The Many Faces of Strategy Complexity

James C. A. Main[1]
Based on joint work with
Michal Ajdarów[3]   Petr Novotný[3]   Mickaël Randour[2,1]

[1]UMONS – Université de Mons, Belgium

[2]F.R.S.-FNRS, Belgium
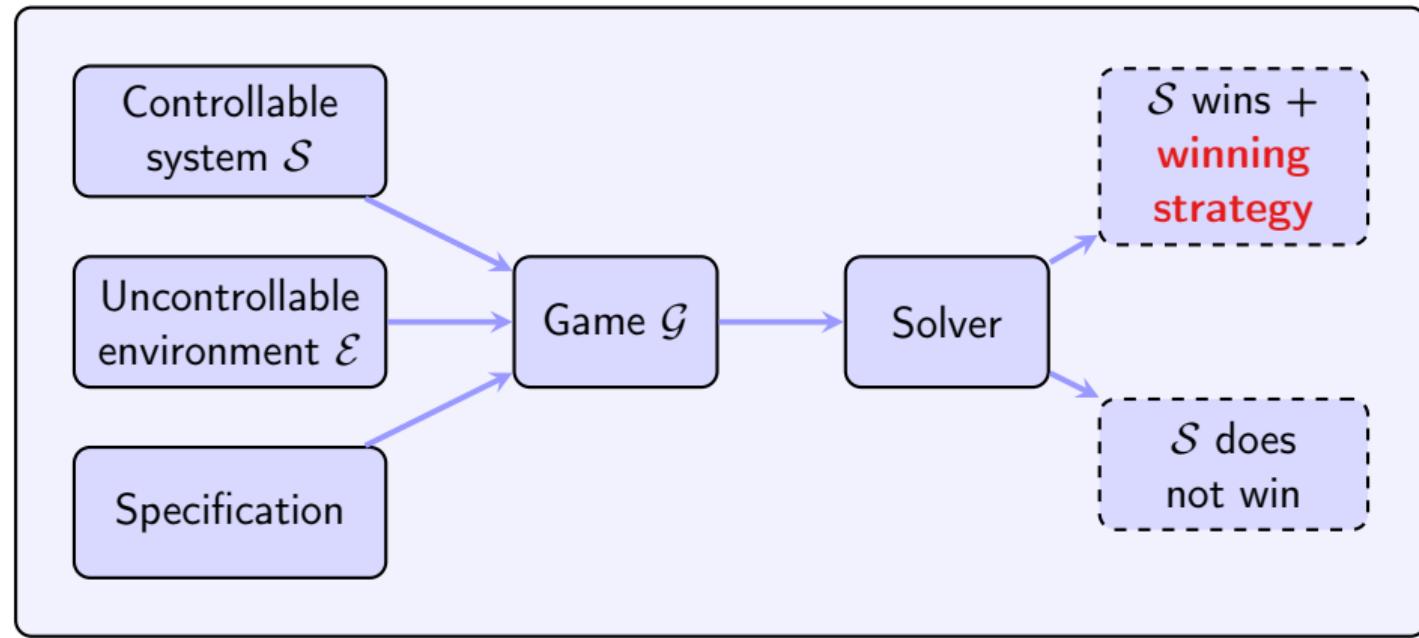
[3]Masaryk University, Brno, Czech Republic

UMONS

Institute of Science and Technology Austria – December 2, 2025

Reactive synthesis via games on graphs

We study **games on graphs** for **reactive synthesis**.

Overview of the talk

Strategies are the formal counterpart of **controllers for reactive systems**.

We are interested in **simple strategies** to obtain simple controllers.

What is a simple strategy?

Strategy complexity is **multifaceted**.
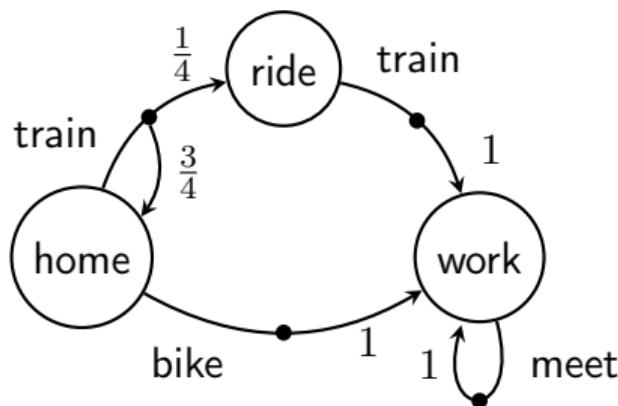
We mainly focus on three directions:
- **finite-memory** strategies;
- **randomised** strategies;
- **alternative** representations of strategies.

Table of contents

Markov decision processes

Markov decision process (MDP)

An MDP is a tuple $\mathcal{M} = (S, A, \delta)$ where

- S is a countable **state space**;
- A is a countable **action space**;
- $\delta \colon S \times A \to \mathcal{D}(S)$ is a **transition** function.

Play: sequence in $(SA)^\omega$ coherent with δ.
Ex.: home train home bike (work meet)$^\omega$
History: prefix of a play ending in a state.

Strategies

Non-determinism in games is resolved through **strategies**.

Pure strategies

A **pure strategy** is a function $\sigma \colon \mathsf{Hist}(\mathcal{M}) \to A$.

A **memoryless strategy** only looks at the current state.

When fixing a **strategy** σ and an initial state s, we obtain an induced **Markov chain**.

