Open-ended Learning in Symmetric Zero-sum Games

David Balduzzi [1] Marta Garnelo [1] Yoram Bachrach [1] Wojciech M. Czarnecki [1] Julien Perolat [1]
Max Jaderberg [1] Thore Graepel [1]

Abstract

Zero-sum games such as chess and poker are, abstractly, functions that evaluate pairs of agents, for example labeling them 'winner' and 'loser'. If the game is approximately transitive, then self-play generates sequences of agents of increasing strength. However, nontransitive games, such as rock-paper-scissors, can exhibit strategic cycles, and there is no longer a clear objective – we want agents to increase in strength, but against whom is unclear. In this paper, we introduce a geometric framework for formulating agent objectives in zero-sum games, in order to construct adaptive sequences of objectives that yield open-ended learning. The framework allows us to reason about population performance in nontransitive games, and enables the development of a new algorithm (rectified Nash response, $\mathsf{PSRO_{rN}}$) that uses game-theoretic niching to construct diverse populations of effective agents, producing a stronger set of agents than existing algorithms. We apply $\mathsf{PSRO_{rN}}$ to two highly nontransitive resource allocation games and find that $\mathsf{PSRO_{rN}}$ consistently outperforms the existing alternatives.

1. Introduction

A story goes that a Cambridge tutor in the mid-19th century once proclaimed: "I'm teaching the smartest boy in Britain." His colleague retorted: "I'm teaching the best test-taker." Depending on the version of the story, the first boy was either Lord Kelvin or James Clerk Maxwell. The second boy, who indeed scored highest on the Tripos, is long forgotten.

Modern learning algorithms are outstanding test-takers: once a problem is packaged into a suitable objective, deep (reinforcement) learning algorithms often find a good solution. However, in many multi-agent domains, the question of what test to take, or what objective to optimize, is not clear. This paper proposes algorithms that adaptively and continually pose new, useful objectives which result in open-ended learning in two-player zero-sum games. This setting has a large scope of applications and is general enough to include function optimization as a special case.

Learning in games is often conservatively formulated as training agents that tie or beat, on average, a fixed set of opponents. However, the dual task, that of *generating useful opponents to train and evaluate against*, is under-studied. It is not enough to beat the agents you know; it is also important to generate better opponents, which exhibit behaviours that you don't know.

There are very successful examples of algorithms that pose and solve a series of increasingly difficult problems for themselves through forms of self-play (Silver et al., 2018; Jaderberg et al., 2018; Bansal et al., 2018; Tesauro, 1995). Unfortunately, it is easy to encounter nontransitive games where self-play cycles through agents without improving overall agent strength – simultaneously improving against one opponent and worsening against another. In this paper, we develop a mathematical framework for analyzing nontransitive games, and present algorithms that systematically uncover and solve the latent problems embedded in a game.

Overview. The paper starts in Section 2 by introducing functional-form games (FFGs) as a new mathematical model of zero-sum games played by parametrized agents such as neural networks. Theorem 1 decomposes any FFG into a sum of transitive and cyclic components. Transitive games, and closely related monotonic games, are the natural setting for self-play, but the cyclic components, present in non-transitive games, require more sophisticated algorithms which motivates the remainder of the paper.

The main problem in tackling nontransitive games, where there is not necessarily a best agent, is understanding what the objective should be. In Section 3, we formulate the global objective in terms of **gamescapes** – convex polytopes that encode the interactions between agents in a game. If the game is transitive or monotonic, then the gamescape degenerates to a one-dimensional landscape. In nontransitive games, the gamescape can be high-dimensional because training against one agent can be fundamentally different

[1] DeepMind. Correspondence to: <dbalduzzi@google.com>.

Proceedings of the 36th International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

from training against another.

Measuring the performance of individual agents is vexed in nontransitive games. Therefore, in Section 3, we develop tools to analyze populations of agents, including a population-level measure of performance, definition 3. An important property of population-level performance is that it increases transitively as the gamescape polytope expands in a nontransitive game. Thus, we reformulate the problem of learning in games from finding the best agent to growing the gamescape. We consider two approaches to do so, one directly performance related, and the other focusing on a measure of diversity, definition 4. Crucially, the measure quantifies diverse *effective behaviors* – we are not interested in differences in policies that do not lead to differences in outcomes, nor in agents that lose in new and surprising ways.

Section 4 presents two algorithms, one old and one new, for growing the gamescape. The algorithms can be seen as specializations of the policy space response oracle (PSRO) introduced in Lanctot et al. (2017). The first algorithm is Nash response ($\mathsf{PSRO_N}$), which is an extension to functional-form games of the double oracle algorithm from McMahan et al. (2003). Given a population, Nash response creates an objective to train against by averaging over the Nash equilibrium. The Nash serves as a proxy for the notion of 'best agent', which is not guaranteed to exist in general zero-sum games. A second, complementary algorithm is the rectified Nash response ($\mathsf{PSRO_{rN}}$). The algorithm amplifies strategic diversity in populations of agents by adaptively constructing *game-theoretic niches* that encourage agents to 'play to their strengths and ignore their weaknesses'.

Finally, in Section 5, we investigate the performance of these algorithms in Colonel Blotto (Borel, 1921; Tukey, 1949; Roberson, 2006) and a differentiable analog we refer to as differentiable Lotto. Blotto-style games involve allocating limited resources, and are highly nontransitive. We find that $\mathsf{PSRO_{rN}}$ outperforms $\mathsf{PSRO_N}$, both of which greatly outperform self-play in these domains. We also compare against an algorithm that responds to the uniform distribution $\mathsf{PSRO_U}$, which performs comparably to $\mathsf{PSRO_N}$.

Related work. There is a large literature on novelty search, open-ended evolution, and curiosity, which aim to continually expand the frontiers of game knowledge within an agent (Lehman & Stanley, 2008; Taylor et al., 2016; Banzhaf et al., 2016; Brant & Stanley, 2017; Pathak et al., 2017; Wang et al., 2019). A common thread is that of *adaptive objectives* which force agents to keep improving. For example, in novelty search, the target objective constantly changes – and so cannot be reduced to a fixed objective to be optimized once-and-for-all.

We draw heavily on prior work on learning in games, especially Heinrich et al. (2015); Lanctot et al. (2017) which are discussed below. Our setting resembles multiobjective optimization (Fonseca & Fleming, 1993; Miettinen, 1998). However, unlike multiobjective optimization, we are concerned with *both generating and optimizing* objectives. Generative adversarial networks (Goodfellow et al., 2014) are zero-sum games that do *not* fall under the scope of this paper due to lack of symmetry, see appendix **??**.

Notation. Vectors are columns. The constant vectors of zeros and ones are $\mathbf{0}$ and $\mathbf{1}$. We sometimes use $\mathbf{p}[i]$ to denote the i^{th} entry of vector $\mathbf{p}$. Proofs are in the appendix.

