CS 405/6001: Game Theory and Algorithmic Mechanism Design **Jul–Nov 2026**

Lecture 9: Imperfect Information Extensive Form Games

Lecturer: Swaprava Nath *Scribe(s): Claude Scribe*

Disclaimer: *These notes aggregate content from several texts and have not been subjected to the usual scrutiny deserved by formal publications. If you find errors, please bring to the notice of the Instructor.*

9.1 Recap and Motivation

In Lectures 7 and 8 we built perfect information extensive form games (PIEFGs) and their equilibrium notion, subgame perfection. Everything there rested on one assumption that we never named: at every history, the player about to move knows exactly which history she is at. She sees the whole board.

That assumption is wrong for most games we care about. In poker you do not see the opponent's hole cards. In reconnaissance blind chess[1] you do not see the opponent's pieces at all. In a sealed-bid auction you do not see the other bids. And, more basic than any of these, a PIEFG cannot represent a simultaneous move: the tree forces somebody to move first, and whoever moves second sees it.

This lecture removes the assumption. The object we add is the **information set**, and once we have it we can ask a question that has no analogue in the perfect information world: *is it always good to know more?*

9.2 Information Sets

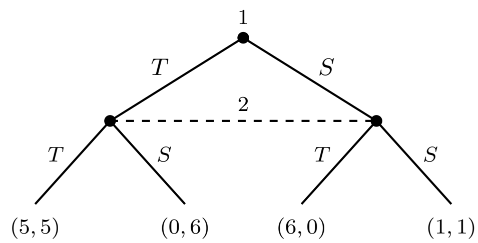
Example 9.1 (Two AI labs, one product launch) *Two AI labs are racing to launch a new product. Each chooses T, test thoroughly before shipping, or S, ship fast and iterate in public. Both commit before either sees what the other is doing.*

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

This is a normal form game. Can we draw it as a tree?

We can, provided we are allowed to say that the second mover cannot tell which branch was taken. Draw Lab 1's move at the root, then Lab 2's two decision nodes, and connect those two nodes with a dashed line:

[1] https://rbc.jhuapl.edu/



The dashed line says: Lab 2 knows she is at one of these two nodes, and knows nothing more. The two histories form an **information set** for player 2.

Remark 9.2 (What an information set is, for a machine learning audience) *An information set is the* observation *an agent conditions on. The partition of a player's decision nodes into information sets* is *that player's sensor. Two histories in the same cell are two world states the sensor cannot separate. A partially observable Markov decision process makes exactly this move, and for exactly this reason.*

Definition 9.3 (IIEFG, MSZ §6) *An* ***imperfect information extensive form game*** *is a 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* $i \in N$,

$$I_i = (I_i^1, I_i^2, \ldots, I_i^{k(i)})$$

is a partition of $\{h \in H \setminus Z : P(h) = i\}$ *with the property that*

$$X(h) = X(h') \quad \text{and} \quad P(h) = P(h') = i \qquad \text{whenever } \exists j \text{ s.t. } h, h' \in I_i^j.$$

Each I_i^j *is an* ***information set*** *of player* i.

Example 9.4 (A support agent, moving twice) *A support agent (player 1) handles a ticket. At the root she chooses* S*, resolve it herself, or* E*, escalate it to a specialist. If she escalates, the specialist (player 2) privately approves (*a*) or denies (*d*) a budget; the agent does not see which. Either way, the agent then chooses* F*, follow up, or* W*, wait. Payoffs* (u_1, u_2) *are* $(2, 2)$ *after* S*;* $(5, 3)$ *after* E, a, F*;* $(3, 3)$ *after* E, a, W*;* $(1, 4)$ *after* E, d, F*;* $(4, 2)$ *after* E, d, W.

Here $N = \{1, 2\}$*, and player 1 moves twice. Her first decision is a singleton information set,* $I_1^1 = \{\emptyset\}$ *(the root). Because she never observes whether the specialist approved or denied, her second information set groups both histories into one cell:* $I_1^2 = \{h_a, h_d\}$*, of size* 2*. This is exactly the definition above with* $k(1) = 2$*:* $X(h_a) = X(h_d) = \{F, W\}$*, which is the common-action-set condition the next remark explains. Her strategy set is the Cartesian product over her two information sets,* $S_1 = X(I_1^1) \times X(I_1^2) = \{S, E\} \times \{F, W\}$*, four pure strategies in all:* (S, F)*,* (S, W)*,* (E, F)*,* (E, W)*, where the second coordinate is only consulted when she has escalated.*

Two remarks on the last condition, which is not decoration.

