Stability and Exploitation in Differentiable Games

Alistair Letcher

Mathematical Institute

University of Oxford

A thesis submitted for the degree of

MSc in Mathematics and Foundations of Computer Science

Trinity 2018

Abstract

While machine learning has traditionally focused on optimising single loss functions, a growing number of algorithms deal with multiple interacting goals, from GANs to multi-agent RL. Naively transposing gradient descent has been shown to fail, while state-of-the-art proposals are tailored to specific applications (e.g. two-player zero-sum games) or lack strong theoretical guarantees. We provide a unified approach to this problem and prove local convergence of Consensus Optimisation, Symplectic Gradient Adjustment and lookahead in *all* differentiable games. Learning with Opponent-Learning Awareness (LOLA) takes a different approach by shaping and exploiting opponent learning, reaching better equilibria and outperforming previous methods. On the flip side, we construct the first example where this backlashes in self-play, producing arrogant behaviour and poor losses. We address and solve this catch-22 with a new algorithm named Stable Opponent Shaping (SOS), inheriting strong convergence guarantees from lookahead and shaping capacity from LOLA. We obtain convergence of LOLA to equilibria in two-player zero-sum and n-player cooperative games as a corollary.

Contents

1 Introduction **1**

2 Multi-Loss Optimisation **3**
2.1 Differentiable Games . . . 3
2.2 Stable Fixed Points versus Nash Equilibria . . . 8
2.2.1 Potential and Hamiltonian Games . . . 8
2.2.2 General Games . . . 10
2.3 Second-Order Gradient Methods . . . 15
2.3.1 Consensus Optimization . . . 15
2.3.2 Symplectic Gradient Adjustment . . . 16
2.3.3 Learning with Opponent-Learning Awareness . . . 18
2.4 Arrogance and Symmetry . . . 24
2.4.1 Killing the Shaping . . . 27
2.4.2 Higher-Order LOLA . . . 29

3 Theoretical Results **32**
3.1 Local Convergence . . . 32
3.2 Consensus Optimization . . . 34
3.3 Symplectic Gradient Adjustment . . . 35
3.4 Symmetric Lookahead . . . 37
3.5 Lookahead . . . 39
3.6 Non-Convergence . . . 43
3.7 Caveat . . . 46

4 Stable Opponent Shaping **48**
4.1 Partial LOLA . . . 48
4.2 SOS : Rescuing LOLA . . . 49
4.3 Theoretical Guarantees . . . 51

5 Experimental Results & Discussion **58**
5.1 Experimental Setup . . . 58
5.1.1 Iterated Bimatrix Games . . . 58
5.1.2 Dual Game . . . 60

5.1.3 Running Times 61
5.2 Results & Discussion 61
5.2.1 Iterated Bimatrix Games 62
5.2.2 Dual Game 65

6 Conclusion **67**

A Linear Algebra **69**

B Analysis and Topology **72**

Bibliography **77**

Chapter 1

Introduction

Much of supervised machine learning is based on optimisation of a single loss function, constructed from a given dataset and a chosen function approximator. A dataset $\mathcal{D} = (x^i, y^i)$ generically consists of n input-output pairs, where $x_i \in \mathcal{X}$ is an input vector representing an object of interest (say pixels of an image) and $y_i \in \mathcal{Y}$ is a label associated to this input (say 1 or 0 for 'dog' or 'not dog'). A function approximator $f_\theta : \mathcal{X} \to \mathcal{Y}$ is parametrised by a vector of numbers $\theta \in \mathbb{R}^d$, say the weights associated to each feature in a linear function, or the weights of nodes in a neural network. A *loss function* $L : \mathbb{R}^d \to \mathbb{R}$ is constructed to capture how close the approximator is from predicting the correct labels on each input. A standard example is the mean squared error

$$L(\theta) = \frac{1}{n} \sum_i \|y^i - f_\theta(x^i)\|^2 ,$$

measuring the difference between predicted and true labels in the dataset (possibly in expectation). The most common way to minimise this function is by *gradient descent* (GD), optimising the parameters by following the direction of steepest descent. More precisely, at each step we update the parameters as

$$\theta \leftarrow \theta - \alpha \nabla L$$

for some *learning rate* α. ∇L is the direction in $\mathbb{R}^d$ in which L increases most sharply, so following the opposite direction is guaranteed to decrease L for sufficiently small α. The success of deep learning has partially been enabled by the guarantee that gradient descent almost surely converges to minima if saddles are strict [Lee] [Pan], while the gradient ∇L is easily computed in a neural network by backpropagation through the layers. If non-strict saddles are present, they can still be avoided through a noisy estimate of the true gradient – also known as *stochastic* GD. Although local minima may be nowhere close to global ones, deep neural networks have been hugely successful and experimentally display good minima.

Nonetheless, this is an unrealistic paradigm in comparison to how humans learn, or how we might expect artificial intelligence to evolve. Human brains do not learn by optimising a single objective, but through a fine balancing act between multiple and highly dependent goals. In this light, a growing number of learning algorithms rely on optimising many losses at once: generative adversarial networks (GANs) [Goo], intrinsic curiosity [Pat], imaginative agents [Rac], synthetic gradients [Jad] and multi-agent reinforcement learning (RL) in general. The models are effectively trained as games played by competing modules.

Multi-loss optimisation can be viewed as a differentiable game, where each loss is associated to a 'player' or 'agent'. The naive approach is for each agent to optimise their loss simultaneously and independently, by gradient descent. The hope is to reach local 'minima' (Nash equilibria) in this way. Instead, circular behaviour can emerge and prevent convergence entirely, calling for a more involved approach. Even in the simple game of matching pennies, we will see that players diverge away from Nash for *any* learning rate.

The intuition behind failure is that each agent treats others as stationary at each learning step, thereby missing crucial information on the correct update. Each component we are trying to optimise is isolated from the rest, while they are highly depend on each other. In RL lingo, each agent treats others as part of the environment. This points towards a third perspective on multi-loss optimisation, where agents not only try to minimise their loss but also to influence and exploit others. In this light, [Foe] propose Learning with Opponent-Learning Awareness (LOLA). Each agent predicts the opponent's learning step and optimises under the resulting modified loss. In the case of two agents, agent 1 optimises

$$L^1(\theta^1, \theta^2 - \alpha_2 \nabla_{\theta^2} L^2)$$

with respect to θ^1, instead of $L^1(\theta^1, \theta^2)$. This accounts for nonstationarity by predicting opponent parameters after one step of gradient descent. In the case of matching pennies, this solves the problem and both agents converge to the Nash equilibrium. We will see that LOLA moreover shapes opponent learning by differentiating through their learning steps, encouraging cooperative behaviour where previously impossible.

Although appealing and experimentally successful, there are no guarantees that LOLA converges to equilibria. In Section 2.4, we present an example where LOLA fails drastically in self-play. LOLA does not preserve fixed points of the original game, and can converge to parameters producing worse losses for all agents involved. This stems partly from the assumption that other agents are naive, which is false in self-play and produces 'arrogant' behaviour. One aim of this report is to deal with this problem of asymmetry.

On the other hand, [Bal] propose a novel approach named Symplectic Gradient Adjustment (SGA), for which minor theoretical guarantees are known. We strengthen these through a unified analytical approach, producing strong convergence results for a number of algorithms including SGA. However, the method is not framed from the perspective of each agent, and can instead produce unselfish individual policies. SGA is concerned with convergence more than individual losses, making it less realistic for multi-agent RL. Finally, SGA differs from LOLA in failing to shape opponent learning. The main goal of this project is to find a middle ground between SGA's convergence properties and LOLA's exploitative capacities.

The second chapter introduces the required background on multi-loss optimisation and surveys existing learning schemes in detail. We show that LOLA is capable of exploitation while also prone to harmful ('arrogant') behaviour. Chapter 3 provides a unified framework for tackling local convergence, with novel proofs strengthening the theoretical guarantees of algorithms including SGA. We also investigate non-convergence to unstable / saddle points. The fourth chapter introduces our main contribution, a new algorithm named Stable Opponent Shaping (SOS), achieving both provable local convergence and shaping on par with LOLA. The final chapter displays this exploitative capacity with experimental results in a number of games.

Chapter 2

Multi-Loss Optimisation

Before we begin, note that 'minimising' and 'optimising' are used interchangeably in this report, despite the former applying to losses and the latter to parameters. For linguistic convenience, we also use 'stability' to mean 'convergence to good points, divergence from bad points, preservation of fixed points etc'. Unless stated otherwise, all proofs in this report are original. It will be mentioned explicitly if parts are inspired from another source, even if worked through independently.

2.1 Differentiable Games

We frame the problem of multi-loss optimisation as a game, where each player's goal is to minimise their individual loss. The following definition is adapted from [Bal], insisting only on differentiability beyond the standard notion from game theory. This condition is virtually always satisfied in machine learning, making the problem as general as possible.

Definition 2.1. A *differentiable game* is a collection of n players with parameters $\theta = (\theta^1, \dots, \theta^n) \in \mathbb{R}^d$ and twice continuously differentiable losses $L^i : \mathbb{R}^d \to \mathbb{R}$, where $\theta^i \in \mathbb{R}^{d_i}$ for each i and $\sum_i d_i = d$.

In game theory, parameters are often probabilities and each θ^i would be restricted to the probability simplex. We do not impose such a condition, though this may be recovered via sigmoid functions if necessary. From the viewpoint of player i, the parameters can be written as $\theta = (\theta^i, \theta^{-i})$, where θ^{-i} contains all other players' parameters. This is not consistent with player order, so one should be careful with this abuse of notation.

In a differentiable game, each player wants to minimise their loss. If $n = 1$, the 'game' is simply to minimise a given loss function. In this case one can reach local minima by (possibly stochastic) gradient descent, which is a *fixed point* of the game since the player cannot further minimise their loss locally. For arbitrary n, it is unlikely that some point $\bar{\theta}$ locally minimises each loss function simultaneously. Instead, the closest analogue and most widespread concept of 'solution' to the game is a *Nash equilibrium*.

Definition 2.2. A point $\bar{\theta} \in \mathbb{R}^d$ is a (local) Nash equilibrium if for each i, there is a neighbourhood U_i of θ^i such that

$$L^i(\theta^i, \bar{\theta}^{-i}) \geq L^i(\bar{\theta})$$

for all $\theta^i \in U_i$. In other words, each player cannot improve their losses locally if the other players' parameters are fixed. In game theory lingo, each player's strategy is a local *best response* to the other players'.

We will omit the word 'local' for convenience. If each neighbourhood U_i can be taken as $\mathbb{R}^{d_i}$, then $\bar{\theta}$ is a *global* Nash equilibrium. Note that Nash equilibria may not exist, just as the function $f(x) = x$ has no local minima. If they do exist, we can only expect to reach local Nash just as we cannot expect to find the global minimum of a function by gradient descent. For convenience we write

$$\nabla_i L^k = \nabla_{\theta^i} L^k \qquad \text{and} \qquad \nabla_{ij} L^k = \nabla_{\theta^j} \nabla_{\theta^i} L^k$$

for any i, j, k. Note that $\nabla_{ij} L^k$ means 'first gradient with respect to θ^i, then with respect to θ^j'. Define the *simultaneous gradient* of the game as the concatenation of each player's gradient,

$$\xi = \begin{pmatrix} \nabla_1 L^1 \\ \vdots \\ \nabla_n L^n \end{pmatrix} \in \mathbb{R}^d \,.$$

The ith component of ξ is the direction of greatest increase in L^i with respect to θ^i. Often ξ will be written as a row vector for the sake of spacing, but really is defined as a column vector. If each agent minimises their loss independently, they perform simultaneous GD on their component $\nabla_i L^i$ with a learning rate α_i. This is also called naive learning (NL). Hence the overall game parameters θ follow the opposite of ξ:

$$\theta \leftarrow \theta - \alpha \circ \xi$$

where $\alpha = (\alpha_1, \ldots, \alpha_n)^\intercal$ and $\circ$ is element-wise multiplication. This reduces to

$$\theta \leftarrow \theta - \alpha\xi$$

if all agents have the same learning rate. We will always assume this for notational simplicity, though any result in this report applies to the general case. Before giving a failure example of NL, we provide a useful sufficient condition for Nash, in terms of simultaneous and higher-order gradients.

Proposition 2.3. Assume $\xi(\bar{\theta}) = 0$ and for each i, $\nabla_{ii} L^i(\theta) \succeq 0$ for all θ in a neighbourhood of $\bar{\theta}$. Then $\bar{\theta}$ is a Nash equilibrium.

This is implicitly assumed in [Bal, Lemma 2]. We have not found a proof in the literature, and provide our own below.

Proof. Assume $\xi(\bar{\theta}) = 0$ and $\nabla_{ii} L^i(\theta) \succeq 0$ for all θ in a neighbourhood U of $\bar{\theta}$. In particular there is a neighbourhood U_i of $\bar{\theta}^i$ such that $\nabla_{ii} L^i(\theta^i, \bar{\theta}^{-i}) \succeq 0$ for each i. By Taylor's theorem with Lagrange remainder in many variables [Fol], we have

$$\begin{aligned} L(\theta^i, \bar{\theta}^{-i}) &= L^i(\bar{\theta}) + \nabla_i L^i(\bar{\theta})^\intercal (\theta^i - \bar{\theta}^i) + \frac{1}{2}(\theta^i - \bar{\theta}^i)^\intercal \nabla_{ii} L^i(\nu, \bar{\theta}^{-i})(\theta^i - \bar{\theta}^i) \\ &= L^i(\bar{\theta}) + \frac{1}{2}(\theta^i - \bar{\theta}^i)^\intercal \nabla_{ii} L^i(\nu, \bar{\theta}^{-i})(\theta^i - \bar{\theta}^i) \end{aligned}$$

for some $\nu \in U_i$. By assumption of positive semi-definiteness we obtain

$$L(\theta^i, \bar{\theta}^{-i}) = L^i(\bar{\theta}) + \frac{1}{2}(\theta^i - \bar{\theta}^i)^\intercal \nabla_{ii} L^i(\nu, \bar{\theta}^{-i})(\theta^i - \bar{\theta}^i) \geq L^i(\bar{\theta})$$

for all $\theta^i \in U_i$ and each i, so $\bar{\theta}$ is a Nash equilibrium. □

Remark 2.4. The converse does not quite hold. It is true that a Nash equilibrium must satisfy $\xi(\bar{\theta}) = 0$ and $\nabla_{ii} L^i(\bar{\theta}) \succeq 0$ for each i, see Proposition B.1. However positive semi-definiteness may not hold in a neighbourhood of Nash, even for single losses. For instance,

$$L(x, y) = x^2 y^2$$

has a local minimum at $(0, 0)$ with Hessian

$$H = \nabla^2 L = 2 \begin{pmatrix} y^2 & 2xy \\ 2xy & x^2 \end{pmatrix}$$

which is trivially positive semi-definite at $(0, 0)$, but not in any neighbourhood. Indeed any such neighbourhood contains (ϵ, ϵ) for some $\epsilon > 0$, where the Hessian has determinant

$$\det(H) = 2(\epsilon^4 - 4\epsilon^4) = -6\epsilon^4 < 0\,.$$

Since $\det(H) = \sigma_1 \sigma_2$ where σ_k are the eigenvalues of H, exactly one of them is negative and so $H(\epsilon, \epsilon)$ is not positive semi-definite by Proposition A.7.

Remark 2.5. On the other hand, it is not enough to assume only the weaker condition that $\nabla_{ii} L^i(\bar{\theta}) \succeq 0$ for each i. For single losses, this is known as inconclusiveness of the second partial derivative test. For instance,

$$L(x, y) = x^3$$

gives $\nabla L = 3x^2$ with a single fixed point at the origin, while

$$H = \nabla^2 L = 6x$$

which is positive semi-definite (i.e. non-negative) at $x = 0$. However the point is not a Nash equilibrium (local minimum) since any neighbourhood intersects $x = -\epsilon$ for some $\epsilon > 0$, where

$$L(-\epsilon) = -\epsilon^3 < 0 = L(0)\,.$$

This failure arises precisely because $\nabla^2 L \not\succeq 0$ in any neighbourhood of the origin. Combining this remark with the previous, there is no necessary *and* sufficient characterisation of Nash equilibria in terms of first- and second-order gradients. This is well-known even for local minima of single losses.

Remark 2.6. A point $\bar{\theta}$ with $\xi(\bar{\theta}) = 0$ and $\nabla_{ii} L^i(\bar{\theta}) \succ 0$ for each i is a Nash equilibrium since $\nabla_{ii} L^i(\theta) \succ 0$ in a neighbourhood, by continuity. This corresponds to the well-known second partial derivative test. The converse trivially fails, since any point of

$$L \equiv 0$$

is a Nash equilibrium, but no point has positive definite Hessian.

As mentioned in the previous section, NL can fail to converge to Nash. The following example displays this precisely, even in the simple case of a two-player, two-parameter, zero-sum game.

Example 2.7 (Cyclic game)**.** Consider the game given by

$$L^1(x, y) = xy \qquad \text{and} \qquad L^2(x, y) = -xy$$

where players 1 and 2 control the x and y parameters respectively. Since the losses sum to 0, we cannot have optimal values for both players simultaneously. The simultaneous gradient is

$$\xi = (y, -x)\,,$$

with $\xi = 0$ only at $(0, 0)$. It is a Nash equilibrium since the second-order derivatives are $\nabla_{11}L^1 = \nabla_{22}L^2 = 0 \geq 0$ everywhere. There are no other equilibria. However the vector field ξ can be seen to cycle around this point, as displayed in Figure 2.1.

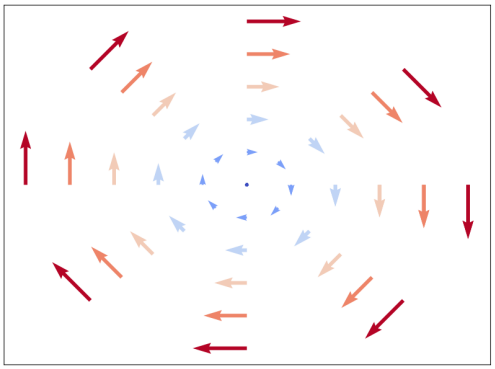

Figure 2.1: Plot of the vector field $\xi = (y, -x)$.

It follows that simultaneous gradient descent will always fail to converge. For any positive learning rate $\alpha > 0$, the agents will overshoot and move further away from the centre. Even for infinitesimal α, they would stay on a circle of fixed radius around the origin. To see this, consider the *Hamiltonian*

$$\mathcal{H} := \frac{1}{2}\|\xi\|^2 = \frac{1}{2}(x^2 + y^2)\,.$$

We have $\nabla\mathcal{H} = (x, y)$ and thus

$$\langle \xi, \nabla\mathcal{H} \rangle = xy - yx = 0\,,$$

so ξ preserves the level sets of $\mathcal{H}$. In other words, ξ is orthogonal to the direction in which $\|\xi\|$ increases or decreases, so taking infinitesimal steps along ξ will cycle around $(0, 0)$. More explicitly, for any current parameters (x, y), a step of naive learning yields

$$(x, y) \leftarrow (x, y) - \alpha(y, -x) = (x - \alpha y, y + \alpha x)$$

which has distance from the origin

$$(x^2 - 2\alpha xy + \alpha^2 y^2) + (y^2 + 2\alpha xy + \alpha^2 x^2) = (1 + \alpha^2)(x^2 + y^2) > (x^2 + y^2)$$

for any $\alpha > 0$ and $(x, y) \neq 0$. We conclude that agents diverge away from the origin for any $\alpha > 0$. Note however that gradient descent on the Hamiltonian converges to Nash, since $\mathcal{H} = (x^2 + y^2)$ is convex with global minimum at $(0, 0)$. This will be the motivation behind Consensus Optimization in Section 2.3.

The 'cyclic game' corresponds secretly to the game of matching pennies, where two players choose heads or tails on their respective pennies. The first player's goal is to match their opponent's penny, and the second player's goal is to differ. They win or lose 1 unit depending on the payoff matrix in Table 2.1. If p_1 and p_2 are the probabilities of selecting heads for each of the players, the losses in this game are given by

$$L^1 = \begin{pmatrix} p_1 & 1 - p_1 \end{pmatrix} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} p_2 \\ 1 - p_2 \end{pmatrix} = -L^2 \,.$$

Changing variables through $p_1 = (1 + x)/2$ and $p_2 = (1 + y)/2$, we recover

$$\begin{aligned} L^1 &= \frac{1}{4} \begin{pmatrix} 1 + x & 1 - x \end{pmatrix} \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} \begin{pmatrix} 1 + y \\ 1 - y \end{pmatrix} \\ &= \frac{1}{4} \begin{pmatrix} 1 + x & 1 - x \end{pmatrix} \begin{pmatrix} 2y \\ 2y \end{pmatrix} = xy \end{aligned}$$

and $L^2 = -L^1 = -xy$ as required. Matching pennies is one of the most basic settings in game theory, essentially a two-action version of rock-paper-scissors. The failure of convergence in this simple case illustrates that NL is not a suitable learning technique in general. Before moving on to second-order gradient methods to address this, we consider an alternative solution concept named *Stable Fixed Points* (SFP).

	Heads	Tails
Heads	$(1, -1)$	$(-1, 1)$
Tails	$(-1, 1)$	$(1, -1)$

Table 2.1: Payoff matrix for players $(1, 2)$ in Matching Pennies

2.2 Stable Fixed Points versus Nash Equilibria

First define the *Hessian* of the game as the block matrix

$$H = \nabla\xi = \begin{pmatrix} \nabla_{11}L^1 & \cdots & \nabla_{1n}L^1 \\ \vdots & \cdots & \vdots \\ \nabla_{n1}L^n & \cdots & \nabla_{nn}L^n \end{pmatrix} \in \mathbb{R}^{d\times d}\,.$$

This can equivalently be viewed as the Jacobian of the vector field ξ. Note crucially that H is not always symmetric unless $n = 1$, in which case we recover the usual Hessian

$$H = \nabla^2 L\,.$$

Its diagonal entries are $\nabla_{ii}L^i$, which are relevant to Nash equilibria by Proposition 2.3. Its symmetric and antimmetric parts are defined as

$$S = \frac{1}{2}(H + H^\intercal) \qquad \text{and} \qquad A = \frac{1}{2}(H - H^\intercal)$$

respectively, so that $H = S + A$.

2.2.1 Potential and Hamiltonian Games

Potential Games: A game is called *potential* if there is no antisymmetric part, namely $A \equiv 0$. To appreciate this concept, recall Example 2.7 where NL cycles around the equilibrium. The cause of failure is that ξ is not the gradient of a single function, implying that each agent's loss is inherently dependent on others. This results in a contradiction between the non-stationarity of each agent, and the optimisation of each loss independently from others. Potential games are precisely the class where there exists an underlying 'potential' function $\phi : \mathbb{R}^d \to \mathbb{R}$ such that

$$\nabla_i L^i = \nabla_i \phi\,,$$

so that $\xi = \nabla\phi$ is the gradient of a single function. This is the definition of *exact* potential games as introduced by [Mon], which is equivalent to

$$\nabla_{ij}L^i = \nabla_{ij}\phi = (\nabla_{ji}\phi)^\intercal = (\nabla_{ji}L^j)^\intercal$$

for all i, j. Finally this is equivalent to

$$\nabla_{ij}L^i - (\nabla_{ji}L^j)^\intercal = 0 = H_{ij} - H_{ij}^\intercal$$

for all i, j, namely $A \equiv 0$ as in our definition. The existence of a potential function lifts multi-loss optimisation into gradient descent on a single function, which is well-understood. More precisely, GD on $\xi = \nabla\phi$ converges locally fixed points that are either local minima or saddles of ϕ. Gradient descent almost always avoids strict saddle points by [Lee] [Pan], so convergence to local minima of ϕ is guaranteed under mild assumptions. We show that local minima of ϕ are Nash equilibria as follows. [Mon] show that potential games are equivalently defined as

$$\phi(\theta_1^i, \theta^{-i}) - \phi(\theta_2^i, \theta^{-i}) = L^1(\theta_1^i, \theta^{-i}) - L^1(\theta_2^i, \theta^{-i})$$

for all i and $\theta_1^i, \theta_2^i, \theta^{-i}$. Now if $\bar{\theta}$ is a local minima of ϕ we have

$$\phi(\bar{\theta}^i, \bar{\theta}^{-i}) \leq \phi(\theta^i, \bar{\theta}^{-i})$$

for all θ^i in a neighbourhood of $\bar{\theta}^i$, and hence

$$L^i(\bar{\theta}^i, \bar{\theta}^{-i}) \leq {}^i(\theta^i, \bar{\theta}^{-i})$$

also. Thus $\bar{\theta}$ is a Nash equilibria, and convergence is well-understood in potential games.

Hamiltonian Games: On the opposite end of the spectrum, a game is called *Hamiltonian* if there is no symmetric part, namely $S \equiv 0$. This was the case of Example 2.7, where we introduced the Hamiltonian

$$\mathcal{H} := \frac{1}{2}\|\xi\|^2 .$$

It was shown in this example that gradient descent on $\mathcal{H}$ converges to Nash, and that $\mathcal{H}$ preserves the level sets of ξ. This is true of all Hamiltonian games, thus the name, by [Bal, Th. 3] reproduced below.

Theorem 2.8. If a game is Hamiltonian then (i) $\nabla\mathcal{H} = A^\intercal\xi$ and (ii) ξ preserves the level sets of $\mathcal{H}$. If the Hessian is invertible and $\mathcal{H} \to \infty$ as $\|\theta\| \to \infty$ then (iii) gradient descent on $\mathcal{H}$ converges to a Nash equilibrium.

We provide a proof in our own words below, following [Bal] with more detail.

Proof. The first claim is trivial since $S \equiv 0$ implies

$$\nabla\mathcal{H} = \frac{1}{2}\nabla(\xi^\intercal\xi) = (\nabla\xi)^\intercal\xi = H^\intercal\xi = A^\intercal\xi .$$

Preservation of the level sets follows from antisymmetry of A:

$$\xi^\intercal A^\intercal\xi = (\xi^\intercal A^\intercal\xi)^\intercal = \xi^\intercal A\xi = -\xi^\intercal A^\intercal\xi$$

implying

$$\langle\xi, \nabla\mathcal{H}\rangle = \xi^\intercal A^\intercal\xi = 0 .$$

Finally, gradient descent on a function $\mathcal{H}$ converges to a point $\bar{\theta}$ such that $\nabla\mathcal{H} = 0$, if $\mathcal{H} \to \infty$ as $\|\theta\| \to \infty$. Then $H^\intercal\xi(\bar{\theta}) = 0$ and by invertibility, $\xi(\bar{\theta}) = 0$. Now $S \equiv 0$ implies in particular that $\nabla_{ii}L^i = 0$ everywhere, which is positive semi-definite. Hence $\bar{\theta}$ is a Nash equilibrium by Proposition 2.3. □

We conclude that both potential and Hamiltonian games are well-understood and essentially solved. The difficult part is to extend such convergence results to general games, where S and A are both non-trivial.

