Local Saddle Point Optimization: A Curvature Exploitation Approach

Leonard Adolphs   **Hadi Daneshmand**   **Aurelien Lucchi**   **Thomas Hofmann**
Department of Computer Science, ETH Zurich

Abstract

Gradient-based optimization methods are the most popular choice for finding local optima for classical minimization and saddle point problems. Here, we highlight a systemic issue of gradient dynamics that arise for saddle point problems, namely the presence of undesired stable stationary points that are no local optima. We propose a novel optimization approach that exploits curvature information in order to escape from these undesired stationary points. We prove that different optimization methods, including gradient method and ADAGRAD, equipped with curvature exploitation can escape non-optimal stationary points. We also provide empirical results on common saddle point problems which confirm the advantage of using curvature exploitation.

1 INTRODUCTION

We consider the problem of finding a *structured*[1] saddle point of a smooth objective, namely solving an optimization problem of the form

$$\min_{\mathbf{x} \in \mathbb{R}^k} \max_{\mathbf{y} \in \mathbb{R}^d} f(\mathbf{x}, \mathbf{y}). \tag{1}$$

Here, we assume that f is smooth in $\mathbf{x}$ and $\mathbf{y}$ but **not** necessarily convex in $\mathbf{x}$ or concave in $\mathbf{y}$. This particular problem arises in many applications, such as generative adversarial networks (GAN) [15], robust optimization [4], and game theory [37, 23]. Solving the saddle point problem in Eq. (1) is equivalent to finding a point $(\mathbf{x}^*, \mathbf{y}^*)$ such that

$$f(\mathbf{x}^*, \mathbf{y}) \leq f(\mathbf{x}^*, \mathbf{y}^*) \leq f(\mathbf{x}, \mathbf{y}^*) \tag{2}$$

---

[1]Throughout this work, we aim to find saddles that satisfy a particular (local) min-max structure in the input parameters.

holds for all $\mathbf{x} \in \mathbb{R}^k$ and $\mathbf{y} \in \mathbb{R}^d$. For a non convex-concave function f, finding such a saddle point is computationally infeasible. Instead of finding a *global* saddle point for Eq. (1), we aim for a more modest goal: finding a *locally* optimal saddle point, i.e. a point $(\mathbf{x}^*, \mathbf{y}^*)$ for which the condition in Eq. (2) holds true in a local neighbourhood around $(\mathbf{x}^*, \mathbf{y}^*)$.

There is a rich literature on saddle point optimization for the particular class of convex-concave functions, i.e. when f is convex in $\mathbf{x}$ and concave in $\mathbf{y}$. Although this type of objective function is commonly encountered in applications such as constrained convex minimization, many saddle point problems of interest do not satisfy the convex-concave assumption. Two popular examples that recently emerged in machine learning are distributionally robust optimization [12, 38], as well as training generative adversarial networks [15]. These applications can be framed as saddle point optimization problems which - due to the complex functional representation of the neural networks used as models - do not fulfill the convexity-concavity condition.

First-order methods are commonly used to solve problem (1) as they have a cheap per-iteration cost and are therefore easily scalable. One particular method of choice is simultaneous gradient descent/ascent, which performs the following iterative updates,

$$(\mathbf{x}_{t+1}, \mathbf{y}_{t+1}) = (\mathbf{x}_t, \mathbf{y}_t) + \eta_t \left(-\nabla_{\mathbf{x}} f_t, \nabla_{\mathbf{y}} f_t \right) \tag{3}$$
$$f_t := f(\mathbf{x}_t, \mathbf{y}_t),$$

where $\eta_t > 0$ is a chosen step size which can, e.g., decrease with time t or be a bounded constant (i.e. $\eta_t = \eta$). The convergence analysis of the above iterate sequence is typically tied to a strong/strict convexity-concavity property of the objective function defining the dynamics. Under such conditions, the gradient method is guaranteed to converge to a desired saddle point [3]. These conditions can also be relaxed to some extent, which will be further discussed in Section 2.

It is known that the gradient method is locally asymptotically stable [26]; but stability alone is not sufficient to guarantee convergence to a locally *optimal* saddle point. Through an example, we will later illustrate that the gradient method is indeed stable at some undesired

stationary points, at which the *structural* min-max property [2] is not met. This is in clear contrast to minimization problems where all stable stationary points of the gradient dynamics are local minima. The stability of these undesired stationary points is therefore an additional difficulty that one has to consider for escaping from such saddles. While a standard trick for escaping saddles in minimization problems consists of adding a small perturbation to the gradient, we will demonstrate that this does not guarantee avoiding undesired stationary points.

Throughout the paper, we will refer to a *desired* local saddle point as a local minimum in $\mathbf{x}$ and maximum in $\mathbf{y}$. This characterization implies that the Hessian matrix at $(\mathbf{x}, \mathbf{y})$ does not have a negative curvature direction in $\mathbf{x}$ (which corresponds to an eigenvector of $\nabla_{\mathbf{x}}^2 f$ with a negative associated eigenvalue) and a positive curvature direction in $\mathbf{y}$ (which corresponds to an eigenvector of $\nabla_{\mathbf{y}}^2 f$ with a positive associated eigenvalue). In that regard, curvature information can be used to certify whether the desired min-max structure is met.

In this work, we propose the first saddle point optimization that exploits curvature to guide the gradient trajectory towards the desired saddle points that respect the min-max structure. Since our approach only makes use of the eigenvectors corresponding to the maximum and minimum eigenvalue (rather than the whole eigenspace), we will refer to it as *extreme* curvature exploitation. We will prove that this type of curvature exploitation avoids convergence to undesired saddles–albeit not guarantees convergence on a general non-convex-concave saddle point problem. Our contribution is linked to the recent research area of stability analysis for gradient-based optimization in general saddle point problems. Nagarajan et al. [27] have shown that the gradient method is stable at locally optimal saddles. Here, we complete the picture by showing that this method is unfavourably stable at some points that are not locally optimal. Our empirical results also confirm the advantage of curvature exploitation in saddle point optimization.

2 RELATED WORK

Asymptotical Convergence In the context of optimizing a Lagrangian, the pioneering works of [20, 3] popularized the use of the primal-dual dynamics to arrive at the saddle points of the objective. The work of [3] analyzed the stability of this method in continuous time proving global stability results under strict convex-concave assumptions. This result was extended

in [39] for a discrete-time version of the subgradient method with a constant step size rule, proving that the iterates converge to a neighborhood of a saddle point. Results for a decreasing step size were provided in [14, 25] while [32] analyzed an adaptive step size rule with averaged parameters. The work of [7] has shown that the conditions of the objective can be relaxed, proving asymptotic stability to the set of saddle points is guaranteed if either the convexity or concavity properties are strict, and convergence is pointwise. They also proved that the strictness assumption can be dropped under other linearity assumptions or assuming strongly joint quasiconvex-quasiconcave saddle functions.

