Proceedings of Machine Learning Research vol 125:1–17, 2020 33rd Annual Conference on Learning Theory

Finite Regret and Cycles with Fixed Step-Size via Alternating Gradient Descent-Ascent

James P. Bailey JAMESPBAILEY@TAMU.EDU
Texas A&M University

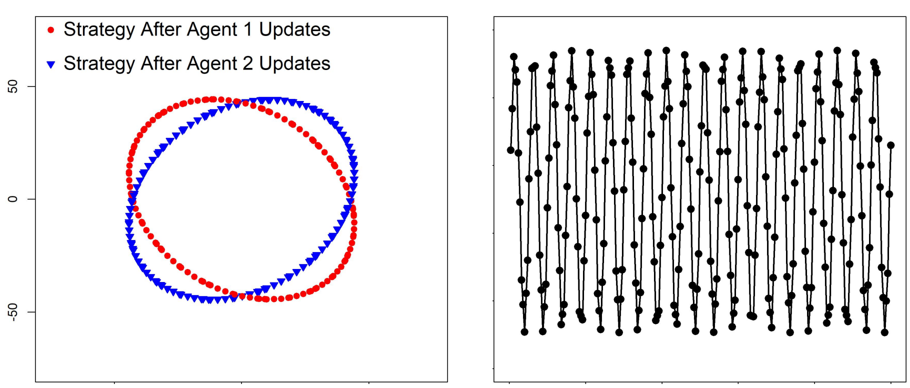
Gauthier Gidel GIDELGAU@MILA.QUEBEC
Mila & DIRO, University of Montreal, ElementAI

Georgios Piliouras GEORGIOS@SUTD.EDU.SG
Singapore University of Technology and Design

Editors: Jacob Abernethy and Shivani Agarwal

Abstract

Gradient descent is arguably one of the most popular online optimization methods with a wide array of applications. However, the standard implementation where agents simultaneously update their strategies yields several undesirable properties; strategies diverge away from equilibrium and regret grows over time. In this paper, we eliminate these negative properties by considering a different implementation to obtain $O(1/T)$ time-average regret via arbitrary fixed step-size. We obtain this surprising property by having agents take turns when updating their strategies. In this setting, we show that an agent that uses gradient descent with any linear loss function obtains bounded regret – regardless of how their opponent updates their strategies. Furthermore, we show that in adversarial settings that agents' strategies are bounded and cycle when both are using the alternating gradient descent algorithm.



(a) Player strategies cycle around the max-min equilibrium.

(b) Agent 1's regret remains bounded while oscillating back and forth.

Figure 1: 125 Iterations of Alternating Gradient Descent-Ascent applied in a zero-sum game with initial condition $(x_1^0, x_2^0) = (35, 35)$, $A = [1]$ and learning rate $\eta_1 = \eta_2 = 1/2$.

1. Introduction

Zero-sum games and more generally max-min optimization are amongst the most well studied settings in game theory. Dating back to classic work of von Neumann (1928), which initiated the field of game theory as a whole, it is well understood that zero-sum games admit a "solution". The safety level that each agent can guarantee for themselves, if they were forced to commit to their strategies first, is exactly equal to the best case payoff they will get if they play second, with full information, against a rational opponent. This fact that the ordering of the agents does not matter is captured by arguably the most famous aphorism in game theory, max-min is equal to min-max.

Despite the classically resolved issue of equilibrium computation in zero-sum games, the question of analyzing dynamics in zero-sum games is much less understood. Possibly, the most well known result in the area is that regret minimizing dynamics converge in a time average sense to max-min equilibria (e.g., Freund and Schapire (1999)). However, up until recently, the day-to-day behavior of standard classes of online learning dynamics were not understood. For example, does the day-to-day behavior converge to equilibrium, does it diverge away from it, or does it cycle at a fixed distance from it? The answer to the above questions turns out to be, Yes, Yes and Yes! Or, to be more precise, the answer depends critically on the choice of the dynamics.

When studying dynamics in continuous-time, e.g., the continuous-time-analogue of Multiplicative Weights Update, replicator dynamics, the dynamics "cycle" around at a constant Kullback-Leibler divergence from equilibrium as shown by Piliouras and Shamma (2014). In fact, this result generalizes for all continuous-time variants of all Follow-the-Regularized Leader (FTRL) algorithms (Mertikopoulos et al., 2018). Moreover, these dynamics have bounded (total/aggregate) regret *in arbitrary games*. This is an impressive level of regularity and predictability of the dynamics, which despite not being equilibrating, allow us to make strong predictions about their day-to-day behavior. Behind this clockwork kind of regularity lies the fact that these dynamics are Hamiltonian (Bailey and Piliouras, 2019a). As in the case, e.g., of planetary orbits, or a pendulum, there is a lot of hidden structure in the motion, laws that bind and control the evolution of all particles.

Unfortunately, this level of regularity comes at a cost of using a continuous-time model. This is of course a simplifying modelling assumption. It does not capture the reality of how games, economic competition is played out in practice. More importantly, it fails to capture the reality of some modern engineering applications, such as Generative Adversarial Networks (GANs) (Goodfellow et al., 2014), where online training algorithms compete against each other to improve two (opposing) AI algorithms. Hopefully, we could just naively discretize the aforementioned dynamics and their behavior would for the most part stay intact. Unfortunately, this is far from the truth.

Bailey and Piliouras (2018) first proved that for all Follow-the-Regularized-Leader algorithms (e.g. Gradient Descent or Multiplicative Weights) diverge away from the Nash equilibrium in zero-sum games. This is a robust finding that holds regardless of the step-size that the agents are using, even if the agents are using different or shrinking step-sizes, or even if they are using different dynamics (i.e., mix-and-match regularizers). The proof by picture is as follows: If gradient descent-ascent in continuous-time moves along a Euclidean ball centered at the equilibrium then the naive discretization takes a discrete, non-negligible step along the tangent. Hence, the distance (radius) from the equilibrium grows and we keep moving away from the equilibrium. Even more distressingly, not only are equilibria unstable but furthermore the dynamics are formally chaotic as small perturbations of initial conditions are amplified exponentially fast (Cheung and Piliouras, 2019). In a nutshell, by discretizing gradient descent, moving from a differential equation to an actual implementable algorithm, all system regularity is lost. The discrete and continuous-time behavior may only differ by a little bit in each step, but these errors snowball quickly.

Given the instability of the naive, standard discretization of gradient descent, several other algorithms have been suggested which converge provably to Nash equilibria in zero-sum games such as the extragradient method (Korpelevich, 1976) and its variants (Gidel et al., 2019a; Mertikopoulos et al., 2019), optimistic mirror descent (Rakhlin and Sridharan, 2013; Daskalakis et al., 2018; Daskalakis and Panageas, 2018) and some other methods using negative momentum or second order information (Gidel et al., 2019b; Balduzzi et al., 2018; Abernethy et al., 2019).

