Direct-Search for a Class of Stochastic Min-Max Problems

Sotiris Anagnostidis
Institute for Machine Learning
ETH Zürich, Switzerland
sotirios.anagnostidis@inf.ethz.ch

Aurelien Lucchi
Institute for Machine Learning
ETH Zürich, Switzerland
aurelien.lucchi@inf.ethz.ch

Youssef Diouane
ISAE-SUPAERO
Université de Toulouse, France
youssef.diouane@isae-supaero.fr

Abstract

Recent applications in machine learning have renewed the interest of the community in min-max optimization problems. While gradient-based optimization methods are widely used to solve such problems, there are however many scenarios where these techniques are not well-suited, or even not applicable when the gradient is not accessible. We investigate the use of direct-search methods that belong to a class of derivative-free techniques that only access the objective function through an oracle. In this work, we design a novel algorithm in the context of min-max saddle point games where one sequentially updates the min and the max player. We prove convergence of this algorithm under mild assumptions, where the objective of the max-player satisfies the Polyak-Łojasiewicz (PL) condition, while the min-player is characterized by a nonconvex objective. Our method only assumes dynamically adjusted accurate estimates of the oracle with a fixed probability. To the best of our knowledge, our analysis is the first one to address the convergence of a direct-search method for min-max objectives in a stochastic setting.

1 INTRODUCTION

Recent applications in the field of machine learning, including generative models (Goodfellow et al., 2014) or robust optimization (Ben-Tal et al., 2009), have triggered significant interest for the optimization of min-max functions of the form

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x},\mathbf{y}) = \mathbb{E}[\tilde{f}(\mathbf{x},\mathbf{y},\xi)], \tag{1}$$

where ξ is a random variable characterized by some distribution. In machine learning, ξ is for instance often drawn from a distribution that depends on the training data.

In practice, min-max problems are often solved using gradient-based algorithms, especially simultaneous gradient descent ascent (GDA) that simply alternates between a gradient descent step for $\mathbf{x}$ and a gradient ascent step for $\mathbf{y}$. While these algorithms are attractive due to their simplicity, there are however cases where the gradient of the objective function is not accessible, such as when modelling distributions with categorical variables (Jang et al., 2016), tuning hyper-parameters (Audet and Orban, 2006; Marzat et al., 2011) and multi-agent reinforcement learning with bandit feedback (Zhang et al., 2019). A resurgence of interest has recently emerged for applications in black-box optimization (Bogunovic et al., 2018; Liu et al., 2019) and black-box poisoning attack (Liu et al., 2020), where an attacker deliberately modifies the training data in order to tamper with the model's predictions. This can be formulated as a min-max optimization problem, where only stochastic accesses to the objective function are available (Wang et al., 2020).

In this work, we investigate the use of direct-search methods to optimize min-max objective functions without requiring access to the gradients of the objective f. Direct-search methods have a long history in the field of optimization, dating back from the seminal paper of Hooke and Jeeves (1961). The appeal of these methods is due to their simplicity but also potential ability to deal with non-trivial objective functions. Although there are variations among random search techniques, most of them can be summarized conceptually as sampling random directions from a search space and moving towards directions that decrease the objective function value. We note that these

Proceedings of the 24th International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

techniques are sometimes named derivative-free methods, but it is important to distinguish them from other techniques that try to estimate derivatives based on finite difference (Spall, 2003) or smoothing (Nesterov and Spokoiny, 2017). We refer the reader to the surveys by Lewis et al. (2000); Rios and Sahinidis (2013) for a comprehensive review of direct-search methods.

Solving the saddle point problem (1) is equivalent to finding a saddle point[1] $(\mathbf{x}^*, \mathbf{y}^*)$ such that

$$f(\mathbf{x}^*, \mathbf{y}) \leq f(\mathbf{x}^*, \mathbf{y}^*) \leq f(\mathbf{x}, \mathbf{y}^*) \quad \forall \mathbf{x} \in \mathcal{X}, \quad \forall \mathbf{y} \in \mathcal{Y}.$$

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 $\mathbf{y}$) that are differentiable. 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. This for instance includes applications such as Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), robust optimization (Ben-Tal et al., 2009; Bogunovic et al., 2018) and multi-agent reinforcement learning (Omidshafiei et al., 2017). For min-max problems without access to derivatives, the literature is in fact very scarce. Most existing techniques such as Hare and Macklem (2013); Hare and Nutini (2013); Custódio et al. (2021) consider finite-max functions, i.e., functions of the form $f(\mathbf{x}) = \max\{f_i(\mathbf{x}) : i = 1, 2, \ldots, N\}$ where $N > 0$ is finite and each f_i is continuously differentiable. Other techniques such as Bertsimas and Nohadani (2010); Bertsimas et al. (2010) are restricted to functions f that are convex with respect to $\mathbf{x}$ or only provide asymptotic convergence analysis (Menickelly and Wild, 2020). We refer the reader to Section 2 for a more detailed discussion of prior approaches.

Motivated by a wide range of applications, we therefore focus on a nonconvex and nonconcave stochastic setting where the max player satisfies the PL condition (see Definition 5 in Section 4), which is known to be a weaker assumption compared to convexity (Karimi and Schmidt, 2015). In summary, our main contributions are:

- We design a novel direct-search algorithm for such min-max problems and provide non-asymptotic convergence guarantees in terms of first-order Nash equilibrium. Concretely, we prove convergence to an ϵ-first-order Nash Equilibrium (for a definition see Section 3) in $\mathcal{O}(\epsilon^{-2}\log(\epsilon^{-1}))$ iterations, which is comparable to the rate achieved by gradient-based techniques (Nouiehed et al., 2019).
- We derive theoretical convergence guarantees in a stochastic setting where one only has access to accurate estimates of the objective function, with some fixed probability. We prove our results for the case where the min player optimizes a nonconvex function while the max player optimizes a PL function.
- We validate empirically our theoretical findings, including settings where derivatives are not available.

2 RELATED WORK

Direct-search methods for minimization problems The general principle behind direct-search methods is to optimize a function $f(\mathbf{x})$ without having access to its gradient $\nabla f(\mathbf{x})$. There is a large number of algorithms that are part of this broad family including golden-section search techniques or random search (Rastrigin, 1963). Among the most popular algorithms in machine learning are evolution strategies and population-based algorithms that have demonstrated promising results in reinforcement learning (Salimans et al., 2017; Maheswaranathan et al., 2018) and bandit optimization (Flaxman et al., 2004). At a high-level, these techniques work by maintaining a distribution over parameters and duplicate the individuals in the population with higher fitness. Often these algorithms are initialized at a random point and then adapt their search space, depending on which area contains the best samples (i.e. the lowest function value when minimizing $f(\mathbf{x})$). New samples are then generated from the best regions in a process repeated until convergence. The most well-known algorithms that belong to this class are evolutionary-like algorithms, including for instance CMA-ES (Hansen et al., 2003). Evolutionary strategies have recently been shown to be able to solve various complex tasks in reinforcement learning such as Atari games or robotic control problems, see e.g. Salimans et al. (2017). Their advantages in the context of reinforcement learning are their reduced sensitivity to noisy or uninformative gradients (potentially increasing their ability to avoid local minima (Conti et al., 2017)) and the ease with which one can implement a distributed or parallel version.

Convergence guarantees for direct-search methods Proofs of convergence for direct-search methods are based on a specific construction for the sampling directions, often that they positively span the whole search space (Conn et al., 2009), or that they are dense in certain types of directions (known as refining directions) at the limit point (Audet and Dennis Jr, 2006). In addition, they also typically

[1] In the game theory literature, such point is commonly referred to as (global) Nash equilibrium, see e.g. Liang and Stokes (2018).

rely on the use of a forcing function that imposes each new selected iterate to decrease the function value adequately. This technique has been analyzed in Vicente (2013) who proved convergence under mild assumptions in $\mathcal{O}(\epsilon^{-2})$ iterations for the goal $\|\nabla f(\mathbf{x})\| < \epsilon$. The number of required steps is reduced to $\mathcal{O}(\epsilon^{-1})$ for convex functions f, and to $\mathcal{O}(\log(\epsilon^{-1}))$ for strongly-convex functions (Kone**v**cnỳ and Richtárik, 2014). This is on par with the steepest descent method for unconstrained optimization (Nesterov, 2013) apart from some constants that depend on the dimensionality of the problem.

Stochastic estimates of the function In our analysis, we only assume access to stochastic estimates of the objective function.

$$f(\mathbf{x}, \mathbf{y}) = \mathbb{E}[\tilde{f}(\mathbf{x}, \mathbf{y}, \xi)], \tag{2}$$

where ξ is a random variable that captures the randomness of the objective function. The origin of the noise could be privacy related, or caused by a noisy adversary. Most commonly, it might arise from online streaming data, distributed and batch-sized updates due to the sheer size of the problem. Stochastic gradient descent is often used to optimize Eq. (2), where one often assumes access to accurate estimates of f and consider updates only in expectation (Johnson and Zhang, 2013). To establish similar convergence rates to the deterministic case, an alternative solution consists of adapting the accuracy of these estimates dynamically, which can be ensured by averaging multiple samples together. This approach has for instance been analyzed in the context of trust-region methods (Blanchet et al., 2019) and line-search methods (Paquette and Scheinberg, 2018; Bergou et al., 2018), including direct-search for the minimization of nonconvex functions (Dzahini, 2020).

Algorithms for finding equilibria in games Since the pioneering work of von Neumann (1928), equilibria in games have received great attention. Most past results focus on convex-concave settings (Chen et al., 2014; Hien et al., 2017). Notably, Cherukuri et al. (2017) studied convergence of the GDA algorithm under strictly convex-concave assumptions. For problems where the function does not satisfy this condition however, convergence to a saddle point is not guaranteed. More recent results focus on relaxing these conditions. The work of Nouiehed et al. (2019) analyzed gradient descent-ascent under a similar scenario, where the objective of the max player satisfies the PL condition and where the min player optimizes a nonconvex objective. Ostrovskii et al. (2020); Wang et al. (2020) analyze a nonconvex-concave class of problems, while Lin et al. (2020) present a two-scale variant of the GDA algorithm for a similar scenario, providing a replacement for the alternating updates scheme.

We take inspiration from the work of Liu et al. (2019); Nouiehed et al. (2019); Sanjabi et al. (2018) to design a novel alternating direct-search algorithm, where the inner maximization problem is solved almost exactly before performing a single step towards improving the strategy of the minimization player. We are able to prove convergence of our direct-search algorithm under this procedure, which has been proven to be more stable than the analogous simultaneous one, as rigorously shown in Gidel et al. (2018) and Zhang and Yu (2019) for a variety of algorithms.

3 PRELIMINARIES

Throughout, we use $\|.\|$ to denote the Euclidean norm; that is, for $\mathbf{x} \in \mathbb{R}^n$ we have $\|\mathbf{x}\| = \sqrt{\mathbf{x}^\intercal \mathbf{x}}$.

3.1 Min-Max Games

We consider the optimization problem defined in Eq. (1) for which a common notion of optimality is the concept of Nash equilibrium as mentioned previously, which is formally defined as follows.

Definition 1. *We say that a point* $(\mathbf{x}^*, \mathbf{y}^*) \in \mathcal{X} \times \mathcal{Y}$ *is a Nash equilibrium of the game if*

$$f(\mathbf{x}^*, \mathbf{y}) \leq f(\mathbf{x}^*, \mathbf{y}^*) \leq f(\mathbf{x}, \mathbf{y}^*) \quad \forall \mathbf{x} \in \mathcal{X}, \quad \forall \mathbf{y} \in \mathcal{Y}.$$

A Nash equilibrium is a point where the change of strategy of each player individually does not lead to an improvement from her viewpoint. Such a Nash equilibrium point always exists for convex-concave games (Jin et al., 2019), but not necessarily for nonconvex-nonconcave games. Even when they exist, finding Nash equilibria is known to be a NP-hard problem, which has led to the introduction of local characterizations as discussed in Jin et al. (2019); Adolphs et al. (2018). Here we use the notion of a first-order Nash equilibrium (FNE) (for a definition we refer to Pang and Razaviyayn (2016)). We focus on the problem of converging to such a FNE point, or an approximate FNE defined as follows (adapted from Nouiehed et al. (2019) in the absence of constraints).

Definition 2. *For a function* $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, *a point* $(\mathbf{x}^*, \mathbf{y}^*) \in \mathbb{R}^n \times \mathbb{R}^m$ *is said to be an* ϵ*-first-order Nash Equilibrium (*ϵ*-FNE) if:* $\|\nabla_{\mathbf{x}} f(\mathbf{x}^*, \mathbf{y}^*)\| \leq \epsilon$ *and* $\|\nabla_{\mathbf{y}} f(\mathbf{x}^*, \mathbf{y}^*)\| \leq \epsilon$.

