Preprint – IJCAI 2025: This is the accepted version made available for conference attendees.
Do not cite. The final version will appear in the IJCAI 2025 proceedings.

Randomised Optimism via Competitive Co-Evolution for Matrix Games with Bandit Feedback

Shishen Lin
School of Computer Science, University of Birmingham, Birmingham, United Kingdom
sxl1242@student.bham.ac.uk

Abstract

Learning in games is a fundamental problem in machine learning and artificial intelligence, with numerous applications. This work investigates two-player zero-sum matrix games with an unknown payoff matrix and bandit feedback, where each player observes their actions and the corresponding noisy payoff. Prior studies have proposed algorithms for this setting, with demonstrating the effectiveness of deterministic optimism (e.g., UCB for matrix games) in achieving sublinear regret. However, the potential of randomised optimism in matrix games remains theoretically unexplored.

We propose Competitive Co-evolutionary Bandit Learning (COEBL), a novel algorithm that integrates evolutionary algorithms (EAs) into the bandit framework to implement randomised optimism through EA variation operators. We prove that COEBL achieves sublinear regret, matching the performance of deterministic optimism-based methods. To the best of our knowledge, this is the first theoretical regret analysis of an evolutionary bandit learning algorithm in matrix games.

Empirical evaluations on diverse matrix game benchmarks demonstrate that COEBL not only achieves sublinear regret but also consistently outperforms classical bandit algorithms, including EXP3, the variant EXP3-IX, and UCB. These results highlight the potential of evolutionary bandit learning, particularly the efficacy of randomised optimism via evolutionary algorithms in game-theoretic settings.

1 Introduction

1.1 Two-Player Zero-Sum Games

Learning in games is a fundamental problem in machine learning and artificial intelligence, with numerous applications [Silver *et al.*, 2016; Schrittwieser *et al.*, 2020]. Triggered by von Neumann's seminal work [Von Neumann, 1928; Von Neumann *et al.*, 1953], the maximin optimisation problem (i.e., $\max_{x\in\mathcal{X}} \min_{y\in\mathcal{Y}} g(x,y)$) has become a major research topic in machine learning and optimisation. In particular, two-player zero-sum games, represented by a payoff matrix $A \in \mathbb{R}^{m\times m}$, constitute a widely studied problem class in the machine learning and AI literature [Littman, 1994; Auger *et al.*, 2015; O'Donoghue *et al.*, 2021; Cai *et al.*, 2023]. The row player selects $i \in [m]$, the column player selects $j \in [m]$ and these choices, leading to a payoff A_{ij} (i.e. the row player receives the payoff A_{ij} and the column player receives the payoff $-A_{ij}$). The objective is to find the optimal mixed strategy, which is a probability distribution over actions for each player. Formally, we define our problem as follows: to find $x^*, y^* \in \Delta_m$, where Δ_m denotes the probability simplex of dimension $m-1$, satisfying

$$V_A^* := \max_{x\in\Delta_m} \min_{y\in\Delta_m} y^T Ax. \tag{1}$$

By von Neumann's minimax theorem [Von Neumann, 1928], $V_A^* = \min_{y\in\Delta_m} \max_{x\in\Delta_m} y^T Ax$. (x^*, y^*) solving for Eq.(1) is called a Nash equilibrium. V_A^* is the shared optimal quantity at the Nash equilibrium. In this paper, we call it the Nash equilibrium payoff.

Nash's Theorem, or von Neumann's minimax theorem, guarantees the existence of (x^*, y^*) for Eq.(1) [Von Neumann, 1928; Nash, 1950]. If the payoff matrix is given or known, then Eq.(1) can be reformulated as a linear programming problem, and it can be solved in polynomial time using algorithms including the ellipsoid method or interior point method [Bubeck and others, 2015; Maiti *et al.*, 2023]. Now, if the payoff matrix is unknown, let the row and column player play an iterative two-player zero-sum game. At each iteration, they select actions and observe the corresponding payoff entry from the matrix. Based on the observed rewards, both players update their strategies. This iterative setting is referred to as a repeated matrix game (or matrix games, for short). Our goal is to design algorithms that can perform competitively in such games. A standard measure of algorithmic performance is regret, which we define formally later. We are also interested in whether the algorithm can approximate the Nash equilibrium (x^*, y^*), measured using divergence metrics such as KL-divergence or total variation distance.

1.2 Evolutionary Reinforcement Learning

Evolutionary Algorithms (EAs) are randomised heuristics inspired by natural selection, designed to solve optimisation problems [Popovici *et al.*, 2012; Eiben and Smith, 2015].

EAs aim to find global optima with minimal knowledge about fitness functions, making them well-suited for some black-box or oracle settings compared to gradient-based methods [Eiben and Smith, 2015]. EAs are powerful and useful tools for discovering effective reinforcement learning policies because they can identify good representations, manage continuous action spaces, and handle partial observability [Whiteson, 2012]. Due to these strengths, evolutionary reinforcement learning (ERL) techniques have shown strong empirical success, and we refer readers to [Whiteson, 2012; Bai *et al.*, 2023; Li *et al.*, 2024a] for detailed reviews of ERL.

Coevolution, rooted in evolutionary biology, involves the simultaneous evolution of multiple interacting populations [Anderson and May, 1982]. Interactions can be cooperative (e.g., humans and gut bacteria) or competitive (e.g., predator-prey dynamics). These co-evolutionary dynamics have been studied and applied in ERL, demonstrating empirical effectiveness in many applications [Whiteson, 2012; Xue *et al.*, 2024; Li *et al.*, 2024a]. For example, co-evolutionary algorithms (CoEAs), a subset of EAs, have been applied in many black-box optimisation problems under various game-theoretic scenarios [Xue *et al.*, 2024; Gomes *et al.*, 2014; Hemberg *et al.*, 2021; Flores *et al.*, 2022; Fajardo *et al.*, 2023; Hevia Fajardo *et al.*, 2024; Benford and Lehre, 2024b].

Despite the practical success of evolutionary reinforcement learning in domains such as game playing, robotics, and optimisation [Moriarty *et al.*, 1999; Khadka and Tumer, 2018; Pourchot and Sigaud, 2019; HAO *et al.*, 2023; Li *et al.*, 2024b; Li *et al.*, 2024c], there is a lack of rigorous theoretical analysis [Li *et al.*, 2024a]. In particular, the theoretical foundations of coevolutionary learning in matrix games remain largely unexplored. As a starting point, in this work, we aim to bridge this gap by combining evolutionary heuristics with bandit learning and analysing their performance in matrix games from both theoretical and empirical perspectives.

