Bisimilarity in Fresh-Register Automata

Andrzej Murawski, Steven Ramsay & Nikos Tzevelekos

University of Warwick & Queen Mary University of London

Highlights of Logic, Games and Automata, Sep 2015

Supported by a Royal Academy of Engineering Research Fellowship

What this talk is about

This talk is about (fresh) register automata

which are an automata model over infinite alphabets akin to finite-state automata:

- finite-state
- registers
- freshness

We examine the complexity of their bisimilarity problem and derive bounds for different subcases/variants

[T.'11], based on RA [Kaminski & Fraencez '94]

Fresh-Register Automata (FRA)

Let $\Sigma = \{a_1, a_2, \ldots, a_n, \ldots\}$ be an infinite alphabet of names

$i \in \{0, 1, \ldots, R\}$
$X \in \mathcal{P}\{1, \ldots, R\} \cup \{\circledast\}$
$Z \subseteq \{1, \ldots, R\}$

finitely many (say R) registers

registers store names

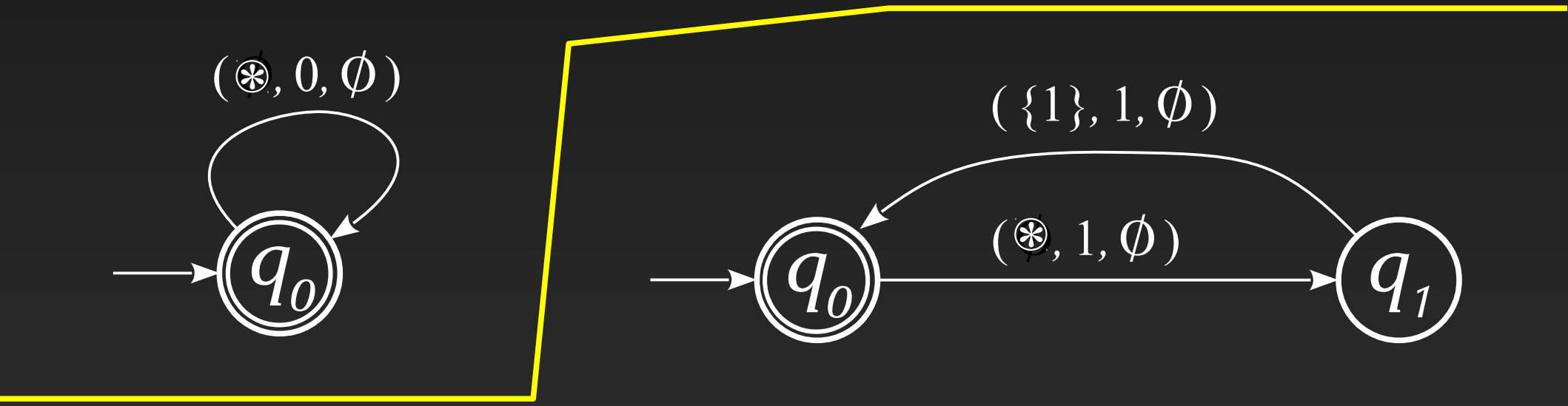
read as:

- accept name appearing *exactly* in registers X or *globally fresh* one ($X = \circledast$)
- add it to register i
- and erase registers Z

Examples

$$L_{\text{fr}} = \{\, a_1 a_2 \ldots a_n \in \Sigma^* \mid n \geq 0,\ \forall i < j.\ a_i \neq a_j \,\}$$

(all strings of pairwise distinct names)



$$L_{\text{fr}}' = \{\, a_1 a_1 a_2 a_2 \ldots a_n a_n \in \Sigma^* \mid n \geq 0,\ \forall i < j.\ a_i \neq a_j \,\}$$

(F)RA properties

- Not closed under complement & not determinisable

[Kaminski & Fraencez '94]

- Universality / equivalence undecidable

[Neven, Schwentick & Vianu '01]

