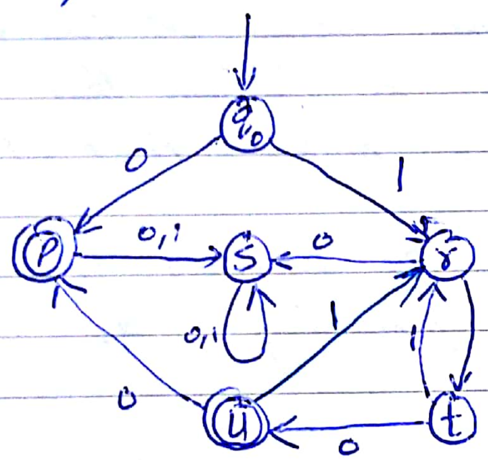
Non-Deterministic Finite Automata

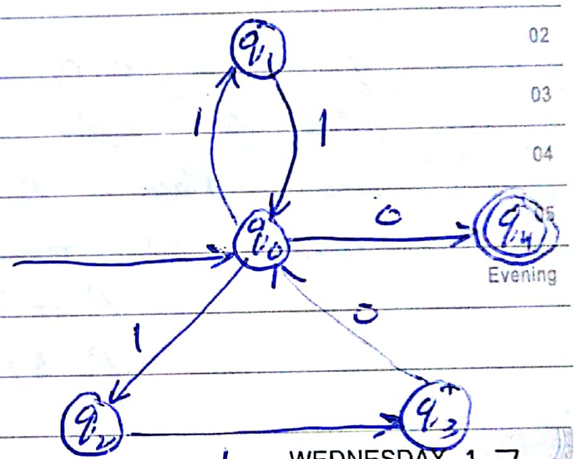
→ Similar to FA but with relaxed rules

→ NFA's accepts exactly the same language as FA's

Example



(a)



(b)

figure 4.1 (A simple approach to accepting $\{11, 110\}^*\{0\}$

Difference

→ for some states there are not transitions for both input symbols e.g q_4

→ there is more than one arrow Corresponding to same input e.g q_0 for input '1'

Definition A NFA is a 5-tuple $M = \{Q, \Sigma, q_0, A, \delta\}$ where Q and Σ are non-empty finite sets, $q_0 \in Q$, $A \subseteq Q$ and

$$\delta : Q \times \Sigma \rightarrow 2^Q$$

Q is the set of states, Σ is the alphabet, q_0 is the initial state and A is the set of accepting states.

Non_Recursive Defn of δ^* for an NFA

if $x = a_1 a_2 \text{---} a_n$, then saying M can be in state q after processing x means that there are states $P_0, P_1 \text{---} P_n$ so that $P_0 = P$, $P_n = q$ and M moves as

P_0 or (P) to P_1 by processing a_1

P_1 to P_2 by processing a_2

P_{n-1} to P_n (or q) by processing a_n

A simple way to say M can get from P_{i-1} to P_i by processing a_i is

$$\boxed{P_i \in \delta(P_{i-1}, a_i) \text{ for each } i \text{ with } 1 \le i \le n}$$

we can say that

$$P_n \in \delta^*(P_0, a_1 a_2 \text{---} a_n)$$

for $i \geq 1$

$$P_i \in \delta^*(P_0, a_1 a_2 \text{---} a_i)$$

Recursive Defn of δ^* for an NFA

~~in order~~ Let $i = n-1$ and $y = a_1 a_2 — a_{n-1}$

Then

$$p_{n-1} \in \delta^*(p, y)$$

$$\Rightarrow \quad p_n \in \delta(p_{n-1}, a_n)$$

In other words if $q \in \delta^*(p, ya_n)$, then there is a state 'r' ($r = p_{n-1}$) in the set $\delta^*(p,y)$, so then $q \in \delta(r, a_n)$

if we consider it reverse order then we can say if $q \in \delta(r, a_n)$ for some $r \in \delta^*(p,y)$, then we conclude that $q \in \delta^*(p, ya_n)$

