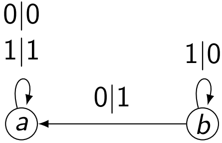
Mealy Automata, Singular Points and Wang Tilings

Daniele D'Angeli, Thibault Godin, Ines Klimann, Matthieu Picantin, and Emanuele Rodaro

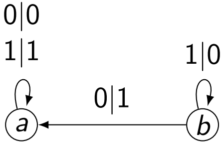
Journées SDA2, 3 juillet 2017

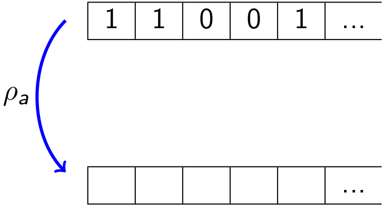
ANR JCJC 12 JS02 012 01

How to Generate Groups with a (Mealy) Automaton

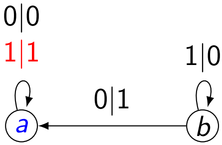


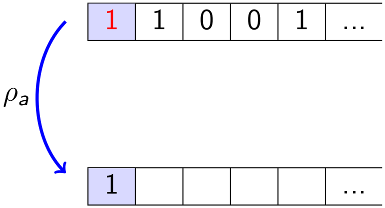
How to Generate Groups with a (Mealy) Automaton



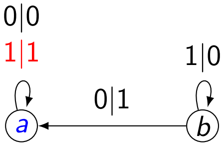


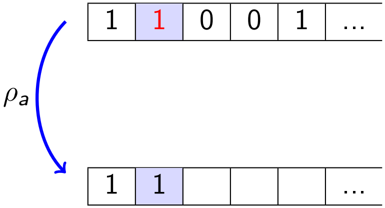
How to Generate Groups with a (Mealy) Automaton



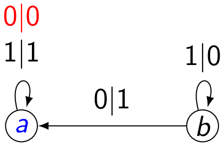


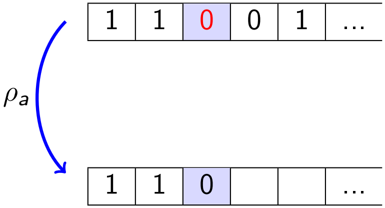
How to Generate Groups with a (Mealy) Automaton



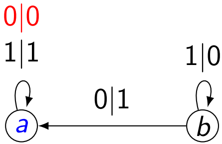


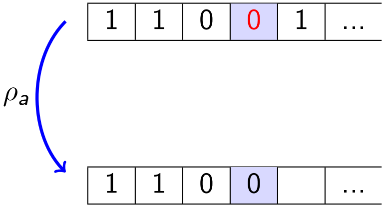
How to Generate Groups with a (Mealy) Automaton



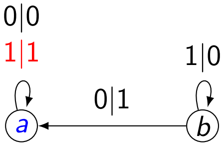


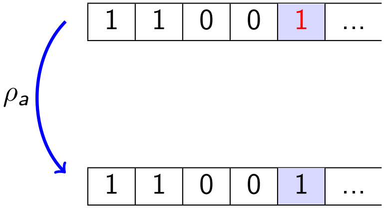
How to Generate Groups with a (Mealy) Automaton



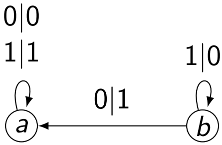


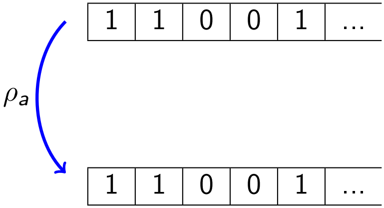
How to Generate Groups with a (Mealy) Automaton





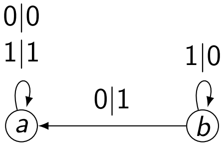
How to Generate Groups with a (Mealy) Automaton

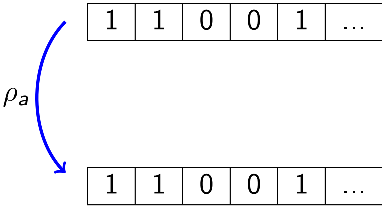




$\rho_a : \Sigma^* \to \Sigma^*$

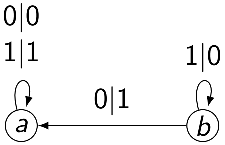
How to Generate Groups with a (Mealy) Automaton

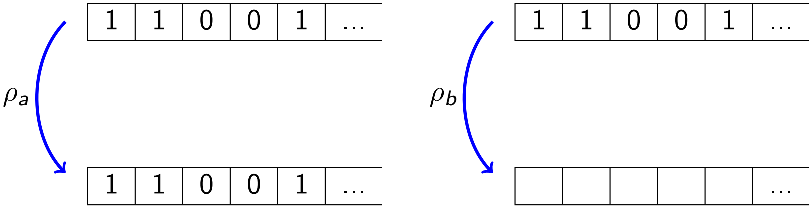




$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

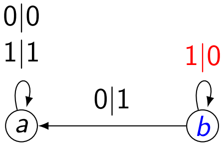
How to Generate Groups with a (Mealy) Automaton

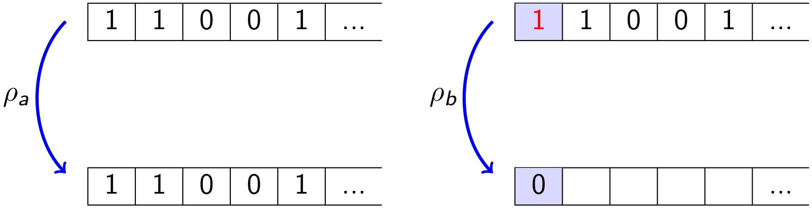




$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

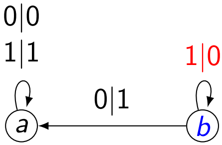
How to Generate Groups with a (Mealy) Automaton

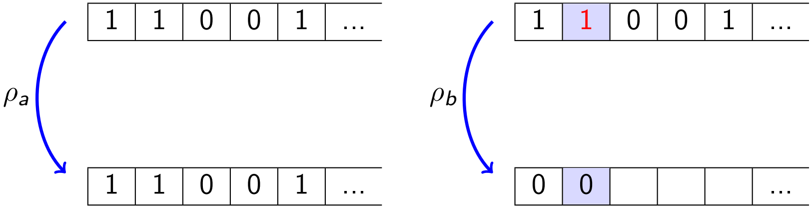




$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

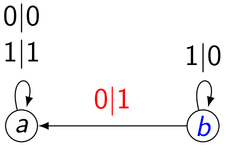
How to Generate Groups with a (Mealy) Automaton

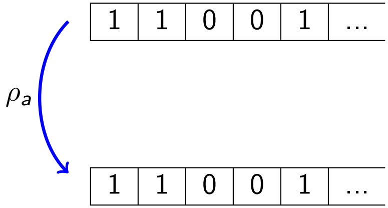


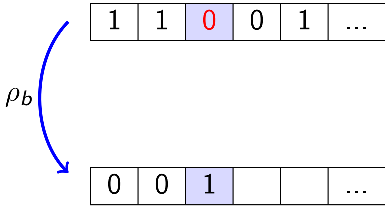


$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

How to Generate Groups with a (Mealy) Automaton

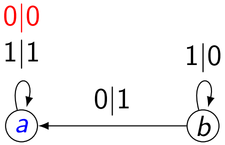


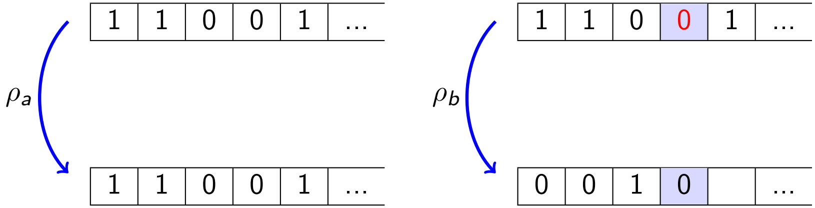




$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

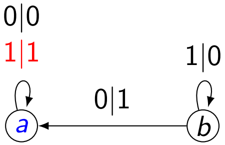
How to Generate Groups with a (Mealy) Automaton

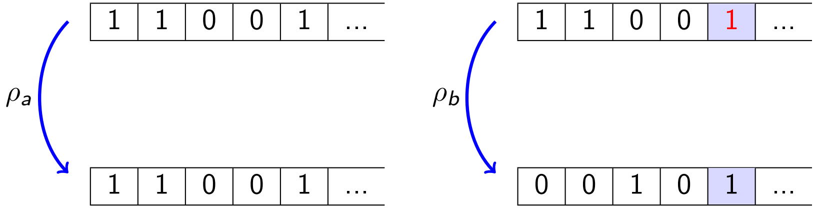




$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

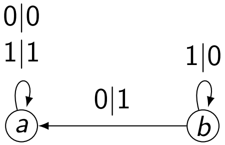
How to Generate Groups with a (Mealy) Automaton

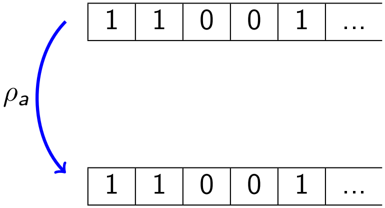


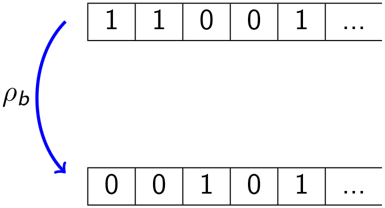


$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

How to Generate Groups with a (Mealy) Automaton



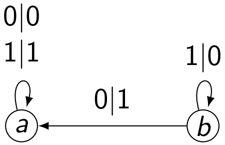


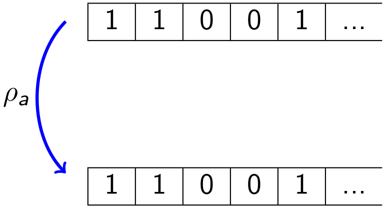


$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

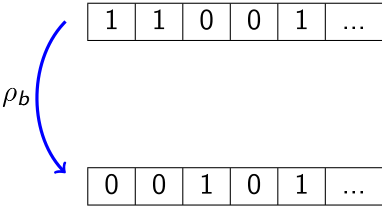
$$\rho_b : \Sigma^* \to \Sigma^*$$

How to Generate Groups with a (Mealy) Automaton



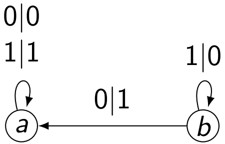


$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

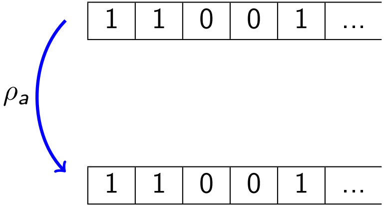


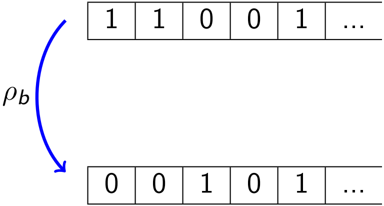
$$\rho_b : \Sigma^* \to \Sigma^*$$
$$u \mapsto u + 1$$

How to Generate Groups with a (Mealy) Automaton



$\langle \mathcal{A} \rangle = \langle \rho_a, \rho_b \rangle$

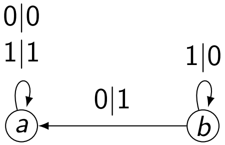




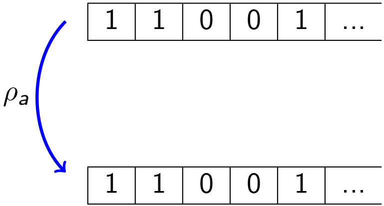
$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

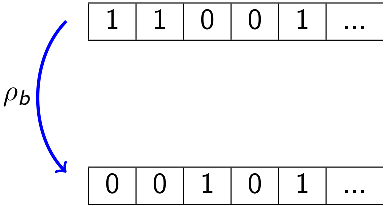
$$\rho_b : \Sigma^* \to \Sigma^*$$
$$u \mapsto u + 1$$

How to Generate Groups with a (Mealy) Automaton



$\langle \mathcal{A} \rangle = \langle \rho_a, \rho_b \rangle = \langle \rho_a, \rho_b, \rho_a^{-1}, \rho_b^{-1} \rangle$

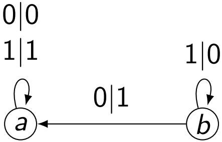




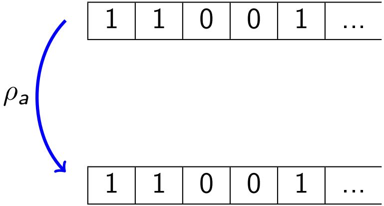
$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

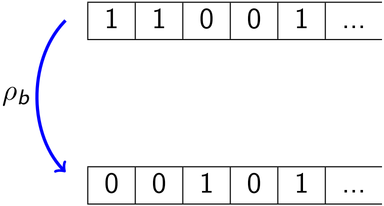
$$\rho_b : \Sigma^* \to \Sigma^*$$
$$u \mapsto u + 1$$

How to Generate Groups with a (Mealy) Automaton



$$\langle \mathcal{A} \rangle = \langle \rho_a, \rho_b \rangle = \langle \rho_a, \rho_b, \rho_a^{-1}, \rho_b^{-1} \rangle \; \simeq \mathbb{Z}$$