1.3 Contributions

This paper introduces evolutionary algorithms for learning in matrix games with bandit feedback. To the best of our knowledge, this is the first work to provide a rigorous regret analysis of evolutionary reinforcement learning (i.e., COEBL) in matrix games with bandit feedback. Specifically, we show that randomised optimism implemented via evolutionary algorithms can achieve sublinear regret in this setting. Our empirical results demonstrate that COEBL outperforms existing bandit learning baselines for matrix games, including EXP3, UCB, and the EXP3-IX variant. These findings highlight the significant potential of evolutionary algorithms for bandit learning in game-theoretic environments and reveal the role of randomness in effective game-play. This work serves as a first step towards a rigorous theoretical understanding of evolutionary reinforcement learning in game-theoretic settings.

1.4 Related Works

Regret Analysis of Bandit Learning in Matrix Games

Theoretical analysis of bandit learning algorithms in matrix games has been extensively studied. Recent works, such as [Auger *et al.*, 2015; O'Donoghue *et al.*, 2021; Cai *et al.*, 2023], have studied classical bandit algorithms under settings where only rewards or payoffs are observed. In particular, O'Donoghue *et al.* [2021] conducted an in-depth regret analysis of the UCB algorithm, Thompson Sampling, and K-Learning, showing that these methods achieve sublinear regret in matrix games. Neu [2015] established a sublinear regret bound for EXP3-IX, which was subsequently extended to matrix games by Cai *et al.* [2023] through a new variant. Additionally, Auger *et al.* [2015] provided convergence analyses of bandit algorithms in sparse binary zero-sum games, while Cai *et al.* [2023] extended these results to uncoupled learning in two-player zero-sum Markov games. A more recent study by Li *et al.* [2024d] investigates adversarial regret for Optimistic Thompson Sampling, exploiting repeated-game structures and partial observability to anticipate opponent strategies.

In contrast, our work aligns with [O'Donoghue *et al.*, 2021] in focusing on Nash regret in two-player zero-sum matrix games with bandit feedback. While their approach highlights adversarial dynamics, ours demonstrates the potential of randomised optimism via evolutionary algorithms. We provide a novel regret analysis that complements stochastic optimism-based methods in achieving sublinear Nash regret. Moreover, the theoretical understanding of evolutionary bandit learning remains largely unexplored. This paper aims to fill that gap, marking a first step towards the rigorous study of evolutionary bandit learning in matrix games, an area that remains both promising and under-explored.

Runtime Analysis of Co-evolutionary Algorithms

Recent studies have conducted runtime analyses of cooperative and competitive co-evolutionary algorithms [Jansen and Wiegand, 2004; Lehre, 2022]. Here, runtime refers to the number of function evaluations required for the algorithm to find the Nash equilibrium. For a comprehensive overview of these contributions to competitive co-evolutionary algorithms, we refer readers to the recent papers [Lehre, 2022; Hevia Fajardo and Lehre, 2023; Fajardo *et al.*, 2023; Lehre and Lin, 2024b; Benford and Lehre, 2024b; Benford and Lehre, 2024a; Lehre and Lin, 2024a; Lehre and Lin, 2025]. Although we do not analyse the runtime of COEBL in this paper, an interesting future research direction is to explore how the runtime of COEBL could be analysed in the context of matrix games with bandit feedback. The idea of competitive coevolution in game-theoretic settings, as explored in the aforementioned works, serves as the foundation for our application of evolutionary methods to bandit learning algorithms.

2 Preliminaries

2.1 Notations

Given $n \in \mathbb{N}$, we write $[n] := \{1, 2, \cdots, n\}$. $\mathbb{F}_p$ denotes the finite field of p (prime number) elements. For example, $\mathbb{F}_3$ denotes the finite field of three elements, $\{-1, 0, 1\}$. We denote the row player by the x-player and the column player by the y-player. $f(n) \in O(h(n))$ if there exists some constant $c > 0$ such that $f(n) \leq ch(n)$. $f(n) \in \tilde{O}(h(n))$ if there exists some constant $k > 0$ such that $f \in O(h(n) \log^k(h(n)))$. We define the $(m-1)$-dimensional probability simplex as

$\Delta_m := \{z \in \mathbb{R}^m \mid \sum_{i=1}^m z_i = 1, z_i \geq 0\}$. In each round $t \in \mathbb{N}$, the row player chooses $i_t \in [m]$, and the column player chooses $j_t \in [m]$; and then r_t is the reward obtained by the row player. We define the corresponding filtration $\mathcal{F}_t$ prior to round t by $\mathcal{F}_t := (i_1, j_1, r_1, \ldots, i_{t-1}, j_{t-1}, r_{t-1})$. We denoted $\mathrm{E}_t(\cdot) := \mathrm{E}(\cdot \mid \mathcal{F}_t)$. For any real number x, we define $1 \vee x := \max(1, x)$. Given $x \in \{0,1\}^n$, $|x|_1 := \sum_{i=1}^n x_i$.

Definition 1. A random variable $X \in \mathbb{R}$ is σ^2-sub-Gaussian with variance proxy σ^2 if $\mathrm{E}(X) = 0$ and satisfies $\mathrm{E}(\exp(sX)) \leq \exp\left(\frac{\sigma^2 s^2}{2}\right)$, for all $s \in \mathbb{R}$.

2.2 Two-Player Zero-Sum Games and Nash Regret

A two-player game is characterised by the strategy spaces $\mathcal{X}$ and $\mathcal{Y}$, along with payoff functions $g_i : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$, where $i \in [2]$. Here, $g_i(x, y)$ denotes the payoff received by player i when player 1 plays strategy x and player 2 plays strategy y.

Definition 2. Given a two-player game with strategy spaces $\mathcal{X}$ and $\mathcal{Y}$, and a prime number $p \in \mathbb{N}$, let the payoff functions $g_1, g_2 : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ represent the payoffs for player 1 and player 2, respectively. The game is said to be *zero-sum* if the gain of one player is exactly the loss of the other, i.e., $g_1(x, y) + g_2(x, y) = 0$ for all $x \in \mathcal{X}$ and $y \in \mathcal{Y}$.

Many classical games where the outcomes are win, lose and draw, such as Rock-Paper-Scissors, Tic-Tac-Toe and Go, can be modelled as ternary zero-sum games,where $g(x, y) = 1$ denotes a win for player 1, $g(x, y) = -1$ a win for player 2, and $g(x, y) = 0$ a draw. In this paper, we mainly focus on ternary two-player zero-sum games. In matrix games, we evaluate performance using the *Nash regret*, defined as the cumulative difference between the Nash equilibrium payoff in Eq. (1) and the actual rewards obtained by the players [O'Donoghue *et al.*, 2021].

