Antichains: A New Algorithm for Checking Universality of Finite Automata

Laurent Doyen
Université Libre de Bruxelles

Joint work with

Martin De Wulf, Tom Henzinger, Jean-François Raskin

CAV, Seattle, 17th August, 2006

Outline of the talk

- Motivation

- Universality - A Game Approach

- Example

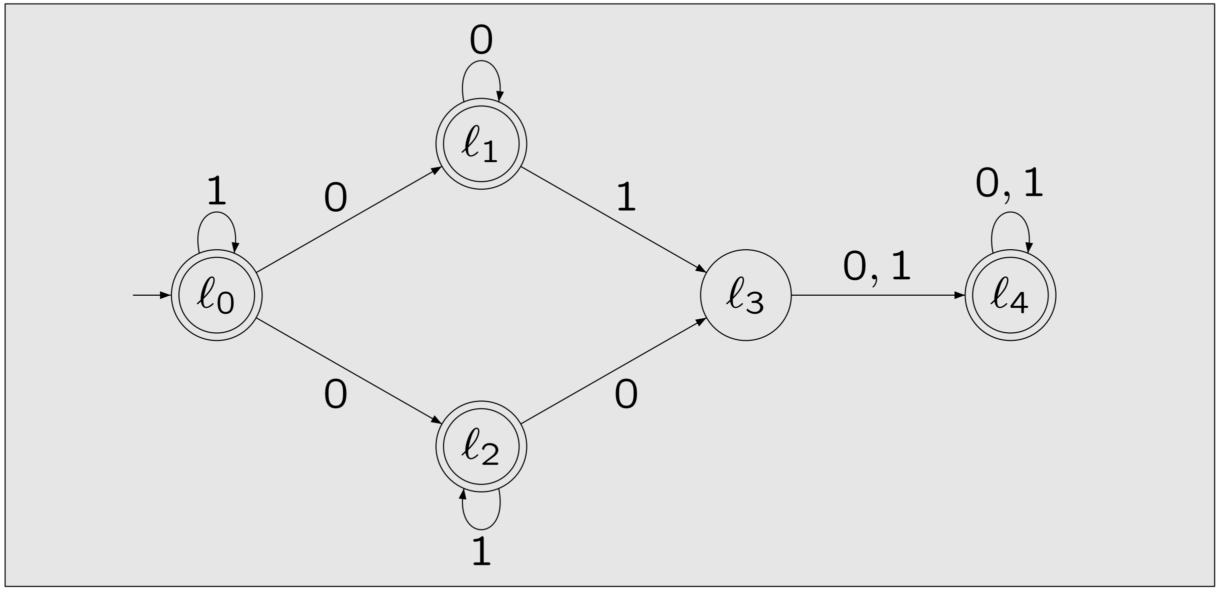
- Experimental Results

- Conclusion

Finite State Automaton

Finite automaton: $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta, F \rangle$

with $\delta : \mathsf{Loc} \times \Sigma \to 2^{\mathsf{Loc}}$ (non-deterministic)



For $w \in \Sigma^*$, we have $\begin{cases} w \in L(\mathcal{A}) \text{ iff some path on } w \text{ accepts.} \\ w \notin L(\mathcal{A}) \text{ iff all paths on } w \text{ reject.} \end{cases}$

Language Inclusion and Universality

An implementation $\mathcal{A}$ of a program is correct with regard to its specification $\mathcal{B}$ if:

$$L(\mathcal{A}) \subseteq L(\mathcal{B})$$

non-deterministic

deterministic

Language Inclusion and Universality

$$L(\mathcal{A}) \subseteq L(\mathcal{B})$$

iff $L(\mathcal{A} \cap \mathcal{B}^c)$ is empty

- Computing $\mathcal{B}^c$: hard (via determinization)
- Checking emptiness: easy

iff $L(\mathcal{A}^c \cup \mathcal{B})$ is universal

- Computing $\mathcal{A}^c$: easy
- Checking universality: hard

Language Inclusion and Universality

$$L(\mathcal{A}) \subseteq L(\mathcal{B})$$

iff $L(\mathcal{A} \cap \mathcal{B}^c)$ is empty

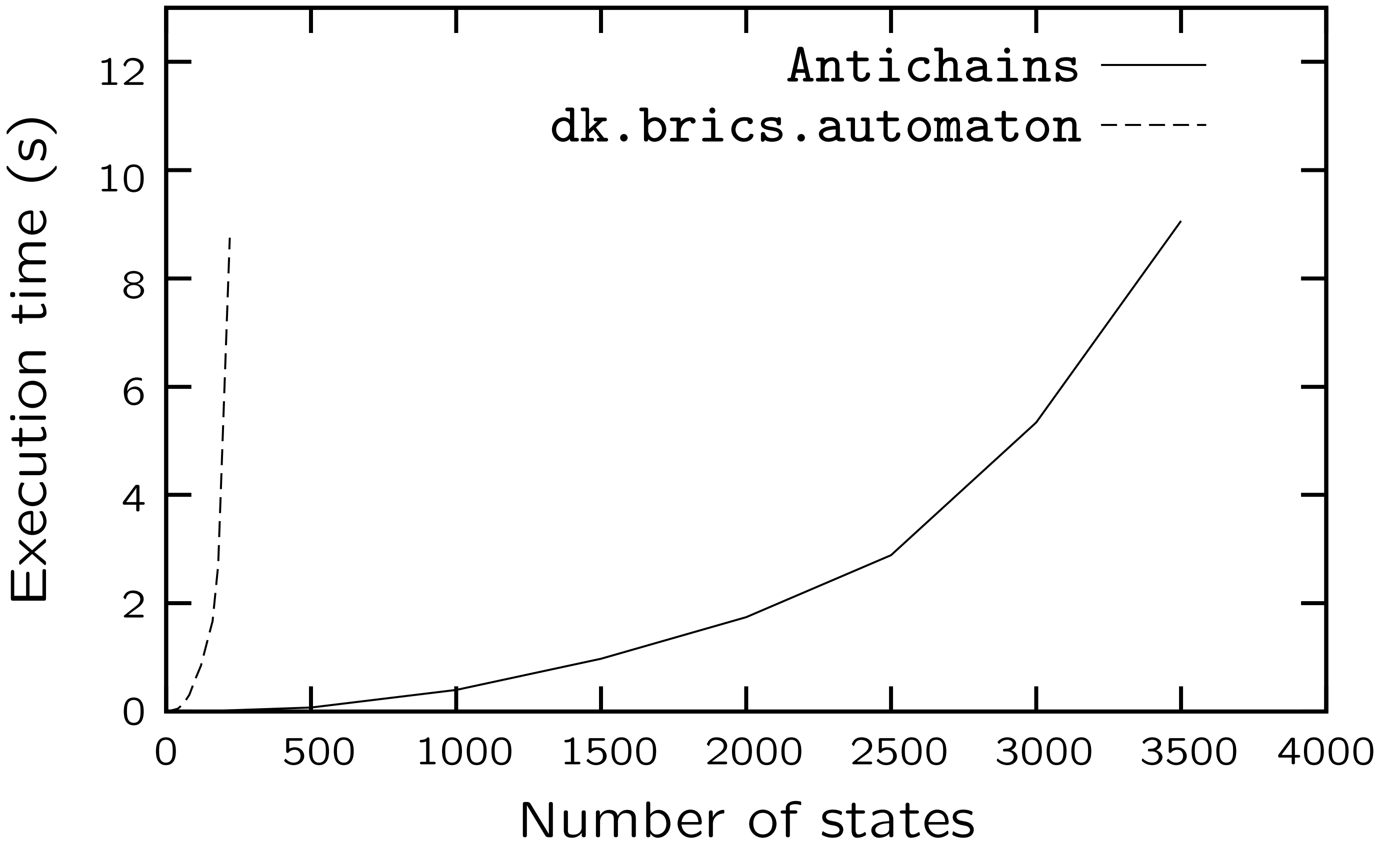
- Computing $\mathcal{B}^c$: hard (via determinization)
- Checking emptiness: easy

iff $L(\mathcal{A}^c \cup \mathcal{B})$ is universal

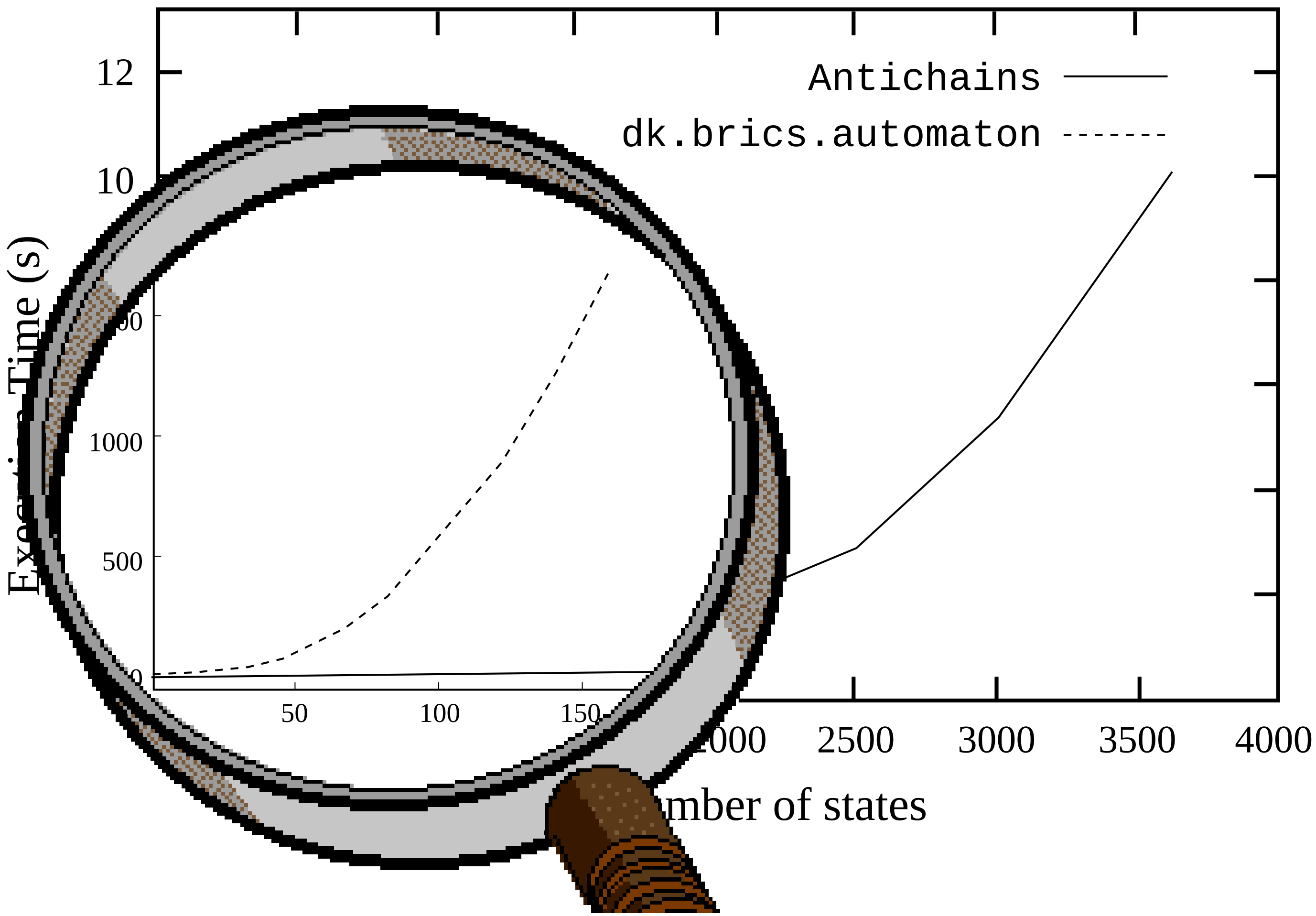
- Computing $\mathcal{A}^c$: easy
- Checking universality: hard

  not so hard in practice with antichains.

