Learning Stronger Notions of Equilibrium

Brian Zhang

Recap: CCEs in Normal-Form Games

X_i = set of **pure** strategies of player i

Correlated strategy profile:

Note: not $\Delta(X_1) \times \cdots \times \Delta(X_n)$

$$\bar{\mu}^T := \frac{1}{T}\sum_{t=1}^{T}\underbrace{(\mu_1^t \otimes \mu_2^t \otimes \cdots \mu_n^t)}_{} \in \Delta(X_1 \times \cdots \times X_n)$$

the product distribution in $\Delta(X_1) \times \cdots \times \Delta(X_n)$ *whose marginal on* X_i *is* $\mu_i^t \in \Delta(X_i)$

Regret guarantee: for all players i:

$$\max_{x_i^*}\frac{1}{T}\sum_{t=1}^{T}\left[u_i(x_i^*, x_{-i}^t) - u_i(x_i^t, x_{-i}^t)\right] \le O_n\left(\frac{1}{\sqrt{T}}\right)$$

$$= \max_{x_i^*} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim\bar{\mu}^T}\left[u_i(x_i^*, x_{-i}) - u_i(x_i, x_{-i})\right]$$

$\bar{\mu}^T$ is an ϵ-*"coarse-correlated equilibrium" (CCE)* where $\epsilon = O_n(1/\sqrt{T})$

Works for extensive-form games too: use CFR!

Coarse-Correlated Equilibria

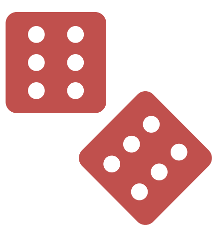
Def: $\mu \in \Delta(X_1 \times \cdots \times X_n)$ is a coarse-correlated equilibrium (CCE) if

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mu}\left[u_i(x_i^*, x_{-i}) - u_i(x_i, x_{-i})\right] \le 0$$

for all players i and all strategies $x_i^* \in X_i$

I will sample $\boldsymbol{x} \sim \mu$. You can either **commit to playing the strategy I sample**, or **play a strategy of your choice**

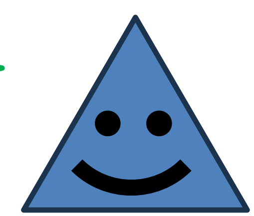
"Correlation device"
"Mediator"



CCE:

I will **commit to playing your sampled strategy**, whatever it is.

Player i



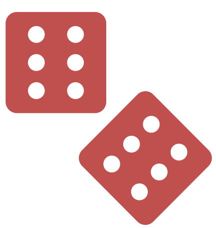
Coarse-Correlated Equilibria

Def: $\mu \in \Delta(X_1 \times \cdots \times X_n)$ is a coarse-correlated equilibrium (CCE) if

$$\mathop{\mathbb{E}}_{x \sim \mu} [u_i(x_i^*, x_{-i}) - u_i(x_i, x_{-i})] \le 0$$

for all players i and all strategies $x_i^* \in X_i$

"Correlation device"
"Mediator"

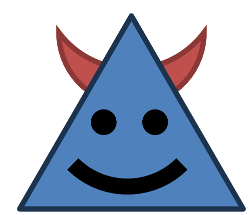


I will sample $\boldsymbol{x} \sim \mu$. You can either **commit to playing the strategy I sample**, or **play a strategy of your choice**

Not CCE:

I think $\boldsymbol{x}_i^*$ is a unilaterally profitable deviation, and I'll play that instead

Player i



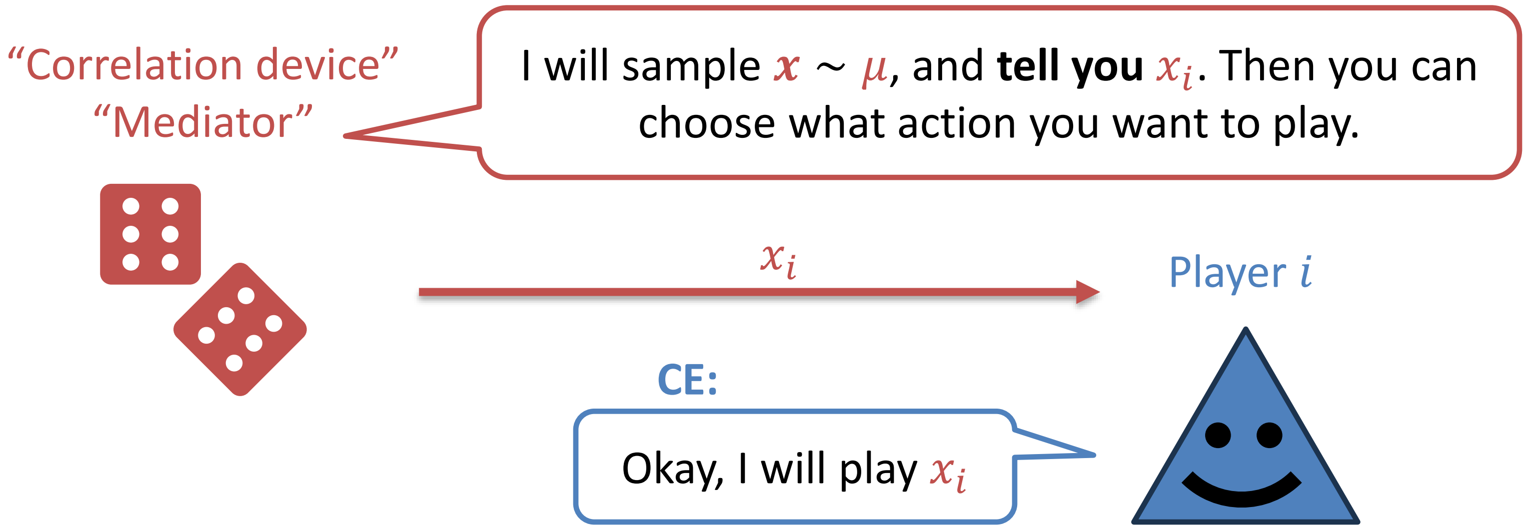
Fairly **weak notion**: Player must commit **before seeing the sampled strategy**
e.g., CCEs can include **dominated strategies** (HW1)

Correlated Equilibria

Def: $\mu \in \Delta(X_1 \times \cdots \times X_n)$ is a correlated equilibrium (CE) if

$$\mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mu} [u_i(\phi_i(x_i), x_{-i}) - u_i(x_i, x_{-i})] \leq 0$$

for all players i and all functions $\phi_i : X_i \to X_i$

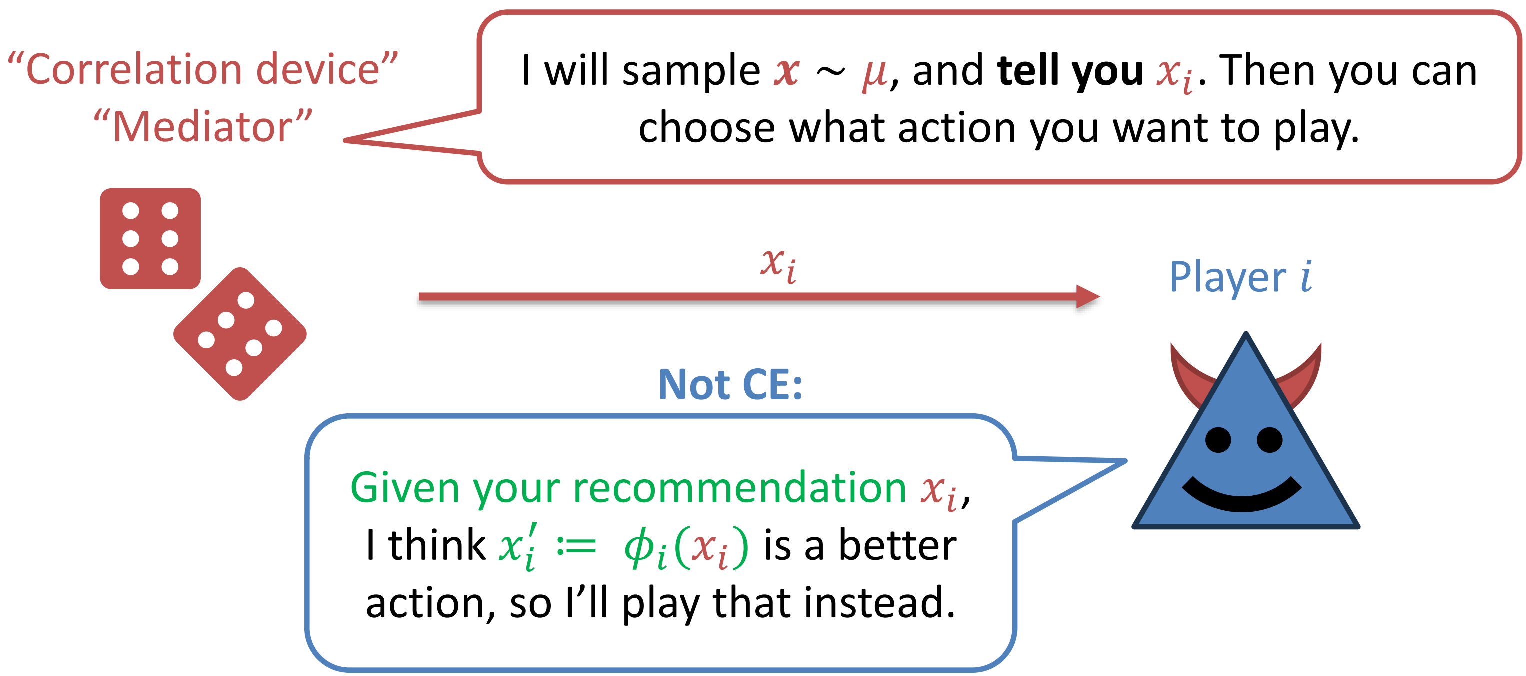


Correlated Equilibria

Def: $\mu \in \Delta(X_1 \times \cdots \times X_n)$ is a correlated equilibrium (CE) if

