Lecture 5: Online Learning + Extensive Form Games

10 Sept 2025

CS6208 Fall 2025: Computational Game Theory

Admin Matters

Homework 1 has been released

- Have about 2.5 weeks left to do it

Project briefing today [middle of lecture]

- Due one week after HW1 due

Updated grading:

- Added Quizzes [new*]
- Homework 1 (20%) + Quiz 1 (10%) = 30%
- Homework 2 (20%) + Quiz 2 (10%) = 30%
- Project = 40%
- Think about homework as having an individual / group component

Quiz 1 is due the same time as HW1

Quizzes

2 Quizzes, 10% each

Submit on Canvas→Quizzes
- Do not submit paper copies
- Done **individually**, can discuss if you want (can’t stop you from collaborating)
- Can retake as many times as you want before deadline (solutions not released)

Format
- All MCQs, True/False, "Check all which apply”
- Mix of definitions and conceptual questions, no heavy proofs required
- Not all questions are equal in difficulty

Should take no more than 2-3 hours per quiz

Recall our setting for zero-sum games

Nash as a minimax problem

The Von Neumann minimax theorem

Solving zero-sum games using linear programming

Unresolved issues

- Poly time, but how fast exactly?
- I cheated by outsourcing the problem to LP solver
- Seems to be overkill? LP and games are closely linked, but LP *solving* seems too general an algorithm? Is there something more intuitive?

$$\min_{x\in\mathbb{R}^n}\max_{y\in\mathbb{R}^m} x^T A y$$

$$\text{such that } 1^T x = 1^T y = 1$$

$$x, y \geq 0$$

Duals

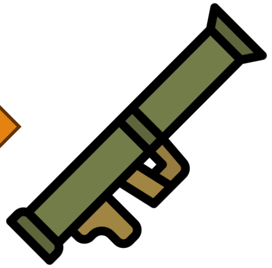
$$\min_{x\in\mathbb{R}^n, V\in\mathbb{R}} V$$

$$\text{such that } 1^T x = 1$$

$$A^T x \leq V$$

$$x \geq 0$$

Throw to LP solver



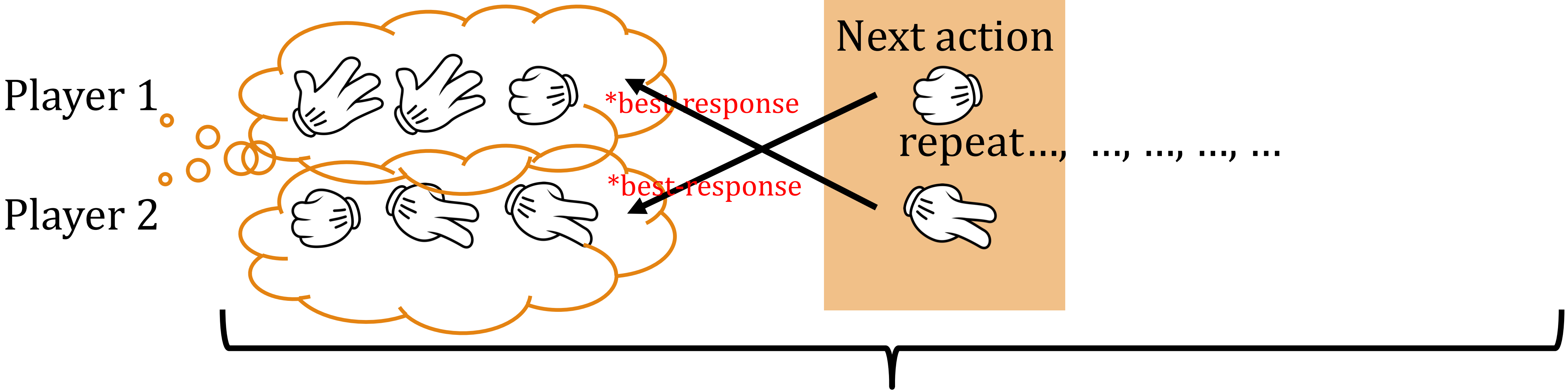
Review: Fictitious play

Recall: $A = -B$: Your pain is my pleasure

- Nice properties (e.g., exchangeability, unique game value, polytime solvers)

Fictitious Play (Brown, 1951): *Self play* can lead to Nash

- Players best-respond to the *empirical distribution* of past opponent actions



Empirical distribution of self-play converges to Nash

Review: Approximate Equilibrium

(x_0, y_0) is a an ϵ-approximate Nash (ϵ- Nash) if expected utility from unilaterally deviating does not increase by more than ϵ

$$\min_{x \in \Delta_n} x^T A y_0 \geq x_0^T A y_0 - \epsilon \quad \textbf{AND} \quad \max_{y \in \Delta_m} x_0^T A y \leq x_0^T A y_0 + \epsilon$$

extends to general-sum NE also

- Sanity Check: A NE itself is a 0-approximate NE
- There is another definition (well supported approximated equilibrium) based on what actions are allowed in the support

Saddle point residual = ϵ → ϵ-approximate NE

$$\min_{x \in \Delta_n} x_0^T A y_0 - x^T A y_0 = \epsilon_1 \qquad \max_{y \in \Delta_m} x_0^T A y - x_0^T A y_0 = \epsilon_2$$

$$\max_{y \in \Delta_m} x_0^T A y - \min_{x \in \Delta_n} x^T A y_0 = \epsilon_1 + \epsilon_2 \geq \max(\epsilon_1, \epsilon_2)$$

Let's minimize the saddle-point residual instead!

Review: no-regret learning

Comparing to best sequence in hindsight is too harsh

More reasonable: compare against best **fixed strategy in hindsight**

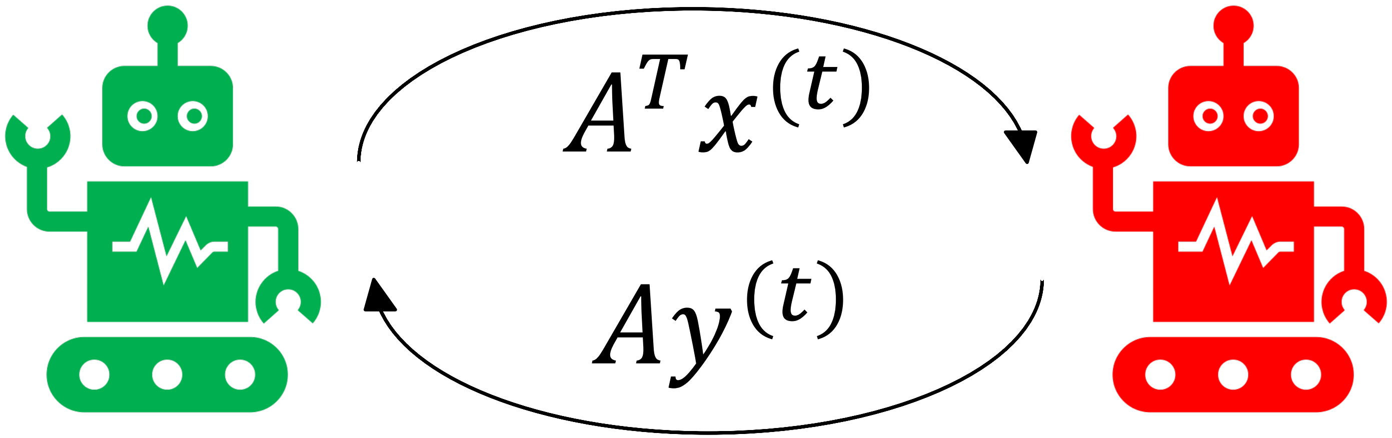
$$\cancel{\sum_{t=1}^{T} \langle x^{(t)}, g^{(t)} \rangle - \sum_{t=1}^{T} \min_x \langle x, g^{(t)} \rangle}$$

$$\underbrace{\sum_{t=1}^{T} \langle x^{(t)}, g^{(t)} \rangle - \min_x \sum_{t=1}^{T} \langle x, g^{(t)} \rangle}$$

Want regret to grow
sublinearly in T

Review: Self-Play for game solving

$$\left.\begin{array}{l}\textsc{NextStrategy}() \\ \textsc{ObserveUtility}(\boldsymbol{\ell}^{(t)})\end{array}\right\}\text{Loop}$$



Average strategies converge to Nash (saddle point residual drops to 0)

Review: Rate of convergence

Sum of regrets is sublinear in t

$$\text{Recall: Saddle point residual} \leq \frac{R_1+R_2}{t}$$

Instantiate self-play using **any pair** of regret minimizers!

For Hedge, saddle-point residual drops at rate

$$\mathcal{O}(\log(\max(n,m)) \cdot \frac{1}{\sqrt{T}})$$

$$\text{Recall: Saddle point residual} \leq \epsilon \implies \epsilon\text{--NE}$$

Example: regret minimization

	Expert 1	Expert 2	Expert 3
Player 😎	$x_1^{(1)}$=0.25	$x_2^{(1)}$=0.5	$x_3^{(1)}$=0.25
Adversary 😈	$g_1^{(1)}$=1	$g_2^{(1)}$=0.5	$g_3^{(1)}$=0.8

Loss incurred at time 1 = $\left\langle x^{(1)}, g^{(1)} \right\rangle = 0.7$

Best **expert-so-far** in hindsight= 0.5

Total regret = 0.7-0.5=0.2

Player 😎	$x_1^{(2)}$=0.1	$x_2^{(2)}$=0.7	$x_3^{(2)}$=0.2
Adversary 😈	$g_1^{(2)}$=0.1	$g_2^{(2)}$= 0.8	$g_3^{(2)}$=0.2

Loss incurred at time 2 = $\left\langle x^{(2)}, g^{(2)} \right\rangle = 0.61$

Best **expert-so-far** in hindsight= 1.0

Total regret = 0.61+0.7-1.0=0.31

+time

Example: regret minimization

	Expert 1	Expert 2	Expert 3
Player 😎	$x_1^{(1)}$=0.5	$x_2^{(1)}$=0.5	$x_3^{(1)}$=0.0
Adversary 😈	$g_1^{(1)}$=1	$g_2^{(1)}$=0.1	$g_3^{(1)}$=0.0

Loss incurred at time 1 = $\left\langle x^{(1)}, g^{(1)} \right\rangle =?$

Best expert-so-far in hindsight= ?

Total regret = ?

Player 😎	$x_1^{(2)}$=0.1	$x_2^{(2)}$=0.3	$x_3^{(2)}$=0.6
Adversary 😈	$g_1^{(2)}$=0.1	$g_2^{(2)}$= 0.2	$g_3^{(2)}$=1.0

Loss incurred at time 2 = $\left\langle x^{(2)}, g^{(2)} \right\rangle =?$

Best expert-so-far in hindsight= ?

Total regret = ?

+time

Review: Hedge ($\eta = 1$)

	Expert 1	Expert 2	Expert 3
Weights ⚖️	1	1	1
Player 😎	$x_1^{(1)}$=1/3	$x_2^{(1)}$=1/3	$x_3^{(1)}$=1/3
Adversary 😈	$g_1^{(1)}$=1	$g_2^{(1)}$=0.5	$g_3^{(1)}$=0.8
Weights ⚖️	$\exp(-\eta \cdot 1) \cdot 1$	$\exp(-\eta \cdot 0.5) \cdot 1$	$\exp(-\eta \cdot 0.8) \cdot 1$
Player 😎	$x_1^{(2)} \propto \exp(-\eta)$	$x_2^{(2)} \propto \exp(-\eta 0.5)$	$x_3^{(2)} \propto \exp(-\eta 0.8)$
Adversary 😈	$g_1^{(2)}$=0.1	$g_2^{(2)}$= 0.8	$g_3^{(2)}$=0.2
Weights ⚖️	$\exp(-\eta \cdot 1.1) \cdot 1$	$\exp(-\eta \cdot 1.3) \cdot 1$	$\exp(-\eta \cdot 1.0) \cdot 1$

+time

Review Question 1

Which of the following are True?

- Running self-play on multiplayer general-sum games leads to iterates that converge on average to a NE
- Running self-play on 2-player general-sum games leads to iterates that converge on average to a NE
- Running self-play on multiplayer zero-sum games leads to iterates that converge on average to a NE
- Running self-play on 2-player zero-sum games leads to iterates that converge on average to a NE

Review Question 2

Which of the following are True in **2-player zero-sum games** with a game value of v? Assume player 1 **minimizes**, player 2 maximizes, and the utility matrix is A.

- If x^* is a NE for player 1, then there must exist some y such that $x^{*T}Ay = v$
- If x^* is a NE for player 1, there cannot exist any y such that $x^{*T}Ay < v$
- For every fixed x, there cannot exist some y such that $x^T Ay < v$
- For every fixed x, there must exist some y such that $x^T Ay \geq v$

Review Question 3

Recall that we want total regret to be sublinear in time.

Can regret (by our definition) be ever be negative?

Review Question 4

If we allowed the player to cheat by observing the adversary's choice of $\ell^{(t)}$ before choosing $x^{(t)}$, does the resultant sequence of x's achieve sublinear total regret?

Player 😎	$x_1^{(1)}$=0.5	$x_2^{(1)}$=0.5	$x_3^{(1)}$=0.0
Adversary 😈	$g_1^{(1)}$=1	$g_2^{(1)}$=0.1	$g_3^{(1)}$=0.0

Adversary 😈	$g_1^{(1)}$=1	$g_2^{(1)}$=0.1	$g_3^{(1)}$=0.0
Player 😎	$x_1^{(1)}$=0.0	$x_2^{(1)}$=0.0	$x_3^{(1)}$=1.0

Self-play in general-sum games?

