How to Play Optimally for Regular Objectives?

Patricia Bouyer[1], Nathanaël Fijalkow[2],
Mickael Randour[3], Pierre Vandenhove[1,3]

[1]Université Paris-Saclay, CNRS, ENS Paris-Saclay, LMF, France

[2]CNRS, LaBRI and Université de Bordeaux, France & University of Warsaw, Poland

[3]F.R.S.-FNRS & UMONS – Université de Mons, Belgium

July 11, 2023 – ICALP'23

Outline

Synthesis problem

Synthesizing **controllers** for **reactive systems** with an **objective**.
Systems and their environment modeled with **zero-sum games**.

How to play?

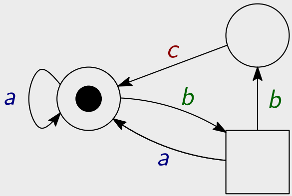
Given an objective, what are the **optimal controllers**?
What are the **smallest** ones?

Results

Characterization of finite-state controllers for *regular objectives*; **computational complexity** of finding small ones.

Games

Zero-sum turn-based games on **graphs**



- $C = \{a, b, c\}$, arena $\mathcal{A} = (V_1, V_2, E)$.
- Two players $\mathcal{P}_1$ ($\bigcirc$) and $\mathcal{P}_2$ ($\square$)
- **Objective** of $\mathcal{P}_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $\mathcal{P}_2$ is $C^\omega \setminus W$.

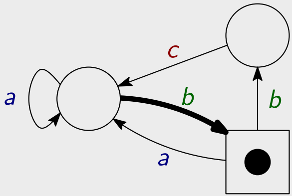
How to play?

A **strategy** of a player is a function $\sigma\colon E^* \to E$.

A strategy σ of $\mathcal{P}_1$ is **winning for** W **from** $v \in V$ if all infinite paths from v *consistent with* σ induce an infinite word in W.

Games

Zero-sum turn-based games on **graphs**



- $C = \{a, b, c\}$, arena $\mathcal{A} = (V_1, V_2, E)$.
- Two players $\mathcal{P}_1$ ($\bigcirc$) and $\mathcal{P}_2$ ($\square$) generate an infinite word $w = b$
- **Objective** of $\mathcal{P}_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $\mathcal{P}_2$ is $C^\omega \setminus W$.

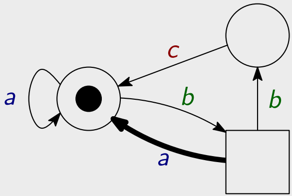
How to play?

A **strategy** of a player is a function $\sigma \colon E^* \to E$.

A strategy σ of $\mathcal{P}_1$ is **winning for** W **from** $v \in V$ if all infinite paths from v *consistent with* σ induce an infinite word in W.

Games

Zero-sum turn-based games on **graphs**



- $C = \{a, b, c\}$, arena $\mathcal{A} = (V_1, V_2, E)$.
- Two players $\mathcal{P}_1$ ($\bigcirc$) and $\mathcal{P}_2$ ($\square$) generate an infinite word $w = ba$
- **Objective** of $\mathcal{P}_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $\mathcal{P}_2$ is $C^\omega \setminus W$.

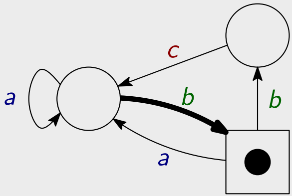
How to play?

A **strategy** of a player is a function $\sigma\colon E^* \to E$.

A strategy σ of $\mathcal{P}_1$ is **winning for** W **from** $v \in V$ if all infinite paths from v *consistent with* σ induce an infinite word in W.

Games

Zero-sum turn-based games on graphs



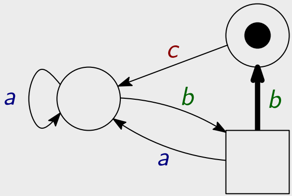
- $C = \{a, b, c\}$, arena $\mathcal{A} = (V_1, V_2, E)$.
- Two players $\mathcal{P}_1$ ($\bigcirc$) and $\mathcal{P}_2$ ($\square$) generate an infinite word $w = bab$
- **Objective** of $\mathcal{P}_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $\mathcal{P}_2$ is $C^\omega \setminus W$.

How to play?

