

Equilibrium Computation in Normal-form Games: Linear Programming and Online Learning



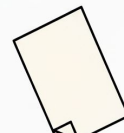
15 888 **Computational Game Solving** (Fall 2026)
Ioannis Anagnostides

Today's lecture

- Solving (two-player) zero-sum games through linear programming
 - Polynomial-time algorithms
 - Proof of the minimax theorem
- Simple iterative algorithms for zero-sum games
 - Best-response dynamics
 - Fictitious play
- Online learning and regret minimization
 - Basic learning algorithms (FTRL, MD, RM)
 - Connections to game-theoretic equilibria

Recap on normal-form games

- We think of them as *simultaneous-move* interactions
- Each player selects an action from a finite set
- **Utility function** maps actions to a real value
- Players can randomize by specifying distributions
- Players maximize expected utility
- Any finite game can be represented in normal form
- The normal-form is often inefficient (more on this in the next lecture)



Zero-sum games

- Two players with exactly **opposing** interests
- The row player wants to solve

$$\min_{\mathbf{x} \in \Delta(\mathcal{A}_1)} \max_{\mathbf{y} \in \Delta(\mathcal{A}_2)} \mathbf{x}^\top \mathbf{A} \mathbf{y} = \min_{\mathbf{x} \in \Delta(\mathcal{A}_1)} \max_{\mathbf{y} \in \Delta(\mathcal{A}_2)} \sum_{a_1 \in \mathcal{A}_1} \sum_{a_2 \in \mathcal{A}_2} x[a_1] A[a_1, a_2] y[a_2].$$

- The row player first specifies a mixed strategy, and then the column player responds optimally to that strategy

Minimax through linear programming

$$\min_{\mathbf{x} \in \Delta(\mathcal{A}_1)} \max_{\mathbf{y} \in \Delta(\mathcal{A}_2)} \mathbf{x}^\top \mathbf{A} \mathbf{y} = \sum_{a_1 \in \mathcal{A}_1} \sum_{a_2 \in \mathcal{A}_2} x[a_1] \mathbf{A}[a_1, a_2] y[a_2].$$

- Equivalent to $\min_{\mathbf{x} \in \Delta(\mathcal{A}_1)} \max_{a_2 \in \mathcal{A}_2} \mathbf{x}^\top \mathbf{A}[:, a_2]$
- The column player can always respond optimally through a *pure strategy*
- What if we introduce an auxiliary variable? $t \geq \mathbf{x}^\top \mathbf{A}[:, a_2]$ for all $a_2 \in \mathcal{A}_2$

Linear programming formulation

LP for row player

$$\begin{array}{ll}\text{minimize} & t \\ \text{subject to} & t \geq \mathbf{x}^\top \mathbf{A}[:, a_2] \text{ for all } a_2 \in \mathcal{A}_2, \\ & \sum_{a_1 \in \mathcal{A}_1} \mathbf{x}[a_1] = 1, \\ & \mathbf{x} \geq 0.\end{array}$$

Linear programming formulation

LP for row player

$$\begin{array}{ll}\text{minimize} & t \\ \text{subject to} & t \geq \mathbf{x}^\top \mathbf{A}[:, a_2] \text{ for all } a_2 \in \mathcal{A}_2, \\ & \sum_{a_1 \in \mathcal{A}_1} \mathbf{x}[a_1] = 1, \\ & \mathbf{x} \geq 0.\end{array}$$

LP for column player

$$\begin{array}{ll}\text{maximize} & t \\ \text{subject to} & t \leq \mathbf{y}^\top \mathbf{A}[a_1, :] \text{ for all } a_1 \in \mathcal{A}_1, \\ & \sum_{a_2 \in \mathcal{A}_2} \mathbf{y}[a_2] = 1, \\ & \mathbf{y} \geq 0.\end{array}$$

Linear programming formulation

LP for row player

$$\begin{array}{ll}\text{minimize} & t \\ \text{subject to} & t \geq \mathbf{x}^\top \mathbf{A}[:, a_2] \text{ for all } a_2 \in \mathcal{A}_2, \\ & \sum_{a_1 \in \mathcal{A}_1} \mathbf{x}[a_1] = 1, \\ & \mathbf{x} \geq 0.\end{array}$$

LP for column player

$$\begin{array}{ll}\text{maximize} & t \\ \text{subject to} & t \leq \mathbf{y}^\top \mathbf{A}[a_1, :] \text{ for all } a_1 \in \mathcal{A}_1, \\ & \sum_{a_2 \in \mathcal{A}_2} \mathbf{y}[a_2] = 1, \\ & \mathbf{y} \geq 0.\end{array}$$



These are duals!

