

**As an introduction to
the study of weighted automata:
on rational languages
with equal generating function**

Jacques Sakarovitch

LTCI – CNRS/ENST

The result on rational languages with equal generating function, as well as those on which it relies, are taken from a joint work with

Marie-Pierre Béal and Sylvain Lombardy,
IGM, Université Paris-Est, Marne-la-Vallée,

published in

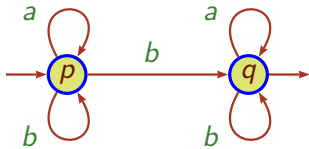
On the equivalence and conjugacy of weighted automata.

in *Proc. of CSR'06*, LNCS 3967.

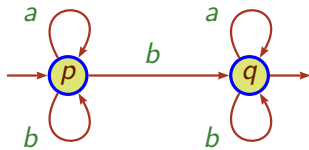
Part I

Finite automata and rational languages

A first example

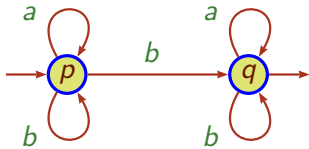


A first example



bab

A first example

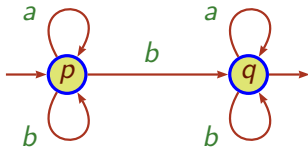


bab

$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

A first example



bab

$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

$$L(\mathcal{A}) = \{w \in A^* \mid w \in A^* b A^*\} = \{w \in A^* \mid |w|_b \geq 1\}$$

Rational (or regular) languages

Languages accepted (or recognized) by finite automata

=

Languages described by rational (or regular) expressions

=

Languages defined by MSO formula

Remarkable features of the finite automaton model

Decidable equivalence (decidable inclusion)

Closure under complement

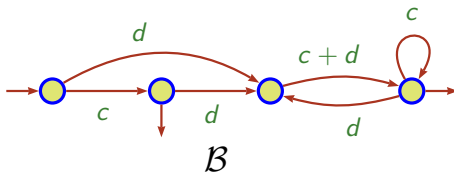
Canonical automaton (minimal deterministic automaton)

Another example

$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$

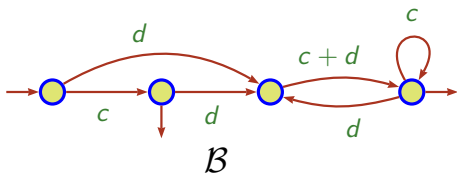
Another example

$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$



Another example

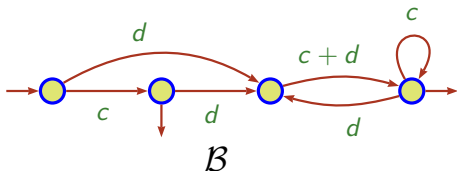
$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$



c	cdc	$cdcc$	$dcdd$
	cdd	$cddc$	$ddcc$
dc	dcc	$dccc$	$dddc$
dd	ddc	$dcdc$	$dddd$

Another example

$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$



<i>c</i>	<i>cdc</i>	<i>cdcc</i>	<i>dcdd</i>
	<i>cdd</i>	<i>cddc</i>	<i>ddcc</i>
<i>dc</i>	<i>dcc</i>	<i>dccc</i>	<i>dddc</i>
<i>dd</i>	<i>ddc</i>	<i>dcdc</i>	<i>dddd</i>

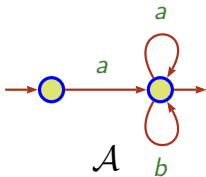
$$\forall n \in \mathbb{N} \quad g_K(n) = \text{Card}(K \cap \{c, d\}^n) = 2^{n-1}$$

Yet another example

$$L = a(a + b)^*$$

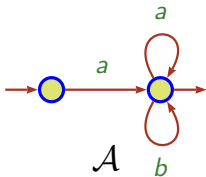
Yet another example

$$L = a(a + b)^*$$



Yet another example

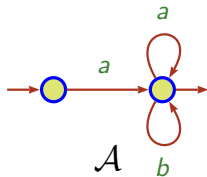
$$L = a(a + b)^*$$



<i>a</i>	<i>aaa</i>	<i>aaaa</i>	<i>abaa</i>
	<i>aab</i>	<i>aaab</i>	<i>abab</i>
<i>aa</i>	<i>aba</i>	<i>aaba</i>	<i>abba</i>
<i>ab</i>	<i>abb</i>	<i>aabb</i>	<i>abbb</i>

Yet another example

$$L = a(a + b)^*$$



<i>a</i>	<i>aaa</i>	<i>aaaa</i>	<i>abaa</i>
	<i>aab</i>	<i>aaab</i>	<i>abab</i>
<i>aa</i>	<i>aba</i>	<i>aaba</i>	<i>abba</i>
<i>ab</i>	<i>abb</i>	<i>aabb</i>	<i>abbb</i>

$$\forall n \in \mathbb{N} \quad g_L(n) = \text{Card}(L \cap \{a, b\}^n) = 2^{n-1}$$

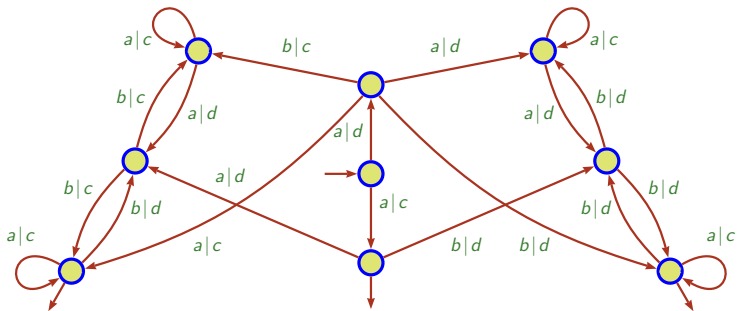
A result

Proposition

*If two regular languages have **the same growth function**,
then there exists a **letter-to-letter rational bijection**
that maps one language onto the other.*

$$L = a(a+b)^*$$

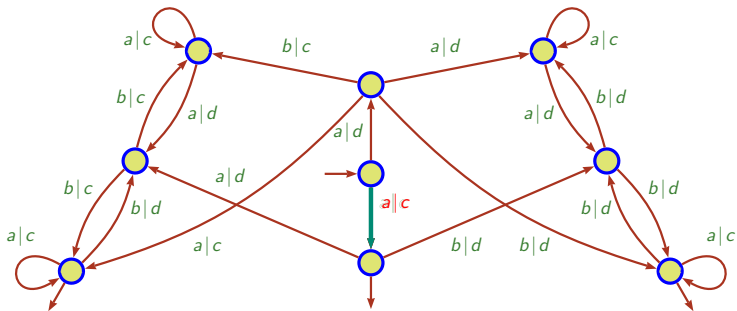
$$K = (c+dc+dd)^* \setminus \{cc(c+d)^* \cup 1_{B^*}\}$$



