CSCI 659 Lecture 4

Spring 2026

Instructor: Haipeng Luo

1 Two-Player Zero-Sum Games

In this lecture, we explore the connection between game theory and online learning, and explain why online learning is the cornerstone for solving large-scale games. We start by considering the simplest two-player zero-sum normal-form games and some fundamental concepts of game theory, before discussing how it is connected to online learning in the next section.

A two-player zero-sum normal-form game can be represented using a payoff matrix $G \in [0,1]^{N\times M}$. Here, one player (called the row or min player, with pronoun "he") has N possible actions and another player (called the column or max player, with pronoun "she") has M possible actions, and entry $G(i,j)$ represents the loss of the row player if he selects action i while the column player selects action j, which is also the reward for the column player (hence zero-sum).

A classical example is the Rock-Paper-Scissors game. If we assign loss 1 for losing the game, 0 for winning, and $1/2$ for a tie, then G is

$$\begin{array}{r c} & \begin{array}{ccc}\text{Rock} & \text{Paper} & \text{Scissors}\end{array} \\ \begin{array}{r}\text{Rock}\\ \text{Paper}\\ \text{Scissors}\end{array} & \left(\begin{array}{ccc} 1/2 & 1 & 0 \\ 0 & 1/2 & 1 \\ 1 & 0 & 1/2 \end{array}\right) \end{array}.$$

The exact same idea applies to much more complicated games, including those that involve sequential structures. For example, a poker game can be formulated in this way with a huge G where each action corresponds to a complete strategy of playing this poker game.

Instead of playing a fixed action (also called "pure strategy"), it often makes more sense to play an action randomly according to a distribution (called "mixed strategy"). For some mixed strategy $p \in \Delta(N)$ for the row player and some mixed strategy $q \in \Delta(M)$ for the column player, the expected loss for the row player, which is also the expected reward of the column player, is denoted by $G(p,q) = p^\top G q = \sum_{i=1}^N \sum_{j=1}^M p(i)q(j)G(i,j)$. We will also use the notation $G(i,q)$ and $G(p,j)$ to denote $\sum_{j=1}^M q(j)G(i,j)$ and $\sum_{i=1}^N p(i)G(i,j)$ respectively.

The most fundamental solution concept in game theory is the *Nash equilibrium*. A pair of mixed strategy (p,q) is called a Nash equilibrium if neither player has the incentive to change his/her strategy given that the opponent is keeping his/hers. In other words, both players are best-responding to each other and thus happy about the current situation. Formally, this means that

$$G(p,j) \le G(p,q) \le G(i,q), \quad \forall i \in [N], j \in [M].$$

One can easily verify that for the Rock-Paper-Scissors game, playing uniformly at random for both players is a Nash Equilibrium (in fact the only one).

On the other hand, minimax solution is also a natural concept for a two-player zero-sum game. Specifically, in the worst case, playing p leads to a loss of at most $\max_q G(p,q)$ for the row player if the column player sees p before making her decision, and therefore in this sense the worst-case optimal strategy for the row player is $p^\star \in \operatorname{argmin}_p \max_q G(p,q)$, which is called the minimax strategy. Similarly, the maximin strategy for the column player is $q^\star \in \operatorname{argmax}_q \min_p G(p,q)$. Together, we call $(p^\star, q^\star)$ a minimax solution of the game.

Therefore, $\min_p \max_q G(p,q)$ and $\max_q \min_p G(p,q)$ are respectively the smallest loss and the largest reward that the respective player can hope for when against an optimal opponent who plays second. How are these two values related? Intuitively, both players are playing optimally in the two expressions, but there should be no disadvantage in playing second. Therefore we should have $\min_p \max_q G(p,q) \geq \max_q \min_p G(p,q)$ (row player playing first on the left and second on the right). Indeed, this is true by a simple argument:

$$\min_p \max_q G(p,q) = \max_q G(p^\star, q) \geq G(p^\star, q^\star) \geq \min_p G(p, q^\star) = \max_q \min_p G(p,q). \tag{1}$$

While one may imagine that this inequality should be strict at least for some cases, the surprising fact is that the reverse inequality is also true and thus the two values are exactly the same! In other words, if both players are playing optimally, there is no difference in playing first or second. This is the celebrated von Neumann's minimax theorem.

Theorem 1 (von Neumann's minimax theorem)**.** *For any two-player zero-sum game $G \in [0,1]^{N\times M}$, we have*

$$\min_{p\in\Delta(N)} \max_{q\in\Delta(M)} G(p,q) = \max_{q\in\Delta(M)} \min_{p\in\Delta(N)} G(p,q).$$

This single value is called the value of the game, denoted by $v(G)$. The original proof relies on a fixed-point theorem, but we will prove it in a different way *by running online learning algorithms* in the next section. For now, we discuss the connection between these different notions we have discussed so far: Nash equilibrium, minimax solution, and the value of the game.