Going back to the picture of simultaneous gradient descent-ascent as a tangent to a ball centered at equilibrium, these approaches alter the dynamics, so that the discrete-step is now facing the interior of the ball, more like a chord than a tangent, decreasing the distance from equilibrium and forcing convergence in the long run.

We take a different approach when it comes to discretizing the system dynamics. We ask, as von Neumann did for equilibrium computation, does the ordering of the agents matter? What if the min and max agents did not update their behavior simultaneously but instead they took turns. This is actually common practice in training neural networks as no extra memory is needed to hold the previous state/parameters of any network. Even for economic competition in markets, this is a rather reasonable model with firms taking turns responding to the last move of the competition. Could it be that this standard alternating gradient descent-ascent implementation recovers some of the impressive regularities of the continuous-time model and if so to what extent?

Our Contributions. *Optimization/Regret.* We study the behavior of gradient descent with fixed step-size in unconstrained two-agent (zero-sum) games. In a twist on the standard theory of online learning, we consider agents that take turns updating their strategies. We establish that if an agent uses gradient descent with arbitrary fixed step-size when agents are sequentially updating their strategies, then she obtains bounded regret (Theorem 1) as depicted in Figure 1(*b*). Moreover, Theorem 1 holds regardless of how her opponent updates his strategies and therefore the result immediately extends to non-zero-sum games. We establish this surprising property by showing that an agent's distance from optimality changes proportionally to her payoff in any given iteration (Lemma 1). This allows us to compute both the regret and utility of an agent with only knowledge of the first and last strategy she used, regardless of how her opponent updates his strategies. The bound on regret quickly follows by considering the worst-case final strategy.

Game Theory. We further explore the asymptotic properties of alternating gradient descent specifically in the setting of zero-sum games. We show that when agents use gradient descent sequentially that the strategies approximately cycle (Theorem 3) as depicted in Figure 1(*a*). More formally, alternating gradient descent admits Poincaré recurrence in the setting of two-agent zero-sum games. Theorem 3 is established in two parts: First, we show that the alternating gradient descent algorithm approximately preserves the distance to the equilibrium (Theorems 5 and 6) therefore implying bounded orbits. This proof relies on directly finding an exact invariant energy function capturing all agent strategies (Theorem 4). Second, we show that the algorithm preserves volume when updating a measurable set of strategies. Together, these two properties imply recurrence. We remark that, Gidel et al. (2019b) first showed that alternating play exhibits bounded orbits. However, we provide a new proof technique that is potentially generalizable to other geometries, provide a clear interpretation in terms of energy conservation and parameters of the game and optimization technique, and uncover an invariant energy function for agent strategies.

2. Preliminaries

A two-agent zero-sum game consists of two agents $\mathcal{N} = \{1, 2\}$ where agent i selects a strategy from $\mathbb{R}^{k_i}$. Utilities of both agents are determined via a payoff matrix $A \in \mathbb{R}^{k_1 \times k_2}$ and linear payoff vectors $b_i \in \mathbb{R}^{k_i}$. Given that agent i selects strategy $x_i \in \mathbb{R}^{k_i}$, agent 1 receives utility $\langle b_2, x_2\rangle + \langle x_1, Ax_2\rangle + \langle x_1, b_1\rangle$ and agent 2 receives utility $-\langle b_2, x_2\rangle - \langle x_1, Ax_2\rangle - \langle x_1, b_1\rangle$. Naturally, both agents want to maximize their payout resulting in the following max-min problem:

$$\max_{x_1 \in \mathbb{R}^{k_1}} \min_{x_2 \in \mathbb{R}^{k_2}} \langle b_2, x_2\rangle + \langle x_1, Ax_2\rangle + \langle x_1, b_1\rangle \qquad \text{(Bilinear Zero-Sum Game)}$$

2.1. Gradient Descent with Simultaneous Play

In many applications of game theory, agents know neither the payoff matrix nor their opponent's strategy. Instead, agents repeatedly play the zero-sum game while updating their strategies iteratively. One of the most common methods for updating strategies is the gradient descent algorithm. In gradient descent, an agent looks at her payout in the previous iteration and then updates her previous strategies by moving in a most beneficial direction. In the setting of (Bilinear Zero-Sum Game), this corresponds to

$$x_1^{t+1} = x_1^t + \eta_1(Ax_2^t + b_1) \qquad \text{and} \qquad x_2^{t+1} = x_2^t - \eta_2(A^\intercal x_1^t + b_2). \qquad \text{(SimGD)}$$

where η_i corresponds to agent i's step-size or learning rate. The larger the step-size, the more rapidly an agent responds to information from previous iterations. Gradient descent is often implemented with time variant step-sizes – most commonly with $\eta_i^t \in \Theta(1/\sqrt{t})$ where agents become less responsive over time. However, in this paper we focus on fixed step-sizes.

In this formulation, agents simultaneously update their strategy. That is, x_1^t and x_2^t are played at the same time. As a result, the cumulative utility of (SimGD) (or any simultaneous update algorithm) for agent 1 after T iterations is $\sum_{t=0}^{T} \left(\langle b_2, x_2^t\rangle + \langle x_1^t, Ax_2^t\rangle + \langle x_1^t, b_1\rangle\right)$.

2.2. Gradient Descent with Alternating Play

In many application of game theory, agents do not update their strategies until they see a change in the system. In the case of two-agent games, this corresponds to agents updating their strategies sequentially, i.e., agent 1 updates her strategy, then agent 2 updates his strategy, then agent 1 updates her strategy and so on. In the setting of gradient descent, this corresponds to

$$x_1^{t+1} = x_1^t + \eta_1(Ax_2^t + b_1) \qquad \text{and} \qquad x_2^{t+1} = x_2^t - \eta_2(A^\intercal x_1^{t+1} + b_2). \qquad \text{(AltGD)}$$

Computing the total utility when agents alternate their updates is slightly different. Agent 1 plays strategy x_1^t against x_2^t when agent 2 updates his strategy and plays x_1^{t+1} against x_2^t when agent 1 updates her strategy. This results in the following cumulative utility after agent 1 updates her strategy T times: $\sum_{t=0}^{T-1} \left(2 \cdot \langle b_2, x_2^t\rangle + \langle x_1^{t+1} + x_1^t, Ax_2^t\rangle + \langle x_1^{t+1} + x_1^t, b_1\rangle\right)$.

2.3. Problem Reduction

In this section we show that the vectors b_1 and b_2 in (Bilinear Zero-Sum Game) can be assumed, without any loss of generality, to be zero. Such result requires the following assumption:

Assumption 1 (Bilinear Zero-Sum Game) *has a solution* (x_1^*, x_2^*).