<i>a</i>	<i>aaa</i>	<i>aaaa</i>	<i>abaa</i>	<i>c</i>	<i>cdc</i>	<i>cdcc</i>	<i>dcdd</i>
	<i>aab</i>	<i>aaab</i>	<i>abab</i>		<i>cdd</i>	<i>cddc</i>	<i>ddcc</i>
<i>aa</i>	<i>aba</i>	<i>aaba</i>	<i>abba</i>	<i>dc</i>	<i>dcc</i>	<i>dccc</i>	<i>dddc</i>
<i>ab</i>	<i>abb</i>	<i>aabb</i>	<i>abbb</i>	<i>dd</i>	<i>ddc</i>	<i>dcdc</i>	<i>dddd</i>

$$L = a(a+b)^*$$

$$K = (c+dc+dd)^* \setminus \{cc(c+d)^* \cup 1_{B^*}\}$$

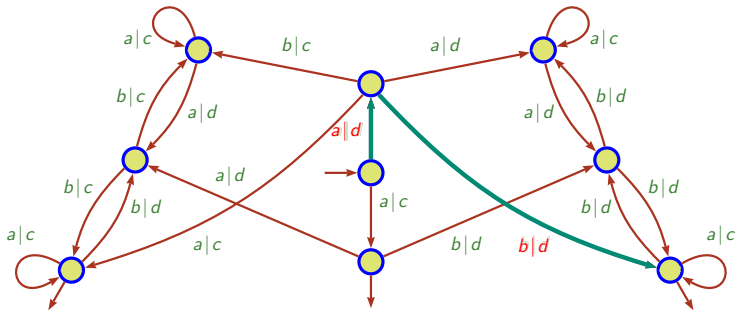


a *aaa* *aaaa* *abaa*
 aab *aaab* *abab*
aa *aba* *aaba* *abba*
ab *abb* *aabb* *abbb*

c *cdc* *cdcc* *dcdd*
 cdd *cddc* *ddcc*
dc *dcc* *dccc* *dddc*
dd *ddc* *dcdc* *dddd*

$$L = a(a+b)^*$$

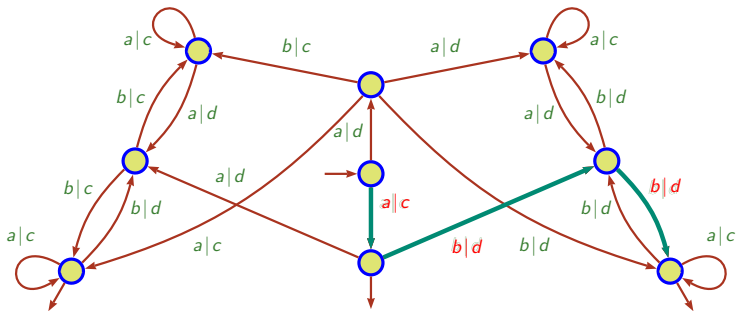
$$K = (c+dc+dd)^* \setminus \{cc(c+d)^* \cup 1_{B^*}\}$$



<i>a</i>	<i>aaa</i>	<i>aaaa</i>	<i>abaa</i>	<i>c</i>	<i>cdc</i>	<i>cdcc</i>	<i>dcdd</i>
	<i>aab</i>	<i>aaab</i>	<i>abab</i>		<i>cdd</i>	<i>cddc</i>	<i>ddcc</i>
<i>aa</i>	<i>aba</i>	<i>aaba</i>	<i>abba</i>	<i>dc</i>	<i>dcc</i>	<i>dccc</i>	<i>dddc</i>
<i>ab</i>	<i>abb</i>	<i>aabb</i>	<i>abbb</i>	<i>dd</i>	<i>ddc</i>	<i>dcdc</i>	<i>dddd</i>

$$L = a(a+b)^*$$

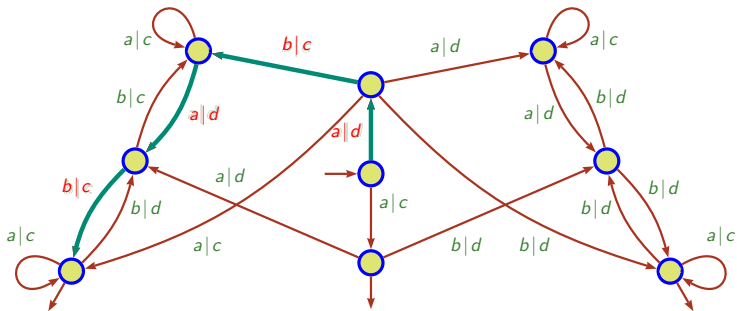
$$K = (c+dc+dd)^* \setminus \{cc(c+d)^* \cup 1_{B^*}\}$$



<i>a</i>	<i>aaa</i>	<i>aaaa</i>	<i>abaa</i>	<i>c</i>	<i>cdc</i>	<i>cdcc</i>	<i>dcdd</i>
	<i>aab</i>	<i>aaab</i>	<i>abab</i>		<i>cdd</i>	<i>cddc</i>	<i>ddcc</i>
<i>aa</i>	<i>aba</i>	<i>aaba</i>	<i>abba</i>	<i>dc</i>	<i>dcc</i>	<i>dccc</i>	<i>dddc</i>
<i>ab</i>	<i>abb</i>	<i>aabb</i>	<i>abbb</i>	<i>dd</i>	<i>ddc</i>	<i>dcdc</i>	<i>dddd</i>

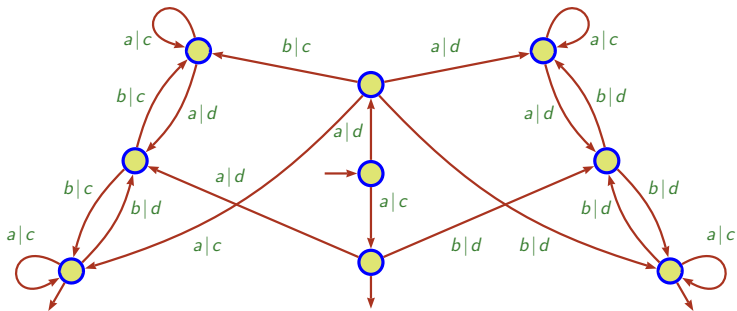
$$L = a(a+b)^*$$

$$K = (c+dc+dd)^* \setminus \{cc(c+d)^* \cup 1_{B^*}\}$$

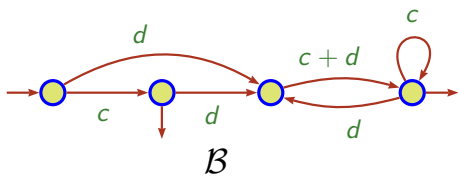
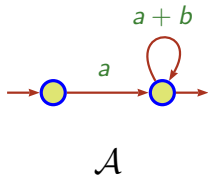


<i>a</i>	<i>aaa</i>	<i>aaaa</i>	<i>abaa</i>	<i>c</i>	<i>cdc</i>	<i>cdcc</i>	<i>dcdd</i>
	<i>aab</i>	<i>aaab</i>	<i>abab</i>		<i>cdd</i>	<i>cddc</i>	<i>ddcc</i>
<i>aa</i>	<i>aba</i>	<i>aaba</i>	<i>abba</i>	<i>dc</i>	<i>dcc</i>	<i>dccc</i>	<i>dddc</i>
<i>ab</i>	<i>abb</i>	<i>aabb</i>	<i>abbb</i>	<i>dd</i>	<i>ddc</i>	<i>dcdc</i>	<i>dddd</i>

How to construct the transducer



from the automata



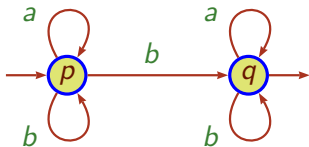
First task

Building the bridge between **growth function** and **finite automata**.

Part II

Weighted automata

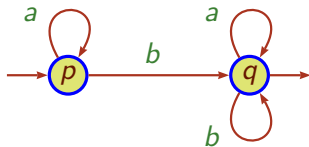
Back to the first example



$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

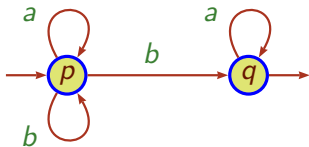
$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

Back to the first example



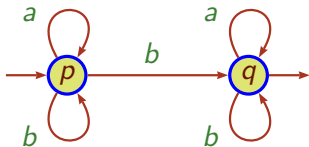
$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

Back to the first example



$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

Back to the first example

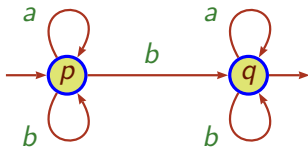


$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

$$bab \mapsto 2$$

Back to the first example

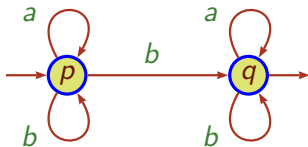


$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

$$bab \mapsto 2 \quad \forall w \in A^* \quad w \mapsto |w|_b$$

Back to the first example



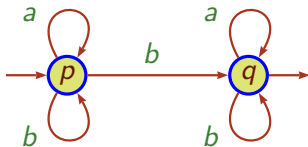
$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

$$bab \mapsto 2 \quad \forall w \in A^* \quad w \mapsto |w|_b$$

$$s: A^* \longrightarrow \mathbb{N} \quad s: w \mapsto \langle s, w \rangle \quad s \in \mathbb{N}^{A^*}$$

