From multiplicity awareness to computation correlation

Jacques Sakarovitch

LTCI – CNRS/ENST

The results presented in this talk 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. The complete journal version is still in preparation. Some of the results have been included in the chapter

Rational and recognizable series

of the *Handbook of Weighted Automata*, Springer, 2009.

Part I

An introductory result

The Rational Bijection Theorem

Proposition

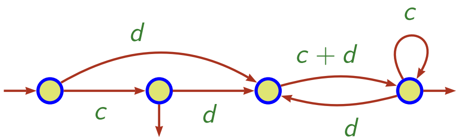
If two regular languages have the same growth function, then there exists a letter-to-letter rational bijection that maps one language onto the other.

An example: a first language

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

An example: a first language

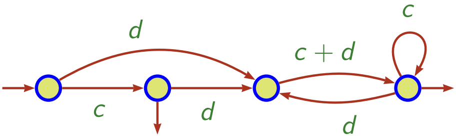
$$K = (c + d\,c + d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$$



$\mathcal{B}$

An example: a first language

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

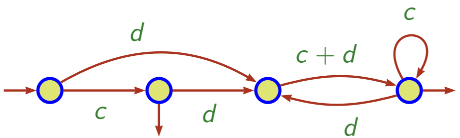


$\mathcal{B}$

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

An example: a first language

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



$\mathcal{B}$

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

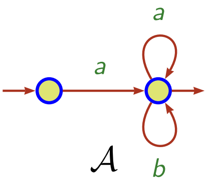
$$\forall n \in \mathbb{N} \qquad \mathrm{g}_K(n) \;=\; \mathrm{Card}\,(K \cap \{c,d\}^n) \;=\; 2^{n-1}$$

An example: a second language

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

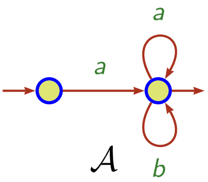
An example: a second language

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



An example: a second language

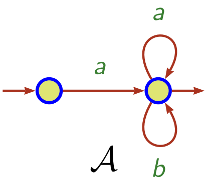
$$L = a(a+b)^*$$



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

An example: a second language

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



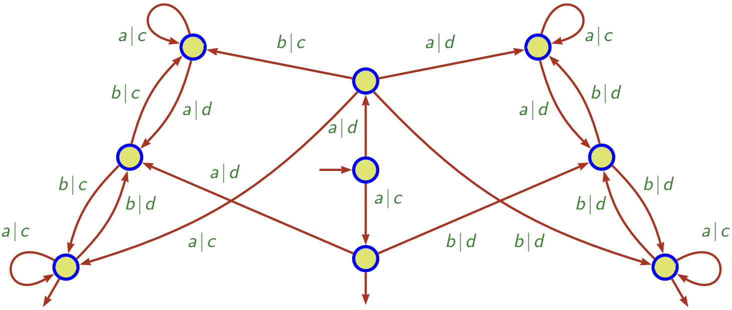
a	aaa	$aaaa$	$abaa$
	aab	$aaab$	$abab$
aa	aba	$aaba$	$abba$
ab	abb	$aabb$	$abbb$

$$\forall n \in \mathbb{N} \qquad \mathrm{g}_L(n) \;=\; \mathrm{Card}\left(L \cap \{a,b\}^n\right) \;=\; 2^{n-1}$$

An example: the rational bijection

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

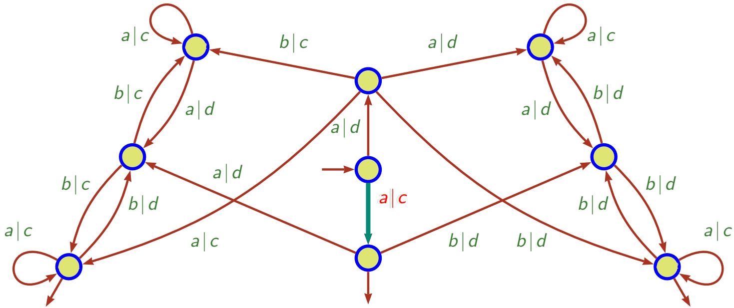
$K = (c+d\,c+d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$



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

An example: the rational bijection

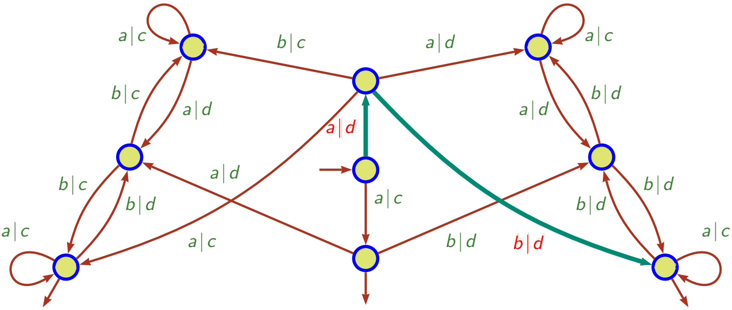
$L = a(a+b)^*$ $K = (c+d\,c+d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$



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

An example: the rational bijection

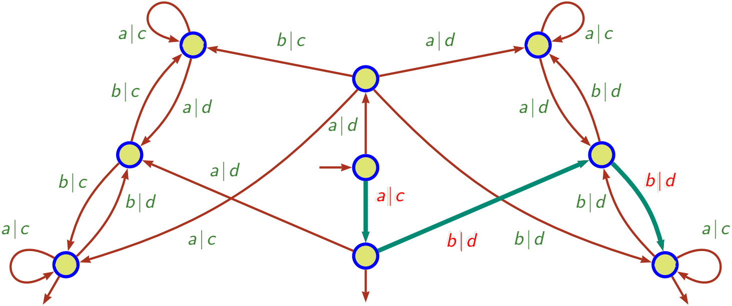
$L = a(a+b)^*$ $\qquad$ $K = (c+d\,c+d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$



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

An example: the rational bijection

$L = a(a+b)^*$ $\qquad$ $K = (c+d\,c+d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$

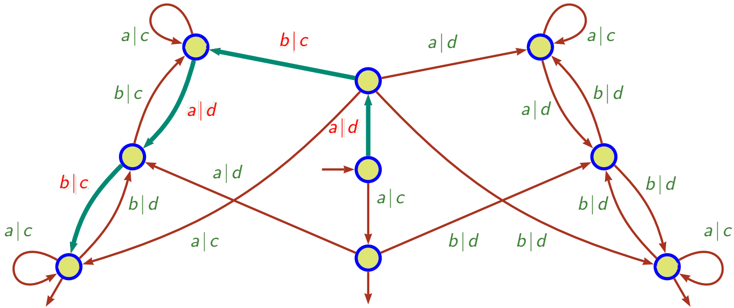


a	*aaa*	*aaaa*	*abaa*
	aab	*aaab*	*abab*
aa	*aba*	*aaba*	*abba*
ab	*abb*	*aabb*	*abbb*