Under Assumption 1 the first order stationary conditions are

$$A^\intercal x_1^* = -b_2 \qquad \text{and} \qquad Ax_2^* = -b_1\,. \tag{1}$$

Considering the changes of variable, $\tilde{x}_1^t = x_1^t - x_1^*$ and $\tilde{x}_2^t = x_2^t - x_2^*$, the equations (SimGD) and (AltGD) can be simplified as,

$$\begin{cases}\tilde{x}_1^{t+1} = \tilde{x}_1^t + \eta_1 A\tilde{x}_2^t \\ \tilde{x}_2^{t+1} = \tilde{x}_2^t - \eta_2 A^\intercal \tilde{x}_1^t\end{cases} \quad \text{and} \quad \begin{cases}\tilde{x}_1^{t+1} = \tilde{x}_1^t + \eta_1 A\tilde{x}_2^t \\ \tilde{x}_2^{t+1} = \tilde{x}_2^t - \eta_2 A^\intercal \tilde{x}_1^{t+1}\,.\end{cases}$$

These update schemes correspond to (SimGD) and (AltGD) applied to (Bilinear Zero-Sum Game) where b_1 and b_2 are null. Thus in the rest of the paper, without any loss of generality, we will consider the following max-min problem,

$$\max_{x_1\in\mathbb{R}^{k_1}} \min_{x_2\in\mathbb{R}^{k_2}} \langle x_1, Ax_2\rangle \tag{Zero-Sum Game}$$

2.4. Regret

The standard way of measuring the performance of an algorithm is by a notion known as regret. Regret compares the total utility gained by a fixed strategy x_1 to the utility agent 1 receives by using an algorithm. In the case of simultaneous updates, as in (SimGD), regret is formally given by

$$\left\langle x_1, \sum_{t=0}^{T} Ax_2^t \right\rangle - \sum_{t=0}^{T} \langle x_1^t, Ax_2^t\rangle \tag{Regret for Simultaneous Play}$$

where the second term corresponds to the utility agent 1 received by using (SimGD) and first term corresponds to the utility she would of received if she played x_1 on every iteration (assuming agent 2 still uses the strategies $\{x_2^t\}_{t=0}^T$). In the case of constrained optimization, regret is typically evaluated where x_1 is the best fixed strategy, i.e., the optimizer of $\left\langle x_1, \sum_{t=1}^T Ax_2^t\right\rangle$. However, in unconstrained optimization there is rarely an optimizer to this expression.

When agents update sequentially, regret is computed slightly differently. As discussed in the previous section, agent 1 sees the strategy x_2^t twice – once when agent 1 updates and once when agent 2 updates. As a result, agent 1's regret when updating sequentially is

$$\left\langle 2x_1, \sum_{t=0}^{T-1} Ax_2^t \right\rangle - \sum_{t=0}^{T-1} \left\langle x_1^{t+1} + x_1^t, Ax_2^t\right\rangle. \tag{Regret for Alternating Play}$$

Typically an algorithm is said to perform well if its regret is bounded above by a sublinear function with respect to any fixed strategy. If regret grows at a rate of $o(T)$ with respect to a fixed strategy, then the average regret grows as at a rate of $o(1)$ and, in the limit, the algorithm performs no worse on average as that fixed strategy. Moreover, in the zero-sum setting, the time-average strategy converges to the set of Nash equilibria.

2.5. Continuous-Time Gradient Descent: A Motivation for Alternating Play

The primary motivation for this paper is the continuous-time analogue of gradient descent. In particular, the integration technique used to obtain (AltGD) from the continuous-time analogue well approximates continuous-time and therefore offer similar guarantees for behavior in the system. The continuous-time analogue of (SimGD) and (AltGD) is

$$x_1(t) = x_1(0) + \eta_1 \int_0^t Ax_2(s)ds \qquad \text{and} \qquad x_2(t) = x_2(0) - \eta_2 \int_0^t A^\intercal x_1(s)ds\,, \quad \text{(Cont.GD)}$$

where η_i denotes the learning rate used by agent i. (SimGD) is obtained from (Cont.GD) via Euler integration, i.e., x_i^t is simply the first order approximation of $x_i(t)$ from the point $x_i(t-1)$.

Mertikopoulos et al. (2018) showed that (Cont.GD) cycles around the equilibrium of the system on convex orbits. Therefore, (SimGD) should diverge from the equilibrium since it is the first order approximation of (Cont.GD). Indeed, this is first formally shown for a more general class of update rules including gradient descent and multiplicative weights by Bailey and Piliouras (2018) and for gradient descent in unconstrained bilinear games by Gidel et al. (2019b). Moreover, if agent 1 uses (Cont.GD) and agent 2 uses any continuous update rule, then agent 1 obtains bounded regret even in non-zero-sum games (Mertikopoulos et al., 2018). Therefore, (Cont.GD) obtains impressive regret guarantees in games.

Unfortunately, continuous-time algorithms are difficult to run and online optimization typically relies on discrete-time algorithms. Regrettably, standard discrete-time algorithms fall short relative to their continuous-time analogues; it has long been believed that (SimGD) with fixed step-size has linear regret and therefore offers no nice long-term guarantees. Recently, however, Bailey and Piliouras (2019b) showed that (SimGD) with arbitrary fixed step-size also obtains $\Theta(\sqrt{T})$ regret in bounded 2-dimensional zero-sum games and offered experimental evidence to suggest the result carries over to higher dimensions. Stronger regret guarantees, i.e., $o(\sqrt{T})$, are possible in discrete-time game theoretic settings via a combination of decreasing step-sizes and tailored algorithms (Rakhlin and Sridharan, 2013; Syrgkanis et al., 2015; Foster et al., 2016), however, even these improved guarantees fall short of the bounded regret obtained by (Cont.GD).

Bailey and Piliouras (2019a) recently offered some insights on the shortcomings of (SimGD). They showed that (Cont.GD) in different classes of games forms a Hamiltonian system – a common dynamical system studied in mathematical physics that models energy preserving systems such as ideal pendulums. In the case of zero-sum games, the conserved energy corresponds to the combined norm of the strategies $||x_1||^2/\eta_1 + ||x_2||^2/\eta_2$ partially explaining the cyclic behavior found by Mertikopoulos et al. (2018). However, Euler integration is well-known to be a poor estimator of Hamiltonian systems and it is therefore unsurprising that (Cont.GD) differs greatly from (SimGD).

To obtain behavior that is similar to (Cont.GD), we must use an integration technique that better preserves the dynamics of the original system. Fortunately, there is a particular class of integrators known as symplectic integrators that are well-known for their ability to approximate Hamiltonian systems (Hairer et al., 2006). In particular, Hairer (2005) showed that symplectic integrators approximately preserve the energy of a Hamiltonian system for exponentially long periods of time relative to the inverse of the fixed step-size. In the setting of Gradient Descent, that means that we could approximately preserve the energy in (Cont.GD) for arbitrarily long periods of time by applying a symplectic integrator with sufficiently small step-size.

