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

Lecture 10: Behavioral Strategies and Perfect Recall

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.*

10.1 Recap

Last lecture we added information sets to the extensive form and asked what a player *knows*. This lecture asks what a player *does* with a coin. In a normal form game there was only one way to randomize: put a distribution on pure strategies. In an extensive form game with information sets there are two, and they are not obviously the same object.

10.2 Two Ways to Be Random

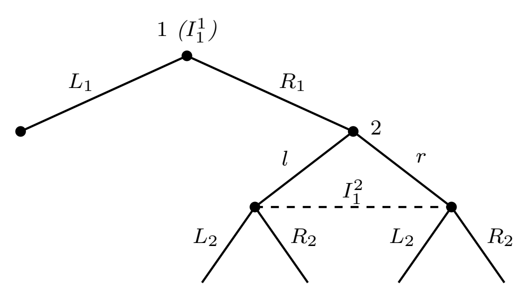
Recall from equation (9.1) that a pure strategy of player i is an element of $S_i = \prod_{j=1}^{k(i)} X(I_i^j)$: a complete plan, one action fixed in advance for every information set she might reach. A **mixed strategy** $\sigma_i \in \Delta(S_i)$ randomizes over these plans. Concretely: draw one plan before the game starts, then follow it deterministically.

The alternative is to carry the coin to the table.

Definition 10.1 (Behavioral strategy) *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.

Remark 10.2 (The distinction for a machine learning audience) *A behavioral strategy is a* stochastic policy $\pi(a \mid o)$*: at each observation you sample an action. A mixed strategy is a distribution over* deterministic policies*, sampled once at episode start and then frozen. These feel interchangeable, and most of the time they are. The content of this lecture is exactly when they are not.*

Example 10.3 (The running tree) *Player 1 moves at the root, information set I_1^1 with actions L_1, R_1. After R_1, player 2 moves with actions l, r. Both of player 2's children belong to a single information set I_1^2 of player 1, with actions L_2, R_2.*



Player 1's pure strategies are $(L_1L_2), (L_1R_2), (R_1L_2), (R_1R_2)$*, so a mixed strategy* σ_1 *is four numbers summing to one. A behavioral strategy* b_1 *is two numbers:* $b_1(I_1^1) \in \Delta(L_1, R_1)$ *and* $b_1(I_1^2) \in \Delta(L_2, R_2)$*.*

Note already the shape of the difference. Mixed strategies of player 1 live in a simplex in $\mathbb{R}^4$; behavioral strategies live in a product of two simplices in $\mathbb{R}^2$. The mixed object looks larger. Whether it is genuinely larger is the question.

Remark 10.4 (Why the size difference matters in practice) *Take a player with* 4 *information sets and* 2 *actions at each. A mixed strategy needs* $2^4 - 1 = 15$ *free variables, a behavioral strategy needs* 4*. With* 20 *information sets the numbers are about* 10^6 *against* 20*. Heads-up no-limit poker has on the order of* 10^{161} *histories; nobody writes down a distribution over plans. Counterfactual regret minimization (CFR) tracks, at each information set, how much better every action would have done against the opponents' play so far, its* regret, *and shifts probability toward the actions with the least regret; repeated self-play drives the average regret to zero and the average behavioral strategy converges to a Nash equilibrium. CFR and its descendants iterate on behavioral strategies, one information set at a time, and are the algorithm behind Libratus, the first bot to beat professional humans at heads-up no-limit poker [ZJBP07,BS18]. The entire computational case for behavioral strategies is this exponent, and Kuhn's theorem below is what makes using them legitimate.*

10.3 Equivalence

To compare a σ_i with a b_i we need a common yardstick. The right one is the probability of reaching each node.

Write $\rho(x;\sigma)$ for the probability of reaching node x under the mixed profile σ, and $\rho(x;b)$ for the same under a behavioral profile b. Different players may use different kinds of strategies, so $\rho(x;\sigma_1,b_2)$ is meaningful too.

