Laboratoire Méthodes Formelles

Memory complexity for winning games on graphs

Patricia Bouyer

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

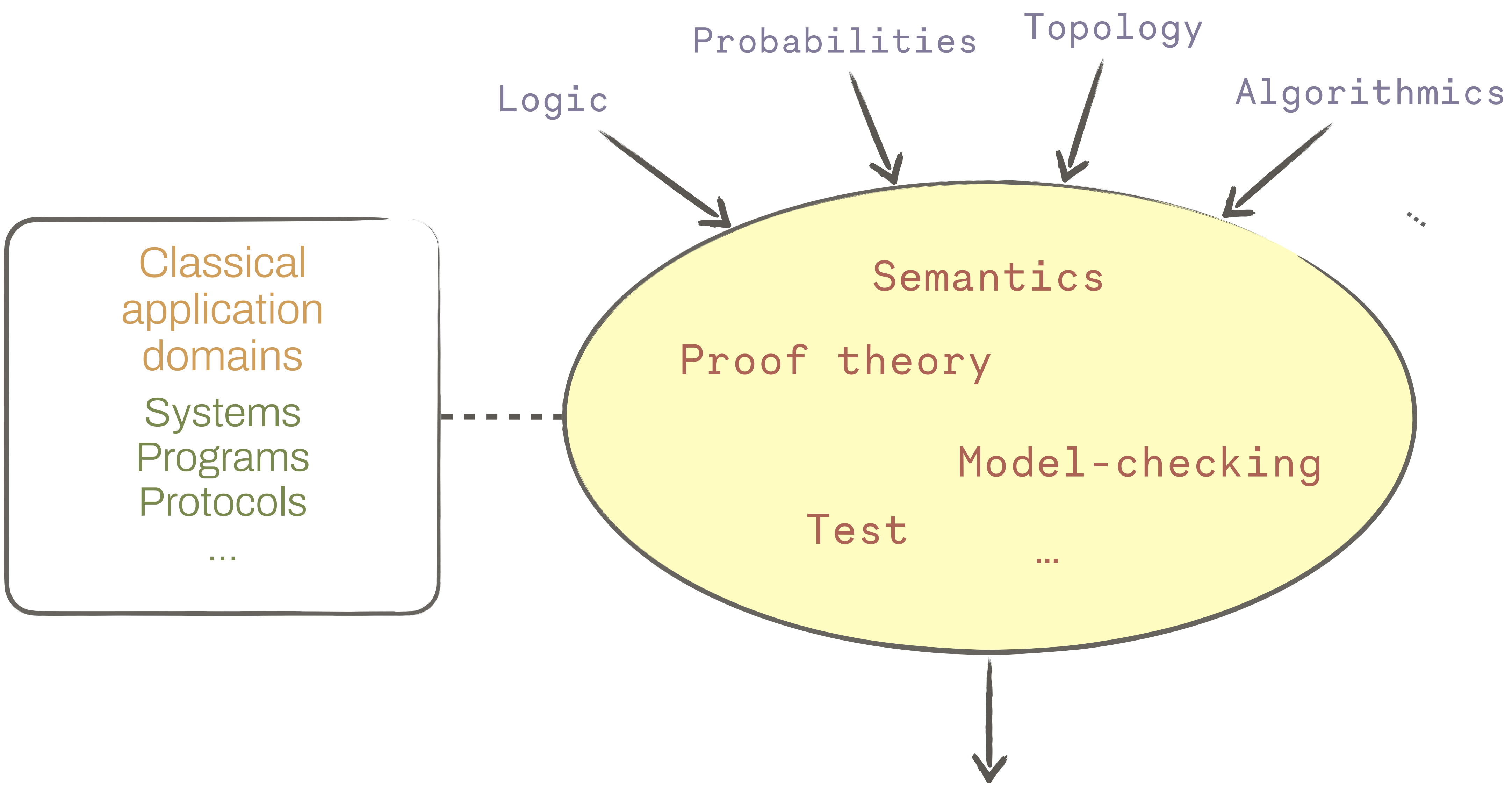
Based on joined work with Stéphane Le Roux, Youssouf Oualhadj, Michael Randour, Pierre Vandenhove. Thanks to Pierre for his slides

Motivation

—

The setting

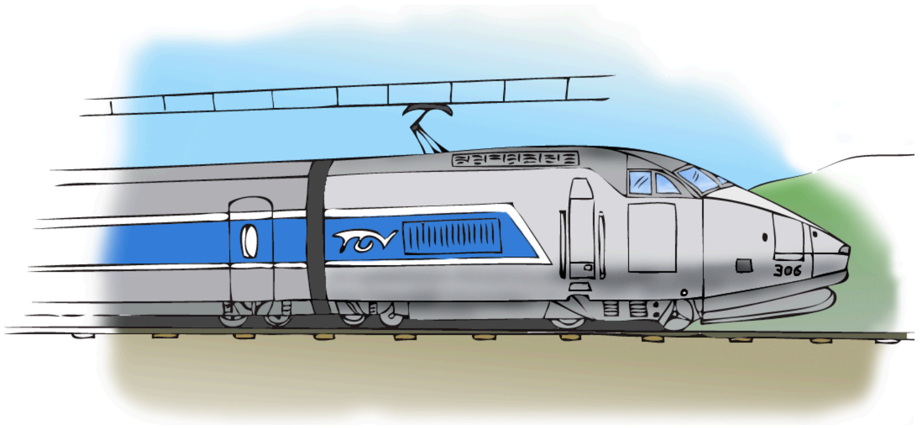
My field of research: Formal methods



Give guarantees (+ certificates) on functionalities or performances

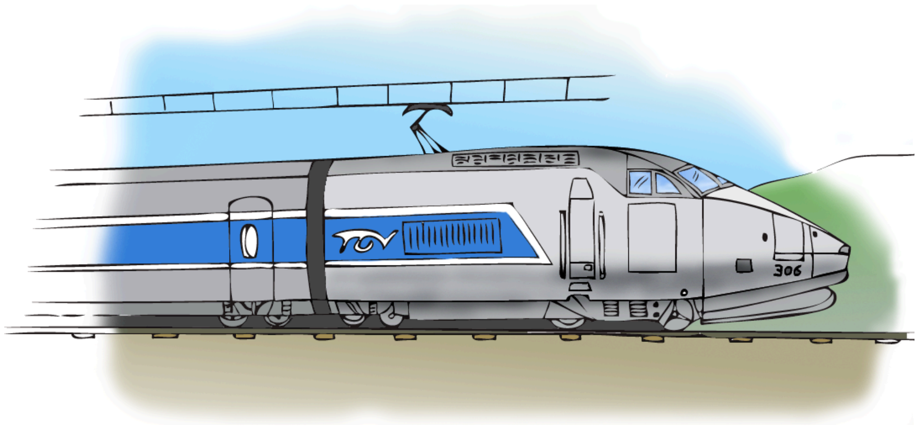
Model-checking

System



Model-checking

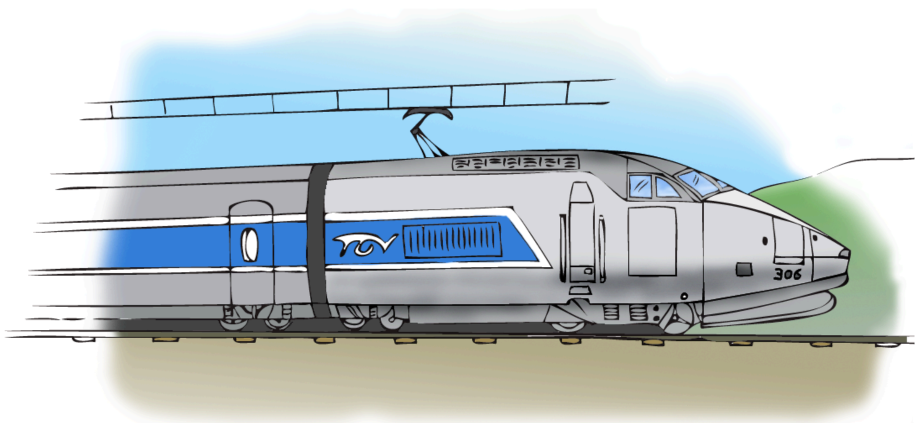
System



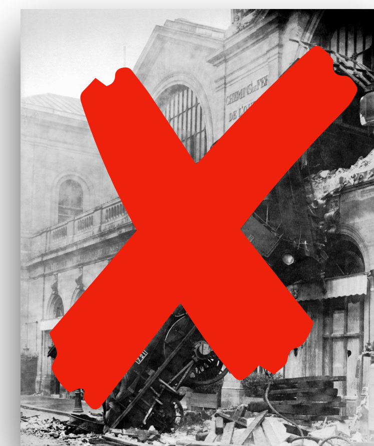
Properties

Model-checking

System

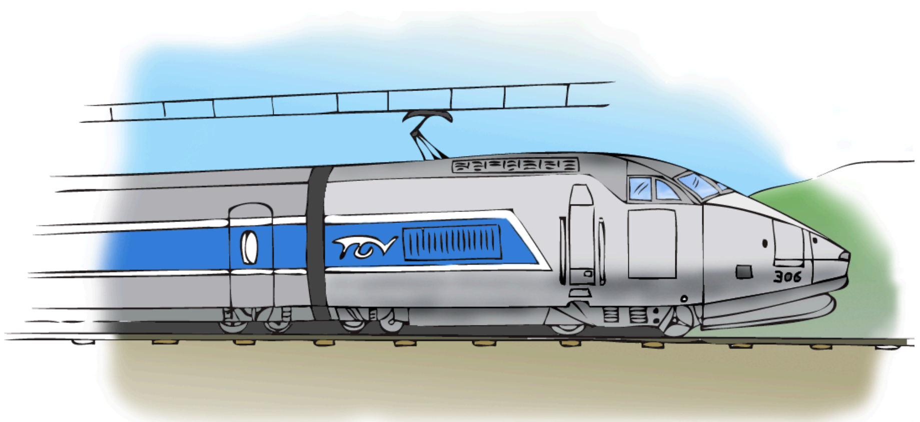


Properties

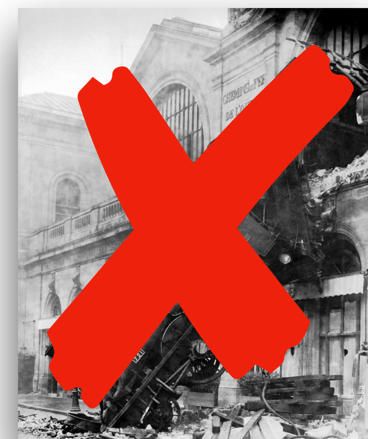


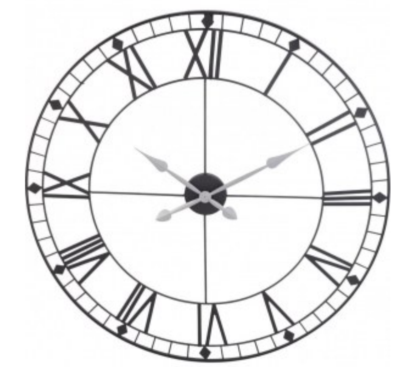
Model-checking

System



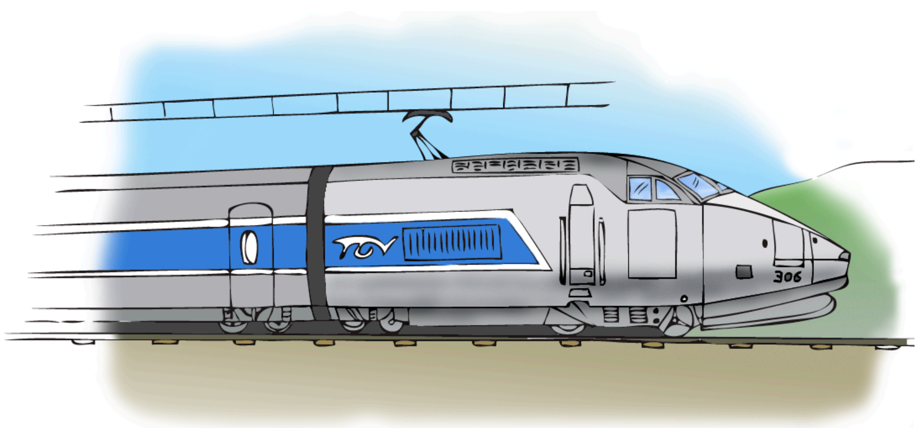
Properties

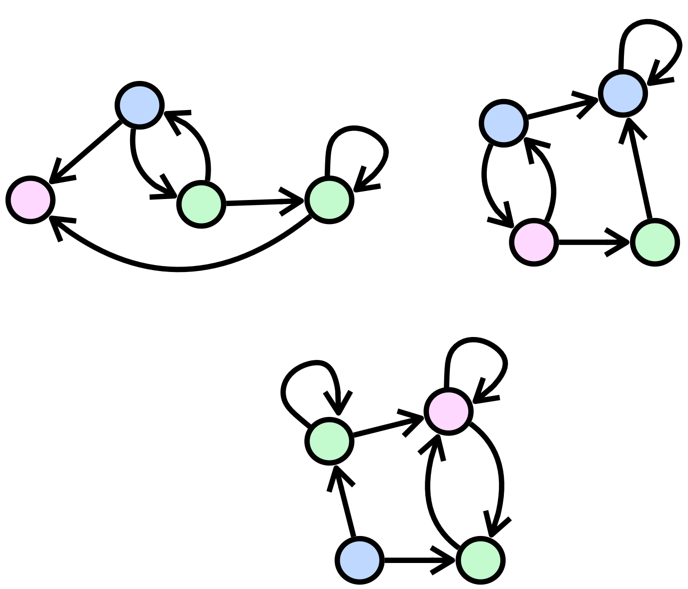




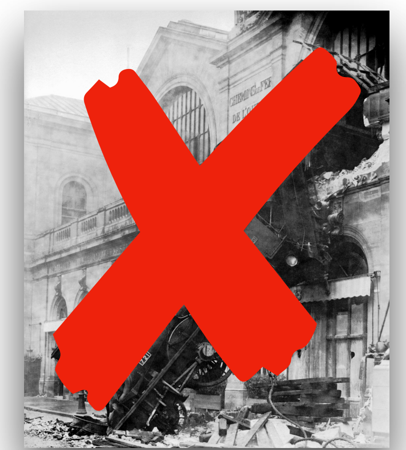
Model-checking

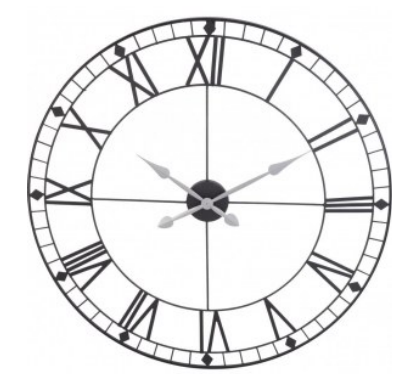
System





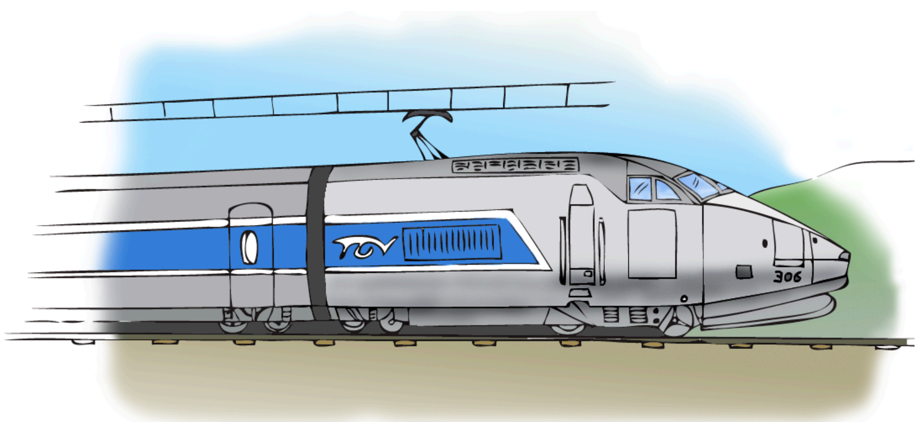
Properties

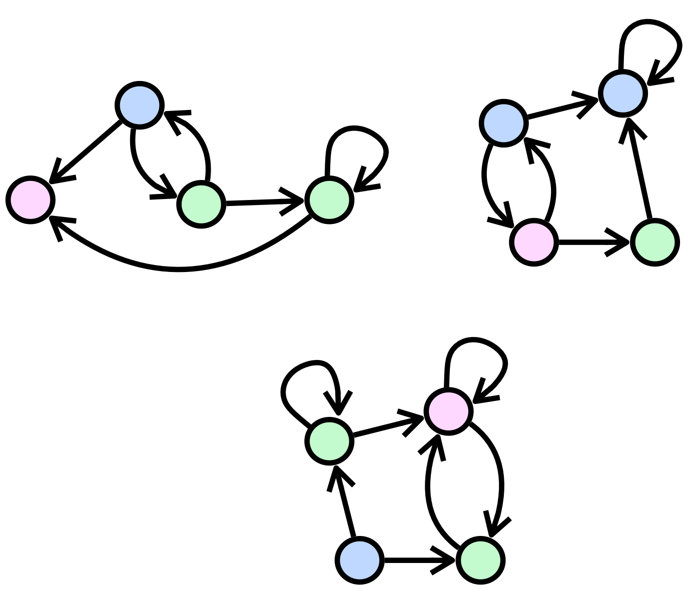




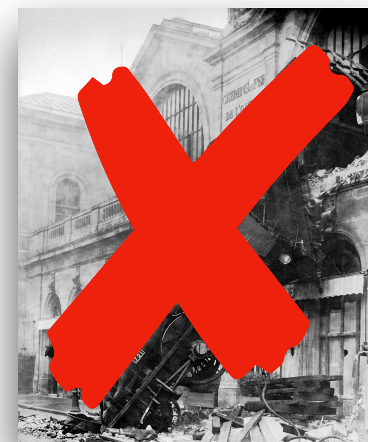
Model-checking

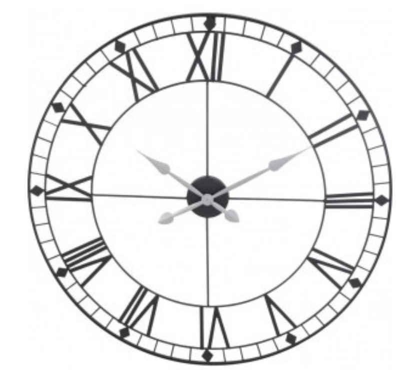
System





Properties

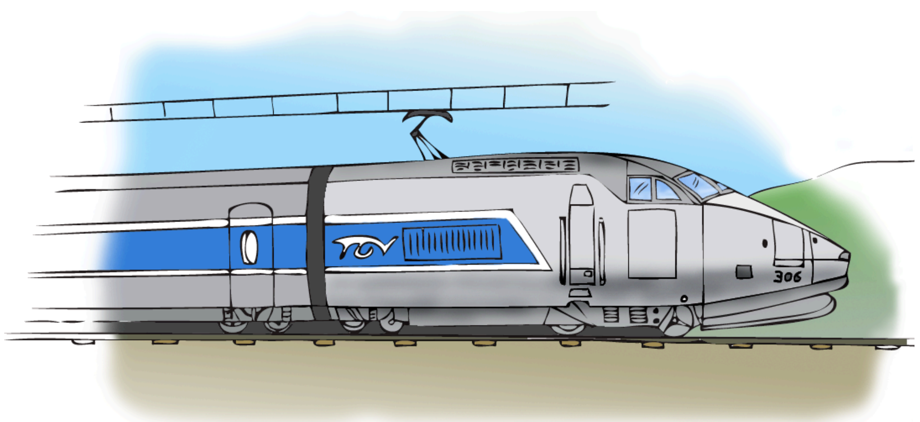




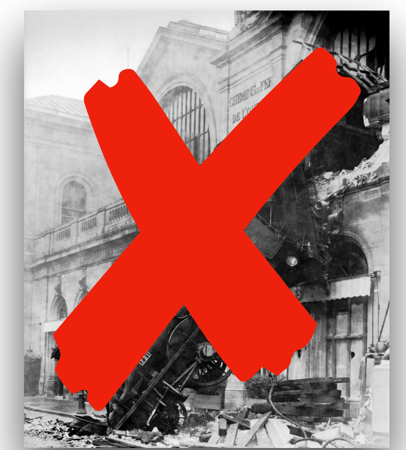
$$\varphi = \mathbf{AG}\,\neg\text{crash} \wedge \left(\mathbb{P}(\mathbf{F}_{\leq 2\text{h}}\,\text{arr}) \geq 0{,}9\right)$$

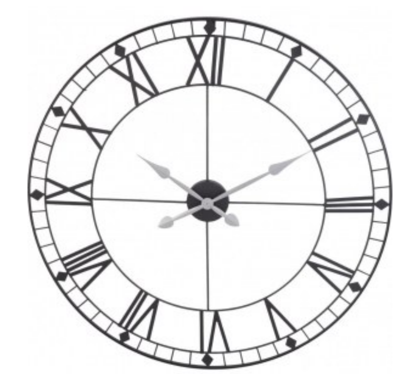
Model-checking

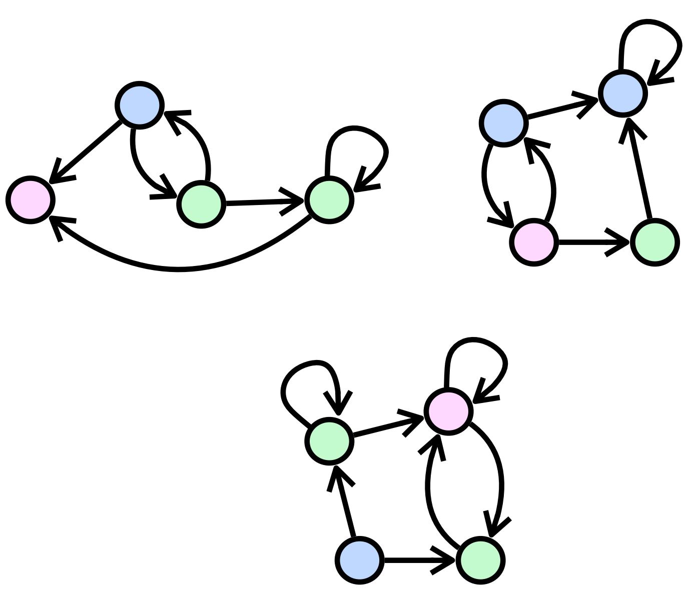
System



Properties





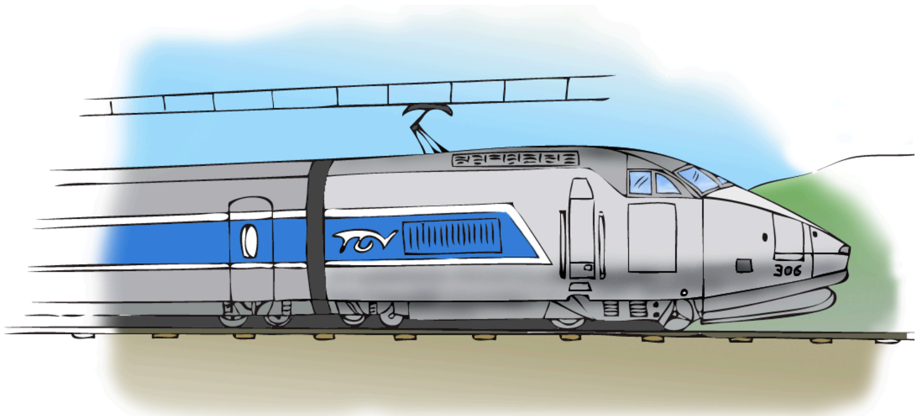


Model-checking algorithm

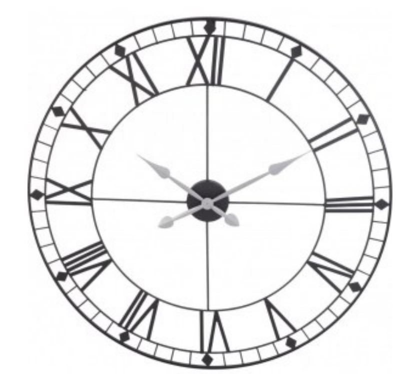
$$\varphi = \mathbf{AG}\,\neg\text{crash} \wedge \left(\mathbb{P}(\mathbf{F}_{\leq 2\text{h}}\,\text{arr}) \geq 0{,}9\right)$$

Model-checking

System

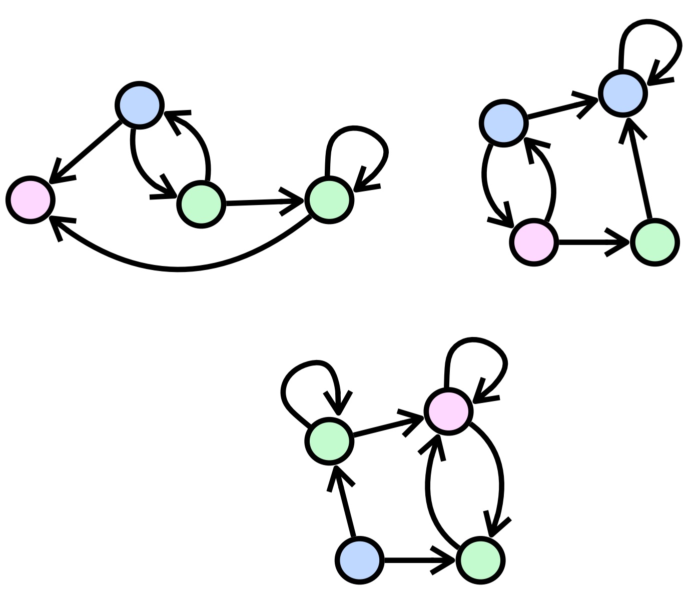


Properties



⇓

⇓



Model-checking algorithm

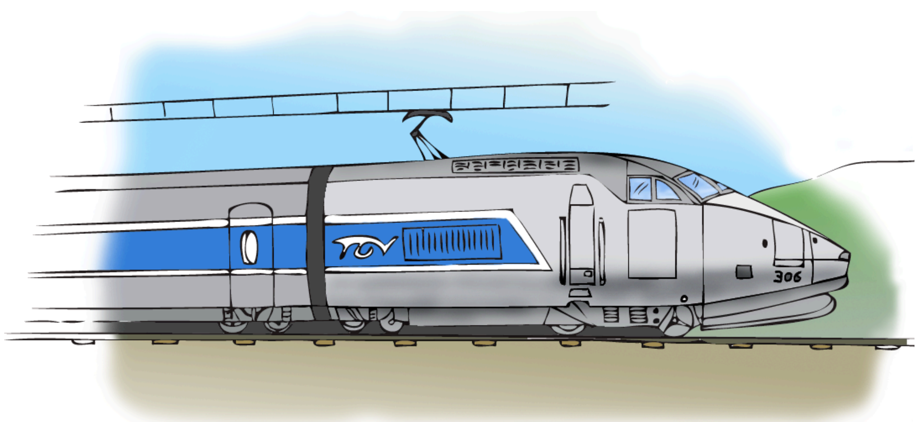
↓

Yes/No/Why?

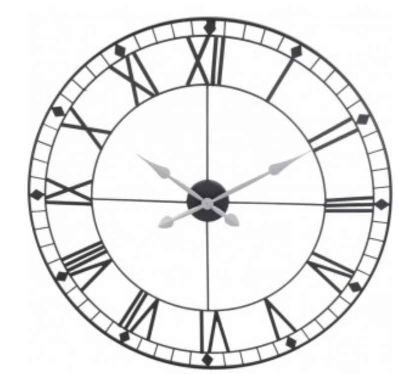
$$\varphi = \mathbf{AG}\,\neg\text{crash} \wedge \left(\mathbb{P}(\mathbf{F}_{\leq 2\text{h}}\text{arr}) \geq 0{,}9\right)$$

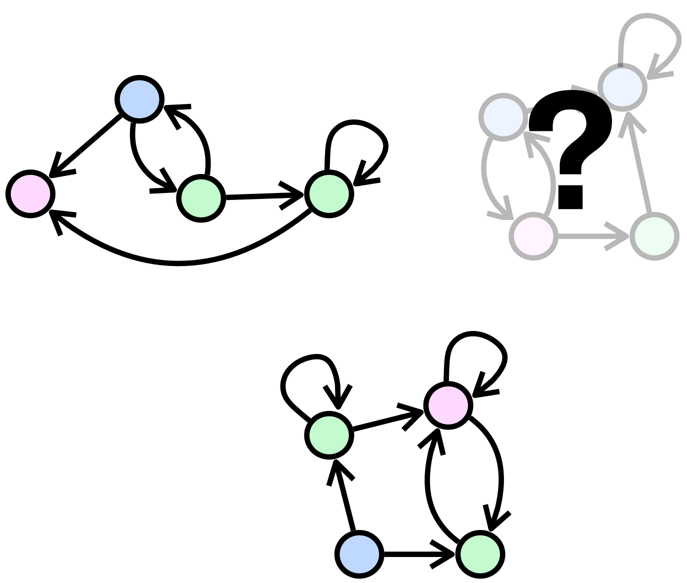
Control or synthesis

System



Properties





Control/synthesis algorithm

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

No/Yes/How?

