Kapitel 2

Reguläre Sprachen

2.3

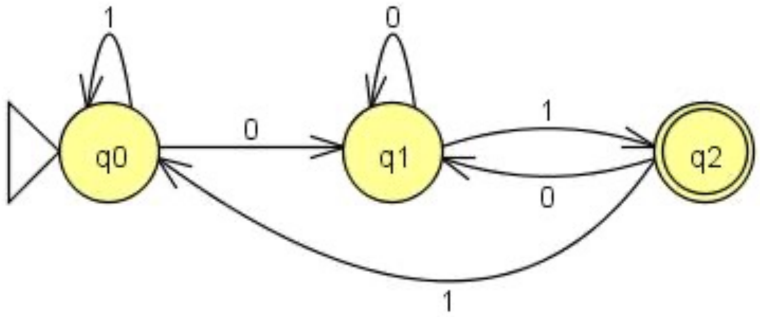
Nichtdeterministischer Endlicher Automat (NEA)

Prof. Dr. Robert Preis
Fachbereich Informatik
Fachhochschule Dortmund
Robert.Preis@fh-dortmund.de

Was ist Nichtdeterminismus ?

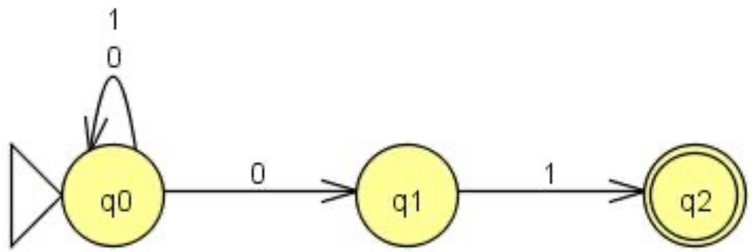
Bisher: DEA (Verhalten eindeutig bestimmt)

- Pro Buchstabe nur ein Folgezustand möglich, d.h. für jeden Buchstaben gibt es genau einen Pfeil
- aufwendig zu entwerfen
- leicht durch Software zu simulieren
- Beispiel: Alle 0/1-Folgen, die auf 01 enden:
- *Wie sieht die Tabelle dazu aus?*



Jetzt: NEA (Verhalten nicht eindeutig bestimmt)

- Automat wählt Folgezustand aus mehreren Möglichkeiten, d.h. für jeden Buchstaben kann es keinen, einen oder mehrere Pfeile geben
- leicht zu entwerfen
- aufwendig durch Software zu simulieren
- Beispiel: Alle 0/1-Folgen, die auf 01 enden:
- *Wie sieht die Tabelle dazu aus?*



Wozu Nichtdeterministische Automaten?

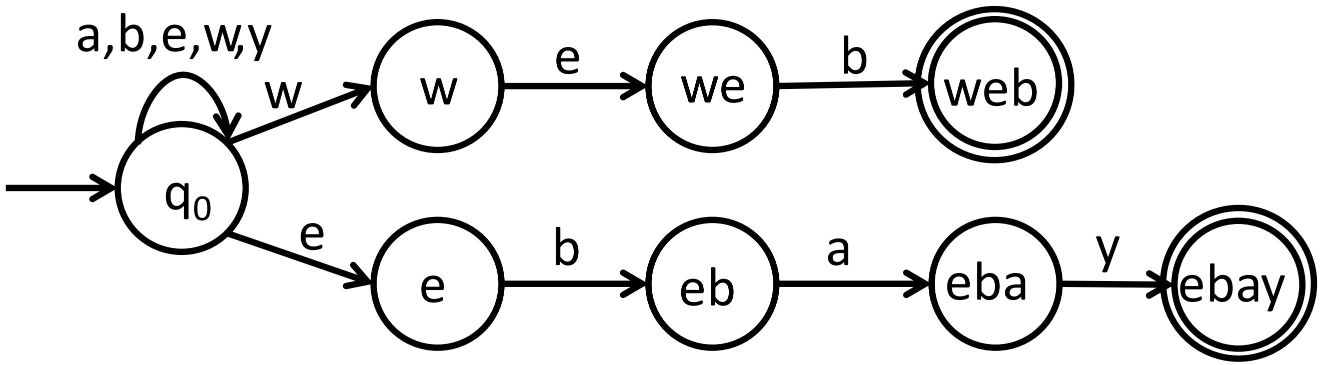
Aufgabe: Suche nach den Wörtern in Texten.

