Understanding Minimax Optimization in Modern Machine Learning

by

Guojun Zhang

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
School of Computer Science

Waterloo, Ontario, Canada, 2021

Examining Committee Membership

The following served on the Examining Committee for this thesis. The decision of the Examining Committee is by majority vote.

External Examiner:	Simon Lacoste-Julien Associate Professor, Department of Computer Science and Operations Research (DIRO) Université de Montréal
Supervisor(s):	Pascal Poupart Professor, School of Computer Science University of Waterloo
	Yaoliang Yu Assistant Professor, School of Computer Science University of Waterloo
Internal Member:	Kimon Fountoulakis Assistant Professor, School of Computer Science University of Waterloo
	Gautam Kamath Assistant Professor, School of Computer Science University of Waterloo
Internal-External Member:	Stephen Vavasis Professor, Dept. of Combinatorics & Optimization University of Waterloo

Author's Declaration

This thesis consists of material all of which I authored or co-authored: see Statement of Contributions included in the thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Statement of Contributions

This thesis consists of the author's previous works during his PhD, including Zhang and Yu (2020), Zhang et al. (2020), Zhang et al. (2021) and Acuna et al. (2021). The first work is published at ICLR 2020; the second work is submitted to JMLR; the third work is accepted at the ICML 2021 workshop on "Beyond First Order Methods in Machine Learning." In the third work I am the first author but I also collaborated with Kaiwen Wu, who helped me implement the algorithms I proposed. The main theory and parts of the experiments are done by myself. Acuna et al. (2021) is published at ICML 2021 and my contribution is on the theoretical side.

Abstract

Recent years has seen a surge of interest in building learning machines through adversarial training. One type of adversarial training is through a discriminator or an auxiliary classifier, such as Generative Adversarial Networks (GANs). For example, in GANs, the discriminator aims to tell the difference between true and fake data. At the same time, the generator aims to generate some fake data that deceives the discriminator. Another type of adversarial training is with respect to the data. If the samples that we learn from are perturbed slightly, a learning machine should still be able to perform tasks such as classification relatively well, although for many state-of-the-art deep learning models this is not the case. People build robust learning machines in order to defend against the attacks on the input data.

In most cases, the formulation of adversarial training is through minimax optimization, or smooth games in a broader sense. In minimax optimization, we have a bi-variate objective function. The goal is to minimize the objective function with respect to one variable, and to maximize the objective function with respect to another. Historically, such a problem has been widely studied with convex-concave functions, where saddle points are a desirable concept. However, due to non-convexity, results with convex-concave functions would often not apply to adversarial training problems. It becomes important to understand the theory of non-convex minimax optimization in these models.

There are mainly two focuses within recent minimax optimization research. One is on the solution concepts: what is a desirable solution concept that is both meaningful in practice and easy to compute? Unfortunately, there is no definite answer for it, especially in GAN training. Besides, since non-convex minimax optimization includes non-convex minimization as a special case, there is no known efficient algorithm that can find global solutions. Therefore, local solution concepts, as surrogates, are necessary. Usually, people use local search methods such as gradient algorithms to find a good solution. So such concept must be at least stationary (critical) points. Based on the notion of stationarity, a solution concept called local minimax points is recently proposed. Local minimax points include local saddle points and they are stationary points at the same time. Moreover, they correspond to the well-known Gradient Descent Ascent (GDA) algorithm to some extent. I provide a comprehensive analysis of local minimax points, such as their relation with global solutions and other local solution concepts, their optimality conditions and the stability of gradient algorithms at local minimax points. My results show that although local minimax points are good surrogates of global solutions in e.g. quadratic functions, we may have to go beyond this minimax formulation since gradient algorithms may not be stable near local minimax points.

Another focus of recent research in the area of minimax optimization is on the algorithms. Including GDA, many old and new algorithms are proposed or analyzed for non-convex minimax optimization. Convergence rates and lower bounds of gradient algorithms are given, improved and compared. Compared to these noticeable contributions, my work focuses more on the stability side of these algorithms, as it is widely-known that gradient algorithms often exhibit some cyclic behaviour around a desirable solution in e.g. GAN training. I use the simplest bilinear case as an illustrative model for understanding the stability. I show that for a wide array of gradient algorithms, updating the two variables one-by-one is often more stable than updating them simultaneously. My stability analysis for bilinear functions can also be extended to general non-linear smooth functions, which allows us to distinguish hyper-parameter choices for more stable algorithms.

Finally, I propose new algorithms for minimax optimization. Most algorithms use gradient information for local search, with few exceptions that use the Hessian information as well to improve stability. I give a synthetic view of the convergence rates of current algorithms that use second-order information, and propose Newton-type methods for minimax optimization. My methods alleviate the problem of ill-conditioning in a local neighborhood, which is inevitable for gradient algorithms. This claim is proved by my theory and verified in my experiments.

Acknowledgements

I would like to first thank my two supervisors Pascal and Yaoliang for their long-lasting support and encouragement. Four years in short. I still remember the time when I first came to Pascal's office and said I wanted to do machine learning, although I basically had no knowledge on it. It is a hard transition from theoretical physics to machine learning, which I would not be able to make without Pascal's kindness and open-mindedness. Also, without Yaoliang's criticism I would not be able to realize the limitation of my knowledge. The often intensive discussion with him improves my eyesight to be more professional.

Four years is also long. From being a layman, to doing my first project, publishing my first paper, receiving feedback, giving presentations, going to conferences, and now, writing a thesis, I have experienced a lot. I want to thank my friends and colleagues, Jingjing, Zeou, Jayden, Kaiwen, Allen, Alix, Yetian, Priyank, Nabiha, Haonan, Victor, Sherry, Wei Zhou, Qi Hu, Yu Gu, Shanming and many others for their company and guidance.

Due to COVID-19, nearly everything is remote at the current time being. I would like to thank everyone who makes Internet communication and online working/studying possible. Many thanks to David, Marc and Sanja for the wonderful summer during my remote internship at NVIDIA.

I also want to thank my collaborators, Kiarash and George from Huawei Noah Ark's Lab, Guodong Zhang and Han Zhao. Specifically, Han and Guodong shared a lot of experience on how to become a successful researcher.

Finally, thanks to my parents and my sister for their support and selfless love during my PhD. Long distance would not separate our hearts and my thanks would never be enough.

Dedication

This thesis is dedicated to my parents and my sister.

Table of Contents

List of Figures

List of Tables

List of Notations

f	a bi-variate function for minimax optimization
x	the variable to minimize over
y	the variable to maximize over
$\mathcal{X}$	the domain of x
$\mathcal{Y}$	the domain of y
$y^*(x)$	a maximizer of the problem $\max_{y\in\mathcal{Y}} f(x,y)$
x^*	a minimizer of the problem $\min_{x\in\mathcal{X}} f(x,y^*(x))$
$(x_\star, y_\star)$	a global/local saddle point
(x^*, y^*)	a global/local minimax point
(x_*, y_*)	a global/local maximin point
$\bar{f}$	$\sup_{y\in\mathcal{Y}} f(\cdot, y)$
$\underline{f}$	$\inf_{x\in\mathcal{X}} f(x,\cdot)$
$\mathcal{N}(\cdot)$	the neighborhood of a point
$\mathcal{R}(\cdot)$	the range of a matrix
$\mathsf{null}(\cdot)$	the kernel of a matrix
∂f	the subdifferential of a function f
$\partial_x f$, $\partial_y f$	partial gradients of a function $f(x,y)$

$\partial^2_{xy} f$, $\partial^2_{yy} f$, $\partial^2_{xx} f$, $\partial^2_{yx} f$ partial Hessians of a function $f(x, y)$

$\mathtt{D}_x f$ $\partial_x f - \partial^2_{xy} f \cdot (\partial^2_{yy} f)^{-1} \partial_y f$

$\mathtt{D}^2_{xx} f$ $\partial_{xx} f - \partial^2_{xy} f \cdot (\partial^2_{yy} f)^{-1} \partial^2_{yx} f$

z concatenated vector (x, y)

$v(z)$ vector field $(-\alpha_1 \partial_x f(z), \alpha_2 \partial_y f(z))$

$H(z)$ the Jacobian of the vector field, $\nabla v(z)$

$\mathtt{F}$ the update rule for the follower y

$\mathtt{L}$ the update rule for the leader x

Chapter 1

Introduction

Deep neural networks have become the default model for extracting features from data, due to their power to approximate arbitrary functions. Deep models are versatile: people use them to classify images, to embed words, to take actions and to generate samples. In the recent decade, there has been successful frameworks that combine several deep neural networks to perform a task, including Generative Adversarial Networks (GANs) (Goodfellow et al., 2014) and Domain Adversarial Neural Networks (DANNs) (Ganin et al., 2016). Such frameworks are known as adversarial training models: there is an auxiliary adversarial network that rectifies the behavior of the main network of interest when it does not perform well.

Another recent trend of modern machine learning is regarding its robustness w.r.t. data. Since neural networks are often over-parametrized, there is a danger that they are over-fitting and only memorizing the dataset. If the samples are slightly perturbed, then the performance of a neural network could be severely degraded (Madry et al., 2018). This limits the application of deep models into the real world: imagine a self-driving car can distinguish pedestrians and stop signs during training, but if the performance decreases quickly at a slightly different scene not met before, then such a system could not be used in real applications. In order for a deep neural network to perform well with perturbation of data, there has been a series of research works on robust deep models (e.g. Goodfellow et al., 2015; Madry et al., 2018; Cohen et al., 2019).

Undoubtedly, optimization is the backbone of machine learning in terms of searching for good model parameters. In deep learning, model parameters are often vectors in Euclidean space, and thus continuous optimization is needed. Different from conventional optimization where the focus is mainly on minimizing a single convex function, there are two new

challenges brought by modern machine learning: the objective function is non-convex and the problem is minimax optimization. Specifically, for the applications I mentioned above, different variables are competing with each other: for the same objective function, we may want to maximize it w.r.t. one variable, and minimize it w.r.t. another. The non-convexity of deep models adds to the difficulty of understanding such optimization, which is still a popular research topic.

In this chapter, I will first introduce adversarial training models which are recently proposed, including learning machines that are robust to sample perturbation. Then I abstract away the exact formulation and define the general optimization problem. I will study this problem in later chapters.

1.1 Adversarial Training Models

In this section I introduce a few adversarial training models, including Generative Adversarial Networks and Domain Adversarial Training, and also an adversarial training procedure for achieving robustness against perturbations of the input. For the first two types of models, there is an auxiliary adversary that helps the main task by doing some discrimination or classification. For the adversarial training procedure, the adversary instead perturbs the samples so as to make sure the model performs well against the worst case perturbation.

1.1.1 Generative Adversarial Networks

Generative Adversarial Networks (GANs) have been a very popular model for generating images from Gaussian noise (Goodfellow et al., 2014). The basic design of a GAN architecture has a generator $G(\theta_g, \cdot)$ and a discriminator $D(\theta_d, \cdot)$, which are neural network functions, with parameters θ_g and θ_d. The generator takes a Gaussian noise as input and outputs a synthetic image. The discriminator takes an image and determines whether it is a real image or is generated from the generator. Through optimization, we want the generator to generate some images that are very close to the real images such that the discriminator cannot tell the difference. This is a minimax game, which can be formulated as:

$$\min_{\theta_g} \max_{\theta_d} V(\theta_g, \theta_d) := \mathbb{E}_{x \sim p_{\text{data}}}[\log D(\theta_d, x)] + \mathbb{E}_{z \sim p_z}[\log(1 - D(\theta_d, G(\theta_g, z)))], \tag{1.1}$$

where p_{data} is the real distribution and p_z is a latent distribution such that the push-forward $G\#p_z$ would approximate the distribution p_{data}. $D(\theta_d, \cdot)$, called the discriminator,

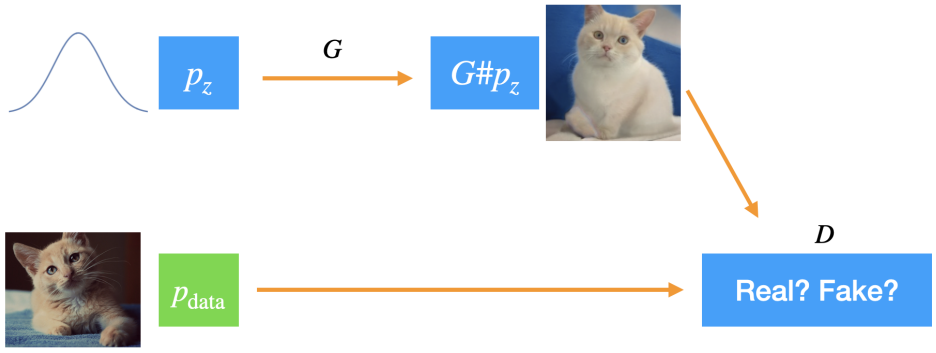


Figure 1.1: An illustration of GAN training. p_z is a latent distribution and the generator G takes a sample from p_z and generate some image. The discriminator tells whether the image is synthetic or real. The generated image sample is taken from StyleGAN (Karras et al., 2019).

is a function from an image to a probability between 0 and 1. According to this objective, for $x \sim p_{\text{data}}$, the discriminator would prefer $D(\theta_d, x) = 1$ and for $z \sim p_z$ the discriminator would prefer $D(\theta_d, G(\theta_g, z)) = 0$. Therefore, the role of the discriminator is to distinguish real data p_{data} from synthetic data $G\#p_z$. On the other hand, the generator tries to fool the discriminator such that the discriminator cannot tell the difference. Figure 1.1 gives an illustration of this learning scheme. Ideally, a successful generator would give $D(\theta_d, x) = D(\theta_d, G(\theta_g, z)) = 0.5$ for any $z \sim p_z$ and $x \sim p_{\text{data}}$.

If the discriminator is expressive enough, then the inner maximization problem has the following solution(s):

$$D(\theta_d^*(\theta_g), x) = \frac{p_{\text{data}}(x)}{p_g(x) + p_{\text{data}}(x)}, \tag{1.2}$$

where p_g is the distribution of $G(\theta_g, z)$ with $z \sim p_z$ and x is on the support $\text{supp}(p_{\text{data}}) \cup \text{supp}(p_g)$. We use $\theta_d^*(\theta_g)$ to denote that the optimal value θ_d^* depends on the choice of θ_g. With this notation, we have

$$V(\theta_d, \theta_g) \leq V(\theta_d^*(\theta_g), \theta_g). \tag{1.3}$$

Suppose the inner maximization problem is solved. We need to find parameter θ_g^* such that for any parameter θ_g, we have:

$$V(\theta_d^*(\theta_g), \theta_g) \geq V(\theta_d^*(\theta_g^*), \theta_g^*). \tag{1.4}$$

In fact, the minimization of $V(\theta_d^*(\theta_g), \theta_g)$ is equivalent to the minimization of the Jensen–Shannon (JS) divergence between p_g and p_{data}. It is possible to replace the JS divergence with other discrepancy terms between two distributions, such as the broader class of f-divergences (Nowozin et al., 2016), the Wasserstein distance (Arjovsky et al., 2017), the maximum mean discrepancy (Li et al., 2017) and the Sobolev integral probability metric (Mroueh et al., 2018). These generalizations follow the same pattern as (1.3) and (1.4), i.e., we first optimize a discriminator fixing the generator, and then minimize the maximum over the generator. This can be interpreted as the problem of finding a global minimax point, as I will demonstrate in Section 2.1.

1.1.2 Domain Adversarial Training

Similar to GANs, domain adversarial training of neural networks (DANN, Ganin et al., 2016) is another type of adversarial training models that aims to solve the problem of unsupervised domain adaptation. In this problem, there is a source domain (i.e. a source distribution), for which we know the labels, and a target domain (i.e. a target distribution), where we only have unlabeled samples. For example, we could have labeled synthetic images generated from computer graphics as a source domain, and unlabeled real images taken from the real world as a target domain. The goal of unsupervised domain adaptation is to find a feature embedding $G(\theta_g, \cdot)$, usually represented with a neural network, such that the push-forwards of the source and target distributions are similar, i.e.,

$$G\#p_S|_x \approx G\#p_T|_x, \tag{1.5}$$

where p_S and p_T denote the (joint) source and target distributions, and $p_S|_x$ and $p_T|_x$ denote the marginal source and target distributions on the input. $G\#p_S|_x$ and $G\#p_T|_x$ denote the push-forward distributions of $p_S|_x$ and $p_T|_x$ separately. Based on this feature embedding, any classifier that performs well on the source domain should perform well on the target domain.

Denote $\mathcal{X}$ as the input space and $\mathcal{Z}$ as the feature space (a subset of a Euclidean space). The framework of DANN has three parts: a feature extractor $G : \mathcal{X} \to \mathcal{Z}$ as we mentioned before; a classifier C that takes a hidden vector in $\mathcal{Z}$ and outputs a label; a discriminator D which tells whether a hidden vector in $\mathcal{Z}$ is from the source domain or the target domain. Through optimization, we want the feature extractor to satisfy (1.5), and also the classifier C to be able to predict the label from the input, no matter whether it is from the source domain or the target domain. In such a case, the discriminator D would not be able to tell whether a feature is from the source domain or the target domain. This is a minimax

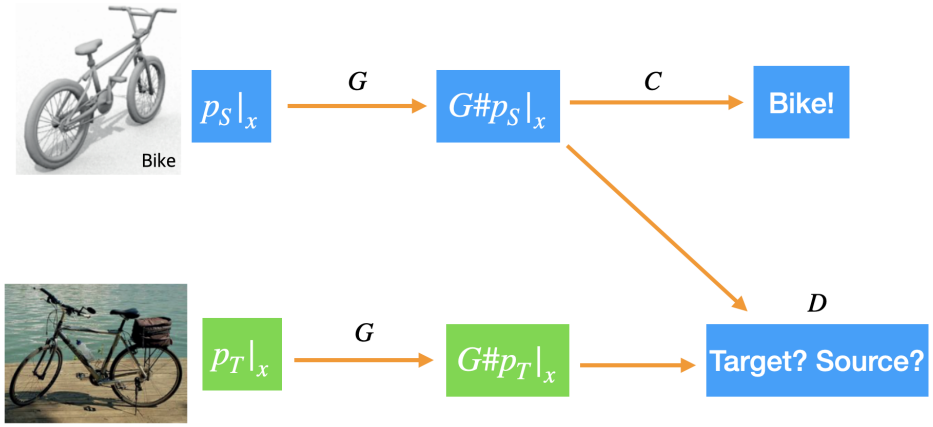


Figure 1.2: An illustration of the DANN framework. The feature embedding G encodes samples from the source and target domains in such a way that the discriminator D would not be able to tell the difference. Therefore, a good source classifier C based on the feature embedding G can also be applied to the target domain. Images taken from Bermúdez-Chacón et al. (2019).

game, which can be formulated as:

$$\begin{aligned}\min_{\theta_g,\theta_c}\max_{\theta_d} V(\theta_g,\theta_c,\theta_d) &:= \mathbb{E}_{(x,y)\sim p_S}[\ell(y, C(\theta_c, G(\theta_g, x)))]+ \\ &+\lambda\left(\mathbb{E}_{x\sim p_S|_x}[\log(D(\theta_d, G(\theta_g,x)))]+\mathbb{E}_{x\sim p_T|_x}[\log(1-D(\theta_d,G(\theta_g,x)))]\right),\end{aligned} \tag{1.6}$$

where $\ell(y,\hat{y})$ is a loss function that tells how good the prediction $\hat{y}$ is compared to the ground truth y, and we use θ_d, θ_c, θ_g to denote the parameters of the neural network functions D, C and G.

We note that the second line of (1.6) resembles the GAN formulation in (1.1). If the discriminator is expressive enough, then the inner maximization problem has the following solution(s):

$$D(\theta_d^*(\theta_g), z) = \frac{(G\#p_S|_x)(z)}{(G\#p_S|_x)(z)+(G\#p_T|_x)(z)}, \tag{1.7}$$

where $G\#p_S|_x$ and $G\#p_T|_x$ are the distributions of $G(\theta_g, x)$ with $x\sim p_S|_x$ and $x\sim p_T|_x$ respectively, and z is on the support $\mathrm{supp}(G\#p_S|_x)\cup\mathrm{supp}(G\#p_T|_x)$. The optimal value θ_d^* depends on the choice of θ_g, and we have:

$$V(\theta_d,\theta_g,\theta_c)\le V(\theta_d^*(\theta_g),\theta_g,\theta_c). \tag{1.8}$$

Suppose the inner maximization problem is solved. We need to find parameters θ_g^* and θ_c^* such that

$$V(\theta_d^*(\theta_g), \theta_g, \theta_c) \geq V(\theta_d^*(\theta_g^*), \theta_g^*, \theta_c^*), \tag{1.9}$$

for any parameters θ_g and θ_c. The minimization of $V(\theta_d^*(\theta_g), \theta_g, \theta_c)$ is equivalent to the following problem:

$$\min_{\theta_g, \theta_c} \mathbb{E}_{(x,y)\sim p_S}[\ell(y, C(\theta_c, G(\theta_g, x)))] + \lambda \mathcal{D}_{\mathrm{JS}}(G\#p_S|_x \| G\#p_T|_x), \tag{1.10}$$

where $\mathcal{D}_{\mathrm{JS}}$ is the Jensen–Shannon (JS) divergence between two distributions. Namely, we are minimizing both the classification loss on the source domain, and the JS divergence between the feature embedding of the source domain and the target domain. This has close relation with the domain adaptation theory by Ben-David et al. (2010).

There are many follow-up works after Ganin et al. (2016) on unsupervised domain adaptation, using adversarial training, such as Shu et al. (2018), Long et al. (2018), Hoffman et al. (2018), Zhang et al. (2019), Acuna et al. (2021), and Shen et al. (2018). Specifically, Acuna et al. (2021) consider generalizing the use of JS divergence to the more general f-divergences, and Shen et al. (2018) consider replacing the JS divergence with Wasserstein distance. In these works, the parameters are solved through a minimax game, similar to (1.6), (1.8) and (1.9).

1.1.3 Adversarial Robustness

Deep neural networks have been more and more widely used in many areas of machine learning, such as computer vision and natural language processing. They are powerful models for feature embedding and classification. At the same time, they are fragile. It has been observed that for many trained neural network classifiers (Szegedy et al., 2014), small perturbations of the samples would decrease the performance significantly (see Figure 1.3), which is undesirable for real-life tasks such as autonomous driving. Efficient attack methods such as Fast Gradient Sign Method (FGSM) (Goodfellow et al., 2015) and multi-step projected gradient descent (Madry et al., 2018), and the corresponding defense methods for robust neural networks have been designed. Specifically, Cohen et al. (2019) give certified robustness through the method of randomized smoothing.

The attack and defense w.r.t. the sample perturbation can be formulated as a minimax game, similar to (1.1) and (1.6). In such a game, the defender aims to find a robust neural network that can classify samples for any small perturbations, and the attacker aims to find

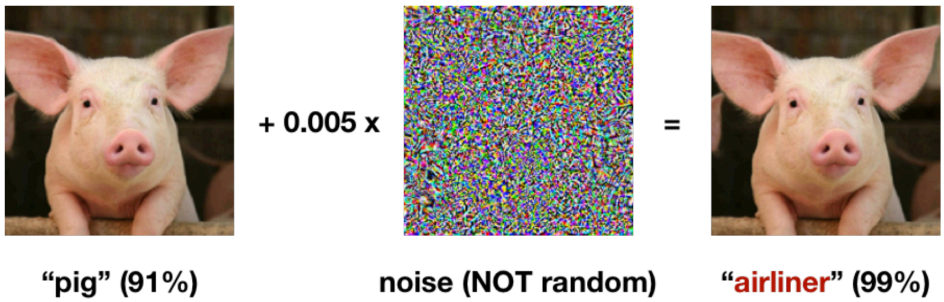


Figure 1.3: Deep neural networks are fragile under small adversarial perturbations in the sense that the prediction label changes even though the image barely changes from human eyes. Image taken from Madry (2019).

the worst small perturbations given a neural network. Mathematically, it can be written as:

$$\min_{\theta} \mathbb{E}_{(x,y)\sim p_{\text{data}}}[\max_{\delta\in\mathcal{S}} \ell(y, C(\theta, x+\delta))], \tag{1.11}$$

where p_{data} is the sample distribution, $\ell(y, \hat{y})$ is a loss function that tells how good the prediction $\hat{y}$ is compared to the ground truth y, $C(\theta, x)$ is a classification neural network with parameter θ and sample x, and $\mathcal{S}$ is the set of allowed perturbations. Usually, $\mathcal{S}$ can be ℓ_2 or ℓ_∞ norm balls.

The task of training a robust classifier can be treated as solving (1.11) in the following way. For a given parameter θ, we find the worst perturbation (possibly set-valued) function $\delta^*(\theta, \cdot) : \text{supp}(p_{\text{data}}) \to \mathcal{S}$, such that:

$$\mathbb{E}_{(x,y)\sim p_{\text{data}}}[\max_{\delta\in\mathcal{S}} \ell(y, C(\theta, x+\delta))] \leq \mathbb{E}_{(x,y)\sim p_{\text{data}}}[\ell(y, C(\theta, x+\delta^*(\theta, x, y)))], \tag{1.12}$$

this is equivalent to saying that given θ, for any $(x, y) \in \text{supp}(p_{\text{data}})$, we have:

$$\max_{\delta\in\mathcal{S}} \ell(y, C(\theta, x+\delta)) \leq \ell(y, C(\theta, x+\delta^*(\theta, x, y))). \tag{1.13}$$

After the worst-case perturbation function is found, we want to find an optimal classifier θ^* such that for any θ, the following holds:

$$\mathbb{E}_{(x,y)\sim p_{\text{data}}}[\ell(y, C(\theta, x+\delta^*(\theta, x, y)))] \geq \mathbb{E}_{(x,y)\sim p_{\text{data}}}[\ell(y, C(\theta^*, x+\delta^*(\theta^*, x, y)))]. \tag{1.14}$$

What we have mentioned in the problem above is making a perturbation for each sample. It is also possible to treat the samples as a distribution and study the distribution shift. This is called distributional robust optimization (DRO). Sinha et al. (2018) proposed measuring the perturbation with Wasserstein distance, and the formulation can be written as:

$$\min_{\theta} \max_{p} \mathsf{DRO}(\theta, p) := \mathbb{E}_{(x,y)\sim p}[\ell(y, C(\theta, x))] - \gamma W(p, p_{\text{data}}), \tag{1.15}$$

where W is the Wasserstein distance (see e.g. Sinha et al. (2018)), $C(\theta, \cdot)$ is the model classifier given an input, and $\rho > 0$, $\gamma > 0$ are hyperparameters. In such a formulation, we first find an adversarially perturbed distribution and then find the parameter θ for (certified) robustness. Namely, we find the adversarial distribution $p^*(\theta)$ such that for any distribution p, we have:

$$\mathsf{DRO}(\theta, p) \leq \mathsf{DRO}(\theta, p^*(\theta)), \tag{1.16}$$

and then we find a robust model parameter θ^* such that:

$$\mathsf{DRO}(\theta, p^*(\theta)) \geq \mathsf{DRO}(\theta^*, p^*(\theta^*)). \tag{1.17}$$

1.2 Minimax Optimization

From the examples above, one can abstract away the exact task and focus on the optimization problem. In general, we have a smooth function $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ and the minimax game is written as:

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x, y). \tag{1.18}$$

The domains of variables $\mathcal{X}$ and $\mathcal{Y}$ can be polymorphic. For example, for GANs and DANNs, x and y are parameters of neural networks, and thus $\mathcal{X}$ and $\mathcal{Y}$ are Euclidean spaces. For adversarial robustness in (1.11), $\mathcal{Y}$ is the set of functions whose values are bounded.

While the problem (1.18) has been studied a long while ago for convex-concave functions (Nemirovsky and Yudin, 1983), recent tasks impose new challenges for the non-convex settings. Since neural network functions are non-convex, knowledge of optimization for convex problems cannot be applied on the more general problem (1.18).

There are mainly two questions regarding the minimax game (1.18). One is:

What is the solution we are trying to find?

From our examples in Section 1.1, the problem (1.18) is solved as bi-level optimization. Given x, one finds the optimal variable $y^*(x)$ such that:

$$f(x, y) \leq f(x, y^*(x)), \forall y \in \mathcal{Y}, \tag{1.19}$$

and then $f(x, y^*(x))$ is minimized so that the minimizer x^* can be found:

$$f(x, y^*(x)) \geq f(x^*, y^*(x^*)), \forall x \in \mathcal{X}. \tag{1.20}$$

As we will see in Section 2.1, the solution $(x^*, y^*(x^*))$ is known as the global minimax point. Another important question is:

What is a good algorithm for finding the solution?

If we constrain the definition of "a solution" to be global minimax points, there is no efficient algorithm for it in general for nonconvex-nonconcave functions. This is because even finding the optimal $y^*(x)$ given x is a non-convex maximization problem and thus NP-hard (Murty and Kabadi, 1987). Therefore, one has to look for other surrogates of global minimax points, and associated algorithms have to be designed and analyzed. I will discuss these problems in following chapters.

1.2.1 General-Sum Games

We can also regard (1.18) as a zero-sum game: we can consider $f(x, y)$ as a utility function of y and $-f(x, y)$ as a utility function of x. Each player aims to maximize its own utility function, and the sum of utility functions is always zero. It is possible to extend zero-sum games to general-sum games. Suppose the utility functions of x and y are $-f_1(x, y)$ and $f_2(x, y)$, the general-sum game can be written as:

$$\min_{x \in \mathcal{X}} f_1(x, y^*(x)),\ y^*(x) \in \operatorname*{argmax}_{y \in \mathcal{Y}} f_2(x, y). \tag{1.21}$$

This is also known as bi-level optimization (Anandalingam and Friesz, 1992). In such a case, f_1 and f_2 do not have to be equal. This is a more general formulation than the zero-sum minimax game, and thus finding (or even defining) a good solution is more difficult. There are real applications where the problem can be formulated as a general-sum game but not a zero-sum game. For example, in neural architecture search (Liu et al., 2018) and

hyperparameter optimization (Maclaurin et al., 2015), the following problem needs to be solved:

$$\min_{\alpha} \mathcal{L}_{\text{val}}(\theta^*(\alpha), \alpha),\ \theta^*(\alpha) \in \operatorname*{argmin}_{\theta} \mathcal{L}_{\text{train}}(\theta, \alpha), \tag{1.22}$$

where α denotes the hyperparameters and θ denotes the model parameters. $\mathcal{L}_{\text{train}}$ and $\mathcal{L}_{\text{val}}$ are the training loss and the validation loss. We first train a model given some hyperparameters, and then tune the hyperparameters automatically based on the validation loss.

1.2.2 Roadmap

In the following chapters, I will focus on minimax optimization although parts of the results can be extended to general-sum games as well. Chapter 2 talks about global and local solution concepts in nonconvex minimax optimization; Chapter 3 is about the stability of gradient algorithms; Chapter 4 describes second-order methods for minimax optimization.

Chapter 2

Solution Concepts

In the first chapter we have seen several problems in machine learning where (nonconvex) minimax optimization is relevant. In these problems, we have a bi-variate function $f(x, y)$, and the optimal solution is defined in the following way: first, we find $y^*(x)$ such that:

$$y^*(x) \in \operatorname*{argmax}_{y \in \mathcal{Y}} f(x, y),\ x^* \in \operatorname*{argmin}_{x \in \mathcal{X}} f(x, y^*(x)). \tag{2.1}$$

Such a solution is called a global minimax point, as we will see in Section 2.1.

In this chapter I further explore global minimax points and their relation with the more widely-known global saddle points in Section 2.1. Since global minimax points are difficult to find, I study local solution concepts in Section 2.2 as surrogates of the global solutions. I explore optimality conditions of local solutions in Section 2.3. Since we used local optimal solutions as surrogates, it is important to understand the relation between local and global solutions, which I study in Section 2.4 for quadratic games. I find that in quadratic games local and global solutions are in some sense equivalent, but this is not true in general non-convex-non-concave cases.

We assume $\mathcal{X} \subseteq \mathbb{R}^n$ and $\mathcal{Y} \subseteq \mathbb{R}^m$ are subsets of Euclidean spaces.

2.1 Global Solution Concepts

In this section we study two global solution concepts for minimax optimization: global saddle points and global minimax points. In game theory, they are also called Nash equilibria and Stackelberg equilibria. Saddle points are widely studied when the function f

is convex-concave. However, in general non-convex-non-concave (NCNC) cases, they may not even exist. Global minimax points are a broader solution concept. They include global saddle points and are better suited for NCNC minimax optimization.

2.1.1 Global Saddle Point

In the convex setting, the following solution concept is well-known:

Definition 2.1.1 (**global saddle point**). *We call* $(x_\star, y_\star)$ *global saddle if for all* $x \in \mathcal{X}$ *and* $y \in \mathcal{Y}$*:*

$$f(x_\star, y) \leq f(x_\star, y_\star) \leq f(x, y_\star). \tag{2.2}$$

In other words, we simultaneously *have:*

$$x_\star \in \operatorname*{argmin}_{x \in \mathcal{X}} \; f(x, y_\star), \quad y_\star \in \operatorname*{argmax}_{y \in \mathcal{Y}} \; f(x_\star, y). \tag{2.3}$$

Global saddle points correspond to *Nash equilibria* (Nash, 1950), where each player has no incentive to deviate from his/her current strategy even after knowing the opponent's strategy exactly.

2.1.2 Global Minimax Point

Definition 2.1.2 (**global envelope function**). *Global envelope functions are defined as:*

$$\bar{f}(x) := \sup_{y \in \mathcal{Y}} \; f(x, y), \; \underline{f}(y) := \inf_{x \in \mathcal{X}} \; f(x, y). \tag{2.4}$$

For envelope functions, we allow $\bar{f}$ to take value $+\infty$ and $\underline{f}$ to take value $-\infty$. Definition 2.1.2 occurs if one player is doing robust optimization. For example, x could minimize the worst-case payoff, i.e., $\bar{f}(x)$, which is a nonconvex, non-smooth function (even when f is itself smooth):

$$\min_{x \in \mathcal{X}} \; \bar{f}(x). \tag{2.5}$$

On the other hand, player y simply maximizes $f(x, \cdot)$ given any $x \in \mathcal{X}$. This leads immediately to the following solution concept:

Definition 2.1.3 (**global minimax and maximin**). $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ *is global minimax if*

$$x^* \in \operatorname*{argmin}_{x \in \mathcal{X}} \bar{f}(x),\ y^* \in \operatorname*{argmax}_{y \in \mathcal{Y}}\ f(x^*, y). \tag{2.6}$$

In other words, for all $x \in \mathcal{X}$ *and* $y \in \mathcal{Y}$*:*

$$f(x^*, y) \le f(x^*, y^*) = \bar{f}(x^*) \le \bar{f}(x). \tag{2.7}$$

Similarly, we call $(x_*, y_*) \in \mathcal{X} \times \mathcal{Y}$ *global maximin if*

$$y_* \in \operatorname*{argmax}_{y \in \mathcal{Y}} \underline{f}(y),\ \ x_* \in \operatorname*{argmin}_{x \in \mathcal{X}}\ f(x, y_*). \tag{2.8}$$

In other words, for all $y \in \mathcal{Y}$ *and* $x \in \mathcal{X}$*:*

$$\underline{f}(y) \le \underline{f}(y_*) = f(x_*, y_*) \le f(x, y_*). \tag{2.9}$$

The concept of global minimax points is used widely in adversarial training, as we have seen in Section 1.1.

Remark 2.1.4 (**difficulty of finding global minimax points**). *Although the notion of global minimax is well-defined, it suffers from some major issues once we enter the NCNC world:*

- *We are not aware of an efficient algorithm (Murty and Kabadi, 1987) for finding a global minimizer* x^* *for the nonconvex function* $\bar{f}$*. This can be mitigated by contending with a local minimizer or even stationary point.*
- *Given* x^**, it is NP-hard to find a global maximizer* y^* *for the non-concave function* $f(x^*, y)$*. While it is tempting to relax again to a local solution, this will unfortunately affect our notion of optimality for* x^* *in the first place. We will return to this issue in the next section.*
- *The envelope function* $\bar{f}$ *is not smooth even when* f *is. Although we can turn to non-smooth optimization techniques, it will be inevitably slow to optimize* $\bar{f}$*.*

If we define the "mirror" function $\curlywedge(y, x) = f(x, y)$, then (x_*, y_*) is global maximin for f iff (y_*, x_*) is global minimax for $-\curlywedge$. For this reason, we will limit our discussion

mainly to minimax. Such a definition arises in the optimization literature as well since Definition 2.1.3 can be treated as a global solution to the minimax optimization problem:

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

We note that the ordering of x and y, i.e. which player moves first, matters: for instance, to get a global minimax pair (x^*, y^*), we must first find x^* and then conditioned on x^* we find the "certificate" y^*. In game-theoretic terms, this is also known as a Stackelberg game (von Stackelberg, 1934), where x is the leader while y is the follower.

It is well-known that weak duality, namely the inequality

$$\max_{y\in\mathcal{Y}} \underline{f}(y) \leq \min_{x\in\mathcal{X}} \bar{f}(x) \tag{2.10}$$

always holds. *Strong duality*, namely when equality is attained in (2.10), holds only under stringent conditions. The following theorem easily follows from the definitions:

Theorem 2.1.5 (e.g. Facchinei and Pang 2007, Theorem 1.4.1)**.** *For* any *function f, the pair $(x_\star, y_\star) \in \mathcal{X}\times\mathcal{Y}$ is global saddle iff it is both global minimax and global maximin iff strong duality holds and*

$$x_\star \in \operatorname*{argmin}_{x\in\mathcal{X}} \bar{f}(x),\ \ y_\star \in \operatorname*{argmax}_{y\in\mathcal{Y}} \underline{f}(y). \tag{2.11}$$

Let us give some examples to digest the definitions. In general, it is possible to find a game where both global maximin and minimax points exist, but there is no saddle point:

Example 2.1.6 (**both global minimax and maximin points exist; no saddle point**)**.** *Consider the bivariate function*

$$f(x,y) = x^4/4 - x^2/2 + xy \tag{2.12}$$

defined on $\mathbb{R}\times\mathbb{R}$. Global minimax points are clearly $\{0\}\times\mathbb{R}$ with value 0. On the other hand, global maximin points are $(\pm 1, 0)$ with value $-1/4$. Indeed,

$$\max_y \min_x\ x^4/4 - x^2/2 + xy \leq \max_y \min_x\ x^4/4 - x^2/2 \leq -\tfrac{1}{4}, \tag{2.13}$$

with equality attained at $(\pm 1, 0)$. Note that we have $xy \leq 0$ in the first inequality since we can always take $x \to -x$ to decrease the objective if $xy > 0$. The failure of strong duality proves the non-existence of saddle points (Theorem 2.1.5).

Note that given a global saddle pair $(x_\star, y_\star)$, $y_\star \in \mathcal{Y}_\star := \operatorname{argmax}_{y\in\mathcal{Y}} f(x_\star, y)$ but not every certificate $\bar{y} \in \mathcal{Y}_\star$ forms a global saddle pair with $x_\star$. This is known as "instability," which is the reason underlying the non-convergence of the gradient descent ascent (GDA) algorithm (Golshtein, 1972; Nemirovsky and Yudin, 1983).

Example 2.1.7 (**instability of GDA**). *Consider the bilinear (hence convex-concave)*

$$f(x, y) = xy$$

defined on $\mathbb{R} \times \mathbb{R}$. *It is easy to verify that global minimax points are precisely the set* $\{0\} \times \mathbb{R}$ *while global maximin points are* $\mathbb{R} \times \{0\}$. *Taking the intersection we have the unique global saddle point* $(0, 0)$. *This bilinear function is unstable, since given* $x^* = 0$, *not every global minimax certificate (namely the entire* $\mathbb{R}$*) forms a global saddle point with* x^*. *The last iterates of GDA do not converge to the unique global saddle point for this function with any (constant or not) step size, provided that it is not initialized at the saddle point (Nemirovsky and Yudin, 1983, p. 211).*

Another interesting example consists of quadratic games, which we completely classify in Section 2.4. Below we give a one-dimensional example where there is no global maximin or saddle point, but global minimax points exist.

Example 2.1.8 (**global minimax points exist; no global maximin or saddle points**). *Let* $f(x, y) = ax^2 + by^2 + cxy$ *with* $a < 0, b < 0$ *and* $c^2 \geq ab$. *According to the characterization in Theorem 2.4.1,* f *only admits global minimax points. Note that for quadratic games, the existence of both global minimax and maximin points implies the existence of a saddle point, in sharp contrast with Example 2.1.6.*

From the example above, we see that even for simple quadratic games, saddle points may not exist. In fact, unconstrained quadratic games are often given as typical examples for NCNC minimax optimization (Jin et al., 2020; Daskalakis and Panageas, 2018; Ibrahim et al., 2020; Wang et al., 2020). Locally, they can also be regarded as second-order approximations of any smooth function, and thus seem to be good representatives of NCNC games. However, we will show in Section 2.4 that they are special in many aspects.

2.2 Local Solution Concepts

In the last section I studied global solution concepts. The biggest problem is that we do not know efficient algorithms for finding them. Therefore, we have to resort to local solution

concepts. The main concepts I present are local versions of saddle points and minimax points in Section 2.1. Our results extend Jin et al. (2020). Specifically, we show that local saddle points are a special subclass of local minimax points called uniformly local minimax points. I will also discuss the relation between local and global minimax points.

Let us study definitions of local optimal points based on envelope functions and infinitesimal robustness (in the same spirit as Hampel (1974)). Compared to global optimal points, for local versions, we assume that we only have access to local information of f, i.e., given a point (x, y), we only know f over a neighborhood $\mathcal{N}(x) \times \mathcal{N}(y)$. Therefore, each player can only evaluate its current strategy by comparing with other strategies in the current neighborhood, corresponding to the notion of a local minimum (maximum). This can be achieved with the following local envelope functions. In the definition below, we denote

$$\mathcal{N}(y^*, \epsilon) := \{y \in \mathcal{Y} : \|y - y^*\| \leq \epsilon\}, \tag{2.14}$$

as the intersection of $\mathcal{Y}$ with a ball of radius ϵ surrounding y^* in $\mathbb{R}^m$, and similarly for $\mathcal{N}(x^*, \varepsilon)$. The exact form of the ball depends on the norm we choose.

Definition 2.2.1 (**local envelope function**). *Fix a reference point $y^* \in \mathcal{Y}$ and radius $\epsilon \geq 0$, we localize the envelope function:*

$$\bar{f}_\epsilon(x) = \bar{f}_{\epsilon,y^*}(x) := \max_{y \in \mathcal{N}(y^*,\epsilon)} f(x, y). \tag{2.15}$$

The definition for $\underline{f}_\epsilon(y) = \underline{f}_{\epsilon,x^*}(y)$ is similar if we fix some $x^* \in \mathcal{X}$.

2.2.1 Stationary Point

Perhaps the easiest way to define a local optimal solution for a differentiable function is to say that it is a fixed point of gradient algorithms. In other words, (x^*, y^*) in the interior of $\mathcal{X} \times \mathcal{Y}$ is locally optimal if:

$$\partial_x f(x^*, y^*) = \partial_y f(x^*, y^*) = \mathbf{0}. \tag{2.16}$$

This shares similarity with the stationary point x^* of a differentiable uni-variate function $h(x)$, which satisfies:

$$\nabla h(x^*) = \mathbf{0}. \tag{2.17}$$

For the constrained minimization problem $\min_{x\in\mathcal{X}} h(x)$, a stationary point (Bertsekas, 1997) $x^* \in \mathcal{X}$ is defined such that:

$$\nabla h(x^*)^\top (x - x^*) \geq 0,\ \forall x \in \mathcal{X}. \tag{2.18}$$

This condition is equivalent to being a global minimum in the case when $\mathcal{X}$ and h are convex. Similarly, we can define the stationary point for the minimax problem (1.18):

$$\partial_x f(x^*, y^*)^\top (x - x^*) \geq 0 \geq \partial_y f(x^*, y^*)^\top (y - y^*),\ \forall x \in \mathcal{X},\ y \in \mathcal{Y}. \tag{2.19}$$

The only difference with (2.18) is that the problem is a minimax game, and we are also maximizing over y. If f is convex-concave, and both $\mathcal{X}$ and $\mathcal{Y}$ are convex, then a stationary point is equivalent to a global minimax/global saddle point.

2.2.2 Local Saddle Point

In the NCNC setting, it is natural to consider local versions of saddle points (c.f. Definition 2.1.1) by localizing around neighborhoods $\mathcal{N}(x_\star, \epsilon)$ and $\mathcal{N}(y_\star, \epsilon)$. Below, when we mention the local envelope functions $\bar{f}_\epsilon(x)$ and $\underline{f}_\varepsilon(y)$ (see Definition 2.2.1) the centers and the neighborhoods are often omitted since they are clear from the context.

Definition 2.2.2 (**local saddle**). *We call the pair $(x_\star, y_\star) \in \mathcal{X} \times \mathcal{Y}$ local saddle if there exists $\epsilon > 0$, such that for all $x \in \mathcal{N}(x_\star, \epsilon)$ and $y \in \mathcal{N}(y_\star, \epsilon)$, $f(x_\star, y) \leq f(x_\star, y_\star) \leq f(x, y_\star)$. In other words,*

- *Fixing $x_\star$ then $y_\star$ is a local maximizer of $\underline{f}_{0,x_\star}(y) = f(x_\star, y)$;*
- *Fixing $y_\star$ then $x_\star$ is a local minimizer of $\bar{f}_{0,y_\star}(x) = f(x, y_\star)$.*

In the above definition, each player contends with the local optimality of its strategy by comparing with other strategies in a neighborhood. For local saddle points, we can WLOG take the norm $\|\cdot\|$ in the neighborhood definition (see (2.14)) to be Euclidean.

2.2.3 Local Minimax Point

We can now generalize the definition above. One player may not be aware of the exact strategy of the opponent, and thus doing robust optimization, given a certain range of the opponent's strategy. If x is doing (a sequence of) local robust optimization and y is doing usual optimization given the strategy of x, we have the following definition:

Definition 2.2.3 (**local minimax**). *We call* $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ *a local minimax point if*

- *Fixing* x^* *then* y^* *is a local maximizer of* $\underline{f}_{0,x^*}(y) = f(x^*, y)$*;*
- *Fixing* y^* *then* x^* *is a local minimizer of* $\bar{f}_{\epsilon_n, y^*}(x)$ *for all* ϵ_n *in some sequence* $0 < \epsilon_n \to 0$*.*

Furthermore, if the neighborhood over which x^* *is a local minimizer of* $\bar{f}_{\epsilon_n}$ *can be chosen to be independent of* ϵ_n*, then we call* (x^*, y^*) uniformly local minimax*.*

In the definition above, we defined uniformly local minimax points. By uniformity we meant that the neighborhood $\mathcal{N}$ does not depend on the element in the sequence. We will show a close relation between local saddle points and uniformly local minimax points in Proposition 2.2.7.

Definition 2.2.3 reveals the asymmetric position between the two players x and y: y needs only be a local certificate to testify the local optimality of x, but x only has an inexact estimate of y and thus minimizes the envelope function $\bar{f}_\epsilon(x)$ as the worst-case payoff. By switching the role of x and y we obtain a similar notion of local maximin.

In Proposition 2.2.6 we will see that Definition 2.2.3 has a seemingly stronger but equivalent form. To help digesting the somewhat complicated definition, we mention the following interpretation (e.g. Wang et al., 2020):

Theorem 2.2.4 (**sufficient and necessary condition of local minimax when** $\partial^2_{yy} f$ **is invertible**). *Let* $\mathcal{X} = \mathbb{R}^n, \mathcal{Y} = \mathbb{R}^m$ *and* $f : \mathbb{R}^n \to \mathbb{R}^m$ *be twice continuously differentiable. Suppose* $\partial^2_{yy} f(x^*, y^*)$ *is invertible, then* (x^*, y^*) *is local minimax iff*

- $\partial_y f(x^*, y^*) = \mathbf{0}$*,* $\partial^2_{yy} f(x^*, y^*) \prec \mathbf{0}$*, and*
- x^* *is a local minimizer of the total function* $f(x, y(x))$ *where the domain of* y *is an open set that contains* x^* *through the nonlinear equation*

$$\partial_y f(x, y) = \mathbf{0}. \tag{2.20}$$

Proof. Given that $\partial^2_{yy} f(x^*, y^*)$ is invertible, the first condition is clearly equivalent to y^* being a local maximizer of $f(x^*, \cdot)$. Consider the nonlinear equation (2.20), whose solution is determined by the implicit function theorem as a continuously differentiable function $y(x)$ defined near x^*. Fix any ϵ. Since $y(x^*) = y^*$, shrinking the neighbourhood around x^*

if necessary we may assume $y(x) \in \mathcal{N}(y^*, \epsilon)$ so that $\bar{f}_\epsilon(x) = f(x, y(x))$. Thus, if (x^*, y^*) is local minimax, then for x near x^*:

$$f(x^*, y(x^*)) = f(x^*, y^*) = \bar{f}_\epsilon(x^*) \leq \bar{f}_\epsilon(x) = f(x, y(x)), \tag{2.21}$$

so, x^* is a local minimizer of the total function. Reversing the argument proves the converse. □

We emphasize that, unlike the definition in Jin et al. (2020), we do not allow ϵ_n to take 0 in Definition 2.2.3 for two reasons: (a) This allows us to better separate local saddle from local minimax; (b) It is unnecessary to have $\epsilon_n = 0$, which we will see in Proposition 2.2.9.

We now show how to simplify Definition 2.2.3, starting with the following key lemma:

Lemma 2.2.5. *Suppose y^* maximizes $f(x^*, y)$ over some neighborhood $\mathcal{N}(y^*, \epsilon_0)$. If x^* is a local minimizer of $\bar{f}_{\epsilon,y^*}$ (for some $0 \leq \epsilon \leq \epsilon_0$), then it remains a local minimizer (even over the same local neighborhood) of $\bar{f}_{\mathcal{N}}(x) := \max_{y \in \mathcal{N}} f(x, y)$ for any $\mathcal{N}(y^*, \epsilon) \subseteq \mathcal{N} \subseteq \mathcal{N}(y^*, \epsilon_0)$.*

Proof. We first note that since y^* maximizes $f(x^*, y)$ over $\mathcal{N}(y^*, \epsilon_0)$, we clearly have for all $y^* \in \mathcal{N} \subseteq \mathcal{N}(y^*, \epsilon_0)$:

$$\bar{f}_{\mathcal{N}}(x^*) = f(x^*, y^*). \tag{2.22}$$

Moreover, for any $\mathcal{N} \supseteq \mathcal{N}(y^*, \epsilon)$ and any $x \in \mathcal{X}$:

$$\bar{f}_{\mathcal{N}}(x) \geq \bar{f}_{\epsilon,y^*}(x) =: \bar{f}_\epsilon(x). \tag{2.23}$$

Since x^* is a local minimizer of $\bar{f}_\epsilon$, say over the neighborhood $\mathcal{M}$, we have for all $x \in \mathcal{M}$ and $\mathcal{N}(y^*, \epsilon) \subseteq \mathcal{N} \subseteq \mathcal{N}(y^*, \epsilon_0)$:

$$\bar{f}_{\mathcal{N}}(x) \geq \bar{f}_\epsilon(x) \geq \bar{f}_\epsilon(x^*) = f(x^*, y^*) = \bar{f}_{\mathcal{N}}(x^*), \tag{2.24}$$

i.e., x^* is a local minimizer of $\bar{f}_{\mathcal{N}}(x)$ over the same local neighborhood $\mathcal{M}$. □

Note that in the lemma above we allow $\epsilon = 0$. Lemma 2.2.5 reveals a key property of the local minimax point in Definition 2.2.3: the norm in the neighborhood definition (see (2.14)) is immaterial (since we can shrink the neighborhood using Lemma 2.2.5 without impairing local minimaximality). In other words, the definition of local minimax points is topological and it does not depend on the norm we actually choose.

Using Lemma 2.2.5 we can "strengthen" the notion of local minimax even more. In particular, if Definition 2.2.3 holds for one sequence such that $\epsilon_0 \geq \epsilon_n \to 0$ then it automatically holds for *all* sequences that satisfy this same condition. We can even extend the sequence to an interval of ϵ's:

Proposition 2.2.6 (**equivalent definition of local minimax**). *The pair* $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ *is a local minimax point iff*

- *Fixing* x^* *then* y^* *is a local maximizer of* $\underline{f}_{0,x^*}(y) = f(x^*, y)$*;*
- *Fixing* y^* *then* x^* *is a local minimizer of* $\bar{f}_{\epsilon,y^*}(x)$ *for all* $\epsilon \in (0, \epsilon_0]$ *with some* $\epsilon_0 > 0$*.*

Proof. We need only prove if (x^*, y^*) is local minimax according to Definition 2.2.3, then there exists some $\epsilon_0 > 0$ such that x^* is a local minimizer of $\bar{f}_\epsilon(x)$ for all $\epsilon \in (0, \epsilon_0]$. Indeed, from Definition 2.2.3 we know $f(x^*, y)$ is maximized at y^* over some neighborhood $\mathcal{N}(y^*, \epsilon_0)$ for some $\epsilon_0 > 0$. For any $0 < \epsilon \leq \epsilon_0$, one can find $0 < \epsilon_n < \epsilon$ since the promised sequence $\epsilon_n \to 0$. By definition x^* is a local minimizer for $\bar{f}_{\epsilon_n}$, hence by Lemma 2.2.5 it remains a local minimizer for $\bar{f}_\epsilon$. □

From Definition 2.2.3, every uniformly local minimax point is local minimax. In fact, much more can be said between uniformly local minimax and local saddle:

Proposition 2.2.7 (**local saddle and uniformly local minimax**). *Every local saddle point is uniformly local minimax. If for any* $x \in \mathcal{X}$*,* $f(x, \cdot)$ *is upper semi-continuous, then every uniformly local minimax point is local saddle.*

Proof. Let $(x_\star, y_\star)$ be local saddle, i.e., $y_\star$ maximizes $f(x_\star, \cdot)$ over the neighborhood $\mathcal{N}(y_\star, \epsilon)$ and $x_\star$ minimizes $\bar{f}_{0,y_\star} = f(\cdot, y_\star)$ over the neighborhood $\mathcal{N}(x_\star, \epsilon)$. We fix the neighborhood $\mathcal{N}(x_\star) = \mathcal{N}(x_\star, \epsilon)$ and choose any sequence $\{\epsilon_n\} \subset (0, \epsilon]$. Applying Lemma 2.2.5 we know $x_\star$ remains a minimum for all $\bar{f}_{\epsilon_n}$ over the (fixed) neighborhood $\mathcal{N}(x_\star)$. Thus, $(x_\star, y_\star)$ is uniformly local minimax.