Back to the first example



$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

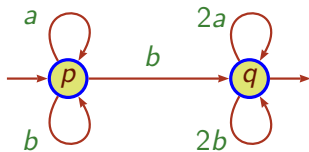
$$p \xrightarrow{b} q \xrightarrow{a} q \xrightarrow{b} q$$

$$bab \mapsto 2 \quad \forall w \in A^* \quad w \mapsto |w|_b$$

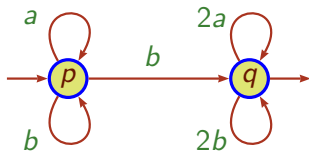
$$s: A^* \longrightarrow \mathbb{N} \quad s: w \mapsto \langle s, w \rangle \quad s \in \mathbb{N}^{A^*}$$

$$s = b + ab + ba + 2bb + aab \\ + aba + 2abb + baa + 2bab + \dots$$

Variation



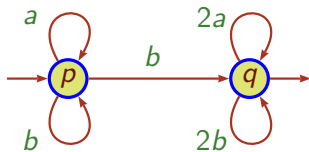
Variation



$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{2a} q \xrightarrow{2b} q$$

Variation

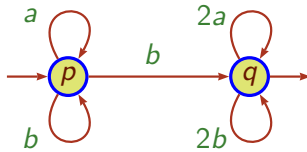


$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{2a} q \xrightarrow{2b} q$$

$$bab \mapsto 5$$

Variation



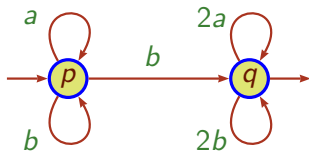
$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{2a} q \xrightarrow{2b} q$$

$$bab \mapsto 5$$

$$\forall w \in A^* \quad w \mapsto \langle w \rangle_2$$

Variation



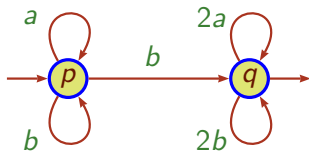
$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

$$p \xrightarrow{b} q \xrightarrow{2a} q \xrightarrow{2b} q$$

$$bab \mapsto 5 \quad \forall w \in A^* \quad w \mapsto \langle w \rangle_2$$

$$s: A^* \longrightarrow \mathbb{N} \quad s: w \mapsto \langle s, w \rangle \quad s \in \mathbb{N}^{A^*}$$

Variation



$$p \xrightarrow{b} p \xrightarrow{a} p \xrightarrow{b} q$$

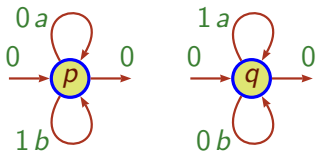
$$p \xrightarrow{b} q \xrightarrow{2a} q \xrightarrow{2b} q$$

$$bab \mapsto 5 \quad \forall w \in A^* \quad w \mapsto \langle w \rangle_2$$

$$s: A^* \longrightarrow \mathbb{N} \quad s: w \mapsto \langle s, w \rangle \quad s \in \mathbb{N}^{A^*}$$

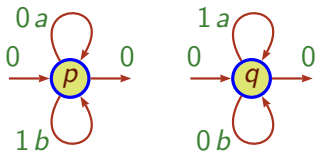
$$s = b + ab + 2ba + 3bb + aab \\ + 2aba + 3abb + 4baa + 5bab + \dots$$

Another example



Weights in $\mathbb{K} = \mathcal{M} = \langle \mathbb{N} \cup \{+\infty\}, \min, + \rangle$

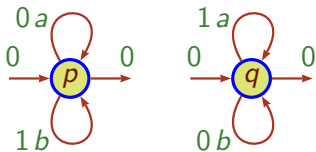
Another example



Weights in $\mathbb{K} = \mathcal{M} = \langle \mathbb{N} \cup \{+\infty\}, \min, + \rangle$

$$\begin{array}{l} p \xrightarrow{1b} p \xrightarrow{0a} p \xrightarrow{1b} p \\ q \xrightarrow{0b} q \xrightarrow{1a} q \xrightarrow{0b} q \end{array}$$

Another example

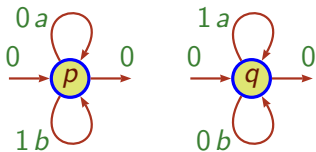


Weights in $\mathbb{K} = \mathcal{M} = \langle \mathbb{N} \cup \{+\infty\}, \min, + \rangle$

$$\begin{aligned} p &\xrightarrow{1b} p \xrightarrow{0a} p \xrightarrow{1b} p \\ q &\xrightarrow{0b} q \xrightarrow{1a} q \xrightarrow{0b} q \end{aligned}$$

$$bab \mapsto 1$$

Another example

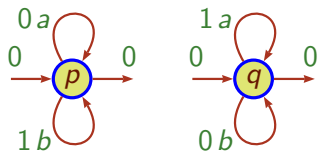


Weights in $\mathbb{K} = \mathcal{M} = \langle \mathbb{N} \cup \{+\infty\}, \min, + \rangle$

$$\begin{array}{l}
 p \xrightarrow{1b} p \xrightarrow{0a} p \xrightarrow{1b} p \\
 q \xrightarrow{0b} q \xrightarrow{1a} q \xrightarrow{0b} q
 \end{array}$$

$$bab \mapsto 1 \quad \forall w \in A^* \quad w \mapsto \min\{|w|_a, |w|_b\}$$

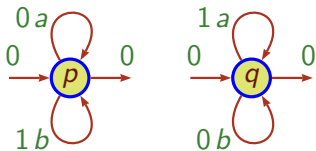
Another example



Weights in $\mathbb{K} = \mathcal{M} = \langle \mathbb{N} \cup \{+\infty\}, \min, + \rangle$

$$s: A^* \longrightarrow \mathcal{M} \quad s: w \longmapsto \langle s, w \rangle \quad s \in \mathcal{M}^{A^*}$$

Another example



Weights in $\mathbb{K} = \mathcal{M} = \langle \mathbb{N} \cup \{+\infty\}, \min, + \rangle$

$$s: A^* \longrightarrow \mathcal{M} \quad s: w \longmapsto \langle s, w \rangle \quad s \in \mathcal{M}^{A^*}$$

$$s = 01_{A^*} \oplus 0a \oplus 0b \oplus 0aa \oplus 1ab \oplus 1ba \oplus 0bb \\ \oplus 0aaa \oplus 1aab \oplus 1aba \oplus 1abb \oplus \dots$$

Series play the role of languages

$\mathbb{K}\langle\langle A^* \rangle\rangle$ plays the role of $\mathfrak{P}(A^*)$

Richness of the model of weighted automata

- ▶ \mathbb{B} 'classic' automata
- ▶ \mathbb{N} 'usual' counting
- ▶ $\mathbb{Z}, \mathbb{Q}, \mathbb{R}$ numerical multiplicity
- ▶ $\mathcal{M} = \langle \mathbb{N}, \min, + \rangle$ Min-plus automata
- ▶ $\mathfrak{P}(B^*) = \mathbb{B}\langle\langle B^* \rangle\rangle$ transducers
- ▶ $\mathbb{N}\langle\langle B^* \rangle\rangle$ weighted transducers
- ▶ $\mathfrak{P}(F(B))$ pushdown automata

Equivalence of weighted automata

The equivalence of weighted automata with weights in

Equivalence of weighted automata

The equivalence of weighted automata with weights in the Boolean semiring \mathbb{B} is decidable

Equivalence of weighted automata

The equivalence of weighted automata with weights in

the Boolean semiring \mathbb{B} is decidable
a field is decidable

Equivalence of weighted automata

The equivalence of weighted automata with weights in

the Boolean semiring \mathbb{B} is decidable
a subsemiring of a field is decidable

