Elementary Automata Theory

Bow-Yaw Wang

Institute of Information Science
Academia Sinica, Taiwan

July 1, 2009

Outline

Finite Automata

- A finite automaton is a 5-tuple $(Q, \Sigma, \delta, q_0, F)$ where
  - Q is a finite set of states;
  - Σ is a finite input alphabet;
  - $\delta \subseteq Q \times \Sigma \times Q$ is a transition relation;
  - $q_0 \in Q$ is the initial state;
  - $F \subseteq Q$ is a set of accepting states.
- If the transition relation is in fact a function from $Q \times \Sigma$ to Q, it is a deterministic finite automaton (DFA). Otherwise, it is a non-deterministic finite automaton (NFA).

Example

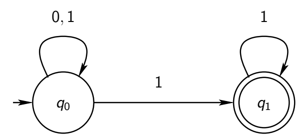


Figure: NFA M_0

- $M_0 = (Q, \Sigma, \delta, q_0, F)$ where
  - $Q = \{q_0, q_1\}$;
  - $\Sigma = \{0, 1\}$;
  - $\delta = \{(q_0, 0, q_0), (q_0, 1, q_0), (q_0, 1, q_1), (q_1, 1, q_1)\}$;
  - $F = \{q_1\}$.

Input Sequences and Runs

- Let $M = (Q, \Sigma, \delta, q_0, F)$ be an NFA.
- An input sequence $\alpha = a_1 a_2 \cdots a_n$ is a finite sequence of symbols over the alphabet Σ.
  - The finite sequence without any symbol is denoted by ϵ.
- A run $\rho = q_0 q_1 \cdots q_{n+1}$ on an input sequence $\alpha = a_1 a_2 \cdots a_n$ is a sequence of states such that

$$\text{for all } 0 \leq i < n, (q_i, a_{i+1}, q_{i+1}) \in \delta.$$

- A run $\rho = q_0 q_1 \cdots q_{n+1}$ of M over $\alpha = a_1 a_2 \cdots a_n$ is accepting if $q_{n+1} \in F$.
- An input sequence α is accepted by M if there is an accepting run ρ of M over α.

Example (cont'd)

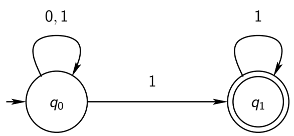


Figure: NFA M_0

- For the input sequence 0000, there is only one run $q_0 q_0 q_0 q_0 q_0$.
  - 0000 is not accepted by M_0.
- For the input sequence 0011, there are three possible runs:
  - $q_0 q_0 q_0 q_0 q_0$, $q_0 q_0 q_0 q_0 q_1$, and $q_0 q_0 q_0 q_1 q_1$.
  - the dark green ones are accepting.
  - 0011 is accepted by M_0.

Languages

- Given an alphabet Σ, a language is a set of input sequences over Σ.
- Let $M = (Q, \Sigma, \delta, q_0, F)$ be an NFA. Define

$$L(M) = \{\alpha : \alpha \text{ is an input sequence accepted by } M\}.$$

- $L(M)$ is the language accepted (or recognized) by M.
- Thus,

$$\begin{array}{rcl} L(M_0) & = & \{1, 01, 11, 001, 011, 111, \ldots\} \\ & = & \{\alpha : \text{ the last symbol of } \alpha \text{ is } 1\}. \end{array}$$

Expressive Power

- Let M be a DFA. Since a DFA is also an NFA, the language $L(M)$ is accepted by an NFA as well.
- Let N be an NFA. We will prove that $L(N)$ can be accepted by a DFA.
- In other words, nondeterminism does not recognize more languages. For finite automata, it suffces to consider deterministic fintie automata.

Subset Construction

Theorem

Let L be a language accepted by an NFA. Then there is a DFA M such that $L(M) = L$.

Proof.

Let $N = (Q, \Sigma, \delta, q_0, F)$ be an NFA and $L(N) = L$.
Consider $M = (2^Q, \Sigma, \delta', \{q_0\}, F')$ where

- $\delta'(X, a) = \bigcup_{x \in X} \delta(x, a)$;
- $F' = \{X \subseteq Q : X \cap F \neq \emptyset\}$.

We can show that $L(N) = L(M)$ by induction on the length of input sequences. □

Example

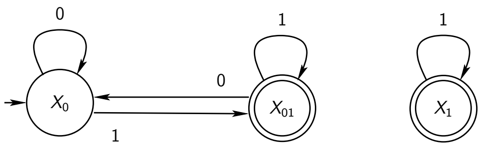


Figure: DFA M_1

- Let us find a DFA M_1 such that $L(M_1) = L(M_0)$.
- $M_1 = (Q', \Sigma, \delta', \{q_0\}, F')$ where
  - $Q' = \{X_\emptyset, X_0, X_1, X_{01}\}$ where

$X_\emptyset$	X_0	X_1	X_{01}
$\emptyset$	$\{q_0\}$	$\{q_1\}$	$\{q_0, q_1\}$

  - $\delta' = \{(X_0, 0, X_0), (X_0, 1, X_{01}), (X_1, 1, X_1), (X_{01}, 0, X_0), (X_{01}, 1, X_{01})\}$;
  - $F' = \{X_1, X_{01}\}$.

Operations on Languages

- Let Σ be a finite alphabet, and L, L_0, L_1 be languages over Σ.
- The concatenation of L_0 and L_1 (denoted by L_0L_1) is defined by

$$L_0L_1 = \{\alpha\beta : \alpha \in L_0, \beta \in L_1\}.$$

- Define $L^0 = \{\epsilon\}$ and $L^i = LL^{i-1}$ for $i \geq 1$.
- The Kleene closure (or just closure) of L (denoted by L^*) is defined by

$$L^* = \bigcup_{i=0}^{\infty} L^i.$$

- The positive closure of L (denoted by L^+) is defined by

$$L^+ = \bigcup_{i=1}^{\infty} L^i.$$

Regular Expressions

- Let Σ be an alphabet. The regular expressions over Σ are defined as follows.
  1. $\emptyset$ is a regular expression denoting the empty set;
  2. ϵ is a regular expression denoting the set $\{\epsilon\}$;
  3. For each $a \in \Sigma$, a is a regular expression denoting the set $\{a\}$;
  4. If r and s are regular expressions denoting the sets R and S respectively, then $r + s$, rs, and r^* are regular expressions denoting $R \cup S$, RS, and R^* respectively.

Example

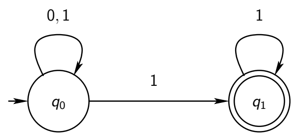


Figure: NFA M_0

- Let $\Sigma = \{0, 1\}$. $L_0 = \{\epsilon, 00\}$ and $L_1 = \{1, 111\}$.
  - $L_0 L_1 = \{1, 111, 001, 00111\}$;
  - $L_0^+ = \{\epsilon, 00, 0000, \ldots\} = \{0^{2i} : i \geq 0\}$;
  - $L_1^* = \{\epsilon, 1, 11, 111, \ldots\} = \{1^i : i \geq 0\}$.
- Also note that $L_0 \subseteq \Sigma^*$ and $L_1 \subseteq \Sigma^*$.
  - Thus, a language is a subset of Σ^*.
- We have $L(M_0) = (0+1)^* 1^+$

NFA with ϵ-Transitions

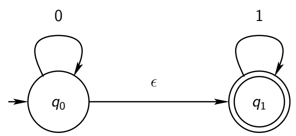


Figure: NFA M_2

- Since $\epsilon \notin \Sigma$, we do not allow, for example, (p, ϵ, q) in the transition relation of finite automata.
- A transition with ϵ as its input symbol is called an ϵ-transition.
  - Intuitively, it represents that the finite automaton can move to another state without consuming any input symbol.
- Consider the NFA M_2. We have $L(M_2) = 0^* 1^*$.

