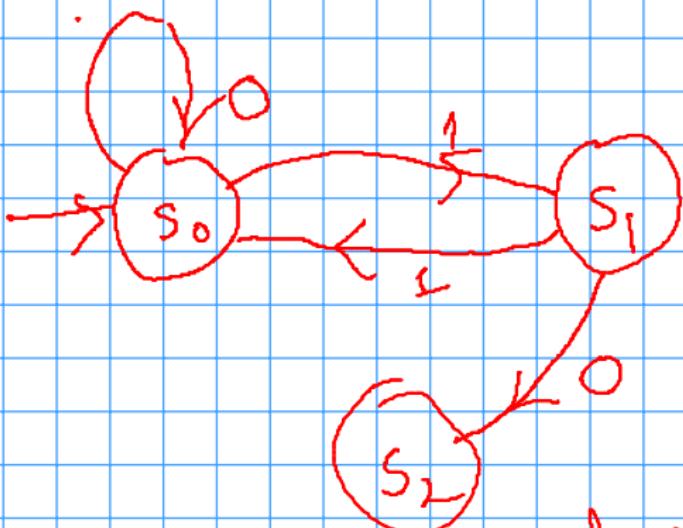


$$\{x \in \{0,1\}^* \mid d(x) \equiv 0 \pmod{3}\} \quad 00110$$

$$d(x) \% 3 = 0$$



$$n \equiv 0 \pmod{3}$$

$$\Rightarrow 2n \equiv 0 \pmod{3}$$

COMPLETE
the automaton!

$$S_2 = \{x \mid d(x) \equiv 2 \pmod{3}\} \quad 0 \leq i < 3$$

Prove that $x \in S_2 \Rightarrow d(x) \equiv 2 \pmod{3}$

CS310M: Automata Theory (Minor)

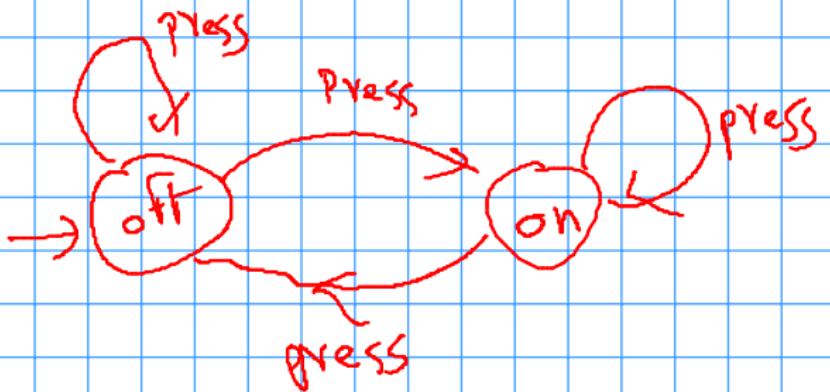
Topic 3: Nondeterministic Finite State Automata

Paritosh Pandya

Indian Institute of Technology, Bombay

Course URL: <https://cse.iitb.ac.in/~pandya58/CS310M/automata.html>

Autumn, 2021



Today's Topics:

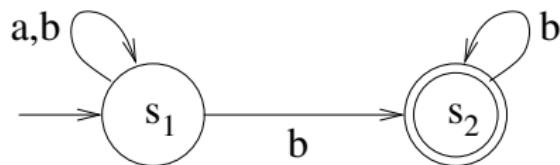
- Nondeterminism in automata
- Syntax, Semantics of NFA, Examples
- Extended Transition Function
- Determinization: Subset Construction

Source: Kozen, Lectures 5 and 6.

Nondeterminism in Automaton

Nondeterministic Finite State Automaton (NFA)

NFA A_2



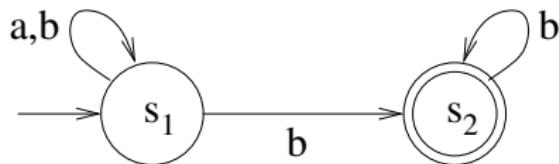
- A run of A_2 over the word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_2 \xrightarrow{b} s_2.$$

• •

Nondeterministic Finite State Automaton (NFA)

NFA A_2



- A run of A_2 over the word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_2 \xrightarrow{b} s_2.$$

✓ accepting

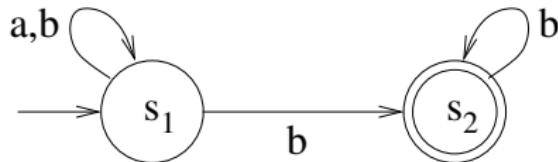
- Another run over the same word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1.$$

non-accepting

Nondeterministic Finite State Automaton (NFA)

NFA A_2



- A run of A_2 over the word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_2 \xrightarrow{b} s_2.$$

- Another run over the same word $abbb$ is:

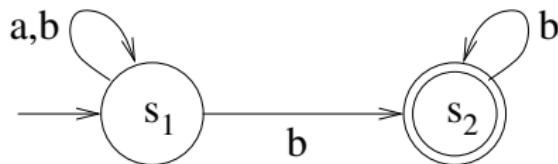
$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1.$$

- A_2 has no accepting run over the word aba as:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_2 \xrightarrow{a} \times$$

Nondeterministic Finite State Automaton (NFA)

NFA A_2



- A run of A_2 over the word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_2 \xrightarrow{b} s_2.$$

- Another run over the same word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1.$$

- A_2 has no ~~accepting~~ run over the word aba as:

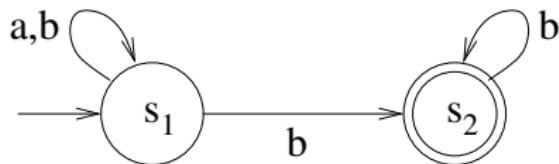
$$\cancel{s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_2 \xrightarrow{a} \times}$$

NFA can have 0, 1 or more than one runs on a given word.