Equivalence of weighted automata

The equivalence of weighted automata with weights in

the Boolean semiring \mathbb{B}	decidable
a subsemiring of a field	decidable
$(\mathbb{Z}, \min, +)$	is undecidable

Equivalence of weighted automata

The equivalence of weighted automata with weights in

the Boolean semiring \mathbb{B}	decidable
a subsemiring of a field	decidable
$(\mathbb{Z}, \min, +)$	undecidable
$\text{Rat } B^*$	is undecidable

The equivalence of

transducers is undecidable

Equivalence of weighted automata

The equivalence of weighted automata with weights in

the Boolean semiring \mathbb{B}	decidable
a subsemiring of a field	decidable
$(\mathbb{Z}, \min, +)$	undecidable
$\text{Rat } B^*$	undecidable
$\text{NRat } B^*$ is	decidable

The equivalence of

transducers	undecidable
transducers with multiplicity in \mathbb{N} is	decidable

Equivalence of weighted automata

The equivalence of weighted automata with weights in

the Boolean semiring \mathbb{B}	decidable
a subsemiring of a field	decidable
$(\mathbb{Z}, \min, +)$	undecidable
$\text{Rat } B^*$	undecidable
$\text{NRat } B^*$	decidable

The equivalence of

transducers	undecidable
transducers with multiplicity in \mathbb{N}	decidable
functional transducers	is decidable

Equivalence of weighted automata

The equivalence of weighted automata with weights in

the Boolean semiring \mathbb{B}	decidable
a subsemiring of a field	decidable
$(\mathbb{Z}, \min, +)$	undecidable
$\text{Rat } B^*$	undecidable
$\text{NRat } B^*$	decidable

The equivalence of

transducers	undecidable
transducers with multiplicity in \mathbb{N}	decidable
functional transducers	decidable
$(\mathbb{Z}, \min, +)$ -unambiguous automata	is decidable

Back to the growth function

a language

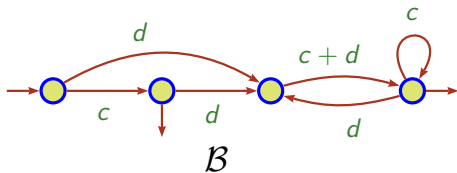
$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$

Back to the growth function

a language

$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$

an *unambiguous* automaton

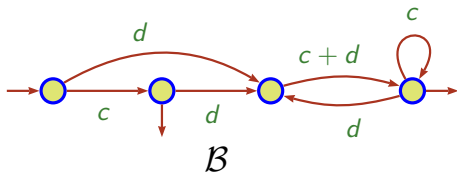


Back to the growth function

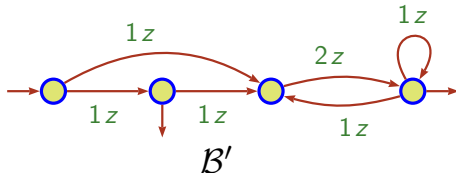
a language

$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$

an *unambiguous* automaton



is transformed into an automaton over $\{z\}^*$ with weight in \mathbb{N}

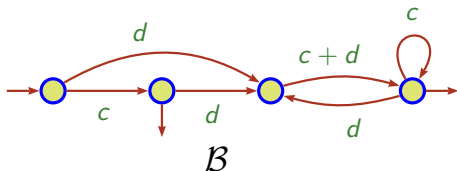


Back to the growth function

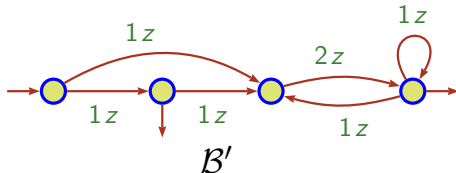
a language

$$K = (c + dc + dd)^* \setminus \{cc(c + d)^* \cup 1_{B^*}\}$$

an *unambiguous* automaton



is transformed into an automaton over $\{z\}^*$ with weight in \mathbb{N}



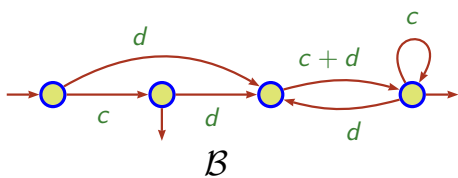
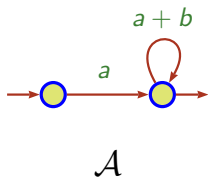
which realises the generating function $G_K(z) = \sum_{n \in \mathbb{N}} g_K(n) z^n$

Part III

The path to the proof of the result

What is behind the sentence:

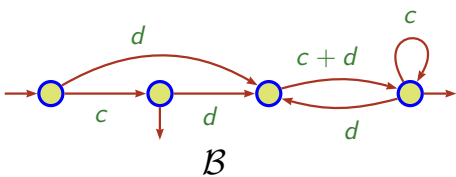
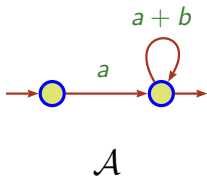
'Two regular languages with equal growth functions'?



What is behind the sentence:

'Two regular languages with equal growth functions'?

- (i) Two finite automata \mathcal{A} and \mathcal{B} , preferably **unambiguous**

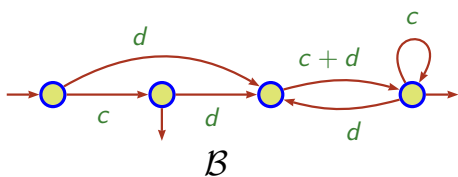
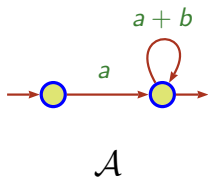


What is behind the sentence:

'Two regular languages with equal growth functions'?

- (i) Two finite automata \mathcal{A} and \mathcal{B} , preferably **unambiguous**
- (ii) transformed into \mathcal{A}' and \mathcal{B}' , over $\{z\}^*$ with multiplicity in \mathbb{N} , which realise the *generating functions* $G_L(z)$ and $G_K(z)$:

$$G_L(z) = \sum_{n \in \mathbb{N}} g_L(n) z^n \quad \text{and} \quad G_K(z) = \sum_{n \in \mathbb{N}} g_K(n) z^n .$$

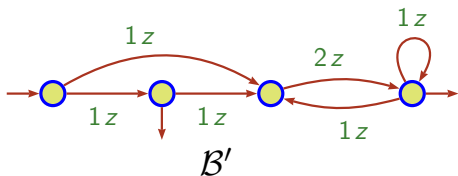
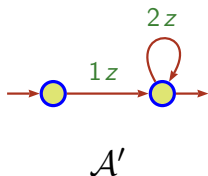


What is behind the sentence:

'Two regular languages with equal growth functions'?

- (i) Two finite automata \mathcal{A} and \mathcal{B} , preferably **unambiguous**
- (ii) transformed into \mathcal{A}' and \mathcal{B}' , over $\{z\}^*$ with multiplicity in \mathbb{N} , which realise the *generating functions* $G_L(z)$ and $G_K(z)$:

$$G_L(z) = \sum_{n \in \mathbb{N}} g_L(n) z^n \quad \text{and} \quad G_K(z) = \sum_{n \in \mathbb{N}} g_K(n) z^n .$$



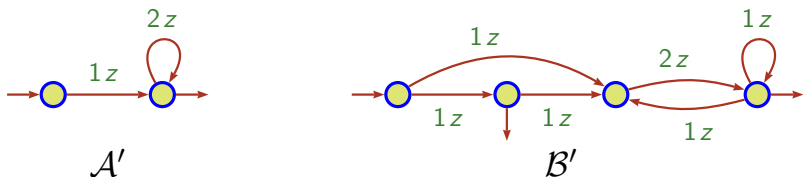
What is behind the sentence:

'Two regular languages with equal growth functions'?