What if we were to run self-play in general-sum games?

- Nothing is stopping us from doing it
- Both players will still have sublinear regret → not incentivized to deviate
- Isn't this sound like Nash? Does that mean that we get NE from self-play?

Ans: not quite

- Player strategies may end up *correlated* (this is true even in 0-sum games)

Average of *joint strategies* converges to a coarse-correlated equilibrium (CCE)

- CCE is a superset of Nash
- For zero-sum games they coincide (up to payoff equivalence)!

External, Internal, Swap regret, Phi regret →CCE, CE … [different eqm]

- We are dealing with *external regret* now. More in later lectures

Regret Matching (Plus)

Another regret minimizer on the simplex

Regret Matching

Can be derived using Blackwell Approachability

Maintain at timestep t, $\mathrm{r}_i^{(t)}$, the regret associated to action i

- "How much regret I have from doing what I did instead of action i"

$$r_i^{(t)} = \sum_{\tau=1}^{t} \langle x^{(\tau)}, g^{(\tau)} \rangle - \sum_{\tau=1}^{t} g_i^{(\tau)}$$

$$r^{(t)} = \sum_{\tau=1}^{t} \langle x^{(\tau)}, g^{(\tau)} \rangle \mathbf{1} - \sum_{\tau=1}^{t} g^{(\tau)}$$ *in vector form

At time $t+1$, play

$$x_i^{(t+1)} = \frac{\max(r_i^{(t)}, 0)}{\sum_j \max(r_j^{(t)}, 0)}$$

Threshold at 0, then play proportionately

If all $r^{(t)} \leq 0$, play uniformly

RM

	Expert 1	Expert 2	Expert 3
Regrets ⚖️	0	0	0
Player 😎	$x_1^{(1)}$=1/3	$x_2^{(1)}$=1/3	$x_3^{(1)}$=1/3
Adversary 😈	$g_1^{(1)}$=1	$g_2^{(1)}$=0.5	$g_3^{(1)}$=0.8

Loss incurred at time 1 $=$ 2.3/3

Regrets ⚖️	$r_1^{(1)}$=2.3/3-1=-0.233	$r_2^{(1)}$=2.3/3-0.5=0.267	$r_3^{(1)}$=2.3/3-0.8=-0.033
Player 😎	$x_1^{(2)} = 0$	$x_2^{(2)} \propto 0.267 = 1$	$x_3^{(2)} = 0$
Adversary 😈	$g_1^{(2)}$=0.1	$g_2^{(2)}$= 0.8	$g_3^{(2)}$=0.2

Loss incurred at time 2 $=$ 0.8, total loss = 1.67

Regrets ⚖️	$r_1^{(2)}$=1.67-1.1=0.57	$r_2^{(2)}$=1.67-1.3=0.37	$r_3^{(2)}$=1.67-1.0=0.67
Player 😎	$x_1^{(3)} \propto 0.57$	$x_2^{(3)} \propto 0.37$	$x_3^{(3)} \propto 0.67$

+time

Why another regret minimizer?

Isn't Hedge already optimal?

Hedge (technically) depends on a learning rate

- Depends on horizon, can be set carefully, but quite annoying
- Another way is to decay the learning rate
- RM is free of learning rate

Theory vs. practice

- RM works well in practice
- RM is easy to code, only requires a memory of size n
  - So is Hedge technically…

RM+

Same as RM, but when we threshold regrets at 0, we do it **permanently**

Regrets ⚖️	0	0	0
Player 😎	$x_1^{(1)}$=1/3	$x_2^{(1)}$=1/3	$x_3^{(1)}$=1/3
Adversary 😈	$g_1^{(1)}$=1	$g_2^{(1)}$=0.5	$g_3^{(1)}$=0.8

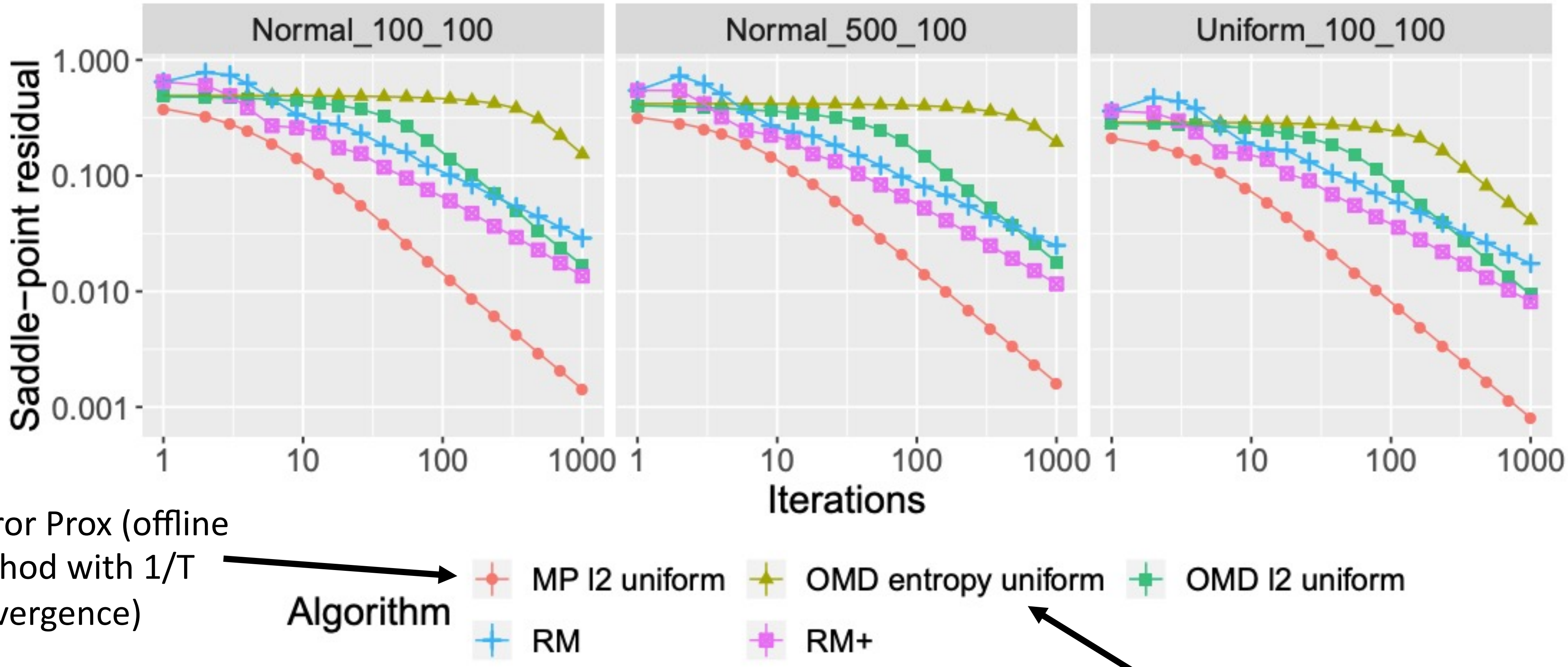
Loss incurred at time 1 $= 2.3/3$

Regrets ⚖️	$r_1^{(1)}$=2.3/3-1=-0.233→ 0	$r_2^{(1)}$=2.3/3-0.5=0.267	$r_3^{(1)}$=2.3/3-0.8=-0.033→0
Player 😎	$x_1^{(2)} = 0$	$x_2^{(2)} \propto 0.267 = 1$	$x_3^{(2)} = 0$
Adversary 😈	$g_1^{(2)}$=0.1	$g_2^{(2)}$= 0.8	$g_3^{(2)}$=0.2

Loss incurred at time 2 $= 0.8$

Regrets ⚖️	$r_1^{(2)}$=0+0.8-0.1=0.7	$r_2^{(2)}$=0.267+0.8-0.8=0.267	$r_3^{(2)}$=0+0.8-0.2=0.6
Player 😎	$x_1^{(3)} \propto 0.7$	$x_2^{(3)} \propto 0.267$	$x_3^{(3)} \propto 0.6$

Example convergence rates



Mirror Prox (offline method with 1/T convergence)

MW/Hedge

Source: http://www.columbia.edu/~ck2945/files/main_ai_games_markets.pdf

Blackwell Approachability (optional)

We are going to construct RM, a more **practical** regret minimizer

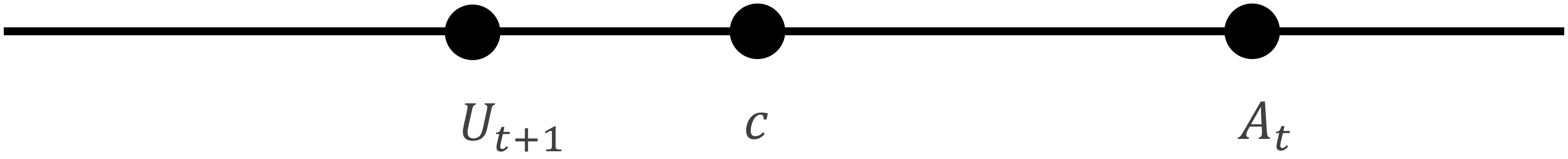
Approachability in Scalars

Sequence of **bounded** scalars $\{U_t\}, U_t \in \mathbb{R}$

Let average be $A_T = \frac{1}{T}\sum_{t=1}^{T} U_t$

Let $c \in \mathbb{R}$ be a target.

Assume $\{U_t\}$ is constrained such that $(U_{T+1} - c)(A_T - c) \leq 0$



Then $\lim\limits_{T\to\infty} A_T = c$

Intuition: being on the "opposite" side gives enough "power" to reach c, boundedness of U ensures no oscillations.

Approachability in Vectors

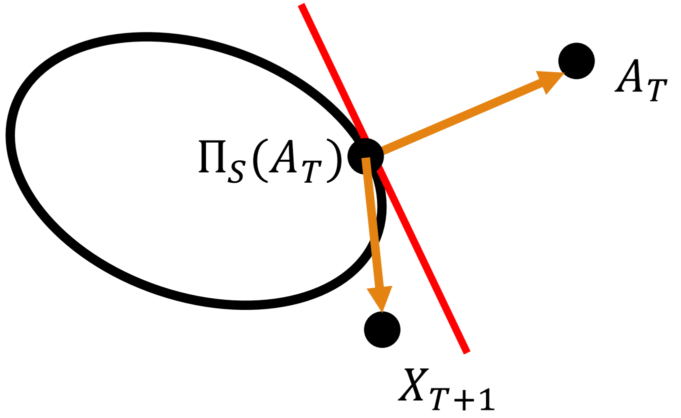
Sequence of **bounded** vectors $\{U_t\}, U_t \in \mathbb{R}^K$

Let average be $A_T = \frac{1}{T}\sum_{t=1}^{T} U_t$

Let $S \in \mathbb{R}$ be a **convex target set**.

- Let $\Pi_S(A_t)$ be the closest point (projection) of A_t onto S

Assume $\{U_t\}$ is such that $(U_{T+1} - \Pi_S(A_T)) \cdot (A_T - \Pi_S(A_T)) \leq 0$



Then $d(A_T, S) \to 0$

Intuition: Always walking “towards” the tangent hyperplane with enough “power”

Approachability in Vectors in Expectation

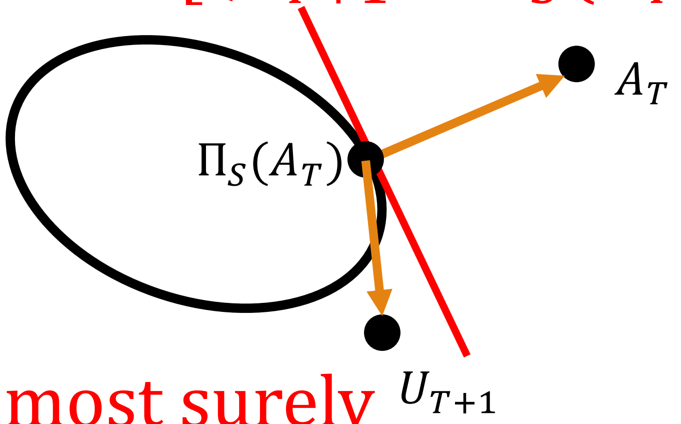
Sequence of **bounded** random vectors $\{U_t\}$, $U_t \in \mathbb{R}^K$

Let average be $A_T = \frac{1}{T}\sum_{t=1}^{T} U_t$

Let $S \in \mathbb{R}$ be a **convex target set.**

- Let $\Pi_S(A_t)$ be the closest point (projection) of A_t onto S

Assume $\{U_t\}$ is such that $\mathrm{E}[(U_{T+1} - \Pi_S(A_T)) \cdot \big(A_T - \Pi_S(A_T)\big)] \leq 0$



Then $d(A_T, S) \to 0$ almost surely

U_t's do not have to be iid. In fact, the expectation doesn't even have to be conditioned on the past!