2.2.2 General Games

In Hamiltonian games, we are guaranteed convergence to Nash under mild assumptions. Under similar conditions, we also have convergence to local minima of the potential function ϕ in potential games, which are Nash equilibria. Note however that local convergence to *all* Nash equilibria is not guaranteed, since not all Nash are minima of ϕ. This will be displayed in Example 2.15 explicitly, where NL actually *diverges* away from Nash for any learning rate and any neighbourhood.

As such, even in the simple class of potential games, local convergence to all Nash fails for naive learning. Higher-order methods involving the Hessian $H = \nabla\xi = \nabla^2\phi$ are also likely to fail, since the game is governed by ϕ whose local minima are the relevant points – not Nash equilibria. The following example shows moreover that local convergence to *all* Nash would be undesirable, even in potential games.

Example 2.9. Consider the simple potential game given by

$$L^1 = L^2 = xy = \phi$$

where players control the x and y parameters respectively. The optimal solution is $(x, y) \to \pm(\infty, -\infty)$, since then $L^1 = L^2 \to -\infty$. However the origin $(0, 0)$ is a global Nash equilibrium since $L^1(x, 0) = 0 \geq L^1(0, 0)$ and $L^2(0, y) = 0 \geq L^1(0, 0)$ for all $x, y \in \mathbb{R}$. It is undesirable to converge to Nash in this game, since infinitely better losses can be reached by following the anti-diagonal direction.

In general games, it is all the more impossible/undesirable to prove local convergence to *all* Nash. Instead, the aim is to prove local convergence to a subclass captured by ϕ. Stable fixed points were introduced by [Bal] and correspond to local minima of ϕ, recalling that such minima satisfy $\xi = \nabla\phi = 0$ and $H = \nabla^2\phi \succeq 0$. We impose the slightly stronger condition of $H \succeq 0$ in a *neighbourhood* to ensure that SFP are a subset of Nash, though either definition has its pros and cons. For instance, local convergence is proved using only the weaker condition, so the strengthened version is superficial. We discussed this with [Bal] and keep the definition intact for now, though may alter it in future work.

Definition 2.10. A point $\bar{\theta}$ is a *fixed point* if $\xi(\bar{\theta}) = 0$. It is *stable* if $H(\theta) \succeq 0$ for all θ in a neighbourhood of $\bar{\theta}$, *unstable* if $H(\bar{\theta}) \prec 0$ and a *strict saddle* if $H(\bar{\theta})$ has a negative eigenvalue.

The name 'fixed point' is in line with naive gradient descent, since $\xi(\bar{\theta}) = 0$ implies an update

$$\bar{\theta} \leftarrow \bar{\theta} - \alpha\xi(\bar{\theta}) = \bar{\theta}$$

which stays fixed. In potential games, we have $H = \nabla\xi = \nabla^2\phi$ so stable fixed points are local minima of ϕ. Note that local convergence of gradient descent on single functions can only be guaranteed for points such that $H \succeq 0$ since they are strict saddles otherwise, which are almost always avoided by [Lee] [Pan]. It is thus reasonable to prove local convergence only to their multi-loss counterpart, namely SFP.

Finally note that $H(\bar{\theta}) \succ 0$ implies $H(\theta) \succ 0$ in a neighbourhood, hence being equivalent to the definition in [Bal]. Also recall from Definition A.5 that a complex eigenvalue is called negative if its real part is negative.

It follows that unstable points are a subset of strict saddles: if $H(\bar{\theta}) \prec 0$ then all eigenvalues are negative since any eigenpair (v, λ) satisfies

$$0 > \mathrm{Re}(v^\intercal H v) = \mathrm{Re}(\lambda v^\intercal v) = \mathrm{Re}(\lambda)\,.$$

We introduce strict saddles in this report as a generalisation of unstable FP, for which we can prove identical results regarding non-convergence. The name is chosen to be identical for single losses as defined in [Lee].

Remark 2.11. Proposition A.4 shows that a matrix H is positive semi-definite iff its symmetric part S is, so the definition of stability above could equivalently be stated as $S(\theta) \succeq 0$. This is the original formulation given in [Bal], but we find our formulation intuitively closer to the well-known connections between Hessian and local minima of single functions.

Remark 2.12. Throughout the report we will see that assuming invertibility of $H(\bar{\theta})$ is central to convergence results. The same assumption is present both throughout [Bal] and [Mes], though the latter forgets to specify in Corollary 8 that local convergence to Nash only holds if $H(\bar{\theta})$ invertible. We make it explicit in this remark that all results on local convergence to SFP $\bar{\theta}$ assume invertibility of $H(\bar{\theta})$, omitted from now on. On the contrary, our non-convergence results apply equally to singular H.

Proposition 2.13. A stable fixed point is a Nash equilibrium.

Proof. If $H(\theta)$ is positive semi-definite in a neighbourhood then so are its diagonal blocks $\nabla_{ii} L^i(\theta)$, so we are done by Proposition 2.3. □

Proposition 2.14. The converse holds in Hamiltonian games.

Proof. A Nash equilibria is a fixed point, which is always stable in a Hamiltonian game since $S \equiv 0 \succeq 0$. □

The original version of [Bal, Lemma 2] claimed that the converse *always* holds, which is untrue since a matrix with positive semi-definite diagonal blocks may not be positive semi-definite. We pointed this out to the authors along with the counter-example below, and a correction was made in the final version of the paper. He kindly acknowledged us and included this example.

Example 2.15. It is enough to consider Example 2.9 above, but we give a variant whose Nash equilibrium is strict (strict inequality in the definition) to display a stronger case. Consider the game given by

$$\begin{aligned} L^1 &= x^2/2 + 2xy \\ L^2 &= y^2/2 + 2xy\,. \end{aligned}$$

The gradient and Hessian are given by

$$\xi = \begin{pmatrix} x + 2y \\ y + 2x \end{pmatrix} \qquad \text{and} \qquad H = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}.$$

Note that the game is potential since we have $A \equiv 0$, with potential function

$$\phi = x^2/2 + 2xy + y^2/2\,.$$

The only fixed point is given by $x = y = 0$, which is a strict Nash equilibrium since

$$L^1(x, 0) = x^2/2 > 0 = L^1(0, 0)$$

for all $x \neq 0$ and similarly for y. On the other hand, $(0, 0)$ is *not* a stable fixed point since $S = H$ has eigenvalues 3 and -1:

$$\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \end{pmatrix} = 3 \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \text{and} \qquad \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ -1 \end{pmatrix} = - \begin{pmatrix} 1 \\ -1 \end{pmatrix} .$$

Moreover, $(0, 0)$ is not a local minimum of ϕ despite being a Nash equilibrium, since

$$\phi(\epsilon, -\epsilon) = -\epsilon^2 < 0 = \phi(0, 0) .$$

Both players can reach better losses by following the anti-diagonal direction $(x, -x)$, where the losses are

$$L^1 = L^2 = x^2/2 - 2x^2 = -3x^2/2 < 0 .$$

This is actually what happens if we follow NL, as is intuitively clear from the plots of L^1 and L^2 in Figure 2.2. We will prove this formally in the next chapter.

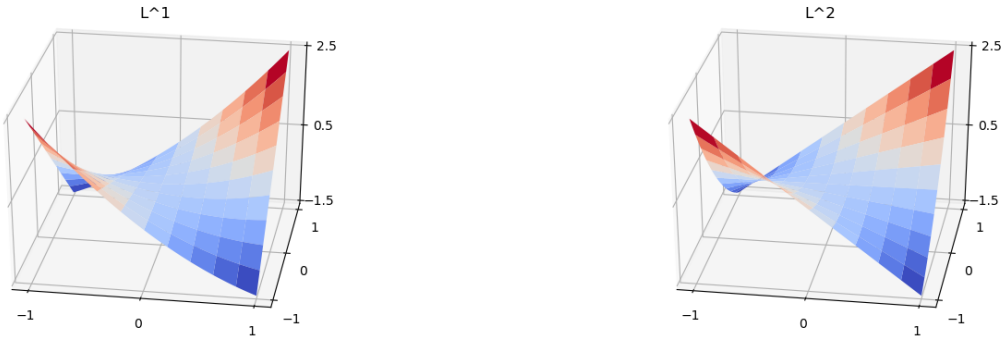

Figure 2.2: Plots of L^1 and L^2.

Appendix A of [Bal] was revised to mention briefly that Nash cannot be the right concept in light of this example, and that SFP are preferable. Nonetheless there is no guarantee that SFP is a better sub-class in general. To establish this, one would need to show that all non-SFP Nash equilibria are undesirable. Unfortunately this does not hold, by adding a simple term to the previous example.

Example 2.16 (Good non-SFP)**.** Consider

$$L^1 = x^2/2 + 2xy + 2y^2$$
$$L^2 = y^2/2 + 2xy + 2x^2 .$$

The dynamics will be identical to the previous example, since the extra terms $2y^2$ and $2x^2$ are functions of opponent parameters only. In other words, all gradients wrt to one's own parameters are identical. It follows that any algorithm based only on gradients wrt our own parameters will behave likewise in both examples. In particular NL diverges in the direction $(x, -x)$, where

$$L^1 = x^2/2 = L^2 > 0$$

and thus reach infinite losses as they learn, which is the worst possible behaviour. We will see in Example 3.20 that this even occurs for Symplectic Gradient Adjustment, an important caveat to keep in mind. The silver lining is that each algorithm fails only because there are terms in each loss function that are purely functions of other players' parameters. They are beyond each player's control entirely, a pathological setting where I can increase my opponent's loss with no possible counter-play.

Before moving on, we point out a mistake in [Bal]. It is claimed in Lemma 8 that in a two-player zero-sum game, Nash equilibria are equivalent to SFP. It is true that $H(\bar{\theta}) \succeq 0$ for any Nash equilibria $\bar{\theta}$ in such games, but this fails to imply that $H \succeq 0$ in a neighbourhood. This was pointed out in Remark 2.4, but we extend the counter-example for two-player zero-sum games below.

Example 2.17. Consider

$$L^1(x, y) = x^2 y^2 \qquad \text{and} \qquad L^2(x, y) = -x^2 y^2$$

with a Nash equilibrium at $(0, 0)$, since

$$L^1(x, 0) = 0 \geq L^1(0, 0) \qquad \text{and} \qquad L^2(0, y) = 0 \geq L^2(0, 0)\,.$$

Now

$$H = 2\begin{pmatrix} y^2 & 2xy \\ -2xy & -x^2 \end{pmatrix} \qquad \text{and} \qquad S = 2\begin{pmatrix} y^2 & 0 \\ 0 & -x^2 \end{pmatrix}$$

which is positive semi-definite at $(0, 0)$, but not in any neighbourhood since S has an eigenvalue $-x^2 < 0$. Hence $(0, 0)$ is a non-SFP Nash equilibrium, in a two-player zero-sum game.

If neither Nash equilibria nor SFP are the right solution concepts, what is? We propose that of Nash equilibria $\bar{\theta}$ robust to small perturbations. More precisely, the players cannot move together to a nearby point which is better for all players. This is a local version of *strong Nash equilibria*, as defined for instance in [Nes, Def. 2.1]. We formulate this in our own words below.

Definition 2.18. A (local) *strong Nash equilibrium* is a Nash equilibrium $\bar{\theta}$ with a neighbourhood U such that for all $\theta \in U$,

$$L^i(\theta) \geq L^i(\bar{\theta})$$

for some player i. Equivalently, there exists no $\theta \in U$ such that

$$L^i(\theta) < L^i(\bar{\theta})$$

for all i. We omit the word 'local' for convenience.

Strong Nash are sensitive to the presence of pure opponent terms in each agent's loss, thus distinguishing between Examples 2.15 and 2.16 above. On the contrary, SFP is agnostic to them since ξ discards pure functions of opponent parameters when taking gradients wrt to our own, and so does H. More precisely, $(0,0)$ is a non-SFP Nash in both examples. It is not a strong Nash in the first since any neighbourhood includes $(\epsilon, -\epsilon)$ for some $\epsilon > 0$, for which

$$L^1 = L^2 = -3\epsilon^2/2 < 0 = L^1(0,0) = L^2(0,0)\,.$$

On the other hand, it is a strong Nash in the second since $(0,0)$ is a global minimum of each loss function. This is precisely the behaviour we are looking for in the 'ideal' solution concept: the first equilibrium is poor (moving away produces better losses) while the second is desirable (a local minimum of each loss).

Note that strong Nash and SFP are not subsets of each other. Example 2.16 shows that not all strong Nash are SFP. For the converse, consider

$$\begin{aligned} L^1 &= 2x^2 + 3xy \\ L^2 &= 2y^2 + 3xy\,. \end{aligned}$$

with

$$\xi = \begin{pmatrix} 4x + 3y \\ 4y + 3x \end{pmatrix}$$

and

$$H = S = \begin{pmatrix} 4 & 3 \\ 3 & 4 \end{pmatrix} \succ 0$$

since $4 > 0$ and $\det(S) > 0$ (Sylvester's criterion). Then $(0,0)$ is an SFP (with invertible Hessian), for which $L^1 = L^2 = 0$. However it is not a strong Nash since any neighbourhood includes $(\epsilon, -\epsilon)$ for some $\epsilon > 0$, at which

$$L^1 = L^2 = -\epsilon^2/2 < 0 = L^1(0,0) = L^2(0,0)\,.$$

A more detailed understanding of strong Nash is left for future work, not being central to our thesis. Note only for now that they appear to be a better/ideal solution concept, as exemplified by the correct characterisation of the equilibria in 2.15 and 2.16. On the other hand, they cannot be classified only through ξ and higher-order terms including $H = \nabla\xi$, as explained above. In particular it is impossible to prove convergence of algorithms to strong Nash if the method is based only on such terms. They are thus intractable for those presented downstream; we hope to find better methods to address this problem in the future.

Despite being imperfect, SFP is a class fully characterised by ξ and H. This is central to the development of (non-)convergence results in Chapter 3.

2.3 Second-Order Gradient Methods

Example 2.7 demonstrates that circular behaviour can emerge, since both agents fail to take into account their opponent's parameter adjustment at each step. We present two methods developed to address this issue, each making use of second-order gradients of the loss functions.

2.3.1 Consensus Optimization

Recall the Hessian of the game

$$H = \nabla\xi = \begin{pmatrix} \nabla_{11}L^1 & \cdots & \nabla_{1n}L^1 \\ \vdots & \cdots & \vdots \\ \nabla_{n1}L^n & \cdots & \nabla_{nn}L^n \end{pmatrix} \in \mathbb{R}^{d\times d}$$

and notice that the Hamiltonian

$$\mathcal{H} = \frac{1}{2}\|\xi\|^2$$

has gradient

$$\nabla\mathcal{H} = \frac{1}{2}\nabla(\xi^\intercal\xi) = (\nabla\xi)^\intercal\xi = H^\intercal\xi\,.$$

As suggested in Example 2.7 and Section 2.2.1, gradient descent on the *single* function $\mathcal{H}$ may help convergence to Nash by minimising the size of ξ. Indeed, if $\mathcal{H}\to\infty$ as $\|\theta\|\to\infty$ then GD converges to a point $\bar\theta$ such that

$$\nabla\mathcal{H}(\bar\theta) = 0 = H^\intercal(\bar\theta)\xi(\bar\theta)\,.$$

If the Hessian is invertible at this point, this implies $\xi(\bar\theta)=0$ and so $\bar\theta$ is a fixed point of the game. However there is no guarantee that $\bar\theta$ is a *stable* fixed point, so gradient descent on the Hamiltonian is not a general solution. This approach can even fail for optimisation of a single loss.

Example 2.19. Consider the 'game' given by

$$L(x) = \frac{-x^2}{2}\,.$$

This is a trivial optimisation problem, and any reasonable algorithm should take $x\to\pm\infty$ to reach $L\to-\infty$. However GD on the Hamiltonian

$$\mathcal{H} = \frac{1}{2}\|\xi\|^2 = \frac{1}{2}\|\nabla L\|^2 = \frac{1}{2}x^2$$

converges to $x=0$, the global minimum of $\mathcal{H}$. This is the global *maximum* of L!

Instead, [Mes] propose Consensus Optimization (CO), a gradient adjustment of the form

$$\text{CO} = \xi + \gamma\nabla\mathcal{H} = (I+\gamma H^\intercal)\xi$$

for some parameter $\gamma>0$. In other words, each agent i performs gradient descent on the modified loss

$$L^i + \gamma\mathcal{H}$$

instead of L^i only. This encourages cooperation between the agents by minimising the size of ξ *as well as* the individual losses. Note that this algorithm is not necessarily selfish, since a player may sacrifice their immediate loss in favour of the second term. Moreover, γ may have to be chosen extremely small to avoid convergence towards unstable fixed points.

Example 2.20. We reproduce [Bal, Ex. 5] with more detail. Consider the game given by

$$L^1(x,y) = L^2(x,y) = -\kappa/2(x^2+y^2)$$

with κ very large. We have

$$\xi = -\kappa \begin{pmatrix} x \\ y \end{pmatrix} \qquad \text{and} \qquad H = S = -\kappa I\,.$$

The only fixed point is $(0,0)$, which is unstable since $S \prec 0$ everywhere. The desirable behaviour would be to diverge away from this global maximum. Now

$$\text{CO} = (I + \gamma H^\intercal)\xi = (1-\gamma\kappa)\xi = \kappa(\gamma\kappa - 1)\begin{pmatrix} x \\ y \end{pmatrix}\,,$$

so for $\gamma > 1/\kappa$ the gradient adjustment points in the direction (x,y). Gradient descent updates parameters by following the negative of this direction, so CO converges to $(0,0)$ if $\gamma > 1/\kappa$. In other words, γ needs to be chosen very small if we are to avoid the unstable fixed point.

Nonetheless, CO achieves local convergence to stable fixed points for any $\gamma > 0$ and small enough learning rate α. This was only proven in [Mes] for two-player zero-sum games, and we extend this for n-player general-sum games in Section 3.2. For sufficiently small γ, we also establish non-convergence from unstable fixed points in Section 3.6.

2.3.2 Symplectic Gradient Adjustment

More recently, Symplectic Gradient Adjustment (SGA) [Bal] takes a geometric approach improving on CO in terms of parameter flexibility. Recall the symmetric and antisymmetric parts of H as

$$S = \frac{1}{2}(H + H^\intercal) \qquad \text{and} \qquad A = \frac{1}{2}(H - H^\intercal)$$

respectively, then SGA is given by

$$\text{SGA} = \xi + \lambda A^\intercal \xi = (I + \lambda A^\intercal)\xi\,.$$

The parameter $\lambda \in \mathbb{R}$ is chosen according to a criterion specified further on. According to [Bal, App. E], the "antisymmetric matrix A captures the infinitesimal tendency of ξ to rotate at each point". This is directed specifically at naive learning's circular behaviour, while CO is agnostic to the underlying geometry

in minimising $\|\xi\|$. SGA and CO are identical iff the game is Hamiltonian, since $H \equiv A$ iff $S \equiv 0$. More details on interpreting A can be found in [Bal], but the essential fact is that

$$A = d\xi$$

in coordinates (up to musical isomorphism), where $d\xi$ is the differential of ξ. Introducing differential forms in detail is beyond this report, but d is isomorphic to curl for three-dimensional vector fields (through the Hodge star operator). Hence A corresponds to $\mathrm{curl}(\xi)$, which represents the infinitesimal rotations of ξ.

While curl only exists in three dimensions, this argument is somewhat generalised by noticing that the Lie algebra of infinitesimal rotations is given by antisymmetric matrices. As noted by Wikipedia's article on curl, this "allows one to interpret the differential of a 1-vector field as its infinitesimal rotations". Since $A = d\xi$, this helps understand the claim in [Bal]. The paper does not explain why multiplying A with ξ then gives the correct quantity, but a nice answer can be found in [Gem] for three dimensions. Indeed, a standard result on cross products is that

$$v \times \mathrm{curl}(F) = \nabla_F (F \cdot v) - (v \cdot \nabla) F$$

where ∇_F treats v as constant, so that

$$v \times \mathrm{curl}(F) = (\nabla F)^\intercal v - \sum_i v_i \nabla_i F \,.$$

Applying this to $F = v = \xi$, we obtain

$$\xi \times \mathrm{curl}(\xi) = H^\intercal \xi - \sum_i \xi_i \nabla_i \xi$$

where the jth entry of the RHS term is

$$\sum_i \xi_i \nabla_i \xi_j = \sum_i \nabla_{ji} L^j \xi_i = (H\xi)_j \,.$$

Hence we obtain

$$\xi \times \mathrm{curl}(\xi) = H^\intercal \xi - H\xi = 2A^\intercal \xi \,,$$

so that $A^\intercal \xi$ is perpendicular both to ξ and the axis of rotation $\mathrm{curl}(\xi)$. This is theoretically pleasing in three dimensions, though generalisation is unclear. We hope further understanding to emerge in the future, though not the subject of this report. Unlike CO, SGA is repelled from unstable fixed points even for large λ, provided the sign is chosen according to the following criterion:

$$\mathrm{sign}(\lambda) = \mathrm{sign}\langle \xi, \nabla \mathcal{H} \rangle \cdot \mathrm{sign}\langle A^\intercal \xi, \nabla \mathcal{H} \rangle \,.$$

If this criterion is met, [Bal, Prop. 6] proves that SGA is attracted to SFP while repelled from unstable ones. This result is independent of $|\lambda|$, which is set to 1 in pratice. This provides a flexibility on parameters which CO lacks, as shown in Example 2.20. On the other hand, players are not guaranteed to be selfish, and may act against their immediate loss as in CO.

We prove local convergence of SGA to stable fixed points in section 3.3, strengthening [Bal, Th. 5] from 'attraction' to strict convergence. The main downside of SGA is its formulation as an adjustment for all players simultaneously, rather than through each agent's standpoint. We have not found a way to write SGA as each player optimising some modified loss, making it less intuitive and realistic for applications to RL.

2.3.3 Learning with Opponent-Learning Awareness

As introduced in the motivation, failure of NL occurs because each agent treats others as stationary. Instead, LOLA agents predict opponent learning and optimise the modified loss instead. This perspective is not only more individual and applicable to RL, but allows for opponent shaping (displayed below). For simplicity we derive LOLA for $n = 2$, though the same applies in general. Agent 1 optimises the modified loss

$$L^1(\theta^1, \theta^2 + \Delta\theta^2)$$

with respect to θ^1, where $\Delta\theta^2$ is the predicted learning step. The assumption in [Foe] is that opponents are naive, namely learn by naive gradient descent

$$\theta^2 \leftarrow \theta^2 - \alpha_2 \nabla_2 L^2 \,,$$

so that

$$\Delta\theta^2 = -\alpha_2 \nabla_2 L^2 \,.$$

This is an accurate prediction if the opponent is naive, but can lead to poor behaviour in self-play (see Section 2.4). After first-order Taylor expansion, the loss is approximately given by

$$L^1 + \nabla_2 L^1 \cdot \Delta\theta^2$$

where we omit the arguments (θ^1, θ^2) for convenience. The first term is our usual loss, while the second represents the alignment between the opponent's learning direction and its impact on our loss. The agent optimises this quantity with respect to θ^1 by gradient descent. Differentiating with respect to θ^1, the adjustment for agent 1 is thus given by

$$\nabla_1 L^1 + \left(\nabla_{21} L^1\right)^\intercal \Delta\theta^2 + \left(\nabla_1 \Delta\theta^2\right)^\intercal \nabla_2 L^1 \,.$$

By explicitly differentiating through $\Delta\theta^2$, LOLA actively shapes future opponent learning by choosing parameters that align their next learning step with our goals. As such, this term helps to exploit opponent dynamics and encourage cooperation. We use the word 'exploitation' to mean exploitation of opponent dynamics, not opponents themselves. LOLA differs from policy prediction as in [Zha], which optimises

$$L^1(\theta^1, \hat{\theta}^2 + \Delta\theta^2(\hat{\theta}^1, \hat{\theta}^2))$$

where $\hat{\theta}^1, \hat{\theta}^2$ are the current parameters. After Taylor expansion, the gradient wrt θ^1 is given by

$$\nabla_1 L^1 + \left(\nabla_{21} L^1\right)^\intercal \Delta\theta^2$$

since $\Delta\theta^2(\hat{\theta}^1, \hat{\theta}^2)$ does not depend on θ^1. LOLA contains one further term, by assuming that our opponent's learning step $\Delta\theta^2$ depends on our current optimisation with respect to θ^1. This is inaccurate since the opponent cannot see our updated parameters until the next step, but allows for shaping (influence) in future steps. If α_2 is small enough, the opponent's parameters will not change drastically and our optimisation will be a good approximation to shaping at the *next* step. This is what mostly happens in practice.

In [Foe], the middle term (common with plain lookahead) is dropped because "LOLA focuses on this shaping of the learning direction of the opponent". We do not find it necessary to eliminate this term, and preserving both will in fact be key to finding a stable *and* exploitative algorithm. Experimentally, LOLA displays very encouraging results, both against naive and LOLA agents. A number of examples including Iterated Prisoner's Dilemma (IPD) are given in [Foe], also discussed and reproduced in Chapter 5. In the IPD, self-play LOLA most often converges to tit-for-tat policy where the losses are -1 for both. Instead, NL/SGA/CO almost always converge to defect-defect, where the losses are -2. Both policies are Nash equilibria, but tit-for-tat is strictly superior. We illustrate LOLA's capacity to reach better equilibria through opponent shaping in the following novel example.

