On guidable index of tree automata

Damian Niwiński, Michał Skrzypczak

MFCS 2021, Tallinn

UNIVERSITY
OF WARSAW

Infinite trees t

Infinite trees t

$$\mathrm{Tr}_{\Sigma} \ni t\colon \{\mathtt{L}, \mathtt{R}\}^{*} \to \Sigma$$

Infinite trees t

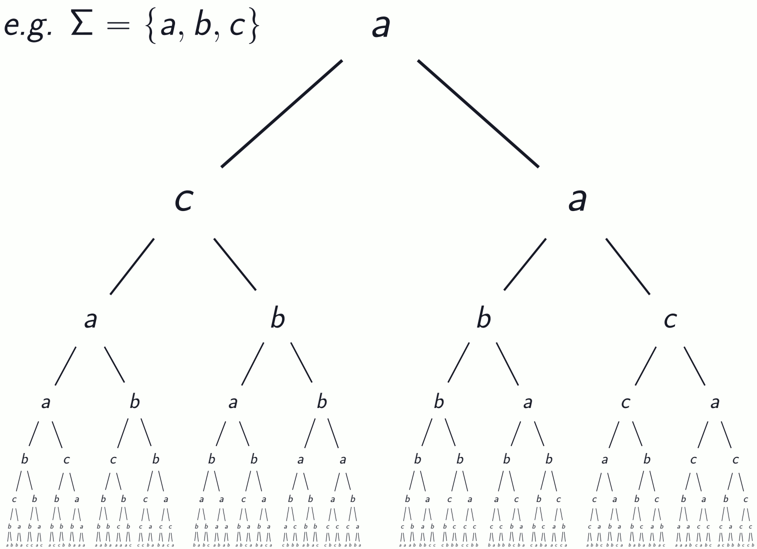
$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L}, \mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$

Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

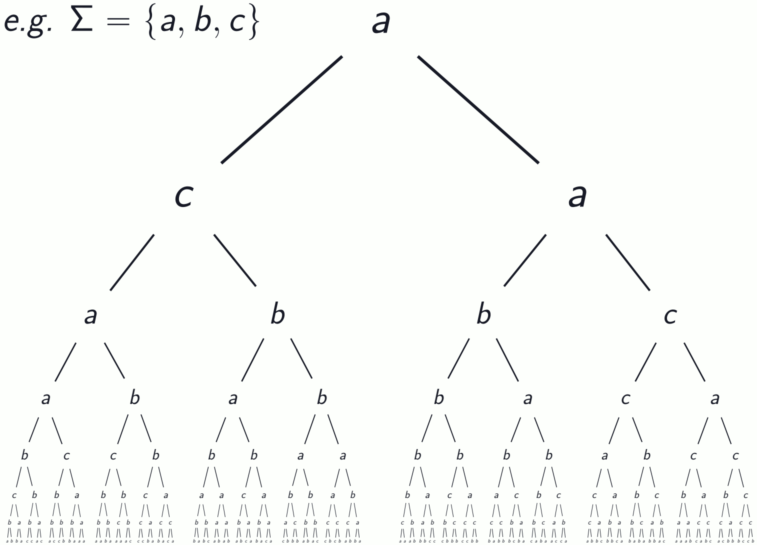
e.g. $\Sigma = \{a, b, c\}$



Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L}, \mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$

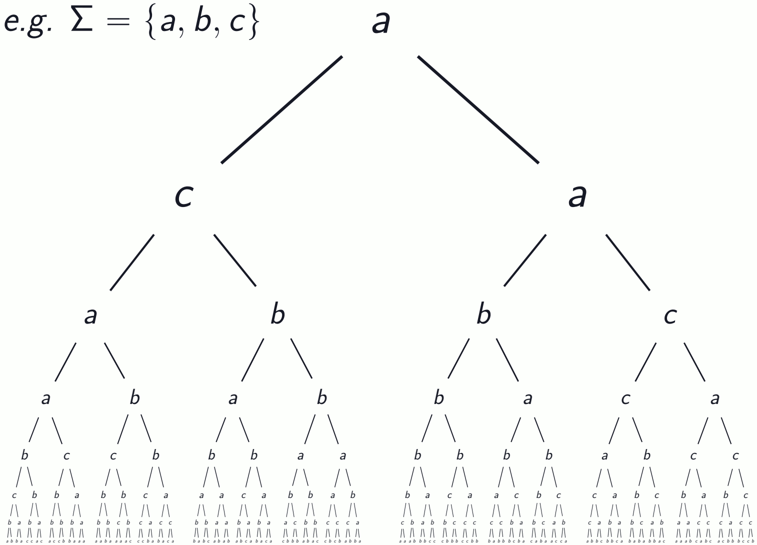


Tree automata $\mathcal{A}$

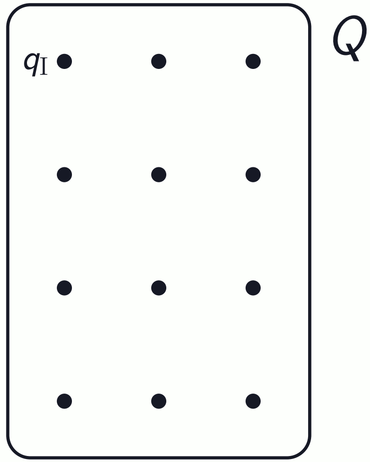
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



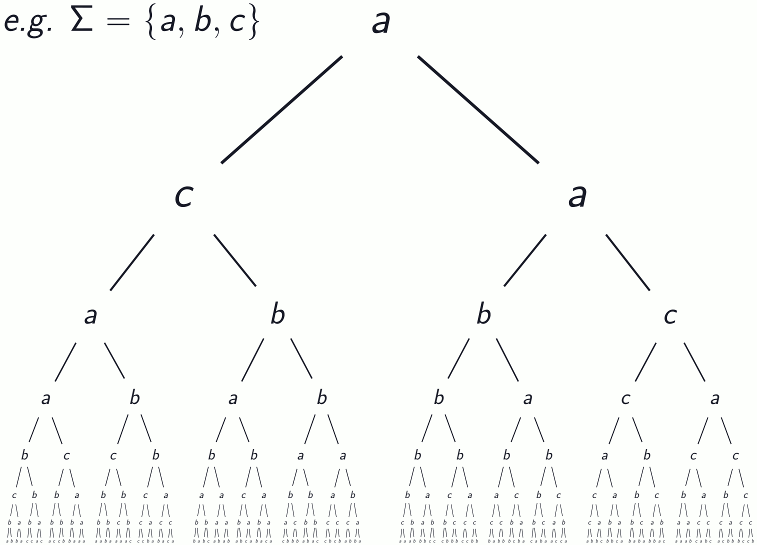
Tree automata $\mathcal{A}$



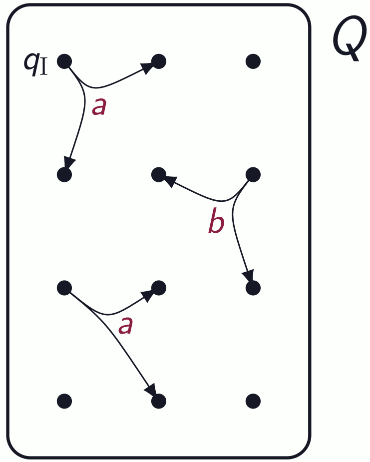
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



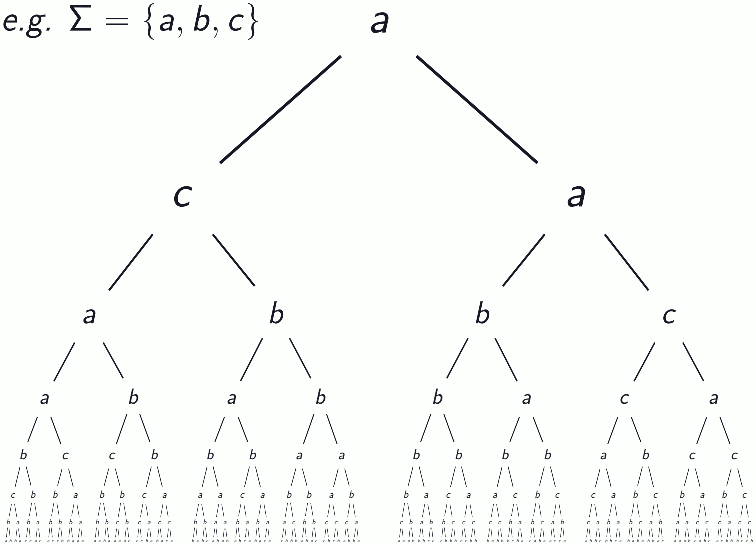
Tree automata $\mathcal{A}$



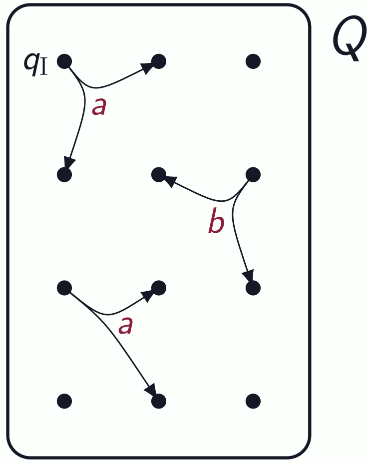
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L}, \mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



Tree automata $\mathcal{A}$

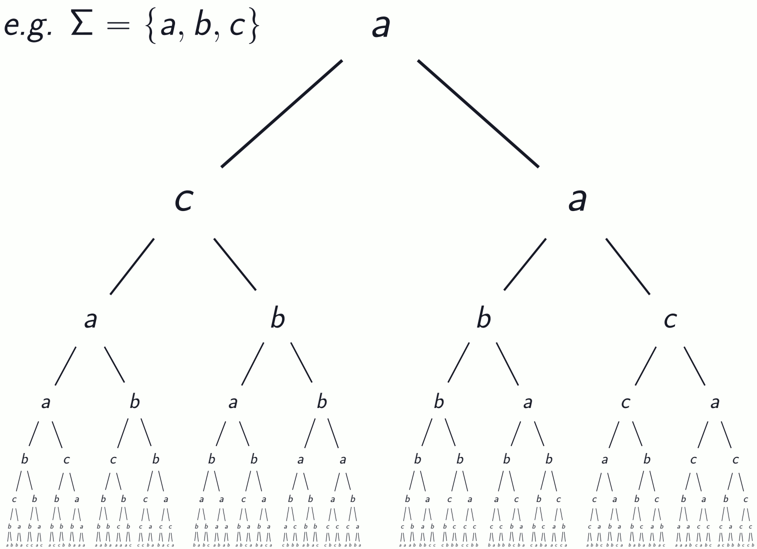


$$\Omega\colon Q \to \{i, \dots, j\} \subseteq \mathbb{N}$$

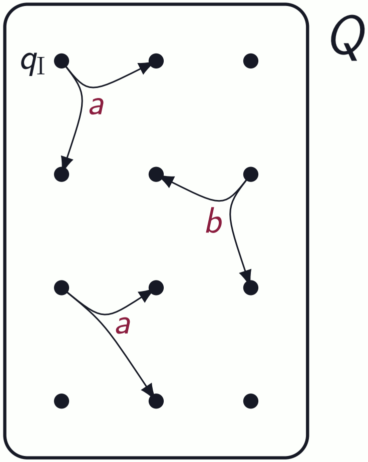
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