Proof of the minimax theorem

Linear programming duality implies

$$\min_{\mathbf{x} \in \Delta(\mathcal{A}_1)} \max_{\mathbf{y} \in \Delta(\mathcal{A}_2)} \mathbf{x}^\top \mathbf{A} \mathbf{y} = \max_{\mathbf{y} \in \Delta(\mathcal{A}_2)} \min_{\mathbf{x} \in \Delta(\mathcal{A}_1)} \mathbf{x}^\top \mathbf{A} \mathbf{y} = v.$$

Theorem. There is a polynomial-time algorithm for finding minimax (or Nash) equilibria in zero-sum games.

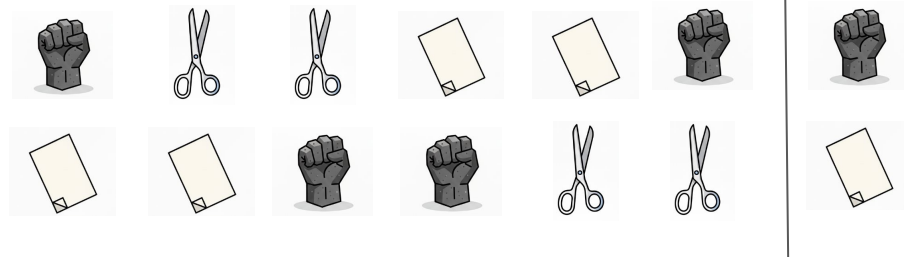
Are we done? Unfortunately, LP solvers scale poorly in large games

Iterative algorithms: best-response dynamics

- Simple idea: best-respond to each other's strategy
- Does it converge?

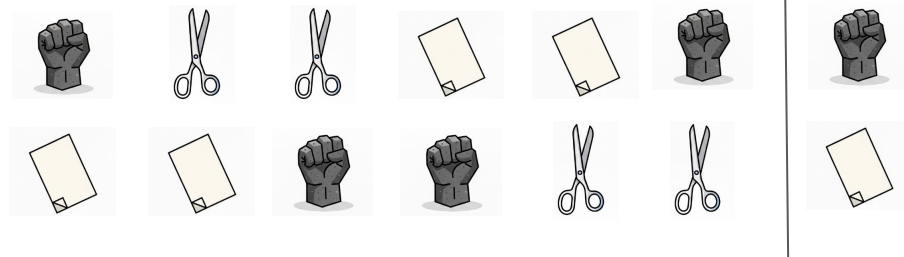
Iterative algorithms: best-response dynamics

- Simple idea: best-respond to each other's strategy
- Does it converge?
- Some games don't even have pure-strategy equilibria



Iterative algorithms: best-response dynamics

- Simple idea: best-respond to each other's strategy
- Does it converge?
- Some games don't even have pure-strategy equilibria
- What about the average?
- What if winning by paper is twice more valuable?



Iterative algorithms: fictitious play

- A more sophisticated algorithm is **fictitious play**
- It best-responds not to the previous strategy, but to the **average strategy** of the opponent so far
- Basic opponent modeling
- Julia Robinson showed it always converges
- But the rate of convergence can be exponentially slow (Daskalakis and Pan, 2014)
- Open question beyond adversarial tiebreaks



Iterative algorithms: fictitious play

- A more sophisticated algorithm is **fictitious play**
- It best-responds not to the previous strategy, but to the **average strategy** of the opponent so far
- Basic opponent modeling
- Julia Robinson showed it always converges
- But the rate of convergence can be exponentially slow (Daskalakis and Pan, 2014)
- Open question beyond adversarial tiebreaks
- If our opponent knows we are using fictitious play, are we **exploitable**?