Definition 3 (Nash Regret)**.** Consider any matrix game with payoff matrix $A \in \mathbb{R}^{m\times m}$ and the reward for the row player choosing action $i_t \in [m]$ and the column player choosing action $j_t \in [m]$ is given by $r_t = A_{i_t j_t} + \eta_t$, where η_t is zero-mean noise, independent and identically distributed from a known distribution at iteration $t \in \mathbb{N}$. Given an algorithm ALG that maps the filtration $\mathcal{F}_t$ to a distribution over actions $x \in \Delta_m$, we define the Nash regret with respect to the Nash equilibrium payoff $V_A^* \in \mathbb{R}$ by

$$\mathcal{R}\left(A, \textsc{Alg}, T\right) := \mathrm{E}_{\eta,\textsc{Alg}}\left(\sum_{t=1}^{T} V_A^* - r_t\right).$$

Given any class of games $A \in \mathcal{A}$, for any $T \in \mathbb{N}$, we define

$$\textsc{WorstCaseRegret}\left(\mathcal{A}, \textsc{Alg}, T\right) := \max_{A\in\mathcal{A}} \mathcal{R}\left(A, \textsc{Alg}, T\right).$$

Given a fixed unknown payoff matrix A, the regret $\mathcal{R}\left(A, \textsc{Alg}, T\right)$ represents the expected cumulative difference between the Nash equilibrium payoff and the actual rewards obtained by player 1 using algorithm ALG over T iterations. The worst-case regret $\textsc{WorstCaseRegret}\left(\mathcal{A}, \textsc{Alg}, T\right)$ denotes the maximum regret of algorithm ALG across all possible payoff matrices within the class of games $\mathcal{A}$, thus capturing performance in the worst-case scenario.

Nash regret serves as a fundamental measure for evaluating an agent's performance against a best-response opponent, enabling fair comparison with prior work [O'Donoghue *et al.*, 2021]. It emphasises long-term convergence to equilibrium strategies, thereby reflecting robust and generalised behaviour in game-theoretic settings. Although it may be less informative for analysing intermediate behaviours [Li *et al.*, 2024d], it offers critical guarantees regarding the agent's capacity to adapt towards optimal and resilient strategies against adversarial best-response opponents.

3 Co-evolutionary Bandit Learning

3.1 Learning in Games and COEBL

The study of learning dynamics in games seeks to understand how players can adapt their strategies to approach the equilibrium when interacting with rational opponents [Fudenberg and Levine, 1998]. A commonly used metric for evaluating algorithmic performance in such settings is regret, as formally defined in Definition 3. Alternative evaluation criteria include convergence to Nash equilibrium, which may be measured using KL-divergence or total variation distance.

In this section, we only present the algorithm for the x-player, noting that the counterpart for the y-player is symmetric. The proposed method, COEBL (Co-Evolutionary Bandit Learning), leverages co-evolutionary approach in bandit feedback settings. We denote the empirical mean of the rewards sampled from entry A_{ij} by $\bar{A}_{ij}^t$, and the number of times up to round t that the row player has selected action i while the column player has selected action j, by $n_{ij}^t \in [t] \cup \{0\}$.

Algorithm 1 COEBL for matrix games

Require: $\text{Fitness}(x, B) := \min_{y\in\Delta_m} y^T B x$ where $B \in \mathbb{R}^{m\times m}$ and $x \in \Delta_m$.
Initialisation: $x_0, y_0 = (1/m, \ldots, 1/m)$ and $n_{ij}^0 = 0$ for all $i, j \in [m]$
for round $t = 1, 2, \ldots, T$ **do**
  for all $i, j \in [m]$ **do**
    Compute $\tilde{A}_{ij}^t = \text{Mutate}(\bar{A}_{ij}^t, 1/1 \vee n_{ij}^t)$
  end for
  Obtain $x' \in \arg\max_{x\in\Delta_m} \min_{y\in\Delta_m} y^T \tilde{A}^t x$
  if $\text{Fitness}(x', \tilde{A}^t) >$ Fitness $(x_{t-1}, \tilde{A}^t)$ **then**
    Update policy $x_t := x'$
  else
    Update policy $x_t := x_{t-1}$
  end if
  Update the query number of each entry in the payoff matrix n_{ij}^t for all $i, j \in [m]$
end for

The following mutation variant is considered in this paper.

$$\text{Mutate}(\bar{A}_{ij}^t, \frac{1}{1 \vee n_{ij}^t}) = \bar{A}_{ij}^t + \mathcal{N}\left(\sqrt{\frac{c\log(2T^2m^2)}{1 \vee n_{ij}^t + 1}}, \frac{1}{(1 \vee n_{ij}^t)^2}\right),$$

where $\mathcal{N}\left(\mu, \sigma^2\right)$ denotes a Gaussian random variable with mean μ and variance σ^2, and c is some constant with respect to T and m.

Evolutionary algorithms typically consist of two main components: variation operators and selection mechanisms. Variation operators generate new candidate policies from the current population, while the selection mechanism retains the most promising policies based on a predefined fitness function. In COEBL, we define the fitness function as $\text{Fitness}(x, B) := \min_{y \in \Delta_m} y^T B x$, which evaluates the performance of a policy x against the best response of the opponent, given the payoff matrix B. Initially, COEBL applies a Gaussian mutation operator to perturb the estimated payoff matrix $\tilde{A}^t$, and generates a mutated policy x' for the row player. As the estimated matrix $\tilde{A}$ is fully observable by the x-player, the optimal response x' in line 6 is computed by solving a linear programming problem [Bubeck and others, 2015; Maiti *et al.*, 2023]. Between lines 7 and 10, the new policy x' is evaluated using the fitness function and compared against the previous policy x_{t-1}. If x' strictly outperforms x_{t-1}, the policy is updated; otherwise, we retain x_{t-1} to avoid potential instability in the maximin solution.

The central idea behind COEBL is to adopt the principle of *optimism in the face of uncertainty* (OFU) to explore the action space and exploit the opponent's best response [Bubeck *et al.*, 2012; Lattimore and Szepesvári, 2020]. However, unlike traditional bandit algorithms such as those in the UCB family, COEBL implements *randomised optimism* via evolutionary algorithms. Through the variation operator, COEBL generates diverse estimated payoff matrices, which in turn lead to a broader range of diverse candidate policies. The selection mechanisms then guide the evolutionary process towards higher fitness.

Although COEBL shares similar priors as Thompson Sampling using Gaussian priors by [Agrawal and Goyal, 2017] and Optimism-then-NoRegret (OTN) framework [Li *et al.*, 2024d], it differs in two key ways. First, our Gaussian prior is defined via upper confidence bounds rather than empirical means alone, with a higher variance term $1/\left(1 + n_{ij}^t\right)$ compared with one by [Agrawal and Goyal, 2017] and a different scaling constant than used in OTN [Li *et al.*, 2024d]. Second, COEBL incorporates a selection mechanism, which implicitly adjusts the sampling distribution and is uncommon in Thompson Sampling. These two key changes are proven to be beneficial for adversarial matrix game settings in later sections.