Conversely, let f be upper semi-continuous (in y for any x) and (x^*, y^*) uniformly local minimax over the fixed neighborhood $\mathcal{N}(x^*)$. By definition y^* maximizes $f(x^*, \cdot)$ over some neighborhood $\mathcal{N}(y^*, \epsilon_0)$, and x^* minimizes all $\bar{f}_{\epsilon_n}$ over the fixed neighborhood $\mathcal{N}(x^*)$, where the positive sequence $\epsilon_n \to 0$. Fix any $x \in \mathcal{N}(x^*)$. Since $f(x, \cdot)$ is upper semi-continuous at y^*, we have for any $\delta > 0$, there exists $\epsilon_n \in (0, \epsilon_0]$ such that:

$$f(x^*, y^*) = \bar{f}_{\epsilon_n}(x^*) \leq \bar{f}_{\epsilon_n}(x) \leq f(x, y^*) + \delta. \tag{2.25}$$

Letting $\delta \to 0$ we know $f(x, y^*) \geq f(x^*, y^*)$ for any $x \in \mathcal{N}(x^*)$. □

Thus, for upper semi-continuous functions (in y), surprisingly, local saddle points coincide with uniformly local minimax points. We cannot drop the semi-continuity assumption:

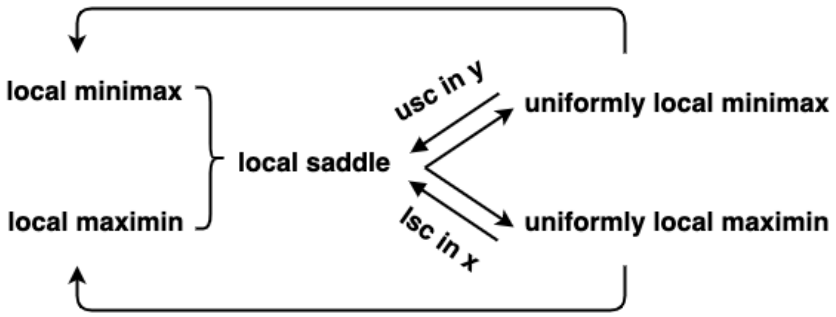


Figure 2.1: The relationship among different notions of local optimality. usc: upper semi-continuity and lsc: lower semi-continuity. The arrow and the bracket signs mean "to imply." For example, a uniformly local minimax point is *bona fide* local minimax, and if a point is both local minimax and local maximin, it is local saddle.

Example 2.2.8 (**uniformly local minimax does not imply local saddle without semi-continuity**). *Fix any $y^* \in \mathcal{Y}$ and consider the lower semi-continuous function*

$$f(x,y) = \begin{cases} -x^2, & y = y^* \\ x^2, & y \neq y^* \end{cases}, \quad \textit{with} \quad \bar{f}_{\epsilon,y^*}(x) = \begin{cases} -x^2, & \epsilon = 0 \\ x^2, & \epsilon \neq 0 \end{cases}. \tag{2.26}$$

$(0, y^*)$ is uniformly local minimax but not local saddle.

Figure 2.1 shows the relation between local saddle and (uniformly) local minimax (maximin) points. Finally, we prove our Definition 2.2.3 coincides with the seemingly different one in Jin et al. (2020, Definition 14). Effectively, we manage to remove the continuity assumption in their Lemma 16 (c.f. Proposition 2.2.6).

Proposition 2.2.9 (**equivalence with Jin et al. (2020)**). *The pair (x^*, y^*) is local minimax w.r.t. the function f iff there exists $\delta_0 > 0$ and a nonnegative function h satisfying $h(\delta) \to 0$ as $\delta \to 0$, such that for any $\delta \in (0, \delta_0]$ and any $(x, y) \in \mathcal{N}(x^*, \delta) \times \mathcal{N}(y^*, \delta)$ we have*

$$f(x^*, y) \leq f(x^*, y^*) \leq \left[\max_{y' \in \mathcal{N}(y^*, h(\delta))} f(x, y') \right] =: \bar{f}_{h(\delta)}(x). \tag{2.27}$$

Proof. ($\Longleftarrow$) Suppose (x^*, y^*) satisfies (2.27). Then clearly, y^* maximizes $f(x^*, \cdot)$ over the neighborhood $\mathcal{N}(x^*, \delta_0)$. Take an arbitrary positive sequence $\{\delta_n\}$ with $\delta_n \to 0$ and let $\epsilon_n = \sup_{m \geq n} h(\delta_n)$. Since $h(\delta) \to 0$ as $\delta \to 0$, we may assume WLOG that ϵ_n is well-defined

and bounded from above. If $h(\delta_n) = 0$ for some n then (x^*, y^*) is local saddle and hence local minimax thanks to Proposition 2.2.7. Otherwise we have $\epsilon_n > 0$ for all ϵ_n and $\epsilon_n \to 0$ since $\lim_{\delta \to 0} h(\delta) = 0$. WLOG we assume $\epsilon_1 \leq \delta_0$ (for otherwise we may discard the head of the sequence $\{\epsilon_n\}$). From (2.27) we know for any $x \in \mathcal{N}(x^*, \delta_n)$:

$$\bar{f}_{h(\delta_n)}(x) \geq f(x^*, y^*) = \bar{f}_{h(\delta_n)}(x^*), \tag{2.28}$$

since $h(\delta_n) \leq \epsilon_1 \leq \delta_0$ and y^* maximizes $f(x^*, y)$ over $\mathcal{N}(x^*, \delta_0)$. Therefore, x^* is a local minimizer of $\bar{f}_{h(\delta_n)}$ hence also of $\bar{f}_{\epsilon_n}$ thanks to Lemma 2.2.5.

($\Longrightarrow$) Suppose (x^*, y^*) is local minimax (see Definition 2.2.3). Then, y^* maximizes $f(x^*, \cdot)$ over some neighborhood $\mathcal{N}(y^*, \epsilon_0)$ where $\epsilon_0 > 0$. Since x^* is a local minimizer of $\bar{f}_{\epsilon_n}$, it minimizes $\bar{f}_{\epsilon_n}$ over some neighborhood $\mathcal{N}(x^*, \delta_n')$ with $\delta_n' > 0$. From $\{\delta_n'\}$ we construct another positive sequence $\{\delta_n\}$ where $\delta_0 = \min\{\delta_1', 1, \epsilon_0\}$ ¿ 0 and

$$\delta_n = \min\{\delta_n', \delta_{n-1}, 1/n\}, \quad n = 1, 2, \ldots, \tag{2.29}$$

which is diminishing by construction. Define $h(\delta) = \epsilon_n$ if $\delta_{n+1} < \delta \leq \delta_n$. Since $\epsilon_n \to 0$, $\lim_{\delta \to 0} h(\delta) = 0$. WLOG we assume $\epsilon_1 \leq \epsilon_0$ and by definition $\delta_0 \leq \epsilon_0$. For any $\delta \in (0, \delta_0]$ there exists some n such that $\delta \in (\delta_{n+1}, \delta_n]$. Thus, for any $(x, y) \in \mathcal{N}(x^*, \delta_n') \times \mathcal{N}(y^*, \epsilon_0)$:

$$\bar{f}_{h(\delta)}(x) = \bar{f}_{\epsilon_n}(x) \geq \bar{f}_{\epsilon_n}(x^*) = f(x^*, y^*) \geq f(x^*, y). \tag{2.30}$$

Since $\delta \leq \delta_n \leq \delta_n'$ and $\delta \leq \epsilon_0$, the above still holds over the smaller neighborhood $\mathcal{N}(x^*, \delta) \times \mathcal{N}(y^*, \delta)$, which is exactly (2.27). □

From this equivalence, we can also derive that every local saddle point is local minimax (Jin et al., 2020, Proposition 17). However, our Proposition 2.2.7 gives a more detailed depiction of local saddle points. For functions that are convex in x and concave in y, we naturally expect that local optimality is somehow equivalent to global optimality:

Theorem 2.2.10 (local and global minimax points in the convex-concave case). *Let the differentiable function $f(x, y)$ be convex in x and concave in y. Then, an interior point (x, y) is local minimax iff it is stationary, i.e., $\partial_x f(x, y) = \mathbf{0}$ and $\partial_y f(x, y) = \mathbf{0}$ iff it is saddle. In particular, local minimax implies global minimax.*

Proof. Suppose (x^*, y^*) is stationary. For any small $\epsilon > 0$,

$$\bar{f}_\epsilon(x) = \max_{y \in \mathcal{N}(y^*, \epsilon)} f(x, y) \tag{2.31}$$

is convex by our assumption. To see that x^* is a local (hence global) minimizer of $\bar{f}_\epsilon$, we need only verify that $\mathbf{0} \in \partial \bar{f}_\epsilon(x^*)$. Since y^* maximizes $f(x^*, \cdot)$ by assumption, we know from Danskin's theorem that $\partial \bar{f}_\epsilon(x^*) \ni \partial_x f(x^*, y^*) = \mathbf{0}$ since (x^*, y^*) is stationary.

Now suppose (x^*, y^*) is local minimax. Then, y^* is a local hence global maximizer of $f(x^*, \cdot)$. Also, x^* is a local hence global minimizer of $\bar{f}_\epsilon$. Thus,

$$\bar{f}(x) \geq \bar{f}_\epsilon(x) \geq \bar{f}_\epsilon(x^*) = f(x^*, y^*) = \bar{f}(x^*), \tag{2.32}$$

i.e., x^* is a global minimizer of $\bar{f}$. □

However, non-stationary global minimax points cannot be local minimax, see Example 2.1.7 and Proposition 2.3.1 (below). Even with stationarity, the convex-concave assumption in Theorem 2.2.10 cannot be appreciably weakened, as illustrated in the following example:

Example 2.2.11 (**stationary global minimax points are not local minimax in the nonconvex case**). *Let $f(x, y) = x^3 y$ which is nonconvex in x but linear in y. The point $(x^*, y^*) = (0, 1)$ is clearly stationary and global minimax. We verify that*

$$\bar{f}_\epsilon(x) = \begin{cases} (1+\epsilon)x^3, & x \geq 0 \\ (1-\epsilon)x^3, & x \leq 0 \end{cases}, \tag{2.33}$$

hence $x^* = 0$ is not a local minimizer of $\bar{f}_\epsilon$ (for any $\epsilon < 1$) and $(0, 1)$ is not local minimax. This counterexample is constructed by performing the $\mathcal{C}^1$ homeomorphic transformation $(x, y) \mapsto (x^3, y)$ of the bilinear game $b(x, y) = xy$. We can verify that (separate) homeomorphisms transform local/global minimax points accordingly. However, $\mathcal{C}^1$ homeomorphisms can turn non-stationary points into stationary (which is not possible in presence of convexity since in convex settings stationarity equates minimality which is preserved under homeomorphisms).

Nevertheless, for quadratic games, we can remove the convexity-concavity assumption, as will be shown in Theorem 2.4.1 below.

2.2.4 Other Notions of Local/Global Optimality

Besides stationary points, saddle points, and minimax points, there are also other definitions of local/global optimality, which we will briefly introduce here for completeness.

Evtushenko (1974a) proposed a different notion of local minimax points:

Definition 2.2.12 (**Evtushenko's local minimax**). *We call $z^* = (x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ a local optimal solution of f if there exists a neighborhood $\mathcal{N}(x^*) \times \mathcal{N}(y^*)$ such that z^* is a global minimax point of the problem*

$$\min_{x \in \mathcal{N}(x^*)} \max_{y \in \mathcal{N}(y^*)} f(x, y). \tag{2.34}$$

However, different from Definition 2.2.3, this definition may not always satisfy stationarity, as we can adapt Example 2.1.7 to construct such a counterexample. Interestingly, the sufficient condition for Definition 2.2.3 (see Corollary 2.3.14):

$$\partial^2_{yy} f \prec \mathbf{0} \text{ and } \partial^2_{xx} f - \partial^2_{xy} f (\partial^2_{yy} f)^{-1} \partial^2_{yx} f \succ \mathbf{0},$$

is also sufficient for Definition 2.2.12 (Evtushenko, 1974a). Such points are called *strict local minimax points*. Several methods using second-order information have been proposed for finding strict local minimax points, including the ones we will see in Chapter 4.

The next two definitions are proposed for GAN training, but can also be written for general settings. In Farnia and Ozdaglar (2020), the authors proposed an interpolation between global saddle points and global minimax points:

Definition 2.2.13 (**λ-proximal equilibrium**). *We call $z^* = (x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ a λ-proximal equilibrium (with $\lambda \geq 0$) if*

$$f(x^*, y) \leq f(x^*, y^*) \leq \max_{y \in \mathcal{Y}} f(x, y) - \lambda \|y - y^*\|^2, \, \forall x \in \mathcal{X}, y \in \mathcal{Y}. \tag{2.35}$$

Since the right hand side satisfies:

$$f(x, y^*) \leq \max_{y \in \mathcal{Y}} f(x, y) - \lambda \|y - y^*\|^2 \leq \max_{y \in \mathcal{Y}} f(x, y), \tag{2.36}$$

we know that every global saddle point is a λ-proximal equilibrium and every λ-proximal equilibrium is a global minimax point. Based on this definition, they proposed a new algorithm for training GANs, which improves the state-of-the-art in terms of inception scores.

Finally, to define a computable solution concept, some people use the dynamics of gradient algorithms as a definition. For example, Mazumdar et al. (2018); Berard et al. (2020) define local stable stationary points, by computing the Jacobian of the vector field $v(x, y) = (\partial_x f(x, y), -\partial_y f(x, y))$ and analyzing the spectrum of the Jacobian:

Definition 2.2.14 (**local stable saddle point**). *We call* $z^* = (x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ *a locally stable stationary point (LSSP) if we have* $v(x, y) = (\partial_x f(x, y), -\partial_y f(x, y)) = \mathbf{0}$ *and* $\Re(\lambda) > 0$ *for any* λ *in the spectrum of the Jacobian:*

$$\nabla v(x^*, y^*) = \begin{bmatrix} \partial^2_{xx} f(x^*, y^*) & \partial^2_{xy} f(x^*, y^*) \\ -\partial^2_{yy} f(x^*, y^*) & -\partial^2_{yy} f(x^*, y^*) \end{bmatrix}. \tag{2.37}$$

It can be shown that if at a stationary point $z^* = (x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ such that

$$\partial_x f(x^*, y^*) = \partial_y f(x^*, y^*) = \mathbf{0}, \partial^2_{xx} f(x^*, y^*) \succ \mathbf{0} \succ \partial^2_{yy} f(x^*, y^*), \tag{2.38}$$

then it is an LSSP (Mazumdar et al., 2018). Such a stationary point is also a saddle point. Berard et al. (2020) shows that in GAN training, the final solution we want to converge to can be an LSSP but not a saddle point.

2.3 Optimality Conditions

Optimality conditions are an indispensable part of optimization (Bertsekas, 1997) since they help us identify local optimal points and design new algorithms. In this section, we provide first- and second-order necessary and sufficient conditions for local minimax (maximin) points. Our results extend existing ones in Jin et al. (2020) to cases where the domains $\mathcal{X}$ and $\mathcal{Y}$ are constrained and where the Hessian for the max-player $\partial^2_{yy} f$ is not invertible. We assume $\mathcal{X}$ and $\mathcal{Y}$ are closed and thus $\mathcal{N}(y^*, \epsilon)$ is compact. We build on some classical results in nonsmooth analysis, for which we provide a self-contained review in Appendix A.1, including the definition of the directional derivative $\mathsf{D}\bar{f}(x; t)$ of an envelope function $\bar{f}$ at x along direction t:

$$\mathsf{D}\bar{f}(x; t) = \lim_{\alpha \to 0^+} \frac{\bar{f}(x + \alpha t) - \bar{f}(x)}{\alpha}. \tag{2.39}$$

Specifically, if f and $\partial_x f$ are jointly continuous (continuous w.r.t. (x, y)), then the directional derivative $\mathsf{D}\bar{f}(x; t)$ always exist (Theorem A.1.9). In the following subsections, $f \in \mathcal{C}^p$ means that f is p^{th} continuously differentiable.

2.3.1 First-order Optimality Conditions

Theorem 2.3.1 (**first-order necessary, local minimax**). *Let* $f \in \mathcal{C}^1$. *At a local minimax point* (x^*, y^*), *we have:*

$$\partial_x f(x^*, y^*)^\top \bar{t} \geq 0 \geq \partial_y f(x^*, y^*)^\top \underline{t}, \tag{2.40}$$

for any directions $\bar{t} \in \mathsf{K}_{\mathsf{d}}(\mathcal{X}, x^*)$, $\underline{t} \in \mathsf{K}_{\mathsf{d}}(\mathcal{Y}, y^*)$, *where the cone*

$$\mathsf{K}_{\mathsf{d}}(\mathcal{X}, x) := \liminf_{\alpha \to 0^+} \frac{\mathcal{X} - x}{\alpha} := \{t : \forall \{\alpha_k\} \to 0^+ \ \exists \{\alpha_{k_i}\} \to 0^+, \{t_{k_i}\} \to t, \\ \text{such that } x + \alpha_{k_i} t_{k_i} \in \mathcal{X}\}$$

and $\mathsf{K}_{\mathsf{d}}(\mathcal{Y}, y)$ *is defined similarly.*

Proof. Use Theorem A.1.3, Theorem A.1.9 and the assumption that $f \in \mathcal{C}^1$. □

In the theorem above, $\mathsf{K}_{\mathsf{d}}(\mathcal{X}, x)$ is known as the derivable cone (Rockafellar and Wets, 2009, p. 198), which may strictly include the feasible tangent cone. When the set $\mathcal{X}$ is closed and convex, $\mathsf{K}_{\mathsf{d}}(\mathcal{X}, x)$ is the same as the tangent cone (Hiriart-Urruty and Lemaréchal, 2004, p. 65):

$$\mathsf{K}_{\mathsf{d}}(\mathcal{X}, x) = \overline{\text{cone}}(\mathcal{X} - x) := \text{cl}(d \in \mathbb{R}^n : d = \alpha(y - x),\ y \in \mathcal{X},\ \alpha \geq 0), \tag{2.41}$$

with cl denoting the closure of a set. We can derive a similar reduction when $\mathcal{Y}$ is closed and convex. If both $\mathcal{X}$ and $\mathcal{Y}$ are closed and convex, then (2.40) reduces to:

$$\partial_x f(x^*, y^*)^\top (x - x^*) \geq 0 \geq \partial_y f(x^*, y^*)^\top (y - y^*), \ \text{ for any } x \in \mathcal{X},\ y \in \mathcal{Y}. \tag{2.42}$$

This can be regarded as a bi-variate version of first-order (necessary) optimality condition for a local minimum (Bertsekas, 1997, Prop. 2.1.2). Solutions that satisfy (2.42) are often called stationary points (c.f. Section 2.2.1). It extends the result in Jin et al. (2020) to the constrained case. Specifically, if (x^*, y^*) is in the interior, in particular when $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y} = \mathbb{R}^m$, then Proposition 2.3.1 simplifies to

$$\partial_x f(x^*, y^*) = \mathbf{0}, \quad \partial_y f(x^*, y^*) = \mathbf{0}, \tag{2.43}$$

which agrees with Jin et al. (2020). Local minimax points have the same necessary conditions, (2.40), (2.42) and (2.43), as local saddle points (e.g. Barazandeh and Razaviyayn, 2020, Definition 2). It also implies that in the convex-concave case, all notions of optimality agree:

Corollary 2.3.2 (local optimal solutions in the convex-concave case). *Let* $\mathcal{X}$ *and* $\mathcal{Y}$ *be convex and the function* $f(x, y)$ *be convex in* x *and concave in* y. *A point is local (global) saddle iff it is local minimax (maximin) iff it is a stationary point.*

Proof. In convex cases, stationarity is equivalent to optimality (Bertsekas, 1997, Prop. 2.1.2). □

However, this corollary does not hold in the non-convex setting, see Examples 2.4.3.

Let us define the *active sets* of the *zeroth* order (by "zeroth" we mean that only the function values are involved):

$$\mathcal{Y}_0(x^*;\epsilon) = \{y \in \mathcal{N}(y^*,\epsilon) : \bar{f}_\epsilon(x^*) = f(x^*,y)\}, \tag{2.44}$$

We derive the first-order sufficient conditions for local minimax points (which follow from the sufficient condition in Theorem A.1.5 and Danskin's theorem in Theorem A.1.9):

Theorem 2.3.3 (**first-order sufficient condition, local minimax**). *Assume $\partial_x f(x,y)$ is continuous. If $f(x^*,\cdot)$ is maximized at y^* over a neighborhood around y^*, and there exists $\epsilon_0 > 0$ such that for any $\epsilon \in (0,\epsilon_0)$,*

$$\mathbf{0} \neq t \in \mathsf{K}_\mathsf{c}(\mathcal{X},x^*) \implies \mathsf{D}\bar{f}_\epsilon(x^*;t) = \max_{y\in\mathcal{Y}_0(x^*;\epsilon)} \partial_x f(x^*,y)^\top t > 0, \tag{2.45}$$

where the contingent cone is defined as:

$$\mathsf{K}_\mathsf{c}(\mathcal{X},x) := \limsup_{\alpha\to 0^+} \frac{\mathcal{X}-x}{\alpha} := \{t : \exists\{\alpha_k\}\to 0^+,\ \{t_k\}\to t,\ \textit{such that } x+\alpha_k t_k \in \mathcal{X}\},$$

then (x^*,y^*) is a local minimax point.

In the case when $\mathcal{X}$ is a convex set. $\mathsf{K}_\mathsf{c}(\mathcal{X},x)$ reduces to:

$$\mathsf{K}_\mathsf{c}(\mathcal{X},x) = \overline{\text{cone}}(\mathcal{X}-x) := \text{cl}(d\in\mathbb{R}^n : d = \alpha(y-x),\ y\in\mathcal{X},\ \alpha\geq 0). \tag{2.46}$$

(2.45) thus becomes that for any $x^* \neq x \in \mathcal{X}$:

$$\max_{y\in\mathcal{Y}_0(x^*;\epsilon)} \partial_x f(x^*,y)^\top (x-x^*) > 0,\ \forall x\in\mathcal{X}. \tag{2.47}$$

Let us demonstrate the first order condition with the following example:

Example 2.3.4 (**application of the first-order sufficient condition of local minimax points**). *Suppose $f(x,y) = xy$ is bilinear. At $(x^*,y^*) = (0,0)$, we have:*

$$\bar{f}_\epsilon(x^*) = f(x^*,y) = 0,\ \forall y\in\mathbb{R}. \tag{2.48}$$

Therefore, according to (2.44), $\mathcal{Y}_0(x^*;\epsilon) = \mathcal{N}(y^*,\epsilon)$. Also, $\partial_x f(x^*,y) = y$ and

$$\mathsf{D}\bar{f}_\epsilon(x^*;x-x^*) = \max_{\mathcal{N}(y^*,\epsilon)} y(x-x^*) = \epsilon|x| > 0, \forall x\neq x^*. \tag{2.49}$$

According to Theorem 2.3.3, (x^*,y^*) is a local minimax point.

2.3.2 Second-order Optimality Conditions

We now turn to the second-order necessary condition of local minimax points. We sometimes use $\partial^2_{xx} f$ as a shorthand for the second-order derivative $\partial^2_{xx} f(x^*, y^*)$, and similarly for other second-order partial derivatives. For a local minimax point (x^*, y^*), y^* maximizes $f(x^*, \cdot)$ locally, and thus we have the property that $\bar{f}_\epsilon(x^*) = f(x^*, y^*)$ for any small ϵ, from which we can make significant simplifications. The following technical lemma, when combined with the necessity condition in Theorem A.1.3, allows us to classify the directions:

Lemma 2.3.5 (**directional derivatives for different** $\bar{f}_\epsilon$)**.** *Suppose f and $\partial_x f$ are jointly continuous and thus the directional derivative exists. If y^* is a local maximizer of $f(x^*, \cdot)$ over a neighborhood $\mathcal{N}(y^*, \epsilon_0)$, then for any $0 \le \epsilon_1 \le \epsilon_2 \le \epsilon_0$, $\mathcal{Y}_0(x^*; \epsilon_1) \subseteq \mathcal{Y}_0(x^*; \epsilon_2)$ and for each $t \in \mathsf{K}_\mathsf{d}(\mathcal{X}, x^*)$, $\mathsf{D}\bar{f}_{\epsilon_2}(x^*; t) \ge \mathsf{D}\bar{f}_{\epsilon_1}(x^*; t)$.*

Indeed, for a local minimax point (x^*, y^*) and any direction $t \in \mathsf{K}_\mathsf{d}(\mathcal{X}, x^*)$, we know from the necessity condition in Theorem A.1.3 that $\mathsf{D}\bar{f}_\epsilon(x^*; t) \ge 0$ for all small ϵ, which, combined with Lemma 2.3.5 above, leaves us with two possibilities:

1. $\mathsf{D}\bar{f}_\epsilon(x^*; t) > 0$ for all $\epsilon > 0$ smaller than some $\epsilon_0(t)$;
2. $\mathsf{D}\bar{f}_\epsilon(x^*; t) = 0$ for all $\epsilon > 0$ smaller than some $\epsilon_0(t)$.

We call the direction t a *critical direction* in the second case above. With this distinction among directions, we derive the second-order necessary condition for local minimax points:

Theorem 2.3.6 (**second-order necessary condition, local minimax**)**.** *Suppose $f, \partial_x f$ and $\partial^2_{xx} f$ are all (jointly) continuous. If (x^*, y^*) is a local minimax point, then for each direction $t \in \mathsf{K}_\mathsf{d}(\mathcal{X}, x^*)$, one of the following holds:*

1. *$\mathsf{D}\bar{f}_\epsilon(x^*; t) > 0$ for all $\epsilon > 0$ smaller than some $\epsilon_0(t)$;*
2. *$\mathsf{D}\bar{f}_\epsilon(x^*; t) = 0$ for all $\epsilon > 0$ smaller than some $\epsilon_0(t)$ (i.e. t is critical), in which case we further have*

$$t^\top \partial^2_{xx} f(x^*, y^*) t + \tfrac{1}{2} \limsup_{z \to y^*} \left[\max\{\partial_x f(x^*, z)^\top t,\, 0\}^2 (f(x^*, y^*) - f(x^*, z))^\dagger \right] \ge 0, \tag{2.50}$$

where $t^\dagger = 1/t$ if $t \neq 0$ and 0 otherwise.

The important point to take from Theorem 2.3.6 is that we should test the second order condition (2.50) only for critical directions, and the second-order derivatives of f may not fully capture the second-order derivatives of the envelope function $\bar{f}_\epsilon$, which can be clearly demonstrated from the following examples:

Example 2.3.7 (**the importance of critical directions**). *Let*

$$f(x,y) = -x^2 + xy^3$$

be defined over $\mathcal{X} = \mathcal{Y} = \mathbb{R}$ and consider the local minimax point $(x^*, y^*) = (0,0)$. Indeed, for any $\epsilon > 0$, x^* is a local minimizer of $\bar{f}_\epsilon(x) = |x|\epsilon^3 - x^2$. However, $\partial^2_{xx} f = -2$ while $f(x^*, y^*) = f(x^*, z) = 0$ for any z. Thus, the second-order condition (2.50) fails at the directions $t = \pm 1$. However, there is no contradiction since these directions are not critical: Indeed, using Theorem A.1.9 we can verify that $\mathsf{D}\bar{f}_\epsilon(x^*; \pm 1) = \epsilon^3 > 0$.

Example 2.3.8 (**the importance of critical directions; high dimensional**). *Let*

$$f(x,y) = -x_2^2 + x_2 y_2^3 - (y_1 + y_2)^2 + 2x_1(y_1 + y_2)$$

be defined over $\mathcal{X} = \mathcal{Y} = \mathbb{R}^2$ and consider the local minimax point $(x^*, y^*) = (\mathbf{0}, \mathbf{0})$: Indeed, $f(x^*, \cdot)$ is clearly maximized locally at $y^* = \mathbf{0}$ and upon choosing $y_1 = x_1 - sgn(x_2)\epsilon/2, y_2 = sgn(x_2)\epsilon/2$ and considering $|x_1| < \epsilon/2$ and $|x_2| < (\epsilon/2)^3$, we have

$$\|y - x\|_\infty \le \epsilon/2 + (\epsilon/2)^3,\ \bar{f}_\epsilon(x) \ge f(x,y) = x_1^2 + |x_2|(\epsilon/2)^3 - x_2^2 \ge 0 = \bar{f}_\epsilon(x^*), \tag{2.51}$$

where we choose WLOG the ℓ_∞ norm in our neighborhood definition (2.14). The second-order derivatives are:

$$\partial^2_{yx} f = \begin{bmatrix} 2 & 0 \\ 2 & 0 \end{bmatrix},\ \partial^2_{yy} f = \begin{bmatrix} -2 & -2 \\ -2 & -2 \end{bmatrix},\ \partial^2_{xx} f = \begin{bmatrix} 0 & 0 \\ 0 & -2 \end{bmatrix}. \tag{2.52}$$

We have $\mathcal{Y}_0(x^*; \epsilon) = \{y \in \mathcal{N}_\infty(x^*, \epsilon) : y_1 + y_2 = 0\}$ and for any direction t,

$$\mathsf{D}\bar{f}_\epsilon(x^*; t) = \max_{y \in \mathcal{Y}_0(x^*;\epsilon)} t^\top \partial_x f(x^*, y) = \epsilon^3 |t_2| \ge 0. \tag{2.53}$$

It follows that the critical directions satisfy $t_2 = 0$. Take a non-critical direction $t = (1,3)$, we easily verify that $(\partial^2_{yx} f)t = (2,2)$ lies in the range space of $\partial^2_{yy} f$. However,

$$\begin{aligned} &\limsup_{z \to y^*} \left[\max\{\partial_x f(x^*, z)^\top t,\, 0\}^2 (f(x^*, y^*) - f(x^*, z))^\dagger\right] \\ &= \limsup_{z \to \mathbf{0}, z_1 + z_2 \neq 0} \frac{[2(z_1 + z_2) + 3z_2^3]_+^2}{(z_1 + z_2)^2} = 4, \end{aligned} \tag{2.54}$$

so that the second-order condition in (2.50), which in this case coincides with

$$t^\top(\partial^2_{xx}f - \partial^2_{xy}f(\partial^2_{yy}f)^\dagger\partial^2_{yx}f)t,$$

does not hold ($-18+2=-16 \not\succeq 0$). Nevertheless, along a critical direction t (where $t_2=0$):

$$t^\top\partial^2_{xx}f(x^*,y^*)t = 0,\ f(x^*,z) = -(z_1+z_2)^2,\ \partial_x f(x^*,z)^\top t = 2t_1(z_1+z_2), \tag{2.55}$$

and thus the left-hand side of (2.50) simplifies to $2t_1^2 \geq 0$. In other words, the second-order condition indeed holds for critical directions.

Example 2.3.9 (high order derivatives might be involved in Theorem 2.3.6). *The second term in (2.50) may involve higher order information of f, rather than the standard second-order optimality condition for the minimizer of a smooth function that only relies on second order derivatives. The higher order term comes from the difference of function values. Let $f(x,y) = -x^2 - y^4 + 4xy^2$ and consider the local minimax point $(x^*,y^*) = (0,0)$. We have $\mathcal{Y}_0(x^*;\epsilon) = \{y^*\}$ hence every direction is critical. In the direction $t=1$, the l.h.s. of (2.50) becomes $-2 + \max\{4z^2t, 0\}^2/(2z^4) = 6 > 0$.*

Under the condition that $\partial^2_{yy}f$ is invertible, we show the following as in Jin et al. (2020):

Corollary 2.3.10 (second-order necessary condition, invertible). *Let $f \in \mathcal{C}^2$. At a local minimax point (x^*,y^*) in the interior of $\mathcal{X}\times\mathcal{Y}$, if $\partial^2_{yy}f$ß is invertible, then*

$$\partial^2_{yy}f \prec \mathbf{0} \text{ and } \partial^2_{xx}f - \partial^2_{xy}f(\partial^2_{yy}f)^{-1}\partial^2_{yx}f \succeq \mathbf{0}. \tag{2.56}$$

Proof. It is easy to prove $\partial^2_{yy}f \preceq \mathbf{0}$ and since $\partial^2_{yy}f$ is invertible, we have $\partial^2_{yy}f \prec \mathbf{0}$. By expanding $f(x^*,z)$ to the second order, the second term in (2.50) becomes:

$$\limsup_{z\to y^*} \frac{\max\{(z-y^*)^\top(\partial^2_{yx}f)t, 0\}^2}{(z-y^*)^\top(-\partial^2_{yy}f)(z-y^*)}. \tag{2.57}$$

With a change of variables $z - y^* = (-\partial^2_{yy}f)^{-1/2}(w-y^*)$ and using Cauchy–Schwarz inequality, we obtain $-t^\top\partial^2_{xy}f(\partial^2_{yy}f)^{-1}(\partial^2_{yx}f)t$. It follows that $\partial^2_{xx}f - \partial^2_{xy}f(\partial^2_{yy}f)^{-1}\partial^2_{yx}f \succeq \mathbf{0}$. □

Finally, we can compare our second order necessary condition with Jin et al. (2020, Proposition 19), which applies to quadratic functions (cf. Example 2.4.2). The difference is that Jin et al. (2020, Proposition 19) did not take the critical directions and higher order derivatives into consideration, as demonstrated by Examples 2.3.7 and 2.3.9.

Second-order sufficient conditions

We introduce two second-order sufficient conditions for local minimax points, with the help of results from non-smooth optimization literature (Seeger, 1988; Kawasaki, 1992). Our results extend Jin et al. (2020) to a case when $\partial^2_{yy} f$ is not invertible, which may happen in real applications.

In the following theorem, we define $x_+ = \max\{x, 0\}$ and the first order activation set:

$$\mathcal{Y}_1(x^*; \epsilon; t) = \{y \in \mathcal{Y}_0(x^*, \epsilon) : \mathsf{D}\bar{f}_\epsilon(x^*; t) = \partial_x f(x^*, y)^\top t\}. \tag{2.58}$$

Theorem 2.3.11 (**second-order sufficient condition, local minimax**). *Assume $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y}$ is convex and f, $\partial_x f$, $\partial^2_{xx} f$ are (jointly) continuous. At a stationary point (x^*, y^*), if there exists $\epsilon_0 > 0$ such that:*

- *$f(x^*, \cdot)$ is maximized at y^* on $\mathcal{N}(y^*, \epsilon_0)$;*
- *along each critical direction $t \neq \mathbf{0}$:*

$$t^\top \partial^2_{xx} f(x^*, y^*) t + \frac{1}{2} \limsup_{z \to y^*} \left(((\partial_x f(x^*, z)^\top t)_+)^2 (f(x^*, y^*) - f(x^*, z))^\dagger\right) > 0, \tag{2.59}$$

and in any direction $d \in \mathbb{R}^m$, there exist $\alpha, \beta \neq 0$ and $p, q > 0$ such that for every $y \in \mathcal{Y}_1(x^*; \epsilon_0; t)$, the following Taylor expansion holds:

$$f(x^*, y + \delta d) = f(x^*, y) + \alpha\delta^p + o(\delta^p),\ \partial_x f(x^*, y + \delta d)^\top t = \beta\delta^q + o(\delta^q), \tag{2.60}$$

then (x^*, y^*) is a local minimax point.

Proof. It follows from Theorem A.1.17. From Danskin's theorem $\mathsf{D}\bar{f}_\epsilon(x^*; t) \geq 0$ for any small $\epsilon > 0$. Besides, for any small enough ϵ, (A.72) is satisfied since $y^* \in \mathcal{Y}_1(x^*; \epsilon_0; t)$. Noting that $\bar{f}_\epsilon(x^*) = f(x^*, y^*) = f(x^*, y)$ for any $0 \leq \epsilon < \epsilon_0$ and $y \in \mathcal{Y}_1(x^*; \epsilon_0; t)$, (2.60) follows from Assumption A.1.16. □

Note that in the statement above, the variables α, β and p, q may depend on the direction d. If $f \in \mathcal{C}^\infty$ is smooth and both $f(x^*, \cdot)$ and $\partial_x f(x^*, \cdot)^\top t$ have non-zero Taylor expansions, then (2.60) is always true for every $y \in \mathcal{Y}_1(x^*; \epsilon_0; t)$. Here by "critical direction" we mean that $\mathsf{D}\bar{f}_\epsilon(x^*; t) = 0$ for some $\epsilon_0 > 0$ and any $\epsilon \in [0, \epsilon_0]$. Another second-order sufficient condition for $f \in \mathcal{C}^2$ is:

Theorem 2.3.12 (**second-order sufficient condition, local minimax**). *Assume $f \in \mathcal{C}^2$ and let $\mathcal{X}$ be convex. Suppose y^* is a local maximizer of $f(x^*, \cdot)$ and that (x^*, y^*) is an interior stationary point. If there is $\epsilon_0 > 0$ such that for any $\epsilon \in (0, \epsilon_0]$, there exists $R, r > 0$ such that for any feasible direction $\|t\| = 1$ such that $0 \leq \mathsf{D}\bar{f}_\epsilon(x^*; t) \leq r$,*

$$\max_{y \in \mathcal{Y}_0(x^*;\epsilon)} \max_{\substack{v \in \mathcal{V}(x^*,y;t) \\ \|v\| \leq R}} \max_{\substack{w \in \mathsf{K_d}(\Omega,y;v), \\ \|w\| \leq R}} \left\langle \begin{bmatrix} \partial^2_{xx} f(x^*, y) & \partial^2_{xy} f(x^*, y) \\ \partial^2_{yx} f(x^*, y) & \partial^2_{yy} f(x^*, y) \end{bmatrix} \begin{pmatrix} t \\ v \end{pmatrix}, \begin{pmatrix} t \\ v \end{pmatrix} \right\rangle + \\ + \langle \partial_y f(x^*, y), w \rangle > 0, \tag{2.61}$$

then this point is local minimax, where $\mathcal{V}(x, y; t) := \{v \in \mathsf{K_d}(\Omega, y) : \mathsf{D}\bar{f}_\epsilon(x; t) = \partial_x f(x, y)^\top t + \partial_y f(x, y)^\top v\}$, $\Omega := \mathcal{N}(y^*, \epsilon)$ and

$$\mathsf{K_d}(\Omega, y; v) := \liminf_{t \to 0^+} \frac{\Omega - y - tv}{t^2/2} := \{g : \forall \{t_k\} \downarrow 0 \ \exists \{t_{k_i}\} \downarrow 0, \{g_{k_i}\} \to g, \\ y + t_{k_i} v + t^2_{k_i} g_{k_i}/2 \in \Omega\}. \tag{2.62}$$

Proof. Since $y^* \in \mathcal{Y}_0(x^*; \epsilon)$, from Danskin's theorem (Theorem A.1.9) we know that $\mathsf{D}\bar{f}_\epsilon(x^*; t) \geq 0$ for any ϵ small enough. We then combine Theorem A.1.6 with Theorem A.1.11. Note that all the directions t, v, w are bounded. □

The definition of feasible directions for convex sets can be seen in Hiriart-Urruty and Lemaréchal (e.g. 2013). We used the standard notation that if we are maximizing over an empty set, then the maximum is $-\infty$. Specifically, if there exists $y \in \mathcal{Y}_0(x^*, \epsilon)$ such that it is in the interior of $\mathcal{Y}$, Theorem 2.3.12 can be simplified as:

Corollary 2.3.13 (**second-order sufficient condition, interior version**). *Assume $f \in \mathcal{C}^2$ and let $\mathcal{X}$ be convex. Suppose y^* is a local maximizer of $f(x^*, \cdot)$ and that (x^*, y^*) is an interior stationary point. If there is $\epsilon_0 > 0$ such that $\mathcal{N}(y^*, \epsilon_0) \subset \mathcal{Y} \subset \mathbb{R}^m$, and for any $\epsilon \in (0, \epsilon_0)$, there exist $R, r > 0$ such that for any feasible direction $\|t\| = 1$ that satisfies $0 \leq \mathsf{D}\bar{f}_\epsilon(x^*; t) \leq r$, we have:*

$$\max_{y \in \mathcal{Y}_0(x^*;\epsilon)} \max_{\substack{v \in \mathcal{V}(x^*,y;t) \\ \|v\| \leq R}} \max_{\|w\| \leq R} \left\langle \begin{bmatrix} \nabla^2_{xx} f(x^*, y) & \nabla^2_{xy} f(x^*, y) \\ \nabla^2_{yx} f(x^*, y) & \nabla^2_{yy} f(x^*, y) \end{bmatrix} \begin{pmatrix} t \\ v \end{pmatrix}, \begin{pmatrix} t \\ v \end{pmatrix} \right\rangle + \langle \nabla_y f(x^*, y), w \rangle > \mathbf{0}, \tag{2.63}$$

then this point is local minimax, where $\mathcal{V}(x, y; t) := \{v \in \mathbb{R}^m : \mathsf{D}\bar{f}_\epsilon(x; t) = \nabla_x f(x, y)^\top t + \nabla_y f(x, y)^\top v\}$.

Proof. If $y \in \mathcal{N}(y^*, \epsilon)$, then we have $\mathsf{K_d}(\Omega, y) = \mathsf{K_d}(\Omega, y; v) = \mathbb{R}^m$. □

In the special case when $\partial^2_{yy} f(x^*, y^*) \prec \mathbf{0}$, we have the following corollary. This special type of local minimax points that satisfy (2.64) are also known as *strict local minimax points* (Jin et al., 2020).

Corollary 2.3.14 (**second-order sufficient condition, invertible, Jin et al. (2020)**). *Let f be twice continuously differentiable. At an interior stationary point $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$, if*

$$\partial^2_{yy} f \prec \mathbf{0} \text{ and } \partial^2_{xx} f - \partial^2_{xy} f (\partial^2_{yy} f)^{-1} \partial^2_{yx} f \succ \mathbf{0}, \tag{2.64}$$

then (x^*, y^*) is a local minimax point.

Proof. The active set $\mathcal{Y}_0(x^*; \epsilon) = \{y^*\}$ is a singleton. From Danskin's theorem (Theorem A.1.9) all directions are critical. The l.h.s. of (2.61) becomes $t^\top (\partial^2_{xx} f - \partial^2_{xy} f (\partial^2_{yy} f)^{-1} \partial^2_{yx} f) t$ if we choose $R = \|(\partial^2_{yy} f)^{-1} \partial^2_{yx} f\|$. □

However, Corollary 2.3.14 does not fully cover Theorem 2.3.12 when ∂^2_{yy} is not invertible:

Example 2.3.15 (**Theorem 2.3.12 strictly includes Corollary 2.3.14**). *Take*

$$f(x, y) = xy^2 + x^2$$

and a stationary point $(x^*, y^*) = (0, 0)$. $\mathsf{D}\bar{f}_\epsilon(x^*; t) = \epsilon^2$ if $t = 1$ and $\mathsf{D}\bar{f}_\epsilon(x^*; t) = 0$ if $t = -1$. Take $r = \epsilon^2/2$. Along the critical direction $t = -1$, the l.h.s. of (2.61) becomes $2 > 0$, since $\partial_y f(x^*, y) = 0$, and $\mathcal{V}(x^*, y; t) = \varnothing$ if $y \neq 0$ and $\mathbb{R}$ if $y = 0$. So, $(0, 0)$ is local minimax from Theorem 2.3.12. Note that Corollary 2.3.11 does not apply since $f(x^*, y)$ does not have a non-zero Taylor expansion.

We also give an example when Theorem 2.3.12 is not applicable but Corollary 2.3.11 is:

Example 2.3.16 (**application of Theorem 2.3.11 where Theorem 2.3.12 cannot be applied**). *Take*

$$f(x, y) = xy^3 - y^6$$

and a stationary point $(x^*, y^*) = (0, 0)$. Fixing $x^* = 0$, $f(x^*, \cdot)$ is maximized at 0, and for any $t \neq 0$, $\mathsf{D}\bar{f}_\epsilon(x^*; t) = \max_{y^6 = 0} y^3 t = 0$. Since $\partial_x f(x^*, z) = z^3 t$ and $f(x^*, y^*) - f(x^*, z) = z^6$, the l.h.s. of (2.59) is $t^2/2 > 0$. Moreover, $\mathcal{Y}_1(x^*; \epsilon_0; t) = \{y^*\}$ for any $\epsilon_0 > 0$, and

$$f(x^*, y^* + \delta d) = -\delta^6 d^6, \ \partial_x f(x^*, y^* + \delta d)^\top t = \delta^3 d^3 t.$$

So, $(0, 0)$ is a local minimax point. Note that Theorem 2.3.12 does not apply since $\mathcal{Y}_0(x^*; \epsilon) = \{0\}$ and all second-order derivatives are zero.

2.4 Quadratic Games: A Case Study

In this section we study quadratic games with the following form:

$$q(x,y) = \frac{1}{2}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}^\top \begin{bmatrix} A & C & a \\ C^\top & B & b \\ a^\top & b^\top & c \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{2.65}$$

where $x \in \mathcal{X} = \mathbb{R}^n$ and $y \in \mathcal{Y} = \mathbb{R}^m$. In particular, a game is *bilinear* if A, B vanish and *homogeneous* if a, b vanish. Since quadratic games are continuous, local saddle points are the same as uniformly local minimax points (see Proposition 2.2.7).

Our first result completely characterizes stationary, global minimax and local minimax points for homogeneous quadratic games:

Theorem 2.4.1 (**sufficient and necessary conditions for optimality in quadratic games**). *For (homogeneous) unconstrained quadratic games, a pair (x, y) is*

- *stationary iff*

$$\begin{bmatrix} A & C \\ C^\top & B \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{0}; \tag{2.66}$$

- *global minimax iff $B \preceq \mathbf{0}$, $P_L^\perp (A - CB^\dagger C^\top) P_L^\perp \succeq \mathbf{0}$ where $L = CP_B^\perp$, and*

$$\begin{bmatrix} P_L^\perp & \\ & I \end{bmatrix} \begin{bmatrix} A & C \\ C^\top & B \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{0}; \tag{2.67}$$

  (Recall that $P_L^\perp = I - LL^\dagger$ is the orthogonal projection onto the null space of $L^\top$.)

- *local minimax iff $B \preceq \mathbf{0}$, $P_L^\perp (A - CB^\dagger C^\top) P_L^\perp \succeq \mathbf{0}$, and stationary (i.e. (2.66) holds). In particular, local minimax points are always global minimax.*

Proof. The first claim follows directly from the definition of stationarity.

To prove the second claim, we note that fixing x, $q(x, \cdot)$ is clearly quadratic in y. Thus, it admits a local (hence also global) maximizer y iff

$$B \preceq \mathbf{0}, \tag{2.68}$$

$$C^\top x + By = \mathbf{0}. \tag{2.69}$$

Note that there exists some y to satisfy (2.69) iff $C^\top x$ belongs to the range space of B iff

$$P_B^\perp C^\top x = \mathbf{0}, \ i.e. \ L^\top x = \mathbf{0}, \tag{2.70}$$

or equivalently $x = P_L^\perp z$ for some $z \in \mathbb{R}^m$. Therefore, we have the envelope function:

$$\bar{q}(x) = \begin{cases} \frac{1}{2} x^\top (A - CB^\dagger C^\top) x, & L^\top x = \mathbf{0} \\ \infty, & \text{otherwise} \end{cases}. \tag{2.71}$$

Thus, the quadratic function $\bar{q}$ (when restricted to the null space of $L^\top$) admits a local (hence also global) minimizer iff

$$P_L^\perp (A - CB^\dagger C^\top) P_L^\perp \succeq \mathbf{0}, \tag{2.72}$$

in which case the minimizer x satisfies

$$L^\top x = \mathbf{0} = P_L^\perp (A - CB^\dagger C^\top) x, \tag{2.73}$$

whereas the maximizer y satisfies (2.69). It is easy to verify that (2.73) and (2.69) are equivalent to (2.67). For the last claim, note first that we have proved in Theorem 2.3.1 that any local minimax point is stationary. Moreover, if (x^*, y^*) is local minimax, then x^* locally minimizes $\bar{q}_{\epsilon, y^*}$ (for all small ϵ), i.e., for x close to x^*, we have

$$\bar{q}(x) \geq \bar{q}_{\epsilon, y^*}(x) \geq \bar{q}_{\epsilon, y^*}(x^*) = q(x^*, y^*) = \bar{q}(x^*), \tag{2.74}$$

where the last equality follows since fixing x^*, y^* is a local hence also global maximizer of the quadratic function $q(x^*, \cdot)$. We have shown above that any local minimizer of $\bar{q}(x)$ is necessarily global. Therefore, (x^*, y^*) is global minimax.

Lastly, we prove the converse of the last claim. Let $B \preceq \mathbf{0}$, $P_L^\perp (A - CB^\dagger C^\top) P_L^\perp \succeq \mathbf{0}$, and (x^*, y^*) be stationary, i.e. they satisfy (2.66). Fixing y^* we have for all small $\epsilon > 0$:

$$2\bar{q}_\epsilon(x) = 2\bar{q}_{\epsilon, y^*}(x) = \max_{\|y - y^*\| \leq \epsilon} \begin{bmatrix} x \\ y \end{bmatrix}^\top \begin{bmatrix} A & C \\ C^\top & B \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}. \tag{2.75}$$

We are left to prove x^* is a local minimizer of $\bar{q}_\epsilon$ for all small ϵ.[1] Let $c = \max\{\|B^\dagger C^\top\|, \|A - CB^\dagger C^\top\|\}$. We assume first $c > 0$ and $L \neq \mathbf{0}$. Let σ be the smallest positive singular value of $L = CP_B^\perp$. Consider any x such that $\|x - x^*\| \leq \epsilon(\sigma \wedge 1)/(3c)$. We decompose

$$x - x^* = \boldsymbol{\delta}_\| + \boldsymbol{\delta}_\perp, \ \text{where } \boldsymbol{\delta}_\perp = P_L^\perp (x - x^*), \tag{2.76}$$

[1] Unfortunately we cannot use the sufficient conditions in Section 2.3.2 since x^* may not be an isolated local minimizer.

and define

$$y - y^* = -B^\dagger C^\top (x - x^*) + \epsilon L^\top (x - x^*)/(2\|L^\top (x - x^*)\|), \tag{2.77}$$

where by convention $0/0 := 0$. Clearly, $\|y - y^*\| \leq \epsilon/3 + \epsilon/2 < \epsilon$. Thus, using the stationarity of (x^*, y^*):

$$2\bar{q}_\epsilon(x) \geq 2q(x, y) = \begin{bmatrix} x - x^* \\ y - y^* \end{bmatrix}^\top \begin{bmatrix} A & C \\ C^\top & B \end{bmatrix} \begin{bmatrix} x - x^* \\ y - y^* \end{bmatrix} \tag{2.78}$$

$$(\text{note } BL^\top = \mathbf{0}) = (x - x^*)^\top (A - CB^\dagger C^\top)(x - x^*) + \epsilon \|L^\top (x - x^*)\| \tag{2.79}$$

$$= \delta_\|^\top (A - CB^\dagger C^\top)\delta_\| + 2\delta_\|^\top (A - CB^\dagger C^\top)\delta_\perp +$$
$$+ \delta_\perp^\top (A - CB^\dagger C^\top)\delta_\perp + \epsilon \|L^\top \delta_\|\| \tag{2.80}$$

$$\geq -\epsilon\sigma\|\delta_\|\|/3 - 2\epsilon\sigma\|\delta_\|\|/3 + 0 + \epsilon\sigma\|\delta_\|\| = 0 = 2\bar{q}_\epsilon(x^*), \tag{2.81}$$

where we used the fact that $\|\delta_\|\| \vee \|\delta_\perp\| \leq \epsilon\sigma/(3c)$ and $P_L^\perp (A - CB^\dagger C^\top) P_L^\perp \succeq \mathbf{0}$. Finally, we note that if $c = 0$, then $A - CB^\dagger C^\top = \mathbf{0}$ hence the proof still goes through (with c replaced by 1 say). Similarly, if $L = \mathbf{0}$, then $\delta_\| = \mathbf{0}$ hence the proof again goes through (with σ replaced by 1 say). □

Comparing Theorem 2.4.1 with Theorem 2.2.10, we find that in both cases, local minimax points are global minimax, which is not true in general (Example 2.4.9). This shows that there exists some "hidden convexity" in quadratic games when local/global minimax points exist: fixing any x, $q(x, \cdot)$ is concave in y; $\bar{q}(x)$ is convex in x (c.f. (2.71)).

Remark 2.4.2 (**application of Theorem 2.3.6 in quadratic games**). *We could also use Theorem 2.3.6 to obtain the necessary condition of local minimax points for quadratic games. First write*

$$f(x^*, y^*) - f(x^*, y) = -y^\top By/2 \text{ and } -\partial_x f(x^*, y)^\top t = -y^\top C^\top t$$

and $\mathsf{D}\bar{f}_\epsilon(x^*; t) \geq \delta\|P_B^\perp C^\top t\|$ *for some* $\delta > 0$. *The critical directions are* $t \in \mathcal{N}(P_B^\perp C^\top)$. *If* $BC^\top = \mathbf{0}$, *then* $\partial_x f(x^*, y)^\top t = 0$ *for any* y *and thus the second term in (2.50) is zero. So, we have* $P_L^\perp A P_L^\perp \succeq 0$ *with* $L = CP_B^\perp$. *Otherwise, take critical directions* t *such that* $t \in \mathcal{N}(P_B^\perp C^\top)$. *The second term in (2.50) becomes* $-t^\top CB^\dagger C^\top t$ *(using Cauchy–Schwarz). Combining with the case* $BC^\top = \mathbf{0}$, *we have* $P_L^\perp (A - CB^\dagger C^\top) P_L^\perp \succeq \mathbf{0}$.

We remark that the last claim of Theorem 2.4.1 does not follow from Theorem 2.2.10:

Example 2.4.3 (quadratic games can be nonconvex). *Let $A = -1, C = 1, B = 0, a = b = 0$. Then, from Theorem 2.4.1 $(x, y) = (0, 0)$ is local and global minimax. However, $q(x, y) = -\frac{1}{2}x^2 + xy$ is clearly non-convex in x (although $\bar{q}$ is convex). Also, $(0, 0)$ is not local saddle since $q(x, 0) \geq q(0, 0)$ does not hold.*

Theorem 2.4.4 (equivalence between global and local minimax in quadratic games). *An unconstrained quadratic game admits a global minimax point iff it admits a local minimax point iff*

$$B \preceq \mathbf{0}, \quad P_L^{\perp}(A - CB^{\dagger}C^{\top})P_L^{\perp} \succeq \mathbf{0}, \text{ and } \begin{bmatrix} a \\ b \end{bmatrix} \in \mathcal{R}\left(\begin{bmatrix} A & C \\ C^{\top} & B \end{bmatrix}\right). \tag{2.82}$$

For such quadratic games, local minimax coincides with stationarity and are global minimax.