Theorem 2. *A pair of mixed strategy $(p^\star, q^\star)$ is a minimax solution if and only if it is also a Nash equilibrium. Moreover, $G(p^\star, q^\star)$ is the value of the game.*

Proof. Suppose that $(p^\star, q^\star)$ is a minimax solution. Then Eq. (1) holds, but the minimax theorem implies that both inequalities in Eq. (1) are in fact equalities, which further implies $G(p^\star, q^\star) = v(G)$ and that $(p^\star, q^\star)$ is a Nash equilibrium.

For the other direction, if $(p^\star, q^\star)$ is a Nash equilibrium, then by definition and optimality, we have

$$\min_p \max_q G(p,q) \leq \max_q G(p^\star, q) = G(p^\star, q^\star) = \min_p G(p, q^\star) \leq \max_q \min_p G(p,q).$$

Again, by the minimax theorem, the two inequalities above are in fact equalities, which implies $G(p^\star, q^\star) = v(G)$ and also that $(p^\star, q^\star)$ is a minimax solution. □

By this theorem and the fact that minimax solutions always exist (due to the compactness of the simplex), Nash equilibria also always exist.

Question 1. *Theorem 2 asserts that a minimax solution $(p^\star, q^\star)$ is a Nash equilibrium and thus $q^\star \in \operatorname{argmax}_q G(p^\star, q)$. Is it true that $(p^\star, q)$ for any $q \in \operatorname{argmax}_q G(p^\star, q)$ is a Nash equilibrium?*

2 Repeated Play and Connections to Online Learning

How should we play a game? If we know the matrix G, then playing with the minimax solutions seems to be a good strategy. However, what if G is unknown? Moreover, minimax solutions might also be overly conservative. For example, if we play Rock-Paper-Scissors with a friend who we know prefers to play Paper, then should we still play uniformly at random? In general, how do we exploit the fact that the opponent might not be optimal?

If the game is only played once, then there is little we can do. However, it is often the case that a game is repeatedly played for many times. In this case, there is hope to apply learning algorithms to learn to play well against a specific opponent. We take the row player as an example and formulate the learning model as follows: at round $t = 1, \ldots, T$,

1. the row player chooses a mixed strategy $p_t \in \Delta(N)$;

2. the column player chooses a mixed strategy $q_t \in \Delta(M)$ (which may or may not depend on p_t);
3. the row player observes $G(i, q_t)$ for all $i \in [N]$.

The feedback model might look unrealistic in this case, but all discussions in this section extend trivially to the weaker model where the row player only observes $G(i, j_t)$ for all i, where j_t is sampled from q_t. In other words, for the realized action j_t of the opponent, the player knows the loss of each of his possible action. Of course, the most challenging case is when the player only observes $G(i_t, j_t)$ where $i_t \sim p_t$ and $j_t \sim q_t$ are the realized action of the respective player. This is essentially a bandit feedback setting, which we will discuss in the second half of the course.

A very natural idea for the player is simply to run an expert algorithm such as Hedge, treating each available action i as an expert. Specifically, given an expert algorithm as a blackbox, p_t will be the output of this algorithm at round t, and the loss vector to be fed to the algorithm is ℓ_t such that $\ell_t(i) = G(i, q_t),\ \forall i$. Let $\mathcal{R}_T$ be the regret of this algorithm and $\bar{q} = \frac{1}{T}\sum_{t=1}^T q_t$. Then we have

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^T \langle p_t, \ell_t\rangle = \frac{1}{T}\sum_{t=1}^T G(p_t, q_t) &= \min_p \frac{1}{T}\sum_{t=1}^T G(p, q_t) + \frac{\mathcal{R}_T}{T} \\ &= \min_p G(p, \bar{q}) + \frac{\mathcal{R}_T}{T} \\ &\le \max_q \min_p G(p, q) + \frac{\mathcal{R}_T}{T}.\end{aligned}$$

Therefore, if $\mathcal{R}_T = o(T)$ and T is large enough, the average loss of the row player is very close to the value of the game, which again is the smallest possible loss if against an optimal opponent. However, by using a learning algorithm instead of a minimax solution directly (if it is available), the average loss can also be much smaller in the case when the opponent is not exactly optimal (that is, when $\bar{q}$ is not close to the maximin strategy and the last inequality is loose).