However, for problems where the function considered is not strictly convex-concave, convergence to a saddle point is not guaranteed, with the gradient dynamics leading instead to oscillatory solutions [16]. These oscillations can be addressed by averaging the iterates [32] or using the extragradient method (a perturbed version of the gradient method) [19, 13].

There are also instances of saddle point problems that do not satisfy the various conditions required for convergence. A notable example are generative adversarial networks (GANs) for which the work of [27] proved local asymptotic stability under certain suitable conditions on the representational power of the two players (called discriminator and generator). Despite these recent advances, the convergence properties of GANs are still not well understood.

Non-asymptotical Convergence An explicit convergence rate for the subgradient method with a constant stepsize was proved in [30] for reaching an approximate saddle point, as opposed to asymptotically exact solutions. Assuming the function is convex-concave, they proved a sub-linear rate of convergence. Rates of convergence have also been derived for the extragradient method [19] as well as for mirror descent [31].

In the context of GANs, [34] showed that a single-step gradient method converges to a saddle point in a neighborhood around the saddle point in which the function is strongly convex-concave. The work of [24] studied the theory of non-asymptotic convergence to a local Nash equilibrium. They prove that–assuming local strong convexity-concavity–simultaneous gradient descent achieves an exponential rate of convergence near a stable local Nash equilibrium. They also extended this result to other discrete-time saddle point dynamics such as optimistic mirror descent or predictive methods.

Negative Curvature Exploitation The presence of negative curvature in the objective function indicates the existence of a potential descent direction, which is

---

[2]This property refers to the function being a local minimum in $\mathbf{x}$ and a maximum in $\mathbf{y}$.

commonly exploited in order to escape saddle points and reach a local minimizer. Among these approaches are trust-region methods that guarantee convergence to a second-order stationary point [8, 33, 6]. While a naïve implementation of these methods would require the computation and inversion of the Hessian of the objective, this can be avoided by replacing the computation of the Hessian by Hessian-vector products that can be computed efficiently in $\mathcal{O}(nd)$ [35]. This is applied e.g. using matrix-free Lanczos iterations [9] or online variants such as Oja's algorithm [1]. Sub-sampling the Hessian can furthermore reduce the dependence on n by using various sampling schemes [18, 40]. Finally, [2, 41] showed that first-order information can act as a noisy Power method allowing to find a negative curvature direction.

In contrast to these classical results that "blindly" try to escape any type of saddle-point, our aim is to exploit curvature information to reach a specific type of stationary point that satisfies the min-max condition required at the optimum of the objective function.

3 PRELIMINARIES

Definition: Locally Optimal Saddles Let us define a γ-neighbourhood around the point $(\mathbf{x}^*, \mathbf{y}^*)$ as

$$\mathcal{K}_\gamma^* = \{(\mathbf{x}, \mathbf{y}) \big| \ \|\mathbf{x} - \mathbf{x}^*\| \leq \gamma, \|\mathbf{y} - \mathbf{y}^*\| \leq \gamma\} \quad (4)$$

with a sufficiently small $\gamma > 0$. Throughout the paper, we follow a common approach, see e.g. [26, 28], for this type of problem and relax the condition of Eq. (2) to hold only in a local neighbourhood.

Definition 1. *The point $(\mathbf{x}^*, \mathbf{y}^*)$ is a locally optimal saddle point of the problem in Eq. (1) if*

$$f(\mathbf{x}^*, \mathbf{y}) \leq f(\mathbf{x}^*, \mathbf{y}^*) \leq f(\mathbf{x}, \mathbf{y}^*) \quad (5)$$

holds for $\forall (\mathbf{x}, \mathbf{y}) \in \mathcal{K}_\gamma^*$.

Assumptions For the sake of further analysis, we require the function f to be sufficiently smooth, and its second order derivatives with respect to the parameters $\mathbf{x}$ and $\mathbf{y}$ to be non-degenerate at the optimum $(\mathbf{x}^*, \mathbf{y}^*)$.

Assumption 2 (Smoothness). *We assume that $f(\mathbf{z}) := f(\mathbf{x}, \mathbf{y})$ is a C^2 function, and that its gradient and Hessian are Lipschitz with respect to the parameters $\mathbf{x}$ and $\mathbf{y}$, i.e. we assume that the following*

inequalities hold:

$$\|\nabla f(\mathbf{z}) - \nabla f(\widetilde{\mathbf{z}})\| \leq L_{\mathbf{z}} \|\mathbf{z} - \widetilde{\mathbf{z}}\| \quad (6)$$

$$\|\nabla^2 f(\mathbf{z}) - \nabla^2 f(\widetilde{\mathbf{z}})\| \leq \rho_{\mathbf{z}} \|\mathbf{z} - \widetilde{\mathbf{z}}\| \quad (7)$$

$$\|\nabla_{\mathbf{x}} f(\mathbf{z}) - \nabla_{\mathbf{x}} f(\widetilde{\mathbf{z}})\| \leq L_{\mathbf{x}} \|\mathbf{z} - \widetilde{\mathbf{z}}\| \quad (8)$$

$$\|\nabla_{\mathbf{x}}^2 f(\mathbf{z}) - \nabla_{\mathbf{x}}^2 f(\widetilde{\mathbf{z}})\| \leq \rho_{\mathbf{x}} \|\mathbf{z} - \widetilde{\mathbf{z}}\| \quad (9)$$

$$\|\nabla_{\mathbf{y}} f(\mathbf{z}) - \nabla_{\mathbf{y}} f(\widetilde{\mathbf{z}})\| \leq L_{\mathbf{y}} \|\mathbf{z} - \widetilde{\mathbf{z}}\| \quad (10)$$

$$\|\nabla_{\mathbf{y}}^2 f(\mathbf{z}) - \nabla_{\mathbf{y}}^2 f(\widetilde{\mathbf{z}})\| \leq \rho_{\mathbf{y}} \|\mathbf{z} - \widetilde{\mathbf{z}}\| \quad (11)$$

Moreover, we assume bounded gradients, i.e.

