Opponent Anticipation via Conjectural Variations

Benjamin Chasnov, Tanner Fiez, Lillian Ratliff
{bchasnov,fiezt, ratliffl}@uw.edu
Electrical & Computer Engineering
University of Washington

Abstract

We introduce a framework for multi-agent learning in which agents anticipate each others' reactions by forming *conjectures* about their learning processes, and devise learning rules using a variational perspective relative to these conjectures. The conjecture learning schemes lead to an alternative equilibrium concept, a *differential general conjectural variations equilibrium*. When compared to simultaneous gradient play, we empirically observe that implicit conjecture learning leads to a more equitable solution in a zero-sum polynomial game, while gradient and fast conjecture learning decrease the rotational components of the joint dynamics. The framework provides techniques for future synthesis of novel heterogeneous multi-agent learning rules.

1 Introduction

Learning in games is the study of the learning processes that agents undergo while interacting with others [8]. Currently lacking in this field are unified methods to synthesize learning rules for heterogeneous agents that are cognizant of others' learning dynamics. In this paper, we incorporate models of other agents' learning by adopting a dynamic perspective of decision-making. Agents form *conjectures* about each other's learning processes, giving rise to a new class of strategic multi-agent learning schemes. The conjecture framework provides a principled method to perform gradient-based learning in multiagent settings. Research on conjectural variations from a game-theoretic perspective has been on-going for decades [3, 5, 7, 10]. Yet, motivated by recent advancements on improving the convergence properties of large-scale machine learning tasks [1, 6, 9, 16, 11], we seek a unifying perspective that incorporates strategic information of opponents and speeds up convergence to desired equilibria. We begin by presenting the conjecture framework where the actions of opponents are a function of one's own actions. The formulation leads to the solution concept of a *differential general conjectural variations equilibrium*. Employing other classes of conjectures will lead to rich and novel behaviors for non-cooperative agents in continuous games.

2 Multi-agent Learning with Conjectures

Towards introducing the conjectures framework, we define the game-theoretic abstraction which describes players objectives. A continuous n-player general-sum game is a collection of costs $(f_1, \dots, f_n)$ defined on $X = X_1 \times \cdots \times X_n$ where $f_i \in C^r(X, \mathbb{R})$ with $r \geq 2$ is player i's cost function, $X_i = \mathbb{R}^{d_i}$ is their action space, and $d = \sum_{i=1}^n d_i$. Each player $i \in \mathcal{I} = \{1, \dots, n\}$ aims to select an action $x_i \in X_i$ that minimizes their cost $f_i(x_i, x_{-i})$ given the actions of all other players, namely $x_{-i} \in X_{-i}$. The form of learning rule studied in this paper is given by

$$x_{k+1,i} = x_{k,i} - \gamma_{k,i} g_i(x_{k,i}, \xi_i(x_{k,i}, x_{k,-i})),\ i \in \mathcal{I}$$

where g_i is derived from gradient information of the player's cost function and ξ_i is player i's conjecture about the rest of the players given by $\xi_i(x_k) = (\xi_i^j(x_k))_{j \neq i}$ with $\xi_i^j(x_k)$ denoting player i's

Abstract for: Smooth Games Optimization and Machine Learning Workshop (NeurIPS 2019), Vancouver, Canada.

conjecture about player j. In particular, $g_i \equiv D_{x_i} f_i(\cdot, \xi_i(\cdot, x_{-i}))$ (or $g_i \equiv \widehat{D_{x_i} f_i}(\cdot, \xi_i(\cdot, x_{-i}))$ with $\mathbb{E}[g_i] = D_{x_i} f_i(\cdot, \xi_i(\cdot, x_{-i}))$ in the stochastic case). To provide some intuition, consider simultaneous gradient play which is defined by taking $\xi_i(x_k) = x_k$—that is, player i conjectures that players $-i$ select a *static best response policy* in the sense that they conjecture opponents will play the same strategy as in the previous round. More generally, consider a two player game (f_1, f_2), and for simplicity the deterministic case in which players have oracle access to their updates g_i. Then, player 1's update is $x_{k+1,1} = x_{k,1} - \gamma_{k,1}(D_1 f_1(x_k) + D_1 \xi_1(x_k)^\top D_2 f_1(x_k))$, and similarly for player 2; that is, g_i is the derivative of player i's cost, conditioned on the conjecture ξ_i, with respect to x_i.