Example 2.21 (Logistic game)**.** Define the *logistic* function $\sigma : \mathbb{R} \to (0, 1)$ by

$$\sigma(x) = \frac{1}{1 + e^{-x}} ,$$

often used as an node activation function in neural networks. Consider the game given by

$$l^1(x, y) = 4\sigma(x)(1 - 2\sigma(y)) \qquad \text{and} \qquad l^2(x, y) = 4\sigma(y)(1 - 2\sigma(x)) ,$$

with each loss plotted in Figure 2.3. Note that $\sigma'(x) = \sigma(x)\sigma(-x)$, so

$$\xi = 4 \begin{pmatrix} \sigma(x)\sigma(-x)(1 - 2\sigma(y)) \\ \sigma(y)\sigma(-y)(1 - 2\sigma(x)) \end{pmatrix}$$

which only vanishes at $\sigma(x) = \sigma(y) = 1/2$, namely $x = y = 0$. Hence the only fixed point is $(0, 0)$, but is not a Nash equilibrium since

$$l^1(\epsilon, \epsilon) = l^2(\epsilon, \epsilon) = 4\sigma(\epsilon)(1 - 2\sigma(\epsilon)) < 0 = L^1(0, 0)$$

for any $\epsilon > 0$. There are no Nash equilibria in this game, though it is clear that the desirable behaviour is for agents to move in the direction (x, x) where losses converge to $(-4, -4)$, rather than $(-x, -x)$ where losses converge to $(0, 0)$. To build explicit equilibria, we can add the function

$$B(x, y) = x^2y^2/1000 + (x - y)^2(x + y)^2/1000$$

to both losses, which increases in all directions while being small in a decent neighbourhood of the origin. Note that $f(x, y) = x^2y^2$ increases in all directions except along the lines $x = 0$ and $y = 0$. We can rotate this surface by $\pi/4$ to obtain an increase along these missing lines, through a rescaled rotation matrix:

$$f\left(\sqrt{2}\begin{pmatrix} \cos(\pi/4) & -\sin(\pi/4) \\ \sin(\pi/4) & \cos(\pi/4) \end{pmatrix}\begin{pmatrix} x \\ y \end{pmatrix}\right) = f(x - y, x + y) = (x - y)^2(x + y)^2 .$$

This will increase in all directions except along the lines $x = y$ and $x = -y$, hence adding both terms in B successfully explodes everywhere away from the origin. This bends the surface upwards, giving birth to two Nash equilibria as visualised in Figure 2.3. The 'logistic' game is thus given by

$$L^1 = l^1(x, y) + B(x, y) \qquad \text{and} \qquad L^2(x, y) = l^1(x, y) + B(x, y)$$

and we obtain two solutions to $\xi = 0$, both SFP. Computed numerically, they are given by $\bar{\theta}_{\pm} \approx \pm(5.03, 5.03)$ with corresponding losses

$$L^1(\bar{\theta}_+) = L^2(\bar{\theta}_+) \approx -3.86 \qquad \text{and} \qquad L^1(\bar{\theta}_-) = L^2(\bar{\theta}_-) \approx 0.09\,.$$

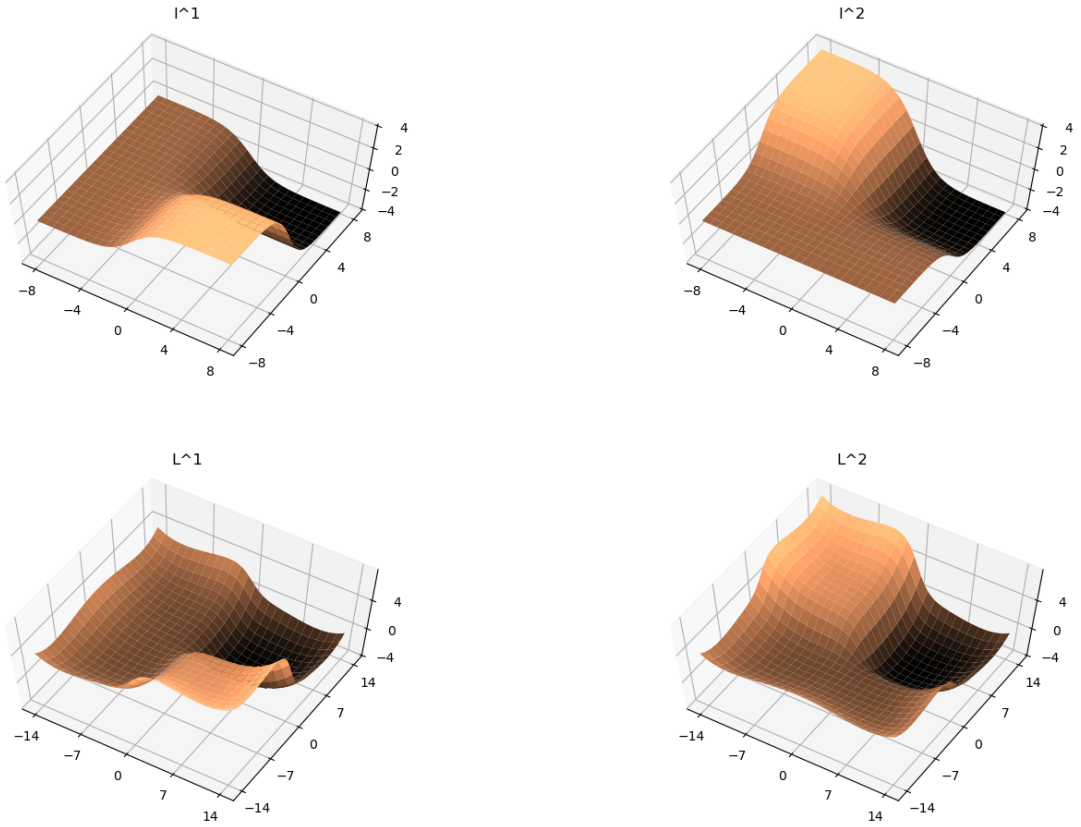

Figure 2.3: Plots of l^1, l^2 (top) and L^1, L^2 (bottom).

It is desirable for an algorithm to reach the better equilibrium $\bar{\theta}_+$, but all previous algorithms are less successful than LOLA in this respect. Note that *local* convergence to both SFP is guaranteed for CO and SGA, but depends on the parameter initialisation. Intuitively, if x_0, y_0 are both positive then $\bar{\theta}_+$ will be reached, while if both are negative then $\bar{\theta}_-$ will be reached. If they have opposite signs then this depends on their size.

On the contrary, LOLA successfully reaches $\bar{\theta}_+$ even from poor initialisations with both x_0, y_0 negative. LOLA will still reach $\bar{\theta}_-$ if the initial parameters are both strongly negative, say $x_0 = y_0 = -1$, but on random initialisation LOLA succeeds more often than CO or SGA. To display this experimentally, we run 300 episodes of training runs with $\alpha = \gamma = 1$. Recall that λ is chosen according to the criterion specified in the previous section, with modulus 1. Each run consists of 100 learning steps, while we initialise $-0.5 < x_0, y_0 < 0.5$ uniformly at random. In Table 2.2 we show that LOLA succeeds in reaching $\bar{\theta}_+$ almost 100% of the time, hence reaching lower losses than SGA/CO/NL on average. This is displayed explicitly in Figure 2.4, where losses are averaged across all episodes with shaded standard deviations.

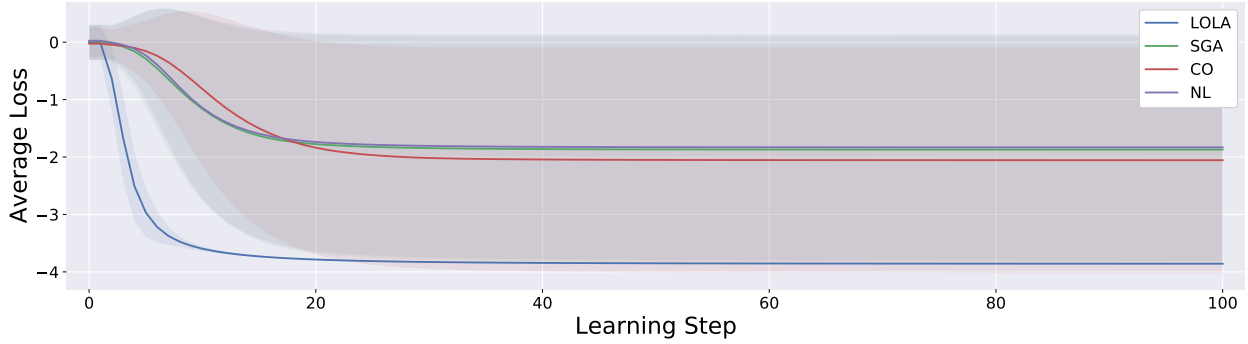

Figure 2.4: Average losses in the logistic game at each learning step, across 300 episodes, with shaded standard deviations. LOLA reaches lower losses than SGA/CO/NL through opponent shaping.

	LOLA	SGA	CO	NL
Mean(std)	$\mathbf{-3.86(10^{-7})}$	$-1.87(1.97)$	$-2.06(1.97)$	$-1.83(1.97)$
% $\bar{\theta}_+$	**100.0**	49.7	54.3	48.7

Table 2.2: Results at the end of 300 training runs in the logistic game. Top column: mean and standard deviations of agent losses. Bottom column: percentage of convergence to $\bar{\theta}_+$. Best result in bold.

On the downside, there is no guarantee that LOLA converges to SFP, unlike SGA and CO. In fact, we will see in section 2.4 that LOLA fails to preserve fixed points of the game. The aim of this project is to find a middle ground between the exploitative capabilities of LOLA, and the convergence guarantees of SGA.

We begin by formulating LOLA in vectorial form, enabling a higher-level comparison with CO and SGA. For simplicity we derive this for two-player games and write the general expression without proof. First write H_d and H_o for the diagonal and off-diagonal of H respectively, and $L = (L^1, \dots, L^n)$ the vector field of loss functions. Finally define the operator $\mathrm{diag} : \mathbb{R}^{d\times d} \to \mathbb{R}^d$ constructing a vector from the matrix diagonal, namely $\mathrm{diag}(M)_i = M_{ii}$ for each entry i.

Proposition 2.22. Let $\chi = \mathrm{diag}(H_o^\intercal \nabla L)$. The LOLA gradient adjustment is given by

$$\mathrm{LOLA} = (I - \alpha H_o)\xi - \alpha\chi\,.$$

Remark 2.23. Just as learning rates are hyperparameters for each player, each agent can treat opponent learning rates as hyperparameters instead of using the true rate. For instance, agent 1 can optimise

$$L^1(\theta^1, \theta^2 - \alpha_2^1 \nabla_2 L^2, \dots, \theta^n - \alpha_n^1 \nabla_n L^n)$$

where α_i^1 is *chosen* by agent 1. This may seem a little unnatural, but in practice can be useful if true learning rates are very small. Indeed, the iterative procedure for LOLA is given by

$$\theta \leftarrow \theta - \alpha\mathrm{LOLA} = \theta - \alpha\xi - \alpha^2(H_o\xi + H_o^\intercal\chi)$$

at each step. If α is very small then α^2 is tiny, rendering LOLA almost identical to naive learning

$$\theta \leftarrow \theta - \alpha\xi\,.$$

This may be undesirable in practical implementations, so opponent rates can be made larger to prevent α^2 from virtually vanishing. This is an issue to keep in mind for practitioners, who may wish to set opponent rates to (say) 1 as in [Foe]. This yields adjustment terms of order α instead of α^2.

Proof. Recall the modified objective

$$L^1(\theta^1, \theta^2 - \alpha\nabla_2 L^2, \ldots, \theta^n - \alpha\nabla_n L^n)$$

for agent 1, and so on for each agent. First-order Taylor expansion yields

$$L^1 - \alpha \sum_{j\neq 1} (\nabla_j L^1)^\intercal \nabla_j L^j$$

and similarly for each agent. Differentiating with respect to θ^i, the gradient adjustment for player i is

$$\begin{aligned}
\text{LOLA}_i &= \nabla_i \left[L^i - \alpha \sum_{j\neq i} (\nabla_j L^i)^\intercal \nabla_j L^j \right] \\
&= \nabla_i L^i - \alpha \sum_{j\neq i} (\nabla_{ji} L^i)^\intercal \nabla_j L^j + (\nabla_{ji} L^j)^\intercal \nabla_j L^i \\
&= \nabla_i L^i - \alpha \sum_{j\neq i} \nabla_{ij} L^i \nabla_j L^j - \alpha \sum_{j\neq i} (\nabla_{ji} L^j)^\intercal \nabla_j L^i \\
&= \xi_i - \alpha \sum_j (H_o)_{ij} \xi_j - \alpha \sum_j (H_o^\intercal)_{ij} (\nabla L)_{ji} \\
&= \xi_i - \alpha (H_o \xi)_i - \alpha (H_o^\intercal \nabla L)_{ii} \\
&= \left[\xi - \alpha H_o \xi - \alpha \operatorname{diag}(H_o^\intercal \nabla L) \right]_i
\end{aligned}$$

and thus

$$\text{LOLA} = (I - \alpha H_o)\xi - \alpha\chi\,.$$ □

For easier comparison with LOLA, note that

$$A = \frac{1}{2}\left(H - H^\intercal\right) = \frac{1}{2}\left(H_o - H_o^\intercal\right)$$

since the diagonal elements of H are cancelled out. Adjusting λ by a factor of 2 in SGA, we obtain

$$\text{SGA} = \xi + \lambda\left(H_o^\intercal - H_o\right)\xi$$

$$\text{CO} = \xi + \gamma\left(H_o^\intercal + H_d^\intercal\right)\xi\,.$$

Note that LOLA and SGA have two common terms ξ and $H_o^\intercal\xi$, while differing in one. We prove the novel result that in two-player zero-sum games, LOLA and SGA become identical. It is enough to assume constant-sum, though these concepts are virtually identical.

Proposition 2.24. In a two-player constant-sum game, LOLA is identical to SGA with $\alpha = \lambda$.

Proof. We have $L^1 = -L^2 + c$ for some constant c, so $\nabla_1 L^2 = -\nabla_1 L^1$ and $\nabla_2 L^1 = -\nabla_2 L^2$. This implies

$$\chi = \operatorname{diag}\begin{pmatrix} 0 & \nabla_{12}L^2 \\ \nabla_{21}L^1 & 0 \end{pmatrix}\begin{pmatrix} \nabla_1 L^1 & \nabla_1 L^2 \\ \nabla_2 L^1 & \nabla_2 L^2 \end{pmatrix} = \operatorname{diag}\begin{pmatrix} \nabla_{12}L^2\nabla_2 L^1 & \cdot \\ \cdot & \nabla_{21}L^1\nabla_1 L^2 \end{pmatrix}$$
$$= \begin{pmatrix} \nabla_{12}L^2\nabla_2 L^1 \\ \nabla_{21}L^1\nabla_1 L^2 \end{pmatrix} = -\begin{pmatrix} \nabla_{12}L^2\nabla_2 L^2 \\ \nabla_{21}L^1\nabla_1 L^1 \end{pmatrix} = -\begin{pmatrix} 0 & \nabla_{12}L^2 \\ \nabla_{21}L^1 & 0 \end{pmatrix}\begin{pmatrix} \nabla_1 L^1 \\ \nabla_2 L^2 \end{pmatrix} = -H_o^\intercal \xi\,.$$

It follows immediately that

$$\text{LOLA} = (I - \alpha H_o)\xi + \alpha H_o^\intercal \xi = \text{SGA}\,. \qquad \square$$

We prove a second novel simplification for LOLA in the class of n-player fully cooperative games.

Proposition 2.25. In a fully cooperative game,

$$\text{LOLA} = (I - 2\alpha H_o)\xi\,.$$

Proof. Since $L^i = L^1$ for all i, we have

$$H_o = \begin{pmatrix} 0 & \cdots & \nabla_{1n}L^1 \\ \vdots & \cdots & \vdots \\ \nabla_{n1}L^1 & \cdots & 0 \end{pmatrix} = H_o^\intercal$$

since $\nabla_{ij}L^1 = (\nabla_{ji}L^1)^\intercal$ for all i, j. Now

$$\xi = (\nabla_1 L^1, \cdots, \nabla_n L^1)^\intercal$$

and thus

$$H_o \nabla L = H_o \begin{pmatrix} \nabla_1 L^1 & \cdots & \nabla_1 L^1 \\ \vdots & \cdots & \vdots \\ \nabla_n L^1 & \cdots & \nabla_n L^1 \end{pmatrix} = H_o \begin{pmatrix} \xi & \cdots & \xi \end{pmatrix} = \begin{pmatrix} H_o\xi & \cdots & H_o\xi \end{pmatrix},$$

from which we obtain

$$\chi = \operatorname{diag}(H_o^\intercal \nabla L) = H_o \xi\,.$$

Finally we conclude

$$\text{LOLA} = (I - \alpha H_o)\xi - \alpha\chi = (I - 2\alpha H_o)\xi\,. \qquad \square$$

Although LOLA is conceptually attractive, applicable to RL and capable of opponent shaping, there are no theoretical guarantees whatsoever. We will prove in Chapter 4 that LOLA converges locally to SFP in two-player zero-sum and n-player fully cooperative games. The next section demonstrates that in general games, LOLA can fail entirely due to false ('arrogant') assumptions about opponents.

2.4 Arrogance and Symmetry

LOLA's core weakness is its failure to preserve fixed points of the game. More precisely, assume $\bar{\theta}$ is a fixed point of the game, so that $\xi(\bar{\theta}) = 0$. Then

$$\text{LOLA} = (I - \alpha H_o)\xi(\bar{\theta}) - \alpha\chi(\bar{\theta}) = -\alpha\chi(\bar{\theta})$$

which may be non-zero. If the agents are at a Nash equilibrium of the original game, one should expect a reasonable algorithm to keep them there, since the optimal policy is the current one. On the contrary, LOLA pushes them away at the next step whenever the term above is non-zero.

Nonetheless, note that Nash equilibria are only optimal for each player if others are stationary. One might hope that moving away from Nash produces better losses for all agents simultaneously; in other words, that the new fixed points are better. This can happen, as in the following example.

Example 2.26. Consider a variant of the cyclic game given by

$$L^1(x, y) = xy - y \qquad \text{and} \qquad L^2(x, y) = -xy + x\,,$$

with

$$\xi = (y, -x)$$

as before. Note that all extra terms are pure opponent so the dynamics of NL, CO and SGA will not be altered. More precisely, CO and SGA are guaranteed to converge to the origin, which is an SFP since

$$H = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} = H_o$$

and thus $S \equiv 0 \succeq 0$. Now

$$\chi = \operatorname{diag}\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} y & -y+1 \\ x-1 & -x \end{pmatrix} = \begin{pmatrix} -x+1 \\ -y+1 \end{pmatrix} \neq 0$$

at $(0, 0)$, so LOLA fails to preserve the fixed point. However, LOLA's fixed point (x', y') is obtained by solving the equation

$$\text{LOLA} = \xi - \alpha H_o\xi - \alpha\chi = 0\,,$$

which gives

$$(x', y') = \frac{\alpha}{1 + 4\alpha^2}(2\alpha - 1, 2)$$

and corresponding losses

$$L^1(x', y') = \frac{-2\alpha(1 + \alpha + 4\alpha^2)}{(1 + 4\alpha^2)^2} \qquad \text{and} \qquad L^2(x', y') = \frac{\alpha(2\alpha - 1)(1 + 4\alpha^2 - 2\alpha)}{(1 + 4\alpha^2)^2}\,.$$

These are both negative for $0 < \alpha < 1/2$, which is better for both agents than the losses $L^1 = L^2 = 0$ at the original fixed point $(0, 0)$.

This example highlights that LOLA, unlike previous algorithms, is capable of dealing with pure opponent terms and adjust parameters accordingly. The dynamics of the game above are *not* identical to the cyclic game for LOLA, since pure opponent terms do not vanish in χ. In particular, $(0,0)$ is not a strong Nash equilibrium and LOLA successfully captures this information. This points to further work regarding higher-order algorithms capable of converging/detecting strong Nash equilibria.

Unfortunately, moving away from Nash in this way does not always produce better losses, as shown in the following example. The problem emerges from the false assumption that an agent can influence opponent learning instantaeously, whereas this actually occurs at the next step. A second cause for failure is the assumption that opponents are naive, when they are actually LOLA in self-play.

Example 2.27 (Humility game)**.** Consider the game given by

$$L^1(x,y) = (x+y)^2/2 - 10x \qquad \text{and} \qquad L^2(x,y) = (x+y)^2/2 - 10y\,.$$

Intuitively, each agent wants to have $x \approx -y$ since $(x+y)^2$ is the leading loss, but also wants to have positive x and y respectively. These are incompatible desires, so the agents must make concessions. The Nash equilibria are given by

$$\xi = \begin{pmatrix} x+y-10 \\ x+y-10 \end{pmatrix} = 0\,,$$

namely any pair $(x, 10-x)$. The corresponding losses are

$$L^1 = 10(5-x) \qquad \text{and} \qquad L^2 = 10(x-5)$$

and sum to 0 for any x. Note that the only Nash equilibrium which is symmetric in both agents is when $L^1 = L^2 = 0$, namely when $x = y = 5$. We have

$$H = S = \begin{pmatrix} 1 & 1 \\ 1 & 1 \end{pmatrix} \succeq 0$$

everywhere. We prove directly that NL converges to Nash from any starting point z_0, while LOLA won't. The eigenvectors of S are

$$u = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \text{and} \qquad v = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

with eigenvalues 2 and 0. Writing $z_k = a_k u + b_k v$, the iteration is given by

$$z_{k+1} = F(z_k) = z_k - \alpha\xi\,.$$

Since F is linear, a Taylor expansion around $(5,5)$ gives

$$\begin{aligned} z_{k+1} &= F(5,5) + \nabla F \cdot (z_k - (5,5)) \\ a_{k+1}u + b_{k+1}v &= (5,5) + (I - \alpha\nabla\xi)^\intercal (z_k - 5u) \\ (a_{k+1}-5)u + b_{k+1}v &= (I - \alpha S)(z_k - 5u) \\ &= (1-2\alpha)(a_k - 5)u + b_k v\,. \end{aligned}$$

By induction it follows that

$$a_k = 5 + (1-2\alpha)^k(a_0 - 5) \qquad \text{and} \qquad b_k = b_0\,,$$

and thus

$$\lim_{k\to\infty} z_k = 5u + b_0 v = \begin{pmatrix} 5 + b_0 \\ 5 - b_0 \end{pmatrix}$$

for $0 < \alpha < 1$. This is a Nash equilibrium, so naive descent converges to Nash for small α. Note that $A = 0$ here, so SGA is identical to NL and we obtain the same guarantee.

However, LOLA converges to fixed points which are not Nash, and are strictly worse for both players. This arises because each LOLA agent overshoots $x + y = 5$, assuming that the other agent will decrease their parameter in response since they are naive. But the other agent is not naive and also overshoots, leading to larger parameters and thus larger losses for both. Each agent becomes 'arrogant' in this way, arising from incorrect assumptions regarding naivety of the opponent *and* immediate response to our update. Formally,

$$\begin{aligned}
\text{LOLA} &= \xi - \alpha H_o \xi - \alpha \chi \\
&= \begin{pmatrix} x+y-10 \\ x+y-10 \end{pmatrix} - \alpha \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x+y-10 \\ x+y-10 \end{pmatrix} - \alpha\,\text{diag} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} x+y-10 & x+y \\ x+y & x+y-10 \end{pmatrix} \\
&= \begin{pmatrix} (x+y)(1-2\alpha) - 10(1-\alpha) \\ (x+y)(1-2\alpha) - 10(1-\alpha) \end{pmatrix}.
\end{aligned}$$

The fixed points are thus pairs (x, y) such that

$$x + y = \frac{10(1-\alpha)}{1-2\alpha} > 10$$

for $\alpha > 0$. For a reasonably small learning rate like $\alpha = 0.05$, the symmetric fixed point occurs at

$$x = y = \frac{10(1-0.05)}{2(1-0.1)} \approx 5.28\,.$$

This leads to losses

$$L^1 = L^2 \approx (10.6)^2/2 - 53 \approx 3$$

which are much larger than 0, the original symmetric fixed point of the game. In general, any fixed point yields

$$L^1 + L^2 \approx 6 \gg 0\,.$$

This decreases to 0 as the learning rate becomes smaller, but is always positive for $\alpha > 0$. Moreover, taking α extremely small is not a viable solution since convergence will be correspondingly slow. We can similarly prove that LOLA converges to these points, which are always worse than Nash. LOLA is thus not a strong algorithm candidate for general games.

We present a few attempts to solve this problem. The principal culprit is LOLA's shaping term, which prevents the algorithm from preserving fixed points. The first subsection investigates the idea of 'killing' this term in a coherent way. This is more detailed than necessary to provide intuition and understanding of the idea behind SOS. In a different direction, the assumption that opponents are naive is also at fault. The second subsection will deal with assuming the opponent is a LOLA agent instead. This is not central to our report and will not be used further on, but an interesting exploration nonetheless.

2.4.1 Killing the Shaping

Consider removing the shaping term entirely, obtaining a gradient adjustment

$$(I - \alpha H_o)\xi\,.$$

This preserves fixed points since $\xi = 0$ implies a zero adjustment. But what does this quantity mean, and is it reasonable to simply kill the shaping term without further justification? Luckily, there is a natural way to view this proposition. Instead of optimising the LOLA objective

$$L^1(\theta^1, \theta^2 - \alpha\nabla_2 L^2(\theta^1, \theta^2))\,,$$

recall from Section 2.3.3 the variant

$$L^1(\theta^1, \theta^2 - \alpha\nabla_2 L^2(\hat{\theta}^1, \hat{\theta}^2))$$

where $\hat{\theta}_1, \hat{\theta}_2$ are the current parameters. In other words, each agent predicts the behaviour of opposite agents after a step of naive learning – but assume that this step will occur independently of our current optimisation. This is the correct assumption, since our parameter update can only influence future steps. We discovered this independently as a cure to the false assumption of dynamic response, but later found that it was originally proposed in [Zha]. We name this method 'lookahead' (LA) for future convenience. Finally we reformulate the objective using stop-gradients, a computational concept introduced below.

Definition 2.28. Let $\perp$ be the *stop-gradient* computational operator, known in PyTorch as `detach` and in Tensorflow as `stop_gradient`. This operator acts on functions, setting their gradient to zero artificially while keeping their value intact. In other words,

$$\perp f(x) = f(x)$$

when evaluated at x, while

$$\nabla(\perp f)(x) = 0$$

for any x. This operator is not well-defined as a mathematical object, since using the equality $\perp f(x) = f(x)$ implies $\perp f = f$ as functions, and thus

$$\nabla(\perp f) = \nabla f \neq 0\,.$$

To avoid confusion we write $\perp f \rightarrowtail f$ and $\nabla(\perp f) \rightarrowtail 0$ as in [Foe2], where $\rightarrowtail$ represents evaluation. In computational terms, 'return $f(x)$ if $\perp f$ is evaluated at x, return 0 if $\nabla(\perp f)$ is evaluated at x'.

In this new language, optimising

$$L^1(\theta^1, \theta^2 - \alpha \nabla_2 L^2(\hat{\theta}^1, \hat{\theta}^2))$$

with respect to θ^1 is equivalent to optimising

$$L^1(\theta^1, \theta^2 - \alpha \perp \nabla_2 L^2)$$

since $L^2(\hat{\theta}^1, \hat{\theta}^2)$ is not a function of θ^1. After first-order Taylor expansion, this is approximately given by

$$L^1 - \alpha \nabla_2 L^1 \cdot \perp \nabla_2 L^2 \,.$$

Differentiating with respect to θ^1, the gradient adjustment is

$$\begin{aligned} &\nabla_1 L^1 - \alpha \left(\nabla_{21} L^1\right)^\intercal \nabla_2 L^2 \\ &= \nabla_1 L^1 - \alpha \nabla_{12} L^1 \nabla_2 L^2 \end{aligned}$$

where the third term present in LOLA disappears thanks to the stop-gradient. The derivation is similar for agent 2, resulting in a gradient adjustment

$$\begin{pmatrix} \nabla_1 L^1 - \alpha \nabla_{12} L^1 \nabla_2 L^2 \\ \nabla_2 L^2 - \alpha \nabla_{21} L^2 \nabla_1 L^1 \end{pmatrix} = \xi - \alpha \begin{pmatrix} 0 & \nabla_{12} L^1 \\ \nabla_{21} L^2 & 0 \end{pmatrix} \begin{pmatrix} \nabla_1 L^1 \\ \nabla_2 L^2 \end{pmatrix} = (I - \alpha H_o)\xi \,.$$

This is precisely LOLA deprived of its shaping term! We can view this as a 'soft' version of LOLA where the shaping term is dropped, while the best-response to naive learning is kept. The main result of [Zha] is that lookahead converges to Nash equilibria in two-player, two-action bimatrix games, through case-by-case analysis of the different game dynamics. In Section 3.5 we prove that LA converges locally to SFP in *any* differentiable game. We also provide a vastly shorter proof of the two-player two-action special case – though applying only locally to SFP, rather than globally to Nash.

