



भारतीय प्रौद्योगिकी संस्थान मुंबई
Indian Institute of Technology Bombay

CS 6001: Game Theory and Algorithmic Mechanism Design

Week 5 – Imperfect Information, Behavioral Strategies, and Perfect Recall

Swaprava Nath

August, 2026

Week 5 Roadmap



Lecture	Chunks (20–25 min each)	Quizzes
Lecture 9	1. What You Cannot See: Information Sets 2. IIEFG, Formally 3. Is More Information Always Good? 4. When Information Hurts Everyone	Quiz 1 after Chunk 2 Quiz 2 after Chunk 4
Lecture 10	1. Plans vs Policies 2. Equivalence of the Two 3. When Equivalence Breaks 4. Perfect Recall and Kuhn's Theorem	Quiz 1 after Chunk 2 Quiz 2 after Chunk 4



► Lecture 9: Imperfect Information Extensive Form Games

► Lecture 10: Behavioral Strategies and Perfect Recall



Chunk 1

What You Cannot See: Information Sets

(≈ 20 minutes)



The story so far

- EFGs studied so far assume perfect information: every player knows everything that happened so far
- Limits: some agents cannot see the state (poker, reconnaissance blind chess^a), and simultaneous-move games cannot be represented at all

^a<https://rbc.jhuapl.edu/>

Where We Stopped



The story so far

- EFGs studied so far assume perfect information: every player knows everything that happened so far
- Limits: some agents cannot see the state (poker, reconnaissance blind chess^a), and simultaneous-move games cannot be represented at all

^a<https://rbc.jhuapl.edu/>

Question

Chess engines have out-played humans since 1997. Was chess ever *solved*, like tic-tac-toe?

Where We Stopped



The story so far

- EFGs studied so far assume perfect information: every player knows everything that happened so far
- Limits: some agents cannot see the state (poker, reconnaissance blind chess^a), and simultaneous-move games cannot be represented at all

^a<https://rbc.jhuapl.edu/>

Question

Chess engines have out-played humans since 1997. Was chess ever *solved*, like tic-tac-toe?

Answer

No. **Solved** means holding the exact optimal strategy everywhere – true only for small games like checkers (2007). Chess engines are just **superhuman**: full observability lets deep search work directly on the board. Poker is imperfect information and still unsolved; Libratus (2017) and Pluribus (2019) instead **approximate a Nash equilibrium** via regret minimization over **information sets** – exactly what we build today.

Two AI Labs, One Product Launch



Two AI labs are racing to release a new model next quarter.

Two AI Labs, One Product Launch



Two AI labs are racing to release a new model next quarter.

- **T:** test thoroughly

Two AI Labs, One Product Launch



Two AI labs are racing to release a new model next quarter.

- **T**: test thoroughly
- **S**: ship fast

Two AI Labs, One Product Launch



Two AI labs are racing to release a new model next quarter.

- **T**: test thoroughly
- **S**: ship fast

Plans lock before the quarter starts. **Neither lab sees the other's plan.**

Two AI Labs, One Product Launch



Two AI labs are racing to release a new model next quarter.

- **T**: test thoroughly
- **S**: ship fast

Plans lock before the quarter starts. **Neither lab sees the other's plan.**

		Lab 2	
		T	S
Lab 1	T	5,5	0,6
	S	6,0	1,1

The Launch Race

Two AI Labs, One Product Launch



Two AI labs are racing to release a new model next quarter.

- **T**: test thoroughly
- **S**: ship fast

Plans lock before the quarter starts. **Neither lab sees the other's plan.**

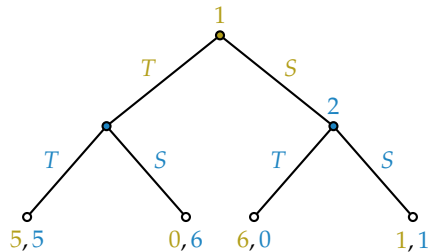
		Lab 2	
		T	S
Lab 1	T	5,5	0,6
	S	6,0	1,1

The Launch Race

Question

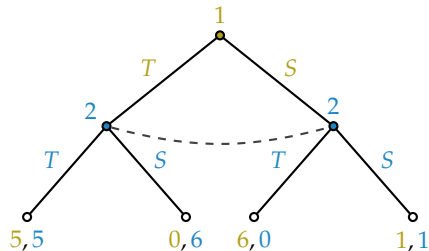
A repackaged **Prisoner's Dilemma**. Can we draw it as a tree?

Drawing a Simultaneous Move as a Tree





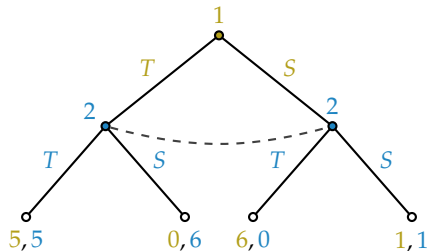
Drawing a Simultaneous Move as a Tree



- In IIEFG, indistinguishable nodes are connected via a dotted line.



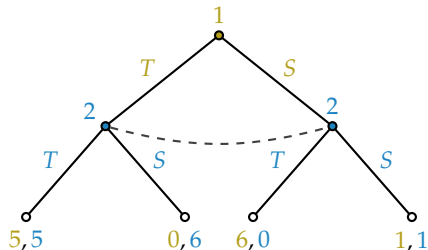
Drawing a Simultaneous Move as a Tree



- In IIEFG, indistinguishable nodes are connected via a dotted line.
- Lab 2 does not know which node/history the game is in



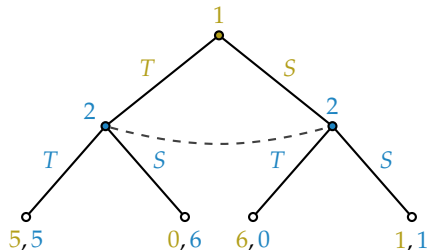
Drawing a Simultaneous Move as a Tree



- In IIEFG, indistinguishable nodes are connected via a dotted line.
- Lab 2 does not know which node/history the game is in
- These indistinguishable histories form an **information set** for player 2.



Drawing a Simultaneous Move as a Tree

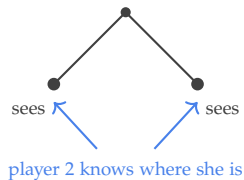


- In IIEFG, indistinguishable nodes are connected via a dotted line.
- Player 2 does not know which node/history the game is in
- These indistinguishable histories form an **information set** for player 2.
- More general representation than PIEFG since information sets can be singleton

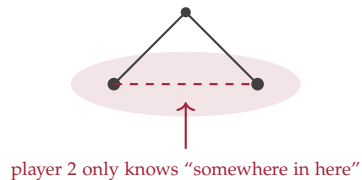
The Picture to Keep in Your Head



Perfect information



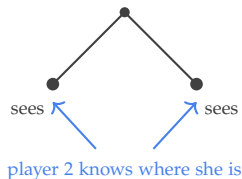
Imperfect information



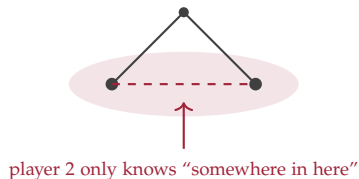
The Picture to Keep in Your Head



Perfect information



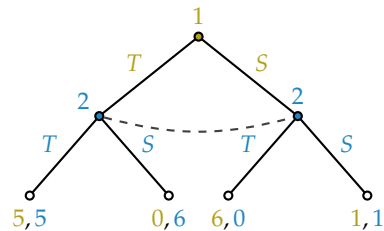
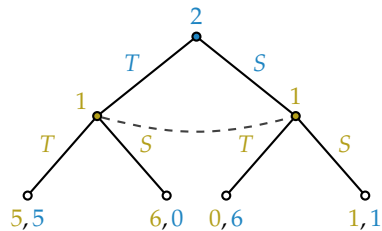
Imperfect information



Translation for the ML crowd

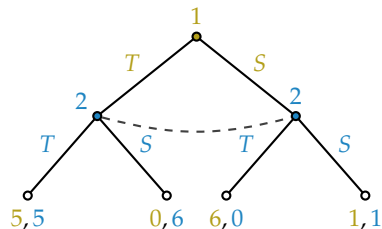
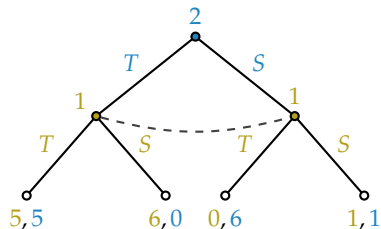
An information set is the **observation** an agent conditions on. The partition into information sets is the agent's sensor. Two histories in the same cell are two world states her sensor cannot separate.

The Tree Is Not Unique





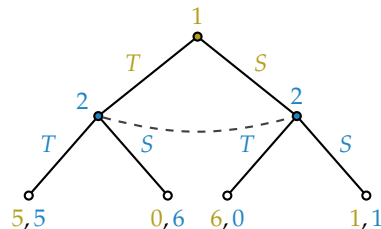
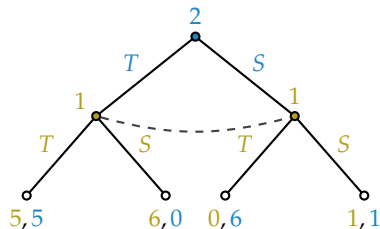
The Tree Is Not Unique



- The launch race can also be represented with the information set of player 1 being non-singleton.



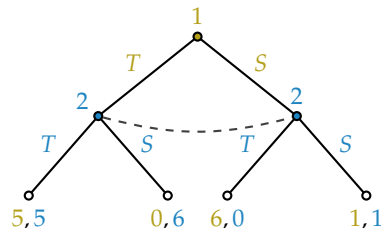
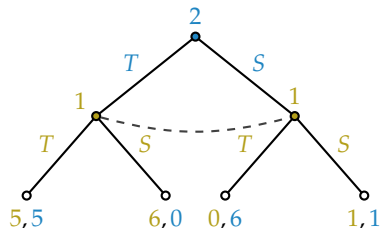
The Tree Is Not Unique



- The launch race can also be represented with the information set of player 1 being non-singleton.
- IIEFGs are not unique for a given simultaneous move game



The Tree Is Not Unique



- The launch race can also be represented with the information set of player 1 being non-singleton.
- IIEFGs are not unique for a given simultaneous move game
- Whoever we draw “first” is a bookkeeping choice, not a modeling claim



Chunk 2

IIEFG, Formally

(≈ 20 minutes)

Games with Imperfect Information



Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$

Games with Imperfect Information



Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG

Games with Imperfect Information



Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$

Games with Imperfect Information



Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$

Games with Imperfect Information



Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$ with the property that $X(h) = X(h')$ and $P(h) = P(h') = i$, whenever $\exists j$ s.t. $h, h' \in I_i^j$.

Games with Imperfect Information



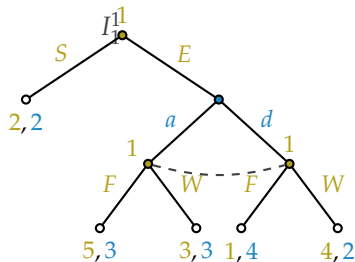
Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$ with the property that $X(h) = X(h')$ and $P(h) = P(h') = i$, whenever $\exists j$ s.t. $h, h' \in I_i^j$.

- I_i^j s are called an **information set** of player i and I_i is the collection of information sets of i .
- At an information set, the player and her available actions are the same.
- The player is uncertain about which history in the information set is reached.



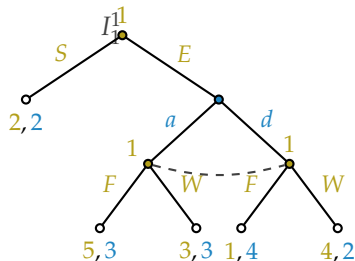
A Worked Example: Reading the Notation



- Player 1 moves twice: at the root, resolve (S) or escalate (E); after the specialist's private call, follow up (F) or wait (W)



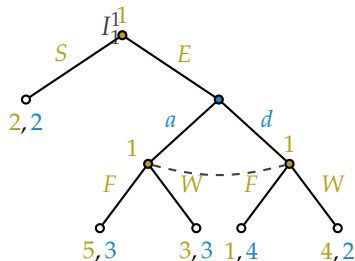
A Worked Example: Reading the Notation



- Player 1 moves twice: at the root, resolve (S) or escalate (E); after the specialist's private call, follow up (F) or wait (W)
- First move: singleton information set $I_1^1 = \{w\}$



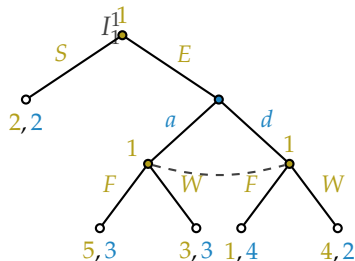
A Worked Example: Reading the Notation



- Player 1 moves twice: at the root, resolve (S) or escalate (E); after the specialist's private call, follow up (F) or wait (W)
- First move: singleton information set $I_1^1 = \{w\}$
- She never observes approval (a) vs. denial (d), so $I_1^2 = \{h_a, h_d\}$, size 2



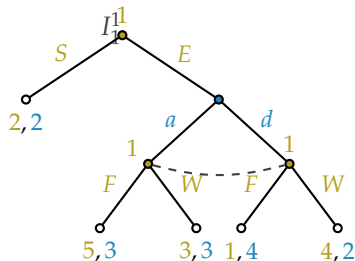
A Worked Example: Reading the Notation



- Player 1 moves twice: at the root, resolve (S) or escalate (E); after the specialist's private call, follow up (F) or wait (W)
- First move: singleton information set $I_1^1 = \{w\}$
- She never observes approval (a) vs. denial (d), so $I_1^2 = \{h_a, h_d\}$, size 2
- $X(h_a) = X(h_d) = \{F, W\}$: same actions at both, exactly the condition the next slide makes precise

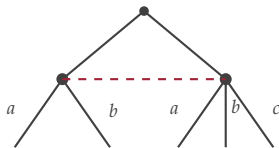


A Worked Example: Reading the Notation



- Player 1 moves twice: at the root, resolve (S) or escalate (E); after the specialist's private call, follow up (F) or wait (W)
- First move: singleton information set $I_1^1 = \{w\}$
- She never observes approval (a) vs. denial (d), so $I_1^2 = \{h_a, h_d\}$, size 2
- $X(h_a) = X(h_d) = \{F, W\}$: same actions at both, exactly the condition the next slide makes precise
- $S_1 = \prod_{j=1}^2 X(I_j^1) = \{S, E\} \times \{F, W\}$: four pure strategies

Why That Last Condition Is Not Optional



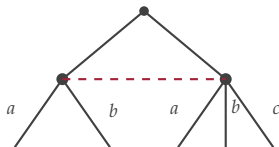
Not allowed

If the two nodes offered different action sets, the player could tell them apart just by *looking at her own menu*.

