Topics in Algorithmic Game Theory and Economics

Pieter Kleer

Max Planck Institute for Informatics
Saarland Informatics Campus

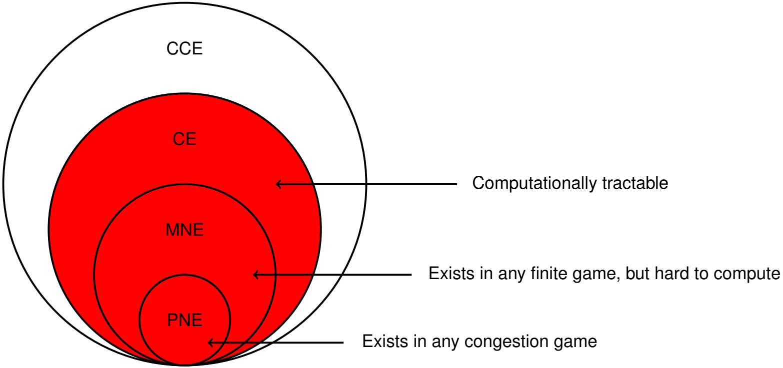
December 16, 2020

Lecture 6
Finite games III - Computation of CE and CCE

Hierarchy of equilibrium concepts

Finite (cost minimization) game $\Gamma = (N, (\mathcal{S}_i)_{i \in N}, (C_i)_{i \in N})$ consists of:

- Finite set N of players.
- Finite strategy set $\mathcal{S}_i$ for every player $i \in N$.
- Cost function $C_i : \times_j \mathcal{S}_j \to \mathbb{R}$ for every $i \in N$.



Two-player games with mixed strategies (recap)

Two-player game (A, B) given by matrices $A, B \in \mathbb{R}^{m \times n}$.

- Row player Alice and column player Bob independently choose strategy $x \in \Delta_A$ and $y \in \Delta_B$.

Gives product distribution $\sigma_{x,y} : \mathcal{S}_A \times \mathcal{S}_B \to [0, 1]$ over strategy profiles:

- $\sigma_{x,y}(a_k, b_\ell) = \sigma_{k\ell} = x_k y_\ell$ for $k = 1, \dots, m$ and $\ell = 1, \dots, n$.

Example

Distribution over strategy profiles is given by

$$\begin{pmatrix} x_1 y_1 & x_1 y_2 & x_1 y_3 \\ x_2 y_1 & x_2 y_2 & x_2 y_3 \end{pmatrix}$$

	b_1	b_2	b_3
a_1	$(0, 2)$	$(1, 0)$	$(2, 1)$
a_2	$(3, 0)$	$(0, 1)$	$(1, 4)$

Then expected cost (for Alice) $C_A(\sigma_{x,y})$ is

$$x^T A y = \mathbb{E}_{(a_k, b_\ell) \sim \sigma_{x,y}}[C_A(a_k, b_\ell)] = \sum_{(a_k, b_\ell) \in \mathcal{S}_A \times \mathcal{S}_B} \sigma_{k\ell} C_A(a_k, b_\ell)$$

- Remember that $A_{k\ell} = C_A(a_k, b_\ell)$.

Beyond mixed strategies

Equilibrium concepts as distributions over $\mathcal{S}_A \times \mathcal{S}_B$

We have seen the following equilibrium concepts:

- PNE: Strategy profile $s = (s_A, s_B) \in \mathcal{S}_A \times \mathcal{S}_B$.
  - Gives indicator distribution σ over $\mathcal{S}_A \times \mathcal{S}_B$ with

$$\sigma(t) = \begin{cases} 1 & t = s \\ 0 & t \neq s \end{cases}.$$

- MNE: Mixed strategies (x, y) with $x \in \Delta_A$, $y \in \Delta_B$.
  - Gives product distribution σ over $\mathcal{S}_A \times \mathcal{S}_B$, where

$$\sigma(a_k, b_\ell) = \sigma_{k\ell} = x_k y_\ell.$$

- **(C)CE: (Coarse) correlated equilibrium will be defined as general distribution σ over $\mathcal{S}_A \times \mathcal{S}_\mathcal{B}$.**
  - I.e., not induced by specific player actions.