Proof. If (2.82) holds, let

$$\begin{bmatrix} A & C \\ C^{\top} & B \end{bmatrix} \begin{bmatrix} x^* \\ y^* \end{bmatrix} = \begin{bmatrix} a \\ b \end{bmatrix}. \tag{2.83}$$

Then, performing the translation $(x, y) \leftarrow (x - x^*, y - y^*)$ we reduce to the homogeneous case and applying Theorem 2.4.1 we obtain the existence of a local (or global) minimax point. If a local minimax point exists, then stationarity yields the range condition. Performing translation and applying Theorem 2.4.1 again establishes all conditions in (2.82).

All we are left to prove is when a global minimax point (x^*, y^*) exists the range condition holds. Indeed, fixing x^*, y^* maximizes the quadratic $q(x^*, \cdot)$ hence from stationarity:

$$C^{\top}x^* + By^* = b. \tag{2.84}$$

The above equation has a solution y^* iff $P_B^{\perp}C^{\top}x^* = P_B^{\perp}b$, i.e. $L^{\top}x^* = P_B^{\perp}b$ (recall that $L := CP_B^{\perp}$). Solving y and plugging back in q we obtain: for all x such that $L^{\top}x = P_B^{\perp}b$,

$$\bar{q}(x) = \tfrac{1}{2}x^{\top}(A - CB^{\dagger}C^{\top})x + x^{\top}CB^{\dagger}b - a^{\top}x. \tag{2.85}$$

Since x^* is a global minimizer of $\bar{q}$, we obtain the stationarity condition:

$$P_L^{\perp}[(A - CB^{\dagger}C^{\top})x^* + CB^{\dagger}b - a] = \mathbf{0}. \tag{2.86}$$

Combined with (2.84) we obtain:

$$P_L^{\perp}[Ax^* + CB^{\dagger}By^* - a] = \mathbf{0} \iff Ax^* + CB^{\dagger}By^* - a = Lz = CP_B^{\perp}z \text{ for some } z \tag{2.87}$$

$$\iff Ax^* + C(B^{\dagger}By^* + P_B^{\perp}z) = a \tag{2.88}$$

From (2.84) and (2.88) we deduce $(x^*, B^{\dagger}By^* + P_B^{\perp}z)$ satisfies the range condition (2.83). □

In this theorem we used $\mathcal{R}(\cdot)$ to denote the range of a matrix. It is clear that stationary, global minimax, and local minimax points are characterized in the same way as in Theorem 2.4.1: we need only replace $\mathbf{0}$ on the right-hands of (2.66) and (2.67) with the vector $[a; b]$. These points always form an affine subspace for quadratic games.

Theorem 2.4.4 allows us to completely classify (unconstrained) quadratic games:

- no stationary points (hence no local or global minimax points);
- exist stationary points but no global or local minimax point;
- exist local minimax points which coincide with global minimax points.
- exist local minimax points which are strictly contained in global minimax points.

Clearly, for homogeneous (unconstrained) quadratic games, stationary points always exist hence only the last three cases can happen. For (nontrivial) bilinear games, only the last case can happen:

Corollary 2.4.5 (**bilinear games**). *For (homogeneous) unconstrained bilinear games ($A = \mathbf{0}, B = \mathbf{0}, C \neq \mathbf{0}, a = \mathbf{0}, b = \mathbf{0}$), global minimax points are $\mathsf{null}(C^\top) \times \mathbb{R}^n$ while local minimax points (i.e. stationary points) are $\mathsf{null}(C^\top) \times \mathsf{null}(C)$.*

It is thus clear that even in bilinear games, there exist global minimax points that are not local minimax. From Theorem 2.4.4, we can derive that:

Corollary 2.4.6 (**saddle points in quadratic games**). *For (unconstrained) quadratic games, the following statements are equivalent:*

1. *Local saddle points exist.*
2. *Local maximin and minimax points exist.*
3. *Global saddle points exist.*
4. *Global maximin and minimax points exist.*
5. *$A \succeq \mathbf{0} \succeq B$, and*

$$\begin{bmatrix} a \\ b \end{bmatrix} \in \mathcal{R}\left(\begin{bmatrix} A & C \\ C^\top & B \end{bmatrix}\right). \tag{2.89}$$

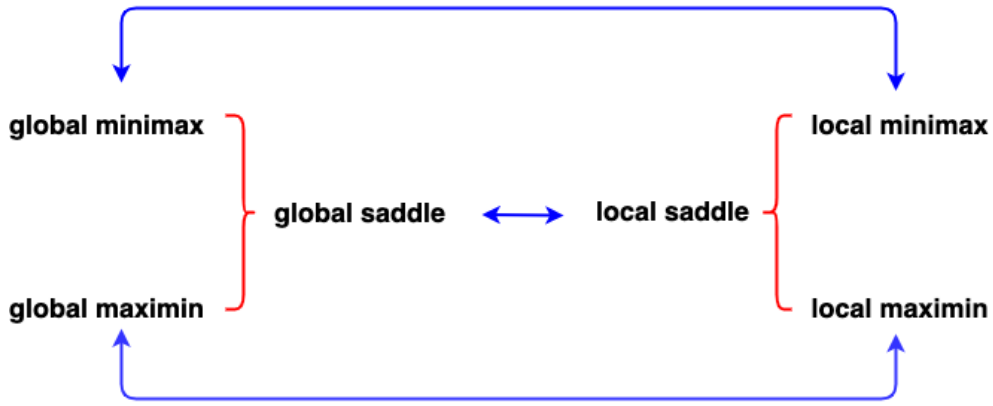


Figure 2.2: The relation among definitions in quadratic games. $A \longleftrightarrow B$ means A exists iff B exists. The brackets also show the existence relation. For example, global saddle points exist iff both global minimax and maximin points exist.

6. *stationary points exist and they are all local (global) saddle.*

A summary of Corollary 2.4.6 and Theorem 2.4.1 can be visualized at Figure 2.2. Note that we used $\mathcal{R}(\cdot)$ to denote the range of a matrix. We remark that Theorem 2.4.6 does not follow from typical minimax theorems (such as Sion's) since our domain is unbounded and we do not assume convexity-concavity from the outset. Thus, Theorem 2.4.6 reveals strong duality under weaker assumptions than the usual convexity-concavity. This is in stark contrast with generic NCNC games (see Example 2.1.6).

Remark 2.4.7 (**non-uniformly local minimax in quadratic games**). *Since quadratic functions are continuous (and thus upper semi-continuous), from Prop. 2.2.7 we know that local saddle points are equivalent to uniformly minimax points. By comparing Theorem 2.4.6 and Theorem 2.4.4, whenever $A \succeq \mathbf{0} \succeq B$ and (2.89) holds, local saddle points and thus uniformly local minimax points exist. However, if (2.82) holds but $A \succeq \mathbf{0}$ does not hold, local saddle points/uniformly local minimax points do not exist from Theorem 2.4.6, but local minimax points still exist from Theorem 2.4.4 which are hence non-uniform. We can see it more clearly from Exammple 2.4.3. One can compute $\bar{q}_\epsilon(x) = \epsilon|x| - \frac{1}{2}x^2$, and obtain that $\bar{q}_\epsilon(x) \geq \bar{q}_\epsilon(0) = 0$ iff $|x| \leq 2\epsilon$. According to Definition 2.2.3 the point $(0, 0)$ is non-uniformly local minimax.*

Theorem 2.4.6 reveals some fundamental and surprising properties of quadratic games. On the one hand, quadratic games consist of an important theoretical tool for understanding general smooth NCNC games (through local Taylor expansion) (e.g. Daskalakis and

Panageas, 2018; Jin et al., 2020; Ibrahim et al., 2020; Wang et al., 2020). On the other hand, they are really special and many of their unique properties do not carry over to general smooth NCNC games, as we demonstrate in the following examples:

Example 2.4.8 (**stationary/global minimax points exist, no local minimax points**). *For general NCNC games, the existence of a global minimax point may not imply the existence of local minimax points. Indeed, consider*

$$f(x,y) = -y^4/4 + y^2/2 - xy, \quad x \in \mathbb{R}, \quad y \in \mathbb{R}. \tag{2.90}$$

We claim $(\pm 1, 0)$ *are the only global minimax points. Indeed,*

$$\bar{f}(x) = \max_y -y^4/4 + y^2/2 - xy = \max_{y \geq 0} -y^4/4 + y^2/2 + |x|y \geq \max_{y \geq 0} -y^4/4 + y^2/2 = 1/4.$$

Clearly, the inequality is attained only at $x_* = 0$ *and* $y_* = \pm 1$. *Its only stationary point is* $(x, y) = (0, 0)$. *However,* $\partial^2_{yy} f(0,0) = 1$ *hence* $y = 0$ *cannot be a local maximizer of* $f(0, \cdot)$.

Note that in this example the global minimax points are not stationary. For an example where a stationary and global minimax point exists with no local minimax point, please refer to Example 2.2.11.

Example 2.4.9 (**local minimax exists, no global minimax**). *This is possible even for separable functions, such as* $f(x,y) = x^3 - x - y^2$ *defined on* $\mathbb{R} \times \mathbb{R}$. *Clearly, it has a local minimax point at* $(1/\sqrt{3}, 0)$ *but no global minimax points exist.*

Example 2.4.10 (**local minimax and local maximin points exist; no local saddle**). *We can also construct an example when both local minimax and local maximin points exist but there is no local saddle point. Take* $f_1(x,y) = g(x,y)h(x,y)$, *where*

$$g(x,y) = xy - x^2, \text{ and } h(x,y) = \exp\left(-\frac{1}{1-x^2}\right) \mathbf{1}_{|x|<1} \exp\left(-\frac{1}{1-y^2}\right) \mathbf{1}_{|y|<1}$$

is a bump function that smoothly interpolates between the unit box and the outside. By numerically computing the stationary points and checking the second order conditions, we found there is no such a point where $\partial^2_{xx} f_1 \geq 0$ *and* $\partial^2_{yy} f_1 \leq 0$ *in the open box* $\mathcal{B}_1 = \{(x,y) : |x| < 1, |y| < 1\}$. *In other words, local saddle points do not exist. There is a local minimax point* $(0,0)$ *since*

$$\bar{f}_\epsilon(x) \geq (\epsilon|x| - x^2) \exp(-1/(1-x^2)) \exp(-1/(1-\epsilon^2)) \geq 0$$

when $|x| \leq \epsilon$ *and* $\epsilon^2 < 1$. *Similarly we can construct* $f_2(x,y) = -g(y-10, x-10)h(x-10, y-10)$ *where there is a local maximin point but no local saddle point in the open box* $\mathcal{B}_2 = \{(x,y) : |x-10| < 1, |y-10| < 1\}$. *Therefore,* $f(x,y) = f_1(x,y) + f_2(x,y)$ *has both local minimax and local maximin points, but there is no local saddle point on* $\mathcal{B}_1 \cup \mathcal{B}_2$.

Chapter 3

Stability of Gradient Algorithms

In minimax optimization, it is well-known that gradient algorithms may not always be stable at a desirable optimal solution. It is important to understand if a gradient algorithm would even converge before we prove some convergence result. In this chapter we study the stability of gradient algorithms. With a powerful tool from control theory called Schur's theorem, we are able to characterize exactly the hyper-parameter choices (e.g. step size, momentum coefficient) with which a gradient algorithm could (locally and linearly) converge. For instance, in bilinear games, we show that adding momentum to simultaneous Gradient Descent Ascent would not yield convergence; in general cases, we find that having a more aggressive extra-gradient step could enhance stability.

3.1 Linear Dynamical System and Schur's Theorem

In this section, we study *linear dynamical systems* (LDSs, a.k.a. matrix iterative processes, Varga (1962)). We define a general k-step LDS as follows:

$$z^{(t)} = \sum_{i=1}^{k} A_i z^{(t-i)} + d, \tag{3.1}$$

where $d \in \mathbb{R}^b$, $z^{(i)} \in \mathbb{R}^b$ for $i = 1, 2, \ldots,$ and $A_i \in \mathbb{R}^{b \times b}$ for $i = 1, 2, \ldots, k$. On the l.h.s. of (3.1), $t \geq k$ and we initialize from $\{z^{(0)}, \ldots, z^{(k-1)}\}$. As we will see later, iterative gradient algorithms can be reduced to linear dynamical systems in terms of bilinear games as well as local dynamics. Understanding the stability of gradient algorithms reduces to understanding the stability of LDSs.

Define the characteristic polynomial of our LDS (3.1), with $A_0 = -I$:

$$p(\lambda) := \det(\textstyle\sum_{i=0}^{k} A_i \lambda^{k-i}). \tag{3.2}$$

The following well-known result decides when such a k-step LDS converges for any initialization:

Theorem 3.1.1 (e.g. Gohberg et al. (1982))**.** *The LDS in eq. (3.1) converges for any initialization $(z^{(0)}, \ldots, z^{(k-1)})$ iff the spectral radius $r := \max\{|\lambda| : p(\lambda) = 0\} < 1$, in which case $\{z^{(t)}\}$ converges linearly with an (asymptotic) exponent r.*

Therefore, understanding the dynamics of an LDS reduces to root analysis of the characteristic polynomial.

3.1.1 Schur's Theorem

The (sufficient and necessary) convergence condition in Theorem 3.1.1 reduces to that all roots of the characteristic polynomial $p(\lambda)$ lie in the (open) unit disk, which can be conveniently analyzed through the celebrated Schur's theorem (Schur, 1917):

Theorem 3.1.2 (Schur (1917))**.** *The roots of a real polynomial $p(\lambda) = a_0\lambda^n + a_1\lambda^{n-1} + \cdots + a_n$ are within the (open) unit disk of the complex plane iff $\forall k \in \{1, 2, \ldots, n\}$, $\det(P_k P_k^{\mathsf{H}} - Q_k^{\mathsf{H}} Q_k) > 0$, where P_k, Q_k are $k \times k$ matrices defined as: $[P_k]_{i,j} = a_{i-j}\mathbf{1}_{i \geq j}$, $[Q_k]_{i,j} = a_{n-i+j}\mathbf{1}_{i \leq j}$.*

In the theorem above, we denoted $\mathbf{1}_S$ as the indicator function of the event S, i.e. $\mathbf{1}_S = 1$ if S holds and $\mathbf{1}_S = 0$ otherwise. The superscript H denotes the Hermitian conjugate of a matrix. For a nice summary of related stability tests, see Mansour (2011). We therefore define *Schur stable* polynomials to be those polynomials whose roots all lie within the (open) unit disk of the complex plane. For real polynomials, Schur's theorem has the following corollary:

Corollary 3.1.3 (**real polynomial**, e.g. Mansour (2011))**.** *A real quadratic polynomial $\lambda^2 + a\lambda + b$ is Schur stable iff $b < 1$, $|a| < 1 + b$; A real cubic polynomial $\lambda^3 + a\lambda^2 + b\lambda + c$ is Schur stable iff $|c| < 1$, $|a + c| < 1 + b$, $b - ac < 1 - c^2$; A real quartic polynomial $\lambda^4 + a\lambda^3 + b\lambda^2 + c\lambda + d$ is Schur stable iff $|c - ad| < 1 - d^2$, $|a + c| < b + d + 1$, and $b < (1 + d) + (c - ad)(a - c)/(d - 1)^2$.*

Proof. It suffices to prove the result for quartic polynomials. We write down the matrices:

$$P_1 = [1],\, Q_1 = [d], \tag{3.3}$$

$$P_2 = \begin{bmatrix} 1 & 0 \\ a & 1 \end{bmatrix},\, Q_2 = \begin{bmatrix} d & c \\ 0 & d \end{bmatrix}, \tag{3.4}$$

$$P_3 = \begin{bmatrix} 1 & 0 & 0 \\ a & 1 & 0 \\ b & a & 1 \end{bmatrix}, Q_3 = \begin{bmatrix} d & c & b \\ 0 & d & c \\ 0 & 0 & d \end{bmatrix}, \tag{3.5}$$

$$P_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ a & 1 & 0 & 0 \\ b & a & 1 & 0 \\ c & b & a & 0 \end{bmatrix}, Q_4 = \begin{bmatrix} d & c & b & a \\ 0 & d & c & b \\ 0 & 0 & d & c \\ 0 & 0 & 0 & d \end{bmatrix}. \tag{3.6}$$

We require $\det(P_k P_k^\top - Q_k^\top Q_k) =: \delta_k > 0$, for $k = 1, 2, 3, 4$. If $k = 1$, we have $1 - d^2 > 0$, namely, $|d| < 1$. $\delta_2 > 0$ reduces to $(c - ad)^2 < (1 - d^2)^2$ and thus $|c - ad| < 1 - d^2$ due to the first condition. $\delta_4 > 0$ simplifies to:

$$-((a + c)^2 - (b + d + 1)^2)((b - d - 1)(d - 1)^2 - (a - c)(c - ad))^2 < 0, \tag{3.7}$$

which yields $|a + c| < |b + d + 1|$. Finally, $\delta_3 > 0$ reduces to:

$$((b - d - 1)(d - 1)^2 - (a - c)(c - ad))((d^2 - 1)(b + d + 1) + (c - ad)(a + c)) > 0. \tag{3.8}$$

Denote $p(\lambda) := \lambda^4 + a\lambda^3 + b\lambda^2 + c\lambda + d$, we must have $p(1) > 0$ and $p(-1) > 0$, as otherwise there is a real root λ_0 with $|\lambda_0| \geq 1$. Hence we obtain $b + d + 1 > |a + c| > 0$. Also, from $|c - ad| < 1 - d^2$, we know that:

$$|c - ad| \cdot |a + c| < |b + d + 1|(1 - d^2) = (b + d + 1)(1 - d^2). \tag{3.9}$$

So, the second factor in (3.8) is negative and the positivity of the first factor reduces to:

$$b < (1 + d) + \frac{(c - ad)(a - c)}{(d - 1)^2}. \tag{3.10}$$

To obtain the Schur condition for cubic polynomials, we take $d = 0$, and the quartic Schur condition becomes:

$$|c| < 1,\, |a + c| < b + 1,\, b - ac < 1 - c^2. \tag{3.11}$$

To obtain the Schur condition for quadratic polynomials, we take $c = 0$ in the above and write:

$$b < 1,\, |a| < 1 + b. \tag{3.12}$$

The proof is now complete. □

We may also encounter complex polynomials in the study of local dynamics, and we give a corollary for complex quadratic polynomials:

Corollary 3.1.4 (**complex polynomial**). *For complex quadratic polynomials $\lambda^2+a\lambda+b$, the exact convergence condition is:*

$$|b| < 1,\ (1-|b|^2)^2 + 2\Re(a^2\bar{b}) > |a|^2(1+|b|^2). \tag{3.13}$$

Proof. For quadratic polynomials, we compute

$$P_1 = [1],\ Q_1 = [b], \tag{3.14}$$

$$P_2 = \begin{bmatrix} 1 & 0 \\ a & 1 \end{bmatrix},\ Q_2 = \begin{bmatrix} b & a \\ 0 & b \end{bmatrix}, \tag{3.15}$$

We require $\det(P_kP_k^{\mathsf{H}} - Q_k^{\mathsf{H}}Q_k) =: \delta_k > 0$, for $k = 1, 2$. If $k = 1$, we have $1-|b|^2 > 0$. If $k = 2$, we have:

$$P_kP_k^{\mathsf{H}} - Q_k^{\mathsf{H}}Q_k = \begin{bmatrix} 1-|b|^2 & \bar{a}-a\bar{b} \\ a-\bar{a}b & 1-|b|^2 \end{bmatrix}, \tag{3.16}$$

where $\bar{a}$ means the complex conjugate. The determinant should be positive, so we have:

$$(1-|b|^2)^2 + 2\Re(a^2\bar{b}) > |a|^2(1+|b|^2). \tag{3.17}$$

□

3.1.2 Solving Stability Conditions through `Mathematica`

In later sections we will study the stability conditions of gradient algorithms using Corollary 3.1.3 and Corollary 3.1.4. This requires simplification of polynomial inequality arrays. Although there are systematic ways for it using tools in algebraic geometry (Collins, 1975), they will be inevitably tedious computation. For this regard, we will rely on Mathematica code in our proofs (mostly with the built-in function `Reduce`) and in principle the code can be verified manually using cylindrical algebraic decomposition.[1]

[1] See the online Mathematica documentation https://reference.wolfram.com/language/tutorial/SomeNotesOnInternalImplementation.html.

3.2 Bilinear Games

In the study of GAN training, bilinear games are often regarded as a simple yet important example for theoretically analyzing and understanding new algorithms and techniques (e.g. Daskalakis et al., 2018; Gidel et al., 2019a,b; Liang and Stokes, 2019). It captures the difficulty in GAN training and can represent some simple GAN formulations (Arjovsky et al., 2017; Daskalakis et al., 2018; Gidel et al., 2019a; Mescheder et al., 2018). Mathematically, *bilinear* zero-sum games can be formulated as the following minimax optimization problem:

$$\min_{x\in\mathbb{R}^n} \max_{y\in\mathbb{R}^n} \quad x^\top E y + b^\top x + c^\top y. \tag{3.18}$$

The set of all saddle points (Definition 2.1.1) is:

$$\{(x, y) \,|\, Ey + b = \mathbf{0},\ E^\top x + c = \mathbf{0}\}. \tag{3.19}$$

Throughout, for simplicity we assume E to be invertible. We also assume x and y to have the same dimension. The analysis is not fundamentally different if x and y have different dimensions or E is non-invertible (Zhang and Yu, 2020). The linear terms are not essential in our analysis and we take $b = c = \mathbf{0}$ throughout this section[2]. In this case, the only saddle point is $(\mathbf{0}, \mathbf{0})$. For bilinear games, it is well-known that simultaneous gradient descent ascent does not converge (Nemirovsky and Yudin, 1983) and other gradient-based algorithms tailored for minimax optimization have been proposed (Korpelevich, 1976; Daskalakis et al., 2018; Gidel et al., 2019a; Mescheder et al., 2017).

3.2.1 Gradient Algorithms

We define some popular gradient algorithms for finding saddle points in the general unconstrained minimax optimization problem

$$\min_{x\in\mathbb{R}^n} \max_{y\in\mathbb{R}^m} f(x, y). \tag{3.20}$$

We present gradient algorithms for a general (bivariate) function f. Note that we introduced more "step sizes" for our refined analysis, as we find that the enlarged parameter space often contains choices for faster linear convergence. All hyperparameters below including α_1, α_2, γ_1, γ_2, β_1, β_2 are positive.

[2]If they are not zero, one can translate x and y to cancel the linear terms, see e.g. Gidel et al. (2019b).

Gradient Descent Ascent The generalized GDA update has the following form:

$$x^{(t+1)} = x^{(t)} - \alpha_1 \partial_x f(x^{(t)}, y^{(t)}), \qquad y^{(t+1)} = y^{(t)} + \alpha_2 \partial_y f(x^{(t)}, y^{(t)}). \tag{3.21}$$

When $\alpha_1 = \alpha_2$, the convergence of averaged iterates (a.k.a. Cesari convergence) for convex-concave games is analyzed in (Bruck, 1977; Nemirovski and Yudin, 1978; Nedić and Ozdaglar, 2009). Recent progress on interpreting GDA with dynamical systems can be seen in, e.g., Mertikopoulos et al. (2018a); Bailey et al. (2019); Bailey and Piliouras (2018).

Extra-Gradient We study a generalized version of EG, defined as follows:

$$x^{(t+1/2)} = x^{(t)} - \gamma_1 \partial_x f(x^{(t)}, y^{(t)}),\ y^{(t+1/2)} = y^{(t)} + \gamma_2 \partial_y f(x^{(t)}, y^{(t)}); \tag{3.22}$$

$$x^{(t+1)} = x^{(t)} - \alpha_1 \partial_x f(x^{(t+1/2)}, y^{(t+1/2)}),\ y^{(t+1)} = y^{(t)} + \alpha_2 \partial_y f(x^{(t+1/2)}, y^{(t+1/2)}). \tag{3.23}$$

EG was first proposed in Korpelevich (1976) with the restriction $\alpha_1 = \alpha_2 = \gamma_1 = \gamma_2$, under which linear convergence was proved for bilinear games. Convergence of EG on convex-concave games was analyzed in Nemirovski (2004); Monteiro and Svaiter (2010), and Mertikopoulos et al. (2019) provides convergence guarantees for specific non-convex-non-concave problems. For bilinear games, a slightly more generalized version was proposed in Liang and Stokes (2019) where $\alpha_1 = \alpha_2$, $\gamma_1 = \gamma_2$, with linear convergence proved. For later convenience we define $\beta_1 = \alpha_2\gamma_1$ and $\beta_2 = \alpha_1\gamma_2$.

Optimistic Gradient Descent We study a generalized version of OGD, defined as follows:

$$x^{(t+1)} = x^{(t)} - \alpha_1 \partial_x f(x^{(t)}, y^{(t)}) + \beta_1 \partial_x f(x^{(t-1)}, y^{(t-1)}), \tag{3.24}$$

$$y^{(t+1)} = y^{(t)} + \alpha_2 \partial_y f(x^{(t)}, y^{(t)}) - \beta_2 \partial_y f(x^{(t-1)}, y^{(t-1)}). \tag{3.25}$$

The original version of OGD was given in Popov (1980) with $\alpha_1 = \alpha_2 = 2\beta_1 = 2\beta_2$ and rediscovered in the GAN literature (Daskalakis et al., 2018). Its linear convergence for bilinear games was proved in Liang and Stokes (2019). A slightly more generalized version with $\alpha_1 = \alpha_2$ and $\beta_1 = \beta_2$ was analyzed in Peng et al. (2020); Mokhtari et al. (2020), again with linear convergence proved. The stochastic case was analyzed in Hsieh et al. (2019).

It has been observed recently in Mokhtari et al. (2020) that for convex-concave games, EG ($\alpha_1 = \alpha_2 = \gamma_1 = \gamma_2 = \eta$) and OGD ($\alpha_1/2 = \alpha_2/2 = \beta_1 = \beta_2 = \eta$) can be treated as approximations of the proximal point algorithm (Martinet, 1970; Rockafellar, 1976) when η is small. With this result, one can show that EG and OGD converge to saddle points sublinearly for smooth convex-concave games (Mokhtari et al., 2019).

Momentum Methods Generalized heavy ball method was analyzed in Gidel et al. (2019b):

$$x^{(t+1)} = x^{(t)} - \alpha_1 \partial_x f(x^{(t)}, y^{(t)}) + \beta_1 (x^{(t)} - x^{(t-1)}), \tag{3.26}$$

$$y^{(t+1)} = y^{(t)} + \alpha_2 \partial_y f(x^{(t)}, y^{(t)}) + \beta_2 (y^{(t)} - y^{(t-1)}). \tag{3.27}$$

This is a modification of Polyak's heavy ball (HB) (Polyak, 1964), which also motivated Nesterov's accelerated gradient algorithm (NAG) (Nesterov, 1983). Note that for both x-update and the y-update, we *add* a scalar multiple of the successive difference (e.g. proxy of the momentum). For this algorithm our result below improves those obtained in Gidel et al. (2019b), as will be discussed the next section.

3.2.2 Simultaneous and Alternating Updates

In the last subsection we have introduced a few gradient algorithms. In fact, they are all algorithms with simultaneous updates, i.e., the variables x and y are updated simultaneously. In numerical linear algebra this is also known as the *Jacobi update*. There is a different way to update the variables, called the alternating update, which means updating the variables one after another. For instance, the alternating update version of GDA (3.21) is:

$$x^{(t+1)} = x^{(t)} - \alpha_1 \partial_x f(x^{(t)}, y^{(t)}), \qquad y^{(t+1)} = y^{(t)} + \alpha_2 \partial_y f(x^{(t+1)}, y^{(t)}), \tag{3.28}$$

where we update $y^{(t+1)}$ based on $x^{(t+1)}$ rather than $x^{(t)}$. In this way, we hope the algorithm is more stable. Alternating updates are also called *Gauss–Seidel updates*. A well-known result is the Stein–Rosenberg theorem (Stein and Rosenberg, 1948), which shows that Gauss–Seidel (GS) updates converge faster than Jacobi updates if the update matrices are entry-wise non-negative.

Let us formally define Jacobi and GS updates. Suppose Jacobi updates take the form

$$x^{(t)} = T_1(x^{(t-1)}, y^{(t-1)}, \ldots, x^{(t-k)}, y^{(t-k)}),\ y^{(t)} = T_2(x^{(t-1)}, y^{(t-1)}, \ldots, x^{(t-k)}, y^{(t-k)}).$$

Then Gauss–Seidel updates replace $x^{(t-i)}$ with the more recent $x^{(t-i+1)}$ in operator T_2:

$$x^{(t)} = T_1(x^{(t-1)}, y^{(t-1)}, \ldots, x^{(t-k)}, y^{(t-k)}),\ y^{(t)} = T_2(x^{(t)}, y^{(t-1)}, \ldots, x^{(t+1-k)}, y^{(t-k)}),$$

where $T_1, T_2 : \mathbb{R}^{nk} \times \mathbb{R}^{nk} \to \mathbb{R}^n$ can be any update functions. We can apply this replacement to all algorithms in Section 3.2.1. Inspired by Stein–Rosenberg theorem for element-wise

non-negative matrices, our goal is to understand the relation between Jacobi and GS updates for gradient algorithms in bilinear games. Note that all gradient algorithms in Section 3.2.1 can be written as linear dynamical systems (Varga, 1962).

We find a nice relation between the characteristic polynomials of Jacobi and GS updates in Theorem 3.2.1, which turns out to greatly simplify our subsequent analyses:

Theorem 3.2.1 (Jacobi vs. Gauss–Seidel). *Let* $p(\lambda, \gamma) = \det(\sum_{i=0}^{k}(\gamma L_i + U_i)\lambda^{k-i})$, *where* $A_i = L_i + U_i$ *and* L_i *is strictly lower block triangular. Then, the characteristic polynomial of Jacobi updates is* $p(\lambda, 1)$ *while that of Gauss–Seidel updates is* $p(\lambda, \lambda)$.

Proof. Let us first consider the *block* linear iterative process in the sense of Jacobi (i.e., all blocks are updated *simultaneously*):

$$z^{(t)} = \begin{bmatrix} z_1^{(t)} \\ \vdots \\ z_b^{(t)} \end{bmatrix} = \sum_{i=1}^{k} A_i \begin{bmatrix} z_1^{(t-i)} \\ \vdots \\ z_b^{(t-i)} \end{bmatrix} = \sum_{i=1}^{k} \left[\sum_{j=1}^{l-1} A_{i,j} z_j^{(t-i)} + \sum_{j=l}^{b} A_{i,j} z_j^{(t-i)} \right] + d, \tag{3.29}$$

where $A_{i,j}$ is the j-th column block of A_i. For each matrix A_i, we decompose it into the sum

$$A_i = L_i + U_i, \tag{3.30}$$

where L_i is the strictly lower *block* triangular part and U_i is the upper (including diagonal) *block* triangular part. Theorem 3.1.1 indicates that the convergence behaviour of (3.29) is governed by the largest modulus of the roots of the characteristic polynomial:

$$\det\left(-\lambda^k I + \sum_{i=1}^{k} A_i \lambda^{k-i}\right) = \det\left(-\lambda^k I + \sum_{i=1}^{k} (L_i + U_i)\lambda^{k-i}\right). \tag{3.31}$$

Alternatively, we can also consider the updates in the sense of Gauss–Seidel (i.e., blocks are updated *sequentially*):

$$z_l^{(t)} = \sum_{i=1}^{k} \left[\sum_{j=1}^{l-1} A_{i,j} z_j^{(t-i+1)} + \sum_{j=l}^{b} A_{i,j} z_j^{(t-i)} \right]_l + d_l, \quad l = 1, \ldots, b. \tag{3.32}$$

We can rewrite the Gauss–Seidel update elegantly[3] as:

$$(I - L_1) z^{(t)} = \sum_{i=1}^{k} (L_{i+1} + U_i) z^{(t-i)} + d, \tag{3.33}$$

[3] This is well-known when $k = 1$, see e.g. Saad (2003).

i.e.,

$$z^{(t)} = \sum_{i=1}^{k} (I - L_1)^{-1}(L_{i+1} + U_i) z^{(t-i)} + (I - L_1)^{-1} d, \tag{3.34}$$

where $L_{k+1} := \mathbf{0}$. Applying Theorem 3.1.1 again we know the convergence behaviour of the Gauss–Seidel update is governed by the largest modulus of roots of the characteristic polynomial:

$$\det\left(-\lambda^k I + \sum_{i=1}^{k} (I - L_1)^{-1}(L_{i+1} + U_i)\lambda^{k-i} \right) \tag{3.35}$$

$$= \det\left((I - L_1)^{-1}\Big(-\lambda^k I + \lambda^k L_1 + \sum_{i=1}^{k} (L_{i+1} + U_i)\lambda^{k-i} \Big) \right) \tag{3.36}$$

$$= \det(I - L_1)^{-1} \cdot \det\left(\sum_{i=0}^{k} (\lambda L_i + U_i)\lambda^{k-i} \right) \tag{3.37}$$

Note that $A_0 = -I$ and the factor $\det(I - L_1)^{-1}$ can be discarded since multiplying a characteristic polynomial by a non-zero constant factor does not change its roots. □

Compared to the Jacobi update, in some sense the Gauss–Seidel update amounts to *shifting the strictly lower block triangular matrices L_i one step to the left*, as $p(\lambda, \lambda)$ can be rewritten as $\det\left(\sum_{i=0}^{k} (L_{i+1} + U_i)\lambda^{k-i}\right)$, with $L_{k+1} := \mathbf{0}$. This observation will significantly simplify our comparison between Jacobi and Gauss–Seidel updates.

3.2.3 Stability Analysis of Gradient Algorithms

We are now ready to compare Jacobi and Gauss–Seidel updates for gradient algorithms. The following lemma is well-known and easy to verify using Schur's complement:

Lemma 3.2.2. *Given $M \in \mathbb{R}^{2n \times 2n}$, $A \in \mathbb{R}^{n \times n}$ and*

$$M = \begin{bmatrix} A & B \\ C & D \end{bmatrix}. \tag{3.38}$$

If C and D commute, then we have $\det M = \det(AD - BC)$.

We formulate necessary and sufficient conditions under which a gradient-based algorithm converges for bilinear games (3.18). We use "J" as a shorthand for Jacobi style updates and "GS" for Gauss–Seidel style updates. For each algorithm, we first write down the characteristic polynomials for both Jacobi and GS updates, and present the exact conditions for convergence. We used the term "convergence region" to denote a subset of the parameter space (with parameters α, β or γ) where the algorithm converges. We show that in many cases the GS convergence regions strictly include the Jacobi convergence regions. Our result shares similarity with the celebrated Stein–Rosenberg theorem (Stein and Rosenberg, 1948), which only applies to solving linear systems with non-negative matrices. In this sense, our results extend the Stein–Rosenberg theorem to cover nontrivial bilinear games.

Gradient descent ascent (GDA) From (3.21) the update equation of Jacobi GDA can be derived as:

$$z^{(t+1)} = \begin{bmatrix} I & -\alpha_1 E \\ \alpha_2 E^\top & I \end{bmatrix} z^{(t)}, \tag{3.39}$$

and with Lemma 3.2.2, we compute the characteristic polynomial as in eq. (3.2):

$$\det \begin{bmatrix} (\lambda-1)I & \alpha_1 E \\ -\alpha_2 E^\top & (\lambda-1)I \end{bmatrix} = \det[(\lambda-1)^2 I + \alpha_1\alpha_2 EE^\top], \tag{3.40}$$

With spectral decomposition we obtain (3.41). Taking $\alpha_2 \to \lambda\alpha_2$ and with Theorem 3.2.1 we obtain the corresponding GS updates. Therefore, the characteristic polynomials for GDA are:

$$\text{J: } (\lambda-1)^2 + \alpha_1\alpha_2\sigma^2 = 0, \text{ GS: } (\lambda-1)^2 + \alpha_1\alpha_2\sigma^2\lambda = 0. \tag{3.41}$$

Scaling symmetry From Section 3.2.3 we obtain a scaling symmetry

$$(\alpha_1, \alpha_2) \to (t\alpha_1, \alpha_2/t),$$

with $t > 0$. With this symmetry we can always fix $\alpha_1 = \alpha_2 = \alpha$. This symmetry also holds for EG and momentum. For OGD, the scaling symmetry is slightly different with $(\alpha_1, \beta_1, \alpha_2, \beta_2) \to (t\alpha_1, t\beta_1, \alpha_2/t, \beta_2/t)$, but we can still use this symmetry to fix $\alpha_1 = \alpha_2 = \alpha$.

Theorem 3.2.3 (GDA). *Jacobi GDA and Gauss–Seidel GDA do not converge if the initialization is not a saddle point. However, Gauss–Seidel GDA can have a limit cycle while Jacobi GDA always diverges.*

Proof. With the notations in Corollary 3.1.3, for Jacobi GDA, we have $b = 1 + \alpha^2\sigma^2 > 1$. For Gauss–Seidel GDA, we have $b = 1$. The Schur conditions are violated. □

In the constrained case, Mertikopoulos et al. (2018a) and Bailey and Piliouras (2018) show that Follow-The-Regularized-Leader, a more generalized algorithm of GDA, does not converge for polymatrix games. When $\alpha_1 = \alpha_2$, the result of Gauss–Seidel GDA has been shown in Bailey et al. (2019).

Extra-gradient From eq. (3.22) and eq. (3.23), the update of Jacobi EG is:

$$z^{(t+1)} = \begin{bmatrix} I - \beta_2 EE^\top & -\alpha_1 E \\ \alpha_2 E^\top & I - \beta_1 E^\top E \end{bmatrix} z^{(t)}, \tag{3.42}$$

and the characteristic polynomial is:

$$\det \begin{bmatrix} (\lambda - 1)I + \beta_2 EE^\top & \alpha_1 E \\ -\alpha_2 E^\top & (\lambda - 1)I + \beta_1 E^\top E \end{bmatrix}. \tag{3.43}$$

Since we assumed $\alpha_2 > 0$, we can left multiply the second row by $\beta_2 E/\alpha_2$ and add it to the first row. Hence, we obtain:

$$\det \begin{bmatrix} (\lambda - 1)I & \alpha_1 E + (\lambda - 1)\beta_2 E/\alpha_2 + \beta_1\beta_2 EE^\top E/\alpha_2 \\ -\alpha_2 E^\top & (\lambda - 1)I + \beta_1 E^\top E \end{bmatrix}. \tag{3.44}$$

With Lemma 3.2.2 the equation above becomes:

$$\det[(\lambda - 1)^2 I + (\beta_1 + \beta_2)E^\top E(\lambda - 1) + (\alpha_1\alpha_2 E^\top E + \beta_1\beta_2 E^\top EE^\top E)], \tag{3.45}$$

which simplifies to (3.46) with spectral decomposition. Note that to obtain the GS polynomial, we simply take $\alpha_2 \to \lambda\alpha_2$ in the Jacobi polynomial as shown in Theorem 3.2.1. For the ease of reading we copy the characteristic equations for generalized EG:

$$\text{J: } (\lambda - 1)^2 + (\beta_1 + \beta_2)\sigma^2(\lambda - 1) + (\alpha_1\alpha_2\sigma^2 + \beta_1\beta_2\sigma^4) = 0, \tag{3.46}$$

$$\text{GS: } (\lambda - 1)^2 + (\alpha_1\alpha_2 + \beta_1 + \beta_2)\sigma^2(\lambda - 1) + (\alpha_1\alpha_2\sigma^2 + \beta_1\beta_2\sigma^4) = 0. \tag{3.47}$$

Theorem 3.2.4 (EG). *For generalized EG with $\alpha_1 = \alpha_2 = \alpha$ and $\gamma_i = \beta_i/\alpha$, Jacobi and Gauss–Seidel updates achieve linear convergence iff for any singular value σ of E, we have:*

$$\begin{aligned} &\text{J} : |\beta_1\sigma^2 + \beta_2\sigma^2 - 2| < 1 + (1 - \beta_1\sigma^2)(1 - \beta_2\sigma^2) + \alpha^2\sigma^2, \\ &(1 - \beta_1\sigma^2)(1 - \beta_2\sigma^2) + \alpha^2\sigma^2 < 1, \end{aligned} \tag{3.48}$$

$$\begin{aligned} &\text{GS} : |(\beta_1 + \beta_2 + \alpha^2)\sigma^2 - 2| < 1 + (1 - \beta_1\sigma^2)(1 - \beta_2\sigma^2), \\ &(1 - \beta_1\sigma^2)(1 - \beta_2\sigma^2) < 1. \end{aligned} \tag{3.49}$$

*If $\beta_1 + \beta_2 + \alpha^2 < 2/\sigma_1^2$, the convergence region of GS updates **strictly** includes that of Jacobi updates.*

Proof. Both characteristic polynomials can be written as a quadratic polynomial $\lambda^2 + a\lambda + b$, where:

$$\text{J: } a = (\beta_1 + \beta_2)\sigma^2 - 2,\ b = (1 - \beta_1\sigma^2)(1 - \beta_2\sigma^2) + \alpha^2\sigma^2, \tag{3.50}$$

$$\text{GS: } a = (\beta_1 + \beta_2 + \alpha^2)\sigma^2 - 2,\ b = (1 - \beta_1\sigma^2)(1 - \beta_2\sigma^2). \tag{3.51}$$

Compared to Jacobi EG, the only difference between Gauss–Seidel and Jacobi updates is that the $\alpha^2\sigma^2$ in b is now in a, which agrees with Theorem 3.2.1. Using Corollary 3.1.3, we can derive the Schur conditions (3.48) and (3.49).

More can be said if $\beta_1 + \beta_2$ is small. For instance, if $\beta_1 + \beta_2 + \alpha^2 < 2/\sigma_1^2$, then (3.48) implies (3.49). In this case, the first conditions of (3.48) and (3.49) are equivalent, while the second condition of (3.48) strictly implies that of (3.49). Hence, the Schur region of Gauss–Seidel updates includes that of Jacobi updates. The same holds true if $\beta_1 + \beta_2 < \frac{4}{3\sigma_1^2}$.

More precisely, to show that the GS convergence region strictly contains that of the Jacobi convergence region, simply take $\beta_1 = \beta_2 = \beta$. The Schur condition for Jacobi EG and Gauss–Seidel EG are separately:

$$\text{J: } \alpha^2\sigma^2 + (\beta\sigma^2 - 1)^2 < 1, \tag{3.52}$$

$$\text{GS: } 0 < \beta\sigma^2 < 2 \text{ and } |\alpha\sigma| < 2 - \beta\sigma^2. \tag{3.53}$$

It can be shown that if $\beta = \alpha^2/3$ and $\alpha \to 0$, (3.52) is always violated whereas (3.53) is always satisfied.

Conversely, we give an example when Jacobi EG converges while GS EG does not. Let $\beta_1\sigma^2 = \beta_2\sigma^2 \equiv \frac{3}{2}$, then Jacobi EG converges iff $\alpha^2\sigma^2 < \frac{3}{4}$ while GS EG converges iff $\alpha^2\sigma^2 < \frac{1}{4}$. □

Optimistic gradient descent We can compute the LDS for OGD with eq. (3.24) and eq. (3.25):

$$z^{(t+2)} = \begin{bmatrix} I & -\alpha_1 E \\ \alpha_2 E^\top & I \end{bmatrix} z^{(t+1)} + \begin{bmatrix} \mathbf{0} & \beta_1 E \\ -\beta_2 E^\top & \mathbf{0} \end{bmatrix} z^{(t)}, \tag{3.54}$$

With eq. (3.2), the characteristic polynomial for Jacobi OGD is

$$\det \begin{bmatrix} (\lambda^2 - \lambda) I & (\lambda\alpha_1 - \beta_1) E \\ (-\lambda\alpha_2 + \beta_2) E^\top & (\lambda^2 - \lambda) I \end{bmatrix}. \tag{3.55}$$

Taking the determinant and with Lemma 3.2.2 we obtain (3.56). The characteristic polynomial for GS updates in (3.57) can be subsequently derived with Theorem 3.2.1, by taking $(\alpha_2, \beta_2) \to (\lambda\alpha_2, \lambda\beta_2)$. The characteristic equations can be computed as:

$$\text{J: } \lambda^2(\lambda - 1)^2 + (\lambda\alpha_1 - \beta_1)(\lambda\alpha_2 - \beta_2)\sigma^2 = 0, \tag{3.56}$$

$$\text{GS: } \lambda^2(\lambda - 1)^2 + (\lambda\alpha_1 - \beta_1)(\lambda\alpha_2 - \beta_2)\lambda\sigma^2 = 0. \tag{3.57}$$

Using the characteristic polynomials and Corollary 3.1.3, we obtain the following theorem, with the detailed proof in Appendix B.1.

Theorem 3.2.5 (OGD). *For generalized OGD with $\alpha_1 = \alpha_2 = \alpha$, Jacobi and Gauss–Seidel updates achieve linear convergence iff for any singular value σ of E, we have:*

$$\text{J}: \begin{cases} |\beta_1\beta_2\sigma^2| < 1,\ (\alpha - \beta_1)(\alpha - \beta_2) > 0,\ 4 + (\alpha + \beta_1)(\alpha + \beta_2)\sigma^2 > 0, \\ \alpha^2\left(\beta_1^2\sigma^2 + 1\right)\left(\beta_2^2\sigma^2 + 1\right) < (\beta_1\beta_2\sigma^2 + 1)(2\alpha(\beta_1 + \beta_2) + \beta_1\beta_2(\beta_1\beta_2\sigma^2 - 3)); \end{cases} \tag{3.58}$$

$$\text{GS}: \begin{cases} (\alpha - \beta_1)(\alpha - \beta_2) > 0, (\alpha + \beta_1)(\alpha + \beta_2)\sigma^2 < 4, \\ (\alpha\beta_1\sigma^2 + 1)(\alpha\beta_2\sigma^2 + 1) > (1 + \beta_1\beta_2\sigma^2)^2. \end{cases} \tag{3.59}$$

*The convergence region of GS updates **strictly** includes that of Jacobi updates.*

Momentum method With eq. (3.26) and eq. (3.27), the LDS for the momentum method is:

$$z^{(t+2)} = \begin{bmatrix} (1 + \beta_1) I & -\alpha_1 E \\ \alpha_2 E^\top & (1 + \beta_2) I \end{bmatrix} z^{(t+1)} + \begin{bmatrix} -\beta_1 I & \mathbf{0} \\ \mathbf{0} & -\beta_2 I \end{bmatrix} z^{(t)}, \tag{3.60}$$

From eq. (3.2), the characteristic polynomial for Jacobi momentum is

$$\det \begin{bmatrix} (\lambda^2 - \lambda(1 + \beta_1) + \beta_1) I & \lambda\alpha_1 E \\ -\lambda\alpha_2 E^\top & (\lambda^2 - \lambda(1 + \beta_2) + \beta_2) I \end{bmatrix}. \tag{3.61}$$

Taking the determinant and with Lemma 3.2.2 we obtain (3.62), while (3.63) can be derived with Theorem 3.2.1, by taking $\alpha_2 \to \lambda\alpha_2$. For the ease of reading we copy the characteristic polynomials from the main text as:

$$\text{J: } (\lambda-1)^2(\lambda-\beta_1)(\lambda-\beta_2)+\alpha_1\alpha_2\sigma^2\lambda^2=0, \tag{3.62}$$

$$\text{GS: } (\lambda-1)^2(\lambda-\beta_1)(\lambda-\beta_2)+\alpha_1\alpha_2\sigma^2\lambda^3=0. \tag{3.63}$$

Using the characteristic polynomials and Corollary 3.1.3, we obtain the following theorem, with the detailed proof in Appendix B.1.

Theorem 3.2.6 (momentum). *For the generalized momentum method with $\alpha_1 = \alpha_2 = \alpha$, the Jacobi updates never converge, while the GS updates converge iff for any singular value σ of E, we have:*

$$\begin{aligned}&|\beta_1\beta_2|<1, |-\alpha^2\sigma^2+\beta_1+\beta_2+2|<\beta_1\beta_2+3,\ 4(\beta_1+1)(\beta_2+1)>\alpha^2\sigma^2,\\ &\alpha^2\sigma^2\beta_1\beta_2<(1-\beta_1\beta_2)(2\beta_1\beta_2-\beta_1-\beta_2).\end{aligned} \tag{3.64}$$

*This condition implies that at least one of β_1, β_2 is **negative**.*

Prior to this work, only sufficient conditions for linear convergence were given for the usual EG and OGD. For the momentum method, our result improves upon Gidel et al. (2019b) where they only considered specific cases of parameters. For example, they only considered $\beta_1 = \beta_2 \geq -1/16$ for Jacobi momentum (but with explicit rate of divergence), and $\beta_1 = -1/2$, $\beta_2 = 0$ for GS momentum (with convergence rate). Our Theorem 3.2.6 gives a more complete picture and formally justifies the necessity of negative momentum.

The stability regions of EG, OGD and the momentum method can be visualized in Figure 3.1. In this figure, we take $\alpha_1 = \alpha_2 = \alpha$ and $\beta_1 = \beta_2 = \beta$ in these algorithms. It can be seen that the stability regions for Gauss–Seidel updates are generally larger than Jacobi updates, and thus verifies the statement that Gauss–Seidel updates are more stable.

3.3 General Local Stability

From Theorem 2.3.1 we know that local minimax points are fixed points of gradient algorithms. In this section, we extend our stability analysis in Section 3.2 and study the following problem:

$$\min_{x\in\mathbb{R}^n}\max_{y\in\mathbb{R}^m} f(x,y), \tag{3.65}$$

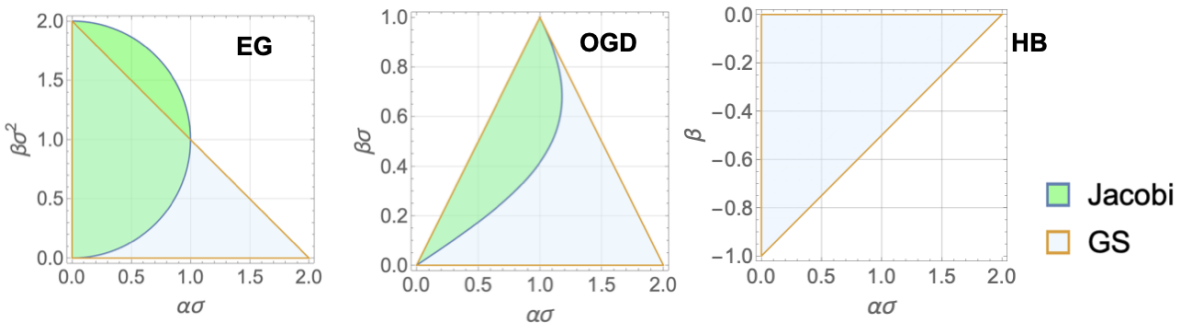


Figure 3.1: Stability regions of Extra-gradient (EG), Optimistic gradient descent (OGD) and the Heavy ball method (HB). We take $\alpha_1 = \alpha_2 = \alpha$, $\beta_1 = \beta_2 = \beta$ for illustration purpose.

with f twice continuous differentiable. We use z_t instead of $z^{(t)}$ in this section for the iterative updates.

We focus on *local linear convergence* at stationary points z^* using spectral analysis, by which we mean that when initialized in a neighborhood of z^*, an iterative method would give:

$$\|z_{t+1} - z^*\| \leq r\|z_t - z^*\|, \tag{3.66}$$

where $0 \leq r < 1$. Spectral analysis of a matrix A mainly involves two types of quantities: the spectrum of A, $\mathrm{Sp}(A) := \{\lambda : \lambda \text{ is an eigenvalue of } A\}$, as well as the spectral radius, $\rho(A) := \max_{\lambda \in \mathrm{Sp}(A)} |\lambda|$. An algorithm is *exponentially stable* if the spectral radius of its Jacobian matrix is less than one, which guarantees local linear convergence (Polyak, 1987). A more rigorous definition uses the Hartman–Grobman theorem (Katok and Hasselblatt, 1995). Below when we refer to convergence, we always mean local linear convergence.

To obtain convergence near local minimax points, we consider two-time-scale (2TS)[4] gradient algorithms, as proposed in Heusel et al. (2017) to train GANs. Also, Jin et al. (2020) proved the "equivalence" between the stable points of 2TS-GDA and strict local minimax points (see Corollary 2.3.14). The intuition is that 2TS algorithms help the convergence by taking a much larger step w.r.t. the maximization variable y. We denote $z_t = (x_t, y_t)$ and define the vector field for the gradient update

$$v(z) = (-\alpha_1 \partial_x f(z), \alpha_2 \partial_y f(z)).$$

[4]This terminology comes from analogy with the continuous training dynamics. In our paper we simply mean choosing two different step sizes.

Local stability results can be obtained by analyzing the Jacobian of $v(z)$ at a stationary point (x^*, y^*):

$$H_{\alpha_1,\alpha_2} = H_{\alpha_1,\alpha_2}(f) := \begin{bmatrix} -\alpha_1 \partial^2_{xx} f & -\alpha_1 \partial^2_{xy} f \\ \alpha_2 \partial^2_{yx} f & \alpha_2 \partial^2_{yy} f \end{bmatrix}. \tag{3.67}$$

Define $\alpha_2 = \gamma \alpha_1$, and $H_{\alpha_1,\alpha_2} = \alpha_1 H_{1,\gamma}$. Note that $H_{\alpha_1,\alpha_2}(f)$ may not be symmetric, hence its spectrum lies on the complex plane. We also define $H := H_{\alpha,\alpha}/\alpha$ which is independent of α. To characterize the stable set of an algorithm, we ask the following question:

> Given hyper-parameters $\{\mu_i\}_{i=0}^k$ (e.g. step size, momentum coefficient) of an algorithm A, what exactly are the geometric conclusions on the spectrum of H_{α_1,α_2} such that A is exponentially stable at z^*?

Similar questions have been asked in Niethammer and Varga (1983) for problems of linear equations, where the Jacobian is a constant matrix. Such geometric characterizations allow us to analyze the convergence near local saddle, local minimax and local robust points.

Note that in this section we are mostly considering one type of algorithmic modification in sequential games using two-time-scale (except in Prop. 3.4.5) and simultaneous updates. For non-convex sequential smooth games, it is possible to use alternating updates in algorithms as studied in the previous section for bilinear games.

3.3.1 Stable Sets of Extra-gradient (EG) and Optimistic Gradient Descent (OGD)

We consider the generalized extra-gradient method EG$(\alpha_1, \alpha_2, \beta)$ (Korpelevich, 1976) (the original version has $\beta = 1$):

$$z_{t+1} = z_t + v(z_{t+1/2})/\beta,\ z_{t+1/2} = z_t + v(z_t). \tag{3.68}$$

and the generalized optimistic gradient descent (Peng et al., 2020), which we denote as OGD(k, α_1, α_2)):

$$z_{t+1} = z_t + kv(z_t) - v(z_{t-1}). \tag{3.69}$$

EG has recently been studied in e.g. (Mertikopoulos et al., 2019) for special NCNC games, and in Azizian et al. (2020a,b) for convex-concave settings using spectral analysis. OGD was originally proposed in Popov (1980) as the past extra-gradient method, and recently studied in the GAN literature (e.g. Daskalakis et al., 2018). We show a close connection between EG and OGD, as observed recently (Hsieh et al., 2019; Mokhtari et al., 2019):

Lemma 3.3.1 (**equivalence between past extra-gradient and OGD**). *The past extra-gradient method*

$$z_{t+1} = z_t + v(z_{t+1/2})/\beta,\ z_{t+1/2} = z_t + v(z_{t-1/2}) \tag{3.70}$$

can be rewritten as $z'_{t+1} = z'_t + kv(z'_t) - v(z'_{t-1})$ *with* $k = 1 + 1/\beta$ *and* $z'_t = z_{t-1/2}$.

