Bridging the Gap Between Nash Equilibria and Coarse-Correlated Equilibria

Alan Xu

Professor Manolis Zampetakis
Alkis Kalavasis
Computer Science and Economics
Yale University
Spring 2024

Abstract

With the growing importance of computational methods in game theory, much recent work has been devoted to studying the computational complexity of various equilibrium concepts, with particular focus on the standard notions of Nash equilibrium and coarse-correlated equilibrium (CCE). In particular, it has been shown that finding Nash equilibria is computationally hard while CCEs can be efficiently computed. However, there exists a wide gap between these two equilibrium concepts, and the computational complexity of finding equilibria weaker than Nash equilibria but stronger than CCEs is still poorly understood.

In this paper, we investigate a class of equilibria called rank-k CCEs that lie in between Nash equilibria and CCEs. We first provide a graphical characterization of rank-2 CCEs in Rock Paper Scissors and show that the region is non-convex. Furthermore, we show that the unique Nash equilibrium occurs precisely at a singular point in the region of rank-2 CCEs. We then examine the close relationship between no-regret online learning algorithms and rank-k CCEs. We present the results of our experiment tracking the number of periods online learning takes to converge to a CCE as the number of strategies grows and provide evidence to support the conjecture that the $T = O\left(\frac{\log m}{\epsilon^2}\right)$ upper bound is in fact tight. This result also suggests that finding rank-$O(1)$ CCEs is likely computationally hard.

Finally, we present our initial attempt at proving that finding rank-2 CCEs is PPAD-hard. To that end, we try to follow the methods and ideas presented by Constantinos Daskalakis in his proof showing that finding Nash equilibria is PPAD-hard [Das08]. However, rank-2 CCEs introduce a series of complications that make it difficult to replicate the proof, and we conclude with a discussion of these challenges.

Contents

1 Introduction

1.1 Background

Ever since John Nash proved in 1951 that there exists a Nash equilibrium in *every* game [Nas51], game theorists have meticulously studied this powerful solution concept in a wide variety of contexts because it seems to provide a rigorous method for analyzing and predicting behavior when individuals interact with each other in strategic environments. However, despite this existence guarantee, there is currently no known polynomial-time algorithm to compute Nash equilibria in general games. In fact, Constantinos Daskalakis showed in 2008 that the problem of finding Nash equilibria in general games is PPAD-complete [Das08], and while unproven, it is widely believed that PPAD $\neq$ P. Therefore, it is highly unlikely that Nash equilibria can be efficiently computed, thereby dealing a heavy blow to the hopes of using Nash equilibria as a computationally feasible solution concept.

Fortunately, a different notion of equilibrium called (coarse)-correlated equilibrium (CCE) *can* be computed efficiently using algorithms like no-regret online learning [FV97; HM00; Ana+24]. CCEs are weaker than Nash equilibria because the requirement that every player choose their strategy independently is relaxed. Instead, all players collectively 'agree' on a joint probability distribution over all possible outcomes. Unfortunately, this level of communication between players is often infeasible in many real-world scenarios.

Given these results, it seems natural to wonder whether there might exist an efficient algorithm to compute some notion of equilibrium that is weaker than Nash equilibrium but stronger than CCE. In this paper, we begin the process of exploring such a class of equilibria called rank-k coarse-correlated equilibria.

We start by focusing primarily on rank-2 CCEs because they lie only one step above Nash equilibria, which are equivalent to rank-1 CCEs. First, we try to gain an initial understanding of rank-2 CCEs by analyzing them in simple games like Rock Paper Scissors.

Then, we explore the close relationship between rank-k CCEs and no-regret online learning algorithms and provide evidence to support the conjecture that computing rank-2 CCEs, like computing Nash equilibria, is likely PPAD-hard. Finally, we present our initial approach toward attempting to prove that computing rank-2 CCEs is PPAD-hard before concluding with a discussion of challenges encountered and potential directions for future work.

1.2 Definitions

Before we begin our exploration of rank-2 coarse-correlated equilibria, we must first establish a few foundational definitions that will be important to our discussion:

Definition 1.1: (adapted from [Das08]) a **normal-form game** has $n \geq 2$ players where each player i has a finite set of available pure strategies S_i, and the set of all pure strategy profiles is $S = S_1 \times S_2 \times ... \times S_n$. For each $s = (s_i, s_{-i}) \in S$, let s_i denote player i's strategy, s_{-i} denote the other players' strategies, and $u_i(s)$ denote the reward/payoff/utility that player i receives (the set $\{u_i(s)\}_{s \in S}$ is referred to as player i's reward matrix). Finally, for notational convenience, denote $[t] = \{1, 2, ..., t\}$ for all $t \in \mathbb{N}$.

In any particular instance of the game, players play according to a probability distribution σ over S that is defined by a **probability matrix** $\{\sigma_s\}_{s \in S}$ where σ_s is the probability that players play the pure strategy profile $s \in S$ in this particular instance of the game. Each player's **expected reward** in this instance is given by $r_i = \sum\limits_{s \in S} u_i(s) \cdot \sigma_s$.

Definition 1.2: a **mixed strategy** for player i is a probability distribution p_i over player i's pure strategies S_i, where $p_i[j]$ denotes the probability that player i plays strategy $j \in S_i$. If every player i independently plays a mixed strategy p_i, then σ is a product distribution $\bigotimes_{i=1}^n p_i$ such that for every $s = (s_1, s_2, ..., s_n) \in S$, $\sigma_s = \prod_{i=1}^n p_i[s_i]$. An **$\epsilon$-approximate Nash equilibrium** (where a 0-approximate Nash equilibrium is simply a Nash equilibrium) is a product distribution $\sigma^* = \bigotimes_{i=1}^n p_i^*$ over S such that $\forall i \in [n]$:

$$\sum_{s \in S} [u_i(s_i, s_{-i}) \cdot \sigma_s^*] \geq \sum_{s \in S} [u_i(s_i', s_{-i}) \cdot \sigma_s^*] - \epsilon \quad \forall s_i' \in S_i \tag{1.1}$$

That is, σ^* maximizes player i's expected reward over all of i's mixed strategies, for every $i \in [n]$, so no player has any incentive to unilaterally deviate from the equilibrium.