Tree automata $\mathcal{A}$

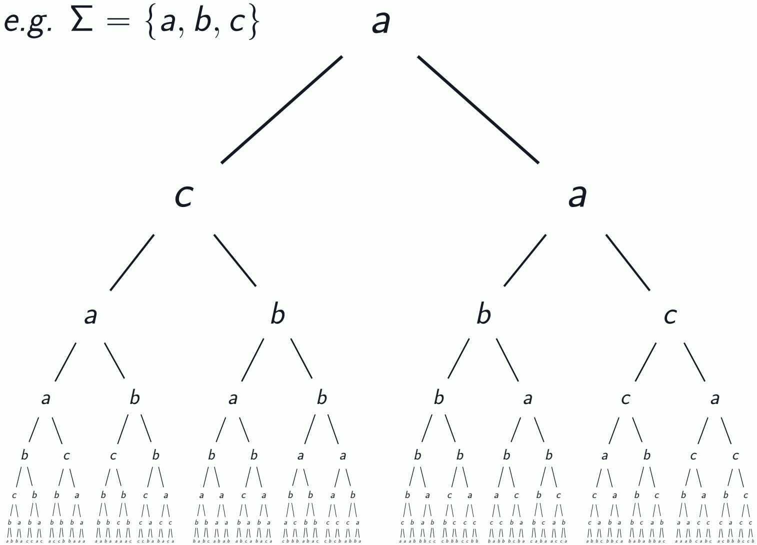


$$\Omega\colon Q \to \underbrace{\{i,\ldots,j\}}_{\text{index}} \subseteq \mathbb{N}$$

Infinite trees t

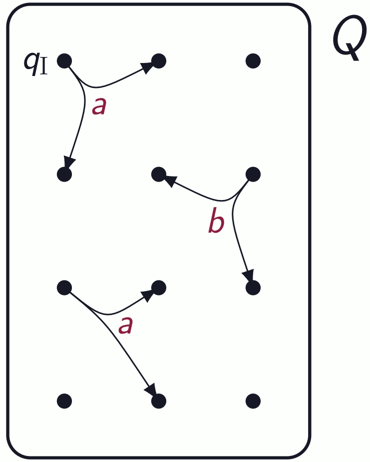
$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid$$

Tree automata $\mathcal{A}$

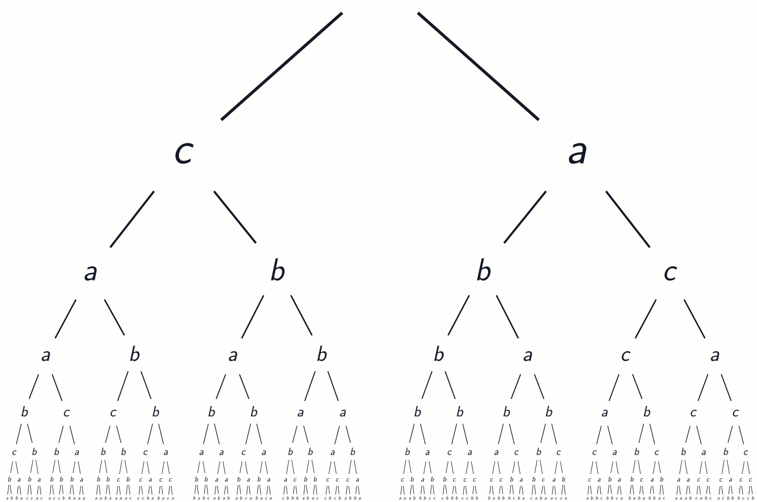


$$\Omega\colon Q \to \underbrace{\{i, \ldots, j\}}_{\text{index}} \subseteq \mathbb{N}$$

Infinite trees t

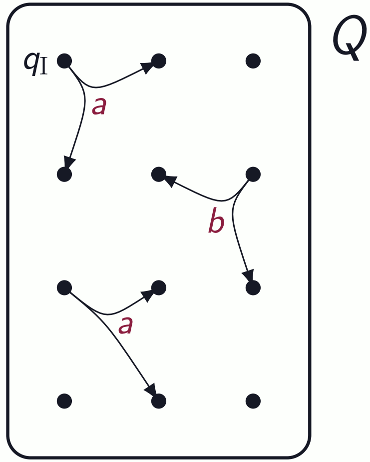
$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L}, \mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}$$

Tree automata $\mathcal{A}$

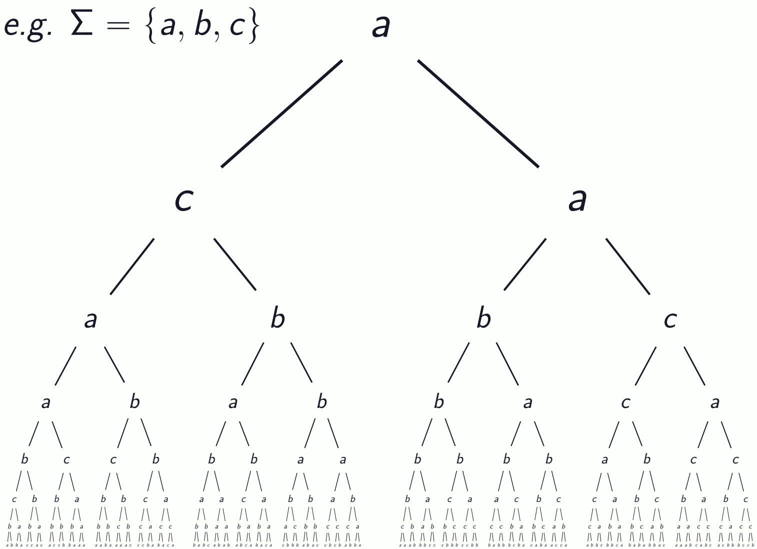


$$\Omega\colon Q \to \underbrace{\{i, \ldots, j\}}_{\text{index}} \subseteq \mathbb{N}$$

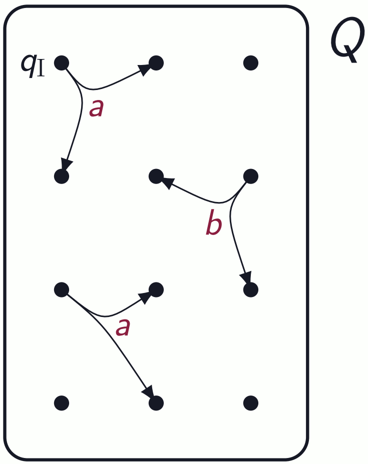
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



Tree automata $\mathcal{A}$



$$\Omega\colon Q \to \underbrace{\{i,\ldots,j\}}_{\text{index}} \subseteq \mathbb{N}$$

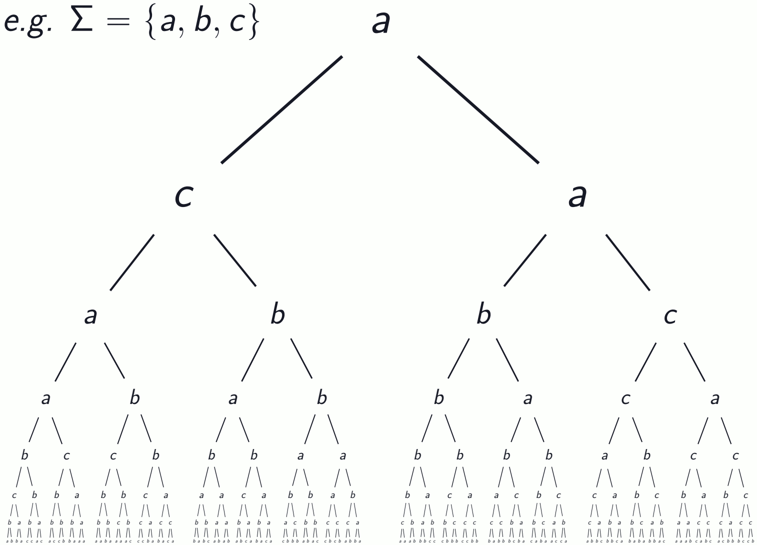
$$\mathrm{L}(\mathcal{A}) \overset{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q.}_{\text{run}}\ \underbrace{\forall \pi \in \{\mathtt{L},\mathtt{R}\}^\omega}_{\text{branch}}$$

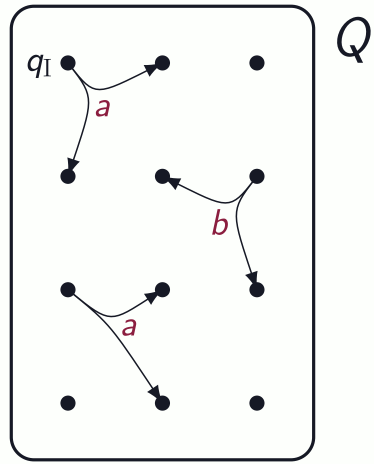
Infinite trees t

Tree automata $\mathcal{A}$

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L}, \mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$