Counting your options is free information. So $X(h) = X(h')$ is forced on us.



Why That Last Condition Is Not Optional



Not allowed

If the two nodes offered different action sets, the player could tell them apart just by *looking at her own menu*.

Counting your options is free information. So $X(h) = X(h')$ is forced on us.

Same idea, in code

If your agent's action mask differs across two states, the mask has already leaked the state. An information set must have one mask.

Games with Imperfect Information (contd.)



Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$ with the property that $X(h) = X(h')$ and $P(h) = P(h') = i$, whenever $\exists j$ s.t. $h, h' \in I_i^j$.

- Some differences with PIEFG

Games with Imperfect Information (contd.)



Definition (IIEFG)

An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$ with the property that $X(h) = X(h')$ and $P(h) = P(h') = i$, whenever $\exists j$ s.t. $h, h' \in I_i^j$.

- Some differences with PIEFG

- Since actions at an information set are identical, X (action set function) can be defined over I_i^j s i.e.,
$$X(h) = X(h') = X(I_i^j), \forall h, h' \in I_i^j$$

Games with Imperfect Information (contd.)



Definition (IIIEFG)

An IIIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$ with the property that $X(h) = X(h')$ and $P(h) = P(h') = i$, whenever $\exists j$ s.t. $h, h' \in I_i^j$.

- Some differences with PIEFG

- Since actions at an information set are identical, X (action set function) can be defined over I_i^j s i.e.,
$$X(h) = X(h') = X(I_i^j), \forall h, h' \in I_i^j$$
- Strategies can also be defined over information sets, i.e., strategy set of a player $i \in N$ is defined as the Cartesian product of actions available to i at her information sets

$$S_i = \prod_{I' \in I_i} X(I') = \prod_{j=1}^{j=k(i)} X(I_i^j)$$

Games with Imperfect Information (contd.)



Definition (IIIEFG)

An IIIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$ with the property that $X(h) = X(h')$ and $P(h) = P(h') = i$, whenever $\exists j$ s.t. $h, h' \in I_i^j$.

- Some differences with PIEFG
 - Since actions at an information set are identical, X (action set function) can be defined over I_i^j s i.e.,
$$X(h) = X(h') = X(I_i^j), \forall h, h' \in I_i^j$$
 - Strategies can also be defined over information sets, i.e., strategy set of a player $i \in N$ is defined as the Cartesian product of actions available to i at her information sets
$$S_i = \prod_{I' \in I_i} X(I') = \prod_{j=1}^{j=k(i)} X(I_i^j)$$
- With IIIEFG, NFGs can be represented using EFGs, although not very succinct.

Games with Imperfect Information (contd.)

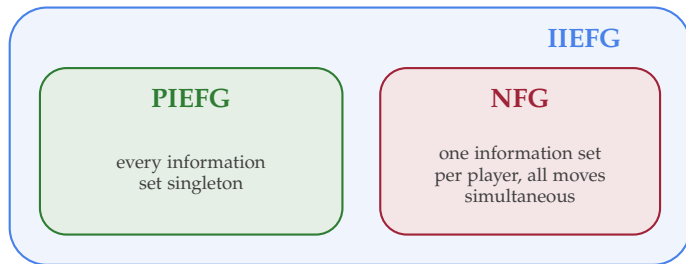


Definition (IIEFG)

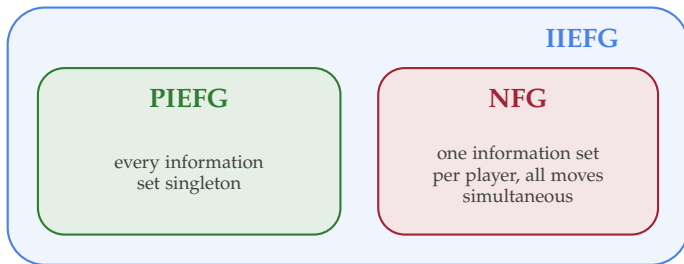
An IIEFG is tuple $\langle N, A, H, X, P, (u_i)_{i \in N}, (I_i)_{i \in N} \rangle$ where $\langle N, A, H, X, P, (u_i)_{i \in N} \rangle$ is a PIEFG and for every $i \in N$, $I_i = (I_i^1, I_i^2, I_i^3, \dots, I_i^{k(i)})$ is a partition of $\{h \in H \setminus Z : P(h) = i\}$ with the property that $X(h) = X(h')$ and $P(h) = P(h') = i$, whenever $\exists j$ s.t. $h, h' \in I_i^j$.

- Some differences with PIEFG
 - Since actions at an information set are identical, X (action set function) can be defined over I_i^j s i.e.,
$$X(h) = X(h') = X(I_i^j), \forall h, h' \in I_i^j$$
 - Strategies can also be defined over information sets, i.e., strategy set of a player $i \in N$ is defined as the Cartesian product of actions available to i at her information sets
$$S_i = \prod_{I' \in I_i} X(I') = \prod_{j=1}^{j=k(i)} X(I_i^j)$$
- With IIEFG, NFGs can be represented using EFGs, although not very succinct.
- IIEFG is a richer representation than both NFG and PIEFG.

The Representation Hierarchy

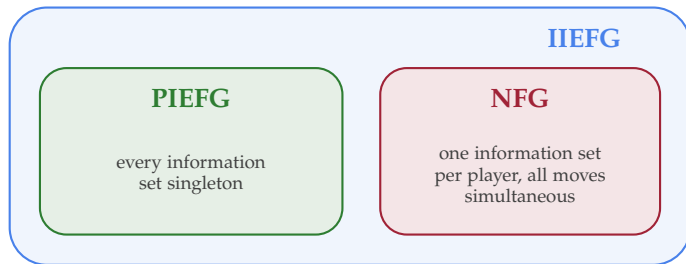


The Representation Hierarchy



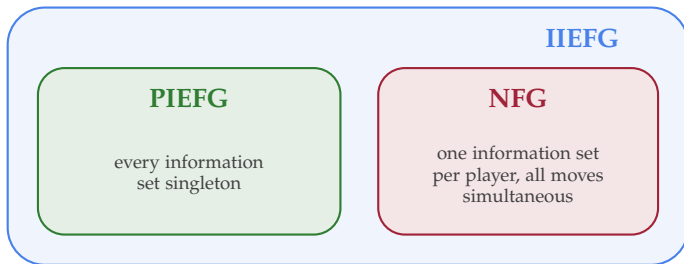
- Collapse every information set to a singleton and you are back in a PIIEFG

The Representation Hierarchy



- Collapse every information set to a singleton and you are back in a PIIEFG
- Give the second mover one big information set and you are back in an NFG

The Representation Hierarchy



- Collapse every information set to a singleton and you are back in a PIEFG
- Give the second mover one big information set and you are back in an NFG
- Everything we proved about PIEFGs is a special case, not a separate theory



In-Class Quiz 1

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.



Chunk 3

Is More Information Always Good?

(≈ 20 minutes)

A Red Team Against a Guardrail



- An API serves one of two model checkpoints. A red-teamer (Player I) attacks, a guardrail (Player II) filters.

A Red Team Against a Guardrail



- An API serves one of two model checkpoints. A red-teamer (Player I) attacks, a guardrail (Player II) filters.
- The number in each cell is the **probability the attack gets through**. Player I wants it high, Player II wants it low: a two-player zero-sum game.



A Red Team Against a Guardrail

- An API serves one of two model checkpoints. A red-teamer (Player I) attacks, a guardrail (Player II) filters.
- The number in each cell is the **probability the attack gets through**. Player I wants it high, Player II wants it low: a two-player zero-sum game.
- Each of the following matrices are chosen w.p. $\frac{1}{2}$, but no player sees the realization of this randomization process

		Player II				Player II	
		<i>L</i>	<i>R</i>			<i>L</i>	<i>R</i>
Player I	<i>T</i>	0	$\frac{1}{2}$	Player I	<i>T</i>	1	0
	<i>B</i>	0	1		<i>B</i>	$\frac{1}{2}$	0
Matrix G_1				Matrix G_2			



A Red Team Against a Guardrail

- An API serves one of two model checkpoints. A red-teamer (Player I) attacks, a guardrail (Player II) filters.
- The number in each cell is the **probability the attack gets through**. Player I wants it high, Player II wants it low: a two-player zero-sum game.
- Each of the following matrices are chosen w.p. $\frac{1}{2}$, but no player sees the realization of this randomization process

		Player II				Player II	
		<i>L</i>	<i>R</i>			<i>L</i>	<i>R</i>
Player I	<i>T</i>	0	$\frac{1}{2}$	Player I	<i>T</i>	1	0
	<i>B</i>	0	1		<i>B</i>	$\frac{1}{2}$	0
Matrix G_1				Matrix G_2			

- *T*: tool-use framing, *B*: obfuscated payload; *L*: lexical filter, *R*: reasoning classifier



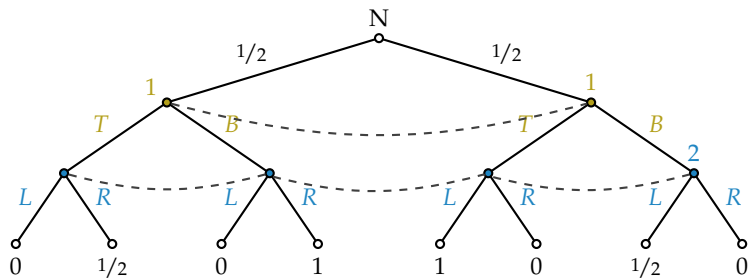
A Red Team Against a Guardrail

- An API serves one of two model checkpoints. A red-teamer (Player I) attacks, a guardrail (Player II) filters.
- The number in each cell is the **probability the attack gets through**. Player I wants it high, Player II wants it low: a two-player zero-sum game.
- Each of the following matrices are chosen w.p. $\frac{1}{2}$, but no player sees the realization of this randomization process

		Player II				Player II	
		<i>L</i>	<i>R</i>			<i>L</i>	<i>R</i>
Player I	<i>T</i>	0	$\frac{1}{2}$	Player I	<i>T</i>	1	0
	<i>B</i>	0	1		<i>B</i>	$\frac{1}{2}$	0
Matrix G_1				Matrix G_2			

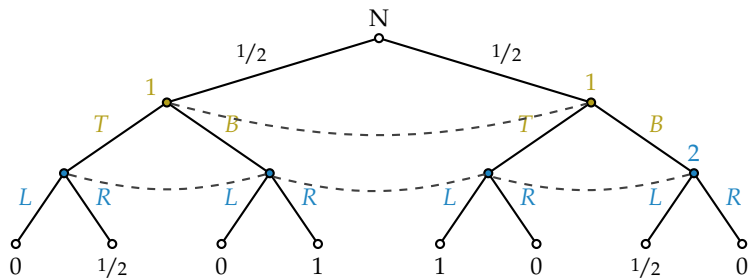
- *T*: tool-use framing, *B*: obfuscated payload; *L*: lexical filter, *R*: reasoning classifier
- **What is the extensive form representation?**

The Same Game as a Tree





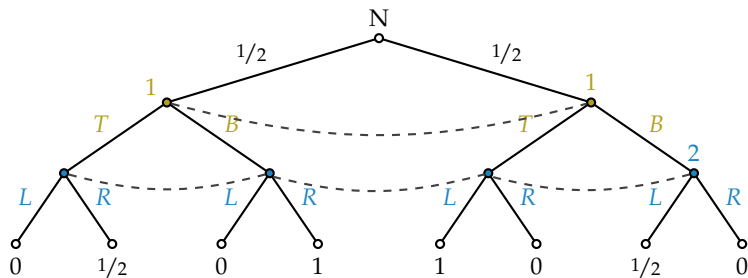
The Same Game as a Tree



- N at the root is **Nature**: it picks the checkpoint, G_1 on the left, G_2 on the right



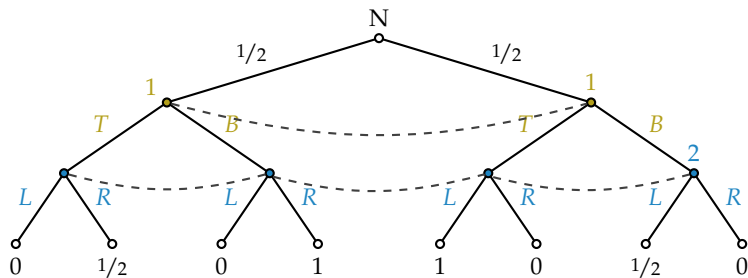
The Same Game as a Tree



- N at the root is **Nature**: it picks the checkpoint, G_1 on the left, G_2 on the right
- Player I has **one** information set: she does not know which checkpoint is live



