Automates, algèbre, applications - AAA

CM 3

Uli Fahrenberg Sven Dziadek Philipp Schlehuber
Adrien Pommellet Etienne Renault

EPITA

S6 2022

Foreword

Program of the course

1. CM 1 : Weighted automata — 5 May
2. TD : Weighted automata — 12 May
3. CM 2 : LTL model checking — 12 May
4. CM 3 : ω-Automata — 19 May
5. DM (noté)
6. TP : ω-Automata — 2 June
7. CM 4 : Automata learning — 16 June

Program of the course

1. CM 1 : Weighted automata — 5 May
2. TD : Weighted automata — 12 May
3. CM 2 : LTL model checking — 12 May
4. CM 3 : ω-Automata — 19 May
5. DM (noté)
6. TP : ω-Automata — 2 June
7. CM 4 : Automata learning — 16 June

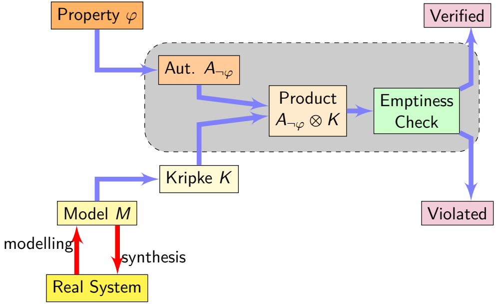
CM 3 : ω-Automata

Sources:

- Farwer, B, ω-automata. In: Automata, Logics, and Infinite Games (pp. 3-21), Springer, 2002.
- `https://spot.lrde.epita.fr/concepts.html`

Introduction

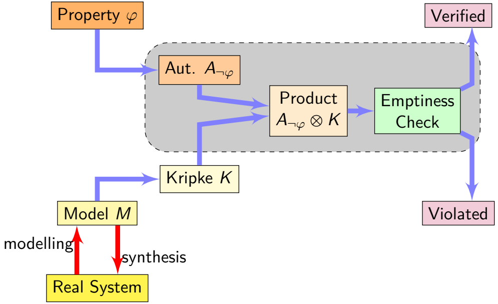
Model Checking



Recall

Model checking translates LTL to automata and checks for emptiness of intersection of model and specification.

Model Checking



Recall

Model checking translates LTL to automata and checks for emptiness of intersection of model and specification.

Goal of this lecture

Understand ω-automata

Ordinal Numbers

0

1

2

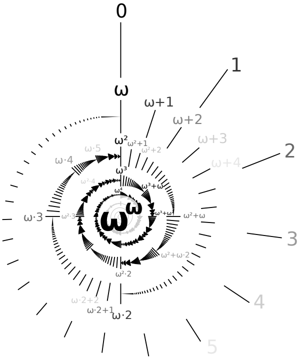
⋮

n

⋮

ω order type of $\mathbb{N}$

⋮



specify indices in ordered sequences

ω-Words

Σ: finite alphabet, $n \in \mathbb{N} \cup \{\omega\}$: ordinal number.

Definition

An *n-word* w over Σ is a sequence of length n,
i.e., a function $w : [0, n[\to \Sigma$.

ω-Words

Σ: finite alphabet, $n \in \mathbb{N} \cup \{\omega\}$: ordinal number.

Definition

An *n-word* w over Σ is a sequence of length n,
i.e., a function $w : [0, n[\to \Sigma$.

Notations:

Σ^n set of n-words,

Σ^* set of finite words $(n < \omega)$,

Σ^ω set of ω-words $(n = \omega)$,

$\Sigma^\infty = \Sigma^* \cup \Sigma^\omega$ set of all words $(n \leq \omega)$,

ω-Rational Expressions

Definition (ω-Rational Expressions)

ω-languages are called ω-regular if they are of the form:

Above: L is a regular language (of finite words)
K, K_1, K_2 are ω-regular languages

ω-Rational Expressions

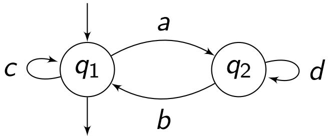
Definition (ω-Rational Expressions)

ω-languages are called ω-regular if they are of the form:

- $K_1 \cup K_2$,
- $LK = \{vw \mid v \in L, w \in K\}$,
- $L^\omega = \{w_0 w_1 w_2 \cdots \mid w_i \in L\}$ for $\{\varepsilon\} \notin L$,

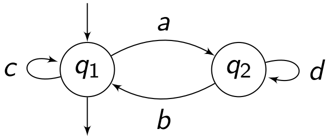
Above: L is a regular language (of finite words)
K, K_1, K_2 are ω-regular languages

Example



Which words are accepted?

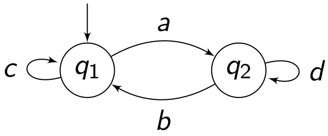
Example



Which words are accepted?

$$(c^* a d^* b)^* c^*$$

Example

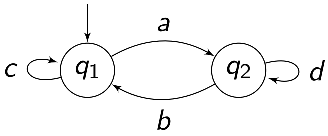


Which words are accepted?

$$(c^* a d^* b)^* c^*$$

Which ω-paths exist?

