
Proceedings Paper:
Loftin, R., Çelikok, M.M., van Hoof, H. et al. (2 more authors) (2024) Uncoupled learning of differential Stackelberg equilibria with commitments. In: Proceedings of AAMAS-2024. The 23rd International Conference on Autonomous Agents and Multi-Agent Systems (AAMAS 2024), 06-10 May 2024, Auckland, New Zealand. International Foundation for Autonomous Agents and Multiagent Systems (IFAAMAS) , pp. 1265-1273. ISBN 978-1-4007-0486-4

Uncoupled Learning of Differential Stackelberg Equilibria with Commitments

Robert Loftin*
The University of Sheffield
Sheffield, United Kingdom
r.loftin@sheffield.ac.uk

Mustafa Mert Çelikok*
Delft University of Technology
Delft, The Netherlands
m.m.celikok@tudelft.nl

Herke van Hoof
University of Amsterdam
Amsterdam, The Netherlands
h.c.vanhoof@uva.nl

Samuel Kaski
Aalto University
Helsinki, Finland
The University of Manchester
Manchester, United Kingdom
samuel.kaski@aalto.fi

Frans A. Oliehoek
Delft University of Technology
Delft, The Netherlands
f.a.oliehoek@tudelft.nl

ABSTRACT

In multi-agent problems requiring a high degree of cooperation, success often depends on the ability of the agents to adapt to each other's behavior. A natural solution concept in such settings is the Stackelberg equilibrium, in which the "leader" agent selects the strategy that maximizes its own payoff given that the "follower" agent will choose their best response to this strategy. Recent work has extended this solution concept to two-player differentiable games, such as those arising from multi-agent deep reinforcement learning, in the form of the *differential* Stackelberg equilibrium. While this previous work has presented learning dynamics which converge to such equilibria, these dynamics are "coupled" in the sense that the learning updates for the leader's strategy require some information about the follower's payoff function. As such, these methods cannot be applied to truly decentralised multi-agent settings, particularly ad hoc cooperation, where each agent only has access to its own payoff function. In this work we present "uncoupled" learning dynamics based on zeroth-order gradient estimators, in which each agent's strategy update depends only on their observations of the other's behavior. We analyze the convergence of these dynamics in general-sum games, and prove that they converge to differential Stackelberg equilibria under the same conditions as previous coupled methods. Furthermore, we present an online mechanism by which symmetric learners can negotiate leader-follower roles. We conclude with a discussion of the implications of our work for multi-agent reinforcement learning and ad hoc collaboration more generally.

KEYWORDS

multi-agent reinforcement learning; ad hoc collaboration; ad hoc teamwork; learning dynamics; differentiable games; differential stackelberg equilibrium

*Equal contribution.

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

ACM Reference Format:
Robert Loftin, Mustafa Mert Çelikok, Herke van Hoof, Samuel Kaski, and Frans A. Oliehoek. 2024. Uncoupled Learning of Differential Stackelberg Equilibria with Commitments. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 9 pages.

1 INTRODUCTION

A central goal of multi-agent systems research has been to understand the long-term behavior of *autonomous* learning agents that optimize their individual strategies through repeated interaction. The challenge here is that the agents do not have any direct control over each other, cannot see into to the "minds" of others, and must learn based solely on their observable behavior. In this work, we focus on this problem in collaborative settings, where agents benefit from cooperation, though they may not share the same payoff functions [13, 29]. An important example of such a setting is *ad hoc teamwork* [33, 47], where agents that have never encountered one another before must learn to collaborate without any prior coordination. Such settings might arise when different companies create their own learning agents to interact with other agents or with humans. These agents will need to collaborate, bargain, and negotiate with each other, but would definitely prefer keeping their utilities and internal learning mechanisms private.

Theoretical analysis of multi-agent learning dynamics allows us to determine if and when autonomous learning agents will converge to a fixed joint strategy, characterize the strategies they are likely to converge to, and ultimately design improved learning algorithms (e.g. [3, 11, 20]). Additionally, recent years have seen a surge of interest in the dynamics of multi-agent learning, driven by the recognition that many machine learning problems (such as actor-critic methods in RL [49]) can be formulated as games with continuous, high-dimensional strategy spaces and differentiable payoff functions. Results on such *differentiable* games have also found applications to multi-agent reinforcement learning [2].

In this work we consider the problem of finding *hierarchical* solutions in two-player, general-sum differentiable games. In the hierarchical model of play, the "leader" player selects their strategy first, after which the other player (the "follower") selects their best-response to this strategy. The natural solution concept for the hierarchical model is the *Stackelberg equilibrium* (SE), in which the

leader's strategy is optimal under the assumption that the follower will play their best response to whatever strategy the leader chooses. The hierarchical model is well suited to cooperative settings, where the leader can play their half of an optimal joint strategy knowing that the follower will respond appropriately.

It has recently been argued that the Stackelberg equilibrium is also a more useful solution concept for differentiable games than the Nash equilibrium (NE), as the SE exists in games where the NE does not [22]. This fact has motivated the development of "hierarchical" gradient ascent methods for finding *differential* Stackelberg equilibria (DSE), the local analogue of SE, in differentiable games. In particular, Fiez et al. [15] have presented a hierarchical gradient update that is shown to converge to DSE in certain differentiable games. However, this *coupled* learning update is designed with *centralized training* in mind, and the leader's update requires knowledge of the follower's payoff function. Such coupled learning methods cannot be applied to independent learning settings, where the other agent's payoff function is unknown. The coupled hierarchical update also requires the Hessian of the follower's payoff function, and so may be computationally intractable in settings where second-order derivatives are expensive to estimate (such as reinforcement learning). Finally, as with most existing literature, this learning update assumes that the roles of leader and follower are assigned beforehand (a form of prior coordination).

The main contribution of this work is a novel *uncoupled* learning update called *Hierarchical learning with Commitments* (Hi-C), which does not require the leader to have access to the follower's payoff function or learning algorithm. Hi-C estimates the leader's gradient update by sampling strategies close to the leader's current strategy, and then committing to these "perturbed" strategies long enough that the follower has time to adapt to them. As such, unlike previous coupled methods, the Hi-C update is applicable to fully independent multi-agent learning settings such as ad hoc teamwork. As an added benefit, our method is a tractable alternative to coupled hierarchical updates for problems where estimating the higher-order derivatives of the payoff functions is possible but impractical. Our main theoretical results show that, under the same conditions as previous coupled methods, Hi-C converges to a DSE for the leader as long as the follower's own strategy converges to its best response sufficiently fast. We mathematically specify what sufficiently fast means in this context, and as an illustrative example derive a commitment schedule for the case where the follower's payoff function is strongly concave. Furthermore, we introduce a mechanism by which agents can "negotiate" the leader–follower role assignment in an online fashion. This allows symmetric learners to negotiate their roles while simultaneously solving the underlying differentiable game. To our knowledge, this is the first negotiation process that addresses the open question, presented in Basar [4], of determining roles online in hierarchical play.

2 BACKGROUND

We consider the class of two-player, general-sum differentiable games. Let $\mathcal{X} \subseteq \mathfrak{R}^{d_1}$ and $\mathcal{Y} \subseteq \mathfrak{R}^{d_2}$ be the strategy spaces for players 1 and 2 respectively. In hierarchical play, we let player 1 be the leader and player 2 the follower, unless stated otherwise. Let $f_i : \mathcal{X} \times \mathcal{Y} \mapsto \mathfrak{R}$ be the payoff function of player i, with $f_i \in C^2(\mathcal{X} \times \mathcal{Y}, \mathbb{R})$ for all $i \in \{1, 2\}$ (i.e. f_i is twice continuously differentiable). Let $\nabla_x f_i(x, y)$ and $\nabla_y f_i(x, y)$ denote the gradients of f_i w.r.t. player 1 and player 2's strategies respectively. We denote by $\nabla_{xy} f_i(x, y) = \nabla_y[\nabla_x f_i(x, y)]$ the Jacobian of the gradient $\nabla_x f_i(x, y)$ w.r.t. y, and define $\nabla_{yx} f_i(x, y)$, $\nabla_{xx} f_i(x, y)$, and $\nabla_{yy} f_i(x, y)$ similarly. Finally we let $\|\cdot\|$ denote the Euclidean norm throughout.