Blackwell Approachability Game

First, P1 selects action $x_t \in \mathcal{X}$

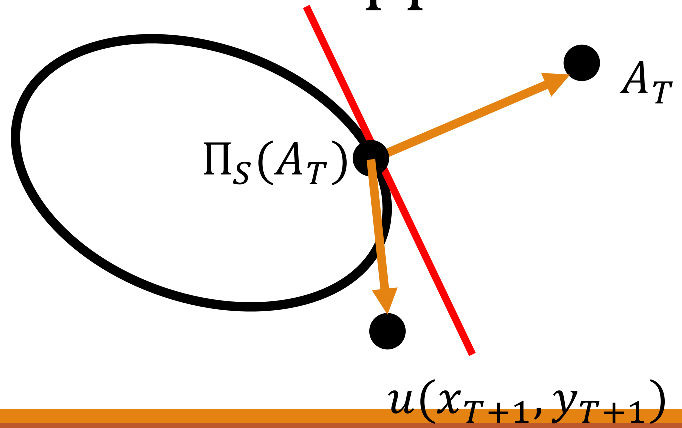
Then, P2 selects action $y_t \in \mathcal{Y}$, adversarial w.r.t. all x_t thus far

P1 incurs a **vector-valued** payoff $u(x_t, y_t)$. Typically, u is biaffine.

P1's goal is to force the average u's to converge to target set S

$$\min_{\hat{s} \in S} \left\| \hat{s} - \frac{1}{T} \sum_{t=1}^{T} u(x_t, y_t) \right\| \to 0 \text{ as } T \to \infty$$

Idea: Let's use Blackwell approachability



*Want to be able to choose x_T such that no matter how y_{T+1} is chosen, $u(x_{T+1}, y_{T+1})$ will always be on left side of hyperplane!

Forcing Halfspaces and Actions

Convex sets can be difficult to deal with: lets work with halfspaces

Let's consider halfspaces tangent to S: call it $\mathcal{H}$

$$\mathcal{H} = \{x \in \mathbb{R}^K | a^T x \leq b\}$$

$\mathcal{H}$ is forceable if there exists a strategy in x^* such that $u(x^*, y) \in \mathcal{H}$ for all possible choices of y

- x^* is called a **forcing action**

Blackwell: P1's goal will if every halfspace $H \supseteq S$ is forceable

Constructive Proof:

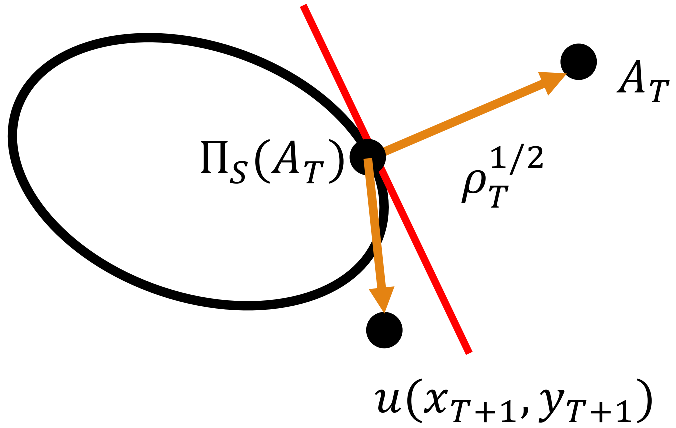
- At T , if $A_T \in S$, choose any $x^* \in \mathcal{X}$
- If not, let $\mathcal{H}$ be halfspace tangent to S containing $\Pi_S(A_T)$, choose x^* to be forcing action of $\mathcal{H}$.

Some derivations (optional)

We could just use Blackwell's theorem, but since this is deterministic it is easy to explicitly show that $d(A_T, S)$ decreases at rate of $1/\sqrt{T}$

$$A_{T+1} = \frac{1}{T+1}\sum_{t=1}^{T+1} u(x_t, y_t) = \frac{T}{T+1}A_T + \frac{1}{T+1}u(x_{T+1}, y_{T+1})$$

$$\rho_T = ||\Pi_S(A_T) - A_T||^2 = \min_{\hat{s}\in\mathcal{S}} ||\hat{s} - A_T||^2$$



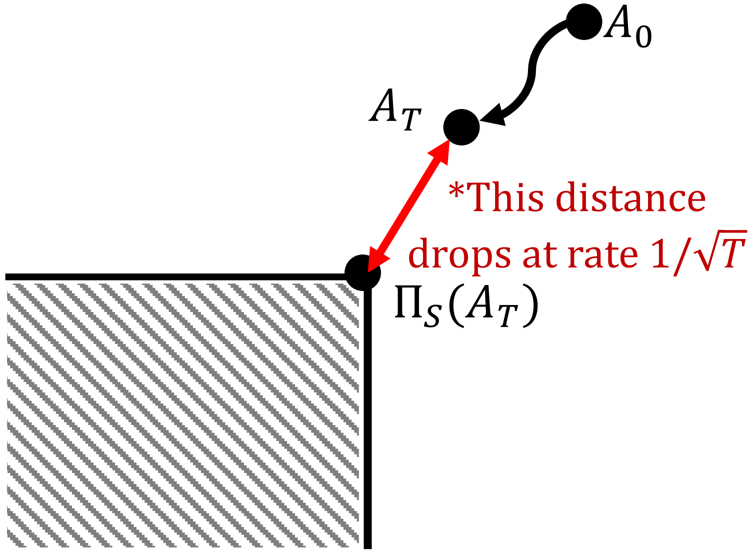
$$\begin{aligned}
\rho_{T+1} &= ||\Pi_S(A_{T+1}) - A_{T+1}||^2 \\
&\leq ||\Pi_S(A_T) - A_{T+1}||^2 && \text{Projection must be shortest distance} \\
&= ||\Pi_S(A_T) - \frac{T}{T+1}A_T - \frac{1}{T+1}u(x_{T+1}, y_{T+1})||^2 && \text{Rewrite} \\
&= ||\frac{T}{T+1}(\Pi_S(A_T) - A_T) + \frac{1}{T+1}(\Pi_S(A_T) - u(x_{T+1}, y_{T+1})||^2 && \text{Expand} \\
&= \left(\frac{T}{T+1}\right)^2 \rho_T + \left(\frac{1}{T+1}\right)^2 \underbrace{||\Pi_S(A_T) - u(x_{T+1}, y_{T+1})||^2}_{\text{Bounded by Diameter } \Omega^2} + \frac{2T}{(T+1)^2}\underbrace{\langle \Pi_S(A_T) - A_T, \Pi_S(A_T) - u(x_{T+1}, y_{T+1})\rangle}_{\leq 0 \text{ because forcing action}}
\end{aligned}$$

$$(T+1)^2\rho_{T+1} - T^2\rho_T \leq \Omega^2 \implies \rho_{T+1} \leq \frac{\Omega^2}{T+1} \implies \min_{\hat{s}\in\mathcal{S}} ||\hat{s} - A_T||_2 \leq \frac{\Omega}{\sqrt{T}}$$

No-regret as a Blackwell Game

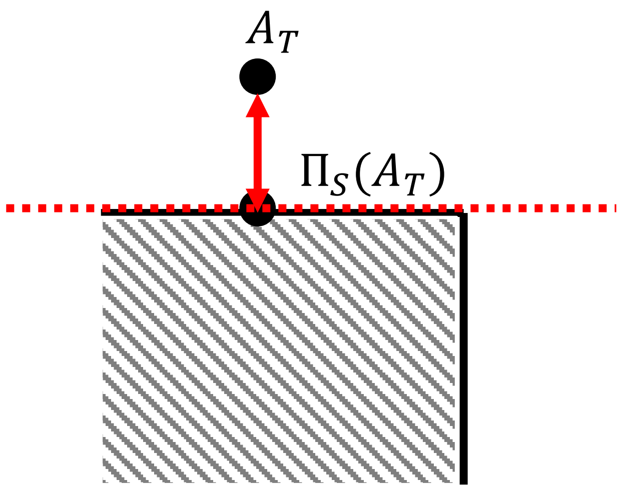
Instantiate

- $u(x_t, y_t) = \ell_t - \langle \ell_t, x_t \rangle \mathbf{1}$, i.e., regret incurred at t
- Hence, $A_T = \frac{1}{T}\sum_{t=1}^{T} u(x_t, y_t) = R_T/T$ gives average regret up till T
- $S = \{s \in \mathbb{R}^k | s \leq 0\}$, i.e., nonpositive quadrant
- Hence, if A_T tends to S then we are no-regret (roughly speaking)!

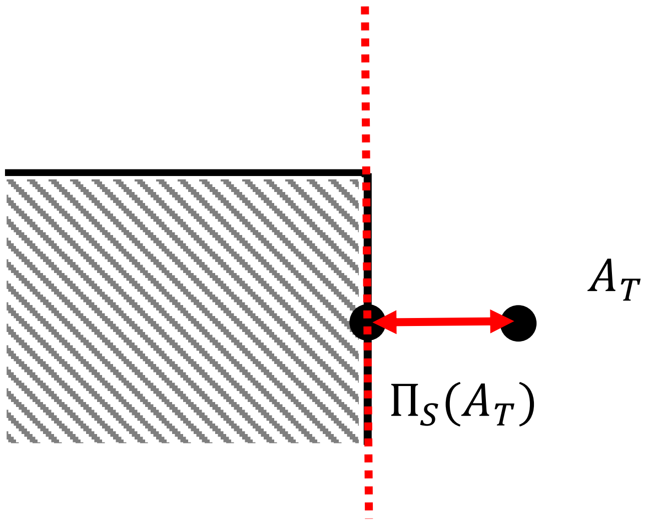


Theorem: The average regret is no greater than $d(A_T, S)$

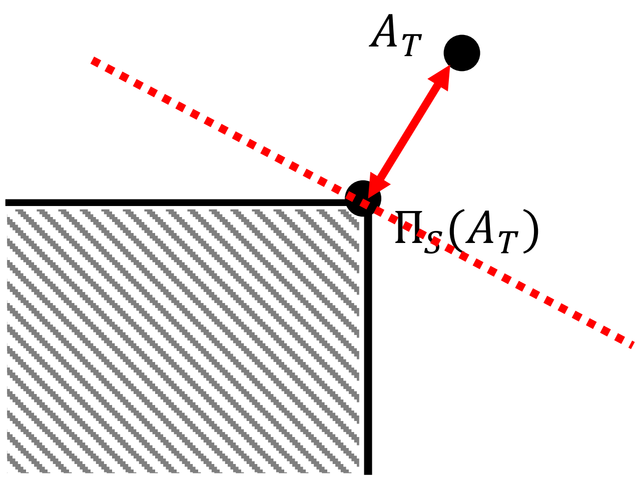
Regret Matching (RM)



Always play action corresponding to vertical-axis



Always play action corresponding to horizontal-axis



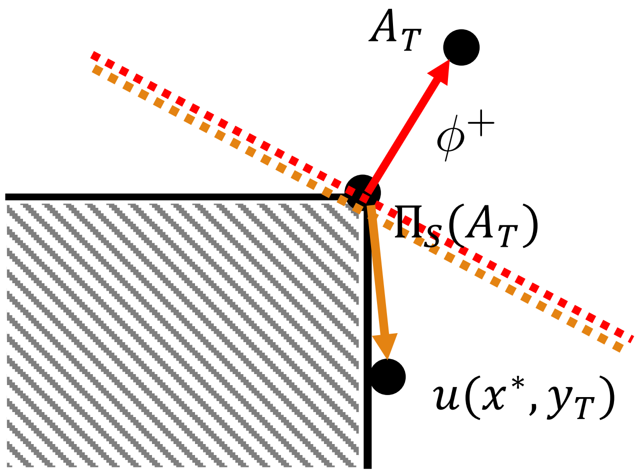
Play according to ratio of nonnegative average regrets (?)

Regret Matching Proof

Projection onto nonpositive orthant

$$A_T = [-2, 5, 2, -4] \implies \Pi_S(A_T) = [-2, 0, 0, -4], \underbrace{A_T - \Pi_S(A_T)}_{\triangleq\, \phi^+} = [0, 5, 2, 0]$$

*Assume ≠ 0



$$\mathcal{H} = \{x \in \mathbb{R}^K | \langle \phi^+, z \rangle \leq 0\}$$

$$u(x^*, y_T) \in \mathcal{H} \qquad \forall y_T$$

$$\iff \langle \phi^+, u(x^*, y_T)\rangle \leq 0 \qquad \forall y_T \quad \text{Definition}$$

$$\iff \langle \phi^+, \ell_T - \langle \ell_T, x^* \rangle 1 \rangle \leq 0 \qquad \forall \ell_T \in \mathbb{R}^K \quad \text{Definition}$$

$$\iff \langle \phi^+, \ell_T \rangle - \langle \ell_T, x^* \rangle ||\phi^+||_1 \rangle \leq 0 \qquad \forall \ell_T \in \mathbb{R}^K \quad \text{Rearrange}$$

$$\iff \langle \ell_T, \frac{\phi^+}{||\phi^+||_1} - x^* \rangle \leq 0 \qquad \forall \ell_T \in \mathbb{R}^K$$

Forcing action: Just choose $x^* = \dfrac{\phi^+}{||\phi^+||_1}$

RM and RM+

= reward vector Py_t

$\textsc{ObserveUtility}(\boldsymbol{\ell}_t)$

$$\underbrace{A_{T+1}}_{\text{New average regret}} = \max\left(\frac{T}{T+1}\underbrace{A_T}_{\text{Old average regret}} + \frac{1}{T+1}\underbrace{(\ell_T - \langle \ell_T, x_T \rangle \mathbf{1})}_{\text{Regret to accumulate for this round}}\right)^{+}$$

RM+: change average/cumulative regrets to 0 if negative

$\textsc{NextStrategy}()$

If $\phi^+ = 0$ just choose x^* uniformly at random

$$\underbrace{A_{T+1} = [-2, 5, 2, -4]}_{\text{Average regret}} \implies \underbrace{\phi^+ = [0, 5, 2, 0]}_{\text{Truncate negative regrets}} \implies \underbrace{x^* = [0, 5/7, 2/7, 0]}_{\text{Renormalize}}$$

Note: To make things simpler we could just work with cumulative regret all the way

Recall: convergence at rate $1/\sqrt{T}$

Project Briefing

Components

Project Topic Proposal ←deadline is around 3-4 weeks from now

Feedback from me (either canvas, email or meetings)

Final Proposal

Project Proposal

2-3 pages (appendix allowed)
- Due one week after HW1

Done in teams of 2 or 3, submit on Canvas
- One person per group submits.
- No need to create groups on Canvas, but make sure to indicate teammate **clearly**

50% background
- Existing or related work. **At least one paper**
- Framing of problem. E.g., cooperative or competitive? What type of equilibrium, if any? Sequential or not?