Example



Which words are accepted?

$$(c^* a d^* b)^* c^*$$

Which ω-paths exist?

$$(c^* a d^* b)^\omega + (c^* a d^* b)^* c^\omega + (c^* a d^* b)^* c^* a d^\omega$$

Büchi Automata

Infinite Sequences

Definition

$$\mathrm{Inf}((q_i)_{i\geq 0}) = \{q \mid q = q_i \text{ for infinitely many } i \geq 0\}$$

Example

$$\mathrm{Inf}(0,1,2,3,4,5,\ldots) = \emptyset$$
$$\mathrm{Inf}(0,1,2,3,3,3,\ldots) = \{3\}$$
$$\mathrm{Inf}(0,1,2,3,2,3,2,3,\ldots) = \{2,3\}$$

Büchi Automata: Definition

Definition (Büchi Automaton)

$\mathcal{A} = (Q, I, T, F)$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $F \subseteq Q$: Büchi-accepting states.

Büchi Automata: Definition

Definition (Büchi Automaton)

$\mathcal{A} = (Q, I, T, F)$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $F \subseteq Q$: Büchi-accepting states.

Run of $\mathcal{A}$ on $w \in \Sigma^\omega$: $\rho = (q_i)_{i \geq 0}$ with

- $q_0 \in I$,
- for all $i \geq 0$: $q_i \xrightarrow{w_i} q_{i+1} \in T$

Büchi Automata: Definition

Definition (Büchi Automaton)

$\mathcal{A} = (Q, I, T, F)$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $F \subseteq Q$: Büchi-accepting states.

Run of $\mathcal{A}$ on $w \in \Sigma^\omega$: $\rho = (q_i)_{i \geq 0}$ with

- $q_0 \in I$,
- for all $i \geq 0$: $q_i \xrightarrow{w_i} q_{i+1} \in T$

Run ρ Büchi-accepted if
an accepting state occurs infinitely often,

Büchi Automata: Definition

Definition (Büchi Automaton)

$\mathcal{A} = (Q, I, T, F)$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $F \subseteq Q$: Büchi-accepting states.

Run of $\mathcal{A}$ on $w \in \Sigma^\omega$: $\rho = (q_i)_{i \geq 0}$ with

- $q_0 \in I$,
- for all $i \geq 0$: $q_i \xrightarrow{w_i} q_{i+1} \in T$

Run ρ Büchi-accepted if
an accepting state occurs infinitely often,
i.e., $\mathrm{Inf}(\rho) \cap F \neq \emptyset$.

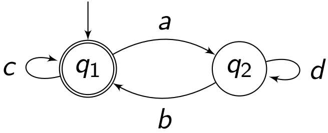
Büchi Automata: Definition 2

The language recognized by a Büchi Automaton $\mathcal{A}$:

$$\mathcal{L}(\mathcal{A}) = \{w \in \Sigma^\omega \mid \exists \text{ Büchi-accepted run of } \mathcal{A} \text{ on } w\}$$

A language recognized by Büchi Automata is called Büchi-recognizable.

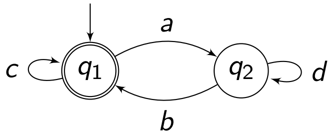
Büchi Automata: Example



$$F = \{q_1\}$$

Which ω-words are accepted? $(\mathrm{Inf}(\rho) \cap F \neq \emptyset)$

Büchi Automata: Example

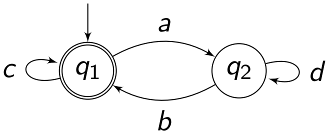


$$F = \{q_1\}$$

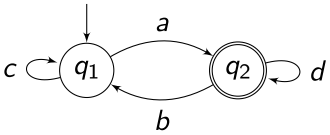
Which ω-words are accepted? $(\mathrm{Inf}(\rho) \cap F \neq \emptyset)$

$$(c^* a d^* b)^\omega + (c^* a d^* b)^* c^\omega$$

Büchi Automata: Example



$F = \{q_1\}$

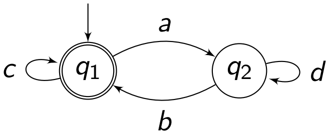


$F = \{q_2\}$

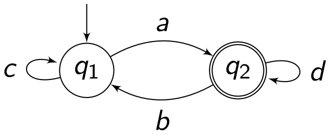
Which ω-words are accepted? $(\mathrm{Inf}(\rho) \cap F \neq \emptyset)$

$$(c^* a d^* b)^\omega + (c^* a d^* b)^* c^\omega$$

Büchi Automata: Example



$F = \{q_1\}$



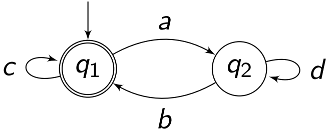
$F = \{q_2\}$

Which ω-words are accepted? $(\mathrm{Inf}(\rho) \cap F \neq \emptyset)$

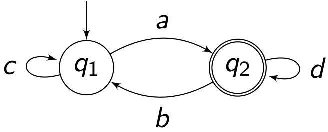
$(c^* ad^* b)^\omega + (c^* ad^* b)^* c^\omega$