Definition 1.3: an **ϵ-coarse-correlated equilibrium** (ϵ-CCE) is any probability distribution σ over S such that inequality (1.1) holds $\forall i \in [n]$. The only difference between Nash equilibria and CCEs is that Nash equilibria enforce the requirement that σ^* is a product distribution while CCEs can be any arbitrary probability distribution over S. Thus, we can infer from these definitions that all Nash equilibria are CCEs.

In order to define rank-k CCEs, it is helpful to first examine the probability matrices for Nash equilibria and CCEs in two-player games where both players have n strategies each:

$$\begin{bmatrix} p_1q_1 & p_1q_2 & \cdots & p_1q_n \\ p_2q_1 & p_2q_2 & \cdots & p_2q_n \\ \vdots & \vdots & \ddots & \vdots \\ p_nq_1 & p_nq_2 & \cdots & p_nq_n \end{bmatrix} \qquad \begin{bmatrix} \sigma_{11} & \sigma_{12} & \cdots & \sigma_{1n} \\ \sigma_{21} & \sigma_{22} & \cdots & \sigma_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \sigma_{n1} & \sigma_{n2} & \cdots & \sigma_{nn} \end{bmatrix}$$

The probability matrix for a Nash equilibrium where the players independently play mixed strategies $p = (p_1, ..., p_n)$ and $q = (q_1, ..., q_n)$ is on the left; the probability matrix for a CCE where $\sigma = (\sigma_{11}, ..., \sigma_{nn})$ is on the right. Note that the Nash equilibrium probability matrix always has rank 1 because each column is a scalar q_i times p. On the other hand, since σ can be any arbitrary probability distribution, the probability matrix for a CCE can have any rank $\leq n$. This observation leads us to the following definition for rank-k CCEs.

Definition 1.4 (Matrix Version): a **rank-k ϵ-coarse-correlated equilibrium** (rank-k ϵ-CCE) is a probability distribution σ over S such that inequality (1.1) holds for all players i *and* σ defines a probability matrix that has rank $\leq k$. Note that under this definition, CCEs are equivalent to rank-n CCEs, and Nash equilibria are equivalent to rank-1 CCEs. Furthermore, all rank-j CCEs are rank-k CCEs for all $j \leq k$.

Definition 1.5 (General Version): an alternative definition for rank-k CCEs that generalizes beyond two-player games with 2D probability matrices is a probability distribution σ over S such that inequality (1.1) holds $\forall i \in [n]$ *and* σ has the following form:

$$\sigma = \sum_{j=1}^{k} w^{(j)} \cdot \bigotimes_{i=1}^{n} p_i^{(j)}$$

such that for every $s = (s_1, ..., s_n) \in S, \sigma_s = \sum_{j=1}^{k} w^{(j)} \cdot \prod_{i=1}^{n} p_i^{(j)}[s_i]$, where $\sum_{j=1}^{k} w^{(j)} = 1$. Each $p_i^{(j)}$ denotes a probability distribution over S_i referred to as player i's j-th **component** mixed strategy. Note that since σ collapses into a product distribution when $k = 1$, rank-1 CCEs are still equivalent to Nash equilibria under this definition. Also, by the definition of matrix rank, any probability matrix with rank k can always be split into k components and written in the above form, and any probability distribution that satisfies this definition can be written as a probability matrix with rank k. Therefore, in two-player games with 2D probability matrices, these definitions are in fact equivalent.

In particular, for rank-2 CCEs, for every $s = (s_1, ..., s_n) \in S$

$$\sigma_s = w \cdot \prod_{i=1}^{n} p_i^{(1)}[s_i] + (1 - w) \cdot \prod_{i=1}^{n} p_i^{(2)}[s_i]$$

As noted in [HM00], CCEs can alternatively be interpreted as the "distribution of play instructions given to the players by some 'device' or 'referee.'" In equilibrium, this 'referee' selects a strategy profile by drawing from the commonly known joint probability distribution and sends a private signal to each player telling them which pure strategy to play.

Rank-2 CCEs have a similarly intuitive interpretation. First, all players observe the result of flipping a w-weighted coin. Then, depending on the result, all players play according to their first (heads) or second (tails) component. Note that in contrast to CCEs where the 'referee' sends each player a signal containing the exact strategy that they should play, rank-2 CCEs require far less communication as the players only observe some common coin flip before independently playing according to their own component mixed strategies.

2 Rock Paper Scissors

2.1 Description

In order to get an initial understanding of how rank-2 coarse-correlated equilibria differ from Nash equilibria and CCEs, we first begin by examining and characterizing rank-2 CCEs in the game Rock Paper Scissors, which can be described by the following reward matrix:

		Player 2		
		R	P	S
	R	$0, 0$	$-1, 1$	$1, -1$
Player 1	P	$1, -1$	$0, 0$	$-1, 1$
	S	$-1, 1$	$1, -1$	$0, 0$

Table 2.1: RPS Reward Matrix

We start with Rock Paper Scissors (RPS) because RPS is perhaps the simplest and most canonical game that we can analyze in the context of rank-2 CCEs. In particular, two-player, 3x3 games are the smallest possible games where rank-2 CCEs are a strict subset of CCEs, and since RPS is a symmetric, zero-sum game, it is relatively easy to characterize CCEs and rank-2 CCEs by parameterizing the probability distributions in terms of only a few variables.

2.2 Nash Equilibrium

It is generally well known at this point that RPS has a unique Nash equilibrium where both players play a mixed strategy that selects rock, paper, or scissors with equal probability:

$$p = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} \quad q = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix}$$

We skip the proof of this result here so that we can focus on its relevance in the context of rank-2 CCEs. In particular, we can observe that in this unique Nash equilibrium, the probability distribution over strategy profiles is uniform, so the rank of the resulting probability matrix is one. Thus, we can confirm that this Nash equilibrium is in fact a rank-1 CCE.