Universality - Experimental results



Universality - Experimental results



Universality - Exexution times (in milliseconds)

Number of states	20	40	60	80	100	175	500
Determinization	23	50	141	309	583	2257	-
Antichains	1	2	2	3	5	14	76

Number of states	1000	1500	2000	2500	3000	3500	4000
Determinization	-	-	-	-	-	-	-
Antichains	400	973	1741	2886	5341	9063	13160

Outline of the talk

- Motivation
- Universality - A Game Approach
- Example
- Experimental Results
- Conclusion

Universality - A game approach

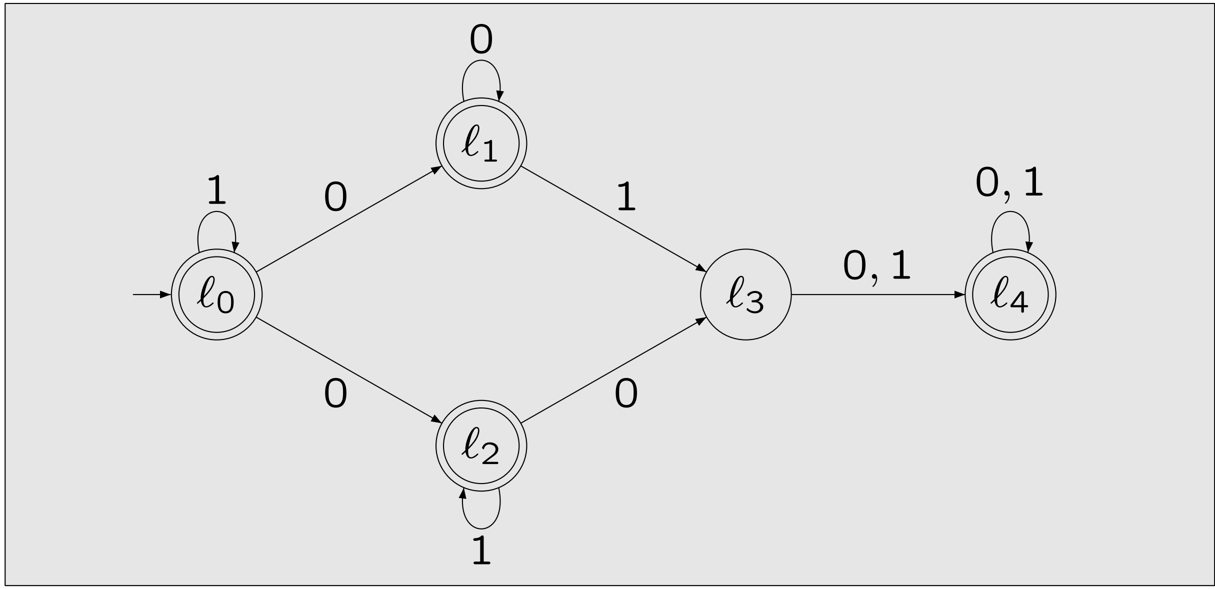
Consider a game played by a protagonist and an antagonist

The protagonist wants to establish that $\mathcal{A}$ is not universal.

The protagonist has to provide a finite word w such that no matter how the antagonist reads it using $\mathcal{A}$, the automaton ends up in a rejecting location.

$\Longrightarrow$ This is a one-shot game.

Universality - A game approach



an antagonist

not universal.

such that no

e automaton

Example: Protagonist: $w = 101$

Antagonist: $\pi = \ell_0 \xrightarrow{1} \ell_0 \xrightarrow{0} \ell_2 \xrightarrow{1} \ell_2$

Antagonist wins the play since ℓ_2 is accepting.

Universality - A game approach

Consider a game played by a protagonist and an antagonist

The protagonist wants to establish that $\mathcal{A}$ is not universal.

The protagonist has to provide a finite word w such that no matter how the antagonist reads it using $\mathcal{A}$, the automaton ends up in a rejecting location.

$\Longrightarrow$ This is a one-shot game.

Protagonist has a strategy to win this game iff $\mathcal{A}$ is not universal

Universality - A game approach

Consider a game played by a protagonist and an antagonist

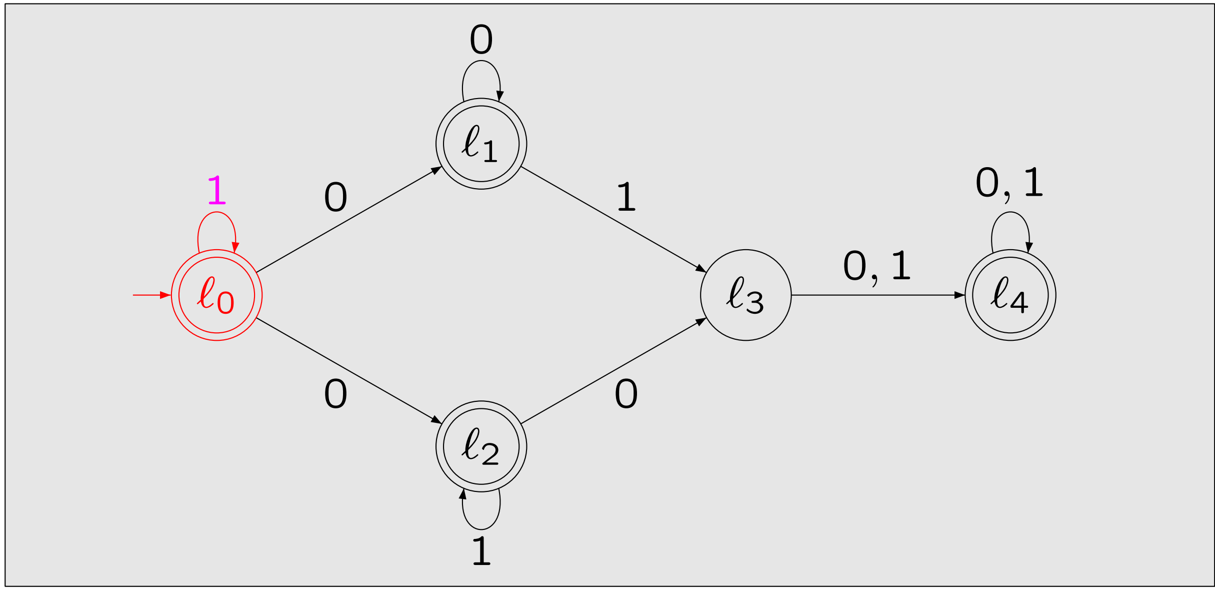
The protagonist wants to establish that $\mathcal{A}$ is not universal.

The game is turn-based:

- Protagonist provides a word w one letter at a time;
- Antagonist updates the state of $\mathcal{A}$ accordingly.

Universality - A game approach

Consider a game played by a protagonist and an antagonist



not universal.

at a time;

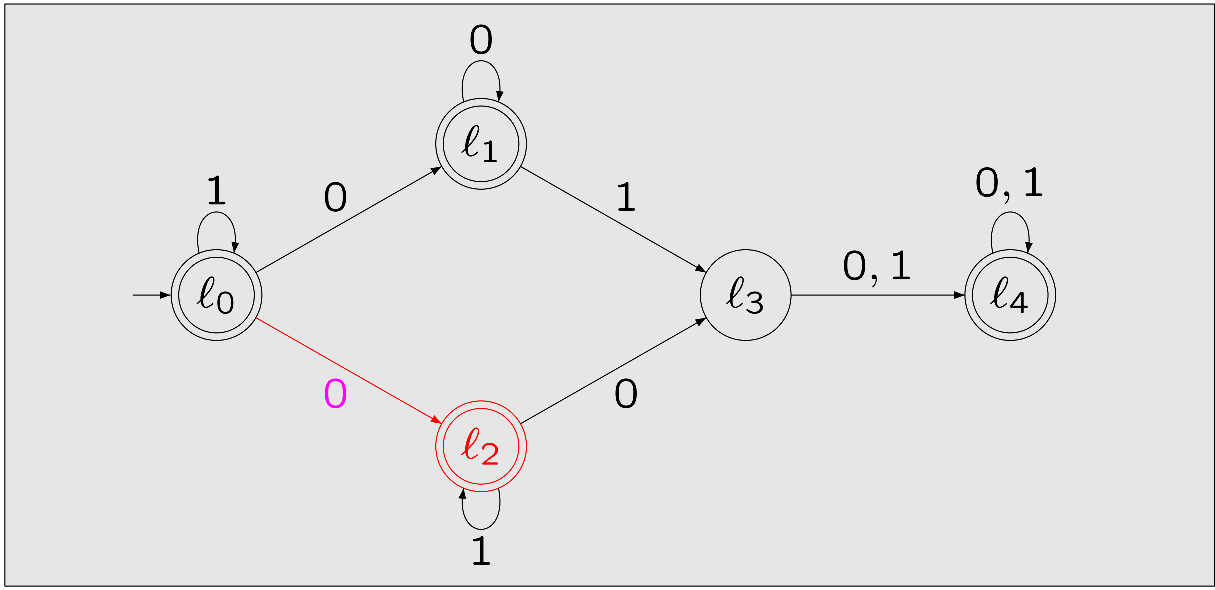
dingly.

Example: Protagonist: $w = 1$

Antagonist: $\pi = \ell_0 \xrightarrow{1} \ell_0$

Universality - A game approach

Consider a game played by a protagonist and an antagonist



not universal.

at a time;

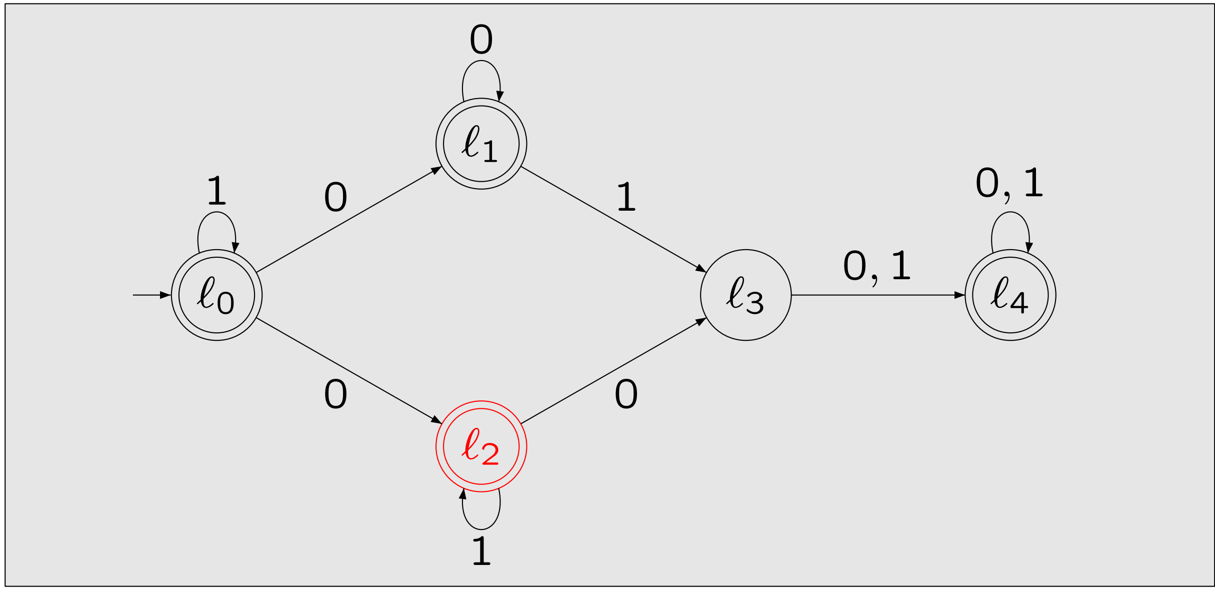
dingly.

Example: Protagonist: $w = 10$

Antagonist: $\pi = \ell_0 \xrightarrow{1} \ell_0 \xrightarrow{0} \ell_2$

Universality - A game approach

Consider a game played by a protagonist and an antagonist



not universal.

at a time;

dingly.

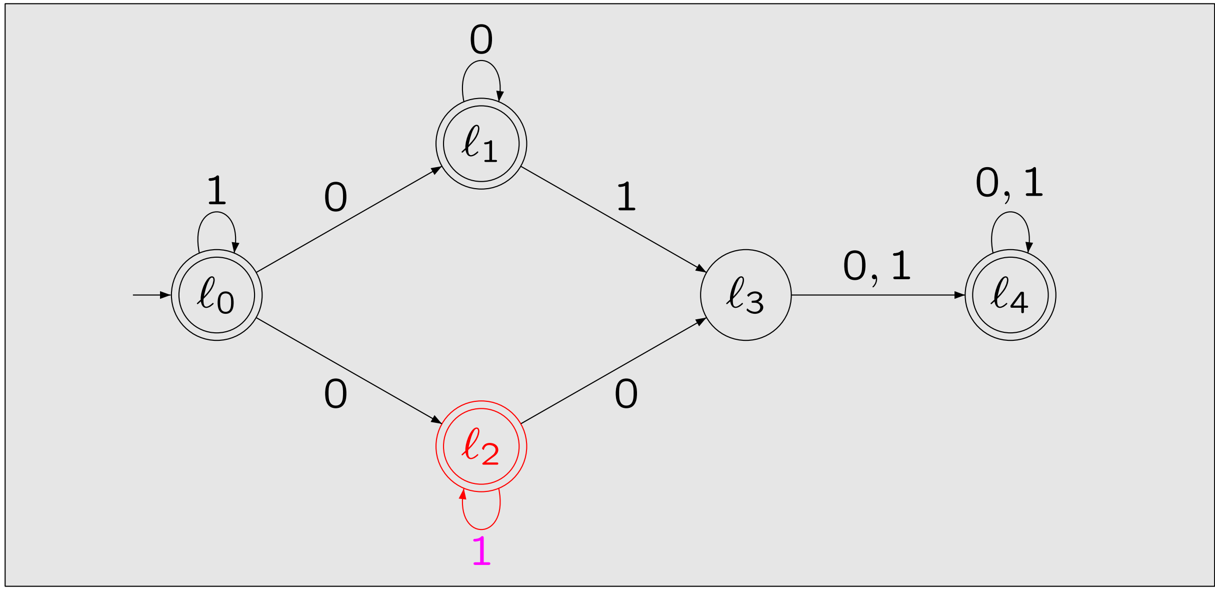
Example: Protagonist: $w = 10$

Antagonist: $\pi = \boxed{?} \xrightarrow{1} \boxed{?} \xrightarrow{0} \boxed{?}$

$\{\ell_0\}$ $\{\ell_0\}$ $\{\ell_1, \ell_2\}$

Universality - A game approach

Consider a game played by a protagonist and an antagonist



not universal.

at a time;

dingly.