While O'Donoghue *et al.* [2021] has demonstrated that deterministic optimism enables UCB to achieve sublinear regret and outperform classic EXP3, and other bandit baselines, we will show that randomised optimism (via evolution) also exhibits sublinear Nash regret. More importantly, we will demonstrate that randomised optimism in matrix games can be more effective and adaptive in preventing exploitation by the opponent than deterministic optimism, and thus outperforms existing bandit baselines. Specifically, it outperforms the current bandit baseline algorithms for matrix games, including EXP3, UCB and the EXP3-IX variant on the matrix game benchmarks considered in this paper.

3.2 Regret Analysis of COEBL

In this section, we conduct the regret analysis of COEBL in matrix games. Before our analysis, we need some technical lemmas. We defer these lemmas to the appendix[1].

We follow the setting in [O'Donoghue *et al.*, 2021] and consider the case where there is 1-sub-Gaussian noise when querying the payoff matrix. Assume that given $t \in \mathbb{N}$:

(A): The noise process η_t is 1-sub-Gaussian and the payoff matrix satisfies $A \in [0, 1]^{m \times m}$.

Lemma 1. *Suppose Assumption (A) holds with $T \geq 2m^2 \geq 2$ and $\delta := \left(1/2T^2m^2\right)^{c/8}$ where $c > 0$ is the mutation rate in* COEBL. *For each iteration $t \in \mathbb{N}$, given $\tilde{A}^t$ in Algorithm 1, we have:*

$$\Pr\left(A_{ij} - (\tilde{A}_t)_{ij} \leq 0\right) \geq 1 - \delta, \quad \textit{for all } i, j \in [m]. \quad (2)$$

Theorem 2 (Main Result)**.** *Consider any two-player zero-sum matrix game. Under Assumption (A) with $T \geq 2m^2 \geq 2$ and $\delta = \left(1/2T^2m^2\right)^{c/8}$, where $c > 0$ is the mutation rate in* COEBL, *the worst-case Nash regret of* COEBL *for $c \geq 8$ is bounded by $2\sqrt{2cTm^2 \log(2T^2m^2)}$, i.e., $\tilde{O}(\sqrt{m^2T})$.*

Sketch of Proof. Due to page limit, we defer the full proofs of Lemma 1 and Theorem 2 to the appendix and provide a simple proof sketch here. First, we bound the regret under the case where all the entries of the estimated payoff matrix are greater than those of the real, unknown payoff matrix (this event is denoted by E_t^c at iteration $t \in \mathbb{N}$). Secondly, we use the law of total probability to consider both cases: when all the entries of the estimated payoff matrix are greater than the real payoff matrix, and the converse (i.e., event E_t). We already have the upper bound for the first part; the second part can be trivially bounded by 1 in each iteration. Using Lemma 1, we can obtain the upper bound of probability of event E_t. Combining these bounds provides us with the upper bound for the regret of COEBL. □

Theorem 2 demonstrates that, under the worst-case scenario (assuming the best response of the opponent across all possible matrix game instances under Assumption (A)), COEBL achieves sublinear regret. Specifically, the regret of COEBL is bounded by $\tilde{O}(\sqrt{m^2T})$, matching the regret bound of UCB. This result suggests that deterministic optimism in the face of uncertainty, as highlighted in [O'Donoghue *et al.*, 2021], is not the sole determinant for achieving sublinear regret. In fact, it indicates that the mechanism of optimism: whether deterministic or stochastic is not necessarily critical to the asymptotic regret guarantees in adversarial settings.

However, a key distinction lies in the practical robustness of randomised (stochastic) optimism via evolution compared to deterministic optimism. As we demonstrate in later sections, randomised optimism offers greater adaptability and robustness in game-playing scenarios, enabling COEBL to outperform other algorithms in benchmark tasks.

[1] A full version of this paper can be found in https://arxiv.org/abs/2505.13562.

Intuitively, stochastic optimism facilitates a more balanced exploration-exploitation trade-off by incorporating upper-confidence-bound randomness, which helps prevent premature convergence to sub-optimal strategies and allows the algorithm to respond more flexibly to dynamic and adversarial environments.

This analysis highlights the potential advantages of stochastic methods in algorithm design for complex environments. Notably, the current analysis of COEBL assumes $c \geq 8$ due to technical constraints. We conjecture that the regret bound can be improved by considering smaller values of c, which may further enhance the practical performance of COEBL. Therefore, we recommend hyper-parameter tuning to optimise performance across various problem settings.

4 Empirical Results

In this section, we present empirical results comparing the discussed algorithms. We are interested in empirical regret in specific game instances, measured by cumulative (absolute) regret, i.e.,

$$\sum_{t=1}^{T} |V_A^* - r_t| \quad \text{and} \quad \sum_{t=1}^{T} V_A^* - r_t \tag{3}$$

where r_t is the obtained reward at round t. We focus on two scenarios, including self-play and ALG 1-vs-ALG 2. In the self-play scenario, both row and column players use the same algorithm with the same information. We use the absolute regret (the first metric) to measure the performance of the algorithms in this case. The ALG 1-vs-ALG 2 is a generalisation of the self-play scenario. We use the second metric in Eq. 3 to measure the performance of the algorithms. The ALG 1-vs-ALG 2 means the row player uses ALG 1, and the column player uses ALG 2 with the same information. As in the setting of [O'Donoghue *et al.*, 2021], the plots below show the regret (not absolute regret) from the maximiser's (ALG 1) perspective. A positive regret value means that the minimiser (ALG 2) is, on average winning and *vice versa*. This allows us to compare our algorithms directly.

Moreover, to measure how far the players are from the Nash equilibrium, we use the KL-divergence between the policies of both players and the Nash equilibrium or the total variation distance (for the case where the KL-divergence is not well-defined), i.e., $\mathrm{KL}(x_t, x^*) + \mathrm{KL}(y_t, y^*)$ and $\mathrm{TV}(x_t, x^*) + \mathrm{TV}(y_t, y^*)$, where

$$\mathrm{KL}(a, b) := \sum_i a(i) \ln\left(\frac{a(i)}{b(i)}\right)$$

$$\mathrm{TV}(a, b) := \frac{1}{2} \sum_i |a(i) - b(i)|$$

for any $a, b \in \Delta_m$ and (x^*, y^*) is the Nash equilibrium of A.