Online learning

- A learner interacts with an environment over a sequence of rounds
- The learner first selects a mixed strategy
- The environment provides as feedback some utility vector $\langle \mathbf{x}^{(t)}, \mathbf{u}^{(t)} \rangle$

Online learning

- A learner interacts with an environment over a sequence of rounds
- The learner first selects a mixed strategy
- The environment provides as feedback some utility vector $\langle \mathbf{x}^{(t)}, \mathbf{u}^{(t)} \rangle$

Regret is the most common measure of performance

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



Regret can be negative!



0	1	0	1	0	1	0	1	0	1	0
1	0	1	0	1	0	1	0	1	0	1

Is fictitious play no-regret?

$$\mathbf{x}^{(t)} \in \operatorname{argmax}_{\mathbf{x} \in \Delta^m} \sum_{\tau=1}^{t-1} \langle \mathbf{x}, \mathbf{u}^{(\tau)} \rangle.$$

Aka. “follow the leader”

What happens if fictitious play encounters the following sequence of utilities?

	ε	0	1	0	1	0	1	0	1
		0	1	0	1	0	1	0	1

Is fictitious play no-regret?

$$\mathbf{x}^{(t)} \in \operatorname{argmax}_{\mathbf{x} \in \Delta^m} \sum_{\tau=1}^{t-1} \langle \mathbf{x}, \mathbf{u}^{(\tau)} \rangle.$$

Aka. “follow the leader”

What happens if fictitious play encounters the following sequence of utilities?



ε	0	1	0	1	0	1	0	1
0	1	0	1	0	1	0	1	0

Theorem. There is a sequence of utilities such that fictitious play incurs linear regret.

Follow the regularized leader (FTRL)

A small tweak to fictitious play:

$$\mathbf{x}^{(t)} = \operatorname{argmax}_{\mathbf{x} \in \Delta^m} \left\{ \sum_{\tau=1}^{t-1} \langle \mathbf{x}, \mathbf{u}^{(\tau)} \rangle - \frac{1}{\eta} \mathcal{R}(\mathbf{x}) \right\}$$

- $\mathcal{R}$ is a regularizer *strongly convex* with respect to a norm $\|\cdot\|$
- η is the learning rate

Multiplicative weights update:

Let $\mathcal{R} : \mathbf{x} \mapsto \sum_{a \in \mathcal{A}} \mathbf{x}[a] \ln \mathbf{x}[a]$

$$\mathbf{x}^{(t)}[a] \propto \exp \left(\eta \sum_{\tau=1}^{t-1} \mathbf{u}^{(\tau)}[a] \right)$$

Euclidean regularization:

Let $\mathcal{R} : \mathbf{x} \mapsto \frac{1}{2} \sum_{a \in \mathcal{A}} (\mathbf{x}[a])^2 = \frac{1}{2} \|\mathbf{x}\|_2^2$

$$\mathbf{x}^{(t)} = \Pi_{\Delta(\mathcal{A})} \left(\eta \sum_{\tau=1}^{t-1} \mathbf{u}^{(\tau)} \right)$$

FTRL has no-regret

Theorem. Under any sequence of utilities, the regret of FTRL is bounded as

$$\text{Reg}^{(T)} \leq \frac{R}{\eta} + \eta \sum_{t=1}^T \|\mathbf{u}^{(t)}\|_*^2.$$

By selecting the learning rate optimally,

$$\text{Reg}^{(T)} \leq 2B\sqrt{RT}.$$

- R is the range of the regularizer
- $\|\cdot\|_*$ is the *dual* norm

FTRL has no-regret

Theorem. Under any sequence of utilities, the regret of FTRL is bounded as

$$\text{Reg}^{(T)} \leq \frac{R}{\eta} + \eta \sum_{t=1}^T \|\mathbf{u}^{(t)}\|_*^2.$$