When analysing learning dynamics, the "pure" strategies x and y of the differentiable game can be thought of as representing the parameters of the (potentially stochastic) strategies followed by the agents in some underlying game. For instance, any $N \times N$ matrix game can be described as a differentiable game by choosing $\mathcal{X}$ and $\mathcal{Y}$ to be the N-dimensional probability simplices. Then $f_i(x, y)$ would be the expected payoff for i in the matrix game under the mixed strategy profile $\langle x, y \rangle$. Deep reinforcement learning agents in a Markov game can be represented similarly, where x and y would be the parameters of neural networks representing stochastic policies. The learning dynamics therefore capture how the players update their mixed strategies / stochastic policies over time.

2.1 Simultaneous Gradient Ascent and Differential Nash Equilibria

A straightforward approach to solving differentiable games is *simultaneous gradient ascent* (SGA), where player i performs gradient ascent on its own payoff function f_i, treating other players' strategies as fixed. The two-player SGA updates are defined as

$$x_{t+1} = x_t + \alpha_{1,t} \nabla_x f_1(x_t, y_t) \qquad y_{t+1} = y_t + \alpha_{2,t} \nabla_y f_2(x_t, y_t), \quad (1)$$

where the sequences $\{\alpha_{1,t}\}$ and $\{\alpha_{2,t}\}$ are learning rate schedules, which may differ between the players. SGA is often the default approach for problems described by two-player games (such as training GANs [18]). We can also view SGA as a model of ad hoc learning between independent agents. In the ad hoc setting, players are only aware of their own payoff functions, and the strategies other players follow at each *stage t* of the game.

However, as it is generally the case for gradient-based learners, we cannot expect SGA to always find global optima in the strategy space of either player. This motivates the development of *local* alternatives, including the differential Nash equilibrium (DNE).

Definition 2.1 (Differential Nash Equilibrium [41])**.** Let $\omega(x, y) = (\nabla_x f_1(x, y), \nabla_y f_2(x, y))$ be the individual gradients of the players' payoff functions at (x, y). A strategy profile $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ is a differential Nash equilibrium if **(I)** $\omega(x^*, y^*) = 0$, **(II)** $\nabla_{xx} f_1(x^*, y^*)$ and $\nabla_{yy} f_2(x^*, y^*)$ are negative definite.

Previous work has shown that gradient-based learning dynamics such as SGA can converge to DNE in specific classes of games [23, 41]. However, the main issue with DNEs is that they fail to exist in some games, which constrains the class of games they are applicable to. For instance, Nash equilibria exist for convex costs (i.e. concave payoffs) on compact and convex strategy spaces, and DNE exists if these conditions, as described in Başar and Olsder [7, Theorem 4.3 & Chapter 4.9], are met locally within the neighbourhoods DNE are defined [14]. An alternative local solution concept, discussed below, based on Stackelberg equilibria exists in more relaxed conditions, and is thus applicable to a wider class of games.

2.2 Hierarchical Model and Differential Stackelberg Equilibria

In the hierarchical model, the leader selects a strategy first, and the follower selects the best response to the leader's strategy. Thus, the natural solution concept in the hierarchical play is the Stackelberg equilibrium, in which the leader chooses a strategy that maximizes its payoff under the follower's best response.

Definition 2.2 (Stackelberg Equilibrium (SE) [44])**.** Let the set $\mathrm{BR}(x) = \{y \mid f_2(x,y) = \max_{y' \in \mathcal{Y}} f_2(x,y')\}$ denote the follower's set of best-responses when the leader plays x. A joint strategy $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ is a *Stackelberg equilibrium* if $y^* \in \mathrm{BR}(x^*)$ and

$$\min_{y \in \mathrm{BR}(x^*)} f_1(x^*, y) \geq \min_{y \in \mathrm{BR}(x)} f_1(x, y) \tag{2}$$

for all $x \in \mathcal{X}$. Such an x^* is a *Stackelberg solution* for the leader.

Note that for the SE to be well-defined, the follower must have a tie-breaking mechanism, and definition 2.2 assumes that the follower breaks ties so as to minimize the leader's payoffs. Therefore, a Stackelberg solution maximizes the leader's worst-case payoff assuming the follower will act rationally, and in zero-sum games the Stackelberg solution guarantees the leader will receive at least its security value. Recent work [15, 22] has shown that the hierarchical model can be applied to differentiable games as well. While a differentiable game may possess no Nash equilibria, a Stackelberg equilibrium will always exist so long as the strategy spaces $\mathcal{X}$ and $\mathcal{Y}$ are compact [7, Theorem 4.8 & Chapter 4.9]. Algorithms based on the hierarchical model have proven successful in training generative adversarial networks [14, 32] and actor–critic methods [49].

Definition 2.2 assumes that both the leader and the follower have found *global* optima in their respective strategy spaces. As gradient-based learners cannot guarantee convergence to a global optimum in general, applying the hierarchical model to differentiable games requires a local version of the SE referred to as the *differential Stackelberg equilibrium* (DSE) [14]. The definition of DSE starts with the following observation. Given a point $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ such that $\nabla_y f_2(x^*, y^*) = 0$ and $det(\nabla_{yy}[f_2(x^*, y^*)]) \neq 0$, there exists a continuously differentiable local best-response function $r : \mathcal{U}_1 \mapsto \mathcal{Y}$ for the follower, where $\mathcal{U}_1 \subset \mathcal{X}$ is a neighbourhood of (x^*, y^*). Under this notation, the leader's objective function becomes $f_1(x, r(x))$, and so a local optimum x^* for the leader will satisfy $\nabla_x[f_1(x, r(x))] = 0$, where $\nabla_x[f_1(x, r(x))] = \nabla_x f_1(x, r(x)) + [\nabla_y f_1(x, r(x)]^\top \nabla_x r(x)$. Then, a DSE is defined as follows.

Definition 2.3 (Differential Stackelberg Equilibrium [14])**.** A strategy profile $(x^*, y^*) \in \mathcal{X} \times \mathcal{Y}$ with $r(x^*) = y^*$, is a differential Stackelberg equilibrium (DSE) if: **(I)** $\nabla_x[f_1(x^*, r(x^*))] = 0$ and $\nabla_y[f_2(x^*, y^*)] = 0$, and **(II)** $\nabla_{xx}[f_1(x^*, r(x^*))]$ and $\nabla_{yy}[f_2(x^*, y^*)]$ are negative definite. Furthermore, any x^* satisfying these conditions is a differential Stackelberg solution (DSS) for the leader.

Intuitively, the condition **(II)** ensures that x^* and y^* are local maxima of the player's individual objectives, rather than minima or saddle points. Note that conditions **(I)** and **(II)** do not imply that $\nabla_x f_1(x^*, y^*) = 0$, and so DSE may not always be stable under gradient ascent on f_1. In fully-cooperative games, all agents have the same reward function (i.e. $f_1 = f_2$). In that case, we have the following proposition, which states that learning the DSE instead of DNE does not break results for the fully-cooperative case.

