Automi e Linguaggi Formali

Linguaggi regolari e automi a stati finiti

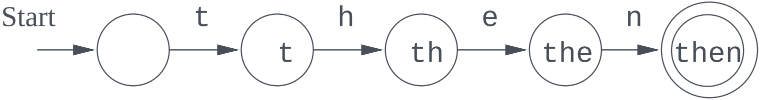
06 Ottobre 2014

A.A. 2014-2015
Enrico Mezzetti
emezzett@math.unipd.it

Università degli Studi di Padova

Recap

- Definizione informale di **automi a stati finiti**
- *Diagramma delle Transizioni*



- Elementi fondamentali:
  - stati finiti (iniziale, finale)
  - input
  - transizione ed *etichetta* (label)

- Rappresentazioni strutturali
  - Grammatiche, es. $E \Rightarrow E + E$
  - Espressioni regolari, es. `'[A-Z][a-z]*[][A-Z][A-Z]'`

Recap - 2

- Concetti di base della Teoria degli automi
  - **Alfabeto**: insieme finito e non vuoto di simboli
    $\Sigma = \{a, b, c, \ldots, z\}$ lettere minuscole
  - **Stringa**: sequenza finita di simboli da Σ (ϵ stringa vuota)
    $0011001,\ 111 \in \Sigma = \{0, 1\}$
  - **Lunghezza** di una stringa
    $|0110| = 4, |\epsilon| = 0$
  - **Potenza k-esima di un alfabeto**: come insieme delle stringhe di lunghezza k con simboli da Σ
    $\Sigma^2 = \{00, 01, 10, 11\}$ se $\Sigma = \{0, 1\}$
  - **Concatenazione**: unione di due stringhe una stringa
    $x = 01101, y = 110, xy = 01101110$
  - **Linguaggio**: insieme di stringhe scelte da Σ^*
    L e' un *linguaggio su* Σ se $L \subseteq \Sigma^*$
- Problema dell'**appartenenza** ad un linguaggio
  - Decidere se stringa w e' un elemento di un linguaggio L

Automi a stati finiti deterministici - DFA

- Determinismo vs. non-determinismo in automa
  - Automa non-deterministico *riducibile* ad uno deterministico

- Deterministic Finite Automaton (DFA) definito da:
  1. Un iniseme finito di stati Q
  2. Un insieme finito di simboli in input (alfabeto) Σ
  3. Una *funzione di transizione* $\delta(q, a) \mapsto p \in Q$
  4. Uno stato iniziale $q_0 \in Q$
  5. Un iniseme di stati finali (o accettati) $F \subseteq Q$

$$\text{DFA } A = (Q, \Sigma, \delta, q_0, F)$$

Linguaggio di un DFA

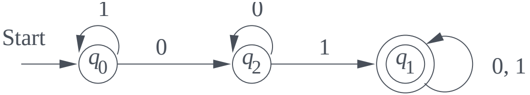
- Linguaggio di un DFA: *insieme di stringhe che il DFA accetta*
- DFA A che accetta tutte le stringhe dell'alfabeto binario che contengono la sequenza 01

$$01,\ 011100,\ 11101111 \ni L = \{x01y : x, y \in \{0,1\}^*\}$$

- Formalmente $A = (\{q_0, q_1, q_2\}, \{0,1\}, \delta, q_0, \{q_1\})$
- *Tabella di transizione*

	0	1
$\rightarrow q_0$	q_2	q_0
$\star q_1$	q_1	q_1
q_2	q_2	q_1

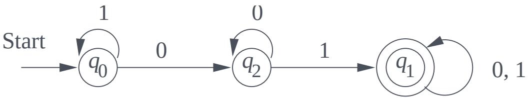
- *Diagramma di transizione*:



Accettazione su diagramma di transizione

- Un automa a stati finiti (FA) *accetta* una stringa $w = a_1 a_2 \cdots a_n$ se esiste un cammino (*path*) nel diagramma di transizione che
  1. Inizia nello stato iniziale q_0
  2. Finisce in uno stato finale $p \in F$
  3. Ha una sequenza di label $a_1 a_2 \cdots a_n$