- **Probability notation**: $\mathbb{P}_s^\sigma$.
- **Expectation notation**: $\mathbb{E}_s^\sigma$.
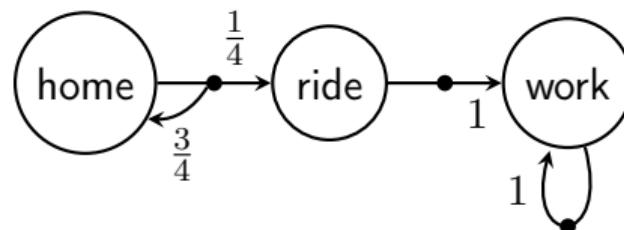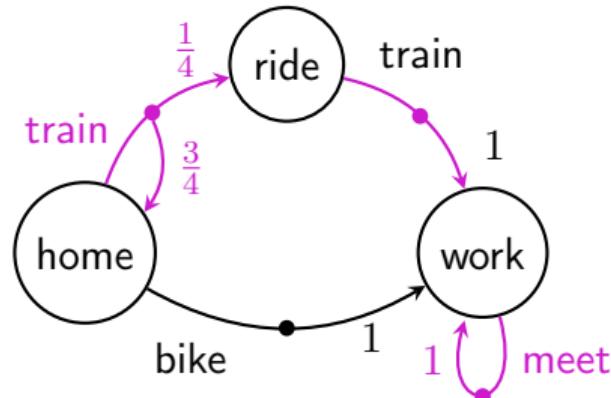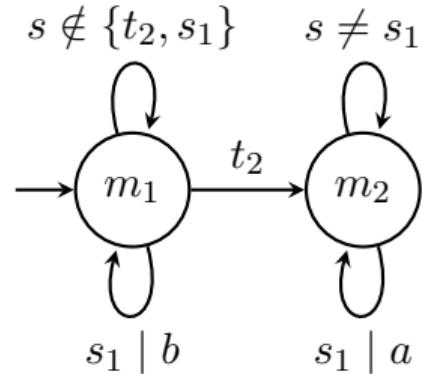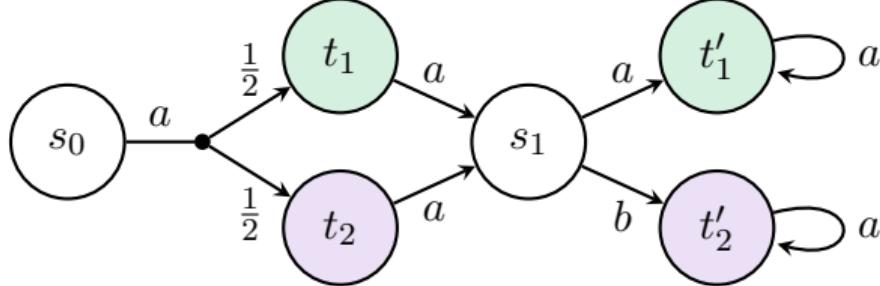
Table of contents

Finite-memory strategies

Representation of pure strategies via Mealy machines

- Set of **memory states** M;
- initial **memory state** m_{init};
- **next-move** function $\text{nxt}_{\mathcal{M}} \colon M \times S \to A$;
- memory **update** function $\text{up}_{\mathcal{M}} \colon M \times S \times A \to M$.

The study of finite memory

The complexity of strategies is traditionally measured by the size of their **memory**.

Key questions for finite-memory strategies

When does finite memory suffice?
⤳ Characterisations of specifications for which finite-memory suffices (e.g., [GZ05; Bou+22]).

How much memory do we need to play optimally?
⤳ Computing memory bounds [Bou+23; CO25].
⤳ Establishing improved bounds (e.g.,[JLS15; Mai24]).

Can we improve memory requirements by considering more general strategies?
⤳ Trading memory for **randomness** (e.g., [CdH04; CRR14]).
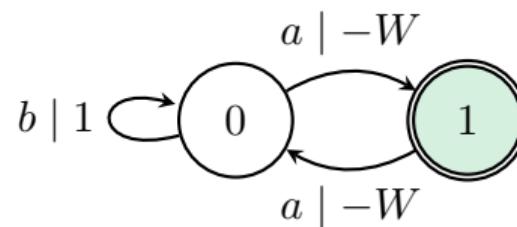
Memory does not tell the whole story (1/2)

Counter-based strategies

Memory does **not fully reflect** the complexity of a strategy.
Consider a game with an **energy-Büchi** objective [CD12], where $W \in \mathbb{N}$.
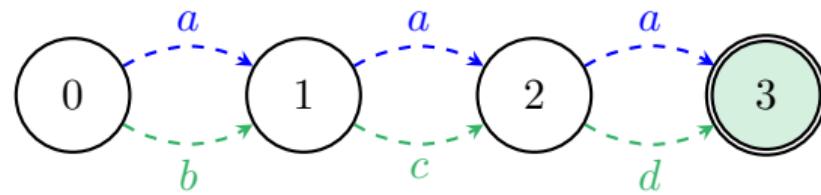
Need memory **exponential** in the binary encoding of W to satisfy the energy-Büchi objective.

Polynomial representation with a **counter**-based approach.

Memory does not tell the whole story (2/2)
Action choices influence simplicity

Memory does **not fully reflect** the complexity of a strategy.
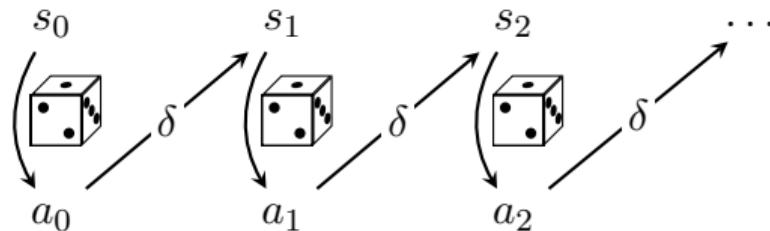
$\rightarrow$ Strategy σ_1 is **simpler to represent** than σ_2

The **action choices** can impact how concise the strategy can be made.

Table of contents
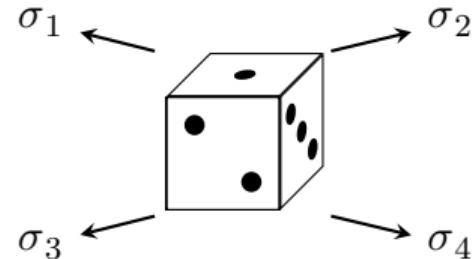
What is a randomised strategy?

Behavioural strategy

$\sigma \colon \mathsf{Hist}(\mathcal{M}) \to \mathcal{D}(A)$

Mixed strategy

$\mathcal{D}(\sigma \colon \mathsf{Hist}(\mathcal{M}) \to A)$

How do these two classes of strategies compare?

Kuhn's theorem: same expressiveness when perfect recall holds.

What about finite-memory strategies?

Components of Mealy machines for **pure** strategies

- Initial **memory state** m_{init};
- **next-move** function $\mathsf{nxt}_{\mathcal{M}} \colon M \times S \to A$;
- memory **update** function $\mathsf{up}_{\mathcal{M}} \colon M \times S \times A \to M$.