50% proposal
- What is novel or interesting? **At least one point**
- How are the methods we learned in class applicable/not applicable

Use any reasonable AI conference latex template (e.g., Neurips/ICLR/ICML)

Potential Projects

Applied

- Implementation + experimental
  - Make sure you are clear of what the novelty is?
- "Insights", e.g., a certain formulation can be thought of as a game
  - Be creative! Remember "players" need not be physical entities. E.g., 100 prisoner's game

Theoretical

- New problem formulations, domains for no-regret learning, rates of convergence
- New equilibrium concepts, equilibrium refinements
- Even though this is a proposal, it cannot *just* contain conjectures.
  - Need some reason to believe conjecture is true/false. E.g., simpler cases, experimental evidence

Avoid

- Surveys, projects containing only literature reviews
- "Findings" style projects

Extensive-Form Games

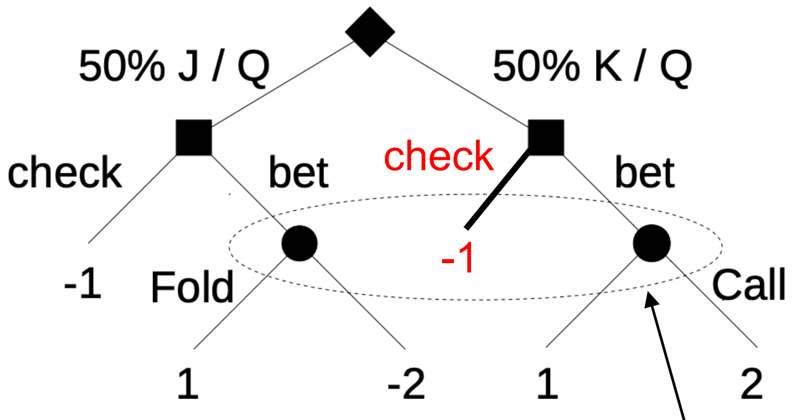
Play some one-card poker: https://www.cs.cmu.edu/~ggordon/poker/

Extensive-Form Games (EFGs)

Can be solved by minimax search

- Begin with a finite game tree
  - 2-players (can be generalized)
  - Chance (with known probabilities)
  - Leaves/Terminal states
    - Game ends there, players collect reward

*this class: zero-sum



From the thesis of Neil Burch

- Factor in information sets (infoset)
  - States within belong to same player, have the same actions
  - States with that **cannot be distinguished by player**
  - Perfect information → infosets are singletons
  - Workhorse behind imperfect information between players

Information set means player 2 cannot know if player 1 got a J or K!

- Actions are taken at infosets, not states
  - Payoffs can depend on states

Kuhn Poker (classic example)

The above game was somewhat easy and we could solve it manually. What about something more complicated?

Where are the information sets?

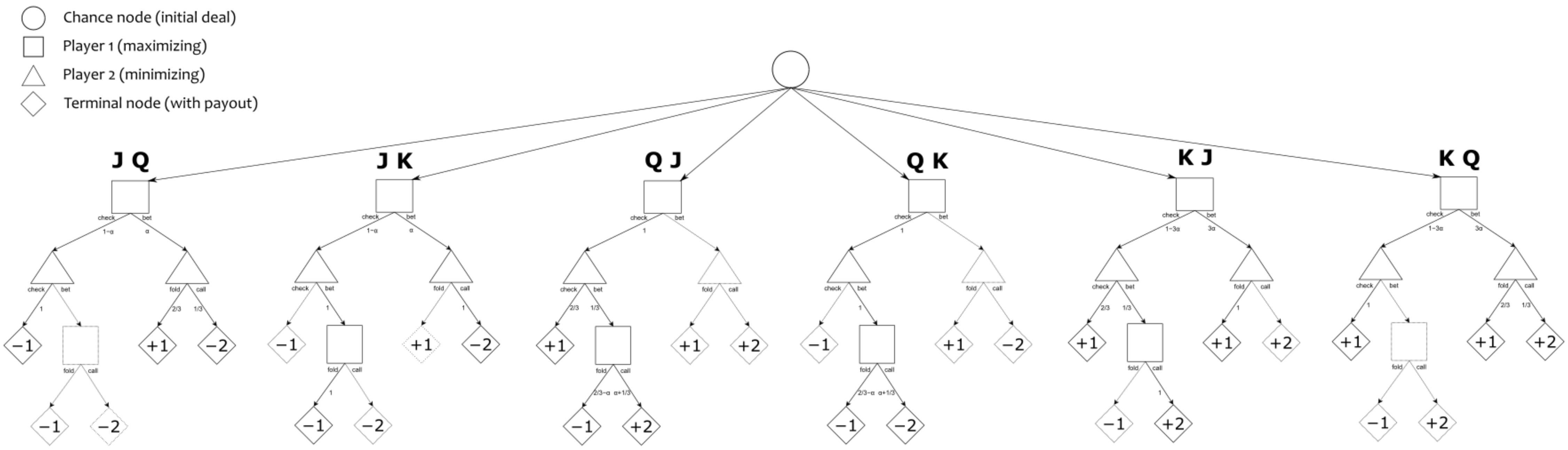
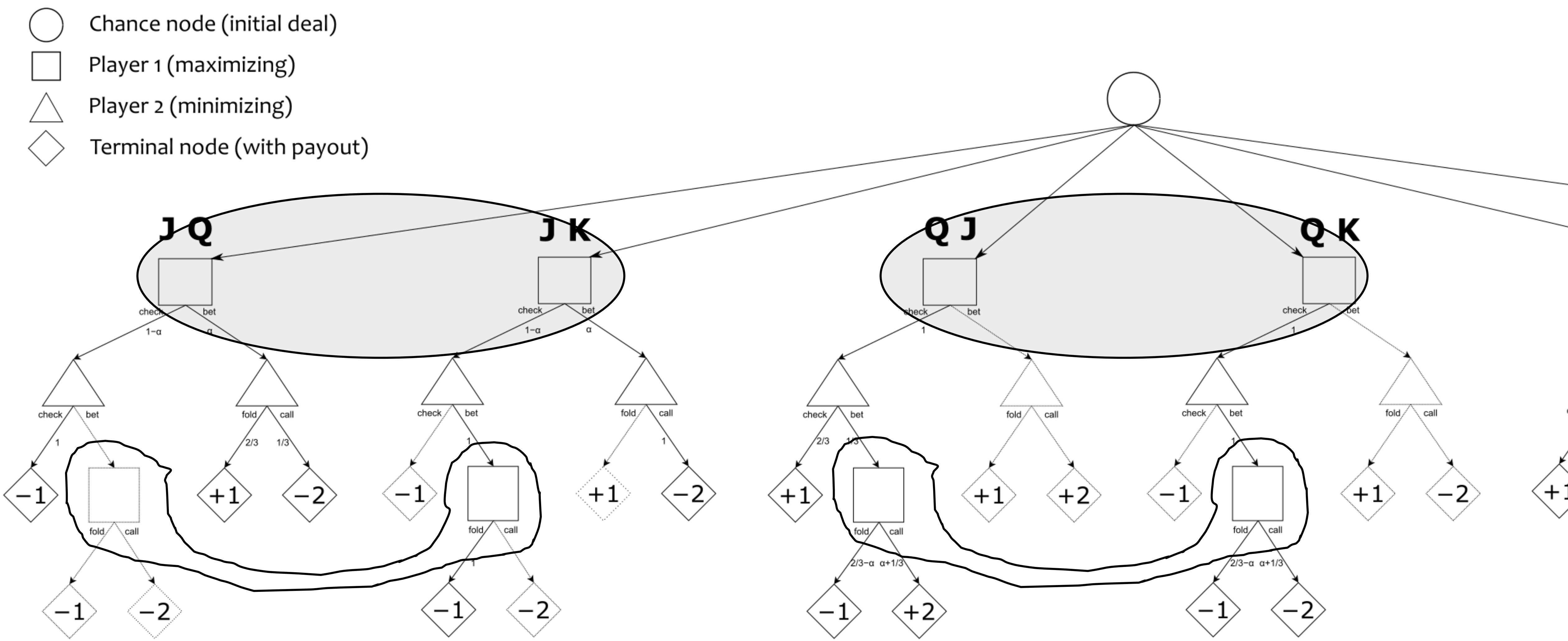
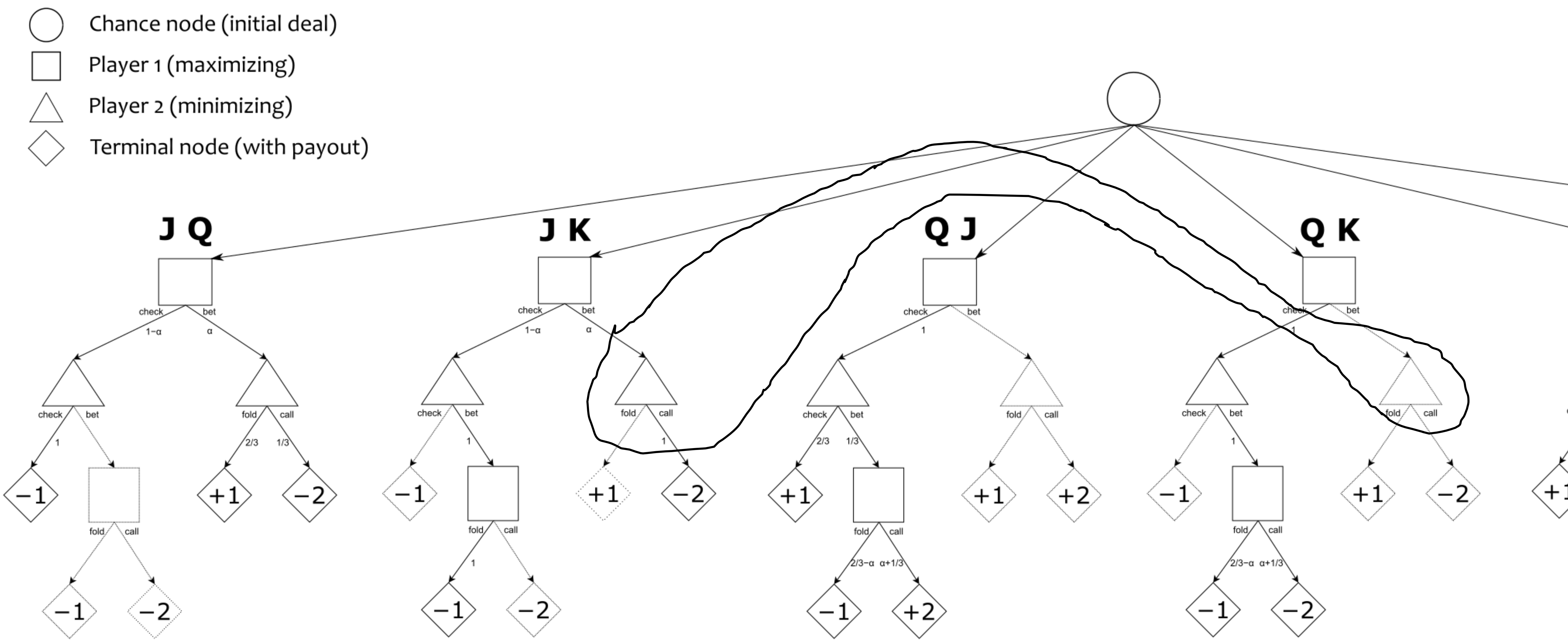


Image credits: https://commons.wikimedia.org/wiki/File:Kuhn_poker_tree.svg

Information sets (P1)