2.3 Coarse-Correlated Equilibria

In order to characterize the CCEs and rank-2 CCEs in RPS, we will use the following probability matrix to denote probability distributions over strategy profiles, where $p_i \geq 0$ for $i = 1, 2, ..., 9$ and $\sum_{i=1}^{9} p_i = 1$ so that the probability distribution is valid:

		Player 2: R	P	S
Player 1	R	p_1	p_2	p_3
	P	p_4	p_5	p_6
	S	p_7	p_8	p_9

Table 2.2: RPS Probability Matrix

Besides the Nash equilibrium, two examples of CCEs in RPS are shown in the following probability matrices. We note that in both examples, the rank of the probability matrix is three (i.e., full rank). Therefore, rank-2 CCEs are in fact a strict subset of CCEs in RPS:

	R	P	S
R	$\frac{1}{3}$	0	0
P	0	$\frac{1}{3}$	0
S	0	0	$\frac{1}{3}$

	R	P	S
R	0	0	$\frac{1}{3}$
P	0	$\frac{1}{3}$	0
S	$\frac{1}{3}$	0	0

We then use Mathematica to find that the following properties must hold for any CCE in RPS (see `rps.nb`), although it still remains to construct a formal proof for this result:

- $p_2 = p_4$
- $p_3 = p_7$
- $p_6 = p_8$
- $p_1 + p_2 + p_3 = \frac{1}{3}$
- $p_4 + p_5 + p_6 = \frac{1}{3}$
- $p_7 + p_8 + p_9 = \frac{1}{3}$

We now show that *any* probability distribution that satisfies the above properties is a CCE in RPS, thereby establishing equivalence between probability distributions that are CCEs in RPS and probability distributions that satisfy the above properties.

Proof: Let the reward for each player in RPS be denoted by r_1 and r_2, respectively, where

$$r_1 = p_3 + p_4 + p_8 - p_2 - p_6 - p_7 \qquad r_2 = -r_1 = p_2 + p_6 + p_7 - p_3 - p_4 - p_8$$

Then, if $p_2 = p_4$, $p_3 = p_7$, and $p_6 = p_8 \implies r_1 = r_2 = 0$.

Furthermore, given that $p_1 + p_2 + p_3 = \frac{1}{3}$, $p_4 + p_5 + p_6 = \frac{1}{3}$, and $p_7 + p_8 + p_9 = \frac{1}{3}$, the probability that each player plays rock, paper, or scissors is $\frac{1}{3}$. Therefore, the reward that either player receives from deviating to only playing rock, only playing paper, or only playing scissors is always $\frac{1}{3}(1) + \frac{1}{3}(-1) + \frac{1}{3}(0) = 0$. Thus, neither player has any incentive to deviate, so any probability distribution that satisfies the above properties must also be a CCE. □

Intuitively, since RPS is a symmetric, zero-sum game, it makes sense that such an equivalence between CCEs and symmetric probability matrices holds. Given this result, we can now use only three variables (p_1, p_2, and p_5) to parameterize the space of CCEs in RPS:

	R	P	S
R	p_1	p_2	$\frac{1}{3} - p_1 - p_2$
P	p_2	p_5	$\frac{1}{3} - p_2 - p_5$
S	$\frac{1}{3} - p_1 - p_2$	$\frac{1}{3} - p_2 - p_5$	$-\frac{1}{3} + p_1 + 2 \cdot p_2 + p_5$

Table 2.3: RPS CCE Probability Matrix

Plotting this region in (p_1, p_2, p_5)-space gives us the following convex polytope (see `rps.nb`):

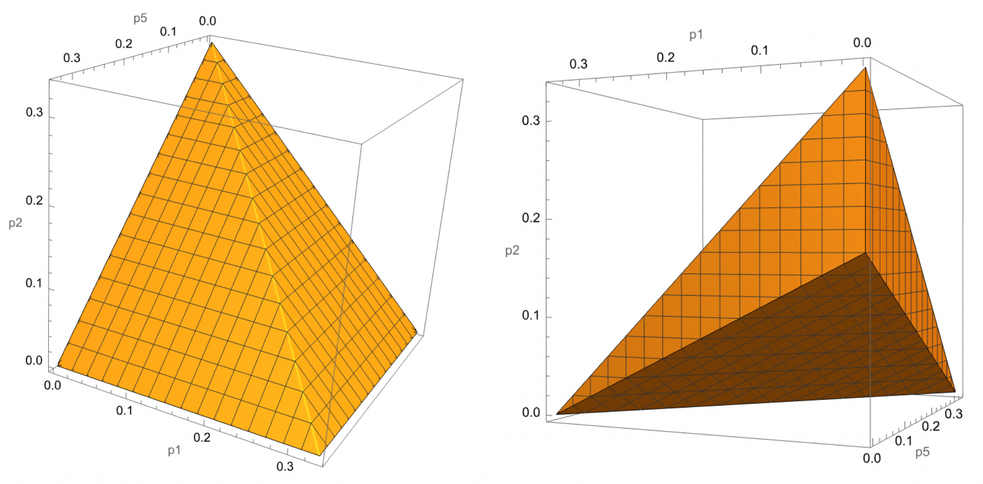

Figure 2.1: Region of CCEs in RPS

2.4 Rank-2 Coarse-Correlated Equilibria

Two examples of rank-2 CCEs in RPS are shown in the following probability matrices:

	R	P	S
R	$\frac{1}{6}$	0	$\frac{1}{6}$
P	0	$\frac{1}{3}$	0
S	$\frac{1}{6}$	0	$\frac{1}{6}$

	R	P	S
R	0	$\frac{2}{9}$	$\frac{1}{9}$
P	$\frac{2}{9}$	0	$\frac{1}{9}$
S	$\frac{1}{9}$	$\frac{1}{9}$	$\frac{1}{9}$

Alternatively, under our more general definition for rank-2 CCEs, these probability matrices can be translated into the following component mixed strategies, where player 1 plays p and player 2 plays q. For the first rank-2 CCE's probability matrix:

- $p^{(1)} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}$
- $q^{(1)} = \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}$
- $w^{(1)} = w = \frac{2}{3}$
- $p^{(2)} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$
- $q^{(2)} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$
- $w^{(2)} = 1 - w = \frac{1}{3}$

And for the second rank-2 CCE's probability matrix:

- $p^{(1)} = \begin{bmatrix} 0 & \frac{2}{3} & \frac{1}{3} \end{bmatrix}$
- $q^{(1)} = \begin{bmatrix} \frac{2}{3} & 0 & \frac{1}{3} \end{bmatrix}$
- $w^{(1)} = w = \frac{1}{2}$
- $p^{(2)} = \begin{bmatrix} \frac{2}{3} & 0 & \frac{1}{3} \end{bmatrix}$
- $q^{(2)} = \begin{bmatrix} 0 & \frac{2}{3} & \frac{1}{3} \end{bmatrix}$
- $w^{(2)} = 1 - w = \frac{1}{2}$

It turns out that even in a relatively simple game like RPS, it is still difficult to provide a full algebraic characterization for rank-2 CCEs. Therefore, we instead seek a graphical representation that can help us gain a better sense of the space of rank-2 CCEs in RPS.

