Les Cahiers du GERAD ISSN: 0711–2440

Convergence of gradient methods on bilinear zero-sum games

G. Zhang,
Y. Yu

G–2020–23–EIW06

April 2020

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2020
– Bibliothèque et Archives Canada, 2020

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2020
– Library and Archives Canada, 2020

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Convergence of gradient methods on bilinear zero-sum games

Guojun Zhang
Yaoliang Yu

University of Waterloo & Waterloo AI Institute,
Waterloo (Ontario), Canada, N2L 3G1

Vector Institute MaRS Centre, Toronto (Ontario),
Canada, M5G 1M1

guojun.zhang@uwaterloo.ca

April 2020
Les Cahiers du GERAD
G–2020–23–EIW06

Abstract: *Min-max formulations have attracted great attention in the ML community due to the rise of deep generative models and adversarial methods, while understanding the dynamics of gradient algorithms for solving such formulations has remained a grand challenge. As a first step, we restrict to bilinear zero-sum games and give a systematic analysis of popular gradient updates, for both simultaneous and alternating versions. We provide exact conditions for their convergence and find the optimal parameter setup and convergence rates. In particular, our results offer formal evidence that alternating updates converge "better" than simultaneous ones.*[1]

1 Introduction

Min-max optimization has received significant attention due to the popularity of generative adversarial networks (GANs) [14], adversarial training [19] and reinforcement learning [8], just to name some examples. Formally, given a (bivariate) objective function $f(\mathbf{x}, \mathbf{y})$, we aim to find a *saddle point* $(\mathbf{x}^*, \mathbf{y}^*)$ such that

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

$\forall \mathbf{x} \in \mathbb{R}^{\mathbf{n}}, \forall \mathbf{y} \in \mathbb{R}^{\mathbf{n}}$. Since the beginning of game theory, various algorithms have been proposed for finding saddle points [2, 7, 13, 16, 26, 3, 3, 18, 22, 9]. Due to its recent resurgence in ML, new algorithms designed for training GANs were proposed [5, 15, 11, 20]. However, due to non-convexity in deep learning formulations, our understanding of the convergence behaviour of new and classic gradient algorithms is still limited, and existing analysis mostly focused on bilinear games [5, 11] or strongly-convex-strongly-concave games [17, 21, 29]. Non-zero-sum bilinear games, on the other hand, are PPAD-complete [4] (for the definition see [24]; for finding approximate Nash equilibria, see e.g. [6]).

In this work, we focus on bilinear zero-sum games as a first step towards understanding general min-max optimization, although our results apply to some simple GAN settings [10]. It is well-known that certain gradient algorithms converge at a linear rate on bilinear zero-sum games [17, 21, 26, 16]. These iterative algorithms usually come with two versions: *Jacobi* style or *Gauss–Seidel* (GS) style. In Jacobi style, we update the two sets of parameters (i.e., $\mathbf{x}$ and $\mathbf{y}$) *simultaneously* whereas in GS style we update them *alternatingly* (i.e., one after the other). Thus, Jacobi style updates are naturally amenable to parallelization while GS style updates have to be sequential, although the latter are usually found to converge faster (and more stable). In numerical linear algebra, the celebrated Stein–Rosenberg theorem [28] formally proves that in solving certain linear systems, GS updates converge *strictly* faster than their Jacobi counterparts, and often with a larger set of convergent instances. However, this result does not readily apply to bilinear zero-sum games (see Section 3). Our main goal here is to answer the following questions about solving bilinear zero-sum games:

- When exactly does a gradient-type algorithm converge?
- What is the optimal convergence rate by tuning the step size or other parameters?
- Can we prove similar things for Jacobi and GS updates as the Stein–Rosenberg theorem?

Contributions In Section 2, we review bilinear games and popular gradient algorithms. On bilinear games, gradient algorithms have a unified formulation. With this new formulation, we give exact convergence conditions, and show that alternating updates are more stable than their simultaneous counterparts in Section 3. We give optimal convergence rates for different algorithms in Section 4 with supporting experiments in Section 5.

[1] A more thorough version is published at ICLR 2020 [30].