Proof. From the second equation of (3.70) we obtain

$$\begin{aligned} z_{t+3/2} &= z_{t+1} + v(z_{t+1/2}) \\ &= z_t + \left(1 + \frac{1}{\beta}\right) v(z_{t+1/2}) + v(z_{t-1/2}) - v(z_{t-1/2}) \\ &= z_{t+1/2} + \left(1 + \frac{1}{\beta}\right) v(z_{t+1/2}) - v(z_{t-1/2}). \end{aligned} \tag{3.71}$$

In the second line we used the first equation of (3.70) and in the third line we used the second equation of (3.70). □

Due to this correspondence, we will only consider OGD with $k > 1$. We now characterize the stable sets of EG and OGD, or the *necessary and sufficient conditions* for local convergence:

Theorem 3.3.2 (**stability of EG/OGD**). *At* (x^*, y^*), *EG*$(\alpha_1, \alpha_2, \beta)$ *is exponentially stable iff for any* $\lambda \in Sp(H_{\alpha_1,\alpha_2})$, $|1 + \lambda/\beta + \lambda^2/\beta| < 1$. OGD$(k, \alpha_1, \alpha_2)$ *is exponentially stable iff for any* $\lambda \in Sp(H_{\alpha_1,\alpha_2})$, $|\lambda| < 1$ *and* $|\lambda|^2(k - 3 + (k+1)|\lambda|^2) < 2\Re(\lambda)(k|\lambda|^2 - 1)$.

Proof. From (3.68) the update of EG can be rewritten as $z_{t+1} = z_t + v(z_t + v(z_t))/\beta$. We compute the Jacobian matrix of this update:

$$J = J(f) = I + H_{\alpha_1,\alpha_2}/\beta + H^2_{\alpha_1,\alpha_2}/\beta.$$

It then follows that $\mathrm{Sp}(J) = 1 + \mathrm{Sp}(H_{\alpha_1,\alpha_2})/\beta + \mathrm{Sp}(H_{\alpha_1,\alpha_2})^2/\beta$, where the operation is element-wise. Therefore, $\rho(J(f)) < 1$ iff

$$\max_{\lambda \in \mathrm{Sp}(H_{\alpha_1,\alpha_2})} |1 + \lambda/\beta + \lambda^2/\beta| < 1.$$

Similarly for OGD, the spectrum can be computed as:

$$\mathrm{Sp}(J_{\mathrm{OGD}}) \quad = \quad \{x : p(x) := x^2 - (1 + k\lambda)x + \lambda = 0,\ \lambda \in H_{\alpha_1,\alpha_2}\}. \tag{3.72}$$

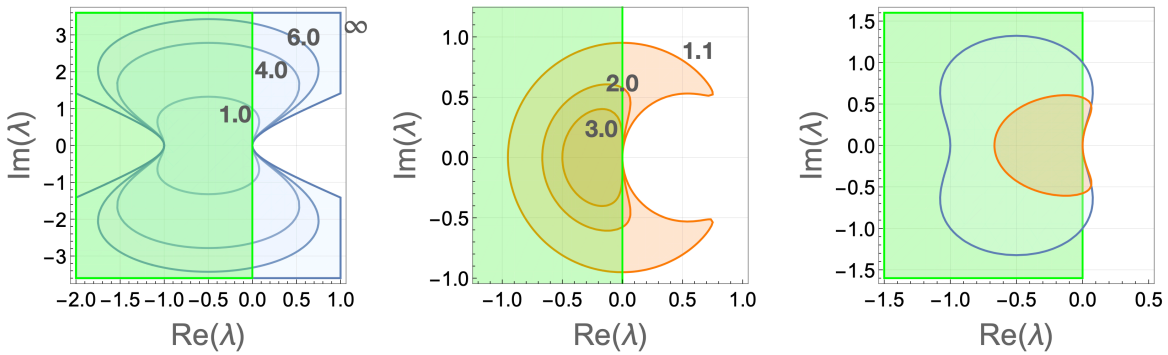


Figure 3.2: The blue region is where EG/OGD is exponentially stable. The green region represents where the eigenvalues of $\mathrm{Sp}(H_{\alpha_1,\alpha_2})$ at local saddle points may occur (Section 3.4.1). (**left**) $\mathrm{EG}(\alpha_1, \alpha_2, \beta)$ with $\beta \in \{1.0, 4.0, 6.0, \infty\}$; (**middle**) $\mathrm{OGD}(k, \alpha_1, \alpha_2)$ with $k \in \{1.1, 2.0, 3.0\}$. (**right**) Comparison between $\mathrm{EG}(\alpha_1, \alpha_2, 1)$ (**blue**) and $\mathrm{OGD}(2, \alpha_1, \alpha_2)$ (**yellow**).

With Corollary 3.1.4, we obtain the necessary and sufficient conditions when the roots of $p(x)$ are in the unit circle:

$$|\lambda| < 1,\ (k-1)|\lambda|^2(k-3+(k+1)|\lambda|^2) < 2(k-1)\Re(\lambda)(k|\lambda|^2-1),\ \forall \lambda \in H_{\alpha_1,\alpha_2}.$$

□

From this theorem, we can plot the stable region of EG and OGD with the original parameters, and find that EG and OGD are indeed similar, as shown on the right of Figure 3.2. For EG, we note that Azizian et al. (2020b) used the spectral shapes of the support of $\mathrm{Sp}(H_{\alpha_1,\alpha_2})$ to give upper and lower bounds of the convergence rates of EG, but our results are orthogonal to it since we do not assume a geometric shape of the support of $\mathrm{Sp}(H_{\alpha_1,\alpha_2})$.

When $\beta \to \infty$, $k \to 1_+$, and the step size of extra-step is much larger than the step size of the gradient step. In Zhang and Yu (2020) it was found that for bilinear games, taking $\beta \to \infty$ gives the best convergence rate among all hyperparameters. We show that larger β increases the local stability as well (see also Hsieh et al. (2020) for saddle point problems):

Theorem 3.3.3 (**more aggressive extra-gradient steps, more stable**). *For $\beta_1 > \beta_2 > 1$, whenever $\mathrm{EG}(\alpha_1, \alpha_2, \beta_2)$ is exponentially stable at (x^*, y^*), $\mathrm{EG}(\alpha_1, \alpha_2, \beta_1)$ is exponentially stable at (x^*, y^*) as well. For $k_1 > k_2 > 1$, whenever $OGD(k_1, \alpha_1, \alpha_2)$ is exponentially stable at (x^*, y^*), $OGD(k_2, \alpha_1, \alpha_2)$ is exponentially stable at (x^*, y^*) as well.*

Proof. Rewriting $\lambda = x + iy$ with $x, y \in \mathbb{R}$ for $\lambda \in H_{\alpha_1,\alpha_2}$ and using Theorem 3.3.2, we run the following Mathematica code ($b_1 \equiv \beta_1$, $b_2 \equiv \beta_2$):

```
Reduce[ForAll[{x, y, b1, b2}, ((y + 2 x y)/b2)^2 +
  (1 + (x + x^2 - y^2)/b2)^2 < 1 && b1 > b2 > 1,
  ((y + 2 x y)/b1)^2 + (1 + (x + x^2 - y^2)/b1)^2 < 1]]
```

The answer is `True`. For the second part, we rewrite the stability condition for OGD as:

$$k|\lambda|^2(1 + |\lambda|^2 - 2\Re(\lambda)) < 3|\lambda|^2 - |\lambda|^4 - 2\Re(\lambda). \tag{3.73}$$

Since $\Re(\lambda) \leq |\lambda|$, $1 + |\lambda|^2 - 2\Re(\lambda) \geq 0$. The left hand side increases with k. □

In the limit when $\beta \to \infty$, the stable region is $\Re(\lambda + \lambda^2) < 0$ whose boundary is a hyperbola. Similarly, when $k \to 1_+$, OGD has the largest convergence region: $\{\lambda \in \mathbb{C} : |\lambda| < 1, |\lambda - 1/2| > 1/2\}$. Figure 3.2 gives a visualization for the stable sets of EG/OGD. Their convergence regions strictly include that of GDA, and thus these algorithms are more stable:

Corollary 3.3.4. *Given $|\lambda| < 1$ with $\lambda \in H_{\alpha_1,\alpha_2}$, whenever GDA$(\alpha_1, \alpha_2)$ converges, EG$(\alpha_1, \alpha_2, 1)$ converges as well. Given $|\lambda| < 1/\sqrt{3}$ with $\lambda \in H_{\alpha_1,\alpha_2}$, whenever GDA$(\alpha_1, \alpha_2)$ converges, OGD$(2, \alpha_1, \alpha_2)$ converges.*

Proof. When $\beta = 0$, (3.80) becomes $|1 + \lambda| < 1$. The first part follows from:

$$|1 + \lambda| < 1 \text{ and } |\lambda| < 1 \Longrightarrow |1 + \lambda + \lambda^2| < 1. \tag{3.74}$$

Taking $k = 2$, from Theorem 3.3.2, the stability condition for OGD is:

$$|\lambda|^2(-1 + 3|\lambda|^2) < 2\Re(\lambda)(2|\lambda|^2 - 1). \tag{3.75}$$

We want to show that for all $|1 + \lambda| < 1$ and $|\lambda| < 1/\sqrt{3}$, (3.75) holds, and thus we define $\lambda = u + iv$ ($u, v \in \mathbb{R}$) and use the following Mathematica code:

```
Reduce[ForAll[{u, v},  (1 + u)^2 + v^2 < 1 &&  u^2 + v^2 < 1/3,
(u^2 + v^2) (-1 + 3 (u^2 + v^2)) < 2 u (-1 + 2 (u^2 + v^2))]]
```

This result is `True`. □

3.3.2 Momentum Algorithms

We study the effect of momentum for convergence to local saddle points, including heavy ball (Polyak, 1964) and Nesterov's momentum (Nesterov, 1983). They are similar to GDA and do not converge even for bilinear games, as proved in Theorem 3.2.6. In the following two subsections, we study the effect of momentum for convergence to local saddle points. GDA is a special case if we take the momentum parameter $\beta = 0$.

Heavy Ball (HB)

We study the heavy ball method $\text{HB}(\alpha_1, \alpha_2, \beta)$ (Polyak, 1964) in the context of minimax optimization, as also studied in Gidel et al. (2019b):

$$z_{t+1} = z_t + v(z_t) + \beta(z_t - z_{t-1}), v(z) = (-\alpha_1 \partial_x f(z), \alpha_2 \partial_y f(z)). \tag{3.76}$$

Theorem 3.3.5 (HB). $\text{HB}(\alpha_1, \alpha_2, \beta)$ *is exponentially stable iff* $\forall\, \lambda \in Sp(H_{\alpha_1,\alpha_2})$, $|\beta| < 1$,

$$2\beta\Re(\lambda^2) - 2(1-\beta)^2(1+\beta)\Re(\lambda) > (1+\beta^2)|\lambda|^2.$$

Proof. With state augmentation $z_t \to (z_{t+1}, z_t)$, the Jacobian for $\text{HB}(\alpha_1, \alpha_2, \beta)$ is:

$$J_{\text{HB}}(f) = \begin{bmatrix} (1+\beta)I_{n+m} + H_{\alpha_1,\alpha_2} & -\beta I_{n+m} \\ I_{n+m} & \mathbf{0} \end{bmatrix}, \tag{3.77}$$

The spectrum can be computed as:

$$\text{Sp}(J_{\text{HB}}(f)) = \{w : p(w) := (w-1)(w-\beta) - w\lambda = 0, \lambda \in H_{\alpha_1,\alpha_2}\}. \tag{3.78}$$

This quadratic equation can be further expanded as:

$$w^2 - (\beta + 1 + \lambda)w + \beta = 0. \tag{3.79}$$

With Corollary 3.1.4, we obtain the necessary and sufficient conditions for which all the roots are within a unit disk:

$$|\beta| < 1, 2\beta\Re(\lambda^2) - 2(1-\beta)^2(1+\beta)\Re(\lambda) > (1+\beta^2)|\lambda|^2. \tag{3.80}$$

□

This theorem can also be derived from Euler transform as in (Niethammer and Varga, 1983, Section 6) which is used in analyzing methods for solving linear equations. The first inequality $|\beta| < 1$ can be easily used to guide hyper-parameter tuning in practice. The second condition in fact describes an ellipsoid centered at $(-\beta - 1, 0)$. If we define $\lambda = u + iv$ and $(u, v) \in \mathbb{R}^2$, then this condition can be simplified as:

$$\frac{(u + \beta + 1)^2}{(\beta + 1)^2} + \frac{v^2}{(\beta - 1)^2} < 1. \tag{3.81}$$

As shown on the left of Figure 3.3, if the momentum factor β is positive, the ellipsoid is elongated in the horizontal direction; otherwise, it is elongated in the vertical direction. This agrees with existing results on negative momentum (Gidel et al., 2019b) and Theorem 3.2.6 on bilinear games.

Corollary 3.3.6 (HB). *For any $|\beta| < 1$, $HB(\alpha, \alpha, \beta)$ is exponentially stable for small enough α at a local saddle point iff at such a point $\Re(\lambda) \neq 0$ for all $\lambda \in Sp(H)$.*

Proof. From Lemma 3.4.1, for any $\lambda \in \mathrm{Sp}(H)$, $\Re(\lambda) \leq 0$. If $\Re(\lambda) \neq 0$ for all $\lambda \in \mathrm{Sp}(H)$, then (3.81) holds for small enough α. If $\Re(\lambda) = 0$ for some $\lambda \in \mathrm{Sp}(H)$, we cannot have (3.81). □

Nesterov's Accelerated Gradient (NAG)

Nesterov's accelerated gradient (Nesterov, 1983) is a variant of Polyak's heavy ball, which achieves the optimal convergence rate for convex functions. It has been widely applied in deep learning (Sutskever et al., 2013). In Bollapragada et al. (2019), the authors analyzed the spectrum of NAG using numerical range in the context of linear regression, which is equivalent to the case when $\mathrm{Sp}(H) \subset \mathbb{R}$ (c.f. Bollapragada et al. (2019, p. 11)).

The key difference between HB and NAG is the order of momentum update and the gradient update. We study Nesterov's momentum for minimax optimization:

$$z_{t+1} = z'_t + \alpha v(z'_t),\ z'_t = z_t + \beta(z_t - z_{t-1}), \tag{3.82}$$

which we denote as $\mathrm{NAG}(\alpha_1, \alpha_2, \beta)$. We have the following stability result for NAG:

Theorem 3.3.7 (NAG). $\mathrm{NAG}(\alpha_1, \alpha_2, \beta)$ *is exponentially stable iff for any $\lambda \in Sp(H_{\alpha_1,\alpha_2})$:*

$$|1 + \lambda|^{-2} > 1 + 2\beta(\beta^2 - \beta - 1)\Re(\lambda) + \beta^2|\lambda|^2(1 + 2\beta),\ |\beta| \cdot |1 + \lambda| < 1. \tag{3.83}$$

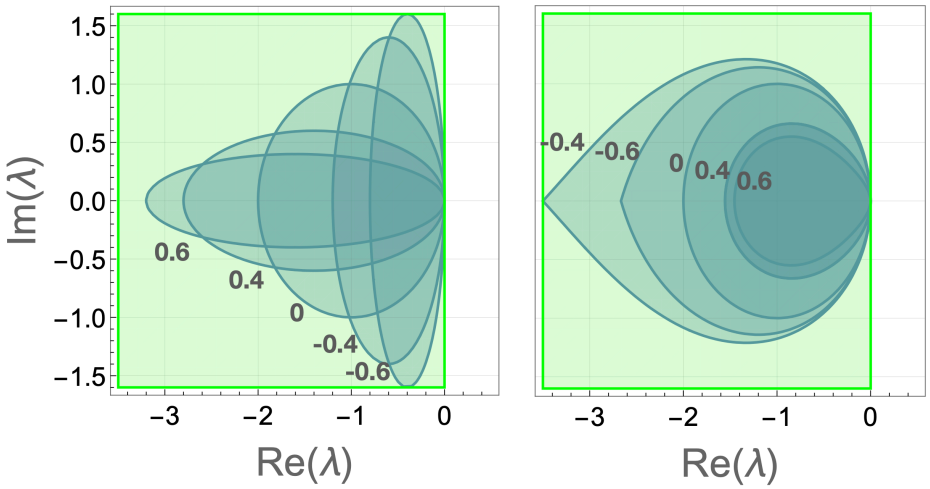


Figure 3.3: Convergence regions of momentum methods with different momentum parameter β: (**left**) HB(α, β); (**right**) NAG(α, β). We take $\beta = 0, \pm 0.4, \pm 0.6$ (as shown in the figure). The green region represents the one where the eigenvalues of $\mathrm{Sp}(H_{\alpha_1,\alpha_2})$ at local saddle points may occur (Section 3.4.1).

Proof. With state augmentation $z_t \to (z_{t+1}, z_t)$, the Jacobian for NAG is:

$$\begin{bmatrix} (1+\beta)(I_{n+m} + H_{\alpha_1,\alpha_2}) & -\beta(I_{n+m} + H_{\alpha_1,\alpha_2}) \\ I_{n+m} & \mathbf{0} \end{bmatrix}.$$

The spectrum can be computed as:

$$\mathrm{Sp}(J(f)) \quad = \quad \{w : p(w) := w^2 - w(1+\beta)(1+\lambda) + \beta(1+\lambda) = 0, \lambda \in H_{\alpha_1,\alpha_2}\}.$$

Comparing with (3.79), we find that the two characteristic polynomials are different only by $O(\alpha\beta)$. With Lemma 3.1.4, the condition for local linear convergence is:

$$|1+\lambda|^{-2} > 1 + 2\beta(\beta^2 - \beta - 1)\Re(\lambda) + \beta^2|\lambda|^2(1+2\beta), \tag{3.84}$$

$$|\beta| \cdot |1+\lambda| < 1. \tag{3.85}$$

□

From Figure 3.3, the convergence region of NAG is better conditioned than HB. However, NAG is still similar to HB and GDA in terms of the local convergence behavior:

Corollary 3.3.8 (NAG). *If $\Re(\lambda) \geq 0$ for some $\lambda \in H_{\alpha_1,\alpha_2}$, then* NAG$(\alpha_1, \alpha_2, \beta)$ *is not exponentially stable.*

Proof. Take $\lambda \in H_{\alpha_1,\alpha_2}$ and assume $\lambda = u + iv$ with $u, v \in \mathbb{R}$. (3.83) can be translated to the following Mathematica code:

```
Reduce[b^2 ((1 + u)^2 + v^2) < 1 && ((1 + u)^2 + v^2) (1 +
2 b (b^2 - b - 1) u + b^2 (u^2 + v^2) (1 + 2 b)) < 1 && u >= 0],
```

and the result is `False`. □

According to Lemma 3.4.1, NAG$(\alpha_1, \alpha_2, \beta)$ never converges on bilinear games.

3.4 Stability at Local Optimal Solutions

After characterizing the stable sets of gradient algorithms, we move on to see the spectral behavior of local optimal points (c.f. Chapter 2). For local saddle points, the spectrum of H_{α_1,α_2} is on the left closed half plane. However, the spectrum of local minimax points can be quite arbitrary. With these results we can study how gradient algorithms (GDA with momentum, EG/OGD) converge to local optimal points.

3.4.1 Local Saddle Points

Even though $H_{\alpha_1,\alpha_2}(f)$ is non-symmetric, it is still negative semi-definite near local saddle points[5]. Therefore, we can prove that its spectrum lies on the left (closed) complex plane:

Lemma 3.4.1 (local saddle). *Suppose $\alpha_1, \alpha_2 > 0$ are fixed. For $f \in \mathcal{C}^2$, at a local saddle point, $\forall\, \lambda \in Sp(H_{\alpha_1,\alpha_2}(f))$, $\Re(\lambda) \leq 0$. $\forall\, z \in \mathbb{C}$ with $\Re(z) \leq 0$, there exists a quadratic function q and a local saddle point (x^*, y^*) such that $z \in Sp(H_{\alpha_1,\alpha_2}(q))$. For bilinear functions, at a local saddle point $\Re(\lambda) = 0$ for all $\lambda \in Sp(H_{\alpha_1,\alpha_2})$.*

Proof. The convergence analysis reduces to the spectral study of $H_{1,\gamma}$. With the similarity transformation:

$$H' = U^{-1} H_{1,\gamma} U = \begin{bmatrix} -\partial^2_{xx} f & -\sqrt{\gamma}\partial^2_{xy} f \\ \sqrt{\gamma}\partial^2_{yx} f & \gamma \partial^2_{yy} f \end{bmatrix}, \; U = \begin{bmatrix} I & \mathbf{0} \\ \mathbf{0} & \sqrt{\gamma} I \end{bmatrix}, \tag{3.86}$$

[5] A real $n \times n$ matrix A is negative semi-definite if for any $x \in \mathbb{R}^n$, $x^\top A x \leq 0$. See e.g. Wang et al. (2010).

It suffices to study the spectrum of H'. For any local saddle point (x^*, y^*), we have:

$$\partial^2_{xx} f(x^*, y^*) \succeq \mathbf{0},\ \partial^2_{yy} f(x^*, y^*) \preceq \mathbf{0}. \tag{3.87}$$

From this necessary condition, $\Re(H') := (H' + H'^\top)/2$ is negative semi-definite, and with the Ky Fan inequality (Fan (1950)) we have $\Re(\mathrm{Sp}(H')) \prec \mathrm{Sp}(\Re(H')) \prec \mathbf{0}$, with "$\prec$" meaning majorization (Marshall et al., 1979). The second part can be proved by assuming $z = -u + iv$ with $u \geq 0$ and $v \in \mathbb{R}$. The quadratic function can be

$$q = \frac{ux^2}{2} - \frac{uy^2}{2\gamma} + \frac{v}{\sqrt{\gamma}} xy,$$

since one can verify that $(0, 0)$ is a local saddle point where:

$$H_{1,\gamma} = \begin{bmatrix} -u & -v/\sqrt{\gamma} \\ v\sqrt{\gamma} & -u \end{bmatrix}, \tag{3.88}$$

whose two eigenvalues are z and $\bar{z}$. For bilinear games $f = x^\top C y + a^\top x + b^\top y$, at any local saddle point, the Jacobian matrix of the vector field is:

$$H_{1,\gamma} = \begin{bmatrix} \mathbf{0} & -C \\ \gamma C^\top & \mathbf{0} \end{bmatrix}. \tag{3.89}$$

The eigenvalues are $\lambda = \pm i\sqrt{\gamma}\sigma$, with σ a singular value of C. □

This result is a slight extension of Daskalakis and Panageas (2018, Lemma 2.4). Combined with Lemma 3.4.1, we can show that EG/OGD converges for any local saddle points where the Jacobian $H(f)$ is non-singular, and also the feasible range of k for OGD:

Theorem 3.4.2 (stability of EG/OGD at local saddle points). *EG$(\alpha, \alpha, 1)$ is exponentially stable at any local saddle point if at such a point, $0 < |\lambda| < 1/\alpha$ for every $\lambda \in Sp(H)$. OGD(k, α, α) is exponentially stable at any local saddle point if $1 < k \leq 2$ and $0 < |\lambda| < 1/(k\alpha)$ for every $\lambda \in Sp(H)$. If $k \geq 3$, OGD(k, α_1, α_2) is not exponentially stable for bilinear games.*

Proof. At a local saddle point, from Lemma 3.4.1, for any $\lambda \in \mathrm{Sp}(H)$, $\Re(\lambda) \leq 0$. The corollary follows with $0 < |\lambda| < 1/\alpha$ for every $\lambda \in \mathrm{Sp}(H)$ and Theorem 3.3.2, since if $\beta = 1$, we can show:

$$\Re(\lambda) \leq 0 \text{ and } 0 < |\lambda| < 1 \Longrightarrow |1 + \lambda + \lambda^2| < 1, \tag{3.90}$$

with the following Mathematica code (rewrite $\lambda = u + iv$ with $u, v \in \mathbb{R}$):

```
Reduce[ForAll[{u, v}, u <= 0 && 0 < u^2 + v^2 < 1, (v + 2 u v)^2
+ (1 + u + u^2 - v^2)^2 < 1]],
```

and the result is `True`. For OGD, if $1 < k \leq 2$, we use Theorem 3.3.2, Lemma 3.4.1, and the following Mathematica code (rewrite $\lambda = u + iv$ with $u, v \in \mathbb{R}$):

```
Reduce[ForAll[{u,v,k}, 0 < u^2+v^2<1/k^2 && u<=0 && 1<k<=2,
(u^2+v^2)(-3+k+(1+k)(u^2+v^2)) <2u(-1+k(u^2+v^2))]].
```

The result is `True`. If $k \geq 3$ and the game is bilinear, from Theorem 3.3.2, Theorem 3.3.3 and Lemma 3.4.1 we must have $4|\lambda|^4 < 0$ to obtain local convergence, which is obviously false. □

3.4.2 Local Minimax Points

Now we study how gradient algorithms converge to local minimax points. We do not have the results in Theorem 3.4.2, since different from local saddle points, the spectrum of the Jacobian $H_{\alpha_1,\alpha_2}(f)$ is quite arbitrary:

Lemma 3.4.3 (**spectrum of local minimax can be arbitrary**). *Given $\alpha_1, \alpha_2 > 0$, for any $z \in \mathbb{C}$, there exists a quadratic function q and a local minimax point (x^*, y^*) where $z \in Sp(H_{\alpha_1,\alpha_2}(q))$.*

Proof. Let us assume $z = u + iv$ with $(u, v) \in \mathbb{R}^2$. We first construct a real polynomial:

$$(\lambda - z)(\lambda - \bar{z}) = \lambda^2 - 2u\lambda + u^2 + v^2 = 0. \tag{3.91}$$

On the other hand, the characteristic polynomial of $H_{\alpha_1,\alpha_2}(q)$ with $q(x, y) = ax^2/2 + by^2/2 + cxy$ is:

$$\lambda^2 + (\alpha_1 a - \alpha_2 b)\lambda + \alpha_1\alpha_2(c^2 - ab) = 0. \tag{3.92}$$

Comparing (3.91) and (3.92), it suffices to require that:

$$\alpha_1 a - \alpha_2 b = -2u,\ \alpha_1\alpha_2(c^2 - ab) = u^2 + v^2, \tag{3.93}$$

which always has real solutions given $(\alpha_1 > 0, \alpha_2 > 0, u, v)$. □

This result shows that local minimax points are a more general class than the class of local stable stationary points (LSSPs) as proposed recently in Berard et al. (2020) (see also Definition 2.2.14), in terms of zero-sum games, since LSSPs are defined such that $\Re(\lambda) < 0$ for any $\lambda \in \mathrm{Sp}(H_{\alpha,\alpha})$ and $\alpha > 0$ (note the slight change of signs due to the difference of the notations). Under certain assumptions, 2TS gradient algorithms can converge to local minimax points. The following result slightly extends Jin et al. (2020) where only GDA is analyzed:

Theorem 3.4.4 (**stability of EG/OGD at strict local minimax points**). *Assume at a stationary point* (x^*, y^*),

$$\partial^2_{yy} f \prec \mathbf{0} \text{ and } \partial^2_{xx} f - \partial^2_{xy} f (\partial^2_{yy} f)^{-1} \partial^2_{yx} f \succ \mathbf{0}. \tag{3.94}$$

Then $\exists\, \gamma_0 > 0$ *and* $\alpha_0 > 0$ *such that* $\forall\, \gamma > \gamma_0, 0 < \alpha_2 < \alpha_0$ *and* $\alpha_1 = \alpha_2/\gamma$, *EG and OGD (with* $k > 1$*) are exponentially stable.*

Proof. Assume $x \in \mathbb{R}^n$ and Using Lemma 36 of Jin et al. (2020), for any $\delta > 0$, there exists $\gamma_0 > 0$, when $\gamma > \gamma_0$, the eigenvalues of $H(1/\gamma, 1)$, $\lambda_1, \ldots, \lambda_n, \lambda_{n+1}, \ldots, \lambda_{m+n}$, are:

$$|\lambda_i + \mu_i/\gamma| < \delta/\gamma,\ \forall i = 1, \ldots, n, |\lambda_{i+n} - \nu_i| < \delta,\ \forall i = 1, \ldots, m, \tag{3.95}$$

where $\mu_i \in \mathrm{Sp}(\partial^2_{xx} f - \partial^2_{xy} f (\partial^2_{yy} f)^{-1} \partial^2_{yx} f)$ and $\nu_i \in \mathrm{Sp}(\partial^2_{yy} f)$. From our assumption, $\mu_i > 0$ and $\nu_i < 0$. With (3.95), there exists γ_0 such that for every $\gamma > \gamma_0$, $\Re(\lambda_i) < 0$ for all $\lambda_i \in H(1/\gamma, 1)$. From Theorem 3.4.2, EG ($\beta = 1$) and OGD ($1 < k \leq 2$) are exponentially stable if α_2 is small enough. □

In fact, the theorem above can be extended to the momentum methods as well (see Section 3.3.2). As we have seen in Corollary 2.3.14, (3.94) is sufficient for being local minimax (see also Fiez et al. (2019); Wang et al. (2020); Zhang et al. (2021) for applications in GANs). However, without the assumption (3.94) (see also Jin et al. (2020, Theorem 28) for GDA), the convergence is more difficult:

Proposition 3.4.5 (**stability of gradient algorithms at general local minimax points**). *There exists a quadratic function (e.g.,* $q(x, y) = -x^2 + xy$*) and a global (thus local, from Theorem 2.4.4) minimax point* $z^* = (x^*, y^*)$ *where*

- *GDA (with momentum or alternating updates) does not converge to* z^**, for any hyper-parameter choice.*
- *EG/OGD do not converge to* z^* *given* $\alpha_1 = \alpha_2$*, or* $\alpha_2 \to 0$*; otherwise there exist hyper-parameter choices such that EG/OGD converge to* z^**.*

- *Alternating OGD does not converge to z^* given $\alpha_2 \to 0$.*

Proof. We consider $q(x, y) := -x^2 + xy$ as the example, with $\mathcal{X} = \mathcal{Y} = \mathbb{R}$. From (2.4.1) we know that $(0, 0)$ is a global minimax point. $(0, 0)$ is also local minimax since it is stationary (cf. Theorem 2.4.4). $H_{1,\gamma}$ at $(0, 0)$ is:

$$H_{1,\gamma} = \begin{bmatrix} 2 & -1 \\ \gamma & 0 \end{bmatrix}. \tag{3.96}$$

If $0 < \gamma \leq 1$, the two eigenvalues are $1 \pm \sqrt{1-\gamma}$ which are both real and positive. One can read from Theorem 3.3.5 (or Figure 3.3) and Theorem 3.3.2 (or Figure 3.2) that GDA (with momentum) and EG/OGD do not converge to $(0, 0)$, locally and globally. Specifically, when $\gamma = 1$, $\alpha_1 = \alpha_2$.

If $\gamma > 1$, the eigenvalues are $\lambda_{1,2} = 1 \pm i\sqrt{\gamma - 1}$, which have positive real parts. From Theorem 3.3.5 (or Figure 3.3), GDA (with momentum) do not converge to $(0, 0)$. Now let us study 2TS-EG and 2TS-OGD, which corresponds to the second point of Proposition 3.4.5.

2TS-EG Taking $\beta \to \infty$ we require that $\Re(\lambda + \lambda^2) < 0$, which simplifies to:

$$\alpha_1 + \alpha_1^2 - \alpha_1^2(\gamma - 1) < 0, \tag{3.97}$$

and thus

$$\alpha_2 > 1 + 2\alpha_1 > 1. \tag{3.98}$$

We cannot take α_2 to be arbitrarily small.

2TS-OGD For 2TS-OGD, we need α_2 to be $\Omega(1)$ as well. From Theorem 3.3.2, we take $k \to 1_+$ so that the convergence region is the largest:

$$|\lambda| < 1, \ |\lambda - 1/2| > 1/2. \tag{3.99}$$

Bringing in the eigenvalues $\alpha_1(1 \pm i\sqrt{\gamma - 1})$, we obtain:

$$\alpha_1 < 1, \ 1/\alpha_1 < \gamma < 1/\alpha_1^2. \tag{3.100}$$

In other words, $1 < \alpha_2 < 1/\alpha_1$. We could take α_1 infinitesimal but not α_2.

Alternating updates Now let us study alternating updates on this example. We use the same framework as Section 3.2.2. We only study GDA and OGD for illustration purpose and other gradient algorithms follow similarly. The alternating GDA can be written as ($\alpha_1 > 0$, $\alpha_2 > 0$):

$$x_{t+1} = x_t - \alpha_1 \partial_x f(x_t, y_t),\ y_{t+1} = y_t + \alpha_2 \partial_y f(x_{t+1}, y_t), \tag{3.101}$$

and the alternating OGD can be written as (c.f. (3.69))($\alpha_1 > 0$, $\alpha_2 > 0$, $k > 1$):

$$x_{t+1} = x_t - k\alpha_1 \partial_x f(x_t, y_t) + \alpha_1 \partial_x f(x_{t-1}, y_{t-1}), \tag{3.102}$$

$$y_{t+1} = y_t + k\alpha_2 \partial_y f(x_{t+1}, y_t) - \alpha_2 \partial_y f(x_t, y_{t-1}). \tag{3.103}$$

Let us denote $A = \partial^2_{xx} f(x^*, y^*)$, $B = \partial^2_{yy} f(x^*, y^*)$ and $C = \partial^2_{xy} f(x^*, y^*)$. Locally, we can treat the gradient algorithms as a linear dynamical system. For instance, the linear dynamical system of simultaneous GDA and simultaneous OGD can be written as:

$$\text{GDA: } \begin{pmatrix} x_{t+1} - x^* \\ y_{t+1} - y^* \end{pmatrix} = \begin{pmatrix} x_t - x^* \\ y_t - y^* \end{pmatrix} + \begin{pmatrix} -\alpha_1 A & -\alpha_1 C \\ \alpha_2 C^\top & \alpha_2 B \end{pmatrix} \begin{pmatrix} x_t - x^* \\ y_t - y^* \end{pmatrix}, \tag{3.104}$$

$$\begin{aligned} \text{OGD: } \begin{pmatrix} x_{t+1} - x^* \\ y_{t+1} - y^* \end{pmatrix} = &\begin{pmatrix} x_t - x^* \\ y_t - y^* \end{pmatrix} + k \begin{pmatrix} -\alpha_1 A & -\alpha_1 C \\ \alpha_2 C^\top & \alpha_2 B \end{pmatrix} \begin{pmatrix} x_t - x^* \\ y_t - y^* \end{pmatrix} - \\ &- \begin{pmatrix} -\alpha_1 A & -\alpha_1 C \\ \alpha_2 C^\top & \alpha_2 B \end{pmatrix} \begin{pmatrix} x_{t-1} - x^* \\ y_{t-1} - y^* \end{pmatrix}. \end{aligned} \tag{3.105}$$

With Theorem 3.1.2, the characteristic equations for alternating GDA and alternating OGD are:

$$\text{GDA: } \det\left((\lambda - 1) I - \begin{pmatrix} -\alpha_1 A & -\alpha_1 C \\ \alpha_2 \lambda C^\top & \alpha_2 B \end{pmatrix} \right) = 0, \tag{3.106}$$

$$\text{OGD: } \det\left((\lambda - 1)\lambda I - (k\lambda - 1) \begin{pmatrix} -\alpha_1 A & -\alpha_1 C \\ \alpha_2 \lambda C^\top & \alpha_2 B \end{pmatrix} \right) = 0. \tag{3.107}$$

For the quadratic example $q(x, y) = -x^2 + xy$ we are considering, we have $A = -2, B = 0, C = 1$. Bringing it to (3.106), we obtain:

$$\text{GDA: } \lambda^2 + (\alpha_1\alpha_2 - 2\alpha_1 - 2)\lambda + 2\alpha_1 + 1 = 0, \tag{3.108}$$

$$\text{OGD: } \lambda^4 + \left(\alpha_1\alpha_2 k^2 - 2\alpha_1 k - 2\right)\lambda^3 + (2\alpha_1 - 2\alpha_1\alpha_2 k + 2\alpha_1 k + 1)\lambda^2 + (\alpha_1\alpha_2 - 2\alpha_1)\lambda = 0. \tag{3.109}$$

From Corollary 3.1.3, alternating GDA is stable iff:

$$2\alpha_1 + 1 < 1,\ |\alpha_1\alpha_2 - 2\alpha_1 - 2| < 2\alpha_1 + 2. \tag{3.110}$$

Note that the first condition can never hold since $\alpha_1 > 0$. Hence, alternating GDA cannot converge to the local minimax point $(0, 0)$ if the initialization is not at $(0, 0)$. For alternating OGD, the second equation of (3.108) can be simplified as $\lambda = 0$ or:

$$\lambda^3 + \left(\alpha_1\alpha_2 k^2 - 2\alpha_1 k - 2\right)\lambda^2 + (2\alpha_1 - 2\alpha_1\alpha_2 k + 2\alpha_1 k + 1)\lambda + \alpha_1(\alpha_2 - 2) = 0. \quad (3.111)$$

Using Corollary 3.1.3 again we know that alternating OGD is stable iff:

$$|c| < 1,\ |a + c| < 1 + b,\ b - ac < 1 - c^2, \quad (3.112)$$

where $a = \alpha_1\alpha_2 k^2 - 2\alpha_1 k - 2$, $b = 2\alpha_1 - 2\alpha_1\alpha_2 k + 2\alpha_1 k + 1$, $c = \alpha_1(\alpha_2 - 2)$. We simplify it on Mathematica:

```
Reduce[Abs[c]<1 && Abs[a+c] < 1 + b && b - a c < 1 - c^2 && k > 1
&& \alpha_1 > 0 && \alpha_2 > 0, {\alpha_1, \alpha_2}]
```

and obtain that:

$$\begin{aligned} &k > 1 \text{ and } 0 < \alpha_1 < \frac{4}{k^2 - 1} \text{ and} \\ &\sqrt{\frac{-2\alpha_1 + \alpha_1^2 k^2 + 1}{\alpha_1^2(k+1)^2}} + \frac{2\alpha_1 + \alpha_1 k - 1}{\alpha_1(k+1)} < \alpha_2 < \frac{4\alpha_1 + 4\alpha_1 k + 4}{\alpha_1 + \alpha_1 k^2 + 2\alpha_1 k}. \end{aligned} \quad (3.113)$$

Since $k > 1$ and

$$\begin{aligned} \sqrt{\frac{-2\alpha_1 + \alpha_1^2 k^2 + 1}{\alpha_1^2(k+1)^2}} + \frac{2\alpha_1 + \alpha_1 k - 1}{\alpha_1(k+1)} &\geq \sqrt{\frac{-2\alpha_1 + \alpha_1^2 + 1}{\alpha_1^2(k+1)^2}} + \frac{2\alpha_1 + \alpha_1 k - 1}{\alpha_1(k+1)} \\ &= \frac{\alpha_1 k + 2\alpha_1 - 1 + |\alpha_1 - 1|}{\alpha_1(k+1)} \\ &\geq \frac{\alpha_1 k + 2\alpha_1 - 1 + 1 - \alpha_1}{\alpha_1(k+1)} \\ &= 1, \end{aligned} \quad (3.114)$$

we have $\alpha_2 > 1$ for alternating updates of OGD. □

Prop. 3.4.5 extends Jin et al. (2020) by studying the degenerate case of $\partial^2_{yy} f$ and gradient algorithms other than GDA. The implication is two-fold:

- On the algorithmic aspect, we may not always rely on the usual ODE analysis (Mescheder et al., 2017; Mertikopoulos et al., 2018b; Fiez et al., 2019) when trying to find global/local minimax points, as such analysis relies on approximating gradient algorithms with their continuous versions, by taking the step sizes to be arbitrarily small. For EG/OGD, the step size of the follower (α_2) has to be large while the step size of the leader can be arbitrarily small, reflecting the asymmetric position of players in Stackelberg games (Jin et al., 2020).
- We may also need new solution concepts in addition to global/local minimax points in machine learning applications (e.g. Farnia and Ozdaglar, 2020; Schaefer et al., 2020), even though many machine learning applications, including GANs (Goodfellow et al., 2014) and adversarial training (Madry et al., 2018) are essentially based on the notion of global minimax points. This is because when applying standard gradient-based algorithms to do local search on machine learning applications, we cannot always expect the final solutions the algorithms find to cover all global/local minimax points.

3.5 Experiments

In this section, we present experimental results on simultaneous (Jacobi) and alternating (Gauss-Seidel) updates in bilinear games and GAN training.

Density plots We show the density plots (heat maps) of the spectral radii in Figure 3.4. We make plots for EG, OGD and momentum with both Jacobi and GS updates. These plots are made when $\beta_1 = \beta_2 = \beta$ and they agree with our theorems in §3.2.3.

Wasserstein GAN As in Daskalakis et al. (2018), we consider a WGAN (Arjovsky et al., 2017) that learns the mean of a Gaussian:

$$\min_\phi \max_\theta f(\phi, \theta) := \mathbb{E}_{x\sim\mathcal{N}(v,\sigma^2 I)}[s(\theta^\top x)] - \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\sigma^2 I)}[s(\theta^\top(z+\phi))], \tag{3.115}$$

with $s(x) := 1/(1+e^{-x})$ the sigmoid function. We study the local behavior near the saddle point $(\theta^*, \phi^*) = (\mathbf{0}, v)$, which depends on the Hessian:

$$\begin{bmatrix} \partial^2_{\phi\phi} f & \partial^2_{\phi\theta} f \\ \partial^2_{\theta\phi} f & \partial^2_{\theta\theta} f \end{bmatrix} = \begin{bmatrix} -\mathbb{E}_\phi[s''(\theta^\top z)\theta\theta^\top] & -\mathbb{E}_\phi[s''(\theta^\top z)\theta z^\top + s'(\theta^\top z)I] \\ (\partial^2_{\phi\theta} f)^\top & \mathbb{E}_v[s''(\theta^\top x)xx^\top] - \mathbb{E}_\phi[s''(\theta^\top z)zz^\top] \end{bmatrix},$$

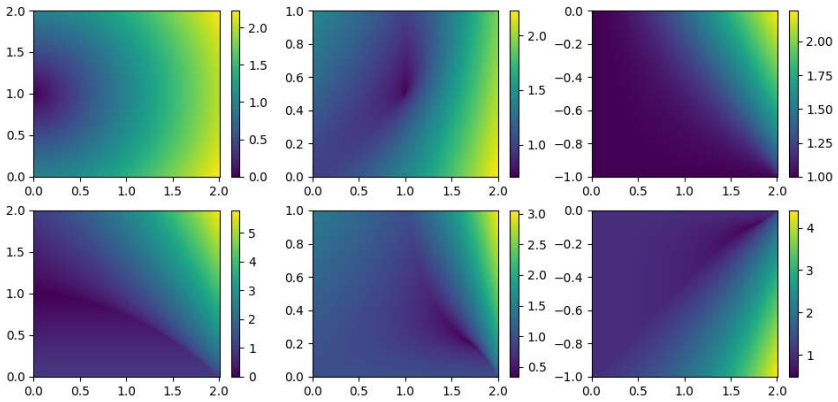


Figure 3.4: Heat maps of the spectral radii of different algorithms. We take $\sigma = 1$ for convenience. The horizontal axis is α and the vertical axis is β. **Top row:** Jacobi updates; **Bottom row:** Gauss–Seidel updates. **Columns** (left to right): EG; OGD; momentum. If the spectral radius is strictly less than one, it means that our algorithm converges. In each column, the Jacobi convergence region is contained in the GS convergence region (for EG we need an additional assumption, see Theorem 3.2.4).

Here $\mathbb{E}_v$ is a shorthand for $\mathbb{E}_{x\sim\mathcal{N}(v,\sigma^2 I)}$ and $\mathbb{E}_\phi$ is for $\mathbb{E}_{z\sim\mathcal{N}(\phi,\sigma^2 I)}$. At the saddle point, the Hessian is simplified as:

$$\begin{bmatrix} \partial^2_{\phi\phi} f & \partial^2_{\phi\theta} f \\ \partial^2_{\theta\phi} f & \partial^2_{\theta\theta} f \end{bmatrix} = \begin{bmatrix} \mathbf{0} & -s'(0)I \\ -s'(0)I & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{0} & -I/4 \\ -I/4 & \mathbf{0} \end{bmatrix}.$$

Therefore, this WGAN is locally a bilinear game. With GS updates, we find that Adam diverges, SGD goes around a limit cycle, and EG converges, as shown in Figure 3.5. We can see that Adam does not behave well even in this simple task of learning a single two-dimensional Gaussian with GAN.

Our next experiment shows that GS updates are more stable than Jacobi updates. In Figure 3.6, we can see that GS updates converge faster and they converge even if the corresponding Jacobi updates do not.

Mixtures of Gaussians (GMMs) Our last experiment is on learning GMMs with a vanilla GAN (Goodfellow et al., 2014) that does not directly fall into our analysis. We choose a 3-hidden layer ReLU network for both the generator and the discriminator, and each hidden layer has 256 units. We find that for GDA and OGD, Jacobi style updates

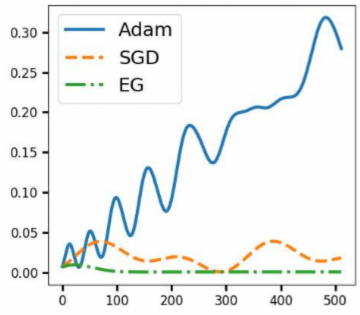


Figure 3.5: Comparison among Adam, SGD (or GDA) and EG in learning the mean of a Gaussian with WGAN with the squared distance.

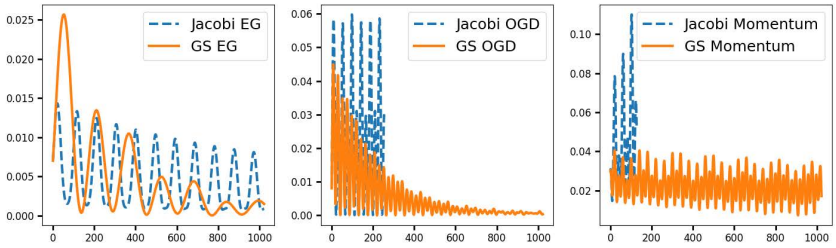


Figure 3.6: Jacobi vs. GS updates. **y-axis:** Squared distance $||\phi - v||^2$. **x-axis:** Number of epochs. **Left:** EG with $\gamma = 0.2, \alpha = 0.02$; **Middle:** OGD with $\alpha = 0.2$, $\beta_1 = 0.1$, $\beta_2 = 0$; **Right:** Momentum with $\alpha = 0.08$, $\beta = -0.1$. We plot only a few epochs for Jacobi if it does not converge. The setting is the same as Figure 3.5.

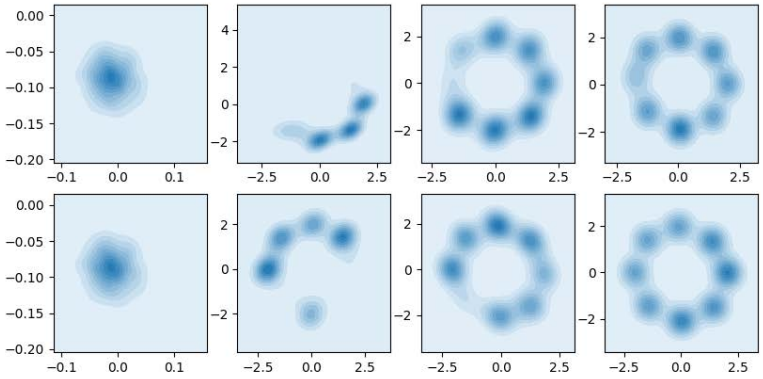


Figure 3.7: Test samples from the generator network trained with stochastic GDA (step size $\alpha = 0.01$). **Top row:** Jacobi updates; **Bottom row:** Gauss–Seidel updates. **Columns**: epoch 0, 10, 15, 20.

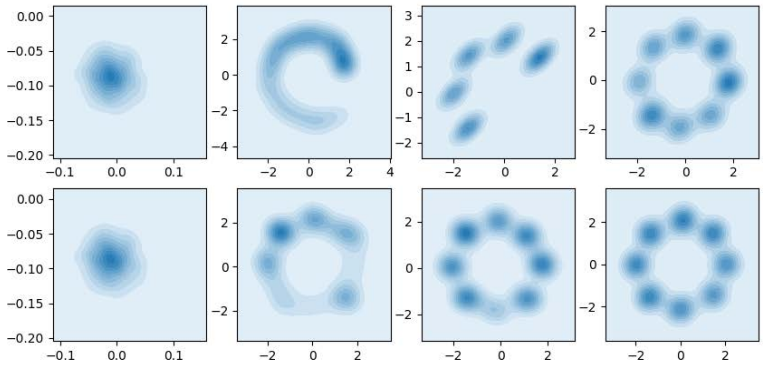


Figure 3.8: Test samples from the generator network trained with stochastic OGD ($\alpha = 2\beta = 0.02$). **Top row:** Jacobi updates; **Bottom row:** Gauss–Seidel updates. **Columns**: epoch 0, 10, 60, 100.

converge more slowly than GS updates, and whenever Jacobi updates converge, the corresponding GS updates converges as well. These comparisons can be found in Figure 3.7 and 3.8, which implies the possibility of extending our results to non-bilinear games.

Chapter 4

Newton-type Algorithms

In this chapter we study Newton-type algorithms for unconstrained minimax optimization:

$$\min_{x\in\mathbb{R}^n}\max_{y\in\mathbb{R}^m} f(x,y). \tag{4.1}$$

In conventional minimization (Bertsekas, 1997), a Newton algorithm requires the invertibility of the Hessian. Similarly, for minimax optimization we need additional assumptions. We mostly focus on strict local minimax points (SLmMs, see Corollary 2.3.14). an SLmM (x^*, y^*) satisfies:

$$\partial_x f(x^*,y^*) = \partial_y f(x^*,y^*) = \mathbf{0}, \tag{4.2}$$
$$\partial^2_{yy} f(x^*,y^*) \prec \mathbf{0},\ \partial^2_{xx} f(x^*,y^*) - \partial^2_{xy} f(x^*,y^*)(\partial^2_{yy} f(x^*,y^*))^{-1}\partial^2_{yx} f(x^*,y^*) \succ \mathbf{0}. \tag{4.3}$$

We will use the shorthand notations for total derivatives:

$$\mathtt{D}_x f := \partial_x f - \partial^2_{xy} f \cdot (\partial^2_{yy} f)^{-1} \cdot \partial_y f, \tag{4.4}$$
$$\mathtt{D}^2_{xx} f := \partial^2_{xx} f - \partial^2_{xy} f \cdot (\partial^2_{yy} f)^{-1} \cdot \partial^2_{yx} f. \tag{4.5}$$

The meaning of $\mathtt{D}_x f$ and $\mathtt{D}^2_{xx} f$ will be apparent in (4.9). The partial derivative operators can be distributed, e.g., $((\partial^2_{yy})^{-1} \cdot \partial_y) f := (\partial^2_{yy} f)^{-1} \cdot \partial_y f$, where the $\cdot$ sign means matrix multiplication.

In fact, many existing algorithms can be treated as *inexact* implementations of Uzawa's approach (Arrow et al., 1958), i.e., fast follower `F` and slow leader `L`. One could use Gradient Ascent (GA) with a large step size for `F` and Gradient Descent (GD) with a small step size for `L`, known as two-time-scale (Borkar, 2008; Heusel et al., 2017; Jin et al., 2020);

or perform k steps of GA update for F after every step of GD update of L (Goodfellow et al., 2014; Madry et al., 2018). Following Uzawa's approach, we use a Newton step for F and a GD update for L, that we call the *Gradient-Descent-Newton* (*a.k.a.* GD-Newton, GDN) algorithm. Although the method sounds simple and natural, surprisingly, it has not been well studied for non-convex-concave minimax problems, especially for the solution concepts such as the differential Stackelberg equilibrium (Fiez et al., 2019) and strict local minimax points (Evtushenko, 1974a; Jin et al., 2020). We compare GDN with similar algorithms that use the Hessian inverse, such as Total Gradient Descent Ascent (TGDA, Evtushenko, 1974b; Fiez et al., 2019) and Follow-the-Ridge (FR, Evtushenko, 1974b; Wang et al., 2020). Although the three algorithms share the same complexity, GDN has faster local convergence when the *follower problem* is ill-conditioned. A similar conclusion can be drawn by comparing with GDA methods.

Algorithms above achieve local linear convergence and still suffer from the ill-conditioning of the *leader problem* (and the follower problem except our GDN). Fortunately, in §4.3.2 we show that the Hessian for the leader is also well-defined and we propose the Complete Newton (CN) algorithm that performs Newton updates for both the leader and follower. CN enjoys local *quadratic* convergence and evades the ill-conditioning of both leader and follower problems in a local neighborhood. To the best of our knowledge, this is the first *genuine* second-order algorithm for nonconvex-nonconcave minimax optimization that (locally) converges super-linearly to (strict) local minimax solutions.[1] Rather surprisingly, we show that CN, being a second-order algorithm, can be implemented in similar computation complexity as the first-order alternatives such as TGDA, FR, and GDN. In §4.4, we verify theoretical properties of our Newton-type algorithms through experiments on training GANs.

In this chapter, we propose two Newton-type algorithms (GDN and CN) for minimax optimization that share similar complexity as existing alternatives but locally converge much faster, especially for ill-conditioned problems. To implement the Newton update, we take a *Hessian-free approach* (Martens, 2010) using only Hessian-vector products and Conjugate Gradient (CG), for which the per iteration complexity and memory usage are linear. Our results are summarized in Table 4.1. We also perform experiments on training GANs to complement our theoretical results which offer empirical insights on the aforementioned algorithms.

Newton-type methods have also been studied for other related problems such as variational inequalities (Han and Sun, 1998; Izmailov and Solodov, 2014) and bi-level opti-

[1] In Evtushenko (1974b), a superlinear algorithm was proposed, but its convergence was not formally proved.

Table 4.1: Comparison among algorithms for minimax optimization. p and p' are the numbers of conjugate gradient (CG) steps to solve $(\partial_{yy}^2)^{-1}\partial_y f$ and $(\mathtt{D}_{xx}^2)^{-1}\partial_x f$ respectively. $\rho_{\mathtt{L}}$ and $\rho_{\mathtt{F}}$ are the asymptotic linear rates defined in Thm. 4.2.3. n and m are dimensions of the leader and the follower. The convergence rates of TGDA/FR/GDN/CN are exact when we take enough CG steps ($p = m$ and $p' = n + m$). By solving ill-conditioning we mean that the convergence rates are not affected by the condition numbers.

