

Infinite Automata, Logics and Games

Angeliki Chalki

NTUA

May 24, 2018

ω -Automata

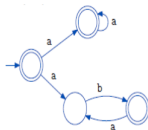
Tree Automata

Ehrenfeucht-Fraïssé Games

A nondeterministic finite automaton (*NFA*) is a quintuple, $(Q, \Sigma, \delta, q_0, F)$, consisting of

- ▶ a finite set of states Q ,
- ▶ a finite set of input symbols Σ ,
- ▶ a transition function $\delta : Q \times \Sigma \rightarrow Pow(Q)$,
- ▶ an initial state $q_0 \in Q$,
- ▶ a set of states F distinguished as accepting (or final) states $F \subseteq Q$.

NFA for $a^* + (ab)^*$:



REG is the class of languages recognised by a finite automaton.

An ω -automaton is a quintuple $(Q, \Sigma, \delta, q_0, Acc)$, where

- ▶ Q is a finite set of states,
- ▶ Σ is a finite alphabet,
- ▶ $\delta : Q \times \Sigma \rightarrow Pow(Q)$ is the state transition function,
- ▶ $q_0 \in Q$ is the initial state,
- ▶ Acc is the acceptance component (this corresponds to F in the case of finite automata).

In a deterministic ω -automaton, a transition function $\delta : Q \times \Sigma \rightarrow Q$ is used.

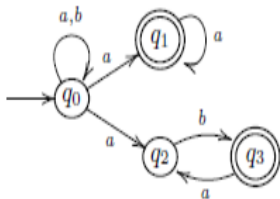
Let $A = (Q, \Sigma, \delta, q_0, Acc)$ be an ω -automaton. A run of A on an ω -word (stream) $\alpha = a_1 a_2 \dots \in \Sigma^\omega$ is a countable infinite state sequence $\rho = \rho(0)\rho(1)\rho(2)\dots \in Q^\omega$, such that the following conditions hold:

1. $\rho(0) = q_0$
2. $\rho(i) \in \delta(\rho(i-1), a_i)$ for $i \geq 1$ if A is nondeterministic,

For a run ρ of an ω -automaton, let $Inf(\rho) = \{q \in Q : \forall i \exists j > i \rho(j) = q\}$ (i.e. the set of states visited infinitely often).

An ω -automaton $A = (Q, \Sigma, \delta, q_0, Acc)$ is called

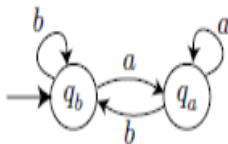
- **Büchi** automaton if $Acc = F \subseteq Q$ and the acceptance condition is the following: A stream $\alpha \in \Sigma^\omega$ is accepted by A iff there exists a run ρ of A on α satisfying the condition: $Inf(\rho) \cap F \neq \emptyset$.



Büchi automaton for $(a + b)^* a^\omega + (a + b)^* (ab)^\omega$ with $F = \{q_1, q_3\}$

An ω -automaton $A = (Q, \Sigma, \delta, q_0, Acc)$ is called

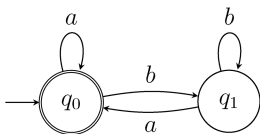
- **Muller** automaton if $Acc = \mathcal{F} \subseteq Pow(Q)$ and the acceptance condition is the following: A stream $\alpha \in \Sigma^\omega$ is accepted by A iff there exists a run ρ of A on α satisfying the condition: $Inf(\rho) \in \mathcal{F}$.



Muller automaton for $(a + b)^* a^\omega + (a + b)^* b^\omega$ with $\mathcal{F} = \{\{q_a\}, \{q_b\}\}$

An ω -automaton $A = (Q, \Sigma, \delta, q_0, Acc)$ is called

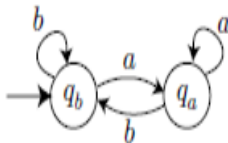
- **Rabin** automaton if $Acc = \{(E_1, F_1), \dots, (E_k, F_k)\}$, with $E_i, F_i \subseteq Q$, $1 \leq i \leq k$, and the acceptance condition is the following: A stream $\alpha \in \Sigma^\omega$ is accepted by A iff there exists a run ρ of A on α satisfying the condition:
 $\exists (E, F) \in Acc (Inf(\rho) \cap E = \emptyset) \wedge (Inf(\rho) \cap F \neq \emptyset)$.