Parameter Settings

Given K is the number of actions for each player and T is the time horizon, for EXP3, we use the exploration rate $\gamma_t = \min\{\sqrt{K \log K/t}, 1\}$ and learning rate $\eta_t = \sqrt{2 \log K/tK}$ as suggested in [O'Donoghue *et al.*, 2021]. For the variant of EXP3-IX, we use the same settings $\eta_t = t^{-k_\eta}$, $\beta_t = t^{-k_\beta}$, $\varepsilon_t = t^{-k_\varepsilon}$ where $k_\eta = 5/8$, $k_\beta = 3/8$, $k_\varepsilon = 1/8$ as suggested in [Cai *et al.*, 2023]. For COEBL, we set the mutation rate $c = 2$ for the RPS game and $c = 8$ for the rest of the games. There is no hyper-parameter needed for UCB. For the observed reward, we consider standard Gaussian noise with zero mean and unit variance, i.e. $r_t = A_{i_t, j_t} + \eta_t$ where $\eta_t \sim \mathcal{N}(0, 1)$. We compute the empirical mean of the regrets and the KL-divergence (or total variation distance), and present the 95% confidence intervals in the plots. We run 50 independent simulations (up to 3000 iterations) for each configuration (over 50 seeds).

4.1 Rock-Paper-Scissors Game

We consider the classic matrix game benchmark: Rock-Paper-Scissors games [Littman, 1994; O'Donoghue *et al.*, 2021], and its payoff matrix is defined as follows.

	R	P	S
R	0	1	-1
P	-1	0	1
S	1	-1	0

Table 1: The payoff matrix of RPS game. R denotes rock, P denotes paper, and S denotes scissors.

It is well known that $x^*, y^* = (1/3, 1/3, 1/3)$ is the unique mixed Nash equilibrium of the RPS game for both players. We conduct experiments using EXP3, EXP3-IX, UCB and compare them with our proposed Algorithm 1 (i.e. COEBL) on the classic matrix game benchmark: the RPS game.

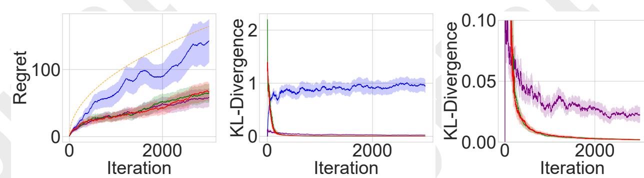
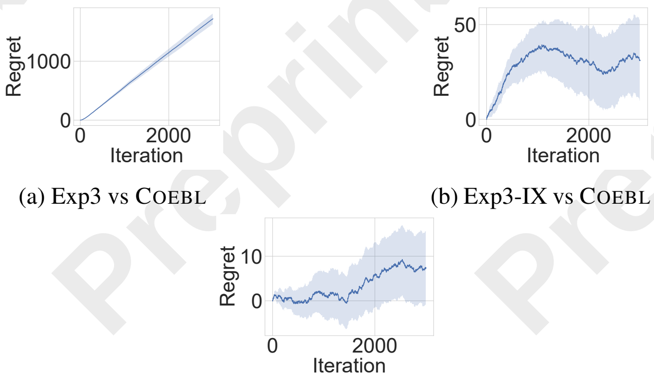


Figure 1: Regret and KL-divergence for Self-Plays on RPS games (Orange: Theoretical Bound $\sqrt{K^2 T}$, Red: COEBL, Green: UCB, Purple: EXP3-IX and Blue: EXP3)

In Figure 1, we present the self-play results of each algorithm. We can observe that COEBL also exhibits sublinear regret in the RPS game, similar to other bandit baselines, and matches our theoretical bound. In terms of the KL-divergence, EXP3, as reported in [O'Donoghue *et al.*, 2021; Cai *et al.*, 2023], diverges from the Nash equilibrium. By zooming in on the KL-divergence plot, we can observe that COEBL and UCB converges to the Nash equilibrium faster than the other algorithms; especially, EXP3-IX has a much slower convergence rate.

Next, we compare the performance of the algorithms by examining their regret bounds and KL-divergence from the Nash equilibrium when algorithms compete with each other using the same information. In Figure 2, we can clearly observe that COEBL outperforms the EXP3 family, including EXP3 and EXP3-IX, in terms of regret. On average, COEBL has a smaller advantage over UCB in terms of regret, since the empirical mean of regret is above 5 but below 10 after iteration 2000.



(a) Exp3 vs COEBL (b) Exp3-IX vs COEBL (c) UCB vs COEBL

Figure 2: Regret for ALG 1-vs-ALG 2 on RPS games

The RPS game with a small number of actions is relative simple for these algorithms to play. Moreover, although COEBL completely outperforms the EXP3 family, it does not have an overwhelming advantage over the UCB. How do these algorithms behave on more complex games with exponentially many actions? Can COEBL still take over the game? Next, we answer these questions by considering DIAGONAL and BIGGERNUMBER games.

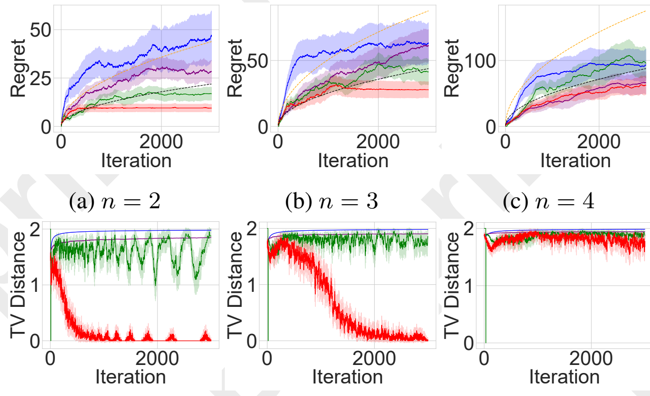
4.2 DIAGONAL Game

DIAGONAL is a pseudo-Boolean maximin-benchmark on which Lehre and Lin [2024b] conducted runtime analysis of coevolutionary algorithms. Both players have an exponential number (i.e. 2^n) of pure strategies. To distinguish between pure strategies that consist of the same number of 1, we modify the original DIAGONAL by introducing a 'draw' outcome. For $\mathcal{U} = \{0,1\}^n$ and $\mathcal{V} = \{0,1\}^n$, the payoff function DIAGONAL : $\mathcal{U} \times \mathcal{V} \to \{0,1\}$ is defined by

$$\text{DIAGONAL}(u,v) := \begin{cases} 1 & |v|_1 < |u|_1 \\ 0 & |v|_1 = |u|_1 \\ -1 & \text{otherwise} \end{cases}.$$

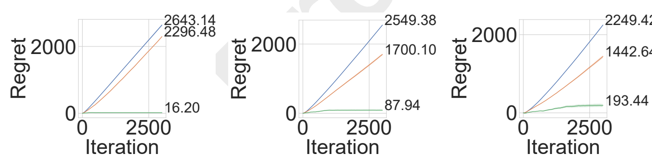
As shown by [Lehre and Lin, 2024b], this game (we provide a simple example in the appendix) exhibits a unique pure Nash equilibrium where both players choose 1^n. This corresponds to the mixed Nash equilibrium where $x^* = (0, \cdots, 1)$ and $y^* = (0, \cdots, 1)$. We conduct experiments using EXP3 to and compare them with our proposed Algorithm 1 (i.e. COEBL) on another matrix game benchmark: the DIAGONAL game. We set the mutation constant $c = 8$ for COEBL and consider $n = 2, 3, 4, 5, 6, 7$ in the experiments.

