Weak Bisimulation Finiteness of Pushdown Systems With Deterministic ε-Transitions Is 2-EXPTIME-Complete

Stefan Göller
University of Kassel

Paweł Parys
University of Warsaw

based on SODA 2023 paper

Pushdown systems

are given by a tuple (Q,Γ,A,R), where

- $Q=\{p,q,r\}$ is a finite set of control states
- $\Gamma=\{X,Y,Z\}$ is a finite set of stack symbols
- $A=\{a,b,c\}$ is a finite set of input symbols and
- R is a finite set of **rewrite rules** of either form:

$\overset{p}{\boxed{X}} \xrightarrow{a} q$ (pop rule) or $\overset{p}{\boxed{X}} \xrightarrow{a} \overset{q}{\begin{array}{|c|}\hline Y\\ \hline Z\\ \hline\end{array}}$ (push rule)

induce an infinite A-edge-labeled transition system…

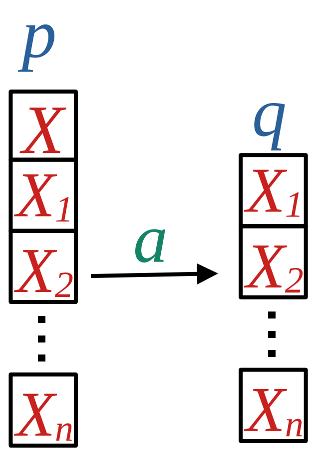
Induced transition system (infinite)

Each pushdown system (Q,Γ,A,R) induces an infinite transition system:

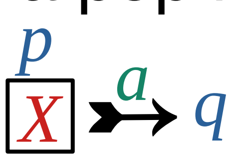
- nodes = state & stack 

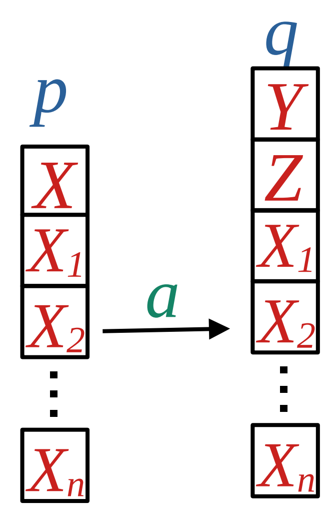
$\in Q \times \Gamma^*$

- transitions (labeled by A):

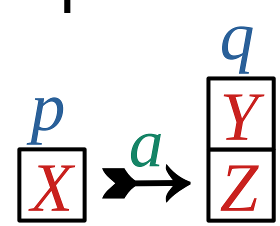


for a pop rule:



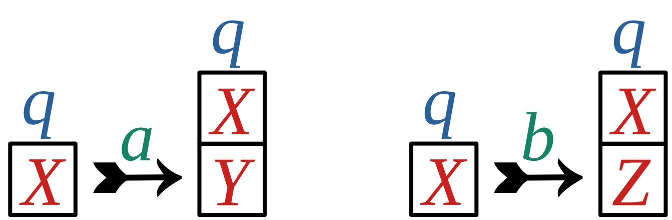


for a push rule:

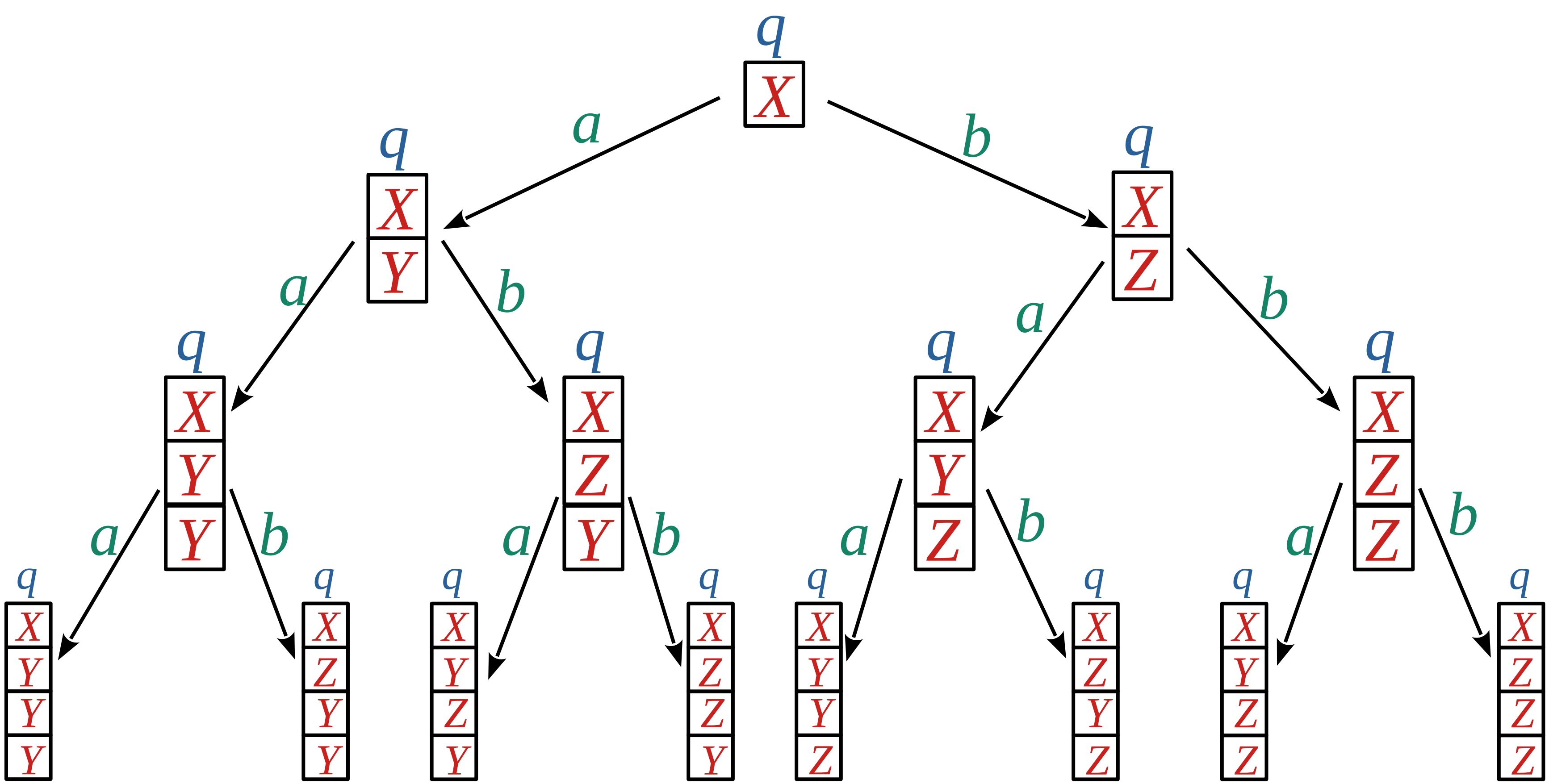


Example pushdown system

The two rules



induce the infinite binary tree



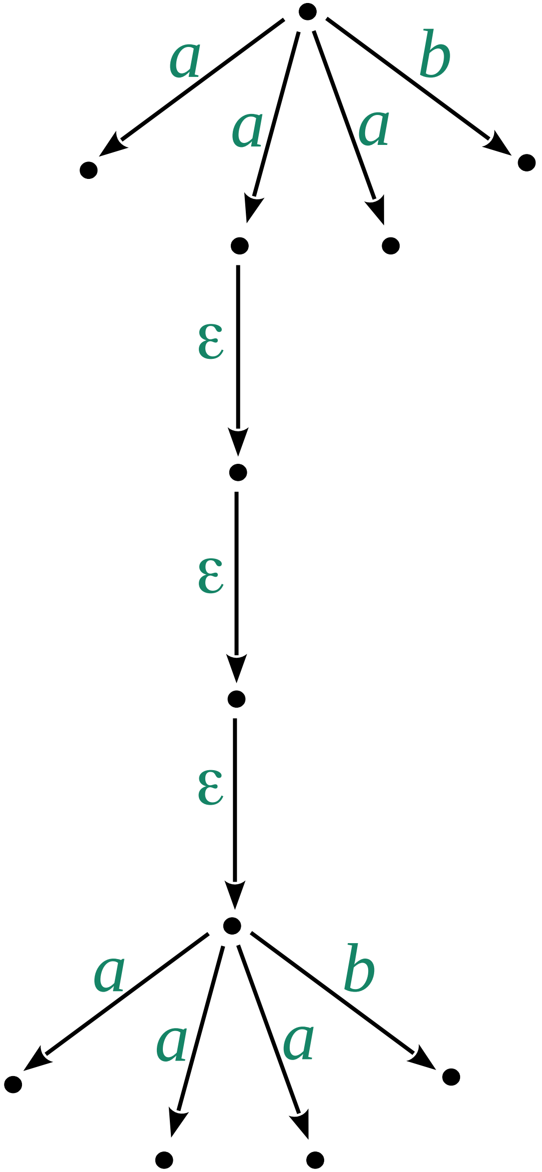
Why study pushdown systems?

Pushdown systems…

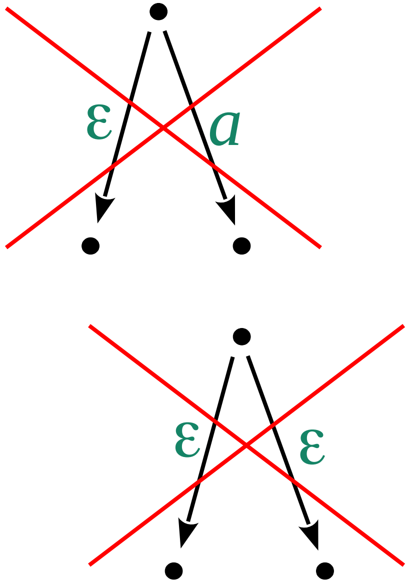
- can be used to model the call and return behavior of recursive programs
- have been used to find bugs in Java programs [Suwimontherabuth/Berger/Schwoon/Esparza 1997]
- equivalence checking (in the deterministic case) has been used to verify security protocols [Chrétien, Cortier, Delaune 2015]
- reachability can be checked in polynomial time [Caucal 1990, Bouajjani/Esparza/Maler 1997]
- have a decidable MSO-theory [Muller/Schupp 1985]
- can be model checked against μ-calculus formulas in exponential time [Walukiewicz 1996]

We allow deterministic ε-transitions

allowed:

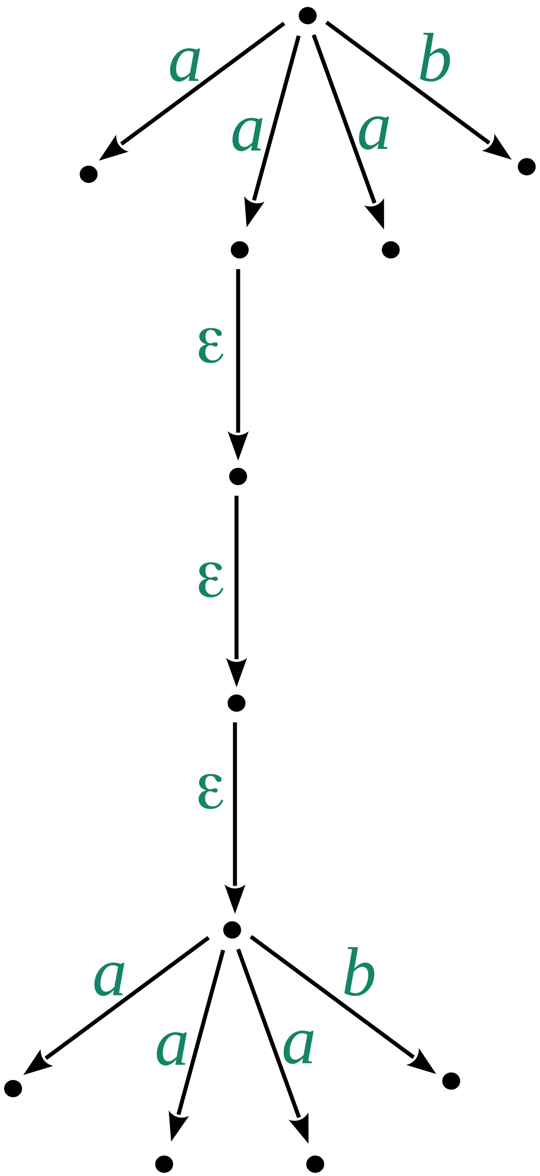


forbidden:

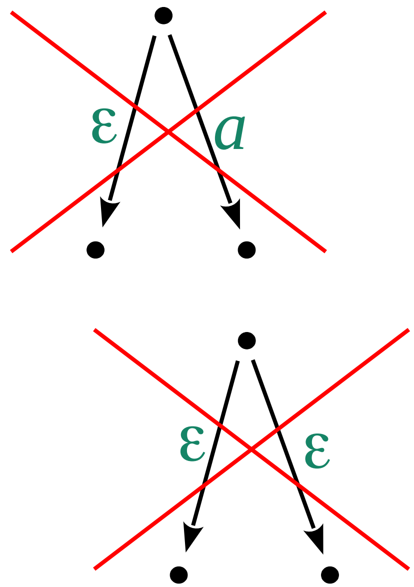


We allow deterministic ε-transitions

allowed:



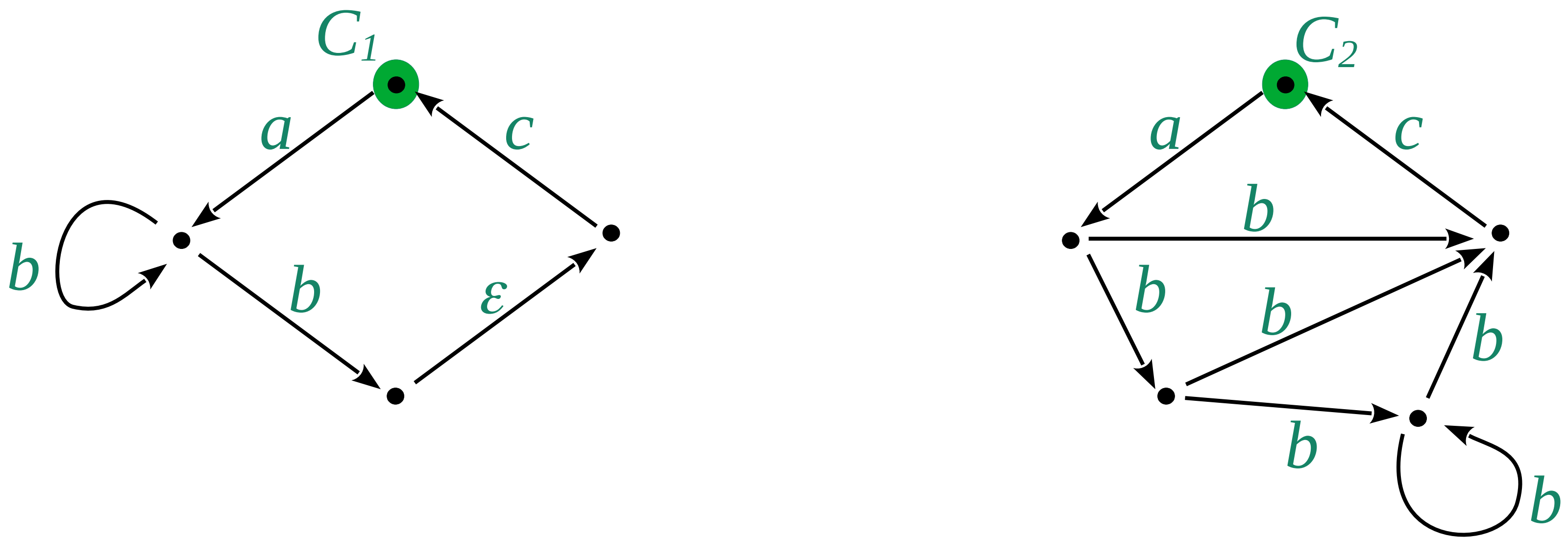
forbidden:



- this version is equivalent to first-order grammars (programs with recursion)
- ε-transitions are useful to pop many symbols from the stack

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

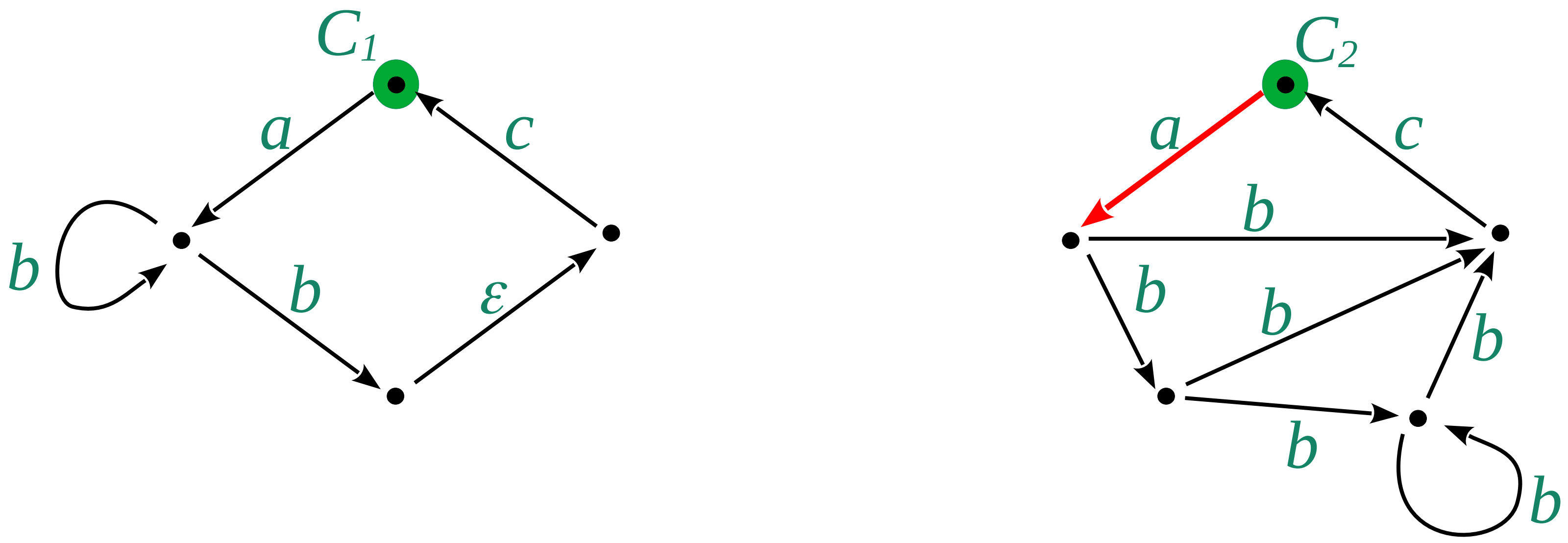


Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

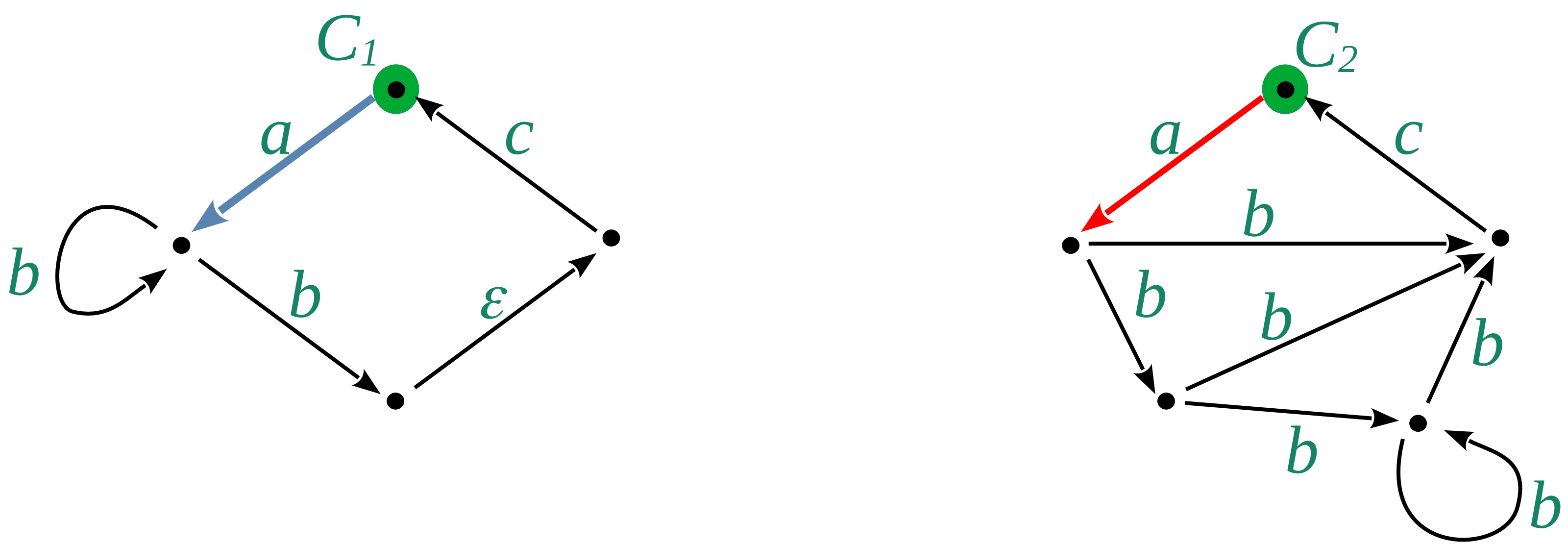


Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

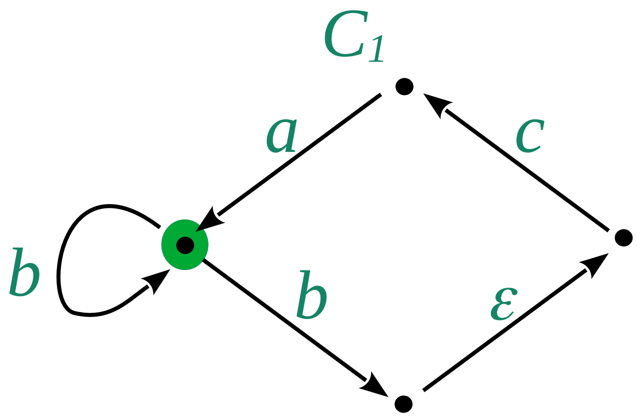


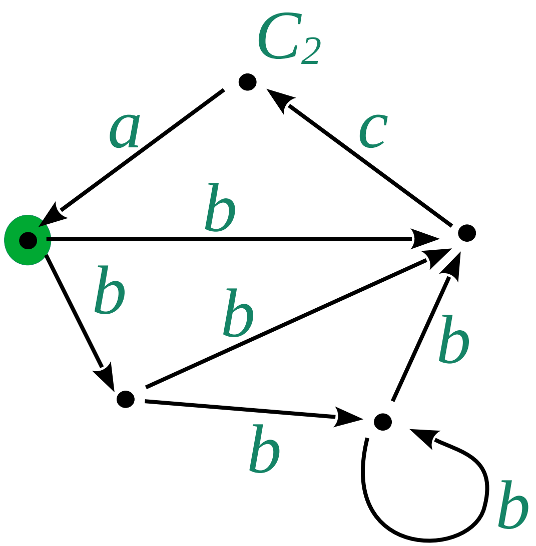
Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.



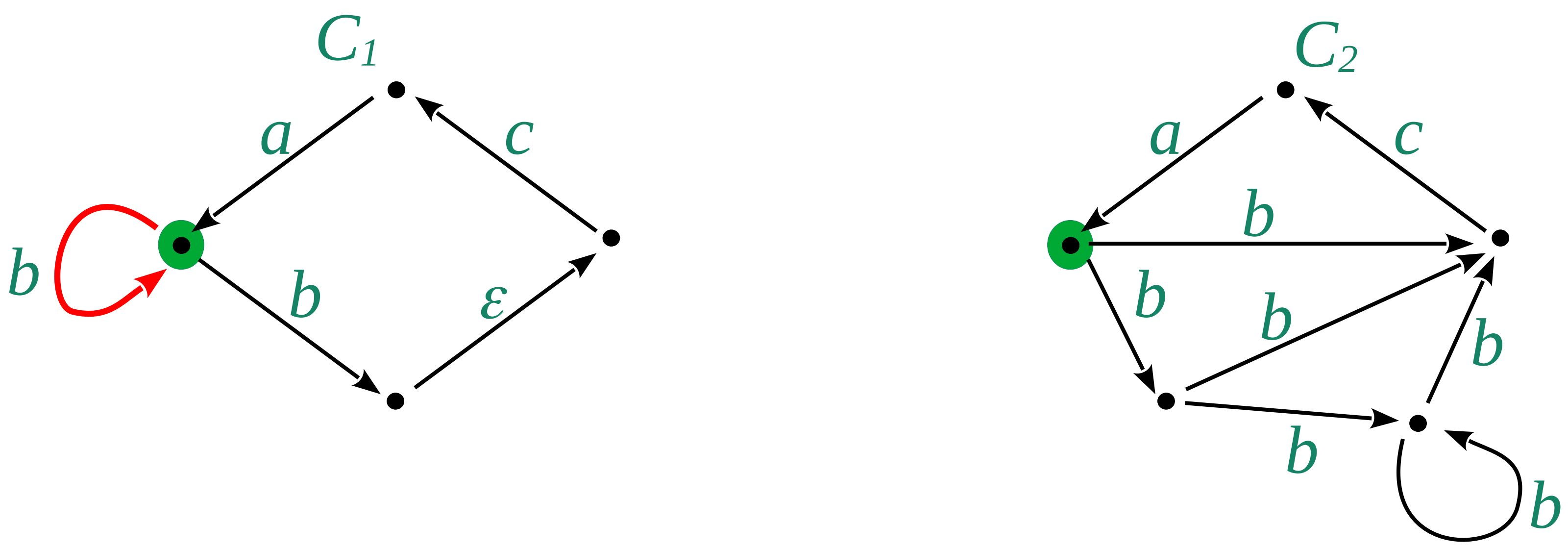


Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

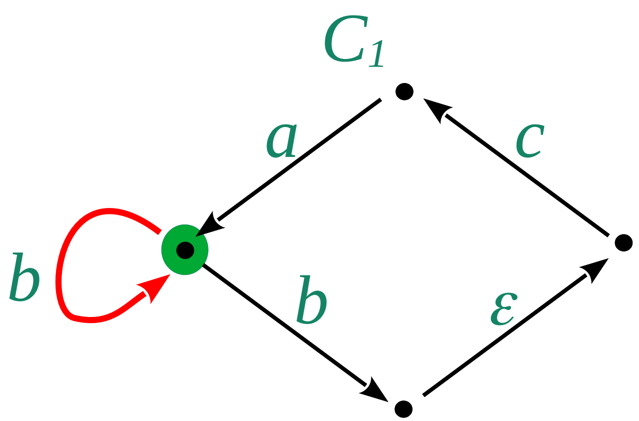


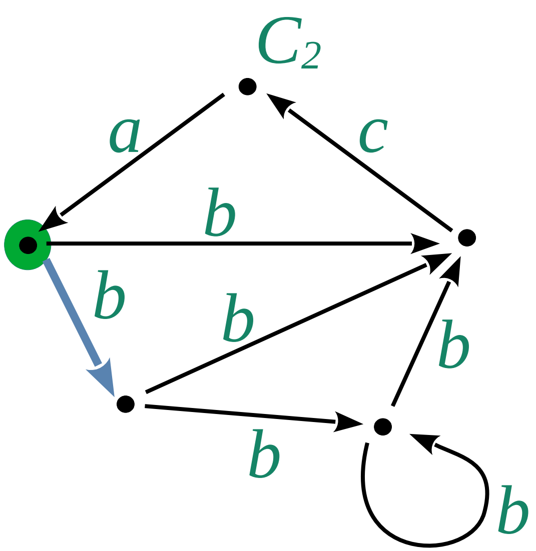
Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.