- Esempio



✓ 11101110001

✗ 100

Definizione formale di linguaggio di un DFA

- *Funzione di transizione estesa*

$$\hat{\delta} : (q, w) \mapsto p \in Q$$

opera su stringhe $\uparrow$

- Definita ricorsivamente per induzione su $|w|$

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

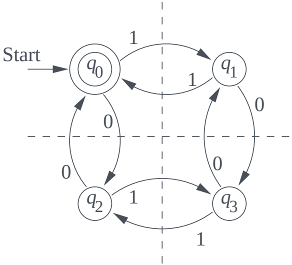
- Formalmente, il *linguaggio accettato da A* e'

$$L(A) = \{w : \hat{\delta}(q_0, w) \in F\}$$

- I linguaggi accettati da automi a stati finiti sono detti **linguaggi regolari**

Esempio di funzione di transizione estesa

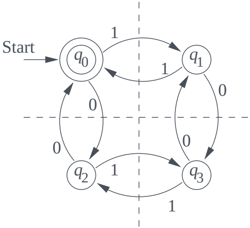
- Consideriamo un DFA che accetta tutte e sole le stringhe con un numero pari di 0 e un numero pari di 1
- DFA che "conta" il numero di 0 e 1 di una stringa in ingresso
  - $\rightarrow\star q_0$ numero di 0 e 1 letti sono entrambi pari
  - q_1 numero di 0 letti e' pari ma di 1 e' dispari
  - q_2 numero di 1 letti e' pari ma di 0 e' dispari
  - q_3 numero di 0 e 1 letti sono entrambi dispari



Accettazione formale

- $A = (\{q_0, q_1, q_2, q_3\}, \{0, 1\}, \delta, q_0, \{q_0\})$

	0	1
$\star \rightarrow q_0$	q_2	q_1
q_1	q_3	q_0
q_2	q_0	q_3
q_3	q_1	q_2

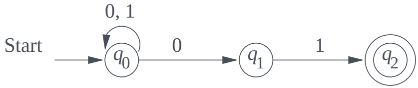


- Verifichiamo che $\hat{\delta}(q_0, w) = q_0$ ad es. per w=110101
  - $\hat{\delta}(q_0, \epsilon) = q_0$
  - $\hat{\delta}(q_0, 1) = \delta(\hat{\delta}(q_0, \epsilon), 1) = \delta(q_0, 1) = q_1$
  - $\hat{\delta}(q_0, 11) = \delta(\hat{\delta}(q_0, 1), 1) = \delta(q_1, 1) = q_0$
  - $\hat{\delta}(q_0, 110) = \delta(\hat{\delta}(q_0, 11), 0) = \delta(q_0, 0) = q_2$
  - $\hat{\delta}(q_0, 1101) = \delta(\hat{\delta}(q_0, 110), 1) = \delta(q_2, 1) = q_3$
  - $\hat{\delta}(q_0, 11010) = \delta(\hat{\delta}(q_0, 1101), 0) = \delta(q_3, 0) = q_1$
  - $\hat{\delta}(q_0, 110101) = \delta(\hat{\delta}(q_0, 11010), 1) = \delta(q_1, 1) = q_0$ □

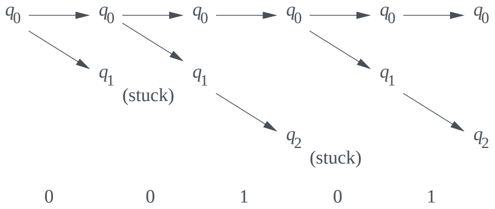
Esercizi su DFA

✎

- Automi a stati finiti *non deterministici*
  - puo' trovarsi contemporaneamente in diversi stati
- Esempio: *automa che accetta solo le stringhe di qualsiasi lunghezza che terminano in 01*



  - Elaborazione dell'input 00101



<u>Note</u>: Non ci sono archi in uscita da q_2, n da q_0 per input 0
Sono ammessi piu' archi in uscita per uno stesso symbolo

Definizione formale di NFA

- Non-deterministic Finite Automaton (NFA) definito da:
  1. Un iniseme finito di stati Q
  2. Un insieme finito di simboli in input (alfabeto) Σ
  3. Una *funzione di transizione* $\delta(q, a) \mapsto Q' \subseteq Q$
  4. Uno stato iniziale $q_0 \in Q$
  5. Un iniseme di stati finali (o accettati) $F \subseteq Q$

$$\text{NFA } A = (Q, \Sigma, \delta, q_0, F)$$

- Differenza evidente con DFA
  - $\delta_{DFA} : (q, a) \mapsto p \in Q$
  - $\delta_{NFA} : (q, a) \mapsto \mathcal{P}(Q)$

