CSCE 631 — Algorithmic Game Theory

Lecture 12: Extensive-Form Games III — Sequence Form

Alan Kuhnle

Summer 2026

Texas A&M University

Today's Plan

Motivation: The Normal-Form Blowup

The Computational Problem

- We know how to solve two-player zero-sum *normal-form* games in polynomial time via LP (minimax theorem)
- Naive approach to extensive-form games:
  1. Convert to normal form
  2. Solve the LP
- **Problem:** the normal form is *exponentially large*

Size Explosion

If player i has K information sets each with m actions, then $|S_i| = m^K$. The normal-form payoff matrix has $|S_1| \times |S_2|$ entries. For a game tree with n nodes, $K = \Theta(n)$, so the normal form can have size $\Omega(m^n)$.

Example: Size Comparison

Simplified poker with 3 card types, 2 betting rounds:

	Game Tree Size	Normal-Form Size
Nodes	~ 200	—
Info sets (P1)	~ 12	—
Info sets (P2)	~ 12	—
Pure strategies (P1)	—	$\sim 2^{12} = 4{,}096$
Pure strategies (P2)	—	$\sim 2^{12} = 4{,}096$
Matrix entries	—	$\sim 16{,}000{,}000$

- A game tree with 200 nodes produces a matrix with 16 million entries
- Texas Hold'em: $\sim 10^{14}$ info sets $\Rightarrow$ normal form is hopeless
- We need algorithms that work on the *tree* directly

Sequences and Realization Plans

Key Idea: Sequences Instead of Strategies

Observation

Under a behavioral strategy, the probability of reaching any node *factorizes* over the information sets on the path (by perfect recall and Kuhn's theorem).

- Instead of enumerating all m^K pure strategies, describe a player's plan by specifying probabilities for *sequences* of actions
- A **sequence** for player i is the ordered list of actions taken by player i on a path from the root to some node
- The number of sequences is at most the number of nodes in the tree: polynomial, not exponential

Sequences: Formal Definition

Definition

A **sequence** σ_i for player i is a (possibly empty) ordered list of (I, a) pairs:

$$\sigma_i = ((I_1, a_1), (I_2, a_2), \ldots, (I_k, a_k))$$

where $I_1, \ldots, I_k$ are information sets of player i encountered in order on some root-to-node path, and $a_j \in A(I_j)$ is the action taken at I_j.

Let Σ_i denote the set of all sequences for player i.

- The **empty sequence** $\emptyset \in \Sigma_i$ corresponds to the root (before player i has acted)
- $|\Sigma_i| \leq |V|$ (each non-root node has at most one associated sequence per player)
- Each sequence is a *prefix* of a root-to-leaf path restricted to player i's actions

Realization Plans

Definition

A **realization plan** for player i is a function $x_i : \Sigma_i \to [0,1]$ satisfying:

1. $x_i(\emptyset) = 1$ (the game always starts)
2. For every information set $I \in \mathcal{I}_i$ and every sequence σ_i leading to I:

$$\sum_{a \in A(I)} x_i(\sigma_i \cdot (I, a)) = x_i(\sigma_i)$$

- $x_i(\sigma_i)$ = probability that player i plays according to sequence σ_i (i.e., takes all actions in σ_i)
- Constraint (2): probability flowing into an information set equals probability flowing out (conservation)
- **Dimension:** $|\Sigma_i|$ variables, $|\mathcal{I}_i| + 1$ constraints

Realization Plans and Behavioral Strategies

Theorem

There is a bijection between realization plans and behavioral strategies (for games with perfect recall). Given behavioral strategy β_i:

$$x_i(\sigma_i) = \prod_{(I_j, a_j) \in \sigma_i} \beta_i(a_j \mid I_j)$$

Conversely, given realization plan x_i with $x_i(\sigma_I) > 0$ where σ_I leads to I:

$$\beta_i(a \mid I) = \frac{x_i(\sigma_I \cdot (I, a))}{x_i(\sigma_I)}$$

- Realization plans are an equivalent, *linear* representation of behavioral strategies
- The probability products become *linear constraints* — this is what makes the LP

Sequence-Form LP

Expected Payoff in Sequence Form

For a two-player game, the expected payoff to Player 1 can be written as:

$$U_1(x_1, x_2) = \sum_{z \in Z} u_1(z) \cdot \pi_c(z) \cdot x_1(\sigma_1(z)) \cdot x_2(\sigma_2(z))$$

where:

- $\sigma_i(z)$ is the sequence of player i on the path to terminal node z
- $\pi_c(z)$ is the product of chance probabilities on the path to z
- $x_i(\sigma_i(z))$ is the realization probability for player i's sequence

Define the **sequence-form payoff matrix** $A \in \mathbb{R}^{|\Sigma_1| \times |\Sigma_2|}$:

$$A[\sigma_1, \sigma_2] = \sum_{\substack{z \in Z: \\ \sigma_1(z) = \sigma_1,\ \sigma_2(z) = \sigma_2}} u_1(z) \cdot \pi_c(z)$$

Then $U_1(x_1, x_2) = x_1^\top A x_2$.