When both players learn. Even more interesting thing happens if the column player also uses an expert algorithm to come up with q_t (by treating the negative rewards as losses). To see this, let $\mathcal{R}'_T = o(T)$ be the regret of the column player:

$$\mathcal{R}'_T = \sum_{t=1}^T -G(p_t, q_t) - \min_q \sum_{t=1}^T -G(p_t, q) = \max_q \sum_{t=1}^T G(p_t, q) - \sum_{t=1}^T G(p_t, q_t).$$

Then we can repeat the earlier calculation, but now for the column player with $\bar{p} = \frac{1}{T}\sum_{t=1}^T p_t$:

$$\begin{aligned}\frac{1}{T}\sum_{t=1}^T G(p_t, q_t) &= \max_q \frac{1}{T}\sum_{t=1}^T G(p_t, q) - \frac{\mathcal{R}'_T}{T} \\ &= \max_q G(\bar{p}, q) - \frac{\mathcal{R}'_T}{T} \\ &\ge \min_p \max_q G(p, q) - \frac{\mathcal{R}'_T}{T}.\end{aligned}$$

Combining the two derivations, we have shown

$$\min_p \max_q G(p, q) - \max_q \min_p G(p, q) \le \max_q G(\bar{p}, q) - \min_p G(p, \bar{q}) = \frac{\mathcal{R}_T}{T} + \frac{\mathcal{R}'_T}{T}. \qquad (2)$$

Since the term $\epsilon_T \triangleq \frac{\mathcal{R}_T}{T} + \frac{\mathcal{R}'_T}{T}$ (called social average regret) can be made arbitrarily close to 0 as T goes to infinity, we must have

$$\min_p \max_q G(p, q) \le \max_q \min_p G(p, q),$$

which means that we just proved the minimax theorem (recall that the other direction is trivial)! This is a classical result taken from Freund and Schapire [1999]. It is quite intriguing because it says that the existence of a no-regret algorithm implies that the minimax theorem must hold, without using heavier tools such as a fixed-point theorem. In fact, the derivation above also tells us:

$$\max_q G(\bar{p}, q) - \epsilon_T = \min_p G(p, \bar{q}) \le G(\bar{p}, \bar{q}) \le \max_q G(\bar{p}, q) = \min_p G(p, \bar{q}) + \epsilon_T,$$

showing that the empirical average strategies $\bar{p}$ and $\bar{q}$ are *approximately minimax solutions* or *approximate Nash equilibrium* with error ϵ. This provides a highly efficient way to approximately find a Nash equilibrium, making regret minimization algorithms (together with other tricks) one of the most practical ways to solve large-scale games.[1] Moreover, note that when using algorithms such as Hedge for both players so that $\mathcal{R}_T = \mathcal{O}(\sqrt{T \ln N})$ and $\mathcal{R}'_T = \mathcal{O}(\sqrt{T \ln M})$, we see that to reach ϵ accuracy, one only needs $T = \mathcal{O}\left(\frac{\ln(NM)}{\epsilon^2}\right)$ rounds, which has logarithmic dependence on the game size NM. This means that one can compute an approximate Nash equilibrium even without knowing the entire game.

3 Faster Convergence via Adaptivity

We know that the worst-case optimal regret for the expert problem is of order $\mathcal{O}(\sqrt{T})$ (ignoring dependence on the number of actions), which means the convergence rate of the above approach is of order $\mathcal{O}(1/\sqrt{T})$. However, since each player is not really dealing with a worst-case environment, a natural question is whether we can achieve even faster convergence in this specific context. The answer turns out to be yes, and the key is the optimistic algorithms we discussed last time. Indeed, by setting $m_t = \ell_{t-1}$, we know that optimistic Hedge's regret depends on the path length of the loss sequence $\sum_{t=1}^{T} \|\ell_t - \ell_{t-1}\|_\infty$, which in this game setting should intuitively depend on the variation of the opponent's strategy and thus should be small due to the stability of these algorithms. In fact, utilizing the negative regret term that we ignored in the past lectures, we can make this argument even stronger, as shown in the following theorem.

Theorem 3. *Suppose that both players apply optimistic Hedge with the predictor being the loss vector of the last round, that is:*

$$p_t(i) \propto \exp\left(-\eta\left(\ell_{t-1}(i) + \sum_{s<t} \ell_s(i)\right)\right), \quad \textit{where } \ell_s(i) = G(i, q_s),$$

$$q_t(j) \propto \exp\left(-\eta\left(g_{t-1}(j) + \sum_{s<t} g_s(j)\right)\right), \quad \textit{where } g_s(j) = -G(p_s, j).$$

Then with $\eta = \frac{1}{4}$, the total regret of the players is bounded as $\mathcal{R}_T + \mathcal{R}'_T = \mathcal{O}(\ln(NM))$, and thus $(\frac{1}{T}\sum_t p_t, \frac{1}{T}\sum_t q_t)$ is an approximate Nash equilibrium with approximation error $\mathcal{O}(\ln(NM)/T)$.