Specifiche di NFA

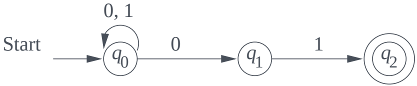
- Esempio: *automa che accetta solo le stringhe di qualsiasi lunghezza che terminano in 01*

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

- Tabella di transizione

	0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\emptyset$	$\{q_2\}$
$\star q_2$	$\emptyset$	$\emptyset$

- Diagramma di transizione



Definizione formale di un linguaggio di un NFA

- Simile a DFA, sfrutta la *funzione di transizione estesa*

$$\hat{\delta} : (q, w) \mapsto \mathcal{P}(Q)$$

- Sempre definita ricorsivamente per induzione su $|w|$

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

  Induzione $\hat{\delta}(q, xa) = \bigcup_{p \in \hat{\delta}(q,x)} \delta(p, a)$

  Ovvero: applica transizione con label a a partire da tutti i $p \in \hat{\delta}(q, x)$

- Esempio: $\hat{\delta}(q, 00101)$ ✎

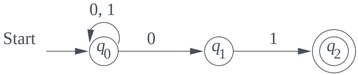
- Formalmente, il *linguaggio accettato da A* e'

$$L(A) = \{w : \hat{\delta}(q_0, w) \cap F \neq \emptyset\}$$

Esempio di accettazione formale

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

	0	1
$\rightarrow q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\emptyset$	$\{q_2\}$
$\star q_2$	$\emptyset$	$\emptyset$



- Verifichiamo che accetta il linguaggio $L = \{x01 : x \in \Sigma^*\}$
  - Per induzione *mutua* su $|w|$ per i tre enunciati seguenti

    0. $w \in \Sigma^* \Rightarrow q_0 \in \hat{\delta}(q_0, w)$
    1. $q_1 \in \hat{\delta}(q_0, w) \Leftrightarrow w = x0$
    2. $q_2 \in \hat{\delta}(q_0, w) \Leftrightarrow w = x01$

Accettazione formale - 2

Base Se $|w| = 0$ allora $w = \epsilon$. Allora l'enunciato (0) segue dalla defnizione Per (1) e (2) entrambi i lati sono falsi per ϵ

Induzione Assumiamo che $w = xa$, dove $a \in \{0,1\}$, $|x| = n$ e gli enunciati (0)–(2) valgono per x. Dobbiamo dimostrare che gli enunciati valgono anche per w.

(0) $q_0 \in \hat{\delta}(q_0, x)$: poiche' q_0 ammette transizioni su se' stesso sia per 0 che per 1, anche $q_0 \in \hat{\delta}(q_0, w)$

(1-if) w termina con 0: allora $a = 0$ e per (0) sappiamo che $q_0 \in \hat{\delta}(q_0, x)$; poiche' esiste una transizione da q_0 a q_1 su input 0 concludiamo che $q_1 \in \hat{\delta}(q_0, w)$

(1-only-if) $q_1 \in \hat{\delta}(q_0, w)$: sappiamo che c' solo un modo di raggiungere q_1 ovvero attraverso una seq. $x0$.

(2-if) w termina con 01: allora $a = 1$ e x termina con 0; per (1) sappiamo che $q_1 \in \hat{\delta}(q_0, x)$. Poiche' esiste una transizione da q_1 a q_2 su input 1, allora $q_2 \in \hat{\delta}(q_0, w)$

(2-only-if) $q_2 \in \hat{\delta}(q_0, w)$: sappiamo che c' solo un modo di raggiungere, ovvero che W sia nelle forma $x1$ dove $q_1 \in \hat{\delta}(q_0, x)$; per (1) sappiamo che x termina con 0 e quindi w termina con 01

Equivalenza di DFA e NFA

- DFA e NFA sono **equivalenti**
  - Accettano la stessa classe di linguaggi (NFA generalmente piu' facili da costruire e meno complessi da tradurre)
- Per ogni NFA N esiste un DFA D tale che $L(D) = L(N)$ (e viceversa)
- Dimostrazione usa *costruzione per sottonsiemi*
  - Considera tutti i sottoinsiemi dell'insieme di stati dell'NFA
  - Descrizione formale di A nei termini di stati e transizioni di B
- Dato un NFA N $= (Q_N, \Sigma, \delta_N, q_0, F_N)$ costruiremo un DFA D $= (Q_D, \Sigma, \delta_D, \{q_0\}, F_D)$ tale che $L(D) = L(N)$