2. Functional-form games (FFGs)

Suppose that, given any pair of agents, we can compute the probability of one beating the other in a game such as Go, Chess, or StarCraft. We formalize the setup as follows.

Definition 1. *Let W be a set of agents parametrized by, say, the weights of a neural net. A **symmetric zero-sum functional-form game** (FFG) is an* antisymmetric *function, $\phi(\mathbf{v},\mathbf{w}) = -\phi(\mathbf{w},\mathbf{v})$, that evaluates pairs of agents*

$$\phi : W \times W \to \mathbb{R}.$$

The higher $\phi(\mathbf{v},\mathbf{w})$, the better for agent $\mathbf{v}$. We refer to $\phi > 0$, $\phi < 0$, and $\phi = 0$ as wins, losses and ties for $\mathbf{v}$.

Note that (i) the strategies in a FFG are *parametrized agents* and (ii) the parametrization of the agents is folded into ϕ, so the game is a composite of the agent's architecture and the environment itself.

Suppose the probability of $\mathbf{v}$ beating $\mathbf{w}$, denoted $P(\mathbf{v} \succ \mathbf{w})$ can be computed or estimated. Win/loss probabilities can be rendered into antisymmetric form via $\phi(\mathbf{v},\mathbf{w}) := P(\mathbf{v} \succ \mathbf{w}) - \frac{1}{2}$ or $\phi(\mathbf{v},\mathbf{w}) := \log \frac{P(\mathbf{v} \succ \mathbf{w})}{P(\mathbf{v} \prec \mathbf{w})}$.

Tools for FFGs. Solving FFGs requires different methods to solving normal form games (Shoham & Leyton-Brown, 2008) due to their continuous nature. We therefore develop the following basic tools.

First, the **curry** operator converts a two-player game into a *function from agents to objectives*

$$\begin{array}{ccc} \Big[\phi : W \times W \longrightarrow \mathbb{R}\Big] & \xrightarrow{\text{curry}} & \Big[W \longrightarrow \big[W \longrightarrow \mathbb{R}\big]\Big] \\ \phi(\mathbf{v},\mathbf{w}) & & \mathbf{w} \mapsto \phi_{\mathbf{w}}(\bullet) := \phi(\bullet,\mathbf{w}) \end{array}$$

Second, an **approximate best-response oracle** that, given agent $\mathbf{v}$ and objective $\phi_{\mathbf{w}}(\bullet)$, returns a new agent $\mathbf{v}' := \mathsf{oracle}(\mathbf{v}, \phi_{\mathbf{w}}(\bullet))$ with $\phi_{\mathbf{w}}(\mathbf{v}') > \phi_{\mathbf{w}}(\mathbf{v}) + \epsilon$, if possible. The oracle could use gradients, reinforcement learning or evolutionary algorithms.

Third, given a population $\mathfrak{P}$ of n agents, the $(n \times n)$ antisymmetric **evaluation matrix** is

$$\mathbf{A}_{\mathfrak{P}} := \Big\{\phi(\mathbf{w}_i, \mathbf{w}_j) \;:\; (\mathbf{w}_i, \mathbf{w}_j) \in \mathfrak{P} \times \mathfrak{P}\Big\} =: \phi(\mathfrak{P} \otimes \mathfrak{P}).$$

Fourth, we will use the (not necessarily unique) **Nash equilibrium** on the zero-sum matrix game specified by $\mathbf{A}_{\mathfrak{P}}$.

Finally, we use the following **game decomposition.** Suppose W is a compact set equipped with a probability measure. The set of integrable antisymmetric functions on W then forms a vector space. Appendix D shows the following:

Theorem 1 (game decomposition)**.** *Every* FFG *decomposes into a sum of a transitive and cyclic game*

$$\mathsf{FFG} = \textit{transitive game} \oplus \textit{cyclic game}.$$

with respect to a suitably defined inner product.

Transitive and cyclic games are discussed below. Few games are purely transitive or cyclic. Nevertheless, understanding these cases is important since general algorithms should, at the very least, work in both special cases.

2.1. Transitive games

A game is **transitive** if there is a 'rating function' f such that performance on the game is the difference in ratings:

$$\phi(\mathbf{v}, \mathbf{w}) = f(\mathbf{v}) - f(\mathbf{w}).$$

In other words, if ϕ admits a 'subtractive factorization'.

Optimization (training against a fixed opponent). Solving a transitive game reduces to finding

$$\mathbf{v}^* := \operatorname*{argmax}_{\mathbf{v} \in W} \phi_{\mathbf{w}}(\mathbf{v}) = \operatorname*{argmax}_{\mathbf{v} \in W} f(\mathbf{v}).$$

Crucially, the choice of opponent $\mathbf{w}$ makes no difference to the solution. The simplest learning algorithm is thus to train against a fixed opponent, see algorithm 1.

Algorithm 1 Optimization (against a fixed opponent)

```
input: opponent $\mathbf{w}$; agent $\mathbf{v}_1$
fix objective $\phi_{\mathbf{w}}(\bullet)$
for $t = 1, \dots, T$ do
    $\mathbf{v}_{t+1} \leftarrow \mathsf{oracle}\big(\mathbf{v}_t, \phi_{\mathbf{w}}(\bullet)\big)$
end for
output: $\mathbf{v}_{T+1}$
```

Monotonic games generalize transitive games. An FFG is monotonic if there is a monotonic function σ such that

$$\phi(\mathbf{v}, \mathbf{w}) = \sigma\big(f(\mathbf{v}) - f(\mathbf{w})\big). \tag{1}$$

For example, Elo (1978) models the probability of one agent beating another by

$$P(\mathbf{v} \succ \mathbf{w}) = \sigma\big(f(\mathbf{v}) - f(\mathbf{w})\big) \text{ for } \sigma(x) = \frac{1}{1 + e^{-\alpha \cdot x}}$$

for some $\alpha > 0$, where f assigns Elo ratings to agents. The model is widely used in Chess, Go and other games.

Optimizing against a fixed opponent fares badly in monotonic games. Concretely, if Elo's model holds then training against a much weaker opponent yields no learning signal because the gradient vanishes $\nabla_{\mathbf{v}}\phi(\mathbf{v}_t, \mathbf{w}) \approx 0$ once the sigmoid saturates when $f(\mathbf{v}_t) \gg f(\mathbf{w})$.

Self-play (algorithm 2) generates a sequence of opponents. Training against a sequence of opponents of increasing strength prevents gradients from vanishing due to large skill differentials, so self-play is well-suited to games modeled by eq. (1). Self-play has proven effective in Chess, Go and other games (Silver et al., 2018; Al-Shedivat et al., 2018).