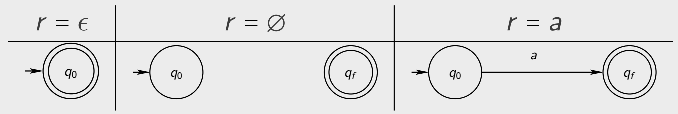
Regular Expressions to NFA with ϵ-Transitions

Theorem

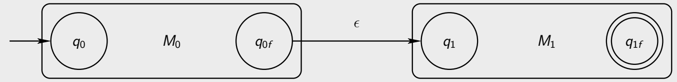
Let r be a regular expression. There is an NFA with ϵ-transition that accepts the language denoted by r.

Proof.

We prove by induction on the r. For the basis, see the following.



For the inductive step, first consider $r = st$. We use

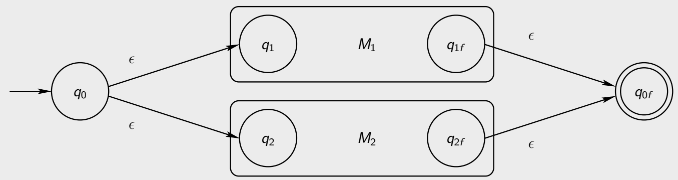


(assuming a single acceptance state q_{0f})

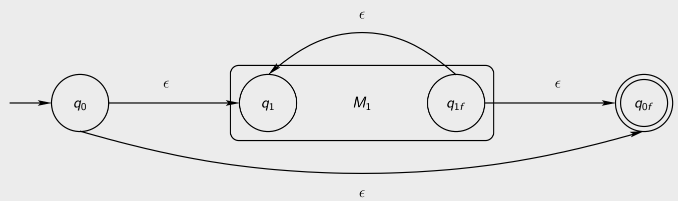
Regular Expressions to NFA with ϵ-Transitions (cont'd)

Proof (cont'd).

For $r = s + t$, we use



Finally, for $r = s^*$, we use



NFA with ϵ-Transitions to DFA

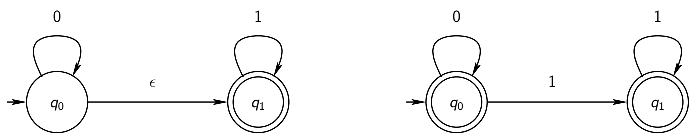


Figure: NFA M_2 to M_3 without ϵ-transition

- It is actually not difficult to see that ϵ-transitions can be removed.
  - The idea is to simulate ϵ-transitions by consuming input symbols.
- We will not give a proof but only consider an example.
- In general, removing ϵ-transitions will result in an NFA.
- We can futher transform an NFA to a DFA.

DFA to Regular Expressions

Theorem

Let D be a DFA. There is a regular expression denoting $L(D)$.

Proof.

Let $D = (\{q_1, \ldots, q_n\}, \Sigma, \delta, q_1, F)$ be a DFA. Define

$$\begin{array}{rcl} R_{ij}^0 & = & \begin{cases} \{a : (q_i, a, q_j) \in \delta\} & \text{if } i \neq j \\ \{a : (q_i, a, q_j) \in \delta\} \cup \{\epsilon\} & \text{if } i = j \end{cases} \\ R_{ij}^k & = & R_{ik}^{k-1}(R_{kk}^{k-1})^* R_{kj}^{k-1} \cup R_{ij}^{k-1} \end{array}$$

Intuitively, R_{ij}^k represents the inputs that cause D to go from q_i to q_j without passing through a state higher than q_k. It is not hard to see that R_{ij}^k can be denoted by regular expressions.
The result follows by observing that $L(D) = \bigcup_{q_j \in F} R_{1j}^n$. □

Example

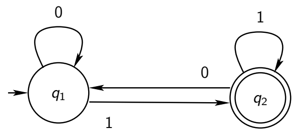


Figure: DFA M_4

	$k = 0$	$k = 1$	$k = 2$
R_{11}^k	0	0^+	
R_{12}^k	1	0^*1	$0^*1(0^*1)^*0^*1 + 0^*1 = (0+1)^*1$
R_{21}^k	0	0^+	
R_{22}^k	1	0^*1	

Regular Languages

- The class $\mathcal{R}$ of regular langauges consists of langauges accepted by deterministic finite automata.

$$\mathcal{R} = \{L(D) : D \text{ is a DFA }\}$$

- Since each NFA can be transformed to a DFA, we have

$$\mathcal{R} = \{L(M) : M \text{ is an NFA }\}$$

- Since each regular expression can be transformed to an NFA, we have

$$\mathcal{R} = \{L(e) : e \text{ is a regular expression }\}$$

Closure Properties

- For any $L_0, L_1 \in \mathcal{R}$, there are regular expressions r_0 and r_1 denoting L_0 and L_1 respectively.
- Moreover, the regular expression $r_0 + r_1$ denotes $L_0 \cup L_1$ and is accepted by an NFA.
- Thus $L_0 \cup L_1 \in \mathcal{R}$ for any $L_0, L_1 \in \mathcal{R}$.
- Similarly, we can prove that
  - $L_0 L_1 \in \mathcal{R}$ for any $L_0, L_1 \in \mathcal{R}$, and
  - $L^* \in \mathcal{R}$ for any $L \in \mathcal{R}$.

Closure Properties (cont'd)

Theorem

For any $L \in \mathcal{R}$, $\Sigma^* \setminus L \in \mathcal{R}$.

Proof.

Let $D = (Q, \Sigma, \delta, q_0, F)$ be a DFA and $L = L(D)$. Then $D' = (Q, \Sigma, \delta, q_0, Q \setminus F)$ accepts the language $\Sigma^* \setminus L$. □

Theorem

For any $L_0, L_1 \in \mathcal{R}$, $L_0 \cap L_1 \in \mathcal{R}$.

Proof.

Observe that $L_0 \cap L_1 = \Sigma^* \setminus ((\Sigma^* \setminus L_0) \cup (\Sigma^* \setminus L_1))$. □

ω-Automata

- We would like to generalize inputs to finite automata.
- Instead of finite input sequences, let us consider an infinite input sequence $\alpha = a_1 a_2 \cdots a_n \cdots$ over Σ.
- Let $M = (Q, \Sigma, \delta, q_0, F)$ be a finite automaton.
- As before, define a run $\rho = q_0 q_1 \cdots q_n \cdots$ on α to be an infinite sequence of states such that

$$\text{for all } i \geq 0, (q_i, a_{i+1}, q_{i+1}) \in \delta.$$

- What is an accepting run then?
  - Problem: there is no "final" state in an infinite run.
  - We cannot reuse the old definition.