Run is **accepting** if it ends in a final state.

Nondeterministic Finite State Automaton (NFA)

NFA A_2 recognises words which end in b .



- A run of A_2 over the word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_2 \xrightarrow{b} s_2.$$

- Another run over the same word $abbb$ is:

$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1 \xrightarrow{b} s_1.$$

- A_2 has no accepting run over the word aba as:

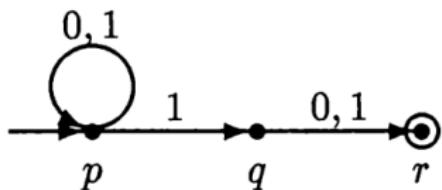
$$s_1 \xrightarrow{a} s_1 \xrightarrow{b} s_2 \xrightarrow{a} \times$$

NFA can have 0, 1 or more than one runs on a given word.

Run is **accepting** if it ends in a final state.

NFA Example

$A = \{x \in \{0, 1\}^* \mid \text{the second symbol from the right is } 1\}$.

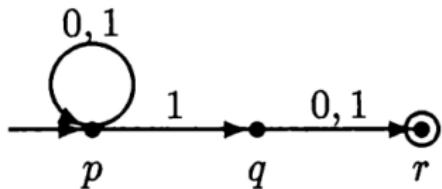


Some possible runs on word **01011**

- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{1} p$

NFA Example

$A = \{x \in \{0, 1\}^* \mid \text{the second symbol from the right is } 1\}$.

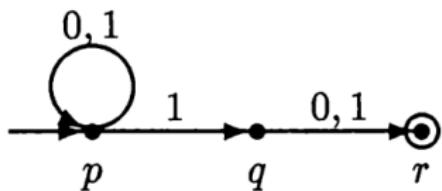


Some possible runs on word **01011**

- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{1} p$
- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} q \xrightarrow{1} r$

NFA Example

$A = \{x \in \{0, 1\}^* \mid \text{the second symbol from the right is } 1\}$.

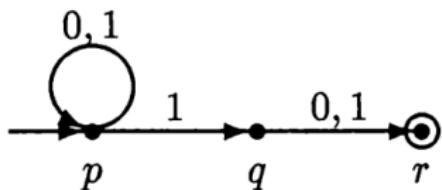


Some possible runs on word **01011**

- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{1} p$
- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} q \xrightarrow{1} r$
- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{1} q$

NFA Example

$A = \{x \in \{0, 1\}^* \mid \text{the second symbol from the right is } 1\}$.



$$\begin{aligned}\Delta(p, \emptyset) &= \{p, q\} \\ \Delta(p, 0) &= \{q\} \\ \Delta(q, 0) &= \emptyset\end{aligned}$$

Some possible runs on word **01011**

- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{1} p \dots$
- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} q \xrightarrow{1} r$
- $p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{0} p \xrightarrow{1} p \xrightarrow{1} q$
- $p \xrightarrow{0} p \xrightarrow{1} q \xrightarrow{0} r \xrightarrow{1} \times$

When will an automaton accept a word? (There can be several runs on the word)

- \forall Nondeterminism Word is accepted if every run is accepting.
- \exists Nondeterminism Word is accepted if there exists a run which is accepting.

In this course we will use \exists nondeterminism throughout. It will just be called nondeterminism.

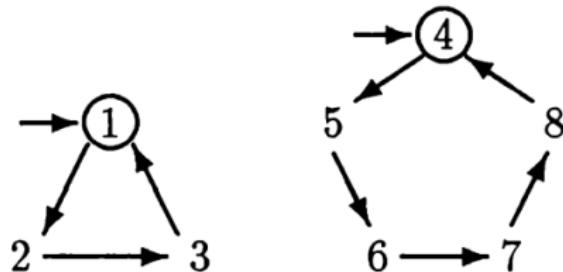
Example: Union

NNFA

Nondeterministic choice of start state

$$Q = \{1, 2, \dots, 8\}$$

$$S = \{1, 4\}$$

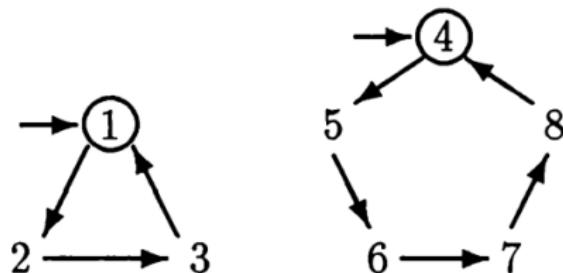


Some runs on word **11111**

- $1 \xrightarrow{1} 2 \xrightarrow{1} 3 \xrightarrow{1} 1 \xrightarrow{1} 2 \xrightarrow{1} 3 \quad \text{X}$

Example: Union

Nondeterministic choice of start state



Some runs on word **11111**

- $1 \xrightarrow{1} 2 \xrightarrow{1} 3 \xrightarrow{1} 1 \xrightarrow{1} 2 \xrightarrow{1} 3$
- $4 \xrightarrow{1} 5 \xrightarrow{1} 6 \xrightarrow{1} 7 \xrightarrow{1} 8 \xrightarrow{1} 4$



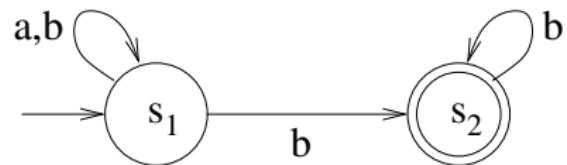
NFA Definition

An **NFA** is $(Q, \Sigma, \Delta, S, F)$

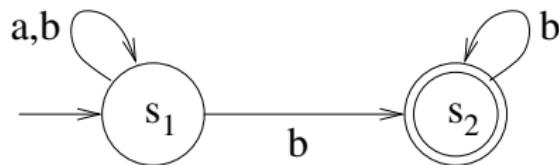
- Q Finite set of states.
- Σ Alphabet – finite set of letters.
- $S \subseteq Q$ set of initial states.
- $F \subseteq Q$ set of final states.
- Nondeterministic transition function $\Delta : Q \times \Sigma \rightarrow 2^Q$
Notation $2^Q = \{A \mid A \subseteq Q\}$.
We use $p \xrightarrow{a} q$ to denote $q \in \Delta(p, a)$.