2 Preliminaries

Mathematically, zero-sum *bilinear* games can be formulated as the following min-max problem:

$$\min_{\mathbf{x}\in\mathbb{R}^\mathbf{n}} \max_{\mathbf{y}\in\mathbb{R}^\mathbf{n}} \quad \mathbf{x}^\top \mathbf{E}\mathbf{y} + \mathbf{b}^\top \mathbf{x} + \mathbf{c}^\top \mathbf{y}. \tag{1}$$

(Throughout for simplicity we assume $\mathbf{E}$ is invertible.) For bilinear games, it is well-known that simultaneous gradient descent does not converge [22] and other gradient-based algorithms tailored for min-max optimization have been proposed [16, 5, 10, 20]. These iterative algorithms all belong to the class of general linear dynamical systems (LDSs), and they can be described as:

$$\mathbf{z}^{(t)} = \textstyle\sum_{i=1}^k \mathbf{A}_i \mathbf{z}^{(t-i)} + \mathbf{d}, \quad \mathbf{z}^{(t)} := (\mathbf{x}^{(t)}, \mathbf{y}^{(t)}).$$

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

Theorem 1 (e.g. [12]) *The LDS* $\mathbf{z}^{(t)} = \sum_{i=1}^k \mathbf{A}_i \mathbf{z}^{(t-i)} + \mathbf{d}$ *converges for any initialization* $(\mathbf{z}^{(0)}, \ldots, \mathbf{z}^{(k-1)})$ *iff the spectral radius* $r := \max\{|\lambda| : \det(\lambda^k \mathbf{I} - \sum_{i=1}^k \mathbf{A}_i \lambda^{k-i}) = 0\} < 1$*, in which case* $\{\mathbf{z}^{(t)}\}$ *converges linearly with (asymptotic) exponent* r*.*

Therefore, understanding the bilinear game dynamics reduces to spectral analysis. The (sufficient and necessary) convergence condition reduces to that all roots of the characteristic polynomial lie in the unit circle, which can be conveniently analyzed through the celebrated Schur's theorem [27].

Let us formally define Jacobi and GS updates: Jacobi updates take the form

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

while Gauss–Seidel updates replace $\mathbf{x}^{(t-i)}$ with the more recent $\mathbf{x}^{(t-i+1)}$ in operator T_2, where $T_1, T_2 : \mathbb{R}^{nk} \times \mathbb{R}^{nk} \to \mathbb{R}^n$ can be any update functions. For LDS updates in (2) we find a nice relation between the characteristic polynomials of Jacobi and GS updates:

Theorem 2 (Jacobi vs. Gauss–Seidel) *Let* $p(\lambda, \gamma) = \det(\sum_{i=1}^k (\gamma \mathbf{L}_i + \mathbf{U}_i)\lambda^{k-i} - \lambda^k \mathbf{I})$*, where* $\mathbf{A}_i = \mathbf{L}_i + \mathbf{U}_i$ *and* $\mathbf{L}_i$ *is strictly lower block triangular. Then, the characteristic polynomial of the Jacobi update is* $p(\lambda, 1)$ *while that of the Gauss–Seidel update is* $p(\lambda, \lambda)$*.*

Next, we define some popular gradient algorithms for finding saddle points in the min-max problem $\min_x \max_y f(\mathbf{x}, \mathbf{y})$. Unlike their usual presentations, we introduced more "step sizes" for refined analysis, as the enlarged parameter space often contain choices for faster linear convergence (see Section 4). We only define the Jacobi updates, while the GS counterparts can be easily inferred.

Extra-gradient (EG) We study a generalized version of EG, defined as follows:

$$\mathbf{x}^{(t+1/2)} = \mathbf{x}^{(t)} - \gamma_2 \nabla_\mathbf{x} f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}), \qquad \mathbf{y}^{(t+1/2)} = \mathbf{y}^{(t)} + \gamma_1 \nabla_\mathbf{y} f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}); \tag{2}$$
$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - \alpha_1 \nabla_\mathbf{x} f(\mathbf{x}^{(t+1/2)}, \mathbf{y}^{(t+1/2)}), \qquad \mathbf{y}^{(t+1)} = \mathbf{y}^{(t)} + \alpha_2 \nabla_\mathbf{y} f(\mathbf{x}^{(t+1/2)}, \mathbf{y}^{(t+1/2)}). \tag{3}$$

EG was first proposed in [16] with the restriction $\alpha_1 = \alpha_2 = \gamma_1 = \gamma_2$, under which linear convergence was proved for bilinear games. A slightly more generalized version was analyzed in [17] where $\alpha_1 = \alpha_2$, $\gamma_1 = \gamma_2$, again with linear convergence proved. For later convenience we define $\beta_i = \alpha_i \gamma_i$.