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

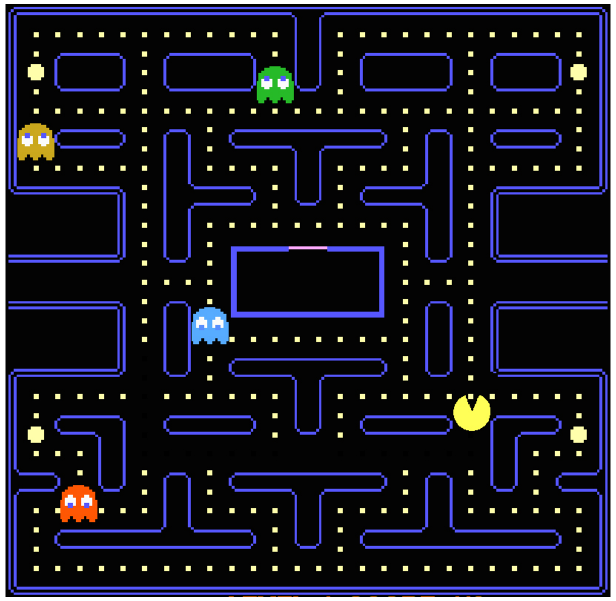
[Tho95] On the synthesis of strategies in infinite games (STACS'95).
[Tho02] Thomas. Infinite games and verification (CAV'02).
[GU08] Grädel, Ummels. Solution concepts and algorithms for infinite multiplayer games (New Perspectives in Games and Interactions, 2008).
[BCJ18] Bloem, Chatterjee, Jobstmann.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

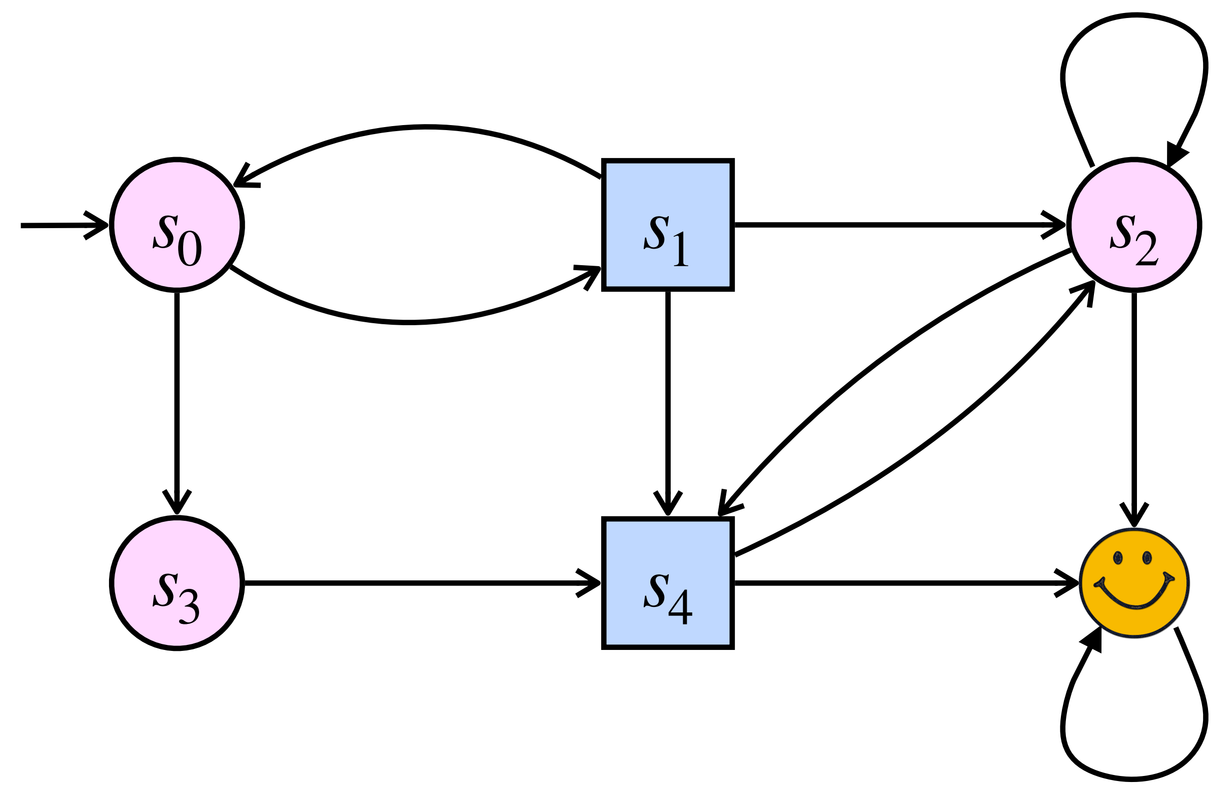
Games on graphs

States, Edges

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

○ : player P_1

□ : player P_2



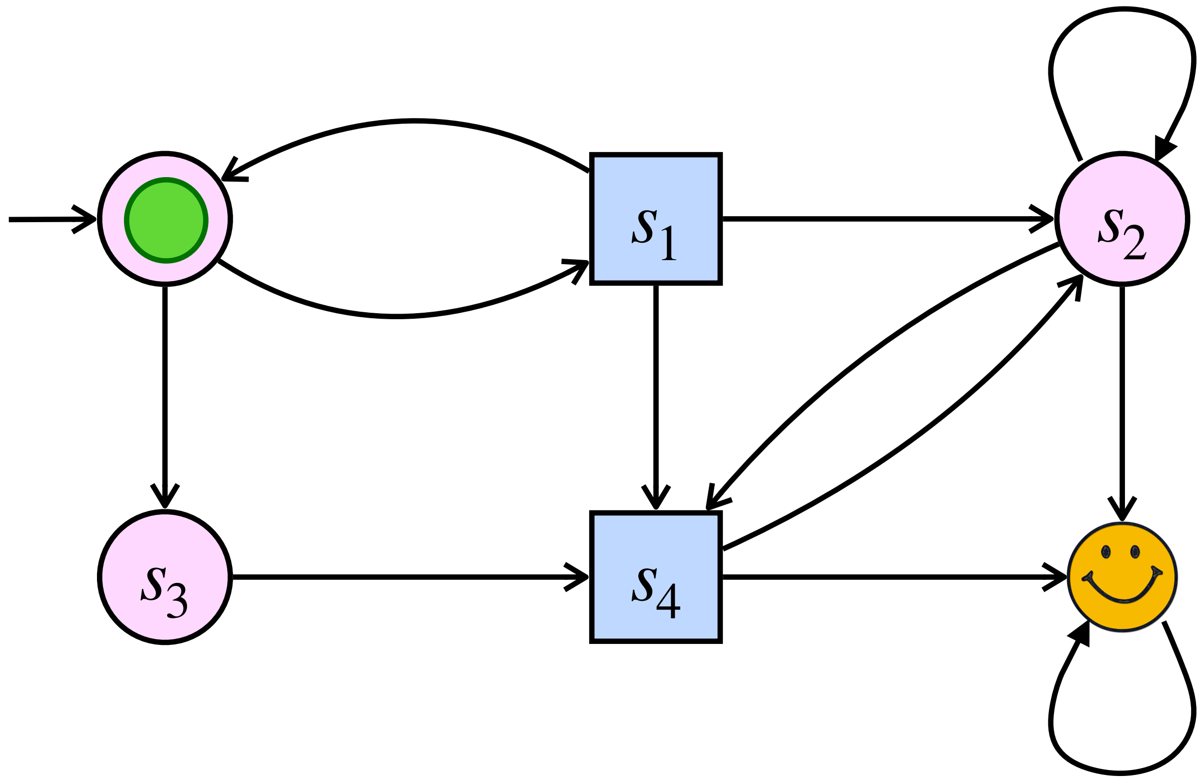
Games on graphs

States → S, Edges → E

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

○ : player P_1

□ : player P_2



s_0

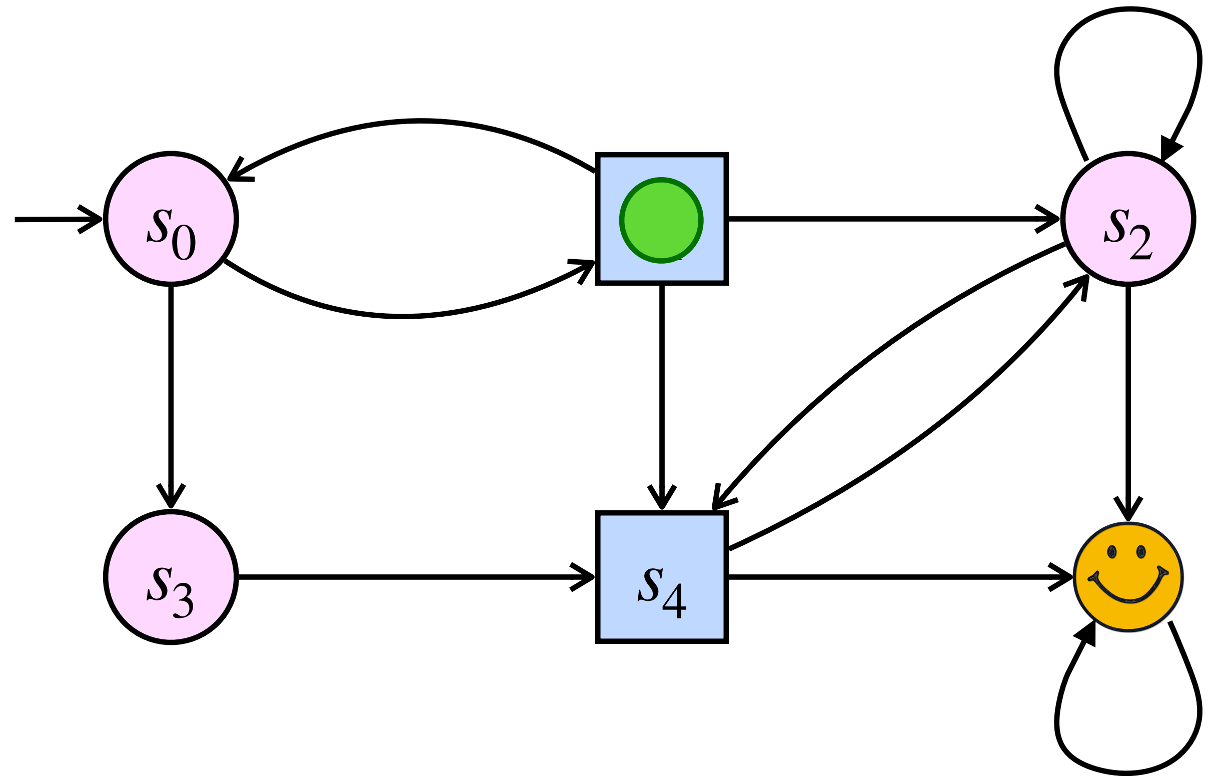
Games on graphs

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)

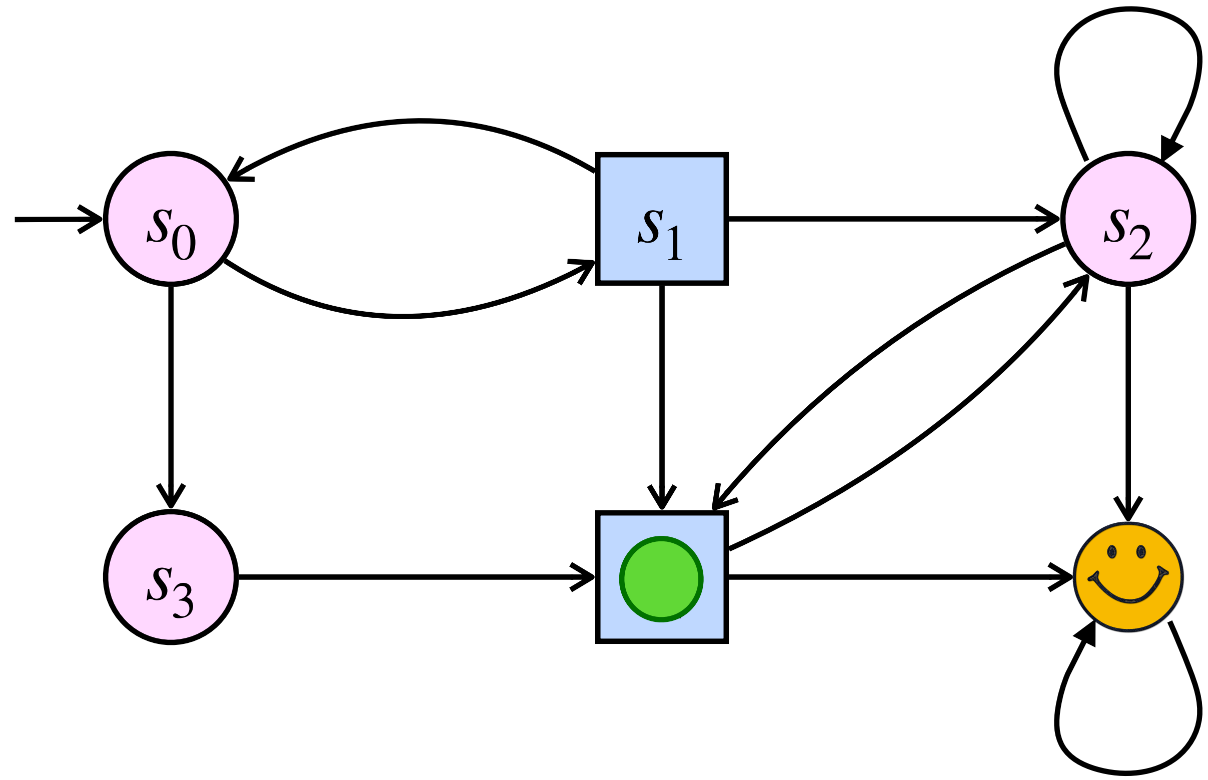
Games on graphs

States → S, Edges → E

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

○ : player P_1

□ : player P_2



$$s_0 \to s_1 \to s_4$$

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

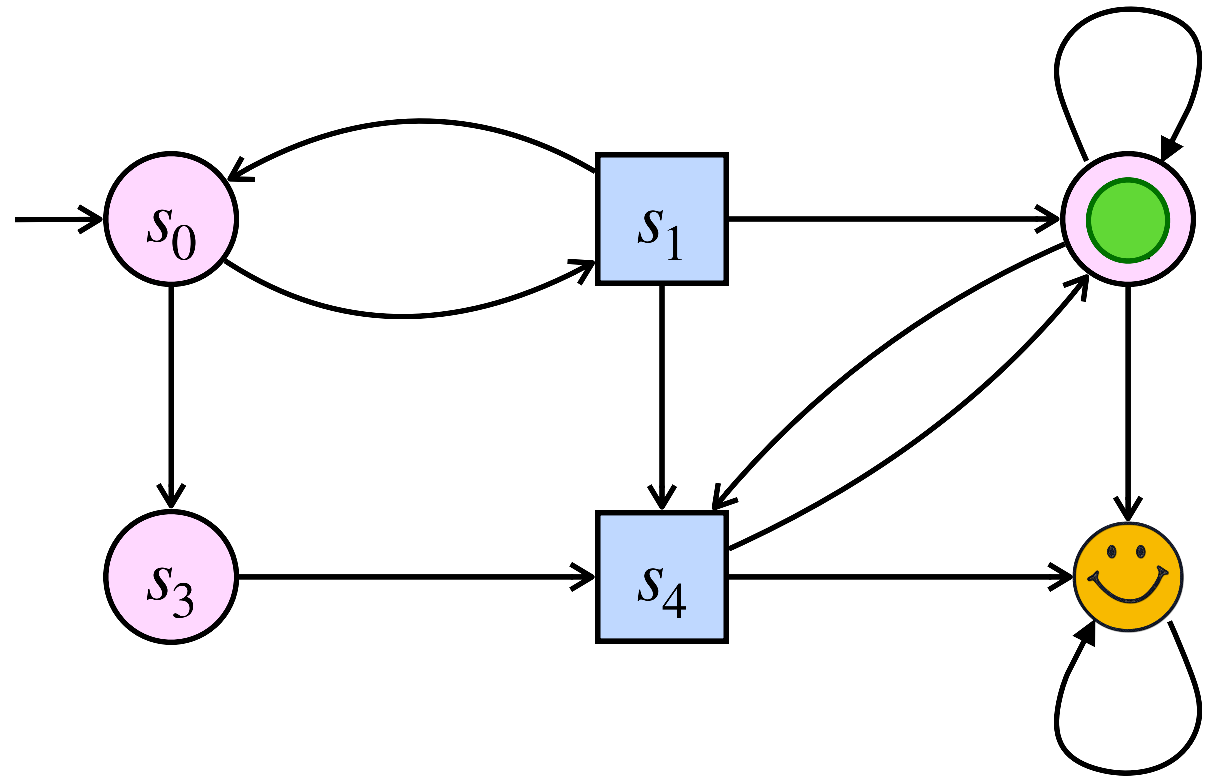
Games on graphs

States → S, Edges → E

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

○ : player P_1

□ : player P_2



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

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)

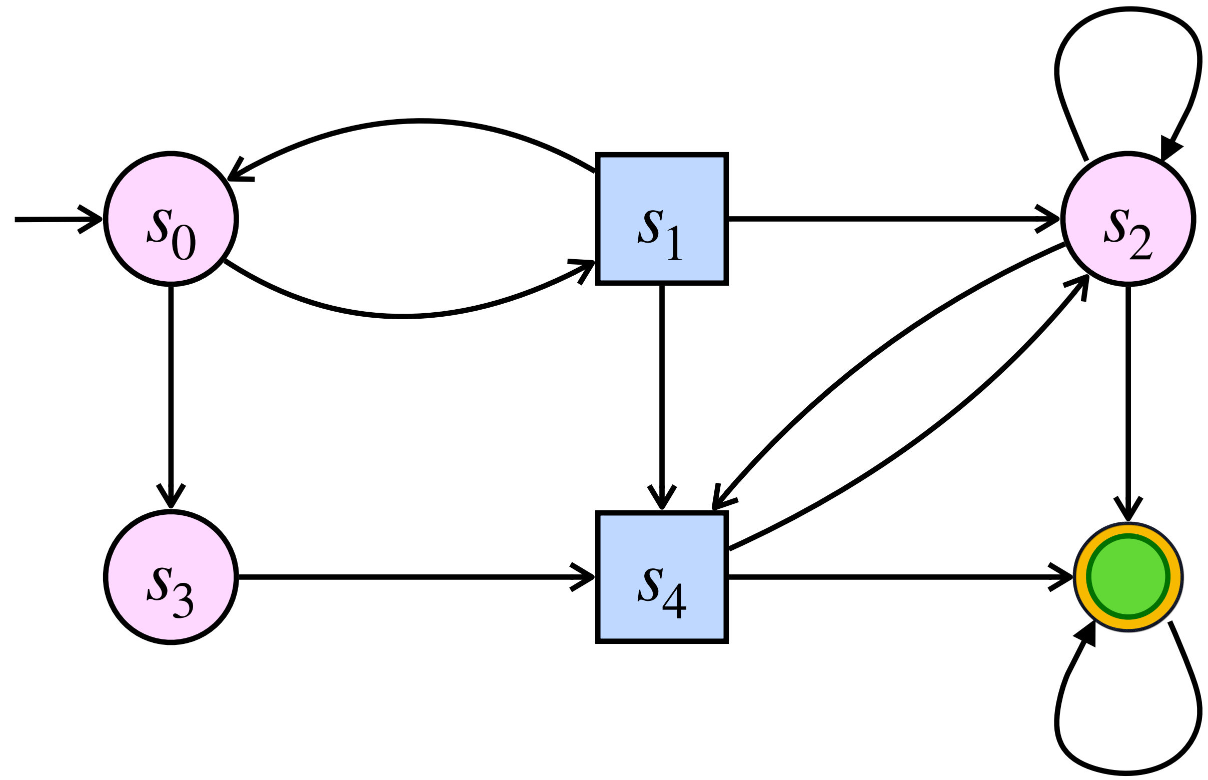
Games on graphs

States — Edges

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

○ : player P_1

□ : player P_2



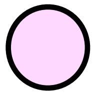
$s_0 \to s_1 \to s_4 \to s_2 \to$ 

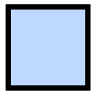
1. P_1 chooses the edge (s_0, s_1)
2. P_2 chooses the edge (s_1, s_4)
3. P_2 chooses the edge (s_4, s_2)
4. P_1 chooses the edge (s_2, )

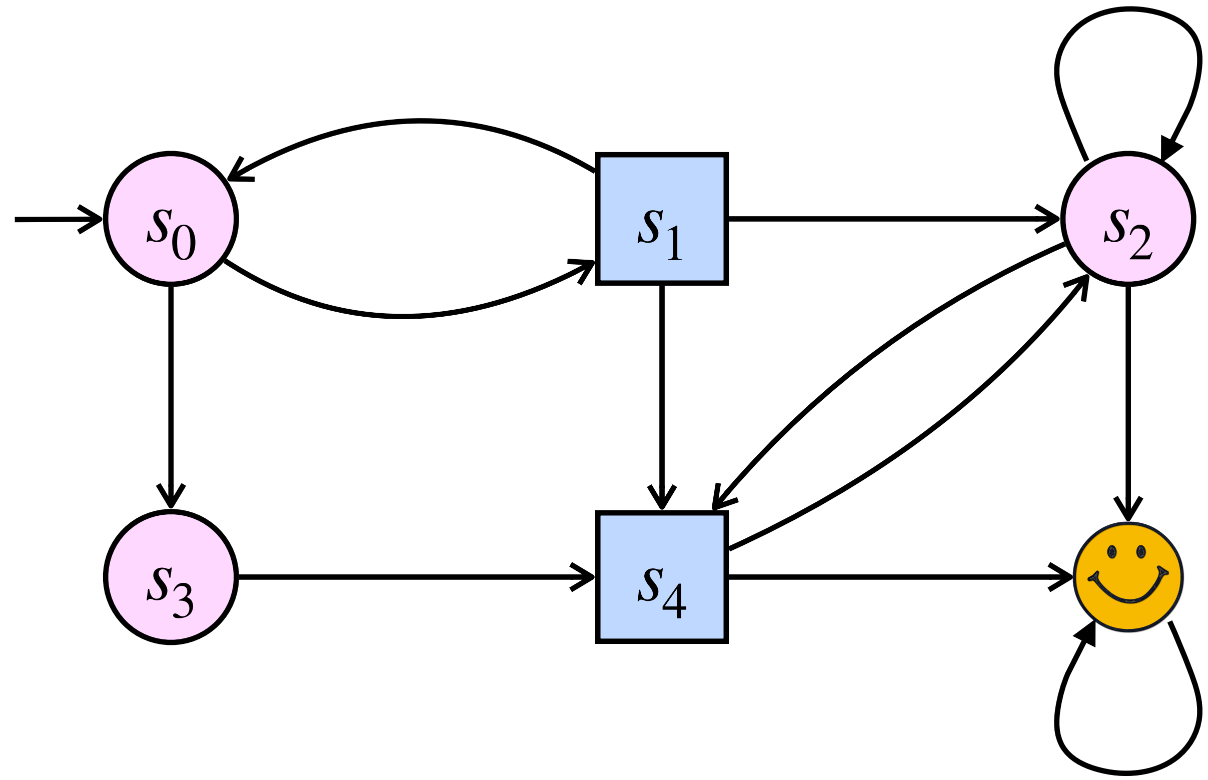
Games on graphs

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, )

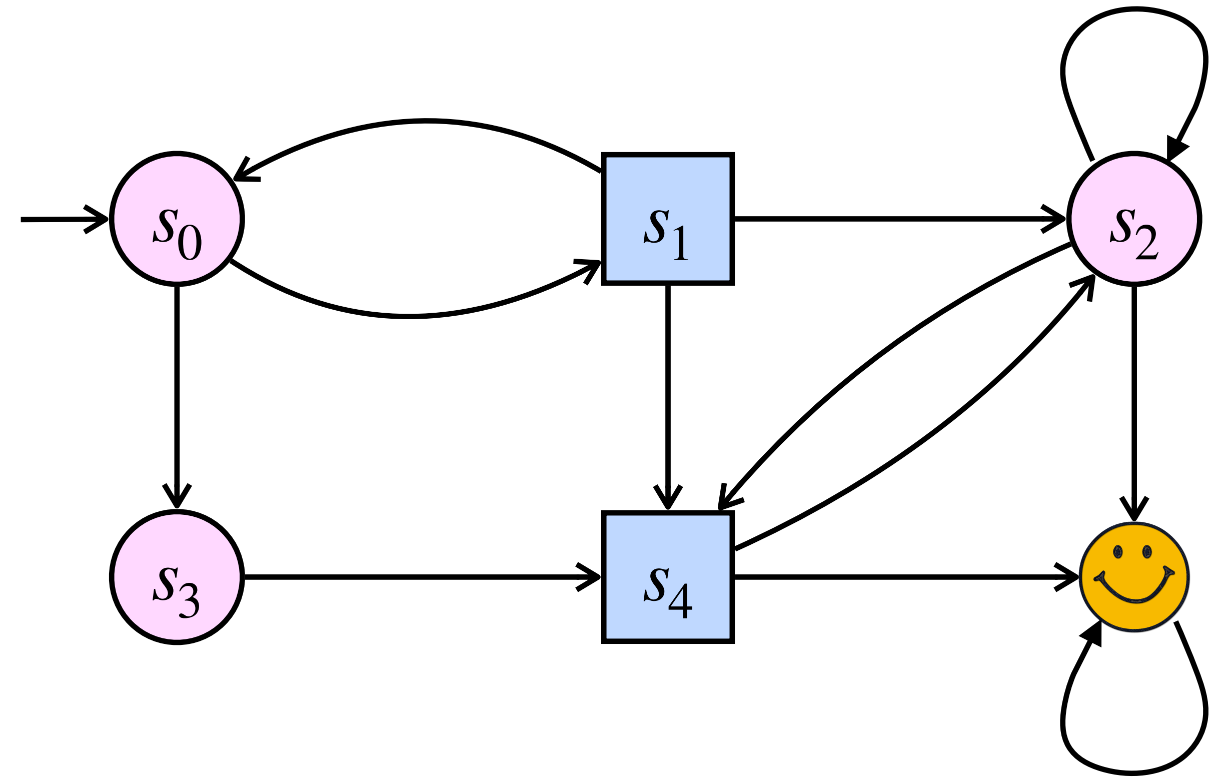
Games on graphs

States, Edges

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

○ : player P_1

□ : player P_2

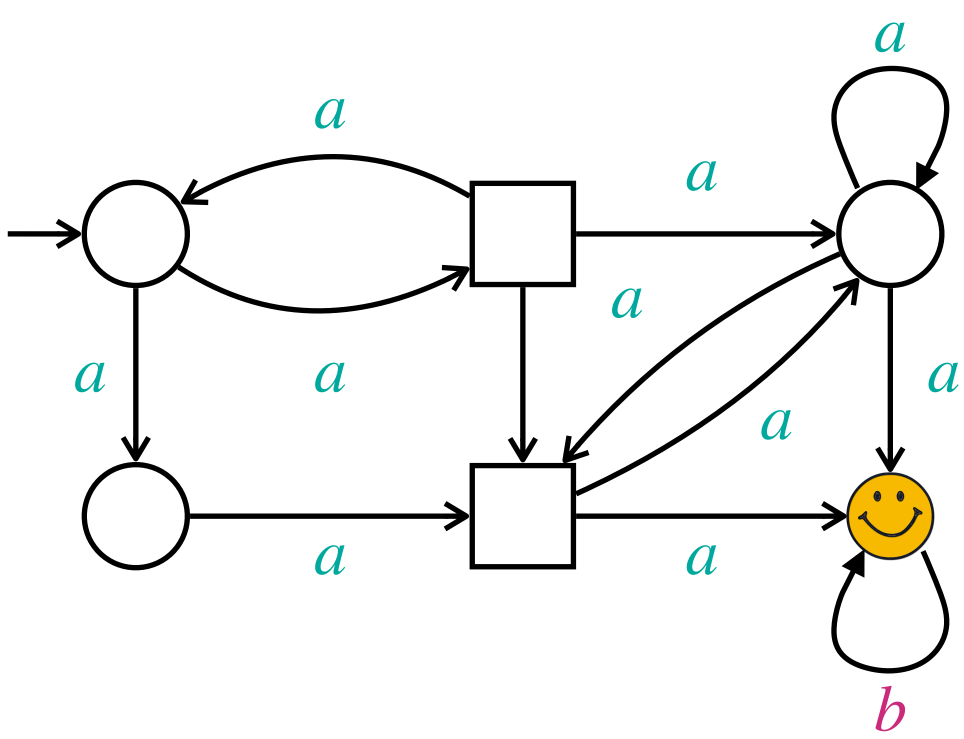


$$s_0 \rightarrow s_1 \rightarrow s_4 \rightarrow s_2 \rightarrow$$ ☺

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, ☺)

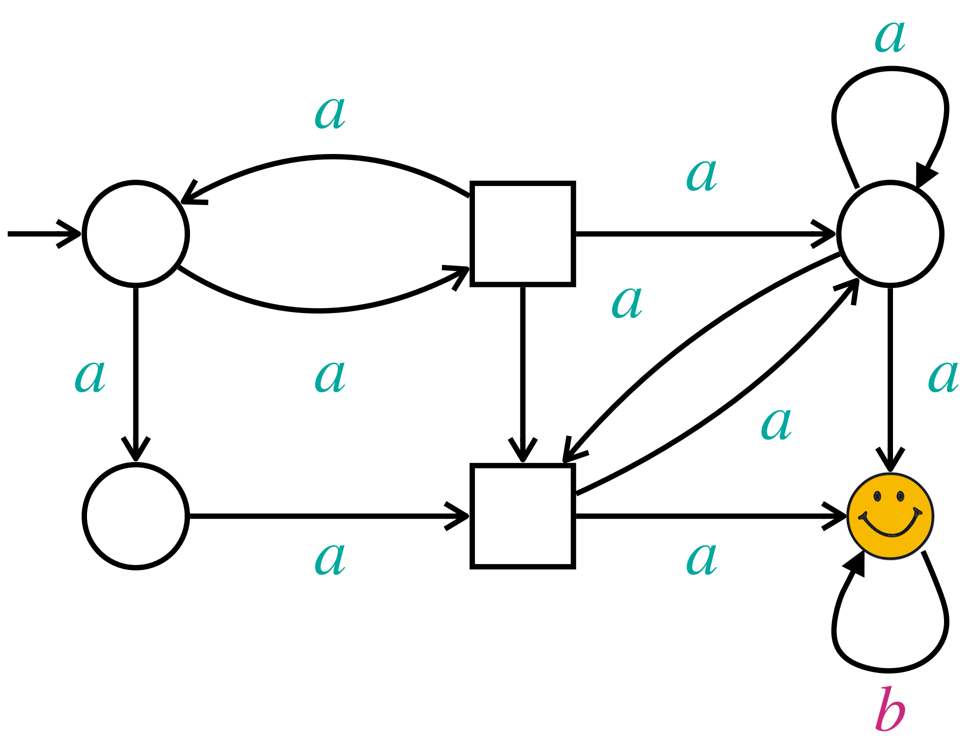
Players use **strategies** to play.
A strategy for P_i is $\sigma_i : S^*S_i \rightarrow E$