Example: Protagonist: $w = 101$

Antagonist: $\pi = \boxed{?} \xrightarrow{1} \boxed{?} \xrightarrow{0} \boxed{?} \xrightarrow{1} \ell_2$

Antagonist wins the play since ℓ_2 is accepting.

Universality - A game approach

Consider a game played by a protagonist and an antagonist

The protagonist wants to establish that $\mathcal{A}$ is not universal.

The game is turn-based:

- Protagonist provides a word w one letter at a time;
- Antagonist updates the state of $\mathcal{A}$ accordingly.

The protagonist cannot observe the state chosen by the antagonist.

$\Longrightarrow$ This is a blind game (or game of null information).

Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

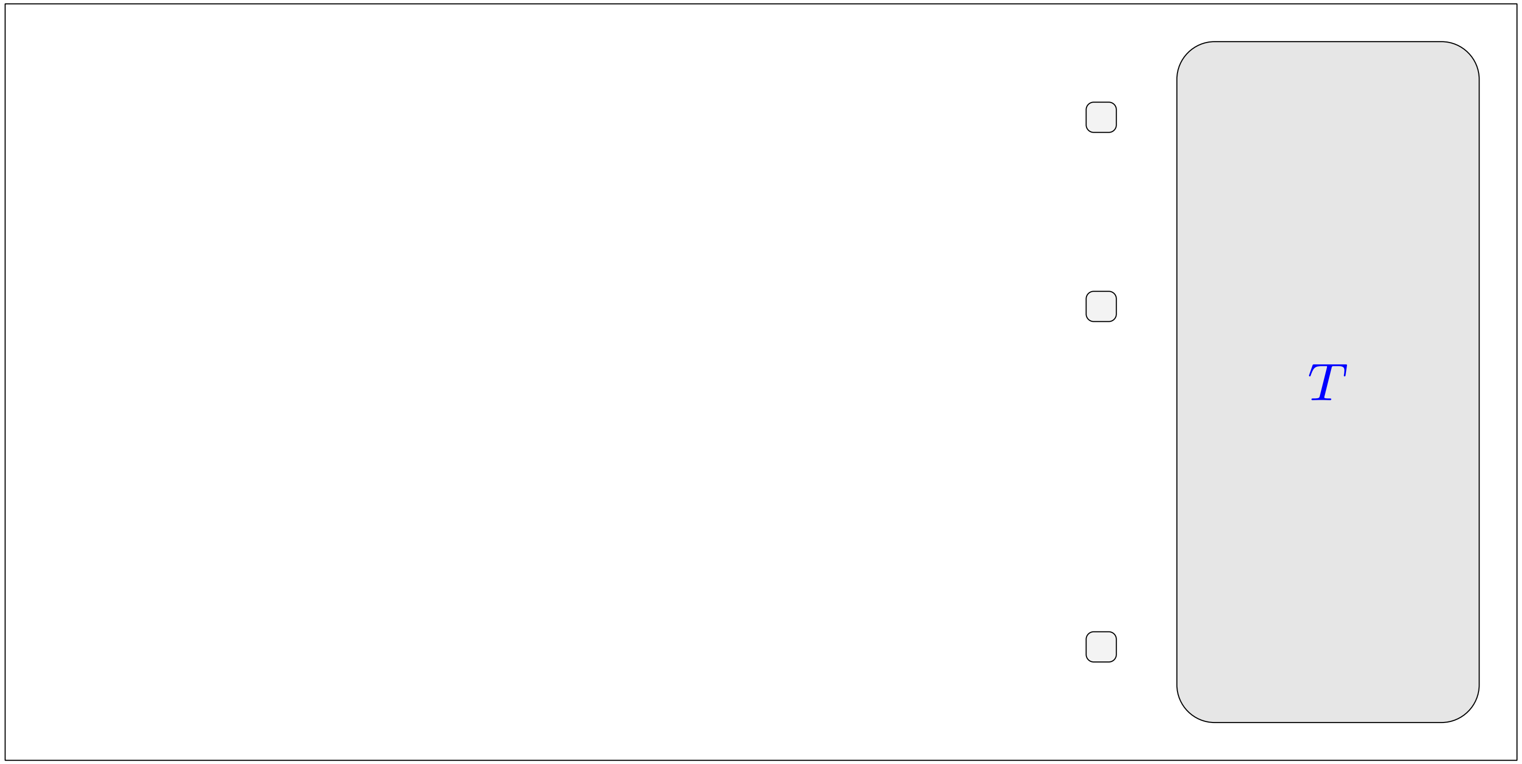
Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

