_{\mathbf{x}\sim\mu}f(\mathbf{x},\mathbf{y})+\epsilon$$

Combine with Proposition 4.3.3, we finish the proof. □

4.7 Proofs for Properties of Local Minmax Points

In this section, we prove the propositions and theorems presented in Section 4.4.

Proposition 4.4.2. *The global minmax point can be neither local minmax nor a stationary point.*

Proof. Consider function $f(x, y) = 0.2xy - \cos(y)$ in region $[-1, 1] \times [-2\pi, 2\pi]$ as shown in Figure 4.2. Clearly, the gradient is equal to $(0.2y, 0.2x + \sin(y))$. And for any fixed x, there are only two maxima $y^\star(x)$ satisfying $0.2x + \sin(y^\star) = 0$ where $y_1^\star(x) \in (-3\pi/2, -\pi/2)$ and $y_2^\star(x) \in (\pi/2, 3\pi/2)$. On the other hand, $f(x, y_1^\star(x))$ is monotonically decreasing with respect to x, while $f(x, y_2^\star(x))$ is monotonically increasing, with $f(0, y_1^\star(0)) = f(0, y_2^\star(0))$ by symmetry. It is not hard to check $y_1^\star(0) = -\pi$ and $y_2^\star(0) = \pi$. Therefore, $(0, -\pi)$ and $(0, \pi)$ are two global solutions of minmax problem. However, the gradients at both points are not 0, thus they are not stationary points. By Proposition 4.4.5 they are also not local minmax points. □

Theorem 4.4.3. *Assume that f is twice differentiable, and for any fixed $\mathbf{x}$, the function $f(\mathbf{x}, \cdot)$ is strongly concave in the neighborhood of local maxima and satisfies the assumption that all local maxima are global maxima. Then the global minmax point of $f(\cdot, \cdot)$ is also a local minmax point.*

Proof. Denote $\mathbf{A} := \nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y})$, $\mathbf{B} := \nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$, $\mathbf{C} := \nabla^2_{\mathbf{xy}} f(\mathbf{x}, \mathbf{y})$, $\mathbf{g_x} := \nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y})$ and $\mathbf{g_y} := \nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$. Let $(\mathbf{x}, \mathbf{y})$ be a global minmax point. Since $\mathbf{y}$ is the global argmax of $f(\mathbf{x}, \cdot)$ and locally strongly concave, we know $\mathbf{g_y} = 0$ and $\mathbf{B} \prec 0$. Let us now consider a second-order Taylor approximation of f around $(\mathbf{x}, \mathbf{y})$.

$$f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}) = f(\mathbf{x}, \mathbf{y}) + \mathbf{g_x}^\top \delta_{\mathbf{x}} + \frac{1}{2}\delta_{\mathbf{x}}^\top \mathbf{A} \delta_{\mathbf{x}} + \delta_{\mathbf{x}}^\top \mathbf{C} \delta_{\mathbf{y}} + \frac{1}{2}\delta_{\mathbf{y}}^\top \mathbf{B} \delta_{\mathbf{y}} + o(\|\delta_{\mathbf{x}}\|^2 + \|\delta_{\mathbf{y}}\|^2)$$

Since by hypothesis, $\mathbf{B} \prec 0$, we see that when $\|\delta_x\|$ is sufficiently small, there is a unique $\delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})$ so that $\mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})$ is a local maximum of $f(\mathbf{x} + \delta_{\mathbf{x}}, \cdot)$, where $\delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}}) = -\mathbf{B}^{-1}\mathbf{C}^\top \delta_{\mathbf{x}} + o(\|\delta_{\mathbf{x}}\|)$. It is clear that $\|\delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})\| \leq (\|\mathbf{B}^{-1}\mathbf{C}^\top\| + 1)\|\delta_{\mathbf{x}}\|$ for sufficiently small $\|\delta_{\mathbf{x}}\|$. Let $h(\delta) = (\|\mathbf{B}^{-1}\mathbf{C}^\top\| + 1)\delta$, we know for small enough δ:

$$f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) = \max_{\|\delta_{\mathbf{y}}\| \leq h(\delta)} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}})$$

Finally, since by assumption for any $f(\mathbf{x}, \cdot)$ all local maxima are global maxima and $\mathbf{x}$ is the global min of $\max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$, we know:

$$f(\mathbf{x}, \mathbf{y}) \leq \max_{\mathbf{y}'} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y}') = f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) = \max_{\|\delta_{\mathbf{y}}\| \leq h(\delta)} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}})$$

which finishes the proof.

□

Proposition 4.4.4. *Any local pure strategy Nash equilibrium is a local minmax point.*

Proof. Let h be the constant function $h(\delta) = 0$ for any δ. Suppose $(\mathbf{x}^\star, \mathbf{y}^\star)$ is a local pure strategy Nash equilibrium, by definition it implies the existence of δ_0, so that for any $\delta \le \delta_0$, and any $(\mathbf{x}, \mathbf{y})$ satisfying $\|\mathbf{x} - \mathbf{x}^\star\| \le \delta$ and $\|\mathbf{y} - \mathbf{y}^\star\| \le \delta$:

$$f_2(\mathbf{x}^\star, \mathbf{y}) \le f_2(\mathbf{x}^\star, \mathbf{y}^\star) \le f(\mathbf{x}, \mathbf{y}^\star) \le \max_{\mathbf{y}' : \|\mathbf{y}' - \mathbf{y}^\star\| \le h(\delta)} f_2(\mathbf{x}, \mathbf{y}').$$

which finishes the proof. □

Proposition 4.4.5 (First-order Necessary Condition)**.** *Assuming that f is continuously differentiable, then any local minmax point $(\mathbf{x}, \mathbf{y})$ satisfies $\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) = 0$ and $\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) = 0$.*

Proof. Since $\mathbf{y}$ is the local maximum of $f(\mathbf{x}, \cdot)$, it implies $\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}) = 0$. Denote local optima $\delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}}) := \mathrm{argmax}_{\|\delta_{\mathbf{y}}\| \le h(\delta)} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}})$. By definition we know, $\|\delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})\| \le h(\delta) \to 0$ as $\delta \to 0$. Thus

$$\begin{aligned}
0 \le & f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) - f(\mathbf{x}, \mathbf{y}) \\
= & f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) - f(\mathbf{x}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) + f(\mathbf{x}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) - f(\mathbf{x}, \mathbf{y}) \\
\le & f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) - f(\mathbf{x}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}})) \\
= & \nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y} + \delta_{\mathbf{y}}^\star(\delta_{\mathbf{x}}))^\top \delta_{\mathbf{x}} + o(\|\delta_{\mathbf{x}}\|) = \nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y})^\top \delta_{\mathbf{x}} + o(\|\delta_{\mathbf{x}}\|)
\end{aligned}$$

holds for any small $\delta_{\mathbf{x}}$, which implies $\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}) = 0$. □

Proposition 4.4.6 (Second-order Necessary Condition)**.** *Assuming that f is twice differentiable, then $(\mathbf{x}, \mathbf{y})$ is a local minmax point implies that $\nabla_{\mathbf{yy}}^2 f(\mathbf{x}, \mathbf{y}) \preceq \mathbf{0}$, and for any $\mathbf{v}$ satisfying $\nabla_{\mathbf{yx}}^2 f(\mathbf{x}, \mathbf{y}) \cdot \mathbf{v} \in \mathrm{column\_span}(\nabla_{\mathbf{yy}}^2 f(\mathbf{x}, \mathbf{y}))$ that $\mathbf{v}^\top [\nabla_{\mathbf{xx}}^2 f - \nabla_{\mathbf{xy}}^2 f (\nabla_{\mathbf{yy}}^2 f)^\dagger \nabla_{\mathbf{yx}}^2 f](\mathbf{x}, \mathbf{y}) \cdot \mathbf{v} \ge \mathbf{0}$. (Here † denotes Moore-Penrose inverse.)*

Proof. Denote $\mathbf{A} := \nabla_{\mathbf{xx}}^2 f(\mathbf{x}, \mathbf{y})$, $\mathbf{B} := \nabla_{\mathbf{yy}}^2 f(\mathbf{x}, \mathbf{y})$ and $\mathbf{C} := \nabla_{\mathbf{xy}}^2 f(\mathbf{x}, \mathbf{y})$. Since $\mathbf{y}$ is the local maximum of $f(\mathbf{x}, \cdot)$, it implies $\mathbf{B} \preceq 0$. On the other hand,

$$f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}) = f(\mathbf{x}, \mathbf{y}) + \frac{1}{2} \delta_{\mathbf{x}}^\top \mathbf{A} \delta_{\mathbf{x}} + \delta_{\mathbf{x}}^\top \mathbf{C} \delta_{\mathbf{y}} + \frac{1}{2} \delta_{\mathbf{y}}^\top \mathbf{B} \delta_{\mathbf{y}} + o(\|\delta_{\mathbf{x}}\|^2 + \|\delta_{\mathbf{y}}\|^2).$$

Since $(\mathbf{x}, \mathbf{y})$ is a local minmax point, by definition, there exists a function h such that Eq.(4.3) holds. Denote $h'(\delta) = 2\|\mathbf{B}^{-1}\mathbf{C}^\top\|\delta$. We note both $h(\delta)$ and $h'(\delta) \to 0$ as $\delta \to 0$. For any $\delta_{\mathbf{x}}$ satisfying $\mathbf{C}^\top \delta_{\mathbf{x}} \in$ *column_span*$(\mathbf{B})$, it is not hard to verify that $\mathrm{argmax}_{\|\delta_{\mathbf{y}}\| \le \max(h(\delta), h'(\delta))} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}) = -\mathbf{B}^\dagger \mathbf{C}^\top \delta_{\mathbf{x}} + o(\|\delta_{\mathbf{x}}\|)$. Since $(\mathbf{x}, \mathbf{y})$ is a local minmax point, we have

$$\begin{aligned}
0 \le & \max_{\|\delta_{\mathbf{y}}\| \le h(\delta)} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}) - f(\mathbf{x}, \mathbf{y}) \le \max_{\|\delta_y\| \le \max(h(\delta), h'(\delta))} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}) - f(\mathbf{x}, \mathbf{y}) \\
= & \frac{1}{2} \delta_{\mathbf{x}}^\top (\mathbf{A} - \mathbf{C}\mathbf{B}^\dagger \mathbf{C}^\top) \delta_{\mathbf{x}} + o(\|\delta_{\mathbf{x}}\|^2).
\end{aligned}$$

Above equation holds for any $\delta_{\mathbf{x}}$ satisfying $\mathbf{C}^\top \delta_{\mathbf{x}} \in$ *column_span*$(\mathbf{B})$, which finishes the proof. □

Proposition 4.4.7 (Second-order Sufficient Condition)**.** *Assume that f is twice differentiable. Any stationary point $(\mathbf{x}, \mathbf{y})$ satisfying $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}) \prec \mathbf{0}$ and*

$$[\nabla^2_{\mathbf{xx}} f - \nabla^2_{\mathbf{xy}} f (\nabla^2_{\mathbf{yy}} f)^{-1} \nabla^2_{\mathbf{yx}} f](\mathbf{x}, \mathbf{y}) \succ \mathbf{0} \tag{4.4}$$

is a local minmax point. We call stationary points satisfying (4.4) strict local minmax points*.*

Proof. Again denote $\mathbf{A} := \nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y})$, $\mathbf{B} := \nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$ and $\mathbf{C} := \nabla^2_{\mathbf{xy}} f(\mathbf{x}, \mathbf{y})$. Since $(\mathbf{x}, \mathbf{y})$ is a stationary point, and $\mathbf{B} \prec 0$, it is clear that $\mathbf{y}$ is the local maximum of $f(\mathbf{x}, \cdot)$. On the other hand, pick $\delta^\dagger_{\mathbf{y}} = \mathbf{B}^{-1}\mathbf{C}^\top \delta_{\mathbf{x}}$ and let $h(\delta) = \|\mathbf{B}^{-1}\mathbf{C}^\top\|\delta$, we know when $\|\delta_{\mathbf{x}}\| \le \delta$, $\|\delta^\dagger_{\mathbf{y}}\| \le h(\delta)$, thus

$$\begin{aligned}\max_{\|\delta_{\mathbf{y}}\| \le h(\delta)} f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta_{\mathbf{y}}) - f(\mathbf{x}, \mathbf{y}) \ge & f(\mathbf{x} + \delta_{\mathbf{x}}, \mathbf{y} + \delta^\dagger_{\mathbf{y}}) - f(\mathbf{x}, \mathbf{y}) \\ = & \frac{1}{2}\delta_{\mathbf{x}}^\top (\mathbf{A} - \mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top)\delta_{\mathbf{x}} + o(\|\delta_{\mathbf{x}}\|^2) > 0\end{aligned}$$

which finishes the proof. □

4.8 Proofs for Limit Points of Gradient Descent Ascent

In this section, we provides proofs for propositions and theorems presented in Section 4.4.

Proposition 4.4.9. *$(\mathbf{x}, \mathbf{y})$ is a strict linearly stable point of γ-GDA if and only if for all the eigenvalues $\{\lambda_i\}$ of following Jacobian matrix,*

$$\mathbf{J}_\gamma = \begin{pmatrix} -(1/\gamma)\nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y}) & -(1/\gamma)\nabla^2_{\mathbf{xy}} f(\mathbf{x}, \mathbf{y}) \\ \nabla^2_{\mathbf{yx}} f(\mathbf{x}, \mathbf{y}) & \nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y}), \end{pmatrix}$$

their real part $\Re(\lambda_i) < 0$ for any i.

Proof. Consider GDA dynamics with step size η, then the Jacobian matrix of this dynamic system is $\mathbf{I} + \eta\mathbf{J}_\gamma$ whose eigenvalues are $\{1 + \eta\lambda_i\}$. Therefore, $(\mathbf{x}, \mathbf{y})$ is a strict linearly stable point if and only if $\rho(\mathbf{I} + \eta\mathbf{J}_\gamma) < 1$, that is $|1 + \eta\lambda_i| < 1$ for all i. When taking $\eta \to 0$, this is equivalent to $\Re(\lambda_i) < 0$ for all i. □

Proposition 4.4.10 ((Daskalakis and Panageas, 2018))**.** *For any fixed γ, for any twice-differentiable f, $\mathit{Local\_Nash} \subset \gamma\text{-}\mathcal{GDA}$, but there exist twice-differentiable f such that $\gamma\text{-}\mathcal{GDA} \not\subset \mathit{Local\_Nash}$.*

Proof. Daskalakis and Panageas (2018) showed the proposition holds for 1-GDA. For completeness, here we show how similar proof goes through for γ-GDA for general γ. Let $\epsilon = 1/\gamma$, and denote $\mathbf{A} := \nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y})$, $\mathbf{B} := \nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$ and $\mathbf{C} := \nabla^2_{\mathbf{xy}} f(\mathbf{x}, \mathbf{y})$.

To prove the statement $\mathit{localNash} \subset \gamma-\mathcal{GDA}$, we note by definition, $(\mathbf{x}, \mathbf{y})$ is a strict linear stable point of $1/\epsilon$-GDA if the real part of the eigenvalues of Jacobian matrix

$$J_\epsilon := \begin{pmatrix} -\epsilon\mathbf{A} & -\epsilon\mathbf{C} \\ \mathbf{C}^\top & \mathbf{B} \end{pmatrix}$$

satisfy that $\Re(\lambda_i) < 0$ for all $1 \le i \le d_1 + d_2$. We first note that:

$$\tilde{J}_\epsilon := \begin{pmatrix} \mathbf{B} & \sqrt{\epsilon}\mathbf{C}^\top \\ -\sqrt{\epsilon}\mathbf{C} & -\epsilon\mathbf{A} \end{pmatrix} = \mathbf{U}J_\epsilon\mathbf{U}^{-1}, \text{ where } \mathbf{U} = \begin{pmatrix} 0 & \sqrt{\epsilon}\mathbf{I} \\ \mathbf{I} & 0 \end{pmatrix}$$

Thus, the eigenvalues of $\tilde{J}_\epsilon$ and J_ϵ are the same. We can also decompose:

$$\tilde{J}_\epsilon = \mathbb{P} + \mathbf{Q}, \text{ where } \mathbb{P} := \begin{pmatrix} \mathbf{B} & \\ & -\epsilon\mathbf{A} \end{pmatrix}, \mathbf{Q} := \begin{pmatrix} 0 & \sqrt{\epsilon}\mathbf{C}^\top \\ -\sqrt{\epsilon}\mathbf{C} & 0 \end{pmatrix}$$

If $(\mathbf{x}, \mathbf{y})$ is a strict local pure strategy Nash equilibrium, then $\mathbf{A} \succ 0, \mathbf{B} \prec 0$, then $\mathbb{P}$ is a negative definite symmetric matrix, and $\mathbf{Q}$ is anti-symmetric matrix, i.e. $\mathbf{Q} = -\mathbf{Q}^\top$. For any eigenvalue λ if $\tilde{J}_\epsilon$, assume $\mathbf{w}$ is the associated eigenvector. That is, $\tilde{J}_\epsilon\mathbf{w} = \lambda\mathbf{w}$, also let $\mathbf{w} = \mathbf{x} + i\mathbf{y}$ where $\mathbf{x}$ and $\mathbf{y}$ are real vectors, and $\bar{\mathbf{w}}$ be the complex conjugate of vector $\mathbf{w}$. Then:

$$\begin{aligned} \Re(\lambda) =& [\bar{\mathbf{w}}^\top\tilde{J}_\epsilon\mathbf{w} + \mathbf{w}^\top\tilde{J}_\epsilon\bar{\mathbf{w}}]/2 = [(\mathbf{x} - i\mathbf{y})^\top\tilde{J}_\epsilon(\mathbf{x} + i\mathbf{y}) + (\mathbf{x} + i\mathbf{y})^\top\tilde{J}_\epsilon(\mathbf{x} - i\mathbf{y})]/2 \\ =& \mathbf{x}^\top\tilde{J}_\epsilon\mathbf{x} + \mathbf{y}^\top\tilde{J}_\epsilon\mathbf{y} = \mathbf{x}^\top\mathbb{P}\mathbf{x} + \mathbf{y}^\top\mathbb{P}\mathbf{y} + \mathbf{x}^\top\mathbf{Q}\mathbf{x} + \mathbf{y}^\top\mathbf{Q}\mathbf{y} \end{aligned}$$

Since $\mathbb{P}$ is negative definite, that is $\mathbf{x}^\top\mathbb{P}\mathbf{x} + \mathbf{y}^\top\mathbb{P}\mathbf{y} < 0$. Meanwhile, since $\mathbf{Q}$ is antisymmtric $\mathbf{x}^\top\mathbf{Q}\mathbf{x} = \mathbf{x}^\top\mathbf{Q}^\top\mathbf{x} = 0$ and $\mathbf{y}^\top\mathbf{Q}\mathbf{y} = \mathbf{y}^\top\mathbf{Q}^\top\mathbf{y} = 0$. This proves $\Re(\lambda) < 0$, that is $(\mathbf{x}, \mathbf{y})$ is a strict linear stable point of $1/\epsilon$-GDA.

To prove the statement $\gamma-\mathcal{GDA} \not\subset \mathit{localNash}$, since ϵ is also fixed, we consider function $f(x, y) = x^2 + 2\sqrt{\epsilon}xy + (\epsilon/2)y^2$. It is easy to see $(0, 0)$ is a fixed point of $1/\epsilon$-GDA, and Hessian $A = 2, B = \epsilon, C = 2\sqrt{\epsilon}$. Thus the Jacobian matrix

$$J_\epsilon := \begin{pmatrix} -2\epsilon & -2\epsilon^{3/2} \\ 2\epsilon^{1/2} & \epsilon \end{pmatrix}$$

has two eigenvalues $\epsilon(-1 \pm i\sqrt{7})/2$. Therefore, $\Re(\lambda_1) = \Re(\lambda_2) < 0$, which implies $(0, 0)$ is a strict linear stable point. However $B = \epsilon > 0$, thus it is not a strict local pure strategy Nash equilibrium. □

Proposition 4.4.11. *For any fixed γ, there exists twice-differentiable f such that $\mathit{Local\_Minmax} \not\subset \gamma-\mathcal{GDA}$; there also exists twice-differentiable f such that $\gamma-\mathcal{GDA} \not\subset \mathit{Local\_Minmax} \cup \mathit{Local\_Maxmin}$.*

Proof. Let $\epsilon = 1/\gamma$, and denote $\mathbf{A} := \nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y})$, $\mathbf{B} := \nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$ and $\mathbf{C} := \nabla^2_{\mathbf{xy}} f(\mathbf{x}, \mathbf{y})$.

