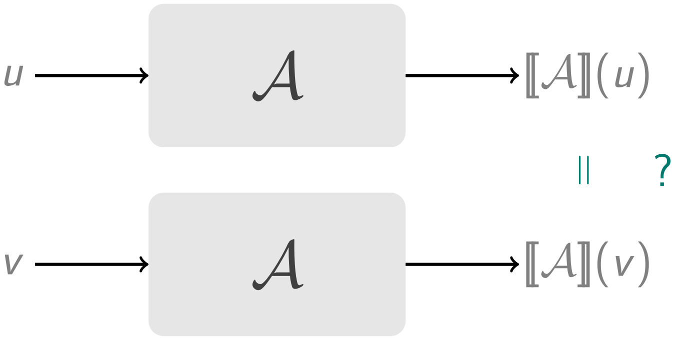
Max-plus automata and Tropical identities

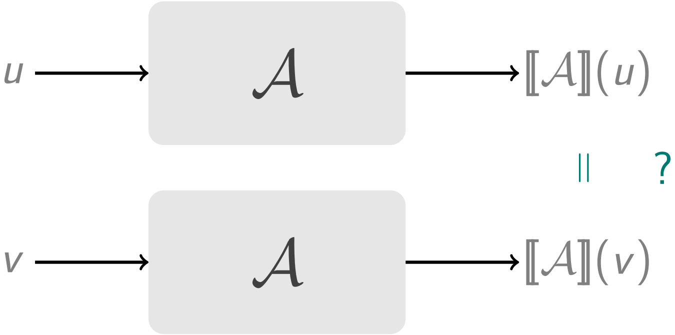
Laure Daviaud
University of Warwick

Oxford, 01-02-2018

A natural and fundamental question:



A natural and fundamental question:



Which pairs of inputs can be distinguished
by a given computational model?

Given a class $\mathcal{C}$ of computational models:

Given a class $\mathcal{C}$ of computational models:

1. For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?

Given a class $\mathcal{C}$ of computational models:

1. For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
2. Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?

Given a class $\mathcal{C}$ of computational models:

1. For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
2. Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?
3. Minimal size to distinguish two given input words?

Boolean automata

Finite alphabet $A = \{a, b\}$
Set of words A^*

Boolean automata

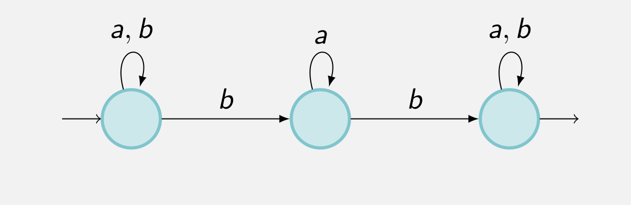
Finite alphabet $A = \{a, b\}$
Set of words A^*

Check if a word has at least two b's.

Boolean automata

Finite alphabet $A = \{a, b\}$
Set of words A^*

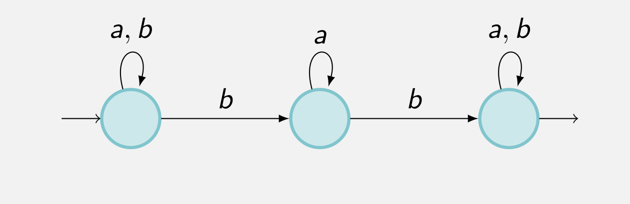
Check if a word has at least two b's.



Boolean automata

Finite alphabet $A = \{a, b\}$
Set of words A^*

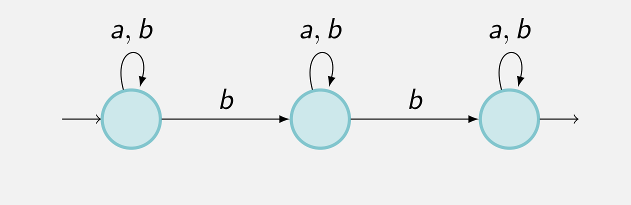
Check if a word has at least two b's.



A word is *accepted* by the automaton if there is a path labelled by the word from an initial state to a final state.

