The true colors of memory: A tour of chromatic-memory strategies in zero-sum games on graphs

Patricia Bouyer

Laboratoire Méthodes Formelles
Université Paris-Saclay, CNRS, ENS Paris-Saclay
France

Line of works developed together with Mickael Randour and Pierre Vandenhove.
Some works are co-authors with other people: Antonio Casares, Nathanaël Fijalkow, Stéphane Le Roux, Youssouf Oualhadj.

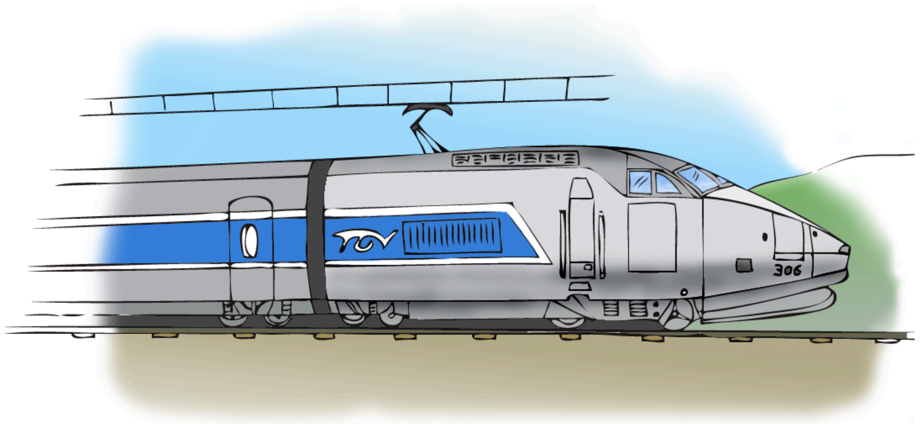
Motivation

—

The setting

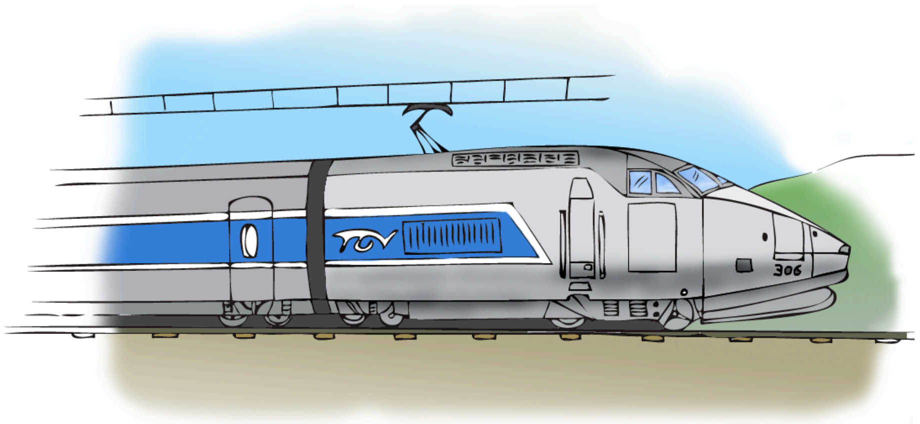
Model-checking

System



Model-checking

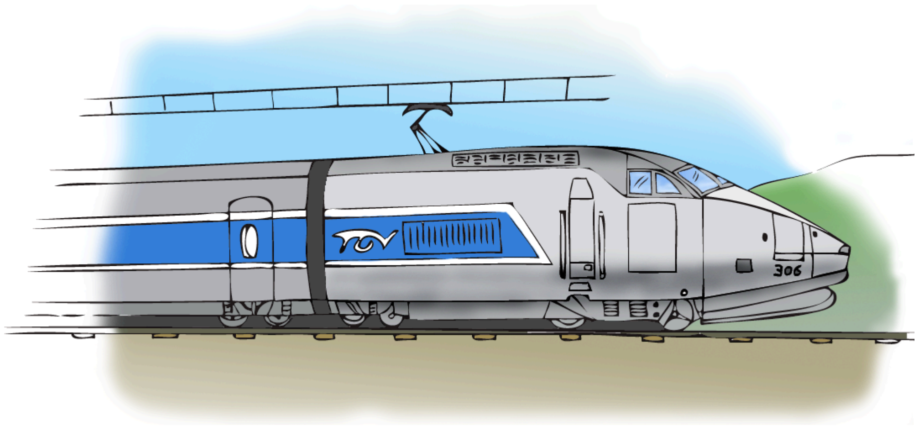
System



Properties

Model-checking

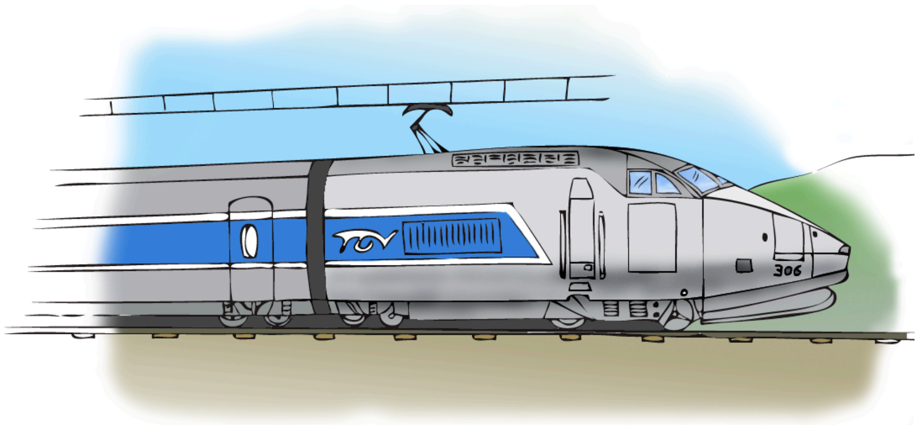
System



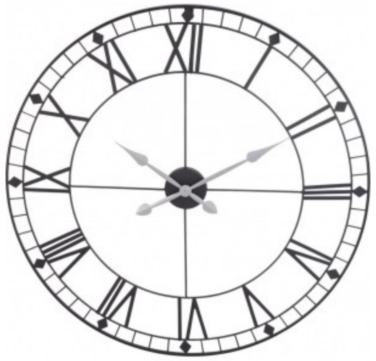
Properties

Model-checking

System

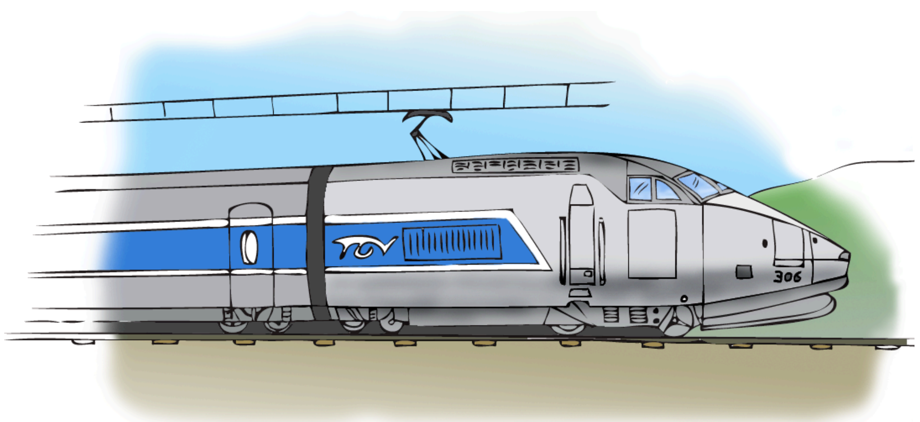


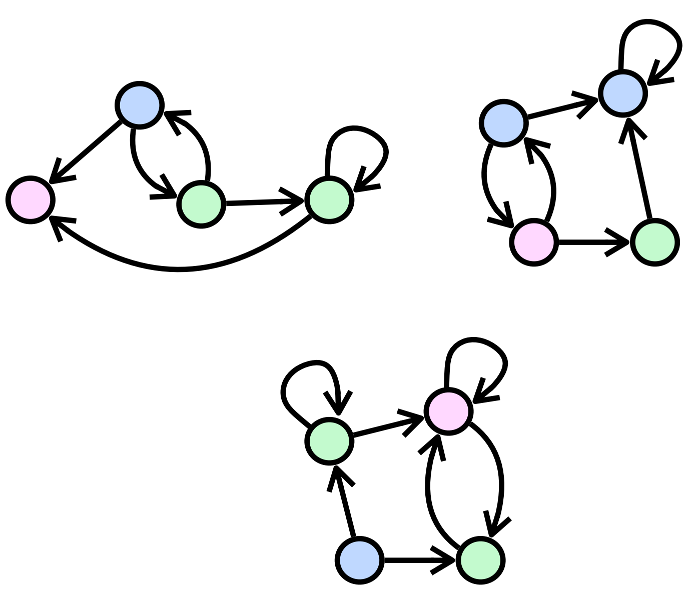
Properties



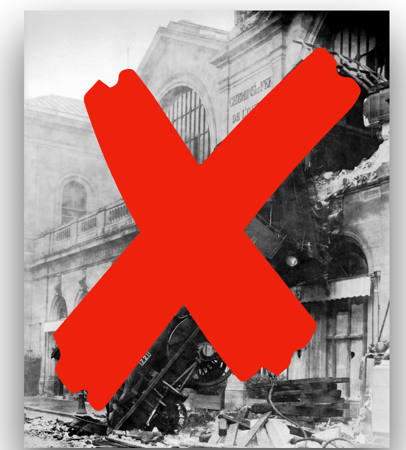
Model-checking

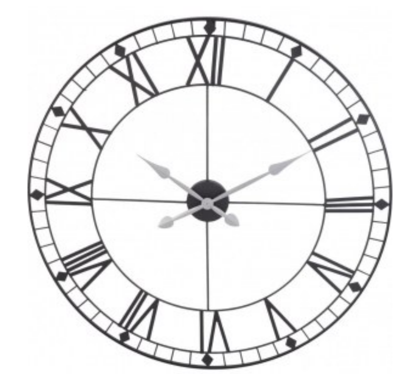
System





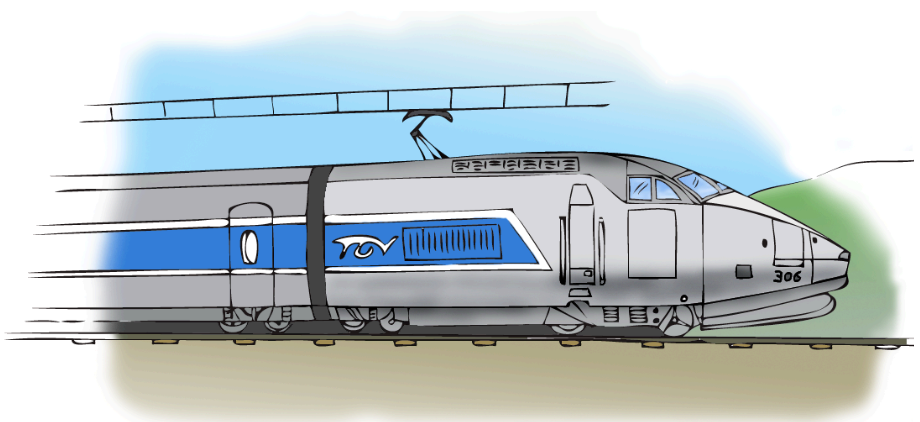
Properties



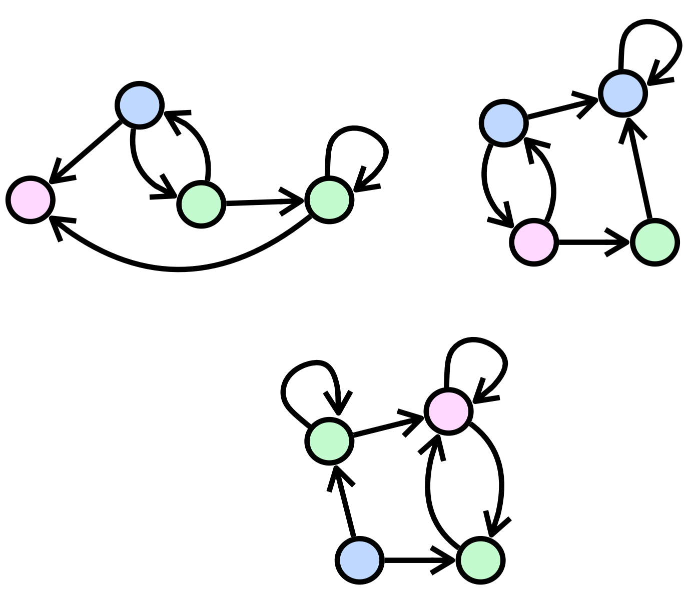


Model-checking

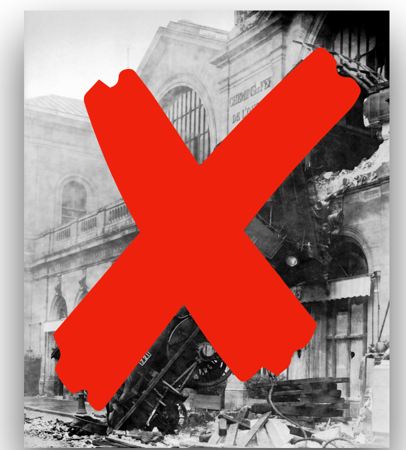
System

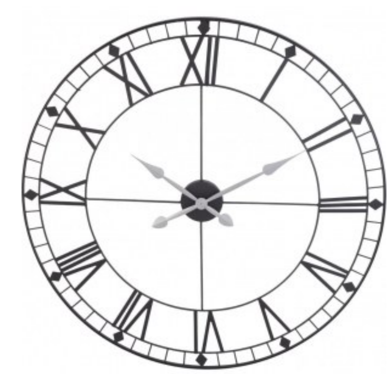


⇩



Properties



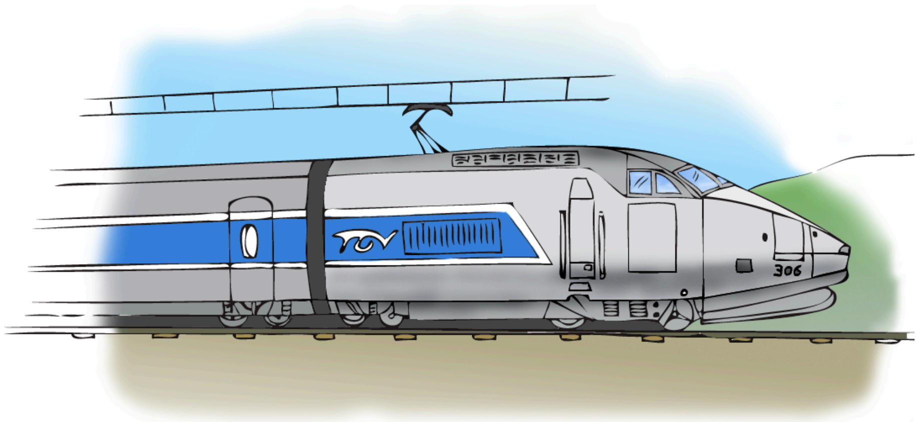


⇩

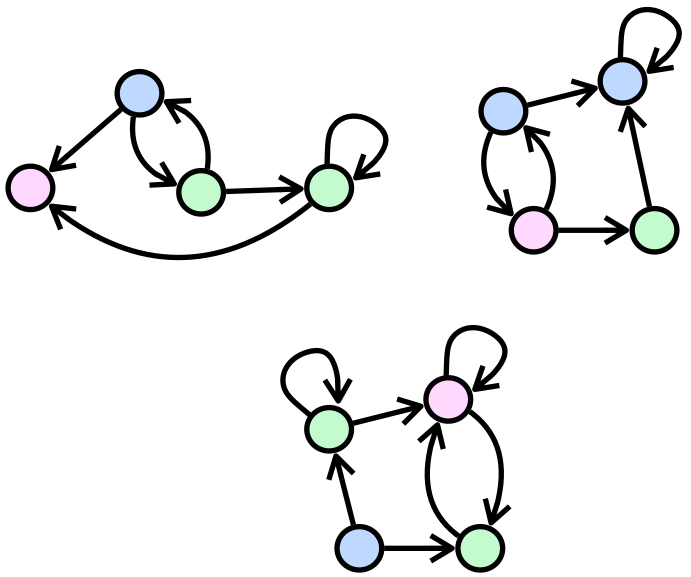
$$\varphi = \mathbf{AG}\,\neg\text{crash} \wedge \left(\mathbb{P}(\mathbf{F}_{\leq 2\text{h}}\text{arr}) \geq 0{,}9\right)$$

Model-checking

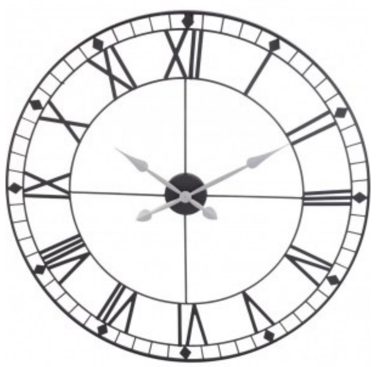
System



⇓



Properties



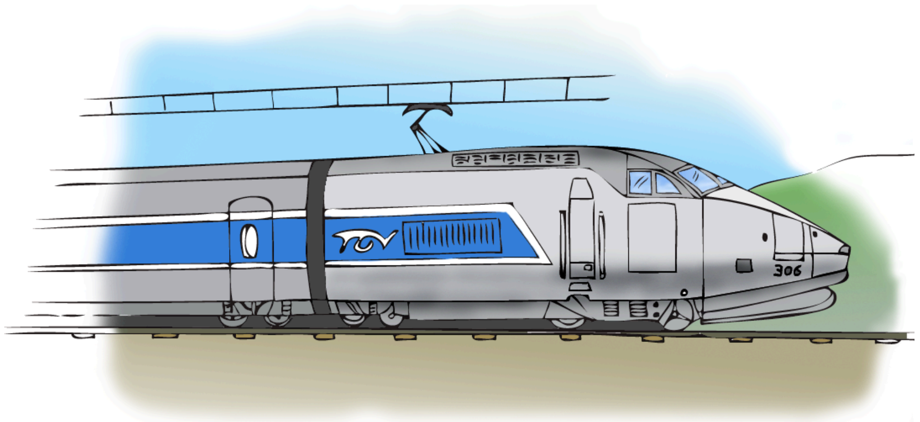
⇓

Model-checking algorithm

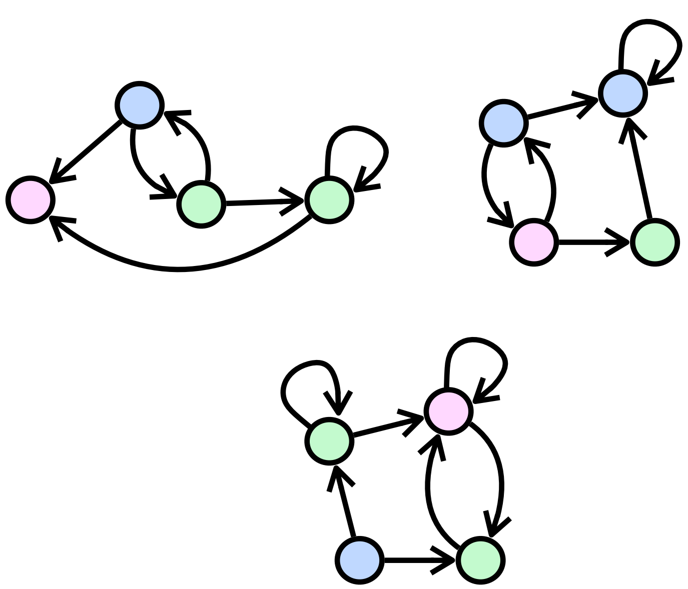
$$\varphi = \mathbf{AG}\,\neg\text{crash} \wedge \left(\mathbb{P}(\mathbf{F}_{\leq 2\text{h}}\text{arr}) \geq 0{,}9\right)$$

Model-checking

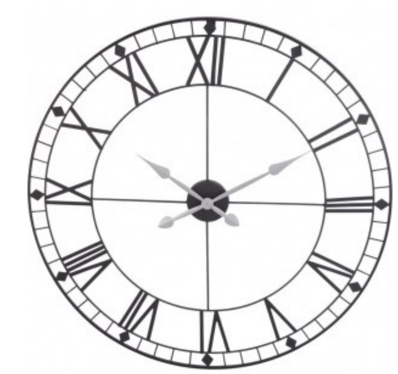
System



⇩



Properties



⇩

$$\varphi = \mathbf{AG}\,\neg\text{crash} \wedge \left(\mathbb{P}(\mathbf{F}_{\leq 2\text{h}}\,\text{arr}) \geq 0{,}9\right)$$

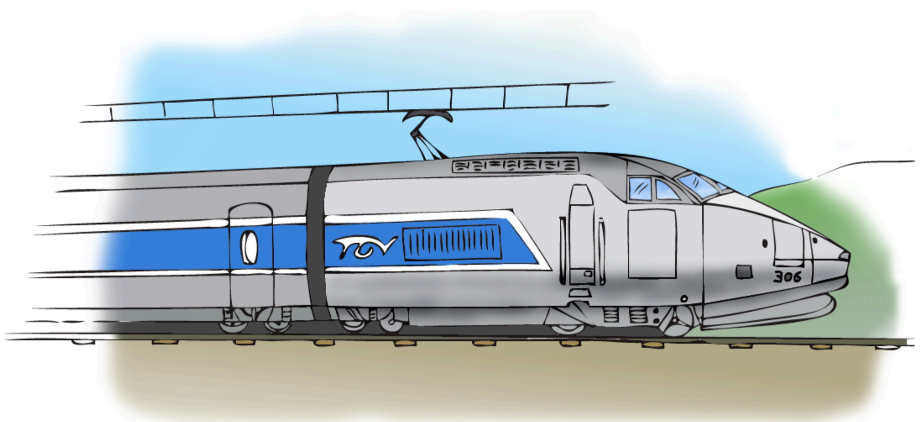
Model-checking algorithm

↓

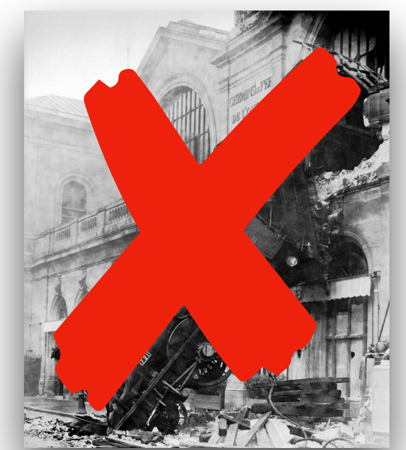
Yes/No/Why?

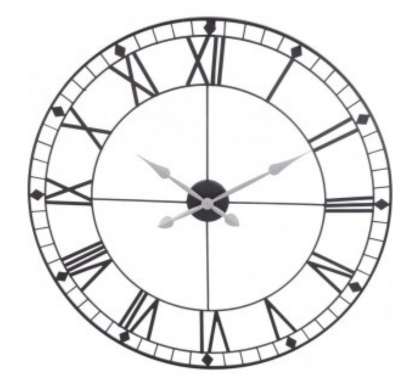
Control or synthesis

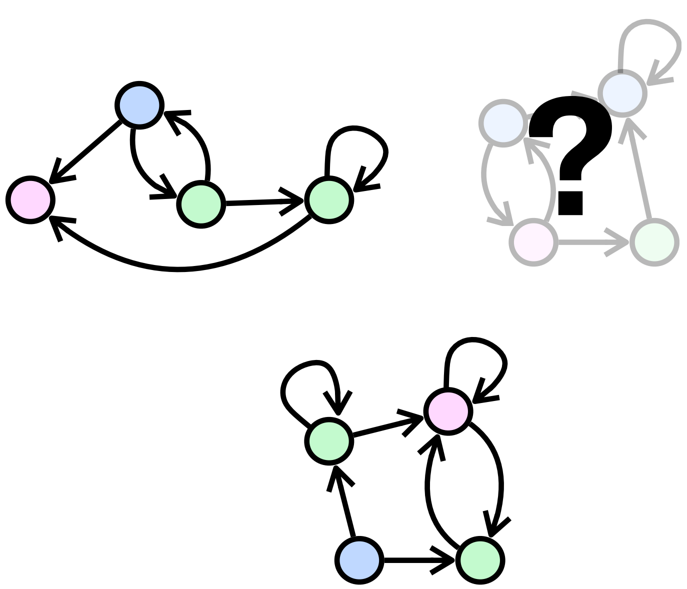
System



Properties







Control/synthesis algorithm

No/Yes/How?

$$\varphi = \mathbf{AG}\,\neg\text{crash} \wedge \Big(\mathbb{P}(\mathbf{F}_{\leq 2\text{h}}\text{arr}) \geq 0{,}9\Big)$$

The talk in one slide

Strategy synthesis for two-player games

Find good and simple controllers for systems interacting with an antagonistic environment

The talk in one slide

Strategy synthesis for two-player games

Find good and simple controllers for systems interacting with an antagonistic environment

Good?

Performance w.r.t. objectives / payoffs / preference relations

The talk in one slide

Strategy synthesis for two-player games

Find good and simple controllers for systems interacting with an antagonistic environment

Good?

Performance w.r.t. objectives / payoffs / preference relations

Simple?

Minimal information for deciding the next steps

The talk in one slide

Strategy synthesis for two-player games

Find good and simple controllers for systems interacting with an antagonistic environment

Good?

Performance w.r.t. objectives / payoffs / preference relations

Simple?

Minimal information for deciding the next steps

When are simple strategies sufficient to play optimally?

Our general approach

[Tho95] On the synthesis of strategies in infinite games (STACS'95).
[Tho02] Thomas. Infinite games and verification (CAV'02).
[GU08] Grädel, Ummels. Solution concepts and algorithms for infinite multiplayer games (New Perspectives in Games and Interactions, 2008).
[BCJ18] Bloem, Chatterjee, Jobstmann. Graph games and reactive synthesis (Handbook of Model-Checking).

Our general approach

- Use **graph-based game models** (state machines) to represent the system and its evolution

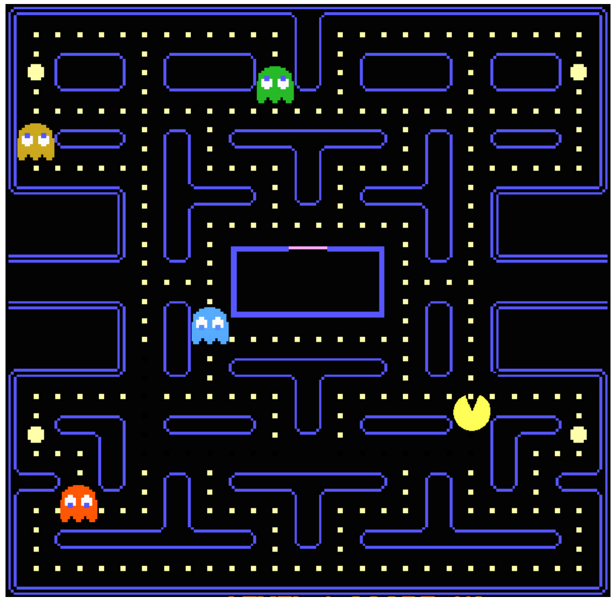
[Tho95] On the synthesis of strategies in infinite games (STACS'95).
[Tho02] Thomas. Infinite games and verification (CAV'02).
[GU08] Grädel, Ummels. Solution concepts and algorithms for infinite multiplayer games (New Perspectives in Games and Interactions, 2008).
[BCJ18] Bloem, Chatterjee, Jobstmann. Graph games and reactive synthesis (Handbook of Model-Checking).

Our general approach

- Use **graph-based game models** (state machines) to represent the system and its evolution
- Use **game theory concepts** to express admissible situations
  - Winning strategies
  - (Pareto-)Optimal strategies
  - Nash equilibria
  - Subgame-perfect equilibria
  - ...

[Tho95] On the synthesis of strategies in infinite games (STACS'95).
[Tho02] Thomas. Infinite games and verification (CAV'02).
[GU08] Grädel, Ummels. Solution concepts and algorithms for infinite multiplayer games (New Perspectives in Games and Interactions, 2008).
[BCJ18] Bloem, Chatterjee, Jobstmann. Graph games and reactive synthesis (Handbook of Model-Checking).

Games
What they often are



Games
A broader sense

Interaction

Goal

- Model and analyze (using math. tools) situations of interactive decision making

Games
A broader sense

Interaction

Goal

- Model and analyze (using math. tools) situations of interactive decision making

Ingredients

- Several decision makers (players)
- Possibly each with different goals
- The decision of each player impacts the outcome of all

Games
A broader sense

Interaction

Goal

- Model and analyze (using math. tools) situations of interactive decision making

Ingredients

- Several decision makers (players)
- Possibly each with different goals
- The decision of each player impacts the outcome of all

Wide range of applicability

« [...] it is a context-free mathematical toolbox. »

- Social science: e.g. social choice theory
- Theoretical economics: e.g. models of markets, auctions
- Political science: e.g. fair division
- Biology: e.g. evolutionary biology
- ...

[MSZ13] Maschler, Solan, Zamir. Game theory (2013).

Games
A broader sense

Interaction

Goal

- Model and analyze (using math. tools) situations of interactive decision making

Ingredients

- Several decision makers (players)
- Possibly each with different goals
- The decision of each player impacts the outcome of all

Wide range of applicability

« [...] it is a context-free mathematical toolbox. »

- Social science: e.g. social choice theory
- Theoretical economics: e.g. models of markets, auctions
- Political science: e.g. fair division
- Biology: e.g. evolutionary biology
- ...

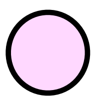
+ Computer science

[MSZ13] Maschler, Solan, Zamir. Game theory (2013).