Objectives for the players



$$C = \{a, b\}$$
$$E \subseteq S \times C \times S$$

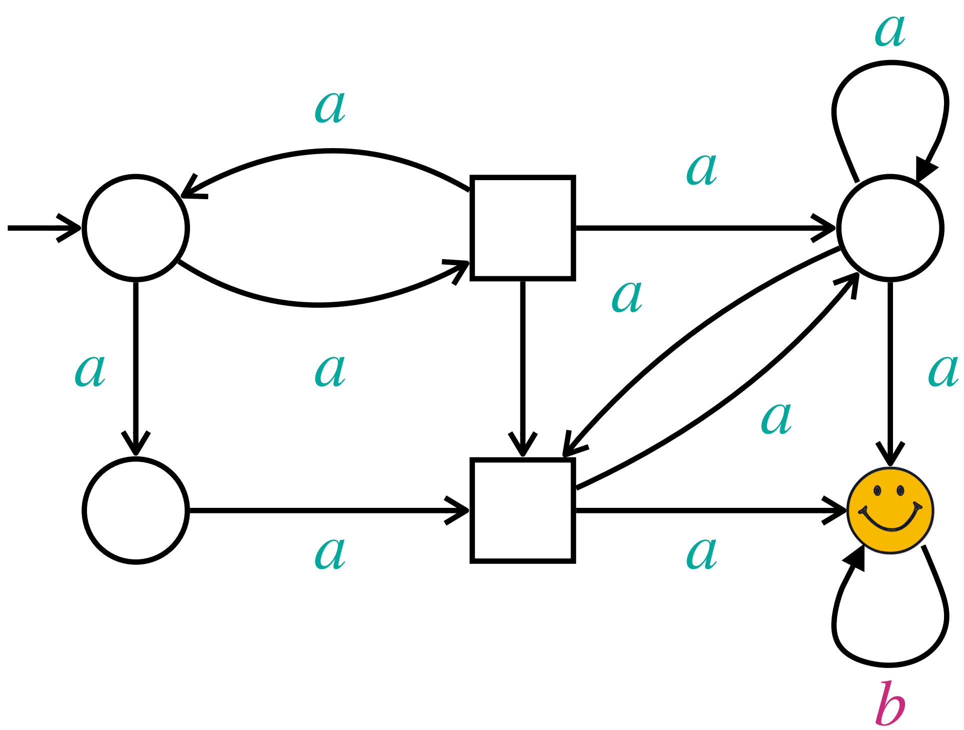
Objectives for the players



$C = \{a, b\}$
$E \subseteq S \times C \times S$

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

Objectives for the players

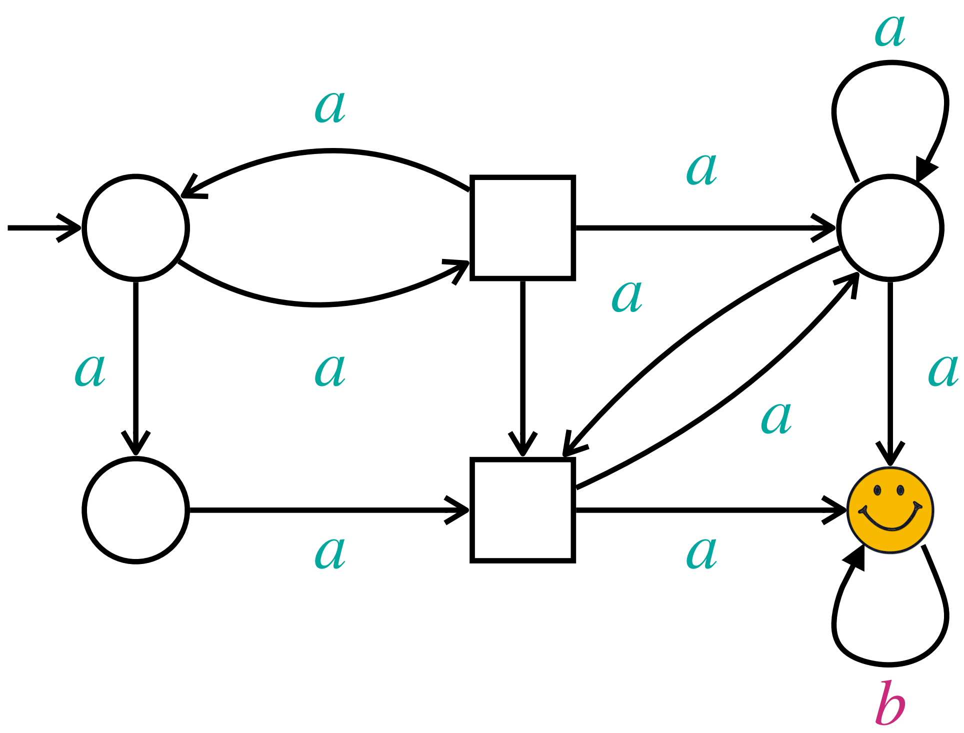


$C = \{a, b\}$
$E \subseteq S \times C \times S$

- 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

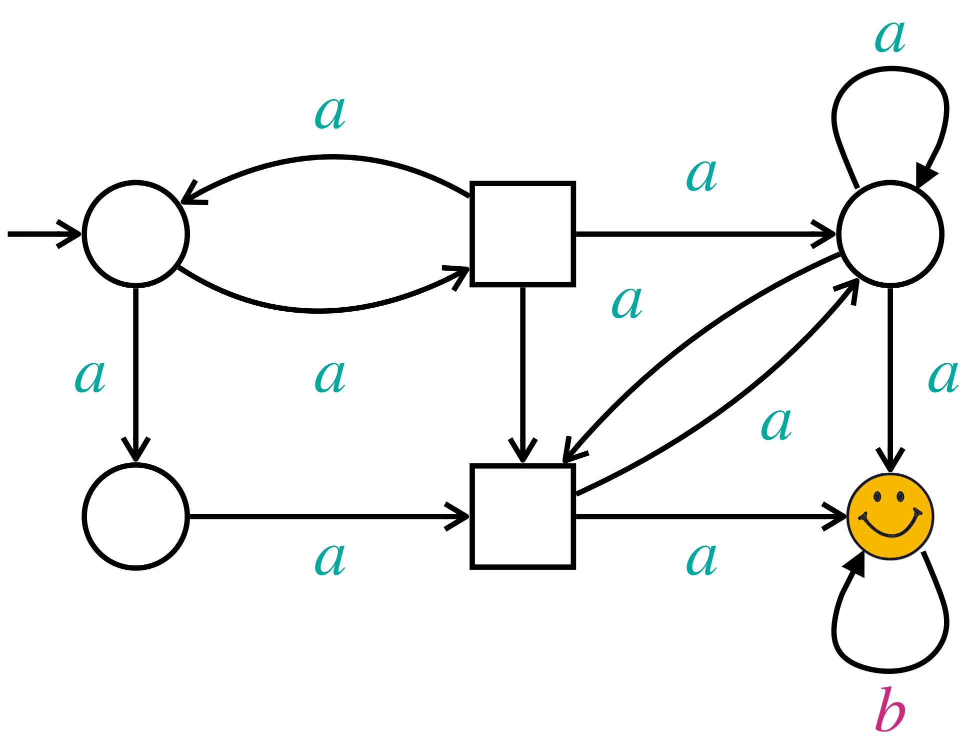
Objectives for the players



$C = \{a, b\}$
$E \subseteq S \times C \times S$

- 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 hypothesis

$C = \{a, b\}$
$E \subseteq S \times C \times S$

- 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)

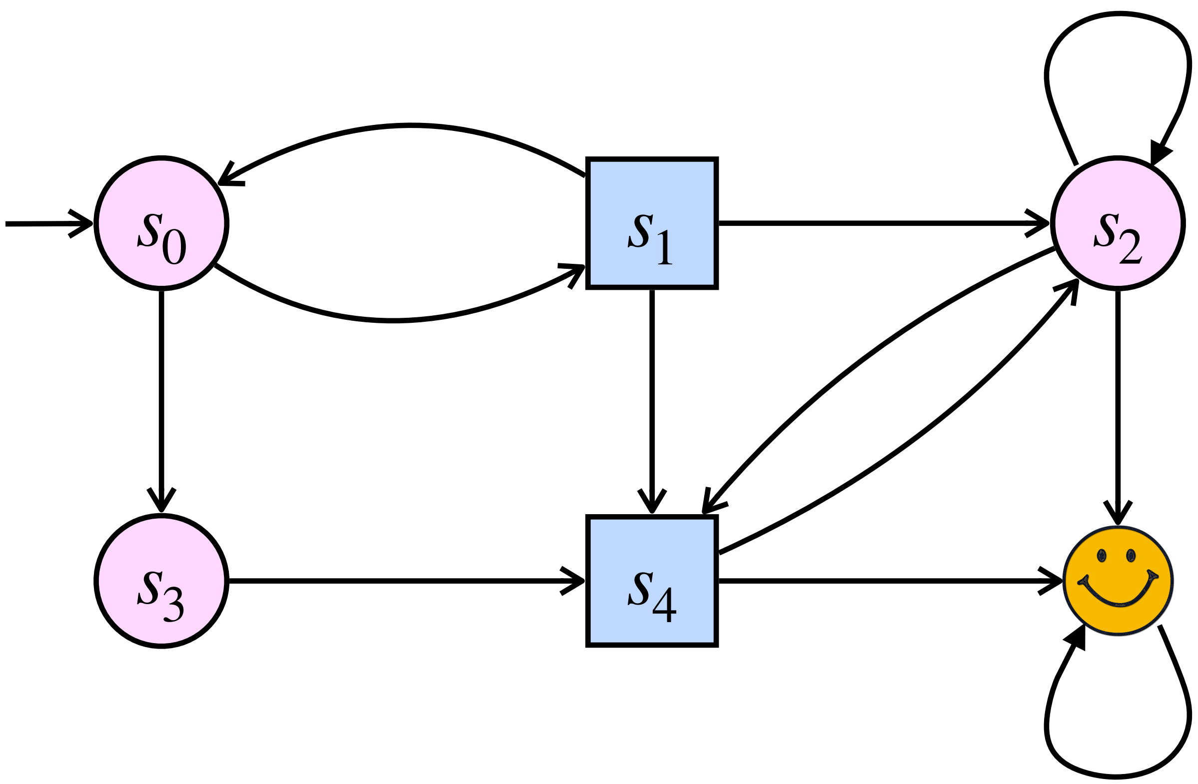
$\sqsubseteq_2 = \sqsubseteq_1^{-1}$

What does it mean to win a game?

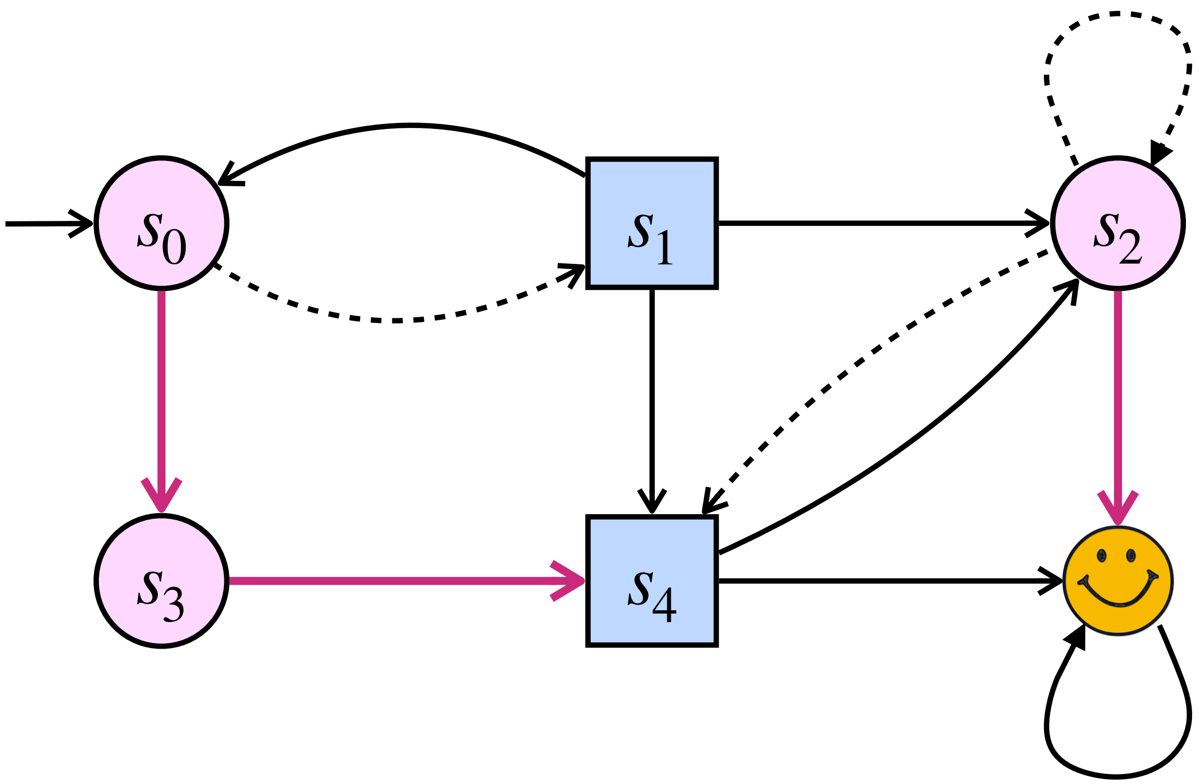
What does it mean to win a game?

- 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

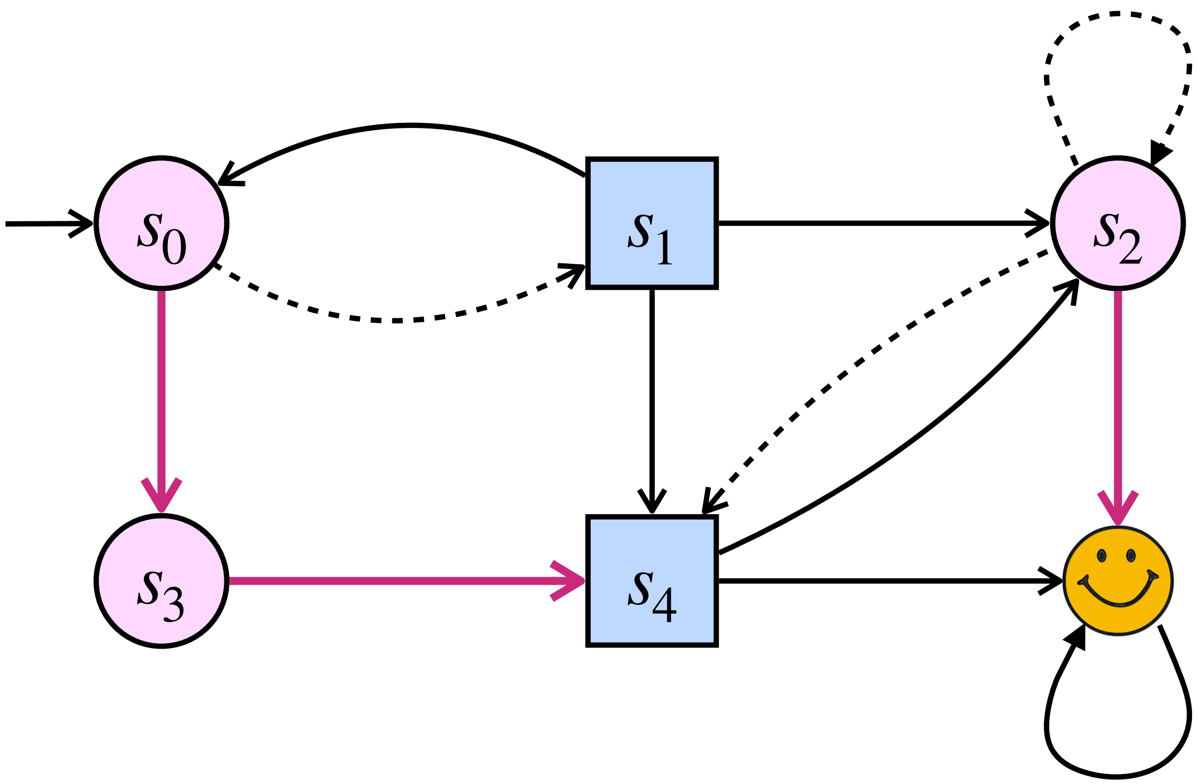


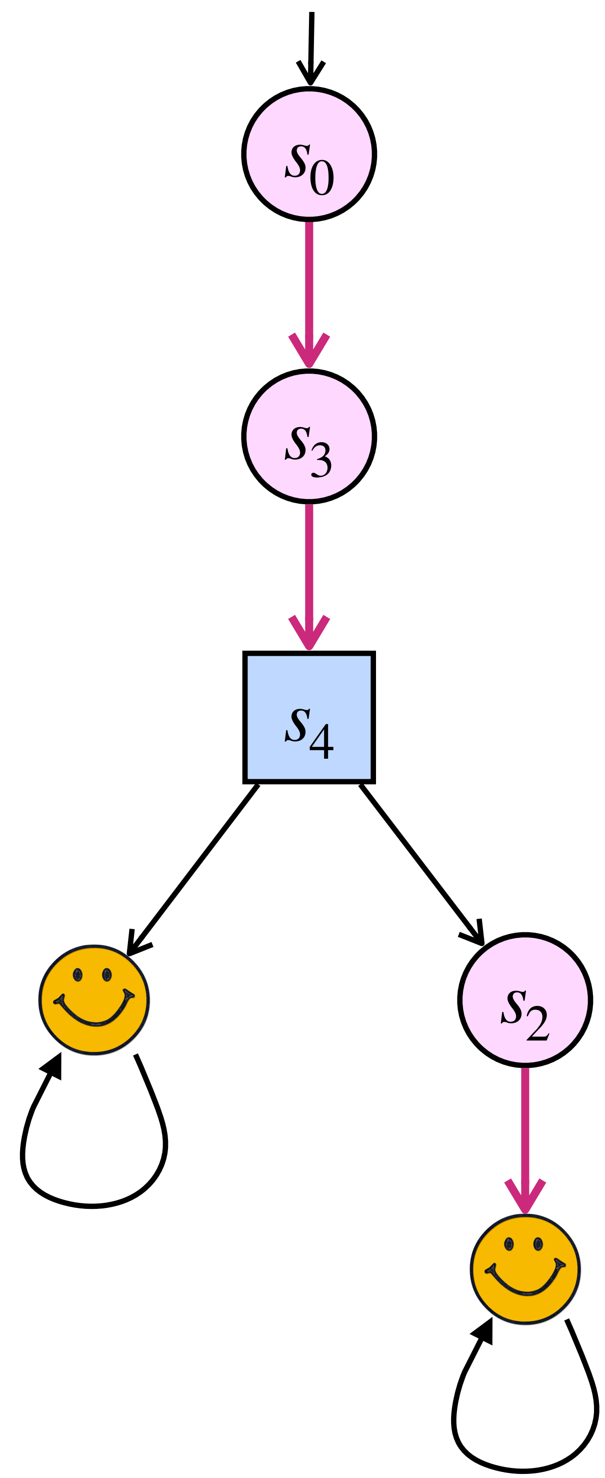
Outcomes of a strategy



- Strategy σ

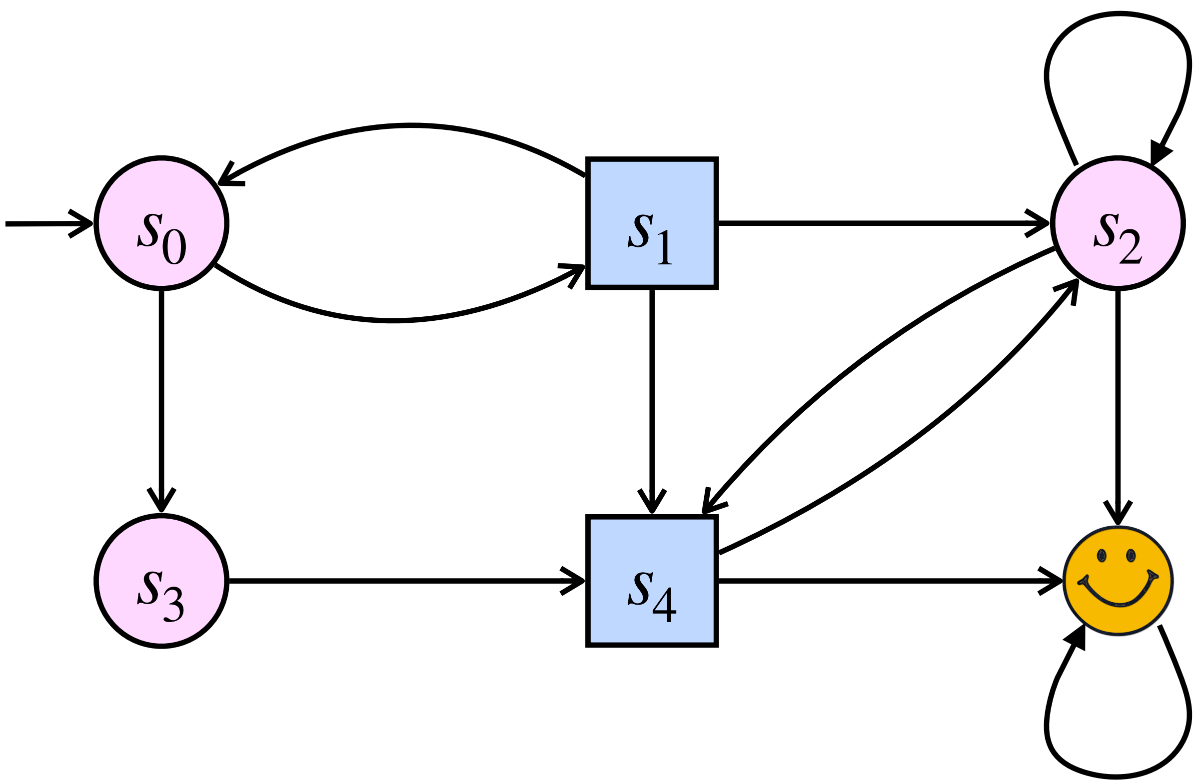
Outcomes of a strategy



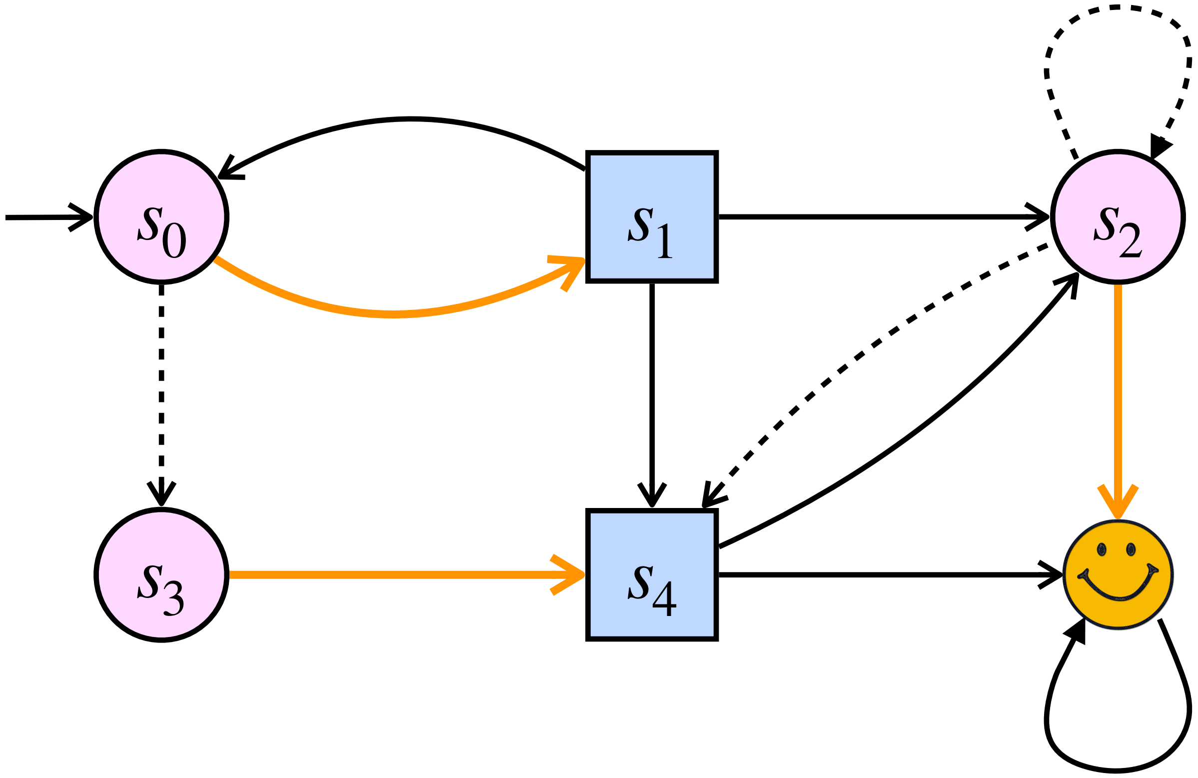


- Strategy σ
- Out(σ) has two plays, which are both winning

Outcomes of a strategy

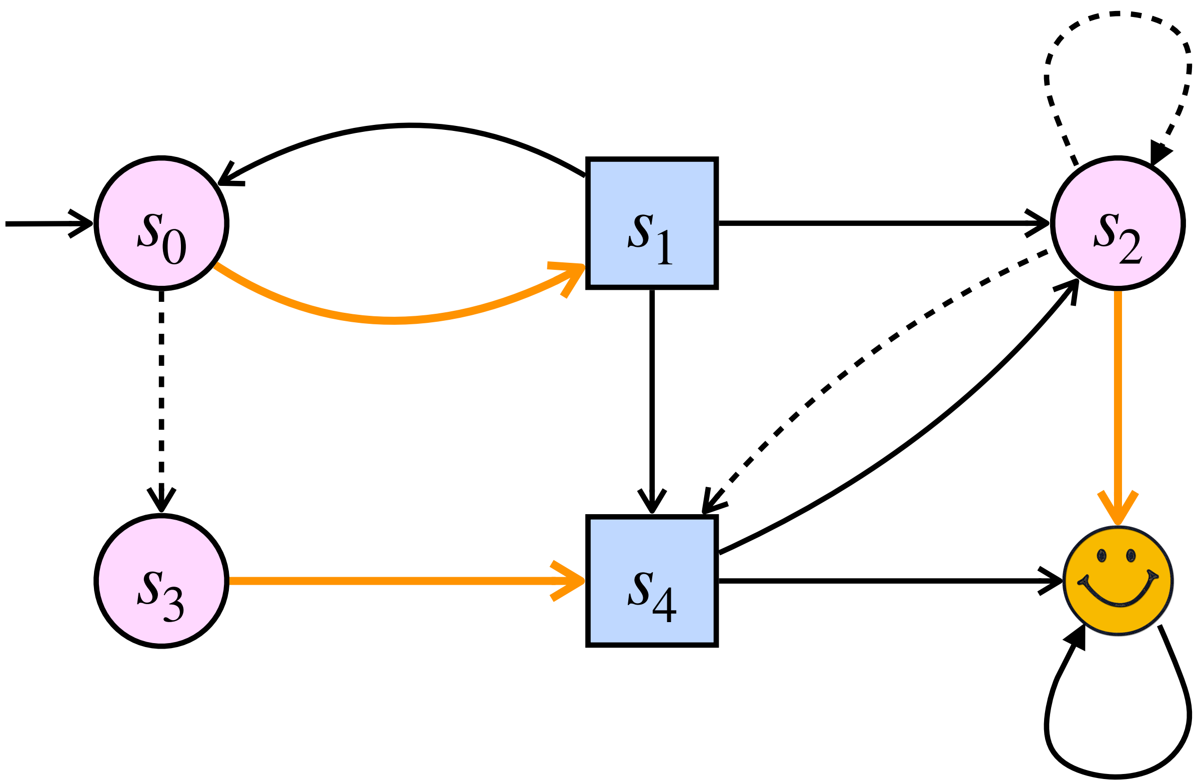


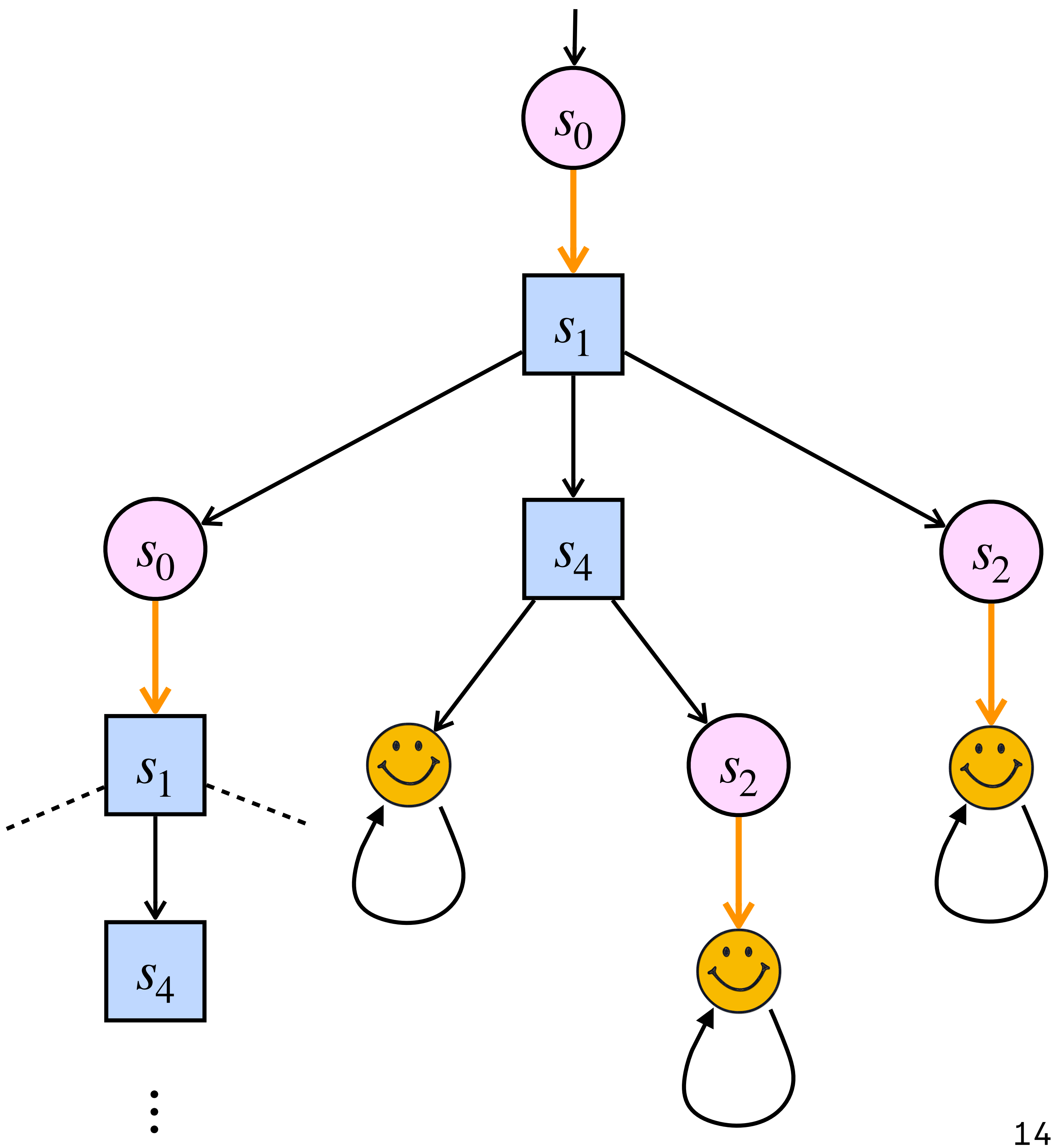
Outcomes of a strategy



- Strategy σ

Outcomes of a strategy





- Strategy σ
- Out(σ) 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

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

What does it mean to win a game?

