
Variance Reduction for Matrix Games

Yair Carmon, Yujia Jin, Aaron Sidford and Kevin Tian
Stanford University
{yairc,yujiajin,sidford,kjtian}@stanford.edu

Abstract

We present a randomized primal-dual algorithm that solves the problem $\min_x \max_y y^\top Ax$ to additive error ϵ in time $\text{nnz}(A) + \sqrt{\text{nnz}(A)n}/\epsilon$, for matrix A with larger dimension n and $\text{nnz}(A)$ nonzero entries. This improves the best known exact gradient methods by a factor of $\sqrt{\text{nnz}(A)/n}$ and is faster than fully stochastic gradient methods in the accurate and/or sparse regime $\epsilon \leq \sqrt{n/\text{nnz}(A)}$. Our results hold for x, y in the simplex (matrix games, linear programming) and for x in an ℓ_2 ball and y in the simplex (perceptron / SVM, minimum enclosing ball). Our algorithm combines the Nemirovski’s “conceptual prox-method” and a novel reduced-variance gradient estimator based on “sampling from the difference” between the current iterate and a reference point.

1 Introduction

Minimax problems—or games—of the form $\min_x \max_y f(x, y)$ are ubiquitous in economics, statistics, optimization and machine learning. In recent years, minimax formulations for neural network training rose to prominence [15, 23], leading to intense interest in algorithms for solving large scale minimax games [10, 14, 20, 9, 18, 24]. However, the algorithmic toolbox for minimax optimization is not as complete as the one for minimization. Variance reduction, a technique for improving stochastic gradient estimators by introducing control variates, stands as a case in point. A multitude of variance reduction schemes exist for finite-sum minimization [cf. 19, 34, 1, 4, 12], and their impact on complexity is well-understood [43]. In contrast, only a few works apply variance reduction to finite-sum minimax problems [3, 39, 5, 26], and the potential gains from variance reduction are not well-understood.

We take a step towards closing this gap by designing variance-reduced minimax game solvers that offer strict runtime improvements over non-stochastic gradient methods, similar to that of optimal variance reduction methods for finite-sum minimization. To achieve this, we focus on the fundamental class of bilinear minimax games,

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} y^\top Ax, \text{ where } A \in \mathbb{R}^{m \times n}.$$

In particular, we study the complexity of finding an ϵ -approximate saddle point (Nash equilibrium), namely x, y with

$$\max_{y' \in \mathcal{Y}} (y')^\top Ax - \min_{x' \in \mathcal{X}} y^\top Ax' \leq \epsilon.$$

In the setting where $\mathcal{X}$ and $\mathcal{Y}$ are both probability simplices, the problem corresponds to finding an approximate (mixed) equilibrium in a matrix game, a central object in game theory and economics. Matrix games are also fundamental to algorithm design due in part to their equivalence to linear programming [8]. Alternatively, when $\mathcal{X}$ is an ℓ_2 ball and $\mathcal{Y}$ is a simplex, solving the corresponding problem finds a maximum-margin linear classifier (hard-margin SVM), a fundamental task in machine learning and statistics [25]. We refer to the former as an ℓ_1 - ℓ_1 game and the latter as an ℓ_2 - ℓ_1 game; our primary focus is to give improved algorithms for these domains.

1.1 Our Approach

Our starting point is Nemirovski’s “conceptual prox-method” [28] for solving $\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y)$, where $f : \mathcal{X} \times \mathcal{Y} \rightarrow \mathbb{R}$ is convex in x and concave in y . The method solves a sequence of subproblems parameterized by $\alpha > 0$, each of the form

$$\text{find } x, y \text{ s.t. } \forall x', y' \quad \langle \nabla_x f(x, y), x - x' \rangle - \langle \nabla_y f(x, y), y - y' \rangle \leq \alpha V_{x_0}(x') + \alpha V_{y_0}(y') \quad (1)$$

for some $(x_0, y_0) \in \mathcal{X} \times \mathcal{Y}$, where $V_a(b)$ is a norm-suitable Bregman divergence from a to b : squared Euclidean distance for ℓ_2 and KL divergence for ℓ_1 . Combining each subproblem solution with an extragradient step, the prox-method solves the original problem to ϵ accuracy by solving $\tilde{O}(\alpha/\epsilon)$ subproblems.¹ (Solving (1) with $\alpha = 0$ is equivalent to solving $\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} f(x, y)$.)

Our first contribution is showing that if a stochastic unbiased gradient estimator $\tilde{g}$ satisfies the “variance” bound

$$\mathbb{E} \|\tilde{g}(x, y) - \nabla f(x_0, y_0)\|_*^2 \leq L^2 \|x - x_0\|^2 + L^2 \|y - y_0\|^2 \quad (2)$$

for some $L > 0$, then $O(L^2/\alpha^2)$ regularized stochastic mirror descent steps using $\tilde{g}$ solve (1) in a suitable probabilistic sense. We call unbiased gradient estimators that satisfy (2) “centered.”

Our second contribution is the construction of “centered” gradient estimators for ℓ_1 - ℓ_1 and ℓ_2 - ℓ_1 bilinear games, where $f(x, y) = y^\top A x$. Our ℓ_1 estimator has the following form. Suppose we wish to estimate $g^\times = A^\top y$ (the gradient of f w.r.t. x), and we already have $g_0^\times = A^\top y_0$. Let $p \in \Delta^m$ be some distribution over $\{1, \dots, m\}$, draw $i \sim p$ and set

$$\tilde{g}^\times = g_0^\times + A_{i\cdot} \frac{[y]_i - [y_0]_i}{p_i},$$

where $A_{i\cdot}$ is the i th column of $A^\top$. This form is familiar from variance reduction techniques [19, 44, 1], that typically use a fixed distribution p . In our setting, however, a fixed p will not produce sufficiently low variance. Departing from prior variance-reduction work and building on [16, 6], we choose p based on y according to

$$p_i(y) = \frac{|[y]_i - [y_0]_i|}{\|y - y_0\|_1},$$

yielding exactly the variance bound we require. We call this technique “sampling from the difference.”

For our ℓ_2 gradient estimator, we sample from the *squared* difference, drawing $\mathcal{X}$ -block coordinate $j \sim q$, where $q_j(x) = ([x]_j - [x_0]_j)^2 / \|x - x_0\|_2^2$. To strengthen our results for ℓ_2 - ℓ_1 games, we consider a refined version of the “centered” criterion (2) which allows regret analysis using local norms [37, 6]. To further facilitate this analysis we follow [6] and introduce gradient clipping. We extend our proofs to show that stochastic regularized mirror descent can solve (1) despite the (distance-bounded) bias caused by gradient clipping.