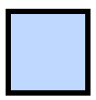
Turn-based games on graphs

States   Edges

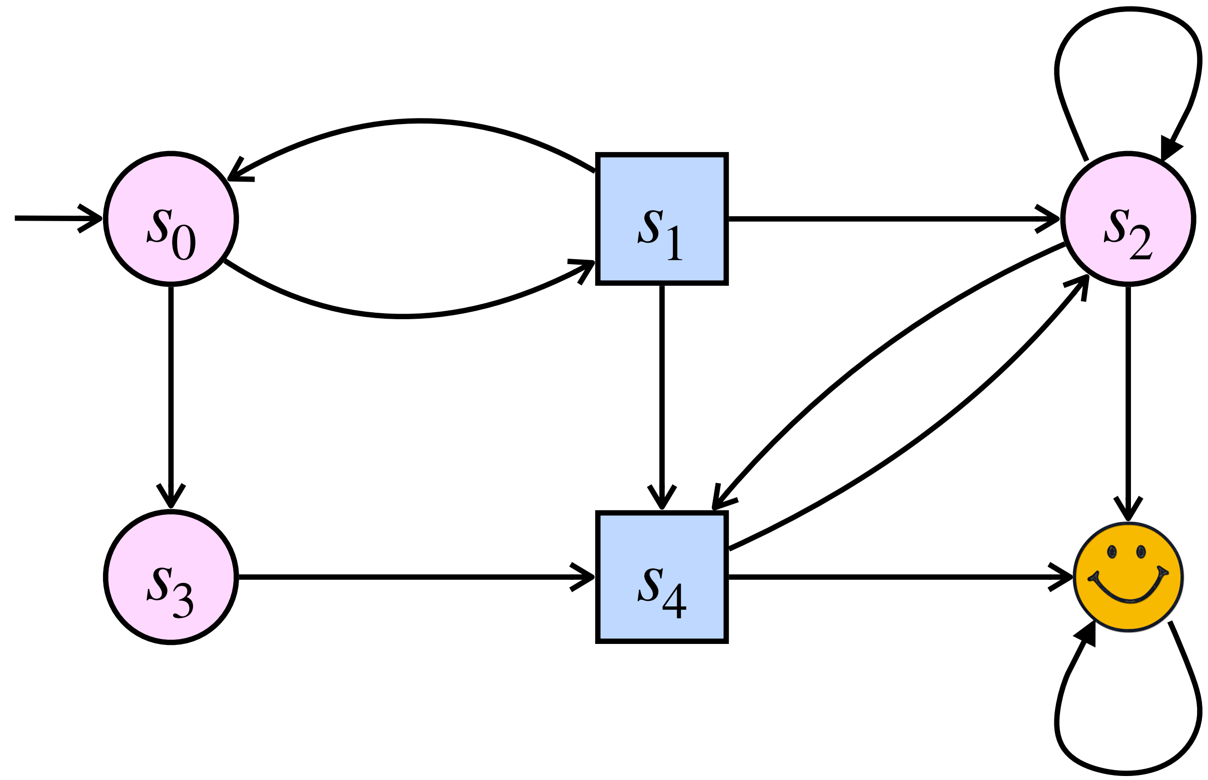
$$\mathcal{G} = (S, s_0, S_1, S_2, E)$$



: player P_1



: player P_2



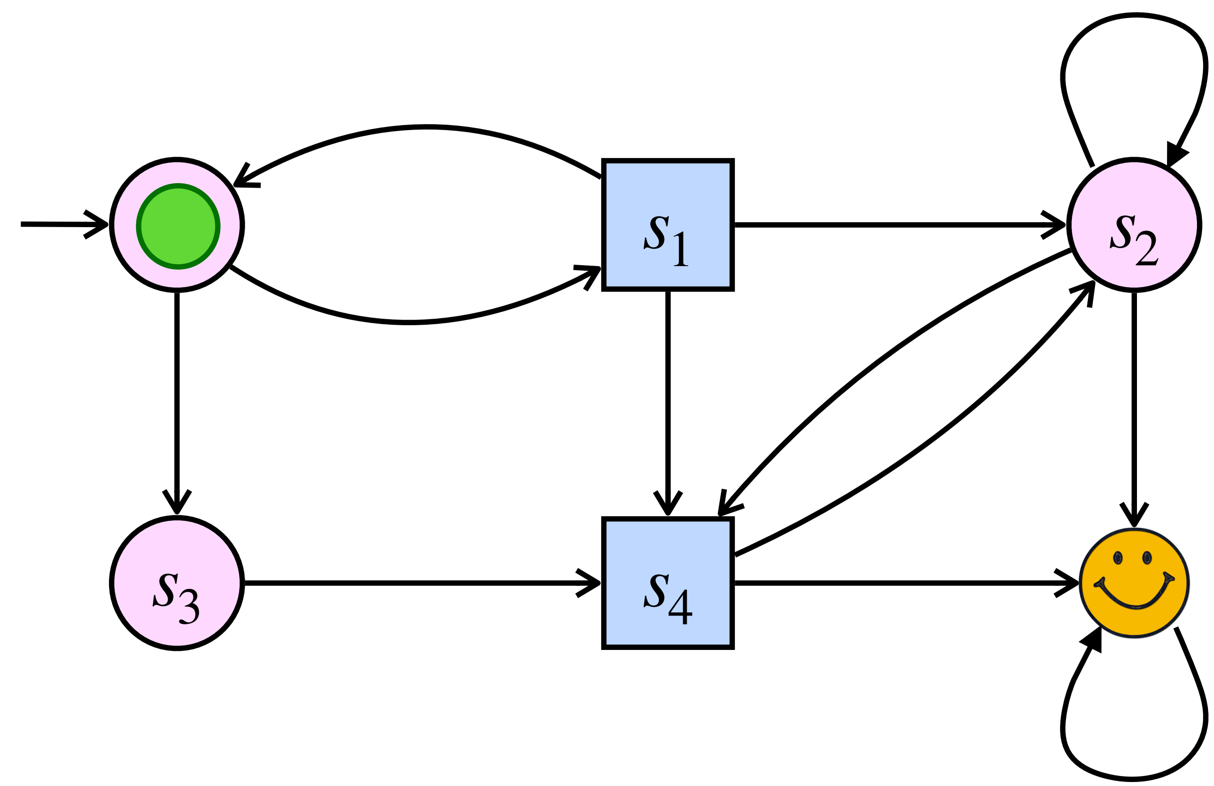
Turn-based games on graphs

States Edges

$$\mathcal{G} = (S, s_0, S_1, S_2, E)$$

: player P_1

: player P_2



s_0

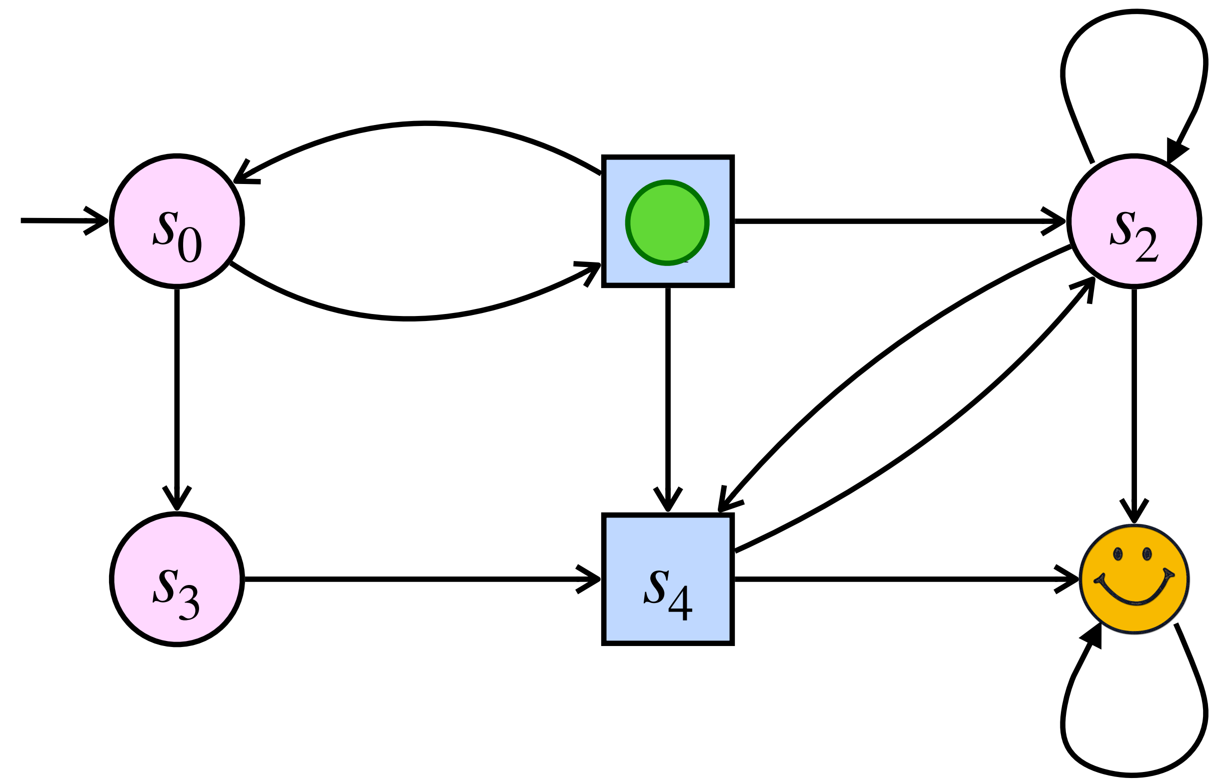
Turn-based games on graphs

States — Edges

$$\mathcal{G} = (S, s_0, S_1, S_2, E)$$

○ : player P_1

□ : player P_2



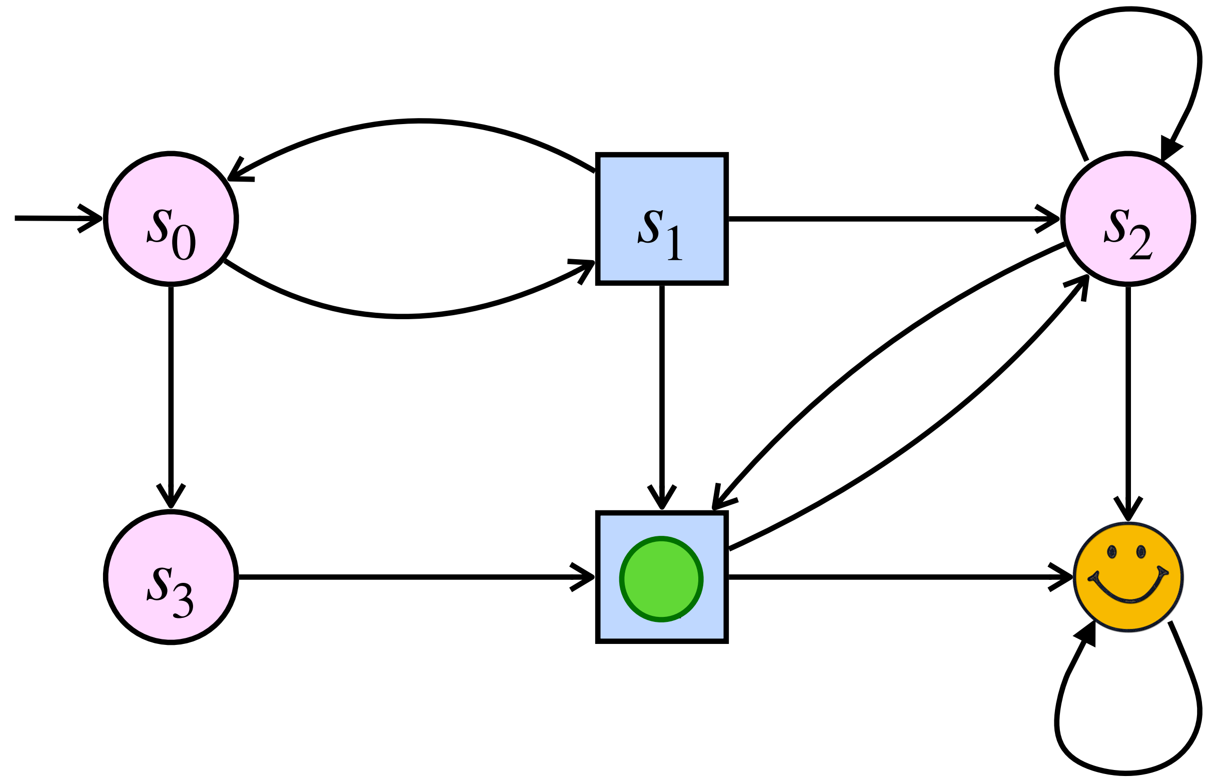
$$s_0 \rightarrow s_1$$

1. P_1 chooses the edge (s_0, s_1)

Turn-based games on graphs

States → S, Edges → E

$$\mathscr{G} = (S, s_0, S_1, S_2, E)$$



○ : player P_1

□ : player P_2

$$s_0 \rightarrow s_1 \rightarrow s_4$$

1. P_1 chooses the edge (s_0, s_1)
2. P_2 chooses the edge (s_1, s_4)

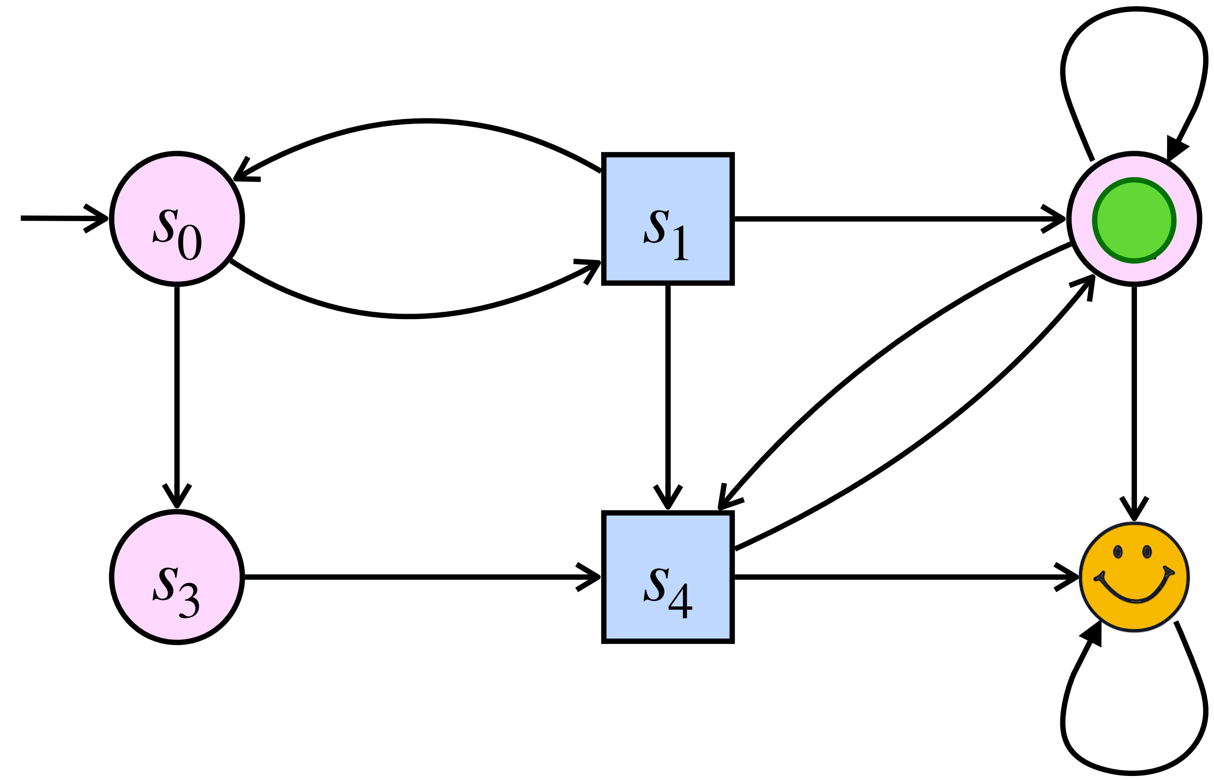
Turn-based games on graphs

States → S, Edges → E

$$\mathcal{G} = (S, s_0, S_1, S_2, E)$$

○ : player P_1

□ : player P_2



$$s_0 \rightarrow s_1 \rightarrow s_4 \rightarrow s_2$$

1. P_1 chooses the edge (s_0, s_1)
2. P_2 chooses the edge (s_1, s_4)
3. P_2 chooses the edge (s_4, s_2)

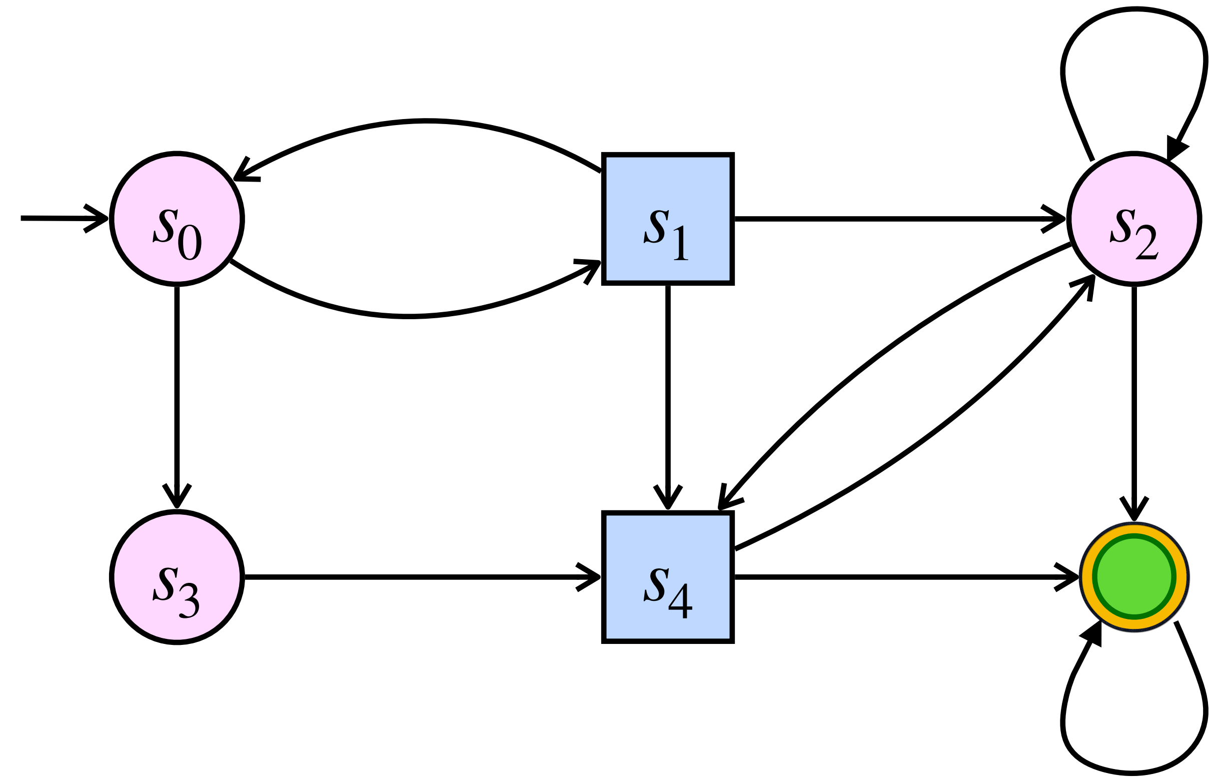
Turn-based games on graphs

States — Edges

$$\mathcal{G} = (S, s_0, S_1, S_2, E)$$

○ : player P_1

□ : player P_2



$s_0 \to s_1 \to s_4 \to s_2 \to$ 

1. P_1 chooses the edge (s_0, s_1)
2. P_2 chooses the edge (s_1, s_4)
3. P_2 chooses the edge (s_4, s_2)
4. P_1 chooses the edge (s_2, )

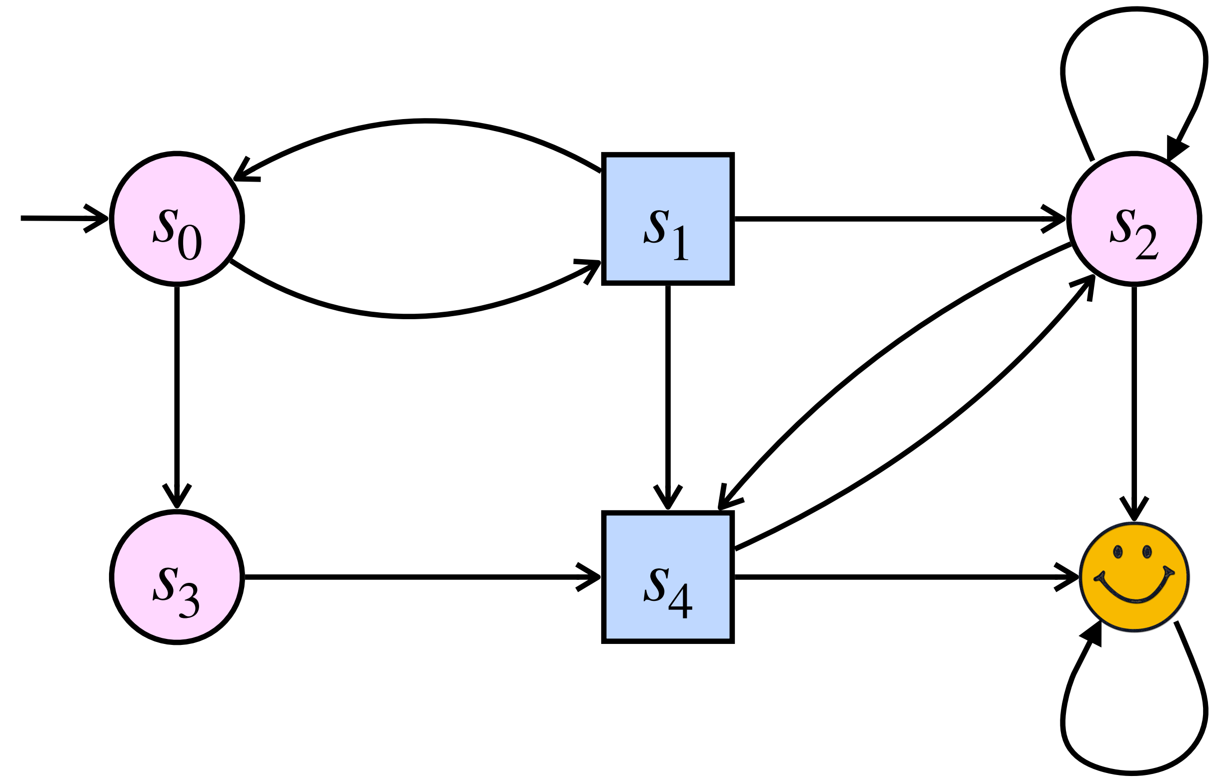
Turn-based games on graphs

States → S, Edges → E

$$\mathcal{G} = (S, s_0, S_1, S_2, E)$$

○ : player P_1

□ : player P_2



$s_0 \rightarrow s_1 \rightarrow s_4 \rightarrow s_2 \rightarrow$ 

1. P_1 chooses the edge (s_0, s_1)
2. P_2 chooses the edge (s_1, s_4)
3. P_2 chooses the edge (s_4, s_2)
4. P_1 chooses the edge $(s_2,$ $)$

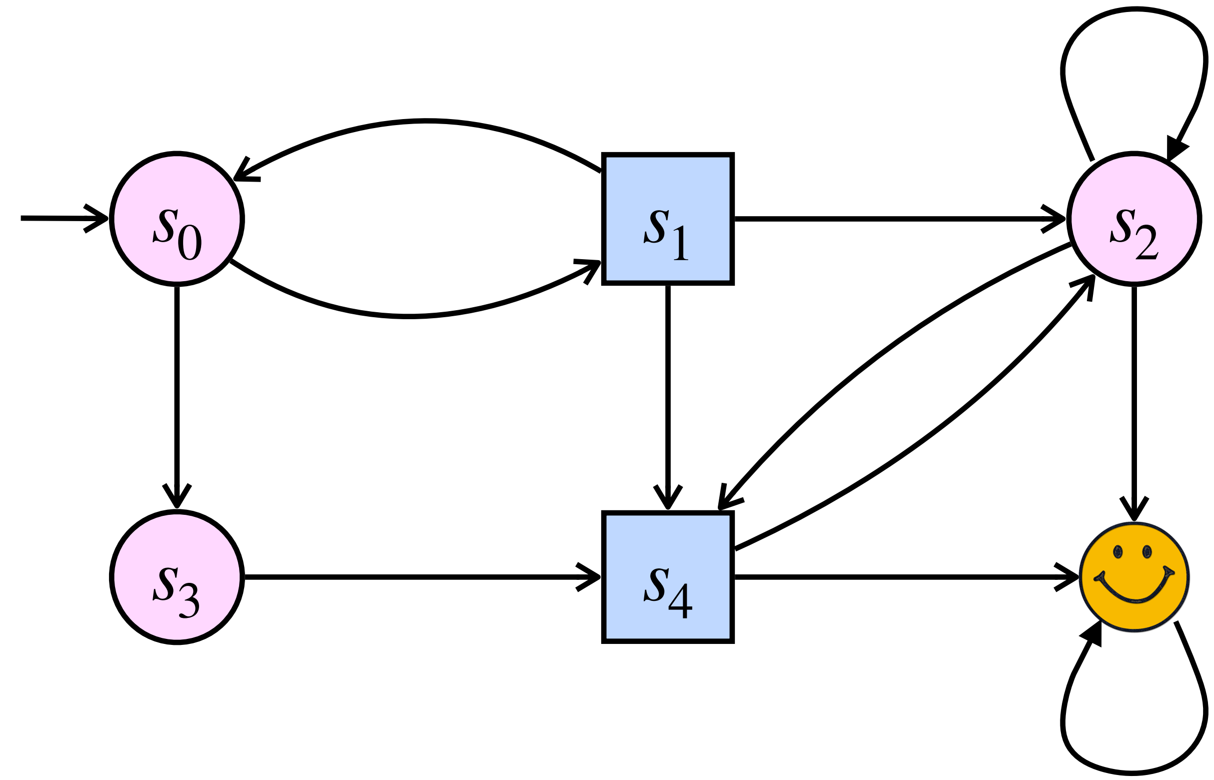
Turn-based games on graphs

States   Edges

$$\mathcal{G} = (S, s_0, S_1, S_2, E)$$

○ : player P_1

□ : player P_2

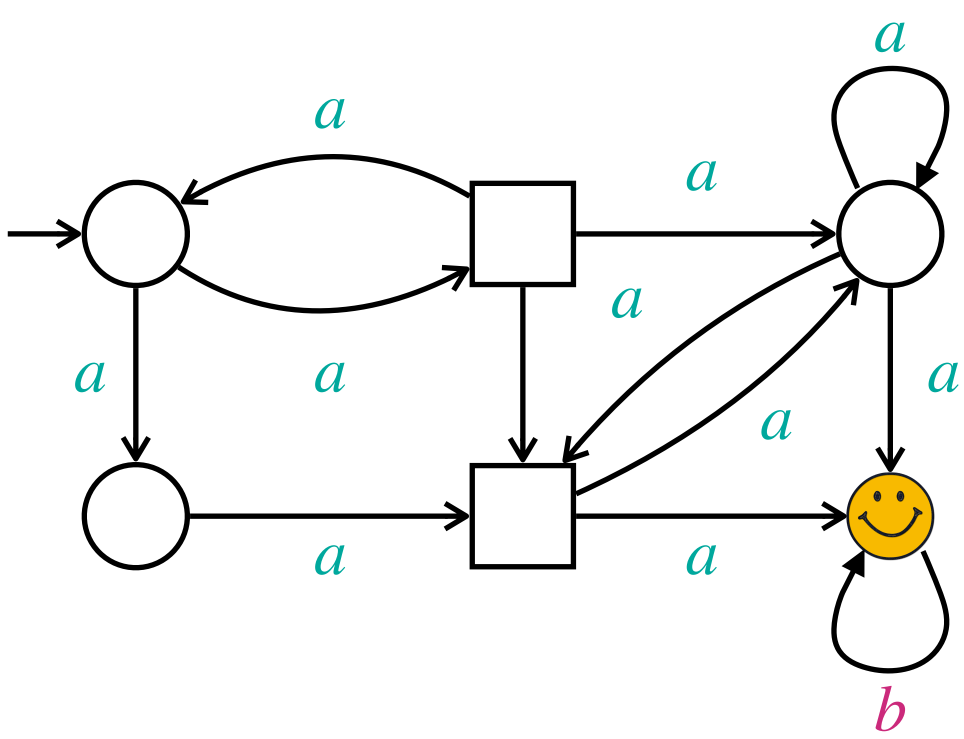


$$s_0 \to s_1 \to s_4 \to s_2 \to$$ 

1. P_1 chooses the edge (s_0, s_1)
2. P_2 chooses the edge (s_1, s_4)
3. P_2 chooses the edge (s_4, s_2)
4. P_1 chooses the edge $(s_2,$ $)$

Players use **strategies** to play.
A strategy for P_i is $\sigma_i : S^*S_i \to E$

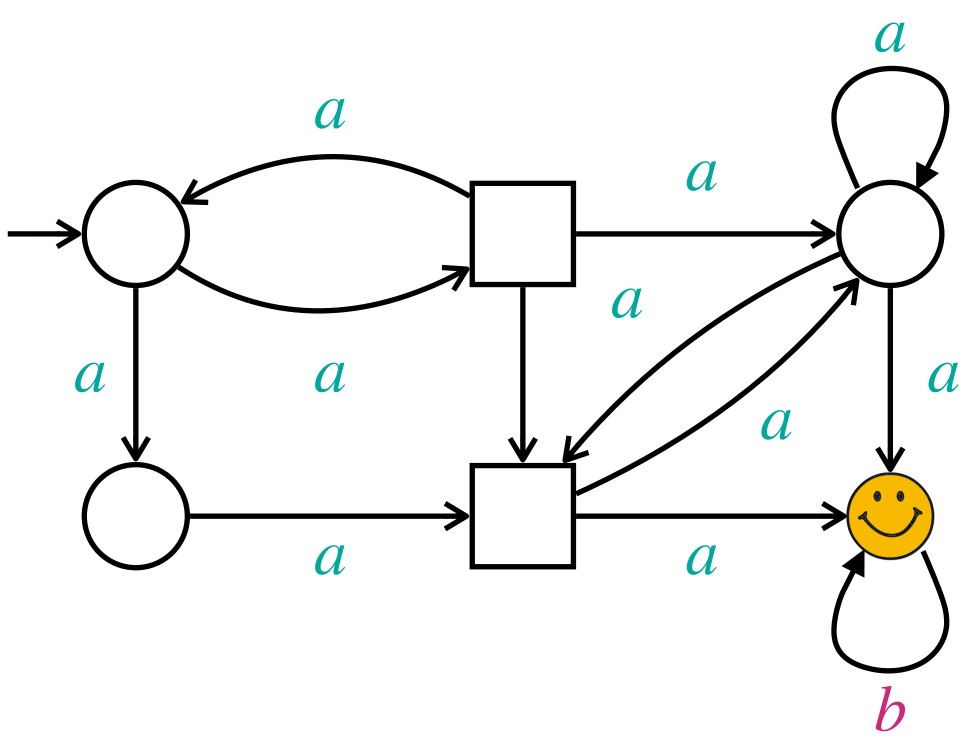
Objectives for the players



$$C = \{a, b\}$$ set of colors

$$E \subseteq S \times C \times S$$

Objectives for the players

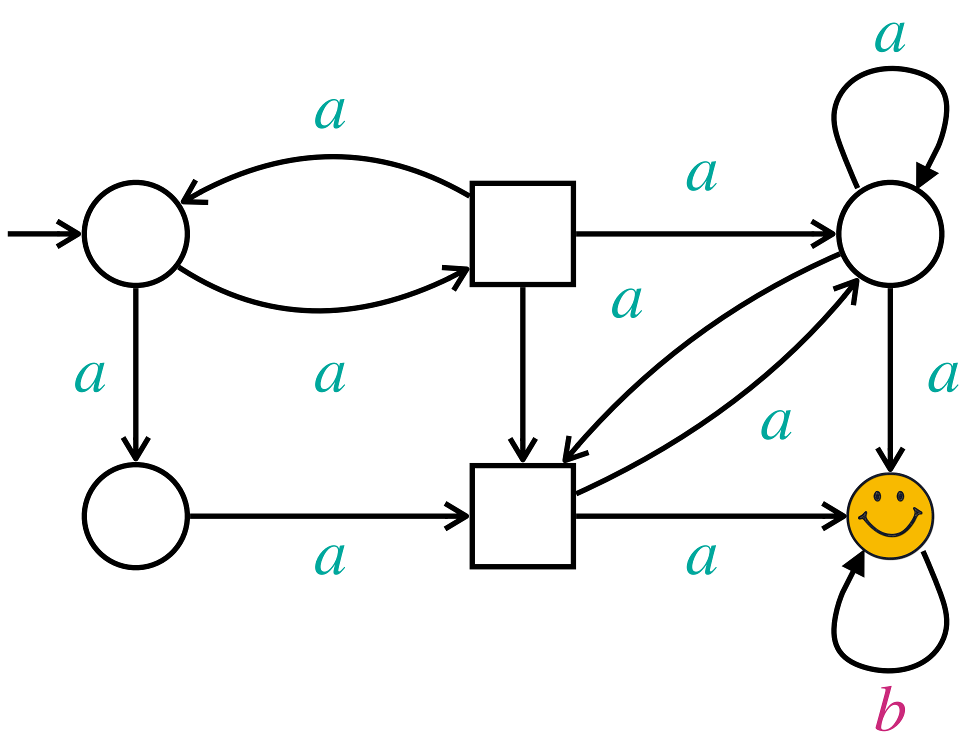


$C = \{a, b\}$ set of colors

$E \subseteq S \times C \times S$

- Winning objective for P_i: $W_i \subseteq C^\omega$, e.g. $W_1 = C^* \cdot b \cdot C^\omega$

Objectives for the players



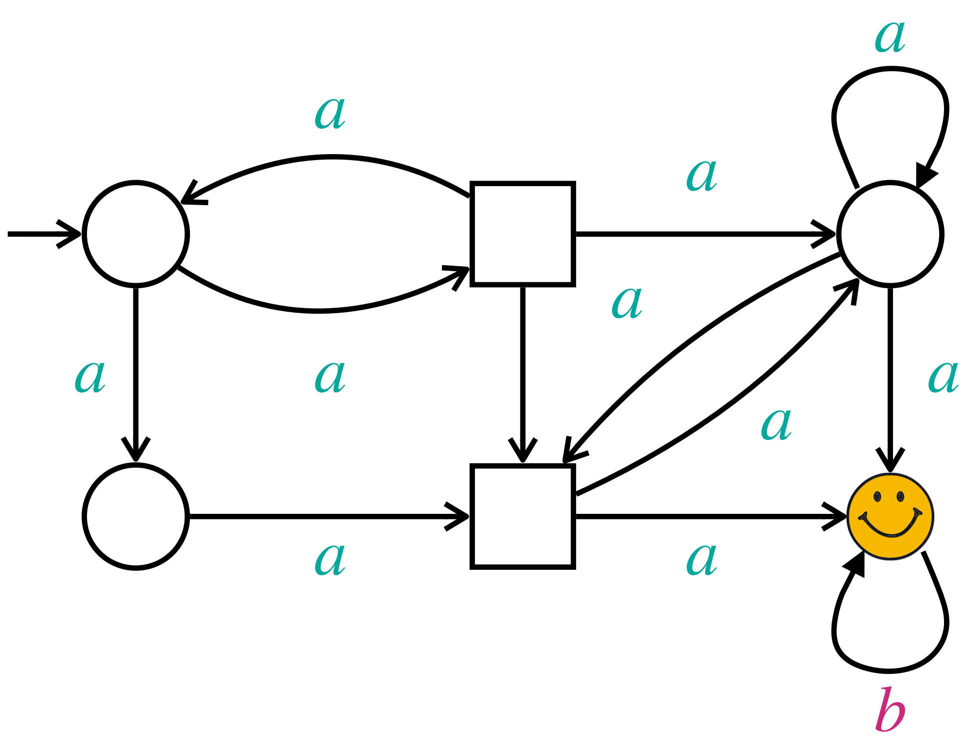
$C = \{a, b\}$ set of colors

$E \subseteq S \times C \times S$

- Winning objective for P_i: $W_i \subseteq C^\omega$, e.g. $W_1 = C^* \cdot b \cdot C^\omega$

- Payoff function: $p_i : C^\omega \to \mathbb{R}$, e.g. mean-payoff

Objectives for the players



$C = \{a, b\}$ set of colors

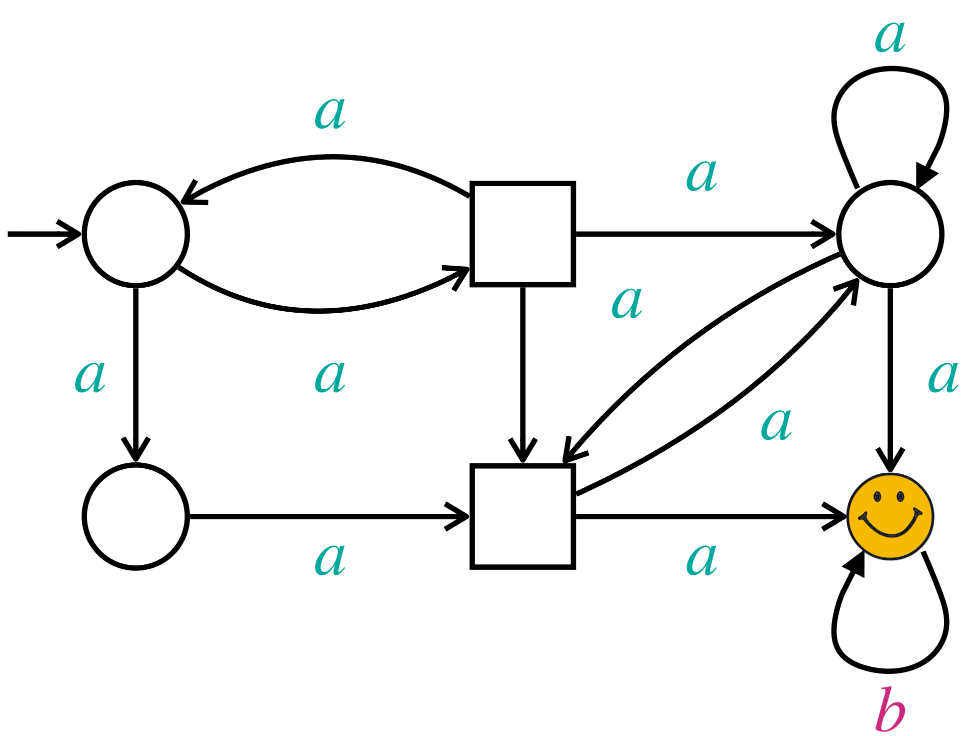
$E \subseteq S \times C \times S$

- Winning objective for P_i: $W_i \subseteq C^\omega$, e.g. $W_1 = C^* \cdot b \cdot C^\omega$

- Payoff function: $p_i : C^\omega \to \mathbb{R}$, e.g. mean-payoff

- Preference relation: $\sqsubseteq_i \subseteq C^\omega \times C^\omega$ (total preorder)

Objectives for the players



Zero-sum assumption

$C = \{a, b\}$ set of colors

$E \subseteq S \times C \times S$

- Winning objective for P_i: $W_i \subseteq C^\omega$, e.g. $W_1 = C^* \cdot b \cdot C^\omega$

$W_2 = W_1^c$

- Payoff function: $p_i : C^\omega \to \mathbb{R}$, e.g. mean-payoff

$p_1 + p_2 = 0$

- Preference relation: $\sqsubseteq_i \subseteq C^\omega \times C^\omega$ (total preorder)

$\sqsubseteq_2 = \sqsubseteq_1^{-1}$

Objectives for the players

Zero-sum assumption

$C = \{a, b\}$ set of colors

$E \subseteq S \times C \times S$

- Winning objective for P_i: $W_i \subseteq C^\omega$, e.g. $W_1 = C^* \cdot b \cdot C^\omega$

$W_2 = W_1^c$

- We focus on winning objectives, and write W for W_1

0

- Preference relation: $\sqsubseteq_i \subseteq C^\omega \times C^\omega$ (total preorder)

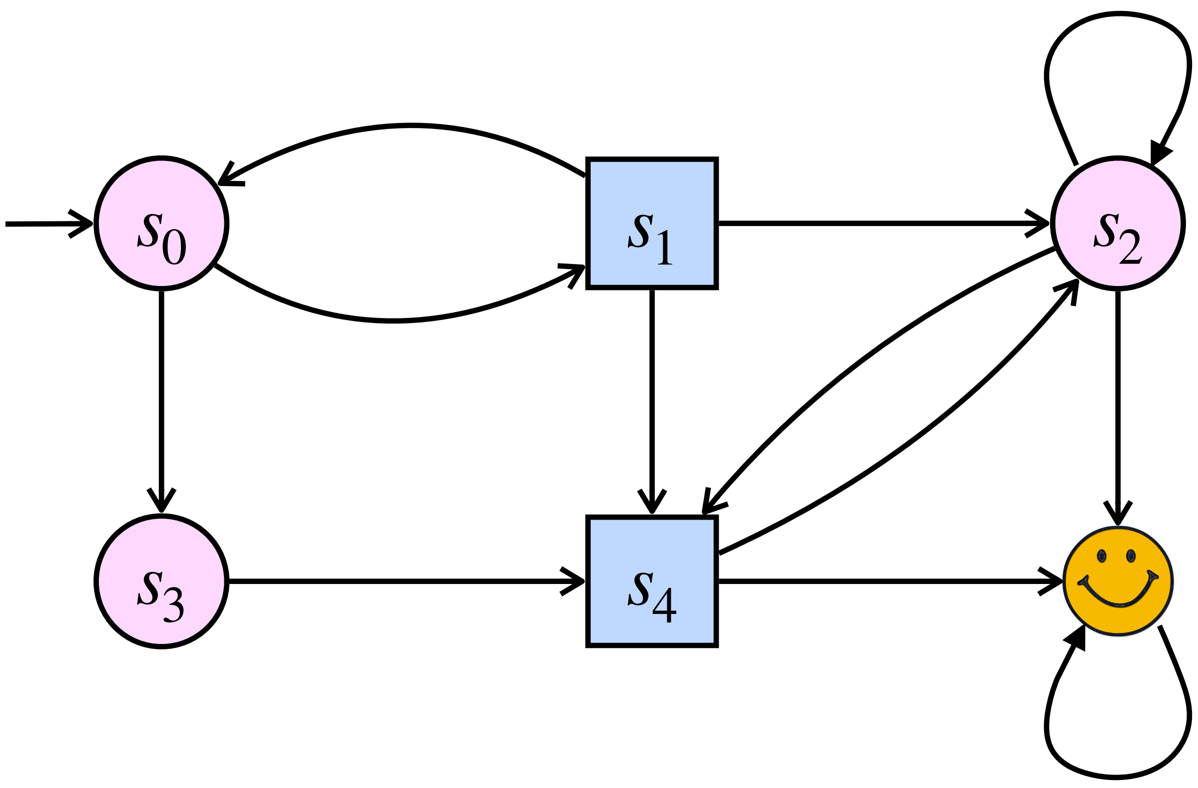
$\sqsubseteq_2 = \sqsubseteq_1^{-1}$

What does it mean to win a game?

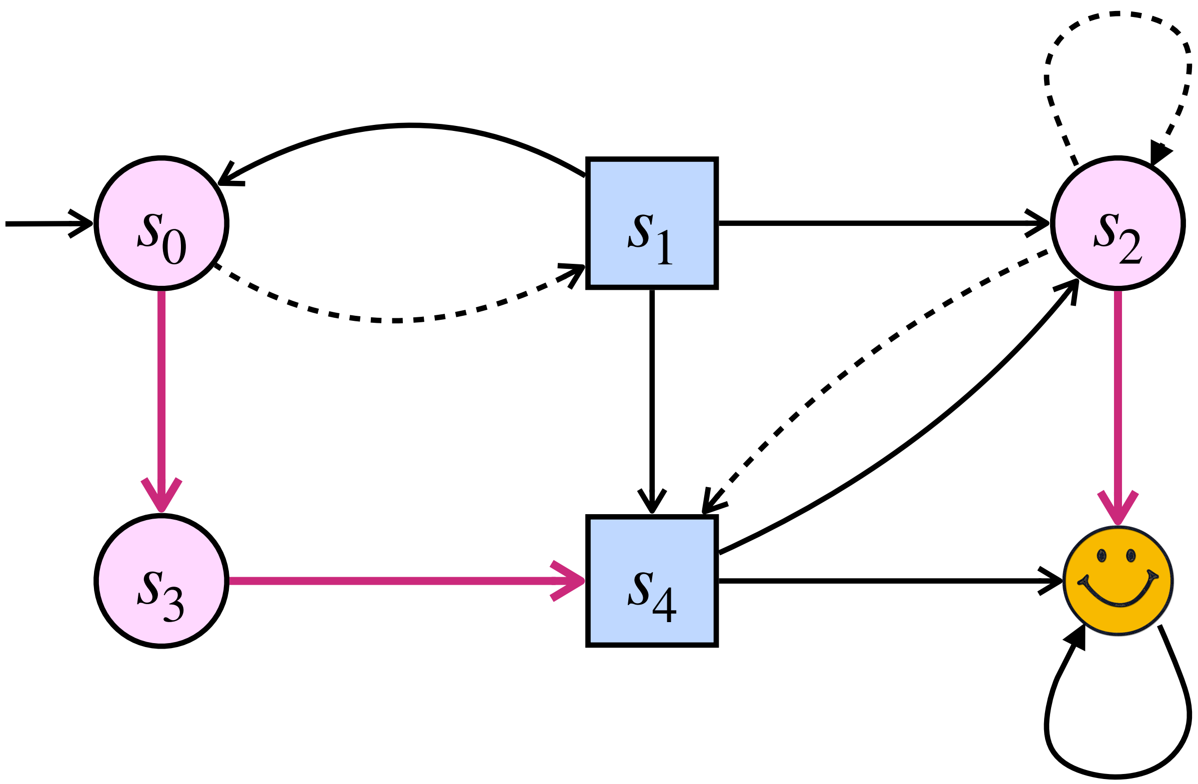
What does it mean to win a game?

- Play $\rho = s_0 s_1 s_2 \ldots$ is compatible with σ_i whenever $s_j \in S_i$ implies $(s_j, s_{j+1}) = \sigma_i\left(s_0 s_1 \ldots s_j\right)$. We write Out$(\sigma_i)$.