Proof. Directly applying Theorem 1 of Lecture 3, we have

$$\mathcal{R}_T \le \frac{\ln N}{\eta} + \eta \sum_{t=1}^{T} \|\ell_t - \ell_{t-1}\|_\infty^2 - \frac{1}{4\eta}\sum_{t=2}^{T} \|p_t - p_{t-1}\|_1^2 . \tag{3}$$

Further using the definition of ℓ_t, we bound the path-length term in terms of the stability of the opponent: for any $t > 1$,

$$\|\ell_t - \ell_{t-1}\|_\infty = \max_i |G(i, q_t) - G(i, q_{t-1})| = \max_i |\langle G(i, \cdot), q_t - q_{t-1}\rangle| \le \|q_t - q_{t-1}\|_1 ,$$

where we use $G(i, \cdot)$ to denote the i-th row of G. This implies:

$$\mathcal{R}_T \le \frac{\ln N}{\eta} + \eta + \eta \sum_{t=2}^{T} \|q_t - q_{t-1}\|_1^2 - \frac{1}{4\eta}\sum_{t=2}^{T} \|p_t - p_{t-1}\|_1^2 . \tag{4}$$

The exact same argument applies to the column player as well, meaning

$$\mathcal{R}'_T \le \frac{\ln N}{\eta} + \eta + \eta \sum_{t=2}^{T} \|p_t - p_{t-1}\|_1^2 - \frac{1}{4\eta}\sum_{t=2}^{T} \|q_t - q_{t-1}\|_1^2 ,$$

[1] At this point, you should revisit one statement we made in Lecture 1: even though the definition of regret seemingly does not make sense at all for an adaptive adversary, it can still be very meaningful.

and thus the sum is bounded as

$$\begin{aligned}
\mathcal{R}_T + \mathcal{R}'_T &\le \frac{\ln(NM)}{\eta} + 2\eta + \left(\eta - \frac{1}{4\eta}\right)\sum_{t=2}^{T}\left(\|p_t - p_{t-1}\|_1^2 + \|q_t - q_{t-1}\|_1^2\right) \\
&= 4\ln(NM) + \frac{1}{2} - \frac{3}{4}\sum_{t=2}^{T}\left(\|p_t - p_{t-1}\|_1^2 + \|q_t - q_{t-1}\|_1^2\right) \qquad (5) \\
&\le 4\ln(NM) + \frac{1}{2}
\end{aligned}$$

where the equality is by plugging in the value of η (which is $1/4$). This completes the proof. □

Hence, using optimistic Hedge significantly speeds up the convergence rate from $\mathcal{O}(1/\sqrt{T})$ to $\mathcal{O}(1/T)$. In other words, to get an ϵ-approximate Nash equilibrium, we only need $\mathcal{O}(1/\epsilon)$ iterations instead of $\mathcal{O}(1/\epsilon^2)$; for example, for $\epsilon = 0.001$, this is an improvement from one million iterations to just one thousand.

We point out that this result is also quite robust. For example, the learning rate does not have to be exactly $1/4$ nor identical for the two players — it is easy to see from the proof that if the row player uses η and the column player uses η', then as long as $4\eta\eta' \le 1$, the regret is at most $\mathcal{O}(\frac{\ln N}{\eta} + \frac{\ln M}{\eta'})$. Moreover, the key of the proof clearly only relies on having an adaptive regret bound of the form (3) and is independent of the details of the algorithm. There are indeed many other algorithms that enjoy a similar bound and thus the same fast convergence rate; see Syrgkanis et al. [2015] for details.

Another interesting phenomenon is the behavior of the strategy sequence $(p_1, q_1), (p_2, q_2), \ldots$. While any no-regret algorithm ensures that the average of these strategies converges, this sequence itself might not converge. For example, the sequence generated by vanilla Hedge could circle around the equilibrium and never gets close to it. On the other hand, the sequence generated by Optimistic Hedge has been proven to converge to the equilibrium in recent years. Such property is often called *last-iterate convergence*, and is another reason why Optimistic Hedge is much more favorable for such problems. Notice that the original goal of the players is not to find the equilibrium, but instead simply to selfishly minimize their own loss by exploiting the weakness of the opponent. However, Nash equilibrium emerges as a natural long-term outcome of this selfish process.