3.2 Direct-Search Methods

Spanning set Direct-search methods typically rely on the smoothness of the objective function, which we denote by $f : \mathbb{R}^n \rightarrow \mathbb{R}$ in this section[2], and on appropriate choice of sampling points to prove convergence. The key idea to guarantee convergence is that one of the sampled directions will form an acute angle with the negative gradient. This can be ensured by sampling from a Positive Spanning Set (PSS). The quality of a spanning set $\mathcal{D}$ is typically measured using a notion of cosine measure defined as

$$\kappa(\mathcal{D}) = \min_{\mathbf{0} \neq \mathbf{u} \in \mathbb{R}^n} \max_{\mathbf{d} \in \mathcal{D}} \frac{\mathbf{u}^T \mathbf{d}}{\|\mathbf{u}\| \|\mathbf{d}\|}. \tag{3}$$

In the following, we will consider positive spanning sets such that $\kappa(\mathcal{D}) \geq \kappa_{\min} > 0$ and $d_{\min} \leq \|\mathbf{d}\| \leq d_{\max}, \forall \mathbf{d} \in \mathcal{D}$. These assumptions require $|\mathcal{D}| \geq n+1$. Common choices are i) the positive and negative orthonormal bases $\mathcal{D} = [I_n \; -I_n] = [\mathbf{e}_1, \ldots, \mathbf{e}_n, -\mathbf{e}_1, \ldots, -\mathbf{e}_n]$ of size $|\mathcal{D}| = 2n$, ii) a minimal positive basis with uniform angles of size $|\mathcal{D}| = n+1$ (see Corollary 2.6 of Conn et al. (2009) and Kolda et al. (2003)) or iii) even rotations of these matrices (Gratton et al., 2016).

```
Algorithm 1: DIRECT-SEARCH($f, \mathbf{x}_0, c, T$)
Input: $f$: objective function, with $f_k$ it's
         estimate at step $k$
       $c$: forcing function constant
       $T$: number of steps
Initialize step size value $\sigma_0$. Choose $\gamma > 1$.
Create the Positive Spanning Set $\mathcal{D}$.
for $k$ = 0, ..., T - 1 do
    1. Offspring generation:
    Generate the points
        $\mathbf{x}^i = \mathbf{x}_k + \sigma_k \mathbf{d}^i, \quad \forall \mathbf{d}^i \in \mathcal{D}$.
    2. Parent Selection:
    Choose $\mathbf{x}' = \arg\min_i f_k(\mathbf{x}^i)$.
    3. Sufficient Decrease:
    if $f_k(\mathbf{x}') < f_k(\mathbf{x}_k) - \rho(\sigma_k)$ then
        (Iteration is successful)
        Update and increase step size
        $\mathbf{x}_{k+1} = \mathbf{x}', \sigma_{k+1} = \min\{\sigma_{\max}, \gamma\sigma_k\}$.
    else
        (Iteration is unsuccessful)
        Decrease step size
        $\mathbf{x}_{k+1} = \mathbf{x}_k$, $\sigma_{k+1} = \gamma^{-1}\sigma_k$.
    end
end
return $\mathbf{x}_T$
```

Forcing function Another critical component to guarantee that the function value decreases at each step appropriately is a forcing function ρ that satisfies $\frac{\rho(\sigma)}{\sigma} \rightarrow 0$ when $\sigma \rightarrow 0$. Given such σ, direct-search methods sample new points according to the rule

$$\mathbf{x}' = \mathbf{x} + \sigma \mathbf{d}, \tag{4}$$

and accept points for which

$$f(\mathbf{x}') < f(\mathbf{x}) - \rho(\sigma), \tag{5}$$

$\mathbf{d} \in \mathcal{D}$. If the previous condition holds for some $\mathbf{d} \in \mathcal{D}$, then the new point is accepted, the step is deemed successful and the σ parameter is increased, otherwise σ is decreased and the above process is repeated. We use a parameter γ to indicate these updates of the step size. For convenience and without loss of generality, we will only consider spanning sets with vectors of unitary length $d_{\min} = d_{\max} = 1$ and a forcing function

$$\rho(\sigma) = c\sigma^2.$$

The direct-search scheme is displayed in Algorithm 1.

4 STOCHASTIC DIRECT-SEARCH

The full algorithm we analyze to solve the min-max objective is presented in Algorithm 2. It consists of two steps: i) first solve the maximization problem w.r.t. the $\mathbf{y}$ variable using Algorithm 1, and ii) perform one update step for the $\mathbf{x}$ variable. In this section, we first analyze the convergence properties of Algorithm 1 in the setting where we only have access to estimates of the objective function f,

$$f(\mathbf{x}) = \mathbb{E}[\tilde{f}(\mathbf{x}, \xi)].$$

Let $(\Omega, \mathcal{F}, P)$ be a probability space with elementary events denoted with ω. We denote the random quantities for the iterate by $\mathbf{x}_k = \mathbf{X}_k(\omega)$ and for the step size by $\sigma_k = \Sigma_k(\omega)$. Similarly let $\{F_k^0, F_k^\sigma\}$ be the estimates of $f(\mathbf{X}_k)$ and $f(\mathbf{X}_k + \Sigma_k \mathbf{d}_k)$, for each $\mathbf{d}_k$ in a set $\mathcal{D}$, with their realizations $f_k^0 = F_k^0(\omega)$, $f_k^\sigma = F_k^\sigma(\omega)$. At each iteration the influence of the noise on function evaluations is random. We will assume that, when conditioned on all the past iterates, these estimates are sufficiently accurate with a sufficiently high probability. We formalize this concept in the two definitions below.

Definition 3. *(ϵ_f-accurate) The estimates $\{F_k^0, F_k^\sigma\}$ are said to be ϵ_f-accurate with respect to the corresponding sequence if*

$$|F_k^0 - f(\mathbf{X}_k)| \leq \epsilon_f \Sigma_k^2 \;\; \textit{and} \;\; |F_k^\sigma - f(\mathbf{X}_k + \Sigma_k \mathbf{d}_k)| \leq \epsilon_f \Sigma_k^2.$$

Definition 4. *(p_f-probabilistically ϵ_f-accurate) The estimates $\{F_k^0, F_k^\sigma\}$ are said to be p_f-probabilistically*

[2] When using the function f with one set of variables, we consider the minimization problem. When using two sets of variables, we instead consider the min-max problem as defined in Eq. (1).

ϵ_f-accurate with respect to the corresponding sequence if the events

$$J_k = \{\textit{The estimates } \{F_k^0, F_k^\sigma\} \textit{ are } \epsilon_f\textit{-accurate}\}$$

satisfy the condition[3]

$$P(J_k \mid \mathcal{F}_{k-1}) = \mathbb{E}[1_{J_k} \mid \mathcal{F}_{k-1}] \geq p_f,$$

where $\mathcal{F}_{k-1}$ is the sigma-algebra generated by the sequence $\{F_0^0, F_0^\sigma, \dots, F_{k-1}^0, F_{k-1}^\sigma\}$.

As the step size σ gets smaller, meaning that we are getting closer to the optimum, we require the accuracy over the function values to increase. However, the probability to encounter a good estimation remains the same throughout. A significant challenge arises, as steps may satisfy our sufficient decrease condition specified in Eq. (5) falsely, leading to a potential increase in terms of the objective value. This increase can potentially be very large, leading to divergence, and we therefore need to require an additional assumption regarding the variance of the error.

Assumption 1. *The sequence of estimates $\{F_k^0, F_k^\sigma\}$ are said to satisfy a l_f-variance condition if for all $k \geq 0$*

$$\mathbb{E}[|F_k^0 - f(\mathbf{X}_k)|^2 \mid \mathcal{F}_{k-1}] \leq l_f^2 \Sigma_k^4,$$
$$\mathbb{E}[|F_k^\sigma - f(\mathbf{X}_k + \Sigma_k \mathbf{d}_k)|^2 \mid \mathcal{F}_{k-1}] \leq l_f^2 \Sigma_k^4.$$

Based on the above assumptions, we reach the following conclusion regarding inaccurate steps (similar to Lemma 2.5 in Paquette and Scheinberg (2018)).

Lemma 1. *Let Assumption 1 hold for p_f-probabilistically ϵ_f-accurate estimates of a function. Then for $k \geq 0$ we have*

$$\mathbb{E}[1_{J_k^c} |F_k^0 - f(\mathbf{X}_k)| \mid \mathcal{F}_{k-1}] \leq (1-p_f)^{1/2} l_f \Sigma_k^2,$$
$$\mathbb{E}[1_{J_k^c} |F_k^\sigma - f(\mathbf{X}_k + \Sigma_k \mathbf{d}_k)| \mid \mathcal{F}_{k-1}] \leq (1-p_f)^{1/2} l_f \Sigma_k^2.$$

Computing the estimates In order to satisfy Assumption 1 we can perform multiple function evaluations and average them out (see for instance Tropp (2015)). We therefore get an estimate $F_k^0 = \frac{1}{|S_k^0|} \sum_{\xi_i \in S_k^0} \tilde{f}(\mathbf{X}_k, \xi_i)$, where S_k^0, S_k^σ correspond to independent samples for F_k^0 and F_k^σ respectively. Assuming bounded variance, i.e. $\mathbb{E}[|\tilde{f}(\mathbf{x}, \xi) - f(\mathbf{x})|^2] \leq \sigma_f^2$, known concentration results (see e.g. Tripuraneni et al. (2018); Chen et al. (2018)) guarantee that we can obtain p_f-probabilistically ϵ_f-accurate estimates for

$$|S_k^0| \geq \mathcal{O}(1)\left(\frac{\sigma_f^2}{\epsilon_f^2 \Sigma_k^4} \log\left(\frac{1}{1-p_f}\right)\right)$$

number of evaluations (the same result holds for S_k^σ). To also satisfy Assumption 1, we additionally require $|S_k^0| \geq \frac{\sigma_f^2}{l_f \Sigma_k^4}$.

4.1 Convergence of Stochastic Direct-Search

In order to study the convergence properties of Algorithm 1, we introduce the following (random) Lyapunov function:

$$\Phi_k = v(f(\mathbf{X}_k) - f^*) + (1-v)\Sigma_k^2,$$

where $v \in (0,1)$ is a constant. We denote by f^* the minimum of the function f, assumed to exist and potentially achieved at multiple positions. The Lyapunov function Φ_k will be used to track the progress of the gradient norm $\|\nabla f(\mathbf{X}_k)\|$, which will serve as a measure of convergence.

Theorem 2 presented below ensures that the Lyapunov function decreases over iterations. Using this result, one can guarantee that the sequence of step-sizes decreases and then exploit the fact that for sufficiently small step sizes (and accurate estimates), the steps are successful, i.e. they decrease the objective function. The proof of the next Theorem is mainly inspired by Dzahini (2020); Audet et al. (2021).

Theorem 2. *Let a function f with a minimum value f^*, with Lipschitz continuous gradients with a constant L. Let also f be p_f-probabilistically ϵ_f-accurate, while also having bounded noise variance according to Assumption 1 with constant l_f. Then:*

$$\mathbb{E}[\Phi_{k+1} - \Phi_k \mid \mathcal{F}_{k-1}] \leq -p_f(1-v)(1-\frac{1}{\gamma^2})\frac{\Sigma_k^2}{2}. \quad (6)$$

The constants c, v and p_f should satisfy

$$c - 2\epsilon_f > 0, \qquad \frac{p_f}{\sqrt{1-p_f}} \geq \frac{4vl_f}{(1-v)(1-\gamma^{-2})},$$
$$\frac{v}{1-v} \geq \frac{1}{c-2\epsilon_f}(\gamma^2 - \frac{1}{\gamma^2}).$$

Next, we characterize the number of steps required to converge by using a renewal-reward process adapted from Blanchet et al. (2019). Let us define the random process $\{\Phi_k, \Sigma_k\}$, with $\Phi_k \geq 0$ and $\Sigma_k \geq 0$. Let us also denote with W_k a random walk process and $\mathcal{F}_k$ the

[3] We use 1_A to denote the indicator function for the set A and A^c to denote its complement.

σ-algebra generated by $\{\Phi_0, \Sigma_0, W_0, \ldots, \Phi_k, \Sigma_k, W_k\}$ with $W_0 = 1$,

$$\begin{aligned} P(W_{k+1} = 1 \mid \mathcal{F}_k) &= p, \\ P(W_{k+1} = -1 \mid \mathcal{F}_k) &= 1 - p. \end{aligned} \tag{7}$$

We also define a family of stopping times $\{T_\epsilon\}_{\epsilon>0}$ with respect to $\{\mathcal{F}_k\}_{k\geq 0}$ for $\epsilon > 0$.

Assumption 2. *Given the random quantities $\{\Phi_k, \Sigma_k, W_k\}$, we make the following assumptions.*

i. There exists $\lambda > 0$ such that $\Sigma_{\max} = \Sigma_0 e^{\lambda j_{\max}}$ for $j_{\max} \in \mathbb{Z}$, and $\Sigma_k \leq \Sigma_{\max}$ for all k.

ii. There exists $\Sigma_\epsilon = \Sigma_0 e^{\lambda j_\epsilon}$ with $j_\epsilon \in \mathbb{Z}$, such that

$$1_{T_\epsilon > k} \Sigma_{k+1} \geq 1_{T_\epsilon > k} \min\{\Sigma_k e^{\lambda W_{k+1}}, \Sigma_\epsilon\}$$

where W_{k+1} satisfies Equation (7) with probability $p > \frac{1}{2}$.

iii. There exists a nondecreasing function $h(\cdot)\colon [0, \infty] \to (0, \infty)$ and a constant $\Theta > 0$ such that

$$1_{T_\epsilon > k} \mathbb{E}[\Phi_{k+1} \mid \mathcal{F}_k] \leq 1_{T_\epsilon > k}(\Phi_k - \Theta h(\Sigma_k)).$$

Assumption 2 (ii) requires that step sizes tend to increase when below a specific threshold, while Assumption 2 (iii) requires that the random function Φ decreases in expectation (already proved in Theorem 2). Under this assumption, the following results hold for the stopping time T_ϵ (Blanchet et al., 2019).

Theorem 3. *Under Assumption 2, we have*

$$\mathbb{E}[T_\epsilon] \leq \frac{p}{2p-1} \frac{\Phi_0}{\Theta h(\Sigma_\epsilon)} + 1.$$

In our case, we use the fundamental result of convergence for direct-search methods, that comes from correlating the norm of the gradient with the step size for unsuccessful iterations (generalization of results in Vicente (2013); Gratton et al. (2016)).