The equilibrium notion we consider is known as a *general conjectural variations equilibrium* [5]. Unlike a Nash equilibrium, which is defined with respect to the cost evaluation of each player at a candidate point, GCVE is defined with respect to first order conditions on the cost function of each player. Consequently, players may have an incentive to deviate. Towards mitigating such limitations, we define the notion of a differential GCVE which includes a second order condition so that locally no player has a direction in which they can adjust their action and benefit.

Definition 1 (General Conjectural Variations Equilibrium (GCVE)). *For a game $\mathcal{G} = (f_1, \ldots, f_n)$ with each $f_i \in C^r(X, \mathbb{R})$ and conjectures $\xi_i : X \to X_{-i}$, a point $x^* \in X$ constitutes a* general conjectural variations equilibrium *if $D_i f_i(x^*) + D_i \xi_i(x^*)^\top D_{-i} f_i(x^*) = 0$, $i \in \mathcal{I}$. Further, a GCVE $x^* \in X$ is a differential GCVE if, for each $i \in \mathcal{I}$, $D_i^2 f_i(x_i, \xi_i(x^*)) > 0$.*

To reduce notational overhead, in the remainder we consider two-player deterministic settings with constant learning rates; the extension to n–player games is largely straightforward.

Implicit Conjectures. While there are also numerous classes of conjectures ξ_i, a natural conjecture is that an opponent is playing a best response at each iteration; that is, player 1 assumes that $x_{k,2}$ satisfies the sufficient condition that $D_2 f_2(x_{k,1}, x_{k,2}) = 0$. Under sufficient regularity conditions on cost functions, the implicit function theorem gives rise to $D_1 \xi_1(x_k) = -(D_2^2 f_2(x_k))^{-1} D_{21} f_2(x_k)$, and similarly for player 2. Together, this leads to the so-called *implicit conjectures* update which can be written in vector form as $x_{k+1} = x_k - \Gamma(g(x_k) - \mathrm{diag}(J_{\mathrm{o}}^\top(x_k) J_{\mathrm{d}}^{-1}(x_k) \nabla f(x_k)))$ where $J_{\mathrm{d}}(x)$ is the block diagonal matrix $J(x) - J_{\mathrm{o}}(x)$.

Gradient Conjectures. Motivated by applications in machine learning, we can approximate these dynamics by replacing J_{d}^{-1} with a conjectured learning rate matrix $\widetilde{\Gamma}$, thereby leading to *gradient-based conjectures*, where player i assumes their opponent player j is simply playing simultaneous gradient play with response map $x_{k+1,j} = x_{k,j} - \widetilde{\gamma}_i D_j f_j(x_k)$ where $\widetilde{\gamma}_i$ is player i's conjectured learning rate for player j. In a two-player game with constant conjectured learning rates $\widetilde{\gamma}_1$ and $\widetilde{\gamma}_2$, the conjectures are defined as $\xi_1(x_1, x_2) = x_2 - \widetilde{\gamma}_1 D_2 f_2(x_1, x_2)$ and $\xi_2(x_1, x_2) = x_1 - \widetilde{\gamma}_2 D_1 f_1(x_1, x_2)$. Since $D\xi_i(x) = -\widetilde{\gamma}_i D_{ji} f_j(x)$, the update rules are given by $x_{k+1} = x_k - \Gamma g(x) - \mathrm{diag}(\widetilde{\Gamma} J_{\mathrm{o}}^\top(x) \nabla f(x))$ with $g(x) = (D_1 f_1(x), D_2 f_2(x))$, $\widetilde{\Gamma}$ a diagonal matrix of the conjectured learning rates, and $J_{\mathrm{o}}(x)$ defined to be the block off-diagonal components of $J(x)$, the Jacobian of $g(x)$. For zero-sum games, the update reduces to $(I + \widetilde{\Gamma} J_{\mathrm{o}}^\top) g(x)$, since the vector $(D_1 f_2(x), D_2 f_1(x)) = -g(x)$. The inclusion of the term $D_j f_i$ where $i \neq j$ in the learning rule is the key insight of the gradient-based conjecture update. It allows agent i to anticipate how the other agent j's actions affect their own cost. The map $D_{ij} f_j$ transforms this strategic information into the agents' own coordinates.