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

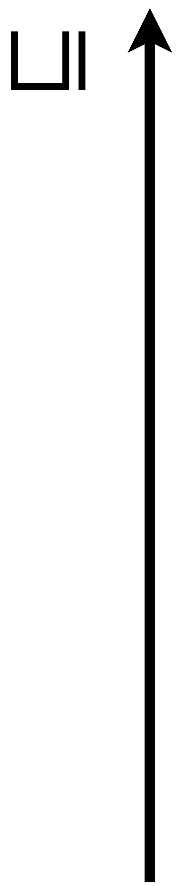
- σ_i is **winning** if all plays compatible with σ_i belong to W_i

Martin's determinacy theorem

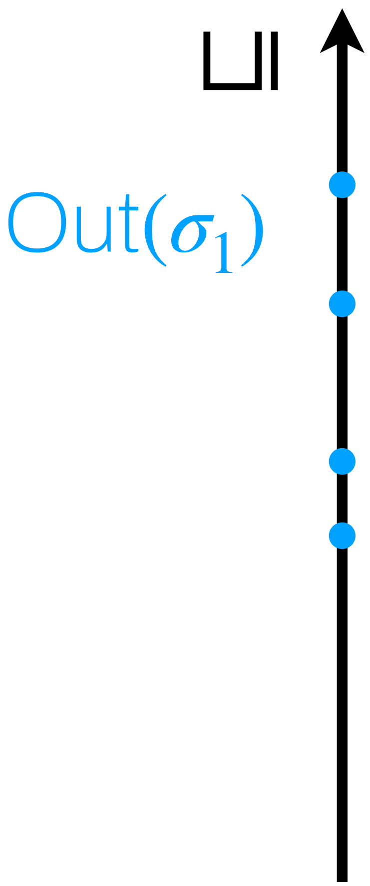
Turn-based zero-sum games are determined for Borel winning objectives: in every game, either P_1 or P_2 has a winning strategy.

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

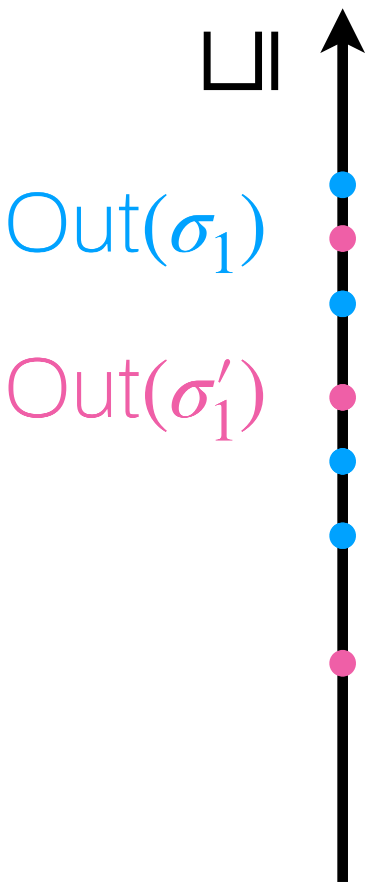
Optimality of strategies



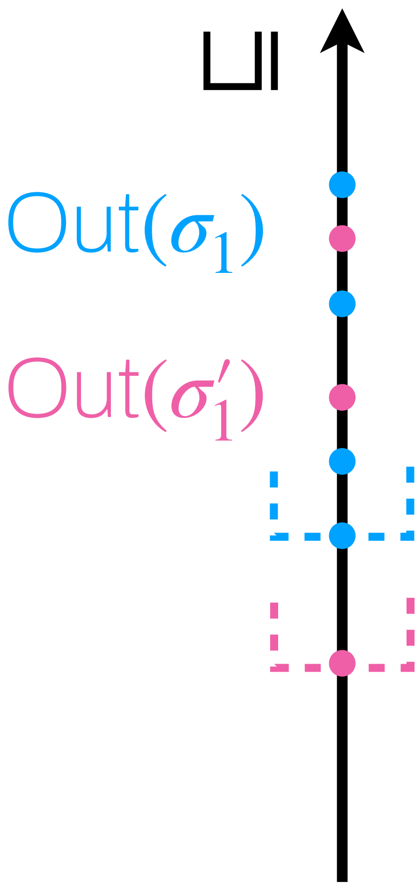
Optimality of strategies



Optimality of strategies

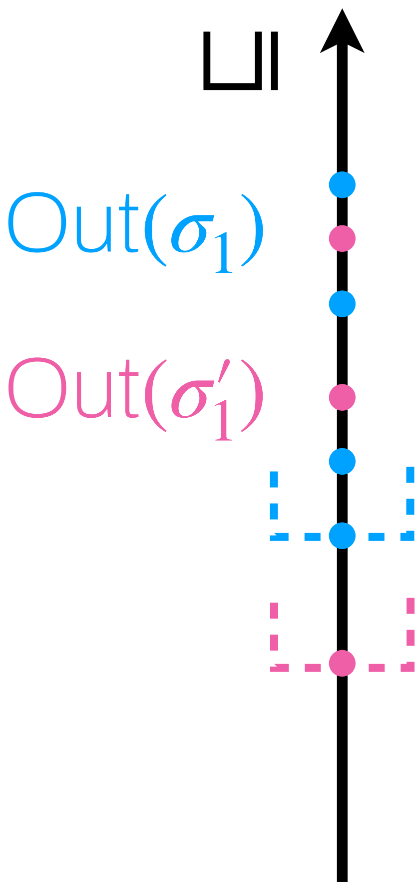


Optimality of strategies



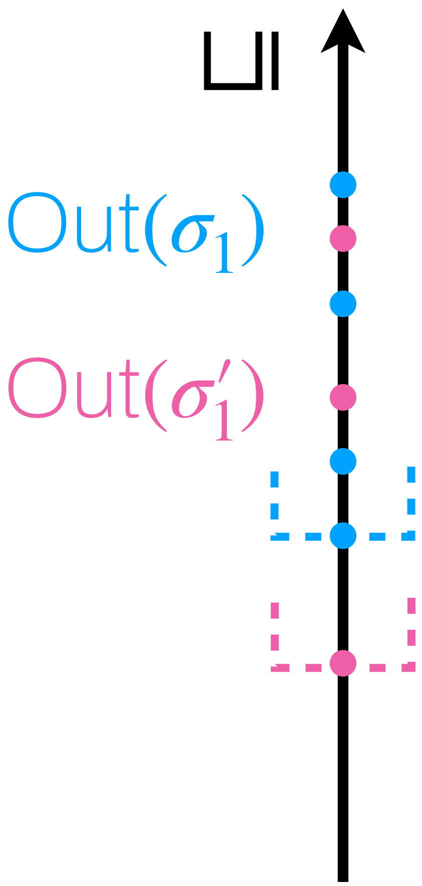
- σ_1 is better than σ_1' whenever $\mathrm{Out}(\sigma_1)^{\uparrow} \subseteq \mathrm{Out}(\sigma_1')^{\uparrow}$

Optimality of strategies



- σ_1 is better than σ_1' whenever $\mathrm{Out}(\sigma_1)^{\uparrow} \subseteq \mathrm{Out}(\sigma_1')^{\uparrow}$
- σ_1 is **optimal** whenever it is better than any other σ_1'

Optimality of strategies

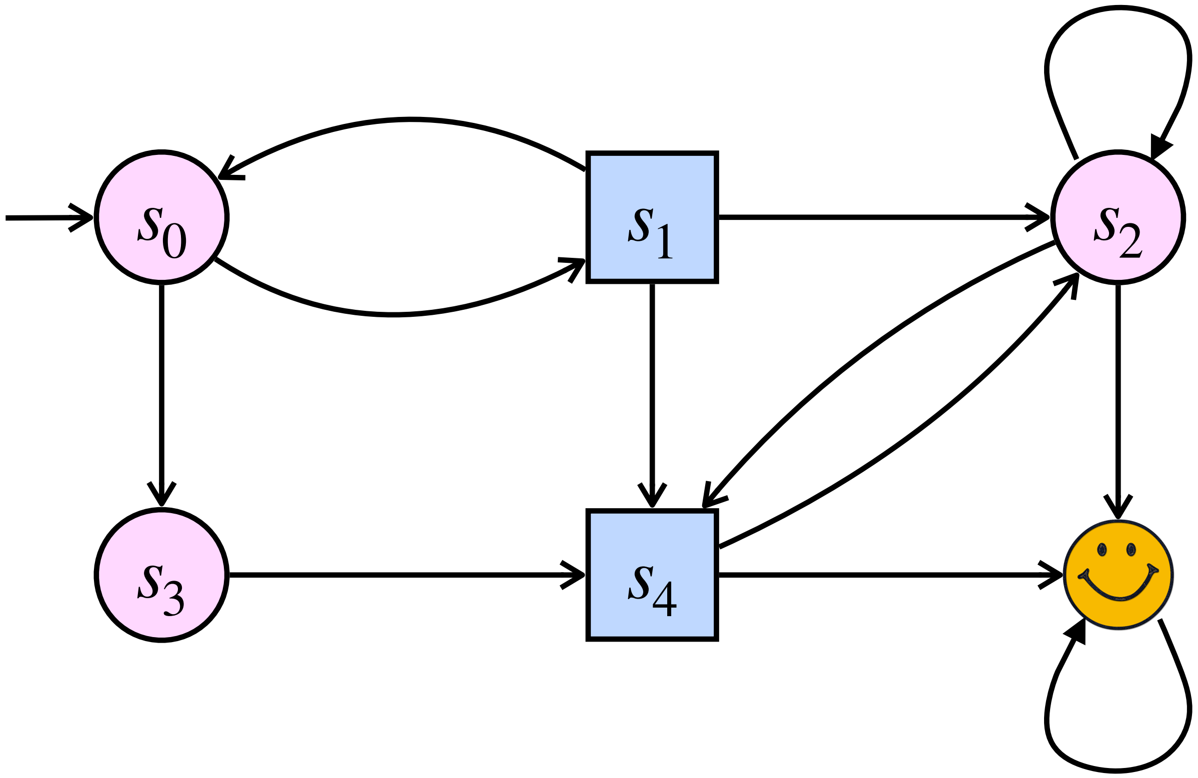


- σ_1 is better than σ_1' whenever Out$(\sigma_1)^{\uparrow} \subseteq$ Out$(\sigma_1')^{\uparrow}$
- σ_1 is **optimal** whenever it is better than any other σ_1'

Remark

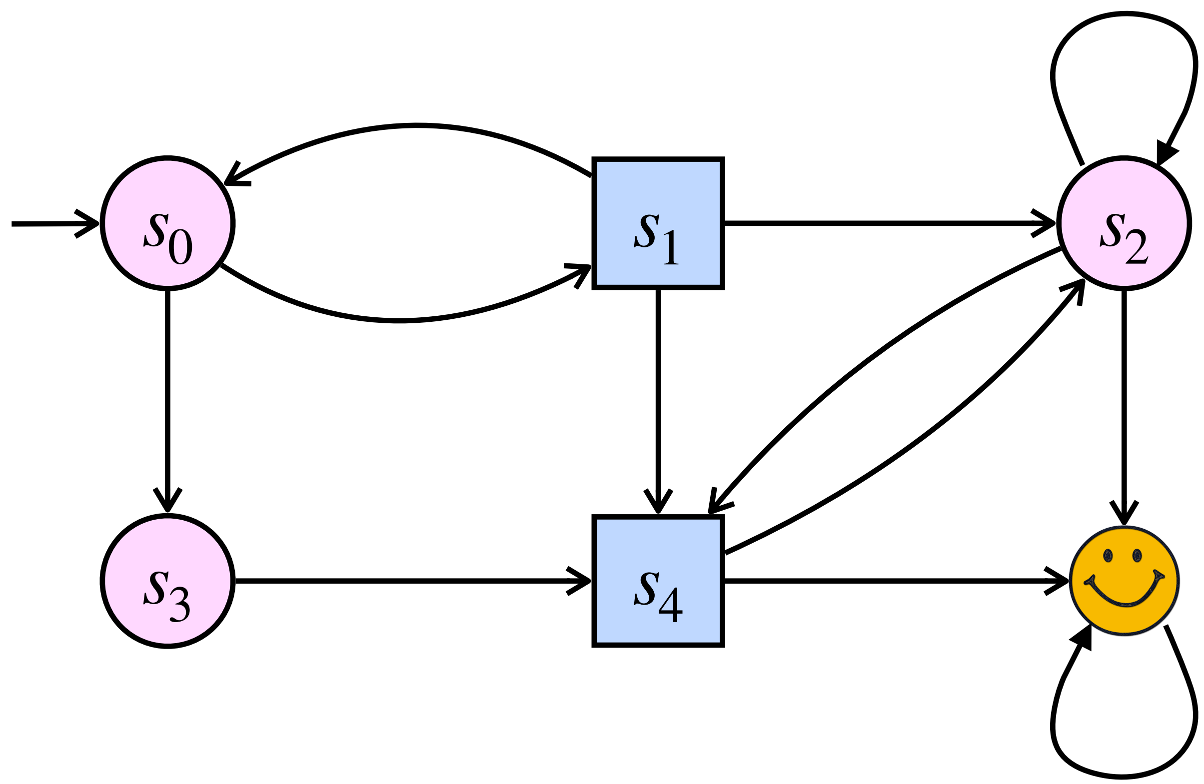
- Optimal strategies might not exist
- If $\sqsubseteq$ given by a payoff function, notion of ε-optimal strategies
- Optimality vs subgame-optimality

Relevant questions



φ = Reach(🙂)

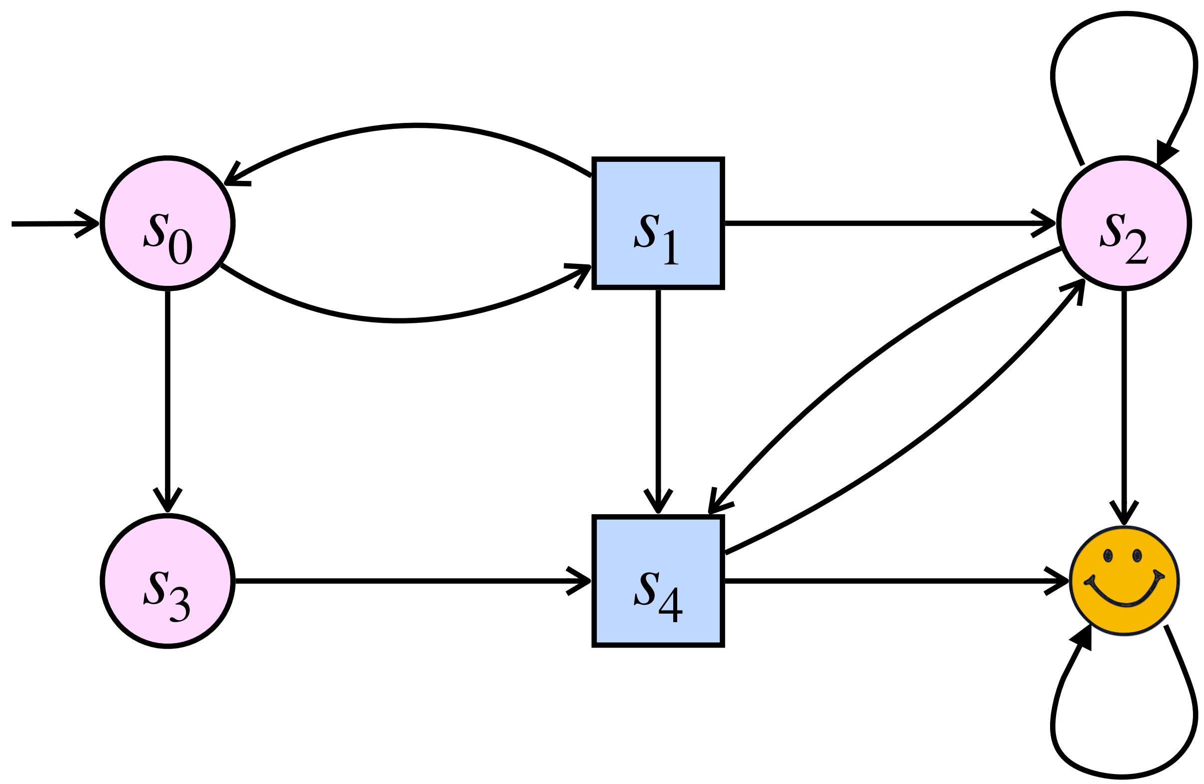
Relevant questions



φ = Reach(☺)

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

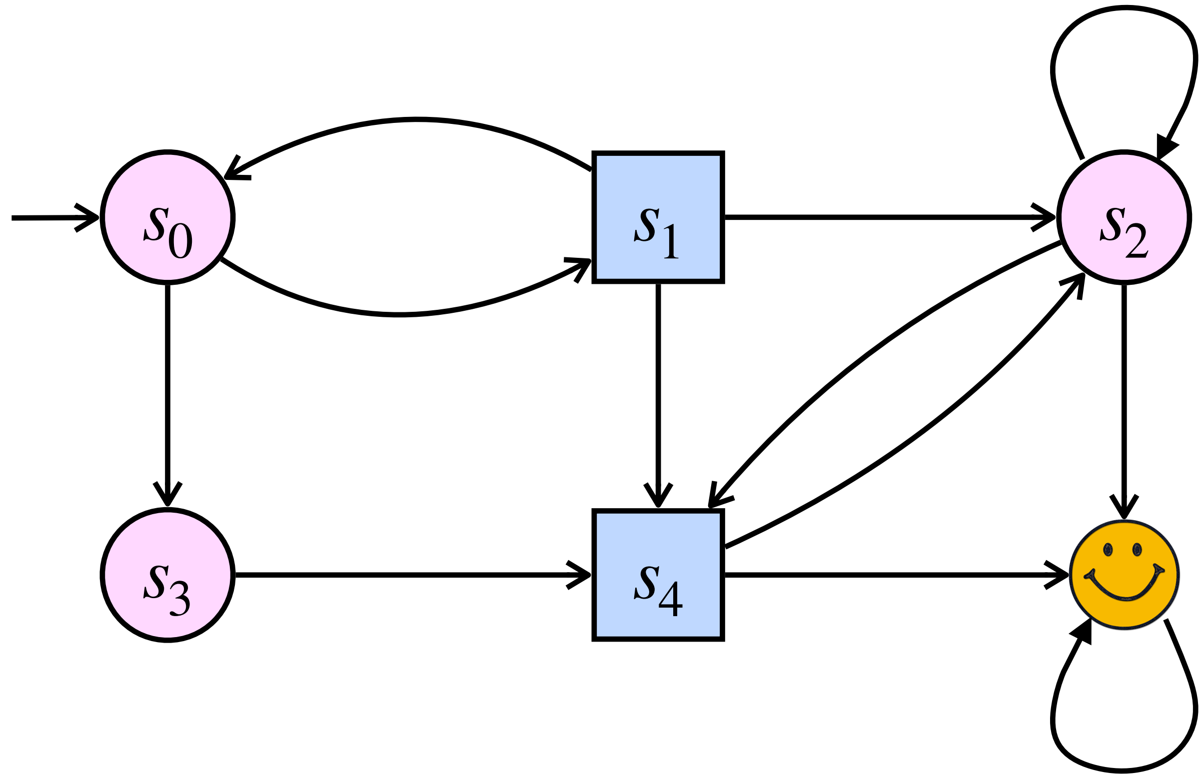
Relevant questions



φ = Reach(☺)

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

Relevant questions



$\varphi =$ Reach(☺)

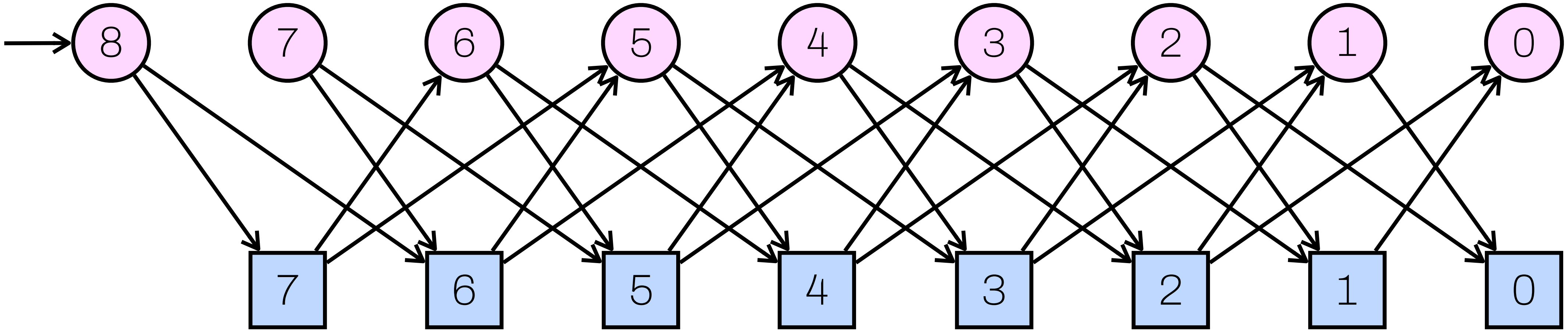
- Can P_1 win the game, i.e. does P_1 have a winning strategy?
  Can P_1 play optimally?
- Is there an effective (efficient) way of winning?
- How complex is it to win?