Example 10.5 (Computing ρ) *In Example 10.3, let* x *be the right-hand child of player 2's node, that is, the node reached by* R_1 *then* r*. Then*

$$\rho(x;\sigma) = \sigma_1(R_1)\,\sigma_2(r) = (\sigma_1(R_1L_2) + \sigma_1(R_1R_2)) \cdot \sigma_2(r),$$
$$\rho(x;b) = b_1(I_1^1)(R_1) \cdot b_2(I_2^1)(r),$$
$$\rho(x;\sigma_1,b_2) = (\sigma_1(R_1L_2) + \sigma_1(R_1R_2)) \cdot b_2(I_2^1)(r).$$

Definition 10.6 (Equivalence) *A mixed strategy* σ_i *and a behavioral strategy* b_i *of player* i *are* ***equivalent*** *if for every mixed or behavioral strategy* ξ_{-i} *of the other players and* every *vertex* x *in the game tree,*

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

For the tree of Example 10.3 the translation is explicit:

$$\begin{aligned} b_1(I_1^1)(L_1) &= \sigma_1(L_1L_2) + \sigma_1(L_1R_2), \\ b_1(I_1^1)(R_1) &= \sigma_1(R_1L_2) + \sigma_1(R_1R_2), \\ b_1(I_1^2)(L_2) &= \sigma_1(L_2 \mid R_1), \qquad b_1(I_1^2)(R_2) = \sigma_1(R_2 \mid R_1). \end{aligned} \tag{10.1}$$

Read the recipe in words: at the first information set, *add up* the plans that begin with L_1; at the second, *condition* on having arrived there. The conditioning step needs the arrival event to have positive probability, and it needs "arriving at I_1^2" to be a well-defined event that does not depend on which node of I_1^2 we mean. Both requirements will fail in Section 10.4, and both failures are the same failure.

Claim 10.7 *It is enough to check the equivalence of Definition 10.6 at the leaf nodes.*

Proof: Pick an arbitrary non-leaf node x. Every play that reaches x continues to exactly one leaf in the subtree rooted at x, and every leaf of that subtree is reached only through x. So $\rho(x;\cdot)$ is the sum of $\rho(z;\cdot)$ over leaves z in the subtree of x. If the leaf probabilities agree, so does every $\rho(x;\cdot)$. ■

Theorem 10.8 (Utility equivalence, MSZ) *If σ_i and b_i are equivalent, then for every mixed or behavioral strategy vector ξ_{-i} of the other players,*

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

Proof: Utility is the leaf-probability-weighted sum of payoffs, $u_j(\cdot) = \sum_{z \in Z} \rho(z;\cdot)\, u_j(z)$. By Claim 10.7 the leaf probabilities already agree, so the sums agree term by term. ■

Theorem 10.8 is still a statement 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 player $i \in N$.

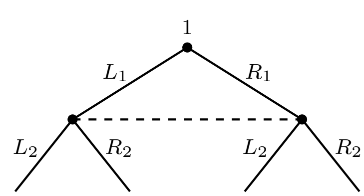
Corollary 10.9 *If σ and b are equivalent, then $u_i(\sigma) = u_i(b)$ for all $i \in N$.*

Note how strong Theorem 10.8 is: the conclusion holds for *every* player j, not just for the player i who swapped representations. This is what lets us replace one by the other inside an equilibrium argument without checking anything else.

10.4 When Equivalence Breaks

The natural conjecture is that the construction (10.1) always works. Two small trees kill it, one in each direction.

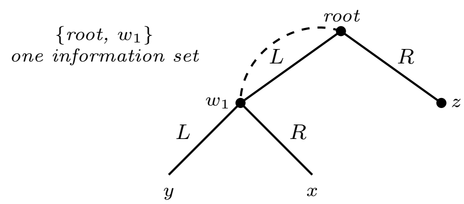
Example 10.10 (Forgetting which action you took) *Player 1 moves twice in a row, and her two second-stage nodes lie in a single information set: she remembers that she moved but not what she played.*