Outcomes of a strategy

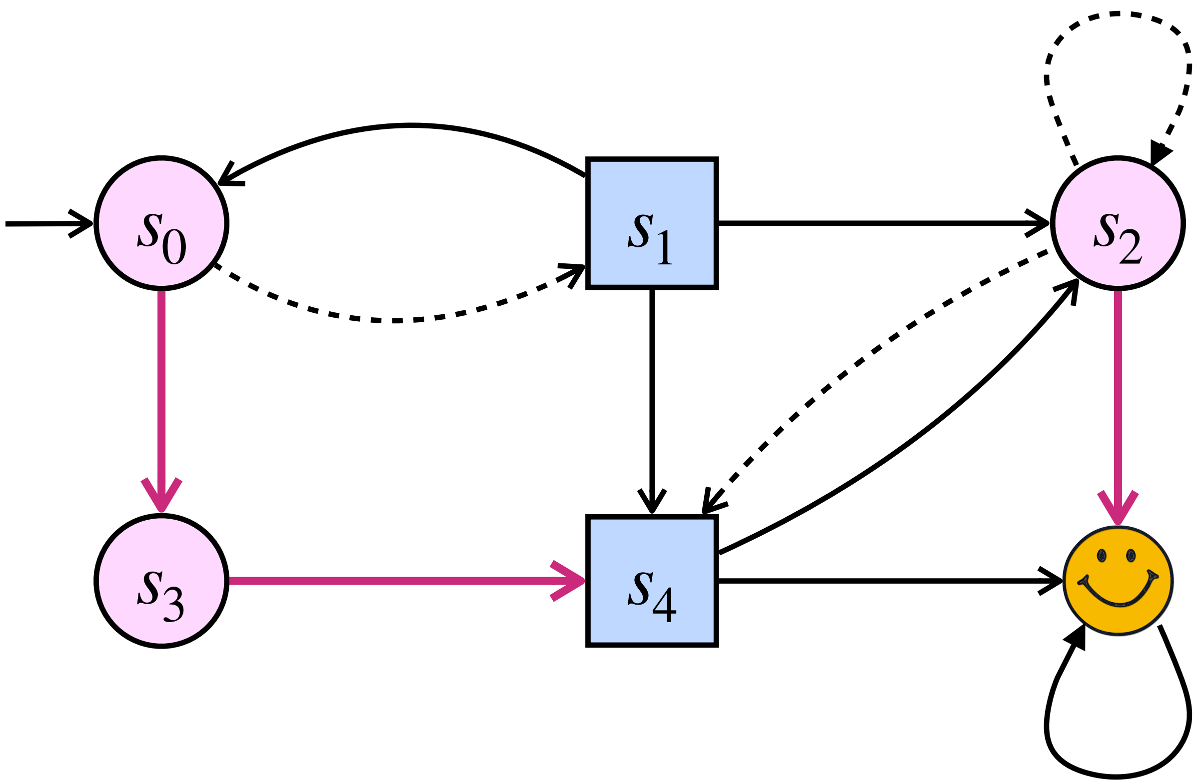


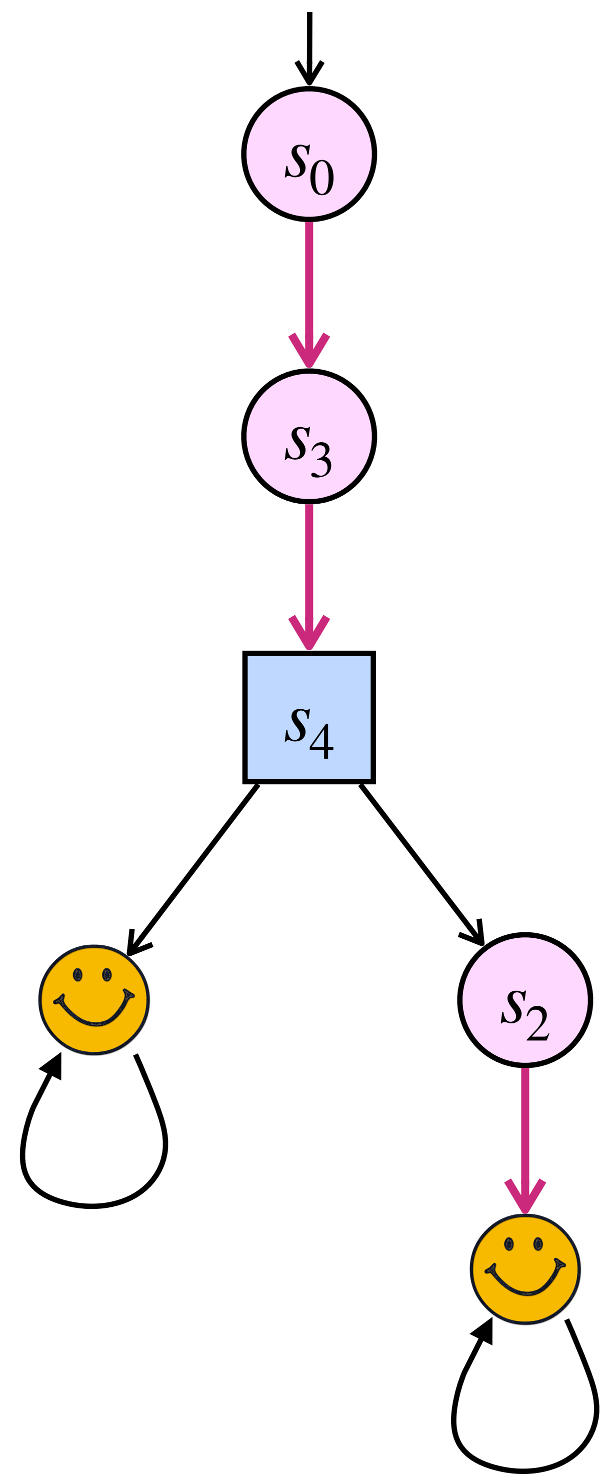
Outcomes of a strategy



- Strategy σ

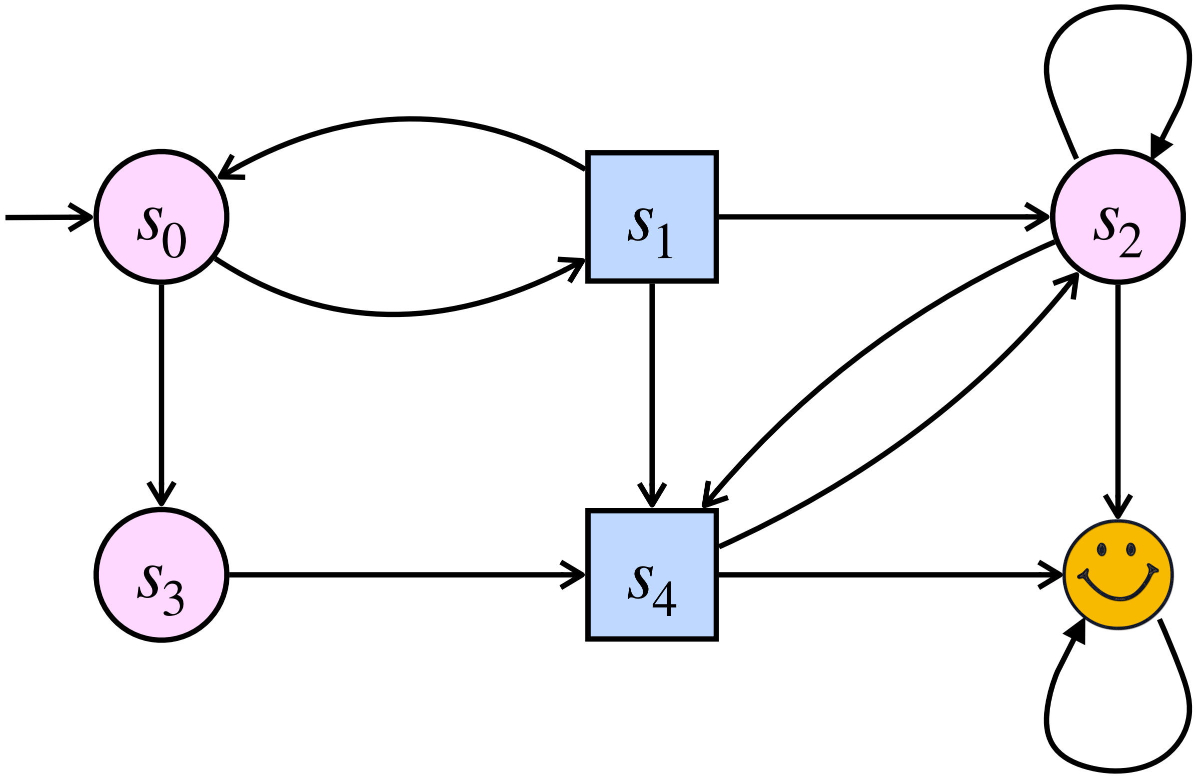
Outcomes of a strategy



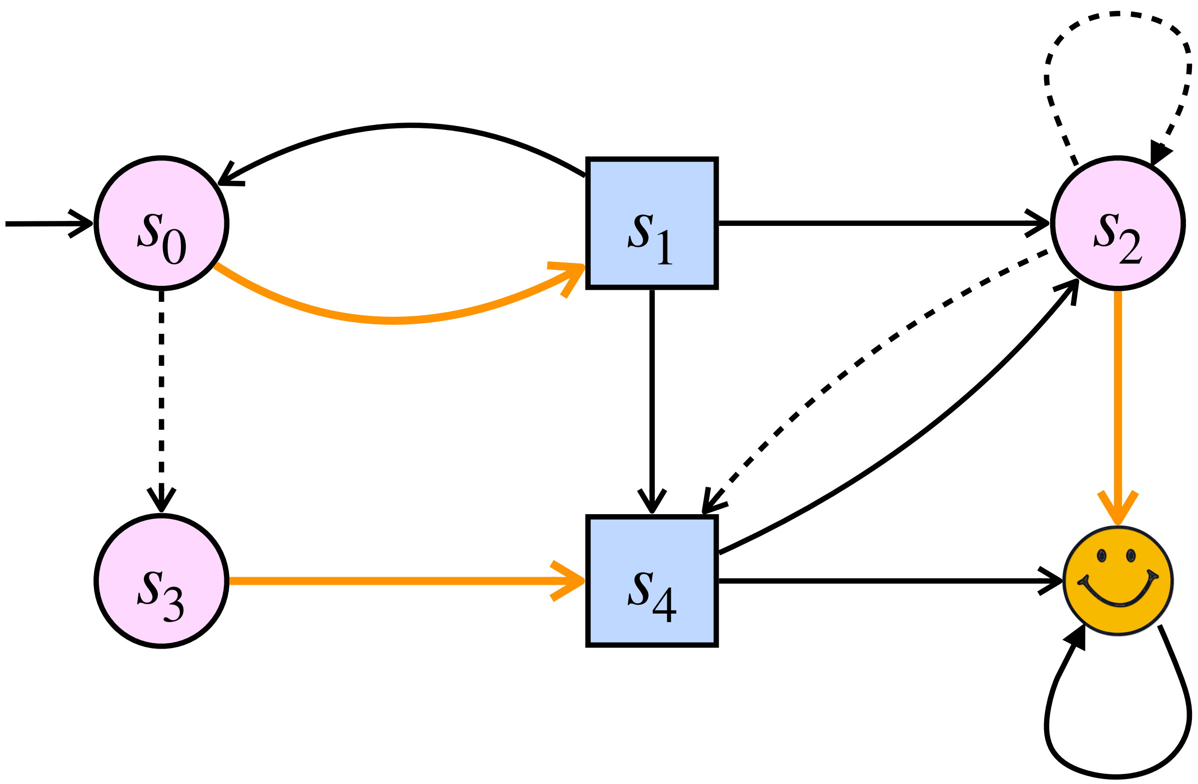


- Strategy σ
- $\mathrm{Out}(\sigma)$ has two plays, which are both winning

Outcomes of a strategy

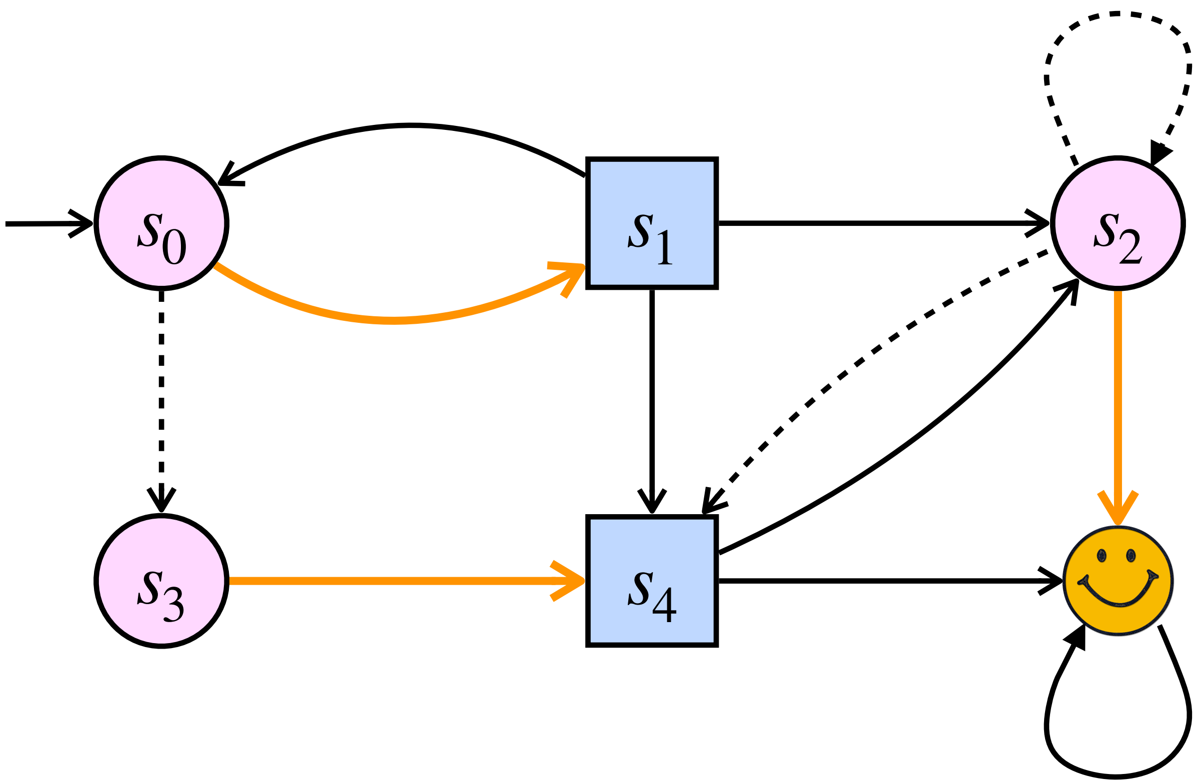


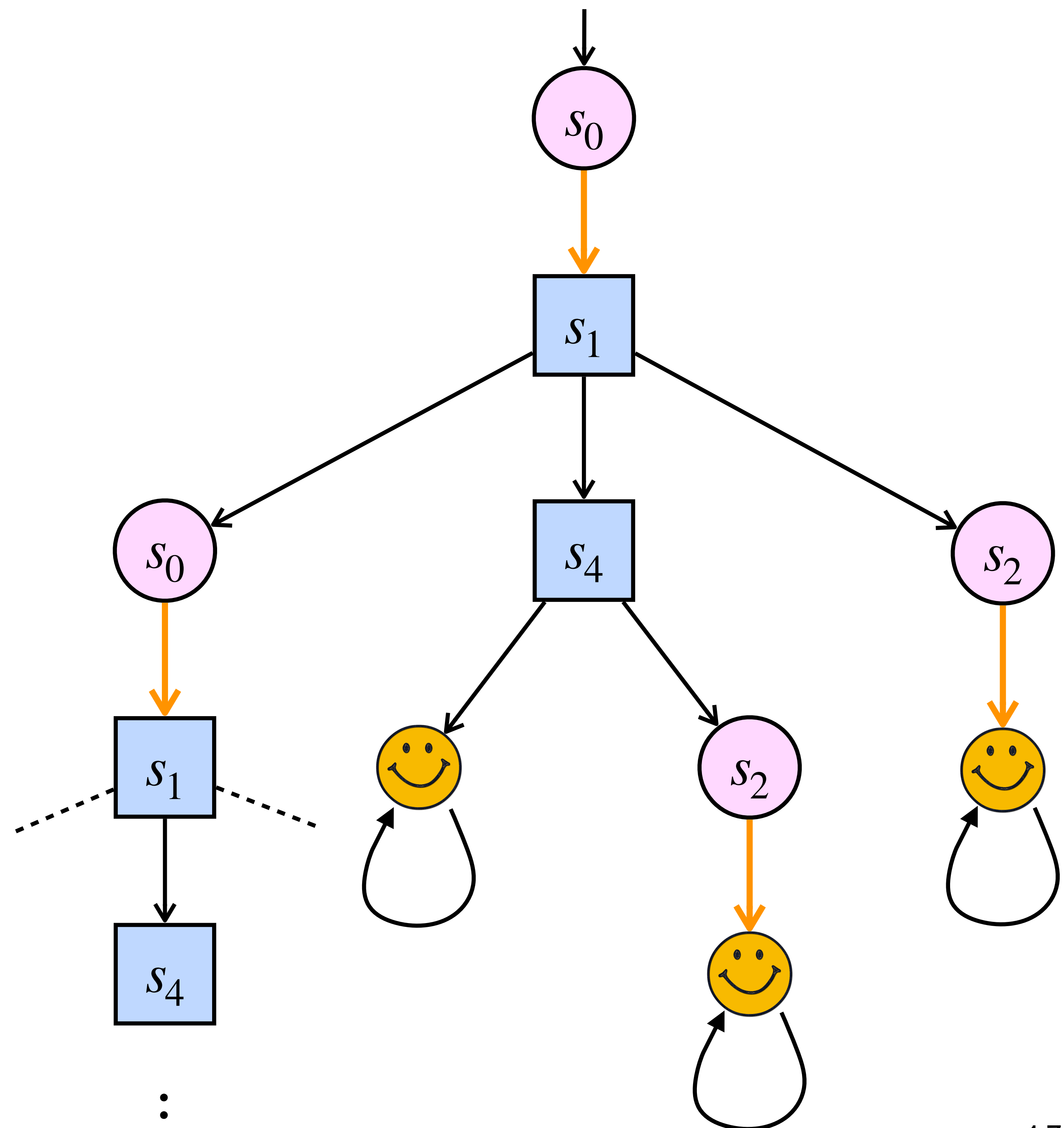
Outcomes of a strategy



- Strategy σ

Outcomes of a strategy





- Strategy σ
- $\mathrm{Out}(\sigma)$ has infinitely many plays, some of them are not winning

What does it mean to win a game?

- Play $\rho = s_0 s_1 s_2 \ldots$ is compatible with σ_i whenever $s_j \in S_i$ implies $(s_j, s_{j+1}) = \sigma_i\left(s_0 s_1 \ldots s_j\right)$. We write Out($\sigma_i$).

- σ_i is **winning** if all plays compatible with σ_i belong to W_i
  σ_i is **optimal** if it is winning or if the initial state is losing

[Mar75] Martin. Borel determinacy (Annals of Mathematics).

What does it mean to win a game?

- Play $\rho = s_0 s_1 s_2 \ldots$ is compatible with σ_i whenever $s_j \in S_i$ implies $(s_j, s_{j+1}) = \sigma_i\left(s_0 s_1 \ldots s_j\right)$. We write Out$(\sigma_i)$.

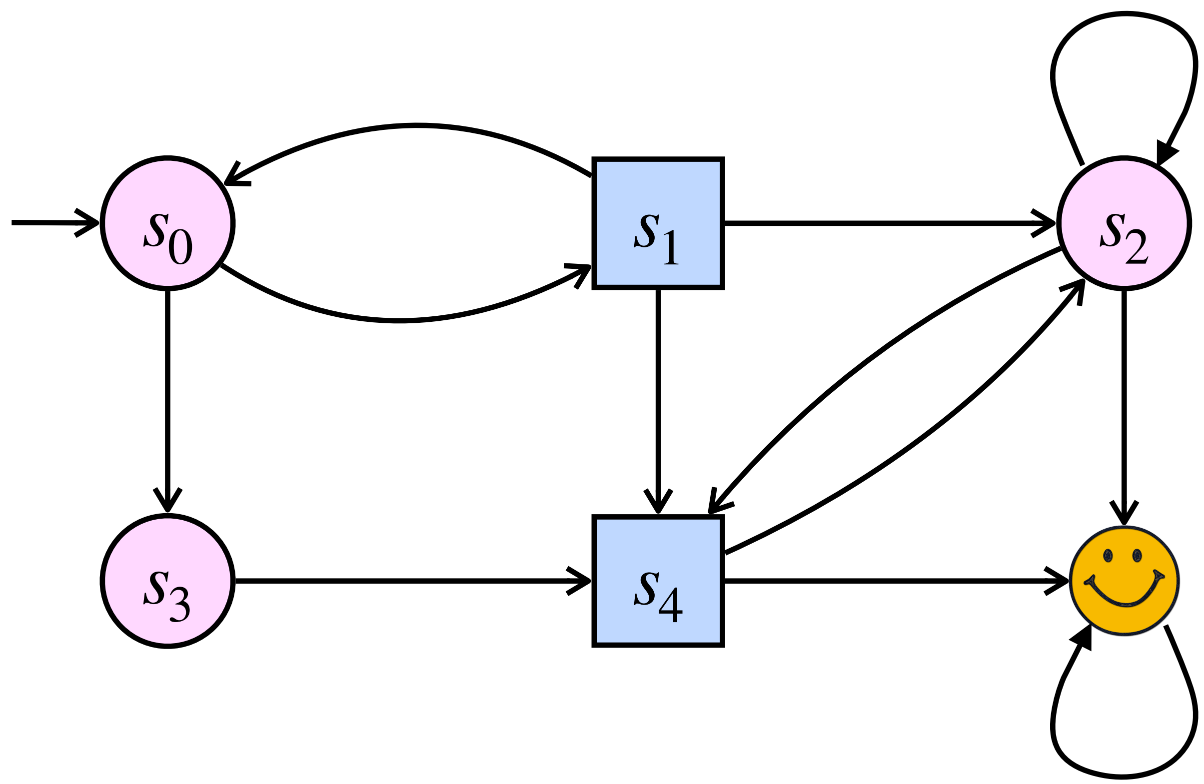
- σ_i is **winning** if all plays compatible with σ_i belong to W_i
  σ_i is **optimal** if it is winning or if the initial state is losing

Martin's determinacy theorem

Turn-based zero-sum games are determined for Borel winning objectives: in every game, either P_1 or P_2 has a winning strategy.

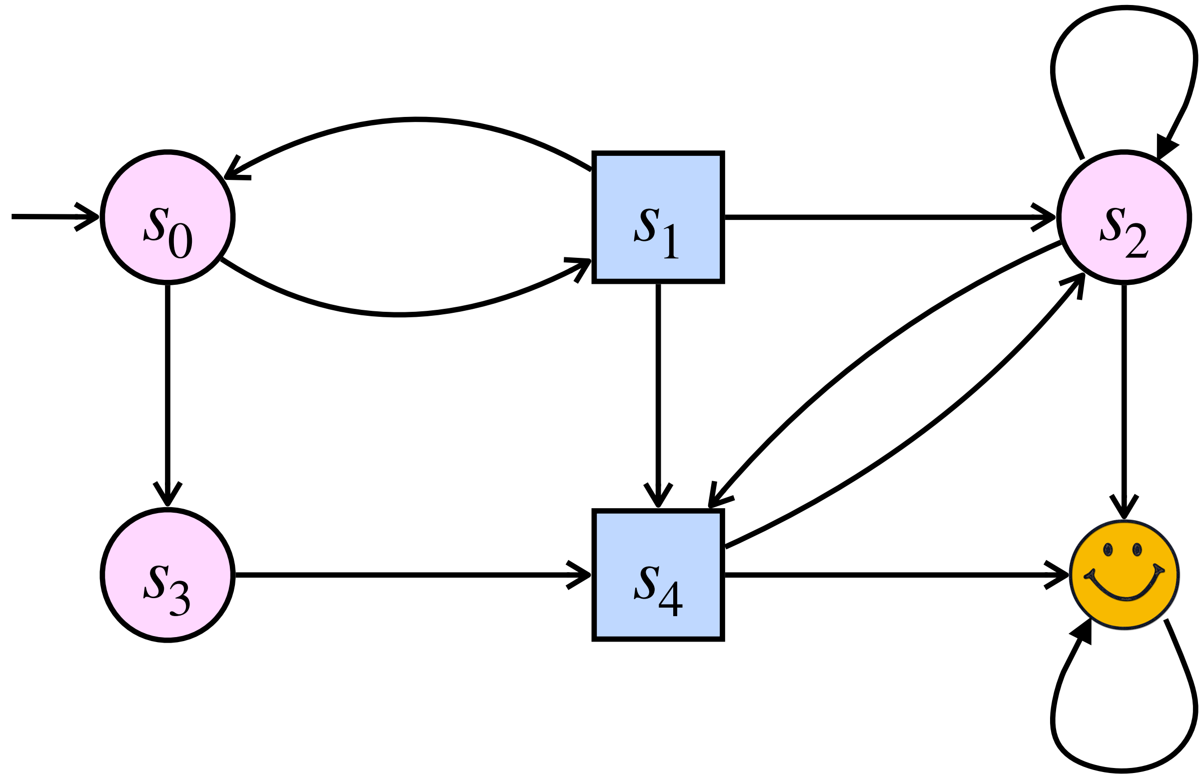
[Mar75] Martin. Borel determinacy (Annals of Mathematics).

Relevant questions



φ = Reach(☺)

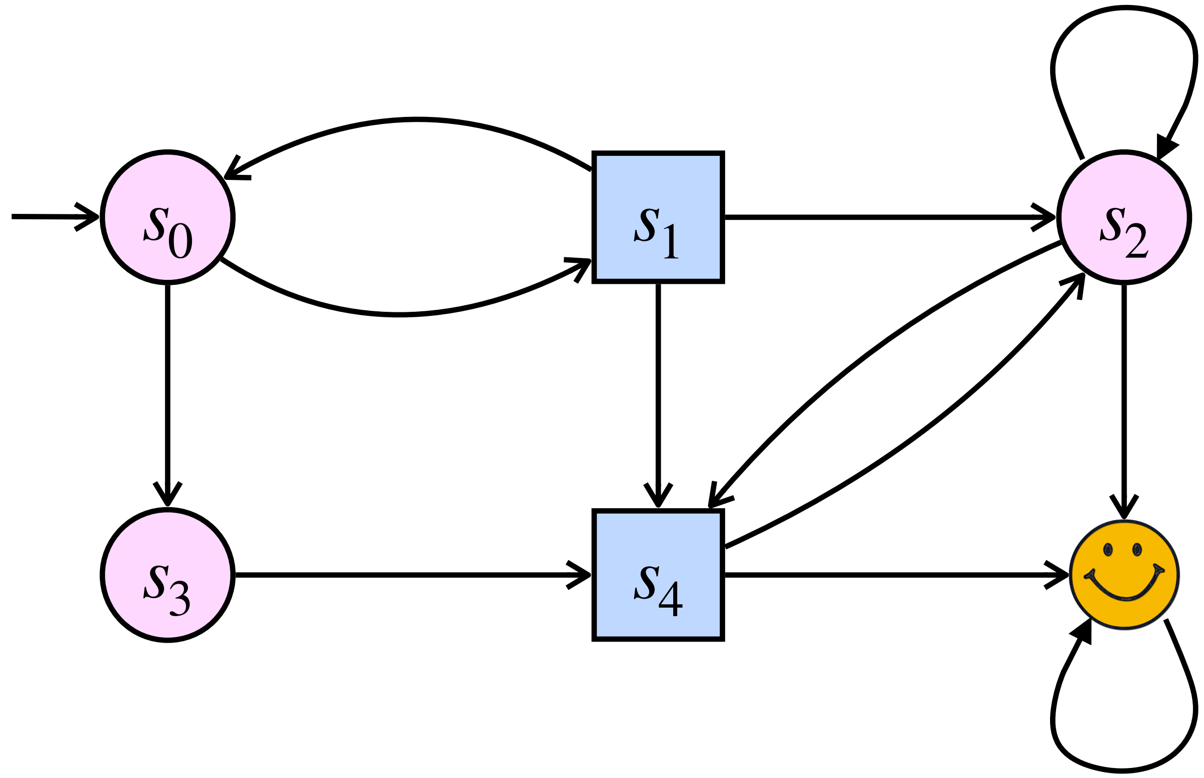
Relevant questions



φ = Reach(☺)

- Can P_1 win the game, i.e. does P_1 have a winning strategy?

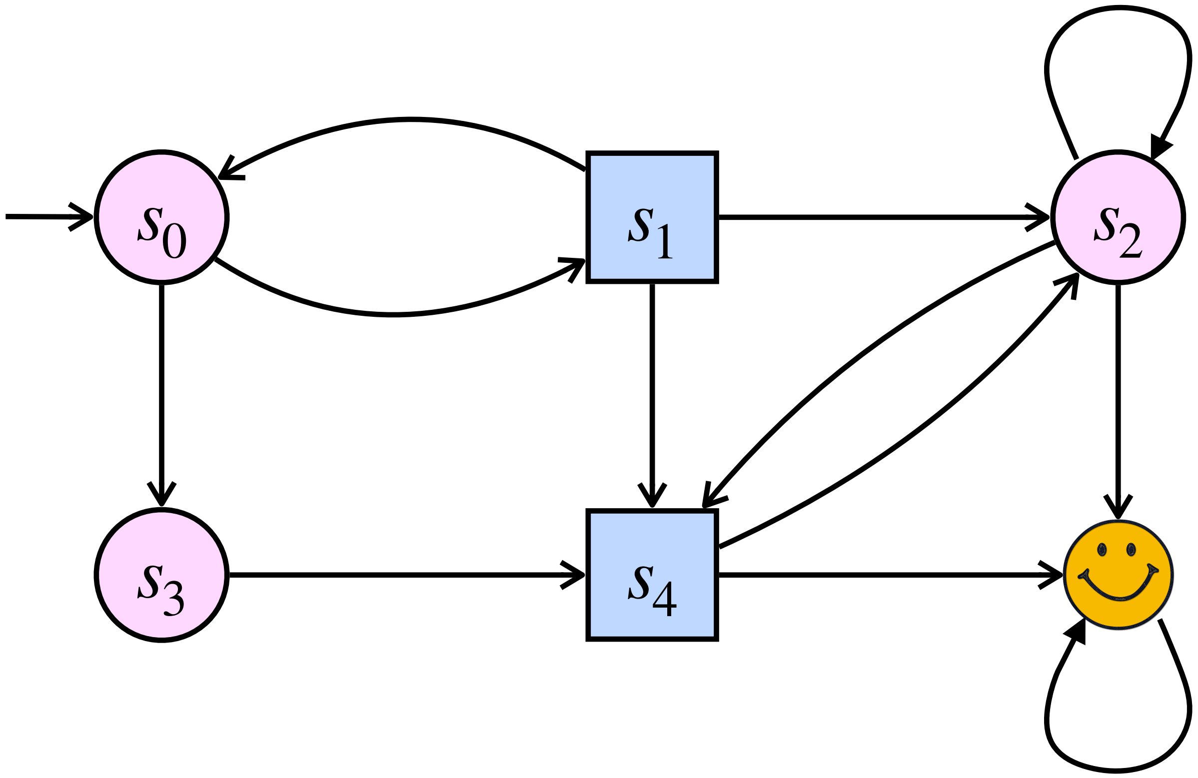
Relevant questions



φ = Reach(☺)

- Can P_1 win the game, i.e. does P_1 have a winning strategy?
- Is there an effective (efficient) way of winning?

Relevant questions



φ = Reach(☺)

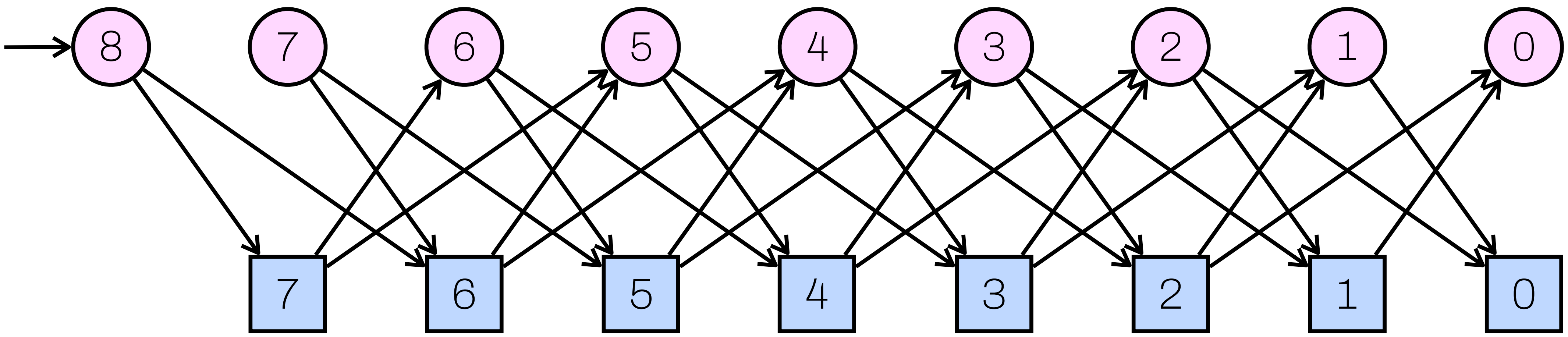
- Can P_1 win the game, i.e. does P_1 have a winning strategy?
- Is there an effective (efficient) way of winning?
- How complex is it to win?

Example: the Nim game

- Players alternate
- Each player can take one or two sticks
- The player who takes the last one wins
- P_1 starts

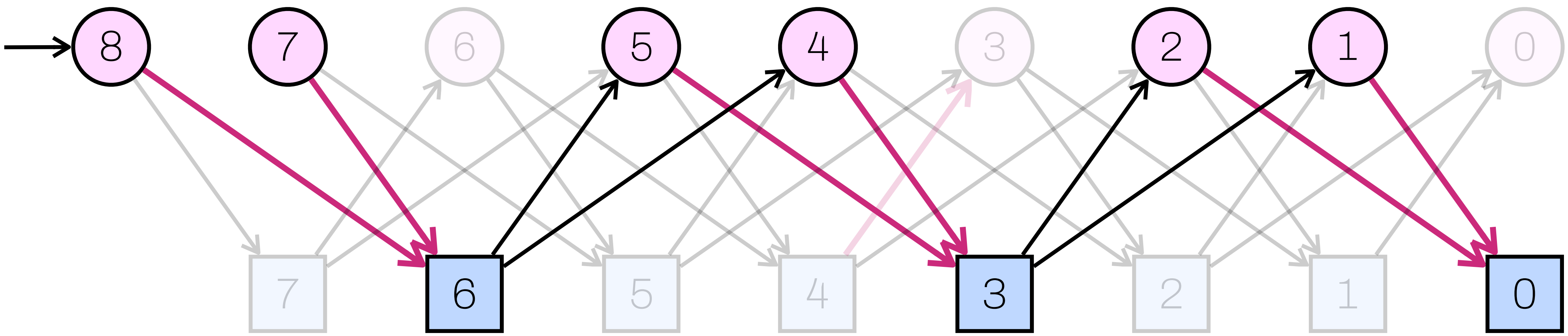
Example: the Nim game

- Players alternate
- Each player can take one or two sticks
- The player who takes the last one wins
- P_1 starts



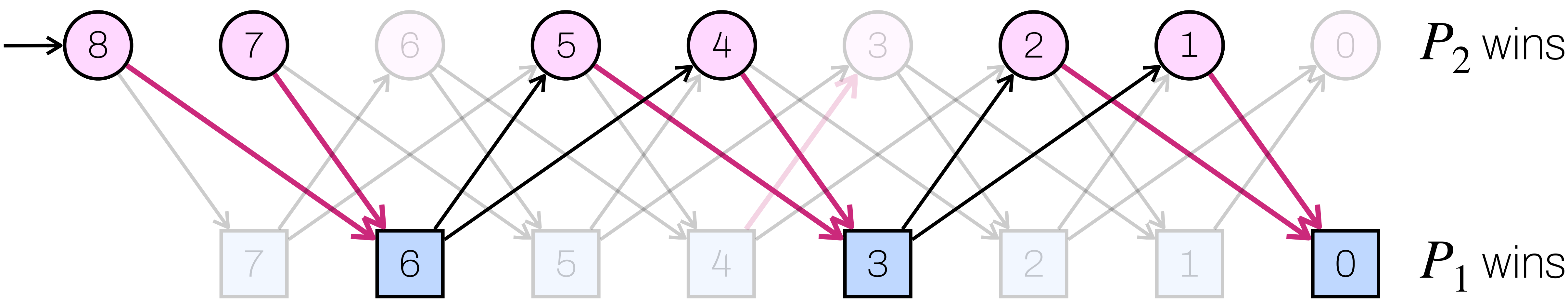
Example: the Nim game

- Players alternate
- Each player can take one or two sticks
- The player who takes the last one wins
- P_1 starts



Example: the Nim game

- Players alternate
- Each player can take one or two sticks
- The player who takes the last one wins
- P_1 starts

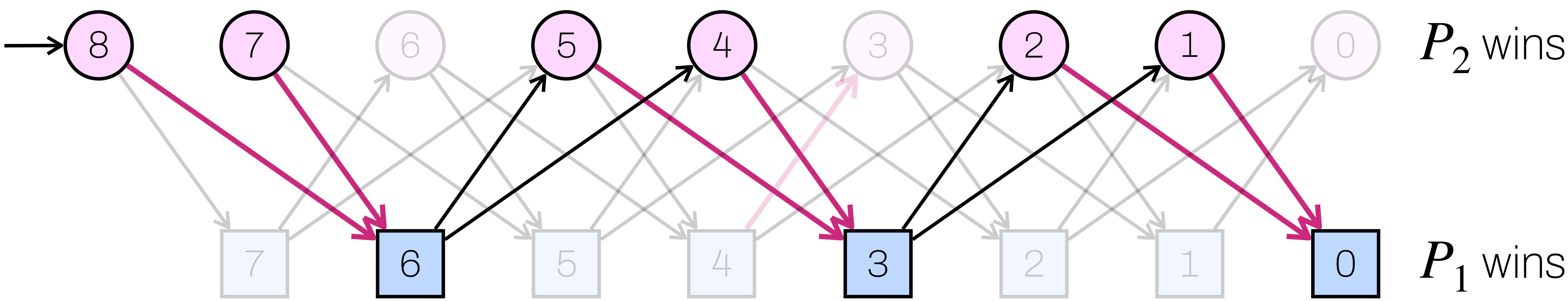


P_1 wins

- from all ◯ $\equiv 1$ or $2 \mod 3$
- from all ◻ $\equiv 0 \mod 3$

Example: the Nim game

- Players alternate
- Each player can take one or two sticks
- The player who takes the last one wins
- P_1 starts



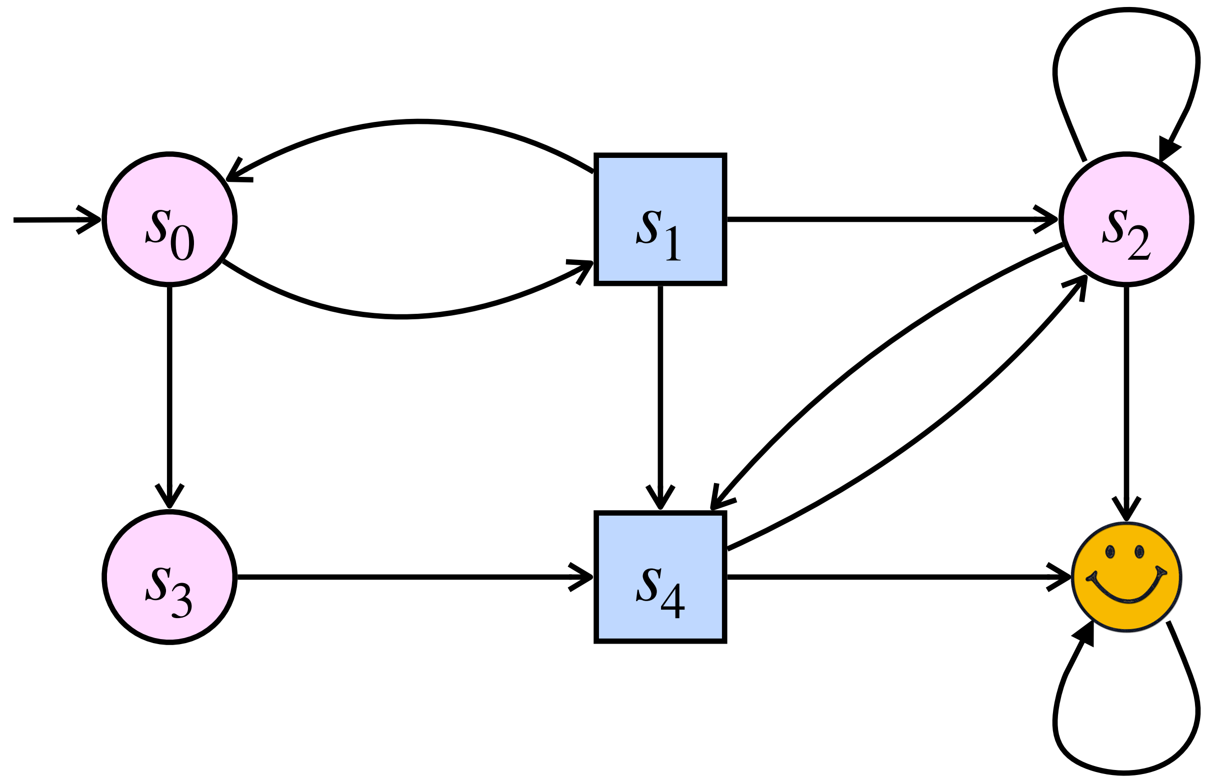
P_1 wins

- from all ○ $\equiv 1$ or $2 \mod 3$
- from all □ $\equiv 0 \mod 3$

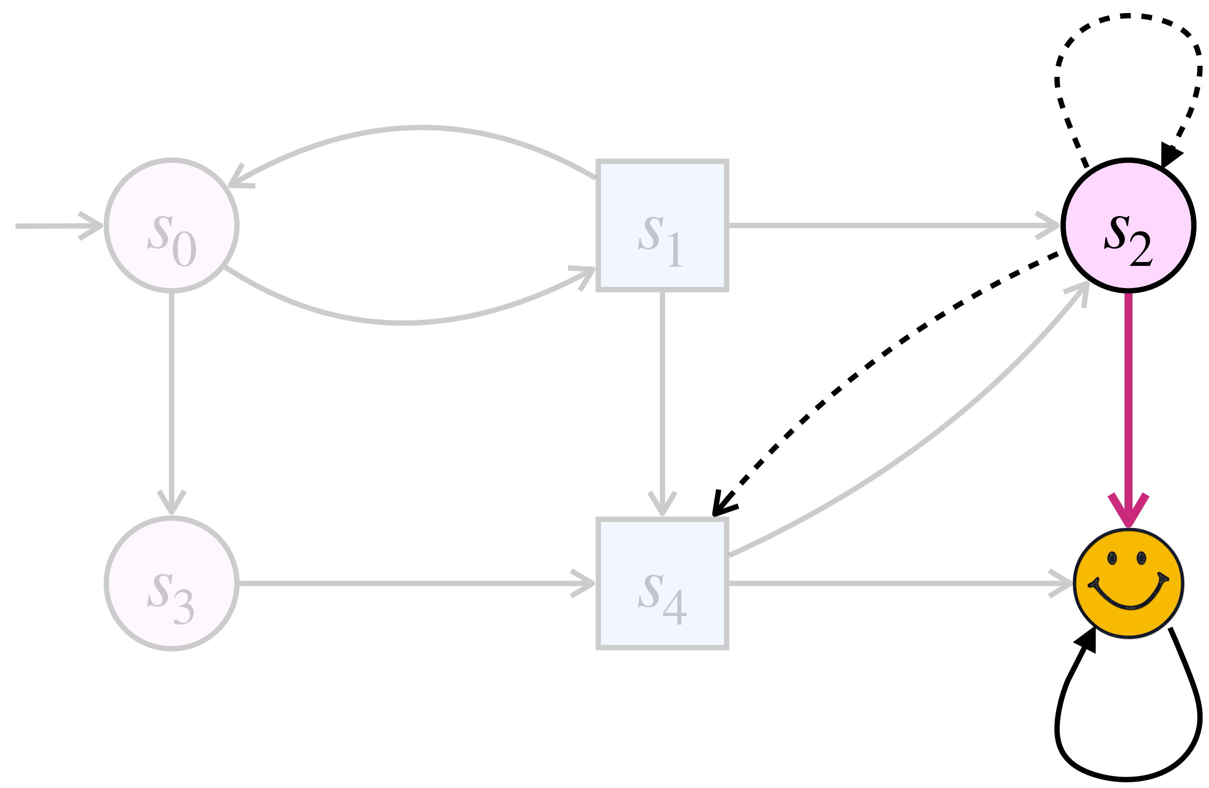
P_2 wins

- from all ○ $\equiv 0 \mod 3$
- from all □ $\equiv 1$ or $2 \mod 3$

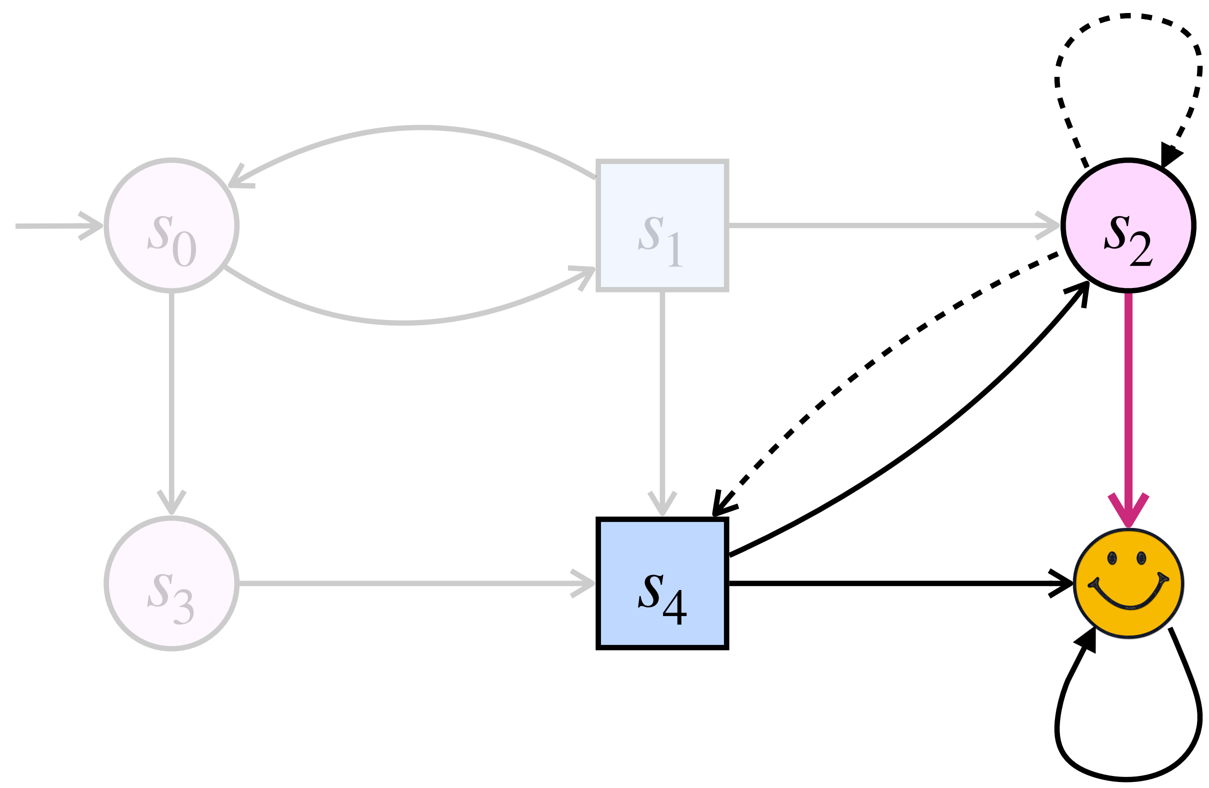
Computation of winning states in the running example