- Decidable emptiness:
  - complexity depends on register “mode” (NL → NP → PSPACE)

[Sakamoto & Ikeda '00; Demri & Lazić '09]

- Bisimilarity: decidable [T.11], complexity open

Bisimilarity

Semantics of FRAs given by configuration graphs:

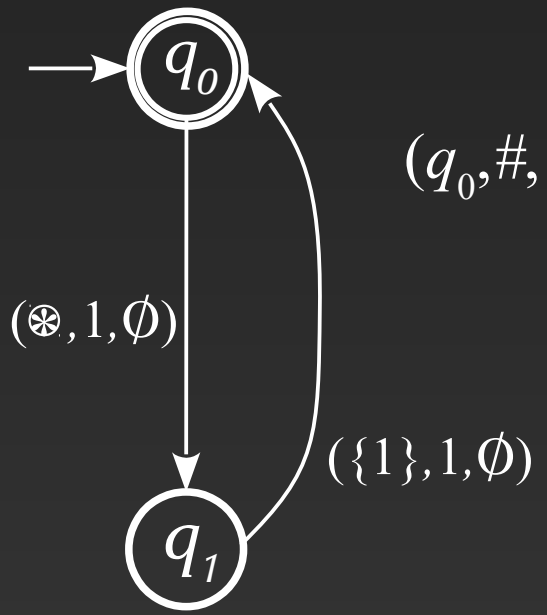
configuration

$$(q,\rho,H) \xrightarrow{a} (q',\rho',H')$$

state

register assignment: $\rho : \{1,\ldots,R\} \to \Sigma \cup \{\#\}$

history: $H \subseteq_{\text{fin}} \Sigma$



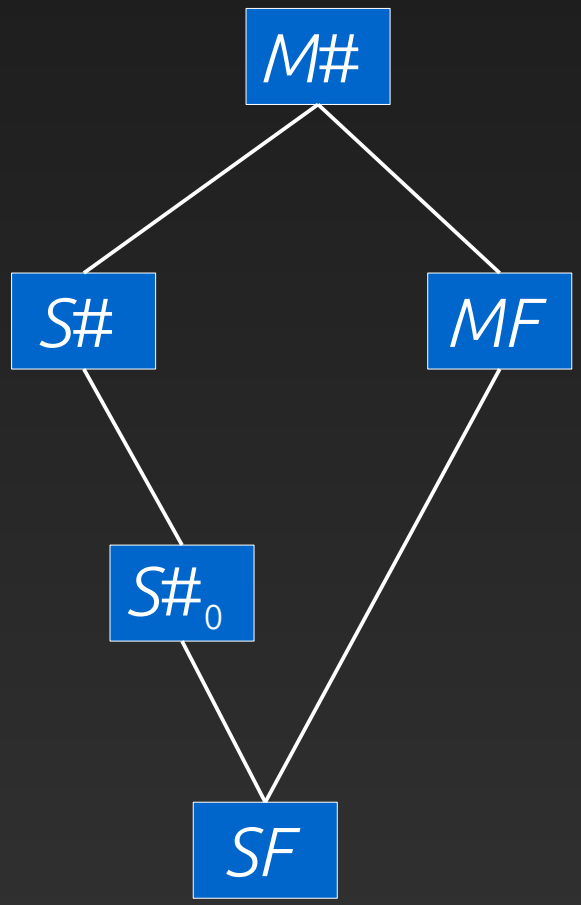
$$(q_0,\#,\emptyset) \xrightarrow{a_1} (q_1,a_1,\{a_1\}) \xrightarrow{a_1} (q_1,a_1,\{a_1\}) \xrightarrow{b_1} (q_1,b_1,\{a_1,b_1\})$$
$$\xrightarrow{a_2} (q_1,a_2,\{a_2\}) \xrightarrow{a_2} (q_1,a_2,\{a_2\}) \xrightarrow{b_2} (q_1,b_2,\{a_1,b_2\})$$
$$\ldots \qquad \ldots \qquad \ldots$$