Note that every DFA is an NFA.

Three Representations of NFA



Three Representations of NFA

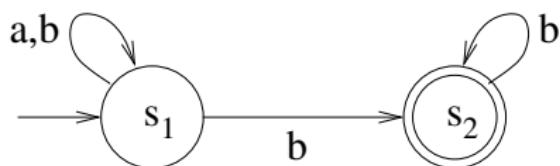


Five Tuple

$(\{s_1, s_2\}, \{a, b\}, \Delta, \{s_1\}, \{s_2\})$ where

$$\begin{array}{ll} \Delta(s_1, a) = \{s_1\}, & \Delta(s_1, b) = \{s_1, s_2\} \\ \Delta(s_2, a) = \emptyset, & \Delta(s_2, b) = \{s_2\} \end{array}$$

Three Representations of NFA



Transiday
diagram

Five Tuple

$Q \quad \Sigma \quad f_n \quad S_f$. final.
($\{s_1, s_2\}$, $\{a, b\}$, Δ , $\{s_1\}$, $\{s_2\}$) where

$$\begin{array}{ll} \Delta(s_1, a) = \{s_1\}, & \Delta(s_1, b) = \{s_1, s_2\} \\ \Delta(s_2, a) = \emptyset, & \Delta(s_2, b) = \{s_2\} \end{array}$$

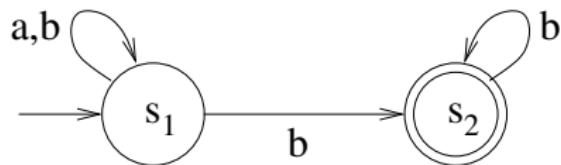
5 tuple

Transition Table

	a	b
s_1	$\{s_1\}$	$\{s_1, s_1\}$
s_2	\emptyset	$\{s_2\}$

Transition
table

Three Representations of NFA



Five Tuple

$(\{s_1, s_2\}, \{a, b\}, \Delta, \{s_1\}, \{s_2\})$ where

$$\begin{array}{ll} \Delta(s_1, a) = \{s_1\}, & \Delta(s_1, b) = \{s_1, s_2\} \\ \Delta(s_2, a) = \emptyset, & \Delta(s_2, b) = \{s_2\} \end{array}$$

Transition Table

		<i>a</i>	<i>b</i>
\rightarrow	s_1	$\{s_1\}$	$\{s_1, s_1\}$
s_2	F	\emptyset	$\{s_2\}$



- A **run** of NFA N on $x = a_0, a_1, \dots, a_{n-1}$ is a sequence of states q_0, q_1, \dots, q_n s.t. $q_i \xrightarrow{a_i} q_{i+1}$ for $0 \leq i < n$ and $q_0 \in I$.

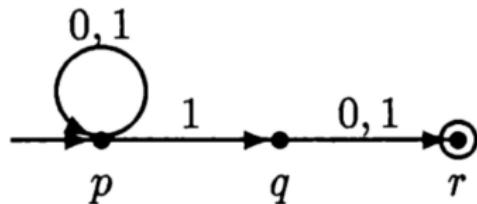
- A **run** of NFA N on $x = a_0, a_1, \dots, a_{n-1}$ is a sequence of states q_0, q_1, \dots, q_n s.t. $q_i \xrightarrow{a_i} q_{i+1}$ for $0 \leq i < n$ and $q_0 \in I$.
- For a given word x the NFA can have zero, one or more runs.

- A **run** of NFA N on $x = a_0, a_1, \dots, a_{n-1}$ is a sequence of states q_0, q_1, \dots, q_n s.t. $q_i \xrightarrow{a_i} q_{i+1}$ for $0 \leq i < n$ and $q_0 \in I$.
- For a given word x the NFA can have zero, one or more runs.
- A run **accepting** if last state $q_n \in F$.

- A **run** of NFA N on $x = a_0, a_1, \dots, a_{n-1}$ is a sequence of states q_0, q_1, \dots, q_n s.t. $q_i \xrightarrow{a_i} q_{i+1}$ for $0 \leq i < n$ and $q_0 \in I$.
- For a given word x the NFA can have zero, one or more runs.
- A run **accepting** if last state $q_n \in F$.
- Language accepted by N is
$$L(N) = \{x \in \Sigma^* \mid N \text{ has an accepting run on } x\}.$$

Extended Transition Function

Let $\hat{\Delta}(\{q\}, x)$ give all the states which can be reached at the end of some run starting from q on word x .



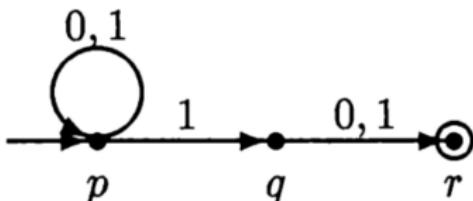
Extended Transition Function (2)

For $A \subseteq Q$, $\hat{\Delta}(A, x)$ give all the states which can be reached at the end of some run starting from some $q \in A$ on word x .