Büchi Acceptance

- Let $\rho = q_0 q_1 \cdots q_n \cdots$ be an infinite run.
- Define

$$\text{Inf}(\rho) = \{q \in Q : q \text{ occurs infinitely many times in } \rho\}.$$

- An infinite run ρ of $M = (Q, \Sigma, \delta, q_0, F)$ over α is accepting if $\text{Inf}(\rho) \cap F \neq \emptyset$.
  - This is called Büchi acceptance
- An infinite input sequence α is accepted by M if there is an accepting infinite run ρ of M over α.
- Finally, define

$$L_\omega(M) = \{\alpha : \alpha \text{ is an infinite input sequence accepted by } M\}.$$

Example

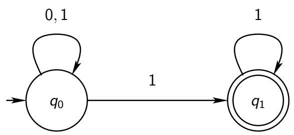


Figure: NFA M_0

- Let us reconsider M_0.
- $L_\omega(M_0) = \{\alpha : \alpha$ has only finitely many 0's$\}$.
  - If there are infintiely many 0's, M_0 has to stay in q_0. It cannot pass q_1 infinitely many times.
- We will write the expression $(0+1)^*1^\omega$ to denote $L(M_0)$.

Nondeterminism

- For finite automata over finite input sequences, we know nondeterminism does not give us more expressive power.
- However, nondeterministic finite automata with Büchi acceptance over infinite input sequences can recognize more languages than deterministic ones.

Theorem

$(0+1)^*1^\omega$ *cannot be accepted by any DFA with Büchi acceptance.*

Proof.

Suppose $D = (Q, \Sigma, \delta, q_0, F)$ is a DFA and $L(D) = (0+1)^*1^\omega$. Consider 1^ω. There is n_0 such that 1^{n_0} causes D to reach an accepting state. Now consider $1^{n_0}01^\omega$. There is n_1 such that $1^{n_0}01^{n_1}$ causes D to reach an accepting state. We can therefore construct $1^{n_0}01^{n_1}01^{n_2}0\cdots$ to cause D to pass through F infinitely many times. A contradiction. □

Remark

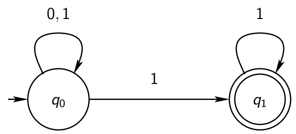


Figure: NFA M_0

- The proof does not work for NFA.
- Consider again the NFA M_0.
- 1 causes M_0 to reach q_1. 101 causes M_0 to reach q_1, etc. There is no problem.
- However, 101 passes q_1 only once. Similarly, 10101, 1010101, ... pass q_1 only once.
- Because M_0 is nondeterministic, infinite runs may not be the "limit" of their finite prefixes.

The Class of Regular ω-Languages

- Define

$$\mathcal{R}_\omega = \{L_\omega(M) : M \text{ is an NFA with Büchi acceptance }\}.$$

- $\mathcal{R}_\omega$ is called the class of regular ω-languages.
- Under Büchi acceptance, nondeterminism increases the expressive power. We have

$$\{L_\omega(D) : D \text{ is a DFA with Büchi acceptance }\} \subsetneq \mathcal{R}_\omega.$$

- In addition to Büchi acceptance, we will discuss three different acceptances.

Muller Acceptance

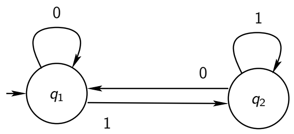


Figure: DFA M_5

- Let $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ be a finite automaton with $\mathcal{F} \subseteq 2^Q$.
- An infinite run ρ over an input sequence α on M is accepting if $\text{Inf}(\rho) \in \mathcal{F}$.
  - This is called Muller acceptance.
- Consider the DFA M_5 with $\mathcal{F} = \{\{q_2\}\}$.
- With Muller acceptance, we have $L_\omega(M_5) = (0+1)^*1^\omega$.
  - Note that M_5 is deterministic
  - Also note that $(01)^\omega$ is not accepted with Muller acceptance.

Rabin Acceptance

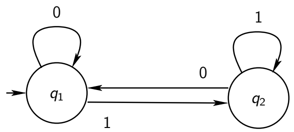


Figure: DFA M_5

- Let $M = (Q, \Sigma, \delta, q_0, \Omega)$ be a finite automaton with $\Omega = \{(E_0, F_0), \ldots, (E_k, F_k)\}$ and $E_i, F_i \subseteq Q$.
- An infinite run ρ over an input sequence α on M is accepting if

$$\exists (E, F) \in \Omega \text{ such that } \text{Inf}(\rho) \cap E = \varnothing \text{ and } \text{Inf}(\rho) \cap F \neq \varnothing.$$

- Consider the DFA M_5 with $\Omega = \{(\{q_1\}, \{q_2\})\}$.
- With Rabin acceptance, we have $L_\omega(M_5) = (0+1)^*1^\omega$.
  - $\text{Inf}(\rho) \cap \{q_1\} = \varnothing$ forbids 0 to occur infinitely many times.
  - $\text{Inf}(\rho) \cap \{q_2\} \neq \varnothing$ forces 1 to occur infinitely many times.

Streett Acceptance

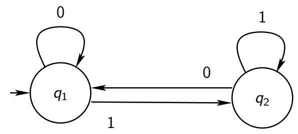


Figure: DFA M_5

- Let $M = (Q, \Sigma, \delta, q_0, \Omega)$ be a finite automaton with $\Omega = \{(E_0, F_0), \ldots, (E_k, F_k)\}$ and $E_i, F_i \subseteq Q$.
- An infinite run ρ over an input sequence α on M is accepting if

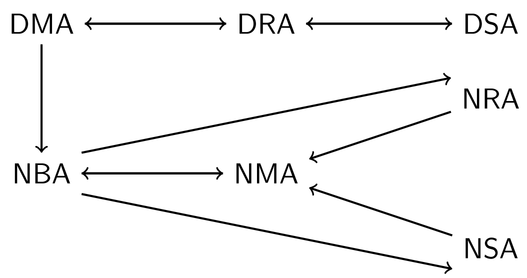
$$\forall (E, F) \in \Omega, \text{Inf}(\rho) \cap E \neq \varnothing \text{ or } \text{Inf}(\rho) \cap F = \varnothing.$$

- Observe that Rabin acceptance and Streett acceptance are complementary.
- Consider the DFA M_5 with $\Omega = \{(\{q_2\}, \{q_1, q_2\}), (\varnothing, \{q_1\})\}$.
  - $(\{q_2\}, \{q_1, q_2\})$ forces 1 to occur infinitely many times.
  - $(\varnothing, \{q_1\})$ forbids 0 to occur infinitely many times.

Expressive Power

- An important question in ω-automata theory is to compare the expressive power of various acceptances.
- We have shown that non-deterministic Büchi acceptance is strictly more expressive than deterministic Büchi acceptance.
- What is the relation between non-deterministic Büchi acceptance and non-deterministic Muller acceptance
  - Similarly, what about non-deterministic Rabin acceptance and non-deterministic Streett acceptance?
- What is the relation between deterministic Büchi acceptance and deterministic Muller acceptance
  - And between deterministic Rabin acceptnace and deterministic Streett acceptance?
- We will address these questions shortly.

Expressive Power (Overview)



D: Deterministic, N: Nondeterministic
B: Büchi, M: Muller, R: Rabin, S: Streett
A: Automata
X → Y: X can be translated to Y

(The graph here only covers translations in this lecture and hence is not complete.)