To prove the first statement $\mathit{localminmax} \not\subset \gamma\text{-}\mathcal{GDA}$, since ϵ is also fixed, we consider function $f(x, y) = -x^2 + 2\sqrt{\epsilon}xy - (\epsilon/2)y^2$. It is easy to see $(0, 0)$ is a fixed point of $1/\epsilon$-GDA, and Hessian $A = -2, B = -\epsilon, C = 2\sqrt{\epsilon}$. It is easy to verify that $B < 0$ and $A - CB^{-1}C = 2 > 0$, thus $(0, 0)$ is a local minmax point. However, inspect the Jacobian matrix of $1/\epsilon$-GDA:

$$J_\epsilon := \begin{pmatrix} 2\epsilon & -2\epsilon^{3/2} \\ 2\epsilon^{1/2} & -\epsilon \end{pmatrix}$$

We know the two eigenvalues are $\epsilon(1 \pm i\sqrt{7})/2$. Therefore, $\Re(\lambda_1) = \Re(\lambda_2) > 0$, which implies $(0, 0)$ is not a strict linear stable point.

To prove the second statement $\gamma\text{-}\mathcal{GDA} \not\subset \mathit{localminmax} \cup \mathit{localmaxmin}$, since ϵ is also fixed, we consider function $f(\mathbf{x}, \mathbf{y}) = x_1^2 + 2\sqrt{\epsilon}x_1y_1 + (\epsilon/2)y_1^2 - x_2^2/2 + 2\sqrt{\epsilon}x_2y_2 - \epsilon y_2^2$. It is easy to see $(\mathbf{0}, \mathbf{0})$ is a fixed point of $1/\epsilon$-GDA, and Hessian $\mathbf{A} = \mathrm{diag}(2, -1), \mathbf{B} = \mathrm{diag}(\epsilon, -2\epsilon), \mathbf{C} = 2\sqrt{\epsilon} \cdot \mathrm{diag}(1, 1)$. Thus the Jacobian matrix

$$J_\epsilon := \begin{pmatrix} -2\epsilon & 0 & -2\epsilon^{3/2} & 0 \\ 0 & \epsilon & 0 & -2\epsilon^{3/2} \\ 2\epsilon^{1/2} & 0 & \epsilon & 0 \\ 0 & 2\epsilon^{1/2} & 0 & -2\epsilon \end{pmatrix}$$

has four eigenvalues $\epsilon(-1 \pm i\sqrt{7})/2$ (each with multiplicity of 2). Therefore, $\Re(\lambda_i) < 0$ for $1 \le i \le 4$, which implies $(\mathbf{0}, \mathbf{0})$ is a strict linear stable point. However, $\mathbf{B}$ is not negative definite, thus $(\mathbf{0}, \mathbf{0})$ is not a strict local minmax point; similarly, $\mathbf{A}$ is also not positive definite, thus $(\mathbf{0}, \mathbf{0})$ is not a strict local maxmin point. □

Theorem 4.4.12 (Main Theorem)**.** *For any twice-differentiable f, $\mathit{Local\_Minmax} \subset \underline{\infty\text{-}\mathcal{GDA}} \subset \overline{\infty\text{-}\mathcal{GDA}} \subset \mathit{Local\_Minmax} \cup \{(\mathbf{x}, \mathbf{y}) | (\mathbf{x}, \mathbf{y})$ is stationary and $\nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$ is degenerate$\}$.*

Proof. For simplicity, denote $\mathbf{A} := \nabla^2_{\mathbf{xx}} f(\mathbf{x}, \mathbf{y})$, $\mathbf{B} := \nabla^2_{\mathbf{yy}} f(\mathbf{x}, \mathbf{y})$ and $\mathbf{C} := \nabla^2_{\mathbf{xy}} f(\mathbf{x}, \mathbf{y})$. Let $\epsilon = 1/\gamma$. Consider sufficiently small ϵ (i.e. sufficiently large γ), we know the Jacobian J of $1/\epsilon$-GDA at $(\mathbf{x}, \mathbf{y})$ is:

$$J_\epsilon := \begin{pmatrix} -\epsilon\mathbf{A} & -\epsilon\mathbf{C} \\ \mathbf{C}^\top & \mathbf{B} \end{pmatrix}$$

According to Lemma 4.8.1, for sufficient ϵ, J_ϵ has $d_1 + d_2$ complex eigenvalues $\{\lambda_i\}_{i=1}^{d_1+d_2}$ with following form for sufficient small ϵ:

$$\begin{aligned} &|\lambda_i + \epsilon\mu_i| = o(\epsilon) && 1 \le i \le d_1 \\ &|\lambda_{i+d_1} - \nu_i| = o(1), && 1 \le i \le d_2 \end{aligned} \tag{4.6}$$

where $\{\mu_i\}_{i=1}^{d_1}$ and $\{\nu_i\}_{i=1}^{d_2}$ are the eigenvalues of matrices $\mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top$ and $\mathbf{B}$ respectively. Now we are ready to prove the three inclusion statement in Theorem 4.4.12 seperately.

First, for $\underline{\infty-\mathcal{GDA}} \subset \overline{\infty-\mathcal{GDA}}$ always holds by their definitions.

Second, for $\overline{\mathcal{Local\_Minmax}} \subset \underline{\infty-\mathcal{GDA}}$ statement, if $(\mathbf{x},\mathbf{y})$ is strict local minmax point, then by its definition:

$$\mathbf{B} \prec 0, \quad \text{and} \quad \mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top \succ 0$$

By Eq.(4.6) the eigenvalue structure of J_ϵ, we know there exists sufficiently small ϵ_0, so that for any $\epsilon<\epsilon_0$, the real part $\Re(\lambda_i)<0$, i.e. $(\mathbf{x},\mathbf{y})$ is a strict linear stable point of $1/\epsilon-$GDA.

Finally, for $\overline{\infty-\mathcal{GDA}} \subset \mathcal{Local\_Minmax} \cup \{(\mathbf{x},\mathbf{y})|(\mathbf{x},\mathbf{y}) \text{ is stationary and } \mathbf{B} \text{ is degenerate}\}$ statement, if $(\mathbf{x},\mathbf{y})$ is strict linear stable point of $1/\epsilon-$GDA for a sufficiently small ϵ, then for any i, the real part of eigenvalue of J_ϵ: $\Re(\lambda_i)<0$. By Eq.(4.6), if $\mathbf{B}$ is invertible, this implies:

$$\mathbf{B} \prec 0, \quad \text{and} \quad \mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top \succeq 0$$

Finally, suppose matrix $\mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top$ has an eigenvalue 0. This means the existence of unit vector $\mathbf{w}$ so that $(\mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top)\mathbf{w}=0$. It is not hard to verify then $J_\epsilon\cdot(\mathbf{w},-\mathbf{B}^{-1}\mathbf{C}^\top\mathbf{w})^\top=0$. This implies J_ϵ has a 0 eigen-value, which contradicts the fact that $\Re(\lambda_i)<0$ for any i. Therefore, we can conclude $\mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top \succ 0$, and $(\mathbf{x},\mathbf{y})$ is a strict local minmax point. □

Lemma 4.8.1. *For any symmetric matrix $\mathbf{A}\in\mathbb{R}^{d_1\times d_1}$, $\mathbf{B}\in\mathbb{R}^{d_2\times d_2}$, and any rectangular matrix $\mathbf{C}\in\mathbb{R}^{d_1\times d_2}$, assume $\mathbf{B}$ is nondegenerate. Then, matrix*

$$\begin{pmatrix} -\epsilon\mathbf{A} & -\epsilon\mathbf{C} \\ \mathbf{C}^\top & \mathbf{B} \end{pmatrix}$$

has d_1+d_2 complex eigenvalues $\{\lambda_i\}_{i=1}^{d_1+d_2}$ with following form for sufficient small ϵ:

$$\begin{aligned} &|\lambda_i+\epsilon\mu_i| = o(\epsilon) && 1\le i\le d_1 \\ &|\lambda_{i+d_1}-\nu_i| = o(1), && 1\le i\le d_2 \end{aligned}$$

where $\{\mu_i\}_{i=1}^{d_1}$ and $\{\nu_i\}_{i=1}^{d_2}$ are the eigenvalues of matrices $\mathbf{A}-\mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top$ and $\mathbf{B}$ respectively.

Proof. By definition of eigenvalues, $\{\lambda_i\}_{i=1}^{d_1+d_2}$ are the roots of characteristic polynomial:

$$p_\epsilon(\lambda) := \det\begin{pmatrix} \lambda\mathbf{I}+\epsilon\mathbf{A} & \epsilon\mathbf{C} \\ -\mathbf{C}^\top & \lambda\mathbf{I}-\mathbf{B} \end{pmatrix}$$

We can expand this polynomial as:

$$p_\epsilon(\lambda) = p_0(\lambda)+\sum_{i=1}^{d_1+d_2}\epsilon^i p_i(\lambda), \qquad p_0(\lambda)=\lambda^{d_1}\cdot\det(\lambda\mathbf{I}-\mathbf{B}).$$

Here, p_i are polynomials of order at most $d_1 + d_2$. It is clear that the roots of p_0 are 0 (with multiplicity d_1) and $\{\nu_i\}_{i=1}^{d_2}$. According to Lemma 4.8.2, we know the roots of p_ϵ satisfy:

$$\begin{aligned} |\lambda_i| &= o(1) && 1 \le i \le d_1 \\ |\lambda_{i+d_1} - \nu_i| &= o(1), && 1 \le i \le d_2 \end{aligned} \tag{4.7}$$

Since $\mathbf{B}$ is non-degenerate, we know when ϵ is small enough, $\lambda_1 \dots \lambda_{d_1}$ are very close to 0 while $\lambda_{d_1+1} \dots \lambda_{d_1+d_2}$ have modulus at least $\Omega(1)$. To provide the sign information of the first d_1 roots, we proceed to lower order characterization.

On the other hand, reparametrize $\lambda = \epsilon\theta$, we have:

$$p_\epsilon(\epsilon\theta) = \det \begin{pmatrix} \epsilon\theta\mathbf{I} + \epsilon\mathbf{A} & \epsilon\mathbf{C} \\ -\mathbf{C}^\top & \epsilon\theta\mathbf{I} - \mathbf{B} \end{pmatrix} = \epsilon^{d_1} \det \begin{pmatrix} \theta\mathbf{I} + \mathbf{A} & \mathbf{C} \\ -\mathbf{C}^\top & \epsilon\theta\mathbf{I} - \mathbf{B} \end{pmatrix}$$

Therefore, we know $q_\epsilon(\theta) := p_\epsilon(\epsilon\theta)/\epsilon^{d_1}$ is still a polynomial, and has polynomial expansion:

$$q_\epsilon(\theta) = q_0(\theta) + \sum_{i=1}^{d_2} \epsilon^i q_i(\lambda), \qquad q_0(\theta) = \det \begin{pmatrix} \theta\mathbf{I} + \mathbf{A} & \mathbf{C} \\ -\mathbf{C}^\top & -\mathbf{B} \end{pmatrix}$$

It is also clear polynomial q_ϵ and p_ϵ have same roots up to ϵ scaling. Furthermore, we have following factorization:

$$\begin{pmatrix} \theta\mathbf{I} + \mathbf{A} & \mathbf{C} \\ -\mathbf{C}^\top & -\mathbf{B} \end{pmatrix} = \begin{pmatrix} \theta\mathbf{I} + \mathbf{A} - \mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top & \mathbf{C} \\ 0 & -\mathbf{B} \end{pmatrix} \begin{pmatrix} \mathbf{I} & 0 \\ \mathbf{B}^{-1}\mathbf{C}^\top & \mathbf{I} \end{pmatrix}$$

Since $\mathbf{B}$ is non-degenerate, we have $\det(\mathbf{B}) \neq 0$, and

$$q_0(\theta) = (-1)^{d_2} \det(\mathbf{B}) \det(\theta\mathbf{I} + \mathbf{A} - \mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top)$$

q_0 is d_1-order polynomial having roots $\{\mu_i\}_{i=1}^{d_1}$, which are the eigenvalues of matrices $\mathbf{A} - \mathbf{C}\mathbf{B}^{-1}\mathbf{C}^\top$. According to Lemma 4.8.2, we know q_ϵ has at least d_1 roots so that $|\theta_i + \mu_i| \le o(1)$. This implies d_1 roots of p_ϵ so that:

$$|\lambda_i + \epsilon\mu_i| = o(\epsilon) \qquad 1 \le i \le d_1$$

By Eq.(4.7), we know p_ϵ has exactly d_1 roots which are of $o(1)$ scaling. This finishes the proof. □

Lemma 4.8.2 (Continuity of roots of polynomials (Zedek, 1965))**.** *Given a polynomial $p_n(z) := \sum_{k=0}^n a_k z^k$, $a_n \neq 0$, an integer $m \ge n$ and a number $\epsilon > 0$, there exists a number $\delta > 0$ such that whenever the $m+1$ complex numbers $b_k, 0 \le k \le m$, satisfy the inequlities*

$$|b_k - a_k| < \delta \quad \text{for} \quad 0 \le k \le n, \quad \text{and} \quad |b_k| < \delta \quad \text{for} \quad n+1 \le k \le m$$

then the roots $\beta_k, 1 \le k \le m$ of the polynomial $q_m(z) := \sum_{k=0}^m b_k z^k$ can be labeled in such a way as to satisfy with respect to the zeros $\alpha_k, 1 \le k \le n$ of $p_n(z)$ the inequalities

$$|\beta_k - \alpha_k| < \epsilon \quad \text{for} \quad 1 \le k \le n, \quad \text{and} \quad |\beta_k| > 1/\epsilon \quad \text{for} \quad n+1 \le k \le m$$

4.9 Proofs for Gradient Descent with Max-oracle

In this section, we present the proof for Theorem 4.4.13 presented in Section 4.4.

Theorem 4.4.13. *Suppose f is ℓ-smooth and L-Lipschitz and define $\phi(\cdot) := \max_{\mathbf{y}} f(\cdot, \mathbf{y})$. Then the output $\bar{\mathbf{x}}$ of GD with Max-oracle (Algorithm 9) with step size $\eta = \gamma/\sqrt{T+1}$ will satisfy*

$$\mathbb{E}\left[\|\nabla \phi_{1/2\ell}(\bar{\mathbf{x}})\|^2\right] \leq 2 \cdot \frac{\left(\phi_{1/2\ell}(\mathbf{x}_0) - \min \phi(\mathbf{x})\right) + \ell L^2 \gamma^2}{\gamma\sqrt{T+1}} + 4\ell\epsilon,$$

where $\phi_{1/2\ell}$ is the Moreau envelope (4.5) of ϕ.

Proof. The proof of this theorem mostly follows the proof of Theorem 2.1 from (Davis and Drusvyatskiy, 2018). The only difference is that y_t in Algorithm 9 is only an approximate maximizer and not exact maximizer. However, the proof goes through fairly easily with an additional error term.

We first note an important equation for the gradient of Moreau envelope.

$$\nabla \phi_\lambda(\mathbf{x}) = \lambda^{-1}\left(\mathbf{x} - \operatorname*{argmin}_{\widetilde{\mathbf{x}}}\left(\phi(\widetilde{\mathbf{x}}) + \frac{1}{2\lambda}\|\mathbf{x} - \widetilde{\mathbf{x}}\|^2\right)\right). \tag{4.8}$$

We also observe that since $f(\cdot)$ is ℓ-smooth and y_t is an approximate maximizer for $\mathbf{x}_t$, we have that any $\mathbf{x}_t$ from Algorithm 9 and $\widetilde{\mathbf{x}}$ satisfy

$$\begin{aligned}\phi(\widetilde{\mathbf{x}}) \geq f(\widetilde{\mathbf{x}}, \mathbf{y}_t) &\geq f(\mathbf{x}_t, \mathbf{y}_t) + \langle \nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t), \widetilde{\mathbf{x}} - \mathbf{x}_t\rangle - \frac{\ell}{2}\|\widetilde{\mathbf{x}} - \mathbf{x}_t\|^2 \\ &\geq \phi(\mathbf{x}_t) - \epsilon + \langle \nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t), \widetilde{\mathbf{x}} - \mathbf{x}_t\rangle - \frac{\ell}{2}\|\widetilde{\mathbf{x}} - \mathbf{x}_t\|^2. \end{aligned} \tag{4.9}$$

Let $\widehat{\mathbf{x}}_t := \operatorname{argmin}_{\mathbf{x}} \phi(\mathbf{x}) + \ell\|\mathbf{x} - \mathbf{x}_t\|^2$. We have:

$$\begin{aligned}\phi_{1/2\ell}(\mathbf{x}_{t+1}) &\leq \phi(\widehat{\mathbf{x}}_t) + \ell\|\mathbf{x}_{t+1} - \widehat{\mathbf{x}}_t\|^2 \\ &\leq \phi(\widehat{\mathbf{x}}_t) + \ell\|\mathbf{x}_t - \eta\nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t) - \widehat{\mathbf{x}}_t\|^2 \\ &\leq \phi(\widehat{\mathbf{x}}_t) + \ell\|\mathbf{x}_t - \widehat{\mathbf{x}}_t\|^2 + 2\ell\eta\langle \nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t), \widehat{\mathbf{x}}_t - \mathbf{x}_t\rangle + \eta^2\ell\|\nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t)\|^2 \\ &\leq \phi_{1/2\ell}(\mathbf{x}_t) + 2\eta\ell\langle \nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t), \widehat{\mathbf{x}}_t - \mathbf{x}_t\rangle + \eta^2\ell\|\nabla_{\mathbf{x}} f(\mathbf{x}_t, \mathbf{y}_t)\|^2 \\ &\leq \phi_{1/2\ell}(\mathbf{x}_t) + 2\eta\ell\left(\phi(\widehat{\mathbf{x}}_t) - \phi(\mathbf{x}_t) + \epsilon + \frac{\ell}{2}\|\mathbf{x}_t - \widehat{\mathbf{x}}_t\|^2\right) + \eta^2\ell L^2, \end{aligned}$$

where the last line follows from (4.9). Taking a telescopic sum over t, we obtain

$$\phi_{1/2\ell}(\mathbf{x}_T) \leq \phi_{1/2\ell}(\mathbf{x}_0) + 2\eta\ell\sum_{t=0}^{T}\left(\phi(\widehat{\mathbf{x}}_t) - \phi(\mathbf{x}_t) + \epsilon + \frac{\ell}{2}\|\mathbf{x}_t - \widehat{\mathbf{x}}_t\|^2\right) + \eta^2\ell L^2 T$$

Rearranging this, we obtain

$$\frac{1}{T+1}\sum_{t=0}^{T}\left(\phi(\mathbf{x}_t)-\phi(\widehat{\mathbf{x}}_t)-\frac{\ell}{2}\|\mathbf{x}_t-\widehat{\mathbf{x}}_t\|^2\right)\leq \epsilon+\frac{\phi_{1/2\ell}(\mathbf{x}_0)-\min_{\mathbf{x}}\phi(\mathbf{x})}{2\eta\ell T}+\frac{\eta L^2}{2}. \tag{4.10}$$

Since $\phi(\mathbf{x})+\ell\|\mathbf{x}-\mathbf{x}_t\|^2$ is ℓ-strongly convex, we have

$$\begin{aligned}
&\phi(\mathbf{x}_t)-\phi(\widehat{\mathbf{x}}_t)-\frac{\ell}{2}\|\mathbf{x}_t-\widehat{\mathbf{x}}_t\|^2\\
&\geq \phi(\mathbf{x}_t)+\ell\|\mathbf{x}_t-\mathbf{x}_t\|^2-\phi(\widehat{\mathbf{x}}_t)-\ell\|\widehat{\mathbf{x}}_t-\mathbf{x}_t\|^2+\frac{\ell}{2}\|\mathbf{x}_t-\widehat{\mathbf{x}}_t\|^2\\
&=\left(\phi(\mathbf{x}_t)+\ell\|\mathbf{x}_t-\mathbf{x}_t\|^2-\min_{\mathbf{x}}\phi(\mathbf{x})+\ell\|\mathbf{x}-\mathbf{x}_t\|^2\right)+\frac{\ell}{2}\|\mathbf{x}_t-\widehat{\mathbf{x}}_t\|^2\\
&\geq \ell\|\mathbf{x}_t-\widehat{\mathbf{x}}_t\|^2=\frac{1}{4\ell}\|\nabla\phi_{1/2\ell}(\mathbf{x}_t)\|^2,
\end{aligned}$$

where we used (4.8) in the last step. Plugging this in (4.10) proves the result. □

Part III

Reinforcement Learning

Chapter 5

On Sample Efficiency of Q-learning

Model-free reinforcement learning (RL) algorithms, such as Q-learning, directly parameterize and update value functions or policies without explicitly modeling the environment. They are typically simpler, more flexible to use, and thus more prevalent in modern deep RL than model-based approaches. However, empirical work has suggested that model-free algorithms may require more samples to learn (Deisenroth and Rasmussen, 2011; Schulman et al., 2015). The theoretical question of "whether model-free algorithms can be made *sample efficient*" is one of the most fundamental questions in RL, and remains unsolved even in the basic scenario with finitely many states and actions.