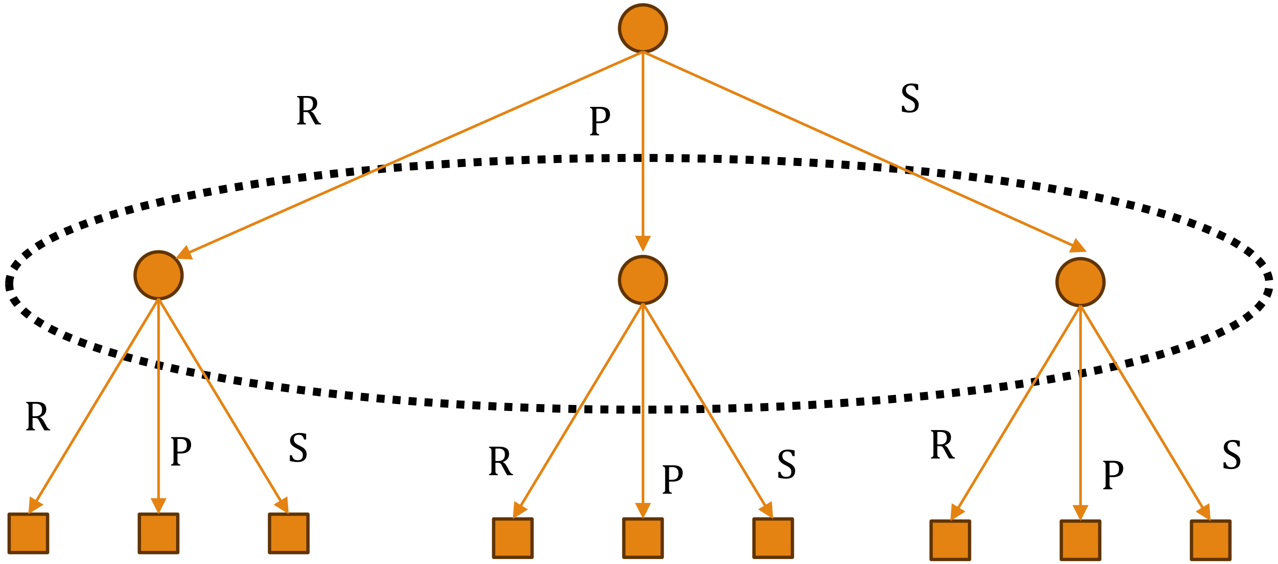


Information sets (P2)



Simulating simultaneous moves

Player 2 doesn't observe player 1's action when taking a move → essentially simultaneous



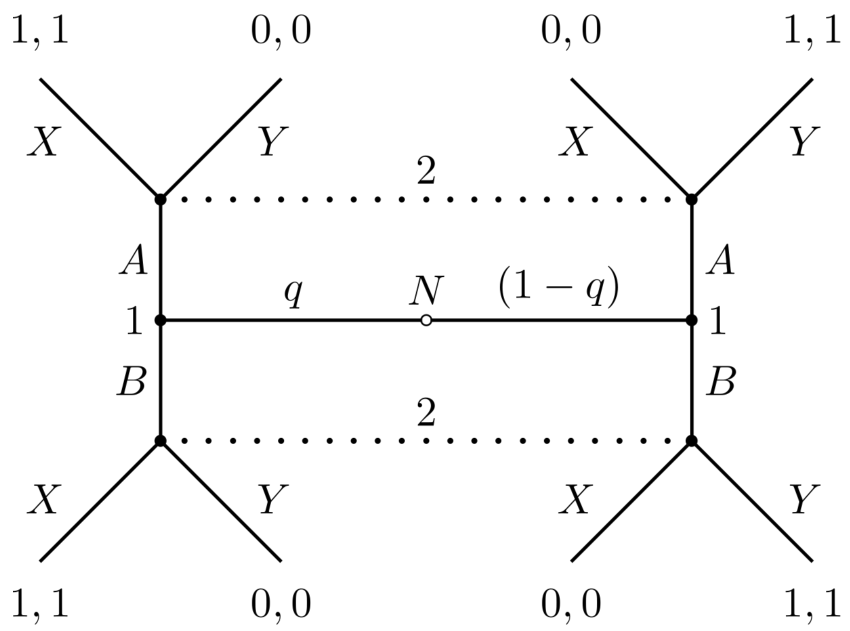
Lewis Signaling Games

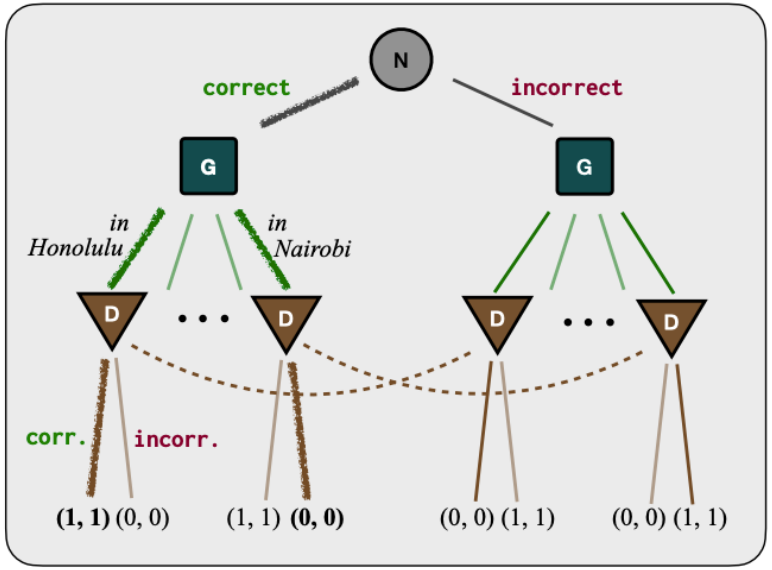
Used all over in economics, other areas of GT

- E.g., emergent communication, equilibrium selection
- We are **not** studying the finer properties of signaling games in this class

Related: Spence's Signaling Game, Cheap Talk

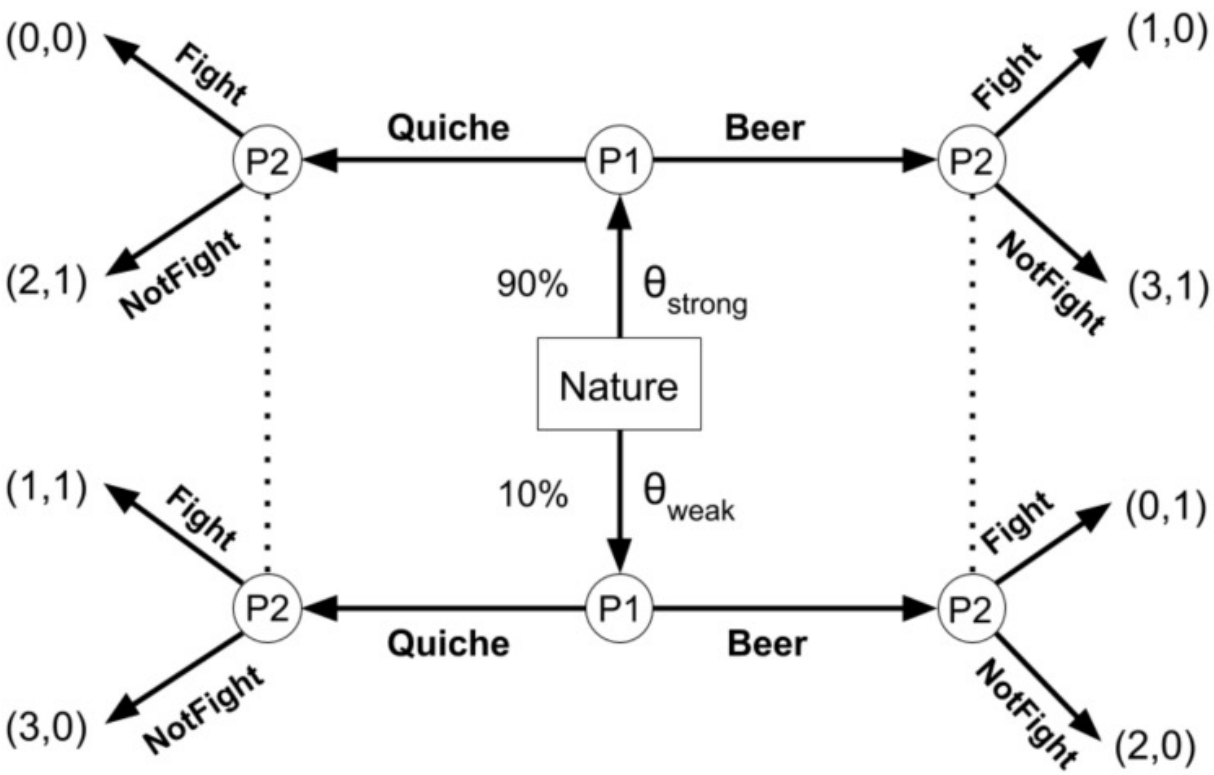
- These typically have continuous actions

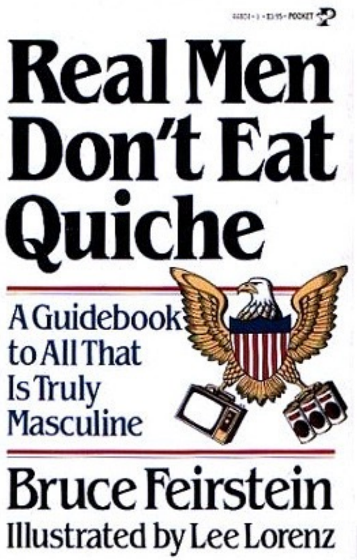




https://openreview.net/pdf?id=lPyHpdj5qO

Beer-Quiche Game





Real men don't need game theory…?

https://dspace.mit.edu/bitstream/handle/1721.1/128515/1702.01819.pdf

Some common mistakes

Infosets vs Markov Games

*Markov games are a topic not covered in this class

- Generally, entire history of actions matter, even if “state” seems the same. Cannot (directly) abstract away history
- Example: in poker, even if contribution to the pot is the same, past call/raise/bets would have revealed different information to the other player
- **Very much unlike MDPs**
- Big part of why EFGs can be trickier to solve

Size of an EFG

- We usually evaluate complexity using the **size of the game tree**
- E.g., we will say something like “space complexity is linear.”
- But, still **exponential in depth**, may not be tractable in practice

Information Structure in EFGs

Perfect Recall

The above games are very simple Bayesian games

- Real world problems can be much more complicated
- Partial information revealed at very specific periods of time
- Need a stronger assumption to make life easier

Players never forget observations and actions they made in the past

- Most algorithms will require perfect recall, including CFR, the main workhorse behind game solving

If state h, h' belong to same infoset of player i, then all paths from root to h, h' traverse the same sequence of infosets and actions of player i.

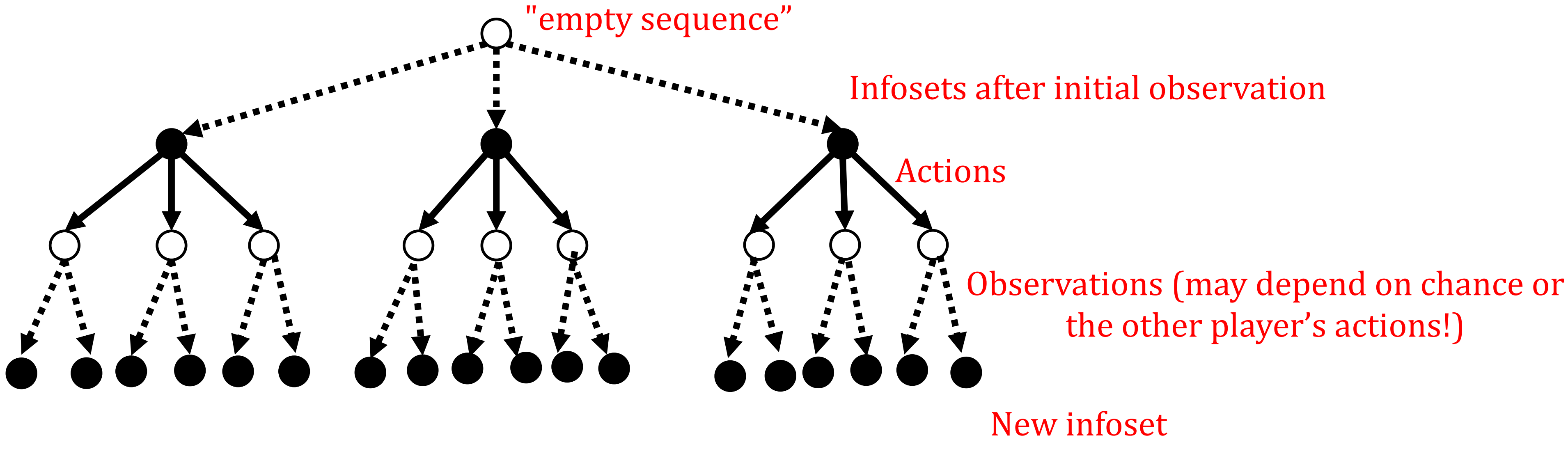
Often defined wrongly…

- E.g., if states h_1, h_2 belong to the different infosets of player i, then their children h_1', h_2' belonging to i must be in different infosets

Perfect recall games have nice structure

From a **single player's** perspective

- Every decision point (filled) has at most a single parent
- Obeys a tree-like structure
- If payoff at leaf actions and probabilities for dotted lines, given, then finding best response tractable via backward induction



Tree-based decision problem

Example of imperfect recall

Forgetting observations (e.g., Bridge, Hanabi)

Can also forget actions
(e.g., A-loss recall)

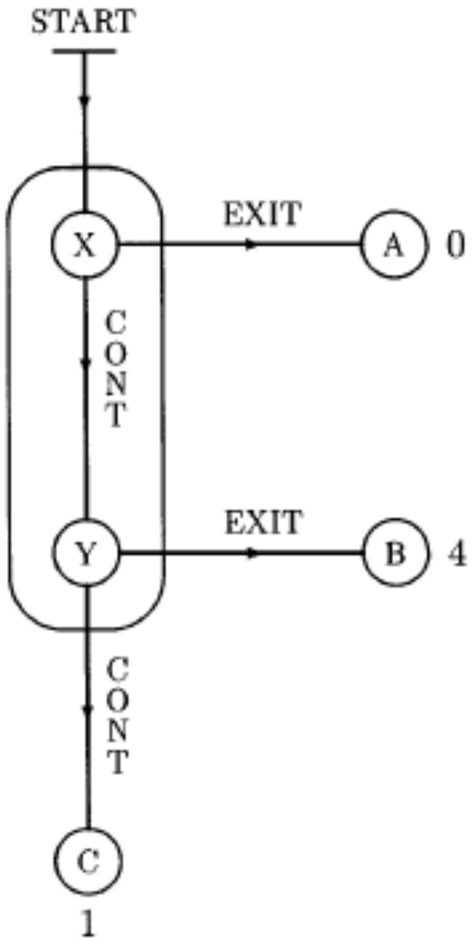


FIG. 1. The absent-minded driver problem.

