

Learning in General-Sum Games: Correlated Equilibria and Phi-Regret



15 888 **Computational Game Solving** (Fall 2026)
Ioannis Anagnostides

Today's lecture

- **Correlated and coarse correlated equilibrium**
 - Interpretation and examples
 - Computational properties
- **Phi-regret**
 - Connections to correlated equilibria
 - Swap regret versus external
- **A framework for minimizing Phi-regret**
 - Reducing Phi-regret to external regret
 - Application to swap regret in normal-form games

Criticism of Nash equilibria

- Nash equilibria are amenable to linear programming in zero-sum games, but they are **hard to compute** in general-sum games
- Besides intractability, there is also the *equilibrium selection* problem—a game can have multiple disparate equilibria
- We don't expect Nash equilibria to arise as the limit points of simple, computationally bounded, algorithms such as regret matching
- But what are these no-regret algorithms converging to?

Correlated distributions

- A key premise in the Nash equilibrium is that players are *randomizing independently*
- What if they don't?
- What if the distribution of outcomes is **correlated**?

$$\mu = \begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix}, \quad \mu' = \begin{bmatrix} 1/6 & 1/6 \\ 1/3 & 1/3 \end{bmatrix}$$

Not a product
distribution!

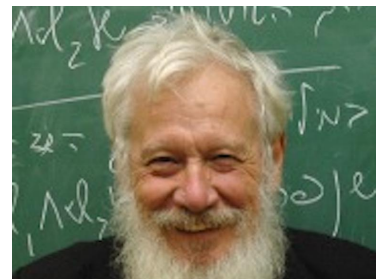
Coarse correlated equilibrium

Definition 1.1 (Coarse correlated equilibrium). A correlated distribution $\mu \in \Delta(\mathcal{A}_1 \times \cdots \times \mathcal{A}_n)$ is an ϵ -coarse correlated equilibrium (CCE) if for any player $i \in [n]$ and deviation $a'_i \in \mathcal{A}_i$,

$$\mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(a_1, \dots, a_n) \geq \mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(a'_i, a_{-i}) - \epsilon.$$

- Under that correlated distribution, no unilateral deviation makes the player better off
- Similar to the Nash equilibrium definition, but *enables correlation*
- A Nash equilibrium is a CCE—an uncorrelated CCE

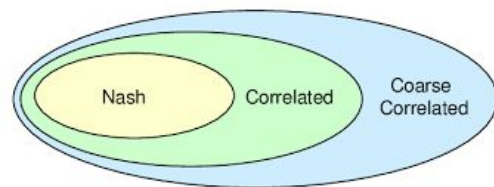
Correlated equilibrium



Definition 1.2 (Correlated equilibrium). A correlated distribution $\mu \in \Delta(\mathcal{A}_1 \times \cdots \times \mathcal{A}_n)$ is an ϵ -correlated equilibrium (CE) if for any player $i \in [n]$ and deviation function $\phi_i : \mathcal{A}_i \rightarrow \mathcal{A}_i$,

$$\mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(a_1, \dots, a_n) \geq \mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(\phi_i(a_i), a_{-i}) - \epsilon.$$

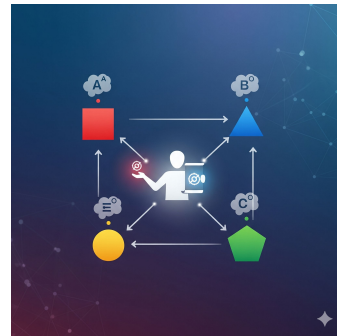
- The class of deviations in CEs is richer than CCEs
- CE is a stronger notion: any CE is also a CCE



By Jason Marden

Interpretation via a mediator or correlation device

- Let's say we have a trusted third-party or mediator
- It first samples a *joint action*
- It recommends to each player the corresponding action from that sample
- In equilibrium, *no player has an incentive to deviate from the recommendation*