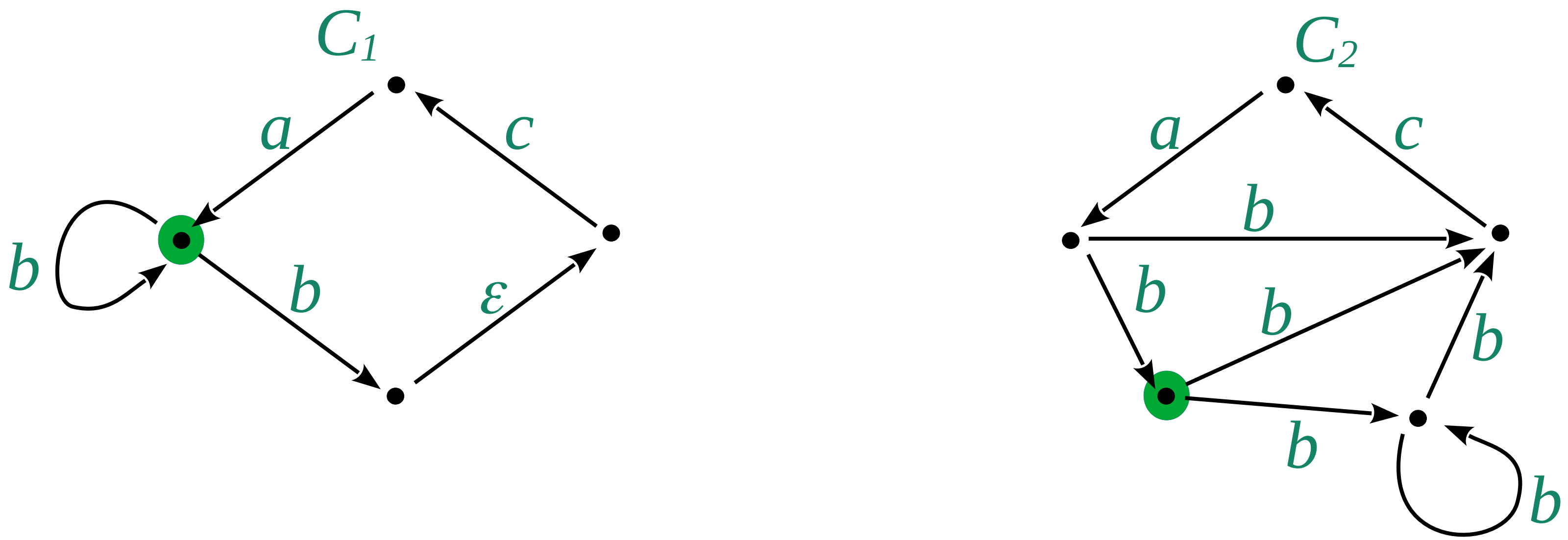


Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

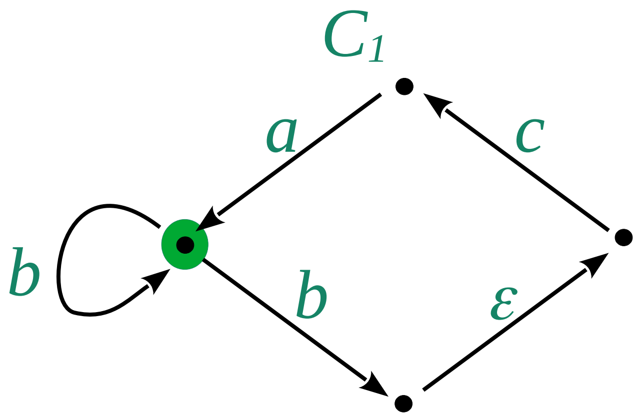


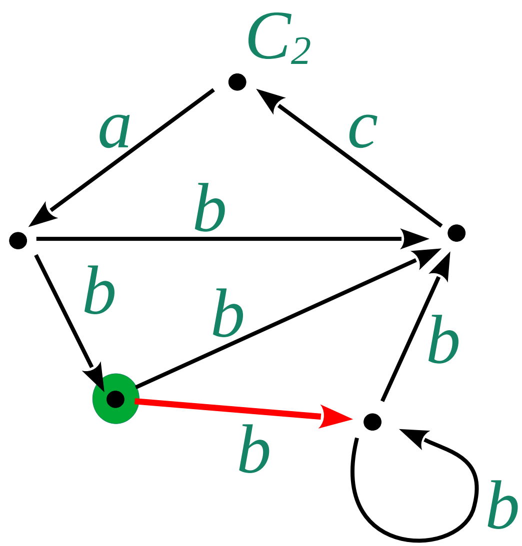
Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.



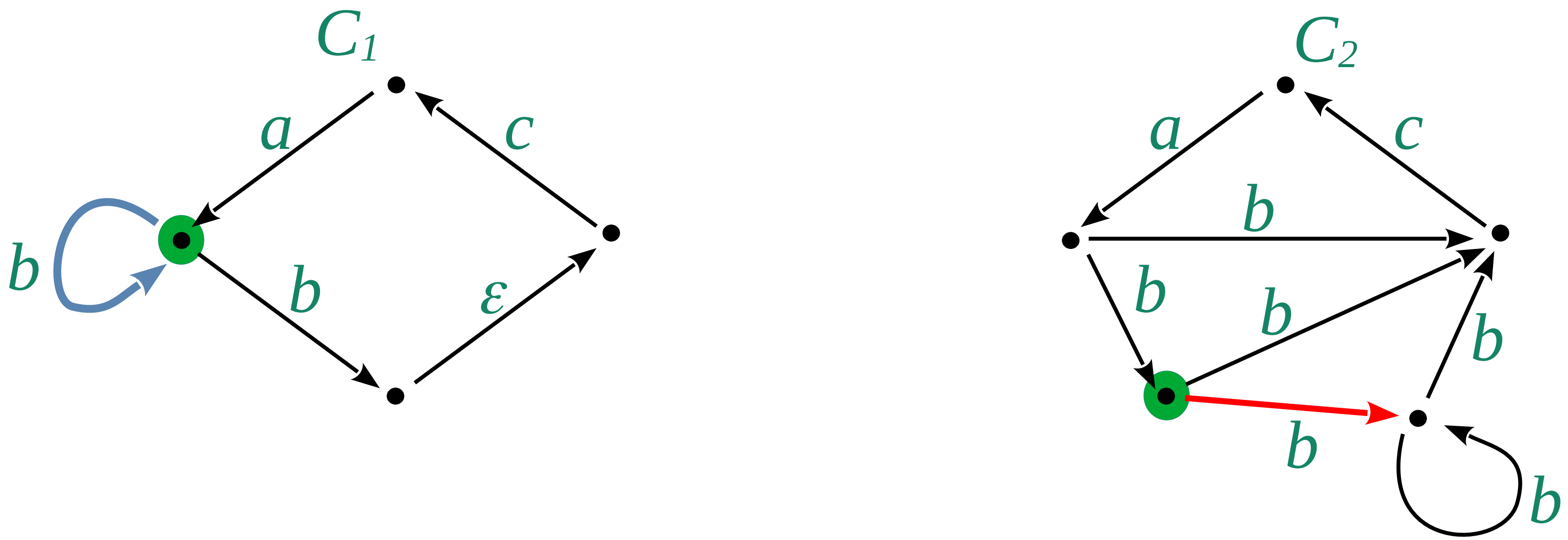


Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

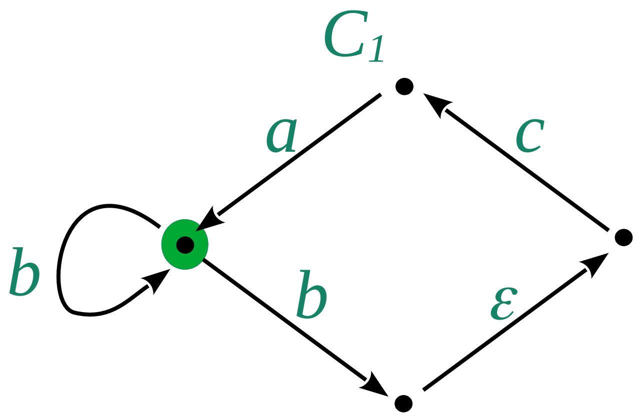


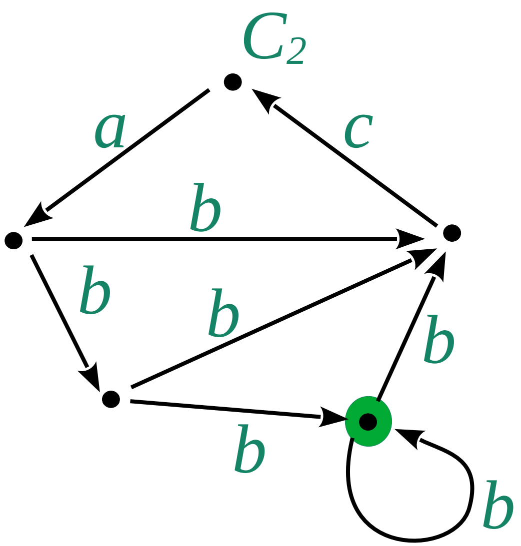
Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.



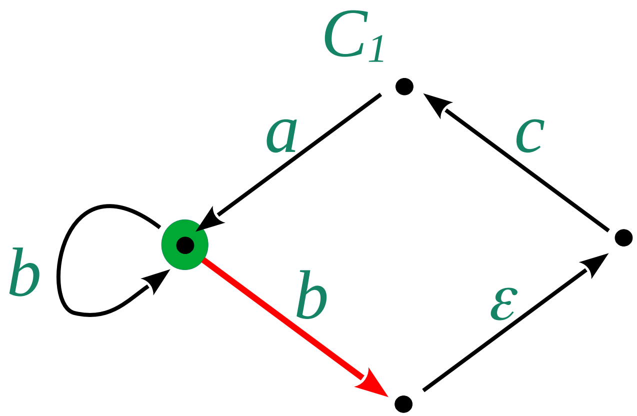


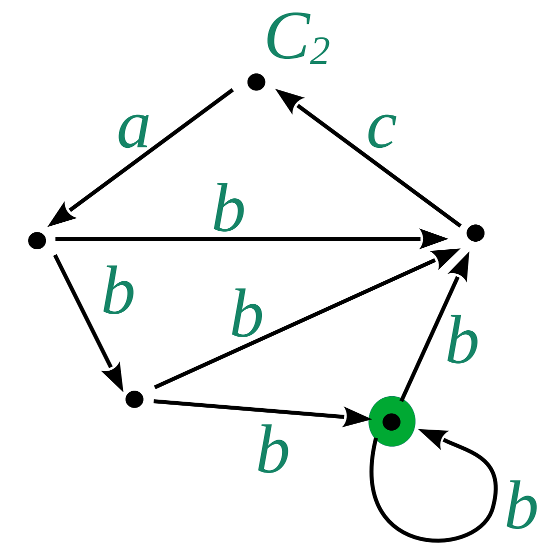
Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.



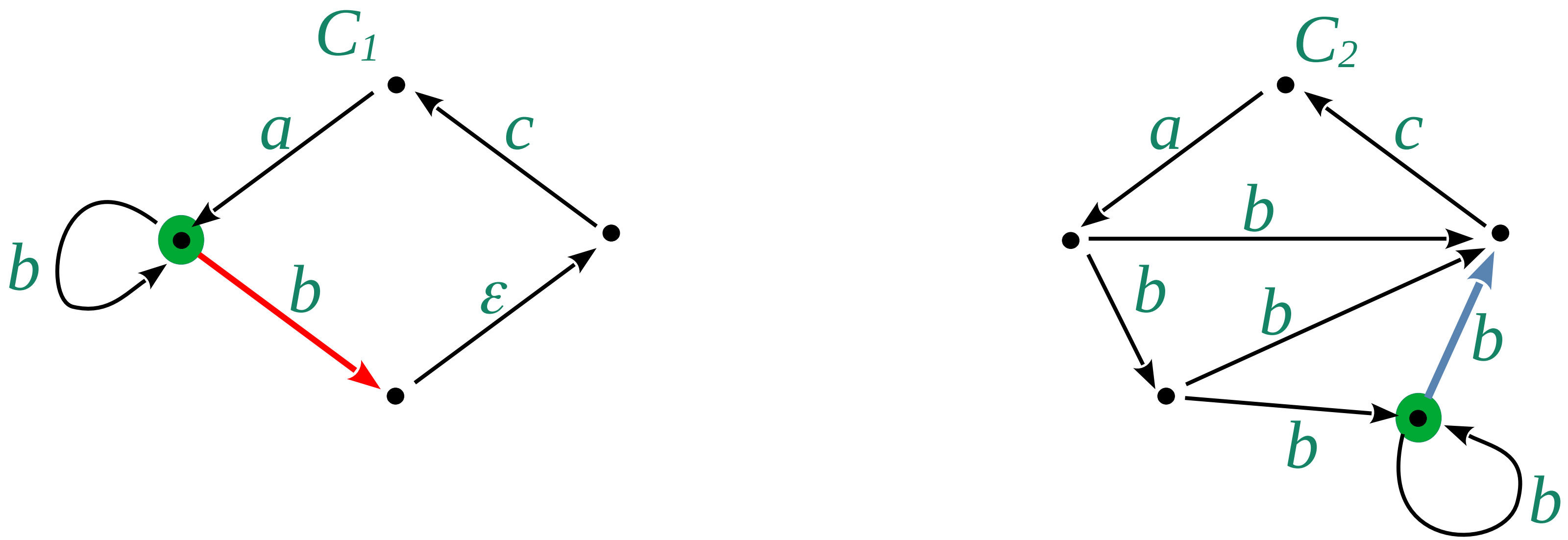


Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

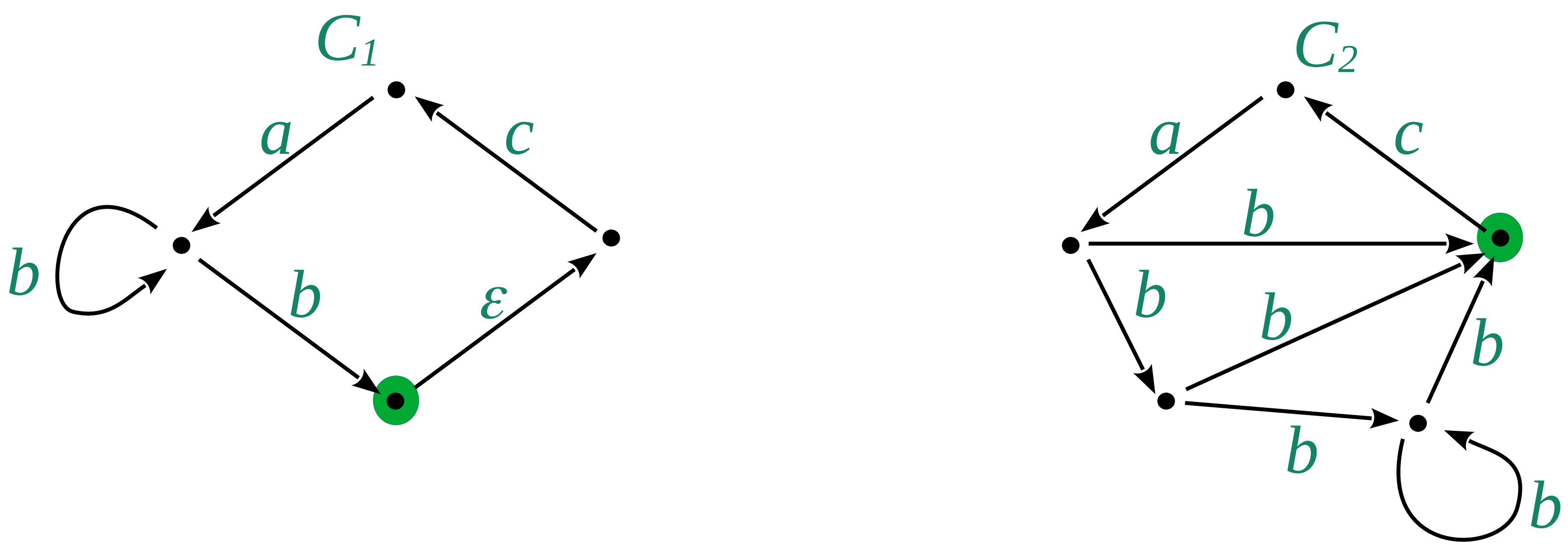


Spoiler claims that $C_1 \nsim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