Absent-mindedness

https://www.lesswrong.com/posts/GfHdNfqxe3cSCfpHL/the-absent-minded-driver

Timeability [optional]

Perfect recall makes many pathologies go away, but not all

- Can have cycles over infosets (*across players*)
- Precedes operation is not transitive

Timeability: infosets are assigned a time such that paths from root to leaf have increasing time

- Can draw tree vertically with time in the y-axis

Examples:

- Game pauses during card games (e.g., MTG) reveal that players hold certain cards

We don’t assume timeability for this class

- But it’s often a useful assumption to make

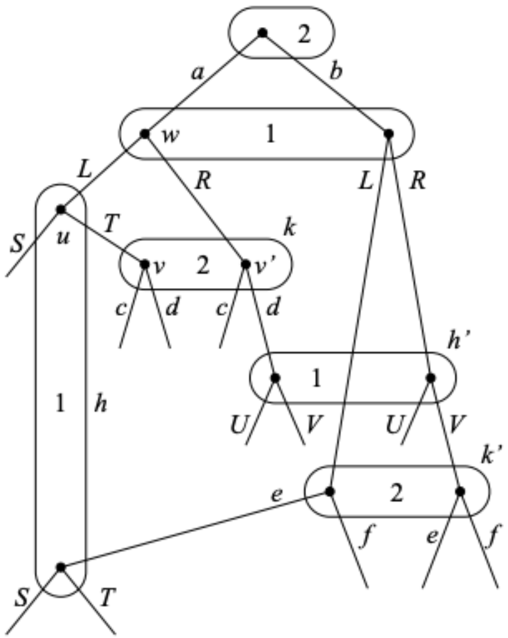


Figure 6 Extensive game of two players with perfect recall where the information sets h, k, h', k' form a cycle with respect to the “precedes” relation.

A note on Markov Games [optional]

MDPs: single player decision making with Markovian transitions, rewards → exists **deterministic** optimum strategy that is **Markovian**

- randomize at any state locally, without caring about past actions
- Policy is a map from state to actions
  - Can also do state to distribution over actions if willing to allow randomization for smoothing

Zero-sum Markov games (with or without discounting)

- Also *exists* a Markovian optimum (analogous to Nash)
  - Who cares what your opponent did in the past? No chance of coordination, since adversarial
- When discounted, solve matrix game at every state with payoffs including future payoffs (minimax Q-learning, minimax version of value iteration)

General-sum Markov games (Nash, correlated, or Stackelberg eqm)

- Whether you observe opponent's actions (and your own) is crucial!
  - Restriction to Markov strategies is an *assumption.* Many papers ignore distinction anyway…
- Observing past opens possibility of threats or coordination, e.g., repeated PD

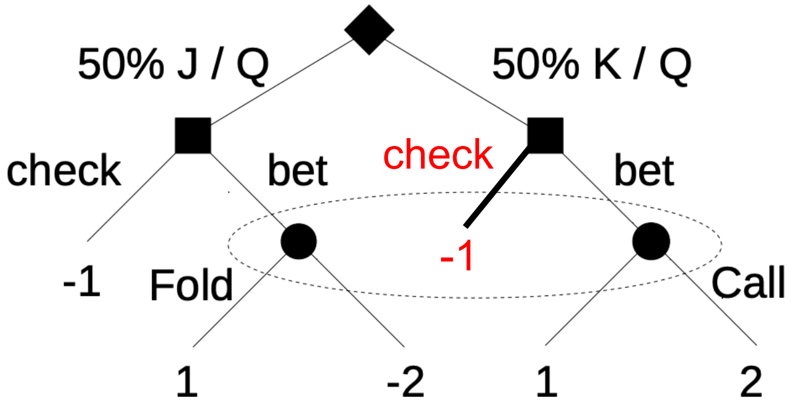
Strategy Representation in EFGs

Method 1: conversion to normal form

Cartesian product of all actions at each infoset

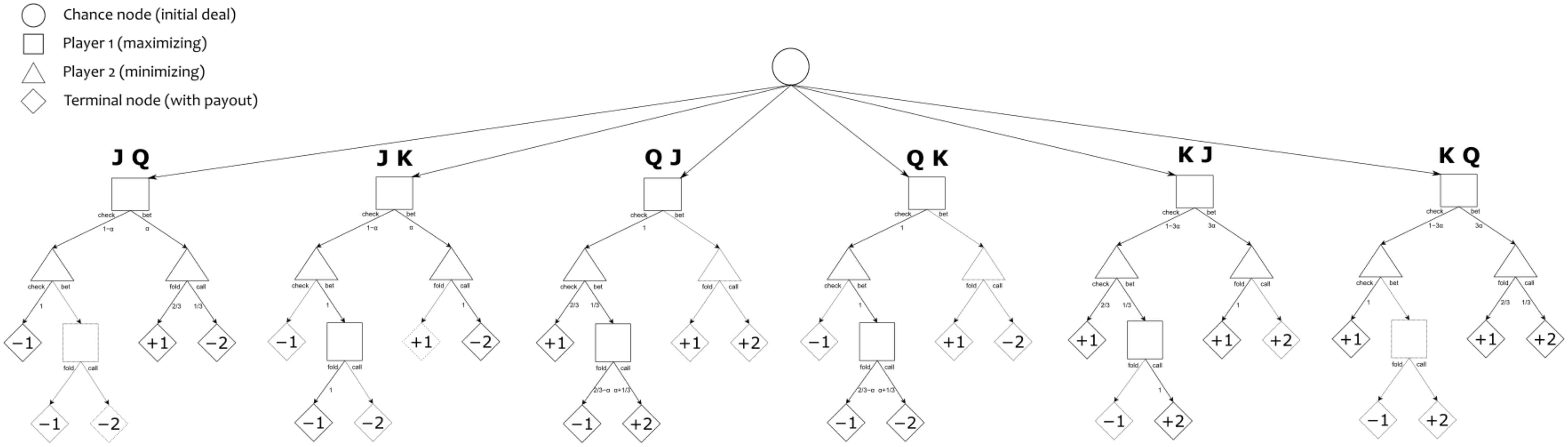
Recall example from Lecture 3

	J→Check K→Check	J→Check K→Bet	J→Bet K→Check	J→Bet K→Bet
Fold	-1	0	0	1
Call	-1	0.5	-1.5	0



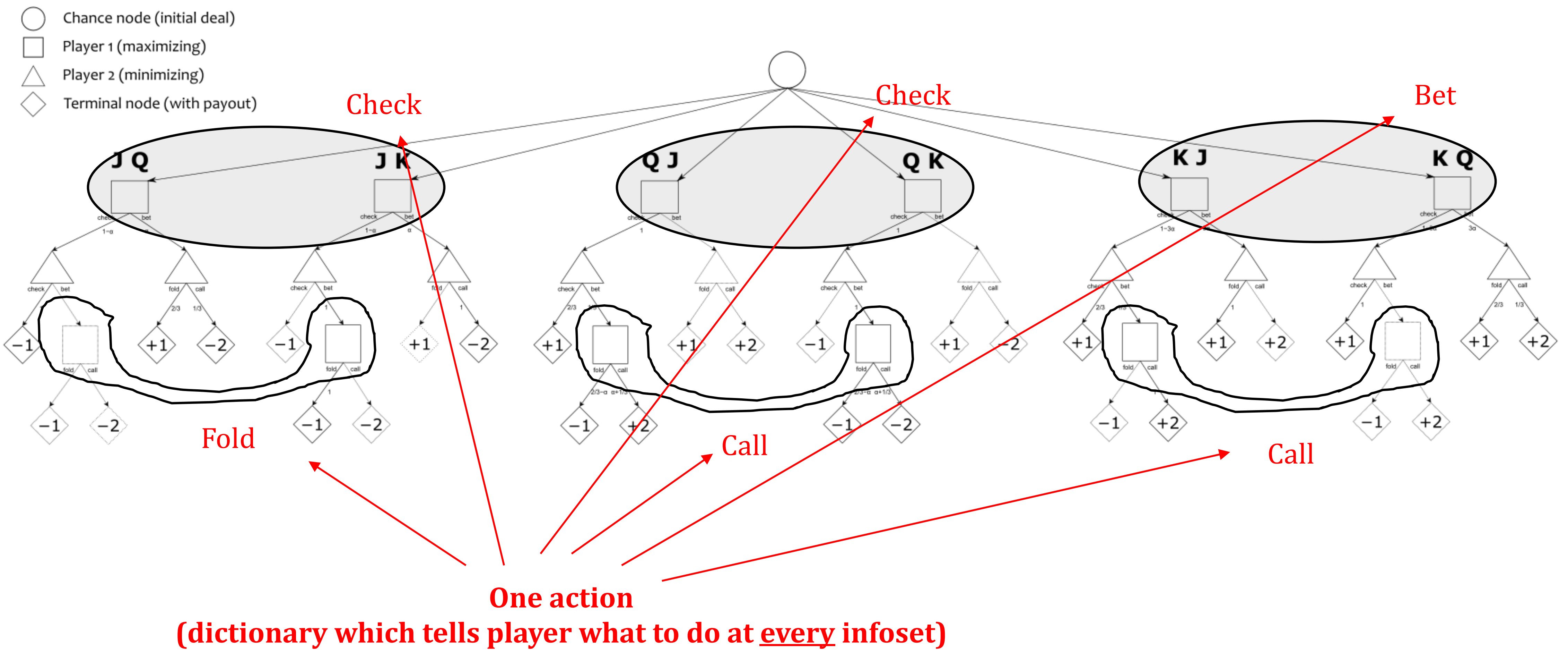
Kuhn Poker revisited

What are the normal form strategies for Player 1?



Kuhn Poker revisited (II)

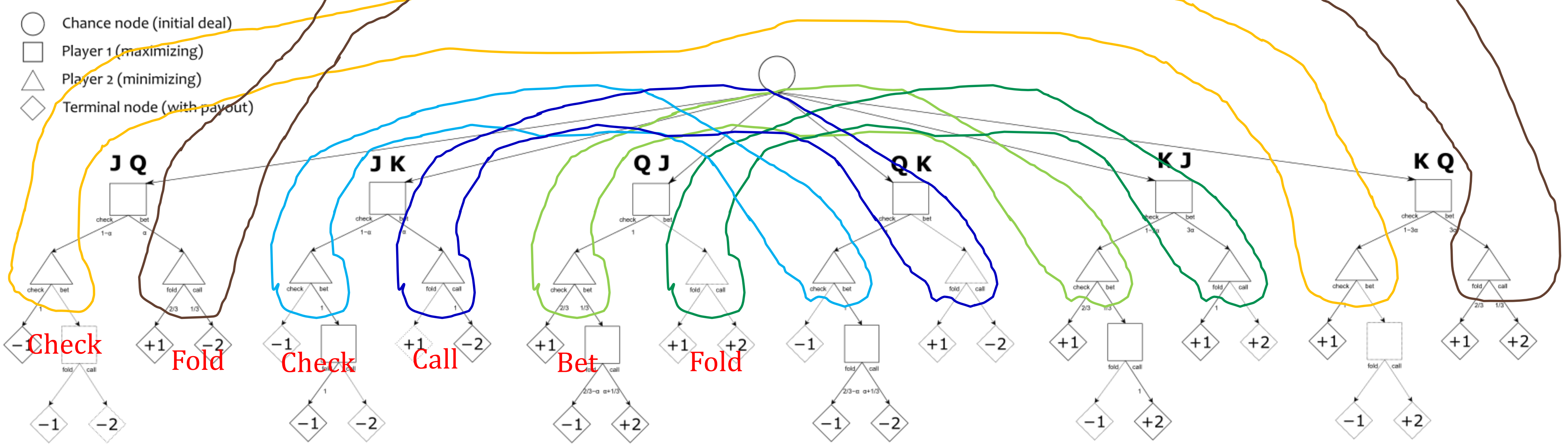
6 infosets, each with 2 actions →2^6=64 actions



Kuhn Poker revisited (III)

What are the infosets for Player 2?

What are the normal form strategies for Player 2?



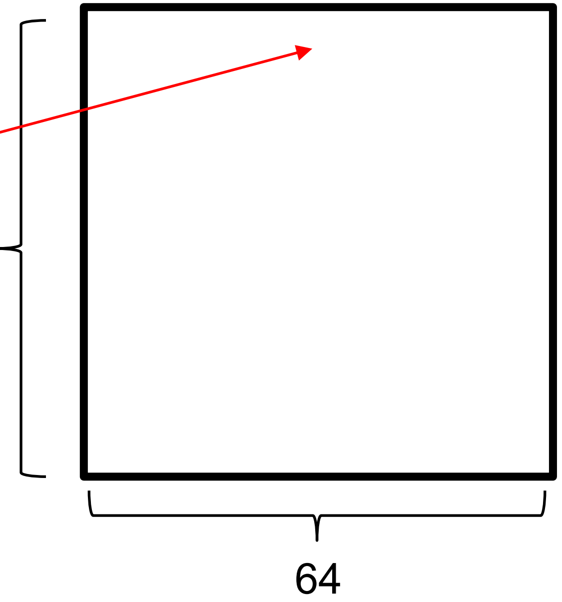
Payoffs under normal form games

Player vertices now become deterministic.