A **strategy** of a player is a function $\sigma \colon E^* \to E$.
A strategy σ of $\mathcal{P}_1$ is **winning for** W **from** $v \in V$ if all infinite paths from v *consistent with* σ induce an infinite word in W.

Games

Zero-sum turn-based games on **graphs**



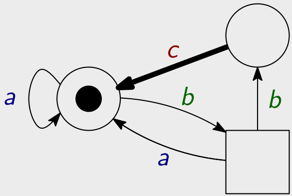
- $C = \{a, b, c\}$, arena $\mathcal{A} = (V_1, V_2, E)$.
- Two players $\mathcal{P}_1$ ($\bigcirc$) and $\mathcal{P}_2$ ($\Box$) generate an infinite word $w = babb$
- **Objective** of $\mathcal{P}_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $\mathcal{P}_2$ is $C^\omega \setminus W$.

How to play?

A **strategy** of a player is a function $\sigma \colon E^* \to E$.
A strategy σ of $\mathcal{P}_1$ is **winning for** W **from** $v \in V$ if all infinite paths from v *consistent with* σ induce an infinite word in W.

Games

Zero-sum turn-based games on graphs



- $C = \{a, b, c\}$, arena $\mathcal{A} = (V_1, V_2, E)$.
- Two players $\mathcal{P}_1$ ($\bigcirc$) and $\mathcal{P}_2$ ($\Box$) generate an infinite word $w = babbc \ldots \in C^\omega$.
- **Objective** of $\mathcal{P}_1$ is a set $W \subseteq C^\omega$.
- **Zero-sum**: objective of $\mathcal{P}_2$ is $C^\omega \setminus W$.

How to play?

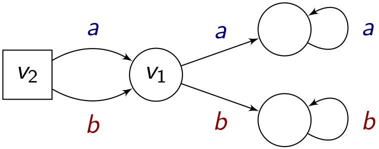
A **strategy** of a player is a function $\sigma \colon E^* \to E$.
A strategy σ of $\mathcal{P}_1$ is **winning for** W **from** $v \in V$ if all infinite paths from v *consistent with* σ induce an infinite word in W.

Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

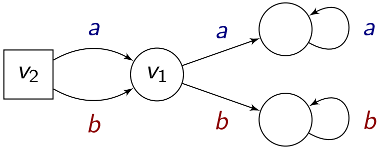


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

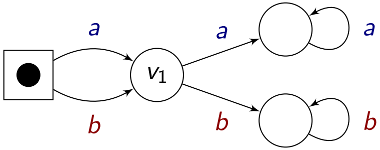


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

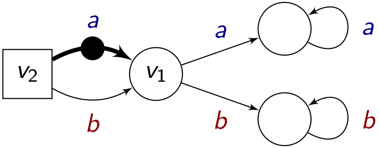


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

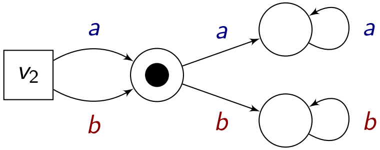


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

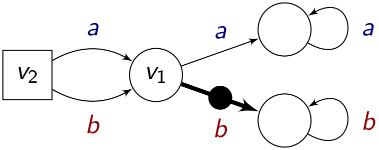


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

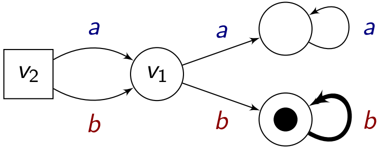


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

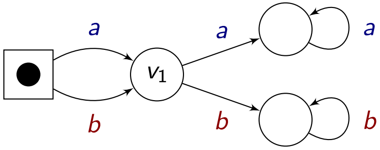


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

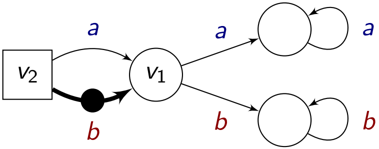


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

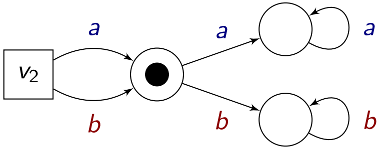


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

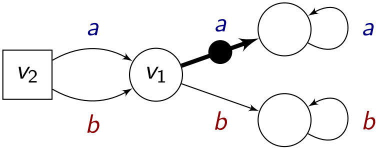


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.