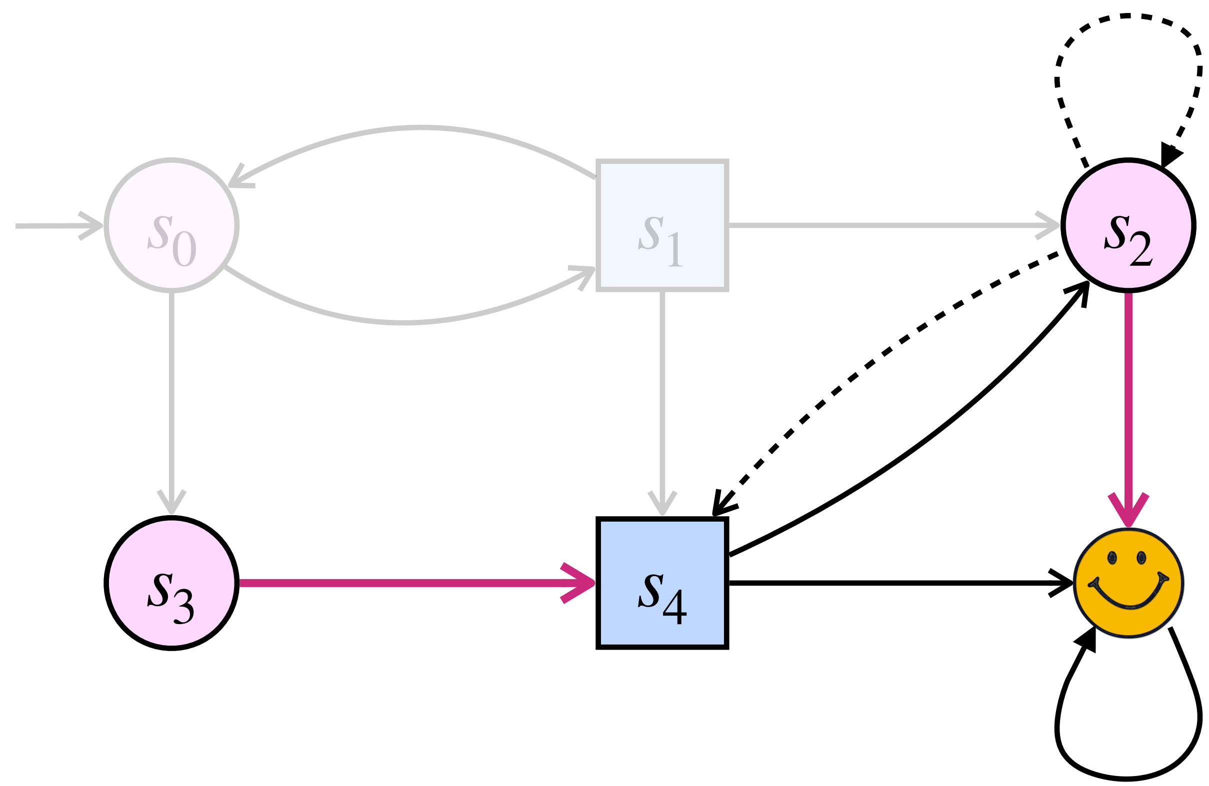
Computation of winning states in the running example



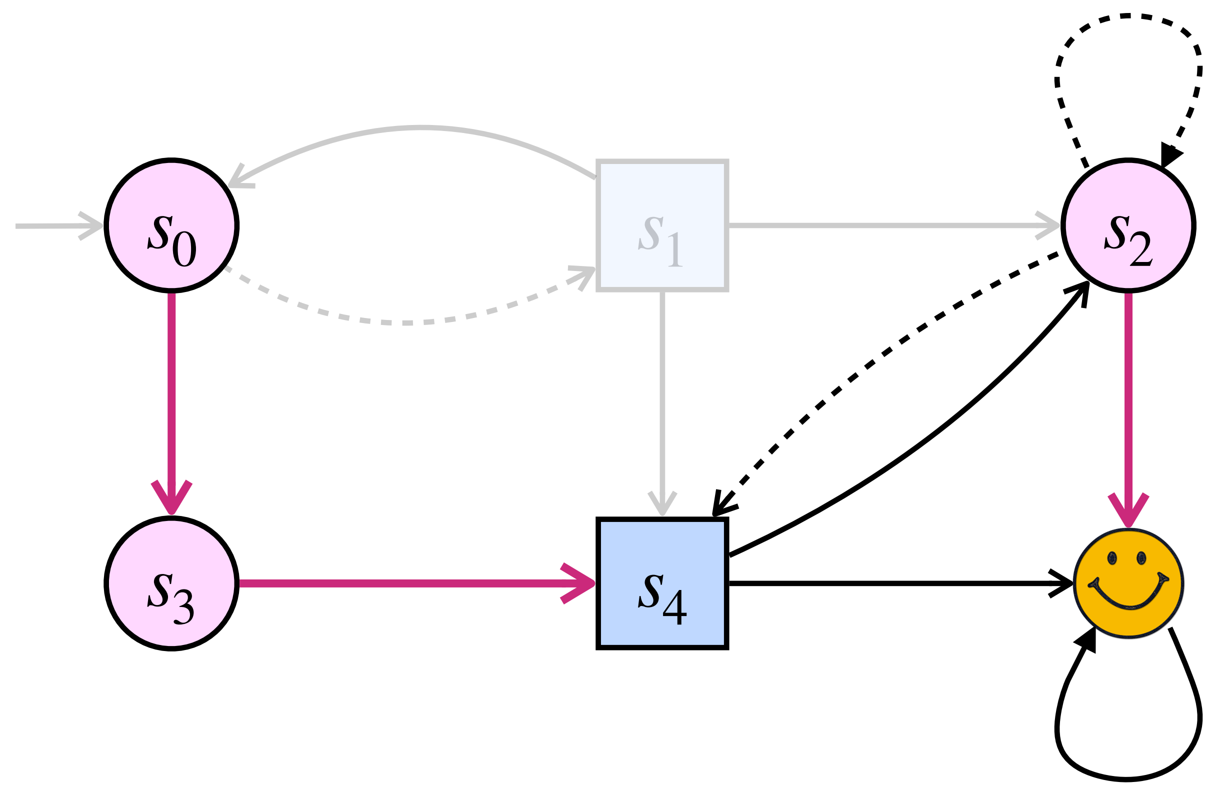
Computation of winning states in the running example



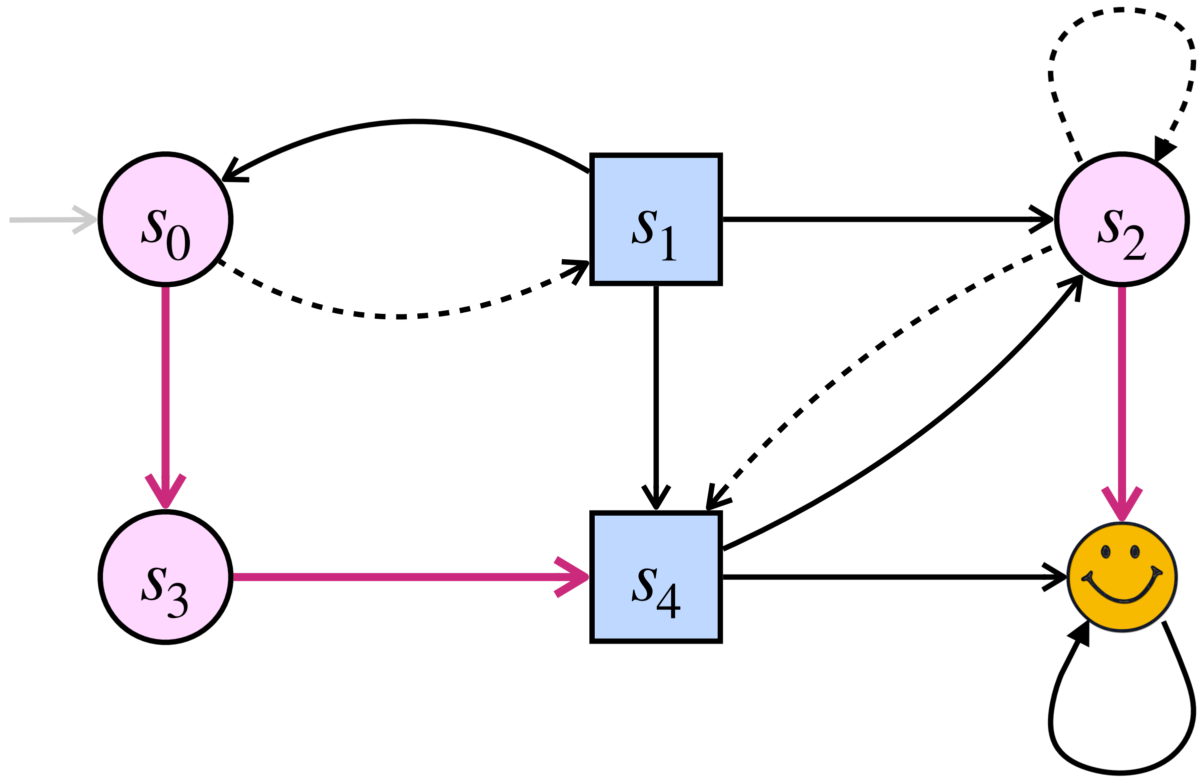
Computation of winning states in the running example



Computation of winning states in the running example

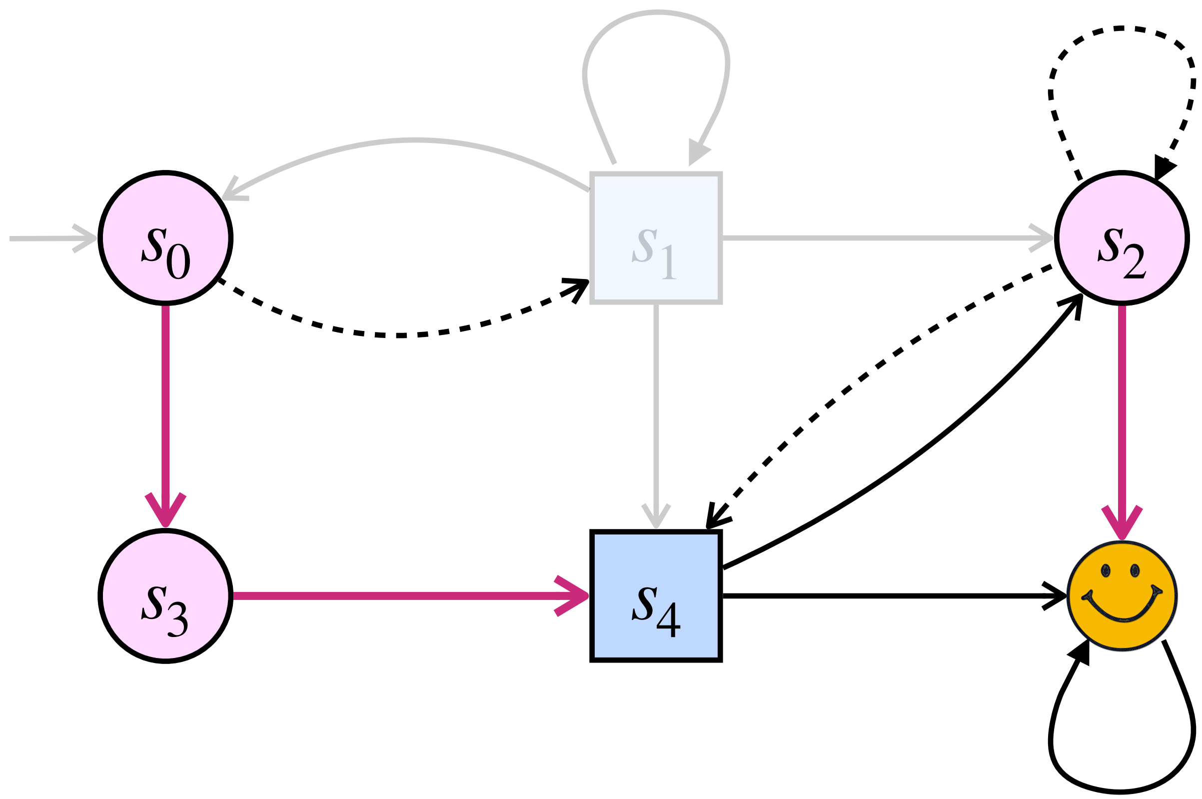


Computation of winning states in the running example



All states are winning for P_1

Computation of winning states in the running example



One state is not winning for P_1

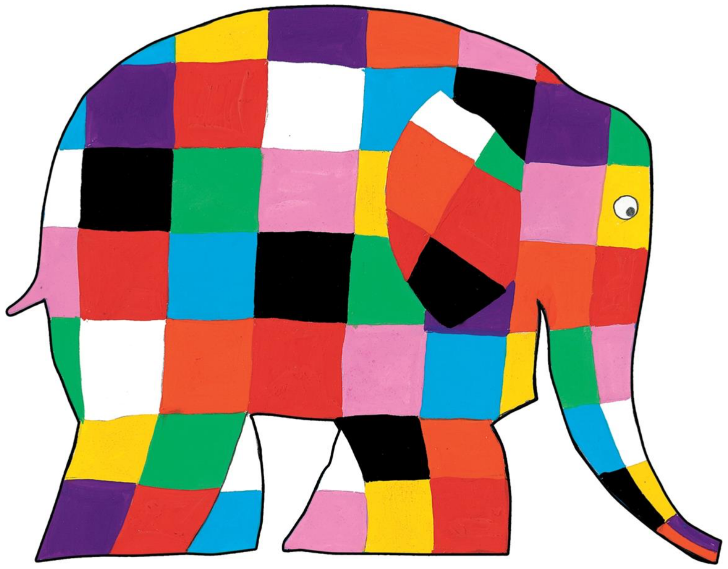
It is winning for P_2

What we do not consider

- Concurrent games
- Stochastic games and strategies
  - Values
  - Determinacy of Blackwell games
- Partial information

Families of strategies

Families of strategies



General strategies

$$\sigma_i : S^*S_i \rightarrow E$$

- May use any information of the past execution
- Information used is therefore potentially infinite
- Not adequate if one targets implementation

On the simplest side: positional strategies

From $\sigma_i : S^*S_i \to E$ to $\sigma_i : S_i \to E$

On the simplest side: positional strategies

From $\sigma_i : S^*S_i \rightarrow E$ to $\sigma_i : S_i \rightarrow E$

- Positional = memoryless

On the simplest side: positional strategies

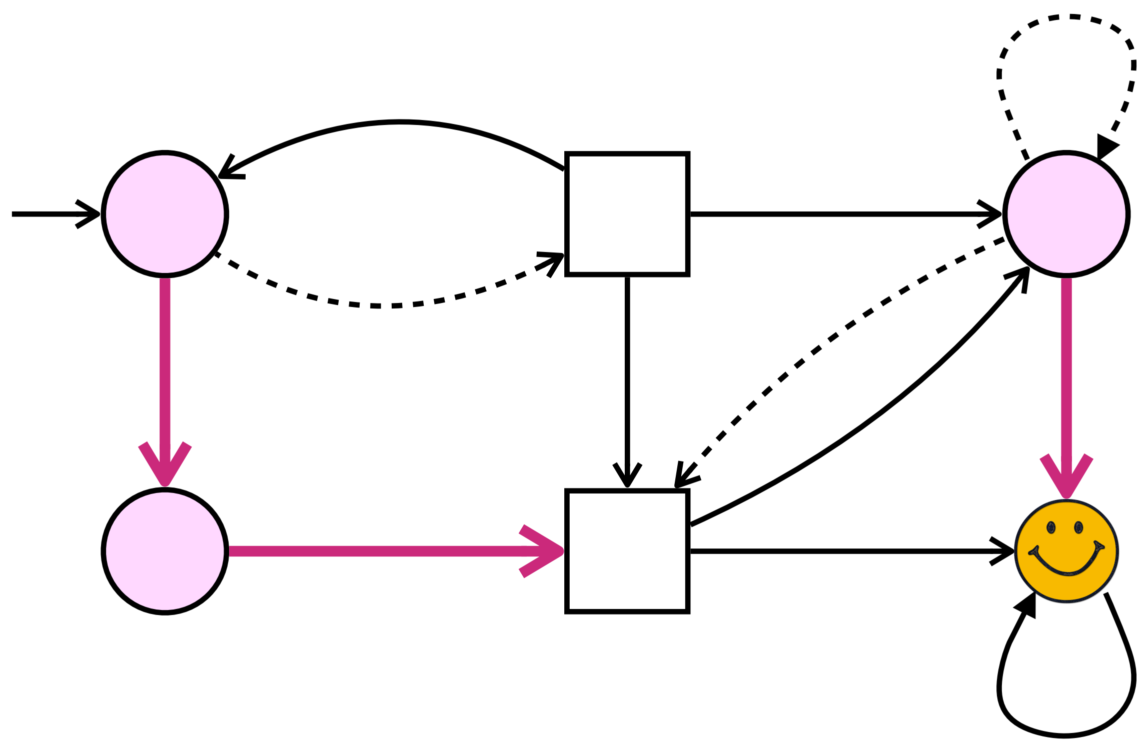
From $\sigma_i : S^*S_i \to E$ to $\sigma_i : S_i \to E$

- Positional = memoryless
- Reachability, parity, mean-payoff, positive energy, …
  $\longrightarrow$ positional strategies are sufficient to win

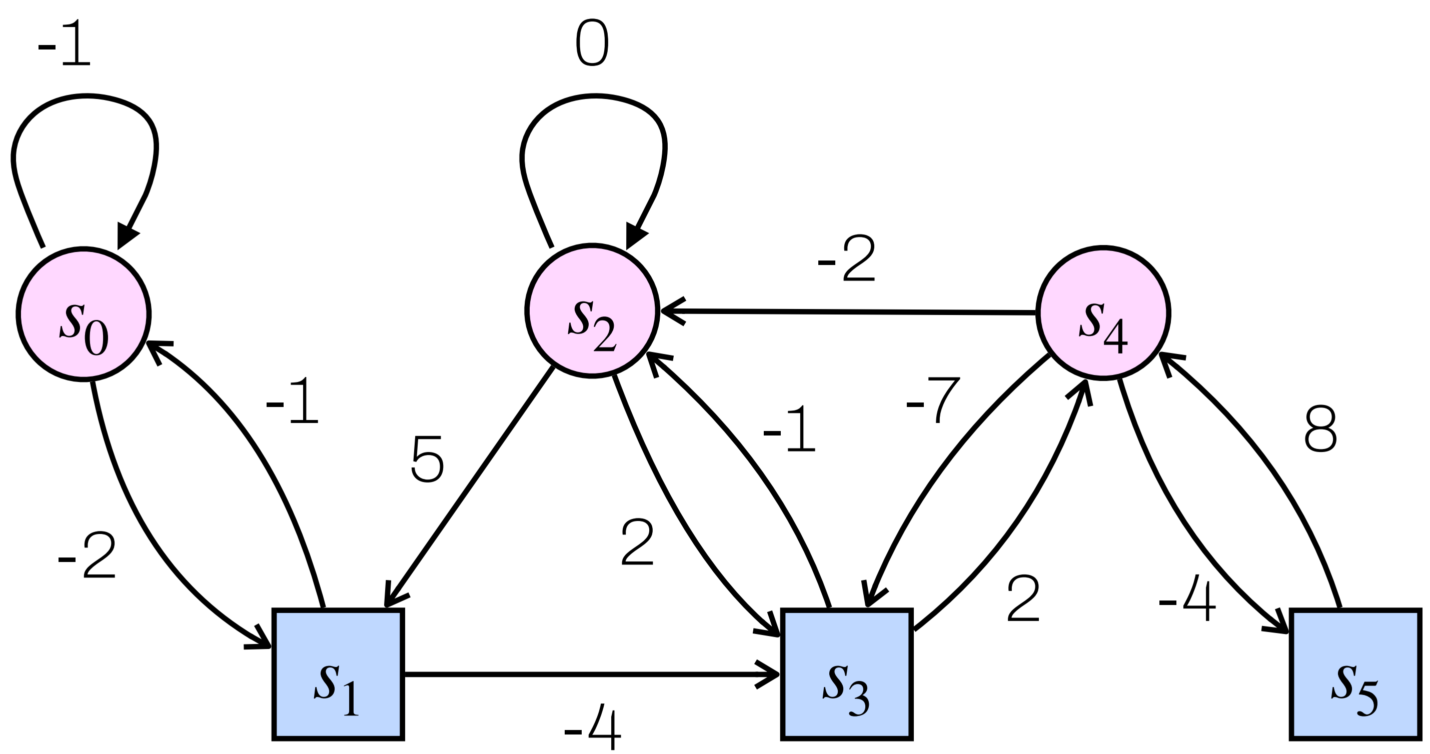
On the simplest side: positional strategies

From $\sigma_i : S^*S_i \to E$ to $\sigma_i : S_i \to E$

- Positional = memoryless
- Reachability, parity, mean-payoff, positive energy, …
  → positional strategies are sufficient to win



Example: mean-payoff

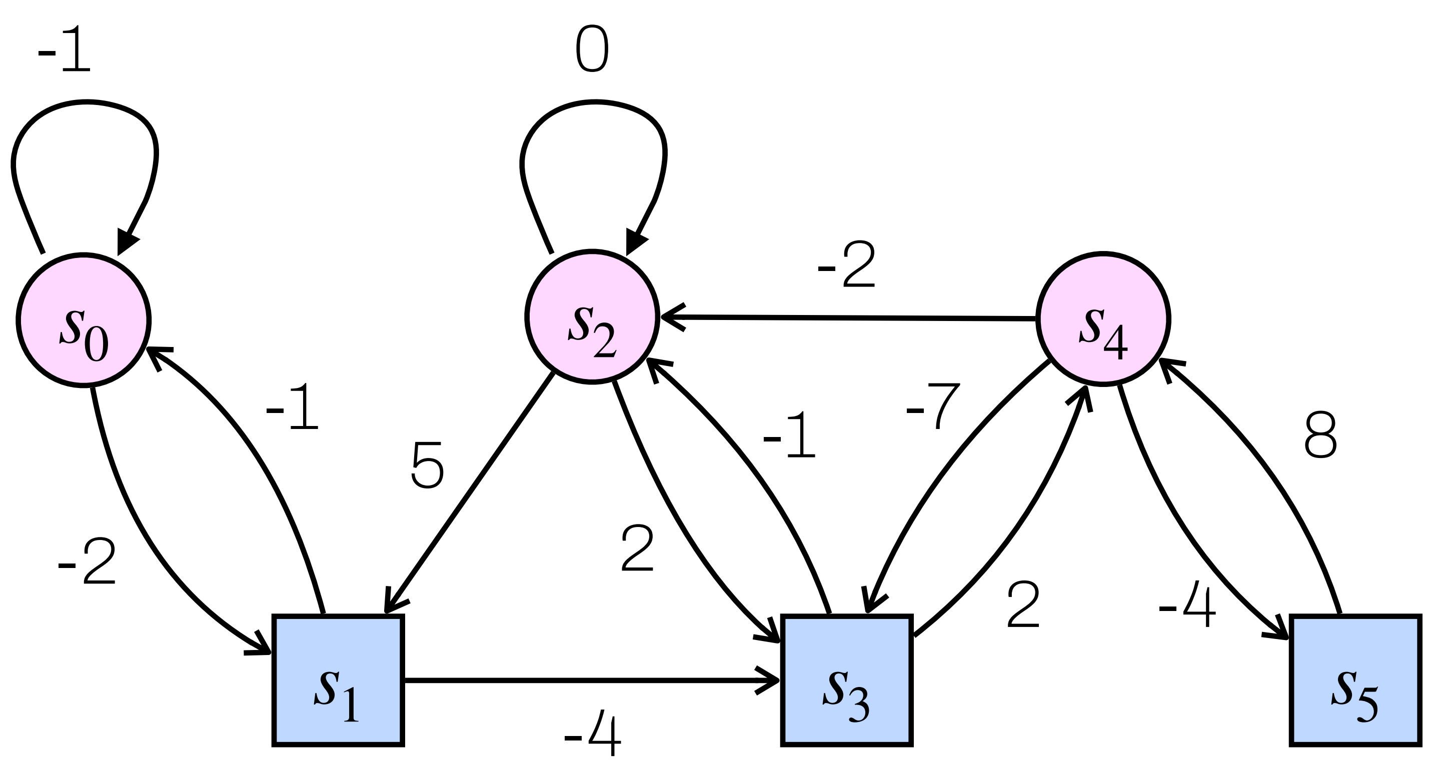


[Ohl21] Ohlmann. Monotonic graphs for Parity and Mean-Payoff games (PhD thesis).

Example: mean-payoff

- P_1 maximizes, P_2 minimizes

$$\overline{\mathsf{MP}} = \limsup_n \frac{\sum_{i \neq n} c_i}{n}$$

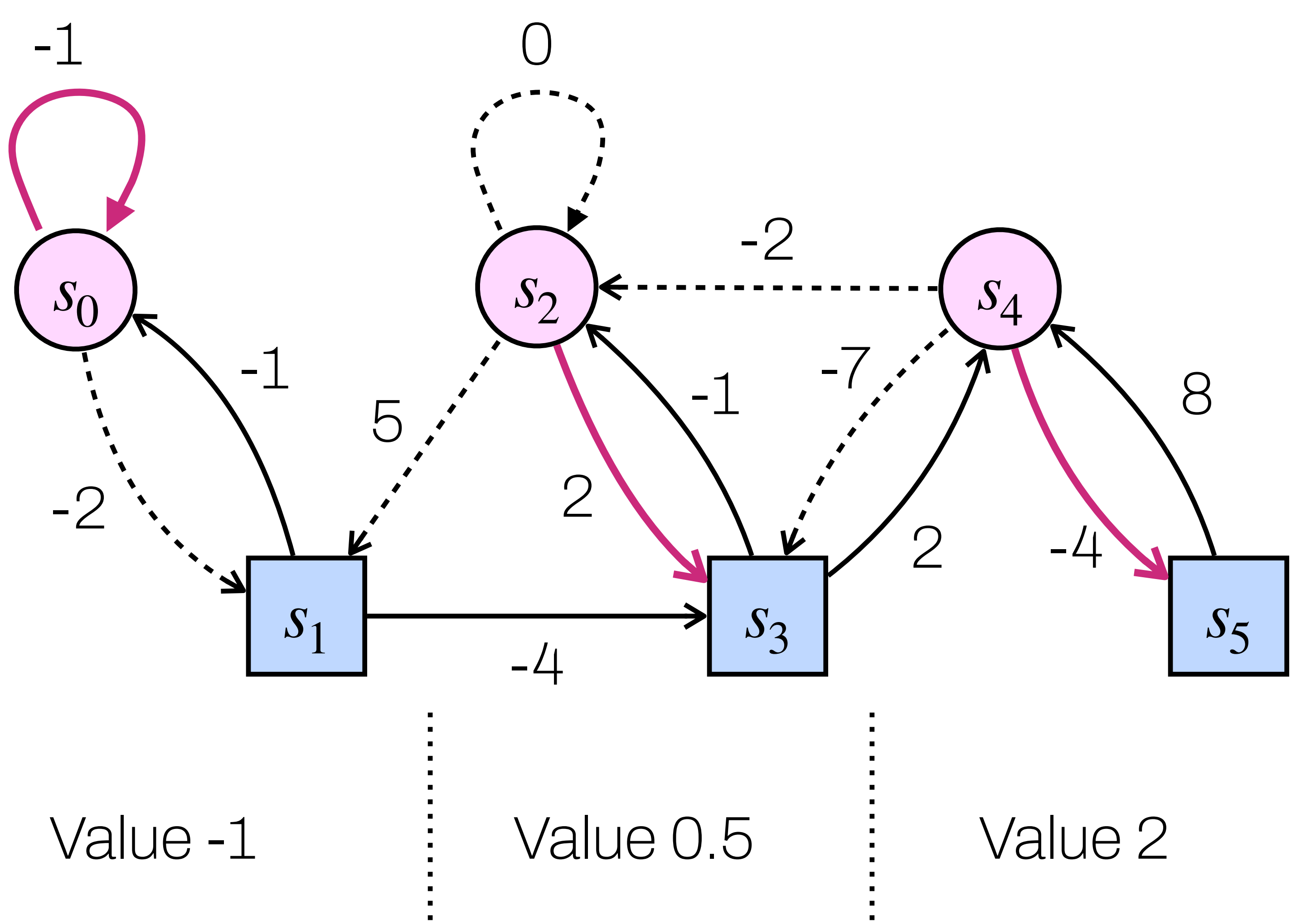


[Ohl21] Ohlmann. Monotonic graphs for Parity and Mean-Payoff games (PhD thesis).

Example: mean-payoff

- P_1 maximizes, P_2 minimizes
- Positional strategies are sufficient to win

$$\overline{\mathrm{MP}} = \limsup_n \frac{\sum_{i \neq n} c_i}{n}$$



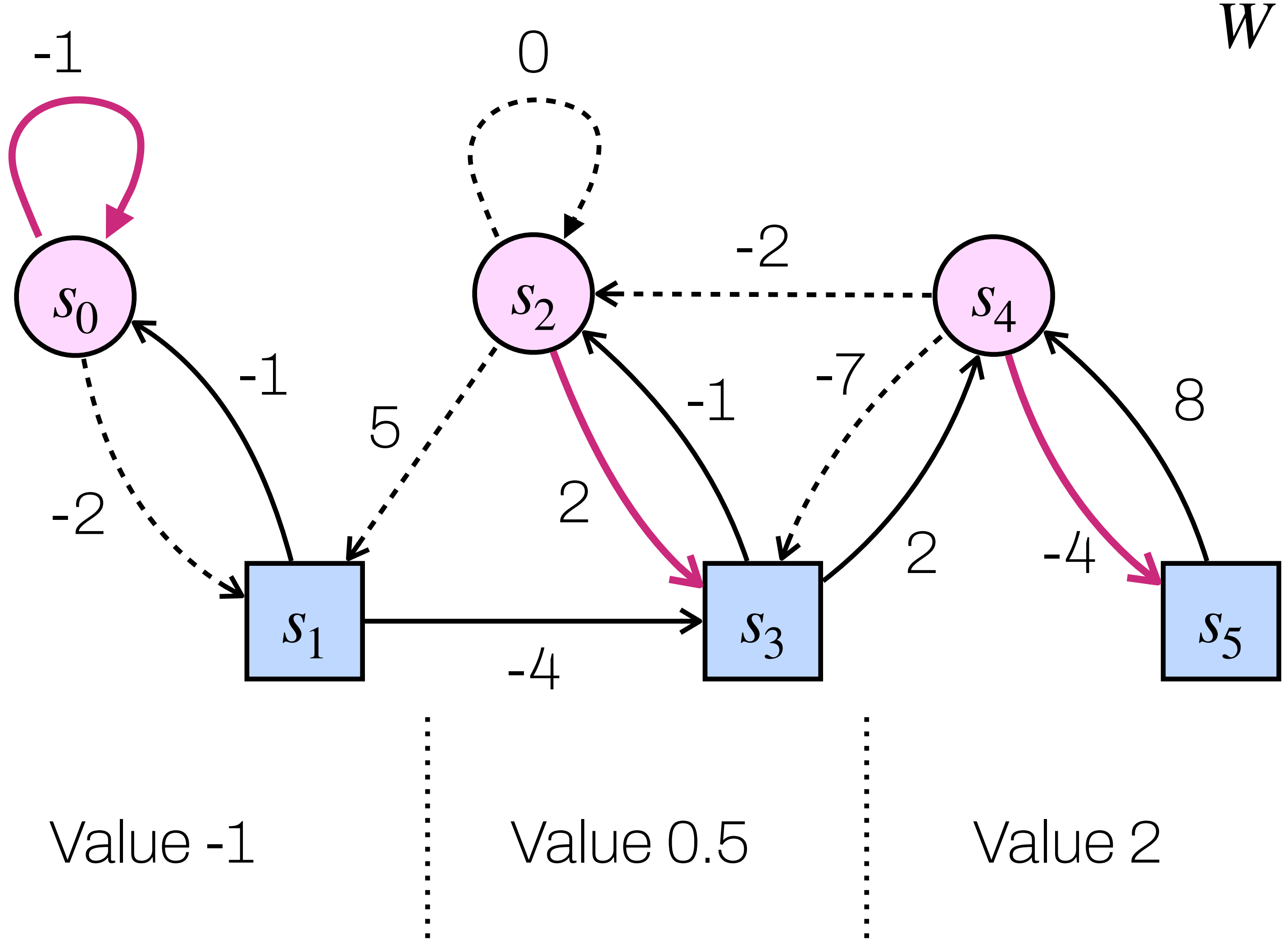
[Ohl21] Ohlmann. Monotonic graphs for Parity and Mean-Payoff games (PhD thesis).

Example: mean-payoff

- P_1 maximizes, P_2 minimizes
- Positional strategies are sufficient to win

$$\overline{\mathsf{MP}} = \limsup_n \frac{\sum_{i \neq n} c_i}{n}$$

$$W = (\overline{\mathsf{MP}} \geq 0)$$



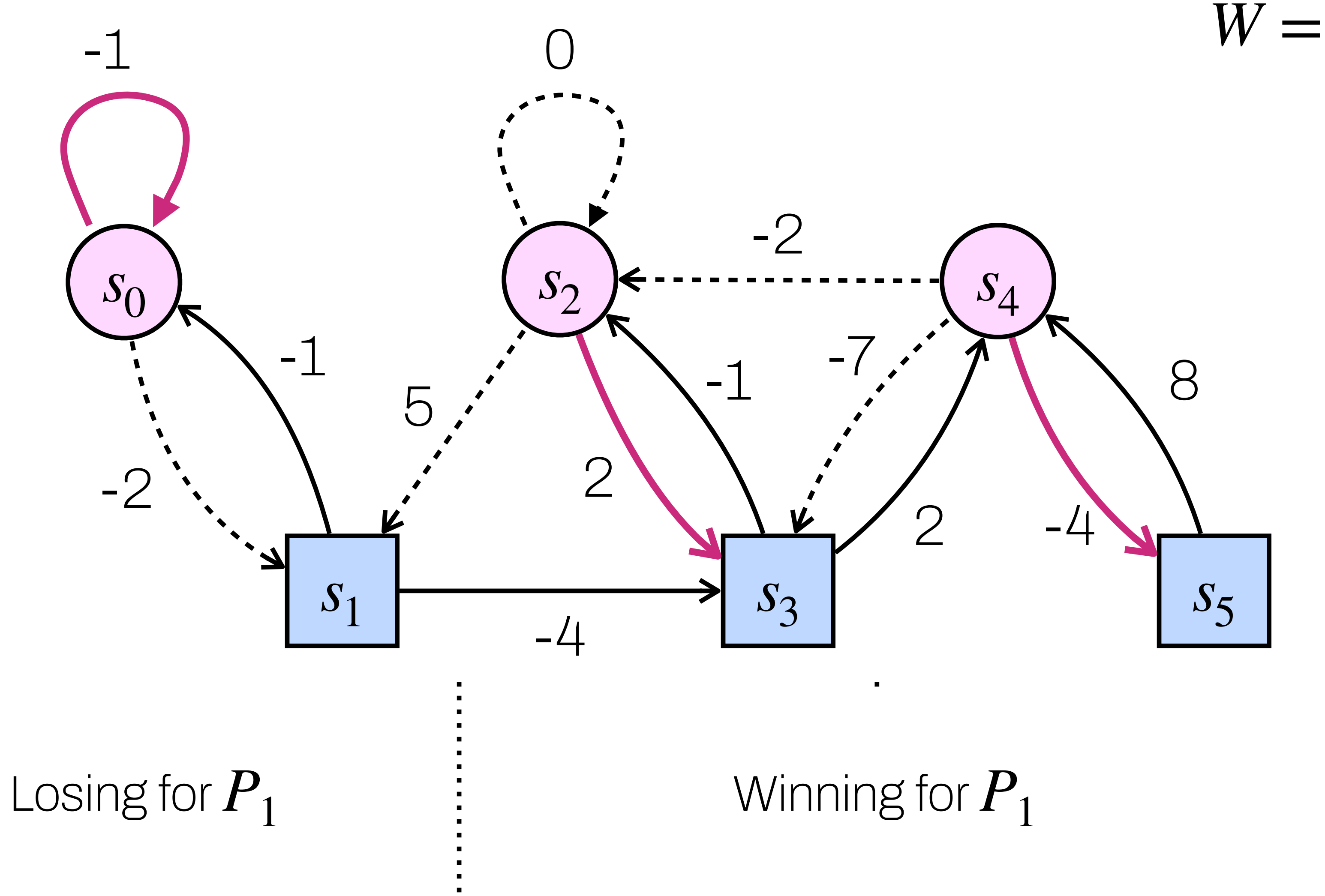
[Ohl21] Ohlmann. Monotonic graphs for Parity and Mean-Payoff games (PhD thesis).

Example: mean-payoff

- P_1 maximizes, P_2 minimizes
- Positional strategies are sufficient to win

$$\overline{\mathrm{MP}} = \limsup_n \frac{\sum_{i \neq n} c_i}{n}$$

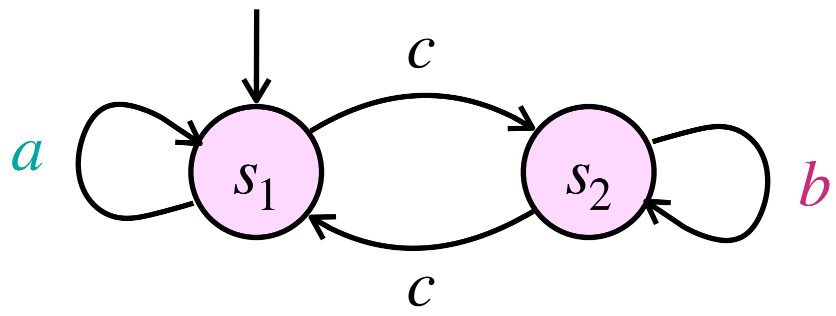
$$W = (\overline{\mathrm{MP}} \geq 0)$$



[Ohl21] Ohlmann. Monotonic graphs for Parity and Mean-Payoff games (PhD thesis).

Do we need more?

