Theoretical Computer Science

Lab Session 4

February 18, 2021

Agenda

- Exercises on FST
- Operations on FSA (Exercises)

Exercises on FST

Finite State Transducer

Finite State Transducer

A Finite State Transducer (FST) is a tuple $\langle Q, I, \delta, q_0, F, O, \eta \rangle$ where

- Q, I, δ, q_0, F: just like acceptors;
- O is the output alphabet;
- $\eta : Q \times I \rightarrow O^*$.

Remark:

- the condition for acceptance remains the same as in acceptors;
- the translation is performed only on accepted strings.

FST: an example

Build a complete FST accepting the following language over the alphabet $A = \{0, 1\}$

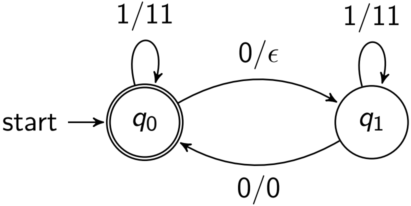
$$L = \{x \in A^* \mid \text{ the number of 0's is even}\}$$

The FST outputs the string obtained by removing every odd occurrence of 0 and doubling every occurrence of 1. Examples of inputs recognised by L and their respective outputs:

- input: 010010, output: 110110
- input: 00, output: 0
- input: 000100011, output: 011001111

FST: an example

$L = \{x \in A^* \mid \text{ the number of 0's is even}\}$



Exercises

Build complete FSAs over the languages given below:

1. $A = \{w, t, a, l, k, e, d\}$ that accepts only the verb "walked" or "talked". The FST will translate the input verb to present form ex: walked to walk.
2. $A = \{a, b\}$ that accepts only strings ending with the letter b. The FST will translate the input string where every second symbol a in the input is erased.

Your Practice Work

Build complete FSAs over the languages given below:

3. $A = \{0, 1\}$ that accepts strings that are binary representation of integers divisible by 2. The FST will translate the input string into result of division by 2.
4. $A = \{0, 1\}$ that accepts strings that are binary representation of integers divisible by 3. The FST will translate the input string into result of division by 3.

Operations on FSA

Complement (Formally)

Suppose $M = (Q, A, \delta, q_0, F)$ is a complete finite state automaton accepting L. A complement M^c is a complete FSA $M^c = (Q, A, \delta, q_0, F^c)$, where

$$F^c = Q \backslash F$$

M^c accepts the language L.

Operations on FSAs (Summary)

Suppose $M_1 = (Q_1, A, \delta_1, q_0^1, F_1)$ and $M_2 = (Q_2, A, \delta_2, q_0^2, F_2)$ are finite automata accepting L_1 and L_2, respectively.
Let M be the FSA $M = (Q, A, \delta, q_0, F)$, where

$$
\begin{aligned}
&Q = Q_1 \times Q_2 \\
&q_0 = (q_0^1, q_0^2) \\
&\delta((q,p),a) = (\delta_1(q,a), \delta_2(p,a))
\end{aligned}
$$

The set of final states will be defined as

$$
\begin{aligned}
\text{Difference } L_1 \setminus L_2 : F &= \{(q,p) \mid q \in F_1 \wedge p \notin F_2\} \\
\text{Union } L_1 \cup L_2\text{: } F &= \{(q,p) \mid q \in F_1 \vee p \in F_2\} \\
\text{Intersection } L_1 \cap L_2\text{: } F &= \{(q,p) \mid q \in F_1 \wedge p \in F_2\}
\end{aligned}
$$

Exercises (Part 1)

Let $A = \{0, 1\}$ be the alphabet.

1. Build a complete FSA M_1 that recognises the language:
   $L_1 = \{x \in A^* \mid \ x \text{ has an even number of 1's}\}$;
2. Build a complete FSA M_2 that recognises the language:
   $L_2 = \{x \in A^* \mid \ x \text{ has an odd number of 0's}\}$;

Exercises (Part 1)

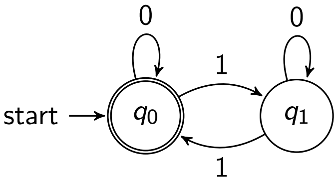
Let $A = \{0, 1\}$ be the alphabet.