Game of Chicken

Game of Chicken

Alice and Bob both approach an intersection.

		Bob	
		Stop	*Go*
Alice	*Stop*	$(0, 0)$	$(3, -1)$
	Go	$(-1, 3)$	$(4, 4)$

- Two PNEs: (Stop, Go), (Go, Stop).
- One MNE: Both players randomize over Stop and Go.

Distributions over strategy profiles (a, b) for these equilibria are

$$\begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \text{ and } \begin{pmatrix} \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} \end{pmatrix}.$$

- Sensible ‘equilibrium’ would be the strategy profile distribution

$$\sigma = \begin{pmatrix} 0 & \frac{1}{2} \\ \frac{1}{2} & 0 \end{pmatrix}.$$

- Cannot be achieved as mixed equilibrium.
  - There are **no** $x \in \Delta_A, y \in \Delta_B$ such that $\sigma_{k\ell} = x_k y_\ell$ for all $k, \ell \in \{1, 2\}$.

Idea is to introduce traffic light (mediator or trusted third party).

- Traffic light samples/draws one of the two strategy profiles from distribution.
- Gives realization as recommendation to the players.
  - Tells Alice to go and Bob to stop (or vice versa)

Conditioned on this recommendation, the best thing for a player to do is to follow it.

Correlated equilibrium (CE), informal

Correlated equilibrium $\sigma : \mathcal{S}_A \times \mathcal{S}_B \to [0, 1]$ can be seen as follows.

- Mediator (third party) draws sample $x = (x_A, x_B) \sim \sigma$.
  - σ is known to Alice and Bob, but not x.
- Gives private recommendation x_A to Alice, and x_B to Bob.
  - Alice and Bob do not know each other's recommendation!
    - Game of Chicken is the exception to the rule.
- Recommendations give players some info on which x was drawn.

Player assumes all other players play private recommendation, i.e., Alice assumes Bob follows his recommendation (and vice versa).

In CE, no player has incentive to deviate given its recommendation.

Remark

We will later see no-regret algorithms whose output is a coarse correlated equilibrium (similar algorithms exist converging to CE). Therefore, for (C)CE, it's not always necessary that all players know the distribution σ up front, nor that there is an actual third party that samples from it.

Example

Distribution over strategy profiles is given by

$$\sigma = \begin{pmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{21} & \sigma_{22} & \sigma_{23} \end{pmatrix} = \begin{pmatrix} 0 & 1/8 & 1/8 \\ 2/8 & 1/8 & 3/8 \end{pmatrix}$$

	b_1	b_2	b_3
a_1	$(0,2)$	$(1,0)$	$(2,1)$
a_2	$(3,0)$	$(0,1)$	$(1,4)$

- Suppose Alice gets second row a_2 as recommendation.
- This gives Alice a (conditional) probability distribution ρ for column privately recommended to Bob:
  - Column b_1 with probability $\frac{\frac{2}{8}}{\frac{2}{8}+\frac{1}{8}+\frac{3}{8}} = \frac{2}{6}$.
  - Column b_2 with probability $\frac{\frac{1}{8}}{\frac{2}{8}+\frac{1}{8}+\frac{3}{8}} = \frac{1}{6}$.
  - Column b_3 with probability $\frac{\frac{3}{8}}{\frac{2}{8}+\frac{1}{8}+\frac{3}{8}} = \frac{3}{6}$.
- Assuming distribution ρ over Bob's recommendation, notion of CE says Alice should have no incentive to deviate to first row a_1 (in expectation).
  - $\mathbb{E}_\rho[\text{Row } a_2] = 3 \times 2/6 + 0 \times 1/6 + 1 \times 3/6 = 9/6$.
  - $\mathbb{E}_\rho[\text{Row } a_1] = 0 \times 2/6 + 1 \times 1/6 + 2 \times 3/6 = 7/6$.