Fast Conjectures. Several existing multi-agent learning algorithms can be derived from the conjectures framework. Towards this end, we consider Taylor expansions of a cost function for a player in order to explain how players envision the reactions of opponents impacts their own objectives. In a two-player game (f_1, f_2) with conjectures (ξ_1, ξ_2) of the form $\xi_i(x_i, x_j) = x_j + v$ for some vector v and each player $i = 1, 2$ such that $i \neq j$. Suppose that at iteration k, player 1 wants to compute a best response by selecting a minimizer in the set $\arg\min_{x_1} f_1(x_1, x_2 + v_k)$, given that player 1 conjectures that player 2 is using the simple update rule $x_2 + v_k$ and an oracle provides the update direction v_k. Given $x_{k,2}$, the cost function f_1 can be approximated locally by its Taylor expansion $f_1(x_1, x_{k,2} + v_k) = f_1(x_1, x_{k,2}) + D_2 f_1(x_1, x_{k,2}) v_k + O(v_k^2)$. Dropping higher order terms, we define the left-hand side as $f'_{k,1}(x_1)$. Now, player 1 faces the optimization problem $\arg\min_{x_1} f'_{k,1}(x_1) = \arg\min_{x_1} \{f_1(x_1, x_{k,2}) + D_2 f_1(x_1, x_{k,2}) v_k\}$. Suppose that player 1 optimizes the Taylor expansion of $f'_{k,1}(x_1)$, $f'_{k,1}(x_1 + w_k) \simeq f_1(x_1, x_{k,2}) + D_2 f_1(x_1, x_{k,2}) v_k + (D_1 f_1(x_1, x_{2,k}) + D_{12} f_1(x_1, x_{2,k}) v_k) w_k + O(w_k^2)$, again after dropping higher order terms. Performing a similar analysis for player 2, this leads to the gradient-based learning rule is of the form

Simultaneous gradient descent	$x_{k+1} = x_k - \Gamma_k g(x_k)$
Conjecture learning	$x_{k+1,i} = x_{k,i} - \gamma_{k,i}(D_i f_i(x_k) + D_i \xi_i^\top(x_k) D_j f_i(x_k)) \quad \forall i$
Implicit conjectures	$x_{k+1} = x_k - \Gamma_k(g(x_k) - \mathrm{diag}(J_{\mathrm{o}}^\top(x_k) J_{\mathrm{d}}^{-1}(x_k) \nabla f(x_k)))$
Gradient conjectures	$x_{k+1} = x_k - \Gamma_k(g(x_k) - \mathrm{diag}(\widetilde{\Gamma}_k J_{\mathrm{o}}^\top(x_k) \nabla f(x_k)))$
Fast conjectures	$x_{k+1} = x_k - \Gamma_k g(x_k) - \Gamma_k J_{\mathrm{o}}(x_k) v_k$, v_k is oracle direction
Lookahead [16]	$x_{k+1} = x_k - \gamma(I - \alpha J_o(x_k)) g(x_k)$
Symplectic gradient	$x_{k+1} = x_k - \gamma(I - \eta(x_k) A^\top(x_k)) g(x_k)$,
adjustment [1]	where $\equiv \frac{1}{2}(J - J^\top)$ and $\eta(x_k) \in \{-1, 1\}$
Learning w/ opponent	$x_{k+1,1} = x_{k,1} - \eta(D_1 f_1(x_k) - \delta(D_{21} f_2(x_k))^\top D_2 f_1(x_k))$
learning awareness [6]	$x_{k+1,2} = x_{k,2} - \eta D_2 f_2(x_k)$
Stable opponent	$x_{k+1} = x_k - \gamma(I - \alpha J_o(x_k)) g(x_k) - p(x_k) \alpha \mathrm{diag}(J_{\mathrm{o}}^\top(x_k) \nabla f(x_k))$,
shaping [9]	where $0 < p(x) < 1$ and $p(x) \to 0$ as $\|g(x)\| \to 0$
Consensus optimization [12]	$x_{k+1} = x_k - \gamma(I - \alpha J^\top(x_k)) g(x_k)$
Competitive gradient	$x_{k+1} = x_k - \Gamma(x_k)(I - \alpha J_o(x_k)) g(x_k)$
descent [14]	where $\Gamma(x_k) = (I - \alpha^2 J_o(x_k) J_o(x_k))^{-1}$