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mu}\left[u_i(\phi_i(x_i), x_{-i}) - u_i(x_i, x_{-i})\right] \leq 0$$

for all players i and all functions $\phi_i : X_i \to X_i$



Correlated Equilibria in Normal-Form Games

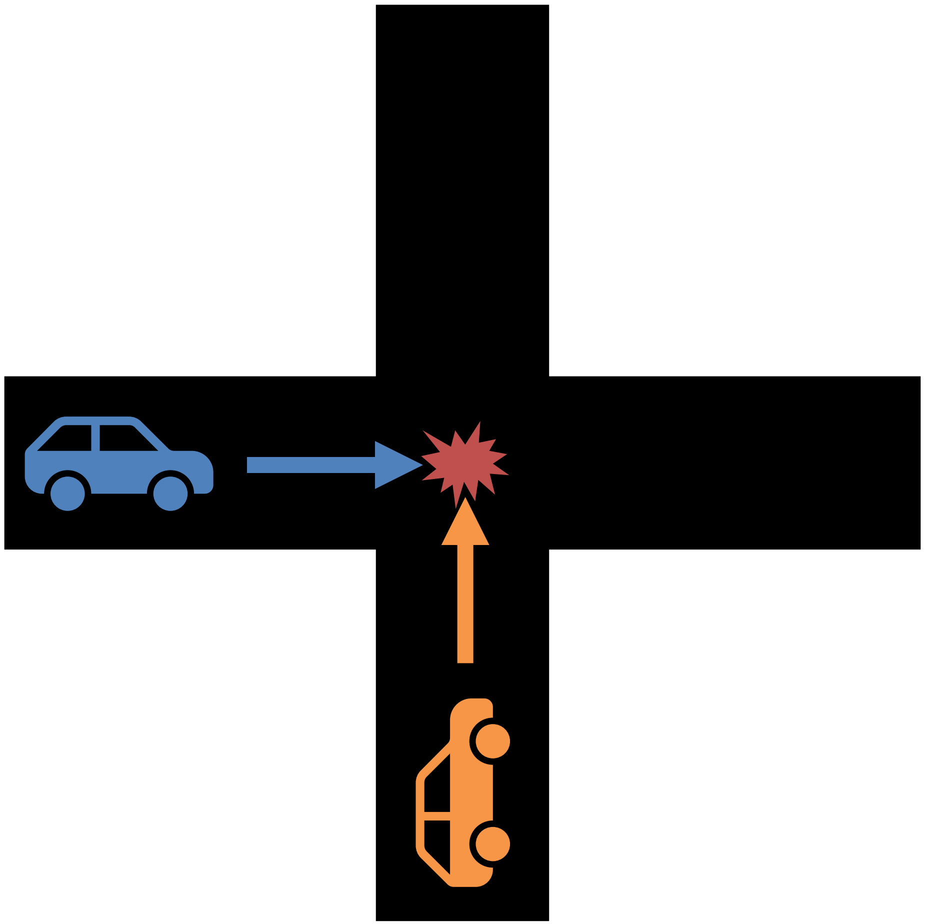
Chicken

	Stop	Go
Stop		
Go		

Correlated Equilibria in Normal-Form Games

Chicken

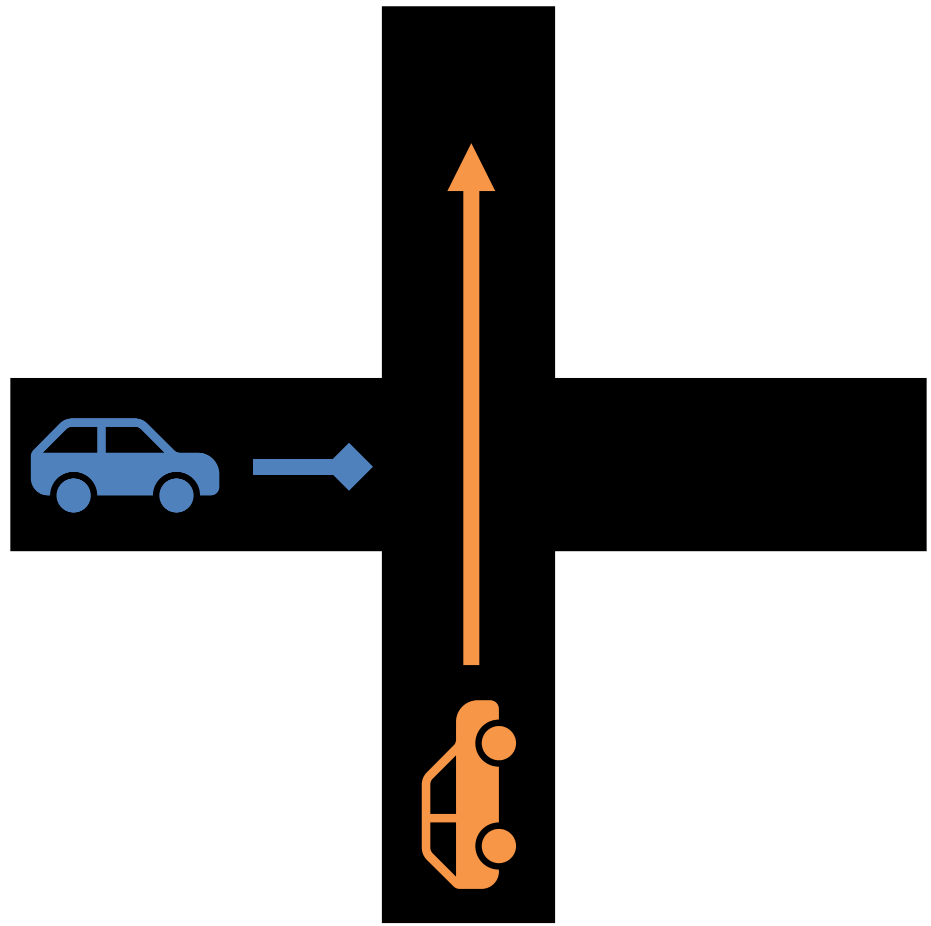
	Stop	Go
Stop		
Go		-5, -5



Correlated Equilibria in Normal-Form Games

Chicken

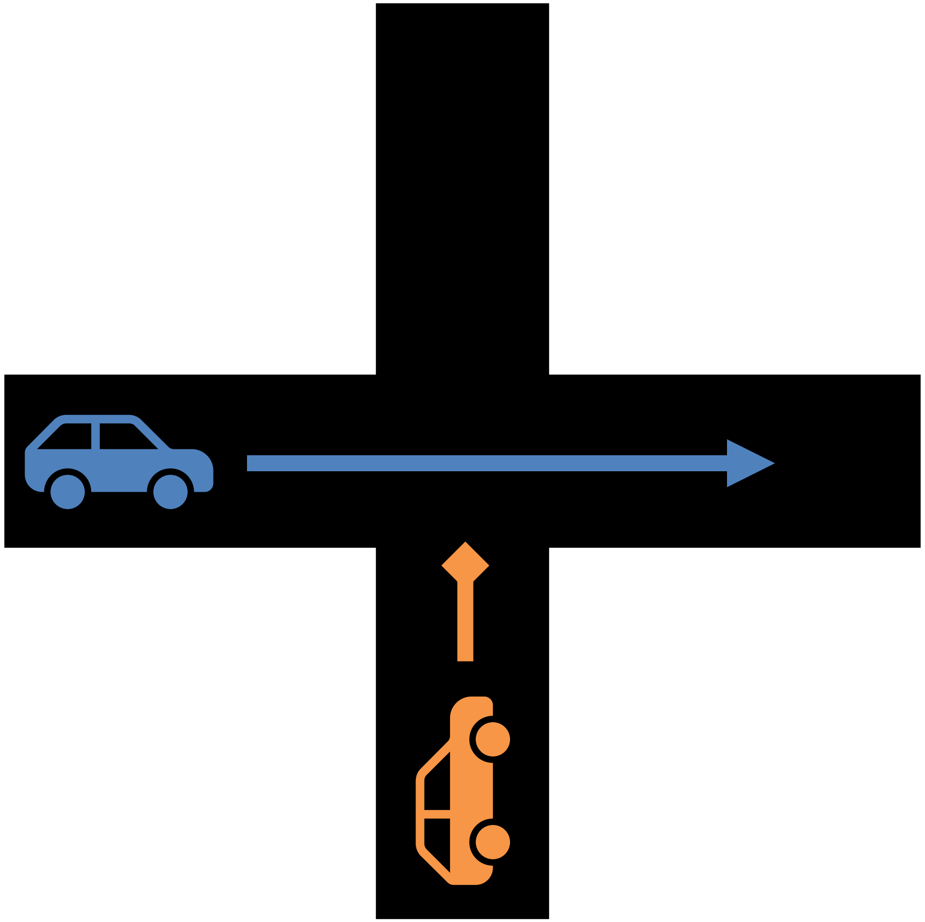
	Stop	Go
Stop		0, 1
Go		-5, -5



Correlated Equilibria in Normal-Form Games

Chicken

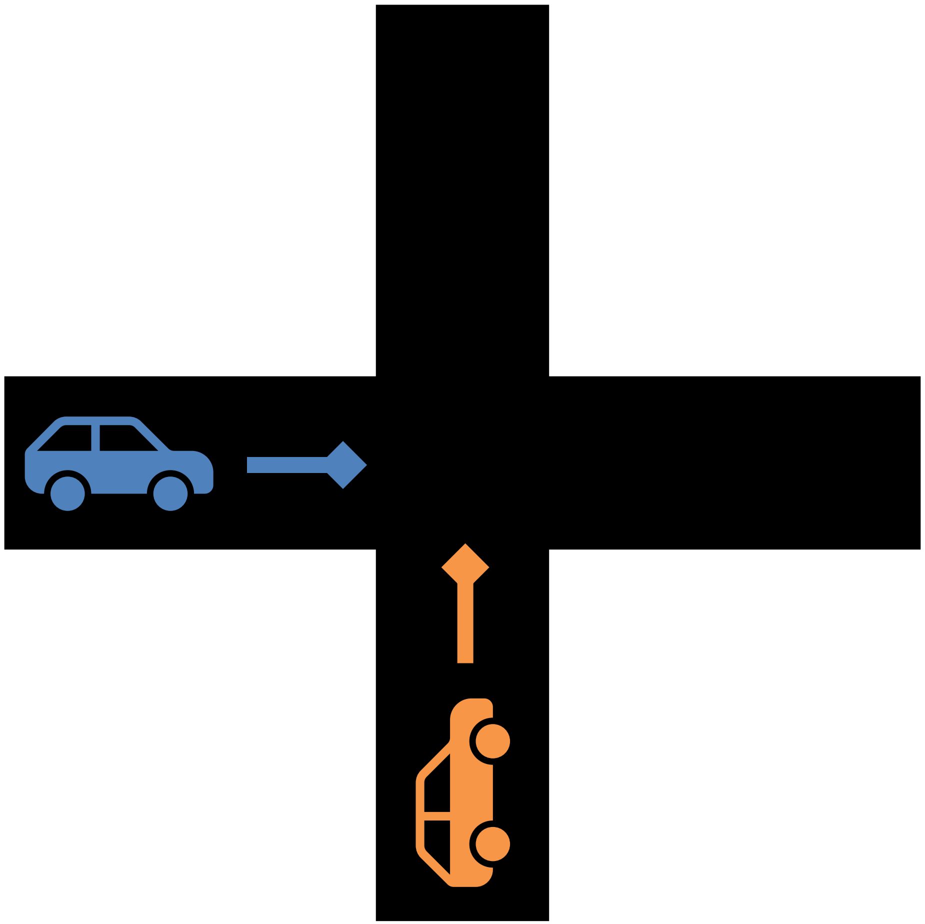
	Stop	Go
Stop		0, 1
Go	1, 0	-5, -5



Correlated Equilibria in Normal-Form Games

Chicken

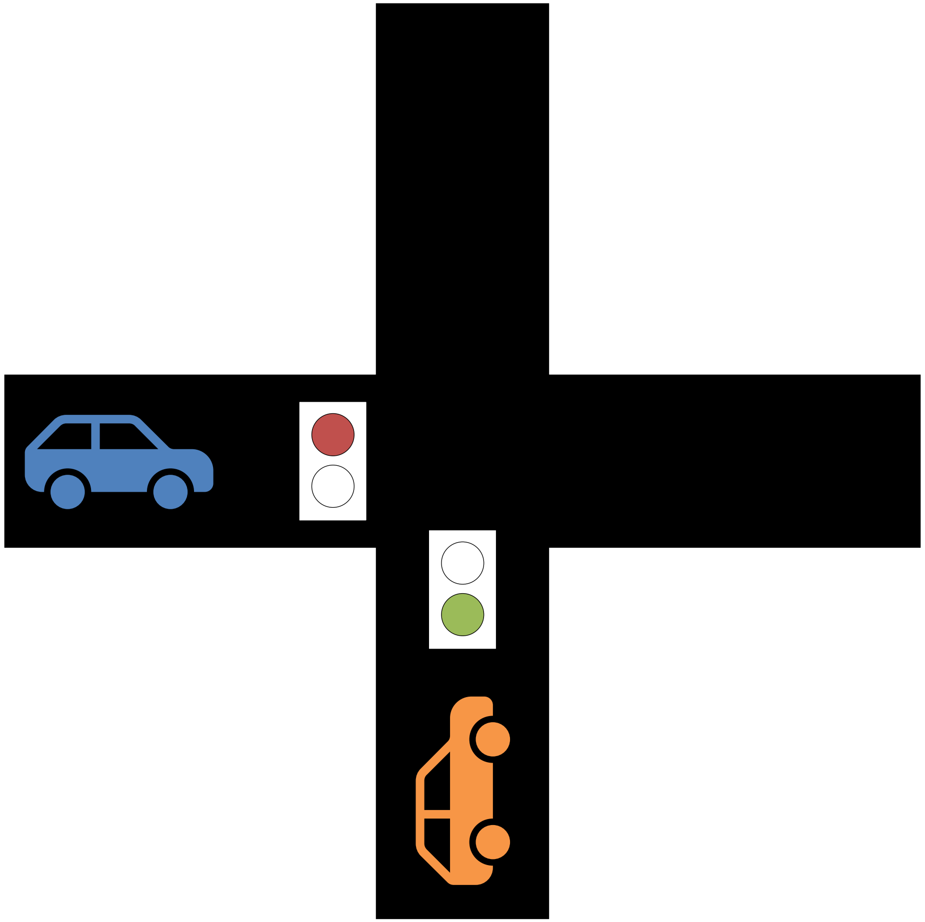
	Stop	Go
Stop	0, 0	0, 1
Go	1, 0	-5, -5



Correlated Equilibria in Normal-Form Games

Chicken

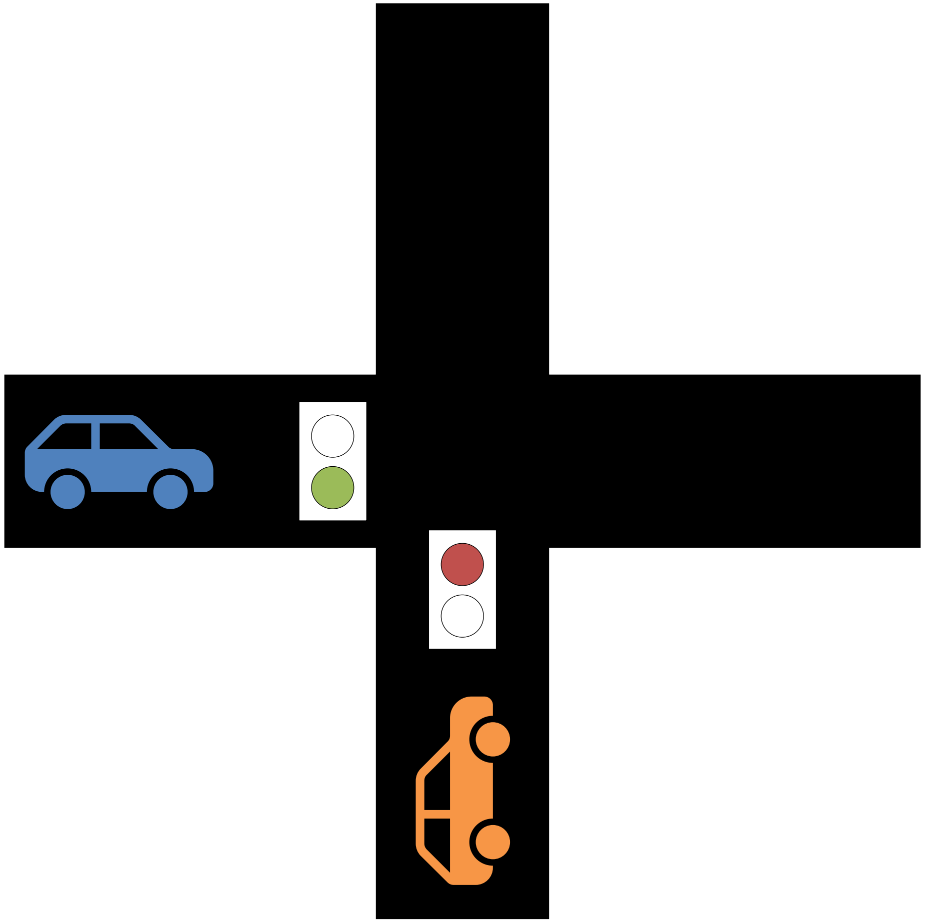
	Stop	Go
Stop	0, 0 0	0, 1 p
Go	1, 0 1-p	-5, -5 0



Correlated Equilibria in Normal-Form Games

Chicken

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



$$\mu = \frac{1}{2}(\mathrm{Stop}, \mathrm{Go}) + \frac{1}{2}(\mathrm{Go}, \mathrm{Stop})$$

is a CE (and a CCE)

CCEs can be learned using any no-regret algorithm.

Question: Can CEs?

Normal-Form Strategy Maps

A map $\phi : X \to X$, where $X \coloneqq \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\} \subset \mathbb{R}^n$, is given by a matrix $\boldsymbol{M} \in \mathbb{R}^{n \times n}$ whose ith column specifies $\phi(\boldsymbol{e}_i) \in X$.

e.g.,

$$\boldsymbol{M} = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\phi(\boldsymbol{x}) = \boldsymbol{M}\boldsymbol{x}$$

Normal-Form Strategy Maps

A randomized map $\phi : X \to \mathrm{conv}(X)$, where $X := \{\boldsymbol{e}_1, \ldots, \boldsymbol{e}_n\} \subset \mathbb{R}^n$, is given by a matrix $\boldsymbol{M} \in \mathbb{R}^{n \times n}$ whose ith column specifies $\phi(\boldsymbol{e}_i) \in \mathrm{conv}(X)$.

e.g.,

$$\boldsymbol{M} = \begin{bmatrix} 0.7 & 1 & 0.2 \\ 0.3 & 0 & 0.6 \\ 0 & 0 & 0.2 \end{bmatrix}$$

$$\phi(\boldsymbol{x}) = \boldsymbol{M}\boldsymbol{x}$$

No-(External-)Regret Learning

Pure strategy set $X \coloneqq \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_n\} \subset \mathbb{R}^n$

On each iteration:

- player outputs **mixed strategy** $\boldsymbol{x}^t \in \operatorname{conv}(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in [-1,1]^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **regret** after T timesteps

$$R_X(T) \coloneqq \max_{\boldsymbol{x}^* \in X} \sum_{t=1}^{T} \langle \boldsymbol{u}^t, \boldsymbol{x}^* - \boldsymbol{x}^t \rangle$$

No-Swap-Regret Learning

Pure strategy set $X \coloneqq \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_n\} \subset \mathbb{R}^n$

On each iteration:

- player outputs **mixed strategy** $\boldsymbol{x}^t \in \mathrm{conv}(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in [-1,1]^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **swap regret** after T timesteps

$$R_X^{\mathrm{Swap}}(T) \coloneqq \max_{\boldsymbol{M} \in S_n} \sum_{t=1}^{T} \langle \boldsymbol{u}^t, \boldsymbol{M}\boldsymbol{x}^t - \boldsymbol{x}^t \rangle$$

S_n = set of $n \times n$ stochastic matrices

Proposition:

If all players in a game achieve swap regret ϵT, then the average strategy profile $\bar{\mu}$ is an ϵ-correlated equilibrium.

The GGM Framework

Blum, Mansour (*JMLR* 2007); Gordon, Greenwald, Marks (*ICML* 2008)

Idea: Use

- a regret minimizer $\mathcal{R}_\Phi$ on S_n (stochastic matrices) with regret $R_\Phi(T)$, and
- fixed points

we'll discuss how to do this in a minute

Algorithm: For each iteration $t = 1, \dots, T$:

1. Obtain matrix $\boldsymbol{M}^t$ from $\mathcal{R}_\Phi$
2. Compute $\boldsymbol{x}^t \in \text{conv}(X)$ such that $\boldsymbol{M}^t \boldsymbol{x}^t = \boldsymbol{x}^t$
3. Play $\boldsymbol{x}^t$, observe utility $\boldsymbol{u}^t$
4. Feed to $\mathcal{R}_\Phi$ the utility $\boldsymbol{M} \mapsto \langle \boldsymbol{u}^t, \boldsymbol{M}\boldsymbol{x}^t \rangle$

Regret analysis: $$R_X^{\text{Swap}}(T) = \max_{\boldsymbol{M} \in S_n} \sum_{t=1}^{T} \langle \boldsymbol{u}^t, \boldsymbol{M}\boldsymbol{x}^t - \boldsymbol{x}^t \rangle$$

The GGM Framework

Blum, Mansour (*JMLR* 2007); Gordon, Greenwald, Marks (*ICML* 2008)

Idea: Use

- a regret minimizer $\mathcal{R}_\Phi$ on S_n (stochastic matrices) with regret $R_\Phi(T)$, and
- fixed points

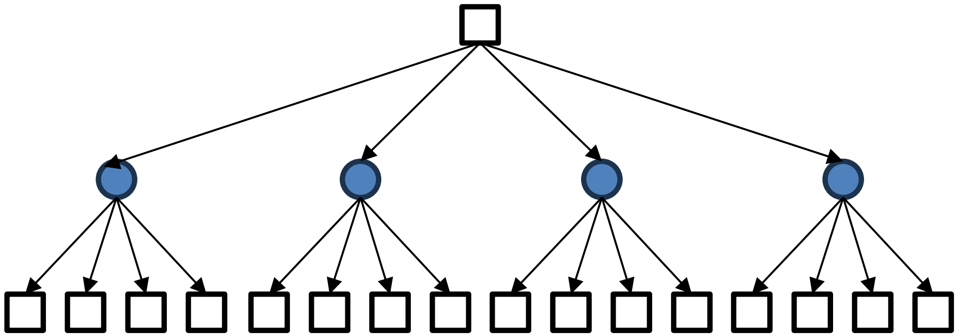
we'll discuss how to do this in a minute

Algorithm: For each iteration $t = 1, \dots, T$:

1. Obtain matrix $\boldsymbol{M}^t$ from $\mathcal{R}_\Phi$
2. Compute $\boldsymbol{x}^t \in \operatorname{conv}(X)$ such that $\boldsymbol{M}^t \boldsymbol{x}^t = \boldsymbol{x}^t$
3. Play $\boldsymbol{x}^t$, observe utility $\boldsymbol{u}^t$
4. Feed to $\mathcal{R}_\Phi$ the utility $\boldsymbol{M} \mapsto \langle \boldsymbol{u}^t, \boldsymbol{M}\boldsymbol{x}^t \rangle$

Regret analysis: $$R_X^{\text{Swap}}(T) = \max_{\boldsymbol{M} \in S_n} \sum_{t=1}^{T} \langle \boldsymbol{u}^t, \boldsymbol{M}\boldsymbol{x}^t - \boldsymbol{M}^t \boldsymbol{x}^t \rangle = R_\Phi(T)$$

Regret Minimization Over $n \times n$ Stochastic Matrices



{ Sequence-form strategies in this tree-form decision problem }

$\cong$

{ 4×4 stochastic matrices }

Use CFR!

$$R_X^{\text{Swap}}(T) = R_\Phi(T) \in \mathcal{O}\left(n\sqrt{T \log n}\right)$$

with MWU at every decision point

Tighter analysis is possible: Blum-Mansour shows $\sqrt{Tn \log n}$

Theorem [Blum & Mansour *JMLR* 2007]
There exists an algorithm for learning CE in normal-form games with convergence rate $\sqrt{(n \log n)/T}$.