Proposition 2.4 (Fully-cooperative Multi-agent RL and DSE)**.** *Differential Stackelberg equilibria and differential Nash equilibria are equivalent in fully-cooperative games where $f_1 = f_2$.*

2.3 Hierarchical Gradient Update

While a natural approach to finding DSE is to perform gradient ascent on the leader's objective function $f_1(x, r(x))$, we will generally not have a closed-form expression for $r(x)$. Fortunately, for a joint strategy $(x, y) \in \mathcal{X} \times \mathcal{Y}$ for which $y = r(x)$ and $\nabla_{yy} f_2(x, y)$ is non-singular, the implicit function theorem provides a closed-form expression for $\nabla_x r(x)$ as a function of x and y [15]. When $y = r(x)$, we have $\nabla_x r(x) = -\left(\nabla_{yy} f_2(x, y)\right)^{-1} \nabla_{xy} f_2(x, y)$. The gradient of the leader's objective then becomes

$$\begin{aligned} \nabla_x[f_1(x, r(x))] &= \nabla_x f_1(x, y) \\ &\quad - \nabla_y f_1(x, y)^\top \left(\nabla_{yy} f_2(x, y)\right)^{-1} \nabla_{xy} f_2(x, y) \\ &= D(x, y) \end{aligned} \tag{3}$$

Evaluating 3 requires knowing the value of $y = r(x)$. One way to compute the leader's gradient update is then to optimize y via gradient ascent on f_2 while keeping x constant, and allowing y to converge to $r(x)$ before performing each gradient step for the leader's strategy. Fiez et al. [15] present a more practical, two-timescale algorithm in which the leader and follower strategies are updated simultaneously, with the follower using a faster learning rate than the leader. When we may only have noisy estimates of the gradients, the two-timescale hierarchical gradient updates become:

$$\begin{aligned} x_{t+1} &= x_t + \alpha_{1,t}(D(x_t, y_t) + w_{1,t}) \\ y_{t+1} &= y_t + \alpha_{2,t}(\nabla_y f_2(x_t, y_t) + w_{2,t}), \end{aligned} \tag{4}$$

where $\{w_{1,t}\}$ and $\{w_{2,t}\}$ are independent zero-mean noise sequences, and the leader's update $D(x, y)$ is defined as in Equation 3. To achieve time-scale separation, the learning rate schedules are chosen so that $\alpha_{2,t} \gg \alpha_{1,t}$, which allows the follower's strategy to "track" its best response to the leader's current strategy. If the limit condition $\lim_{t \to \infty} \frac{\alpha_{1,t}}{\alpha_{2,t}} = 0$ holds, then results on two-timescale stochastic approximation (see [9, Chapter 6.1]) can be used to analyze the convergence properties of 4.

2.4 Limitations of Coupled Learning

We can see that the explicit form of the leader's update in Equation 3 depends on the Hessian $\nabla_{yy} f_2$ of the follower's payoff function, which implies that the leader must know the structure of f_2. This assumption does not hold in ad hoc cooperation, where the leader only has access to the follower's observable behavior. Even in centralized settings where the leader can estimate the gradient and Hessian of f_2 directly, this estimation can be expensive and suffer from high variance. This is particularly true in settings such as reinforcement learning, where gradients (and Hessians) must be estimated through Monte-Carlo simulations. Other learning updates such as LOLA also depend on estimates of the follower Hessian [16], and so suffer form the same limitations. In this section, we remove the limitation of coupled learning by introducing a learning algorithm that estimates $\nabla_x[f_1(x, r(x))]$ from the follower's behavior alone, while maintaining similar convergence guarantees to the two-timescale hierarchical gradient update.

3 UNCOUPLED LEARNING WITH COMMITMENTS

Algorithm 1 The Hi-C learning algorithm, with follower strategies y_t chosen arbitrarily, and with w_n being some zero-mean noise. $t(n) = \sum_{m=0}^{n-1} k_m$ is the stage at which interval n started.

Inputs: Step-sizes $\{\alpha_n\}_{n\geq 0}$, perturbation schedule $\{\delta_n\}_{n\geq 0}$, commitment schedule $\{k_n\}_{n\geq 0}$.
Initialize: sample x_0 from $\mathcal{X}$
for step $n = 0, 1, \ldots$ **do**
  sample Δ_n uniformly from $\{-1, 1\}^{d_1}$.
  $\tilde{x}_n \leftarrow x_n + \delta_n \Delta_n$
  for stage $t = t(n), \ldots, t(n) + k_n - 1$ **do**
    play $\tilde{x}_n$.
    observe $\tilde{y}_n \leftarrow y_t$.
  end for
  for dimension $i = 1, \ldots, d_1$ **do**
    $x^i_{n+1} = x^i_n + \frac{\alpha_n}{\delta_n \Delta^i_n}[f_1(\tilde{x}_n, \tilde{y}_n) + w_n]$
  end for
end for

From the leader's perspective, the problem of finding a differential Stackelberg equilibrium is simply that of finding a local maximum of $f_1(x, r(x))$, where $r(x)$ is the follower's best response when the leader chooses x as their strategy. The challenge in the *uncoupled* setting is that the leader cannot evaluate $\nabla_x[f_1(x, r(x))]$ directly, since it cannot evaluate the Jacobian $\nabla_x r(x)$ as it does not have access to the follower's payoff function f_2 on which $r(x)$ depends. The leader can, however, estimate the value of $r(x)$ (and therefore $f_1(x, r(x))$) by simply observing the follower's response when it plays strategy x. A natural approach then is to replace gradient ascent with a gradient-free learning rule that only requires an unbiased estimate of $f_1(x, r(x))$, and not of $\nabla_x[f_1(x, r(x))]$.

We first consider the hypothetical case where the leader has access to an *oracle* for $r(x)$. This oracle allows the leader to evaluate $f_1(x, r(x))$ for any $x \in \mathcal{X}$. We can then apply *simultaneous perturbation stochastic approximation* (SPSA) [45] to approximate gradient ascent on $f_1(x, r(x))$. Specifically, we will derive Hi-C from the one-sample form of SPSA [46]. For all $n \geq 0$, let Δ_n be independently and uniformly sampled from $\{-1, 1\}^{d_1}$, and let $\{\delta_n\}_{n\geq 0}$ be a decreasing *perturbation schedule*. Let $\{w_n\}_{n\geq 0}$ be a sequence of i.i.d. variables (with zero-mean and uniformly bounded variance) representing noise in the evaluation of f_1. The element-wise one-sample SPSA update is then

$$x^i_{n+1} = x^i_n + \alpha_n \frac{f_1(x_n + \delta_n \Delta_n, r(x_n + \delta_n \Delta_n)) + w_n}{\delta_n \Delta^i_n} \tag{5}$$

for all $i \in [1, d_1]$. SPSA estimates the direction of the gradient by sampling points near the current strategy x_n. Going forward, let $\tilde{x}_n = x_n + \delta_n \Delta_n$ denote the "perturbed" strategy evaluated at step n. The noise terms w_t account for settings the leader can only observe an unbiased estimator of f_1 (e.g., a single policy roll-out).

Estimating $r(\tilde{x}_n)$. In truly uncoupled settings the leader has no way of directly computing $r(x_n)$. What the leader can do is observe the strategies played by the follower, which presumably updates its own strategy so as to maximize its payoff under f_2. This suggests an asynchronous, two-timescale learning process, in which the leader *commits* to playing the perturbed strategy $\tilde{x}_n$ for some k_n stages before updating x_n. For sufficiently large k_n we should hope that after k_n stages the follower's strategy will have approximately converged to its best-response $r(\tilde{x}_n)$.

Under the Hi-C learning update (Algorithm 1), at each interval $n \geq 0$ the leader samples a perturbed strategy $\tilde{x}_n$, and then plays this strategy for the next k_n stages. After k_n stages, the leader updates its strategy element-wise as

$$x^i_{n+1} = x^i_n + \alpha_n \frac{f_1(\tilde{x}_n, \tilde{y}_n) + w_n}{\delta_n \Delta^i_n} \tag{6}$$

where the follower's final strategy within interval n, denoted by $\tilde{y}_n$, is used as an estimate of $r(\tilde{x}_n)$. This gradient estimator has bounded variance, since f_1 is bounded on $X \times Y$ and w_n is i.i.d. noise with bounded variance. To reduce variance we can optionally use $f_1(x_t, \tilde{y}_{n-1})$ as a baseline value, as $\tilde{y}_{n-1}$ is independent of Δ_n.