Rabin automaton for $(a + b)^* a^\omega$ with $Acc = \{(\{q_1\}, \{q_0\})\}$

An ω -automaton $A = (Q, \Sigma, \delta, q_0, Acc)$ is called

- **Streett** automaton if $Acc = \{(E_1, F_1), \dots, (E_k, F_k)\}$, with $E_i, F_i \subseteq Q$, $1 \leq i \leq k$, and the acceptance condition is the following: A stream $\alpha \in \Sigma^\omega$ is accepted by A iff there exists a run ρ of A on α satisfying the condition:
 $\neg(\exists(E, F) \in Acc(\text{Inf}(\rho) \cap E = \emptyset) \wedge (\text{Inf}(\rho) \cap F \neq \emptyset))$, i.e.
 $\forall(E, F) \in Acc(\text{Inf}(\rho) \cap E \neq \emptyset) \vee (\text{Inf}(\rho) \cap F = \emptyset)$ (or
 $\forall(E, F) \in Acc(\text{Inf}(\rho) \cap F \neq \emptyset) \rightarrow (\text{Inf}(\rho) \cap E \neq \emptyset)$).



Streett automaton with $Acc = \{(\{q_b\}, \{q_a\})\}$.

Each stream in the accepted language contains infinitely many a 's only if it contains infinitely many b 's (or equivalently they have finitely many a 's or infinitely many b 's), e.g. $(a + b)^* b^\omega + (a^* b)^\omega$

The Büchi recognizable ω -languages are the ω -languages of the form

$$L = U_1 V_1^\omega + U_2 V_2^\omega \dots U_k V_k^\omega \text{ with } k \in \omega \text{ and } U_i, V_i \in \text{REG for } i = 1, \dots, k.$$

This family of ω -languages is also called the ω -**Kleene closure** of the class of regular languages and is commonly referred to as ω -REG.

The **emptiness problem** for Büchi automata is decidable.

Muller automata are equally expressive as nondeterministic Büchi automata.

Proof: On the board.

Rabin automata and Streett automata are equally expressive as Muller automata.

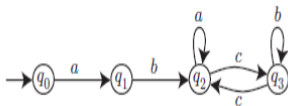
Proof:

- For a Rabin automaton $A = (Q, \Sigma, \delta, q_0, Acc)$, define the Muller automaton $A' = (Q, \Sigma, \delta, q_0, \mathcal{F})$, where $\mathcal{F} = \{G \in Pow(Q) \mid \exists (E, F) \in Acc. G \cap E = \emptyset \wedge G \cap F \neq \emptyset\}$.
For a Streett automaton $A = (Q, \Sigma, \delta, q_0, Acc)$, define the Muller automaton $A' = (Q, \Sigma, \delta, q_0, \mathcal{F})$, where $\mathcal{F} = \{G \in Pow(Q) \mid \forall (E, F) \in Acc. G \cap E \neq \emptyset \vee G \cap F = \emptyset\}$.
- Conversely, given a Muller automaton, transform it into a nondeterministic Büchi automaton.
Büchi acceptance can be viewed as a special case of Rabin acceptance, where $Acc = \{(\emptyset, F)\}$, as well as a special case of Streett acceptance, where $Acc = \{(F, Q)\}$.

An ω -automaton $A = (Q, \Sigma, \delta, q_0, c)$ with acceptance component $c : Q \rightarrow \{1, \dots, k\}$ (where $k \in \omega$) is called **parity** automaton if it is used with the following acceptance condition:

A stream $\alpha \in \Sigma^\omega$ is accepted by A iff there exists a run ρ of A on α with

$$\min\{c(q) \mid q \in \text{Inf}(\rho)\} \text{ is even}$$



Parity automaton A with colouring function c defined by $c(q_i) = i$.

$$L(A) = ab(a^*cb^*c)^*a^\omega$$

Parity automata can be converted into Rabin automata.

Proof: Let $A = (Q, \Sigma, \delta, q_0, c)$ be a parity automaton with $c : Q \rightarrow \{0, \dots, k\}$. An equivalent Rabin automaton $A' = (Q, \Sigma, \delta, q_0, Acc)$ has the acceptance component $Acc = \{(E_0, F_0), \dots, (E_r, F_r)\}$, $r = \lfloor \frac{k}{2} \rfloor$,
 $E_i = \{q \in Q | c(q) < 2i\}$ and $F_i = \{q \in Q | c(q) \leq 2i\}$.