Optimistic gradient descent (OGD) We study a generalized version of OGD, defined as follows:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - \alpha_1 \nabla_\mathbf{x} f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}) + \beta_1 \nabla_\mathbf{x} f(\mathbf{x}^{(t-1)}, \mathbf{y}^{(t-1)}), \tag{4}$$
$$\mathbf{y}^{(t+1)} = \mathbf{y}^{(t)} + \alpha_2 \nabla_\mathbf{y} f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}) - \beta_2 \nabla_\mathbf{y} f(\mathbf{x}^{(t-1)}, \mathbf{y}^{(t-1)}). \tag{5}$$

The original version of OGD was given in [5] with $\alpha_1 = \alpha_2 = 2\beta_1 = 2\beta_2$, and its linear convergence for bilinear games was proved in [17]. A slightly generalized version with $\alpha_1 = \alpha_2$ and $\beta_1 = \beta_2$ was analyzed in [21], again with linear convergence proved.

Momentum method Generalized heavy ball method was proposed and analyzed in [11]:

$$\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} - \alpha_1 \nabla_{\mathbf{x}} f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}) + \beta_1(\mathbf{x}^{(t)} - \mathbf{x}^{(t-1)}), \tag{6}$$

$$\mathbf{y}^{(t+1)} = \mathbf{y}^{(t)} + \alpha_2 \nabla_{\mathbf{y}} f(\mathbf{x}^{(t)}, \mathbf{y}^{(t)}) + \beta_2(\mathbf{y}^{(t)} - \mathbf{y}^{(t-1)}), \tag{7}$$

as a modification of Polyak's heavy ball (HB) [25], which also motivated Nesterov's accelerated gradient algorithm (NAG) [23]. For bilinear games, HB and NAG are the same and hence we call both the momentum method. For this algorithm our result below improves those obtained in [11].

3 Exact conditions

With tools from Section 2, we give necessary and sufficient conditions under which a gradient-based algorithm converges for bilinear games. For simplicity, we mostly take the parameters for the two sets of variables to be the same, i.e., $\alpha_1 = \alpha_2 = \alpha$, $\beta_1 = \beta_2 = \beta$ and $\gamma_1 = \gamma_2 = \gamma$ (if available). The same conditions for more general algorithms can be found in our complete paper.

Theorem 3 (EG) *For generalized EG with $\alpha_1 = \alpha_2 = \alpha$ and $\gamma = \beta/\alpha$, linear convergence is achieved iff for any singular value σ of E, we have $\alpha^2\sigma^2 + (\beta\sigma^2 - 1)^2 < 1$ for the Jacobi update, and $0 < \beta\sigma^2 < 2$ and $|\alpha\sigma| < 2 - \beta\sigma^2$ for the GS update. If $2\beta + \alpha^2 < 2/\sigma_1^2$, the convergence region of GS updates* ***strictly*** *include that of Jacobi updates.*

Theorem 4 (OGD) *For generalized OGD with $\alpha_1 = \alpha_2 = \alpha$, linear convergence is achieved iff for any singular value σ of E, we have: $0 < \beta\sigma < 1$, $\beta < \alpha < \beta\frac{3-\beta^2\sigma^2}{1+\beta^2\sigma^2}$ for the Jacobi update, and $|\alpha + \beta|\sigma < 2$, $|1 + \alpha\beta\sigma^2| > 1 + \beta^2\sigma^2$ for the GS update. The convergence region of GS updates* ***strictly*** *include that of Jacobi updates.*

Theorem 5 (momentum) *For generalized momentum with $\alpha_1 = \alpha_2 = \alpha$, the Jacobi update never converges, while the GS update with $\beta_1 = \beta_2 = \beta$ converges iff for any singular value σ of E, we have $-1 < \beta < 0$, $|\alpha\sigma| < 2(1 + \beta)$. If $\beta_2 = 0$, the exact condition is $-1 < \beta_1 < 0$ and $0 < \alpha\sigma_1 < 2\sqrt{1 + \beta_1}$.*

Prior to our work, only sufficient conditions for linear convergence are given for the usual EG and OGD; see Section 2 above. For the momentum method, our result improves upon [11] where the authors only considered specific cases of parameters. For example, they only considered $\beta \geq -1/16$ for Jacobi momentum, and $\beta_1 = -1/2$, $\beta_2 = 0$ for GS momentum. Our Theorem 5 gives a more complete picture. (For an even more general result please refer to our ICLR paper.)