3.1 Convergence Analysis

In this section we make no assumptions about the specifics of the follower's learning update or their payoff function, and instead prove convergence of the leader's strategy under a generic assumption about the convergence rate of the "tracking error" $\|\tilde{y}_n - r(\tilde{x}_n)\|$ between the follower's strategy and its best-response (asm. 3.5). In general, this assumption can be satisfied by choosing a long enough commitment schedule for the leader. As an illustrative example, in Section 3.2 we derive a commitment schedule $\{k_n\}_{n\leq 0}$ for followers with strongly concave payoff functions that ensures the tracking error will decrease fast enough to satisfy this assumption.

In Hi-C, the follower is assumed to update their strategy at every stage t, while the leader only performs an update after k_n stages. Let $t(n) = \sum_{m=0}^{n-1} k_m$ be the stage at which the leader begins its nth commitment interval, and let $n(t) = \max\{n : t(n) \leq t\}$ be the current interval at stage t. We let x_n ($n \geq 0$) denote the leader's *mean* strategy after n updates, or $t(n)$ stages, and let y_t denote the strategy the follower played at stage t. We then have $\tilde{y}_{n+1} = y_{(t(n)+k_n-1)}$, the last strategy the follower played during the nth commitment interval. We prove convergence of Hi-C under assumptions that are standard in the analysis of previous work from simultaneous perturbation methods [8, Chapter 5] and differential Stackelberg equilibria (asm. 3.1, 3.2, 3.3, and 3.4).

Assumption 3.1. There exists a unique best-response function $r : \mathcal{X} \mapsto \mathcal{Y}$ that maps leader strategies to follower's best-responses. Furthermore, r is L_r-Lipschitz and K_r-smooth.

Note that the assumption 3.1 does not restrict the follower's payoff function to have a unique optimizer for a given x, but simply requires that the follower breaks ties in an arbitrary yet fixed way. This is a common assumption in hierarchical play, and it is needed for the SE to be well-defined. The leader also does not make any assumptions on how the follower breaks ties; the tie-breaking mechanism is abstracted away into the follower's best-response function r, which is estimated from observed behaviour.

Assumption 3.2. x_n and y_t are bounded almost surely:

$$\sup_{n\geq 0} \|x_n\| < \infty \quad \text{and} \quad \sup_{t\geq 0} \|y_t\| < \infty \quad \text{a.s.} \tag{7}$$

This immediately implies that $\tilde{x}_n$ and $\tilde{y}_n$ are bounded a.s., and because r is Lipschitz, it implies $r(\tilde{x}_n)$ is bounded a.s. as well. In practice, the assumption that the strategies remain bounded can be enforced by choosing $\mathcal{X}$ and $\mathcal{Y}$ to be bounded, and projecting the strategies back to $\mathcal{X}$ and $\mathcal{Y}$ whenever necessary.

Assumption 3.3. The leader's payoff function $f_1(x, y)$ is L_1-Lipschitz, and K_1-smooth in both of its arguments.

Assumption 3.3 implies that $\|\nabla_y f_1(x, y)\| \leq L_1$ for all $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Combined with Assumption 3.1, it also implies that the leader's hierarchical objective function $g(x) = f_1(x, r(x))$ is also Lipschitz and smooth.

Assumption 3.4. The step-size schedule $\{\alpha_n\}_{n\geq 0}$ and perturbation schedule $\{\delta_n\}_{n\geq 0}$ satisfy

$$\lim_{n\to\infty} \alpha_n = 0, \quad \lim_{n\to\infty} \delta_n = 0, \quad \sum_{n=0}^{\infty} \alpha_n = \infty, \quad \sum_{n=0}^{\infty} \frac{\alpha_n^2}{\delta_n^2} < \infty. \tag{8}$$

The decreasing magnitude δ_n of the perturbations means that eventually even small errors in the approximation of $r(\tilde{x}_n)$ could lead to large errors in the estimate of the gradient. Therefore, other than the standard assumptions listed above, the following generic assumption on the rate of convergence of the follower's tracking error must be satisfied via an appropriate commitment schedule.

Assumption 3.5. Define $\varepsilon_n = \|\tilde{y}_n - r(\tilde{x}_n)\|$, for $n \geq 0$. The commitment and perturbation schedules $\{k_n\}_{n\geq 0}$ and $\{\delta_n\}_{n\geq 0}$ satisfy

$$\lim_{n\to\infty} \frac{\varepsilon_n}{\delta_n} = 0 \quad \text{and} \quad \sup_{n\geq 0} \frac{\varepsilon_n}{\delta_n} < \infty \quad \text{a.s.} \tag{9}$$

Under Assumption 3.5, the error introduced by using $\tilde{y}_n$ rather than $r(\tilde{x}_n)$ is bounded and $o(1)$ almost surely, and so becomes negligible asymptotically. To see this, we rewrite Equation 6 as

$$x_{n+1}^i = x_n^i + \alpha_n \left(\frac{f_1(\tilde{x}_n, r(\tilde{x}_n)) + w_t}{\delta_n \Delta_n^i} + \eta_n^i \right) \tag{10}$$

where

$$\eta_n^i = \frac{f_1(\tilde{x}_n, \tilde{y}_{n+1}) - f_1(\tilde{x}_n, r(\tilde{x}_n))}{\delta_n \Delta_n^i} \leq \frac{L_1 \|\tilde{y}_n - r(\tilde{x}_n)\|}{\delta_n \Delta_n^i} \tag{11}$$

since f_1 is L_1-Lipschitz by Assumption 3.3. We then have

$$\sup_{n\geq 0} |\eta_n^i| \leq \sup_{n\geq 0} \frac{L_1 \|\tilde{y}_n - r(\tilde{x}_n)\|}{\delta_n} = \sup_{n\geq 0} L_1 \frac{\varepsilon_n}{\delta_n} < \infty \quad \text{a.s.} \tag{12}$$

where the final inequality comes from the fact that $\sup_{n\geq 0} \frac{\varepsilon_n}{\delta_n}$ is bounded almost surely by Assumption 3.5. We can also see that $\lim_{n\to\infty} |\eta_n^i| = 0$ almost surely, as $\frac{\varepsilon_n}{\delta_n} \to 0$ a.s.. We are now ready to state our main convergence result:

Theorem 3.6. *Let $H \subseteq \mathcal{X}$ be the set $\{x \in \mathcal{X} : \nabla_x [f_1(x, r(x))] = 0\}$. Assume $H \neq \emptyset$, and that Assumptions 3.1–3.5 are satisfied under the Hi-C update (Algorithm 1). Then the leader's strategy x_n will converge to H almost surely as $n \to \infty$.*

Proof sketch: This result follows from Bhatnagar et al. [8, Theorem 5.2] by noting that the Hi-C update in Equation 6 is equivalent to the single measurement SPSA update (Equation 5) save for the bounded, $o(1)$ error term η_n^i, which becomes negligible asymptotically (see [9, Chapter 2]. This result shows that the Hi-C update converges to a relatively small subset of $\mathcal{X}$ that, if they exist, contains the differential Stackelberg solutions. With further assumptions on H, we can show that x_n converges to a DSS almost surely.

Corollary 3.7. *Additionally, assume that H consists only of isolated, asymptotically stable equilibria of the ODE $\dot{x}(t) = \nabla_x [f(x(t), r(x(t)))]$. Then, under the Hi-C update, x_n will converge to a differential Stackelberg solution of the game (f_1, f_2) almost surely as $n \to \infty$.*

This follows from the fact that if $x \in H$ is an asymptotically stable equilibrium of $\dot{x}(t) = \nabla_x [f(x(t), r(x(t)))]$, then the Hessian $\nabla_{xx} [f(x(t), r(x(t)))]$ must be negative definite. Combined with $\nabla_x [f(x(t), r(x(t)))] = 0$, this satisfies the requirements of Definition 2.3. At first it may seem contradictory that we can prove convergence to a DSS when these are not guaranteed to exist. The conditions under which Corollary 3.7 holds true, however, are precisely those conditions under which a DSS does exist, that is, when $f_1(x, r(x))$ has a strict local minimum in $\mathcal{X}$.