Regrettably, many symplectic integrators would require agents to coordinate when updating their strategies. By the very nature of a zero-sum game, this coordination would be unnatural and could only be applied in artificial settings such as GANS. (AltGD), however, is directly inspired by combining (Cont.GD) with the symplectic integration technique known as Verlet integration or leapfrogging. The resulting system does not require agents to coordinate, outside the fact that agents have to alternate taking turns. This is in contrast with the more typical assumption of synchronous/simultaneous play (e.g., (Syrgkanis et al., 2015; Foster et al., 2016)). A further contrast, hindering direct comparisons is that this technique calls for a fixed step-size, as opposed to the standard decaying step-sizes. This choice is arguably advantageous as in many ML contexts we would like to use a sufficiently large fixed step-size in order to reduce the training time as a vanishing step-size with an increasing horizon would lead to a prohibitively slow training method.

By the work of Hairer (2005), we expect that (AltGD) with fixed step-size should behave similarly to (Cont.GD) – at least for exponentially long periods of time relative to the step-size. Indeed we actually show a stronger result; (AltGD) with a fixed step-size has the same guarantees of (Cont.GD) forever. Specifically, if agent 1 uses (AltGD) with arbitrary fixed step-size then she obtains bounded regret regardless of how her agent's opponent updates. Moreover, if both agents use (AltGD) then the quantity $\left(||x_1^t||^2/\eta_1 + ||x_2^t||^2/\eta_2 + \langle x_1^t, Ax_2^t \rangle\right)$ is preserved and the strategies $\{x_1^t, x_2^t\}_{t=0}^{\infty}$ cycle for $\sqrt{\eta_1\eta_2} \leq \frac{2}{||A||}$, allowing step-sizes that do not vanish with an infinite horizon.

We proceed by proving bounded regret in Section 3 and recurrent behavior in Section 4.

3. Bounded Regret with Fixed Step-Size in Gradient Descent.

In this section, our focus we will be on the regret generated by an agent playing according to (AltGD). Interesting, our result holds no matter how the opponent updates his strategy. This general setting is particularly interesting because it is able to model an environment with only partial information where the agents might even not know that they are playing a game.

Before stating the main theorem of this section, we present a lemma that provides an interpretation of each agent's payoff in terms of energy fluctuation. The norm of an agent's strategy, rescaled by its step-size, can be seen as an energy that varies proportionally to its payoff.

Lemma 1 *When agent 1 updates via (AltGD), the size of x_1^t increases proportionally to agent 1's payoff since her update in iteration t. Formally,*

$$\frac{||x_1^{t+1}||^2 - ||x_1^t||^2}{\eta_1} = \left\langle x_1^{t+1} + x_1^t, Ax_2^t \right\rangle. \tag{2}$$

Similarly, when agent 2 updates via (AltGD), the size of x_2^t increases proportionally to agent 2's payoff since his update in iteration t. Formally,

$$\frac{||x_2^{t+1}||^2 - ||x_2^t||^2}{\eta_2} = -\left\langle x_1^{t+1}, A(x_2^{t+1} + x_2^t) \right\rangle. \tag{3}$$

Proof Recall the update rule for agent 1 that updates via (AltGD) is $x_1^{t+1} = x_1^t + \eta_1 Ax_2^t$. Thus,

$$\left\langle x_1^{t+1} + x_1^t, Ax_2^t \right\rangle = \frac{\left\langle x_1^{t+1} + x_1^t, x_1^{t+1} - x_1^t \right\rangle}{\eta_1} = \frac{||x_1^{t+1}||^2 - ||x_1^t||^2}{\eta_1} \tag{4}$$

completing the first claim. The second follows by substituting $-\eta_2 A^\intercal x_1^{t+1} = x_2^{t+1} - x_2^t$. ■

From this lemma, an explicit bound on the regret of an agent that uses (AltGD) easily follows independently of what update rule her opponent uses. This result, is a significant improvement in comparison to the $\Theta(\sqrt{T})$ regret of (SimGD) and surprisingly matches the result on continuous-time gradient descent provided by Mertikopoulos et al. (2018, Thm. 3.1).

Theorem 1 (Bounded Regret – i.e., $O\left(1/T\right)$ Time-Average Regret) *If agent 1 uses (AltGD) with an arbitrary fixed step-size η_1 in (Zero-Sum Game), then she obtains bounded regret with respect to any fixed strategy x_1 regardless of how her opponent updates his strategies.*

Proof By Section 2.3, it suffices to show the result for (Zero-Sum Game). Following from Lemma 1, Agent 1's regret with respect to strategy x_1 is

$$\left\langle 2x_1, \sum_{t=0}^{T-1} Ax_2^t \right\rangle - \sum_{t=0}^{T-1} \left\langle x_1^{t+1} + x_1^t, Ax_2^t \right\rangle = \frac{\left\langle 2x_1, x_1^T - x_1^0 \right\rangle}{\eta_1} - \sum_{t=0}^{T} \frac{||x_1^{t+1}||^2 - ||x_1^t||^2}{\eta_1} \tag{5}$$

$$= \frac{\left\langle 2x_1, x_1^T - x_1^0 \right\rangle}{\eta_1} - \frac{||x_1^T||^2 - ||x_1^0||^2}{\eta_1} \tag{6}$$

$$= \frac{\left\langle 2x_1 - x_1^T, x_1^T \right\rangle - \left\langle 2x_1 - x_1^0, x_1^0 \right\rangle}{\eta_1} \tag{7}$$

$$\leq \frac{\left\langle x_1^0 - 2x_1, x_1^0 \right\rangle + \|x_1\|^2}{\eta_1} \tag{8}$$

since the expression $x_1^T \mapsto \left\langle 2x_1 - x_1^T, x_1^T \right\rangle$ is maximized when $x_1^T = x_1$. ∎

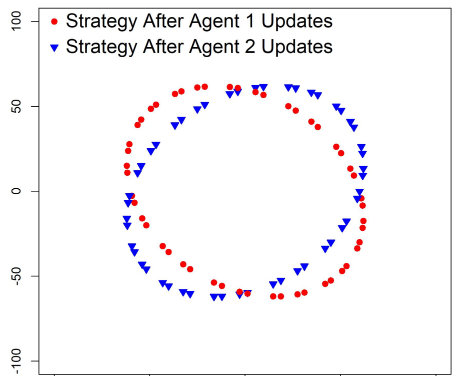
Theorem 2 *If agent 1 uses (AltGD) in any bilinear game (zero-sum or otherwise) with an arbitrary fixed step-size η_1, then she obtains bounded regret with respect to any fixed strategy x_1 regardless of how her opponent updates.*

Neither Lemma 1 nor Theorem 1 make reference to how the opponent updates nor agent 2's utility function and therefore Theorem 2 follows identically.