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

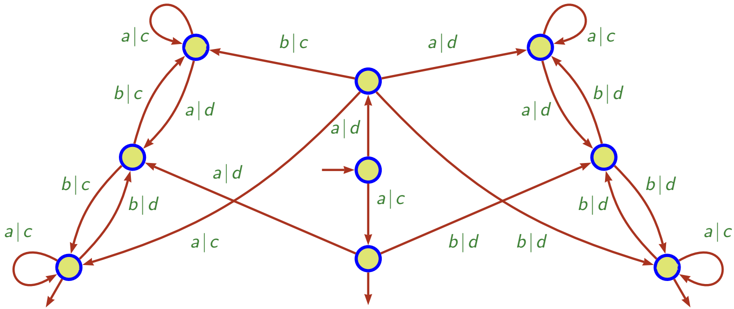
An example: the rational bijection

$L = a(a+b)^*$ $K = (c+d\,c+d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$

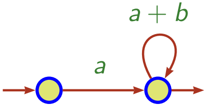


a	aaa	$aaaa$	$abaa$	c	cdc	$cdcc$	$dcdd$
	aab	$aaab$	$abab$		cdd	$cddc$	$ddcc$
aa	aba	$aaba$	$abba$	dc	dcc	$dccc$	$dddc$
ab	abb	$aabb$	$abbb$	dd	ddc	$dcdc$	$dddd$

The result on this example: how to construct the transducer

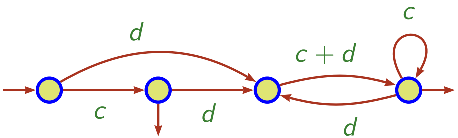


from the automata



$\mathcal{A}$

and



$\mathcal{B}$

Part II

The link between growth functions and automata

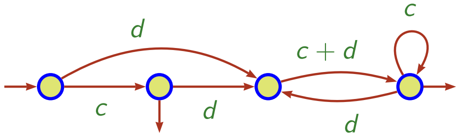
The generating function

A language $\quad K = (c + d\,c + d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\} \quad$ that is,

The generating function

A language $K = (c + d\,c + d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$ that is,

an *unambiguous* automaton

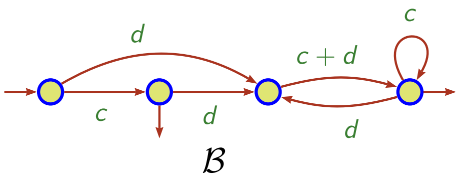


$\mathcal{B}$

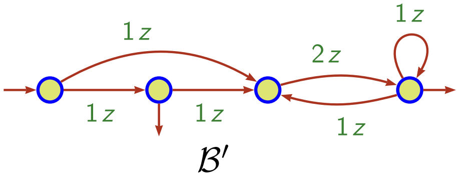
The generating function

A language $K = (c + d\,c + d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$ that is,

an *unambiguous* automaton



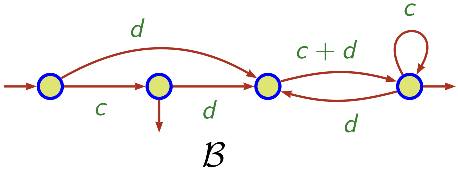
is transformed into an automaton over $\{z\}^*$ with weight in $\mathbb{N}$



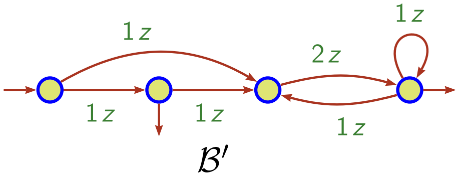
The generating function

A language $K = (c + d\,c + d\,d)^* \setminus \{c\,c\,(c+d)^* \cup 1_{B^*}\}$ that is,

an *unambiguous* automaton



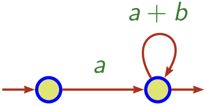
is transformed into an automaton over $\{z\}^*$ with weight in $\mathbb{N}$



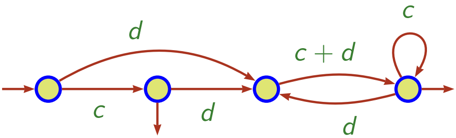
which realises the generating function $\mathrm{G}_K(z) = \sum_{n \in \mathbb{N}} \mathrm{g}_K(n)\, z^n$

Two regular languages with equal growth functions

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



$\mathcal{A}$



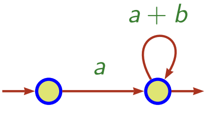
$\mathcal{B}$

Two regular languages with equal growth functions

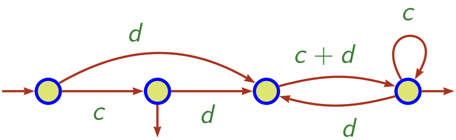
(i) Two finite automata $\mathcal{A}$ and $\mathcal{B}$, preferably unambiguous,

(ii) transformed into $\mathcal{A}'$ and $\mathcal{B}'$, over $\{z\}^*$ with multiplicity in $\mathbb{N}$,
which realise the *generating functions* $\mathsf{G}_L(z)$ and $\mathsf{G}_K(z)$:

$$\mathsf{G}_L(z) = \sum_{n \in \mathbb{N}} \mathsf{g}_L(n)\, z^n \qquad \text{and} \qquad \mathsf{G}_K(z) = \sum_{n \in \mathbb{N}} \mathsf{g}_K(n)\, z^n \ ,$$



$\mathcal{A}$



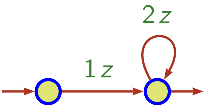
$\mathcal{B}$

Two regular languages with equal growth functions

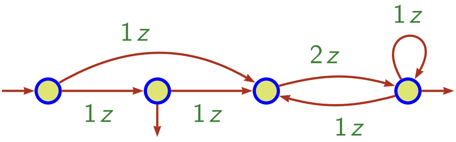
(i) Two finite automata $\mathcal{A}$ and $\mathcal{B}$, preferably unambiguous,

(ii) transformed into $\mathcal{A}'$ and $\mathcal{B}'$, over $\{z\}^*$ with multiplicity in $\mathbb{N}$, which realise the *generating functions* $\mathsf{G}_L(z)$ and $\mathsf{G}_K(z)$:

$$\mathsf{G}_L(z) = \sum_{n\in\mathbb{N}} \mathsf{g}_L(n)\, z^n \qquad \text{and} \qquad \mathsf{G}_K(z) = \sum_{n\in\mathbb{N}} \mathsf{g}_K(n)\, z^n \ ,$$