Table 1: The core of several existing update rules can be derived from the conjecture learning framework by choosing $v_{k,i}$ or $\tilde{\gamma}_{k,i}$ either in a principled or heuristic manner; e.g., Lookahead is fast conjecture learning with $v_{k,i} = \tilde{\gamma}_{k,i} g_i(x_k) \equiv \alpha g_i(x_k)$, learning with opponent-learning awareness is such that one player has a gradient based conjecture and the other a static best response conjecture, and stable opponent shaping is a combination of gradient and fast conjecture with heuristics for improving convergence properties.

$x_{k+1} = x_k - \Gamma(g(x_k) + J_{\mathrm{o}}(x_k) v_k)$, where $v_k = (v_{k,i})_i$ with $v_{k,i}$ player i's conjectured direction for player j, provided by an oracle. We refer to these dynamics as *fast conjecture learning*. The update form gives rise to a vast number of learning schemes depending on how $v_{k,i}$ and γ_i are defined. Essentially, the update rules above give players the ability to adjust for the effect an opponent has on their descent direction. Correcting for such impacts can lead to much faster convergence; however, the set of attractors is not equivalent to the set of simultaneous gradient descent attractors.

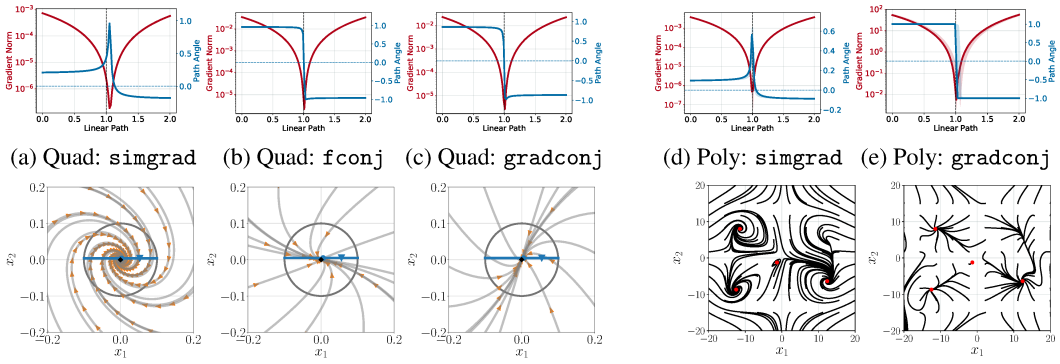
Outside the scope of this short abstract, it is possible to provide convergence guarantees as a function of the learning rate matrices $(\mathrm{diag}((\gamma_{k,i})_i), \mathrm{diag}((\tilde{\gamma}_{k,i})_i)$ in both the deterministic and stochastic settings. Further, we analyze the spectral properties of these learning updates by analyzing the Jacobian of the continuous-time limiting dynamics $\dot{x} = -g_{(\cdot)}(x)$ where $(\cdot)$ is a place holder for the type of conjecture based dynamics being analyzed. Essentially, the structure of the Jacobian at critical points of $g(x) = (D_1 f_1(x), D_2 f_2(x))$ in some classes of games—e.g., zero-sum or potential games—enables us to employ spectral operator theory [15] to bound the real and imaginary components of the spectrum, leading to insights into how the rotational and potential components of the vector field change near critical points.