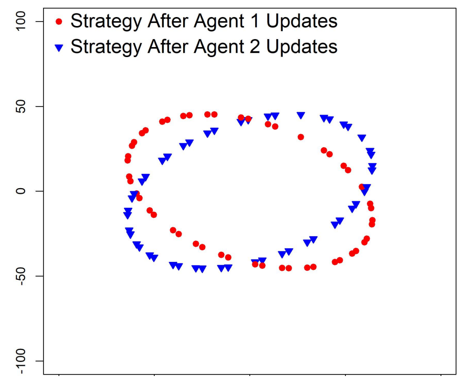
4. Recurrence and Bounded Orbits in Zero-Sum Games

After having shown, in the previous section, that agents that use (AltGD) have bounded regret we would like to investigate the asymptotic properties of their strategies. It has been recently proved that if each agent's strategy are updated though (AltGD), then the energy of the system $\|x_1^t\|^2/\eta_1 + \|x_2^t\|^2/\eta_2$ is bounded above and below (Gidel et al., 2019b, Table 1). Thus, the strategies do not converge to the Nash equilibrium of the game. This boundedness, might indicate a cyclic behavior of the strategies. In the context of high dimensional dynamical system, this cyclic behavior is encompassed by the notion of Poincaré recurrence. Intuitively, a dynamical system is Poincaré recurrent if almost all trajectories return arbitrarily close to their initial position infinitely often.

Indeed, as shown in Figure 2, (AltGD) appears to cycle. In this section, we formally prove the existence of Poincaré recurrence. Our analysis focuses on the strategies after both agents update – i.e., $\{x_1^t, x_2^t\}_{t=0}^{\infty}$ – as depicted by the blue triangles in Figure 2. It also straightforward to extend our analysis to $\{x_1^{t+1}, x_2^t\}_{t=0}^{\infty}$ (depicted by the red circles in Figure 2) through the same techniques.



(a) Strategies "cycle" while approximating $||x_1||^2 + ||x_2||^2 = 60$ with $\eta_1 = \eta_2 = 1/2$



(b) Strategies "cycle" while approximating $||x_1||^2 + 2||x_2||^2 = 60$ with $\eta_1 = 2\eta_2 = 1$.

Figure 2: Initial strategy $(x_1^0, x_2^0) = (60, 0)$ updated with 50 iterations of (AltGD) with $A = [1]$.

More formally, in order to work with this notion of Poincaré recurrence we need to define a measure on $\mathbb{R}^d$. In the following, we will use the Lebesgue measure ℓ. We can thus define the notion of a volume preserving transformation.

Definition 1 (Volume Preserving Transformation (Barreira, 2006)) *A volume preserving transformation is a measurable function $T : \mathbb{R}^d \to \mathbb{R}^d$ such that, for any open set $A \in \mathbb{R}^d$, we have $\ell(A) = \ell(T^{-1}(A))$.*

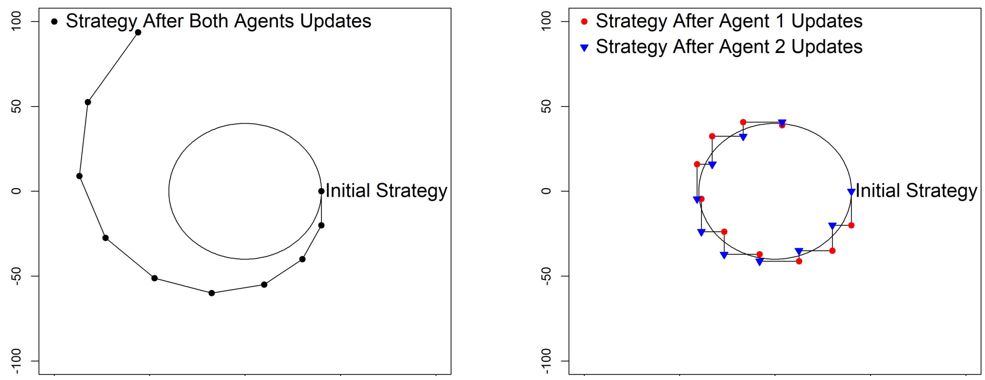
Note that $\ell(A)$ may be infinite. This notion of volume preserving transformation can be more generally defined on a orientable manifold. However, in this work, for simplicity, we will stick with the less general Definition 1. We can thus, state the Poincaré recurrence theorem.

Theorem 3 (Poincaré Recurrence (Poincaré, 1890; Barreira, 2006)) *If a transformation preserves volume and has only bounded orbits then it is Poincaré recurrent, i.e., for each open set there exist orbits that intersect this set infinitely often.*

Furthermore, we can cover any region of $\mathbb{R}^{k_1+k_2}$ by countably many balls of radius $\epsilon/2$, and apply the previous theorem to each ball. We conclude that almost every point returns to within an ϵ of itself. Since $\epsilon > 0$ is arbitrary, we conclude that almost every initial point is recurrent. Formally, we will thus show the following corollary that states the (Poincaré) recurrence of (AltGD).

Corollary 1 *For $\sqrt{\eta_1\eta_2} \leq \frac{2}{||A||}$ the (AltGD) dynamic in (Bilinear Zero-Sum Game) is Poincaré recurrent. Moreover, for almost all initial conditions (x_1^0, x_2^0) there exists an infinite sequence of time periods τ_n such that the $\lim_{n\to\infty}(x_1^{\tau_n}, x_2^{\tau_n}) = (x_1^0, x_2^0)$.*

To show Corollary (1), it suffices to show that (AltGD) has bounded orbits and that (AltGD) preserves volume. In Section 4.1, we first show that the orbits are bounded if $\sqrt{\eta_1\eta_2} \leq \frac{2}{||A||}$, then we show that volume is preserved regardless of the value of η_1 and η_2.



(*a*) Strategies diverging in (SimGD).

(*b*) Strategies preserving energy in (AltGD).

Figure 3: Initial strategy $(x_1^0, x_2^0) = (40, 0)$ updated by 10 iterations of (SimGD) and (AltGD) with $A = [1]$, and $\eta_1 = \eta_2 = 1/2$. The circles denote $\{x : ||x_1||^2 + ||x_2||^2 = 40^2\}$.

4.1. Bounded Orbits and Volume Preservation of Alternating Play

In this section, we prove that if both agents follow (AltGD), then their strategies are bounded. This result was first shown by Gidel et al. (2019b) using linear algebra arguments. However, in this section, we provide the following improvements: a) We provide a new proof technique that is potentially generalizable to other geometries – Gidel et al. (2019b)'s proof heavily relies on the euclidean metric making it challenging to generalize to other geometries. b) For both the upper-bound and the lower-bound, this new proof technique has a clear interpretation in terms of energy conservation and provides an explicit dependence on the constants of the problem. c) We provide an exact invariant function that captures all agent strategies.

The notion of conservation of energy we use in this section, is a perturbed version of the energy used in the continuous case (see Figure 3(*b*) for an illustration). If both agents use (AltGD), the sum of their energies plus agent 1's current utility is constant.