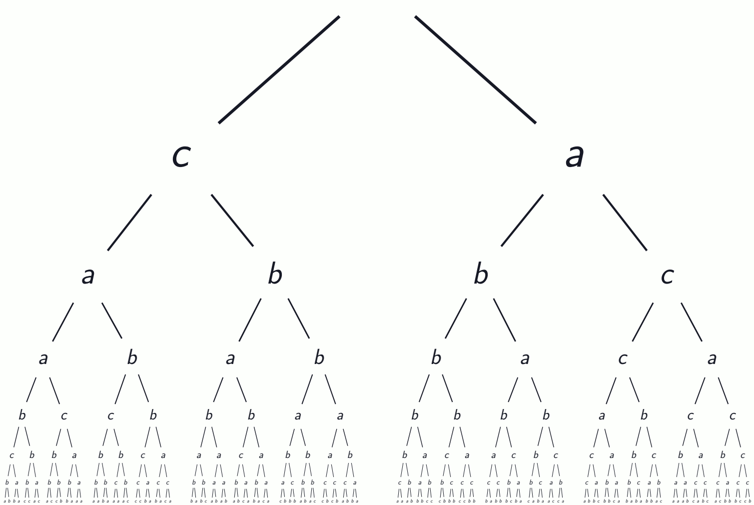
$$\Omega\colon Q \to \underbrace{\{i, \ldots, j\}}_{\text{index}} \subseteq \mathbb{N}$$

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q.}_{\text{run}}\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.}_{\text{branch}} \text{ states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}\}$$

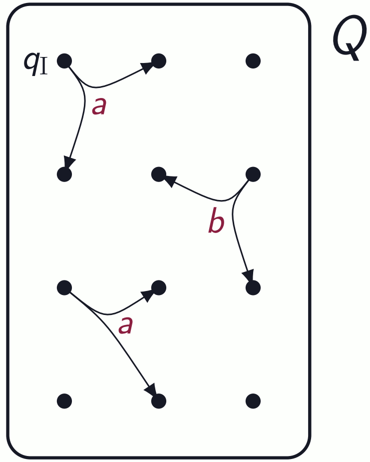
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



Tree automata $\mathcal{A}$



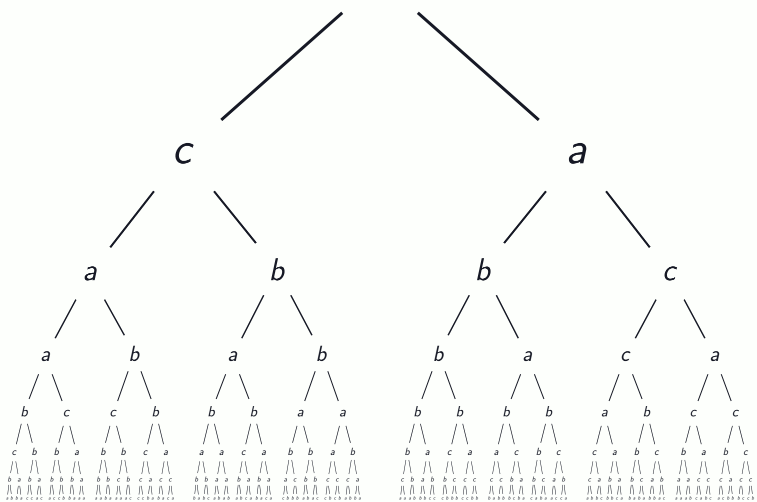
$$\Omega\colon Q \to \underbrace{\{i, \ldots, j\}}_{\text{index}} \subseteq \mathbb{N}$$

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L},\mathtt{R}\}^\omega}_{\text{branch}}.\ \text{states of } \rho \text{ on } \pi \text{ are } \underbrace{\Omega\text{-accepting}}_{\text{parity condition}}\}$$

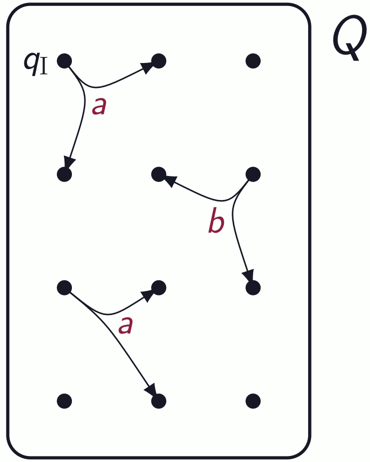
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



Tree automata $\mathcal{A}$



$$\Omega\colon Q \to \underbrace{\{i, \ldots, j\}}_{\text{index}} \subseteq \mathbb{N}$$

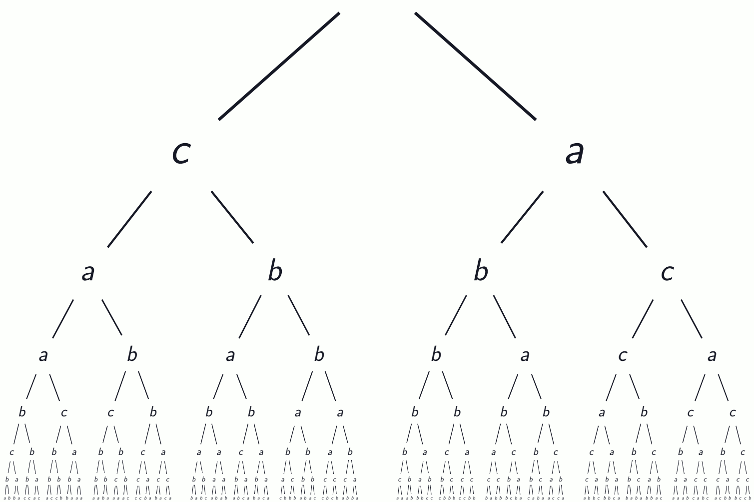
$$\mathrm{L}(\mathcal{A}) \overset{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L},\mathtt{R}\}^\omega}_{\text{branch}}.\ \text{states of } \rho \text{ on } \pi \text{ are } \underbrace{\Omega\text{-accepting}}_{\text{parity condition}}\}$$

⇝ regular tree languages ⇝ model checking of **Monadic Second-Order logic**

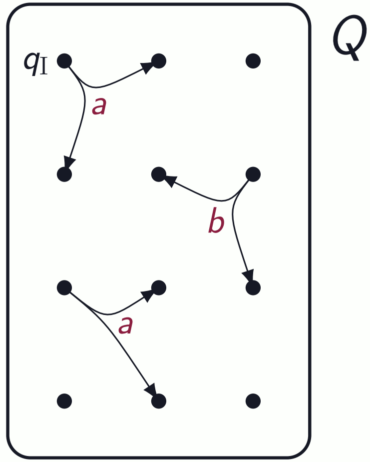
Infinite trees t

$$\mathrm{Tr}_\Sigma \ni t\colon \{\mathtt{L},\mathtt{R}\}^* \to \Sigma$$

e.g. $\Sigma = \{a, b, c\}$



Tree automata $\mathcal{A}$



$$\Omega\colon Q \to \underbrace{\{i, \ldots, j\}}_{\text{index}} \subseteq \mathbb{N}$$

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L},\mathtt{R}\}^\omega}_{\text{branch}}.\ \text{states of } \rho \text{ on } \pi \text{ are } \underbrace{\Omega\text{-accepting}}_{\text{parity condition}}\}$$

⇝ regular tree languages ⇝ model checking of **Monadic Second-Order logic**

Inherent non-determinism

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_{\Sigma} \mid$

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \Big\{ t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}} \Big\}$$

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{ t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}} \}$$

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \overset{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{$$

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{ t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}. \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}} \}$$

$$\mathrm{L}(\mathcal{A}) = \{$$

t_4

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L},\mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{$$

t_4

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \Big\{ t \in \mathrm{Tr}_{\Sigma} \mid \underbrace{\exists \rho \in \mathrm{Tr}_{Q}}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathrm{L}, \mathrm{R}\}^{\omega}.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}} \Big\}$$

$$\mathrm{L}(\mathcal{A}) = \{$$

t_4

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L},\mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

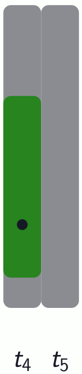
$$\mathrm{L}(\mathcal{A}) = \{\ldots, t_4,$$

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \big\{ t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}. \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega. \text{ states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}} \big\}$$

$$\mathrm{L}(\mathcal{A}) = \{\ldots, t_4,$$

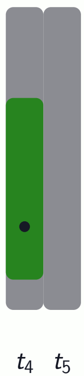


Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathrm{L}, \mathrm{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{\ldots, t_4, \not{t_5},$$

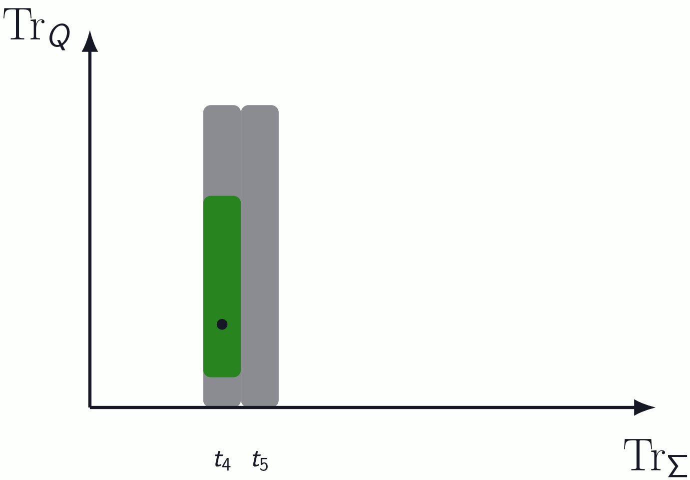


Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$L(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$L(\mathcal{A}) = \{\ldots, t_4, \not{t_5},$$