3 Numerical Examples

We present several examples that illustrate how different conjecture-based updates affect the vector field near equilibria relative to simultaneous gradient play.

Example Class 1. Path-angle near critical points. In the following examples, we show the effects of different conjectures on the vector field near differential Nash equilibria. Warping of the vector field can be beneficial to individual players—by, e.g., reducing the accumulated cost along the learning path—or to all players—by, e.g., speeding up convergence. We consider two examples: 1) a non-zerosum quadratic game (f_1, f_2) defined by costs $f_1(x_1, x_2) = 0.5x_1^2 - 5.1x_1x_2 - 0.5x_2^2$ and $f_2(x_1, x_2) = -5x_1^2 + 2x_1x_2 + 2x_2^2$; 2) a zerosum polynomial game $(f, -f)$ defined by $f(x_1, x_2) = (-(x_1 + ax_2^2)^2 - (bx_1^2 + cx_2)^2) e^{-0.01(x_1^2 + x_2^2)}$ for $(a, b, c) = (0.5, 0.3, 0)$.

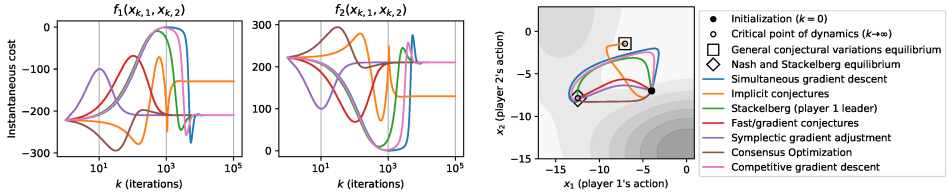
There are several metrics used in machine learning to assess convergence of learning algorithms to game theoretic equilibria including the path-norm (i.e., the norm of the vector field in the neighborhood of a critical point), the eigenvalues of the Jacobian, and the path-angle which is defined by $c(\alpha) = \langle x' - x, g_\alpha \rangle / (\|x' - x\| \|g_\alpha\|)$ for a point x' near a critical point and along the learning path, another point x in a neighborhood, and $g_\alpha = g(\alpha x' - (1 - \alpha)x)$, $\alpha \in [a, b]$ for some $0 < a < b$ [2]. In Fig. 1, we show visualizations of the path-angle and vector field for different updates, where the former metrizes the effect of the update on the latter. In both cases, the vector field near the differen-



(a) Quad: `simgrad` (b) Quad: `fconj` (c) Quad: `gradconj` (d) Poly: `simgrad` (e) Poly: `gradconj`

(f) Quad: `simgrad` (g) Quad: `fconj` (h) Quad: `gradconj` (i) Poly: `simgrad` (j) Poly: `gradconj`

Figure 1: Path-angle (a–e) and vector field/trajectory (f–j) visualizations for non-zerosum quadratic (Quad) and zerosum polynomial (Poly) games using updates: simultaneous gradient play (`simgrad`), fast conjectures (`fconj`), and gradient-based conjectures (`gradconj`) each with $\tilde{\gamma} = 1$ and learning rates $\gamma = 5 \times 10^{-3}$. In (f–h), the black diamond indicates the Nash equilibrium which is at $(0, 0)$ for a quadratic game. In (i–j), the red dots indicate critical points—only the one near $(-1, -1)$ is a non-Nash attractor and the rest are differential Nash.



(a) Player costs along learning trajectories. (b) Learning trajectories in the joint action space.

Figure 2: Polynomial zero-sum game: Comparison of multi-agent learning rules. Implicit conjecture learning converges to a stable GCVE with a more equitable cost for both players, whereas the remaining updates converge to a less equitable solution. The former critical point is neither a Nash nor Stackelberg equilibrium, whereas the latter critical point is simultaneously a stable differential Nash [13], differential Stackelberg [4], and a stable differential GCVE.

tial Nash equilibrium is primarily rotational for `simgrad`. Both `gradconj` and `fconj` dampen out the rotational component, encouraging a vector field that resembles one closer to a potential flow. For the non-zerosum quadratic game, the two conjecture based updates have a similar impact on the path-angle metric, however, they affect the vector fields differently as seen in Figs. 1f–1j.