σ as above is not a CE!

(Coarse) correlated equilibrium

Definition (Correlated equilibrium (CE))

A distribution σ on $\times_i \mathcal{S}_i$ is a correlated equilibrium if for every $i \in N$ and $t_i \in \mathcal{S}_i$, and every unilateral deviation $t_i' \in \mathcal{S}_i$, it holds that

$$\mathbb{E}_{x\sim\sigma}\left[C_i(x) \mid x_i = t_i\right] \leq \mathbb{E}_{x\sim\sigma}\left[C_i(t_i', x_{-i}) \mid x_i = t_i\right].$$

Set-up for coarse correlated equilibrium is similar, but you do not get private recommendation from mediator.

Definition (Coarse correlated equilibrium (CCE))

A distribution σ on $\times_i \mathcal{S}_i$ is a coarse correlated equilibrium if for every $i \in N$, and every unilateral deviation $t_i' \in \mathcal{S}_i$, it holds that

$$\mathbb{E}_{x\sim\sigma}\left[C_i(x)\right] \leq \mathbb{E}_{x\sim\sigma}\left[C_i(t_i', x_{-i})\right].$$

Exercise: Prove that every CE is also CCE.
(Hint: Use "Law of total expectation", i.e., $\mathbb{E}[X] = \mathbb{E}\left[\mathbb{E}[X|Y]\right]$.)

Final remark

Remember MNE is pair of mixed strategies (x, y) that yields product distribution σ over strategy profiles.

MNE through the lens of CE

MNE is special case of CE, where recommendation of mediator gives no extra information.

- Conditional distribution ρ that Alice constructs for Bob's private recommendation is the same for every row recommended to her.
  - It is just the mixed strategy y of Bob!
  - That is, the recommendation is not relevant for Alice.

Exercise: Check this yourself!

Computation of correlated equilibrium

Linear program for computing CE

Once again, linear programming comes to the rescue...

Theorem

For a given finite game Γ*, there is a linear program that computes a correlated equilibrium* $\sigma : \times_i \mathcal{S}_i \to [0, 1]$ *in time polynomial in* $|\times_i \mathcal{S}_i|$ *and the input size of the cost functions.*

- LP has one variable σ_s for every strategy profile $s \in \times_i \mathcal{S}_i$.
  - Polynomial number of variables if number of players $|N|$ is assumed to be constant.
- For two-player games, note that $|\mathcal{S}_A \times \mathcal{S}_B| = mn$.

Conditions in definition CE can be modeled as linear program.

- We will do the 2-player case, and focus on Alice.

Definition (Correlated equilibrium (for Alice))

Distribution σ on $\mathcal{S}_A \times \mathcal{S}_B$ is correlated equilibrium if for every "recommendation" $a_k \in \mathcal{S}_A$ and deviation $a_{k'}$ it holds, with $x = (x_A, x_B)$, that

$$\mathbb{E}_{x\sim\sigma}\left[C_A(x_A, x_B) \mid x_A = a_k\right] \le \mathbb{E}_{x\sim\sigma}\left[C_A(a_{k'}, x_B) \mid x_A = a_k\right].$$

- LP will have variables $\sigma_{k\ell}$ for $k = 1, \ldots, m$, $\ell = 1, \ldots, n$.