$(c^* ad^* b)^\omega + (c^* ad^* b)^* c^* ad^\omega$

Büchi Automata: Example



$F = \{q_1\}$



$F = \{q_2\}$

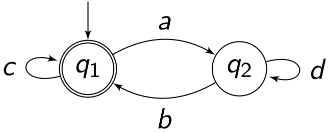
Which ω-words are accepted? $(\mathrm{Inf}(\rho) \cap F \neq \emptyset)$

$$(c^* a d^* b)^\omega + (c^* a d^* b)^* c^\omega$$

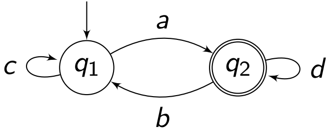
$$(c^* a d^* b)^\omega + (c^* a d^* b)^* c^* a d^\omega$$

$F = \{q_1, q_2\} \quad \Rightarrow$

Büchi Automata: Example



$F = \{q_1\}$



$F = \{q_2\}$

Which ω-words are accepted? $(\mathrm{Inf}(\rho) \cap F \neq \emptyset)$

$(c^* ad^* b)^\omega + (c^* ad^* b)^* c^\omega$

$(c^* ad^* b)^\omega + (c^* ad^* b)^* c^* ad^\omega$

$F = \{q_1, q_2\} \quad \Rightarrow$
$\mathcal{L}(\mathcal{A}) = (c^* ad^* b)^\omega + (c^* ad^* b)^* c^\omega + (c^* ad^* b)^* c^* ad^\omega$

Emptiness is Decidable

Büchi automaton: $\mathcal{A} = (Q, I, T, F)$

- Interpret $\mathcal{A}$ as graph (Q, T)
- Check reachability of states in F from states in I (Floyd-Warshall)
- For every reachable accepting state f:
  Check (non-trivial) path from f to f.

Emptiness is Decidable

Büchi automaton: $\mathcal{A} = (Q, I, T, F)$

- Interpret $\mathcal{A}$ as graph (Q, T)
- Check reachability of states in F from states in I (Floyd-Warshall)
- For every reachable accepting state f:
  Check (non-trivial) path from f to f.

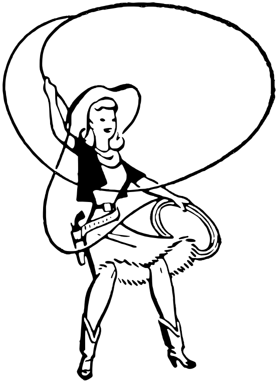
$\Rightarrow$ search for lassos

Emptiness is Decidable

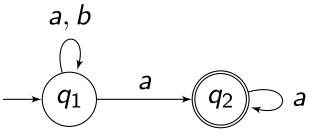
Büchi automaton: $\mathcal{A} = (Q, I, T, F)$

- Interpret $\mathcal{A}$ as graph (Q, T)
- Check reachability of states in F from states in I (Floyd-Warshall)
- For every reachable accepting state f:
  Check (non-trivial) path from f to f.

$\Rightarrow$ search for lassos

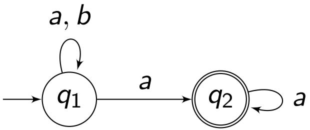


Deterministic Büchi Automata



$$L := \mathcal{L}(\mathcal{A}) = (a+b)^* a^\omega$$

Deterministic Büchi Automata

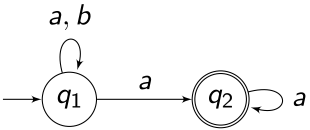


$$L := \mathcal{L}(\mathcal{A}) = (a+b)^* a^\omega$$

Lemma

L is not accepted by any deterministic Büchi Automata

Deterministic Büchi Automata



$$L := \mathcal{L}(\mathcal{A}) = (a+b)^* a^\omega$$

Lemma

L is not accepted by any deterministic Büchi Automata

Proof.

Suppose: $\mathcal{B} = (Q, I, T, F)$ is a det. Büchi automaton recognizing L.
$\mathcal{B}$ accepts a^ω. After some finite prefix of a^ω, $\mathcal{B}$ will visit some state in F, say after i_0 letters.
$\mathcal{B}$ also accepts $a^{i_0} b a^\omega$. Therefore, for some i_1, after the prefix $a^{i_0} b a^{i_1}$, $\mathcal{B}$ will visit some state in F.
Continue: $a^{i_0} b a^{i_1} b a^{i_2} \cdots$ is recognized by $\mathcal{B}$ as states in F are visited infinitely often. But: $a^{i_0} b a^{i_1} b a^{i_2} \cdots \notin L$. $\lightning$

Büchi Automata: Properties

The class of languages accepted by Büchi automata is closed under:

- union (easy)

Büchi Automata: Properties

The class of languages accepted by Büchi automata is closed under:

- union (easy)
- intersection (product construction, see slide 31)

Büchi Automata: Properties

The class of languages accepted by Büchi automata is closed under:

- union (easy)
- intersection (product construction, see slide 31)
- complement (hard)
  Büchi (1960): construction with $2^{2^{\mathrm{O}(n)}}$ states,
  Klarlund (1991), *Safra* (1992): $2^{\mathrm{O}(n \log n)}$

Büchi Automata: Properties

The class of languages accepted by Büchi automata is closed under:

- union (easy)
- intersection (product construction, see slide 31)
- complement (hard)
  Büchi (1960): construction with $2^{2^{\mathrm{O}(n)}}$ states,
  Klarlund (1991), *Safra* (1992): $2^{\mathrm{O}(n \log n)}$

Emptiness is decidable

Büchi Automata: Properties

The class of languages accepted by Büchi automata is closed under:

- union (easy)
- intersection (product construction, see slide 31)
- complement (hard)
  Büchi (1960): construction with $2^{2^{\mathrm{O}(n)}}$ states,
  Klarlund (1991), *Safra* (1992): $2^{\mathrm{O}(n \log n)}$

Emptiness is decidable

But: Büchi automata are not always determinisable.