Our gradient estimators attain the bound (2) with L equal to the Lipschitz constant of ∇f . Specifically,

$$L = \begin{cases} \max_{ij} |A_{ij}| & \text{in the } \ell_1\text{-}\ell_1 \text{ setup} \\ \max_i \|A_{i\cdot}\|_2 & \text{in the } \ell_2\text{-}\ell_1 \text{ setup.} \end{cases} \quad (3)$$

1.2 Method complexity compared with prior art

As per the discussion above, to achieve accuracy ϵ our algorithm solves $\tilde{O}(\alpha/\epsilon)$ subproblems. Each subproblem takes $O(\text{nnz}(A))$ time for computing two exact gradients (one for variance reduction and one for an extragradient step), plus an additional $(m+n)L^2/\alpha^2$ time for the inner mirror descent iterations, with L as in (3). The total runtime is therefore

$$\tilde{O} \left(\left(\text{nnz}(A) + \frac{(m+n)L^2}{\alpha^2} \right) \frac{\alpha}{\epsilon} \right).$$

¹ More precisely, the required number of subproblem solutions is at most $\Theta \cdot \frac{\alpha}{\epsilon}$, where Θ is a “domain size” parameter that depends on $\mathcal{X}$, $\mathcal{Y}$, and the Bregman divergence V (see Section 2). In the ℓ_1 and ℓ_2 settings considered in this paper, we have the bound $\Theta \leq \log(nm)$ and we use the $\tilde{O}$ notation to suppress terms logarithmic in n and m . However, in other settings—e.g., ℓ_∞ - ℓ_1 games [cf. 38, 40]—making the parameter Θ scale logarithmically with the problem dimension is far more difficult.

By setting α optimally to be $\max\{\epsilon, L\sqrt{(m+n)/\text{nnz}(A)}\}$, we obtain the runtime

$$\tilde{O}(\text{nnz}(A) + \sqrt{\text{nnz}(A) \cdot (m+n)} \cdot L \cdot \epsilon^{-1}). \quad (4)$$

Comparison with mirror-prox and dual extrapolation. Nemirovski [28] instantiates his conceptual prox-method by solving the relaxed proximal problem (1) with $\alpha = L$ in time $O(\text{nnz}(A))$, where L is the Lipschitz constant of ∇f , as given in (3). The total complexity of the resulting method is therefore

$$\tilde{O}(\text{nnz}(A) \cdot L \cdot \epsilon^{-1}). \quad (5)$$

The closely related dual extrapolation method of Nesterov [31] attains the same rate of convergence. We refer to the running time (5) as *linear* since it scales linearly with the problem description size $\text{nnz}(A)$. Our running time guarantee (4) is never worse than (5) by more than a constant factor, and improves on (5) when $\text{nnz}(A) = \omega(n+m)$, i.e. whenever A is not extremely sparse. In that regime, our method uses $\alpha \ll L$, hence solving a harder version of (1) than possible for mirror-prox.

Comparison with sublinear-time methods Using a randomized algorithm, Grigoriadis and Khachiyan [16] solve ℓ_1 - ℓ_1 bilinear games in time

$$\tilde{O}((m+n) \cdot L^2 \cdot \epsilon^{-2}), \quad (6)$$

and Clarkson et al. [6] extend this result to ℓ_2 - ℓ_1 bilinear games, with the values of L as in (3). Since these runtimes scale with $n+m \leq \text{nnz}(A)$, we refer to them as *sublinear*. Our guarantee improves on the guarantee (6) when $(m+n) \cdot L^2 \cdot \epsilon^{-2} \gg \text{nnz}(A)$, i.e. whenever (6) is not truly sublinear.

Our method carefully balances linear-time extragradient steps with cheap sublinear-time stochastic gradient steps. Consequently, our runtime guarantee (4) inherits strengths from both the linear and sublinear runtimes. First, our runtime scales linearly with L/ϵ rather than quadratically, as does the linear runtime (5). Second, while our runtime is not strictly sublinear, its component proportional to L/ϵ is $\sqrt{\text{nnz}(A)(n+m)}$, which is sublinear in $\text{nnz}(A)$.

Overall, our method offers the best runtime guarantee in the literature in the regime

$$\frac{\sqrt{\text{nnz}(A)(n+m)}}{\min\{n, m\}^\omega} \ll \frac{\epsilon}{L} \ll \sqrt{\frac{n+m}{\text{nnz}(A)}},$$

where the lower bound on ϵ is due to the best known theoretical runtimes of interior point methods: $\tilde{O}(\max\{n, m\}^\omega \log(L/\epsilon))$ [7] and $\tilde{O}(\text{nnz}(A) + \min\{n, m\}^2) \sqrt{\min\{n, m\} \log(L/\epsilon)}$ [21], where ω is the (current) matrix multiplication exponent.

In the square dense case (i.e. $\text{nnz}(A) \approx n^2 = m^2$), we improve on the accelerated runtime (5) by a factor of $\sqrt{n}$, the same improvement that optimal variance-reduced finite-sum minimization methods achieve over the fast gradient method [44, 1].

1.3 Related work

Matrix games, the canonical form of discrete zero-sum games, have long been studied in economics [32]. The classical mirror descent (i.e. no-regret) method yields an algorithm with running time $\tilde{O}(\text{nnz}(A)L^2\epsilon^{-2})$ [30]. Subsequent work [16, 28, 31, 6] improve this runtime as described above. Our work builds on the extragradient scheme of Nemirovski [28] as well as the gradient estimation and clipping technique of Clarkson et al. [6].

Balamurugan and Bach [3] apply standard variance reduction [19] to bilinear ℓ_2 - ℓ_2 games by sampling elements proportional to squared matrix entries. Using proximal-point acceleration they obtain a runtime of $\tilde{O}(\text{nnz}(A) + \|A\|_F \sqrt{\text{nnz}(A) \max\{m, n\}} \epsilon^{-1} \log \frac{1}{\epsilon})$, a rate we recover using our algorithm (Appendix E). However, in this setting the mirror-prox method has runtime $\tilde{O}(\|A\|_{\text{op}} \text{nnz}(A) \epsilon^{-1})$, which may be better than the result of [3] by a factor of $\sqrt{mn/\text{nnz}(A)}$ due to the discrepancy in the norm of A . Naive application of [3] to ℓ_1 domains results in even greater potential losses. Shi et al. [39] extend the method of [3] to smooth functions using general Bregman divergences, but their extension is unaccelerated and appears limited to a ϵ^{-2} rate.