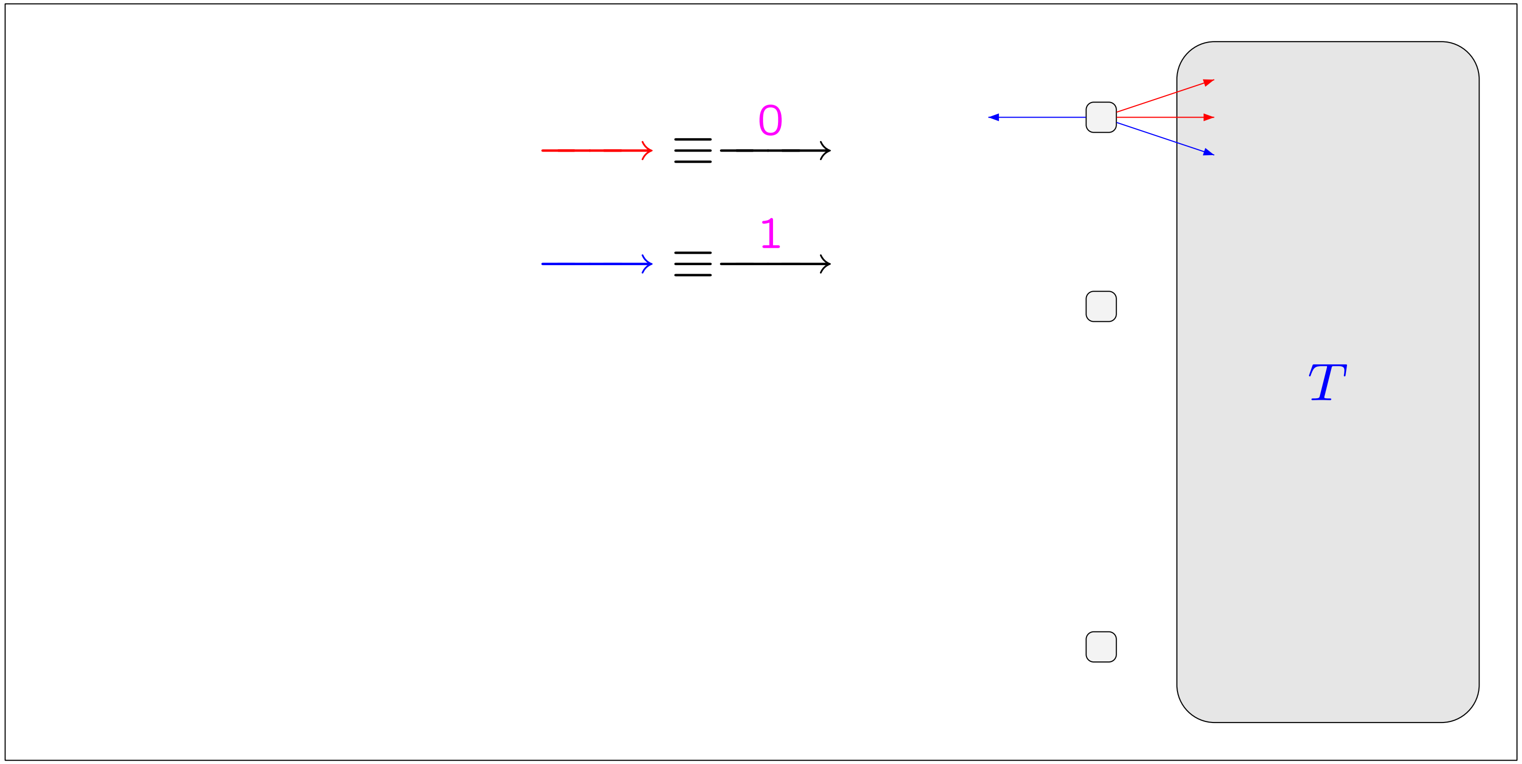


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

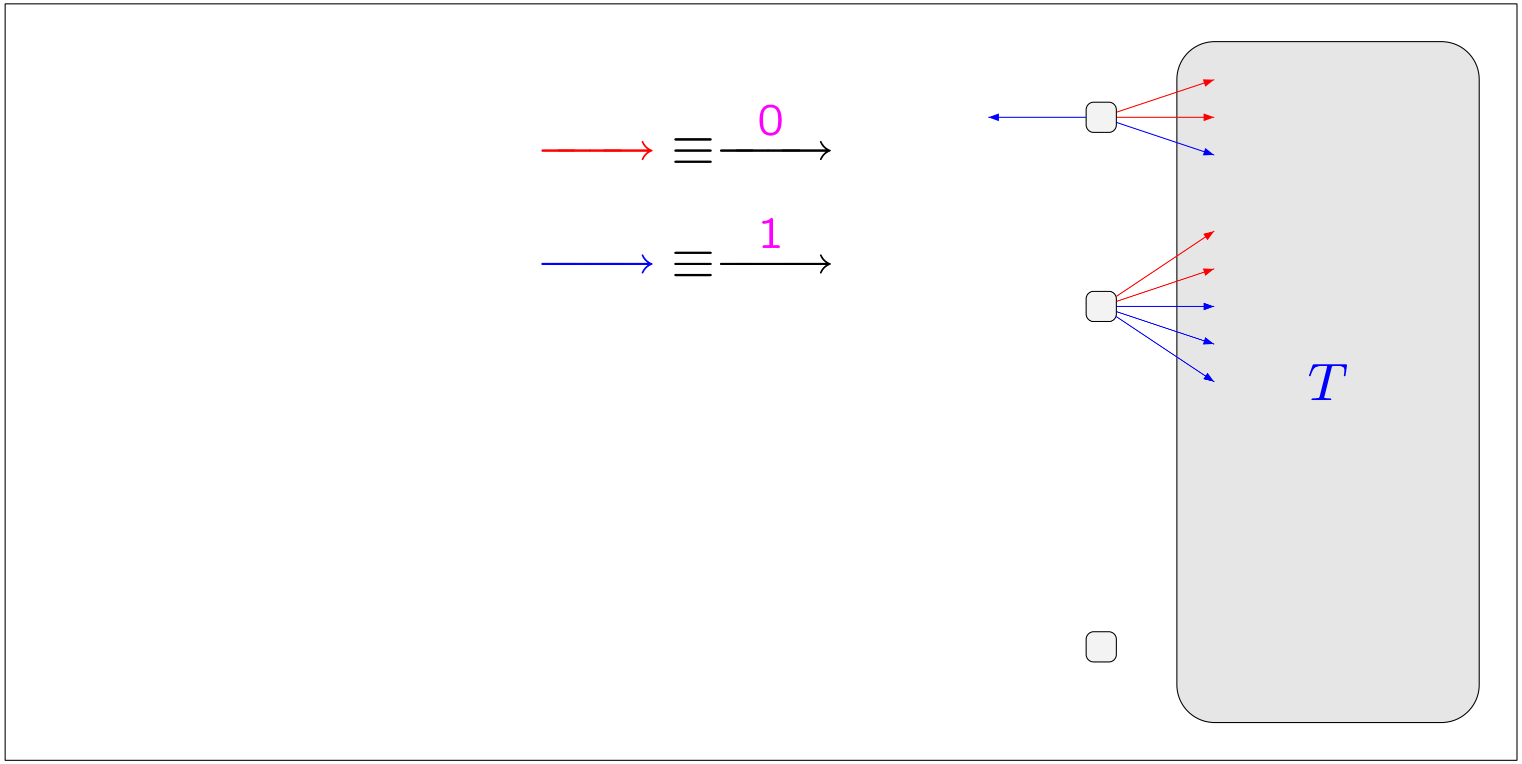


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

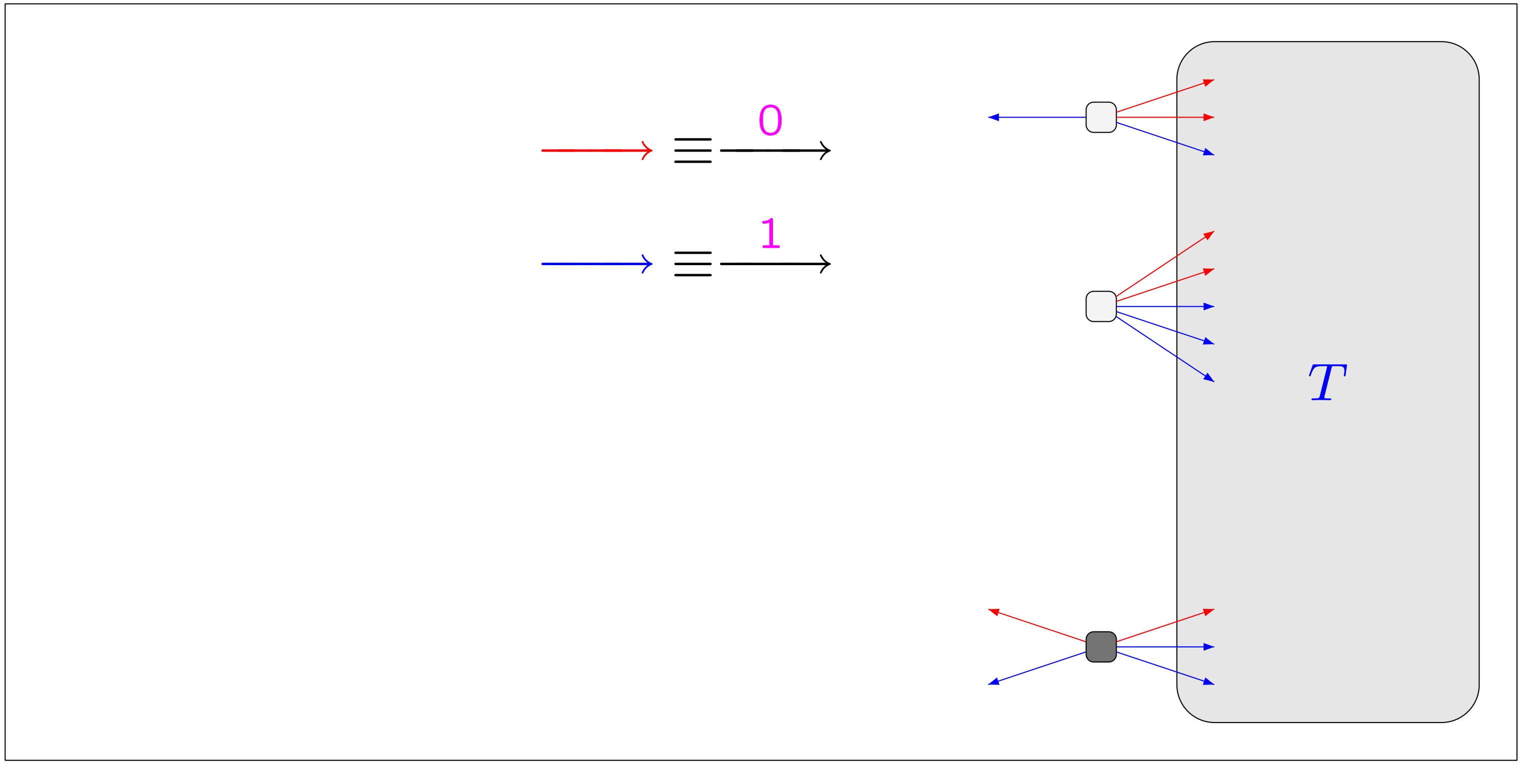


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

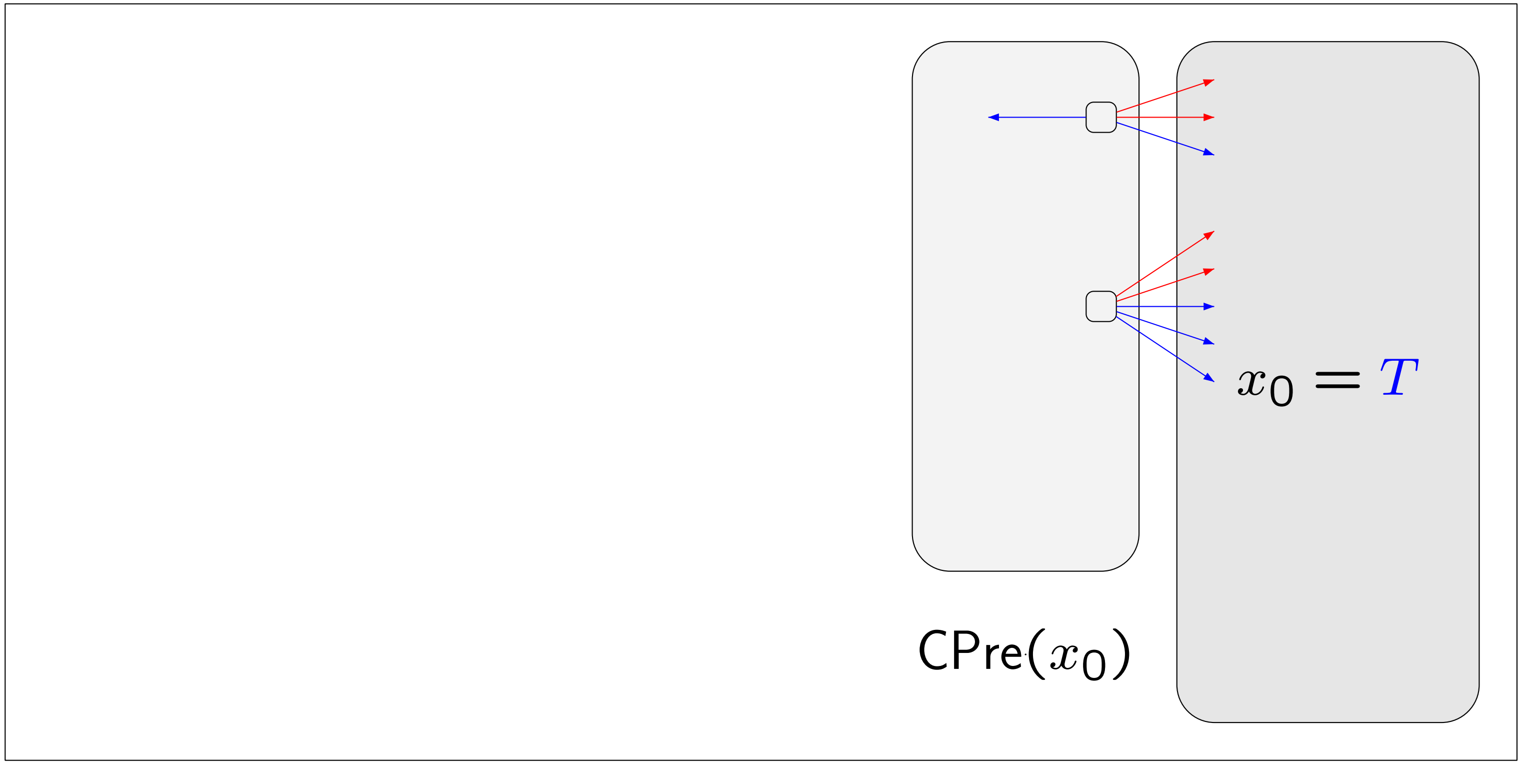


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

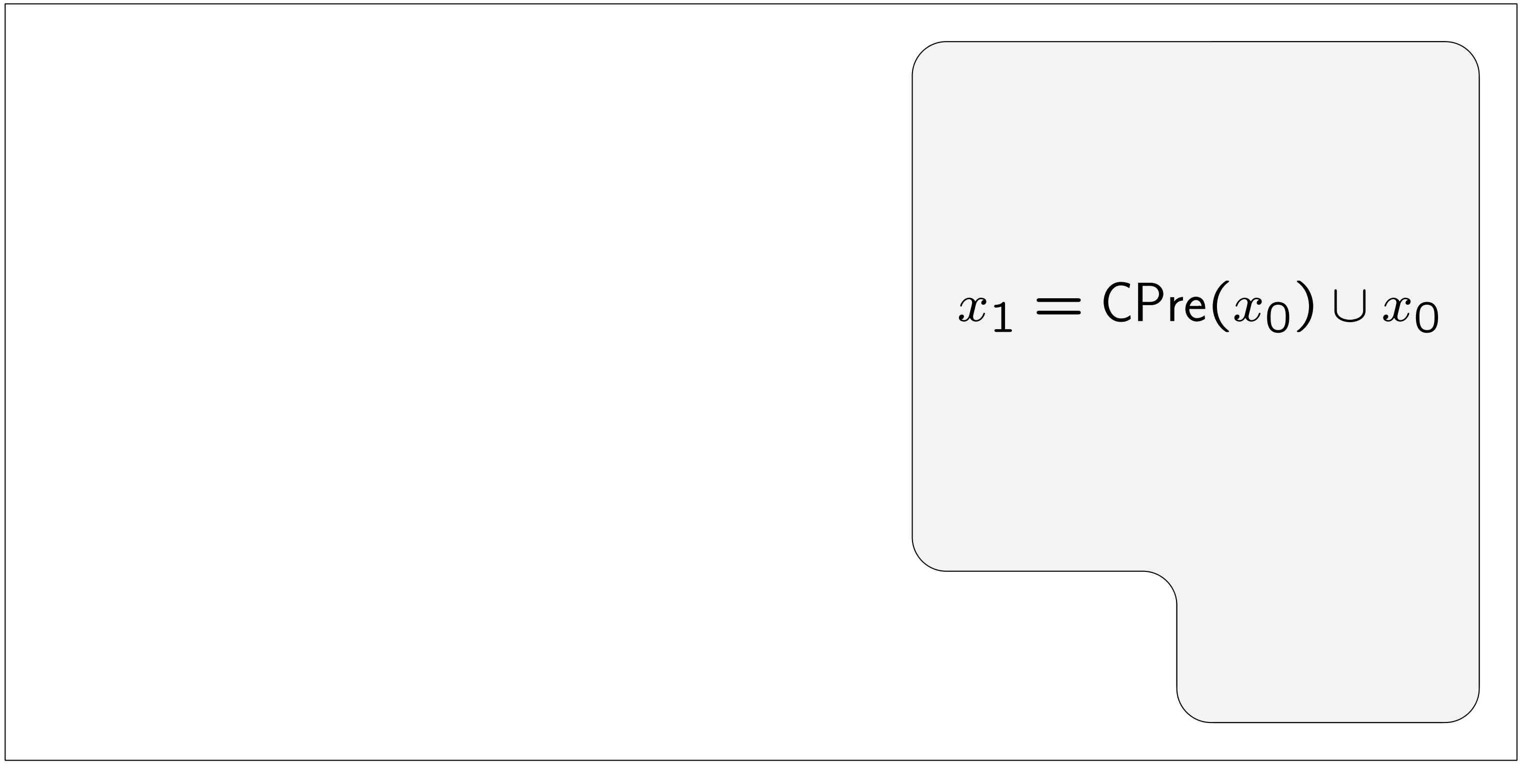


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

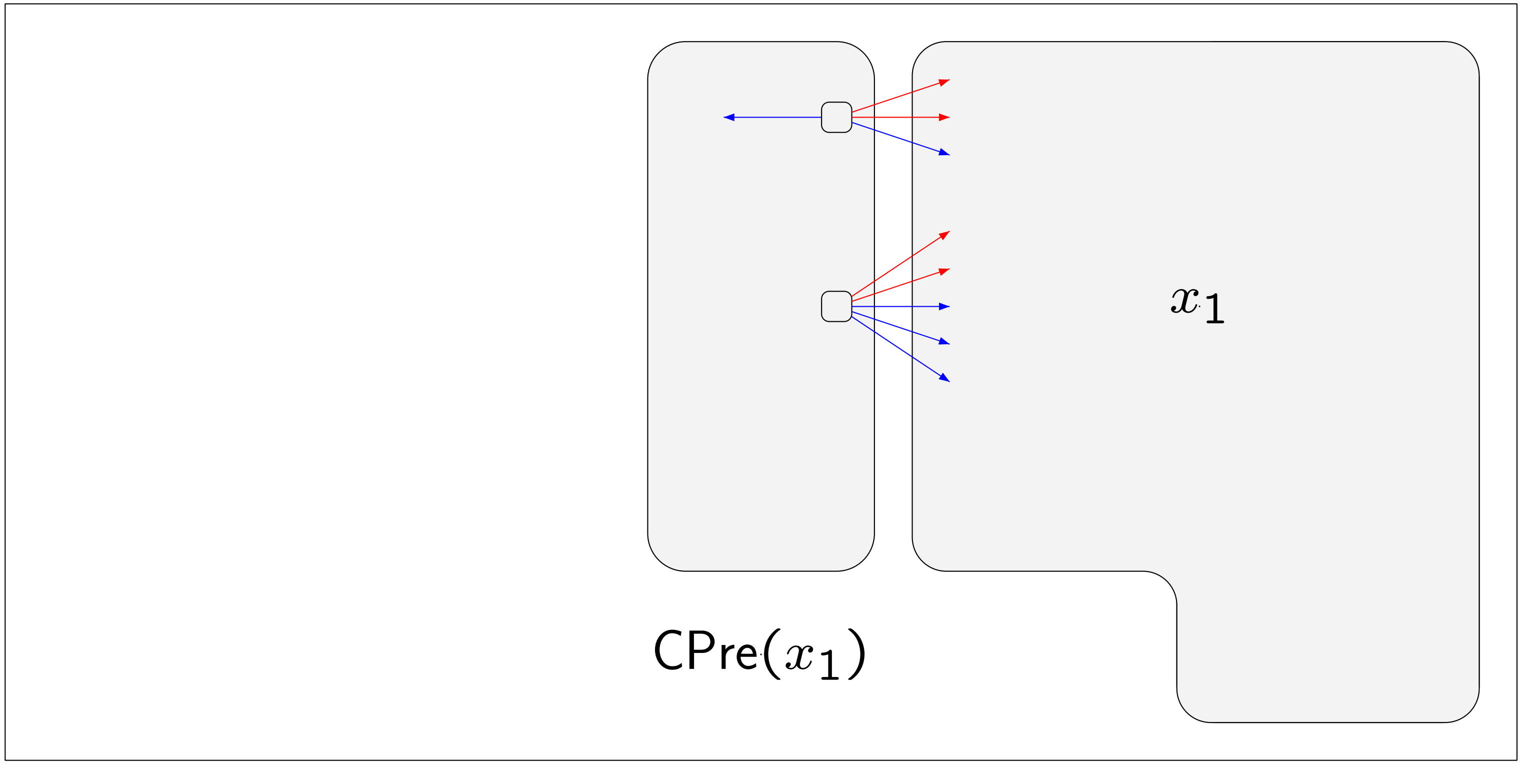


Universality - A game approach

Let $\mathcal{A} = \langle \text{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \text{Loc} \backslash F$.

