Theoretische Grundlagen der Informatik

Release 1.0

Daniel Schmidt

Mar 27, 2018

CONTENTS

CHAPTER

ONE

VORLESUNGSVERZEICHNIS:

1.1 Vorlesungen

1.1.1 1. Vorlesung

Bedingungen zur Klausurzulassung

- 10 Blätter erfolgreich bearbeiten
- 40% pro Blatt richtig bearbeitet
- Anwesenheitspflicht in den Übungen
- 1 Mal Vorrechnen, jeder kann dran kommen
- Abgaben in Zweiergruppen

Organisation

- Skript wird wochenweise online gestellt mit Passwort: "TGI2013"
- Am besten mitschreiben in der Vorlesung

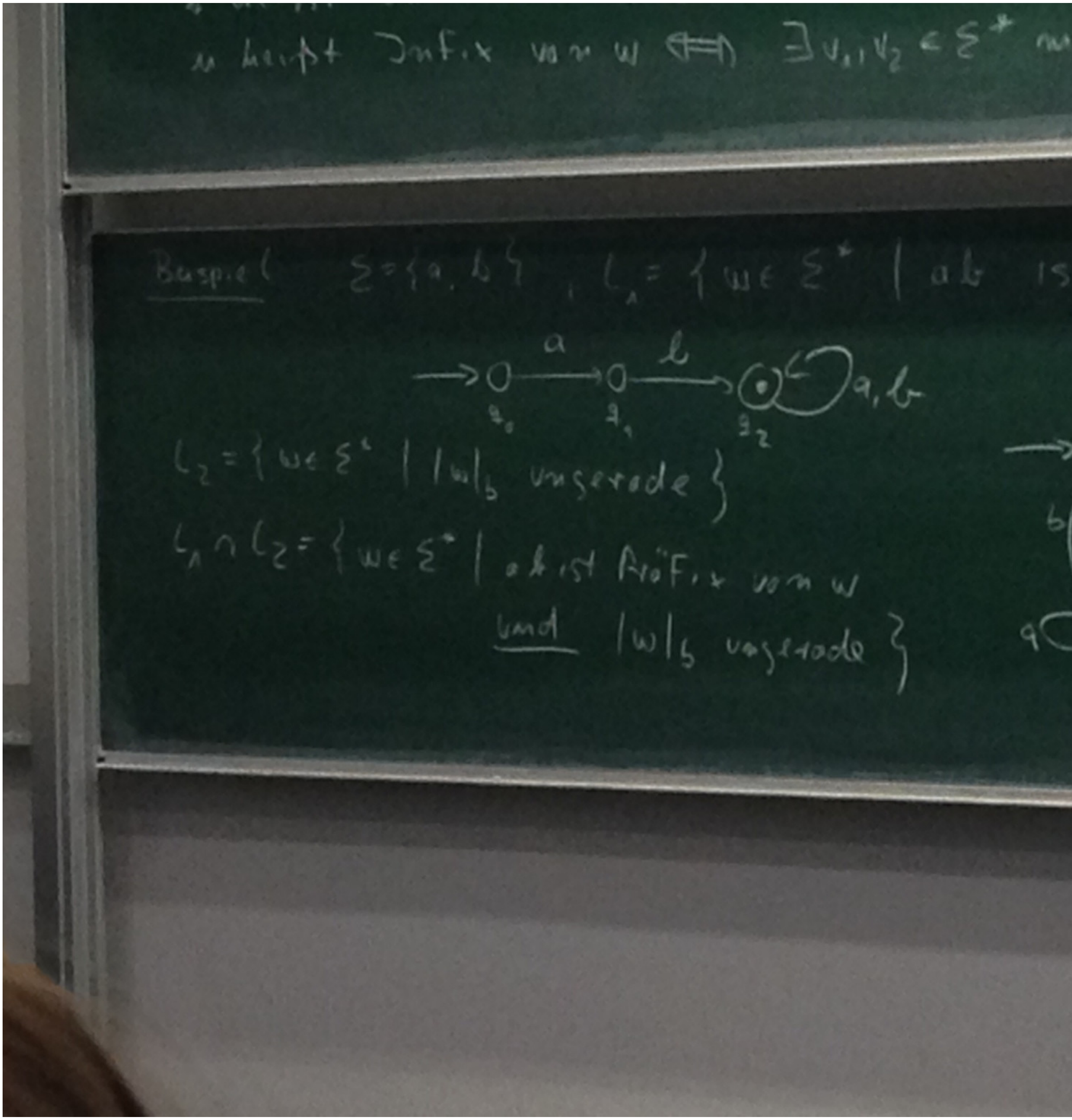
Einführende Definitionen

Es wurden die folgenden Definitionen eingeführt

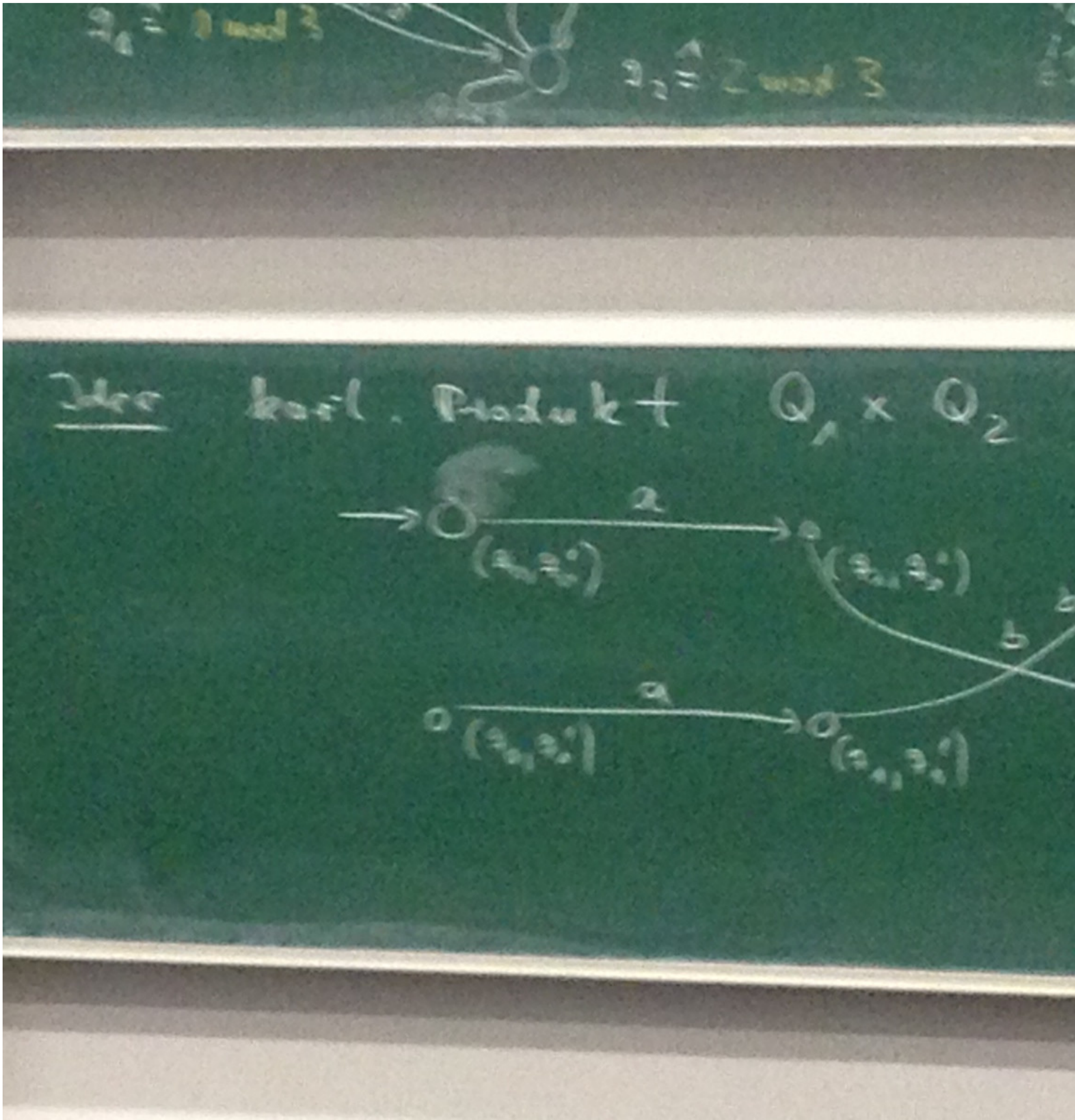
- *Bezeichnungen*
- *Transitionssysteme, endl. Automaten*