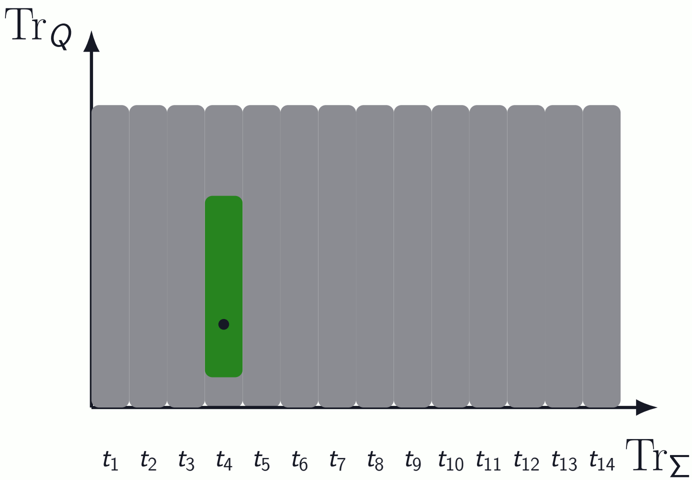


Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \Big\{ t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L},\mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}} \Big\}$$

$$\mathrm{L}(\mathcal{A}) = \{\ldots, t_4, \cancel{t_5}, \ldots\}$$

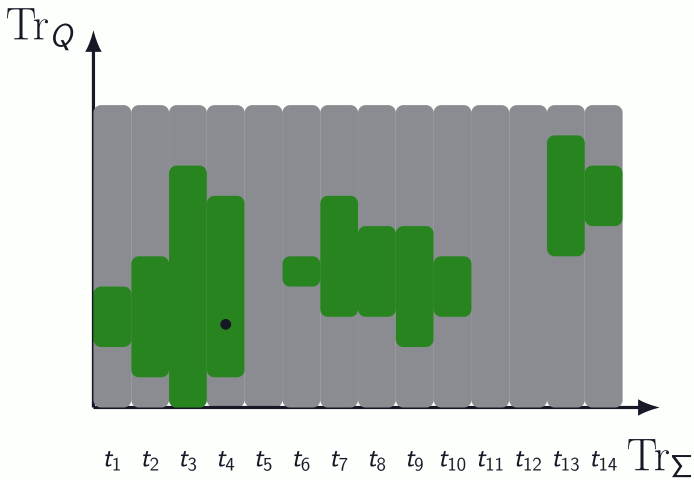


Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}. \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{\dots, t_4, \not{t_5}, \dots\}$$

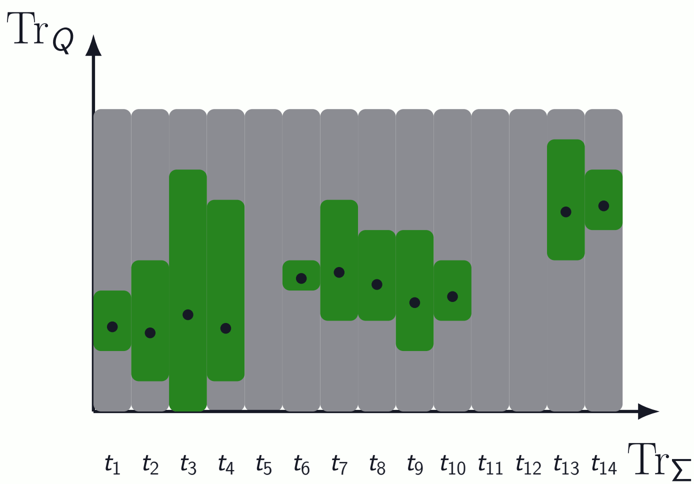


Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{\ldots, t_4, \not{t_5}, \ldots\}$$

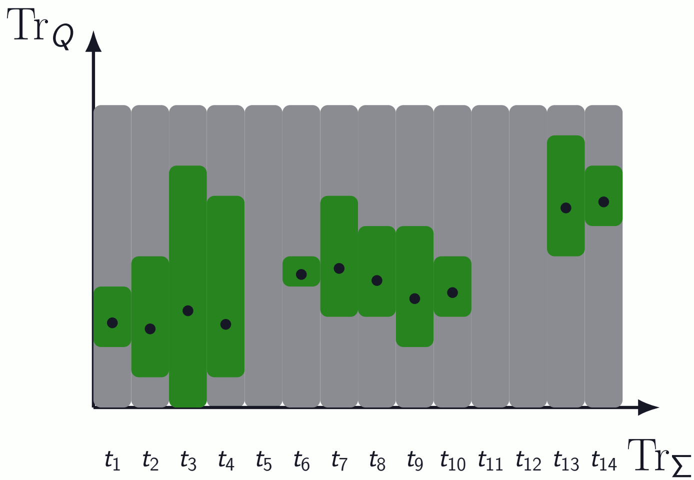


Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{\ldots, t_4, \not{t_5}, \ldots\}$$



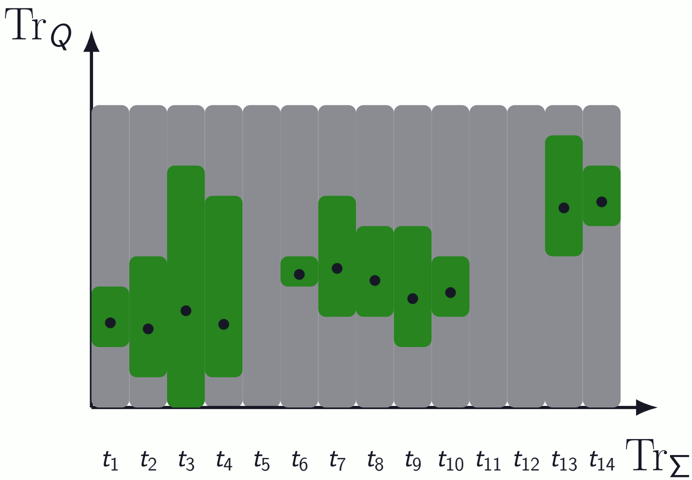
⇝ **no** unambiguity / uniformisation

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{\dots, t_4, \cancel{t_5}, \dots\}$$



⇝ **no** unambiguity / uniformisation

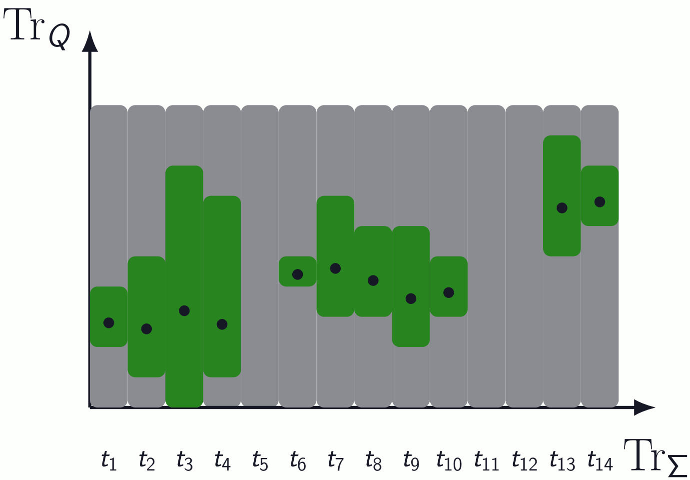
⇝ **no** minimality / canonical form

Inherent non-determinism

See e.g. (Carayol, Löding, Niwiński, Walukiewicz [2010])
(Bilkowski, S. [2013])

$$\mathrm{L}(\mathcal{A}) \stackrel{\text{def}}{=} \{t \in \mathrm{Tr}_\Sigma \mid \underbrace{\exists \rho \in \mathrm{Tr}_Q}_{\text{run}}.\ \underbrace{\forall \pi \in \{\mathtt{L}, \mathtt{R}\}^\omega.\ \text{states of } \rho \text{ on } \pi \text{ are } \Omega\text{-accepting}}_{\text{accepting}}\}$$

$$\mathrm{L}(\mathcal{A}) = \{\ldots, t_4, \not{t_5}, \ldots\}$$



⇝ **no** unambiguity / uniformisation

⇝ **no** minimality / canonical form

⇝ several **decidability questions** still open for automata on infinite trees...

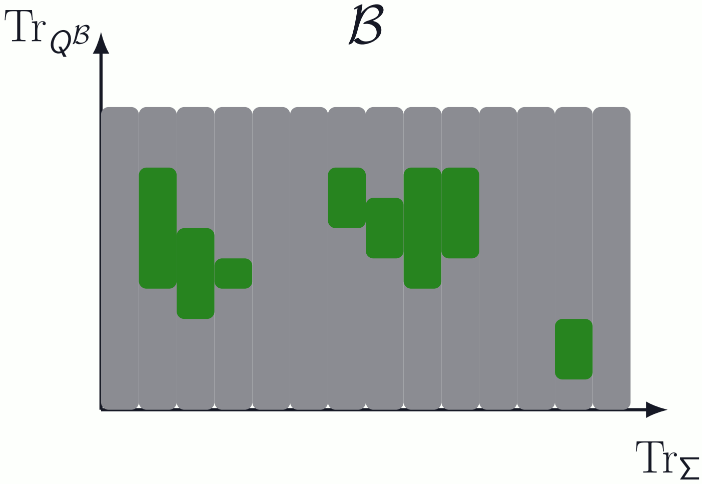
Guidability relation

Guidability relation

(Colcombet, Löding [2008])

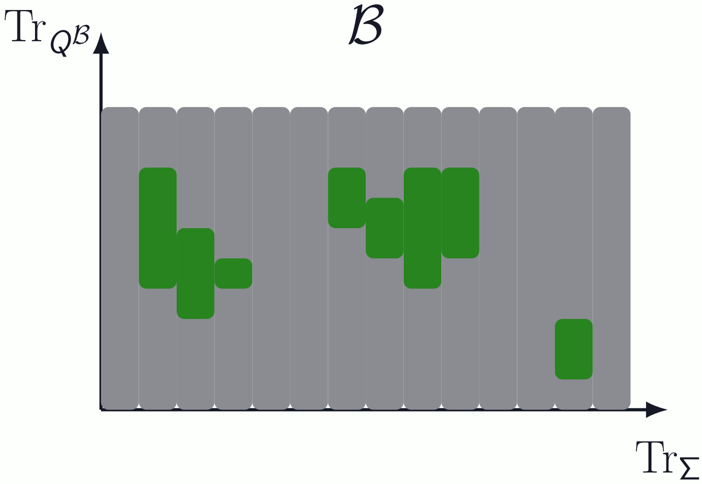
Guidability relation

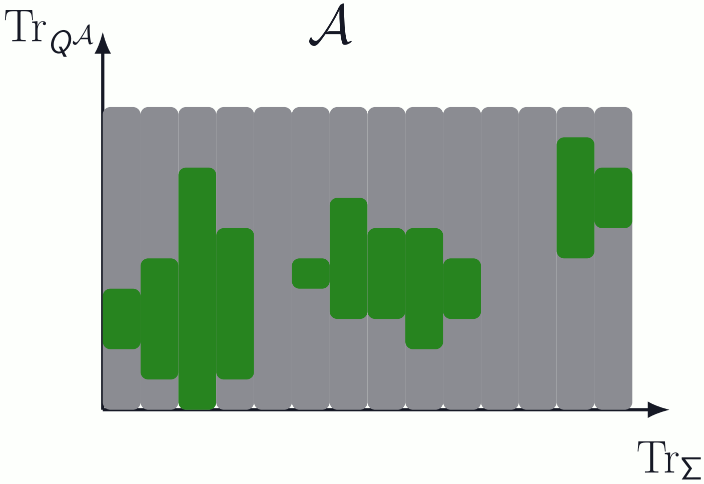
(Colcombet, Löding [2008])