$$\|\nabla_{\mathbf{x}} f(\mathbf{z})\| \leq \ell_{\mathbf{x}}, \ \|\nabla_{\mathbf{y}} f(\mathbf{z})\| \leq \ell_{\mathbf{y}}, \ \|\nabla_{\mathbf{z}} f(\mathbf{z})\| \leq \ell_{\mathbf{z}} \quad (12)$$

Assumption 3 (Non-degenerate Hessian at $(\mathbf{x}^*, \mathbf{y}^*)$). *We assume that the matrices $\nabla_{\mathbf{x}}^2 f(\mathbf{x}^*, \mathbf{y}^*)$ and $\nabla_{\mathbf{y}}^2 f(\mathbf{x}^*, \mathbf{y}^*)$ are non-degenerate for all locally optimal points $(\mathbf{x}^*, \mathbf{y}^*) \in \mathbb{R}^{k+d}$ as defined in Def. 1.*

With the use of Assumption 3, we are able to establish sufficient conditions on $(\mathbf{x}^*, \mathbf{y}^*)$ to be a locally optimal saddle point.

Lemma 4. *Suppose that f satisfies assumption 3; then, $\mathbf{z}^* := (\mathbf{x}^*, \mathbf{y}^*)$ is a locally optimal saddle point on $\mathcal{K}_\gamma^*$ if and only if the gradient with respect to $\mathbf{z}^*$ is zero, i.e.*

$$\nabla f(\mathbf{x}^*, \mathbf{y}^*) = 0, \quad (13)$$

and the second derivative at $(\mathbf{x}^*, \mathbf{y}^*)$ is positive definite in $\mathbf{x}$ and negative definite in $\mathbf{y}$ [3], i.e. there exist $\mu_{\mathbf{x}}, \mu_{\mathbf{y}} > 0$ such that

$$\nabla_{\mathbf{x}}^2 f(\mathbf{x}^*, \mathbf{y}^*) \succ \mu_{\mathbf{x}} \mathbf{I}, \qquad \nabla_{\mathbf{y}}^2 f(\mathbf{x}^*, \mathbf{y}^*) \prec -\mu_{\mathbf{y}} \mathbf{I}. \quad (14)$$

4 UNDESIRED STABILITY

Asymptotic Scenarios There are three different asymptotic scenarios for the gradient iterations in Eq. (3): (i) divergence (i.e. $\lim_{t \to \infty} \|(\mathbf{x}_t, \mathbf{y}_t)\| \to \infty$), (ii) being trapped in a loop (corresponding to $\lim_{t \to \infty} \|\nabla f\| > 0$), and (iii) convergence to a stationary point of the gradient updates (i.e. $\lim_{t \to \infty} \|\nabla f\| = 0$). To the best of our knowledge, there is no convergence guarantee for general saddle point optimization. Typical convergence guarantees require convexity-concavity or somewhat relaxed conditions such as quasiconvexity-quasiconcavity of f [7]. This paper focuses on the third outline case and investigates the theoretical guarantees for a convergent series. We will show that gradient-based optimization can converge to some *undesired* stationary points and propose an optimizer that uses extreme curvature information to alleviate this problem. We specifically highlight that

---

[3] In the game theory literature, such point is commonly referred to as local Nash equilibrium, see e.g. [24].

we do not provide any convergence guarantee. Rather, we investigate if a convergent sequence is guaranteed to yield a valid solution to the local saddle point problem, i.e., if it always converges to a locally optimal saddle point as defined in Def. 1.

Local Stability A stationary point of the gradient iterations can be either stable or unstable. The notion of stability characterizes the behavior of the gradient iterations in a local region around the stationary point. In the neighborhood of a stable stationary point, successive iterations of the method are not able to escape the region. Conversely, we consider a stationary point to be unstable if it is not stable [17]. The stationary point $\mathbf{z}^* = (\mathbf{x}^*, \mathbf{y}^*)$ (for which $\nabla f(\mathbf{z}^*) = 0$ holds) is a locally stable point of the gradient iterations in Eq. 3, if the Jacobian of its dynamics has only eigenvalues λ_i within the unit disk, i.e.

$$\left| \lambda_i \left(\mathbf{I} + \eta \begin{bmatrix} -\nabla_{\mathbf{x}}^2 f(\mathbf{z}^*) & -\nabla_{\mathbf{x},\mathbf{y}}^2 f(\mathbf{z}^*) \\ \nabla_{\mathbf{y},\mathbf{x}}^2 f(\mathbf{z}^*) & \nabla_{\mathbf{y}}^2 f(\mathbf{z}^*) \end{bmatrix} \right) \right| \le 1. \quad (15)$$

Definition 5 (Stable Stationary Point of Gradient Dynamics)**.** *A point $\mathbf{z} = (\mathbf{x}, \mathbf{y})$ is a stable stationary point of the gradient dynamics in Eq. (3) (for an arbitrarily small step size $\eta > 0$) if $\nabla_{\mathbf{z}} f = 0$ and if the eigenvalues of the matrix*

$$\begin{bmatrix} -\nabla_{\mathbf{x}}^2 f(\mathbf{z}) & -\nabla_{\mathbf{x},\mathbf{y}}^2 f(\mathbf{z}) \\ \nabla_{\mathbf{y},\mathbf{x}}^2 f(\mathbf{z}) & \nabla_{\mathbf{y}}^2 f(\mathbf{z}) \end{bmatrix} \quad (16)$$

only have eigenvalues with negative real-part.

Random Initialization In the following, we will use the notion of stability to analyze the asymptotic behavior of the gradient method. We start with a lemma extending known results for general minimization problems that prove that gradient descent with random initialization almost surely converges to a stable stationary point [22].

Lemma 6 (Random Initialization)**.** *Suppose that assumptions 2 and 3 hold. Consider the gradient iterate sequence of Eq. (3) with step size $\eta < \min\left(\frac{1}{L_x}, \frac{1}{L_y}, \frac{1}{\sqrt{2}L_z}\right)$ starting from a random initial point. If the sequence converges to a stationary point, then the stationary point is almost surely stable.*

Undesired Stable Stationary Point If all stable stationary points of the gradient dynamics would be locally optimal saddle points, then the result of Lemma 6 guarantees almost sure convergence to a solution of the saddle point problem in Eq. (1). Previous work by [26, 27] has shown that every locally optimal saddle point is a stable stationary point of the gradient dynamics. While for minimization problems, the set of stable stationary points is the same as the set of local

minima, this might not be the case for the problem we consider here. Indeed, the gradient dynamics might introduce additional stable points that are not locally optimal saddle points. We illustrate this claim in the next example.