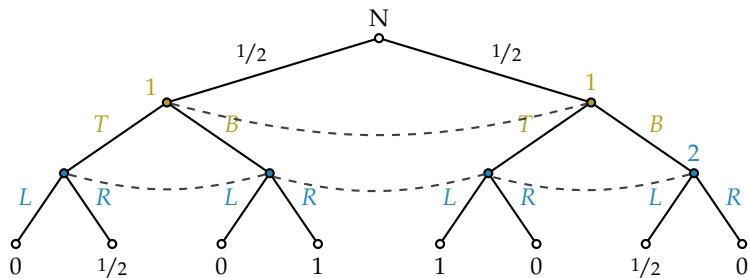
The Same Game as a Tree



- N at the root is **Nature**: it picks the checkpoint, G_1 on the left, G_2 on the right
- Player I has **one** information set: she does not know which checkpoint is live
- Player II has **one** information set spanning all four nodes



The Same Game as a Tree



- N at the root is **Nature**: it picks the checkpoint, G_1 on the left, G_2 on the right
- Player I has **one** information set: she does not know which checkpoint is live
- Player II has **one** information set spanning all four nodes
- **What is the normal form representation?**

Back to a Matrix



		L	R
Player I	T	$1/2$	$1/4$
	B	$1/4$	$1/2$

Expected success probability with nobody informed

Back to a Matrix



		L	R
Player I	T	$1/2$	$1/4$
	B	$1/4$	$1/2$

Expected success probability with nobody informed

- Each entry averages the two checkpoints: e.g. (T, L) gives $\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1 = 1/2$

Back to a Matrix



		L	R
Player I	T	$\frac{1}{2}$	$\frac{1}{4}$
	B	$\frac{1}{4}$	$\frac{1}{2}$

Expected success probability with nobody informed

- Each entry averages the two checkpoints: e.g. (T, L) gives $\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1 = \frac{1}{2}$
- **What is an MSNE of this game?**

Back to a Matrix



		L	R
Player I	T	$1/2$	$1/4$
	B	$1/4$	$1/2$

Expected success probability with nobody informed

- Each entry averages the two checkpoints: e.g. (T, L) gives $\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1 = 1/2$
- **What is an MSNE of this game?**
- **What is the value of this game?**

Back to a Matrix



		L	R
Player I	T	$\frac{1}{2}$	$\frac{1}{4}$
	B	$\frac{1}{4}$	$\frac{1}{2}$

Expected success probability with nobody informed

- Each entry averages the two checkpoints: e.g. (T, L) gives $\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1 = 1/2$
- **What is an MSNE of this game?**
- **What is the value of this game?**
- **MSNE:** $\left(\left(\frac{1}{2}, \frac{1}{2} \right), \left(\frac{1}{2}, \frac{1}{2} \right) \right)$, **value** = $\frac{3}{8}$

Back to a Matrix



		L	R
Player I	T	$\frac{1}{2}$	$\frac{1}{4}$
	B	$\frac{1}{4}$	$\frac{1}{2}$

Expected success probability with nobody informed

- Each entry averages the two checkpoints: e.g. (T, L) gives $\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1 = 1/2$
- **What is an MSNE of this game?**
- **What is the value of this game?**
- **MSNE:** $\left(\left(\frac{1}{2}, \frac{1}{2} \right), \left(\frac{1}{2}, \frac{1}{2} \right) \right)$, **value** = $\frac{3}{8}$
- In words: attacks land about 37.5% of the time

Now the Attacker Can Fingerprint the Checkpoint



- Suppose Player I can tell which checkpoint is live – a version banner, a latency signature, a tokenizer quirk. Player II still cannot.

Now the Attacker Can Fingerprint the Checkpoint

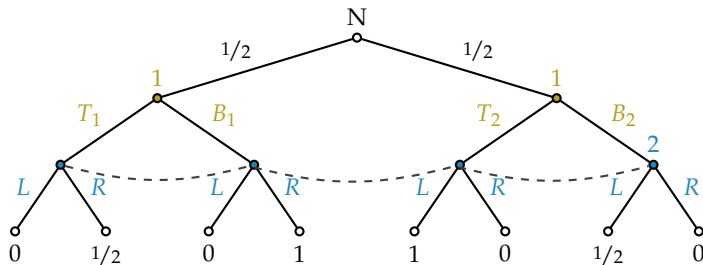


- Suppose Player I can tell which checkpoint is live – a version banner, a latency signature, a tokenizer quirk. Player II still cannot.
- Player I's single information set **splits into two**



Now the Attacker Can Fingerprint the Checkpoint

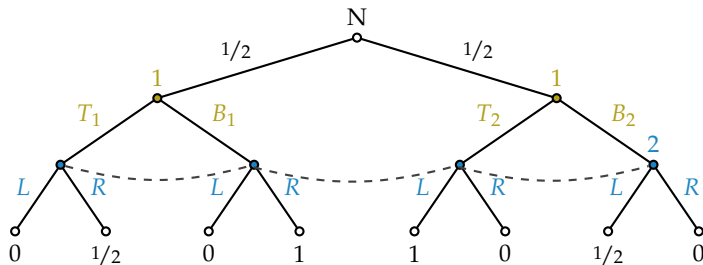
- Suppose Player I can tell which checkpoint is live – a version banner, a latency signature, a tokenizer quirk. Player II still cannot.
- Player I's single information set **splits into two**





Now the Attacker Can Fingerprint the Checkpoint

- Suppose Player I can tell which checkpoint is live – a version banner, a latency signature, a tokenizer quirk. Player II still cannot.
- Player I's single information set **splits into two**



What are the strategies now? What is the NFG representation?

The Informed Attacker's Matrix



		L	R
Player I	T_1T_2	$1/2$	$1/4$
	T_1B_2	$1/4$	$1/4$
	B_1T_2	$1/2$	$1/2$
	B_1B_2	$1/4$	$1/2$

Player I informed

- Player I now has 4 pure strategies: two actions, T and B , at each of the two checkpoints

The Informed Attacker's Matrix



		L	R
Player I	T_1T_2	$1/2$	$1/4$
	T_1B_2	$1/4$	$1/4$
	B_1T_2	$1/2$	$1/2$
	B_1B_2	$1/4$	$1/2$

Player I informed

- Player I now has 4 pure strategies: two actions, T and B , at each of the two checkpoints
- **What is an MSNE and value of this game?**

The Informed Attacker's Matrix



		L	R
Player I	T_1T_2	1/2	1/4
	T_1B_2	1/4	1/4
	B_1T_2	1/2	1/2
	B_1B_2	1/4	1/2

Player I informed

- Player I now has 4 pure strategies: two actions, T and B , at each of the two checkpoints
- **What is an MSNE and value of this game?**
- **MSNE:** $((1(B_1T_2)), (p, 1 - p)), p \in [0, 1], \text{value} = \frac{1}{2}$



The Informed Attacker's Matrix

		L	R
Player I	T_1T_2	1/2	1/4
	T_1B_2	1/4	1/4
	B_1T_2	1/2	1/2
	B_1B_2	1/4	1/2

Player I informed

- Player I now has 4 pure strategies: two actions, T and B , at each of the two checkpoints
- **What is an MSNE and value of this game?**
- **MSNE:** $((1(B_1T_2)), (p, 1 - p)), p \in [0, 1], \text{value} = \frac{1}{2}$
- B_1T_2 **weakly dominates** everything: it is worth $\frac{1}{2}$ whatever Player II does, so Player II has no reply that helps



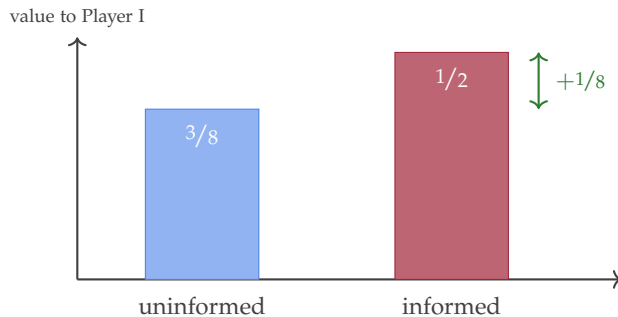
The Informed Attacker's Matrix

		L	R
Player I	T_1T_2	1/2	1/4
	T_1B_2	1/4	1/4
	B_1T_2	1/2	1/2
	B_1B_2	1/4	1/2

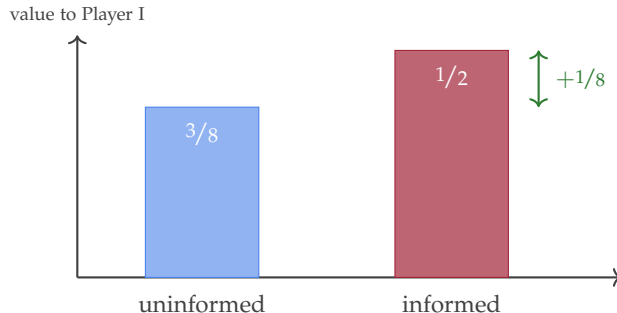
Player I informed

- Player I now has 4 pure strategies: two actions, T and B , at each of the two checkpoints
- **What is an MSNE and value of this game?**
- **MSNE:** $((1(B_1T_2)), (p, 1 - p)), p \in [0, 1]$, **value** = $\frac{1}{2}$
- B_1T_2 **weakly dominates** everything: it is worth $\frac{1}{2}$ whatever Player II does, so Player II has no reply that helps
- Attacker's payoff went from $\frac{3}{8}$ to $\frac{1}{2}$. **The version banner was worth $\frac{1}{8}$.**

Value of Information, Drawn



Value of Information, Drawn



Read this carefully

In a **zero-sum** game, giving one side more information can never hurt that side. Its opponent's loss is exactly its gain, so there is no room for a surprise.

Result on Information Addition in Matrix Games



Theorem

Let Γ be a two-player zero-sum game in extensive form and let Γ' be the game derived from Γ by splitting several information sets of Player I. Then the value of the game Γ' in mixed strategies is greater than or equal to the value of Γ in mixed strategies.

Result on Information Addition in Matrix Games



Theorem

Let Γ be a two-player zero-sum game in extensive form and let Γ' be the game derived from Γ by splitting several information sets of Player I. Then the value of the game Γ' in mixed strategies is greater than or equal to the value of Γ in mixed strategies.

Proof: exercise

Result on Information Addition in Matrix Games



Theorem

Let Γ be a two-player zero-sum game in extensive form and let Γ' be the game derived from Γ by splitting several information sets of Player I. Then the value of the game Γ' in mixed strategies is greater than or equal to the value of Γ in mixed strategies.

Proof: exercise

Proof hint

Every strategy Player I had before is still available after the split – she can simply ignore the extra signal. So her maxmin cannot go down.



Chunk 4

When Information Hurts Everyone

(≈ 20 minutes)

Should You Open Source Your Safety Classifier?



Player II is a platform. Player I is an adversary.

Should You Open Source Your Safety Classifier?



Player II is a platform. Player I is an adversary.

- *M*: ship nothing, live with a modest baseline

Should You Open Source Your Safety Classifier?



Player II is a platform. Player I is an adversary.

- M : ship nothing, live with a modest baseline
- L or R : deploy one of two classifier variants

Should You Open Source Your Safety Classifier?



Player II is a platform. Player I is an adversary.

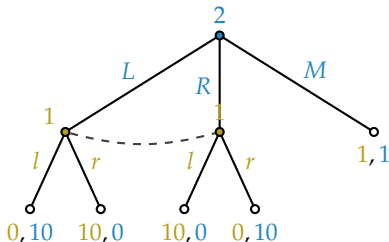
- M : ship nothing, live with a modest baseline
- L or R : deploy one of two classifier variants
- Player I then probes with l or r ; guessing right beats the classifier

Should You Open Source Your Safety Classifier?



Player II is a platform. Player I is an adversary.

- M : ship nothing, live with a modest baseline
- L or R : deploy one of two classifier variants
- Player I then probes with l or r ; guessing right beats the classifier

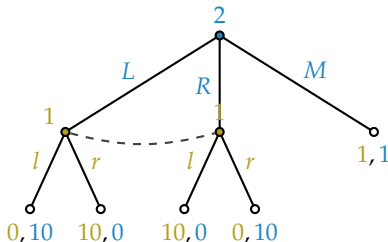


Should You Open Source Your Safety Classifier?



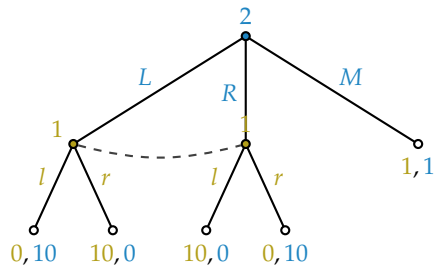
Player II is a platform. Player I is an adversary.

- M : ship nothing, live with a modest baseline
- L or R : deploy one of two classifier variants
- Player I then probes with l or r ; guessing right beats the classifier

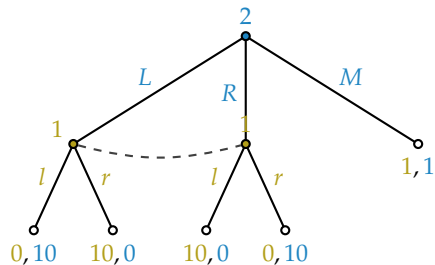


Game A: the variant is kept secret. Find the MSNE!