Algorithm 2 Self-play

```
input: agent $\mathbf{v}_1$
for $t = 1, \dots, T$ do
    $\mathbf{v}_{t+1} \leftarrow \mathsf{oracle}\big(\mathbf{v}_t, \phi_{\mathbf{v}_t}(\bullet)\big)$
end for
output: $\mathbf{v}_{T+1}$
```

Self-play *is* an open-ended learning algorithm: it poses and masters a sequence of objectives, rather than optimizing a pre-specified objective. However, self-play assumes transitivity: that local improvements ($\mathbf{v}_{t+1}$ beats $\mathbf{v}_t$) imply global improvements ($\mathbf{v}_{t+1}$ beats $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_t$). The assumption fails in nontransitive games, such as the disc game below. Since performance is nontransitive, improving against one agent does not guarantee improvements against others.

2.2. Cyclic games

A game is **cyclic** if

$$\int_W \phi(\mathbf{v}, \mathbf{w}) \cdot d\mathbf{w} = 0 \quad \text{for all} \quad \mathbf{v} \in W. \tag{2}$$

In other words, wins against some agents are necessarily counterbalanced with losses against others. Strategic cycles often arise when agents play simultaneous move or imperfect information games such as rock-paper-scissors, poker, or StarCraft.

Example 1 (Disc game)**.** *Fix $k > 0$. Agents are $W = \{\mathbf{x} \in \mathbb{R}^2 : \|\mathbf{x}\|_2^2 \leq k\}$ with the uniform distribution. Set*

$$\phi(\mathbf{v}, \mathbf{w}) = \mathbf{v}^{\intercal} \cdot \begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix} \cdot \mathbf{w} = v_1 w_2 - v_2 w_1.$$

The game is cyclic, see figure 2A.

Example 2 (Rock-paper-scissors embeds in disc game)**.** *Set* $\mathfrak{r}_\epsilon = \frac{\sqrt{3}\epsilon}{2}(\cos 0, \sin 0)$, $\mathfrak{p}_\epsilon = \frac{\sqrt{3}\epsilon}{2}(\cos\frac{2\pi}{3}, \sin\frac{2\pi}{3})$ *and* $\mathfrak{s}_\epsilon = \frac{\sqrt{3}\epsilon}{2}(\cos\frac{4\pi}{3}, \sin\frac{4\pi}{3})$ *to obtain*

$$\mathbf{A}_{\{\mathfrak{r}_\epsilon, \mathfrak{p}_\epsilon, \mathfrak{s}_\epsilon\}} = \begin{bmatrix} 0 & \epsilon^2 & -\epsilon^2 \\ -\epsilon^2 & 0 & \epsilon^2 \\ \epsilon^2 & -\epsilon^2 & 0 \end{bmatrix}.$$

Varying $\epsilon \in [0,1]$ yields a family of $\mathfrak{r}$-$\mathfrak{p}$-$\mathfrak{s}$ *interactions that trend deterministic as ϵ increases, see figure 2B.*

Our goal is to extend self-play to general FFGs. The success of optimization and self-play derives from **(i)** repeatedly applying a local operation that **(ii)** improves a transitive measure. If the measure is *not* transitive, then applying a sequence of local improvements can result in no improvement at all. Our goal is thus to find practical substitutes for **(i)** and **(ii)** in general FFGs.

3. Functional and Empirical Gamescapes

Rather than trying to find a single dominant agent which may not exist, we seek to find all the atomic components in "strategy space" of a zero-sum game. That is, we aim to discover the underlying strategic dimensions of the game, and the best ways of executing them. Given such knowledge, when faced with a new opponent, we will not only be able to react to its behavior conservatively (using the Nash mixture to guarantee a tie), but will also be able to optimally exploit the opponent. As opposed to typical game-theoretic solutions, we do not seek a single agent or mixture, but rather a population that embodies a complete understanding of the strategic dimensions of the game.

To formalize these ideas we introduce **gamescapes**, which geometrically represent agents in functional form games. We show some general properties of these objects to build intuitions for the reader. Finally we introduce two critical concepts: **population performance**, which measures the progress in performance of populations, and **effective diversity**, which quantifies the coverage of the gamescape spanned by a population. Equipped with these tools we present algorithms that guarantee iterative improvements in FFGs.

Definition 2. *The* ***functional gamescape*** *(*FGS*) of $\phi : W \times W \to \mathbb{R}$ is the convex set*

$$\mathcal{G}_\phi := \text{hull}\left(\left\{\phi_{\mathbf{w}}(\bullet) \,:\, \mathbf{w} \in W\right\}\right) \subset \mathcal{C}(W, \mathbb{R}),$$

where $\mathcal{C}(W,\mathbb{R})$ is the space of real-valued functions on W.

Given population $\mathfrak{P}$ of n agents with evaluation matrix $\mathbf{A}_{\mathfrak{P}}$, the corresponding ***empirical gamescape*** *(*EGS*) is*

$$\mathcal{G}_{\mathfrak{P}} := \left\{\textit{convex mixtures of rows of } \mathbf{A}_{\mathfrak{P}}\right\}.$$

The FGS represents all the mixtures of objectives implicit in the game. We cannot work with the FGS directly because we cannot compute $\phi_{\mathbf{w}}(\bullet)$ for infinitely many agents. The EGS is a tractable proxy (Wellman, 2006). The two gamescapes represent all the ways agents can-in-principle and are-actually-observed-to interact respectively. The remainder of this section collects basic facts about gamescapes.

Optimization landscapes are a special case of gamescapes. If $\phi(\mathbf{v},\mathbf{w}) = f(\mathbf{v}) - f(\mathbf{w})$ then the FGS is, modulo constant terms, a single function $\mathcal{G}_\phi = \left\{f(\bullet) - f(\mathbf{w}) : \mathbf{w} \in W\right\}$. The FGS degenerates into a landscape where, for each agent $\mathbf{v}$ there is a unique direction $\nabla\phi_{\mathbf{w}}(\mathbf{v}) = \nabla f(\mathbf{v})$ in weight space which gives the steepest performance increase *against all opponents*. In a monotonic game, the gradient is $\nabla\phi_{\mathbf{w}}(\mathbf{v}) = \sigma' \cdot \nabla f(\mathbf{v})$. There is again a single steepest direction $\nabla f(\mathbf{v})$, with tendency to vanish controled by the ratings differential $\sigma' = \sigma'\big(f(\mathbf{v}) - f(\mathbf{w})\big) \geq 0$.

Redundancy. First, we argue that gamescapes are more fundamental than evaluation matrices. Consider

$$\begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} 0 & 1 & -1 & -1 \\ -1 & 0 & 1 & 1 \\ 1 & -1 & 0 & 0 \\ 1 & -1 & 0 & 0 \end{bmatrix}.$$

The first matrix encodes rock-paper-scissors interactions; the second is the same, but with two copies of scissors. The matrices are difficult to compare since their dimensions are different. Nevertheless, the gamescapes are equivalent triangles embedded in $\mathbb{R}^3$ and $\mathbb{R}^4$ respectively.