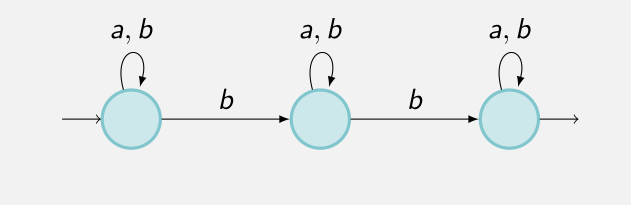
Boolean Automata

$[\![\mathcal{A}]\!] : A^* \to \{Acc, Rej\}$



Boolean Automata

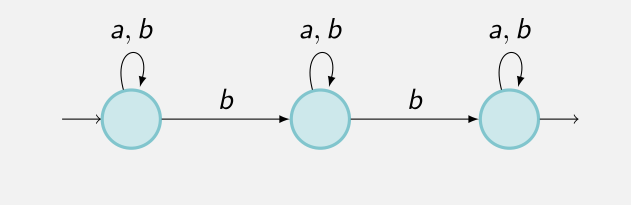
$[\![\mathcal{A}]\!] : A^* \to \{Acc, Rej\}$



1 For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
$\to$ Yes

Boolean Automata

$[\![\mathcal{A}]\!] : A^* \to \{Acc, Rej\}$

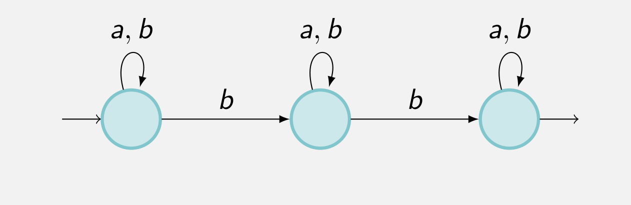


1. For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
   $\to$ Yes

2. Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?
   $\to$ No

Boolean Automata

$[\![\mathcal{A}]\!] : A^* \to \{Acc, Rej\}$

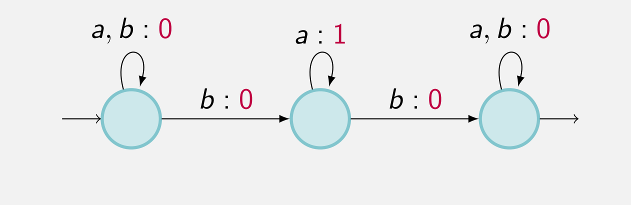


1 For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
$\to$ Yes

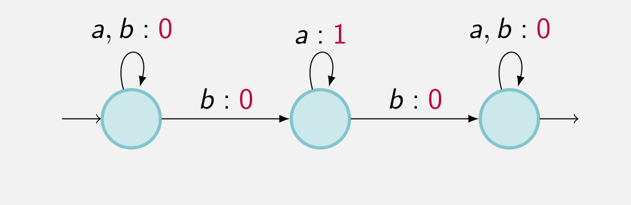
2 Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?
$\to$ No

3 Minimal size to distinguish two given input words?
$\to$ Some lower bounds [Robson], Profinite theory...

Max-plus automata

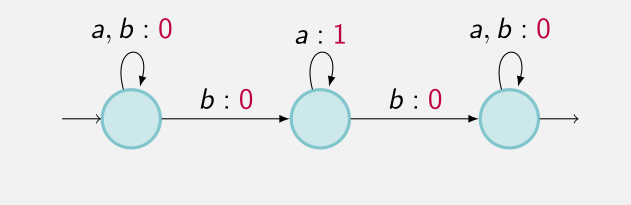


Max-plus automata



Syntax : Non deterministic finite automaton for which each transition is labelled by a non negative integer (weight).

Max-plus automata



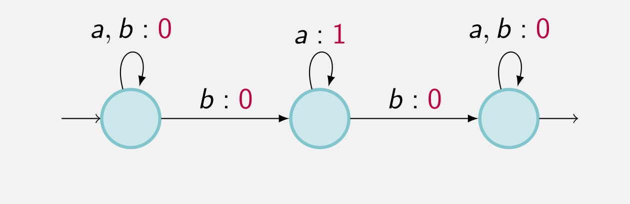
Syntax : Non deterministic finite automaton for which each transition is labelled by a non negative integer (weight).