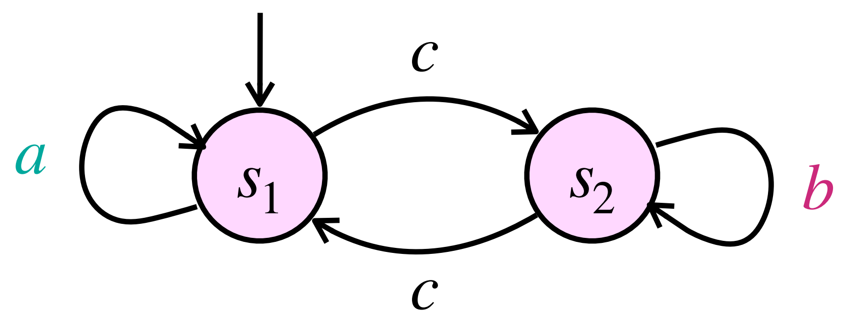
Examples



« See infinitely often both a and b »

Büchi$(a) \wedge$ Büchi(b)

Examples



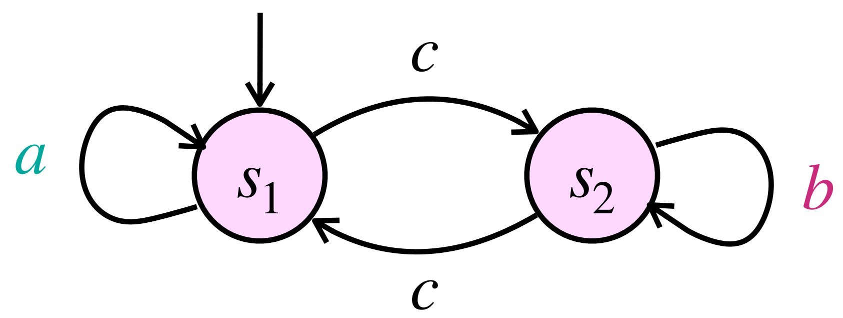
« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)

Winning strategy

- At each visit to s_1, loop once in s_1 and then go to s_2
- At each visit to s_2, loop once in s_2 and then go to s_1
- Generates the sequence $(acbc)^\omega$

Examples

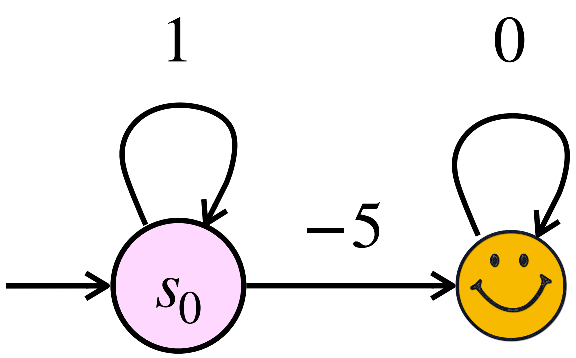


« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)

Winning strategy

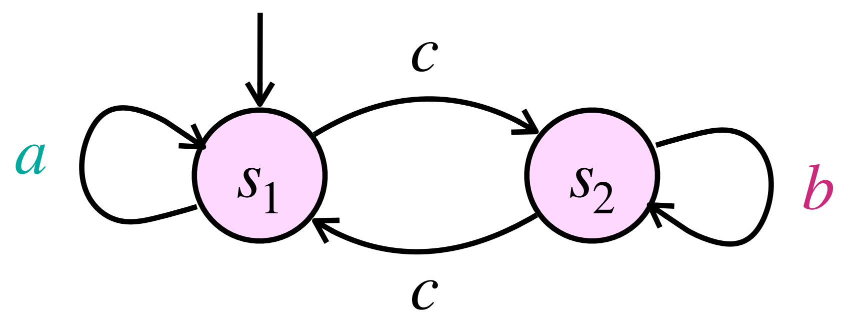
- At each visit to s_1, loop once in s_1 and then go to s_2
- At each visit to s_2, loop once in s_2 and then go to s_1
- Generates the sequence $(acbc)^\omega$



« Reach the target with energy level 0 »

$\mathbf{FG}\ (\mathsf{EL} = 0)$

Examples

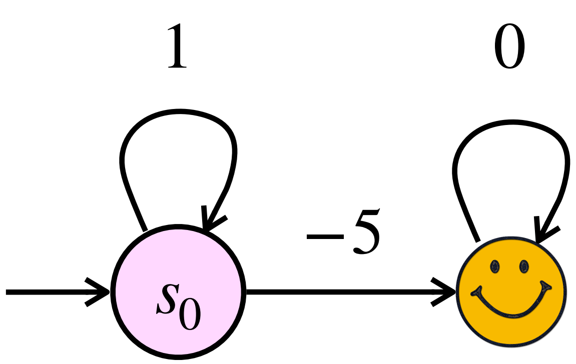


« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)

Winning strategy

- At each visit to s_1, loop once in s_1 and then go to s_2
- At each visit to s_2, loop once in s_2 and then go to s_1
- Generates the sequence $(acbc)^\omega$



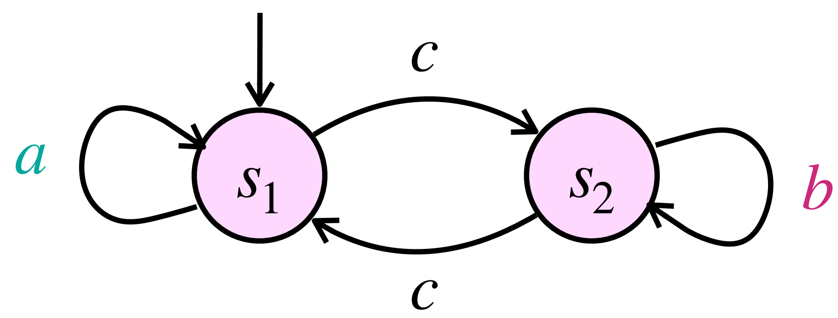
« Reach the target with energy level 0 »

$\mathbf{FG}\ (\text{EL} = 0)$

Winning strategy

- Loop five times in s_0
- Then go to the target
- Generates the sequence of colors $1\ 1\ 1\ 1\ 1\ \ -5\ 0\ 0\ 0\ldots$

Examples

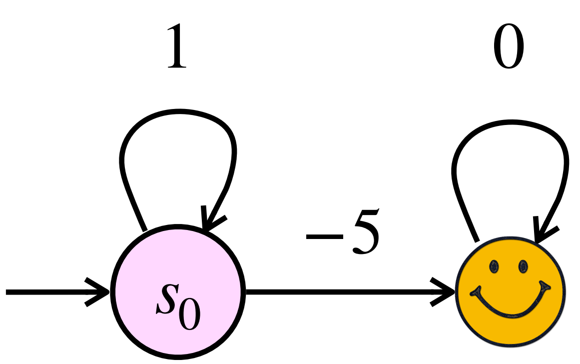


« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)

Winning strategy

- At each visit to s_1, loop once in s_1 and then go to s_2
- At each visit to s_2, loop once in s_2 and then go to s_1
- Generates the sequence $(acbc)^\omega$



« Reach the target with energy level 0 »

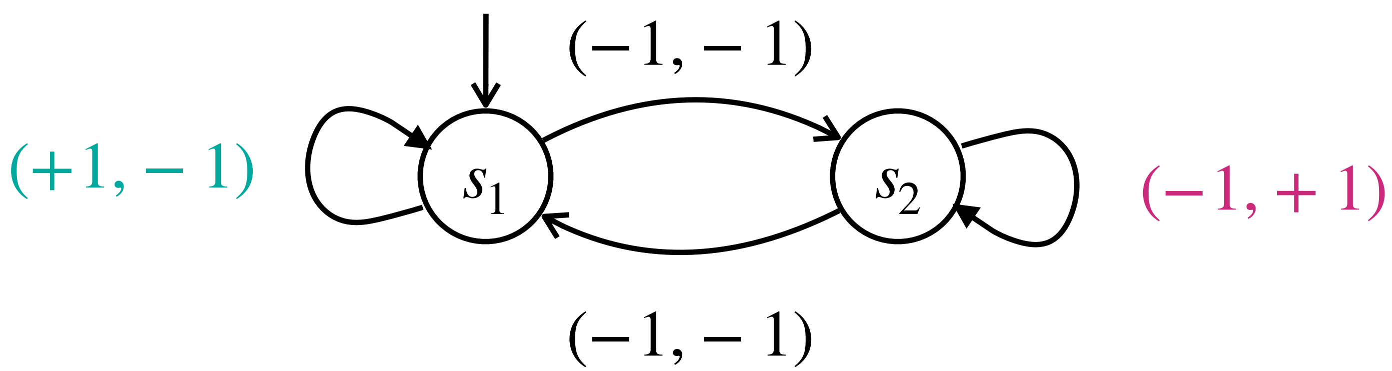
$\mathbf{FG}\ (\text{EL} = 0)$

Winning strategy

- Loop five times in s_0
- Then go to the target
- Generates the sequence of colors $1\ 1\ 1\ 1\ 1\ \ -5\ 0\ 0\ 0\ldots$

These two strategies require only **finite** memory

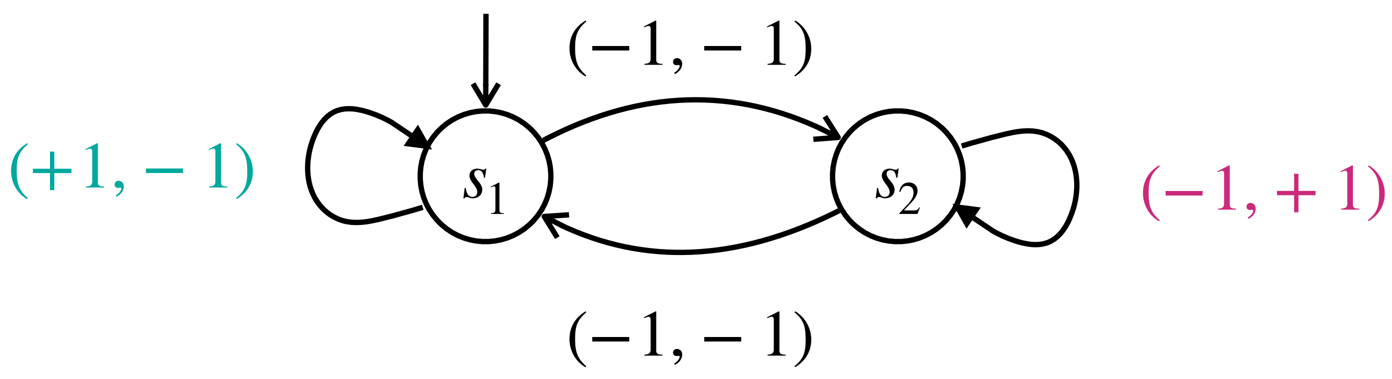
Example: multi-dimensional mean-payoff



« Have a (limsup) mean-payoff ≥ 0 on both dimensions »

So-called *multi-dimensional mean-payoff*

Example: multi-dimensional mean-payoff



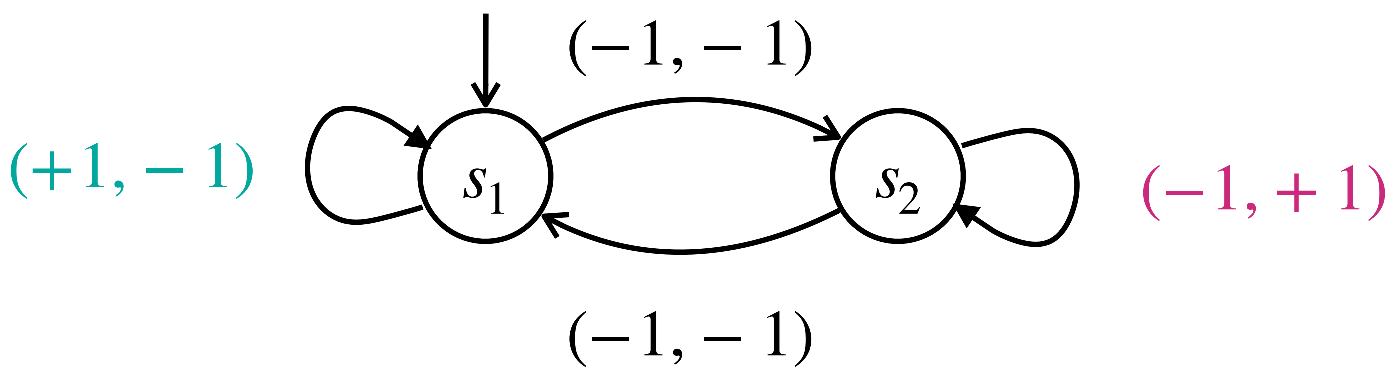
« Have a (limsup) mean-payoff ≥ 0 on both dimensions »

So-called *multi-dimensional mean-payoff*

Winning strategy

- After k-th switch between s_1 and s_2, loop $2k - 1$ times and then switch back
- Generates the sequence
  $(-1, -1)$ $(-1, +1)$ $(-1, -1)(+1, -1)(+1, -1)(+1, -1)$ $(-1, -1)$
  $(-1, +1)(-1, +1)(-1, +1)(-1, +1)(-1, +1)$ $(-1, -1)$
  $(+1, -1)(+1, -1)(+1, -1)(+1, -1)(+1, -1)(+1, -1)(+1, -1)$ $(-1, -1)$...

Example: multi-dimensional mean-payoff



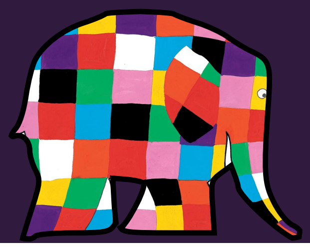
« Have a (limsup) mean-payoff ≥ 0 on both dimensions »
So-called *multi-dimensional mean-payoff*

Winning strategy

- After k-th switch between s_1 and s_2, loop $2k-1$ times and then switch back
- Generates the sequence
$(-1,-1)\,(-1,+1)\,(-1,-1)(+1,-1)(+1,-1)(+1,-1)\,(-1,-1)$
$(-1,+1)(-1,+1)(-1,+1)(-1,+1)(-1,+1)\,(-1,-1)$
$(+1,-1)(+1,-1)(+1,-1)(+1,-1)(+1,-1)(+1,-1)(+1,-1)\,(-1,-1)$...

This strategy requires **infinite** memory, and this is unavoidable

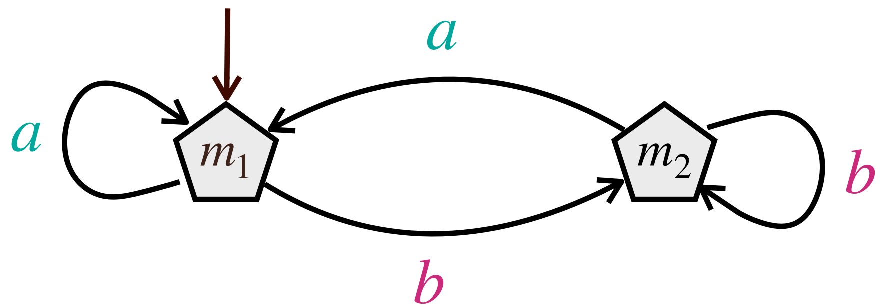
We focus on finite memory!



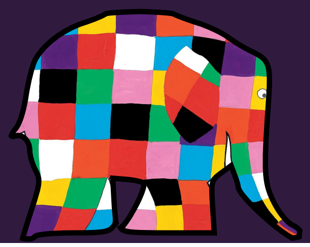
Chromatic* memory

Memory skeleton

$\mathcal{M} = (M, m_{\text{init}}, \alpha_{\text{upd}})$ with $m_{\text{init}} \in M$ and $\alpha_{\text{upd}} : M \times C \to M$



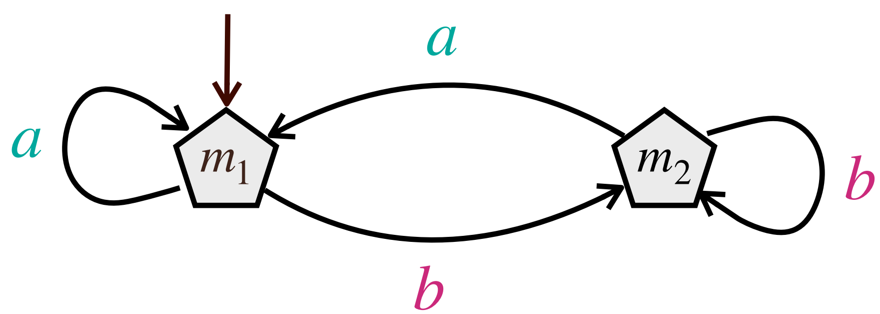
* Terminology by Kopczyński



Chromatic* memory

Memory skeleton

$$\mathscr{M} = (M, m_{\text{init}}, \alpha_{\text{upd}}) \text{ with } m_{\text{init}} \in M \text{ and } \alpha_{\text{upd}} : M \times C \to M$$



Not yet a strategy!

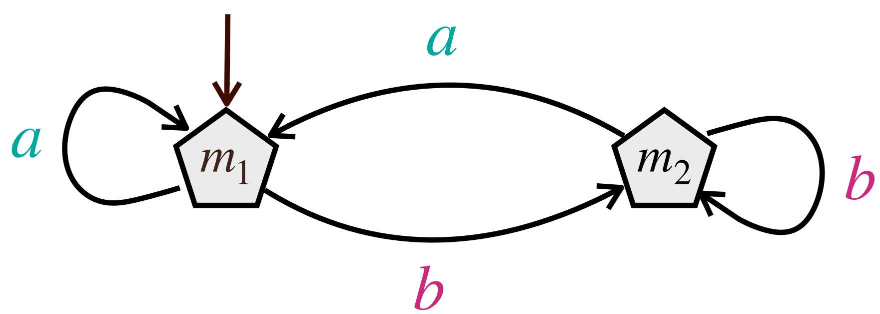
$\sigma_i : S^*S_i \to E$

* Terminology by Kopczyński

Chromatic* memory

Memory skeleton

$\mathcal{M} = (M, m_{\text{init}}, \alpha_{\text{upd}})$ with $m_{\text{init}} \in M$ and $\alpha_{\text{upd}} : M \times C \to M$



Not yet a strategy!

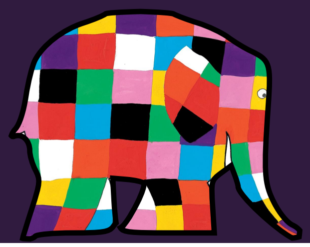
$\sigma_i : S^*S_i \to E$

Strategy with memory $\mathcal{M}$

Additional next-move function $\alpha_{\text{next}} : M \times S_i \to E$

$(\mathcal{M}, \alpha_{\text{next}})$ defines a strategy!

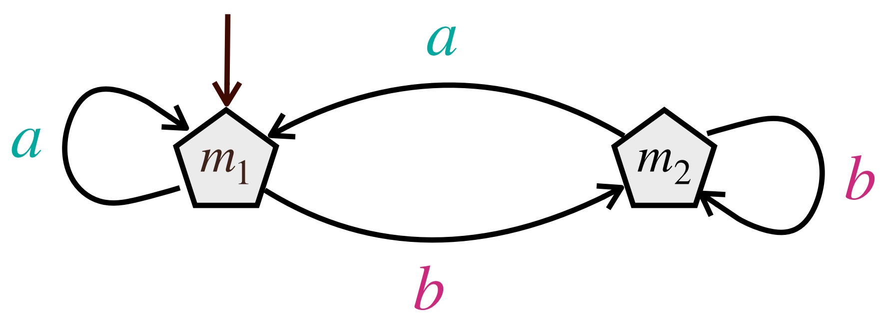
* Terminology by Kopczyński



Chromatic* memory

Memory skeleton

$\mathscr{M} = (M, m_{\text{init}}, \alpha_{\text{upd}})$ with $m_{\text{init}} \in M$ and $\alpha_{\text{upd}} : M \times C \to M$



Not yet a strategy!

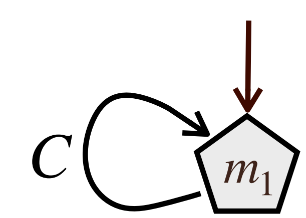
$\sigma_i : S^*S_i \to E$

Strategy with memory $\mathscr{M}$

Additional next-move function $\alpha_{\text{next}} : M \times S_i \to E$

$(\mathscr{M}, \alpha_{\text{next}})$ defines a strategy!

Remark: memoryless strategies are $\mathscr{M}_{\text{triv}}$-strategies, where $\mathscr{M}_{\text{triv}}$ is



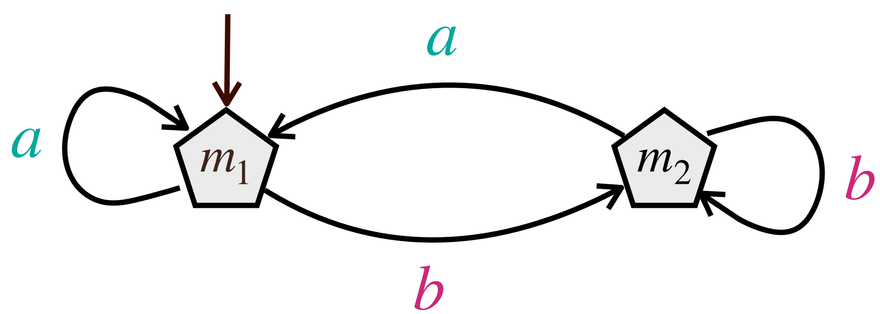
* Terminology by Kopczyński

Chromatic* memory

Memory skeleton

$\mathscr{M} = (M, m_{\text{init}}, \alpha_{\text{upd}})$ with $m_{\text{init}} \in M$ and $\alpha_{\text{upd}} : M \times S \to M$

Chaotic* memory



Not yet a strategy!

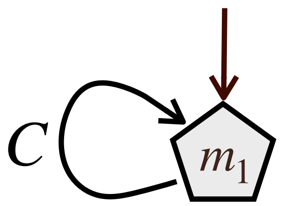
$\sigma_i : S^* S_i \to E$

Strategy with memory $\mathscr{M}$

Additional next-move function $\alpha_{\text{next}} : M \times S_i \to E$

$(\mathscr{M}, \alpha_{\text{next}})$ defines a strategy!

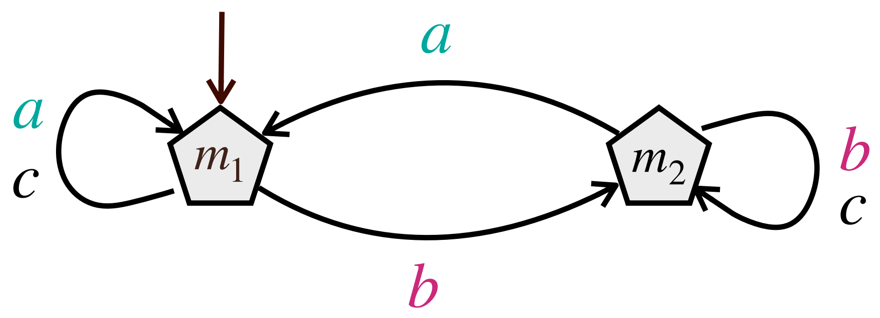
Remark: memoryless strategies are $\mathscr{M}_{\text{triv}}$-strategies, where $\mathscr{M}_{\text{triv}}$ is



* Terminology by Kopczyński

Example of chromatic memory

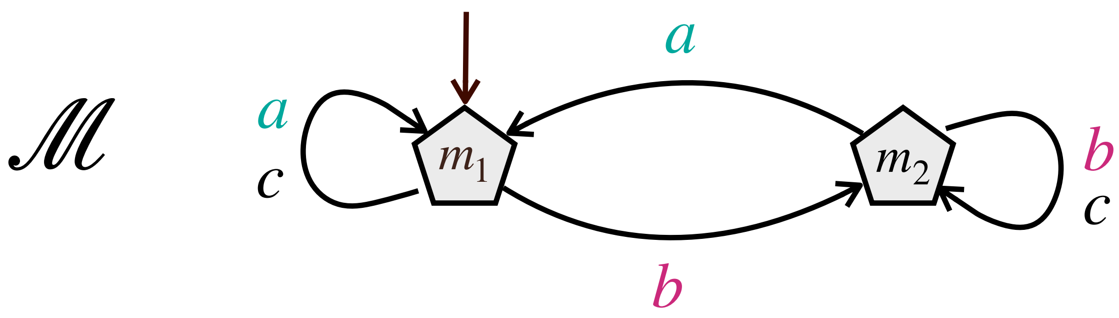
$\mathscr{M}$



This skeleton is sufficient for winning

$W =$ Büchi$(a) \wedge$ Büchi(b) (in any arena)

Example of chromatic memory

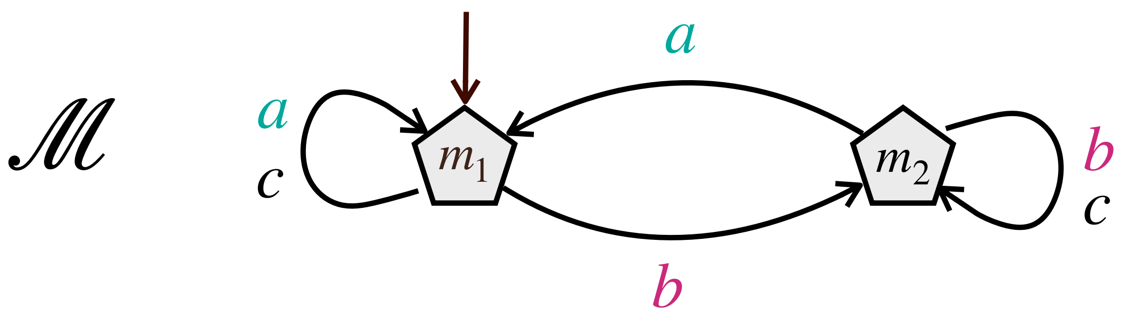


This skeleton is sufficient for winning

$W = \text{Büchi}(a) \wedge \text{Büchi}(b)$ (in any arena)

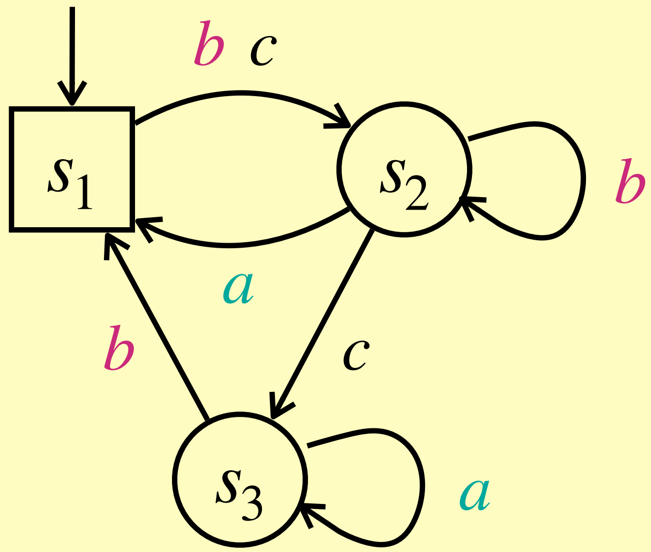
Example

Example of chromatic memory



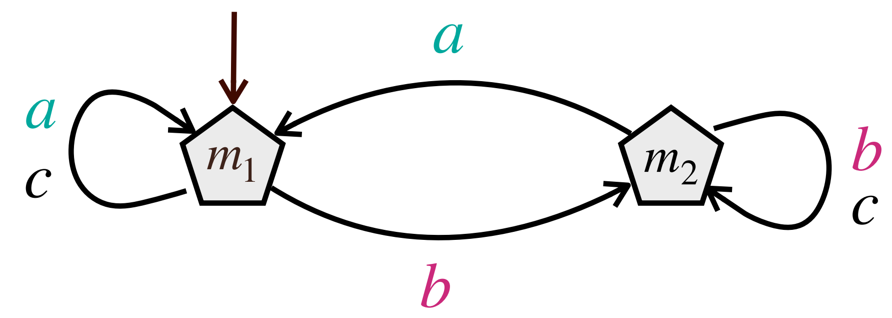
This skeleton is sufficient for winning
$W = \text{Büchi}(a) \wedge \text{Büchi}(b)$ (in any arena)

Example



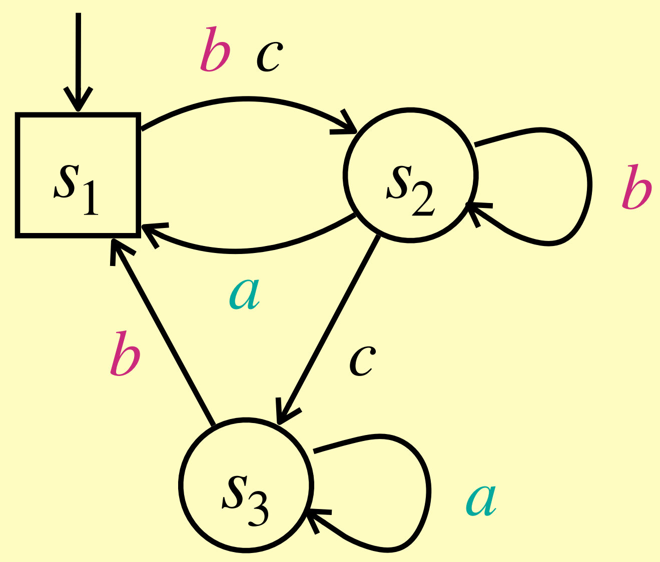
Example of chromatic memory

$\mathcal{M}$



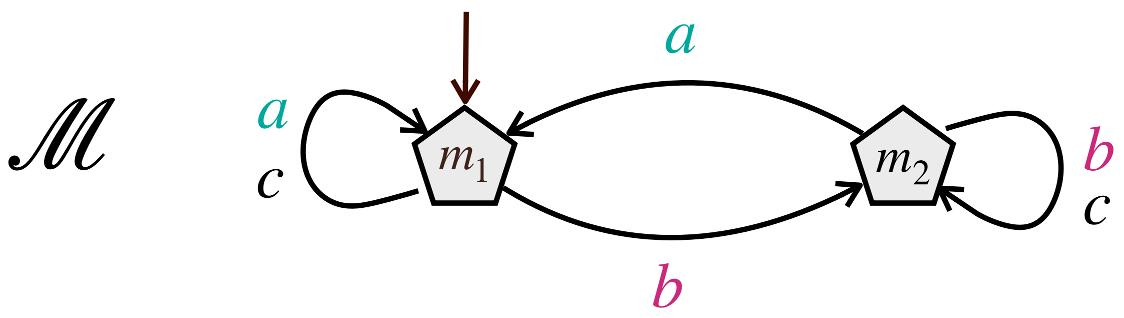
This skeleton is sufficient for winning
$W =$ Büchi(a) ∧ Büchi(b) (in any arena)

Example



$$\alpha_{\text{next}} : \quad M \times S_1 \to E$$
$$(m_1, s_2) \mapsto (s_2, b, s_2)$$
$$(m_2, s_2) \mapsto (s_2, a, s_1)$$
$$(m_\star, s_3) \mapsto (s_3, b, s_1)$$

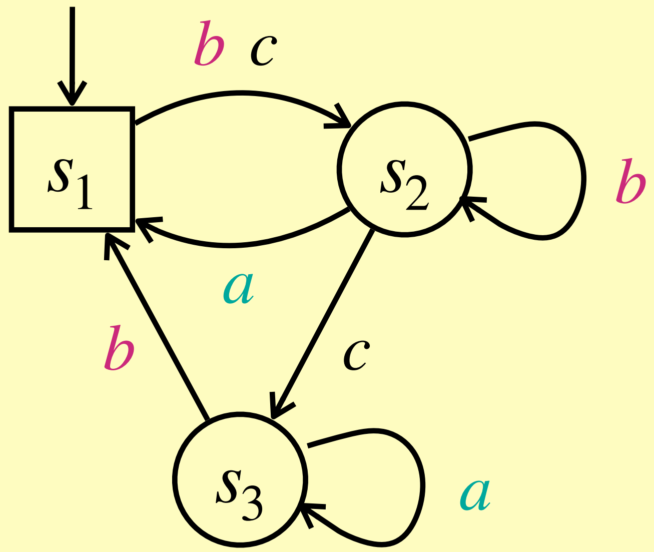
Example of chromatic memory



This skeleton is sufficient for winning

$W =$ Büchi$(a) \wedge$ Büchi(b) (in any arena)

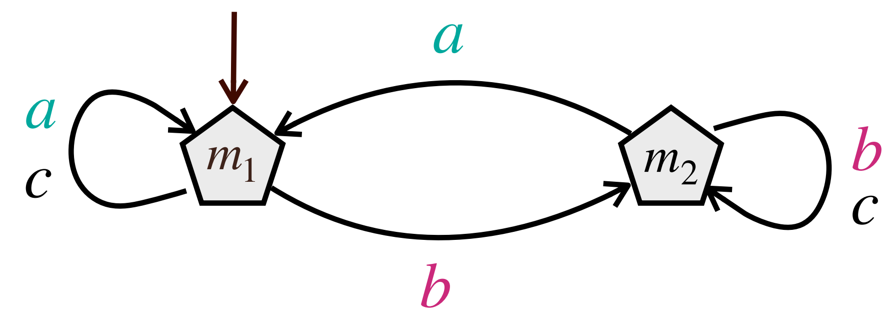
Example



$$\alpha_{\text{next}} : M \times S_1 \to E$$
$$(m_1, s_2) \mapsto (s_2, c, s_3)$$
$$(m_2, s_2) \mapsto (s_2, a, s_1)$$
$$(m_\star, s_3) \mapsto (s_3, b, s_1)$$

Example of chromatic memory

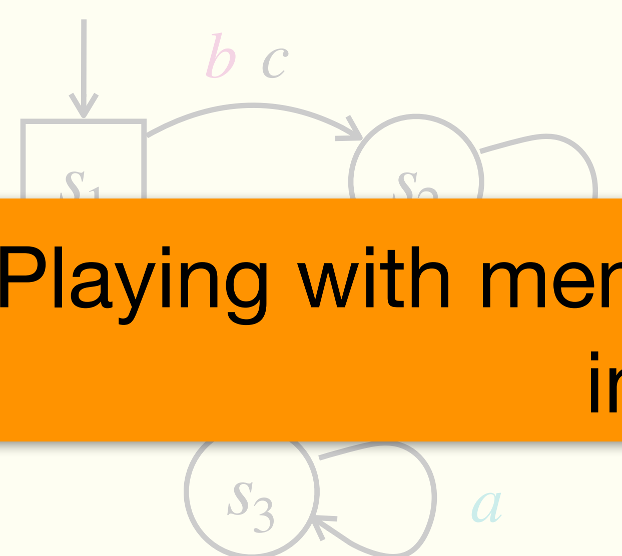
$\mathscr{M}$



This skeleton is sufficient for winning

$W =$ Büchi(a) $\wedge$ Büchi(b) (in any arena)

Example



$\alpha_{\text{next}} : M \times S_1 \to E$

$(m_\star, s_3) \mapsto (s_3, b, s_1)$

Playing with memory $\mathscr{M}$ is like playing memoryless in the product arena

A zoology of notions

A zoology of notions

- Let W be an objective and $i \in \{1,2\}$

A zoology of notions

- Let W be an objective and $i \in \{1,2\}$
- A skeleton $\mathscr{M}$ suffices to win for P_1 (resp. P_2) for W if P_1 (resp. P_2) has an optimal* strategy based on $\mathscr{M}$ in any game $(\mathscr{A}, W)$ (resp. $(\mathscr{A}, W^c)$)

* That is, it is winning whenever it is possible to win

A zoology of notions

in finite arenas
in one-player arenas

- Let W be an objective and $i \in \{1,2\}$
- A skeleton $\mathcal{M}$ suffices to win for P_1 (resp. P_2) for W if P_1 (resp. P_2) has an optimal* strategy based on $\mathcal{M}$ in any game $(\mathcal{A}, W)$ (resp. $(\mathcal{A}, W^c)$)

finite
one-player

* That is, it is winning whenever it is possible to win

A zoology of notions

- Let W be an objective and $i \in \{1,2\}$

in finite arenas
in one-player arenas

- A skeleton $\mathscr{M}$ suffices to win for P_1 (resp. P_2) for W if P_1 (resp. P_2) has an optimal* strategy based on $\mathscr{M}$ in any game $(\mathscr{A}, W)$ (resp. $(\mathscr{A}, W^c)$)

finite
one-player

- W is $\mathscr{M}$-determined if $\mathscr{M}$ suffices to win for both players for W

* That is, it is winning whenever it is possible to win

A zoology of notions

- Let W be an objective and $i \in \{1,2\}$

in finite arenas
in one-player arenas

- A skeleton $\mathcal{M}$ suffices to win for P_1 (resp. P_2) for W if P_1 (resp. P_2) has an optimal* strategy based on $\mathcal{M}$ in any game $(\mathcal{A}, W)$ (resp. $(\mathcal{A}, W^c)$)

finite
one-player

- W is $\mathcal{M}$-determined if $\mathcal{M}$ suffices to win for both players for W
- Memoryless determined = $\mathcal{M}_{\text{triv}}$-determined

* That is, it is winning whenever it is possible to win

A zoology of notions

in finite arenas
in one-player arenas

- Let W be an objective and $i \in \{1,2\}$
- A skeleton $\mathcal{M}$ suffices to win for P_1 (resp. P_2) for W if P_1 (resp. P_2) has an optimal* strategy based on $\mathcal{M}$ in any game $(\mathcal{A}, W)$ (resp. $(\mathcal{A}, W^c)$)

finite
one-player

- W is $\mathcal{M}$-determined if $\mathcal{M}$ suffices to win for both players for W
- Memoryless determined = $\mathcal{M}_{\text{triv}}$-determined
- Finite-memory determined = $\exists \mathcal{M}$ s.t. $\mathcal{M}$-determined

* That is, it is winning whenever it is possible to win

A zoology of notions

in finite arenas
in one-player arenas

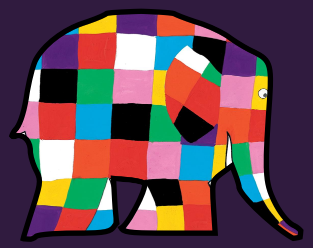
- Let W be an objective and $i \in \{1,2\}$
- A skeleton $\mathcal{M}$ suffices to win for P_1 (resp. P_2) for W if P_1 (resp. P_2) has an optimal* strategy based on $\mathcal{M}$ in any game $(\mathcal{A}, W)$ (resp. $(\mathcal{A}, W^c)$)

finite
one-player

- W is $\mathcal{M}$-determined if $\mathcal{M}$ suffices to win for both players for W
- Memoryless determined = $\mathcal{M}_{\text{triv}}$-determined
- Finite-memory determined = $\exists \mathcal{M}$ s.t. $\mathcal{M}$-determined
- W is half-positional = $\mathcal{M}_{\text{triv}}$ suffices to play optimally for P_1 for W

* That is, it is winning whenever it is possible to win

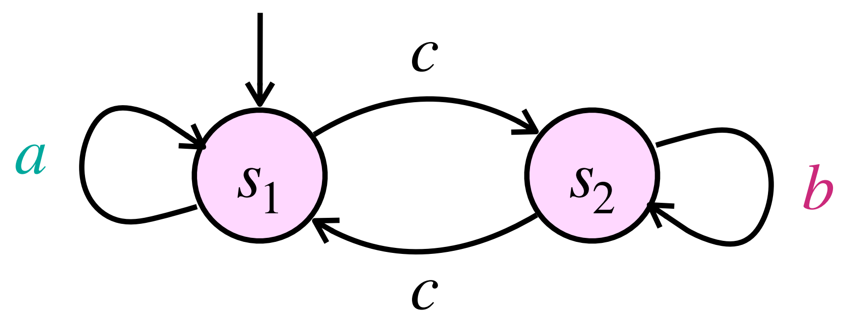
Warning



$\mathscr{M}$-determinacy requires

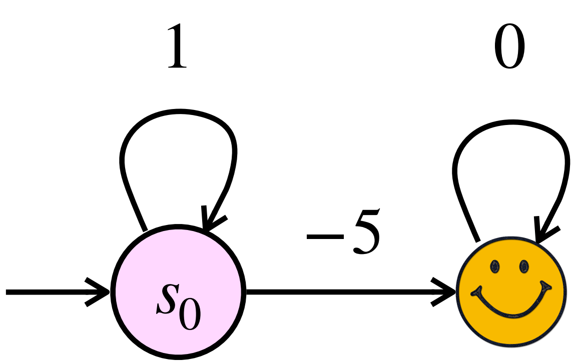
- Chromatic memory: the skeleton is based on colors
- Arena-independent memory: the same memory skeleton is used in all arenas (of the designed class)

Examples



« See infinitely often both a and b »

$\text{Büchi}(a) \wedge \text{Büchi}(b)$



« Reach the target with energy level 0 »

$\mathbf{FG}\ (\text{EL} = 0)$

Winning strategy

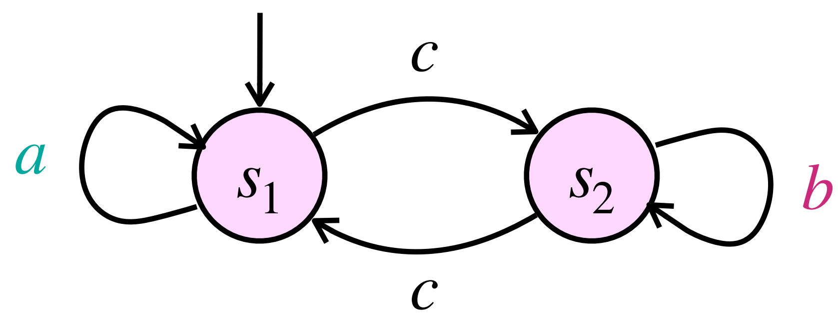
- At each visit to s_1, loop once in s_1 and then go to s_2
- At each visit to s_2, loop once in s_2 and then go to s_1
- Generates the sequence $(acbc)^\omega$

Winning strategy

- Loop five times in s_0
- Then go to the target
- Generates the sequence of colors $1\ 1\ 1\ 1\ 1\ \ -5\ 0\ 0\ 0\ldots$

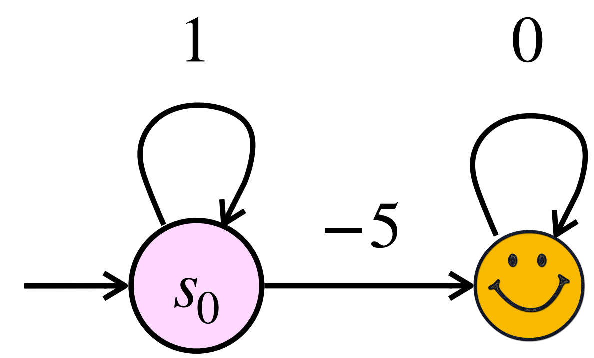
These two strategies require only **finite** memory

Examples



« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)



« Reach the target with energy level 0 »

$\mathbf{FG}\ (\text{EL} = 0)$

Winning strategy

- At each visit to s_1, loop once in s_1 and
- Generates the sequence $(acbc)^\omega$

There is an arena-independent memory based on a skeleton

Winning strategy

- Loop five times in s_0

The memory has to be arena-dependent

These two strategies require only **finite** memory

Our goal

Understand well low-memory specifications

Our goal

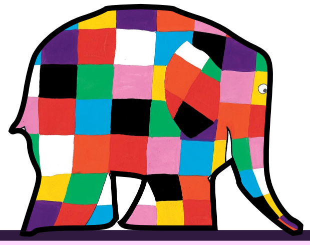
Understand well low-memory specifications

Memoryless / finite-memory determinacy

Is it the case that memoryless (resp. finite-memory) strategies suffice to win when winning strategies exist?

Our goal

Understand well low-memory specifications

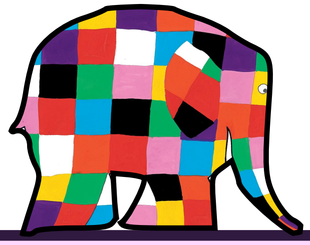


Memoryless / finite-memory determinacy

Is it the case that memoryless (resp. finite-memory) strategies suffice to win when winning strategies exist?

Our goal

Understand well low-memory specifications



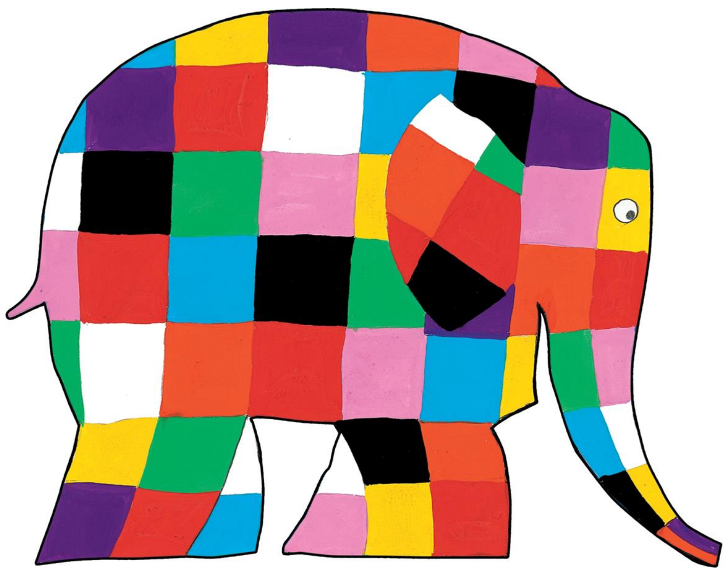
Memoryless / finite-memory determinacy

Is it the case that memoryless (resp. finite-memory) strategies suffice to win when winning strategies exist?

- Finite vs infinite games

Laboratoire Méthodes Formelles

Characterizing positional and chromatic finite-memory determinacy in finite games



A fundamental reference: [GZ05]

Sufficient conditions

- Sufficient conditions to guarantee memoryless optimal strategies for both players [GZ04, AR17]
- Sufficient conditions to guarantee half-positional optimal strategies [Kop06, Gim07, GK14]

[GZ05] Gimbert, Zielonka. Games Where You Can Play Optimally Without Any Memory (CONCUR'05).

A fundamental reference: [GZ05]

Sufficient conditions

- Sufficient conditions to guarantee memoryless optimal strategies for both players [GZ04, AR17]
- Sufficient conditions to guarantee half-positional optimal strategies [Kop06, Gim07, GK14]

- Characterization of winning objectives ensuring **memoryless determinacy** in finite games
- Fundamental reference: [GZ05]

[GZ05] Gimbert, Zielonka. Games Where You Can Play Optimally Without Any Memory (CONCUR'05).