A closer look at the stability. Recall our earlier intuition: in the game setting, the loss path-length $\|\ell_t - \ell_{t-1}\|_\infty^2$ of the row player is bounded by the stability $\|q_t - q_{t-1}\|_1^2$ of the column player, which should be of order η and thus "small". However, in the end, we are in fact choosing a *constant* learning rate $\eta = 1/4$, which seemingly contradicts with our earlier intuition and shows that the both players could be highly unstable instead. Is this truly how these algorithms behave?

It turns out that our earlier intuition is still correct: the algorithms are highly stable, leading to a slowly changing environment for the opponent. To see this, first notice that the social regret is never negative based on Eq. (2). Therefore, rearranging Eq. (5) shows

$$\frac{3}{4}\sum_{t=2}^{T}\left(\|p_t - p_{t-1}\|_1^2 + \|q_t - q_{t-1}\|_1^2\right) \le 4\ln(NM) + \frac{1}{2},$$

and thus $\sum_{t=2}^{T}\|p_t - p_{t-1}\|_1^2 = \mathcal{O}(\ln(NM))$ (same for the column player). This shows that the cumulative movement of the player's strategy, a quantity that could be of order T in the worst case, is in fact completely independent of T. This is also consistent with the earlier point that the sequence $p_1, p_2, \ldots$ is converging (though not a formal proof yet).

This small observation has two important consequences. First, the individual regret of each player ($\mathcal{R}_T$ or $\mathcal{R}'_T$) is in fact also of order $\mathcal{O}(\ln(NM))$. Note that this cannot be simply concluded from the fact $\mathcal{R}_T + \mathcal{R}'_T = \mathcal{O}(\ln(NM))$, since regret can be negative. Instead, one can see this by starting from Eq. (4), ignoring the negative term, and using the stability $\sum_{t=2}^{T}\|q_t - q_{t-1}\|_1^2 = \mathcal{O}(\ln(NM))$ we just proved. Such an individual constant regret bound incentivizes both players to deploy Optimistic Hedge.

Second, this also provides a way to robustify the algorithm. Indeed, the aforementioned good individual regret bound only holds when both players exactly follow Optimistic Hedge, so what if my opponent deviates to something else to exploit my weakness? To address this, ideally we want our algorithm to also enjoy the worst-case $\mathcal{O}(\sqrt{T})$ bound no matter what the opponent actually ends up doing. With a fixed learning rate $\eta = 1/4$, it is not hard to see that Optimistic Hedge does not enjoy such a robustness guarantee. However, this can be easily addressed by the following modification: start with Optimistic Hedge with $\eta = 1/4$, keep track of the path length $\sum_{s \le t} \|\ell_s - \ell_{s-1}\|_\infty^2$, and whenever it exceeds $\mathcal{O}(\ln(NM))$, switch to vanilla Hedge (or any other minimax optimal algorithm). This does not ruin our earlier nice $\mathcal{O}(\ln(NM))$ regret bound if both players follow the same algorithm (since the path length cannot exceed $\mathcal{O}(\ln(NM))$ in this case); on the other hand, no matter what the opponent does, our regret is never worse than $\mathcal{O}(\ln(NM) + \sqrt{T \ln N})$, because before our algorithm switches to vanilla Hedge, its regret is at most $\mathcal{O}(\ln(NM))$ based on Eq. (3). This robustness guarantee further incentivizes both parties to deploy this algorithm.

4 General-Sum Games

Next, we consider general-sum games, a much broader type of games where the players might not have exact opposite interest any more. For simplicity, we still focus on the two-player setting, where Player 1 has N actions and Player 2 has M actions. The loss matrices $G_1 \in [0,1]^{N \times M}$ and $G_2 \in [0,1]^{N \times M}$ are such that $G_1(i,j)$ and $G_2(i,j)$ are the loss for Player 1 and Player 2 respectively if Player 1 picks action i while Player 2 picks action j. The zero-sum setting is clearly a special case with $G_2 = -G_1$.

A classical example is the "game of chicken", where two players are driving toward each other and both can either "Dare" (D) or "Chicken" (C): both dare leads to car crash and the worst case loss of 1 for both; if one dares and other other chickens, the former has no loss and the latter has loss 0.5; finally if both chicken, they both get a small loss 0.1. This simple game captures numerous real-life situations (e.g. nuclear arms race between counties). The loss matrices G_1 and G_2 in this game are:

$$G_1 = \begin{array}{c c} & \begin{array}{cc} D & C \end{array} \\ \begin{array}{c} D \\ C \end{array} & \begin{pmatrix} 1 & 0 \\ 0.5 & 0.1 \end{pmatrix} \end{array}, \qquad G_2 = \begin{array}{c c} & \begin{array}{cc} D & C \end{array} \\ \begin{array}{c} D \\ C \end{array} & \begin{pmatrix} 1 & 0.5 \\ 0 & 0.1 \end{pmatrix} \end{array}.$$