Consider the mixed strategy that puts $\frac{1}{2}$ on L_1L_2 and $\frac{1}{2}$ on R_1R_2, so that $\sigma_1(L_1R_2) = \sigma_1(R_1L_2) = 0$. No behavioral strategy reproduces it. A behavioral strategy flips independently at the two information sets, so if b_1 gives positive weight to both L_1 and R_2, the leaf reached by L_1 then R_2 gets positive probability. Hence: ***a mixed strategy with no equivalent behavioral strategy****.*

Remark 10.11 (Example 10.10 in practice) *This is the agent whose context got compacted. It takes a step, the conversation is summarized, the raw step is dropped, and at the next call the agent knows* that *it acted but not* what *it did: it can only run a* behavioral strategy*, deciding afresh at each information set with no memory of the earlier draw. The mixed strategy above is a* correlated *plan: be consistent, whichever way you go. The correlation lived in the plan, and forgetting destroyed the only place it could be stored.*

Example 10.12 (Forgetting whether you moved) *Player 1 moves at the root; if she plays L she moves again, and the root and that child lie in the* same *information set.*



A pure strategy fixes one action for the whole information set, so it plays L at the root and *L at the child (reaching y), or R at both (reaching z). The path L followed by R, reaching x, is therefore unreachable under every pure strategy, hence under every mixed strategy. A behavioral strategy flips a fresh coin at each visit and reaches x with positive probability. Hence:* ***a behavioral strategy with no equivalent mixed strategy****.*

Remark 10.13 (Example 10.12 in practice) *This is a retry with no idempotency key. The client times out and re-sends without knowing whether the first request already went through, so "before the call" and "after the call" sit in one information set. The disease is visiting one information set twice on a single path.*

The second example generalizes into a clean graph-theoretic statement. Let x be a non-root node, and let the *action leading to* x at an ancestor x_1 be the unique edge out of x_1 on the path from the root to x.

Lemma 10.14 *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 that information set has a behavioral strategy with no equivalent mixed strategy.*

Proof: Let x_1 and x_1' be two such vertices in the same information set I_i^j on the path to x, with different actions leading to x. Any pure strategy of i selects a single action at I_i^j, hence cannot take both of those actions, hence cannot generate a play reaching x. Every mixed strategy is a randomization over pure strategies, so $\rho(x; \sigma_i, \xi_{-i}) = 0$ for every σ_i and every ξ_{-i} that reaches the relevant part of the tree. A behavioral strategy randomizes independently at each visit, so choosing full support at I_i^j gives $\rho(x; b_i, \xi_{-i}) > 0$. No mixed strategy can match it. ■

Theorem 10.15 (MSZ Thm. 6.11) *Consider an IIEFG such that every vertex has at least two actions. Every behavioral strategy has an equivalent mixed strategy for a player if and only if each information set of that player intersects every path emanating from the root at most once.*

Proof: Reading exercise from MSZ. The "only if" direction is Lemma 10.14. ■

10.5 Perfect Recall and Kuhn's Theorem

Theorem 10.15 handles one direction. For the other we need to rule out Example 10.10 as well, and that means writing down what it means for a player not to be forgetful.

Definition 10.16 (Choice of the same action at an information set) *Let $X = (x^0, x^1, \ldots, x^K)$ and $\hat{X} = (x^0, \hat{x}^1, \ldots, \hat{x}^L)$ be two paths in the game tree, and let I_i^j be an information set of player i that intersects these two paths at exactly one vertex each, say x^k and $\hat{x}^l$. The two paths **choose the same action at** I_i^j if $k < K$ and $l < L$, and the action at x^k leading to x^{k+1} equals the action at $\hat{x}^l$ leading to $\hat{x}^{l+1}$, written $a_i(x^k \to x^{k+1}) = a_i(\hat{x}^l \to \hat{x}^{l+1})$.*

Here "leading to" need not relate a parent to an immediate child: paths in a tree are unique, so we may speak of the action at x leading to any descendant y.