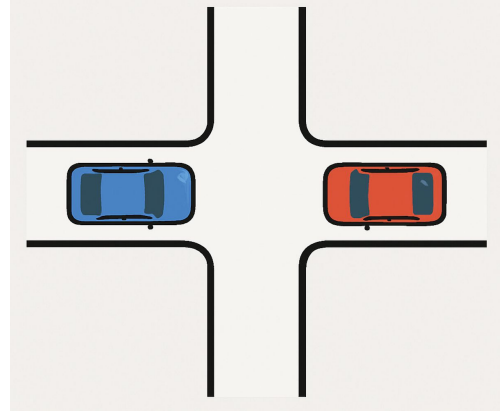
In a CCE, player decides whether to follow *before* seeing the recommendation



Hard to enforce without a binding mechanism

An example: the game of chicken

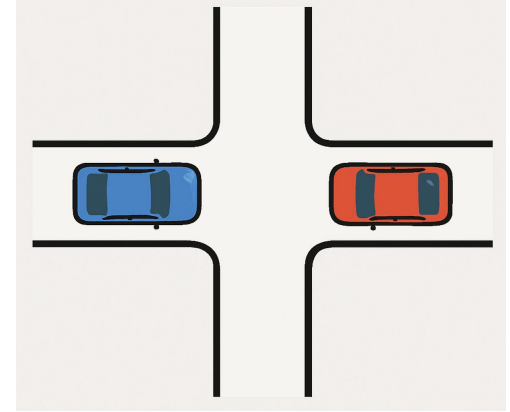
- Two drivers are fast approaching an intersection
- Each has the option of stopping or going
- Nash equilibria are unsatisfactory in this case
 - Either it's unfair for one of the players,
 - Or there is chance of a crash



	Stop	Go
Stop	0, 0	0, 1
Go	1, 0	-5, -5

An example: the game of chicken

- Two drivers are fast approaching an intersection
- Each has the option of stopping or going
- Nash equilibria are unsatisfactory in this case
 - Either it's unfair for one of the players,
 - Or there is chance of a crash
- CCEs and CE unlock new, better outcomes: we can mix between (Stop, Go) and (Go, Stop)



This can be interpreted as a traffic light that correlates the actions of the players



	Stop	Go
Stop	0, 0	0, 1
Go	1, 0	-5, -5

Another example: correlated vs coarse correlated equilibria

- Let's say we have a bimatrix game with four actions
- We are mixing uniformly between (1,1) and (2,2); this is not a product
- Is this distribution a CCE? Is it a CE?

$$\mathbf{R} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 3 & 0 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{bmatrix} \text{ and } \mathbf{C} = \begin{bmatrix} 2 & 0 & 3 & 0 \\ 0 & 2 & 0 & 3 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Computation

Definition 1.2 (Correlated equilibrium). A correlated distribution $\mu \in \Delta(\mathcal{A}_1 \times \cdots \times \mathcal{A}_n)$ is an ϵ -correlated equilibrium (CE) if for any player $i \in [n]$ and deviation function $\phi_i : \mathcal{A}_i \rightarrow \mathcal{A}_i$,

$$\mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(a_1, \dots, a_n) \geq \mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(\phi_i(a_i), a_{-i}) - \epsilon.$$

Theorem. There is a *linear program* that describes the set of (C)CEs.

How many variables?

How many constraints?

Computation

Definition 1.2 (Correlated equilibrium). A correlated distribution $\mu \in \Delta(\mathcal{A}_1 \times \cdots \times \mathcal{A}_n)$ is an ϵ -correlated equilibrium (CE) if for any player $i \in [n]$ and deviation function $\phi_i : \mathcal{A}_i \rightarrow \mathcal{A}_i$,

$$\mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(a_1, \dots, a_n) \geq \mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(\phi_i(a_i), a_{-i}) - \epsilon.$$

Theorem. There is a *linear program* that describes the set of (C)CEs.

How many variables?

- A correlated distribution grows *exponentially* with the number of players!
- It only works for a small number of players

How many constraints?