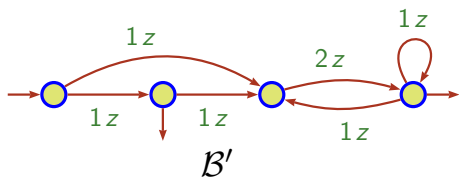
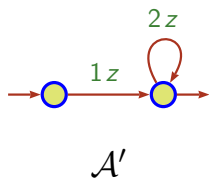
- (i) Two finite automata \mathcal{A} and \mathcal{B} , preferably **unambiguous**
- (ii) transformed into \mathcal{A}' and \mathcal{B}' , over $\{z\}^*$ with multiplicity in \mathbb{N} , which realise the *generating functions* $G_L(z)$ and $G_K(z)$:

$$G_L(z) = \sum_{n \in \mathbb{N}} g_L(n) z^n \quad \text{and} \quad G_K(z) = \sum_{n \in \mathbb{N}} g_K(n) z^n .$$

- (iii) and whose equivalence is decidable (Schütz. 1961, Eil. 1974).



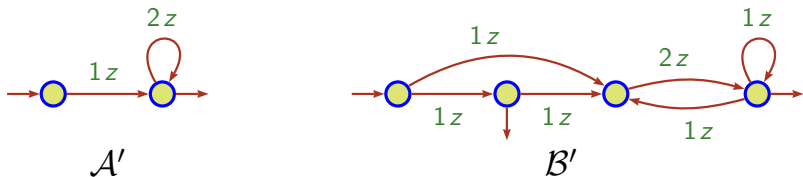
The main theorem



The main theorem

Theorem (BLS)

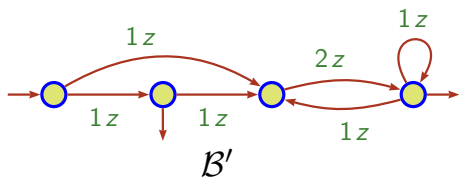
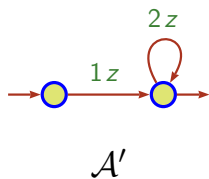
Two \mathbb{N} -automata are equivalent if, and only if they are conjugate to a same third \mathbb{N} -automaton.



The main theorem

A confession

Automata are matrices

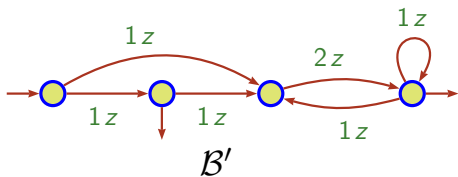
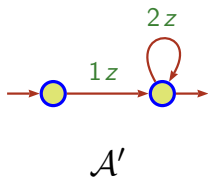


The main theorem

A confession

Automata are matrices

$$\mathcal{A}' = \langle I, E, T \rangle = \left\langle (1 \ 0), \begin{pmatrix} 0 & z \\ 0 & 2z \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$



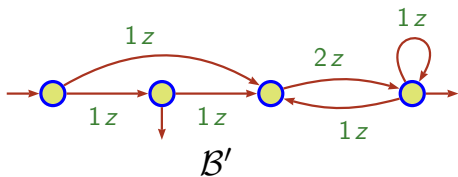
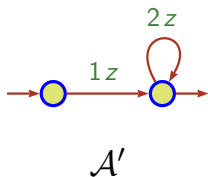
The main theorem

A confession

Automata are matrices

$$\mathcal{A}' = \langle I, E, T \rangle = \left\langle (1 \ 0), \begin{pmatrix} 0 & z \\ 0 & 2z \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$

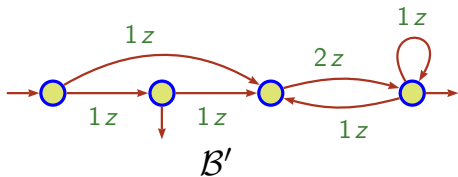
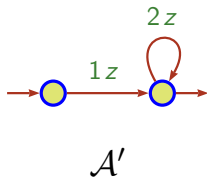
$$|\mathcal{A}'| = I E^* T$$



The main theorem

Theorem (BLS)

Two \mathbb{N} -automata are equivalent if, and only if they are conjugate to a same third \mathbb{N} -automaton.



The main theorem

Theorem (BLS)

Two \mathbb{N} -automata are equivalent if, and only if they are conjugate to a same third \mathbb{N} -automaton.

Definition

Let $\mathcal{A} = \langle I, E, T \rangle$ and $\mathcal{B} = \langle J, F, U \rangle$ be two \mathbb{K} -automata.

\mathcal{A} is conjugate to \mathcal{B}

if there exists a \mathbb{K} -matrix X such that :

$$IX = J, \quad EX = XF, \quad \text{and} \quad T = XU .$$

This is denoted as $\mathcal{A} \xrightarrow{X} \mathcal{B}$.

The main theorem

Theorem (BLS)

Two \mathbb{N} -automata are equivalent if, and only if they are conjugate to a same third \mathbb{N} -automaton.

Definition

Let $\mathcal{A} = \langle I, E, T \rangle$ and $\mathcal{B} = \langle J, F, U \rangle$ be two \mathbb{K} -automata.

\mathcal{A} is conjugate to \mathcal{B}

if there exists a \mathbb{K} -matrix X such that :

$$IX = J, \quad EX = XF, \quad \text{and} \quad T = XU .$$

This is denoted as $\mathcal{A} \xrightarrow{X} \mathcal{B}$.

Conjugacy is a *preorder*

(transitive and reflexive, but not symmetric).

The main theorem

Theorem (BLS)

Two \mathbb{N} -automata are equivalent if, and only if they are conjugate to a same third \mathbb{N} -automaton.

Definition

Let $\mathcal{A} = \langle I, E, T \rangle$ and $\mathcal{B} = \langle J, F, U \rangle$ be two \mathbb{K} -automata.

\mathcal{A} is conjugate to \mathcal{B}

if there exists a \mathbb{K} -matrix X such that :

$$IX = J, \quad EX = XF, \quad \text{and} \quad T = XU .$$

This is denoted as $\mathcal{A} \xrightarrow{X} \mathcal{B}$.

$\mathcal{A} \xrightarrow{X} \mathcal{B}$ implies that \mathcal{A} and \mathcal{B} are *equivalent*.

$$I E E T = I E E X U = I E X F U = I X F F U = J F F U$$

$$\text{and then} \quad I E^* T = J F^* U$$

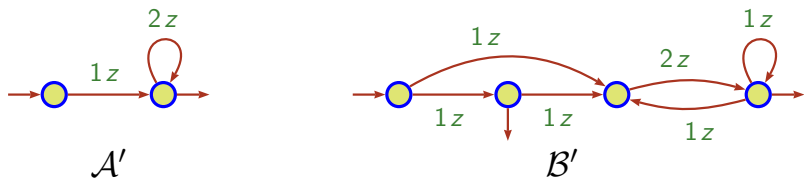
The main theorem

Theorem (BLS)

Two \mathbb{N} -automata \mathcal{A} and \mathcal{B} are equivalent if, and only if, there exist an \mathbb{N} -automaton \mathcal{C} (and \mathbb{N} -matrices X and Y) such that

$$\mathcal{A} \xleftarrow{X} \mathcal{C} \xrightarrow{Y} \mathcal{B}$$

Moreover, \mathcal{C} is effectively computable from \mathcal{A} and \mathcal{B} .



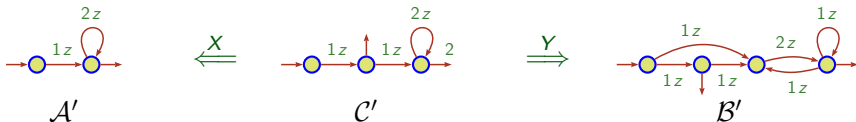
The main theorem

Theorem (BLS)

Two \mathbb{N} -automata \mathcal{A} and \mathcal{B} are equivalent if, and only if, there exist an \mathbb{N} -automaton \mathcal{C} (and \mathbb{N} -matrices X and Y) such that

$$\mathcal{A} \xleftarrow{X} \mathcal{C} \xrightarrow{Y} \mathcal{B}$$

Moreover, \mathcal{C} is effectively computable from \mathcal{A} and \mathcal{B} .