Durchschnittt zweier Sprachen

Beispielsprachen:



Idee: Kartesisches Produkt $Q_1 X Q_2$



1.1.2 2. Vorlesung

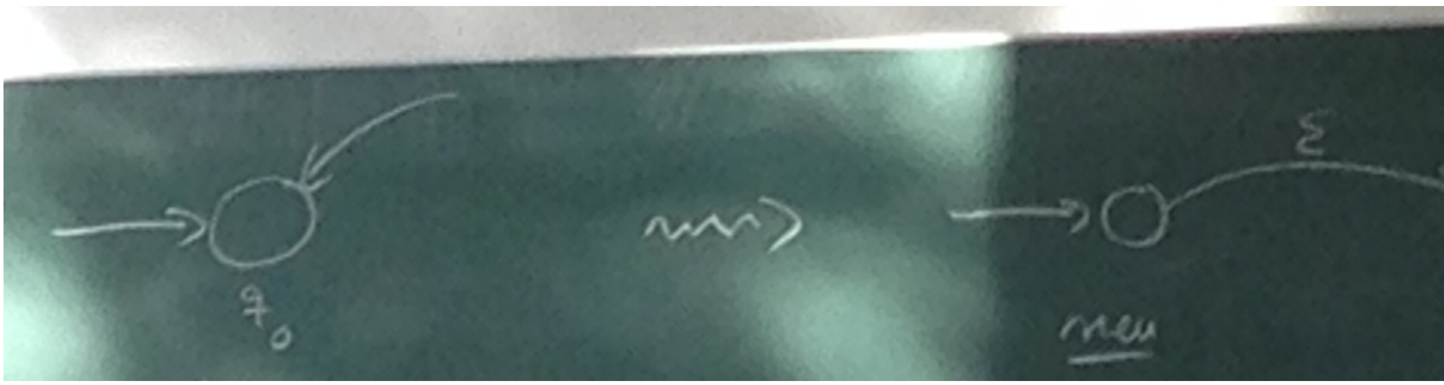
Es wurden die folgenden Definitionen eingeführt

- *Produktautomat*

Lemma: $L(a_1 X a_2) = L(a_1) \cap L(a_2)$

$$
\begin{aligned}
& w = a_1 ... a_n \in L(a_1 X a_2) \\
& \Longleftrightarrow^{\text{Definition Akzeptierbarkeit}} \exists \text{ Pfad } (p_0, r_0) a_1 (p_1, r_1) a_2 ...... \text{ mit} \\
& (p_0, r_0) = (q_1, q_2), ((p_i, r_i), a_{i+1}, (p_{i+1}, r_{i+1})) \in \Delta \text{ und } (p_n, r_n) \in F \\
& \Longleftrightarrow^{\text{Def } \Delta, F} \exists \text{ Pfad } p_0 a_1 p_1 ... a_n p_n \text{ mit } p_0 = q_1, (p_i, q_{i+1}, p_{i+1}) \in \Delta_1, p_n \in F_1 \\
& \text{und } \exists \text{ Pfad } p_0 a_1 p_1 ... a_n p_n \text{ mit } r_0 = q_2, (p_i, q_{i+1}, p_{i+1}) \in \Delta_2, p_n \in F_2 \\
& \Longleftrightarrow^{\text{Definition Akzeptierbarkeit}} w \in L(a_1) \text{ und } w \in L(a_2) \\
& \Longleftrightarrow w \in L(a_1) \cap L(a_2)
\end{aligned}
$$

Satz: Zu jedem NEA a mit Worttransitionen gibt es einen äquivalenten NEA a'.



Lemma 1: Zu jedem NEA a mit Worttransitionen gibt es einen ε-NEA a' mit L(a) = L(a').

Beweis:

Ersetze jede Transition $(p, a_1, ..., a_n, q)$ in a mit $n \geq 2$ durch Transitionen $(p, a_1, q_1), (p, a_2, q_2), ..., (p, a_n, q_n)$ mit jeweils neuen Zuständen $q_1, ..., q_{n-1}$.

Lemma 2: Zu jedem ε-NEA a gibt es einen äquivalenten NEA a'

Beweis:

Sei $a = (Q, \Sigma, q_0, \Delta, F)$ ein ε-NEA. O.B.d.A gibt es keine Transitionen nach q_0. Wir definieren $a' = (Q, \Sigma, q_0, \Delta', F')$ mit

$$
\begin{aligned}
& (p, a, q) \in \Delta' \Longleftrightarrow \exists \text{ ein Pfad } \Pi \text{ durch a von p nach q mit Beschriftung } p(\Pi) = a \\
& F' = F \cup \{q_0\} \text{ falls } a : q_0 \rightarrow^{\varepsilon} F, \text{ ansonsten F}
\end{aligned}
$$

Behauptung: L(a) = L(a')

"<": a akzeptiert w. Sei $p_0 a_1 p_1 ... a_n p_n$ Pfad von q_0 durch ε-NEA a nach F mit Beschriftung w. Seien $a_{i_1}, ..., a_{i_n}$ die $a_i \neq \varepsilon$ (mit $i_1 < i_2 < ... < i_n$) $\Longrightarrow w = a_{i_1}, ..., a_{i_n}$

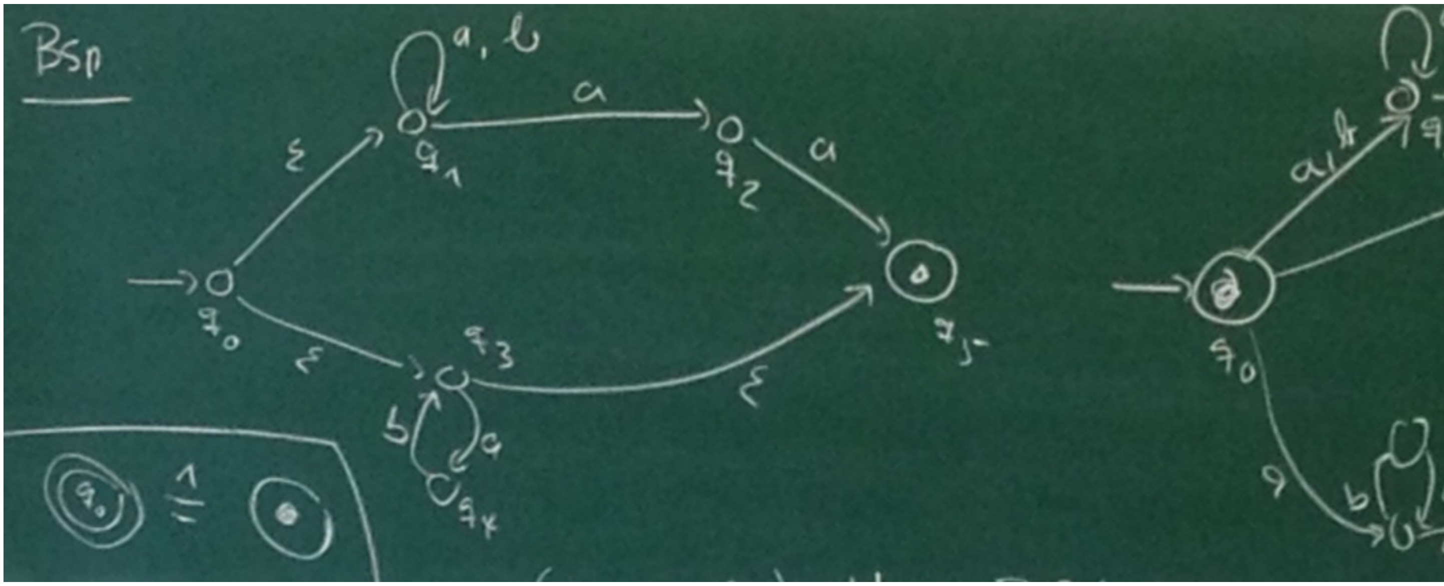
Fall 1: $w = 0 \Longrightarrow w = \varepsilon \Longrightarrow a : q_0 \rightarrow^{\varepsilon} F \Longrightarrow q_0 \in F'$ und a' akzeptiert ε.