More Generally: Φ-Equilibria

Def: Given a tuple of subsets $\Phi = \{\Phi_i\}_{i\in[n]}$ where $\Phi_i \subseteq X_i^{X_i}$, correlated distribution $\mu \in \Delta(X_1 \times \cdots \times X_n)$ is a Φ-equilibrium if

$$\mathop{\mathbb{E}}_{x\sim\mu}\left[u_i(\phi_i(x_i), x_{-i}) - u_i(x_i, x_{-i})\right] \le 0$$

for all players i and all functions $\phi_i \in \Phi_i$

Special cases:

- CCE (constant functions): $\Phi_i = \{\phi_{x_i^*} : x^* \in X_i\}$ where $\phi_{x_i^*}(x_i) = x_i^*$ for all x_i
- CE (all functions): $\Phi_i = X_i^{X_i}$

No-(External-)Regret Learning in Extensive-Form Games

Pure strategy set $X \subseteq \{0,1\}^n$

On each iteration:

- player outputs **tree-form strategy** $\boldsymbol{x}^t \in \mathrm{conv}(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in \mathbb{R}^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **regret** after T timesteps

$$R_X(T) \coloneqq \max_{\boldsymbol{x}^* \in X} \sum_{t=1}^{T} \langle \boldsymbol{u}^t, \boldsymbol{x}^* - \boldsymbol{x}^t \rangle$$

No-(External-)Regret Learning in Extensive-Form Games

Pure strategy set $X \subseteq \{0,1\}^n$

On each iteration:

- player outputs **mixed strategy** $\mu^t \in \Delta(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in \mathbb{R}^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **regret** after T timesteps

$$R_X(T) := \max_{\boldsymbol{x}^* \in X} \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \boldsymbol{x}^* - \boldsymbol{x}^t \rangle$$

No-Φ-Regret Learning

Pure strategy set $X \subseteq \{0,1\}^n$, set of deviations $\Phi \subseteq X^X$

On each iteration:

- player outputs **mixed strategy** $\mu^t \in \Delta(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in \mathbb{R}^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **Φ-regret** after T timesteps

$$R_X^{\Phi}(T) \coloneqq \max_{\phi \in \Phi} \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \phi(\boldsymbol{x}^t) - \boldsymbol{x}^t \rangle$$

Φ	Notion of Regret	Corresponding Notion of Equilibrium
$\Phi_{\text{Ext}} = \{\text{constant functions}\}$	External	Coarse-Correlated
$\Phi_{\text{Swap}} = X^X$ (all functions)	Swap	Correlated

No-Φ-Regret Learning

Pure strategy set $X \subseteq \{0,1\}^n$, set of deviations $\Phi \subseteq X^X$

On each iteration:

- player outputs **mixed strategy** $\mu^t \in \Delta(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in \mathbb{R}^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\mathop{\mathbb{E}}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **Φ-regret** after T timesteps

$$R_X^\Phi(T) := \max_{\phi \in \Phi} \sum_{t=1}^{T} \mathop{\mathbb{E}}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \phi(\boldsymbol{x}^t) - \boldsymbol{x}^t \rangle$$

Proposition

If all players in a game run Φ-regret minimizers that achieve Φ-regret ϵT, then the average strategy profile $\bar{\mu}$ is an ϵ-approximate Φ-equilibrium.

Swap Regret in Extensive-Form Games

Q: Can **swap regret** be efficiently minimized in *extensive-form* games?

Theorem

[Corollary of Blum-Mansour]

There exists a swap regret minimizer for tree-form strategy sets whose swap regret is ϵT after $\mathcal{O}(n \cdot 2^n/\epsilon^2)$ **iterations.**

Bad per-iteration complexity and convergence rate

Theorem

[*Special case of* Peng & Rubinstein *STOC'*24; Dagan, Daskalakis, Fishelson, Golowich *STOC'*24]

There exists a swap regret minimizer for tree-form strategy sets* whose swap regret is ϵT after $n^{\tilde{\mathcal{O}}(1/\epsilon)}$ **iterations.**

*or, indeed, any set $X \subset \mathbb{R}^n$ for which *external* regret is minimizable

⇒ For **constant** ϵ, an ϵ-CE can be computed in **polynomial time!**

Theorem

[Daskalakis, Farina, Golowich, Sandholm, Zhang *arXiv'*24]

There is a constant $c > 0$ such that achieving swap regret ϵT in tree-form strategy sets requires $\exp\left(\Omega(\min\{n, 1/\epsilon\}^c)\right)$ **iterations.**

Open question: Can ϵ-CE be computed in time $\mathrm{poly}(n, 1/\epsilon)$ or even $\mathrm{poly}(n, \log(1/\epsilon))$?

(using something other than adversarial no-swap-regret learning)

Digression: Nonlinear strategy maps

Pure strategy set $X \subseteq \{0,1\}^n$, set of deviations $\Phi \subseteq X^X$

External regret minimizer on X outputs points in $\mathrm{conv}(X)$

Q: For $\boldsymbol{x}^* \in \mathrm{conv}(X)$ and $\phi : X \to X$, what does $\phi(\boldsymbol{x}^*)$ mean?

A1: When $X = \{\boldsymbol{e}_1, \dots, \boldsymbol{e}_n\}$ is a normal-form strategy set, $\mathrm{conv}(X) = \Delta(X)$ and $\phi(\boldsymbol{x}) = \boldsymbol{M}\boldsymbol{x}$ for some $\boldsymbol{M}$, so we can set $\phi(\boldsymbol{x}^*) = \sum_i \boldsymbol{x}_i^* \phi(\boldsymbol{e}_i) = \boldsymbol{M}\boldsymbol{x}^*$.

A2: Take **any** distribution $\mu \in \Delta(X)$ with $\boldsymbol{x}^* = \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mu} \boldsymbol{x}$, and define

$$\phi(\boldsymbol{x}^*) = \mathop{\mathbb{E}}_{\boldsymbol{x} \sim \mu} \phi(\boldsymbol{x}).$$

Warning: When ϕ is nonlinear, this depends on the choice of μ

⇒ "Kuhn's theorem fails when considering nonlinear deviations"

A3: When Φ consists only of linear maps, this doesn't matter (we can use sequence-form strategies + set $\phi(\boldsymbol{x}) = \boldsymbol{M}\boldsymbol{x}$

No-Linear-Swap-Regret Learning

Pure strategy set $X \subseteq \{0,1\}^n$,

On each iteration:

- player outputs **mixed strategy** $\mu^t \in \Delta(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in \mathbb{R}^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **Φ-regret** after T timesteps

$$R_X^\Phi(T) \coloneqq \max_{\boldsymbol{M} \in \Phi_{\text{LIN}}} \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \boldsymbol{M}\boldsymbol{x}^t - \boldsymbol{x}^t \rangle$$

$$\Phi_{\text{LIN}} = \{\boldsymbol{M} : \boldsymbol{M}\boldsymbol{x} \in \text{conv}(X) \quad \forall \boldsymbol{x} \in \text{conv}(X)\}$$

Advantages:

- Natural generalization of stochastic matrices for normal-form games
- GGM applies verbatim, and fixed points are easy (linear program: $\boldsymbol{M}\boldsymbol{x} = \boldsymbol{x},\ \boldsymbol{x} \in \text{conv}(X)$)

No-Linear-Swap-Regret Learning

Pure strategy set $X \subseteq \{0,1\}^n$,

On each iteration:

- player outputs **tree-form strategy** $\boldsymbol{x}^t \in \operatorname{conv}(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in \mathbb{R}^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **Φ-regret** after T timesteps

$$R_X^{\Phi}(T) := \max_{\boldsymbol{M} \in \Phi_{\mathrm{LIN}}} \sum_{t=1}^{T} \langle \boldsymbol{u}^t, \boldsymbol{M}\boldsymbol{x}^t - \boldsymbol{x}^t \rangle$$

$$\Phi_{\mathrm{LIN}} = \{\boldsymbol{M} : \boldsymbol{M}\boldsymbol{x} \in \operatorname{conv}(X) \quad \forall \boldsymbol{x} \in \operatorname{conv}(X)\}$$

Advantages:

- Natural generalization of stochastic matrices for normal-form games
- GGM applies verbatim, and fixed points are easy (linear program: $\boldsymbol{M}\boldsymbol{x} = \boldsymbol{x},\ \boldsymbol{x} \in \operatorname{conv}(X)$)
- We can still work with tree-form strategies (linearity of expectation)

The GGM Framework

Gordon, Greenwald, Marks (*ICML* 2008)

GGM requires two things.

- fixed point oracle $\text{fix} : \Phi_{\text{LIN}} \to \text{conv}(X)$, *i.e.*, $\boldsymbol{Mx} = \boldsymbol{x}$ if $\boldsymbol{x} = \text{fix}(\boldsymbol{M})$, and
  Still easy! Use linear programming or power iteration

- a regret minimizer $\mathcal{R}_\Phi$ on Φ_{LIN}
  How to characterize Φ_{LIN}?

So what does Φ_{LIN} look like?

Warm-up (Special case): What are the affine maps

$$\phi : [0,1]^n \to [0,1]?$$

- Constant functions:

$$\phi(\boldsymbol{x}) = 0, \qquad \phi(\boldsymbol{x}) = 1$$

- Functions that depend on one input coordinate:

$$\phi(\boldsymbol{x}) = x_i, \qquad \phi(\boldsymbol{x}) = 1 - x_i$$

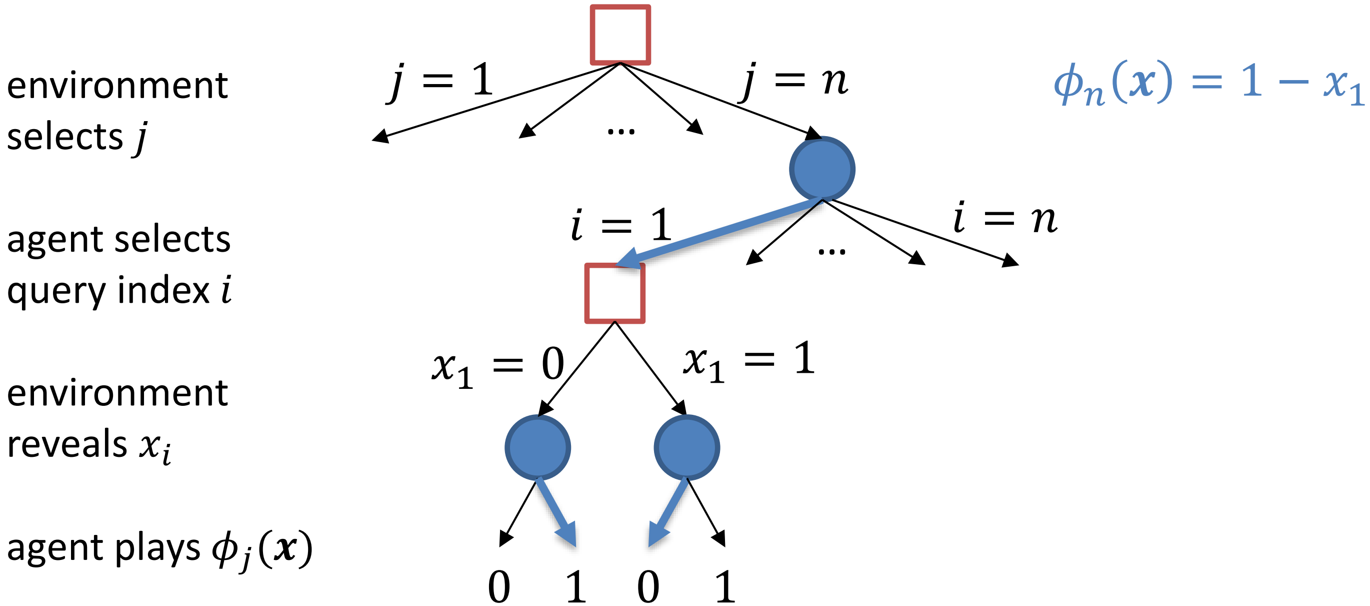
Claim: Every affine $\phi : [0,1]^n \to [0,1]$ is a convex combination of these!

