I dettagli della costruzione per sottoinsiemi:

- $Q_D = \{S : S \subseteq Q_N\}$.

Nota: $|Q_D| = 2^{|Q_N|}$, anche se la maggior parte degli stati in Q_D sono "garbage", cioè non raggiungibili dallo stato iniziale.

- $F_D = \{S \subseteq Q_N : S \cap F_N \neq \emptyset\}$
- Per ogni $S \subseteq Q_N$ e $a \in \Sigma$,

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

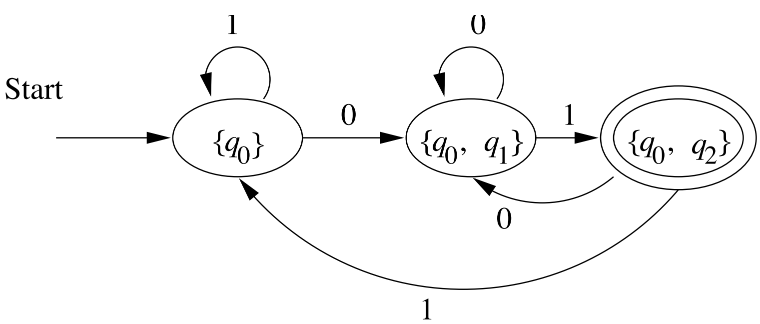
Costruiamo δ_D dall'NFA già visto, quello cioè che accetta le stringhe che terminano con 01.

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

Nota: Gli stati di D corrispondono a sottoinsiemi di stati di N, ma potevamo denotare gli stati di D in un altro modo, per esempio $A - F$.

	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

Per evitare la crescita esponenziale degli stati può essere utile costruire la tabella di transizione per D solo per gli stati accessibili (o raggiungibili) S come segue:
Base: $S = \{q_0\}$ è accessibile in D
Induzione: Se lo stato S è accessibile, lo sono anche gli stati in $\bigcup_{a \in \Sigma} \delta_D(S, a)$ accessibili a partire da S.
Esempio: Il "sottoinsieme" DFA con i soli stati accessibili: B ($\{q_0\}$), E ($\{q_0, q_1\}$) e F ($\{q_0, q_2\}$).



Teorema 2.11: Sia D il DFA ottenuto da un NFA N con la costruzione a sottoinsiemi. Allora $L(D) = L(N)$.
Dimostrazione: Prima mostriamo per induzione su $|w|$ che

$$\hat{\delta}_D(\{q_0\}, w) = \hat{\delta}_N(q_0, w)$$

N.B. Entrambe le funzioni restituiscono un insieme di stati di Q_N.
Base: $w = \epsilon$. Per definizione $\hat{\delta}_D(\{q_0\}, \epsilon) = \hat{\delta}_N(q_0, \epsilon) = \{q_0\}$.

Induzione: $w = xa$, con $|x| = n$ e $a \in \Sigma$. Per ipotesi induttiva $\hat{\delta}_D(\{q_0\}, x) = \hat{\delta}_N(q_0, x)$.

$$\hat{\delta}_D(\{q_0\}, xa) \stackrel{\text{def}}{=} \delta_D(\hat{\delta}_D(\{q_0\}, x), a)$$

$$\stackrel{\text{i.h.}}{=} \delta_D(\hat{\delta}_N(q_0, x), a)$$

$$\stackrel{\text{cst}}{=} \bigcup_{p \in \hat{\delta}_N(q_0, x)} \delta_N(p, a)$$

$$\stackrel{\text{def}}{=} \hat{\delta}_N(q_0, xa)$$

Osservando che sia D che N accettano w sse $\hat{\delta}_D(\{q_0\}, w)$ e $\hat{\delta}_N(q_0, w)$ contengono uno stato in F_N, ne segue che $L(D) = L(N)$.