Proposition 2. *An agent in a population is redundant if it is behaviorally identical to a convex mixture of other agents. The* EGS *is* ***invariant*** *to redundant agents.*

Invariance is explained in appendix E.

Dimensionality. The dimension of the gamescape is an indicator of the complexity of both the game and the agents playing it. In practice we find many FFGs have a low dimensional latent structure.

Figure 1 depicts evaluation matrices of four populations of 40 agents. Although the gamescapes could be 40-dim, they turn out to have one- and two-dim representations. The dimension of the EGS is determined by the rank of the evaluation matrix.

Proposition 3. *The* EGS *of n agents in population $\mathfrak{P}$ can be represented in $\mathbb{R}^r$, where $r = \text{rank}(\mathbf{A}_{\mathfrak{P}}) \leq n$.*

A low-dim representation of the EGS can be constructed via the Schur decomposition, which is the analog of PCA for antisymmetric matrices (Balduzzi et al., 2018b). The length of the longest strategic cycle in a game gives a lower-bound on the dimension of its gamescape:

Example 3 (latent dimension of long cycles)**.** *Suppose n agents form a long cycle: $\mathfrak{P} = \{\mathbf{v}_1 \xrightarrow{beats} \mathbf{v}_2 \to \cdots \to \mathbf{v}_n \xrightarrow{beats} \mathbf{v}_1\}$. Then rank$(\mathbf{A}_{\mathfrak{P}})$ is $n-2$ if n is even and $n-1$ if n is odd.*

Nash equilibria in a symmetric zero-sum game are (mixtures of) agents that beat or tie all other agents. Loosely speaking, they replace the notion of best agent in games

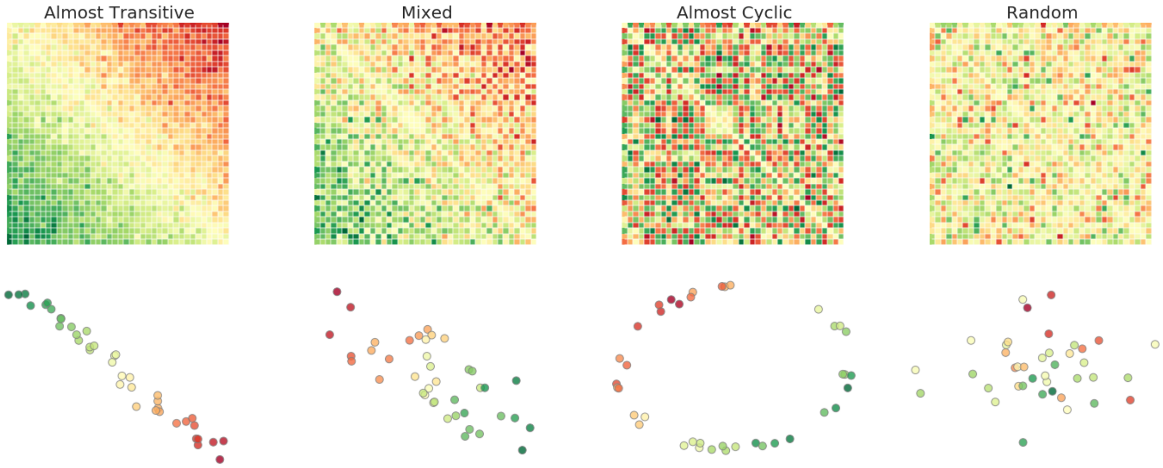

Figure 1. Low-dim gamescapes of various basic game structures. **Top row:** Evaluation matrices of populations of 40 agents each; colors vary from red to green as ϕ ranges over $[-1, 1]$. **Bottom row:** 2-dim embedding obtained by using first 2 dimensions of Schur decomposition of the payoff matrix; Color corresponds to average payoff of an agent against entire population; EGS of the transitive game is a line; EGS of the cyclic game is two-dim near-circular polytope given by convex hull of points. For extended version see Figure 6 in the Appendix.

where there is no best agent. Functional Nash equilibria, in the FGS, are computationally intractable so we work with empirical Nash equilibria over the evaluation matrix.

Proposition 4. *Given population $\mathfrak{P}$, the empirical Nash equilibria are*

$$\mathcal{N}_{\mathfrak{P}} = \{\mathbf{p} \textit{ distribution} : \mathbf{p}^\intercal \mathbf{A}_{\mathfrak{P}} \succeq \mathbf{0}\}.$$

In other words, Nash equilibria correspond to points in the empirical gamescape that intersect the positive quadrant $\{\mathbf{x} \in \mathcal{G}_{\mathfrak{P}} : \mathbf{x} \succeq \mathbf{0}\}$. The positive quadrant thus provides a set of directions in weight space to aim for when training new agents, see PSRO$_\mathsf{N}$ below.

The gap between the EGS and FGS. Observing $\mathfrak{r}$-$\mathfrak{p}$ interactions yields different conclusions from observing $\mathfrak{r}$-$\mathfrak{p}$-$\mathfrak{s}$ interactions; it is always possible that an agent that appears to be dominated is actually part of a partially observed cycle. Without further assumptions about the structure of ϕ, it is impossible to draw strong conclusions about the nature of the FGS from the EGS computed from a finite population. The gap is analogous to the exploration problem in reinforcement learning. To discover unobserved dimensions of the FGS one could train against randomized distributions over opponents, which would eventually find them all.

3.1. Population performance

If $\phi(\mathbf{v}, \mathbf{w}) = f(\mathbf{v}) - f(\mathbf{w})$ then improving performance of agent $\mathbf{v}$ reduces to increasing $f(\mathbf{v})$. In a cyclic game, the performance of individual agents is meaningless: beating one agent entails losing against another by eq. (2). We therefore propose a *population* performance measure.

Definition 3. *Given populations $\mathfrak{P}$ and $\mathfrak{Q}$, let $(\mathbf{p}, \mathbf{q})$ be a Nash equilibrium of the zero-sum game on $\mathbf{A}_{\mathfrak{P},\mathfrak{Q}} := \phi(\mathbf{v}, \mathbf{w})_{\mathbf{v}\in\mathfrak{P},\mathbf{w}\in\mathfrak{Q}}$. The* ***relative population performance*** *is*

$$v(\mathfrak{P}, \mathfrak{Q}) := \mathbf{p}^\intercal \cdot \mathbf{A}_{\mathfrak{P},\mathfrak{Q}} \cdot \mathbf{q} = \sum_{i,j=1}^{n_1,n_2} A_{ij} \cdot p_i q_j.$$

Proposition 5. ***(i)*** *Performance v is independent of the choice of Nash equilibrium.* ***(ii)*** *If ϕ is monotonic then performance compares the best agents in each population*

$$v(\mathfrak{P}, \mathfrak{Q}) = \max_{\mathbf{v}\in\mathfrak{P}} f(\mathbf{v}) - \max_{\mathbf{w}\in\mathfrak{Q}} f(\mathbf{w}).$$

(iii) *If $\text{hull}(\mathfrak{P}) \subset \text{hull}(\mathfrak{Q})$ then $v(\mathfrak{P}, \mathfrak{Q}) \leq 0$ and $v(\mathfrak{P}, \mathfrak{R}) \leq v(\mathfrak{Q}, \mathfrak{R})$ for* ***any*** *population $\mathfrak{R}$.*

The first two properties are sanity checks. Property ***(iii)*** implies growing the polytope spanned by a population improves its performance *against any other population*.