Muller automata can be converted into parity automata (a special case of Rabin automata).

Proof: On the board.

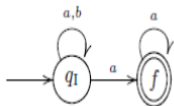
- ▶ Nondeterministic Büchi, Muller, Rabin, Streett, and parity automata are all equivalent in expressive power, i.e. they recognize the same ω -languages.
- ▶ The ω -languages recognized by these ω -automata form the class ω -KC(REG), i.e. the ω -Kleene closure of the class of regular languages.

- NFAs are equivalent to DFAs.
- NPDAs are not equivalent to DPDAs.
- Nondeterministic ω -automata are equivalent to deterministic ones?

Deterministic vs Nondeterministic Büchi Automata

There exist languages which are accepted by some nondeterministic Büchi-automaton but not by any deterministic Büchi automaton.

Proof. The following automaton is a nondeterministic Büchi automaton for $L = (a + b)^* a^\omega$.



Assume that there is a deterministic Büchi automaton A for the language L . Then there exist n_0, n_1, n_2, \dots such that A accepts the stream $w = a^{n_0} b a^{n_1} b a^{n_2} b \dots \notin L$.

- ▶ Deterministic Muller, Rabin, Streett, and parity automata recognize the same ω -languages.
- ▶ The class of ω -languages recognized by any of these types of ω -automata is closed under complementation.

Proof:

- ▶ The transformations between nondeterministic automata work for deterministic ones except for those that use nondeterministic Büchi automata.

NRabin \longrightarrow **NStreett**: NRabin \longrightarrow NMuller \longrightarrow NBüchi \longrightarrow NStreett

DRabin \longrightarrow **DStreett**: DRabin for L \longrightarrow DMuller for L \longrightarrow DMuller for \bar{L}
 \longrightarrow DRabin for \bar{L} \longrightarrow DStreett for L

- ▶ The languages recognizable by deterministic Muller automata are closed under union, intersection and complementation.

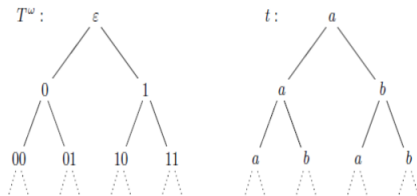
$DMuller = DRabin = DStreett = NBuchi = NMuller = NRabin = NStreett$
↑
 $DBuchi$

Determinization of Büchi Automata

Every nondeterministic Büchi automaton can be transformed into an equivalent deterministic Muller automaton (or a deterministic Rabin automaton).

- ▶ The powerset construction fails in case of Büchi automata.
 - ▶ Muller ('63) presented a faulty construction.
 - ▶ McNaughton ('66) showed that a Büchi automaton can be transformed effectively into an equivalent deterministic Muller automaton.
-
- ▶ Safra's construction ('88) leads to deterministic Rabin or Muller automata: given a nondeterministic Büchi automaton with n states, the equivalent deterministic automaton has $2^{O(n \log n)}$ states.
 - ▶ For Rabin automata, Safra's construction is optimal. The question whether it can be improved for Muller automata is open.
 - ▶ Muller and Schupp ('95) presented a 'more intuitive' alternative, which is also optimal for Rabin automata.

- ▶ The **infinite binary tree** T^ω is the set $\{0, 1\}^*$ of all strings on $\{0, 1\}$.
- ▶ The elements $u \in T^\omega$ are the **nodes** of T^ω where ϵ is the root and $u0, u1$ are the immediate left and right successors of node u .
- ▶ A stream $\pi \in \{0, 1\}^\omega$ is called a **path** of the binary tree T^ω .
- ▶ The set of all **Σ -labelled trees**, T_Σ^ω , contains trees where each node is labelled with a symbol of the alphabet Σ , i.e. trees with a mapping $t : T^\omega \rightarrow \Sigma$.