Büchi vs. ω-Regular Expressions

Theorem

Büchi-recognizable languages are exactly the ω-regular languages

Büchi vs. ω-Regular Expressions

Theorem

Büchi-recognizable languages are exactly the ω-regular languages

Proof Sketch

$\Leftarrow$ Büchi automata closed under operation for ω-regular expressions

Büchi vs. ω-Regular Expressions

Theorem

Büchi-recognizable languages are exactly the ω-regular languages

Proof Sketch

$\Leftarrow$ Büchi automata closed under operation for ω-regular expressions

$\Rightarrow$ Runs ρ in Büchi automata can be divided into two regular parts:

prefix: $\rho_f = q_0 \to \ldots \to q_f$ for $q_0 \in I$ and $q_f \in F$

loop: $\rho'_f = q_f \to \ldots \to q_f$

Büchi vs. ω-Regular Expressions

Theorem

Büchi-recognizable languages are exactly the ω-regular languages

Proof Sketch

$\Leftarrow$ Büchi automata closed under operation for ω-regular expressions

$\Rightarrow$ Runs ρ in Büchi automata can be divided into two regular parts:

prefix: $\rho_f = q_0 \to \ldots \to q_f$ for $q_0 \in I$ and $q_f \in F$

loop: $\rho'_f = q_f \to \ldots \to q_f$

All possible runs: $\bigcup_{f \in F} \rho_f \rho_f'^{\omega}$ is an ω-regular expression

Muller Automata

Muller Automata: Definition

Definition (Muller Automaton)

$\mathcal{A} = (Q, I, T, \mathcal{F})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $\mathcal{F} \subseteq 2^Q$: Muller-accepting state sets.

Muller Automata: Definition

Definition (Muller Automaton)

$\mathcal{A} = (Q, I, T, \mathcal{F})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $\mathcal{F} \subseteq 2^Q$: Muller-accepting state sets.

Muller Automata: Definition

Definition (Muller Automaton)

$\mathcal{A} = (Q, I, T, \mathcal{F})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $\mathcal{F} \subseteq 2^Q$: Muller-accepting state sets.

Run ρ Muller-accepted if
there is $F \in \mathcal{F}$ so that

- we see states in F infinitely often and
- states not in F only finitely often,

Muller Automata: Definition

Definition (Muller Automaton)

$\mathcal{A} = (Q, I, T, \mathcal{F})$ over alphabet Σ where

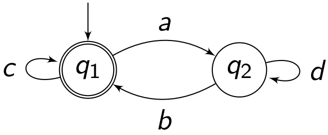
- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $\mathcal{F} \subseteq 2^Q$: Muller-accepting state sets.

Run ρ Muller-accepted if
there is $F \in \mathcal{F}$ so that

- we see states in F infinitely often and
- states not in F only finitely often,

i.e., $\mathrm{Inf}(\rho) \in \mathcal{F}$.

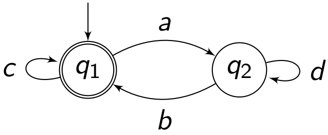
Muller Automata: Example



$$\mathcal{F} = \{\{q_1\}\}$$

Which ω-words are accepted? $(\mathsf{Inf}(\rho) \in \mathcal{F})$

Muller Automata: Example

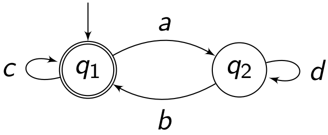


$$\mathcal{F} = \{\{q_1\}\}$$

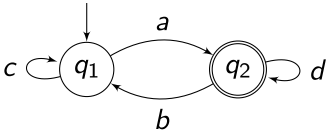
Which ω-words are accepted? $(\mathsf{Inf}(\rho) \in \mathcal{F})$

$$(c^* a d^* b)^* c^\omega$$

Muller Automata: Example



$$\mathcal{F} = \{\{q_1\}\}$$

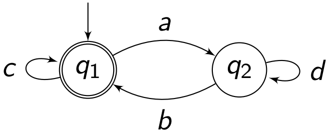


$$\mathcal{F} = \{\{q_2\}\}$$

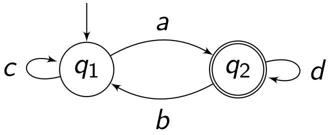
Which ω-words are accepted? $(\mathrm{Inf}(\rho) \in \mathcal{F})$

$$(c^* a d^* b)^* c^\omega$$

Muller Automata: Example



$$\mathcal{F} = \{\{q_1\}\}$$



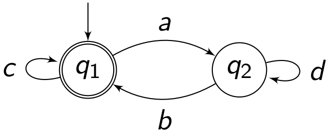
$$\mathcal{F} = \{\{q_2\}\}$$

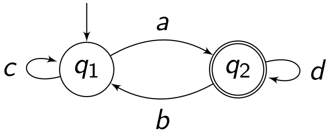
Which ω-words are accepted? $(\mathrm{Inf}(\rho) \in \mathcal{F})$

$$(c^* a d^* b)^* c^\omega$$

$$(c^* a d^* b)^* c^* a d^\omega$$

Muller Automata: Example





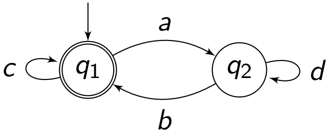
$\mathcal{F} = \{\{q_1\}\}$ $\qquad$ $\mathcal{F} = \{\{q_2\}\}$