How can we **extend** Mealy machines to model **randomised strategies?**

Stochastic Mealy machines – behavioural version

- Initial **memory state** m_{init};
- **randomised next-move** function $\mathsf{nxt}_{\mathcal{M}} \colon M \times S \to \mathcal{D}(A)$;
- memory **update** function $\mathsf{up}_{\mathcal{M}} \colon M \times S \times A \to M$.

What about finite-memory strategies?

Components of Mealy machines for **pure** strategies

- Initial **memory state** m_{init};
- **next-move** function $\text{nxt}_{\mathcal{M}} \colon M \times S \to A$;
- memory **update** function $\text{up}_{\mathcal{M}} \colon M \times S \times A \to M$.

How can we **extend** Mealy machines to model **randomised strategies?**

Stochastic Mealy machines – mixed version

- Initial **memory distribution** $\mu_{\text{init}} \in \mathcal{D}(M)$;
- **next-move** function $\text{nxt}_{\mathcal{M}} \colon M \times S \to A$;
- memory **update** function $\text{up}_{\mathcal{M}} \colon M \times S \times A \to M$.

What about finite-memory strategies?

Components of Mealy machines for **pure** strategies

- Initial **memory state** m_{init};
- **next-move** function $\mathsf{nxt}_{\mathcal{M}} \colon M \times S \to A$;
- memory **update** function $\mathsf{up}_{\mathcal{M}} \colon M \times S \times A \to M$.

How can we **extend** Mealy machines to model **randomised strategies?**

Stochastic Mealy machines – full randomisation

- Initial **memory distribution** $\mu_{\mathsf{init}} \in \mathcal{D}(M)$;
- **randomised next-move** function $\mathsf{nxt}_{\mathcal{M}} \colon M \times S \to \mathcal{D}(A)$;
- **randomised** memory **update** function $\mathsf{up}_{\mathcal{M}} \colon M \times S \times A \to \mathcal{D}(M)$.

Randomisation and finite memory [MR24]

Acronyms **XYZ** where X, Y, Z $\in \{D, R\}$ and D = deterministic and R = random, and

- X characterises initialisation,
- Y characterises the next-move function,
- Z characterises updates.
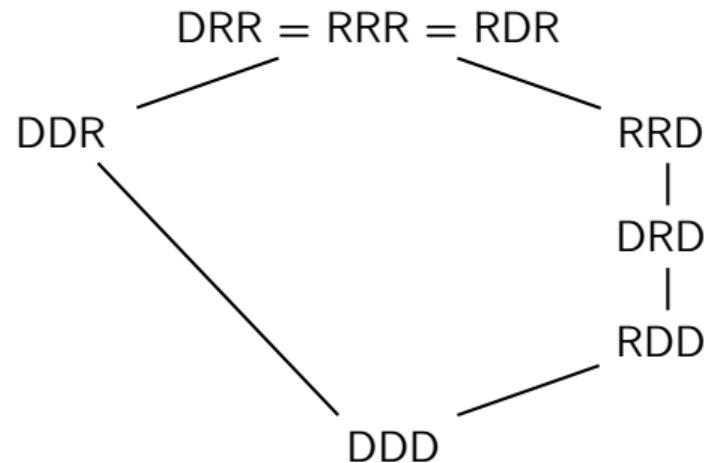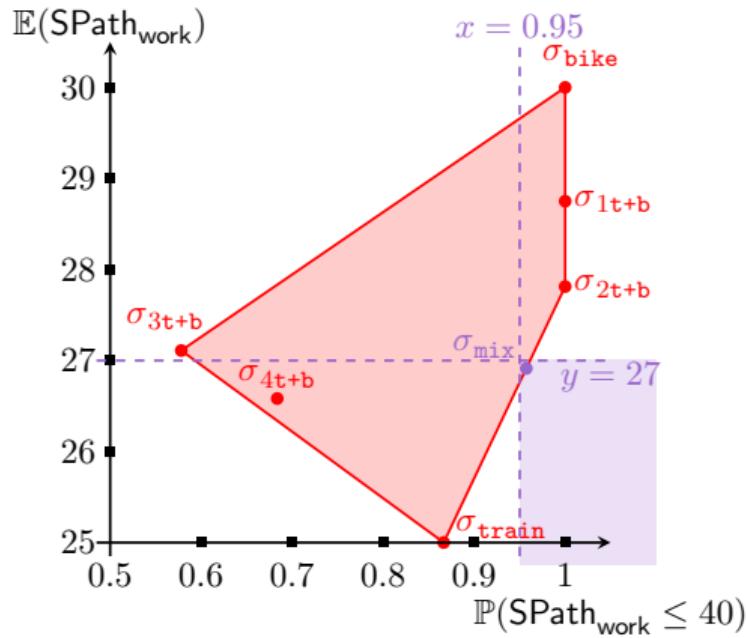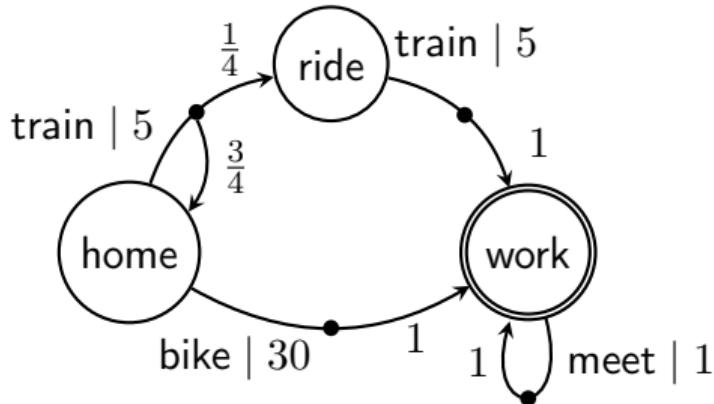
```
                    DRR = RRR = RDR
               DDR                    RRD
                                       |
                                      DRD
                                       |
                                      RDD

                         DDD
```

Table of contents

Randomisation and multiple objectives

Randomisation can be used to **balance multiple goals**. For instance:
- reaching work under 40 minutes with **high probability**;
- minimising the **expected** time to reach work.