1. Build a complete FSA M_1 that recognises the language:
   $L_1 = \{x \in A^* \mid \ x \text{ has an even number of 1's}\}$;
2. Build a complete FSA M_2 that recognises the language:
   $L_2 = \{x \in A^* \mid \ x \text{ has an odd number of 0's}\}$;
3. Build a complete FSA that accepts when either M_1 or M_2 accepts.
4. Build a complete FSA that accepts when both M_1 and M_2 accept.
5. Build a complete FSA that accepts when M_1 accepts and M_2 rejects.
6. Build a complement for M_1.

Part 1 Solution (1)

Let $A = \{0, 1\}$ be the alphabet.

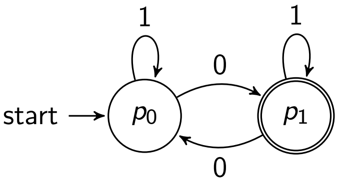
1. Build a complete FSA M_1 that recognises the language:
   $L_1 = \{x \in A^* \mid\ x \text{ has an even number of 1's}\}$;



Part 1 Solution (2)

Let $A = \{0, 1\}$ be the alphabet.

2. Build a complete FSA M_2 that recognises the language:
   $L_2 = \{x \in A^* \mid \ x \text{ has an odd number of 0's}\}$;

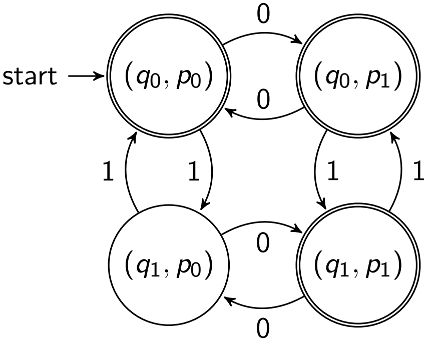


Part 1 Solution (3)

Let $A = \{0, 1\}$ be the alphabet.

3. Build a complete FSA that accepts when either M_1 or M_2 accepts.

Build a complete FSA for $L_1 \cup L_2$:

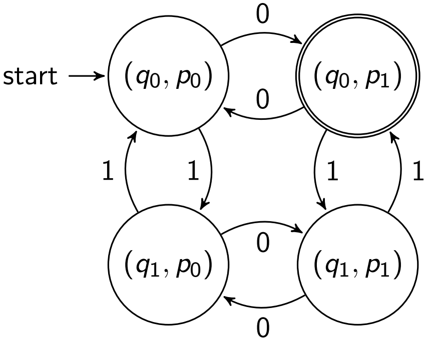


Part 1 Solution (4)

Let $A = \{0, 1\}$ be the alphabet.

4. Build a complete FSA that accepts when both M_1 and M_2 accepts.

Build a complete FSA for $L_1 \cap L_2$:

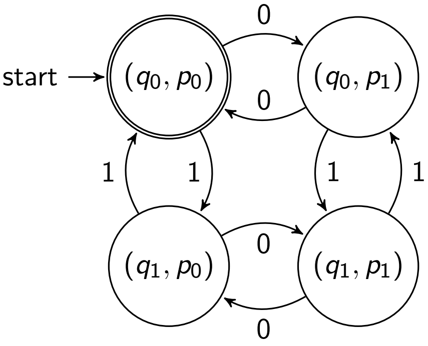


Part 1 Solution (5)

Let $A = \{0, 1\}$ be the alphabet.

5. Build a complete FSA that accepts when M_1 accepts and M_2 rejects

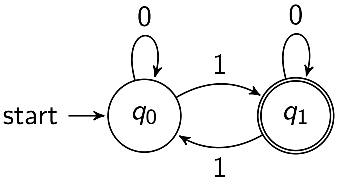
Build a complete FSA for $L_1 \backslash L_2$:



Part 1 Solution (6)

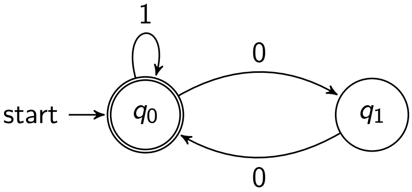
Let $A = \{0, 1\}$ be the alphabet.