Game A: Secrecy

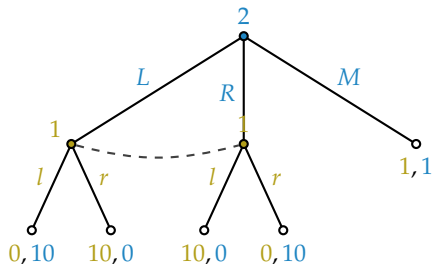


Game A: Secrecy



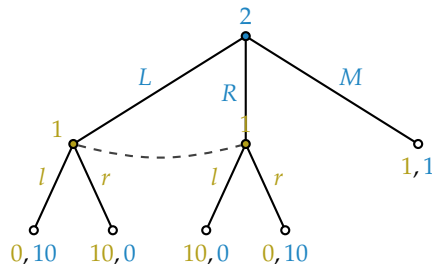
- $\left(\left(\frac{1}{2}(l), \frac{1}{2}(r) \right), \left(\frac{1}{2}(L), \frac{1}{2}(R), 0(M) \right) \right) \implies \text{expected payoff} = (5, 5)$

Game A: Secrecy



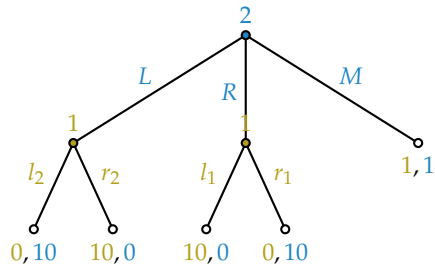
- $\left(\left(\frac{1}{2}(l), \frac{1}{2}(r) \right), \left(\frac{1}{2}(L), \frac{1}{2}(R), 0(M) \right) \right) \implies \text{expected payoff} = (5, 5)$
- The adversary cannot tell L from R , so a coin flip beats him half the time

Game A: Secrecy



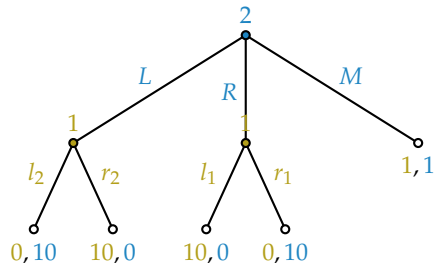
- $\left(\left(\frac{1}{2}(l), \frac{1}{2}(r) \right), \left(\frac{1}{2}(L), \frac{1}{2}(R), 0(M) \right) \right) \implies \text{expected payoff} = (5, 5)$
- The adversary cannot tell L from R , so a coin flip beats him half the time
- $5 > 1$, so the platform **deploys**

Game B: The Variant Is Published



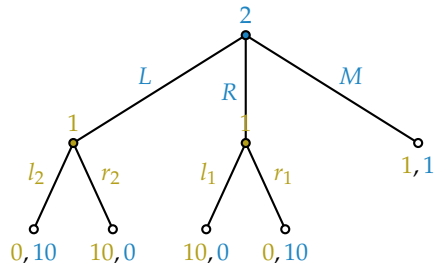
- Find the MSNE of this game!

Game B: The Variant Is Published



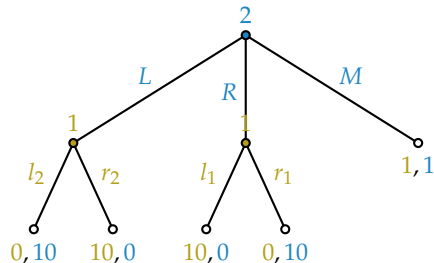
- Find the MSNE of this game!
- $((1(l_1r_2)), (0(L), 0(R), 1(M))) \implies \text{expected payoff} = (1, 1)$

Game B: The Variant Is Published



- Find the MSNE of this game!
- $((1(l_1r_2)), (0(L), 0(R), 1(M))) \implies \text{expected payoff} = (1, 1)$
- The adversary now matches every time, so deploying is worth 0

Game B: The Variant Is Published

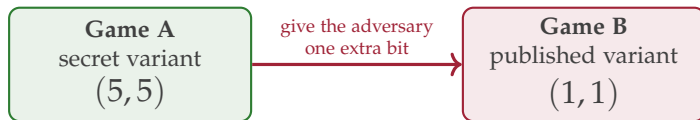


- Find the MSNE of this game!
- $((1(l_1 r_2)), (0(L), 0(R), 1(M))) \implies \text{expected payoff} = (1, 1)$
- The adversary now matches every time, so deploying is worth 0
- The platform retreats to M and **ships nothing**

$(5,5)$ Became $(1,1)$



$(5,5)$ Became $(1,1)$



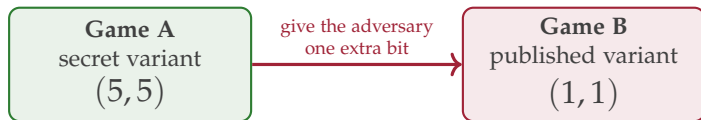
- Player I got **strictly more information** and ended up **worse off**

$(5,5)$ Became $(1,1)$



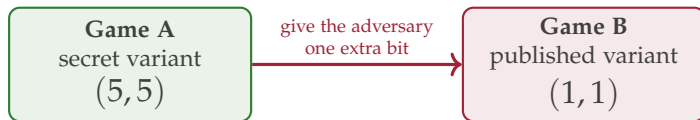
- Player I got **strictly more information** and ended up **worse off**
- So did Player II. Nobody won.

(5,5) Became (1,1)



- Player I got **strictly more information** and ended up **worse off**
- So did Player II. Nobody won.
- The zero-sum theorem does not apply here: this game is general-sum

(5,5) Became (1,1)



- Player I got **strictly more information** and ended up **worse off**
- So did Player II. Nobody won.
- The zero-sum theorem does not apply here: this game is general-sum
- The mechanism: extra information changed **the opponent's behavior**, and that effect swamped the direct gain

(5,5) Became (1,1)



- Player I got **strictly more information** and ended up **worse off**
- So did Player II. Nobody won.
- The zero-sum theorem does not apply here: this game is general-sum
- The mechanism: extra information changed **the opponent's behavior**, and that effect swamped the direct gain

The general lesson

Publishing more information, a model card, a paper, benchmark numbers, changes what a rival optimizes against, exactly like the version banner changed the opponent's best reply above. Before you disclose, ask whose behavior the disclosure will change, and in which direction.



In-Class Quiz 2

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.

Lecture 9: What We Built



- Information set: the histories a player cannot tell apart; the tree's version of an observation

Lecture 9: What We Built



- Information set: the histories a player cannot tell apart; the tree's version of an observation
- IIEFG = PIEFG + a partition I_i per player, with one action menu per cell

Lecture 9: What We Built



- Information set: the histories a player cannot tell apart; the tree's version of an observation
- IIEFG = PIEFG + a partition I_i per player, with one action menu per cell
- Strategies are now maps from **information sets** to actions, not from histories

Lecture 9: What We Built



- Information set: the histories a player cannot tell apart; the tree's version of an observation
- IIEFG = PIEFG + a partition I_i per player, with one action menu per cell
- Strategies are now maps from **information sets** to actions, not from histories
- IIEFG contains both NFG and PIEFG as special cases

Lecture 9: What We Built



- Information set: the histories a player cannot tell apart; the tree's version of an observation
- IIEFG = PIEFG + a partition I_i per player, with one action menu per cell
- Strategies are now maps from **information sets** to actions, not from histories
- IIEFG contains both NFG and PIEFG as special cases
- Zero-sum: splitting your own information sets weakly **raises** your value, $\frac{3}{8} \rightarrow \frac{1}{2}$

Lecture 9: What We Built



- Information set: the histories a player cannot tell apart; the tree's version of an observation
- IIEFG = PIEFG + a partition I_i per player, with one action menu per cell
- Strategies are now maps from **information sets** to actions, not from histories
- IIEFG contains both NFG and PIEFG as special cases
- Zero-sum: splitting your own information sets weakly **raises** your value, $\frac{3}{8} \rightarrow \frac{1}{2}$
- General-sum: more information can make **everyone** worse off, $(5, 5) \rightarrow (1, 1)$

Lecture 9: What We Built



- Information set: the histories a player cannot tell apart; the tree's version of an observation
- IIEFG = PIEFG + a partition I_i per player, with one action menu per cell
- Strategies are now maps from **information sets** to actions, not from histories
- IIEFG contains both NFG and PIEFG as special cases
- Zero-sum: splitting your own information sets weakly **raises** your value, $\frac{3}{8} \rightarrow \frac{1}{2}$
- General-sum: more information can make **everyone** worse off, $(5,5) \rightarrow (1,1)$
- **Next lecture:** if a player cannot see everything, how should she randomize?



End of Lecture 9

See you next class.



► Lecture 9: Imperfect Information Extensive Form Games

► Lecture 10: Behavioral Strategies and Perfect Recall



Chunk 1

Plans vs Policies

(≈ 20 minutes)



- Strategy set of i : $S_i = \prod_{j=1}^{j=k(i)} X(I_i^j)$

Randomized Strategies in IIEFGs



- Strategy set of i : $S_i = \prod_{j=1}^{j=k(i)} X(I_i^j)$
- In NFGs, mixed strategies randomize over pure strategies

Randomized Strategies in IIEFGs



- Strategy set of i : $S_i = \prod_{j=1}^{j=k(i)} X(I_i^j)$
- In NFGs, mixed strategies randomize over pure strategies
- In EFGs, randomization can happen in different ways

Randomized Strategies in IIEFGs



- Strategy set of i : $S_i = \prod_{j=1}^{j=k(i)} X(I_i^j)$
- In NFGs, mixed strategies randomize over pure strategies
- In EFGs, randomization can happen in different ways
 - randomize over the strategies defined at the beginning of the game

Randomized Strategies in IIEFGs



- Strategy set of i : $S_i = \prod_{j=1}^{j=k(i)} X(I_i^j)$
- In NFGs, mixed strategies randomize over pure strategies
- In EFGs, randomization can happen in different ways
 - randomize over the strategies defined at the beginning of the game
 - randomize over the action at an information set: **behavioral strategy**

Randomized Strategies in IIEFGs



- Strategy set of i : $S_i = \prod_{j=1}^{j=k(i)} X(I_i^j)$
- In NFGs, mixed strategies randomize over pure strategies
- In EFGs, randomization can happen in different ways
 - randomize over the strategies defined at the beginning of the game
 - randomize over the action at an information set: **behavioral strategy**

The distinction in one line

A **mixed** strategy draws a complete plan once, before the game, and then obeys it. A **behavioral** strategy carries a coin to every decision point and flips it there.

Two Ways to Be Random



Mixed: sample the plan once

plan library: $L_1L_2, L_1R_2, R_1L_2, R_1R_2$

↓ one draw

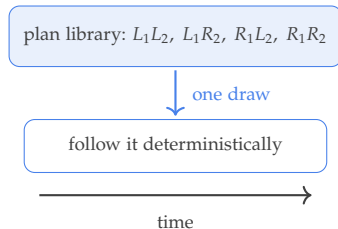
follow it deterministically

→
time

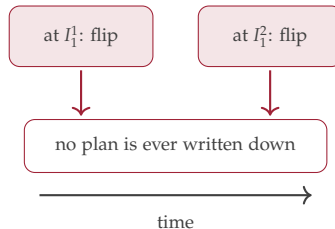
Two Ways to Be Random



Mixed: sample the plan once



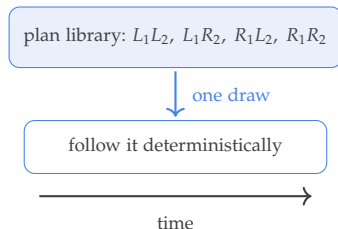
Behavioral: a coin at each node



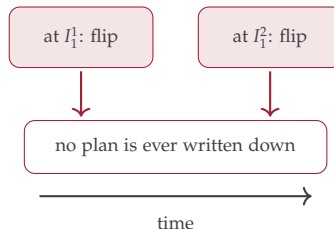


Two Ways to Be Random

Mixed: sample the plan once



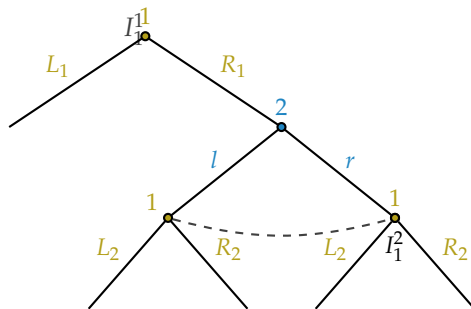
Behavioral: a coin at each node



If you have trained an agent

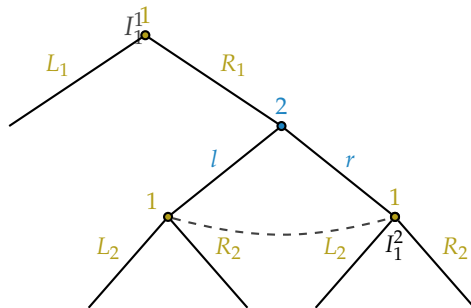
The behavioral strategy is your **stochastic policy** $\pi(a | o)$. The mixed strategy is a distribution over **deterministic policies**, sampled at episode start and frozen. They feel interchangeable. Today we find out exactly when they are.

The Tree We Will Use All Lecture



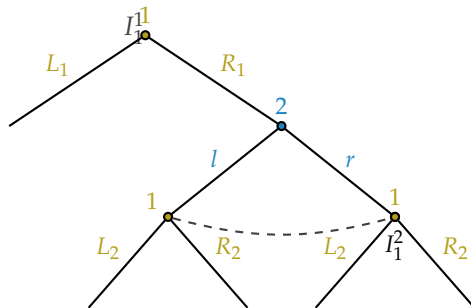
- Strategies of Player 1?

The Tree We Will Use All Lecture



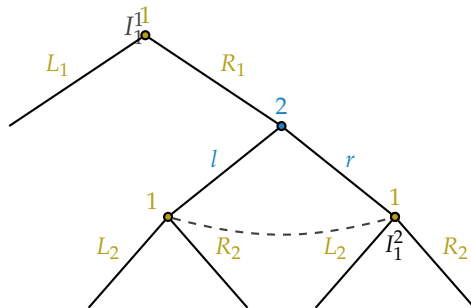
- Strategies of Player 1?
- Pure Strategies (L_1L_2) , (L_1R_2) , (R_1L_2) , (R_1R_2) .