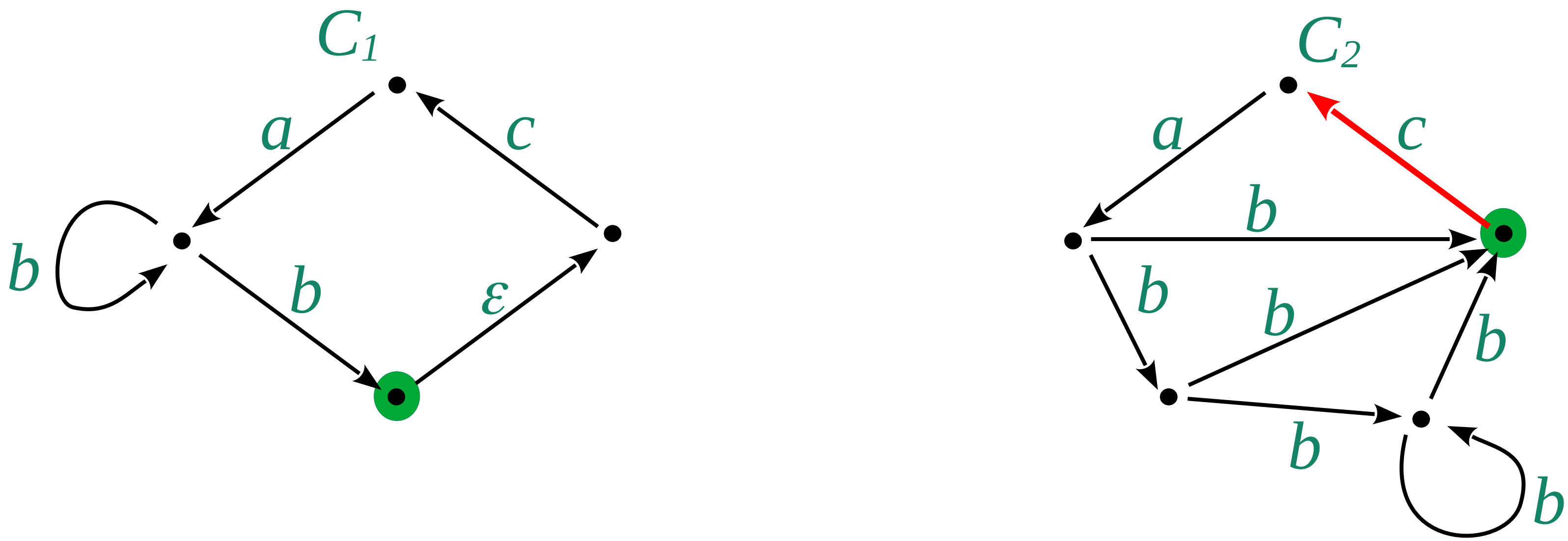


Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.

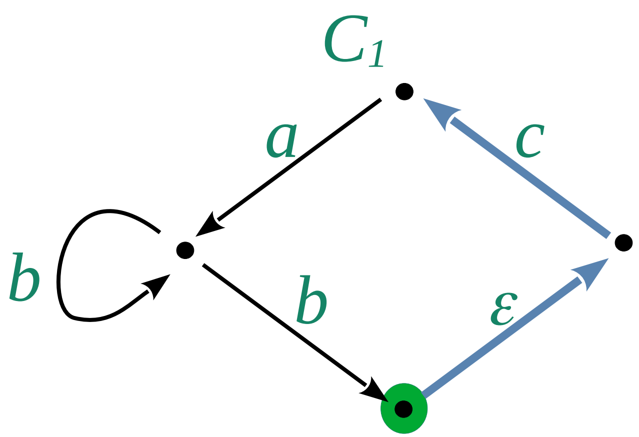


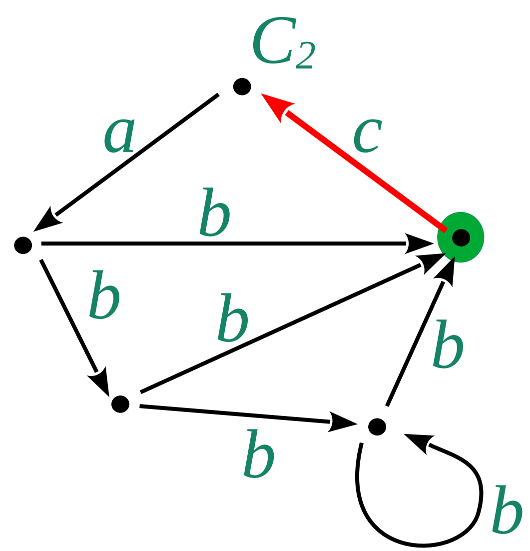
Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

Bisimulation equivalence

can be seen as a two player game between Spoiler and Duplicator.





Spoiler claims that $C_1 \not\sim C_2$

Duplicator claims that $C_1 \sim C_2$

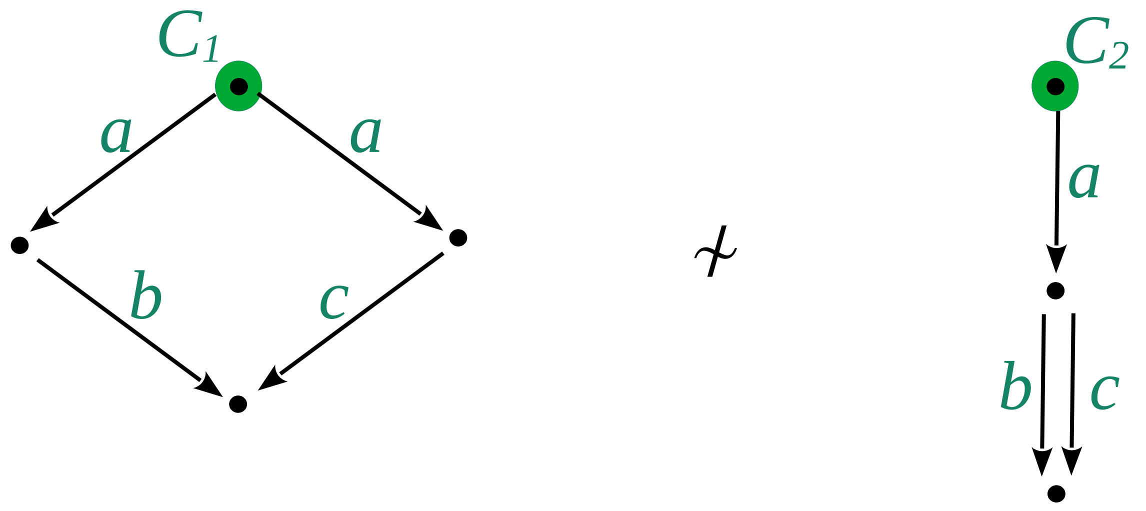
infinite play =
Duplicator wins

Moves = paths $\varepsilon^* a \varepsilon^*$

A.k.a. weak bisimulation
A.k.a. bisimulation after contracting ε-transitions

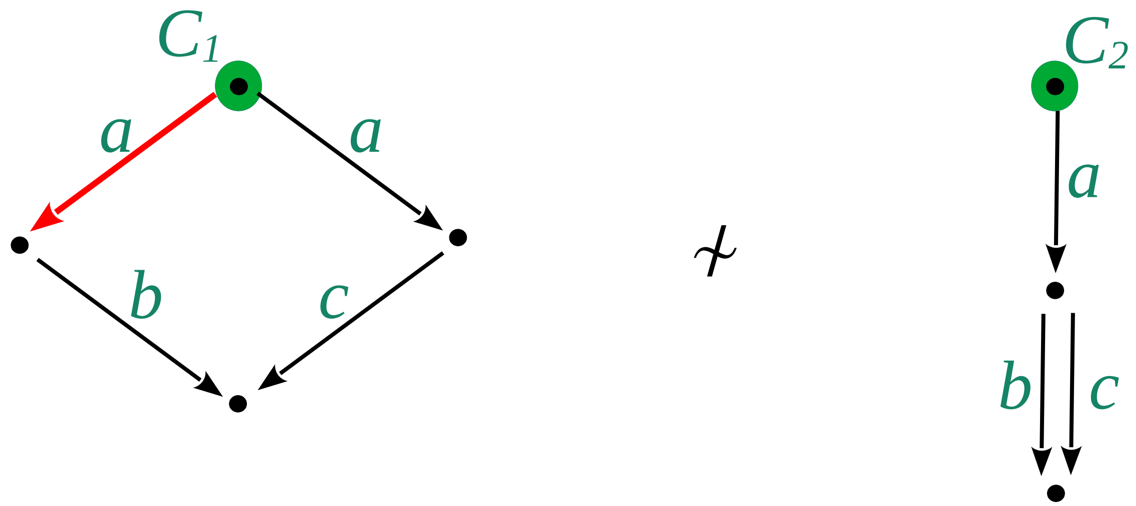
Bisimulation equivalence

Negative example:



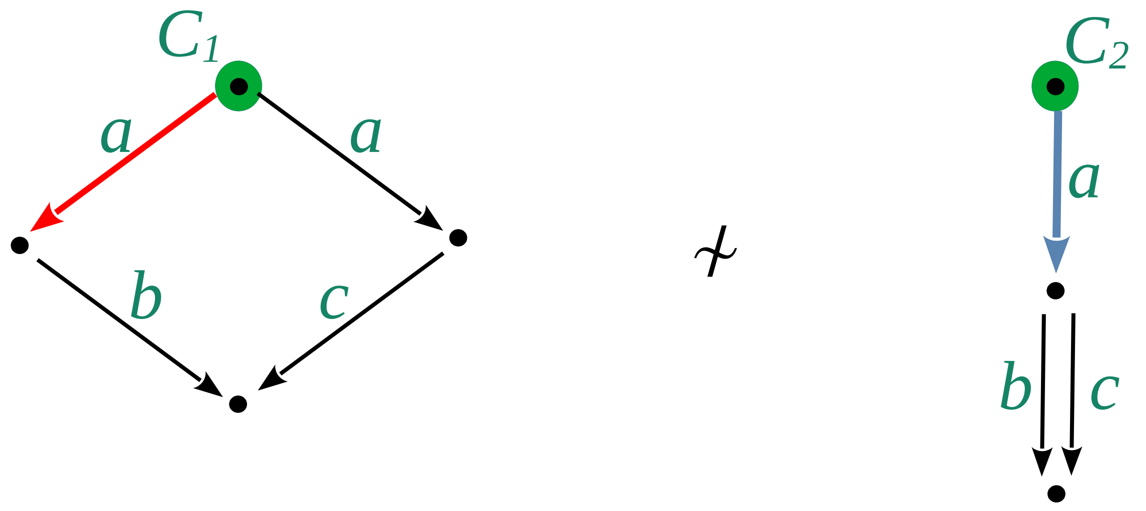
Bisimulation equivalence

Negative example:



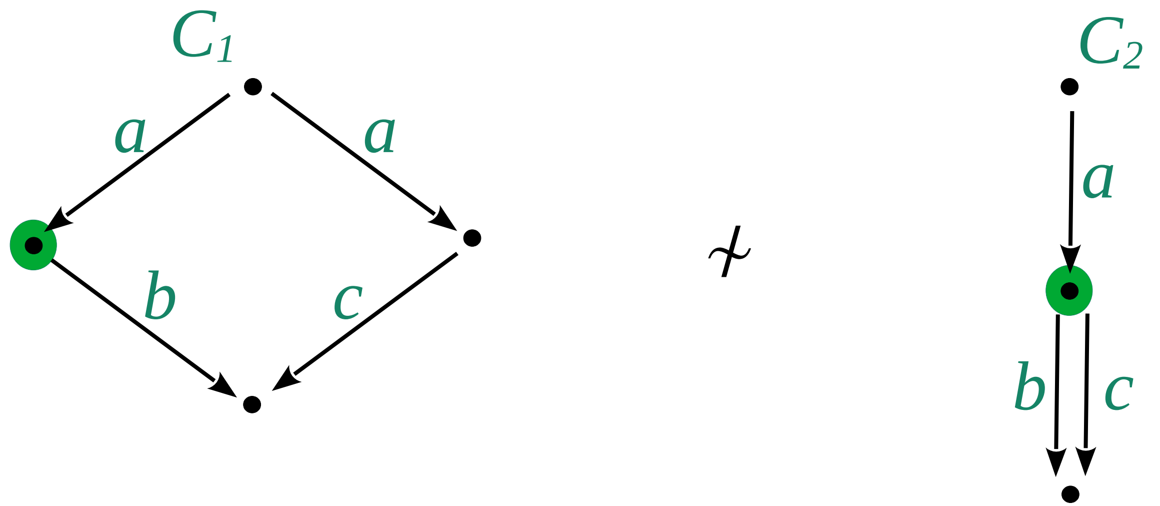
Bisimulation equivalence

Negative example:



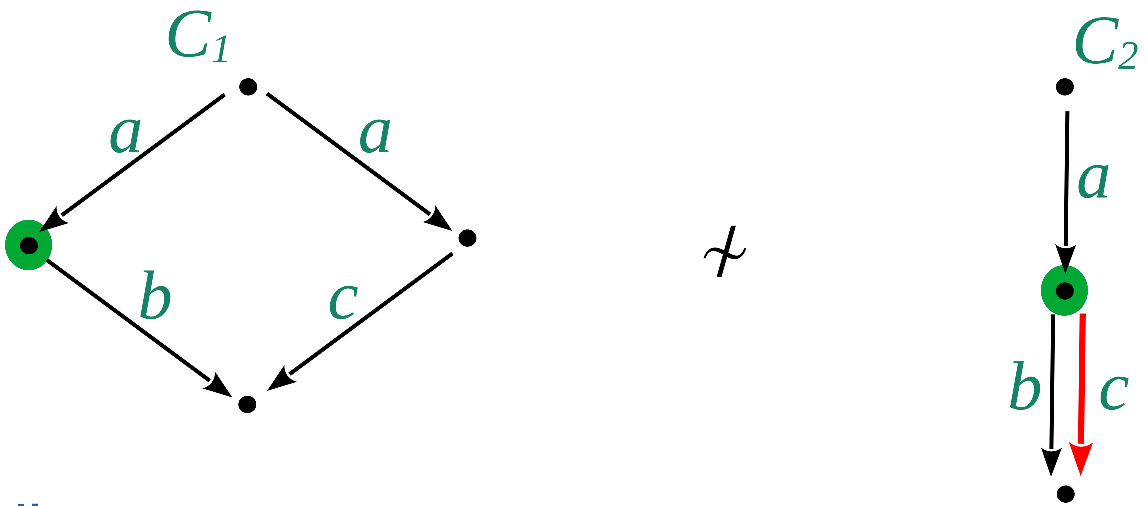
Bisimulation equivalence

Negative example:



Bisimulation equivalence

Negative example:



Duplicator cannot answer

Why bisimulation equivalence?