Beispiel: Suche nach web und ebay am Ende eines Wortes.

Leicht: Wenn man die Wörter in einem DEA sofort findet, dann ist es leicht.

Schwer: *Wenn der nächste Buchstabe nicht passt, wohin muss man als Nächstes?*

Lösung: Wir verfolgen alle Möglichkeiten gleichzeitig:

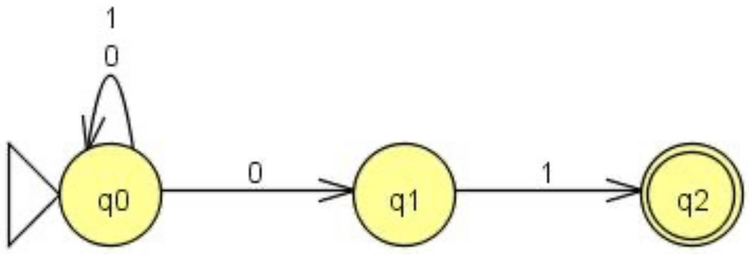


Nichtdeterminismus = verfolge alle Möglichkeiten simultan

Definition eines Nicht-deterministischen Automaten (NEA)

Ein Nichtdeterministischer Endlicher Automat (NEA) ist ein 5-Tupel

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



mit

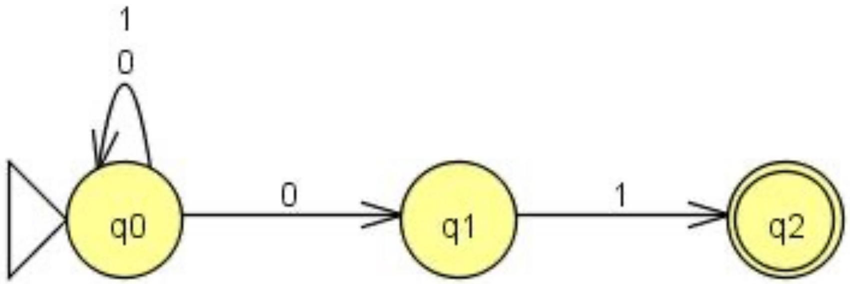
- Q: nichtleere endliche Zustandsmenge
- ∑: endliches Eingabealphabet
- δ: $Q \times \Sigma \rightarrow P(Q)$ Zustandsüberführungsfunktion (Übergangsfunktion)
- $q_0 \in Q$: Startzustand (Anfangszustand)
- $F \subseteq Q$: Menge von akzeptierenden Zuständen (Endzuständen)

Hierbei ist $P(Q) = \{S \mid S \subseteq Q\}$ die Potenzmenge von Q.

Arbeitsweise eines NEA

Beispiel:

Alle Wörter aus {0,1}*, die mit 01 enden.



Abarbeitung von 00101 :

$(q_0,00101) \vdash (q_0,0101) \vdash (q_0,101) \vdash (q_0,01) \vdash (q_0,1) \vdash (q_0,\varepsilon)$

$(q_0,00101) \vdash (q_1,0101)$ Sackgasse

$(q_0,0101) \vdash (q_1,101)$

$(q_1,101) \vdash (q_2,01)$ Sackgasse

$(q_0,01) \vdash (q_1,1)$

$(q_1,1) \vdash$ (q_2,ε) √

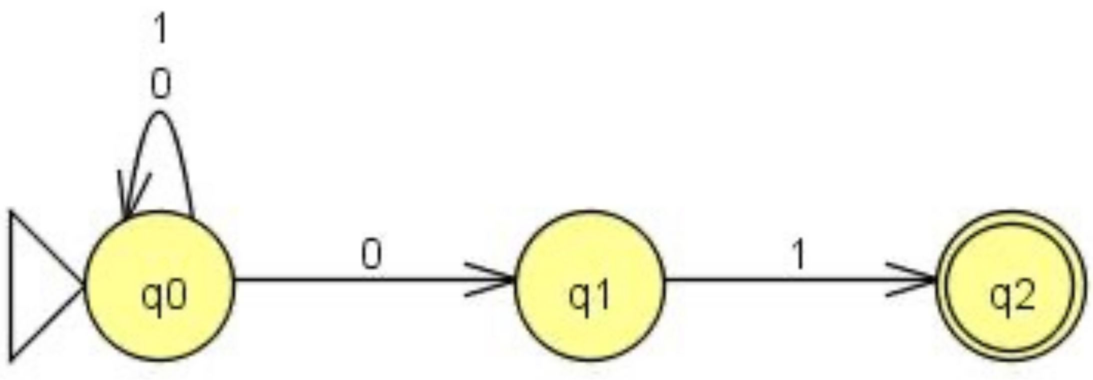
Möglichkeiten:

1. Verzweigung
2. Sackgasse
3. Erfolgreiche Abarbeitung, aber zum Schluss nicht im Endzustand
4. Erfolgreiche Abarbeitung, aber zum Schluss im Endzustand

Ein NEA akzeptiert ein Wort,
wenn mindestens ein Abarbeitungsweg zu einem Endzustand führt !