Büchi to Muller Acceptance

Lemma

Let $B = (Q, \Sigma, \delta, q_0, F)$ be a finite automaton with Büchi acceptance. Define $M = (Q, \Sigma, \delta, q, \mathcal{F})$ with $\mathcal{F} = \{G \subseteq Q : G \cap F \neq \varnothing\}$. Then $L_\omega(B) = L_\omega(M)$.

Proof.

Let α be an input sequence and ρ an infinite run over α on B. $\alpha \in L_\omega(B)$ iff $\text{Inf}(\rho) \cap F \neq \varnothing$ iff $\text{Inf}(\rho) \in \mathcal{F}$ iff $\alpha \in L_\omega(M)$. □

Example

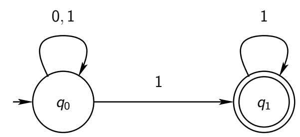


Figure: NFA M_0

- The finite automaton $M = (\{q_0, q_1\}, \{0, 1\}, \delta, q_0, \mathcal{F})$ with Muller acceptance where
  - $\delta = \{(q_0, 0, q_0), (q_0, 1, q_0), (q_0, 1, q_1), (q_1, 1, q_1)\}$;
  - $\mathcal{F} = \{\{q_1\}, \{q_0, q_1\}\}$

  accepts the same ω-language.

Muller to Büchi Acceptance

Lemma

Let $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ be a finite automaton with Muller acceptance. There is a finite automaton $B = (Q', \Sigma, \delta', q_0, F)$ with Büchi acceptance such that $L_\omega(B) = L_\omega(M)$.

Proof.

The idea is to "guess" a set $G \in \mathcal{F}$ and check whether all states in G are visited infinitely many times.
For each $G \in \mathcal{F}$, we define $Q_G = \{q_G : q \in G\}$. Moreover, we use a set to record which states in G have been visited. Define $Q' = Q \cup \bigcup_{G \in \mathcal{F}} (Q_G \times 2^G)$.

$$\begin{aligned} \delta' = \delta \cup & \{(p, a, (q_G, \varnothing)) : (p, a, q) \in \delta\} \cup \\ & \{((p_G, R), a, (q_G, R \cup \{p\})) : (p, a, q) \in \delta, R \neq G\} \cup \\ & \{((p_G, G), a, (q_G, \varnothing)) : (p, a, q) \in \delta\}. \\ F = & \{(q_G, \varnothing) : q_G \in Q_G, G \in \mathcal{F}\}. \end{aligned}$$

□

Example

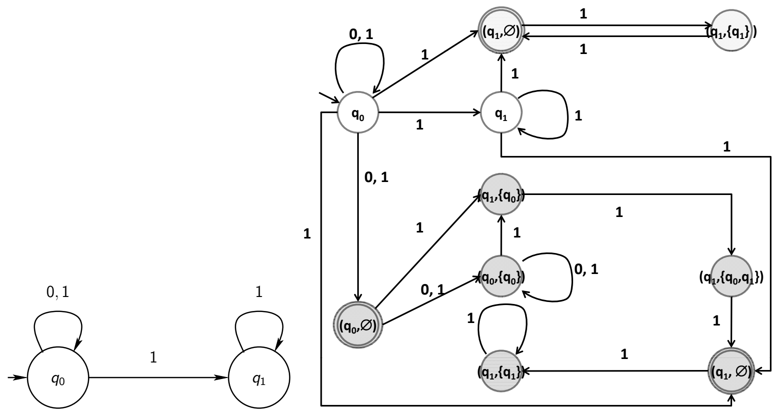


Figure: NFA M_7

- Consider $M = (\{q_0, q_1\}, \{0, 1\}, \delta, q_0, \mathcal{F})$ where $\delta = \{(q_0, 0, q_0), (q_0, 1, q_0), (q_0, 1, q_1), (q_1, 1, q_1)\}$ and $\mathcal{F} = \{\{q_0, q_1\}, \{q_1\}\}$.

Rabin and Streett to Muller Acceptance

Lemma

Let $R = (Q, \Sigma, \delta, q_0, \Omega)$ be a finite automaton with Rabin acceptance. Define $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ with Muller acceptance where

$$\mathcal{F} = \{G \subseteq Q : \exists (E, F) \in \Omega . G \cap E = \emptyset \wedge G \cap F \neq \emptyset\}.$$

Then $L_\omega(R) = L_\omega(M)$.

Lemma

Let $S = (Q, \Sigma, \delta, q_0, \Omega)$ be a finite automaton with Streett acceptance. Define $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ with Muller acceptance where

$$\mathcal{F} = \{G \subseteq Q : \forall (E, F) \in \Omega . G \cap E \neq \emptyset \vee G \cap F = \emptyset\}.$$

Then $L_\omega(S) = L_\omega(M)$.

- These two follow from the definition immediately.

Example

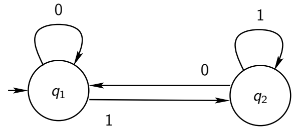


Figure: DFA M_5

- Consider the finite automaton $R = (\{q_1, q_2\}, \{0, 1\}, \delta, q_1, \Omega)$ with Rabin acceptance where
  - $\delta = \{(q_1, 0, q_1), (q_1, 1, q_2), (q_2, 0, q_1), (q_2, 1, q_2)\}$
  - $\Omega = \{\{q_1\}, \{q_2\}\}$.
- The finite automaton $M = (\{q_1, q_2\}, \{0, 1\}, \delta, q_1, \{\{q_2\}\})$ with Muller acceptance accepts the same ω-language.

Büchi to Rabin and Street Acceptance

Lemma

Let $B = (Q, \Sigma, \delta, q_0, F)$ be a finite automaton with Büchi acceptance. Define $R = (Q, \Sigma, \delta, q_0, \Omega)$ with Rabin acceptance where $\Omega = \{(\emptyset, F)\}$. Then $L_\omega(B) = L_\omega(R)$.

Lemma

Let $B = (Q, \Sigma, \delta, q_0, F)$ be a finite automaton with Büchi acceptance. Define $S = (Q, \Sigma, \delta, q_0, \Omega)$ with Rabin acceptance where $\Omega = \{(F, Q)\}$. Then $L_\omega(B) = L_\omega(S)$.

- These two also follow by definition immediately.

Example

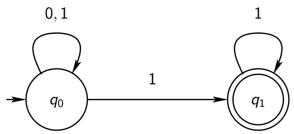


Figure: NFA M_0

- Consider the finite automaton $M_0 = (\{q_0, q_1\}, \{0,1\}, \delta, q_0, \{q_1\})$ with Büchi acceptance where
  - $\delta = \{(q_0, 0, q_0), (q_0, 1, q_0), (q_0, 1, q_1), (q_1, 1, q_1)\}$.
- The finite automaton $R = (\{q_0, q_1\}, \{0,1\}, \delta, q_0, \{(\emptyset, \{q_1\})\})$ with Rabin acceptance recognizes the same ω-language.

Expressive Power

- We have the following transformaion:
  - Büchi to Muller acceptance
  - Muller to Büchi acceptance
  - Rabin and Streett to Muller acceptance
  - Büchi to Rabin and Streett acceptance
- Therefore,

Lemma

The following classes of ω-languages are equivalent:

1. $\{L_\omega(M) : M$ *is an NFA with Büchi acceptance* $\}$;
2. $\{L_\omega(M) : M$ *is an NFA with Muller acceptance* $\}$;
3. $\{L_\omega(M) : M$ *is an NFA with Rabin acceptance* $\}$;
4. $\{L_\omega(M) : M$ *is an NFA with Streett acceptance* $\}$.