In the theorems above, we use the term "convergence region" to denote a set of the parameters (α, β or γ) where the algorithm converges. Our result shares similarity with the Stein–Rosenberg theorem [28], which only applies to solving linear systems with non-negative matrices. In this sense, our results extend the Stein–Rosenberg theorem to cover nontrivial bilinear games.

4 Optimal rates

In this section we study the optimal convergence rates of EG and OGD. We define the exponent of linear convergence as $r = \lim_{t\to\infty} ||\mathbf{z}^{(t)}||/||\mathbf{z}^{(t-1)}||$. For ease of presentation we fix $\alpha_1 = \alpha_2 = \alpha > 0$ and we use r_* to denote the optimal rate (w.r.t. the parameters α, β, γ). In Theorem 7, the exact formula β_* in Jacobi OGD, as well as more relevant results, can be found in our full paper.

Theorem 6 (EG optimal) *Both Jacobi and GS EG achieve the optimal exponent of linear convergence $r_* = (\kappa^2 - 1)/(\kappa^2 + 1)$ at $\alpha \to 0$ and $\beta_1 = \beta_2 = 2/(\sigma_1^2 + \sigma_n^2)$. As $\kappa \to \infty$, $r_* \to 1 - 2/\kappa^2$.*

Theorem 7 (OGD optimal) *For Jacobi OGD with $\beta_1 = \beta_2 = \beta$, to achieve the optimal linear convergence, we must have $\alpha \leq 2\beta$. At $\beta = \alpha/2 = \beta_*$, $r_* \sim 1 - 1/(6\kappa^2)$ at large κ. For GS OGD with $\beta_2 = 0$, $r_* = \sqrt{(\kappa^2 - 1)/(\kappa^2 + 1)} \sim 1 - 1/\kappa^2$, at $\alpha = \sqrt{2}/\sigma_1$ and $\beta_1 = \sqrt{2}\sigma_1/(\sigma_1^2 + \sigma_n^2)$.*

5 Experiments

Bilinear game We experiment on a bilinear game and choose the optimal parameters as suggested in Theorem 6 and 7. The results, shown in Figure 1, agree with our theory.

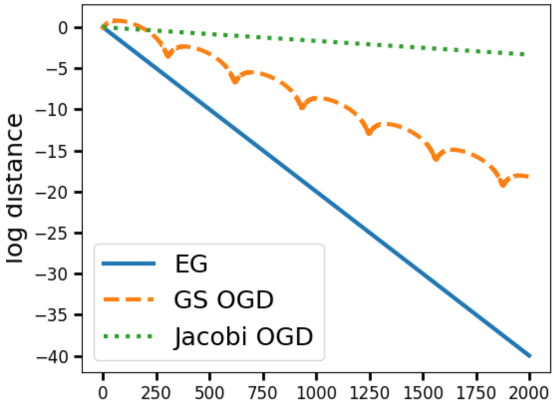

Figure 1: Linear convergence of optimal EG, Jacobi OGD, Gauss–Seidel OGD in a bilinear game

Wasserstein GAN As in [5], we consider a WGAN [1] that learns the mean of a Gaussian: with $s(x)$ the sigmoid function. Near the saddle point $(\theta^*, \phi^*) = (0, v)$ the min-max optimization can be treated as a bilinear game. Since we are doing stochastic versions of the algorithms, we should not expect they will converge exactly to a saddle point. Instead, convergence to a neighborhood is good enough.

With GS updates, we find that Adam [15] diverges, SGD goes around a limit cycle, and EG converges, as shown in the left panel of Figure 2. Our next experiment shows that generalized algorithms may have an advantage over traditional ones. Inspired by Theorem 6, we compare the convergence of two EGs with the same parameter $\beta = \alpha\gamma$, and find that with scaling (decreasing α), EG converges faster to a neighborhood of the saddle point with less oscillation, as shown in the right panel of Figure 2. Note that we always use the squared distance as a measure of convergence.