Linear constraints for Alice

Fix "recommended row" $a_k \in \mathcal{S}_A$ and "deviation" $a_{k'} \in \mathcal{S}_A$. Now,

$$\begin{aligned}
\mathbb{E}_{x\sim\sigma}\left[C_A(x_A, x_B) \mid x_A = a_k\right] &= \sum_{\ell=1,\ldots,n} C_A(a_k, b_\ell)\mathbb{P}[x = (x_A, x_B) | x_A = a_k] \\
&= \sum_{\ell=1,\ldots,n} C_A(a_k, b_\ell)\frac{\sigma_{k\ell}}{\sum_r \sigma_{kr}} \\
&= \frac{1}{\sum_r \sigma_{kr}} \sum_{\ell=1,\ldots,n} C_A(a_k, b_\ell)\sigma_{k\ell} \\
\mathbb{E}_{x\sim\sigma}\left[C_A(a_{k'}, x_B) \mid x_A = a_k\right] &= \sum_{\ell=1,\ldots,n} C_A(a_{k'}, b_\ell)\mathbb{P}[x = (x_A, x_B) | x_A = a_k] \\
&= \frac{1}{\sum_r \sigma_{kr}} \sum_{\ell=1,\ldots,n} C_A(a_{k'}, b_\ell)\sigma_{k\ell}
\end{aligned}$$

Conditions in (1) for Alice and Bob are equivalent to

$$\sum_{\ell=1,\ldots,n} C_A(a_k, b_\ell)\sigma_{k\ell} \le \sum_{\ell=1,\ldots,n} C_A(a_{k'}, b_\ell)\sigma_{k\ell} \qquad \forall a_k, a_{k'} \in \mathcal{S}_A$$
$$\sum_{k=1,\ldots,m} C_B(a_k, b_\ell)\sigma_{k\ell} \le \sum_{k=1,\ldots,m} C_B(a_k, b_{\ell'})\sigma_{k\ell} \quad \forall b_\ell, b_{\ell'} \in \mathcal{S}_B$$

- Note that these are linear constraints in variables $\sigma_{k\ell}$.

Linear program is now as follows

$$\begin{array}{lll}
\max & 0 & \\
\text{s.t.} & \sum_{\ell=1,\ldots,n} C_A(a_k, b_\ell)\sigma_{k\ell} \le \sum_{\ell=1,\ldots,n} C_A(a_{k'}, b_\ell)\sigma_{k\ell} & \forall a_k, a_{k'} \in \mathcal{S}_A \\
& \sum_{k=1,\ldots,m} C_B(a_k, b_\ell)\sigma_{k\ell} \le \sum_{k=1,\ldots,m} C_B(a_k, b_{\ell'})\sigma_{k\ell} & \forall b_\ell, b_{\ell'} \in \mathcal{S}_B \\
& \sum_{k,\ell} \sigma_{k\ell} = 1 & \\
& \sigma_{k\ell} \ge 0 & \forall a_k \in \mathcal{S}_A, b_\ell \in \mathcal{S}_B
\end{array}$$

$$
\begin{array}{lll}
\max & 0 & \\
\text{s.t.} & \sum_{\ell=1,\ldots,n} C_A(a_k, b_\ell)\sigma_{k\ell} \leq \sum_{\ell=1,\ldots,n} C_A(a_{k'}, b_\ell)\sigma_{k\ell} & \forall a_k, a_{k'} \in \mathcal{S}_A \\
& \sum_{k=1,\ldots,m} C_B(a_k, b_\ell)\sigma_{k\ell} \leq \sum_{k=1,\ldots,m} C_B(a_k, b_{\ell'})\sigma_{k\ell} & \forall b_\ell, b_{\ell'} \in \mathcal{S}_B \\
& \sum_{k,\ell} \sigma_{k\ell} = 1 & \\
& \sigma_{k\ell} \geq 0 & \forall a_k \in \mathcal{S}_A, b_\ell \in \mathcal{S}_B
\end{array}
$$

- This is a feasiblity LP, i.e., the goal is to find a feasible solution of the linear system above.
- We know at least one solution exists by Nash's theorem
  - Remember that every MNE is also CE.