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

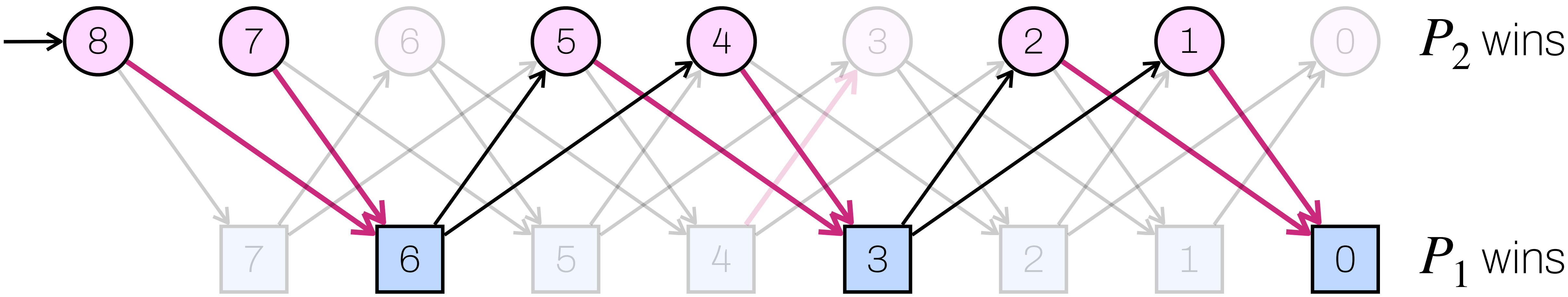
Example: the Nim game

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



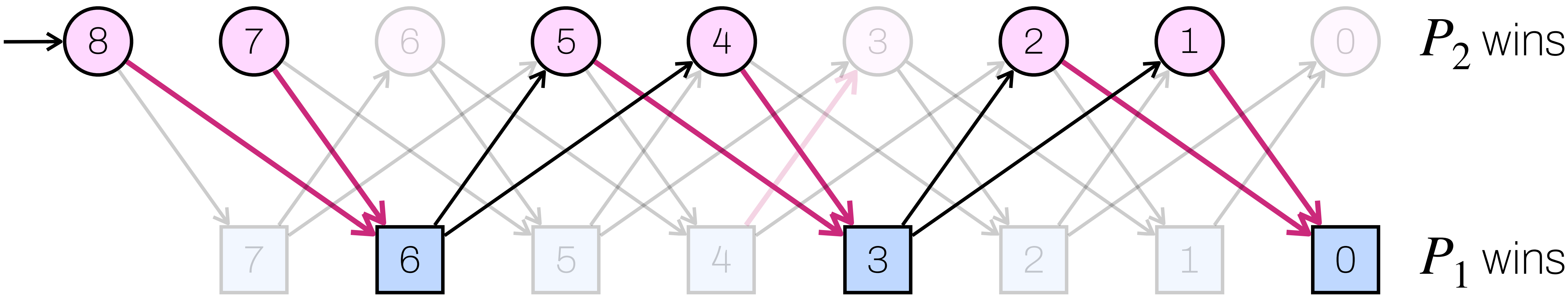
Example: the Nim game

- 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

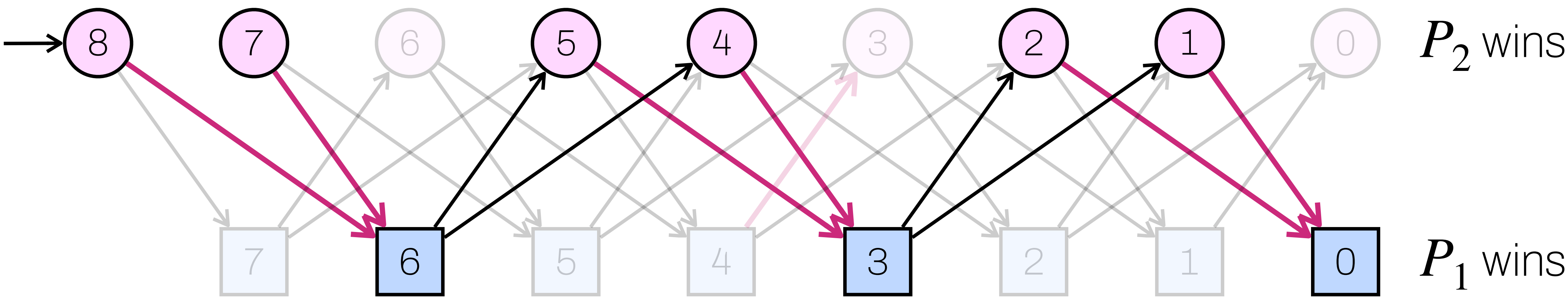


P_1 wins

- 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



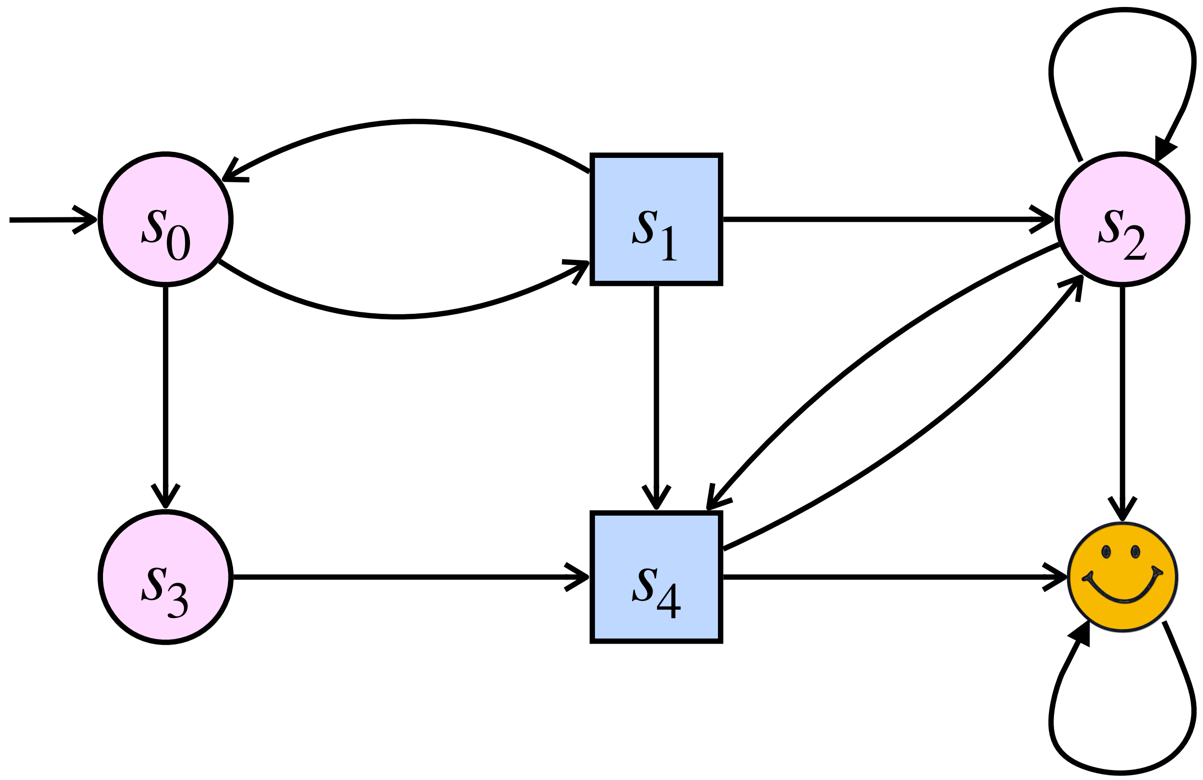
P_1 wins

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

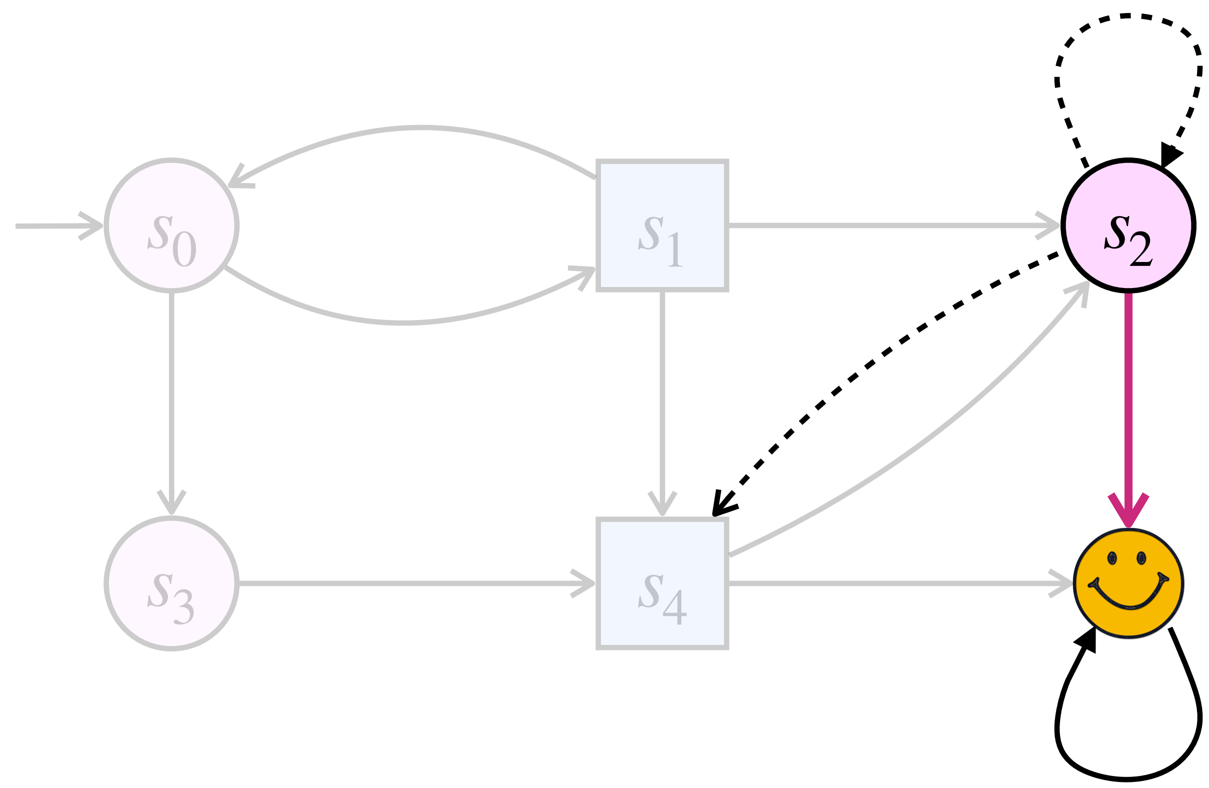
P_2 wins

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

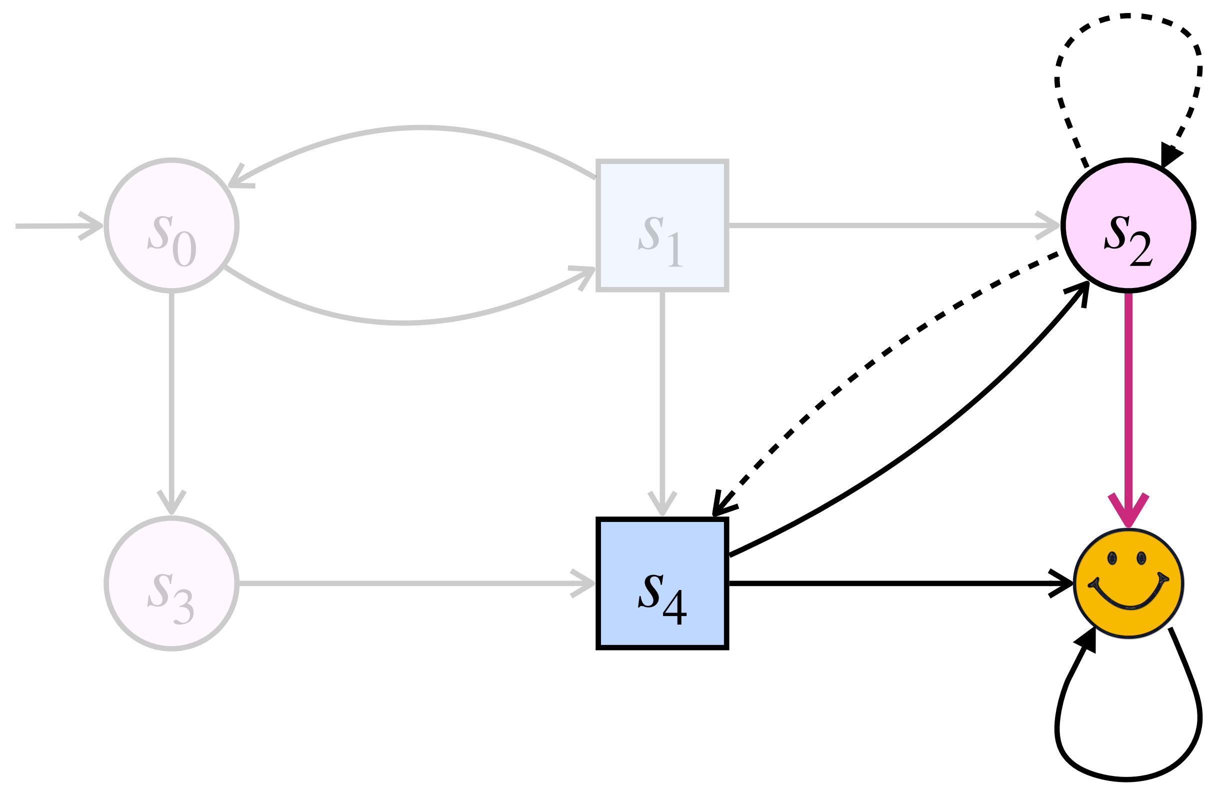
Computation of winning states in the running example



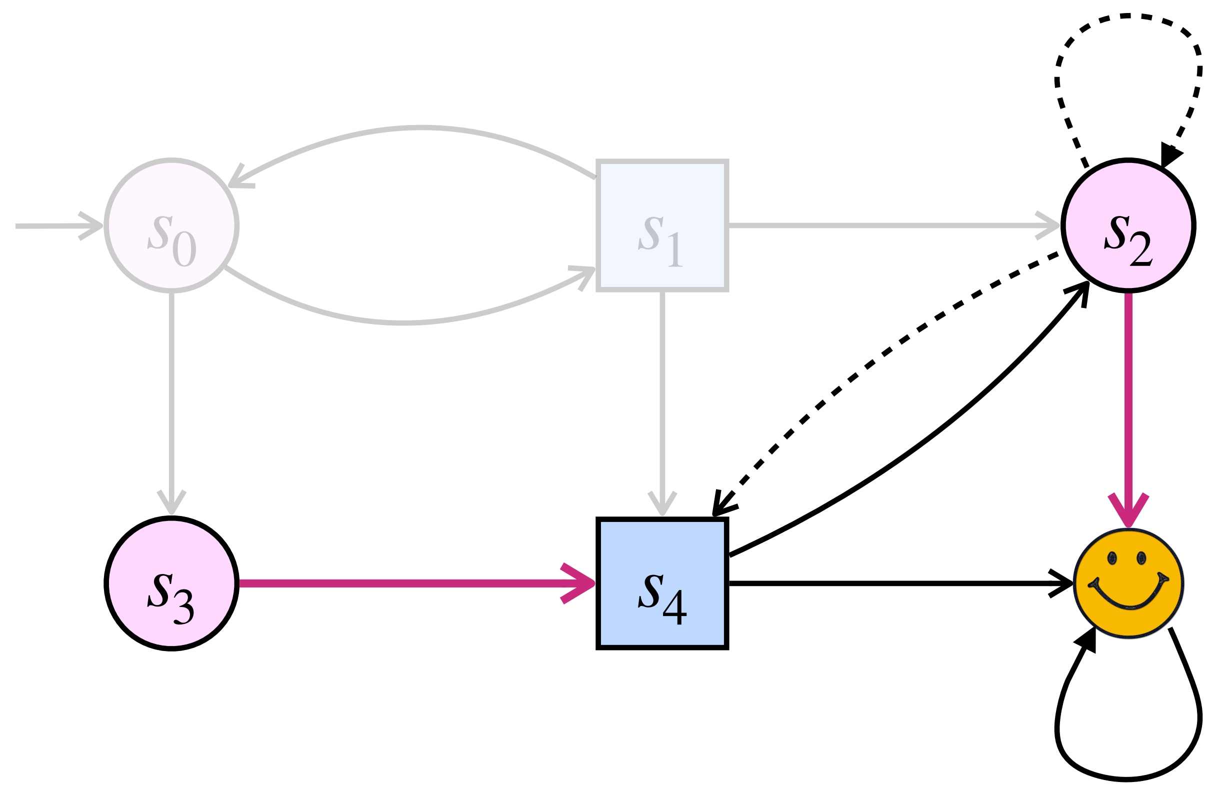
Computation of winning states in the running example



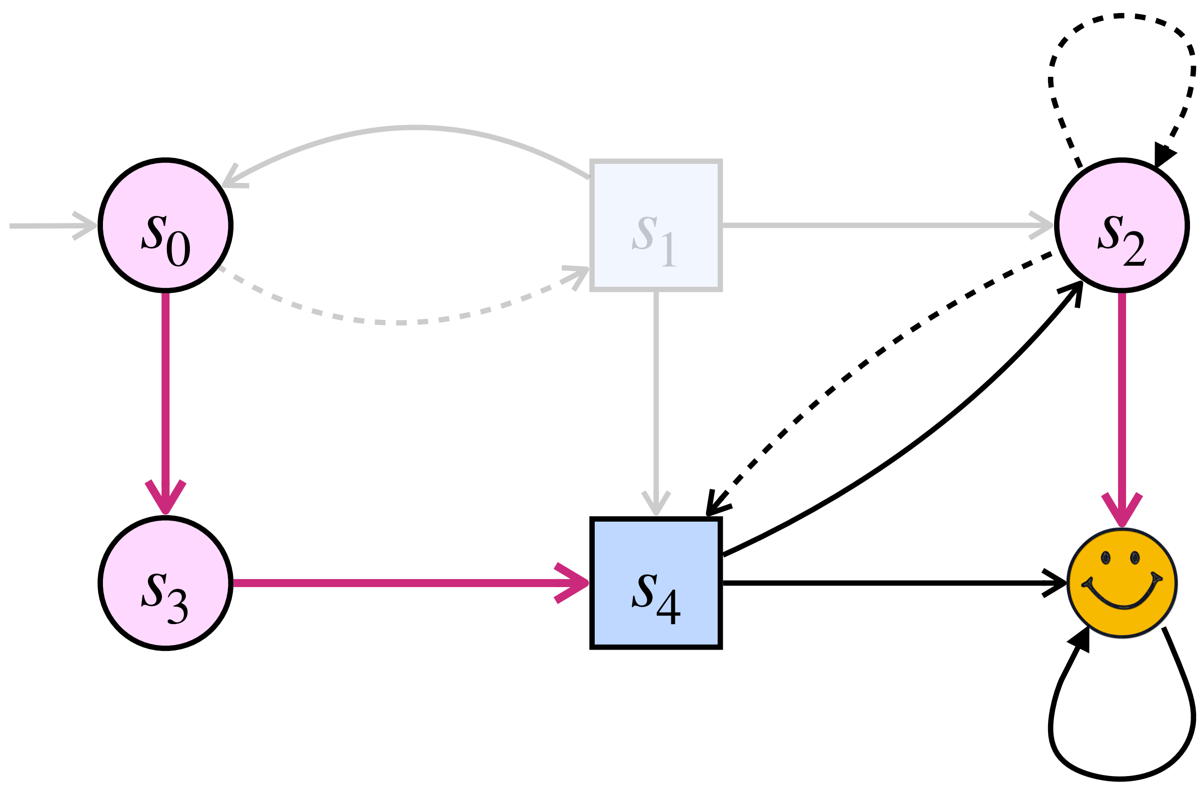
Computation of winning states in the running example



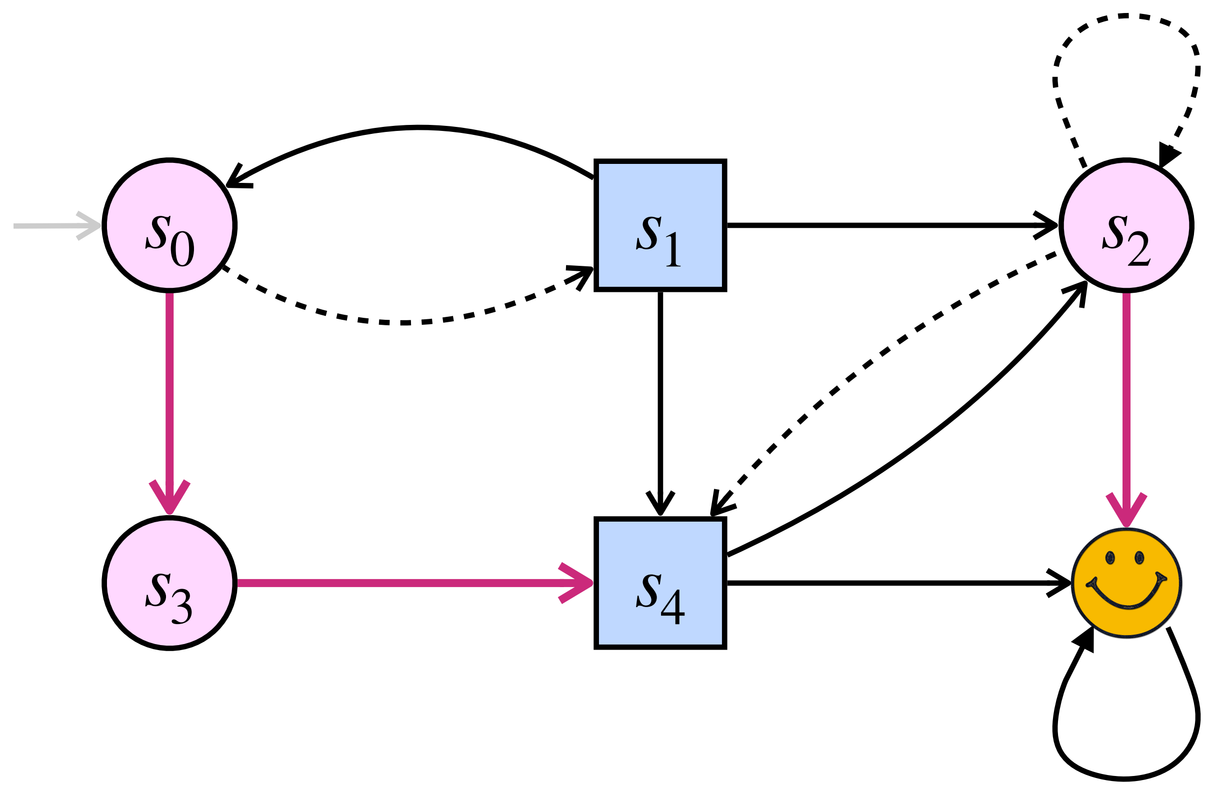
Computation of winning states in the running example



Computation of winning states in the running example

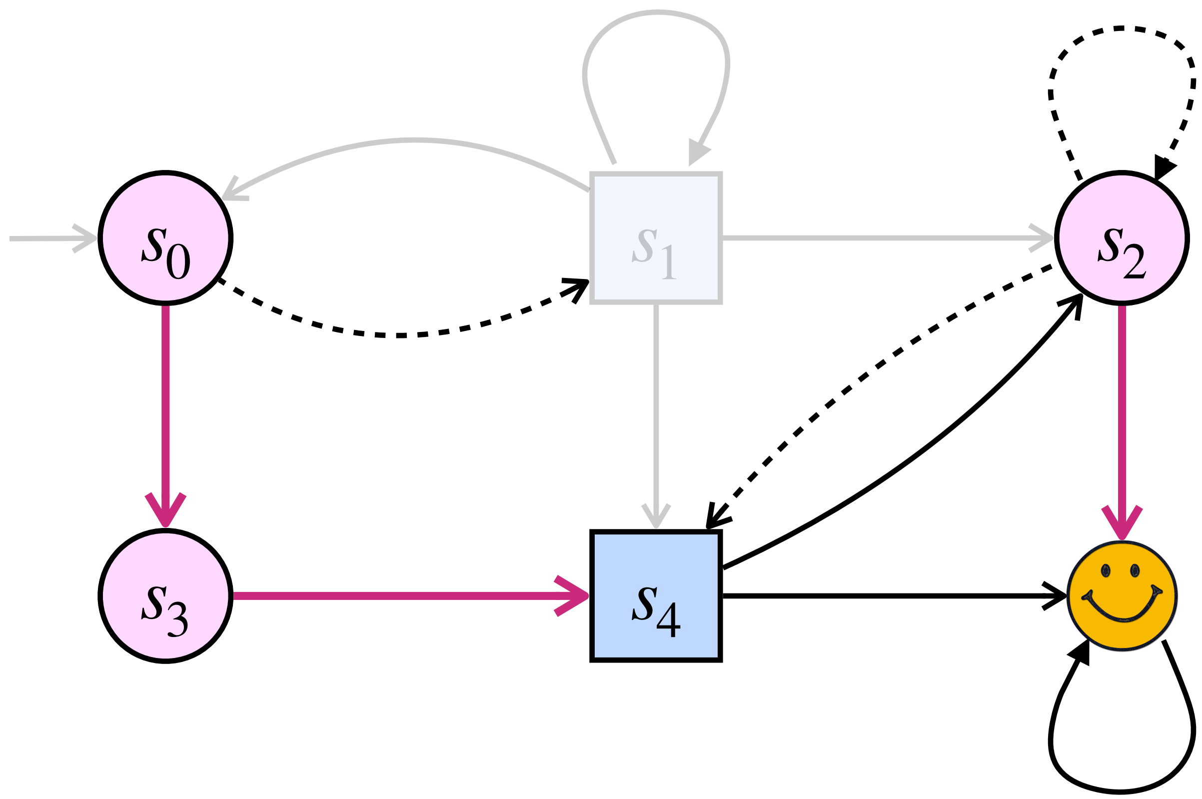


Computation of winning states in the running example



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

Chess game

[Zer13] Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels (Congress Mathematicians, 1912).
[Au89] Aumann. Lectures on Game Theory (1989).

Chess game

Zermelo's Theorem

From every position, either White can force a win, or Black can force a win, or both sides can force at least a draw.

[Zer13] Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels (Congress Mathematicians, 1912).
[Au89] Aumann. Lectures on Game Theory (1989).

Chess game

Zermelo's Theorem

From every position, either White can force a win, or Black can force a win, or both sides can force at least a draw.

- We don't know what is the case for the initial position, and no winning strategy (for either of the players) is known

[Zer13] Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels (Congress Mathematicians, 1912).
[Au89] Aumann. Lectures on Game Theory (1989).

Chess game

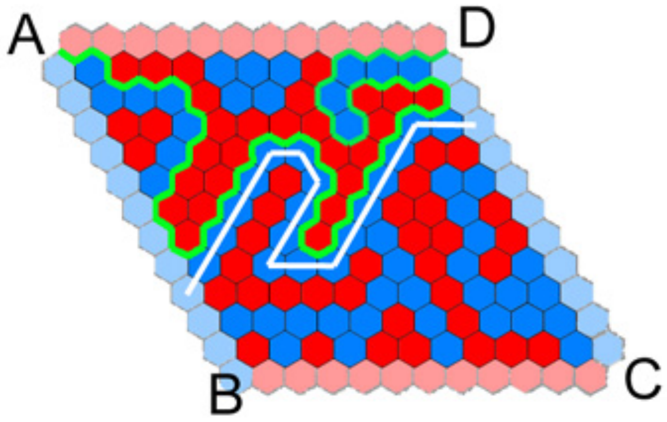
Zermelo's Theorem

From every position, either White can force a win, or Black can force a win, or both sides can force at least a draw.

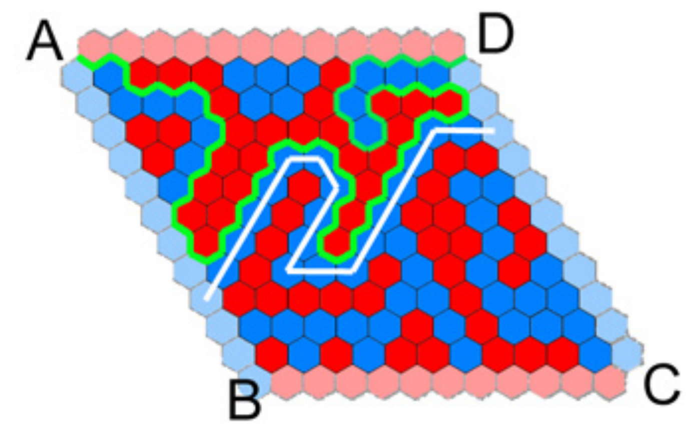
- We don't know what is the case for the initial position, and no winning strategy (for either of the players) is known
- According to Claude Shannon, there are 10^{43} legit positions in chess

[Zer13] Zermelo. Über eine Anwendung der Mengenlehre auf die Theorie des Schachspiels (Congress Mathematicians, 1912).
[Au89] Aumann. Lectures on Game Theory (1989).

Hex game



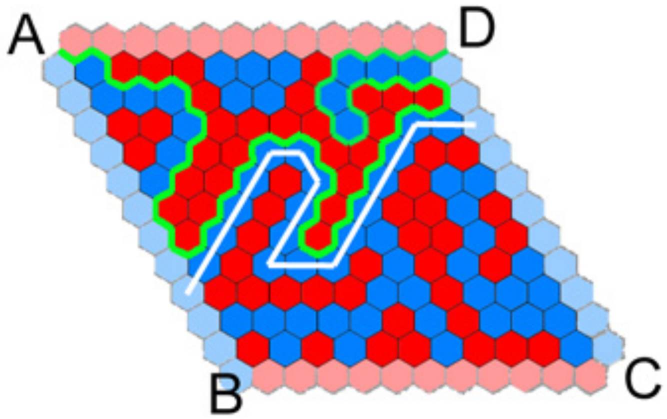
Hex game



Solving the Hex game

First player has always a winning strategy.

Hex game

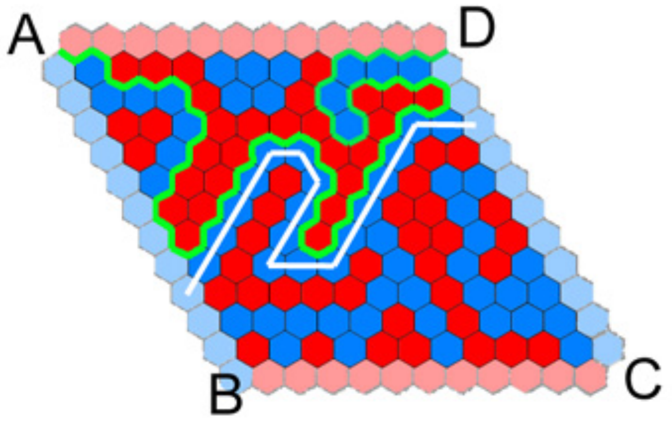


Solving the Hex game

First player has always a winning strategy.

- Determinacy results (no tie is possible) + strategy stealing argument

Hex game



Solving the Hex game

First player has always a winning strategy.

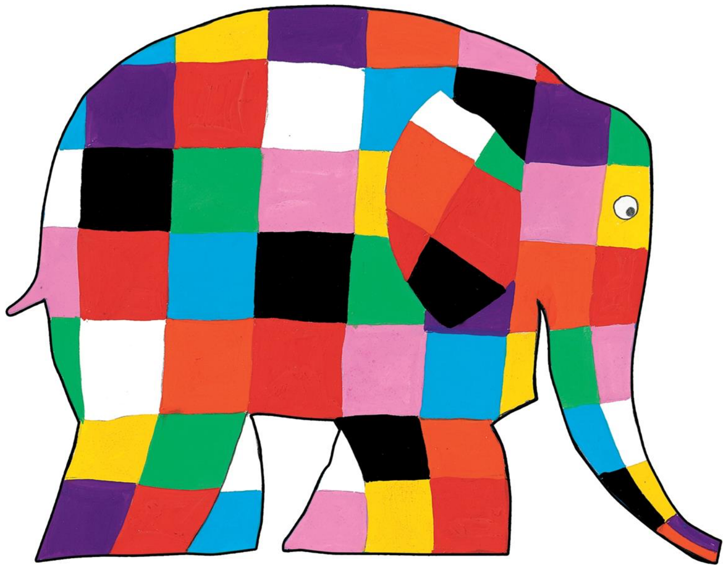
- Determinacy results (no tie is possible) + strategy stealing argument
- A winning strategy is not known yet.

What we do not consider

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

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

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

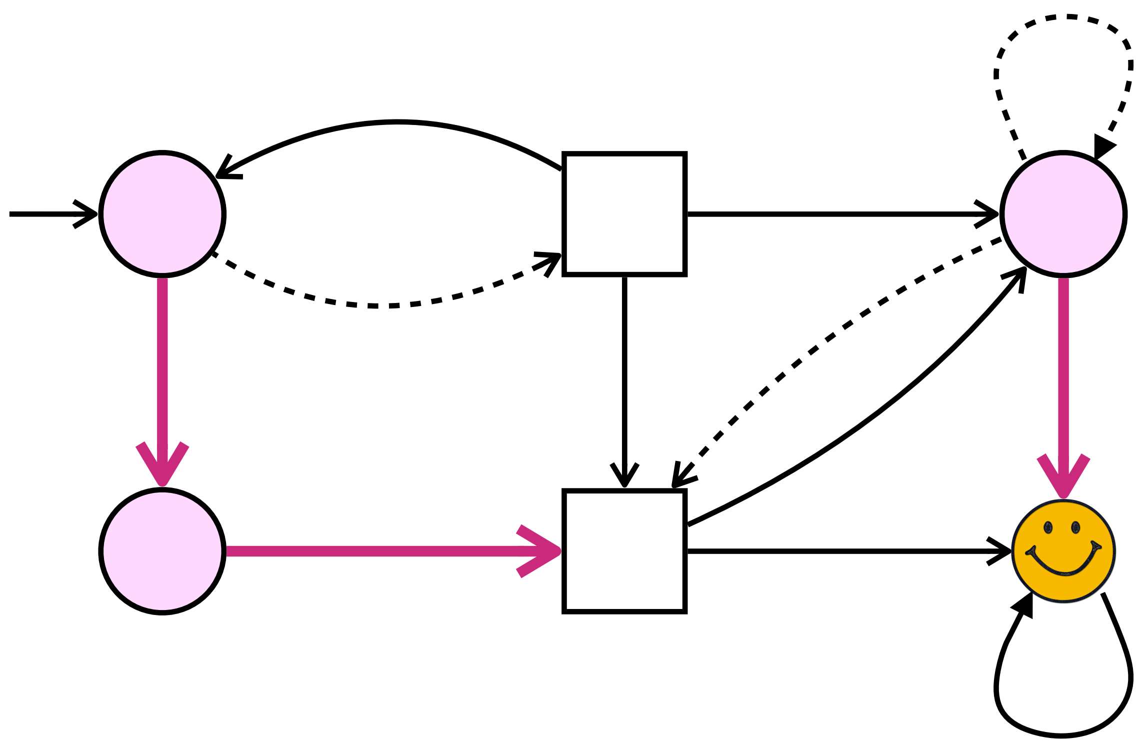
From $\sigma_i : S^*S_i \rightarrow E$ to $\sigma_i : S_i \rightarrow E$

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

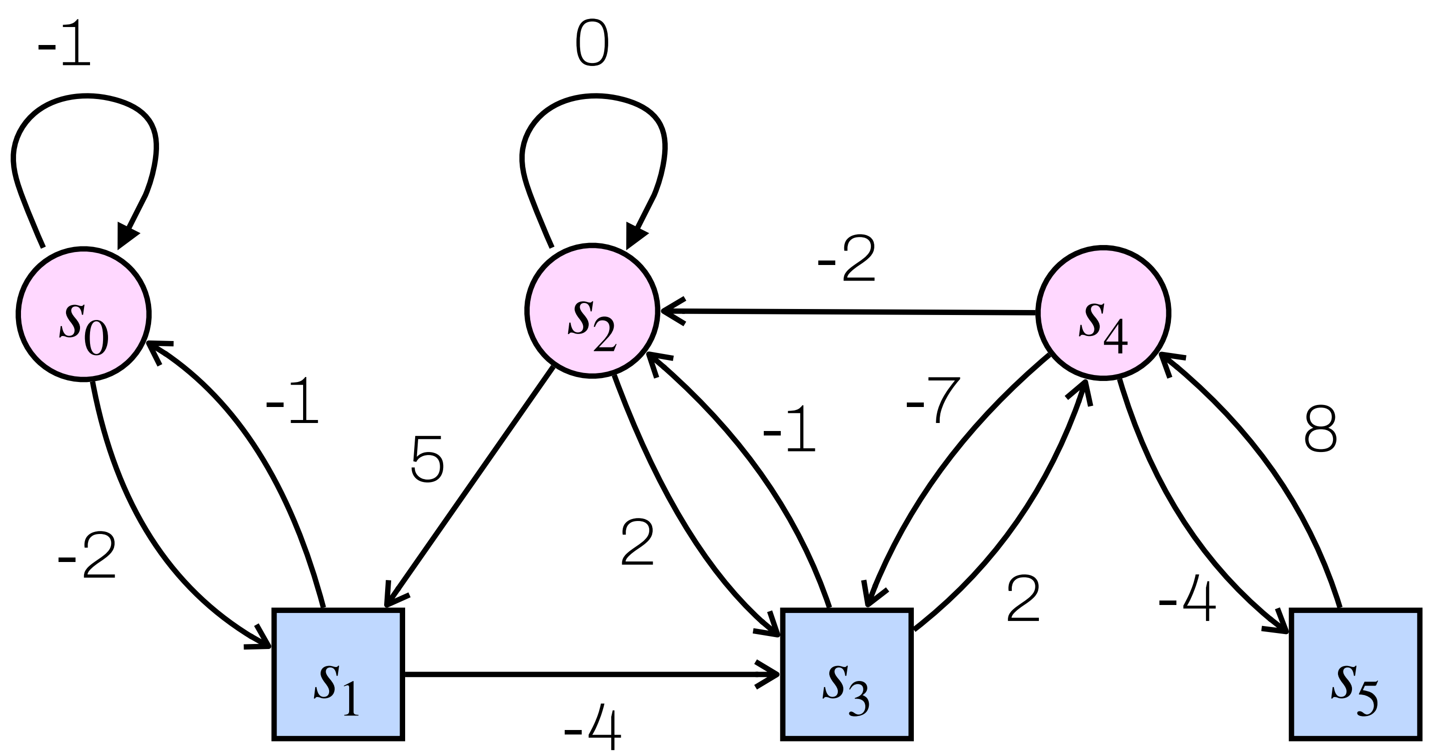
On the simplest side: positional strategies

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

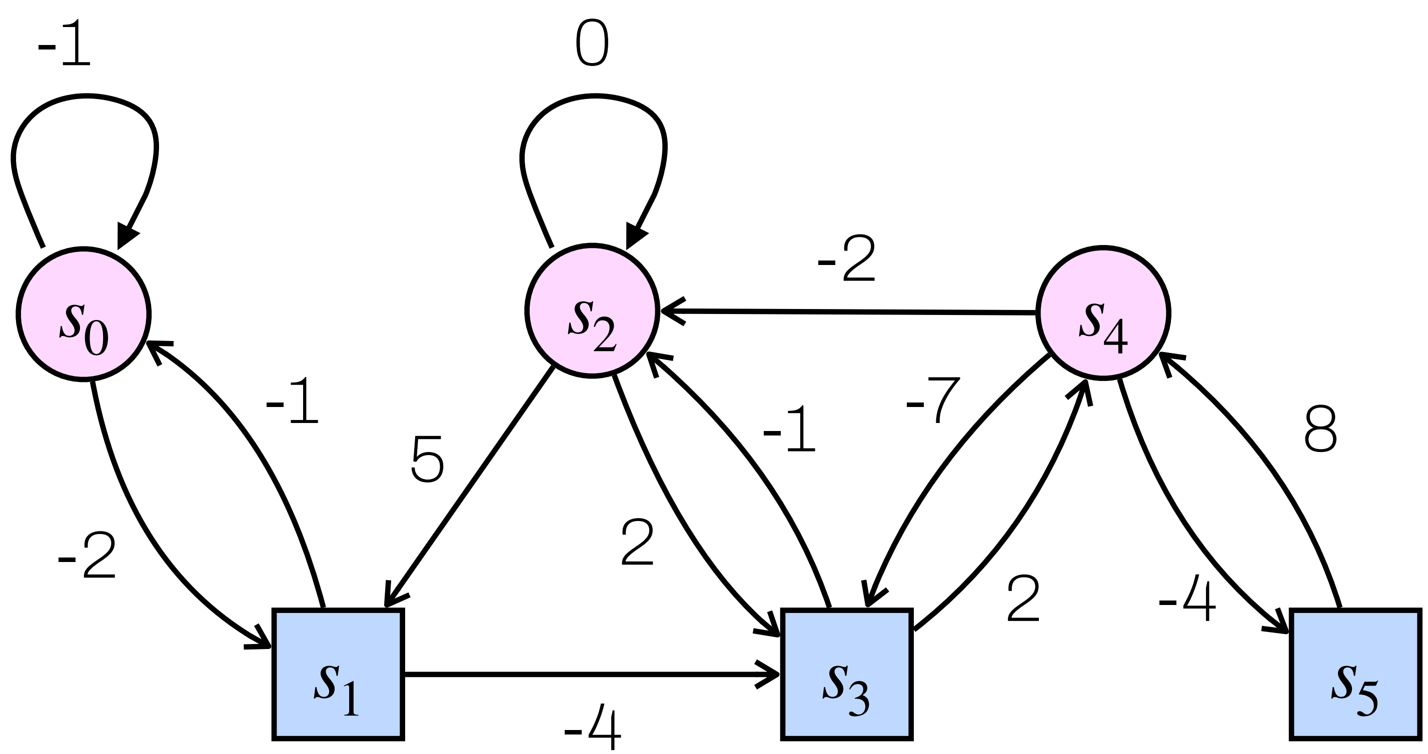


[Ohl21] Ohlmann. Monotonic graphs for Parity and Mean-Payoff games (PhD thesis).