NEA als Diagramm oder Tabelle

Übergangsdiagramm:



Übergangstabelle:

NEA	0	1
$\rightarrow q_0$	$\{q_0,q_1\}$	$\{q_0\}$
q_1	$\{\}$	$\{q_2\}$
$*q_2$	$\{\}$	$\{\}$

Äquivalenz rechtslineare Grammatik (Typ 3) und NEA

Rechtslineare Grammatik
G = (V, T, P, S)

NEA
A = (V, T, δ, S, F)

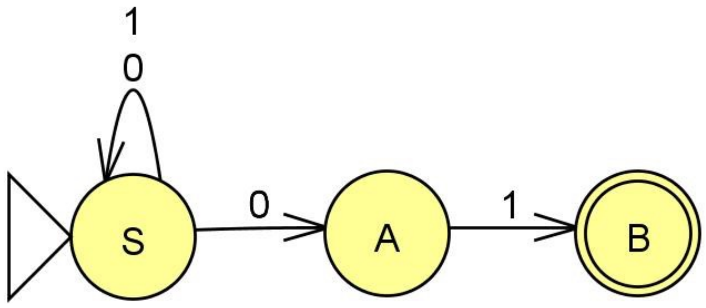
G = ({S,A,B}, {0,1}, P, S) mit

P = { S→0S|0A| 1S,
A→ 1B,
B→ε }

ε-Regeln werden zu Endzuständen und umgekehrt!

A = ({S,A,B}, {0,1}, δ, S, {B}) mit

δ	0	1
→S	{S,A}	{S}
A	{}	{B}
*B	{}	{}



Man kann jede rechtslineare Grammatik (Typ 3)
in einen äquivalenten NEA und umgekehrt umwandeln !

Sprache eines NEA

Die erweiterte Übergangsfunktion $\delta^*(q_0,w)$ ist die Menge aller Zustände, die man von q_0 aus erreicht, wenn man das Wort w abarbeitet.

$(q_0,00101) \vdash (q_0,0101) \vdash (q_0,101) \vdash (q_0,01) \vdash (q_0,1) \vdash (q_0,\varepsilon)$

$(q_0,00101) \vdash (q_1,0101)$ Sackgasse

$(q_0,0101) \vdash (q_1,101)$

$(q_1,101) \vdash (q_2,01)$ Sackgasse

$(q_0,01) \vdash (q_1,1)$

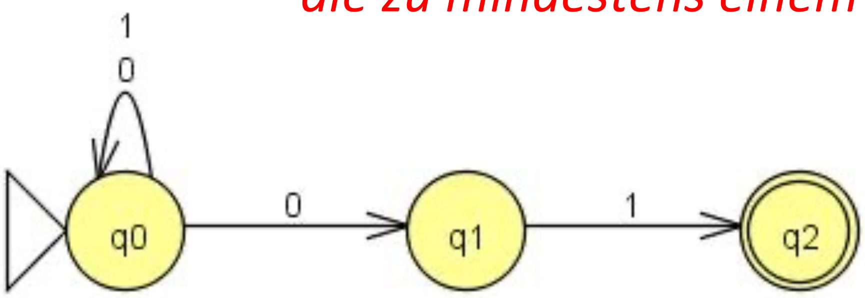
$(q_1,1) \vdash (q_2,\varepsilon)$ √

$\delta^*(q_0,00101) = \{q_0,q_2\}$

Sprache eines NEA $(Q, \Sigma, \delta, q_0, F)$: $L(A) = \{w \in \Sigma^* \mid \delta^*(q_0,w) \cap F \neq \emptyset\}$

L(A) ist die Menge aller Wörter,

die zu mindestens einem akzeptierenden Zustand führen.