Nash equilibria can be defined in the same way for general-sum games: a pair of mixed strategy p and q is a Nash equilibrium if neither player has incentive to deviate:

$$G_1(p,q) \le G_1(i,q),\ \forall i \in [N] \quad \text{and} \quad G_2(p,q) \le G_2(p,j),\ \forall j \in [M].$$

However, in this case there is no "minimax" interpretation of Nash equilibria or the corresponding unique game value any more. For example, the game of chicken has three Nash equilibria: two deterministic ones where one player always dares and the other always chickens, and a mixed one where both players dare with probability $1/6$ (verify it yourself). Each of these equilibria leads to different losses for the two players, illustrating very different characteristics of Nash equilibria for general-sum games. In fact, finding Nash equilibria for general-sum games has also been shown to belong to a complexity class called PPAD (believed to be computationally hard), so there is basically no hope that our earlier discussions on the connection between no-regret learning and finding Nash equilibria can be extended to this case.

Because of this hardness result, weaker notions of equilibria have been studied. Here, we only discuss one of them: *coarse correlated equilibria* (CCE), which is a direct generalization of Nash equilibria from product distributions over action pairs to joint distributions. Formally, a joint distribution $\sigma \in \Delta(NM)$ is a CCE if neither player has incentive to deviate assuming that the opponent sticks with σ:

$$\mathbb{E}_{(i,j)\sim\sigma}[G_1(i,j)] \le G_1(i',\sigma_2),\ \forall i' \in [N] \quad \text{and} \quad \mathbb{E}_{(i,j)\sim\sigma}[G_2(i,j)] \le G_2(\sigma_1,j'),\ \forall j' \in [M],$$

where σ_1 and σ_2 are the marginal distributions over Player 1's actions and Players 2's actions respectively. If a CCE σ happens to be a product distribution, that is $\sigma(i,j) = \sigma_1(i)\sigma_2(j)$, then clearly σ is also a Nash equilibrium by definition. For a general CCE σ that is not a product distribution, one way to interpret it is that if a mediator recommends to both players an action pair drawn from σ, then both players feel rational about accepting the recommendation even without looking at it.

For example, in Rock-Paper-Scissors, a uniform distribution over all the six non-tie action pairs is a CCE. For the game of chicken, picking (C,C) with $5/7$ probability, and (D,C) or (C,D) with probability $1/7$ is a CCE. In fact, this CCE leads to even smaller total losses (that is, higher social welfare) of the two players compared to all the Nash equilibria. (Verify all these yourself.)

4.1 Finding CCEs via No-regret Learning

While Nash equilibria are hard to find, CCEs can be computed efficiently, and in fact can be done via no-regret learning in an *uncoupled* way again, even though CCE, as a joint distribution, is by definition coupled. The learning setup is the same as before: each time Player 1 (respectively Player 2) independently uses an expert algorithm to come up with an action distributions $p_t \in \Delta(N)$ (respectively $q_t \in \Delta(M)$), and then sees the loss vector $G_1(\cdot, q_t)$ (respectively $G_2(p_t, \cdot)$) to be fed to the expert algorithm. This is uncoupled in the sense that both players are simply maintaining a distribution over their own actions, and more importantly, they only need to see their own losses (for example, Player 1 is completely oblivious about G_2). Nevertheless, it can be shown that their average joint behavior converges to a CCE in the following sense.

Theorem 4. *In the learning setup above, if $\mathcal{R}_T$ and $\mathcal{R}'_T$ are the regret of Player 1 and Player 2 respectively, then the joint distribution $\sigma \in \Delta(NM)$ with $\sigma(i,j) = \frac{1}{T}\sum_{t=1}^{T} p_t(i)q_t(j)$ (that is, uniform over the empirical mixed strategy pairs) is a $\max\{\mathcal{R}_T/T, \mathcal{R}'_T/T\}$-approximate CCE.*