The main theorem

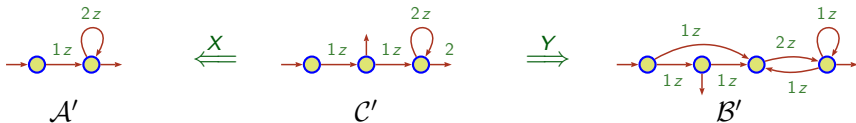
Theorem (BLS)

Two \mathbb{N} -automata \mathcal{A} and \mathcal{B} are equivalent if, and only if, there exist an \mathbb{N} -automaton \mathcal{C} (and \mathbb{N} -matrices X and Y) such that

$$\mathcal{A} \xleftarrow{X} \mathcal{C} \xrightarrow{Y} \mathcal{B}$$

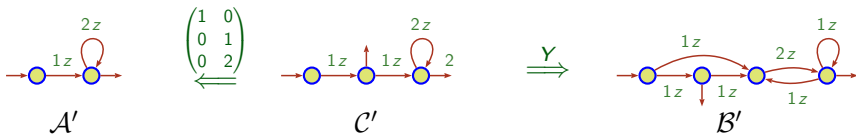
Moreover, \mathcal{C} is effectively computable from \mathcal{A} and \mathcal{B} .

with $X = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{pmatrix}$ and $Y = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 2 \end{pmatrix}$



The main theorem

$$\mathcal{C}' = \left\langle (1 \ 0 \ 0), \begin{pmatrix} 0 & z & 0 \\ 0 & 0 & z \\ 0 & 0 & 2z \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} \right\rangle \quad \mathcal{A}' = \left\langle (1 \ 0), \begin{pmatrix} 0 & z \\ 0 & 2z \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$



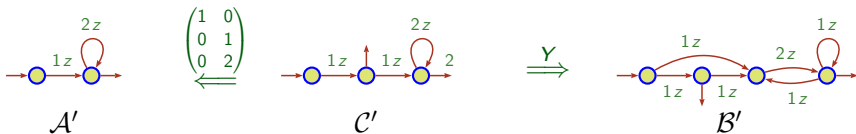
The main theorem

$$\mathcal{C}' = \left\langle (1 \ 0 \ 0), \begin{pmatrix} 0 & z & 0 \\ 0 & 0 & z \\ 0 & 0 & 2z \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} \right\rangle \quad \mathcal{A}' = \left\langle (1 \ 0), \begin{pmatrix} 0 & z \\ 0 & 2z \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \end{pmatrix} \right\rangle$$

$$(1 \ 0 \ 0) \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{pmatrix} = (1 \ 0),$$

$$\begin{pmatrix} 0 & z & 0 \\ 0 & 0 & z \\ 0 & 0 & 2z \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{pmatrix} \cdot \begin{pmatrix} 0 & z \\ 0 & 2z \end{pmatrix},$$

$$\begin{pmatrix} 0 \\ 1 \\ 2 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 2 \end{pmatrix} \cdot \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$



A' \xleftarrow{X}

C'

\xrightarrow{Y}

B'

The second theorem



The second theorem

A structural interpretation of conjugacy

Theorem (BLS)

Let \mathcal{A} and \mathcal{B} be two conjugate \mathbb{N} -automata.

Then, there exist an \mathbb{N} -automaton \mathcal{D}

such that \mathcal{A} is a *co-quotient* of \mathcal{D}

and \mathcal{B} is an *quotient* of \mathcal{D} .

Moreover, \mathcal{D} is effectively computable from \mathcal{A} and \mathcal{B} .



The second theorem

A structural interpretation of conjugacy

Theorem (BLS)

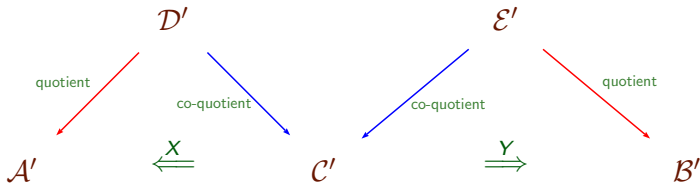
Let \mathcal{A} and \mathcal{B} be two conjugate \mathbb{N} -automata.

Then, there exist an \mathbb{N} -automaton \mathcal{D}

such that \mathcal{A} is a *co-quotient* of \mathcal{D}

and \mathcal{B} is an *quotient* of \mathcal{D} .

Moreover, \mathcal{D} is effectively computable from \mathcal{A} and \mathcal{B} .



The second theorem

A structural interpretation of conjugacy

Theorem (BLS)

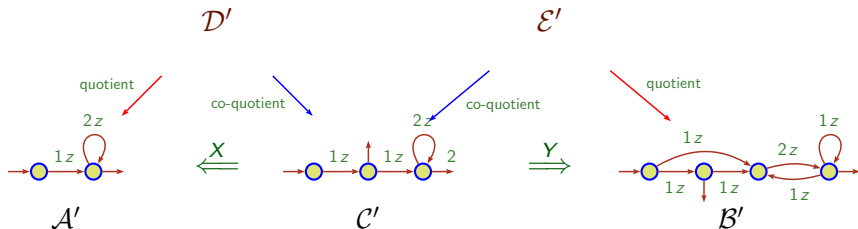
Let \mathcal{A} and \mathcal{B} be two conjugate \mathbb{N} -automata.

Then, there exist an \mathbb{N} -automaton \mathcal{D}

such that \mathcal{A} is a *co-quotient* of \mathcal{D}

and \mathcal{B} is an *quotient* of \mathcal{D} .

Moreover, \mathcal{D} is effectively computable from \mathcal{A} and \mathcal{B} .



The second theorem

A structural interpretation of conjugacy

Theorem (BLS)

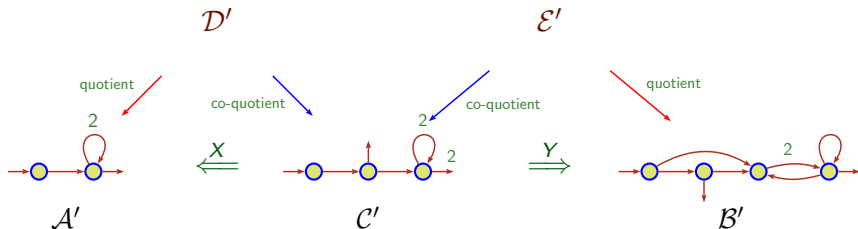
Let \mathcal{A} and \mathcal{B} be two conjugate \mathbb{N} -automata.

Then, there exist an \mathbb{N} -automaton \mathcal{D}

such that \mathcal{A} is a *co-quotient* of \mathcal{D}

and \mathcal{B} is an *quotient* of \mathcal{D} .

Moreover, \mathcal{D} is effectively computable from \mathcal{A} and \mathcal{B} .



The second theorem

A structural interpretation of conjugacy

Theorem (BLS)

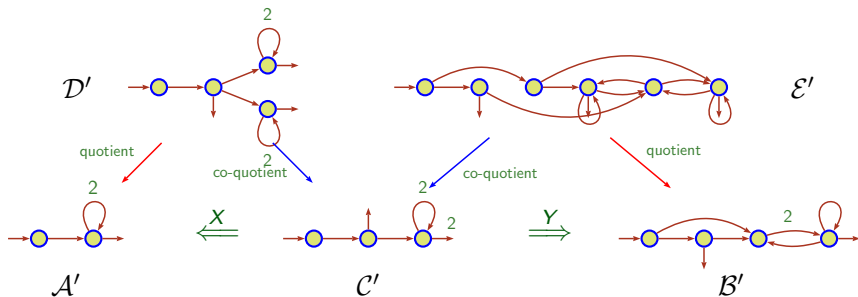
Let \mathcal{A} and \mathcal{B} be two conjugate \mathbb{N} -automata.

Then, there exist an \mathbb{N} -automaton \mathcal{D}

such that \mathcal{A} is a *co-quotient* of \mathcal{D}

and \mathcal{B} is an *quotient* of \mathcal{D} .

Moreover, \mathcal{D} is effectively computable from \mathcal{A} and \mathcal{B} .