Why not use the LP for computing MNE? We would need additional ***non-linear*** *constraint enforcing that σ is product distribution.*

For general finite game $\Gamma = (N, (\mathcal{S}_i), (C_i))$, linear program is as follows.

$$
\begin{aligned}
\max \quad & 0 \\
\text{s.t.} \quad & \sum_{s_{-i} \in \mathcal{S}_{-i}} C_i(s_i, s_{-i}) \sigma(s_i, s_{-i}) \\
& \qquad \leq \sum_{s_{-i} \in \mathcal{S}_{-i}} C_i(s'_i, s_{-i}) \sigma(s_i, s_{-i}) \quad \forall i \in N \text{ and } s_i, s'_i \in \mathcal{S}_i \\
& \sum_{s \in \times_i \mathcal{S}_i} \sigma(s) = 1 \\
& \sigma(s) \geq 0 \qquad \forall s \in \times_i \mathcal{S}_i.
\end{aligned}
$$

- We use notation $\mathcal{S}_{-i} = \mathcal{S}_1 \times \cdots \times \mathcal{S}_{i-1} \times \mathcal{S}_{i+1} \times \cdots \times \mathcal{S}_{|N|}$.
  - And $s_{-i} = (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_{|N|}) \in \mathcal{S}_{-i}$.

No-regret dynamics

The model

Alice, with strategy set $\mathcal{S}_A = \{a_1, \ldots, a_m\}$, plays "game" against adversary.

- Adversary will be used to represent other players later on.
- (Looking ahead: Players will converge to CCE.)

The game dynamics

Game is repeated for T rounds. In every round $t = 1, \ldots, T$:

- Alice picks prob. distr. $p^{(t)} = (p_1^{(t)}, \ldots, p_m^{(t)})$ over $\{a_1, \ldots, a_m\}$.
- Adversary picks cost vector $c^{(t)} : \{a_1, \ldots, a_m\} \to [0, 1]$.
- Strategy $a^{(t)}$ is drawn according to $p^{(t)}$.
  - Alice incurs cost $c^{(t)}(a^{(t)})$ and gets to know cost vector $c^{(t)}$.

Goal of Alice is to minimize average cost $\frac{1}{T}\sum_{t=1}^{T} c^{(t)}(a^{(t)})$

against a benchmark. What should the benchmark be?

Best choices in hindsight

Would be natural to compare against best choices in hindsight, i.e.,

$$\frac{1}{T}\sum_{t=1}^{T}\min_{a\in\mathcal{S}_A} c^{(t)}(a^{(t)}).$$

- Alice's cost if she would have put all prob. mass on strategy minimizing cost vector $c^{(t)}$, in every step t.
  - Said differently, Alice's best choice if adversary would have to announce cost vector first.

"Regret" of Alice, for given realization $a^{(1)},\ldots,a^{(T)}$, would then be defined as

$$\alpha(T) = \frac{1}{T}\left(\sum_{t=1}^{T} c^{(t)}(a^{(t)}) - \sum_{t=1}^{T}\min_{a\in\mathcal{S}_A} c^{(t)}(a)\right)$$

- Alice has no (or vanishing) regret if, in expectation, $\alpha(T)\to 0$ when $T\to\infty$.

We next illustrate that, under the definition $\alpha(T)$, vanishing regret cannot be achieved. (We will give an alternative definition afterwards.)

$$\alpha(T) = \frac{1}{T}\left(\sum_{t=1}^{T} c^{(t)}(a^{(t)}) - \sum_{t=1}^{T} \min_{a \in \mathcal{S}_A} c^{(t)}(a)\right)$$

Example

Suppose Alice has two actions a and b. In every round, when Alice chooses $p^{(t)} = (p_a^{(t)}, p_b^{(t)})$, adversary sets

$$c^{(t)} = (c^{(t)}(a), c^{(t)}(b)) = \begin{cases} (1,0) & p_a^{(t)} \geq 1/2 \\ (0,1) & p_b^{(t)} > 1/2 \end{cases} .$$

- Expected cost in round t is at least $1/2$.
- Best choice in hindsight gives cost of zero in round t.

Expected regret $\alpha(T)$ is at least $1/2$ for every T.

Is there another "sensible" regret definition yielding non-trivial results?