Example: mean-payoff

- P_1 maximizes, P_2 minimizes

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

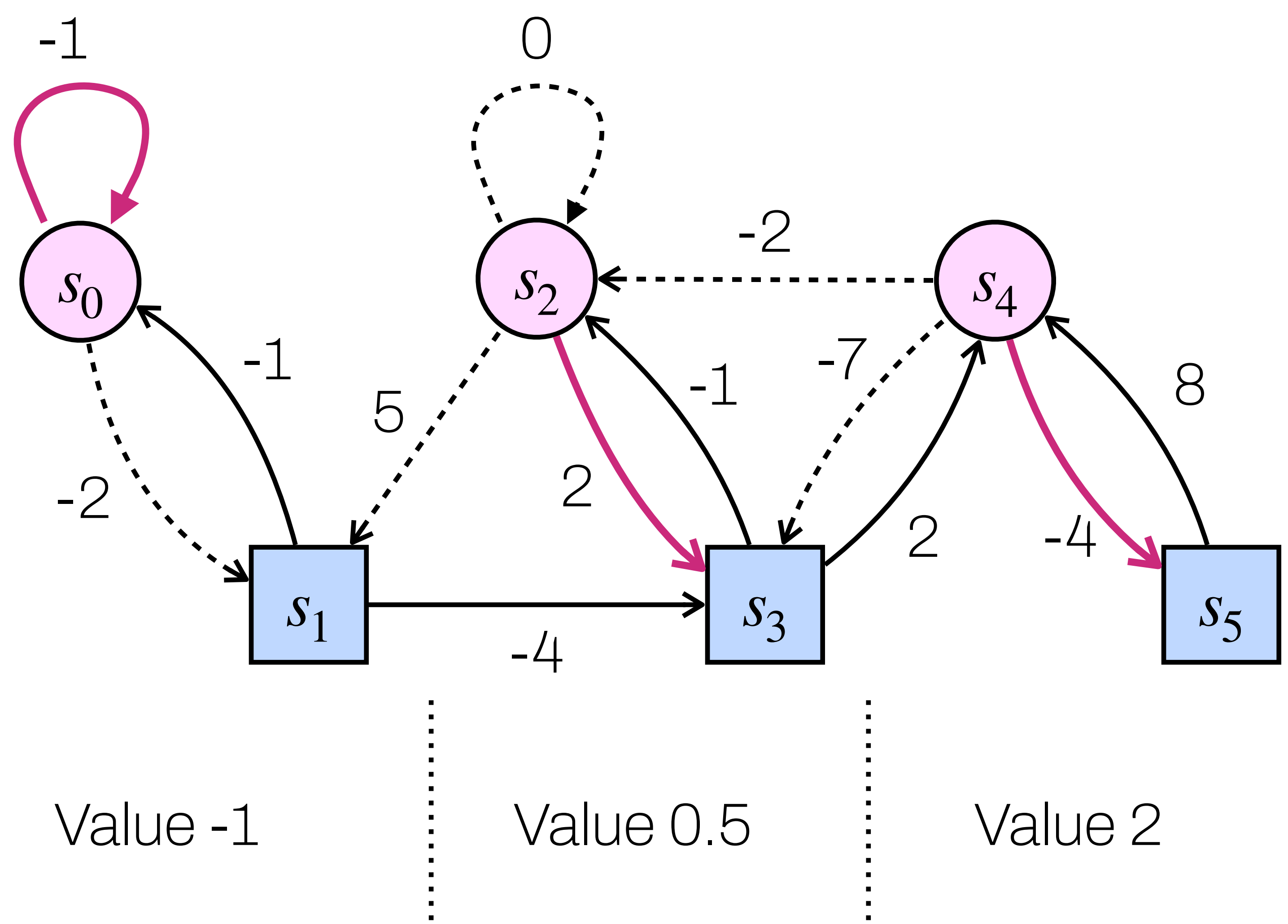


[Ohl21] Ohlmann. Monotonic graphs for Parity and Mean-Payoff games (PhD thesis).

Example: mean-payoff

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

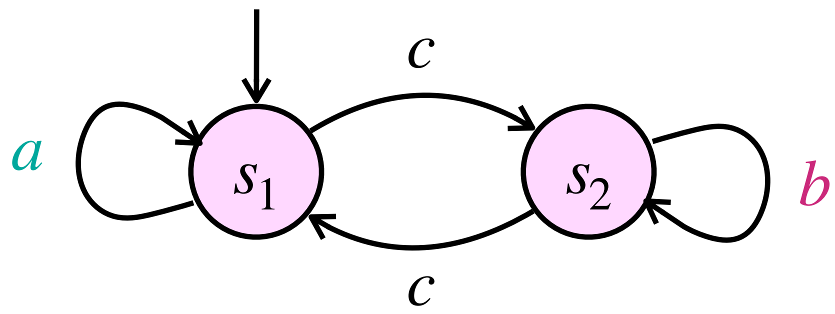
$$\overline{\mathrm{MP}} = \limsup_{n} \frac{\sum_{i \neq n} c_i}{n}$$



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

Do we need more?

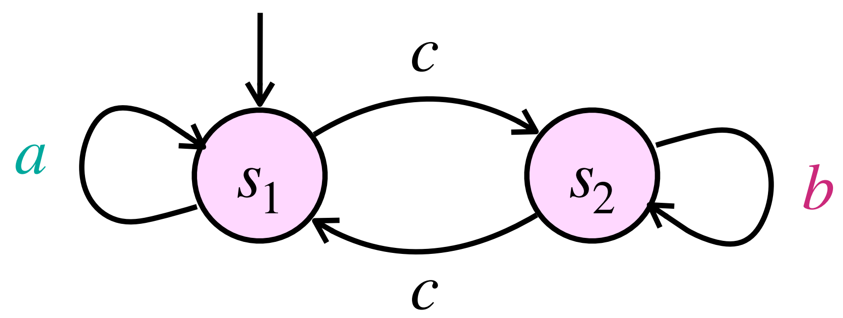
Examples



« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)

Examples



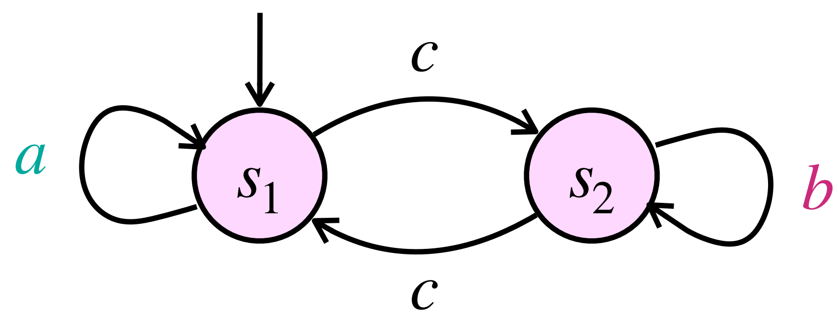
« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)

Winning strategy

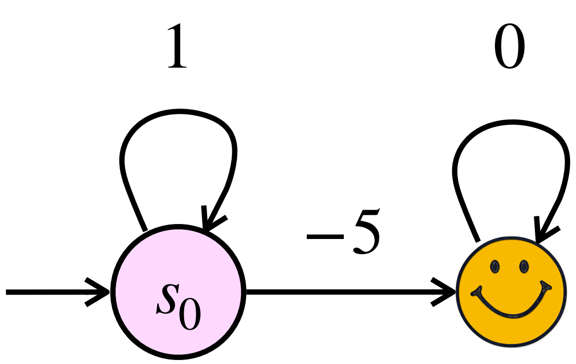
- At each visit to s_1, loop once in s_1 and then go to s_2
- At each visit to s_2, loop once in s_2 and then go to s_1
- Generates the sequence $(acbc)^\omega$

Examples



« See infinitely often both a and b »

Büchi(a) ∧ Büchi(b)



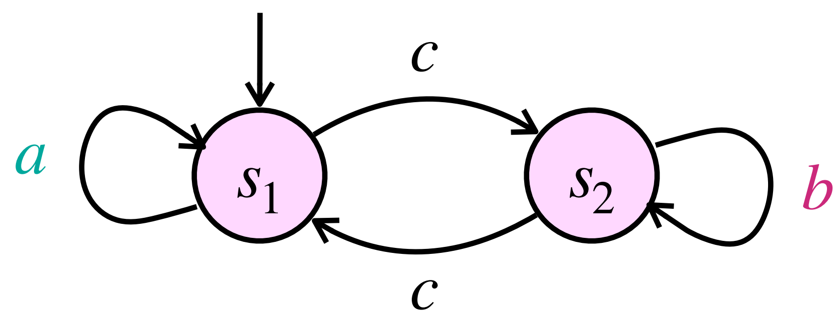
« Reach the target with energy level 0 »

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

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 $(acbc)^\omega$

Examples

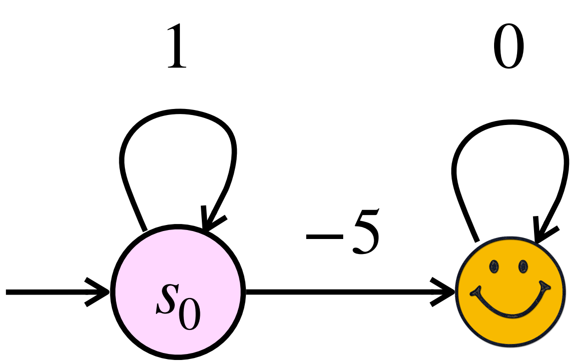


« See infinitely often both a and b »

Büchi$(a) \wedge$ 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 $(acbc)^\omega$



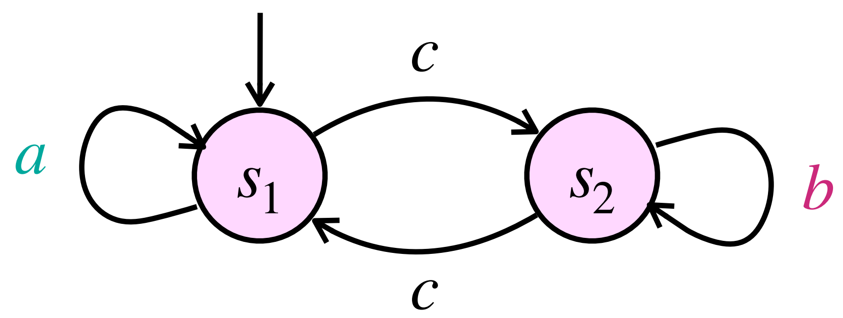
« Reach the target with energy level 0 »

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

Winning strategy

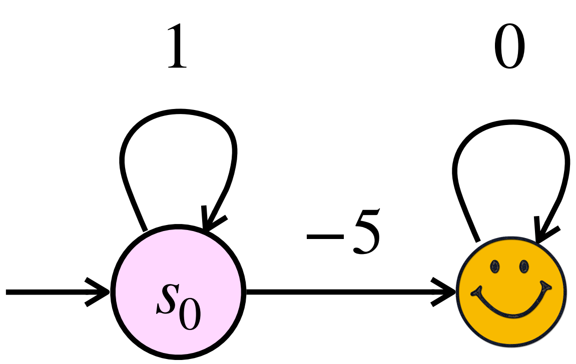
- Loop five times in s_0
- Then go to the target
- Generates the sequence of colors
$1\ 1\ 1\ 1\ 1\ \ -5\ 0\ 0\ 0 \ldots$

Examples



« See infinitely often both a and b »

Büchi$(a) \wedge$ Büchi(b)



« Reach the target with energy level 0 »

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

Winning strategy

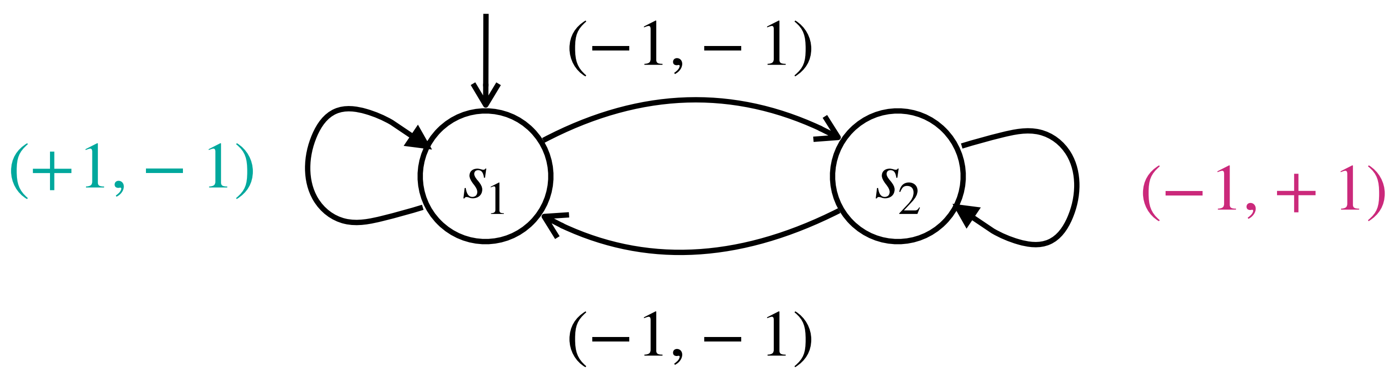
- At each visit to s_1, loop once in s_1 and then go to s_2
- At each visit to s_2, loop once in s_2 and then go to s_1
- Generates the sequence $(acbc)^\omega$

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

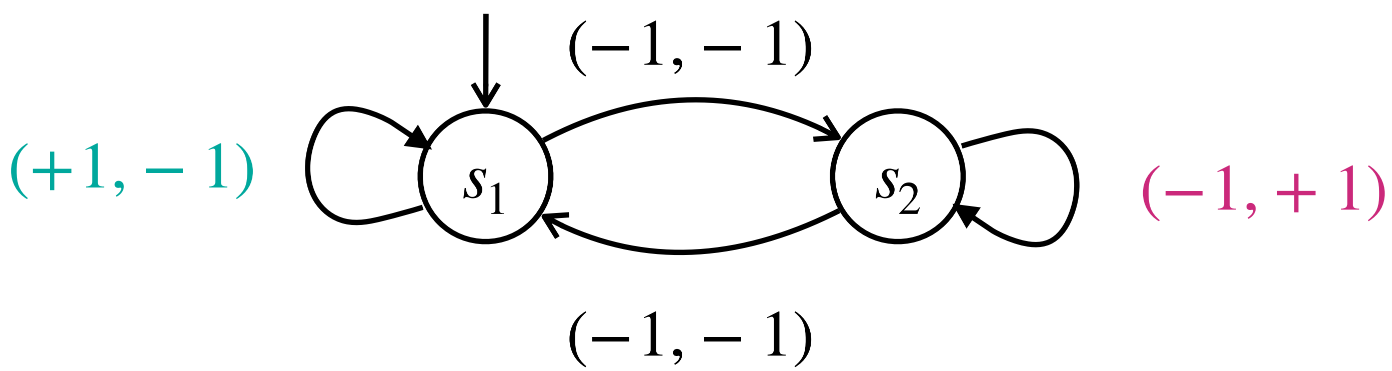
Example: multi-dimensional mean-payoff



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

So-called *multi-dimensional mean-payoff*

Example: multi-dimensional mean-payoff

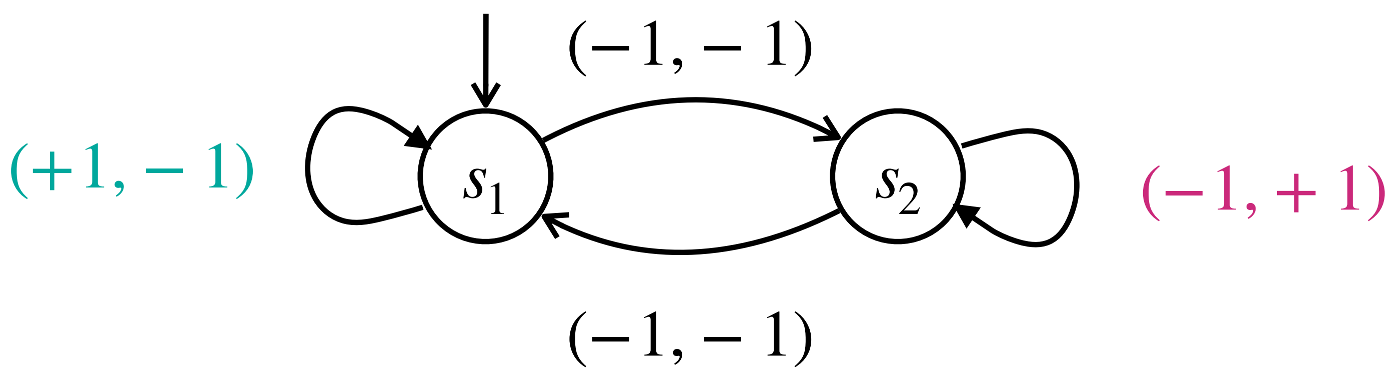


« Have a (limsup) mean-payoff ≥ 0 on both dimensions »
So-called *multi-dimensional mean-payoff*

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
$(-1,-1)\,(-1,+1)\,(-1,-1)(+1,-1)(+1,-1)(+1,-1)\,(-1,-1)$
$(-1,+1)(-1,+1)(-1,+1)(-1,+1)(-1,+1)\,(-1,-1)$
$(+1,-1)(+1,-1)(+1,-1)(+1,-1)(+1,-1)(+1,-1)(+1,-1)\,(-1,-1)$...

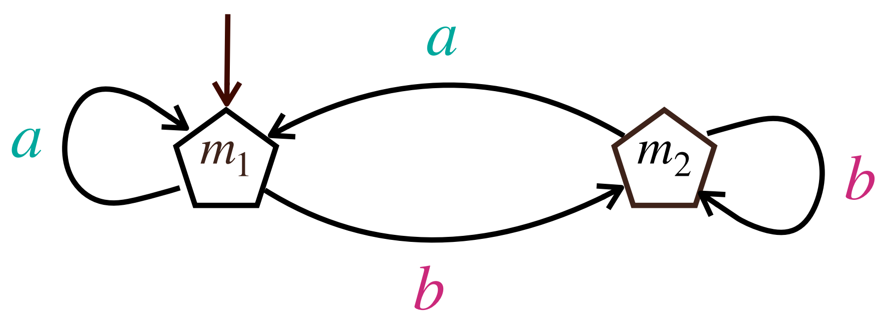
This strategy requires **infinite** memory, and this is unavoidable

We focus on finite memory!

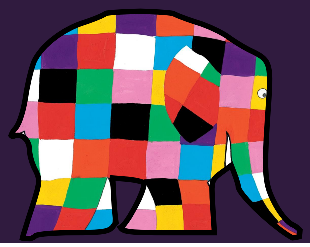
Chromatic* memory

Memory skeleton

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



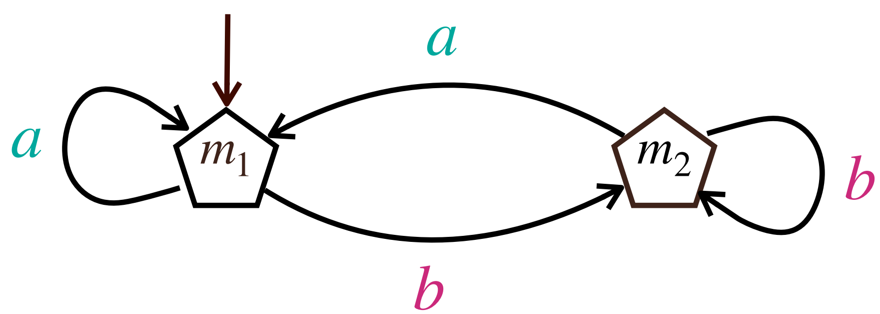
* Terminology by Kopczyński



Chromatic* memory

Memory skeleton

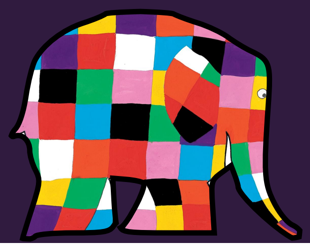
$$\mathscr{M} = (M, m_{\text{init}}, \alpha_{\text{upd}}) \text{ with } m_{\text{init}} \in M \text{ and } \alpha_{\text{upd}} : M \times C \to M$$



Not yet a strategy!

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

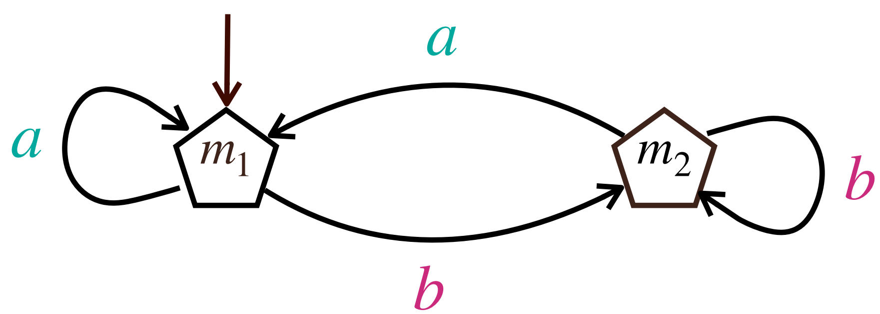
* Terminology by Kopczyński



Chromatic* memory

Memory skeleton

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



Not yet a strategy!

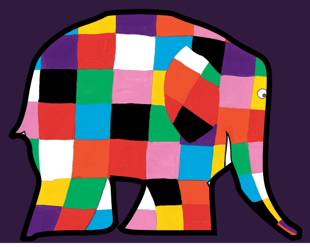
$\sigma_i : S^*S_i \to E$

Strategy with memory $\mathcal{M}$

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

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

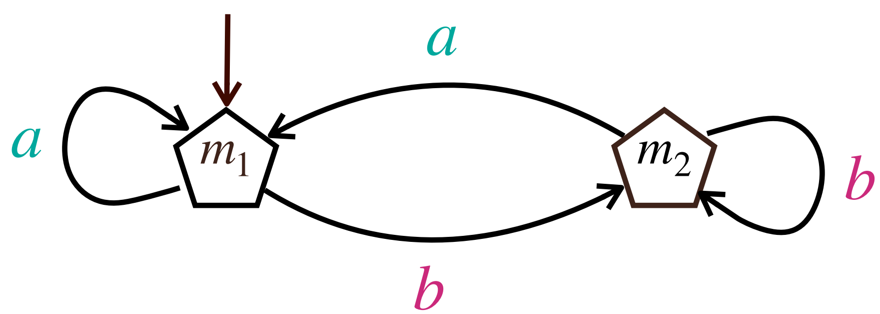
* Terminology by Kopczyński



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!

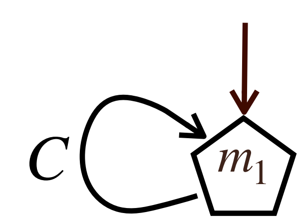
$\sigma_i : S^*S_i \to E$

Strategy with memory $\mathscr{M}$

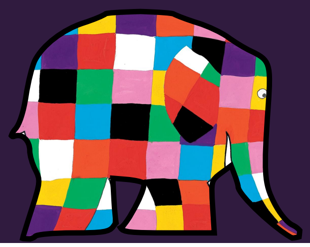
Additional next-move function $\alpha_{\text{next}} : M \times S_i \to E$

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

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



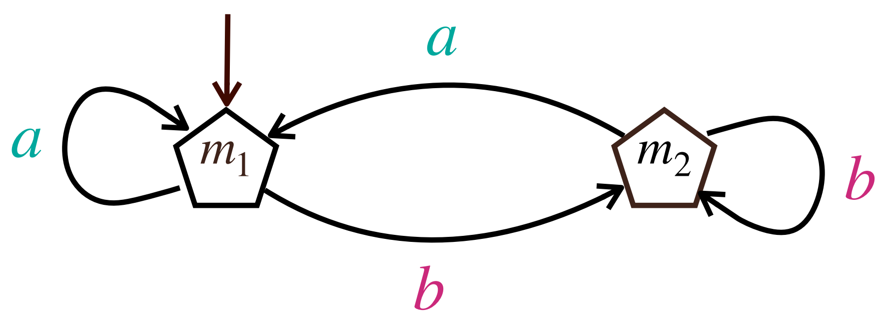
* Terminology by Kopczyński



Chromatic* memory

Memory skeleton

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



Not yet a strategy!

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

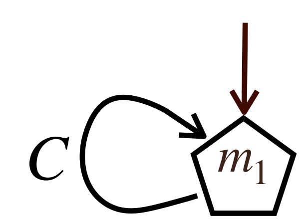
Chaotic* memory

Strategy with memory $\mathcal{M}$

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

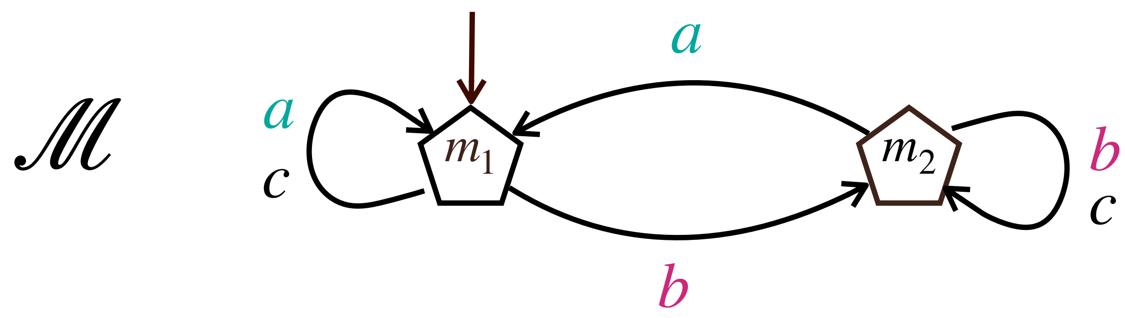
$(\mathcal{M}, \alpha_{\text{next}})$ defines a strategy!