Proof. The proof is simply by definition. First observe that the marginal distribution σ_1 and σ_2 are simply $\frac{1}{T}\sum_{t=1}^{T} p_t$ and $\frac{1}{T}\sum_{t=1}^{T} q_t$. For Player 1, by definition we have:

$$\mathbb{E}_{(i,j)\sim\sigma}[G_1(i,j)] = \sum_{i,j}\sigma(i,j)G_1(i,j) = \frac{1}{T}\sum_{t=1}^{T}\sum_{i,j} p_t(i)q_t(j)G_1(i,j) = \frac{1}{T}\sum_{t=1}^{T} G_1(p_t,q_t)$$

$$= \min_p \frac{1}{T}\sum_{t=1}^{T} G_1(p,q_t) + \frac{\mathcal{R}_T}{T} = \min_p G(p,\sigma_2) + \frac{\mathcal{R}_T}{T} \le \min_p G(p,\sigma_2) + \max\left\{\frac{\mathcal{R}_T}{T}, \frac{\mathcal{R}'_T}{T}\right\},$$

and similarly for Player 2, we have

$$\mathbb{E}_{(i,j)\sim\sigma}[G_2(i,j)] = \min_q G_2(\sigma_1,q) + \frac{\mathcal{R}'_T}{T} \le \min_q G_2(\sigma_1,q) + \max\left\{\frac{\mathcal{R}_T}{T}, \frac{\mathcal{R}'_T}{T}\right\}.$$

Together, this is exactly the definition of σ being a $\max\{\mathcal{R}_T/T, \mathcal{R}'_T/T\}$-approximate CCE. □

This result again provides a highly efficient and uncoupled way to find an approximate CCE, as a long-term outcome of a selfish process where each player's motivation is simply to minimize their own regret. Due to the uncouple nature, the time/space complexity of such algorithms is only in terms of $N + M$, while other approaches (such as those based on linear programming) would depend on NM instead (after all, a joint distribution in $\Delta(NM)$ is by definition an objective with NM dimension). Generalizing this to an arbitrary number of players, say k players each with N actions, then the difference discussed above becomes kN versus N^k, demonstrating an exponential improvement on the number of players for the no-regret learning approach!

Fast convergence. When both players use Optimistic Hedge, one can easily verify that the constant social regret bound of Theorem 3 still holds in this case. However, unlike the zero-sum case, the CCE approximation error is now in terms of $\max\{\mathcal{R}_T, \mathcal{R}'_T\}$, not $\mathcal{R}_T + \mathcal{R}'_T$, so once again we care about the individual regret but not the regret sum. This is where things can get much more complicated — indeed, the earlier argument on getting $\max\{\mathcal{R}_T, \mathcal{R}'_T\} = \mathcal{O}(\ln(NM))$ in the zero-sum case crucially relies on the fact $\mathcal{R}_T + \mathcal{R}'_T \ge 0$, which no longer holds for general-sum games!

In fact, getting $\max\{\mathcal{R}_T, \mathcal{R}'_T\} = \mathcal{O}(\ln(NM))$ in this case is still an open problem, but recent research has made significant progress, with the state-of-the-art bound being $\mathcal{O}(\ln(NM)\ln^4 T)$ by Daskalakis et al. [2021] (using exactly Optimistic Hedge) and $\mathcal{O}(\ln^2(NM)\ln T)$ by Soleymani et al. [2025] (using a variant of Optimistic Hedge). These results again make use of some adaptive property of Optimistic Hedge similar to our discussion, but are too involved to be covered here. Instead, in the rest of this lecture, we discuss two simpler results that achieve $\max\{\mathcal{R}_T, \mathcal{R}'_T\} = o(\sqrt{T})$, taken from earlier work Syrgkanis et al. [2015] and Chen and Peng [2020] respectively.

The first result again uses the fact that the environment is stable due to the stability of learning algorithms.

Theorem 5. *If both players use Optimistic Hedge with the predictor being the last loss vector:*

$$p_t(i) \propto \exp\left(-\eta\left(\ell_{t-1}(i) + \sum_{s<t} \ell_s(i)\right)\right), \quad \textit{where } \ell_s(i) = G_1(i, q_s),$$

$$q_t(j) \propto \exp\left(-\eta\left(g_{t-1}(j) + \sum_{s<t} g_s(j)\right)\right), \quad \textit{where } g_s(j) = G_2(p_s, j).$$

then with $\eta = \left(\ln(NM)/T\right)^{1/4}$, $\max\{\mathcal{R}_T, \mathcal{R}'_T\} = \mathcal{O}(T^{1/4}\ln^{3/4}(NM))$.

Proof. By Lemma 4 of Lecture 2, we have $\|q_t - q_{t-1}\|_1 \le \eta \|2g_{t-1} - g_{t-2}\|_\infty \le 2\eta$, and thus based on Eq (3) and our earlier observation $\|\ell_t - \ell_{t-1}\|_\infty \le \|q_t - q_{t-1}\|_1$, we have

$$\mathcal{R}_T \le \frac{\ln N}{\eta} + \eta \sum_{t=1}^{T} \|\ell_t - \ell_{t-1}\|_\infty^2 \le \frac{\ln N}{\eta} + \eta + \eta \sum_{t=2}^{T} \|q_t - q_{t-1}\|_1^2 \le \frac{\ln N}{\eta} + \eta + 4T\eta^3.$$

The same bound holds for Player 2 (with N replaced by M). Plugging in the (optimal) learning rate value thus proves the theorem. □

With a more careful treatment of the stability, the result above can be improved to the following.