So recursive formula becomes

$$\delta^*(p, ya_n) = \{ q \mid q \in \delta(r, a_n) \text{ for some } r \in \delta^*(p,y) \}$$

or

$$\delta^*(p, ya_n) = \bigcup_{r \in \delta^*(p,y)} \delta(r, a_n)$$

So finally we can say that

1. for any $q \in Q$, $\delta^*(q, \Lambda) = \{q\}$
2. for any $q \in Q$, $y \in \Sigma^*$, and $a \in \Sigma$

$$\delta^*(q, ya) = \bigcup_{r \in \delta^*(q,y)} \delta(r, a)$$

Definition: Acceptance by A NFA

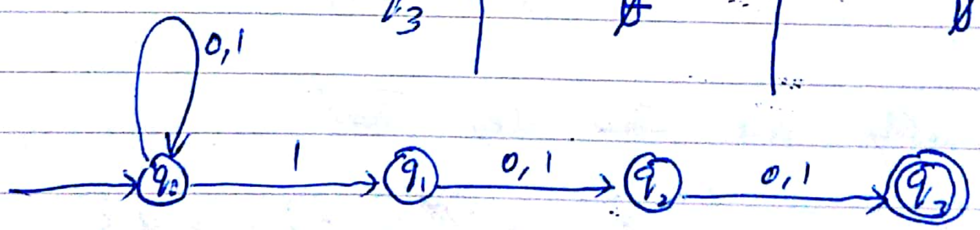
Let $M = (Q, \Sigma, q_0, A, \delta)$ be an NFA, The string $x \in \Sigma^*$ is accepted by M if $\delta^*(q_0, x) \cap A \neq \emptyset$ the language recognized or accepted by M is the set $L(M)$ of all strings accepted by M for any language $L \subseteq \Sigma^*$, L is recognized by M if $L = L(M)$.

Example

Let $M = (Q, \Sigma, q_0, A, \delta)$ where $Q = \{q_0, q_1, q_2, q_3\}$

$\Sigma = \{0, 1\}$; $A = \{q_3\}$ and δ is given by the following table

q	$\delta(q,0)$	$\delta(q,1)$
q_0	$\{q_0\}$	$\{q_0, q_1\}$
q_1	$\{q_2\}$	$\{q_2\}$
q_2	$\{q_3\}$	$\{q_3\}$
q_3	$\emptyset$	$\emptyset$



we have to determine the language recognized by the machine, we can conclude from the table that.

$\delta^*(q_0, 0) = \{q_0\}$ and $\delta^*(q_0, 1) = \{q_0, q_1\}$

Now from Recursive transition concept δ^*.

$$\delta^*(q_0, 11) = \bigcup_{r \in \delta^*(q_0, 1)} \delta(r, 1)$$

$$= \bigcup_{r \in \{q_0, q_1\}} \delta(r, 1)$$

$$= \delta(q_0, 1) \cup \delta(q_1, 1)$$

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

$$= q_0, q_1, q_2$$

Now

$$\delta^*(q_0, 01) = \bigcup_{r \in \delta^*(q_0, 0)} \delta(r, 1)$$

$$= \bigcup_{r \in \{q_0\}} \delta(r, 1)$$

$$= \delta(q_0, 1)$$

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

and

$$\delta^*(q_0, 111) = \bigcup_{r \in \delta^*(q_0 11)} \delta(r, 1)$$

$$= \delta(q_0, 1) \cup \delta(q_1, 1) \cup \delta(q_2, 1)$$

$$= \{q_0, q_1, q_2, q_3\}$$

Now

$$\delta^*(q_0, 011) = \bigcup_{r \in \delta^*(q_0, 01)} \delta(r, 1)$$

$$= \delta(q_0, 1) \cup \delta(q_1, 1)$$

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

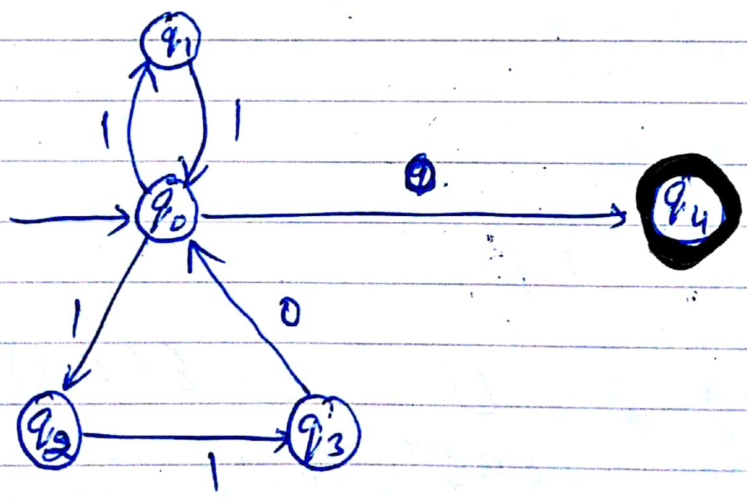
we have seen that 111 is accepted but 011 is not. So the language of M is