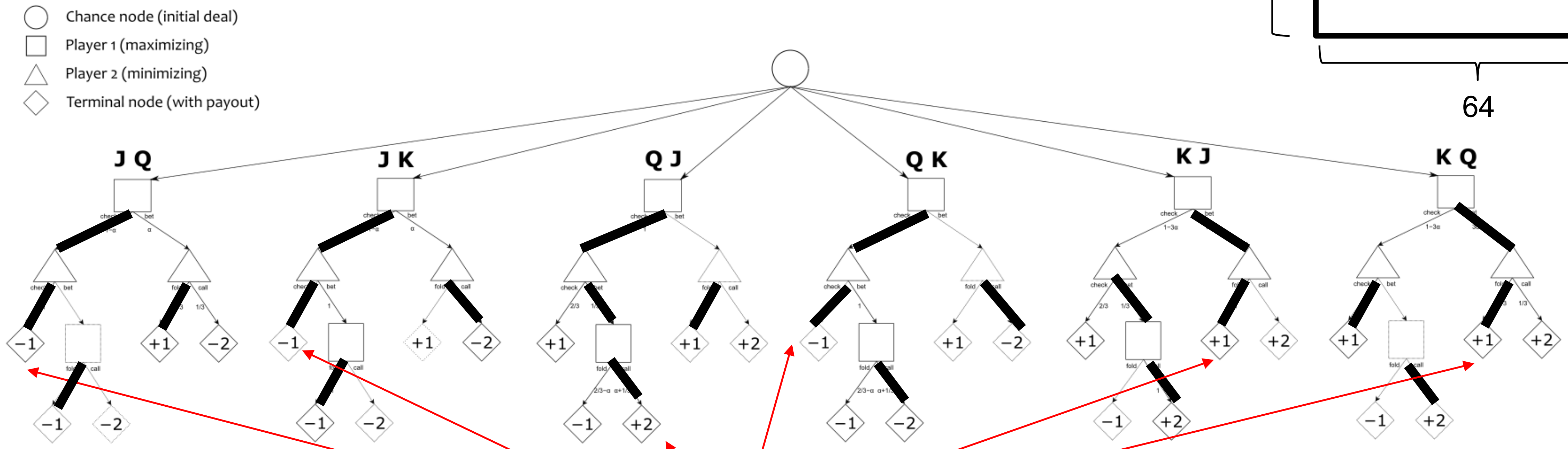
- Traverse according to chance vertices

Example: CCBFCC v.s. CFCCBF

- (-1-1+2-1+1+1)/6=1/6



One of the 64*64 entries in the payoff matrix



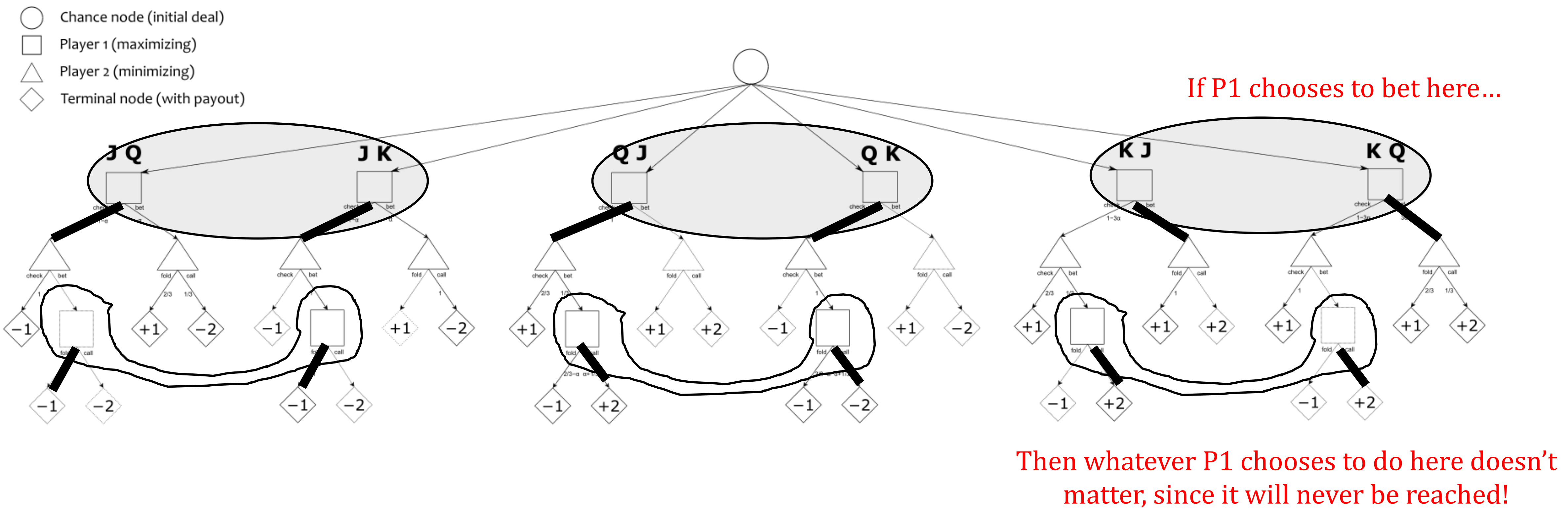
Outcomes, equally weighted (based on the root chance vertex)

The reduced normal form

Performing some actions earlier → some infosets no longer important

Example with Kuhn Poker, Player 1

- CCBFCC and CCBFCF → [**J**: Check-Fold, **Q**: Check-Fold, **K**: Bet]. $3^3 = 27$ actions!
- In literature, will be written as CCBFC*, * denotes any action



More on the reduced normal form

Will trim off more when game tree is very deep

Extreme case, only one player, no chance

- Player simply chooses which leaf it wants

Always applicable, no assumptions on perfect recall yet

But #actions can **still be exponential**

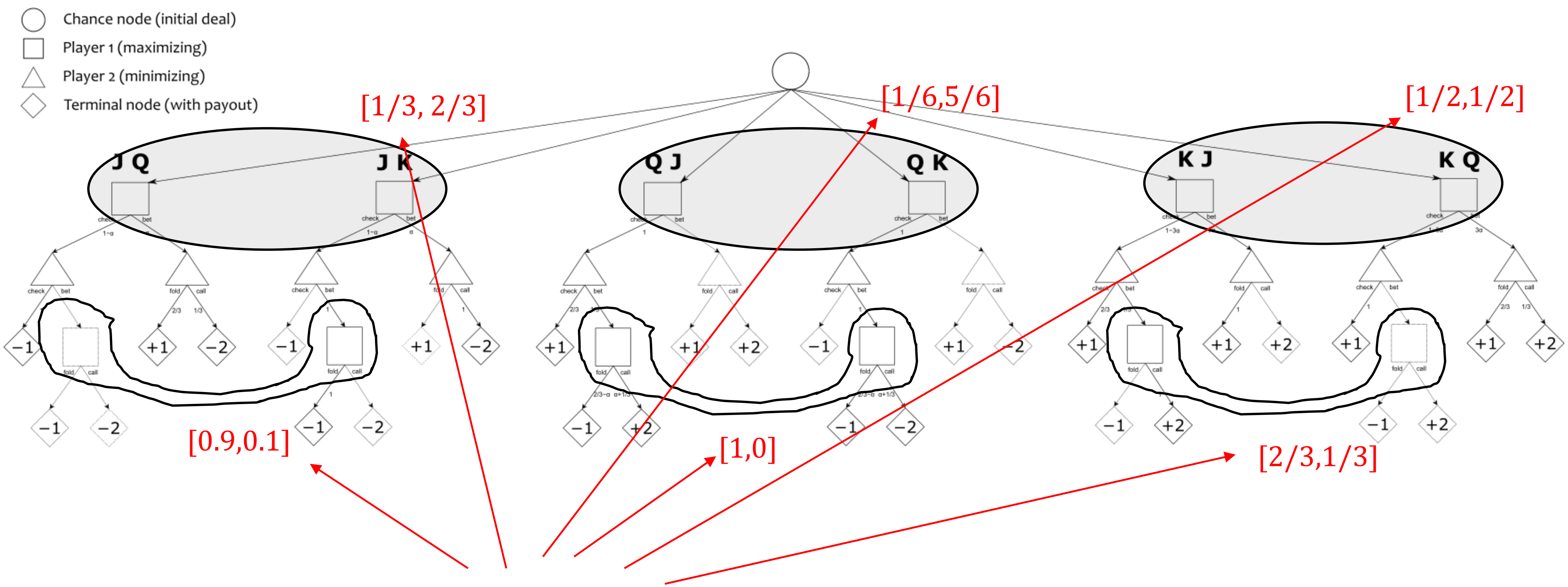
- When many parallel information sets, still need cartesian product, e.g., Player 2
- E.g., what if there were 100 cards?

Warning: we can remove “duplicated actions” since those were payoff equivalent and our choice of equilibrium concept was “nice”

- Under other equilibrium concepts (especially those with bounded rationality, e.g., Quantal response equilibrium), this will change the set of equilibrium
- QRE of the normal form game will favour actions in deeper branches of tree as compared to reduced normal form

Method 2: Behavioral Strategies

- Normal form: randomization is done ex-ante, draw from a distribution of “dictionaries”, but after that, just follow dictionary blindly
- Behavioral strategy is more natural: distribution over actions **locally** for each infoset



Distribution of actions for each infoset

Kuhn's Theorem

Under perfect recall, the space of behavioral strategies and simplex over normal-form strategies is payoff (strategically) equivalent

- Only need to consider behavioral strategies
- Much smaller in dimensions! If $|a|$ actions per infoset and $|I|$ infosets, strategy is a vector of length $|a| \cdot |I|$ rather than $|a|^{|I|}$

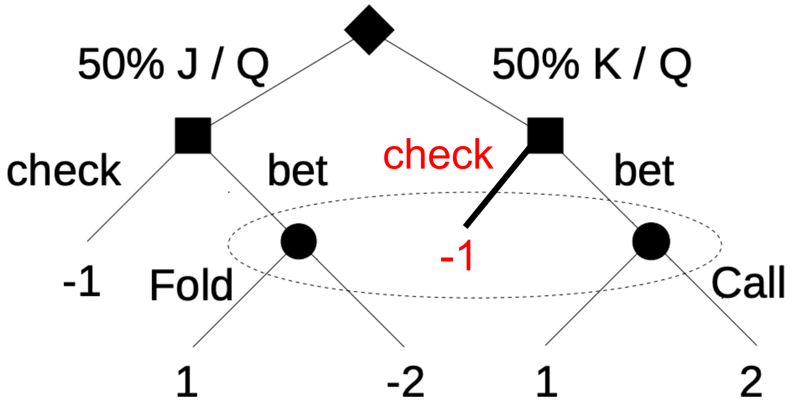
Also much easier to interpret, like MDPs

- Randomly select action when we reach an infostate, rather than sample ex-ante when game starts

Example of Kuhn’s Theorem

Player 1 Normal form strategies:

- 4 strategies, CC, CB, BC, BB
  - J: Check-K: Check
  - J: Check-K: Bet
  - J: Bet-K: Check
  - J: Bet-K: Bet
- Strategy is of the form [P(CC), P(CB), P(BC), P(BB



Player 1 Behavioral strategies are

- [P(C|Jack drawn), P(B|Jack drawn), P(C|King drawn), P(B|King drawn)]

How do we go from Normal Form → Behavioral Strategy?

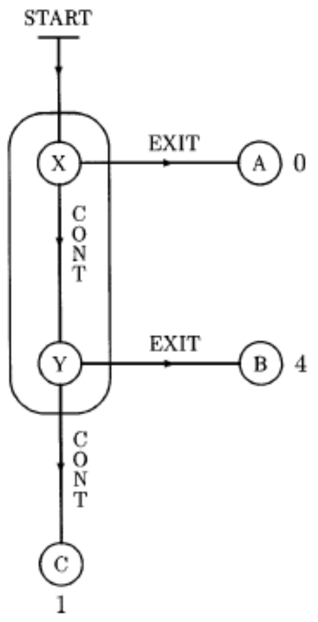
How do we go from Behavioral Form → Normal Form Strategy?