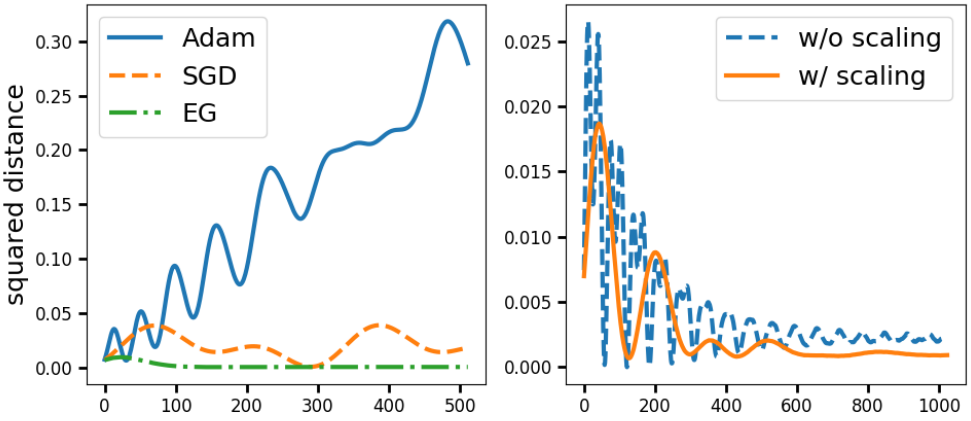

Figure 2: Left: comparison among gradient algorithms; Right: the scaling effect of EG

Finally, we compare Jacobi updates with GS updates. In Figure 3, GS updates converge even when the corresponding Jacobi updates do not.

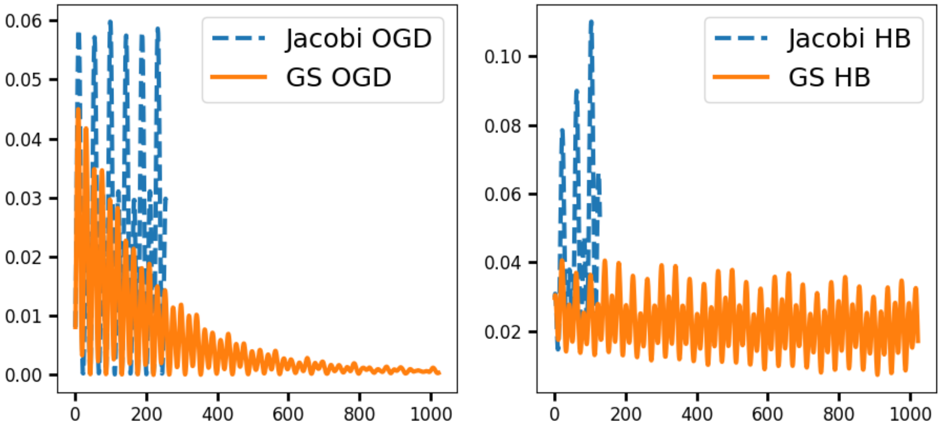

Figure 3: Jacobi vs. GS updates. Left: OGD with $\alpha = 0.2$, $\beta_1 = 0.1$, $\beta_2 = 0$; Right: Momentum with $\alpha = 0.08$, $\beta = -0.1$. We plot only a few epochs for Jacobi updates if they do not converge

6 Conclusions

In this paper, we study convergence of gradient algorithms on bilinear games. Surprisingly, even such a simple game could provide us with great insights for practice. The lessons we have learned are: alternating updates are often more stable than simultaneous updates; by generalizing existing algorithms we can achieve faster convergence rates. We provide guidance for choosing hyper-parameters in bilinear games which could potentially generalize to GAN training.

Acknowledgement

We thank Argyrios Deligkas and Sarath Pattathil for pointing out additional references. This work is supported by NSERC.

References

[1] M. Arjovsky, S. Chintala, and L. Bottou. Wasserstein generative adversarial networks. In International Conference on Machine Learning, 2017.

[2] K. J. Arrow, L. Hurwicz, and H. Uzawa. Studies in linear and non-linear programming. Stanford University Press, 1958.