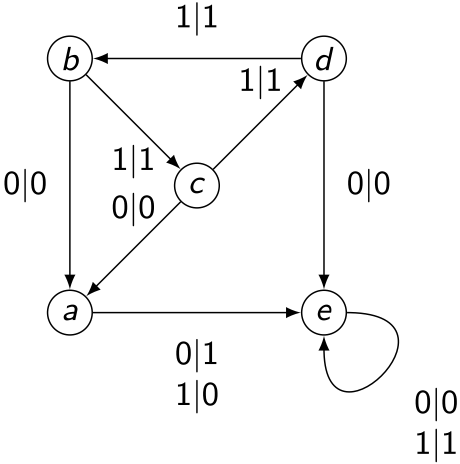




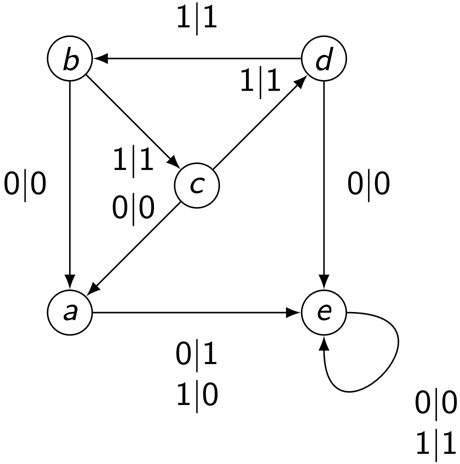
$$\rho_a : \Sigma^* \to \Sigma^*$$
$$u \mapsto u$$

$$\rho_b : \Sigma^* \to \Sigma^*$$
$$u \mapsto u + 1$$

Another Mealy automaton

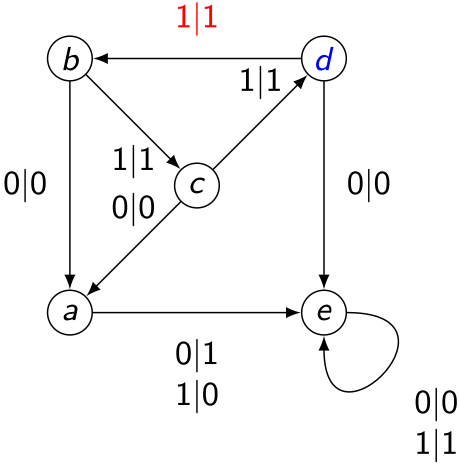


Another Mealy automaton



Grigorchuk automaton

Another Mealy automaton

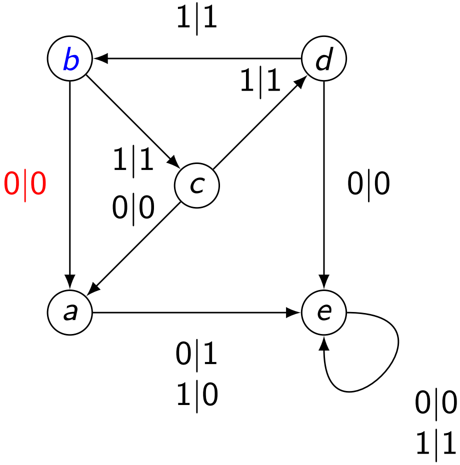


Grigorchuk automaton

$\rho_q : \Sigma^* \to \Sigma^*, \; q \in Q$

	1	0	0	0	1
d					

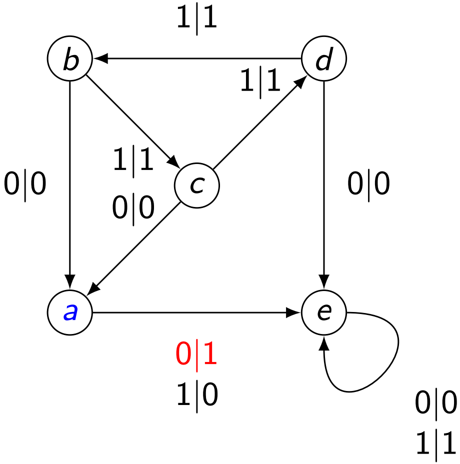
Another Mealy automaton



Grigorchuk automaton

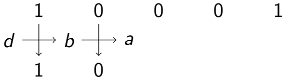
$\rho_q : \Sigma^* \to \Sigma^*, \; q \in Q$

Another Mealy automaton



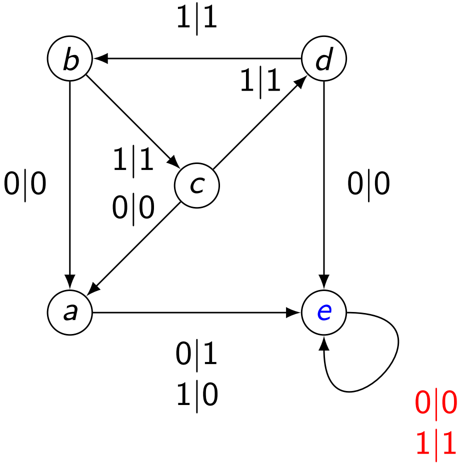
Grigorchuk automaton

$\rho_q : \Sigma^* \to \Sigma^*, \ q \in Q$

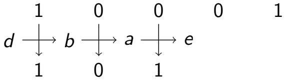


Another Mealy automaton

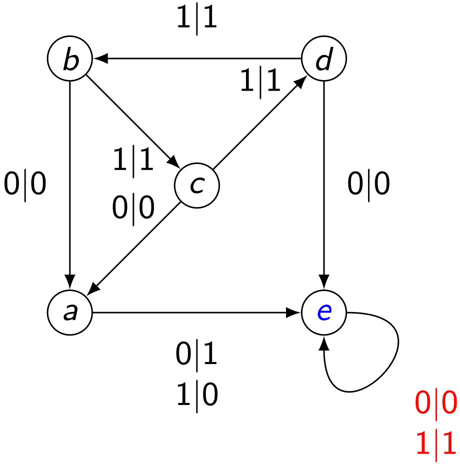
$$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q$$



Grigorchuk automaton

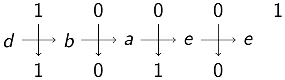


Another Mealy automaton



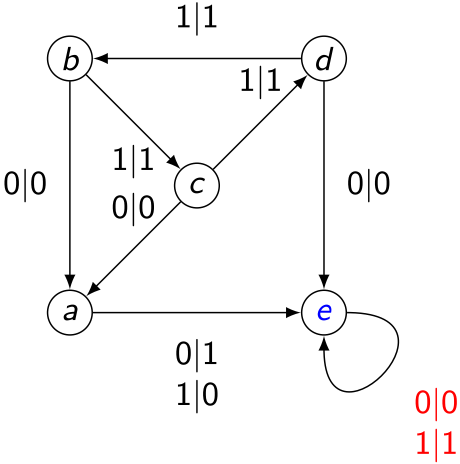
Grigorchuk automaton

$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q$

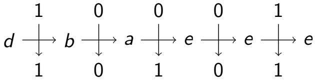


Another Mealy automaton

$$\rho_q : \Sigma^* \to \Sigma^*, \ q \in Q$$

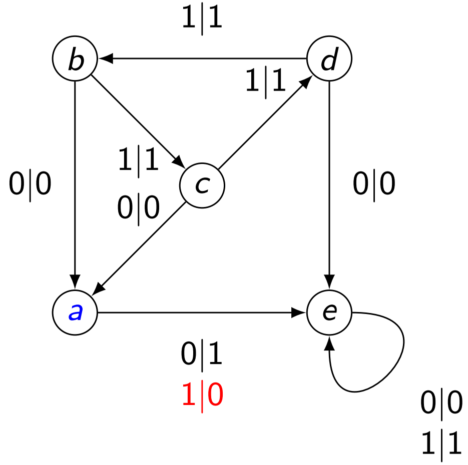


Grigorchuk automaton

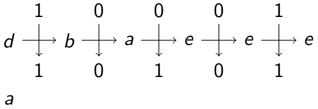


Another Mealy automaton

$$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q^*$$

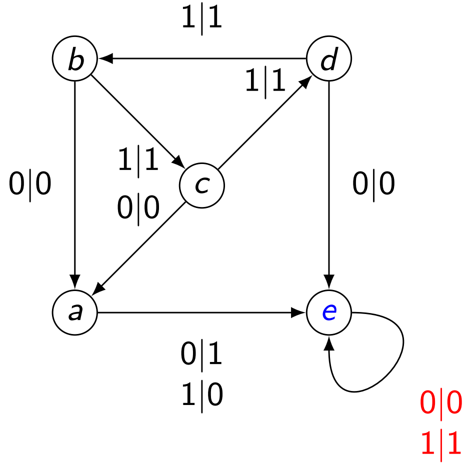


Grigorchuk automaton

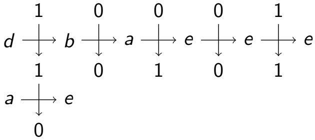


Another Mealy automaton

$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q^*$

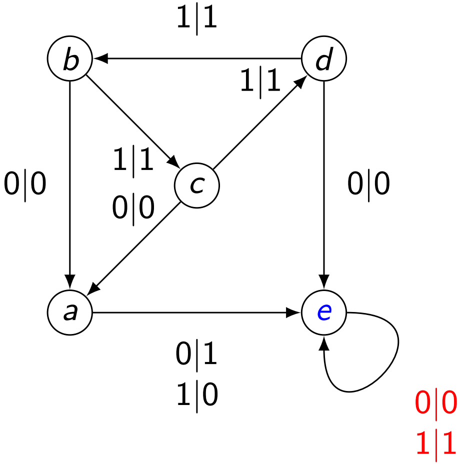


Grigorchuk automaton

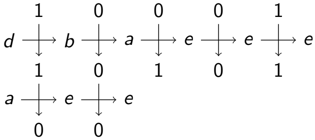


Another Mealy automaton

$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q^*$

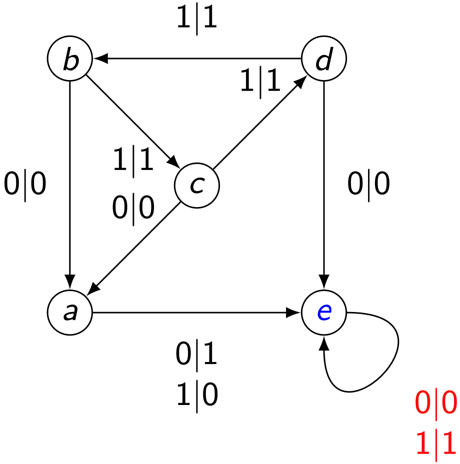


Grigorchuk automaton

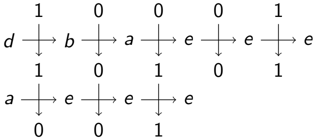


Another Mealy automaton

$\rho_q : \Sigma^* \to \Sigma^*, \ q \in Q^*$

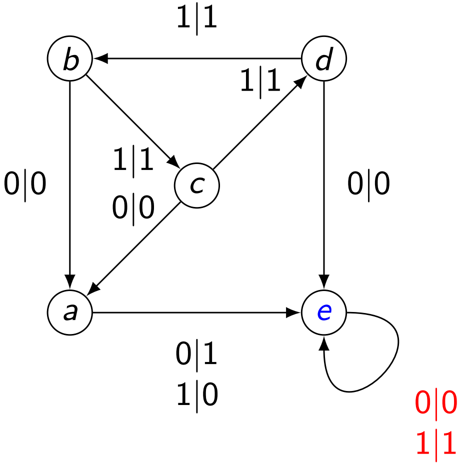


Grigorchuk automaton

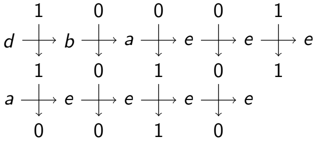


Another Mealy automaton

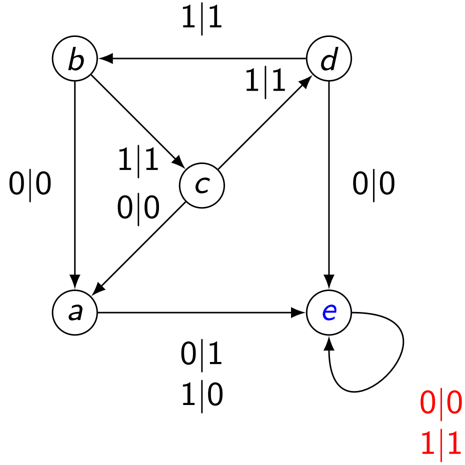
$\rho_q : \Sigma^* \to \Sigma^*, \; q \in Q^*$



Grigorchuk automaton

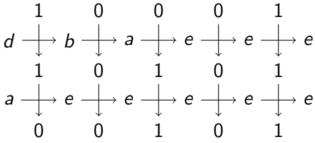


Another Mealy automaton



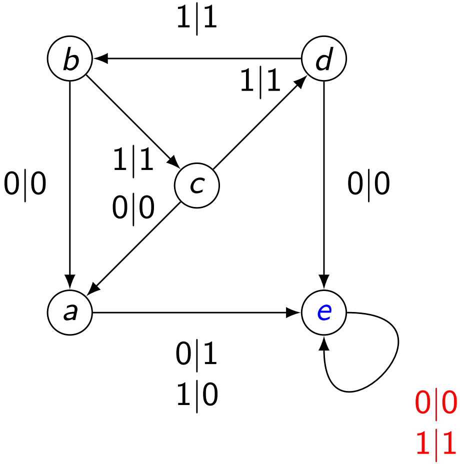
Grigorchuk automaton

$$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q^*$$

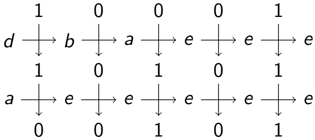


Another Mealy automaton

$$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q^*$$



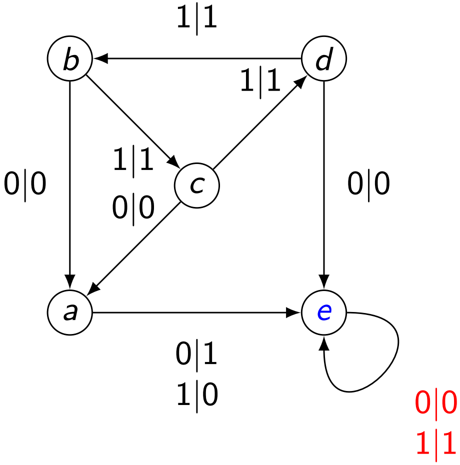
Grigorchuk automaton



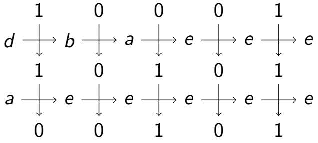
$$\rho_{da}(10001) = \rho_a(\rho_d(10001))$$