We prove that, in an episodic MDP setting, Q-learning with UCB exploration achieves regret $\tilde{\mathcal{O}}(\sqrt{H^3SAT})$, where S and A are the numbers of states and actions, H is the number of steps per episode, and T is the total number of steps. This sample efficiency matches the optimal regret that can be achieved by any model-based approach, up to a single $\sqrt{H}$ factor. To the best of our knowledge, this is the first analysis in the model-free setting that establishes $\sqrt{T}$ regret *without* requiring access to a "simulator."

5.1 Introduction

Reinforcement Learning (RL) is a control-theoretic problem in which an agent tries to maximize its cumulative rewards via interacting with an unknown *environment* through time (Sutton and Barto, 1998). There are two main approaches to RL: model-based and model-free. Model-based algorithms make use of a model for the environment, forming a control policy based on this learned model. Model-free approaches dispense with the model and directly update the *value function*—the expected reward starting from each state, or the *policy*—the mapping from states to their subsequent actions. There has been a long debate on the relative pros and cons of the two approaches (Deisenroth and Rasmussen, 2011).

From the classical Q-learning algorithm (Watkins, 1989) to modern DQN (Mnih et al., 2013), A3C (Mnih et al., 2016), TRPO (Schulman et al., 2015), and others, most state-of-the-art RL has been in the model-free paradigm. Its pros—model-free algorithms are online,

require less space, and, most importantly, are more expressive since specifying the value functions or policies is often more flexible than specifying the model for the environment—arguably outweigh its cons relative to model-based approaches. These relative advantages underly the significant successes of model-free algorithms in deep RL applications (Mnih et al., 2013; Silver et al., 2016).

On the other hand it is believed that model-free algorithms suffer from a higher sample complexity compared to model-based approaches. This has been evidenced empirically in (Deisenroth and Rasmussen, 2011; Schulman et al., 2015), and recent work has tried to improve the sample efficiency of model-free algorithms by combining them with model-based approaches (Nagabandi et al., 2017; Pong et al., 2018). There is, however, little theory to support such blending, which requires a more quantitative understanding of relative sample complexities. Indeed, the following basic theoretical questions remain open:

Can we design model-free algorithms that are sample efficient?
In particular, **is Q-learning provably efficient?**

The answers remain elusive even in the basic tabular setting where the number of states and actions are finite. In this work, we attack this problem head-on in the setting of the episodic Markov Decision Process (MDP) formalism (see Section **??** for a formal definition). In this setting, an episode consists of a run of MDP dynamics for H steps, where the agent aims to maximize total reward over multiple episodes. We do not assume access to a "simulator" (which would allow us to query arbitrary state-action pairs of the MDP) and the agent is not allowed to "reset" within each episode. This makes our setting sufficiently challenging and realistic. In this setting, the standard Q-learning heuristic of incorporating ε-greedy exploration appears to take exponentially many episodes to learn (Kearns and Singh, 2002).

As seen in the literature on bandits, the key to achieving good sample efficiency generally lies in managing the tradeoff between *exploration* and *exploitation*. One needs an efficient strategy to explore the uncertain environment while maximizing reward. In the model-based setting, a recent line of research has imported ideas from the bandit literature—including the use of upper confidence bounds (UCB) and improved design of exploration bonuses—and has obtained asymptotically optimal sample efficiency (Jaksch, Ortner, and Auer, 2010; Agrawal and Jia, 2017; Azar, Osband, and Munos, 2017; Kakade, Wang, and Yang, 2018). In contrast, the understanding of model-free algorithms is still very limited. To the best of our knowledge, the only existing theoretical result on model-free RL that applies to the episodic setting is for *delayed Q-learning*; however, this algorithm is quite sample-inefficient compared to model-based approaches (Strehl et al., 2006).

In this work, we answer the two aforementioned questions affirmatively. We show that Q-learning, when equipped with a UCB exploration policy that incorporates estimates of the confidence of Q values and assign exploration bonuses, achieves total regret $\tilde{\mathcal{O}}(\sqrt{H^3SAT})$. Here, S and A are the numbers of states and actions, H is the number of steps per episode, and T is the total number of steps. Up to a $\sqrt{H}$ factor, our regret matches the information-theoretic optimum, which can be achieved by model-based algorithms (Azar, Osband, and

Munos, 2017; Kakade, Wang, and Yang, 2018). Since our algorithm is just Q-learning, it is online and does not store additional data besides the table of Q values (and a few integers per entry of this table). Thus, it also enjoys a significant advantage over model-based algorithms in terms of time and space complexities. To our best knowledge, this is the first sharp analysis for model-free algorithms—featuring $\sqrt{T}$ regret or equivalently $O(1/\varepsilon^2)$ samples for ε-optimal policy—*without* requiring access to a "simulator."

For practitioners, there are two key takeaways from our theoretical analysis:

1. The use of UCB exploration instead of ε-greedy exploration in the model-free setting allows for better treatment of uncertainties for different states and actions.

2. It is essential to use a learning rate which is $\alpha_t = O(H/t)$, instead of $1/t$, when a state-action pair is being updated for the t-th time. The former learning rate assigns more weight to updates that are more recent, as opposed to assigning uniform weights to all previous updates. This delicate choice of reweighting leads to the crucial difference between our sample-efficient guarantee versus earlier highly inefficient results that require exponentially many samples in H.

Related Work

In this section, we focus our attention on theoretical results for the tabular MDP setting, where the numbers of states and actions are finite. We acknowledge that there has been much recent work in RL for continuous state spaces see, e.g., Jiang et al., 2016; Fazel et al., 2018, but this setting is beyond our scope.

With simulator Some results assume access to a simulator (Koenig and Simmons, 1993) (a.k.a., a generative model (Azar, Munos, and Kappen, 2012)), which is a strong oracle that allows the algorithm to query arbitrary state-action pairs and return the reward and the next state. The majority of these results focus on an infinite-horizon MDP with discounted reward e.g., Even-Dar and Mansour, 2003; Azar et al., 2011; Lattimore and Hutter, 2012; Azar, Munos, and Kappen, 2012; Sidford et al., 2018. When a simulator is available, model-free algorithms (Azar et al., 2011) (variants of Q-learning) are known to be almost as sample efficient as the best model-based algorithms (Azar, Munos, and Kappen, 2012). However, the simulator setting is considered to much easier than standard RL, as it "does not require exploration" (Azar et al., 2011). Indeed, a naive exploration strategy which queries all state-action pairs uniformly at random already leads to the most efficient algorithm for finding

	Algorithm	Regret	Time	Space
Model-based	UCRL2 (Jaksch, Ortner, and Auer, 2010) [1]	at least $\tilde{\mathcal{O}}(\sqrt{H^4S^2AT})$	$\Omega(TS^2A)$	$\mathcal{O}(S^2AH)$
	Agrawal and Jia (2017) [1]	at least $\tilde{\mathcal{O}}(\sqrt{H^3S^2AT})$		
	UCBVI (Azar, Osband, and Munos, 2017) [2]	$\tilde{\mathcal{O}}(\sqrt{H^2SAT})$	$\tilde{\mathcal{O}}(TS^2A)$	
	vUCQ (Kakade, Wang, and Yang, 2018) [2]	$\tilde{\mathcal{O}}(\sqrt{H^2SAT})$		
Model-free	Q-learning (ε-greedy) (Kearns and Singh, 2002) (if 0 initialized)	$\Omega(\min\{T, A^{H/2}\})$	$\mathcal{O}(T)$	$\mathcal{O}(SAH)$
	Delayed Q-learning (Strehl et al., 2006) [3]	$\tilde{\mathcal{O}}_{S,A,H}(T^{4/5})$		
	Q-learning (UCB-H)	$\tilde{\mathcal{O}}(\sqrt{H^4SAT})$		
	Q-learning (UCB-B)	$\tilde{\mathcal{O}}(\sqrt{H^3SAT})$		
	lower bound	$\Omega(\sqrt{H^2SAT})$	-	-

Table 5.1: Regret comparisons for RL algorithms on episodic MDP. $T = KH$ is totally number of steps, H is the number of steps per episode, S is the number of states, and A is the number of actions. For clarity, this table is presented for $T \geq \text{poly}(S, A, H)$, omitting low order terms.

optimal policy (Azar, Munos, and Kappen, 2012).

Without simulator Reinforcement learning becomes much more challenging without the presence of a simulator, and the choice of exploration policy can now determine the behavior of the learning algorithm. For instance, Q-learning with ε-greedy may take exponentially

[1] Jaksch, Ortner, and Auer (2010) and Agrawal and Jia (2017) apply to the more general setting of weakly communicating MDPs with S' states and diameter D; our episodic MDP is a special case obtained by augmenting the state space so that $S' = SH$ and $D \geq H$.

[2] Azar, Osband, and Munos (2017) and Kakade, Wang, and Yang (2018) assume equal transition matrices $\mathbb{P}_1 = \cdots = \mathbb{P}_H$; in the setting of this work $\mathbb{P}_1, \cdots, \mathbb{P}_H$ can be entirely different. This adds a factor of $\sqrt{H}$ to their total regret.

[3] Strehl et al. (2006) applies to MDPs with S' states and discount factor γ; our episodic MDP can be converted to that case by setting $S' = SH$ and $1 - \gamma = 1/H$. Their result only applies to the stochastic setting where initial states x_1^k come from a fixed distribution, and only gives a PAC guarantee. We have translated it to a regret guarantee (see Section 5.3).

many episodes to learn the optimal policy (Kearns and Singh, 2002) (for the sake of completeness, we present this result in our episodic language in Appendix 5.5).

In the model-based setting, UCRL2 (Jaksch, Ortner, and Auer, 2010) and Agrawal and Jia (2017) form estimates of the transition probabilities of the MDP using past samples, and add upper-confidence bounds (UCB) to the estimated transition matrix. When applying their results to the episodic MDP scenario, their total regret is at least $\tilde{\mathcal{O}}(\sqrt{H^4S^2AT})$ and $\tilde{\mathcal{O}}(\sqrt{H^3S^2AT})$ respectively.[1] In contrast, the information-theoretic lower bound is $\tilde{\mathcal{O}}(\sqrt{H^2SAT})$. The additional $\sqrt{S}$ and $\sqrt{H}$ factors were later removed by the UCBVI algorithm (Azar, Osband, and Munos, 2017) which adds a UCB bonus directly to the Q values instead of the estimated transition matrix.[2] The vUCQ algorithm (Kakade, Wang, and Yang, 2018) is similar to UCBVI but improves lower-order regret terms using variance reduction.

We note that despite the sharp regret guarantees, all of the results in this line of research require estimating and storing the entire transition matrix and thus suffer from unfavorable time and space complexities compared to model-free algorithms.

In the model-free setting, Strehl et al. (2006) introduced delayed Q-learning, where, to find an ε-optimal policy, the Q value for each state-action pair is updated only once every $m = \tilde{\mathcal{O}}(1/\varepsilon^2)$ times this pair is visited. In contrast to the incremental update of Q-learning, delayed Q-learning always replaces old Q values with the average of the most recent m experiences. When translated to the setting of this work, this gives $\tilde{\mathcal{O}}(T^{4/5})$ total regret, ignoring factors in S, A and H.[3] This is quite suboptimal compared to the $\tilde{\mathcal{O}}(\sqrt{T})$ regret achieved by model-based algorithm.

5.2 Preliminary

We consider the setting of a tabular episodic Markov decision process, $\mathrm{MDP}(\mathcal{S}, \mathcal{A}, H, \mathbb{P}, r)$, where $\mathcal{S}$ is the set of states with $|\mathcal{S}| = S$, $\mathcal{A}$ is the set of actions with $|\mathcal{A}| = A$, H is the number of steps in each episode, $\mathbb{P}$ is the transition matrix so that $\mathbb{P}_h(\cdot|x, a)$ gives the distribution over states if action a is taken for state x at step $h \in [H]$, and $r_h \colon \mathcal{S} \times \mathcal{A} \to [0, 1]$ is the deterministic reward function at step h.[4]

In each episode of this MDP, an initial state x_1 is picked arbitrarily by an adversary. Then, at each step $h \in [H]$, the agent observes state $x_h \in \mathcal{S}$, picks an action $a_h \in \mathcal{A}$, receives reward $r_h(x_h, a_h)$, and then transitions to a next state, x_{h+1}, that is drawn from the distribution $\mathbb{P}_h(\cdot|x_h, a_h)$. The episode ends when x_{H+1} is reached.

A policy π of an agent is a collection of H functions $\{\pi_h : \mathcal{S} \to \mathcal{A}\}_{h\in[H]}$. We use $V_h^\pi \colon \mathcal{S} \to \mathbb{R}$ to denote the value function at step h under policy π, so that $V_h^\pi(x)$ gives the expected sum of remaining rewards received under policy π, starting from $x_h = x$, until the

[4]While we study deterministic reward functions for notational simplicity, our results generalize to randomized reward functions. Also, we assume the reward is in $[0, 1]$ without loss of generality.

Algorithm 10 Q-learning with UCB-Hoeffding

initialize $Q_h(x,a) \leftarrow H$ and $N_h(x,a) \leftarrow 0$ for all $(x,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$.
for episode $k = 1, \ldots, K$ **do**
  receive x_1.
  for step $h = 1, \ldots, H$ **do**
    Take action $a_h \leftarrow \operatorname{argmax}_{a'} Q_h(x_h, a')$, and observe x_{h+1}.
    $t = N_h(x_h, a_h) \leftarrow N_h(x_h, a_h) + 1$; $b_t \leftarrow c\sqrt{H^3\iota/t}$.
    $Q_h(x_h, a_h) \leftarrow (1-\alpha_t)Q_h(x_h, a_h) + \alpha_t[r_h(x_h, a_h) + V_{h+1}(x_{h+1}) + b_t]$.
    $V_h(x_h) \leftarrow \min\{H, \max_{a' \in \mathcal{A}} Q_h(x_h, a')\}$.

end of the episode. In symbols:

$$V_h^\pi(x) := \mathbb{E}\left[\sum_{h'=h}^H r_{h'}(x_{h'}, \pi_{h'}(x_{h'}))|x_h = x\right] \ .$$

Accordingly, we also define $Q_h^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ to denote Q-value function at step h so that $Q_h^\pi(x,a)$ gives the expected sum of remaining rewards received under policy π, starting from $x_h = x, a_h = a$, till the end of the episode. In symbols:

$$Q_h^\pi(x,a) := r_h(x,a) + \mathbb{E}[\sum_{h'=h+1}^H r_{h'}(x_{h'}, \pi_{h'}(x_{h'}))|x_h = x, a_h = a] \ .$$

Since the state and action spaces, and the horizon, are all finite, there always exists (see, e.g., (Azar, Osband, and Munos, 2017)) an optimal policy $\pi^\star$ which gives the optimal value $V_h^\star(x) = \sup_\pi V_h^\pi(x)$ for all $x \in \mathcal{S}$ and $h \in [H]$. For simplicity, we denote $[\mathbb{P}_h V_{h+1}](x,a) := \mathbb{E}_{x' \sim \mathbb{P}(\cdot|x,a)} V_{h+1}(x')$. Recall the Bellman equation and the Bellman optimality equation:

$$\begin{cases} V_h^\pi(x) = Q_h^\pi(x, \pi_h(x)) \\ Q_h^\pi(x,a) := (r_h + \mathbb{P}_h V_{h+1}^\pi)(x,a) \\ V_{H+1}^\pi(x) = 0 \qquad \forall x \in \mathcal{S} \end{cases} \quad \text{and} \quad \begin{cases} V_h^\star(x) = \max_{a \in \mathcal{A}} Q_h^\star(x,a) \\ Q_h^\star(x,a) := (r_h + \mathbb{P}_h V_{h+1}^\star)(x,a) \\ V_{H+1}^\star(x) = 0 \qquad \forall x \in \mathcal{S} \ . \end{cases} \tag{5.1}$$

The agent plays the game for K episodes $k = 1, 2, \ldots, K$, and we let the adversary pick a starting state x_1^k for each episode k, and let the agent choose a policy π_k before starting the k-th episode. The total (expected) regret is then

$$\text{Regret}(K) = \sum_{k=1}^K \left[V_1^\star(x_1^k) - V_1^{\pi_k}(x_1^k)\right] \ .$$

5.3 Main Results

In this section, we present our main theoretical result—a sample complexity result for a variant of Q-learning that incorporates UCB exploration. We also present a theorem that establishes an information-theoretic lower bound for episodic MDP.

As seen in the bandit setting, the choice of exploration policy plays an essential role in the efficiency of a learning algorithm. In episodic MDP, Q-learning with the commonly used ε-greedy exploration strategy can be very inefficient: it can take exponentially many episodes to learn (Kearns and Singh, 2002) (see also Appendix 5.5). In contrast, our algorithm (Algorithm 10), which is Q-learning with an upper-confidence bound (UCB) exploration strategy, will be seen to be efficient. This algorithm maintains Q values, $Q_h(x,a)$, for all $(x,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$ and the corresponding V values $V_h(x) \leftarrow \min\{H, \max_{a' \in \mathcal{A}} Q_h(x,a')\}$. If, at time step $h \in [H]$, the state is $x \in \mathcal{S}$, the algorithm takes the action $a \in \mathcal{A}$ that maximizes the current estimate $Q_h(x,a)$, and is apprised of the next state $x' \in \mathcal{S}$. The algorithm then updates the Q values:

$$Q_h(x,a) \leftarrow (1-\alpha_t) Q_h(x,a) + \alpha_t [r_h(x,a) + V_{h+1}(x') + b_t] \ ,$$

where t is the counter for how many times the algorithm has visited the state-action pair (x,a) at step h, b_t is the confidence bonus indicating how certain the algorithm is about current state-action pair, and α_t is a learning rate defined as follows:

$$\alpha_t := \frac{H+1}{H+t} \ . \tag{5.2}$$

As mentioned in the introduction, our choice of learning rate α_t scales as $O(H/t)$ instead of $O(1/t)$—this is crucial to obtain regret that is not exponential in H.

We present analyses for two different specifications of the upper confidence bonus b_t in this work:

Q-learning with Hoeffding-style bonus The first (and simpler) choice is $b_t = O(\sqrt{H^3 \iota / t})$. (Here, and throughout this work, we use $\iota := \log(SAT/p)$ to denote a log factor.) This choice of bonus makes sense intuitively because: (1) Q-values are upper-bounded by H, and, accordingly, (2) Hoeffding-type martingale concentration inequalities imply that if we have visited (x,a) for t times, then a confidence bound for the Q value scales as $1/\sqrt{t}$. For this reason, we call this choice *UCB-Hoeffding* (UCB-H). See Algorithm 10.