Chavdarova et al. [5] propose a variance-reduced extragradient method with applications to generative adversarial training. In contrast to our algorithm, which performs extragradient steps in the outer loop,

the method of [5] performs stochastic extragradient steps in the inner loop, using finite-sum variance reduction as in [19]. Chavdarova et al. [5] analyze their method in the convex-concave setting, showing improved stability over direct application of the extragradient method to noisy gradients. However, their complexity guarantees are worse than those of linear-time methods. Following up on [5], Mishchenko et al. [26] propose to reduce the variance of the stochastic extragradient method by using the same stochastic sample for both the gradient and extragradient steps. In the Euclidean strongly convex case, they show a convergence guarantee with a relaxed variance assumption, and in the noiseless full-rank bilinear case they recover the guarantees of [27]. In the general convex case, however, they only show an ϵ^{-2} rate of convergence.

1.4 Paper outline and additional contributions

We define our notation in Section 2. In Section 3.1, we review Nemirovski’s conceptual prox-method and introduce the notion of a relaxed proximal oracle; we implement such oracle using variance-reduced gradient estimators in Section 3.2. In Section 4, we construct these gradient estimators for the ℓ_1 - ℓ_1 and ℓ_2 - ℓ_1 domain settings, and complete the analyses of the corresponding algorithms; in Appendix E we provide analogous treatment for the ℓ_2 - ℓ_2 setting, recovering the results of [3].

In Appendix F we provide three additional contributions: variance-reduction-based computation of proximal points for arbitrary convex-concave functions (Appendix F.1); extension of our results to “composite” saddle point problems of the form $\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} \{f(x, y) + \phi(x) - \psi(y)\}$, where f admits a centered gradient estimator and ϕ, ψ are “simple” convex functions (Appendix F.2); and a number of alternative centered gradient estimators for the ℓ_2 - ℓ_1 and ℓ_2 - ℓ_2 settings (Appendix F.3).

2 Notation

Problem setup. A *setup* is the triplet $(\mathcal{Z}, \|\cdot\|, r)$ where: (i) $\mathcal{Z}$ is a compact and convex subset of $\mathbb{R}^n \times \mathbb{R}^m$, (ii) $\|\cdot\|$ is a norm on $\mathcal{Z}$ and (iii) r is 1-strongly-convex w.r.t. $\mathcal{Z}$ and $\|\cdot\|$, i.e. such that $r(z') \geq r(z) + \langle \nabla r(z), z' - z \rangle + \frac{1}{2} \|z' - z\|^2$ for all $z, z' \in \mathcal{Z}$.² We call r the *distance generating function* and denote the *Bregman divergence* associated with it by

$$V_z(z') := r(z') - r(z) - \langle \nabla r(z), z' - z \rangle \geq \frac{1}{2} \|z' - z\|^2.$$

We also denote $\Theta := \max_{z'} r(z') - \min_z r(z)$ and assume it is finite.

Norms and dual norms. We write $\mathcal{S}^*$ for the set of linear functions on $\mathcal{S}$. For $\zeta \in \mathcal{S}^*$ we define the dual norm of $\|\cdot\|$ as $\|\zeta\|_* := \max_{\|z\| \leq 1} \langle \zeta, z \rangle$. For $p \geq 1$ we write the ℓ_p norm $\|z\|_p = (\sum_i z_i^p)^{1/p}$ with $\|z\|_\infty = \max_i |z_i|$. The dual norm of ℓ_p is ℓ_q with $q^{-1} = 1 - p^{-1}$.

Domain components. We assume $\mathcal{Z}$ is of the form $\mathcal{X} \times \mathcal{Y}$ for convex and compact sets $\mathcal{X} \subset \mathbb{R}^n$ and $\mathcal{Y} \subset \mathbb{R}^m$. Particular sets of interest are the simplex $\Delta^d = \{v \in \mathbb{R}^d \mid \|v\|_1 = 1, v \geq 0\}$ and the Euclidean ball $\mathbb{B}^d = \{v \in \mathbb{R}^d \mid \|v\|_2 \leq 1\}$. For any vector in $z \in \mathbb{R}^n \times \mathbb{R}^m$,

we write z^x and z^y for the first n and last m coordinates of z , respectively.

When totally clear from context, we sometimes refer to the $\mathcal{X}$ and $\mathcal{Y}$ components of z directly as x and y . We write the i th coordinate of vector v as $[v]_i$.

Matrices. We consider a matrix $A \in \mathbb{R}^{m \times n}$ and write $\text{nnz}(A)$ for the number of its nonzero entries. For $i \in [n]$ and $j \in [m]$ we write $A_{i:}$, $A_{:j}$ and A_{ij} for the corresponding row, column and entry, respectively.³ We consider the matrix norms $\|A\|_{\max} := \max_{ij} |A_{ij}|$, $\|A\|_{p \rightarrow q} := \max_{\|x\|_p \leq 1} \|Ax\|_q$ and $\|A\|_F := (\sum_{i,j} A_{ij}^2)^{1/2}$.

² For non-differentiable r , let $\langle \nabla r(z), w \rangle := \sup_{\gamma \in \partial r(z)} \langle \gamma, w \rangle$, where $\partial r(z)$ is the subdifferential of r at z .

³ For $k \in \mathbb{N}$, we let $[k] := \{1, \dots, k\}$.

3 Primal-dual variance reduction framework

In this section, we establish a framework for solving the saddle point problem

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

where f is convex in x and concave y , and admits a (variance-reduced) stochastic estimator for the continuous and monotone⁴ gradient mapping

$$g(z) = g(x, y) := (\nabla_x f(x, y), -\nabla_y f(x, y)).$$

Our goal is to find an ϵ -approximate saddle point (Nash equilibrium), i.e. $z \in \mathcal{Z} := \mathcal{X} \times \mathcal{Y}$ such that

$$\text{Gap}(z) := \max_{y' \in \mathcal{Y}} f(z^x, y') - \min_{x' \in \mathcal{X}} f(x', z^y) \leq \epsilon. \quad (7)$$

We achieve this by generating a sequence $z_1, z_2, \dots, z_K$ such that $\frac{1}{K} \sum_{k=1}^K \langle g(z_k), z_k - u \rangle \leq \epsilon$ for every $u \in \mathcal{Z}$ and using the fact that

$$\text{Gap} \left(\frac{1}{K} \sum_{k=1}^K z_k \right) \leq \max_{u \in \mathcal{Z}} \frac{1}{K} \sum_{k=1}^K \langle g(z_k), z_k - u \rangle \quad (8)$$

due to convexity-concavity of f (see proof in Appendix A.1).