Example Class 2. Costs along trajectory of zero-sum polynomial game. In this example, we plot the players' costs along the learning trajectories of each learning algorithm for polynomial zero-sum game as above with $(a, b, c) = (0.5, 0.3, 0.25)$. Fig. 2(a) shows the cost of player 1 and 2 over time and Fig. 2(b) shows their learning paths. Implicit conjectures converges to a general conjectural variations equilibrium that has a more equitable cost than the equilibrium of the other updates. Fast and gradient conjectures yield the same learning path in zero-sum games. We use learning rates $\gamma_i = 10^{-4}$ and conjecture learning rates $\tilde{\gamma}_i = 0.2$ and compare the other updates from Table 1 using $\alpha = 0.2$ for consensus optimization and $\alpha = 0.01$ for competitive gradient descent.

4 Discussion

The work presented in this paper explores a framework for synthesizing multi-agent learning schemes. By choosing different conjectures for implicit, gradient or fast conjectures, both the dynamics and the equilibria can be designed with desirable properties. Future research may pursue decoupled learning schemes where agents do not require access to each other's costs or gradients. As applications in machine learning demand coupled losses, such as markets or reinforcement learning, the community should seek out principled tools to synthesize novel and effective game dynamics.

References

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

[2] Hugo Berard, Gauthier Gidel, Amjad Almahairi, Pascal Vincent, and Simon Lacoste-Julien. A closer look at the optimization landscapes of generative adversarial networks. *arXiv preprint arXiv:1906.04848*, 2019.

[3] Timothy F Bresnahan. Duopoly models with consistent conjectures. *The American Economic Review*, 71(5):934–945, 1981.

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

[5] Charles Figuières. *Theory of conjectural variations*, volume 2. World Scientific, 2004.

[6] Jakob Foerster, Richard Y Chen, Maruan Al-Shedivat, Shimon Whiteson, Pieter Abbeel, and Igor Mordatch. Learning with opponent-learning awareness. In *International Conference on Autonomous Agents and MultiAgent Systems*, pages 122–130, 2018.

[7] James W Friedman and Claudio Mezzetti. Bounded rationality, dynamic oligopoly, and conjectural variations. *Journal of Economic Behavior & Organization*, 49(3):287–306, 2002.

[8] Drew Fudenberg, Fudenberg Drew, David K Levine, and David K Levine. *The theory of learning in games*, volume 2. MIT press, 1998.

[9] Alistair Letcher, Jakob Foerster, David Balduzzi, Tim Rocktäschel, and Shimon Whiteson. Stable opponent shaping in differentiable games. In *International Conference on Learning Representations*, 2019.

[10] Thomas Lindh. The inconsistency of consistent conjectures: Coming back to cournot. *Journal of Economic Behavior & Organization*, 18(1):69–90, 1992.

[11] E. Mazumdar, M. Jordan, and S. S. Sastry. On finding local nash equilibria (and only local nash equilibria) in zero-sum games. *arxiv:1901.00838*, 2019.

[12] Lars Mescheder, Sebastian Nowozin, and Andreas Geiger. The numerics of gans. In *Advances in Neural Information Processing Systems*, pages 1825–1835, 2017.

[13] L. J. Ratliff, S. A. Burden, and S. S. Sastry. On the Characterization of Local Nash Equilibria in Continuous Games. *IEEE Transactions on Automatic Control*, 61(8):2301–2307, 2016.

[14] Florian Schäfer and Anima Anandkumar. Competitive gradient descent. *arXiv preprint arXiv:1905.12103*, 2019.

[15] C. Tretter. *Spectral Theory of Block Operator Matrices and Applications*. World Scientific, 2008.

[16] Chongjie Zhang and Victor Lesser. Multi-agent learning with policy prediction. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.