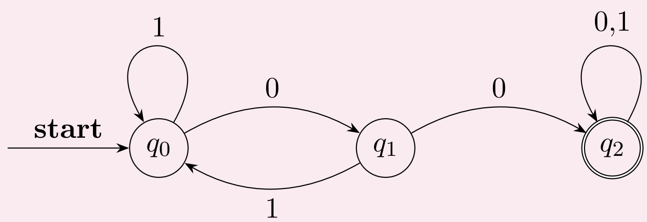
Deterministic Finite Automata

Jubair Ahammad Akter

February 23, 2026

1. Deterministic Finite Automata (DFA)

Transition Diagram



The above picture is called the **Transition Diagram**.

Example Specification

Set of States:

$$Q = \{q_0, q_1, q_2\}$$

Input Alphabet:

$$\Sigma = \{0, 1\}$$

Transition Function :

State	0	1
q_0	$\delta(q_0, 0) = q_1$	$\delta(q_0, 1) = q_0$
q_1	$\delta(q_1, 0) = q_2$	$\delta(q_1, 1) = q_0$
q_2	$\delta(q_2, 0) = q_2$	$\delta(q_2, 1) = q_2$

Starting State:

$$q_0$$

Accepting State Set:

$$F = \{q_2\}$$

Tuple Representation

A **Deterministic Finite Automaton (DFA)** is defined by a 5-tuple:

$$M = (Q, \Sigma, \delta, q_0, F)$$

Where:

- $Q = \{q_0, q_1, q_2\}$ is a finite set of states
- $\Sigma = \{0, 1\}$ is the input alphabet
- δ is the transition function
- q_0 is the start state
- $F = \{q_2\}$ is the set of accepting states

So, the **Tuple Representation** of the give transition diagram is:

$$M = (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_2\})$$

Transition Table

State	0	1
q_0	q_1	q_0
q_1	q_2	q_0
q_2	q_2	q_2

Alphabet Powers

Let $\Sigma = \{0, 1\}$

- $\Sigma^0 = \{\epsilon\} \rightarrow$ empty string only
- $\Sigma^1 = \{0, 1\} \rightarrow$ strings of length 1
- $\Sigma^2 = \{00, 01, 10, 11\} \rightarrow$ strings of length 2
- $\Sigma^* = \Sigma^0 \cup \Sigma^1 \cup \Sigma^2 \cup \ldots \rightarrow$ all possible strings (including empty string)
- $\Sigma^+ = \Sigma^1 \cup \Sigma^2 \cup \Sigma^3 \cup \ldots \rightarrow$ all non-empty strings

Extended Transition Function ($\hat{\delta}$)

The transition function δ works for one symbol only. To process a whole string, we use the **Extended Transition Function**:

$$\hat{\delta} : Q \times \Sigma^* \to Q$$

Rules:

$$\hat{\delta}(q, \epsilon) = q$$

$$\hat{\delta}(q, xa) = \delta(\hat{\delta}(q, x), a)$$

Extended Transition Function For the String "01101"

Processed Input	Formula Applied	$\hat{\delta}(q_0, \text{processed input})$
ϵ	$\hat{\delta}(q_0, \epsilon) = q_0$	q_0
0	$\hat{\delta}(q_0, 0) = \delta(\hat{\delta}(q_0, \epsilon), 0) = \delta(q_0, 0)$	q_1
01	$\hat{\delta}(q_0, 01) = \delta(\hat{\delta}(q_0, 0), 1) = \delta(q_1, 1)$	q_0
011	$\hat{\delta}(q_0, 011) = \delta(\hat{\delta}(q_0, 01), 1) = \delta(q_0, 1)$	q_0
0110	$\hat{\delta}(q_0, 0110) = \delta(\hat{\delta}(q_0, 011), 0) = \delta(q_0, 0)$	q_1
01101	$\hat{\delta}(q_0, 01101) = \delta(\hat{\delta}(q_0, 0110), 1) = \delta(q_1, 1)$	q_0

Example of Extended Transition Function

Example: 3-bit strings over $\Sigma = \{0, 1\}$:
All 3-bit strings: $\Sigma^3 = \{000, 001, 010, 011, 100, 101, 110, 111\}$
Step-by-step transitions using $\hat{\delta}$:

w	$\hat{\delta}(q_0, \epsilon)$	$\hat{\delta}(q_0, w_1)$	$\hat{\delta}(q_0, w_1 w_2)$	$\hat{\delta}(q_0, w)$
000	q_0	q_1	q_2	$\mathbf{q_2}$
001	q_0	q_1	q_2	$\mathbf{q_2}$
010	q_0	q_1	q_0	$\mathbf{q_1}$
011	q_0	q_1	q_0	$\mathbf{q_0}$
100	q_0	q_0	q_1	$\mathbf{q_2}$
101	q_0	q_0	q_1	$\mathbf{q_0}$
110	q_0	q_0	q_0	$\mathbf{q_1}$
111	q_0	q_0	q_0	$\mathbf{q_0}$

Language Accepted by DFA

$$L(M) = \{w \in \Sigma^* \mid \hat{\delta}(q_0, w) \in F\}$$

Meaning:
All strings that take the machine from start state q_0 to some accepting state in F.