In Figures 3 and 7[2], we present the self-play results of each algorithm on DIAGONAL game for various values of n. Our results show that COEBL consistently exhibits sublinear regret in the DIAGONAL game, aligning with our theoretical bounds and similar to other bandit algorithms. As n increases, the regret of the baseline algorithms grows as expected. COEBL remains more adaptive and robust in more challenging games, maintaining sublinear regret beneath the theoretical bound ($0.1\sqrt{K^2T}$), as indicated by the black dotted line. We also observe that the regrets of all algorithms increases as n grows, which is expected due to the exponential increase in the number of pure strategies and the corresponding complexity of the game. In terms of convergence measured by TV-distance, COEBL converges to the Nash equilibrium for $n = 2, 3$, while the baseline algorithms do not converge. However, for $n \geq 4$, as the number of strategies grows exponentially, COEBL also struggles to converge to the Nash equilibrium. In Figures 4 and 8[3], we present the regrets for ALG 1-vs-ALG 2 on DIAGONAL. The empirical regrets across all algorithms exceed 16.2, with a maximum of 389.8 for $n = 6$, indicating that the minimiser is dominant. In other words, COEBL outperforms the other bandit algorithms across all values of n, from 2 to 7.



(a) $n = 2$ (b) $n = 3$ (c) $n = 4$
(d) $n = 2$ (e) $n = 3$ (f) $n = 4$

Figure 3: Regret and TV Distance for Self-Plays on DIAGONAL (Orange: Theoretical Bound $0.2\sqrt{K^2T}$, Black: Theoretical Bound: $0.1\sqrt{K^2T}$, Red: COEBL, Green: UCB, Purple: EXP3-IX and Blue: EXP3)



(a) $n = 2$ (b) $n = 3$ (c) $n = 4$

Figure 4: Regret for ALG 1-vs-ALG 2 on DIAGONAL (Blue: EXP3-vs -COEBL, Orange: EXP3-IX-vs-COEBL, Green: UCB-vs-COEBL)

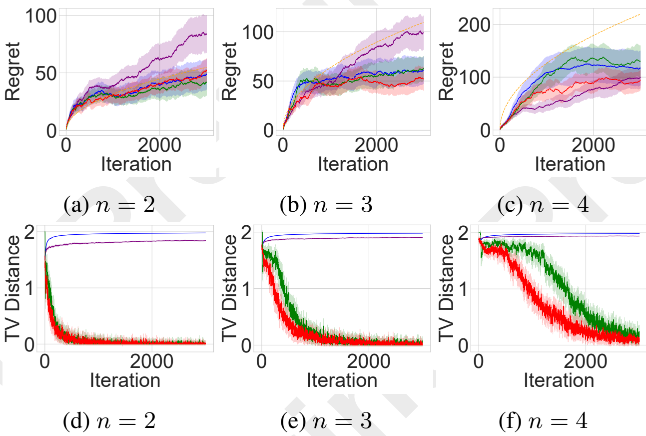
4.3 BIGGERNUMBER Game

BIGGERNUMBER is another challenging two-player zero-sum game proposed and analysed by [Zhang and Sandholm, 2024]. In this game, each player aims to select a number that is larger than their opponent's. The players' action space is

[2] Figure 7 is in the appendix.

[3] Figure 8 is in the appendix.

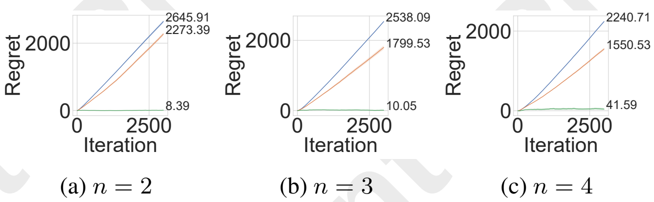
$\mathcal{X} = \{0,1\}^n$, representing binary bitstrings of length n corresponding to natural numbers in the range $[0, 2^n - 1]$. A formal definition and the complete results can be found in the appendix. We present part of the results here.



(a) $n = 2$ (b) $n = 3$ (c) $n = 4$

(d) $n = 2$ (e) $n = 3$ (f) $n = 4$

Figure 5: Regret and TV Distance for Self-Plays on BIGGERNUMBER (Orange: Theoretical Bound $0.25\sqrt{K^2T}$, Red: COEBL, Green: UCB, Purple: EXP3-IX and Blue: EXP3)



(a) $n = 2$ (b) $n = 3$ (c) $n = 4$

Figure 6: Regret for ALG 1-vs-ALG 2 on BIGGERNUMBER (Blue: EXP3-vs -COEBL, Orange: EXP3-IX-vs-COEBL, Green: UCB-vs-COEBL)

In summary, we conducted extensive experiments on three matrix games: the RPS game, the DIAGONAL game, and the BIGGERNUMBER game. In terms of regret performance, COEBL in self-play aligns with our theoretical bounds. Moreover, COEBL consistently outperforms other bandit baselines when competing across various matrix game benchmarks, as shown in Figures 4 and 6. COEBL matches the performance of UCB and converges more quickly than EXP3-IX in the RPS game. COEBL converges to the Nash equilibrium for $n = 2, 3$ and for $n = 2, 3, 4$, respectively, while the other baselines do not converge, as shown in Figures 3 and 5. Therefore, we conclude that COEBL is a promising algorithm for matrix games, demonstrating sublinear regret, outperforming other bandit baselines, and achieving convergence to the Nash equilibrium in several matrix game instances. However, as the number of strategies grows exponentially, COEBL, like other algorithms, fails to converge to the Nash equilibrium. This observation points out the current limitation of existing algorithms in exponentially large matrix games, and it will be an exciting path for future research.

5 Conclusion and Discussion

This paper addresses the unsolved problem of learning in unknown two-player zero-sum matrix games with bandit feedback, proposing a novel algorithm, COEBL, which integrates evolutionary algorithms with bandit learning. To the best of our knowledge, this is the first work that combines evolutionary algorithms and bandit learning for matrix games and provides the first regret analysis of evolutionary bandit learning (EBL) algorithms in this context. This paper demonstrates that randomised or stochastic optimism, particularly through evolutionary algorithms, can also enjoy a sublinear regret in matrix games, offering a more robust and adaptive solution compared to traditional methods in practice.