Guidability relation

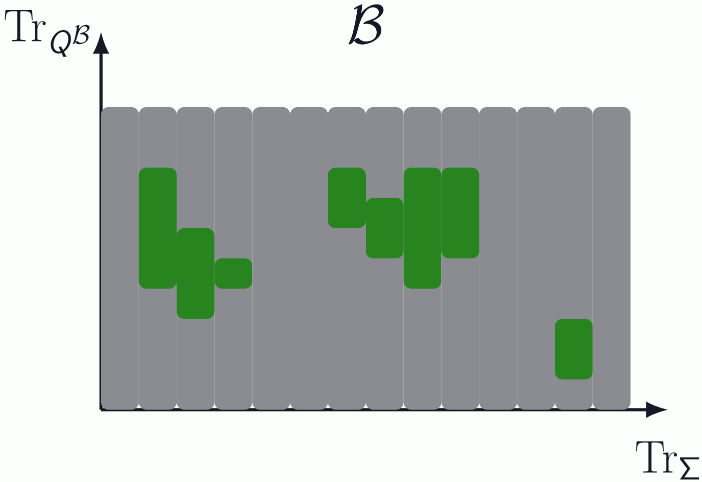
(Colcombet, Löding [2008])

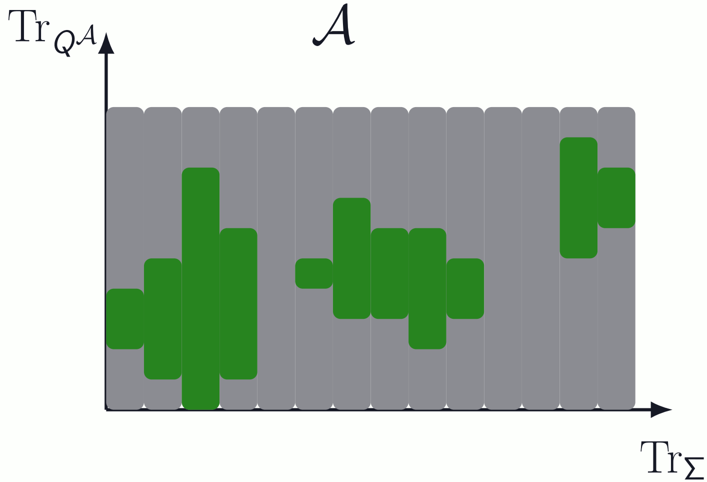




Guidability relation

(Colcombet, Löding [2008])

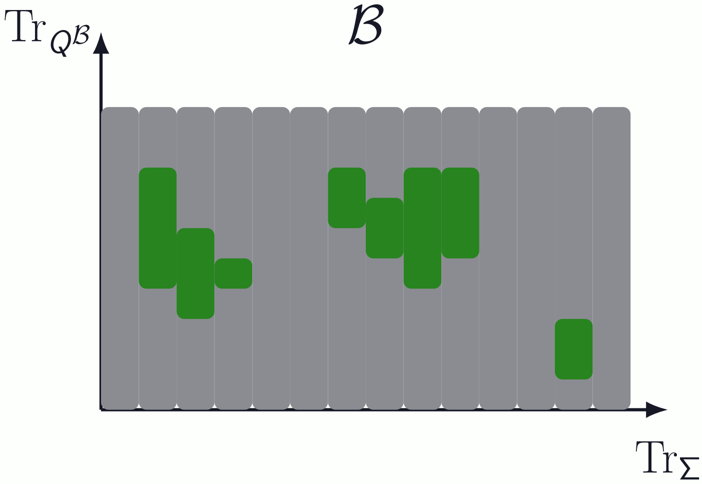


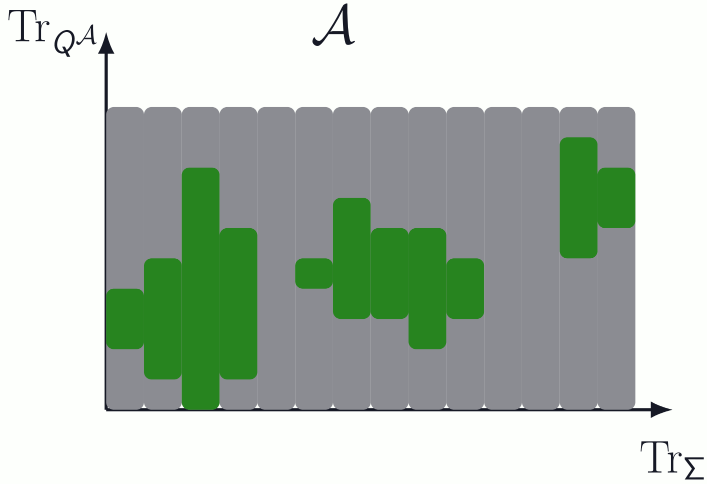


$\mathcal{B}$ **guides** $\mathcal{A}$

Guidability relation

(Colcombet, Löding [2008])

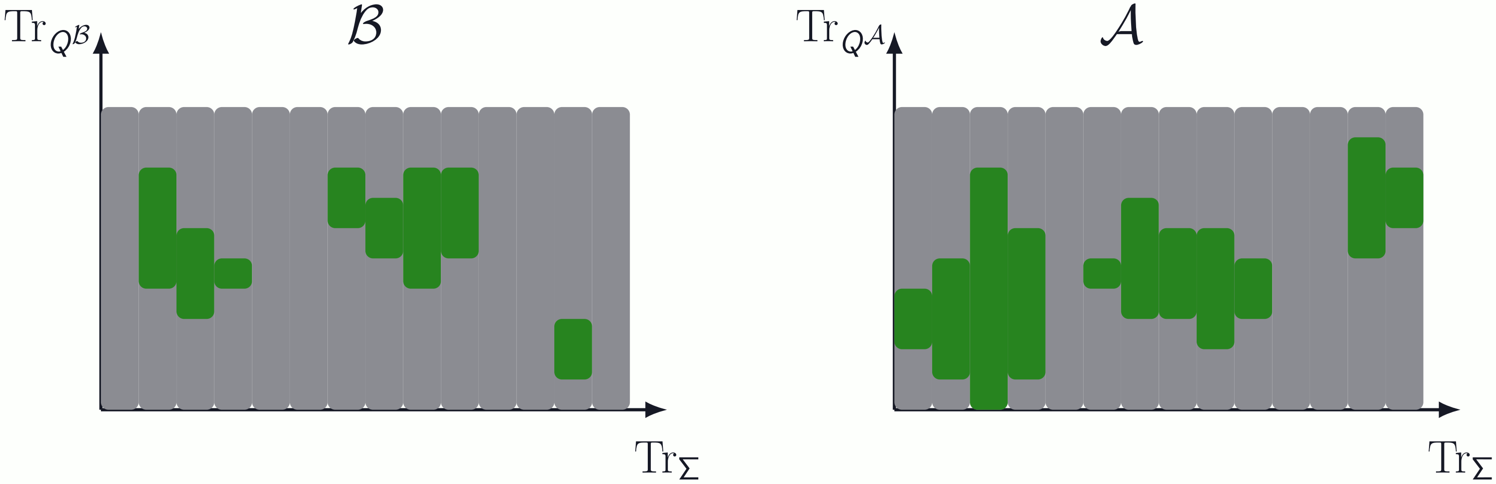




$\mathcal{B}$ **guides** $\mathcal{A}$ (denoted $\mathcal{B} \hookrightarrow \mathcal{A}$)

Guidability relation

(Colcombet, Löding [2008])



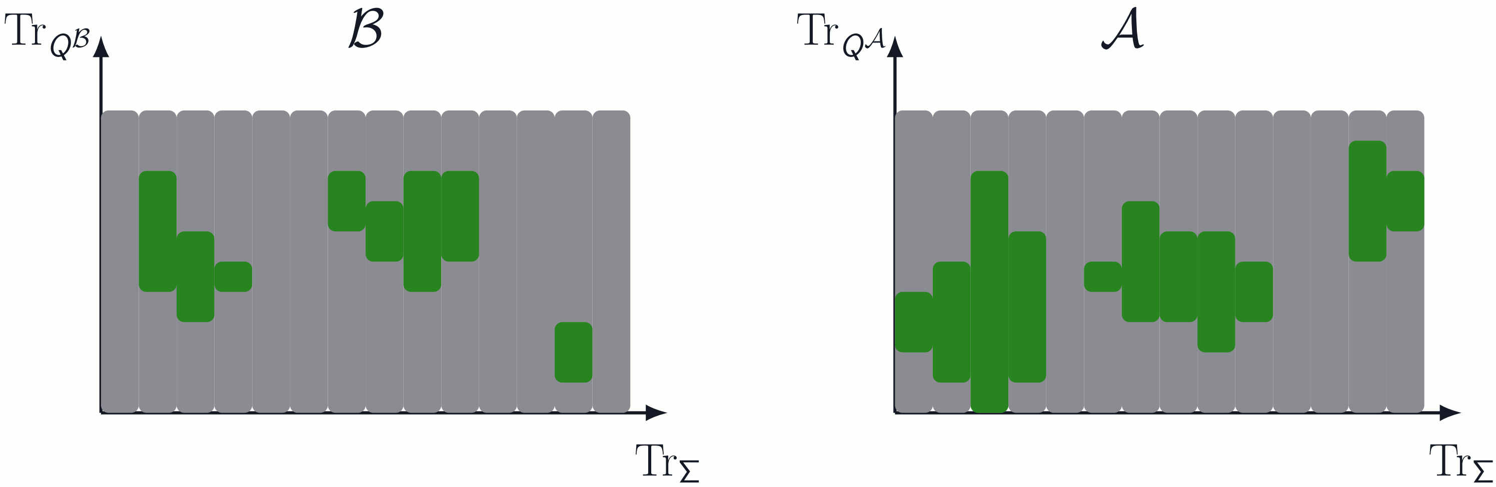
$\mathcal{B}$ **guides** $\mathcal{A}$ (denoted $\mathcal{B} \hookrightarrow \mathcal{A}$)

iff

There exists a **transducer** $\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t)$

Guidability relation

(Colcombet, Löding [2008])