The second theorem

A structural interpretation of conjugacy

Theorem (BLS)

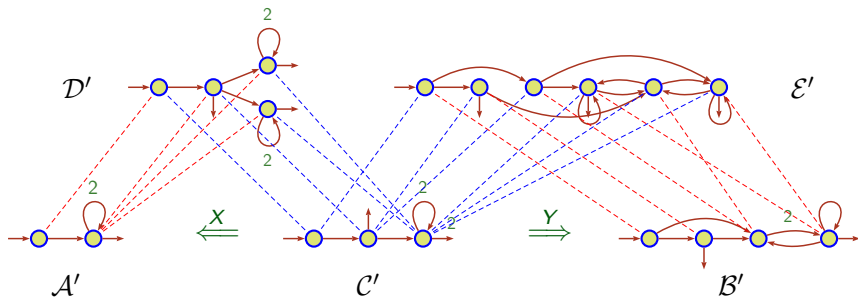
Let \mathcal{A} and \mathcal{B} be two conjugate \mathbb{N} -automata.

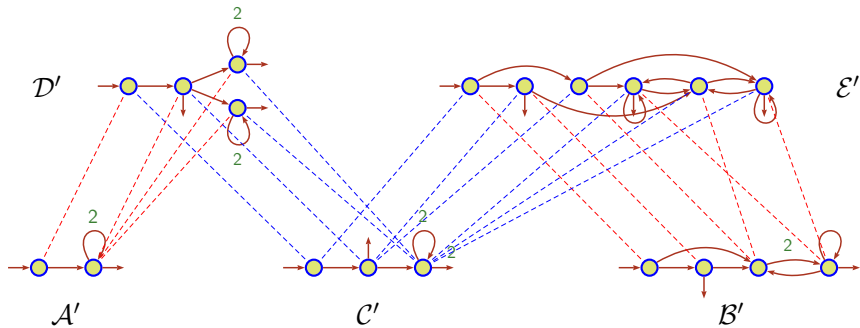
Then, there exist an \mathbb{N} -automaton \mathcal{D}

such that \mathcal{A} is a *co-quotient* of \mathcal{D}

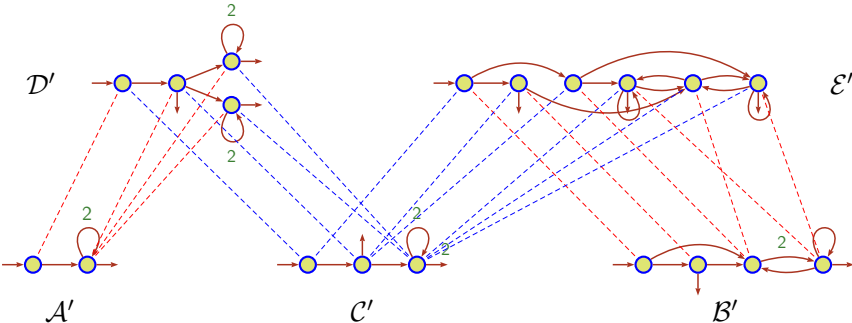
and \mathcal{B} is an *quotient* of \mathcal{D} .

Moreover, \mathcal{D} is effectively computable from \mathcal{A} and \mathcal{B} .