Bisimilarity (~): between configurations

Register modes

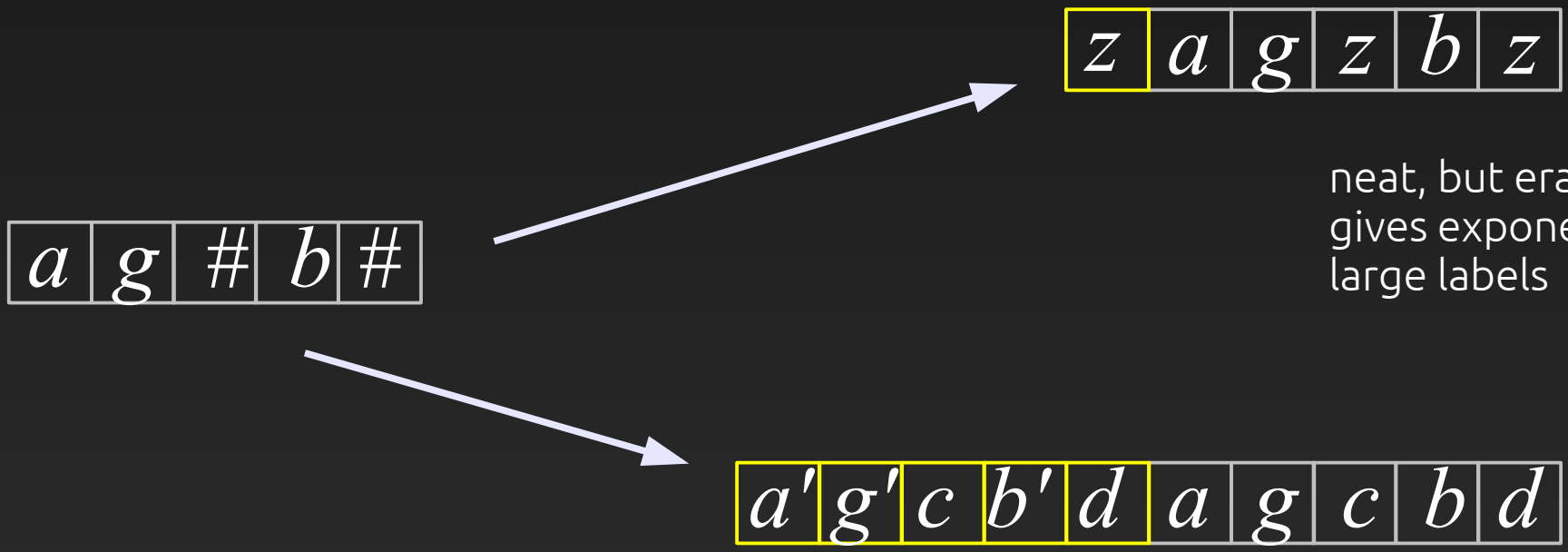
We can restrict FRAs according to specified disciplines on registers:

- *Name multiplicity* :
  - (*S*) single
  - (*M*) multiple
- *Register fullness* :
  - (*F*) full
  - ($\#_0$) initially empty
  - (#) eraseable



NOTE: # means "empty register content"