- R is the range of the regularizer
- $\|\cdot\|_*$ is the *dual* norm

By selecting the learning rate optimally,

$$\text{Reg}^{(T)} \leq 2B\sqrt{RT}.$$

For MWU,

$$\text{Reg}^{(T)} \leq 2\sqrt{T \ln m}.$$

Online mirror descent

Another class of online algorithms:

$$\mathbf{x}^{(t)} := \operatorname{argmax}_{\mathbf{x} \in \Delta(\mathcal{A})} \left\{ \langle \mathbf{x}, \mathbf{u}^{(t-1)} \rangle - \frac{1}{\eta} \mathcal{B}_{\mathcal{R}}(\mathbf{x}, \mathbf{x}^{(t-1)}) \right\}.$$

We measure distance through the Bregman divergence:

$$\mathcal{B}_{\mathcal{R}}(\mathbf{x}, \mathbf{x}') := \mathcal{R}(\mathbf{x}) - \mathcal{R}(\mathbf{x}') - \langle \nabla \mathcal{R}(\mathbf{x}'), \mathbf{x} - \mathbf{x}' \rangle$$

Online gradient descent:

$$\mathbf{x}^{(t)} = \Pi_{\Delta(\mathcal{A})}(\mathbf{x}^{(t-1)} + \eta \mathbf{u}^{(t-1)}).$$



For some regularizers, FTRL is equivalent to MD



The regret bound of MD is similar to that of FTRL

Regret matching

- MWU can be expressed as

$$\mathbf{x}^{(t)}[a] \propto \exp\left(\eta \sum_{\tau=1}^{t-1} (\mathbf{u}^{(\tau)}[a] - \langle \mathbf{x}^{(\tau)}, \mathbf{u}^{(\tau)} \rangle)\right) = \exp(\eta \mathbf{r}^{(t-1)}[a])$$

- FTRL with Euclidean regularization can be expressed as

$$\mathbf{x}^{(t)} := \Pi_{\Delta(\mathcal{A})}(\eta \mathbf{r}^{(t-1)})$$

Regret matching

- MWU can be expressed as

$$\mathbf{x}^{(t)}[a] \propto \exp\left(\eta \sum_{\tau=1}^{t-1} (\mathbf{u}^{(\tau)}[a] - \langle \mathbf{x}^{(\tau)}, \mathbf{u}^{(\tau)} \rangle)\right) = \exp(\eta \mathbf{r}^{(t-1)}[a])$$

- FTRL with Euclidean regularization can be expressed as

$$\mathbf{x}^{(t)} := \Pi_{\Delta(\mathcal{A})}(\eta \mathbf{r}^{(t-1)})$$

- What about $\mathbf{x}^{(t)} \propto \max(\mathbf{r}^{(t-1)}, \mathbf{0}) := [\mathbf{r}^{(t-1)}]^+$

Regret matching

Algorithm 1: Regret matching (RM)

```
1 Initialize cumulative regrets  $\mathbf{r}^{(0)} := \mathbf{0}$ ;  
2 for  $t = 1, \dots, T$  do  
3   Define  $\boldsymbol{\theta}^{(t)} := \max(\mathbf{r}^{(t-1)}, \mathbf{0})$ ;  
4   if  $\boldsymbol{\theta}^{(t)} = \mathbf{0}$  then  
5     | Let  $\mathbf{x}^{(t)} \in \Delta(\mathcal{A})$  be arbitrary  
6   else  
7     | Compute  $\mathbf{x}^{(t)} := \boldsymbol{\theta}^{(t)} / \|\boldsymbol{\theta}^{(t)}\|_1$ ;  
8   Output strategy  $\mathbf{x}^{(t)} \in \Delta(\mathcal{A})$  ;  
9   Observe utility  $\mathbf{u}^{(t)} \in [0, 1]^{\mathcal{A}}$ ;  
10   $\mathbf{r}^{(t)} := \mathbf{r}^{(t-1)} + \mathbf{u}^{(t)} - \langle \mathbf{x}^{(t)}, \mathbf{u}^{(t)} \rangle \mathbf{1}$ ;
```