Randomisation and multiple objectives

In **multi-objective MDPs**, **randomised strategies** may be necessary for some specifications.

Main questions

- What **type of randomisation** do we need for multi-objective queries?
- What is the relationship between expected payoffs of **pure strategies** and expected payoffs of **general strategies**?

Applicability of our results

A **payoff** is a measurable function $f \colon \mathrm{Plays}(\mathcal{M}) \to \bar{\mathbb{R}}$.

We want results that apply to a **broad class of payoffs**.

Which payoffs f do we consider?

- A payoff f is **good** if it has a **well-defined expectation** under all strategies from all initial states.
- A payoff f is **universally integrable** if its expectation is **finite** under all strategies from all initial states.

For a **multi-dimensional payoff** $\bar{f} = (f_1, \ldots, f_d)$ and $s \in S$, we study:

- $\mathrm{Pay}_s(\bar{f}) = \{\mathbb{E}_s^\sigma(\bar{f}) \mid \sigma \text{ strategy}\}$;
- $\mathrm{Pay}_s^{\mathsf{pure}}(\bar{f}) = \{\mathbb{E}_s^\sigma(\bar{f}) \mid \sigma \text{ pure strategy}\}$.

Universally integrable payoffs

Theorem (M., Randour, 2025)

Let $\bar{f}$ be **universally integrable**. Then for all $s \in S$,

$$\mathsf{Pay}_s(\bar{f}) = \mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})).$$

Proof idea: reasoning on **lexicographic multi-objective MDPs**.

Lemma (M., Randour, 2025)

If $\bar{f}$ is **universally integrable**, then for all strategies σ, there exists a **pure strategy** τ such that $\mathbb{E}_s^\sigma(\bar{f}) \leq_{\mathsf{lex}} \mathbb{E}_s^\tau(\bar{f})$.

By reducing to **one-dimensional MDPs**, we can prove that

$$\mathrm{cl}(\mathsf{Pay}_s(\bar{f})) = \mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f}))).$$

Mixing for universally integrable payoffs
Proof of the weaker result

Let $\bar{f}$ be universally integrable and $s \in S$.

Goal: show that $\mathsf{Pay}_s(\bar{f}) \subseteq \mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})))$.

Fix a strategy σ and let $\mathbf{q} = \mathbb{E}_s^\sigma(\bar{f})$.

Proof by contradiction.
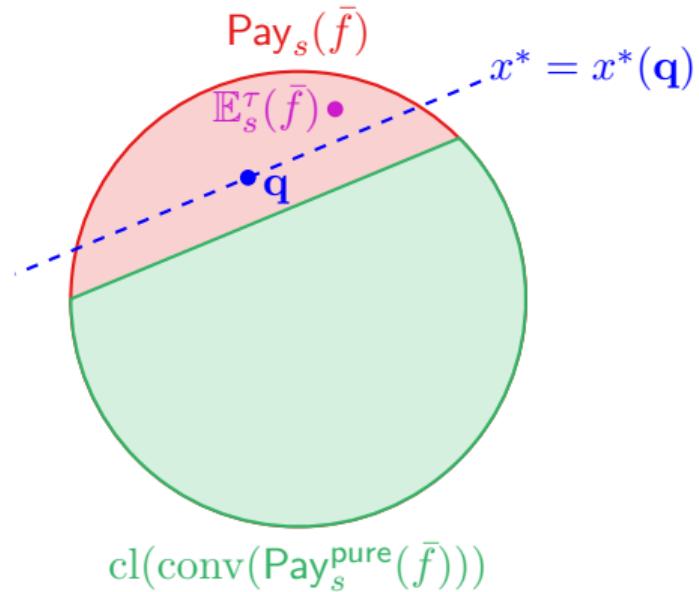
Assume $\mathbf{q} \in \mathsf{Pay}_s(\bar{f}) \setminus \mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})))$.

Separate $\mathbf{q}$ and $\mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})))$ with x^*.

By the Lemma, there is a **pure strategy** τ such that
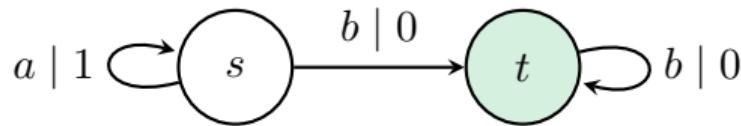
$$x^*(\mathbf{q}) = \mathbb{E}_s^\sigma(x^* \circ \bar{f}) \leq \mathbb{E}_s^\tau(x^* \circ \bar{f}).$$

This contradicts the fact that x^* is separating. $\square$

Beyond universally integrable payoffs

**What if $\bar{f}$ is not universally integrable?**

$$f(\pi) = \begin{cases} k & \text{if } \pi = (sa)^k s(bt)^\omega \\ 0 & \text{otherwise.} \end{cases}$$

The theorem for universally integrable payoffs does not generalise:

- $\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f}) = \mathbb{N} \implies \mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})) = [0, +\infty[,$
- $+\infty \in \mathsf{Pay}_s(\bar{f}).$

Other results

What can we say about good payoffs in general?

Theorem (M., Randour, 2025)

*Let $\bar{f} = (f_1, \ldots, f_d)$ be a **good payoff** and $s \in S$. Then*

$$\mathrm{cl}(\mathsf{Pay}_s(\bar{f})) = \mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f}))).$$

How many strategies do we have to mix?

Theorem (M., Randour, 2025)

- *Payoffs of **finite-support mixed strategies** can be **obtained** by **mixing** $d+1$ **strategies**.*
- *Payoffs of **finite-support mixed strategies** can be **dominated** by **mixing** d **strategies**.*

Table of contents

Memoryless strategies in one-counter MDPs

We study **one-counter Markov decision processes**.

We consider counter-based strategies with a **compact representation** that we call **interval strategies**.

Our contribution (Ajdarów, M., Novotný, Randour, ICALP 2025)
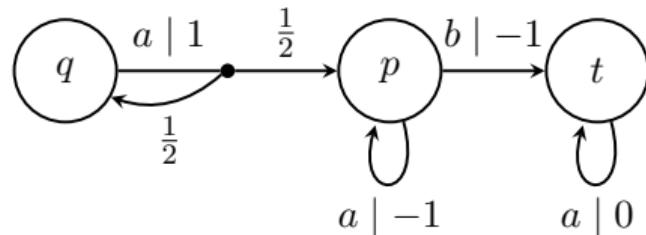
- PSPACE **verification** algorithms for interval strategies.
- PSPACE **realisability** algorithms for **structurally-constrained** interval strategies.