Computation

Definition 1.2 (Correlated equilibrium). A correlated distribution $\mu \in \Delta(\mathcal{A}_1 \times \cdots \times \mathcal{A}_n)$ is an ϵ -correlated equilibrium (CE) if for any player $i \in [n]$ and deviation function $\phi_i : \mathcal{A}_i \rightarrow \mathcal{A}_i$,

$$\mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(a_1, \dots, a_n) \geq \mathbb{E}_{(a_1, \dots, a_n) \sim \mu} u_i(\phi_i(a_i), a_{-i}) - \epsilon.$$

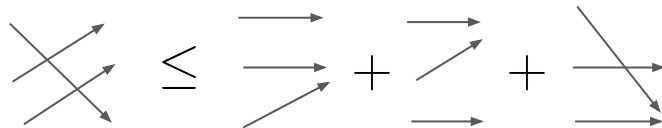
Theorem. There is a *linear program* that describes the set of (C)CEs.

How many variables?

- A correlated distribution grows *exponentially* with the number of players!
- It only works for a small number of players

How many constraints?

- There are exponentially many *swap deviations*
- But it's enough to consider only ones that change a single action



Phi-regret

A more general measure of performance in the online learning setting

$$\Phi\text{Reg}^{(T)} := \max_{\phi \in \Phi} \left\{ \sum_{t=1}^T \langle \phi(\mathbf{x}^{(t)}), \mathbf{u}^{(t)} \rangle \right\} - \sum_{t=1}^T \langle \mathbf{x}^{(t)}, \mathbf{u}^{(t)} \rangle.$$

- Φ is a set of *strategy deviations*
- When it contains only *constant functions*, we have **external regret**
 - This is the usual notion of regret, already covered
- The richer the set of deviations, the stronger the notion of hindsight rationality
- **Swap regret** contains all possible deviations

Swap regret versus external regret

What's the relation between external and swap regret?

1	0	0	1	0	0	1	0	0	1
2	0	1	0	0	1	0	0	1	0
3	1	0	0	1	0	0	1	0	0

- The learner experiences *zero external regret*, but the swap regret grows linearly with the time horizon
- Algorithms such as MWU can have large swap regret
- We need new algorithmic ideas to minimize swap regret

Connecting Phi-regret with correlated equilibria

Definition 1.7 (Φ -equilibrium). A correlated distribution $\mu \in \Delta(\mathcal{X}_1 \times \cdots \times \mathcal{X}_n)$ is an ϵ - Φ -equilibrium if for any player $i \in [n]$ and deviation function $\Phi_i \ni \phi_i : \mathcal{X}_i \rightarrow \mathcal{X}_i$,

$$\mathbb{E}_{(\mathbf{x}_1, \dots, \mathbf{x}_n) \sim \mu} u_i(\mathbf{x}_1, \dots, \mathbf{x}_n) \geq \mathbb{E}_{(\mathbf{x}_1, \dots, \mathbf{x}_n) \sim \mu} u_i(\phi_i(\mathbf{x}_i), \mathbf{x}_{-i}) - \epsilon.$$



This captures both correlated and coarse correlated equilibria

Theorem. If each player is minimizing Phi-regret, the average correlated distribution of play converges to a Phi-equilibrium.

A framework for minimizing Phi-regret

The framework of Gordon et al. shows how to minimize Phi-regret using the following two basic oracles

1. For any deviation function, an oracle that returns a **fixed point** of that function
 - a. A fixed point exists by Brouwer's fixed-point theorem
 - b. It is easy to compute a fixed point of a linear function
2. An *external* regret minimizer for the *set of deviations*
 - a. This increases the complexity since the set of deviations is more complex

$$\max_{\phi \in \Phi} \left\{ \sum_{t=1}^T u_{\Phi}^{(t)}(\phi) \right\} - \sum_{t=1}^T u_{\Phi}^{(t)}(\phi^{(t)})$$

The algorithm of Gordon et al.