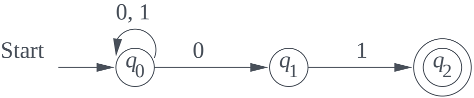
Construzione per sottonsiemi

- Osservazioni su $\mathrm{N} = (Q_N, \Sigma, \delta_N, q_0, F_N)$ e
  $\mathrm{D} = (Q_D, \Sigma, \delta_D, \{q_0\}, F_D)$
  - Condividono lo stesso alfabeto Σ
  - $Q_D = \{S : S \subseteq Q_N\}$ (*powerset* $|Q_D| = 2^{|Q_N|}$)
    (in effetti non tutti raggiungibili da q_0)
  - $F_D = \{S \subseteq Q_N : S \cap F_N \neq \emptyset\}$
  - $\forall S \subseteq Q_N$ e $\forall a \in \Sigma$,

$$\delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$$

Per calcolare $\delta_D(S, a)$ dobbiamo operare su tutti i $p \in S$

Construzione per sottonsiemi - Esempio



- Costruiamo δ_D dell' NFA gia' visto

	0	1
$\emptyset$	$\emptyset$	$\emptyset$
$\rightarrow \{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_1\}$	$\emptyset$	$\{q_2\}$
$\star\{q_2\}$	$\emptyset$	$\emptyset$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$\star\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\star\{q_1, q_2\}$	$\emptyset$	$\{q_2\}$
$\star\{q_0, q_1, q_2\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$

$\xrightarrow{DFA}$

	0	1
A	A	A
$\rightarrow B$	E	B
C	A	D
$\star D$	A	A
E	E	F
$\star F$	E	B
$\star G$	A	D
$\star H$	E	F

Che caratteristica hanno gli stati in δ_D?

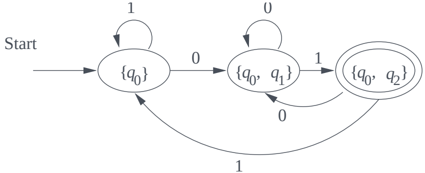
Valutazione "differita" dei sottoinsiemi

- Per evitare la crescita esponenziale degli stati
  - Tabella di transizione per D con solo stati accessibili S
- Determinazione induttiva di S accessibili

Base $q_0 \in S$ per definizione di stato iniziale

Induzione Sia S accessibile, allora lo sono anche gli stati in $\delta_D(S, a), \forall a$

- Tornando all'esempio:
  - q_0
  - $\delta_D(q_0, a) = \{\{q_0\}, \{q_0, q_1\}\}$
  - $\delta_D(\{q_0, q_1\}, a) = \{\{q_0, q_1\}, \{q_0, q_2\}\}$
  - $\delta_D(\{q_0, q_2\}, a) = \{\{q_0\}, \{q_0, q_1\}\}$



Giustificazione formale costruzione per sottoinsiemi

Th 2.11: Sia D il DFA ottenuto da un NFA N con la costruzione per sottoinsiemi. Allora $L(D) = L(N)$

Prova: $L(D) = L(N)$ se D e N accettano le stesse stringhe

1. Dimostriamo per induzione su $|w|$ che
$$\hat{\delta}_D(\{q_0\}, w) = \hat{\delta}_N(q_0, w)$$

Base $|w| = 0$ $(w = \epsilon)$ per definizione

Induzione $|w| = n + 1$, $w = xa$ e enunciato vale per x

$$\begin{aligned}
\hat{\delta}_D(\{q_0\}, xa) &\overset{\text{def}}{=} \delta_D(\hat{\delta}_D(\{q_0\}, x), a) && (1)\\
&\overset{\text{i.h.}}{=} \delta_D(\hat{\delta}_N(q_0, x), a) && (2)\\
&\overset{\text{cst}}{=} \bigcup_{p \in \hat{\delta}_N(q_0, x)} \delta_N(p, a) && (3)\\
&\overset{\text{def}}{=} \hat{\delta}_N(q_0, xa) && (4)
\end{aligned}$$

2. Se D e N accettano w sia $\hat{\delta}_D(q_0, w)$ e $\hat{\delta}_N(q_0, w)$ contengono uno stato in F_N ∎

Generalizzazione

Th 2.12: Un linguaggio L e' accettato da un DFA se e solo se L e' accettato da un NFA.

Prova: Parte "se" Th. 2.11

Parte "solo se"

Notiamo che un qualsiasi DFA puo' essere convertito in un NFA equivalente in cui in ogni momento esiste solo una scelta di transizione possibile.