Removing the shaping term not only preserves fixed points, but produces strong convergence guarantees. Nonetheless, the whole purpose of LOLA was to incorporate a shaping term giving room for exploitation and cooperation. By dismissing the problematic element, we also discarded LOLA's strengths. Recall from the motivation that one of our aims is to find an algorithm both stable *and* capable of opponent shaping. The initial idea of this project was to combine SGA and LOLA in some way to accomplish this. The difficulty is that SGA and LOLA involve very different terms in general games, so combining them is highly artificial. A better approach is to combine LOLA with lookahead, a natural variant. More precisely, consider optimising

$$L^1(\theta^1, \theta^2 - \alpha \left(p \nabla_2 L^2 + (1-p) \perp \nabla_2 L^2\right))$$

where $p \in [0, 1]$. This corresponds to LA at $p = 0$, and LOLA at $p = 1$. After Taylor expansion, the adjustment is simply given by

$$(I - \alpha H_o)\xi - \alpha p \chi \,.$$

This is a natural interpolation between the two algorithms, where p is a parameter trading between exploitation and stability. Fixing any $p > 0$ will still fail to preserve fixed points, so the question is how to choose p

dynamically. The most naive approach would be to take $p(t)$ as a function of time, with $p(t) \to 0$ as $t \to \infty$. This promises to begin with opponent shaping while converging to SFP eventually. The humility problem is solved since each agent becomes less arrogant over time. But any choice of decreasing function such as $p(t) = 1/t$ is arbitrary, and may either converge too slowly or shape opponents for too little time. Moreover, this prevents p from growing again in the future if necessary, a core component of dynamic exploitation.

Another approach may involve a choice of p based on the accuracy of our assumptions on immediate influence. By measuring the difference between predicted and opponent update at each step, we can adjust p accordingly. In the humility game, agent 1 expects agent 2 to decrease its parameter – but this does not occur. The agent can thus decrease p to mirror the understanding that our opponent will react only in future updates, thus becoming more humble. This still feels like an ad-hoc approach. Instead we develop a more consistent and rational choice criterion for p in Chapter 4, leading to Stable Opponent Shaping (SOS).

2.4.2 Higher-Order LOLA

In the humility game, each agent becomes arrogant because they assume that opponents are naive. Another possibility is to assume that opponents are LOLA agents instead. One might hope to circumvent the arrogance problem in this way. Unfortunately a problem of symmetry arises: the agents are no longer LOLA but a higher-order variant, making their assumption on opponents still incorrect. Moreover, assumption of immediate response to our parameter update still prevents preservation of fixed points.

To see this in detail, we formally define higher-order LOLA agents as follows. Recall that learning with opponent-learning awareness is defined as optimising the modified objective

$$L^1(\theta^1, \theta^2 + \Delta\theta^2)\,,$$

where $\Delta\theta^2$ is the step we expect our opponent to take. In 'first-order' LOLA, we assume the other agent is naive and thus

$$\Delta\theta^2 = -\alpha\nabla_2 L^2(\theta^1, \theta^2)\,.$$

For higher orders, recursively define

$$\Delta\theta_i^2 = -\alpha\nabla_2 L^2(\theta^1 + \Delta\theta_{i-1}^1, \theta^2)$$
$$\Delta\theta_i^1 = -\alpha\nabla_1 L^1(\theta^1, \theta^2 + \Delta\theta_{i-1}^2)$$

for any $i > 0$, and $\Delta\theta_0^1 = \theta_0^2 = 0$. Then a LOLA^i agent is defined to minimise the objective

$$L^1(\theta^1, \theta^2 + \Delta\theta_i^2)\,,$$

i.e. the opponent is assumed to be a LOLA^{i-1} agent. As usual, this is done by gradient descent

$$\theta^1 \leftarrow \theta^1 - \alpha\nabla_1 L^1(\theta^1, \theta^2 + \Delta\theta_i^2)$$

for some learning rate $\alpha > 0$. This is hard or impossible to compute in settings like (model-free) reinforcement learning, where agents can typically only compute the value function and gradients at *current* parameters. Hence we perform a first-order Taylor expansion to obtain

$$\theta^1 \leftarrow \theta^1 - \alpha\nabla_1 L^1 - \alpha\nabla_1(\nabla_2 L^1 \cdot \Delta\theta_i^2)$$

where we omit the arguments (θ^1, θ^2) for convenience. In particular, LOLA^0 is a naive learner (assumes the opponents are stationary), while LOLA^1 is the usual LOLA agent above. For higher orders, $\Delta\theta_i^2$ must also be Taylor-expanded. For instance, for $i = 2$ we have

$$\begin{aligned}\Delta\theta_2^2 &= -\alpha\nabla_2 L^2(\theta^1 - \alpha\nabla_1 L^1, \theta^2) \\ &\approx -\alpha\nabla_2 L^2 + \alpha^2\nabla_2(\nabla_1 L^2 \cdot \nabla_1 L^1)\end{aligned}$$

and obtain the convoluted update

$$\theta^1 \leftarrow \theta^1 - \alpha\nabla_1 L^1 + \alpha^2\nabla_1(\nabla_2 L^1 \cdot \nabla_2 L^2) - \alpha^3\nabla_1(\nabla_2 L^1 \cdot \nabla_2(\nabla_1 L^2 \cdot \nabla_1 L^1))\,.$$

On top of computational tractability, performing these Taylor expansions helps to frame the LOLA^i learning process as comprised of the LOLA^{i-1} component (the first three above), along with an adjustment/response term (the last). Moreover, each higher-order term is smaller than the previous by a factor of α. In particular, LOLA^1 is a simple 'deformation' of naive learning attempting to predict and shape opponent behaviour.

What happens if we play the humility game with LOLA^2 agents? The initial intuition might be that they respond to LOLA^1 overshooting by undershooting, thus converging to the correct Nash equilibrium. Unfortunately this optimism is flawed. Performing similar computations as for LOLA^1, the symmetric fixed point of the new objective is given by

$$x = y \approx 5.23$$

which agents converge to. This produces losses

$$L^1 = L^2 \approx 2.4\,,$$

which is still worse than the Nash equilibrium where both losses are 0. This arises because each agent treats the opponent adjustment term $\Delta\theta_2^2$ as dynamically responsive to our change, whereas it actually occurs a step later. Moreover, the assumption that our opponent is a LOLA^1 agent is still wrong, since they are now LOLA^2. The symmetry in self-play prevents any higher-order agent from being accurate regarding their opponent, since by definition a LOLA^i agent assumes its opponent is LOLA^{i-1}. At any order, the agent believes it is smarter than its competitors, leading to arrogance. Following the trend in the previous subsection, one might wish to consider a soft version of LOLA^2, defined as optimising

$$L^1(\theta^1, \theta^2 + \perp\Delta\theta_2^2)$$

with respect to θ^1. This still fails to preserve fixed points, since the $\Delta\theta_2^2$ term itself involves χ. Nonetheless it turns out that soft LOLA^2 agents converge to parameters with slightly negative losses in the humility game, which is strictly better than the Nash equilibrium. Moreover, we can establish something along these lines as a theoretical guarantee for all games. This is of independent theoretical interest.

Proposition 2.29. If $\nabla_{12}L^1, \nabla_{21}L^2$ are invertible and soft LOLA^2 agents move away from a Nash equilibrium, then both losses decrease for small enough learning rate α.

Proof. Indeed, at Nash we have $\nabla_1 L^1 = \nabla_2 L^2 = 0$ and thus

$$\begin{aligned}\theta^1 &\leftarrow \theta^1 - \alpha\nabla_1 L^1 - \alpha\nabla_1(\nabla_2 L^1 \cdot \perp \Delta\theta_2^2) \\ &= \theta^1 - \alpha(\nabla_{21} L^1)^\intercal(-\alpha\nabla_2 L^2 + \alpha^2\nabla_2(\nabla_1 L^2 \cdot \nabla_1 L^1)) \\ &= \theta^1 - \alpha^3\nabla_{12} L^1(\nabla_{12} L^1)^\intercal \nabla_1 L^2 \,.\end{aligned}$$

A similar derivation holds for the second agent. By Taylor expansion of the loss at the next step,

$$\begin{aligned}&L^1(\theta^1 - \alpha^3\nabla_{12} L^1(\nabla_{12} L^1)^\intercal \nabla_1 L^2, \theta^2 - \alpha^3\nabla_{21} L^2(\nabla_{21} L^1)^\intercal \nabla_2 L^1) \\ &= L^1 - \alpha^3\nabla_1 L^1 \cdot \nabla_{12} L^1(\nabla_{12} L^2)^\intercal \nabla_1 L^2 - \alpha^3\nabla_2 L^1 \cdot \nabla_{21} L^1(\nabla_{21} L^1)^\intercal \nabla_2 L^1 + O(\alpha^4) \\ &= L^1 - \alpha^3(\nabla_2 L^1)^\intercal \nabla_{21} L^1(\nabla_{21} L^1)^\intercal \nabla_2 L^1 + O(\alpha^4) \,.\end{aligned}$$

Note that $\nabla_{21} L^1(\nabla_{21} L^1)^\intercal$ is positive semi-definite and symmetric, so semi-definite by the invertibility assumption. Hence

$$L^1 - \alpha^3(\nabla_2 L^1)^\intercal \nabla_{21} L^1(\nabla_{21} L^1)^\intercal \nabla_2 L^1 + O(\alpha^4) < L^1$$

for small $\alpha > 0$ as required. A similar derivation holds for the second loss, so any small step away from a Nash equilibrium produces lower losses for all. □

This points to soft LOLA^2 as a potentially strong algorithm in discovering better fixed points for all agents, capable of incorporating pure opponent terms while not displaying arrogance in this game. Although soft LOLA^2 may therefore be a strong strategy in self-play, why not use the soft version of LOLA^1, lookahead, in the first place? We will see in Section 3.5 that lookahead not only preserves fixed points of the game, but converges locally to stable fixed points. This is untrue of soft LOLA^2, since preservation of fixed points fails despite the possible advantage that may be. Further inquiry regarding convergence of soft LOLA^2 to *new* fixed points is left for future work.

Chapter 3

Theoretical Results

3.1 Local Convergence

This section introduces Ostrowski's Theorem, a standard result on fixed-point iterations, as a unified framework for proving local convergence of gradient-based methods. This approach is inspired from [Mes]. We recommend going through Appendix A for a quick review of linear algebraic concepts including positive definiteness and stability. Recall only that a matrix M is *positive stable* if all its eigenvalues have positive real part, which holds in particular if M is positive definite. The following is adapted from [Ber, p. 231] and [Ort, 10.1.3]. We provide intuition and a sketch proof in Proposition B.2.

Theorem 3.1 (Ostrowski)**.** Let $F : \Omega \to \mathbb{R}^d$ be continuously differentiable on an open subset $\Omega \subseteq \mathbb{R}^d$, and assume $\bar{x} \in \Omega$ is a fixed point. If all eigenvalues of $\nabla F(\bar{x})$ are strictly in the unit circle of $\mathbb{C}$, then there is an open neighbourhood U of $\bar{x}$ such that for all $x_0 \in U$, the sequence $F^{(k)}(x_0)$ converges to $\bar{x}$. Moreover, the rate of convergence is at least linear in k.

This frames the problem of local convergence as an eigenvalue analysis of ∇F, where F is defined as taking a step in some gradient-based direction. For familiarity we begin with gradient descent on a single loss L. In this case, F is given by

$$F(x) = x - \alpha \nabla L(x)$$

for some learning rate $\alpha > 0$. The gradient is

$$\nabla F = I - \alpha H \,,$$

where $H = \nabla^2 L$ is the Hessian. If H has eigenvalues $\lambda_k = a_k + ib_k$, the eigenvalues of ∇F are

$$1 - \alpha a_k - i\alpha b_k \,.$$

These are in the unit circle if and only if $a_i > 0$ and α is small enough, since the real component is pulled below 1 and the imaginary component is made arbitrarily small. More precisely, we need

$$\begin{aligned} & |1 - \alpha a_k - i\alpha b_k|^2 < 1 \\ \iff \quad & 1 - 2\alpha a_k + \alpha^2 a_k^2 + \alpha^2 b_k^2 < 1 \\ \iff \quad & 0 < \alpha < \frac{2a_k}{a_k^2 + b_k^2} \end{aligned} \tag{$\dagger$}$$

which is possible for any $a_k > 0$. As such, gradient descent is guaranteed to converge locally to a fixed point $\bar{x}$ if $H(\bar{x})$ is positive stable. For a single loss, this is equivalent to being positive definite since H is symmetric, see Proposition A.8. Assuming $H(\bar{x}) \succ 0$, or equivalently $H(\bar{x}) \succeq 0$ and invertible, is thus sufficient to prove local convergence of GD to $\bar{x}$. Note however that Ostrowski does *not* apply if $H(\bar{x}) \not\succ 0$, namely if $\bar{x}$ is not an SFP. In this context, it follows that SFP really is the correct (tractable) subclass of Nash to consider, even for single losses. Further investigation into convergence guarantees to a larger class of Nash, through more involved means than Ostrowski, is left for future work.

For multiple losses, H is not symmetric and NL can fail, as demonstrated in the cyclic game. Formally,

$$F(x) = x - \alpha\xi(x)$$

and $H = \nabla\xi$ is not guaranteed to have positive eigenvalues at stable fixed points, *even if* H is invertible. Indeed, the cyclic game with $L^1 = xy = -L^2$ has an SFP at $(0, 0)$ where

$$H = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

is positive semi-definite and invertible – but antisymmetric and therefore has pure imaginary eigenvalues. Although the conditions in Ostrowski's theorem are sufficient but not necessary, this yields theoretical insight into why simultaneous GD fails for multiple losses. In the next sections, we will investigate second-order algorithms given by

$$F(x) = x - \alpha X\xi(x)$$

for some matrix X. The following proposition will help establish local convergence of these methods.

Proposition 3.2. Assume $\bar{x}$ is a fixed point of a differentiable game such that

$$XH(\bar{x})$$

is positive stable. Then the iterative procedure

$$F(x) = x - \alpha X\xi(x)$$

converges locally to $\bar{x}$ for small enough $\alpha > 0$.

Proof. Since $\bar{x}$ is a fixed point, we have $\xi(\bar{x}) = 0$ and so

$$\nabla[X\xi](\bar{x}) = \nabla X(\bar{x})\xi(\bar{x}) + X(\bar{x})\nabla\xi(\bar{x}) = XH(\bar{x})$$

is positive stable. As in the derivation (†) for single losses, it follows that

$$\nabla F(\bar{x}) = I - \alpha\nabla[X\xi](\bar{x})$$

has eigenvalues in the unit circle for small $\alpha > 0$. Since $\bar{x}$ is also a fixed point of F, we are done by Ostrowkski's Theorem. □

3.2 Consensus Optimization

As introduced in Section 2.3, CO is given by

$$F(\theta) = \theta - \alpha X \xi(\theta)$$

where

$$X = (I + \gamma H^\intercal).$$

Theorem 3.3. Assume H is invertible and positive semi-definite. Then

$$(I + \gamma H^\intercal)H$$

is positive stable for all $\gamma > 0$.

Proof. We have

$$u^\intercal (I + \gamma H^\intercal) H u = u^\intercal H u + \gamma \|Hu\|^2 \geq \gamma \|Hu\|^2 > 0$$

since $Hu \neq 0$ for all non-zero u by invertibility. Hence the matrix is positive definite and in particular, positive stable. □

Corollary 3.4. CO converges locally to stable fixed points $\bar{\theta}$, for any $\gamma > 0$ and sufficiently small $\alpha > 0$.

Proof. Since $\bar{\theta}$ is an SFP, it is a fixed point and $H(\bar{\theta}) \succeq 0$ invertible. We are done by Proposition 3.2 and the Theorem above. □

The main result in [Mes] is that CO converges locally to Nash equilibria in two-player zero-sum games. In this subset of general games, this seems stronger than our result concerned only with SFP. This appearance is misleading. In a two-player zero sum game, $L^1 = -L^2$ and thus

$$H = \begin{pmatrix} \nabla_{11} L^1 & \nabla_{12} L^1 \\ -\nabla_{21} L^1 & \nabla_{22} L^2 \end{pmatrix}.$$

Since $\nabla_{21} L^1 = (\nabla_{12} L^1)^\intercal$, the symmetric part is

$$S = \begin{pmatrix} \nabla_{11} L^1 & 0 \\ 0 & \nabla_{22} L^2 \end{pmatrix}.$$

If $\bar{\theta}$ is a Nash equilibrium then $\xi(\bar{\theta}) = 0$ and $\nabla_{ii} L^i(\bar{\theta}) \succeq 0$ for both players by Proposition B.1. It follows immediately that $S(\bar{\theta}) \succeq 0$ since

$$u^\intercal S u = u_1^\intercal \nabla_{11} L^1 u_1 + u_2^\intercal \nabla_{22} L^2 u_2 \geq 0$$

for any non-zero real vector $u = (u_1, u_2)^\intercal$. It is not necessarily true that $\bar{\theta}$ is an SFP since positive semi-definiteness may not hold in a neighbourhood, see Remark 2.4. Nonetheless, local convergence holds exactly as in the proof above since only $H(\bar{\theta}) \succeq 0$ is assumed for Ostrowski's Theorem to apply. Hence local convergence to all Nash holds in two-player zero-sum games.

Remark 3.5. One might ask why SFP are defined to have $H \succeq 0$ in a neighbourhood, if this assumption is more than necessary for local convergence. This definition was chosen because SFP would not be subsets of Nash equilibria otherwise, as discussed previously and exemplified in Remark 2.5.

We conclude that a minor variation on the theorem above gives local convergence to all Nash equilibria in two-player zero-sum games. As such, this section provides strictly stronger results than [Mes], applying to *all* differentiable games. On the other hand, our extension is a natural generalisation of the arguments presented in [Mes], with little difference in proof strategy. This will not be true of the next sections, where the unified framework of fixed-point iterations will bear genuinely new consequences.

3.3 Symplectic Gradient Adjustment

SGA is given by

$$F(\theta) = \theta - \alpha X \xi(\theta)$$

where

$$X = (I + \lambda A^\intercal)\,.$$

Theorem 3.6. Assume H is invertible and positive semi-definite, with antisymmetric part A. Then there exists $\epsilon > 0$ such that

$$(I + \lambda A^\intercal)H$$

is positive stable for all $0 < \lambda < \epsilon$.

Proof. The proof of this result is inspired from [Bal, Th. 5]. First let S be the symmetric part of H, which has positive real eigenvalues $\sigma_{max} \geq \cdots \geq \sigma_{min}$ by Propositions A.4 and A.7. Define the *additive condition number* of S as $\kappa = \sigma_{max} - \sigma_{min} \geq 0$. Then [Bal, Th. 5] states that

$$\langle \text{SGA}, \nabla\mathcal{H} \rangle > 0$$

for all $0 < \lambda < 4/\kappa$, noting that $\epsilon := 4/\kappa \in (0, \infty]$. Equivalently,

$$\begin{aligned}\langle \text{SGA}, \nabla\mathcal{H} \rangle &= \langle (I + \lambda A^\intercal)\xi, H^\intercal \xi \rangle \\ &= \xi^\intercal H (I + \lambda A^\intercal)\xi > 0\,.\end{aligned}$$

We would like to prove the stronger statement

$$u^\intercal H (I + \lambda A^\intercal) u > 0$$

for all non-zero real vectors u. This would prove positive definiteness of $H(I + \alpha A^\intercal)$ and hence positive stability of $(I + \alpha A^\intercal)H$, see further down. Crucially, the proof of [Bal, Th. 5] does not rely on any properties of the specific vector ξ, and can easily be extended by replacing u everywhere. Details can be found in [Bal, App. B] involving the explicit bound $4/\kappa$, but we provide intuition for why this holds with a short sketch. First decompose

$$u^\intercal H (I + \lambda A^\intercal) u = u^\intercal H u + \lambda u^\intercal H A^\intercal u$$

and notice that $u^\intercal Hu > 0$ implies

$$u^\intercal H(I + \lambda A^\intercal)u = u^\intercal Hu + O(\lambda) > 0$$

for small λ. Otherwise, $u^\intercal Hu = 0 = u^\intercal Su$ and by Cholesky decomposition, there exists a matrix T such that $S = T^\intercal T$. Hence $0 = u^\intercal Su = \|Tu\|^2$, implying $Tu = 0$ and in turn $Su = 0$. Now H is invertible so $Hu = Au + Su = Au \neq 0$, and we obtain

$$u^\intercal H(I + \lambda A^\intercal)u = \lambda u^\intercal HA^\intercal u = \lambda u^\intercal (S + A)A^\intercal u = \|Au\|^2 > 0$$

for all $\lambda > 0$, since $u^\intercal S = u^\intercal S^\intercal = (Su)^\intercal = 0$. In both cases, we obtain

$$u^\intercal H(I + \lambda A^\intercal)u > 0$$

for $\lambda > 0$ small enough. This can be extended uniformly in u to obtain a bound ϵ, as demonstrated in the proof of Theorem 3.12. Alternatively, an explicit bound $\epsilon = 4/\kappa$ is obtained through the argument in [Bal, App. B]. Either way, we conclude that

$$u^\intercal H(I + \lambda A^\intercal)u > 0$$

for all non-zero u and $0 < \lambda < \epsilon$. Hence $H(I + \lambda A^\intercal)$ is positive definite for such λ, and in particular positive stable. Any matrices AB and BA have identical eigenvalues by Proposition A.11, so $(I + \lambda A^\intercal)H$ is also positive stable for $0 < \lambda < \epsilon$. □

Corollary 3.7. SGA converges locally to stable fixed points $\bar{\theta}$, for sufficiently small $\lambda, \alpha > 0$.

Proof. Since $\bar{\theta}$ is an SFP, it is a fixed point and $H(\bar{\theta}) \succeq 0$ invertible. We are done by Proposition 3.2 and the Theorem above. □

Remark 3.8. This corollary, though dependent on an adaptation of [Bal, Th. 5], is a far stronger result. The latter states only that there exists $\epsilon > 0$ such that

$$\langle \text{SGA}, \nabla\mathcal{H} \rangle > 0$$

for all $0 < \lambda < \epsilon$ in neighbourhoods of SFP, which is interpreted as 'SGA points in the same direction as $\nabla\mathcal{H}$'. As mentioned in Section 2.3.1, $\nabla\mathcal{H}$ points in the direction of fixed points since following this gradient minimises $\mathcal{H} = \frac{1}{2}\|\xi\|^2$. In particular, $\mathcal{H}$ points towards SFP in their neighbourhoods, and moreover converges locally to them. The intuition behind [Bal, Th. 5] is that SGA points in the same direction as $\nabla\mathcal{H}$, which is locally convergent. But this is not enough to prove local convergence of SGA in itself! Figure 3.1 clarifies this visually, by noting that $\nabla\mathcal{H}$ may point in *the direction* of SFP while not pointing *at* SFP, which may lead to SGA pointing *away* from SFP.

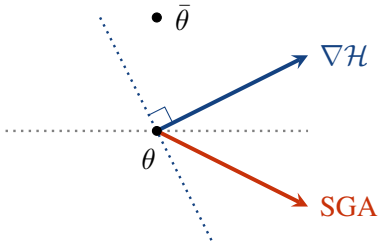

Figure 3.1: Illustration where $\nabla\mathcal{H}$ points towards the SFP $\bar{\theta}$, SGA points in the same direction as $\mathcal{H}$, but SGA does not point towards $\bar{\theta}$.

Nonetheless, it happens that $\nabla\mathcal{H}$ is precisely the direction for which the idea above *does* hold – provided

$$\langle SGA, \nabla\mathcal{H}\rangle = \langle (I + \lambda A^\intercal)\xi, H^\intercal\xi\rangle = \xi^\intercal H(I + \lambda A^\intercal)\xi > 0$$

holds for all non-zero vectors u, not just ξ. Indeed, the fact that $\nabla\mathcal{H} = H^\intercal\xi$ involves the Hessian is crucial and yields positive definiteness of

$$H(I + \lambda A^\intercal)\,,$$

from which we obtain positive stability of $(I + \lambda A^\intercal)H$. Again, pointing in the right direction is not really the fundamental component here, since the inequality must hold for all u. After discussion, the authors of [Bal] were not aware of this and thus 'lucky' in their choice of $\mathcal{H}$ as the quantity to follow. Regardless, our result achieves rigorous local convergence to SFP.

Remark 3.9. What *would* be sufficient for local convergence is truly pointing towards $\bar{\theta}$, namely

$$\langle \text{SGA}, (\bar{\theta} - \theta)\rangle > 0$$

in neighbourhoods of $\bar{\theta}$, a much stronger condition related to local monotonicity. This approach is closer to *variational inequalities*, a technique reducing the problem of finding Nash equilibria to a solution of some functional inequality. This topic is well reviewed in [Scu], and can provide stronger results of global convergence if applicable. Unfortunately we found these methods not to apply in general-sum n-player games, since the assumptions required to obtain monotonicity are far too strong. For instance, [Scu] immediately make the "blanket assumption that the [loss] functions $[L^i]$ are [...], as a function of $[\theta^i]$ alone, convex". This fails to apply in general games, and it seems difficult to obtain monotonicity even with weaker assumptions. Variational inequalities may provide global guarantees in some interesting sub-classes of games, but we chose the local approach of fixed-point iterations for a more successful treatment of the general problem.

3.4 Symmetric Lookahead

Recall that (asymmetric) lookahead was introduced in Section 2.4.1 as best-response to opponent learning given by a gradient direction

$$(I - \alpha H_o)\xi\,.$$

Symmetric lookahead (SLA) is a variant which we discovered, and have not encountered in the literature. The algorithm achieves local convergence, while the proof is suprisingly simple. We derive the gradient

adjustment and results before moving on to lookahead. Instead of best-response to *opponent* learning, each agent responds to *all* agents learning, including oneself. This is a little unintuitive since the agent makes an update based on their own predicted update, but has a natural convergence proof which fails for lookahead. Formally, agent 1 minimises

$$L^1(\theta^1 - \alpha\nabla_1 L^1(\hat{\theta}^1, \hat{\theta}^2), \theta^2 - \alpha\nabla_2 L^2(\hat{\theta}^1, \hat{\theta}^2))$$

with respect to θ^1, where $\hat{\theta}_1, \hat{\theta}_2$ are the current parameters. Using stop-gradients, this is equivalent to minimising

$$L^1(\theta^1 - \alpha\perp\nabla_1 L^1, \theta^2 - \alpha\perp\nabla_2 L^2)\,.$$

After first-order Taylor expansion, this is approximately given by

$$L^1 - \alpha\left[\nabla_1 L^1 \cdot \perp\nabla_1 L^1 + \nabla_2 L^1 \cdot \perp\nabla_2 L^2\right]\,.$$

Differentiating with respect to θ^1, the gradient adjustment is

$$\begin{aligned}&\nabla_1 L^1 - \alpha\left[\left(\nabla_{11} L^1\right)^\intercal \nabla_1 L^1 + \left(\nabla_{21} L^1\right)^\intercal \nabla_2 L^2\right]\\ &= \nabla_1 L^1 - \alpha\left[\nabla_{11} L^1 \nabla_1 L^1 + \nabla_{12} L^1 \nabla_2 L^2\right]\,.\end{aligned}$$

The derivation is similar for agent 2. Putting everything together, we have

$$\begin{aligned}\text{SLA} &= \begin{pmatrix}\nabla_1 L^1 - \alpha\left[\nabla_{11} L^1 \nabla_1 L^1 + \nabla_{12} L^1 \nabla_2 L^2\right]\\ \nabla_2 L^2 - \alpha\left[\nabla_{22} L^2 \nabla_2 L^2 + \nabla_{21} L^2 \nabla_1 L^1\right]\end{pmatrix}\\ &= \xi - \alpha\begin{pmatrix}\nabla_{11} L^1 & \nabla_{12} L^1\\ \nabla_{21} L^2 & \nabla_{22} L^2\end{pmatrix}\begin{pmatrix}\nabla_1 L^1\\ \nabla_2 L^2\end{pmatrix}\\ &= \xi - \alpha H\xi = (I - \alpha H)\xi\,.\end{aligned}$$

The only difference is that (asymmetric) lookahead fails to incorporate diagonal blocks of H, making the proof substantially harder than the symmetric case. The iterative procedure for SLA is given by

$$F(\theta) = \theta - \alpha X\xi(\theta)$$

where

$$X = (I - \alpha H)\,.$$

Theorem 3.10. Assume H is invertible and positive semi-definite. Then there exists $\epsilon > 0$ such that

$$(I - \alpha H)H$$

is positive stable for all $0 < \lambda < \epsilon$.