Deterministic Muller to Rabin Acceptance

Lemma

Let $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ be a DFA with Muller acceptance. Assume $Q = \{1, 2, \ldots, k\}$ and $q_0 = 1$. Consider $R = (Q', \Sigma, \delta', q_0', \Omega)$ with Rabin acceptance where

- $Q' = \{w \in (Q \cup \{\natural\})^* : \forall q \in Q \cup \{\natural\}, q$ *occurs in* w *exactly once.* $\}$.
- $q_0' = \natural\, k \cdots 1$.
- $\delta'(m_1 \cdots m_r \natural m_{r+1} \cdots m_k, a) = m_1 \cdots m_{s-1} \natural m_{s+1} \cdots m_k m_s$ *if* $\delta(m_k, a) = m_s$.
- $\Omega = \{(E_0, F_0), \ldots, (E_k, F_k)\}$ *with*
  - $E_i = \{u \natural v : |u| < i\}$
  - $F_i = \{u \natural v : |u| < i\} \cup \{u \natural v : |u| = i$ *and* $\{m \in Q : m$ *occurs in* $v\} \in \mathcal{F}\}$.

We have $L_\omega(M) = L_\omega(R)$.

Deterministic Muller to Rabin Acceptance

Proof (sketch).

Let us consider a run ρ of M with $\text{Inf}(\rho) = J = \{m_1, \ldots, m_j\}$. In the corresponding run on R, states in $Q \setminus J$ will eventually move before $\natural$. Hence, R will finally visits states of the form $u \natural v$ where u contains all states in $Q \setminus J$. Therefore, $|u| \geq |Q \setminus J|$ and $|v| \leq |J| = j$ eventually. Since J are visited infinitely often, we have $|v| = |J| = j$ infinitely often. Moreover, the states in v when $|v| = j$ are precisely the set J. □

Example

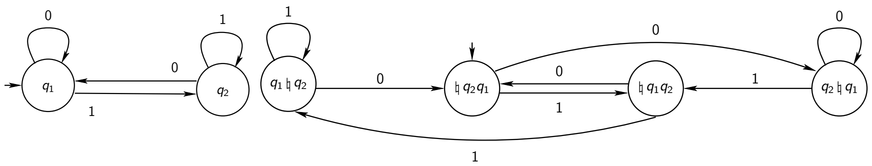


Figure: DFA M_8

- Consider $M_5 = (\{q_1, q_2\}, \{0,1\}, \delta, q_1, \{\{q_2\}\})$ with Muller acceptance where
  - $\delta = \{(q_1, 0, q_1), (q_1, 1, q_2), (q_2, 0, q_1), (q_2, 1, q_2)\}$.
- The DFA $M_8 = (Q, \{0,1\}, \delta', \{(E_0, F_0), (E_1, F_1), (E_2, F_2)\})$ with Rabin acceptance where
  - $Q = \{\natural q_1 q_2, \natural q_2 q_1, q_1 \natural q_2, q_2 \natural q_1\}$
  - $(E_0, F_0) = (\emptyset, \emptyset)$
  - $(E_1, F_1) = (\{\natural q_1 q_2, \natural q_2 q_1\}, \{\natural q_1 q_2, \natural q_2 q_1, q_1 \natural q_2\})$
  - $(E_2, F_2) = (Q, Q)$

  recognizes the same language.

Deterministic Rabin to Muller Acceptance

Lemma

Let $R = (Q, \Sigma, \delta, q_0, \Omega)$ be a DFA with Rabin acceptance. Define $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ with Muller acceptance where

$$\mathcal{F} = \{G \subseteq Q : \exists (E, F) \in \Omega . G \cap E = \emptyset \wedge G \cap F \neq \emptyset\}.$$

Then $L_\omega(R) = L_\omega(M)$.

- This is the same construction for the non-deterministic case.

Deterministic Rabin to Streett Acceptance

Lemma

Let $D = (Q, \Sigma, \delta, q_0, \Omega)$ be a DFA with Rabin acceptance. Consider $E = (Q, \Sigma, \delta, q_0, \Omega)$ as a DFA with Streett acceptance. Then $L_\omega(D) = \Sigma^\omega \setminus L_\omega(E)$.

Proof.

Rabin acceptance and Streett acceptance are complementary. □

Lemma

Let $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ be a DFA with Muller acceptance. Define $M' = (Q, \Sigma, \delta, q_0, 2^Q \setminus \mathcal{F})$. Then $L_\omega(M) = \Sigma^\omega \setminus L_\omega(M')$.

Proof.

By definition. □

Deterministic Rabin to Streett Acceptance

Lemma

Let R be a DFA with Rabin acceptance. There is a DFA S with Streett acceptance such that $L_\omega(R) = L_\omega(S)$.

Proof.

We construct a DFA M with Muller acceptance such that $L_\omega(M) = L_\omega(R)$. Build M' with Muller acceptance such that $L_\omega(M') = \Sigma^\omega L_\omega(M)$. Then we construct a DFA R' with Rabin acceptance such that $L_\omega(R') = L_\omega(M')$. Then $S = R'$ with Street acceptance is what we want. We have the following equation:

$$\begin{aligned} & L_\omega(S) \text{ with Streett acceptance} \\ = \; & \Sigma^\omega \setminus L_\omega(R') \text{ with Rabin acceptance} \\ = \; & \Sigma^\omega \setminus L_\omega(M') \text{ with Muller acceptance} \\ = \; & \Sigma^\omega \setminus (\Sigma^\omega \setminus L_\omega(M)) \text{ with Muller acceptance} \\ = \; & L_\omega(M) \text{ with Muller acceptance} \\ = \; & L_\omega(R) \text{ with Rabin acceptance.} \end{aligned}$$

□

Example

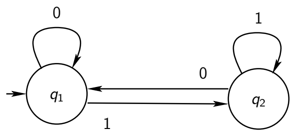


Figure: Rabin to Muller Acceptance

- Consider the DFA $R = (\{q_1, q_2\}, \{0, 1\}, \delta, q_0, \Omega)$ with Rabin acceptance where
  - $\delta = \{(q_1, 0, q_1), (q_1, 1, q_2), (q_2, 0, q_1), (q_2, 1, q_2)\}$
  - $\Omega = \{(\{q_1\}, \{q_2\})\}$
- The DFA $M = (\{q_1, q_2\}, \{0, 1\}, \delta, q_1, \{\{q_2\}\})$ with Muller acceptance recognizes the same language.

Example

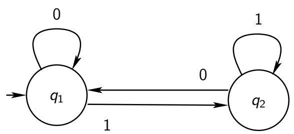


Figure: Muller Complementation

- Consider the DFA $M = (\{q_1, q_2\}, \{0, 1\}, \delta, q_1, \{\{q_2\}\})$ with Muller acceptance where
  - $\delta = \{(q_1, 0, q_1), (q_1, 1, q_2), (q_2, 0, q_1), (q_2, 1, q_2)\}$
- The DFA $M' = (\{q_1, q_2\}, \{0, 1\}, \delta, q_1, \{\varnothing, \{q_1\}, \{q_1, q_2\}\})$ with Muller acceptance recognizes $\Sigma^\omega \setminus L_\omega(M)$.