6. Build a complement of M_1



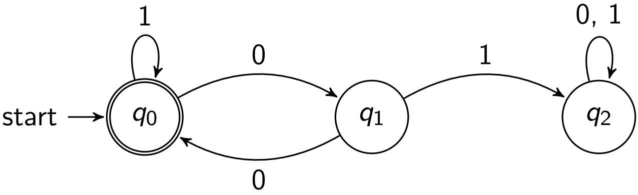
Exercises (Part 2)

Construct a complement for the following FSA

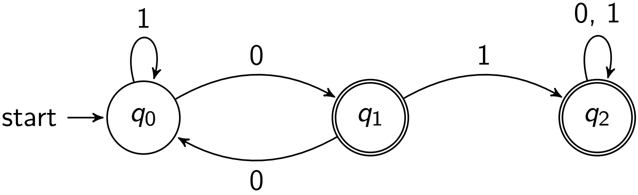


Solution (Part 2)

First, we have to complete the FSA



The complement:



Exercises (Part 3)

Let $A = \{0, 1\}$ be the alphabet.

1. Build a complete FSA M_a that recognises the language:
   $L_a = \{x \in A^* \mid x$ is the binary representation of an integer, and it is divisible by $2\}$[1];
2. Build a complete FSA M_b that recognises the language:
   $L_b = \{x \in A^* \mid x$ is the binary representation of an integer, and it is divisible by $3\}$;

[1] For simplicity assume that ϵ is a part of L_a and L_b

Exercises (Part 3)

Let $A = \{0, 1\}$ be the alphabet.

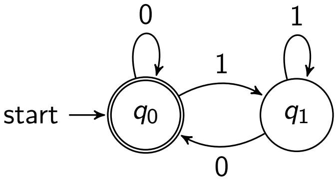
1. Build a complete FSA M_a that recognises the language: $L_a = \{x \in A^* \mid x$ is the binary representation of an integer, and it is divisible by $2\}$[1];
2. Build a complete FSA M_b that recognises the language: $L_b = \{x \in A^* \mid x$ is the binary representation of an integer, and it is divisible by $3\}$;
3. Build a complete FSA that accepts when both M_a and M_b accept.
4. Build a complete FSA that accepts when either M_a or M_b accepts.
5. Build a complete FSA that accepts when M_a accepts and M_b rejects.

[1] For simplicity assume that ϵ is a part of L_a and L_b

Part 3 Solution (1)

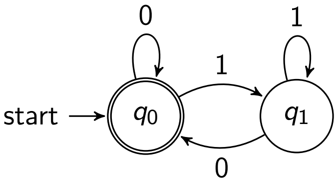
Let $A = \{0, 1\}$ be the alphabet.

1. Build a complete FSA M_a that recognises the language:
   $L_a = \{x \in A^* \mid x$ is the binary representation of an integer, and it is divisible by 2$\}$;



Solution (1) — Another representation of a complete FSA

Graphical Representation - State Transition Diagram



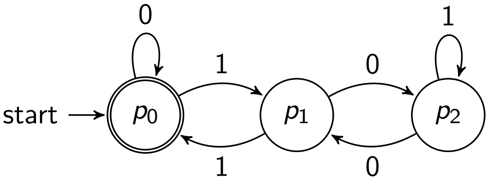
Graphical Representation — State Transition Table

	0	1
$\rightarrow^* q_0$	q_0	q_1
q_1	q_0	q_1

Part 3 Solution (2)

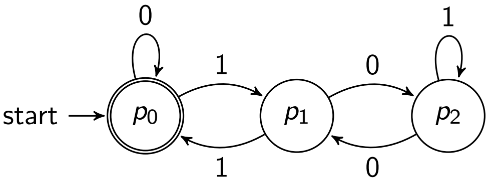
Let $A = \{0, 1\}$ be the alphabet.