Not a 1-1 mapping, NF strategies can map to same behavioral one

Example 1 of why PR is important

In imperfect recall games with absentmindedness, i.e., the case where paths go through some infoset twice

- Exiting at B is impossible for normal form strategies
  - If action is to exit, then we will end up at A. If action is to continue, then end up at C.
- For behavioral strategies, there are at least two interpretations
  - Sample each action at every infoset based on behavioral strategy at the start of game
  - OR, sample an action at infoset "online", each time we reach it
- The first interpretation can never end up at B.
- The second one does so with probability p(1-p)
- There is no "right" or "wrong" interpretation here
  - Matter of defining what a "strategy" is
  - Most people coming from AI choose the second interpretation



Example 2 of why PR is important

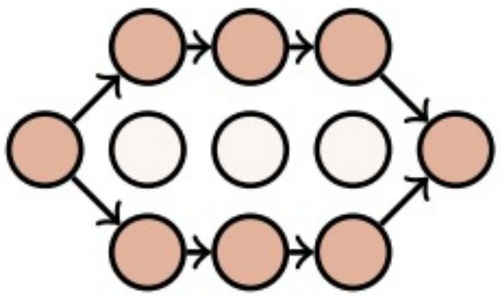
IR makes it such that there is some “low-rank” constraint

From Lecture 3: Professor pursuing a student over T steps

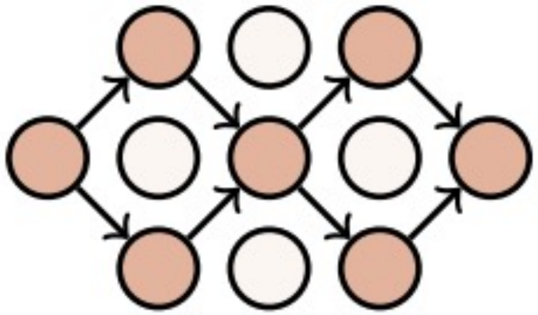
- If professor meets student, student will be assigned work
- Number of times met doesn’t matter, just binary

NE under perfect recall is for student is to go UU, DD w.p. 0.5

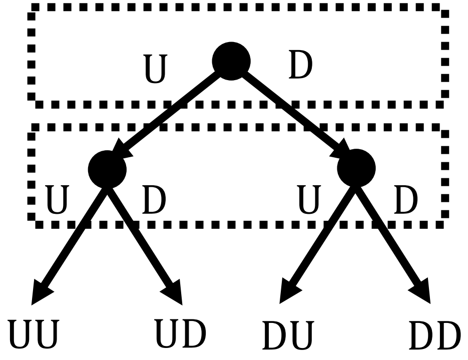
- Can be obtained by normal form strategies
- **Cannot** be obtained by behavioral strategies!



Professor



Student



Student’s perspective

The Limitation of Behavioral Strategies

Probability of reaching leaf =

- Product of player 1 action probabilities along path to leaf ×
- Product of player 2 action probabilities along path to leaf ×
- Product of chance probabilities along path to leaf

NOT bilinear, cannot write utilities in the form $x^T Ay$ where x, y are behavioral strategies

Nonconvex in this form, not as useful for computation

Summary:

- Normal form is useful for game solving, but too big
- Behavioral form is small, but not as useful for game solving

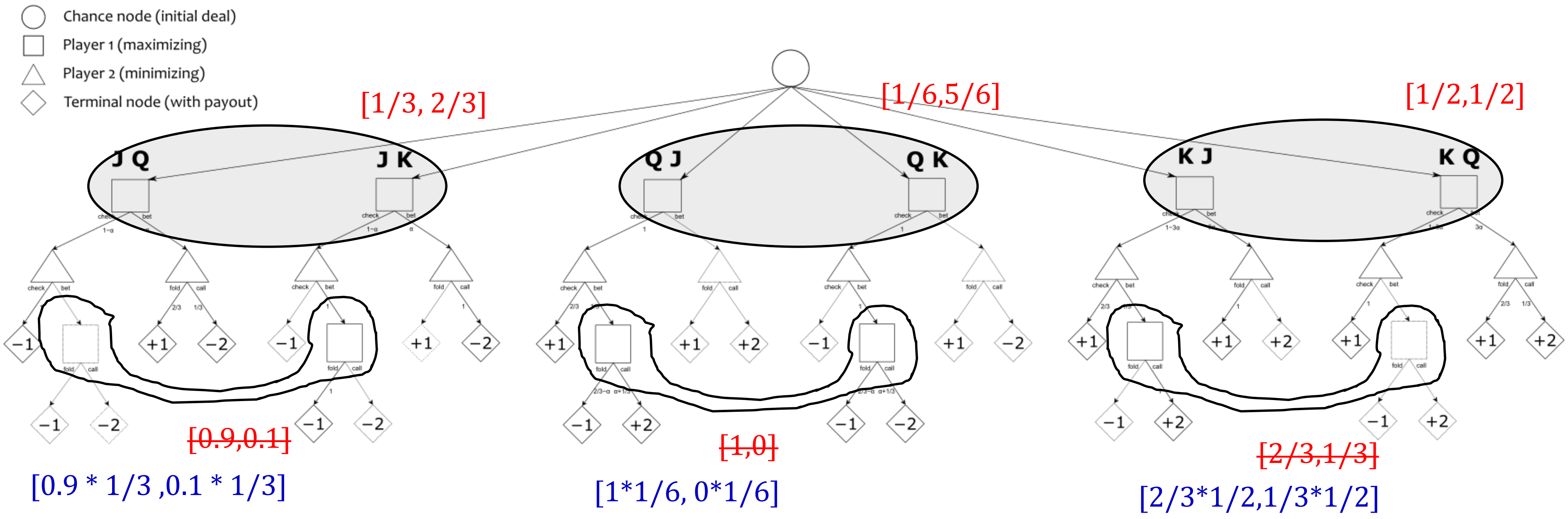
Sequence form: try to get the best of both worlds

Sequence Form and Treeplexes

Method 3: Sequence Form

Instead of probabilities of actions, use probability of **sequences**

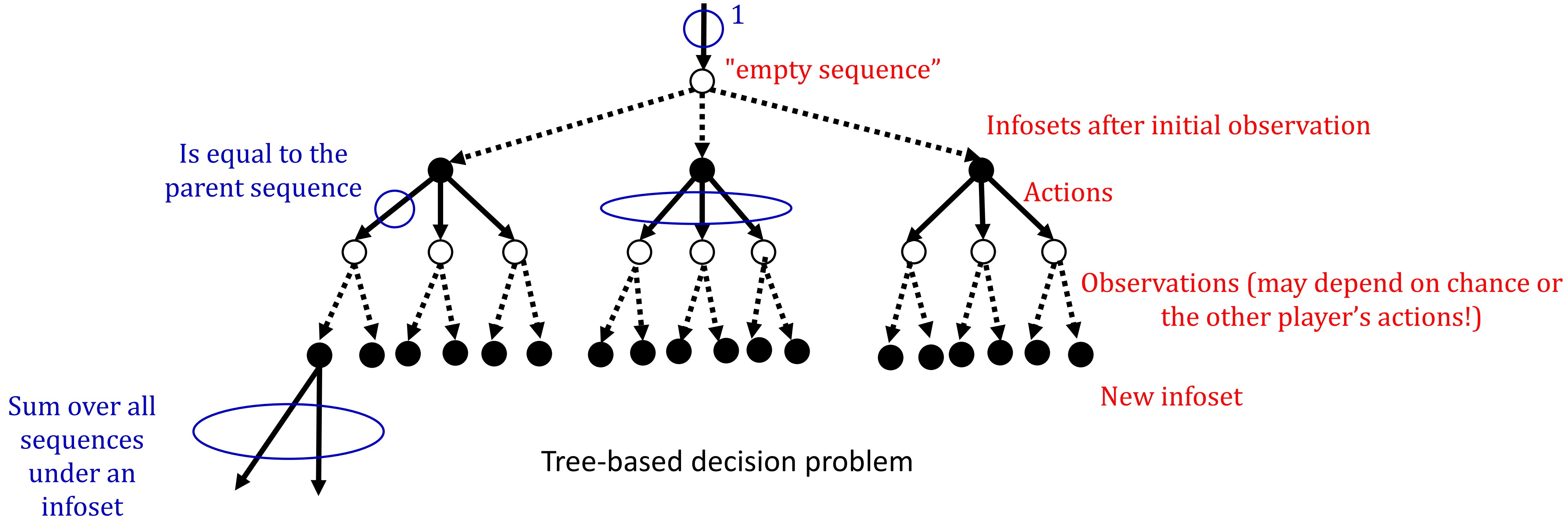
- Sequences already account for probabilities in parent sequences (past actions) taken
- Converting between the 2 is simply a matter of traversing the tree



Treeplexes

Natural strategy space for tree-based decision problems

- Recall that assuming PR we end up with a tree-like structure



Tree-based decision problem

Representing a Treeplex as polytope

Instead of the simplex, we use the **treeplex** as domains

- n = number of sequences
- $Ex = e$ gives "these sum-of-children=parent" constraints
  - e is all 0's (for all the "non-root" constraints), except for one entry, where it sums to be parent sequence (which is by default 1)

$$\mathcal{X} = \left\{ x \in \mathbb{R}^n_+ \middle| Ex = e \right\}$$

Clearly, treeplex is a generalization of the simplex

- Treeplex with one infoset is a simplex

Treeplex is convex, compact

Vertices of Treeplex are pure/deterministic strategies

Solving zero-sum EFGs using LPs

Bilinear Saddle-Point Problem in Simplices

$$\min_{x \in \mathbb{R}^n} \max_{y \in \mathbb{R}^m} x^T A y$$

$$\text{such that } \quad 1^T x = 1^T y = 1$$
$$x, y \geq 0$$

Bilinear Saddle-Point Problem in Treeplexes

$$\min_{x \in \mathcal{X}} \max_{y \in \mathcal{Y}} x^T A y$$

*Sequence Form Polytopes

*Sequence Form Payoff Matrix

Since vertices of treeplex are deterministic strategies, the saddle point is a NE

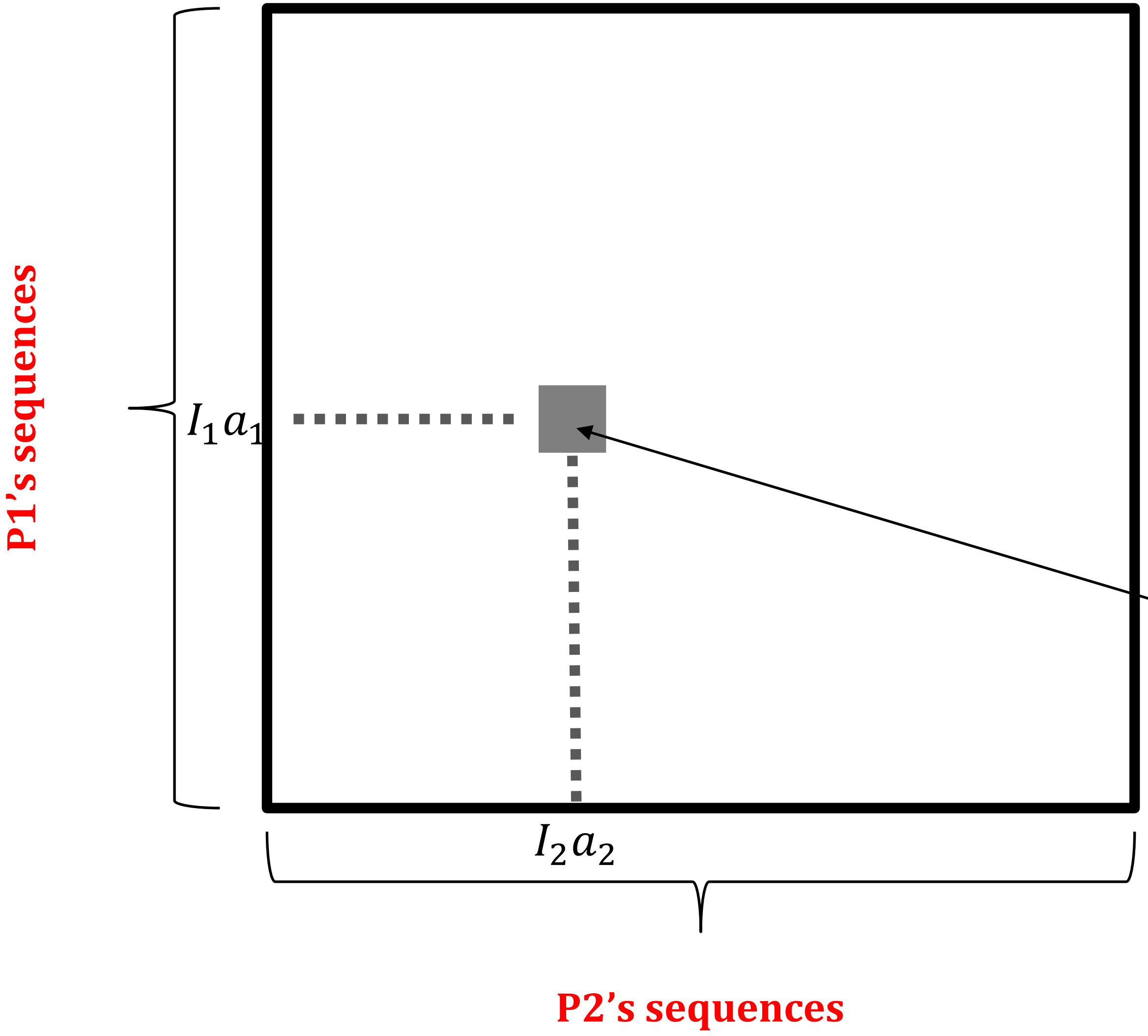
Domains of x, y are themselves polytopes, convex, compact

- Minimax theorem holds

Can find the saddle point the usual way

- Dualize the inner max problem (can be more complicated) to give a min-min problem

The Sequence Form Payoff Matrix



Recall that the probability of reaching each leaf can be decomposed into P1's, P2's, and nature probabilities

Utility of leaf × nature probabilities

Sum over *all* leaves terminating with sequences $I_1 a_1, I_2 a_2$.

*Sequence Form Payoff Matrix is MUCH smaller and sparser than normal form payoff matrix!