Lemma 4. *Let $f : \mathbf{x} \in \mathbb{R}^n \to \mathbb{R}$ be a continuous differentiable function with Lipschitz continuous gradients of constant L. Let also $\mathcal{D}$ be a positive spanning set with $\kappa(\mathcal{D}) = \kappa_{\min} > 0$ and vectors $\mathbf{d}$ satisfying $\|\mathbf{d}\| = 1$, $\forall \mathbf{d} \in \mathcal{D}$. For a forcing function $\rho(\sigma) = c\sigma^2$ and an ϵ_f-accurate estimates of the function, for an unsuccessful step k it holds that*

$$\sigma_k \geq C\|\nabla f(\mathbf{x}_k)\|, \quad C = \frac{2\kappa_{\min}}{L + 2c + 4\epsilon_f}. \tag{8}$$

In this analysis, our goal is to show that the norm of the gradient decreases below a threshold

$$T_\epsilon = \inf\{k \geq 0 : \|\nabla f(\mathbf{X}_k)\| \leq \epsilon\}.$$

We assume that Assumption 2 (i) holds by the choice of $\Sigma_{\max}$. We also know from Lemma 4 that for $\|\nabla f(\mathbf{X})\| > \epsilon$ and $\Sigma \leq C\epsilon$ then a successful step occurs, provided that estimates are accurate. Then following Lemma 4.10 from Paquette and Scheinberg (2018) we get that Assumption 2 (ii) also holds, for $\Sigma_\epsilon = C\epsilon$. Based on the results of Theorem 3 and Lemma 4, we can now prove convergence for a nonconvex bounded function.

Theorem 5. *Assume that the Assumptions of Theorem 2 hold with additionally $p_f > \frac{1}{2}$. Then to get $\|\nabla f(\mathbf{X}_k)\| \leq \epsilon$, the expected stopping time of Algorithm 1 is*

$$\mathbb{E}[T_\epsilon] \leq \mathcal{O}(1) \frac{\kappa_{\min}^{-2}}{2p_f - 1} (f(\mathbf{X}_0) - f^* + \Sigma_0^2)(L + c + \epsilon_f)^2 \frac{1}{\epsilon^2}.$$

Note that for the deterministic scenario where $\epsilon_f = l_f = 0$, the above bound matches known results of direct-search in the nonconvex case (Vicente, 2013; Konevcný and Richtárik, 2014). We now establish faster convergence for a function f, additionally satisfying the PL condition, defined below.

Definition 5. *(Polyak-Łojasiewicz Condition). A differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ with the minimum value $f^* = \min_{\mathbf{x}} f(\mathbf{x})$ is said to be μ-Polyak-Łojasiewicz (μ-PL) if:*

$$\frac{1}{2}\|\nabla f(\mathbf{x})\|^2 \geq \mu(f(\mathbf{x}) - f^*).$$

The PL condition is the weakest among a large family of function classes that include convex functions and other nonconvex ones (Karimi and Schmidt, 2015). Again we can guarantee convergence that closely matches results for deterministic direct-search under strong convexity, by proving that the number of iterations required to halve the distance to the optimum objective value is constant in terms of the accuracy ϵ.

Theorem 6. *Let a function f with a minimum value f^* and satisfying the PL condition with a constant μ and Lipschitz continuous gradients with a constant L. Let also f be p_f-probabilistically ϵ_f-accurate, while also having bounded noise variance according to Assumption 1 with constant l_f. Then to get $\|\nabla f(\mathbf{X}_k)\| \leq \epsilon$, the expected stopping time of*

Algorithm 1 is

$$\mathbb{E}[T_\epsilon] \leq \mathcal{O}(1) \frac{\kappa_{\min}^{-2}(c+L)^2}{(2p_f-1)\mu} \left(1+\frac{1}{c}\right) \log\left(\frac{L(f(\mathbf{X}_0)-f^*)}{\epsilon}\right). \tag{9}$$

The constants c, v and $p_f > \frac{1}{2}$ should satisfy

$$c > \max\{4\epsilon_f, 2\sqrt{2}l_f\}, \ \ \frac{p_f}{\sqrt{1-p_f}} \geq \frac{4vl_f}{(1-v)(1-\gamma^{-2})}$$

$$\textit{and } \frac{v}{1-v} \geq \max\left\{\frac{1}{c-2\epsilon_f}(\gamma^2-\frac{1}{\gamma^2}), \frac{72\gamma^2}{c}\right\}.$$

5 ALGORITHM & CONVERGENCE GUARANTEES

We now focus on the min-max problem presented in Eq. (1). To proceed, we make the following standard assumptions regarding the smoothness of f.

Assumption 3. *The function f is continuously differentiable in both $\mathbf{x}$ and $\mathbf{y}$ and there exist constants L_{11}, L_{12}, L_{21} and L_{22} such that for every $\mathbf{x}, \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$ and $\mathbf{y}, \mathbf{y}_1, \mathbf{y}_2 \in \mathcal{Y}$*

$$\begin{aligned}
\|\nabla_{\mathbf{x}} f(\mathbf{x}_1, \mathbf{y}) - \nabla_{\mathbf{x}} f(\mathbf{x}_2, \mathbf{y})\| &\leq L_{11}\|\mathbf{x}_1 - \mathbf{x}_2\|, \\
\|\nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}_1) - \nabla_{\mathbf{x}} f(\mathbf{x}, \mathbf{y}_2)\| &\leq L_{21}\|\mathbf{y}_1 - \mathbf{y}_2\|, \\
\|\nabla_{\mathbf{y}} f(\mathbf{x}_1, \mathbf{y}) - \nabla_{\mathbf{y}} f(\mathbf{x}_2, \mathbf{y})\| &\leq L_{12}\|\mathbf{x}_1 - \mathbf{x}_2\|, \\
\|\nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}_1) - \nabla_{\mathbf{y}} f(\mathbf{x}, \mathbf{y}_2)\| &\leq L_{22}\|\mathbf{y}_1 - \mathbf{y}_2\|.
\end{aligned}$$

We require that the objective of the max-player satisfies the PL condition.

Assumption 4. *There exists a constant $\mu > 0$ such that the function $-f(\mathbf{x}, \mathbf{y})$ in problem (1) is μ-PL for any $\mathbf{x} \in \mathcal{X}$.*

Following prior works on PL games, e.g. Nouiehed et al. (2019), we propose a sequential scheme for the updates of the two players presented in Algorithm 2 (for simplicity some of the algorithm's constants are not depicted). This multi-step algorithm solves the maximization problem up to some accuracy, and it then performs a single (successful) Direct-Search (DR) step for the minimization problem (see Algorithm 3).

We formalize our Assumptions and our final result.

Assumption 5. *The function f is defined on the whole domain $\mathcal{X} \times \mathcal{Y} = \mathbb{R}^{|\mathcal{X}|} \times \mathbb{R}^{|\mathcal{Y}|}$. We also require f to be bounded below for every $\mathbf{y} \in \mathcal{Y}$ and bounded above for every $\mathbf{x} \in \mathcal{X}$.*

Algorithm 2: Min-Max-Direct-search

```
Input: $f$: objective function
       $(\mathbf{x}_0, \mathbf{y}_0)$: initial point
       $\sigma_0$: initial step for the min problem
for $t$ = 1, ..., $T$ do
    $\mathbf{y}_t$ = Direct-search$(-f(\mathbf{x}_{t-1}, .), \mathbf{y}_{t-1})$
    $\mathbf{x}_t, \sigma_t$ = One-Step-Direct-search
                  $(f(., \mathbf{y}_t), \mathbf{x}_{t-1}, \sigma_{t-1})$
end
return $(\mathbf{x}_T, \mathbf{x}_T)$.
```

Theorem 7. *Suppose that the objective function $f(\mathbf{x}, \mathbf{y})$ satisfies Assumptions 3, 4 and 5. If the estimates are deterministic, then Algorithm 2 converges to an ϵ-FNE within $\mathcal{O}(\epsilon^{-2}\log(\epsilon^{-1}))$ steps. When $f(\mathbf{x}, \mathbf{y})$ is ϵ_x-accurate with probability p_x for every $\mathbf{y}$ satisfying assumptions of Theorem 5 and ϵ_y-accurate with probability p_y for every $\mathbf{x}$, satisfying assumptions of Theorem 6, then with a probability at least δ, Algorithm 2 convergences and the expected number of steps to converge to reach an ϵ-FNE is*

$$\mathcal{O}\Big(\frac{1}{(2p_x-1)(2p_y-1)}\epsilon^{-2}\Big(\log(\epsilon^{-1}) + \left[\log\left(\frac{1-p_x}{p_x}\right)\right]^{-1} \log\left(1-e^{\frac{1}{(2p_x-1)\epsilon^{-2}}\log\delta}\right)\Big)\Big).$$

Algorithm 2 performs in total $\mathcal{O}(\epsilon^{-2})$ updates for the minimization problem, and each minimization update requires $\mathcal{O}(\log(\epsilon^{-1}))$ updates for the maximization problem. The proof of Theorem 7 consists in showing that the maximization problem is solved with sufficient accuracy, for which we invoke the result of Theorem 6. We then proceed by showing that iteratively solving the minimization problem allows us to converge in terms of the min-max objective, which is done using the result of Theorem 5. We note that the sufficient decrease condition allows us to prove convergence for the last iterate instead of relying on the existence of an iterate k in the whole sequence that satisfies the required inequalities (as proven in the corresponding gradient based method by Nouiehed et al. (2019)).

6 EXPERIMENTS

One advantage of direct-search methods is their abilities to explore the space of parameters. This however comes at the price of a high dependency to the size of the parameter space (Vicente, 2013). For nonconvex optimization problems in $\mathbb{R}^n$, the complexity of DS methods is of the order $\mathcal{O}(n^2)$ (Dodangeh et al.,

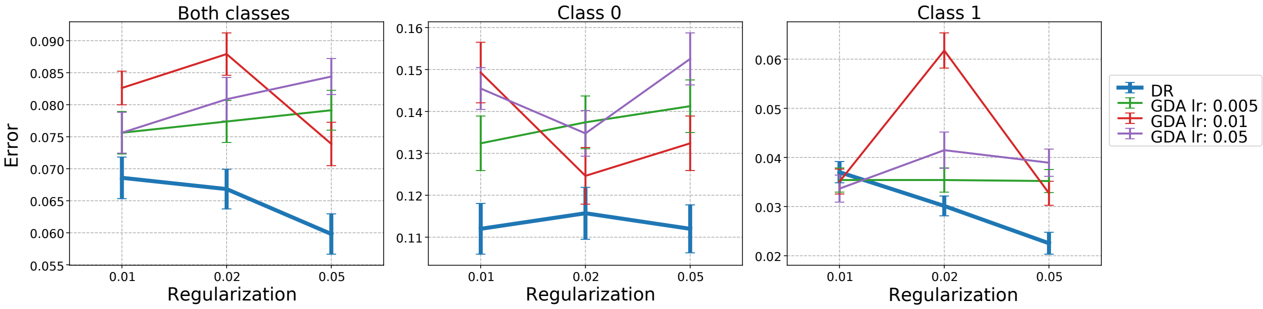

Figure 1: Zero-one loss for each method across classes. The term "lr" stands for different learning rates used. Error bars correspond to 20% of the standard deviation across 10-fold cross validation.

2016). However, recent works by Gratton et al. (2015); Bergou et al. (2018) have shown that replacing the sampling procedure from a PSS by one that correlates with the gradient direction probabilistically, it is possible to achieve a dependence of the order $\mathcal{O}(n)$. The sequential aspect of our method allows us to adopt this probabilistic perspective for the experiments to follow, thus lowering the computation cost.

6.1 Robust Optimization

Robustly-regularized estimators have been successfully used in prior work (Namkoong and Duchi, 2017) to deal with situations in which the empirical risk minimizer is susceptible to high amounts of noise. Formally, the problem of empirical risk minimization can be formulated as follows,

$$\min_{\boldsymbol{\theta}} \sup_{P \in \mathcal{P}} [f(\mathbf{X};\boldsymbol{\theta},\mathcal{P}) = \{\mathbb{E}_{\mathcal{P}}[l(\mathbf{X};\boldsymbol{\theta})] \; : \; D(\mathcal{P} \parallel \hat{\mathcal{P}}_n) \leq \frac{\rho}{n}\}], \tag{10}$$

where $l(\mathbf{X};\boldsymbol{\theta})$ denotes the loss function, $\mathbf{X}$ the data and $D(\mathcal{P} \parallel \hat{\mathcal{P}}_n)$ a distance function that measures the divergence between the true data distribution $\mathcal{P}$ and the empirical data distribution $\mathcal{P}_n$. For the specific case of a binary classification problem, as for instance considered in Adolphs et al. (2018), Eq. (10) can be reformulated as

$$\min_{\boldsymbol{\theta}} \max_{\mathbf{p}} \{ -\sum_{i=1}^{n} p_i [y_i \log(\hat{y}(\mathbf{X}_i;\boldsymbol{\theta})) + (1-y_i) \log(1-\hat{y}(\mathbf{X}_i;\boldsymbol{\theta}))] - \lambda \sum_{i=1}^{n} \left(p_i - \frac{1}{n}\right)^2 \},$$

where y_i and $\hat{y}(\mathbf{X}_i;\boldsymbol{\theta})$ correspond to the true and the predicted class of data point $\mathbf{X}_i$ and $\lambda > 0$ controls the amount of regularization. Note that the aforementioned function is strongly-concave w.r.t $\mathbf{p}$ (i.e. it satisfies our PL assumption) and can thus be solved efficiently. We consider this optimization problem on the Wisconsin breast cancer data set[4], comparing the performance between our proposed direct-search method and GDA, using the same neural network as classifier. The zero-one loss is shown in Fig. 1 which clearly shows that our algorithm can consistently outperform GDA for different choices of regularization parameters.