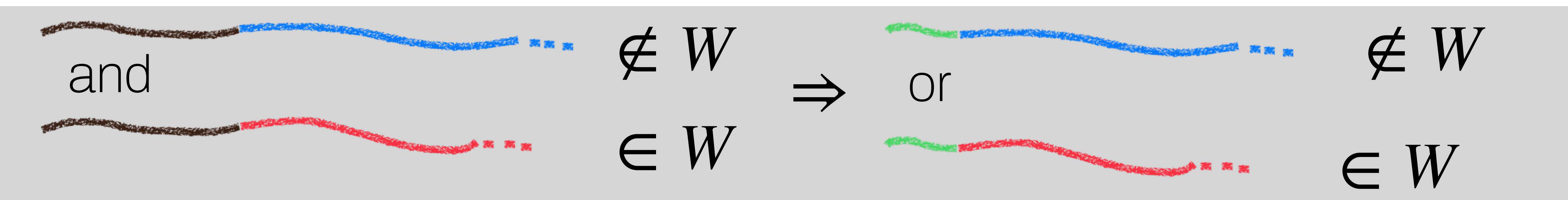
Monotony and selectivity

Monotony and selectivity

- Let $W \subseteq C^{\omega}$ be an objective

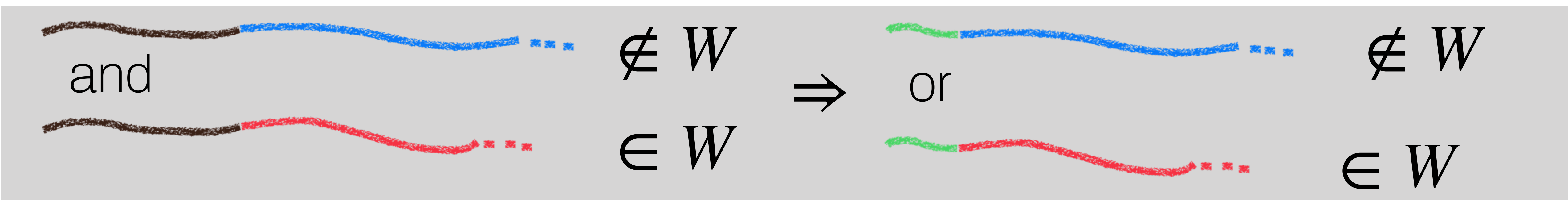
Monotony and selectivity

- Let $W \subseteq C^{\omega}$ be an objective
- W is **monotone** whenever:

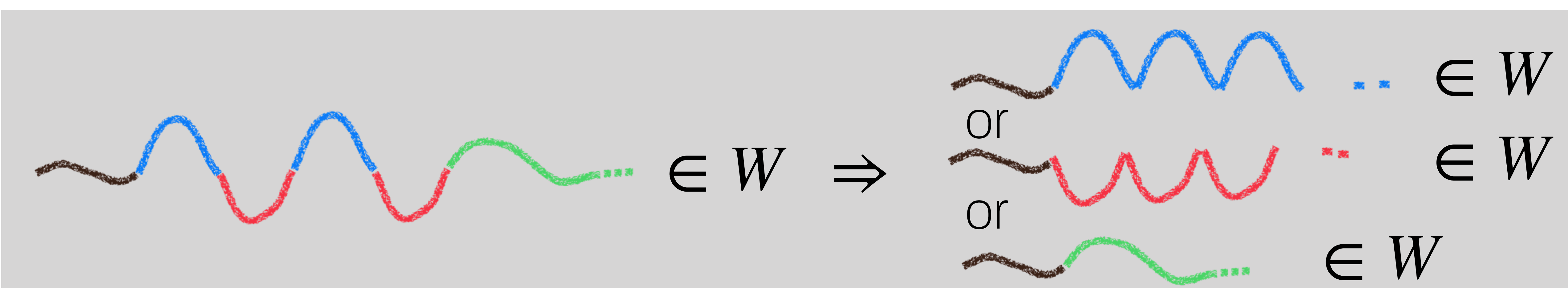


Monotony and selectivity

- Let $W \subseteq C^\omega$ be an objective
- W is **monotone** whenever:



- W is **selective** whenever:



Two characterizations

Let W be an objective

Characterization - Two-player games

The two following assertions are equivalent:

1. W is memoryless-determined in finite arenas;
2. Both W and W^c are monotone and selective.

Two characterizations

Let W be an objective

Characterization - Two-player games

The two following assertions are equivalent:

1. W is memoryless-determined in finite arenas;
2. Both W and W^c are monotone and selective.

Characterization - One-player games

The two following assertions are equivalent:

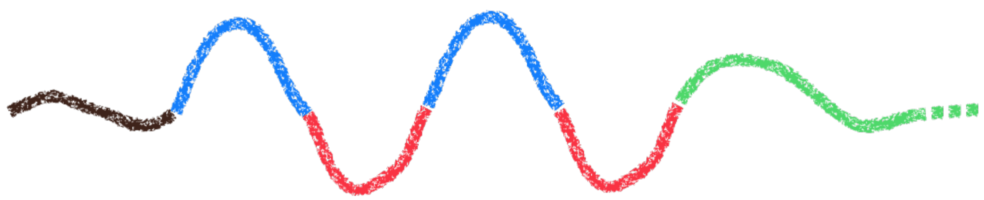
1. W is memoryless-determined in finite P_1-arenas;
2. W is monotone and selective.

Why? Proof hint (1)

Assume all P_1-games have optimal memoryless strategies.

Why? Proof hint (1)

Assume all P_1-games have optimal memoryless strategies.

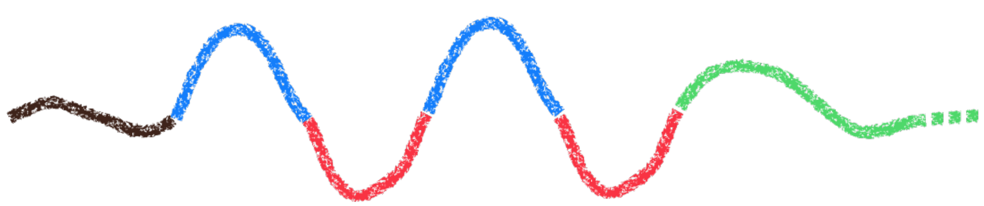
If 

is winning

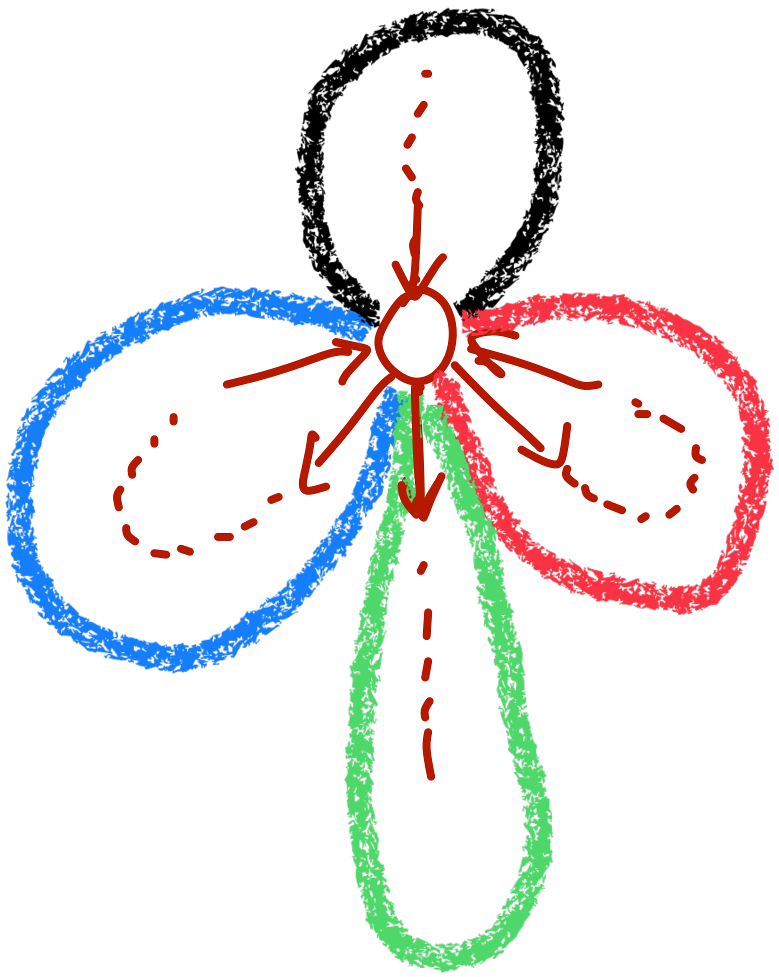
Why? Proof hint (1)

Assume all P_1-games have optimal memoryless strategies.

If

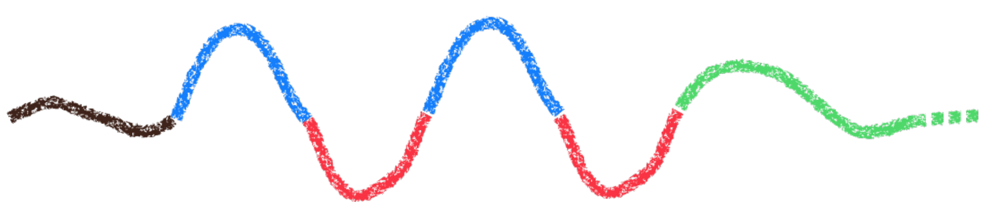


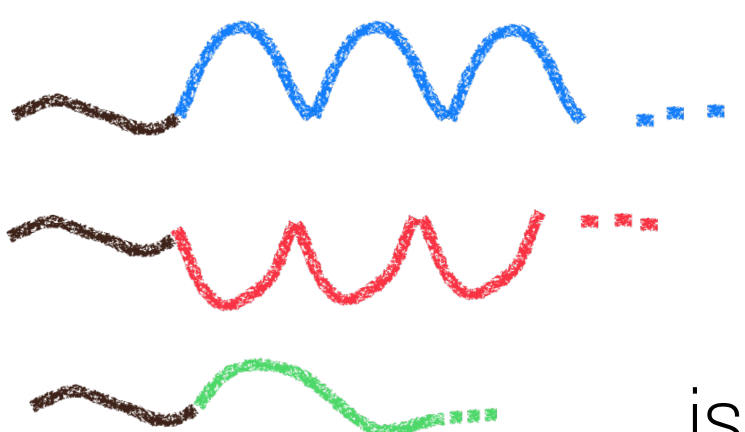
is winning

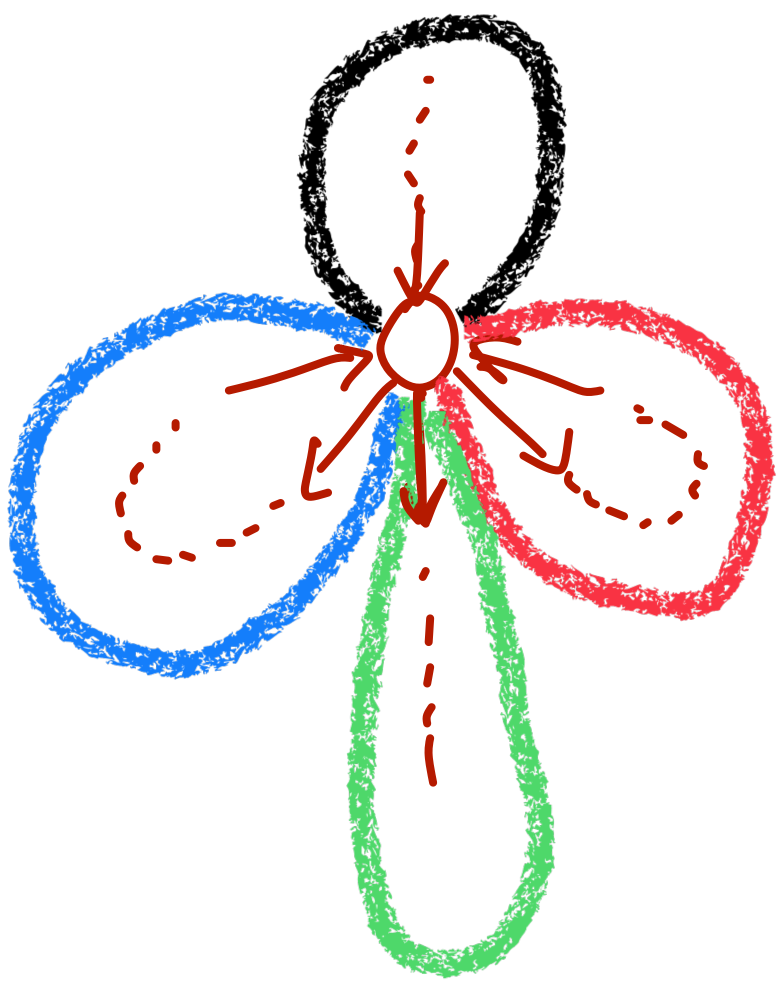


Why? Proof hint (1)

Assume all P_1-games have optimal memoryless strategies.

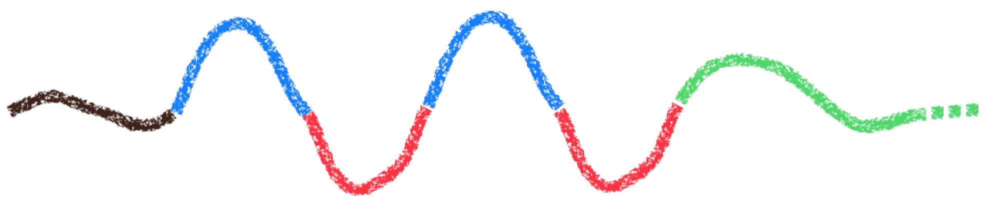
If  is winning

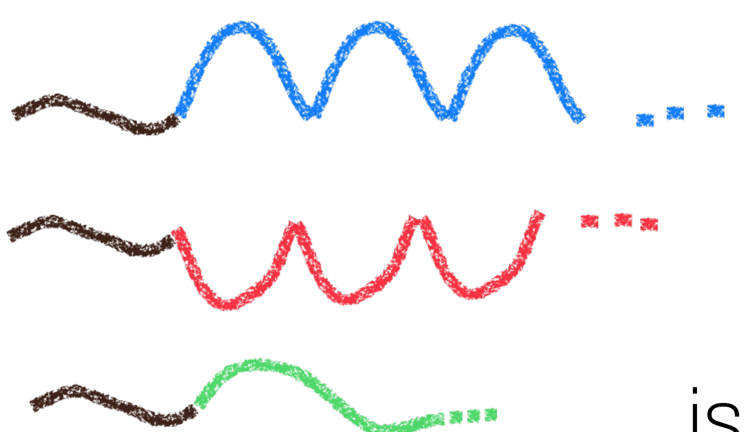
Then one of  is winning

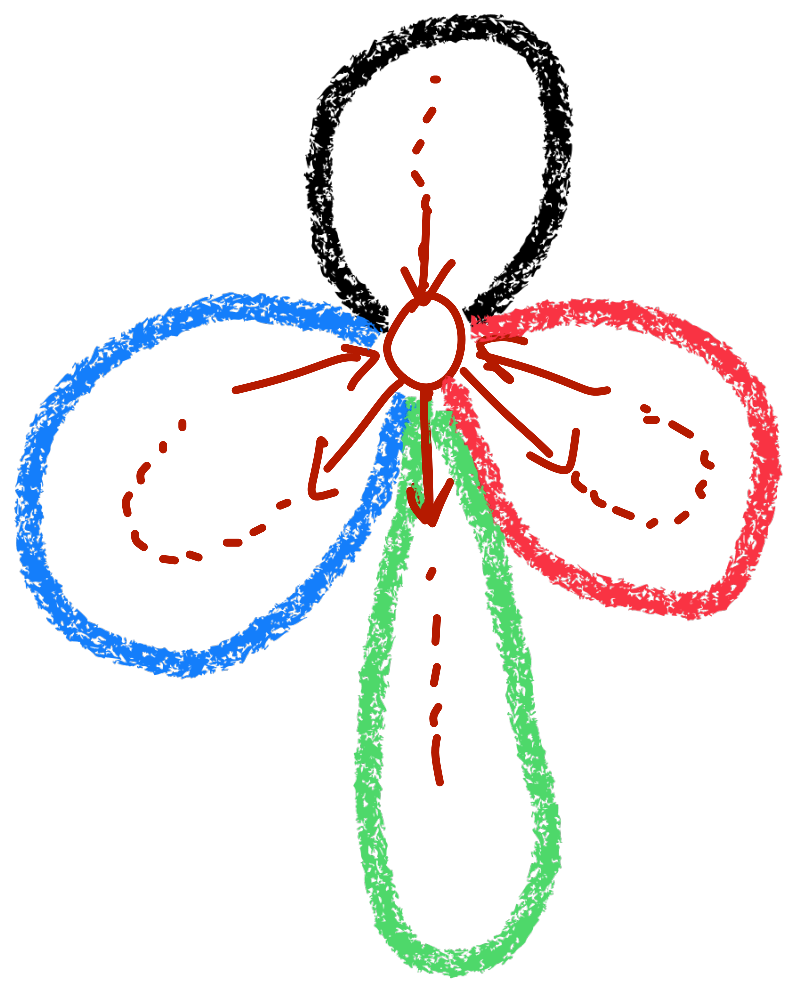


Why? Proof hint (1)

Assume all P_1-games have optimal memoryless strategies.

If  is winning

Then one of  is winning



W is selective

Why? Proof hint (2)

Assume W is monotone and selective.

Why? Proof hint (2)

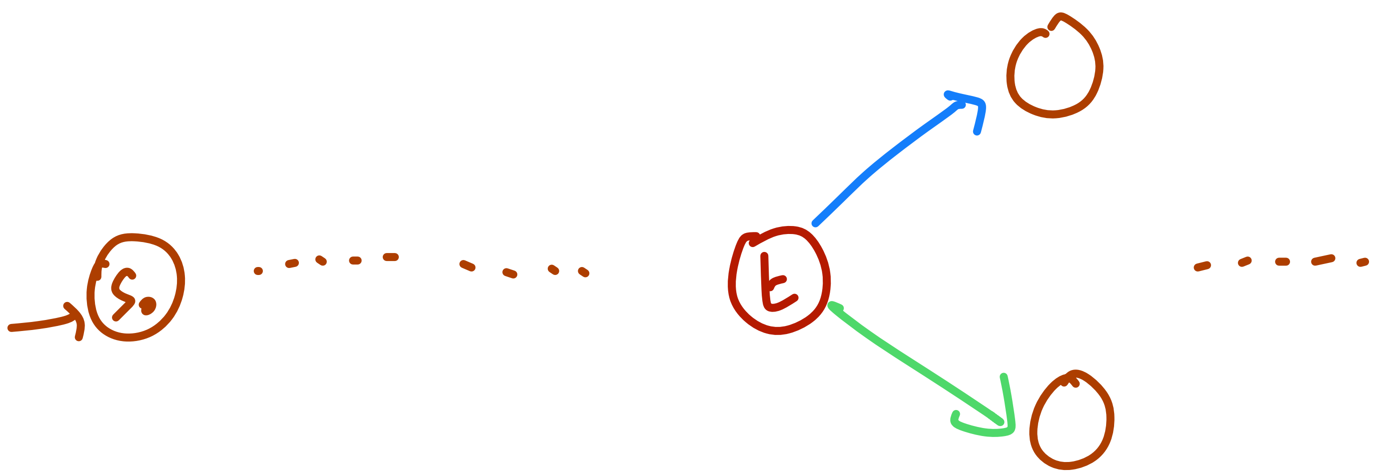
Assume W is monotone and selective.

The case of one-player arenas

Why? Proof hint (2)

Assume W is monotone and selective.

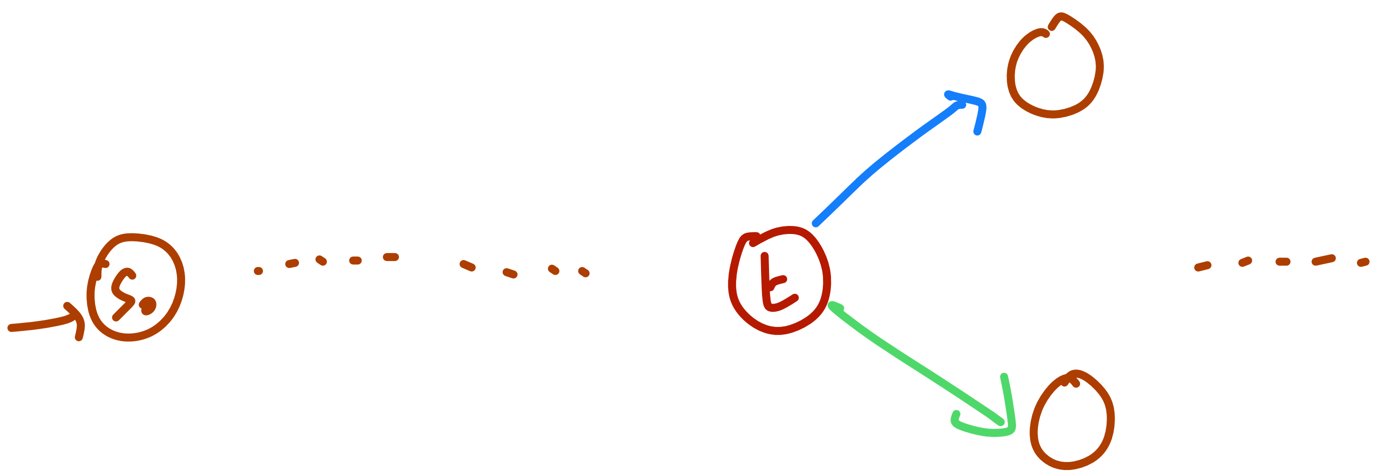
The case of one-player arenas



Why? Proof hint (2)

Assume W is monotone and selective.

The case of one-player arenas

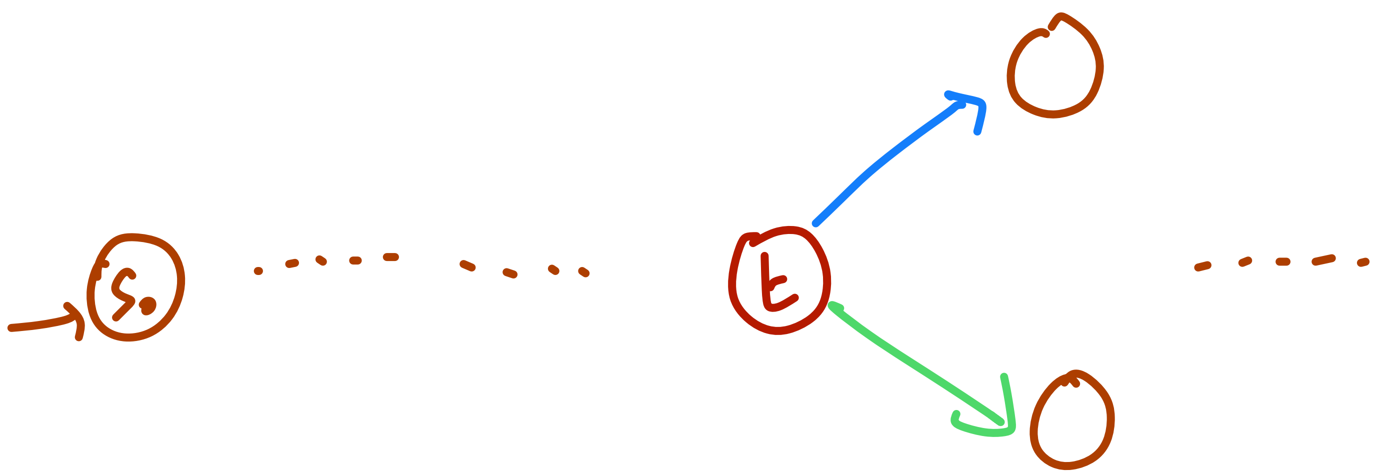


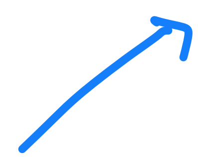
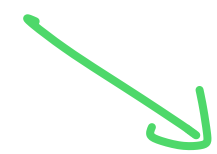
one best choice between ↗ and ↘ (monotony)

Why? Proof hint (2)

Assume W is monotone and selective.

The case of one-player arenas

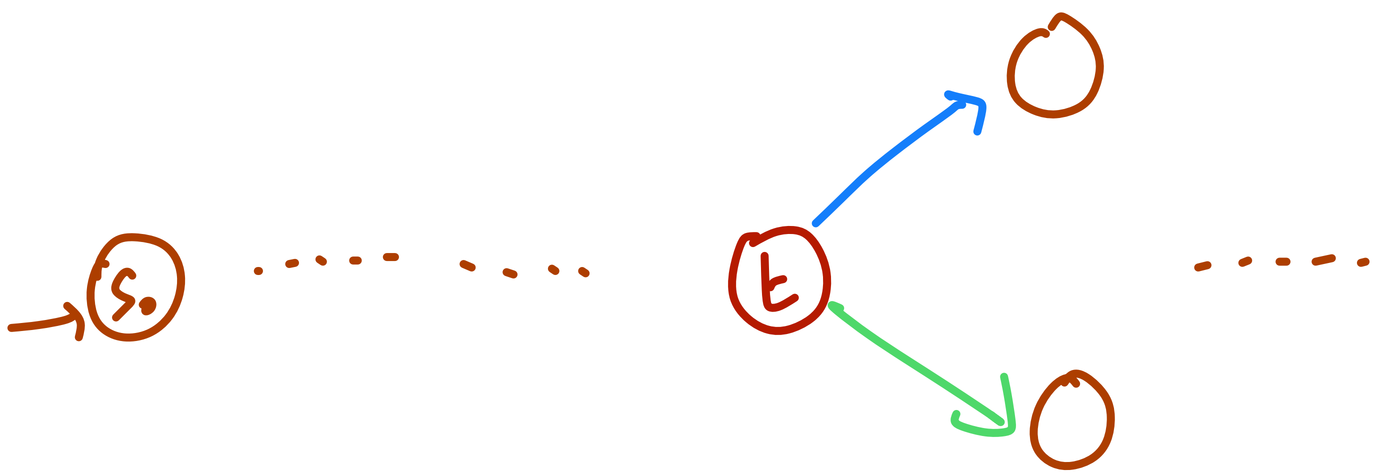


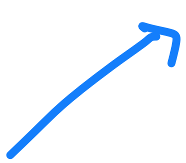
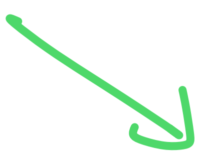
one best choice between  and  (monotony)
+ no reason to swap at t (selectivity)

Why? Proof hint (2)

Assume W is monotone and selective.

The case of one-player arenas



one best choice between  and  (monotony)
+ no reason to swap at t (selectivity)

No memory required at t!

Two characterizations

Let W be an objective

Characterization - Two-player games

The two following assertions are equivalent:

1. W is memoryless-determined in finite arenas;
2. Both W and W^c are monotone and selective.

Two characterizations

Let W be an objective

Characterization - Two-player games

The two following assertions are equivalent:

1. W is memoryless-determined in finite arenas;
2. Both W and W^c are monotone and selective.

Characterization - One-player games

The two following assertions are equivalent:

1. W is memoryless-determined in finite P_1-arenas;
2. W is monotone and selective.

Applications

Lifting theorem

Memoryless strategies suffice for W for P_i ($i = 1,2$) in finite P_i-arenas

$$\Downarrow$$

W is memoryless-determined in finite arenas

Applications

Lifting theorem

Memoryless strategies suffice for W for P_i $(i = 1{,}2)$ in finite P_i-arenas

$$\Downarrow$$

W is memoryless-determined in finite arenas

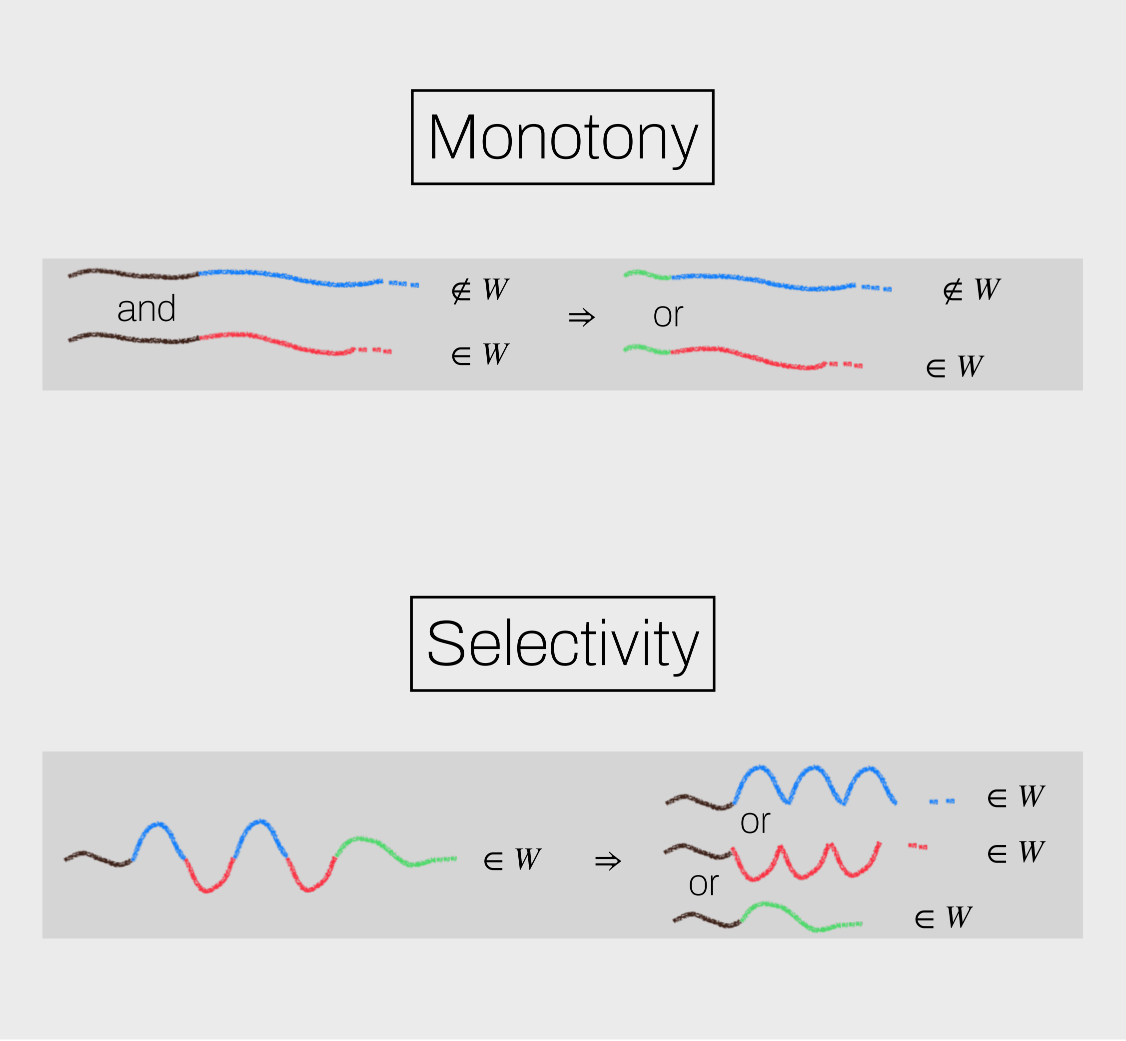
Very powerful and extremely useful in practice

- Easy to analyse the one-player case (graph reasoning)
  - Mean-payoff, average-energy [BMRLL15]
- Lift to two-player games via the theorem

[BMRLL15] Bouyer, Markey, Randour, Larsen, Laursen. Average-energy games (GandALF'15).

Discussion of examples

- Reachability, safety:
  - Monotone (though not prefix-independent)
  - Selective
- Parity, mean-payoff:
  - Prefix-independent hence monotone
  - Selective
- Average-energy games [BMRLL15]
  - Lifting theorem!!



[BMRLL15] Bouyer, Markey, Randour, Larsen, Laursen. Average-energy games (GandALF'15).

Can we lift [GZ05] to finite memory?

Can we lift [GZ05] to finite memory?

- No, in general

Can we lift [GZ05] to finite memory?

- No, in general

- Consider the objective W defined by

$$\liminf_n \sum_{i=1}^{n} c_i = +\infty \text{ or } \exists^\infty n \text{ s.t. } \sum_{i=1}^{n} c_i = 0$$

Can we lift [GZ05] to finite memory?

- No, in general

- Consider the objective W defined by

$$\liminf_{n} \sum_{i=1}^{n} c_i = +\infty \text{ or } \exists^\infty n \text{ s.t. } \sum_{i=1}^{n} c_i = 0$$

  - Optimal finite-memory strategies in one-player games

Can we lift [GZ05] to finite memory?

- No, in general

- Consider the objective W defined by

$$\liminf_n \sum_{i=1}^{n} c_i = +\infty \text{ or } \exists^\infty n \text{ s.t. } \sum_{i=1}^{n} c_i = 0$$

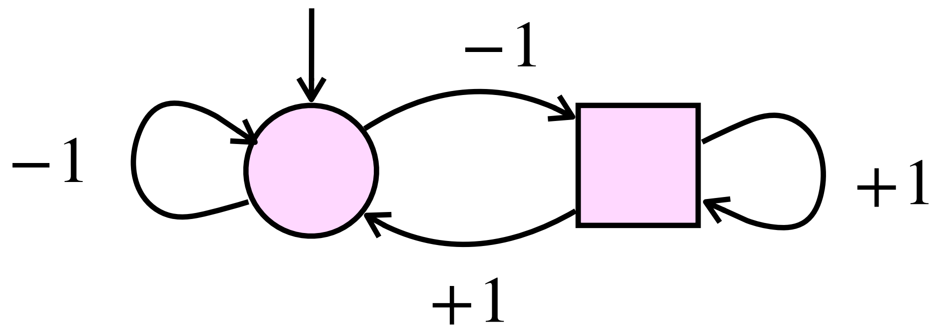
  - Optimal finite-memory strategies in one-player games
  - But not in two-player games!!

Can we lift [GZ05] to finite memory?

- No, in general
- Consider the objective W defined by

$$\liminf_n \sum_{i=1}^{n} c_i = +\infty \text{ or } \exists^\infty n \text{ s.t. } \sum_{i=1}^{n} c_i = 0$$

  - Optimal finite-memory strategies in one-player games
  - But not in two-player games!!

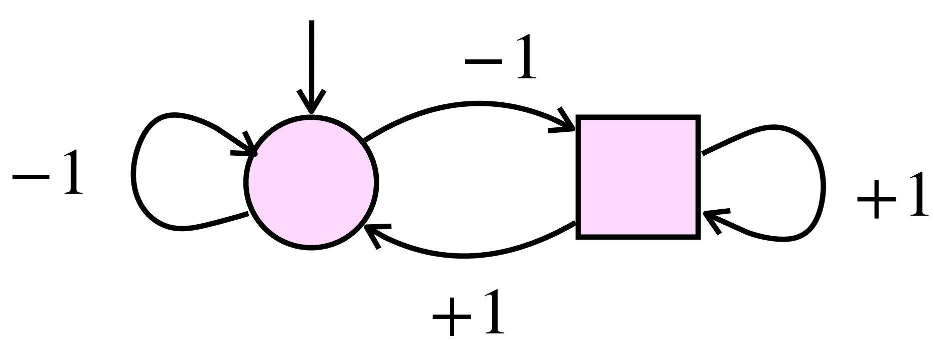


Can we lift [GZ05] to finite memory?

- No, in general
- Consider the objective W defined by

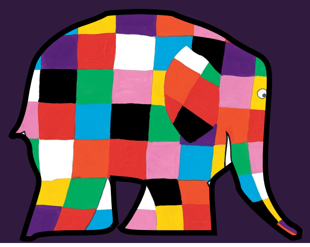
$$\liminf_n \sum_{i=1}^{n} c_i = +\infty \text{ or } \exists^\infty n \text{ s.t. } \sum_{i=1}^{n} c_i = 0$$

  - Optimal finite-memory strategies in one-player games
  - But not in two-player games!!



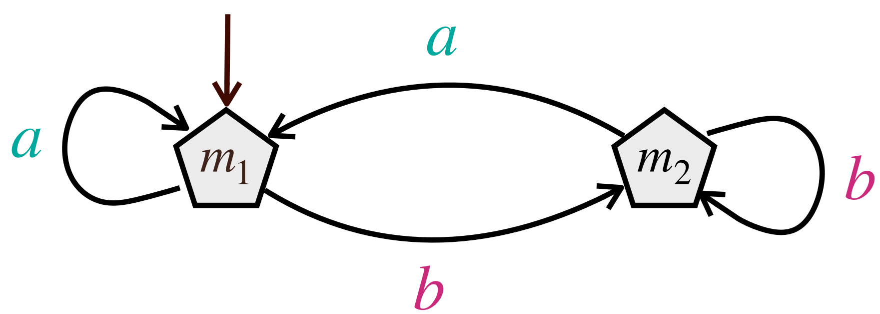
P_1 wins but requires infinite memory

Chromatic memory



Memory skeleton

$\mathcal{M} = (M, m_{\text{init}}, \alpha_{\text{upd}})$ with $m_{\text{init}} \in M$ and $\alpha_{\text{upd}} : M \times C \to M$



Not yet a strategy!

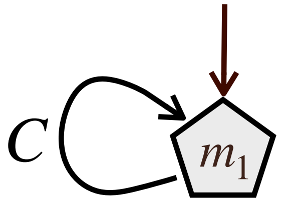
$\sigma_i : S^*S_i \to E$

Strategy with memory $\mathcal{M}$

Additional next-move function $\alpha_{\text{next}} : M \times S_i \to E$

$(\mathcal{M}, \alpha_{\text{next}})$ defines a strategy!

Remark: memoryless strategies are $\mathcal{M}_{\text{triv}}$-strategies, where $\mathcal{M}_{\text{triv}}$ is



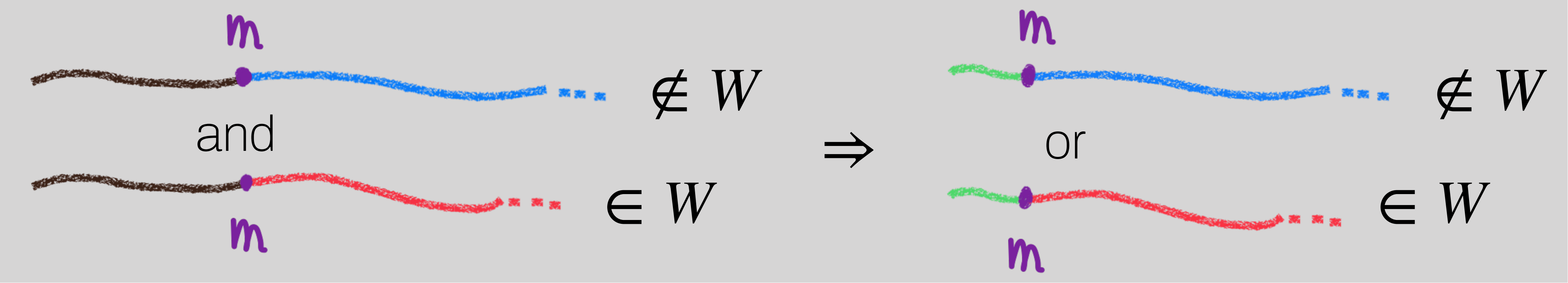
Adding memory

Adding memory

- Let W be a winning objective and $\mathscr{M}$ be a memory skeleton

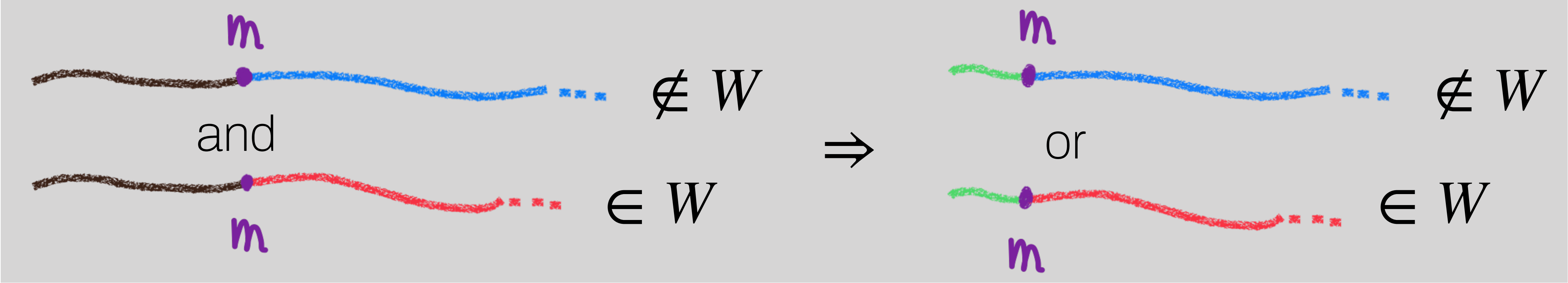
Adding memory

- Let W be a winning objective and $\mathscr{M}$ be a memory skeleton
- W is $\mathscr{M}$**-monotone** whenever:

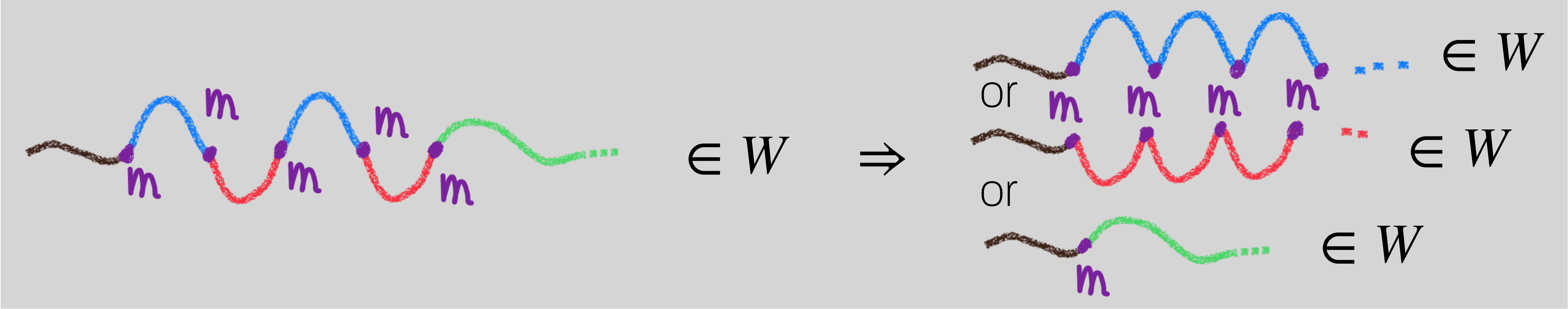


Adding memory

- Let W be a winning objective and $\mathscr{M}$ be a memory skeleton
- W is $\mathscr{M}$**-monotone** whenever:



- W is $\mathscr{M}$**-selective** whenever:



Two characterizations

Let W be a winning objective and $\mathscr{M}$ be a memory skeleton

Characterization - Two-player games

The two following assertions are equivalent:

1. W is $\mathscr{M}$-determined in finite arenas;
2. Both W and W^c are $\mathscr{M}$-monotone and $\mathscr{M}$-selective.