Note that these results make no direct assumptions about the follower's payoff function or learning update. Indeed, if we relax the second part of Assumption 3.1 they could be satisfied for finite $\mathcal{Y}$ and discontinuous $r(x)$. We simply require that for every $x \in \mathcal{X}$ the follower's strategy will converge to some unique fixed point $r(x)$ at a sufficiently fast rate *relative* to the leader's commitment schedule. In the next section we will consider some specific scenarios in which this requirement is satisfied, and how we can select a suitable commitment schedule given some information about the follower.

3.2 Choosing the Commitment Schedule

In order to derive specific commitment schedules that provably satisfy Assumption 3.5, we need finite-time convergence rate guarantees for the follower's strategy. It is important to note that the Hi-C algorithm itself does not require the knowledge of the convergence rate. The leader can always choose the commitment schedule with respect to the worst known upper-bounds of first-order optimisation methods, assuming the slowest rate for the follower. However, when more is known about the follower's rate of convergence, we can use the rate to derive better commitment schedules. To illustrate how to derive k_n from known rates, we will consider one such well-studied case where the follower's objective function is strongly concave. Throughout this section we will make additional assumptions on the payoff functions f_2, and the best-response function r. These assumptions are needed only for the results within section 3.2.

Assumption 3.8. $\forall x \in \mathcal{X}$, $f_2(x, y)$ is K_2-smooth and μ-strongly concave w.r.t. y.

Under the assumption 3.8, deterministic gradient ascent on f_2 with a fixed step-size schedule $\beta_t = \beta$ is sufficient for the follower's strategy to converge to its best-response.

Proposition 3.9 (Nesterov [34, Chapter 2])**.** *Let the follower update its strategy using deterministic gradient ascent with a fixed step-size $\beta \in (0, \frac{1}{K_2}]$, such that*

$$y_{t+1} = y_t + \beta \nabla_y f_2(\tilde{x}_{n(t)}, y_t) \tag{13}$$

then for any stage $t \geq 0$, and any $k \in [1, k_{n(t)}]$, we have

$$\|y_{t+k} - r(\tilde{x}_{n(t)})\| \leq (1 - \beta\mu)^{\frac{k}{2}} \|y_t - r(\tilde{x}_{n(t)})\|. \tag{14}$$

Imagine we are given step-size and perturbation schedules $\{\alpha_n\}_{n\geq 0}$ and $\{\delta_n\}_{n\geq 0}$ satisfying Assumption 3.4. To find a suitable commitment schedule, we choose an arbitrary sequence $\{\xi_n\}_{n\geq 0}$ such that $\lim_{n\to\infty} \xi_n = 0$ and $\sup_n \xi_n < \infty$. We then need a commitment schedule $\{k_n\}_{n\geq 0}$ such that:

$$\frac{1}{\delta_n}\|\tilde{y}_n - r(\tilde{x}_n)\| \leq \xi_n \tag{15}$$

for all $n \geq 0$. To apply Proposition 3.9, we need to be able to bound $\|y_t - r(\tilde{x}_{n(t)})\|$ for all $t \geq 0$. Previously we simply required that the strategies be bounded almost surely (Assumption 3.2). Now will will assume that this bound is deterministic, and known in advance.

Assumption 3.10. There exists a deterministic constant $B < \infty$ such that $\sup_{t\geq 0} \|y_t\| < \frac{B}{2}$ and $\sup_{n\geq 0} \|r(\tilde{x}_n)\| < \frac{B}{2}$ almost surely.

This always holds if y_t is constrained to a bounded set $\mathcal{Y}$. We then have $\sup_{t\geq 0} \|y_t - r(\tilde{x}_n(t))\| \leq B$ almost surely. Then, under Assumptions 3.8 and 3.10, using the proposition 3.9, we have that

$$\frac{1}{\delta_n}\|\tilde{y}_n - r(\tilde{x}_n)\| \leq \frac{1}{\delta_n}(1-\beta\mu)^{\frac{k_n}{2}} B. \tag{16}$$

Upper-bounding this by ξ_n, we have

$$\frac{1}{\delta_n}(1-\beta\mu)^{\frac{k_n}{2}} B \leq \xi_n \tag{17}$$

$$2\frac{\ln \delta_n\xi_n - \ln B}{\ln(1-\beta\mu)} \leq k_n. \tag{18}$$

Setting $\xi_n = \frac{1}{n^p}$ for $p > 0$, we obtain the following convergence result:

Corollary 3.11. *For the leader's perturbation schedule $\{\delta_n\}_{n\geq 0}$, define the commitment times as*

$$k_n = \left\lceil 2\frac{\ln \delta_n - \ln B - p\ln n}{\ln(1-\beta\mu)} \right\rceil \tag{19}$$

for $p > 0$ and $\beta \in (0, \frac{1}{K_2}]$. Under Assumptions 3.1 through 3.4, 3.8 and 3.10, if the follower updates their strategy using deterministic gradient ascent with step-size β, then the leader strategies x_n computed by Hi-C converge to H almost surely as $n \to \infty$.

The specified k_n gives us $\frac{1}{\delta_n}\|\tilde{y}_n - r(\tilde{x}_n)\| \leq \frac{1}{n^p}$, and so $\{k_n\}_{n\geq 0}$ satisfies Assumption 3.5. The result then follows immediately from Theorem 3.6. For $\delta_n = \frac{\bar{\delta}}{n^q}$, this yields $k_n = O(\ln n)$.

The assumption that the follower's payoffs are strongly concave makes intuitive sense in this setting. The only information the leader can obtain about the follower's asymptotic best-responses is through their finite time adaptation to the leader's current strategy. If, over some subset $S \subset \mathcal{Y}$ containing $r(x)$, the curvature of f_2 is allowed to be arbitrarily small, then once the follower's strategy reaches S it may converge to $r(x)$ arbitrarily slowly. From the leader's perspective, it would appear that the follower has nearly settled on a best-response, such that the leader may over- or underestimate the value of their current strategy. It is therefore reasonable to require that the leader have some information about how fast the follower's strategy should converge. In Corollary 3.11, the necessary commitment schedule depends on $\beta\mu$, which determines the follower's convergence rate.

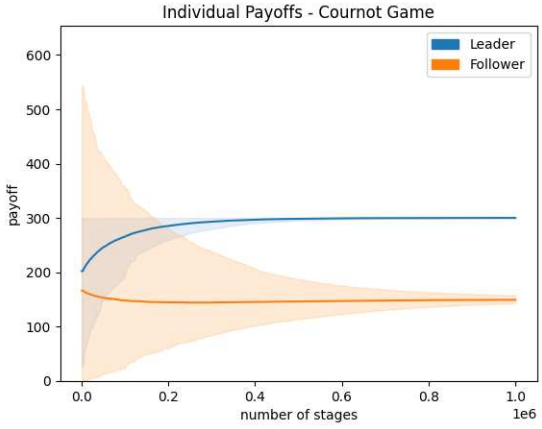

Figure 1: Hi-C paired with gradient ascent in the Cournot duopoly. Averaged over 32 runs (shaded regions show ranges).