Another Mealy automaton

$$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q^*$$



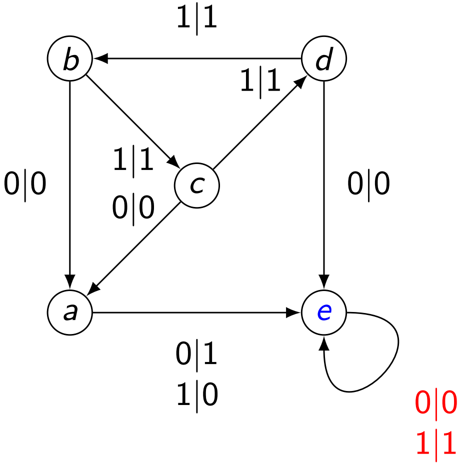
Grigorchuk automaton



$$\rho_{da}(10001) = \rho_a(\rho_d(10001))$$

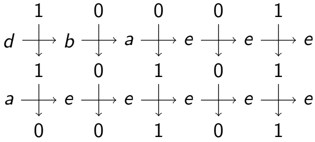
$$\langle \mathcal{A} \rangle := \langle \rho_q \mid q \in Q^* \rangle$$

Another Mealy automaton



Grigorchuk automaton

$\rho_q : \Sigma^* \to \Sigma^*,\ q \in Q^*$



$$\rho_{da}(10001) = \rho_a(\rho_d(10001))$$

$$\langle \mathcal{A} \rangle := \langle \rho_q \mid q \in Q^* \rangle$$

$da \in \mathcal{A}^2$

Order

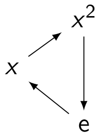
Order of an element

$x \in G$ has finite order if $\exists n \geq 1, x^n = \mathrm{e}$

Order

Order of an element

$x \in G$ has finite order if $\exists n \geq 1, x^n = e$

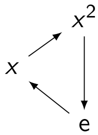


$x^3 = e$, x is torsion

Order

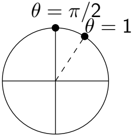
Order of an element

$x \in G$ has finite order if $\exists n \geq 1, x^n = e$



$x^3 = e$, x is torsion

- $\mathbb{Z}/n\mathbb{Z}$ is torsion (all its elements have finite order)
- $\mathbb{Z}$ is torsion-free (0 is the only element of finite order)
- On the circle $\mathbb{R}/2\pi\mathbb{Z}$; $\pi/2$ has finite order but 1 has infinite order



The Burnside problem

Burnside Problem (1902):

Can a finitely generated group have all elements of finite order and be infinite?

The Burnside problem

Burnside Problem (1902):

Can a finitely generated group have all elements of finite order and be infinite?

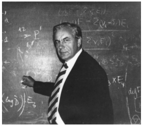
$$G = \langle a_1, a_2, \dots, a_\ell \rangle = \langle \Sigma \rangle \; ; \; g = w \in \Sigma^* \; ; \; \forall g, \exists n, g^n = e$$

The Burnside problem

Burnside Problem (1902):

Can a finitely generated group have all elements of finite order and be infinite?

$$G = \langle a_1, a_2, \ldots, a_\ell \rangle = \langle \Sigma \rangle \ ; \ g = w \in \Sigma^* \ ; \ \forall g, \exists n, g^n = e$$



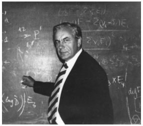
Golod and Shafarevich: Yes! (1964)

The Burnside problem

Burnside Problem (1902):

Can a finitely generated group have all elements of finite order and be infinite?

$$G = \langle a_1, a_2, \ldots, a_\ell \rangle = \langle \Sigma \rangle \; ; \; g = w \in \Sigma^* \; ; \; \forall g, \exists n, g^n = e$$



Golod and Shafarevich: Yes! (1964)

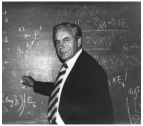
Grigorchuk: example generated by a Mealy automaton (1980)

The Burnside problem

Burnside Problem (1902):

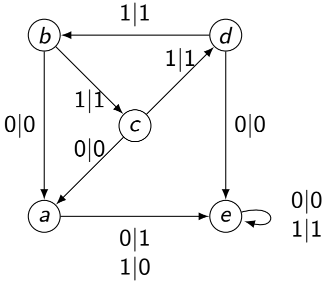
Can a finitely generated group have all elements of finite order and be infinite?

$$G = \langle a_1, a_2, \ldots, a_\ell \rangle = \langle \Sigma \rangle \;;\; g = w \in \Sigma^* \;;\; \forall g, \exists n, g^n = e$$



Golod and Shafarevich: Yes! (1964)

Grigorchuk: example generated by a Mealy automaton (1980)



Growth

Cayley Graph: $\Gamma(G, S)$

$g \xrightarrow{s} h \qquad g.s = h,\ g, h \in G,\ s \in S$

Growth

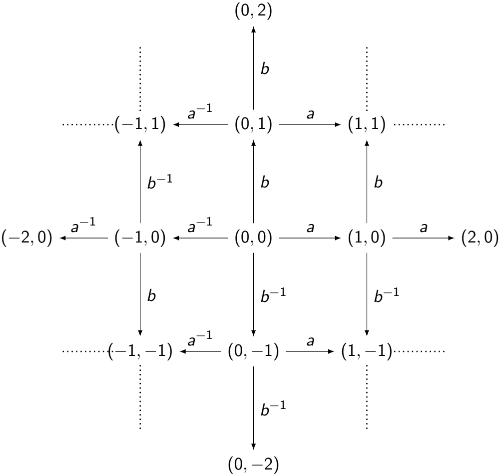
Cayley Graph: $\Gamma(G, S)$ $ex : \mathbb{Z}^2,\ \{a = (0,1), b = (1,0)\}$

$g \xrightarrow{s} h$ $\qquad g.s = h,\ g, h \in G,\ s \in S$

Growth

Cayley Graph: $\Gamma(G, S)$ $ex: \mathbb{Z}^2,\ \{a = (0,1), b = (1,0)\}$

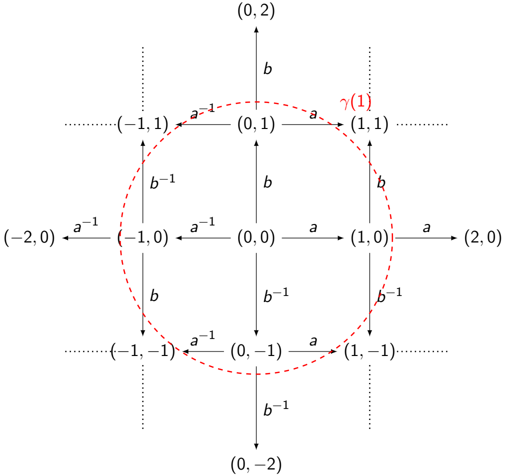
$g \xrightarrow{s} h \qquad g.s = h,\ g, h \in G,\ s \in S$



Growth

Cayley Graph: $\Gamma(G, S)$ $ex : \mathbb{Z}^2, \ \{a = (0, 1), b = (1, 0)\}$

$g \xrightarrow{s} h \qquad g.s = h, \ g, h \in G, \ s \in S$



$\gamma(0) = 1$

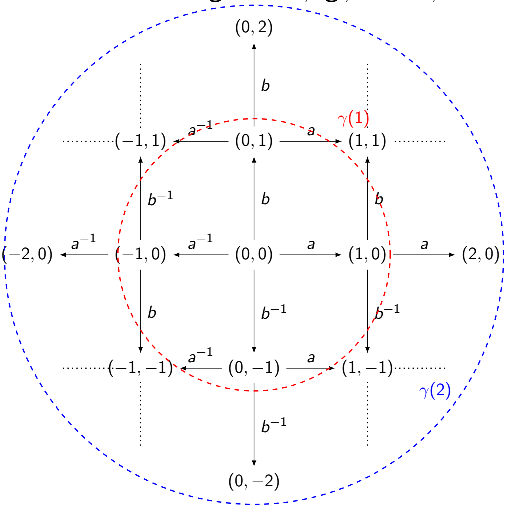
$\gamma(1) = 5$

Growth

Cayley Graph: $\Gamma(G, S)$ $ex : \mathbb{Z}^2, \{a = (0,1), b = (1,0)\}$

$g \xrightarrow{s} h$

$g.s = h, \; g, h \in G, \; s \in S$



$\gamma(0) = 1$

$\gamma(1) = 5$

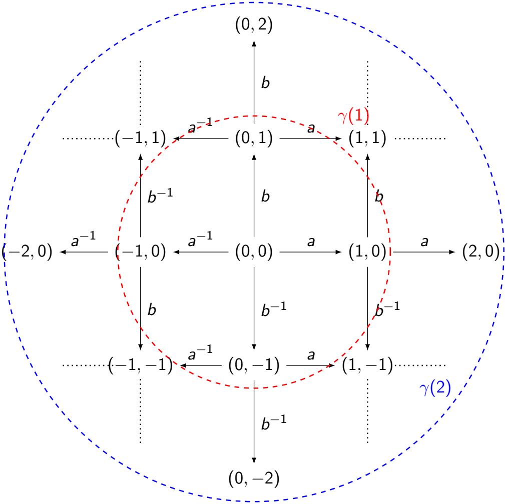
$\gamma(2) = 13$

Growth

Cayley Graph: $\Gamma(G, S)$ $ex : \mathbb{Z}^2,\ \{a = (0,1), b = (1,0)\}$

$g \xrightarrow{s} h$

$g.s = h,\ g, h \in G,\ s \in S$



$\gamma(0) = 1$

$\gamma(1) = 5$

$\gamma(2) = 13$

$\vdots$

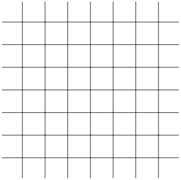
$\gamma(n) = 2n^2 + 2n + 1$

Milnor's Problem

- growth bounded: finite groups

Milnor's Problem

- growth bounded: finite groups
- polynomial growth: $\mathbb{Z}^d$, Abelian groups

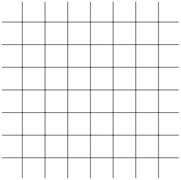


$\Gamma(\mathbb{Z}^2)$

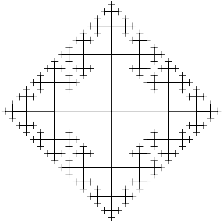
Milnor's Problem

- growth bounded: finite groups
- polynomial growth: $\mathbb{Z}^d$, Abelian groups

- exponential growth: $\mathbb{F}_d$



$\Gamma(\mathbb{Z}^2)$



$\Gamma(\mathbb{F}_2)$

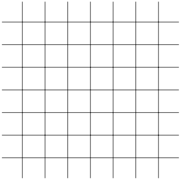
Milnor's Problem

- growth bounded: finite groups
- polynomial growth: $\mathbb{Z}^d$, Abelian groups

Milnor's Problem (1968):

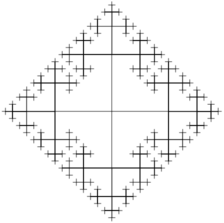
Do groups with growth growth between polynomial and exponential exist?

- exponential growth: $\mathbb{F}_d$



$\Gamma(\mathbb{Z}^2)$

$\Gamma(?)$



$\Gamma(\mathbb{F}_2)$

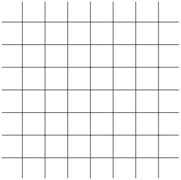
Milnor's Problem

- growth bounded: finite groups
- polynomial growth: $\mathbb{Z}^d$, Abelian groups

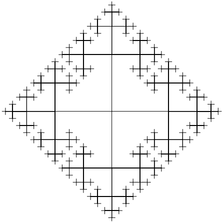
Milnor's Problem (1968):

Do groups with growth growth between polynomial and exponential exist?
1983 (Grigorchuk) Yes, automaton-generated example

- exponential growth: $\mathbb{F}_d$



$\Gamma(\mathbb{Z}^2)$ $\Gamma(?)$ $\Gamma(\mathbb{F}_2)$



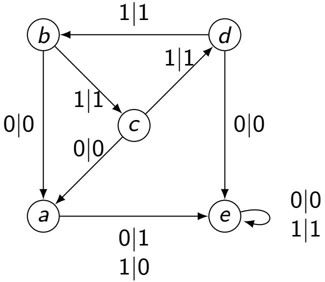
Milnor's Problem

- growth bounded: finite groups
- polynomial growth: $\mathbb{Z}^d$, Abelian groups

Milnor's Problem (1968):

Do groups with growth growth between polynomial and exponential exist?
1983 (Grigorchuk) Yes, automaton-generated example

- exponential growth: $\mathbb{F}_d$



$$e^{n^{0.51}} \leq \gamma(n) \leq e^{n^{0.77}}$$

And more!

How to get a strongly aperodic SFT in the Grigorchuk group

Sebastián Barbieri [1]

1 : ENS DE LYON
École Normale Supérieure - Lyon

An action of a finitely generated group over a Cantor set is called effectively closed if there is a Turing machine which recieves as input a cylinder and a generator and computes an effective aproximation of the complement of the image of such cylinder under the generator. I will show that every effectively closed action of a finitely generated group G can be realized as a factor of the G-subaction of an SFT in $G \times H_1 \times H_2$ for any pair of infinite f.g. groups H_1,H_2. As a corollary we obtain that any group of the form $G_1 \times G_2 \times G_3$ admits a strongly aperiodic SFT whenever all the G_i are finitely generated and have decidable word problem. In particular, we show how this theorem implies the existence of strongly aperiodic SFTs in the Grigorchuk group.

Type :	:	contribution soumise
Thématiques	:	SDA*
PDF version	:	PDF version

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Computer science problems:

- (random) generation

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Computer science problems:

- (random) generation
- decision problem

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Computer science problems:

- (random) generation
- decision problem
  $\rightarrow$ order, conjugacy, Engel, knapsack

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Computer science problems:

- (random) generation
- decision problem
  $\rightarrow$ order, conjugacy, Engel, knapsack
- prediction of properties via automaton structure

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Computer science problems:

- (random) generation
- decision problem
  $\rightarrow$ order, conjugacy, Engel, knapsack
- prediction of properties via automaton structure
  $\rightarrow$ patern prohibiting infinite Burnside groups

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Computer science problems:

- (random) generation
- decision problem
  $\rightarrow$ order, conjugacy, Engel, knapsack
- prediction of properties via automaton structure
  $\rightarrow$ patern prohibiting infinite Burnside groups
- dynamical aspects of the action

Linking Mealy automata and computer science?