Theorem 5.3.1 (Hoeffding)**.** *There exists an absolute constant $c > 0$ such that, for any $p \in (0,1)$, if we choose $b_t = c\sqrt{H^3 \iota / t}$, then with probability $1-p$, the total regret of Q-learning with UCB-Hoeffding (see Algorithm 10) is at most $O(\sqrt{H^4 SAT\iota})$, where $\iota := \log(SAT/p)$.*

Theorem 5.3.1 shows, under a rather simple choice of exploration bonus, Q-learning can be made very efficient, enjoying a $\tilde{\mathcal{O}}(\sqrt{T})$ regret which is optimal in terms of dependence on T. To the best of our knowledge, this is the first analysis of a model-free procedure that features a $\sqrt{T}$ regret *without* requiring access to a "simulator."

Compared to the previous model-based results, Theorem 5.3.1 shows that the regret (or equivalently the sample complexity; see discussion in Section 5.3) of this version of Q-learning is as good as the best model-based one in terms of the dependency on the number of states S, actions A and the total number of steps T. Although our regret slightly increases the dependency on H, the algorithm is online and does not store additional data besides the table of Q values (and a few integers per entry of this table). Thus, it enjoys an advantage over model-based algorithms in time and space complexities, especially when the number of states S is large.

Q-learning with Bernstein-style bonus Our second specification of b_t makes use of a Bernstein-style upper confidence bound. The key observation is that, although in the worst case the value function is at most H for any state-action pair, if we sum up the "total variance of the value function" for an entire episode, we obtain a factor of only $O(H^2)$ as opposed to the naive $O(H^3)$ bound (see Lemma 5.7.6). This implies that the use of a Bernstein-type martingale concentration result could be sharper than the Hoeffding-type bound by an additional factor of H.[5] (The idea of using Bernstein instead of Hoeffding for reinforcement learning applications has appeared in previous work; see, e.g., (Azar, Munos, and Kappen, 2012; Azar, Munos, and Kappen, 2013; Lattimore and Hutter, 2012).)

Using Bernstein concentration requires us to design the bonus term b_t more carefully, as it now depends on the empirical variance of $V_{h+1}(x')$ where x' is the next state over the previous t visits of current state-action (x, a). This empirical variance can be computed in an online fashion without increasing the space complexity of Q-learning. We defer the full specification of b_t to Algorithm 11 in Appendix 5.7. We now state the regret theorem for this approach.

Theorem 5.3.2 (Bernstein)**.** *For any $p \in (0, 1)$, one can specify b_t so that with probability $1 - p$, the total regret of Q-learning with UCB-Bernstein (see Algorithm 11) is at most $O(\sqrt{H^3SAT\iota} + \sqrt{H^9S^3A^3} \cdot \iota^2)$.*

Theorem 5.3.2 shows that for Q-learning with UCB-B exploration, the leading term in regret (which scales as $\sqrt{T}$) improves by a factor of $\sqrt{H}$ over UCB-H exploration, at the price of using a more complicated exploration bonus design. The asymptotic regret of UCB-B is now only one $\sqrt{H}$ factor worse than the best regret achieved by model-based algorithms.

We also note that Theorem 5.3.2 has an additive term $O(\sqrt{H^9S^3A^3} \cdot \iota^2)$ in its regret, which dominates the total regret when T is not very large compared with S, A and H. It is not clear whether this lower-order term is essential, or is due to technical aspects of the current analysis.

[5] Recall that for independent zero-mean random variables $X_1, \ldots, X_T$ satisfying $|X_i| \le M$, their summation does not exceed $\tilde{\mathcal{O}}(M\sqrt{T})$ with high probability using Hoeffding concentration. If we have in hand a better variance bound, this can be improved to $\tilde{\mathcal{O}}\big(M + \sqrt{\sum_i \mathbb{E}[X_i]^2}\big)$ using Bernstein concentration.

Information-theoretical limit To demonstrate the sharpness of our results, we also note an information-theoretic lower bound for the episodic MDP setting studied in this work:

Theorem 5.3.3. *For the episodic MDP problem studied in this work, the expected regret for any algorithm must be at least* $\Omega(\sqrt{H^2SAT})$.

Theorem 5.3.3 (see Appendix 5.8 for details) shows that both variants of our algorithm are nearly optimal, in the sense they differ from the optimal regret by a factor of H and $\sqrt{H}$, respectively.

From Regret to PAC Guarantee

Recall that the probably approximately correct (PAC) learning setting for RL provides sample complexity guarantee to find a near-optimal policy (Kakade, 2003). In this setting, the initial state $x_1 \in \mathcal{S}$ is sampled from a fixed initial distribution, rather than being chosen adversarially. Without loss of generality, we only discuss here the case in which x_1 is fixed; the general case reduces to this case by adding an additional time step at the beginning of each episode. The PAC-learning question is "how many samples are needed to find an ε-optimal policy π satisfying $V_1^\star(x_1) - V_1^\pi(x_1) \le \varepsilon$?"

Any algorithm with total regret sublinear in T yields a finite sample complexity in the PAC setting. Indeed, suppose we have total regret $\sum_{k=1}^K [V_1^\star(x_1) - V_1^{\pi_k}(x_1)] \le C \cdot T^{1-\alpha}$, where $\alpha \in (0,1)$ is a absolute constant, and C is independent of T. Then, by randomly selecting $\pi = \pi_k$ for $k = 1, 2, \ldots, K$, we have $V_1^\star(x_1) - V_1^\pi(x_1) \le 3CH \cdot T^{-\alpha}$ with probability at least $2/3$. Therefore, for every $\varepsilon \in (0, H]$, our Theorem 5.3.1 (for UCB-H) and Theorem 5.3.2 (for UCB-B) also find ε-optimal policies in the PAC setting using $\tilde{\mathcal{O}}(H^5SA/\varepsilon^2)$ and $\tilde{\mathcal{O}}(H^4SA/\varepsilon^2)$ samples respectively.

Conversely, any algorithm with finite sample complexity in the PAC setting translates to sublinear total regret in non-adversarial case (assuming x_1 is chosen from a fixed distribution). Suppose the algorithm finds ε-optimal policy π using $T_1 = C \cdot \varepsilon^{-\beta}$ samples where $\beta \ge 1$ is a constant. Then, we can use this π to play the game for another $T - T_1$ steps, giving total regret $T_1 + \varepsilon(T - T_1)/H$. After balancing T and T_1 optimally, this gives $\tilde{\mathcal{O}}\big(C^{1+\beta} \cdot (T/H)^{\beta/(1+\beta)}\big)$ total regret. For instance, Strehl et al. (2006) gives sampling complexity $\propto 1/\varepsilon^4$ in the PAC setting, and this translates to $\propto T^{4/5}$ total regret.

5.4 Proof for Q-learning with UCB-Hoeffding

In this section, we provide the full proof of Theorem 5.3.1. Intuitively, the episodic MDP with H steps per epsiode can be viewed as a contextual bandit of H "layers." The key challenge here is to control the way error and confidence propagate through different "layers" in an online fashion, where our specific choice of exploration bonus and learning rate make the regret as sharp as possible.

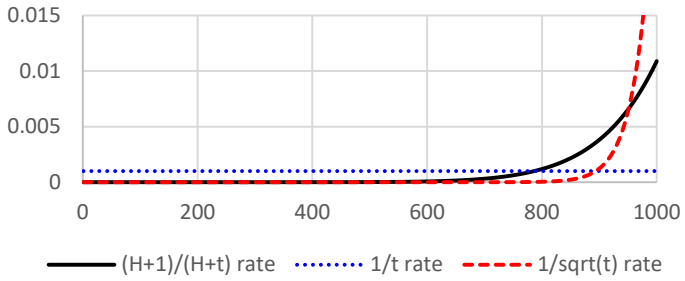

Figure 5.1: Illustration of $\{\alpha^i_{1000}\}_{i=1}^{1000}$ for learning rates $\alpha_t = \frac{H+1}{H+t}$, $\frac{1}{t}$ and $\frac{1}{\sqrt{t}}$ when $H = 10$.

Notation We denote by $\mathbb{I}[A]$ the indicator function for event A. We denote by (x_h^k, a_h^k) the actual state-action pair observed and chosen at step h of episode k. We also denote by Q_h^k, V_h^k, N_h^k respectively the Q_h, V_h, N_h functions at the *beginning* of episode k. Using this notation, the update equation at episode k can be rewritten as follows, for every $h \in [H]$:

$$Q_h^{k+1}(x,a) = \begin{cases} (1-\alpha_t)Q_h^k(x,a) + \alpha_t[r_h(x,a) + V_{h+1}^k(x_{h+1}^k) + b_t] & \text{if } (x,a) = (x_h^k, a_h^k) \\ Q_h^k(x,a) & \text{otherwise .} \end{cases} \tag{5.3}$$

Accordingly,

$$V_h^k(x) \leftarrow \min\left\{H, \max_{a' \in \mathcal{A}} Q_h^k(x,a')\right\}, \quad \forall x \in \mathcal{S} \ .$$

Recall that we have $[\mathbb{P}_h V_{h+1}](x,a) := \mathbb{E}_{x' \sim \mathbb{P}_h(\cdot|x,a)} V_{h+1}(x')$. We also denote its empirical counterpart of episode k as $[\hat{\mathbb{P}}_h^k V_{h+1}](x,a) := V_{h+1}(x_{h+1}^k)$, which is defined only for $(x,a) = (x_h^k, a_h^k)$.

Recall that we have chosen the learning rate as $\alpha_t := \frac{H+1}{H+t}$. For notational convenience, we also introduce the following related quantities:

$$\alpha_t^0 = \textstyle\prod_{j=1}^t (1-\alpha_j), \qquad \alpha_t^i = \alpha_i \textstyle\prod_{j=i+1}^t (1-\alpha_j) \ . \tag{5.4}$$

It is easy to verify that (1) $\sum_{i=1}^t \alpha_t^i = 1$ and $\alpha_t^0 = 0$ for $t \geq 1$; (2) $\sum_{i=1}^t \alpha_t^i = 0$ and $\alpha_t^0 = 1$ for $t = 0$.

Favoring Later Updates At any $(x,a,h,k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$, let $t = N_h^k(x,a)$ and suppose (x,a) was previously taken at step h of episodes $k_1, \ldots, k_t < k$. By the update equation (5.3) and the definition of α_t^i in (5.4), we have:

$$Q_h^k(x,a) = \alpha_t^0 H + \sum_{i=1}^t \alpha_t^i \left[r_h(x,a) + V_{h+1}^{k_i}(x_{h+1}^{k_i}) + b_i\right] \ . \tag{5.5}$$

According to (5.5), the Q value at episode k equals a weighted average of the V values of the "next states" with weights $\alpha_t^1, \ldots, \alpha_t^t$. As one can see from Figure 5.1, our choice of

the learning rate $\alpha_t = \frac{H+1}{H+t}$ ensures that, approximately speaking, the last $1/H$ fraction of the indices i is given non-negligible weights, whereas the first $1 - 1/H$ fraction is forgotten. This ensures that the information accumulates smoothly across the H layers of the MDP. If one were to use $\alpha_t = \frac{1}{t}$ instead, the weights $\alpha_t^1, \ldots, \alpha_t^t$ would all equal $1/t$, and using those V values from earlier episodes would hurt the accuracy of the Q function. In contrast, if one were to use $\alpha_t = 1/\sqrt{t}$ instead, the weights $\alpha_t^1, \ldots, \alpha_t^t$ would concentrate too much on the most recent episodes, which would incur high variance.

Proof Details

We first present an auxiliary lemma which exhibits some important properties that result from our choice of learning rate. The proof is based on simple manipulations on the definition of α_t, and is provided in Appendix 5.6.

Lemma 5.4.1. *The following properties hold for α_t^i:*

1. $\frac{1}{\sqrt{t}} \le \sum_{i=1}^t \frac{\alpha_t^i}{\sqrt{i}} \le \frac{2}{\sqrt{t}}$ *for every* $t \ge 1$.
2. $\max_{i \in [t]} \alpha_t^i \le \frac{2H}{t}$ *and* $\sum_{i=1}^t (\alpha_t^i)^2 \le \frac{2H}{t}$ *for every* $t \ge 1$.
3. $\sum_{t=i}^\infty \alpha_t^i = 1 + \frac{1}{H}$ *for every* $i \ge 1$.

We note that property (c) is especially important—as we will show later, each step in one episode can blow up the regret by a multiplicative factor of $\sum_{t=i}^\infty \alpha_t^i$. With our choice of learning rate, we ensure that this blow-up is at most $(1+1/H)^H$, which is a constant factor.

We now proceed to the formal proof. We start with a lemma that gives a recursive formula for $Q - Q^\star$, as a weighted average of previous updates.

Lemma 5.4.2 (recursion on Q)**.** *For any $(x, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$ and episode $k \in [K]$, let $t = N_h^k(x, a)$ and suppose (x, a) was previously taken at step h of episodes $k_1, \ldots, k_t < k$. Then:*

$$(Q_h^k - Q_h^\star)(x,a) = \alpha_t^0 (H - Q_h^\star(x,a)) + \sum_{i=1}^t \alpha_t^i \left[(V_{h+1}^{k_i} - V_{h+1}^\star)(x_{h+1}^{k_i}) + [(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h) V_{h+1}^\star](x,a) + b_i \right] .$$

Proof of Lemma 5.4.2. From the Bellman optimality equation, $Q_h^\star(x,a) = (r_h + \mathbb{P}_h V_{h+1}^\star)(x,a)$, our notation $[\hat{\mathbb{P}}_h^{k_i} V_{h+1}](x,a) := V_{h+1}(x_{h+1}^{k_i})$, and the fact that $\sum_{i=0}^t \alpha_t^i = 1$, we have

$$Q_h^\star(x,a) = \alpha_t^0 Q_h^\star(x,a) + \sum_{i=1}^t \alpha_t^i \left[r_h(x,a) + \left(\mathbb{P}_h - \hat{\mathbb{P}}_h^{k_i}\right) V_{h+1}^\star(x,a) + V_{h+1}^\star(x_{h+1}^{k_i}) \right] .$$

Subtracting the formula (5.5) from this equation, we obtain Lemma 5.4.2. □

Next, using Lemma 5.4.2 and the Azuma-Hoeffding concentration bound, our next lemma shows that Q^k is always an upper bound on $Q^\star$ at any episode k, and the difference between Q^k and $Q^\star$ can be bounded by quantities from the next step.

Lemma 5.4.3 (bound on $Q^k - Q^\star$)**.** *There exists an absolute constant $c > 0$ such that, for any $p \in (0,1)$, letting $b_t = c\sqrt{H^3\iota/t}$, we have $\beta_t = 2\sum_{i=1}^t \alpha_t^i b_i \le 4c\sqrt{H^3\iota/t}$ and, with probability at least $1-p$, the following holds simultaneously for all $(x,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$:*

$$0 \le (Q_h^k - Q_h^\star)(x,a) \le \alpha_t^0 H + \sum_{i=1}^t \alpha_t^i (V_{h+1}^{k_i} - V_{h+1}^\star)(x_{h+1}^{k_i}) + \beta_t \ ,$$

where $t = N_h^k(x,a)$ and $k_1, \ldots, k_t < k$ are the episodes where (x,a) was taken at step h.

Proof of Lemma 5.4.3. For each fixed $(x,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$, let us denote $k_0 = 0$, and denote

$$k_i = \min\left(\left\{k \in [K] \mid k > k_{i-1} \wedge (x_h^k, a_h^k) = (x,a)\right\} \cup \{K+1\}\right) \ .$$

That is, k_i is the episode of which (x,a) was taken at step h for the ith time (or $k_i = K+1$ if it is taken for fewer than i times). The random variable k_i is clearly a stopping time. Let $\mathcal{F}_i$ be the σ-field generated by all the random variables until episode k_i, step h. Then, $\left(\mathbb{I}[k_i \le K]\cdot[(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h)V_{h+1}^\star](x,a)\right)_{i=1}^\tau$ is a martingale difference sequence w.r.t the filtration $\{\mathcal{F}_i\}_{i\ge 0}$. By Azuma-Hoeffding and a union bound, we have that with probability at least $1 - p/(SAH)$:

$$\forall \tau \in [K]: \quad \left|\sum_{i=1}^\tau \alpha_\tau^i \cdot \mathbb{I}[k_i \le K]\cdot[(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h)V_{h+1}^\star](x,a)\right| \le \frac{cH}{2}\sqrt{\sum_{i=1}^\tau (\alpha_\tau^i)^2 \cdot \iota} \le c\sqrt{\frac{H^3\iota}{\tau}} \ , \tag{5.6}$$

for some absolute constant c. Because inequality (5.6) holds for all fixed $\tau \in [K]$ uniformly, it also holds for $\tau = t = N_h^k(x,a) \le K$, which is a random variable, where $k \in [K]$. Also note $\mathbb{I}[k_i \le K] = 1$ for all $i \le N_h^k(x,a)$. Putting everything together, and using a union bound, we see that with least $1-p$ probability, the following holds simultaneously for all $(x,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$:

$$\left|\sum_{i=1}^t \alpha_t^i [(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h)V_{h+1}^\star](x,a)\right| \le c\sqrt{\frac{H^3\iota}{t}} \quad \text{where} \quad t = N_h^k(x,a) \ . \tag{5.7}$$

On the other hand, if we choose $b_t = c\sqrt{H^3\iota/t}$ for the same constant c in Eq. (5.6), we have $\beta_t/2 = \sum_{i=1}^t \alpha_t^i b_i \in [c\sqrt{H^3\iota/t}, 2c\sqrt{H^3\iota/t}]$ according to Lemma 1. Then the right-hand side of Lemma 5.4.3 follows immediately from Lemma 5.4.2 and inequality (5.7). The left-hand side also follows from Lemma 5.4.2 and Eq. (5.7) and induction on $h = H, H-1, \ldots, 1$. □

We are now ready to prove Theorem 5.3.1. The proof decomposes the regret in a recursive form, and carefully controls the error propagation with repeated usage of Lemma 5.4.3.

Proof of Theorem 5.3.1. Denote by

$$\delta_h^k := (V_h^k - V_h^{\pi_k})(x_h^k) \quad \text{and} \quad \phi_h^k := (V_h^k - V_h^\star)(x_h^k) \ .$$

By Lemma 5.4.3, we have that with $1-p$ probability, $Q_h^k \geq Q_h^\star$ and thus $V_h^k \geq V_h^\star$. Thus, the total regret can be upper bounded:

$$\text{Regret}(K) = \textstyle\sum_{k=1}^K (V_1^\star - V_1^{\pi_k})(x_1^k) \leq \sum_{k=1}^K (V_1^k - V_1^{\pi_k})(x_1^k) = \sum_{k=1}^K \delta_1^k \ .$$

The main idea of the rest of the proof is to upper bound $\sum_{k=1}^K \delta_h^k$ by the next step $\sum_{k=1}^K \delta_{h+1}^k$, thus giving a recursive formula to calculate total regret. We can obtain such a recursive formula by relating $\sum_{k=1}^K \delta_h^k$ to $\sum_{k=1}^K \phi_h^k$.