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



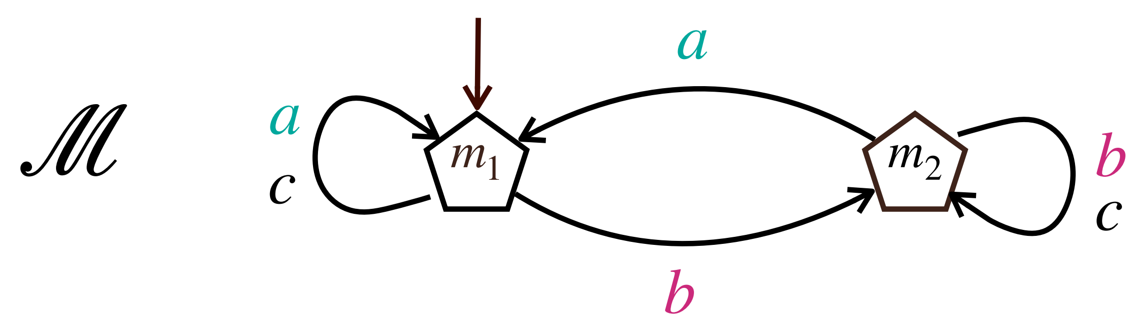
* Terminology by Kopczyński

Example of chromatic memory



This skeleton is sufficient for the winning condition
Büchi$(a) \wedge$ Büchi(b)

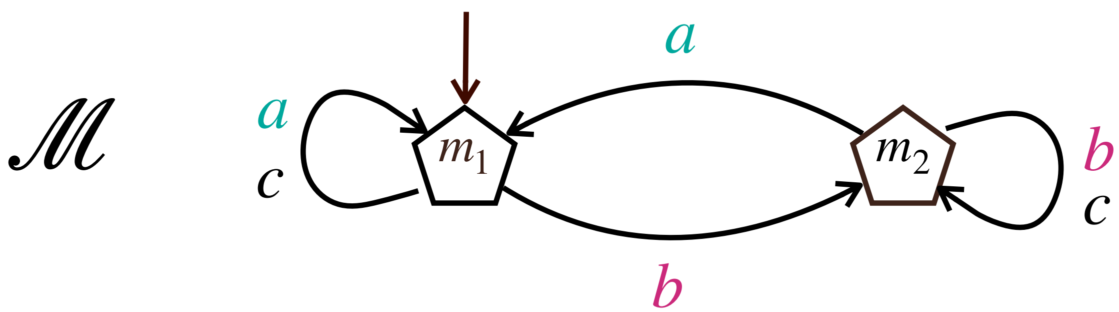
Example of chromatic memory



This skeleton is sufficient for the winning condition
Büchi(a) ∧ Büchi(b)

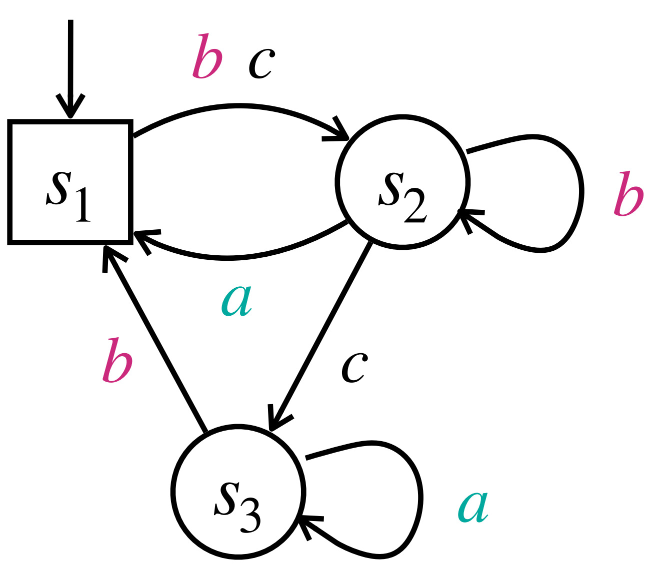
That is, for every game, if there is a winning strategy, there is one based on this skeleton

Example of chromatic memory

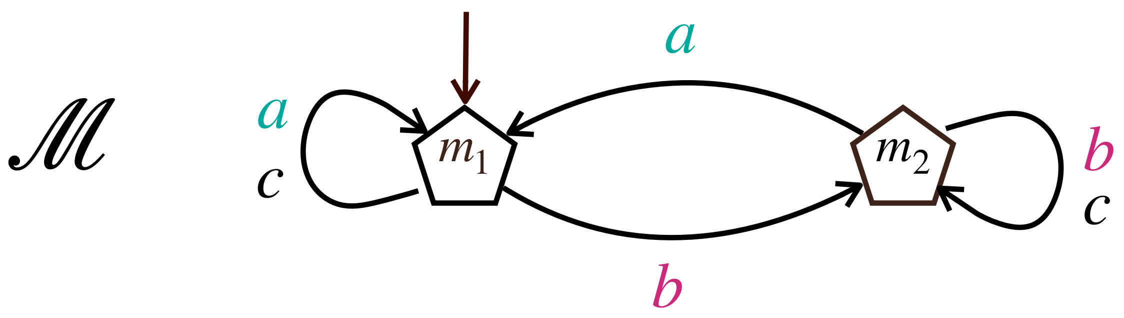


This skeleton is sufficient for the winning condition
Büchi(a) ∧ Büchi(b)

That is, for every game, if there is a winning strategy, there is one based on this skeleton

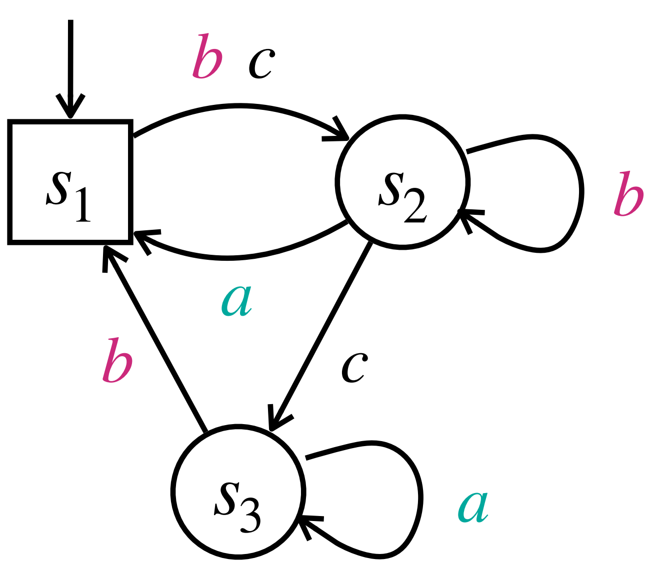


Example of chromatic memory



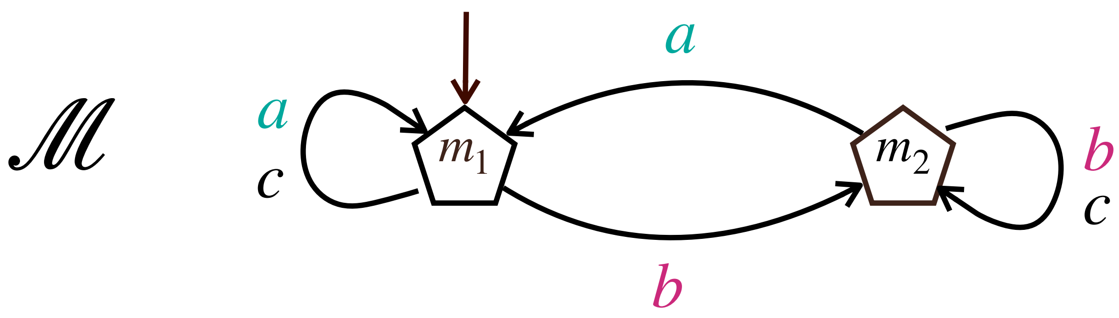
This skeleton is sufficient for the winning condition
Büchi(a) ∧ Büchi(b)

That is, for every game, if there is a winning strategy, there is one based on this skeleton



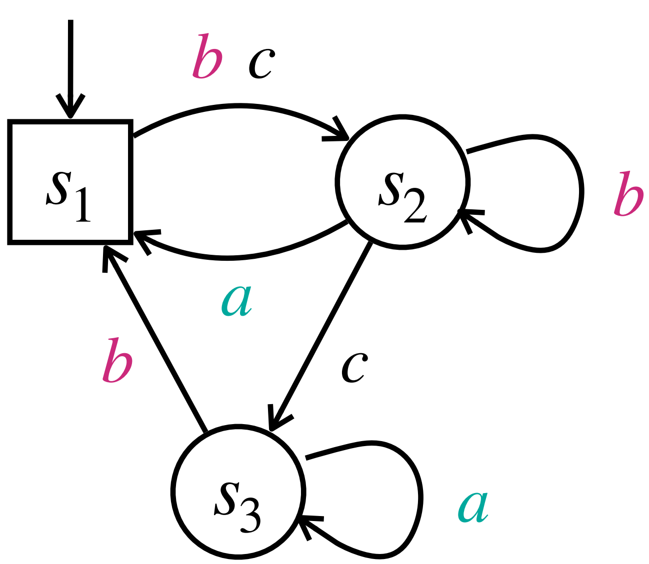
$$\alpha_{\text{next}} : \quad M \times S_1 \to E$$
$$(m_1, s_2) \mapsto (s_2, b, s_2)$$
$$(m_2, s_2) \mapsto (s_2, a, s_1)$$
$$(m_\star, s_3) \mapsto (s_3, b, s_1)$$

Example of chromatic memory



This skeleton is sufficient for the winning condition
Büchi(a) $\wedge$ Büchi(b)

That is, for every game, if there is a winning strategy, there is one based on this skeleton



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

Our goal

Understand well low-memory specifications

Our goal

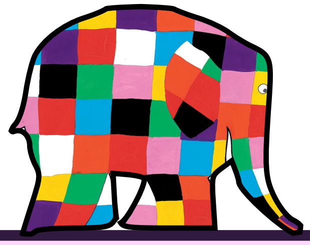
Understand well low-memory specifications

Positional / finite-memory determinacy

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

Our goal

Understand well low-memory specifications

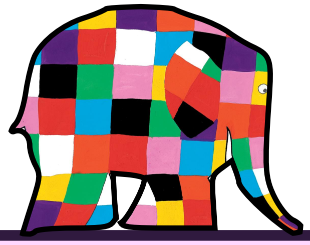


Positional / finite-memory determinacy

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

Our goal

Understand well low-memory specifications



Positional / finite-memory determinacy

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

- Finite vs infinite games

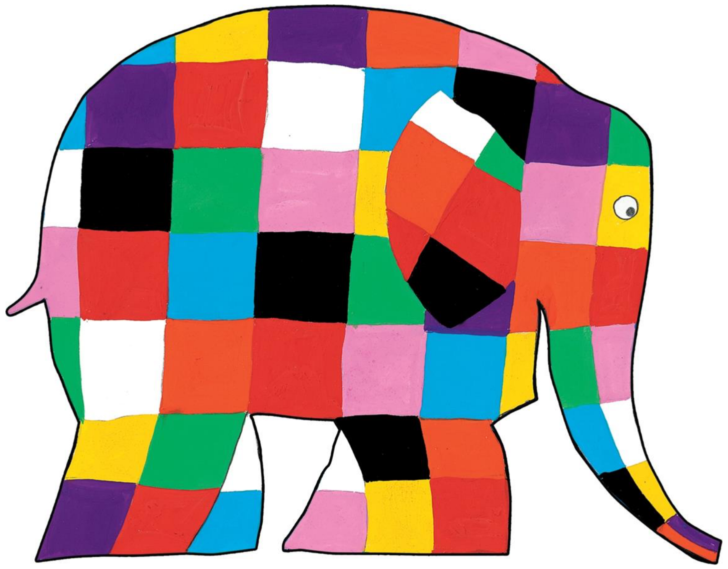
Laboratoire Méthodes Formelles

université PARIS-SACLAY

cnrs

école normale supérieure paris—saclay

Characterizing positional and chromatic finite-memory determinacy in finite games



The approach

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

The approach

- Characterize winning objectives ensuring **memoryless determinacy**, that is, the existence of positional winning strategies (for both players) in all finite games

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

The approach

- Characterize winning objectives ensuring **memoryless determinacy**, that is, the existence of positional winning strategies (for both players) in all finite games
- Should apply to reachability/safety objectives, mean-payoff, parity, ...

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

The approach

- Characterize winning objectives ensuring **memoryless determinacy**, that is, the existence of positional winning strategies (for both players) in all finite games
- Should apply to reachability/safety objectives, mean-payoff, parity, ...
- Fundamental reference: [GZ05]

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

Properties of preference relations

Properties of preference relations

- Let $\sqsubseteq$ be a preference relation (for P_1).
- Let $W \subseteq C^{\omega}$ be a winning objective (for P_1).

Properties of preference relations

- Let $\sqsubseteq$ be a preference relation (for P_1).
- Let $W \subseteq C^\omega$ be a winning objective (for P_1).
- It is said **monotone** whenever:

Properties of preference relations

- Let $\sqsubseteq$ be a preference relation (for P_1).
- Let $W \subseteq C^\omega$ be a winning objective (for P_1).

- It is said **monotone** whenever:

- It is said **selective** whenever:

Properties of preference relations

- Let $\sqsubseteq$ be a preference relation (for P_1).
- Let $W \subseteq C^\omega$ be a winning objective (for P_1).
- It is said **monotone** whenever:

- It is said **selective** whenever:

If this is in W then one of those is in W

Two characterizations

Let $\sqsubseteq$ be a preference relation (for P_1).

Characterization - Two-player games

The two following assertions are equivalent:

1. All finite games have positional optimal strategies for both players;
2. Both $\sqsubseteq$ and $\sqsubseteq^{-1}$ are monotone and selective.

Two characterizations

Let $\sqsubseteq$ be a preference relation (for P_1).

Characterization - Two-player games

The two following assertions are equivalent:

1. All finite games have positional optimal strategies for both players;
2. Both $\sqsubseteq$ and $\sqsubseteq^{-1}$ are monotone and selective.

Characterization - One-player games

The two following assertions are equivalent:

1. All finite P_1-games have positional optimal strategies;
2. $\sqsubseteq$ is monotone and selective.

Applications

Lifting theorem

P_i has positional optimal strategies in all finite P_i-games

$$\Downarrow$$

Both players have positional optimal strategies in all finite 2-player games.

Applications

Lifting theorem

P_i has positional optimal strategies in all finite P_i-games

$$\Downarrow$$

Both players have positional optimal strategies in all finite 2-player games.

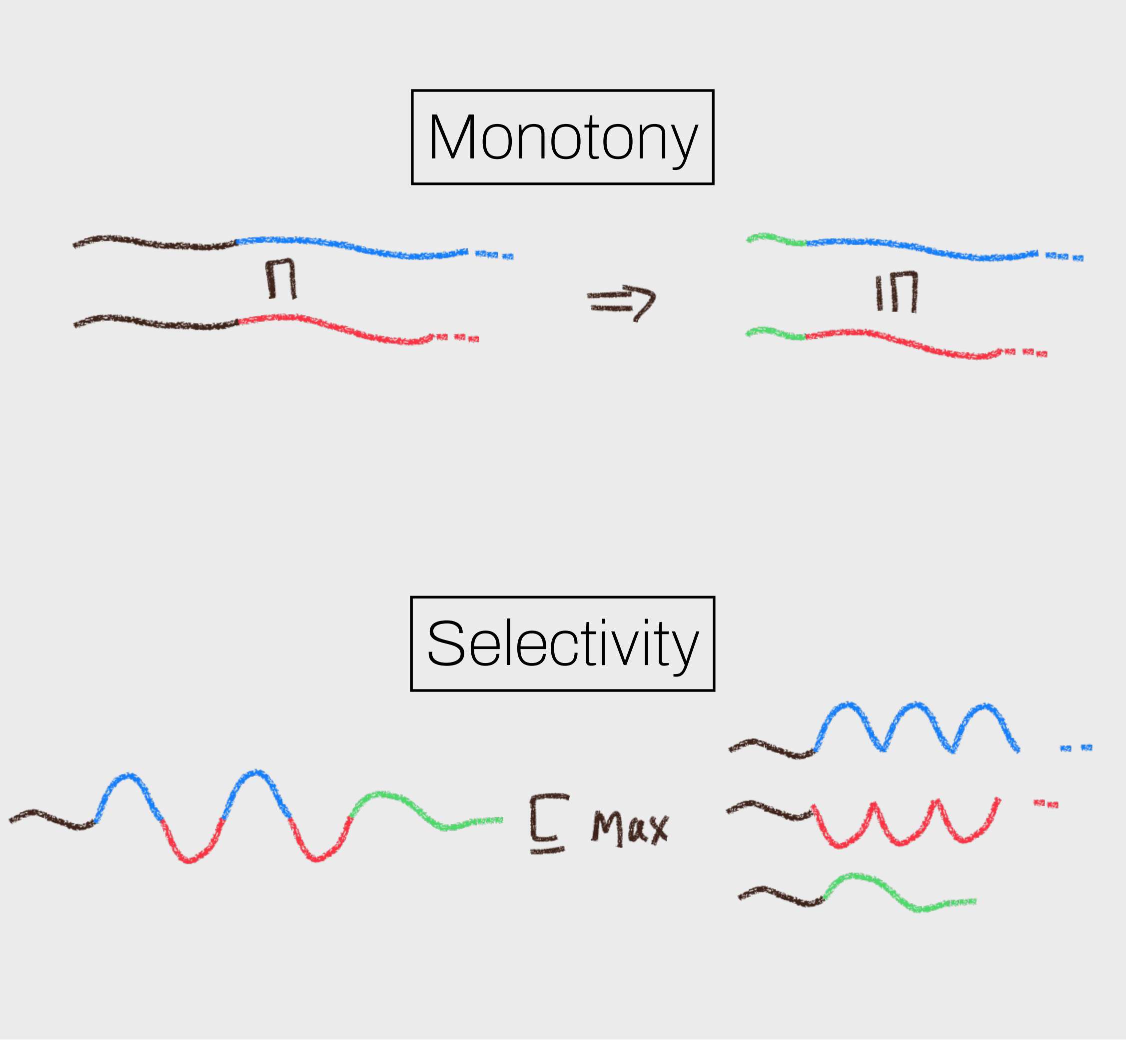
Very powerful and extremely useful in practice

- Easy to analyse the one-player case (graph analysis)
  - Mean-payoff, average-energy [BMRLL15]

[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).

Properties of preference relations — Adding memory

- Let $\sqsubseteq$ be a preference relation (for P_1).
  Let $\mathscr{M}$ be a memory skeleton.

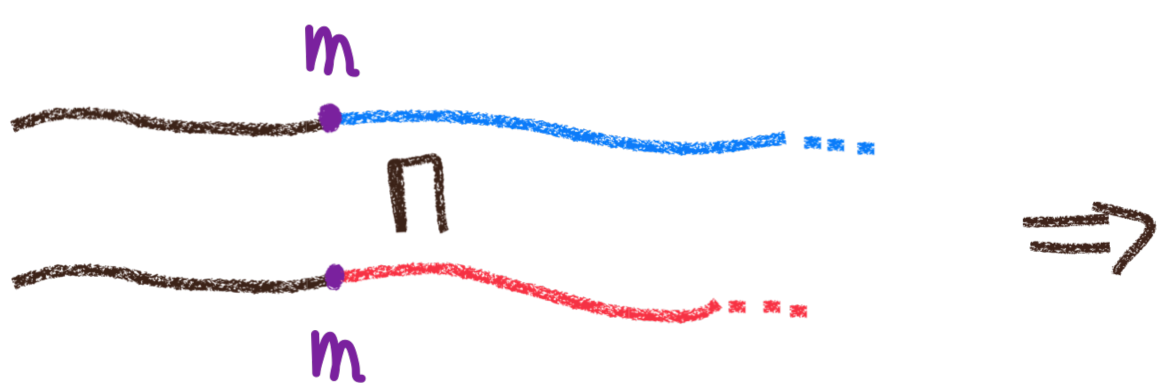
- It is said $\mathscr{M}$**-monotone** whenever:

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

Properties of preference relations — Adding memory

- Let $\sqsubseteq$ be a preference relation (for P_1).
  Let $\mathscr{M}$ be a memory skeleton.

- It is said **$\mathscr{M}$-monotone** whenever:

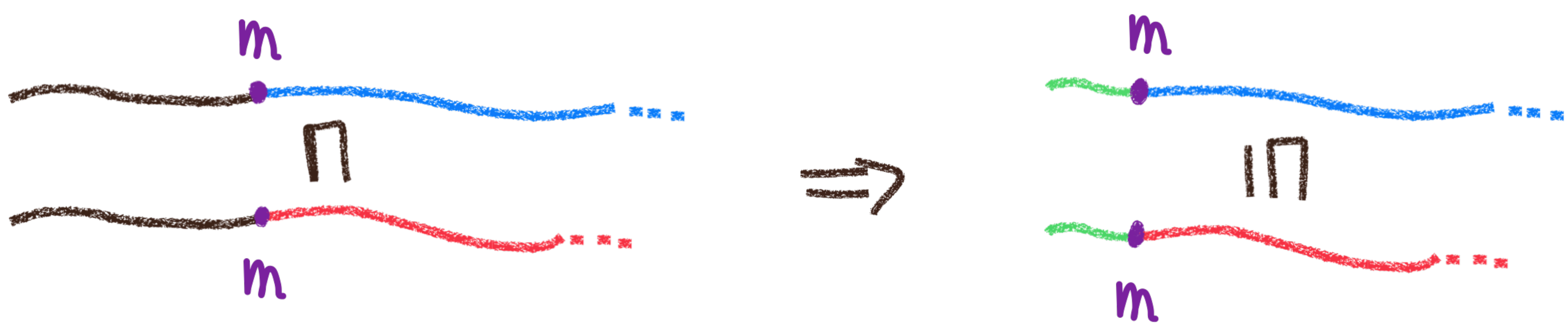


- It is said **$\mathscr{M}$-selective** whenever:

Properties of preference relations — Adding memory

- Let $\sqsubseteq$ be a preference relation (for P_1).
  Let $\mathscr{M}$ be a memory skeleton.

- It is said $\mathscr{M}$**-monotone** whenever:

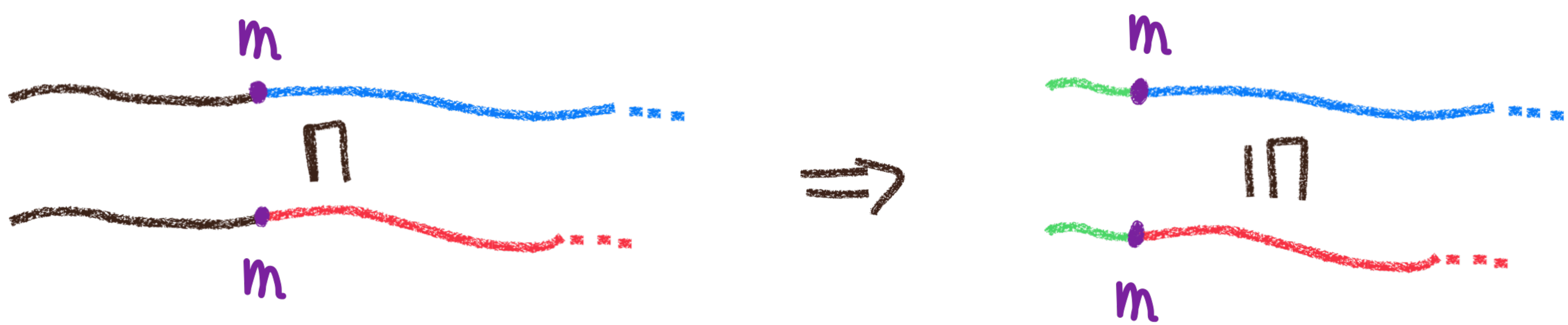


- It is said $\mathscr{M}$**-selective** whenever:

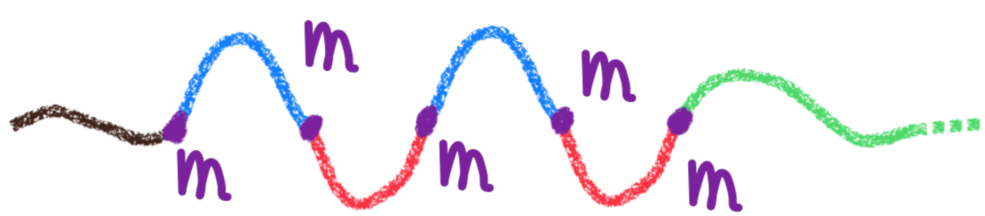
Properties of preference relations — Adding memory

- Let $\sqsubseteq$ be a preference relation (for P_1).
  Let $\mathscr{M}$ be a memory skeleton.

- It is said $\mathscr{M}$-**monotone** whenever:



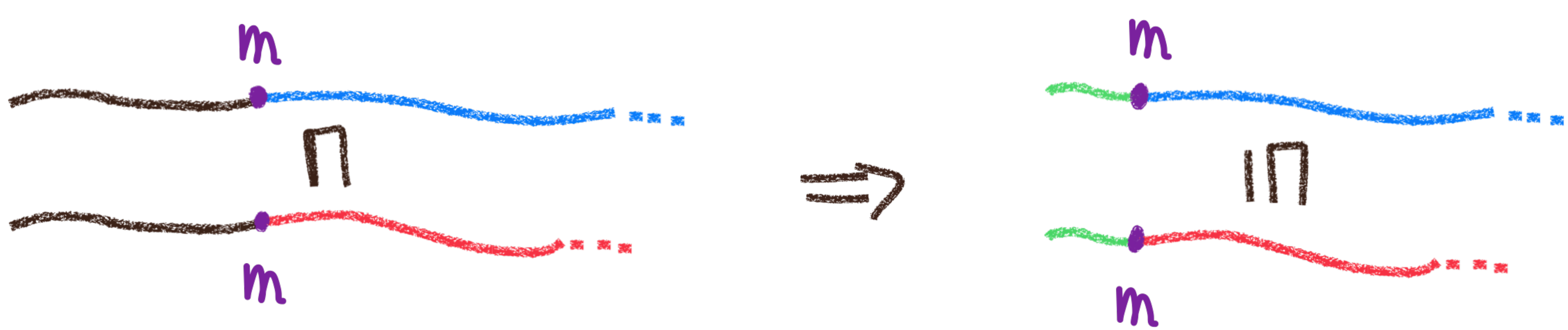
- It is said $\mathscr{M}$-**selective** whenever:



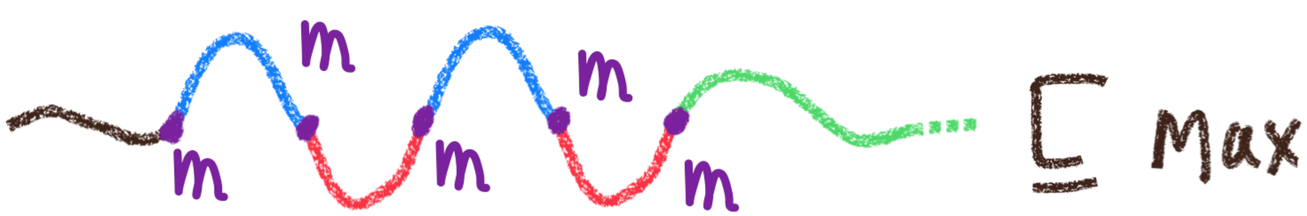
Properties of preference relations — Adding memory

- Let $\sqsubseteq$ be a preference relation (for P_1).
  Let $\mathscr{M}$ be a memory skeleton.

- It is said **$\mathscr{M}$-monotone** whenever:



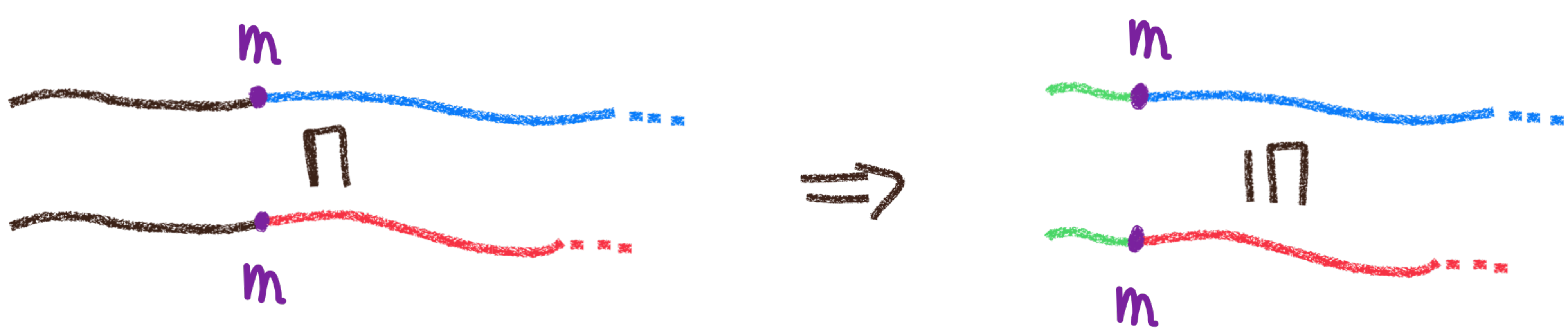
- It is said **$\mathscr{M}$-selective** whenever:



Properties of preference relations — Adding memory

- Let $\sqsubseteq$ be a preference relation (for P_1).
  Let $\mathscr{M}$ be a memory skeleton.

- It is said $\mathscr{M}$**-monotone** whenever:



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

Two characterizations

Let $\sqsubseteq$ be a preference relation (for P_1) and $\mathcal{M}$ be a memory skeleton.

Characterization - Two-player games

The two following assertions are equivalent:

1. All finite games have $\mathcal{M}$-based optimal strategies for both players;
2. Both $\sqsubseteq$ and $\sqsubseteq^{-1}$ are $\mathcal{M}$-monotone and $\mathcal{M}$-selective.

Two characterizations

Let $\sqsubseteq$ be a preference relation (for P_1) and $\mathscr{M}$ be a memory skeleton.

Characterization - Two-player games

The two following assertions are equivalent:

1. All finite games have $\mathscr{M}$-based optimal strategies for both players;
2. Both $\sqsubseteq$ and $\sqsubseteq^{-1}$ are $\mathscr{M}$-monotone and $\mathscr{M}$-selective.

Characterization - One-player games

The two following assertions are equivalent:

1. All finite P_1-games have $\mathscr{M}$-based optimal strategies;
2. $\sqsubseteq$ is $\mathscr{M}$-monotone and $\mathscr{M}$-selective.

Two characterizations

Let $\sqsubseteq$ be a preference relation (for P_1) and $\mathcal{M}$ be a memory skeleton.

Characterization - Two-player games

The two following assertions are equivalent:

1. All finite games have $\mathcal{M}$-based optimal strategies for both players;
2. Both $\sqsubseteq$ and $\sqsubseteq^{-1}$ are $\mathcal{M}$-monotone and $\mathcal{M}$-selective.

Characterization - One-player games

The two following assertions are equivalent:

1. All finite P_1-games have $\mathcal{M}$-based optimal strategies;
2. $\sqsubseteq$ is $\mathcal{M}$-monotone and $\mathcal{M}$-selective.

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

Applications

Lifting theorem

P_i has $\mathscr{M}_i$-based optimal strategies in all finite P_i-games

$$\Downarrow$$

Both players have $(\mathscr{M}_1 \times \mathscr{M}_2)$-based optimal strategies in all finite two-player games.

Applications

Lifting theorem

P_i has $\mathcal{M}_i$-based optimal strategies in all finite P_i-games

$$\Downarrow$$

Both players have $(\mathcal{M}_1 \times \mathcal{M}_2)$-based optimal strategies in all finite two-player games.