"Easy" to handle:

- finitely generated
- automata tools (product, dualisation, ...)
- word problem decidable (by minimisation)

Computer science problems:

- (random) generation
- decision problem
  $\rightarrow$ order, conjugacy, Engel, knapsack
- prediction of properties via automaton structure
  $\rightarrow$ patern prohibiting infinite Burnside groups
- dynamical aspects of the action

Stabilisers

Stabilisers

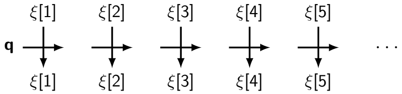
Stabilisers

$\mathsf{Stab}_{\langle \mathcal{A} \rangle}(\xi) = \{g \in \langle \mathcal{A} \rangle \mid \rho_g(\xi) = \xi\}$

Stabilisers

Stabilisers

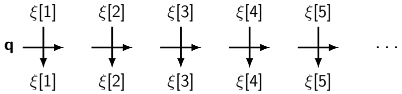
$\mathsf{Stab}_{\langle \mathcal{A} \rangle}(\xi) = \{g \in \langle \mathcal{A} \rangle \mid \rho_g(\xi) = \xi\}$



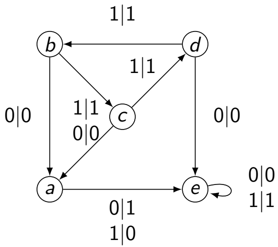
Stabilisers

Stabilisers

$\mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi) = \{g \in \langle \mathcal{A} \rangle \mid \rho_g(\xi) = \xi\}$



Example

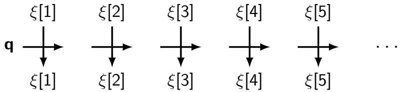


$e, b, c, d \in \mathrm{Stab}_{\langle \mathcal{G} \rangle}(1^\omega)$
studied by Y. Vorobets

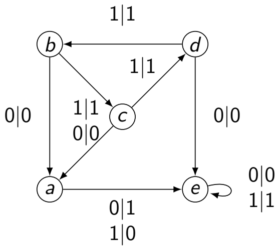
Stabilisers

Stabilisers

$\mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi) = \{g \in \langle \mathcal{A} \rangle \mid \rho_g(\xi) = \xi\}$

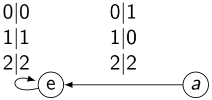


Example



e, b, c, $d \in \mathrm{Stab}_{\langle \mathcal{G} \rangle}(1^\omega)$
studied by Y. Vorobets

Interesting elements

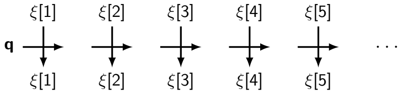


2^ω is stabilised by a

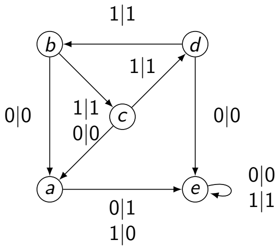
Stabilisers

Stabilisers

$\mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi) = \{g \in \langle \mathcal{A} \rangle \mid \rho_g(\xi) = \xi\}$

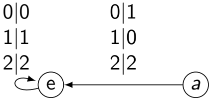


Example



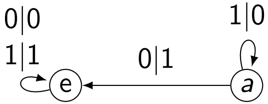
e, $b, c, d \in \mathrm{Stab}_{\langle \mathcal{G} \rangle}(1^\omega)$
studied by Y. Vorobets

Interesting elements

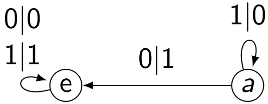


2^ω is stabilised by a
Avoid ending in e

Action on a regular rooted tree

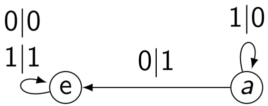


Action on a regular rooted tree



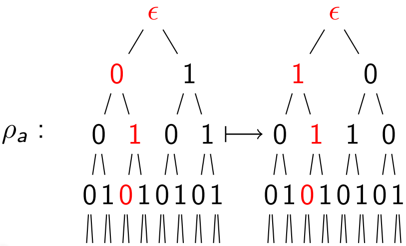
Set of words $\Sigma^* \simeq$ regular rooted tree T

Action on a regular rooted tree

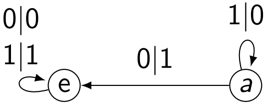


Set of words $\Sigma^* \simeq$ regular rooted tree T

Action

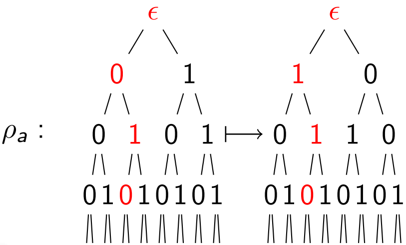


Action on a regular rooted tree

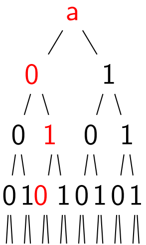


Set of words $\Sigma^* \simeq$ regular rooted tree T

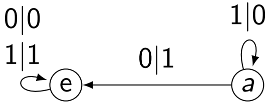
Action



Action

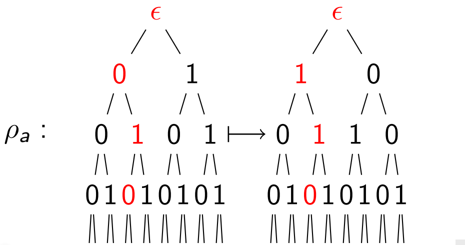


Action on a regular rooted tree

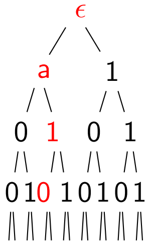


Set of words $\Sigma^* \simeq$ regular rooted tree T

Action

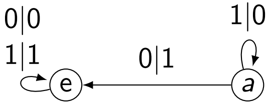


Action



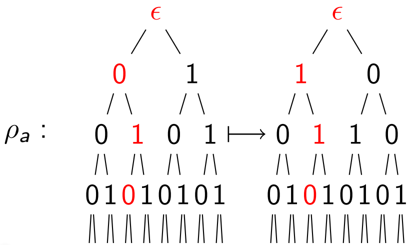
$\rho_a(0) = 1,\ \delta_0(a) = \mathrm{e}$

Action on a regular rooted tree

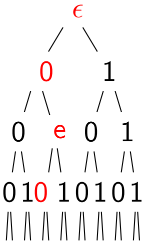


Set of words $\Sigma^* \simeq$ regular rooted tree T

Action

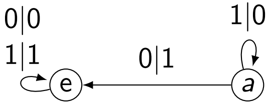


Action



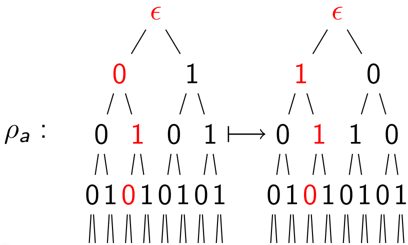
$\rho_{\mathrm{e}}(1) = 1,\ \delta_1(\mathrm{e}) = \mathrm{e}$

Action on a regular rooted tree

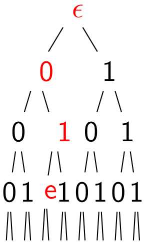


Set of words $\Sigma^* \simeq$ regular rooted tree T

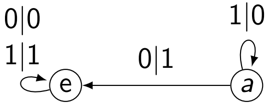
Action



Action

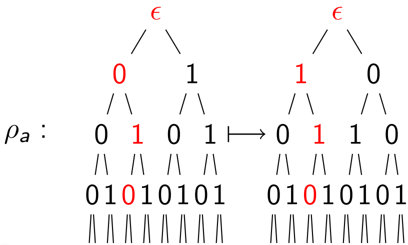


Action on a regular rooted tree

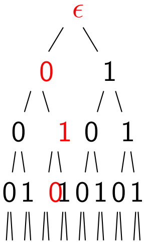


Set of words $\Sigma^* \simeq$ regular rooted tree T

Action



Action



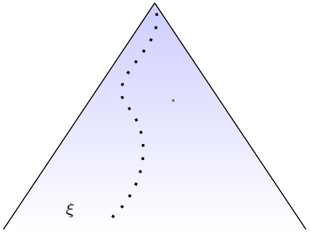
Singular points

$$St : \xi \mapsto \mathsf{Stab}_{\langle \mathcal{A} \rangle}(\xi)$$

Singular points

$$St : \xi \mapsto \mathsf{Stab}_{\langle \mathcal{A} \rangle}(\xi)$$

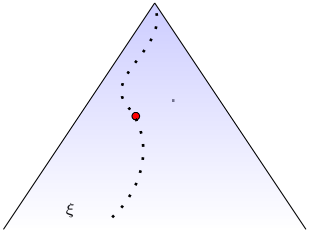
Neighbourhood



Singular points

$$St : \xi \mapsto \mathsf{Stab}_{\langle \mathcal{A} \rangle}(\xi)$$

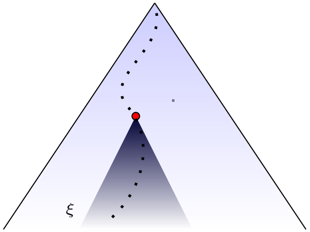
Neighbourhood



Singular points

$$St : \xi \mapsto \mathsf{Stab}_{\langle\mathcal{A}\rangle}(\xi)$$

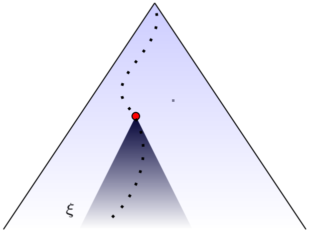
Neighbourhood



Singular points

$$St : \xi \mapsto \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi)$$

Neighbourhood



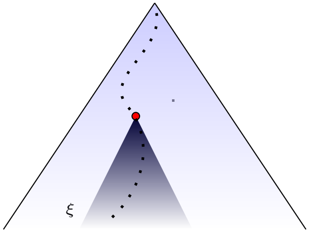
Definition

ξ singular $\Leftrightarrow$ St is not continuous in ξ

Singular points

$$St : \xi \mapsto \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi)$$

Neighbourhood



Definition

ξ singular $\Leftrightarrow$ St is not continuous in ξ

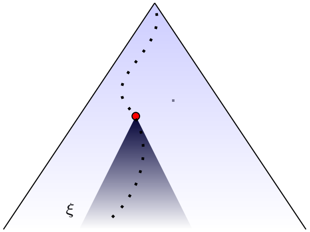
Lemma

ξ not singular $\iff$
$\forall g \in \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi),\ \exists n,\ \delta_{\xi[:n]}(g) = \mathrm{e}$

Singular points

$$St : \xi \mapsto \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi)$$

Neighbourhood



Definition

ξ singular $\Leftrightarrow$ St is not continuous in ξ

Lemma

ξ not singular $\iff$
$\forall g \in \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\xi)$, $\exists n$, $\delta_{\xi[:n]}(g) = \mathrm{e}$

Theorem

The set of singular points has measure 0.

Characterising singular points

Theorem

The set of singular points has measure 0.

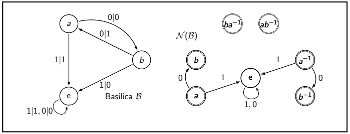
Characterising singular points

Theorem

The set of singular points has measure 0.

Expressing Sing as a language:

(fractal) contracting automata



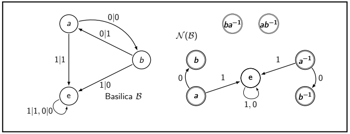
Characterising singular points

Theorem

The set of singular points has measure 0.

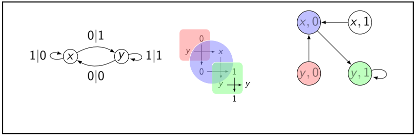
Expressing Sing as a language:

(fractal) contracting automata



Consider specific stabilisers, via commuting pairs:

(bi)reversible automata



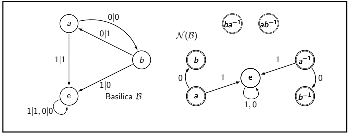
Characterising singular points

Theorem

The set of singular points has measure 0.