Verification logics		Classical logics	
Modal logic	$=$	$\mathrm{FO}_{\sim}$	[van Benthem 1976]
μ-calculus	$=$	$\mathrm{MSO}_{\sim}$	[Janin/Walukiewicz 1996]
CTL^*	$=$	$\mathrm{MPL}_{\sim}$	[Moller/Rabinovich 2003]
	$\vdots$		

Bisimulation equivalence is the central notion of equivalence in formal verification!

Bisimulation finiteness

is the following decision problem:

INPUT: a pushdown system P

QUESTION: is P bisimilar to some finite system?

(the finite system is NOT part of the input)

Bisimulation finiteness

is the following decision problem:

INPUT: a pushdown system P

QUESTION: is P bisimilar to some finite system?

(the finite system is NOT part of the input)

Theorem [Jančar 2016]
This problem is decidable.

Proof: two semi-decision procedures;
oracle calls to the bisimulation equivalence problem

Bisimulation equivalence

is the following decision problem:

INPUT: two pushdown systems P_1, P_2

QUESTION: does $P_1 \sim P_2$?

Theorem

This problem is decidable [Sénizergues 1998]

and ACKERMANN-complete [Zhang/Yin/Long/Xu 2020, Schmitz/Jancar 2019]

Bisimulation equivalence

is the following decision problem:

INPUT: two pushdown systems P_1, P_2

QUESTION: does $P_1 \sim P_2$?

Theorem
This problem is decidable [Sénizergues 1998]
and ACKERMANN-complete [Zhang/Yin/Long/Xu 2020, Schmitz/Jancar 2019]

Bisimulation equivalence with a finite system

INPUT: a pushdown system P, a finite system F

QUESTION: does $P \sim F$?

Theorem [Kučera/Mayr 2010]
This problem is PSPACE-complete.

Bisimulation finiteness

INPUT: a pushdown system P

QUESTION: is P bisimilar to some finite system?

(the finite system is NOT part of the input)

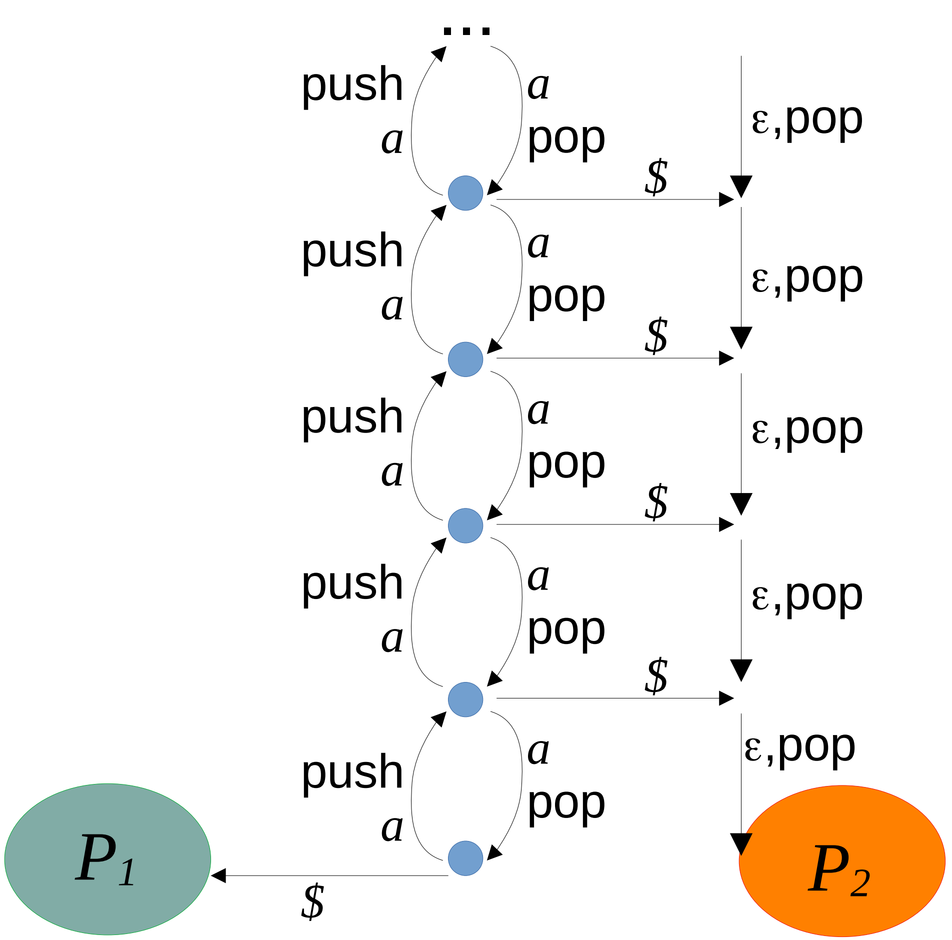
- This problem is decidable (in ACKERMANN) [Jančar 2016]
- For P without ε-transitions, it is in 6-EXPSPACE [Göller/Parys 2020]
- **This paper: the problem is 2-EXPTIME-complete**

Our main result

Bisimulation finiteness is 2-EXPTIME-complete

Proof strategy (lower bound)

- Suppose that P_1, P_2 are bisimulation finite systems.
  Then we can construct $P(P_1,P_2)$ that is bisimulation finite iff $P_1 \sim P_2$



Our main result

Bisimulation finiteness is 2-EXPTIME-complete

Proof strategy (lower bound)

- Suppose that P_1, P_2 are bisimulation finite systems.
  Then we can construct $P(P_1,P_2)$ that is bisimulation finite iff $P_1 \sim P_2$
- We reduce from alternating EXPSPACE Turing machines.
  We have to construct bisimulation finite systems P_1, P_2 such that $P_1 \sim P_2$ iff M accepts.

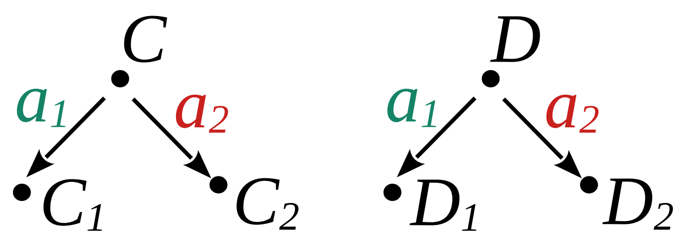
Our main result

Bisimulation finiteness is 2-EXPTIME-complete

Proof strategy (lower bound)

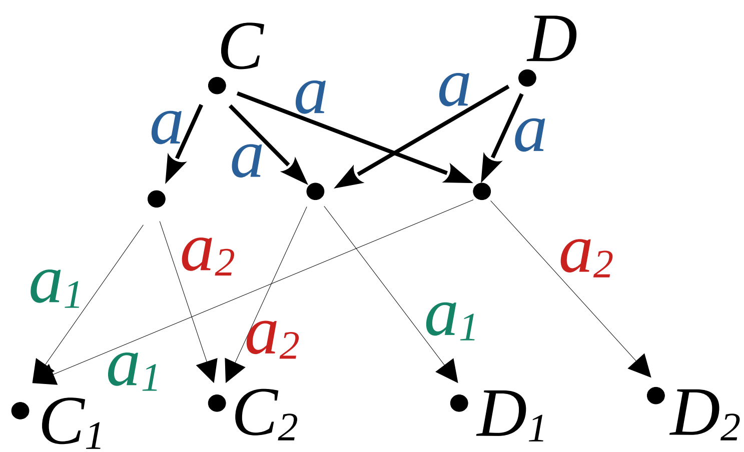
- We have to construct bisimulation finite systems P_1, P_2 such that $P_1 \sim P_2$ iff an alternating EXPSPACE Turing machine M accepts.
- AND realized directly:

$C \sim D$ iff $C_1 \sim D_1 \wedge C_2 \sim D_2$



- OR realized by „Defender's forcing" gadget [Jančar/Srba 2008]:

$C \sim D$ iff $C_1 \sim D_1 \vee C_2 \sim D_2$



Our main result

Bisimulation finiteness is 2-EXPTIME-complete

Proof strategy (upper bound)

Thm 1: If $P \sim F$ for some F then $P \sim F'$ for some F' of size $< 2^{2^{|P|^c}}$

Use of Thm 1: Try to generate minimal F bisimilar to P;
stop when F too large (a new, polynomial algorithm)

Thm 1: If $P \sim F$ for some F then $P \sim F'$ for some F' of size $< 2^{2^{|P|^c}}$

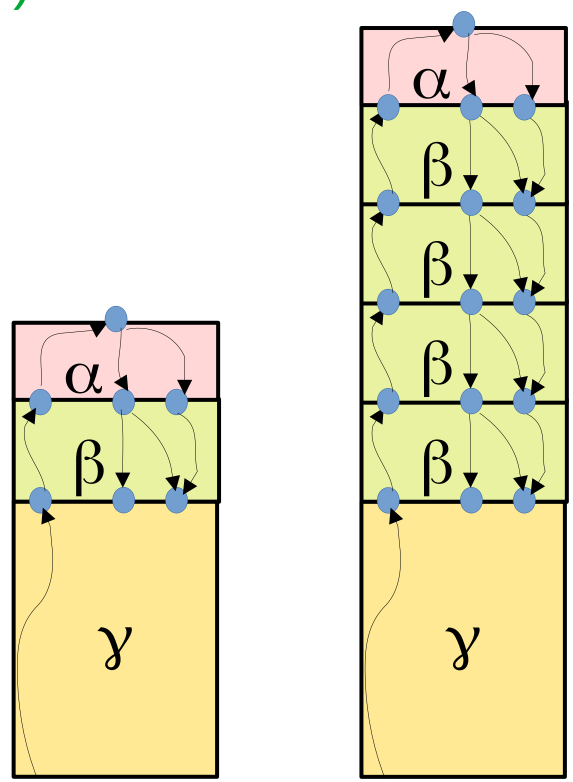
- This presentation: no ε-transitions
- Consider a reachable configuration $q\delta$

Thm 1: If $P \sim F$ for some F then $P \sim F'$ for some F' of size $< 2^{2^{|P|^c}}$

- This presentation: no ε-transitions
- Consider a reachable configuration $q\delta$

Step 1: represent $\delta=\alpha\beta\gamma$ to allow pumping:

- all $q\alpha\beta^i\gamma$ reachable
- set of states after popping $\alpha\beta^j$ from $q\alpha\beta^i\gamma$ the same for all j
- α, β short (exponential size)



Thm 1: If $P \sim F$ for some F then $P \sim F'$ for some F' of size $< 2^{2^{|P|^c}}$

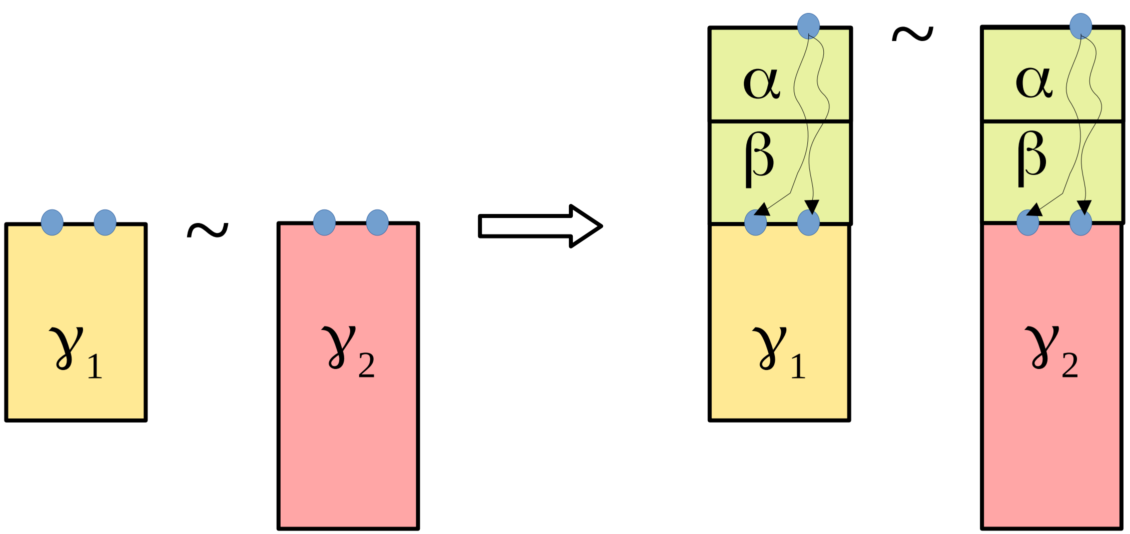
- This presentation: no ε-transitions
- Consider a reachable configuration $q\delta$