Fall 2: $w > 0$ Dann ist die folgende Folge ein Pfad durch a': $p_0 a_{i_1} p_{i_1} a_{i_2} p_{i_2}, ..., p_{i_{n-1}} a_{i_n} p_{i_n}$ Dazu zeige

$$(p_0, a_{i_1}, p_{i_1}) \in \Delta'$$
$$(p_{i_1}, a_{i_2}, p_{i_2}) \in \Delta'$$
$$...$$
$$(p_{i_{n-1}}, a_{i_n}, p_{i_n}) \in \Delta'$$

$\Longrightarrow w \in L(a')$

">" ist analog zu zeigen.



$$(q_0, a, q_1) \in \Delta' \Longleftrightarrow \exists \text{ Pfad } \Pi = q_0 \varepsilon q_1 a q_1, \beta(\Pi) = a$$

Verfahren zur Entscheidung ob ε-NEA $a : p \rightarrow q$ mit Pfad Π

Hierzu reicht ein Verfahren zur Entscheidung, ob $a : p \rightarrow^{\varepsilon} p'$ bzw $a : q \rightarrow^{\varepsilon} q'$ für $p', q' \in Q$ durch ε - Pfade.

Gegeben: ε-NEA $a = (Q, \Sigma, q_0, \Delta, F)$ mit $Q = \{q_0, q_1, ..., q_{n-1}\}$

Definiere $\varepsilon_{ij} = 1 falls (q_i, \varepsilon, q_j) \in \Delta$ oder i = j, ansonsten 0

Gesucht: Werte c_{ij} mit $c_{ij} = 1 falls a : q_i \rightarrow^{\varepsilon} q_j$, ansonsten 0

Dazu berechne rekursiv

$$c_{ij}^{(r)} = 1 falls \exists \text{ Pfad von } q_i \text{ nach } q_j \text{ der Länge } \leq r$$

Behauptung: $c_{ij} = c_{ij}^{(n-1)}$ für $n = \mid Q \mid$

Zeige: Es existiert ein ε - Pfad von q_i nach q_j $\not\Leftrightarrow$ Es existiert ein ε - Pfad von q_i nach q_j der Länge $\leq n-1$

<= ist trivial

=>: Es sei Π ein ε - Pfad von q_i nach q_j minimaler Länge Zeige: Länge von Π ist $\leq n-1$

Annahme: $\Pi = q_i = p_0 \varsigma p_1 \varsigma ... \varsigma p_m = q_j \wedge m \geq n$

=> Es gibt eine Zustandswiederholung $p_k = p_{k'}$ mit $k \subset k'$

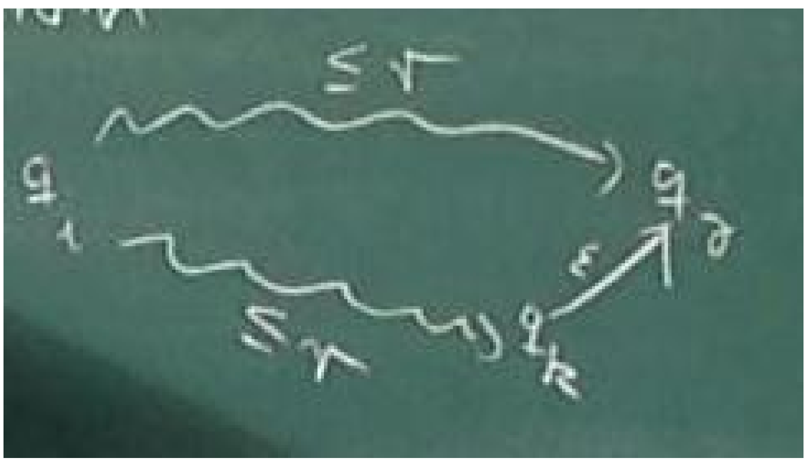
=> Es gibt einen kürzeren ε - Pfad von q_i nach q_j, was ein Wiederspruch zur Minimalität ist.

Wie berechnet man $c_{ij}^{(r)}$?

$$c_{ij}^{(0)} = 1 \text{ falls } i = j, \text{ ansonsten } 0$$

$$c_{ij}^{(r+1)} = c_{ij}^{(r)} \vee \bigvee_{k=0}^{n-1} (c_{ik}^{(r)} \wedge \varsigma_{kj})$$

Das heißt es gibt einen Pfad von q_i nach q_j der Länge $\leq r+1$, wenn es einen Pfad der Länge $\leq r$ gibt **oder** es gibt einen Zwischenknoten q_k und einen Pfad der Länge $\leq r$ von q_i nach q_j und eine ε - Transition $(q_k, \varepsilon, q_j) \in \Delta$



Zeitaufwand

O(n^4), reduzierbar auf 0(n^3)

1.1.3 3. Vorlesung

Es wurden die folgenden Definitionen eingeführt

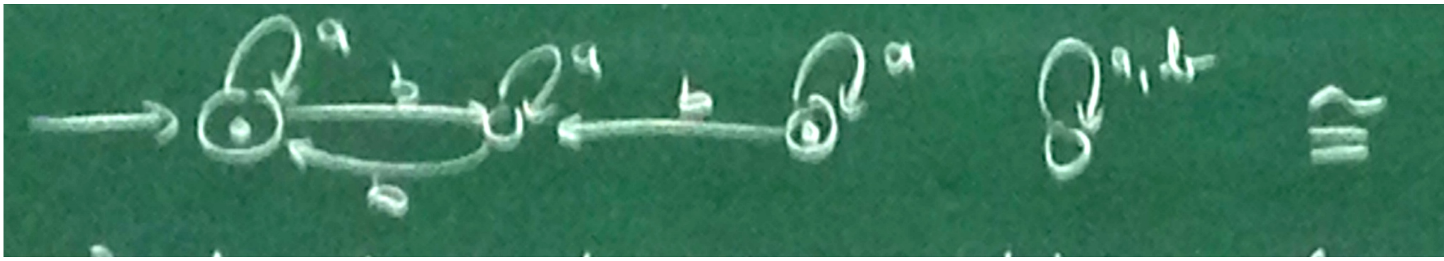
- *Deterministisch endlicher Automat (DEA)*
- *Fortsetzung*

Satz: Potenzmengenkonstruktion

Zu jedem NEA kann man einen äquivalenten DEA konstruieren.

Idee:

Neue Folge ausgehend von $\{q_o\}$ längs aller Folgen von Buchstaben die Mengen von Zuständen, die so erreichbar sind.