In Section 3.1 we define the notion of a (randomized) *relaxed proximal oracle*, and describe how Nemirovski's mirror-prox method leverages it to solve the problem (3). In Section 3.2 we define a class of *centered* gradient estimators, whose variance is proportional to the squared distance from a reference point. Given such a centered gradient estimator, we show that a regularized stochastic mirror descent scheme constitutes a relaxed proximal oracle. For a technical reason, we limit our oracle guarantee in Section 3.2 to the bilinear case $f(x, y) = y^\top A x$, which suffices for the applications in Section 4. We lift this limitation in Appendix F.1, where we show a different oracle implementation that is valid for general convex-concave f , with only a logarithmic increase in complexity.

3.1 The mirror-prox method with a randomized oracle

Recall that we assume the space $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ is equipped with a norm $\|\cdot\|$ and distance generating function $r : \mathcal{Z} \rightarrow \mathbb{R}$ that is 1-strongly-convex w.r.t. $\|\cdot\|$ and has range Θ . We write the induced Bregman divergence as $V_z(z') = r(z') - r(z) - \langle \nabla r(z), z' - z \rangle$. We use the following fact throughout the paper: by definition, the Bregman divergence satisfies, for any $z, z', u \in \mathcal{Z}$,

$$-\langle \nabla V_z(z'), z' - u \rangle = V_z(u) - V_{z'}(u) - V_z(z'). \quad (9)$$

For any $\alpha > 0$ we define the α -proximal mapping $\text{Prox}_z^\alpha(g)$ to be the solution of the variational inequality corresponding to the strongly monotone operator $g + \alpha \nabla V_z$, i.e. the unique $z_\alpha \in \mathcal{Z}$ such that $\langle g(z_\alpha) + \alpha \nabla V_z(z_\alpha), z_\alpha - u \rangle \leq 0$ for all $u \in \mathcal{Z}$ [cf. 11]. Equivalently (by (9)),

$$\text{Prox}_z^\alpha(g) := \text{the unique } z_\alpha \in \mathcal{Z} \text{ s.t. } \langle g(z_\alpha), z_\alpha - u \rangle \leq \alpha V_z(u) - \alpha V_{z_\alpha}(u) - \alpha V_z(z_\alpha) \quad \forall u \in \mathcal{Z}. \quad (10)$$

When $V_z(z') = V_x^x(x') + V_y^y(y')$, $\text{Prox}_z^\alpha(g)$ is also the unique solution of the saddle point problem

$$\min_{x' \in \mathcal{X}} \max_{y' \in \mathcal{Y}} \{f(x', y') + \alpha V_x^x(x') - \alpha V_y^y(y')\}.$$

Consider iterations of the form $z_k = \text{Prox}_{z_{k-1}}^\alpha(g)$, with $z_0 = \arg \min_z r(z)$. Averaging the definition (10) over k , using the bound (8) and the nonnegativity of Bregman divergences gives

$$\text{Gap} \left(\frac{1}{K} \sum_{k=1}^K z_k \right) \leq \max_{u \in \mathcal{Z}} \frac{1}{K} \sum_{k=1}^K \langle g(z_k), z_k - u \rangle \leq \max_{u \in \mathcal{Z}} \frac{\alpha (V_{z_0}(u) - V_{z_K}(u))}{K} \leq \frac{\alpha \Theta}{K}.$$

Thus, we can find an ϵ -suboptimal point in $K = \alpha \Theta / \epsilon$ exact proximal steps. However, computing $\text{Prox}_z^\alpha(g)$ exactly may be as difficult as solving the original problem. Nemirovski [28] proposes a relaxation of the exact proximal mapping, which we slightly extend to include the possibility of randomization, and formalize in the following.

⁴ A mapping $q : \mathcal{Z} \rightarrow \mathcal{Z}^*$ is monotone if and only if $\langle q(z') - q(z), z' - z \rangle \geq 0$ for all $z, z' \in \mathcal{Z}$; g is monotone due to convexity-concavity of f .

Definition 1 ((α, ε) -relaxed proximal oracle). Let g be a monotone operator and $\alpha, \varepsilon > 0$. An (α, ε) -relaxed proximal oracle for g is a (possibly randomized) mapping $\mathcal{O} : \mathcal{Z} \rightarrow \mathcal{Z}$ such that $z' = \mathcal{O}(z)$ satisfies

$$\mathbb{E} \left[\max_{u \in \mathcal{Z}} \{ \langle g(z'), z' - u \rangle - \alpha V_z(u) \} \right] \leq \varepsilon.$$

Note that $\mathcal{O}(z) = \text{Prox}_z^\alpha(g)$ is an $(\alpha, 0)$ -relaxed proximal oracle. Algorithm 1 describes the “conceptual prox-method” of Nemirovski [28], which recovers the error guarantee of exact proximal iterations. The k th iteration consists of (i) a relaxed proximal oracle call producing $z_{k-1/2} = \mathcal{O}(z_{k-1})$, and (ii) a *linearized* proximal (mirror) step where we replace $z \mapsto g(z)$ with the constant function $z \mapsto g(z_{k-1/2})$, producing $z_k = \text{Prox}_{z_{k-1}}^\alpha(g(z_{k-1/2}))$. We now state the convergence guarantee for the mirror-prox method, first shown in [28] (see Appendix B.1 for a simple proof).

Algorithm 1: OuterLoop($\mathcal{O}$) (Nemirovski [28])

Input: (α, ε) -relaxed proximal oracle $\mathcal{O}(z)$ for gradient mapping g , distance-generating r

Parameters: Number of iterations K

Output: Point $\bar{z}_K$ with $\mathbb{E} \text{Gap}(\bar{z}) \leq \frac{\alpha\Theta}{K} + \varepsilon$

```

1  $z_0 \leftarrow \arg \min_{z \in \mathcal{Z}} r(z)$ 
2 for  $k = 1, \dots, K$  do
3    $z_{k-1/2} \leftarrow \mathcal{O}(z_{k-1})$     ▷ We implement  $\mathcal{O}(z_{k-1})$  by calling InnerLoop( $z_{k-1}, \tilde{g}_{z_{k-1}}, \alpha$ )
4    $z_k \leftarrow \text{Prox}_{z_{k-1}}^\alpha(g(z_{k-1/2})) = \arg \min_{z \in \mathcal{Z}} \{ \langle g(z_{k-1/2}), z \rangle + \alpha V_{z_{k-1}}(z) \}$ 
5 return  $\bar{z}_K = \frac{1}{K} \sum_{k=1}^K z_{k-1/2}$ 

```

Proposition 1 (Mirror prox convergence via oracles). Let $\mathcal{O}$ be an (α, ε) -relaxed proximal oracle with respect to gradient mapping g and distance-generating function r with range at most Θ . Let $z_{1/2}, z_{3/2}, \dots, z_{K-1/2}$ be the iterates of Algorithm 1 and let $\bar{z}_K$ be its output. Then

$$\mathbb{E} \text{Gap}(\bar{z}_K) \leq \mathbb{E} \max_{u \in \mathcal{Z}} \frac{1}{K} \sum_{k=1}^K \langle g(z_{k-1/2}), z_{k-1/2} - u \rangle \leq \frac{\alpha\Theta}{K} + \varepsilon.$$

3.2 Implementation of an $(\alpha, 0)$ -relaxed proximal oracle

We now explain how to use stochastic variance-reduced gradient estimators to design an efficient $(\alpha, 0)$ -relaxed proximal oracle. We begin by introducing the bias and variance properties of the estimators we require.