$\mathcal{A}'$



$\mathcal{B}'$

Two regular languages with equal growth functions

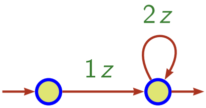
(i) Two finite automata $\mathcal{A}$ and $\mathcal{B}$, preferably unambiguous,

(ii) transformed into $\mathcal{A}'$ and $\mathcal{B}'$, over $\{z\}^*$ with multiplicity in $\mathbb{N}$, which realise the *generating functions* $\mathsf{G}_L(z)$ and $\mathsf{G}_K(z)$:

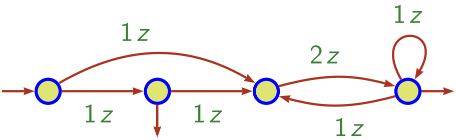
$$\mathsf{G}_L(z) = \sum_{n\in\mathbb{N}} \mathsf{g}_L(n)\, z^n \qquad \text{and} \qquad \mathsf{G}_K(z) = \sum_{n\in\mathbb{N}} \mathsf{g}_K(n)\, z^n \ ,$$

(iii) and whose equivalence is decidable

(Schützenberger 1961, Eilenberg 1974).



$\mathcal{A}'$



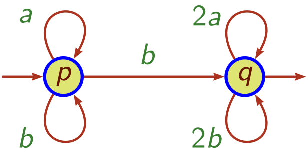
$\mathcal{B}'$

Generating functions

are realised

by weighted automata

Weighted automata, a first look



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

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

$$b\,a\,b \quad \longmapsto \quad 5 \qquad\qquad \forall w \in A^* \qquad w \quad \longmapsto \quad \langle w \rangle_2$$

$$s \colon A^* \longrightarrow \mathbb{N} \qquad s \;:\; w \quad \longmapsto \quad <s, w> \qquad\qquad s \in \mathbb{N}\langle\langle A^* \rangle\rangle$$

$$s = b + a\,b + 2\,b\,a + 3\,b\,b + a\,a\,b$$
$$+ 2\,a\,b\,a + 3\,a\,b\,b + 4\,b\,a\,a + 5\,b\,a\,b + \ldots$$

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 $\mathbb{B}$		decidable
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	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
$\mathrm{Rat}\, B^*$	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
$\mathrm{Rat}\, B^*$		undecidable
$\mathbb{N}\mathrm{Rat}\, B^*$	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
$(\mathbb{Z}, \mathsf{min}, +)$	undecidable
$\mathrm{Rat}\, B^*$	undecidable
$\mathbb{N}\mathrm{Rat}\, B^*$	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
$\mathrm{Rat}\, B^*$	undecidable
$\mathbb{N}\mathrm{Rat}\, B^*$	decidable

The equivalence of

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

Part III

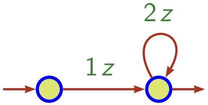
Proof of the Rational Bijection Theorem

The Rational Bijection Theorem

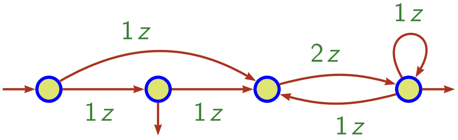
Proposition

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.

The Conjugacy Theorem



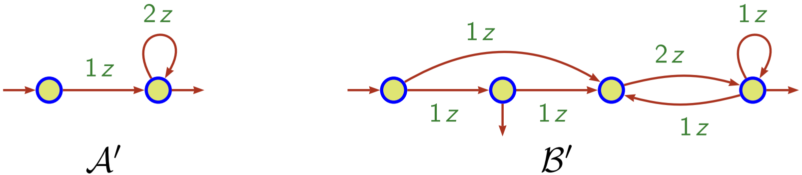
$\mathcal{A}'$



$\mathcal{B}'$

The Conjugacy Theorem

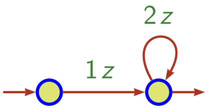
Theorem (BLS)
Two $\mathbb{N}$-automata are equivalent if, and only if they are conjugate to a same third $\mathbb{N}$-automaton.



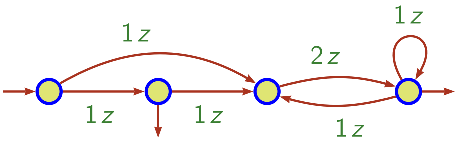
The Conjugacy Theorem

A confession

Automata are matrices



$\mathcal{A}'$

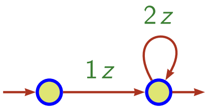


$\mathcal{B}'$

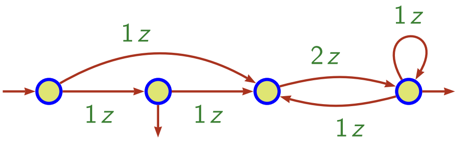
A confession

Automata are matrices

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



$\mathcal{A}'$



$\mathcal{B}'$

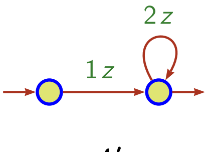
The Conjugacy Theorem

A confession

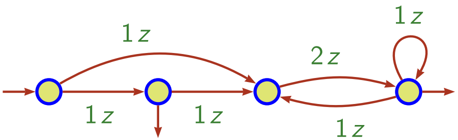
Automata are matrices

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

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



$\mathcal{A}'$

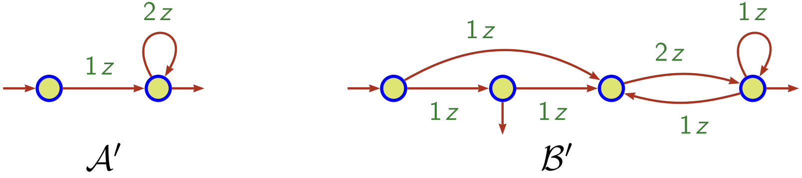


$\mathcal{B}'$

The Conjugacy Theorem

Theorem (BLS)

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



The Conjugacy 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 :

$$I\,X = J, \qquad E\,X = X\,F, \quad \text{and} \quad T = X\,U \ .$$

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

The Conjugacy 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 :

$$I\,X = J, \qquad E\,X = X\,F, \quad \text{and} \quad T = X\,U \ .$$

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

Conjugacy is a *preorder*

(transitive and reflexive, but not symmetric).