0100

P

0
1
2
3
4
5
6



$$\hat{\Delta}(\{p\}, 01) = \{p, q\}$$

$$\hat{\Delta}(\{p\}, 0100) = \{p\}$$

Properties

- $\hat{\Delta}(A, x) = \bigcup_{q \in A} \hat{\Delta}(\{q\}, x)$

Properties

- $\hat{\Delta}(A, x) = \bigcup_{q \in A} \hat{\Delta}(\{q\}, x)$
- $\hat{\Delta}(A, \epsilon) = A$

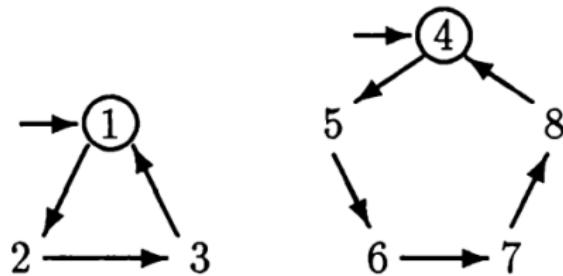
Properties

- $\hat{\Delta}(A, x) = \bigcup_{q \in A} \hat{\Delta}(\{q\}, x)$
- $\hat{\Delta}(A, \epsilon) = A$
- $\hat{\Delta}(A, a) = \bigcup_{q \in A} \Delta(q, a)$

Properties

- $\hat{\Delta}(A, x) = \bigcup_{q \in A} \hat{\Delta}(\{q\}, x)$
- $\hat{\Delta}(A, \epsilon) = A$
- $\hat{\Delta}(A, a) = \bigcup_{q \in A} \Delta(q, a)$
- $\hat{\Delta}(A, \textcolor{red}{x}a) = \bigcup_{q \in \hat{\Delta}(A, x)} \Delta(q, a)$

Example 2



$$\hat{\Delta}(\{1, 5\}, 11) = \{3, 7\}$$

Definition (Extended Nondeterministic Transition Function)

Define $\hat{\Delta} : 2^Q \times \Sigma^* \rightarrow 2^Q$ using structural induction:

$$\hat{\Delta}(A, \epsilon) = A$$

$$\hat{\Delta}(A, \textcolor{red}{xa}) = \bigcup_{q \in \hat{\Delta}(A, x)} \Delta(q, a)$$

Formal Definition of Extended Transition Function

Definition (Extended Nondeterministic Transition Function)

Define $\hat{\Delta} : 2^Q \times \Sigma^* \rightarrow 2^Q$ using structural induction:

$$\hat{\Delta}(A, \epsilon) = A$$

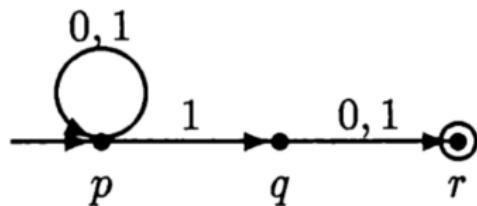
$$\hat{\Delta}(A, \textcolor{red}{xa}) = \bigcup_{q \in \hat{\Delta}(A, x)} \Delta(q, a)$$

Note that

$$\begin{aligned}\hat{\Delta}(A, a) &= \bigcup_{q \in \hat{\Delta}(A, \epsilon)} \Delta(q, a) \\ &= \bigcup_{q \in A} \Delta(q, a)\end{aligned}$$

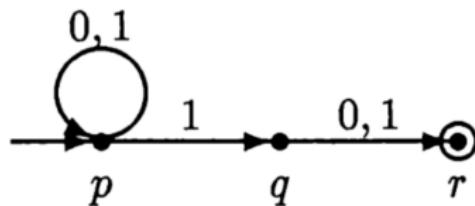



Extended Transition Function Computation



Calculate $\hat{\Delta}(\{p\}, 010)$

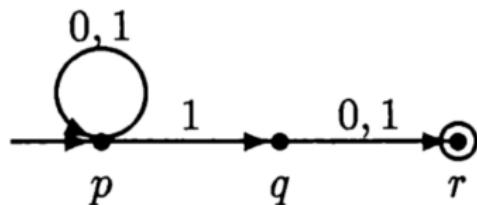
Extended Transition Function Computation



Calculate $\hat{\Delta}(\{p\}, 010)$

- $\hat{\Delta}(\{p\}, \epsilon) = \{p\}$

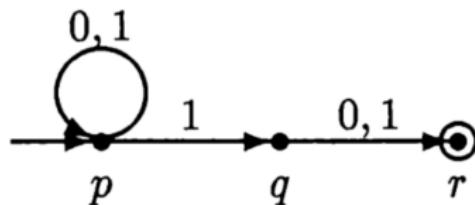
Extended Transition Function Computation



Calculate $\hat{\Delta}(\{p\}, 010)$

- $\hat{\Delta}(\{p\}, \epsilon) = \{p\}$
- $\begin{aligned}\hat{\Delta}(\{p\}, 0) &= \bigcup_{s \in \{p\}} \Delta(s, 0) \\ &= \{p\}\end{aligned}$

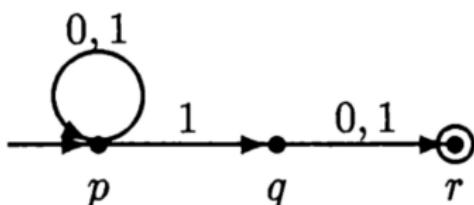
Extended Transition Function Computation