Two characterizations

Let W be a winning objective and $\mathscr{M}$ be a memory skeleton

Characterization - Two-player games

The two following assertions are equivalent:

1. W is $\mathscr{M}$-determined in finite arenas;
2. Both W and W^c are $\mathscr{M}$-monotone and $\mathscr{M}$-selective.

Characterization - One-player games

The two following assertions are equivalent:

1. W is $\mathscr{M}$-determined in finite P_1-arenas;
2. W is $\mathscr{M}$-monotone and $\mathscr{M}$-selective.

Two characterizations

Let W be a winning objective and $\mathcal{M}$ be a memory skeleton

Characterization - Two-player games

The two following assertions are equivalent:

1. W is $\mathcal{M}$-determined in finite arenas;
2. Both W and W^c are $\mathcal{M}$-monotone and $\mathcal{M}$-selective.

Characterization - One-player games

The two following assertions are equivalent:

1. W is $\mathcal{M}$-determined in finite P_1-arenas;
2. W is $\mathcal{M}$-monotone and $\mathcal{M}$-selective.

→ We recover [GZ05] with $\mathcal{M} = \mathcal{M}_{\text{triv}}$

Technical tool: Memory-covered arenas

Technical tool: Memory-covered arenas

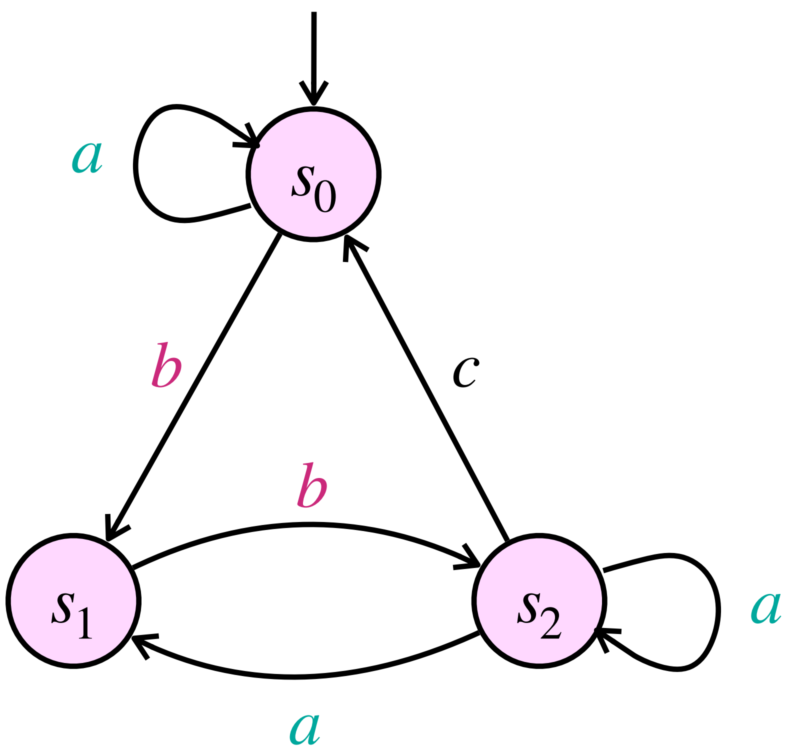
If the arena has enough information from $\mathscr{M}$,
then memoryless strategies will be sufficient

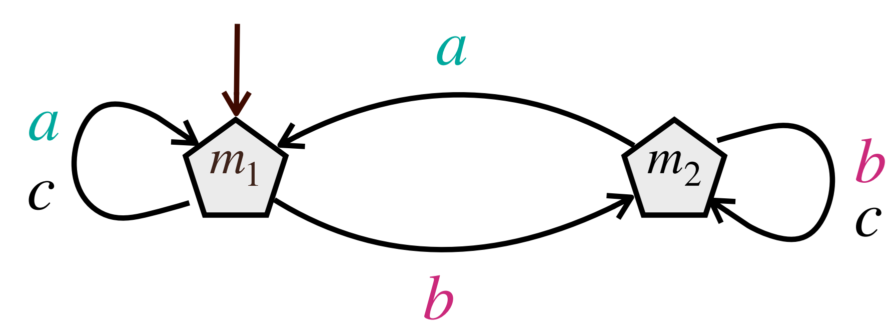
Covered arenas = same properties as product arenas

Technical tool: Memory-covered arenas

If the arena has enough information from $\mathcal{M}$, then memoryless strategies will be sufficient

Covered arenas = same properties as product arenas

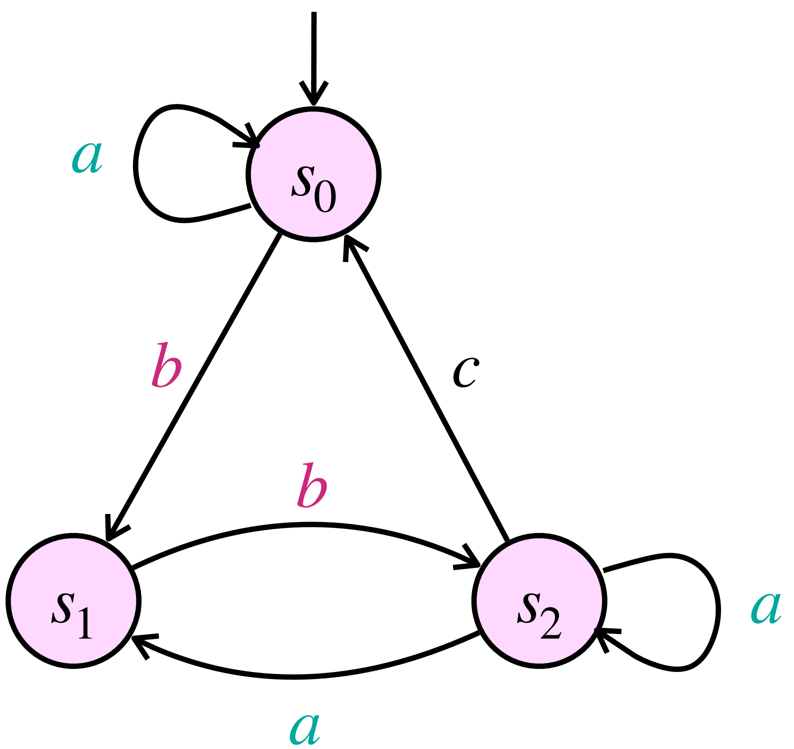


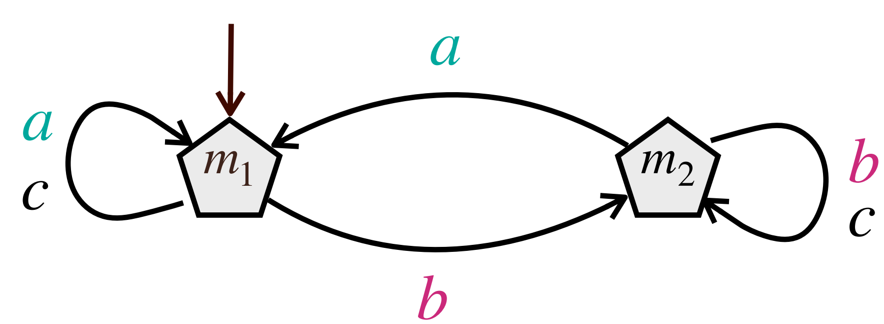


Technical tool: Memory-covered arenas

If the arena has enough information from $\mathcal{M}$, then memoryless strategies will be sufficient

Covered arenas = same properties as product arenas





Hence one can apply a [GZ05]-like reasoning to $\mathcal{M}$-covered arenas

Applications

Lifting theorem

Strategies based on $\mathscr{M}_i$ suffice for W for P_i in finite P_i-arenas

$$\Downarrow$$

W is $(\mathscr{M}_1 \otimes \mathscr{M}_2)$-determined in finite arenas

Applications

Lifting theorem

Strategies based on $\mathscr{M}_i$ suffice for W for P_i in finite P_i-arenas

$$\Downarrow$$

W is $(\mathscr{M}_1 \otimes \mathscr{M}_2)$-determined in finite arenas

Very powerful and extremely useful in practice

- Easy to analyse the one-player case (graph analysis)
  - Conjunction of ω-regular objectives
- Lift to two-player games via the theorem

Example of application

W = Reach(a) ∧ Reach(b)

Example of application

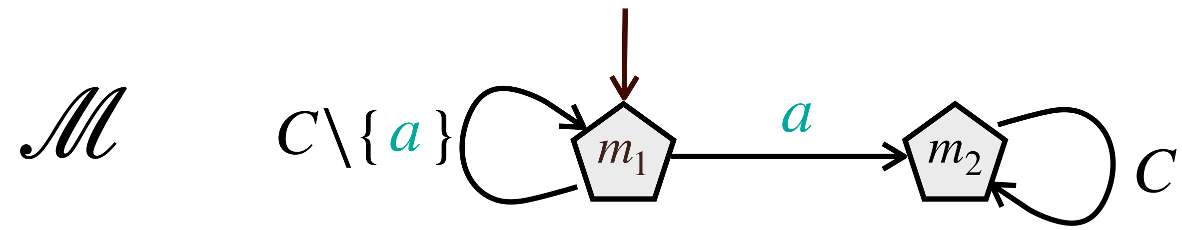
W = Reach(a) $\wedge$ Reach(b)

- W is not $\mathscr{M}_{\text{triv}}$-monotone

Example of application

W = Reach(a) ∧ Reach(b)

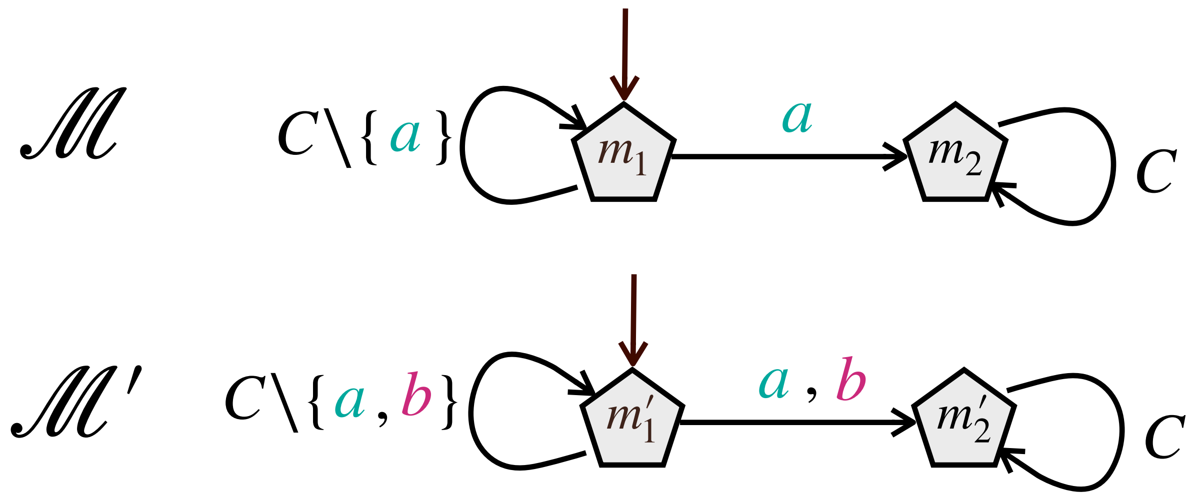
- W is not $\mathscr{M}_{\text{triv}}$-monotone
- W is $\mathscr{M}$-monotone
  but not $\mathscr{M}$-selective



Example of application

W = Reach(a) ∧ Reach(b)

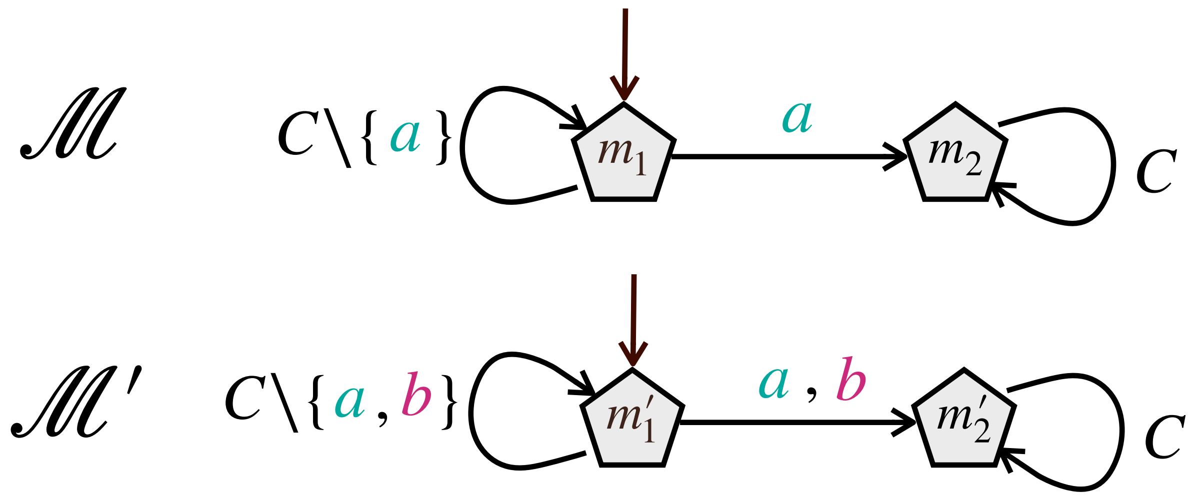
- W is not $\mathcal{M}_{\text{triv}}$-monotone
- W is $\mathcal{M}$-monotone but not $\mathcal{M}$-selective
- W is $\mathcal{M}'$-selective



Example of application

$W = \text{Reach}(a) \land \text{Reach}(b)$

- W is not $\mathcal{M}_{\text{triv}}$-monotone
- W is $\mathcal{M}$-monotone
  but not $\mathcal{M}$-selective
- W is $\mathcal{M}'$-selective



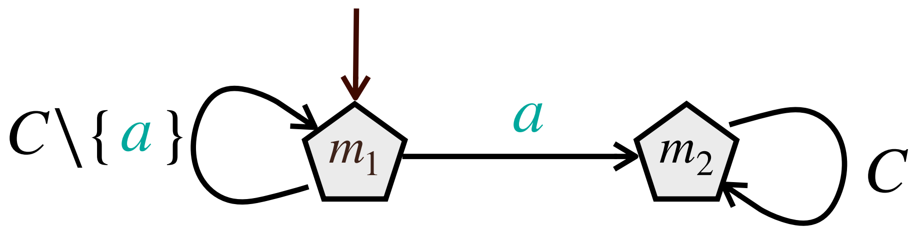
- W is $\mathcal{M}$-monotone and $\mathcal{M}'$-selective
- W^c is $\mathcal{M}$-monotone and $\mathcal{M}_{\text{triv}}$-selective

Example of application

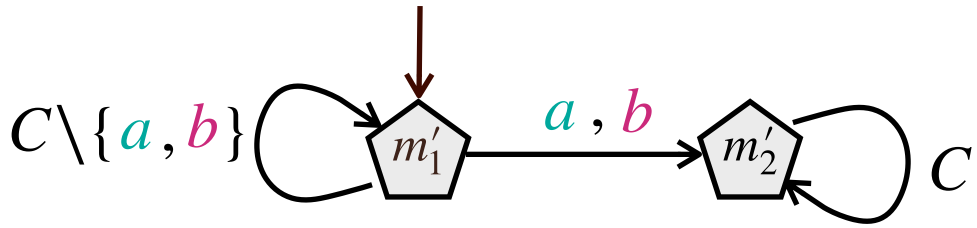
W = Reach(a) ∧ Reach(b)

- W is not $\mathscr{M}_{\text{triv}}$-monotone
- W is $\mathscr{M}$-monotone
  but not $\mathscr{M}$-selective
- W is $\mathscr{M}'$-selective

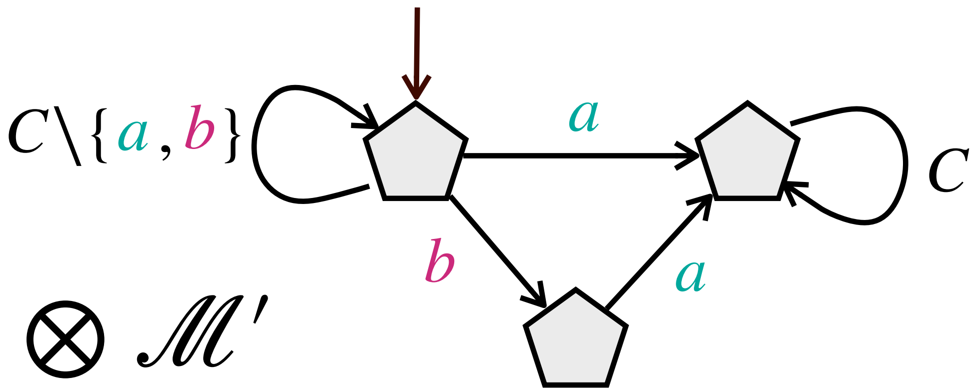
$\mathscr{M}$



$\mathscr{M}'$



- W is $\mathscr{M}$-monotone and $\mathscr{M}'$-selective
- W^c is $\mathscr{M}$-monotone and $\mathscr{M}_{\text{triv}}$-selective



$\mathscr{M} \otimes \mathscr{M}'$

→ Memory $\mathscr{M} \otimes \mathscr{M}'$ is sufficient for both players in all finite games

Partial conclusion

Finite games

Partial conclusion

Finite games

- Complete characterization of winning objectives (and even preference relations) that ensure (chromatic) finite-memory determinacy (for both players)

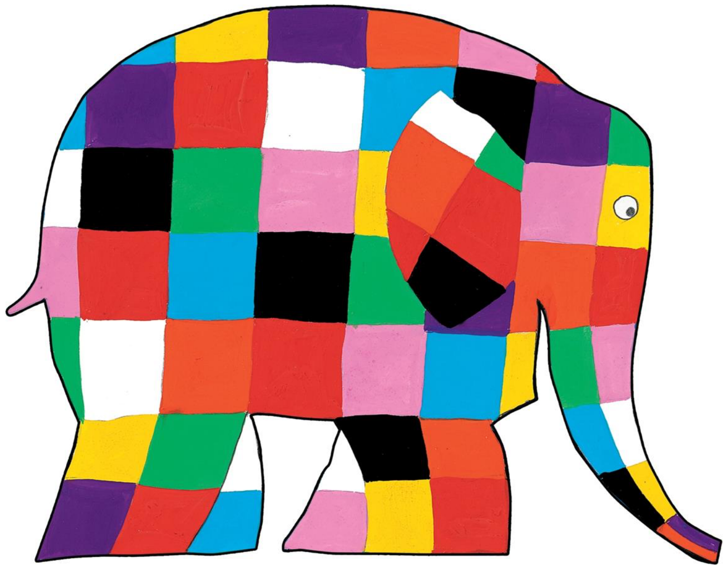
Partial conclusion

Finite games

- Complete characterization of winning objectives (and even preference relations) that ensure (chromatic) finite-memory determinacy (for both players)

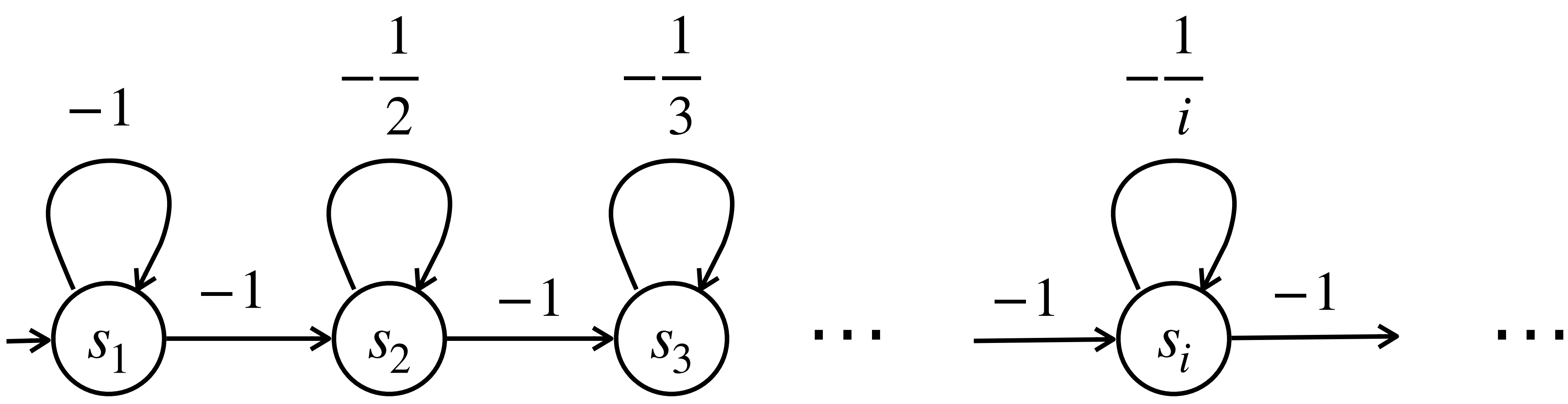
- One-to-two-player lifts
  (requires chromatic finite memory determinacy in one-player games for both players;
  ensures chromatic finite memory determinacy in two-players games for both players)

Characterizing positional and chromatic finite-memory determinacy in infinite games



The case of mean-payoff

- Objective for P_1: get non-negative (limsup) mean-payoff
- In finite games: **memoryless** strategies are sufficient to win
- In infinite games: **infinite memory** is required to win



A first insight [CN06]

- Let W be a prefix-independent objective.

[CN06] Colcombet and Niwiński. On the positional determinacy of edge-labeled games (ICALP'06).
[Zie98] Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees (TCS).

A first insight [CN06]

- Let W be a prefix-independent objective.

Characterization - Two-player games

The two following assertions are equivalent:

1. Positional optimal strategies are sufficient for W in all (infinite) games for both players;
2. W is a parity condition

   That is, there are $n \in \mathbb{N}$ and $\gamma : C \to \{0,1,\ldots,n\}$ such that
   $W = \{c_1c_2\ldots \in C^\omega \mid \limsup_i \gamma(c_i) \text{ is even}\}$

[CN06] Colcombet and Niwiński. On the positional determinacy of edge-labeled games (ICALP'06).
[Zie98] Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees (TCS).

A first insight [CN06]

Limitations

- Let W be a prefix-independent objective.

Characterization - Two-player games

The two following assertions are equivalent:

1. Positional optimal strategies are sufficient for W in all (infinite) games for both players;
2. W is a parity condition

   That is, there are $n \in \mathbb{N}$ and $\gamma : C \to \{0,1,\ldots,n\}$ such that $W = \{c_1c_2\ldots \in C^\omega \mid \limsup_i \gamma(c_i) \text{ is even}\}$

[CN06] Colcombet and Niwiński. On the positional determinacy of edge-labeled games (ICALP'06).
[Zie98] Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees (TCS).

A first insight [CN06]

Limitations

- Let W be a prefix-independent objective.

Characterization - Two-player games

The two following assertions are equivalent:

1. Positional optimal strategies are sufficient for W in all (infinite) games for both players;
2. W is a parity condition

   That is, there are $n \in \mathbb{N}$ and $\gamma : C \to \{0,1,\ldots,n\}$ such that
   $W = \{c_1c_2\ldots \in C^\omega \mid \limsup_i \gamma(c_i) \text{ is even}\}$

[CN06] Colcombet and Niwiński. On the positional determinacy of edge-labeled games (ICALP'06).
[Zie98] Zielonka. Infinite games on finitely coloured graphs with applications to automata on infinite trees (TCS).

Some language theory (1)

- Let $L \subseteq C^*$ be a language of finite words

Right congruence

- Given $x, y \in C^*$,

$$x \sim_L y \Leftrightarrow \forall z \in C^*, \left(x \cdot z \in L \Leftrightarrow y \cdot z \in L\right)$$

Some language theory (1)

- Let $L \subseteq C^*$ be a language of finite words

Right congruence

- Given $x, y \in C^*$,

$$x \sim_L y \Leftrightarrow \forall z \in C^*, \left(x \cdot z \in L \Leftrightarrow y \cdot z \in L\right)$$

Myhill-Nerode Theorem

- L is regular if and only if $\sim_L$ has finite index;
  - There is an automaton whose states are classes of $\sim_L$, which recognizes L.

Some language theory (2)

- Let $W \subseteq C^\omega$ be a language of infinite words

Right congruence

- Given $x, y \in C^*$,

$$x \sim_W y \Leftrightarrow \forall z \in C^\omega, \left(x \cdot z \in W \Leftrightarrow y \cdot z \in W\right)$$

Some language theory (2)

- Let $W \subseteq C^\omega$ be a language of infinite words

Right congruence

- Given $x, y \in C^*$,

$$x \sim_W y \Leftrightarrow \forall z \in C^\omega, \left(x \cdot z \in W \Leftrightarrow y \cdot z \in W\right)$$

Link with ω-regularity?

- If W is ω-regular, then $\sim_W$ has finite index;
  - The automaton $\mathcal{M}_W$ based on $\sim_W$ is a **prefix-classifier**;
- The converse does not hold (e.g. all prefix-independent languages are such that $\sim_W$ has only one element).

Characterization [BRV22]

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

[CN06] Colcombet, Niwiński. On the positional determinacy of edge-labeled games (Theor. Comp. Science).
[BRV22] Bouyer, Randour, Vandenhove. Characterizing Omega-Regularity through Finite-Memory Determinacy of Games on Infinite Graphs (STACS'22).
[EJ91] Emerson, Jutla. Tree automata, mu-calculus and determinacy (FoCS'91).
[Zie98] Zielonka. Infinite games on finitely colored graphs with applications to automata on infinite Trees (TCS)

Characterization [BRV22]

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

Characterization - Two-player games

W is finite-memory-determined if and only if W is ω-regular. Moreover, if $\mathcal{M}$ is an adapted memory skeleton for W, then W is recognized by a deterministic parity automaton built on top of $\mathcal{M} \otimes \mathcal{M}_W$.

[CN06] Colcombet, Niwiński. On the positional determinacy of edge-labeled games (Theor. Comp. Science).
[BRV22] Bouyer, Randour, Vandenhove. Characterizing Omega-Regularity through Finite-Memory Determinacy of Games on Infinite Graphs (STACS'22).
[EJ91] Emerson, Jutla. Tree automata, mu-calculus and determinacy (FoCS'91).
[Zie98] Zielonka. Infinite games on finitely colored graphs with applications to automata on infinite Trees (TCS)

Characterization [BRV22]

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

Characterization - Two-player games

W is finite-memory-determined if and only if W is ω-regular. Moreover, if $\mathcal{M}$ is an adapted memory skeleton for W, then W is recognized by a deterministic parity automaton built on top of $\mathcal{M} \otimes \mathcal{M}_W$.

→ Generalizes [CN06] where both $\mathcal{M}$ and $\mathcal{M}_W$ are trivial

[CN06] Colcombet, Niwiński. On the positional determinacy of edge-labeled games (Theor. Comp. Science).
[BRV22] Bouyer, Randour, Vandenhove. Characterizing Omega-Regularity through Finite-Memory Determinacy of Games on Infinite Graphs (STACS'22).
[EJ91] Emerson, Jutla. Tree automata, mu-calculus and determinacy (FoCS'91).
[Zie98] Zielonka. Infinite games on finitely colored graphs with applications to automata on infinite Trees (TCS)

Characterization [BRV22]

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

Characterization - Two-player games

W is finite-memory-determined if and only if W is ω-regular. Moreover, if $\mathcal{M}$ is an adapted memory skeleton for W, then W is recognized by a deterministic parity automaton built on top of $\mathcal{M} \otimes \mathcal{M}_W$.

$\rightarrow$ Generalizes [CN06] where both $\mathcal{M}$ and $\mathcal{M}_W$ are trivial

- The proof of $\Leftarrow$ is given by [EJ91,Zie98]

[CN06] Colcombet, Niwiński. On the positional determinacy of edge-labeled games (Theor. Comp. Science).
[BRV22] Bouyer, Randour, Vandenhove. Characterizing Omega-Regularity through Finite-Memory Determinacy of Games on Infinite Graphs (STACS'22).
[EJ91] Emerson, Jutla. Tree automata, mu-calculus and determinacy (FoCS'91).
[Zie98] Zielonka. Infinite games on finitely colored graphs with applications to automata on infinite Trees (TCS)

Proof idea for $\Rightarrow$

Assume W is $\mathscr{M}$-determined. Then:

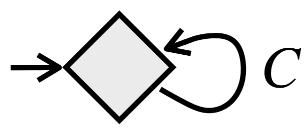
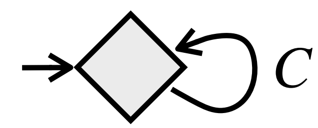
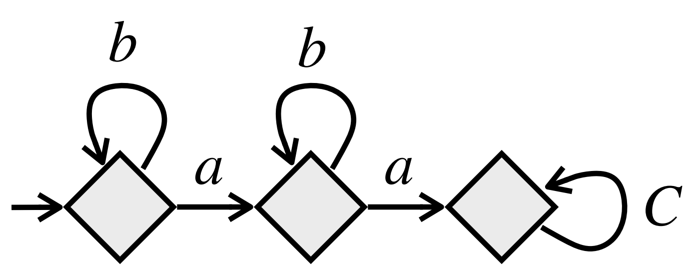
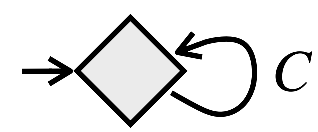
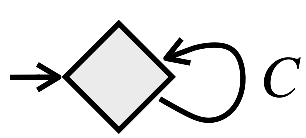
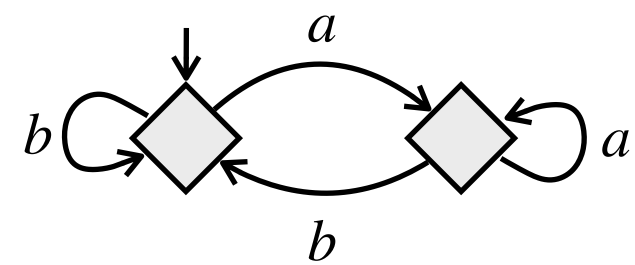
- $\mathscr{M}_W$ is finite (which implies that W is $\mathscr{M}_W$-prefix-independent);
- W is $\mathscr{M}$-cycle-consistent: after a finite word u, if $(w_i)_i$ are winning cycles of $\mathscr{M}$ (after u), then $uw_1w_2w_3\cdots$ is winning; Idem for losing cycles

$\rightarrow$ W is $(\mathscr{M} \otimes \mathscr{M}_W)$-prefix-independent and $(\mathscr{M} \otimes \mathscr{M}_W)$-cycle-consistent

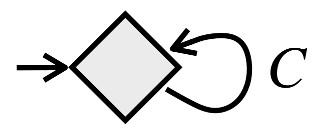
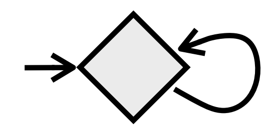
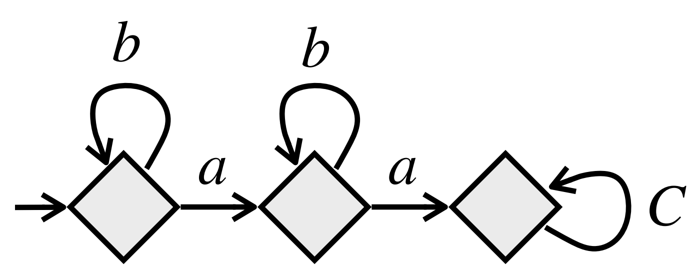
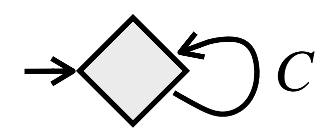
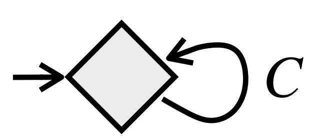
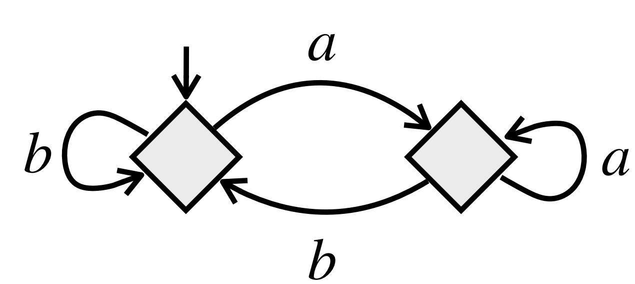
$\rightarrow$ Hence W can be recognized by a DPA built on top of $\mathscr{M} \otimes \mathscr{M}_W$
(relies on ordering cycles according to how good they are for winning)

Difficult part of the proof

Examples

Objective W	**Prefix classifier $\mathscr{M}_W$**	**Memory $\mathscr{M}$**
Parity objective		
$C = \{a, b\}$ $W = b^*ab^*aC^\omega$		
$C = \{a, b\}$ $W = C^*(ab)^\omega$		

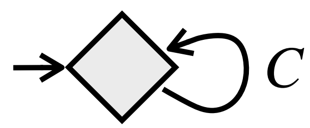
Examples

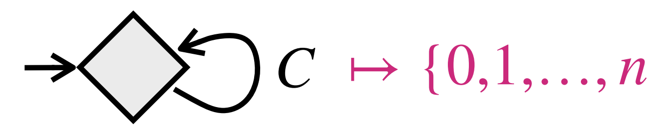
Objective W	**Prefix classifier** $\mathscr{M}_W$	**Memory** $\mathscr{M}$
Parity objective		 $\mapsto \{0,1,\ldots,n\}$
$C = \{a, b\}$ $W = b^*ab^*aC^\omega$		
$C = \{a, b\}$ $W = C^*(ab)^\omega$		

Examples

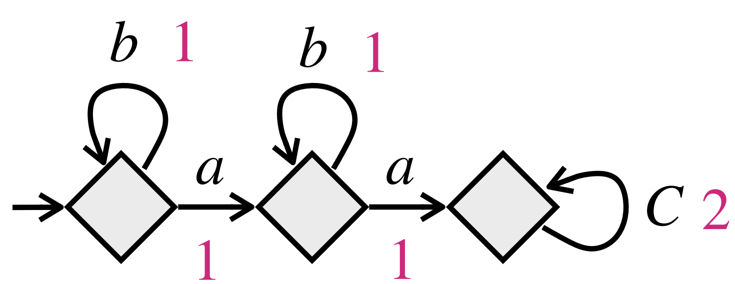
Objective W | **Prefix classifier $\mathscr{M}_W$** | **Memory $\mathscr{M}$**

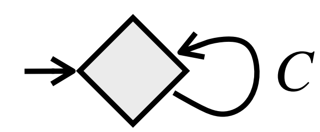
Parity objective



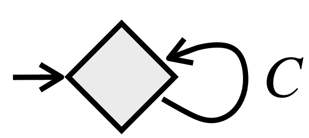


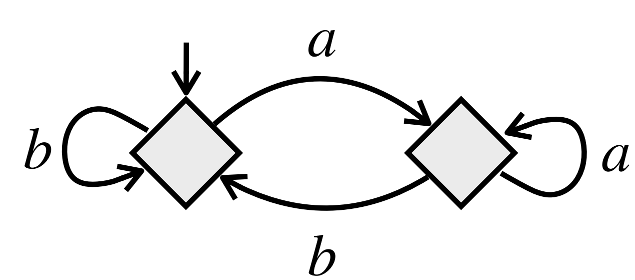
$C = \{a, b\}$
$W = b^*ab^*aC^\omega$



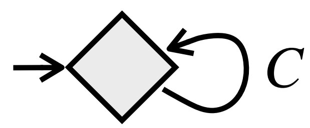
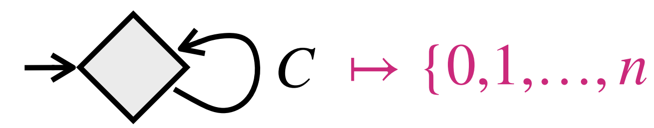
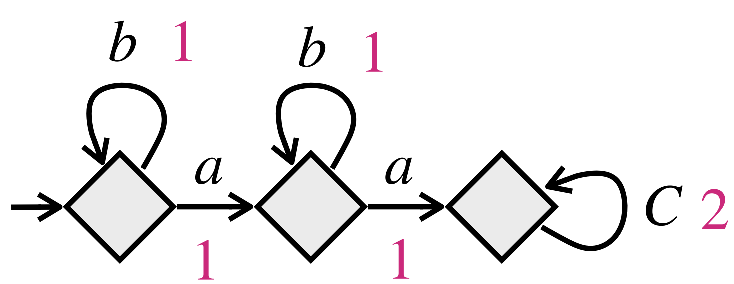
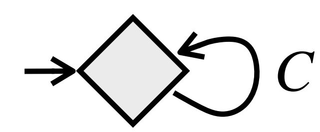
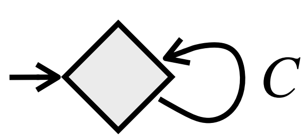
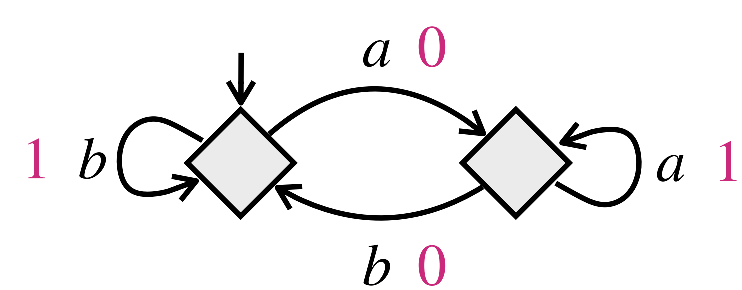


$C = \{a, b\}$
$W = C^*(ab)^\omega$





Examples

Objective W	**Prefix classifier** $\mathscr{M}_W$	**Memory** $\mathscr{M}$
Parity objective		
$C = \{a, b\}$ $W = b^*ab^*aC^\omega$		
$C = \{a, b\}$ $W = C^*(ab)^\omega$		

Corollary

Lifting theorem

If W and W^c are finite-memory-determined in one-player infinite games, then W and W^c are finite-memory-determined in two-player infinite games.

Corollary

Lifting theorem

If W and W^c are finite-memory-determined in one-player infinite games, then W and W^c are finite-memory-determined in two-player infinite games.

Very powerful and extremely useful in practice

- Easier to analyse the one-player case (graph reasoning)
- Lift to two-player games via the theorem

Some consequences

Some consequences

- Mean-payoff ≥ 0 is not ω-regular (even though it is memoryless determined in finite games)

Some consequences

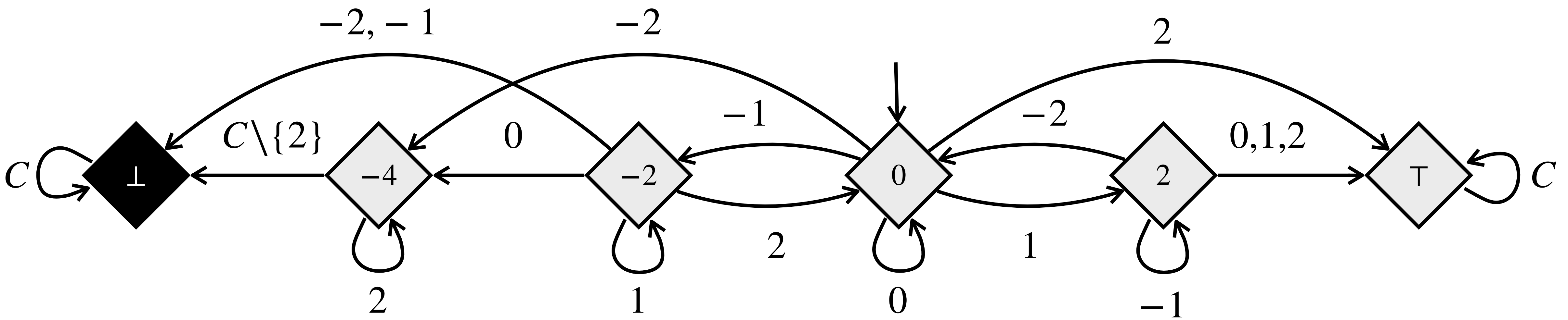
- Mean-payoff ≥ 0 is not ω-regular (even though it is memoryless determined in finite games)

- Some discounted objectives are ω-regular:
  The set of infinite words over $C = \{-2, -1, 0, 1, 2\}$ satisfying $\mathrm{DS}_{\frac{1}{2}}^{\geq 0}$ is the set of infinite words accepted by the DBA below:

Some consequences

- Mean-payoff ≥ 0 is not ω-regular (even though it is memoryless determined in finite games)

- Some discounted objectives are ω-regular:
The set of infinite words over $C = \{-2, -1, 0, 1, 2\}$ satisfying $\mathsf{DS}_{\frac{1}{2}}^{\geq 0}$ is the set of infinite words accepted by the DBA below:



Partial conclusion

Infinite games

Partial conclusion

Infinite games

- Complete characterization of winning objectives that ensure chromatic finite-memory determinacy in infinite games = ω-regular

Partial conclusion

Infinite games

- Complete characterization of winning objectives that ensure chromatic finite-memory determinacy in infinite games = ω-regular

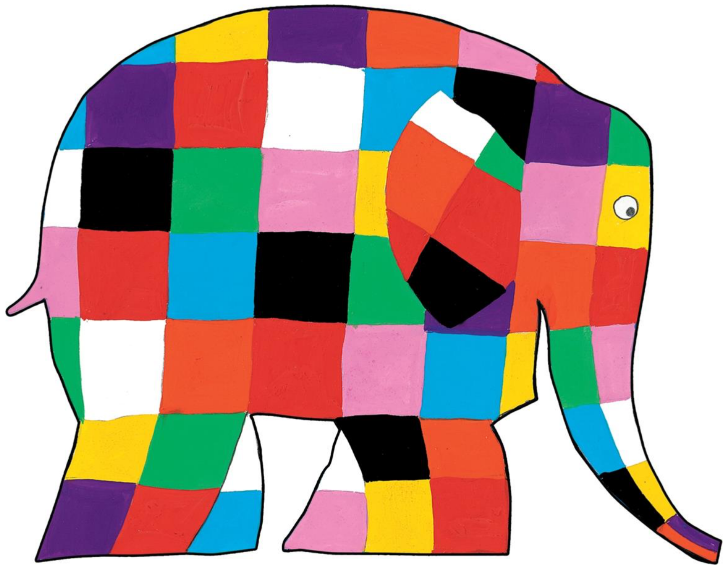
- One-to-two-player lift
  (requires chromatic finite memory determinacy in one-player games for both players;
  ensures chromatic finite memory determinacy in two-players games for both players)

Partial conclusion

Infinite games

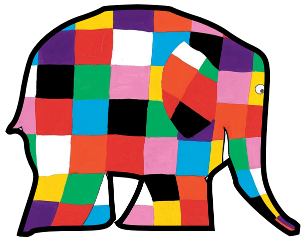
- Complete characterization of winning objectives that ensure chromatic finite-memory determinacy in infinite games = ω-regular
- One-to-two-player lift
(requires chromatic finite memory determinacy in one-player games for both players;
ensures chromatic finite memory determinacy in two-players games for both players)
- Further questions:
  - Different results when assuming finite branching?