6.2 Categorical Data

Generative Adversarial Networks (Goodfellow et al., 2014) are formulated as the saddle point problem:

$$\begin{aligned}\min_{\mathbf{x}} \max_{\mathbf{y}} f(\mathbf{x},\mathbf{y}) =& \mathbb{E}_{\boldsymbol{\theta} \sim p_{data}} [\log \mathcal{D}_{\mathbf{y}}(\boldsymbol{\theta})] \\ &+ \mathbb{E}_{\mathbf{z} \sim p_{\mathbf{z}}} [\log(1 - \mathcal{D}_{\mathbf{y}}(\mathcal{G}_{\mathbf{x}}(\mathbf{z})))],\end{aligned}$$

where $\mathcal{D}_{\mathbf{y}} : \mathbb{R}^n \to [0,1]$ and $\mathcal{G}_{\mathbf{x}} : \mathbb{R}^m \to \mathbb{R}^n$ are the discriminator and generator networks. Although GANs have been used in a wide variety of applications (Goodfellow, 2016), very few approaches can deal with discrete data. The most severe impeding factor in such settings is the non existence of the gradient due to the non-smooth nature of the objective function. One advantage of direct-search techniques over gradient-based methods is that they can be used in such a context where gradients are not accessible. In some cases, we note that ℓ_2 regularization can be used to increase the smoothness constant of the objective function.We illustrate the performance of our direct-search algorithm on a simple example consisting of correlated categorical data, in Figure 2. For a more detailed discussion and more experimental results we refer the reader to the Appendix.

Scaling direct-search to higher dimensions still remains an active area of research, where recent developments

[4] `https://archive.ics.uci.edu/ml/datasets/Breast+Cancer+Wisconsin+(Diagnostic)`

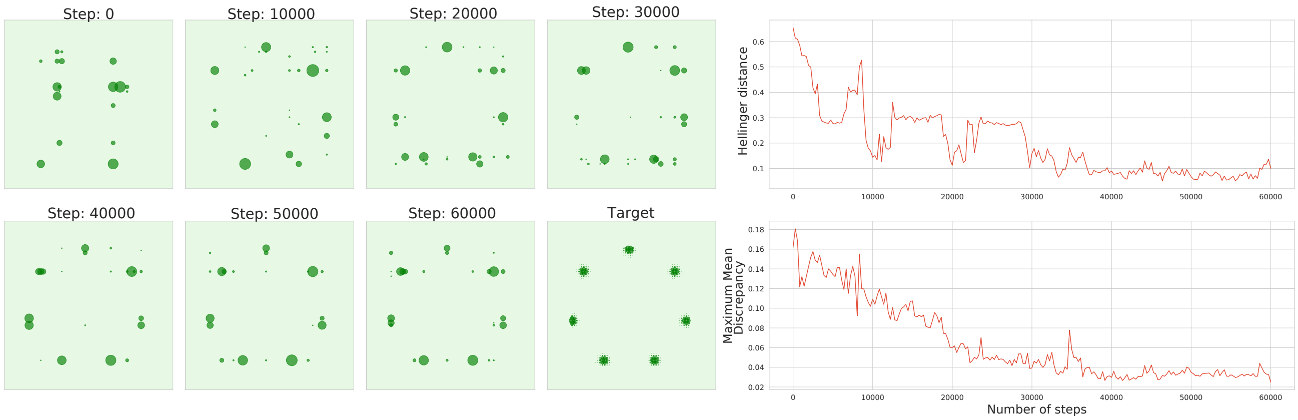

Figure 2: Learning a discretized mixture of Gaussian processes using direct-search methods. Both the Hellinger distance and maximum mean discrepancy decrease as DR learns the modes of the distribution.

include guided search (Maheswaranathan et al., 2018) and projection-based approaches (Wang et al., 2016). In this work, we focus on the theoretical guarantees or our algorithm in the stochastic min-max setting. While we demonstrate a good empirical behavior on relatively small-scale problems, scaling our algorithm to large-scale problems will require further modifications to improve its scalability.

7 CONCLUSION

We presented and proved convergence results for a direct-search method in a stochastic minimization setting for both nonconvex and PL objective functions. We then extended these results to prove convergence for min-max objective functions, where the objective of the max-player satisfies the (PL) condition, while the min-player objective is nonconvex. Our experimental results establish that direct-search can outperform traditionally adopted optimization schemes, while also presenting a promising alternative for categorical settings. A potential direction for future work is to improve the scalability of our algorithm in order to run it on large-scale problems, such as adversarial poisoning attacks on benchmark computer vision datasets. Additional extensions of our work include the use of momentum to accelerate convergence as in Gidel et al. (2018) or developing an optimistic variant of our algorithm as in Daskalakis et al. (2017); Daskalakis and Panageas (2018).

8 Acknowledgements

Sotiris Anagnostidis is supported by by the Onassis Foundation - Scholarship ID: F ZP 002-1/2019-2020.

References

Leonard Adolphs, Hadi Daneshmand, Aurelien Lucchi, and Thomas Hofmann. Local saddle point optimization: A curvature exploitation approach. *arXiv preprint arXiv:1805.05751*, 2018.

Charles Audet and John E Dennis Jr. Mesh adaptive direct search algorithms for constrained optimization. *SIAM Journal on optimization*, 17(1):188–217, 2006.

Charles Audet and Dominique Orban. Finding optimal algorithmic parameters using derivative-free optimization. *SIAM Journal on Optimization*, 17(3):642–664, 2006.

Charles Audet, Kwassi Joseph Dzahini, Michael Kokkolaras, and Sébastien Le Digabel. StoMADS: Stochastic blackbox optimization using probabilistic estimates. *To appear in Computational Optimization and Applications.*, 2021.

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

El Houcine Bergou, Youssef Diouane, Vyacheslav Kungurtsev, and Clément W Royer. A subsampling line-search method with second-order results. *arXiv preprint arXiv:1810.07211*, 2018.

Dimitris Bertsimas and Omid Nohadani. Robust optimization with simulated annealing. *Journal of Global Optimization*, 48(2):323–334, 2010.

Dimitris Bertsimas, Omid Nohadani, and Kwong Meng Teo. Robust optimization for unconstrained simulation-based problems. *Operations research*, 58(1):161–178, 2010.

Jose Blanchet, Coralia Cartis, Matt Menickelly, and Katya Scheinberg. Convergence rate analysis of a stochastic trust-region method via supermartingales. *INFORMS journal on optimization*, 1(2):92–119, 2019.

Ilija Bogunovic, Jonathan Scarlett, Stefanie Jegelka, and Volkan Cevher. Adversarially robust optimization with gaussian processes. In *Advances in neural information processing systems*, pages 5760–5770, 2018.

Ruobing Chen, Matt Menickelly, and Katya Scheinberg. Stochastic optimization using a trust-region method and random models. *Mathematical Programming*, 169(2): 447–487, 2018.

Yunmei Chen, Guanghui Lan, and Yuyuan Ouyang. Optimal primal-dual methods for a class of saddle point problems. *SIAM Journal on Optimization*, 24(4):1779–1814, 2014.

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.

Andrew R Conn, Katya Scheinberg, and Luis Nunes Vicente. *Introduction to derivative-free optimization*, volume 8. Siam, 2009.

Edoardo Conti, Vashisht Madhavan, Felipe Petroski Such, Joel Lehman, Kenneth O Stanley, and Jeff Clune. Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents. *arXiv preprint arXiv:1712.06560*, 2017.

Ana Luisa Custódio, Youssef Diouane, Rohollah Garmanjani, and Elisa Riccietti. Worst-case complexity bounds of directional direct-search methods for multiobjective optimization. *Journal of Optimization Theory and Applications*, 188(1):73–93, 2021.

Constantinos Daskalakis and Ioannis Panageas. The limit points of (optimistic) gradient descent in min-max optimization. In *Advances in Neural Information Processing Systems*, pages 9236–9246, 2018.

Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training gans with optimism. *arXiv preprint arXiv:1711.00141*, 2017.

Mahdi Dodangeh, Luís Nunes Vicente, and Zaikun Zhang. On the optimal order of worst case complexity of direct search. *Optimization Letters*, 10(4):699–708, 2016.

Kwassi Joseph Dzahini. Expected complexity analysis of stochastic direct-search. *arXiv preprint arXiv:2003.03066*, 2020.

Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. *arXiv preprint cs/0408007*, 2004.

Gauthier Gidel, Reyhane Askari Hemmat, Mohammad Pezeshki, Remi Lepriol, Gabriel Huang, Simon Lacoste-Julien, and Ioannis Mitliagkas. Negative momentum for improved game dynamics. *arXiv preprint arXiv:1807.04740*, 2018.

Ian Goodfellow. Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

Serge Gratton, Clément W Royer, Luís Nunes Vicente, and Zaikun Zhang. Direct search based on probabilistic descent. *SIAM Journal on Optimization*, 25(3):1515–1541, 2015.

Serge Gratton, Clément W Royer, and Luís Nunes Vicente. A second-order globally convergent direct-search method and its worst-case complexity. *Optimization*, 65(6):1105–1128, 2016.

Paulina Grnarova, Kfir Y Levy, Aurelien Lucchi, Thomas Hofmann, and Andreas Krause. An online learning approach to generative adversarial networks. *arXiv preprint arXiv:1706.03269*, 2017.

Nikolaus Hansen, Sibylle D Müller, and Petros Koumoutsakos. Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (cma-es). *Evolutionary computation*, 11(1):1–18, 2003.

Warren Hare and Mason Macklem. Derivative-free optimization methods for finite minimax problems. *Optimization Methods and Software*, 28(2):300–312, 2013.

Warren Hare and Julie Nutini. A derivative-free approximate gradient sampling algorithm for finite minimax problems. *Computational Optimization and Applications*, 56(1):1–38, 2013.

Le Thi Khanh Hien, Renbo Zhao, and William B Haskell. An inexact primal-dual smoothing framework for large-scale non-bilinear saddle point problems. *arXiv preprint arXiv:1711.03669*, 2017.

Robert Hooke and Terry A Jeeves. Direct search solution of numerical and statistical problems. *Journal of the ACM (JACM)*, 8(2):212–229, 1961.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Chi Jin, Praneeth Netrapalli, and Michael I Jordan. Minmax optimization: Stable limit points of gradient descent ascent are locally optimal. *arXiv preprint arXiv:1902.00618*, 2019.

Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in neural information processing systems*, pages 315–323, 2013.

Hamed Karimi and Mark Schmidt. Linear convergence of proximal-gradient methods under the polyak-lojasiewicz condition. *arXiv preprint arXiv:1608.04636*, 2015.

Tamara G Kolda, Robert Michael Lewis, and Virginia Torczon. Optimization by direct search: New perspectives on some classical and modern methods. *SIAM review*, 45(3):385–482, 2003.

Jakub Konevcnỳ and Peter Richtárik. Simple complexity analysis of simplified direct search. *arXiv preprint arXiv:1410.0390*, 2014.

Robert Michael Lewis, Virginia Torczon, and Michael W Trosset. Direct search methods: then and now. *Journal of computational and Applied Mathematics*, 124(1-2): 191–207, 2000.

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

Tianyi Lin, Chi Jin, and Michael Jordan. On gradient descent ascent for nonconvex-concave minimax problems. In *International Conference on Machine Learning*, pages 6083–6093. PMLR, 2020.

Sijia Liu, Songtao Lu, Xiangyi Chen, Yao Feng, Kaidi Xu, Abdullah Al-Dujaili, Minyi Hong, and Una-May Obelilly. Min-max optimization without gradients: Convergence and applications to adversarial ml. *arXiv preprint arXiv:1909.13806*, 2019.

Sijia Liu, Songtao Lu, Xiangyi Chen, Yao Feng, Kaidi Xu, Abdullah Al-Dujaili, Mingyi Hong, and Una-May O'Reilly. Min-max optimization without gradients: Convergence and applications to black-box evasion and poisoning attacks. In *International Conference on Machine Learning*, pages 6282–6293. PMLR, 2020.

Niru Maheswaranathan, Luke Metz, George Tucker, Dami Choi, and Jascha Sohl-Dickstein. Guided evolutionary strategies: Augmenting random search with surrogate gradients. *arXiv preprint arXiv:1806.10230*, 2018.

Julien Marzat, Hélène Piet-Lahanier, and Eric Walter. Min-max hyperparameter tuning, with application to fault detection. *IFAC Proceedings Volumes*, 44(1): 12904–12909, 2011.

Matt Menickelly and Stefan M Wild. Derivative-free robust optimization by outer approximations. *Mathematical Programming*, 179(1-2):157–193, 2020.

Hongseok Namkoong and John C Duchi. Variance-based regularization with convex objectives. In *Advances in neural information processing systems*, pages 2971–2980, 2017.

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

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

John von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295–320, 1928.

Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. Solving a class of non-convex min-max games using iterative first order methods. In *Advances in Neural Information Processing Systems*, pages 14905–14916, 2019.

Shayegan Omidshafiei, Jason Pazis, Christopher Amato, Jonathan P How, and John Vian. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 2681–2690. JMLR. org, 2017.

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

Jong-Shi Pang and Meisam Razaviyayn. A unified distributed algorithm for non-cooperative games., 2016.

Courtney Paquette and Katya Scheinberg. A stochastic line search method with convergence rate analysis. *arXiv preprint arXiv:1807.07994*, 2018.

LA Rastrigin. The convergence of the random search method in the extremal control of a many parameter system. *Automaton & Remote Control*, 24:1337–1342, 1963.

Luis Miguel Rios and Nikolaos V Sahinidis. Derivative-free optimization: a review of algorithms and comparison of software implementations. *Journal of Global Optimization*, 56(3):1247–1293, 2013.

Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*, 2017.

Maziar Sanjabi, Jimmy Ba, Meisam Razaviyayn, and Jason D Lee. On the convergence and robustness of training gans with regularized optimal transport. In *Advances in Neural Information Processing Systems*, pages 7091–7101, 2018.

J Spall. Stochastic approximation and the finite-difference method. *Introduction to Stochastic Search and Optimization: Estimation, Simulation and Control, number*, pages 150–175, 2003.

Nilesh Tripuraneni, Mitchell Stern, Chi Jin, Jeffrey Regier, and Michael I Jordan. Stochastic cubic regularization for fast nonconvex optimization. In *Advances in neural information processing systems*, pages 2899–2908, 2018.