2. Build a complete FSA M_b that recognises the language:
   $L_b = \{x \in A^* \mid x$ is the binary representation of an integer, and it is divisible by 3$\}$



Solution (2) – Another representation of a complete FSA

Graphical Representation - State Transition Diagram



Graphical Representation — State Transition Table

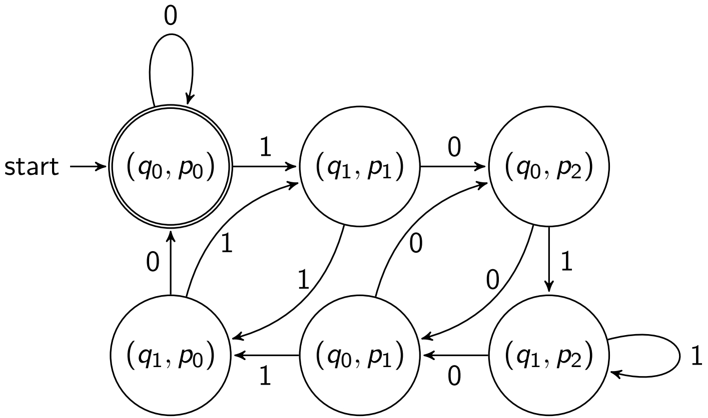
	0	1
$\rightarrow^* p_0$	p_0	p_1
p_1	p_2	p_0
p_2	p_1	p_2

Part 3 Solution (3)

Let $A = \{0, 1\}$ be the alphabet.

3. Build a complete FSA that accepts when both M_a and M_b accepts.

Build and complete FSA for $L_a \cap L_b$:

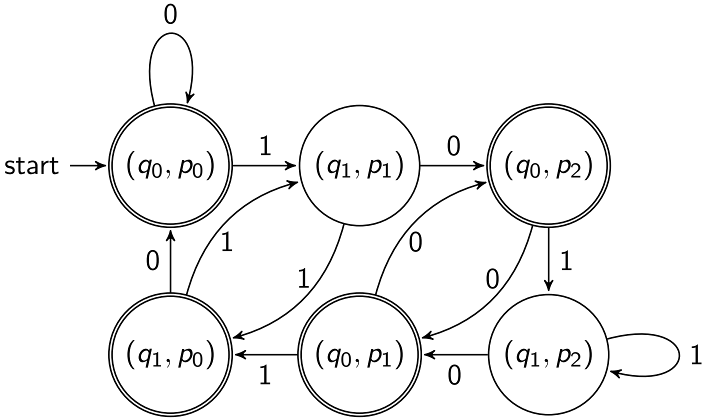


Part 3 Solution (4)

Let $A = \{0, 1\}$ be the alphabet.

4. Build a complete FSA that accepts when either M_a or M_b accept.

Build a complete FSA for $L_a \cup L_b$:

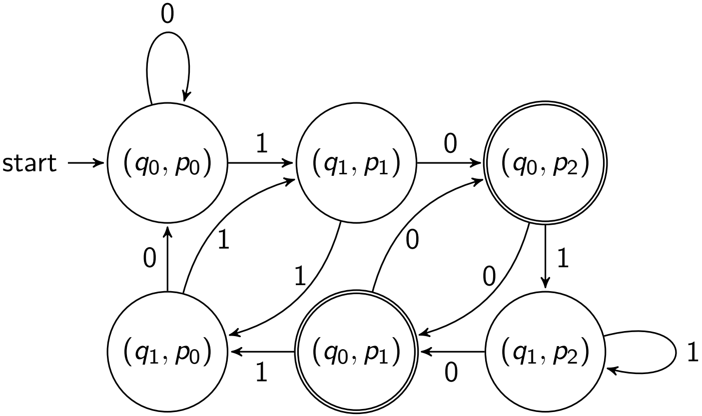


Part 3 Solution (5)

Let $A = \{0, 1\}$ be the alphabet.

5. Build a complete FSA that accepts when M_a accepts and M_b rejects.

Build and complete FSA for $L_a \backslash L_b$:



Exercises (Part 4)

Let M_1 and M_2 be the complete FSAs depicted below, accepting languages L_1 and L_2, respectively. Draw complete FSAs accepting the following languages.

i $L_1 \cup L_2$
ii $L_1 \cap L_2$
iii $L_1 \backslash L_2$