

Supplementary reading S1

A second look at the minimax theorem

Instructor: Prof. Gabriele Farina

In Lecture 3, we introduced the notion of coarse correlated equilibria. As we discussed, coarse correlated equilibria sidestep various difficulties (including topological and related to use of irrational numbers) that come with Nash equilibria. In this lecture, we show a powerful centralized algorithm for computing coarse correlated equilibria. (Soon in this course, we will also see that coarse correlated equilibria can also be *learned* efficiently in a multi-agent setting, in a distributed fashion.)

We have seen at the end of Lecture 3 that a coarse correlated equilibrium can be computed by solving a linear program, in which the variables correspond to the probabilities $\mu_{a_1, \dots, a_n}$ of the joint actions, and the constraints correspond to the incentive constraints of the players. While this is a perfectly valid way to compute a coarse correlated equilibrium, it has the drawback that the linear program has a number of variables that is exponential in the number of players. This becomes an issue quickly, if we want to consider games with many players. It also is a problem for those games in which the payoff tensor has a succinct representation (for example, a sparse factorization that we can exploit); there, we would ideally want an algorithm that runs in polynomial time in the size of such a succinct representation. The latter is often the case in structured games, which we will see later in this course.

In this lecture, we will see a different algorithm for computing coarse correlated equilibria, which does not suffer from the above issues and requires a number of variables that scales with the *sum* (rather than *product*!) of the number of actions of the players. The algorithm is called Ellipsoid-Against-Hope, and was introduced by [Papadimitriou, C. H., & Roughgarden, T. \[PR08\]](#), with later extensions by other authors [\[FP24; HS08; JL11\]](#).

The algorithm is based on a constructive proof of the minimax theorem, which we will also present in this lecture.

S1.1 Revisiting the existence of coarse correlated equilibria

In order to understand why there is hope to compute coarse correlated equilibria more efficiently, it is useful to understand better how we can prove that these equilibria exist in the first place. We will then turn such an existence proof into a computational algorithm.

So far, we have justified the existence of coarse correlated and correlated equilibria through the existence of Nash equilibria. However, one might wonder if there is a more direct way to prove the existence of CEs and CCEs, which does not rely on the existence of a much harder notion. The answer is yes, and the idea comes from a very neat proof by [Hart, S., & Schmeidler, D. \[HS89\]](#), which includes some ideas that will set the stage for the Ellipsoid-Against-Hope algorithm.

While the original proof of [Hart, S., & Schmeidler, D. \[HS89\]](#) is for CE, I will present here a version of the proof simplified for the case of CCEs.

As a reminder, by definition a coarse correlated equilibrium is a distribution $\mu \in \Delta(A_1 \times \dots \times A_n)$ such that

$$\mathbb{E}_{a \sim \mu}[u_i(a'_i, a_{-i})] \leq \mathbb{E}_{a \sim \mu}[u_i(a_i, a_{-i})] \quad \forall i \in [n], a'_i \in A_i,$$

or equivalently,

$$\max_{\substack{i \in [n] \\ a'_i \in A_i}} \mathbb{E}_{a \sim \mu}[u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0.$$

A CCE then exists if and only if

$$\min_{\mu} \max_{\substack{i \in [n] \\ a'_i \in A_i}} \mathbb{E}_{a \sim \mu}[u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0.$$

How can we prove the above inequality without resorting to the existence of Nash equilibria? *The rescue comes from the minimax theorem.*

Before we can use the minimax theorem, we have to “convexify” the inner problem however, since the maximum is currently on a discrete set. To convexify the problem, we will simply allow the possibility for the internal maximization problem to propose a *distribution* ν over deviations (i, a_i) , and we will rewrite the problem as

$$\min_{\mu} \max_{\nu} \mathbb{E}_{a \sim \mu} \mathbb{E}_{(i, a'_i) \sim \nu}[u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0.$$

By the minimax theorem and swapping the order of the expectations, the above min-max value is equal to

$$\max_{\nu} \min_{\mu} \mathbb{E}_{(i, a'_i) \sim \nu} \mathbb{E}_{a \sim \mu}[u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})].$$

Can we show that this value is ≤ 0 ? The answer is yes, and constructive: given any ν , we can find a μ in closed form—in fact, a *product* distribution—such that the value is ≤ 0 .