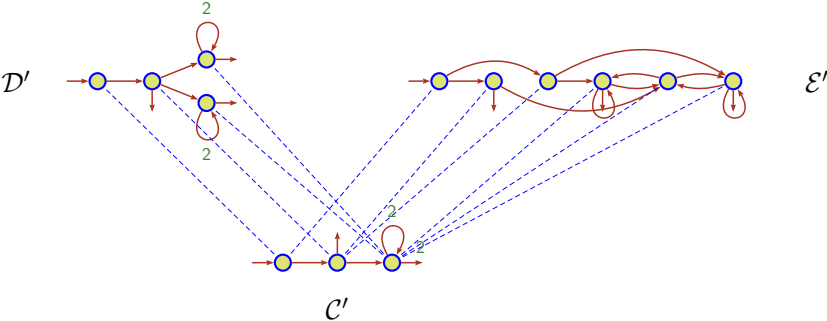




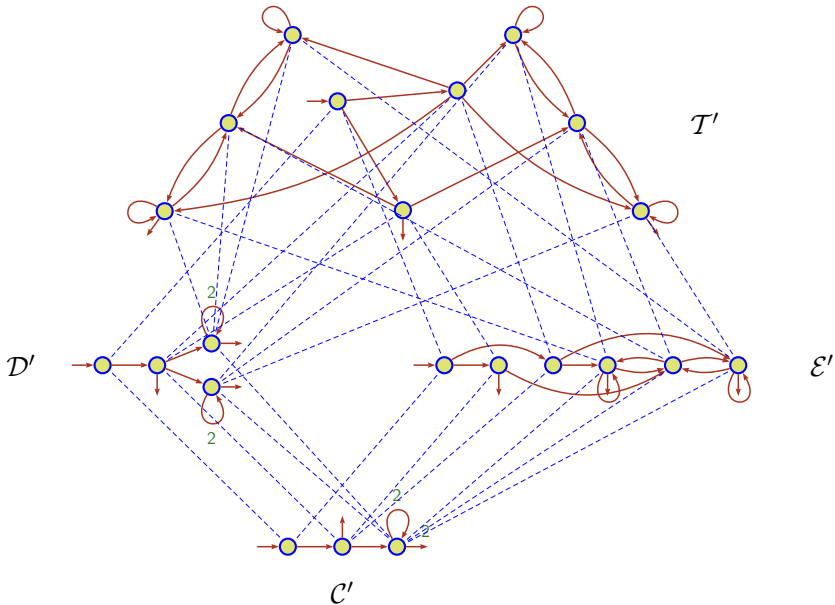
A technical proposition



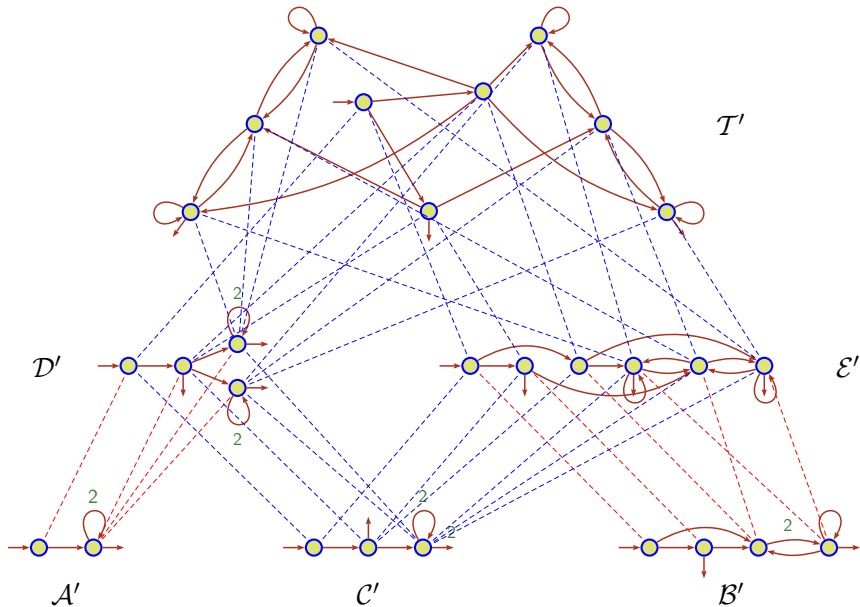
A technical proposition



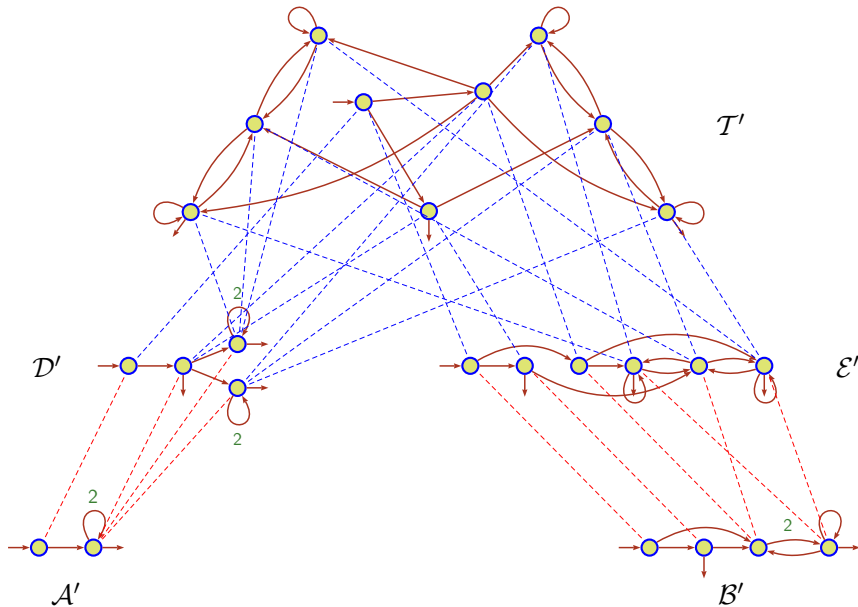
A technical proposition



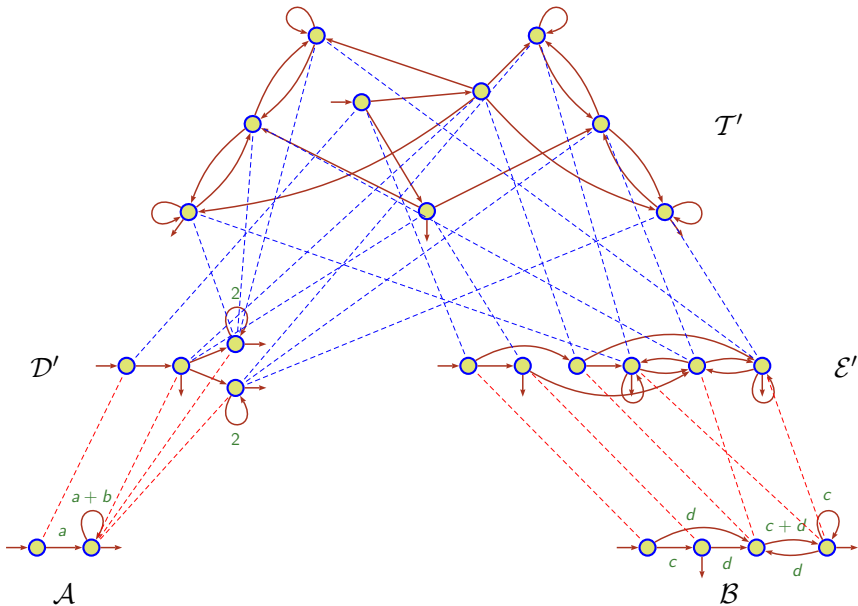
The harvest



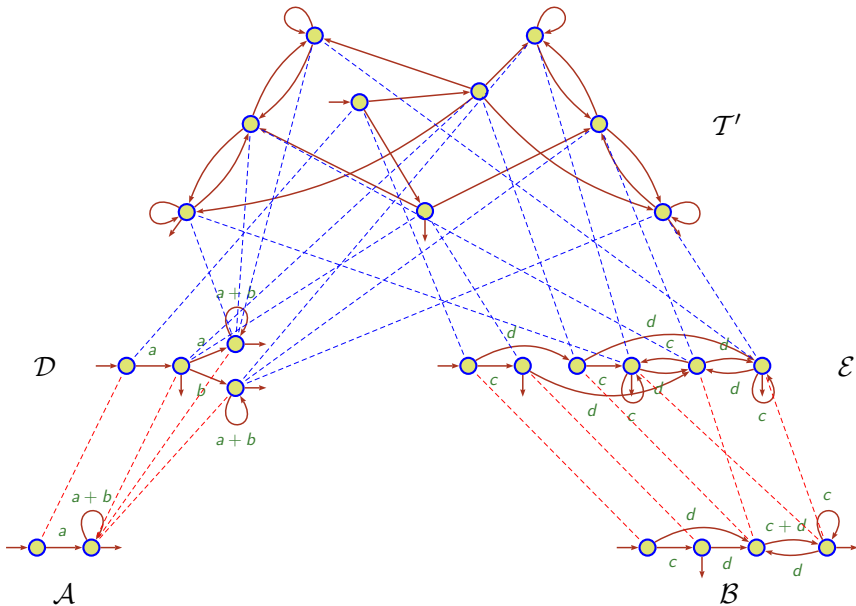
The harvest



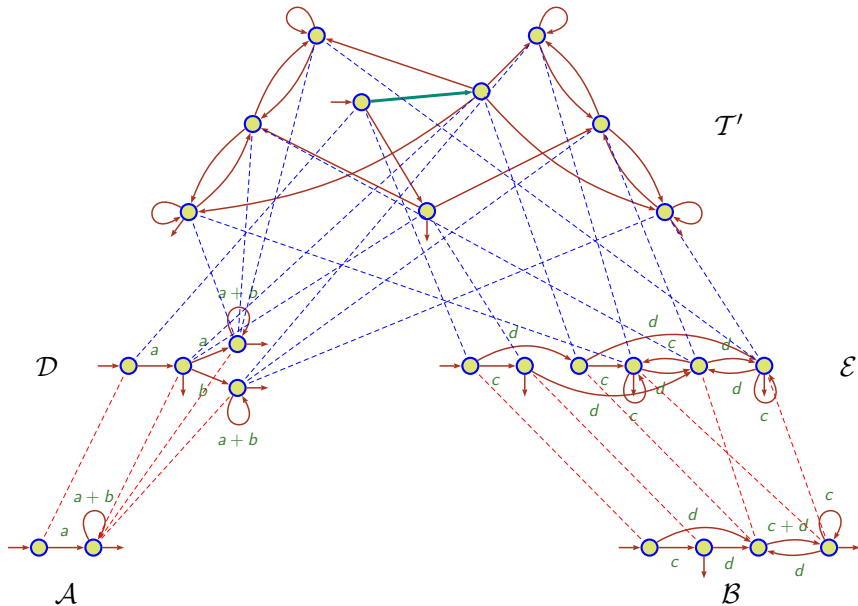
The harvest



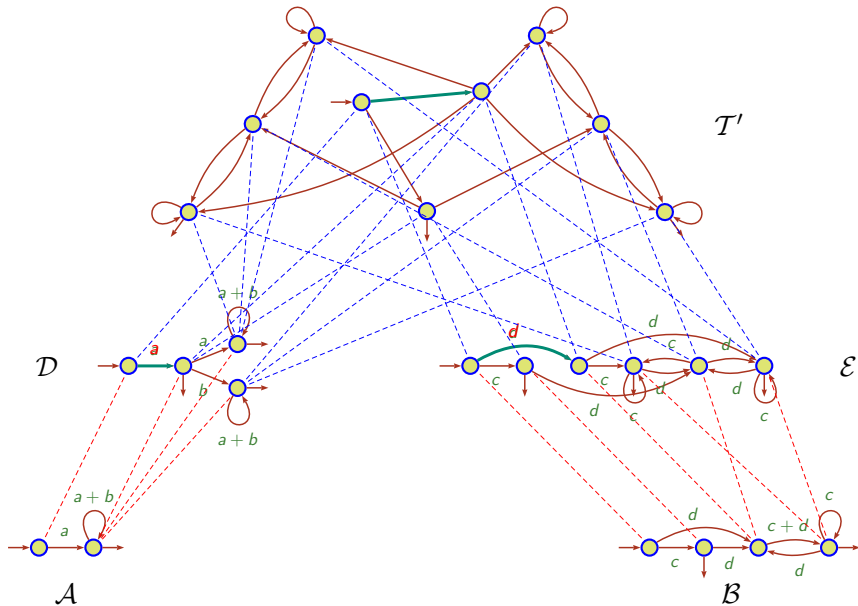
The harvest



The harvest



The harvest



The harvest