By the Invertible Matrix Theorem, a 3x3 probability matrix has rank ≤ 2 if and only if its determinant is 0. We also know from our exploration of CCEs in RPS that a probability matrix is a CCE in RPS if and only if it has the form described by Table 2.3. Therefore, a probability matrix P_{mat} is a rank-2 CCE if and only if:

$$\det P_{mat} = -\frac{1}{9}p_1 + \frac{2}{9}p_2 - \frac{1}{9}p_5 - p_2^2 + p_1 p_5 = 0$$

Now, if we solve for p_5 in terms of p_1 and p_2, we can parameterize the entire space of rank-2 CCEs in RPS in terms of just two variables (p_1 and p_2):

$$p_5 = \begin{cases} \frac{\frac{2}{9}p_2 - \frac{1}{9}p_1 - p_2^2}{\frac{1}{9} - p_1} & \text{if } p_1 \neq \frac{1}{9} \\ \left[0, \frac{2}{9}\right] & \text{if } p_1 = \frac{1}{9} \end{cases}$$

It is worth noting that when $p_1 = \frac{1}{9}$, $\det P_{mat} = 0 \iff p_2 = \frac{1}{9}$, so as long as $p_5 \in \left[0, \frac{2}{9}\right]$, then the probability constraints will still be satisfied. We can now plot the region of rank-2 CCEs in RPS in both (p_1, p_2)-space and in (p_1, p_2, p_5)-space (see `rps.nb`):

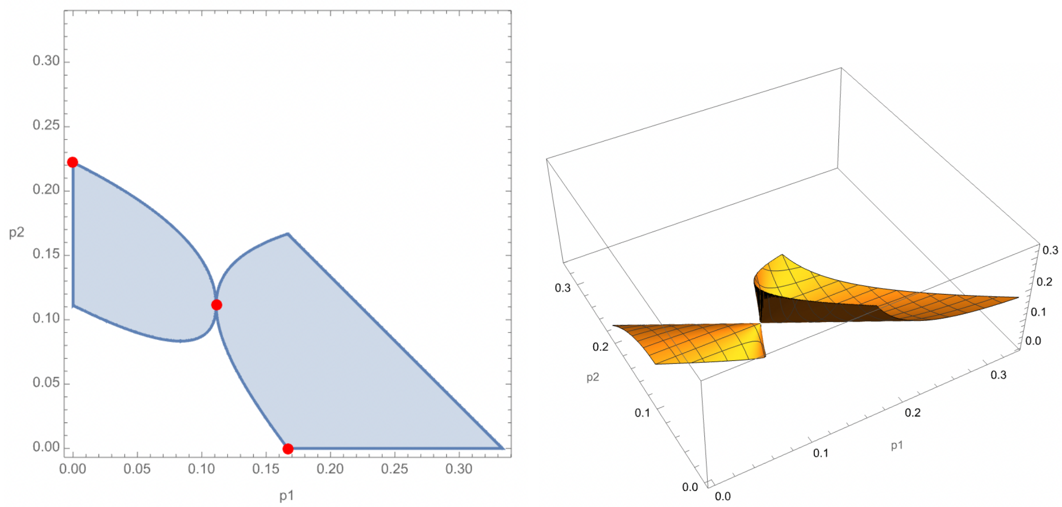

Figure 2.2: Region of Rank-2 CCEs in RPS

We can observe that unlike the space of CCEs, the region of rank-2 CCEs in RPS is non-convex, which helps explain why it is so difficult to characterize rank-2 CCEs even in a relatively simple game like RPS. The three points labelled in the two-dimensional plot represent the two example rank-2 CCEs shown earlier as well as the unique Nash equilibrium. It is interesting to note that the Nash equilibrium appears to occur right at a singular point in the region of rank-2 CCEs; it is still unclear whether this phenomenon is unique to RPS (or some special class of games) and occurs merely by coincidence or whether it may hint at some underlying connection between Nash equilibria and rank-2 CCEs, and it may be worth exploring other games to see whether we can observe a similar relationship.

3 Online Learning Experiment

3.1 Description

We now begin our discussion on the computational complexity of finding rank-k CCEs; we start by first investigating the close relationship between rank-k CCEs and no-regret online learning algorithms. To summarize, no-regret online learning algorithms generate a series of mixed strategies $\{p_i^{(t)}\}_{t\in[T]}$ for each player i such that in each period t, each player uses some no-regret algorithm to compute their mixed strategy $p_i^{(t)}$ for the current period based on the rewards earned in previous periods. In particular, the two no-regret algorithms that we focus on here are multiplicative weights update (MWU) and its optimistic counterpart (OMWU) defined as follows [Ana+24]:

$$p_i^{(t)}[s_i] = \frac{p_i^{(t-1)}[s_i]\exp\left(\eta_i r_i^{(t-1)}[s_i]\right)}{\sum_{s_i'\in S_i} p_i^{(t-1)}[s_i']\exp\left(\eta_i r_i^{(t-1)}[s_i']\right)} \qquad \forall s_i \in S_i \tag{MWU}$$

$$p_i^{(t)}[s_i] = \frac{p_i^{(t-1)}[s_i]\exp\left(\eta_i (2r_i^{(t-1)}[s_i] - r_i^{(t-2)}[s_i])\right)}{\sum_{s_i'\in S_i} p_i^{(t-1)}[s_i']\exp\left(\eta_i (2r_i^{(t-1)}[s_i'] - r_i^{(t-2)}[s_i'])\right)} \qquad \forall s_i \in S_i \tag{OMWU}$$

where $p_i^{(t)}[s_i]$ denotes the probability that player i plays strategy s_i in period t, $r_i^{(t)}[s_i]$ denotes player i's expected reward from playing only strategy s_i in period t, and η_i denotes the learning rate. For simplicity, we initialize $p_i^{(-1)}$ and $p_i^{(0)}$ to the uniform distribution.