Joel A Tropp. An introduction to matrix concentration inequalities. *arXiv preprint arXiv:1501.01571*, 2015.

Luís Nunes Vicente. Worst case complexity of direct search. *EURO Journal on Computational Optimization*, 1(1-2): 143–153, 2013.

Zhongruo Wang, Krishnakumar Balasubramanian, Shiqian Ma, and Meisam Razaviyayn. Zeroth-order algorithms for nonconvex minimax problems with improved complexities. *arXiv preprint arXiv:2001.07819*, 2020.

Ziyu Wang, Frank Hutter, Masrour Zoghi, David Matheson, and Nando de Feitas. Bayesian optimization in a billion dimensions via random embeddings. *Journal of Artificial Intelligence Research*, 55:361–387, 2016.

Guojun Zhang and Yaoliang Yu. Convergence of gradient methods on bilinear zero-sum games. *arXiv preprint arXiv:1908.05699*, 2019.

Kaiqing Zhang, Zhuoran Yang, and Tamer Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *arXiv preprint arXiv:1911.10635*, 2019.

Direct-Search for a Class of Stochastic Min-Max Problems: Supplementary Materials

A ALGORITHMS

We present omitted algorithms. Algorithm 3 depicts the updates for the minimization problem. At each outer iteration of Algorithm 2, a single successful step for the minimization problem is performed. In contrast to standard direct-search algorithms, we do not increase the step size parameter immediately after a successful step, but instead, before the start of the next search for a new successful step, to simplify notation for the upcoming proofs.

```
Algorithm 3: ONE-STEP-DIRECT-SEARCH($f, \mathbf{x}_0, \sigma_0$)
Input: $f$: objective function, with $f_k$ it's estimate at step $k$
       $\mathbf{x}$: initial point
       $\sigma_0$: step size value
$c$: forcing function constant
$\gamma > 1$: step size update parameter
Create the Positive Spanning Set $\mathcal{D}$ for the variables $\mathbf{x}$.
Update $\sigma_1 = \min\{\gamma\sigma_0, \sigma_{\max}\}$ as last update was successful.
for $k$ = 1, ... do
    1. Offspring generation:
    Generate the points
        $\mathbf{x}^i = \mathbf{x} + \sigma_k \mathbf{d}^i, \quad \forall \mathbf{d}^i \in \mathcal{D}.$
    2. Parent Selection:
    Choose $\mathbf{x}' = \arg\min_i f_k(\mathbf{x}^i)$.
    3. Sufficient Decrease: if $f_k(\mathbf{x}') < f_k(\mathbf{x}) - \rho(\sigma_k)$ then
        (Iteration is successful)
        return $\mathbf{x}', \sigma_k$.
    else
        (Iteration is unsuccessful)
        Decrease step size $\sigma_{k+1} = \gamma^{-1}\sigma_k$.
    end
end
```

B PROOFS OF SECTION 4

B.1 Proof of Lemma 1

Proof. The result follows by applying Holder's inequality.

$$\mathbb{E}\left[\frac{1_{J_k^c}|F_k^0 - f(\mathbf{X}_k)|}{l_f \Sigma_k^2} \mid \mathcal{F}_{k-1}\right] \leq \left(\mathbb{E}[1_{J_k^c}|\mathcal{F}_{k-1}]\right)^{1/2} \left(\mathbb{E}\left[\frac{|F_k^0 - f(\mathbf{X}_k)|^2}{l_f^2 \Sigma_k^4}\right]\right)^{1/2}.$$

By Assumption 1, it holds that $\left(\mathbb{E}\left[\frac{|F_k^0 - f(\mathbf{x}_k)|^2}{l_f^2 \Sigma_k^4}\right]\right)^{1/2} \leq 1$ and the result follows. Following the same steps, the second inequality of the Lemma holds as well.

□

B.2 Proof of Theorem 2

Proof. We begin by taking separate cases according to if the estimates are accurate or not and if the steps of Algorithm 1 are successful or not. We use 1_{Succ_k} to denote the event that step k is successful.

Case 1: Accurate estimates.

- **Successful step.**

  At a successful step with accurate estimates we have that:

$$\begin{aligned} &1_{\text{Succ}_k} 1_{J_k}(f(\mathbf{X}_{k+1}) - f(\mathbf{X}_k)) \\ &\qquad = 1_{\text{Succ}_k} 1_{J_k}(f(\mathbf{X}_{k+1}) - f_k(\mathbf{X}_{k+1}) + f_k(\mathbf{X}_{k+1}) - f_k(\mathbf{X}_k) + f_k(\mathbf{X}_k) - f(\mathbf{X}_k)) \\ &\qquad \leq 1_{\text{Succ}_k} 1_{J_k}(-(c - 2\epsilon_f)\Sigma_k^2). \end{aligned}$$

  Therefore

$$\begin{aligned} &1_{\text{Succ}_k} 1_{J_k}(\Phi_{k+1} - \Phi_k) \\ &\qquad = 1_{\text{Succ}_k} 1_{J_k}(v(f(\mathbf{X}_{k+1}) - f(\mathbf{X}_k)) + (1-v)\Sigma_{k+1}^2 - (1-v)\Sigma_k^2) \\ &\qquad \leq 1_{\text{Succ}_k} 1_{J_k}(-v(c - 2\epsilon_f)\Sigma_k^2 + (1-v)(\gamma^2 - 1)\Sigma_k^2). \end{aligned}$$

- **Unsuccessful step.**

$$\begin{aligned} 1_{\text{Succ}_k^c} 1_{J_k}(\Phi_{k+1} - \Phi_k) &= 1_{\text{Succ}_k^c} 1_{J_k}((1-v)\Sigma_{k+1}^2 - (1-v)\Sigma_k^2) \\ &= 1_{\text{Succ}_k^c} 1_{J_k}(-(1-v)(1 - \frac{1}{\gamma^2})\Sigma_k^2). \end{aligned}$$

Combining the above results and given that

$$\frac{v}{1-v} \geq \frac{1}{c - 2\epsilon_f}(\gamma^2 - \frac{1}{\gamma^2}) \implies -v(c - 2\epsilon_f) + (1-v)(\gamma^2 - 1) \leq -(1-v)(1 - \frac{1}{\gamma^2}),$$

in the case of accurate estimates we have

$$\mathbb{E}[1_{J_k}(\Phi_{k+1} - \Phi_k) \mid \mathcal{F}_{k-1}] \leq -p_f(1-v)(1 - \frac{1}{\gamma^2})\Sigma_k^2. \tag{11}$$

Case 2: Inaccurate estimates.

- **Successful step.**

$$\begin{aligned} 1_{\text{Succ}_k} 1_{J_k^c}(\Phi_{k+1} - \Phi_k) &= 1_{\text{Succ}_k} 1_{J_k^c}(v(f(\mathbf{X}_{k+1}) - f(\mathbf{X}_k)) + (1-v)\Sigma_{k+1}^2 - (1-v)\Sigma_k^2) \\ &= 1_{\text{Succ}_k} 1_{J_k^c}(v(f(\mathbf{X}_{k+1}) - f_k(\mathbf{X}_{k+1}) + f_k(\mathbf{X}_{k+1}) - f_k(\mathbf{X}_k) \\ &\qquad + f_k(\mathbf{X}_k) - f(\mathbf{X}_k)) + (1-v)\Sigma_{k+1}^2 - (1-v)\Sigma_k^2) \\ &\leq 1_{\text{Succ}_k} 1_{J_k^c}(-vc\Sigma_k^2 + v|f(\mathbf{X}_{k+1}) - f_k(\mathbf{X}_{k+1})| + v|f(\mathbf{X}_k) - f_k(\mathbf{X}_k)| \\ &\qquad - (1-v)(\gamma^2 - 1)\Sigma_k^2), \end{aligned}$$

  where we will later bound terms $|f(\mathbf{X}_{k+1}) - f_k(\mathbf{X}_{k+1})|, |f(\mathbf{X}_k) - f_k(\mathbf{X}_k)|$ using Lemma 1.

- **Unsuccessful step.**

  As before:

$$\begin{aligned}1_{\mathrm{Succ}_k^c}1_{J_k^c}(\Phi_{k+1}-\Phi_k) &= 1_{\mathrm{Succ}_k^c}1_{J_k^c}((1-v)\Sigma_{k+1}^2-(1-v)\Sigma_k^2)\\ &= 1_{\mathrm{Succ}_k^c}1_{J_k^c}(-(1-v)(1-\frac{1}{\gamma^2})\Sigma_k^2).\end{aligned}$$

In total for inaccurate estimates and by using Assumption 1 and Lemma 1

$$\mathbb{E}[1_{J_k^c}(\Phi_{k+1}-\Phi_k) \mid \mathcal{F}_{k-1}] \leq 2v(1-p_f)^{1/2}l_f\Sigma_k^2. \tag{12}$$

Finally, integrating both successful and unsuccessful iterations

$$\begin{aligned}\mathbb{E}[\Phi_{k+1}-\Phi_k \mid \mathcal{F}_{k-1}] &\leq -p_f(1-v)(1-\frac{1}{\gamma^2})\Sigma_k^2+2v(1-p_f)^{1/2}l_f\Sigma_k^2\\ &\leq -p_f(1-v)(1-\frac{1}{\gamma^2})\frac{\Sigma_k^2}{2},\end{aligned}$$

for our requirement of $\frac{p_f}{\sqrt{1-p_f}} \geq \frac{4vl_f}{(1-v)(1-\gamma^{-2})}$.

□

B.3 Proof of Lemma 4

Proof. Similar to Conn et al. (2009), for an unsuccessful step with accurate estimates, we have that for some $\mathbf{d}_k \in \mathcal{D}$

$$\kappa(\mathcal{D})\|\nabla f(\mathbf{x}_k)\|\|\mathbf{d}_k\| \leq -\nabla f(\mathbf{x}_k)^\top \mathbf{d}_k. \tag{13}$$

By the mean value theorem, for some $\eta_k \in [0,1]$,

$$f(\mathbf{x}_k+\sigma_k\mathbf{d}_k)-f(\mathbf{x}_k)=\sigma_k\nabla f(\mathbf{x}_k+\eta_k\sigma_k\mathbf{d}_k)^\top\mathbf{d}_k.$$

Since k is the index of an unsuccessful iteration,

$$f_k(\mathbf{x}_k+\sigma_k\mathbf{d}_k)-f_k(\mathbf{x}_k)+\rho(\sigma_k)\geq 0$$

and since estimates are accurate

$$\begin{aligned}f(\mathbf{x}_k+\sigma_k\mathbf{d}_k)-f(\mathbf{x}_k) &= f(\mathbf{x}_k+\sigma_k\mathbf{d}_k)-f_k(\mathbf{x}_k+\sigma_k\mathbf{d}_k)\\ &\qquad + f_k(\mathbf{x}_k+\sigma_k\mathbf{d}_k)-f_k(\mathbf{x}_k)+f_k(\mathbf{x}_k)-f(\mathbf{x}_k)\\ &\geq -\epsilon_f\sigma_k^2-\rho(\sigma_k)-\epsilon_f\sigma_k^2\\ &= -(c+2\epsilon_f)\sigma_k^2.\end{aligned}$$

Combining the above equations,

$$\begin{aligned}&\sigma_k\nabla f(\mathbf{x}_k+\eta_k\sigma_k\mathbf{d}_k)^\top\mathbf{d}_k+(c+2\epsilon_f)\sigma_k^2\geq 0\\ \Longrightarrow\ &\nabla f(\mathbf{x}_k+\eta_k\sigma_k\mathbf{d}_k)^\top\mathbf{d}_k+(c+2\epsilon_f)\sigma_k\geq 0\\ \Longrightarrow\ &-\nabla f(\mathbf{x}_k)^\top\mathbf{d}_k\leq(\nabla f(\mathbf{x}_k+\eta_k\sigma_k\mathbf{d}_k)-\nabla f(\mathbf{x}_k))^\top\mathbf{d}_k+(c+2\epsilon_f)\sigma_k,\end{aligned} \tag{14}$$

where in the last inequality, we subtracted $\nabla f(\mathbf{x}_k)^\top \mathbf{d}_k$ from both sides.

Finally, Eq. (13) implies

$$\begin{aligned}
&\kappa(\mathcal{D})\|\nabla f(\mathbf{x}_k)\|\|\mathbf{d}_k\| \leq (\nabla f(\mathbf{x}_k + \eta_k\sigma_k\mathbf{d}_k) - \nabla f(\mathbf{x}_k))^\top \mathbf{d}_k + (c + 2\epsilon_f)\sigma_k \\
\implies\ &\kappa(\mathcal{D})\|\nabla f(\mathbf{x}_k)\| \leq \|\nabla f(\mathbf{x}_k + \eta_k\sigma_k\mathbf{d}_k) - \nabla f(\mathbf{x}_k)\| + (c + 2\epsilon_f)\sigma_k \\
&\qquad\qquad\qquad\quad \leq \frac{L}{2}\sigma_k + (c + 2\epsilon_f)\sigma_k.
\end{aligned}$$

□

B.4 Proof of Theorem 5

Proof. By Theorem 2 and Lemma 4.10 from Paquette and Scheinberg (2018) we get that Assumption 2 is satisfied, for $\Sigma_\epsilon = C\epsilon$. Then by an application of Theorem 3 we get

$$\mathbb{E}[T_\epsilon] \leq \frac{p_f}{2p_f - 1}\frac{v(f(\mathbf{X}_0) - f^*) + (1-v)\Sigma_0^2}{p_f(1-v)(1-\frac{1}{\gamma^2})\frac{C^2\epsilon^2}{2}}.$$

The result follows.

□

B.5 Proof of Theorem 6

We will also use the additional result holding for any function with Lipschitz-continuous gradients.