The Conjugacy 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 :

$$I\,X = J, \qquad E\,X = X\,F, \quad \text{and} \quad T = X\,U \ .$$

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

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

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

and then $\quad I\,E^*\,T = J\,F^*\,U$

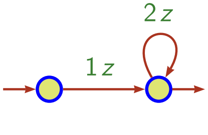
The Conjugacy Theorem

Theorem (BLS)

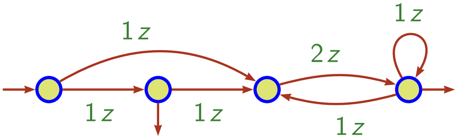
Two $\mathbb{N}$-automata $\mathcal{A}$ and $\mathcal{B}$ are equivalent if, and only if, there exist an $\mathbb{N}$-automaton $\mathcal{C}$ (and $\mathbb{N}$-matrices X and Y) such that

$$\mathcal{A} \overset{X}{\Longleftarrow} \mathcal{C} \overset{Y}{\Longrightarrow} \mathcal{B}$$

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



$\mathcal{A}'$



$\mathcal{B}'$

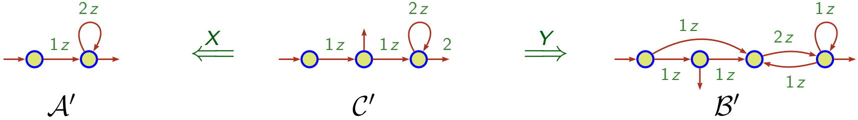
The Conjugacy Theorem

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

$$\mathcal{A} \stackrel{X}{\Longleftarrow} \mathcal{C} \stackrel{Y}{\Longrightarrow} \mathcal{B}$$

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



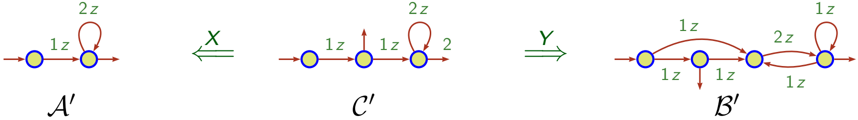
The Conjugacy Theorem

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

$$\mathcal{A} \overset{X}{\Longleftarrow} \mathcal{C} \overset{Y}{\Longrightarrow} \mathcal{B}$$

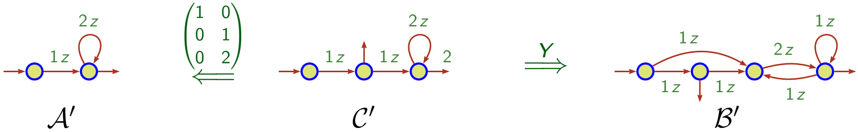
Moreover, $\mathcal{C}$ is effectively computable from $\mathcal{A}$ and $\mathcal{B}$.

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



The Conjugacy Theorem

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



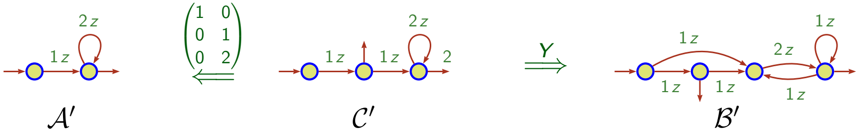
The Conjugacy Theorem

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

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

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

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



$$\mathcal{A}' \quad \stackrel{X}{\Longleftarrow} \quad \mathcal{C}' \quad \stackrel{Y}{\Longrightarrow} \quad \mathcal{B}'$$

The Finite Equivalence Theorem for automata

$$\mathcal{A}' \quad \stackrel{X}{\Longleftarrow} \quad \mathcal{C}' \quad \stackrel{Y}{\Longrightarrow} \quad \mathcal{B}'$$

The Finite Equivalence Theorem for automata

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* $\mathcal{B}$ *.*

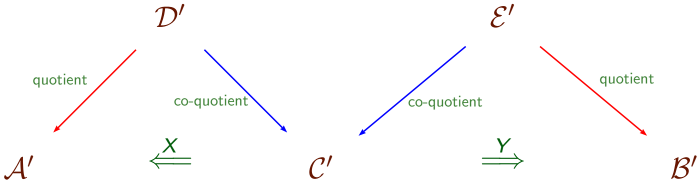
$$\mathcal{A}' \quad \stackrel{X}{\Longleftarrow} \quad \mathcal{C}' \quad \stackrel{Y}{\Longrightarrow} \quad \mathcal{B}'$$

The Finite Equivalence Theorem for automata

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 $\mathcal{B}$.*

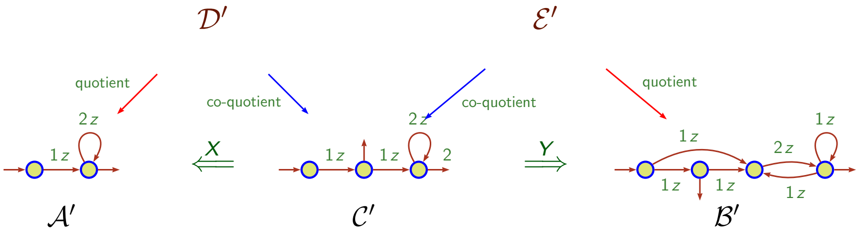


The Finite Equivalence Theorem for automata

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 $\mathcal{B}$.

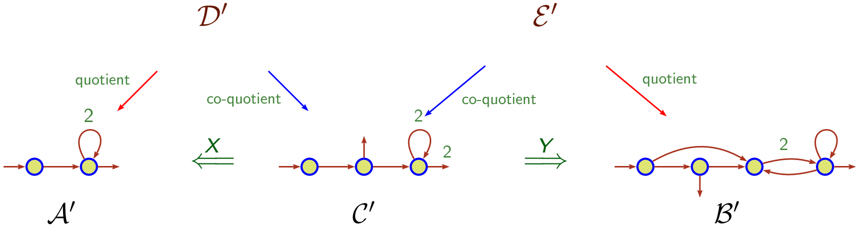


The Finite Equivalence Theorem for automata

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 $\mathcal{B}$.

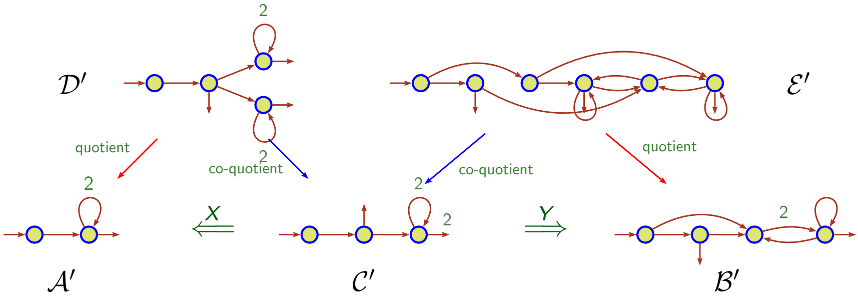


The Finite Equivalence Theorem for automata

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 $\mathcal{B}$.