Remark 9.5 (Why the action sets must agree) *Suppose two nodes in one information set offered different action menus. Then the player could distinguish them just by looking at her own list of options. Counting your available moves is free information, so allowing different menus would silently contradict the claim that the nodes are indistinguishable. In implementation terms: if your agent's action mask differs across two states, the mask has already leaked the state. One information set, one mask.*

Remark 9.6 (The tree is not unique) *Example 9.1 can equally be drawn with Lab 2 at the root and Lab 1's two nodes joined by the dashed line. Both trees represent the same simultaneous move game. Who is drawn "first" is bookkeeping, not a modeling claim.*

Because actions agree across an information set, we may write $X(I_i^j)$ for the common action set, and a strategy becomes a map from information sets to actions:

$$S_i = \prod_{I' \in I_i} X(I') = \prod_{j=1}^{k(i)} X(I_i^j). \tag{9.1}$$

This is the substantive change from the PIEFG definition, where a strategy was a map from *histories* to actions. A player with $k(i)$ information sets and two actions at each has $2^{k(i)}$ pure strategies, no matter how many histories the tree contains.

Remark 9.7 (The representation hierarchy) *Collapse every information set to a singleton and an IIEFG is a PIEFG. Give the second mover one information set containing all her nodes and it is an NFG. So IIEFG is strictly richer than both, and every result we proved for PIEFGs is a special case rather than a separate theory. The representation is not succinct, though: writing an $n \times n$ NFG as a tree costs n^2 leaves.*

9.3 Is More Information Always Good?

We now have the vocabulary to ask whether information helps. The answer splits cleanly along the zero-sum boundary, and the two halves of this section are meant to be read against each other.

9.3.1 Zero-sum: information cannot hurt

Example 9.8 (Red team against a guardrail) *An API serves one of two model checkpoints. A red-teamer (Player I) picks an attack style, a guardrail (Player II) picks a filter. The number in each cell is the probability the attack gets through: Player I wants it high, Player II wants it low, so the game is zero-sum. Nature picks the checkpoint with probability $\frac{1}{2}$ each, and* neither *player sees the realization.*

		Player II	
		L	R
Player I	T	0	$\frac{1}{2}$
	B	0	1

		Player II	
		L	R
Player I	T	1	0
	B	$\frac{1}{2}$	0

Matrix G_1 on the left, matrix G_2 on the right. Here T is a tool-use framing and B an obfuscated payload; L is a lexical filter and R a reasoning classifier.

The extensive form has a Nature node at the root with two branches of probability $\frac{1}{2}$. Player I has a single information set spanning both of Nature's children, since she does not know which checkpoint is live. Player II has a single information set spanning all four of Player I's children, since she sees neither Nature's draw nor Player I's move.

Both players have exactly two pure strategies, so the normal form is the entry-wise average of G_1 and G_2:

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

For instance the (T, L) entry is $\frac{1}{2} \cdot 0 + \frac{1}{2} \cdot 1 = \frac{1}{2}$. This is a 2×2 matrix game with no saddle point, and its MSNE is

$$\left(\left(\tfrac{1}{2}, \tfrac{1}{2}\right), \left(\tfrac{1}{2}, \tfrac{1}{2}\right)\right), \qquad \text{value} = \tfrac{3}{8}.$$

Attacks land about 37.5% of the time.

Now 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* singletons, one per checkpoint, and her pure strategies become pairs (action at G_1, action at G_2):

		Player II	
		L	R
Player I	T_1T_2	$\frac{1}{2}$	$\frac{1}{4}$
	T_1B_2	$\frac{1}{4}$	$\frac{1}{4}$
	B_1T_2	$\frac{1}{2}$	$\frac{1}{2}$
	B_1B_2	$\frac{1}{4}$	$\frac{1}{2}$

The row B_1T_2 weakly dominates every other row and is worth $\frac{1}{2}$ against either column, so

$$\left((1(B_1T_2)), (p, 1-p)\right), \ p \in [0, 1], \qquad \text{value} = \tfrac{1}{2}.$$

The attacker's payoff rose from $\frac{3}{8}$ to $\frac{1}{2}$. The version banner was worth $\frac{1}{8}$.