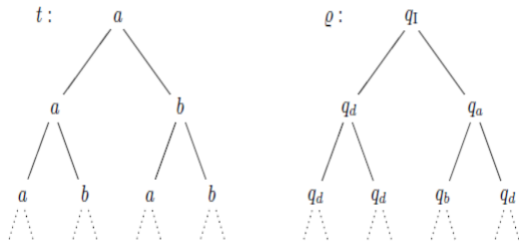
A **Muller tree automaton** is a quintuple $A = (Q, \Sigma, \delta, q_0, \mathcal{F})$, where

- ▶ Q is a finite set of states ,
- ▶ Σ is a finite alphabet,
- ▶ $\delta : Q \times \Sigma \rightarrow Pow(Q \times Q)$ denotes the transition relation,
- ▶ q_0 is an initial state,
- ▶ $\mathcal{F} \subseteq Pow(Q)$ is a set of designated state sets.

- ▶ A **run** of A on an input tree $t \in T_\Sigma$ is a tree $\rho \in T_Q$, satisfying $\rho(\epsilon) = q_0$ and for all $w \in \{0, 1\}^*$: $\delta(\rho(w), t(w)) = (\rho(w0), \rho(w1))$.
- ▶ A run is called **successful** if for each path $\pi \in \{0, 1\}^\omega$ the Muller acceptance condition is satisfied, that is, if $Inf(\rho|\pi) \in \mathcal{F}$.
- ▶ A accepts the tree t if there is a successful run of A on t .
- ▶ The tree language recognized by A is the set $T(A) = \{t \in T^\omega \mid A \text{ accepts } t\}$.

Example: $A = (\{q_0, q_a, q_b, q_d\}, \{a, b\}, \delta, q_0, \mathcal{F})$, where δ includes:

$$\begin{aligned} \delta(q_0, a) &= (q_a, q_d), \delta(q_0, a) = (q_d, q_a), \delta(q_0, b) = (q_b, q_d), \delta(q_0, b) = (q_d, q_b), \\ &\delta(q_d, a) = (q_d, q_d), \delta(q_d, b) = (q_d, q_d), \\ \delta(q_a, b) &= (q_b, q_d), \delta(q_a, b) = (q_d, q_b), \delta(q_a, a) = (q_0, q_d), \delta(q_a, a) = (q_d, q_0), \\ \delta(q_b, a) &= (q_a, q_d), \delta(q_b, a) = (q_d, q_a), \delta(q_b, b) = (q_0, q_d), \delta(q_b, b) = (q_d, q_0). \end{aligned}$$



First transitions of ρ

Example: The Muller tree automaton $A = (\{q_0, q_a, q_b, q_d\}, \{a, b\}, \delta, q_0, \mathcal{F})$, where δ includes:

$$\begin{aligned} \delta(q_0, a) &= (q_a, q_d), \delta(q_0, b) = (q_d, q_b), \\ \delta(q_a, a) &= (q_d, q_a), \delta(q_a, b) = (q_d, q_d), \\ \delta(q_b, a) &= (q_a, q_d), \delta(q_b, b) = (q_0, q_d), \\ \delta(q_d, a) &= (q_d, q_d), \delta(q_d, b) = (q_d, q_d), \\ \delta(q_a, a) &= (q_0, q_d), \delta(q_a, b) = (q_d, q_0), \\ \delta(q_b, a) &= (q_a, q_d), \delta(q_b, b) = (q_0, q_d), \delta(q_b, b) = (q_d, q_0). \end{aligned}$$

and $\mathcal{F} = \{\{q_a, q_b\}, \{q_d\}\}$ recognizes the tree language

$$T = \{t \in T_{\{a,b\}} \mid \text{there is a path } \pi \text{ through } t \text{ such that } t|_{\pi} \in (a+b)^*(ab)^\omega\}.$$

Example: The Muller tree automaton $A = (\{q_0, q_1, q_2\}, \{a, b\}, \delta, q_0, \{\{q_0\}\})$, where δ includes the transitions:

$$\begin{aligned}\delta(q_0, a) &= (q_0, q_0), \delta(q_0, b) = (q_1, q_1), \\ \delta(q_1, b) &= (q_1, q_1), \delta(q_1, a) = (q_0, q_0).\end{aligned}$$

recognizes the tree language

$T = \{t \in T_{\{a,b\}} \mid \text{any path through } t \text{ carries only finitely many } b's\}$.

The above language T can not be recognized by a Büchi tree automaton.

Büchi tree automata are strictly weaker than Muller tree automata.

Muller, Rabin, Streett, and parity tree automata all recognize the same tree languages.