Definition 10.17 (Perfect recall) *Player i has **perfect recall** if*

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

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

Equivalently: for every I_i^j and every pair $x, y \in I_i^j$, if the decision vertices of i along the two root paths are $x_i^1, \ldots, x_i^L = x$ and $y_i^1, \ldots, y_i^{L'} = y$, then $L = L'$; $x_i^l, y_i^l \in I_i^k$ for some common k; and $a_i(x_i^l \to x_i^{l+1}) = a_i(y_i^l \to y_i^{l+1})$ for all $l = 1, \ldots, L-1$.

Remark 10.18 (Reading the two clauses) *Clause 1 bans "did I already move?" and is exactly the hypothesis of Theorem 10.15; Example 10.12 violates it. Clause 2 bans "what did I play?"; Example 10.10 violates it. Note what perfect recall does* not *require: it is a condition on memory, not on observation. Poker has perfect recall. You always remember your own bets; you simply never see the hole cards.*

Remark 10.19 (Example 1 and Example 2 of the slides) *The two three-level trees shown in class differ only in the label of the root. When the root belongs to player 3, player 1's two information sets are reached along paths that agree on player 1's own history, and both clauses hold. When the root belongs to player 1, player 1 takes two* different *actions at her first information set to reach two different vertices of her second, violating clause 2 and hence perfect recall.*

Let $S_i^*(x)$ denote the set of pure strategies of player i under which i chooses the actions leading to x, that is, the members of S_i that agree with the path from the root to x at every decision vertex of i on it.

Theorem 10.20 *If i has perfect recall and x, x' lie in the same information set of i, then $S_i^*(x) = S_i^*(x')$.*

Proof: By Definition 10.17, the paths to x and to x' pass through the same information sets of i in the same order and choose the same action at each. A pure strategy of i is a choice of one action per information set, so the constraints it must satisfy to be consistent with the path to x are literally the same constraints as for x'. ∎

Theorem 10.20 is the technical heart of what follows. It says that under perfect recall, "player i reaches this information set" is a well-defined event that does not depend on which node of the set we have in mind, which is exactly what makes the conditioning step of (10.1) legal.

Theorem 10.21 (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.*

Proof: Reading exercise, MSZ Theorem 6.15. The proof is constructive: it starts from the mixed strategy and builds the behavioral strategy by the marginalize-and-condition recipe of (10.1), using Theorem 10.20 to see that the conditioning is well defined, and then verifies that the probability of reaching each leaf is unchanged. ∎

The converse direction needs nothing new: clause 1 of Definition 10.17 is precisely the hypothesis of Theorem 10.15, so perfect recall already delivers behavioral-to-mixed. Putting the two together, in a game of perfect recall the two notions of randomization are interchangeable.

Remark 10.22 (Why this is a licence, and where the licence expires) *Every theorem proved about mixed strategies, Nash existence included, transfers to behavioral strategies in a perfect recall game. This is why a solver may search a space of size $O(\sum_j |X(I_i^j)|)$ and still make claims about a space of size $O(\prod_j |X(I_i^j)|)$. The warning is that* abstraction*, the standard trick of shrinking a poker tree by merging information sets, can destroy perfect recall: merged sets may be reachable by paths that took different actions earlier. When that happens Kuhn's theorem no longer covers you, and the guarantees have to be argued separately.*

10.6 Summary and Preview

We distinguished mixed strategies, which randomize over complete plans once, from behavioral strategies, which randomize at each information set (Definition 10.1). Equivalence is defined through node-reaching probabilities (Definition 10.6), leaves suffice to check it (Claim 10.7), and equivalent strategies give identical utilities to every player (Theorem 10.8). Two forms of forgetfulness break the correspondence in opposite directions (Examples 10.10 and 10.12); Theorem 10.15 characterizes one direction by a path condition, and Kuhn's theorem (Theorem 10.21) supplies the other under perfect recall.

That closes our analysis of games as given objects. From next week we change roles: instead of asking how players behave in a game someone handed us, we ask how to *design* the game so that the behavior we want is the behavior that emerges. This is mechanism design.

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. (Often cited with the 1957 date of the companion volume; the slides use that convention.)

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

[ZJBP07] M. Zinkevich, M. Johanson, M. Bowling and C. Piccione, "Regret minimization in games with incomplete information," *NeurIPS*, 2007.