3.3 Numerical Experiments

We demonstrate the convergence behavior of Hi-C in a simple differentiable game corresponding to the Cournot duopoly model [43] with linear prices and costs. Here $x, y \in \Re$ are the quantities of some good produced by each player, with payoffs defined as

$$f_1(x,y) = x[50 - (x+y)] - x \tag{20}$$

$$f_2(x,y) = y[50 - (x+y)] - y \tag{21}$$

The payoff each player receives depends on the price per unit, which is a strictly decreasing function of the total quantity produced. Note that the unique Stackelberg equilibrium of this game (as well as the unique DSE) is $x = 24.5$, $y = 12.25$, such that $f_1(x,y) \approx 300$ and $f_2(x,y) \approx 150$. Figure 1 shows that Hi-C converges to this solution when paired with a gradient ascent learner in the Cournot game. Initially the gradient ascent learner (the follower) maximizes its payoff under the assumption that the leader's strategy is fixed. Over time Hi-C (the leader) increases its production quantity, knowing that the follower will reduce its own production to maintain its profits. In these experiments Hi-C uses a learning rate schedule of $\alpha_n = .001n^{-.1}$ and a perturbation schedule of $\delta_n = n^{-.6}$, with the corresponding commitment schedule computed by Equation 19. The gradient ascent learner uses a fixed learning rate of 0.1.[1]

4 ROLE NEGOTIATION

So far we have assumed that the leader and follower roles are assigned by some external process. In this section we will briefly explore ways in which uncoupled learners might "negotiate" who will lead and who will follow during the training process itself. In principle, two independent agents could use a variety of protocols (e.g., a coin toss) to agree upon their respective roles before training begins. Here, however, we consider whether roles can themselves be *learned* in a way that maximizes each player's individual payoffs. In some settings learning roles as part of the training process may be necessary or advantageous. If two independent agents have never interacted with one another before, they are unlikely to

[1]Code available at: https://github.com/rtloftin/Hi-C/tree/aamas2024

Algorithm 2 The meta-learning update for role negotiation. Maintains a "meta-strategy" parameterized by w_n, and samples its role from its current strategy at each meta-learning interval.

Inputs: Step-sizes $\{\beta_n\}_{n\geq 0}$, perturbation schedule $\{\kappa_n\}_{n\geq 0}$, commitment schedule $\{\tau_n\}_{n\geq 0}$.
Initialize: $w_0 \leftarrow 0$
for step $n = 0, 1, \ldots$ **do**
  sample Δ uniformly from $\{-1, 1\}$.
  $\tilde{w}_n \leftarrow w_n + \kappa_n \Delta_n$.
  sample z_n uniformly from $[0, 1]$.
  if $z_n < \sin^2(\tilde{w}_n)$ **then**
    follow the Hi-C update (Algorithm 1) on f_1 for τ_n stages
  else
    follow the gradient ascent update on f_1 for τ_n stages
  end if
  collect observed joint strategies $\{\bar{x}_i^n\}$ and $\{\bar{y}_i^n\}$
  $a_n \leftarrow \frac{1}{\tau_n} \sum_{i=1}^{\tau_n} f_1(x_i^n, y_i^n)$
  $w_{n+1} = w_n + \frac{\beta_n a_n}{\kappa_n \Delta_n}$
end for

share a convention for negotiating roles. In a fully cooperative task, hierarchical learning will be unnecessary, and it would be desirable if both players learned to use the follower's faster gradient update.

Our goal then is to allow each player to learn which role (leader or follower) will yield the highest payoffs given its partner's behavior. A straightforward approach is to wrap the hierarchical learning process in a "meta-learning" process, where a pair of independent meta-learners each commits to a particular role for some pre-determined number of steps, and then evaluates their average payoff under that role given the role their partner chose. Importantly, the meta-learning process is *symmetric*, with no leader-follower hierarchy.

Algorithm 2 describes the meta-learning update for player 1 (the update for player 2 only differs in the use of the payoff function f_2). The length of the meta-learning intervals are described by a fixed schedule $\{\tau_n\}_{n\geq 0}$, which, as with the Hi-C update, we assume will grow arbitrarily large over time. This commitment schedule, along with the step-size schedule $\{\beta_n\}_{n\geq 0}$, is assumed to be shared between both meta-learners. Each meta-learner maintains a "meta-strategy" parameterized by a single scalar value $w_n \in \mathfrak{R}$. We let $p(w_n) = \sin^2(w_n)$ be the probability of choosing to lead at interval n, with $1 - p(w_n) = \cos^2(w_n)$ being the probability of choosing to follow. This parameterization allows w to be unbounded, and allows us to represent *pure* strategies using finite values of w. At each interval n the meta-learner for player 1 approximates gradient ascent on its expected payoff

$$f_n(w^1) = [p(w^1), 1 - p(w^1)]\hat{\mathbf{U}}_n^1[p(w_n^2), 1 - p(w_n^2)]^\top \tag{22}$$

where w_n^2 denotes player 2's meta-strategy, while the matrix $\hat{\mathbf{U}}_n^1$ denotes the expected average payoff for player 1 under each of the four possible joint role assignments. Note that the expectation $\hat{\mathbf{U}}_n^1$ is conditioned on the underlying joint strategy (x, y) at the start of interval n. As with the Hi-C update, the meta-learner updates w using the single-sample variant of SPSA. Note that other approaches to estimating the gradient $\nabla_{w^1} f_n(w^1)$, for example, using the log-likelihood trick, would suffer from singularities when evaluated at pure strategies (e.g., $w^1 = 0$ or $w^1 = \frac{\pi}{2}$).

The fact that $\hat{\mathbf{U}}_n^1$ may depend on the underlying joint strategy means that, in general, we cannot apply standard stochastic approximation results to the meta-learning process alone. If we can assume, however, that the long-term behavior of underlying learning process is asymptotically independent of its initial state (for any possible leader-follower role assignment) then we can describe the joint meta-learning process by the stochastic approximation

$$w_{n+1}^1 = w_n^1 + \beta_n \left[[\nabla p(w_n^1), -\nabla p(w_n^1)]\mathbf{U}^1[p(w_n^2), 1 - p(w_n^2)]^\top + \zeta_n^1 + \epsilon_n^1 \right] \tag{23}$$

$$w_{n+1}^2 = w_n^1 + \beta_n \left[[p(w_n^1), 1 - p(w_n^1)]\mathbf{U}^2[\nabla p(w_n^2), -\nabla p(w_n^2)]^\top + \zeta_n^2 + \epsilon_n^2 \right] \tag{24}$$

where $U^1 = \mathrm{E}[\lim_{n\to\infty} \hat{U}_n^1]$ and $U^2 = \mathrm{E}[\lim_{n\to\infty} \hat{U}_n^2]$, ζ_n^1 and ζ_n^2 are the noise terms introduced by SPSA, and ϵ_n^1 and ϵ_n^2 are $o(1)$. This in turn converges (under the standard SA assumptions) to an ICT invariant set of the limiting ODE:

$$\dot{w}^1 = [\nabla p(w^1), -\nabla p(w^1)]\mathbf{U}^1[p(w^2), 1 - p(w^2)]^\top \tag{25}$$

$$\dot{w}^2 = [p(w^1), 1 - p(w^1)]\mathbf{U}^2[\nabla p(w^2), -\nabla p(w^2)]^\top \tag{26}$$

Whether such an invariant set necessarily corresponds to a specific leader–follower ordering will depend on the structure of the game. In general, such a set may not correspond to an equilibrium point of the ODE, with the players never converging to fixed roles. We leave the characterization of games in which role negotiation can be guaranteed to converge as an open question for future work.

5 DISCUSSION

A major motivation for our work is to understand the problem of ad hoc collaboration between autonomous agents, both AI and human. In this case, agents cannot assume anything about how others' behavior will change over time, and need to adapt to one another simultaneously. Previous analysis of *naive* simultaneous learning updates such as SGA has suggested that such learning processes may be highly unstable, and may fail to converge to good joint strategies. Research in differentiable games has in recent years focused on the types of centralized training settings commonly arising in deep learning, where some learners must have detailed knowledge of other's loss functions and learning updates. Thus, these methods and their analyses are not directly applicable to ad hoc collaboration. Hierarchical learning dynamics are well-suited to this setting, but have previously required that the leader have direct access to the follower's payoff function. Our work overcomes this critical limitation. Our approach also has the potential to be useful in centralized training. Compared to the coupled hierarchical gradient update, Hi-C will generally have much lower per-step computational cost, though whether this offsets the potential increase in sample complexity in practice is an open question.