Theorem 6. *For the same algorithm described in Theorem 5, using* $\eta = \left(\ln(NM)/T\right)^{1/6}$ *ensures* $\max\{\mathcal{R}_T, \mathcal{R}'_T\} = \mathcal{O}(T^{1/6}\ln^{5/6}(NM))$.

Proof. Instead of using the final statement of Lemma 4 of Lecture 2, we use the intermediate step (Eq. (3) of Lecture 2): $\|q_t - q_{t-1}\|_1^2 \le \eta \langle q_t - q_{t-1}, g_{t-2} - 2g_{t-1}\rangle$, which further implies

$$\begin{aligned}
&\sum_{t=2}^{T} \|q_t - q_{t-1}\|_1^2 \le \eta \sum_{t=2}^{T} \langle q_t - q_{t-1}, g_{t-2} - 2g_{t-1}\rangle \\
&= \eta \langle q_T, g_{T-2} - 2g_{T-1}\rangle - \eta \langle q_1, g_0 - 2g_1\rangle + \eta \sum_{t=2}^{T-1} \langle q_t, 2g_t - 3g_{t-1} + g_{t-2}\rangle \qquad \text{(rearranging)} \\
&\le \eta \|g_{T-2} - 2g_{T-1}\|_\infty + \eta \|g_0 - 2g_1\|_\infty + \eta \sum_{t=2}^{T-1} \|2g_t - 3g_{t-1} + g_{t-2}\|_\infty \\
&\le 4\eta + \eta \sum_{t=2}^{T-1} \left(2\|g_t - g_{t-1}\|_\infty + \|g_{t-1} - g_{t-2}\|_\infty\right) \\
&\le 5\eta + 3\eta \sum_{t=2}^{T-1} \|g_t - g_{t-1}\|_\infty \le 5\eta + 3\eta \sum_{t=2}^{T-1} \|p_t - p_{t-1}\|_1 .
\end{aligned}$$

Note that we have moved from the (squared) stability of q_t, to the stability of g_t, and finally back to the stability of p_t. Therefore, starting from the regret bound (3), we now have

$$\begin{aligned}
\mathcal{R}_T &\le \frac{\ln N}{\eta} + \eta \sum_{t=1}^{T} \|\ell_t - \ell_{t-1}\|_\infty^2 - \frac{1}{4\eta} \sum_{t=2}^{T} \|p_t - p_{t-1}\|_1^2 \\
&\le \frac{\ln N}{\eta} + \eta + \eta \sum_{t=2}^{T} \|q_t - q_{t-1}\|_1^2 - \frac{1}{4\eta} \sum_{t=2}^{T} \|p_t - p_{t-1}\|_1^2 \\
&\le \frac{\ln N}{\eta} + \eta + 5\eta^2 + 3\eta^2 \sum_{t=2}^{T} \|p_t - p_{t-1}\|_1 - \frac{1}{4\eta} \sum_{t=2}^{T} \|p_t - p_{t-1}\|_1^2 \\
&\le \frac{\ln N}{\eta} + \eta + 5\eta^2 + 9T\eta^5,
\end{aligned}$$

where the last step is due to $3\eta^2 \|p_t - p_{t-1}\|_1 \le \frac{1}{4\eta} \|p_t - p_{t-1}\|_1^2 + 9\eta^5$ since $2\sqrt{ab} \le a + b$. Plugging in the learning rate then finishes the proof (the reasoning for $\mathcal{R}'_T$ is symmetric). □

References

Xi Chen and Binghui Peng. Hedging in games: Faster convergence of external and swap regrets. *Advances in Neural Information Processing Systems*, 33:18990–18999, 2020.

Constantinos Daskalakis, Maxwell Fishelson, and Noah Golowich. Near-optimal no-regret learning in general games. *Advances in Neural Information Processing Systems*, 34:27604–27616, 2021.

Yoav Freund and Robert E Schapire. Adaptive game playing using multiplicative weights. *Games and Economic Behavior*, 29(1-2):79–103, 1999.

Ashkan Soleymani, Georgios Piliouras, and Gabriele Farina. Cautious optimism: A meta-algorithm for near-constant regret in general games. *Economics and Computation*, 2025.

Vasilis Syrgkanis, Alekh Agarwal, Haipeng Luo, and Robert E Schapire. Fast convergence of regularized learning in games. In *Advances in Neural Information Processing Systems 28*, 2015.