The Tree We Will Use All Lecture



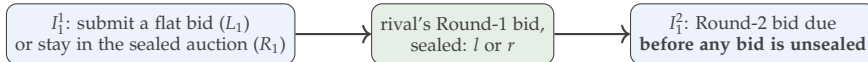
- Strategies of Player 1?
- Pure Strategies (L_1L_2) , (L_1R_2) , (R_1L_2) , (R_1R_2) .
- Mixed Strategy σ_1 , $\sigma_1(L_1L_2)$, $\sigma(L_1R_2)$, $\sigma(R_1L_2)$, $\sigma(R_1R_2)$.

The Tree We Will Use All Lecture

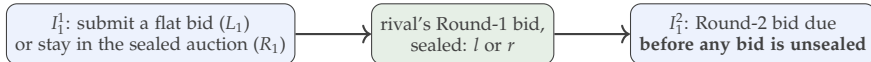


- Strategies of Player 1?
- Pure Strategies (L_1L_2) , (L_1R_2) , (R_1L_2) , (R_1R_2) .
- Mixed Strategy σ_1 , $\sigma_1(L_1L_2)$, $\sigma_1(L_1R_2)$, $\sigma_1(R_1L_2)$, $\sigma_1(R_1R_2)$.
- Behavioral Strategy b_1 , $b_1(I_1^1) \in \Delta(L_1, R_1)$, $b_1(I_1^2) \in \Delta(L_2, R_2)$, $b_2(I_1^1) \in \Delta(l, r)$

Reading the Tree as an Agent Trajectory

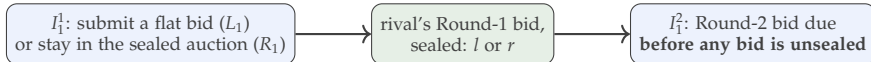


Reading the Tree as an Agent Trajectory



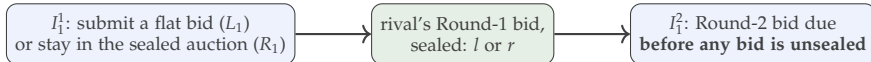
- This is a two-round sealed-bid procurement auction: by rule, every Round-2 bid is due before the auctioneer opens any Round-1 bid, so the bidder truly cannot have seen the rival's move

Reading the Tree as an Agent Trajectory



- This is a two-round sealed-bid procurement auction: by rule, every Round-2 bid is due before the auctioneer opens any Round-1 bid, so the bidder truly cannot have seen the rival's move
- A mixed strategy fixes both bids before Round 1 even opens

Reading the Tree as an Agent Trajectory



- This is a two-round sealed-bid procurement auction: by rule, every Round-2 bid is due before the auctioneer opens any Round-1 bid, so the bidder truly cannot have seen the rival's move
- A mixed strategy fixes both bids before Round 1 even opens
- A behavioral strategy decides afresh at each of I_1^1 and I_1^2



Definition

A **behavioral strategy** of a player in an IIEFG is a function that maps each of her information sets to a probability distribution over the set of possible actions **at that information set**.



Definition

A **behavioral strategy** of a player in an IIEFG is a function that maps each of her information sets to a probability distribution over the set of possible actions **at that information set**.

Question

Why would anyone prefer this to a mixed strategy?



Definition

A **behavioral strategy** of a player in an IIEFG is a function that maps each of her information sets to a probability distribution over the set of possible actions **at that information set**.

Question

Why would anyone prefer this to a mixed strategy?

Answer

Because it is smaller, and because it is what you can actually implement. We make that precise in Chunk 3.

Mixed and Behavioral strategy



Question

What is the relation between mixed and behavioral strategies?



Question

What is the relation between mixed and behavioral strategies?

- In this example, MSs live in $\mathbb{R}^4$, BSs live in two $\mathbb{R}^2$ spaces
- Mixed Strategies look a 'richer' or 'larger' concept

Mixed and Behavioral strategy



Question

What is the relation between mixed and behavioral strategies?

- In this example, MSs live in $\mathbb{R}^4$, BSs live in two $\mathbb{R}^2$ spaces
- Mixed Strategies look a 'richer' or 'larger' concept

Question

Can we have an equivalence?

Mixed and Behavioral strategy



Question

What is the relation between mixed and behavioral strategies?

- In this example, MSs live in $\mathbb{R}^4$, BSs live in two $\mathbb{R}^2$ spaces
- Mixed Strategies look a 'richer' or 'larger' concept

Question

Can we have an equivalence?

Equivalence in terms of the probability of reaching a vertex/history x

Mixed and Behavioral strategy



Question

What is the relation between mixed and behavioral strategies?

- In this example, MSs live in $\mathbb{R}^4$, BSs live in two $\mathbb{R}^2$ spaces
- Mixed Strategies look a 'richer' or 'larger' concept

Question

Can we have an equivalence?

Equivalence in terms of the probability of reaching a vertex/history x

- Say $\rho(x; \sigma)$ is the probability of reaching a node x under mixed strategy profile σ
- Similarly, $\rho(x; b)$ is the same for behavioral strategy profile b

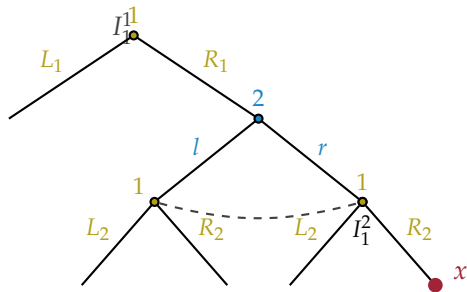


Chunk 2

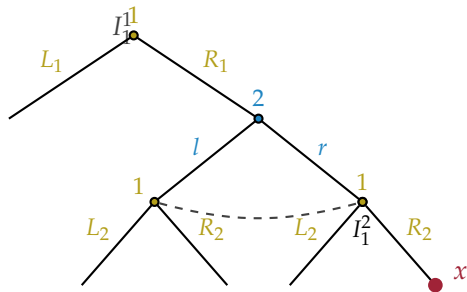
Equivalence of the Two

(≈ 20 minutes)

Computing ρ on Our Tree



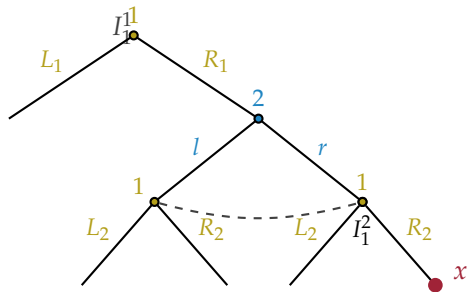
Computing ρ on Our Tree



$$\rho(x; \sigma) = \sigma_1(R_1 R_2) \cdot \sigma_2(r)$$



Computing ρ on Our Tree

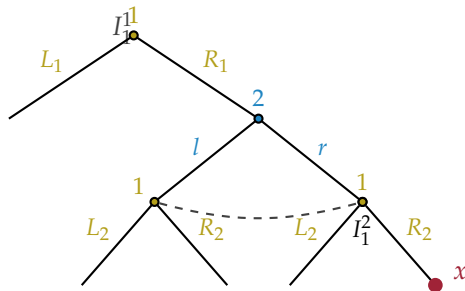


$$\rho(x; \sigma) = \sigma_1(R_1 R_2) \cdot \sigma_2(r)$$

$$\rho(x; b) = b_1(I_1^1)(R_1) \cdot b_2(I_2^1)(r) \cdot b_1(I_1^2)(R_2)$$



Computing ρ on Our Tree



Players can choose different kind of strategies

$$\rho(x; \sigma) = \sigma_1(R_1 R_2) \cdot \sigma_2(r)$$

$$\rho(x; b) = b_1(I_1^1)(R_1) \cdot b_2(I_2^1)(r) \cdot b_1(I_1^2)(R_2)$$

$$\rho(x; \sigma_1, b_2) = \sigma_1(R_1 R_2) \cdot b_2(I_2^1)(r)$$

Equivalence Definition



Definition

A mixed strategy σ_i and a behavioral strategy b_i of a player i in an IIEFG are **equivalent** if for every mixed/behavioral strategy ξ_{-i} of the other players and **every vertex** x in the game tree,

$$\rho(x; \sigma_i, \xi_{-i}) = \rho(x; b_i, \xi_{-i})$$



Equivalence Definition

Definition

A mixed strategy σ_i and a behavioral strategy b_i of a player i in an IIEFG are **equivalent** if for every mixed/behavioral strategy ξ_{-i} of the other players and **every vertex** x in the game tree,

$$\rho(x; \sigma_i, \xi_{-i}) = \rho(x; b_i, \xi_{-i})$$

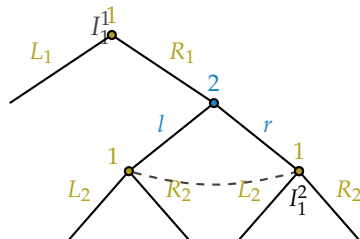
Earlier example (right)

$$b_1(I_1^1)(L_1) = \sigma_1(L_1L_2) + \sigma_1(L_1R_2)$$

$$b_1(I_1^1)(R_1) = \sigma_1(R_1L_2) + \sigma_1(R_1R_2)$$

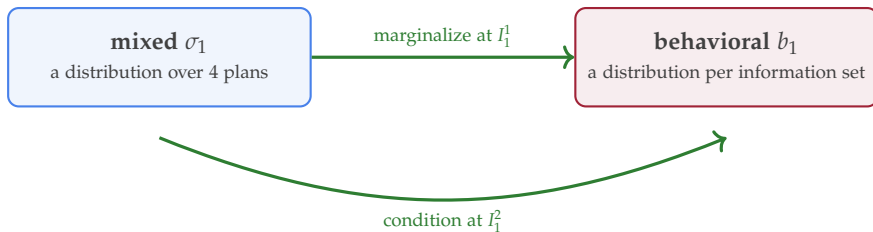
$$b_1(I_1^2)(L_2) = \sigma_1(L_2|R_1)$$

$$b_1(I_1^2)(R_2) = \sigma_1(R_2|R_1)$$

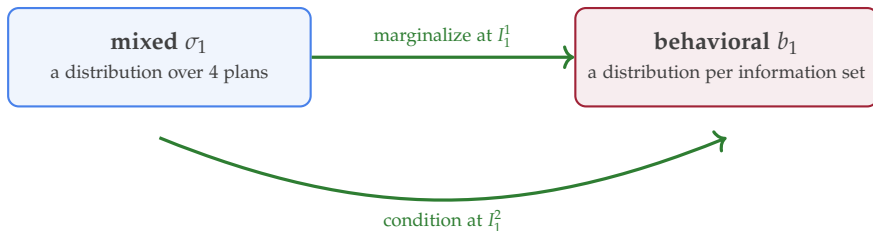


We call b_1 and σ_1 are equivalent.

The Recipe, Said Plainly

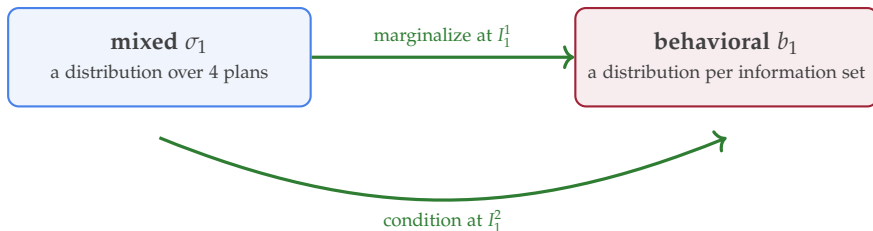


The Recipe, Said Plainly



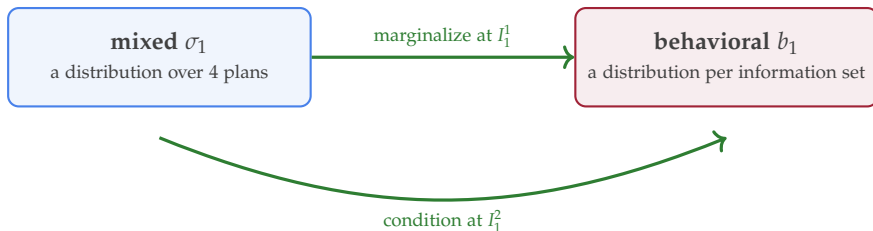
- At the first information set, **add up** the plans that start with L_1

The Recipe, Said Plainly



- At the first information set, **add up** the plans that start with L_1
- At the second, **condition** on having arrived there

The Recipe, Said Plainly



- At the first information set, **add up** the plans that start with L_1
- At the second, **condition** on having arrived there
- Conditioning needs the arrival event to have positive probability – keep that in mind, it is the crack the counterexamples will grow from

More on Equivalent Strategies



The equivalence, by definition, holds at the leaf nodes too

More on Equivalent Strategies



The equivalence, by definition, holds at the leaf nodes too

Claim

It is enough to check the equivalence only at the leaf nodes.

More on Equivalent Strategies



The equivalence, by definition, holds at the leaf nodes too

Claim

It is enough to check the equivalence only at the leaf nodes.

Reason: Pick an arbitrary non-leaf node, the probability of reaching that node is equal to the sum of the probabilities of reaching the leaf nodes in its subtree.

More on Equivalent Strategies



This argument can be extended further

Theorem (Utility Equivalence)

If σ_i and b_i are equivalent, then for every mixed/behavioral strategy vector of the other players ξ_{-i} , the following holds,

$$u_j(\sigma_i, \xi_{-i}) = u_j(b_i, \xi_{-i}), \forall j \in N.$$



More on Equivalent Strategies

This argument can be extended further

Theorem (Utility Equivalence)

If σ_i and b_i are equivalent, then for every mixed/behavioral strategy vector of the other players ξ_{-i} , the following holds,

$$u_j(\sigma_i, \xi_{-i}) = u_j(b_i, \xi_{-i}), \forall j \in N.$$

This theorem is still about one player i swapping representations. We call the whole profiles σ and b **equivalent** when σ_i and b_i are equivalent for *every* $i \in N$; repeat the argument for any such profiles.

