Different Strokes in Randomised Strategies: Revisiting Kuhn's Theorem with Finite-memory Assumptions

UMONS Université de Mons

fnrs FREEDOM TO RESEARCH

1. Overview

In games, randomised strategies can be defined in different ways.

- Mixed strategies randomise over pure strategies at the start.
- Behavioural strategies select a random action at each step.

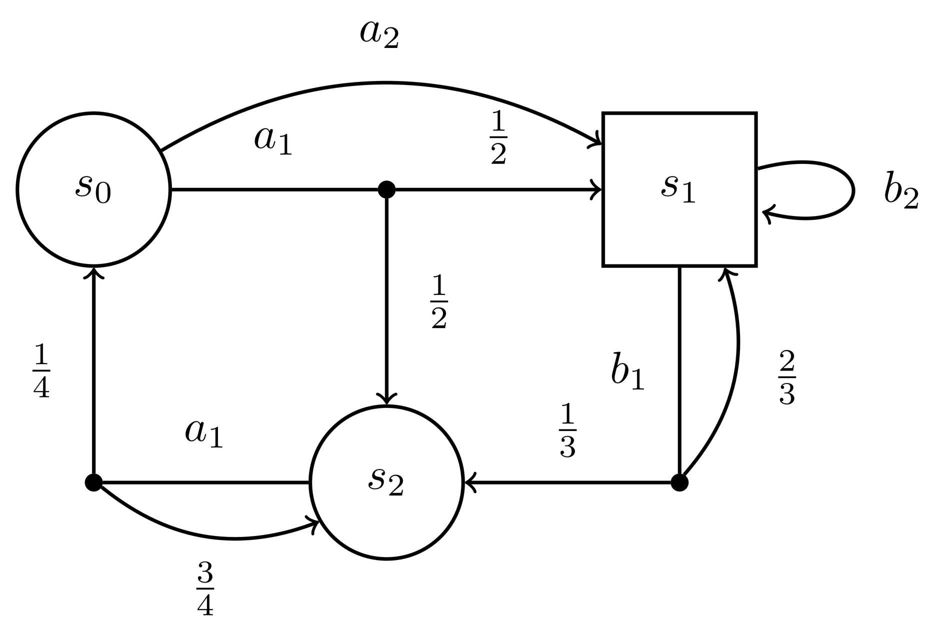
Kuhn's theorem [Aum16]

In games of perfect recall, behavioural and mixed strategies can induce the same outcomes.

Contribution on this poster

Adapting Kuhn's theorem to finite-memory strategies.

2. Stochastic games



- Two-player **turn-based games**: finite set of states S partitioned in S_1 ($\bigcirc$, for $\mathcal{P}_1$) and S_2 ($\square$, for $\mathcal{P}_2$).
- Probabilistic transitions labelled by actions $a \in A$.

3. Strategies and outcome-equivalence

Strategy of $\mathcal{P}_i$: function $(SA)^*S_i \to \mathcal{D}(A)$.

Given an initial state $s_{\mathsf{init}} \in S$, strategies σ_1 and σ_2 of $\mathcal{P}_1$ and $\mathcal{P}_2$ induce a probability distribution over infinite plays denoted $\mathbb{P}^{\sigma_1,\sigma_2}_{s_{\mathsf{init}}}$.

How to **compare** strategies ? Via their induced distributions.

Outcome-equivalence of strategies

Two strategies σ_1 and τ_1 of $\mathcal{P}_1$ are **outcome-equivalent** if for all strategies σ_2 of $\mathcal{P}_2$ and all initial states $s_{\mathsf{init}} \in S$, $\mathbb{P}^{\sigma_1,\sigma_2}_{s_{\mathsf{init}}} = \mathbb{P}^{\tau_1,\sigma_2}_{s_{\mathsf{init}}}$.

Outcome-equivalence is agnostic to specifications.

4. Finite-memory strategies

A strategy of $\mathcal{P}_i$ is **finite-memory** (FM) if it can be induced by a stochastic Mealy machine $\mathcal{M} = (M, \mu_{\mathsf{init}}, \alpha_{\mathsf{up}}, \alpha_{\mathsf{next}})$ where

- M is a finite set of memory states;
- $\mu_{\mathsf{init}} \in \mathcal{D}(M)$ is an initial distribution;
- $\alpha_{\mathsf{up}}\colon M \times S \times A \to \mathcal{D}(M)$ is a stochastic memory update function;
- $\alpha_{\mathsf{next}}\colon M \times S_i \to \mathcal{D}(A)$ is a stochastic next-move function.

5. Kuhn-like equivalence for FM strategies?

Do we have a Kuhn-like equivalence if we require some aspects among μ_{init}, α_{next} or α_{up} to be deterministic ?

$\rightsquigarrow$ Only in some limited cases.