For any fixed $(k,h) \in [K] \times [H]$, let $t = N_h^k(x_h^k, a_h^k)$, and suppose (x_h^k, a_h^k) was previously taken at step h of episodes $k_1, \ldots, k_t < k$. Then we have:

$$\begin{aligned}
\delta_h^k &= (V_h^k - V_h^{\pi_k})(x_h^k) \overset{①}{\leq} (Q_h^k - Q_h^{\pi_k})(x_h^k, a_h^k) \\
&= (Q_h^k - Q_h^\star)(x_h^k, a_h^k) + (Q_h^\star - Q_h^{\pi_k})(x_h^k, a_h^k) \\
&\overset{②}{\leq} \alpha_t^0 H + \textstyle\sum_{i=1}^t \alpha_t^i \phi_{h+1}^{k_i} + \beta_t + [\mathbb{P}_h(V_{h+1}^\star - V_{h+1}^{\pi_k})](x_h^k, a_h^k) \\
&\overset{③}{=} \alpha_t^0 H + \textstyle\sum_{i=1}^t \alpha_t^i \phi_{h+1}^{k_i} + \beta_t - \phi_{h+1}^k + \delta_{h+1}^k + \xi_{h+1}^k \ , \qquad (5.8)
\end{aligned}$$

where $\beta_t = 2\sum \alpha_t^i b_i \leq O(1)\sqrt{H^3\iota/t}$ and $\xi_{h+1}^k := [(\mathbb{P}_h - \hat{\mathbb{P}}_h^k)(V_{h+1}^\star - V_{h+1}^k)](x_h^k, a_h^k)$ is a martingale difference sequence. Inequality ① holds because $V_h^k(x_h^k) \leq \max_{a' \in \mathcal{A}} Q_h^k(x_h^k, a') = Q_h^k(x_h^k, a_h^k)$, and inequality ② holds by Lemma 5.4.3 and the Bellman equation (5.1). Finally, equality ③ holds by definition $\delta_{h+1}^k - \phi_{h+1}^k = (V_{h+1}^\star - V_{h+1}^{\pi_k})(x_{h+1}^k)$.

We turn to computing the summation $\sum_{k=1}^K \delta_h^k$. Denoting by $n_h^k = N_h^k(x_h^k, a_h^k)$, we have:

$$\sum_{k=1}^K \alpha_{n_h^k}^0 H = \sum_{k=1}^K H \cdot \mathbb{I}[n_h^k = 0] \leq SAH \ .$$

The key step is to upper bound the second term in (5.8), which is:

$$\sum_{k=1}^K \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \phi_{h+1}^{k_i(x_h^k, a_h^k)},$$

where $k_i(x_h^k, a_h^k)$ is the episode in which (x_h^k, a_h^k) was taken at step h for the ith time. We regroup the summands in a different way. For every $k' \in [K]$, the term $\phi_{h+1}^{k'}$ appears in the summand with $k > k'$ if and only if $(x_h^k, s_h^k) = (x_h^{k'}, s_h^{k'})$. The first time it appears we have $n_h^k = n_h^{k'} + 1$, the second time it appears we have $n_h^k = n_h^{k'} + 2$, and so on. Therefore

$$\sum_{k=1}^K \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \phi_{h+1}^{k_i(x_h^k, a_h^k)} \leq \sum_{k'=1}^K \phi_{h+1}^{k'} \sum_{t=n_h^{k'}+1}^{\infty} \alpha_t^{n_h^{k'}} \leq \left(1 + \frac{1}{H}\right) \sum_{k=1}^K \phi_{h+1}^k,$$

where the final inequality uses $\sum_{t=i}^{\infty} \alpha_t^i = 1 + \frac{1}{H}$ from Lemma 3. Plugging these back into (5.8), we have:

$$\begin{aligned}\sum_{k=1}^{K} \delta_h^k &\le SAH + \left(1 + \frac{1}{H}\right) \sum_{k=1}^{K} \phi_{h+1}^k - \sum_{k=1}^{K} \phi_{h+1}^k + \sum_{k=1}^{K} \delta_{h+1}^k + \sum_{k=1}^{K} (\beta_{n_h^k} + \xi_{h+1}^k) \\ &\le SAH + \left(1 + \frac{1}{H}\right) \sum_{k=1}^{K} \delta_{h+1}^k + \sum_{k=1}^{K} (\beta_{n_h^k} + \xi_{h+1}^k) \ , \end{aligned} \tag{5.9}$$

where the final inequality uses $\phi_{h+1}^k \le \delta_{h+1}^k$ (owing to the fact that $V^\star \ge V^{\pi_k}$). Recursing the result for $h = 1, 2, \ldots, H$, and using the fact $\delta_{H+1}^K \equiv 0$, we have:

$$\sum_{k=1}^{K} \delta_1^k \le O\Big(H^2 SA + \sum_{h=1}^{H} \sum_{k=1}^{K} (\beta_{n_h^k} + \xi_{h+1}^k)\Big).$$

Finally, by the pigeonhole principle, for any $h \in [H]$:

$$\sum_{k=1}^{K} \beta_{n_h^k} \le O(1) \cdot \sum_{k=1}^{K} \sqrt{\frac{H^3\iota}{n_h^k}} = O(1) \cdot \sum_{x,a} \sum_{n=1}^{N_h^K(x,a)} \sqrt{\frac{H^3\iota}{n}} \overset{①}{\le} O\big(\sqrt{H^3SAK\iota}\big) = O\big(\sqrt{H^2SAT\iota}\big) \tag{5.10}$$

where inequality ① is true because $\sum_{x,a} N_h^K(x,a) = K$ and the left-hand side of ① is maximized when $N_h^K(x,a) = K/SA$ for all x, a. Also, by the AzumaHoeffding inequality, with probability $1 - p$, we have:

$$\Big|\sum_{h=1}^{H} \sum_{k=1}^{K} \xi_{h+1}^k\Big| = \Big|\sum_{h=1}^{H} \sum_{k=1}^{K} [(\mathbb{P}_h - \hat{\mathbb{P}}_h^k)(V_{h+1}^\star - V_{h+1}^k)](x_h^k, a_h^k)\Big| \le cH\sqrt{T\iota}.$$

This establishes $\sum_{k=1}^{K} \delta_1^k \le O\big(H^2SA + \sqrt{H^4SAT\iota}\big)$. We note that when $T \ge \sqrt{H^4SAT\iota}$, we have $\sqrt{H^4SAT\iota} \ge H^2SA$, and when $T \le \sqrt{H^4SAT\iota}$, we have $\sum_{k=1}^{K} \delta_1^k \le HK = T \le \sqrt{H^4SAT\iota}$. Therefore, we can remove the H^2SA term in the regret upper bound.

In sum, we have $\sum_{k=1}^{K} \delta_1^k \le O\big(H^2SA + \sqrt{H^4SAT\iota}\big)$, with probability at least $1 - 2p$. Rescaling p to $p/2$ finishes the proof. □

5.5 Explanation for Q-Learning with ε-Greedy

We recall a construction of a hard instance for Q-learning, known as a "combination lock," and tracing back at least to Koenig and Simmons (1993). In our context of our episodic MDP, this instance corresponds to the following MDP.

Consider a special state $s^\star \in \mathcal{S}$ where the adversary always picks $x_1 = s^\star$. For steps $h = 1, 2, \ldots, H/2$, there is one special action $a^\star \in \mathcal{A}$ where the distribution $\mathbb{P}_h(\cdot|s^\star, a^\star)$ is a

singleton and always leads to a next state $x_{h+1} = s^\star$. For any other state $s \in \mathcal{S} \setminus \{s^\star\}$, or any other action $a \in \mathcal{A} \setminus \{a^\star\}$, the distribution $\mathbb{P}_h(\cdot|s,a)$ is uniform over $\mathcal{S} \setminus \{s^\star\}$. For steps $h = H/2+1, \ldots, H$, $\mathbb{P}_h(\cdot|s,a)$ is always a singleton and leads to the next state $x_{h+1} = s$. Finally, the reward function $r_h(s,a) = 0$ for all s, a, h, except when $s = s^\star$ and $h > H/2$, we have $r_H(s^\star, a^\star) = 1$. It is clear that the optimal policy gives reward $H/2$ (by always selecting action $a^\star$).

For this MDP, for the Q-learning algorithm (or its Sarsa variant) with zero initialization, unless the algorithm picks a path with prefix $(x_1, a_1, x_2, a_2, \ldots, x_{H/2}, a_{H/2}) = (s^\star, a^\star, \ldots, s^\star, a^\star)$, the reward value of the path is always zero and thus the algorithm will not change $Q_h(s,a)$ for any s, a, h. In other words, all Q values remain at zero until the first time $(s^\star, a^\star, \ldots, s^\star, a^\star)$ is visited. Unfortunately, this can happen with probability at most $A^{-H/2}$, and therefore the algorithm must suffer $H/2$ regret per round unless $K \geq \Omega(A^{H/2})$.

5.6 Proof of Lemma 5.4.1

In this section, we derive three important properties implied by our choice of the learning rate. Recall the notation from (5.2) and (5.4):

$$\alpha_t = \frac{H+1}{H+t}, \qquad \alpha_t^0 = \prod_{j=1}^{t}(1-\alpha_j), \qquad \alpha_t^i = \alpha_i \prod_{j=i+1}^{t}(1-\alpha_j) \ .$$

Lemma 5.4.1. *The following properties hold for α_t^i:*

1. $\frac{1}{\sqrt{t}} \leq \sum_{i=1}^{t} \frac{\alpha_t^i}{\sqrt{i}} \leq \frac{2}{\sqrt{t}}$ *for every* $t \geq 1$.
2. $\max_{i \in [t]} \alpha_t^i \leq \frac{2H}{t}$ *and* $\sum_{i=1}^{t} (\alpha_t^i)^2 \leq \frac{2H}{t}$ *for every* $t \geq 1$.
3. $\sum_{t=i}^{\infty} \alpha_t^i = 1 + \frac{1}{H}$ *for every* $i \geq 1$.

Proof of Lemma 5.4.1.

1. The proof is by induction on t. For the base case $t = 1$ we have $\sum_{i=1}^{t} \frac{\alpha_t^i}{\sqrt{i}} = \alpha_1^1 = 1$ so the statement holds. For $t \geq 2$, by the relationship $\alpha_t^i = (1-\alpha_t)\alpha_{t-1}^i$ for $i = 1, 2, \ldots, t-1$ we have:

$$\sum_{i=1}^{t} \frac{\alpha_t^i}{\sqrt{i}} = \frac{\alpha_t}{\sqrt{t}} + (1-\alpha_t) \sum_{i=1}^{t-1} \frac{\alpha_{t-1}^i}{\sqrt{i}} \ .$$

   On the one hand, by induction we have:

$$\frac{\alpha_t}{\sqrt{t}} + (1-\alpha_t) \sum_{i=1}^{t-1} \frac{\alpha_{t-1}^i}{\sqrt{i}} \geq \frac{\alpha_t}{\sqrt{t}} + \frac{1-\alpha_t}{\sqrt{t-1}} \geq \frac{\alpha_t}{\sqrt{t}} + \frac{1-\alpha_t}{\sqrt{t}} = \frac{1}{\sqrt{t}} \ .$$

On the other hand, by induction we have:

$$\frac{\alpha_t}{\sqrt{t}} + (1-\alpha_t)\sum_{i=1}^{t-1}\frac{\alpha_{t-1}^i}{\sqrt{i}} \le \frac{\alpha_t}{\sqrt{t}} + \frac{2(1-\alpha_t)}{\sqrt{t-1}} = \frac{H+1}{\sqrt{t}(H+t)} + \frac{2\sqrt{t-1}}{H+t}$$
$$\le \frac{H+1}{\sqrt{t}(H+t)} + \frac{2\sqrt{t}}{H+t} = \frac{2}{\sqrt{t}} + \frac{1}{\sqrt{t}}\cdot\frac{1-H}{t+H} \le \frac{2}{\sqrt{t}} \ ,$$

where the final inequality holds because $H \ge 1$.

2. We have:

$$\alpha_t^i = \frac{H+1}{i+H}\cdot\Big(\frac{i}{i+1+H}\frac{i+1}{i+2+H}\cdots\frac{t-1}{t+H}\Big)$$
$$= \frac{H+1}{t+H}\cdot\Big(\frac{i}{i+H}\frac{i+1}{i+1+H}\cdots\frac{t-1}{t-1+H}\Big) \le \frac{H+1}{t+H} \le \frac{2H}{t} \ .$$

Therefore, we have proved $\max_{i\in[t]}\alpha_t^i \le 2H/t$. The second inequality, $\sum_{i=1}^t(\alpha_t^i)^2 \le 2H/t$, follows directly since $\sum_{i=1}^t(\alpha_t^i)^2 \le [\max_{i\in[t]}\alpha_t^i]\cdot\sum_{i=1}^t\alpha_t^i$ and $\sum_{i=1}^t\alpha_t^i = 1$.

3. We first note the following identity, which holds for all positive integers n and k with $n \ge k$:

$$\frac{n}{k} = 1 + \frac{n-k}{n+1} + \frac{n-k}{n+1}\frac{n-k+1}{n+2} + \frac{n-k}{n+1}\frac{n-k+1}{n+2}\frac{n-k+2}{n+3} + \cdots \ . \tag{5.11}$$

To verify (5.11), we write the terms of its right-hand side as $x_0 = 1, x_1 = \frac{n-k}{n+1}, \ldots$. It is easy to verify by induction that $\frac{n}{k} - \sum_{i=0}^t x_i = \frac{n-k}{k}\prod_{i=1}^t\frac{n-k+i}{n+i}$. This means $\lim_{t\to\infty}\frac{n}{k} - \sum_{i=0}^t x_i = 0$ and this proves that (5.11) holds. Now, using (5.11) with $n = i+H$ and $k = H$, we have:

$$\sum_{t=i}^{\infty}\alpha_t^i = \frac{H+1}{i+H}\cdot\big(1 + \frac{i}{i+1+H} + \frac{i}{i+1+H}\frac{i+1}{i+2+H} + \cdots\big) = \frac{H+1}{i+H}\cdot\frac{i+H}{H} = \frac{H+1}{H} \ .$$

□

5.7 Proof for Q-learning with UCB-Bernstein

In this section, we prove Theorem 5.3.2.

Notation In addition to the notation of Section 5.4, we define a variance operator $\mathbb{V}_h$:

$$[\mathbb{V}_h V_{h+1}](x,a) := \mathrm{Var}_{x'\sim\mathbb{P}_h(\cdot|x,a)}(V_{h+1}(x')) = \mathbb{E}_{x'\sim\mathbb{P}_h(\cdot|x,a)}\left[V_{h+1}(x') - [\mathbb{P}_h V_{h+1}](x,a)\right]^2$$

We also consider an empirical version of variance that can be computed by the algorithm: when (x,a) was taken at step h for t times at $k_1,\cdots,k_t$ episodes respectively:

$$W_t(x,a,h) := \frac{1}{t}\sum_{i=1}^{t}\left[V_{h+1}^{k_i}(x_{h+1}^{k_i}) - \tfrac{1}{t}\textstyle\sum_{j=1}^{t} V_{h+1}^{k_j}(x_{h+1}^{k_j})\right]^2 \quad . \tag{5.12}$$

In this section, we choose two constants $c_1, c_2 > 0$ and define

$$\beta_t(x,a,h) := \min\left\{c_1\Big(\sqrt{\frac{H}{t}\cdot(W_t(x,a,h)+H)\iota} + \frac{\sqrt{H^7SA}\cdot\iota}{t}\Big), c_2\sqrt{\frac{H^3\iota}{t}}\right\} \ , \tag{5.13}$$

and accordingly,

$$b_1(x,a,h) := \frac{\beta_1(x,a,h)}{2} \qquad b_t(x,a,h) := \frac{\beta_t(x,a,h) - (1-\alpha_t)\beta_{t-1}(x,a,h)}{2\alpha_t} \ . \tag{5.14}$$

It is easy to verify that $\beta_t = 2\sum_{i=1}^{t}\alpha_t^i b_i$ for every $t \geq 1$. We include in Algorithm 11 the efficient implementation for calculating $b_t(x,a,h)$ in $O(1)$ time per time step. Now we restate Theorem 5.3.2.

Theorem 5.7.1. *thm:bernstein[Bernstein, restated] There exist absolute constants $c_1, c_2 > 0$ such that, for any $p \in (0,1)$, if we choose b_t according to (5.14), then with probability $1-p$, the total regret of Q-learning with UCB-Bernstein (see Algorithm 11) is at most $O(\sqrt{H^3SAT\iota} + \sqrt{H^9S^3A^3}\cdot\iota^2)$.*

Proof

We first note that the following recursion, obtained in the proof for the Hoeffding case (see Lemma 5.4.2), still holds here:

Lemma 5.7.2 (recursion on Q)**.** *For any $(x,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$ and episode $k \in [K]$, let $t = N_h^k(x,a)$ and suppose (x,a) was previously taken at step h of episodes $k_1,\ldots,k_t < k$, then*

$$\begin{aligned}(Q_h^k - Q_h^\star)(x,a) &= \alpha_t^0(H - Q_h^\star(x,a)) \\ &\quad + \sum_{i=1}^{t}\alpha_t^i\left[(V_{h+1}^{k_i} - V_{h+1}^\star)(x_{h+1}^{k_i}) + [(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h)V_{h+1}^\star](x,a) + b_i(x,a,h)\right] \ .\end{aligned}$$

Algorithm 11 Q-learning with UCB-Bernstein

for all $(x,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$ **do**
  $Q_h(x,a) \leftarrow H$; $N_h(x,a) \leftarrow 0$; $\mu_h(x,a) \leftarrow 0$; $\sigma_h(x,a) \leftarrow 0$; $\beta_0(x,a,h) \leftarrow 0$.
for episode $k = 1, \ldots, K$ **do**
  receive x_1.
  for step $h = 1, \ldots, H$ **do**
    Take action $a_h \leftarrow \operatorname{argmax}_{a'} Q_h(x_h, a')$, and observe x_{h+1}.
    $t = N_h(x_h, a_h) \leftarrow N_h(x_h, a_h) + 1$.
    $\mu_h(x_h, a_h) \leftarrow \mu_h(x_h, a_h) + V_{h+1}(x_{h+1})$.
    $\sigma_h(x_h, a_h) \leftarrow \sigma_h(x_h, a_h) + \big(V_{h+1}(x_{h+1})\big)^2$.
    $\beta_t(x_h, a_h, h) \leftarrow \min\left\{c_1\left(\sqrt{\frac{H}{t}\frac{\sigma_h(x_h,a_h)-(\mu_h(x_h,a_h))^2}{t} + H)\iota} + \frac{\sqrt{H^7SA}\cdot\iota}{t}\right), c_2\sqrt{\frac{H^3\iota}{t}}\right\}$.
    $b_t \leftarrow \frac{\beta_t(x_h,a_h,h)-(1-\alpha_t)\beta_{t-1}(x_h,a_h,h)}{2\alpha_t}$.
    $Q_h(x_h, a_h) \leftarrow (1-\alpha_t)Q_h(x_h, a_h) + \alpha_t[r_h(x_h, a_h) + V_{h+1}(x_{h+1}) + b_t]$.
    $V_h(x_h) \leftarrow \min\{H, \max_{a' \in \mathcal{A}} Q_h(x_h, a')\}$.

Parallel to the Hoeffding case, we aim at proving an equivalent version of Lemma 5.4.3 that shows that $Q^k - Q^\star$ is (1) nonnegative and (2) bounded from above. However, unlike the Hoeffding case, this new proof becomes very delicate.

We first provide a *coarse* upper bound on $Q^k - Q^\star$ that does not assert whether $Q^k - Q^\star$ is nonnegative or not. This coarse upper bound only makes use of the fact that β_t is at most $O(\sqrt{H^3\iota/t})$, which was precisely how we have chosen β_t in the Hoeffding case and in Lemma 5.4.3.

Lemma 5.7.3 (coarse bound on $Q^k - Q^\star$)**.** *There exists absolute constant $c_2 > 0$ such that, if $\beta_t(x,a,h) \le c_2\sqrt{\frac{H^3\iota}{t}}$ in (5.13), then, with probability at least $1-p$, the following holds*

$$\forall (x,a,h,k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]\colon$$
$$(V_h^k - V_h^\star)(x_h^k) \le \alpha_t^0 H + \sum_{i=1}^{t} \alpha_t^i (V_{h+1}^{k_i} - V_{h+1}^\star)(x_{h+1}^{k_i}) + 4c_2\sqrt{\frac{H^3\iota}{t}} \quad , \quad (5.15)$$

where $t = N_h^k(x,a)$ and $k_1, \ldots, k_t < k$ are the episodes in which (x,a) was taken at step h.