Which ω-words are accepted? $\quad$ $(\mathrm{Inf}(\rho) \in \mathcal{F})$

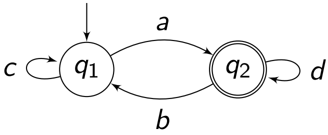
$(c^* ad^* b)^* c^\omega$ $\qquad$ $(c^* ad^* b)^* c^* ad^\omega$

$\mathcal{F} = \{\{q_1\}, \{q_2\}\} \quad \Rightarrow$

Muller Automata: Example



$$\mathcal{F} = \{\{q_1\}\}$$



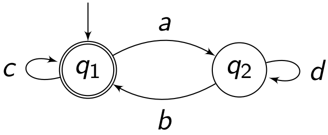
$$\mathcal{F} = \{\{q_2\}\}$$

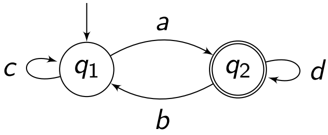
Which ω-words are accepted? $(\mathrm{Inf}(\rho) \in \mathcal{F})$

$$(c^* a d^* b)^* c^\omega \qquad (c^* a d^* b)^* c^* a d^\omega$$

$$\mathcal{F} = \{\{q_1\}, \{q_2\}\} \quad \Rightarrow \quad \mathcal{L}(\mathcal{A}) = (c^* a d^* b)^* c^\omega + (c^* a d^* b)^* c^* a d^\omega$$

Muller Automata: Example





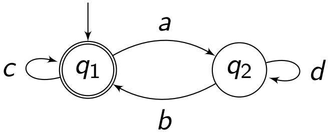
$\mathcal{F} = \{\{q_1\}\}$ $\qquad$ $\mathcal{F} = \{\{q_2\}\}$

Which ω-words are accepted? $\qquad$ $(\mathrm{Inf}(\rho) \in \mathcal{F})$

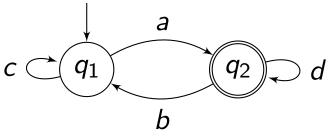
$(c^* ad^* b)^* c^\omega$ $\qquad$ $(c^* ad^* b)^* c^* ad^\omega$

$$\mathcal{F} = \{\{q_1\}, \{q_2\}\} \quad \Rightarrow \quad \mathcal{L}(\mathcal{A}) = (c^* ad^* b)^* c^\omega + (c^* ad^* b)^* c^* ad^\omega$$
$$\mathcal{F} = \{\{q_1, q_2\}\} \quad \Rightarrow$$

Muller Automata: Example



$\mathcal{F} = \{\{q_1\}\}$



$\mathcal{F} = \{\{q_2\}\}$

Which ω-words are accepted? $(\mathrm{Inf}(\rho) \in \mathcal{F})$

$(c^* a d^* b)^* c^\omega$ $(c^* a d^* b)^* c^* a d^\omega$

$$\mathcal{F} = \{\{q_1\}, \{q_2\}\} \quad \Rightarrow \quad \mathcal{L}(\mathcal{A}) = (c^* a d^* b)^* c^\omega + (c^* a d^* b)^* c^* a d^\omega$$
$$\mathcal{F} = \{\{q_1, q_2\}\} \quad \Rightarrow \quad \mathcal{L}(\mathcal{A}) = (c^* a d^* b)^\omega$$

Conversion Büchi to Muller

Easy direction:

Take Büchi automaton $\mathcal{B} = (Q, I, T, F)$ and transform it into Muller automaton $\mathcal{M} = (Q, I, T, \mathcal{F})$ with

$$\mathcal{F} = \{S \subseteq Q \mid S \cap F \neq \emptyset\}$$

Recall: Acceptance condition for Büchi: $\mathrm{Inf}(\rho) \cap F \neq \emptyset$

Conversion Muller to Büchi

Harder direction:

Take Muller automaton $\mathcal{M} = (Q, I, T, \mathcal{F})$.
Note that Büchi automata are closed under union.
Thus, assume $\mathcal{F} = \{F\}$ a singleton.

Conversion Muller to Büchi

Harder direction:

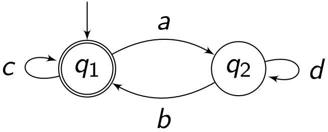
Take Muller automaton $\mathcal{M} = (Q, I, T, \mathcal{F})$.
Note that Büchi automata are closed under union.
Thus, assume $\mathcal{F} = \{F\}$ a singleton.