Formal:

Sei $a = (Q, \Sigma, q_0, \Delta, F)$ gegeben, so definiere $a' = (Q', \Sigma, q_0', \delta', F')$, so gilt:

$$\begin{aligned}
&Q' = 2^Q = \{R \mid E \subseteq Q\} \text{ Potenzmenge von Q} \\
&q_0' = \{q_0\} \\
&\text{Für } R \subset Q : \\
&\delta'(R, a) = \{\underline{q} \mid \exists r \in R, (r, a, \underline{q}) \in \Delta\} \\
&F' = \{R \mid R \subseteq Q, R \cap F \neq Q\}
\end{aligned}$$

Behauptung: $a : p \rightarrow^w q \Longleftrightarrow q \in \delta'(\{p\}, w)$

Beweis: Per Induktion über (w)

$\underline{IA:}$ Sei $\mid w \mid = 0$ und $a : p \rightarrow^\varepsilon q \Longleftrightarrow p = q \Longleftrightarrow q \ \in \delta'(\{p\}, \varepsilon)$.

$\underline{IS:}$ Sei $w = va$ und es gelte als IV $a : p \rightarrow^v r \Longleftrightarrow r \in \delta'(\{p\}, v)$, so gilt:

$$\begin{aligned}
a : p \rightarrow^{va} q &\Longleftrightarrow \exists r \in Q : a : p \rightarrow^v r \\
&\Longleftrightarrow \exists r \in Q : r \in \delta'(\{p\}, v)
\end{aligned}$$

Damit gilt:

$$\begin{aligned}
a : p \rightarrow^{va} q &\Longleftrightarrow \exists r \in Q : a : p \rightarrow^v r, a : r \rightarrow^a q \\
&\Longleftrightarrow^{IV} \exists r \in Q : r \in \delta'(\{p\}, v), (r, a, q) \in \Delta \\
&\Longleftrightarrow^{Def. \delta'} q \in \delta'(R, a) = \delta'(\delta'(\{p\}, v), a) \\
&\Longleftrightarrow^{Fortsetzung \delta' auf \Sigma^*} q \in \delta'(\{p\}, va)
\end{aligned}$$

Für die Akzeptanz gilt also

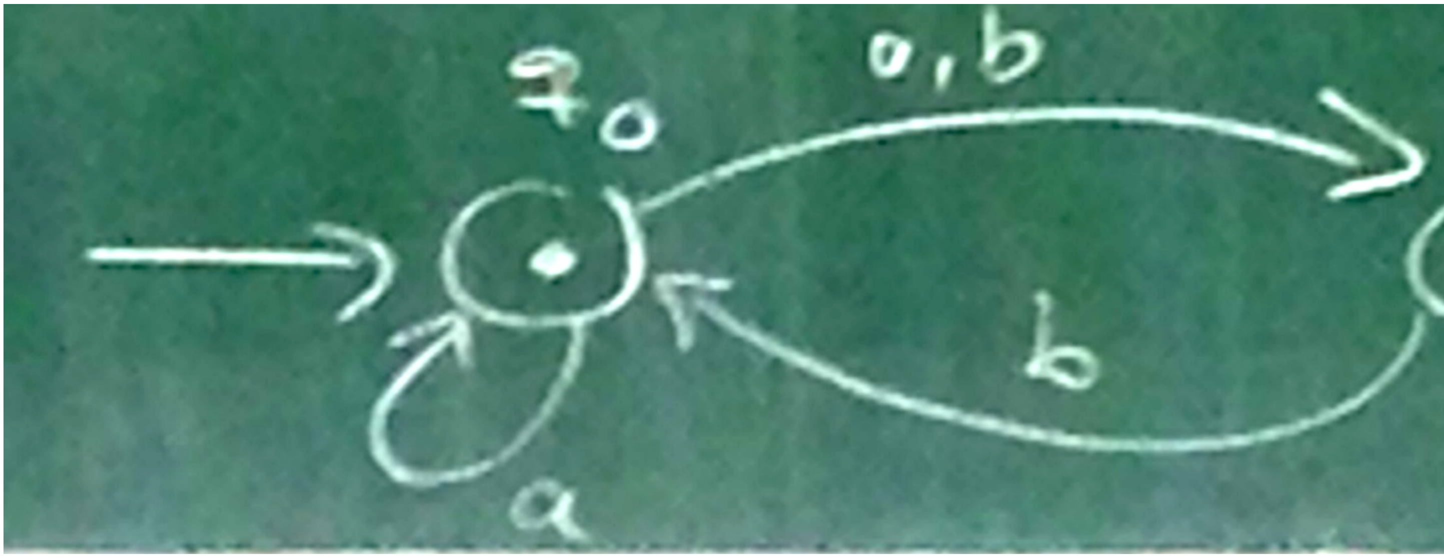
$$\begin{aligned}
\text{a akzeptiert w} &\Longleftrightarrow \exists q \in F : a : q_0 \rightarrow^w q \\
&\Longleftrightarrow^{Behauptung} \exists q \in F : q \in \delta'(\{q_0\}, w) \\
&\Longleftrightarrow \delta'(\{q_0\}, w) \cap F \neq \emptyset \\
&\Longleftrightarrow^{Def. F'} \delta'(\{q_0\}, w) \in F' \\
&\Longleftrightarrow^{Def. F'} \text{a' akzeptiert w}
\end{aligned}$$

Das bedeutet a und a' sind äquivalent (L(a) = L(a')).

Es wurden die folgenden Definitionen eingeführt

- *Regulär*
- *Beschränkung auf erreichbare Zuständen*

Identifizierung äquivalenter Zustände



Definition

Zwei Zustände q, q' eines DEA $a = (Q, \Sigma, q_0, \delta, F)$ heißen äquivalent, falls $\forall w \in \Sigma^* : (\delta(q, w) \in F \iff \delta(q', w) \in F)$