Example

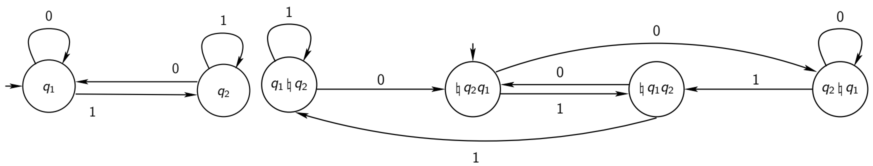


Figure: Muller to Rabin Acceptance

- Consider the DFA $M' = (\{q_1, q_2\}, \{0, 1\}, \delta, q_1, \{\varnothing, \{q_1\}, \{q_1, q_2\}\})$ with Muller acceptance where
  - $\delta = \{(q_1, 0, q_1), (q_1, 1, q_2), (q_2, 0, q_1), (q_2, 1, q_2)\}$
- The DFA $R' = (Q, \{0, 1\}, \delta', \{(E_0, F_0), (E_1, F_1, (E_2, F_2), (E_3, F_3))\})$ with Rabin acceptance where
  - $Q = \{\natural q_1 q_2, \natural q_2 q_1, q_1 \natural q_2, q_2 \natural q_1\}$
  - $(E_0, F_0) = (\varnothing, \{\natural q_1 q_2, \natural q_2 q_1\})$
  - $(E_1, F_1) = (\{\natural q_1 q_2, \natural q_2 q_1\}, \{\natural q_1 q_2, \natural q_2 q_1, q_2 \natural q_1\})$
  - $(E_2, F_2) = (E_3, F_3) = (Q, Q)$

  recognizes $L_\omega(M')$.

Example

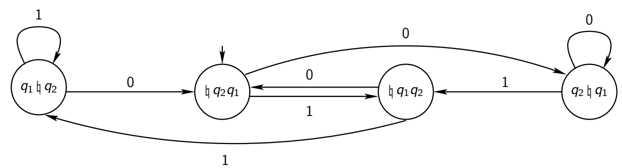


Figure: Rabin Complementation

- Consider the DFA $R' = (Q, \{0, 1\}, \delta', \{(E_0, F_0), (E_1, F_1), (E_2, F_2), (E_3, F_3)\})$ with Rabin acceptance where
  - $Q = \{\natural q_1 q_2, \natural q_2 q_1, q_1 \natural q_2, q_2 \natural q_1\}$
  - $(E_0, F_0) = (\varnothing, \{\natural q_1 q_2, \natural q_2 q_1\})$
  - $(E_1, F_1) = (\{\natural q_1 q_2, \natural q_2 q_1\}, \{\natural q_1 q_2, \natural q_2 q_1, q_2 \natural q_1\})$
  - $(E_2, F_2) = (E_3, F_3) = (Q, Q)$
- The DFA $S = (Q, \{0, 1\}, \delta', \{(E_0, F_0), (E_1, F_1), (E_2, F_2), (E_3, F_3)\})$ with Streett acceptance recognizes $\Sigma^\omega \setminus L_\omega(R')$.

Expressive Power

- In summary, we have shown Muller, Rabin, and Streett acceptaces are equivalent for deterministic finite automata.

Theorem

The following classes of ω-languages are equivalent:

1. $\{L_\omega(D) : D$ *is a DFA with Muller acceptance* $\}$;
2. $\{L_\omega(D) : D$ *is a DFA with Rabin acceptance* $\}$;
3. $\{L_\omega(D) : D$ *is a DFA with Streett acceptance* $\}$.

Corollary

The following classes are closed under union, intersection, and complementation:

1. $\{L_\omega(D) : D$ *is a DFA with Muller acceptance* $\}$;
2. $\{L_\omega(D) : D$ *is a DFA with Rabin acceptance* $\}$;
3. $\{L_\omega(D) : D$ *is a DFA with Streett acceptance* $\}$.

Relating Nondeterministic and Deterministic Classes

- We have shown that Büchi, Muller, Rabin, Streett acceptances are equivalent for nondeterministic finite automata
- We also know that Muller, Rabin, Streett acceptances are equivalent for deterministic finite automata
- Are these two classes of ω-languages equivalent?
  - YES!
- We can in fact compute the complement of NFA with Büchi acceptance
  - Transform NFA with Büchi acceptance to DFA with, say, Muller acceptance
  - Find the complement of the DFA with Muller acceptance
  - Transform DFA with Muller acceptance to NFA with Büchi acceptance
- In Prof. Tsay's lecture, a construction for complementation will be given. (Have fun!)

Second-Order Logic

- Second-order logic (SO) is an extension of first-order logic.
- It allows relational variables $X, Y, Z, \ldots$.
- Terms in second-order logic includes
  - All terms in first-order logic; and
  - $Xt_1 \cdots t_n$ where X is an n-ary relational variable and $t_1, \ldots, t_n$ are terms.
- Well-formed formulae in second-order logic includes
  - All well-formed formulae in first-order logic; and
  - $\exists X \phi$ where X is a relational variable and ϕ a formula.

Monadic Second-Order Logic: Syntax

- A 1-ary relational symbol is called monadic.
- Monadic second-order logic (MSO) is a subclass of second-order logic where all relational variables are monadic.
- The syntax of monadic second-order logic over vocabulary σ (MSO$[\sigma]$) is as follows.
  - If $X, Y \in \sigma$ are monadic, $X \subseteq Y$ is in MSO$[\sigma]$;
  - If $R, Y_1, Y_2, \ldots, Y_k$ are in MSO$[\sigma]$ and R has arity k, then $RY_1 Y_2 \cdots Y_k$ is in MSO$[\sigma]$;
  - If ϕ and ψ are in MSO$[\sigma]$, so are $\neg\phi$ and $\phi \vee \psi$;
  - If ϕ is in MSO$[\sigma \cup \{X\}]$ and X is monadic, then $\exists X \phi$ is in MSO$[\sigma]$.

Monadic Second-Order Logic: Semantics

- The satisfication relation $\models$ is defined as follows. Let $\mathfrak{U}$ be a model over the vocabulary σ.
  - $\mathfrak{U} \models X \subseteq Y$ if $X^{\mathfrak{U}} \subseteq Y^{\mathfrak{U}}$;
  - $\mathfrak{U} \models RY_1 \cdots Y_k$ if $R^{\mathfrak{U}} \cap (Y_1^{\mathfrak{U}} \times \cdots \times Y_k^{\mathfrak{U}}) \neq \emptyset$;
  - $\mathfrak{U} \models \neg\phi$ is not $\mathfrak{U} \models \phi$;
  - $\mathfrak{U} \models \phi \vee \psi$ if $\mathfrak{U} \models \phi$ or $\mathfrak{U} \models \psi$;
  - $\mathfrak{U} \models \exists X \phi$ if there is an extension model $\mathfrak{B}$ of $\mathfrak{U}$ over $\sigma \cup \{X\}$ such that $\mathfrak{B} \models \phi$.
- Semantically, a monadic symbol represents a set of objects
- Where is the first-order quantification?
  - $\exists x \phi$ is not in MSO$[\sigma]$!