Best fixed strategy in hindsight

Another possibility is to compare with best fixed strategy in hindsight, i.e.,

$$\min_{a \in \mathcal{S}_A} \frac{1}{T} \sum_{t=1}^{T} c^{(t)}(a).$$

- We interchange "minimum" and "summation".
- Cost if Alice would have been allowed to switch to (fixed) strategy a in every step.
  - This is still w.r.t. to adversarial cost vectors chosen by adversary based on prob. distributions $p^{(t)}$.

Definition (Regret)

For given prob. distr. $p^{(1)}, \ldots, p^{(T)}$ and adversarial cost vectors $c^{(1)}, \ldots, c^{(T)}$, the (time-averaged) regret of Alice is defined as

$$\rho(T) = \frac{1}{T}\left(\sum_{t=1}^{T} c^{(t)}(a^{(t)}) - \min_{a \in \mathcal{S}_A} \sum_{t=1}^{T} c^{(t)}(a)\right),$$

where $a^{(t)}$ is sample according to distribution $p^{(t)}$. Alice has no regret (w.r.t. chosen distributions) if $\rho(T) \to 0$ when $T \to \infty$, in expectation.

$$\rho(T) = \frac{1}{T}\left(\sum_{t=1}^{T} c^{(t)}(a^{(t)}) - \min_{a \in \mathcal{S}_A} \sum_{t=1}^{T} c^{(t)}(a)\right)$$

More general, "Alice" is called an online decision making algorithm.

- Such an algorithm can use cost vectors $c^{(1)}, \ldots, c^{(t)}$, distributions $p^{(1)}, \ldots, p^{(t)}$, and realizations $a^{(t)}$, to define distribution $p^{(t+1)}$.
- Adversary can use the same information, including the chosen $p^{(t+1)}$, to define adversarial cost vector $c^{(t+1)}$.

Theorem

There exist no-regret (online decision making) algorithms with

$$\rho(T) = O\left(\sqrt{\log(m)/T}\right)$$

where m is the number of strategies.

- $T = O(\log(m)/\epsilon^2)$ steps enough to get regret below ϵ.
- Will later see Multiplicative Weights (MW) algorithm achieving this.

No-regret dynamics

Convergence to (approximate) CCE

No-regret player dynamics

Let $\Gamma = (N, (\mathcal{S}_i), (C_i))$, with $C_i : \times_j \mathcal{S}_j \to [0, 1]$, and assume every $i \in N$ is equipped with no-regret algorithm $\mathcal{A}_i$.

- At this point, consider the $\mathcal{A}_i$ as "blackbox" algorithms.
- We write $m_i = |\mathcal{S}_i|$ for number of strategies of player $i \in N$.

No-regret (player) dynamics

In every round $t = 1, \dots, T$, every player $i \in N$ does the following:

- Use $\mathcal{A}_i$ to compute prob. distr. $p_i^{(t)} = (p_{i,1}^{(t)}, \dots, p_{i,m_i}^{(t)})$ over $\mathcal{S}_i$.
- The adversarial cost vector $c_i^{(t)} : \mathcal{S}_i \to [0, 1]$ is defined as

$$c_i^{(t)}(a) = \mathbb{E}_{s_{-i}^{(t)} \sim \sigma_{-i}^{(t)}} C_i\left(a, s_{-i}^{(t)}\right) \quad \forall a \in \mathcal{S}_i$$

  where $\sigma_{-i}^{(t)}$ is product distribution formed by the $p_j^{(t)}$ with $j \in N \setminus \{i\}$.
- Strategy $a^{(t)} \sim p_i^{(t)}$ is drawn, and player i incurs corresponding cost.

That is, $\sigma_{-i}^{(t)} : \mathcal{S}_{-i} \to [0, 1]$ is prob. distribution given by $\sigma_{-i}^{(t)}(s_{-i}) = \prod_{j \neq i} p_{j,s_j}^{(t)}$, where $s_{-i} = (s_1, \dots, s_{i-1}, s_{i+1}, \dots, s_{|N|}) \in \mathcal{S}_{-i}$.