So what does Φ_{LIN} look like?

Warm-up (Special case): What are the affine maps

$$\phi : [0,1]^n \to [0,1]^{n}?$$

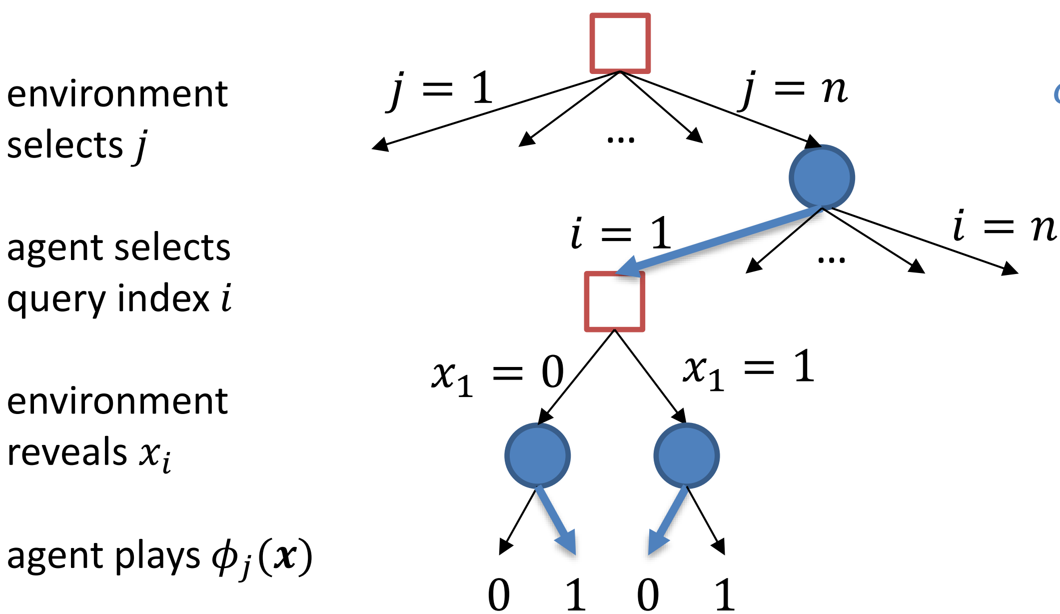
Each coordinate j is an affine map $\phi_j : [0,1]^n \to [0,1]$
$\Rightarrow$ Each ϕ_j makes ≤ 1 query to the input



So what does Φ_{LIN} look like?

Insight:

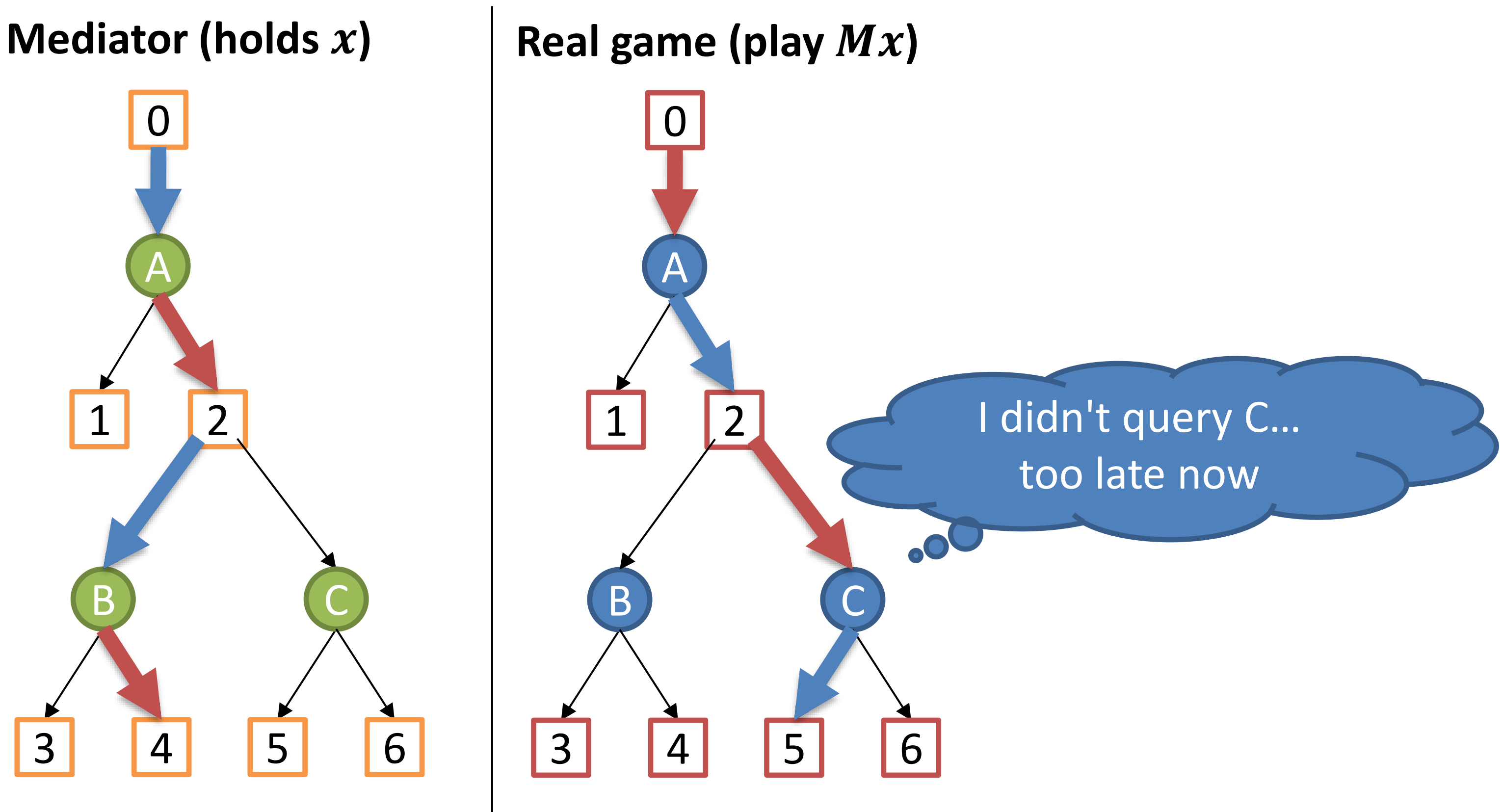
Affine maps $\phi : [0,1]^n \to [0,1]^n$ $\equiv$ Tree-form strategies with **one query**



$\phi_n(\boldsymbol{x}) = 1 - x_1$

Does this generalize?

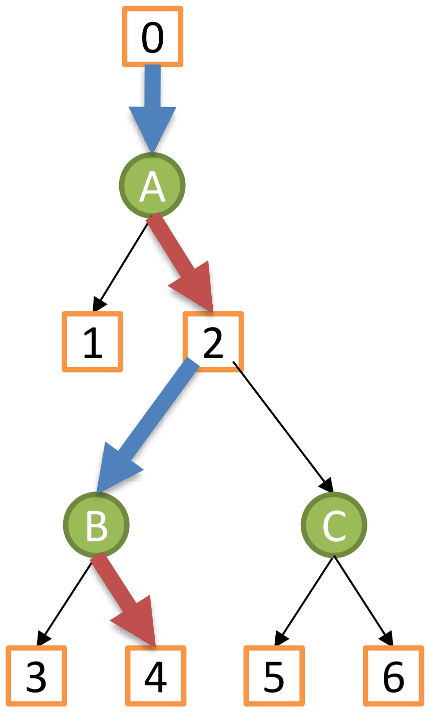
What is the generalization of a "query" to an arbitrary tree-form strategy space?



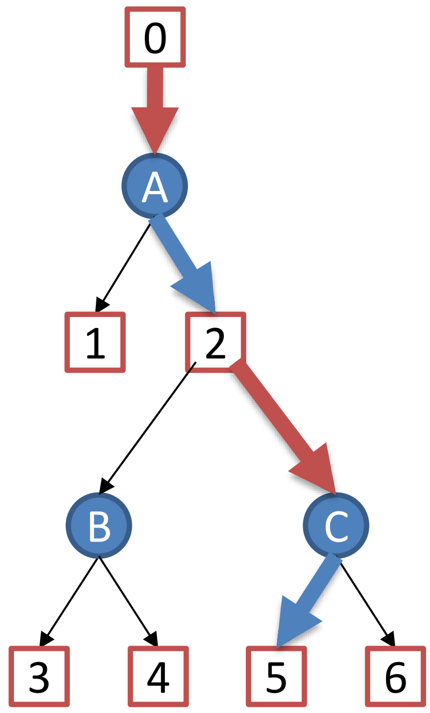
Does this generalize?

What is the generalization of a "query" to an arbitrary tree-form strategy space?

Mediator (holds x)



Real game (play Mx)



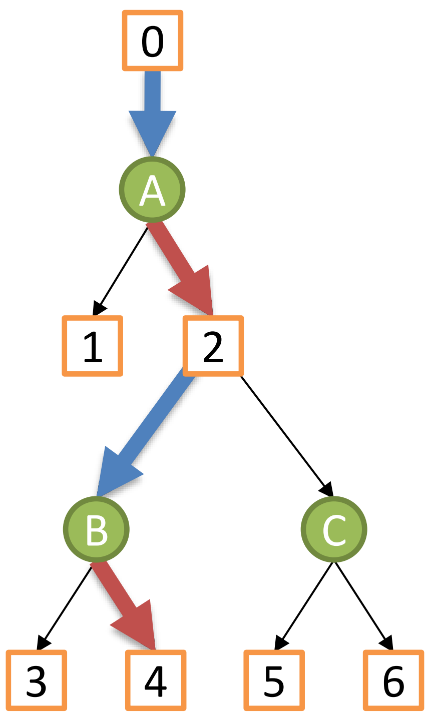
These are the **untimed communication (UTC) deviations**

Communication: Player has two-way communication with mediator to gain information

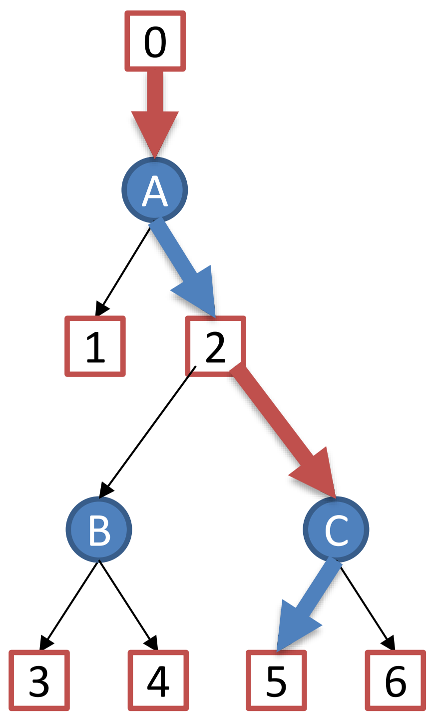
Untimed: Player can send zero, one, or multiple queries between real game actions