Algorithm	**Time per step**	**Memory**	**Convergence rate**
GDA	$O(n+m)$	$O(n+m)$	linear; $\rho_{\mathtt{L}} \vee \rho_{\mathtt{F}}$ at best
TGDA/FR	$O(n+mp)$	$O(n+m)$	linear; $\rho_{\mathtt{L}} \vee \rho_{\mathtt{F}}$
GDA-k	$O(n+mk)$	$O(n+m)$	linear; $\rho_{\mathtt{L}}$ at best
GDN	$O(n+mp)$	$O(n+m)$	linear; $\rho_{\mathtt{L}}$
CN	$O((n+m)p'+mp)$	$O(n+m)$	quadratic

mization (Fliege et al., 2021). However, our work focuses on using Newton methods to find strict local minimax points in nonconvex-nonconcave minimax optimization, which is different from the previous settings.

4.1 Strict Local Minimax Points

We first point out that SLmMs are more general than the strict local Nash equilibrium Fiez et al. (2019), by which we mean

$$\partial_{yy}^2 f(x^*, y^*) \prec \mathbf{0} \text{ and } \partial_{xx}^2 f(x^*, y^*) \succ \mathbf{0}. \tag{4.6}$$

Example 4.1.1. $(0,0)$ *is an SLmM of the function* $f(x,y) = -3x^2 + xy^2 - y^2 + 4xy$ *on* $\mathbb{R}^2$ *but not a saddle point, since*

$$\partial_{xx}^2 f(0,0) = -6,\ \partial_{xy}^2 f(0,0) = 4,\ \partial_{yy}^2 f(0,0) = -2,$$

and thus $\mathtt{D}_{xx}^2 f(0,0) = 2$. *This function is nonconvex-nonconcave, because* $\partial_{yy}^2 f(x,y) = 2x - 2$ *is not always negative for* $(x,y) \in \mathbb{R}^2$, *and* $\partial_{xx}^2 f(0,0) < 0$. *It can also be verified that* $f(0,y) \le f(0,0) \le \max_{y\in\mathbb{R}} f(x,y)$, *i.e.,* $(0,0)$ *is a global minimax solution (Definition 2.1.3).*

At an SLmM (x^*, y^*), from

$$\partial_y f(x^*, y^*) = \mathbf{0},\ \partial^2_{yy} f(x^*, y^*) \prec \mathbf{0}$$

and the implicit function theorem, we know that for $f \in \mathcal{C}^2$, there are neighborhoods $\mathcal{N}(x^*) \subset \mathbb{R}^n$, $\mathcal{N}(y^*) \subset \mathbb{R}^m$ and a continuously differentiable function

$$r : \mathcal{N}(x^*) \to \mathcal{N}(y^*) \text{ s.t. } \partial_y f(x, r(x)) = \mathbf{0} \tag{4.7}$$

and $r(x)$ is a local maximizer of the function $f(x, \cdot)$. Also, we have

$$r'(x) = -((\partial^2_{yy})^{-1} \cdot \partial^2_{yx}) f(x, r(x)) \tag{4.8}$$

for any $x \in \mathcal{N}(x^*)$. We call this function the *local best-response function*. The local best response function leads to our definition of total derivatives. Define the "local maximum function" $\psi(x) := f(x, r(x))$ on $\mathcal{N}(x^*)$, from (4.8) we can derive from implicit function theorem that:

$$\psi'(x) = \mathtt{D}_x f(x, r(x)),\ \psi''(x) = \mathtt{D}^2_{xx} f(x, r(x)). \tag{4.9}$$

This is because from the chain rule and (4.8), we have:

$$\begin{aligned}
\psi'(x) &= \partial_x f(x, r(x)) + r'(x)^\top \partial_y f(x, r(x)) \\
&= \partial_x f(x, r(x)) - (\partial^2_{xy} \cdot (\partial^2_{yy})^{-1}) f(x, r(x)) \cdot \partial_y f(x, r(x)) \\
&= (\partial_x - \partial^2_{xy} \cdot (\partial^2_{yy})^{-1} \cdot \partial_y) f(x, r(x)) \\
&= \mathtt{D}_x f(x, r(x)).
\end{aligned} \tag{4.10}$$

Taking the total derivative of x again and using $\partial_y f(x, r(x)) = \mathbf{0}$, we have:

$$\begin{aligned}
\psi''(x) &= \frac{d}{dx} \mathtt{D}_x f(x, r(x)) \\
&= \frac{d}{dx} \partial_x f(x, r(x)) \\
&= \partial^2_{xx} f(x, r(x)) + r'(x)^\top \partial_{yx} f(x, r(x)) \\
&= \partial^2_{xx} f(x, r(x)) - (\partial^2_{xy} \cdot (\partial^2_{yy})^{-1}) f(x, r(x)) \cdot \partial^2_{yx} f(x, r(x)) \\
&= (\partial^2_{xx} - \partial^2_{xy} \cdot (\partial^2_{yy})^{-1} \cdot \partial^2_{yx}) f(x, r(x)) \\
&= \mathtt{D}^2_{xx} f(x, r(x)).
\end{aligned} \tag{4.11}$$

The practical relevance of SLmMs becomes important when training generative adversarial networks (GANs) and distributional robustness models, and we present our analysis in the following subsections.

4.1.1 GAN Training

Consider the following GAN training problem, where we minimize over generator G with parameter θ and maximize over discriminator D with parameter ϕ:

$$\min_\theta \max_\phi \ \ell(\theta, \phi), \quad \text{where}$$
$$\ell(\theta, \phi) = \mathbb{E}_{x\sim p_x}[f(D_\phi(x))] + \mathbb{E}_{z\sim p_z}[f(-D_\phi(G_\theta(z)))].$$

Under some mild assumptions (Nagarajan and Kolter, 2017), at a stationary point the partial Hessians satisfy:

$$\partial^2_{\theta\theta}\ell = \mathbf{0},\ \partial^2_{\phi\phi}\ell = 2f''(0)\mathbb{E}_{x\sim p_x}[\partial_\phi D_\phi(x)\cdot\partial_\phi D_\phi^\top(x)],$$
$$\partial^2_{\theta\phi}\ell = -f'(0)\cdot\partial_\theta\mathbb{E}_{x\sim p_z}[\partial_\phi G_\theta(D_\phi(x))].$$

Typically, $f'(0)\neq 0$ and $f''(0)<0$. For example, for vanilla GAN (Goodfellow et al., 2014),

$$f(x) = -\log(1+e^{-x}),$$

giving $f'(0)=\frac{1}{2}$ and $f''(0)=-\frac{1}{4}$. Therefore, under full rank assumptions (Nagarajan and Kolter, 2017),

$$\partial^2_{\phi\phi}\ell \prec \mathbf{0}, \quad \mathtt{D}^2_{\theta\theta}\ell = (\partial^2_{\theta\theta} - \partial^2_{\theta\phi}\cdot(\partial^2_{\phi\phi})^{-1}\cdot\partial^2_{\phi\theta})\ell \succ \mathbf{0},$$

i.e. the stationary point is an SLmM. The loss ℓ is typically *not* a convex function of the generator parameter θ.

4.1.2 Distributional Robustness

Given N data samples $\{\xi_i\}_{i=1}^N$, the Wasserstein distributional robustness model can be written as:

$$\min_\theta \max_\Omega f(\theta,\Omega) = \sum_{i=1}^N \ell(\theta,\omega_i) - \gamma\|\omega_i-\xi_i\|^2, \tag{4.12}$$

where we denote $\Omega=\{\omega_i\}_{i=1}^N$ as the collection of adversarial samples. Here θ denotes the model parameters, and ℓ is the loss function. The goal of this task is to find robust model parameters θ against adversarial perturbation of samples, ω_i. At a stationary point (θ^*,Ω^*), Sinha et al. (2018) shows that for large γ, $\partial^2_{\Omega\Omega}f(\theta^*,\Omega^*)$ is negative definite. Moreover, the total Hessian $\mathtt{D}^2_{\theta\theta}f(\theta^*,\Omega^*)$ is:

$$\sum_{i=1}^N \partial^2_{\theta\theta}\ell(\theta^*,\omega_i^*) - M_i(\partial^2_{\omega\omega}\ell(\theta^*,\omega_i^*) - 2\gamma I)^{-1}M_i^\top, \tag{4.13}$$

where $M_i := \partial^2_{\theta\omega}\ell(\theta^*, \omega_i^*)$. Under assumptions that θ^* is a local minimum of the adversarial training loss $\sum_{i=1}^N \ell(\cdot, \omega_i^*)$ and that M_i is full row rank for at least one adversarial example ω_i^*, we can show that (θ^*, Ω^*) is an SLmM for large γ. Moreover, (θ^*, Ω^*) is not necessarily a strict local Nash equilibrium since $\partial^2_{\theta\theta}\ell \succ 0$ may not hold.

Proposition 4.1.2. *Suppose $(\theta^*, \Omega^*) = (\theta^*, \omega_1^*, \ldots, \omega_N^*)$ is a stationary point of*

$$f(\theta, \Omega) = \sum_{i=1}^N \ell(\theta, \omega_i) - \gamma\|\omega_i - \xi_i\|^2, \tag{4.14}$$

where ℓ is twice differentiable. If at this point, θ^* is a local minimum of $\sum_{i=1}^N \ell(\cdot, \omega_i^*)$ and there exists at least an adversarial sample ω_i^* such that

$$M_i = \partial^2_{\theta\omega}\ell(\theta^*, \omega_i^*) \tag{4.15}$$

is full row rank, and

$$\gamma > \frac{1}{2}\max_{i=1,\cdots,N} \lambda_{\max}(\partial^2_{\omega\omega}\ell(\theta^*, \omega_i^*)), \tag{4.16}$$

with $\lambda_{\max}(\cdot)$ being the largest eigenvalue of a matrix, then (θ^*, Ω^*) is an SLmM of f but not necessarily a strict local Nash equilibrium.

Before we move on to the proof, let us first interpret the stationary point. Solving the condition that:

$$\partial_\theta f(\theta^*, \Omega^*) = \sum_{i=1}^N \partial_\theta \ell(\theta^*, \omega_i^*) = \mathbf{0},\ \partial_{\omega_i}\ell(\theta^*, \omega_i^*) - 2\gamma(\omega_i - \xi_i) = \mathbf{0}, \tag{4.17}$$

i.e.,

$$\sum_{i=1}^N \partial_\theta \ell(\theta^*, \omega_i^*) = \mathbf{0},\ \omega_i^* = \xi_i + \frac{1}{2\gamma}\partial_{\omega_i}\ell(\theta^*, \omega_i^*). \tag{4.18}$$

For large γ, this tells us that θ^* is a stationary point of the original training loss given the adversarial examples ω_i^*, and ω_i^* is a perturbation of the original samples ξ_i. We furthermore want θ^* to be a local minimum of the loss $\sum_{i=1}^N \ell(\cdot, \omega_i^*)$, and thus from the second-order necessary condition, we have:

$$\sum_{i=1}^N \partial^2_{\theta\theta}\ell(\theta^*, \omega_i^*) \succeq \mathbf{0}. \tag{4.19}$$

Note that it is very common in deep learning that the matrix $\sum_{i=1}^N \partial^2_{\theta\theta}\ell(\theta^*,\omega_i^*)$ is singular Sagun et al. (2016), and thus (θ^*,Ω^*) is not a strict local Nash equilibrium (see (4.6)). However, we can show that (θ^*,Ω^*) is an SLmM under mild assumptions. We note that (4.16) can be guaranteed if γ is greater than some Lipschitz smoothness constant of ℓ, as shown in Sinha et al. (2018).

Proof. We compute from (4.14) that:

$$\begin{aligned}
&\partial_{\omega_i^*,\omega_i^*} f(\theta^*,\Omega^*) = \partial^2_{\omega\omega}\ell(\theta^*,\omega_i^*) - 2\gamma I,\\
&\mathtt{D}^2_{\theta\theta} f(\theta^*,\Omega^*) = \sum_{i=1}^N \partial^2_{\theta\theta}\ell(\theta^*,\omega_i^*) - M_i(\partial^2_{\omega\omega}\ell(\theta^*,\omega_i^*) - 2\gamma I)^{-1}M_i^\top.
\end{aligned} \tag{4.20}$$

If $\gamma > \frac{1}{2}\max_{i=1,\cdots,N}\lambda_{\max}(\partial^2_{\omega\omega}\ell(\theta^*,\omega_i^*))$, then for any $i=1,\cdots,N$,

$$\begin{aligned}
\partial^2_{\omega_i,\omega_i} f(\theta^*,\Omega^*) &= \partial^2_{\omega\omega}\ell(\theta^*,\omega_i^*) - 2\gamma I\\
&\prec (\lambda_{\max}(\partial^2_{\omega\omega}\ell(\theta^*,\omega_i^*)) - \max_{j=1,\cdots,N}\lambda_{\max}(\partial^2_{\omega\omega}\ell(\theta^*,\omega_j^*)))I\\
&\preceq \mathbf{0},
\end{aligned} \tag{4.21}$$

where in the second line, we used the fact that for a symmetric matrix A, we have $A \preceq \lambda_{\max}(A)I$. Hence we obtain that $\partial^2_{\omega_i,\omega_i} f(\theta^*,\Omega^*) \prec \mathbf{0}$. We now compute $\mathtt{D}^2_{\theta\theta} f(\theta^*,\Omega^*)$ as:

$$\begin{aligned}
\mathtt{D}^2_{\theta\theta} f(\theta^*,\Omega^*) &= \sum_{i=1}^N \partial^2_{\theta\theta}\ell(\theta^*,\omega_i^*) - M_i(\partial^2_{\omega\omega}\ell(\theta^*,\omega_i^*) - 2\gamma I)^{-1}M_i^\top\\
&= \sum_{i=1}^N \partial^2_{\theta\theta}\ell(\theta^*,\omega_i^*) - \sum_{i=1}^N M_i(\partial^2_{\omega_i,\omega_i} f(\theta^*,\Omega^*))^{-1}M_i^\top.
\end{aligned} \tag{4.22}$$

We assumed that θ^* is a local minimum of the training loss and thus the first term is positive semi-definite. We note that the second term is negative semi-definite because for any model parameter θ_0 and any sample ω_i^*, we can write:

$$\theta_0^\top M_i(\partial^2_{\omega_i,\omega_i} f(\theta^*,\Omega^*))^{-1}M_i^\top\theta_0 = (M_i^\top\theta_0)^\top(\partial^2_{\omega_i,\omega_i} f(\theta^*,\Omega^*))^{-1}M_i^\top\theta_0 \le 0, \tag{4.23}$$

since $\partial^2_{\omega_i,\omega_i} f(\theta^*,\Omega^*) \prec \mathbf{0}$. Furthermore, if M_i^* has full row rank, (4.23) is always negative for all $\theta_0 \neq \mathbf{0}$, and hence the second term of (4.22) is negative definite, resulting in $\mathtt{D}^2_{\theta\theta} f(\theta^*,\Omega^*) \succ \mathbf{0}$. Assume otherwise. Since $(\partial^2_{\omega_i,\omega_i} f(\theta^*,\Omega^*))^{-1}$ is also negative definite (this can be proved from the spectral decomposition), we must have:

$$M_i^\top\theta_0 = \mathbf{0}. \tag{4.24}$$

Since M_i is full row rank, the row vectors of M_i are linearly independent, and thus we must have $\theta_0 = \mathbf{0}$. This is a contradiction. So we have proved that

$$M_i(\partial^2_{\omega_i,\omega_i} f(\theta^*, \Omega^*))^{-1} M_i^\top \prec \mathbf{0}$$

and thus $\mathtt{D}^2_{\theta\theta} f(\theta^*, \Omega^*) \succ \mathbf{0}$. Therefore, (4.21) and (4.22) tell us that under our assumptions, (θ^*, Ω^*) is an SLmM but not necessarily a strict local Nash equilibrium. □

4.2 Existing Algorithms

In this section we study local convergence rates of some existing algorithms at strict local minimax points.

4.2.1 GDA and its Variants

One of the first algorithms for the minimax problem (2.7) is gradient-descent-ascent (GDA) (Arrow et al., 1958), where we adopt GD as $\mathtt{L}$ for updating the leader while we use GA as $\mathtt{F}$ for updating the follower:

$$\begin{aligned} x_{t+1} &= x_t - \alpha_{\mathtt{L}} \cdot \partial_x f(x_t, y_t), \\ y_{t+1} &= y_t + \alpha_{\mathtt{F}} \cdot \partial_y f(x_t, y_t). \end{aligned} \tag{4.25}$$

We consider two different scales of the step sizes (Heusel et al., 2017; Jin et al., 2020), *i.e.* $\alpha_{\mathtt{L}} = o(\alpha_{\mathtt{F}})$, as is typical in stochastic approximation (Borkar, 2008), to converge linearly at a SLmM. However, in practice two-time-scale GDA (2TS-GDA) is hard to tune, especially when the follower problem is ill-conditioned, as we will verify in our experiments below. 2TS-GDA (locally) converges slower than TGDA and FR, hence also slower than GDN.

Using results from Jin et al. (2020) and similar notations as in Theorem 4.2.3, we derive the following result for 2TS-GDA:

Theorem 4.2.1. *Around a SLmM (x^*, y^*), for any $\delta > 0$, $\exists\, \gamma_0 > 0$ such that for any $\gamma > \gamma_0$, $\alpha_F > 0$ and $\alpha_L = \alpha_F/\gamma$, 2TS-GDA has asymptotic linear convergence rate $\rho = \rho_L \vee \rho_F$, where $\rho_L := (|1 - \alpha_L\lambda_1| + \alpha_L\delta) \vee (|1 - \alpha_L\lambda_n| + \alpha_L\delta)$ and $\rho_F := (|1 - \alpha_F\mu_1| + \alpha_F\delta) \vee (|1 - \alpha_F\mu_m| + \alpha_F\delta)$, with λ_1 and λ_n (resp. μ_1 and μ_m) being the largest and smallest eigenvalue of $\mathtt{D}^2_{xx} f(x^*, y^*)$ (resp. of $-\partial^2_{yy} f(x^*, y^*)$).*

Proof. The Jacobian of 2TS-GDA (4.25) at (x^*, y^*) is:

$$I + \alpha_{\mathtt{F}} \begin{bmatrix} -\gamma^{-1}\partial^2_{xx} f & -\gamma^{-1}\partial^2_{xy} f \\ \partial^2_{yx} f & \partial^2_{yy} f \end{bmatrix} =: I + \alpha_{\mathtt{F}} H. \tag{4.26}$$

Using Jin et al. (2020, Lemma 36), for any $\delta > 0$, there exist $\gamma > 0$ large enough, such that the eigenvalues of H, $\nu_1, \ldots, \nu_n, \nu_{n+1}, \ldots, \nu_{m+n}$ satisfy:

$$|\nu_i + \lambda_i/\gamma| < \delta/\gamma, \, \forall i = 1, \ldots, n, |\nu_{j+n} + \mu_j| < \delta, \, \forall j = 1, \ldots, m, \tag{4.27}$$

where $\lambda_i \in \mathrm{Sp}((\partial_{xx} - \partial^2_{xy} \cdot (\partial^2_{yy})^{-1} \cdot \partial^2_{yx}) f)$ and $\mu_j \in \mathrm{Sp}(-\partial^2_{yy} f)$. The spectral radius is then:

$$\max_{k \in [n+m]} |1 + \alpha_{\mathtt{F}} \nu_k| = \max_{i \in [n]} |1 + \alpha_{\mathtt{F}} \nu_i| \vee \max_{j \in [m]} |1 + \alpha_{\mathtt{F}} \nu_{j+n}|. \tag{4.28}$$

We can use triangle inequality and (4.27) to obtain that for any $\gamma \geq \gamma_0$:

$$|1 + \alpha_{\mathtt{F}} \nu_i| \leq |1 - \alpha_{\mathtt{F}} \mu_i/\gamma| + \alpha_{\mathtt{F}} \delta/\gamma = |1 - \alpha_{\mathtt{L}} \mu_i| + \alpha_{\mathtt{L}} \delta, \, \forall i \in [n]. \tag{4.29}$$

Similarly, $|1 + \alpha_{\mathtt{F}} \nu_{j+n}| \leq |1 - \alpha_{\mathtt{F}} \mu_j| + \alpha_{\mathtt{F}} \delta$. □

k-step gradient descent ascent We also study k-step gradient descent ascent (GDA-k) as proposed in Goodfellow et al. (2014). After each GD update on the leader, GDA-k performs k GA updates on the follower:

$$\begin{aligned} x_{t+1} &= x_t - \alpha \cdot \partial_x f(x_t, y_t), \\ y_{t+1} &= g^{(k)}(y_t) \text{ with } g(y) = y + \alpha \cdot \partial_y f(x_{t+1}, y), \end{aligned}$$

where $g^{(k)}$ means composition for k times. Letting $k \to \infty$ amounts to solving the follower problem exactly by gradient ascent steps (see (4.85)). Continuing with the notation in Thm. 4.2.3, we derive the following result:

Theorem 4.2.2 (GDA-∞). *GDA-k achieves an asymptotic linear convergence rate*

$$\rho_{\mathtt{L}} = |1 - \alpha\lambda_1| \vee |1 - \alpha\lambda_n|$$

at an SLmM (x^*, y^*) when $k \to \infty$ and $\alpha < 2/\mu_1$. If $\mu_1 < \lambda_1 + \lambda_n$, choosing $\alpha = 2/(\lambda_1 + \lambda_n)$ we obtain the optimal convergence rate

$$\frac{\kappa_L - 1}{\kappa_L + 1},$$

otherwise with α approaching $2/\mu_1$ we obtain a suboptimal rate $1 - 2\lambda_n/\mu_1$.

Proof. The Jacobian matrix of the simultaneous version (replacing x_{t+1} with x_t in the update of y) update at (x^*, y^*) is:

$$J_k = \begin{bmatrix} I - \alpha\partial^2_{xx} f & -\alpha\partial^2_{xy} f \\ \alpha\sum_{i=0}^{k-1}(I + \alpha\partial^2_{yy} f)^i \partial^2_{yx} f & (I + \alpha\partial^2_{yy} f)^k \end{bmatrix}. \tag{4.30}$$

This is because $g^{(k)}(y)$, the update in GDA-k, can be written iteratively:

$$g^{(1)} = g(x_t, y), \ldots, g^{(k)} = g(x_t, g^{(k-1)}), \tag{4.31}$$

where $g(x_t, y) := y + \alpha\partial_y f(x_t, y)$. We verify that the total derivative follows $dg^{(k)}x = \partial_x g + \partial_y g \cdot dg^{(k-1)}x$, and prove the derivative over x_t by induction.

$$\sum_{i=0}^{\infty}(I + \alpha\partial^2_{yy} f)^i = (-\alpha\partial^2_{yy} f)^{-1}, \text{ and } (I + \alpha\partial^2_{yy} f)^k \to \mathbf{0}. \tag{4.32}$$

Note that the series converges iff $|1 - \alpha\mu_j| < 1$ for all $\mu_j \in \mathrm{Sp}(-\partial^2_{yy} f)$ (e.g. Meyer, 2000, Chapter 7), i.e. $\alpha < 2/\max_j \mu_j = 2/\mu_1$. Under this condition,

$$J_\infty = \begin{bmatrix} I - \alpha\partial^2_{xx} f & -\alpha\partial^2_{xy} f \\ -((\partial^2_{yy})^{-1} \cdot \partial^2_{yx}) f & \mathbf{0} \end{bmatrix}. \tag{4.33}$$

Using Theorem 3.2.1, the characteristic polynomial of GDA-∞ is:

$$\det \begin{bmatrix} (\lambda - 1)I + \alpha\partial^2_{xx} f & \alpha\partial^2_{xy} f \\ \lambda((\partial^2_{yy})^{-1} \cdot \partial^2_{yx}) f & \lambda I \end{bmatrix} = 0. \tag{4.34}$$

Solving the eigenvalues yields $1 - \alpha\lambda_i$ with $\lambda_i \in \mathrm{Sp}(\mathtt{D}^2_{xx} f)$.

The optimal convergence rate is achieved by optimizing $\max_i |1 - \alpha\lambda_i|$, which is achieved at $\alpha = 2/(\lambda_1 + \lambda_n)$. However, we also impose $\alpha < 2/\mu_1$, which yields the assumption that $\mu_1 < \lambda_1 + \lambda_n$. Otherwise, a suboptimal rate is obtained via taking $\alpha \to 2/\mu_1$.

We note that it is possible to modify GDA-k to be two-time-scale as well, i.e.,

$$x_{t+1} = x_t - \alpha_{\mathtt{L}} \cdot \partial_x f(x_t, y_t), \qquad y_{t+1} = g^{(k)}(y_t) \text{ with } g(y) = y + \alpha_{\mathtt{F}} \cdot \partial_y f(x_{t+1}, y). \tag{4.35}$$

With this modification, it suffices that $\alpha_{\mathtt{F}} < 2/\mu_1$ and the optimal rate is $1 - 2/(\kappa_{\mathtt{L}} + 1)$ with $\alpha_{\mathtt{L}} = 2/(\lambda_1 + \lambda_n)$. We do not need the constraint that $\mu_1 < \lambda_1 + \lambda_n$ and there is no suboptimal rate. However, when $\partial^2_{yy} f$ is ill-conditioned the number of follower steps might be very large to approximate GDA-∞. □

4.2.2 Total Gradient Descent Ascent (TGDA) and Follow-the-Ridge

After studying the local convergence of GDA, we present two algorithms that use the Hessian inverse information, Total Gradient Descent Ascent (TGDA) and Follow-the-Ridge (FR). TGDA takes a GA step for the follower and a *total* gradient ascent step for the leader: Fiez *et al.* (Fiez et al., 2019) proposed TGDA with $\mathtt{F}$ being gradient descent for the follower and $\mathtt{L}$ being *total* gradient ascent for the leader:

$$x_{t+1} = x_t - \alpha_{\mathtt{L}} \cdot \mathtt{D}_x f(x_t, y_t), \qquad y_{t+1} = y_t + \alpha_{\mathtt{F}} \cdot \partial_y f(x_t, y_t), \tag{4.36}$$

where we use the total gradient $\mathtt{D}$ instead of the partial derivative ∂_x for the update on the leader x. Its continuous dynamics was studied in Evtushenko (1974b) with linear convergence proved. More recently, the stochastic setting and the two-time-scale variant are studied in Fiez et al. (2019) for general sum games.

Follow the ridge (FR) Follow-the-ridge was proposed in Evtushenko (1974b) and its variant is recently studied by Wang et al. (2020). In this algorithm, $\mathtt{F}$ is a pre-conditioned gradient update for the follower and $\mathtt{L}$ is the usual gradient update for the leader:

$$x_{t+1} = x_t - \alpha_{\mathtt{L}} \cdot \partial_x f(x_t, y_t), \qquad y_{t+1} = y_t + (\alpha_{\mathtt{F}} \cdot \partial_y + \alpha_{\mathtt{L}} \cdot (\partial^2_{yy})^{-1} \cdot \partial^2_{yx} \cdot \partial_x) f(x_t, y_t). \tag{4.37}$$

In fact, it is not a coincidence that both TGDA and FR can be derived as first-order approximations of GDN—the two are in some sense "transpose" of each other. Indeed, denote $z = (x, y)$ and

$$P = \begin{bmatrix} -\alpha_{\mathtt{L}} I & \alpha_{\mathtt{L}}(\partial^2_{xy} \cdot (\partial^2_{yy})^{-1}) f \\ \mathbf{0} & \alpha_{\mathtt{F}} I \end{bmatrix}, \; \partial_z f = \begin{bmatrix} \partial_x f \\ \partial_y f \end{bmatrix}. \tag{4.38}$$

Then, we can equivalently rewrite TGDA and FR respectively as:

$$\text{TGDA} : z_{t+1} = z_t + P \cdot \partial_z f(z_t), \tag{4.39}$$

$$\text{FR} : z_{t+1} = z_t + P^\top \cdot \partial_z f(z_t). \tag{4.40}$$

In other words, the two algorithms amount to performing some pre-conditioning on GDA, and their preconditioning operators are simply transpose of each other. Since the preconditioning operator P is (block) triangular, it follows that TGDA and FR have the same Jacobian spectrum around a SLmM. We now present their asymptotic local convergence:

Theorem 4.2.3. *TGDA and FR achieve the same asymptotic linear convergence rate* $\rho = \rho_{\mathtt{L}} \vee \rho_{\mathtt{F}}$ *at an SLmM* (x^*, y^*)*,*

$$\begin{aligned} &\textit{where } \rho_{\mathtt{L}} = |1 - \alpha_{\mathtt{L}}\lambda_1| \vee |1 - \alpha_{\mathtt{L}}\lambda_n| \\ &\textit{and } \rho_{\mathtt{F}} = |1 - \alpha_{\mathtt{F}}\mu_1| \vee |1 - \alpha_{\mathtt{F}}\mu_m|, \end{aligned}$$

with λ_1 *and* λ_n *(resp.* μ_1 *and* μ_m*) being the largest and smallest eigenvalue of* $\mathtt{D}^2_{xx} f(x^*, y^*)$ *(resp. of* $-\partial^2_{yy} f(x^*, y^*)$*).*

Note that by an asymptotic linear rate ρ we meant

$$\rho = \limsup_{t \to \infty} \frac{\|z_{t+1} - z^*\|}{\|z_t - z^*\|}.$$

Choosing $\alpha_{\mathtt{L}} = 2/(\lambda_1 + \lambda_n)$, $\alpha_{\mathtt{F}} = 2/(\mu_1 + \mu_m)$ gives the optimal convergence rate

$$\frac{\kappa_{\mathtt{L}} - 1}{\kappa_{\mathtt{L}} + 1} \vee \frac{\kappa_{\mathtt{F}} - 1}{\kappa_{\mathtt{F}} + 1},$$

where $\kappa_{\mathtt{L}} := \lambda_1/\lambda_n$ and $\kappa_{\mathtt{F}} := \mu_1/\mu_m$. A slightly weaker result for FR, using only eigenvalues of the Hessian, has appeared in Wang et al. (2020).

Proof. Let us first prove the following lemma:

Lemma 4.2.4. *Given* $f : \mathbb{R}^d \to \mathbb{R}^{n \times m}$ *and* $g : \mathbb{R}^d \to \mathbb{R}^m$*, assume* g *is Fréchet differentiable at* z *and* $g(z) = \mathbf{0}$*, and* f *is continuous at* z*. Then, the product function* $h = fg$ *is Fréchet differentiable at* z *with* $h'(z) = f(z)g'(z)$*.*

Proof. It suffices to prove that $\|h(z+\delta) - h(z) - f(z)g'(z)^\top \delta\| = o(\|\delta\|)$. This is because:

$$\begin{aligned} &\|h(z+\delta) - h(z) - f(z)g'(z)^\top \delta\| \\ &= \|f(z+\delta)g(z+\delta) - f(z)g(z) - f(z)g'(z)^\top \delta\| \\ &= \|f(z+\delta)g(z+\delta) - f(z)g(z+\delta) + f(z)g(z+\delta) - f(z)g(z) - f(z)g'(z)^\top \delta\| \\ &\le \|(f(z+\delta) - f(z))g(z+\delta)\| + \|f(z)(g(z+\delta) - g(z) - g'(z)^\top \delta)\| \\ &\le \|f(z+\delta) - f(z)\| \cdot \|g(z+\delta)\| + \|f(z)\| \cdot \|g(z+\delta) - g(z) - g'(z)^\top \delta\| \\ &\le o(1) \cdot \|g(z+\delta) - g(z)\| + o(\|\delta\|) \\ &= o(\|\delta\|), \end{aligned} \tag{4.41}$$

where in the second last line, we used $g(z) = \mathbf{0}$, the continuity of f and the Fréchet differentiability of g. □

With Lemma 4.2.4 we compute the Jacobian at (x^*, y^*):

$$J_{\text{TGDA}} = \begin{bmatrix} I - \alpha_{\text{L}}(\partial^2_{xx} - \partial^2_{xy} \cdot (\partial^2_{yy})^{-1} \cdot \partial^2_{yx})f & \mathbf{0} \\ \alpha_{\text{F}}\partial^2_{yx} f & I + \alpha_{\text{F}}\partial^2_{yy} f \end{bmatrix} \tag{4.42}$$

The spectral radius can be easily computed as:

$$\rho(J_{\text{TGDA}}) = \max_i |1 - \alpha_{\text{L}}\lambda_i| \vee \max_j |1 - \alpha_{\text{F}}\mu_j|. \tag{4.43}$$

Now let us show that the Jacobian of FR has the same spectrum as TGDA. From (4.38) and the comment below we know that

$$J_{\text{TGDA}} = I + PHf(x^*, y^*),\ J_{\text{FR}} = I + P^\top Hf(x^*, y^*), \tag{4.44}$$

where

$$P = \begin{bmatrix} -\alpha_{\text{L}} I & \alpha_{\text{L}}(\partial^2_{xy} \cdot (\partial^2_{yy})^{-1})f \\ \mathbf{0} & \alpha_{\text{F}} I \end{bmatrix},\ \text{and}\ H = \begin{bmatrix} \partial^2_{xx} f & \partial^2_{xy} f \\ \partial^2_{yx} f & \partial^2_{yy} f \end{bmatrix}.$$

For simplicity we ignore the argument (x^*, y^*). With the similarity transformation $P^{-1}PHP = HP$, we know that PH has the same spectrum as HP, and also its transpose $(HP)^\top = P^\top H$.

The optimal convergence rate is achieved by optimizing $\max_i |1 - \alpha_{\text{L}}\lambda_i|$ and $\max_i |1 - \alpha_{\text{F}}\mu_i|$ respectively, which is achieved at $\alpha_{\text{L}} = 2/(\lambda_1 + \lambda_n)$ and $\alpha_{\text{F}} = 2/(\mu_1 + \mu_m)$. □

4.3 Newton-type Algorithms

After analyzing existing methods that locally converge to strict local minimax points, we now present our Newton-type methods. We propose Gradient Descent Newton (GDN), which updates x through gradient descent and y through a Newton step. This method solves the ill-conditioning problem of the follower y in terms of local convergence. If we further replace the gradient descent step of x with a Newton step, then we obtain complete Newton (CN), which solves the ill-conditioning problem of both x and y. At the same time, the computation complexity of GDN and CN are similar to TGDA and FR (see Table 4.1).

We assume that the partial Hessians are Lipschitz continuous in our proofs. This is standard in conventional Newton methods for minimization. Based on this assumption, we can derive the constants of Lipschitzness and boundedness for first- and second-order derivatives (see Appendix C.1).

Assumption 4.3.1 (Lipschitz Hessian). *There exist constants L_{xx}, L_{xy}, L_{yy} such that for any $x_1, x_2 \in \mathcal{N}(x^*)$ and $y_1, y_2 \in \mathcal{N}(y^*)$, we have*

$$\begin{aligned}\|\partial^2_{xx} f(z_1) - \partial^2_{xx} f(z_2)\| &\le L_{xx}\|z_1 - z_2\|,\\ \|\partial^2_{xy} f(z_1) - \partial^2_{xy} f(z_2)\| &\le L_{xy}\|z_1 - z_2\|,\\ \|\partial^2_{yy} f(z_1) - \partial^2_{yy} f(z_2)\| &\le L_{yy}\|z_1 - z_2\|,\end{aligned}$$

with $z_i = (x_i, y_i)$ for $i = 1, 2$.

At a local neighborhood of (x^*, y^*), we can also assume that the second-order derivatives are bounded, i.e., for any $(x, y) \in \mathcal{N}(x^*) \times \mathcal{N}(y^*)$,

$$\|\partial^2_{xx} f(x, y)\| \le B_{xx}, \|\partial^2_{xy} f(x, y)\| \le B_{xy}, \|\partial^2_{yy} f(x, y)\| \le B_{yy}. \tag{4.45}$$

This is because we assumed that $f \in \mathcal{C}^2$. With Assumption 4.3.1 and $f \in \mathcal{C}^2$, we can also derive the Lipschitz constants and boundedness constants of first-order derivatives. More details about these constants and the derivation can be found in Appendix C.1.

4.3.1 Gradient Descent Newton

We propose our first Newton-based algorithm (GDN) for solving the nonconvex-nonconcave minimax problem (2.7), and make connections and comparisons to existing algorithms. In the GDA algorithm, the follower takes one gradient ascent step to approximate the best response function. However, such step might be insufficient for the approximation. Instead, we use a Newton step to approximate the local best response function $r(x)$, which is also more appealing if the inner maximization is ill-conditioned.

Many existing algorithms, including GDN, are based on a classic idea that goes back to Uzawa (Arrow et al., 1958): we employ iterative algorithms `F` and `L` for the follower y and leader x, respectively. The key is to allow `F` to *adapt quickly* to the update in `L`. Naturally, we propose to apply gradient descent as `L` and Newton update as `F`:

$$\begin{aligned}x_{t+1} &= x_t - \alpha_{\texttt{L}} \cdot \partial_x f(x_t, y_t),\\ y_{t+1} &= y_t - ((\partial^2_{yy})^{-1} \cdot \partial_y) f(x_{t+1}, y_t),\end{aligned} \tag{4.46}$$

Newton's method is affine invariant (Boyd and Vandenberghe, 2004, Section 9.5.1): under any invertible affine transformation, Newton's update remains essentially the same while gradient updates change drastically. Thus, for ill-conditioned follower problems (where the

largest and smallest eigenvalues of $(\partial^2_{yy} f(x^*, y^*))^{-1}$ differ significantly), we expect Newton's algorithm to behave well while gradient algorithms will largely depend on the condition number.

Newton-CG method. Efficient implementations of Newton's algorithm have been actively explored in deep learning since Martens (2010). The product $((\partial^2_{yy})^{-1} \cdot \partial_y) f$ can be efficiently computed using conjugate gradient (CG) equipped with Hessian-vector products computed by autodiff. Complexity analysis of the Newton-CG method can be found in Royer et al. (2020) and references therein.

We now present the *non-asymptotic* local linear convergence rate of GDN to an SLmM. Our result is on the local convergence, and we need a good initialization that is close to the SLmM. To obtain a good initialization, in practice we consider the method of *pre-training and fine-tuning* (Hinton and Salakhutdinov, 2006), which we will discuss more at the end of §4.3.2 and implement in §4.4.

Define $\mathcal{B}(x^*, \delta_x) := \{x \in \mathbb{R}^n : \|x - x^*\|_2 \le \delta_x\}$, $\mathcal{B}(y^*, \delta_y) := \{y \in \mathbb{R}^n : \|y - y^*\|_2 \le \delta_y\}$ and $\mathcal{B}(z^*) = \mathcal{B}(x^*, \delta_x) \times \mathcal{B}(y^*, \delta_y)$, we have:

Theorem 4.3.2 (GD-Newton). *Given a SLmM (x^*, y^*) and $\delta_x > 0$, $\delta_y > 0$, suppose in the neighborhoods $\mathcal{N}(x^*) = \mathcal{B}(x^*, \delta_x)$ and $\mathcal{N}(y^*) = \mathcal{B}(y^*, \delta_y)$, Assumption 4.3.1 holds and the local best-response function $r : \mathcal{N}(x^*) \to \mathcal{N}(y^*)$ exists. Suppose $\mu_x I \preceq \mathtt{D}^2_{xx} f(x, y) \preceq M_x I$ for any $(x, y) \in \mathcal{N}(x^*) \times \mathcal{N}(y^*)$. Define:*

$$\mathcal{N}_{\mathrm{GDN}} := \left\{ z \in \mathbb{R}^{n+m} : \|x - x^*\| \le \delta, \|y - y^*\| \le 2V\delta \right\} \tag{4.47}$$

where

$$\delta = \min \left\{ \delta_x, \frac{\delta_y}{2V}, \frac{\rho_L}{4V^2 U}, \frac{\epsilon}{\alpha_L M (1 + 4V^2/\rho_L^2)} \right\} \tag{4.48}$$

and $\rho_L = |1 - \alpha_L \mu_x| \vee |1 - \alpha_L M_x|, 0 < \epsilon \le 1 - \rho_L$, and U, V, M satisfy:

$$U := L_{yy}(2\mu_y)^{-1}, V := (B_y L_{yy} + \mu_y L_y)\mu_y^{-2}, M := (B_{xy} + L_x^{\mathtt{D}}) L_{yy} (2\mu_y^3)^{-1} L_y^2, \tag{4.49}$$

where μ_y, B_y, L_x, L_y, $L_x^{\mathtt{D}}$ are constants defined in Lemmas C.1.2 and C.1.5. Given an initialization $(x_1, y_1) \in \mathcal{N}_{\mathrm{GDN}}$, $\|y_1 - y^*\| \le 2V\|x_1 - x^*\|$ and suppose that $(x_2, y_2) \in \mathcal{N}_{\mathrm{GDN}}$ and $\|y_2 - y^*\| \le 2V\|x_2 - x^*\|$, the convergence of GD-Newton to (x^*, y^*) is linear, i.e., *for any $t \ge 2$, we have:*

$$\|x_{t+1} - x^*\| \le (\rho_L + \epsilon)^{t-1} \|x_2 - x^*\|, \ \|y_{t+1} - y^*\| \le 2V(\rho_L + \epsilon)^{t-1} \|x_2 - x^*\|. \tag{4.50}$$

The definition of δ in (4.47) tells us that when ϵ is small, the second term dominates. This means that if we want a better local convergence rate we need to be closer to the SLmM. Also, a smaller $\alpha_{\mathtt{L}}$ can control the neighborhood and thus GDN becomes more stable.

Before we move on to the proof, we observe the dependence of the neighborhood on condition numbers. In fact, for the GD-Newton method, there are two condition numbers. We denote

$$\kappa_{1,y} := \frac{L_y}{\mu_y},\ \kappa_{2,y} := \frac{L_{yy}}{\mu_y}. \tag{4.51}$$

$\kappa_{1,y}$ is the usual condition number when we study first order algorithms. For Newton-type algorithms, $\kappa_{2,y}$ arises (Prop. 1.4.1, Bertsekas (1997)). From (4.49), and Lemma C.1.2, the absolute constants can be written as:

$$U = \kappa_{2,y}/2,\ V = \kappa_{1,y}\kappa_{2,y}(\delta_x + \delta_y) + \kappa_{1,y},\ M = (B_{xy} + L_x^{\mathtt{D}})\kappa_{2,y}\kappa_{1,y}^2/2. \tag{4.52}$$

Namely, the neighborhood size depends on the condition numbers. This is not uncommon in conventional minimization, for both first- and second-order algorithms (e.g. Nesterov (2003), Theorems 1.2.4 and 1.2.5).

We also note that the constant M_x in $\mu_x I \preceq \mathtt{D}_{xx}^2 f(z) \preceq M_x I$ can be taken to be $B_{xx}^{\mathtt{D}}$ as in Lemma C.1.5, because for any $z \in \mathcal{B}(z^*)$ and $x \in \mathbb{R}^n$ such that $\|x\| = 1$, we have $\|\mathtt{D}_{xx}^2 f(z)x\| \le B_{xx}^{\mathtt{D}}$, and thus from Cauchy–Schwarz inequality we have $x^\top \mathtt{D}_{xx}^2 f(z)x \le \|x\| \cdot \|\mathtt{D}_{xx}^2 f(z)x\| \le B_{xx}^{\mathtt{D}}$. Therefore, for any $z \in \mathcal{B}(z^*)$, we obtain $\mathtt{D}_{xx} f(z) \preceq B_{xx}^{\mathtt{D}} I$.

Proof techniques Our proof relies on two parts: the leader takes gradient descent on $\psi(x)$ with approximation error controlled by $r(x_t) - y_t$; the follower takes Newton updates to approximate the local best response $r(x_t)$ at each step. This reflects the sequential nature of the minimax game. The difficulty lies in how to bound the approximation errors.

Proof. Now let us study the exact convergence rate. We can prove that $\mathcal{N}_{\mathrm{GDN}} \subset \mathcal{B}(x^*, \delta_x) \times \mathcal{B}(x^*, \delta_y)$ because for any $z = (x, y) \in \mathcal{N}_{\mathrm{GDN}}$, we have $\|x - x^*\| \le \delta \le \delta_x$ and $\|y - y^*\| \le 2V\delta \le 2V \cdot \frac{\delta_y}{2V} = \delta_y$. Hence, all our results in Appendix C.1 are valid on $\mathcal{N}_{\mathrm{GDN}}$.

Suppose $(x_k, y_k) \in \mathcal{N}_{\mathrm{GDN}}$ for $k \le t$ and $t \ge 2$. We first prove that:

$$\|y_{t+1} - y^*\| \le U\|y_t - y^*\|^2 + V\|x_{t+1} - x^*\|, \tag{4.53}$$

and then

$$\|x_{t+1} - x^*\| \le \rho_{\mathtt{L}}\|x_t - x^*\| + \alpha_{\mathtt{L}} M(\|x_t - x^*\|^2 + \|y_{t-1} - y^*\|^2), \tag{4.54}$$

where

$$\rho_{\mathtt{L}} = |1 - \alpha_{\mathtt{L}}\mu_x| \vee |1 - \alpha_{\mathtt{L}} M_x|,\ M = (B_{xy} + L_x^{\mathtt{D}})L_{yy}(2\mu_y^3)^{-1}L_y^2. \tag{4.55}$$

With these two inequalities, we will prove in Part III that for $t \geq 2$:

$$\|x_{t+1} - x^*\| \leq (\rho_{\mathtt{L}} + \epsilon)^{t-1}\|x_2 - x^*\|,\ \|y_{t+1} - y^*\| \leq 2V(\rho_{\mathtt{L}} + \epsilon)^{t-1}\|x_2 - x^*\|. \tag{4.56}$$

Part I To prove (4.53), note that

$$\begin{aligned}
\|y_{t+1} - y^*\| &= \|y_t - y^* - ((\partial_{yy}^2)^{-1} \cdot \partial_y)f(x^*, y_t) \\
&\quad + ((\partial_{yy}^2)^{-1} \cdot \partial_y)f(x^*, y_t) - ((\partial_{yy}^2)^{-1} \cdot \partial_y)f(x_{t+1}, y_t)\| \\
&\leq \|(\partial_{yy}^2)^{-1} f(x^*, y_t)(\partial_{yy}^2 f(x^*, y_t)(y_t - y^*) - \partial_y f(x^*, y_t))\| + \\
&\quad + \|((\partial_{yy}^2)^{-1} \cdot \partial_y)f(x^*, y_t) - ((\partial_{yy}^2)^{-1} \cdot \partial_y)f(x_{t+1}, y_t)\|.
\end{aligned} \tag{4.57}$$

From the local Lipschitzness of $((\partial_{yy}^2)^{-1} \cdot \partial_y)f$, (C.31), we know that the second term is at most

$$(\mu_y^{-1}L_y + B_y\mu_y^{-2}L_{yy})\|x_{t+1} - x^*\| = V\|x_{t+1} - x^*\|.$$

Since we assumed that $(x_t, y_t) \in \mathcal{N}_{\mathrm{GDN}} \subset \mathcal{B}(z^*)$, we can derive that $(x^*, y_t) \in \mathcal{B}(z^*)$ and $y_t \in \mathcal{N}(y^*)$. The first term can be upper bounded as:

$$\begin{aligned}
&\|(\partial_{yy}^2)^{-1} f(x^*, y_t)\| \cdot \|(\partial_{yy}^2 f(x^*, y_t)(y_t - y^*) - \partial_y f(x^*, y_t) + \partial_y f(x^*, y^*)\| \\
&\leq \mu_y^{-1} \cdot \|(\partial_{yy}^2 f(x^*, y_t)(y_t - y^*) - \partial_y f(x^*, y_t) + \partial_y f(x^*, y^*)\| \\
&\leq \mu_y^{-1} \int_0^1 \|\partial_{yy}^2 f(x^*, y_t) - \partial_{yy}^2 f(x^*, y^* + s(y_t - y^*))\| \cdot \|y_t - y^*\| ds \\
&\leq \mu_y^{-1} \int_0^1 L_{yy}(1 - s)\|y_t - y^*\|^2 ds \\
&= L_{yy}(2\mu_y)^{-1}\|y_t - y^*\|^2 \\
&= U\|y_t - y^*\|^2,
\end{aligned} \tag{4.58}$$

where in the second line we used Lemma C.1.1; in the third line we used the following identity:

$$\partial_y f(x, y_1) - \partial_y f(x, y_2) = \int_0^1 \partial_{yy}^2 f(x, y_2 + s(y_1 - y_2))(y_1 - y_2) ds, \tag{4.59}$$

and in the fourth line we used Assumption 4.3.1. Therefore we have proved (4.53).