Recipe for solving classical reachability games

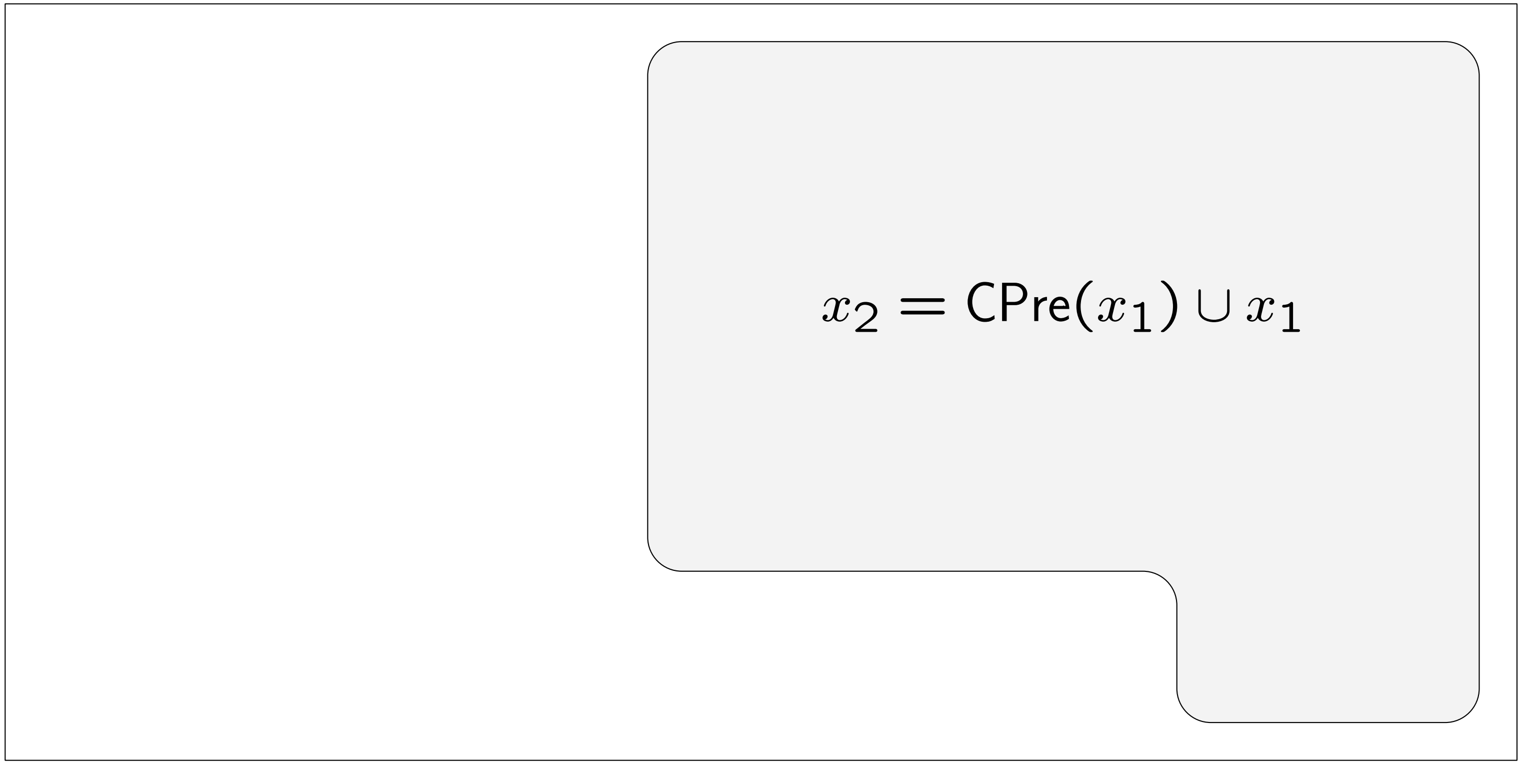


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

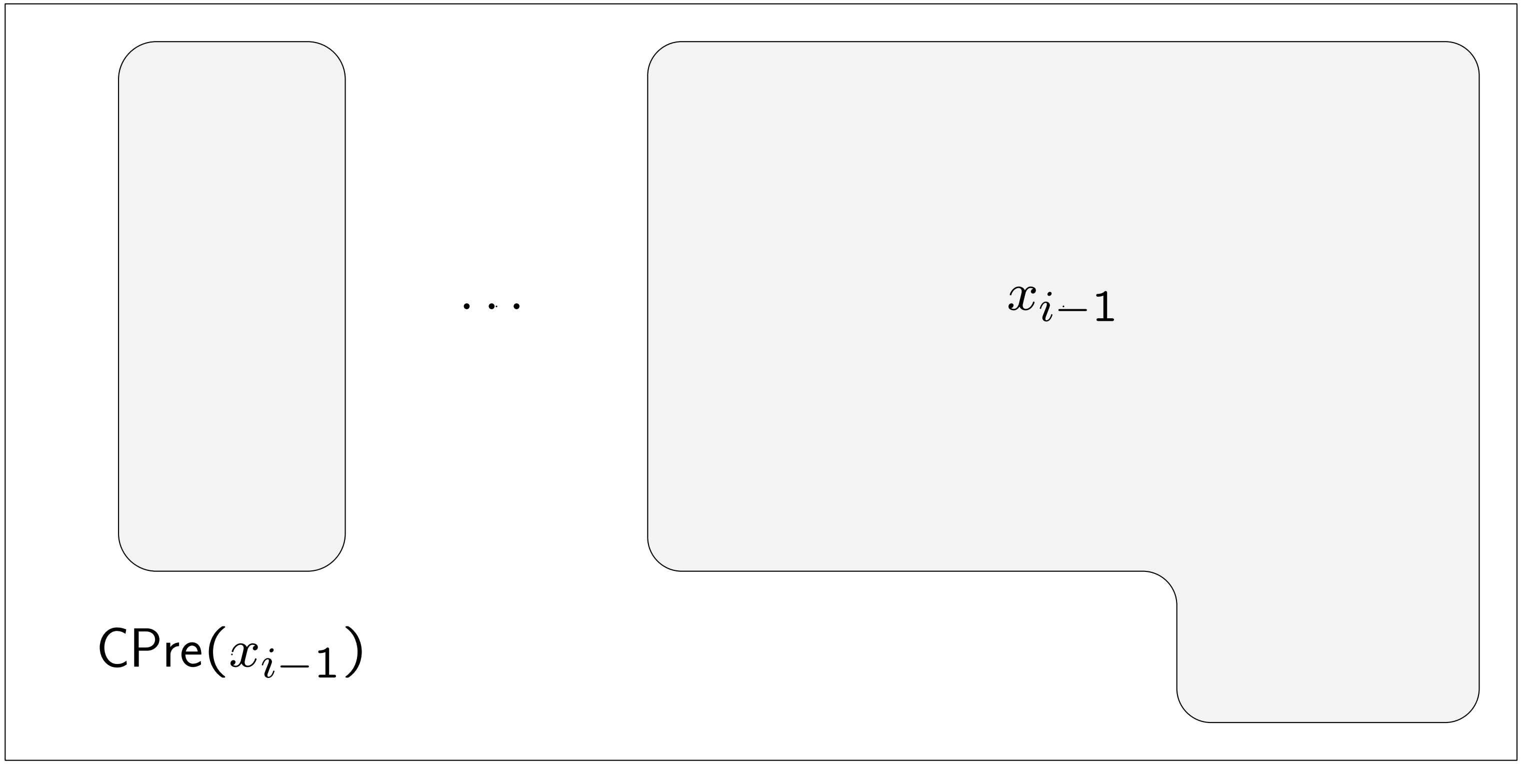


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

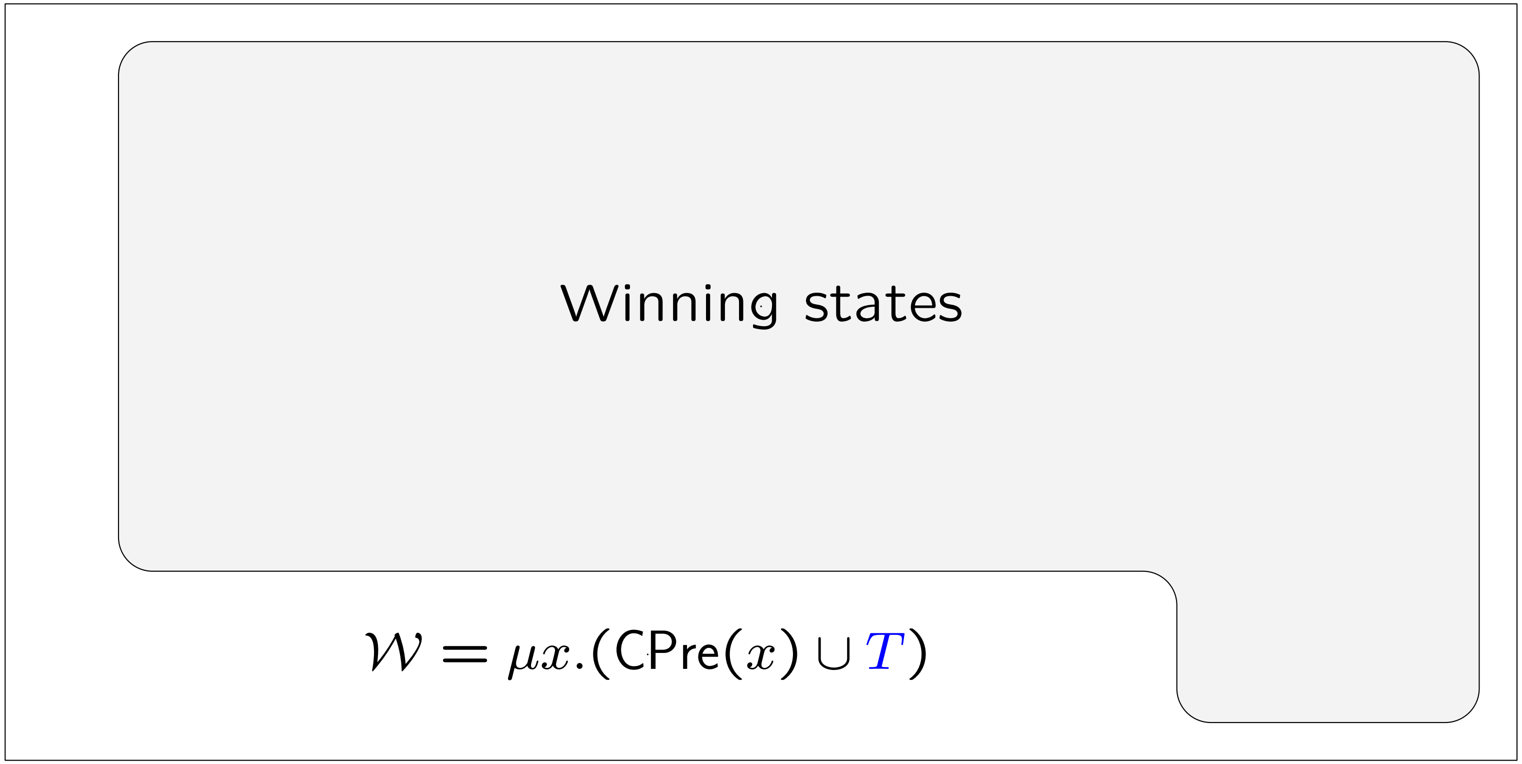


Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Checking universality of $\mathcal{A}$ is equivalent to solving a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games



Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Universality of $\mathcal{A}$ is equivalent to a blind reachability game G_T with target $T = \mathsf{Loc} \backslash F$.