Proof. Let $\lambda_k = a_k + ib_k$ be the eigenvalues of H. Since $H \succeq 0$, we have $a_k \geq 0$ for all k. It follows that

$$(I - \alpha H)H$$

has eigenvalues

$$(1 - \alpha a_k - i\alpha b_k)(a_k + ib_k)$$

with real part

$$r_k = (1 - \alpha a_k)a_k + \alpha b_k^2 .$$

If $a_k > 0$ then

$$r_k \geq a_k - \alpha a_k^2 > 0$$

for $0 < \alpha < 1/a_k$. If $a_k = 0$ then H invertible implies $b_k \neq 0$, and thus

$$r_k = \alpha b_k^2 > 0$$

for all $\alpha > 0$. We conclude that $(I - \alpha H)H$ is positive stable for all

$$0 < \alpha < \min_j \{1/a_j\}$$

as required. □

Corollary 3.11. Symmetric lookahead converges locally to stable fixed points $\bar{\theta}$, for sufficiently small $\alpha > 0$.

Proof. Since $\bar{\theta}$ is an SFP, it is a fixed point and $H(\bar{\theta}) \succeq 0$ invertible. We are done by Proposition 3.2 and the Theorem above. □

3.5 Lookahead

(Asymmetric) lookahead is given by

$$F(\theta) = \theta - \alpha X \xi(\theta)$$

where

$$X = (I - \alpha H_o) .$$

Theorem 3.12. Let H be invertible and positive semi-definite, and H_o the submatrix of off-diagonal blocks. Then there exists $\epsilon > 0$ such that

$$G = (I - \alpha H_o)H$$

is positive stable for all $0 < \alpha < \epsilon$.

Remark 3.13. Note that G may not be positive definite, although one can show this is true for 2×2 matrices and perhaps also 3×3. This fails one dimension up, exemplified by the matrix

$$H = \begin{pmatrix} 9 & -4 & -3 & -3 \\ -2 & 1 & 2 & 1 \\ -3 & 0 & 1 & 0 \\ -3 & 1 & 2 & 1 \end{pmatrix} .$$

By direct computation (symbolic in α), one can show that $G = (I - \alpha H_o)H$ always has positive eigenvalues for small $\alpha > 0$, whereas its symmetric part always has a negative eigenvalue with magnitude in the order of α. As such, a proof involving lower bounds on matrix-vector products (resembling that for SGA) will fail. This makes the result all the more interesting, but more involved. Central to the proof is a novel *similarity transformation* technique we discovered, and have not found in the literature.

Proof. We cannot study the eigenvalues of G directly, since there is no necessary relationship between eigenpairs of H and H_o. In the aim of using analytical tools, the trick is to find a positive definite matrix which is similar to H, thus sharing the same positive eigenvalues. First define

$$G_1 = (I + \alpha H_d)H \qquad \text{and} \qquad G_2 = -\alpha H^2 \,,$$

where H_d is the sub-matrix of diagonal blocks,and rewrite

$$G = (I - \alpha H_o)H = (I - \alpha(H - H_d))H = (I + \alpha H_d)H - \alpha H^2 = G_1 + G_2 \,.$$

Note that H_d is block diagonal with symmetric blocks $\nabla_{ii} L^i \succeq 0$, so $(I + \alpha H_d)$ is symmetric and positive definite for all $\alpha \geq 0$. In particular its principal square root

$$M = (I + \alpha H_d)^{1/2}$$

is unique and invertible. Now note that

$$M^{-1} G_1 M = M^{-1} M^2 H M = M^\intercal H M \,,$$

which is positive semi-definite since

$$u^\intercal M^\intercal H M u = (Mu)^\intercal H (Mu) \geq 0$$

for all non-zero u. In particular M provides a similarity transformation which eliminates H_d from G_1 while simultaneously delivering positive semi-definiteness. We can now prove that

$$M^{-1} G M = M^{-1} G_1 M + M^{-1} G_2 M$$

is positive definite, establishing positive stability of G by similarity. Let $m = d - 1$ where d is the vector space dimension, namely $H \in \mathbb{R}^{d \times d}$. Take any unit vector $u \in S^m$ and consider

$$u^\intercal M^{-1} G M u \,.$$

First note that a Taylor expansion of M in α yields

$$M = (I + \alpha H_d)^{1/2} = I + O(\alpha)$$

and

$$M^{-1} = (I + \alpha H_d)^{-1/2} = I + O(\alpha) \,.$$

This implies in turn that

$$u^\intercal M^{-1} G M u = u^\intercal G u + O(\alpha) \,.$$

There are two cases to distinguish.

(i) If $u^\intercal Hu > 0$ then

$$\begin{aligned} u^\intercal M^{-1}GMu &= u^\intercal Gu + O(\alpha) \\ &= u^\intercal G_1 u + O(\alpha) \\ &= u^\intercal Hu + O(\alpha) > 0 \end{aligned}$$

for sufficiently small α.

(ii) Otherwise, $u^\intercal Hu = 0 = u^\intercal Su$ and by Cholesky decomposition, there exists a matrix T such that $S = T^\intercal T$. In particular $0 = u^\intercal Su = \|Tu\|^2$ implies $Tu = 0$, and in turn $Su = 0$. Since H is invertible and $u \neq 0$, we have $0 \neq Hu = Au$ and so $\|Au\|^2 > 0$. It follows that

$$-\alpha u^\intercal H^2 u = -\alpha u^\intercal (S^\intercal - A^\intercal)(S + A)u = \alpha u^\intercal A^\intercal Au = \alpha \|Au\|^2 > 0\,.$$

Using positive semi-definiteness of $M^{-1}G_1M$,

$$\begin{aligned} u^\intercal M^{-1}GMu &= u^\intercal M^{-1}G_1Mu + u^\intercal M^{-1}G_2Mu \\ &\geq -\alpha u^\intercal M^{-1}H^2Mu \\ &= -\alpha u^\intercal H^2 u + O(\alpha^2) \\ &= \alpha \|Au\|^2 + O(\alpha^2) > 0 \end{aligned}$$

for $\alpha > 0$ small enough.

We conclude that for any $u \in S^m$ there is $\epsilon_u > 0$ such that

$$u^\intercal M^{-1}GMu > 0$$

for all $0 < \alpha < \epsilon_u$, where $g(\alpha, u) = u^\intercal M^{-1}GMu$ is a function $g : \mathbb{R}^+ \times S^m \to \mathbb{R}$ with S^m compact. By a topological argument in Proposition B.3, this can be extended uniformly with some $\epsilon > 0$ such that

$$u^\intercal M^{-1}GMu > 0$$

for all $u \in S^m$ and $0 < \alpha < \epsilon$. By Proposition A.3, $M^{-1}GM$ is positive definite for all $0 < \alpha < \epsilon$ and thus G is positive stable for α in the same range, by similarity. □

Corollary 3.14. LA converges locally to stable fixed points $\bar{\theta}$, for sufficiently small $\alpha > 0$.

Proof. Since $\bar{\theta}$ is an SFP, it is a fixed point and $H(\bar{\theta}) \succeq 0$ invertible. We are done by Proposition 3.2 and the Theorem above. □

Though local convergence is a strong theoretical result, how small does α really need to be? Unlike the proofs for SGA and SLA, we are not able to produce an explicit bound on ϵ in Theorem 3.12 because the similarity transformation through M fails to yield a tractable expression in α. Nonetheless we can generate random matrices H and find ϵ numerically, by standard bisection methods. This is of interest to practitioners, to produce an estimate on what α to choose and consequently, how fast the algorithm may converge. It is all the more important since the same ϵ will ensure convergence of SOS, the algorithm we propose in Chapter 4.

We generate 10^6 random invertible matrices $H \succeq 0$, with number of agents $2 \leq n \leq 5$ and each dimension $1 \leq d_i \leq 4$ chosen uniformly at random. Each matrix H thus has dimension $2 \leq d \leq 20$ (multinomially distributed). The generation procedure is given in Algorithm 1, and the results are in Table 3.1. We find $0 < \epsilon \leq 10$ such that $(I - \alpha H_o)H$ is postive stable for all $0 < \alpha < \epsilon$ using the bisection method to find roots of the smallest eigenvalue, and more bisection/random sampling to check that no smaller roots exist with high probability. We cap $\epsilon \leq 10$ since some upper bound needs to be chosen, while too large a bound might strongly skew the mean and standard deviation.

Algorithm 1: Random generation of invertible, positive semi-definite matrices.

1 Generate $2 \leq n \leq 5$ and $1 \leq d_i \leq 4$ for each $1 \leq i \leq n$, uniformly at random. Let $d = \sum_i d_i$.
2 Generate a random integer $0 \leq k < d$ and random matrices $M \in \mathbb{R}^{d \times (d-k)}$, $N \in \mathbb{R}^{d \times d}$ with entries between -1 and 1, uniformly at random.
3 Define $S = M^\intercal M$ and $A = N - N^\intercal$, so that $S \succeq 0$ and A antisymmetric.
4 Return $H = S + A$.

We prove that Algorithm 2 produces $H \succeq 0$ invertible with probability one. First note that $H \succeq 0$ since $S \succeq 0$, by Proposition A.4. If H is singular then $Hu = 0$ for some non-zero u, hence by decomposition $Hu = 0$ for some real non-zero u also. Then $0 = u^\intercal Hu = u^\intercal Su = \|Mu\|^2$, implying $Mu = 0$ and thus $Su = 0$. Hence $Hu = Su + Au = Au = 0$, so A is singular. Now A is generated uniformly at random in $[-1, 1]^{d \times d}$, so $\{A \mid \det(A) = 0\}$ is a hyperplane of measure zero and H is invertible with probability 1.

Note that M is not always chosen to be in $\mathbb{R}^{d \times d}$, in order to produce a more realistic sample where S is not always positive definite. Notice also that every symmetric matrix $S \geq 0$ has Cholesky decomposition $S = M^\intercal M$, and every antisymmetric matrix A can be written as $A = N - N^\intercal$, helping to produce a fair sample. We do not however claim that the distribution of H is uniform in the space of invertible, positive semi-definite matrices with entries in some bounded domain.

Min	Max	Mean	Median	Std
0.03	10.00	0.62	0.19	1.64

Table 3.1: Statistics on ϵ across 10^6 random matrix generations.

Overall, we find that α can be chosen reasonably large. More precisely, taking $\alpha \approx 10^{-1}$ is likely to guarantee convergence, since the mean and median are larger and the minimum is $3 \cdot 10^{-2}$.

Remark 3.15. For two-player, two-action, bimatrix games as in [Zha], the theorem above can be proved in a few lines. Two-player two-action means that H is a 2×2 matrix, and we do not make use of the bimatrix assumption. If H has a pure imaginary eigenvalue then it has two, so $Tr(H) = Tr(S) = 0$ and thus S has only zero eigenvalues, hence $S = 0$. Then $G = (I - \alpha H)H$ which is positive stable as easily shown for symmetric lookahead. Otherwise, H has positive eigenvalues and so does a minor perturbation. This proves a subset of [Zha, Th. 1], namely replacing Nash by SFP and global by local convergence. Our result in this

restricted case is therefore weaker, but takes four lines instead of pages. Unfortunately this short argument cannot be generalised to $d \times d$ matrices, since H may have pure imaginary eigenvalues without implying $S = 0$. Our result nonetheless applies to *any* differentiable game, for which nothing is known in the literature to the best of our knowledge.

3.6 Non-Convergence

Though local convergence to SFP is guaranteed, can we say that each algorithm *only* converges to SFP? First notice that each of the four above is given by $X\xi$, where X is a perturbation of the identity by some parameter λ, γ or α:

$$X_{\text{CO}} = I + \gamma H^\intercal \qquad X_{\text{SLA}} = I - \alpha H$$
$$X_{\text{SGA}} = I + \lambda A^\intercal \qquad X_{\text{LA}} = I - \alpha H_o \,.$$

The iterative procedure is given by

$$\theta_{k+1} = \theta_k - \alpha X \xi(\theta_k) \,,$$

where $X\xi$ is a continuous function. If θ_k converges to a point $\bar{\theta}$ then taking limits on both sides implies

$$\bar{\theta} = \bar{\theta} - \alpha X \xi(\bar{\theta}) \,,$$

and so $X\xi(\bar{\theta}) = 0$. If X is assumed invertible at $\bar{\theta}$ then $\xi(\bar{\theta}) = 0$, so $\bar{\theta}$ is a fixed point of the game. The invertibility assumption is a mild one, similar to that of Hessian invertibility at stable fixed points, see Remark 2.12. For instance,

$$X_{\text{LA}} = I - \alpha H_o$$

is invertible iff $1/\alpha$ is not an eigenvalue of H_o, which occurs almost surely if (arbitrarily small) noise is added to α at initialisation. This is equally true of SLA and CO. For SGA this holds even without the presence of noise since the eigenvalues ia of A are pure imaginary. It follows that

$$X_{\text{SGA}} = I + \lambda A^\intercal$$

has non-zero eigenvalues $1 + i\lambda a$. This is summarised below.

Proposition 3.16. Assume CO, SGA, SLA or LA converge to a point $\bar{\theta}$ with X invertible. Then $\bar{\theta}$ is a fixed point of the game, namely $\xi(\bar{\theta}) = 0$.

Though $\bar{\theta}$ is a fixed point, is it necessarily stable? Recall that $\bar{\theta}$ is unstable iff $S \prec 0$. Below we prove that each algorithm cannot converge to an unstable fixed point, a novel contribution for all of them.

In [Bal], the argument that SGA is repelled from unstable FP goes as follows. Recall that gradient descent on $\nabla\mathcal{H} = H^\intercal\xi$ converges to a fixed point of the game, provided H is invertible at that point. This implies that $\nabla\mathcal{H}$ points in the direction of fixed points, though these may be unstable. The requirement is therefore

$$\langle \nabla\mathcal{H}, \text{SGA} \rangle < 0$$

in neighbourhoods where $S \prec 0$, implying that SGA points in the opposite direction and thus *away* from unstable FP. Unfortunately this does not formally prove that convergence to unstable FP is impossible, and is prey to the same fallacy discussed above Figure 3.1. Indeed, $\nabla\mathcal{H}$ may point *towards* the unstable FP $\bar{\theta}$ while not pointing *at* $\bar{\theta}$, making it possible for SGA to point away from $\nabla\mathcal{H}$ while pointing towards $\bar{\theta}$. This is illustrated visually in Figure 3.2 below.

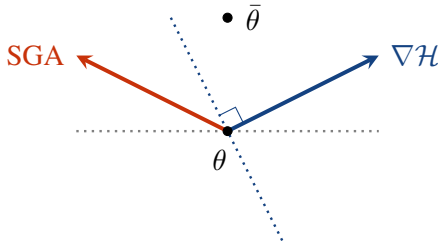

Figure 3.2: Illustration where $\nabla\mathcal{H}$ points towards the unstable FP $\bar{\theta}$, SGA points away from $\nabla\mathcal{H}$, but SGA does not point away from $\bar{\theta}$.

Nonetheless, the condition imposed in [Bal] holds for each algorithm above, provided the parameter (written α without loss of generality) is small enough. Indeed we have $\xi^\intercal H\xi < 0$ at $\bar{\theta}$ and so

$$\langle\nabla\mathcal{H}, X\xi\rangle = \xi^\intercal H\xi + O(\alpha) < 0$$

for sufficiently small α. We provide a novel and rigorous argument for non-convergence to unstable FP, unknown in the literature to the best of our knowledge.

Proposition 3.17. Assume NL, CO, SGA, SLA or LA converges to a point $\bar{\theta}$ with random initialisation. If α is noisy and sufficiently small then $\bar{\theta}$ is almost surely not an unstable fixed point.

Proof. Assume for contradiction that $\bar{\theta}$ is unstable and $\theta_k \to \bar{\theta}$ as $k \to \infty$. We first want to prove that $\theta_k \neq \bar{\theta}$ for all k, almost surely. First note that random initialisation produces $\theta_0 \neq \bar{\theta}$ with probability 1. Now if $\theta_k = \bar{\theta}$ for some smallest $k \geq 1$ then

$$\theta_{k-1} - \alpha\xi_0 = \bar{\theta} \qquad \iff \qquad \theta_{k-1} - \bar{\theta} = \alpha\xi_0$$

where at least some entry i of the LHS vector is non-zero since k is minimal. Hence

$$\alpha = \frac{(\theta_{k-1} - \bar{\theta})_i}{(\xi_0)_i}\,,$$

occurring with probability 0 since α is initialised with noise. $\mathbb{N}$ is a countable set so we obtain

$$\mathbb{P}\left(\bigcup_{k\in\mathbb{N}}\{\theta_k = \bar{\theta}\}\right) = \sum_{k\in\mathbb{N}} P(\theta_k = \bar{\theta}) = \sum_{k\in N} 0 = 0\,.$$

Hence $\theta_k \neq \bar{\theta}$ for all $k \geq 0$ almost surely. With this assumption in hand, let $\delta_k = \|\theta_k - \bar{\theta}\| > 0$ and by Taylor expansion around $\bar{\theta}$,

$$\begin{aligned}\xi_0(\theta_k) &= \xi_0(\bar{\theta}) + \nabla\xi_0(\bar{\theta})(\theta_k - \bar{\theta}) + O(\delta_k^2)\\ &= XH(\theta_k - \bar{\theta}) + O(\delta_k^2)\\ &= H(\theta_k - \bar{\theta}) + O(\alpha) + O(\delta_k^2)\end{aligned}$$

where $H = H(\bar{\theta}) \prec 0$ by definition of instability. For any k we obtain

$$\begin{aligned}
\|\theta_{k+1} - \bar{\theta}\|^2 &= \|\theta_k - \alpha\xi_0(\theta_k) - \bar{\theta}\|^2 \\
&= \left[\theta_k - \bar{\theta} - \alpha\xi_0(\theta_k)\right]^\intercal \left[\theta_k - \bar{\theta} - \alpha\xi_0(\theta_k)\right] \\
&= (\theta_k - \bar{\theta})^\intercal(\theta_k - \bar{\theta}) - 2\alpha(\theta_k - \bar{\theta})^\intercal\xi_0(\theta_k) + O(\alpha^2) \\
&= \|\theta_k - \bar{\theta}\|^2 - 2\alpha(\theta_k - \bar{\theta})^\intercal H(\theta_k - \bar{\theta}) + O(\alpha^2) + O(\delta_k^3) \\
&> \|\theta_k - \bar{\theta}\|^2
\end{aligned}$$

for $\alpha, \delta_k > 0$ sufficiently small. More precisely there exist $\epsilon, \delta > 0$ such that

$$\|\theta_{k+1} - \bar{\theta}\|^2 > \|\theta_k - \bar{\theta}\|^2$$

for all $0 < \alpha < \epsilon$ and $0 < \delta_k < \delta$, rewritten as

$$\delta_{k+1} - \delta_k > 0\,.$$

Now note that $\theta_k \to \bar{\theta}$ implies $\delta_k \to 0$ as $k \to \infty$, so there exists $N \in \mathbb{N}$ such that $\delta_k < \delta$ for all $k \geq N$. In particular this implies

$$\delta_{k'} - \delta_k > 0$$

for all $k' > k \geq N$ and $0 < \alpha < \epsilon$. This is intuitively a contradiction, since the distance between θ_k and $\bar{\theta}$ should *decrease* as k grows, not increase. Formally, $\delta_k > 0$ for all k implies that $\delta_k > \epsilon$ for some $\epsilon > 0$ and $k \geq N$. On the other hand, $\delta_k \to 0$ as $k \to \infty$ implies there exists $M \in \mathbb{N}$ such that

$$\delta_{k'} < \epsilon/2$$

for all $k' \geq M$. Taking $k' > k$ we obtain

$$\delta_{k'} - \delta_k < \epsilon/2 - \epsilon = -\epsilon/2 < 0\,,$$

a contradiction to $\delta_{k'} - \delta_k > 0$. Hence θ_k cannot converge to $\bar{\theta}$. □

Inspired from results in [Lee], we later found that this proposition can be generalised to strict saddles $\bar{\theta}$, namely $H(\bar{\theta})$ has at least one negative eigenvalue. This is true in particular for unstable fixed points. Proving the general result involves a much more sophisticated argument than our proof above, requiring theory from dynamical systems. This goes beyond the purpose of this report, so we state the result without proof. The argument is similar to [Lee, Th. 4.1], though adjustments must be made to account for second-order gradients and our weaker assumption of local (not global) Lipschitz continuity. Our proof will appear in a paper soon to be uploaded on the arXiv.

Proposition 3.18. Assume losses are thrice continuously differentiable and NL, CO, SGA, SLA or LA converges to a point $\bar{\theta}$ with random initialisation. If α is small then $\bar{\theta}$ is almost surely not a strict saddle.

Note that adding noise to α is not necessary, provided loss functions are *thrice* continuously differentiable. This is virtually always satisfied in machine learning.

Remark 3.19. Assuming that $S(\bar{\theta})$ has a negative eigenvalue is not enough to obtain a strict saddle. For instance,

$$H = \begin{pmatrix} 1 & 3 \\ 1 & 1 \end{pmatrix}$$

has only positive eigenvalues $1 \pm \sqrt{3}$, while its symmetric part

$$S = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$$

has a negative eigenvalue -1. In this way, the general result above omits a 'degenerate' type of saddle which fail to produce a negative eigenvalue in H. This is a further difficulty regarding multi-loss optimisation, arising from the lack of Hessian symmetry. We have not found a way to generalise the result above to $\bar{\theta}$ such that $S(\bar{\theta})$ has a negative eigenvalue. This seems impossible using dynamical systems (if not wholly untrue), since $H = \nabla\xi$ is central to the iterative procedure while S is not.

3.7 Caveat

As discussed in Section 2.2, there is no guarantee that any of the algorithms above converge locally to Nash equilibria, and the following example displays a failure case where divergence occurs in the absence of SFP.

Example 3.20. Recall the game given by loss functions

$$L^1 = x^2/2 + 2xy + 2y^2$$
$$L^2 = y^2/2 + 2xy + 2x^2$$

from Example 2.16. The gradient and Hessian are given by

$$\xi = \begin{pmatrix} x + 2y \\ y + 2x \end{pmatrix} \qquad \text{and} \qquad H = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}.$$

The game is potential since we have $A \equiv 0$. The only fixed point is given by $x = y = 0$, which is a (strict) Nash equilibrium since

$$\nabla_{xx} L^1(0,0) = \nabla_{yy} L^2(0,0) = 1 > 0\,.$$

On the other hand, $(0,0)$ is *not* a stable fixed point since $S = H$ has eigenvalues 3 and -1. As such, the game has no SFP and we have no results regarding behaviour of the algorithms above. In fact, we formally prove below that SGA diverges in the direction $(x, -x)$ for any learning rate, any λ and and almost-any initial parameters, where the losses are

$$L^1 = L^2 = x^2/2 > 0\,.$$

Hence the players diverge to infinite losses as they learn, which is the worst possible behaviour.

Proposition. For any learning rate α and almost-any initial parameters $z_0 = (x_0, y_0)$, SGA diverges away from Nash to infinite losses.

Proof. Note that SGA is identical to NL since $A \equiv 0$. Write the eigenvectors of S as

$$u = \begin{pmatrix} 1 \\ 1 \end{pmatrix} \qquad \text{and} \qquad v = \begin{pmatrix} 1 \\ -1 \end{pmatrix}$$

with eigenvalues 3 and -1 respectively. Since they form a basis, we can write

$$z_0 = a_0 u + b_0 v$$

for some $a_0, b_0 \in \mathbb{R}$. The set $\{au \mid a \in \mathbb{R}\}$ is of measure zero in $\mathbb{R}^2$, so we assume that $b_0 \neq 0$. NL follows the negative of ξ, so the iterative procedure is given by

$$z_{k+1} = F(z_k)$$

where

$$F(z) = z - \alpha\xi(z)$$

for some learning rate $\alpha > 0$. Note that F is linear since ξ is linear, so a Taylor expansion around $(0, 0)$ gives

$$F(z) = F(0) + (\nabla F)^\intercal z = 0 + (I - \alpha\nabla\xi)^\intercal z = (I - \alpha H)^\intercal z\,.$$

Writing $z_k = a_k u + b_k v$ and using $H = S$, we obtain

$$z_{k+1} = (I - \alpha S)z_k = a_k(I - \alpha S)u + b_k(I - \alpha S)u = a_k(1 - 3\alpha)u + b_k(1 + \alpha)v\,.$$

By induction it follows that

$$z_k = a_0(1 - 3\alpha)^k u + b_0(1 + \alpha)^k v$$

for all k. For $\alpha < 2/3$, we have $|1 - 3\alpha| < 1$ and thus

$$\lim_{k\to\infty} z_k = \text{sign}(b_0)(\infty, -\infty)\,.$$

For $\alpha \geq 2/3$ the situation is even worse, since both terms blow up but the first oscillates between positive and negative. In both cases the losses become arbitrarily positive. To conclude, SGA diverges towards infinite losses for any learning rate α and almost-any initial parameters. □

The behaviour of SLA and LA is identical to that of SGA in this example, demonstrating a strong failure case for all of them in the absence of SFP. As discussed in Section 2.2.1, the silver lining is that each algorithm fails only because there are terms in each loss function that are purely functions of other players' parameters. They are beyond each player's control entirely, a pathological setting where I can increase my opponent's loss with no possible counter-play. The potential function ϕ disregards these terms, and the Nash equilibrium is a saddle point of ϕ. It is thus 'understandable' for optimisation algorithms to diverge away.

This crucially does not occur for single losses, since the only term beyond one's control is constant, which does not affect local minima. This is one further reason behind the success of gradient descent on a single function. For multiple losses, the only algorithm which takes into account gradients of one's loss with respect to opponent parameters is LOLA, yet even it can be shown to fail in this example.