$$\{0,1\}^* \{1\} \{0,1\}^2$$

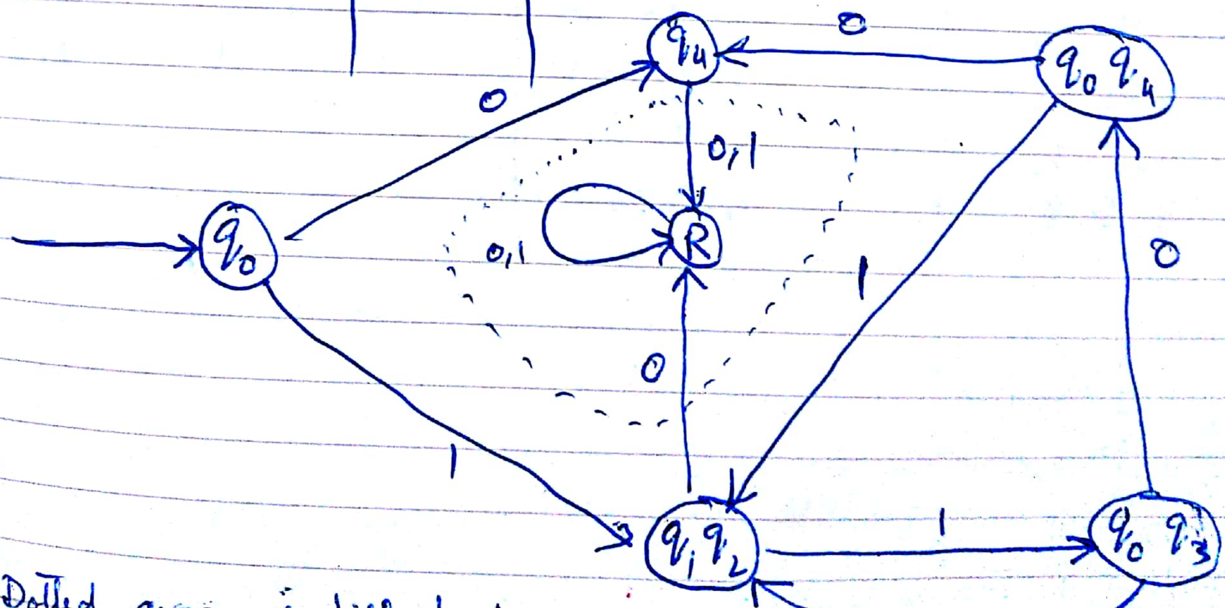
Conversion of a NFA to FA using Subset Construction Method

Example

Given a NFA as

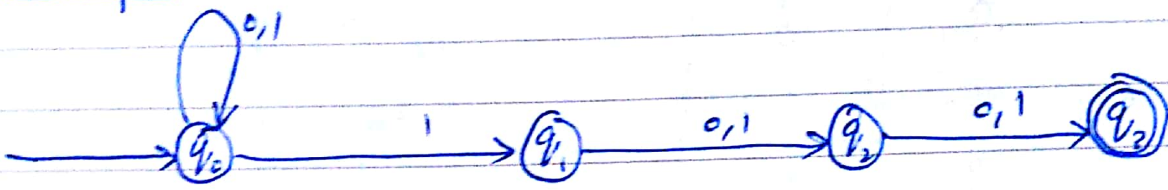


States	input	Resulting states	
q_0	0	q_4	✓
q_0	1	$q_1 q_2$	✓
q_4	0	—	
q_4	1	—	
$q_1 q_2$	0	—	
$q_1 q_2$	1	$q_0 q_3$	✓
$q_0 q_3$	0	$q_0 q_4$	✓
$q_0 q_3$	1	$q_1 q_2$	✓
$q_0 q_4$	0	q_4	✓
$q_0 q_4$	1	$q_1 q_2$	✓



Dotted area is filled to make it DFA.