Proof of Lemma 5.7.3. The result follows from Lemma 5.7.2 and the proof of Lemma 5.4.3. □

In order to apply the Bernstein concentration inequality to the recursive formula in Lemma 5.7.2, we need to estimate the variance of $V^\star$. Unfortunately, $V^\star$ is unknown as its variance. At the kth episode, we are only able to compute the "empirical" version of the variance using V^k, which is W_t as defined in (5.12).

Our next lemma shows that, if $Q^{k'} - Q^\star$ is nonnegative for all episodes $k' < k$, the variance of $V^\star$ (i.e., $\mathbb{V}_h V_{h+1}^\star(x,a)$) and the "empirical" variance of V^k are sufficiently close.

Lemma 5.7.4. *There exists an absolute constant $c > 0$ such that for any $p \in (0,1)$ and $k \in [K]$, with probability at least $1 - p/K$, if*

$$\textit{(5.15) in Lemma 5.7.3 holds and } (Q_h^{k'} - Q_h^\star)(x,a) \geq 0 \textit{ for all } k' < k,$$

then for all $(x,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$:

$$\big|\mathbb{V}_h V_{h+1}^\star(x,a) - W_t(x,a,h)\big| \leq c\Big(\frac{SA\sqrt{H^7\iota}}{t} + \sqrt{\frac{H^7SA\iota}{t}}\Big), \quad \textit{where} \quad t = N_h^k(x,a) \ .$$

Proof of Lemma 5.7.4. For each fixed $(x,a,h) \in \mathcal{S} \times \mathcal{A} \times [H]$, let us denote $k_0 = 0$, and:

$$k_i = \min\big(\big\{k \in [K] \mid k > k_{i-1} \wedge (x_h^k, a_h^k) = (x,a)\big\} \cup \{K+1\}\big) \ .$$

That is, k_i is the episode if which (x,a) was taken at step h for the ith time, and it is clearly a stopping time. Let $\mathcal{F}_i$ be the σ-field generated by all the random variables until episode k_i, step h. We also denote $t = N_h^k(x,a)$.

To bridge the gap between $\mathbb{V}_h V_{h+1}^\star(x,a)$ and $W_t(x,a,h)$, we consider following four quantities:

$$\begin{aligned}
[\mathbb{V}_h V_{h+1}^\star](x,a) =& \mathbb{E}_{x' \sim \mathbb{P}_h(\cdot|x,a)}\big[V_{h+1}^\star(x') - [\mathbb{P}_h V_{h+1}^\star](x,a)\big]^2 && =: P_1 \\
& \frac{1}{t}\sum_{i=1}^t \big[V_{h+1}^\star(x_{h+1}^{k_i}) - [\mathbb{P}_h V_{h+1}^\star](x,a)\big]^2 && =: P_2 \\
& \frac{1}{t}\sum_{i=1}^t \Big[V_{h+1}^\star(x_{h+1}^{k_i}) - \tfrac{1}{t}\textstyle\sum_{j=1}^t V_{h+1}^\star(x_{h+1}^{k_j})\Big]^2 && =: P_3 \\
W_t(x,a,h) =& \frac{1}{t}\sum_{i=1}^t \Big[V_{h+1}^{k_i}(x_{h+1}^{k_i}) - \tfrac{1}{t}\textstyle\sum_{j=1}^t V_{h+1}^{k_j}(x_{h+1}^{k_j})\Big]^2 && =: P_4 \ .
\end{aligned}$$

We shall bound the difference $|P_1 - P_4|$ by $|P_1 - P_2| + |P_2 - P_3| + |P_3 - P_4|$ via the triangle inequality.

Bounding $|P_1 - P_2|$: We notice that for any fixed $\tau \in [k]$, by the Azuma-Hoeffding inequality, there exists a sufficiently large constant $c > 0$ such that, with probability at least $1 - p/(2SAT)$:

$$\Big|\frac{1}{\tau}\sum_{i=1}^\tau \mathbb{I}\big[k_i \leq k\big] \cdot \Big[\big(V_{h+1}^\star(x_{h+1}^{k_i}) - [\mathbb{P}_h V_{h+1}^\star](x,a)\big)^2 - [\mathbb{V}_h V_{h+1}](x,a)\Big]\Big| \leq cH^2\sqrt{\iota/\tau} \ , \quad (5.16)$$

since LHS is a martingale sequence with respect to the filtration $\{\mathcal{F}_i\}$. Because Eq. (5.16) holds for all fixed $\tau \in [k]$ uniformly, it also holds for $\tau = t = N_h^k(x,a) \leq k$ which is a random variable. Also note $\mathbb{I}[k_i \leq k] = 1$ for all $i \leq N_h^k(x,a)$. Therefore, we can conclude $|P_1 - P_2| \leq cH^2\sqrt{\iota/t}$.

Bounding $|P_2 - P_3|$: We calculate

$$|P_2-P_3| \le \frac{2H}{t}\sum_{i=1}^{t}\left|[\mathbb{P}_h V^\star_{h+1}](x,a) - \tfrac{1}{t}\textstyle\sum_{j=1}^{t} V^\star_{h+1}(x^{k_j}_{h+1})\right| = 2H\left|[\mathbb{P}_h V^\star_{h+1}](x,a) - \tfrac{1}{t}\textstyle\sum_{j=1}^{t} V^\star_{h+1}(x^{k_j}_{h+1})\right| \ .$$

Again, for any fixed $\tau \in [k]$, by the Azuma-Hoeffding inequality, with probability $1 - p/(2SAT)$:

$$\left|\frac{1}{\tau}\sum_{i=1}^{\tau}\mathbb{I}\big[k_i \le k\big]\cdot\big[V^\star_{h+1}(x^{k_i}_{h+1}) - \mathbb{P}_h V^\star_{h+1}(x,a)\big]\right| \le cH\sqrt{\iota/\tau} \ . \tag{5.17}$$

By the same argument as above, we also know that Eq. (5.16) holds for the random variable $\tau = t = N^k_h(x,a) \le k$, which implies $|P_2 - P_3| \le 2cH^2\sqrt{\iota/t}$.

Bounding $|P_3 - P_4|$: We calculate that

$$\begin{aligned}|P_3 - P_4| &\le \frac{2H}{t}\sum_{i=1}^{t}\left|V^{k_i}_{h+1}(x^{k_i}_{h+1}) - V^\star_{h+1}(x^{k_i}_{h+1}) - \tfrac{1}{t}\textstyle\sum_{j=1}^{t}\big(V^{k_j}_{h+1}(x^{k_j}_{h+1}) - V^\star_{h+1}(x^{k_j}_{h+1})\big)\right| \\ &\le \frac{4H}{t}\sum_{i=1}^{t}\big|V^{k_i}_{h+1}(x^{k_i}_{h+1}) - V^\star_{h+1}(x^{k_i}_{h+1})\big| \le \frac{4H}{t}\sum_{i=1}^{t}\big(V^{k_i}_{h+1}(x^{k_i}_{h+1}) - V^\star_{h+1}(x^{k_i}_{h+1})\big) \ ,\end{aligned}$$

where the last inequality uses $V^{k'}_{h+1}(x) \ge V^\star_{h+1}(x)$ for all $x \in \mathcal{S}$ and $k' < k$, which follows from our assumption $(Q^{k'}_{h+1} - Q^\star_{h+1})(x,a) \ge 0$ for all $k' < k$.

We apply Lemma 5.7.8 (see Section 5.7 later) with a weight vector w such that $w_{k_i} = \frac{1}{t}$ for all $i \in [t]$, but $w_{k'} = 0$ for all $k' \notin \{k_1, \ldots, k_t\}$ (so $\|w\|_1 = 1$ and $\|w\|_\infty = 1/t$). This tells us that

$$|P_3 - P_4| \le \frac{4H}{t}\sum_{i=1}^{t}\big(V^{k_i}_{h+1}(x^{k_i}_{h+1}) - V^\star_{h+1}(x^{k_i}_{h+1})\big) \le O\Big(\frac{SA\sqrt{H^7\iota}}{t} + \sqrt{\frac{H^7SA\iota}{t}}\Big) \ .$$

Finally, by the triangle inequality $\big|[\mathbb{V}_h V^\star_{h+1} - W^k_h](x,a)\big| \le |P_1 - P_2| + |P_2 - P_3| + |P_3 - P_4|$, and a union bound over $(x,a,h) \in \mathcal{S}\times\mathcal{A}\times[H]$, we finish the proof. □

Now, equipped with Lemma 5.7.3 and Lemma 5.7.4, we can use induction and an Azuma-Bernstein concentration argument to prove that $Q^k - Q^\star$ is nonnegative and upper bounded by β. This gives an analog of Lemma 5.4.3 that we state here.

Lemma 5.7.5 (fine bound on $Q^k - Q^\star$)**.** *For every $p \in (0,1)$, there exists an absolute constant $c_1, c_2 > 0$ such that, under the choice of $\beta_t(x,a,h)$ in (5.13), with probability at least $1 - 2p$, the following holds simultaneously for all $(x,a,h,k) \in \mathcal{S}\times\mathcal{A}\times[H]\times[K]$:*

$$0 \le (Q^k_h - Q^\star_h)(x,a) \le \alpha^0_t H + \sum_{i=1}^{t}\alpha^i_t(V^{k_i}_{h+1} - V^\star_{h+1})(x^{k_i}_{h+1}) + \beta_t \ , \tag{5.18}$$

where $t = N^k_h(x,a)$ and $k_1, \ldots, k_t < k$ are the episodes in which (x,a) was taken at step h.

Proof of Lemma 5.7.5. We first choose $c_2 > 0$ large enough so that Lemma 5.7.3 holds with probability at least $1 - p$.

For each fixed $(x, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$, let us denote $k_0 = 0$, and:

$$k_i = \min\left(\left\{k \in [K] \mid k > k_{i-1} \wedge (x_h^k, a_h^k) = (x, a)\right\} \cup \{K+1\}\right) .$$

By the Azuma-Bernstein inequality, with probability at least $1 - p/(SAT)$, we have for all $\tau \in [K]$:

$$\begin{aligned}\left|\sum_{i=1}^{\tau} \alpha_\tau^i \mathbb{I}[k_i \le K] \cdot [(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h)V_{h+1}^\star](x,a)\right| &\le O(1) \cdot \left[\sqrt{\sum_{i=1}^{\tau}(\alpha_\tau^i)^2 [\mathbb{V}_h V_{h+1}^\star](x,a)\iota} + [\max_{i\in[\tau]} \alpha_\tau^i] H\iota\right] \\ &\le O(1) \cdot \left[\sqrt{\frac{H}{\tau}[\mathbb{V}_h V_{h+1}^\star](x,a)\iota} + \frac{H^2}{\tau}\iota\right] , \qquad (5.19)\end{aligned}$$

where the last inequality is by Lemma 2. Since the inequality (5.19) holds for all fixed $\tau \in [K]$ uniformly, it also holds for the random variable $\tau = t = N_h^k(x,a) \le K$. By a union bound, with probability at least $1 - p$, we have that for all $(x, a, h, k) \in \mathcal{S} \times \mathcal{A} \times [H] \times [K]$

$$\left|\sum_{i=1}^{t} \alpha_t^i \mathbb{I}[k_i \le K] \cdot [(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h)V_{h+1}^\star](x,a)\right| \le O(1) \cdot \left[\sqrt{\frac{H}{t}[\mathbb{V}_h V_{h+1}^\star](x,a)\iota} + \frac{H^2}{t}\iota\right] , \quad (5.20)$$

where $t = N_h^k(x,a)$ and $k_1, \ldots, k_t < k$ are the episodes in which (x,a) was taken at step h.

We are now ready to prove (5.18). We do so by induction over $k \in [K]$. Clearly, the statement is true for $k = 1$, so in the rest of the proof we assume (5.18) holds for all $k' < k$. We denote by $k_1, k_2, \ldots, k_t < k$ all indices of previous episodes where (x,a) is taken at step h. By Lemma 5.7.4, with probability $1 - p/K$, we have for all $(x, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$:

$$\left|[\mathbb{V}_h V_{h+1}^\star(x,a) - W_t(x,a,h)\right| \le O\Big(\sqrt{\frac{SAH^7\iota}{t}} + \frac{SA\sqrt{H^7\iota}}{t}\Big) .$$

Therefore, putting this into (5.20), we have

$$\left|\sum_{i=1}^{t} \alpha_t^i [(\hat{\mathbb{P}}_h^{k_i} - \mathbb{P}_h)V_{h+1}^\star](x,a)\right| \overset{①}{\le} O(1) \cdot \left[\sqrt{\frac{H}{t}(W_t(x,a,h) + H)\iota} + \frac{\sqrt{H^7SA}\cdot\iota}{t}\right] \overset{②}{\le} \frac{\beta_t}{2} ,$$

where inequality ① uses $\sqrt{\frac{H^7SA\iota}{t}} \le H + \frac{H^6SA\iota}{t}$, and inequality ② is due to our choice of β_t in (5.13) and the sufficiently large choice of $c_1 > 0$.

Finally, applying the above inequality to Lemma 5.7.2, we have for all $(x, a, h) \in \mathcal{S} \times \mathcal{A} \times [H]$

$$0 \le (Q_h^k - Q_h^\star)(x,a) - \alpha_t^0(H - Q_h^\star(x,a)) - \sum_{i=1}^{t} \alpha_t^i \big[(V_{h+1}^{k_i} - V_{h+1}^\star)(x_{h+1}^{k_i})\big] \le \beta_t . \qquad (5.21)$$

This proves that (5.18) holds for k with probability at least $1-p/K$. By induction, we know (5.18) holds for all $k \in [K]$ with probability at least $1-p$. Combining this with the $1-p$ probability event for (5.20), we finish the proof that Lemma 5.7.5 holds with probability at least $1-2p$. □

As mentioned in Section 5.3, the key reason why a Bernstein approach can improve by a factor of $\sqrt{H}$ is that, although the value function at each step is at most H, the "total variance of the value function" for an entire episode is at most $O(H^2)$. Or more simply, the total variance for all steps is at most $O(HT)$. This is captured directly in the following lemma.

Lemma 5.7.6. *There exists an absolute constant c, such that with probability at least $1-p$:*

$$\sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{V}_h V_{h+1}^{\pi_k}(x_h^k, a_h^k) \le c(HT + H^3\iota) \ .$$

Proof of Lemma 5.7.6. First, we note for any fixed policy π and initial state x_1, suppose $(x_2, \cdots, x_h)$ is a sequence generated by following policy π starting at x_1, then

$$\begin{aligned} H^2 &\ge \mathbb{E}\left[\Big(\textstyle\sum_{h=1}^{H} r(x_h, \pi(x_h))\Big) - V_1^{\pi}(x_1)\right]^2 \\ &\overset{①}{=} \mathbb{E}\left[\textstyle\sum_{h=1}^{H}[r(x_h, \pi(x_h)) + V_{h+1}^{\pi}(x_{h+1}) - V_h^{\pi}(x_h)]\right]^2 \\ &\overset{②}{=} \mathbb{E}\textstyle\sum_{h=1}^{H}\left[r(x_h, \pi(x_h)) + V_{h+1}^{\pi}(x_{h+1}) - V_h^{\pi}(x_h)\right]^2 = \mathbb{E}\textstyle\sum_{h=1}^{H} \mathbb{V}_h V_{h+1}^{\pi}(x_h, \pi(x_h)) \ , \end{aligned}$$

where equality ① is because $V_{H+1}^{\pi} = 0$, and equality ② uses the independence due to the Markov property. Therefore, letting $\mathcal{F}_{k-1}$ be the σ-field generated by all the random variables over the first $k-1$ episodes, at the kth episode we have:

$$\mathbb{E}\left[X_k \,\middle|\, \mathcal{F}_{k-1}\right] \le H^2 \quad \text{where} \quad X_k := \textstyle\sum_{h=1}^{H} \mathbb{V}_h V_{h+1}^{\pi_k}(x_h^k, \pi_k(x_h^k)) \ .$$

Also, note that $|X_k| \le H^3$ and $\mathrm{Var}[X_k \,|\, \mathcal{F}_{k-1}] \le H^3\mathbb{E}[X_k \,|\, \mathcal{F}_{k-1}] \le H^5$. Therefore, by an Azuma-Bernstein inequality on $X_1 + \cdots + X_K$ with respect to filtration $\{\mathcal{F}_k\}_{k\ge 0}$, we have with probability at least $1-p$,

$$\sum_{k=1}^{K}\sum_{h=1}^{H} \mathbb{V}_h V_{h+1}^{\pi_k}(x_h^k, a_h^k) \le \sum_{k=1}^{K} \mathbb{E}\big[X_k \,|\, \mathcal{F}_{k-1}\big] + O\big(\sqrt{H^5 K\iota} + H^3\iota\big) \le O(HT + H^3\iota) \ ,$$

where the last step is by $ab \le a^2 + b^2$. □

Our last lemma shows that the "empirical" variance of V^k (i.e., $W_t(x,a,h)$) is also upper bounded by the variance $\mathbb{V}_h V_{h+1}^{\pi_k}(x,a)$ (which appeared in Lemma 5.7.6) plus some small terms.

Lemma 5.7.7. *There exist absolute constants $c_1, c_2, c > 0$ such that, letting $(x, a) = (x_h^k, a_h^k)$ and $t = n_h^k = N_h^k(x, a)$, we have that for all $(k, h) \in [K] \times [H]$, with probability at least $1-4p$,*

$$W_t(x,a,h) \le \mathbb{V}_h V_{h+1}^{\pi_k}(x,a) + 2H(\delta_{h+1}^k + \xi_{h+1}^k) + c\Big(\frac{SA\sqrt{H^7\iota}}{t} + \sqrt{\frac{SAH^7\iota}{t}}\Big) \ ,$$

where $\xi_{h+1}^k := [(\mathbb{P}_h - \hat{\mathbb{P}}_h^k)(V_{h+1}^\star - V_{h+1}^k)](x_h^k, a_h^k)$ and $\delta_{h+1}^k := (V_{h+1}^\star - V_{h+1}^k)(x_{h+1}^k)$.

Proof of Lemma 5.7.7. We first assume that Lemma 5.7.5 holds (which happens with probability at least $1-2p$) and Lemma 5.7.3 holds (which happens with probability at least $1-p$). As a consequence, with probability at least $1-p$, Lemma 5.7.4 also holds for all $k \in [K]$. By the triangle inequality, we have:

$$W_t(x,a,h) - \mathbb{V}_h V_{h+1}^{\pi_k}(x,a) \le \big|[\mathbb{V}_h V_{h+1}^\star - W_t(x,a,h)\big| + \big|[\mathbb{V}_h V_{h+1}^\star - \mathbb{V}_h V_{h+1}^{\pi_k}](x,a)\big| \ ,$$

where the first term on the right-hand side is upper bounded by Lemma 5.7.4. For the second term:

$$\big|[\mathbb{V}_h V_{h+1}^\star - \mathbb{V}_h V_{h+1}^{\pi_k}](x,a)\big| \le 2H[\mathbb{P}_h(V_{h+1}^\star - V_{h+1}^{\pi_k})](x_h^k, a_h^k) = 2H(\xi_{h+1}^k + \delta_{h+1}^k) \ . \qquad \square$$

Proof of Theorem 5.3.2

We are now ready to prove Theorem 5.3.2. Again, the proof decomposes the regret in a recursive form, and carefully controls the error propagation via repeated usage of Lemma 5.7.5 and Lemma 5.7.7.

Proof of Theorem 5.3.2. We first assume that Lemma 5.7.6 holds (which happens with probability at least $1-4p$) and Lemma 5.7.7 holds (which happens with probability at least $1-p$).