Recipe for solving classical reachability games

1. Compute the set of states that are winning in one move: $\mathsf{CPre}(T)$
2. Iterate $\mathsf{CPre}(\cdot)$: compute $\mathcal{W} = \mu x.(\mathsf{CPre}(x) \cup T)$
3. Check whether $\ell_I \in \mathcal{W}$

Universality - Controllable predecessor operator

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

- $\mathsf{CPre}(\cdot)$ should encode the blindness of the game:

  "The knowledge of the protagonist is a set of states."

- $\mathsf{CPre}(T)$ contains all the set of states s such that:

  there exists $\sigma \in \Sigma$ such that:
  if protagonist plays σ from s, then the set T is reached no matter the antagonist's move.

$$\exists \sigma \in \Sigma \cdot \underbrace{\forall \ell \in s : \delta_A(\ell, \sigma) \subseteq T}_{\mathsf{post}_\sigma(s) \subseteq T}$$

Universality - Controllable predecessor operator

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Consider the following controllable predecessor operator over sets of sets of locations. For $q \subseteq 2^{\mathsf{Loc}}$, let:

$$\mathsf{CPre}(q) = \left\{ s \mid \exists s' \in q \cdot \exists \sigma \in \Sigma : \mathsf{post}_\sigma(s) \subseteq s' \right\}$$

So $s \in \mathsf{CPre}(q)$ if there is a set $s' \in q$ that is reached from any location in s, reading input letter σ.

$\Longrightarrow$ CPre encodes the blindness of the game.

Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Theorem:

$\{\ell_I\} \in \mu x.(\mathsf{CPre}(x) \cup \{T\})$

iff

Protagonist has a strategy to win G_T

iff

$\mathcal{A}$ is not universal

Claim: For $s_1 \subseteq s_2$, if $\underbrace{\mathsf{post}_\sigma(s_2) \subseteq s'}_{s_2 \in \mathsf{CPre}(\cdot)}$ then $\underbrace{\mathsf{post}_\sigma(s_1) \subseteq s'}_{s_1 \in \mathsf{CPre}(\cdot)}$

Hence, we compute $\subseteq$-downward-closed sets of state sets.

Idea: Keep in $\mathsf{CPre}(x)$ only the maximal elements.

Universality - A game approach

Let $\mathcal{A} = \langle \mathsf{Loc}, \ell_I, \Sigma, \delta_A, F \rangle$.

Definition:

For $q \subseteq 2^{\mathsf{Loc}}$, let:

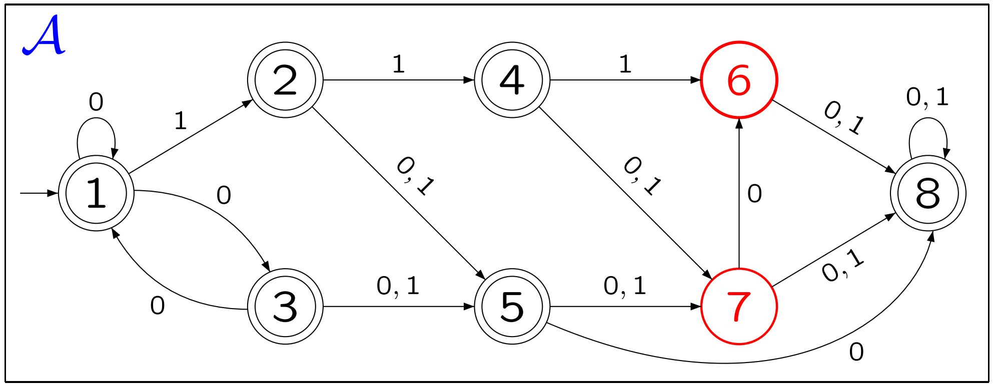
$$\mathsf{CPre}(q) = \mathsf{MaximalSets}(\{s \mid \exists s' \in q \cdot \exists \sigma \in \Sigma : \mathsf{post}_\sigma(s) \subseteq s'\})$$
$$= \lceil \{s \mid \exists s' \in q \cdot \exists \sigma \in \Sigma : \mathsf{post}_\sigma(s) \subseteq s'\} \rceil$$

where $\lceil q \rceil = \{s \in q \mid \nexists s' \in q : s \subset s'\}$ is an antichain of sets of locations (containing only pairwise $\subseteq$-incomparable elements).

Outline of the talk

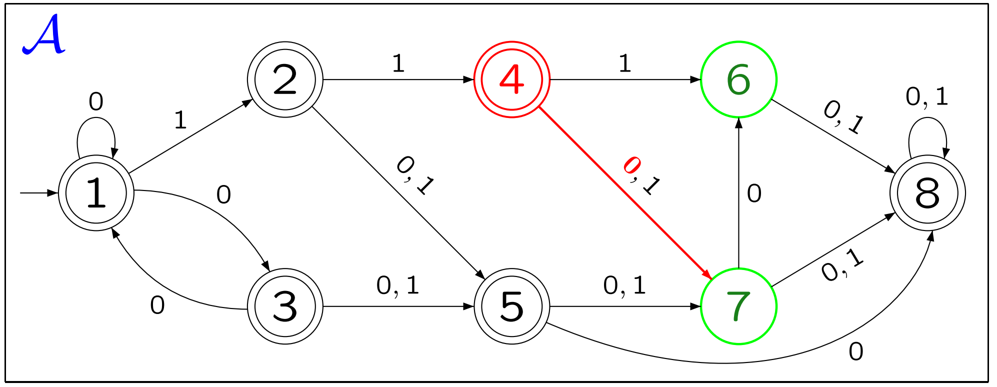
- Motivation
- Universality - A Game Approach
- Example
- Experimental Results
- Conclusion

Universality - Example



$$x_0 = \quad T \quad = \left\{ \{6, 7\} \right\}$$

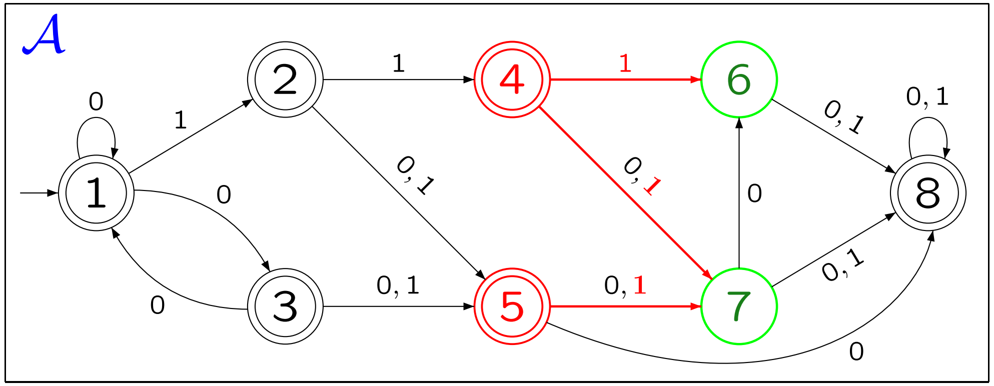
Universality - Example



$$x_0 = \quad T \quad = \left\{ \{6, 7\} \right\}$$

$$x_1 = \quad \mathsf{CPre}(x_0) \cup T \quad = \Big\lceil \Big\{ \{4\}_0,$$

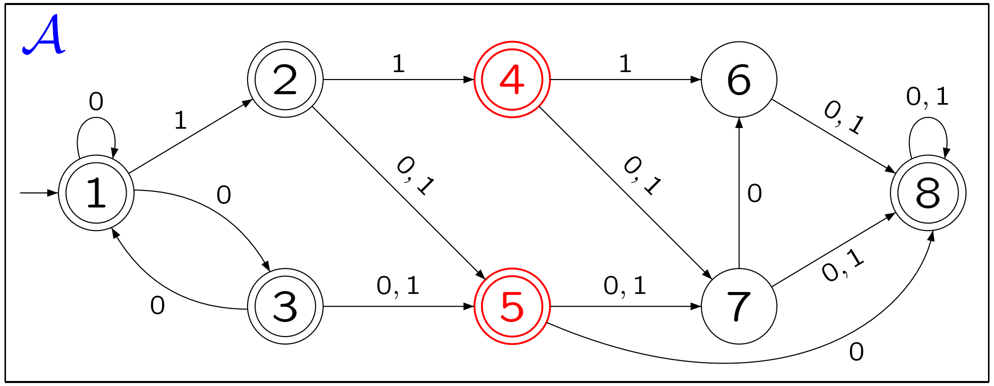
Universality - Example



$$x_0 = \qquad T \qquad = \big\{\{6,7\}\big\}$$

$$x_1 = \quad \mathsf{CPre}(x_0) \cup \{T\} \quad = \Big\lceil \big\{\{4\}_{0,1}, \{4,5\}_1, \{5\}_1, \emptyset\big\} \Big\rceil \cup \big\{\{6,7\}\big\}$$

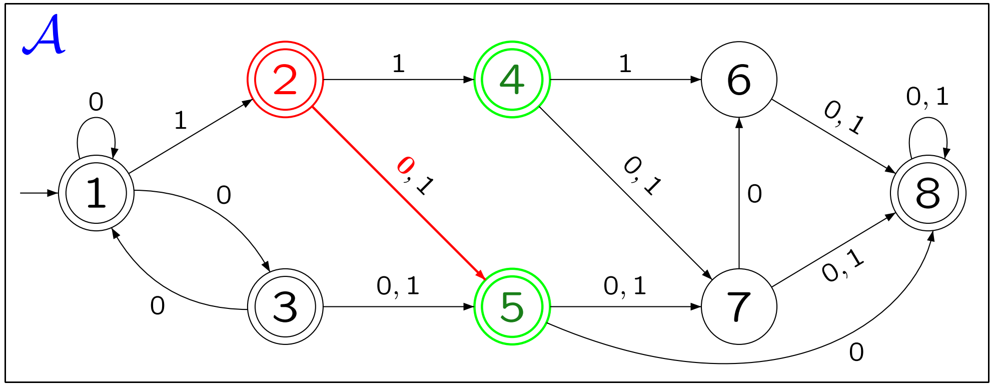
Universality - Example



$$x_0 = \quad T \quad = \{\{6, 7\}\}$$

$$x_1 = \quad \mathsf{CPre}(x_0) \cup \{T\} \quad = \{\{6, 7\}, \{4, 5\}\}$$

Universality - Example

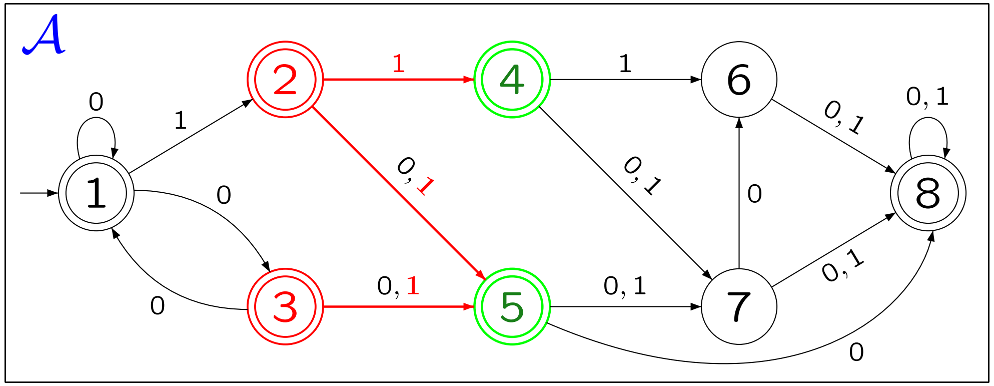


$$x_0 = \quad T \quad = \Big\{\{6, 7\}\Big\}$$

$$x_1 = \quad \mathsf{CPre}(x_0) \cup \{T\} \quad = \Big\{\{6, 7\}, \{4, 5\}\Big\}$$

$$x_2 = \quad \mathsf{CPre}(x_1) \cup \{T\} \quad = \Big\lceil \Big\{\{4, 5\}, \{2\}_0,$$