Step 1: represent $\delta=\alpha\beta\gamma$ to allow pumping:

- all $q\alpha\beta^i\gamma$ reachable
- set of states after popping $\alpha\beta^j$ from $q\alpha\beta^i\gamma$ the same for all j
- α, β short (exponential size)

Goal: prove that the number of classes of configurations $r\gamma$ (reachable by popping from $q\alpha\beta^i\gamma$) is small

- enough, because $[q\alpha\beta\gamma]$ is determined by α, β, and $[r\gamma]$



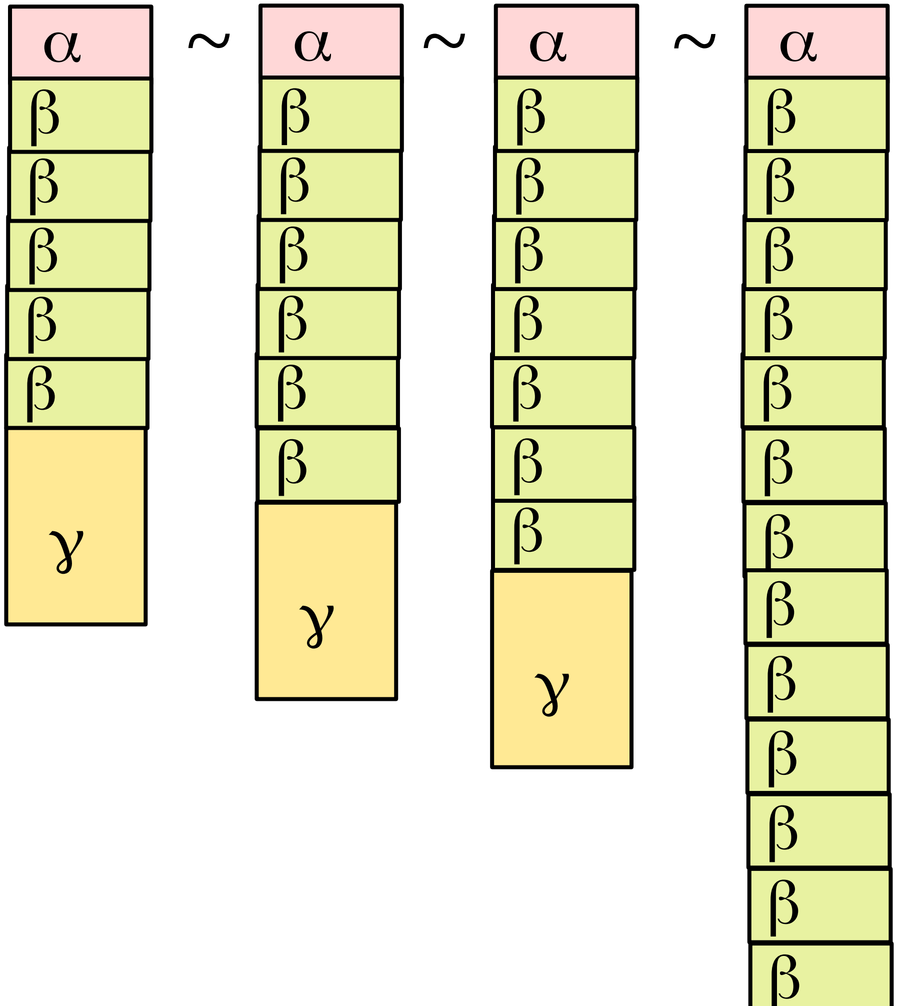
Assumption: $P \sim F$ for some finite F.

Step 1: represent $\delta = \alpha\beta\gamma$ to allow pumping:

- all $q\alpha\beta^i\gamma$ reachable
- set of states after popping $\alpha\beta^j$ from $q\alpha\beta^i\gamma$ the same for all j

Observation: if 2 configurations are not equivalent,
then this can be detected in the first $|F|$ steps.

- Configurations $q\alpha\beta^i\gamma$ for $i > |F|$ are all equivalent.



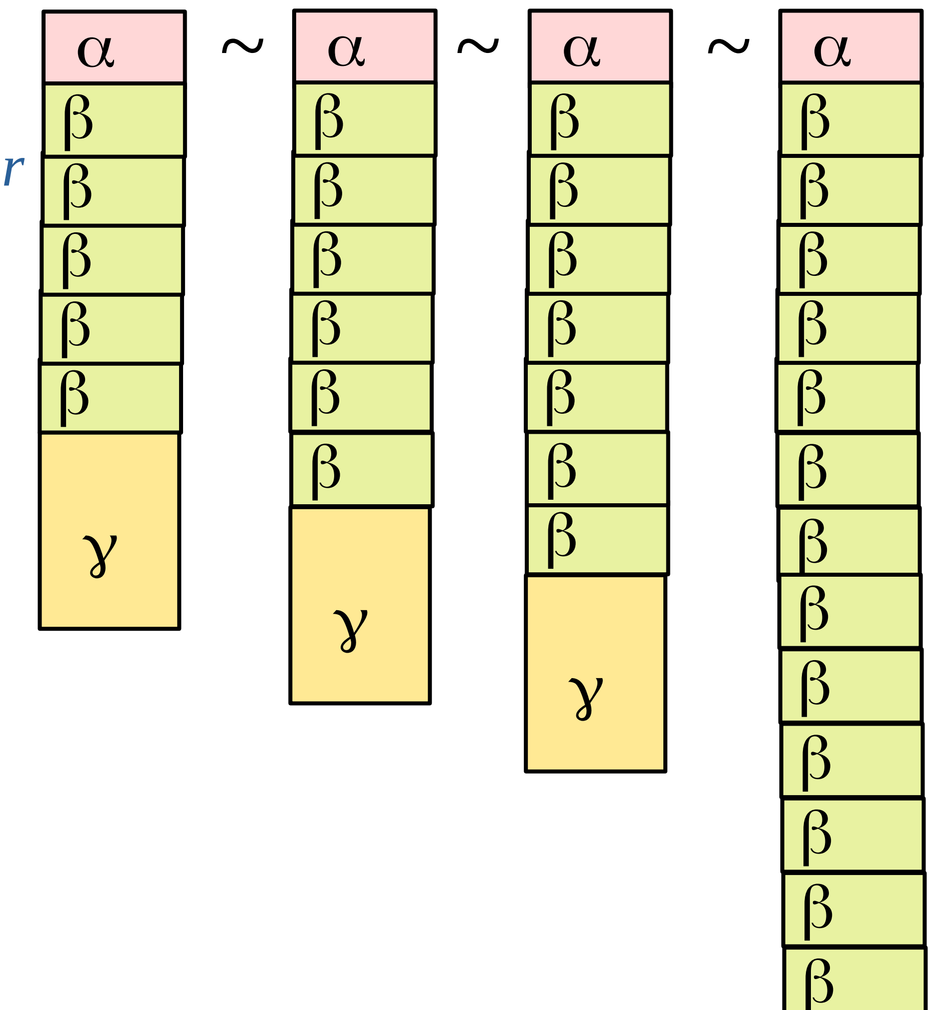
Assumption: $P \sim F$ for some finite F.

Step 1: represent $\delta=\alpha\beta\gamma$ to allow pumping:

- all $q\alpha\beta^i\gamma$ reachable
- set of states after popping $\alpha\beta^j$ from $q\alpha\beta^i\gamma$ the same for all j

Observation: if 2 configurations are not equivalent, then this can be detected in the first $|F|$ steps.

- Configurations $q\alpha\beta^i\gamma$ for $i>|F|$ are all equivalent.



Consider the smallest e such that

$\cancel{q\alpha\beta^e\gamma\sim q\alpha\beta^\infty}$ $r\beta^e\gamma\sim r\beta^\infty$ for all reachable r

We want to prove $e<2^{2^{|P|^c}}$

To this end, we will provide a "short description" of $r\beta^i\gamma$, different for every $i<e$

Assumption: $P \sim F$ for some finite F.

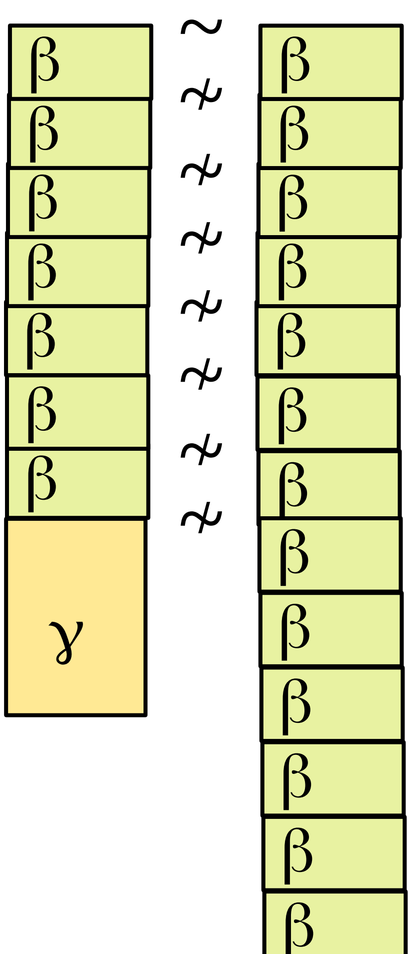
Consider the smallest e such that $r\beta^e\gamma \sim r\beta^\infty$ for all reachable r

We want to prove $e < 2^{2^{|P|^c}}$

For all $i<e$ let M_i = number of steps needed to distinguish $r\beta^i\gamma$ and $r\beta^\infty$

Easy to see: $M_1 < M_2 < M_3 < \ldots < M_{e-1}$

In particular $[r\beta^i\gamma] \neq [r\beta^j\gamma]$

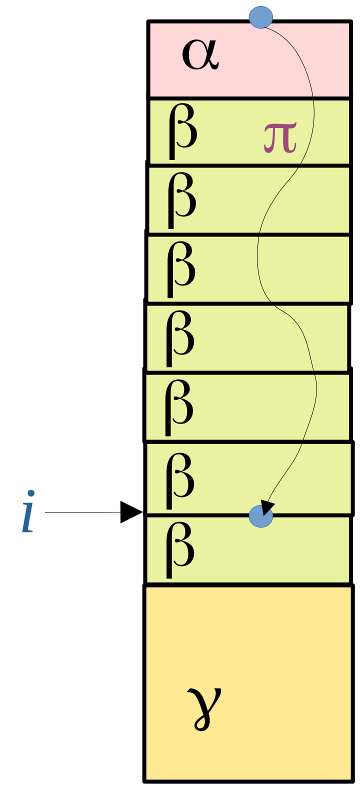


Assumption: $P \sim F$ for some finite F.

Consider the smallest e such that $r\beta^e\gamma \sim r\beta^\infty$ for all reachable r

We want to prove $e < 2^{2^{|P|^c}}$

Let $i < e$. Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\beta^i\gamma$.



Assumption: $P \sim F$ for some finite F.

Consider the smallest e such that $r\beta^e\gamma \sim r\beta^\infty$ for all reachable r

We want to prove $e < 2^{2^{|P|^c}}$

Let $i<e$. Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\beta^i\gamma$.

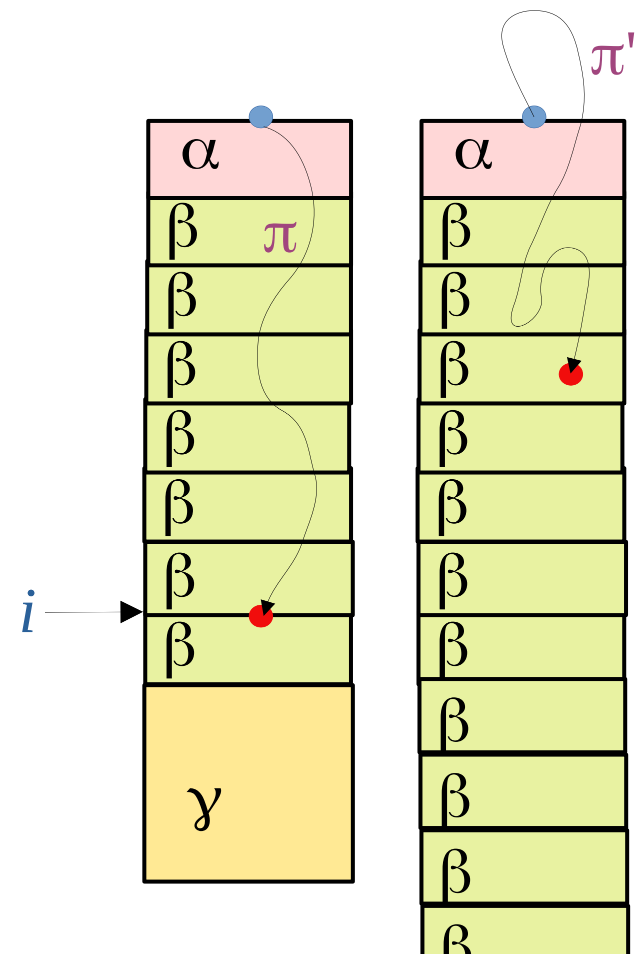
There exists a run π' from $q\alpha\beta^\infty$ visiting the same classes.

Two possibilities for the shape of π':

1) π' mostly pops the stack

it ends with $\beta'\beta^\infty$ for some small β'

→ small number of possibilities



Assumption: $P \sim F$ for some finite F.

Consider the smallest e such that $r\beta^e\gamma \sim r\beta^\infty$ for all reachable r

We want to prove $e < 2^{2^{|P|^c}}$

Let $i<e$. Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\beta^i\gamma$.

There exists a run π' from $q\alpha\beta^\infty$ visiting the same classes.