For a **two-player zero-sum** game, we can compute the value and optimal strategies via LP. By strong LP duality:

Player 1 (max-player):

$$\begin{aligned} \max_{x_1} \quad & p^\top x_1 \\ \text{s.t.} \quad & E_1 x_1 = e_1 \\ & x_1 \geq 0 \end{aligned}$$

where $p = A x_2^*$ (or we take the dual formulation).

Combined primal-dual LP:

$$\max_{x_1 \geq 0} \min_{x_2 \geq 0} x_1^\top A x_2 \quad \text{s.t.} \quad E_1 x_1 = e_1, \quad E_2 x_2 = e_2$$

E_i encodes the realization-plan constraints; e_i is the right-hand side ($e_i[0] = 1$, rest

The Sequence-Form LP (Explicit)

LP Formulation (von Stengel, 1996)

$$
\begin{aligned}
\max_{x_1, v_2} \quad & e_2^\top v_2 \\
\text{s.t.} \quad & E_1 x_1 = e_1 \\
& A^\top x_1 - E_2^\top v_2 \leq 0 \\
& x_1 \geq 0
\end{aligned}
$$

Dual:

$$
\begin{aligned}
\min_{x_2, v_1} \quad & e_1^\top v_1 \\
\text{s.t.} \quad & E_2 x_2 = e_2 \\
& A x_2 - E_1^\top v_1 \geq 0
\end{aligned}
$$

Computational Advantage

Size Comparison: Normal Form vs. Sequence Form

	Normal Form LP	**Sequence Form LP**
Variables	$\lvert S_1\rvert + \lvert S_2\rvert$ (exponential in tree)	$\lvert\Sigma_1\rvert + \lvert\Sigma_2\rvert$ (linear in tree)
Constraints	$\lvert S_1\rvert + \lvert S_2\rvert + 2$ (exponential)	$\lvert\mathcal{I}_1\rvert + \lvert\mathcal{I}_2\rvert + \lvert\Sigma_1\rvert + \lvert\Sigma_2\rvert$ (linear in tree)
Matrix size	$\lvert S_1\rvert \times \lvert S_2\rvert$ (doubly exponential)	$\lvert\Sigma_1\rvert \times \lvert\Sigma_2\rvert$ (polynomial)
Sparsity	Dense in general	**Sparse:** $O(\lvert Z\rvert)$ nonzeros

The sequence-form LP is **polynomial in the size of the game tree**.

The Main Theorem

Theorem (Koller, Megiddo, von Stengel, 1996)
A two-player zero-sum extensive-form game with perfect recall can be solved in time **polynomial** in the size of the game tree.

Proof idea:

1. The sequence-form LP has $O(|V|)$ variables and $O(|V|)$ constraints
2. The LP can be solved in polynomial time (e.g., interior-point methods)
3. Therefore, the total running time is poly$(|V|)$

- This was a breakthrough: prior to the sequence form, solving even moderate-sized games was impractical
- The result holds for any number of chance nodes and any information structure (as long as perfect recall is satisfied)

Constraint Matrix Structure

The constraint matrix E_i for realization-plan constraints has a **tree structure**:

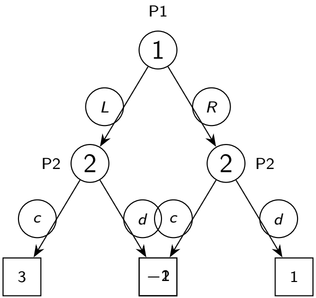
- Each row corresponds to an information set I (plus one row for the root)
- Each column corresponds to a sequence σ_i
- Row for I: has -1 in the column for the sequence leading to I, and $+1$ in columns for sequences extending I by one action
- The matrix is *totally unimodular* (all entries in $\{-1, 0, 1\}$, tree structure)

Example: If I has parent sequence σ and actions $\{a, b\}$:

$$\underbrace{x_i(\sigma \cdot (I, a)) + x_i(\sigma \cdot (I, b))}_{\text{flow out}} = \underbrace{x_i(\sigma)}_{\text{flow in}}$$

Worked Example

Sequence Form: Worked Example



Perfect information, zero-sum.