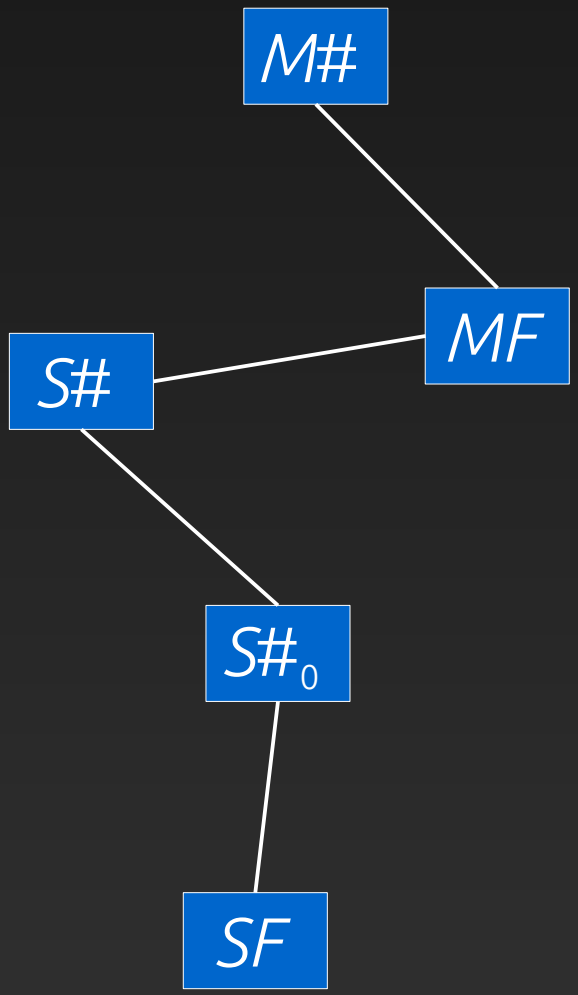
Example: (*S#*) → (*MF*)


a g # b #
z a g z b z
neat, but erasing
gives exponentially
large labels
a' g' c b' d a g c b d
concise, as each
name appears
at most twice

Complexity Picture

Duplicates :
- (*S*) single
- (*M*) multiple

Erasure :
- (*F*) full
- ($\#_0$) initially empty
- (#) eraseable



Complexity Picture

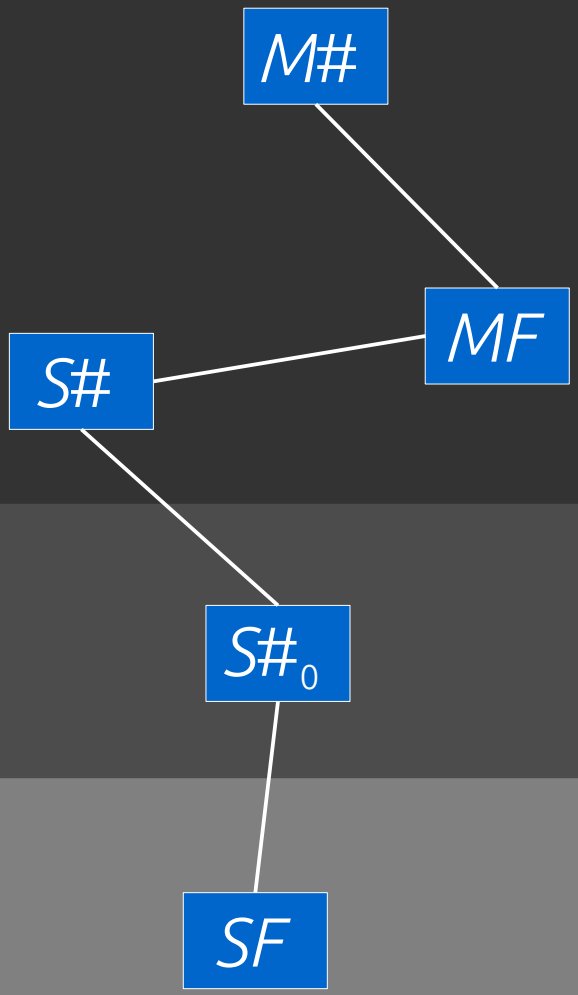
Duplicates:
- (*S*) single
- (*M*) multiple

Erasure:
- (*F*) full
- ($\#_0$) initially empty
- (#) eraseable

EXPTIME-c

PSPACE-c

NP



EXPTIME solvability

To decide bisimilarity of two configurations of size R:

- we need $2R$ names to represent all possible name matchings between them
- plus one name that is “locally fresh”
- and another one for “globally fresh”