Universality - Example

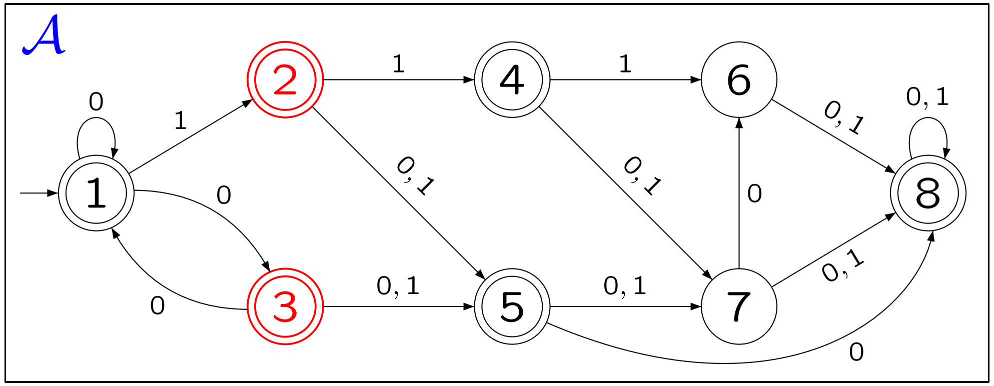


$$x_0 = \quad T \quad = \Big\{\{6, 7\}\Big\}$$

$$x_1 = \quad \mathsf{CPre}(x_0) \cup \{T\} \quad = \Big\{\{6, 7\}, \{4, 5\}\Big\}$$

$$x_2 = \quad \mathsf{CPre}(x_1) \cup \{T\} \quad = \Big\lceil \Big\{\{4, 5\}, \{2\}_{0,1}, \{2, 3\}_1, \{3\}_1, \emptyset\Big\} \Big\rceil \cup \Big\{\{6, 7\}\Big\}$$

Universality - Example

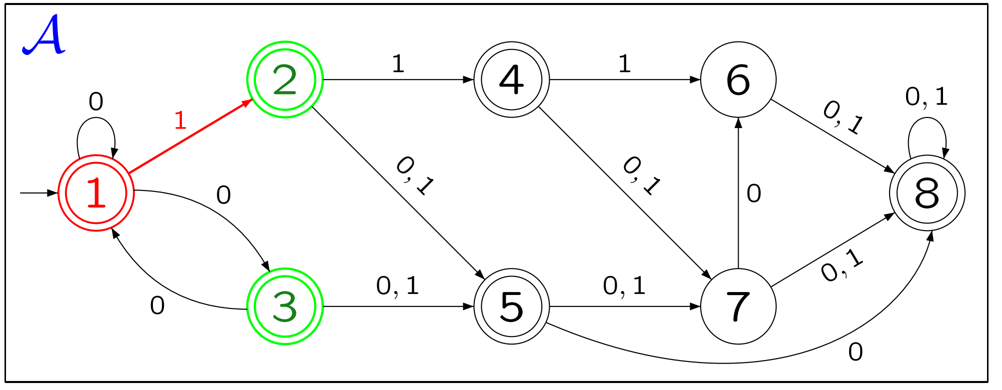


$$x_0 = \quad T \quad = \Big\{\{6,7\}\Big\}$$

$$x_1 = \quad \mathsf{CPre}(x_0) \cup \{T\} \quad = \Big\{\{6,7\},\{4,5\}\Big\}$$

$$x_2 = \quad \mathsf{CPre}(x_1) \cup \{T\} \quad = \Big\{\{6,7\},\{4,5\},\{2,3\}\Big\}$$

Universality - Example



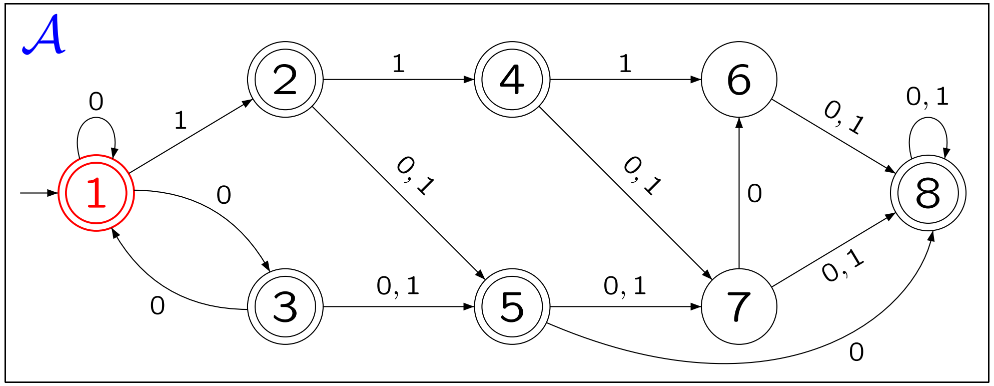
$$x_0 = \quad T \quad = \Big\{\{6,7\}\Big\}$$

$$x_1 = \quad \mathsf{CPre}(x_0) \cup \{T\} \quad = \Big\{\{6,7\},\{4,5\}\Big\}$$

$$x_2 = \quad \mathsf{CPre}(x_1) \cup \{T\} \quad = \Big\{\{6,7\},\{4,5\},\color{green}{\{2,3\}}\Big\}$$

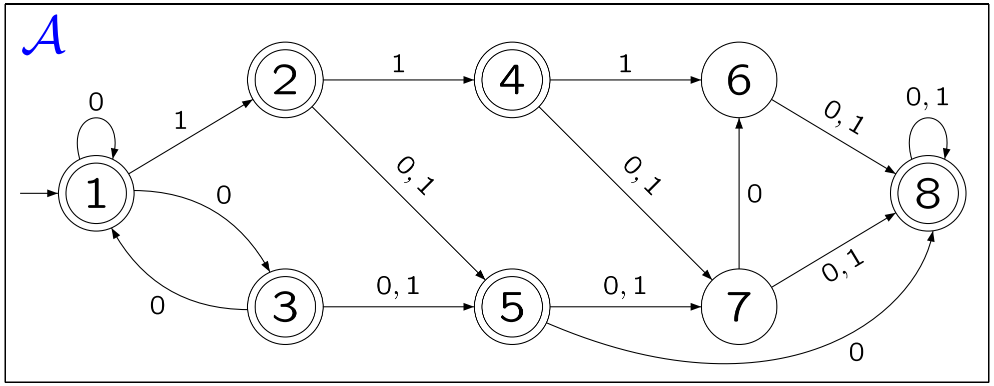
$$x_3 = \quad \mathsf{CPre}(x_2) \cup \{T\} \quad = \Big\lceil \Big\{\{4,5\},\{2,3\},\color{red}{\{1\}_1},\emptyset\Big\} \Big\rceil \cup \Big\{\{6,7\}\Big\}$$

Universality - Example



$$
\begin{array}{lcl}
x_0 = & T & = \big\{\{6,7\}\big\} \\
x_1 = & \mathsf{CPre}(x_0) \cup \{T\} & = \big\{\{6,7\},\{4,5\}\big\} \\
x_2 = & \mathsf{CPre}(x_1) \cup \{T\} & = \big\{\{6,7\},\{4,5\},\{2,3\}\big\} \\
x_3 = & \mathsf{CPre}(x_2) \cup \{T\} & = \big\{\{6,7\},\{4,5\},\{2,3\},\{1\}\big\}
\end{array}
$$

Universality - Example



$$x_0 = \quad T \quad = \Big\{\{6,7\}\Big\}$$

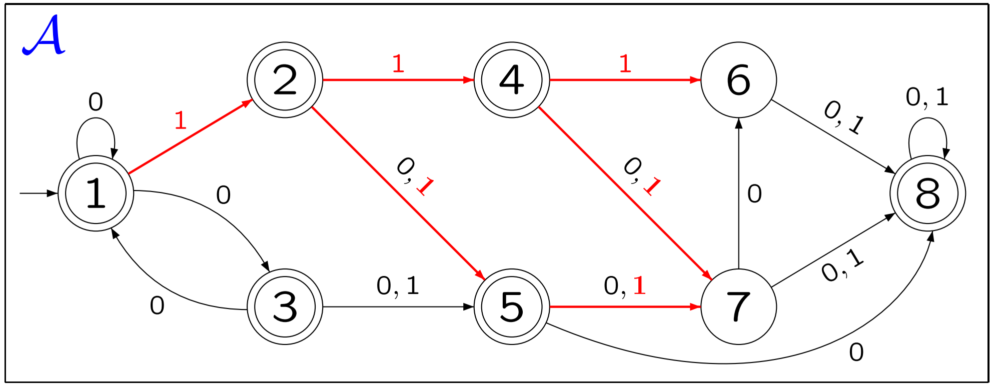
$$x_1 = \quad \mathsf{CPre}(x_0) \cup \{T\} \quad = \Big\{\{6,7\},\{4,5\}\Big\}$$

$$x_2 = \quad \mathsf{CPre}(x_1) \cup \{T\} \quad = \Big\{\{6,7\},\{4,5\},\{2,3\}\Big\}$$

$$x_3 = \quad \mathsf{CPre}(x_2) \cup \{T\} \quad = \Big\{\{6,7\},\{4,5\},\{2,3\},\{1\}\Big\}$$

$$x_4 = \quad \mathsf{CPre}(x_3) \cup \{T\} \quad = x_3$$

Universality - Example



$$x_0 = T = \{\{6,7\}\}$$

$$x_1 = \mathsf{CPre}(x_0) \cup \{T\} = \{\{6,7\},\{4,5\}\}$$

$$x_2 = \mathsf{CPre}(x_1) \cup \{T\} = \{\{6,7\},\{4,5\},\{2,3\}\}$$

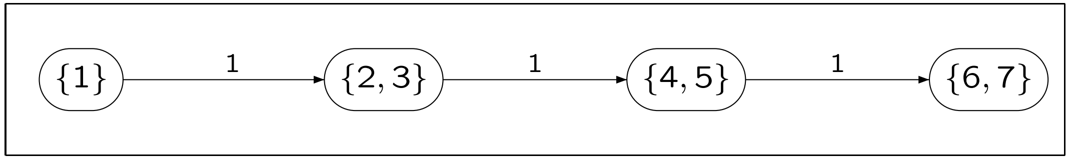
$$x_3 = \mathsf{CPre}(x_2) \cup \{T\} = \{\{6,7\},\{4,5\},\{2,3\},\{1\}\}$$

$$x_4 = \mathsf{CPre}(x_3) \cup \{T\} = x_3$$

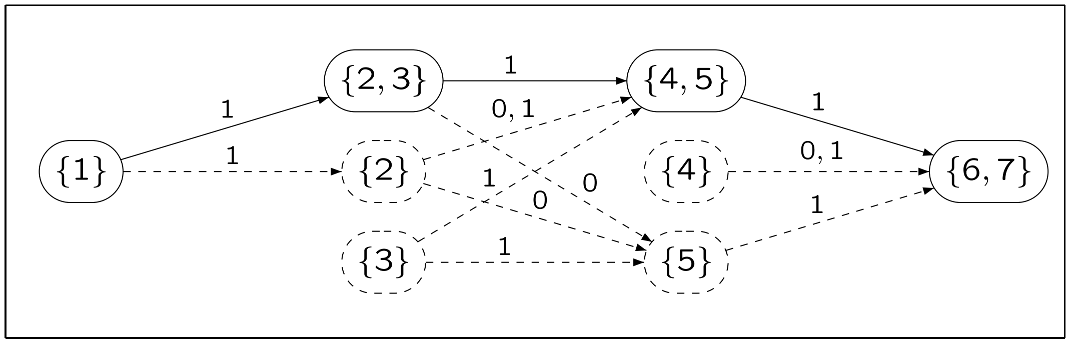
Protagonist has a strategy to win G_T (*e.g.*: $w = 111$)
$\iff \mathcal{A}$ is not universal

