Vom NEA zum DEA

Einführung

Nichtdeterministische Automaten (NEA) besitzen für die Modellierung von Sprachen eine wichtige Bedeutung. Mit diesen Automaten können Sprachen erkannt oder definiert werden. Alle Wörter, die von NEA erkannt werden bilden die Menge der Wörter, die die Sprache des Automaten definiert.
Die Frage, die sich auftut, ist die, ob die NEA mehr können als die DEA. Es zeigt sich aber, dass beide Automatentypen äquivialent sind. Zu jedem NEA kann mithilfe des **Potenzmengenalgorithmus** ein äquivalenter DEA konstruiert werden. .....

Wir betrachten folgenden Automaten:

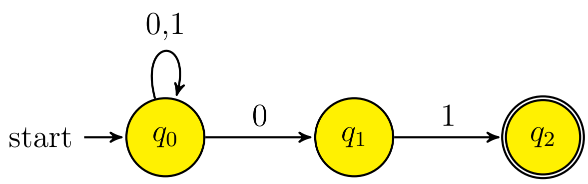

Abbildung 1: Ein NEA mit 3 Zuständen

Dieser Nichtdeterministische Automat akzeptiert alle Zeichenreihen, die auf 01 enden. Somit gilt $\mathcal{L}(\mathcal{A}) = \{w \mid (0|1)^*01\}$. Um die Arbeitsweise dieses NEA zu verdeutlichen wird in einer Grafik die Darstellung aller möglichen Folgezustände q_i nach folgendem Schema dargestellt:

Aufgabe: Verarbeite die Eingabe $w = 00101$!

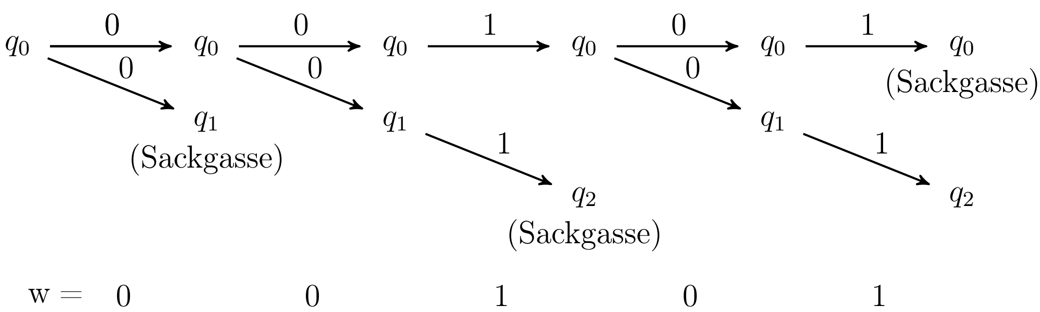

Abbildung 2: Darstellung der Ableitung eines Wortes w

Man kann nun erkennen, dass nicht alle Pfade (endliche Folge aufeinanderfolgender Kanten) in einer Sackgasse enden. Es gibt eine Folge von Zuständen q_i, beginnend in q_0, die im Endzustand q_2 endet. Dann wird dieses Wort akzeptiert.

Definition 1: Nichtdeterministischer Endlicher Automat (NEA)

Ein NEA ist ein 5-Tupel (Q, Σ, δ, q_0, F) mit

- Q ist eine endliche Menge von Zuständen
- Σ endliche Menge von Eingabesymbolen
- $\delta : Q \times \Sigma \to \mathcal{P}(Q)$ die Übergangsfunktion mit $\delta(q_i, a) = \{q_i\}$
- q_0 - der Startzustand
- die Menge der Endzustände $F \in Q$

Das δ (auch δ_N bezeichnet) eines NEA stellt also ein Element der Übergangsrelation $Q \times \Sigma \times Q$ dar. Das bedeutet also, der Nachfolgezustand q_i ist nicht eindeutig definiert. Diese Eindeutigkeit ist beim DEA gegeben, was dann δ_D zu einer Überführungsfunktion $Q \times \Sigma \to Q$ macht.

Für unseren obigen Automaten heißt das N $= (\{q_0, q_1, q_2\}, \{0, 1\}, \delta, q_0, \{q_2\})$.

Eine weitere Möglichkeit der Beschreibung eines Automaten sind **Übergangstabellen**. Für unser Beispiel sieht die Tabelle wie folgt aus:

	0	1
$\to q_0$	$\{q_0, q_1\}$	$\{q_0\}$
q_1	$\varnothing$	$\{q_2\}$
$*q_2$	$\varnothing$	$\varnothing$

Für die Festlegung der **Sprache** $\mathcal{L}(\mathcal{A})$ zeigt sich, dass die Erweiterung der Überführungsfunktion δ zur erweiterten Überführungsfunktion $\hat{\delta}$ sinnvoll ist.
Wir stellen in Wort $w \in \Sigma^*$ in der Form $w = xa$ dar, mit $a \in \Sigma, x \in \Sigma^*$

Sei $\hat{\delta}(q_i, \epsilon) = \{q_i\}$ mit $q_i \in Q$ beliebiger Zustand.