By the same argument as in the proof of Theorem 5.3.1 (in particular, inequality (5.9)) we have:

$$\sum_{k=1}^K \delta_h^k \le \left(1 + \frac{1}{H}\right) \sum_{k=1}^K \delta_{h+1}^k + SAH + \sum_{k=1}^K (\beta_{n_h^k}(x_h^k, a_h^k, h) + \xi_{h+1}^k) \ ,$$

where $\xi_{h+1}^k := [(\mathbb{P}_h - \hat{\mathbb{P}}_h^k)(V_{h+1}^\star - V_{h+1}^k)](x_h^k, a_h^k)$ and $\delta_{h+1}^k := (V_{h+1}^\star - V_{h+1}^k)(x_{h+1}^k)$. As a result, for any $h \in H$, by recursing the above formula for $h, h+1, \ldots, H$, we have:

$$\sum_{k=1}^K \delta_h^k \le SAH^2 + \sum_{h'=h}^H \sum_{k=1}^K (\beta_{n_{h'}^k}(x_{h'}^k, a_{h'}^k, h') + \xi_{h'+1}^k) \tag{5.22}$$

By the Azuma-Hoeffding inequality, with probability $1-p$, we have:

$$\forall h \in [H]\colon \Big|\sum_{h'=h}^H \sum_{k=1}^K \xi_{h'+1}^k\Big| = \Big|\sum_{h'=h}^H \sum_{k=1}^K [(\mathbb{P}_{h'} - \hat{\mathbb{P}}_{h'}^k)(V_{h'+1}^\star - V_{h'+1}^k)](x_{h'}^k, a_{h'}^k)\Big| \le O\big(H\sqrt{T\iota}\big) \ . \tag{5.23}$$

Also, recall $\beta_t(x,a,h) \le c\sqrt{H^3\iota/t}$ so $\sum_{k=1}^K \beta_{n_h^k} \le O\big(\sqrt{H^2SAT\iota}\big)$ according to (5.10). Putting these into (5.22), we derive that $\sum_{k=1}^K \delta_h^k \le O\big(SAH^2 + \sqrt{H^4SAT\iota}\big)$. Note when $T \ge \sqrt{H^4SAT\iota}$, we have $\sqrt{H^4SAT\iota} \ge H^2SA$; when $T \le \sqrt{H^4SAT\iota}$, we have $\sum_{k=1}^K \delta_h^k \le HK = T \le \sqrt{H^4SAT\iota}$. Therefore, we can simply write

$$\sum_{k=1}^K \delta_h^k \le O\big(\sqrt{H^4SAT\iota}\big) \ . \tag{5.24}$$

By our choice of β_t, we have:

$$\sum_{k=1}^K \sum_{h=1}^H \beta_{n_h^k} \le O(1) \cdot \sum_{k=1}^K \sum_{h=1}^H \left[\sqrt{\frac{H}{n_h^k} \cdot (W_{n_h^k}(x,a,h) + H)} + \frac{\sqrt{H^7SA} \cdot \iota}{n_h^k} \right] \tag{5.25}$$

The summation of the second term in (5.25) is upper bounded by

$$\sum_{k=1}^K \sum_{h=1}^H \frac{\sqrt{H^7SA} \cdot \iota}{n_h^k} \le \sqrt{H^9S^3A^3\iota^4} \ ,$$

because $1 + \frac{1}{2} + \frac{1}{3} + \cdots \le \iota$. The summation of the first term in (5.25) can be upper bounded by

$$\begin{aligned} \sum_{k=1}^K \sum_{h=1}^H \sqrt{\frac{H}{n_h^k} \cdot (W_{n_h^k}(x,a,h) + H)} \le & \sqrt{\left(\sum_{k=1}^K \sum_{h=1}^H (W_{n_h^k}(x,a,h) + H) \right) \left(\sum_{k=1}^K \sum_{h=1}^H \frac{H}{n_h^k} \right)} \\ \le & \sqrt{\sum_{k=1}^K \sum_{h=1}^H W_{n_h^k}(x,a,h)} \cdot \sqrt{H^2SA\iota} + \sqrt{H^3SAT\iota} \ . \end{aligned} \tag{5.26}$$

We calculate

$$\begin{aligned} \sum_{k=1}^K \sum_{h=1}^H W_{n_h^k}(x,a,h) \overset{①}{\le} & \sum_{k=1}^K \sum_{h=1}^H \left[\mathbb{V}_h V_{h+1}^{\pi_k}(x_h^k, a_h^k) + 2H(\delta_{h+1}^k + \xi_{h+1}^k) + O\Big(\frac{SA\sqrt{H^7\iota}}{n_h^k} + \sqrt{\frac{SAH^7\iota}{n_h^k}} \Big) \right] \\ \overset{②}{\le} & \sum_{k=1}^K \sum_{h=1}^H \big[\mathbb{V}_h V_{h+1}^{\pi_k}(x_h^k, a_h^k) + 2H(\delta_{h+1}^k + \xi_{h+1}^k) \big] + O\Big(S^2A^2\sqrt{H^9\iota^3} + SA\sqrt{H^8T\iota} \Big) \\ \overset{③}{\le} & 2H \sum_{k=1}^K \sum_{h=1}^H (\delta_{h+1}^k + \xi_{h+1}^k) + O\Big(HT + H^3\iota + S^2A^2\sqrt{H^9\iota^3} + SA\sqrt{H^8T\iota} \Big) \\ \overset{④}{\le} & O\Big(\sqrt{H^8SAT\iota} + HT + H^3\iota + S^2A^2\sqrt{H^9\iota^3} + SA\sqrt{H^8T\iota} \Big) \\ \le & O\Big(HT + S^2A^2H^7\iota + S^2A^2\sqrt{H^9\iota^3} \Big) \ . \end{aligned} \tag{5.27}$$

Here, inequality ① uses Lemma 5.7.7; inequality ② uses $\sum_{k=1}^{K}(n_h^k)^{-1} \le SA\iota$ and $\sum_{k=1}^{K}(\sqrt{n_h^k})^{-1/2} \le O(\sqrt{KSA})$; inequality ③ uses Lemma 5.7.6; and inequality ④ uses (5.23) and (5.24).

Putting (5.27) and (5.26) back to (5.25), we have

$$\sum_{k=1}^{K}\sum_{h=1}^{H}\beta_{n_h^k} \le O\Big(\sqrt{H^3SAT\iota} + \sqrt{S^3A^3H^9\iota^4}\Big) \ . \tag{5.28}$$

Finally, putting this and (5.23) back to (5.22), we finish the proof that with probability at least $1-6p$, for every $h \in [H]$

$$\sum_{k=1}^{K}\delta_h^k \le O\Big(\sqrt{H^3SAT\iota} + \sqrt{S^3A^3H^9\iota^4}\Big) \ .$$

Since we also have $\text{Regret}(K) \le \sum_{k=1}^{K}\delta_1^k$ as in the proof of Theorem 5.3.1, rescaling p to $p/6$ finishes the proof. □

Proof of Auxiliary Lemma

The next lemma shows how the weighted sum over $(V_h^k - V_h^\star)(x_h^k)$ is upper bounded by the infinity norm and the one-norm of the weights w. This lemma provides the key to prove Lemma 5.7.4.

Lemma 5.7.8. *Suppose (5.15) in Lemma 5.7.3 holds. For any $h \in [H]$, let $\phi_h^k := (V_h^k - V_h^\star)(x_h^k)$, and letting $w = (w_1, \ldots, w_k)$ be a nonnegative weight vector, we have:*

$$\sum_{k=1}^{K} w_k\phi_h^k \le O\big(SA\|w\|_\infty\sqrt{H^5\iota} + \sqrt{SA\|w\|_1\|w\|_\infty H^5\iota}\big) \ ,$$

where $\phi_h^k := (V_h^k - V_h^\star)(x_h^k)$.

Proof of Lemma 5.7.8. For any fixed $(k,h) \in [K]\times[H]$, let $t = N_h^k(x_h^k, a_h^k)$, and suppose (x_h^k, a_h^k) was previously taken at step h of episodes $k_1, \ldots, k_t < k$. We then have, for some absolute constant c:

$$\phi_h^k = (V_h^k - V_h^\star)(x_h^k) \overset{①}{\le} (Q_h^k - Q_h^\star)(x_h^k, a_h^k) \overset{②}{\le} \alpha_t^0 H + \sum_{i=1}^{t}\alpha_t^i\phi_{h+1}^{k_i} + O\Big(\sqrt{\frac{H^3\iota}{t}}\Big) \ . \tag{5.29}$$

Here, inequality ① holds from $V_h^k(x_h^k) \le \max_{a'\in\mathcal{A}} Q_h^k(x_h^k, a') = Q_h^k(x_h^k, a_h^k)$ and the Bellman optimality equation $V_h^\star(x_h^k) = \max_{a'\in\mathcal{A}} Q_h^\star(x_h^k, a') \ge Q_h^\star(x_h^k, a_h^k)$. Inequality ② holds by the assumption that (5.15) in Lemma 5.7.3 holds.

Next, let us compute the summation $\sum_{k=1}^{K} w_k \delta_h^k$. Denoting $n_h^k = N_h^k(x_h^k, a_h^k)$, we have:

$$\sum_{k=1}^{K} w_k \alpha_{n_h^k}^0 H = \sum_{k=1}^{K} H w_k \cdot \mathbb{I}[n_h^k = 0] \le HSA\|w\|_\infty \ ; \text{ and} \tag{5.30}$$

$$\begin{aligned}\sum_{k=1}^{K} w_k \sqrt{\frac{H^3\iota}{n_h^k}} &\overset{①}{=} O(1) \cdot \sum_{x,a} \sum_{i=1}^{N_h^K(x,a)} w_{k_i(x,a)} \sqrt{\frac{H^3\iota}{i}} \\ &\overset{②}{\le} O\big(SA\|w\|_\infty + \sqrt{SA\|w\|_1\|w\|_\infty}\big) \cdot \sqrt{H^3\iota} \ .\end{aligned} \tag{5.31}$$

Above,

- Equality ① is by reordering the indices $k \in [K]$ so that the ones with the same $(x, a) = (x_h^k, a_h^k)$ are grouped together; and we denote by $k_i(x, a) = k$ where k is the ith episode where (x, a) is taken at step h.
- Inequality ② is because $\sum_{x,a} \sum_{i=1}^{N_h^K(x,a)} w_{k_i(x,a)} = \|w\|_1$. Therefore, the left-hand side of ② is maximized when the weights are distributed to those indices i that have smaller values:

$$\sum_{x,a} \sum_{i=1}^{N_h^K(x,a)} w_{k_i(x,a)} \sqrt{\frac{1}{i}} \le \|w\|_1 + \sum_{x,a} \sum_{i=1}^{\left\lfloor \frac{\|w\|_1}{SA\|w\|_\infty} \right\rfloor} \|w\|_\infty \sqrt{\frac{1}{i}} \le O\big(SA\|w\|_\infty + \sqrt{SA\|w\|_1\|w\|_\infty}\big) \ .$$

To bound the second term in (5.29), which is

$$\sum_{k=1}^{K} w_k \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \phi_{h+1}^{k_i(x_h^k, a_h^k)} \ , \tag{5.32}$$

we regroup the summands in (5.32) in a different way. For every $k' \in [K]$, we group all terms $\phi_{h+1}^{k'}$ that appear in the inner summand of (5.32)—denoting their total weight by $w'_{k'}$—and write:

$$\sum_{k=1}^{K} w_k \sum_{i=1}^{n_h^k} \alpha_{n_h^k}^i \phi_{h+1}^{k_i(x_h^k, a_h^k)} = \sum_{k'=1}^{K} w'_{k'} \cdot \phi_{h+1}^{k'} \ . \tag{5.33}$$

We make two key observations

- We have $\|w'\|_1 \le \|w\|_1$ because $\sum_{i=1}^{t} \alpha_t^i \le 1$.
- For every $k' \in [K]$, we note that the term $\phi_{h+1}^{k'}$ only appears on the left-hand side of (5.33) in episode $k \ge k'$, where $(x_h^k, s_h^k) = (x_h^{k'}, s_h^{k'})$. Suppose it appears in episodes

$k'_1, k'_2, \ldots$. Then, letting $\tau = n_h^{k'}$, we have corresponding weight is $w_{k'}\alpha_\tau^\tau, w_{k'_1}\alpha_{\tau+1}^\tau, w_{k'_2}\alpha_{\tau+2}^\tau \cdots$. Therefore, the total weight satisfies

$$w'_{k'} \le \|w\|_\infty \sum_{t=n_h^{k'}+1}^{\infty} \alpha_t^{n_h^{k'}} \le \left(1 + \frac{1}{H}\right)\|w\|_\infty \ ,$$

where the final inequality uses $\sum_{t=i}^{\infty} \alpha_t^i = 1 + \frac{1}{H}$ from Lemma 3.

Plugging (5.30), (5.31), and (5.33) back into (5.29), we have:

$$\sum_{k=1}^{K} w_k \phi_h^k \le HSA\|w\|_\infty + \sum_{k'=1}^{K} w'_{k'} \cdot \phi_{h+1}^{k'} + O\big(SA\|w\|_\infty + \sqrt{SA\|w\|_1\|w\|_\infty}\big) \cdot \sqrt{H^3\iota} \ ,$$

with $\|w'\|_\infty \le (1 + \frac{1}{H})\|w\|_\infty$ and $\|w'\|_1 \le \|w\|_\infty$. Recursing this for $h, h+1, \ldots, H$, we conclude that

$$\sum_{k=1}^{K} w_k \phi_h^k \le O\big(SA\|w\|_\infty \sqrt{H^5\iota} + \sqrt{SA\|w\|_1\|w\|_\infty H^5\iota}\big) \ . \qquad \square$$

5.8 Proof of Lower Bound

Recall that Jaksch, Ortner, and Auer (2010) showed that for any algorithm, there is an MDP with diameter D, S states and A actions, such that the algorithm's regret must be at least $\Omega(\sqrt{DSAT})$. The natural analogous notion of the diameter in the episodic setting is H, and thus this suggests a lower bound in $\Omega(\sqrt{HSAT})$, as presented in (Osband and Van Roy, 2016; Azar, Osband, and Munos, 2017).

We show that, in our episodic setting of this work, one can obtain a stronger lower bound:

Theorem 5.3.3. *For the episodic MDP problem studied in this work, the expected regret for any algorithm must be at least* $\Omega(\sqrt{H^2SAT})$.

This result seemingly contradicts the $O(\sqrt{HSAT})$ regret bound of Azar, Osband, and Munos (2017). There is no contradiction, however, because Azar, Osband, and Munos (2017) assumes that the transition matrix $\mathbb{P}_h$ is the *same* at each step $h \in [H]$. On the contrary, in this work we consider the more general setting where the transition matrices $\mathbb{P}_1, \ldots, \mathbb{P}_H$ are distinct for each step. Our setting can be viewed as a special case of the non-episodic MDP studied by Jaksch, Ortner, and Auer (2010), obtained by augmenting the state space to $\mathcal{S}' = \mathcal{S} \times [H]$.

Rather than providing a formal proof of Theorem 5.3.3 we give the intuition behind the construction and its analysis. The formalization itself is an easy exercise following well-known lower-bound techniques from the multi-armed bandit literature; see, e.g., (Bubeck

and Cesa-Bianchi, 2012). For the sake of simplicity, we consider $A = 2$ and $S = 2$ (again the generalization to arbitrary A and S is routine).

We start by recalling the construction from Jaksch, Ortner, and Auer (2010), which we will refer to as the "JAO MDP." The reward does not depend on actions: state 1 always has reward 1 and state 0 always has reward 0. From state 1, any action takes the agent to state 0 with probability δ, and to state 1 with probability $1 - \delta$. In state 0, there is one action $a^\star$ takes the agent to state 1 with probability $\delta + \epsilon$, and the other action a takes the agent to 1 with probability δ. A standard Markov chain exercise shows that the stationary distribution of the optimal policy (that is, the one that in state 0 takes action $a^\star$) has a probability of being in state 1 of

$$\frac{\frac{1}{\delta}}{\frac{1}{\delta} + \frac{1}{\delta+\epsilon}} = \frac{\delta + \epsilon}{2\delta + \epsilon} \geq \frac{1}{2} + \frac{\epsilon}{6\delta} \text{ for } \epsilon \leq \delta \,.$$

In contrast, acting sub-optimally (that is, taking action a in state 0) leads to a uniform distribution over the two states, or equivalently a regret per time step of order ϵ/δ. Moreover, in order to identify the two actions $a, a^\star$ (each with probability δ and $\delta + \varepsilon$), the number of observations in state 0 needs to be at least $\Omega(\delta/\epsilon^2)$. Thus, taking the latter quantity to be T, one obtains the following lower bound on total regret:

$$T \times \Omega(\epsilon/\delta) = \Omega(\sqrt{T/\delta}) \,.$$

In the JAO MDP, the diameter is $D = \Theta(1/\delta)$. This proves the $\sqrt{DT}$ lower bound from Jaksch, Ortner, and Auer (2010).

The natural analogue of the JAO MDP for the episodic setting is to put the JAO MDP in "series" for H steps (in other words, one takes H steps in the JAO MDP and then restarts, say starting in state 0). The main difference with the non-episodic version is that, in H steps, one may not have time to *mix*, i.e., to reach the stationary distribution over the two states. Using standard theory of Markov chains, one can show that the optimal policy on this episodic MDP has a mixing time of $\Theta(1/\delta)$. By choosing H to be slightly larger than $\Theta(1/\delta)$, we have a sufficient number of steps (in each episode) to mix, and thus the previous non-episodic argument remains valid for the episodic case. This leads to a lower bound $\Omega(\sqrt{HT})$ for the episodic case, as illustrated by (Osband and Van Roy, 2016; Azar, Osband, and Munos, 2017).

Finally, recall that in our episodic setting, the transition matrices $\mathbb{P}_1, \ldots, \mathbb{P}_H$ may not necessarily be the same. Therefore, we can further strengthen this lower bound to $\Omega(H\sqrt{T})$ in the following way.

Let us use H *distinct* JAO MDPs, each with a different optimal action $a_h^\star$, when putting them in series. In other words, for at least half of the steps $h \in H$, one has to identify the correct action $a_h^\star$ for that specific step. (If not, the per-iteration regret will again be $\Omega(\varepsilon/\delta)$.) However the number of observations in that specific step h is only T/H, and thus one now needs to take $T/H = O(\delta/\epsilon^2)$ (instead of $T = \Omega(\delta/\epsilon^2)$ previously). This gives the claimed $\Omega\big(H\sqrt{T}\big)$ lower bound.

Bibliography

[1] Leonard Adolphs et al. "Local Saddle Point Optimization: A Curvature Exploitation Approach". In: *arXiv preprint arXiv:1805.05751* (2018).

[2] Naman Agarwal et al. "Finding approximate local minima faster than gradient descent". In: *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*. ACM. 2017, pp. 1195–1199.

[3] Shipra Agrawal and Randy Jia. "Optimistic posterior sampling for reinforcement learning: worst-case regret bounds". In: *Conference on Neural Information Processing Systems*. Curran Associates Inc. 2017, pp. 1184–1194.

[4] Zeyuan Allen-Zhu. "Natasha 2: Faster non-convex optimization than sgd". In: *Advances in Neural Information Processing Systems*. 2018, pp. 2680–2691.

[5] Zeyuan Allen-Zhu and Yuanzhi Li. "Neon2: Finding Local Minima via First-Order Oracles". In: *arXiv preprint arXiv:1711.06673* (2017).

[6] Zeyuan Allen-Zhu and Lorenzo Orecchia. "Linear coupling: An ultimate unification of gradient and mirror descent". In: *arXiv preprint arXiv:1407.1537* (2014).

[7] Animashree Anandkumar and Rong Ge. "Efficient approaches for escaping higher order saddle points in non-convex optimization". In: *Conference on learning theory*. 2016, pp. 81–102.

[8] Sanjeev Arora et al. "Generalization and equilibrium in generative adversarial nets (gans)". In: *arXiv preprint arXiv:1703.00573* (2017).

[9] Mohammad Azar, Rémi Munos, and Hilbert J. Kappen. "Minimax PAC bounds on the sample complexity of reinforcement learning with a generative model". In: *Machine Learning* 91.3 (2013), pp. 325–349.

[10] Mohammad Azar, Rémi Munos, and Hilbert J. Kappen. "On the sample complexity of reinforcement learning with a generative model". In: *Proceedings of the 29th International Conference on Machine Learning (ICML)*. 2012.

[11] Mohammad Azar, Ian Osband, and Rémi Munos. "Minimax Regret Bounds for Reinforcement Learning". In: *Proceedings of the 34th International Conference on Machine Learning (ICML)*. 2017, pp. 263–272.