- Remember $\mathcal{S}_{-i} = \mathcal{S}_1 \times \cdots \times \mathcal{S}_{i-1} \times \mathcal{S}_{i+1} \times \cdots \times \mathcal{S}_{|N|}$.

No-regret (player) dynamics

In every round $t = 1, \ldots, T$, every player $i \in N$ does the following:

- Use $\mathcal{A}_i$ to compute prob. distr. $p_i^{(t)} = (p_{i,1}^{(t)}, \ldots, p_{i,m_i}^{(t)})$ over $\mathcal{S}_i$.
- The adversarial cost vector $c_i^{(t)} : \mathcal{S}_i \to [0,1]$ is defined as

$$c_i^{(t)}(a) = \mathbb{E}_{s_{-i}^{(t)} \sim \sigma_{-i}^{(t)}} C_i\left(a, s_{-i}^{(t)}\right) \quad \forall a \in \mathcal{S}_i$$

  where $\sigma_{-i}^{(t)}$ is product distribution formed by the $p_j^{(t)}$ with $j \in N \setminus \{i\}$.
- Strategy $a^{(t)} \sim p_i^{(t)}$ is drawn, and player i incurs corresponding cost.

Set $\sigma^{(t)} = \prod_j p_j^{(t)}$ and let $\sigma_T = \frac{1}{T} \sum_{t=1}^{T} \sigma^{(t)}$ be the time average of all production distributions obtained in steps $t = 1, \ldots, T$.

Theorem

The time average σ_T is a $\rho_i(T)$-approximate CCE, i.e., it satisfies

$$\mathbb{E}_{s \sim \sigma_T}[C_i(s)] \leq \mathbb{E}_{s \sim \sigma_T}[C_i(s_i', s_{-i})] + \rho_i(T)$$

for $i \in N$ and fixed $s_i' \in \mathcal{S}_i$.

Theorem

The time average σ_T is a $\rho_i(T)$-approximate CCE, i.e., it satisfies

$$\mathbb{E}_{s\sim\sigma_T}\left[C_i(s)\right] \leq \mathbb{E}_{s\sim\sigma_T}\left[C_i(s_i', s_{-i})\right] + \rho_i(T)$$

for $i \in N$ and fixed $s_i' \in \mathcal{S}_i$.

Proof (sketch): First note that expected cost $\mathbb{E}_{t_i\sim p_i^{(t)}}\left[c_i^{(t)}(t_i)\right]$ incurred by player i in round t boils down to $\mathbb{E}_{s\sim\sigma^{(t)}}\left[C_i(s)\right]$. Then

$$\begin{aligned}
\mathbb{E}_{s\sim\sigma_T}\left[C_i(s)\right] &= \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{a\sim p_i^{(t)}}\left[c_i^{(t)}(a)\right] && (\textit{time average})\\
&= \min_{s_i\in\mathcal{S}_i}\frac{1}{T}\sum_{t=1}^{T}c_i^{(t)}(s_i) + \rho_i(T) && (\textit{definition of } \rho_i(T))\\
&= \min_{s_i\in\mathcal{S}_i}\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{s_{-i}^{(t)}\sim\sigma_{-i}^{(t)}}C_i\left(s_i, s_{-i}^{(t)}\right) + \rho_i(T) && (\textit{definition of } c_i^{(t)})\\
&\leq \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}_{s^{(t)}\sim\sigma^{(t)}}C_i\left(s_i', s_{-i}^{(t)}\right) + \rho_i(T) && (\textit{plugging in fixed } s_i')\\
&= \mathbb{E}_{s\sim\sigma_T}C_i\left(s_i', s_{-i}\right) + \rho_i(T) && (\textit{time average}) \quad \square
\end{aligned}$$