Only randomised outputs vs. only randomised initialisation

In one-player deterministic game graphs:

- randomised outputs can induce infinitely many paths;
- randomised initialisation can only induce finitely many.

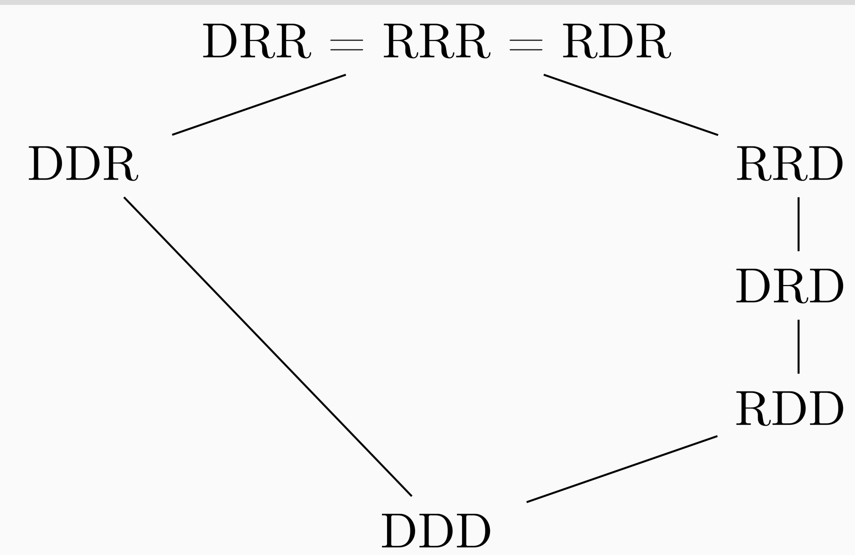
6. Classifying finite-memory strategies

We use **acronyms** to define classes of Mealy machines (hence of FM strategies): we use XYZ where X, Y, Z $\in$ {D, R} where D stands for deterministic and R for random, and

- X characterises initialisation,
- Y characterises outputs (next-move function),
- Z characterises updates.

We have the following hierarchy of FM strategies:

Kuhn-like strict classification of FM strategies [MR22]



7. RDD $\subsetneq$ DRD: trading random initialisation for outputs

For an RDD Mealy machine $\mathcal{M} = (M, \mu_{\mathsf{init}}, \alpha_{\mathsf{up}}, \alpha_{\mathsf{next}})$.

- We use an adaptation of the **subset construction**.
- State space of functions $\mathsf{supp}(\mu_{\mathsf{init}}) \to (M \cup \{\bot\})$:
  - We simulate the strategy from each initial state.
  - We interrupt inconsistent simulations (symbolised by $\bot$).

8. RRR $\subseteq$ DRR: determinising initialisation

For an RRR Mealy machine $\mathcal{M} = (M, \mu_{\mathsf{init}}, \alpha_{\mathsf{up}}, \alpha_{\mathsf{next}})$.

- To derive a DRR Mealy machine from $\mathcal{M}$, we add a **new initial state** m_{new} to the memory state space.
- Using stochastic updates, we return to $\mathcal{M}$ from m_{new} after the first step.
- Transition probabilities use conditional probabilities: some actions may be available only in some initial memory states of $\mathcal{M}$.

9. RRR $\subseteq$ RDR: determinising outputs

For an RRR Mealy machine $\mathcal{M} = (M, \mu_{\mathsf{init}}, \alpha_{\mathsf{up}}, \alpha_{\mathsf{next}})$.

- To derive an RDR Mealy machine from $\mathcal{M}$, we **augment memory states** with pure memoryless strategies $\sigma_i\colon S_i \to A$.
- We use stochastic initialisation and updates to integrate the randomisation over actions in the transitions.

$\rightsquigarrow$ We can avoid a blow-up by $|A|^{|S_i|}$ as follows.

Example: choosing $|A| \cdot |S_i|$ pure memoryless strategies

For $m \in M$, we represent the probability intervals of actions in each state: we get the pure memoryless strategies as segments.

	$[0, \frac{1}{3}]$	$[\frac{1}{3}, \frac{1}{2}]$	$[\frac{1}{2}, \frac{2}{3}]$	$[\frac{2}{3}, 1]$
s_1	a_1		a_2	
s_2	a_1	a_2		a_3
s_3	a_1	a_2	a_3	
σ_i	σ_1	σ_2	σ_3	σ_4

0 $\frac{1}{3}$ $\frac{1}{2}$ $\frac{2}{3}$ 1

References

[Aum16] Robert J . Aumann. *28. Mixed and Behavior Strategies in Infinite Extensive Games*, pages 627–650. Princeton University Press, 2016.

James C. A. Main
Joint work with Mickael Randour

[MR22] Different Strokes in Randomised Strategies: Revisiting Kuhn's Theorem with Finite-memory Assumptions, *Submitted*