Example Consider the following two-dimensional saddle point problem[4]

$$\min_{x \in \mathbb{R}} \max_{y \in \mathbb{R}} \left[f(x,y) = 2x^2 + y^2 + 4xy + \frac{4}{3}y^3 - \frac{1}{4}y^4 \right] \quad (17)$$

with $x, y \in \mathbb{R}$. The critical points of the function, i.e. points for which $\nabla f(x,y) = 0$, are

$$\mathbf{z}_0 = (0,0) \qquad \mathbf{z}_1 = (-2 - \sqrt{2}, 2 + \sqrt{2})$$
$$\mathbf{z}_2 = (-2 + \sqrt{2}, 2 - \sqrt{2}) \quad (18)$$

Evaluating the Hessians at the three critical points gives rise to the following three matrices:

$$\mathbf{H}(\mathbf{z}_0) = \begin{bmatrix} 4 & 4 \\ 4 & 2 \end{bmatrix} \qquad \mathbf{H}(\mathbf{z}_1) = \begin{bmatrix} 4 & 4 \\ 4 & -4\sqrt{2} \end{bmatrix}$$
$$\mathbf{H}(\mathbf{z}_2) = \begin{bmatrix} 4 & 4 \\ 4 & 4\sqrt{2} \end{bmatrix}. \quad (19)$$

We see that only $\mathbf{z}_1$ is a locally optimal saddle point, namely that $\nabla_{\mathbf{x}}^2 f(\mathbf{z}_1) = 4 > 0$ and $\nabla_{\mathbf{y}}^2 f(\mathbf{z}_1) = -4\sqrt{2} < 0$, whereas the two other points are both a local minimum in the parameter $\mathbf{y}$, rather than a maximum. However, figure 1a illustrates gradient steps converging to the undesired stationary point $\mathbf{z}_0$ because it is a locally stable point of the dynamics[5]. Hence, even small perturbations of the gradients in each step can not avoid convergence to this point (see Figure 1b).

5 EXTREME CURVATURE EXPLOITATION

The previous example has shown that gradient iterations on the saddle point problem introduce undesired stable points. In this section, we propose a strategy to escape from these points. Our approach is based on exploiting curvature information as in [9].

Extreme Curvature Direction Let $\lambda_{\mathbf{x}}$ be the minimum eigenvalue of $\nabla_{\mathbf{x}}^2 f(\mathbf{z})$ with its associated eigenvector $\mathbf{v}_{\mathbf{x}}$, and $\lambda_{\mathbf{y}}$ be the maximum eigenvalue of $\nabla_{\mathbf{y}}^2 f(\mathbf{z})$

---

[4]To guarantee smoothness, one can restrict the domain of f to a bounded set.

[5]This can be easily shown by observing that the real-part of the eigenvalues of the matrix in Eq. 16, evaluated at $\mathbf{z}_0$, are all negative.
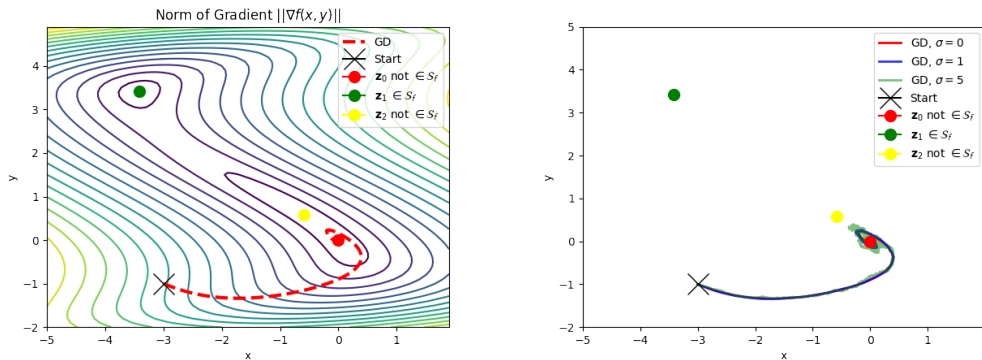
(a) The contour lines correspond to the norm of the gradient $\|\nabla f(x, y)\|$.

(b) Optimization trajectory when adding Gaussian noise from $\mathcal{N}(0, \sigma)$ to the gradient in every step.

Figure 1: Optimization trajectory of the gradient method on the function $f(x, y)$ in Eq. (17) with a step size of $\eta = 0.001$. The method converges to the critical point $(0,0)$, even though it is not a locally optimal saddle point, and therefore not a solution to the problem defined in (17). We denote the set of locally optimal saddle points of the function f with $\mathcal{S}_f$.

with its associated eigenvector $\mathbf{v_y}$. Then, we define

$$\mathbf{v_z}^{(-)} = \mathbb{1}_{\{\lambda_\mathbf{x} < 0\}} \frac{\lambda_\mathbf{x}}{2\rho_\mathbf{x}} \mathrm{sgn}(\mathbf{v_x}^\top \nabla_\mathbf{x} f(\mathbf{z})) \mathbf{v_x} \qquad (20)$$

$$\mathbf{v_z}^{(+)} = \mathbb{1}_{\{\lambda_\mathbf{y} > 0\}} \frac{\lambda_\mathbf{y}}{2\rho_\mathbf{y}} \mathrm{sgn}(\mathbf{v_y}^\top \nabla_\mathbf{y} f(\mathbf{z})) \mathbf{v_y} \qquad (21)$$

where $\mathrm{sgn} : \mathbb{R} \to \{-1, 1\}$ is the sign function. Using the above vectors, we define $\mathbf{v_z} := (\mathbf{v_z}^{(-)}, \mathbf{v_z}^{(+)})$ as the *extreme* curvature direction at $\mathbf{z}$.

Algorithm Using the extreme curvature direction, we modify the gradient steps as follows:

$$(\mathbf{x}_{t+1}, \mathbf{y}_{t+1}) = (\mathbf{x}_t, \mathbf{y}_t) + \mathbf{v}_{\mathbf{z}_t} + \eta(-\nabla_\mathbf{x} f_t, \nabla_\mathbf{y} f_t) \quad (22)$$
$$f_t := f(\mathbf{x}_t, \mathbf{y}_t).$$

This new update step is constructed by adding the extreme curvature direction to the gradient method of Eq. (3). From now on, we will refer to this modified update as the CESP (curvature exploitation for the saddle point problem) method. Note that the algorithm reduces to gradient-based optimization in regions where there are only positive eigenvalues in $\mathbf{x}$ and negative eigenvalues in $\mathbf{y}$ as the extreme curvature vector is zero. This includes the region around any locally optimal saddle point.