Theorem 4 *If both agents use (AltGD), we have that the following perturbed energy is constant,*

$$\frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2} + \langle x_1^t, Ax_2^t \rangle = \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2} + \langle x_1^0, Ax_2^0 \rangle \ .$$

The proof of Theorem 4 appears in Appendix A.1. If both agents' strategies are unidimentional, Theorem 4 has a geometric interpretation: the orbit of the joint strategy $\{(x_1^t, x_2^t)\}_{t=0}^{\infty}$ belongs to a conic section determined by the equation $({x_1}/{\sqrt{\eta_1}})^2 + ({x_2}/{\sqrt{\eta_2}})^2 + a \cdot x_1 x_2 = 0$. We can show that this conic section is an ellipse if and only if $a^2 - \frac{4}{\eta_1\eta_2} \leq 0$. Thus, for $\sqrt{\eta_1\eta_2} \leq \frac{2}{a}$, the strategies are bounded. The eccentricity and the directions of the principal axis heavily depend on the values of η_1 and η_2. In Figure 2, we observe these elliptic trajectories in the joint strategies space for different values of the step-sizes. This geometric argument can be generalized to strategies belonging to $\mathbb{R}^{k_1} \times \mathbb{R}^{k_2}$ using the singular vectors of A, however, for simplicity, we provide a result in terms of weighted norms using a more concise proof.

Theorem 5 (Bounded orbits) *If both agents use (AltGD) in (Zero-Sum Game) with fixed step-sizes such that* $\sqrt{\eta_1\eta_2} \le \frac{2}{\|A\|}$*, we have that, for all* $t \ge 0$*,*

$$\frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2} \le \frac{\langle x_1^0, Ax_2^0\rangle + \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2}}{1 - \frac{\sqrt{\eta_1\eta_2}\|A\|}{2}}. \tag{9}$$

This theorem is enough to insure that (AltGD) has bounded orbits in order to satisfy the hypothesis of Theorem 3. However, it is worth noting that with the same proof technique we can derive a lower bound on a weighted sum of the norms of each agent's strategies.

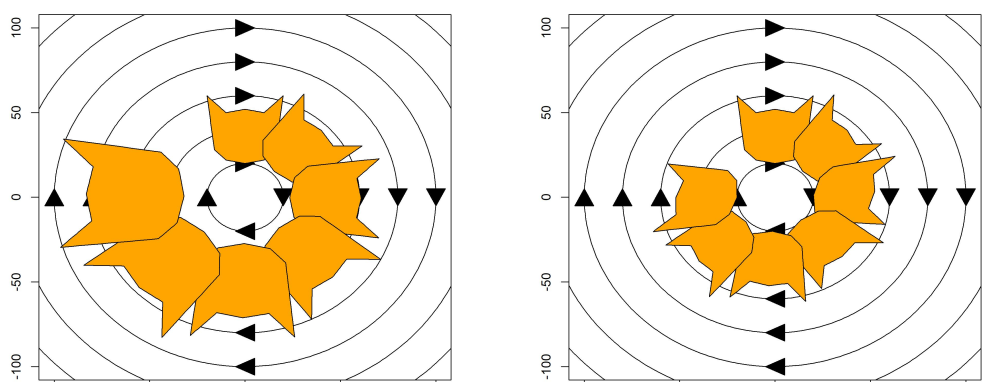
Theorem 6 *If both agents use (AltGD) in (Zero-Sum Game) with* $||x_1^0||^2 + ||x_2^0||^2 > 0$ *and fixed step-sizes such that* $\sqrt{\eta_1\eta_2} \le \frac{2}{\|A\|}$*, then their strategies are bounded away from the equilibrium* $(\mathbf{0}, \mathbf{0})$*. Formally,*

$$\frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2} \ge \frac{2 - \sqrt{\eta_1\eta_2}\|A\|}{2 + \sqrt{\eta_1\eta_2}\|A\|}\left(\frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2}\right).$$

The proof of both theorems appear in Appendix A.2. It is straightforward to extend both proofs to show that the distance to a Nash equilibrium is preserved in (Bilinear Zero-Sum Game) as well.

Together, Theorems 5 and 6 show that (AltGD) approximately preserves the energy $||x_1||^2/\eta_1 + ||x_2||^2/\eta_2$ as depicted in Figures 2 and 3(b). The smaller $\eta_1\eta_2$ is, the closer to a circle the trajectories are. This energy preservation does not occur for (SimGD) as illustrated in Figure 3(a).

Finally, the transformation (AltGD) is volume preserving (Def. 1) as depicted in Figure 4(b). This is in contrast to (SimGD) which expands (see Fig. 4(a) and (Cheung and Piliouras, 2019)).



(a) Volume expands when strategies are updated with (SimGD).

(b) Volume is preserved when strategies are updated with (AltGD).

Figure 4: A collection of strategies (a cat) updated by $0, 4, \ldots, 24$ iterations of (SimGD) and (AltGD) with $x_1, x_2 \in \mathbb{R}$, $A = [1]$, and $\eta_1 = \eta_2 = \frac{1}{5}$.

Theorem 7 (Volume Preservation) *(AltGD) is volume preserving for any step-sizes and any measurable set of initial conditions.*

This theorem results from the fact that the Jacobian of the operator of (AltGD) is block triangular with identity matrices on the diagonal. The full proof appears in Appendix A.3. The Poincaré Recurrence of (AltGD) (Corollary 1) directly follows from Theorem 5 and 7.

5. Extensions to Follow-The-Regularized Leader Algorithms

Many of the approaches and ideas we discuss within this paper extend to the more general set of Follow-The-Regularized Leader (FTRL) algorithms, e.g., multiplicative weights and gradient descent. It is straightforward to extend Theorem 7 to show that alternating play in FTRL preserve volume with respect to the cumulative payoff vectors $y_1^{t+1} = y_1^t + \eta_1 A x_2^t$ and $y_2^{t+1} = y_2^t - \eta_2 A^\intercal x_1^t$ even when the set of feasible strategies are bounded. Moreover, by Bailey and Piliouras (2019a), the continuous-time version of FTRL is a Hamiltonian system where the energy corresponds to the Fenchel-coupling between cumulative payoff vectors and the Nash equilibrium. By Hairer (2005), alternating play will approximately preserve the energy of this Hamiltonian system for exponential time. Thus, it is likely that other variants of FTRL implemented with alternating play have bounded orbits, demonstrate Poincaré recurrence, and have bounded regret even when the strategy space is bounded. (AltGD) in the unbounded case is special however; it's energy perturbation is simply proportional to the utility obtained in the most recent iteration (Theorem 4).

6. Conclusion

We study a natural implementation of gradient descent dynamics in unconstrained zero-sum games. In this implementation, the max and min agent take turns updating their strategies after observing the behavior of their opponent. This dynamic has remarkable properties. First, agents have bounded regret. In fact, this is true not only in zero-sum games but in any general game and online optimization setting. Moreover, in the max-min optimization setting the agents' strategies remain bounded for all time and the dynamics preserve volume. In combination these last two properties imply recurrence, i.e., that the orbits cycle back infinitely often arbitrarily close to their initial conditions. Such advantageous properties were formerly only known for continuous-time dynamics (e.g., (Piliouras and Shamma, 2014; Mertikopoulos et al., 2018)) and moreover are not true for simultaneous gradient descent-ascent updates, which is divergent away from equilibrium (Bailey and Piliouras, 2018) and in fact, formally chaotic (Cheung and Piliouras, 2019).