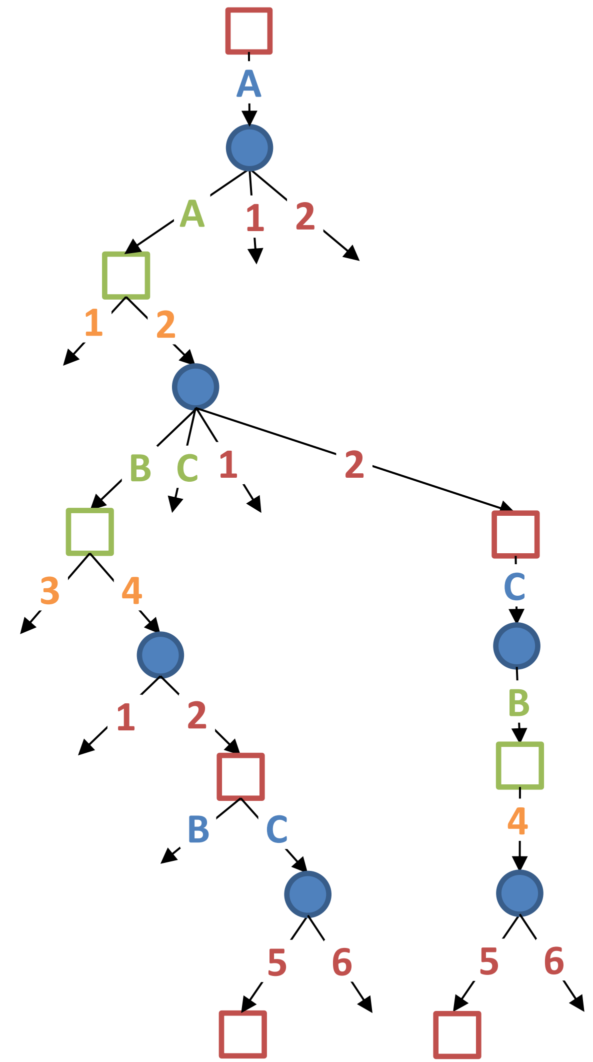
Untimed communication deviations as tree-form decision problems

Mediator (holds x)



Real game (play Mx)



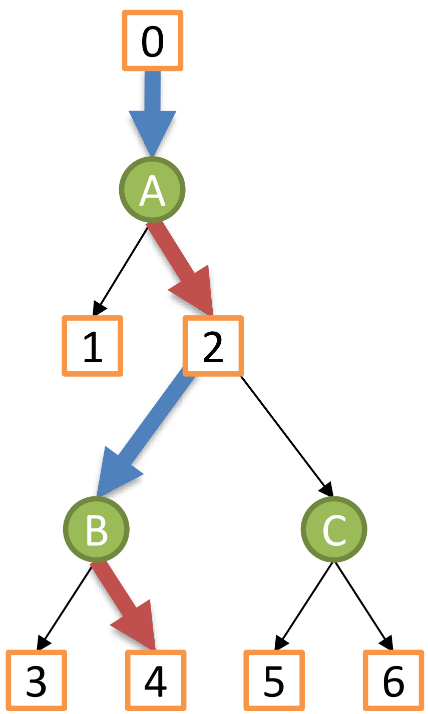


Untimed communication deviations as ~~tree~~-form decision problems

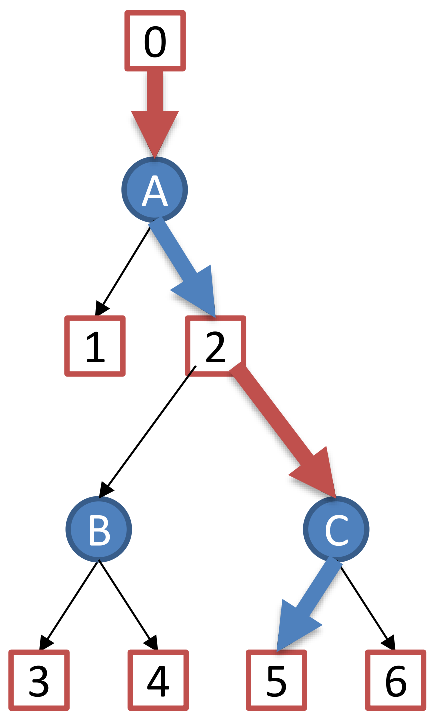
DAG

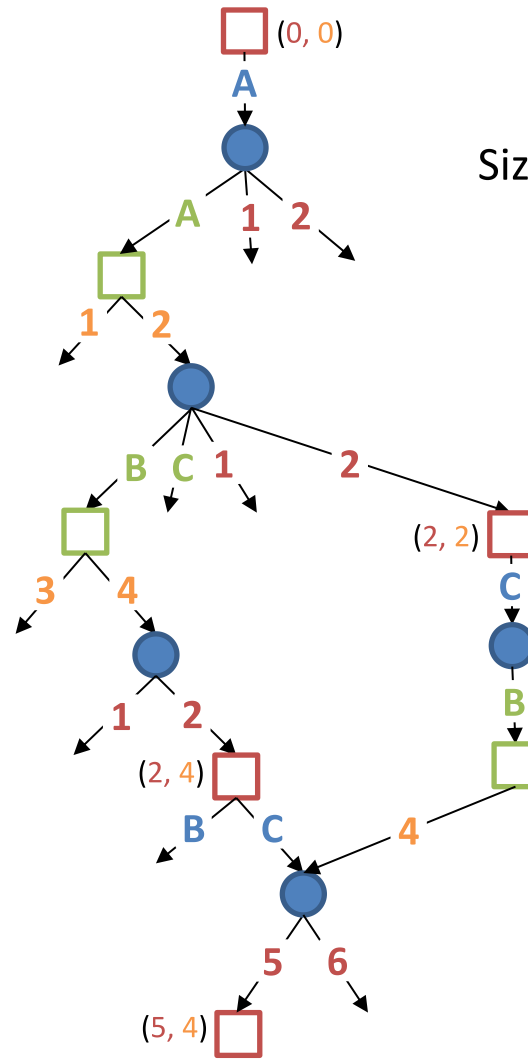
Strategy in DAG ⇒ function $\phi : X \to X$

Mediator (holds x)



Real game (play Mx)





Size of DAG:
$O(n^2)$

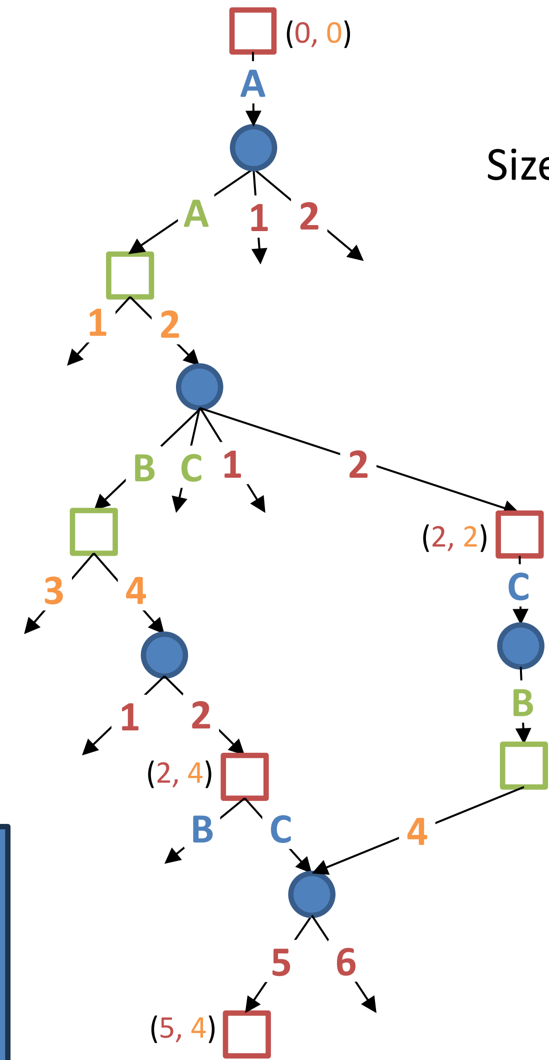
Untimed communication deviations as ~~tree~~-form decision problems

DAG

$$\underbrace{\phi(\boldsymbol{x})[\sigma]}_{\phi(x)\text{ plays to }\sigma} = \sum_{\sigma'} \underbrace{M[\sigma,\sigma']}_{\phi(x)\text{ plays to }\sigma\text{ if }x\text{ plays to }\sigma'} \underbrace{x[\sigma']}_{x\text{ plays to }\sigma'}$$

$$\phi(\boldsymbol{x}) = \boldsymbol{Mx}$$

$$\Rightarrow \Phi_{\text{UTC}} \subseteq \Phi_{\text{LIN}}$$



Size of DAG: $O(n^2)$

THEOREM

[Zhang, Farina, Sandholm ICLR'24]

$\Phi_{\text{UTC}} = \Phi_{\text{LIN}}$.

The UTC functions are
exactly the linear functions
[Zhang, Farina, Sandholm *ICLR*'24]

\+

Regret minimization on DAGs of size $m = n^2$
is possible with regret $m\sqrt{T}$ using CFR + scaled extensions
[Zhang, Farina, Sandholm *ICML*'23]

\+

Fixed-point solving using LP or power iteration

↓ GGM

COROLLARY
[Zhang, Farina, Sandholm *ICLR*'24]
Φ_{LIN}-regret minimization on tree-form decision problems is possible with regret $n^2\sqrt{T}$

Beyond Linear Deviations

Pure strategy set $X \subseteq \{0,1\}^n$, set of deviations $\Phi \subseteq X^X$

On each iteration:

- player outputs **mixed strategy** $\mu^t \in \Delta(X)$
- environment outputs (possibly adversarial) **utility vector** $\boldsymbol{u}^t \in \mathbb{R}^n$
- player observes $\boldsymbol{u}^t$ and gets reward $\mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \boldsymbol{x}^t \rangle \in [-1,1]$

Goal: minimize **Φ-regret** after T timesteps

$$R_X^{\Phi}(T) \coloneqq \max_{\phi \in \Phi} \sum_{t=1}^{T} \mathbb{E}_{\boldsymbol{x}^t \sim \mu^t} \langle \boldsymbol{u}^t, \phi(\boldsymbol{x}^t) - \boldsymbol{x}^t \rangle$$

The GGM Framework

Gordon, Greenwald, Marks (*ICML* 2008)

Pure strategy set $X \subseteq \{0,1\}^n$, set of deviations $\Phi \subseteq X^X$

GGM requires two things.

- Fixed point oracle $\mathrm{fix} : \Phi \to \mathrm{conv}(X)$, *i.e.*, $\phi(\boldsymbol{x}) = \boldsymbol{x}$ if $\boldsymbol{x} = \mathrm{fix}(\phi)$

  Problem: *$\phi : X \to X$ is a discrete function!*

  - *It may not have a fixed point*
  - *Even if we make some assumption like ϕ being continuous, fixed points are PPAD-hard to compute*

- Regret minimizer $\mathcal{R}_\Phi$ on Φ

  Problem: *if $X = \{0,1\}^n$ then $|\Phi| > 2^{n \cdot 2^n}$. How can we hope to minimize regret efficiently?*

The GGM Framework: Upgraded

Zhang, Anagnostides, Farina, Sandholm (arXiv 2024)

Pure strategy set $X \subseteq \{0,1\}^n$, set of deviations $\Phi \subseteq X^X$

GGM requires two things.