Calculate $\hat{\Delta}(\{p\}, 010)$

- $\hat{\Delta}(\{p\}, \epsilon) = \{p\}$
- $\hat{\Delta}(\{p\}, 0) = \bigcup_{s \in \{p\}} \Delta(s, 0)$
= $\{p\}$
- $\hat{\Delta}(\{p\}, 01) = \bigcup_{s \in \hat{\Delta}(\{p\}, 0)} \Delta(s, 1)$
= $\bigcup_{s \in \{p\}} \Delta(s, 1) = \Delta(p, 1)$
= $\{p, q\}$

Extended Transition Function Computation

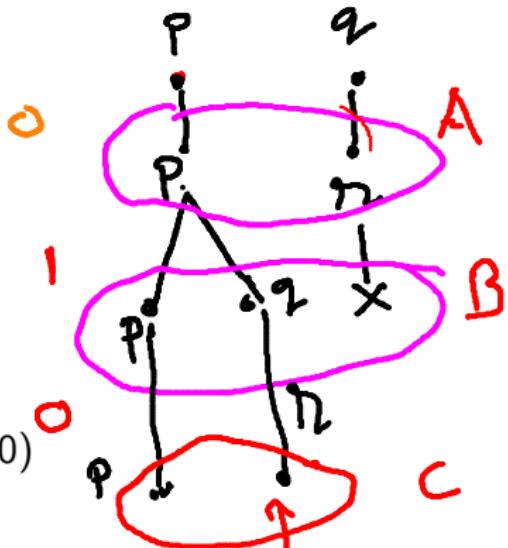


$$= \{p, q\}$$

$$\hat{\Delta}(\{p, q\}, 010)$$

Calculate $\hat{\Delta}(\{p\}, 010)$

- $\hat{\Delta}(\{p\}, \epsilon) = \{p\}$
- $\hat{\Delta}(\{p\}, 0) = \bigcup_{s \in \{p\}} \Delta(s, 0) = \{p\}$
- $\hat{\Delta}(\{p\}, 01) = \bigcup_{s \in \hat{\Delta}(\{p\}, 0)} \Delta(s, 1) = \bigcup_{s \in \{p\}} \Delta(s, 1) = \Delta(p, 1) = \{p, q\}$
- $\hat{\Delta}(\{p\}, \underline{010}) = \bigcup_{s \in \hat{\Delta}(\{p\}, 01)} \Delta(s, 0) = \bigcup_{s \in \{p, q\}} \Delta(s, 0) = \Delta(p, 0) \cup \Delta(q, 0) = \{p\} \cup \{r\}$



Theorem

$$\hat{\Delta}(A, xy) = \hat{\Delta}(\hat{\Delta}(A, x), y)$$

Proof: see Kozen Lemma 6.1

The proof is exactly identical to the similar proof for DFA given in the last class.

Some Properties

$$N = (Q, \Sigma, \Delta, S, F)$$

Theorem

$$\hat{\Delta}(A, xy) = \hat{\Delta}(\hat{\Delta}(A, x), y)$$
 Proof: see Kozen Lemma 6.1

The proof is exactly identical to the similar proof for DFA given in the last class.

Theorem

If $\hat{\Delta}(A, x) = B$ then for every $q \in B$ there exists $p \in A$ s.t. there is a run from p to q on word x .

Suppose $\hat{\Delta}(S, x) = B$
when will $x \in L(N)$?

Some Properties

$$N = (Q, \Sigma, \Delta, S, F)$$

Theorem

$$\hat{\Delta}(A, xy) = \hat{\Delta}(\hat{\Delta}(A, x), y)$$
 Proof: see Kozen Lemma 6.1

The proof is exactly identical to the similar proof for DFA given in the last class.

Theorem

If $\hat{\Delta}(A, x) = B$ then for every $q \in B$ there exists $p \in A$ s.t. there is a run from p to q on word x .

Theorem

$x \in L(N)$ if and only if $\hat{\Delta}(S, x) \cap F \neq \emptyset$



Definition

Language of NFA Given NFA $N = (Q, \Sigma, \Delta, S, F)$

- N accepts word x iff $(\hat{\Delta}(S, x) \cap F) \neq \emptyset$
- Language recognized by N is
 $L(N) = \{x \in \Sigma^* \mid N \text{ accepts } x\}$

Is NFA a machine?

NFA

Specification
or Model

DFA

implementation

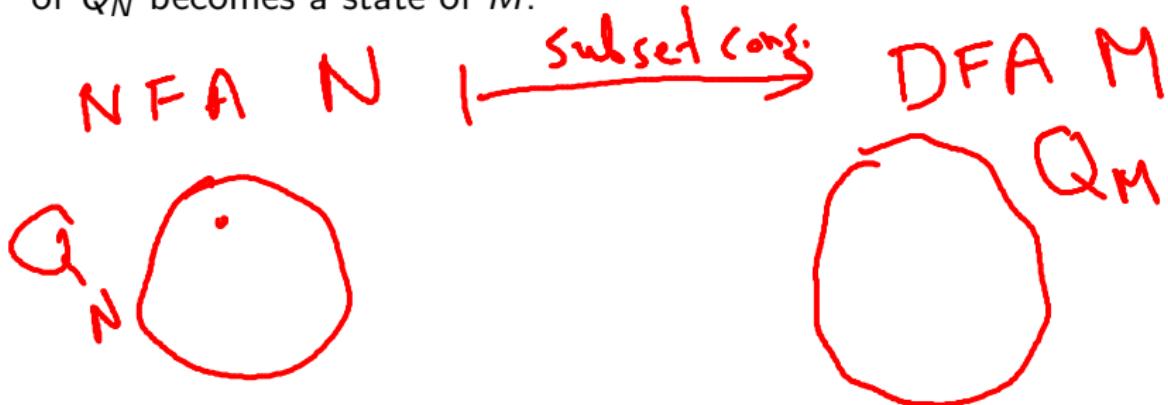
Is given NFA
implementable?