Per induzione su $|w|$:

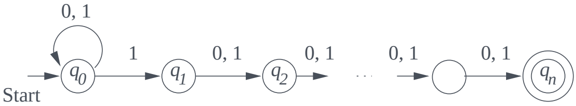
Base Se $\delta_D(q, a) = p$, allora $\delta_N(q, a) = \{p\}$.

Induzione Si dimostra che se $\hat{\delta}_D(q_0, w) = p$, allora $\hat{\delta}_N(q_0, w) = \{p\}$.

Crescita esponenziale degli stati

- Trasformazione vista e' un caso favorevole ($\sim$stati)
- Esempio conversione con $Q_N = n + 1$ e $Q_D = 2^n$

Insieme delle stringhe in $\{0,1\}$ tali che l'n-esimo simbolo dalla fine sia 1.



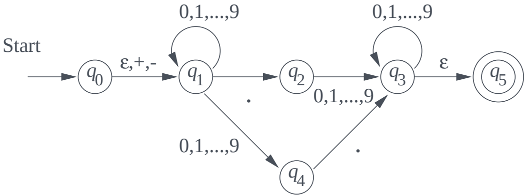
- Esistono 2^n sotto-sequenze distinte di lunghezza n
- Se DFA avesse meno di 2^n stati allora almeno uno stato q sarebbe raggiungibile da due sequenze distinte $a_1 a_2 \cdots a_n \neq b_1 b_2 \cdots b_n \Rightarrow \exists\ i$ s.t. $a_i \neq b_i$ (*pigeonhole principle*)

Crescita esponenziale degli stati - 2

- **Caso 1:** $i = 1$
  - $1a_2 \cdots a_n$ e $0b_2 \cdots b_n$ (o viceversa)
  - ✗ q e' allo stesso momento uno stato accettante e non accettante
- **Caso 2:** $i \neq 1$
  - $a_1 \cdots a_{i-1} 1 a_{i+1} \cdots a_n$
    $b_1 \cdots b_{i-1} 0 b_{i+1} \cdots b_n$
  - ✗ State $p := q0^{i-1}$:
    $\hat{\hat{\delta}}_N(q_0, a_1 \cdots a_{i-1} 1 a_{i+1} \cdots a_n 0^{i-1}) =$
    $\hat{\hat{\delta}}_N(q_0, b_1 \cdots b_{i-1} 0 b_{i+1} \cdots b_n 0^{i-1})$
    p e' allo stesso momento uno stato accettante e non accettante

Automi a stati finiti con transizioni ϵ

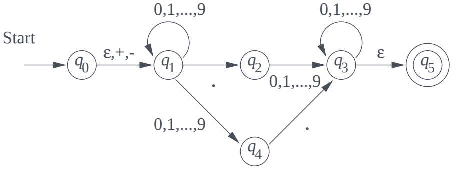
- Estensione di FA che ammettono transizioni su ϵ
  - Zucchero sintattico non estende l'espressivita'
  - ϵ-FA strettamente legati alle *espressioni regolari*
- Esempio: ϵ-NFA che accetta numeri decimali



  - Numeri decimali in notazione anglosassone:
    1. Un segno + o -, opzionale
    2. Una stringa di cifre decimali
    3. Un punto decimale
    4. Un'altra stringa di cifre decimali

  Una sola sottosequenza tra (2) e (4) puo' essere vuota

- Un ϵ-NFA e' una quintupla $(Q, \Sigma, \delta, q_0, F)$ dove
$$\delta : Q \times \Sigma \cup \{\epsilon\} \mapsto \mathcal{P}(Q)$$
- Es. ϵ-NFA numeri decimali
$$E = (\{q_0, q_1, \ldots, q_5\}, \{., +, -, 0, 1, \ldots, 9\}\ \delta, q_0, \{q_5\})$$



	ϵ	+,-	.	$0, \ldots, 9$
$\rightarrow q_0$	$\{q_1\}$	$\{q_1\}$	$\emptyset$	$\emptyset$
q_1	$\emptyset$	$\emptyset$	$\{q_2\}$	$\{q_1, q_4\}$
q_2	$\emptyset$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_3	$\{q_5\}$	$\emptyset$	$\emptyset$	$\{q_3\}$
q_4	$\emptyset$	$\emptyset$	$\{q_3\}$	$\emptyset$
$\star q_5$	$\emptyset$	$\emptyset$	$\emptyset$	$\emptyset$