Stability Extreme curvature exploitation has already been used for escaping from unstable stationary points (i.e. saddle points) of gradient descent for minimization problems [9]. In saddle point problems, curvature exploitation is advantageous not only for escaping *unstable* stationary points but also for escaping undesired *stable* stationary points of the gradient iterates. The

upcoming two lemmas prove that the set of stable stationary points of the CESP dynamics and the set of locally optimal saddle points are the same – therefore, the optimizer only converges to a solution of the local saddle point problem. The issue of gradient based optimization as well as the theoretical guarantees of the CESP method are visualized in the Venn diagram in figure 2: while for minimization problems the set of stable points of gradient descent equals the set of local minima, we see that for saddle point problems gradient-based optimization introduces additional stable points outside of the set of locally optimal solutions. However, by exploiting extreme curvatures with our proposed CESP method, all points outside of the set of locally optimal saddles become non-stationary. Hence, every convergent sequence of the CESP method yields a solution to the local saddle point problem.

Lemma 7. *A point $\mathbf{z} := (\mathbf{x}, \mathbf{y})$ is a stationary point of the iterates in Eq. (22) if and only if $\mathbf{z}$ is a locally optimal saddle point.*

We can conclude from the result of Lemma 7 that every stationary point of the CESP dynamics is a locally optimal saddle point. The next Lemma establishes the stability of these points.

Lemma 8. *Suppose that assumptions 2 and 3 hold. Let $\mathbf{z}^* := (\mathbf{x}^*, \mathbf{y}^*)$ be a locally optimal saddle point, i.e.*

$$\nabla f(\mathbf{z}) = 0, \ \nabla_\mathbf{x}^2 f(\mathbf{z}^*) \succeq \mu_\mathbf{x} \mathbf{I}, \ \nabla_\mathbf{y}^2 f(\mathbf{z}^*) \preceq -\mu_\mathbf{y} \mathbf{I}, \quad (23)$$
$$(\mu_\mathbf{x}, \mu_\mathbf{y} > 0)$$

Then the iterates of Eq.(22) are stable in $\mathcal{K}_\gamma^*$ for

$$\gamma \leq \min\{\mu_\mathbf{x}/(\sqrt{2}\rho_\mathbf{x}), \mu_\mathbf{y}/(\sqrt{2}\rho_\mathbf{y})\}. \qquad (24)$$
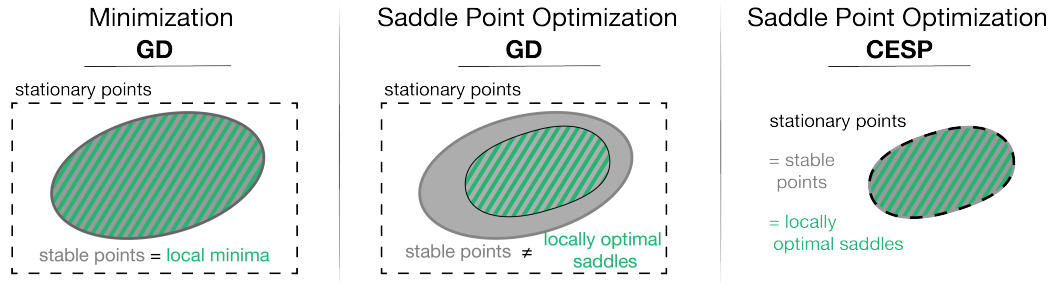
Figure 2: This Venn diagram summarizes the issue of gradient-based optimization that introduces *undesired* stable points. This problem is unique to saddle point optimization and solved by the proposed CESP method. [Best viewed in color.]

Escaping From Undesired Saddles Extreme curvature exploitation allows us to escape from undesired saddles. In the next lemma, we show that the optimization trajectory of CESP stays away from all undesired stationary points of the gradient dynamics.

Lemma 9. *Suppose that* $\mathbf{z}^* := (\mathbf{x}^*, \mathbf{y}^*)$ *is an undesired stationary point of the gradient dynamics, namely*

$$\nabla f(\mathbf{z}^*) = 0, \|\mathbf{v}_{\mathbf{z}^*}\| > 0. \tag{25}$$

Consider the iterates of Eq. (22) starting from $\mathbf{z}_0 = (\mathbf{x}_0, \mathbf{y}_0)$ *in a* γ*-neighbourhood of* $\mathbf{z}^*$. *After one step the iterates escape the* γ*-neighbourhood of* $\mathbf{z}^*$, *i.e.*

$$\|\mathbf{z}_1 - \mathbf{z}^*\| \geq \gamma \tag{26}$$

for a sufficiently small $\gamma = \mathcal{O}(\|\mathbf{v}_{\mathbf{z}^*}\|)$.

Implementation with Hessian-vector products Since storing and computing the Hessian in high dimensions is very costly, we need to find a way to efficiently extract the extreme curvature direction. The most common approach for obtaining the eigenvector corresponding to the largest absolute eigenvalue, (and the eigenvalue itself) of $\nabla_{\mathbf{x}}^2 f(\mathbf{z})$ is to run power iterations on $\mathbf{I} - \beta \nabla_{\mathbf{x}}^2 f(\mathbf{z})$ as

$$\mathbf{v}_{t+1} = (\mathbf{I} - \beta \nabla_{\mathbf{x}}^2 f(\mathbf{z}))\mathbf{v}_t \tag{27}$$

where $\mathbf{v}_0$ is a random vector and $\mathbf{v}_{t+1}$ is normalized after every iteration. The parameter $\beta > 0$ is chosen such that $\mathbf{I} - \beta \nabla_{\mathbf{x}}^2 f(\mathbf{z}) \succeq 0$. Since this method only requires implicit Hessian computation through a Hessian-vector product, it can be implemented as efficiently as gradient evaluations [35]. The results of [21] provide a bound on the number of required iterations to extract the extreme curvature: for the case $\lambda_{\min}(\nabla_{\mathbf{x}}^2 f(\mathbf{z})) \leq -\gamma$, $\frac{1}{\gamma} \log(k/\delta^2) L_x$ iterations suffice to find a vector $\hat{\mathbf{v}}$ such that $\hat{\mathbf{v}}^\top \nabla_{\mathbf{x}}^2 f(\mathbf{z})\hat{\mathbf{v}} \leq -\frac{\gamma}{2}$ with probability $1 - \delta$ (cf. [22]).