- **Expected** fixed point oracle $\text{fix}: \Phi \to \Delta(X)$, *i.e.*, $\mathbb{E}_{\boldsymbol{x}\sim\mu} \boldsymbol{x} = \mathbb{E}_{\boldsymbol{x}\sim\mu} \phi(\boldsymbol{x})$ if $\mu = \text{fix}(\phi)$
  - Always exist
  - Easy to compute! $\mu \coloneqq \text{Unif}\{\boldsymbol{x}, \phi(\boldsymbol{x}), \phi^2(\boldsymbol{x}), \dots, \phi^{L-1}(\boldsymbol{x})\}$ satisfies

$$\mathbb{E}_{\boldsymbol{x}\sim\mu}[\phi(\boldsymbol{x}) - \boldsymbol{x}] = \frac{1}{L}\sum_{\ell=0}^{L-1}\left[\phi^{\ell+1}(\boldsymbol{x}) - \phi^{\ell}(\boldsymbol{x})\right] = \frac{1}{L}\left[\phi^L(\boldsymbol{x}) - \boldsymbol{x}\right] \to 0$$

- Regret minimizer $\mathcal{R}_\Phi$ on Φ

  When $\Phi = \{\text{degree-}k \text{ polynomials}\}$ and the game tree is balanced, regret minimizers with regret $\exp\big(\text{poly}(k, \log n)\big)\sqrt{T}$ exist

Theorem: There exist efficient regret minimizers with regret $\exp\big(\text{poly}(k, \log n)\big)\sqrt{T}$ against the set Φ_k of degree-k polynomials.

Swap Regret in Extensive-Form Games

Q: Can **swap regret** be efficiently minimized in *extensive-form* games?

Theorem

[Corollary of Blum-Mansour]

There exists a swap regret minimizer for tree-form strategy sets whose swap regret is ϵT after $\mathcal{O}(n \cdot 2^n/\epsilon^2)$ **iterations.**

Bad per-iteration complexity and convergence rate

Theorem

[*Special case of* Peng & Rubinstein *STOC'*24; Dagan, Daskalakis, Fishelson, Golowich *STOC'*24]

There exists a swap regret minimizer for tree-form strategy sets* whose swap regret is ϵT after $n^{\tilde{\mathcal{O}}(1/\epsilon)}$ **iterations.**

*or, indeed, any set $X \subset \mathbb{R}^n$ for which *external* regret is minimizable

⇒ For **constant** ϵ, an ϵ-CE can be computed in **polynomial time!**

Theorem

[Daskalakis, Farina, Golowich, Sandholm, Zhang *arXiv'*24]

There is a constant $c > 0$ such that achieving swap regret ϵT in tree-form strategy sets requires $\exp\left(\Omega(\min\{n, 1/\epsilon\}^c)\right)$ **iterations.**

Open question: Can ϵ-CE be computed in time $\mathrm{poly}(n, 1/\epsilon)$ or even $\mathrm{poly}(n, \log(1/\epsilon))$?

(using something other than adversarial no-swap-regret learning)

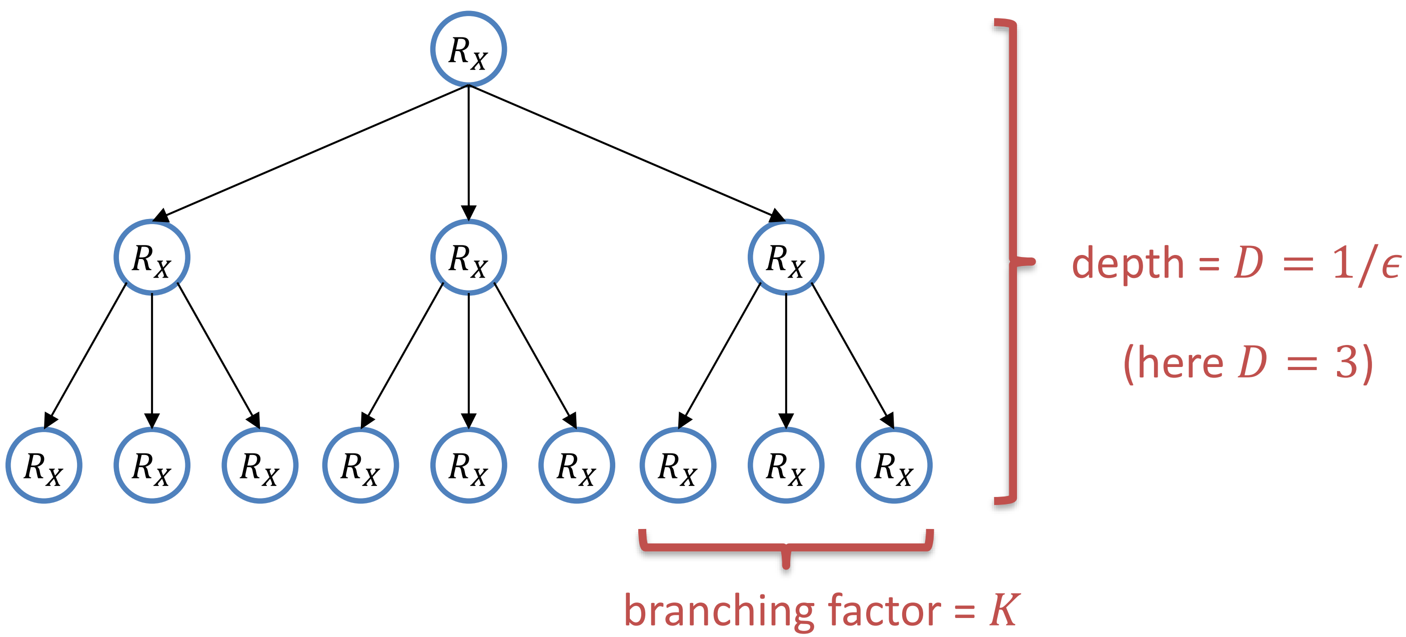
TreeSwap

Peng & Rubinstein (*STOC'24*); Dagan, Daskalakis, Fishelson, Golowich (*STOC'24*)

Given: External regret minimizer R_X on $X \subset [0,1]^n$ achieving ϵK regret after K steps (e.g., for extensive-form games, CFR gives $K = n^2/\epsilon^2$)

Goal: Build a **swap regret minimizer** on X

Idea:



TreeSwap

Peng & Rubinstein (*STOC'24*); Dagan, Daskalakis, Fishelson, Golowich (*STOC'24*)

Given: External regret minimizer R_X on $X \subset [0,1]^n$ achieving ϵK regret after K steps (e.g., for extensive-form games, CFR gives $K = n^2/\epsilon^2$)

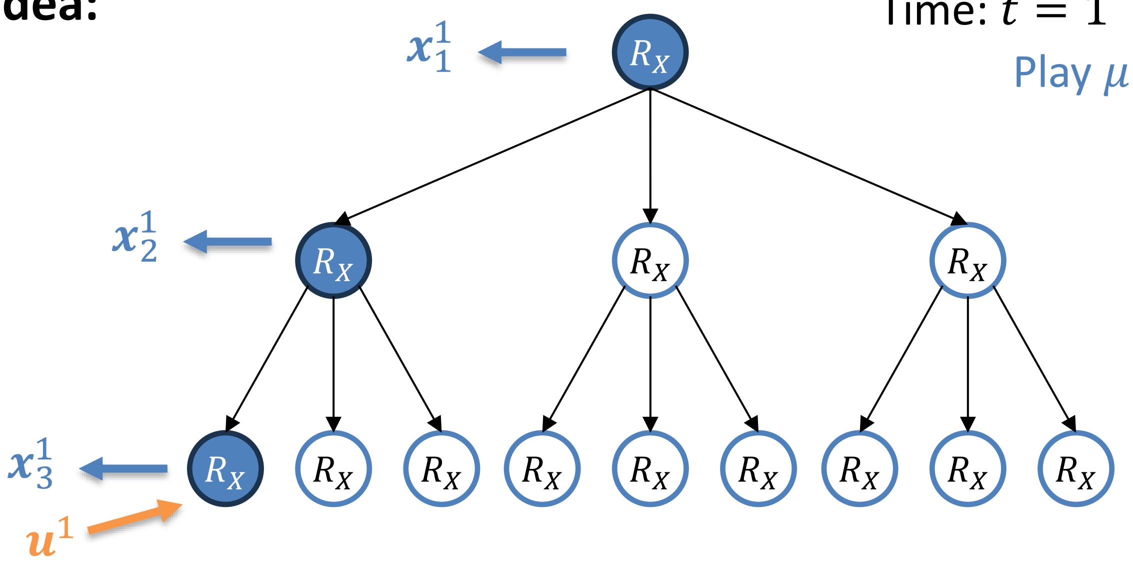
Goal: Build a **swap regret minimizer** on X

Idea:

Time: $t = 1$

Play $\mu^1 := \mathrm{Unif}\{\boldsymbol{x}_1^1, \dots, \boldsymbol{x}_{D-1}^1, \boldsymbol{x}_D^1\}$

(Mixed strategy!)



TreeSwap

Peng & Rubinstein (*STOC'*24); Dagan, Daskalakis, Fishelson, Golowich (*STOC'*24)

Given: External regret minimizer R_X on $X \subset [0,1]^n$ achieving ϵK regret after K steps (e.g., for extensive-form games, CFR gives $K = n^2/\epsilon^2$)

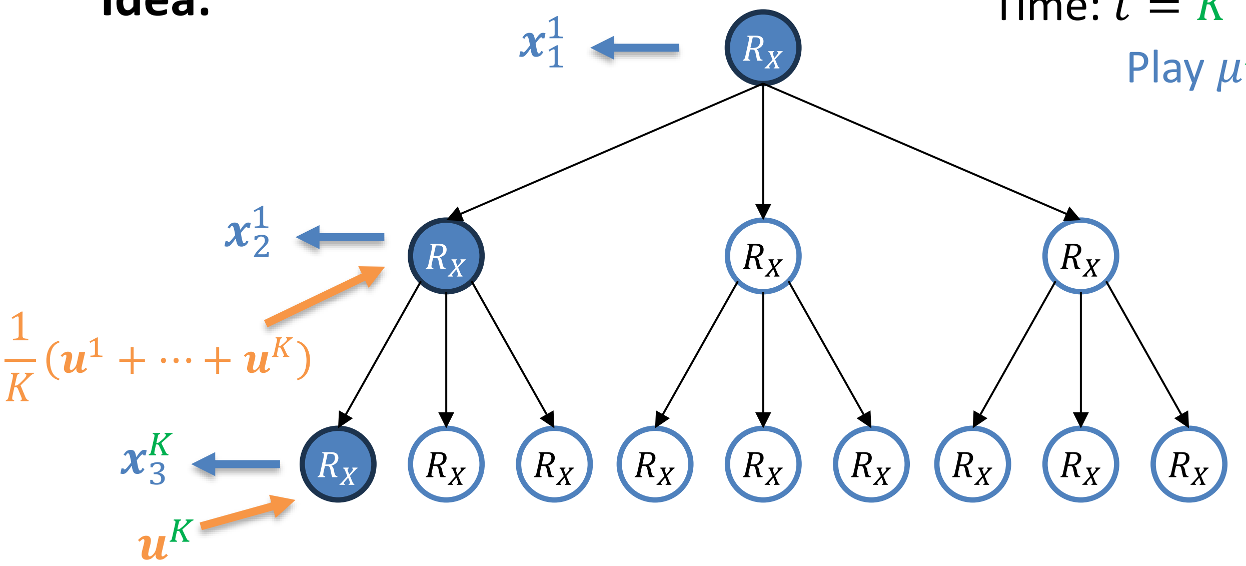
Goal: Build a **swap regret minimizer** on X

Idea:

Time: $t = K$

Play $\mu^K := \text{Unif}\{\boldsymbol{x}_1^1, \dots, \boldsymbol{x}_{D-1}^1, \boldsymbol{x}_D^K\}$

(Mixed strategy!)