Schreibweise

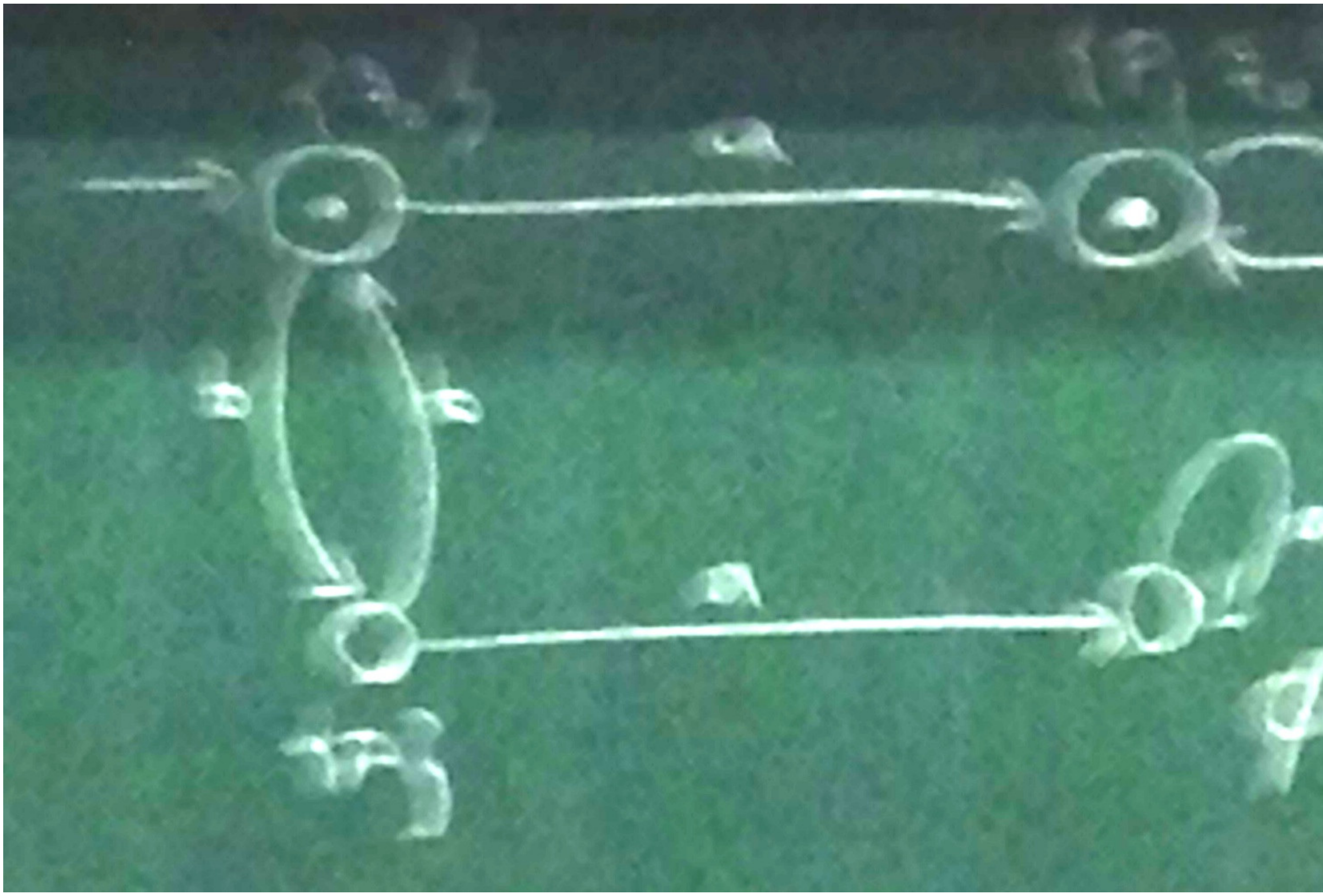
$q \sim q'$

Bemerkungen

1. $\sim$ ist eine Äquivalenzrelation. Es sei $\overline{q}$ die Äquivalenzklasse von q ($\overline{q} = \{p \mid p \sim q\}$)
2. $q \sim q', a \in \Sigma \Rightarrow \delta(q, a) \sim \delta(q', a)$

   Ansonsten:

$$\begin{aligned}\exists w \in \Sigma^* :& [\delta(\delta(q,a),w) \in F \wedge \delta(\delta'(q',a),w) \notin F \\ & \text{oder } \delta(\delta(q,a),w) \notin F \wedge \delta(\delta'(q',a),w) \in F] \\ & \implies \exists w \in \Sigma^* : \\ [\delta(q,aw) \in F \wedge \delta'(q',aw) \notin F \\ & \text{oder } \delta(q,aw) \notin F \wedge \delta(q',aw) \in F] \\ & \implies q \nsim q' \implies WSP.\end{aligned}$$



Es wurden die folgenden Definitionen eingeführt

- *Äquivalenzklassen*
- *Reduzierter DEA*

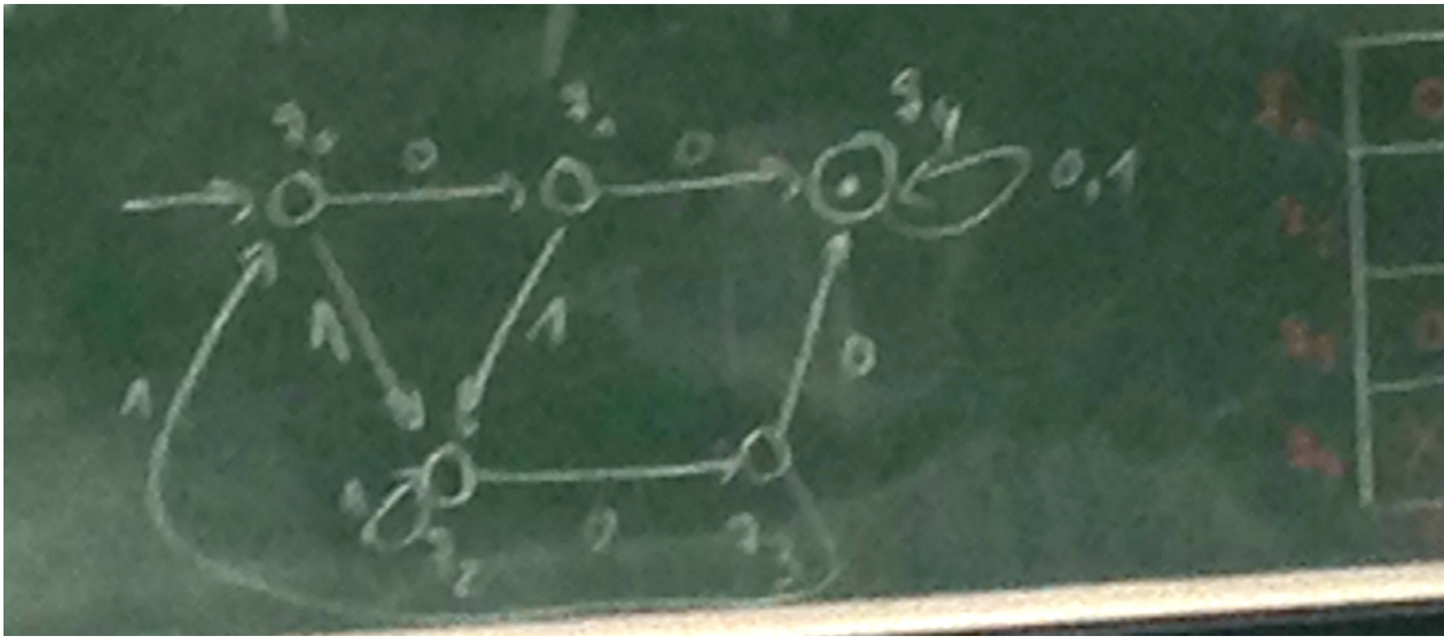
1.1.4 4. Vorlesung

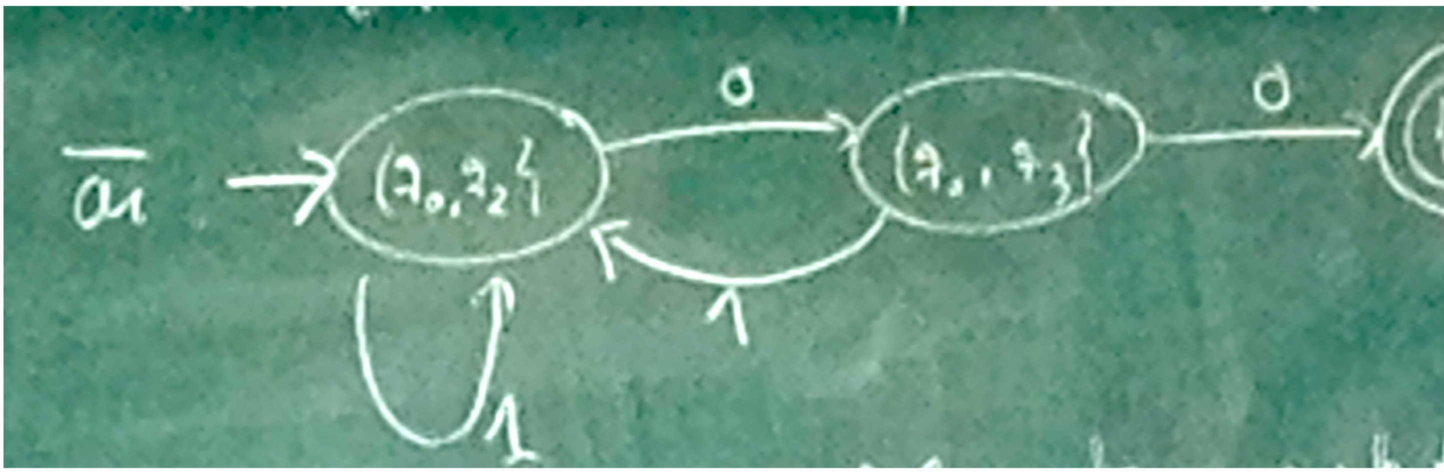
Allgemein zum Bestimmen von äuqivalenten Zuständen

Eingabe

DEA a, der nur erreichbare Zustände enthält

1. stelle eine Tabelle aller Zustandspaare $\{q, q'\}$ mit $q \neq q'$ von a auf.
2. markiere alle Paare $\{q, q'\}$ mit $q \in F \wedge q' \notin F$ (bzw. umgekehrt)
3. für jedes noch unmarkierte Paar $\{q, q'\}$ und jedes $a \in \Sigma$ teste, ob $\{\delta(q, a), \delta(q', a)\}$ bereits markiert ist. Wenn ja markiere auch $\{q, q'\}$
4. wiederhole Schritt (3) bis sich keine Änderungen mehr in der Tabelle ergeben.
5. alle noch unmarkierten Paare sind äuqivalente Zustände.