Theorem S1.1 ([Hart, S., & Schmeidler, D. \[HS89\]](#)). Given any distribution ν over pairs $(i, a'_i) : i \in [n], a'_i \in A'_i$, we can explicitly and efficiently construct a product distribution $\mu \in \Delta(A_1) \otimes \dots \otimes \Delta(A_n)$ such that

$$\mathbb{E}_{(i, a'_i) \sim \nu} \mathbb{E}_{a \sim \mu}[u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0.$$

The above theorem immediately implies that

$$\max_{\nu} \min_{\mu} \mathbb{E}_{(i, a'_i) \sim \nu} \mathbb{E}_{a \sim \mu} [u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0,$$

and using the minimax theorem, we conclude the existence of coarse correlated equilibria.

■ S1.2 Turning the minimax theorem into an efficient algorithm

Like what we did for the Nash equilibrium in Lecture 2, it is worth inspecting where the “magic” happens in the above proof. If we squint our eyes a bit, the argument of the proof looked like this:

1. We want to prove that $\min_{\mu} \max_{\nu} g(\mu, \nu) \leq 0$, for an appropriate g that is linear in both μ and ν . That is, we want to show that there *exists* a μ such that *for all* ν , $g(\mu, \nu) \leq 0$.
2. To do that, we instead show that *for all* ν , there *exists* a μ (dependent on ν) such that $g(\mu, \nu) \leq 0$. This shows that $\max_{\nu} \min_{\mu} g(\mu, \nu) \leq 0$.
3. Then, we use the minimax theorem to swap the order of the quantifiers, and conclude that $\min_{\mu} \max_{\nu} g(\mu, \nu) \leq 0$. The use of the minimax theorem in our case was justified because $g(\mu, \nu)$ is linear in both μ and ν .

It is easy to brush away swapping the order of the quantifiers as just one of the many results in mathematics that are concerned with swapping orders of operators. But let us stop to consider how powerful this is. The original problem was to find a single μ^* that works for all ν (Step 1). However, the minimax theorem (Step 3) tells us that as long as for any specific ν we can construct a $\mu(\nu)$ that works *for that ν specifically*, then a μ^* that works for any ν (not for one specific) must exist (Step 2). It feels way less simple than it looks at first sight, right?

The Ellipsoid-Against-Hope algorithm can then be seen as a way to convert the minimax theorem from a tool guaranteeing existence into a computational algorithm. In particular, the idea is the following:

- We will query a few “well-chosen” distributions ν_t , and for each of them, construct the corresponding $\mu(\nu_t)$ that works for that specific ν_t . The number of queries will be small, polynomial in the sum of the number of actions of the players and $\log(1/\epsilon)$.
- Then, we will combine all the $\mu(\nu_t)$ we have constructed into a single μ^* that works for all ν . In particular, the μ^* will be a convex combination of the $\mu(\nu_t)$ we have constructed, with coefficients that can be computed efficiently by solving a linear program with a number of variables that is again polynomial in the sum of the number of actions of the players and $\log(1/\epsilon)$.

Combining the two steps above, we will have constructed a μ^* that is an ϵ -coarse correlated equilibrium, and that can be represented as a convex combination of product distributions. In other words, we have shown the following corollary.

Corollary S1.2. There exists an algorithm that computes an ϵ -coarse correlated equilibrium in time polynomial in the sum of the number of actions of the players and $\log(1/\epsilon)$.

Such a coarse correlated equilibrium is represented as a convex combination of product distributions.

■ S1.2.1 Sketch of the Ellipsoid-Against-Hope algorithm

In more detail, what Theorem S1.1 implies is that the following open polytope must be empty:

$$\left\{ \nu \in \Delta\{(i, a'_i) : i \in [n], a_i \in A_i\} : \mathbb{E}_{(i, a'_i) \sim \nu} \mathbb{E}_{a \sim \mu} [u_i(a_i, a_{-i}) - u_i(a'_i, a_{-i})] > 0 \right. \\ \left. \forall \mu \in \Delta(A_1 \times \dots \times A_n) \right\}.$$

Furthermore, for any ν , we know how to prove that at least one of the constraints is violated. The key idea is then to use the ellipsoid method to *certify* the emptiness of the polytope. Normally, the ellipsoid method is used to find a point in a set, but in our case, the point does not exist and we want to use the ellipsoid method to isolate constraints that prove the emptiness of the set. For this reason, the algorithm was called Ellipsoid-Against-Hope by Papadimitriou, C. H., & Roughgarden, T. [PR08].