Semantic : Weight of a run: sum of the weights of the transitions.

$A^+ \to \mathbb{N} \cup \{-\infty\}$

$w \mapsto$ Max of the weights of the accepting runs labelled by w
($-\infty$ if no such run)

Max-plus automata



Syntax : Non deterministic finite automaton for which each transition is labelled by a non negative integer (weight).

Semantic : Weight of a run: sum of the weights of the transitions.

$A^+ \to \mathbb{N} \cup \{-\infty\}$

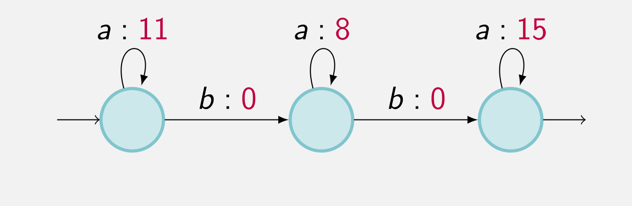
$w \mapsto$ Max of the weights of the accepting runs labelled by w

($-\infty$ if no such run)

$$a^{n_0} b a^{n_1} b \cdots b a^{n_{k+1}} \mapsto \max(n_1, \ldots, n_k)$$

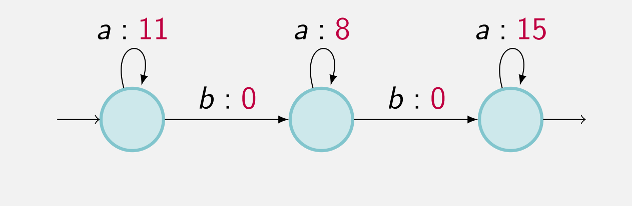
Weighted automata *[Schützenberger]*

$[\![\mathcal{A}]\!] : A^* \to S$



Weighted automata *[Schützenberger]*

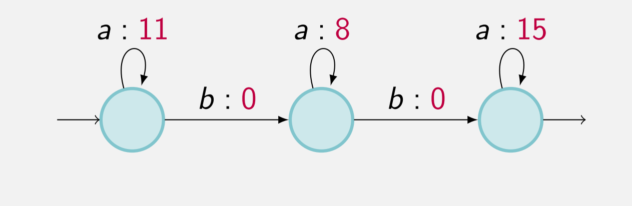
$[\![\mathcal{A}]\!] : A^* \to S$



Semiring $(S, \oplus, \otimes)$: transitions are weighted by elements of S

Weighted automata *[Schützenberger]*

$[\![\mathcal{A}]\!] : A^* \to S$



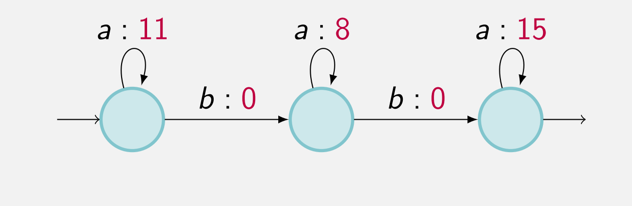
Semiring $(S, \oplus, \otimes)$: transitions are weighted by elements of S

Paths: $\otimes$

Non-determinism: $\oplus$

Weighted automata *[Schützenberger]*

$[\![\mathcal{A}]\!] : A^* \to S$



Semiring $(S, \oplus, \otimes)$: transitions are weighted by elements of S

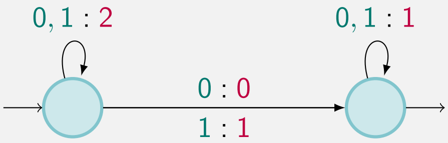
Paths: $\otimes$ Non-determinism: $\oplus$

$$[\![\mathcal{A}]\!] : w \mapsto \bigoplus_{\substack{\rho \text{ accepting path} \\ \text{labelled by } w}} \left(\rho_1 \otimes \rho_2 \otimes \cdots \otimes \rho_{|w|}\right)$$

Automata weighted over $(\mathbb{R}, +, \times)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{R}$