Chapter 4

Stable Opponent Shaping

CO, SGA and lookahead have strong (non-)convergence guarantees, but fail to shape opponent learning. The opposite holds for LOLA. The aim of this chapter is to solve this catch-22 with Stable Opponent Shaping (SOS). The idea is to interpolate between the two with a parameter p, as first introduced in Section 2.4. The central issue is to find a criterion for choosing p such that both components are preserved.

4.1 Partial LOLA

We begin with p-LOLA, where p stands for partial. This learning scheme is given by

$$p\text{-LOLA} = p \cdot \text{LOLA} + (1-p) \cdot \text{LA}\,,$$

interpolating from LA to LOLA from $p = 0$ to $p = 1$ respectively. Now define

$$\text{LO} = -\alpha\chi$$

as the shaping (but non-preserving) term appearing exclusively in LOLA, and recall that

$$\text{LA} = (I - \alpha H_o)\xi$$

while

$$\text{LOLA} = (I - \alpha H_o)\xi - \alpha\chi = \text{LO} + \text{LA}\,.$$

Then by definition,

$$p\text{-LOLA} = p \cdot (\text{LO} + \text{LA}) + (1-p) \cdot \text{LA} = p \cdot \text{LO} + \text{LA}\,,$$

allowing for an easy mnemonic. We write

$$\xi_p = p\text{-LOLA}$$

for simplicity, where ξ_0 corresponds to lookahead. Another way to formulate p-LOLA is to define the interpolated stop-gradient

$$\perp^p \coloneqq p\perp + (1-p)I$$

where I is the identity operator. Then the first p-LOLA agent optimises the modified objective

$$L^1(\theta^1, \theta^2 - \alpha \perp^{1-p} \nabla_2 L^2)\,,$$

equating the lookahead objective

$$L^1(\theta^1, \theta^2 - \alpha \perp \nabla_2 L^2)$$

at $p = 0$ and the LOLA objective

$$L^1(\theta^1, \theta^2 - \alpha \nabla_2 L^2)$$

at $p = 1$. One can recover the expression for p-LOLA by first-order Taylor expansion

$$L^1 - \alpha(1-p)\nabla_2 L^1 \cdot \perp \nabla_2 L^2 - \alpha p \nabla_2 L^1 \cdot \nabla_2 L^2$$

and similarly for the second. The learning direction is then given by

$$\begin{aligned} &\nabla_1 L^1 - \alpha(1-p)\nabla_{12} L^1 \nabla_2 L^2 - \alpha p \left[\nabla_{12} L^1 \nabla_2 L^2 + \nabla_{12} L^2 \nabla_2 L^1\right] \\ =&\nabla_1 L^1 - \alpha \nabla_{12} L^1 \nabla_2 L^2 - \alpha p \nabla_{12} L^2 \nabla_2 L^1 \end{aligned}$$

or in vector form,

$$(I - \alpha H_o)\xi - p\alpha\chi = p \cdot \text{LO} + \text{LA}$$

as above. We obtain an algorithm trading between stability and exploitation, as a function of p. Since p-LOLA is a perturbation of lookahead, whose gradient is positive stable, the gradient of p-LOLA is also positive stable for p small enough. However this fails to imply that p-LOLA converges locally to SFP, since Ostrowski's Theorem crucially relies on preservation of fixed points. For any fixed point $\bar{\theta}$,

$$p\text{-LOLA}(\bar{\theta}) = -p\alpha\chi(\bar{\theta}) \neq 0$$

for any $p > 0$ in general, so convergence cannot be guaranteed. If p is very small then fixed points will almost be preserved and convergence guaranteed... But for very small p, the algorithm will be virtually identical to LA, losing the whole purpose of interpolation and LOLA's exploitation properties.

Instead, one should find a choice criterion for p such that both shaping and local convergence are preserved. The criterion should allow p to be large whenever possible, while strict enough to provably converge.

4.2 SOS : Rescuing LOLA

The first part of our proposal is to choose p such that p-LOLA points in the same direction as LA, which is known to converge. This will not be enough to guarantee convergence of p-LOLA by itself, but ensures that convergence to new (potentially poor) fixed points cannot occur. In particular this will be shown to prevent failure in the humility game. Criterion 1 is formally given by

$$\langle \xi_p, \xi_0 \rangle \geq 0\,,$$

which is always possible since

$$\langle \xi_p, \xi_0 \rangle = p\langle \text{LO}, \xi_0 \rangle + \|\xi_0\|^2 > 0$$

for $p > 0$ small enough if $\xi_0 \neq 0$, and

$$\langle \xi_p, \xi_0 \rangle = 0$$

otherwise, which virtually never occurs in practice. If $\langle \text{LO}, \xi_0 \rangle \geq 0$ then $\langle \xi_p, \xi_0 \rangle \geq 0$ automatically, so one should choose $p = 1$ to obtain maximal exploitation. Otherwise,

$$p = \min \left\{ 1, \frac{-a\|\xi_0\|^2}{\langle \text{LO}, \xi_0 \rangle} \right\}$$

for some hyperparameter $0 < a < 1$ yields

$$\begin{aligned} \langle \xi_p, \xi_0 \rangle &= p\langle \text{LO}, \xi_0 \rangle + \|\xi_0\|^2 \\ &\geq -a\|\xi_0\|^2 + \|\xi_0\|^2 \\ &= \|\xi_0\|^2(1-a) > 0\,. \end{aligned}$$

This criterion ensures that p-LOLA always points in the same direction as lookahead. The hyperparameter a governs how stable or exploitative we want our agent to be, where a close to 0 yields closest behaviour to lookahead. However, as discussed in Section 3.3, local convergence of an algorithm does *not* follow from pointing in the same direction as some convergent algorithm. A direct proof using Ostrowski's Theorem is also unfeasible since χ does not inherit useful properties from the definition of SFP, while ξ and H do. Nonetheless we were not able to find a non-convergent example despite many adversarial attempts.

We propose a second part to the criterion, thanks to which local convergence holds. The idea is to scale p by a function of $\|\xi\|$ if $\|\xi\|$ is small enough. This will decrease p in neighbourhoods of SFP, pushing p-LOLA arbitrarily close to lookahead and promising convergence. More precisely, take a hyperparameter $0 < b < 1$ and define $p_2 = \|\xi\|^2$ if $\|\xi\| < b$, otherwise $p_2 = 1$. The size of ξ is squared only for purposes of differentiability, as the norm function is not differentiable at the origin. Choosing p_1 and p_2 according to criteria 1 and 2, the *dual* criterion is obtained by taking $p = \min\{p_1, p_2\}$. This is summarised in Algorithm 2.

Algorithm 2: Stable Opponent Shaping

input : Losses $L^1, \ldots, L^n$, learning rate α, hyperparameters $0 < a, b < 1$.
output: Parameters θ, hoping to minimise losses simultaneously.

Initialise θ randomly.
while *not done* **do**
  Compute $\xi_0 = (I - \alpha H_o)\xi(\theta)$ and $\text{LO} = -\alpha\chi(\theta)$.
  if $\langle \text{LO}, \xi_0 \rangle > 0$ **then** $p_1 = 1$ **else** $p_1 = \min\left\{1, \frac{-a\|\xi_0\|^2}{\langle \text{LO}, \xi_0 \rangle}\right\}$
  if $\|\xi\| < b$ **then** $p_2 = \|\xi\|^2$ **else** $p_2 = 1$
  Let $p = \min\{p_1, p_2\}$ and compute $\xi_p = p \cdot \text{LO} + \xi_0$.
  $\theta \leftarrow \theta - \alpha\xi_p$.
end

SOS incorporates two hyperparameters $0 < a, b < 1$, though local convergence is crucially independent from them. Instead, a and b are responsible for more/less exploitation and faster/slower convergence respectively.

SOS preserves fixed points of the game, unlike LOLA. If $\xi(\bar{\theta}) = 0$ then $p = \min\{p_1, 0\} = 0$ and so

$$\xi_p = \xi_0 = (I - \alpha H_o)\xi(\bar{\theta}) = 0$$

as required. This is not only a theoretical guarantee but a practical one, in that ξ small implies $(I - \alpha H_o)\xi$ small and p small, hence ξ_p is also small. This would *not* have been true in the previous criterion, since one can have $p_1 = 1$ even for ξ extremely small, so that

$$\xi_p = \xi_1 = (I - \alpha H_o)\xi - \alpha\chi$$

which may be large if χ is large. Note that only criterion 2 is used in showing preservation of fixed points, though criterion 1 will be equally crucial in obtaining convergence *only* to fixed points.

Remark 4.1. On the practical end, SOS is only slightly slower than LOLA or SGA, the difference being linear in d (the dimension of θ). The same computations of ξ_0 and LO are involved for LOLA, but the criterion for choosing p adds a minor cost of three inner products at each learning step, namely $\langle \text{LO}, \xi_0 \rangle$, $\langle \xi_0, \xi_0 \rangle$ and $\langle \xi, \xi \rangle$. Each inner product has linear running time in d. Numerical results on Algorithm 2's running time will be provided for iterated games in the next chapter, see Table 5.3.

Note also that H_o should not be computed separately from ξ, which would have $O(d^2)$ time complexity. Instead we compute $H_o\xi$ through the following trick. Without loss of generality, the entries of $H_o\xi$ will be given by a sum of terms

$$\nabla_{ij} L^i \nabla_j L^j \,.$$

Instead of computing $\nabla_{ij} L^i$ in squared running time, first compute

$$\nabla_j L^i \cdot \perp \nabla_j L^j$$

for each j, where $\perp$ is the stop-gradient operator. This has $O(d_j)$ complexity for each j and thus $O(d)$ overall. Then compute the required gradient

$$\nabla_i(\nabla_j L^i \cdot \perp \nabla_j L^j) = (\nabla_{ji} L^i)^\intercal \nabla_j L^j = \nabla_{ij} L^i \nabla_j L^j$$

for each i, again totalling $O(d)$ running time. We can perform this to obtain $H_o\xi$ in linear time $O(d)$ accordingly. A similar trick works for χ.

4.3 Theoretical Guarantees

Theorem 4.2. SOS converges locally to SFP for any $a, b \in (0, 1)$ and $\alpha > 0$ sufficiently small.

Proof. Though the criterion is dual, we will only use the second part. More precisely,

$$p = \min\{p_1, p_2\} \leq p_2 = \|\xi\|$$

if $\|\xi\| < b$. The aim is to show that if $\bar{\theta}$ is an SFP then $\nabla\xi_p(\bar{\theta})$ is positive stable for small α, using Ostrowski to conclude as usual. The first problem we face is that $\nabla\xi_p$ does not exist everywhere, since $p(\theta)$ is not a

continuous function. We can nonetheless prove that ξ_p is continuously differentiable in a neighbourhood of $\bar{\theta}$, as follows. First note that $\xi(\bar{\theta}) = 0$ so there is a neighbourhood U of $\bar{\theta}$ such that

$$\|\xi(\theta)\|^2 < b^2$$

for all $\theta \in U$. In particular $p_2(\theta) = \|\xi(\theta)\|^2$ by definition of criterion 2. We want to show that $p(\theta) = p_2(\theta)$ near $\bar{\theta}$, or equivalently $p_1(\theta) \geq p_2(\theta)$. Assume for contradiction that all neighbourhoods have $p_1(\theta) < p_2(\theta)$ for some θ. Then

$$p_1(\theta) = \frac{-a\|\xi_0\|^2}{\langle \mathrm{LO}, \xi_0 \rangle}$$

since otherwise $p_1(\theta) = 1 > b^2 > p_2(\theta)$ according to criterion 1. Now by Theorem 3.12 there exists $\epsilon > 0$ such that

$$(I - \alpha H_o)H(\bar{\theta})$$

is positive stable for all $0 < \alpha < \epsilon$, and by Cauchy-Schwartz we have

$$\frac{-a\|\xi_0\|^2}{\langle \mathrm{LO}, \xi_0 \rangle} \geq \frac{a\|\xi_0\|}{\|\mathrm{LO}\|}\,.$$

By boundedness of U there exists $k > 0$ such that $\|\mathrm{LO}(\theta)\| = \alpha^2\|\chi(\bar{\theta})\| < k$ for all $\theta \in U$ and $\alpha < \epsilon$, hence

$$p_1 = \frac{-a\|\xi_0\|^2}{\langle \mathrm{LO}, \xi_0 \rangle} \geq \frac{a\|\xi\|}{k} > \|\xi\|^2 = p_2$$

for all $\|\xi\| < a/k$. Finally there is a sub-neighbourhood $V \subset U$ such that $\|\xi(\theta)\| < a/k$ for all $\theta \in V$, in which

$$p_1(\theta) = \frac{-a\|\xi_0\|^2}{\langle \mathrm{LO}, \xi_0 \rangle} > p_2(\theta)\,.$$

This is a contradiction, hence $p(\theta) = p_2(\theta) = \|\xi(\theta)\|^2$ for all $\theta \in V$. This is a continuously differentiable function in V, with gradient

$$\nabla p(\bar{\theta}) = 2H^{\mathsf{T}}\xi(\bar{\theta}) = 0$$

at the SFP. Note that $p(\bar{\theta}) = \|\xi(\bar{\theta})\|^2 = 0$, so we obtain

$$\nabla\xi_p(\bar{\theta}) = (I - \alpha H_o)H(\bar{\theta}) - \alpha\nabla p(\bar{\theta})\chi(\bar{\theta}) - \alpha p(\bar{\theta})\nabla\chi(\bar{\theta}) = (I - \alpha H_o)H(\bar{\theta})$$

which is identical to lookahead. This is positive stable for all $0 < \alpha < \epsilon$, and $\bar{\theta}$ is a fixed point of the iteration since

$$\xi_p(\bar{\theta}) = (I - \alpha H_o)\xi(\bar{\theta}) - \alpha p(\bar{\theta})\chi(\bar{\theta}) = 0\,.$$

We conclude by Ostrowski that SOS converges locally to SFP for any $a, b \in (0, 1)$ and α sufficiently small. □

Remark 4.3. Convergence is achieved provided α is as small as required in lookahead, since the same ϵ is used in the proof. The same numerical experiments provided for lookahead therefore hold. Only the radius of convergence will be altered from lookahead to SOS, controlled by b. If b is small then the radius may be smaller, since p is only scaled down when $\|\xi\| < b$.

If the second part of the criterion is discarded in the proof, why not remove it in the first place? The answer is that arrogant behaviour may still be displayed if p-LOLA does not point in the same direction as lookahead, despite preservation of fixed points. More precisely, note that local convergence to SFP does not prevent an algorithm from converging to non-fixed points. Criterion 1 prevents this from happening.

Proposition 4.4. If SOS converges to a point $\bar{\theta}$ with $I - \alpha H_o$ invertible then $\bar{\theta}$ is a fixed point of the game, namely $\xi(\bar{\theta}) = 0$.

The invertibility assumption is identical to that for lookahead, see Proposition 3.16, occurring with probability 1 if (arbitrarily small) noise is added to α. The proof for CO/SGA/SLA/LA was trivial since each of them preserved fixed points in the first place, with the converse being a simple matter of invertibility. For LOLA the proposition above would be untrue, since the shaping term involves χ instead of ξ. The first criterion guarantees this nonetheless, though the proof is more involved.

Proof. The algorithm is an iterative procedure given by

$$\theta_{k+1} = F(\theta_k) = \theta_k - \alpha \xi_p(\theta_k)\,.$$

If $\theta_k \to \bar{\theta}$ as $k \to \infty$ then taking limits on both sides of the iteration yields

$$\bar{\theta} = \bar{\theta} - \alpha \lim_{k\to\infty} \xi_p(\theta_k)$$

and so $\lim_k \xi_p(\theta_k) = 0$, omitting $k \to \infty$ for convenience. It follows by continuity that

$$\xi_0(\bar{\theta}) + \lim_k p(\theta_k)\mathrm{LO}(\bar{\theta}) = 0\,,$$

noting that $p(\theta)$ is not a continuous function. Assume for contradiction that $\xi_0(\bar{\theta}) \neq 0$.

(i) Assume $\langle \mathrm{LO}, \xi_0\rangle(\bar{\theta}) \geq 0$. Note that $\lim_k p(\theta_k) \geq 0$ since $p(\theta) \geq 0$ for all θ, and so

$$\langle \lim_k \xi_p(\theta_k), \xi_0(\bar{\theta})\rangle = \lim_k p(\theta_k)\langle \mathrm{LO}, \xi_0\rangle(\bar{\theta}) + \|\xi_0(\bar{\theta})\|^2 > 0\,.$$

This is a contradiction since $\lim_k \xi_p(\theta_k) = 0$.

(ii) Otherwise, $\langle \mathrm{LO}, \xi_0\rangle(\bar{\theta}) < 0$ and hence $\langle \mathrm{LO}, \xi_0\rangle(\theta) < 0$ in a neighbourhood. In particular there exists $N \in \mathbb{N}$ such that

$$\langle \mathrm{LO}, \xi_0\rangle(\theta_k) < 0$$

for all $k \geq N$. In particular

$$p_1(\theta_k) = \min\left\{1, \frac{-a\|\xi_0(\theta_k)\|^2}{\langle \mathrm{LO}, \xi_0\rangle(T_k)}\right\}$$

for all $k \geq N$. Now notice that

$$\lim_k p(\theta_k) = \lim_k \min\left\{1, \frac{-a\|\xi_0(\theta_k)\|^2}{\langle \mathrm{LO}, \xi_0\rangle(T_k)}, p_2(\theta_k)\right\}\,,$$

which implies

$$\lim_k p(\theta_k) \leq \lim_k \frac{-a\|\xi_0(\theta_k)\|^2}{\langle \mathrm{LO}, \xi_0\rangle(T_k)} = \frac{-a\|\xi_0(\bar{\theta})\|^2}{\langle \mathrm{LO}, \xi_0\rangle(\bar{\theta})}$$

by Proposition B.5 and continuity. Finally we conclude

$$\langle \lim_k \xi_p, \xi_0 \rangle(\theta_k) = \lim_k p(\theta_k)\langle \text{LO}, \xi_0 \rangle(\bar{\theta}) + \|\xi_0(\bar{\theta})\|^2 \geq -a\|\xi_0(\bar{\theta})\|^2 + \|\xi_0(\bar{\theta})\|^2 > 0$$

for any $a \in (0, 1)$, a contradiction.

In both cases a contradiction is obtained, hence $\xi_0(\bar{\theta}) = 0 = (I - \alpha H_o)\xi(\bar{\theta})$. By assumption of invertibility, we obtain $\xi(\bar{\theta}) = 0$ as required. □

Together, criteria 1 and 2 fulfil distinct but complementary requirements of a strong algorithm. The second guarantees preservation of fixed points and local convergence to SFP, while the first ensures its 'converse': SOS can *only* converge to fixed points. Finally p-LOLA is repelled by unstable FP for *any* p, through an argument identical to Proposition 3.17.

Proposition 4.5. Assume p-LOLA converges to a point $\bar{\theta}$ with random initialisation, noisy α and any $p \in [0, 1]$. Then $\bar{\theta}$ is not an unstable fixed point with probability 1, for α sufficiently small.

Proof. Note that

$$\xi_p = \xi - \alpha H_o \xi - p\alpha\chi = \xi + O(\alpha)$$

for any $p \in [0, 1]$. Reproducing the proof for Proposition 3.17, convergence to unstable FP produces a contradiction with probability 1, for α sufficiently small. □

Note that the result above holds for any p, applying in particular to LOLA. This is a novel result. As in Proposition 3.18, this can be generalised to non-convergence to strict saddles $\bar{\theta}$. This is again stated without proof for now. Combining forces into SOS, we obtain the following summary.

Corollary 4.6. For any $a, b \in (0, 1)$, $\alpha > 0$ noisy and sufficiently small, SOS converges locally to SFP. If it converges to a point $\bar{\theta}$ then $\bar{\theta}$ is almost surely a fixed point of the game, and cannot be a strict saddle if losses are thrice continuously differentiable.

Remark 4.7. The reader may be concerned that SOS is not framed from the perspective of each agent, instead incorporating a criterion governed by the global learning direction. This can easily be resolved by the following variant. Each agent applies a local version of the criterion to obtain their component p^i of the *vector* p, so that

$$\xi_p = p \circ \text{LO} + \xi_0$$

where $\circ$ is componentwise multiplication. The criterion is simply an agentwise version of SOS: p_1^i is chosen such that $\langle \xi_p^i, \xi_0^i \rangle > 0$ in the same way with hyperparameter a^i, and $p_2^i = \|\xi^i\|^2$ if $\|\xi^i\| < b^i$ with hyperparameter b^i. Note that $\text{LO}^i, \xi_0^i, \xi^i$ are simply the ith component of the corresponding vectors, while a and b are now hyperparameter *vectors* whose entries must satisfy $0 < a^i, b^i < 1$. Finally we choose $p^i = \min\{p_1^i, p_2^i\}$ for each i. Imposing these local conditions is stronger than SOS, since they imply the global criteria:

$$\langle \xi_p, \xi_0 \rangle = \sum_i \langle \xi_p^i, \xi_0^i \rangle > 0$$

and if $\|\xi\| < b$ then certainly $\|\xi^i\| < b$, so that

$$p_2^i = \|\xi^i\|^2 < b^2$$

is small for each i. The theoretical guarantees for SOS will therefore hold identically for this version, simply by transposing every global argument to the ith component. On the other hand, this variant may be more flexible by keeping some components of p large while others are smaller, helping each player to exploit opponent dynamics *individually*. In particular, the corollary above holds for 'agentwise' or 'local' SOS.

All in all, SOS is theoretically sound. On the practical side, its exploitative capacities are shown to be on par with LOLA in Chapter 5, for suitable choices of a, b. Before moving on, can we say anything about p-LOLA in general? Are there classes of games for which p-LOLA provably converges to SFP for *any* p? This may be of interest to practitioners who disagree with our criterion choice above. Moreover, any such result would equally apply to LOLA since one possible criterion is $p \equiv 1$. No theoretical guarantees are known for LOLA, so any result would be useful and novel.

We are able to establish local convergence of p-LOLA in two-player constant-sum games and n-player fully cooperative games, for any choice of p. In these cases LOLA has a simpler expression, as proven in Propositions 2.24 and 2.25. These are small but interesting subclasses in multi-loss optimisation, for instance applying to the generative adversarial networks in [Goo]. For constant-sum games, we have

$$\text{LOLA} = (I - \alpha H_o + H_o^\intercal)\xi = (I + \alpha A^\intercal)\xi = \text{SGA}$$

by Proposition 2.24 and so

$$p\text{-LOLA} = p \cdot \text{SGA} + (1-p) \cdot \text{LA}\,,$$

an interpolation between two convergent algorithms. It is thus natural to expect local convergence in this case, though the proof is not as immediate as we may hope. Writing

$$K = (I + \alpha A^\intercal)H \qquad \text{and} \qquad G = (I - \alpha H_o)H$$

for the gradients of SGA and LA at a stable fixed point $\bar{\theta}$, recall that G is not necessarily positive definite. Both G and K are positive stable, but we cannot immediately conclude that

$$pK + (1-p)G$$

is positive stable since bounding the eigenvalues of a sum of *non-Hermitian* matrices is an open problem. Nonetheless, we can establish the result by a variant on the proof for lookahead, see Theorem 3.12. Further details can be found there if the slightly more succinct arguments below are insufficient.

Proposition 4.8. p-LOLA is locally convergent to SFP in two-player zero-sum games, for any $p \in [0, 1]$ and α sufficiently small.

Proof. The gradient adjustment is given by

$$\xi_p = (I - \alpha H_o + \alpha p H_o^\intercal)\xi\,,$$

with gradient

$$G = (I - \alpha H_o + \alpha p H_o^\intercal)H$$

at any SFP $\bar{\theta}$ since $\xi(\bar{\theta}) = 0$. The aim is to prove positive stability of G, from which we are done by Proposition 3.2. We use the same similarity transformation trick. First re-write

$$\begin{aligned} G &= (I - \alpha H + \alpha H_d + \alpha p H^\intercal - \alpha p H_d^\intercal) H \\ &= (I + \alpha(1-p) H_d) H - \alpha H^2 + \alpha p H^\intercal H \\ &= G_1 + G_2 \,. \end{aligned}$$

Note that $(I+\alpha(1-p)H_d)$ is symmetric and positive definite for $\alpha \geq 0$ and $p \in [0,1]$, since H_d is symmetric and positive semi-definite. In particular its principal square root

$$M = (I + \alpha(1-p)H_d)^{1/2}$$

is unique and invertible. Now

$$M^{-1} G_1 M = M^{-1} M^2 G_1 M = M^\intercal H M \succeq 0$$

since $H \succeq 0$. We use the same tricks of Taylor expansion and two-case analysis to conclude that $M^{-1}GM \succ 0$, as follows. Take any unit vector u. If $u^\intercal H u > 0$ then

$$u^\intercal M^{-1} G M u = u^\intercal G_1 u + O(\alpha) = u^\intercal H u + O(\alpha) > 0$$

for sufficiently small α and bounded $p \geq 0$. Otherwise $u^\intercal H u = 0 = u^\intercal S u$ implies $Su = 0$ and so $Au \neq 0$ by invertibility of H. Hence

$$\begin{aligned} u^\intercal M^{-1} G M u &= u^\intercal M^{-1} G_1 M u + u^\intercal M^{-1} G_2 M u \\ &\geq \alpha \left[-u^\intercal M^{-1} H^2 M u + p M^{-1} H^\intercal H M u \right] \\ &= \alpha \left[-u^\intercal H^2 u + p H^\intercal H u \right] + O(\alpha^2) \\ &\geq \alpha u^\intercal A^\intercal A u + O(\alpha^2) \\ &= \alpha \|Au\|^2 + O(\alpha^2) > 0 \end{aligned}$$

for α small enough and any $p \geq 0$. It follows that for any $u \in S^m$ there is $\epsilon_u > 0$ such that

$$u^\intercal M^{-1} G M u > 0$$

for all $0 < \alpha < \epsilon_u$ and $p \in [0,1]$, and we conclude by Proposition B.3 that there exists $\epsilon > 0$ such that $M^{-1}GM \succ 0$ for all $0 < \alpha < \epsilon$ and $p \in [0,1]$. It follows that G is positive stable in the same range by similarity, as required. □

For (fully) cooperative games, recall that

$$\text{LOLA} = (I - 2\alpha H_o)\xi$$

by Proposition 2.25, which is identical to lookahead with α replaced by 2α. It follows that

$$p\text{-LOLA} = p \cdot \text{LOLA} + (1-p) \cdot \text{LA} = (I - \alpha(1+p)H_o)\xi \,.$$

Local convergence of p-LOLA hence becomes an easy consequence of Theorem 3.12.

Proposition 4.9. p-LOLA is locally convergent to SFP in n-player cooperative games, for any $p \in [0, 1]$ and α sufficiently small.