Consider the concentric rock-paper-scissors populations in figure 2B and example 2. The Nash equilibrium is $(0, 0)$, which is a uniform mixture over any of the populations. Thus, the relative performance of any two populations is zero. However, the outer population is better than the inner population at *exploiting* an opponent that only plays, say, rock because the outer version of paper wins more deterministically than the inner version.

Finding a population that contains the Nash equilibrium is necessary but not sufficient to fully solve an FFG. For example, adding the ability to always force a tie to an FFG makes finding the Nash trivial. However, the game can still exhibit rich strategies and counter-strategies that are worth discovering.

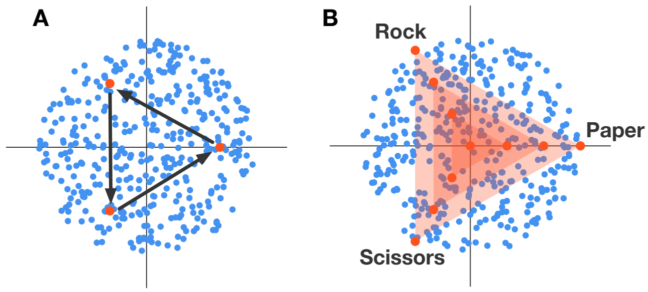

Figure 2. The disc game. **A:** A set of possible agents from the disc game is shown as blue dots. Three agents with non-transitive rock-paper-scissors relations are visualized in red. **B:** Three concentric gamescapes spanned by populations with rock-paper-scissor interactions of increasing strength.

3.2. Effective diversity

Measures of diversity typically quantify differences in weights or behavior of agents but ignore performance. *Effective* diversity measures the variety of effective agents (agents with support under Nash):

Definition 4. *Denote the rectifier by* $\lfloor x \rfloor_+ := x$ *if* $x \geq 0$ *and* $\lfloor x \rfloor_+ := 0$ *otherwise. Given population* $\mathfrak{P}$*, let* $\mathbf{p}$ *be a Nash equilibrium on* $\mathbf{A}_{\mathfrak{P}}$*. The* ***effective diversity*** *of the population is:*

$$d(\mathfrak{P}) := \mathbf{p}^\intercal \cdot \lfloor \mathbf{A}_{\mathfrak{P}} \rfloor_+ \cdot \mathbf{p} = \sum_{i,j=1}^{n} \lfloor \phi(\mathbf{w}_i, \mathbf{w}_j) \rfloor_+ \cdot p_i p_j.$$

Diversity quantifies how the best agents (those with support in the maximum entropy Nash) exploit each other. If there is a dominant agent then diversity is zero.

Effective diversity is a matrix norm, see appendix F.2. It measures the $\ell_{1,1}$ volume spanned by Nash supported agents. In figure 2B, there are four populations spanning concentric gamescapes: the Nash at $(0,0)$ and three variants of $\mathfrak{r}$-$\mathfrak{p}$-$\mathfrak{s}$. Going outwards to large gamescapes yields agents that are more diverse and better exploiters.

4. Algorithms

We now turn attention to constructing objectives that when trained against, produce new, effective agents. We present two algorithms that construct a sequence of fruitful local objectives that, when solved, iteratively add up to transitive population-level progress. Importantly, these algorithms output populations, unlike self-play which outputs single agents.

Concretely, we present algorithms that expand the empirical gamescape in useful directions. Following Lanctot et al. (2017), we assume access to a subroutine, or **oracle**, that finds an approximate best response to any mixture $\sum_i p_i \phi_i(\mathbf{w}_i)$ of objectives. The subroutine could be a gradient-based, reinforcement learning or evolutionary algorithm. The subroutine returns a vector in weight-space, in which existing agents can be shifted to create new agents. Any mixture constitutes a valid training objective. However, many mixtures do not grow the gamescape, because the vector could point towards redundant or weak agents.

4.1. Response to the Nash (PSRO$_\mathsf{N}$)

Since the notion of 'the best agent' – one agent that beast all others – does not necessarily exist in nontransitive games, a natural substitute is the mixture over the Nash equilibrium on the most recent population $\mathfrak{P}_t$. The policy space response to the Nash (PSRO$_\mathsf{N}$) iteratively generates new agents that are approximate best responses to the Nash mixture. If the game is transitive then PSRO$_\mathsf{N}$ degenerates to self-play. The algorithm is an extension of the double oracle algorithm (McMahan et al., 2003) to FFGs, see also (Zinkevich et al., 2007; Hansen et al., 2008).

Algorithm 3 Response to Nash (PSRO$_\mathsf{N}$)

```
input: population $\mathfrak{P}_1$ of agents
for $t = 1, \ldots, T$ do
    $\mathbf{p}_t \leftarrow$ Nash on $\mathbf{A}_{\mathfrak{P}_t}$
    $\mathbf{v}_{t+1} \leftarrow \text{oracle}\left(\mathbf{v}_t, \sum_{\mathbf{w}_i \in \mathfrak{P}_t} \mathbf{p}_t[i] \cdot \phi_{\mathbf{w}_i}(\bullet)\right)$
    $\mathfrak{P}_{t+1} \leftarrow \mathfrak{P}_t \cup \{\mathbf{v}_{t+1}\}$
end for
output: $\mathfrak{P}_{T+1}$
```

The following result shows that PSRO$_\mathsf{N}$ strictly enlarges the empirical gamescape:

Proposition 6. *If* $\mathbf{p}$ *is a Nash equilibrium on* $\mathbf{A}_{\mathfrak{P}}$ *and* $\sum_i p_i \phi_{\mathbf{w}_i}(\mathbf{v}) > 0$*, then adding* $\mathbf{v}$ *to* $\mathfrak{P}$ *strictly enlarges the empirical gamescape:* $\mathcal{G}_{\mathfrak{P}} \subsetneq \mathcal{G}_{\mathfrak{P} \cup \{\mathbf{v}\}}$.