Comparison to second-order optimization We would like to draw the attention of the reader to the

fact that the CESP method only uses *extreme* curvature which makes it conceptually different from second-order Newton-type optimization. Although there is a rich literature on second-order optimization for variational inequalities and convex-concave saddle point problems, to the best of our knowledge, there is neither theoretical nor practical evidence for success of these methods on general smooth saddle point problems.

6 CURVATURE EXPLOITATION FOR LINEAR-TRANSFORMED GRADIENT STEPS

Linear-Transformed Gradient Optimization Applying a linear transformation to the gradient updates is commonly used to accelerate optimization for various types of problems. The resulting updates can be written in the general form

$$\mathbf{z}_{t+1} = \mathbf{z}_t + \eta \mathbf{A}_{\mathbf{z}_t}(-\nabla_{\mathbf{x}} f_t, \nabla_{\mathbf{y}} f_t), \ f_t = f(\mathbf{z}_t) \tag{28}$$

where $\mathbf{A}_{\mathbf{z}} = \begin{bmatrix} \mathcal{A} & 0 \\ 0 & \mathcal{B} \end{bmatrix}$ is a symmetric, block-diagonal $((k+d) \times (k+d))$-matrix. Different optimization methods use a different linear transformation $\mathbf{A}_{\mathbf{z}}$. Table 1 in section B in the appendix illustrates the choice of $\mathbf{A}_{\mathbf{z}}$ for different optimizers. ADAGRAD [11], one of the most popular optimization methods in machine learning, belongs to this category.

Extreme Curvature Exploitation We can adapt CESP to the linear-transformed variant:

$$\mathbf{z}_{t+1} = \mathbf{z}_t + \mathbf{v}_{\mathbf{z}_t} + \eta \mathbf{A}_{\mathbf{z}_t}(-\nabla_{\mathbf{x}} f_t, \nabla_{\mathbf{y}} f_t). \tag{29}$$

where we choose the linear transformation matrix $\mathbf{A}_{\mathbf{z}_t}$ to be positive definite. This variant of CESP is also able to filter out the undesired stable stationary points of the gradient method for the saddle point problem. The following lemma proves that it has the same properties as the non-transformed version.

Lemma 10. *The set of locally optimal saddle points as defined in Def. 1 and the set of stable points of the*

linear-transformed CESP *update method in Eq. (29) are the same.*

A direct implication of Lemma 10 is that we can also use curvature exploitation for ADAGRAD. Later, we will experimentally show the advantage of using curvature exploitation for this method.

7   EXPERIMENTS

7.1   Escaping From Undesired Stationary Points of the Toy Example

Previously, we saw that for the two dimensional saddle point problem on the function of Eq. (17), gradient iterates may converge to an undesired stationary point that is not locally optimal. As shown in Figure 3, CESP solves this issue. In this example, simultaneous gradient iterates converge to the undesired stationary point $\mathbf{z}_0 = (0,0)$ for many different initialization parameters, whereas our method always converges to the locally optimal saddle point. A plot of the basin of attraction of the two different optimizers on this example is presented in Figure 6 in the appendix.

7.2   Robust Optimization

Although robust optimization [4] is often formulated as a convex-concave saddle point problem, we consider robust optimization on neural networks that do not fulfill this assumption. The optimization problem that we target here is an application of robust optimization in empirical risk minimization [29], namely solving

$$\min_{\mathbf{x}} \sup_{P \in \mathcal{P}} \big[f(\mathbf{X}; \mathbf{x}, \mathcal{P})$$
$$= \Big\{ \mathbb{E}_P[l(\mathbf{X}; \theta)] : D(P\|\hat{P}_n) \leq \frac{\rho}{n} \Big\} \big] \quad (30)$$

where $l(X; \mathbf{x})$ denotes the cost function to minimize, $\mathbf{X}$ the data, and $D(P\|\hat{P}_n)$ a divergence measure between the true data distribution P and the empirical data distribution $\hat{P}_n$.

We use this framework on the Wisconsin breast cancer data set, which is a binary classification task with 30 attributes and 569 samples, and choose a multilayer perceptron with a non-convex sigmoid activation as the classifier. Due to the relatively small sample size, we can compute the gradient exactly in this case. We choose the objective $f(X; \mathbf{x}, \mathcal{P})$ in this setting to be

$$f(X; \mathbf{x}, \mathcal{P})$$
$$= - \sum_{i=1}^{n} p_i^* \left[y_i \log(\hat{y}(\mathbf{X}_i)) + (1 - y_i) \log(1 - \hat{y}(\mathbf{X}_i)) \right]$$
$$- \lambda \sum_{i=1}^{n} (p_i^* - \frac{1}{n})^2 \quad (31)$$

where we add a regularization term with $\lambda > 0$ to enforce the divergence constraint. Figure 4 shows the comparison of the gradient method (GD) and our CESP optimizer on this problem in terms of the minimum eigenvalue of $\nabla_{\mathbf{x}}^2 f(\mathbf{X}; \mathbf{x}, \mathcal{P})$. Note that f is concave with respect to $\mathcal{P}$ and therefore its Hessian is constant negative. The results indicate the anticipated behavior that CESP is able to more reliably drive a convergent series towards a solution where the minimum eigenvalue of $\nabla_{\mathbf{x}}^2 f(\mathbf{X}; \mathbf{x}, \mathcal{P})$ is positive.

7.3   Generative Adversarial Networks

This experiment evaluates the performance of the CESP method for training a Generative Adversarial Network (GAN), which reduces to solving the saddle point problem

$$\min_{\mathbf{x}} \max_{\mathbf{y}} \big[f(\mathbf{x}, \mathbf{y}) = \mathbb{E}_{\theta \sim p_d} \log(D_{\mathbf{x}}(\theta))$$
$$+ \mathbb{E}_{\mathbf{z} \sim p_z} \log(1 - D_{\mathbf{x}}(G_{\mathbf{y}}(\mathbf{z}))) \big] \quad (32)$$

where the functions $D : \mathbb{R}^n \to [0,1]$ and $G : \mathbb{R}^m \to \mathbb{R}^n$ are represented by neural networks parameterized with the variables $\mathbf{x}$ and $\mathbf{y}$, respectively. We use the MNIST data set and a simple GAN architecture with 1 hidden layer and 100 units. More details about the network architecture and parameters are summarized in table 2 in the Appendix.