Expressing Sing as a language:

(fractal) contracting automata



Contracting Automata

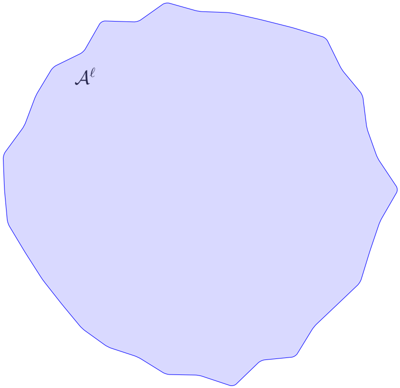
Definition

$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$

Contracting Automata

Definition

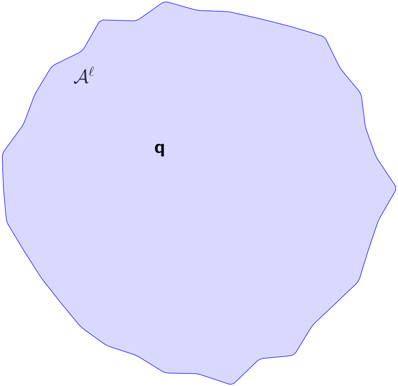
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



Contracting Automata

Definition

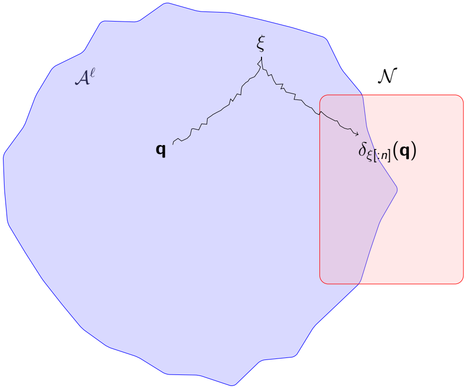
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



Contracting Automata

Definition

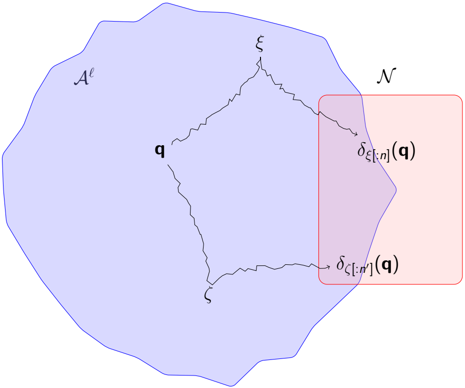
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



Contracting Automata

Definition

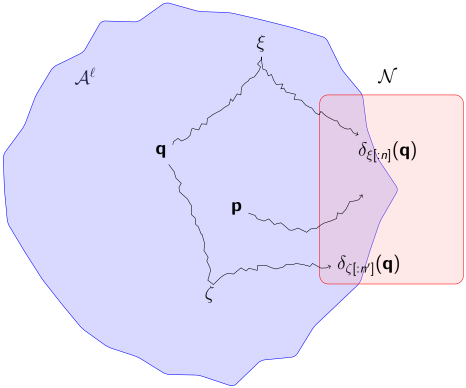
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



Contracting Automata

Definition

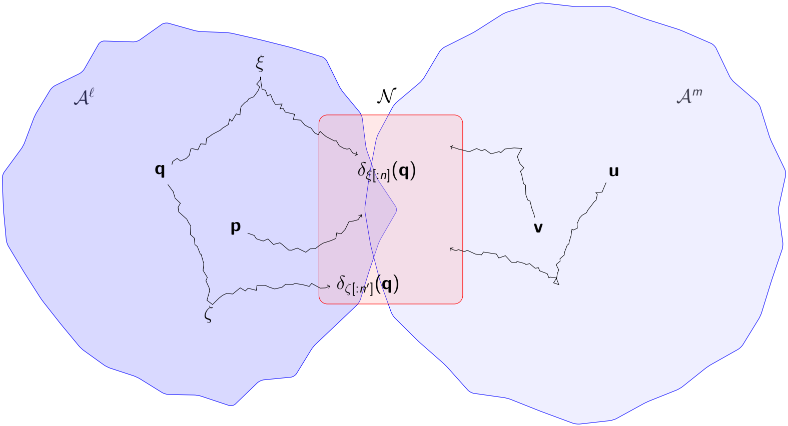
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A}), \forall \mathbf{q}, \forall \xi, \exists n, \ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



Contracting Automata

Definition

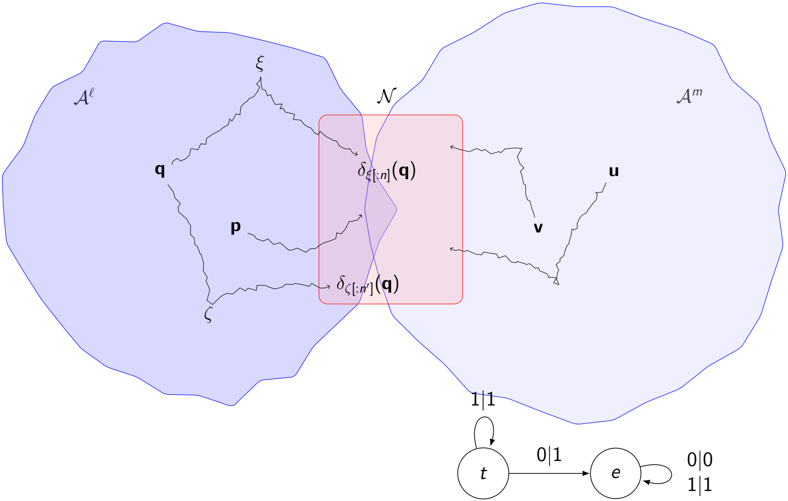
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



Contracting Automata

Definition

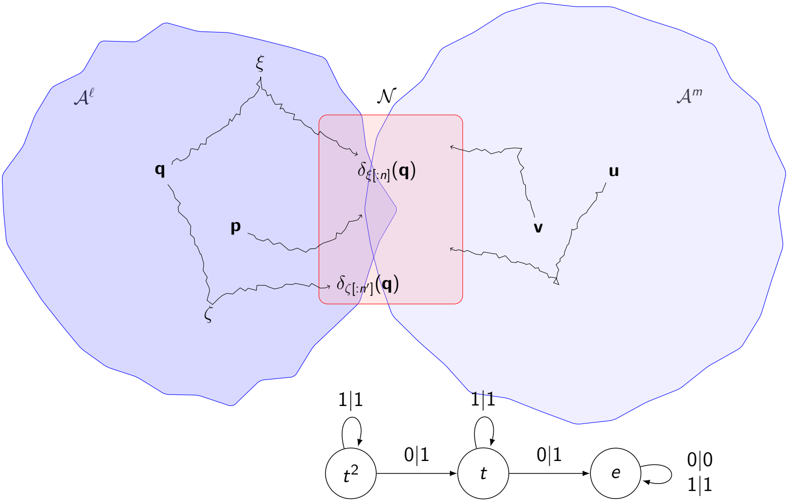
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



Contracting Automata

Definition

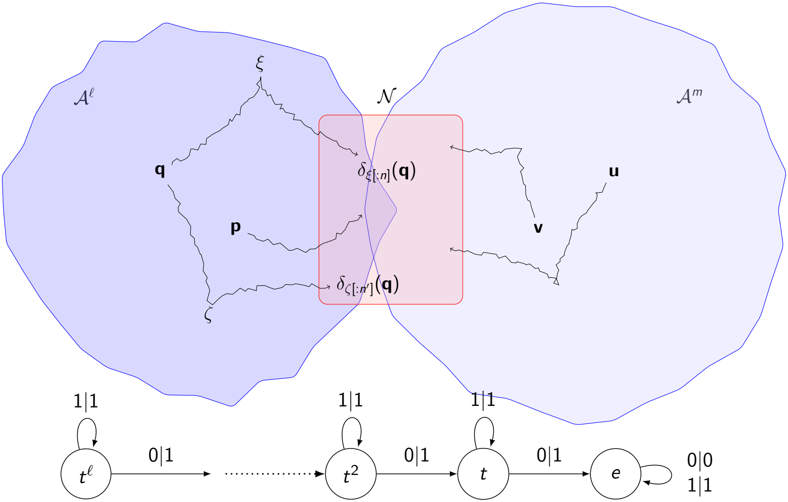
$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$



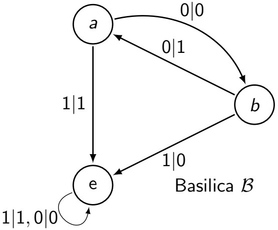
Contracting Automata

Definition

$\mathcal{A}$ contracting $\iff \exists$ finite $\mathcal{N}(\mathcal{A})$, $\forall \mathbf{q}, \forall \xi, \exists n,\ \delta_{\xi[:n]}(\mathbf{q}) \in \mathcal{N}(\mathcal{A})$

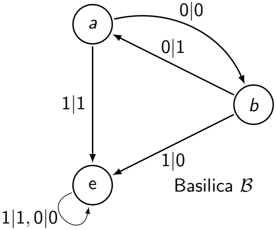


Contracting Automata and singular points



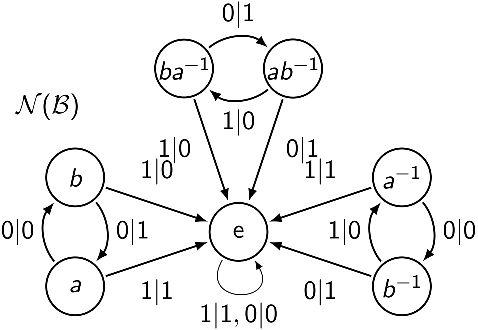
Basilica $\mathcal{B}$

Contracting Automata and singular points

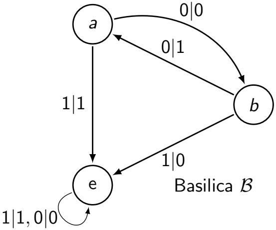


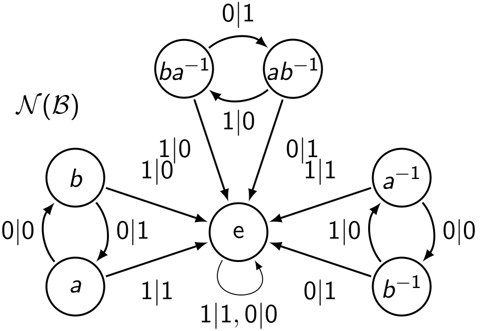
Basilica $\mathcal{B}$

$\mathcal{N}(\mathcal{B})$



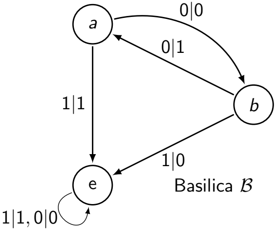
Contracting Automata and singular points





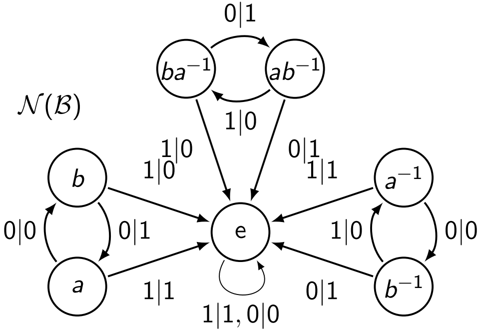
$\delta_{0^\omega}(aba)$

Contracting Automata and singular points



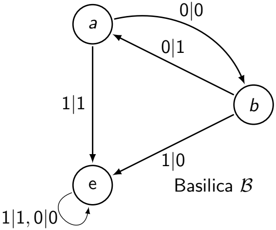
Basilica $\mathcal{B}$

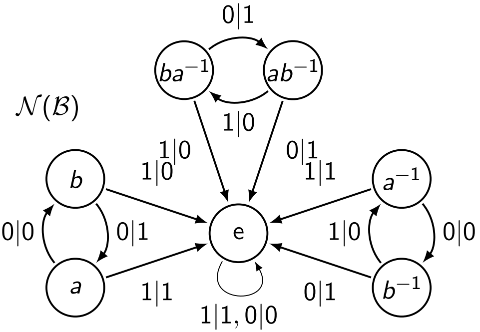
$\mathcal{N}(\mathcal{B})$



$\delta_{0^\omega}(aba) : \delta_0(aba) = ba$

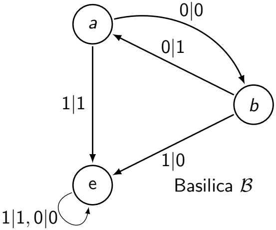
Contracting Automata and singular points

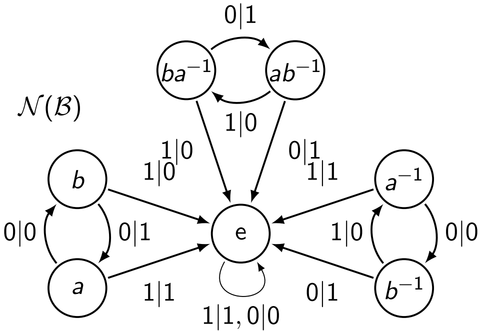




$\delta_{0^\omega}(aba) : \delta_0(aba) = ba; \delta_0(ba) = a$

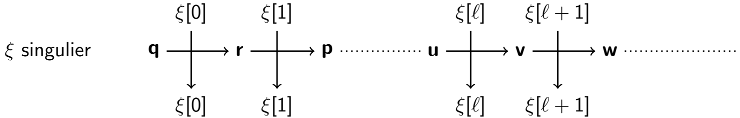
Contracting Automata and singular points

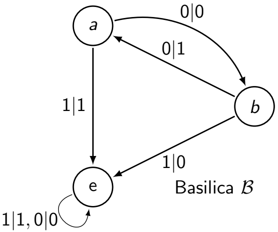


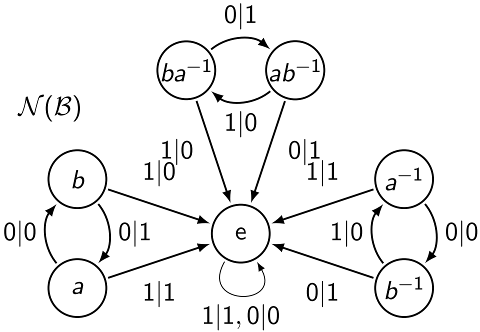


$\delta_{0^\omega}(aba) : \delta_0(aba) = ba; \delta_0(ba) = a \in \mathcal{N}(\mathcal{B})$

Contracting Automata and singular points

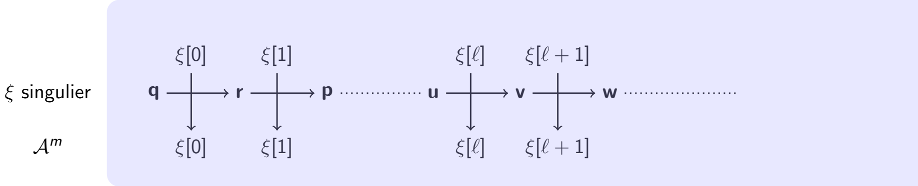


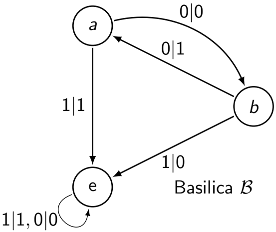




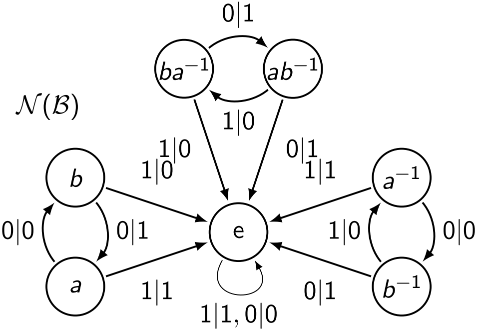
$$\delta_{0^\omega}(aba) : \delta_0(aba) = ba; \delta_0(ba) = a \in \mathcal{N}(\mathcal{B})$$