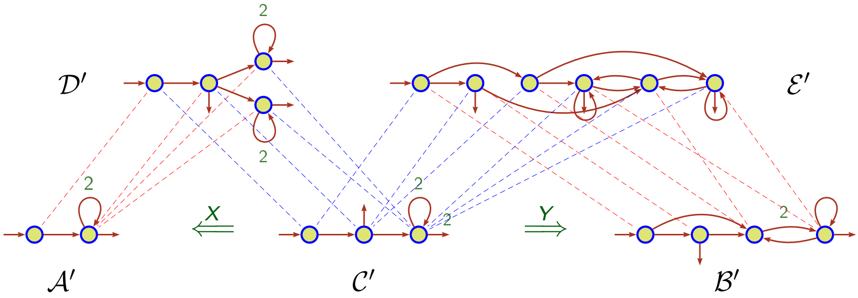


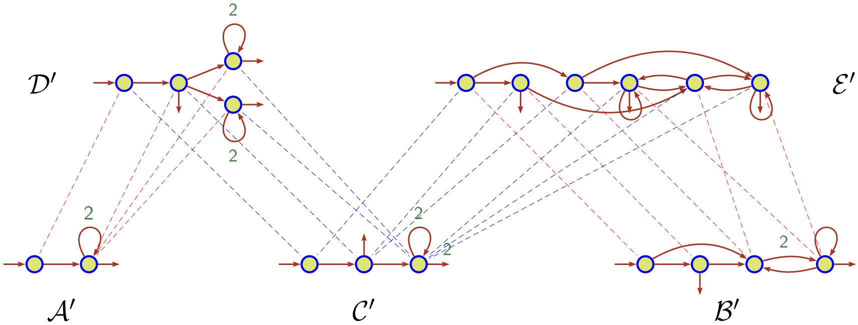
The Finite Equivalence Theorem for automata

A structural interpretation of conjugacy

Theorem (BLS)

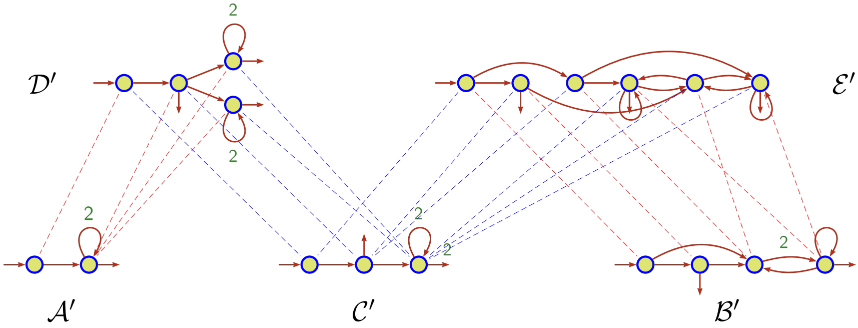
*Let $\mathcal{A}$ and $\mathcal{B}$ be two conjugate $\mathbb{N}$-automata.
Then, there exist an $\mathbb{N}$-automaton $\mathcal{D}$
such that $\mathcal{A}$ is a co-quotient of $\mathcal{D}$
and $\mathcal{B}$ is an quotient of $\mathcal{D}$.
Moreover, $\mathcal{D}$ is effectively computable from $\mathcal{A}$ and $\mathcal{B}$.*



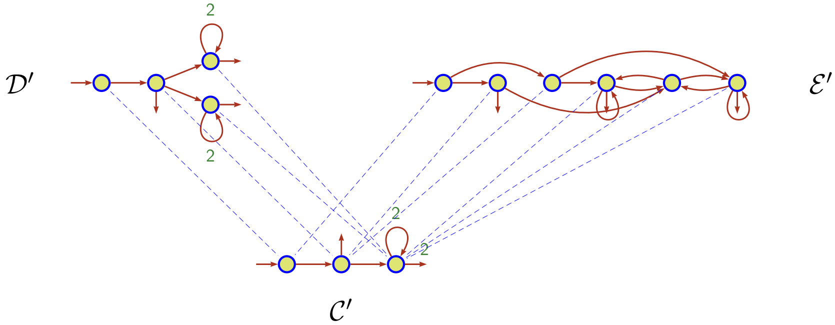


$\mathcal{D}'$
$\mathcal{E}'$
$\mathcal{A}'$
$\mathcal{C}'$
$\mathcal{B}'$
2
2
2
2
2
2