Universality - Example

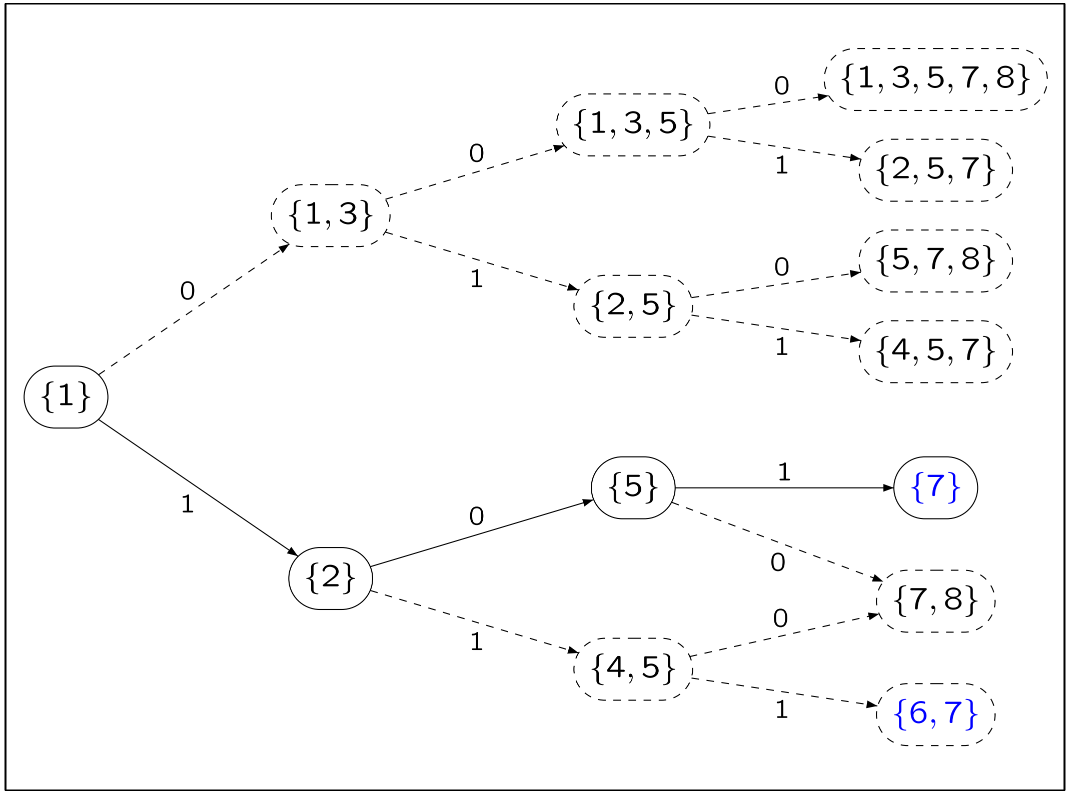
We have explored/constructed



instead of



Universality - Determinization



Outline of the talk

- Motivation
- Universality - A Game Approach
- Example
- Experimental Results
- Conclusion

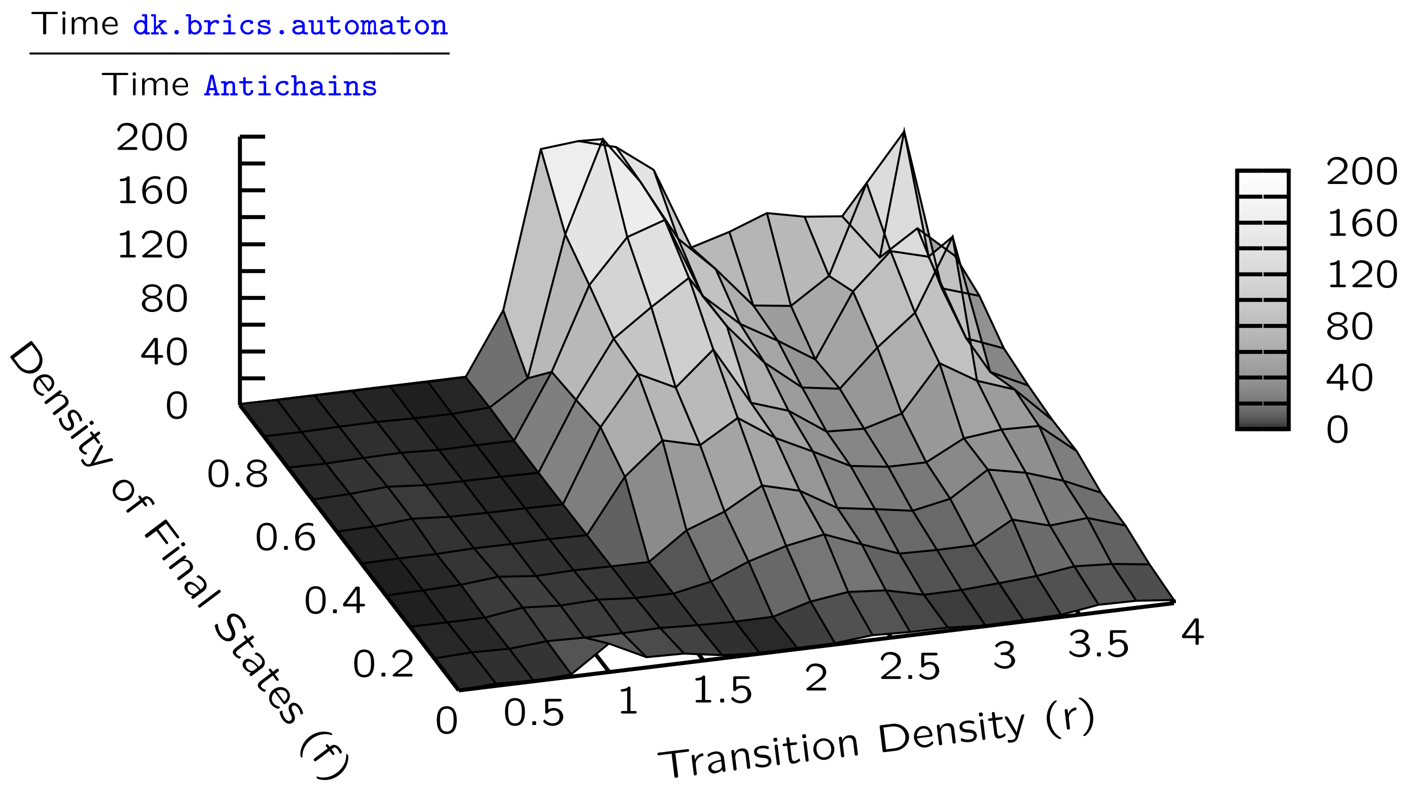
Universality - Experimental results (1)

- We compare our algorithm `Antichains` with the best[(1)] known algorithm `dk.brics.automaton` by Anders Møller.

  [(1)] According to *”D. Tabakov, M. Y. Vardi. Experimental Evaluation of Classical Automata Constructions. LPAR 2005”*.

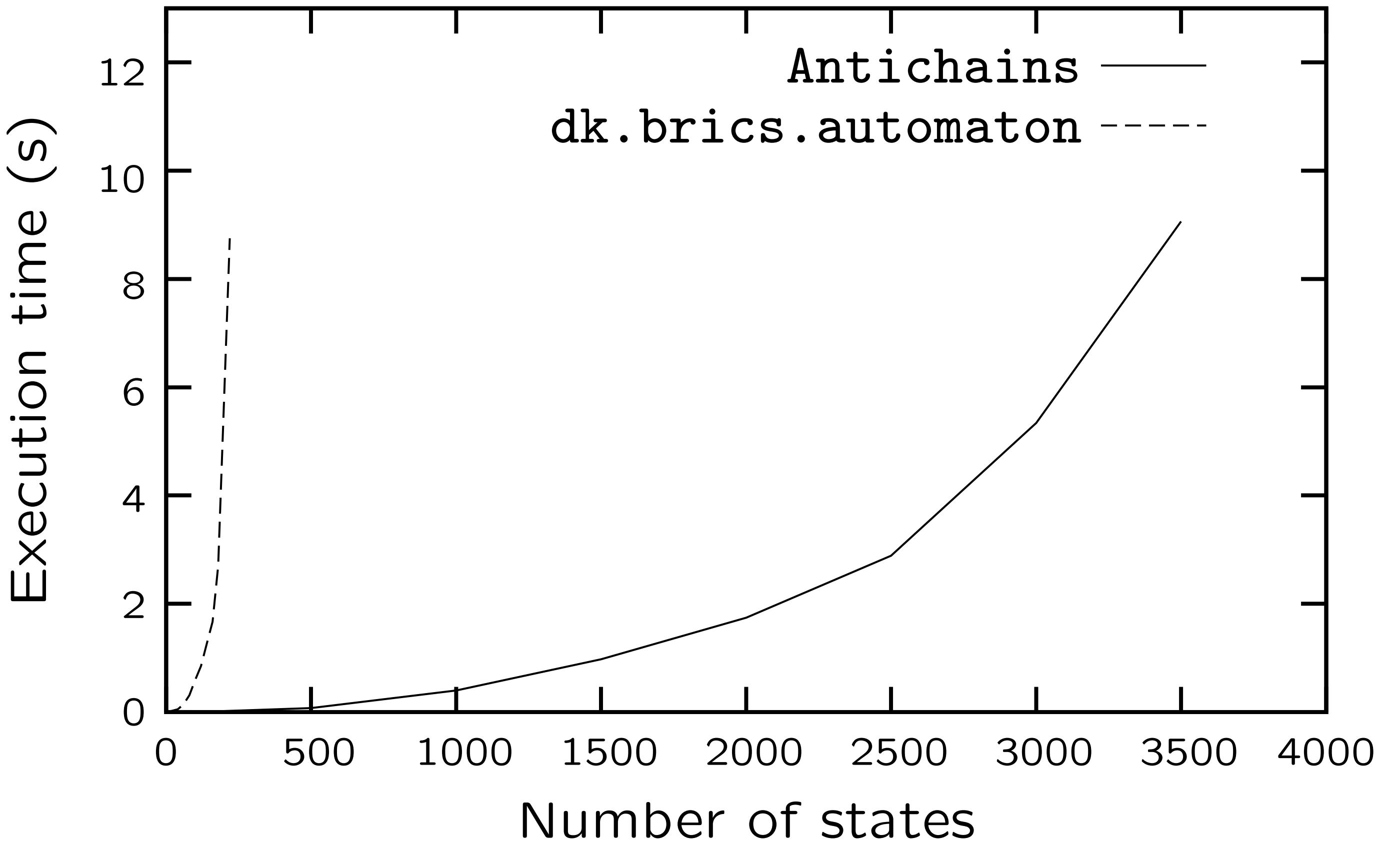
- We use a randomized model to generate the instances (automata of 175 locations). Two parameters:
  - Transition density: $r \geq 0$
  - Density of accepting states: $0 \leq f \leq 1$

Universality - Experimental results (2)



Each sample point: 100 automata with $|\mathsf{Loc}| = 175$, $\Sigma = \{0, 1\}$.

Universality - Experimental results (3)



- Transition density: $r = 2$.
- Density of accepting states: $f = 1$.

Determinization - Average Number of sets (100 instances)

# states	20	40	60	80	100	120	140	160
All instances	71	176	415	713	1120	1404	1750	2084
Univ. inst.	116	388	826	1563	2364	2805	3850	4758
¬Univ. inst.	11	28	64	98	61	162	32	67

Antichains - Average Number of sets (same 100 instances)

# states	20	40	60	80	100	120	140	160
All instances	3	4	6	7	9	9	9	12
Univ. inst.	3	6	7	9	12	13	14	19
¬Univ. inst.	3	3	4	6	6	6	5	7

Outline of the talk

- Motivation
- Universality - A Game Approach
- Example
- Experimental Results
- Conclusion

Beyond Universality

- Universality ($L(\mathcal{A}) = \Sigma^*$): antichains over 2^{Loc_A}.

  $$\mathsf{CPre}(q) = \left\lceil \{s \mid \exists s' \in q \cdot \exists \sigma \in \Sigma : \mathsf{post}_\sigma(s) \subseteq s'\} \right\rceil$$

- Language inclusion ($L(\mathcal{A}) \subseteq L(\mathcal{B})$): antichains over $\mathsf{Loc}_A \times 2^{\mathsf{Loc}_B}$.

  $$\mathsf{CPre}(q) = \left\lceil \{(\ell, s) \mid \exists (\ell', s') \in q \cdot \exists \sigma \in \Sigma : \ell' \in \delta^A(\ell, \sigma) \wedge \mathsf{post}^B_\sigma(s) \subseteq s'\} \right\rceil$$

- Emptiness of AFA ($L(\mathcal{A}) = \emptyset$): antichains over 2^{Loc_A}.

  $$\mathsf{CPre}(q) = \left\lceil \{s \mid \exists s' \in q \cdot \exists \sigma \in \Sigma \cdot \forall \ell \in s : s' \models \delta(\ell, \sigma)\} \right\rceil$$

Conclusion and perspectives

The antichains algorithms apply to:

- Universality of FSA,
- Language inclusion of FSA,
- Emptiness of finite alternating automata.

- . . . and soon to automata over infinite words (Büchi)? (work in progress)

Thank you

Questions ???