Korrektheit

1. es werden nur nicht-äquivalente Zustandspaare makiert.
2. der Algorithmus terminiert (es gibt nur endl. viele Paare)
3. bei Abbruch des Algorithmus sind alle nicht-äquivalenten Paare markiert:

Voraussetzung: Der Algorithmus endet nach m+1 Iterationen von Schritt 3.

Annahme: Ein nicht-äquivalentes Paar (p,q) ist noch nicht markiert. Wähle l > m +1 minimal aus, sodass

$$a : p \rightarrow^w F, a : q/rightarrow^w \notin F \wedge \mid w \mid = l$$

Sei w = u v mit $\mid v \mid = m + 1$, so gilt:

$$a : p \rightarrow^u p' \rightarrow^v F, a : q \rightarrow^u q' \rightarrow^v \notin F$$

Das Paar (p', q') ist in der (m + 1) Iteration trennbar mit $a : p' \rightarrow^v F, a : q' \rightarrow^v \notin F$, aber nicht in einer früheren Iteration (ansonsten wäre l nicht minimal). Daraus folgt, dass in der (m + 1) Iteration noch ein nicht-äquivalentes Paar gefunden wurde, woraus $\geq (m + 2)$ Iterationen und daher ein Wiederspruch.

Es wurden die folgenden Definitionen eingeführt

- aequiv
- aequivWord
- indexOf

Satz von Nerode

Behauptung:

$L \subset \Sigma^*$regulär $\Longleftrightarrow$ Index von $\simeq_L$ ist endlich

Beweis:

$\Rightarrow$:

Sei a = (Q, Σ, q_0, δ, F) ein DEA, der L erkennt, so ist zu zeigen, dass verschiedene $\simeq_L$-Klassen in Σ^* bestimmen verschiedene $\simeq_a$-Klassen in Q.

Dann: $\mid Q \mid \geq Index(\simeq_a) \geq Index(\simeq_L)$

Seien $u \not\simeq_L v$, d.h. $[u]_L \neq [v]_L$ gegeben. Bilde nun $\delta(q_0, u), \delta(q_0, v)$ und $\delta(q_0, u) \not\sim_a \delta(q_0, v)$.

Annahme: $\delta(q_0, u) \sim_a \delta(q_0, v)$

$$\begin{aligned}
&\delta(q_0, u) \sim_a \delta(q_0, v)\\
&\forall w \in \Sigma^* : \delta(\delta(q_0, u), w) \in F \Longleftrightarrow \delta(\delta(q_0, v), w) \in F\\
&\forall w \in \Sigma^* : uw \in L \Longleftrightarrow vw \in L\\
&u \simeq_L v = WSP
\end{aligned}$$

$\Leftarrow$:

Sei Index ($\simeq_L$) endlich. Bilde DEA a, der L erkennt.

a = $(Q^\sim, \Sigma, q_0^\sim, \delta^\sim, F^\sim)$ mit

$Q^\sim$ = Menge der $\simeq_L$ - Äquivalenzklassen

$q_0^\sim = [\varepsilon]$

$F^\sim = \{[u] \mid u \in L\}$

und $\forall a \in \Sigma : \delta^\sim([u], a) = [ua]$

Beachte:

$\forall w \in \Sigma^* : [u] = [v] \Longrightarrow [uw] = [vw]$

Hieraus folgt: $\delta^\sim$ ist wohldefiniert und es gilt ferner (per Induktion über $\mid w \mid$):

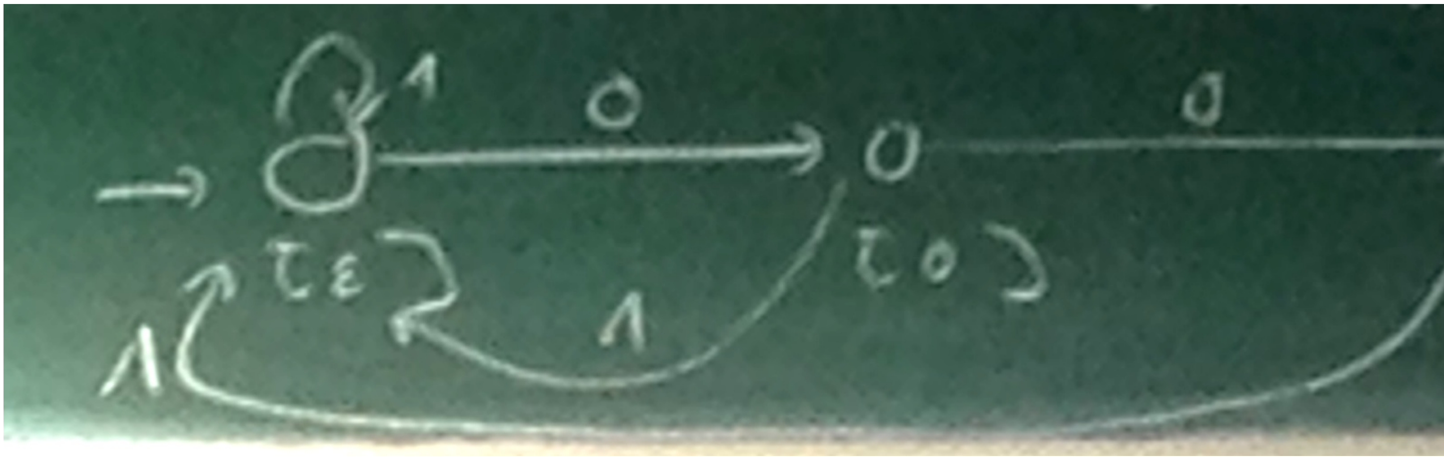
$\forall w \in \Sigma^* : \delta^\sim([u], w) = [uw]$

Dann gilt

$$\begin{aligned}
w \in L(a) &\Longleftrightarrow \text{a akzeptiert w}\\
&\Longleftrightarrow \delta^\sim([\varepsilon], w) \in F^\sim\\
&\Longleftrightarrow [\varepsilon w] = w \in F\\
&\Longleftrightarrow w \in L. \Rightarrow L(a) = L
\end{aligned}$$

Satz:

Der kanonische DEA a ist isomorph zu jedem DEA $a^{\sim}$, der gemäß dem Reduktionsverfahren aus einem belibigen DEA a, der L erkennt, entsteht.



1.1.5 5. und 6. Vorlesung

Angabe von nicht-regulären Sprachen

Dies folgt aus dem Satz von Nerode.

$$\text{L ist nicht-regulär} \Longleftrightarrow \text{Index von } \simeq_L \text{ ist unedlich (d.h. unendl. viele Äquivalenzklassen)}$$

Beziehungsweise es gibt Wörter $u_i \in \Sigma^*$ für beliebige $i \geq 0$, so dass für $i \neq j$ jeweils ein Wort w existiert mit $u_i w \in L$ und $u_j w \notin L$.

Beispiel $L_i = \{a^i b^i \mid i \geq 0\}$

Wähle $u_i = a^i$. Für $i \neq j$ gilt mit $w = b^i$:

$$u_i w = a^i b^i \in L_1 \text{ und}$$
$$u_j w = a^j b^i \notin L_1 (\text{da } i \notin j)$$

Daraus folgt, dass L_1 nicht regulär ist.