NFA Determinization

Theorem (Rabin-Scott)

For every NFA $N = (Q_N, \Sigma, \Delta_N, S_N, F_N)$ we can construct DFA $M = (Q_M, \Sigma, \delta_M, s_M, F_M)$ such that $L(N) = L(M)$.

Construction of M is called the **subset construction**. Each subset of Q_N becomes a state of M .



NFA Determinization

Theorem (Rabin-Scott)

For every NFA $N = (Q_N, \Sigma, \Delta_N, S_N, F_N)$ we can construct DFA $M = (Q_M, \Sigma, \delta_M, s_M, F_M)$ such that $L(N) = L(M)$.

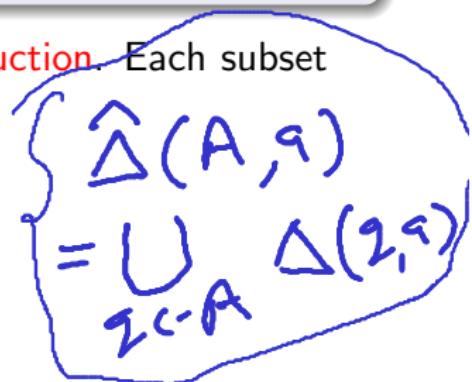
Construction of M is called the **subset construction**. Each subset of Q_N becomes a state of M .

$$Q_M \stackrel{\text{def}}{=} 2^{Q_N},$$

$\times \quad \delta_M(A, a) \stackrel{\text{def}}{=} \hat{\Delta}_N(A, a), \leftarrow$

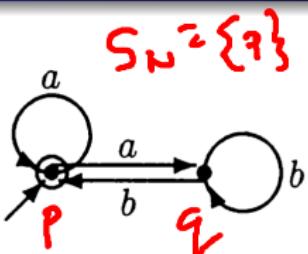
$$s_M \stackrel{\text{def}}{=} S_N,$$

$$F_M \stackrel{\text{def}}{=} \{A \subseteq Q_N \mid A \cap F_N \neq \emptyset\}.$$



Subset Construction by Example

NFA {?}

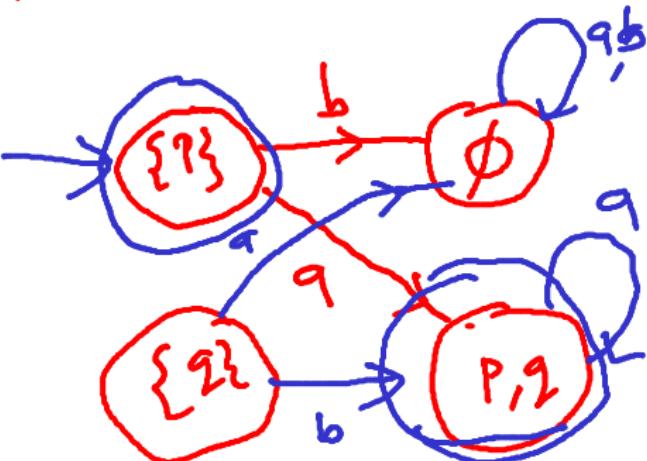


$$Q_N = \{p, q\}$$

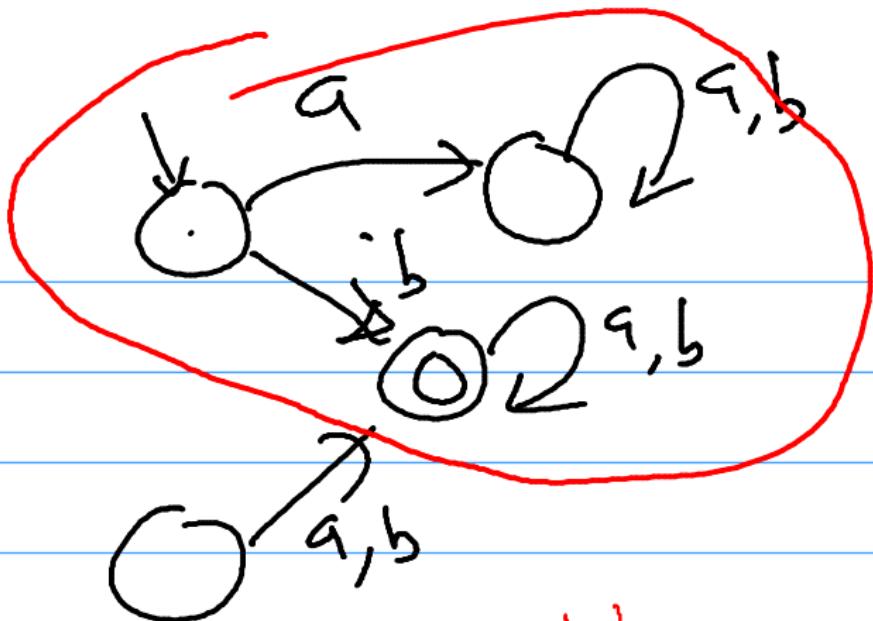
$$G_m = \{\emptyset, \{p\}, \{q\}, \{p, q\}\}$$

GMOFA

	a	b
s_0	Φ	Φ
$s_1 \rightarrow \{P\}$	$\{\emptyset, P\}$	\emptyset
$s_2 \rightarrow \{N\}$	\emptyset	$\{P, \emptyset\}$
$s_3 \rightarrow \{P, N\}$	$\{\emptyset, P, \emptyset\}$	$\{P, \emptyset\}$