Dann ist $\hat{\delta}(q_0, w) = \{r_1, \cdots, r_m\}$ mit $\hat{\delta}(q_0, x) = \{p_1, \cdots, p_k\}$ und $\bigcup_{i=1}^{k} \delta(p_i, a) = \{r_1, \cdots, r_m\}$

Also muss, entsprechend der Abbildung 2 auf Seite 1, mindestens ein Endzustand in der Ergebnismenge von $\hat{\delta}$ vorhanden sein, um w zu akzeptieren.

Definition 2: Die Sprache $\mathcal{L}$ eines NEA

Sei F die Menge der Endzustände und $w \in \Sigma^*$ ein beliebiges Wort aus Σ. Dann ist die Sprache eines NEA definiert: $\mathcal{L}(\mathcal{A}) = \{w \mid \hat{\delta}(q_0, w) \cap F \neq \varnothing\}$

Übung: Beschreibe $\hat{\delta}$ für das Beispiel von oben!
Lösung:

Wir betrachten nun die Umwandlung eines NEA in einen äquivalenten DEA. Dazu betrachtet man zuerst die Übergangstabelle des NEA und entwickelt dazu die Transformationstabelle die angibt, welche Ergbnismengen δ_N liefert. Diese werden dann als neue Zustände des DEA betrachtet. Wir behandeln die Umwandlung am Beispiel des NEA in Abbildung 1 auf Seite 1.

Voraussetzung: $\mathrm{N} = (Q_N, \Sigma, \delta_N, q_0, F_N) \to D = (Q_D, \Sigma, \delta_D, q_0, F_D)$ mit $\mathcal{L}(N) = \mathcal{L}(D)$
Idee:

- Q_D ist die Menge der Teilmengen von Q_N (Potenzmenge)
- F_D ist die Menge S der Teilmengen von Q_N mit $S \cap F_N \neq \varnothing$
- $\forall S \subseteq Q_N$ und $\forall a \in \Sigma : \delta_D(S, a) = \bigcup_{p \in S} \delta_N(p, a)$

	0	1
$\to q_0$	$\{q_0, q_1\}$	$\{q_0, \}$
q_1	$\varnothing$	$\{q_2\}$
$*q_2$	$\varnothing$	$\varnothing$

Übergangstabelle

	0	1
$\to \{q_0\}$	$\{q_0, q_1\}$	$\{q_0\}$
$\{q_0, q_1\}$	$\{q_0, q_1\}$	$\{q_0, q_2\}$
$*\{q_0, q_2\}$	$\{q_0, q_1\}$	$\{q_0\}$

Transformationstabelle

Man sieht, dass nach jedem Übergang des NEA alle neu entstandenen Ergebnismengen als neue Zustände in der Transformationstabelle aufgefasst werden und entsprechend der Definiton von δ_N für jedes Element und jedem Zeichen a die neuen Ergebnismengen ermittelt werden. Tauchen diese noch nicht in der linken Spalte als neue Zustände des DEA

auf, dann werden diese Ergänzt. Dies macht man, bis alle Ergebnismengen des NEA ermittelt wurden.
Das Verfahren terminiert, weil es bei $|Q_N| =$ n nur 2^n Teilmengen gibt.

Somit erhalten wir folgenden DEA:

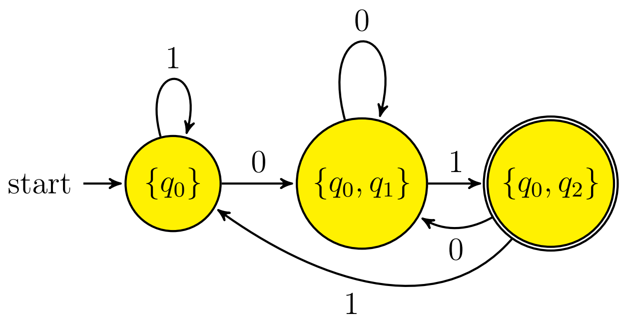

Abbildung 3: Dem NEA zugeprdneter DEA mit 3 Zuständen

Nach entsprechender Umbenennung der Zustände liegt ein DEA mit gewohnter Bezeichung vor.

Diese Transformation ist nicht vollständig, sondern *lazy*. Zu jedem NEA kann eine vollständige Transformationstabelle mit allen möglichen 2^n Übergängen konstruiert werden. Abschließend kann folgender Satz formuliert werden:

Satz 1: Äquivalenz von DEA und zugehörigem NEA

Für jeden NEA gibt es einen äquivalenten DEA, der dieselbe Sprache akzeptiert.

Übung 1: Gegeben ist ein regulärer Ausdruck $(0+1)^*1(0+1)$ (und damit ein NEA). Gib die übergangstabelle und einen (den?) äquivalenten DEA an.