Proof. The gradient adjustment is given by

$$\xi_p = (I - \alpha(1 + p)H_o)\xi \,,$$

with gradient

$$(I - \alpha(1 + p)H_o)H \,.$$

By Theorem 3.12, there exists $\epsilon > 0$ such that

$$(I - \alpha H_o)H \succ 0$$

for all $0 < \alpha < \epsilon$. In particular

$$(I - \alpha(1 + p)H_o)H \succ 0$$

for all $0 < \alpha < \epsilon/2$ and $p \in [0, 1]$, since then $0 < \alpha(1 + p) < 2\alpha < \epsilon$. Local convergence follows as usual by Proposition 3.2. □

In particular, p-LOLA is locally convergent with *any* criterion in each class of games, allowing for simultaneously exploitative and stable capacities. The results holds for $p \equiv 1$, providing previously unknown convergence guarantees for LOLA.

Corollary 4.10. LOLA is locally convergent to SFP in two-player zero-sum games and n-player cooperative games, for α sufficiently small.

If the agents know in advance that the game is two-player zero-sum or fully cooperative, $p \equiv 1$ may be the best criterion since convergence is achieved while full exploitation holds. This is virtually never true in RL, where agents only have access to losses at the *current* parameters. Even for a large sample of losses at different points, they cannot determine with high probability that a game is zero-sum, cooperative or any such 'meta' property of the game. It is thus wiser to use SOS by default, if no prior knowledge about the environment is given. This will guarantee local convergence to true fixed points in *any* differentiable game.

Chapter 5

Experimental Results & Discussion

5.1 Experimental Setup

We evaluate the performance of SOS on a number of explicit differentiable games. The 'dual game' is constructed to showcase the advantages of SOS, while the iterated matching pennies and prisoner's dilemma are taken from [Foe] for realistic comparison with LOLA. We add the iterated stag hunt for further diversity. In future work we hope to implement SOS on more involved settings like GANs and deep multi-agent RL (via policy gradient approximation). We have obtained preliminary results on learning Gaussian mixtures using GANs, with state-of-the-art results for SOS. This is not displayed here for lack of time/rigorous comparison, though appearing soon on the arXiv. We hope the examples in this chapter to give a taste for the stable and exploitative capacities of SOS, at least as proof of concept. In each experiment we compare performance with LOLA, LA, SGA and NL for opponent diversity. This also improves on the results in [Foe], showing that LOLA outperforms SGA and LA in the iterated games.

In each game we run 300 training episodes for each algorithm, where a run consists of 500 learning steps. The parameters are initialised following a normal distribution around 0 (corresponding to $1/2$ probability in iterated bimatrix games). The only hyperparameter is the learning rate α, on top of $0 < a, b < 1$ for SOS. Recall that λ in SGA is chosen according to the criterion specified in Section 2.3.2, with modulus 1. In all experiments we fix $\alpha = 1$ as in [Foe], allowing for a large opponent shaping term. Similarly we choose $a = 0.5$ and $b = 0.1$ everywhere; the first is an arbitrary middle ground between 0 and 1, while the latter is intentionally small to ensure that SOS avoids poor SFP. We found the results to be dependent on b being small enough in games like the IPD, while a is a more robust hyperparameter.

5.1.1 Iterated Bimatrix Games

A (two-player) bimatrix game is given by matrices A and B, where players 1 and 2 choose actions i, j and receive payoffs A_{ij}, B_{ij} respectively. For instance, Example 2.7 corresponds to the game of matching pennies where each player can choose Heads or Tails and

$$A = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix} = -B\,.$$

This is often summarised by writing both matrices into a table, as shown in Table 5.1

	Heads	Tails
Heads	$(1,-1)$	$(-1,1)$
Tails	$(-1,1)$	$(1,-1)$

Table 5.1: Payoff matrix for players $(1,2)$ in Matching Pennies.

If players 1 and 2 can choose between n and m possible actions respectively, A and B are matrices of dimension $n \times m$ and the parameters are $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ respectively. The losses are given by

$$L^1 = -x^\intercal A y \qquad \text{and} \qquad L^2 = -x^\intercal B y$$

where minus signs account for losses being opposite to payoffs. The parameters x, y can be interpreted as probabilities of choosing each action, either via sigmoid functions or an appropriate change of coordinates as illustrated in Example 2.7. Iterated bimatrix games consist of an infinitely repeated sequence of such games, where the loss is discounted by a factor $\gamma \in [0,1)$ at each repetition. This is identical to the non-iterated version if the agents cannot condition their actions on previous history, since the loss would be

$$L^1 = \sum_{t \geq 0} -x^\intercal A y \cdot \gamma^t = \frac{-x^\intercal A y}{1-\gamma}\,.$$

This is unrealistic since agents should be able to adapt their actions depending on the past – an inherent part of human 'learning'. As such, agents are usually enriched with a memory of length K. This would lead to a high number of parameters for each agent, growing exponentially in K. [Pre, App. A] prove that the performance of a player with memory 1 is independent from the opponent's memory $K \geq 1$ in iterated games; we consider memory-1 agents only in this report, as in [Foe]. In all examples below, we have $n = m = 2$ so each player can choose between 2 actions at each step. In effect, there is only one parameter given by the probability of choosing action 1, though there is one such parameter for each memory state.

We model iterated bimatrix games (IBG) as Markov Decision Processes (MDP). Each agent's policy at time $t \geq 1$ is determined by the current state s_t, which is given by the actions of each agent at the previous step:

$$s_t = (a^1_{t-1}, a^2_{t-1})$$

for $t > 0$ and $s_0 = \varnothing$ otherwise. For $n = m = 2$ there are thus five states

$$r_0 = \varnothing, \quad r_1 = (1,1), \quad r_2 = (1,2), \quad r_3 = (2,1), \quad r_4 = (2,2)\,.$$

For example, iterated matching pennies has

$$r_0 = \varnothing, \quad r_1 = HH, \quad r_2 = HT, \quad r_3 = TH, \quad r_4 = TT$$

where H, T correspond to playing Heads or Tails. The policy of agent i is thus fully determined by 5 parameters, namely the probability

$$\theta^{i,j} = \pi^i(a_t = 1 \mid s_t = r_j)$$

of choosing action 1 (playing Heads) given the current state r_j. In Proposition B.6 we derive analytical loss functions for these IBG, allowing us to implement SOS without approximation methods.

The first game is iterated matching pennies (IMP), with payoff matrix in Table 5.1. The IMP has a single Nash equilibrium, given by 50/50 strategy of playing heads or tails. Deviating from this stategy allows for exploitation by the opponent: if agent 1 plays Heads (or Tails) with more than 50% probability, agent 2 can play Tails (or Heads) more often to receive higher reward. Since the game is zero-sum, agent 1 correspondingly wins less. The same holds for agent 2, so deviating should be avoided in the long run. The discount factor is chosen to be $\gamma = 0.9$ as in [Foe].

	Cooperate	Defect
Cooperate	$(-1,-1)$	$(-3,0)$
Defect	$(0,-3)$	$(-2,-2)$

	Stag	Hare
Stag	$(2,2)$	$(0,1)$
Hare	$(1,0)$	$(1,1)$

Table 5.2: Payoff matrices for players $(1,2)$ in Prisoner's Dilemma (left) and Stag Hunt (right).

The second game is iterated prisoner's dilemma (IPD), with payoff matrix in Table 5.2. This is a famous example where two criminals are arrested and isolated from each other. Each prisoner can choose to cooperate with her colleague by remaining silent, or defect to betray them. The payoff $-c$ depends on both decisions, corresponding to c years in prison. In the one-shot version, the only Nash equilibrium is for the players to always defect (DD). In the iterated game, DD is also an equilibrium where the reward is -2 per step. There is a better equilibrium named *tit-for-tat* (TFT), where each player begins by cooperating and then mimicks the opponent's previous action. This leads to rewards of -1 for both, since they begin by cooperating and then always cooperate. Neither has an incentive to start defecting since the opponent will defect at the next step otherwise, leading to lower rewards. The discount factor is chosen to be $\gamma = 0.96$ as in [Foe], with higher value allowing for cooperation in the long run.

The final game is iterated Stag Hunt (ISH), with payoff matrix in Table 5.2. Stag Hunt is a game whereby two hunters can individually choose to hunt a stag or a hare. Each is capable of hunting a hare by themselves, but require the other's help to hunt a stag. The pure Nash equilibria are both hunting stag (SS) or both hunting hare (HH) in the one-shot game. In the iterated version, SS and HH are still equilbria with reward 2 and 1 per step respectively, and so is tit-for-tat. This example was not present in [Foe], added to show versatility in opponent shaping for SOS and LOLA. The discount factor is $\gamma = 0.96$ for the same reason as IPD.

5.1.2 Dual Game

The results on iterated games will show that SOS is on par with LOLA with respect to exploitation. On the other hand, recall that SOS was built to overcome LOLA's shortcomings by preserving fixed points and converging locally. In particular, LOLA fails to preserve the SFP $(x, 10-x)$ and converges to worse parameters in the humility game, given by

$$L^1(x,y) = (x+y)^2/2 - 10x \qquad \text{and} \qquad L^2(x,y) = (x+y)^2/2 - 10y\,.$$

Instead, SOS is guaranteed to converge to the correct SFP. Though a strong improvement over LOLA, this is also true of other stable algorithms like CO/SGA/LA. On the other hand, the latter cannot shape opponent learning. This is technically sufficient to conclude that SOS is superior to both classes, but we construct an artificial game to display *simultaneous* superiority. We add the logistic losses from Example 2.21 to the humility game above. Recall that SGA/CO/NL more often fail to reach the better equilibrium in the logistic game, while LOLA succeeds. Explicitly, let f^i and g^i be the losses for logistic and humility games respectively. The 'dual' game is defined as

$$L^i(x, y, z, w) = f^i(x, y) + g^i(z, w)/5$$

where x, z and y, w are the first and second agent's parameters respectively. This is equivalent to playing logistic and humility games simultaneously. The factor of 5 is chosen to make the losses of logistic and humility games closer in size, for simplicity of comparison in the results. It is important to combine these games with distinct parameters, as the humility game will easily dominate the shape of the losses near the origin otherwise. Though an artificial example, this game will point to experimental superiority of SOS in a succinct way, through simultaneous stability and exploitation.

5.1.3 Running Times

Following Remark 4.1, running times are given in Table 5.3 for each algorithm in the iterated games. The running time spans all 300 training runs, with 500 learning steps in each episode. Note the small disparity between SOS and LOLA, with time ratios of $1.13/1.12/1.10$ in the IMP / IPD / ISH.

	SOS	LOLA	LA	SGA	NL
IMP	170.3	150.4	128.6	145.2	101.3
IPD	167.5	149.2	128.3	144.3	101.6
ISH	166.4	150.7	127.6	145.0	102.4

Table 5.3: Running times for each game across all 300 training runs, in seconds.

5.2 Results & Discussion

The results for each game are given in Figures 5.1, 5.2, 5.3 and 5.5. Parameters at the end of each training run are displayed in part (A), with agents 1 and 2 corresponding to x and y axes respectively. Only 50 runs are shown for parameter visibility. Losses at each learning step are displayed in part (B) of each figure, averaged across 300 episodes with shaded standard deviations. To avoid confusion, recall that losses are the opposite of payoffs/rewards in iterated games. Finally losses are displayed only for agent 1 for visibility, though self-play imposes identical results for agent 2 up to random initialisation. Tables 5.4 and 5.5 summarise the results for iterated and dual games respectively. Note also that our results for LOLA are virtually identical to those in [Foe], despite their implementation discarding the 'middle' term $H_o\xi$. Performance is therefore not conditional on the absence of this term – while its presence is central to the idea of interpolation and SOS.

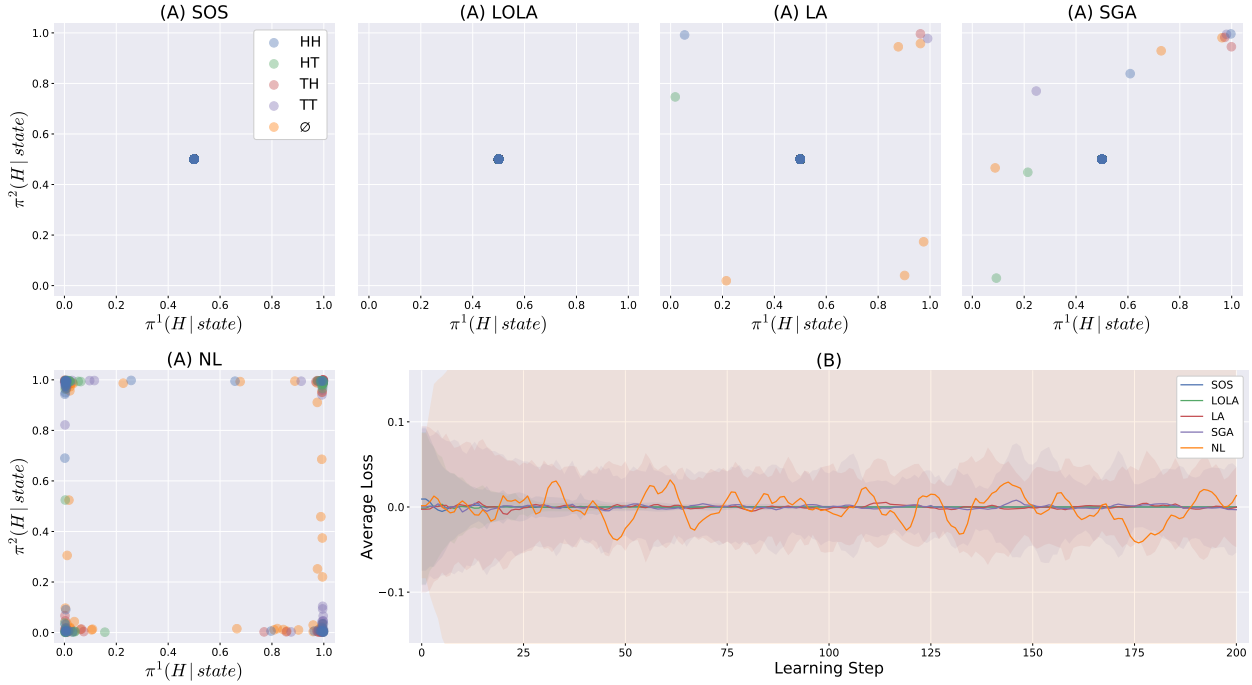

Figure 5.1: Results in the IMP. (A) Probability that SOS, LOLA, LA, SGA, NL agents play Heads, given memory state, at the end of 50 training runs. SOS and LOLA always play Heads with 50%, while NL always (and LA/SGA sometimes) deviate. (B) Average loss at each learning step with shaded standard deviations, across 300 runs. Only 200 steps are shown for visibility.

5.2.1 Iterated Bimatrix Games

The results for IMP are given in Figure 5.1. Recall that parameters in (A) are the end-run probabilities of playing Heads for each memory state, encoded by different colours.

SOS and LOLA succeed in converging to Nash, namely always playing Heads with 50% probability. This is displayed by the accumulation of points in the centre of (A) plots. SGA and LA sometimes deviate from this strategy, but mostly succeed. NL fails to converge entirely, as shown by the absence of points in the center. Failure of convergence to Nash is also shown in part (B), where losses are highly erratic for NL at each step. The standard deviation is extremely large for NL, smaller but still significant for SGA/LA, and virtually zero for SOS/LOLA. This is also displayed numerically in Table 5.4, including the $\%$ of runs converging to Nash.

Finally, we expected SGA/LA to be on par with SOS/LOLA regarding convergence to Nash in the IMP. This is almost true since convergence to Nash occurs in 97.0% and 96.1% of runs, but some runs fail and the standard deviations do not decrease after the 100th learning step. Though SGA/LA have theoretical guarantees, these only apply for α small enough. In this case, $\alpha = 1$ may be too large for convergence to occur in all training runs. This points to another advantage of opponent shaping for SOS/LOLA, namely the capacity to encourage cooperation and hence convergence – despite a relatively large learning rate.

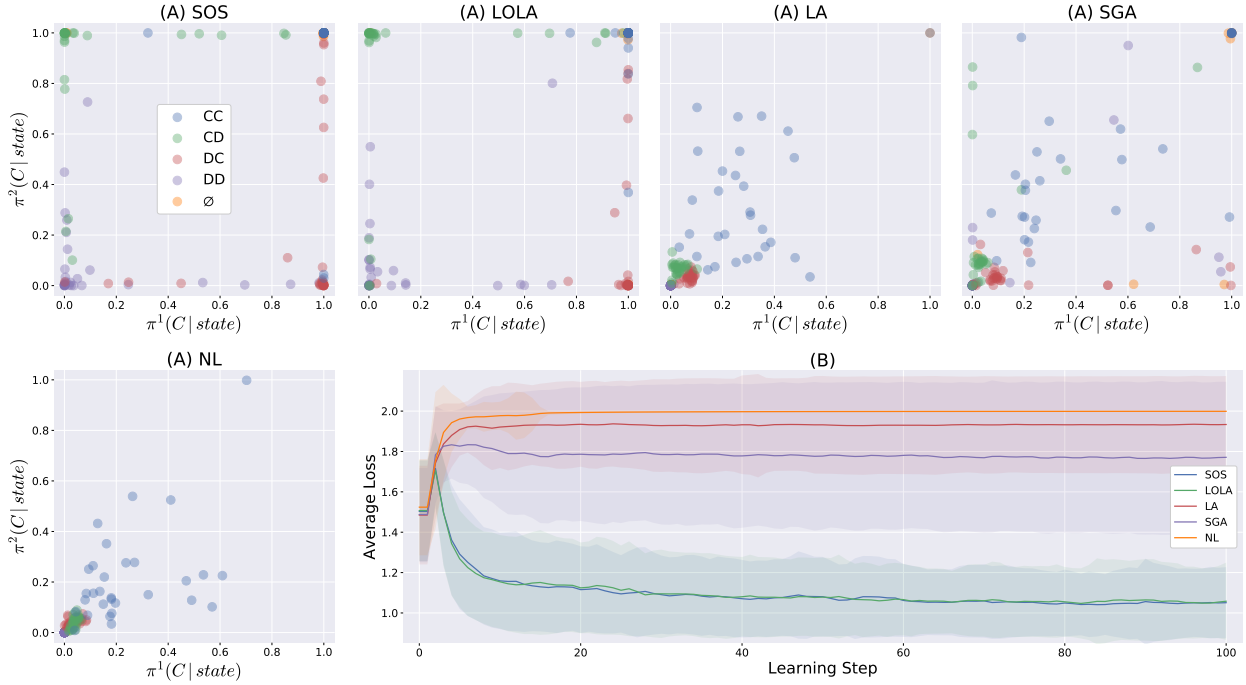

Figure 5.2: Results in the IPD. (A) Probability that SOS, LOLA, LA, SGA, NL agents cooperate, given memory state, at the end of 50 training runs. SOS and LOLA often play tit-for-tat, while SGA/LA/NL mostly defect. (B) Average loss at each learning step with shaded standard deviations, across 300 runs. Only 100 steps are shown for visibility.

The results for IPD are given in Figure 5.2. Recall that parameters in (A) are the end-run probabilities of cooperating for each memory state, encoded by different colours.

SOS and LOLA mostly succeed in playing tit-for-tat, displayed by the accumulation of points in the correct corners of (A) plots. For instance, if agent 1 cooperates then agent 2 responds by cooperating (blue and green points are mostly at the top). Yellow points are mostly hidden behind the blue points at the top right corner, so the agents also cooperate in the first game iteration. Tit-for-tat strategy is further indicated by the losses close to 1 in part (B). On the other hand, most points for LA/SGA/NL are accumulated at the bottom left, namely agents almost always defect. This is also displayed by losses close to 2 in (B), though NL is worse than LA which is worse than SGA.

The percentage of parameters obeying TFT policy is displayed for each algorithm in Table 5.4. This is obtained by counting the number of points in the correct corner, as in [Foe]. Note that 20% TFT policy is automatically achieved by always defecting, since defecting when both agents defected (a fifth of parameters) is also true in TFT policy. SOS and LOLA are able to shape opponent learning to encourage cooperation and thus convergence to TFT with 80% probability, while SGA/LA/NL mostly defect. Finally note that the TFT policy is sufficient but not necessary to obtain losses near 1, since SOS has lower losses than LOLA while having a lower TFT percentage. Nonetheless, TFT is a good indicator of cooperation and opponent shaping.

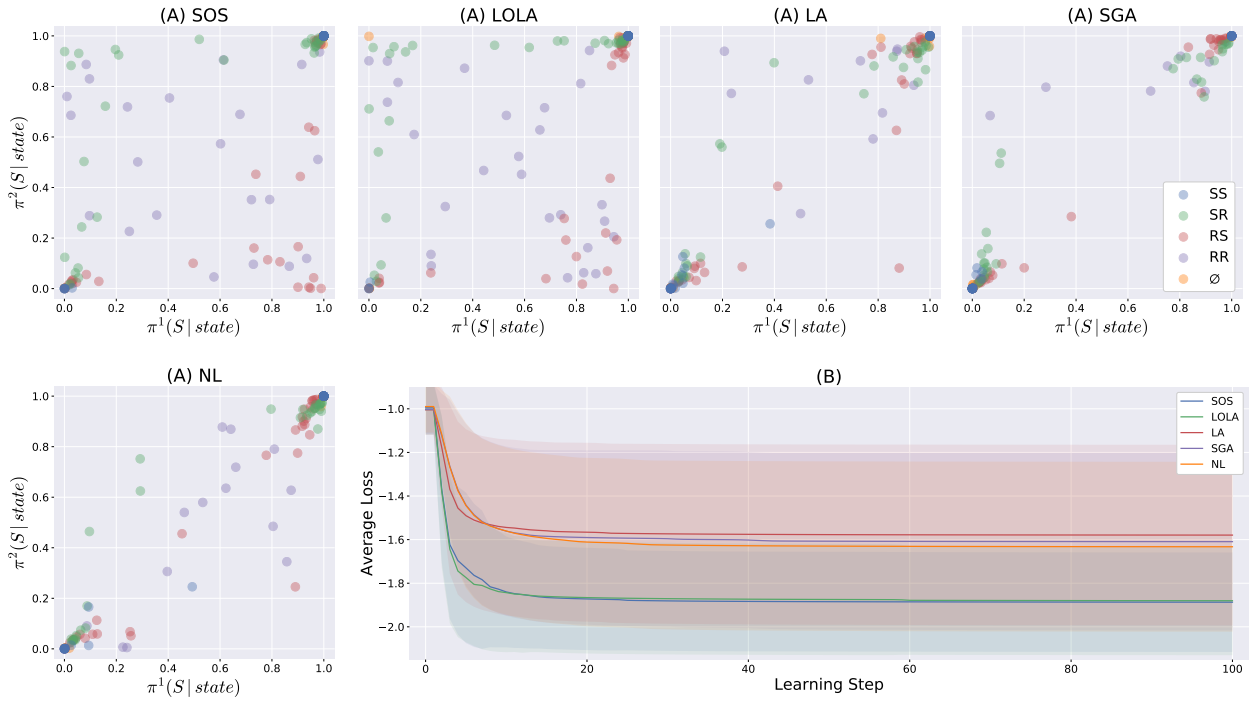

Figure 5.3: Results in the ISH. (A) Probability that SOS, LOLA, LA, SGA, NL agents hunt the stag, given memory state, at the end of 50 training runs. SOS and LOLA have more noisy parameters but more often play tit-for-tat. (B) Average loss at each learning step with shaded standard deviations, across 300 runs. Only 100 steps are shown for visibility.

The results for ISH are given in Figure 5.3. Recall that parameters in (A) are the end-run probabilities of hunting a stag for each memory state, encoded by different colours.

SOS and LOLA reach significantly lower losses than SGA/LA/NL, as displayed in (B). Unlike the IPD, this is less evident from the parameter plots in (A), which are much noisier. The percentage of points in the top right corner (both agents hunting stag) is displayed in Table 5.4, which is only $5-10$ points higher for SOS/LOLA than LA/SGA/NL. Perhaps ISH is more sensitive to random initialisation than IPD, whereby hunters lose all incentive to hunt the stag if parameters begin with high probability of rabbit-hunting.

Note however that tit-for-tat policy reaches equally good losses of -2, which SOS/LOLA do reach more often. This is displayed in (A) with the presence of red points in the lower right and green points in the top right corners – though purple points are all over the place.

Finally recall that p is chosen in SOS according to the dual criterion. We plot the average p at each learning step for each game, across all 300 episodes, in Figure 5.4. Note that p decays to zero in each of the games. This is to be expected, since the second part of the criterion scales p down as SOS converges to fixed points.

	IMP		IPD		ISH	
	Mean(std)	%Nash	Mean(std)	%TFT	Mean(std)	%SS
SOS	$\mathbf{0(10^{-8})}$	**100.0**	$\mathbf{1.06(0.19)}$	78.9	$\mathbf{-1.90(0.22)}$	**62.9**
LOLA	$\mathbf{0(10^{-8})}$	**100.0**	$1.09(0.23)$	**80.3**	$-1.89(0.24)$	61.7
LA	$0(0.06)$	96.1	$1.93(0.24)$	24.2	$-1.58(0.41)$	52.3
SGA	$0(0.05)$	97.0	$1.77(0.39)$	34.9	$-1.61(0.41)$	57.5
NL	$0(0.48)$	0.00	$2.00(10^{-5})$	20.3	$-1.63(0.39)$	53.2

Table 5.4: Summary results for each iterated game. Mean and standard deviation of losses at the end of 300 training runs, and percentage of convergence to fixed points. Best result in bold.

Figure 5.4: Average p for SOS at each step, with shaded deviations across 300 runs, for each game.

Across all iterated games, we see that SOS performs on par with LOLA in each of them, while outperforming all others. Meanwhile, SOS is stronger than LOLA both theoretically through its convergence guarantees, and practically by converging to SFP in the humility game. To show this more explicitly, we move on to the 'dual game' below where SOS outperforms all algorithms simultaneously.

5.2.2 Dual Game

The results for the dual game are given in Figure 5.5. The parameters in (A) are the end-run (x, y) and (z, w) parameters corresponding to logistic and humility sub-games respectively, encoded by different colours. Recall that the logistic game has two SFP $\bar{\theta}_+$ and $\bar{\theta}_-$ occurring at $(x, y) \approx \pm(5, 5)$, where $\bar{\theta}_+$ gives better losses. On the other hand, the humility game has a line of SFP at $y = 10 - x$ which LOLA fails to preserve, instead overshooting towards $y \approx 11 - x$ with worse losses. The percentage of training runs reaching $\bar{\theta}_+$ and the 'humble' line of SFP is displayed in Table 5.5.