Abbreviations

- We use the following abbreviations:

$$
\begin{array}{lll}
\phi \wedge \psi & \text{for} & \neg(\neg\phi \vee \neg\psi) \\
\phi \rightarrow \psi & \text{for} & \neg\phi \vee \psi \\
\forall X \phi & \text{for} & \neg\exists X \neg\phi \\
X = \emptyset & \text{for} & \forall Y X \subseteq Y \\
\text{sing}(x) & \text{for} & \neg x = \emptyset \wedge \forall X(X \subseteq x \rightarrow (x \subseteq X \vee X = \emptyset)) \\
x \in P & \text{for} & \text{sing}(x) \wedge x \subseteq P \\
P = Q & \text{for} & P \subseteq Q \wedge Q \subseteq P \\
\exists x \in P \phi & \text{for} & \exists x(x \in P \wedge \phi) \\
\forall x \in P \phi & \text{for} & \forall x(x \in P \rightarrow \phi).
\end{array}
$$

- Note that sing(x) means that x is a singleton set
  - x is a 1-ary relation and $o \in x$ for exactly one object o

Weak Monadic Second-Order Logic

- Weak Monadic Second-Order Logic (WMSO) has the same syntax as MSO. Its semantics however is slightly different:
  - $\mathfrak{U} \models_W \exists X \phi$ if there is an extension model $\mathfrak{B}$ over $\sigma \cup \{X\}$ such that $\mathfrak{B} \models_w \phi$ and $X^{\mathfrak{B}}$ is finite.
- In other words, the second-order quantification in WMSO is over finite sets.
  - On the other hand, we can quantify arbitrary sets in MSO.

Infinite Inputs as Structures

- Let Σ be a finite alphabet.
- Consider the structure $\mathfrak{I} = (\mathbb{Z}^+, S^{\mathfrak{I}}, (P_a^{\mathfrak{I}})_{a \in \Sigma})$ where
  - $S^{\mathfrak{I}} = \{(n, n+1) : n \in \mathbb{Z}^+\}$;
  - $P_a^{\mathfrak{I}} \subseteq \mathbb{Z}^+$ for all $a \in \Sigma$.
- Intuitively, each positive integer represents a position in an input sequence.
- A position in the set $P_a^{\mathfrak{I}}$ means that the symbol a appears in the position
- We can represent an infinite input with such a structure.

Example

- Let $\Sigma = \{0, 1\}$.
- The input sequence 0^ω corresponds to $\mathfrak{I}_0 = (\mathbb{Z}^+, S^{\mathfrak{I}_0}, P_0^{\mathfrak{I}_0} = \mathbb{Z}^+, P_1^{\mathfrak{I}_0} = \varnothing)$.
- The input sequence $(01)^\omega$ corresponds to $\mathfrak{I}_1 = (\mathbb{Z}^+, S^{\mathfrak{I}_1}, P_0^{\mathfrak{I}_1} = \{2k+1 : k \in \mathbb{N}\}, P_1^{\mathfrak{I}_1} = \{2k : k \in \mathbb{Z}^+\})$.

S1S and WS1S

- Monadic Second-Order Logic with One Successor (S1S) is the logic MSO over infinite inputs.
  - That is, the satisfication relation $\models$ is restricted to infinite inputs on the left
- Weak Monadic Second-Order Logic with One Successor (WS1S) is the logic WMSO over infinite inputs.

Initially Closed Sets

- A set P of $\mathbb{Z}^+$ is initially closed if

$$\text{for all } x, y \in \mathbb{Z}^+ (y \in P \wedge x \leq y \rightarrow x \in P).$$

- Consider the following formula:

$$\text{InCl}(P) = \forall x \forall y ((\text{sing}(x) \wedge Sxy \wedge y \in P) \rightarrow x \in P).$$

- Then

Lemma

For any infinite input structure $\mathfrak{I}$, the following are equivalent:

- $\mathfrak{I} \models InCl(P)$;
- $\mathfrak{I} \models_W InCl(P)$;
- *P is initially closed.*

Transitive Closure of Successor

- Consider the following binary relations:

$$\begin{array}{lcl} < & = & \{(n, n+m) : n, m \in \mathbb{Z}^+\} \\ \leq & = & < \cup \{(n, n) : n \in \mathbb{Z}^+\}. \end{array}$$

- We can represent these relations in (W)S1S:

$$\begin{array}{lcl} x \leq y & = & \text{sing}(y) \wedge \forall P ((\text{InCl}(P) \wedge y \in P) \rightarrow x \in P) \\ x < y & = & x \leq y \wedge \neg(x = y). \end{array}$$

- Thus, we are free to use $x < y$ and $x \leq y$ in (W)S1S.

Infiniteness

- Let $\mathfrak{I}$ be an infinite input structure.
- Consider the following S1S formula:

$$\mathsf{Inf}(P) = \exists P'(P' \neq \emptyset \wedge \forall x' \in P' \exists y \in P \exists y' \in P'(x' < y \wedge x' < y')).$$

- We have $\mathfrak{I} \models \mathsf{Inf}(P)$ if P is an infinite subset of $\mathbb{Z}^+$.
  - Informally, P is an infinite subset of $\mathbb{Z}^+$ if there are infinite $x'_0 < x'_1 < x'_2 < \cdots$ such that for each i, there is a y_i such that $x'_i < y_i$.

Logic and Finite Automata

- Let α be an infinite input over Σ and $\mathfrak{I}_\alpha$ its infinite input structure.
- We have two formalisms to define ω-languages over Σ:
  - $L_\omega(M) = \{\alpha : \alpha \text{ is accepted by the DFA } M\}$;
  - $L_\omega(\phi) = \{\alpha : \mathfrak{I}_\alpha \models \phi, \phi \text{ is an S1S formula}\}$.
- An important question (as in DFA's and NFA's) is to determine the expressive power of finite automata over infinite inputs and S1S over infinite input structures. More precisely,
  - Given a DFA M with Muller acceptance, is there an S1S formula ϕ such that $L_\omega(M) = L_\omega(\phi)$?
  - Given an S1S formula ϕ, is there a DFA M with Muller acceptance such that $L_\omega(\phi) = L_\omega(M)$?
- We will show that finite automata and S1S formulae are equally expressive.

Finite Automata to S1S

Lemma

For each NFA M with Muller acceptance, there is a formula $\phi_M \in S1S$ such that $\forall \alpha \in \Sigma^\omega$, M accepts α iff $\mathfrak{I}_\alpha \models \phi_M$.

Proof.

Let $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$. Define $\overline{R} = (R_q)_{q \in Q}$. Consider

$$\phi_M = \exists \overline{R}(\mathsf{Part} \wedge \mathsf{Init} \wedge \mathsf{Trans} \wedge \mathsf{Accept}).$$

Part formalizes that the states on the run form a partition. Let

$$\begin{aligned}\mathsf{State}_q(x) &= x \in R_q \wedge \textstyle\bigwedge_{q' \in Q \setminus \{q\}} \neg(x \in R_{q'})\\ \mathsf{Part} &= \forall x(\mathsf{sing}(x) \rightarrow \textstyle\bigvee_{q \in Q} \mathsf{State}_q(x)).\end{aligned}$$

Finite Automata to S1S

Proof.