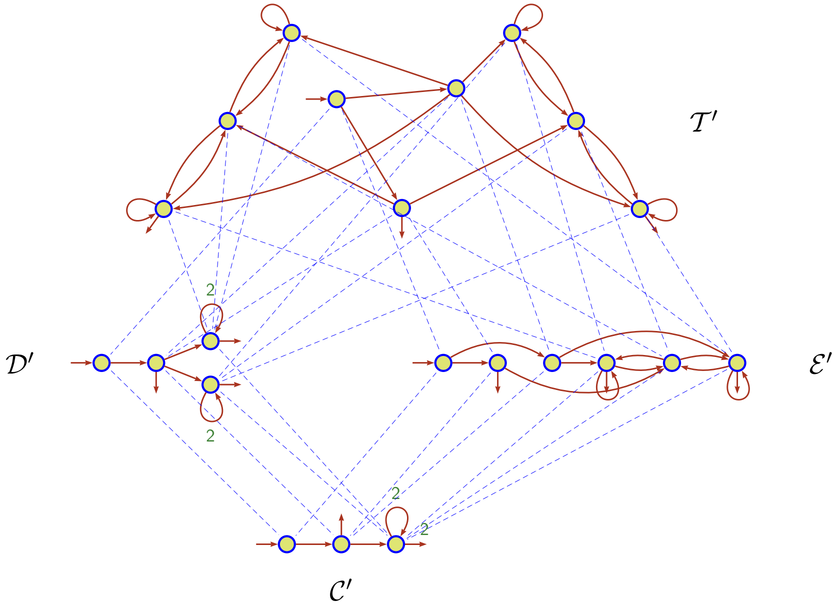
A technical proposition



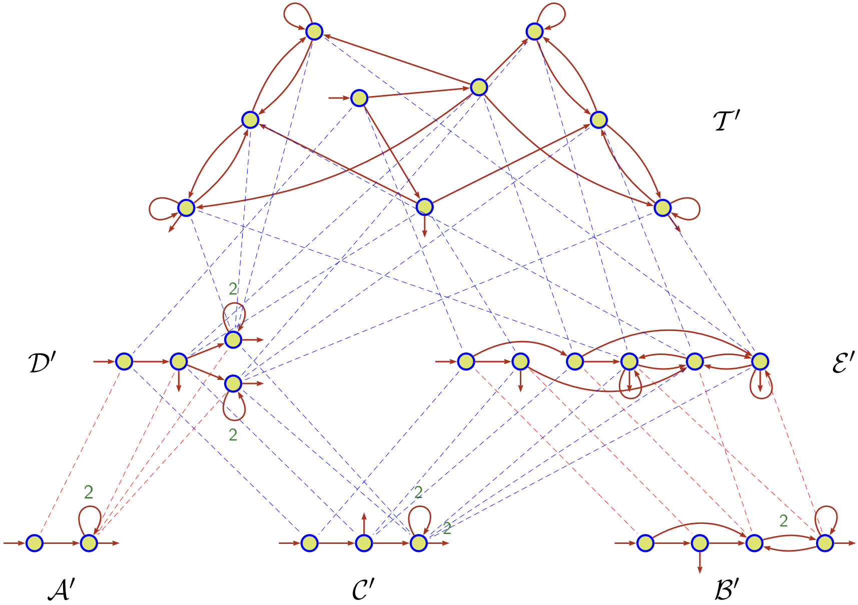
A technical proposition



A technical proposition

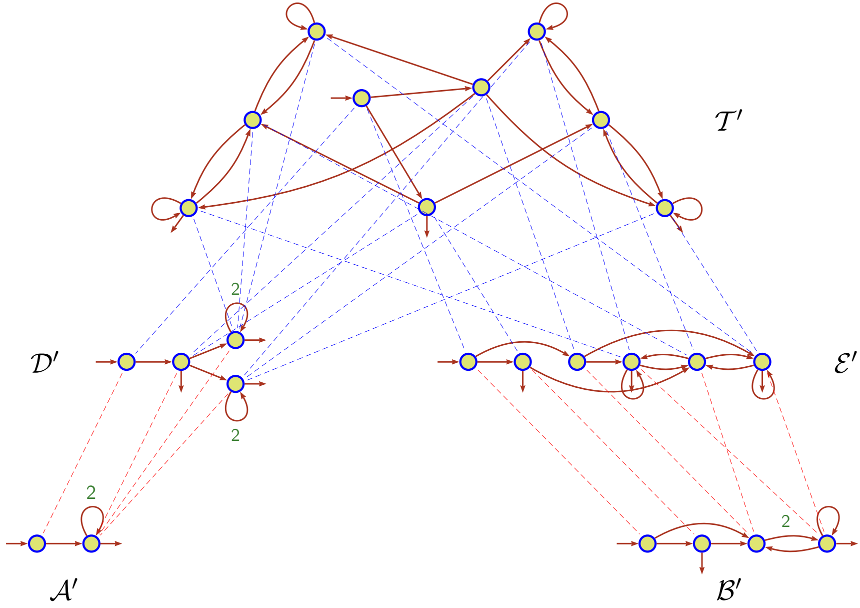


The harvest

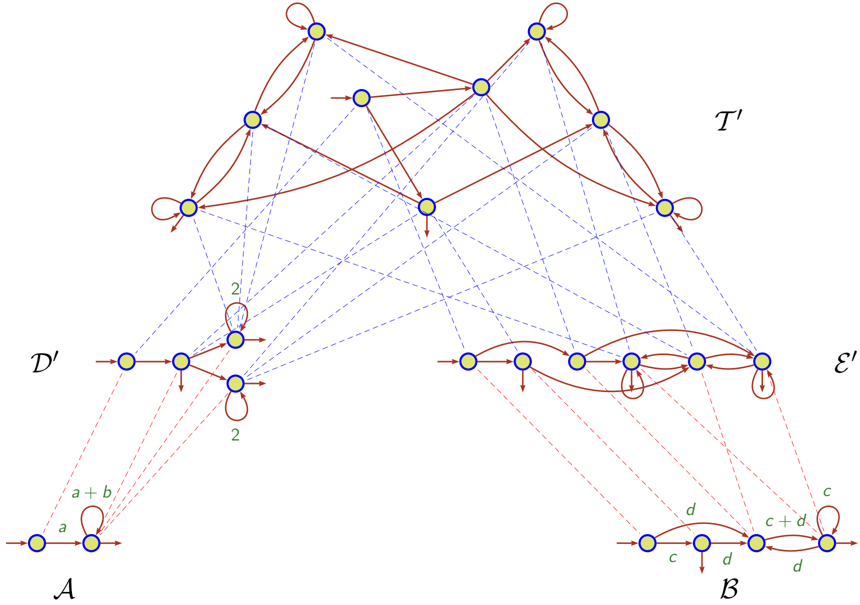


$\mathcal{T}'$
$\mathcal{D}'$
$\mathcal{E}'$
$\mathcal{A}'$
$\mathcal{C}'$
$\mathcal{B}'$
2
2
2
2
2
2