Definition 2. Let $z_0 \in \mathcal{Z}$ and $L > 0$. A stochastic gradient estimator $\tilde{g}_{z_0} : \mathcal{Z} \rightarrow \mathcal{Z}^*$ is called (z_0, L) -centered for g if for all $z \in \mathcal{Z}$

1. $\mathbb{E} [\tilde{g}_{z_0}(z)] = g(z)$,
2. $\mathbb{E} \|\tilde{g}_{z_0}(z) - g(z_0)\|_*^2 \leq L^2 \|z - z_0\|^2$.

Lemma 1. A (z_0, L) -centered estimator for g satisfies $\mathbb{E} \|\tilde{g}_{z_0}(z) - g(z)\|_*^2 \leq (2L)^2 \|z - z_0\|^2$.

Proof. Writing $\tilde{\delta} = \tilde{g}_{z_0}(z) - g(z_0)$, we have $\mathbb{E} \tilde{\delta} = g(z) - g(z_0)$ by the first centered estimator property. Therefore,

$$\mathbb{E} \|\tilde{g}_{z_0}(z) - g(z)\|_*^2 = \mathbb{E} \|\tilde{\delta} - \mathbb{E} \tilde{\delta}\|_*^2 \stackrel{(i)}{\leq} 2\mathbb{E} \|\tilde{\delta}\|_*^2 + 2\|\mathbb{E} \tilde{\delta}\|_*^2 \stackrel{(ii)}{\leq} 4\mathbb{E} \|\tilde{\delta}\|_*^2 \stackrel{(iii)}{\leq} (2L)^2 \|z - z_0\|^2,$$

where the bounds follow from (i) the triangle inequality, (ii) Jensen’s inequality and (iii) the second centered estimator property. $\square$

Remark 1. A gradient mapping that admits a (z, L) -centered gradient estimator for every $z \in \mathcal{Z}$ is $2L$ -Lipschitz, since by Jensen’s inequality and Lemma 1 we have for all $w \in \mathcal{Z}$

$$\|g(w) - g(z)\|_* = \|\mathbb{E} \tilde{g}_z(w) - g(z)\|_* \leq (\mathbb{E} \|\tilde{g}_z(w) - g(z)\|_*^2)^{1/2} \leq 2L \|w - z\|.$$

Remark 2. Definition 2 bounds the gradient variance using the distance to the reference point. Similar bounds are used in variance reduction for bilinear saddle-point problems with Euclidean norm [3], as well as for finding stationary points in smooth nonconvex finite-sum problems [2, 33, 12, 45]. However, known variance reduction methods for smooth convex finite-sum minimization require stronger bounds [cf. 1, Section 2.1].

With the variance bounds defined, we describe Algorithm 2 which (for the bilinear case) implements a relaxed proximal oracle. The algorithm is stochastic mirror descent with an additional regularization term around the initial point w_0 . Note that we do not perform extragradient steps in this stochastic method. When combined with a centered gradient estimator, the iterates of Algorithm 2 provide the following guarantee, which is one of our key technical contributions.

Algorithm 2: InnerLoop($w_0, \tilde{g}_{w_0}, \alpha$)

Input: Initial $w_0 \in \mathcal{Z}$, gradient estimator $\tilde{g}_{w_0}$, oracle quality $\alpha > 0$

Parameters: Step size η , number of iterations T

Output: Point $\bar{w}_T$ satisfying Definition 1 (for appropriate $\tilde{g}_{w_0}, \eta, T$)

```

1 for  $t = 1, \dots, T$  do
2    $w_t \leftarrow \arg \min_{w \in \mathcal{Z}} \left\{ \langle \tilde{g}_{w_0}(w_{t-1}), w \rangle + \frac{\alpha}{2} V_{w_0}(w) + \frac{1}{\eta} V_{w_{t-1}}(w) \right\}$ 
3 return  $\bar{w}_T = \frac{1}{T} \sum_{t=1}^T w_t$ 

```

Proposition 2. Let $\alpha, L > 0$, let $w_0 \in \mathcal{Z}$ and let $\tilde{g}_{w_0}$ be (w_0, L) -centered for monotone g . Then, for $\eta = \frac{\alpha}{10L^2}$ and $T \geq \frac{4}{\eta\alpha} = \frac{40L^2}{\alpha^2}$, the iterates of Algorithm 2 satisfy

$$\mathbb{E} \max_{u \in \mathcal{Z}} \left[\frac{1}{T} \sum_{t \in [T]} \langle g(w_t), w_t - u \rangle - \alpha V_{w_0}(u) \right] \leq 0. \quad (11)$$

Before discussing the proof of Proposition 2, we state how it implies the relaxed proximal oracle property for the bilinear case.

Corollary 1. Let $A \in \mathbb{R}^{m \times n}$ and let $g(z) = (A^\top z^y, -Az^x)$. Then, in the setting of Proposition 2, $\mathcal{O}(w_0) = \text{InnerLoop}(w_0, \tilde{g}_{w_0}, \alpha)$ is an $(\alpha, 0)$ -relaxed proximal oracle.

Proof. Note that $\langle g(z), w \rangle = -\langle g(w), z \rangle$ for any $z, w \in \mathcal{Z}$ and consequently $\langle g(z), z \rangle = 0$. Therefore, the iterates $w_1, \dots, w_T$ of Algorithm 2 and its output $\bar{w}_T = \frac{1}{T} \sum_{t=1}^T w_t$ satisfy for every $u \in \mathcal{Z}$,

$$\frac{1}{T} \sum_{t \in [T]} \langle g(w_t), w_t - u \rangle = \frac{1}{T} \sum_{t \in [T]} \langle g(u), w_t \rangle = \langle g(u), \bar{w}_T \rangle = \langle g(\bar{w}_T), \bar{w}_T - u \rangle.$$

Substituting into the bound (11) yields the $(\alpha, 0)$ -relaxed proximal oracle property in Definition 1. $\square$

More generally, the proof of Corollary 1 shows that Algorithm 2 implements a relaxed proximal oracle whenever $z \mapsto \langle g(z), z - u \rangle$ is convex for every u . In Appendix F.1 we implement an (α, ε) -relaxed proximal oracle without such an assumption.