Part II To prove (4.54), we observe that:

$$\begin{aligned}\|x_{t+1}-x^*\| &= \|x_t - x^* - \alpha_{\mathtt{L}}\partial_x f(x_t,y_t)\| \\ &= \|x_t - x^* - \alpha_{\mathtt{L}}\mathtt{D}_x f(x_t,r(x_t)) + \alpha_{\mathtt{L}}(\mathtt{D}_x f(x_t,r(x_t)) - \mathtt{D}_x f(x_t,y_t)) \\ &+ \alpha_{\mathtt{L}}(\mathtt{D}_x f(x_t,y_t) - \partial_x f(x_t,y_t))\| \\ &= \|x_t - x^* - \alpha_{\mathtt{L}}\mathtt{D}_x f(x_t,r(x_t))\| + \alpha_{\mathtt{L}}\|\mathtt{D}_x f(x_t,r(x_t)) - \mathtt{D}_x f(x_t,y_t)\| \\ &+ \alpha_{\mathtt{L}}\|\mathtt{D}_x f(x_t,y_t) - \partial_x f(x_t,y_t)\|. \end{aligned} \tag{4.60}$$

Note that $r(x_t) \in \mathcal{N}(y^*)$ because of (4.7). So $(x_t, r(x_t)) \in \mathcal{B}(z^*)$ and our analysis is valid. Now let us bound the three terms separately. The first term can be computed as

$$\begin{aligned}\|x_t - x^* - \alpha_{\mathtt{L}}\mathtt{D}_x f(x_t,r(x_t))\| &= \|x_t - x^* - \alpha_{\mathtt{L}}(\psi'(x_t) - \psi'(x^*))\| \\ &= \|x_t - x^* - \alpha_{\mathtt{L}}\int_0^1 \psi''(x^* + s(x_t - x^*))(x_t - x^*)ds\| \\ &= \|\int_0^1 (I - \alpha_{\mathtt{L}}\psi''(x^* + s(x_t - x^*)))(x_t - x^*)ds\| \\ &\leq \int_0^1 \|I - \alpha_{\mathtt{L}}\psi''(x^* + s(x_t - x^*))\| \cdot \|x_t - x^*\|ds \\ &\leq \rho_{\mathtt{L}}\|x_t - x^*\|, \end{aligned} \tag{4.61}$$

where in the first line we used from Lemma C.1.6, $\psi'(x) = \mathtt{D}_x f(x, r(x))$ and $\psi'(x^*) = \mathtt{D}_x f(x^*, r(x^*)) = \mathbf{0}$. In the last line, we used from Lemma C.1.6 that $\psi''(x) = \mathtt{D}^2_{xx} f(x, r(x))$ and our assumption $\mu_x I \preceq \mathtt{D}^2_{xx} f(z) \preceq M_x I$ for any $z \in \mathcal{B}(z^*)$. More specifically, since

$$x_s = x^* + s(x_t - x^*) \in \mathcal{N}(x^*), \tag{4.62}$$

we have

$$I - \alpha_{\mathtt{L}}\psi''(x_s) = I - \alpha_{\mathtt{L}}\mathtt{D}^2_{xx}(x_s, r(x_s)), \tag{4.63}$$

$$(1 - \alpha\mathtt{L}M_x)I \preceq I - \alpha_{\mathtt{L}}\mathtt{D}^2_{xx}(x_s, r(x_s)) \preceq (1 - \alpha\mathtt{L}\mu_x)I. \tag{4.64}$$

Therefore,

$$\|I - \alpha_{\mathtt{L}}\mathtt{D}^2_{xx}(x_s, r(x_s))\| \leq |1 - \alpha_{\mathtt{L}}\mu_x| \vee |1 - \alpha_{\mathtt{L}}M_x| = \rho_{\mathtt{L}}. \tag{4.65}$$

From Lemma C.1.5 the second term can be bounded as:

$$\alpha_{\mathtt{L}}\|\mathtt{D}_x f(x_t, r(x_t)) - \mathtt{D}_x f(x_t, y_t)\| \leq \alpha_{\mathtt{L}} L^{\mathtt{D}}_x \|r(x_t) - y_t\|, \tag{4.66}$$

From the definition of $\mathtt{D}_x f := \partial_x f - \partial^2_{xy} f \cdot (\partial^2_{yy} f)^{-1} \cdot \partial_y f$, the third term can be bounded as:

$$\begin{aligned}\alpha_{\mathtt{L}} \|\mathtt{D}_x f(z_t) - \partial_x f(z_t)\| &= \alpha_{\mathtt{L}} \|(\partial^2_{xy} \cdot (\partial^2_{yy})^{-1} \cdot \partial_y) f(z_t)\| \\ &\le \alpha_{\mathtt{L}} \|\partial^2_{xy} f(z_t)\| \cdot \|(\partial^2_{yy} f(z_t))^{-1}\| \cdot \|\partial_y f(z_t)\| \\ &\le \alpha_{\mathtt{L}} B_{xy} \mu_y^{-1} \|\partial_y f(z_t)\|, \end{aligned} \tag{4.67}$$

where we used Lemma C.1.2 and the assumption $z_t \in \mathcal{B}(z^*)$ from induction. To upper bound $\|\partial_y f(z_t)\|$, note that:

$$\|\partial_y f(z_t)\| = \|\partial_y f(x_t, y_t)\| = \|\partial_y f(x_t, y_{t-1} - \Delta y)\|, \tag{4.68}$$

with $\Delta y = ((\partial^2_{yy})^{-1} \cdot \partial_y) f(x_t, y_{t-1})$. Therefore,

$$\begin{aligned}\|\partial_y(z_t)\| = \|\partial_y f(x_t, y_{t-1} - \Delta y)\| &= \|\partial_y f(x_t, y_{t-1} - \Delta y) - \partial_y f(x_t, y_{t-1}) - \partial^2_{yy} f(x_t, y_{t-1})(-\Delta y)\| \\ &= \| \int_0^1 \left(\partial^2_{yy} f(x_t, y_{t-1} - s\Delta y) - \partial^2_{yy} f(x_t, y_{t-1})\right)(-\Delta y) ds\| \\ &\le \int_0^1 \| \left(\partial^2_{yy} f(x_t, y_{t-1} - s\Delta y) - \partial^2_{yy} f(x_t, y_{t-1})\right)(-\Delta y)\| ds \\ &\le \int_0^1 \|\partial^2_{yy} f(x_t, y_{t-1} - s\Delta y) - \partial^2_{yy} f(x_t, y_{t-1})\| \cdot \|(-\Delta y)\| ds \\ &\le \int_0^1 L_{yy} s \|\Delta y\|^2 ds \\ &= \tfrac{1}{2} L_{yy} \|\Delta y\|^2 \\ &= \tfrac{1}{2} L_{yy} \|((\partial^2_{yy})^{-1} \cdot \partial_y) f(x_t, y_{t-1})\|^2 \\ &\le \tfrac{1}{2} L_{yy} \|(\partial^2_{yy})^{-1} f(x_t, y_{t-1})\|^2 \cdot \|\partial_y f(x_t, y_{t-1})\|^2 \\ &\le L_{yy} (2\mu_y^2)^{-1} \|\partial_y f(x_t, y_{t-1}) - \partial_y f(x^*, y^*)\|^2 \\ &\le L_{yy} (2\mu_y^2)^{-1} L_y^2 \left(\|x_t - x^*\|^2 + \|y_{t-1} - y^*\|^2\right), \end{aligned} \tag{4.69}$$

where in the second line we used (4.59); in the fifth line we used Assumption 4.3.1; in the seventh line we used the definition of Δy; in the second last line we used $\|(\partial^2_{yy})^{-1} f(z)\| \le \mu_y^{-1}$ any $z \in \mathcal{B}(z^*)$ from Lemma C.1.2 and $\partial_y f(z^*) = \mathbf{0}$; in the last line we used the Lipschitz condition in Lemma C.1.2. Note that $z_t, z_{t-1} \in \mathcal{B}(z^*)$, and thus $y_{t-1} \in \mathcal{N}(y^*)$, $x_t \in \mathcal{N}(x^*)$ and $(x_t, y_{t-1}) \in \mathcal{B}(z^*)$. So all our discussion is within the neighborhood $\mathcal{B}(z^*)$ and thus valid. On the other hand, from $\partial^2_{yy} f(z) \preceq -\mu_y I$ for all $z \in \mathcal{B}(z^*)$, as in Lemma

C.1.2, and the Cauchy-Schwarz inequality, we obtain:

$$\begin{aligned}
&\|r(x_t) - y_t\| \cdot \|\partial_y f(x_t, r(x_t)) - \partial_y f(x_t, y_t)\| \\
&\geq -(r(x_t) - y_t)^\top (\partial_y f(x_t, r(x_t)) - \partial_y f(x_t, y_t)) \\
&= -(r(x_t) - y_t)^\top \partial^2_{yy} f(x_t, y_\xi)(r(x_t) - y_t) \\
&\geq \mu_y \|r(x_t) - y_t\|^2,
\end{aligned} \tag{4.70}$$

where in the third line we used the mean-value theorem and that y_ξ is on the line segment with y_t and $r(x_t)$ as two endpoints; in the fourth line we used the definition of μ_y in Lemma C.1.2. Therefore, from (4.70), $\partial_y f(x_t, r(x_t)) = \mathbf{0}$ and (4.69) we obtain:

$$\begin{aligned}
\|r(x_t) - y_t\| &\leq \mu_y^{-1} \|\partial_y f(x_t, r(x_t)) - \partial_y f(x_t, y_t)\| \\
&= \mu_y^{-1} \|\partial_y f(x_t, y_t)\| \\
&\leq L_{yy} (2\mu_y^3)^{-1} L_y^2 \left(\|x_t - x^*\|^2 + \|y_{t-1} - y^*\|^2 \right).
\end{aligned} \tag{4.71}$$

Combining (4.60), (4.61), (4.66), (4.67) we obtain that:

$$\begin{aligned}
\|x_{t+1} - x^*\| &\leq \rho_{\mathtt{L}} \|x_t - x^*\| + \alpha_{\mathtt{L}} L_x^{\mathtt{D}} \|r(x_t) - y_t\| + \alpha_{\mathtt{L}} B_{xy} \mu_y^{-1} \|\partial_y f(z_t)\| \\
&\leq \rho_{\mathtt{L}} \|x_t - x^*\| + \alpha_{\mathtt{L}} (B_{xy} + L_x^{\mathtt{D}}) \mu_y^{-1} \|\partial_y f(z_t)\| \\
&\leq \rho_{\mathtt{L}} \|x_t - x^*\| + \alpha_{\mathtt{L}} M (\|x_t - x^*\|^2 + \|y_{t-1} - y^*\|^2),
\end{aligned} \tag{4.72}$$

where in the second line we used (4.71) and in the third line we used (4.69), and

$$M = (B_{xy} + L_x^{\mathtt{D}}) L_{yy} (2\mu_y^3)^{-1} L_y^2. \tag{4.73}$$

Part III Denote $a_t = \|x_t - x^*\|$ and $b_t = \|y_t - y^*\|$, we have proved the following claim in Part I and Part II:

Claim 4.3.3. *Suppose for $t \geq 2$, if $\{z_k\}_{k=1}^t \subset \mathcal{N}_{\mathrm{GDN}}$, then we have:*

$$a_{t+1} \leq \rho_{\mathtt{L}} a_t + M(a_t^2 + b_{t-1}^2),\ b_{t+1} \leq U b_t^2 + V a_{t+1}. \tag{4.74}$$

Suppose now that $z_t \in \mathcal{N}_{\mathrm{GDN}}$ for any $1 \leq t \leq T$, let us prove $z_{T+1} \in \mathcal{N}_{\mathrm{GDN}}$. From Claim 4.3.3 we know that (4.74) holds for all $t = 2, \cdots, T$. Define the upper bounding sequence $\{\bar{a}_k\}_{k=1}^{T+1}$ and $\{\bar{b}_k\}_{k=1}^{T+1}$ such that $\bar{a}_i = a_i$ and for $i = 1, 2$, and

$$\bar{a}_{t+1} = \rho_{\mathtt{L}} \bar{a}_t + M(\bar{a}_t^2 + \bar{b}_{t-1}^2), \bar{b}_{t+1} = U \bar{b}_t^2 + V \bar{a}_{t+1},\ \text{for } t = 2, \ldots, T. \tag{4.75}$$

One can show that for any $1 \le t \le T+1$, we have:

$$a_t \le \bar{a}_t,\ b_t \le \bar{b}_t, \tag{4.76}$$

which follows from induction. To prove $z_{T+1} \in \mathcal{N}_{\text{GDN}}$, it suffices to show that for any $t = 2, \ldots, T+1$, we have:

$$\bar{b}_t \le 2V\bar{a}_t, \bar{a}_t \le \delta, \tag{4.77}$$

which is true for $t = 1, 2$ from our assumption that $\bar{a}_i = a_i$ and $b_i = b_i$ for $i = 1, 2$ and the definition of $\mathcal{N}_{\text{GDN}}$. This is because we can simply apply (4.77) for $t = T+1$ and use (4.76). Suppose (4.77) holds for $k \le t$ and $t \ge 2$:

$$\bar{b}_k \le 2V\bar{a}_k, \bar{a}_k \le \delta \text{ for all } k \le t. \tag{4.78}$$

Taking $\bar{b}_t \le 2V\bar{a}_t$ from (4.78) we obtain:

$$\begin{aligned}
\bar{b}_{t+1} &= V\bar{a}_{t+1} + U\bar{b}_t^2 \\
&\le V\bar{a}_{t+1} + 4V^2U\bar{a}_t^2 \\
&\le V\left(1 + 4\frac{V^2U}{\rho_{\text{L}}}\bar{a}_t\right)\bar{a}_{t+1} \\
&\le 2V\bar{a}_{t+1},
\end{aligned} \tag{4.79}$$

where in the third line we used $\rho_{\text{L}}\bar{a}_t \le \bar{a}_{t+1}$ that can be derived from (4.75); in the last line we used the assumption in (4.78) and $\bar{a}_t \le \delta \le \frac{\rho_{\text{L}}}{4V^2U}$. Also, from (4.75), we have

$$\begin{aligned}
\bar{a}_{t+1} &= \rho_{\text{L}}\bar{a}_t + M(\bar{a}_t^2 + \bar{b}_{t-1}^2) \\
&\le \rho_{\text{L}}\bar{a}_t + M(\bar{a}_t^2 + 4V^2\bar{a}_{t-1}^2) \\
&\le \rho_{\text{L}}\bar{a}_t + M\left(1 + \frac{4V^2}{\rho_{\text{L}}^2}\right)\bar{a}_t^2 \\
&= \left(\rho_{\text{L}} + M\left(1 + \frac{4V^2}{\rho_{\text{L}}^2}\right)\bar{a}_t\right)\bar{a}_t \\
&\le (\rho_{\text{L}} + \epsilon)\bar{a}_t \qquad (4.80) \\
&\le \bar{a}_t \le \delta, \qquad (4.81)
\end{aligned}$$

where in the second line, we used $b_{t-1} \le 2V\bar{a}_{t-1}$ as in (4.78); in the third line, we used $\bar{a}_t \ge \rho_{\text{L}}\bar{a}_{t-1}$ which can be derived from (4.75); in the second last line we used the assumption in (4.78) that $\bar{a}_t \le \delta \le \frac{\epsilon}{M(1+(4V^2/\rho_{\text{L}}))}$; in the last line we used $0 < \epsilon < 1 - \rho_{\text{L}}$. By induction, we have proved that for any $t = 2, \ldots, T+1$, we have (4.77) and thus $z_{T+1} \in \mathcal{N}_{\text{GDN}}$.

So far, we have proved that for any $t \geq 1$, $z_t \in \mathcal{N}_{\text{GDN}}$. This implies that for any $t \geq 2$, (4.74) is true. Taking the upper bounding sequence again as in (4.76). We have in fact proved from (4.79) and (4.80) that for any $t \geq 2$,

$$\bar{a}_{t+1} \leq (\rho_{\text{L}} + \epsilon)\bar{a}_t,\ \bar{b}_{t+1} \leq 2V\bar{a}_{t+1}. \tag{4.82}$$

Therefore, we obtain from the above that for $t \geq 2$:

$$\bar{a}_{t+1} \leq (\rho_{\text{L}} + \epsilon)^{t-1}\bar{a}_2 = (\rho_{\text{L}} + \epsilon)^{t-1}a_2, \tag{4.83}$$

and thus for any $t \geq 2$:

$$\begin{aligned}
&\|x_{t+1} - x^*\| = a_{t+1} \leq \bar{a}_{t+1} \leq (\rho_{\text{L}} + \epsilon)^{t-1}a_2 = (\rho_{\text{L}} + \epsilon)^{t-1}\|x_2 - x^*\|, \\
&\|y_{t+1} - y^*\| = b_{t+1} \leq \bar{b}_{t+1} \leq 2V\bar{a}_{t+1} \leq 2V(\rho_{\text{L}} + \epsilon)^{t-1}a_2 = 2V(\rho_{\text{L}} + \epsilon)^{t-1}\|x_2 - x^*\|.
\end{aligned} \tag{4.84}$$

□

In fact, GDN is an approximation of Uzawa's approach:

$$x_{t+1} = x_t - \alpha_{\text{L}} \cdot \partial_x f(x_t, y_t),\ y_{t+1} = r(x_{t+1}), \tag{4.85}$$

where recall that r is the local best response. The update (4.85) is essentially the original proposal by Uzawa (Arrow et al., 1958) as also in e.g. Fiez et al. (2019, Sec. 3.1) and Jin et al. (2020, Sec. 4) for different settings. In Theorem 4.2.2 we will see another approach to approximate (4.85).

As expected, the condition number of the follower Hessian $\partial^2_{yy} f$ has no effect on the local convergence rate of GDN thanks to the Newton update on y. When $\alpha_{\text{L}} = 2/(\mu_x + M_x)$, ρ_{L} is minimized to be

$$\rho_{\text{L}} = \frac{\kappa_{\text{L}} - 1}{\kappa_{\text{L}} + 1} \text{ where } \kappa_{\text{L}} = M_x/\mu_x \text{ is the condition number.}$$

The condition number κ_{L} of the leader problem does appear, since GDN still employs a gradient update for the leader x. We will see how to remove this dependence in §4.3.2.

To fully appreciate our method, we make comparisons between GDN and existing alternative algorithms and reveal interesting connections. Note that in Theorem 4.3.2, if we take $\epsilon \to 0$, we obtain an asymptotic (local) linear convergence rate ρ_{L}:

$$\rho_{\text{L}} = |1 - \alpha_{\text{L}}\lambda_1| \vee |1 - \alpha_{\text{L}}\lambda_n|, \tag{4.86}$$

with λ_1 and λ_n being the largest and smallest eigenvalues of $\mathtt{D}^2_{xx} f(z^*)$.

Compared to TGDA and FR in Thm. 4.3.2, the local convergence of GDN is always faster, especially when the follower problem is ill-conditioned (*i.e.* when $\kappa_{\mathtt{F}}$ is large compared to $\kappa_{\mathtt{L}}$), a point that we will verify in our experiments.

Comparing Thm. 4.2.2 with (4.86), we find that GDN and GDA-∞ share the same local convergence rate, confirming that when sufficiently close to an optimum, a single Newton step is as good as solving the problem exactly. When μ_1 is large (meaning the follower problem has a sharp curvature), we have to use a small step size α for updating the leader, and the resulting rate can be slower than GDN. Similar to 2TS-GDA, it is hard to gauge how many GA steps we need to approximate the exact algorithm (4.85) sufficiently well. When the follower problem is ill-conditioned, the number of GA steps may grow excessively large and we have to use a small step size α to ensure convergence.

4.3.2 Complete Newton

Although our first Newton-type algorithm, GDN, evades possible ill-conditioning of the follower problem, it may still converge slowly if the leader problem is ill-conditioned, *i.e.*, the largest and the smallest eigenvalues of $\mathtt{D}^2_{xx} f$ differ significantly. We propose a new Newton-type algorithm that evades ill-conditioning of both leader and follower problems, and locally converges super-linearly to an SLmM. With total second-order derivatives, we replace the gradient update of the leader in GDN with a Newton update, which we call the *Complete Newton* (CN) method:

$$\begin{aligned} x_{t+1} &= x_t - ((\mathtt{D}^2_{xx})^{-1} \cdot \partial_x) f(x_t, y_t), \\ y_{t+1} &= y_t - ((\partial^2_{yy})^{-1} \cdot \partial_y) f(x_{t+1}, y_t). \end{aligned} \tag{4.87}$$

CN is a *genuine* second-order method that (we prove below) achieves a super-linear rate, as compared to other methods in Section 4.2.2 that use the Hessian inverse. The Newton update $(\mathtt{D}^2_{xx})^{-1} f \cdot \partial_x f = \left(\partial^2_{xx} - \partial^2_{xy}(\partial^2_{yy})^{-1}\partial^2_{yx}\right)^{-1} f \cdot \partial_x f$ can be efficiently implemented as solving a single linear system of size $(m+n) \times (m+n)$ (see Lemma 4.3.4):

Lemma 4.3.4. *If D and $S := A - BD^{-1}C$ are invertible, then the matrix $[A\,B; C\,D]$ is invertible, with:*

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix}^{-1} = \begin{bmatrix} S^{-1} & -S^{-1}BD^{-1} \\ -D^{-1}CS^{-1} & D^{-1} + D^{-1}CS^{-1}BD^{-1} \end{bmatrix}. \tag{4.88}$$

Proof. Multiply $[A, B; C, D]$ with the right hand side of (4.88) and use simple algebra. □

$$\begin{bmatrix}\partial^2_{xx}f & \partial^2_{xy}f\\ \partial^2_{yx}f & \partial^2_{yy}f\end{bmatrix}\begin{bmatrix}\Delta x\\ \Delta v\end{bmatrix}=\begin{bmatrix}\partial_x f\\ \mathbf{0}\end{bmatrix}\iff$$
$$\Delta x=\begin{bmatrix}I & \mathbf{0}\end{bmatrix}\begin{bmatrix}\partial^2_{xx}f & \partial^2_{xy}f\\ \partial^2_{yx}f & \partial^2_{yy}f\end{bmatrix}^{-1}\begin{bmatrix}\partial_x f\\ \mathbf{0}\end{bmatrix}=((\mathtt{D}^2_{xx})^{-1}\cdot\partial_x)f.$$

As a result, when $m\approx n$, CN has the same complexity as TGDA, FR and GDN, which all use second order information (Table 4.1).

However, only CN enjoys the following local *quadratic* convergence rate:

Theorem 4.3.5 (Complete Newton). *Given a SLmM (x^*,y^*) and $\delta_x>0$, $\delta_y>0$, suppose in the neighborhoods $\mathcal{N}(x^*)=\mathcal{B}(x^*,\delta_x)$ and $\mathcal{N}(y^*)=\mathcal{B}(y^*,\delta_y)$, Assumption 4.3.1 holds and the local best-response function $r:\mathcal{N}(x^*)\to\mathcal{N}(y^*)$ exists. Define the neighborhood $\mathcal{N}_{\text{CN}}$ as*

$$\mathcal{N}_{\text{CN}}:=\{z\in\mathbb{R}^{n+m}:\|z-z^*\|\le\min\{\delta_x,\delta_y,\tfrac{1}{3L}\}\},$$

where

$$L=U+(V+1)(\tfrac{1}{2}\mu_x^{-1}L^{\psi}_{xx}+W),\tag{4.89}$$

Here U, V are the same as in Theorem 4.3.2 and

$$W:=(L_x\mu_x^{-1}+B_x\mu_x^{-2}L^{\mathtt{D}}_{xx})L_{yy}(2\mu_y^3)^{-1}L_y^2.\tag{4.90}$$

μ_x,μ_y, B_x, B_y, L_x, L_y, $L^{\mathtt{D}}_x$, $L^{\mathtt{D}}_{xx}$, L^{ψ}_{xx}, L_{yy} are defined in Lemmas C.1.2, C.1.5, C.1.6 and C.1.7. The local convergence of CN to $z^*=(x^*,y^*)$ is at least quadratic, i.e.:

$$\|z_t-z^*\|\le\frac{1}{2L}\max\{2L\|z_1-z^*\|,2L\|z_2-z^*\|\}^{2^{\lfloor(t-1)/2\rfloor}}.\tag{4.91}$$

with the initializations $z_1\in\mathcal{N}_{\text{CN}}, z_2\in\mathcal{N}_{\text{CN}}$.

Before we move on to the proof. We first interpret the constant L. In (4.51) we defined the condition numbers of y as:

$$\kappa_{1,y}:=\frac{L_y}{\mu_y},\ \kappa_{2,y}:=\frac{L_{yy}}{\mu_y}.\tag{4.92}$$

from which the absolute constants U, V can be written as:

$$U=\kappa_{2,y}/2,\ V=\kappa_{1,y}\kappa_{2,y}(\delta_x+\delta_y)+\kappa_{1,y}.\tag{4.93}$$

Similarly, we define the condition numbers on x as:

$$\kappa_{1,x} := \frac{L_x}{\mu_x},\ \kappa_{2,x}^{\mathtt{D}} := \frac{L_{xx}^{\mathtt{D}}}{\mu_x},\ \kappa_{2,x}^{\psi} := \frac{L_{xx}^{\psi}}{\mu_x}, \tag{4.94}$$

and using Lemma C.1.2, (4.90) can be written as:

$$W = (\kappa_{1,x} + (\delta_x + \delta_y)\kappa_{1,x}\kappa_{2,x}^{\mathtt{D}})\kappa_{2,y}\kappa_{1,y}^2/2. \tag{4.95}$$

Putting everything together, (4.89) becomes:

$$L = \frac{\kappa_{2,y}}{2} + \frac{1}{2}(\kappa_{1,y}\kappa_{2,y}(\delta_x + \delta_y) + \kappa_{1,y} + 1)(\kappa_{2,x}^{\psi} + (\kappa_{1,x} + (\delta_x + \delta_y)\kappa_{1,x}\kappa_{2,x}^{\mathtt{D}})\kappa_{2,y}\kappa_{1,y}^2). \tag{4.96}$$

This interpretation shows us that the size of the neighborhood $\mathcal{N}_{\text{CN}}$ that guarantees the local quadratic convergence can be very small, since W is a product of condition numbers on both x and y. The dependence of the neighborhood on condition numbers is not uncommon in conventional minimization, for both first- and second-order algorithms (e.g. Nesterov (2003), Theorems 1.2.4 and 1.2.5).

Proof. We assume first that $z_k \in \mathcal{N}_{\text{CN}}$ for $k \leq t$ and $t \geq 2$. Note that $\mathcal{N}_{\text{CN}} \subset \mathcal{B}(z^*)$ because for any $(x, y) \in \mathcal{N}_{\text{CN}}$,

$$\|x - x^*\| \leq \|z - z^*\| \leq \delta_x,\ \|y - y^*\| \leq \|z - z^*\| \leq \delta_y. \tag{4.97}$$

This satisfies our definition of $\mathcal{B}(z^*) = \mathcal{B}(x^*, \delta_x) \times \mathcal{B}(y^*, \delta_y)$ in (C.2). $\mathcal{N}_{\text{CN}} \subset \mathcal{B}(z^*)$ tells us that we can use all the local Lipschitzness and boundedness results in Appendix C.1.

Since the update of y is the same as GDN, we can borrow (4.53) to have:

$$\|y_{t+1} - y^*\| \leq U\|y_t - y^*\|^2 + V\|x_{t+1} - x^*\|. \tag{4.98}$$

We prove next that:

$$\|x_{t+1} - x^*\| \ \ \leq (\tfrac{1}{2}\mu_x^{-1}L_{xx}^{\psi} + W)\|x_t - x^*\|^2 + W\|y_{t-1} - y^*\|^2. \tag{4.99}$$

where

$$W := (L_x\mu_x^{-1} + B_x\mu_x^{-2}L_{xx}^{\mathtt{D}})L_{yy}(2\mu_y^3)^{-1}L_y^2. \tag{4.100}$$

Part I To prove (4.99), we note that:

$$\begin{aligned}\|x_{t+1}-x^*\| &= \|x_t-x^*-((\mathtt{D}^2_{xx})^{-1}\cdot\partial_x)f(x_t,r(x_t))+((\mathtt{D}^2_{xx})^{-1}\cdot\partial_x)f(x_t,r(x_t))-\\ &- ((\mathtt{D}^2_{xx})^{-1}\cdot\partial_x)f(x_t,y_t)\|\\ &\leq \|(\mathtt{D}^2_{xx})^{-1}f(x_t,r(x_t))(\mathtt{D}^2_{xx}f(x,r(x_t))(x_t-x^*)-\partial_x f(x_t,r(x_t)))\|+\\ &+ \|((\mathtt{D}^2_{xx})^{-1}\cdot\partial_x)f(x_t,r(x_t))-((\mathtt{D}^2_{xx})^{-1}\cdot\partial_x)f(x_t,y_t)\|. \end{aligned}\tag{4.101}$$

We observe that $r(x_t)\in\mathcal{N}(y^*)$ because of (4.7). So $(x_t,r(x_t))\in\mathcal{B}(z^*)$ and our analysis is valid. The first term can be computed as:

$$\begin{aligned}&\|(\mathtt{D}^2_{xx})^{-1}f(x_t,r(x_t))(\mathtt{D}^2_{xx}f(x,r(x_t))(x_t-x^*)-\partial_x f(x_t,r(x_t)))\|\\ &\leq \|(\mathtt{D}^2_{xx})^{-1}f(x_t,r(x_t))\|\cdot\|(\mathtt{D}^2_{xx}f(x,r(x_t))(x_t-x^*)-\partial_x f(x_t,r(x_t)))\|\\ &\leq \mu_x^{-1}\|\psi''(x_t)(x_t-x^*)-\mathtt{D}_x f(x_t,r(x_t))+\mathtt{D}_x f(x^*,y^*)\|\\ &= \mu_x^{-1}\|\psi''(x_t)(x_t-x^*)-\psi'(x_t)+\psi'(x^*)\|\\ &= \mu_x^{-1}\|\psi''(x_t)(x_t-x^*)-\int_0^1\psi''(x^*+s(x_t-x^*))(x_t-x^*)ds\|\\ &= \mu_x^{-1}\|\int_0^1(\psi''(x_t)-\psi''(x^*+s(x_t-x^*)))(x_t-x^*)ds\|\\ &\leq \mu_x^{-1}\int_0^1\|\psi''(x_t)-\psi''(x^*+s(x_t-x^*))\|\cdot\|x_t-x^*\|ds\\ &\leq \mu_x^{-1}L^\psi_{xx}\int_0^1(1-s)\|x_t-x^*\|^2ds\\ &= \tfrac{1}{2}\mu_x^{-1}L^\psi_{xx}\|x_t-x^*\|^2, \end{aligned}\tag{4.102}$$

where in the third line we used that for $x\in\mathcal{N}(x^*)$, we have from $\partial_y f(x,r(x))=\mathbf{0}$ in (4.7):

$$\mathtt{D}_x f(x,r(x))=\partial_x f(x,r(x))-(\partial^2_{xy}f\cdot(\partial^2_{yy})^{-1}f\cdot\partial_y f)(x,r(x))=\partial_x f(x,r(x)),\tag{4.103}$$

and thus $\mathtt{D}_x f(x^*,y^*)=\partial_x f(x^*,y^*)=\mathbf{0}$; the fifth line we used that for $x_1,x_2\in\mathcal{N}(x^*)$:

$$\psi'(x_1)-\psi'(x_2)=\int_0^1\psi''(x_2+s(x_1-x_2))(x_1-x_2)ds,\tag{4.104}$$

and in the second last line we used Lemma C.1.6 and the definition of L^ψ_{xx} in (C.45).

From Lemma C.1.7 we know that on $\mathcal{B}(z^*)$, $(\mathtt{D}^2_{xx})^{-1}f:=(\mathtt{D}^2_{xx}f(\cdot))^{-1}$ is $\mu_x^{-2}L^{\mathtt{D}}_{xx}$-Lipschitz continuous and μ_x^{-1}-bounded. From Lemma C.1.2, we know that on $\mathcal{B}(z^*)$, $\partial_x f$ is L_x-Lipschitz continuous and B_x-bounded. Therefore, from Lemma C.1.4, $(\mathtt{D}^2_{xx})^{-1}f\cdot\partial_x f$ is

$(L_x\mu_x^{-1}+B_x\mu_x^{-2}L_{xx}^{\mathtt{D}})$ Lipschitz continuous. The second term of (4.101) can thus be bounded as:

$$\begin{aligned}(L_x\mu_x^{-1}+B_x\mu_x^{-2}L_{xx}^{\mathtt{D}})\|r(x_t)-y_t\| \;&\leq (L_x\mu_x^{-1}+B_x\mu_x^{-2}L_{xx}^{\mathtt{D}})L_{yy}(2\mu_y^3)^{-1}\times\\ &\quad L_y^2\left(\|x_t-x^*\|^2+\|y_{t-1}-y^*\|^2\right),\end{aligned}\tag{4.105}$$

where we used (4.71). Note that the update of y_t is the same for both GDN and CN. To avoid heavy notation, we define

$$W := (L_x\mu_x^{-1}+B_x\mu_x^{-2}L_{xx}^{\mathtt{D}})L_{yy}(2\mu_y^3)^{-1}L_y^2.\tag{4.106}$$

From (4.101), (4.102) and (4.105), we obtain that:

$$\begin{aligned}\|x_{t+1}-x^*\| &\leq \tfrac{1}{2}\mu_x^{-1}L_{xx}^{\psi}\|x_t-x^*\|^2+W\left(\|x_t-x^*\|^2+\|y_{t-1}-y^*\|^2\right)\\ &= (\tfrac{1}{2}\mu_x^{-1}L_{xx}^{\psi}+W)\|x_t-x^*\|^2+W\|y_{t-1}-y^*\|^2\end{aligned}\tag{4.107}$$

Part II So far, we have:

$$\|y_{t+1}-y^*\|\leq U\|y_t-y^*\|^2+V\|x_{t+1}-x^*\|,\tag{4.108}$$

and

$$\|x_{t+1}-x^*\| \;\leq (\tfrac{1}{2}\mu_x^{-1}L_{xx}^{\psi}+W)\|x_t-x^*\|^2+W\|y_{t-1}-y^*\|^2.\tag{4.109}$$

where

$$W=(L_x\mu_x^{-1}+B_x\mu_x^{-2}L_{xx}^{\mathtt{D}})L_{yy}(2\mu_y^3)^{-1}L_y^2.\tag{4.110}$$

Bringing (4.109) to (4.108), we obtain that:

$$\|y_{t+1}-y^*\|\leq U\|y_t-y^*\|^2+V(\tfrac{1}{2}\mu_x^{-1}L_{xx}^{\psi}+W)\|x_t-x^*\|^2+VW\|y_{t-1}-y^*\|^2,\tag{4.111}$$

With (4.109) and (4.111), we can prove:

$$\begin{aligned}\|z_{t+1}-z^*\| &\leq \|x_{t+1}-x^*\|+\|y_{t+1}-y^*\|\\ &\leq U\|y_t-y^*\|^2+(V+1)(\tfrac{1}{2}\mu_x^{-1}L_{xx}^{\psi}+W)\|x_t-x^*\|^2+(V+1)W\|y_{t-1}-y^*\|^2\\ &\leq U\|z_t-z^*\|^2+(V+1)(\tfrac{1}{2}\mu_x^{-1}L_{xx}^{\psi}+W)\|z_t-z^*\|^2+(V+1)W\|z_{t-1}-z^*\|^2\\ &\leq L(\|z_t-z^*\|^2+\|z_{t-1}-z^*\|^2),\end{aligned}\tag{4.112}$$

where in the third line we used $\|y_t - y^*\| \le \|z_t - z^*\|$, $\|x_t - x^*\| \le \|z_t - z^*\|$ and $\|y_{t-1} - y^*\| \le \|z_{t-1} - z^*\|$. Note also that we defined:

$$L = U + (V+1)(\tfrac{1}{2}\mu_x^{-1} L_{xx}^{\psi} + W). \tag{4.113}$$

Now let us prove that $z_{t+1} = (x_{t+1}, y_{t+1})$ is still in $\mathcal{N}_{\mathrm{CN}}$. This is because from (4.112),

$$\begin{aligned}
\|z_{t+1} - z^*\| &\le L\|z_t - z^*\| \cdot \|z_t - z^*\| + L\|z_{t-1} - z^*\| \cdot \|z_{t-1} - z^*\| \\
&\le L \cdot \tfrac{1}{3L} \cdot \|z_t - z^*\| + L \cdot \tfrac{1}{3L} \cdot \|z_{t-1} - z^*\| \\
&= \tfrac{1}{3}\|z_t - z^*\| + \tfrac{1}{3}\|z_{t-1} - z^*\| \\
&\le \tfrac{1}{3}\min\{\delta_x, \delta_y, \tfrac{1}{3L}\} + \tfrac{1}{3}\min\{\delta_x, \delta_y, \tfrac{1}{3L}\} \\
&\le \min\{\delta_x, \delta_y, \tfrac{1}{3L}\},
\end{aligned} \tag{4.114}$$

where in the second line we used that $\|z_t - z^*\| \le \frac{1}{3L}$ and $\|z_{t-1} - z^*\| \le \frac{1}{3L}$ and in the fourth line we used the assumption $\|z_t - z^*\| \le \min\{\delta_x, \delta_y, \frac{1}{3L}\}$ and $\|z_{t-1} - z^*\| \le \min\{\delta_x, \delta_y, \frac{1}{3L}\}$. These results follow from our assumption $z_t, z_{t-1} \in \mathcal{N}_{\mathrm{CN}}$ from induction. Therefore, we have proved that $\{z_t\}_{t=1}^{\infty} \subset \mathcal{N}_{\mathrm{CN}}$ given $z_1, z_2 \in \mathcal{N}_{\mathrm{CN}}$.

Denote $u_t = \|z_t - z^*\|$, we have:

$$u_{t+1} \le L(u_t^2 + u_{t-1}^2), \tag{4.115}$$

as in (4.112) for $t \ge 2$. Multiplying both sides by $2L$, we have:

$$2Lu_{t+1} \le \frac{(2Lu_t)^2 + (2Lu_{t-1})^2}{2}. \tag{4.116}$$

Define $v_t = 2Lu_t$ for $t \ge 1$ and let us prove by induction that for any $k \ge 1$, we have:

$$v_k \le q^{2^{\lfloor (k-1)/2 \rfloor}}, \; q = \max\{2Lu_1, 2Lu_2\}, \tag{4.117}$$

which is true for $k = 1, 2$. Since $z_1, z_2 \in \mathcal{N}_{\mathrm{CN}}$, we have $u_1 = \|z_1 - z^*\| \le \frac{1}{3L}$ and $u_2 = \|z_2 - z^*\| \le \frac{1}{3L}$, and thus $q < 1$. Suppose (4.117) is true for $k \le t$ and $t \ge 2$, then from (4.116) we can obtain:

$$\begin{aligned}
v_{t+1} &\le \frac{1}{2}\left((q^{2^{\lfloor (t-1)/2 \rfloor}})^2 + (q^{2^{\lfloor (t-2)/2 \rfloor}})^2\right) \\
&= \frac{1}{2}\left(q^{2^{\lfloor (t+1)/2 \rfloor}} + q^{2^{\lfloor t/2 \rfloor}}\right) \\
&\le \frac{1}{2}\left(q^{2^{\lfloor t/2 \rfloor}} + q^{2^{\lfloor t/2 \rfloor}}\right) \\
&= q^{2^{\lfloor (t+1-1)/2 \rfloor}},
\end{aligned} \tag{4.118}$$

where in the third line we used $q < 1$ and $\lfloor \frac{t+1}{2} \rfloor \geq \lfloor \frac{t}{2} \rfloor$. So, we have proved by induction that for any $t \geq 1$, the following holds:

$$2Lu_t = v_t \leq q^{2^{\lfloor (t-1)/2 \rfloor}}, \; q = \max\{2Lu_1, 2Lu_2\}, \tag{4.119}$$

namely, for any $t \geq 1$, we have

$$\|z_t - z^*\| \leq \frac{1}{2L} \max\{2L\|z_1 - z^*\|, 2L\|z_2 - z^*\|\}^{2^{\lfloor (t-1)/2 \rfloor}}. \tag{4.120}$$

□

The local super-linear convergence of CN means that this method is not heavily affected by the ill-conditioning of either the leader or the follower problem, when the initialization is close to the SLmM z^*. To obtain a good initialization, we consider the following method of *pre-training and fine-tuning*:

Pre-training and fine-tuning. We point out the sensitivity to initialization of our Newton-based algorithms: CN and GDN require the initialization to be close to the optimal solution, similar to the conventional Newton algorithm for minimization (Bertsekas, 1997). Fortunately, we can employ a "pre-training + fine-tuning" approach (Hinton and Salakhutdinov, 2006). For instance, we may run GDA for the initial phases, even though GDA is slowed down by the ill-conditioning, as soon as it goes in the neighborhood where Newton-type algorithms have convergence guarantees, we can switch to CN or GDN to converge quickly and to evade ill-conditioning, as we will show in Section 4.4.

4.3.3 Damping and Regularization

It is well-known that Newton-type methods only work in a neighborhood of the optimal solution. Therefore, for convergence to a SLmM, we can use gradient descent-ascent to converge to a neighborhood of a local minimax point, and then use Newton-type methods such as GDN or CN. Another modification might be to add damping and regularization. For example, for the Newton step in GDN, we can instead apply:

$$y' \leftarrow y - \gamma(\partial^2_{yy} - \lambda I)^{-1} \partial_y f(x, y), \tag{4.121}$$

where $\lambda > 0$ and $0 < \gamma \leq 1$. We call γ *the damping coefficient* and λ *the regularization coefficient*. If $\lambda = 0$ and $\gamma = 1$, then it is the pure Newton phase. If $\lambda \to \infty$ while γ/λ stay fixed then the algorithm is simply gradient ascent. We could modify GDN by taking

an adaptive scheme of γ and λ to stabilize this method. In a similar way, the Newton step of x in CN could be modified as:

$$x' \leftarrow x - \gamma(\mathtt{D}_{xx}^2 + \lambda I)^{-1}\partial_x f(x, y), \tag{4.122}$$

We could also choose an adaptive scheme of γ and λ, by choosing two sequences $\{\gamma_n\}$ and $\{\lambda_n\}$ such that $\gamma_n \to 1$ and $\lambda_n \to 0$ as the iteration step goes to infinity. Another way to choose γ is through line search (e.g. Boyd and Vandenberghe, 2004).

4.4 Experiments

We present experiments for Newton-type algorithms. Our numerical experiments confirm:

- The concept of strict local minimax is applicable in GAN training and ill-conditioned problems may arise even when learning simple distributions using GANs;
- Newton's algorithms can address the ill-conditioning problem and achieve much faster local convergence rate while keeping similar running time with existing algorithms such as GDA-k, TGDA and FR.

All our experiments in this section are run on an `Intel i9-7940X CPU` and a `NVIDIA TITAN V GPU`.

4.4.1 Learning a Gaussian Distribution

Consider learning a Gaussian distribution $x \sim \mathcal{N}(\mu, \Sigma)$ using a JS-GAN (Goodfellow et al., 2014), where the latent variable z follows a standard Gaussian.

First, we estimate the mean μ with two different covariance matrices: a well-conditioned covariance $\Sigma = I$ and an ill-conditioned covariance $\Sigma = \mathrm{diag}(1, 0.05)$. We use a discriminator $D(x)$ and a generator $G(z)$, such that

$$D(x) = \sigma\left(\omega^\top x\right), \; G(z) = z + \eta \tag{4.123}$$

The corresponding GAN training problems are not convex-concave, yet the optimal solutions are SLmMs. The minimax problem of GAN training can be written as:

$$\min_{\eta\in\mathbb{R}^2}\max_{\omega\in\mathbb{R}^2} \; \ell(\eta, \omega) := \mathbb{E}_{x\sim\mathcal{N}(\mathbf{0},\Sigma)} \log\sigma(\omega^\top x) + \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\Sigma)} \log\left(1 - \sigma(\omega^\top(z+\eta))\right), \tag{4.124}$$

The problem is concave-concave. It is easy to check that $(\eta^*, \omega^*) = (\mathbf{0}, \mathbf{0})$ is a global minimax point. We have the gradients:

$$\partial_\eta \ell(\eta, \omega) = -\mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\Sigma)}\sigma(\omega^\top(z+\eta))\omega \tag{4.125}$$

$$\partial_\omega \ell(\eta, \omega) = \mathbb{E}_{x\sim\mathcal{N}(\mathbf{0},\Sigma)}(1-\sigma(\omega^\top x))x - \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\Sigma)}\sigma(\omega^\top(z+\eta))(z+\eta) \tag{4.126}$$

and the partial Hessians:

$$\partial^2_{\eta\eta} \ell(\eta, \omega) = -\mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\Sigma)}\sigma(\omega^\top(z+\eta))(1-\sigma(\omega^\top(z+\eta)))\omega\omega^\top \tag{4.127}$$

$$\partial^2_{\omega\omega} \ell(\eta, \omega) = -\mathbb{E}_{x\sim\mathcal{N}(\mathbf{0},\Sigma)}\sigma(\omega^\top x)(1-\sigma(\omega^\top x))xx^\top, \tag{4.128}$$

$$- \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\Sigma)}\sigma(\omega^\top(z+\eta))(1-\sigma(\omega^\top(z+\eta)))(z+\eta)(z+\eta)^\top, \tag{4.129}$$

$$\partial^2_{\eta\omega} \ell(\eta, \omega) = -\mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},\Sigma)}(\sigma(\omega^\top(z+\eta))I + \sigma'(\omega^\top(z+\eta))(z+\eta)\omega^\top). \tag{4.130}$$

At (η^*, ω^*), we have

$$\partial^2_{\eta\eta} \ell(\eta^*, \omega^*) = \mathbf{0},\ \partial^2_{\omega\omega} \ell(\eta^*, \omega^*) = -\frac{1}{2}\Sigma,\ \partial^2_{\eta\omega} \ell(\eta^*, \omega^*) = -\frac{1}{2}I, \tag{4.131}$$

and thus this point is a SLmM.

Now let us consider learning the covariance of a Gaussian:

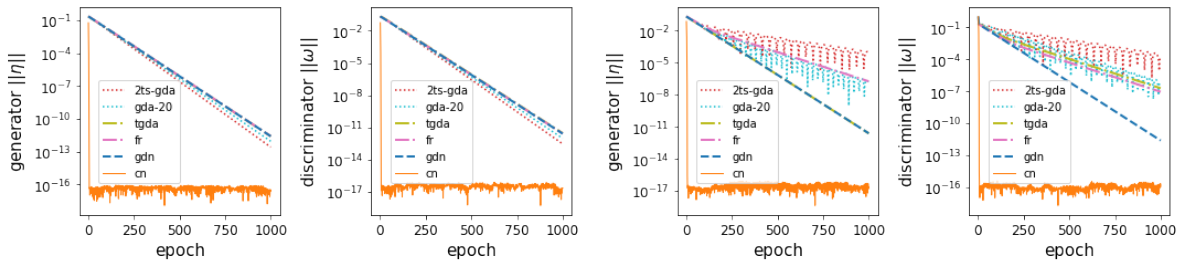
$$\min_{V\in\mathbb{R}^{2\times 2}} \max_{W\in\mathbb{R}^{2\times 2}} \ell(V, W) := \mathbb{E}_{x\sim\mathcal{N}(\mathbf{0},\Sigma)} \log\sigma\left(x^\top W x\right) + \mathbb{E}_{z\sim\mathcal{N}(\mathbf{0},I)} \log\left(1-\sigma\left(z^\top V^\top W V z\right)\right), \tag{4.132}$$

with $x \in \mathbb{R}^2$ and $z \in \mathbb{R}^2$. The generator is $G(z) = Vz$ and the discriminator is $D(x) = \sigma(x^\top W x)$. The optimal solution satisfies $VV^\top = \Sigma = \mathrm{diag}(1, 0.04)$ and $W + W^\top = \mathbf{0}$.

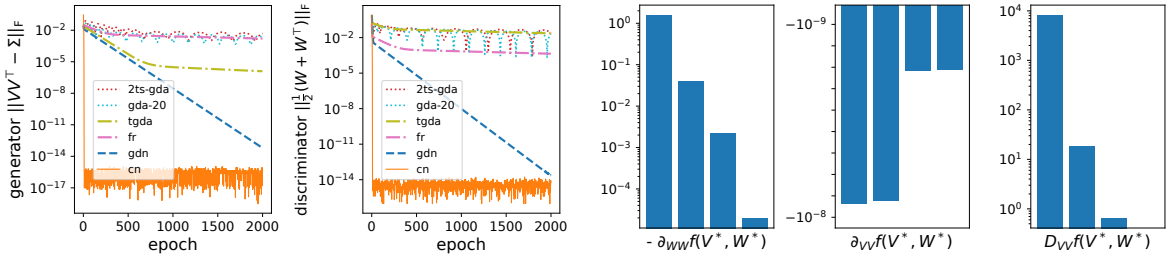
Experiments First, let us estimate the mean of a Gaussian distribution. Comparison among algorithms are presented in Figures 4.1a and 4.1b. While the convergence rates for most algorithms on the well-conditioned Gaussian are similar, all existing methods severely slow down on the ill-conditioned Gaussian. Only Newton-type methods retain their fast convergence, confirming our theory that they can cope with ill-conditioned problems. In particular, in both cases CN converges to a high precision solution only in a few iterations, verifying its superlinear convergence rate.

The covariance of the data distribution determines the condition of ω. We compare convergence speed in two cases: a well-conditioned covariance

$$\Sigma = I \tag{4.133}$$



(a) well-conditioned Gaussian mean estimation (b) ill-conditioned Gaussian mean estimation



(c) ill-conditioned Gaussian covariance estimation (d) eigenvalues of $-\partial_{WW}\ell$, $\partial_{VV}\ell$ and $\mathtt{D}_{VV}\ell$ at the SLmM.

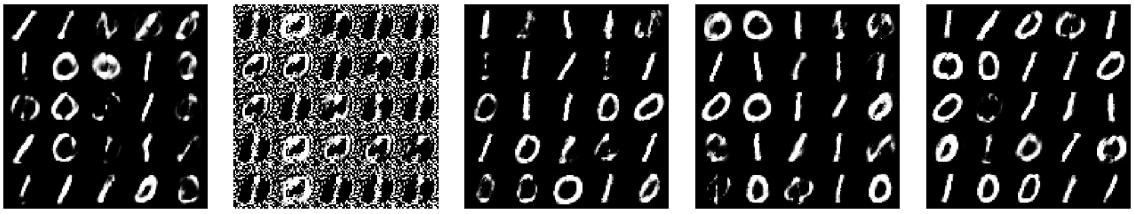
Figure 4.1: Convergence on learning Gaussian distributions using JS-GAN. **Top:** Estimating the mean of a Gaussian. We compare the convergence rate in a well-conditioned and an ill-conditioned setting, and plot the norm of the generator and the discriminator respectively. **Bottom:** Estimating the covariance of a Gaussian. We plot the convergence behavour of different algorithms and the eigenvalues at the SLmM. In both cases, CN quickly reaches the *precision limit* of double precision floating point numbers.

and an ill-conditioned covariance

$$\Sigma = \begin{bmatrix} 1 & 0 \\ 0 & 0.05 \end{bmatrix}. \tag{4.134}$$

We set $\alpha_{\mathtt{L}} = 0.05$, $\alpha_{\mathtt{F}} = 0.5$ for all algorithms. For GDA-k we set $\alpha = 0.05$. We run conjugate gradient for up to 8 iterations and terminate it whenever the norm of residual is smaller than 10^{-40}. The size of training data is 10000. We randomly initialize the parameters in running all algorithms using a zero-mean Gaussian with standard deviation 0.1.

Second, we estimate an ill-conditioned covariance $\Sigma = \mathrm{diag}(1, 0.04)$ with a fixed mean $\mu = \mathbf{0}$. We set $\alpha_{\mathtt{L}} = 0.02$, $\alpha_{\mathtt{F}} = 0.2$ for all algorithms. For GDA-k we set $\alpha = 0.02$.



(a) GDA-20 (b) TGDA (c) FR (d) **GDN** (e) **CN**

Figure 4.2: Digits generated by different algorithms on MNIST 0/1 subset. We draw samples from the latent distribution and pass them to the generator learned with different algorithms.

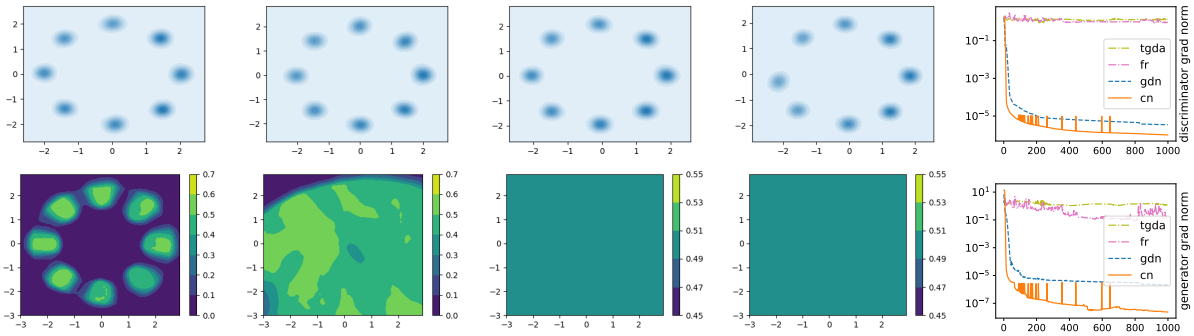
We run conjugate gradient for up to 16 iterations and terminate it whenever the norm of residual is smaller than 10^{-30}. The size of the training data is 10000. A ℓ_2 norm regularization is added on the discriminator and the regularization coefficient is 10^{-5}. We randomly initialize the parameters in running all algorithms using a zero-mean Gaussian with standard deviation 0.01.

We plot the eigenvalues at the optimal solution in Figure 4.1d:

- the solution here is almost an SLmM, as the total derivative $\mathtt{D}_{VV}\ell$ is approximately positive definite (the only negative eigenvalue is on the order of 10^{-9}), and $\partial_{WW}\ell$ is negative definite;
- the problem is ill-conditioned, as the condition number of $\partial_{WW}\ell$ is greater than 10^4.

Because of the poor conditioning, we observe again that GDA and TGDA/FR severely slow down, while only GDN and CN can retain their fast convergence rate (Figure 4.1c). In particular, CN converges superlinearly and reaches the precision limit of floating numbers in only a few iterations. Note that the solution is not a saddle point, as ∂_{VV} in Figure 4.1d is negative definite. Thus algorithms for strongly-convex-strongly-concave functions may not work.

From Thm. 4.2.3, TGDA and FR have the same convergence rate since their preconditioners on GDA are transpose of each other (Section 4.2.2). The convergence behaviour of the leader and the follower slightly differ: TGDA converges faster on the generator while FR converges faster on the discriminator.



(a) TGDA (b) FR (c) GD-Newton (d) CN (e) Grad Norm

Figure 4.3: Convergence on a mixture of 8 Gaussians. **Top:** samples from generator. **Bottom:** discriminator prediction. **Last column:** gradient norms during training. The x-axis is epoch.

4.4.2 Learning Mixture of Gaussians

We learn a mixture of Gaussians using JS-GAN in Figure 4.3. Both the discriminator and the generator are 3-hidden-layer ReLU networks with 256 neurons in each hidden layer. The latent variable z is sampled from a 100 dimensional standard Gaussian distribution. The size of training data is 10000. We first use GDA ($\alpha_{\mathtt{L}} = \alpha_{\mathtt{F}} = 0.01$) with batch size 256 to find the initialization for other methods. TGDA, FR and GD-Newton use $\alpha_{\mathtt{L}} = 0.01$ and $\alpha_{\mathtt{F}} = 0.02$. We run conjugate gradient for 20 iterations to solve linear systems and terminate it whenever the norm of residual is smaller than 10^{-40}. For CN, we choose the damping coefficient $\gamma = 0.1$ (see (4.122)) with 20 CG iterations for the discriminator, and 32 CG iterations for the generator. We also add a regularization factor $\lambda = 0.1$ for the generator as in (4.122).

We plot the distribution learned by the generator, the discriminator prediction, and gradient norms during training. The discriminator trained by GDN/CN is totally fooled by the generator, predicting constant $\frac{1}{2}$ almost everywhere, and the gradient norms shrink quickly after a few epochs. In contrast, the gradient norms of TGDA and FR decrease, if at all, very slowly.

Although this is a two dimensional example, the minimax optimization problem has several hundred thousand variables since the generator and the discriminator are deep networks, demonstrating the moderate scalability of Newton-type algorithms to high dimensional problems.

Table 4.2: Running times per epoch on MNIST.

method	GDA-20	TGDA	FR	GDN	CN
time (in sec)	2.78	6.08	6.22	4.46	7.04

4.4.3 MNIST

We compare different algorithms for generating digits on the 0/1 MNIST subset. We use Wasserstein GAN (Arjovsky et al., 2017) to learn the distribution, with 2-hidden-layer MLPs (512 neurons for each hidden layer) for both the generator and the discriminator, and we impose spectral normalization (Miyato et al., 2018) on the discriminator. We first run GDA, which is oscillating in a neighborhood, and use its output as initialization. We compare the per epoch running time of different algorithms in Table 4.2. TGDA, FR, GDN and CN have similar running times since they solve linear systems of similar sizes in their updates. Since we choose a small number of CG iterations (`max_iteration = 16` for the discriminator and `max_iteration = 8` for the generator), they have similar running times to GDA-20, as predicted by our Table 4.1.

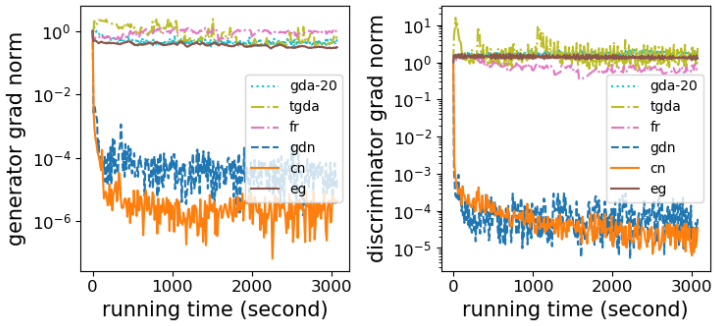


Figure 4.4: Gradient norms on MNIST 0/1 subset.

Even though all of our algorithms have similar running times, we find the convergence speeds are quite different. We plot the change of their gradient norms with respect to the running time in Figure 4.4, where we also compared with the method of extra-gradient (EG, Korpelevich (1976)). TGDA/GDA-20/FR do not converge or converge quite slowly. In contrast, GDN converges much faster than all these algorithms above with the same step sizes, as predicted by Theorems 4.3.2, 4.2.3 and 4.2.2. The convergence speed can be further improved by CN, where the gradient norms diminish faster even if we take a small number of CG iterations. We plot the digits learned by these algorithms in Figure 4.2. It can be seen that GDN/CN generate high-quality digits that are as good as, if not better

than, other optimizers.

Chapter 5

Conclusions

In this chapter I conclude the thesis and discuss possible directions for future work.

The aim of Chapter 2 is to provide a comprehensive study of the recently proposed local minimax points (Jin et al., 2020). I discussed the relations between local saddle and local minimax points, between local and global minimax points, and interpreted local minimax points based on infinitesimal robustness. I presented the first- and second-order optimality conditions of these local optimal solutions, which extend Jin et al. (2020) to the constrained and degenerate cases. Specifically, in (potentially non-convex) quadratic games, local minimax points are (in some sense) equivalent to global minimax points. I also studied the stability of popular gradient algorithms near local optimal solutions, which provides insights for the design of algorithms to find minimax points.

The implication of this work is two-fold: **(a)** we may need new algorithms for smooth games, since I have shown in Proposition 3.4.5 that our common intuition might fail w.r.t. the convergence to a local and global minimax point; **(b)** we need to think about new solution concepts other than global/local minimax points. As many theoretical works aim to go beyond the definition of Nash equilibria (a.k.a. saddle points) such as Jin et al. (2020); Farnia and Ozdaglar (2020); Berard et al. (2020), to name a few, we may need to take one step further, beyond the definition of Stackelberg equilibria (a.k.a minimax points), as also pointed out in Schaefer et al. (2020).

In Chapter 3 I focus on the local stability of gradient-based algorithms. By drawing a connection to discrete linear dynamical systems and using Schur's theorem, I provide necessary and sufficient conditions for a variety of gradient algorithms, for both simultaneous (Jacobi) and alternating (Gauss–Seidel) updates. My results show that Gauss–Seidel updates converge more easily than Jacobi updates in bilinear games, by proving that the

feasible hyperparameters of GS updates strictly include the feasible hyperparameters of the corresponding Jacobi updates. I performed a number of experiments to validate my theoretical findings and suggest further analysis.

In Chapter 4, I developed two Newton-type algorithms for local convergence of unconstrained *nonconvex-nonconcave* minimax optimization which have wide applications in, e.g., GAN training and adversarial robustness. My algorithms

- share the same computational complexity as existing alternatives that explore second-order information;
- have much faster local convergence, especially for ill-conditioned problems.