Let m denote the number of strategies available to players. It has been shown that no matter the number of players or structure of the reward matrix, after $T = O\left(\frac{\log m}{\epsilon^2}\right)$ periods, the following probability distribution σ is guaranteed to converge to an ϵ-CCE [FV97; HM00]:

$$\sigma = \frac{1}{T}\sum_{t=1}^{T}\left[\bigotimes_{i=1}^{n} p_i^{(t)}\right]$$

Observe that σ exactly matches our definition of rank-k ϵ-CCE, where $k = T = O\left(\frac{\log m}{\epsilon^2}\right)$ and $w^{(j)} = \frac{1}{T}$ for all $j \in [k]$. Therefore, not only can no-regret online learning algorithms efficiently compute ϵ-CCEs, but these ϵ-CCEs are more precisely rank-$O\left(\frac{\log m}{\epsilon^2}\right)$ ϵ-CCEs.

While $T = O\left(\frac{\log m}{\epsilon^2}\right)$ provides an upper bound on the number of periods it takes to converge to an ϵ-CCE, it is currently unknown whether this bound is tight. This result is closely related to our discussion on the computational complexity of rank-2 CCEs because if it was discovered that T grows sub-logarithmically with respect to the number of strategies m, then we could use no-regret online learning algorithms to efficiently compute sub-logarithmic-rank ϵ-CCEs; in particular, if T was constant with respect to the number of strategies m, then we could use online learning to efficiently compute rank-$O(1)$ ϵ-CCEs.

In order to get an initial understanding of the relationship between T and m, we ran no-regret online learning algorithms on a large number of randomly generated games and recorded the number of periods it took to converge to an ϵ-CCE. The generated data serves as empirical evidence that supports the proposition that $T = \Theta\left(\frac{\log m}{\epsilon^2}\right)$.

3.2 Methods

The following procedure was used to generate our simulated data (see `online-learning.py`):

(1) All data was generated using an ϵ-approximation of 0.01. We selected $\epsilon = 0.01$ because it is small enough to give a close approximation for CCEs but still large enough so that online learning terminates in a computationally reasonable number of periods.

(2) For simplicity, all data was generated using two-player games where each player has N strategies and rewards in the interval $[-1, 1)$.

(3) For each $N = 2, 3, ..., 128$, a random reward matrix was generated with rewards drawn uniformly at random from the interval $[-1, 1)$, a no-regret online learning algorithm was run, and the number of periods it took to converge to an ϵ-CCE was recorded.

(4) Due to computational limitations, it was infeasible to run online learning for every $128 < N < 1024$. However, in the interest of generating more data on a logarithmic scale, step (3) was also repeated for $N = 256$, 512, and 1024.

(5) Steps (3) and (4) were repeated over 100 iterations to reduce the effects of randomness.

In order to test whether the relationship between T and m differs between different no-regret algorithms, we repeated this experiment using both MWU and OMWU. We also repeated this experiment with zero-sum games to examine how T grows in a more structured class of games. Due to a lack of time and available computational resources, these experiments were only run with $N = 2, 3, ..., 88$, which is still enough data to observe trends in the relationship.

3.3 Results

3.3.1 Graphs

The linear scale plot below is plotted only for $N \in [2, 128]$ to improve readability and avoid over-compression of the x-axis. The log scale plot still includes data for all N.