Algorithm 1: The template of Gordon et al. [2008] for minimizing Φ -regret.

```
1 Input: An external regret minimizer  $\mathfrak{R}_\Phi$  for the set  $\Phi$ 
2 NEXTSTRATEGY():
3   Set  $\phi^{(t)} := \mathfrak{R}_\Phi.\text{NEXTSTRATEGY}()$ ;
4   return  $\mathcal{X} \ni \mathbf{x}^{(t)} = \phi^{(t)}(\mathbf{x}^{(t)})$ ;

5 OBSERVEUTILITY( $\mathbf{u}^{(t)}$ ):
6   Set  $u_\Phi^{(t)} : \phi \mapsto \langle \phi(\mathbf{x}^{(t)}), \mathbf{u}^{(t)} \rangle$ ;
7    $\mathfrak{R}_\Phi.\text{OBSERVEUTILITY}(u_\Phi^{(t)})$ ;
```

From Phi-regret to external regret

Theorem. The Phi-regret of the algorithm is equal to the external regret with respect to the set of deviations.

$$\begin{aligned}\Phi\text{Reg}^{(T)} &= \max_{\phi \in \Phi} \left\{ \sum_{t=1}^T \langle \phi(\mathbf{x}^{(t)}), \mathbf{u}^{(t)} \rangle \right\} - \sum_{t=1}^T \langle \mathbf{x}^{(t)}, \mathbf{u}^{(t)} \rangle \\ &= \max_{\phi \in \Phi} \left\{ \sum_{t=1}^T \langle \phi(\mathbf{x}^{(t)}), \mathbf{u}^{(t)} \rangle \right\} - \sum_{t=1}^T \langle \phi^{(t)}(\mathbf{x}^{(t)}), \mathbf{u}^{(t)} \rangle\end{aligned}$$

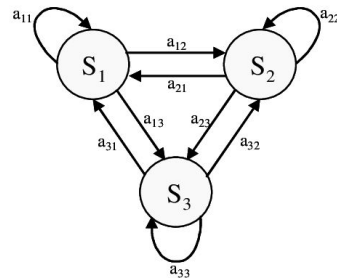
since $\mathbf{x}^{(t)} = \phi^{(t)}(\mathbf{x}^{(t)})$.

$$\Phi\text{Reg}^{(T)} = \max_{\phi \in \Phi} \left\{ \sum_{t=1}^T u_{\Phi}^{(t)}(\phi) \right\} - \sum_{t=1}^T u_{\Phi}^{(t)}(\phi^{(t)}) = \text{Reg}^{(T)}.$$

Swap regret in normal-form games



The set of deviations is the set of **stochastic matrices**



$$\{[(\mathbf{x}_a)_{a \in \mathcal{A}}] : \mathbf{x}_a \in \Delta(\mathcal{A}) \quad \forall a \in \mathcal{A}\}$$

- Any deviation gives rise to a *Markov chain*
- Any stationary distribution is a fixed point—easy to compute
- The set of stochastic matrices is a Cartesian product of probability distributions; we can use regret circuits!

The algorithm of Blum-Mansour



Algorithm 1: Blum-Mansour algorithm for minimizing swap regret

```
1 Input: A regret minimizer  $\mathfrak{R}_a$  for each action  $a \in \mathcal{A}$ 
2 NEXTSTRATEGY():
3   for each action  $a \in \mathcal{A}$  do
4      $\Delta(\mathcal{A}) \ni \mathbf{x}_a^{(t)} := \mathfrak{R}_a.\text{NEXTSTRATEGY}();$ 
5   Set  $\mathbf{M}^{(t)} := [(\mathbf{x}_a^{(t)})_{a \in \mathcal{A}}]$ 
6   return  $\Delta(\mathcal{A}) \ni \mathbf{x}^{(t)} = \mathbf{M}^{(t)} \mathbf{x}^{(t)};$ 

7 OBSERVEUTILITY( $\mathbf{u}^{(t)} \in \mathbb{R}^{\mathcal{A}}$ ):
8   for each action  $a \in \mathcal{A}$  do
9      $\mathbf{u}_a^{(t)} := \mathbf{x}^{(t)}[a] \mathbf{u}^{(t)} ;$ 
10     $\mathfrak{R}_a.\text{OBSERVEUTILITY}(\mathbf{u}_a^{(t)});$ 
```