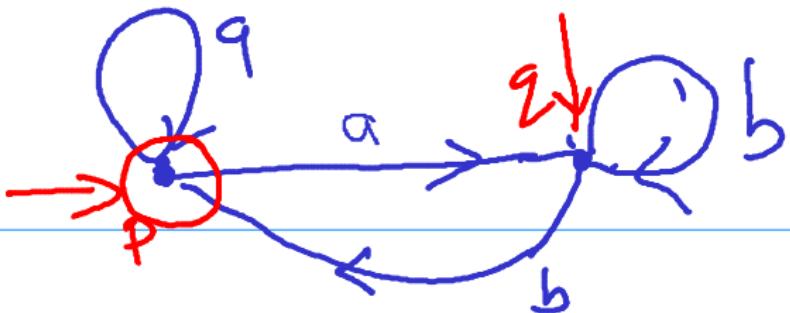


$$\hat{D}_N(\{p,q\}, a) = \{p,q\}$$



unreachable

NFA
N



$$\Sigma = \{a, b\}$$

DFA

$$Q_N = \{p, q\}$$

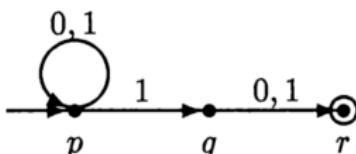
$$S_N = \{p, q\}$$

$$F_N = \{p\}$$

	a	b
\emptyset	\emptyset	\emptyset
$\{\text{p}\}^c$	$\{\text{p}, q\}$	\emptyset
$\{\text{q}\}$	\emptyset	$\{\text{p}, q\}$
$\rightarrow \{\text{p}, \text{q}\}^c$	$\{\text{p}, \text{q}\}$	$\{\text{p}, \text{q}\}$

Subset construction Example 2

NFA N



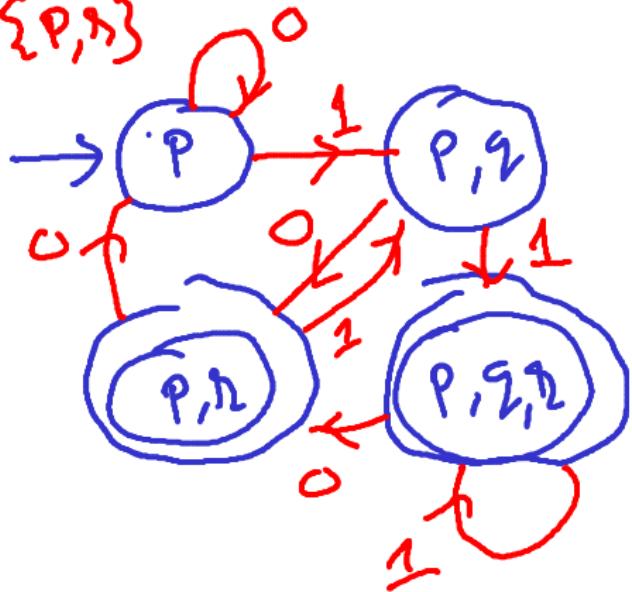
$$Q_N = \{p, q, r\}$$

$$S_N = \{p\}$$

$$F_N = \{r\}$$

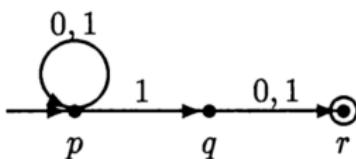
DFA A

	0	1
$\rightarrow \{p\}$	$\{p\}$	$\{p, q\}$
$\{p, q\}$	$\{p, q\}$	$\{p, q, r\}$
$\{p, q, r\}$	$\{p, q\}$	$\{p, q, r\}$
$\{p, q, r\}$	$\{p, q\}$	$\{p, q, r\}$



Subset construction Example 2

N



$$S_N = \{\emptyset\}$$

$$F_N = \{r\}$$

DFA
 M

→ $\{p\}$ $\{q\}$ $\{r\} F$

• $\{p, q\}$ $\{p, r\} F$ $\{q, r\} F$ $\{p, q, r\} F$

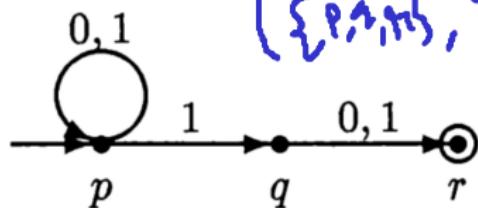
	0	1
\emptyset	\emptyset	\emptyset
$\{p\}$	$\{p\}$	$\{p, q\}$
$\{r\}$	$\{r\}$	$\{r\}$
\emptyset	\emptyset	\emptyset
$\{p, r\}$	$\{p, r\}$	$\{p, q, r\}$
$\{p\}$	$\{p\}$	$\{p, q\}$
$\{r\}$	$\{r\}$	$\{r\}$
$\{p, r\}$	$\{p, r\}$	$\{p, q, r\}$

Red annotations: $\Delta(\{\emptyset, \{p\}, \{r\}\}, 0) = \{p, r\}$ and $\Delta(\{\emptyset, \{p\}, \{r\}\}, 1) = \{p, q, r\}$

Subset construction by An Example

Keep only Reachable subsets
($\{p, q\}, \{q, r\}, \Delta, \{q\}, \{r\}$)

dfa N



dfa M

	0	1
$\rightarrow \{p\}$	$\{\underline{p}\}$	$\{\underline{p}, q\}$
$\{p, q\}$	$\{\underline{p}, r\}$	$\{\underline{p}, q, r\}$
$\{p, r\}F$	$\{\underline{p}\}$	$\{\underline{p}, q\}$
$\{p, q, r\}F$	$\{p, r\}$	$\{p, q, r\}$

Properties

- $\hat{\Delta}(A, x) = \bigcup_{q \in A} \hat{\Delta}(q, x)$
- $\hat{\Delta}(A, \epsilon) = A$
- $\hat{\Delta}(A, a) = \bigcup_{q \in A} \Delta(q, a)$

Theorem

If $\hat{\Delta}(A, x) = B$ then for every $q \in B$ there exists $p \in A$ s.t. there is a run from p to q on word x .