Our algorithms are based on a **finite abstraction** of an **infinite system**.

One-counter Markov decision processes
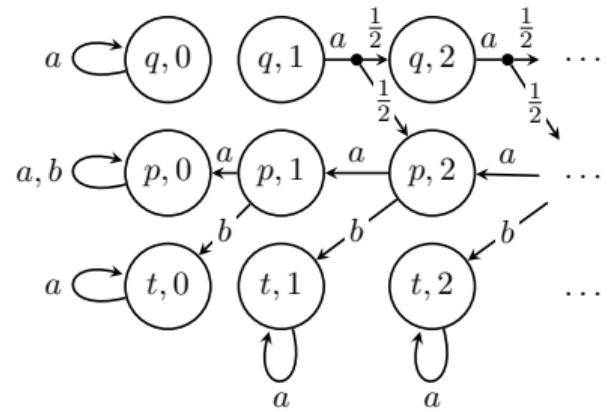
One-counter MDP (OC-MDP) $\mathcal{Q}$

- **Finite** MDP (Q, A, δ).
- **Weight function**
  $w \colon Q \times A \to \{-1, 0, 1\}$.
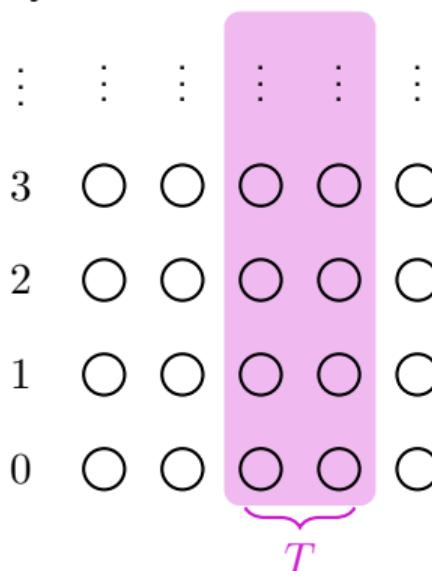
MDP $\mathcal{M}^{\leq \infty}(\mathcal{Q})$ induced by $\mathcal{Q}$

- **Countable** MDP over $S = Q \times \mathbb{N}$.
- State transitions via δ.
- Counter updates via w.
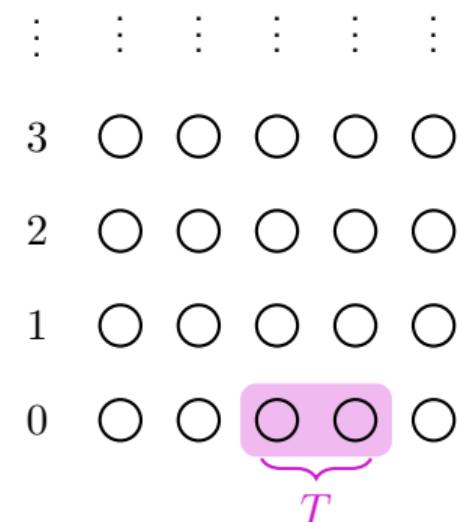
Objectives

- An **objective** is a measurable set of plays.
- Let $T \subseteq Q$ **be a target**.
- We study variants of **reachability objectives**.

State reachability $\mathrm{Reach}(T)$

Selective termination $\mathrm{Term}(T)$

Interval strategies

We study a restricted class of **memoryless strategies** of $\mathcal{M}^{\leq\infty}(\mathcal{Q})$.

## Open-ended interval strategies (OEIS)

σ is an OEIS if $\exists\, k_0 \in \mathbb{N}$ s.t. $\forall\, q \in Q$ and $\forall\, k \geq k_0$, $\sigma(q, k) = \sigma(q, k_0)$.

$\mathbb{N}_0$	1	2	$\cdots$	$k_0 - 1$	k_0	$k_0 + 1$	$\cdots$
Q	σ_1	σ_2	$\cdots$	σ_{k_0-1}	σ_{k_0}	σ_{k_0}	$\cdots$

Group counter values in intervals

constant

Inter.	I_1	I_2	$\cdots$	$I_d = [\![k_0, \infty]\!]$
Q	τ_1	τ_2	$\cdots$	$\tau_d = \sigma_{k_0}$

$\rightsquigarrow$ **Finite partition** of $\mathbb{N}_0$ into **intervals**

Verification of interval strategies

Verification problem. When following a given interval strategy, do we **reach a target state** with probability greater than or equal to some given threshold?

Challenges

- **Infinite** Markov chain.
- Compressed Markov chains have **irrational** or **very precise** probabilities.

Solutions

- **Compression** to finite Markov chain.
- Transition probabilities can be represented by small **logical formulae**.