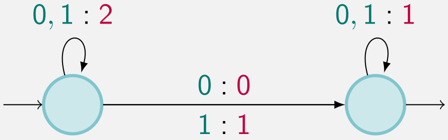
An example with $A = \{0, 1\}$



Automata weighted over $(\mathbb{R}, +, \times)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{R}$

An example with $A = \{0, 1\}$

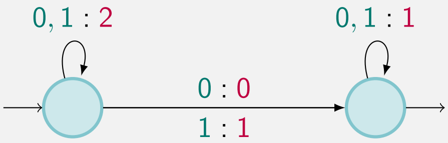


$100101 \mapsto 2^0 + 0 + 0 + 2^3 + 0 + 2^5$

Automata weighted over $(\mathbb{R}, +, \times)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{R}$

An example with $A = \{0, 1\}$



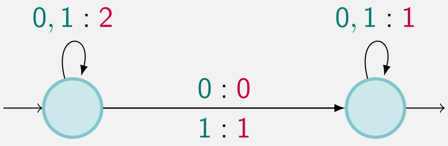
$100101 \mapsto 2^0 + 0 + 0 + 2^3 + 0 + 2^5$

1 For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
$\to$ Yes

Automata weighted over $(\mathbb{R}, +, \times)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{R}$

An example with $A = \{0, 1\}$



$100101 \mapsto 2^0 + 0 + 0 + 2^3 + 0 + 2^5$

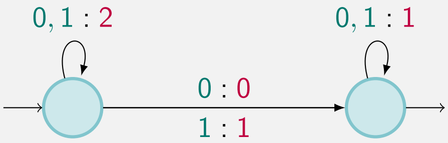
1 For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
$\to$ Yes

2 Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?
$\to$ Yes

Automata weighted over $(\mathbb{R}, +, \times)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{R}$

An example with $A = \{0, 1\}$



$100101 \mapsto 2^0 + 0 + 0 + 2^3 + 0 + 2^5$

1. For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
   $\to$ Yes
2. Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?
   $\to$ Yes
3. Minimal size to distinguish two given input words?
   $\to$ 1 or 2 states

Max-plus automata

Semiring $(\mathbb{N} \cup \{-\infty\}, \max, +)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{N} \cup \{-\infty\}$

$$[\![\mathcal{A}]\!] : w \mapsto \max_{\substack{\rho \text{ accepting path} \\ \text{labelled by } w}} \left(\rho_1 + \rho_2 + \cdots + \rho_{|w|}\right)$$

Max-plus automata

Semiring $(\mathbb{N} \cup \{-\infty\}, \max, +)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{N} \cup \{-\infty\}$

$$[\![\mathcal{A}]\!] : w \mapsto \max_{\substack{\rho \text{ accepting path} \\ \text{labelled by } w}} \left(\rho_1 + \rho_2 + \cdots + \rho_{|w|}\right)$$

1 For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
$\to$ Yes

Max-plus automata

Semiring $(\mathbb{N} \cup \{-\infty\}, \max, +)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{N} \cup \{-\infty\}$

$$[\![\mathcal{A}]\!] : w \mapsto \max_{\substack{\rho \text{ accepting path} \\ \text{labelled by } w}} \left(\rho_1 + \rho_2 + \cdots + \rho_{|w|}\right)$$

1. For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
   $\to$ Yes

2. Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?
   $\to$ No

Max-plus automata

Semiring $(\mathbb{N} \cup \{-\infty\}, \max, +)$

$[\![\mathcal{A}]\!] : A^* \to \mathbb{N} \cup \{-\infty\}$

$$[\![\mathcal{A}]\!] : w \mapsto \max_{\substack{\rho \text{ accepting path} \\ \text{labelled by } w}} \left(\rho_1 + \rho_2 + \cdots + \rho_{|w|}\right)$$

1. For all $u \neq v$, is there $\mathcal{A} \in \mathcal{C}$ which distinguishes u and v?
   $\to$ Yes

2. Is there $\mathcal{A} \in \mathcal{C}$ which distinguishes all pairs $u \neq v$?
   $\to$ No

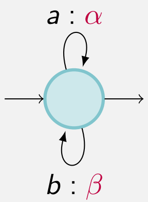
3. Minimal size to distinguish two given input words?
   $\to$??????