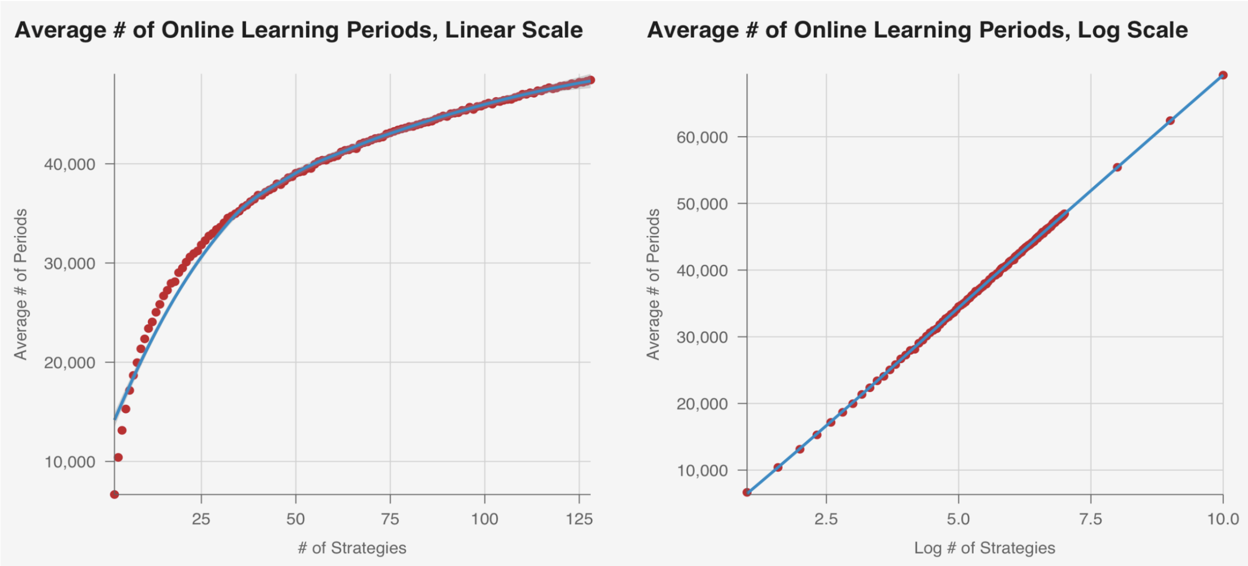

Figure 3.1: Online Learning w/ MWU in General Games

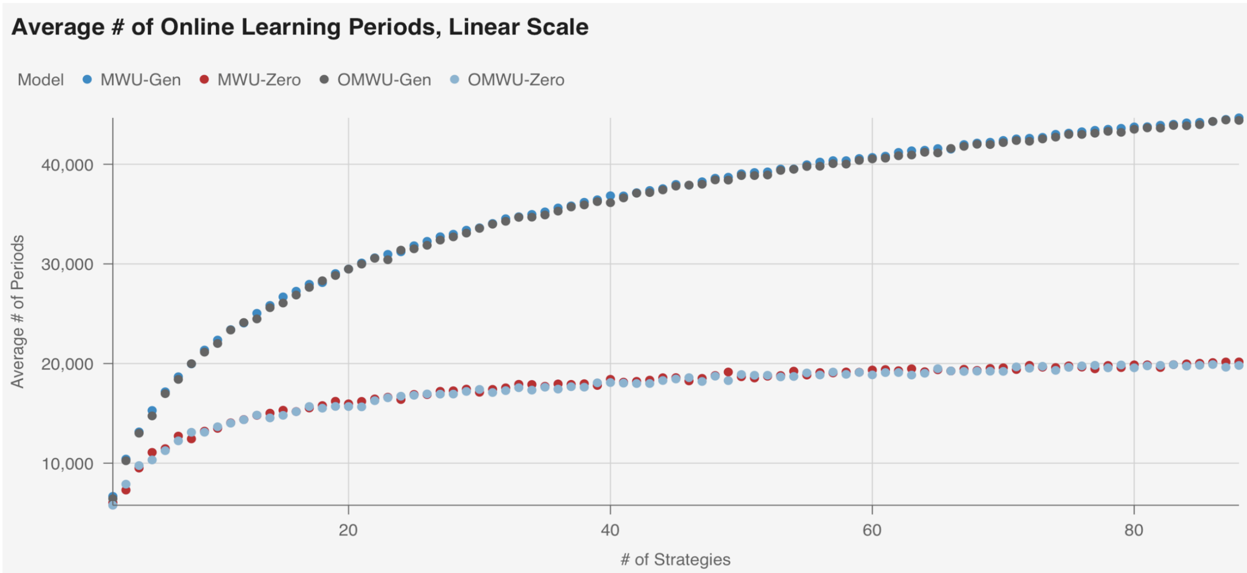

Figure 3.2: Online Learning, Linear Scale

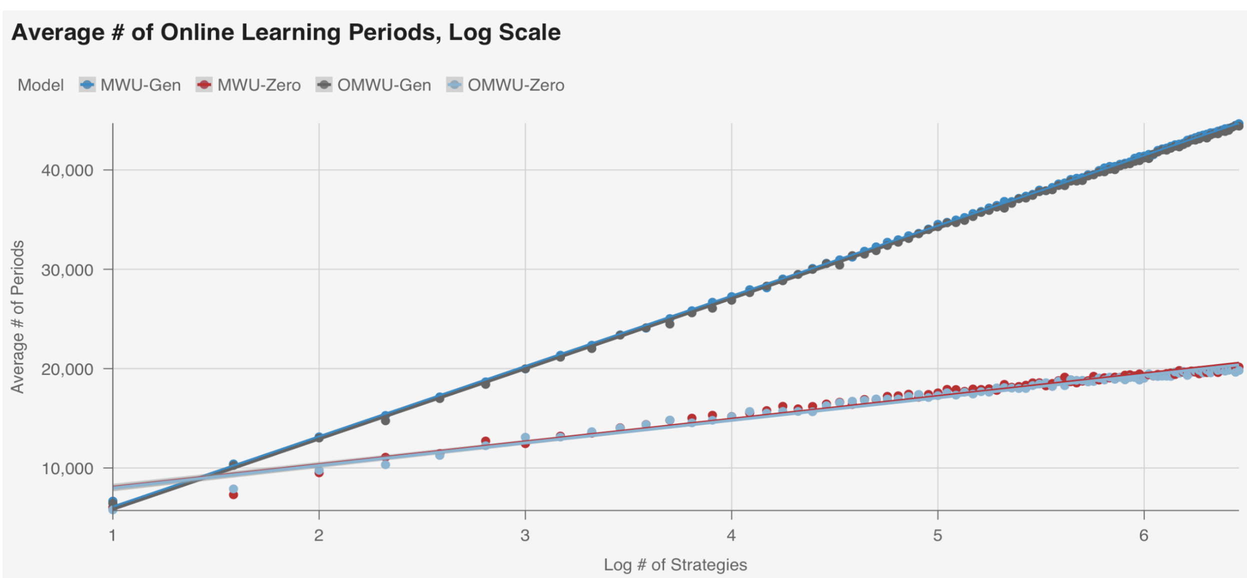

Figure 3.3: Online Learning, Log Scale

3.3.2 Discussion

For online learning in general games, the relationship between the number of periods and the number of strategies appears (somewhat remarkably) almost exactly logarithmic for both MWU and OMWU no-regret algorithms. Thus, this experiment provides empirical evidence that supports the proposition that the $O\left(\frac{\log m}{\epsilon^2}\right)$ upper bound on T is in fact a tight bound. To further generalize these results, it may be worth repeating this experiment in games with different numbers of players, different values of ϵ, different numbers of strategies for each player, and different distributions for drawing rewards. It will be interesting to see whether it can be proven that $T = \Theta\left(\frac{\log m}{\epsilon^2}\right)$ is the best achievable bound for any no-regret online learning algorithm, but we leave this work to future research. For now, since it appears probable that the bound is tight, we can conjecture that finding rank-$O(1)$ CCEs (e.g., rank-2 CCEs) is likely computationally hard.

It is interesting to note that the average number of periods it took for online learning to converge to an ϵ-CCE in zero-sum games was far lower than in general games. This result seems reasonable because the reward matrix in zero-sum games enforces extra structure that likely makes it easier to converge to an ϵ-CCE quickly. However, the relationship between the average number of periods and the number of strategies still appears close to logarithmic even in zero-sum games, although the trend is not as exact as in general games, and it appears that the average number of periods may actually grow slightly faster than logarithmically in zero-sum games. It is important to note that in two-player zero-sum games, Nash equilibria *can* be efficiently computed by using linear programming duality [Das08], so the fact that the number of periods (and thus the rank of the CCE) still appears to grow close to logarithmically even in zero-sum games supports the idea that there exists some inherent connection between no-regret online learning algorithms and logarithmic growth.

4 Approach to Proving PPAD-Hardness

4.1 Background

We finally turn our attention to a discussion on our initial approach toward constructing a proof that finding rank-2 CCEs is computationally hard. No complete results have been proven yet, and much more further research is likely required, but we present ideas we tried and challenges we encountered that seem to make such a proof difficult.