[3] R. E. Bruck. On the weak convergence of an ergodic iteration for the solution of variational inequalities for monotone operators in Hilbert space. Journal of Mathematical Analysis and Applications, 61(1):159–164, 1977.

[4] X. Chen, X. Deng, and S.-H. Teng. Settling the complexity of computing two-player Nash equilibria. Journal of the ACM, 56(3):14, 2009.

[5] C. Daskalakis, A. Ilyas, V. Syrgkanis, and H. Zeng. Training GANs with optimism. In International Conference on Learning Representations, 2018.

[6] A. Deligkas, J. Fearnley, R. Savani, and P. Spirakis. Computing approximate Nash equilibria in polymatrix games. Algorithmica, 77(2):487–514, 2017.

[7] V. F. Dem'yanov and A. B. Pevnyi. Numerical methods for finding saddle points. USSR Computational Mathematics and Mathematical Physics, 12(5):11–52, 1972.

[8] S. S. Du, J. Chen, L. Li, L. Xiao, and D. Zhou. Stochastic variance reduction methods for policy evaluation. In International Conference on Machine Learning, pages 1049–1058, 2017.

[9] Y. Freund and R. E. Schapire. Adaptive game playing using multiplicative weights. Games and Economic Behavior, 29(1-2):79–103, 1999.

[10] G. Gidel, H. Berard, G. Vignoud, P. Vincent, and S. Lacoste-Julien. A variational inequality perspective on generative adversarial networks. In International Conference on Learning Representations, 2019.

[11] G. Gidel, R. A. Hemmat, M. Pezeshki, G. Huang, R. Lepriol, S. Lacoste-Julien, and I. Mitliagkas. Negative momentum for improved game dynamics. In AISTATS, 2019.

[12] I. Gohberg, P. Lancaster, and L. Rodman. Matrix polynomials. Academic Press, 1982.

[13] E. G. Gol'shtein. A generalized gradient method for finding saddlepoints. Ekonomika i matematicheskie metody, 8(4):569–579, 1972.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In Advances in Neural Information Processing Systems, pages 2672–2680, 2014.

[15] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. In International Conference on Learning Representations, 2015.

[16] G. M. Korpelevich. The extragradient method for finding saddle points and other problems. Matecon, 12:747–756, 1976.

[17] T. Liang and J. Stokes. Interaction matters: A note on non-asymptotic local convergence of generative adversarial networks. In AISTATS, 2019.

[18] P. L. Lions. Une méthode itérative de résolution d'une inéquation variationnelle. Israel Journal of Mathematics, 31(2):204–208, 1978.

[19] 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.

[20] L. Mescheder, S. Nowozin, and A. Geiger. The numerics of GANs. In Advances in Neural Information Processing Systems, pages 1825–1835, 2017.

[21] 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.

[22] A. S. Nemirovski and D. B. Yudin. Problem complexity and method efficiency in optimization. Wiley, 1983.

[23] Y. Nesterov. A method for unconstrained convex minimization problem with the rate of convergence $O(1/k^2)$. Doklady Akademii Nauk, 269:543–547, 1983.

[24] Christos H Papadimitriou. On the complexity of the parity argument and other inefficient proofs of existence. Journal of Computer and system Sciences, 48(3):498–532, 1994.

[25] B. T. Polyak. Some methods of speeding up the convergence of iteration methods. USSR Computational Mathematics and Mathematical Physics, 4(5):1–17, 1964.

[26] R. T. Rockafellar. Monotone operators and the proximal point algorithm. SIAM journal on control and optimization, 14(5):877–898, 1976.

[27] I. Schur. Über Potenzreihen, die im Innern des Einheitskreises beschränkt sind. Journal für die reine und angewandte Mathematik, 147:205–232, 1917.

[28] P. Stein and R. L. Rosenberg. On the solution of linear simultaneous equations by iteration. Journal of the London Mathematical Society, 1(2):111–118, 1948.

[29] P. Tseng. On linear convergence of iterative methods for the variational inequality problem. Journal of Computational and Applied Mathematics, 60(1-2):237–252, 1995.

[30] Guojun Zhang and Yaoliang Yu. Convergence behaviour of some gradient-based methods on bilinear games. In International Conference on Learning Representations, 2020.