→ $2R+2$ names, that we can encode inside states:

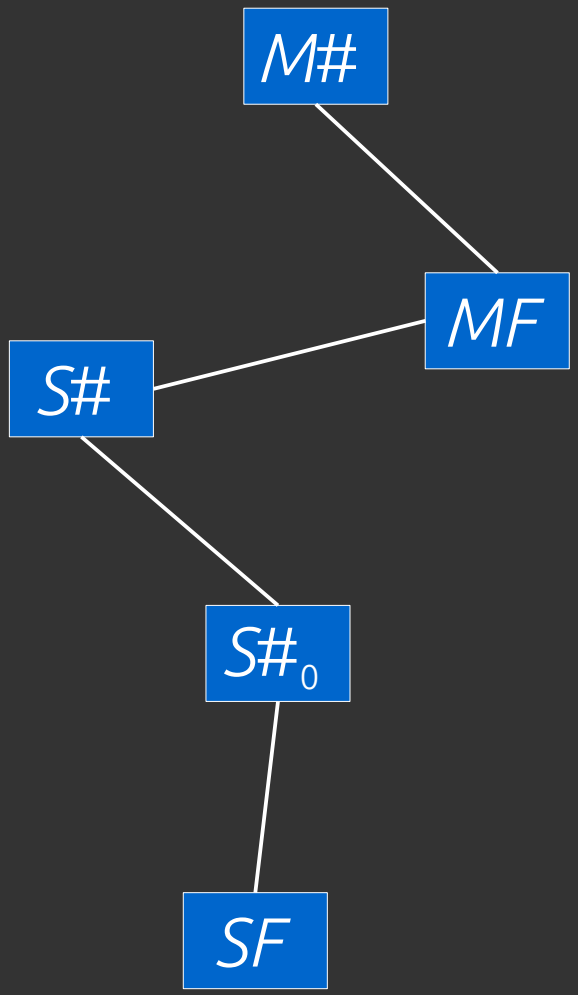
$$Q \longrightarrow Q \times (2R+2)^R$$

and bisimilarity for finite-state automata is in PTIME

Complexity Picture

Duplicates:
- (*S*) single
- (*M*) multiple

Erasure:
- (*F*) full
- ($\#_0$) initially empty
- (#) eraseable

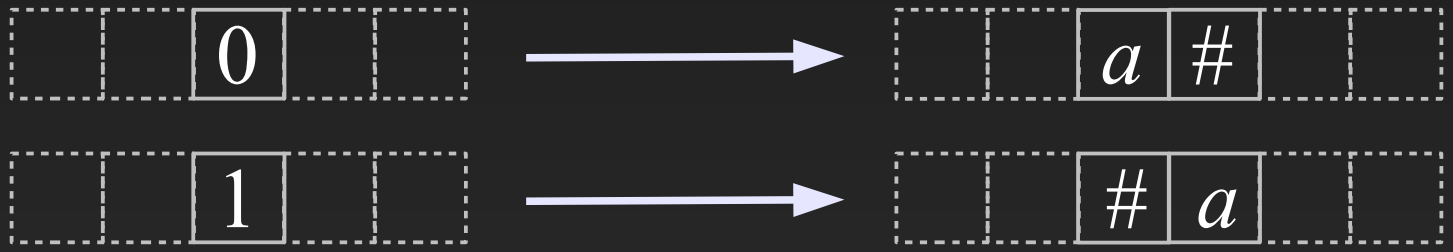


EXPTIME

EXPTIME hardness

The (*S*#) case is EXPTIME-hard:

- reduce from alternating TMs with linear-size tape (ALBA)
- model each cell by two registers:

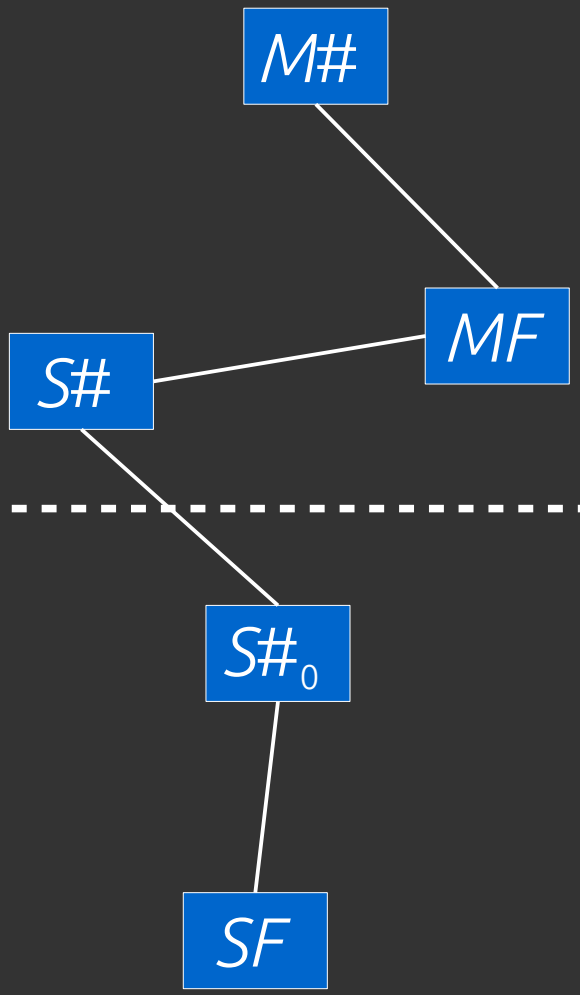


- arrange for non-bisimilarity at rejecting final states
- use Defender Forcing [Jancar & Srba '08] for existential states

Complexity Picture

Duplicates:
- (*S*) single
- (*M*) multiple

Erasure:
- (*F*) full
- ($\#_0$) initially empty
- (#) eraseable



EXPTIME

EXPTIME

The original case ($S\#_0$)

Disallowing erasures makes impossible our modelling of a linear-size tape...

In fact, the problem is PSPACE complete

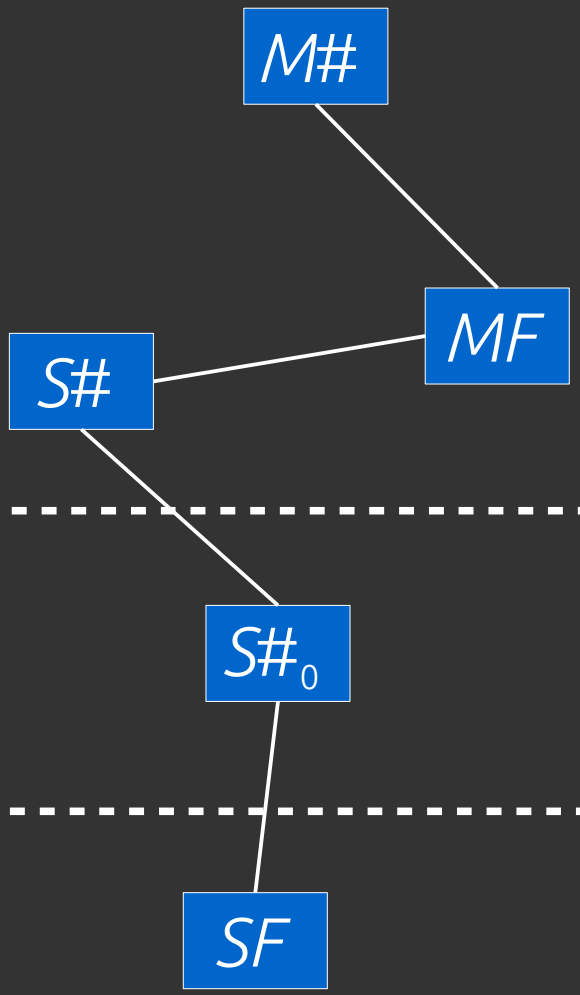
First, we can model boolean assignments (cf. write-once tape), which are enough for PSPACE-hardness:

- we reduce from QBF
- Attacker chooses universal variables
- Defender chooses existential ones (via forcing)

Complexity Picture

Duplicates :
- (*S*) single
- (*M*) multiple

Erasure :
- (*F*) full
- ($\#_0$) initially empty
- (#) eraseable



EXPTIME

EXPTIME

PSPACE

PSPACE solvability: difficult

Our best bet is APTIME = PSPACE

- problem: while we cannot simulate a linear tape, we still have a lot of combinations of configurations!
  - even for RAs: exponentially many

We look into internal symmetries of FRAs:

- symbolic reasoning: we are only interested in how configurations are related, not their actual content
- group representations: we express these interrelations compactly via permutation groups
- bounded history: it suffices to consider histories of size up to $2R$

PSPACE solvability

$$\overset{0}{\sim} \supseteq \overset{1}{\sim} \supseteq \overset{2}{\sim} \supseteq \ldots \supseteq \overset{i}{\sim} \supseteq \ldots \quad \text{and} \quad \sim \; = \bigcup_{i \in \omega} \overset{i}{\sim}$$

Reasoning symbolically:

- each decrease in the indexed chain can be traced back to one of polynomially many factors!

use the fact that strict subgroup chains have bounded length

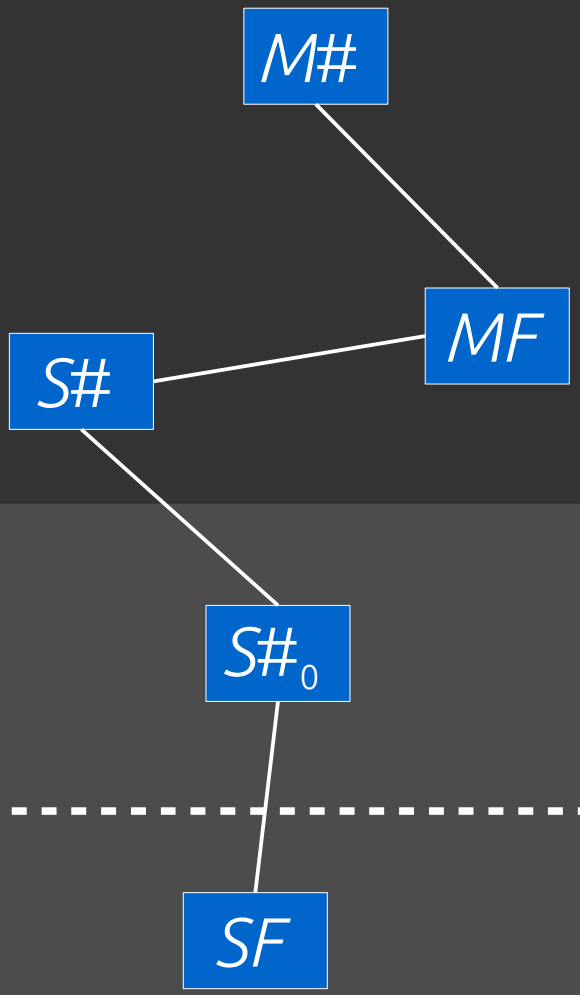
This means there is a final polynomial-size i

- gives a polynomial bound for the bisimulation game, hence APTIME

Complexity Picture

Duplicates :
- (*S*) single
- (*M*) multiple

Erasure :
- (*F*) full
- ($\#_0$) initially empty
- (#) eraseable



EXPTIME

PSPACE

PSPACE

Complexity Picture

Duplicates:
- (*S*) single
- (*M*) multiple

Erasure:
- (*F*) full
- ($\#_0$) initially empty
- (#) eraseable

EXPTIME-c

PSPACE-c

NP