Given a positive integer n,
are there $u \neq v$ such that
for all max-plus automata $\mathcal{A}$ with at most n states:

$$[\![\mathcal{A}]\!](u) = [\![\mathcal{A}]\!](v) \quad ?$$

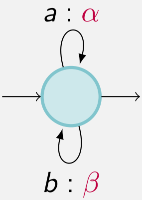
If $n = 1$

$A = \{a, b\}$



If $n = 1$

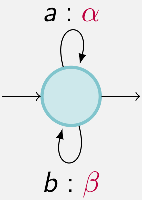
$A = \{a, b\}$



$w \mapsto \alpha|w|_a + \beta|w|_b$

If $n = 1$

$A = \{a, b\}$



$w \mapsto \alpha|w|_a + \beta|w|_b$

Max-plus automata with one state can distinguish words with different contents (in particular different lengths), and only these ones.

If $n = 2$ or $n = 3$

There exist pairs of distinct words with the same values for all automata with at most 3 states...

But we do not know much more.

If $n = 2$ or $n = 3$

There exist pairs of distinct words with the same values for all automata with at most 3 states...

But we do not know much more.

2 states [Izhakian, Margolis] - words of length 20

If $n = 2$ or $n = 3$

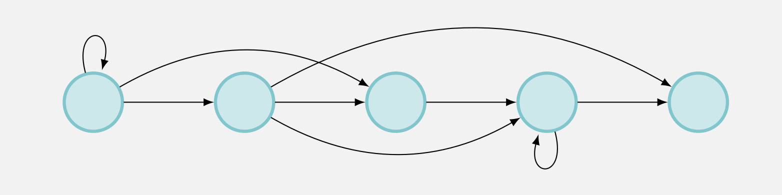
There exist pairs of distinct words with the same values for all automata with at most 3 *states...*

But we do not know much more.

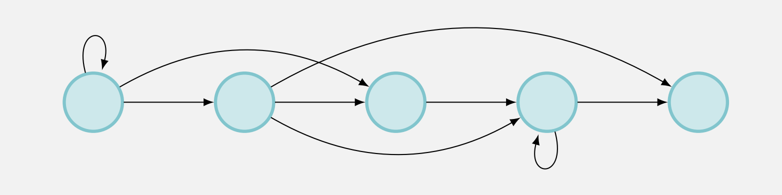
2 states [Izhakian, Margolis] - words of length 20

3 states [Shitov] - words of length 1795308

Triangular automata



Triangular automata



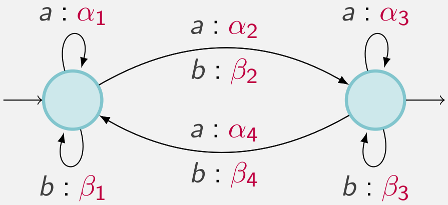
Theorem [Izhakian]

For all n, there exist a pair of distinct words $u \neq v$ such that for all triangular automata $\mathcal{A}$ with at most n states,