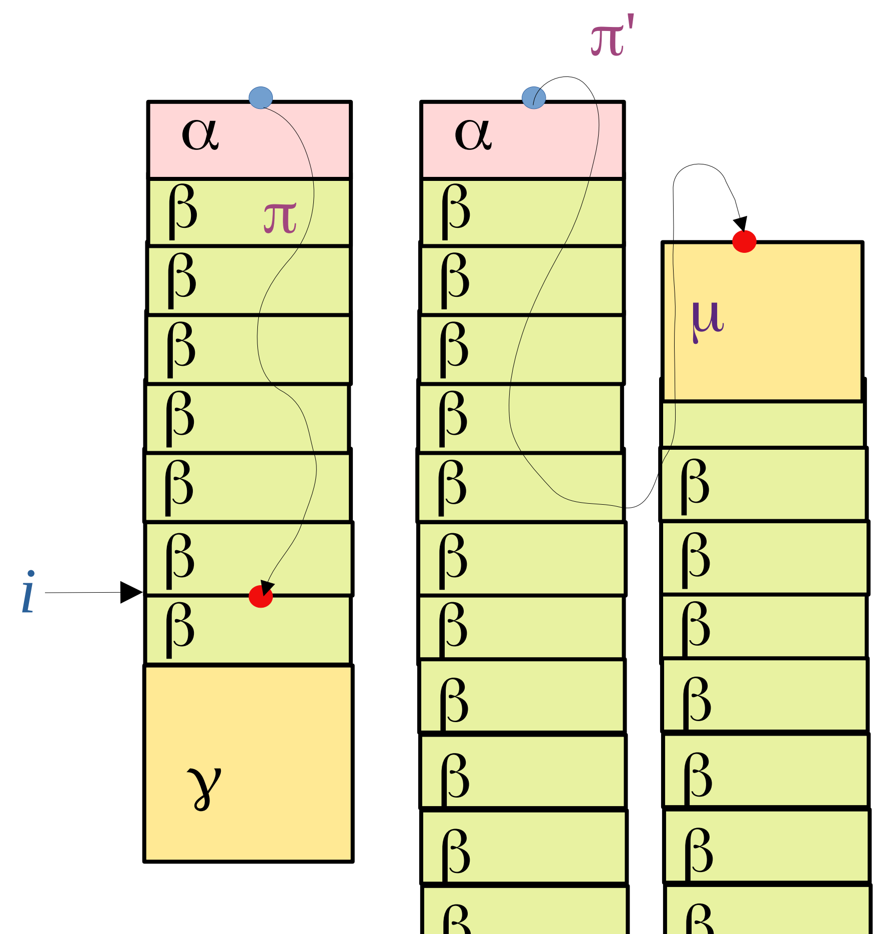
Two possibilities for the shape of π':

1) π' mostly pops the stack

it ends with $\beta'\beta^\infty$ for some small β'

→ small number of possibilities

2) π' pushes some μ of exponential size



Assumption: $P\sim F$ for some finite F.

Consider the smallest e such that $r\beta^e\gamma\sim r\beta^\infty$ for all reachable r

We want to prove $e<2^{2^{|P|^c}}$

Let $i<e$. Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\beta^i\gamma$.

There exists a run π' from $q\alpha\beta^\infty$ visiting the same classes.

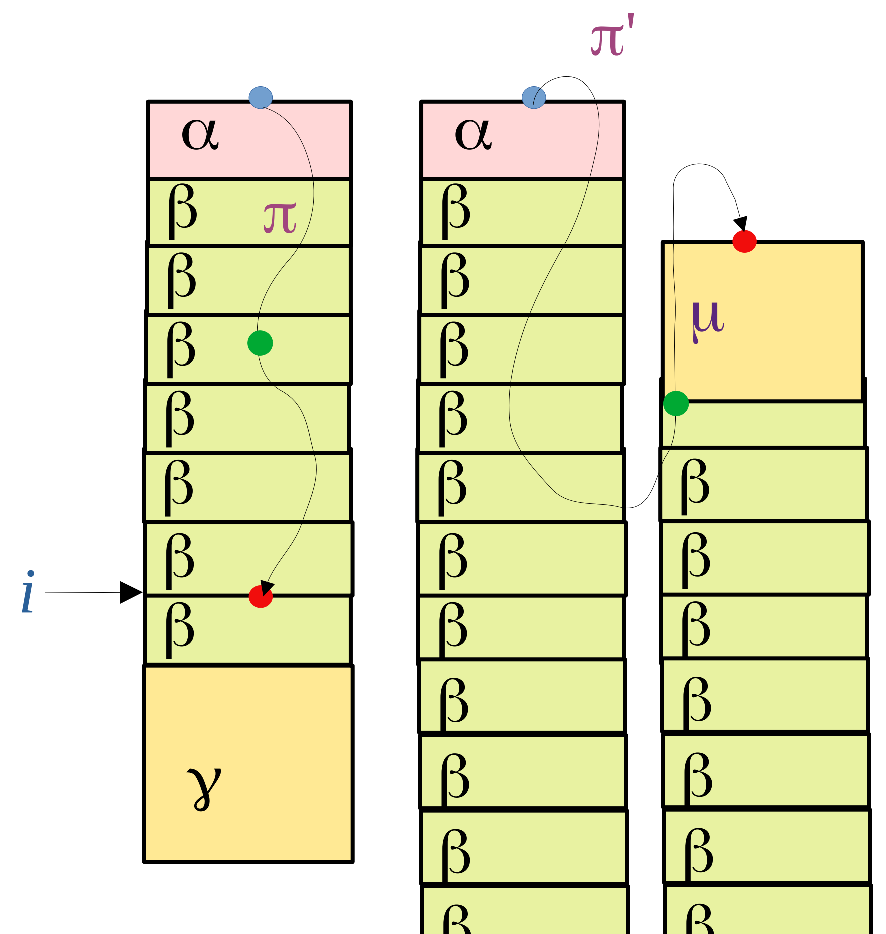
Two possibilities for the shape of π':

1) π' mostly pops the stack

it ends with $\beta'\beta^\infty$ for some small β'

➔ small number of possibilities

2) π' pushes some μ of exponential size



Assumption: $P \sim F$ for some finite F.

Consider the smallest e such that $r\beta^e\gamma \sim r\beta^\infty$ for all reachable r

We want to prove $e < 2^{2^{|P|^c}}$

Let $i < e$. Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\beta^i\gamma$.

There exists a run π' from $q\alpha\beta^\infty$ visiting the same classes.

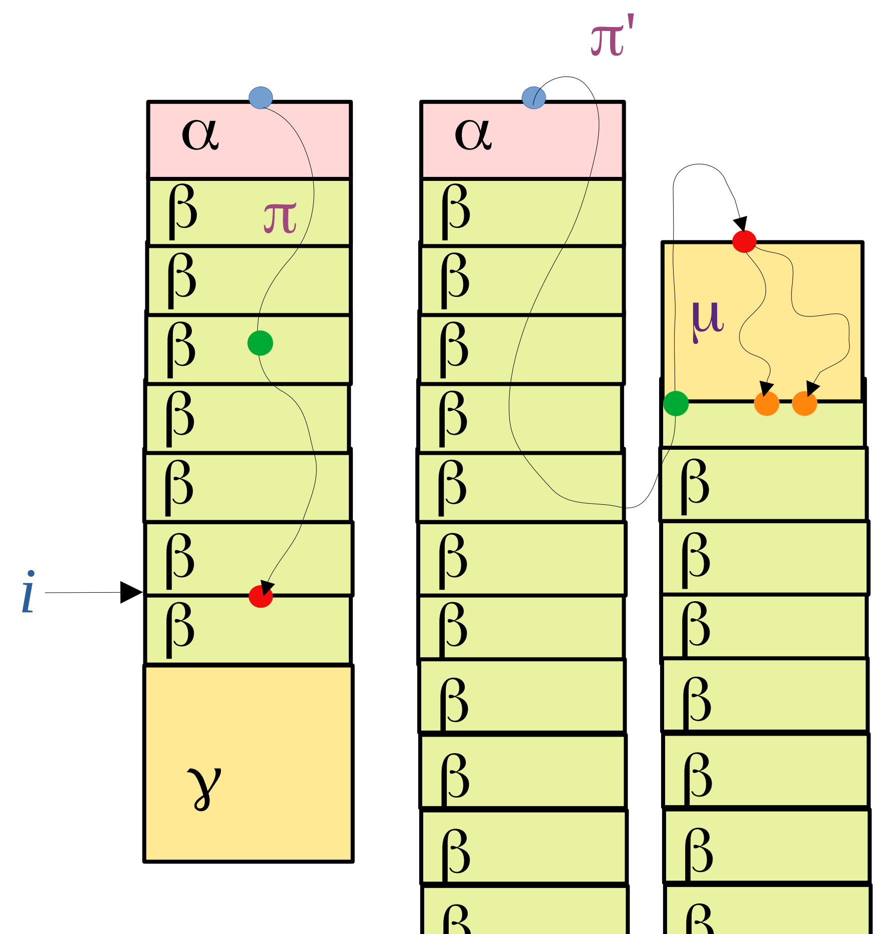
Two possibilities for the shape of π':

1) π' mostly pops the stack

it ends with $\beta'\beta^\infty$ for some small β'

→ small number of possibilities

2) π' pushes some μ of exponential size



Assumption: $P\sim F$ for some finite F.

Consider the smallest e such that $r\beta^e\gamma\sim r\beta^\infty$ for all reachable r

We want to prove $e<2^{2^{|P|^c}}$

Let $i<e$. Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\beta^i\gamma$.

There exists a run π' from $q\alpha\beta^\infty$ visiting the same classes.

Two possibilities for the shape of π':

1) π' mostly pops the stack

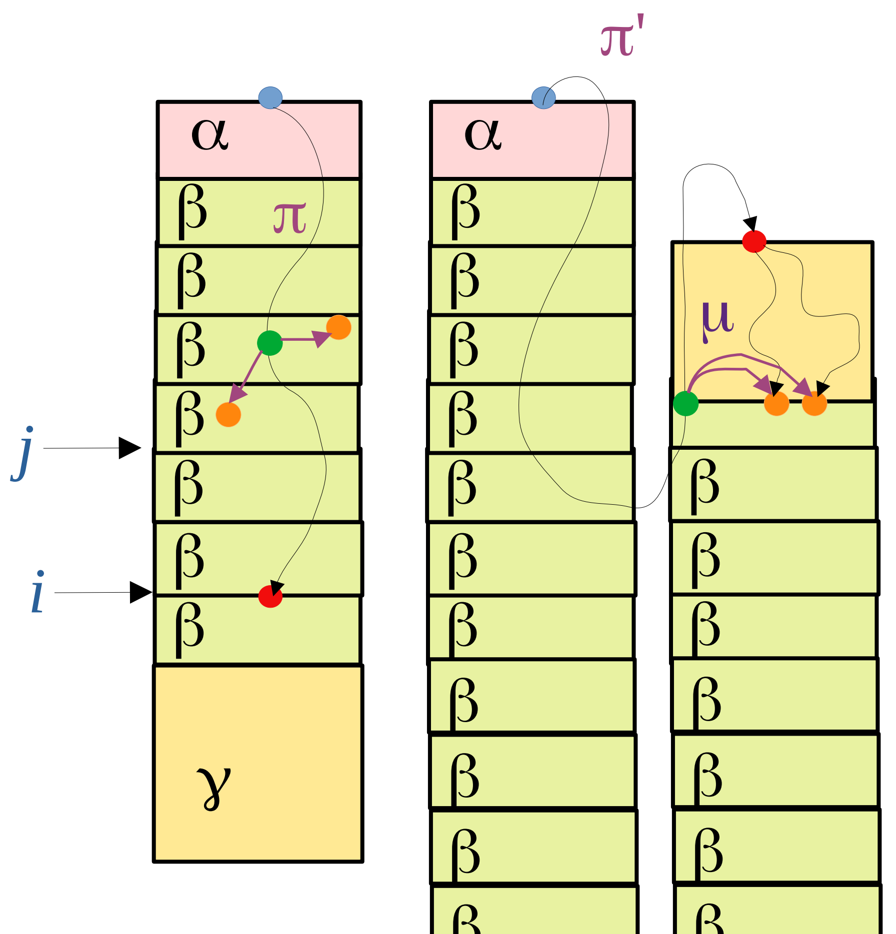
it ends with $\beta'\beta^\infty$ for some small β'

→ small number of possibilities

2) π' pushes some μ of exponential size

$[r\beta^i\gamma]$ is characterized by classes $[r\beta^j\gamma]$

and ch_i=(μ, stacks above $\beta^j\gamma$)



Assumption: $P \sim F$ for some finite F.

Consider the smallest e such that $r\beta^e\gamma \sim r\beta^\infty$ for all reachable r

We want to prove $e < 2^{2^{|P|^c}}$

Let $i<e$. Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\beta^i\gamma$.

There exists a run π' from $q\alpha\beta^\infty$ visiting the same classes.

Two possibilities for the shape of π':

1) π' mostly pops the stack

it ends with $\beta'\beta^\infty$ for some small β'

→ small number of possibilities

2) π' pushes some μ of exponential size

$[r\beta^i\gamma]$ is characterized by classes $[r\beta^j\gamma]$

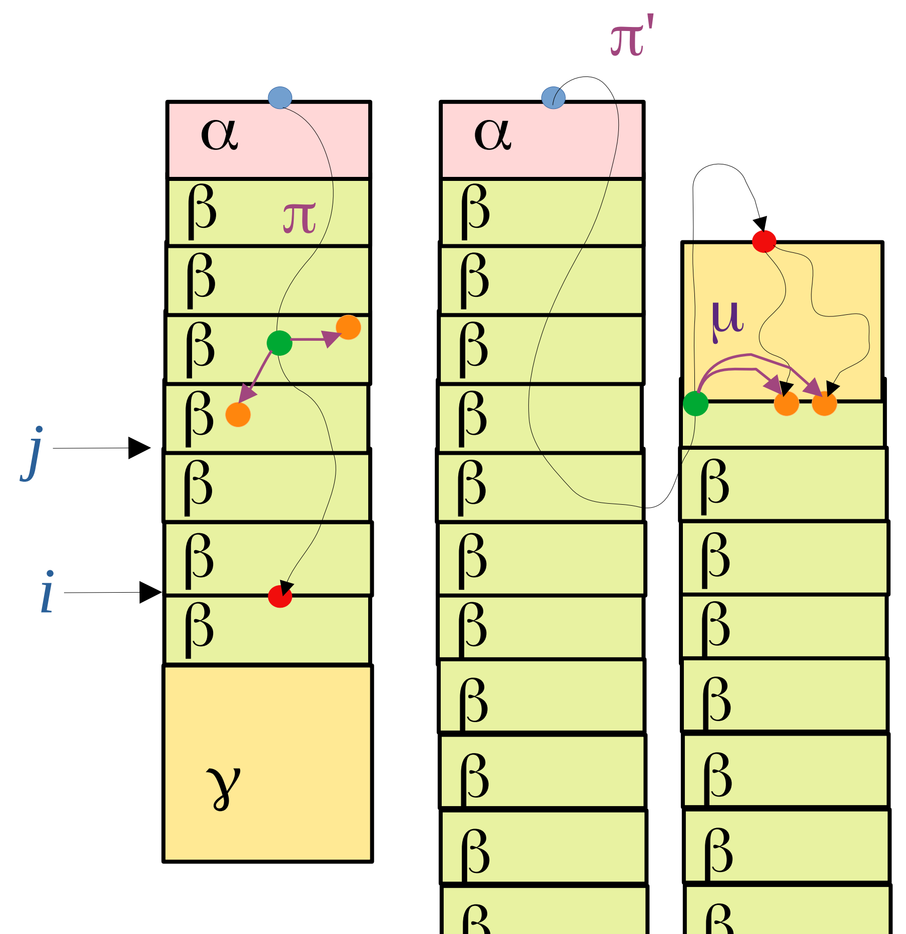
and ch_i=(μ, stacks above $\beta^j\gamma$)

We cannot have $ch_i = ch_{i'}$

(bisimulation game from $r\beta^i\gamma$, $r\beta^{i'}\gamma$

can go to $r\beta^j\gamma$, $r\beta^{j'}\gamma$, which are higher)

We obtain $e < 2^{2^{|P|^c}}$



Assumption: $P \sim F$ for some finite F.