Contracting Automata and singular points



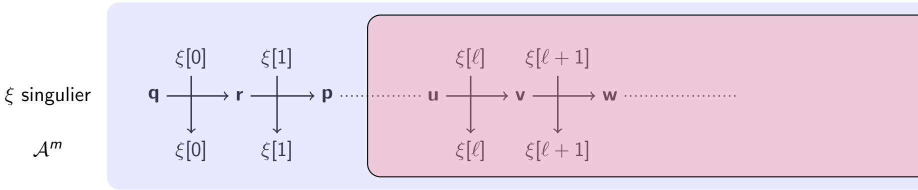


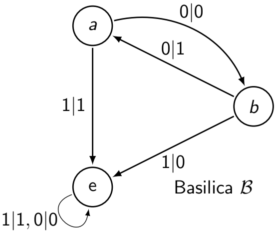
Basilica $\mathcal{B}$

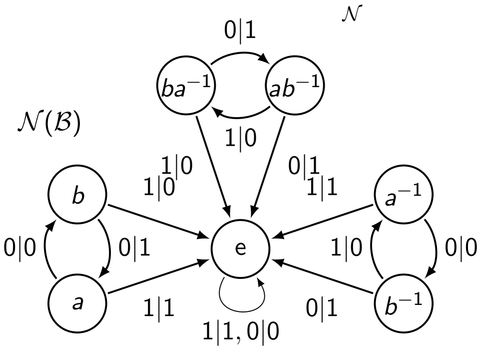


$$\delta_{0^\omega}(aba) : \delta_0(aba) = ba; \delta_0(ba) = a \in \mathcal{N}(\mathcal{B})$$

Contracting Automata and singular points

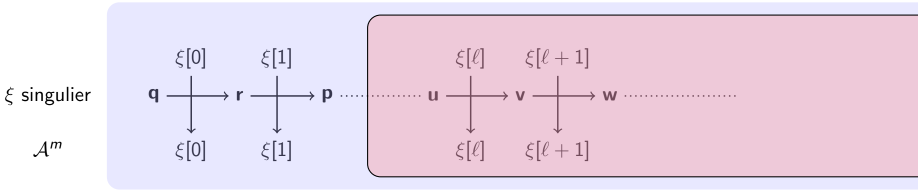


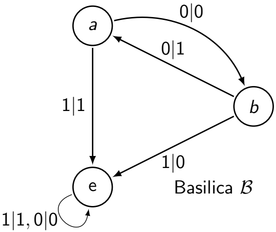


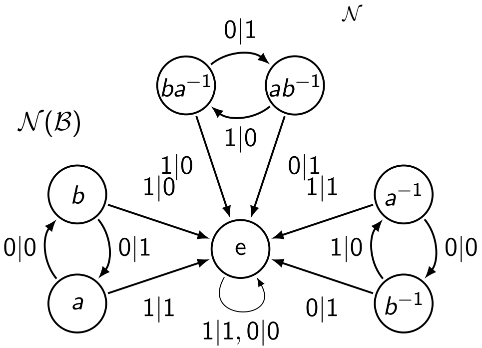


$\delta_{0^\omega}(aba) : \delta_0(aba) = ba; \delta_0(ba) = a \in \mathcal{N}(\mathcal{B})$

Contracting Automata and singular points

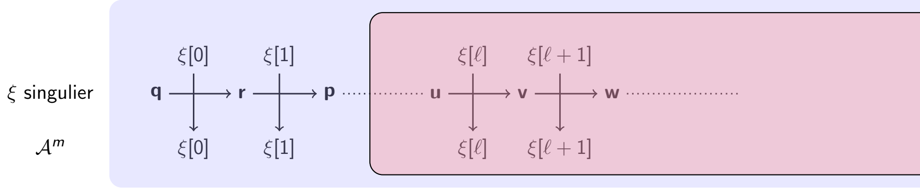


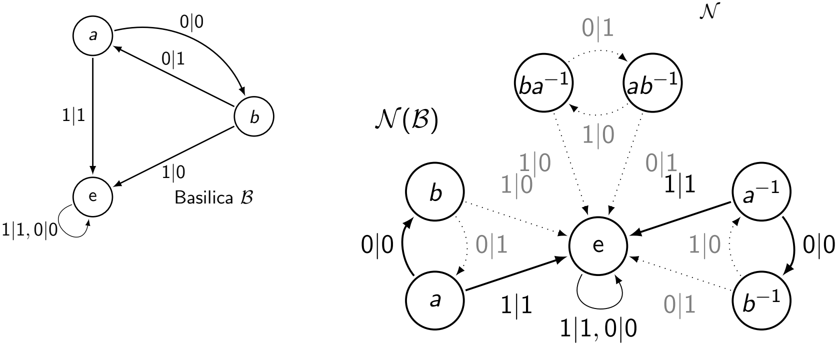




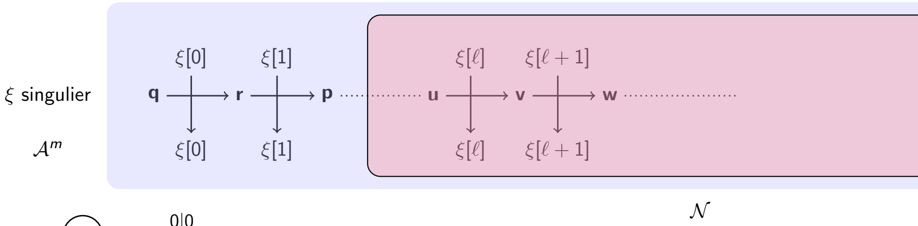
$$\delta_{0^\omega}(aba) : \delta_0(aba) = ba; \delta_0(ba) = a \in \mathcal{N}(\mathcal{B})$$

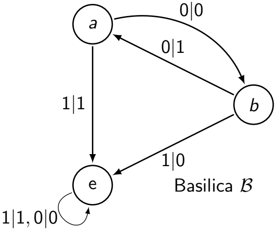
Contracting Automata and singular points





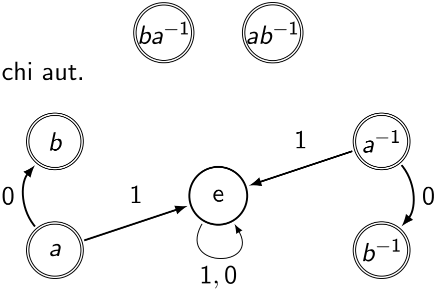
Contracting Automata and singular points





Basilica $\mathcal{B}$

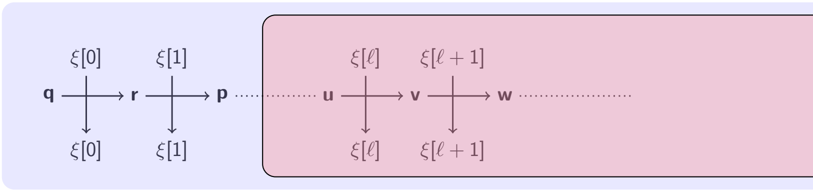
Büchi aut.

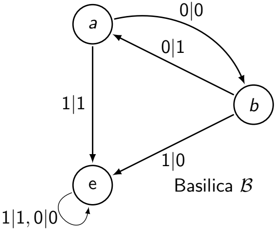


Contracting Automata and singular points

ξ singulier

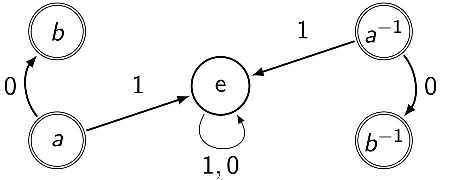
$\mathcal{A}^m$





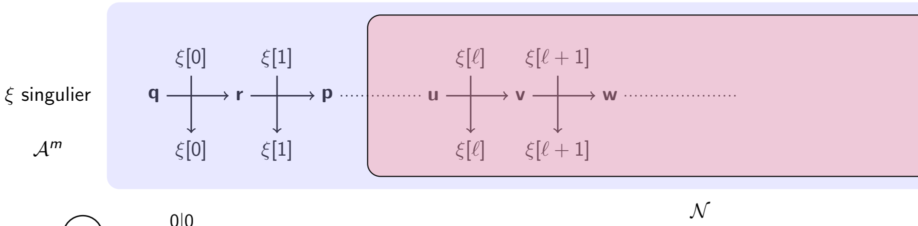
$\mathcal{N}$

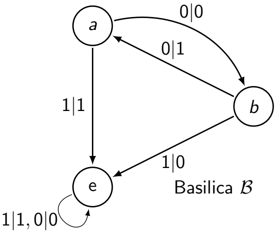
Büchi aut.



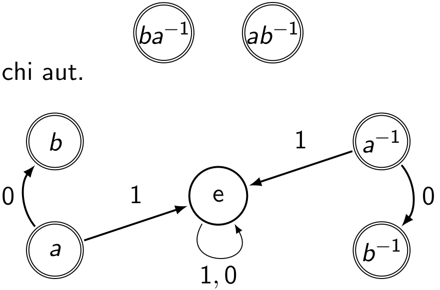
$\delta_{0^\omega}(aba) : \delta_0(aba) = ba; \delta_0(ba) = a \in \mathcal{N}(\mathcal{B})$ no path from a avoids e

Contracting Automata and singular points





Büchi aut.



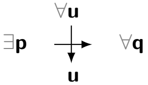
Lemma

$\mathrm{Sing}(\mathcal{B}) = \emptyset$

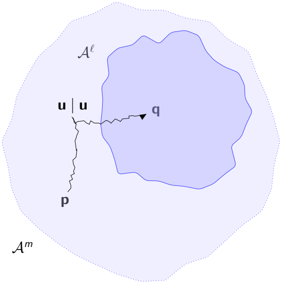
Fractal

Definition

$\mathcal{A}$ fractal $\iff \forall \mathbf{q}, \forall \mathbf{u}, \exists \mathbf{p} \in \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\mathbf{u}),\ \delta_u(\mathbf{p}) = \mathbf{q}$

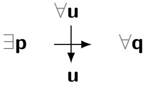


Fractal

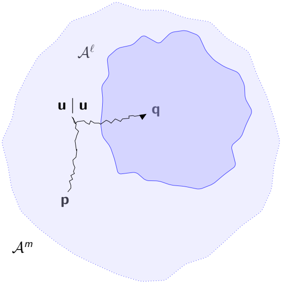


Definition

$\mathcal{A}$ fractal $\iff \forall \mathbf{q}, \forall \mathbf{u}, \exists \mathbf{p} \in \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\mathbf{u}),\ \delta_u(\mathbf{p}) = \mathbf{q}$

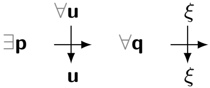


Fractal

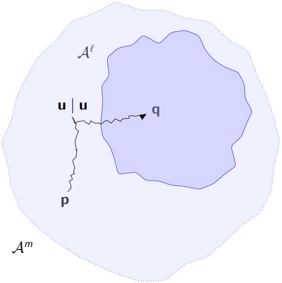


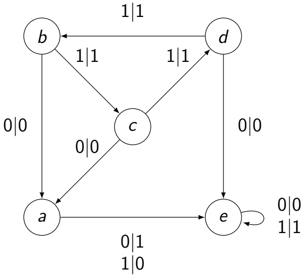
Definition

$$\mathcal{A} \text{ fractal} \iff \forall \mathbf{q}, \forall \mathbf{u}, \exists \mathbf{p} \in \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\mathbf{u}),\ \delta_u(\mathbf{p}) = \mathbf{q}$$



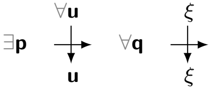
Fractal

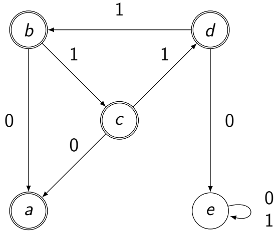




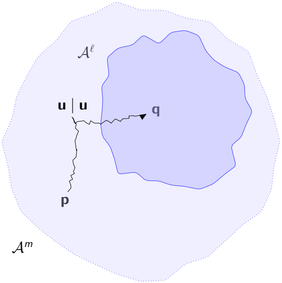
Definition

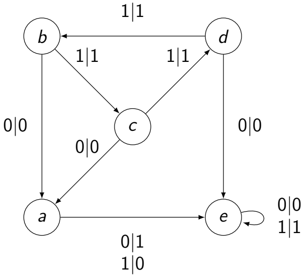
$\mathcal{A}$ fractal $\iff \forall \mathbf{q}, \forall \mathbf{u}, \exists \mathbf{p} \in \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\mathbf{u}),\ \delta_u(\mathbf{p}) = \mathbf{q}$





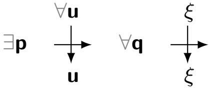
Fractal

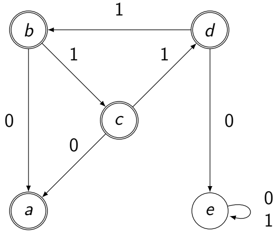




Definition

$\mathcal{A}$ fractal $\iff \forall \mathbf{q}, \forall \mathbf{u}, \exists \mathbf{p} \in \mathrm{Stab}_{\langle \mathcal{A} \rangle}(\mathbf{u}),\ \delta_u(\mathbf{p}) = \mathbf{q}$

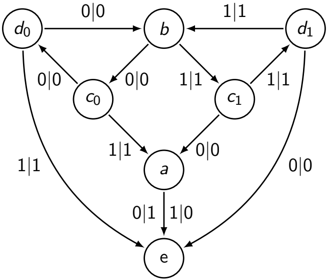




Proposition [Vorobets, AGKPR]

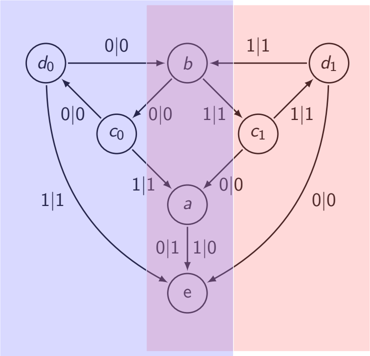
$\mathrm{Sing}(\mathcal{G}) = (0+1)^* 1^\omega$

Singular points



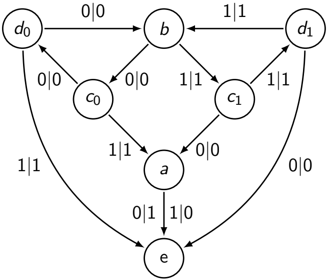
Automaton $\mathcal{W}$

Singular points



Automaton $\mathcal{W}$

Singular points



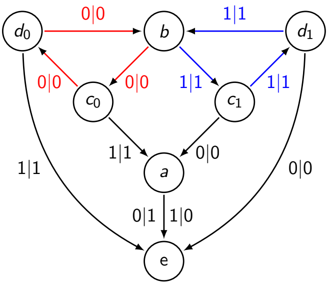
Automaton $\mathcal{W}$

- Contracting

nucleus of size 1027

- Fractal

Singular points



Automaton $\mathcal{W}$

- Contracting

nucleus of size 1027

- Fractal

Corollary

$(000 + 111)^\omega \subset \mathrm{Sing}(\mathcal{W})$

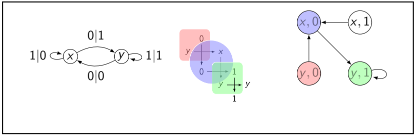
Characterising singular points

Theorem

The set of singular points has measure 0.