Algorithm
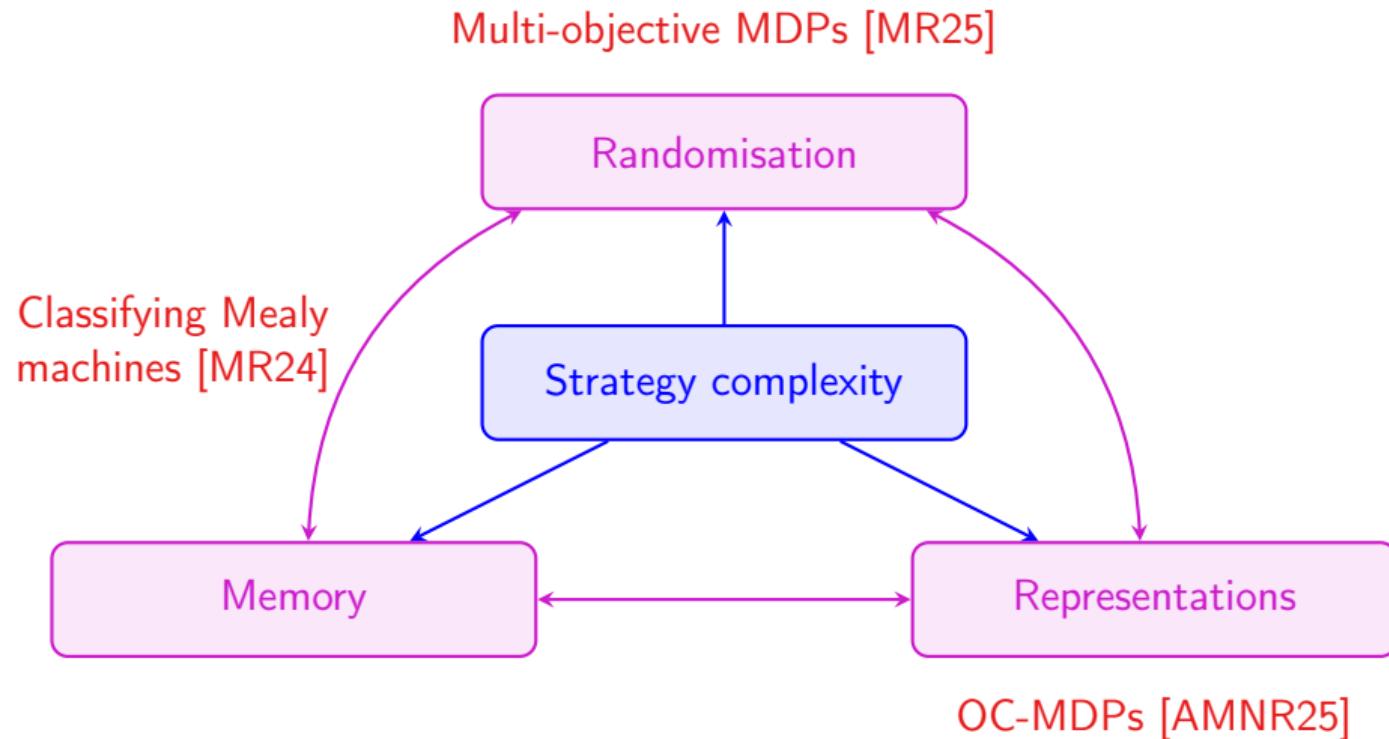
Construct a **universal logical formula** and check if it is satisfied in the **theory of the reals**.

We have also built on these logical formulae to design **synthesis algorithms**.

Table of contents

Conclusion

Multi-objective MDPs [MR25]

Randomisation

Classifying Mealy
machines [MR24]

Strategy complexity

Memory

Representations

OC-MDPs [AMNR25]

References I

[Ajd+25]   Michal Ajdarów et al. "Taming Infinity one Chunk at a Time: Concisely Represented Strategies in One-Counter MDPs". In: *Proceedings of the 52th International Colloquium on Automata, Languages, and Programming, ICALP 2025, July 8–11, 2025, Aarhus, Denmark*. Ed. by Keren Censor-Hillel et al. Vol. 334. LIPIcs. Schloss Dagstuhl –Leibniz-Zentrum für Informatik, 2025.

[Bou+22]   Patricia Bouyer et al. "Games Where You Can Play Optimally with Arena-Independent Finite Memory". In: *Logical Methods in Computer Science* 18.1 (2022). DOI: `10.46298/lmcs-18(1:11)2022`.

References II

[Bou+23]   Patricia Bouyer et al. "How to Play Optimally for Regular Objectives?" In: *Proceedings of the 50th International Colloquium on Automata, Languages, and Programming, ICALP 2023, July 10–14, 2023, Paderborn, Germany*. Ed. by Kousha Etessami, Uriel Feige, and Gabriele Puppis. Vol. 261. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2023, 118:1–118:18. DOI: `10.4230/LIPIcs.ICALP.2023.118`.

[CD12]   Krishnendu Chatterjee and Laurent Doyen. "Energy parity games". In: *Theoretical Computer Science* 458 (2012), pp. 49–60. DOI: `10.1016/j.tcs.2012.07.038`. URL: `https://doi.org/10.1016/j.tcs.2012.07.038`.

References III

[CdH04] Krishnendu Chatterjee, Luca de Alfaro, and Thomas A. Henzinger. "Trading Memory for Randomness". In: *Proceedings of the 1st International Conference on Quantitative Evaluation of Systems, QEST 2004, Enschede, The Netherlands, 27–30 September 2004*. IEEE Computer Society, 2004, pp. 206–217. DOI: 10.1109/QEST.2004.1348035.

[CO25] Antonio Casares and Pierre Ohlmann. "The Memory of ω-Regular and BC(Σ_2^0) Objectives". In: *Proceedings of the 52nd International Colloquium on Automata, Languages, and Programming, ICALP 2025, July 8–11, 2025, Aarhus, Denmark*. Ed. by Keren Censor-Hillel et al. Vol. 334. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2025, 149:1–149:18. DOI: 10.4230/LIPICS.ICALP.2025.149. URL: https://doi.org/10.4230/LIPIcs.ICALP.2025.149.

References IV

[CRR14] Krishnendu Chatterjee, Mickael Randour, and Jean-François Raskin. "Strategy synthesis for multi-dimensional quantitative objectives". In: *Acta Informatica* 51.3-4 (2014), pp. 129–163. DOI: `10.1007/s00236-013-0182-6`.

[EWY10] Kousha Etessami, Dominik Wojtczak, and Mihalis Yannakakis. "Quasi-Birth-Death Processes, Tree-Like QBDs, Probabilistic 1-Counter Automata, and Pushdown Systems". In: *Performance Evaluation* 67.9 (2010), pp. 837–857. DOI: `10.1016/J.PEVA.2009.12.009`. URL: `https://doi.org/10.1016/j.peva.2009.12.009`.

[GZ05] Hugo Gimbert and Wiesław Zielonka. "Games Where You Can Play Optimally Without Any Memory". In: *Proceedings of the 16th International Conference on Concurrency Theory, CONCUR 2005, San Francisco, CA, USA, August 23–26, 2005.* Ed. by Martín Abadi and Luca de Alfaro. Vol. 3653. Lecture Notes in Computer Science. Springer, 2005, pp. 428–442. ISBN: 3-540-28309-9. DOI: `10.1007/11539452_33`.