$\mathcal{B}$ **guides** $\mathcal{A}$ (denoted $\mathcal{B} \hookrightarrow \mathcal{A}$)

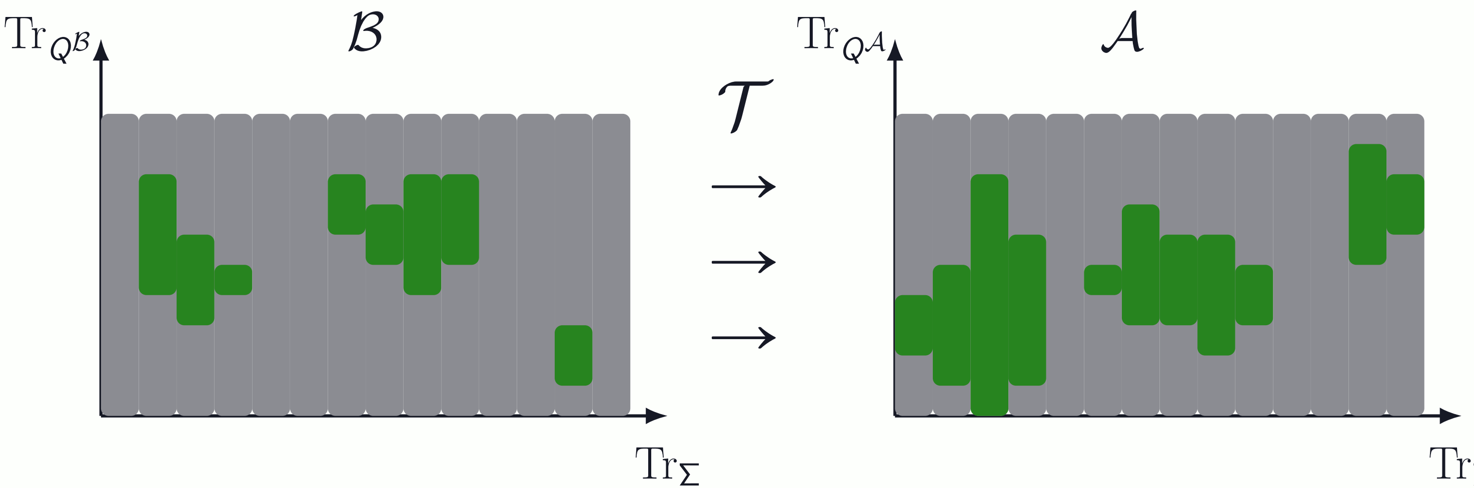
iff

There exists a **transducer** $\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t)$

which maps **accepting runs** $\rho^{\mathcal{B}}$ into **accepting runs** $\rho^{\mathcal{A}}$.

Guidability relation

(Colcombet, Löding [2008])



$\mathcal{B}$ **guides** $\mathcal{A}$ (denoted $\mathcal{B} \hookrightarrow \mathcal{A}$)

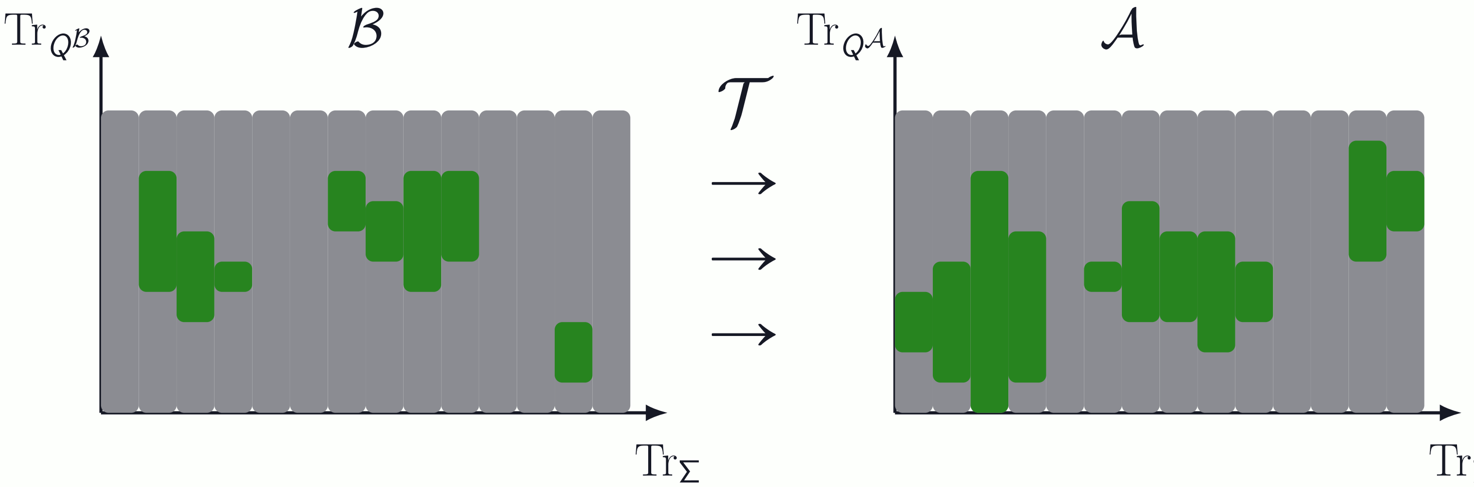
iff

There exists a **transducer** $\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t)$

which maps **accepting runs** $\rho^{\mathcal{B}}$ into **accepting runs** $\rho^{\mathcal{A}}$.

Guidability relation

(Colcombet, Löding [2008])



$\mathcal{B}$ **guides** $\mathcal{A}$ (denoted $\mathcal{B} \hookrightarrow \mathcal{A}$)

iff

There exists a **transducer** $\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t)$

which maps **accepting runs** $\rho^{\mathcal{B}}$ into **accepting runs** $\rho^{\mathcal{A}}$.

In particular: $\mathcal{B} \hookrightarrow \mathcal{A} \implies \mathrm{L}(\mathcal{B}) \subseteq \mathrm{L}(\mathcal{A})$

Guidable automata

Guidable automata

$\mathcal{A}$ is **guidable**

Guidable automata

$\mathcal{A}$ is **guidable**

iff

for every $\mathcal{B}$ s.t. $\mathrm{L}(\mathcal{B}) \subseteq \mathrm{L}(\mathcal{A})$ we have $\mathcal{B} \hookrightarrow \mathcal{A}$

Guidable automata

$\mathcal{A}$ is **guidable**

iff

for every $\mathcal{B}$ s.t. $\mathrm{L}(\mathcal{B}) \subseteq \mathrm{L}(\mathcal{A})$ we have $\mathcal{B} \hookrightarrow \mathcal{A}$

- $\mathcal{A}$ is *universal*

  — every automaton embeds into $\mathcal{A}$

Guidable automata

$\mathcal{A}$ is **guidable**

iff

for every $\mathcal{B}$ s.t. $\mathrm{L}(\mathcal{B}) \subseteq \mathrm{L}(\mathcal{A})$ we have $\mathcal{B} \hookrightarrow \mathcal{A}$

- $\mathcal{A}$ is *universal*

  — every automaton embeds into $\mathcal{A}$

- $\mathcal{A}$ has *most general* non-determinism

  — any way of guessing suffices to guide $\mathcal{A}$

Guidable automata

$\mathcal{A}$ is **guidable**

iff

for every $\mathcal{B}$ s.t. $\mathrm{L}(\mathcal{B}) \subseteq \mathrm{L}(\mathcal{A})$ we have $\mathcal{B} \hookrightarrow \mathcal{A}$

- $\mathcal{A}$ is *universal*

  — every automaton embeds into $\mathcal{A}$

- $\mathcal{A}$ has *most general* non-determinism

  — any way of guessing suffices to guide $\mathcal{A}$

- $\mathcal{A}$ involves *minimal amount* of guessing

  — even deterministic-like automata can guide $\mathcal{A}$

Guidable automata

$\mathcal{A}$ is **guidable**

iff

for every $\mathcal{B}$ s.t. $\mathrm{L}(\mathcal{B}) \subseteq \mathrm{L}(\mathcal{A})$ we have $\mathcal{B} \hookrightarrow \mathcal{A}$

- $\mathcal{A}$ is *universal*

  — every automaton embeds into $\mathcal{A}$

- $\mathcal{A}$ has *most general* non-determinism

  — any way of guessing suffices to guide $\mathcal{A}$

- $\mathcal{A}$ involves *minimal amount* of guessing

  — even deterministic-like automata can guide $\mathcal{A}$

Theorem (Colcombet, Löding [2008])

For every automaton one can construct an **equivalent guidable** automaton.

Guidable automata

$\mathcal{A}$ is **guidable**

iff

for every $\mathcal{B}$ s.t. $\mathrm{L}(\mathcal{B}) \subseteq \mathrm{L}(\mathcal{A})$ we have $\mathcal{B} \hookrightarrow \mathcal{A}$

- $\mathcal{A}$ is *universal*

  — every automaton embeds into $\mathcal{A}$

- $\mathcal{A}$ has *most general* non-determinism

  — any way of guessing suffices to guide $\mathcal{A}$

- $\mathcal{A}$ involves *minimal amount* of guessing

  — even deterministic-like automata can guide $\mathcal{A}$

Theorem (Colcombet, Löding [2008])

For every automaton one can construct an **equivalent guidable** automaton.

$\rightsquigarrow$ every regular tree language is recognised by some guidable automaton

Structure of guidable automata

Structure of guidable automata

Lemma (Colcombet, Löding [2008]; Löding [2009])

If $\mathcal{B} \hookrightarrow \mathcal{A}$ then there exists a **memoryless** transducer

$$\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t).$$

Structure of guidable automata

Lemma (Colcombet, Löding [2008]; Löding [2009])

If $\mathcal{B} \hookrightarrow \mathcal{A}$ then there exists a **memoryless** transducer

$$\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t).$$

Theorem (implicit in Löding [2009])

The relation $\hookrightarrow$ is transitive.

Structure of guidable automata

Lemma (Colcombet, Löding [2008]; Löding [2009])

If $\mathcal{B} \hookrightarrow \mathcal{A}$ then there exists a **memoryless** transducer

$$\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t).$$

Theorem (implicit in Löding [2009])

The relation $\hookrightarrow$ is transitive.