$L(A) = \{w \in \{0,1\}^* \mid w \text{ endet mit } 01\}$

$= \{w \in \{0,1\}^* \mid \exists\, u \in \{0,1\}^*, w=u01\}$

Unterschiede zwischen DEA und NEA

Bei einem DEA ist man immer in einem Zustand, bei einem NEA immer gleichzeitig in mehreren.

$(q_0,00101)$	$\vdash$	$(q_0,0101)$	$\vdash$	$(q_0,101)$	$\vdash$	$(q_0,01)$	$\vdash$	$(q_0,1)$	$\vdash$	(q_0,ε)
	$\vdash$	$(q_1,0101)$	$\vdash$	$(q_1,101)$	$\vdash$	$(q_2,01)$	$\vdash$	$(q_1,1)$	$\vdash$	(q_2,ε) √
		Sackgasse				Sackgasse				

Genauer: Bei einem NEA ist immer einer Teilmenge der Zustände aktiv.

Genau das verwendet man, um einen NEA in einen DEA umzuwandeln:

Teilmengen von NEA-Zuständen werden als DEA Zustände codiert.

Aus n Zuständen beim NEA werden

- in der Theorie maximal 2^n Zustände beim DEA.
- in der Praxis etwa genau so viele Zustände beim DEA.

Umwandlung eines NEA in einen DEA:Teil(Potenz)mengenkonstruktion

Sei

$$A_N = (Q_N, \Sigma, \delta_N, q_N, F_N)$$

ein nichtdeterministischer Automat (NEA).

Konstruiere äquivalenten deterministischen Automat (DEA)

$$A_D = (Q_D, \Sigma, \delta_D, q_D, F_D)$$

mit

- $Q_D = P(Q_N)$ Potenzmenge (Menge aller Teilmengen)
- $q_D = \{q_N\}$ Anfangszustand ist einelementige Menge
- $F_D = \{S \in Q_D \mid S \cap F_N \neq \emptyset\}$ Alle Teilmengen mit einem Endzustand
- $\delta_D(S, a) = \bigcup_{q \in S} \delta_N(q,a)$ Vereinigung aller Möglichkeiten

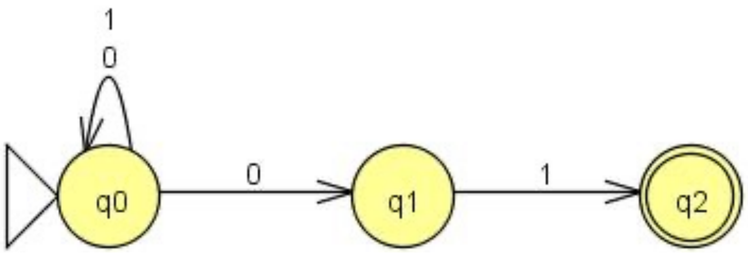
Es gilt $L(A_D) = L(A_N)$!

Konstruktion benötigt $2^{|Q_N|}$ Zustände (Optimierung möglich)

Beispiel für Teilmengenkonstruktion

NEA $(Q_N, \{0,1\}, \delta_N, q_N, F_N)$

Bsp.: $(\{q0,q1,q2\}, \{0,1\}, \delta, q_0, \{q_2\})$



NEA δ_N	0	1
$\rightarrow q_0$	$\{q_0,q_1\}$	$\{q_0\}$
q_1	$\{\}$	$\{q_2\}$
$*q_2$	$\{\}$	$\{\}$

DEA: Konstruierter deterministischer Automat $(Q_D, \{0,1\}, \delta_D, q_D, F_D)$ mit

$Q_D = P(Q_N) = P(\{q_0,q_1,q_2\})$
$= \{\{\},\{q_0\},\{q_1\},\{q_2\},$
$\{q_0,q_1\},\{q_0,q_2\},\{q_1,q_2\},\{q_0,q_1,q_2\}\}$

$q_D = \{q_N\} \quad = \{q_0\}$

$F_D = \{S \in Q_D \mid S \cap F_N \neq \emptyset\}$
$= \{\{q_2\},\{q_0,q_2\},\{q_1,q_2\},\{q_0,q_1,q_2\}\}$

DEA δ_D	0	1
$\{\}$	$\{\}$	$\{\}$
$\rightarrow\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_1\}$	$\{\}$	$\{q_2\}$
$*\{q_2\}$	$\{\}$	$\{\}$
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$
$*\{q_1,q_2\}$	$\{\}$	$\{q_2\}$
$*\{q_0,q_1,q_2\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$

Umbenennung einer Teilmengenkonstruktion

Diese Übergangstabelle gehört zu einem DEA, d.h. die Einträge in der Tabelle sind einzelne Zustände und nicht Mengen von Zuständen!