Theoretically, we prove that COEBL exhibits sublinear regret in matrix games, extending the rigorous understanding of evolutionary approaches in bandit learning. Practically, we show through extensive experiments on various matrix games, including RPS, DIAGONAL, and BIGGERNUMBER that COEBL outperforms existing bandit baselines, offering practitioners a new tool (randomised optimism via evolution) for handling matrix games playing with bandit feedback.

Despite these promising results, our work has some limitations. Theoretically, we only consider two-player zero-sum games, which is consistent with prior studies such as [O'Donoghue *et al.*, 2021; Cai *et al.*, 2023]. Extending COEBL to general-sum games with more players or to Markov games represents an exciting and challenging avenue for future research. More technically, we conjecture whether Theorem 2 could also hold for smaller value of $c < 8$ with certain threshold. Additionally, our analysis assumes sub-Gaussian noise. Investigating the algorithm's performance under different noise distributions, such as sub-exponential noise, could yield further insights. From an experimental perspective, testing on more diverse problem instances would strengthen the current empirical analysis.

Future work could focus on both theoretical and practical extensions of evolutionary bandit learning. From a theoretical perspective, it would be worthwhile to explore how COEBL or other evolutionary bandit learning algorithms can be adapted to more complex game structures, such as multi-player or general-sum games. On the practical side, improving COEBL by incorporating more sophisticated mutation operators, additional crossover operator, non-elitist selection mechanisms, or population-based evolutionary algorithms could enhance its performance in more complex settings. EBL could be a suitable class of algorithms, serving as a starting point for more general evolutionary reinforcement learning algorithms.

Ethical Statement

There are no ethical issues.

Acknowledgements

We would like to thank Professor Per Kristian Lehre, Dr Alistair Benford and Dr Mario Alejandro Hevia Fajardo for the fruitful discussion and comments on an earlier draft of this paper and also thank the anonymous reviewers for their helpful reviews. Shishen is supported by International PhD Studentship at School of Computer Science, the University of

Birmingham. The computations were performed using the University of Birmingham's BlueBEAR high performance computing (HPC) service.

References

[Agrawal and Goyal, 2017] Shipra Agrawal and Navin Goyal. Near-Optimal Regret Bounds for Thompson Sampling. *Journal of the ACM (JACM)*, 64(5):1–24, 2017.

[Anderson and May, 1982] Roy M Anderson and Robert M May. Coevolution of hosts and parasites. *Parasitology*, 85(2):411–426, 1982.

[Auger *et al.*, 2015] David Auger, Jianlin Liu, Sylkvie Ruette, David Saint-Pierre, and Oliver Teytaud. Sparse binary zero-sum games. In *Proceedings of the Sixth Asian Conference on Machine Learning*, Proceedings of Machine Learning Research, pages 173–188, Nha Trang City, Vietnam, 26–28 Nov 2015. PMLR.

[Bai *et al.*, 2023] Hui Bai, Ran Cheng, and Yaochu Jin. Evolutionary Reinforcement Learning: A Survey. *Intelligent Computing*, 2:0025, 2023.

[Benford and Lehre, 2024a] Alistair Benford and Per Kristian Lehre. Runtime Analysis of a Coevolutionary Algorithm on Impartial Combinatorial Games. *arXiv preprint arXiv:2409.04177*, 2024.

[Benford and Lehre, 2024b] Alistair Benford and Per Kristian Lehre. Runtime Analysis of Coevolutionary Algorithms on a Class of Symmetric Zero-Sum Games. In *GECCO'24: Genetic and Evolutionary Computation Conference*. Association for Computing Machinery (ACM), 2024.

[Bubeck and others, 2015] Sébastien Bubeck et al. Convex Optimization: Algorithms and Complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

[Bubeck *et al.*, 2012] Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret Analysis of Stochastic and Nonstochastic Multi-Armed Bandit Problems. *Foundations and Trends® in Machine Learning*, 5(1):1–122, 2012.

[Cai *et al.*, 2023] Yang Cai, Haipeng Luo, Chen-Yu Wei, and Weiqiang Zheng. Uncoupled and Convergent Learning in Two-Player Zero-Sum Markov Games with Bandit Feedback. In *Thirty-seventh Conference on Neural Information Processing Systems*, volume 36, 2023.

[Eiben and Smith, 2015] A.E. Eiben and J.E. Smith. *Introduction to Evolutionary Computing*. Natural Computing Series. Springer Berlin Heidelberg, Berlin, Heidelberg, 2015.

[Fajardo *et al.*, 2023] Mario Alejandro Hevia Fajardo, Per Kristian Lehre, and Shishen Lin. Runtime analysis of a co-evolutionary algorithm: Overcoming negative drift in maximin-optimisation. In *Proceedings of the 17th ACM/SIGEVO Conference on Foundations of Genetic Algorithms*, FOGA '23, page 73–83, New York, NY, USA, 2023. Association for Computing Machinery.

[Flores *et al.*, 2022] Diana Flores, Erik Hemberg, Jamal Toutouh, and Una-May O'Reily. Coevolutionary generative adversarial networks for medical image augumentation at scale. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 367–376, 2022.

[Fudenberg and Levine, 1998] Drew Fudenberg and David K Levine. *The Theory of Learning in Games*, volume 2. MIT press, 1998.

[Gomes *et al.*, 2014] Jorge Gomes, Pedro Mariano, and Anders Lyhne Christensen. Novelty Search in Competitive Coevolution. In *Parallel Problem Solving from Nature–PPSN XIII: 13th International Conference, Ljubljana, Slovenia, September 13-17, 2014. Proceedings 13*, pages 233–242. Springer, 2014.

[HAO *et al.*, 2023] Jianye HAO, Pengyi Li, Hongyao Tang, YAN ZHENG, Xian Fu, and Zhaopeng Meng. ERL-re^2: Efficient evolutionary reinforcement learning with shared state representation and individual policy representation. In *The Eleventh International Conference on Learning Representations*, 2023.

[Hemberg *et al.*, 2021] Erik Hemberg, Jamal Toutouh, Abdullah Al-Dujaili, Tom Schmiedlechner, and Una-May O'Reilly. Spatial Coevolution for Generative Adversarial Network Training. *ACM Transactions on Evolutionary Learning and Optimization*, 1(2):1–28, 2021.

[Hevia Fajardo and Lehre, 2023] Mario Alejandro Hevia Fajardo and Per Kristian Lehre. How Fitness Aggregation Methods Affect the Performance of Competitive Coeas on Bilinear Problems. In *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 1593–1601, 2023.