Sei $L_2 = \{0^i 1^j \mid ggT(i,j) = 1\}$ und wähle $u_P = 0^P$, wobei p Primzahl ist (beachte es gibt unendlich viele Primzahlen). Für $P \neq q$ gilt mit $w = 1^q$:

$$u_p w = 0^P 1^q \in L_2, \text{ weil } ggT(p,q) = 1$$
$$u_p w = 0^q 1^q \notin L_2, \text{ weil } ggT(q,q) = 1 \geq 2$$

Pumping-Lemma

Es sei L eine reguläre Sprache. Dann gibt es eine Zahl $n \in N$, sodass sich alle Wörter $x \in L$ mit $\mid x \mid \geq n$ zerlegen lassen in x = u v w mit den folgenden Eigenschaften:

1. $\mid v \mid \geq 1$
2. $\mid uv \mid \leq n$
3. $\forall i \geq 0 : uv^i w \in L$

Formal

$$\text{L ist regulär} \Rightarrow \exists n \in N : \forall x \in L :\mid x \mid \geq n \Rightarrow x = uvw \wedge 1, 2 \text{ und } 3$$

Beweis

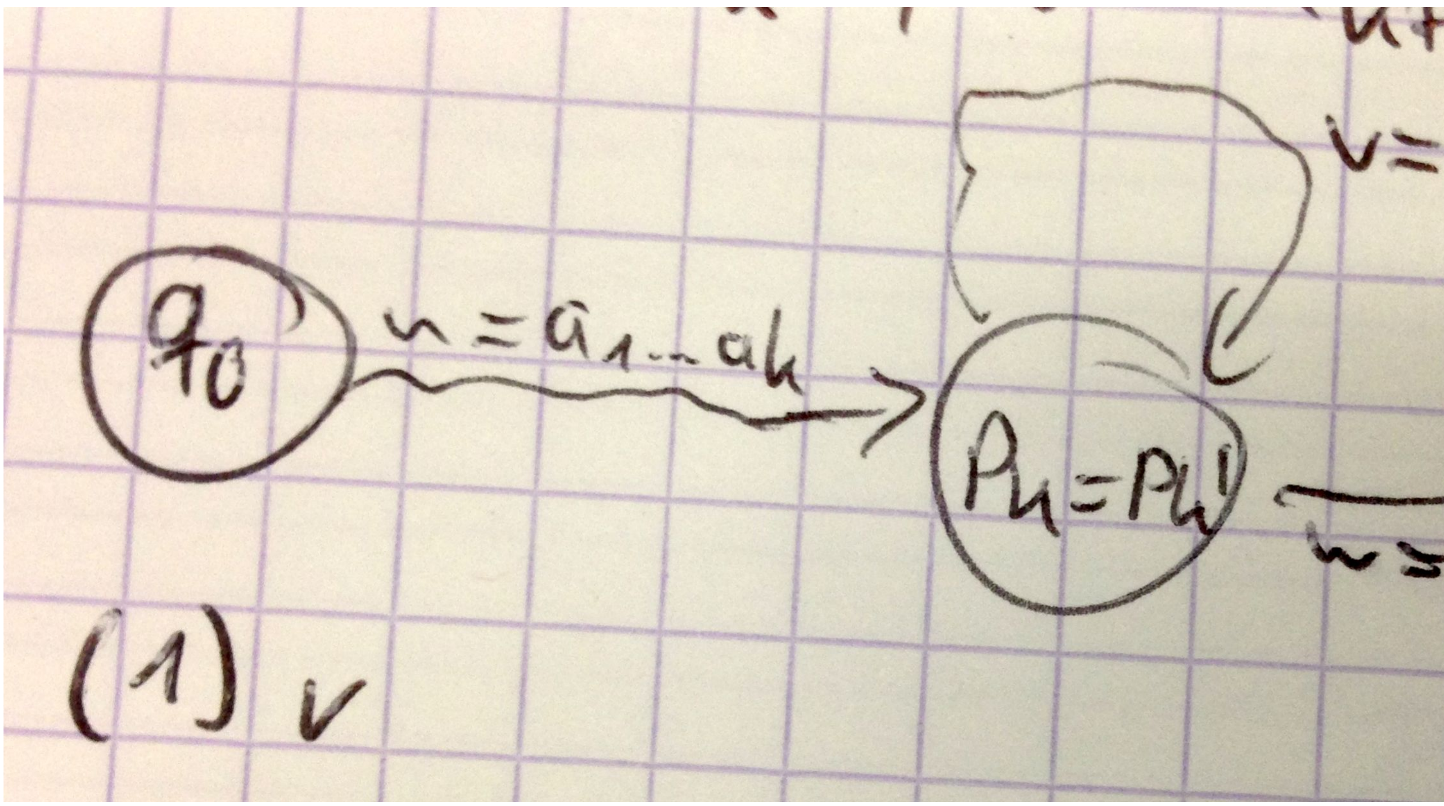
Sei a ein DEA der L akzeptiert. Wir wählen $n =\mid Q \mid$. Sei $x \in L$ mit $\mid x \mid= m \geq n$ und $x = a_1, ..., a_n$ mit $a_i \in \Sigma$. Dann folgt daraus, dass ein Pfad existiert mit $\Pi = p_0 a_1 p_1 a_2 p_2 .... a_m p_m$ mit $p_0 = q_0$ und $p_w \in F$.

Die Anzahl der Zustände in Π ist $m + 1 \geq n + 1$ (d.h. ein Zustand kommt mindestens zweimal vor). Daraus folgt, dass unter den ersten n+1 Zuständen $p_0, ..., p_n$ existiert ein Paar k,k' mit $0 \leq k < k' \leq n$ und $p_k = p_{k'}$. Wähle $u = a_1 ... a_k$, $v = a_{k+1} ... a_{k'}$, $w = a_{k'+1} ... a_m$.

Das heißt, das wegen $p_k = p_{k'}$ gilt:

$$\forall i \geq 0 : uv^i w \in L(a) = L$$

Da $k < k'$ gilt $\mid v \mid\geq 1$ und da $k' \leq n$ gilt $\mid nv \mid\geq n$.



Beispiel 1

Sei $L = \{a^i b^i \mid i \geq 1\}$ nicht-regulär.

Annahme: L ist regulär, also gibt es ein $n \in N$, sodass alle Wörter $x \in L$ die Länge $x \geq n$ haben. Wähle $x = a^n b^n$ der Länge 2n, so folgt darus:

$$\begin{aligned} &\exists Zerlegung x = uvw : (1) \mid v \mid \geq 1, (2) \mid u, v \mid \leq n \\ \Rightarrow & uv = a^m : m \leq n \wedge m \geq 1 \\ \Rightarrow & v = a^{m'} : 1 \leq m' \leq m \\ \Rightarrow & uw = a^{n-m'} b^n \in L \end{aligned}$$

Dies ist ein Widerspruch da $n - m' \neq n$ und nach Annahme $a^n b^n$ kann dies nicht gelten.

Beispiel 2

$L = \{0^m \mid$ m ist Quadratzahl$\}$ ist nicht regulär.

Annahme: L ist regulär, woraus folgt, dass $\exists n \in N$, sodass alle Wörter $x \in L$ die Länge n haben. Wähle $x = 0^{n^2}$, woraus folgt, dass $x \in L$ ist und $\mid x \mid = n^2 \geq n$. Betrachtet man nun eine beliebige Zerlegung x = u v w mit $1 \leq \mid v \mid \leq \mid uv \mid \leq n$, so ergibt sich $uv^2w \in L$. Andererseits gilt:

$$\begin{aligned} n^2 &= \mid x \mid \\ &= \mid uvw \mid \\ &<^{\mid v \mid \geq 1} \mid uv^2w \mid \\ &= \mid uvw \mid + \mid v \mid \\ &= n^2 + \mid v \mid \\ &\leq n^2 + \mid uv \mid \\ &\leq n^2 + n \\ &< n^2 + 2n + 1 \\ &= (n+1)^2 \end{aligned}$$

Dies ist ein Widerspruch, da es eine Quadratzahl ist.