Next step: do the same for $i=0$, when γ is not fixed

Consider a fast run π from $q\alpha\beta^e\gamma$ to $r\gamma$.

There exists a run π' from $q\alpha\beta^\infty$ visiting the same classes.

Two possibilities for the shape of π':

1) π' mostly pops the stack

it ends with $\beta'\beta^\infty$ for some small β'

→ small number of possibilities

2) π' pushes some μ of exponential size

3) $[r\gamma]$ is characterized by classes $[r\gamma]$

and $ch_\gamma=(j, \mu$, stacks above $\beta^j\gamma)$

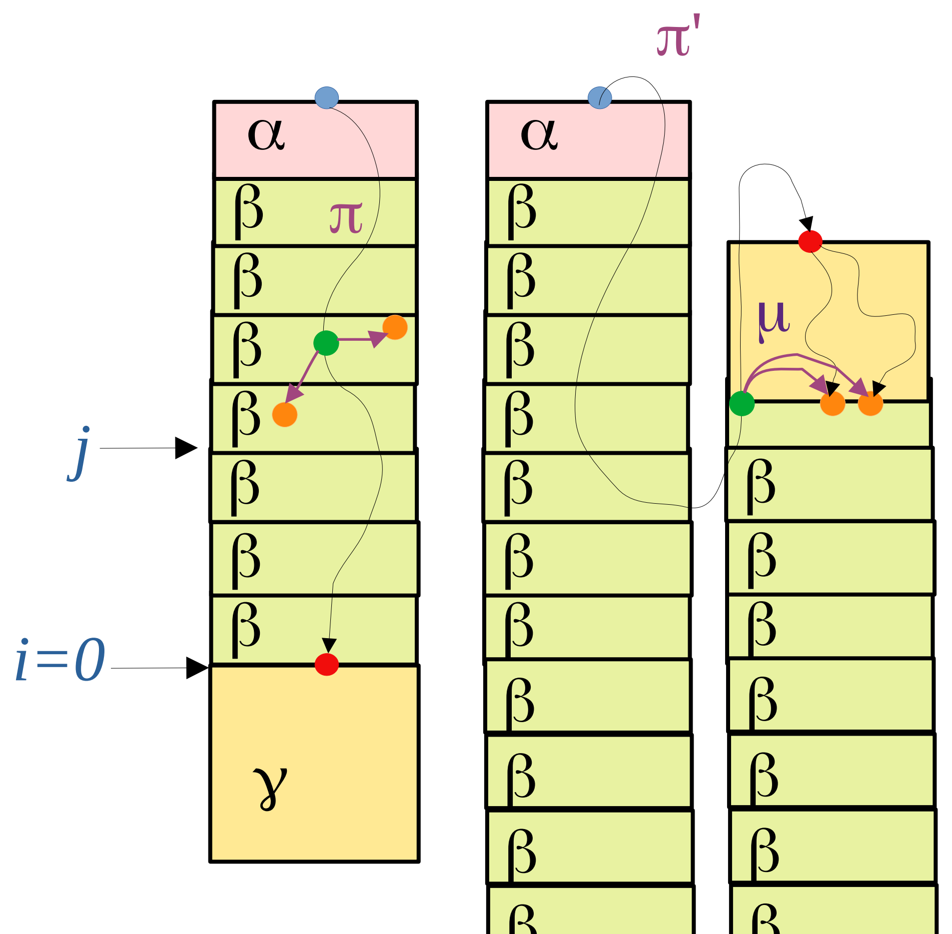
We cannot have $ch_\gamma = ch_{\gamma'}$ if $[r\gamma]\neq[r\gamma']$

(bisimulation game from $r\gamma$, $r\gamma'$

can go back to $r\gamma$, $r\gamma'$;

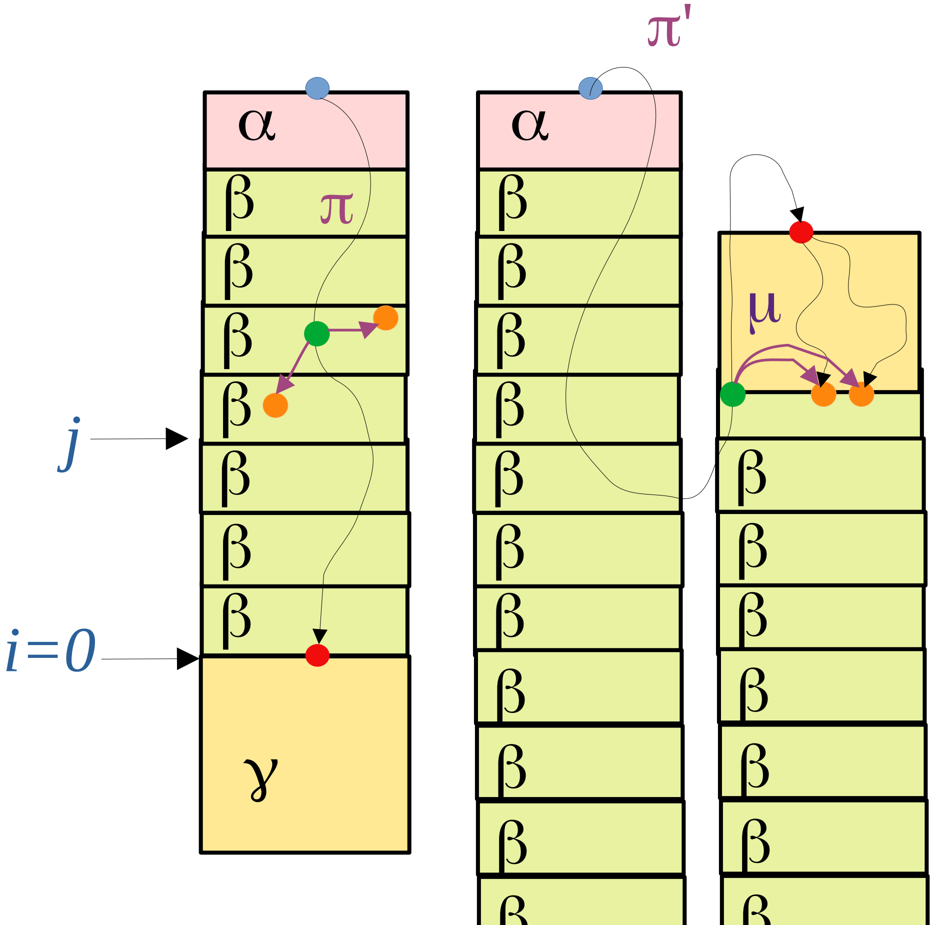
this can be repeated forever)

We obtain the theorem.



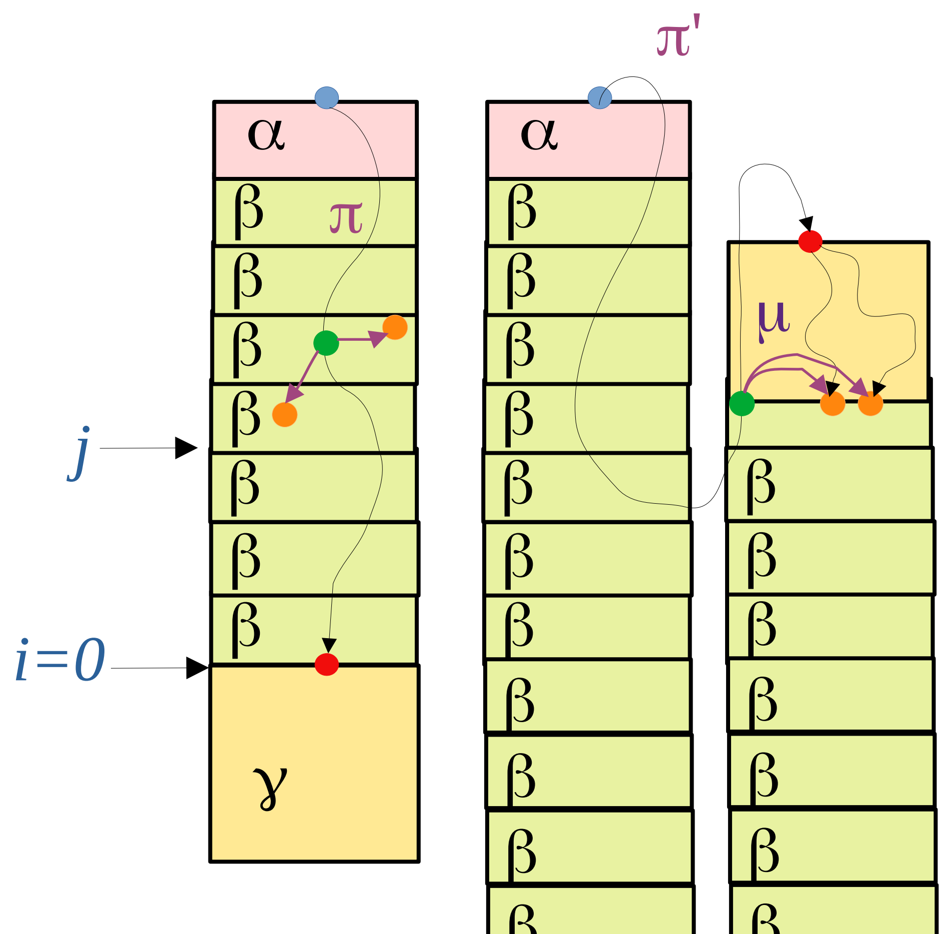
Without assumption that P for is ε-free?

- Needed e.g. to say that at least one letter is read during the loop from $r\gamma$, $r\gamma'$ to (configurations equivalent to) $r\gamma$, $r\gamma'$.



Without assumption that P for is ε-free?

- Needed e.g. to say that at least one letter is read during the loop from $r\gamma,\ r\gamma'$ to (configurations equivalent to) $r\gamma,\ r\gamma'$.
- Enough: ≥1 letter read while popping β.



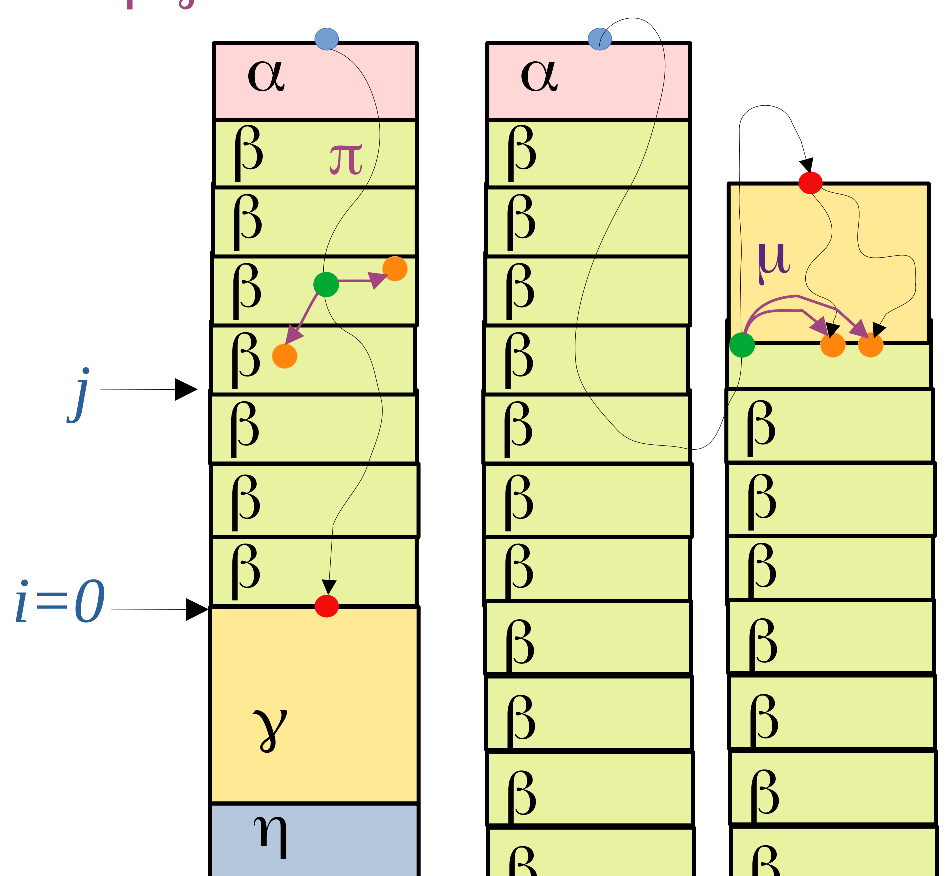
Without assumption that P for is ε-free?

- Needed e.g. to say that at least one letter is read during the loop from $r\gamma$, $r\gamma'$ to (configurations equivalent to) $r\gamma$, $r\gamma'$.
- Enough: ≥1 letter read while popping β.

General case: Decompose $\delta=\alpha\beta\gamma\eta$, where if an ε-run pops β, then it also pops γ.

- We either proceed as previously,
- or we leave the image, popping the whole $\beta^i\gamma$.

We create a nested decomposition with these properties.



Conclusion

- Bisimulation finiteness of pushdown systems with deterministic ε-transitions is 2-EXPTIME-complete
  (thus much easier than bisimulation equivalence)
- Open problem: complexity for systems without ε-transitions
  - upper bound: 2-EXPTIME
  - lower bound: EXPTIME [Kučera/Mayr 02, Srba 02]
- Generalize the proof to other classes of infinite systems