The ellipsoid will maintain a search space which can be thought of as a suitable subset of the deviator's set. At every iteration t , the algorithm will compute the center point ν_t of the set. Then, it will find a violated constraint using the distribution $\mu_t := \mu(\nu_t)$ in the proof of Theorem S1.1. The violated constraint implies that the deviator set must be curtailed, and the ellipsoid will be updated accordingly reducing the size of the search space by a constant. The algorithm will continue until the search space is small enough to guarantee that the set is empty. In the process, it takes $O(\log(1/\epsilon))$ iterations for the search space to shrink to size ϵ . By the last iteration T , the algorithm will have produced several violated constraints, each of which is associated with a mediator strategy μ_t . The set

$$\left\{ \nu \in \Delta\{(i, a_i) : i \in [n], a_i \in A_i\} : \mathbb{E}_{(i, a'_i) \sim \nu} \mathbb{E}_{a \sim \mu_1} [u_i(a_i, a_{-i}) - u_i(a'_i, a_{-i})] > 0 \right. \\ \vdots \\ \left. \mathbb{E}_{(i, a'_i) \sim \nu} \mathbb{E}_{a \sim \mu_T} [u_i(a_i, a_{-i}) - u_i(a'_i, a_{-i})] > 0 \right\}.$$

The constraints of the set are all linear in ν , and the set is empty. By Farkas' lemma, there must exist a convex combination of the constraints that makes all the coefficients on the left-hand side non-positive. In other words, there must exist $\alpha_1, \dots, \alpha_T \geq 0$ such that

$$\sum_{t=1}^T \alpha_t \mathbb{E}_{a \sim \mu_t} [u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0, \quad \forall i \in [n], a'_i \in A_i.$$

Letting $\bar{\mu} := \sum_{t=1}^T \alpha_t \mu_t$, we can then write

$$\mathbb{E}_{a \sim \bar{\mu}} [u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0, \quad \forall i \in [n], a'_i \in A_i,$$

and hence $\bar{\mu}$ is a coarse correlated equilibrium. The only question is whether this combination $\{\alpha_t\}$ can be computed efficiently. This is indeed the case, as we can use linear programming directly to find such a combination, by solving the feasibility program

$$\begin{aligned}
& \text{find } \alpha_1, \dots, \alpha_T \geq 0 \\
& \text{s.t. } \sum_{t=1}^T \alpha_t \mathbb{E}_{a \sim \mu_t} [u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \leq 0 \quad \forall i \in [n], a'_i \in A_i \\
& \alpha_1 + \dots + \alpha_T = 1.
\end{aligned}$$

This completes the sketch of the proof of the correctness of the Ellipsoid-Against-Hope algorithm.

■ S1.2.2 Applications beyond normal-form games

The above argument mostly uses ideas from convex optimization. In particular, it generalizes verbatim to any *convex game*, that is, any setting with the following properties:

- Player i 's strategy set is a convex set of some dimension d_i . For normal-form games, a strategy is an element of $\Delta(A_i)$, that is, a distribution over the player's strategies. We assume we have oracle access to the set, which runs in time polynomial in d_i .
- The utility function $u_i(x_1, \dots, x_n)$ is linear in each player's strategy. For normal-form games, this is true since the utility is just an expectation.
- The utility function $u_i(x_1, \dots, x_n)$ can be evaluated efficiently, let's say in time R .

The Ellipsoid-Against-Hope algorithm can then be applied and runs in time polynomial in $\sum_i^n d_i$, R , and $\log(1/\epsilon)$. Examples of games that satisfy the above properties include polymatrix games, congestion games, and sequential imperfect-information (extensive-form) games. We will see some of these games later in this course.

■ S1.3 Bibliographic remarks

If you are curious to read more, the following papers contains extensions and refinements of the idea of Ellipsoid-Against-Hope.