Sequences:

- P1: $\emptyset, L, R \quad (|\Sigma_1| = 3)$
- P2: $\emptyset, c_L, d_L, c_R, d_R \quad (|\Sigma_2| = 5)$

Payoff matrix A (3×5):

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 3 & -1 & 0 & 0 \\ 0 & 0 & 0 & -2 & 1 \end{pmatrix}$$

Realization constraints (P1):

$x_1(\emptyset) = 1$, $x_1(L) + x_1(R) = x_1(\emptyset)$

3 variables + 5 variables = 8 total

vs. normal form: $2 \times 4 = 8$ entries

Solving the Example

Sequence-form LP for Player 1:

$$\begin{aligned} \max \quad & v_2 \\ \text{s.t.} \quad & x_L + x_R = 1 \\ & 3x_L \leq v_{2,L}, \quad -x_L \leq v_{2,L} \\ & -2x_R \leq v_{2,R}, \quad x_R \leq v_{2,R} \\ & v_{2,L} + v_{2,R} \leq v_2 \\ & x_L, x_R \geq 0 \end{aligned}$$

where $v_{2,L}, v_{2,R}$ are the values P2 gets to guarantee at each subgame.

Solution:

- Left subgame: P2 plays d, value $= -1$
- Right subgame: P2 plays c, value $= -2$

Applications to Poker

Poker: Why Sequence Form Matters

- Poker is the canonical imperfect-information game
- Hidden cards $\Rightarrow$ large information sets
- Two-player limit poker variants are zero-sum

Game	Info sets	Sequences
Kuhn poker (3 cards)	12	13
Leduc Hold'em	936	1,200
Rhode Island Hold'em	3.1M	3.9M
Heads-up limit Texas Hold'em	3.2×10^{14}	3.2×10^{14}

- Kuhn, Leduc: solvable directly by sequence-form LP
- Rhode Island: solvable with sparse LP solvers
- Texas Hold'em: too large even for sequence-form LP — need iterative methods

Abstraction for Large Games

When the sequence form is still too large:

1. **Card abstraction:** group similar card holdings into buckets (e.g., treat $K\heartsuit Q\heartsuit$ and $K\spadesuit Q\spadesuit$ identically)
2. **Action abstraction:** restrict betting actions to a finite set (e.g., fold/call/pot/all-in instead of arbitrary bet sizes)
3. Solve the abstracted game exactly (sequence-form LP or CFR)
4. Map the solution back to the original game

Abstraction Quality

The quality of the solution in the original game depends on the abstraction. Coarser abstractions $\Rightarrow$ faster to solve but worse approximation. This is an active research area (action translation, potential-aware abstractions).

Normal Form vs. Sequence Form: Summary

Comparison Summary

Aspect	**Normal Form**	**Sequence Form**
Representation	Strategy profiles	Realization plans
Size	Exponential in depth	Linear in tree size
LP variables	$\prod_i m_i^{K_i}$	$\sum_i \lvert\Sigma_i\rvert$
Applies to	Any game	Perfect recall
General-sum	Support enum.	Harder (not just LP)
Zero-sum 2P	LP (huge)	LP (compact)
Equivalent?	Yes (Kuhn)	Yes (Kuhn)

Key Takeaway

The sequence form exploits the *tree structure* and *perfect recall* to achieve an exponential reduction in representation size while preserving all strategic content.

Beyond Two-Player Zero-Sum

- **Two-player general-sum:** sequence form still provides a compact representation, but finding NE is no longer an LP
  - Can use Lemke–Howson type methods on the sequence form
  - Complexity: PPAD-complete in general
- **Multi-player:** sequence form extends naturally, but solving is harder (complementarity problems, PPAD-hard for ≥ 3 players)
- **Correlated equilibria:** can be computed in polynomial time via LP even in multi-player sequence form (Huang and von Stengel, 2008)

Looking Ahead

For games too large for LP (e.g., full poker), we need iterative methods. **Next lecture:** Counterfactual Regret Minimization (CFR).

Summary

Summary

- The normal-form representation of extensive-form games is **exponentially large**
- **Sequences:** ordered lists of a player's information-set/action pairs along a root-to-node path; $|\Sigma_i| = O(|V|)$
- **Realization plans:** probability assignments to sequences satisfying flow conservation; equivalent to behavioral strategies
- The **sequence-form LP** solves two-player zero-sum extensive-form games in time polynomial in $|V|$
- The LP has $O(|V|)$ variables, $O(|V|)$ constraints, and $O(|Z|)$ nonzeros — exploiting tree structure and sparsity
- Applications: small/medium poker variants solved exactly; larger games require abstraction + iterative methods

Next time: Counterfactual Regret Minimization (CFR) for iteratively solving large extensive-form games