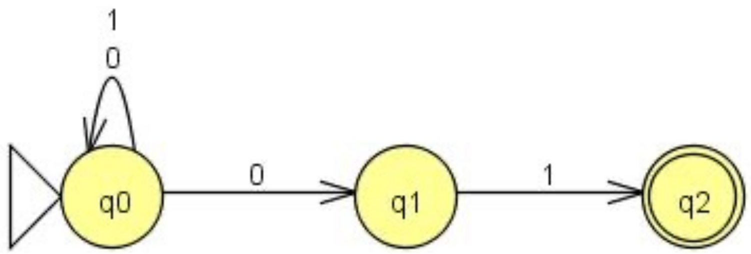
Dies wird durch eine Umbenennung deutlich.

DEA	0	1
$\{\}$	$\{\}$	$\{\}$
$\rightarrow\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_1\}$	$\{\}$	$\{q_2\}$
$*\{q_2\}$	$\{\}$	$\{\}$
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$
$*\{q_1,q_2\}$	$\{\}$	$\{q_2\}$
$*\{q_0,q_1,q_2\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$

DEA	0	1
A	A	A
→B	E	B
C	A	D
*D	A	A
E	E	F
*F	E	B
*G	A	D
*H	E	F

Benutzen Sie in dieser Veranstaltung bitte die originale Benennung (links).

Optimierte Teilmengenkonstruktion: nicht erreichbare Zustände



DEA	0	1
{}	{}	{}
→$\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_1\}$	{}	$\{q_2\}$
*$\{q_2\}$	{}	{}
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
*$\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$
*$\{q_1,q_2\}$	{}	$\{q_2\}$
*$\{q_0,q_1,q_2\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$

Viele überflüssige Zustände:

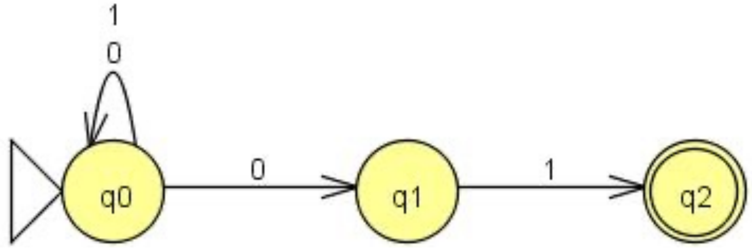
nur 3 Zustände sind von $\{q_0\}$ erreichbar!

Bessere Idee:

1. Fang mit dem Startzustand $\{q_0\}$ an und entwickle diese Zeile.
2. Falls dabei neue Zustände auftauchen
   1. schreibe sie auch in die 1. Spalte
   2. entwickle sie auch.
3. Ende, wenn keine neuen Zustände mehr auftauchen.

Optimierte Teilmengenkonstruktion

Fachhochschule Dortmund
University of Applied Sciences and Arts
we focus on students



Optimierung:

Q_D = erreichbare Mengenzustände im NEA
= erreichbare Zustände im DEA

NEA	0	1
$\rightarrow q_0$	$\{q_0,q_1\}$	$\{q_0\}$
q_1	$\{\}$	$\{q_2\}$
$*q_2$	$\{\}$	$\{\}$

Konstruiere Q_D gleichzeitig mit δ_D:

Algorithmus

- Start:
  $Q_0 := \{ \{q_0\} \}$
- Schritt:
  $Q_{i+1} := Q_i \cup \{\delta_D(S, a) \mid S \in Q_i, a \in \Sigma \}$
- Ende wenn
  $Q_{i+1} = Q_i$ (Setze dann $Q_D := Q_i$)

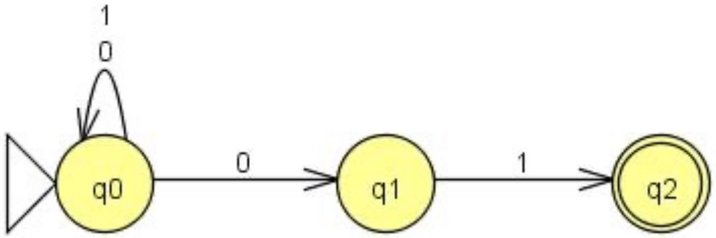
DEA	0	1
$\rightarrow\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$