Transform $\mathcal{M}$ into Büchi automaton $\mathcal{B} = (Q', I, T', F')$ with:

$$
\begin{aligned}
Q' &= Q \cup (Q \times 2^F) \\
T' &= T \\
&\qquad \cup \{(q, \sigma, (q', \{q'\})) && \mid (q, \sigma, q') \in T, q' \in F\} \\
&\qquad \cup \{((q, S), \sigma, (q', S \cup \{q'\})) && \mid (q, \sigma, q') \in T, q, q' \in F, S \subsetneq F\} \\
&\qquad \cup \{((q, F), \sigma, (q', \{q'\})) && \mid (q, \sigma, q') \in T, q, q' \in F\} \\
F' &= F \times \{F\}
\end{aligned}
$$

Conversion Muller to Büchi: Example

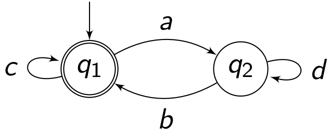
$\mathcal{M}$:



$$\mathcal{F} = \{F\} = \{\{q_1\}\}$$

Conversion Muller to Büchi: Example

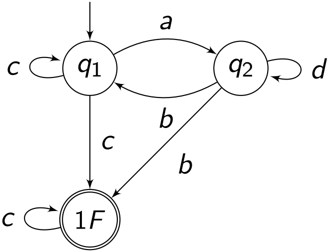
$\mathcal{M}$:



$$\mathcal{F} = \{F\} = \{\{q_1\}\}$$

translates to

$\mathcal{B}$:



From Büchi to Spot

Transition-Based Generalized Büchi Automata

Definition (Transition-Based Generalized Büchi Automata (TGBA))

$\mathcal{A} = (Q, I, T, \mathcal{M})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times 2^{\mathcal{M}} \times Q$: transitions,
- $\mathcal{M}$: finite set of acceptance conditions (colors).

Transition-Based Generalized Büchi Automata

Definition (Transition-Based Generalized Büchi Automata (TGBA))

$\mathcal{A} = (Q, I, T, \mathcal{M})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times 2^{\mathcal{M}} \times Q$: transitions,
- $\mathcal{M}$: finite set of acceptance conditions (colors).

Transition-Based Generalized Büchi Automata

Definition (Transition-Based Generalized Büchi Automata (TGBA))

$\mathcal{A} = (Q, I, T, \mathcal{M})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times 2^{\mathcal{M}} \times Q$: transitions,
- $\mathcal{M}$: finite set of acceptance conditions (colors).

Run $\rho = q_0 \xrightarrow{w_0}_{M_0} q_1 \xrightarrow{w_1}_{M_1} q_2 \to \dots$
is TGBA-accepted if *all colors are seen infinitely often*
i.e., $\forall m \in \mathcal{M}. \forall i \in \mathbb{N}. \exists j > i. m \in M_j$.

Automata Over Propositions

The following can be done for any automaton:

Take a (finite) set of atomic propositions AP.
Put $\Sigma = 2^{AP}$.

Automata Over Propositions

The following can be done for any automaton:

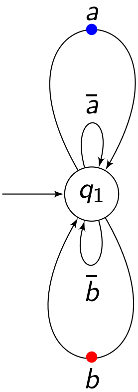
Take a (finite) set of atomic propositions AP.
Put $\Sigma = 2^{AP}$.

Note that we could interpret the set of minterms over AP as Σ.

Example

$$(a \wedge b \wedge \bar{c}); (a \wedge \bar{b} \wedge \bar{c}); (a \wedge \bar{b} \wedge c); \ldots$$

Transition-Based Generalized Büchi Automata: Example



Degeneralization

Lemma

TGBA can be degeneralized.

Construction is similar to Muller -> Büchi translation.

Product Construction

Let $A_1 = (Q_1, I_1, T_1, \mathcal{M}_1)$ and $A_2 = (Q_2, I_2, T_2, \mathcal{M}_2)$ be two TGBA over the same set of propositions Σ.

The product construction of A_1 and A_2 is $B = (Q, I, T, \mathcal{M})$ with

- $Q = Q_1 \times Q_2$
- $I = I_1 \times I_2$
- $T = \Big\{ ((q_1, q_2), p_1 \cap p_2, M_1 \cup M_2, (q_1', q_2')) \mid$ $(q_1, p_1, M_1, q_1') \in T_1, (q_2, p_2, M_2, q_2') \in T_2,\ p_1 \cap p_2 \neq \emptyset \Big\}$
- $\mathcal{M} = \mathcal{M}_1 \uplus \mathcal{M}_2$ (disjoint union)

Product Construction

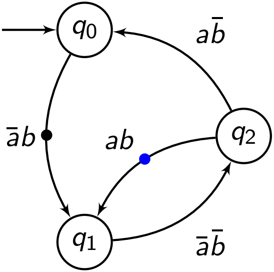
Let $A_1 = (Q_1, I_1, T_1, \mathcal{M}_1)$ and $A_2 = (Q_2, I_2, T_2, \mathcal{M}_2)$ be two TGBA over the same set of propositions Σ.