No-regret dynamics

Multiplicative Weights algorithm

No-regret dynamics

No-regret (player) dynamics

In every round $t = 1, \dots, T$, every player $i \in N$ does the following:

- Use $\mathcal{A}_i$ to compute prob. distr. $p_i^{(t)} = (p_{i,1}^{(t)}, \dots, p_{i,m_i}^{(t)})$ over $\mathcal{S}_i$.
- The adversarial cost vector $c_i^{(t)} : \mathcal{S}_i \to [0, 1]$ is defined as

$$c_i^{(t)}(a) = \mathbb{E}_{s_{-i}^{(t)} \sim \sigma_{-i}^{(t)}} C_i\left(a, s_{-i}^{(t)}\right) \quad \forall a \in \mathcal{S}_i$$

  where $\sigma_{-i}^{(t)}$ is product distribution formed by the $p_j^{(t)}$ with $j \in N \setminus \{i\}$.
- Strategy $a^{(t)} \sim p_i^{(t)}$ is drawn, and player i incurs corresponding cost.

We next give promised MW algorithm that can be used for the $\mathcal{A}_i$, and that has the no-regret property. That is, in expectation,

$$\rho_i(T) = \frac{1}{T}\left(\sum_{t=1}^{T} c_i^{(t)}(a^{(t)}) - \min_{a \in \mathcal{S}_i} \sum_{t=1}^{T} c_i^{(t)}(a)\right) \to 0$$

where $a^{(t)} \sim p_i^{(t)}$.

Multiplicative Weights (MW) algorithm

Fix player $i \in N$. The MW algorithm maintains weight $w_a^{(t)}$ for every $a \in \mathcal{S}_i$ and chooses distribution for round t as

$$p_{i,a}^{(t)} = \frac{w_a^{(t)}}{\sum_{r \in \mathcal{S}_i} w_r^{(t)}}.$$

Weight update procedure

Given is input parameter $\eta \in (0, 1/2]$.

- Initial weights are set at $w_a^{(1)} = 1$ for $a \in \mathcal{S}_i$ (uniform distribution over $\mathcal{S}_i$).
- For round $t = 1, \ldots, T$:
  - After seeing cost vector $c_i^{(t)}$, weights for round $t+1$ are defined as

$$w_a^{(t+1)} = (1-\eta)^{c_i^{(t)}(a)} \cdot w_a^{(t)}$$

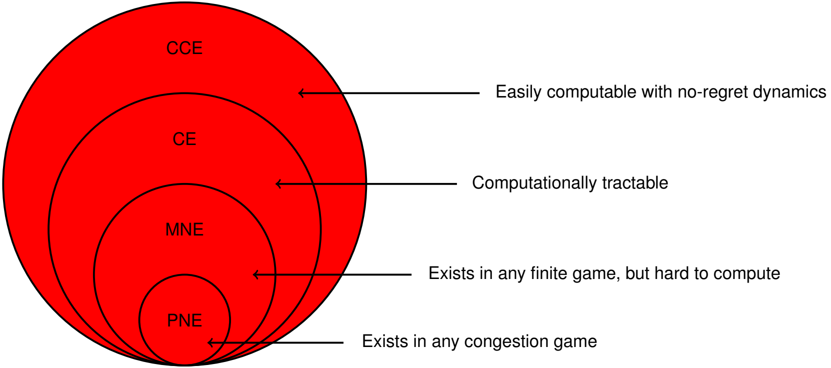
- High cost strategies get smaller (relative) weight in next round.

Theorem (Littlestone and Warmuth, 1994)

MW algorithm, with $\eta = \sqrt{\log(m_i)/T}$*, has regret* $\rho_i(T) \leq 2\sqrt{\log(m_i)/T}$

Overview

Hierarchy of equilibrium concepts



Final remarks

- CE can also be obtained through certain player dynamics.
  - See, e.g., Chapter 18 [R2016].
- Recall that PoA bounds, that we derived for PNE, extend to CCE by means of the smoothness framework.