Experiments show that my algorithms cope with the ill-conditioning that arises from practical GAN training problems. Since I only study the local convergence of Newton-type methods, I consider them as a strategy to "fine-tune" the solution and accelerate the local convergence, after finding a good initialization or pre-training with other methods, such as GDA or damped Newton. How to use second-order information to obtain fast global convergence to local optimal solutions in nonconvex minimax optimization with theoretical guarantees remains an important problem.

Minimax optimization has many applications in modern machine learning as I have discussed in Chapter 1. Despite recent theoretical works including my thesis, there is still a big gap between theory and applications. On the one side, many concurrent works focus on general minimax optimization. On the other side, current applications of minimax optimization usually have specific problem structures, which are largely unexplored. In the future I plan to study more applications of minimax optimization, including domain adversarial training (Acuna et al., 2021) and domain generalization. Understanding the solution concepts and stability in such problems would be important to improve the optimization and thus the training process.

References

Acuna, D., Zhang, G., Law, M. T., and Fidler, S. (2021). f-domain-adversarial learning: Theory and algorithms. In *International Conference on Machine Learning*.

Anandalingam, G. and Friesz, T. L. (1992). Hierarchical optimization: An introduction. *Annals of Operations Research*, 34(1):1–11.

Arjovsky, M., Chintala, S., and Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning*.

Arrow, K., Hurwicz, L., and Uzawa, H. (1958). *Studies in linear and non-linear programming*. Stanford University Press.

Azizian, W., Mitliagkas, I., Lacoste-Julien, S., and Gidel, G. (2020a). A tight and unified analysis of extragradient for a whole spectrum of differentiable games. In *the 23rd International Conference on Artificial Intelligence and Statistics*.

Azizian, W., Scieur, D., Mitliagkas, I., Lacoste-Julien, S., and Gidel, G. (2020b). Accelerating smooth games by manipulating spectral shapes. In *the 23rd International Conference on Artificial Intelligence and Statistics*.

Bailey, J. P., Gidel, G., and Piliouras, G. (2019). Finite regret and cycles with fixed step-size via alternating gradient descent-ascent. *arXiv preprint arXiv:1907.04392*.

Bailey, J. P. and Piliouras, G. (2018). Multiplicative weights update in zero-sum games. In *Proceedings of the 2018 ACM Conference on Economics and Computation*, pages 321–338. ACM.

Barazandeh, B. and Razaviyayn, M. (2020). Solving non-convex non-differentiable min-max games using proximal gradient method. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 3162–3166. IEEE.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. (2010). A theory of learning from different domains. *Machine learning*, 79(1):151–175.

Ben-Tal, A. and Zowe, J. (1982). Necessary and sufficient optimality conditions for a class of nonsmooth minimization problems. *Mathematical Programming*, 24(1):70–91.

Ben-Tal, A. and Zowe, J. (1985). Directional derivatives in nonsmooth optimization. *Journal of Optimization Theory and Applications*, 47(4):483–490.

Berard, H., Gidel, G., Almahairi, A., Vincent, P., and Lacoste-Julien, S. (2020). A closer look at the optimization landscapes of generative adversarial networks. In *International Conference on Learning Representations*.

Bermúdez-Chacón, R., Salzmann, M., and Fua, P. (2019). Domain adaptive multibranch networks. In *International Conference on Learning Representations*.

Bertsekas, D. P. (1997). Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334.

Bollapragada, R., Scieur, D., and d'Aspremont, A. (2019). Nonlinear acceleration of primal-dual algorithms. In *the 22nd International Conference on Artificial Intelligence and Statistics*, pages 739–747.

Borkar, V. S. (2008). *Stochastic Approximation: A Dynamical Systems Viewpoint*. Springer.

Boyd, S. and Vandenberghe, L. (2004). *Convex optimization*. Cambridge university press.

Bruck, R. E. (1977). On the weak convergence of an ergodic iteration for the solution of variational inequalities for monotone operators in Hilbert space. *Journal of Mathematical Analysis and Applications*, 61(1):159–164.

Cheng, S. S. and Chiou, S. S. (2007). Exact stability regions for quartic polynomials. *Bulletin of the Brazilian Mathematical Society*, 38(1):21–38.

Clarke, F. H. (1990). *Optimization and Nonsmooth Analysis*. SIAM.

Cohen, J., Rosenfeld, E., and Kolter, Z. (2019). Certified adversarial robustness via randomized smoothing. In *International Conference on Machine Learning*, pages 1310–1320. PMLR.

Collins, G. E. (1975). Quantifier elimination for real closed fields by cylindrical algebraic decompostion. In *Automata theory and formal languages*, pages 134–183. Springer.

Cominetti, R. and Correa, R. (1990). A Generalized Second-Order Derivative in Nonsmooth Optimization. *SIAM Journal on Control and Optimization*, 28(4):789–809.

Danskin, J. M. (1966). The Theory of Max-Min, with Applications. *SIAM Journal on Applied Mathematics*, 14(4):641–664.

Daskalakis, C., Ilyas, A., Syrgkanis, V., and Zeng, H. (2018). Training GANs with optimism. In *the 6th International Conference on Learning Representations*.

Daskalakis, C. and Panageas, I. (2018). The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pages 9236–9246.

Dem'yanov, V. F. (1966). On the solution of several minimax problems. I. *Cybernetics*, 2:47–53.

Dem'yanov, V. F. (1970). Sufficient conditions for a local minimax. *USSR Computational Mathematics and Mathematical Physics*, 10(5):53–63.

Dem'yanov, V. F. (1973). Second-order directional derivatives of a function of the maximum. *Cybernetics*, 9:797—800.

Dem'yanov, V. F. and Malozemov, V. N. (1974). *Introduction to Minimax*. Wiley.

Evtushenko, Y. (1974a). Some local properties of minimax problems. *USSR Computational Mathematics and Mathematical Physics*, 14(3):129 – 138.

Evtushenko, Y. G. (1974b). Iterative methods for solving minimax problems. *USSR Computational Mathematics and Mathematical Physics*, 14(5):52–63.

Facchinei, F. and Pang, J.-S. (2007). *Finite-dimensional variational inequalities and complementarity problems*. Springer Science & Business Media.

Fan, K. (1950). On a theorem of weyl concerning eigenvalues of linear transformations: II. *Proceedings of the National Academy of Sciences of the United States of America*, 36(1):31.

Farnia, F. and Ozdaglar, A. (2020). Do GANs always have Nash equilibria? In *International Conference on Machine Learning*, pages 3029–3039. PMLR.

Fiez, T., Chasnov, B., and Ratliff, L. J. (2019). Convergence of learning dynamics in Stackelberg games. *arXiv*. arXiv:1906.01217.

Fliege, J., Tin, A., and Zemkoho, A. (2021). Gauss–newton-type methods for bilevel optimization. *Computational Optimization and Applications*, 78(3):793–824.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030.

Gidel, G., Berard, H., Vignoud, G., Vincent, P., and Lacoste-Julien, S. (2019a). A variational inequality perspective on generative adversarial networks. In *International Conference on Learning Representations*.

Gidel, G., Hemmat, R. A., Pezeshki, M., Huang, G., Lepriol, R., Lacoste-Julien, S., and Mitliagkas, I. (2019b). Negative momentum for improved game dynamics. In *the 22nd International Conference on Artificial Intelligence and Statistics*.

Gohberg, I., Lancaster, P., and Rodman, L. (1982). *Matrix Polynomials*. Academic Press.

Golshtein, E. G. (1972). A generalized gradient method for finding saddlepoints. *Ekonomika i matematicheskie*, 8(4):36–52.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Goodfellow, I. J., Shlens, J., and Szegedy, C. (2015). Explaining and harnessing adversarial examples. In *ICLR*.

Hampel, F. R. (1974). The influence curve and its role in robust estimation. *Journal of the american statistical association*, 69(346):383–393.

Han, J. and Sun, D. (1998). Newton-type methods for variational inequalities. In *Advances in Nonlinear Programming*, pages 105–118. Springer.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. (2017). GANs trained by a two time-scale update rule converge to a local Nash equilibrium. In *Advances in neural information processing systems*, pages 6626–6637.

Hinton, G. E. and Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507.

Hiriart-Urruty, J.-B. and Lemaréchal, C. (2004). *Fundamentals of convex analysis*. Springer Science & Business Media.

Hiriart-Urruty, J.-B. and Lemaréchal, C. (2013). *Convex analysis and minimization algorithms I: Fundamentals*, volume 305. Springer.

Hoffman, J., Tzeng, E., Park, T., Zhu, J.-Y., Isola, P., Saenko, K., Efros, A., and Darrell, T. (2018). Cycada: Cycle-consistent adversarial domain adaptation. In *International conference on machine learning*, pages 1989–1998. PMLR.

Hsieh, Y.-G., Iutzeler, F., Malick, J., and Mertikopoulos, P. (2019). On the convergence of single-call stochastic extra-gradient methods. In *NeurIPS*, pages 6936–6946.

Hsieh, Y.-G., Iutzeler, F., Malick, J., and Mertikopoulos, P. (2020). Explore aggressively, update conservatively: Stochastic extragradient methods with variable stepsize scaling. *arXiv preprint arXiv:2003.10162*.

Ibrahim, A., Azizian, W., Gidel, G., and Mitliagkas, I. (2020). Linear lower bounds and conditioning of differentiable games. In *International conference on machine learning*, pages 6356–6366.

Izmailov, A. F. and Solodov, M. V. (2014). *Newton-type methods for optimization and variational problems*. Springer.

Jin, C., Netrapalli, P., and Jordan, M. (2020). What is local optimality in nonconvex-nonconcave minimax optimization? In *International conference on machine learning*, pages 5735–5744.

Karras, T., Laine, S., and Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4401–4410.

Katok, A. and Hasselblatt, B. (1995). *Introduction to the modern theory of dynamical systems*, volume 54. Cambridge university press.

Kawasaki, H. (1988). The upper and lower second order directional derivatives of a sup-type function. *Mathematical Programming*, 41(1-3):327–339.

Kawasaki, H. (1991). Second order necessary optimality conditions for minimizing a sup-type function. *Mathematical programming*, 49(1-3):213–229.

Kawasaki, H. (1992). Second-order necessary and sufficient optimality conditions for minimizing a sup-type function. *Applied Mathematics and Optimization*, 26(2):195–220.

Korpelevich, G. (1976). The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756.

Li, C.-L., Chang, W.-C., Cheng, Y., Yang, Y., and Póczos, B. (2017). MMD GAN: Towards deeper understanding of moment matching network. In *NIPS*.

Liang, T. and Stokes, J. (2019). Interaction matters: A note on non-asymptotic local convergence of generative adversarial networks. In *the 22nd International Conference on Artificial Intelligence and Statistics*.

Liu, H., Simonyan, K., and Yang, Y. (2018). Darts: Differentiable architecture search. In *International Conference on Learning Representations*.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. (2018). Conditional adversarial domain adaptation. In *NeurIPS*.

Maclaurin, D., Duvenaud, D., and Adams, R. (2015). Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning*, pages 2113–2122. PMLR.

Madry, A. (2019). Adversarial Robustness: Theory, Practice and Beyond. *Waterloo ML + Security + Verification Workshop*.

Madry, A., Makelov, A., Schmidt, L., Tsipras, D., and Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. In *the 6th International Conference on Learning Representations*.

Mansour, M. (2011). Discrete-time and sampled-data stability tests.

Marshall, A. W., Olkin, I., and Arnold, B. C. (1979). *Inequalities: theory of majorization and its applications*, volume 143. Springer.

Martens, J. (2010). Deep learning via Hessian-free optimization. In *Proceedings of the 27th International Conference on Machine Learning*, pages 735—742.

Martinet, B. (1970). Régularisation d'inéquations variationnelles par approximations successives. *ESAIM: Mathematical Modelling and Numerical Analysis: Modélisation Mathématique et Analyse Numérique*, 4(R3):154–158.

Mazumdar, E., Ratliff, L. J., and Sastry, S. (2018). On the convergence of gradient-based learning in continuous games. *arXiv preprint arXiv:1804.05464*.

Mertikopoulos, P., Lecouat, B., Zenati, H., Foo, C.-S., Chandrasekhar, V., and Piliouras, G. (2019). Optimistic mirror descent in saddle-point problems: Going the extra (gradient) mile. In *the 7th International Conference on Learning Representations*.

Mertikopoulos, P., Papadimitriou, C., and Piliouras, G. (2018a). Cycles in adversarial regularized learning. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2703–2717. SIAM.

Mertikopoulos, P., Papadimitriou, C., and Piliouras, G. (2018b). Cycles in adversarial regularized learning. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2703–2717.

Mescheder, L., Geiger, A., and Nowozin, S. (2018). Which training methods for GANs do actually converge? In *International Conference on Machine Learning*.

Mescheder, L., Nowozin, S., and Geiger, A. (2017). The numerics of GANs. In *Advances in Neural Information Processing Systems*, pages 1825–1835.

Meyer, C. D. (2000). *Matrix analysis and applied linear algebra*, volume 71. Siam.

Miyato, T., Kataoka, T., Koyama, M., and Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. In *International Conference on Learning Representations*.

Mokhtari, A., Ozdaglar, A., and Pattathil, S. (2019). Proximal point approximations achieving a convergence rate of $o(1/k)$ for smooth convex-concave saddle point problems: Optimistic gradient and extra-gradient methods. *arXiv:1906.01115*.

Mokhtari, A., Ozdaglar, A., and Pattathil, S. (2020). A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach.

Monteiro, R. D. C. and Svaiter, B. F. (2010). On the complexity of the hybrid proximal extragradient method for the iterates and the ergodic mean. *SIAM Journal on Optimization*, 20(6):2755–2787.

Mroueh, Y., Li, C.-L., Sercu, T., Raj, A., and Cheng, Y. (2018). Sobolev GAN. In *International Conference on Learning Representations*.

Murty, K. G. and Kabadi, S. N. (1987). Some np-complete problems in quadratic and nonlinear programming. *Mathematical programming*, 39(2):117–129.

Nagarajan, V. and Kolter, J. Z. (2017). Gradient descent GAN optimization is locally stable. In *Advances in Neural Information Processing Systems*, pages 5585–5595.

Nash, J. F. (1950). Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36(1):48–49.

Nedić, A. and Ozdaglar, A. (2009). Subgradient methods for saddle-point problems. *Journal of optimization theory and applications*, 142(1):205–228.

Nemirovski, A. (2004). Prox-method with rate of convergence $O(1/t)$ for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251.

Nemirovski, A. S. and Yudin, D. B. (1978). Cesari convergence of the gradient method of approximating saddle points of convex-concave functions. In *Doklady Akademii Nauk*, volume 239, pages 1056–1059. Russian Academy of Sciences.

Nemirovsky, A. S. and Yudin, D. B. (1983). *Problem complexity and method efficiency in optimization*. Wiley.

Nesterov, Y. (1983). A method for unconstrained convex minimization problem with the rate of convergence $o(1/k^2)$. *Doklady AN USSR*, 269:543–547.

Nesterov, Y. (2003). *Introductory lectures on convex optimization: A basic course*, volume 87. Springer Science & Business Media.

Niethammer, W. and Varga, R. S. (1983). The analysis of k-step iterative methods for linear systems from summability theory. *Numerische Mathematik*, 41(2):177–206.

Nowozin, S., Cseke, B., and Tomioka, R. (2016). f-gan: Training generative neural samplers using variational divergence minimization. In *NIPS*.

Peng, W., Dai, Y.-H., Zhang, H., and Cheng, L. (2020). Training GANs with centripetal acceleration. *Optimization Methods and Software*, 35(5):955–973.

Polyak, B. (1987). *Introduction to Optimization*. Optimization Software Inc.

Polyak, B. T. (1964). Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17.

Popov, L. D. (1980). A modification of the Arrow–Hurwicz method for search of saddle points. *Mathematical Notes*, 28(5):845–848.

Rockafellar, R. T. (1976). Monotone operators and the proximal point algorithm. *SIAM journal on control and optimization*, 14(5):877–898.

Rockafellar, R. T. and Wets, R. J.-B. (2009). *Variational analysis*, volume 317. Springer Science & Business Media.

Royer, C. W., O'Neill, M., and Wright, S. J. (2020). A Newton-CG algorithm with complexity guarantees for smooth unconstrained optimization. *Mathematical Programming*, 180(1):451–488.

Saad, Y. (2003). *Iterative Methods for Sparse Linear Systems*. SIAM, 2nd edition.

Sagun, L., Bottou, L., and LeCun, Y. (2016). Singularity of the hessian in deep learning. *arXiv preprint arXiv:1611.07476*.

Schaefer, F., Zheng, H., and Anandkumar, A. (2020). Implicit competitive regularization in GANs. In *International Conference on Machine Learning*, pages 8533–8544. PMLR.

Schur, I. (1917). Über potenzreihen, die im innern des einheitskreises beschränkt sind. *Journal für die reine und angewandte Mathematik*, 147:205–232.

Seeger, A. (1988). Second order directional derivatives in parametric optimization problems. *Mathematics of Operations Research*, 13(1):124–139.

Shen, J., Qu, Y., Zhang, W., and Yu, Y. (2018). Wasserstein distance guided representation learning for domain adaptation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Shu, R., Bui, H. H., Narui, H., and Ermon, S. (2018). A DIRT-T approach to unsupervised domain adaptation. In *Proc. 6th International Conference on Learning Representations*.

Sinha, A., Namkoong, H., and Duchi, J. (2018). Certifying some distributional robustness with principled adversarial training. In *International Conference on Learning Representations*.

Stein, P. and Rosenberg, R. (1948). On the solution of linear simultaneous equations by iteration. *Journal of the London Mathematical Society*, 1(2):111–118.

Sutskever, I., Martens, J., Dahl, G., and Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In *International conference on machine learning*, pages 1139–1147.

Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2014). Intriguing properties of neural networks. iclr. 2014. *arXiv preprint arXiv:1312.6199*.

Varga, R. S. (1962). *Iterative analysis*. Springer.

von Stackelberg, H. (1934). *Market structure and equilibrium*. Springer.

Wang, Y., Xiu, N., and Han, J. (2010). On cone of nonsymmetric positive semidefinite matrices. *Linear algebra and its applications*, 433(4):718–736.

Wang, Y., Zhang, G., and Ba, J. (2020). On solving minimax optimization locally: A follow-the-ridge approach. In *the 8th International Conference on Learning Representations*.

Zhang, G., Poupart, P., and Yu, Y. (2020). Optimality and stability in non-convex smooth games. arXiv:2002.11875.

Zhang, G., Wu, K., Poupart, P., and Yu, Y. (2021). Newton-type methods for minimax optimization. ICML workshop on Beyond First Order Methods in Machine Learning.

Zhang, G. and Yu, Y. (2020). Convergence of gradient methods on bilinear zero-sum games. In *the 8th International Conference on Learning Representations*.

Zhang, Y., Liu, T., Long, M., and Jordan, M. (2019). Bridging theory and algorithm for domain adaptation. In *International Conference on Machine Learning*, pages 7404–7413. PMLR.

APPENDICES

Appendix A

Supplementary Material for Chapter 2

A.1 Nonsmooth Analysis: A Short Detour

We give a short detour on some classical optimality conditions in nonsmooth optimization. These results will be used in Section 2.2 to yield necessary and sufficient conditions for local optimality in zero-sum two-player games, since the optimality conditions for local optimal points can be reduced to those for the envelope functions, which are in general non-smooth.

Let h be a function defined on some set $\mathcal{X} \subseteq \mathbb{R}^m$. Its upper and lower (Dini) directional derivatives are defined as:

$$\mathsf{D}h(x;d) := \limsup_{t\to 0^+} \frac{h(x+td)-h(x)}{t},\ \mathsf{D}_+h(x;d) := \liminf_{t\to 0^+} \frac{h(x+td)-h(x)}{t}. \tag{A.1}$$

When the two limits coincide, we use the notation $\mathsf{D}h(x;d)$ and call the function h directionally differentiable (at x along direction d). We can similarly define the upper and lower

second-order directional derivatives[1] according to Ben-Tal and Zowe (1982):

$$\mathsf{H}h(x;d,g) = \limsup_{t\to 0^+} \frac{h(x+td+t^2g/2) - h(x) - t\cdot \mathsf{D}h(x;d)}{t^2/2}, \tag{A.2}$$

$$\mathsf{H}_+h(x;d,g) = \liminf_{t\to 0^+} \frac{h(x+td+t^2g/2) - h(x) - t\cdot \mathsf{D}h(x;d)}{t^2/2}. \tag{A.3}$$

Similarly, when the two limits coincide we use the simplified notation $\mathsf{H}h(x;d,g)$ and call h twice directionally differentiable (at x along parabolic (d,g)). Note that, when $d=\mathbf{0}$, we recover the directional derivative:

$$\mathsf{H}h(x;\mathbf{0},g) = \mathsf{H}_+h(x;\mathbf{0},g) = \mathsf{D}h(x;g), \tag{A.4}$$

while if $g=\mathbf{0}$,

$$\mathsf{H}h(x;d) := \mathsf{H}h(x;d,\mathbf{0}),\ \mathsf{H}_+h(x;d) := \mathsf{H}_+h(x;d,\mathbf{0}),\ \mathsf{H}h(x;d) := \mathsf{H}h(x;d,\mathbf{0}) \tag{A.5}$$

reduces to the second-order directional derivatives of Dem'yanov (1973). The advantage of the definition of Ben-Tal and Zowe (1982) is evidenced in the following chain rule:

Theorem A.1.1 (Ben-Tal and Zowe 1982)**.** *Let $h:\mathbb{R}^m\to\mathbb{R}$ be locally Lipschitz and $k:\mathbb{R}^n\to\mathbb{R}^m$ be (twice) directionally differentiable. Then,*

$$\mathsf{D}(h\circ k)(x;d) = \mathsf{D}h\big(k(x);\mathsf{D}k(x;d)\big), \tag{A.6}$$

$$\mathsf{H}(h\circ k)(x;d,g) = \mathsf{H}h\big(k(x);\mathsf{D}k(x;d),\mathsf{H}k(x;d,g)\big). \tag{A.7}$$

(The same result holds for the lower derivatives, and hence the derivatives when they exist.)

In contrast, the definition of Dem'yanov (1973) fails to satisfy the chain rule above. Indeed, if h is differentiable, then

$$\mathsf{D}h(x;d) = \langle \nabla h(x), d\rangle \tag{A.8}$$

while if h is twice differentiable, then

$$\mathsf{H}h(x;d,g) = \mathsf{D}h(x;g) + \mathsf{H}h(x;d) = \langle \nabla h(x), g\rangle + \left\langle d, \nabla^2 h(x) d\right\rangle, \tag{A.9}$$

where ∇h and $\nabla^2 h$ are the gradient and Hessian of h, respectively. (A slightly more general setting is discussed in Seeger 1988, Proposition 1.1.) The following properties of the directional derivatives are clear:

[1] A popular directional derivative in nonsmooth analysis, due to Clarke (1990), is to replace $h(x+td)$ with $h(y+td)$ for some sequence $y\to x$. The second-order counterpart appeared in Cominetti and Correa (1990). For our purpose here, the classical Dini definitions suffice.

Theorem A.1.2. *For any $\lambda \geq 0$ we have*

$$\mathsf{D}h(x;\lambda d) = \lambda \cdot \mathsf{D}h(x;d), \tag{A.10}$$

$$\mathsf{H}h(x;\lambda d, \lambda^2 g) = \lambda^2 \cdot \mathsf{H}h(x;d,g) \tag{A.11}$$

If h is locally Lipschitz around x, then $\mathsf{D}h(x;\cdot)$ and $\mathsf{H}h(x;d,\cdot)$ are Lipschitz continuous. (Similar results hold for the upper and lower derivatives.)

A.1.1 Necessary Conditions

Consider the nonsmooth optimization problem

$$\min_{x \in \mathcal{X} \subseteq \mathbb{R}^m} \; h(x). \tag{A.12}$$

We define three tangent cones of the (closed) constraint set $\mathcal{X}$:

$$\mathsf{K}_{\mathsf{f}}(\mathcal{X},x) := \{d : \forall \{t_k\} \to 0^+ \; \exists \{t_{k_i}\} \to 0^+, x + t_{k_i} d \in \mathcal{X}\} \subseteq \mathrm{cone}(\mathcal{X} - x) \tag{A.13}$$

$$\mathsf{K}_{\mathsf{d}}(\mathcal{X},x) := \liminf_{t \to 0^+} \frac{\mathcal{X} - x}{t} := \{d : \forall \{t_k\} \to 0^+ \; \exists \{t_{k_i}\} \to 0^+, \{d_{k_i}\} \to d, x + t_{k_i} d_{k_i} \in \mathcal{X}\} \tag{A.14}$$

$$\mathsf{K}_{\mathsf{c}}(\mathcal{X},x) := \limsup_{t \to 0^+} \frac{\mathcal{X} - x}{t} := \{d : \exists \{t_k\} \to 0^+, \; \{d_k\} \to d, x + t_k d_k \in \mathcal{X}\}. \tag{A.15}$$

Obviously, the (feasible) cone K_{f} is contained in the (derivable) cone K_{d}, which is itself contained in the (contingent) cone K_{c}. K_{d} and K_{c} are always closed while K_{f} may not be so (even when $\mathcal{X}$ is closed). On the other hand, if $\mathcal{X}$ is convex (and $x \in \mathcal{X}$), then all three tangent cones are convex, $\mathsf{K}_{\mathsf{f}} = \mathrm{cone}(\mathcal{X} - x)$ and $\mathsf{K}_{\mathsf{d}} = \mathsf{K}_{\mathsf{c}} = \overline{\mathsf{K}_{\mathsf{f}}}$. Note that for all tangent cones, we have

$$\forall x \notin \bar{\mathcal{X}}, \; \mathsf{K}(\mathcal{X},x) = \emptyset, \; \text{and } \forall x \in \mathcal{X}^\circ, \mathsf{K}(\mathcal{X},x) = \mathbb{R}^m, \tag{A.16}$$

where $\bar{\mathcal{X}}$ and $\mathcal{X}^\circ$ denote the closure and interior of $\mathcal{X}$, respectively. The following necessary condition is well-known:

Theorem A.1.3 (first-order necessary condition, e.g. Dem'yanov (1966))**.** *Let x^* be a local minimizer of h over $\mathcal{X}$. Then,*

$$\forall d \in \mathsf{K}_{\mathsf{f}}(\mathcal{X},x^*), \;\; \mathsf{D}_+ h(x^*;d) \geq 0. \tag{A.17}$$

The converse is also true if h and $\mathcal{X}$ are both convex around x^*. If h is locally Lipschitz, then

$$\forall d \in \mathsf{K}_{\mathsf{d}}(\mathcal{X},x^*), \;\; \mathsf{D}_+ h(x^*;d) \geq 0. \tag{A.18}$$

Proof. We first prove the converse part. Suppose to the contrary there exists x around x^* so that $h(x) < h(x^*)$. Then, $d = x - x^* \in \mathsf{K_f}(\mathcal{X}, x^*)$ and we have

$$\mathsf{D}_+h(x^*; d) = \liminf_{t\to 0^+} \frac{h((1-t)x^* + tx) - h(x^*)}{t} \leq h(x) - h(x^*) < 0, \tag{A.19}$$

which is a contradiction.

To see the claim when h is locally Lipschitz, note that $d \in \mathsf{K_d}(\mathcal{X}, x^*)$ implies for any $\{t_k\} \to 0$ there exist $\{t_{k_i}\} \to 0^+$ and $\{d_{k_i}\} \to d$ such that $x^* + t_{k_i} d_{k_i} \in \mathcal{X}$. For sufficiently large k_i we have $h(x^* + t_{k_i} d_{k_i}) \geq h(x^*)$ since x^* by assumption is a local minimizer. Thus,

$$\begin{aligned}
\liminf_{t\to 0^+} \frac{h(x^* + td) - h(x^*)}{t} &:= \lim_{t_k \to 0^+} \frac{h(x^* + t_k d) - h(x^*)}{t_k} && \text{(A.20)}\\
&\geq \limsup_{t_{k_i}\to 0^+} \frac{h(x^* + t_{k_i} d_{k_i}) - h(x^*)}{t_{k_i}} - \\
&\quad - \limsup_{t_{k_i}\to 0^+} \frac{h(x^* + t_{k_i} d) - h(x^* + t_{k_i} d_{k_i})}{t_{k_i}} && \text{(A.21)}\\
&\geq 0 - 0 = 0. && \text{(A.22)}
\end{aligned}$$

The proof for a general function h is similar. □

To derive second-order conditions, we define similarly the second-order tangent cones:

$$\mathsf{K_f}(\mathcal{X}, x; d) := \{g : \forall \{t_k\} \downarrow 0 \; \exists \{t_{k_i}\} \downarrow 0, x + t_{k_i} d + t_{k_i}^2 g/2 \in \mathcal{X}\}, \tag{A.23}$$

$$\begin{aligned}
\mathsf{K_d}(\mathcal{X}, x; d) &:= \liminf_{t\to 0^+} \frac{\mathcal{X} - x - td}{t^2/2} \\
&:= \{g : \forall \{t_k\} \downarrow 0 \; \exists \{t_{k_i}\} \downarrow 0, \{g_{k_i}\} \to g, x + t_{k_i} d + t_{k_i}^2 g_{k_i}/2 \in \mathcal{X}\}.
\end{aligned} \tag{A.24}$$

The proof of the following result is completely similar to that of Theorem A.1.3:

Theorem A.1.4 (second-order necessary condition, e.g. Ben-Tal and Zowe 1985)**.** *Let h be directionally differentiable and x^* be a local minimizer of h over $\mathcal{X}$. Then,*

$$\forall d \in \mathsf{K_f}(\mathcal{X}, x^*), \forall g \in \mathsf{K_f}(\mathcal{X}, x^*; d), \quad \mathsf{D}h(x^*; d) = 0 \implies \mathsf{H}_+h(x^*; d, g) \geq 0. \tag{A.25}$$

If h is locally Lipschitz, then

$$\forall d \in \mathsf{K_d}(\mathcal{X}, x^*), \forall g \in \mathsf{K_d}(\mathcal{X}, x^*; d), \quad \mathsf{D}h(x^*; d) = 0 \implies \mathsf{H}_+h(x^*; d, g) \geq 0. \tag{A.26}$$

A.1.2 Sufficient Conditions

We give sufficient conditions for a nonsmooth function to attain an isolated minimum.

Theorem A.1.5 (first-order, e.g. Dem'yanov 1970; Ben-Tal and Zowe 1985)**.** *Let h be locally Lipschitz. If*

$$\forall \mathbf{0} \neq d \in \mathsf{K_c}(\mathcal{X}, x^*),\ \mathsf{D}_+h(x^*; d) > 0, \tag{A.27}$$

then x^* is an isolated local minimum of h over $\mathcal{X}$.

Proof. Suppose to the contrary there exists a sequence $x_k \in \mathcal{X}$ converging to x^* so that $h(x_k) \leq h(x^*)$. Let $t_k := \|x_k - x^*\|$ and $d_k := (x_k - x^*)/\|x_k - x^*\|$. By passing to a subsequence we may assume $d_k \to d \neq \mathbf{0}$, where clearly $d \in \mathsf{K_c}(\mathcal{X}, x^*)$ since $x^* + t_k d_k = x_k \in \mathcal{X}$. But then

$$\begin{aligned}
\mathsf{D}_+h(x^*; d) &\leq \liminf_{t_k \to 0^+} \frac{h(x^* + t_k d) - h(x^*)}{t_k} && \text{(A.28)}\\
&\leq \liminf_{t_k \to 0^+} \frac{h(x^* + t_k d_k) - h(x^*)}{t_k} + \limsup_{t_k \to 0^+} \frac{h(x^* + t_k d) - h(x^* + t_k d_k)}{t_k} && \text{(A.29)}\\
&\leq 0 + 0 = 0, && \text{(A.30)}
\end{aligned}$$

arriving at a contradiction. □

Note that when $\mathcal{X}$ is convex, we may replace $\mathsf{K_c} = \overline{\mathsf{K_f}}$ with $\mathsf{K_f}$ (recall the Lipschitz continuity in Theorem A.1.2).

Theorem A.1.6 (second-order, e.g. Dem'yanov 1970)**.** *Let h be locally Lipschitz and directional differentiable, and $\mathcal{X}$ be convex. If*

1. $\forall d \in \mathsf{K_f}(\mathcal{X}, x^*),\ \mathsf{D}h(x^*; d) \geq 0$,
2. $\exists \gamma > 0$ *such that for all* $d \in \mathsf{K_f}(\mathcal{X}, x^*), \|d\| = 1, \mathsf{D}h(x^*; d) \in [0, \gamma]$ *we have for all small t and* uniformly *on bounded sets in d:*

$$\frac{h(x^* + td) - h(x^*) - t\mathsf{D}h(x^*; d)}{t^2/2} \geq \mathsf{A}_h(x^*; d) > 0, \tag{A.31}$$

then x^* is an isolated local minimum of h over $\mathcal{X}$.

Proof. Let $x \in \mathcal{X}$ and $x \neq x^*$, then $d := (x - x^*)/\|x - x^*\| \in \mathsf{K}_\mathsf{f}(\mathcal{X}, x^*)$ (since $\mathcal{X}$ is convex). Suppose $\mathsf{D}h(x^*, d) \geq \gamma > 0$, then

$$h(x^* + td) = h(x^*) + t\mathsf{D}h(x^*; d) + o(t) \geq h(x^*) + \gamma t + o(t) > h(x^*) + \gamma t/2, \tag{A.32}$$

for sufficiently small $t \leq t_d$. Since the function $d \mapsto h(x^* + td)$ is locally Lipschitz, we may choose a nonempty open subset from each set $\{v : \forall t \in (0, t_d],\ h(x^* + tv) > h(x^*)\}$. Hence, using a standard compactness argument, we know for all small positive t,

$$d \in \mathsf{K}_\mathsf{f}(\mathcal{X}, x^*), \|d\| = 1, \mathsf{D}h(x^*, d) \geq \gamma \implies h(x^* + td) > h(x^*). \tag{A.33}$$

Suppose instead $\mathsf{D}h(x^*, d) \in [0, \gamma]$, then for all small positive t and uniformly in d we have

$$\begin{aligned} h(x^* + td) &\geq h(x^*) + t\mathsf{D}h(x^*; d) + \tfrac{1}{2}t^2\mathsf{A}_h(x^*; d) && \text{(A.34)}\\ &\geq h(x^*) + \tfrac{1}{2}t^2\mathsf{A}_h(x^*; d) && \text{(A.35)}\\ &> h(x^*). && \text{(A.36)} \end{aligned}$$

Finally, combining the above two cases completes the proof. □

We make a few remarks regarding Theorem A.1.6:

- In general we cannot let $\gamma = 0$ (for an explicit counterexample, see Dem'yanov 1970). This is one of the subtleties to work with directional derivatives: even when $\mathsf{D}h(x^*; d)$ vanishes for some direction d we may still have $\mathsf{D}h(x^*; d)$ approaching 0 for other directions, but with $\gamma = 0$ we will not know how $\mathsf{A}_h(x^*; d)$ behaves (e.g. negative) along the latter directions.
- It is clear that $\mathsf{H}_+h \geq \mathsf{A}_h$. In some cases it is easier to verify the uniformity (along directions) in (A.31) if we relax the lower 2nd-order directional derivative H_+h to some convenient function A_h. See Theorem A.1.11 for an example.
- If $\mathcal{X} = \mathbb{R}^m$ and h is Fréchet differentiable with locally Lipschitz gradient ∇h around x^*, then we can verify the uniformity in (A.31) as follows. Note first that we have $\nabla h(x^*) = \mathbf{0}$ from the necessary condition. Second, for all small t we have

$$\begin{aligned} \frac{h(x^* + t\bar{d}) - h(x^*)}{t^2/2} &= \frac{h(x^* + td + t(\bar{d} - d)) - h(x^*)}{t^2/2} && \text{(A.37)}\\ &= \frac{h(x^* + td) - h(x^*) + t\left\langle \nabla h(x^* + \theta td) - \nabla h(x^*), \bar{d} - d \right\rangle}{t^2/2} && \text{(A.38)}\\ &\geq \frac{h(x^* + td) - h(x^*)}{t^2/2} - 2L\|d\|\|\bar{d} - d\|, && \text{(A.39)} \end{aligned}$$

where $\theta \in [0,1]$ and L is the local Lipschitz constant of ∇h. Thus, if $\frac{h(x^*+td)-h(x^*)}{t^2/2} > 0$ then for all nearby $\bar{d}$ we also have $\frac{h(x^*+t\bar{d})-h(x^*)}{t^2/2} > 0$. In this case we may let $\mathsf{A}_h = \mathsf{H}_+ h$ and recover (Ben-Tal and Zowe, 1985, Theorem 3.2).

Another result that directly uses the second-order derivative is:

Theorem A.1.7 (second-order sufficient condition, e.g. Dem'yanov and Malozemov 1974)**.** *Suppose h is uniformly first-order and second-order directional differentiable (at x^*) and $\mathcal{X}$ is convex. If there exist $r, q > 0$ such that for all normalized feasible direction t, $\mathsf{D}h(x^*;t) \geq 0$, and*

$$0 \leq \mathsf{D}h(x^*;t) < r \Longrightarrow \mathsf{H}h(x^*;t) \geq q > 0, \tag{A.40}$$

then x^* is an isolated local minimum.

Proof. If $\mathsf{D}h(x^*;t) \geq r$, it reduces to the proof of Thm. A.1.5. Otherwise, (A.40) holds, and

$$h(x^*+\alpha t) = h(x^*) + \alpha \mathsf{D}h(x^*;t) + \frac{\alpha^2}{2}\mathsf{H}h(x^*;t) + o(\alpha^2;t). \tag{A.41}$$

Since h is uniformly second-order directional differentiable in any direction t, there exist $0 < \alpha_1 < \alpha_0$ such that for any $0 < \alpha < \alpha_1$ and for any $\|t\| = 1$, $o(\alpha^2;t) \geq -q\alpha^2/4$. Therefore, for any $x \in \mathcal{N}(x^*, \alpha_1)$ not equal to x^*, we can take $t = (x-x^*)/\|x-x^*\|$ (which is feasible from convexity of $\mathcal{X}$) , $\alpha = \|x - x^*\|$ and obtain:

$$h(x) = h(x^*+\alpha t) \geq h(x^*) + \alpha^2 q/4 > h(x^*). \tag{A.42}$$

□

In the theorem above, we are considering "approximately" critical directions, rather than only the second order derivatives along the critical directions. The following example demonstrates this point, as inspired by Ben-Tal and Zowe (1985, Example 2.1):

Example A.1.8. *We cannot take $r = 0$ in (2.61). Consider $f((x_1,x_2),y) = (2x_1 + x_1^2 + x_2^2)y + x_1^3$ and $(x^*,y^*) = (\mathbf{0},0)$. $\bar{f}_\epsilon(x_1,x_2) = \epsilon|2x_1 + x_1^2 + x_2^2| + x_1^3$ and it is uniformly twice directional differentiable. We can evaluate $\mathsf{D}\bar{f}_\epsilon((0,0);(t_1,t_2)) = 2\epsilon|t_1|$ and*

$$\mathsf{H}\bar{f}_\epsilon((0,0);(t_1,t_2)) = \begin{cases} 2\epsilon(t_1^2+t_2^2) & t_1 > 0, \\ 2\epsilon t_2^2 & t_1 = 0, \\ -2\epsilon(t_1^2+t_2^2) & t_1 < 0. \end{cases}$$

The critical directions are $(0, t_2)$ *along which* $\mathsf{H}\bar{f}_\epsilon(\mathbf{0}, t) = 2\epsilon t_2^2 > 0$. *However,*

$$\bar{f}_\epsilon((0,0),(x_1,\sqrt{-2x_1 - x_1^2})) = x_1^3 < 0$$

if $-2 \le x_1 \le 0$.

A.1.3 Envelope Function

Our main interest in this section is the envelope function:

$$\bar{f}(x) := \max_{y \in \mathcal{Y}} \; f(x,y) \tag{A.43}$$

where $\mathcal{Y}$ is some compact topological Hausdorff space[2]. It is easy to verify:

- If $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is (jointly) continuous, then so is $\bar{f}$ (in x).
- If also $\partial_x f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is (jointly) continuous, then $\bar{f}$ is locally Lipschitz.

The envelope function turns out to be directionally differentiable:

Theorem A.1.9 (e.g. Danskin 1966; Dem'yanov 1966)**.** *Let* f *and* $\partial_x f$ *be (jointly) continuous. Then, the envelope function* $\bar{f}$ *is directionally differentiable:*

$$\mathsf{D}\bar{f}(x;d) = \max_{y \in \mathcal{Y}_0(x)} \; \langle \partial_x f(x,y), d \rangle \,, \;\; \textit{where} \;\; \mathcal{Y}_0(x) := \{y \in \mathcal{Y} : \bar{f}(x) = f(x;y)\}. \tag{A.44}$$

Clearly, $\mathsf{D}\bar{f}(x;\cdot)$ *is Lipschitz continuous.*

The following theorem explains the necessity of the function A_h in Theorem A.1.6:

Theorem A.1.10 (Seeger 1988; Dem'yanov 1970)**.** *Let* f *and* $\partial_x f$ *be continuous. Then,*

$$\mathsf{D}\bar{f}(x;d) = \max_{y \in \mathcal{Y}_0(x)} \langle \partial_x f(x,y), d \rangle \,, \quad \mathcal{Y}_0(x) := \{y \in \mathcal{Y} : \bar{f}(x) = f(x,y)\} \tag{A.45}$$

$$\mathsf{H}_+\bar{f}(x;d,g) \ge \max_{y \in \mathcal{Y}_1(x;d)} \mathsf{H}_+ f(x,y;d,g),$$

$$\mathcal{Y}_1(x;d) := \{y \in \mathcal{Y}_0(x) : \mathsf{D}\bar{f}(x;d) = \langle \partial_x f(x,y), d \rangle\}. \tag{A.46}$$

[2]Results in this section can be extended to the more general case where the constraint set $\mathcal{Y}$ depends on x (in some semicontinuous manner); see Seeger (1988) for an excellent treatment. For our purpose here it suffices to consider a constant $\mathcal{Y}$.

If $\partial^2_{xx} f$ is also (jointly) continuous, then

$$\mathsf{A}_{\bar{f}}(x;d) := \max_{y \in \mathcal{Y}_1(x;d)} \left\langle \partial^2_{xx} f(x,y) d, d \right\rangle \tag{A.47}$$

satisfies the uniformity condition in Theorem A.1.6.

Proof. We need only prove the last claim. Indeed

$$\begin{aligned} \frac{\bar{f}(x+td) - \bar{f}(x) - t\mathsf{D}\bar{f}(x;d)}{t^2/2} &\geq \max_{y \in \mathcal{Y}_1(x;d)} \frac{f(x+td,y) - f(x,y) - t\left\langle \partial_x f(x,y), d \right\rangle}{t^2/2} \\ &= \max_{y \in \mathcal{Y}_1(x;d)} \left\langle \partial^2_{xx} f(x + t\theta(y,d)\cdot d, y) d, d \right\rangle . \end{aligned} \tag{A.48}$$

Since $\partial^2_{xx} f$ is continuous (hence uniformly continuous over compact sets), the right-hand side converges to $\mathsf{A}_{\bar{f}}(x;d)$ uniformly on bounded sets in d as t goes to 0. □

When $\mathcal{Y}$ has limit points, proving $\mathsf{A}_{\bar{f}}(x;d) = \mathsf{H}\bar{f}(x;d)$ may be difficult (even with additional regularity conditions). Nevertheless, we can still apply the sufficient condition in Theorem A.1.6.

Seeger (1988) pointed out the following equivalence:

$$\mathsf{D}\bar{f}(x;d) = \max_{y \in \mathcal{Y}_0(x)} \mathsf{D}f(x,y;d) = \max_{y \in \mathcal{Y}_0(x)} \sup_{v \in \mathsf{K}_\mathsf{d}(\mathcal{Y},y)} \mathsf{D}f(x,y;(d,v)), \tag{A.49}$$

where the first two directional derivatives are taken wrt x only while the last directional derivative is joint wrt (x,y). Indeed, when f is (jointly) continuously differentiable, $\mathsf{D}f(x,y;(d,v)) = \langle \partial_x f(x,y), d\rangle + \langle \partial_y f(x,y), v\rangle$. However, since $y \in \mathcal{Y}_0(x)$, we know from the necessary condition in Theorem A.1.3 that $\langle \partial_y f(x,y), v\rangle \leq 0$ for all $v \in \mathsf{K}_\mathsf{d}(\mathcal{Y},y)$. Surprisingly, the second order counterparts are no longer equivalent:

Theorem A.1.11 (Seeger 1988)**.** *Let $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ be continuously differentiable. Then,*

$$\mathsf{H}_+\bar{f}(x;d,g) \geq \max_{y \in \mathcal{Y}_0(x)} \sup_{v \in \mathcal{V}(x,y;d)} \sup_{w \in \mathsf{K}_\mathsf{d}(\mathcal{Y},y;v)} \mathsf{H}_+ f(x,y;(d,v),(g,w)), \tag{A.50}$$

where $\mathcal{Y}_0(x) = \{y \in \mathcal{Y} : \bar{f}(x) = f(x,y)\}$ and $\mathcal{V}(x,y;d) := \{v \in \mathsf{K}_\mathsf{d}(\mathcal{Y},y) : \mathsf{D}\bar{f}(x;d) = \mathsf{D}f(x,y;(d,v))\}$.

If the second-order derivative of f is also (jointly) continuous, then

$$\begin{aligned} \mathsf{A}_{\bar{f}}(x;d) := \max_{y \in \mathcal{Y}_0(x)} \sup_{v \in \mathcal{V}(x,y;d)} \sup_{w \in \mathsf{K}_\mathsf{d}(\mathcal{Y},y;v)} & \left\langle \begin{bmatrix} \partial^2_{xx} f(x,y) & \partial^2_{xy} f(x,y) \\ \partial^2_{yx} f(x,y) & \partial^2_{yy} f(x,y) \end{bmatrix} \begin{pmatrix} d \\ v \end{pmatrix}, \begin{pmatrix} d \\ v \end{pmatrix} \right\rangle + \\ & + \langle \partial_y f(x,y), w \rangle \end{aligned} \tag{A.51}$$

satisfies the uniformity condition in Theorem A.1.6, provided that the directions d, v and w are bounded.

Proof. We assume $\mathsf{K}_\mathsf{d}(\mathcal{Y}, y; v)$ is not empty for otherwise the theorem is vacuous. For any $w \in \mathsf{K}_\mathsf{d}(\mathcal{Y}, y; v)$ we know for any sequence $t_k \downarrow 0$ there exist a subsequence $t_{k_i} \downarrow 0$ and $w_{k_i} \to w$ such that $y + t_{k_i} v + t_{k_i}^2 w_k \in \mathcal{Y}$. Thus, fix any $y \in \mathcal{Y}_0(x)$, $v \in \mathcal{V}(x, y; d)$ and $w \in \mathsf{K}_\mathsf{d}(\mathcal{Y}, y; v)$, we know (after passing to a subsequence if necessary)

$$\frac{\bar{f}(x + t_k d + t_k^2 g/2) - \bar{f}(x) - t_k \mathsf{D}\bar{f}(x; d)}{t_k^2/2} \tag{A.52}$$

$$\geq \frac{f(x + t_k d + t_k^2 g/2, y + t_k v + t_k^2 w_k/2) - f(x, y) - t_k \mathsf{D}f(x, y; (d, v))}{t_k^2/2} \tag{A.53}$$

$$\geq \frac{f(x + t_k d + t_k^2 g/2, y + t_k v + t_k^2 w/2) - f(x, y) - t_k \mathsf{D}f(x, y; (d, v))}{t_k^2/2} + \tag{A.54}$$

$$+ \frac{f(x + t_k d + t_k^2 g/2, y + t_k v + t_k^2 w_k/2) - f(x + t_k d + t_k^2 g/2, y + t_k v + t_k^2 w/2)}{t_k^2/2} \tag{A.55}$$

$$= \mathsf{H}_+ f(x, y; (d, v), (g, w)) + o(t_k), \tag{A.56}$$

where the small order term $o(t_k)$ is independent of d, v and w if they are bounded. □

By setting $y \in \mathcal{Y}_1(x; d), v = w = \mathbf{0}$, we see that the lower bounds in Theorem A.1.11 are always shaper than the ones in Theorem A.1.10. However, note that Theorem A.1.10 only requires $\mathcal{Y}$ to be any compact topological space while Theorem A.1.11 only applies when $\mathcal{Y}$ is a compact set of some finite dimensional vector space.

Example A.1.12 (Seeger 1988)**.** *Let* $\mathcal{Y} = \mathbb{R}^m$ *and* $f(x, y) = \binom{x}{y}^\top \left\{ \frac{1}{2} \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix} \binom{x}{y} + \binom{\mathbf{p}}{\mathbf{q}} \right\}$. *Assume* $C \prec \mathbf{0}$. *Then,* $\mathcal{Y}_0(x)$ *is a singleton,* $\mathcal{Y}_1 = \mathbb{R}^m$, *and WLOG* $w = \mathbf{0}$. *Therefore,*

$$\mathsf{A}_{\bar{f}}(x; d) = d^\top (A - BC^{-1}B^\top) d, \tag{A.57}$$

whence $(x, y) = \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix}^{-1} \binom{\mathbf{p}}{\mathbf{q}}$ *is a (unique) global saddle point if* $C \prec \mathbf{0}$ *and* $A - BC^{-1}B^\top \succ \mathbf{0}$.

However, if we apply Theorem A.1.10 we can only conclude that

$$\mathsf{A}_{\bar{f}}(x; d) = d^\top A d, \tag{A.58}$$

which is clearly a looser lower bound (recall that $C \prec \mathbf{0}$*).*

In principle, one should use the lower second-order directional derivative) $\mathsf{H}_+(x^*; d, g) \geq 0$ for a stronger necessary condition. However, to our knowledge, we do not have an appropriate formula for it. We therefore look into *upper* second-order derivatives instead for which Kawasaki (1988) showed a result. From this result, we are able to introduce the second-order necessary conditions for x^* being a local minimizer of $\bar{f}(x)$:

Theorem A.1.13 (Kawasaki 1988)**.** *Let f be twice (jointly) continuously differentiable. Then,*

$$\mathsf{H}\bar{f}(x; d, g) = \max_{y \in \mathcal{Y}_1(x,d)} \langle \partial_x f(x, y), g \rangle + \left\langle \partial^2_{xx} f(x, y)d, d \right\rangle + \limsup_{z \to y} \tfrac{1}{2} v^2_-(z; d)u^\dagger(z), \quad \text{(A.59)}$$

where $(t)_- = \min\{t, 0\}$, $t^\dagger = \begin{cases} 1/t, & t \neq 0 \\ 0, & t = 0 \end{cases}$, *and*

$$u(y) := \bar{f}(x) - f(x, y) \geq 0,\ v(y; d) := \mathsf{D}\bar{f}(x; d) - \mathsf{D}f(x, y; d). \quad \text{(A.60)}$$

Proof. We give a direct (and arguably simpler) proof of this result. Denote

$$\Delta(t) := \frac{\bar{f}(x + td + t^2 g/2) - \bar{f}(x) - t\mathsf{D}\bar{f}(x; d)}{t^2/2}. \quad \text{(A.61)}$$

Using the definitions of u and v we have

$$\Delta(t) = \frac{\bar{f}(x + td + t^2 g) - f(x, z) - t\mathsf{D}f(x, z; d) - u(z) - tv(z; d)}{t^2/2}, \quad \text{(A.62)}$$

which holds for any $z \in \mathcal{Y}$. Let us first choose $z = z_t \in \mathcal{Y}_0(x + td + t^2 g)$:

$$\Delta(t) = \frac{f(x + td + t^2 g, z_t) - f(x, z_t) - t\mathsf{D}f(x, z_t; d)}{t^2/2} - \frac{u(z_t) + tv(z_t; d)}{t^2/2}. \quad \text{(A.63)}$$

Let $y \in \mathcal{Y}_0(x)$ be a limit point of z_t. Suppose $y \in \mathcal{Y}_0(x) \setminus \mathcal{Y}_1(x; d)$. Then, for small t we have (in the corresponding subsequence) $v(z_t; d) \approx v(y; d) > 0$ hence $\liminf_t \Delta(t) = \mathsf{H}_+\bar{f}(x; d, g) = -\infty$, contradicting Theorem A.1.10. Thus, $y \in \mathcal{Y}_1(x; d)$. Optimizing t for the second term we obtain

$$\Delta(t) \leq \frac{f(x + td + t^2 g, z_t) - f(x, z_t) - t\mathsf{D}f(x, z_t; d)}{t^2/2} + \tfrac{1}{2} v^2_-(z_t; d)u^\dagger(z_t), \quad \text{(A.64)}$$

where we used the fact that if $u(z_t) = 0$ then $v(z_t; d) \geq 0$ (see Theorem A.1.9). Taking limits on both sides proves the $\leq$ part in (A.59).

For the converse, let $y \in \mathcal{Y}_1(x;d)$ and $z_k \to y$ attain the maximum and limsup in (A.59), respectively. We need only consider $\lim_{z_k \to y} \frac{1}{2} v_-^2(z_k;d) u^\dagger(z_k) > 0$, for otherwise the $\geq$ part in (A.59) would already follow from Theorem A.1.10. We obviously have $u(z_k) > 0$ and $v(z_k;d) < 0$ for sufficiently large t. Since $u(z_k) \to u(y) = 0$ we also have $v(z_k;d) \to v(y;d) = 0$. We claim that (after passing to a subsequence if necessary) $\lim_k u(z_k)/v(z_k;d) = 0$, for otherwise $\lim v^2(z_k;d)/u(z_k) = 0$, contradicting to its strict positivity. Now, setting $t_k = -2u(z_k)/v(z_k;d)$ we have (for large k):

$$\Delta(t_k) \geq \frac{f(x + t_k d + t_k^2 g, z_k) - f(x, z_k) - t_k \mathsf{D} f(x, z_k; d) - u(z_k) - t_k v(z_k; d)}{t_k^2/2} \tag{A.65}$$

$$= \frac{f(x + t_k d + t_k^2 g, z_k) - f(x, z_k) - t_k \mathsf{D} f(x, z_k; d)}{t_k^2/2} + \tfrac{1}{2} v_-^2(z_k;d) u^\dagger(z_k). \tag{A.66}$$

Taking limits on both sides we obtain the $\geq$ part in (A.59). □

For later convenience, we remind that

$$\mathcal{Y}_0(x) = \{y : u(y) = 0\},\ \mathcal{Y}_1(x;d) = \{y : u(y) = v(y;d) = 0\}. \tag{A.67}$$

and denote $\bar{E}(y;t) = \limsup_{z \to y} \frac{1}{2} v_-^2(z;d) u^\dagger(z)$.