The proof of Proposition 2 is a somewhat lengthy application of existing techniques for stochastic mirror descent analysis in conjunction with Definition 2. We give it in full in Appendix B.2 and sketch it briefly here. We view Algorithm 2 as mirror descent with stochastic gradients $\tilde{\delta}_t = \tilde{g}_{w_0}(w_t) - g(w_0)$ and composite term $\langle g(w_0), z \rangle + \frac{\alpha}{2} V_{w_0}(z)$. For any $u \in \mathcal{Z}$, the standard mirror descent analysis (see Lemma 4 in Appendix A.2) bounds the regret $\sum_{t \in [T]} \langle \tilde{g}_{w_0}(w_t) + \frac{\alpha}{2} \nabla V_{w_0}(w_t), w_t - u \rangle$ in terms of the distance to initialization $V_{w_0}(u)$ and the stochastic gradient norms $\|\tilde{\delta}_t\|_*^2$ for $t \in [T]$. Bounding these norms via Definition 2 and rearranging the $\langle \nabla V_{w_0}(w_t), w_t - u \rangle$ terms, we show that $\mathbb{E} \left[\frac{1}{T} \sum_{t \in [T]} \langle g(w_t), w_t - u \rangle - \alpha V_{w_0}(u) \right] \leq 0$ for all $u \in \mathcal{Z}$. To reach our desired result we must swap the order of the expectation and “for all.” We do so using the “ghost iterate” technique due to Nemirovski et al. [29].

4 Application to bilinear saddle point problems

We now construct centered gradient estimators (as per Definition 2) for the linear gradient mapping

$$g(z) = (A^\top z^\mathcal{Y}, -Az^\mathcal{X}) \text{ corresponding to the bilinear saddle point problem } \min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} y^\top A x.$$

Sections 4.1 and 4.2 consider the ℓ_1 - ℓ_1 and ℓ_2 - ℓ_1 settings, respectively; in Appendix E we show how our approach naturally extends to the ℓ_2 - ℓ_2 setting as well. Throughout, we let w_0 denote the “center” (i.e. reference point) of our stochastic gradient estimator and consider a general query point $w \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. We also recall the notation $[v]_i$ for the i th entry of vector v .

4.1 ℓ_1 - ℓ_1 games

Setup. Denoting the d -dimensional simplex by Δ^d , we let $\mathcal{X} = \Delta^n$, $\mathcal{Y} = \Delta^m$ and $\mathcal{Z} = \mathcal{X} \times \mathcal{Y}$. We take $\|\cdot\|$ to be the ℓ_1 norm with conjugate norm $\|\cdot\|_* = \|\cdot\|_\infty$. We take the distance generating function r to be the negative entropy, i.e. $r(z) = \sum_i [z]_i \log [z]_i$. We note that both $\|\cdot\|_1$ and r are separable and in particular separate over the $\mathcal{X}$ and $\mathcal{Y}$ blocks of $\mathcal{Z}$. Finally we set

$$\|A\|_{\max} := \max_{i,j} |A_{ij}|$$

and note that this is the Lipschitz constant of the gradient mapping g under the chosen norm.

Gradient estimator. Given $w_0 = (w_0^\mathcal{X}, w_0^\mathcal{Y})$ and $g(w_0) = (A^\top w_0^\mathcal{Y}, -Aw_0^\mathcal{X})$, we describe the reduced-variance gradient estimator $\tilde{g}_{w_0}(w)$. First, we define the probabilities $p(w) \in \Delta^m$ and $q(w) \in \Delta^n$ according to,

$$p_i(w) := \frac{|[w^\mathcal{Y}]_i - [w_0^\mathcal{Y}]_i|}{\|w^\mathcal{Y} - w_0^\mathcal{Y}\|_1} \text{ and } q_j(w) := \frac{|[w^\mathcal{X}]_j - [w_0^\mathcal{X}]_j|}{\|w^\mathcal{X} - w_0^\mathcal{X}\|_1}. \quad (12)$$

To compute $\tilde{g}_{w_0}$ we sample $i \sim p(w)$ and $j \sim q(w)$ independently, and set

$$\tilde{g}_{w_0}(w) := \left(A^\top w_0^\mathcal{Y} + A_{i:} \frac{[w^\mathcal{Y}]_i - [w_0^\mathcal{Y}]_i}{p_i(w)}, -Aw_0^\mathcal{X} - A_{:j} \frac{[w^\mathcal{X}]_j - [w_0^\mathcal{X}]_j}{q_j(w)} \right), \quad (13)$$

where $A_{i:}$ and $A_{:j}$ are the i th row and j th column of A , respectively. Since the sampling distributions $p(w), q(w)$ are proportional to the absolute value of the difference between blocks of w and w_0 , we call strategy (12) “sampling from the difference.” Substituting (12) into (13) gives the explicit form

$$\tilde{g}_{w_0}(w) = g(w_0) + (A_{i:} \|w^\mathcal{Y} - w_0^\mathcal{Y}\|_1 \text{sign}([w^\mathcal{Y} - w_0^\mathcal{Y}]_i), -A_{:j} \|w^\mathcal{X} - w_0^\mathcal{X}\|_1 \text{sign}([w^\mathcal{X} - w_0^\mathcal{X}]_j)). \quad (14)$$

A straightforward calculation shows that this construction satisfies Definition 2.

Lemma 2. *In the ℓ_1 - ℓ_1 setup, the estimator (14) is (w_0, L) -centered with $L = \|A\|_{\max}$.*

Proof. The first property ($\mathbb{E} \tilde{g}_{w_0}(w) = g(w)$) follows immediately by inspection of (13). The second property follows from (14) by noting that

$$\|\tilde{g}_{w_0}(w) - g(w_0)\|_\infty = \max \{ \|A_{i:}\|_\infty \|w^\mathcal{Y} - w_0^\mathcal{Y}\|_1, \|A_{:j}\|_\infty \|w^\mathcal{X} - w_0^\mathcal{X}\|_1 \} \leq \|A\|_{\max} \|w - w_0\|_1$$

for all i, j , and therefore $\mathbb{E} \|\tilde{g}_{w_0}(w) - g(w_0)\|_\infty^2 \leq \|A\|_{\max}^2 \|w - w_0\|_1^2$. $\square$

The proof of Lemma 2 reveals that the proposed estimator satisfies a stronger version of Definition 2: the last property and also Lemma 1 hold with probability 1 rather than in expectation.