Corollary

Assume that $\mathcal{B}$ is guidable and $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$. Then

Structure of guidable automata

Lemma (Colcombet, Löding [2008]; Löding [2009])

If $\mathcal{B} \hookrightarrow \mathcal{A}$ then there exists a **memoryless** transducer

$$\mathcal{T} \colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t).$$

Theorem (implicit in Löding [2009])

The relation $\hookrightarrow$ is transitive.

Corollary

Assume that $\mathcal{B}$ is guidable and $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$. Then

$$\mathcal{A} \text{ is guidable} \quad \textbf{iff} \quad \mathcal{B} \hookrightarrow \mathcal{A}$$

Structure of guidable automata

Lemma (Colcombet, Löding [2008]; Löding [2009])

If $\mathcal{B} \hookrightarrow \mathcal{A}$ then there exists a **memoryless** transducer

$$\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t).$$

Theorem (implicit in Löding [2009])

The relation $\hookrightarrow$ is transitive.

Corollary

Assume that $\mathcal{B}$ is guidable and $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$. Then

$\mathcal{A}$ is guidable **iff** $\mathcal{B} \hookrightarrow \mathcal{A}$

Fact

Top-down deterministic automata are guidable.

Structure of guidable automata

Lemma (Colcombet, Löding [2008]; Löding [2009])

If $\mathcal{B} \hookrightarrow \mathcal{A}$ then there exists a **memoryless** transducer

$$\mathcal{T}\colon (\rho^{\mathcal{B}}, t) \mapsto (\rho^{\mathcal{A}}, t).$$

Theorem (implicit in Löding [2009])

The relation $\hookrightarrow$ is transitive.

Corollary

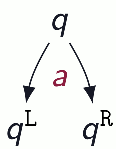
Assume that $\mathcal{B}$ is guidable and $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$. Then

$$\mathcal{A} \text{ is guidable} \quad \textbf{iff} \quad \mathcal{B} \hookrightarrow \mathcal{A}$$

Fact

Top-down deterministic automata are guidable.

(Top-down deterministic:) For each $q \in Q$, $a \in \Sigma$
exactly one transition

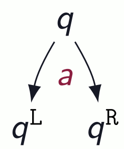


Deterministic and game automata

Deterministic and game automata

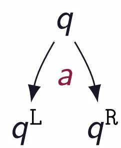
Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition



Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition

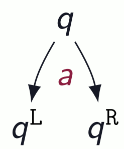


$\rightsquigarrow$ **unique** run on every tree

Deterministic and game automata

Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition



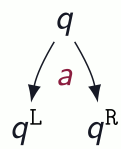
$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \stackrel{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

Deterministic and game automata

Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition



$\rightsquigarrow$ **unique** run on every tree

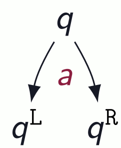
Acceptance game for $t \stackrel{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

$$\left[\ (q_n, u_n) \rightarrow \qquad\qquad\qquad \right]$$

Deterministic and game automata

Deterministic automata

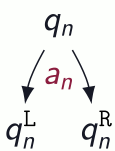
For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition



$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \overset{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

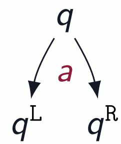
$$\left[\begin{array}{l} (q_n, u_n) \rightarrow \\ \textsc{Tree}\colon\ a_n = t(u_n), \end{array} \right]$$



Deterministic and game automata

Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition

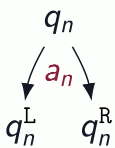


$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \overset{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

$(q_n, u_n) \to$

$\textsc{Tree}$: $a_n = t(u_n)$,

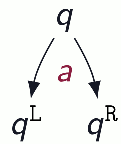


$\forall$: $d_n \in \{\mathtt{L}, \mathtt{R}\}$

Deterministic and game automata

Deterministic automata

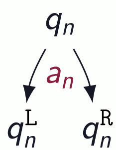
For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition



$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \overset{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

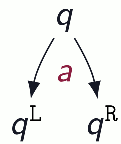
$$\left[\begin{array}{l} (q_n, u_n) \longrightarrow \\ \textsc{Tree}\colon\ a_n = t(u_n), \\ \qquad \forall\colon\ d_n \in \{\mathtt{L}, \mathtt{R}\} \\ \longrightarrow (q_n^{d_n}, u_n d_n) \end{array}\right]$$



Deterministic and game automata

Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition

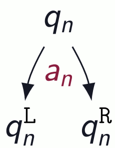


$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \overset{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

$(q_n, u_n) \longrightarrow$

$\textsc{Tree}$: $a_n = t(u_n)$,



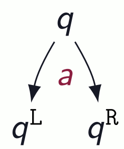
$\forall$: $d_n \in \{\mathtt{L}, \mathtt{R}\}$

$\longrightarrow (q_n^{d_n}, u_n d_n)$

$\textsc{Tree}$ wins
iff
$(q_n)_{n \in \omega}$ is accepting

Deterministic automata

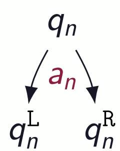
For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition



$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \overset{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

$$\left[\begin{array}{l} (q_n, u_n) \longrightarrow \\ \textsc{Tree}\colon\ a_n = t(u_n), \\ \quad \forall\colon\ d_n \in \{\mathtt{L}, \mathtt{R}\} \\ \longrightarrow (q_n^{d_n}, u_n d_n) \end{array}\right]$$



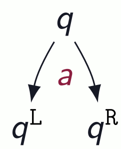
Tree wins
iff
$(q_n)_{n\in\omega}$ is accepting

Deterministic ≡ *Conjunctive*

Deterministic and game automata

Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition

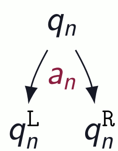


$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \overset{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

$(q_n, u_n) \rightarrow$

Tree: $a_n = t(u_n)$,



$\forall$: $d_n \in \{\mathtt{L}, \mathtt{R}\}$

$\rightarrow (q_n^{d_n}, u_n d_n)$

Tree wins
iff
$(q_n)_{n \in \omega}$ is accepting

Deterministic $\equiv$ *Conjunctive*

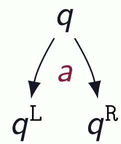
Game automata (Duparc, Facchini, Murlak [2009])

As above, but the player $\exists$ / $\forall$ choosing d_n depends on (q_n, a_n).

Deterministic and game automata

Deterministic automata

For each $q \in Q$, $a \in \Sigma$ there is a **unique** transition

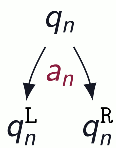


$\rightsquigarrow$ **unique** run on every tree

Acceptance game for $t \overset{?}{\in} \mathrm{L}(\mathcal{A})$: positions are $(q, u) \in Q \times \{\mathtt{L}, \mathtt{R}\}^*$

$(q_n, u_n) \rightarrow$

Tree: $a_n = t(u_n)$,



$\forall$: $d_n \in \{\mathtt{L}, \mathtt{R}\}$

$\rightarrow (q_n^{d_n}, u_n d_n)$

Tree wins
iff
$(q_n)_{n\in\omega}$ is accepting

Deterministic $\equiv$ *Conjunctive*

Game automata (Duparc, Facchini, Murlak [2009])

As above, but the player $\exists$ / $\forall$ choosing d_n depends on (q_n, a_n).

Det. $\subsetneq$ Game $\subsetneq$ Non-det.

Game automata are guidable

Game automata are guidable

Theorem

If $\mathcal{A}$ is a **game automaton** then $\mathcal{A}$ is **guidable**.

Game automata are guidable

Theorem

If $\mathcal{A}$ is a **game automaton** then $\mathcal{A}$ is **guidable**.

Proof

The procedure of **constructing** a guidable automaton from $\mathcal{A}$ produces $\mathcal{A}$.

Game automata are guidable

Theorem

If $\mathcal{A}$ is a **game automaton** then $\mathcal{A}$ is **guidable**.

Proof

The procedure of **constructing** a guidable automaton from $\mathcal{A}$ produces $\mathcal{A}$.

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**

Game automata are guidable

Theorem

If $\mathcal{A}$ is a **game automaton** then $\mathcal{A}$ is **guidable**.

Proof

The procedure of **constructing** a guidable automaton from $\mathcal{A}$ produces $\mathcal{A}$.

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**

Languages: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $=$ **Non-det.**

Γ -index problem

Γ -index problem

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**

Languages: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $=$ **Non-det.**

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**
Languages: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $=$ **Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}}\colon Q^{\mathcal{B}} \to \{i, \ldots, j\}$

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}}\colon Q^{\mathcal{B}} \to \{i, \ldots, j\}$

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}}\colon Q^{\mathcal{B}} \to \{i, \ldots, j\}$

[the input $\mathcal{A}$ is unrestricted!]

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i,j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i,j), *i.e.* $\Omega^{\mathcal{B}}\colon Q^{\mathcal{B}} \to \{i,\ldots,j\}$

[the input $\mathcal{A}$ is unrestricted!]

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

[**Mostowski-Rabin** index problem = **(Non-det.)**-index problem]

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**

Languages: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $=$ **Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}}\colon Q^{\mathcal{B}} \to \{i, \ldots, j\}$

[the input $\mathcal{A}$ is unrestricted!]

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

[**Mostowski-Rabin** index problem = **(Non-det.)**-index problem]

(Det.)-index problem ✓ (Niwiński, Walukiewicz [1998])

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}}\colon Q^{\mathcal{B}} \to \{i, \ldots, j\}$

[the input $\mathcal{A}$ is unrestricted!]

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

[**Mostowski-Rabin** index problem = **(Non-det.)**-index problem]

(Det.)-index problem	✓	(Niwiński, Walukiewicz [1998])
(Game)-index problem	✓	(Clemente, S. [2021])

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**

Languages: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $=$ **Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}} \colon Q^{\mathcal{B}} \to \{i, \ldots, j\}$

[the input $\mathcal{A}$ is unrestricted!]

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

[**Mostowski-Rabin** index problem = **(Non-det.)**-index problem]

(Det.)-index problem	✓	(Niwiński, Walukiewicz [1998])
(Game)-index problem	✓	(Clemente, S. [2021])
(Guidable)-index problem	?	- - - **this work** - - -

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}} \colon Q^{\mathcal{B}} \to \{i, \ldots, j\}$

[the input $\mathcal{A}$ is unrestricted!]