A failure mode of PSRO$_\mathsf{N}$ arises when the Nash equilibrium of the game is contained in the empirical gamescape. For example, in the disc game in figure 2 the Nash equilibrium of the entire FFG is the agent at the origin $\mathbf{w} = (0,0)$. If a population's gamescape contains $\mathbf{w} = (0,0)$ – which is true of any $\mathfrak{r}$-$\mathfrak{p}$-$\mathfrak{s}$ subpopulation – then PSRO$_\mathsf{N}$ will not expand the gamescape because there is no ϵ-better response to $\mathbf{w} = (0,0)$. The next section presents an algorithm that uses *niching* to meaningfully grow the gamescape, even after finding the Nash equilibrium of the FFG.

Response to the uniform distribution (PSRO$_\mathsf{U}$). A closely related algorithm is fictitious (self-)play (Brown, 1951; Leslie & Collins, 2006; Heinrich et al., 2015). The algorithm finds an approximate best-response to the uniform distribution on agents in the current population: $\sum_{\mathbf{w}_i \in \mathfrak{P}_t} \phi_{\mathbf{w}_i}(\bullet)$. PSRO$_\mathsf{U}$ has guarantees in matrix form games and performs well empirically (see below). However, we do not currently understand its effect on the gamescape.

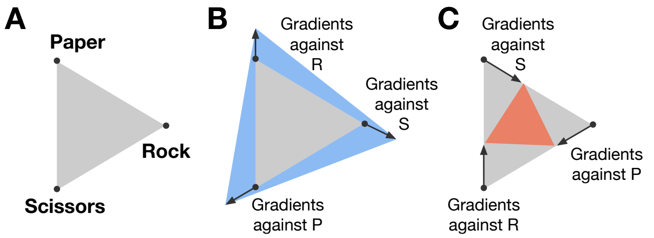

Figure 3. **A:** Rock-paper-scissors. **B:** Gradient updates obtained from PSRO_{rN}, amplifying strengths, grow gamescape (gray to blue). **C:** Gradients obtained by optimizing agents to reduces their losses shrink gamescape (gray to red).

4.2. Response to the rectified Nash (PSRO_{rN})

Response to the *rectified* Nash (PSRO_{rN}), introduces game-theoretic niches. Each effective agent – that is each agent with support under the Nash equilibrium – is trained against the Nash-weighted mixture of agents that it beats or ties. Intuitively, the idea is to encourage agents to 'amplify their strengths and ignore their weaknesses'.

A special case of PSRO_{rN} arises when there is a dominant agent in the population, that beats all other agents. The Nash equilibrium is then concentrated on the dominant agent, and PSRO_{rN} degenerates to training against the best agent in the population, which can be thought of as a form of self-play (assuming the best agent is the most recent).

Algorithm 4 Response to rectified Nash (PSRO_{rN})

```
input: population $\mathfrak{P}_1$
for $t = 1, \ldots, T$ do
    $\mathbf{p}_t \leftarrow$ Nash on $\mathbf{A}_{\mathfrak{P}_t}$
    for agent $\mathbf{v}_t$ with positive mass in $\mathbf{p}_t$ do
        $\mathbf{v}_{t+1} \leftarrow \text{oracle}\left(\mathbf{v}_t, \sum_{\mathbf{w}_i \in \mathfrak{P}_t} \mathbf{p}_t[i] \cdot \lfloor \phi_{\mathbf{w}_i}(\bullet) \rfloor_+\right)$
    end for
    $\mathfrak{P}_{t+1} \leftarrow \mathfrak{P}_t \cup \{\mathbf{v}_{t+1}$ : updated above$\}$
end for
output: $\mathfrak{P}_{T+1}$
```

Proposition 7. *The objective constructed by rectified Nash response is effective diversity, definition 4.*

Thus, PSRO_{rN} amplifies the positive coordinates, of the Nash-supported agents, in their rows of the evaluation matrix. A pathological mode of PSRO_{rN} is when there are many extremely local niches. That is, every agent admits a specific exploit that does not generalize to other agents. PSRO_{rN} will grow the gamescape by finding these exploits, generating a large population of highly specialized agents.

Rectified Nash responses in the disc game (example 1)**.** The disc game embeds many subpopulations with rock-paper-scissor dynamics. As the polytope they span expands outwards, the interactions go from noisy to deterministic.

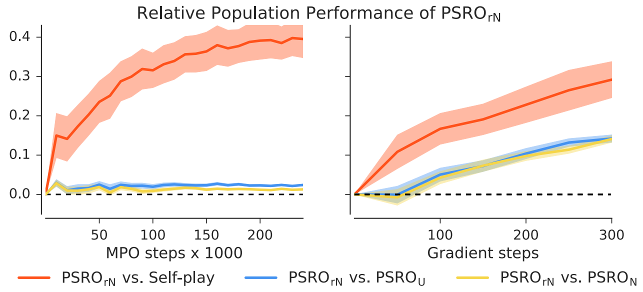

Figure 4. Performance of PSRO_{rN} relative to self-play, PSRO_{U} and PSRO_{N} on Blotto (left) and Differentiable Lotto (right). In all cases, the relative performance of PSRO_{rN} is positive, and therefore outperforms the other algorithms.

The disc game is differentiable, so we can use gradient-based learning for the oracle in PSRO_{rN}. Figure 3B depicts the gradients resulting from training each of rock, paper and scissors against the agent it beats. Since the gradients point *outside* the polytope, training against the rectified Nash mixtures expands the gamescape.

Why ignore weaknesses? A natural modification of PSRO_{rN} is to train effective agents against effective agents that they *lose* to. In other words, to force agents to improve their weak points whilst taking their strengths for granted. Figure 3C shows the gradients that would be applied to each of rock, paper and scissors under this algorithm. They point *inwards*, contracting the gamescape. Training rock against paper would make it more like scissors; similarly training paper against scissors would make it more like rock and so on. Perhaps counter-intuitively, building objectives using the weak points of agents does not encourage diverse niches.

5. Experiments

We investigated the performance of the proposed algorithms in two highly nontransitive resource allocation games.

Colonel Blotto is a resource allocation game that is often used as a model for electoral competition. Each of two players has a budget of c coins which they simultaneously distribute over a fixed number of areas. Area a_i is won by the player with more coins on a_i. The player that wins the most areas wins the game. Since Blotto is not differentiable we use maximum a posteriory policy optimization (MPO) (Abdolmaleki et al., 2018) as best response oracle. MPO is an inference-based policy optimization algorithm; many other reinforcement learning algorithms could be used.