Going further?

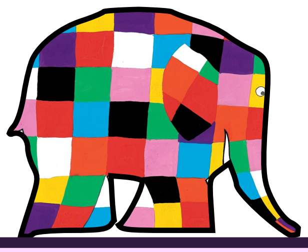


What more?

What more?

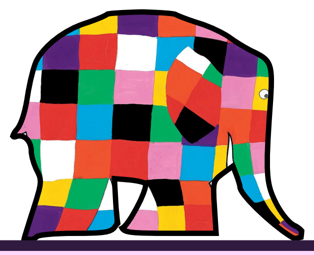


What more?



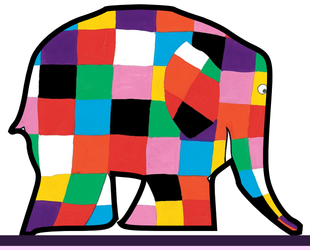
- So far, nice general characterizations

What more?



- So far, nice general characterizations
- However:
  - Memory bounds are not tight in general
  - Makes assumptions on the memory for the two players

What more?

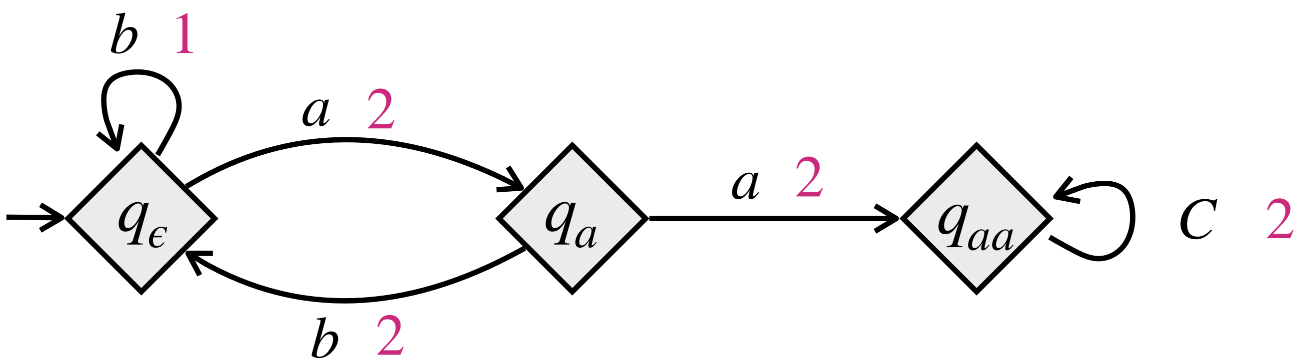


- So far, nice general characterizations
- However:
  - Memory bounds are not tight in general
  - Makes assumptions on the memory for the two players

→ Precise memory of the two players for ω-regular objectives?
(we will see it is non-trivial in general)

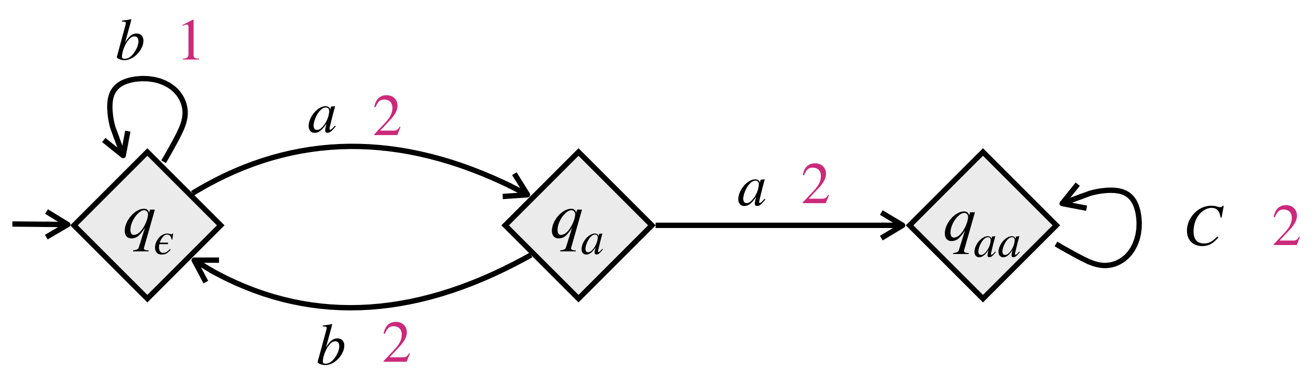
Example

$$W = (b^*a)^\omega \cup C^*aaC^\omega$$



Example

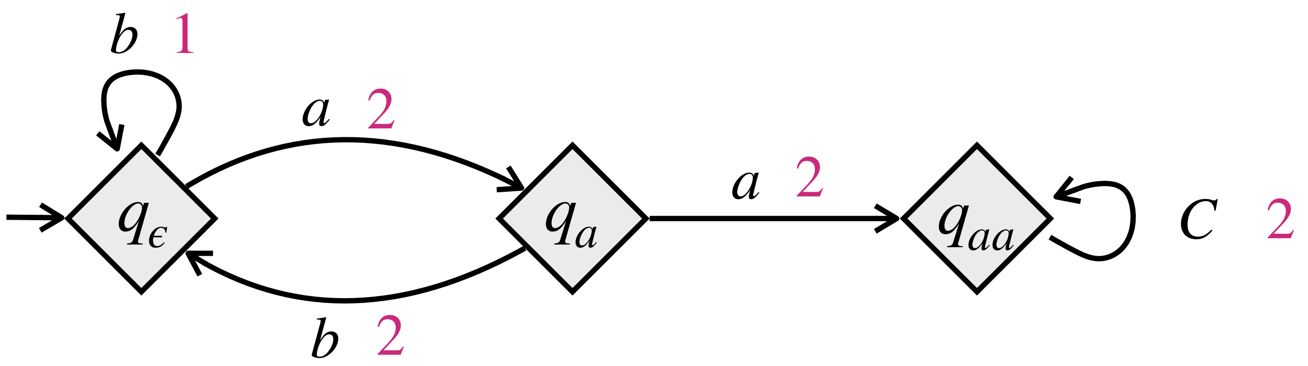
$W = (b^*a)^\omega \cup C^*aaC^\omega$



- Smallest DPA $\mathscr{A}_W$ recognizing W
- The prefix classifier $\mathscr{M}_W$ has the same structure

Example

$W = (b^*a)^\omega \cup C^*aaC^\omega$

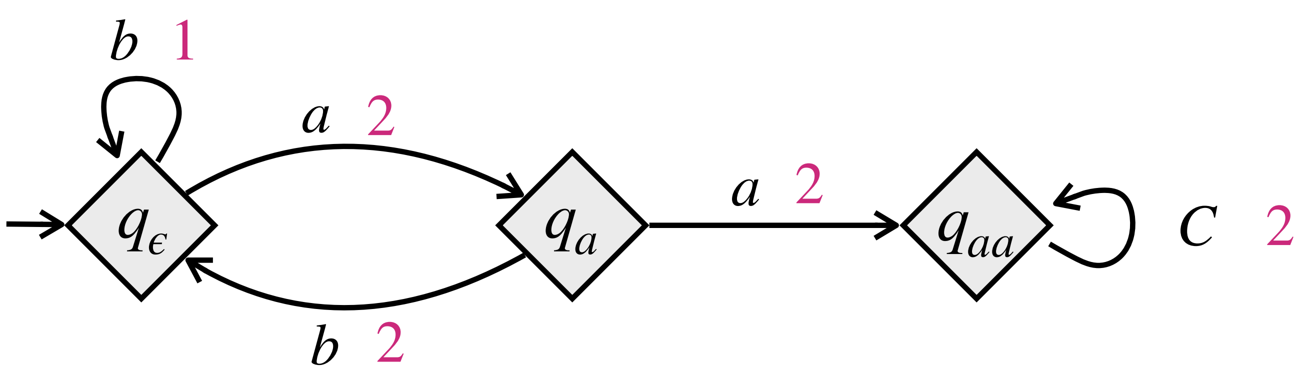


- Smallest DPA $\mathscr{A}_W$ recognizing W
- The prefix classifier $\mathscr{M}_W$ has the same structure

▸ The two players can play optimally with a memory structure based on $\mathscr{A}_W$

Example

$W = (b^*a)^\omega \cup C^*aaC^\omega$

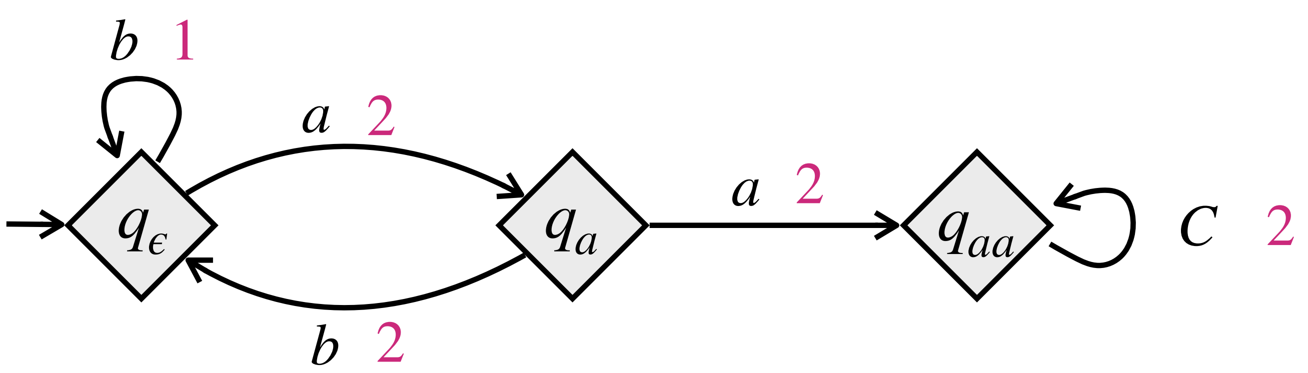


- Smallest DPA $\mathcal{A}_W$ recognizing W
- The prefix classifier $\mathcal{M}_W$ has the same structure

- The two players can play optimally with a memory structure based on $\mathcal{A}_W$
- The memory required stands between one state (memoryless) and three states, for both players

Example

$$W = (b^*a)^\omega \cup C^*aaC^\omega$$

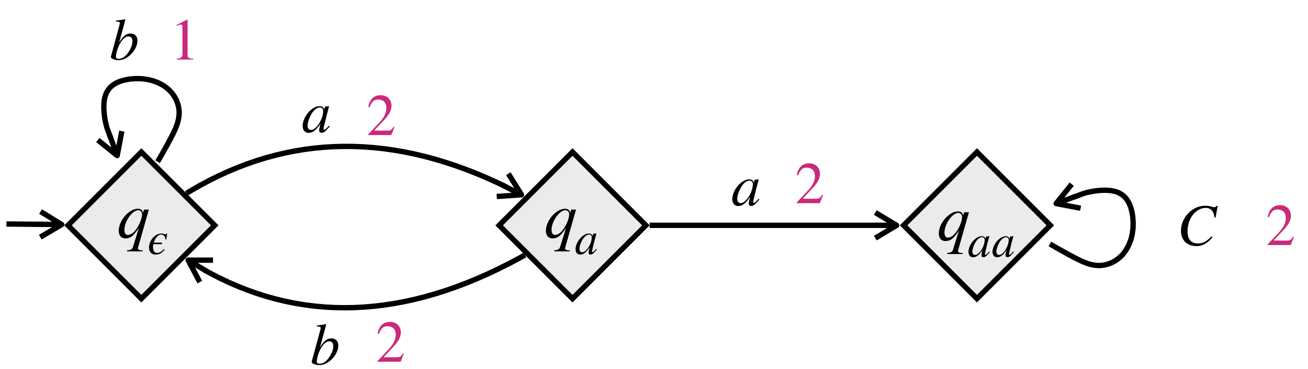


- Smallest DPA $\mathcal{A}_W$ recognizing W
- The prefix classifier $\mathcal{M}_W$ has the same structure

- The two players can play optimally with a memory structure based on $\mathcal{A}_W$
- The memory required stands between one state (memoryless) and three states, for both players
  - W is half-positional: P_1 requires only memoryless strategies to win W

Example

$W = (b^*a)^\omega \cup C^*aaC^\omega$



- Smallest DPA $\mathcal{A}_W$ recognizing W
- The prefix classifier $\mathcal{M}_W$ has the same structure

- The two players can play optimally with a memory structure based on $\mathcal{A}_W$
- The memory required stands between one state (memoryless) and three states, for both players
  - W is half-positional: P_1 requires only memoryless strategies to win W
  - P_2 requires just two states of memory: q_ϵ and q_a

The example of Muller conditions

- $\mathcal{F} \subseteq 2^C$
  $W_{\mathcal{F}} = \{w \in C^\omega \mid \{c \in C \mid \exists^\infty i \text{ s.t. } w_i = c\} \in \mathcal{F}\}$

[Cas22] Casares. On the minimisation of transition-based Rabin automata and the chromatic memory requirements of Muller condition (CSL'22)

The example of Muller conditions

- $\mathcal{F} \subseteq 2^C$
  $W_{\mathcal{F}} = \{w \in C^\omega \mid \{c \in C \mid \exists^\infty i \text{ s.t. } w_i = c\} \in \mathcal{F}\}$

Chromatic memory for $W_{\mathcal{F}}$

A memory structure $\mathcal{M}$ suffices for P_1 for $W_{\mathcal{F}}$ if and only if $W_{\mathcal{F}}$ is recognized by a deterministic Rabin automaton built on top of $\mathcal{M}$ [Cas22]. It is NP-complete to decide whether there is a memory structure of size k that is sufficient to win a Muller condition.

[Cas22] Casares. On the minimisation of transition-based Rabin automata and the chromatic memory requirements of Muller condition (CSL'22)

The special case of objectives given by DBA [BCRV22]

[BCRV22] Bouyer, Casares, Randour, Vandenhove. Half-positional objectives recognized by deterministic Büchi automata (CONCUR'22)

The special case of objectives given by DBA [BCRV22]

- W given by a DBA (= Deterministic Büchi automaton)

[BCRV22] Bouyer, Casares, Randour, Vandenhove. Half-positional objectives recognized by deterministic Büchi automata (CONCUR'22)

The special case of objectives given by DBA [BCRV22]

- W given by a DBA (= Deterministic Büchi automaton)
- Only their half-positionality has been fully characterized

[BCRV22] Bouyer, Casares, Randour, Vandenhove. Half-positional objectives recognized by deterministic Büchi automata (CONCUR'22)

The special case of objectives given by DBA [BCRV22]

- W given by a DBA (= Deterministic Büchi automaton)
- Only their half-positionality has been fully characterized

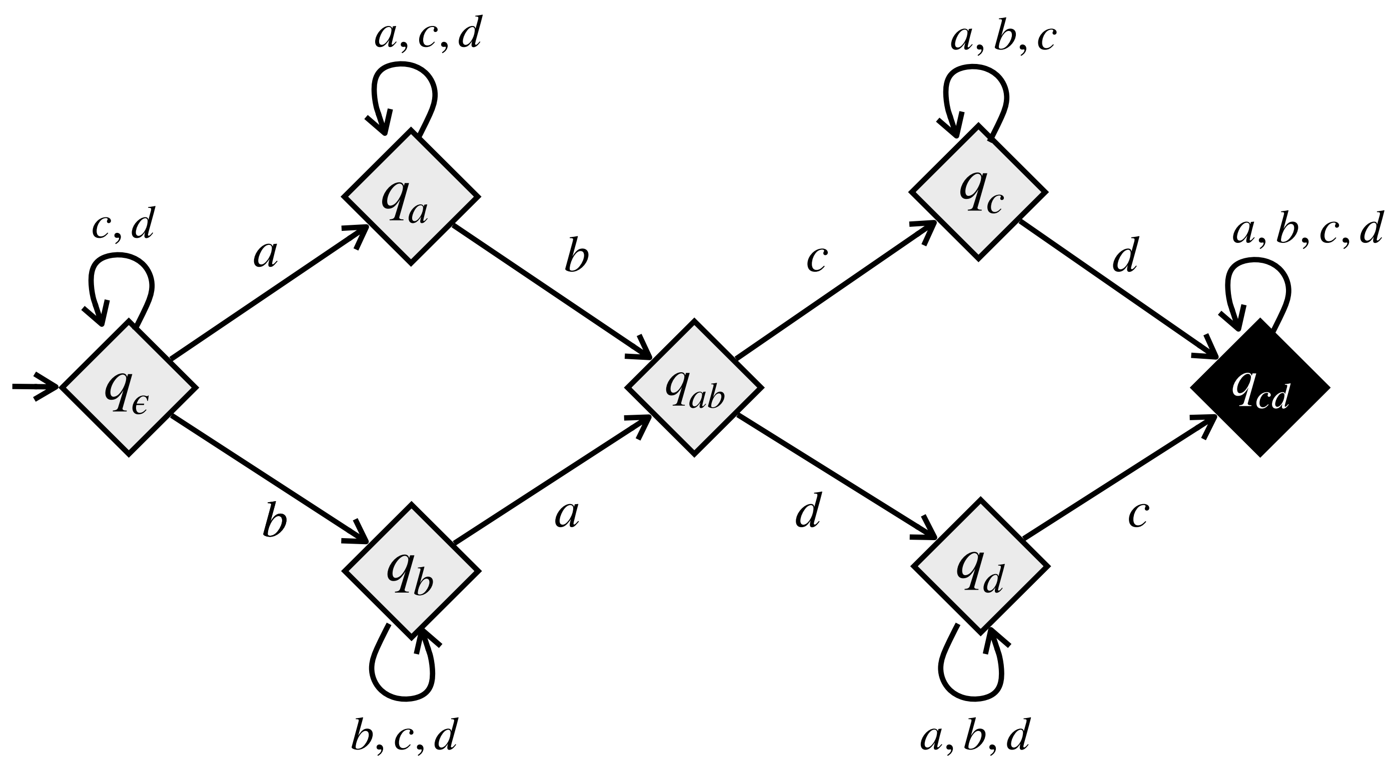
Half-positionality of W can be decided in PTIME

An objective W defined by a DBA is half-positional if and only if:

1. W is monotone;
2. W is progress consistent: if w_2 is a progress after w_1, then $w_1 w_2^\omega$ is winning;
3. W is recognized by a DBA built on top of its prefix classifier

[BCRV22] Bouyer, Casares, Randour, Vandenhove. Half-positional objectives recognized by deterministic Büchi automata (CONCUR'22)

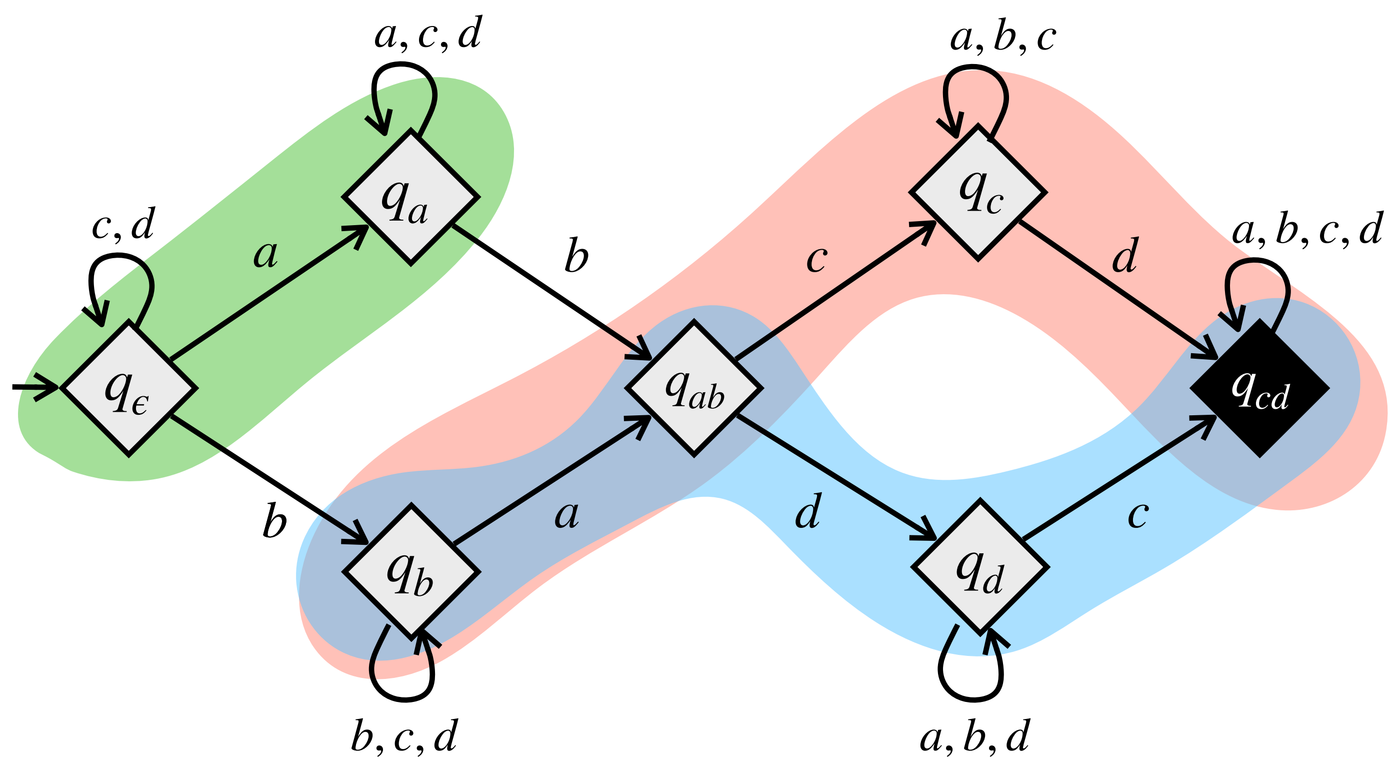
Regular safety and reachability objectives [BFRV22]



$W =$ avoid the rightmost state

[BFRV22] Bouyer, Fijalkow, Randour, Vandenhove. How to Play Optimally for Regular Objectives? (Submitted)

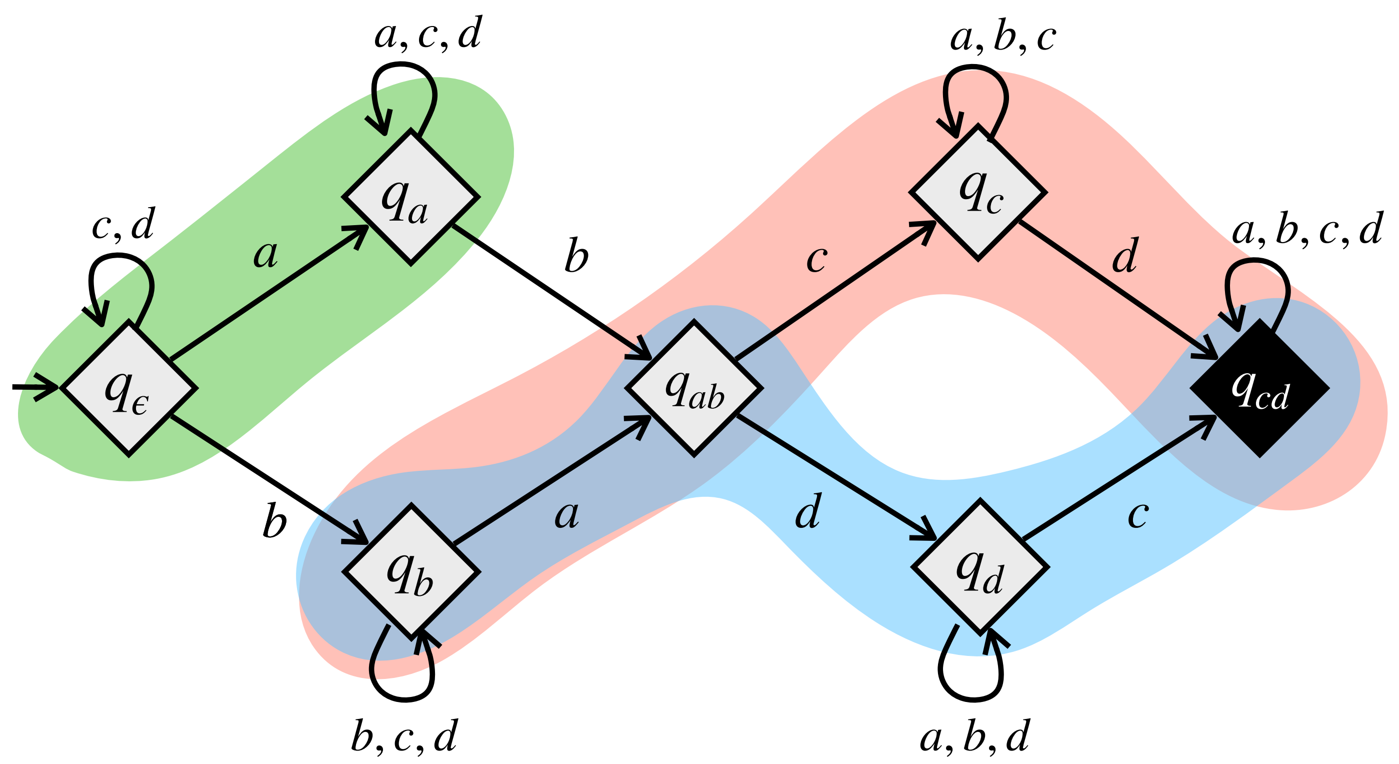
Regular safety and reachability objectives [BFRV22]



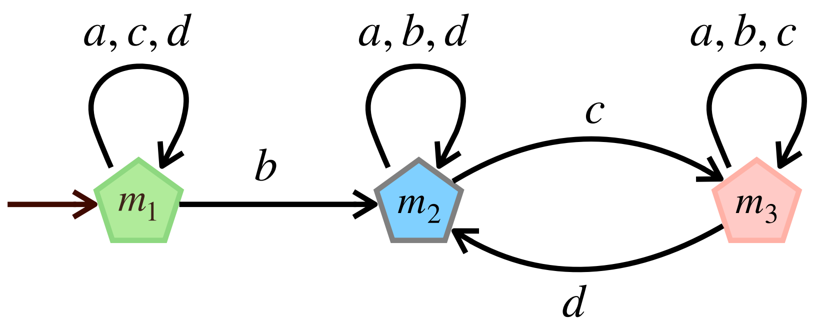
$W =$ avoid the rightmost state

[BFRV22] Bouyer, Fijalkow, Randour, Vandenhove. How to Play Optimally for Regular Objectives? (Submitted)

Regular safety and reachability objectives [BFRV22]



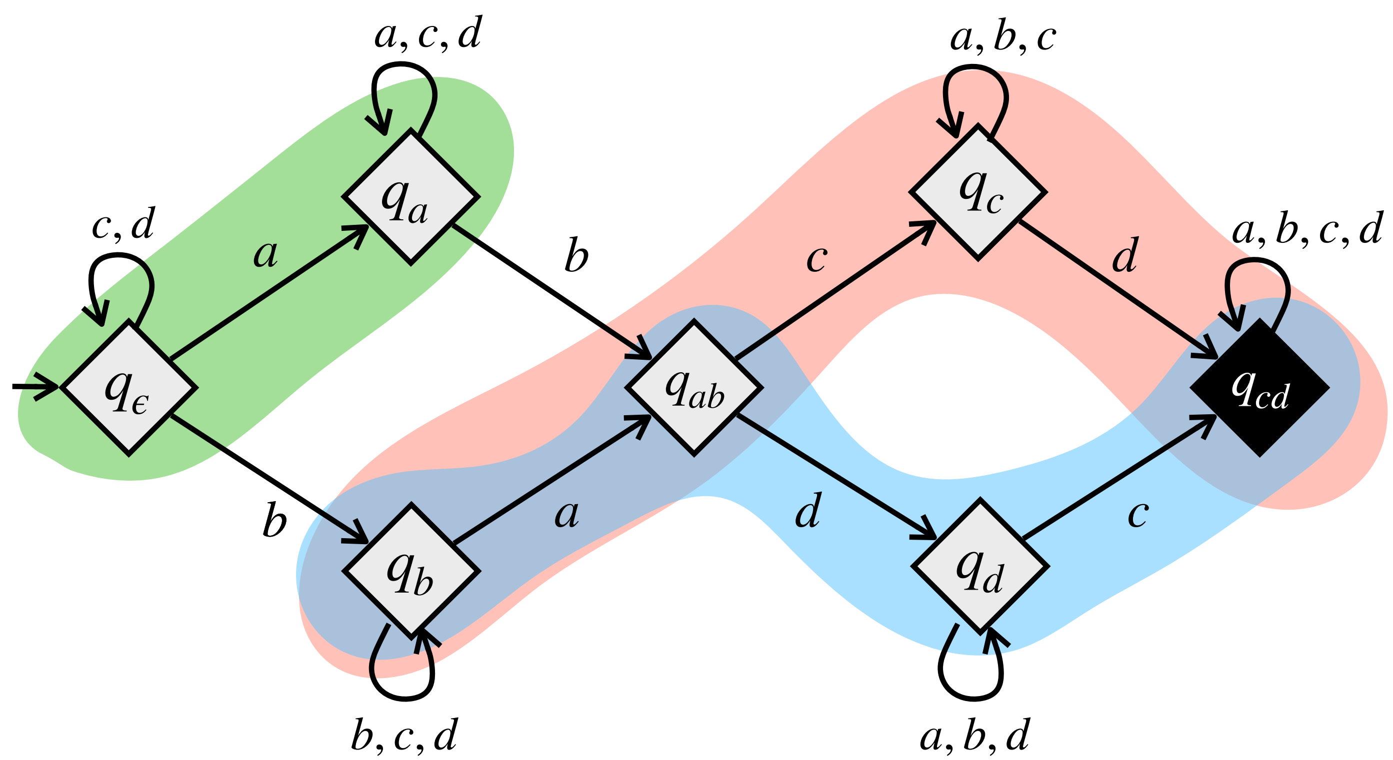
$W =$ avoid the rightmost state



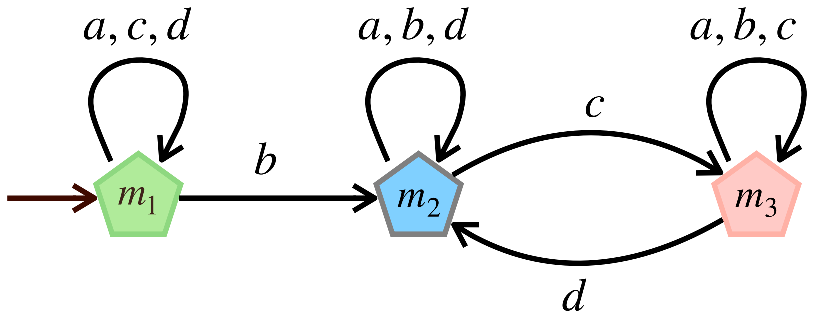
Tightest memory to win W

[BFRV22] Bouyer, Fijalkow, Randour, Vandenhove. How to Play Optimally for Regular Objectives? (Submitted)

Regular safety and reachability objectives [BFRV22]



$W =$ avoid the rightmost state



Tightest memory to win W

It is NP-complete to decide whether there is a memory structure of size k that is sufficient to win a regular safety/reachability objective.

[BFRV22] Bouyer, Fijalkow, Randour, Vandenhove. How to Play Optimally for Regular Objectives? (Submitted)

Double lift

- Let $W \subseteq C^\omega$ be a regular reachability or safety objective

Double lift

- Let $W \subseteq C^\omega$ be a regular reachability or safety objective

The double-lift theorem

If $\mathscr{M}$ suffices to win for W in finite P_1-arenas, then $\mathscr{M}$ suffices to win for W for P_1 in (infinite) two-player arenas.

Double lift

- Let $W \subseteq C^\omega$ be a regular reachability or safety objective

The double-lift theorem

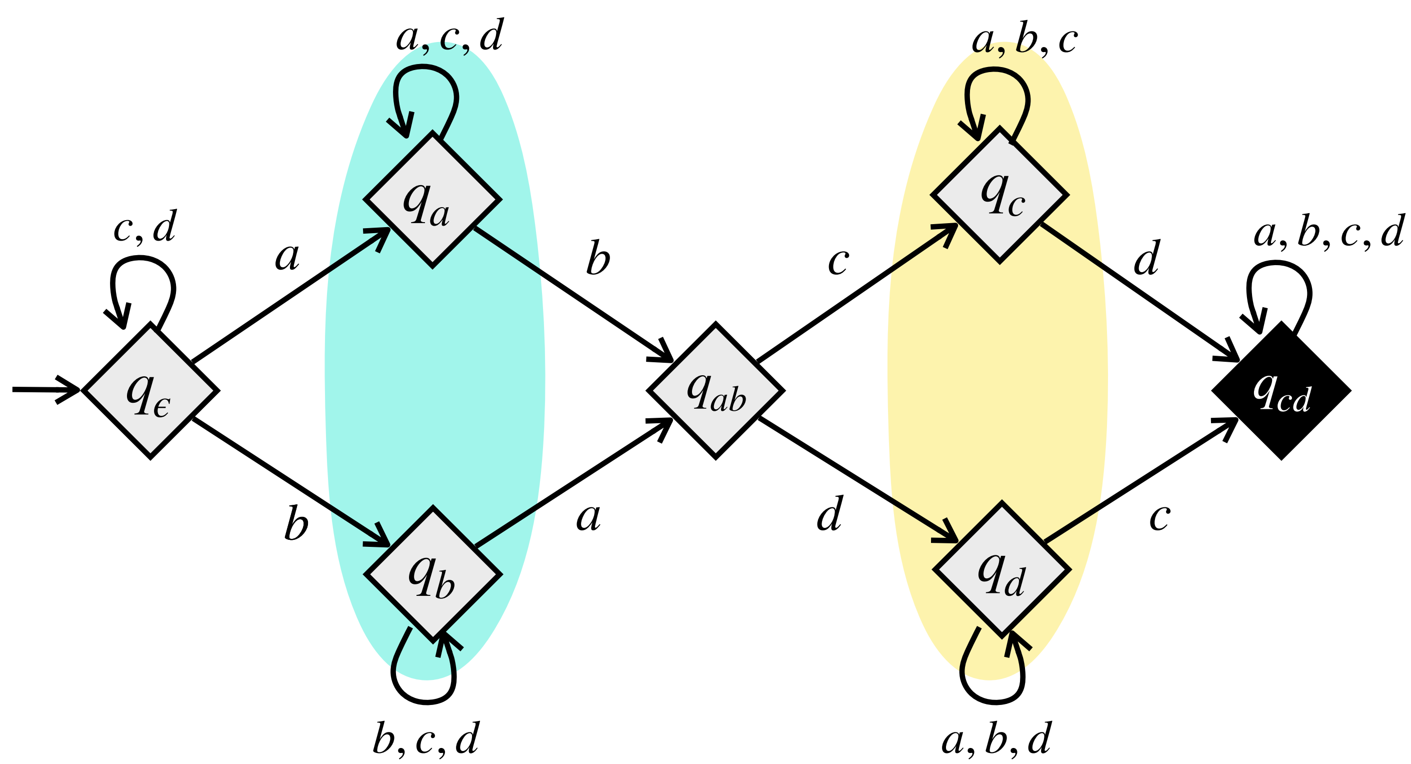
If $\mathscr{M}$ suffices to win for W in finite P_1-arenas, then $\mathscr{M}$ suffices to win for W for P_1 in (infinite) two-player arenas.

Very powerful and extremely useful in practice

- Easy to analyse the one-player finite case (finite graph reasoning)
- Lift to infinite two-player games via the theorem

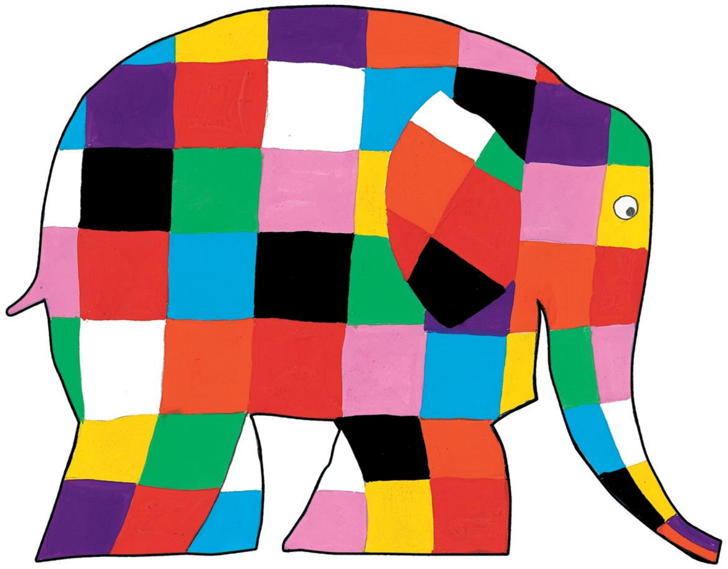
What about chaotic memory?

- Chaotic memory is more difficult to grasp
- In the previous example, only two memory states are sufficient (size of the largest antichain) [CFH14]



[CFH14] Colcombet, Fijalkow, Horn. Playing safe (FSTTCS'14)

Conclusion



What you can bring home

What you can bring home

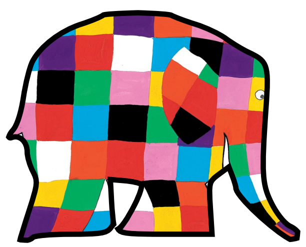
- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)

What you can bring home

- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)
- These concepts (like winning strategies) require manipulating information
  - For simpler strategies, use **low memory**!
  - … even though low memory does not mean it is easy…

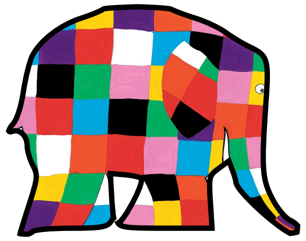
What you can bring home

- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)
- These concepts (like winning strategies) require manipulating information
  - For simpler strategies, use **low memory**!
  - ... even though low memory does not mean it is easy...
- Understand **chromatic** **finite-memory** determined objectives



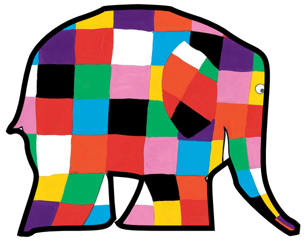
What you can bring home

- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)
- These concepts (like winning strategies) require manipulating information
  - For simpler strategies, use **low memory**!
  - ... even though low memory does not mean it is easy...
- Understand **chromatic finite-memory** determined objectives
- Many **one-to-two-player lifts**



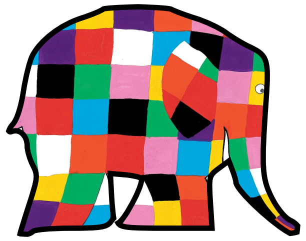
What you can bring home

- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)
- These concepts (like winning strategies) require manipulating information
  - For simpler strategies, use **low memory**!
  - ... even though low memory does not mean it is easy...
- Understand **chromatic finite-memory** determined objectives
- Many **one-to-two-player lifts**
- Fine tune the memory requirements for **ω-regular objectives**
  - Preliminary results, but no general understanding
  - Half-positionality



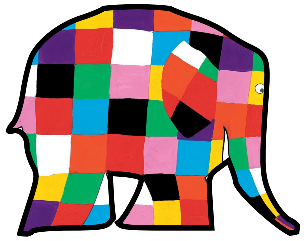
What you can bring home

- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)
- These concepts (like winning strategies) require manipulating information
  - For simpler strategies, use **low memory**!
  - ... even though low memory does not mean it is easy...
- Understand **chromatic** **finite-memory** determined objectives
- Many **one-to-two-player lifts**
- Fine tune the memory requirements for **ω-regular objectives**
  - Preliminary results, but no general understanding
  - Half-positionality
- **Chaotic memory**
  - Link with goof-for-game automata [CCL22]
  - Universal graphs [Ohl22]



What you can bring home

- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)
- These concepts (like winning strategies) require manipulating information
  - For simpler strategies, use **low memory**!
  - ... even though low memory does not mean it is easy...
- Understand **chromatic** **finite-memory** determined objectives
- Many **one-to-two-player lifts**
- Fine tune the memory requirements for ω**-regular objectives**
  - Preliminary results, but no general understanding
  - Half-positionality
- **Chaotic memory**
  - Link with goof-for-game automata [CCL22]
  - Universal graphs [Ohl22]



Quite active area of research

[CCL22] Casares, Colcombet, Lehtinen.On the size of good-for-game Rabin automata and its link with the memory in Muller games (ICALP'22)

[Ohl22] Ohlmann. Characterizing positionality in games of infinite duration over infinite graphs (LICS'22)