Consider specific stabilisers, via commuting pairs:

(bi)reversible automata



Find singular points

Lemma

$\exists\, \xi$ singular $\iff \exists \mathbf{u}^{\omega}$ singular

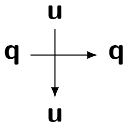
Find singular points

Lemma

$\exists\, \xi$ singular $\iff \exists \mathbf{u}^\omega$ singular

Lemma

$\exists \mathbf{u}, \mathbf{q},\ \delta_{\mathbf{u}}(\mathbf{q}) = \mathbf{q},\ \rho_{\mathbf{q}}(\mathbf{u}) = \mathbf{u}$



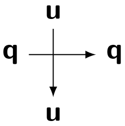
Find singular points

Lemma

$\exists\, \xi$ singular $\iff \exists \mathbf{u}^\omega$ singular

Lemma

$\exists \mathbf{u}, \mathbf{q},\ \delta_{\mathbf{u}}(\mathbf{q}) = \mathbf{q},\ \rho_{\mathbf{q}}(\mathbf{u}) = \mathbf{u}$



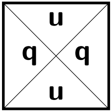
Definition

$(\mathbf{q}, \mathbf{u})$ commuting pair

Find singular points

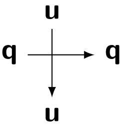
Lemma

$\exists\, \xi$ singular $\iff \exists \mathbf{u}^\omega$ singular



Lemma

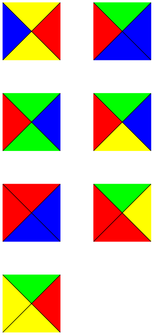
$\exists \mathbf{u}, \mathbf{q},\ \delta_{\mathbf{u}}(\mathbf{q}) = \mathbf{q},\ \rho_{\mathbf{q}}(\mathbf{u}) = \mathbf{u}$



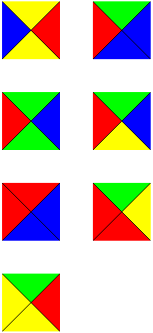
Definition

$(\mathbf{q}, \mathbf{u})$ commuting pair

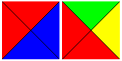
Wang tilings and transducers

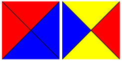


Wang tilings and transducers

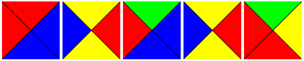


No

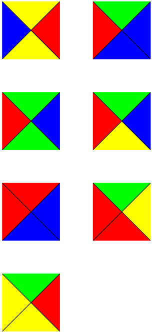




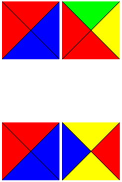
Ok



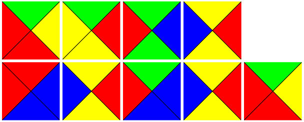
Wang tilings and transducers



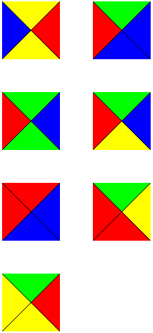
No



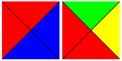
Ok

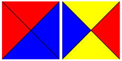


Wang tilings and transducers

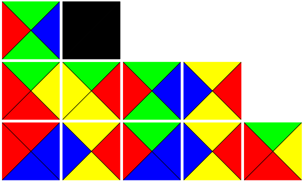


No

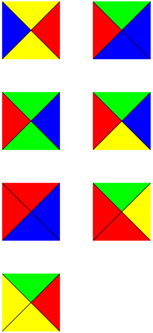


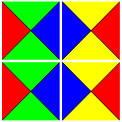


Ok

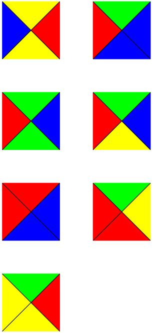


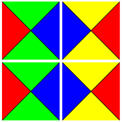
Wang tilings and transducers





Wang tilings and transducers

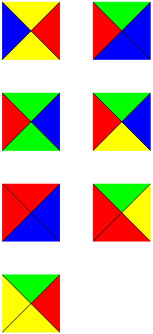


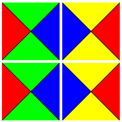


[Berger 1964]

The Domino Problem is undecidable.

Wang tilings and transducers



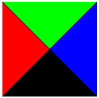


[Berger 1964]

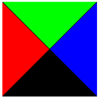
The Domino Problem is undecidable.

Key property: aperiodicity

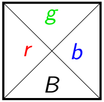
Wang tiling and transducers



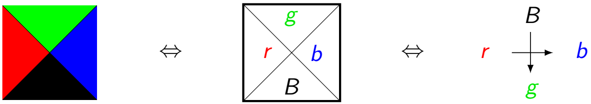
Wang tiling and transducers



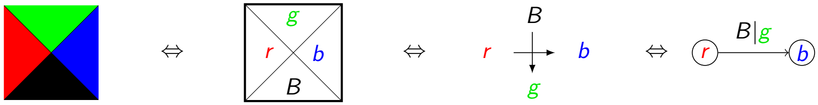
$\Leftrightarrow$



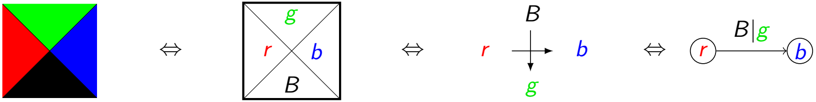
Wang tiling and transducers

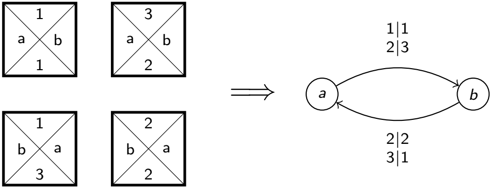


Wang tiling and transducers

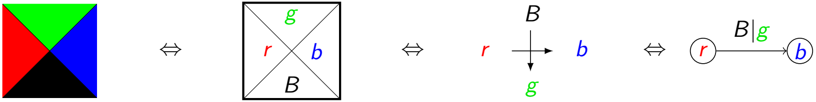


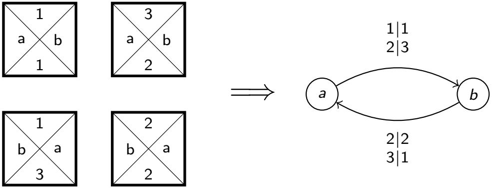
Wang tiling and transducers





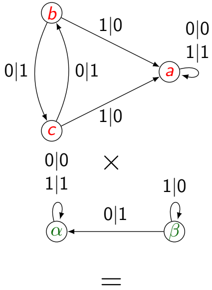
Wang tiling and transducers



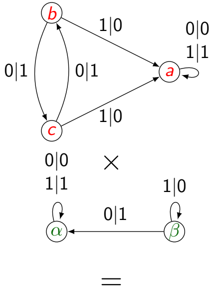


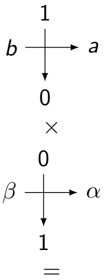
- line of dominoes ⟺ run in the transducer
- multiple lines ⟺ transducer composition

Composition/Product of Transducers

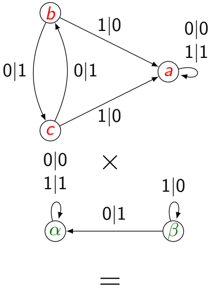


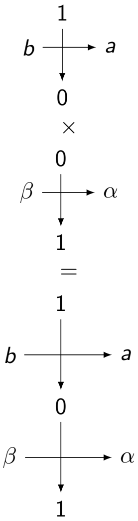
Composition/Product of Transducers



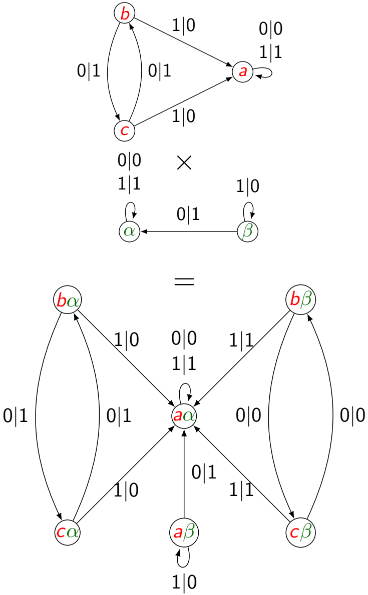


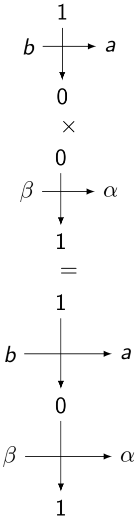
Composition/Product of Transducers



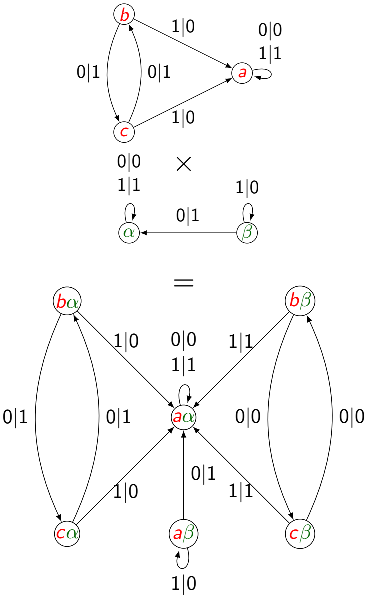


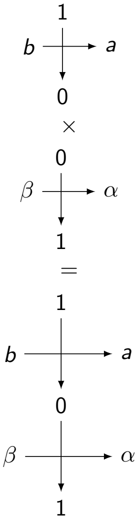
Composition/Product of Transducers

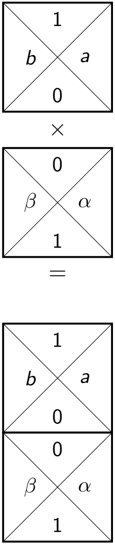




Composition/Product of Transducers



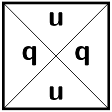




Find singular points

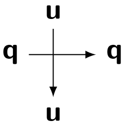
Lemma

$\exists\, \xi$ singular $\iff \exists \mathbf{u}^\omega$ singular



Lemma

$\exists \mathbf{u}, \mathbf{q},\ \delta_{\mathbf{u}}(\mathbf{q}) = \mathbf{q},\ \rho_{\mathbf{q}}(\mathbf{u}) = \mathbf{u}$



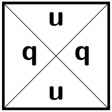
Definition

$(\mathbf{q}, \mathbf{u})$ commuting pair

Find singular points

Lemma

$\exists\, \xi$ singular $\iff \exists \mathbf{u}^\omega$ singular

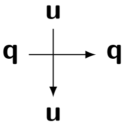


Definition

$(\mathbf{q}, \mathbf{u})$ commuting pair $\iff$ periodic Wang tiling

Lemma

$\exists \mathbf{u}, \mathbf{q},\ \delta_{\mathbf{u}}(\mathbf{q}) = \mathbf{q},\ \rho_{\mathbf{q}}(\mathbf{u}) = \mathbf{u}$



Definition

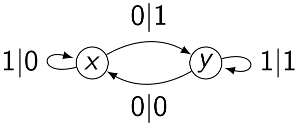
$(\mathbf{q}, \mathbf{u})$ commuting pair

Commuting pair and helix graph

How to find commuting pairs?

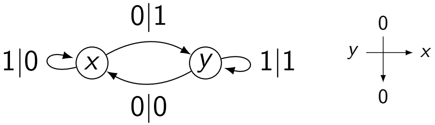
Commuting pair and helix graph

How to find commuting pairs?



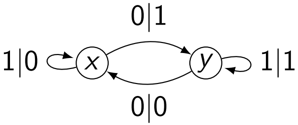
Commuting pair and helix graph

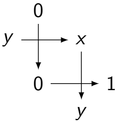
How to find commuting pairs?



Commuting pair and helix graph

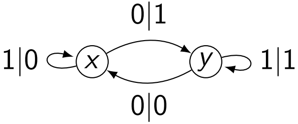
How to find commuting pairs?

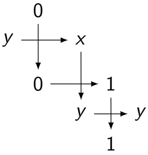




Commuting pair and helix graph

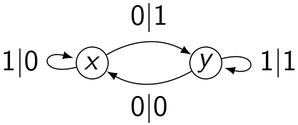
How to find commuting pairs?

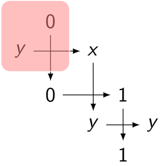




Commuting pair and helix graph

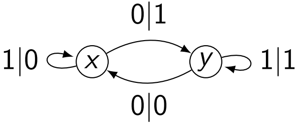
How to find commuting pairs?