Future work. Immediate future directions for this work include expanding the class of follower learning updates and payoff functions for which we can provide concrete convergence guarantees.

This could include more flexible methods such as stochastic gradient descent, or no-regret learning rules such as online mirror descent. Recent work on bi-level optimization such as Liu et al. [25] has also provided theoretical tools for analysing convergence in the case of non-concave follower objectives. The extension of these results to our uncoupled setting is another important question for future work. Finally, there are a number of open questions regarding the dynamics of role negotiation. These include determining in which classes of games the players will converge to fixed roles with high probability, and whether the players' average payoffs can be guaranteed to converge even when the roles themselves do not.

6 RELATED WORK

Differentiable Games. Previous work on gradient ascent in differentiable games has found that simultaneous gradient ascent on individual payoff functions can fail to converge [30, 31]. This has motivated the development of alternative solution concepts that are better suited to differentiable games, such as chain recurrent sets [36] and local Stackelberg equilibria [22]. Others have proposed modified gradient ascent approaches to achieve at least local convergence to fixed-points in certain classes of games [2, 31, 42]. Similar to our approach are methods for two-player games that update the individual strategies on two different timescales [28, 32, 35]. As with our approach, Nouiehed et al. [35] implement timescale separation by having the follower execute multiple gradient steps for every leader update, though unlike our work, their leader does not directly attempt to *shape* the behavior of the follower.

Hierarchical Model of Play and Role Assignment. Assuming the follower plays an immediate best-response, previous work has also provided lower bounds on the sample complexity of identifying Stackelberg equilibria in Stackelberg security games [38], bandit games [1] and Markov games [40]. The challenge in our setting is that we must assume the follower is implementing an incremental learning update, which may only play a true best-response asymptotically. Most closely related to our work is the two-timescale hierarchical gradient update [14, 49]. Unlike our method, the hierarchical gradient update requires that the leader have access to the follower's payoffs. The earliest analysis of leader–follower role assignments was in Basar [4], while Basar and Haurie [6] considered the case where the roles switch between players depending on an exogenous process. More recent work has considered the case where players change roles depending on the game state, where the roles are still pre-assigned for each state [5]. In the context of strategic classification, Zrnic et al. [50] have analysed the setting where a specific player can choose and dictate a role assignment for everyone. To our knowledge, ours is the first result on online negotiation of the roles during hierarchical play.

Multiagent Learning. Our work is also related to *opponent shaping* approaches [16, 48], where one or both learners explicitly account for their partner's learning behavior, and update their strategy accordingly. Of these the model-free opponent shaping (M-FOS) framework of Lu et al. [26] is closest to ours. The key differences from our method are that M-FOS assumes the follower can be "reset" after each interval, and only allows the follower to adapt for a fixed number of stages. In contrast, we do not require such resets, and explicitly account for the fact that the follower's strategy depends on the entire history of interaction. Hi-C also allows the follower to learn over increasing time horizons, enabling asymptotic convergence. Finally, Hi-C is conceptually similar to no-regret learning methods for non-stationary tasks [12] and adaptive partners [39], in which the leader commits to candidate "expert" strategies for increasingly long time intervals.

Bi-level Optimization. The problem of finding differential Stackelberg equilibria can be cast as bi-level optimization. Indeed, the hierarchical gradient update ([14, 49]) corresponds to an approximate implicit differentiation (AID) method for bi-level problems. Iterative differentiation (ITD) methods (e.g. [17, 19, 21, 37]) are conceptually similar to our approach as well. However, both AID and ITD methods require analytically differentiating through the follower's best-response function, which in turn requires the gradients (and Hessians) of the follower's payoff function. Recent work [24] does not use Hessians, but still requires knowledge of the follower's objective functions. Developed for centralized training settings such as GANs, these methods cannot be applied to settings where the learners are truly autonomous and decentralised. While some recent work ([10, 27]) has presented zeroth-order (gradient-free) methods for bi-level optimization, these simulate multiple independent copies of the follower, and so require access to the follower's payoffs and learning update.

7 CONCLUSION

We have presented, to the best of our knowledge, the first *uncoupled* learning update that can be shown to converge to differential Stackelberg solutions for a broad class of general-sum differentiable games. The Hi-C learning update for the leader agent can be implemented without access to the follower's payoff function or the details of their learning update. This also means that Hi-C does not need to estimate the gradients or Hessians of the follower's payoffs. Most importantly, our convergence results provide theoretical insights into uncoupled hierarchical learning processes, where one agent must learn about the preferences of another agent through its observable behavior alone. We have also presented the first online role negotiation dynamics, which illustrate how agents can strategically negotiate a leader–follower ordering as part of the hierarchical learning process.

ACKNOWLEDGMENTS

This work was supported by the Academy of Finland (Flag-ship programme: Finnish Center for Artificial Intelligence, FCAI; grants 319264, 313195, 305780, 292334, 328400, 28400), the UKRI Turing AI World-Leading Researcher Fellowship EP/W002973/1, the Finnish Science Foundation for Technology and Economics (KAUTE), and the Hybrid Intelligence Center, https://hybrid-intelligence-centre.nl, grant number 024.004.022.

REFERENCES

[1] Yu Bai, Chi Jin, Huan Wang, and Caiming Xiong. 2021. Sample-efficient learning of Stackelberg equilibria in general-sum games. *Advances in Neural Information Processing Systems* 34 (2021), 25799–25811.

[2] David Balduzzi, Sebastien Racaniere, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. 2018. The mechanics of n-player differentiable games. In *International Conference on Machine Learning*. PMLR, 354–363.

[3] Wolfram Barfuss. 2022. Dynamical systems as a level of cognitive analysis of multi-agent learning: Algorithmic foundations of temporal-difference learning dynamics. *Neural Computing and Applications* 34, 3 (2022), 1653–1671.

[4] Tamer Basar. 1973. On the relative leadership property of Stackelberg strategies. *Journal of Optimization Theory and Applications* 11, 6 (1973), 655–661.

[5] Tamer Başar, Alain Bensoussan, and Suresh P Sethi. 2010. Differential games with mixed leadership: The open-loop solution. *Appl. Math. Comput.* 217, 3 (2010), 972–979.

[6] Tamer Basar and Alain Haurie. 1982. *Feedback equilibria in differential games with structural and modal uncertainties*. École des hautes études commerciales.

[7] Tamer Başar and Geert Jan Olsder. 1998. *Dynamic noncooperative game theory*. SIAM.

[8] S. Bhatnagar, H.L. Prasad, and L.A. Prashanth. 2012. *Stochastic Recursive Algorithms for Optimization: Simultaneous Perturbation Methods*. Springer London.

[9] Vivek S Borkar. 2009. *Stochastic approximation: a dynamical systems viewpoint*. Vol. 48. Springer.

[10] Lesi Chen, Jing Xu, and Jingzhao Zhang. 2023. On Bilevel Optimization without Lower-level Strong Convexity. *arXiv preprint arXiv:2301.00712* (2023).

[11] Aleksander Czechowski and Georgios Piliouras. 2023. Non-Chaotic Limit Sets in Multi-Agent Learning. *Autonomous Agents and Multi-Agent Systems* 37, 2 (jul 2023), 24. https://doi.org/10.1007/s10458-023-09612-x

[12] Daniela Pucci de Farias and Nimrod Megiddo. 2003. How to combine expert (or novice) advice when actions impact the environment. *Advances in Neural Information Processing Systems* 17 (2003).

[13] Ishan Durugkar, Elad Liebman, and Peter Stone. 2021. Balancing individual preferences and shared objectives in multiagent reinforcement learning. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*. 2505–2511.