We investigate the advantage of curvature exploitation for ADAGRAD, which is a member of the class of linear-transformed gradient methods often used for saddle point problems. Moreover, we make use of Power iterations as described in section 5 to efficiently approximate the extreme curvature vector. Note that since we're using mini batches in this experiment, we do not have access to the correct gradient information but also rely on an approximation here.

As before, we evaluate the efficacy of the negative curvature step in terms of the spectrum of f at a (approximately) convergent solution $\mathbf{z}^*$. We compare CESP to the vanilla ADAGRAD optimizer. Since we are interested in a solution that gives rise to a locally optimal saddle point, we track (an approximation of) the smallest eigenvalue of $\nabla_{\mathbf{x}}^2 f(\mathbf{z}^*)$ and the largest eigenvalue of $\nabla_{\mathbf{y}}^2 f(\mathbf{z}^*)$ through the optimization. Using these estimates, we can evaluate if a method has converged to a locally optimal saddle point. The results are shown in figure 5. The decrease in terms of the squared norm of the gradients indicates that both methods converge to a solution. Moreover, both fulfill the condition for a locally optimal saddle point for the parameter $\mathbf{y}$, i.e. the maximum eigenvalue of $\nabla_{\mathbf{y}}^2 f(\mathbf{z}^*)$ is negative. However, the graph of the minimum eigenvalue of $\nabla_{\mathbf{x}}^2 f(\mathbf{z}^*)$ shows that CESP converges faster, and with less frequent and
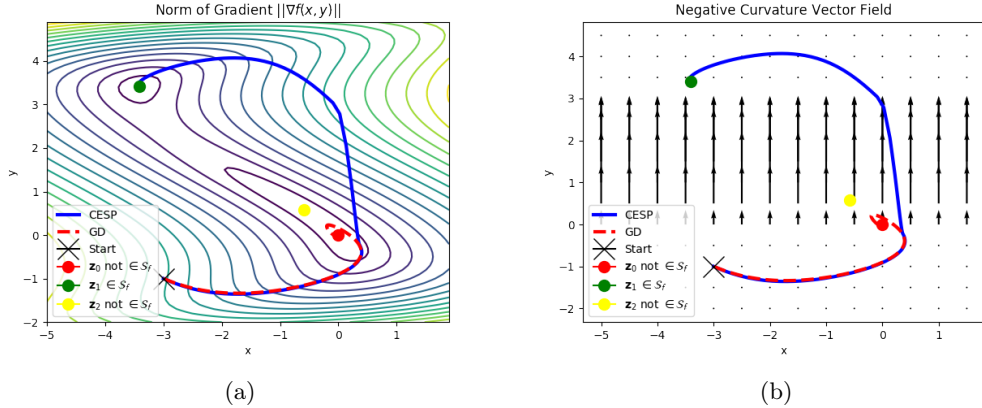
(a)   (b)

Figure 3: Comparison of the trajectory of GD and the CESP proposal on the function in Eq. (17) from the starting point $(-3, -1)$. The background in the right plot shows the vector field of the extreme curvature, as defined in Eq. (20). Note that for the function in Eq. (17) the curvature in the x-dimension is constant and positive, and therefore $\mathbf{v_z^{(-)}}$ is always zero.

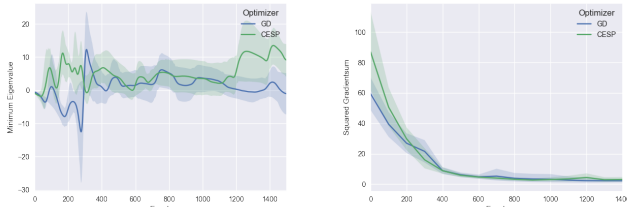Figure 4: The left plot shows the minimum eigenvalue of $\nabla_\mathbf{x}^2 f(X; \mathbf{x}, \mathcal{P})$ and the right plot the squared gradient norm. The solid line shows the mean value whereas the blurred area indicates the 90th percentile. The CESP optimzer is superior in this example because for convergent series where the gradientnorm ≈ 0 the minimum eigenvalue of $\nabla_\mathbf{x}^2 f(X; \mathbf{x}, \mathcal{P}) > 0$; for gradient-based optimization this is not the case.

Figure 5: The first two plots show the minimum eigenvalue of $\nabla_\mathbf{x}^2 f(\mathbf{x}, \mathbf{y})$ and the maximum eigenvalue of $\nabla_\mathbf{y}^2 f(\mathbf{x}, \mathbf{y})$, respectively. The third plot shows $\|\nabla f(\mathbf{z}_t)\|^2$. The transparent graph shows the raw computed values, whereas the solid graph is smoothed with a Gaussian filter.

severe spikes, to a solution where the minimum eigenvalue is zero. Hence, the negative curvature step seems to be able to drive the optimization procedure to regions that yield points closer to a locally optimal saddle point.

Even though this empirical result highlights the benefits of using curvature exploitation, we observe zero eigenvalues of $\nabla_\mathbf{x}^2 f(\mathbf{z}^*)$ for convergent solutions, which violates the conditions required for our analysis. This observation is in accordance with recent empirical evidence [36] showing that the Hessian is actually degenerate for common deep learning architectures. The phenomenon itself as well as approaches to address it are left as future work. One potential direction would be to investigate if high-order derivatives could be used at points where the Hessian is degenerate.
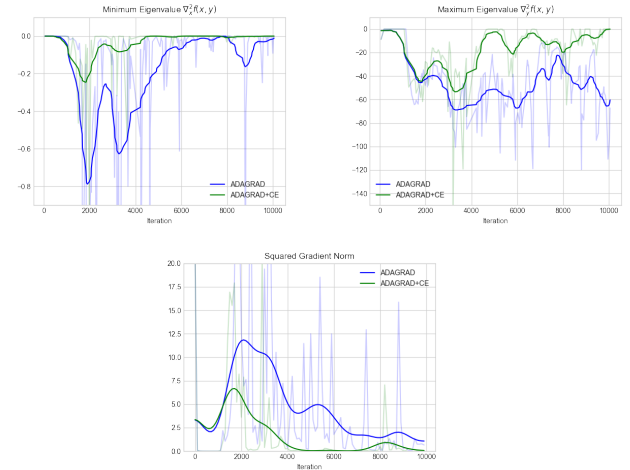
8 CONCLUSION

We focused our study on reaching a solution to the local saddle point problem. First, we have shown that gradient methods have stable stationary points that are not locally optimal, which is a problem exclusively arising in saddle point optimization. Second, we proposed a novel approach that exploits extreme curvature information to avoid the undesired stationary points. We believe this work highlights the benefits of using curvature information for saddle point problems and might open the door to other novel algorithms with stronger global convergence guarantees.