$$[\![\mathcal{A}]\!](u) = [\![\mathcal{A}]\!](v)$$

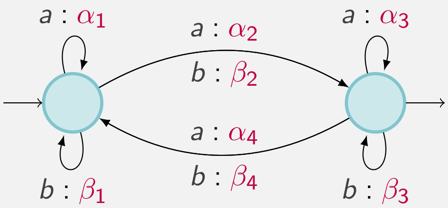
Let's go back to automata with 2 states

$A = \{a, b\}$



Let's go back to automata with 2 states

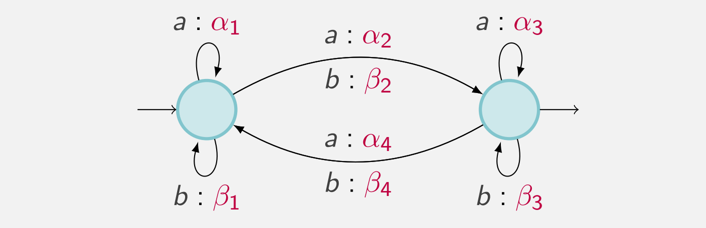
$A = \{a, b\}$



First attempt: Restrict the class of automata we have to consider

Let's go back to automata with 2 states

$A = \{a, b\}$

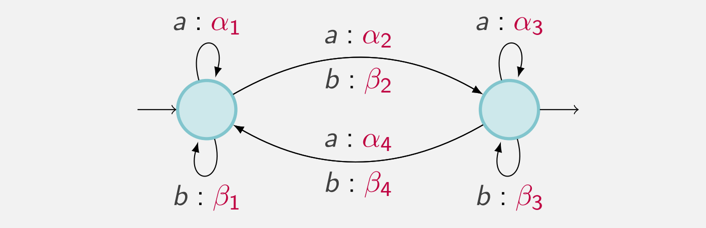


First attempt: Restrict the class of automata we have to consider

- $\mathbb{R} \longrightarrow \mathbb{Q} \longrightarrow \mathbb{Z} \longrightarrow \mathbb{N}$

Let's go back to automata with 2 states

$A = \{a, b\}$

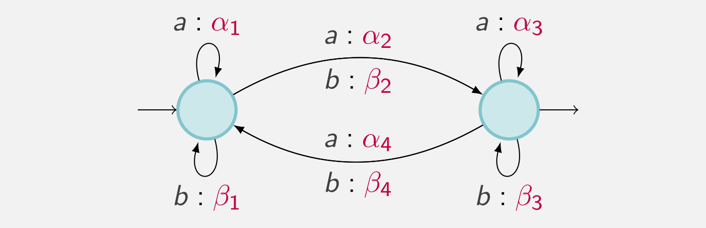


First attempt: Restrict the class of automata we have to consider

- $\mathbb{R} \longrightarrow \mathbb{Q} \longrightarrow \mathbb{Z} \longrightarrow \mathbb{N}$
- Complete automaton

Let's go back to automata with 2 states

$A = \{a, b\}$

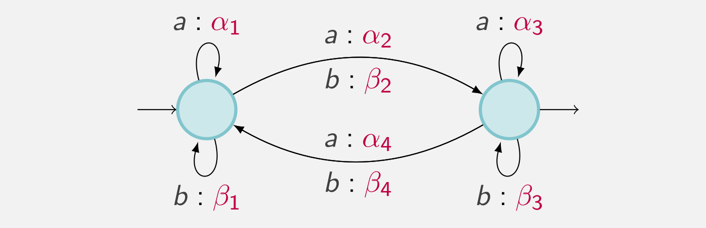


First attempt: Restrict the class of automata we have to consider

- $\mathbb{R} \longrightarrow \mathbb{Q} \longrightarrow \mathbb{Z} \longrightarrow \mathbb{N}$
- Complete automaton
- Only one initial and one final states

Let's go back to automata with 2 states

$A = \{a, b\}$



First attempt: Restrict the class of automata we have to consider

- $\mathbb{R} \longrightarrow \mathbb{Q} \longrightarrow \mathbb{Z} \longrightarrow \mathbb{N}$
- Complete automaton
- Only one initial and one final states
- Reduce the number of parameters

Let's go back to automata with 2 states

Second attempt: Give a list of criteria which can be checked

Let's go back to automata with 2 states

Second attempt: Give a list of criteria which can be checked

- Content, length

Let's go back to automata with 2 states

Second attempt: Give a list of criteria which can be checked

- Content, length
- ...

Let's go back to automata with 2 states

Second attempt: Give a list of criteria which can be checked

- Content, length
- ...

Theorem [D., Johnson] - counter-example to a conjecture of Izhakian

There are two pairs of distinct words of minimal length which cannot be distinguished by any max-plus automata with two states:

$$a^2b^3a^3babab^3a^2 = a^2b^3ababa^3b^3a^2 \text{ and } ab^3a^4baba^2b^3a = ab^3a^2baba^4b^3a$$

Let's go back to automata with 2 states

Second attempt: Give a list of criteria which can be checked

- Content, length
- ...