Lemma 8. *Let $f : \mathbf{x} \in \mathbb{R}^n \to \mathbb{R}$ be a continuous differentiable function with Lipschitz continuous gradient with a constant L and a minimum value achieved for $\mathbf{x}^*$. Then*

$$f(\mathbf{x}) - f(\mathbf{x}^*) \geq \frac{1}{2L}\|\nabla f(\mathbf{x})\|^2. \tag{15}$$

Proof. By smoothness and for $\mathbf{y} = \mathbf{x} - \frac{1}{L}\nabla f(\mathbf{x})$ we have

$$\begin{aligned}
f(\mathbf{x}) - f(\mathbf{x}^*) &\geq f(\mathbf{x}) - f(\mathbf{y}) \\
&\geq \langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle - \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2 \\
&= \frac{1}{L}\|\nabla f(\mathbf{x})\|^2 - \frac{1}{2L}\|\nabla f(\mathbf{x})\|^2 \\
&= \frac{1}{2L}\|\nabla f(\mathbf{x})\|^2.
\end{aligned}$$

□

We can now proceed with the proof of Theorem 6.

Proof. We note that for the conditions on the constants c, v and p_f, requirements of Theorem 2 are also satisfied. We define as $T_i = \inf\{k \geq 0 : f(\mathbf{X}_k) - f^* \leq \frac{f(\mathbf{X}_0) - f^*}{2^i}\}$, with $T_0 = 0$. We will also use the random variable $\Lambda_i = T_i - T_{i-1}$.

We will assume without loss of generality that

$$\Sigma_0^2 \leq \frac{9\gamma^2}{c}(f(\mathbf{X}_0) - f^*) \triangleq A(f(\mathbf{X}_0) - f^*), \tag{16}$$

for $A = \frac{9\gamma^2}{c}$. We apply Theorem 3. Given that $f(\mathbf{X}_{T_{i-1}}) - f^* \le \frac{f(\mathbf{X}_{T_0}) - f^*}{2^{i-1}}$, and that $f(\mathbf{X}_k) - f^* > \frac{f(\mathbf{X}_{T_0}) - f^*}{2^i}$ for $k \in [T_{i-1}, T_i)$ (possibly an empty set), Lemma 4 and the Definition 5, then for step sizes $\Sigma_k^2 \le C^2 \mu \frac{f(\mathbf{X}_0) - f^*}{2^{i-1}}$ and accurate estimates, steps are successful. Then by Theorem 3 for an application of the results from Theorem 2 as before, we have

$$
\begin{aligned}
\mathbb{E}[\Lambda_i \mid \mathcal{F}_{T_{i-1}-1}] &\le \frac{p_f}{2p_f - 1} \frac{v(f(\mathbf{X}_{T_{i-1}}) - f^*) + (1-v)\Sigma_{T_{i-1}}^2}{p_f(1-v)(1-\gamma^{-2})\frac{1}{2}C^2\mu\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}} \\
&= \frac{2}{(2p_f - 1)(1-\gamma^{-2})C^2\mu} \left(\frac{v}{1-v} \frac{f(\mathbf{X}_{T_{i-1}}) - f^*}{\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}} + \frac{\Sigma_{T_{i-1}}^2}{\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}} \right) \\
&\le \frac{2}{(2p_f - 1)(1-\gamma^{-2})C^2\mu} \left(\frac{v}{1-v} \frac{\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}}{\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}} + \frac{\Sigma_{T_{i-1}}^2}{\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}} \right) \\
&= \frac{2}{(2p_f - 1)(1-\gamma^{-2})C^2\mu} \left(\frac{v}{1-v} + \frac{\Sigma_{T_{i-1}}^2}{\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}} \right) \qquad (17)
\end{aligned}
$$

We will further show with induction that $\mathbb{E}[\Sigma_{T_i}^2] \le A\frac{f(\mathbf{X}_0)-f^*}{2^i}$. As a result

$$
\begin{aligned}
\mathbb{E}[\Lambda_i] &\le \frac{2}{(2p_f - 1)(1-\gamma^{-2})C^2\mu} \left(\frac{v}{1-v} + \frac{\mathbb{E}[\Sigma_{T_{i-1}}^2]}{\frac{f(\mathbf{X}_0)-f^*}{2^{i-1}}} \right) \\
&\le \frac{2}{(2p_f - 1)(1-\gamma^{-2})C^2\mu} \left(\frac{v}{1-v} + A \right). \qquad (18)
\end{aligned}
$$

The final complexity will be:

$$
\begin{aligned}
\mathbb{E}[T_{\lceil \log \frac{2L(f(\mathbf{x}_0)-f^*)}{\epsilon^2} \rceil}] &= \mathbb{E}[\Lambda_1 + \Lambda_2 + \cdots + \Lambda_{\lceil \log \frac{2L(f(\mathbf{x}_0)-f^*)}{\epsilon^2} \rceil}] \\
&\le \frac{2}{(2p_f - 1)(1-\gamma^{-2})C^2\mu} \left(\frac{v}{1-v} + A \right) \lceil \log \left(\frac{2L(f(\mathbf{X}_0) - f^*)}{\epsilon^2} \right) \rceil. \qquad (19)
\end{aligned}
$$

Getting $\|\nabla f(\mathbf{x}_0)\|^2 \le 2L(f(\mathbf{x}_0) - f^*)$ from Lemma 8, the result follows.

It remains to show the result that $\mathbb{E}[\Sigma_{T_i}^2] \le A\frac{f(\mathbf{X}_0)-f^*}{2^i}$. By assumption, as aforementioned, it holds for T_0. We then assume that it holds for T_{i-1} and show that it also holds for T_i. For each T_i, the last step $T_i - 1$ was a successful one as the parameter $\mathbf{X}$ was updated to satisfy the goal $f(\mathbf{X}_{T_i}) - f^* \le \frac{f(\mathbf{X}_0)-f^*}{2^i}$. As in Theorem 2 we differentiate between the events of this step being accurate or not.

Since we have a successful step $f_{T_i-1}(\mathbf{X}_{T_i}) - f_{T_i-1}(\mathbf{X}_{T_i-1}) \le -c\Sigma_{T_i-1}^2$. Then

$$
\begin{aligned}
f(\mathbf{X}_{T_i}) - f(\mathbf{X}_{T_i-1}) = f(\mathbf{X}_{T_i}) - f_{T_i-1}(\mathbf{X}_{T_i}) + \\
f_{T_i-1}(\mathbf{X}_{T_i}) - f_{T_i-1}(\mathbf{X}_{T_i-1}) + f_{T_i-1}(\mathbf{X}_{T_i-1}) - f(\mathbf{X}_{T_i-1}) \qquad (20)
\end{aligned}
$$

We denote with p_{Acc} the probability of this last step being accurate. Note that this is not the same as p_f as we are conditioning on a successful step. Then we distinguish the two cases.

- **Accurate estimates.**

  By Assumption 3 we get that

$$
1_{\text{Acc}}(f(\mathbf{X}_{T_i}) - f(\mathbf{X}_{T_i-1})) \le 1_{\text{Acc}}(-(c - 2\epsilon_f)\Sigma_{T_i-1}^2). \qquad (21)
$$

- **Inaccurate.**

  In this case, similarly to the proof of Lemma 1 we get

$$\mathbb{E}[1^c_{\text{Acc}}(f(\mathbf{X}_{T_i}) - f(\mathbf{X}_{T_i-1})) \mid \mathcal{F}_{T_i-2}] \leq -(1-p_{\text{Acc}})c\Sigma^2_{T_i-1} + 2\sqrt{1-p_{\text{Acc}}}l_f\Sigma^2_{T_i-1}. \tag{22}$$

Combining the above cases, we get

$$\begin{aligned}
\mathbb{E}[f(\mathbf{X}_{T_i}) - f(\mathbf{X}_{T_i-1}) \mid \mathcal{F}_{T_i-2}] &\leq -p_{\text{Acc}}(c-2\epsilon_f)\Sigma^2_{T_i-1} - (1-p_{\text{Acc}})c\Sigma^2_{T_i-1} + 2\sqrt{1-p_{\text{Acc}}}l_f\Sigma^2_{T_i-1} \\
&\leq -c\Sigma^2_{T_i-1}(1 - \frac{p_{\text{Acc}}}{2} - \sqrt{\frac{1-p_{\text{Acc}}}{2}}), \text{ for } c > \max\{4\epsilon_f, 2\sqrt{2}l_f\} \\
&\leq -\frac{c}{4}\Sigma^2_{T_i-1} \\
\implies \Sigma^2_{T_i-1} &\leq \frac{4\mathbb{E}[f(\mathbf{X}_{T_i-1}) - f^*|\mathcal{F}_{T_i-2}]}{c}.
\end{aligned} \tag{23}$$

Furthermore, by Theorem 2 we get that

$$\begin{aligned}
\mathbb{E}[\Phi_{T_i-1} \mid \mathcal{F}_{T_{i-1}-1}] &\leq \Phi_{T_{i-1}} \\
\mathbb{E}[v(f(\mathbf{X}_{T_i-1}) - f^*) \mid \mathcal{F}_{T_{i-1}-1}] &\leq v(f(\mathbf{X}_{T_{i-1}}) - f^*) + (1-v)\Sigma^2_{T_{i-1}} \\
\mathbb{E}[f(\mathbf{X}_{T_i-1}) - f^*] &\leq \frac{f(\mathbf{X}_0) - f^*}{2^{i-1}} + \frac{(1-v)}{v}\mathbb{E}[\Sigma^2_{T_{i-1}}] \\
&\leq \frac{f(\mathbf{X}_0) - f^*}{2^{i-1}} + \frac{(1-v)}{v}A\frac{f(\mathbf{X}_0) - f^*}{2^{i-1}} \\
&\leq \frac{f(\mathbf{X}_0) - f^*}{2^i}2(1 + \frac{1-v}{v}A).
\end{aligned} \tag{24}$$

By combining (23) and (24) and using the law of iterated expectation we have that

$$\begin{aligned}
\mathbb{E}[\Sigma^2_{T_i}] &= \mathbb{E}[\gamma^2\Sigma^2_{T_i-1}] \\
&\leq \frac{f(\mathbf{X}_0) - f^*}{2^i}\frac{8\gamma^2}{c}(1 + \frac{1-v}{v}A) \\
&\leq A\frac{f(\mathbf{X}_0) - f^*}{2^i},
\end{aligned} \tag{25}$$

for $\frac{v}{1-v} \geq \frac{72\gamma^2}{c}$ and $A = \frac{9\gamma^2}{c}$. The proof is complete.

□

C PROOFS OF SECTION 5

We first present some additional results, required for our proof.

From Karimi and Schmidt (2015), for a function that satisfies the PL condition, it additionally satisfies the Quadratic Growth (GQ) condition.

Lemma 9. *A differentiable function f that satisfies the PL condition with parameter μ, also satisfies the QG condition with parameter 4μ:*

$$f(\mathbf{x}) - f^* \geq 2\mu\|\mathbf{x}^* - \mathbf{x}\|^2,$$

where $\mathbf{x}^*$ belongs to the solution set $\mathcal{X}^*$.

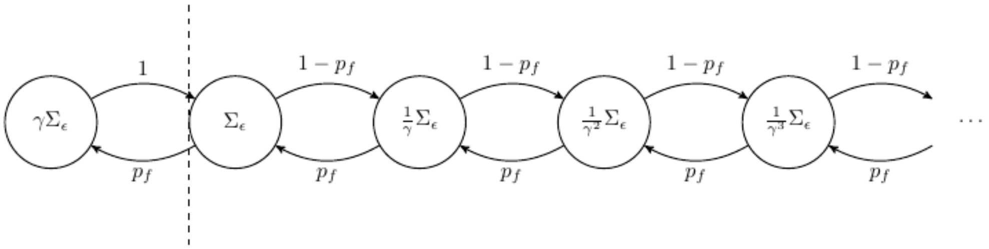
1
$1-p_f$
$1-p_f$
$1-p_f$
$1-p_f$
$\gamma\Sigma_\epsilon$
Σ_ϵ
$\frac{1}{\gamma}\Sigma_\epsilon$
$\frac{1}{\gamma^2}\Sigma_\epsilon$
$\frac{1}{\gamma^3}\Sigma_\epsilon$
$\cdots$
p_f
p_f
p_f
p_f
p_f

Figure 3: Worst case scenario for step sizes. Ignoring steps for $\Sigma > \Sigma_\epsilon$, it corresponds to a biased reflected random walk. The dotted line indices the barrier at position 0, indicating a step size of Σ_ϵ.

Based on the previous Lemma, we can easily prove the following result.

Lemma 10. *Let a differentiable μ-PL function f and also $\mathbf{x}^* \in \arg\min_{\mathbf{x}} f(\mathbf{x})$. If we know that $\|\nabla f(\mathbf{x})\| \leq \epsilon$ then:*

$$\|\mathbf{x} - \mathbf{x}^*\| \leq \frac{1}{2\mu}\epsilon.$$

Proof. By Lemma 9 and the definition of the PL condition we have that:

$$\|\mathbf{x} - \mathbf{x}^*\| \leq \sqrt{\frac{1}{2\mu}(f(\mathbf{x}) - f^*)} \leq \frac{1}{2\mu}\|\nabla f(\mathbf{x})\| \leq \frac{1}{2\mu}\epsilon.$$

□

Lemma 11. *(Lemma A.3 from Nouiehed et al. (2019)) Assume that $-f(\mathbf{x}, \mathbf{y})$ for a specific $\mathbf{x}$, is a class of μ-PL functions in $\mathbf{y}$. Define the set of optimal solutions $\mathbb{Y}(\mathbf{x}) = \arg\max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$. Then for every $\mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}$ and $\mathbf{y}_1^* \in \mathbb{Y}(\mathbf{x}_1), \mathbf{y}_2^* \in \mathbb{Y}(\mathbf{x}_2)$ it holds that:*

$$\|\mathbf{y}_1^* - \mathbf{y}_2^*\| \leq L_{xy}\|\mathbf{x}_1 - \mathbf{x}_2\|,$$

where we denote with $L_{xy} = \frac{L_{12}}{2\mu}$.