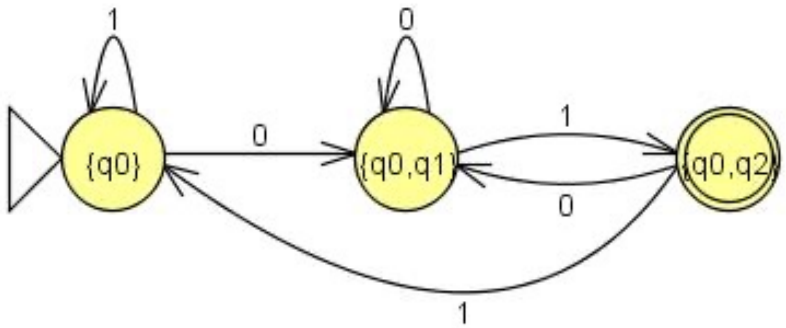
Der DEA enthält dann keine überflüssigen Zustände

Abarbeitungen im NEA und DEA

NEA	0	1
$\rightarrow q_0$	$\{q_0,q_1\}$	$\{q_0\}$
q_1	$\{\}$	$\{q_2\}$
$*q_2$	$\{\}$	$\{\}$

DEA	0	1
$\rightarrow\{q_0\}$	$\{q_0,q_1\}$	$\{q_0\}$
$\{q_0,q_1\}$	$\{q_0,q_1\}$	$\{q_0,q_2\}$
$*\{q_0,q_2\}$	$\{q_0,q_1\}$	$\{q_0\}$





Abarbeitung von 00101
im NEA:

$(q_0,00101) \vdash (q_0,0101) \vdash (q_0,101) \vdash (q_0,01) \vdash (q_0,1) \vdash (q_0,\varepsilon)$

$(q_0,00101) \vdash (q_1,0101)$

$(q_0,0101) \vdash (q_1,101) \vdash (q_2,01)$

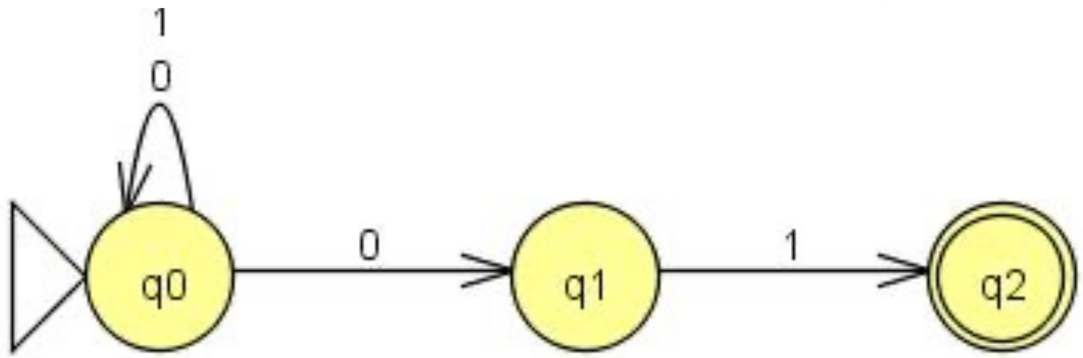
$(q_0,01) \vdash (q_1,1) \vdash (q_2,\varepsilon)$ √

im DEA:

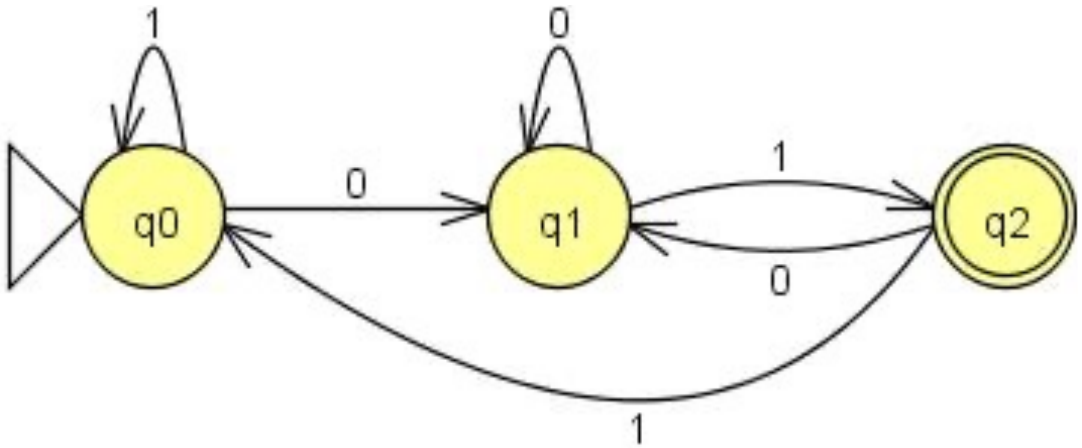
$(\{q_0\},00101) \vdash (\{q_0,q_1\},0101) \vdash (\{q_0,q_1\},101) \vdash (\{q_0,q_2\},01) \vdash (\{q_0,q_1\},1) \vdash (\{q_0,q_2\},\varepsilon)$ √