We first provide some definitions that are relevant to discussing the computational complexity of equilibria. The class TFNP is the set of all problems, called total search problems, for which at least one solution always exists. By Brouwer's Fixed Point Theorem and Nash's Theorem, the problems of finding fixed points and computing Nash equilibria are in TFNP.

Informally, the class PPAD is defined as all problems in TFNP whose totality is established by invoking the following lemma on a graph whose vertices represent the problem's solution space: "in any directed graph with one unbalanced node (node with outdegree different from its indegree), there is another unbalanced node" [Das08]. Both the problem of finding Brouwer fixed points and computing Nash equilibria are in PPAD. (We defer to Daskalakis's paper [Das08] for a more formal definition of these complexity classes as well as a proof that the problem of finding Nash equilibria is in PPAD). Note that since all Nash equilibria are rank-2 CCEs, the problem of finding rank-2 CCEs must also be in PPAD.

4.2 Computing Nash Equilibria is PPAD-Hard [Das08]

Since our initial approach to proving that computing rank-2 CCEs is PPAD-hard closely follows Daskalakis's proof showing that computing Nash equilibria is PPAD-hard, we present a brief summary of relevant ideas and methods from Daskalakis's paper here.

The core component of the proof relies on small graphical games called 'gadgets', where graphical games are games that can be represented by a graph mapping vertices to players and edges to interactions between players. These 'gadgets' act as arithmetic gates and comparators, and they can be put together to form larger graphical games that can perform circuit-like computations outputting real numbers that represent mixed strategies.

The proof begins with a PPAD-complete problem related to Brouwer's fixed point theorem that looks for fixed points in special kinds of functions mapping the unit cube to itself. Each point in the cube corresponds to the mixed strategies (which can be represented as real numbers) of three players who have two strategies each. Then, functions from the cube to itself can be implemented by the players' best responses in a 'gadget'-based graphical game so that Nash equilibria in the graphical game correspond to fixed points on the cube. Finally, by proving polynomial-time equivalence between finding Nash equilibria in such graphical games and finding Nash equilibria in general normal-form games, it can be shown that the problem of finding Nash equilibria is PPAD-complete [Das08].

In order to reconstruct such a proof for rank-2 CCEs, it becomes necessary to find similar 'gadgets' that can act as arithmetic gates and comparators in the context of rank-2 CCEs. In particular, we focus on a (simplified) version of the game $\mathcal{G}_+$ that Daskalakis's paper presents in Proposition 3.3 and Figure 3.3 [Das08]:

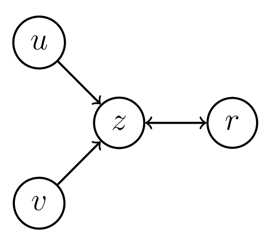

Figure 4.1: Graph of $\mathcal{G}_+$ Game

$u \backslash v$	0	1
0	0	$\frac{1}{2}$
1	$\frac{1}{2}$	1

z's reward if $z = 0$

r	
0	0
1	1

z's reward if $z = 1$

$r \backslash z$	0	1
0	0	1
1	1	0

r's reward

To simplify notation, $u, v, r,$ and z denote the probabilities that the respective players play strategy 1. As shown in Proposition 3.3 of Daskalakis's paper, in any Nash equilibrum of $\mathcal{G}_+$, $r = \frac{1}{2}u + \frac{1}{2}v$, so this 'gadget' implements a form of addition, hence the name $\mathcal{G}_+$ [Das08].

We would like to find a similar 'gadget' $\mathcal{G}_+^{(2)}$ such that in any rank-2 CCE of $\mathcal{G}_+^{(2)}$, some kind of addition property holds between $r, u,$ and v as it does in $\mathcal{G}_+$. For example:

$$w \cdot r^{(1)} + (1-w) \cdot r^{(2)} = w \cdot (\tfrac{1}{2}u^{(1)} + \tfrac{1}{2}v^{(1)}) + (1-w) \cdot (\tfrac{1}{2}u^{(2)} + \tfrac{1}{2}v^{(2)})$$

Unfortunately, this property does not hold in $\mathcal{G}_+$, and as we will discuss later, it is surprisingly difficult to even define what 'addition' means in the context of rank-2 CCEs.

4.3 Simplifying With α

To simplify the problem, we define $\alpha^{(j)} = \frac{1}{2}u^{(j)} + \frac{1}{2}v^{(j)}$ for $j = 1, 2$. This gives us the following reward matrix for a game played between r and z:

		z	
		0	1
r	0	$0, \alpha$	$1, 0$
	1	$1, \alpha$	$0, 1$

We can now see why the problem quickly becomes complicated when we move from Nash equilibria to rank-2 CCEs. In particular, not only do two different components show up in the probability distribution, but they also show up in the reward matrix because α depends on u and v. In fact, it can be shown that if $\alpha^{(1)} = \alpha^{(2)} = \alpha$, then $wr^{(1)} + (1-w)r^{(2)} = \alpha = \frac{1}{2}u + \frac{1}{2}v$ (see appendix), which seems like a natural notion of addition in this context, but such a property is much more difficult to derive once α is separated into two components.

4.4 Expanding to Three Strategies

The next major roadblock to finding an addition 'gadget' for rank-2 CCEs is the seeming need to expand the game so that r and z have three strategies each. In 2x2 games, rank-2 CCEs are equivalent to CCEs, and it wouldn't make sense to find an addition 'gadget' for CCEs because we already know that CCEs are efficiently computable. Therefore, it seems necessary to expand the game so that each player has three strategies. However, this adjustment leads to two complications that are difficult to navigate.

First, it is difficult to figure out how to populate the 3x3 reward matrix to produce a game that might enforce some addition property. In the 2x2 game, there is some intuition that z can guarantee reward α by playing strategy 0 and will only play strategy 1 to try and match r playing strategy 1. On the other hand, r tries to avoid z by playing the opposite strategy that z plays. In equilibrium, $r = \alpha$ because these forces balance each other out. Adding a third strategy makes the game significantly more difficult to interpret, and it is not entirely clear how to proceed with populating a 3x3 reward matrix.