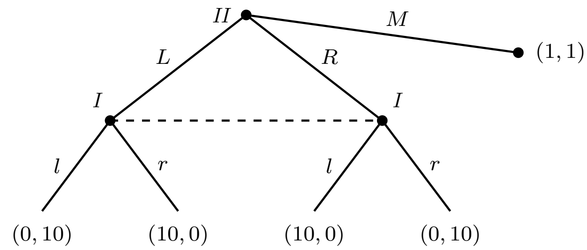
Theorem 9.9 (Information addition, MSZ) *Let Γ be a two-player zero-sum game in extensive form and let Γ' be the game derived from Γ by splitting several information sets of Player I. Then the value of Γ' in mixed strategies is greater than or equal to the value of Γ in mixed strategies.*

Proof: Exercise. Hint: every strategy Player I had in Γ is still available in Γ', since she can simply ignore the extra signal and play the same action at both of the split information sets. Her maxmin therefore cannot fall, and in a zero-sum game the maxmin is the value. ■

The hint is the whole idea, and it also tells us where the argument must break. It uses only that Player I's strategy set grew. That is enough in a zero-sum game because Player II's response cannot help Player I anyway: whatever Player II does, Player I's guarantee is computed against the worst case. In a general-sum game, Player II's response *can* help Player I, and taking it away is a real loss.

9.3.2 General-sum: information can hurt everyone

Example 9.10 (Should you open source your safety classifier?) *Player II is a platform, Player I an adversary. Player II chooses M, ship nothing and live with a modest baseline, or deploys one of two classifier variants L or R. If a variant is deployed, Player I probes with l or r; guessing right beats the classifier. Payoffs (u_I, u_{II}):*



Call this ***Game A****. The dashed line says the adversary cannot tell L from R.*

In Game A the subgame reached after L or R is a matching-pennies game, so the adversary cannot do better than a coin flip and neither can the platform. The MSNE is

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

Since $5 > 1$, the platform deploys.

Now let the platform publish which variant is live. The dashed line disappears and the adversary's information set splits into two singletons, with actions relabelled l_1, r_1 after R and l_2, r_2 after L. Call this **Game B**. The adversary now matches whichever variant is deployed, so deploying is worth 0 to the platform, and

$$((1(l_1 r_2)), (0(L), 0(R), 1(M))) \implies \text{expected payoff } (1, 1).$$

Remark 9.11 (The point of the pair) *Player I received strictly more information and ended up strictly worse off, from* 5 *to* 1*. So did Player II. Theorem 9.9 does not apply, because Game A is general-sum. The mechanism is that the extra information changed the* opponent's *behavior, and that indirect effect swamped the direct benefit of knowing more. This generalizes: publishing more information, a model card, a paper, benchmark numbers, changes what a rival optimizes against. Before you disclose, ask whose behavior the disclosure will change, and in which direction.*

9.4 Summary and Preview

We introduced information sets and defined imperfect information extensive form games (Definition 9.3), noting that the common-action-set requirement is forced rather than convenient. Strategies are now maps from information sets to actions, equation (9.1), and the IIEFG representation contains both the NFG and the PIEFG as special cases. We then asked whether information helps: in two-player zero-sum games, splitting your own information sets weakly raises your value (Theorem 9.9), illustrated by $\frac{3}{8} \to \frac{1}{2}$ in Example 9.8; in general-sum games it can make everyone worse off, illustrated by $(5, 5) \to (1, 1)$ in Example 9.10.

Next lecture we keep the game fixed and ask a different question: once a player cannot see everything, *how* should she randomize? Randomizing over complete plans and randomizing afresh at each information set turn out to be genuinely different, and the condition under which they coincide is a condition on memory.

References

[BS18] N. Brown and T. Sandholm, "Superhuman AI for heads-up no-limit poker: Libratus beats top professionals," *Science*, 359(6374), 2018, pp. 418–424.

[Kuh53] H. W. Kuhn, "Extensive games and the problem of information," in *Contributions to the Theory of Games II*, Annals of Mathematics Studies 28, Princeton University Press, 1953, pp. 193–216.

[MSZ13] M. Maschler, E. Solan and S. Zamir, *Game Theory*, Chapter 6, Cambridge University Press, 2013.