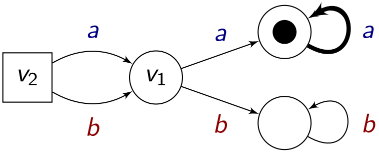


Example of a game

Let $C = \{a, b\}$ and

$$W = \{c_1 c_2 \ldots \in C^\omega \mid \exists i \geq 1, c_i = a \ \wedge \ \exists j \geq 1, c_j = b\}.$$

In the following arena, $\mathcal{P}_1$ has a winning strategy from v_2.



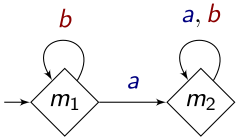
Finite-memory strategy

In general, a strategy $\sigma \colon E^* \to E$ is an **infinite** object.
For synthesis, great if it has a **finite** representation:

Memory structure

Memory structure $(M, m_{\mathsf{init}}, \alpha_{\mathsf{upd}})$: finite set of states M, initial state m_{init}, update function $\alpha_{\mathsf{upd}} \colon M \times C \to M$.

Ex.: remember whether a has already been seen:



Given an arena $\mathcal{A} = (V_1, V_2, E)$: *next-action function* $\alpha_{\mathsf{nxt}} \colon V_i \times M \to E$.

General quest

Quest

- Given an objective, **characterize the memory structures** that suffice **in all arenas** for each player.
- From a representation of an objective, compute **minimal** ones.

Missing piece: **regular** objectives

Well-studied: *Muller conditions*,[1,2] focusing on what is seen infinitely often. We take the **opposite** stance.

Regular objectives

- A **regular reachability objective** is a set LC^ω with $L \subseteq C^*$ regular.
- A **regular safety objective** is a set $C^\omega \setminus LC^\omega$.

- A player wants to **realize** a word in L, the other wants to **prevent** it.
- Expressible as standard **deterministic finite automata**.
- *Special cases of open and closed sets, at the first level of the Borel hierarchy*.

[1] Dziembowski, Jurdziński, and Walukiewicz, "How Much Memory is Needed to Win Infinite Games?", 1997.

[2] Casares, "On the Minimisation of Transition-Based Rabin Automata and the Chromatic Memory Requirements of Muller Conditions", 2022.

Precise quest

Memory requirements of regular objectives

Characterize the memory structures that suffice to make optimal decisions for **regular objectives**. Compute **minimal** ones.

Ideas

- A **DFA recognizing the language** L, seen as a memory structure, always suffices for both players. . .
- . . . but minimal memory structures can be **much smaller**!

In this talk: characterization for regular safety objectives.

Comparing words

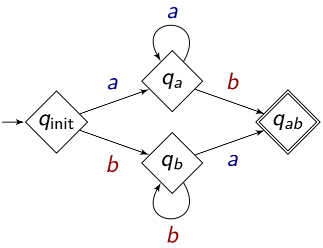
Let $W \subseteq C^\omega$ be an objective.

Preorder on finite words

For $x, y \in C^*$, $x \preceq_W y$ if for all $z \in C^\omega$, $xz \in W \Rightarrow yz \in W$.

$\rightsquigarrow$ y is a better situation than x.

Example: let W be the regular **safety** objective induced by this DFA.



E.g., $a \prec_W \varepsilon$, $ab \prec_W a$,
a and b are incomparable for $\preceq_W$.

Necessary condition for the memory

Let $W \subseteq C^\omega$ be an objective.

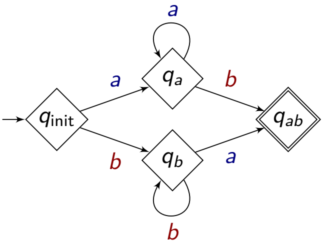
Lemma

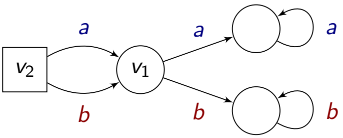
A memory structure $\mathcal{M} = (M, m_{\mathsf{init}}, \alpha_{\mathsf{upd}})$ needs to **distinguish incomparable words**, i.e.,

if $x, y \in C^*$ are incomparable for $\preceq_W$,
then x and y must lead to **different memory states** of $\mathcal{M}$.