Differentiable Lotto is inspired by continuous Lotto (Hart, 2008). The game is defined over a fixed set $\mathcal{C}$ of c 'customers', each being a point in $\mathbb{R}^2$. An agent $(\mathbf{p}, \mathbf{v}) = \{(p_1, \mathbf{v}_1), \ldots, (p_k, \mathbf{v}_k)\}$ distributes one unit of mass over

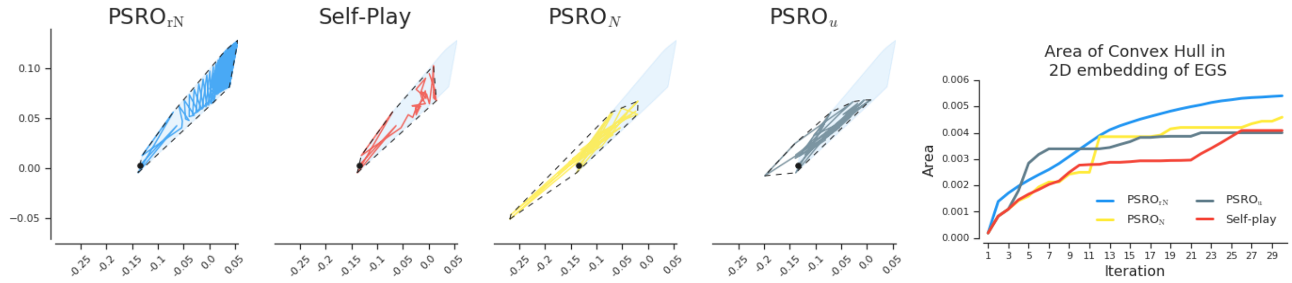

Figure 5. Visualizations of training progress in Differentiable Lotto experiment. **Left:** Comparison of trajectories taken by each algorithm in the 2-dim Schur embedding of the EGS; a black dot represents first agent found by the algorithm and a dashed line represents the convex full. Shaded blue region shows area of the convex hull of PSRO$_{\mathsf{rN}}$. Notice the PSRO$_{\mathsf{rN}}$ consistent expansion of the convex hull through ladder-like movements. See Figure 7 for an extended version. **Right:** Area of convex hull spanned by populations over time. Note that only PSRO$_{\mathsf{rN}}$ consistently increases the convex hull in all iterations.

k servers, where each server is a point $\mathbf{v}_i \in \mathbb{R}^2$. Roughly, given two agents $(\mathbf{p}, \mathbf{v})$ and $(\mathbf{q}, \mathbf{w})$, customers are softly assigned to the nearest servers, determining the agents' payoffs. More formally, the payoff is

$$\phi\big((\mathbf{p}, \mathbf{v}), (\mathbf{q}, \mathbf{w})\big) := \sum_{i,j=1}^{c,k} \big(p_j v_{ij} - q_j w_{ij}\big),$$

where the scalars v_{ij} and w_{ij} depend on the distance between customer i and the servers:

$$(v_{i1}, \ldots, w_{ik}) := \text{softmax}(-\|\mathbf{c}_i - \mathbf{v}_1\|^2, \ldots, -\|\mathbf{c}_i - \mathbf{w}_k\|^2).$$

The *width* of a cloud of points is the expected distance from the barycenter. We impose agents to have width equal one. We use gradient ascent as our oracle.

Experimental setups. The experiments examine the performance of self-play, PSRO$_{\mathsf{U}}$, PSRO$_{\mathsf{N}}$, and PSRO$_{\mathsf{rN}}$. We investigate performance under a fixed computational budget. Specifically, we track queries made to the oracle, following the standard model of computational cost in convex optimization (Nemirovski & Yudin, 1983). To compare two algorithms, we report the relative population performance (definition 3), of the populations they output. Computing evaluation matrices is expensive, $\mathcal{O}(n^2)$, for large populations. This cost is not included in our computational model since populations are relatively small. The relative cost of evaluations and queries to the oracle depends on the game.

In Blotto, we investigate performance for $a = 3$ areas and $c = 10$ coins over $k = 1000$ games. An agent outputs a vector in $\mathbb{R}^3$ which is passed to a softmax, ×10 and discretized to obtain three integers summing to 10. Differentiable Lotto experiments are from $k = 500$ games with $c = 9$ customers chosen uniformly at random in the square $[-1, 1]^2$.

Results. Fig 4 shows the relative population performance, definition 3, between PSRO$_{\mathsf{rN}}$ and each of PSRO$_{\mathsf{N}}$, PSRO$_{\mathsf{U}}$ and self-play: the more positive the number is, the more PSRO$_{\mathsf{rN}}$ outperforms the alternative method. We find that PSRO$_{\mathsf{rN}}$ outperforms the other approaches across a wide range of allowed compute budgets. PSRO$_{\mathsf{U}}$ and PSRO$_{\mathsf{N}}$ perform comparably, and self-play performs the worst. Self-play, algorithm 2, outputs a single agent, so the above comparison only considers the final agent. If we upgrade self-play to a population algorithm (by tracking all agents produced over), then it still performs the worst in differentiable Lotto, but by a smaller margin. In Blotto, suprisingly, it slightly outperforms PSRO$_{\mathsf{N}}$ and PSRO$_{\mathsf{U}}$.

Figure 5 shows how gamescapes develop during training. From the left panel, we see that PSRO$_{\mathsf{rN}}$ grows the polytope in a more uniform manner than the other algorithms. The right panel shows the area of the empirical gamescapes generated by the algorithms (the areas of the convex hulls). All algorithms increase the area, but PSRO$_{\mathsf{rN}}$ is the only method that increases the area at every iteration.

6. Conclusion

We have proposed a framework for open-ended learning in two-player symmetric zero-sum games – where strategies are agents, with a differentiable parametrization. We propose the goal of learning should be **(i)** to discover the underlying strategic components that constitute the game and **(ii)** to master each of them. We formalized these ideas using gamescapes, which geometrically represent the latent objectives in games, and provided tools to analyze them. Finally, we proposed and empirically validated a new algorithm, PSRO$_{\mathsf{rN}}$, for uncovering strategic diversity within functional form games.

The algorithms discussed here are simple and generic, providing the foundations for methods that unify modern gradient and reinforcement-based learning with the adaptive objectives derived from game-theoretic considerations. Future work lies in expanding this understanding and applying it to develop practical algorithms for more complex games.

References

Abdolmaleki, A., Springenberg, J. T., Tassa, Y., Munos, R., Heess, N., and Riedmiller, M. Maximum a Posteriori Policy Optimisation. In *ICLR*, 2018.

Al-Shedivat, M., Bansal, T., Burda, Y., Sutskever, I., Mordatch, I., and Abbeel, P. Continuous Adaptation via Meta-Learning in Nonstationary and Competitive Environments. In *ICLR*, 2018.

Baker, M. Hodge theory in combinatorics. *Bull. AMS*, 55 (1):57–80, 2018.

Balduzzi, D., Racanière, S., Martens, J., Foerster, J., Tuyls, K., and Graepel, T. The mechanics of n-player differentiable games. In *ICML*, 2018a.

Balduzzi, D., Tuyls, K., Perolat, J., and Graepel, T. Reevaluating Evaluation. In *NeurIPS*, 2018b.

Bansal, T., Pachocki, J., Sidor, S., Sutskever, I., and Mordatch, I. Emergent complexity via multi-agent competition. *ICLR*, 2018.