Next, we will need to establish a lower bound on the step size Σ. In the deterministic case, Lemma 4 establishes such a lower bound for unsuccessful steps, guaranteeing that if $\Sigma = \Sigma_\epsilon$, then $\|\nabla f(\mathbf{x})\| \leq \epsilon$. However, in the stochastic case, inaccurate steps may occur. We want to ensure a lower bound on the step size parameter with high probability.

To do so, we will consider the worst-case scenario where step sizes get as small as possible. This corresponds to the case where for all step sizes $\Sigma > \Sigma_\epsilon$, unsuccessful steps occur. So do all of the inaccurate estimates, with probability $1 - p_f$. For convenience, we will ignore steps above the value Σ_ϵ since we only require a bound. This corresponds to a random walk with a reflection barrier at position 0 (which corresponds to the step size Σ_ϵ) and an increment probability $1 - p_f$, where p_f is the probability of accurate estimates. We, therefore, use the following Lemma to get a probabilistic lower bound on the step sizes.

Lemma 12. *Let a random walk starting at position 0, with a reflection barrier at position 0 and a transition probability matrix*

$$\begin{bmatrix} p_f & 1-p_f & & & \\ p_f & 0 & 1-p_f & & \\ & p_f & 0 & 1-p_f & \\ & & \ddots & \ddots & \ddots \end{bmatrix}$$

for $p_f > \frac{1}{2}$. Then for $k \geq \frac{\log(1 - e^{\frac{1}{n}\log(\delta)})}{\log\left(\frac{1-p_f}{p_f}\right)} - 1$, the random walk of length n, stays confined within the space $[0, k]$ with a probability at least $\delta > 0$.

Proof. Let a random walk $S_n = \max\{S_{n-1} + X_n, 0\}$, with $S_0 = 0$ and $\mathsf{P}(X_n = 1) = 1 - p_f$, $\mathsf{P}(X_n = -1) = p_f$, for $p_f > \frac{1}{2}$. The probability that the random walk stays until position k, $\mathsf{P}(S_i \leq k, \forall i \leq n)$, is bounded below by the probability of n randomly chosen points from the stationary distribution to be at positions lower or equal to k.

Let us denote with $p_{i,n}$ the probability that the random walk is at position i after n total steps. We first prove by induction that

$$p_{i,n} \geq \frac{p_f}{1-p_f} p_{i+1,n}. \tag{26}$$

It obviously holds for $n = 0$, as $p_{0,0} = 1$ and $p_{i,0} = 0$, $\forall i \geq 1$. Assume that it holds for n. As shown in Fig. 3, with probability $(1 - p_f)$, position i is incremented, therefore for $i \geq 1$

$$\begin{aligned} p_{i,n+1} &= p_{i-1,n}(1-p_f) + p_{i+1,n}p_f \\ &\geq \frac{p_f}{1-p_f} p_{i,n}(1-p_f) + \frac{p_f}{1-p_f} p_{i+2,n}p_f, \text{ by induction} \\ &= \frac{p_f}{1-p_f}(p_{i,n}(1-p_f) + p_{i+2,n}p_f) \\ &= \frac{p_f}{1-p_f} p_{i+1,n+1} \end{aligned}$$

and for $i = 0$

$$\begin{aligned} p_{0,n+1} &= p_{0,n}p_f + p_{1,n}p_f \\ &\geq p_{0,n}p_f + \frac{p_f}{1-p_f} p_{2,n}p_f, \text{ by induction} \\ &= \frac{p_f}{1-p_f}(p_{0,n}(1-p_f) + p_{2,n}p_f) \\ &= \frac{p_f}{1-p_f} p_{1,n+1}. \end{aligned}$$

Let us now consider the probability that the random walk resides in the first k positions. Then:

$$\begin{aligned} \sum_{i=0}^{k} p_{i,n+1} &= p_{0,n}p_f + p_{1,n}p_f + \sum_{i=1}^{k}(p_{i-1,n}(1-p_f) + p_{i+1,n}p_f) \\ &= \sum_{i=0}^{k} p_{i,n} - p_{k,n}(1-p_f) + p_{k+1,n}p_f \\ &\leq \sum_{i=0}^{k} p_{i,n}, \text{ by (26)}, \end{aligned}$$

where the equality in the second line is due to the terms telescoping in the sum in the first line.

As a result, we can lower bound the probability $\sum_{i=0}^{k} p_{i,n}$ with the corresponding one for $n \to \infty$, which corresponds to a stationary distribution. Also

$$\begin{aligned} \mathsf{P}(S_i \leq k \mid S_{i-1} \leq k) &= \mathsf{P}(S_i \leq k \mid S_{i-1} = k)\, \mathsf{P}(S_{i-1} = k) + \mathsf{P}(S_i \leq k \mid S_{i-1} < k)\, \mathsf{P}(S_{i-1} < k) \\ &= p_f\, \mathsf{P}(S_{i-1} = k) + \mathsf{P}(S_{i-1} < k) \geq p_f \end{aligned}$$

and

$$\begin{aligned}
\mathsf{P}(S_i \le k \mid S_{i-1} > k) &= \mathsf{P}(S_i \le k \mid S_{i-1} = k+1)\,\mathsf{P}(S_{i-1} = k+1) \\
&\qquad + \mathsf{P}(S_i \le k \mid S_{i-1} > k+1)\,\mathsf{P}(S_{i-1} > k+1) \\
&= p_f\,\mathsf{P}(S_{i-1} = k+1) + 0\,\mathsf{P}(S_{i-1} > k+1) \\
&\le p_f \ \le \mathsf{P}(S_i \le k \mid S_{i-1} \le k).
\end{aligned}$$

As a result

$$\begin{aligned}
\mathsf{P}(S_i \le k) = \mathsf{P}(S_i \le k \mid S_{i-1} \le k)\,\mathsf{P}(S_{i-1} \le k) + \mathsf{P}(S_i \le k \mid S_{i-1} > k)\,\mathsf{P}(S_{i-1} > k) \\
\le \mathsf{P}(S_i \le k | S_{i-1} \le k).
\end{aligned} \tag{27}$$

The probability of a random walk of length n to stay between the first $k \ge 0$ positions is thus

$$\begin{aligned}
\mathsf{P}(S_i \le k, \forall i \le n) &= \mathsf{P}(S_0 \le k) \prod_{i=1}^{n} \mathsf{P}(S_i \le k \mid S_j \le k, \forall j \in [0, i-1]) \\
&= \prod_{i=1}^{n} \mathsf{P}(S_i \le k \mid S_{i-1} \le k), \text{ with } \mathsf{P}(S_0 \le k) = 1, \forall k \ge 0 \\
&\ge \prod_{i=1}^{n} \mathsf{P}(S_i \le k), \text{ by Eq. (27)} \\
&= \prod_{j=1}^{n} \left(\sum_{i=0}^{k} p_{i,j} \right) \\
&\ge \left(\sum_{i=0}^{k} \pi_i \right)^n ,
\end{aligned}$$

where π_i denotes the stationary probability of the random walk for $i \in \mathbb{N}$. From the recursive relation, we get $\pi_i = (\frac{p_f}{1-p_f})\pi_{i+1}$, which means $\pi_i = (\frac{1-p_f}{p_f})^i \pi_0$.

We now calculate the probability $\pi_{\le k}$ of a randomly chosen point to be part of the first k positions for the stationary distribution

$$\pi_{\le k} = \sum_{i=0}^{k} \pi_i = \frac{\pi_0 \sum_{i=0}^{k} (\frac{1-p_f}{p_f})^i}{\pi_0 \sum_{i=0}^{\infty} (\frac{1-p_f}{p_f})^i} = 1 - \left(\frac{1-p_f}{p_f} \right)^{k+1}.$$

Thus the required probability must be lower bounded by

$$\begin{aligned}
\pi_{\le k}^n \ge \delta \implies \left(1 - \left(\frac{1-p_f}{p_f} \right)^{k+1} \right)^n &\ge \delta \\
\log \left(1 - \left(\frac{1-p_f}{p_f} \right)^{k+1} \right) &\ge \frac{1}{n} \log(\delta) \\
\left(\frac{1-p_f}{p_f} \right)^{k+1} &\le 1 - e^{\frac{1}{n} \log(\delta)} \\
k &\ge \frac{\log \left(1 - e^{\frac{1}{n} \log(\delta)} \right)}{\log \left(\frac{1-p_f}{p_f} \right)} - 1,
\end{aligned} \tag{28}$$

where for the last step we used the fact that $\log \left(\frac{1-p_f}{p_f} \right) < 0$ since $p_f > \frac{1}{2}$ implies $\frac{1-p_f}{p_f} < 1$. □

We can now move to the proof of Theorem 7.

Proof. We denote with c_x the constant used for the sufficient decrease condition of the min problem. We first prove the deterministic case. In the deterministic case, Theorems 5 and 6 hold deterministically, meaning that we can reduce the norm of the gradient below a threshold ϵ for the nonconvex case in $\mathcal{O}(\epsilon^{-2})$ iterations and for the case that the function satisfies our PL condition in $\mathcal{O}(\log(\epsilon^{-1}))$ iterations.

At each step, the max problem is solved almost exactly, which is guaranteed by Theorem 6 and Algorithm 1. Then

$$\|\nabla_{\mathbf{y}} f(\mathbf{x}_{t-1}, \mathbf{y}_t)\| \leq \epsilon^{\max},$$

for an accuracy $\epsilon^{\max}$ to be specified later. In the proof, we will show that for a particular choice of a forcing function constant, the improvement on the minimization problem is better than possible deterioration caused by the updates of the max problem. By Assumption 3 of Lipschitz continuity

$$\begin{aligned} \|\nabla_{\mathbf{y}} f(\mathbf{x}_t, \mathbf{y}_t) - \nabla_{\mathbf{y}} f(\mathbf{x}_{t-1}, \mathbf{y}_t)\| &\leq L_{12}\|\mathbf{x}_t - \mathbf{x}_{t-1}\| = L_{12}\sigma_t \\ \implies \|\nabla_{\mathbf{y}} f(\mathbf{x}_t, \mathbf{y}_t)\| &\leq L_{12}\sigma_t + \epsilon^{\max}, \end{aligned} \tag{29}$$

for a successful update. Here σ_t is used to denote the step size used for the minimization step throughout Algorithms 2 and 3. We note that σ_t always belongs to a successful step, by the notation used in Algorithm 2. Also by triangle inequality we have that (let $\mathbf{y}_t^*$ and $\mathbf{y}_{t+1}^*$ belong to the optimal solution sets at iterations t and $t+1$ respectively)

$$\begin{aligned} \|\mathbf{y}_{t+1} - \mathbf{y}_t\| &= \|\mathbf{y}_{t+1} - \mathbf{y}_{t+1}^* + \mathbf{y}_{t+1}^* - \mathbf{y}_t^* + \mathbf{y}_t^* - \mathbf{y}_t\| \\ &\leq \|\mathbf{y}_{t+1} - \mathbf{y}_{t+1}^*\| + \|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\| + \|\mathbf{y}_t^* - \mathbf{y}_t\|. \end{aligned} \tag{30}$$

By Lemma 11 we have that $\|\mathbf{y}_{t+1}^* - \mathbf{y}_t^*\| \leq L_{xy}\sigma_t$, since $\mathbf{y}_{t+1}^* \in \mathbb{Y}(\mathbf{x}_t)$ and $\mathbf{y}_t^* \in \mathbb{Y}(\mathbf{x}_{t-1})$ (we remind that $\mathbb{Y}(\mathbf{x}) = \arg\max_{\mathbf{y}} f(\mathbf{x}, \mathbf{y})$). Also, as a consequence of Definition 5 and Lemma 10 we have that both

$$\|\mathbf{y}_{t+1} - \mathbf{y}_{t+1}^*\| \leq \frac{\epsilon^{\max}}{2\mu} \quad \text{and} \quad \|\mathbf{y}_t - \mathbf{y}_t^*\| \leq \frac{\epsilon^{\max}}{2\mu}.$$

As a result

$$\|\mathbf{y}_{t+1} - \mathbf{y}_t\| \leq \frac{\epsilon^{\max}}{\mu} + L_{xy}\sigma_t. \tag{31}$$

Finally, for a successful update of the Algorithm 3 we have

$$\begin{aligned} f(\mathbf{x}_t, \mathbf{y}_{t+1}) - f(\mathbf{x}_t, \mathbf{y}_t) &\leq \langle \nabla_{\mathbf{y}} f(\mathbf{x}_t, \mathbf{y}_t), \mathbf{y}_{t+1} - \mathbf{x}_t \rangle + \frac{L_{22}}{2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \\ &\leq (L_{12}\sigma_t + \epsilon^{\max})(L_{xy}\sigma_t + \frac{\epsilon^{\max}}{\mu}) \\ &\quad + \frac{L_{22}}{2}(L_{xy}\sigma_t + \frac{\epsilon^{\max}}{\mu})^2 \\ &= D_1\sigma_t^2 + D_2\sigma_t\epsilon^{\max} + D_3(\epsilon^{\max})^2, \end{aligned} \tag{32}$$

for $D_1 \triangleq L_{12}L_{xy} + \frac{L_{22}}{2}L_{xy}^2$, $D_2 = \frac{L_{12}}{\mu} + L_{xy} + \frac{L_{22}L_{xy}}{\mu}$ and $D_3 = \frac{1}{\mu}(1 + \frac{L_{22}}{2\mu})$. During the updates of the minimization problem we have that

$$\sigma \geq \sigma_{\min} = \sigma_\epsilon, \ \text{ with } \sigma_\epsilon = C\epsilon.$$