Vergleich der Umwandlung

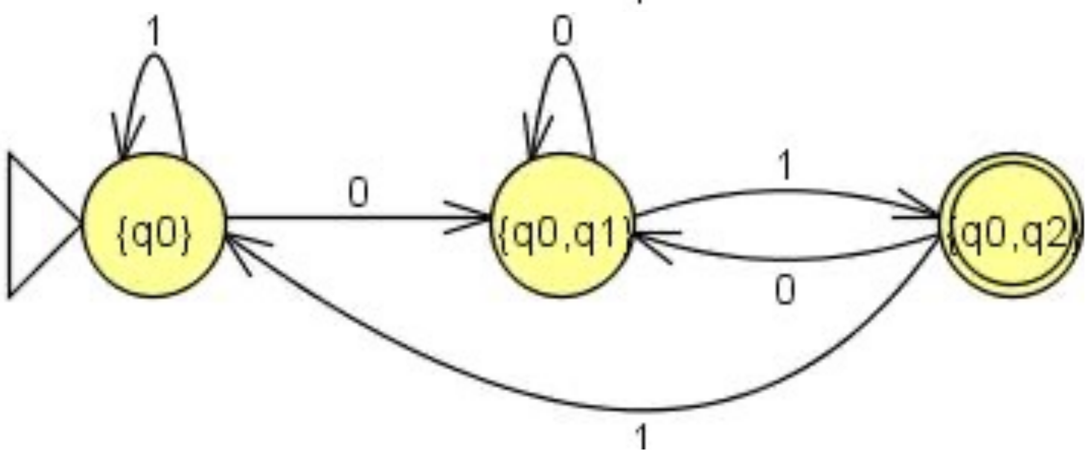
NEA selbst gemacht:



DEA selbst gemacht:



DEA durch
Teilmengenkonstruktion:



Sprachen von DEAs und NEAs

Satz: Wenn ein NEA A_N durch die Teilmengenkonstruktion in einen DEA A_D überführt wird, dann gilt

$$L(A_N) = L(A_D) .$$

Beobachtungen:

1. Aus einem NEA können wir immer einen DEA bauen, der dieselbe Sprache akzeptiert (Teilmengenkonstruktion).
2. Aus einem DEA können wir immer einen NEA bauen, der dieselbe Sprache akzeptiert (ein DEA ist praktisch auch ein NEA).

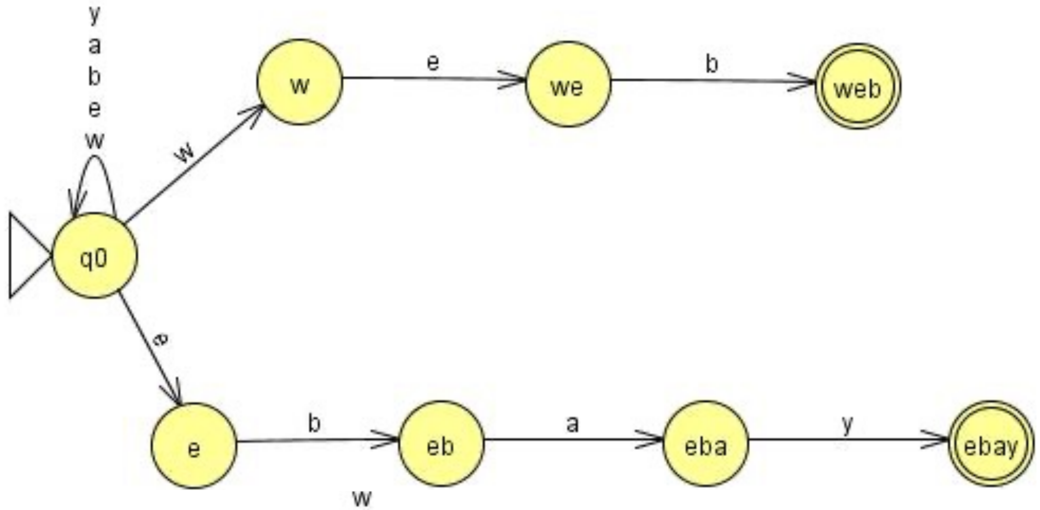
Schlussfolgerung:

Die von DEAs und NEAs akzeptierten Sprachen sind identisch !