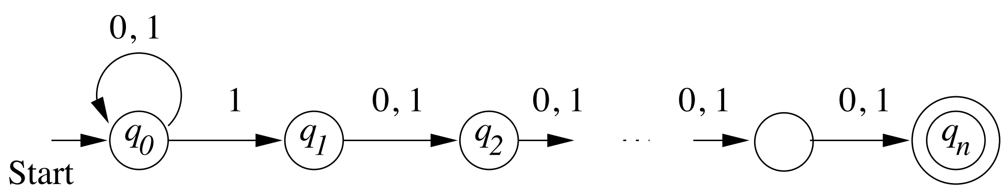
Teorema 2.12: Un linguaggio L è accettato da un DFA se e solo se L è accettato da un NFA.
Dimostrazione: La parte "se" è il Teorema 2.11.
Per la parte "solo se" notiamo che un qualsiasi DFA può essere convertito in un NFA equivalente modificando la δ_D in δ_N secondo la regola seguente:

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

Per induzione su $|w|$ si può facilmente mostrare che se $\hat{\delta}_D(q_0, w) = p$, allora $\hat{\delta}_N(q_0, w) = \{p\}$.
Di conseguenza la stringa w viene accettata da D sse viene accettata da N e l'enunciato del teorema segue.

Crescita esponenziale degli stati

Esiste un NFA N con $n+1$ stati che non ha nessun DFA equivalente con meno di 2^n stati



$$L(N) = \{x1c_2c_3 \cdots c_n : x \in \{0,1\}^*, c_i \in \{0,1\}\}$$

D deve ricordare gli ultimi n simboli che ha letto.
Dato che ci sono 2^n sequenze di n bit, se esistesse un DFA equivalente con meno di 2^n stati, allora esisterebbe uno stato q e due sequenze $a_1a_2 \cdots a_n$ e $b_1b_2 \cdots b_n$, con almeno $i : a_i \neq b_i$:
$q \in \hat{\delta}_N(q_0, a_1a_2 \cdots a_n), q \in \hat{\delta}_N(q_0, b_1b_2 \cdots b_n), a_1a_2 \cdots a_n \neq b_1b_2 \cdots b_n$

Caso 1: $i = 1$

$1a_2 \cdots a_n$

$0b_2 \cdots b_n$

Allora q deve essere sia uno stato di accettazione che uno stato di non accettazione.

Caso 2: $i > 1$

$a_1 \cdots a_{i-1}1a_{i+1} \cdots a_n$

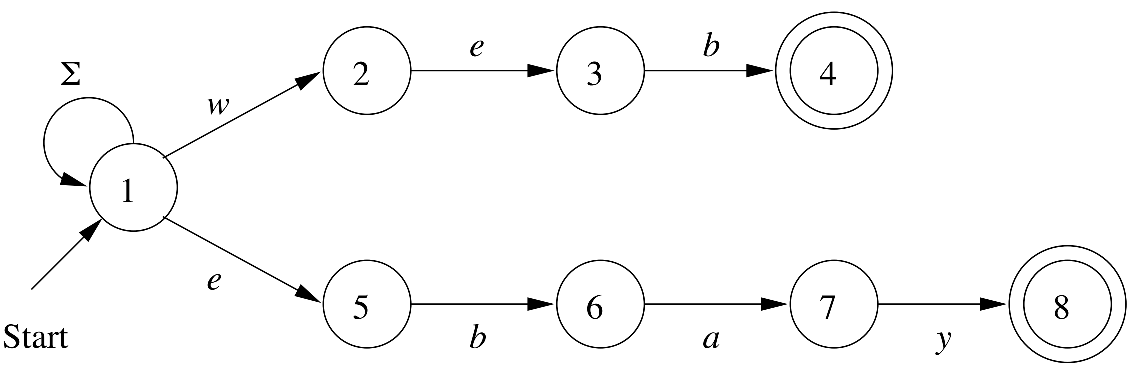
$b_1 \cdots b_{i-1}0b_{i+1} \cdots b_n$

Ora $\hat{\delta}_N(q_0, a_1 \cdots a_{i-1}1a_{i+1} \cdots a_n0^{i-1}) =$
$\hat{\delta}_N(q_0, b_1 \cdots b_{i-1}0b_{i+1} \cdots b_n0^{i-1})$

e $\hat{\delta}_N(q_0, a_1 \cdots a_{i-1}1a_{i+1} \cdots a_n0^{i-1}) \in F_D$
$\hat{\delta}_N(q_0, b_1 \cdots b_{i-1}0b_{i+1} \cdots b_n0^{i-1}) \notin F_D$

NFA per ricerche testuali

Un NFA che accetta l'insieme di parole chiave $\{\texttt{ebay}, \texttt{web}\}$



Naturalmente l'NFA va poi implementato.