Here C, which is defined in Lemma 4, entails the constants for the min problem. We want to ensure that

$$f(\mathbf{x}_t, \mathbf{y}_{t+1}) - f(\mathbf{x}_{t-1}, \mathbf{y}_t) < -K\sigma_t^2, \tag{33}$$

for some $K > 0$ and for $\sigma_t \geq \sigma_{\min}$, to then apply Theorem 2, for f^* the minimum of f at each $\mathbf{y}_t$. Taking also into account our sufficient decrease condition, we want to make sure that the following holds for the polynomial p

$$p(\sigma_t) \triangleq K\sigma_t^2 - c_x\sigma_t^2 + D_1\sigma_t^2 + D_2\sigma_t\epsilon^{\max} + D_3(\epsilon^{\max})^2 \leq 0 \tag{34}$$

for every $\sigma_t \geq \sigma_{\min}$. To establish this we just need to ensure that for the quadratic with negative second degree coefficient (for $c_x > D_1 + K$) the maximum occurs at position:

$$\frac{D_2\epsilon^{\max}}{2(c_x - K - D_1)} \leq C\epsilon \implies \epsilon^{\max} \leq \epsilon\frac{2C(c_x - K - D_1)}{D_2} \tag{35}$$

and also that

$$\begin{aligned} p(C\epsilon) \leq 0 \iff \\ (-c_x + K + D_1)C^2\epsilon^2 + D_2C\epsilon\epsilon^{\max} + D_3(\epsilon^{\max})^2 \leq 0. \end{aligned} \tag{36}$$

For the final condition to hold

$$\epsilon^{\max} \leq \epsilon\frac{C(-D_2 + \sqrt{D_2^2 + 4(c_x - K - D_1)D_3}}{2D_3}. \tag{37}$$

In the stochastic case, we apply Theorems 5 and 6 as is to get the expected number of steps. In this case however, the step size may become smaller than the pre-specified $\sigma_{\min}$ parameter, due to inaccurate estimates. We can then use Lemma 12, to get a bound with high probability, regarding this minimum step size value. More specifically for the number of iterates n specified by Theorem 5, for $k \geq \frac{\log(1-e^{\frac{1}{n}\log(\delta)})}{\log(\frac{1-p_x}{p_x})} - 1$, throughout the updates

$$\sigma \geq \sigma'_{\min} = \frac{1}{\gamma^k}\sigma_{\min},$$

with probability at least $\delta > 0$, where γ is the update parameter for the min problem in Algorithm 3. We then get the similar bounds

$$\epsilon^{\max} \leq \epsilon \min\left\{\frac{2C(c_x - K - D_1)}{\gamma^k D_2}, \frac{C(-D_2 + \sqrt{D_2^2 + 4(c_x - K - D_1)D_3}}{2D_3\gamma^k}\right\}.$$

We note that K acts as a new sufficient decrease constant and should be taken into account for all assumptions of Theorem 5, namely $K > 2\epsilon_x$, which holds for the constant $c_x > D_1 + K > D_1 + 2\epsilon_x$.

□

D EXPERIMENTAL SETUP

D.1 Robust Optimization

The Wisconsin breast cancer data set, is a binary classification task with 569 samples in total, each having 30 attributes. We use a simple neural network with a hidden layer of size 50 and a LeakyReLU activation. This choice of activation accommodates the GDA baseline providing additional gradient information. All networks across methods and folds are initialized with the same weights. For the GDA method we tried a range of different learning rates from the set $\{0.1, 0.05, 0.01, 0.005, 0.001, 0.0005\}$, but only present results for the cases that converged.

In Fig. 4 we present the evolution of the zero-one error across epochs for each method. We stress that one epoch for the GDA approach corresponds to one update each for the max and the min problem, whereas one epoch for DR corresponds to a series of updates for the max problem (at most 10) followed by a single update for the min problem. GDA was run for a total of 10000 epochs and DR for a maximum of 2000 epochs but usually converges a lot faster than that. GDA suffers considerably more by poor initializations compared to DR. In Fig. 4 constant large errors correspond to a constant output of the network for a specific class of the problem (for this unbalanced dataset with rates 0.63 and 0.37).

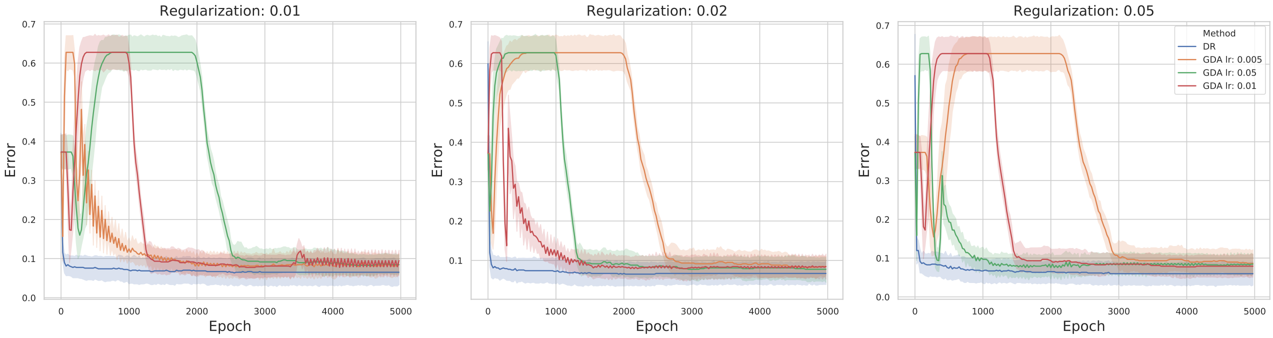

Figure 4: Misclassification error across epochs for each method.

D.2 Toy Examples

Although in the examples following, the objective of the max player is nonconcave and does not satisfy the PL condition, empirical results demonstrate that the proposed algorithm can be successful. We begin by illustrating examples of GANs learning different 2D underlying distributions for a continuous case in Fig. 5. Both the generator and the discriminator have 2 hidden layers of size 20 (64 for learning a mixture of Gaussian in a grid formation) with Tanh activations, while we also use spectral normalization for the discriminator. In all scenarios, we sample the latent code from a lower-dimensional space $N(0, I_2)$, such that it matches the data dimensionality, allowing the generator to learn a simpler mapping (as in Grnarova et al. (2017)).

Motivated by encouraging results, we proceed in a discrete setting, where each of the 2 dimensions of the underlying distributions is parametrized by a categorical variable. The choice of this categorical variable makes the objective function of the generator nondifferentiable. As aforementioned, our algorithm can support multi-categorical data. In the current literature, the most popular methods to deal with this kind of scenario are baselines based on the Gumbel-softmax or the REINFORCE algorithm. Due to their sampling techniques though, dependence on the number of parameters of the model is exponential for these baselines.

We describe shortly how training is performed for each of the baselines used. Based on the output logits o of size n, the result of a projection layer, each method samples a new point y.

Gumbel-softmax Using the Gumbel-max trick, the sampling can be parametrized as

$$y = \text{one-hot}(\arg\max_{1 \leq i \leq n}(o^{(i)} + g^{(i)})), \tag{38}$$

where g are sampled from the *i.i.d.* Gumbel distribution. To enable the calculation of gradients, this is relaxed to the result of a softmax operation

$$\hat{y} = \sigma(\frac{o + g}{T}), \tag{39}$$

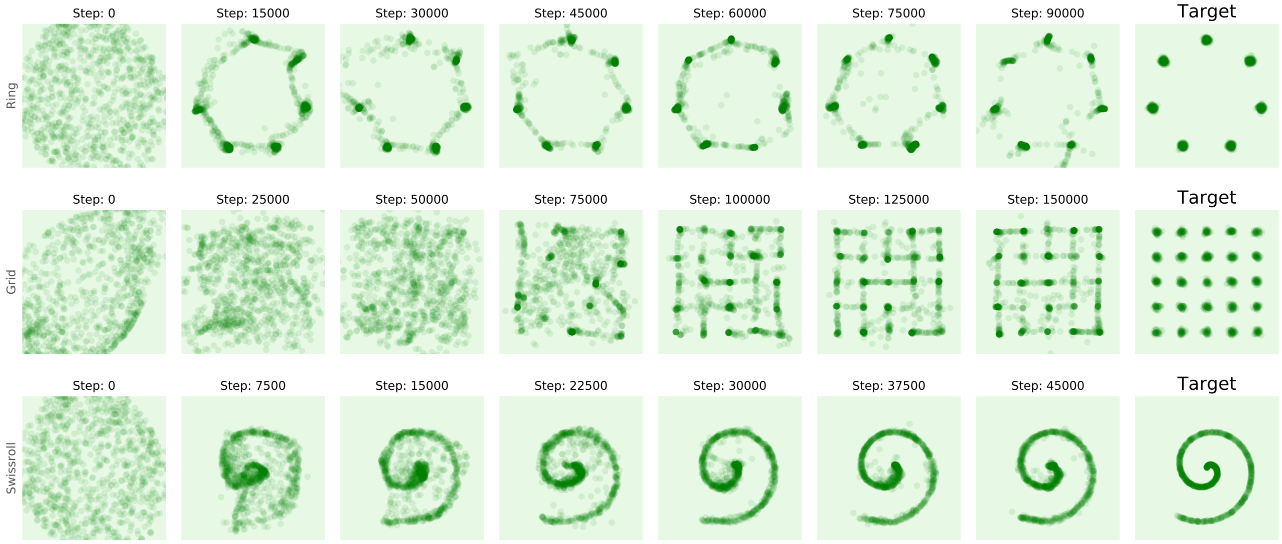

Figure 5: Mode collapse check for direct-search methods for three different problems in the continuous setting.

with $T > 0$, the temperature, controlling the softness of the sampling. A high initial temperature value forces more exploration. As the temperature decreases, $\hat{y}$ becomes a better approximation of y, leading however to steeper gradients and more instabilities. This is also the reason why gradient clipping is crucial for the stability of this method. Untimely updates of the temperature have been known to bolster mode collapse. For all experiments, we use an exponential decay update scheme, decreasing the temperature after a predefined number of steps.

REINFORCE We sample a new point s from the output logits and appoint a specific reward according to the output of the discriminator $r = 2 * (\mathcal{D}(s) - 0.5)$ (we remind that the discriminator uses a sigmoid output), rewarding positively samples that manage to fool the discriminator and negatively those that fail to do so. Subtracting the baseline value of 0.5 helps reduce variance. To alleviate the large variance introduced by the sampling, we increase the number of steps taken by the generator compared to the other methods.

Direct-search Direct-search method just chooses the output with the highest probability

$$y = \text{one-hot}(\arg\max_{1 \leq i \leq n} o^{(i)}). \tag{40}$$

For the discrete toy example illustrated, we draw samples from an evenly weighted and evenly spaced mixture of Gaussians with 7 components in a 2-dimensional space. The 2d sample is then discretized, according to a specific level, leading to 51 possible values for each of the two dimensions of the problem. These categorical datapoints are then transformed to the corresponding continuous one, by a linear mapping, and given as input to the discriminator. This leads to an ordinal relationship between the categorical points (we only display ordinal data for visualization purposes). To monitor the learning curve of the generator compared to the true distribution, we also calculate the Hellinger distance, which is a nonparametric method that calculates the difference of two discrete distributions as

$$\mathcal{H}(\mathcal{P}, \mathcal{Q}) = \frac{1}{\sqrt{2}} \sqrt{\sum_{i=1}^{k} (\sqrt{p_i} - \sqrt{q_i})^2}, \tag{41}$$

as well as the maximum mean discrepancy, comparing the generated samples with the underlying data. Note that this example suffers from stochasticity due to the way samples are collected and batches are created. Results for all baselines are provided in Figure 6.

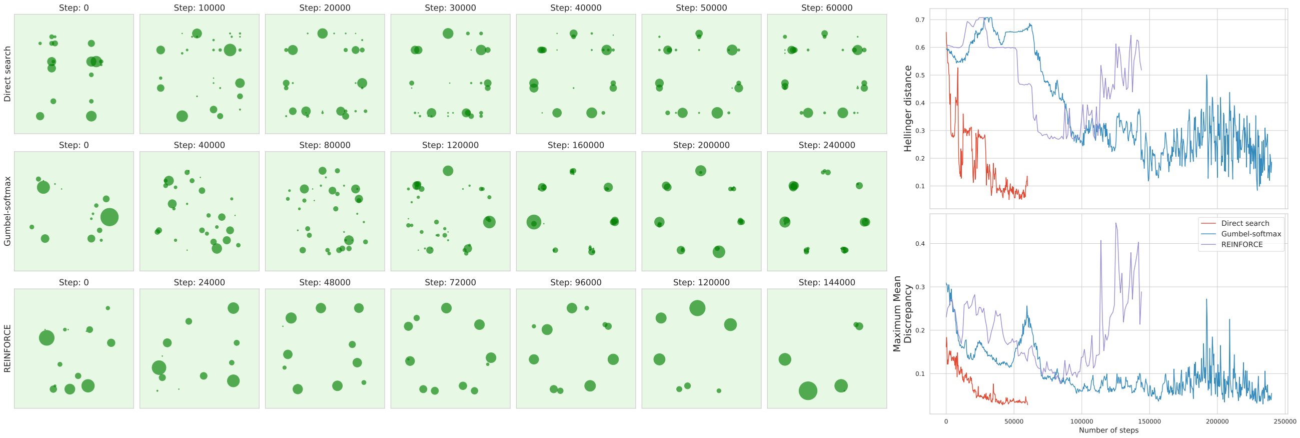

Figure 6: Comparison of direct-search with baselines for the discrete toy example.

When using DR, to overcome the nonsmoothness of the objective function, increasing the ℓ_2 regularization can help with the convergence. Too large of an increase inevitably leads to mode collapse. On the other hand, too small of a regularization parameter requires a large enough step size parameter to enable progress and thus can make learning more difficult. In general, we found that DR converges to similar solutions for a wide range of regularization parameter choices without significant variation in terms of the number of steps required to do so.