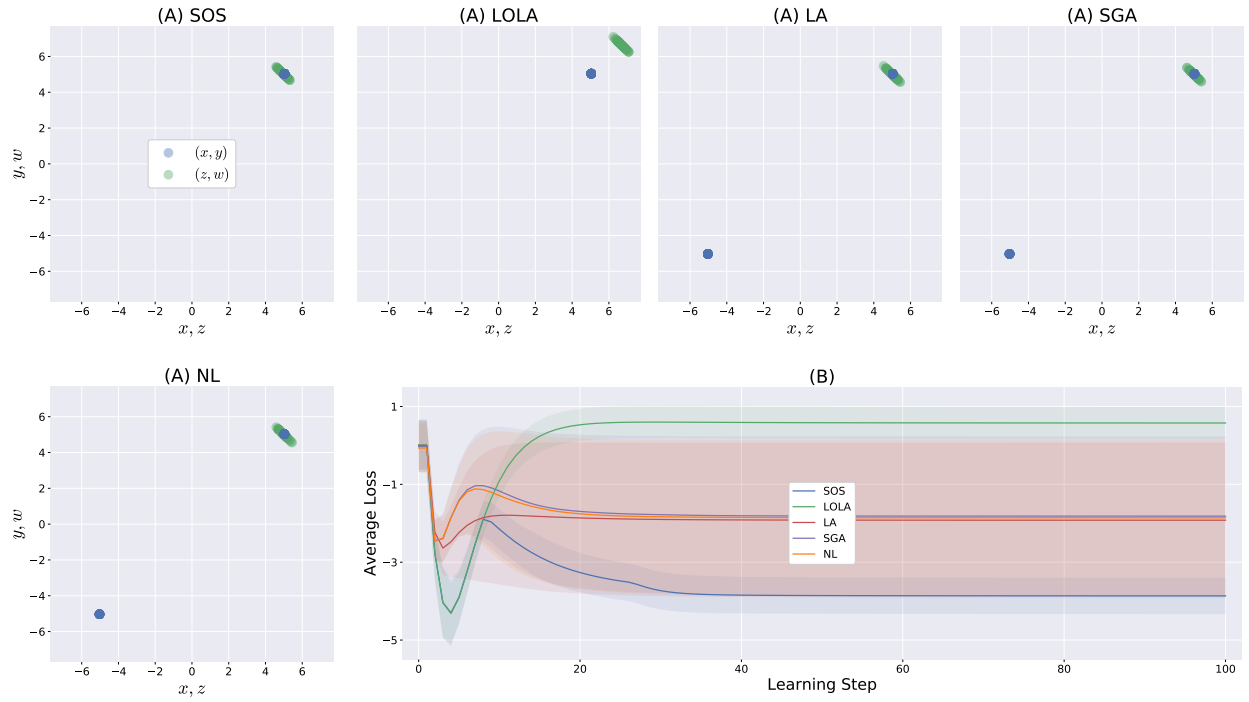

Figure 5.5: Results in the dual game. (A) Parameters for SOS, LOLA, LA, SGA, NL agents at the end of 50 training runs. (B) Average loss at each learning step, across 300 runs. Only 100 steps are shown for visibility.

	Mean(std)	$\%\bar{\theta}_+$	%Humble
SOS	$\mathbf{-3.85(0.46)}$	99.8	**100.0**
LOLA	$0.59(0.41)$	**99.9**	0.0
LA	$-1.90(2.01)$	50.3	**100.0**
SGA	$-1.79(2.01)$	47.7	**100.0**
NL	$-1.89(2.01)$	49.3	**100.0**

Table 5.5: Summary results in the dual game. Mean and standard deviation of losses at the end of 300 training runs, and percentage of convergence to logistic/humility SFP. Best result in bold.

SOS is the only algorithm to succeed in reaching $\bar{\theta}_+$ through opponent shaping, while also staying humble. This is displayed in part (A), where blue points (hidden behind each other) are at $(x, y) \approx (5, 5)$ while green points are on the line $y = 10 - x$. On the other hand, green points for LOLA are on the line $y \approx 11 - x$ through arrogant overshooting. SGA/LA/NL only reach $\bar{\theta}_+$ half of the time, as displayed by the blue points at $(x, y) \approx -(5, 5)$. Outperformance is further manifested in part (B), where losses are lowest for SOS. This is the behaviour we sought to find: a middle ground between stability and exploitation, exemplified in this construction beyond our purely theoretical guarantees.

Chapter 6

Conclusion

Machine learning is increasingly moving from optimising a single loss for a specific task, to dealing with multiple interacting goals at once. Though the choice of architecture is central to both paradigms, success equally hinges on strong optimisation techniques. Naively transposing gradient descent fails, while state-of-the-art algorithms including CO and SGA are either tailored to specific problems (e.g. two-player zero-sum games) or lack strong convergence guarantees. It is additionally unclear which solution concept best generalises local minima in multi-loss optimisation.

The first contribution of this thesis was to clarify the differences between solution concepts. We showed that not all Nash equilibria are desirable, while presenting strong Nash as an ideal but elusive alternative. Instead we argued for the use of stable fixed points as the tractable analogue of local minima by extension from potential games, enabling convergence guarantees in general games.

We reviewed a number of higher-order methods attempting to improve on naive gradient descent. We pointed out that multi-agent RL calls for learning from the perspective of selfish agents, while SGA is "not concerned with the losses of the players *per se*" [Bal]. Instead, SGA and CO both prioritise convergence to equilibria above individual incentive. LOLA overcomes these flaws and is capable of shaping opponent learning. Though intuitively sound, we constructed the first example where failure transpires.

Our third contribution is a number of novel (non-)convergence results on CO, SGA, symmetric and asymmetric lookahead. Ostrowski's Theorem allowed for a unified treatment of each algorithm, though each proof required different techniques ranging from linear algebra and analysis to topology, including a novel similarity transformation trick. We proved local convergence to SFP and non-convergence to unstable FP (generalising to strict saddles) for each algorithm.

LOLA is not locally convergent, but can exploit opponent dynamics to reach better equilibria than its convergent counterparts. This begged the question: can exploitation and stability be coherent? We answered this in Chapter 4 with a resounding *yes*. By interpolating between LOLA and its convergent counterpart lookahead, one can achieve the best of both worlds. The main contribution of this report is SOS, a robust algorithm achieving (non-)convergence in *all* differentiable games – while exploiting opponent dynamics. Convergence of LOLA in two-player zero-sum and n-player cooperative games was obtained as a corollary.

In future work we aim to go beyond game theoretic tasks by applying SOS to deep RL (with policy gradient approximation), GANs and other multi-objective loss training. On the theoretical side, we hope to develop results on divergence from more general saddle fixed points, though more difficult than those for gradient descent in [Lee] [Pan] and perhaps infeasible using dynamical systems. A better understanding of solution concepts including SFP may also be key to understanding the dynamics of multi-loss optimisation. Finally we intend to work on finding an algorithm which provably converges to strong Nash, thus capable of incorporating pure opponent terms while not breaking stability.

Appendix A

Linear Algebra

All matrices in this report are real, so we omit further specification.

Definition A.1. A matrix M is called *positive definite*, written $M \succ 0$, if

$$x^\mathsf{T} M x > 0$$

for all non-zero real vectors x, This is equivalent to

$$\mathrm{Re}(z^* M z) > 0$$

for all non-zero complex vectors z.

Remark A.2. All definitions and results in this appendix can be extended to the 'semi' version by replacing $>$ with $\geq$ everywhere. For instance, a matrix is called positive *semi*-definite, written $M \succeq 0$, if

$$x^\mathsf{T} M x \geq 0$$

for all non-zero real vectors x.

Proposition A.3. A matrix M is positive definite iff

$$x^\mathsf{T} M x > 0$$

for all *unit* real vectors x.

Proof. One direction is trivial since a unit vector is non-zero. Conversely, assume

$$x^\mathsf{T} M x > 0$$

for all unit real vectors $x \in S^{d-1}$, where d is the dimension of M. Now S^{d-1} is compact and the function $g : S^{d-1} \to \mathbb{R}^+$ defined by $g(x) = x^\mathsf{T} M x$ is continuous, so $g(S^{d-1})$ is compact. By Heine-Borel it is closed and bounded, in particular admitting its infimum

$$\epsilon = \inf g(S^{d-1}) > 0\,.$$

Hence

$$x^\mathsf{T} M x \geq \epsilon$$

for all unit x, and for any non-zero y we have

$$y^\mathsf{T} M y = \|y\|^2 \frac{y^\mathsf{T}}{\|y\|} M \frac{y}{\|y\|} \geq \|y\|^2 \epsilon > 0$$

as required. □

Proposition A.4. A matrix M is positive definite iff its symmetric part S is positive definite.

Proof. Decomposing $M = S + A$, we have

$$x^\mathsf{T} A x = (x^\mathsf{T} A x)^\mathsf{T} = x^\mathsf{T} A^\mathsf{T} x = -x^\mathsf{T} A x$$

for any real x since A is antisymmetric, and thus

$$x^\mathsf{T} M x = x^\mathsf{T} S x + x^\mathsf{T} A x = x^\mathsf{T} S x \,.$$ □

Definition A.5. A complex eigenvalue $a + ib$ is called *positive* if $a > 0$.

Definition A.6. A matrix M is called *positive stable* if all its eigenvalues are positive.

Proposition A.7. A positive definite matrix is positive stable.

Proof. Let $\lambda = a + ib$ be any eigenvalue of M, with (normalised) eigenvector v. We have

$$0 < \mathrm{Re}(v^* M v) = \mathrm{Re}(\lambda) = a \,.$$ □

The converse is well-known to hold for symmetric matrices.

Proposition A.8. A symmetric matrix with positive eigenvalues is positive definite.

Proof. Since M is symmetric, there is a diagonalisation $M = PDP^{-1}$ with $P = P^\mathsf{T}$ (orthogonal), whose columns are orthonormal eigenvectors u_i of M and D is diagonal with real eigenvalues $\lambda_i > 0$. Now for any non-zero $x = \sum_i a_i u_i$ we have

$$\begin{aligned} x^\mathsf{T} M x &= \sum_{i,j} a_j a_i u_j^\mathsf{T} P D (P^{-1} u_i) = \sum_{i,j} a_j a_i u_j^\mathsf{T} P D e_i \\ &= \sum_{i,j} a_i a_j u_j^\mathsf{T} P \lambda_i e_i = \sum_{i,j} a_i a_j \lambda_i u_j^\mathsf{T} u_i = \sum_i a_i^2 \lambda_i > 0 \end{aligned}$$

as required. Alternatively note that P orthogonal implies

$$x^\mathsf{T} M x = x^\mathsf{T} P D P^\mathsf{T} x > 0$$

for all non-zero x if and only if

$$y^\mathsf{T} D y > 0$$

for all non-zero y, by change of variable $y = P^\mathsf{T} x$. The latter inequality is trivially true since D is diagonal with positive entries. □

This fails for general (non-symmetric) matrices. For example,

$$M = \begin{pmatrix} 1 & 3 \\ 1 & 1 \end{pmatrix}$$

has positive eigenvalues $1 \pm \sqrt{3}$, while its symmetric part

$$S = \begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$$

has eigenvalues -1 and 3, implying that M is not positive definite.

Proposition A.9. A positive semi-definite, invertible matrix is not necessarily positive definite.

Proof. Consider

$$H = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$$

with symmetric part $S = 0 \succeq 0$. Then H is positive semi-definite by Proposition A.4, while $\det(H) = 1 \neq 0$. Nonetheless, H is not positive definite since S isn't, or alternatively since

$$\begin{pmatrix} 1 & 0 \end{pmatrix} H \begin{pmatrix} 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 \end{pmatrix} H \begin{pmatrix} 0 \\ -1 \end{pmatrix} = 0 \,.$$

□

Proposition A.10. A *symmetric* positive semi-definite, invertible matrix is positive definite.

Proof. A symmetric positive semi-definite matrix has real, real eigenvalues. By invertibility, they are positive and we are done by Proposition A.8. □

Proposition A.11. Let A and B any matrices. Then AB and BA have identical eigenvalues.

Proof. Assume AB has a non-zero eigenvalue

$$ABv = \lambda v \,.$$

Then $Bv \neq 0$ and

$$(BA)Bv = B(ABv) = \lambda Bv \,,$$

so λ is also an eigenvalue of BA. Otherwise, AB has a zero eigenvalue and

$$\det(BA) = \det(B)\det(A) = \det(AB) = 0 \,,$$

so BA also has a zero eigenvalue. The converse argument holds identically, so we are done. □

Appendix B

Analysis and Topology

Proposition B.1. If $\bar{\theta}$ is a Nash equilibrium then $\xi(\bar{\theta}) = 0$ and $\nabla_{ii} L^i(\bar{\theta}) \succeq 0$ for each i.

Proof. By definition of Nash, $(\bar{\theta}^i, \bar{\theta}^{-i})$ is a local minimum of $L^i(\theta^i, \bar{\theta}^{-i})$ as a function of θ^i only. It is an elementary result in analysis that local minima have zero derivative, as can be proved by definition of derivative or Taylor expansion. Hence $\nabla_i L^i(\bar{\theta}^i, \bar{\theta}^{-i}) = 0$ for each i and thus $\xi(\bar{\theta}) = 0$. Now assume for contradiction that for some i, $\nabla_{ii} L^i(\bar{\theta}) \not\succeq 0$. Then there exists a non-zero real vector $u \in \mathbb{R}^{d_i}$ such that

$$u^\intercal \nabla_{ii} L^i(\bar{\theta}) < 0\,.$$

In particular, by Taylor expansion we have

$$\begin{aligned} L^i(\bar{\theta}^i + \epsilon u, \bar{\theta}^{-i}) &= L^i(\bar{\theta}) + \epsilon \nabla_i L^i(\bar{\theta}) \cdot u + \epsilon^2 u^\intercal \nabla_{ii} L^i(\bar{\theta}) u + O(\epsilon^3) \\ &= L^i(\bar{\theta}) + \epsilon^2 u^\intercal \nabla_{ii} L^i(\bar{\theta}) u + O(\epsilon^3) \\ &< L^i(\bar{\theta}) \end{aligned}$$

for small enough $\epsilon > 0$, which contradicts the definition of Nash. Hence $\nabla_{ii} L^i(\bar{\theta}) \succeq 0$ for each i as required. □

Proposition B.2. Let $F : \Omega \to \mathbb{R}^d$ be continuously differentiable on an open subset $\Omega \subseteq \mathbb{R}^d$, and assume $\bar{x} \in \Omega$ is a fixed point. If all eigenvalues of $\nabla F(\bar{x})$ are strictly in the unit circle of $\mathbb{C}$, then there is an open neighbourhood U of $\bar{x}$ such that for all $x_0 \in U$, the sequence $F^{(k)}(x_0)$ converges to $\bar{x}$. Moreover, the rate of convergence is at least linear in k.

The intuition for this result is that dynamics near a fixed point of F are governed primarily by the gradient of F, by Taylor expansion. If the eigenvalues of ∇F are in the unit circle then nearby points are attracted and pulled to the fixed point, resulting in convergence.

Proof. This is a sketch proof following [Ort] closely, with further details in the book. By assumption, the eigenvalues of $\nabla F(\bar{x})$ all satisfy $|\lambda| < \delta$ for some $\delta < 1$. For any $\epsilon > 0$, [Ort, 2.2.8] proves the existence of a norm on $\mathbb{R}^d$ such that

$$\|\nabla F(\bar{x})\| \leq \delta + \epsilon\,.$$

The intuition is that $\nabla F(\bar{x}) = PJP^{-1}$ with Jordan normal form J with diagonal entries λ_i, in turn similar to J^ϵ obtained by replacing off-diagonal 1's by ϵ. Now this matrix has induced norm

$$\|J^\epsilon\|_1 = \max_{1\leq j\leq d} \sum_{i=1}^{d} \left|J^\epsilon_{ij}\right| = |\lambda_i + \epsilon| \leq \delta + \epsilon\,.$$

Finally one can construct the norm $\|x\| \coloneqq \|Px\|_1$ and obtain

$$\|\nabla F(\bar{x})\| \leq \delta + \epsilon$$

also. Further details can be found in [Ort]. In particular, take $\epsilon > 0$ such that $\delta + 2\epsilon < 1$. Now F is continuously differentiable at $\bar{x}$, so in particular Fréchet differentiable and thus

$$\lim_{x\to\bar{x}} \frac{\|F(x) - F(\bar{x}) - \nabla F(\bar{x})(x-\bar{x})\|}{\|x-\bar{x}\|} = 0\,.$$

By definition of limits, there exists a neighbourhood U of $\bar{x}$ such that

$$\|F(x) - F(\bar{x}) - \nabla F(\bar{x})\| \leq \epsilon\|x-\bar{x}\|$$

for all $x \in U$. Now using the assumption $F(\bar{x}) = \bar{x}$ and the triangle inequality, we have

$$\begin{aligned}\|F(x) - \bar{x}\| &= \|F(x) - F(\bar{x})\| \\ &\leq \|F(x) - F(\bar{x}) - \nabla F(\bar{x})(x-\bar{x})\| + \|\nabla F(\bar{x})\|\|x-\bar{x}\| \\ &\leq (\delta + 2\epsilon)\|x-\bar{x}\|\,.\end{aligned}$$

By induction we obtain

$$\|F^k(x_0) - \bar{x}\| \leq (\delta + 2\epsilon)^k \|x - \bar{x}\|$$

with $\delta + 2\epsilon < 1$ and hence

$$\lim_{k\to\infty} F^k(x_0) = \bar{x}$$

for any $x_0 \in U$. Moreover the rate of convergence is at least linear in k since

$$\frac{F^{k+1}(x_0) - \bar{x}}{F^k(x_0) - \bar{x}} \leq (\delta + 2\epsilon) < 1$$

for all k. □

Proposition B.3. Let $g : \mathbb{R}^+ \times Y \to Z$ continuous with Y compact and $Z \subseteq \mathbb{R}$. Assume that for any $u \in Y$ there is $\epsilon_u > 0$ such that

$$g(\alpha, u) > 0$$

for all $0 < \alpha < \epsilon_u$. Then there exists $\epsilon > 0$ such that

$$g(\alpha, u) > 0$$

for all $0 < \alpha < \epsilon$ and $u \in Y$.

Proof. For any $u \in Y$ there is $\epsilon_u > 0$ such that

$$(0, \epsilon_u) \times \{u\} \subseteq g^{-1}(0, \infty)\,.$$

We would like to extend this uniformly in u, namely prove that

$$(0, \epsilon) \times Y \subseteq g^{-1}(0, \infty)\,.$$

for some $\epsilon > 0$. Now $g^{-1}(0, \infty)$ is open by continuity of g, so each $(0, \epsilon_u) \times \{u\}$ has a neighbourhood X_u contained in $g^{-1}(0, \infty)$. Open sets in a product topology are unions of open products, so

$$X_u = \bigcup_x U_x \times V_x\,.$$

In particular $(0, \epsilon_u) \subseteq \bigcup_x U_x$ and at least one V_x contains u, so we can take the open neighbourhood to be $(0, \epsilon_u) \times V_u$ for some neighbourhood V_u of u. In particular

$$Y \subseteq \bigcup_{u \in Y} V_u\,,$$

and by compactness there is a finite cover

$$Y \subseteq \bigcup_{i=1}^{k} V_{u_i}\,.$$

Letting $\epsilon = \min\{\epsilon_i\}_{i=1}^k > 0$, we have

$$\begin{aligned}(0, \epsilon) \times Y &\subseteq (0, \epsilon) \times \bigcup_{i=1}^{k} V_{u_i} \\ &= \bigcup_{i=1}^{k} (0, \epsilon) \times V_{u_i} \\ &\subseteq \bigcup_{i=1}^{k} (0, \epsilon_i) \times V_{u_i} \subseteq g^{-1}(0, \infty)\end{aligned}$$

as required. □

Remark B.4. Another proof idea might be to construct an explicit continuous function $f : Y \to Z$ such that $f(Y) \subset \mathbb{R}^+$ and

$$g(\alpha, u) > 0$$

for all $0 < \alpha < f(u)$. A continuous function on a compact set attains its infimum, so

$$g(\alpha, u) > 0$$

for all $0 < \alpha < \inf_u f(u)$ with $\inf_u f(u) > 0$. Even lower semi-continuity would be sufficient for this argument, namely that $f^{-1}(c, \infty)$ is open for any $c > 0$. Unfortunately it is not obvious how to construct such a function, because ϵ_u does not arise explicitly. One choice might be

$$f(u) = \sup\{\delta > 0 \mid u^\intercal M^{-1} G M u > 0 \text{ for all } 0 < \alpha < \delta\}\,,$$

for which we were not quite able to prove lower semi-continuity. In any case, the proof above is shorter.

Proposition B.5. Let a_k and b_k be sequences of real numbers, and define $c_k = \min\{a_k, b_k\}$. If

$$L = \lim_{k\to\infty} c_k \qquad \text{and} \qquad L' = \lim_{k\to\infty} a_k$$

both exist then $L \leq L'$.

Proof. Assume for contradiction that $L > L'$, then there exists $\delta > 0$ such that $L > L' + \delta$. By definition of limits, there exist $M, N \in \mathbb{N}$ such that

$$|c_k - L| < \delta/2$$

and

$$\left|a_{k'} - L'\right| < \delta/2$$

for all $k \geq M$, $k' \geq N$. Expanding the absolute value, this implies

$$L - \delta/2 < c_k < L + \delta/2 \qquad \text{and} \qquad L' - \delta/2 < a_k < L' + \delta/2$$

for all $k \geq \max\{M, N\}$. Now $c_k \leq a_k$ for all k, hence

$$L - \delta/2 < c_k \leq a_k < L' + \delta/2$$

which implies the contradiction

$$L < L' + \delta\,.$$

□

Proposition B.6. Consider any two-player IBG with discount factor γ and payoff matrices A, B of dimension 2×2. Recall the transition probabilities

$$\theta^{i,j} = \pi^i(a_t = 1 \mid s_t = r_j)$$

for player i and state r_j. Define the parameter vector

$$\theta^i = (\theta^{i,1}, \theta^{i,2}, \theta^{i,3}, \theta^{i,4})^\intercal$$

for each player i, initial distribution

$$p = (\theta^{1,0}\theta^{2,0}, \theta^{1,0}(1-\theta^{2,0}), (1-\theta^{1,0})\theta^{2,0}, (1-\theta^{1,0})(1-\theta^{2,0}))^\intercal$$

and transition matrix

$$P = (\theta^1 \circ \theta^2, \theta^1 \circ (1-\theta^2), (1-\theta^1) \circ \theta^2, (1-\theta^1) \circ (1-\theta^2))\,.$$

Finally define the reward vectors

$$v^1 = \text{flatten}(A)^\intercal \qquad \text{and} \qquad v^2 = \text{flatten}(B)^\intercal$$

where the 'flatten' operator places the rows of a matrix side by side into a single row vector. If R_t^i is the reward of agent i at time t then the loss function associated to this IBG is given by

$$L^i \coloneqq -\sum_t \gamma^t R_t^i = -p^\intercal (I - \gamma P)^{-1} v^i\,.$$

Proof. We follow [Foe, App. A.2] in our own words, with further detail. By definition of P we have

$$P(s_t = r_k \mid s_{t-1} = r_j) = P_{jk}$$

and hence

$$P(s_t = r_k \mid s_0 = r_j) = P^t_{jk}$$

for all $t \geq 1$. It follows immediately that

$$R^i_0 = \sum_k P(s_0 = r_k) v^i_k = \sum_k p_k v^i_k = p^\intercal v^i$$

and

$$\begin{aligned} R^i_t &= \sum_k P(s_t = r_k) v^i_k \\ &= \sum_{j,k} P(s_0 = r_j) P(s_t = r_k \mid s_0 = r_j) v^i_k \\ &= \sum_{j,k} p_j P^t_{jk} v^i_k = p^\intercal P^t v^i \end{aligned}$$

for $t > 0$. The loss function of an iterated game with discount factor γ is given by

$$L^i = -\sum_t \gamma^t R^i_t = -p^\intercal \sum_t \gamma^t P^t v^i \,,$$

and since P is a stochastic matrix we obtain

$$L^i = -p^\intercal (I - \gamma P)^{-1} v^i \,. \qquad \square$$

Bibliography

[Ber] D.P. Bertsekas. *Constrained Optimization and Lagrange Multiplier Methods*. Academic Press, 2014.

[Bal] D. Balduzzi, S. Racaniere, J. Martens, J. Foerster, K. Tuyls, and T. Graepel. The Mechanics of n-Player Differentiable Games. *ICML*, 2018.

[Foe] J. N. Foerster, R. Y. Chen, M. Al-Shedivat, S. Whiteson, P. Abbeel, and I. Mordatch. Learning with Opponent-Learning Awareness. *AAMAS*, 2018.

[Foe2] J. N. Foerster, G. Farquhar, M. Al-Shedivat, T. Rocktäschel, E. P. Xing, and S. Whiteson. DiCE: The Infinitely Differentiable Monte-Carlo Estimator. *ICML*, 2018.

[Fol] G. B. Folland. Higher-Order Derivatives and Taylor's Formula in Several Variables. *https://sites.math.washington.edu/ folland/Math425/taylor2.pdf*, 2005.

[Gem] I. Gemp and S. Mahadevan. Global Convergence to the Equilibrium of GANs using Variational Inequalities. *ArXiv e-prints*, 2018.

[Goo] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative Adversarial Networks. *NIPS*, 2014.

[Jad] M. Jaderberg, W. M. Czarnecki, S. Osindero, O. Vinyals, A. Graves, D. Silver, and K. Kavukcuoglu. Decoupled Neural Interfaces using Synthetic Gradients. *ICML*, 2017.

[Lee] J. D. Lee, M. Simchowitz, M. I. Jordan, and B. Recht. Gradient Descent Only Converges to Minimizers. In *29th Annual Conference on Learning Theory*, volume 49 of *Proceedings of Machine Learning Research*, pages 1246–1257, 2016.

[Mes] L. Mescheder, S. Nowozin, and A. Geiger. The Numerics of GANs. *NIPS*, 2017.

[Mon] D. Monderer and L. S. Shapley. Potential games. *Games and Economic Behavior*, 14(1):124–143, 1996.

[Nes] R. Nessah and G. Tian. On the existence of strong Nash equilibria. *Journal of Mathematical Analysis and Applications*, 414(2):871 – 885, 2014.

[Ort] J. Ortega and W. Rheinboldt. *Iterative Solution of Nonlinear Equations in Several Variables*. Society for Industrial and Applied Mathematics, 2000.

[Pat] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell. Curiosity-driven Exploration by Self-supervised Prediction. *ICML*, 2017.

[Pre] W. H. Press and F. J. Dyson. Iterated Prisoner's Dilemma contains strategies that dominate any evolutionary opponent. *Proceedings of the National Academy of Sciences*, 109(26):10409–10413, 2012.

[Pan] I. Panageas and G. Piliouras. Gradient Descent Only Converges to Minimizers: Non-Isolated Critical Points and Invariant Regions. In *ITCS 2017*, volume 67 of *Leibniz International Proceedings in Informatics*, pages 2:1–2:12, 2017.

[Rac] S. Racanière, T. Weber, D. P. Reichert, L. Buesing, A. Guez, D. Jimenez Rezende, A. Puigdomènech Badia, O. Vinyals, N. Heess, Y. Li, R. Pascanu, P. Battaglia, D. Hassabis, D. Silver, and D. Wierstra. Imagination-Augmented Agents for Deep Reinforcement Learning. *NIPS*, 2017.

[Scu] G. Scutari, D. P. Palomar, F. Facchinei, and J. Pang. Convex Optimization, Game Theory, and Variational Inequality Theory. *IEEE Signal Processing Magazine*, 27(3):35–49, May 2010.

[Zha] C. Zhang and V. Lesser. Multi-Agent Learning with Policy Prediction. *AAAI Conference on Artificial Intelligence*, 2010.