The harvest

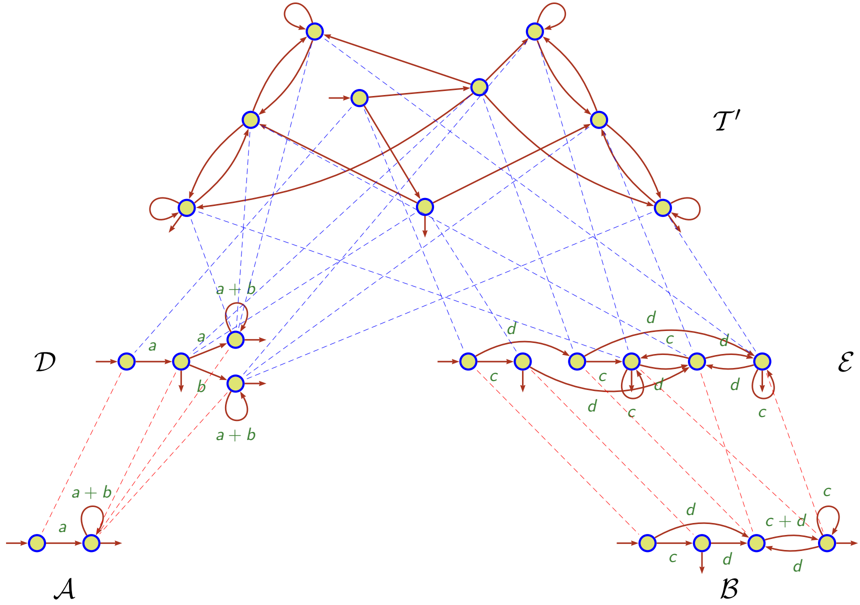


The harvest

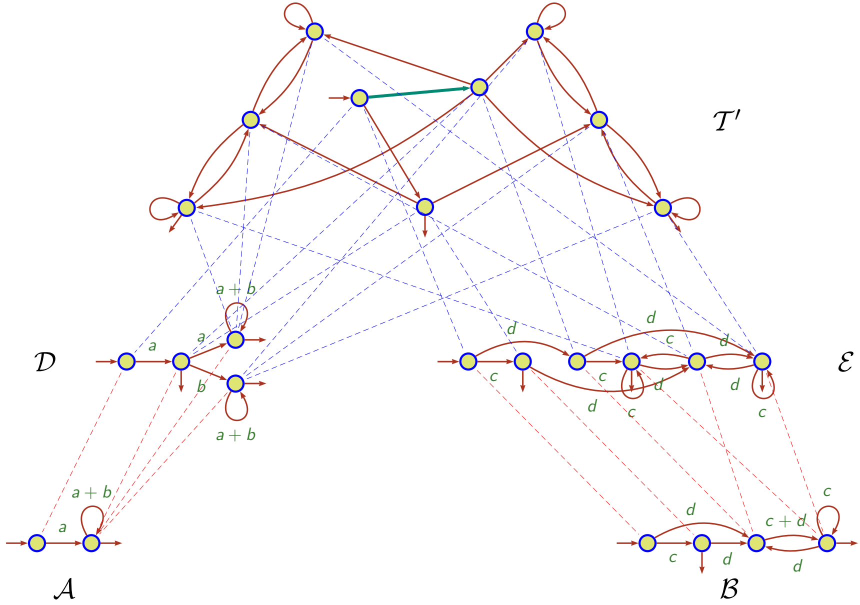


$\mathcal{T}'$
$\mathcal{D}'$
$\mathcal{E}'$
2
2
$a+b$
a
$\mathcal{A}$
d
$c+d$
c
c
d
d
$\mathcal{B}$

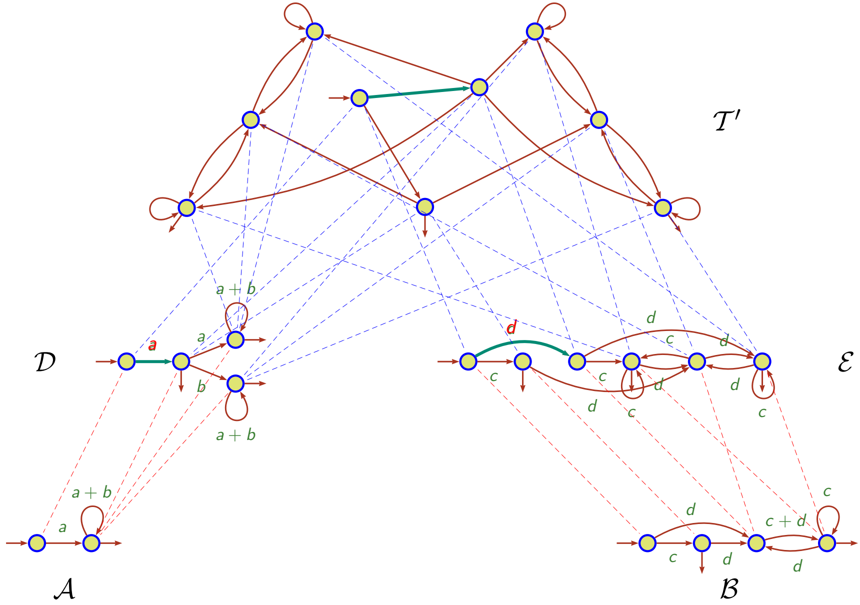
The harvest



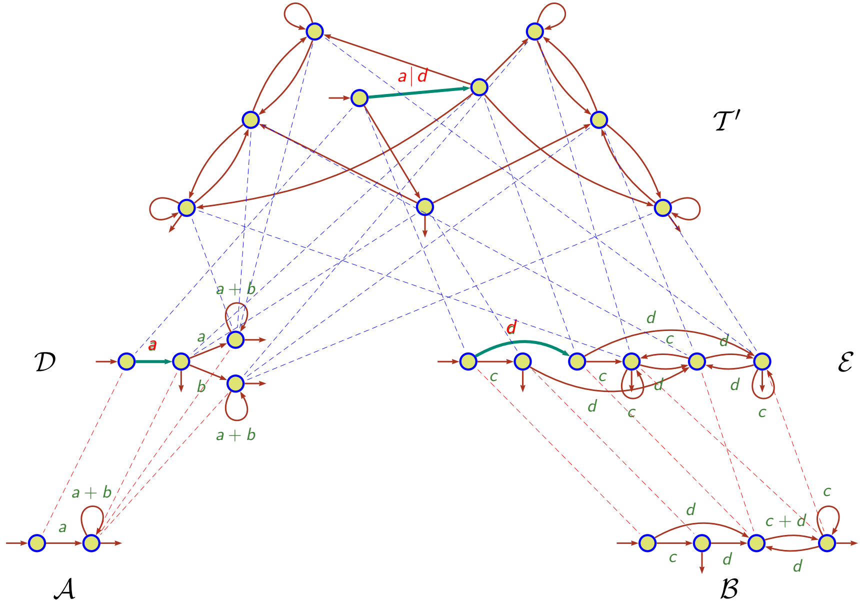
The harvest



The harvest

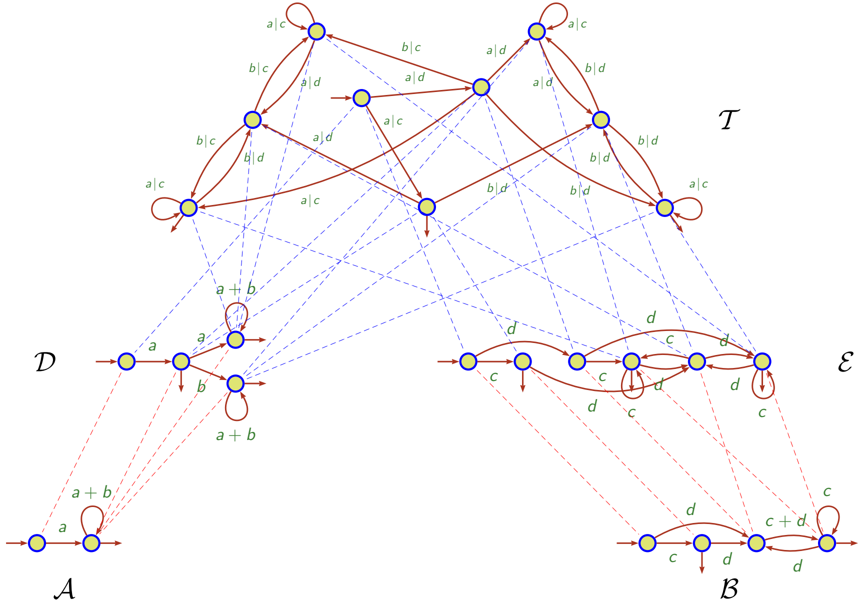


The harvest



$\mathcal{T}'$
$a|d$
$\mathcal{D}$
a
a
b
$a+b$
$a+b$
$\mathcal{E}$
d
c
c
d
c
d
d
d
c
c
d
$\mathcal{A}$
$a+b$
a
$\mathcal{B}$
d
c
d
$c+d$
d
c