References V

[JLS15]   Marcin Jurdzinski, Ranko Lazic, and Sylvain Schmitz. "Fixed-Dimensional Energy Games are in Pseudo-Polynomial Time". In: *Proceedings (Part II) of the 42nd International Colloquium on Automata, Languages, and Programming, ICALP 2015, Kyoto, Japan, July 6–10, 2015*. Ed. by Magnús M. Halldórsson et al. Vol. 9135. Lecture Notes in Computer Science. Springer, 2015, pp. 260–272. DOI: `10.1007/978-3-662-47666-6\_21`. URL: `https://doi.org/10.1007/978-3-662-47666-6\_21`.

[Mai24]   James C. A. Main. "Arena-Independent Memory Bounds for Nash Equilibria in Reachability Games". In: *41st International Symposium on Theoretical Aspects of Computer Science, STACS 2024, March 12-14, 2024, Clermont-Ferrand, France*. Ed. by Olaf Beyersdorff et al. Vol. 289. LIPIcs. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2024, 50:1–50:18. DOI: `10.4230/LIPICS.STACS.2024.50`. URL: `https://doi.org/10.4230/LIPIcs.STACS.2024.50`.

References VI

[MR24]   James C. A. Main and Mickael Randour. "Different strokes in randomised strategies: Revisiting Kuhn's theorem under finite-memory assumptions". In: *Information and Computation* 301 (2024), p. 105229. DOI: `10.1016/J.IC.2024.105229`. URL: `https://doi.org/10.1016/j.ic.2024.105229`.

[MR25]   James C. A. Main and Mickael Randour. "Mixing Any Cocktail with Limited Ingredients: On the Structure of Payoff Sets in Multi-Objective MDPs and its Impact on Randomised Strategies". In: *CoRR* abs/2502.18296 (2025). DOI: `10.48550/arXiv.2502.18296`.

Mixing for universally integrable payoffs

Proof

Let $\bar{f}$ be universally integrable and $s \in S$.

Goal: show that $\mathsf{Pay}_s(\bar{f}) \subseteq \mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f}))$.
Fix a strategy σ and $\mathbf{q} = \mathbb{E}_s^\sigma(\bar{f})$.
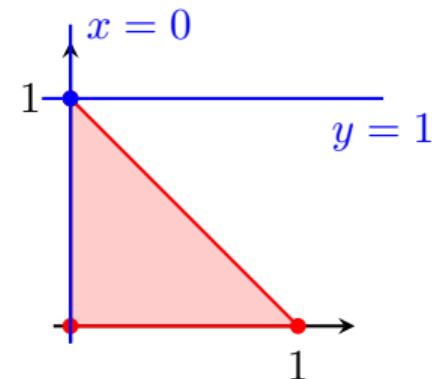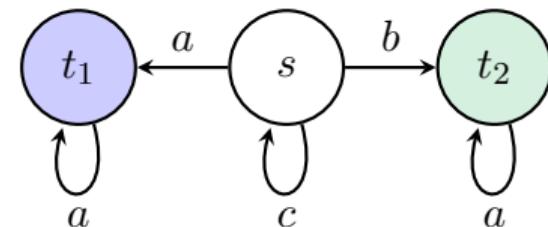
Step 1: **isolate q** as much as possible with an intersection of **supporting hyperplanes**.

Example 1: $\mathbf{q} = (0, 1)$.

- First hyperplane: $x = 0 \rightsquigarrow x_1^*(x, y) = -x$.
- Second hyperplane: $y = 1 \rightsquigarrow x_2^*(x, y) = y$

σ is **lexicographically optimal** for $(x_1^*, x_2^*) \circ \bar{f}$
$\quad \implies \mathbf{q} \in \mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})$.

$f_1 = \mathbb{1}_{\mathsf{Reach}(t_1)}$ $f_2 = \mathbb{1}_{\mathsf{Reach}(t_2)}$

Mixing for universally integrable payoffs
Proof

Let $\bar{f}$ be universally integrable and $s \in S$.

Goal: show that $\mathsf{Pay}_s(\bar{f}) \subseteq \mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f}))$.
Fix a strategy σ and $\mathbf{q} = \mathbb{E}_s^{\sigma}(\bar{f})$.
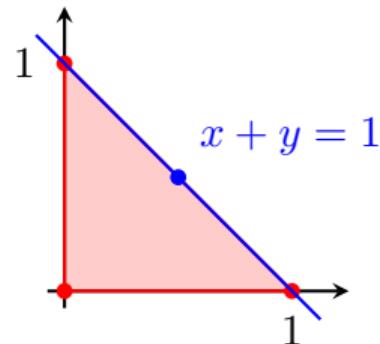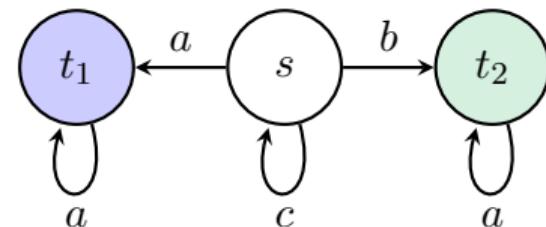
Step 1: **isolate** $\mathbf{q}$ as much as possible with an intersection of **supporting hyperplanes**.

Example 2: $\mathbf{q} = (\frac{1}{2}, \frac{1}{2})$.

We construct $L_{\mathbf{q}}$ **linear** such that:

- σ **lexicographically optimal** from s for $L_{\mathbf{q}} \circ \bar{f}$;
- $\mathbf{q} \in \mathrm{ri}(\mathsf{Pay}_s(\bar{f}) \cap V))$ for $V = L_{\mathbf{q}}^{-1}(L_{\mathbf{q}}(\mathbf{q}))$

$f_1 = \mathbb{1}_{\mathsf{Reach}(t_1)}$ $f_2 = \mathbb{1}_{\mathsf{Reach}(t_2)}$

$x + y = 1$

Mixing for universally integrable payoffs
Proof – continued

Goal: $\mathbf{q} \in \mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f}))$.