Theorem

$x \in L(M)$ if and only if $\hat{\Delta}(S, x) \cap F \neq \emptyset$

Theorem (Rabin-Scott)

For every NFA $N = (Q_N, \Sigma, \Delta_N, S_N, F_N)$ we can construct DFA $M = (Q_M, \Sigma, \delta_M, s_M, F_M)$ such that $L(N) = L(M)$. ↩

Construction of M is called the subset construction. Each subset of Q_N becomes a state of M .

NFA Determinization

Theorem (Rabin-Scott)

For every NFA $N = (Q_N, \Sigma, \Delta_N, S_N, F_N)$ we can construct DFA $M = (Q_M, \Sigma, \delta_M, s_M, F_M)$ such that $L(N) = L(M)$.

Construction of M is called the **subset construction**. Each subset of Q_N becomes a state of M .

$$Q_M \stackrel{\text{def}}{=} 2^{Q_N},$$

$$\delta_M(A, a) \stackrel{\text{def}}{=} \widehat{\Delta}_N(A, a),$$

$$s_M \stackrel{\text{def}}{=} S_N,$$

$$F_M \stackrel{\text{def}}{=} \{A \subseteq Q_N \mid A \cap F_N \neq \emptyset\}.$$

$$S_n(X, a) = \hat{\Delta}_N(X, a)$$

definition.

$$\forall X, x^*$$

$$\hat{g}(X, x^*) = \hat{\Delta}_N(X, x^*)$$

Proof Of Determinization Theorem

Lemma 6.9.

For any $A \subseteq Q_N$ and $x \in \Sigma^*$,

$$\widehat{\delta}_M(A, x) = \widehat{\Delta}_N(A, x).$$

Proof. Induction on $|x|$. ~~x~~.

Basis

For $x = \epsilon$, we want to show

$$\widehat{\delta}_M(A, \epsilon) = \widehat{\Delta}_N(A, \epsilon). \neq A.$$

But both of these are A , by definition of $\widehat{\delta}_M$ and $\widehat{\Delta}_N$.

Proof Of Determinization Theorem (2)

$\forall x \in \Sigma^*$

Induction Step:

Assume that

$$\hat{\delta}_M(A, x) = \hat{\Delta}_N(A, x).$$

We want to show the same is true for xa , $a \in \Sigma$.

$$\begin{aligned} \text{LHS} &= \hat{\delta}_M(A, xa) = \delta_M(\hat{\delta}_M(A, x), a) && \text{definition of } \hat{\delta}_M \\ &= \delta_M(\hat{\Delta}_N(A, x), a) && \text{induction hypothesis} \\ &= \hat{\Delta}_N(\hat{\Delta}_N(A, x), a) && \text{definition of } \delta_M \\ &= \hat{\Delta}_N(A, xa) && \text{Lemma 6.1.} \end{aligned}$$

Proof Of Determinization Theorem (3)

Theorem

For NFA N and constructed DFA M we have $L(M) = L(N)$

Proof. For any $x \in \Sigma^*$,

$$x \in L(M)$$

$$\iff \widehat{\delta}_M(s_M, x) \in F_M$$

$$\iff \widehat{\Delta}_N(S_N, x) \cap F_N \neq \emptyset$$

$$\iff x \in L(N)$$

definition of acceptance for M

definition of s_M and F_M , Lemma 1

definition of acceptance for N .

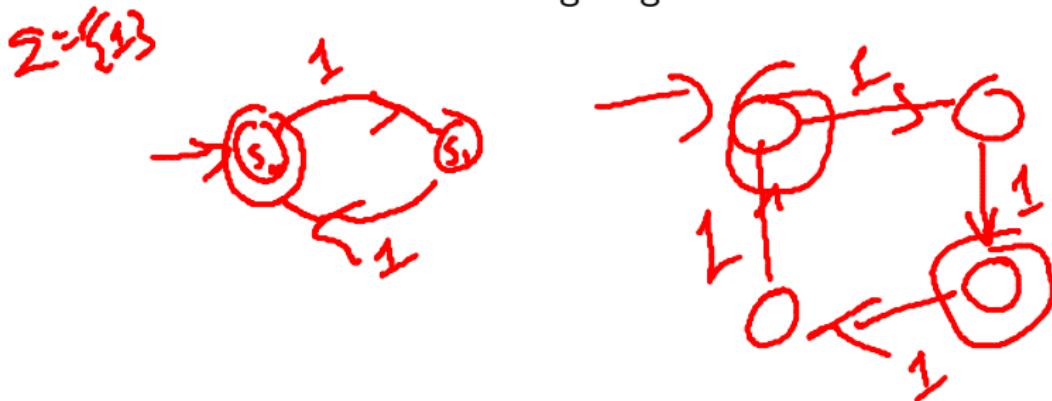
Size of the Constructed DFA

Given NFA N with n states what is the maximum number of states in the DFA M constructed by the subset construction?

Size of the Constructed DFA

Given NFA N with n states what is the maximum number of states in the DFA M constructed by the subset construction?

Is there a better construction giving smaller size DFA?



Size of the Constructed DFA

Given NFA N with n states what is the maximum number of states in the DFA M constructed by the subset construction?

Is there a better construction giving smaller size DFA?

Answer: No in worst case.