- [PR08] C. H. Papadimitriou and T. Roughgarden, "Computing correlated equilibria in multi-player games," *Journal of the ACM (JACM)*, vol. 55, no. 3, pp. 1–29, 2008.
- [JL11] A. X. Jiang and K. Leyton-Brown, "Polynomial-time computation of exact correlated equilibrium in compact games," in *Proceedings of the 12th ACM conference on Electronic commerce*, 2011, pp. 119–126.
- [HS08] W. Huang and B. von Stengel, "Computing an extensive-form correlated equilibrium in polynomial time," in *International Workshop on Internet and Network Economics*, 2008, pp. 506–513.
- [FP24] G. Farina and C. Pipis, "Polynomial-Time Computation of Exact Φ -Equilibria in Polyhedral Games", in *Neural Information Processing Systems (NeurIPS)*, 2024.
- [HS89] S. Hart and D. Schmeidler, "Existence of Correlated Equilibria," *Mathematics of Operations Research*, vol. 14, no. 1, pp. 18–25, 1989, Accessed: Nov. 07, 2024. [Online]. Available: <http://www.jstor.org/stable/3689835>

S1.A Appendix: Proof of Theorem S1.1

The key here is to pick μ in a way that depends on ν . In particular, we will pick μ to be the *product* distribution that outputs

$$(a_1, \dots, a_n) \text{ with probability proportional to } \nu_{1,a_1} \cdot \dots \cdot \nu_{n,a_n}.$$

With this choice and some simple manipulations,

$$\begin{aligned} & \mathbb{E}_{(i,a'_i) \sim \nu} \mathbb{E}_{a \sim \mu} [u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \\ &= \sum_i \sum_{a'_i \in A_i} \nu_{i,a'_i} \mathbb{E}_{a \sim \mu} [u_i(a'_i, a_{-i}) - u_i(a_i, a_{-i})] \\ &= \sum_i \sum_{a'_i \in A_i} \nu_{i,a'_i} (\mathbb{E}_{a \sim \mu} [u_i(a'_i, a_{-i})] - \mathbb{E}_{a \sim \mu} [u_i(a_i, a_{-i})]) \\ &= \sum_i \left(\sum_{a'_i \in A_i} \nu_{i,a'_i} \mathbb{E}_{a \sim \mu} [u_i(a'_i, a_{-i})] - \sum_{a'_i \in A_i} \nu_{i,a'_i} \mathbb{E}_{a \sim \mu} [u_i(a_i, a_{-i})] \right) \\ &= \sum_i \left(\underbrace{\sum_{a'_i \in A_i} \nu_{i,a'_i} \mathbb{E}_{a \sim \mu} [u_i(a'_i, a_{-i})]}_{(\spadesuit)} - \underbrace{\left(\sum_{a'_i \in A_i} \nu_{i,a'_i} \right) \mathbb{E}_{a \sim \mu} [u_i(a_i, a_{-i})]}_{(\clubsuit)} \right). \end{aligned}$$

We now expand $(\spadesuit)$, using the symbol $\nu_{-i,a_{-i}}$ to mean the product of $\nu_{1,a_1}, \dots, \nu_{i-1,a_{i-1}}, \nu_{i+1,a_{i+1}}, \dots, \nu_{n,a_n}$. Using the construction of μ as a product distribution, we can write

$$\begin{aligned} (\spadesuit) &= \sum_{a'_i \in A_i} \nu_{i,a'_i} \sum_{a_i \in A_i} \sum_{a_{-i} \in A_{-i}} \nu_{i,a_i} \nu_{-i,a_{-i}} u_i(a_i, a_{-i}) \\ &= \sum_{a'_i \in A_i} \nu_{i,a'_i} \left(\left(\sum_{a_{-i} \in A_{-i}} \nu_{-i,a_{-i}} u_i(a_i, a_{-i}) \right) \left(\sum_{a_i \in A_i} \nu_{i,a_i} \right) \right) \\ &= \left(\sum_{a'_i \in A_i} \nu_{i,a'_i} \sum_{a_{-i} \in A_{-i}} \nu_{-i,a_{-i}} u_i(a_i, a_{-i}) \right) \left(\sum_{a_i \in A_i} \nu_{i,a_i} \right) \\ &= \left(\sum_{a'_i \in A_i} \sum_{a_{-i} \in A_{-i}} \nu_{i,a'_i} \nu_{-i,a_{-i}} u_i(a_i, a_{-i}) \right) \left(\sum_{a_i \in A_i} \nu_{i,a_i} \right) = (\clubsuit). \end{aligned}$$

Hence,

$$\max_{\nu} \min_{\mu} \mathbb{E}_{(i,a'_i) \sim \nu} \mathbb{E}_{a \sim \mu} [u_i(a_i, a_{-i}) - u_i(a_i, a_{-i})] \leq \max_{\nu} 0 = 0,$$

completing the proof.