At its core, our approach is based on recent research advances that enable connections between traditionally separate areas such as game theory, online optimization, Hamiltonian dynamics and numerical analysis. Specifically, Bailey and Piliouras (2019a) show a formal interpretation of continuous-time dynamics in games as Hamiltonian systems. Based on this connection, and the numerous advantageous properties of the continuous-time dynamics, it makes sense to discretize them in a way that mimics the continuous dynamics. Instead of Euler integration, more elaborate tools have been developed, e.g., symplectic integrators that satisfy the volume preserving property of Hamiltonian dynamics as well as other advantageous properties, e.g., approximate energy preservation (see Hairer et al. (2006); Hairer (2005)). Alternating gradient descent-ascent, is directly inspired by a symplectic integration technique known as leapfrogging (Verlet integration) and is thus bringing this point of view to game dynamics.

We hope that this link will allow for the development of new exciting results. One application of particular promise is the understanding of the empirical success of CFR+ (Tammelin, 2014; Bowling et al., 2015) in poker. CFR+ is a variant of Counter Factual Regret minimization (CFR) (Zinkevich et al., 2008) with better empirical performance. A key change of CFR+ is switching from simultaneous updates to alternating updates. So far, there has been no theoretical justification of why such a change should significantly improve performance. We provide the first such result and pave the path for formal arguments for CFR+.

Acknowledgments

Georgios Piliouras acknowledges AcRF Tier 2 grant 2016-T2-1-170, grant PIE-SGP-AI-2018-01, NRF2019-NRF-ANR095 ALIAS grant and NRF 2018 Fellowship NRF-NRFF2018-07.

References

Jacob Abernethy, Kevin A. Lai, and Andre Wibisono. Last-iterate convergence rates for min-max optimization. *arXiv e-prints*, art. arXiv:1906.02027, Jun 2019.

James P. Bailey and Georgios Piliouras. Multiplicative weights update in zero-sum games. In *ACM Conference on Economics and Computation*, 2018.

James P. Bailey and Georgios Piliouras. Multi-agent learning in network zero-sum games is a hamiltonian system. In *18th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, 2019a.

James P. Bailey and Georgios Piliouras. Fast and Furious Learning in Zero-Sum Games: Vanishing Regret with Non-Vanishing Step Sizes. *ArXiv e-prints*, 2019b.

David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. The mechanics of n-player differentiable games. In *ICML*, 2018.

Luis Barreira. Poincare recurrence: old and new. In *XIVth International Congress on Mathematical Physics. World Scientific.*, pages 415–422, 2006.

Michael Bowling, Neil Burch, Michael Johanson, and Oskari Tammelin. Heads-up limit holdem poker is solved. *Science*, 347(6218):145–149, 2015.

Yun Kuen Cheung and Georgios Piliouras. Vortices instead of equilibria in minmax optimization: Chaos and butterfly effects of online learning in zero-sum games. In *COLT6*, 2019.

Constantinos Daskalakis and Ioannis Panageas. Last-iterate convergence: Zero-sum games and constrained min-max optimization. *arXiv preprint arXiv:1807.04252*, 2018.

Constantinos Daskalakis, Andrew Ilyas, Vasilis Syrgkanis, and Haoyang Zeng. Training GANs with optimism. In *ICLR*, 2018.

Dylan J Foster, Thodoris Lykouris, Karthik Sridharan, and Eva Tardos. Learning in games: Robustness of fast convergence. In *Advances in Neural Information Processing Systems*, pages 4727–4735, 2016.

Yoav Freund and Robert E. Schapire. Schapire: Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, page 133, 1999.

Gauthier Gidel, Hugo Berard, Gatan Vignoud, Pascal Vincent, and Simon Lacoste-Julien. A variational inequality perspective on generative adversarial networks. In *ICLR*, 2019a. URL https://openreview.net/forum?id=r1laEnA5Ym.

Gauthier Gidel, Reyhane Askari Hemmat, Mohammad Pezeshki, Gabriel Huang, Rmi Lepriol, Simon Lacoste-Julien, and Ioannis Mitliagkas. Negative momentum for improved game dynamics. In *AISTATS*, 2019b.

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.

Ernst Hairer. Long-time energy conservation of numerical integrators. *Foundation of Computational Mathematics*, pages 162–180, 2005.

Ernst Hairer, Christian Lubich, and Gerhard Wanner. *Geometric numerical integration: structure-preserving algorithms for ordinary differential equations*, volume 31. Springer Science & Business Media, 2006.

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

Panayotis Mertikopoulos, Christos Papadimitriou, and Georgios Piliouras. Cycles in adversarial regularized learning. In *ACM-SIAM Symposium on Discrete Algorithms*, 2018.

Panayotis Mertikopoulos, Bruno Lecouat, Houssam Zenati, Chuan-Sheng Foo, Vijay Chandrasekhar, and Georgios Piliouras. Optimistic mirror descent in saddle-point problems: Going the extra(-gradient) mile. In *ICLR*, 2019. URL https://openreview.net/forum?id=Bkg8jjC9KQ.

G. Piliouras and J. S. Shamma. Optimization despite chaos: Convex relaxations to complex limit sets via Poincaré recurrence. In *Symposium of Discrete Algorithms (SODA)*, 2014.

Henri Poincaré. Sur le problème des trois corps et les équations de la dynamique. *Acta mathematica*, 13(1):A3–A270, 1890.

Sasha Rakhlin and Karthik Sridharan. Optimization, learning, and games with predictable sequences. In *Advances in Neural Information Processing Systems*, pages 3066–3074, 2013.

Walter Rudin. Real and complex analysis (mcgraw-hill international editions: Mathematics series). 1987.

Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E Schapire. Fast convergence of regularized learning in games. In *Advances in Neural Information Processing Systems*, pages 2989–2997, 2015.

Oskari Tammelin. Solving large imperfect information games using cfr+. *arXiv preprint arXiv:1407.5042*, 2014.

John von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische Annalen*, 100:295–300, 1928.

Martin Zinkevich, Michael Johanson, Michael Bowling, and Carmelo Piccione. Regret minimization in games with incomplete information. In *Advances in neural information processing systems*, pages 1729–1736, 2008.