The product construction of A_1 and A_2 is $B = (Q, I, T, \mathcal{M})$ with

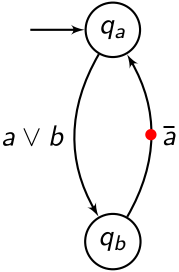
- $Q = Q_1 \times Q_2$
- $I = I_1 \times I_2$
- $T = \Big\{ ((q_1, q_2), p_1 \cap p_2, M_1 \cup M_2, (q_1', q_2')) \mid$
  $(q_1, p_1, M_1, q_1') \in T_1, (q_2, p_2, M_2, q_2') \in T_2,\ p_1 \cap p_2 \neq \emptyset \Big\}$
- $\mathcal{M} = \mathcal{M}_1 \uplus \mathcal{M}_2$ (disjoint union)

We can show: $\mathcal{L}(B) = \mathcal{L}(A_1) \cap \mathcal{L}(A_2)$.

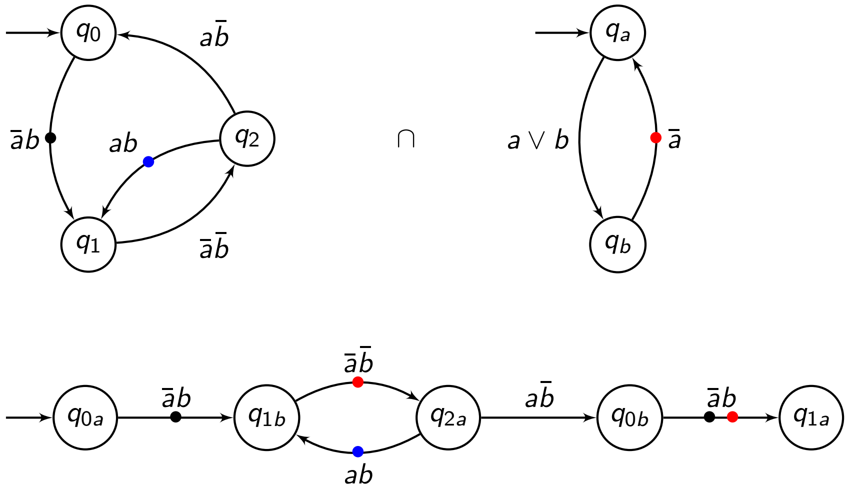
Product Construction: Example



$\cap$



Product Construction: Example



Emerson-Lei Automata

Description (Emerson-Lei Automaton)

- contains set of colors $\mathcal{M}$

  either states or transitions can be marked by colors
- acceptance condition: positive Boolean formula over the atoms
  - `t` or `f`: true and false
  - `Fin(m)`: color m occurs finitely many times
  - `Inf(m)`: color m occurs infinitely many times

Emerson-Lei Automata

Description (Emerson-Lei Automaton)

- contains set of colors $\mathcal{M}$

  either states or transitions can be marked by colors
- acceptance condition: positive Boolean formula over the atoms
  - `t` or `f`: true and false
  - `Fin(m)`: color *m* occurs finitely many times
  - `Inf(m)`: color *m* occurs infinitely many times

Emerson-Lei implemented in Spot (see TP in two weeks!)

Example + Further Acceptance Conditions

none	`f`
all	`t`
Buchi	`Inf(0)`
gen.-Buchi	`Inf(0) & Inf(1) & ...`
co-Buchi	`Fin(0)`
gen.-co-Buchi	`Fin(0) | Fin(1) | ...`
Rabin	`(Fin(0)&Inf(1)) | (Fin(2)&Inf(3)) | ...`
Streett	`(Fin(0)|Inf(1)) & (Fin(2)|Inf(3)) & ...`
parity min odd 5	`Fin(0)&(Inf(1)|(Fin(2)&(Inf(3)|Fin(4))))`

Summary: Expressive Power

Overview: Expressive Power in Σ^ω

- ω-rational expressions $= \omega$-regular languages
- **Büchi automata**
- GBA
- **TGBA**
- Muller automata
- Emerson-Lei automata

- **LTL**

- Deterministic Büchi automata (incl. GBA, TGBA)


That's all Folks!

Backup

Notes

Generalized Büchi Automata

Definition (Generalized Büchi Automata (GBA))

$\mathcal{A} = (Q, I, T, \mathcal{F})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $\mathcal{F} \subseteq 2^Q$: Büchi-accepting state sets.

Generalized Büchi Automata

Definition (Generalized Büchi Automata (GBA))

$\mathcal{A} = (Q, I, T, \mathcal{F})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $\mathcal{F} \subseteq 2^Q$: Büchi-accepting state sets.

Generalized Büchi Automata

Definition (Generalized Büchi Automata (GBA))

$\mathcal{A} = (Q, I, T, \mathcal{F})$ over alphabet Σ where

- Q: finite set of states,
- $I \subseteq Q$: initial states,
- $T \subseteq Q \times \Sigma \times Q$: transitions,
- $\mathcal{F} \subseteq 2^Q$: Büchi-accepting state sets.

Run ρ generalized-Büchi-accepted if
for all $F \in \mathcal{F}$, ρ is Büchi-accepted,
i.e., $\forall F \in \mathcal{F}. \mathrm{Inf}(\rho) \cap F \neq \emptyset$.

Degeneralization

Lemma

TGBA can be degeneralized.

Proof.