[12] Mohammad Azar et al. "Speedy Q-learning". In: *Conference on Neural Information Processing Systems*. Curran Associates Inc. 2011, pp. 2411–2419.

[13] Afonso S Bandeira, Nicolas Boumal, and Vladislav Voroninski. "On the low-rank approach for semidefinite programs arising in synchronization and community detection". In: *Conference on Learning Theory*. 2016, pp. 361–382.

[14] Amir Beck and Marc Teboulle. "A fast iterative shrinkage-thresholding algorithm for linear inverse problems". In: *SIAM Journal on Imaging Sciences* 2.1 (2009), pp. 183–202.

[15] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. 1st. Athena Scientific, 1995. ISBN: 1886529124.

[16] Srinadh Bhojanapalli, Behnam Neyshabur, and Nati Srebro. "Global optimality of local search for low rank matrix recovery". In: *Advances in Neural Information Processing Systems*. 2016, pp. 3873–3881.

[17] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.

[18] Nicolas Boumal, Vlad Voroninski, and Afonso Bandeira. "The non-convex Burer-Monteiro approach works on smooth semidefinite programs". In: *Advances in Neural Information Processing Systems*. 2016, pp. 2757–2765.

[19] Anton Bovier et al. "Metastability in reversible diffusion processes I: Sharp asymptotics for capacities and exit times". In: *Journal of the European Mathematical Society* 6.4 (2004), pp. 399–424.

[20] Sébastien Bubeck. "Convex optimization: Algorithms and complexity". In: *Foundations and Trends® in Machine Learning* 8.3-4 (2015), pp. 231–357.

[21] Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. "Regret analysis of stochastic and nonstochastic multi-armed bandit problems". In: *Foundations and Trends in Machine Learning* 5.1 (2012), pp. 1–122.

[22] Sébastien Bubeck, Yin Tat Lee, and Mohit Singh. "A geometric alternative to Nesterov's accelerated gradient descent". In: *arXiv preprint arXiv:1506.08187* (2015).

[23] Yair Carmon and John C Duchi. "Gradient Descent Efficiently Finds the Cubic-Regularized Non-Convex Newton Step". In: *arXiv preprint arXiv:1612.00547* (2016).

[24] Yair Carmon et al. "Accelerated Methods for Non-Convex Optimization". In: *arXiv preprint arXiv:1611.00756* (2016).

[25] Yair Carmon et al. "Convex until Proven Guilty: Dimension-Free Acceleration of Gradient Descent on Non-Convex Functions". In: *arXiv preprint arXiv:1705.02766* (2017).

[26] Yair Carmon et al. "Lower bounds for finding stationary points I". In: *arXiv preprint arXiv:1710.11606* (2017).

[27] Yair Carmon et al. "Lower Bounds for Finding Stationary Points II: First-Order Methods". In: *arXiv preprint arXiv:1711.00841* (2017).

[28] Coralia Cartis, Nicholas Gould, and Ph L Toint. "On the complexity of steepest descent, Newton's and regularized Newton's methods for nonconvex unconstrained optimization problems". In: *Siam journal on optimization* 20.6 (2010), pp. 2833–2852.

[29] Louis Augustin Cauchy. "Méthode générale pour la résolution des systémes d'équations simultanees". In: *C. R. Acad. Sci. Paris* (1847).

[30] Robert S Chen et al. "Robust optimization for non-convex objectives". In: *Advances in Neural Information Processing Systems*. 2017, pp. 4705–4714.

[31] Ashish Cherukuri, Bahman Gharesifard, and Jorge Cortes. "Saddle-point dynamics: conditions for asymptotic stability of saddle points". In: *SIAM Journal on Control and Optimization* 55.1 (2017), pp. 486–511.

[32] Anna Choromanska et al. "The Loss Surface of Multilayer Networks". In: *arXiv:1412.0233* (2014).

[33] Frank E Curtis, Daniel P Robinson, and Mohammadreza Samadi. "A trust region algorithm with a worst-case iteration complexity of $O(\epsilon^{-3/2})$ for nonconvex optimization". In: *Mathematical Programming* (2014), pp. 1–32.

[34] Hadi Daneshmand et al. "Escaping Saddles with Stochastic Gradients". In: *arXiv preprint arXiv:1803.05999* (2018).

[35] Constantinos Daskalakis and Ioannis Panageas. "The Limit Points of (Optimistic) Gradient Descent in Min-Max Optimization". In: *Advances in Neural Information Processing Systems*. 2018, pp. 9256–9266.

[36] Constantinos Daskalakis et al. "Training GANs with Optimism". In: *arXiv preprint arXiv:1711.00141* (2017).

[37] Yann N Dauphin et al. "Identifying and attacking the saddle point problem in high-dimensional non-convex optimization". In: *Advances in Neural Information Processing Systems*. 2014, pp. 2933–2941.

[38] Damek Davis and Dmitriy Drusvyatskiy. "Stochastic subgradient method converges at the rate $O(k^{\frac{-1}{4}})$ on weakly convex functions". In: *arXiv preprint arXiv:1802.02988* (2018).

[39] Marc Deisenroth and Carl E Rasmussen. "PILCO: A model-based and data-efficient approach to policy search". In: *Proceedings of the 28th International Conference on machine learning (ICML)*. 2011, pp. 465–472.

[40] Simon S Du et al. "Gradient descent can take exponential time to escape saddle points". In: *Advances in neural information processing systems*. 2017, pp. 1067–1077.

[41] Eyal Even-Dar and Yishay Mansour. "Learning rates for Q-learning". In: *Journal of Machine Learning Research* 5.Dec (2003), pp. 1–25.

[42] Cong Fang, Zhouchen Lin, and Tong Zhang. "Sharp Analysis for Nonconvex SGD Escaping from Saddle Points". In: *arXiv preprint arXiv:1902.00247* (2019).

[43] Cong Fang et al. "Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator". In: *Advances in Neural Information Processing Systems*. 2018, pp. 687–697.

[44] Maryam Fazel et al. "Global Convergence of Policy Gradient Methods for Linearized Control Problems". In: *arXiv preprint arXiv:1801.05039* (2018).

[45] Uriel Feige, Yishay Mansour, and Robert Schapire. "Learning and inference in the presence of corrupted inputs". In: *Conference on Learning Theory*. 2015, pp. 637–657.

[46] Rong Ge, Chi Jin, and Yi Zheng. "No Spurious Local Minima in Nonconvex Low Rank Problems: A Unified Geometric Analysis". In: *International Conference on Machine Learning*. 2017, pp. 1233–1242.

[47] Rong Ge, Jason D Lee, and Tengyu Ma. "Matrix completion has no spurious local minimum". In: *Advances in Neural Information Processing Systems*. 2016, pp. 2973–2981.

[48] Rong Ge et al. "Escaping from saddle pointsonline stochastic gradient for tensor decomposition". In: *Conference on Learning Theory*. 2015, pp. 797–842.

[49] Saeed Ghadimi and Guanghui Lan. "Accelerated gradient methods for nonconvex nonlinear and stochastic programming". In: *Mathematical Programming* 156.1-2 (2016), pp. 59–99.

[50] Saeed Ghadimi and Guanghui Lan. "Stochastic first-and zeroth-order methods for nonconvex stochastic programming". In: *SIAM Journal on Optimization* 23.4 (2013), pp. 2341–2368.

[51] Gauthier Gidel et al. "Negative momentum for improved game dynamics". In: *arXiv preprint arXiv:1807.04740* (2018).

[52] Irving L Glicksberg. "A further generalization of the Kakutani fixed point theorem, with application to Nash equilibrium points". In: *Proceedings of the American Mathematical Society* 3.1 (1952), pp. 170–174.

[53] Alon Gonen and Elad Hazan. "Learning in Non-convex Games with an Optimization Oracle". In: *arXiv preprint arXiv:1810.07362* (2018).

[54] Ian Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[55] Paulina Grnarova et al. "An online learning approach to generative adversarial networks". In: *arXiv preprint arXiv:1706.03269* (2017).

[56] Elad Hazan. "Introduction to online convex optimization". In: *Foundations and Trends® in Optimization* 2.3-4 (2016), pp. 157–325.

[57] Kaiming He et al. "Deep residual learning for image recognition". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[58] Martin Heusel et al. "Gans trained by a two time-scale update rule converge to a local nash equilibrium". In: *Advances in Neural Information Processing Systems*. 2017, pp. 6626–6637.

[59] Sepp Hochreiter and Jürgen Schmidhuber. "Long short-term memory". In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[60] Ya-Ping Hsieh, Chen Liu, and Volkan Cevher. "Finding Mixed Nash Equilibria of Generative Adversarial Networks". In: *arXiv preprint arXiv:1811.02002* (2018).

[61] Thomas Jaksch, Ronald Ortner, and Peter Auer. "Near-optimal Regret Bounds for Reinforcement Learning". In: *Journal of Machine Learning Research* 11 (2010), pp. 1563–1600.

[62] Nan Jiang et al. "Contextual Decision Processes with Low Bellman Rank are PAC-Learnable". In: *arXiv preprint arXiv:1610.09512* (2016).

[63] Chi Jin, Praneeth Netrapalli, and Michael I Jordan. "Accelerated gradient descent escapes saddle points faster than gradient descent". In: *arXiv preprint arXiv:1711.10456* (2017).

[64] Chi Jin, Praneeth Netrapalli, and Michael I Jordan. "Minmax Optimization: Stable Limit Points of Gradient Descent Ascent are Locally Optimal". In: *arXiv preprint arXiv:1902.00618* (2019).

[65] Chi Jin et al. "A Short Note on Concentration Inequalities for Random Vectors with SubGaussian Norm". In: *arXiv preprint arXiv:1902.03736* (2019).

[66] Chi Jin et al. "How to Escape Saddle Points Efficiently". In: *International Conference on Machine Learning*. 2017, pp. 1724–1732.

[67] Chi Jin et al. "Is q-learning provably efficient?" In: *Advances in Neural Information Processing Systems*. 2018, pp. 4863–4873.

[68] Chi Jin et al. "Stochastic Gradient Descent Escapes Saddle Points Efficiently". In: *arXiv preprint arXiv:1902.04811* (2019).

[69] Sham Kakade. "On the sample complexity of reinforcement learning". PhD thesis. University College London, England, 2003.

[70] Sham Kakade, Mengdi Wang, and Lin F Yang. "Variance Reduction Methods for Sublinear Reinforcement Learning". In: *ArXiv e-prints* abs/1802.09184 (Apr. 2018).

[71] Kenji Kawaguchi. "Deep learning without poor local minima". In: *Advances In Neural Information Processing Systems*. 2016, pp. 586–594.

[72] Michael Kearns and Satinder Singh. "Near-optimal reinforcement learning in polynomial time". In: *Machine Learning* 49.2-3 (2002), pp. 209–232.

[73] Sven Koenig and Reid G Simmons. "Complexity analysis of real-time reinforcement learning". In: *AAAI Conference on Artificial Intelligence*. 1993, pp. 99–105.

[74] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks". In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[75] Tor Lattimore and Marcus Hutter. "PAC bounds for discounted MDPs". In: *International Conference on Algorithmic Learning Theory*. 2012, pp. 320–334.

[76] Jason D Lee et al. "Gradient descent only converges to minimizers". In: *Conference on Learning Theory*. 2016, pp. 1246–1257.

[77] Yin Tat Lee and Aaron Sidford. "Efficient accelerated coordinate descent methods and faster algorithms for solving linear systems". In: *Foundations of Computer Science (FOCS)*. IEEE. 2013, pp. 147–156.

[78] Kfir Y Levy. "The Power of Normalization: Faster Evasion of Saddle Points". In: *arXiv preprint arXiv:1611.04831* (2016).

[79] Huan Li and Zhouchen Lin. "Provable Accelerated Gradient Method for Nonconvex Low Rank Optimization". In: *arXiv preprint arXiv:1702.04959* (2017).

[80] Qihang Lin et al. "Solving Weakly-Convex-Weakly-Concave Saddle-Point Problems as Weakly-Monotone Variational Inequality". In: *arXiv preprint arXiv:1810.10207* (2018).

[81] Aleksander Madry et al. "Towards deep learning models resistant to adversarial attacks". In: *arXiv preprint arXiv:1706.06083* (2017).

[82] Eric Mazumdar and Lillian J Ratliff. "On the Convergence of Gradient-Based Learning in Continuous Games". In: *arXiv preprint arXiv:1804.05464* (2018).

[83] Eric V Mazumdar, Michael I Jordan, and S Shankar Sastry. "On Finding Local Nash Equilibria (and Only Local Nash Equilibria) in Zero-Sum Games". In: *arXiv preprint arXiv:1901.00838* (2019).

[84] Song Mei et al. "Solving SDPs for synchronization and MaxCut problems via the Grothendieck inequality". In: *Conference on Learning Theory (COLT)*. 2017, pp. 1476–1515.

[85] Volodymyr Mnih et al. "Asynchronous methods for deep reinforcement learning". In: *International Conference on Machine Learning (ICML)*. 2016, pp. 1928–1937.

[86] Volodymyr Mnih et al. "Playing Atari with deep reinforcement learning". In: *arXiv preprint arXiv:1312.5602* (2013).

[87] Roger B Myerson. *Game theory*. Harvard university press, 2013.

[88] Anusha Nagabandi et al. "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning". In: *arXiv preprint arXiv:1708.02596* (2017).

[89] Vaishnavh Nagarajan and J Zico Kolter. "Gradient descent GAN optimization is locally stable". In: *Advances in Neural Information Processing Systems*. 2017, pp. 5585–5595.

[90] Ion Necoara, Yurii Nesterov, and Francois Glineur. "Linear convergence of first order methods for non-strongly convex optimization". In: *arXiv preprint arXiv:1504.06298* (2015).

[91] Yurii Nesterov. "A method of solving a convex programming problem with convergence rate O (1/k2)". In: *Soviet Mathematics Doklady* 27 (1983), pp. 372–376.

[92] Yurii Nesterov. "Efficiency of coordinate descent methods on huge-scale optimization problems". In: *SIAM Journal on Optimization* 22.2 (2012), pp. 341–362.

[93] Yurii Nesterov. *Introductory Lectures on Convex Optimization*. Vol. 87. Springer Science & Business Media, 2004.

[94] Yurii Nesterov. *Introductory Lectures on Convex Programming Volume I: Basic course*. Springer, 1998.

[95] Yurii Nesterov. "Squared functional systems and optimization problems". In: *High performance optimization*. Springer, 2000, pp. 405–440.

[96] Yurii Nesterov and Boris T Polyak. "Cubic regularization of Newton method and its global performance". In: *Mathematical Programming* 108.1 (2006), pp. 177–205.

[97] J v Neumann. "Zur theorie der gesellschaftsspiele". In: *Mathematische annalen* 100.1 (1928), pp. 295–320.

[98] Shayegan Omidshafiei et al. "Deep decentralized multi-task multi-agent reinforcement learning under partial observability". In: *arXiv preprint arXiv:1703.06182* (2017).

[99] Michael O'Neill and Stephen J Wright. "Behavior of Accelerated Gradient Methods Near Critical Points of Nonconvex Problems". In: *arXiv preprint arXiv:1706.07993* (2017).

[100] Alan V. Oppenheim and Ronald W. Schafer. *Discrete-time Signal Processing*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989. ISBN: 0-13-216292-X.

[101] Ian Osband and Benjamin Van Roy. "On Lower Bounds for Regret in Reinforcement Learning". In: *ArXiv e-prints* abs/1608.02732 (Apr. 2016).

[102] Ioannis Panageas and Georgios Piliouras. "Gradient descent only converges to minimizers: Non-isolated critical points and invariant regions". In: *arXiv preprint arXiv:1605.00405* (2016).

[103] Vitchyr Pong et al. "Temporal Difference Models: Model-Free Deep RL for Model-Based Control". In: *arXiv preprint arXiv:1802.09081* (2018).

[104] Hassan Rafique et al. "Non-convex min-max optimization: Provable algorithms and applications in machine learning". In: *arXiv preprint arXiv:1810.02060* (2018).

[105] Sashank J Reddi et al. "A generic approach for escaping saddle points". In: *arXiv preprint arXiv:1709.01434* (2017).

[106] Herbert Robbins and Sutton Monro. "A stochastic approximation method". In: *The annals of mathematical statistics* (1951), pp. 400–407.

[107] Gareth O Roberts, Richard L Tweedie, et al. "Exponential convergence of Langevin distributions and their discrete approximations". In: *Bernoulli* 2.4 (1996), pp. 341–363.

[108] Ralph Tyrell Rockafellar. *Convex analysis*. Princeton university press, 2015.

[109] Clément W Royer and Stephen J Wright. "Complexity analysis of second-order line-search algorithms for smooth nonconvex optimization". In: *arXiv preprint arXiv:1706.03131* (2017).

[110] John Schulman et al. "Trust region policy optimization". In: *International Conference on Machine Learning (ICML)*. 2015, pp. 1889–1897.

[111] Shai Shalev-Shwartz and Tong Zhang. "Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization". In: *International Conference on Machine Learning (ICML)*. 2014, pp. 64–72.

[112] Aaron Sidford et al. "Variance Reduced Value Iteration and Faster Algorithms for Solving Markov Decision Processes". In: *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM. 2018, pp. 770–787.

[113] David Silver et al. "Mastering the game of Go with deep neural networks and tree search". In: *Nature* 529.7587 (2016), pp. 484–489.

[114] Alexander L Strehl et al. "PAC model-free reinforcement learning". In: *Proceedings of the 23rd International Conference on Machine Learning*. ACM. 2006, pp. 881–888.

[115] Weijie Su, Stephen Boyd, and Emmanuel J Candes. "A differential equation for modeling Nesterov's accelerated gradient method: theory and insights". In: *Journal of Machine Learning Research* 17.153 (2016), pp. 1–43.

[116] Ju Sun, Qing Qu, and John Wright. "A geometric analysis of phase retrieval". In: *Information Theory (ISIT), 2016 IEEE International Symposium on*. IEEE. 2016, pp. 2379–2383.

[117] Ju Sun, Qing Qu, and John Wright. "Complete dictionary recovery over the sphere I: Overview and the geometric picture". In: *IEEE Transactions on Information Theory* (2016).

[118] Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. MIT press Cambridge, 1998.

[119] Nilesh Tripuraneni et al. "Stochastic cubic regularization for fast nonconvex optimization". In: *Advances in Neural Information Processing Systems*. 2018, pp. 2904–2913.

[120] Christopher Watkins. "Learning from delayed rewards". PhD thesis. King's College, Cambridge, 1989.

[121] A. Wibisono, Ashia C Wilson, and Michael I Jordan. "A variational perspective on accelerated methods in optimization". In: *Proceedings of the National Academy of Sciences* 133 (2016), E7351–E7358.

[122] Ashia C Wilson, Benjamin Recht, and Michael I Jordan. "A Lyapunov analysis of momentum methods in optimization". In: *arXiv preprint arXiv:1611.02635* (2016).

[123] Ashia C Wilson et al. "The marginal value of adaptive gradient methods in machine learning". In: *Advances in Neural Information Processing Systems*. 2017, pp. 4148–4158.

[124] Yi Xu, Jing Rong, and Tianbao Yang. "First-order stochastic algorithms for escaping from saddle points in almost linear time". In: *Advances in Neural Information Processing Systems*. 2018, pp. 5535–5545.

[125] Mishael Zedek. "Continuity and location of zeros of linear combinations of polynomials". In: *Proceedings of the American Mathematical Society* 16.1 (1965), pp. 78–84.

[126] Yuchen Zhang, Percy Liang, and Moses Charikar. "A hitting time analysis of stochastic gradient langevin dynamics". In: *arXiv preprint arXiv:1702.05575* (2017).

[127] Dongruo Zhou and Quanquan Gu. "Stochastic Recursive Variance-Reduced Cubic Regularization Methods". In: *arXiv preprint arXiv:1901.11518* (2019).

[128] Dongruo Zhou, Pan Xu, and Quanquan Gu. "Finding local minima via stochastic nested variance reduction". In: *arXiv preprint arXiv:1806.08782* (2018).