Appendix A. Proof of Theorems and Lemmas

A.1. Proof of Theorem 4

Theorem 4 *If both agents use (AltGD), we have that the following perturbed energy is constant,*

$$\frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2} + \langle x_1^t, Ax_2^t \rangle = \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2} + \langle x_1^0, Ax_2^0 \rangle .$$

Proof Combining (2) and (3) of Lemma 1 yields

$$\frac{||x_1^{t+1}||^2 - ||x_1^t||^2}{\eta_1} + \frac{\|x_2^{t+1}\|^2 - \|x_1^t\|^2}{\eta_2} = \langle x_1^t, Ax_2^t \rangle - \langle x_1^{t+1}, Ax_2^{t+1} \rangle . \tag{10}$$

The result then follows by summing (10) and cancelling out terms. ■

A.2. Proofs of Theorems 5 and 6

Theorem 5 (Bounded orbits) *If both agents use (AltGD) in (Zero-Sum Game) with fixed step-sizes such that $\sqrt{\eta_1\eta_2} \leq \frac{2}{\|A\|}$, we have that, for all $t \geq 0$,*

$$\frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2} \leq \frac{\langle x_1^0, Ax_2^0 \rangle + \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2}}{1 - \frac{\sqrt{\eta_1\eta_2}\|A\|}{2}} . \tag{9}$$

Proof Starting from Theorem 4, we have that,

$$\begin{aligned} \frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2} &= \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2} + \langle x_1^0, Ax_2^0 \rangle - \langle x_1^t, Ax_2^t \rangle & (11)\\ &\leq \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2} + \langle x_1^0, Ax_2^0 \rangle + ||x_1^t|| \cdot ||Ax_2^t|| & (12)\\ &\leq \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2} + \langle x_1^0, Ax_2^0 \rangle + ||A|| \cdot ||x_1|| \cdot ||x_2|| & (13)\\ &\leq \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2} + \langle x_1^0, Ax_2^0 \rangle + \frac{\sqrt{\eta_1\eta_2}||A||}{2}\left(\frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2}\right) & (14) \end{aligned}$$

where (12) is the Cauchy-Schwarz inequality, (13) follows from the definition of the ℓ_2 matrix norm, and (14) follows since $(||x_1||/\sqrt{\eta_1} - ||x_2||/\sqrt{\eta_2})^2 \geq 0$. Rearranging terms yields (9).

Finally, to show that x_i^t is bounded, observe that $\sqrt{\eta_1\eta_2} \leq \frac{2}{||A||} \Rightarrow 1 - \frac{\sqrt{\eta_1\eta_2}\|A\|}{2} \geq 0$ and

$$||x_1^t||^2 \leq \frac{\langle x_1^0, Ax_2^0 \rangle + \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2}}{\frac{1}{\eta_1} - \sqrt{\frac{\eta_2}{\eta_1}}\frac{\|A\|}{2}}. \tag{15}$$

Symmetrically,

$$||x_2^t||^2 \leq \frac{\langle x_1^0, Ax_2^0 \rangle + \frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2}}{\frac{1}{\eta_2} - \sqrt{\frac{\eta_1}{\eta_2}}\frac{\|A\|}{2}} , \tag{16}$$

thereby completing the proof of the theorem. ■

Theorem 6 *If both agents use (AltGD) in (Zero-Sum Game) with $||x_1^0||^2 + ||x_2^0||^2 > 0$ and fixed step-sizes such that $\sqrt{\eta_1\eta_2} \leq \frac{2}{||A||}$, then their strategies are bounded away from the equilibrium $(\mathbf{0}, \mathbf{0})$. Formally,*

$$\frac{||x_1^t||^2}{\eta_1} + \frac{||x_2^t||^2}{\eta_2} \geq \frac{2 - \sqrt{\eta_1\eta_2}||A||}{2 + \sqrt{\eta_1\eta_2}||A||}\left(\frac{||x_1^0||^2}{\eta_1} + \frac{||x_2^0||^2}{\eta_2}\right).$$

Proof The proof follows identically to Theorem 5 after replacing the Cauchy-Schwarz inequality $-\langle x_1^t, Ax_2^t\rangle \leq ||x_1^t|| \cdot ||Ax_2^t||$ with the Cauchy-Schwarz inequality $\langle x_1^t, Ax_2^t\rangle \leq ||x_1^t|| \cdot ||Ax_2^t||$. By taking η_1 and η_2 sufficiently small and $||x_1^0||^2 + ||x_2^0||^2 > 0$, the right hand side of 10 is positive and (x_1^t, x_2^t) is bounded away from $(\mathbf{0}, \mathbf{0})$. ■

A.3. Proof of Theorem 7

To show this result, we make use of the following Theorem from (Rudin, 1987).

Theorem 8 (Rudin (1987) Theorem 7.26) *Let $\mathcal{X}$ be an open set in $\mathbb{R}^k$ and $T : \mathcal{X} \to \mathbb{R}^k$ be an injective differentiable function with continuous partial derivatives, the Jacobian of which is non-zero for every $x \in \mathcal{X}$. Then for any real-valued, compactly supported, continuous function f, with support contained in $T(\mathcal{X})$,*

$$\int_{T(\mathcal{X})} f(v)dv = \int_{\mathcal{X}} f(T(x))|det(J_T)(x)|dx. \tag{17}$$

In particular, taking $f(v) = 1$,

$$\int_{T(\mathcal{X})} dv = \int_{\mathcal{X}} |det(J_T)(x)|dx \tag{18}$$

and T is volume preserving if T is continuous differentiable, injective, and $|det(J_T)| = 1$.

Theorem 7 (Volume Preservation) *(AltGD) is volume preserving for any step-sizes and any measurable set of initial conditions.*

Proof (AltGD) can be written as the following two-stage update where agent 1 first updates her strategy:

$$\begin{aligned} x_1^{t+1/2} &= x_1^t + \eta_1 A x_2^t \\ x_2^{t+1/2} &= x_2^t \end{aligned} \tag{Stage 1}$$

followed by agent 2 updating his strategy:

$$\begin{aligned} x_1^{t+1} &= x_1^{t+1/2} \\ x_2^{t+1} &= x_2^{t+1/2} - \eta_2 A^\intercal x_1^{t+1/2}. \end{aligned} \tag{Stage 2}$$

The red circles in Figure 2 refer to (Stage 1) while the blue triangles refer to (Stage 2). To show (AltGD) is volume preserving, it suffices to show that (Stage 1) and (Stage 2) are volume preserving. Both arguments are identical and we show the result only for (Stage 1).

By Theorem 8, it suffices to show (Stage 1) is continuously differentiable, injective, and has a Jacobian with determinant equal to one. Trivially, (Stage 1) is continuously differentiable. Next, we show it is injective.

Suppose (y_1^t, y_2^t) map to the same $(x_1^{t+1/2}, x_2^{t+1/2})$ when updated with (Stage 1), i.e.,

$$x_1^{t+1/2} = x_1^t + Ax_2^t = y_1^t + Ay_2^t, \tag{19}$$

$$x_2^{t+1/2} = x_2^t = y_2^t. \tag{20}$$

Combining both equalities yields $(x_1^t, x_2^t) = (y_1^t, y_2^t)$ and (Stage 1) is injective.

Next, we show that the determinant of the Jacobian in (Stage 1) is one. The Jacobian is

$$J_1 = \begin{bmatrix} I_{k_1} & \eta_1 A \\ 0 & I_{k_2} \end{bmatrix} \tag{21}$$

and $\det(J_1) = \det(I_{k_1}) \cdot \det(I_{k_2}) = 1$. (Stage 1) satisfies all three conditions and therefore is volume-preserving thereby completing the proof of the theorem. ■