Banzhaf, W., Baumgaertner, B., Beslon, G., Doursat, R., Foster, J. A., McMullin, B., de Melo, V. V., Miconi, T., Spector, L., Stepney, S., , and White, R. Defining and Simulating Open-Ended Novelty: Requirements, Guidelines, and Challenges. *Theory in Biosciences*, 2016.

Borel, E. La théorie du jeu et les équations intégrales à noyau symétrique. *Comptes rendus de l'Académie des Sciences*, 1921.

Brant, J. C. and Stanley, K. O. Minimal Criterion Coevolution: A New Approach to Open-Ended Search. In *GECCO*, 2017.

Brown, G. Iterative Solutions of Games by Fictitious Play. In Koopmans, T. C. (ed.), *Activity Analysis of Production and Allocation*. Wiley, 1951.

Candogan, O., Menache, I., Ozdaglar, A., and Parrilo, P. A. Flows and Decompositions of Games: Harmonic and Potential Games. *Mathematics of Operations Research*, 36(3):474–503, 2011.

de Jong, E. D. A Monotonic Archive for Pareto-Coevolution. *Evolutionary Computation*, 15(1):61–93, 2001.

Elo, A. E. *The Rating of Chess players, Past and Present*. Ishi Press International, 1978.

Ficici, S. G. and Pollack, J. B. A Game-Theoretic Approach to the Simple Coevolutionary Algorithm. In *Parallel Problem Solving from Nature (PPSN)*, 2000.

Ficici, S. G. and Pollack, J. B. Pareto Optimality in Coevolutionary Learning. In *ECAL*, 2001.

Fonseca, C. and Fleming, P. Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In *GECCO*, 1993.

Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. Generative Adversarial Nets. In *NeurIPS*, 2014.

Hansen, T. D., Miltersen, P. B., and Sørensen, T. B. On Range of Skill. In *AAAI*, 2008.

Hart, S. Discrete Colonel Blotto and General Lotto games. *Int J Game Theory*, 36:441–460, 2008.

Heinrich, J., Lanctot, M., and Silver, D. Fictitious Self-Play in Extensive-Form Games. In *ICML*, 2015.

Hillis, W. D. Co-evolving parasites improve simulated evolution as an optimization procedure. *Physica D: Nonlinear Phenomena*, 42(1-3):228–234, 1990.

Jaderberg, M., Czarnecki, W. M., Dunning, I., Marris, L., Lever, G., Castaneda, A. G., Beattie, C., Rabinowitz, N. C., Morcos, A. S., Ruderman, A., Sonnerat, N., Green, T., Deason, L., Leibo, J. Z., Silver, D., Hassabis, D., Kavukcuoglu, K., and Graepel, T. Human-level performance in first-person multiplayer games with population-based deep reinforcement learning. *arXiv:1807.01281*, 2018.

Jiang, X., Lim, L.-H., Yao, Y., and Ye, Y. Statistical ranking and combinatorial Hodge theory. *Math. Program., Ser. B*, 127:203–244, 2011.

Lanctot, M., Zambaldi, V., Gruslys, A., Lazaridou, A., Tuyls, K., Perolat, J., Silver, D., and Graepel, T. A Unified Game-Theoretic Approach to Multiagent Reinforcement Learning. In *NeurIPS*, 2017.

Lehman, J. and Stanley, K. O. Exploiting Open-Endedness to Solve Problems Through the Search for Novelty. In *ALIFE*, 2008.

Leslie, D. and Collins, E. J. Generalised weakened fictitious play. *Games and Economic Behavior*, 56(2):285–298, 2006.

McMahan, H. B., Gordon, G., and Blum, A. Planning in the presence of cost functions controlled by an adversary. In *ICML*, 2003.

Miettinen, K. *Nonlinear Multiobjective Optimization*. Springer, 1998.

Monroy, G. A., Stanley, K. O., and Miikkulainen, R. Coevolution of neural networks using a layered Pareto archive. In *GECCO*, 2006.

Nemirovski, A. and Yudin, D. *Problem Complexity and Method Efficiency in Optimization*. Wiley-Interscience, 1983.

Nolfi, S. and Floreano, D. Coevolving predator and prey robots: Do "arms races" arise in artificial evolution? *Artificial Life*, 4(4):311–335, 1998.

Pathak, D., Agrawal, P., Efros, A. A., and Darrell, T. Curiosity-driven Exploration by Self-supervised Prediction. In *ICML*, 2017.

Popovici, E., Bucci, A., Wiegand, R. P., and Jong, E. D. D. Coevolutionary principles. In Rozenberg, G., Bck, T., and Kok, J. N. (eds.), *Handbook of Natural Computing*, pp. 987–1033. Springer, Berlin, Heidelberg, 2012.

Roberson, B. The Colonel Blotto game. *Economic Theory*, 29(1), 2006.

Rosin, C. D. New methods for competitive coevolution. *Evolutionary Computation*, 5(1):1–29, 1997.

Schmidt, M. D. and Lipson, H. Coevolution of Fitness Predictors. *IEEE Transactions on Evolutionary Computation*, 12(6):736–759, 2008.

Shoham, Y. and Leyton-Brown, K. *Multiagent Systems: Algorithmic, Game-Theoretic, and Logical Foundations*. Cambridge University Press, 2008.

Silver, D., Hubert, T., Schrittwieser, J., Antonoglou, I., Lai, M., Guez, A., Lanctot, M., Sifre, L., Kumaran, D., Graepel, T., Lillicrap, T., Simonyan, K., and Hassabis, D. A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play. *Science*, 362: 1140–1144, 2018.

Taylor, T., Bedau, M., Channon, A., Ackley, D., Banzhaf, W., Beslon, G., Dolson, E., Froese, T., Hickinbotham, S., Ikegami, T., McMullin, B., Packard, N., Rasmussen, S., Virgo, N., Agmon, E., McGregor, E. C. S., Ofria, C., Ropella, G., Spector, L., Stanley, K. O., Stanton, A., Timperley, C., Vostinar, A., and Wiser, M. Open-Ended Evolution: Perspectives from the OEE Workshop in York. *Artificial Life*, 22:408423, 2016.

Tesauro, G. Temporal difference learning and TD-Gammon. *Communications of the ACM*, 38(3):58–68, 1995.

Tukey, J. W. A problem of strategy. *Econometrica*, 17, 1949.

Wang, R., Lehman, J., Clune, J., and Stanley, K. O. Paired Open-Ended Trailblazer (POET): Endlessly Generating Increasingly Complex and Diverse Learning Environments and Their Solutions. In *arXiv:1901.01753*, 2019.

Wellman, M. P. Methods for empirical game-theoretic analysis. In *AAAI*, pp. 1552–1556, 2006.

Zinkevich, M., Bowling, M., and Burch, N. A New Algorithm for Generating Equilibria in Massive Zero-Sum Games. In *AAAI*, 2007.