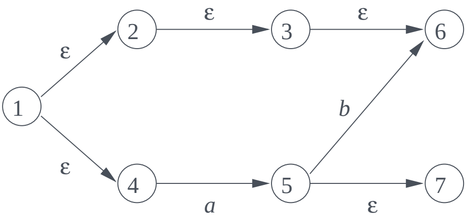
ϵ-chiusura

- Quali sono le stringhe e i linguaggi accettati da ϵ-NFA?
- Concetto fondamentale ϵ-CLOSURE di uno stato q
  - Insieme di stati raggiungibili da q attraverso ϵ-transizioni
  - Definizione induttiva:

Base $q \in \text{ECLOSE}(q)$

Induzione $p \in \text{ECLOSE}(q)$ and $r \in \delta(p, \epsilon) \Rightarrow r \in \text{ECLOSE}(q)$

- Esempio:



$$\text{ECLOSE}(1) = \{1, 2, 3, 4, 6\}$$

Funzione di transizione per ϵ-NFA

- Definizione di $\hat{\delta}$ per automi ϵ-NFA
  - Per induzione su $|w|$ (ϵ non contribuisce a w)

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

Induzione $\hat{\delta}(q, xa) = \bigcup_{p \in \delta(\hat{\delta}(q,x),a)} \text{ECLOSE}(p)$

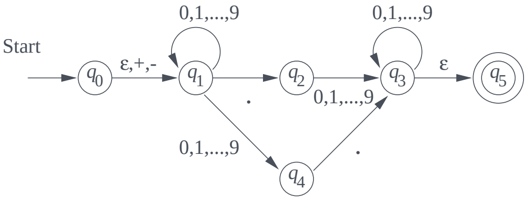
- Esempio: $\hat{\delta}(q_0, 5.6)$ per l'NFA dei numeri decimali ✎

Da ϵ-NFA a DFA

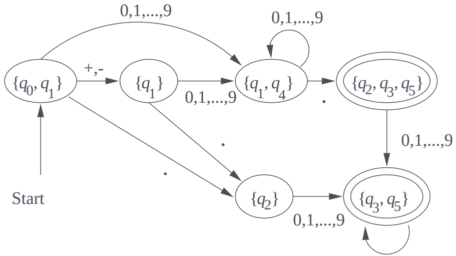
- Dato un ϵ-NFA $E = (Q_E, \Sigma, \delta_E, q_0, F_E)$
  - Costruire un DFA $D = (Q_D, \Sigma, \delta_D, q_D, F_D)$
  - Tale che $L(D) = L(E)$

- Caratterizzazione di D
  - $Q_D = \{S : S \subseteq Q_E \text{ e } S = \text{ECLOSE}(S)\}$
  - $q_D = \text{ECLOSE}(q_0)$
  - $F_D = \{S : S \in Q_D \text{ e } S \cap F_E \neq \emptyset\}$
  - $\forall a,\ \delta_D(S, a) = \bigcup\{\text{ECLOSE}(p) : p \in \delta(t, a) \text{ per alcuni } t \in S\}$

- ϵ-NFA E



- DFA D corrispondente



Equivalenza tra ϵ-NFA e DFA

Th 2.22: Un linguaggio L e' accettato da un ϵ-NFA E se e solo se L e' accettato da un DFA

Prova: Parte "se" Facile

Parte "solo se"

Usiamo D costruito come sopra e mostriamo per induzione su $|w|$ che $\hat{\delta}_E(q_0, w) = \hat{\delta}_D(q_D, w)$

Base $|w| = 0$ $(w = \epsilon)$

$$\hat{\delta}_E(q_0, \epsilon) = \text{ECLOSE}(q_0) = q_D = \hat{\delta}(q_D, \epsilon)$$

Induzione $|w| = n + 1$, $w = xa$ e enunciato vale per x

$$\begin{aligned}
\hat{\delta}_E(q_0, xa) &= \bigcup_{p \in \delta_E(\hat{\delta}_E(q_0, x), a)} \text{ECLOSE}(p) \\
&= \bigcup_{p \in \delta_D(\hat{\delta}_D(q_D, x), a)} \text{ECLOSE}(p) \\
&= \bigcup_{p \in \hat{\delta}_D(q_D, xa)} \text{ECLOSE}(p) \\
&= \hat{\delta}_D(q_D, xa)
\end{aligned}$$