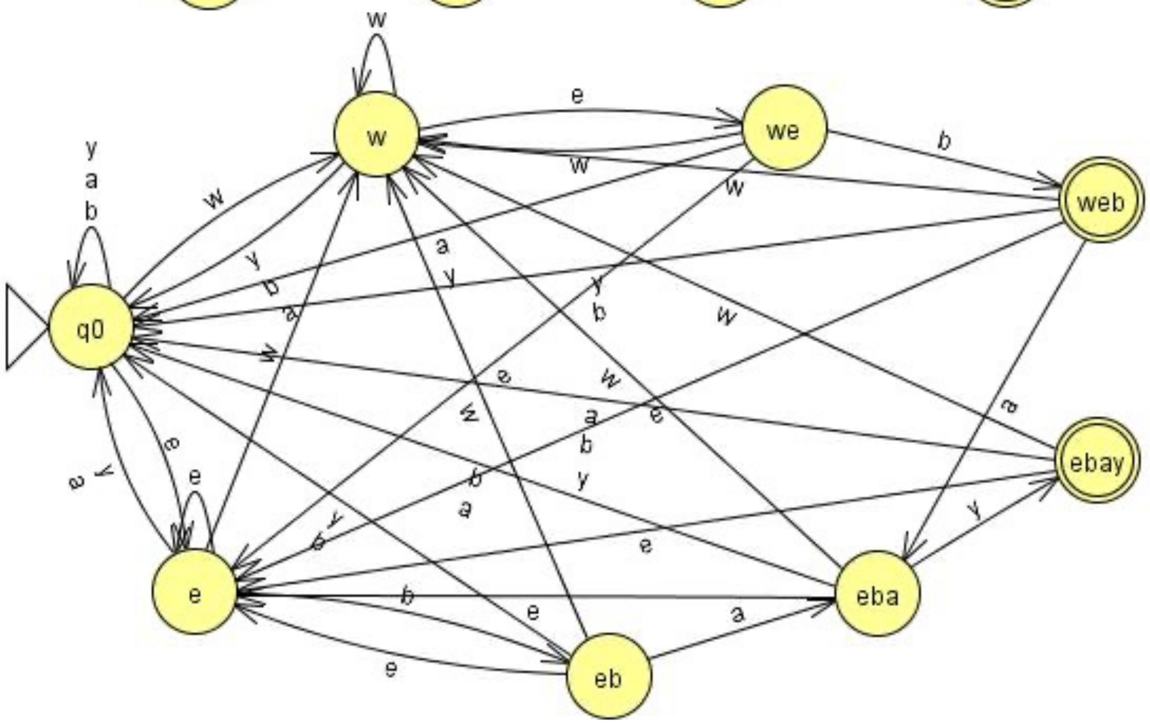
…dies sind die regulären Sprachen (Typ 3)!

Deterministische Automaten für die Textanalyse

Typischer NEA
für die Textanalyse:



Teilmengenkonstruktion:



Geht das noch schlimmer?

Zusammenfassung

- Ein NEA ist wie ein DEA, aber mit mengenwertiger Überführungsfunktion, d.h. keine oder mehrere Folgezustände sind möglich.
- Man kann jede rechtslineare Grammatik in einen äquivalenten NEA und umgekehrt umwandeln.
- Bei der Ableitung eines Wortes in einem NEA kann es zu Sackgassen und zu Abzweigungen kommen.
- Ein NEA akzeptiert ein Wort genau dann, wenn es von dem Startzustand in mindestens einen Endzustand abgeleitet werden kann.
- Ein NEA ist einfacher zu konstruieren als ein DEA, aber schwieriger zu simulieren.
- Jeder NEA kann durch Teilmengenkonstruktion in einen DEA transformiert werden, der die selbe Sprache akzeptiert. Im schlimmsten Fall hat der neue DEA 2^n viele Zustände. Im Normalfall hat der DEA ähnlich viele Zustände, nur mehr Übergänge.