Corollary

If σ and b are equivalent, then $u_i(\sigma) = u_i(b)$, $\forall i \in N$.



More on Equivalent Strategies

This argument can be extended further

Theorem (Utility Equivalence)

If σ_i and b_i are equivalent, then for every mixed/behavioral strategy vector of the other players ξ_{-i} , the following holds,

$$u_j(\sigma_i, \xi_{-i}) = u_j(b_i, \xi_{-i}), \forall j \in N.$$

This theorem is still about one player i swapping representations. We call the whole profiles σ and b **equivalent** when σ_i and b_i are equivalent for *every* $i \in N$; repeat the argument for any such profiles.

Corollary

If σ and b are equivalent, then $u_i(\sigma) = u_i(b)$, $\forall i \in N$.

Why it follows: utility is a leaf-probability-weighted sum of payoffs, and the leaf probabilities already agree.



In-Class Quiz 1

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.



Chunk 3

When Equivalence Breaks

(≈ 20 minutes)

Equivalence of strategies in IIEFGs



Question

Why behavioral strategies are desirable?

Answer

- More natural in large IIEFGs

Equivalence of strategies in IIEFGs



Question

Why behavioral strategies are desirable?

Answer

- More natural in large IIEFGs
 - players plan at every stage (information set) of the game rather than a master plan

Equivalence of strategies in IIEFGs



Question

Why behavioral strategies are desirable?

Answer

- More natural in large IIEFGs
 - players plan at every stage (information set) of the game rather than a master plan
- A smaller number of variables to deal with

Equivalence of strategies in IIEFGs



Question

Why behavioral strategies are desirable?

Answer

- More natural in large IIEFGs
 - players plan at every stage (information set) of the game rather than a master plan
- A smaller number of variables to deal with
 - Consider a player having 4 information sets with 2 actions each

Equivalence of strategies in IIEFGs



Question

Why behavioral strategies are desirable?

Answer

- More natural in large IIEFGs
 - players plan at every stage (information set) of the game rather than a master plan
- A smaller number of variables to deal with
 - Consider a player having 4 information sets with 2 actions each
 - needs $(2^4 - 1)$ variables to represent mixed strategies

Equivalence of strategies in IIEFGs



Question

Why behavioral strategies are desirable?

Answer

- More natural in large IIEFGs
 - players plan at every stage (information set) of the game rather than a master plan
- A smaller number of variables to deal with
 - Consider a player having 4 information sets with 2 actions each
 - needs $(2^4 - 1)$ variables to represent mixed strategies
 - needs 4 variables for behavioral strategies

Equivalence of strategies in IIEFGs



Question

Why behavioral strategies are desirable?

Answer

- More natural in large IIEFGs
 - players plan at every stage (information set) of the game rather than a master plan
- A smaller number of variables to deal with
 - Consider a player having 4 information sets with 2 actions each
 - needs $(2^4 - 1)$ variables to represent mixed strategies
 - needs 4 variables for behavioral strategies

Question

Can we construct one from another?

OR

Does equivalence always hold?

15 Numbers or 4?



mixed
 $2^4 - 1 = 15$



behavioral
4

Now imagine 20 information sets.

$2^{20} - 1 \approx 10^6$ against 20.

This is why solvers optimize over behavioral strategies, not mixed ones.

15 Numbers or 4?



mixed
 $2^4 - 1 = 15$



behavioral
4

Now imagine 20 information sets.

$2^{20} - 1 \approx 10^6$ against 20.

This is why solvers optimize over behavioral strategies, not mixed ones.

The real stakes

Heads-up no-limit poker has around 10^{161} histories. Nobody writes down a distribution over plans. CFR and its descendants iterate on **behavioral** strategies, one information set at a time.

- **CFR** = Counterfactual Regret Minimization: track each action's **regret** at every information set, shift probability toward low-regret actions



15 Numbers or 4?



mixed
 $2^4 - 1 = 15$



behavioral
4

Now imagine 20 information sets.

$2^{20} - 1 \approx 10^6$ against 20.

This is why solvers optimize over behavioral strategies, not mixed ones.

The real stakes

Heads-up no-limit poker has around 10^{161} histories. Nobody writes down a distribution over plans. CFR and its descendants iterate on **behavioral** strategies, one information set at a time.

- **CFR** = Counterfactual Regret Minimization: track each action's **regret** at every information set, shift probability toward low-regret actions
- Self-play drives average regret to zero; the average behavioral strategy converges to a Nash equilibrium

15 Numbers or 4?



mixed
 $2^4 - 1 = 15$



behavioral
4

Now imagine 20 information sets.

$2^{20} - 1 \approx 10^6$ against 20.

This is why solvers optimize over behavioral strategies, not mixed ones.

The real stakes

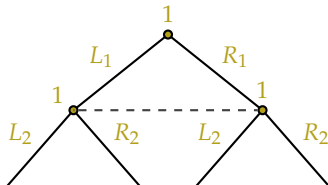
Heads-up no-limit poker has around 10^{161} histories. Nobody writes down a distribution over plans. CFR and its descendants iterate on **behavioral** strategies, one information set at a time.

- **CFR** = Counterfactual Regret Minimization: track each action's **regret** at every information set, shift probability toward low-regret actions
- Self-play drives average regret to zero; the average behavioral strategy converges to a Nash equilibrium
- Powers Libratus and Pluribus, the first bots to beat professional humans at heads-up and multi-player no-limit poker

Equivalence of strategies in IIEFGs (Example 1)



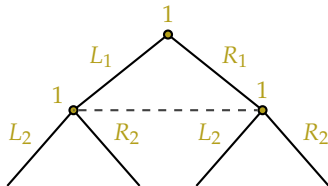
Player remembers that it made a move but forgets which move



Equivalence of strategies in IIEFGs (Example 1)



Player remembers that it made a move but forgets which move

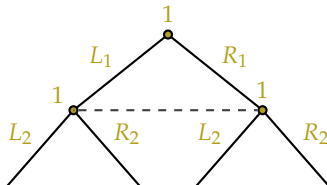


- consider mixed strategy $\sigma_1(L_1L_2), \sigma_1(L_1R_2), \sigma_1(R_1L_2), \sigma_1(R_1R_2)$



Equivalence of strategies in IIEFGs (Example 1)

Player remembers that it made a move but forgets which move

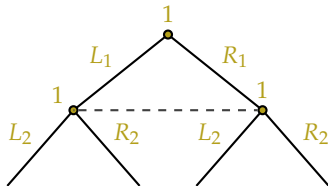


- consider mixed strategy $\sigma_1(L_1L_2), \sigma_1(L_1R_2), \sigma_1(R_1L_2), \sigma_1(R_1R_2)$
- behavioral strategy $b_1(L_1), b_1(R_1), b_1(L_2), b_1(R_2)$.



Equivalence of strategies in IIEFGs (Example 1)

Player remembers that it made a move but forgets which move

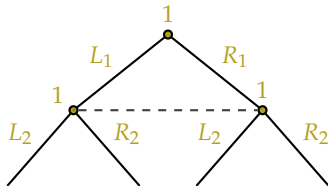


- consider mixed strategy $\sigma_1(L_1L_2), \sigma_1(L_1R_2), \sigma_1(R_1L_2), \sigma_1(R_1R_2)$
- behavioral strategy $b_1(L_1), b_1(R_1), b_1(L_2), b_1(R_2)$.
- mixed strategy has more control over profiles, e.g., $\sigma_1(L_1R_2) = \sigma_1(R_1L_2) = 0$



Equivalence of strategies in IIEFGs (Example 1)

Player remembers that it made a move but forgets which move

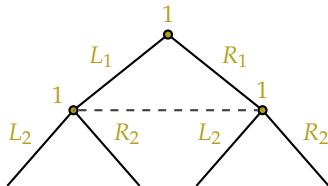


- consider mixed strategy $\sigma_1(L_1L_2), \sigma_1(L_1R_2), \sigma_1(R_1L_2), \sigma_1(R_1R_2)$
- behavioral strategy $b_1(L_1), b_1(R_1), b_1(L_2), b_1(R_2)$.
- mixed strategy has more control over profiles, e.g., $\sigma_1(L_1R_2) = \sigma_1(R_1L_2) = 0$
- not possible in behavioral strategies



Equivalence of strategies in IIEFGs (Example 1)

Player remembers that it made a move but forgets which move



- consider mixed strategy $\sigma_1(L_1L_2), \sigma_1(L_1R_2), \sigma_1(R_1L_2), \sigma_1(R_1R_2)$
- behavioral strategy $b_1(L_1), b_1(R_1), b_1(L_2), b_1(R_2)$.
- mixed strategy has more control over profiles, e.g., $\sigma_1(L_1R_2) = \sigma_1(R_1L_2) = 0$
- not possible in behavioral strategies
- **Mixed strategy with no equivalent behavioral strategies**

Example 1, In Practice



The agent whose context got compacted

Your agent takes a step, then the conversation is summarized and the raw step is dropped. At the next call it knows *that* it acted, not *what* it did. This is a **behavioral strategy** in the wild: it decides afresh at each information set, with no memory of the earlier draw.

Example 1, In Practice



The agent whose context got compacted

Your agent takes a step, then the conversation is summarized and the raw step is dropped. At the next call it knows *that* it acted, not *what* it did. This is a **behavioral strategy** in the wild: it decides afresh at each information set, with no memory of the earlier draw.

- The mixed strategy " L_1L_2 or R_1R_2 , each with probability $\frac{1}{2}$ " is a **correlated** plan: be consistent, whichever way you go



Example 1, In Practice

The agent whose context got compacted

Your agent takes a step, then the conversation is summarized and the raw step is dropped. At the next call it knows *that* it acted, not *what* it did. This is a **behavioral strategy** in the wild: it decides afresh at each information set, with no memory of the earlier draw.

- The mixed strategy " L_1L_2 or R_1R_2 , each with probability $\frac{1}{2}$ " is a **correlated** plan: be consistent, whichever way you go
- A behavioral strategy cannot correlate the two moves – it flips independently at each information set, so L_1R_2 always gets positive weight if both L_1 and R_2 do



Example 1, In Practice

The agent whose context got compacted

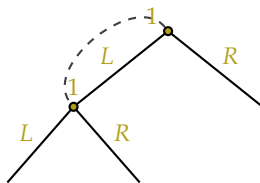
Your agent takes a step, then the conversation is summarized and the raw step is dropped. At the next call it knows *that* it acted, not *what* it did. This is a **behavioral strategy** in the wild: it decides afresh at each information set, with no memory of the earlier draw.

- The mixed strategy " L_1L_2 or R_1R_2 , each with probability $\frac{1}{2}$ " is a **correlated** plan: be consistent, whichever way you go
- A behavioral strategy cannot correlate the two moves – it flips independently at each information set, so L_1R_2 always gets positive weight if both L_1 and R_2 do
- The correlation lived in the plan, and forgetting destroyed the only place it could be stored

Equivalence of strategies in IIEFGs (Example 2)



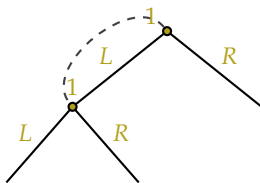
Player forgets whether it made a move or not



Equivalence of strategies in IIEFGs (Example 2)



Player forgets whether it made a move or not

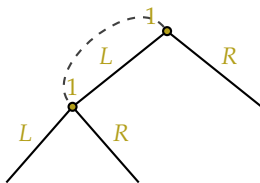


Behavioral strategy: positive mass on the second leaf node possible, not for mixed strategy

Equivalence of strategies in IIEFGs (Example 2)



Player forgets whether it made a move or not



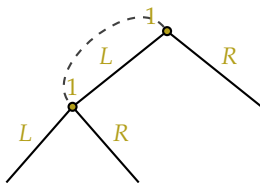
Behavioral strategy: positive mass on the second leaf node possible, not for mixed strategy

A behavioral strategy with no equivalent mixed strategy



Equivalence of strategies in IIEFGs (Example 2)

Player forgets whether it made a move or not



Behavioral strategy: positive mass on the second leaf node possible, not for mixed strategy

A behavioral strategy with no equivalent mixed strategy

Answer

The equivalence does not hold if the players are forgetful

Example 2, In Practice



The retry with no idempotency key

The client times out and retries. It does not know whether the first request already went through. Root and child sit in the *same* information set.

Example 2, In Practice



The retry with no idempotency key

The client times out and retries. It does not know whether the first request already went through. Root and child sit in the *same* information set.

- A pure strategy picks one action for the whole information set, so it plays L at the root *and* L at the child, or R at both

Example 2, In Practice



The retry with no idempotency key

The client times out and retries. It does not know whether the first request already went through. Root and child sit in the *same* information set.

- A pure strategy picks one action for the whole information set, so it plays L at the root *and* L at the child, or R at both
- Then the path L followed by R is **unreachable** under every pure strategy, hence under every mixed strategy

Example 2, In Practice



The retry with no idempotency key

The client times out and retries. It does not know whether the first request already went through. Root and child sit in the *same* information set.

- A pure strategy picks one action for the whole information set, so it plays L at the root *and* L at the child, or R at both
- Then the path L followed by R is **unreachable** under every pure strategy, hence under every mixed strategy
- A behavioral strategy flips a fresh coin at each visit, so it reaches that leaf with positive probability

Example 2, In Practice



The retry with no idempotency key

The client times out and retries. It does not know whether the first request already went through. Root and child sit in the *same* information set.

- A pure strategy picks one action for the whole information set, so it plays L at the root *and* L at the child, or R at both
- Then the path L followed by R is **unreachable** under every pure strategy, hence under every mixed strategy
- A behavioral strategy flips a fresh coin at each visit, so it reaches that leaf with positive probability
- Two visits to one information set is the whole disease

Equivalence of strategies in IIEFGs



Question

When does behavioral strategy have no equivalent mixed strategy?