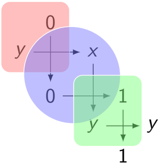


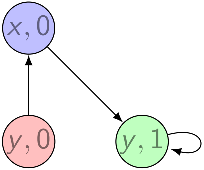


Commuting pair and helix graph

How to find commuting pairs?

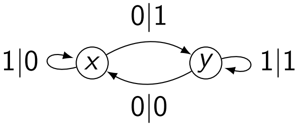


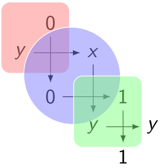


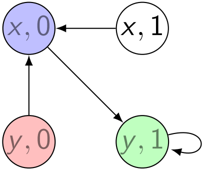


Commuting pair and helix graph

How to find commuting pairs?

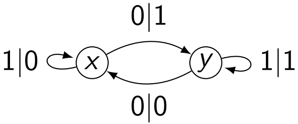


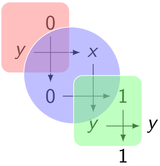


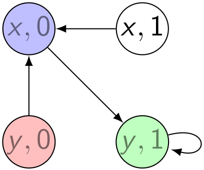


Commuting pair and helix graph

How to find commuting pairs?





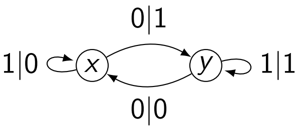


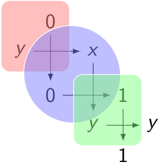
Lemma

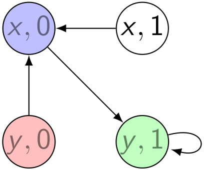
$\exists\, (\mathbf{q}, \mathbf{u})$ commuting pair

Commuting pair and helix graph

How to find commuting pairs?







Lemma

$\exists\,(\mathbf{q}, \mathbf{u})$ commuting pair

Lemma

Tileset associated with Mealy automaton $\Longrightarrow$ periodic tiling

Restricted tilesets and undecidability

$(\mathsf{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

Restricted tilesets and undecidability

$(\mathrm{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

Idea: avoid some states (*e.g.* e)

Restricted tilesets and undecidability

$(\mathrm{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

Idea: avoid some states (*e.g.* e)

`RESTRICTED COMMUTING PAIRS`:

- **Input:** $\mathcal{A} = (Q, \Sigma, \delta, \rho)$, $Q' \subsetneq Q$.
- **Output:** Does $\mathcal{A}$ have commuting pairs restricted to the stateset Q'?

Restricted tilesets and undecidability

$(\mathrm{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

Idea: avoid some states (*e.g.* e)

`RESTRICTED COMMUTING PAIRS`:

- **Input:** $\mathcal{A} = (Q, \Sigma, \delta, \rho)$, $Q' \subsetneq Q$.
- **Output:** Does $\mathcal{A}$ have commuting pairs restricted to the stateset Q'?

`Undecidable`

Restricted tilesets and undecidability

$(\mathrm{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

Idea: avoid some states (*e.g.* e)

RESTRICTED COMMUTING PAIRS:

- **Input:** $\mathcal{A} = (Q, \Sigma, \delta, \rho)$, $Q' \subsetneq Q$.
- **Output:** Does $\mathcal{A}$ have commuting pairs restricted to the stateset Q'?

Undecidable

Reduction to the (4-way deterministic periodic) Domino Problem

Restricted tilesets and undecidability

$(\mathsf{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

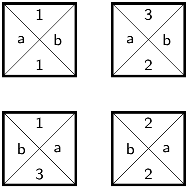
Idea: avoid some states (*e.g.* e)

RESTRICTED COMMUTING PAIRS:

- **Input:** $\mathcal{A} = (Q, \Sigma, \delta, \rho)$, $Q' \subsetneq Q$.
- **Output:** Does $\mathcal{A}$ have commuting pairs restricted to the stateset Q'?

Undecidable

Reduction to the (4-way deterministic periodic) Domino Problem



Restricted tilesets and undecidability

$(\mathrm{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

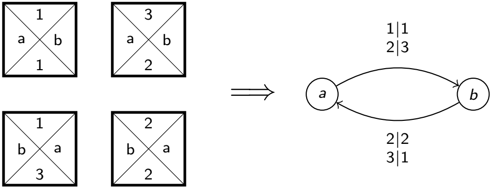
Idea: avoid some states (*e.g.* e)

`RESTRICTED COMMUTING PAIRS`:

- **Input:** $\mathcal{A} = (Q, \Sigma, \delta, \rho)$, $Q' \subsetneq Q$.
- **Output:** Does $\mathcal{A}$ have commuting pairs restricted to the stateset Q'?

Undecidable

Reduction to the (4-way deterministic periodic) Domino Problem



Restricted tilesets and undecidability

$(\mathrm{e}, \mathbf{u})$ is always a commuting pair but $\mathbf{u}^\omega$ is not always singular

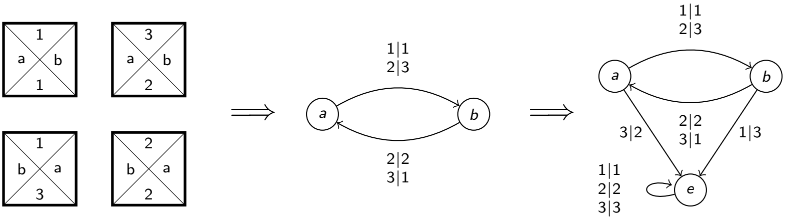
Idea: avoid some states (*e.g.* e)

RESTRICTED COMMUTING PAIRS:

- **Input:** $\mathcal{A} = (Q, \Sigma, \delta, \rho)$, $Q' \subsetneq Q$.
- **Output:** Does $\mathcal{A}$ have commuting pairs restricted to the stateset Q'?

Undecidable

Reduction to the (4-way deterministic periodic) Domino Problem



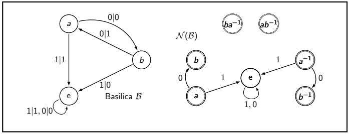
Characterising singular points

Theorem

The set of singular points has measure 0.

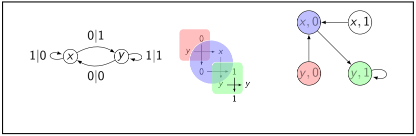
Expressing Sing as a language:

(fractal) contracting automata



Consider specific stabilisers, via commuting pairs:

(bi)reversible automata



Thanks

If you enjoyed that talk

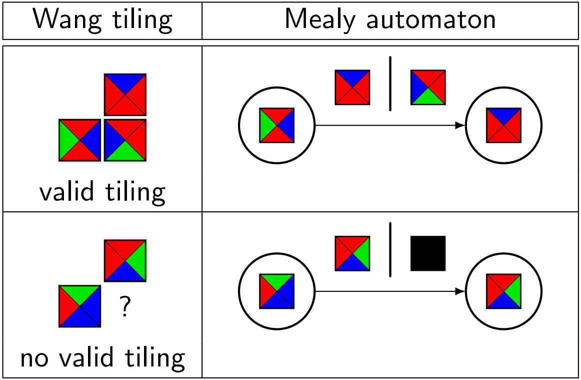
MealyM final event July 10–13

- Nathalie Aubrun
- Laurent Bartholdi
- Tara Brough
- Alan J. Cain
- Daniele D'Angeli
- Martin Delacourt
- Francesca Fiorenzi
- Pierre Gillibert
- Markus Lohrey
- Jean Mairesse
- Irène Marcovici
- Cyril Nicaud
- Nicolas Ollinger
- Emanuele Rodaro
- Dmytro Savchuk
- Reem Yassawi
- Matthieu Picantin's HdR defense
- my PhD defense

https://mealym.sciencesconf.org/

Thanks

Gillibert's connection between Wang tiling and Mealy automata



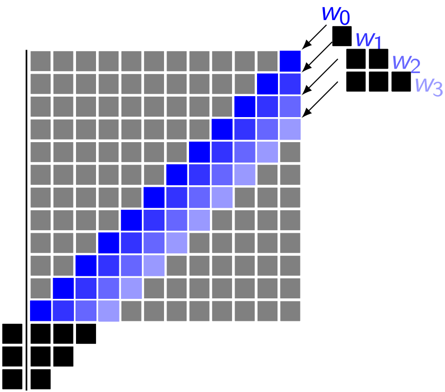
The Mealy automaton $\mathcal{W}_{\mathcal{T}}$ associated with a *wn*-deterministic Wang tileset $\mathcal{T}$ according to Gillibert.

Lemma

If $\mathbb{Z}^2$ admits a valid Wang tiling for $\mathcal{T}$, then $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is infinite.

Lemma

If $\mathbb{Z}^2$ admits a valid Wang tiling for $\mathcal{T}$, then $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is infinite.



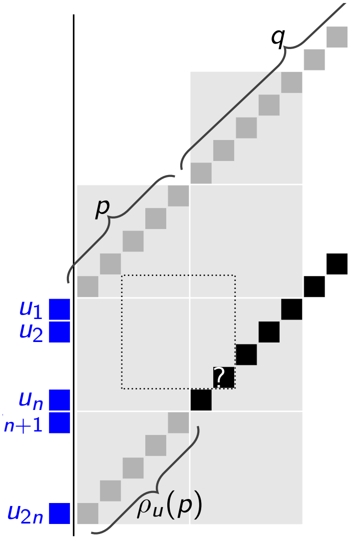
The action of the tile ■ corresponds to an element of infinite order.

Lemma

If $\mathbb{Z}^2$ admits no valid Wang tiling for $\mathcal{T}$, then $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is finite.

Lemma

If $\mathbb{Z}^2$ admits no valid Wang tiling for $\mathcal{T}$, then $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is finite.



If some tile was not ■, the $n \times n$ square) could be tiled.

Theorem [Gillibert '13]

The semigroup $\langle\mathcal{A}(\mathcal{T})\rangle_+$ is infinite if and only if $\mathcal{T}$ can tile the discrete plane $\mathbb{Z}^2$.

Theorem [Gillibert '13]

The semigroup $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is infinite if and only if $\mathcal{T}$ can tile the discrete plane $\mathbb{Z}^2$.

Corollary [Gillibert + Kari '13]

The finiteness problem for automaton semigroups is undecidable.

Theorem [Gillibert '13]

The semigroup $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is infinite if and only if $\mathcal{T}$ can tile the discrete plane $\mathbb{Z}^2$.

Corollary [Gillibert + Kari '13]

The finiteness problem for automaton semigroups is undecidable.

- exists classes where order problem is decidable

Theorem [Gillibert '13]

The semigroup $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is infinite if and only if $\mathcal{T}$ can tile the discrete plane $\mathbb{Z}^2$.

Corollary [Gillibert + Kari '13]

The finiteness problem for automaton semigroups is undecidable.

- exists classes where order problem is decidable
  $\rightarrow$ bounded, just invertible-reversible automata

Theorem [Gillibert '13]

The semigroup $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is infinite if and only if $\mathcal{T}$ can tile the discrete plane $\mathbb{Z}^2$.

Corollary [Gillibert + Kari '13]

The finiteness problem for automaton semigroups is undecidable.

- exists classes where order problem is decidable
  $\rightarrow$ bounded, just invertible-reversible automata
- in some classes, the problem is open

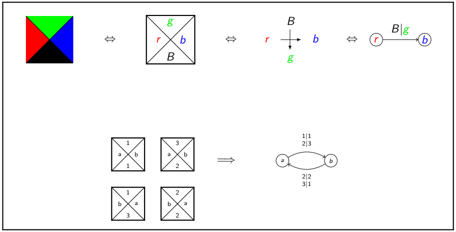
Theorem [Gillibert '13]

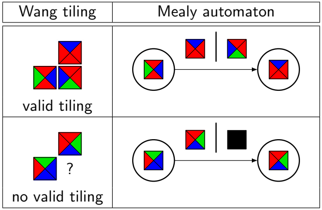
The semigroup $\langle \mathcal{A}(\mathcal{T}) \rangle_+$ is infinite if and only if $\mathcal{T}$ can tile the discrete plane $\mathbb{Z}^2$.

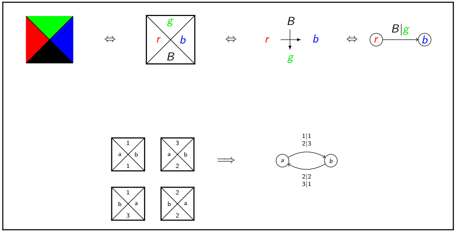
Corollary [Gillibert + Kari '13]

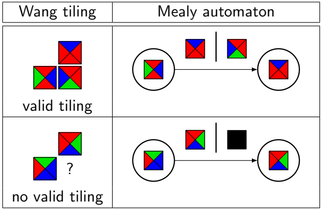
The finiteness problem for automaton semigroups is undecidable.

- exists classes where order problem is decidable
  $\rightarrow$ bounded, just invertible-reversible automata
- in some classes, the problem is open
  $\rightarrow$ bireversible automata

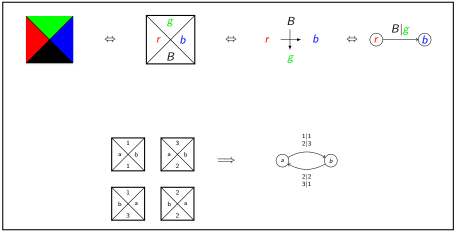

g
r
b
B
⇔
B
r
b
g
⇔
B|g
r
b
1
a
b
1
3
a
b
2
1
b
a
3
2
b
a
2
⟹
1|1
2|3
a
b
2|2
3|1

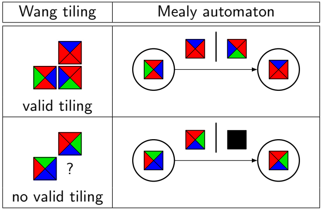

Wang tiling
Mealy automaton
valid tiling
?
no valid tiling





- Explore in depth these connections
- Find eventual new translations
- Specialize to certain subclasses





- Explore in depth these connections
- Find eventual new translations
- Specialize to certain subclasses

Thanks