References

[1] Zeyuan Allen-Zhu. Natasha 2: Faster non-convex optimization than sgd. *arXiv preprint arXiv:1708.08694*, 2017.

[2] Zeyuan Allen-Zhu and Yuanzhi Li. Neon2: Finding local minima via first-order oracles. *arXiv preprint arXiv:1711.06673*, 2017.

[3] Kenneth Joseph Arrow, Leonid Hurwicz, Hirofumi Uzawa, and Hollis Burnley Chenery. Studies in linear and non-linear programming. 1958.

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

[5] Michele Benzi, Gene H. Golub, and Jörg Liesen. Numerical solution of saddle point problems. *ACTA NUMERICA*, 14:1–137, 2005.

[6] Coralia Cartis, Nicholas IM Gould, and Philippe L Toint. Adaptive cubic regularisation methods for unconstrained optimization. part i: motivation, convergence and numerical results. *Mathematical Programming*, 127(2):245–295, 2011.

[7] Ashish Cherukuri, Bahman Gharesifard, and Jorge Cortes. Saddle-point dynamics: conditions for asymptotic stability of saddle points. *SIAM Journal on Control and Optimization*, 55(1):486–511, 2017.

[8] Andrew R Conn, Nicholas IM Gould, and Philippe L Toint. *Trust region methods*. SIAM, 2000.

[9] Frank E Curtis and Daniel P Robinson. Exploiting negative curvature in deterministic and stochastic optimization. *arXiv preprint arXiv:1703.00412*, 2017.

[10] Yann N Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2933–2941. Curran Associates, Inc., 2014.

[11] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. Technical Report UCB/EECS-2010-24, EECS Department, University of California, Berkeley, Mar 2010.

[12] Rui Gao and Anton J Kleywegt. Distributionally robust stochastic optimization with wasserstein distance. *arXiv preprint arXiv:1604.02199*, 2016.

[13] Gauthier Gidel, Hugo Berard, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial nets. *arXiv preprint arXiv:1802.10551*, 2018.

[14] EG Golshtein. Generalized gradient method for finding saddlepoints. *Matekon*, 10(3):36–52, 1974.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2672–2680. Curran Associates, Inc., 2014.

[16] Thomas Holding and Ioannis Lestas. On the convergence to saddle points of concave-convex functions, the gradient method and emergence of oscillations. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, pages 1143–1148. IEEE, 2014.

[17] H.K. Khalil. *Nonlinear Systems*. Pearson Education. Prentice Hall, 2002.

[18] Jonas Moritz Kohler and Aurelien Lucchi. Sub-sampled cubic regularization for non-convex optimization. In *International Conference on Machine Learning*, 2017.

[19] GM Korpelevich. The extragradient method for finding saddle points and other problems. *Matecon*, 12:747–756, 1976.

[20] T Kose. Solutions of saddle value problems by differential equations. *Econometrica, Journal of the Econometric Society*, pages 59–70, 1956.

[21] J. Kuczyński and H. Woźniakowski. Estimating the largest eigenvalue by the power and lanczos algorithms with a random start. *SIAM Journal on Matrix Analysis and Applications*, 13(4):1094–1122, 1992.

[22] Jason D Lee, Max Simchowitz, Michael I Jordan, and Benjamin Recht. Gradient descent converges to minimizers. *arXiv preprint arXiv:1602.04915*, 2016.

[23] Kevin Leyton-Brown and Yoav Shoham. *Essentials of Game Theory: A Concise, Multidisciplinary Introduction*. Morgan and Claypool Publishers, 1st edition, 2008.

[24] Tengyuan Liang and James Stokes. Interaction matters: A note on non-asymptotic local convergence of generative adversarial networks. *arXiv preprint arXiv:1802.06132*, 2018.

[25] D Maistroskii. Gradient methods for finding saddle points. *Matekon*, 14(1):3–22, 1977.

[26] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. *arXiv preprint arXiv:1705.10461*, 2017.

[27] Vaishnavh Nagarajan and J Zico Kolter. Gradient descent gan optimization is locally stable. In *Advances in Neural Information Processing Systems*, pages 5591–5600, 2017.

[28] Vaishnavh Nagarajan and J. Zico Kolter. Gradient descent GAN optimization is locally stable. *CoRR*, abs/1706.04156, 2017.

[29] Hongseok Namkoong and John C. Duchi. Variance-based regularization with convex objectives. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 2975–2984, 2017.

[30] Angelia Nedić and Asuman Ozdaglar. Subgradient methods for saddle-point problems. *Journal of optimization theory and applications*, 142(1):205–228, 2009.

[31] Arkadi Nemirovski. Prox-method with rate of convergence o (1/t) for variational inequalities with lipschitz continuous monotone operators and smooth convex-concave saddle point problems. *SIAM Journal on Optimization*, 15(1):229–251, 2004.

[32] AS Nemirovskii and DB Yudin. Cezare convergence of gradient method approximation of saddle points for convex-concave functions. *Doklady Akademii Nauk SSSR*, 239:1056–1059, 1978.

[33] Yurii Nesterov and Boris T Polyak. Cubic regularization of newton method and its global performance. *Mathematical Programming*, 108(1):177–205, 2006.

[34] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in Neural Information Processing Systems*, pages 271–279, 2016.

[35] Barak A. Pearlmutter. Fast exact multiplication by the hessian. *Neural Computation*, 6:147–160, 1994.

[36] Levent Sagun, Leon Bottou, and Yann LeCun. Eigenvalues of the hessian in deep learning: Singularity and beyond. *arXiv preprint arXiv:1611.07476*, 2016.

[37] Satinder Singh, Michael Kearns, and Yishay Mansour. Nash convergence of gradient dynamics in general-sum games. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 541–548. Morgan Kaufmann Publishers Inc., 2000.

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

[39] Hirofumi Uzawa. Iterative methods for concave programming. *Studies in linear and nonlinear programming*, 6:154–165, 1958.

[40] Peng Xu, Farbod Roosta-Khorasani, and Michael W Mahoney. Newton-type methods for non-convex optimization under inexact hessian information. *arXiv preprint arXiv:1708.07164*, 2017.

[41] Yi Xu and Tianbao Yang. First-order stochastic algorithms for escaping from saddle points in almost linear time. *arXiv preprint arXiv:1711.01944*, 2017.