The harvest



a|c
b|c
a|d
b|d
$\mathcal{T}$
a + b
a
b
$\mathcal{D}$
c
d
c + d
$\mathcal{E}$
$\mathcal{A}$
$\mathcal{B}$

Part IV

The foundations

The conjugacy theorems

Theorem

Let $\mathbb{K}$ *be* $\mathbb{B}$, $\mathbb{N}$, $\mathbb{Z}$ *, or any (skew) fields.*

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

$$\mathcal{A} \overset{X}{\Longleftarrow} \mathcal{C} \overset{Y}{\Longrightarrow} \mathcal{B}$$

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

The conjugacy theorems

Theorem

Let $\mathbb{K}$ *be* $\mathbb{B}$, $\mathbb{N}$, $\mathbb{Z}$, *or any (skew) fields.*
Two $\mathbb{K}$*-automata* $\mathcal{A}$ *and* $\mathcal{B}$ *are equivalent if, and only if, there exist a* $\mathbb{K}$*-automaton* $\mathcal{C}$ *(and* $\mathbb{K}$*-matrices* X *and* Y*) such that*

$$\mathcal{A} \stackrel{X}{\Longleftarrow} \mathcal{C} \stackrel{Y}{\Longrightarrow} \mathcal{B}$$

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

Theorem

Two functional transducers $\mathcal{A}$ *and* $\mathcal{B}$ *are equivalent if, and only if, there exist a functional transducer* $\mathcal{C}$ *(and word-matrices* X *and* Y*) such that*

$$\mathcal{A} \stackrel{X}{\Longleftarrow} \mathcal{C} \stackrel{Y}{\Longrightarrow} \mathcal{B}$$

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

The conjugacy theorems

Theorem

Let $\mathbb{K}$ be $\mathbb{B}$, $\mathbb{N}$, $\mathbb{Z}$, or any (skew) fields.
Two $\mathbb{K}$-automata $\mathcal{A}$ and $\mathcal{B}$ are equivalent if, and only if, there exist a $\mathbb{K}$-automaton $\mathcal{C}$ (and $\mathbb{K}$-matrices X and Y) such that

$$\mathcal{A} \overset{X}{\Longleftarrow} \mathcal{C} \overset{Y}{\Longrightarrow} \mathcal{B}$$

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

The path to the theorem:

- understanding reduction
- understanding reduction as a conjugacy
- performing joint reduction

Finite Equivalence Theorems for weighted automata

The Finite Equivalence Theorem

A standard result in symbolic dynamics

Theorem

Two irreducible sofic shifts are finitely equivalent if, and only if, they have the same entropy.

Finite Equivalence Theorems for weighted automata

Theorem

Let $\mathbb{K} = \mathbb{B}$ *or* $\mathbb{N}$, $\mathcal{A}$ *and* $\mathcal{B}$ *two trim* $\mathbb{K}$*-automata.*

Then $\mathcal{A} \stackrel{X}{\Longrightarrow} \mathcal{B}$ *if, and only if, there exists a* $\mathbb{K}$*-automaton* $\mathcal{C}$ *which is a co-*$\mathbb{K}$*-covering of* $\mathcal{A}$ *and a* $\mathbb{K}$*-covering of* $\mathcal{B}$.

Finite Equivalence Theorems for weighted automata

Theorem

Let $\mathbb{K} = \mathbb{B}$ or $\mathbb{N}$, $\mathcal{A}$ and $\mathcal{B}$ two trim $\mathbb{K}$-automata.
Then $\mathcal{A} \stackrel{X}{\Longrightarrow} \mathcal{B}$ if, and only if, there exists a $\mathbb{K}$-automaton $\mathcal{C}$ which is a co-$\mathbb{K}$-covering of $\mathcal{A}$ and a $\mathbb{K}$-covering of $\mathcal{B}$.

Definition

$\mathcal{C}$ is a $\mathbb{K}$-covering of $\mathcal{B}$ if $\mathcal{C} \stackrel{H_\varphi}{\Longrightarrow} \mathcal{B}$
where H_φ is the matrix of a surjective map.

$\mathcal{C}$ is a co-$\mathbb{K}$-covering of $\mathcal{A}$ if $\mathcal{C}^{\mathrm{t}}$ is a $\mathbb{K}$-covering of $\mathcal{A}^{\mathrm{t}}$

that is, if $\mathcal{A} \stackrel{H_\psi^{\mathrm{t}}}{\Longrightarrow} \mathcal{C}$ where H_ψ is the matrix of a surjective map.

Finite Equivalence Theorems for weighted automata

Theorem

Let $\mathbb{K} = \mathbb{B}$ *or* $\mathbb{N}$, $\mathcal{A}$ *and* $\mathcal{B}$ *two trim* $\mathbb{K}$*-automata.*
Then $\mathcal{A} \overset{X}{\Longrightarrow} \mathcal{B}$ *if, and only if, there exists a* $\mathbb{K}$*-automaton* $\mathcal{C}$
which is a co-$\mathbb{K}$*-covering of* $\mathcal{A}$ *and a* $\mathbb{K}$*-covering of* $\mathcal{B}$.

Theorem

Let $\mathbb{K} = \mathbb{Z}$ *or a field* $\mathbb{F}$, $\mathcal{A}$ *and* $\mathcal{B}$ *two* $\mathbb{K}$*-automata.*
Then $\mathcal{A} \overset{X}{\Longrightarrow} \mathcal{B}$ *if, and only if,*
$\exists$ $\mathbb{K}$*-automata* $\mathcal{C}$ *and* $\mathcal{D}$ *and a circulation matrix* D
$\mathcal{C}$ *co-*$\mathbb{K}$*-covering of* $\mathcal{A}$, $\mathcal{D}$ $\mathbb{K}$*-covering of* $\mathcal{B}$, *and* $\mathcal{C} \overset{D}{\Longrightarrow} \mathcal{D}$.