Equivalence of strategies in IIEFGs



Question

When does behavioral strategy have no equivalent mixed strategy?

Observation from a graph viewpoint

- 1 Let x be a non-root node

Equivalence of strategies in IIEFGs



Question

When does behavioral strategy have no equivalent mixed strategy?

Observation from a graph viewpoint

- 1 Let x be a non-root node
- 2 action at x_1 leading to x : the unique edge emanating from x_1 that is on the path from root to x

Equivalence of strategies in IIEFGs



Question

When does behavioral strategy have no equivalent mixed strategy?

Observation from a graph viewpoint

- 1 Let x be a non-root node
- 2 action at x_1 leading to x : the unique edge emanating from x_1 that is on the path from root to x
- 3 In the second example, there is a node that has a path from the root that crosses the same information set twice

Equivalence of strategies in IIEFGs



Question

When does behavioral strategy have no equivalent mixed strategy?

Observation from a graph viewpoint

- 1 Let x be a non-root node
- 2 action at x_1 leading to x : the unique edge emanating from x_1 that is on the path from root to x
- 3 In the second example, there is a node that has a path from the root that crosses the same information set twice
- 4 If the path from the root to x passes through vertices x_1 and x'_1 that are in the same information set of player i , and the action leading to x at x_1 and x'_1 is different, then no **pure strategy** can ever lead to x



Equivalence of strategies in IIEFGs

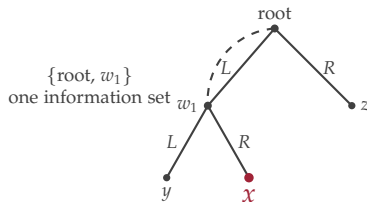
Question

When does behavioral strategy have no equivalent mixed strategy?

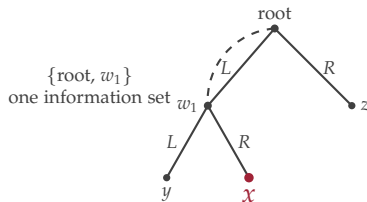
Observation from a graph viewpoint

- 1 Let x be a non-root node
- 2 action at x_1 leading to x : the unique edge emanating from x_1 that is on the path from root to x
- 3 In the second example, there is a node that has a path from the root that crosses the same information set twice
- 4 If the path from the root to x passes through vertices x_1 and x'_1 that are in the same information set of player i , and the action leading to x at x_1 and x'_1 is different, then no **pure strategy** can ever lead to x
- 5 Since mixed strategy is a randomization over pure strategies, every mixed strategy will put zero probability mass on x but behavioral strategy randomizes on every vertex **independently**, hence x may be reached in behavioral strategies with a positive probability

Seeing Why x Has No Pure-Strategy Reach

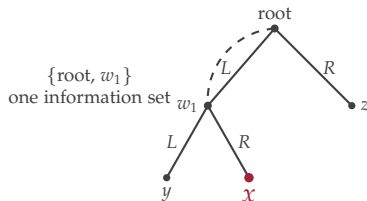


Seeing Why x Has No Pure-Strategy Reach



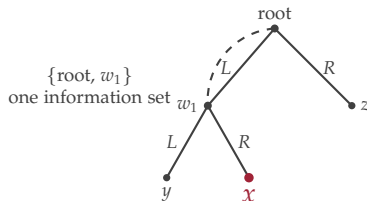
- Example 2 again. The dashed curve is the information set: at w_1 , player 1 cannot tell whether she already moved once (L) or is still at the root

Seeing Why x Has No Pure-Strategy Reach



- Example 2 again. The dashed curve is the information set: at w_1 , player 1 cannot tell whether she already moved once (L) or is still at the root
- **Pure strategy:** one fixed action for the *whole* information set – play L throughout (reaching y) or R throughout (reaching z). Reaching x needs L then R : never chosen

Seeing Why x Has No Pure-Strategy Reach



- Example 2 again. The dashed curve is the information set: at w_1 , player 1 cannot tell whether she already moved once (L) or is still at the root
- **Pure strategy:** one fixed action for the *whole* information set – play L throughout (reaching y) or R throughout (reaching z). Reaching x needs L then R : never chosen
- **Behavioral strategy:** a fresh coin at *each visit* – L then R reaches x with positive probability

Equivalence of strategies in IIEFGs



The last observation can be stated as a lemma

Equivalence of strategies in IIEFGs



The last observation can be stated as a lemma

Lemma

*If there exists a path from the root to some vertex x that passes through the same information set at least twice, and if the action leading to x is **not** the same at each of those vertices, then the player at the information set has a behavioral strategy that has no equivalent mixed strategy.*

Equivalence of strategies in IIEFGs



The last observation can be stated as a lemma

Lemma

*If there exists a path from the root to some vertex x that passes through the same information set at least twice, and if the action leading to x is **not** the same at each of those vertices, then the player at the information set has a behavioral strategy that has no equivalent mixed strategy.*

This lemma helps in proving the following characterization result of equivalence.

Equivalence of strategies in IIEFGs



The last observation can be stated as a lemma

Lemma

*If there exists a path from the root to some vertex x that passes through the same information set at least twice, and if the action leading to x is **not** the same at each of those vertices, then the player at the information set has a behavioral strategy that has no equivalent mixed strategy.*

This lemma helps in proving the following characterization result of equivalence.

Theorem (6.11 of MSZ)

Consider an IIEFG such that every vertex has at least two actions. Every behavioral strategy has an equivalent mixed strategy for a player iff each information set of that player intersects every path emanating from the root at most once.

Equivalence of strategies in IIEFGs



The last observation can be stated as a lemma

Lemma

*If there exists a path from the root to some vertex x that passes through the same information set at least twice, and if the action leading to x is **not** the same at each of those vertices, then the player at the information set has a behavioral strategy that has no equivalent mixed strategy.*

This lemma helps in proving the following characterization result of equivalence.

Theorem (6.11 of MSZ)

Consider an IIEFG such that every vertex has at least two actions. Every behavioral strategy has an equivalent mixed strategy for a player iff each information set of that player intersects every path emanating from the root at most once.

Proof.

Homework. Reading exercise from MSZ.





Chunk 4

Perfect Recall and Kuhn's Theorem

(≈ 25 minutes)

Behavioral Strategy equivalent to Mixed Strategy



Theorem (6.11 of MSZ)

Consider an IIEFG such that every vertex has at least two actions. Every behavioral strategy has an equivalent mixed strategy for a player iff each information set of that player intersects every path emanating from the root at most once.

Behavioral Strategy equivalent to Mixed Strategy



Theorem (6.11 of MSZ)

Consider an IIEFG such that every vertex has at least two actions. Every behavioral strategy has an equivalent mixed strategy for a player iff each information set of that player intersects every path emanating from the root at most once.

To formalize (i.e., to set the conditions when the equivalence holds), we need to formalize the **forgetfulness** of a player.

- saw few examples of players' forgetfulness.
- our conditions need to ensure that none of the previous types of forgetfulness happens.

Behavioral Strategy equivalent to Mixed Strategy



Definition (Choice of **same action at an information set**)

Let $X = (x^0, x^1, \dots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \dots, \hat{x}^L)$ be two paths in the game tree.

Behavioral Strategy equivalent to Mixed Strategy



Definition (Choice of **same action at an information set**)

Let $X = (x^0, x^1, \dots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \dots, \hat{x}^L)$ be two paths in the game tree. Let I_i^j be an information set of player i that intersects these two paths only at one vertex, say x_k and x_l respectively.

Behavioral Strategy equivalent to Mixed Strategy



Definition (Choice of **same action at an information set**)

Let $X = (x^0, x^1, \dots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \dots, \hat{x}^L)$ be two paths in the game tree. Let I_i^j be an information set of player i that intersects these two paths only at one vertex, say x_k and x_l respectively.

These two paths **choose the same action at information set I_i^j** if

Behavioral Strategy equivalent to Mixed Strategy



Definition (Choice of **same action at an information set**)

Let $X = (x^0, x^1, \dots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \dots, \hat{x}^L)$ be two paths in the game tree. Let I_i^j be an information set of player i that intersects these two paths only at one vertex, say x_k and x_l respectively.

These two paths **choose the same action at information set I_i^j** if

- $k < K$ and $l < L$



Behavioral Strategy equivalent to Mixed Strategy

Definition (Choice of **same action at an information set**)

Let $X = (x^0, x^1, \dots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \dots, \hat{x}^L)$ be two paths in the game tree. Let I_i^j be an information set of player i that intersects these two paths only at one vertex, say x_k and x_l respectively.

These two paths **choose the same action at information set I_i^j** if

- $k < K$ and $l < L$
- actions x^k leading to x^{k+1} and $\hat{x}^l$ leading to $\hat{x}^{l+1}$ are identical, and are denoted by $a_i(x^k \rightarrow x^{k+1}) = a_i(\hat{x}^l \rightarrow \hat{x}^{l+1})$



Behavioral Strategy equivalent to Mixed Strategy

Definition (Choice of **same action at an information set**)

Let $X = (x^0, x^1, \dots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \dots, \hat{x}^L)$ be two paths in the game tree. Let I_i^j be an information set of player i that intersects these two paths only at one vertex, say x_k and x_l respectively.

These two paths **choose the same action at information set I_i^j** if

- $k < K$ and $l < L$
- actions x^k leading to x^{k+1} and $\hat{x}^l$ leading to $\hat{x}^{l+1}$ are identical, and are denoted by $a_i(x^k \rightarrow x^{k+1}) = a_i(\hat{x}^l \rightarrow \hat{x}^{l+1})$

'Leading to' may not be a relation between parent and child nodes, it can be any descendant of the former since the path is unique in a tree.



Behavioral Strategy equivalent to Mixed Strategy

Definition (Choice of **same action at an information set**)

Let $X = (x^0, x^1, \dots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \dots, \hat{x}^L)$ be two paths in the game tree. Let I_i^j be an information set of player i that intersects these two paths only at one vertex, say x_k and x_l respectively.

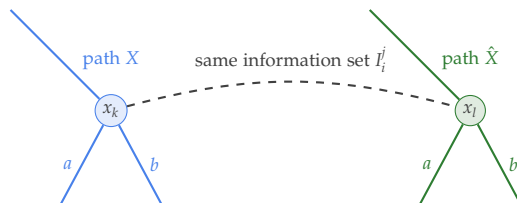
These two paths **choose the same action at information set I_i^j** if

- $k < K$ and $l < L$
- actions x^k leading to x^{k+1} and $\hat{x}^l$ leading to $\hat{x}^{l+1}$ are identical, and are denoted by $a_i(x^k \rightarrow x^{k+1}) = a_i(\hat{x}^l \rightarrow \hat{x}^{l+1})$

'Leading to' may not be a relation between parent and child nodes, it can be any descendant of the former since the path is unique in a tree.

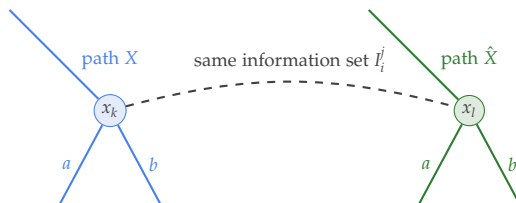
We may use it for a path leading from a node x to some node y which is not an immediate child of x .

The 'Same Action' Definition, Drawn





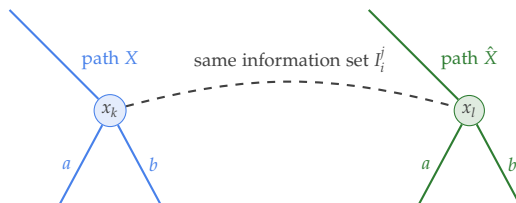
The 'Same Action' Definition, Drawn



- x_k (on path X) and x_l (on path $\hat{X}$) are the one vertex where each path meets I_i^j – same information set, so the same action menu $\{a, b\}$ is on offer at both



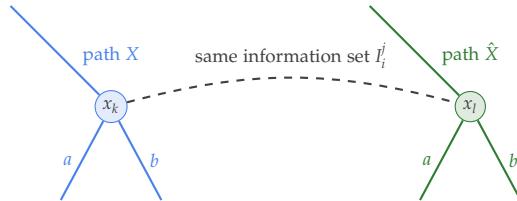
The 'Same Action' Definition, Drawn



- x_k (on path X) and x_l (on path $\hat{X}$) are the one vertex where each path meets I_i^j – same information set, so the same action menu $\{a, b\}$ is on offer at both
- If X continues with a from x_k and $\hat{X}$ continues with a from x_l : the two paths **choose the same action** at I_i^j



The 'Same Action' Definition, Drawn



- x_k (on path X) and x_l (on path $\hat{X}$) are the one vertex where each path meets I_i^j – same information set, so the same action menu $\{a, b\}$ is on offer at both
- If X continues with a from x_k and $\hat{X}$ continues with a from x_l : the two paths **choose the same action** at I_i^j
- If X continues with a but $\hat{X}$ continues with b : they choose **different** actions – this is exactly what perfect recall's second clause will rule out

Games with Perfect Recall



Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied



Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once.**



Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two paths that end in the same information set of player i pass through **the same information sets of i in the same order**



Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two paths that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.