Theorem [D., Johnson] - counter-example to a conjecture of Izhakian

There are two pairs of distinct words of minimal length which cannot be distinguished by any max-plus automata with two states:

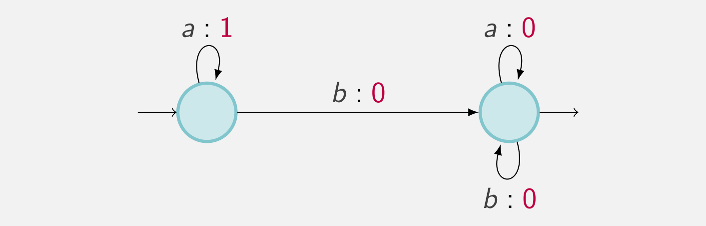
$$a^2b^3a^3babab^3a^2 = a^2b^3ababa^3b^3a^2 \text{ and } ab^3a^4baba^2b^3a = ab^3a^2baba^4b^3a$$

$\longrightarrow$ Eliminate the shortest pairs by using the list of criteria
$\longrightarrow$ Checking the pairs directly using the restrictions

A closer look at the list of criteria

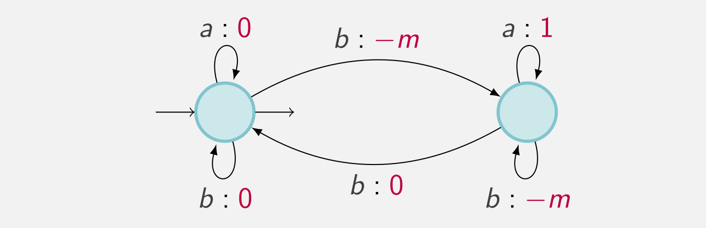
A closer look at the list of criteria

- First and last blocks



A closer look at the list of criteria

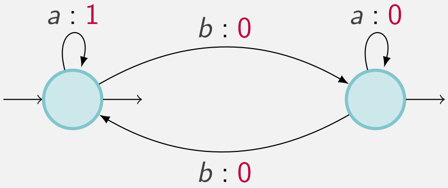
- First and last blocks
- Bloc-permutation



A closer look at the list of criteria

- First and last blocks
- Bloc-permutation
- "Counting modulo 2" criteria

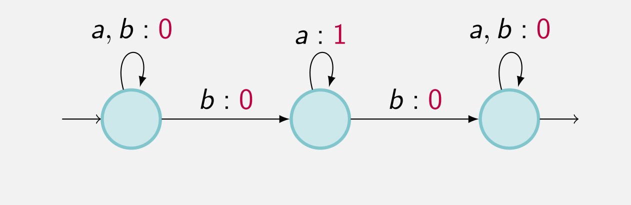
Number of *a*'s after an even number of *b*'s



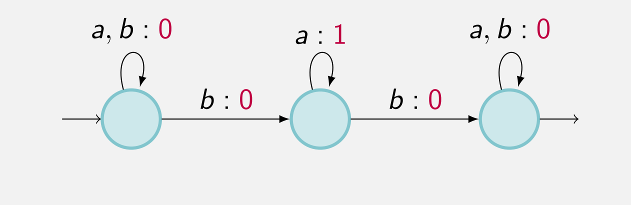
A closer look at the list of criteria

- First and last blocks
- Bloc-permutation
- “Counting modulo 2” criteria
- Triangular automata with two states

Matrix representation

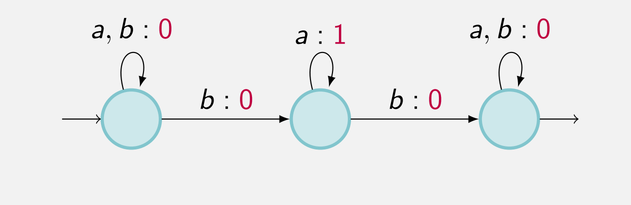


Matrix representation



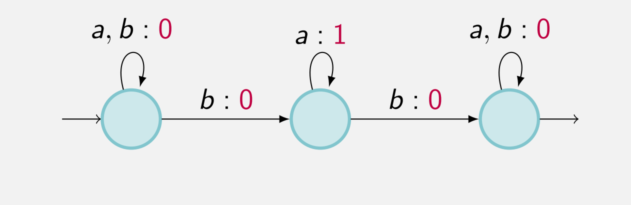
$$\mu(a) = \begin{pmatrix} 0 & -\infty & -\infty \\ -\infty & 1 & -\infty \\ -\infty & -\infty & 0 \end{pmatrix}$$