With Carathédory's theorem for convex hulls, one can obtain from (A.59) the following necessary condition for envelope functions:

Theorem A.1.14 (**Kawasaki (1991)**). *Assume $f \in \mathcal{C}^2$ and $\mathcal{X} = \mathbb{R}^n$. If x^* is a local minimum of $\bar{f}(x)$, then for each $d \in \mathbb{R}^n$ satisfying $\mathsf{D}\bar{f}(x^*;d) = 0$, there exist at most $n+1$ points $y_1, \ldots, y_{n+1} \in \mathcal{Y}_1(x^*;d)$ and $\lambda_1, \ldots, \lambda_n \geq 0$ not all zero, such that:*

$$\sum_{i=1}^{a} \lambda_i \partial_x f(x^*, y_i) = \mathbf{0},\ \sum_{i=1}^{a} \lambda_i \left(d^\top \partial_{xx}^2 f(x^*, y_i) d + \bar{E}(y_i; d) \right) \geq 0. \tag{A.68}$$

Proof. We borrow the result from Kawasaki (1991). In order to write down the second-order derivative formula in Kawasaki (1988), we define

$$Y_0(t) := \{y \in \mathcal{Y} : \text{there exists a sequence } \{z_k\} \to y,\ u(z_k) > 0 \text{ and } v(z_k;t)/u(z_k) \to -\infty\},$$

and the following upper semi-continuous function (Kawasaki, 1988):

$$\bar{E}'(y;t) = \begin{cases} \sup_{\{z_k\} \to y} \limsup_k v(z_k;t)^2/(2u(z_k)) & y \in Y_0(t) \text{ and } \{z_k\} \text{ is in } Y_0(t), \\ 0 & u(y) = v(y;t) = 0 \ \&\ y \notin Y_0(t) \\ -\infty & \text{otherwise.} \end{cases} \tag{A.69}$$

As shown in Kawasaki (1991), $u(y) = v(y;t) = 0$ whenever $y \in Y_0(t)$. We simplify the definition above:

Lemma A.1.15. *Denoting* $x_- := \min\{x, 0\}$, $x^\dagger = 1/x$ *if* $x \neq 0$ *and* $x^\dagger = 0$ *otherwise, then for any* $u(y) = v(y;t) = 0$,

$$\bar{E}(y;t) = \limsup_{z_k \to y} v_-(z_k;t)^2 u^\dagger(z_k)/2. \tag{A.70}$$

Proof. It suffices to consider those sequences $\{z_k\} \subset \mathcal{Y}$ such that $u(z_k) \geq 0$. We want to prove that $\bar{E}(y;t) = \bar{E}'(y;t)$. We first prove $\bar{E}(y;t) \geq \bar{E}'(y;t)$. If $y \in Y_0(t)$, then for any $\delta > 0$, there exists a sequence $\{z_k\}$ such that

$$\limsup_k v(z_k;t)^2/(2u(z_k)) \geq \bar{E}'(y;t) - \delta,$$

$u(z_k) > 0$ and $v(z_k;t)/u(z_k) \to -\infty$. For large enough m, $v(z_k;t) < 0$, and thus we take the same sequence in (A.70) to obtain $\bar{E}(y;t) \geq \bar{E}'(y;t) - \delta$. Since the above holds for any $\delta > 0$, we have $\bar{E}(y;t) \geq \bar{E}'(y;t)$. If $y \notin Y_0(t)$, then $\bar{E}(y;t) \geq 0 = \bar{E}'(y;t)$.

Now let us prove that $\bar{E}(y;t) \leq \bar{E}'(y;t)$. Assume for any $\delta > 0$, $\{z_k\}$ is the sequence s.t.

$$\limsup_k v_-(z_k;t)^2 u^\dagger(z_k)/2 \geq \bar{E}(y;t) - \delta.$$

If $u(z_k) > 0$ or $v(z_k;t) < 0$ for finite number of m, then $\bar{E}(y;t) = 0 \leq \bar{E}'(y;t)$. Assume WLOG now that for any m, $u(z_k) > 0$ and $v(z_k;t) < 0$, if $v(z_k;t)/u(z_k)$ is bounded, then since $v(y;t) = 0$, $\bar{E}(y;t) = 0 \leq \bar{E}'(y;t)$. So we can assume further that $v(z_k;t)/u(z_k) \to -\infty$. Using the same sequence in (A.69), we know $\bar{E}'(y;t) \geq \bar{E}(y;t) - \delta$ for any $\delta > 0$, and thus $\bar{E}'(y;t) \geq \bar{E}(y;t)$. □

□

Moreover, the following assumption guarantees the existence of $\mathsf{H}\bar{f}(x;d,g)$ from which we can get second-order sufficient conditions:

Assumption A.1.16 (Kawasaki (1992)). *For each* $y \in \mathcal{Y}_1(x^*;t)$ *with* $t \neq \mathbf{0}$ *and* $\mathsf{D}\bar{f}(x^*;t) = 0$, *and for each non-zero* $d \in \mathbb{R}^m$, *there exist* $\alpha, \beta \neq 0$ *and* $p, q > 0$ *such that the following approximation holds:*

$$u(y + \delta d) = \alpha\delta^p + o(\delta^p),\ v(y + \delta d;t) = \beta\delta^q + o(\delta^q), \tag{A.71}$$

whenever $y + \delta d \in \mathcal{N}(y^*, \epsilon)$ *and* $\delta > 0$. *Note that*

$$u(y) := \bar{f}(x^*) - f(x^*, y),\ v(y;d) := \mathsf{D}\bar{f}(x^*;d) - \mathsf{D}f(x^*, y;d).$$

Theorem A.1.17 (**second-order sufficient condition, Kawasaki (1992)**). *Assume Assumption A.1.16 holds at x^*. Let $\mathcal{X} = \mathbb{R}^n$ and $\mathcal{Y}$ be convex. x^* is an isolated local minimum of $\bar{f}(x)$ if for any $d \in \mathbb{R}^n$, $\mathsf{D}\bar{f}(x^*; d) > 0$, or $\mathsf{D}\bar{f}(x^*; d) = 0$, $d \neq \mathbf{0}$ and there exist $a \geq 1$ points $y_1, \ldots, y_a \in \mathcal{Y}_1(x^*; d)$ and $\lambda_1, \ldots, \lambda_a > 0$ such that:*

$$\sum_{i=1}^{a} \lambda_i \partial_x f(x^*, y_i) = \mathbf{0}, \sum_{i=1}^{a} \lambda_i \left(d^\top \partial_{xx}^2 f(x^*, y_i) d + \bar{E}(y_i; d) \right) > 0. \tag{A.72}$$

Appendix B

Supplementary Material for Chapter 3

B.1 Proofs

We present full proofs of OGD and the Momentum method (Heavy Ball) in this appendix.

B.1.1 Proof of OGD

Theorem 3.2.5 (OGD). *For generalized OGD with $\alpha_1 = \alpha_2 = \alpha$, Jacobi and Gauss–Seidel updates achieve linear convergence iff for any singular value σ of E, we have:*

$$\text{J}: \quad \begin{cases} |\beta_1\beta_2\sigma^2| < 1,\ (\alpha-\beta_1)(\alpha-\beta_2) > 0,\ 4+(\alpha+\beta_1)(\alpha+\beta_2)\sigma^2 > 0, \\ \alpha^2\left(\beta_1^2\sigma^2+1\right)\left(\beta_2^2\sigma^2+1\right) < (\beta_1\beta_2\sigma^2+1)(2\alpha(\beta_1+\beta_2)+\beta_1\beta_2(\beta_1\beta_2\sigma^2-3)); \end{cases} \tag{3.58}$$

$$\text{GS}: \quad \begin{cases} (\alpha-\beta_1)(\alpha-\beta_2) > 0, (\alpha+\beta_1)(\alpha+\beta_2)\sigma^2 < 4, \\ (\alpha\beta_1\sigma^2+1)(\alpha\beta_2\sigma^2+1) > (1+\beta_1\beta_2\sigma^2)^2. \end{cases} \tag{3.59}$$

*The convergence region of GS updates **strictly** includes that of Jacobi updates.*

Proof. The Jacobi characteristic polynomial is now quartic in the form $\lambda^4+a\lambda^3+b\lambda^2+c\lambda+d$, with

$$a = -2,\ b = \alpha^2\sigma^2+1,\ c = -\alpha(\beta_1+\beta_2)\sigma^2,\ d = \beta_1\beta_2\sigma^2. \tag{B.1}$$

Comparably, the GS polynomial (3.57) can be reduced to a cubic one $\lambda^3 + a\lambda^2 + b\lambda + c$ with

$$a = -2 + \alpha^2\sigma^2,\ b = -\alpha(\beta_1 + \beta_2)\sigma^2 + 1,\ c = \beta_1\beta_2\sigma^2. \tag{B.2}$$

First we derive the Schur conditions (3.58) and (3.59). Note that other than Corollary 3.1.3, an equivalent Schur condition can be read from Cheng and Chiou (2007, Theorem 1) as:

Theorem B.1.1 (**Cheng and Chiou (2007)**). *A real quartic polynomial* $\lambda^4 + a\lambda^3 + b\lambda^2 + c\lambda + d$ *is Schur stable iff:*

$$\begin{aligned} &|d| < 1,\ |a| < d + 3,\ |a + c| < b + d + 1, \\ &(1 - d)^2 b + c^2 - a(1 + d)c - (1 + d)(1 - d)^2 + a^2 d < 0. \end{aligned} \tag{B.3}$$

With (B.1) and Theorem B.1.1, it is straightforward to derive (3.58). With (B.2) and Corollary 3.1.3, we can derive (3.59) without much effort.

Now, let us study the relation between the convergence region of Jacobi OGD and GS OGD, as given in (3.58) and (3.59). Namely, we want to prove the last sentence of Theorem 3.2.5. The outline of our proof is as follows. We first show that each region of $(\alpha, \beta_1, \beta_2)$ described in (3.58) (the Jacobi region) is contained in the region described in (3.59) (the GS region). Since we are only studying one singular value, we slightly abuse the notations and rewrite $\beta_i\sigma$ as β_i $(i = 1, 2)$ and $\alpha\sigma$ as α. From (3.56) and (3.57), β_1 and β_2 can switch. WLOG, we assume $\beta_1 \geq \beta_2$. There are four cases to consider:

- $\beta_1 \geq \beta_2 > 0$. The third Jacobi condition in (3.58) now is redundant, and we have $\alpha > \beta_1$ or $\alpha < \beta_2$ for both methods. Solving the quadratic feasibility condition for α gives:

$$0 < \beta_2 < 1,\ \beta_2 \leq \beta_1 < \frac{\beta_2 + \sqrt{4 + 5\beta_2^2}}{2(1 + \beta_2^2)},\ \beta_1 < \alpha < \frac{u + \sqrt{u^2 + tv}}{t}, \tag{B.4}$$

  where $u = (\beta_1\beta_2 + 1)(\beta_1 + \beta_2)$, $v = \beta_1\beta_2(\beta_1\beta_2 + 1)(\beta_1\beta_2 - 3)$, $t = (\beta_1^2 + 1)(\beta_2^2 + 1)$. On the other hand, assume $\alpha > \beta_1$, the first and third GS conditions are automatic. Solving the second gives:

$$0 < \beta_2 < 1,\ \beta_2 \leq \beta_1 < \frac{-\beta_2 + \sqrt{8 + \beta_2^2}}{2},\ \beta_1 < \alpha < -\frac{1}{2}(\beta_1 + \beta_2) + \frac{1}{2}\sqrt{(\beta_1 - \beta_2)^2 + 16}. \tag{B.5}$$

Define $f(\beta_2) := -\beta_2 + \sqrt{8+\beta_2^2}/2$ and $g(\beta_2) := (\beta_2 + \sqrt{4+5\beta_2^2})/(2(1+\beta_2^2))$, and one can show that

$$f(\beta_2) \geq g(\beta_2). \tag{B.6}$$

Furthermore, it can also be shown that given $0 < \beta_2 < 1$ and $\beta_2 \leq \beta_1 < g(\beta_2)$, we have

$$(u + \sqrt{u^2+4v})/t < -(\beta_1+\beta_2)/2 + (1/2)\sqrt{(\beta_1-\beta_2)^2+16}. \tag{B.7}$$

- $\beta_1 \geq \beta_2 = 0$. The Schur condition for Jacobi and Gauss–Seidel updates reduces to:

$$\text{Jacobi: } 0 < \beta_1 < 1,\ \beta_1 < \alpha < \frac{2\beta_1}{1+\beta_1^2}, \tag{B.8}$$

$$\text{GS: } 0 < \beta_1 < \sqrt{2},\ \beta_1 < \alpha < \frac{-\beta_1 + \sqrt{16+\beta_1^2}}{2}. \tag{B.9}$$

  One can show that given $\beta_1 \in (0,1)$, we have $2\beta_1/(1+\beta_1^2) < (-\beta_1 + \sqrt{16+\beta_1^2})/2$.

- $\beta_1 \geq 0 > \beta_2$. Reducing the first, second and fourth conditions of (3.58) yields:

$$\beta_2 < 0,\ 0 < \beta_1 < \frac{\beta_2 + \sqrt{4+5\beta_2^2}}{2(1+\beta_2^2)},\ \beta_1 < \alpha < \frac{u+\sqrt{u^2+tv}}{t}. \tag{B.10}$$

  This region contains the Jacobi region. It can be similarly proved that even within this larger region, GS Schur condition (3.59) is always satisfied.

- $\beta_2 \leq \beta_1 < 0$. We have $u < 0$, $tv < 0$ and thus $\alpha < (u + \sqrt{u^2+tv})/t < 0$. This contradicts our assumption that $\alpha > 0$.

Combining the four cases above, we know that the Jacobi region is contained in the GS region.

To show the strict inclusion, take $\beta_1 = \beta_2 = \alpha/5$ and $\alpha \to 0$. One can show that as long as α is small enough, all the Jacobi regions do not contain this point, each of which is described with a singular value in (3.58). However, all the GS regions described in (3.59) contain this point.

The proof above is still missing some details. We provide the proofs of (B.4), (B.6), (B.7) and (B.10) in the sub-sub-sections below, with the help of Mathematica, although one can also verify these claims manually. Moreover, a one line proof of the inclusion can be given with Mathematica code, as shown in Section B.1.1.

Proof of equation B.4

The fourth condition of (3.58) can be rewritten as:

$$\alpha^2 t - 2u\alpha - v < 0, \tag{B.11}$$

where $u = (\beta_1\beta_2 + 1)(\beta_1 + \beta_2)$, $v = \beta_1\beta_2(\beta_1\beta_2 + 1)(\beta_1\beta_2 - 3)$, $t = (\beta_1^2 + 1)(\beta_2^2 + 1)$. The discriminant is $4(u^2 + tv) = (1 - \beta_1\beta_2)^2(1 + \beta_1\beta_2)(\beta_1^2 + \beta_2^2 + \beta_1^2\beta_2^2 - \beta_1\beta_2) \geq 0$. Since if $\beta_1\beta_2 < 0$,

$$\beta_1^2 + \beta_2^2 + \beta_1^2\beta_2^2 - \beta_1\beta_2 = \beta_1^2 + \beta_2^2 + \beta_1\beta_2(\beta_1\beta_2 - 1) > 0,$$

If $\beta_1\beta_2 \geq 0$,

$$\beta_1^2 + \beta_2^2 + \beta_1^2\beta_2^2 - \beta_1\beta_2 = (\beta_1 - \beta_2)^2 + \beta_1\beta_2(1 + \beta_1\beta_2) \geq 0,$$

where we used $|\beta_1\beta_2| < 1$ in both cases. So, (B.11) becomes:

$$\frac{u - \sqrt{u^2 + tv}}{t} < \alpha < \frac{u + \sqrt{u^2 + tv}}{t}. \tag{B.12}$$

Combining with $\alpha > \beta_1$ or $\alpha < \beta_2$ obtained from the second condition, we have:

$$\frac{u - \sqrt{u^2 + tv}}{t} < \alpha < \beta_2 \text{ or } \beta_1 < \alpha < \frac{u + \sqrt{u^2 + tv}}{t}. \tag{B.13}$$

The first case is not possible, with the following code:

```
u = (b1 b2 + 1) (b1 + b2); v = b1 b2 (b1 b2 + 1) (b1 b2 - 3);
t = (b1^2 + 1) (b2^2  + 1);
Reduce[b2 t > u - Sqrt[u^2 + t v] && b1 >= b2 > 0
&& Abs[b1 b2] < 1],
```

and we have:

```
False.
```

Therefore, the only possible case is $\beta_1 < \alpha < (u + \sqrt{u^2 + tv})/t$. Where the feasibility region can be solved with:

```
Reduce[b1 t < u + Sqrt[u^2+t v]&&b1>=b2>0&&Abs[b1 b2] < 1].
```

What we get is:

```
0<b2<1 &&
b2<=b1<b2/(2 (1+b2^2))+1/2 Sqrt[(4+5 b2^2)/(1+b2^2)^2].
```

Therefore, we have proved (B.4).

Proof of equation B.6

With

```
Reduce[-(b2/2) + Sqrt[8 + b2^2]/2 >=
(b2 + Sqrt[4 + 5 b2^2])/(2 (1 + b2^2)) && 0 < b2 < 1],
```

we can remove the first constraint and get:

```
0 < b2 < 1.
```

Proof of equation B.7

Given

```
Reduce[-1/2 (b1 + b2) + 1/2 Sqrt[(b1 - b2)^2 + 16] >
(u + Sqrt[u^2 + t v])/t &&
0 < b2 < 1 &&
b2 <= b1 < (b2 + Sqrt[4 + 5 b2^2])/(2 (1 + b2^2)), {b2, b1}],
```

we can remove the first constraint and get:

```
0 < b2 < 1 &&
b2 <= b1 < b2/(2 (1 + b2^2)) +
1/2 Sqrt[(4 + 5 b2^2)/(1 + b2^2)^2].
```

Proof of equation B.10

The second Jacobi condition simplifies to $\alpha > \beta_1$ and the fourth simplifies to (B.12). Combining with the first Jacobi condition:

```
Reduce[Abs[b1 b2] < 1 &&
a > b1 && (u - Sqrt[u^2 + t v])/t < a < (u + Sqrt[u^2 + t v])/t
&& b1 >= 0 && b2 < 0, {b2, b1, a} ] // Simplify,
```

we have:

```
   b2 < 0 && b1 > 0 &&
b2/(1 + b2^2) + Sqrt[(4 + 5 b2^2)/(1 + b2^2)^2] > 2 b1 &&
b1 < a < (b1 + b2 + b1^2 b2 + b1 b2^2)/((1 + b1^2) (1 + b2^2)) +
  Sqrt[((-1 + b1 b2)^2 (b1^2 + b2^2 + b1 b2 (-1 + b2^2) +
     b1^3 (b2 + b2^3)))/((1 + b1^2)^2 (1 + b2^2)^2)].
```

This can be further simplified to achieve (B.10).

One line proof

In fact, there is another very simple proof:

```
Reduce[ForAll[{b1, b2, a}, (a - b1) (a - b2) > 0
&& (a + b1) (a + b2) > -4 && Abs[b1 b2] < 1 &&
a^2 (b1^2 + 1) (b2^2 + 1) < (b1 b2 + 1) (2 a (b1 + b2) +
b1 b2 (b1 b2 - 3)), (a - b1) (a - b2) > 0 &&
(a + b1) (a + b2) < 4
&& (a b1 + 1) (a b2 + 1) > (1 + b1 b2)^2], {b2, b1, a}]
True.
```

However, this proof does not tell us much information about the range of our variables.

□

B.1.2 Proof of Momentum

Theorem 3.2.6 (**momentum**). *For the generalized momentum method with $\alpha_1 = \alpha_2 = \alpha$, the Jacobi updates never converge, while the GS updates converge iff for any singular value σ of E, we have:*

$$|\beta_1\beta_2| < 1, |-\alpha^2\sigma^2 + \beta_1 + \beta_2 + 2| < \beta_1\beta_2 + 3,\ 4(\beta_1+1)(\beta_2+1) > \alpha^2\sigma^2,$$
$$\alpha^2\sigma^2\beta_1\beta_2 < (1-\beta_1\beta_2)(2\beta_1\beta_2 - \beta_1 - \beta_2). \tag{3.64}$$

*This condition implies that at least one of β_1, β_2 is **negative**.*

Proof. **Jacobi condition** We first rename $\alpha\sigma$ as `a1` and β_1, β_2 as `b1`, `b2`. With Theorem B.1.1:

```
  {Abs[d] < 1, Abs[a] < d + 3,
 a + b + c + d + 1 > 0, -a + b - c + d + 1 >
  0, (1 - d)^2 b  - (c - a d) (a - c) - (1 + d) (1 - d)^2 <
  0} /. {a -> -2 - b1 - b2, b -> al^2 + 1 + 2 (b1 + b2) + b1 b2,
 c -> -b1 - b2 - 2 b1 b2, d -> b1 b2} // FullSimplify.
```

We obtain:

```
  {Abs[b1 b2] < 1, Abs[2 + b1 + b2] < 3 + b1 b2, al^2 > 0,
 al^2 + 4 (1 + b1) (1 + b2) > 0, al^2 (-1 + b1 b2)^2 < 0}.
```

The last condition is never satisfied and thus Jacobi momentum never converges.

Gauss–Seidel condition With Theorem B.1.1, we compute:

```
{Abs[d] < 1, Abs[a] < d + 3,
 a + b + c + d + 1 > 0, -a + b - c + d + 1 >
  0, (1 - d)^2 b  + c^2 - a (1 + d) c - (1 + d) (1 - d)^2 + a^2 d <
  0} /. {a -> al^2 - 2 - b1 - b2, b -> 1 + 2 (b1 + b2) + b1 b2,
 c -> -b1 - b2 - 2 b1 b2, d -> b1 b2} // FullSimplify.
```

The result is:

```
{Abs[b1 b2] < 1, Abs[2 - al^2 + b1 + b2] < 3 + b1 b2, al^2 > 0,
 4 (1 + b1) (1 + b2) > al^2,
 al^2 (b1 + b2 + (-2 + al^2 - b1) b1 b2 + b1 (-1 + 2 b1) b2^2) < 0},
```

which can be further simplified to (3.64).

Negative momentum With Theorem 3.2.6, we can actually show that in general at least one of β_1 and β_2 must be negative. There are three cases to consider, and in each case we simplify (3.64):

1. $\beta_1\beta_2 = 0$. WLOG, let $\beta_2 = 0$, and we obtain

$$-1 < \beta_1 < 0 \text{ and } \alpha^2\sigma^2 < 4(1+\beta_1). \tag{B.14}$$

2. $\beta_1\beta_2 > 0$. We have

$$-1 < \beta_1 < 0,\ -1 < \beta_2 < 0\,, \alpha^2\sigma^2 < 4(1+\beta_1)(1+\beta_2). \tag{B.15}$$

3. $\beta_1\beta_2 < 0$. WLOG, we assume $\beta_1 \geq \beta_2$. We obtain:

$$-1 < \beta_2 < 0, 0 < \beta_1 < \min\left\{-\frac{1}{3\beta_2}, \left|-\frac{\beta_2}{1+2\beta_2}\right|\right\}. \tag{B.16}$$

The constraints for α are $\alpha > 0$ and:

$$\max\left\{\frac{(1-\beta_1\beta_2)(2\beta_1\beta_2-\beta_1-\beta_2)}{\beta_1\beta_2}, 0\right\} < \alpha^2\sigma^2 < 4(1+\beta_1)(1+\beta_2). \tag{B.17}$$

These conditions can be further simplified by analyzing all singular values. They only depend on σ_1 and σ_n, the largest and the smallest singular values. Now, let us derive (B.15), (B.16) and (B.17) more carefully. Note that we use `a` for $\alpha\sigma$.

Proof of equation B.15

By running the following `Mathematica` code:

```
  Reduce[Abs[b1 b2] < 1 && Abs[-a^2 + b1 + b2 + 2] < b1 b2 + 3 &&
4 (b1 + 1) (b2 + 1) > a^2 &&
a^2 b1 b2 < (1 - b1 b2) (2 b1 b2 - b1 - b2) && b1 b2 > 0 &&
a > 0, {b2, b1, a}]
```

we obtain:

```
-1 < b2 < 0 && -1 < b1 < 0 && 0 < a < Sqrt[4 + 4 b1 + 4 b2 + 4 b1 b2]
```

Proof of equations B.16 and B.17

By running the following `Mathematica` code:

```
  Reduce[Abs[b1 b2] < 1 && Abs[-a^2 + b1 + b2 + 2] < b1 b2 + 3 &&
4 (b1 + 1) (b2 + 1) > a^2 &&
a^2 b1 b2 < (1 - b1 b2) (2 b1 b2 - b1 - b2) && b1 b2 < 0 &&
b1 >= b2 && a > 0, {b2, b1, a}]
```

we obtain:

```
(-1 < b2 <= -(1/3) && ((0 < b1 <= b2/(-1 + 2 b2) &&
   0 < a < Sqrt[4 + 4 b1 + 4 b2 + 4 b1 b2]) || (b2/(-1 + 2 b2) <
    b1 < -(1/(3 b2)) &&
   Sqrt[(-b1 - b2 + 2 b1 b2 + b1^2 b2 + b1 b2^2 - 2 b1^2 b2^2)/(
```

```
    b1 b2)] < a < Sqrt[4 + 4 b1 + 4 b2 + 4 b1 b2])) || (-(1/3) <
b2 < 0 && ((0 < b1 <= b2/(-1 + 2 b2) &&
   0 < a < Sqrt[4 + 4 b1 + 4 b2 + 4 b1 b2]) || (b2/(-1 + 2 b2) <
    b1 < -(b2/(1 + 2 b2)) &&
   Sqrt[(-b1 - b2 + 2 b1 b2 + b1^2 b2 + b1 b2^2 - 2 b1^2 b2^2)/(
    b1 b2)] < a < Sqrt[4 + 4 b1 + 4 b2 + 4 b1 b2]))
```

Some further simplification yields (B.16) and (B.17).

□

Appendix C

Supplementary Material for Chapter 4

C.1 Local Boundedness and Lipschitzness

The purpose of this section is to derive the local Lipschitzness and boundedness of various first-order and second-order derivatives based on the assumption that f is twice continuous differentiable and that the Hessian of f is Lipschitz continuous (Assumption 4.3.1). Based on the derivations in Appendix C.1, we derive the non-asymptotic local convergence of Newton-type algorithms, including GD-Newton and Complete Newton. In order to quantify the absolute constants we mentioned in Theorems 4.3.2 and 4.3.5, we first quantify WLOG that the neighborhoods in (4.7) to be:

$$\mathcal{N}(x^*) = \mathcal{B}(x^*, \delta_x) := \{x \in \mathbb{R}^n : \|x - x^*\| \leq \delta_x\},$$
$$\mathcal{N}(y^*) = \mathcal{B}(y^*, \delta_y) := \{y \in \mathbb{R}^m : \|y - y^*\| \leq \delta_y\},$$

where $\delta_x > 0$ and $\delta_y > 0$. Since f is twice continuous differentiable, at its SLmM (x^*, y^*), the second-order derivatives are bounded. There exist positive constants B_{xx}, B_{xy}, B_{yy} such that for any $(x, y) \in \mathcal{B}(x^*, \delta_x) \times \mathcal{B}(y^*, \delta_y)$,

$$\|\partial^2_{xx} f(x, y)\| \leq B_{xx}, \|\partial^2_{xy} f(x, y)\| \leq B_{xy}, \|\partial^2_{yy} f(x, y)\| \leq B_{yy}. \tag{C.1}$$

Since $\partial^2_{yx} f(z) = (\partial^2_{xy} f(z))^\top$ for $f \in \mathcal{C}^2$ (Schwarz's theorem), we have $\|\partial^2_{yx} f(x, y)\| \leq B_{xy}$ (the fact that the matrix A and its transpose $A^\top$ have the same spectral norm can be derived from the SVD decomposition). For later convenience, we denote

$$\mathcal{B}(z^*) := \mathcal{B}(x^*, \delta_x) \times \mathcal{B}(y^*, \delta_y). \tag{C.2}$$

Since $\partial^2_{yy} f(x^*, y^*) \prec \mathbf{0}$ and $f \in \mathcal{C}^2$, we can assume w.l.o.g. that for any $z \in \mathcal{B}(z^*)$, $\partial^2_{yy} f(z) \preceq -\mu_y I$. Therefore, $(\partial^2_{yy} f(\cdot))^{-1}$ is bounded on $\mathcal{B}(z^*)$, i.e.,

$$\|(\partial^2_{yy} f(z))^{-1}\| \leq \mu_y^{-1}, \forall z \in \mathcal{B}(z^*). \tag{C.3}$$

This is because of the following lemma:

Lemma C.1.1 (**Local Lipschitzness and boundedness of the inverse**). *Suppose in a neighborhood $\mathcal{N} \subset \mathbb{R}^d$, there exists $\mu > 0$ such that there exists a matrix-valued function $A : \mathcal{N} \to \mathbb{R}^{k \times k}$ that satisfies:*

$$\text{for any } z \in \mathcal{N},\ A(z) \preceq -\mu I \text{ or for any } z \in \mathcal{N}, A(z) \succeq \mu I. \tag{C.4}$$

then for any $z \in \mathcal{N}$, $A(z)$ is invertible and $\|A^{-1}(z)\| \leq \mu^{-1}$, with $A^{-1} : z \mapsto (A(\cdot))^{-1}$. Moreover, if A is L-Lipschitz continuous, i.e., for any $z_1, z_2 \in \mathcal{N}$, we have

$$\|A(z_1) - A(z_2)\| \leq L\|z_1 - z_2\|, \tag{C.5}$$

then $A^{-1} := (A(\cdot))^{-1}$ is $\mu^{-2}L$-Lipschitz continuous, i.e., for any $z_1, z_2 \in \mathcal{N}$, we have

$$\|A^{-1}(z_1) - A^{-1}(z_2)\| \leq \mu^{-2} L\|z_1 - z_2\|. \tag{C.6}$$

Proof. WLOG we only need to prove the case when $A(z) \succeq \mu I$ for any $z \in \mathcal{N}$, because we can take $B = -A$ for the other case and apply the result on B. The invertibility of $A(z)$ follows from the positive definiteness. From the definition of spectral norm we have that for any $z \in \mathcal{N}$:

$$\|A^{-1}(z)\| = \sup_{\|w'\|=1} \|A^{-1}(z)w'\| = \sup_{\|A(z)w\|=1} \|w\| \tag{C.7}$$

On the other hand, $A(z) \succeq \mu I$ tells us that for any $w \in \mathbb{R}^d$ and $\|A(z)w\| = 1$, we can write:

$$\mu\|w\|^2 \leq w^\top A(z) w \leq \|w\| \cdot \|A(z)w\| = \|w\|, \tag{C.8}$$

where we used Cauchy–Schwarz inequality. Combining (C.7) and (C.8) above we obtain that for any $\|A(z)w\| = 1$, we have $\|w\| \leq \mu^{-1}$ and thus for any $z \in \mathcal{N}$:

$$\|A^{-1}(z)\| \leq \mu^{-1}. \tag{C.9}$$

Therefore, for $z_1, z_2 \in \mathcal{N}$, we have from the Lipschitzness of A that

$$\begin{aligned}\|A^{-1}(z_1) - A^{-1}(z_2)\| &= \|A^{-1}(z_1)A(z_1)A^{-1}(z_2) - A^{-1}(z_1)A(z_2)A^{-1}(z_2)\| \\ &\leq \|A^{-1}(z_1)(A(z_1) - A(z_2))A^{-1}(z_2)\| \\ &\leq \|A^{-1}(z_1)\| \cdot \|A(z_1) - A(z_2)\| \cdot \|A^{-1}(z_2)\| \\ &\leq \mu^{-2}L\|z_1 - z_2\|, \end{aligned} \tag{C.10}$$

where in the third line we used that for two matrices $U \in \mathbb{R}^d \to \mathbb{R}^{k\times k}$, $W \in \mathbb{R}^d \to \mathbb{R}^{k\times k}$,

$$\|UV\| = \sup_{\|z\|=1} \|UVz\| \leq \sup_{\|z\|=1} \|U\| \cdot \|Vz\| = \|U\| \sup_{\|z\|=1} \|Vz\| = \|U\| \cdot \|V\|. \tag{C.11}$$

□

Lemma C.1.1 tells that $(\partial_{yy}^2)^{-1} f := (\partial_{yy}^2 f(\cdot))^{-1}$ is $\mu_y^{-2} L_{yy}$-Lipschitz continuous under Assumption 4.3.1.

Now let us derive the local Lipschitzness of the partial derivatives $\partial_x f$ and $\partial_y f$ from the local boundedness of the partial Hessians. For any $z_1, z_2 \in \mathcal{B}(z^*)$, we have:

$$\begin{aligned}\|\partial_x f(z_1) - \partial_x f(z_2)\| &= \|\partial_x f(x_1, y_1) - \partial_x f(x_2, y_2)\| \\ &= \|\partial_x f(x_1, y_1) - \partial_x f(x_1, y_2) + \partial_x f(x_1, y_2) - \partial_x f(x_2, y_2)\| \\ &\leq \|\partial_x f(x_1, y_1) - \partial_x f(x_1, y_2)\| + \|\partial_x f(x_1, y_2) - \partial_x f(x_2, y_2)\| \\ &\leq \|\partial_{xy}^2 f(x_1, y_\xi)\| \cdot \|y_1 - y_2\| + \|\partial_{xx}^2 f(x_\gamma, y_2)\| \cdot \|x_1 - x_2\| \\ &\leq B_{xy}\|y_1 - y_2\| + B_{xx}\|x_1 - x_2\| \\ &\leq (B_{xy} + B_{xx})\|z_1 - z_2\|, \end{aligned} \tag{C.12}$$

where in the fourth line we used the mean-value theorem and that $y_\xi \in [y_1, y_2]$ and $x_\gamma \in [x_1, x_2]$ ($[a, b]$ denotes a line segment with end points a and b); in the second last line we used (C.1); in the last line we used $\|x_1 - x_2\| \leq \|z_1 - z_2\|$ and $\|y_1 - y_2\| \leq \|z_1 - z_2\|$. Similarly, we can derive that:

$$\|\partial_y f(z_1) - \partial_y f(z_2)\| \leq (B_{xy} + B_{yy})\|z_1 - z_2\|. \tag{C.13}$$

The local Lipschitzness of $\partial_x f$ and $\partial_y f$ also leads to their local boundedness. On the neighborhood $\mathcal{B}(z^*)$, we can derive:

$$\|\partial_x f(z)\| = \|\partial_x f(z) - \partial_x f(z^*)\| \leq L_x\|z - z^*\| \leq L_x(\|x - x^*\| + \|y - y^*\|) \leq L_x(\delta_x + \delta_y), \tag{C.14}$$

where we defined $L_x := B_{xy} + B_{xx}$ to be the Lipschitz constant of $\partial_x f$ on the neighborhood. Similarly, we can derive that $\|\partial_y f(z)\| \le L_y(\delta_x + \delta_y)$ for any $z \in \mathcal{B}(z^*)$ with $L_y := B_{xy} + B_{yy}$. To summarize we have the following lemma.

Lemma C.1.2 (**Local Lipschitzness and boundedness**). *At an SLmM z^* of a function $f \in \mathcal{C}^2$, there exist positive constants B_{xx}, B_{xy}, B_{yy} and μ_y such that for any $z \in \mathcal{B}(z^*)$:*

$$\begin{aligned} &\|\partial^2_{xx} f(z)\| \le B_{xx}, \|\partial^2_{xy} f(z)\| \le B_{xy}, \|\partial^2_{yy} f(z)\| \le B_{yy}, \\ &\partial^2_{yy} f(z) \preceq -\mu_y I, \ \|(\partial^2_{yy} f(z))^{-1}\| \le \mu_y^{-1}, \end{aligned} \tag{C.15}$$

and $\partial_x f$ and $\partial_y f$ are locally Lipschitz, i.e. for any $z_1, z_2 \in \mathcal{B}(z^*)$, we have

$$\begin{aligned} \|\partial_x f(z_1) - \partial_x f(z_2)\| &\le L_x \|z_1 - z_2\| := (B_{xx} + B_{xy})\|z_1 - z_2\|, \\ \|\partial_y f(z_1) - \partial_y f(z_2)\| &\le L_y \|z_1 - z_2\| := (B_{xy} + B_{yy})\|z_1 - z_2\|. \end{aligned} \tag{C.16}$$

Moreover, $\partial_x f(z)$ and $\partial_y f(z)$ are bounded, i.e., for any

$$\|\partial_x f(z)\| \le B_x := L_x(\delta_x + \delta_y), \ \|\partial_y f(z)\| \le B_y := L_y(\delta_x + \delta_y). \tag{C.17}$$

Suppose Assumption 4.3.1 holds on the neighborhood $\mathcal{B}(z^*)$, then $(\partial^2_{yy})^{-1} f := (\partial^2_{yy} f(\cdot))^{-1}$ is $\mu_y^{-2} L_{yy}$-Lipschitz continuous, i.e. for any $z_1, z_2 \in \mathcal{B}(z^*)$, we have

$$\|(\partial^2_{yy} f(z_1))^{-1} - (\partial^2_{yy} f(z_2))^{-1}\| \le \mu_y^{-2} L_{yy} \|z_1 - z_2\|. \tag{C.18}$$

Let us now derive the local Lipschitzness of $\mathtt{D}_x f$ and $\mathtt{D}_{xx} f$. We need the composition rules of the Lipschitzness and boundedness of addition and product. Recall from Assumption 4.3.1 that for any $z_1, z_2 \in \mathcal{B}(z^*)$, we have:

$$\begin{aligned} &\|\partial^2_{xx} f(z_1) - \partial^2_{xx} f(z_2)\| \le L_{xx}\|z_1 - z_2\|, \ \|\partial^2_{xy} f(z_1) - \partial^2_{xy} f(z_2)\| \le L_{xy}\|z_1 - z_2\|, \\ &\|\partial^2_{yy} f(z_1) - \partial^2_{yy} f(z_2)\| \le L_{yy}\|z_1 - z_2\|. \end{aligned} \tag{C.19}$$

Lemma C.1.3 (**Local Lipschitzness and boundedness of addition**). *Suppose that in a neighborhood $\mathcal{N} \subset \mathbb{R}^d$, we have matrix-valued functions $A : \mathcal{N} \to \mathbb{R}^{k \times k}$, $B : \mathcal{N} \to \mathbb{R}^{k \times k}$ and vector-valued functions $v : \mathcal{N} \to \mathbb{R}^k$, $u : \mathcal{N} \to \mathbb{R}^k$. Suppose that on the neighborhood $\mathcal{N}$, A is L_A-Lipschitz continuous, B is L_B-Lipschitz continuous, v is L_v-Lipschitz continuous and u is L_u-Lipschitz continuous. Namely, for any $z_1, z_2 \in \mathcal{N}$, we have:*

$$\begin{aligned} &\|A(z_1) - A(z_2)\| \le L_A\|z_1 - z_2\|, \ \|B(z_1) - B(z_2)\| \le L_B\|z_1 - z_2\|, \\ &\|v(z_1) - v(z_2)\| \le L_v\|z_1 - z_2\|, \|u(z_1) - u(z_2)\| \le L_u\|z_1 - z_2\|. \end{aligned} \tag{C.20}$$

Then, the matrix-matrix addition function $A+B : z \mapsto A(z)+B(z)$ is (L_A+L_B)-Lipschitz continuous and the vector-vector addition function $u+v : z \mapsto u(z)+v(z)$ is (L_v+L_u)-Lipschitz continuous, i.e. for any $z_1, z_2 \in \mathcal{N}$, we have:

$$\|(A+B)(z_1)-(A+B)(z_2)\| \le (L_A+L_B)\cdot\|z_1-z_2\|, \tag{C.21}$$
$$\|(u+v)(z_1)-(u+v)(z_2)\| \le (L_u+L_v)\cdot\|z_1-z_2\|. \tag{C.22}$$

Suppose A, B, v, u are B_A, B_B, B_v, B_u bounded on the neighborhood $\mathcal{N}$, respectively, i.e. *for any $z \in \mathcal{N}$, we have:*

$$\|A(z)\| \le B_A,\ \|B(z)\| \le B_B,\ \|v(z)\| \le B_v,\ \|u(z)\| \le B_u. \tag{C.23}$$

Then, $A+B$ is (B_A+B_B)-bounded and $u+v$ is (B_u+B_v)-bounded on the neighborhood $\mathcal{N}$.

Proof. For any $z_1, z_2 \in \mathcal{N}$, we write:

$$\begin{aligned}\|(A+B)(z_1)-(A+B)(z_2)\| &= \|A(z_1)-A(z_2)+B(z_1)-B(z_2)\| \\ &\le \|A(z_1)-A(z_2)\|+\|B(z_1)-B(z_2)\| \\ &\le L_A\|z_1-z_2\|+L_B\|z_1-z_2\| \\ &= (L_A+L_B)\|z_1-z_2\|. \end{aligned}\tag{C.24}$$

Similarly, we can prove $\|(u+v)(z_1)-(u+v)(z_2)\| \le (L_u+L_v)\cdot\|z_1-z_2\|$. The last sentence of Lemma C.1.3 follows from the triangle inequalities of norms. □

Lemma C.1.4 (Local Lipschitzness and boundedness of product). *Suppose that in a neighborhood $\mathcal{N} \subset \mathbb{R}^d$, we have matrix-valued functions $A : \mathcal{N} \to \mathbb{R}^{k\times k}$, $B : \mathcal{N} \to \mathbb{R}^{k\times k}$ and a vector-valued function $v : \mathcal{N} \to \mathbb{R}^k$. Suppose that on the neighborhood $\mathcal{N}$, A is L_A-Lipschitz continuous and B_A bounded, B is L_B-Lipschitz continuous and B_B bounded, v is L_v-Lipschitz continuous and B_v bounded. Namely, for any $z_1, z_2 \in \mathcal{N}$, we have:*

$$\begin{aligned}&\|A(z_1)-A(z_2)\| \le L_A\|z_1-z_2\|,\ \|B(z_1)-B(z_2)\| \le L_B\|z_1-z_2\|, \\ &\|v(z_1)-v(z_2)\| \le L_v\|z_1-z_2\|,\end{aligned}\tag{C.25}$$

and for any $z \in \mathcal{N}$,

$$\|A(z)\| \le B_A, \|B(z)\| \le B_B, \|v(z)\| \le B_v. \tag{C.26}$$

Then, the matrix-matrix product function $AB : z \mapsto A(z)B(z)$ *and the matrix-vector product function* $Av : z \mapsto A(z)v(z)$ *on the neighborhood* $\mathcal{N}$ *are also Lipschitz, i.e. for any* $z_1, z_2 \in \mathcal{N}$*, we have:*

$$\|A(z_1)B(z_1) - A(z_2)B(z_2)\| \leq (B_A L_B + B_B L_A) \cdot \|z_1 - z_2\|, \tag{C.27}$$
$$\|A(z_1)v(z_1) - A(z_2)v(z_2)\| \leq (B_A L_v + B_v L_A) \cdot \|z_1 - z_2\|. \tag{C.28}$$

Moreover, AB *is* $B_A B_B$*-bounded and* Av *is* $B_A B_v$*-bounded on the neighborhood* $\mathcal{N}$*.*

Proof. For any $z_1, z_2 \in \mathcal{N}$, we have:

$$\begin{aligned}
\|A(z_1)B(z_1) - A(z_2)B(z_2)\| &= \|A(z_1)B(z_1) - A(z_1)B(z_2) + A(z_1)B(z_2) - A(z_2)B(z_2)\| \\
&\leq \|A(z_1)B(z_1) - A(z_1)B(z_2)\| + \|A(z_1)B(z_2) - A(z_2)B(z_2)\| \\
&= \|A(z_1)(B(z_1) - B(z_2))\| + \|(A(z_1) - A(z_2))B(z_2)\| \\
&\leq \|A(z_1)\| \cdot \|B(z_1) - B(z_2)\| + \|A(z_1) - A(z_2)\| \cdot \|B(z_2)\| \\
&\leq (B_A L_B + B_B L_A)\|z_1 - z_2\|,
\end{aligned} \tag{C.29}$$

where in the fourth line we used (C.11). Similarly, we can derive that for $z_1, z_2 \in \mathcal{N}$, we have:

$$\|A(z_1)v(z_1) - A(z_2)v(z_2)\| \leq (B_A L_v + B_v L_A)\|z_1 - z_2\|. \tag{C.30}$$

The final claim follows from (C.11) and that for any $z \in \mathcal{N}$, $\|A(z)v(z)\| \leq \|A(z)\| \cdot \|v(z)\|$. □

We can now derive the local Lipschitzness of $\mathtt{D}_x f$ and $\mathtt{D}_{xx} f$ under Assumption 4.3.1. On the neighborhood $\mathcal{B}(z^*)$, since $(\partial_{yy}^2)^{-1} f$ is $\mu_y^{-2} L_{yy}$-Lipschitz continuous from Lemma C.1.1 and μ_y^{-1}-bounded from Lemma C.1.2, and $\partial_y f$ is L_y-Lipschitz continuous and B_y-bounded, from Lemma C.1.4,

$$(\partial_{yy}^2)^{-1} f \cdot \partial_y f \text{ is } (\mu_y^{-1} L_y + B_y \mu_y^{-2} L_{yy})\text{-Lipschitz continuous, and } \mu_y^{-1} B_y \text{ bounded.} \tag{C.31}$$

Since $\partial_{xy}^2 f$ is B_{xy}-bounded and L_{xy}-Lipschitz continuous from Assumption 4.3.1, $\partial_{xy}^2 f \cdot (\partial_{yy}^2)^{-1} f \cdot \partial_y f$ is

$$L_{xy}\mu_y^{-1} B_y + B_{xy}(\mu_y^{-1} L_y + B_y \mu_y^{-2} L_{yy}) \tag{C.32}$$

Lipschitz continuous and

$$B_{xy}\mu_y^{-1} B_y \tag{C.33}$$

bounded on $\mathcal{B}(z^*)$. Finally, from Lemma C.1.3, $\mathtt{D}_x f = \partial_x f - \partial^2_{xy} f \cdot (\partial^2_{yy})^{-1} f \cdot \partial_y f$ is:

$$L_x + L_{xy}\mu_y^{-1}B_y + B_{xy}(\mu_y^{-1}L_y + B_y\mu_y^{-2}L_{yy}) \tag{C.34}$$

Lipschitz continuous and $B_x + B_{xy}\mu_y^{-1}B_y$ bounded. In a similar way, $\mathtt{D}_{xx} f = \partial_{xx} f - \partial^2_{xy} f \cdot (\partial^2_{yy})^{-1} f \cdot \partial_{yx} f$ is

$$L_{xx} + 2L_{xy}B_{xy}\mu_y^{-1} + B_{xy}^2\mu_y^{-2}L_{yy} \tag{C.35}$$

Lipschitz continuous and $B_{xx} + B_{xy}^2\mu_y^{-1}$ bounded. We summarize our result as follows:

Lemma C.1.5 (Local Lipschitzness and boundedness of $\mathtt{D}_x f$ and $\mathtt{D}^2_{xx} f$). *Suppose on the neighborhood $\mathcal{B}(z^*)$ of an SLmM z^* of a function $f \in \mathcal{C}^2$, Assumption 4.3.1 holds. $\mathtt{D}_x f = \partial_x f - \partial^2_{xy} f \cdot (\partial^2_{yy})^{-1} f \cdot \partial_y f$ is:*

$$L_x^{\mathtt{D}} := L_x + L_{xy}\mu_y^{-1}B_y + B_{xy}(\mu_y^{-1}L_y + B_y\mu_y^{-2}L_{yy}) \tag{C.36}$$

Lipschitz continuous and

$$B_x^{\mathtt{D}} := B_x + B_{xy}\mu_y^{-1}B_y$$

bounded, i.e., for any $z, z_1, z_2 \in \mathcal{B}(z^*)$, we have that:

$$\|\mathtt{D}_x f(z_1) - \mathtt{D}_x f(z_2)\| \le L_x^{\mathtt{D}}\|z_1 - z_2\|,\ \|\mathtt{D}_x f(z)\| \le B_x^{\mathtt{D}}. \tag{C.37}$$

In a similar way, $\mathtt{D}_{xx} f = \partial_{xx} f - \partial^2_{xy} f \cdot (\partial^2_{yy})^{-1} f \cdot \partial_{yx} f$ is

$$L_{xx}^{\mathtt{D}} := L_{xx} + 2L_{xy}B_{xy}\mu_y^{-1} + B_{xy}^2\mu_y^{-2}L_{yy} \tag{C.38}$$

Lipschitz continuous and

$$B_{xx}^{\mathtt{D}} := B_{xx} + B_{xy}^2\mu_y^{-1}$$

bounded, where the constants are the same as in Lemma C.1.2. i.e., for any $z, z_1, z_2 \in \mathcal{B}(z^*)$, we have:

$$\|\mathtt{D}^2_{xx} f(z_1) - \mathtt{D}^2_{xx} f(z_2)\| \le L_{xx}^{\mathtt{D}}\|z_1 - z_2\|,\ \|\mathtt{D}^2_{xx} f(z)\| \le B_{xx}^{\mathtt{D}}. \tag{C.39}$$

Finally, we derive the local Lipschitzness of the derivatives of the local maximum function $\psi(x) = f(x, r(x))$ where $x \in \mathcal{N}(x^*)$, i.e.,

$$\psi'(x) = \mathtt{D}_x f(x, r(x)),\ \psi''(x) = \mathtt{D}^2_{xx} f(x, r(x)). \tag{C.40}$$

From Lemma C.1.5, for any $x_1, x_2 \in \mathcal{N}(x^*) = \mathcal{B}(x^*, \delta_x)$, we have that

$$\begin{aligned}
\|\psi'(x_1) - \psi'(x_2)\| &= \|\mathtt{D}_x f(x_1, r(x_1)) - \mathtt{D}_x f(x_2, r(x_2))\| \\
&\le L_x^{\mathtt{D}} \|(x_1, r(x_1)) - (x_2, r(x_2))\| \\
&\le L_x^{\mathtt{D}} (\|x_1 - x_2\| + \|r(x_1) - r(x_2)\|) \\
&= L_x^{\mathtt{D}} (\|x_1 - x_2\| + \||r'(x_\gamma)(x_1 - x_2)\|) \\
&\le L_x^{\mathtt{D}} (1 + \|r'(x_\gamma)\|) \|x_1 - x_2\| \\
&= L_x^{\mathtt{D}} (1 + \| - ((\partial_{yy}^2)^{-1} \cdot \partial_{yx}^2) f(x_\gamma, r(x_\gamma))\|) \|x_1 - x_2\| \\
&\le L_x^{\mathtt{D}} (1 + \mu_y^{-1} B_{xy}) \|x_1 - x_2\|, \qquad \text{(C.41)}
\end{aligned}$$

where in the fourth line we used the mean-value theorem and that x_γ is on the line segment with end points x_1 and x_2. In the sixth line we used (4.8) and in the last line we used the local boundedness of $(\partial_{yy}^2)^{-1} f$ and $\partial_{yx}^2 f$ in Lemma C.1.2 and (C.11). Similarly, we can derive that:

$$\begin{aligned}
\|\psi''(x_1) - \psi''(x_2)\| &= \|\mathtt{D}_{xx}^2 f(x_1, r(x_1)) - \mathtt{D}_{xx}^2 f(x_2, r(x_2))\| \\
&\le L_{xx}^{\mathtt{D}} \|(x_1, r(x_1)) - (x_2, r(x_2))\| \\
&\le L_{xx}^{\mathtt{D}} (1 + \mu_y^{-1} B_{xy}) \|x_1 - x_2\|. \qquad \text{(C.42)}
\end{aligned}$$

We summarize these conclusions:

Lemma C.1.6 (Local Lipschitzness of $\psi'(x)$ and $\psi''(x)$). *Under the same assumption as in Lemma C.1.5 we define*

$$\psi(x) := f(x, r(x)) \text{ where } x \in \mathcal{N}(x^*).$$

We have $\psi'(x) = \mathtt{D}_x f(x, r(x))$ and $\psi''(x) = \mathtt{D}_{xx}^2 f(x, r(x))$. Moreover, $\psi'(x)$ and $\psi''(x)$ are Lipschitz continuous on $\mathcal{N}(x^*)$, namely, for any $x \in \mathcal{N}(x^*)$, we have that:

$$\|\psi'(x_1) - \psi'(x_2)\| = \|\mathtt{D}_x f(x_1, r(x_1)) - \mathtt{D}_x f(x_2, r(x_2))\| \le L_x^{\psi} \|x_1 - x_2\|, \qquad \text{(C.43)}$$

$$\|\psi''(x_1) - \psi''(x_2)\| = \|\mathtt{D}_{xx}^2 f(x_1, r(x_1)) - \mathtt{D}_{xx}^2 f(x_2, r(x_2))\| \le L_{xx}^{\psi} \|x_1 - x_2\|, \qquad \text{(C.44)}$$

where we define

$$L_x^{\psi} := L_x^{\mathtt{D}} (1 + \mu_y^{-1} B_{xy}) \text{ and } L_{xx}^{\psi} := L_{xx}^{\mathtt{D}} (1 + \mu_y^{-1} B_{xy}), \qquad \text{(C.45)}$$

and the constants $L_x^{\mathtt{D}}, L_{xx}^{\mathtt{D}}, \mu_y, B_{xy}$ are defined in Lemmas C.1.2 and C.1.5.

Since $\mathtt{D}^2_{xx} f \succ \mathbf{0}$ for any $z \in \mathcal{B}(z^*)$ and we have proved in Lemma C.1.5 that $\mathtt{D}^2_{xx} f$ is (Lipschitz) continuous, there exists a positive constant $\mu_x > 0$ such that

$$\mathtt{D}^2_{xx} f(z) \succeq \mu_x I, \text{ for any } z \in \mathcal{B}(z^*). \tag{C.46}$$

From Lemma C.1.1 we obtain that:

Lemma C.1.7. *At an SLmM z^* of a function $f \in \mathcal{C}^2$, suppose that Assumption 4.3.1 holds on the neighborhood $\mathcal{B}(z^*)$. There exists $\mu_x > 0$ such that $(\mathtt{D}^2_{xx})^{-1} f := (\mathtt{D}^2_{xx} f(\cdot))^{-1}$ is locally bounded and Lipschitz continuous, i.e., for any $z \in \mathcal{B}(z^*)$, we have:*

$$\|(\mathtt{D}^2_{xx})^{-1} f(z)\| \leq \mu_x^{-1}, \tag{C.47}$$

and for any $z_1, z_2 \in \mathcal{B}(z^*)$, we have:

$$\|(\mathtt{D}^2_{xx})^{-1} f(z_1) - (\mathtt{D}^2_{xx})^{-1} f(z_2)\| \leq \mu_x^{-2} L^{\mathtt{D}}_{xx} \|z_1 - z_2\|, \tag{C.48}$$

where $L^{\mathtt{D}}_{xx}$ is defined in Lemma C.1.5.