Runtime bound. Combining the centered gradient estimator (13), the relaxed oracle implementation (Algorithm 2) and the extragradient outer loop (Algorithm 1), we obtain our main result for ℓ_1 - ℓ_1 games: an accelerated stochastic variance reduction algorithm. We write the resulting complete method explicitly as Algorithm 3 in Appendix C.1. The algorithm enjoys the following runtime guarantee (see proof in Appendix C.2).

Theorem 1. Let $A \in \mathbb{R}^{m \times n}$, $\epsilon > 0$, and $\alpha \geq \epsilon / \log(nm)$. Algorithm 3 outputs a point $z = (z^x, z^y)$ such that $\mathbb{E} [\max_{y \in \Delta^m} y^\top A z^x - \min_{x \in \Delta^n} (z^y)^\top A x] = \mathbb{E} [\max_i [A z^x]_i - \min_j [A^\top z^y]_j] \leq \epsilon$, and runs in time

$$O \left(\left(\text{nnz}(A) + \frac{(m+n) \|A\|_{\max}^2}{\alpha^2} \right) \frac{\alpha \log(mn)}{\epsilon} \right). \quad (15)$$

Setting α optimally, the running time is

$$O \left(\text{nnz}(A) + \frac{\sqrt{\text{nnz}(A)(m+n)} \|A\|_{\max} \log(mn)}{\epsilon} \right). \quad (16)$$

4.2 ℓ_2 - ℓ_1 games

Setup. We set $\mathcal{X} = \mathbb{B}^n$ to be the n -dimensional Euclidean ball of radius 1, while $\mathcal{Y} = \Delta^m$ remains the simplex. For $z = (z^x, z^y) \in \mathcal{Z} = \mathcal{X} \times \mathcal{Y}$ we define a norm by

$$\|z\|^2 = \|z^x\|_2^2 + \|z^y\|_1^2 \quad \text{with dual norm} \quad \|g\|_*^2 = \|g^x\|_2^2 + \|g^y\|_\infty^2.$$

For distance generating function we take $r(z) = r^x(z^x) + r^y(z^y)$ with $r^x(x) = \frac{1}{2} \|x\|_2^2$ and $r^y(y) = \sum_i y_i \log y_i$; r is 1-strongly convex w.r.t. to $\|\cdot\|$ and has range $\frac{1}{2} + \log m \leq \log(2m)$. Finally, we denote

$$\|A\|_{2 \rightarrow \infty} = \max_{i \in [m]} \|A_{i:}\|_2,$$

and note that this is the Lipschitz constant of g under $\|\cdot\|$.

Gradient estimator. To account for the fact that $\mathcal{X}$ is now the ℓ_2 unit ball, we modify the sampling distribution q in (12) to $q_j(w) = \frac{([w^x]_j - [w_0^x]_j)^2}{\|w^x - w_0^x\|_2^2}$, and keep p the same. As we explain in detail in Appendix D.1.1, substituting these probabilities into the expression (13) yields a centered gradient estimator with a constant $(\sum_{j \in [n]} \|A_{j:}\|_\infty^2)^{1/2}$ that is larger than $\|A\|_{2 \rightarrow \infty}$ by a factor of up to $\sqrt{n}$. Using local norms analysis allows us to tighten these bounds whenever the stochastic steps have bounded infinity norm. Following Clarkson et al. [6], we enforce such bound on the step norms via gradient clipping. The final gradient estimator is

$$\tilde{g}_{w_0}(w) := \left(A^\top w_0^y + A_{i:} \frac{\|w^y - w_0^y\|_1}{\text{sign}([w^y - w_0^y]_i)}, -A w_0^x - \mathsf{T}_\tau \left(A_{j:} \frac{\|w^x - w_0^x\|_2^2}{[w^x]_j - [w_0^x]_j} \right) \right),$$

where $[\mathsf{T}_\tau(v)]_i = \begin{cases} -\tau & [v]_i < -\tau \\ [v]_i & -\tau \leq [v]_i \leq \tau \\ \tau & [v]_i > \tau, \end{cases}$

The clipping operation T_τ introduces bias to the gradient estimator, which we account for by carefully choosing a value of τ for which the bias is on the same order as the variance, and yet the resulting steps are appropriately bounded; see Appendix D.1.2. In Appendix F.3.1 we describe an alternative gradient estimator for which the distribution q does not depend on the current iterate w .

Runtime bound. Algorithm 4 in Appendix D.5 combines our clipped gradient estimator with our general variance reduction framework. The analysis in Appendix D gives the following guarantee.

Theorem 2. Let $A \in \mathbb{R}^{m \times n}$, $\epsilon > 0$, and any $\alpha \geq \epsilon / \log(2m)$. Algorithm 4 outputs a point $z = (z^x, z^y)$ such that $\mathbb{E} [\max_{y \in \Delta^m} y^\top A z^x - \min_{x \in \mathbb{B}^n} (z^y)^\top A x] = \mathbb{E} [\max_i [A z^x]_i + \|A^\top z^y\|_2] \leq \epsilon$, and runs in time

$$O \left(\left(\text{nnz}(A) + \frac{(m+n) \|A\|_{2 \rightarrow \infty}^2}{\alpha^2} \right) \frac{\alpha \log(2m)}{\epsilon} \right). \quad (17)$$

Setting α optimally, the running time is

$$O \left(\text{nnz}(A) + \frac{\sqrt{\text{nnz}(A)(m+n)} \|A\|_{2 \rightarrow \infty} \log(2m)}{\epsilon} \right). \quad (18)$$

Acknowledgments

YC and YJ were supported by Stanford Graduate Fellowships. AS was supported by the NSF CAREER Award CCF-1844855. KT was supported by the NSF Graduate Fellowship DGE1656518.