Algorithm 2: Regret matching⁺ (RM⁺)

```
1 Initialize cumulative regrets  $\mathbf{r}^{(0)} := \mathbf{0}$ ;  
2 for  $t = 1, \dots, T$  do  
3   Define  $\boldsymbol{\theta}^{(t)} := \mathbf{r}^{(t-1)}$ ;  
4   if  $\boldsymbol{\theta}^{(t)} = \mathbf{0}$  then  
5     | Let  $\mathbf{x}^{(t)} \in \Delta(\mathcal{A})$  be arbitrary  
6   else  
7     | Compute  $\mathbf{x}^{(t)} := \boldsymbol{\theta}^{(t)} / \|\boldsymbol{\theta}^{(t)}\|_1$ ;  
8   Output strategy  $\mathbf{x}^{(t)} \in \Delta(\mathcal{A})$  ;  
9   Observe utility  $\mathbf{u}^{(t)} \in [0, 1]^{\mathcal{A}}$ ;  
10   $\mathbf{r}^{(t)} := [\mathbf{r}^{(t-1)} + \mathbf{u}^{(t)} - \langle \mathbf{x}^{(t)}, \mathbf{u}^{(t)} \rangle \mathbf{1}]^+$ ;
```

Analysis of RM

Theorem. For any sequence of utilities, the regret of RM is at most $\sqrt{mT}$.

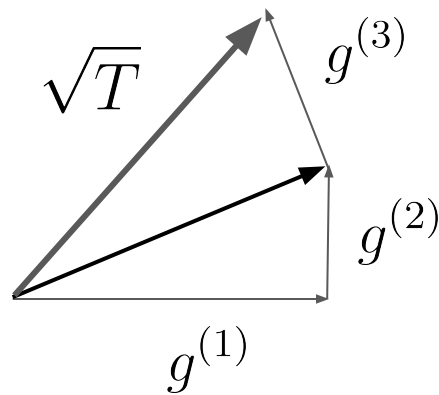
Instantaneous regret: $g^{(t)} = u^{(t)} - \langle x^{(t)}, u^{(t)} \rangle \mathbf{1}$

Pythagorean lemma:



Instantaneous regret is always
orthogonal to the regret
accumulated thus far

$$\langle x^{(t)}, g^{(t)} \rangle = 0 = \langle [r^{(t-1)}]^+, g^{(t)} \rangle$$



Self-play in zero-sum games

- Both players follow a no-regret algorithm
- This is called “self-play”

Theorem. If both players have no-regret, the **average strategies** converge to a minimax equilibrium.

$$\text{Reg}_1^{(T)} = \max_{\mathbf{x}' \in \Delta(\mathcal{A}_1)} \sum_{t=1}^T \langle \mathbf{x}' - \mathbf{x}^{(t)}, -\mathbf{A}\mathbf{y}^{(t)} \rangle = \sum_{t=1}^T \langle \mathbf{x}^{(t)}, \mathbf{A}\mathbf{y}^{(t)} \rangle - T \min_{\mathbf{x}' \in \Delta(\mathcal{A}_1)} \langle \mathbf{x}', \mathbf{A}\bar{\mathbf{y}}^{(T)} \rangle.$$

Similarly,

$$\text{Reg}_2^{(T)} = \max_{\mathbf{y}' \in \Delta(\mathcal{A}_2)} \sum_{t=1}^T \langle \mathbf{y}' - \mathbf{y}^{(t)}, \mathbf{A}^\top \mathbf{x}^{(t)} \rangle = T \max_{\mathbf{y}' \in \Delta(\mathcal{A}_2)} \langle \mathbf{y}', \mathbf{A}^\top \bar{\mathbf{x}}^{(T)} \rangle - \sum_{t=1}^T \langle \mathbf{x}^{(t)}, \mathbf{A}\mathbf{y}^{(t)} \rangle.$$

Self-play in general-sum games

Definition 4.10 (Coarse correlated equilibrium). A correlated distribution $\mu \in \Delta(\mathcal{A}_1 \times \cdots \times \mathcal{A}_n)$ is an ϵ -coarse correlated equilibrium if for any player $i \in [n]$ and deviation $a'_i \in \mathcal{A}_i$,

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

Theorem. If all players follow a no-regret algorithm, the average correlated distribution of play converges to a coarse correlated equilibrium.