[Hevia Fajardo *et al.*, 2024] Mario Hevia Fajardo, Per Kristian Lehre, Jamal Toutouh, Erik Hemberg, and Una-May O'Reilly. Analysis of a Pairwise Dominance Coevolutionary Algorithm with Spatial Topology. In *Genetic Programming Theory and Practice XX*, pages 19–44. Springer, 2024.

[Jansen and Wiegand, 2004] Thomas Jansen and R. Paul Wiegand. The Cooperative Coevolutionary (1+1) EA. *Evol. Comput.*, 12(4):405–434, dec 2004.

[Khadka and Tumer, 2018] Shauharda Khadka and Kagan Tumer. Evolution-Guided Policy Gradient in Reinforcement Learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[Lattimore and Szepesvári, 2020] Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. Cambridge University Press, 2020.

[Lehre and Lin, 2024a] Per Kristian Lehre and Shishen Lin. No free lunch theorem and black-box complexity analysis for adversarial optimisation. In *Proceedings of the 38th Annual Conference on Neural Information Processing Systems (NeurIPS)*, Vancouver, Canada, 2024.

[Lehre and Lin, 2024b] Per Kristian Lehre and Shishen Lin. Overcoming Binary Adversarial Optimisation with Competitive Coevolution. In *Proceedings of the 18th Interna-*

tional Conference on Parallel Problem Solving From Nature PPSN, page 18, 2024.

[Lehre and Lin, 2025] Per Kristian Lehre and Shishen Lin. Towards runtime analysis of population-based co-evolutionary algorithms on sparse binary zero-sum game. In *Proceedings of the 39th AAAI Conference on Artificial Intelligence (AAAI)*, Philadelphia, Pennsylvania, USA, 2025.

[Lehre, 2022] Per Kristian Lehre. Runtime Analysis of Competitive co-Evolutionary Algorithms for Maximin Optimisation of a Bilinear Function. In *Proceedings of the Genetic and Evolutionary Computation Conference*, GECCO '22, page 1408–1416, 2022.

[Li *et al.*, 2024a] Pengyi Li, Jianye Hao, Hongyao Tang, Xian Fu, Yan Zhen, and Ke Tang. Bridging Evolutionary Algorithms and Reinforcement Learning: A Comprehensive Survey on Hybrid Algorithms. *IEEE Transactions on Evolutionary Computation*, 2024.

[Li *et al.*, 2024b] Pengyi Li, Jianye Hao, Hongyao Tang, Yan Zheng, and Fazl Barez. Value-Evolutionary-Based Reinforcement Learning. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 27875–27889. PMLR, 21–27 Jul 2024.

[Li *et al.*, 2024c] Pengyi Li, Yan Zheng, Hongyao Tang, Xian Fu, and Jianye Hao. EvoRainbow: Combining Improvements in Evolutionary Reinforcement Learning for Policy Search. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 29427–29447. PMLR, 21–27 Jul 2024.

[Li *et al.*, 2024d] Yingru Li, Liangqi Liu, Wenqiang Pu, Hao Liang, and Zhi-Quan Luo. Optimistic Thompson Sampling for No-Regret Learning in Unknown Games. *arXiv preprint arXiv:2402.09456*, 2024.

[Littman, 1994] Michael L Littman. Markov Games as a Framework for Multi-Agent Reinforcement Learning. In *Machine learning proceedings 1994*, pages 157–163. Elsevier, 1994.

[Maiti *et al.*, 2023] Arnab Maiti, Ross Boczar, Kevin Jamieson, and Lillian J Ratliff. Query-Efficient Algorithms to Find the Unique Nash Equilibrium in a Two-Player Zero-Sum Matrix Game. *arXiv preprint arXiv:2310.16236*, 2023.

[Moriarty *et al.*, 1999] David E Moriarty, Alan C Schultz, and John J Grefenstette. Evolutionary Algorithms for Reinforcement Learning. *Journal of Artificial Intelligence Research*, 11:241–276, 1999.

[Nash, 1950] John F Jr Nash. Equilibrium Points in n-Person Games. *Proceedings of the national academy of sciences*, 36(1):48–49, 1950.

[Neu, 2015] Gergely Neu. Explore No More: Improved High-Probability Regret Bounds for Non-Stochastic Bandits. *Advances in Neural Information Processing Systems*, 28, 2015.

[O'Donoghue *et al.*, 2021] Brendan O'Donoghue, Tor Lattimore, and Ian Osband. Matrix games with bandit feedback. In *Uncertainty in Artificial Intelligence*, pages 279–289. PMLR, 2021.

[Popovici *et al.*, 2012] Elena Popovici, Anthony Bucci, R. Paul Wiegand, and Edwin D. De Jong. Coevolutionary Principles. In Grzegorz Rozenberg, Thomas Bäck, and Joost N. Kok, editors, *Handbook of Natural Computing*, pages 987–1033. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[Pourchot and Sigaud, 2019] Pourchot and Sigaud. CEM-RL: Combining Evolutionary and Gradient-Based Methods for Policy Search. In *International Conference on Learning Representations*, 2019.

[Schrittwieser *et al.*, 2020] Julian Schrittwieser, Ioannis Antonoglou, Thomas Hubert, Karen Simonyan, Laurent Sifre, Simon Schmitt, Arthur Guez, Edward Lockhart, Demis Hassabis, Thore Graepel, et al. Mastering Atari, Go, Chess and Shogi by Planning with a Learned Model. *Nature*, 588(7839):604–609, 2020.

[Silver *et al.*, 2016] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature*, 529(7587):484–489, 2016.

[Von Neumann *et al.*, 1953] John Von Neumann, Oskar Morgenstern, et al. Theory of Games and Economic Behavior. *Princeton*, 1953.

[Von Neumann, 1928] John Von Neumann. Zur theorie der gesellschaftsspiele. *Mathematische annalen*, 100(1):295, 1928.

[Whiteson, 2012] Shimon Whiteson. Evolutionary Computation for Reinforcement Learning. *Reinforcement Learning: State-of-the-art*, pages 325–355, 2012.

[Xue *et al.*, 2024] Ke Xue, Ren-Jian Wang, Pengyi Li, Dong Li, HAO Jianye, and Chao Qian. Sample-Efficient Quality-Diversity by Cooperative Coevolution. In *The Twelfth International Conference on Learning Representations*, 2024.

[Zhang and Sandholm, 2024] Brian Hu Zhang and Tuomas Sandholm. Exponential Lower Bounds on the Double Oracle Algorithm in Zero-Sum Games. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 3032–3039. International Joint Conferences on Artificial Intelligence Organization, 8 2024.