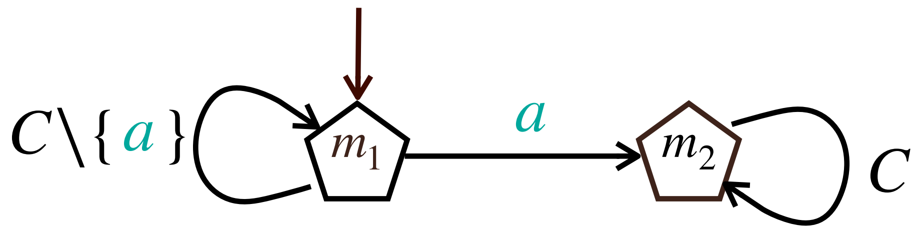
Very powerful and extremely useful in practice

- Easy to analyse the one-player case (graph analysis)
  - Conjunction of ω-regular objectives

Example of application

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

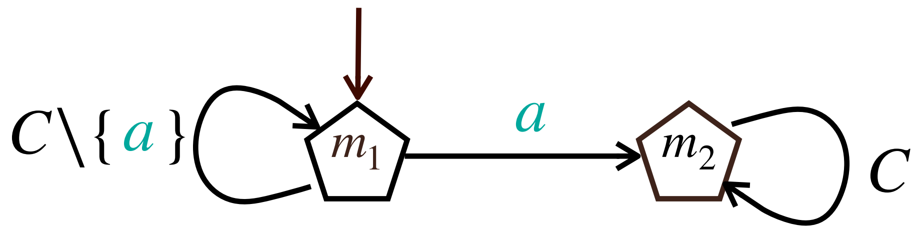
$\mathcal{M}_1$



Example of application

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

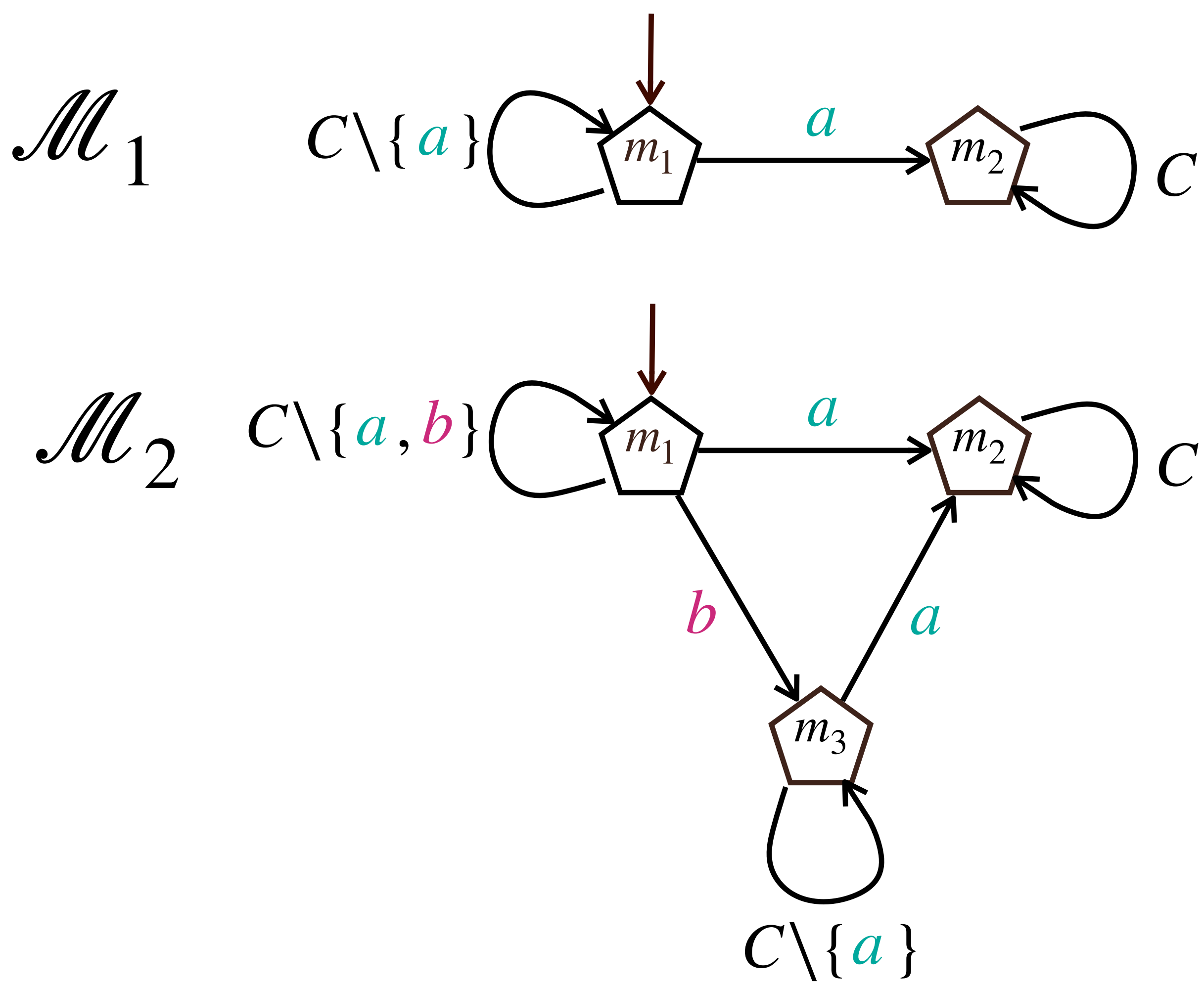
$\mathscr{M}_1$



$\sqsubseteq_W$ is $\mathscr{M}_1$-monotone
but not $\mathscr{M}_1$-selective

Example of application

W = Reach(a) ∧ Reach(b)

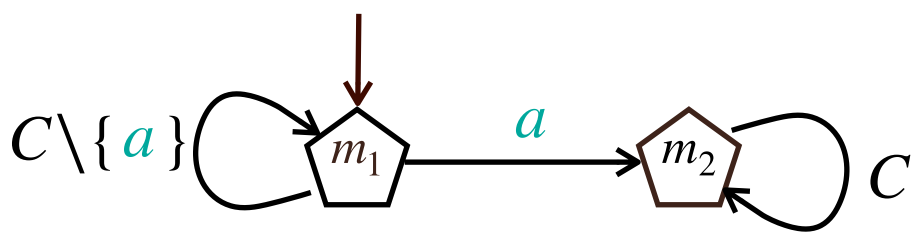


$\sqsubseteq_W$ is $\mathcal{M}_1$-monotone but not $\mathcal{M}_1$-selective

Example of application

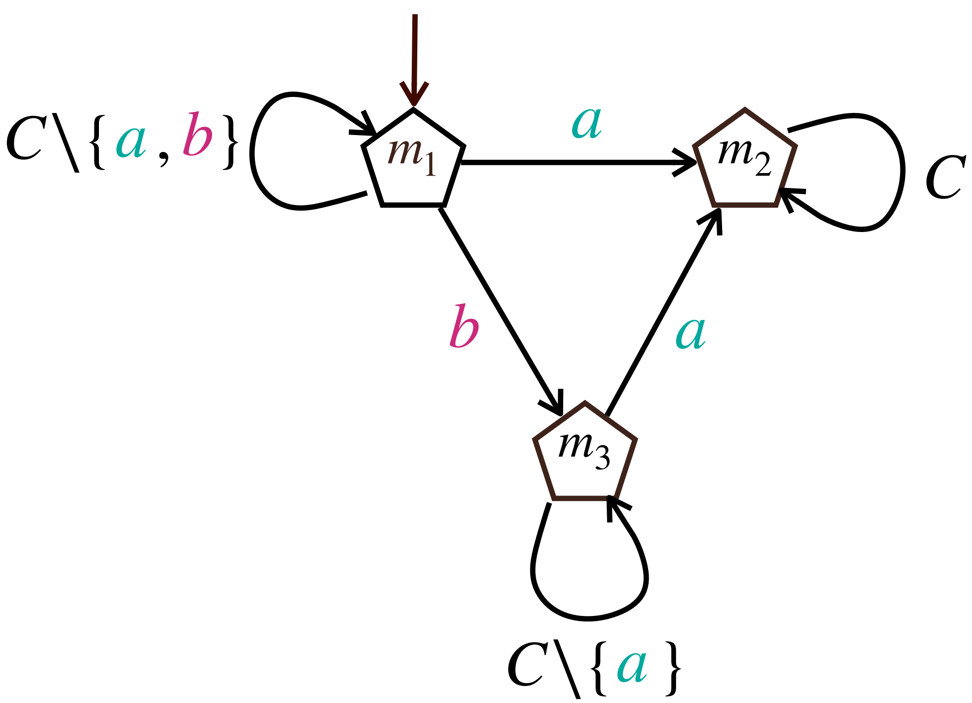
$$W = \text{Reach}(a) \wedge \text{Reach}(b)$$

$\mathscr{M}_1$



$\sqsubseteq_W$ is $\mathscr{M}_1$-monotone but not $\mathscr{M}_1$-selective

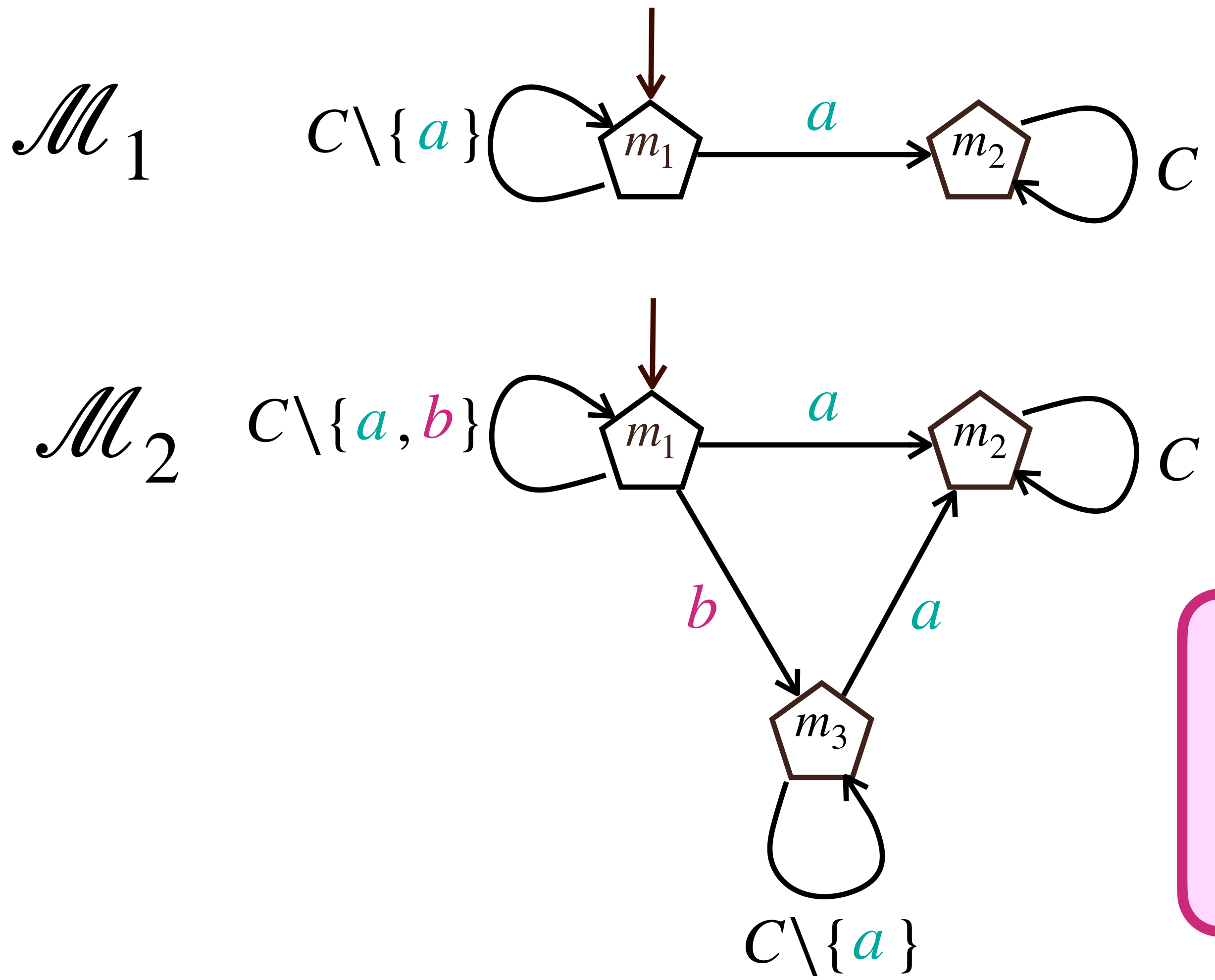
$\mathscr{M}_2$



$\sqsubseteq_W$ is $\mathscr{M}_2$-selective

Example of application

$W = \mathrm{Reach}(a) \wedge \mathrm{Reach}(b)$



$\sqsubseteq_W$ is $\mathcal{M}_1$-monotone but not $\mathcal{M}_1$-selective

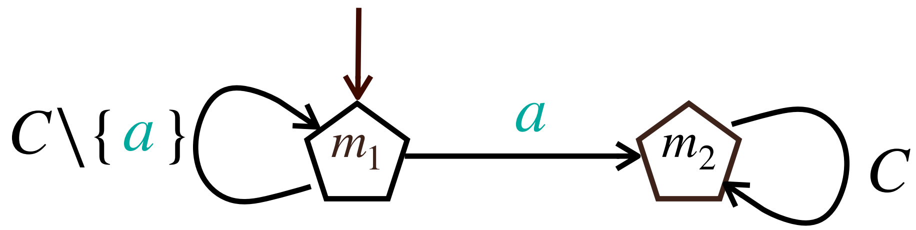
$\sqsubseteq_W$ is $\mathcal{M}_2$-selective

- $\sqsubseteq_W$ is $\mathcal{M}_1$-monotone and $\mathcal{M}_2$-selective
- $\sqsubseteq_W^{-1}$ is $\mathcal{M}_1$-monotone and $\mathcal{M}_{\mathrm{triv}}$-selective

Example of application

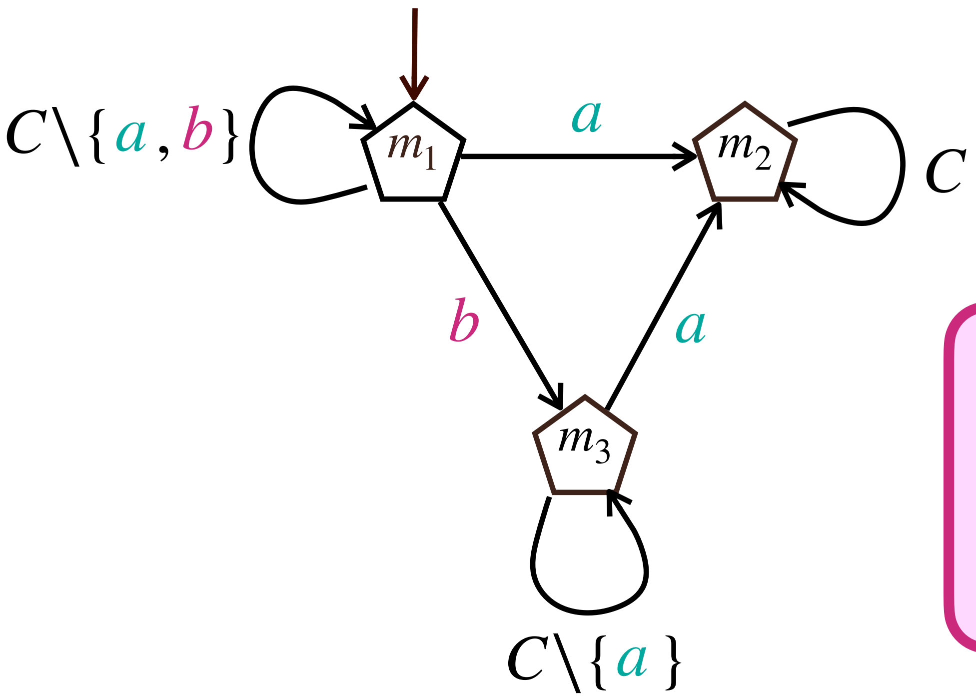
W = Reach(a) ∧ Reach(b)

$\mathcal{M}_1$



$\sqsubseteq_W$ is $\mathcal{M}_1$-monotone but not $\mathcal{M}_1$-selective

$\mathcal{M}_2$



$\sqsubseteq_W$ is $\mathcal{M}_2$-selective

- $\sqsubseteq_W$ is $\mathcal{M}_1$-monotone and $\mathcal{M}_2$-selective
- $\sqsubseteq_W^{-1}$ is $\mathcal{M}_1$-monotone and $\mathcal{M}_{\text{triv}}$-selective

→ Memory $\mathcal{M}_2$ 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)

Partial conclusion

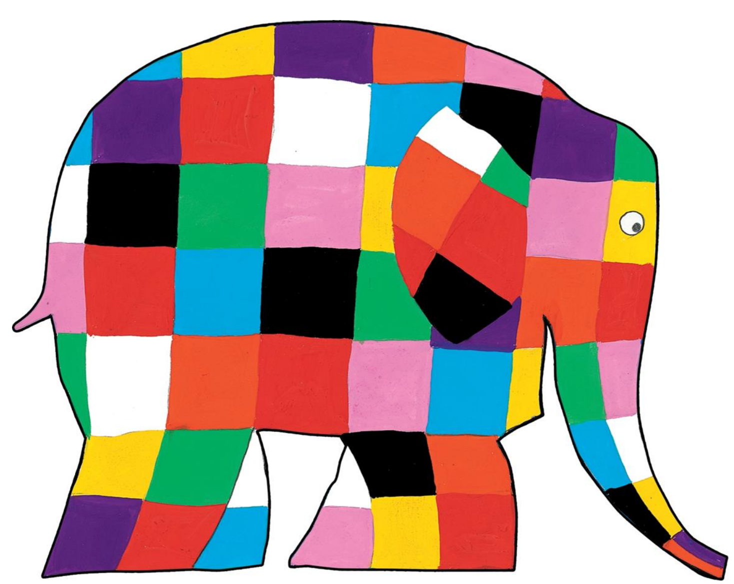
Finite games

- 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)
- Further questions:
  - Can we reduce/optimize the memory?
  - What about chaotic finite memory?
  - Can we focus on one player (so-called half-positionality)?

LMF Laboratoire Méthodes Formelles

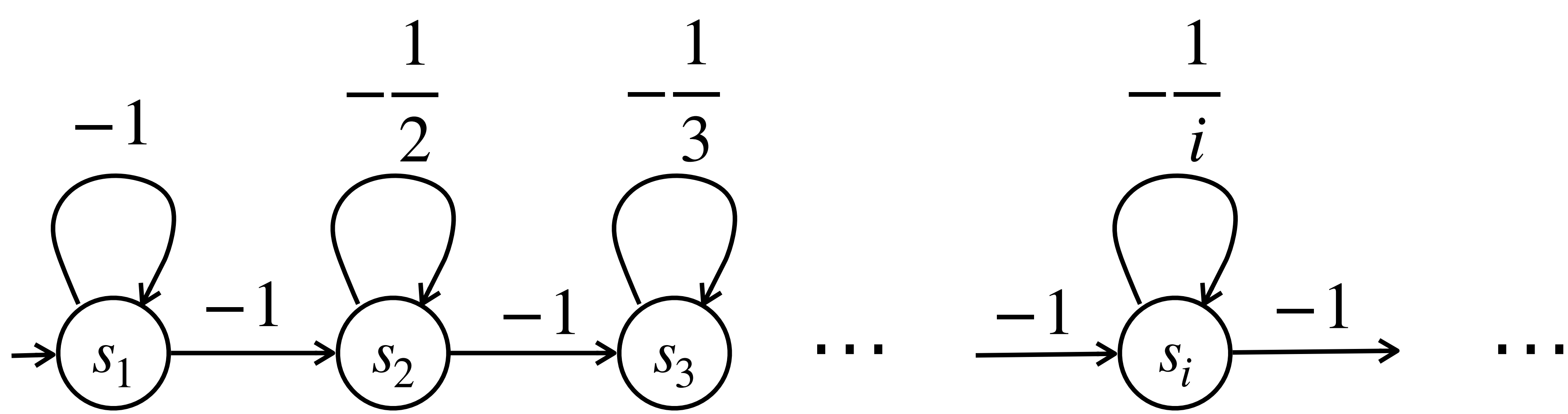
université PARIS-SACLAY · cnrs · école normale supérieure paris—saclay

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: **positional** 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 1998).

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 1998).

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 1998).

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 1998).

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 $L \subseteq C^\omega$ be a language of infinite words

Right congruence

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

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

Some language theory (2)

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

Right congruence

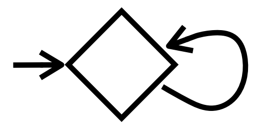
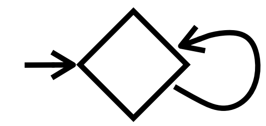
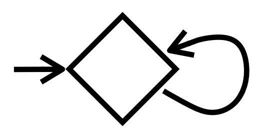
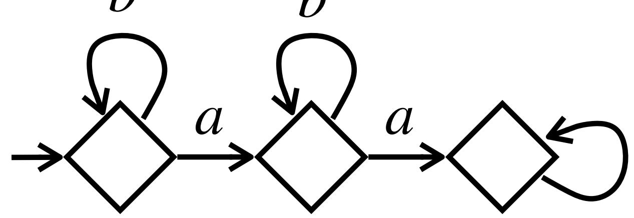
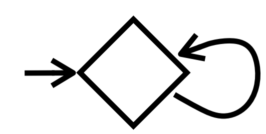
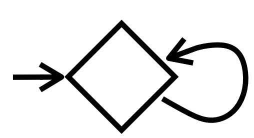
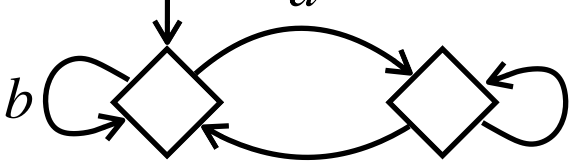
- Given $x, y \in C^*$,

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

Link with ω-regularity?

- If L is ω-regular, then $\sim_L$ has finite index;
  - The automaton based on $\sim_L$ is a so-called prefix-classifier;
- The converse does not hold (e.g. all prefix-independent languages are such that $\sim_L$ has only one element).

Four examples

Objective	Prefix classifier $\mathscr{M}_{\sim}$	Suffcient memory
Parity objective		
Mean-payoff ≥ 0		No finite automaton
$C = \{a, b\}$ $W = b^*ab^*aC^\omega$		
$C = \{a, b\}$ $W = C^*(ab)^\omega$		

Characterization

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

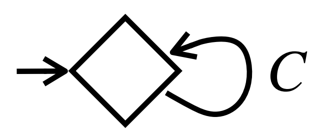
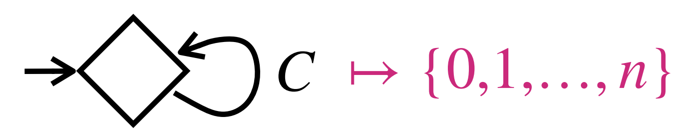
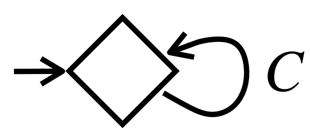
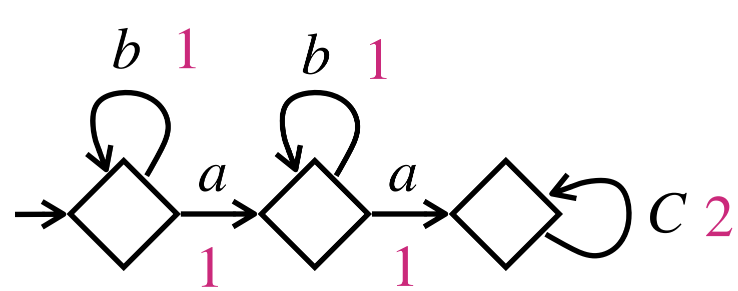
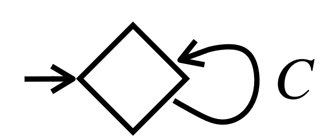
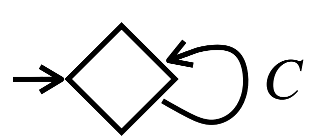
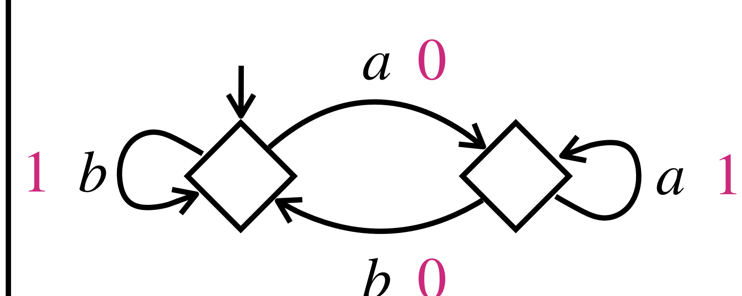
Characterization - Two-player games

If a finite memory structure $\mathcal{M}$ suffices to play optimally in one-player infinite arenas for both players, then the prefix-classifier $\mathcal{M}_\sim$ is finite and W is recognized by a parity automaton $(\mathcal{M}_\sim \otimes \mathcal{M}, \gamma)$, with $\gamma : M \times C \to \{0,1,\ldots,n\}$.

→ Generalizes [CN06] where both $\mathcal{M}$ and $\mathcal{M}_\sim$ 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).

Four examples

<table>
<tr><th>Objective</th><th>Prefix classifier $\mathscr{M}_\sim$</th><th>One-player memory</th></tr>
<tr><td>Parity objective</td><td>
</td><td>
</td></tr>
<tr><td>Mean-payoff ≥ 0</td><td>
</td><td>No finite automaton</td></tr>
<tr><td>$C = \{a, b\}$
$W = b^*ab^*aC^\omega$</td><td>
</td><td>
</td></tr>
<tr><td>$C = \{a, b\}$
$W = C^*(ab)^\omega$</td><td>
</td><td>
</td></tr>
</table>

Corollaries

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.

Corollaries

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.

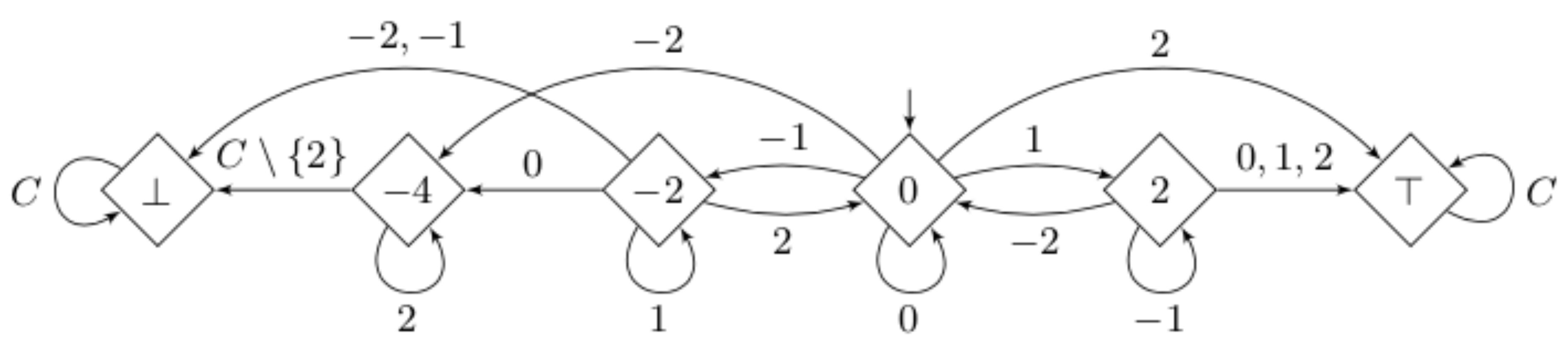
Characterization

W is finite-memory-determined in (two-player) infinite games if and only if W is ω-regular.

Some consequences

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

- Some discounted objectives are ω-regular:
e.g. condition $\mathsf{DS}_{\lambda}^{\geq 0}$ (with $\lambda \in (0,1) \cap \mathbb{Q}$, $C = [-k, k] \cap \mathbb{Z}$) is ω-regular if and only if $k < \frac{1}{\lambda} - 1$ or $\lambda = \frac{1}{n}$ for some $n \in \mathbb{N}_{>0}$



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

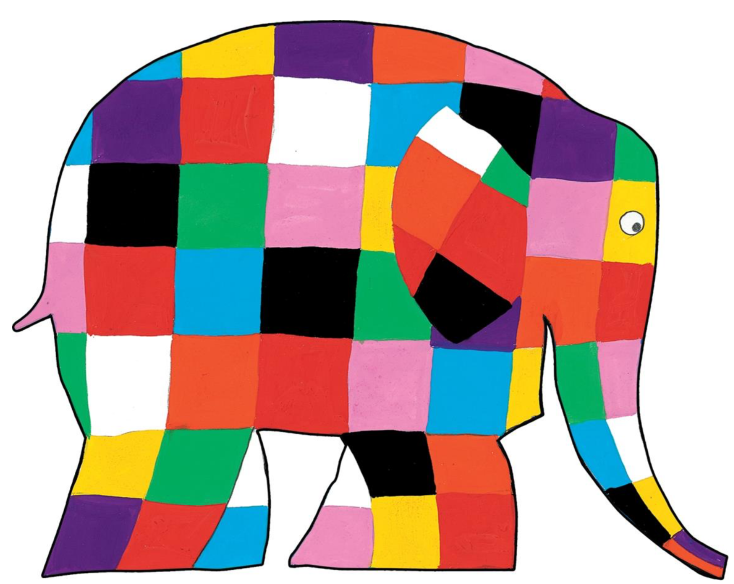
- Complete characterization of winning objectives that ensure chromatic finite-memory determinacy in infinite games = ω-regular
- One-to-two-player lift
  (requires chromatic finite memory determinacy in one-player games for both players; ensures chromatic finite memory determinacy in two-players games 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:
  - Can be reduce/optimize the memory?
    E.g. is $\mathscr{M}_{\sim}$ necessary in the memory for two players?
  - What about chaotic finite memory?
  - Can we focus on one player (so-called half-positionality)?
  - What about finite branching?

Conclusion



What you can bring home

What you can bring home

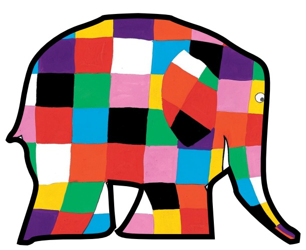
- Use of models and **concepts from game theory** in formal methods (e.g. controller in reactive systems)

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…

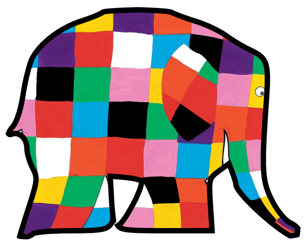
What you can bring home

- 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



- Going further:
  - Games under **partial observation**, e.g. players with their own knowledge (of the game, of the other's choices, ...)
  - Half-positionality or half-finite-memory of objectives (preliminary result [BCRV22])

[BCRV22] Bouyer, Casares, Randour, Vandenhove. Half-Positional Objectives Recognized by Deterministic Büchi Automata (submitted).