Init formalizes the initial condition.

$$\mathsf{Init} = \exists x(\mathsf{State}_{q_0}(x) \wedge \forall y(\mathsf{sing}(y) \rightarrow x \leq y).$$

Trans expresses the transition relation.

$$\begin{aligned}\mathsf{Trans} = {} & \forall x \forall x'((\mathsf{sing}(x) \wedge \mathsf{sing}(x') \wedge Sxx') \rightarrow\\ & \textstyle\bigvee_{(q,a,q') \in \delta}(\mathsf{State}_q(x) \wedge x \in P_a \wedge \mathsf{State}_{q'}(x'))).\end{aligned}$$

Accept represents the Muller acceptance. Consider

$$\begin{aligned}\mathsf{InfOcc}_q(P) &= \exists Q(Q \subset P \wedge Q \subseteq R_q \wedge \mathsf{Inf}(Q))\\ \mathsf{Muller}(P) &= \textstyle\bigvee_{F \in \mathcal{F}}(\bigwedge_{q \in F} \mathsf{InfOcc}_q(P) \wedge \bigwedge_{q \notin F} \neg\mathsf{InfOcc}_q(P))\\ \mathsf{Path}(P) &= \mathsf{Inf}(P) \wedge \mathsf{InCl}(P) \wedge\\ & \quad \forall Q((\mathsf{Inf}(Q) \wedge \mathsf{InCl}(Q) \wedge Q \subseteq P) \rightarrow Q = P)\\ \mathsf{Accept} &= \forall P(\mathsf{Path}(P) \rightarrow \mathsf{Muller}(P))\end{aligned}$$

S1S to Finite Automata

Lemma

For each S1S formula ϕ, there is a DFA M_ϕ with Muller acceptance such that $\Im_\alpha \models \phi$ iff $\forall \alpha \in \Sigma^\omega, M_\phi$ accepts α.

Proof.

By induction on ϕ, we construct a DFA M over 2^Σ.
For $\phi = P_a \subseteq P_b$, define $M_\phi = (\{q\}, 2^\Sigma, \delta, q, \{q\})$ where

$$\delta = \{(q, A, q) : A \subseteq \Sigma, \text{ and } a \in A \text{ implies } b \in A\}.$$

For $\phi = SP_aP_b$, define $M_\phi = (\{q_0, q_1, q_2\}, 2^\Sigma, \delta, q_0, \{q_2\})$ where

$$\begin{array}{rcl} \delta & = & \{(q_0, A', q_0) : a \notin A', A' \subseteq \Sigma\} \cup \{(q_0, A, q_1) : a \in A, A \subseteq \Sigma\} \cup \\ & & \{(q_1, B', q_0) : b \notin B', B' \subseteq \Sigma\} \cup \{(q_1, B, q_2) : b \in B, B \subseteq \Sigma\} \cup \\ & & \{(q_2, C, q_2) : C \subseteq \Sigma\}. \end{array}$$

S1S to Finite Automata

Proof.

For disjunction and negation, recall that DFA's with Muller acceptance are closed under union and complementation. We apply these constructions in inductive step.
For $\phi = \exists P_a \psi$, assume $M_\psi = (Q, 2^\Sigma, \delta, q_0, \mathcal{F})$. Define $M_\phi = (Q, 2^\Sigma, \delta', q_0, \mathcal{F})$ where

$$\delta' = \{(q, A \setminus \{a\}, q') : (q, A, q') \in \delta\}.$$

□

- Technically, we construct a DFA over 2^Σ not Σ. This is necessary when, for instance, $\phi = P_a \subseteq P_b$.
- Our presentation is overly simplified. We do not consider monadic relational variables (as in $X \subseteq P_a$).
  - We can extend the alphabet to have a fresh symbol for each relational variable.

Muller Acceptance and S1S

- Thus, we have shown that nondeterministic finite automata with Muller acceptance have the same expressive power as S1S.
- Observe that the quantification over infinite subsets is needed in Muller acceptance.
  - Precisely, $\text{InfOcc}_q(P)$ in Accept.
- The proof would not go through for WS1S where only finite subsets can be quantified.
- Is WS1S strictly less expressive than S1S?

Deterministic Muller Acceptance and WS1S

- Interestingly, the answer is negative.
- For deterministic finite automata with Muller acceptance, there is a WS1S formula which recognizes the same ω-language.
- Since deterministic finite automata with Muller acceptance is as expressive as nondeterministic ones, WS1S is as expressive as S1S.
- We will give a WS1S formula ϕ_M for each deterministic finite automata M with Muller acceptance.
  - The idea is to consider all finite prefixes of the accepting run in M.

Deterministic Muller Acceptance to WS1S

Lemma

For each DFA M with Muller acceptance, there is a formula $\phi_M \in S1S$ such that $\forall \alpha \in \Sigma^\omega, M$ accepts α iff $\mathfrak{I}_\alpha \vDash \phi_M$.

Proof.

Let $M = (Q, \Sigma, \delta, q_0, \mathcal{F})$ be a DFA with Muller acceptance. Define

$$\begin{aligned}
\mathsf{State}_q(x) &= x \in R_q \land \textstyle\bigwedge_{q' \in Q \setminus \{q\}} \neg(x \in R_{q'}) \\
\mathsf{Part}(I) &= \forall x \in I(\mathsf{sing}(x) \to \textstyle\bigvee_{q \in Q} \mathsf{State}_q(x)) \\
\mathsf{Init} &= \exists x(\mathsf{State}_{q_0}(x) \land \forall y(\mathsf{sing}(y) \to x \leq y) \\
\mathsf{Trans}(I) &= \forall x \in I \forall x' \in I((\mathsf{sing}(x) \land \mathsf{sing}(x') \land Sxx') \to \\
&\qquad \textstyle\bigvee_{(q,a,q') \in \delta}(\mathsf{State}_q(x) \land x \in P_a \land \mathsf{State}_{q'}(x'))) \\
\mathsf{Occ}_q(x) &= \exists I(\mathsf{InCl}(I) \land x \in I \land \\
&\qquad \exists \overline{R}(\mathsf{Part}(I) \land \mathsf{Init} \land \mathsf{Trans}(I) \land \mathsf{State}_q(x))) \\
\mathsf{InfOcc}_q &= \forall x(\mathsf{sing}(x) \to \exists y(x < y \land \mathsf{Occ}_q(y))) \\
\mathsf{Accept} &= \textstyle\bigvee_{F \in \mathcal{F}}(\bigwedge_{q \in F} \mathsf{InfOcc}_q \land \bigwedge_{q \notin F} \neg \mathsf{InfOcc}_q).
\end{aligned}$$

Deterministic Muller Acceptance to WS1S

Proof.

Let $\phi_M = \mathsf{Accept}$. Then $\mathfrak{I}_\alpha \vDash \phi_M$ iff M accepts α. □

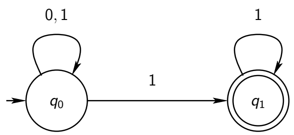


Figure: NFA M_0

- For DFA's, an infinite run is the "limit" of its finite prefixes.
- The formula InfOcc_q correctly expresses that q occurs infinite many times in the run on DFA's.
- On the other hand, InfOcc_q is not correct for NFA's.
  - Consider M_0 as a counterexample.

References

1. Hopcroft and Ullman. Introduction to Automata Theory, Languages, and Computation. Addison-Wesley.
2. Grädel, Thomas, Wilke. Automata, Logics, and Infinite Games - A Guide to Current Research. Springer.