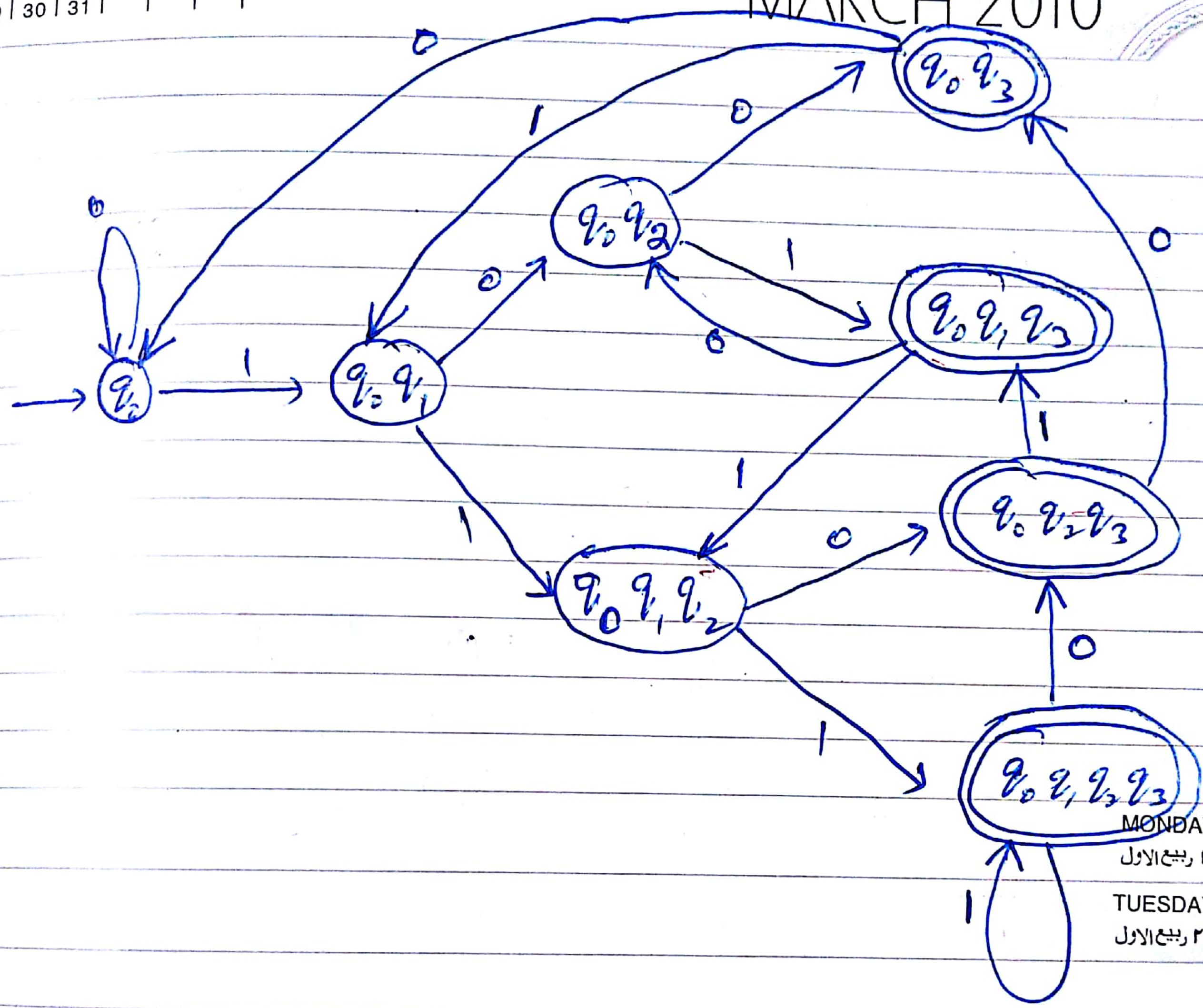
Example



it should have maximum $2^4 = 16$ states in resultant FA.

States	input	Resulting States
q_0	0	q_0 ✓
q_0	1	$q_0 q_1$ ✓
$q_0 q_1$	0	$q_0 q_2$ ✓
$q_0 q_1$	1	$q_0 q_1 q_2$ ✓
$q_0 q_2$	0	$q_0 q_3$ ✓
$q_0 q_2$	1	$q_0 q_1 q_3$ ✓
$q_0 q_1 q_2$	0	$q_0 q_2 q_3$ ✓
$q_0 q_1 q_2$	1	$q_0 q_1 q_2 q_3$ ✓

States	input	Resulting States
$q_0 q_3$	0	q_0 – ✓
$q_0 q_3$	1	$q_0 q_1$ – ✓
$q_0 q_1 q_3$	0	$q_0 q_2$ ✓
$q_0 q_1 q_3$	1	$q_0 q_1 q_2$ ✓
$q_0 q_2 q_3$	0	$q_0 q_3$ ✓
//	1	$q_0 q_1 q_3$ – ✓
$q_0 q_1 q_2 q_3$	0	$q_0 q_2 q_3$ ✓
//	1	$q_0 q_1 q_2 q_3$ ✓



Theorem 4.1

For any NFA $M = \{Q, \Sigma, q_0, A, \delta\}$ accepting a language $L \subseteq \Sigma^*$, there is an FA $M_1 = \{Q, \Sigma, q_1, A_1, \delta_1\}$ that also accepts L.

{See from book}