Let $\mathcal{A} = (Q, I, T, \mathcal{M})$ be a TGBA with $\mathcal{M} = \{m_1, \dots, m_k\}$.
We construct $\bar{\mathcal{A}} = (Q \times \{1, \dots, k\}, I \times \{1\}, \bar{T}, \{\bar{m}\})$ with

$$\begin{aligned} \bar{T} = {} & \{((q, i), \sigma, \emptyset, (q', i)) \mid (q, \sigma, M, q') \in T, m_i \notin M\} \\ & \cup \{((q, i), \sigma, \emptyset, (q', i+1)) \mid i \neq k, (q, \sigma, M, q') \in T, m_i \in M\} \\ & \cup \{((q, k), \sigma, \{m_a\}, (q', 1)) \mid (q, \sigma, M, q') \in T, m_k \in M\} \end{aligned}$$

Degeneralization

Lemma

TGBA can be degeneralized.

Proof.

Let $\mathcal{A} = (Q, I, T, \mathcal{M})$ be a TGBA with $\mathcal{M} = \{m_1, \dots, m_k\}$.
We construct $\bar{\mathcal{A}} = (Q \times \{1, \dots, k\}, I \times \{1\}, \bar{T}, \{\bar{m}\})$ with

$$\begin{aligned} \bar{T} = {} & \{((q,i), \sigma, \emptyset, (q', i)) \mid (q, \sigma, M, q') \in T, m_i \notin M\} \\ & \cup \{((q,i), \sigma, \emptyset, (q', i+1)) \mid i \neq k, (q, \sigma, M, q') \in T, m_i \in M\} \\ & \cup \{((q,k), \sigma, \{m_a\}, (q', 1)) \mid (q, \sigma, M, q') \in T, m_k \in M\} \end{aligned}$$

Degeneralization

Lemma

TGBA can be degeneralized.

Proof.

Let $\mathcal{A} = (Q, I, T, \mathcal{M})$ be a TGBA with $\mathcal{M} = \{m_1, \ldots, m_k\}$.
We construct $\bar{\mathcal{A}} = (Q \times \{1, \ldots, k\}, I \times \{1\}, \bar{T}, \{\bar{m}\})$ with

$$\begin{aligned}\bar{T} = &\{((q,i), \sigma, \emptyset, (q', i)) \mid (q, \sigma, M, q') \in T, m_i \notin M\} \\ &\cup \{((q,i), \sigma, \emptyset, (q', i+1)) \mid i \neq k, (q, \sigma, M, q') \in T, m_i \in M\} \\ &\cup \{((q,k), \sigma, \{m_a\}, (q', 1)) \mid (q, \sigma, M, q') \in T, m_k \in M\}\end{aligned}$$

Degeneralization

Lemma

TGBA can be degeneralized.

Proof.

Let $\mathcal{A} = (Q, I, T, \mathcal{M})$ be a TGBA with $\mathcal{M} = \{m_1, \dots, m_k\}$.
We construct $\bar{\mathcal{A}} = (Q \times \{1, \dots, k\}, I \times \{1\}, \bar{T}, \{\bar{m}\})$ with

$$\begin{aligned}\bar{T} = &\{((q, i), \sigma, \emptyset, (q', i)) \mid (q, \sigma, M, q') \in T, m_i \notin M\}\\ &\cup \{((q, i), \sigma, \emptyset, (q', i+1)) \mid i \neq k, (q, \sigma, M, q') \in T, m_i \in M\}\\ &\cup \{((q, k), \sigma, \{m_a\}, (q', 1)) \mid (q, \sigma, M, q') \in T, m_k \in M\}\end{aligned}$$

Degeneralization

Lemma

TGBA can be degeneralized.

Proof.

Let $\mathcal{A} = (Q, I, T, \mathcal{M})$ be a TGBA with $\mathcal{M} = \{m_1, \dots, m_k\}$.
We construct $\bar{\mathcal{A}} = (Q \times \{1, \dots, k\}, I \times \{1\}, \bar{T}, \{\bar{m}\})$ with

$$\begin{aligned}\bar{T} = &\{((q,i), \sigma, \emptyset, (q', i)) \mid (q, \sigma, M, q') \in T, m_i \notin M\} \\ &\cup \{((q,i), \sigma, \emptyset, (q', i+1)) \mid i \neq k, (q, \sigma, M, q') \in T, m_i \in M\} \\ &\cup \{((q,k), \sigma, \{m_a\}, (q', 1)) \mid (q, \sigma, M, q') \in T, m_k \in M\}\end{aligned}$$

Degeneralization

Lemma

TGBA can be degeneralized.

Proof.

Let $\mathcal{A} = (Q, I, T, \mathcal{M})$ be a TGBA with $\mathcal{M} = \{m_1, \ldots, m_k\}$.
We construct $\bar{\mathcal{A}} = (Q \times \{1, \ldots, k\}, I \times \{1\}, \bar{T}, \{\bar{m}\})$ with

$$\begin{aligned}\bar{T} = {} & \{((q,i), \sigma, \emptyset, (q', i)) \mid (q, \sigma, M, q') \in T, m_i \notin M\} \\ & \cup \{((q,i), \sigma, \emptyset, (q', i+1)) \mid i \neq k, (q, \sigma, M, q') \in T, m_i \in M\} \\ & \cup \{((q,k), \sigma, \{m_a\}, (q', 1)) \mid (q, \sigma, M, q') \in T, m_k \in M\}\end{aligned}$$