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

[**Mostowski-Rabin** index problem = **(Non-det.)**-index problem]

(Det.)-index problem	✓	(Niwiński, Walukiewicz [1998])
(Game)-index problem	✓	(Clemente, S. [2021])
(Guidable)-index problem	**?**	- - - **this work** - - -
(Non-det.)-index problem	**???**	

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Γ -index problem

Problem

Given $\mathcal{A}$ and (i, j) decide if there exists $\mathcal{B}$ such that:

- $\mathcal{B}$ is in Γ
- $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$
- $\mathcal{B}$ has index (i, j), *i.e.* $\Omega^{\mathcal{B}} \colon Q^{\mathcal{B}} \to \{i, \dots, j\}$

[the input $\mathcal{A}$ is unrestricted!]

$\equiv$ find the simplest (in terms of index) automaton from Γ for $\mathrm{L}(\mathcal{A})$

[**Mostowski-Rabin** index problem = **(Non-det.)**-index problem]

(Det.)-index problem	✓	(Niwiński, Walukiewicz [1998])
(Game)-index problem	✓	(Clemente, S. [2021])
(Guidable)-index problem	**?**	- - - **this work** - - -
(Non-det.)-index problem	**???**	attempt in (Colcombet, Löding [2007])

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**
Languages: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** = **Non-det.**

Main result (Guidable)-index problem is decidable!

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j)

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$\left[(p_n, q_n) \to \right]$$

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$\left[(p_n, q_n) \to \exists\text{: } c_n \in \{i, \ldots, j\} \right]$$

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$\left[\begin{array}{lll} (p_n, q_n) \rightarrow & \exists: & c_n \in \{i, \ldots, j\} \\ & \forall: & a_n \in \Sigma \end{array}\right]$$

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

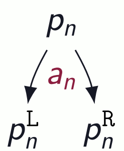
Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$(p_n, q_n) \longrightarrow$ $\exists$: $c_n \in \{i, \ldots, j\}$

$\forall$: $a_n \in \Sigma$

$\forall$:



Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$\left[\ (p_n, q_n) \longrightarrow \quad \begin{array}{ll} \exists: & c_n \in \{i, \ldots, j\} \\ \forall: & a_n \in \Sigma \\ \forall: & p_n \xrightarrow{a_n} p_n^{\mathrm{L}} \quad p_n^{\mathrm{R}} \\ \exists: & q_n \xrightarrow{a_n} q_n^{\mathrm{L}} \quad q_n^{\mathrm{R}} \end{array} \ \right]$$

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$\left[\begin{array}{lll} (p_n, q_n) \longrightarrow & \exists: & c_n \in \{i, \ldots, j\} \\ & \forall: & a_n \in \Sigma \\ & \forall: & p_n \xrightarrow{a_n} (p_n^{\mathrm{L}}, p_n^{\mathrm{R}}) \\ & \exists: & q_n \xrightarrow{a_n} (q_n^{\mathrm{L}}, q_n^{\mathrm{R}}) \\ & \forall: & d_n \in \{\mathtt{L}, \mathtt{R}\} \end{array}\right]$$

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$
\left[\begin{array}{lll}
(p_n, q_n) \longrightarrow & \exists: & c_n \in \{i, \ldots, j\} \\
& \forall: & a_n \in \Sigma \\
& \forall: & p_n \xrightarrow{a_n} (p_n^{\mathtt{L}}, p_n^{\mathtt{R}}) \\
& \exists: & q_n \xrightarrow{a_n} (q_n^{\mathtt{L}}, q_n^{\mathtt{R}}) \\
& \forall: & d_n \in \{\mathtt{L}, \mathtt{R}\} \quad \longrightarrow (p_n^{d_n}, q_n^{d_n})
\end{array}\right]
$$

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$\left[\begin{array}{lll}
(p_n, q_n) \longrightarrow & \exists: & c_n \in \{i, \ldots, j\} \\
& \forall: & a_n \in \Sigma \\
& \forall: & \begin{array}{c} p_n \\ \swarrow a_n \searrow \\ p_n^{\mathrm{L}} \quad p_n^{\mathrm{R}} \end{array} \\
& \exists: & \begin{array}{c} q_n \\ \swarrow a_n \searrow \\ q_n^{\mathrm{L}} \quad q_n^{\mathrm{R}} \end{array} \\
& \forall: & d_n \in \{\mathtt{L}, \mathtt{R}\} \quad \longrightarrow (p_n^{d_n}, q_n^{d_n})
\end{array}\right]$$

$\exists$ wins if:

Main result (Guidable)-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$$\left[\begin{array}{lll} (p_n, q_n) \to & \exists: & c_n \in \{i, \ldots, j\} \\ & \forall: & a_n \in \Sigma \\ & \forall: & p_n \xrightarrow{a_n} (p_n^{\mathrm{L}}, p_n^{\mathrm{R}}) \\ & \exists: & q_n \xrightarrow{a_n} (q_n^{\mathrm{L}}, q_n^{\mathrm{R}}) \\ & \forall: & d_n \in \{\mathrm{L}, \mathrm{R}\} \quad \to (p_n^{d_n}, q_n^{d_n}) \end{array}\right]$$

$\exists$ wins if:

(i) (p_n) is acc. $\Longrightarrow$ (c_n) is acc.

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i, j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i, j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p, q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$(p_n, q_n) \to$ $\exists$: $c_n \in \{i, \ldots, j\}$

$\forall$: $a_n \in \Sigma$

$\forall$: $p_n \xrightarrow{a_n} (p_n^{\mathrm{L}}, p_n^{\mathrm{R}})$

$\exists$: $q_n \xrightarrow{a_n} (q_n^{\mathrm{L}}, q_n^{\mathrm{R}})$

$\forall$: $d_n \in \{\mathtt{L}, \mathtt{R}\}$ $\quad \to (p_n^{d_n}, q_n^{d_n})$

$\exists$ wins if:

(i) (p_n) is acc. $\Longrightarrow$ (c_n) is acc.

(ii) (c_n) is acc. $\Longrightarrow$ (q_n) is acc.

Main result **(Guidable)**-index problem is decidable!

Given **any** automaton $\mathcal{A}$ and index (i,j), one can decide

if there exists a **guidable** automaton $\mathcal{B}$ of index (i,j) s.t. $\mathrm{L}(\mathcal{B}) = \mathrm{L}(\mathcal{A})$.

When exists, such $\mathcal{B}$ can be constructed!

Proof (W.l.o.g. assume that $\mathcal{A}$ is guidable.)

Solve **game**: positions are $(p,q) \in Q^{\mathcal{A}} \times Q^{\mathcal{A}}$

$(p_n, q_n) \to$ $\exists$: $c_n \in \{i,\dots,j\}$

$\forall$: $a_n \in \Sigma$

$\forall$: $p_n \xrightarrow{a_n} (p_n^{\mathrm{L}}, p_n^{\mathrm{R}})$

$\exists$: $q_n \xrightarrow{a_n} (q_n^{\mathrm{L}}, q_n^{\mathrm{R}})$

$\forall$: $d_n \in \{\mathtt{L}, \mathtt{R}\}$ $\to (p_n^{d_n}, q_n^{d_n})$

$\exists$ wins if:

(i) (p_n) is acc. $\Longrightarrow$ (c_n) is acc.

(ii) (c_n) is acc. $\Longrightarrow$ (q_n) is acc.

Claim $\exists$ has a **winning strategy** **iff** there exists such $\mathcal{B}$

Index transfer

Index transfer

Det. $\subsetneq$ **Game** $\subsetneq$ **Guidable**

Index transfer

Det. $\subsetneq$ **Game** $\subsetneq$ **Guidable**

Theorem

Assume that $\mathbf{\Delta} \subseteq \mathbf{\Gamma}$ are some of the above classes.

Index transfer

$$\textbf{Det.} \subsetneq \textbf{Game} \subsetneq \textbf{Guidable}$$

Theorem

Assume that $\mathbf{\Delta} \subseteq \mathbf{\Gamma}$ are some of the above classes.

If

Index transfer

Det. $\subsetneq$ **Game** $\subsetneq$ **Guidable**

Theorem

Assume that $\mathbf{\Delta} \subseteq \mathbf{\Gamma}$ are some of the above classes.

If

- L can be recognised by some $\mathbf{\Gamma}$ automaton of index (i, j),

Index transfer

Det. $\subsetneq$ **Game** $\subsetneq$ **Guidable**

Theorem

Assume that $\boldsymbol{\Delta} \subseteq \boldsymbol{\Gamma}$ are some of the above classes.

If

- L can be recognised by some $\boldsymbol{\Gamma}$ automaton of index (i, j),
- L can be recognised by some $\boldsymbol{\Delta}$ automaton

Index transfer

Det. $\subsetneq$ **Game** $\subsetneq$ **Guidable**

Theorem

Assume that $\Delta \subseteq \Gamma$ are some of the above classes.

If

- L can be recognised by some Γ automaton of index (i, j),
- L can be recognised by some Δ automaton

then

Index transfer

Det. $\subsetneq$ **Game** $\subsetneq$ **Guidable**

Theorem

Assume that $\mathbf{\Delta} \subseteq \mathbf{\Gamma}$ are some of the above classes.

If

- L can be recognised by some $\mathbf{\Gamma}$ automaton of index (i, j),
- L can be recognised by some $\mathbf{\Delta}$ automaton

then

- L can be recognised by some $\mathbf{\Delta}$ automaton of index (i, j).

Index transfer

Det. $\subsetneq$ **Game** $\subsetneq$ **Guidable**

Theorem

Assume that $\mathbf{\Delta} \subseteq \mathbf{\Gamma}$ are some of the above classes.

If

- L can be recognised by some $\mathbf{\Gamma}$ automaton of index (i, j),
- L can be recognised by some $\mathbf{\Delta}$ automaton

then

- L can be recognised by some $\mathbf{\Delta}$ automaton of index (i, j).

Remark

But **no** such transfer for $\mathbf{\Gamma} =$ **(Non-det.)**!

Summary

Summary

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Summary

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Theorem

(Guidable)-index problem is decidable.

Summary

Automata:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$\subsetneq$	**Non-det.**
Languages:	**Det.**	$\subsetneq$	**Game**	$\subsetneq$	**Guidable**	$=$	**Non-det.**

Theorem

(Guidable)-index problem is decidable.

Theorem

Index transfer results between **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable**.

Summary

Automata: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $\subsetneq$ **Non-det.**
Languages: **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable** $=$ **Non-det.**

Theorem

(Guidable)-index problem is decidable.

Theorem

Index transfer results between **Det.** $\subsetneq$ **Game** $\subsetneq$ **Guidable**.

+ New game-based characterisation!