Step 2: it suffices to prove:
$\mathrm{cl}(\mathsf{Pay}_s(\bar{f}) \cap V) = \mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})) \cap V)$.
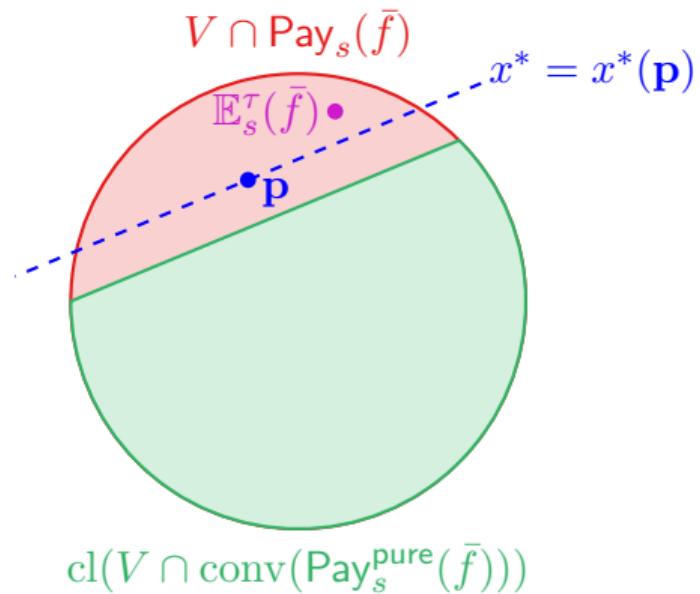
Proof by contradiction.
Let $\mathbf{p} \in \mathsf{Pay}_s(\bar{f}) \cap V \setminus \mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})) \cap V)$.

Separate $\mathbf{p}$ and $\mathrm{cl}(\mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})) \cap V)$ with x^*.

There is a **pure strategy** τ such that
$$\mathbb{E}_s^\tau((L_{\mathbf{q}}, x^*) \circ \bar{f}) \geq_{\mathsf{lex}} (L_{\mathbf{q}}(\mathbf{p}), x^*(\mathbf{p})).$$

$\implies x^*(\mathbb{E}_s^\tau(\bar{f})) \geq x^*(\mathbf{p})$ (**contradiction**).

$V \cap \mathsf{Pay}_s(\bar{f})$

$x^* = x^*(\mathbf{p})$

$\mathbb{E}_s^\tau(\bar{f})$

$\mathbf{p}$
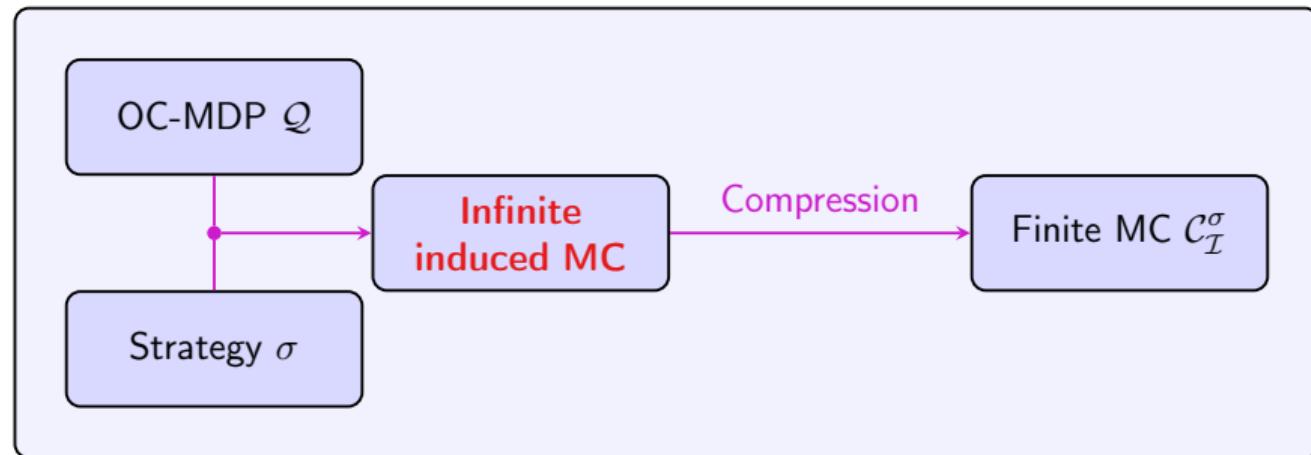
$\mathrm{cl}(V \cap \mathrm{conv}(\mathsf{Pay}_s^{\mathsf{pure}}(\bar{f})))$

Verification

Interval strategy verification problem

Given an **interval strategy** σ, **an objective** $\Omega \in \{\text{Reach}(T), \text{Term}(T)\}$, a **threshold** $\alpha \in \mathbb{Q} \cap [0, 1]$ and an **initial configuration** $s_{\text{init}} \in Q \times \mathbb{N}$, decide whether $\mathbb{P}^{\sigma}_{\mathcal{M}^{\leq\infty}(\mathcal{Q}), s_{\text{init}}}(\Omega) \geq \alpha$

Interval strategy verification problem

We construct a finite **compressed Markov chain** $\mathcal{C}_{\mathcal{I}}^{\sigma}$.

Solving the verification problem through compressed Markov chains

- To compress, we **keep few configurations** and adjust transitions.
- We have formulae (in the signature $\{0, 1, +, -, \cdot, \leq\}$):
  - $\Phi_{\delta}^{\mathcal{I}}(\mathbf{x}, \mathbf{z}^{\sigma})$ for **transition probabilities** of $\mathcal{C}_{\mathcal{I}}^{\sigma}$;
  - $\Phi_{\Omega}^{\mathcal{I}}(\mathbf{x}, \mathbf{y})$ for **termination probabilities** from configurations of $\mathcal{C}_{\mathcal{I}}^{\sigma}$.

We can solve the verification problem by checking if

$$\mathbb{R} \models \forall \mathbf{x} \forall \mathbf{y} \left(\Phi_{\delta}^{\mathcal{I}}(\mathbf{x}, \mathbf{z}^{\sigma}) \wedge \Phi_{\Omega}^{\mathcal{I}}(\mathbf{x}, \mathbf{y}) \right) \implies y_{s_{\text{init}}} \geq \theta.$$

	Unbounded counter	**Bounded counter**
Upper bound	co-ETR	$\mathsf{P}^{\mathsf{PosSLP}}$
Lower bound	Square-root-sum-hard [EWY10]	Square-root-sum-hard