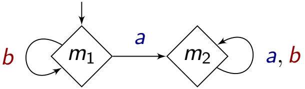
Why is it necessary? Example

Ex.: if a and b are incomparable, they must be distinguished in some arena.





One structure that suffices:



Characterization: safety

Let W be a **regular safety objective**.

Theorem

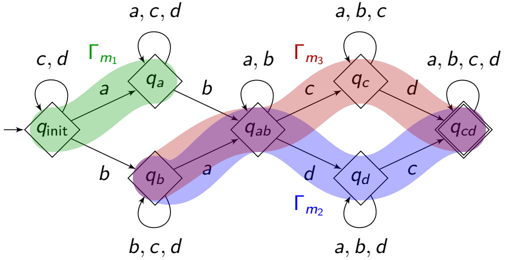
A memory structure $\mathcal{M}$ suffices for winning strategies in all arenas
if and only if
$\mathcal{M}$ distinguishes incomparable words.

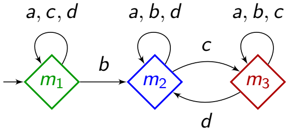
Question

How to find a **smallest** such memory structure?

Automata-theoretic reformulation

$\rightsquigarrow$ Reformulation of "$\mathcal{M}$ distinguishes incomparable prefixes" into a **covering of the automaton states** with **chains**.





Computational complexity: safety

Decision problem

MemorySafe
Input: Automaton $\mathcal{D}$ inducing the regular **safety** objective W and $k \in \mathbb{N}$.
Question: $\exists$ memory structure $\mathcal{M}$ with $\leq k$ states that suffices for W?

Thanks to the covering reformulation (reduction from Hamiltonian cycle):

Theorem

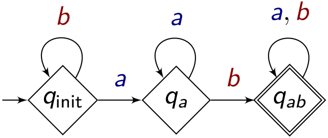
MemorySafe is NP-complete.

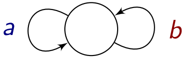
Same problem for regular **reachability** objectives is also NP-complete.

Regular reachability

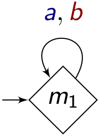
For regular **reachability** objectives, memory structures still need to **distinguish incomparable words**.

But **not sufficient**! E.g., Regular objectives induced by this automaton:

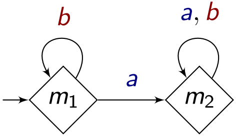




One state suffices for safety:



Two states needed for reachability:

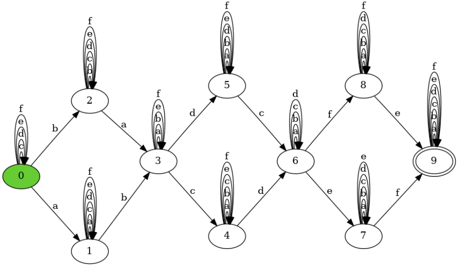


Characterization requires a second property (*not shown here*).

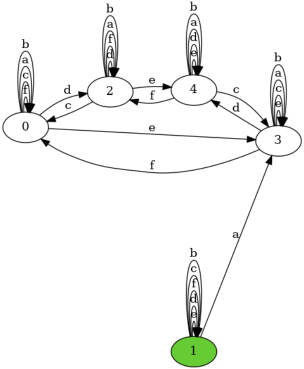
Implementation

Algorithms[3] that find minimal memory structures for regular objectives.

D =



M = memReq.smallest_memory_safety(D) →



[3] https://github.com/pvdhove/regularMemoryRequirements

Conclusion

Future work

- Minimal memory structures for all *ω-**regular** objectives*?
  ✓ Muller conditions,[4,5]
  ✓ deterministic Büchi automata[6] (partially),
  ✓ **regular objectives**.
- Memory model only observes **colors**... but observing **edges** may need fewer memory states.
  Understood for safety,[7] but not for reachability.

Thanks!

.

[4] Dziembowski, Jurdziński, and Walukiewicz, "How Much Memory is Needed to Win Infinite Games?", 1997.

[5] Casares, "On the Minimisation of Transition-Based Rabin Automata and the Chromatic Memory Requirements of Muller Conditions", 2022.

[6] Bouyer, Casares, et al., "Half-Positional Objectives Recognized by Deterministic Büchi Automata", 2022.

[7] Colcombet, Fijalkow, and Horn, "Playing Safe", 2014.