Second, it is no longer clear how to define 'addition'. When each player has only two strategies, each player's mixed strategy can be represented by a real number that corresponds to the probability of playing strategy 1, so it was natural to define addition in terms of these real numbers. Even in the context of rank-2 CCEs, it seems natural to use the weighted sum of components (e.g., $wr^{(1)} + (1-w)r^{(2)}$ for r) because it corresponds to the probability of playing strategy 1 and not 0. However, once a third strategy is added, two real numbers (or two weighted sums of components for rank-2 CCEs) are required to fully characterize any mixed strategy. One idea is to try to continue using the probability of playing strategy 1, but this idea may not work well in the context of the unit cube Brouwer problem discussed earlier where each point corresponds with players' mixtures over two strategies.

4.5 Weighted Inner Products

Let $\langle x_1, ..., x_n\rangle_k = \sum_{j=1}^{k} \left[w^{(j)} \cdot \prod_{i=1}^{n} x_i^{(j)}\right]$ denote the rank-k weighted inner product, which is a compact and convenient notation for writing expressions in the context of rank-k CCEs. For example, we can write the rank-2 CCE constraints in $\mathcal{G}_+$ as follows (omitting subscripts):

- $\langle r, z\rangle \geq \langle \alpha, z\rangle$
- $\langle \alpha\rangle + \langle r, z\rangle \geq \langle r\rangle + \langle \alpha, z\rangle$
- $\langle r\rangle \geq 2\langle r, z\rangle$
- $\langle r\rangle + 2\langle z\rangle \geq 1 + 2\langle r, z\rangle$

In fact, the rank-k CCE constraints can be written exactly in the above form for *any* k, which is problematic. In particular, suppose we can show that some desirable 'addition' property holds if the above constraints are satisfied. If we can construct such a proof fully in terms of weighted inner products, then the exact same proof would work for all rank-k CCEs. However, as discussed previously, it wouldn't make sense to find an addition 'gadget' for all rank-k CCEs because we already know that (rank-m) CCEs are efficiently computable (where m is the number of strategies available to players). Therefore, it seems like any such proof must break up these weighted inner products and establish some relationship between components, which makes analysis significantly more challenging.

5 Conclusion

In order to bridge the gap between Nash equilibria and coarse-correlated equilibria, we began to investigate a class of equilibria lying between the two call rank-k equilibria. To that end, we first explored the region of rank-2 CCEs in Rock Paper Scissors and saw how rank-2 CCEs differed from Nash equilibria and CCEs. In particular, we saw that the region of rank-2 CCEs in RPS is non-convex, which makes it difficult to characterize, and that the unique Nash equilibrium occurs at a singular point in the region of rank-2 CCEs. It may be interesting to continue exploring rank-2 CCEs in other simple two-player 3x3 games (e.g., a coordination game with multiple Nash equilibria) to see whether this result can be generalized.

We then explored the relationship between rank-k CCEs and no-regret online learning algorithms by repeatedly running online learning on simulated games and providing evidence to support the conjecture that there is a tight logarithmic bound on the number of periods it takes to converge to a CCE. It still remains to prove this result generally, but assuming it is true, it is highly unlikely that rank-$O(1)$ CCEs can be efficiently computed.

Finally, we briefly discussed our initial approach toward constructing a proof that finding rank-2 CCEs is computationally hard. In particular, in order to follow methods similar to those presented in Daskalakis's proof that computing Nash equilibria is PPAD-hard, we looked to find a 'gadget' similar to $\mathcal{G}_+$ that works for rank-2 CCEs. Unfortunately, rank-2 CCEs introduce a series of challenges that make it difficult to come up with such a 'gadget', and further work is required to figure out how we might overcome these issues and eventually prove that finding rank-2 CCEs is in fact computationally hard.

References

[Ana+24] Ioannis Anagnostides et al. “On the Complexity of Computing Sparse Equilibria and Lower Bounds for No-Regret Learning in Games”. In: Leibniz International Proceedings in Informatics (2024). Ed. by Venkatesan Guruswami.

[Das08] Constantinos Daskalakis. “The Complexity of Nash Equilibria”. PhD thesis. University of California, Berkeley, 2008.

[FV97] Dean P. Foster and Rakesh V. Vohra. “Calibrated Learning and Correlated Equilibrium”. In: *Games and Economic Behavior* 21 (Oct. 1997), pp. 40–55. DOI: `https://doi.org/10.1006/game.1997.0595`.

[HM00] Sergiu Hart and Andreu Mas-Colell. “A Simple Adaptive Procedure Leading to Correlated Equilibrium”. In: *Econometrica* 68.5 (Sept. 2000), pp. 1127–1150.

[Nas51] John Nash. “Non-Cooperative Games”. In: *Annals of Mathematics* 54.2 (Sept. 1951), pp. 286–295. DOI: `https://doi.org/10.2307/1969529`.

A Appendix

We show that if $\alpha^{(1)} = \alpha^{(2)} = \alpha$, then $wr^{(1)} + (1-w)r^{(2)} = \alpha$.

From section 4.5, the following constraints hold for any rank-2 CCE:

(1) $\langle r, z\rangle \geq \langle \alpha, z\rangle$

(2) $\langle \alpha\rangle + \langle r, z\rangle \geq \langle r\rangle + \langle \alpha, z\rangle$

(3) $\langle r\rangle \geq 2\langle r, z\rangle$

(4) $\langle r\rangle + 2\langle z\rangle \geq 1 + 2\langle r, z\rangle$

$(1) + (3) \implies \langle r\rangle \geq 2\langle \alpha, z\rangle$

$(2) + (3) \implies \langle r\rangle \leq 2\langle \alpha\rangle - 2\langle \alpha, z\rangle$

Note that since $\alpha^{(1)} = \alpha^{(2)} = \alpha$, we can factor α out of the weighted inner products:

$\implies \langle r\rangle \geq 2\alpha\langle z\rangle$

$\implies \langle r\rangle \leq 2\alpha(1 - \langle z\rangle)$

Combining these inequalities gives $\alpha = 0$ or $\langle z\rangle \leq \frac{1}{2}$:

If $\alpha = 0$, then $0 \geq \langle r\rangle \geq 0$, so $\langle r\rangle = \alpha = 0$.

On the other hand, if $z \leq \frac{1}{2}$, then:

$(1) + (4) \implies \langle r\rangle \geq 1 - 2\langle z\rangle + 2\alpha\langle z\rangle \geq \alpha$

$(2) + (4) \implies \langle r\rangle \leq 2\alpha - 1 + 2(1-\alpha)\langle z\rangle \leq \alpha$

$\implies \langle r\rangle = \alpha$ $\square$