[14] Tanner Fiez, Benjamin Chasnov, and Lillian Ratliff. 2020. Implicit learning dynamics in Stackelberg games: Equilibria characterization, convergence analysis, and empirical study. In *International Conference on Machine Learning*. PMLR, 3133–3144.

[15] Tanner Fiez, Benjamin Chasnov, and Lillian J Ratliff. 2019. Convergence of learning dynamics in Stackelberg games. *arXiv preprint arXiv:1906.01217* (2019).

[16] Jakob Foerster, Richard Y. Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. 2018. Learning with Opponent-Learning Awareness. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. 122–130.

[17] Luca Franceschi, Paolo Frasconi, Saverio Salzo, Riccardo Grazzi, and Massimiliano Pontil. 2018. Bilevel programming for hyperparameter optimization and meta-learning. In *International Conference on Machine Learning*. PMLR, 1568–1577.

[18] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems 27*. 2672–2680.

[19] Riccardo Grazzi, Luca Franceschi, Massimiliano Pontil, and Saverio Salzo. 2020. On the iteration complexity of hypergradient computation. In *International Conference on Machine Learning*. PMLR, 3748–3758.

[20] Daniel Hennes, Dustin Morrill, Shayegan Omidshafiei, Rémi Munos, Julien Perolat, Marc Lanctot, Audrunas Gruslys, Jean-Baptiste Lespiau, Paavo Parmas, Edgar Duèñez Guzmán, and Karl Tuyls. 2020. Neural Replicator Dynamics: Multiagent Learning via Hedging Policy Gradients. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems* (Auckland, New Zealand) *(AAMAS '20)*. Richland, SC, 492–501.

[21] Kaiyi Ji, Junjie Yang, and Yingbin Liang. 2021. Bilevel optimization: Convergence analysis and enhanced design. In *International conference on machine learning*. PMLR, 4882–4892.

[22] Chi Jin, Praneeth Netrapalli, and Michael Jordan. 2020. What is local optimality in nonconvex-nonconcave minimax optimization?. In *International conference on machine learning*. PMLR, 4880–4889.

[23] Alistair Letcher, David Balduzzi, Sébastien Racanière, James Martens, Jakob Foerster, Karl Tuyls, and Thore Graepel. 2019. Differentiable Game Mechanics. *Journal of Machine Learning Research* 20, 84 (2019), 1–40.

[24] Bo Liu, Mao Ye, Stephen Wright, Peter Stone, and Qiang Liu. 2022. Bome! bilevel optimization made easy: A simple first-order approach. *Advances in Neural Information Processing Systems* 35 (2022), 17248–17262.

[25] Risheng Liu, Yaohua Liu, Shangzhi Zeng, and Jin Zhang. 2021. Towards gradient-based bilevel optimization with non-convex followers and beyond. *Advances in Neural Information Processing Systems* 34 (2021), 8662–8675.

[26] Christopher Lu, Timon Willi, Christian A Schroeder De Witt, and Jakob Foerster. 2022. Model-free opponent shaping. In *International Conference on Machine Learning*. PMLR, 14398–14411.

[27] Chinmay Maheshwari, S Shankar Sasty, Lillian Ratliff, and Eric Mazumdar. 2023. Convergent First-Order Methods for Bi-level Optimization and Stackelberg Games. *arXiv preprint arXiv:2302.01421* (2023).

[28] Eric V Mazumdar, Michael I Jordan, and S Shankar Sastry. 2019. On finding local Nash equilibria (and only local Nash equilibria) in zero-sum games. *arXiv preprint arXiv:1901.00838* (2019).

[29] Kevin R McKee, Ian Gemp, Brian McWilliams, Edgar A Duèñez-Guzmán, Edward Hughes, and Joel Z Leibo. 2020. Social Diversity and Social Preferences in Mixed-Motive Reinforcement Learning. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*. 869–877.

[30] Panayotis Mertikopoulos, Christos Papadimitriou, and Georgios Piliouras. 2018. Cycles in adversarial regularized learning. In *Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2703–2717.

[31] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. 2017. The numerics of GANs. *Advances in neural information processing systems* 30 (2017).

[32] Luke Metz, Ben Poole, David Pfau, and Jascha Sohl-Dickstein. 2017. Unrolled Generative Adversarial Networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

[33] Reuth Mirsky, Ignacio Carlucho, Arrasy Rahman, Elliot Fosong, William Macke, Mohan Sridharan, Peter Stone, and Stefano V Albrecht. 2022. A Survey of Ad Hoc Teamwork Research. In *Multi-Agent Systems: 19th European Conference, EUMAS 2022, Düsseldorf, Germany, September 14–16, 2022, Proceedings*. Springer, 275–293.

[34] Yurii Nesterov. 2018. *Lectures on convex optimization*. Springer.

[35] Maher Nouiehed, Maziar Sanjabi, Tianjian Huang, Jason D Lee, and Meisam Razaviyayn. 2019. Solving a class of non-convex min-max games using iterative first order methods. *Advances in Neural Information Processing Systems* 32 (2019).

[36] Christos H. Papadimitriou and Georgios Piliouras. 2018. From Nash Equilibria to Chain Recurrent Sets: An Algorithmic Solution Concept for Game Theory. *Entropy* 20, 10 (2018), 782. https://doi.org/10.3390/e20100782

[37] Fabian Pedregosa. 2016. Hyperparameter optimization with approximate gradient. In *International conference on machine learning*. PMLR, 737–746.

[38] Binghui Peng, Weiran Shen, Pingzhong Tang, and Song Zuo. 2019. Learning optimal strategies to commit to. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 33. 2149–2156.

[39] Jan Poland and Marcus Hutter. 2005. Defensive universal learning with experts. In *International Conference on Algorithmic Learning Theory*. Springer, 356–370.

[40] Giorgia Ramponi and Marcello Restelli. 2022. Learning in Markov Games: can we exploit a general-sum opponent?. In *The 38th Conference on Uncertainty in Artificial Intelligence*.

[41] Lillian J. Ratliff, Samuel A. Burden, and S. Shankar Sastry. 2016. On the Characterization of Local Nash Equilibria in Continuous Games. *IEEE Trans. Autom. Control.* 61, 8 (2016), 2301–2307. https://doi.org/10.1109/TAC.2016.2583518

[42] Florian Schäfer and Anima Anandkumar. 2019. Competitive gradient descent. *Advances in Neural Information Processing Systems* 32 (2019).

[43] Carl Shapiro. 1989. Theories of oligopoly behavior. *Handbook of industrial organization* 1 (1989), 329–414.

[44] Marwaan Simaan and Jose B Cruz. 1973. On the Stackelberg strategy in nonzero-sum games. *Journal of Optimization Theory and Applications* 11, 5 (1973), 533–555.

[45] James C Spall. 1992. Multivariate stochastic approximation using a simultaneous perturbation gradient approximation. *IEEE transactions on automatic control* 37, 3 (1992), 332–341.

[46] James C Spall. 1997. A one-measurement form of simultaneous perturbation stochastic approximation. *Automatica* 33, 1 (1997), 109–112.

[47] Peter Stone, Gal Kaminka, Sarit Kraus, and Jeffrey Rosenschein. 2010. Ad hoc autonomous agent teams: Collaboration without pre-coordination. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 24. 1504–1509.

[48] Timon Willi, Alistair Hp Letcher, Johannes Treutlein, and Jakob Foerster. 2022. COLA: consistent learning with opponent-learning awareness. In *International Conference on Machine Learning*. PMLR, 23804–23831.

[49] Liyuan Zheng, Tanner Fiez, Zane Alumbaugh, Benjamin Chasnov, and Lillian J Ratliff. 2022. Stackelberg actor-critic: Game-theoretic reinforcement learning algorithms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 9217–9224.

[50] Tijana Zrnic, Eric Mazumdar, Shankar Sastry, and Michael Jordan. 2021. Who Leads and Who Follows in Strategic Classification?. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 15257–15269.