TreeSwap

Peng & Rubinstein (*STOC'24*); Dagan, Daskalakis, Fishelson, Golowich (*STOC'24*)

Given: External regret minimizer R_X on $X \subset [0,1]^n$ achieving ϵK regret after K steps (e.g., for extensive-form games, CFR gives $K = n^2/\epsilon^2$)

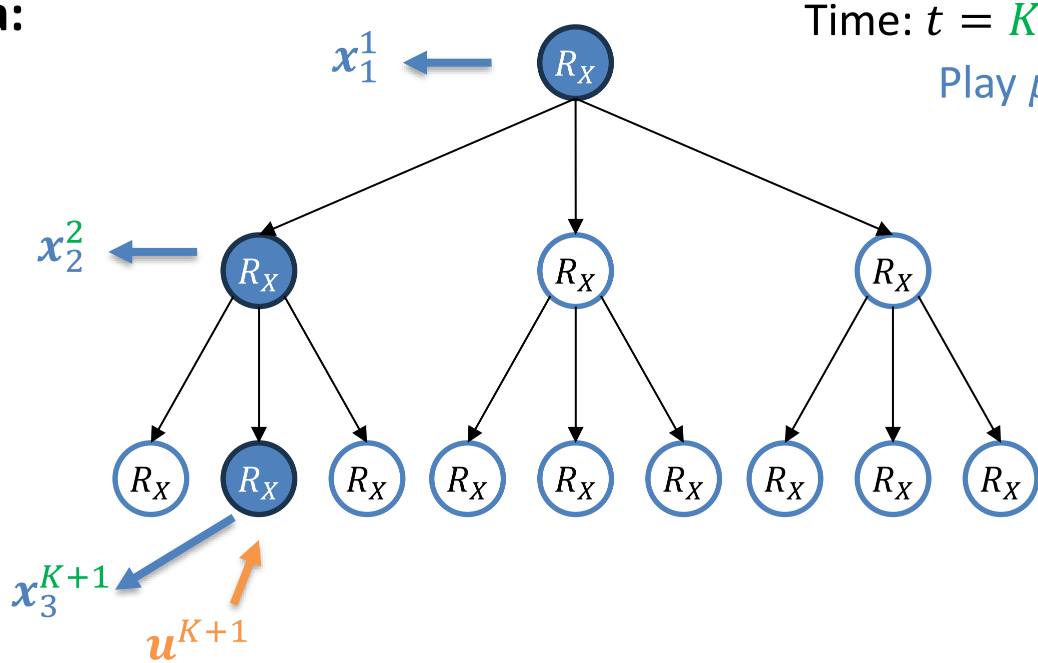
Goal: Build a **swap regret minimizer** on X

Idea:

Time: $t = K + 1$

Play $\mu^{K+1} := \mathrm{Unif}\{\boldsymbol{x}_1^1, \dots, \boldsymbol{x}_{D-1}^2, \boldsymbol{x}_D^{K+1}\}$

(Mixed strategy!)



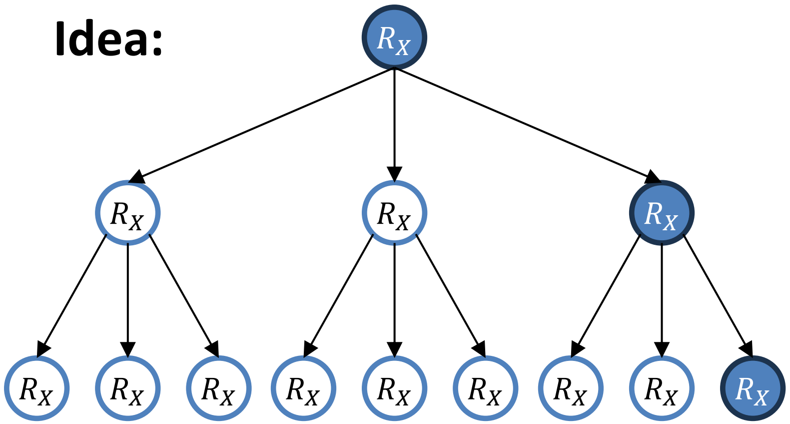
TreeSwap

Peng & Rubinstein (*STOC*'24); Dagan, Daskalakis, Fishelson, Golowich (*STOC*'24)

Given: External regret minimizer R_X on $X \subset [0,1]^n$ achieving ϵK regret after K steps (e.g., for extensive-form games, CFR gives $K = n^2/\epsilon^2$)

Goal: Build a **swap regret minimizer** on X

Idea:



Time: $t = T = K^d$

Play $\mu^{K^d} := \mathrm{Unif}\{\boldsymbol{x}_1^K, \dots, \boldsymbol{x}_{D-1}^{K^{d-1}}, \boldsymbol{x}_D^{K^d}\}$

(Mixed strategy!)

Intuition: In the GGM framework, if $\mu^t = \mathrm{Unif}\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_D\}$ let ϕ^t be the "map" that takes $\boldsymbol{x}_1 \mapsto \boldsymbol{x}_2 \mapsto \cdots \mapsto \boldsymbol{x}_D$

- $\boldsymbol{\mu}^t$ is an expected fixed point of ϕ^t
- each value of ϕ^t is being picked by regret minimizer $\Rightarrow$ Φ-regret is small!

TreeSwap

Peng & Rubinstein (*STOC'24*); Dagan, Daskalakis, Fishelson, Golowich (*STOC'24*)

Given: External regret minimizer R_X on $X \subset [0,1]^n$ achieving ϵK regret after K steps (e.g., for extensive-form games, CFR gives $K = n^2/\epsilon^2$)

Goal: Build a **swap regret minimizer** on X

Theorem:

$$R_X^{\text{Swap}}(T) \le T\left(\epsilon + \frac{1}{D}\right) \le 2\epsilon T$$

from regret of each R_X

from expected fixed point error

Time: $t = T = K^d$

Play $\mu^{K^d} := \text{Unif}\{\boldsymbol{x}_1^K, \dots, \boldsymbol{x}_{D-1}^{K^{d-1}}, \boldsymbol{x}_D^{K^d}\}$

(Mixed strategy!)

Intuition: In the GGM framework, if $\mu^t = \text{Unif}\{\boldsymbol{x}_1, \dots, \boldsymbol{x}_D\}$ let ϕ^t be the "map" that takes $\boldsymbol{x}_1 \mapsto \boldsymbol{x}_2 \mapsto \cdots \mapsto \boldsymbol{x}_D$

- μ^t is an expected fixed point of ϕ^t
- each value of ϕ^t is being picked by regret minimizer $\Rightarrow$ Φ-regret is small!

Summary + some further references

What equilibrium concepts can be reached by ***efficient learning algorithms****?*

Previously believed to be the limit of GGM

Correlated equilibrium concept		Normal-form coarse-correlated	Extensive-form correlated	Linear-swap correlated	Low-degree swap correlated	Normal-form correlated
Set of deviations Φ		Constant functions	"Trigger" functions	Linear functions	Degree-k polynomials	All functions
Best-known algorithm	**# iterations for ϵT regret**	n/ϵ^2	nbd/ϵ^2	n^4/ϵ^2	$n^{\mathcal{O}(kd \log b)^3}/\epsilon^2$	$n^{\tilde{\mathcal{O}}(1/\epsilon)}$
	Per-iteration complexity	n	$\mathrm{FP}(n)$	$\mathrm{FP}(n)$	$n^{\mathcal{O}(kd \log b)^3}/\epsilon$	n/ϵ
	Citation	Farina, Lee, Luo, Kroer *ICML*'22	Farina, Celli, Marchesi, Gatti *JACM*'22	Zhang, Farina, Sandholm *ICLR*'24	Zhang, Anagnostides Farina, Sandholm *arXiv*'24	Peng & Rubinstein *STOC*'24; Dagan, Daskalakis, Fishelson, Golowich *STOC*'24

Notation:
b = branching factor of game
d = depth of game
$\mathrm{FP}(n)$ = time complexity of computing a fixed point of an $n \times n$ matrix
$\mathrm{QP}(n)$ = time complexity of solving an n-variable convex quadratic program

Tighter equilibrium concepts
Larger sets Φ
Harder to learn

Summary + some further references

What equilibrium concepts can be reached by ***efficient learning algorithms****?*

Previously believed to be the limit of GGM

Correlated equilibrium concept	Normal-form coarse-correlated	Extensive-form correlated	Linear-swap correlated	Low-degree swap correlated	Normal-form correlated
Set of deviations Φ	Constant functions	"Trigger" functions	Linear functions	Degree-k polynomials	All functions
Best-known algorithm: # iterations for ϵT regret	n/ϵ^2	[illegible]	n^4/ϵ^2	$n^{\mathcal{O}(kd \log b)^3}/\epsilon^2$	$n^{\tilde{\mathcal{O}}(1/\epsilon)}$
Best-known algorithm: Per-iteration complexity	[illegible]	[illegible]	$\mathrm{FP}(n)$	$n^{\mathcal{O}(kd \log b)^3}/\epsilon$	n/ϵ
Best-known algorithm: Citation	Farina, Lee, Luo, Kroer *ICML*'22	Farina, Celli, Marc…	Zhang, F…	Zhang, …	Peng & Rubinstein *STOC*'24; …askalakis, … 4

One UTC mediator

$O(kd \log b)^3$ UTC mediators + “expected fixed points” to circumvent PPAD-hard computation

…quilibrium concepts
Larger sets Φ
Harder to learn

Notation:
b = branching factor of game
d = depth of game
$\mathrm{FP}(n)$ = time complexity of computing a fixed point of an $n \times n$ matrix
$\mathrm{QP}(n)$ = time complexity of solving an n-variable convex quadratic program

References

- A Blum, Y Mansour (*JMLR* 2007), "From external to internal regret"
- GJ Gordon, A Greenwald, C Marks (*ICML* 2008), "No-regret learning in convex games"
- BH Zhang, G Farina, T Sandholm (*ICML* 2023), "Team belief DAG: generalizing the sequence form to team games for fast computation of correlated team max-min equilibria via regret minimization"
- BH Zhang, G Farina, T Sandholm (*ICLR* 2024), "Mediator Interpretation and Faster Learning Algorithms for Linear Correlated Equilibria in General Extensive-Form Games"
- BH Zhang, I Anagnostides, G Farina, T Sandholm (*arXiv* 2024), "Efficient Φ-Regret Minimization with Low-Degree Swap Deviations in Extensive-Form Games"
- C Daskalakis, G Farina, N Golowich, T Sandholm, BH Zhang (*arXiv* 2024), "A Lower Bound on Swap Regret in Extensive-Form Games"