Matrix representation



$$\mu(a) = \begin{pmatrix} 0 & -\infty & -\infty \\ -\infty & 1 & -\infty \\ -\infty & -\infty & 0 \end{pmatrix} \quad \mu(b) = \begin{pmatrix} 0 & 0 & -\infty \\ -\infty & -\infty & 0 \\ -\infty & -\infty & 0 \end{pmatrix}$$

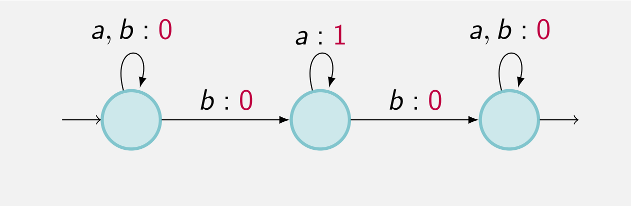
Matrix representation



$$\mu(a) = \begin{pmatrix} 0 & -\infty & -\infty \\ -\infty & 1 & -\infty \\ -\infty & -\infty & 0 \end{pmatrix} \quad \mu(b) = \begin{pmatrix} 0 & 0 & -\infty \\ -\infty & -\infty & 0 \\ -\infty & -\infty & 0 \end{pmatrix}$$

$$I = \begin{pmatrix} 0 & -\infty & -\infty \end{pmatrix} \quad F = \begin{pmatrix} -\infty \\ -\infty \\ 0 \end{pmatrix}$$

Matrix representation



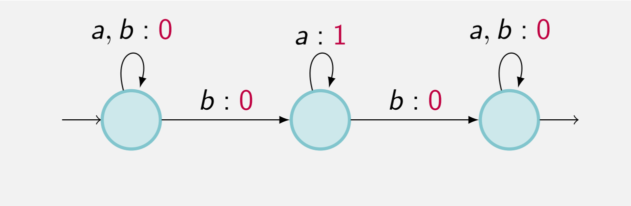
$$\mu(a) = \begin{pmatrix} 0 & -\infty & -\infty \\ -\infty & 1 & -\infty \\ -\infty & -\infty & 0 \end{pmatrix} \quad \mu(b) = \begin{pmatrix} 0 & 0 & -\infty \\ -\infty & -\infty & 0 \\ -\infty & -\infty & 0 \end{pmatrix}$$

$$I = \begin{pmatrix} 0 & -\infty & -\infty \end{pmatrix} \quad F = \begin{pmatrix} -\infty \\ -\infty \\ 0 \end{pmatrix}$$

$\mu(w)_{i,j}$ = max of the weights of the runs from i to j labelled by w

$$[\![\mathcal{A}]\!](w) = I\mu(w)F$$

Matrix representation



$$\mu(a) = \begin{pmatrix} 0 & -\infty & -\infty \\ -\infty & 1 & -\infty \\ -\infty & -\infty & 0 \end{pmatrix} \quad \mu(b) = \begin{pmatrix} 0 & 0 & -\infty \\ -\infty & -\infty & 0 \\ -\infty & -\infty & 0 \end{pmatrix}$$

$$I = \begin{pmatrix} 0 & -\infty & -\infty \end{pmatrix} \quad F = \begin{pmatrix} -\infty \\ -\infty \\ 0 \end{pmatrix}$$

$\mu(w)_{i,j}$ = max of the weights of the runs from i to j labelled by w

$$[\![\mathcal{A}]\!](w) = I\mu(w)F$$

Dimension = Number of states

And now what?

- Ultimate (very far away) goal:
Characterize all the identities holding for the class of max-plus automata with at most n states, for all n...

- Is there a strict subset of max-plus automata containing all their computational power?