Games with Perfect Recall

Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two paths that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Rephrasing

For every I_i^j of player i and every pair of vertices $x, y \in I_i^j$, if the decision vertices of i are $x_i^1, x_i^2, \dots, x_i^{L_i} = x$ and $y_i^1, y_i^2, \dots, y_i^{L_i'} = y$ respectively for the two paths from the root to x and y , then



Games with Perfect Recall

Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two path that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Rephrasing

For every I_i^j of player i and every pair of vertices $x, y \in I_i^j$, if the decision vertices of i are $x_i^1, x_i^2, \dots, x_i^L = x$ and $y_i^1, y_i^2, \dots, y_i^{L'} = y$ respectively for the two paths from the root to x and y , then

- 1 $L = L'$,



Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two path that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Rephrasing

For every I_i^j of player i and every pair of vertices $x, y \in I_i^j$, if the decision vertices of i are $x_i^1, x_i^2, \dots, x_i^L = x$ and $y_i^1, y_i^2, \dots, y_i^{L'} = y$ respectively for the two paths from the root to x and y , then

- 1 $L = L'$,
- 2 $x_i^l, y_i^l \in I_i^k$ for some k ,



Games with Perfect Recall

Definition (Perfect Recall)

Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two path that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Rephrasing

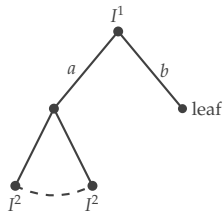
For every I_i^j of player i and every pair of vertices $x, y \in I_i^j$, if the decision vertices of i are $x_i^1, x_i^2, \dots, x_i^L = x$ and $y_i^1, y_i^2, \dots, y_i^{L'} = y$ respectively for the two paths from the root to x and y , then

- 1 $L = L'$,
- 2 $x_i^l, y_i^l \in I_i^k$ for some k ,
- 3 $a_i(x_i^l \rightarrow x_i^{l+1}) = a_i(y_i^l \rightarrow y_i^{l+1}), \forall l = 1, 2, \dots, L - 1$.

Clause 1, Drawn: Does a Path Meet an Info Set Twice?

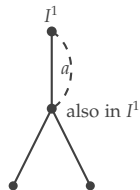


Clause 1 holds



every root-to-leaf path
meets I^2 at most once

Clause 1 fails

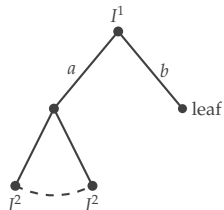


the same path visits
 I^1 twice



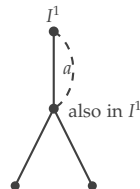
Clause 1, Drawn: Does a Path Meet an Info Set Twice?

Clause 1 holds



every root-to-leaf path
meets I^2 at most once

Clause 1 fails



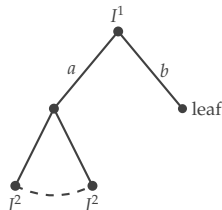
the same path visits
 I^1 twice

- Left: exactly the tree we have used all lecture – I^2 is reached only through R_1 , once



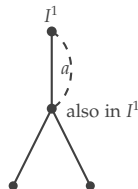
Clause 1, Drawn: Does a Path Meet an Info Set Twice?

Clause 1 holds



every root-to-leaf path
meets I^2 at most once

Clause 1 fails



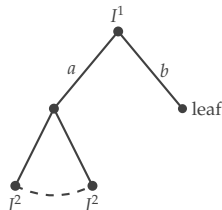
the same path visits
 I^1 twice

- Left: exactly the tree we have used all lecture – I^2 is reached only through R_1 , once
- Right: exactly Example 2 from Chunk 3 – the root and its child sit in the *same* information set, so one path passes through I^1 twice



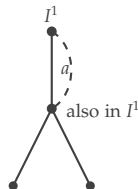
Clause 1, Drawn: Does a Path Meet an Info Set Twice?

Clause 1 holds



every root-to-leaf path
meets I^2 at most once

Clause 1 fails



the same path visits
 I^1 twice

- Left: exactly the tree we have used all lecture – I^2 is reached only through R_1 , once
- Right: exactly Example 2 from Chunk 3 – the root and its child sit in the *same* information set, so one path passes through I^1 twice
- Clause 1 is what makes “did I already move?” always answerable

Perfect Recall Is Just “No Amnesia”



Clause 1 bans

“did I already move?”

no path meets one information
set twice

Clause 2 bans

“what did I play?”

same history of sets, same ac-
tions taken

Example 2
broke clause 1.

Example 1
broke clause 2.

Perfect Recall Is Just “No Amnesia”



Clause 1 bans

“did I already move?”

no path meets one information
set twice

Clause 2 bans

“what did I play?”

same history of sets, same ac-
tions taken

Example 2
broke clause 1.

Example 1
broke clause 2.

- A player with perfect recall never forgets her own past, though she may never have learned the opponent's

Perfect Recall Is Just “No Amnesia”



Clause 1 bans

“did I already move?”

no path meets one information
set twice

Clause 2 bans

“what did I play?”

same history of sets, same ac-
tions taken

Example 2
broke clause 1.

Example 1
broke clause 2.

- A player with perfect recall never forgets her own past, though she may never have learned the opponent's
- Perfect recall is a condition on **memory**, not on **observation**

Perfect Recall Is Just “No Amnesia”



Clause 1 bans

“did I already move?”

no path meets one information
set twice

Clause 2 bans

“what did I play?”

same history of sets, same ac-
tions taken

Example 2
broke clause 1.

Example 1
broke clause 2.

- A player with perfect recall never forgets her own past, though she may never have learned the opponent's
- Perfect recall is a condition on **memory**, not on **observation**
- Poker has perfect recall: you always remember your own bets, you just never see the hole cards



Definition

A game has **perfect recall** if every player has a perfect recall.

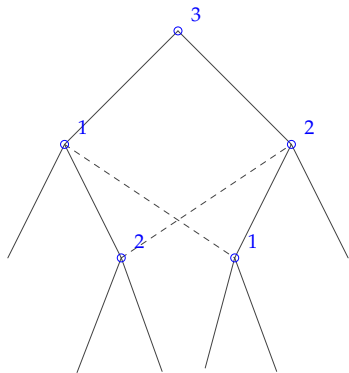


Definition

A game has **perfect recall** if every player has a perfect recall.

Note: Definition of perfect recall subsumes the condition where every behavioral strategy has equivalent mixed strategy

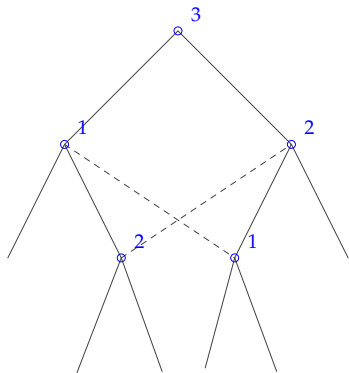
Example 1



Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two paths that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Example 1

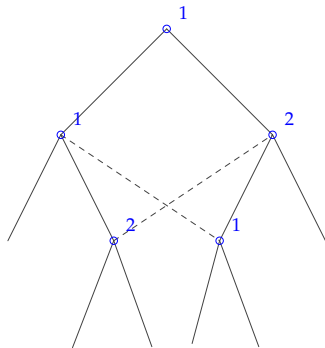


Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two paths that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Game with Perfect Recall: This example satisfies the conditions of the definitions.

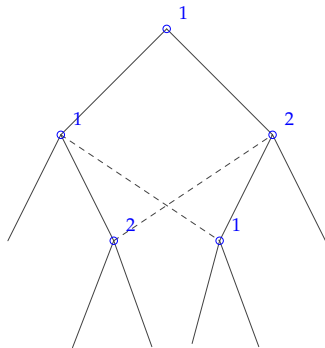
Example 2



Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two path that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Example 2



Player i has **perfect recall** if the following conditions are satisfied

- 1 Every information set of player i intersects **every path from the root to a leaf at most once**.
- 2 Every two path that end in the same information set of player i pass through **the same information sets of i in the same order** and in every such information set **the two paths choose the same action**.

Game with Imperfect Recall: Player 1 takes two different actions at the first information set to reach two different vertices of the second information set.

Implications of Perfect Recall



Let $S_i^*(x)$ be the set of pure strategies of player i at which he chooses actions leading to x , i.e., intersections of members of S_i with the path from root to x .

Theorem

If i is a player with perfect recall and x and x' are the two vertices in the same information set of i , then $S_i^(x) = S_i^*(x')$.*

The above conclusion comes from the same sequence of information sets and same actions. The next implication of mixed and behavioral strategies.



Implications of Perfect Recall

Let $S_i^*(x)$ be the set of pure strategies of player i at which he chooses actions leading to x , i.e., intersections of members of S_i with the path from root to x .

Theorem

If i is a player with perfect recall and x and x' are the two vertices in the same information set of i , then $S_i^(x) = S_i^*(x')$.*

The above conclusion comes from the same sequence of information sets and same actions. The next implication of mixed and behavioral strategies.

Why this is the key lemma

If $S_i^*(x) = S_i^*(x')$, then arriving at the information set is an event that does not depend on *which* node in it you are at. That is exactly what makes the conditioning step of the construction legal.



Theorem (Kuhn 1957)

In every IIEFG, if i is a player with perfect recall, then for every mixed strategy of i , there exists an equivalent behavioral strategy.



Theorem (Kuhn 1957)

In every IIEFG, if i is a player with perfect recall, then for every mixed strategy of i , there exists an equivalent behavioral strategy.

- The converse is already true since the sufficient condition for that is already subsumed in the definition of perfect recall.



Theorem (Kuhn 1957)

In every IIEFG, if i is a player with perfect recall, then for every mixed strategy of i , there exists an equivalent behavioral strategy.

- The converse is already true since the sufficient condition for that is already subsumed in the definition of perfect recall.
- Proof left as reading exercise (MSZ Theorem 6.15)



Theorem (Kuhn 1957)

In every IIEFG, if i is a player with perfect recall, then for every mixed strategy of i , there exists an equivalent behavioral strategy.

- The converse is already true since the sufficient condition for that is already subsumed in the definition of perfect recall.
- Proof left as reading exercise (MSZ Theorem 6.15)
- The proof is constructive. It starts with the mixed strategy and constructs the behavioral strategies such that the probabilities of reaching a leaf are same. The arguments show that such a construction is always possible because of perfect recall.

Kuhn's Theorem Is a License



So you may optimize over the small object and still claim results about the big one.

Kuhn's Theorem Is a License



So you may optimize over the small object and still claim results about the big one.

- Every theorem you proved about mixed strategies – Nash existence included – transfers to behavioral strategies in perfect recall games

Kuhn's Theorem Is a License



So you may optimize over the small object and still claim results about the big one.

- Every theorem you proved about mixed strategies – Nash existence included – transfers to behavioral strategies in perfect recall games
- This is why a poker solver can search a space of size $O(\sum_j |X(I^j)|)$ instead of $O(\prod_j |X(I^j)|)$

Kuhn's Theorem Is a License



So you may optimize over the small object and still claim results about the big one.

- Every theorem you proved about mixed strategies – Nash existence included – transfers to behavioral strategies in perfect recall games
- This is why a poker solver can search a space of size $O(\sum_j |X(I^j)|)$ instead of $O(\prod_j |X(I^j)|)$
- **Warning:** abstraction, the trick that shrinks poker trees by merging information sets, can **destroy** perfect recall. Then Kuhn's theorem no longer covers you, and solvers must argue separately.



In-Class Quiz 2

Open the Google Form now.

Questions refer only to what was just discussed –
you cannot answer them without having been in class.

Lecture 10 and Week 5: What We Built



- Mixed strategy: randomize over complete plans, once. Behavioral strategy: randomize at each information set

Lecture 10 and Week 5: What We Built



- Mixed strategy: randomize over complete plans, once. Behavioral strategy: randomize at each information set
- Equivalence is defined by **node reaching probabilities**, and leaves suffice to check it

Lecture 10 and Week 5: What We Built



- Mixed strategy: randomize over complete plans, once. Behavioral strategy: randomize at each information set
- Equivalence is defined by **node reaching probabilities**, and leaves suffice to check it
- Equivalent strategies give identical utilities to **everyone**, not just to the player who switched

Lecture 10 and Week 5: What We Built



- Mixed strategy: randomize over complete plans, once. Behavioral strategy: randomize at each information set
- Equivalence is defined by **node reaching probabilities**, and leaves suffice to check it
- Equivalent strategies give identical utilities to **everyone**, not just to the player who switched
- Forgetting which action you took kills mixed $\rightarrow$ behavioral; forgetting whether you moved kills behavioral $\rightarrow$ mixed

Lecture 10 and Week 5: What We Built



- Mixed strategy: randomize over complete plans, once. Behavioral strategy: randomize at each information set
- Equivalence is defined by **node reaching probabilities**, and leaves suffice to check it
- Equivalent strategies give identical utilities to **everyone**, not just to the player who switched
- Forgetting which action you took kills mixed $\rightarrow$ behavioral; forgetting whether you moved kills behavioral $\rightarrow$ mixed
- Theorem 6.11: behavioral has an equivalent mixed iff no information set meets a path twice

Lecture 10 and Week 5: What We Built



- Mixed strategy: randomize over complete plans, once. Behavioral strategy: randomize at each information set
- Equivalence is defined by **node reaching probabilities**, and leaves suffice to check it
- Equivalent strategies give identical utilities to **everyone**, not just to the player who switched
- Forgetting which action you took kills mixed $\rightarrow$ behavioral; forgetting whether you moved kills behavioral $\rightarrow$ mixed
- Theorem 6.11: behavioral has an equivalent mixed iff no information set meets a path twice
- Kuhn 1957: perfect recall gives the other direction, so the two notions coincide

Lecture 10 and Week 5: What We Built



- Mixed strategy: randomize over complete plans, once. Behavioral strategy: randomize at each information set
- Equivalence is defined by **node reaching probabilities**, and leaves suffice to check it
- Equivalent strategies give identical utilities to **everyone**, not just to the player who switched
- Forgetting which action you took kills mixed $\rightarrow$ behavioral; forgetting whether you moved kills behavioral $\rightarrow$ mixed
- Theorem 6.11: behavioral has an equivalent mixed iff no information set meets a path twice
- Kuhn 1957: perfect recall gives the other direction, so the two notions coincide
- **Next week:** we stop analyzing given games and start **designing** them



End of Lecture 10

See you next class.



भारतीय प्रौद्योगिकी संस्थान मुंबई

Indian Institute of Technology Bombay