References

- [1] Z. Allen-Zhu. Katyusha: the first direct acceleration of stochastic gradient methods. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 1200–1205, 2017.
- [2] Z. Allen-Zhu and E. Hazan. Variance reduction for faster non-convex optimization. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 699–707, 2016.
- [3] P. Balamurugan and F. R. Bach. Stochastic variance reduction methods for saddle-point problems. In *Advances in Neural Information Processing Systems*, 2016.
- [4] L. Bottou, F. E. Curtis, and J. Nocedal. Optimization methods for large-scale machine learning. *SIAM Review*, 60(2):223–311, 2018.
- [5] T. Chavdarova, G. Gidel, F. Fleuret, and S. Lacoste-Julien. Reducing noise in GAN training with variance reduced extragradient. In *Advances in Neural Information Processing Systems*, 2019.
- [6] K. L. Clarkson, E. Hazan, and D. P. Woodruff. Sublinear optimization for machine learning. In *51th Annual IEEE Symposium on Foundations of Computer Science*, pages 449–457, 2010.
- [7] M. B. Cohen, Y. T. Lee, and Z. Song. Solving linear programs in the current matrix multiplication time. *arXiv preprint arXiv:1810.07896*, 2018.
- [8] G. B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, Princeton, NJ, 1953.
- [9] C. Daskalakis, A. Ilyas, V. Syrgkanis, and H. Zeng. Training GANs with optimism. In *International Conference on Learning Representations*, 2019.
- [10] Y. Drori, S. Sabach, and M. Teboulle. A simple algorithm for a class of nonsmooth convex-concave saddle-point problems. *Operations Research Letters*, 43(2):209–214, 2015.
- [11] J. Eckstein. Nonlinear proximal point algorithms using Bregman functions, with applications to convex programming. *Mathematics of Operations Research*, 18(1):202–226, 1993.
- [12] C. Fang, C. J. Li, Z. Lin, and T. Zhang. SPIDER: near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *Advances in Neural Information Processing Systems*, 2018.
- [13] R. Frostig, R. Ge, S. Kakade, and A. Sidford. Un-regularizing: approximate proximal point and faster stochastic algorithms for empirical risk minimization. In *Proceedings of the 32nd International Conference on Machine Learning*, pages 2540–2548, 2015.
- [14] G. Gidel, T. Jebara, and S. Lacoste-Julien. Frank-Wolfe algorithms for saddle point problems. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 2017.
- [15] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems*, 2014.
- [16] M. D. Grigoriadis and L. G. Khachiyan. A sublinear-time randomized approximation algorithm for matrix games. *Operation Research Letters*, 18(2):53–58, 1995.
- [17] J. Hiriart-Urruty and C. Lemaréchal. *Convex Analysis and Minimization Algorithms I*. Springer, New York, 1993.

- [18] C. Jin, P. Netrapalli, and M. I. Jordan. Minmax optimization: Stable limit points of gradient descent ascent are locally optimal. *arXiv preprint arXiv:1902.00618*, 2019.
- [19] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*, 2013.
- [20] O. Kolossoski and R. D. Monteiro. An accelerated non-Euclidean hybrid proximal extragradient-type algorithm for convex-concave saddle-point problems. *Optimization Methods and Software*, 32(6):1244–1272, 2017.
- [21] Y. T. Lee and A. Sidford. Efficient inverse maintenance and faster algorithms for linear programming. In *IEEE 56th Annual Symposium on Foundations of Computer Science*, pages 230–249, 2015.
- [22] H. Lin, J. Mairal, and Z. Harchaoui. A universal catalyst for first-order optimization. In *Advances in Neural Information Processing Systems*, 2015.
- [23] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.
- [24] P. Mertikopoulos, H. Zenati, B. Lecouat, C.-S. Foo, V. Chandrasekhar, and G. Piliouras. Mirror descent in saddle-point problems: Going the extra (gradient) mile. In *International Conference on Learning Representations*, 2019.
- [25] M. Minsky and S. Papert. *Perceptrons—an introduction to computational geometry*. MIT Press, 1987.
- [26] K. Mishchenko, D. Kovalev, E. Shulgin, P. Richtárik, and Y. Malitsky. Revisiting stochastic extragradient. *arXiv preprint arXiv:1905.11373*, 2019.
- [27] A. Mokhtari, A. Ozdaglar, and S. Pattathil. A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach. *arXiv preprint arXiv:1901.08511*, 2019.
- [28] A. 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.
- [29] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming. *SIAM Journal on optimization*, 19(4):1574–1609, 2009.
- [30] A. Nemirovsky and D. Yudin. *Problem Complexity and Method Efficiency in Optimization*. J. Wiley & Sons, New York, NY, 1983.
- [31] Y. Nesterov. Dual extrapolation and its applications to solving variational inequalities and related problems. *Mathematical Programming*, 109(2-3):319–344, 2007.
- [32] J. V. Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, 100:295–320, 1928.
- [33] S. J. Reddi, A. Hefny, S. Sra, B. Póczos, and A. Smola. Stochastic variance reduction for nonconvex optimization. In *Proceedings of the 33rd International Conference on Machine Learning*, pages 314–323, 2016.
- [34] M. W. Schmidt, N. L. Roux, and F. R. Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1-2):83–112, 2017.
- [35] S. Shalev-Shwartz and T. Zhang. Stochastic dual coordinate ascent methods for regularized loss. *Journal of Machine Learning Research*, 14(1):567–599, 2013.
- [36] S. Shalev-Shwartz and T. Zhang. Accelerated proximal stochastic dual coordinate ascent for regularized loss minimization. *Mathematical Programming*, 155(1-2):105–145, 2016.
- [37] S. Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2012.

- [38] J. Sherman. Area-convexity, ℓ_∞ regularization, and undirected multicommodity flow. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, pages 452–460. ACM, 2017.
- [39] Z. Shi, X. Zhang, and Y. Yu. Bregman divergence for stochastic variance reduction: Saddle-point and adversarial prediction. In *Advances in Neural Information Processing Systems*, 2017.
- [40] A. Sidford and K. Tian. Coordinate methods for accelerating ℓ_∞ regression and faster approximate maximum flow. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science*, pages 922–933. IEEE, 2018.
- [41] T. Strohmer and R. Vershynin. A randomized Kaczmarz algorithm with exponential convergence. *Journal of Fourier Analysis and Applications*, 15(2):262, 2009.
- [42] M. D. Vose. A linear algorithm for generating random numbers with a given distribution. *IEEE Transactions on software engineering*, 17(9):972–975, 1991.
- [43] B. E. Woodworth and N. Srebro. Tight complexity bounds for optimizing composite objectives. In *Advances in Neural Information Processing Systems*, 2016.
- [44] L. Xiao and T. Zhang. A proximal stochastic gradient method with progressive variance reduction. *SIAM Journal on Optimization*, 24(4):2057–2075, 2014.
- [45] D. Zhou, P. Xu, and Q. Gu. Stochastic nested variance reduced gradient descent for nonconvex optimization. In *Advances in Neural Information Processing Systems*, 2018.