Es wurden die folgenden Definitionen eingeführt

- regex

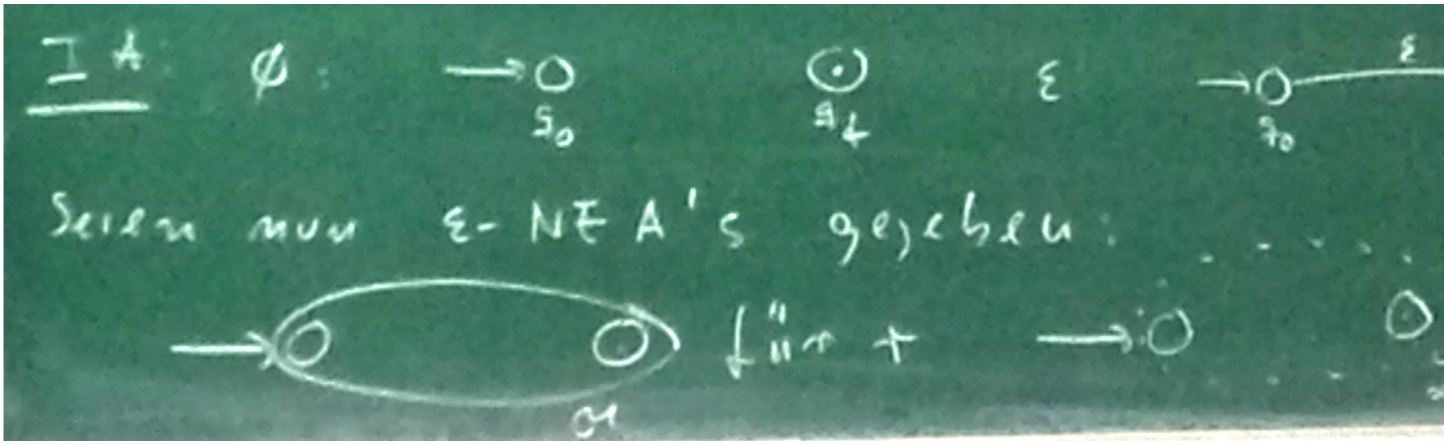
Satz von Kleene

Eine Sprache $L \subseteq \Sigma^*$ ist durch einen NEA erkennbar $\Longleftrightarrow$ L ist durch einen regulären Ausdruck definiert.

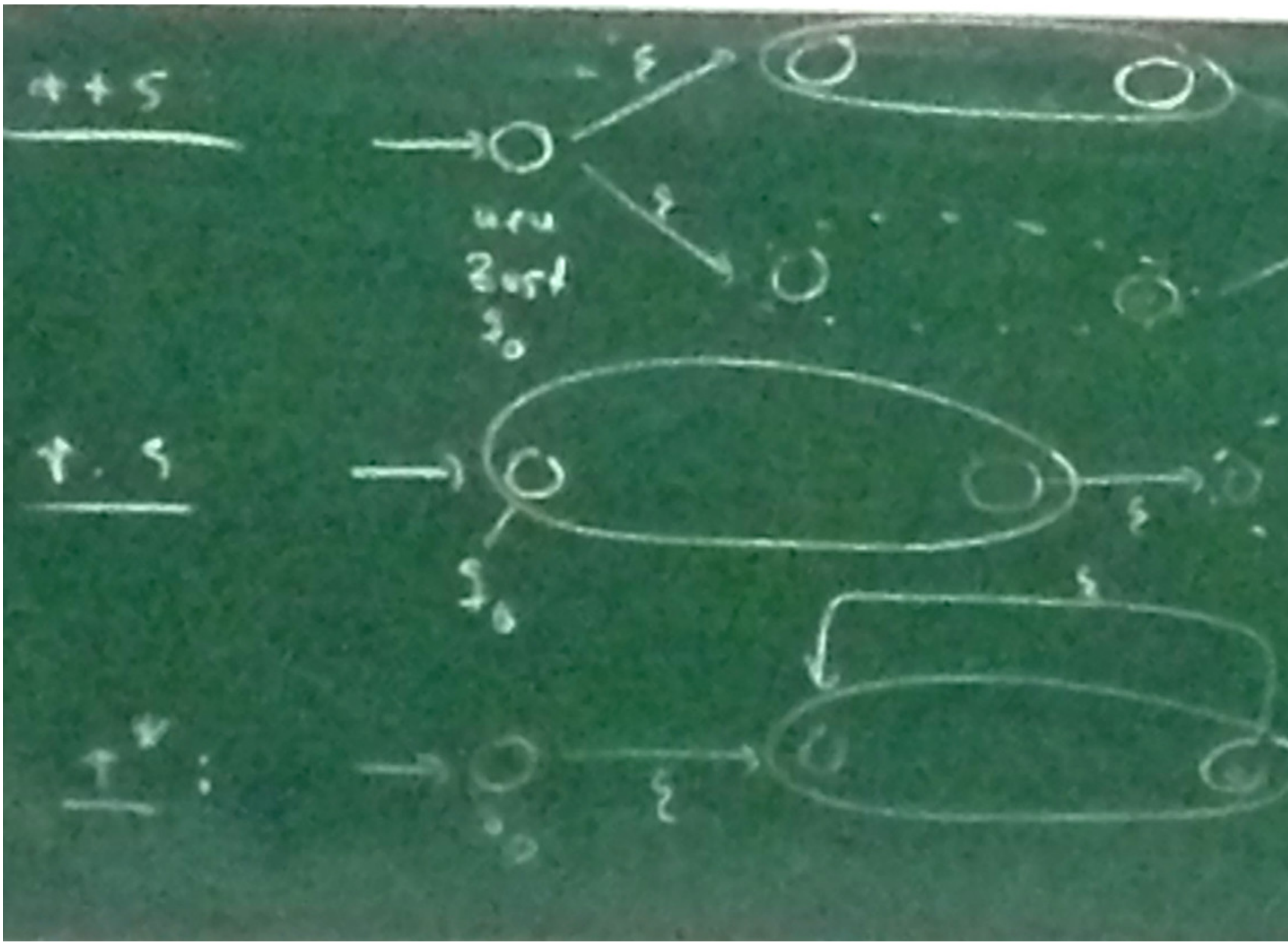
Beweis

$\Longleftarrow$: durch Induktion über Aufbau der regulären Ausdrücke. Finde für jeden regulären Ausdruck r einen ε-NEA a_r mit einem Endzustand $L(r) = L(a_r)$

Induktionsanfang:



Induktionsschluss:



$\Longrightarrow$: Sei $a = (Q, \Sigma, q_0, \Delta, F)$ als NEA gegeben mit $Q = \{q_0, q_1, ..., q_n\}$. Wir definieren für $0 \leq i, j \leq n$:

$$L_{ij} = \{w : a : q_i \to^w q_j\}$$

So gilt

$$L(a) = \bigcup_{q_j \in F} L_{0j}$$

Es genügt nun r_{ij} für L_{ij} zu finden. Hierzu gelte $L(r) = \Sigma_{q_j \in F} r_{0j}$ Der Index(q) für $q \in Q$ sei dasjenige j mit $q_j = q$.

Sei

$L_{ij}^k = \{w \mid \exists P fad \Pi = p_0 a_1 p_1 ... p_{m-1} a_m p_m : w = a_1 ... a_m, p_0 = q_i, p_m = q_j, Index(P_l) < k$ für $l = 1, ..., m-1\}$

Beweis: $L_{ij} = L_{ij}^{n+1}$

Zeige: $\forall k = 0, ..., n+1 : \forall i, j : L_{i,j}^k$ ist durch einen regulären Ausdruck definierbar.

k = 0:

Falls i = j: $L_{ij}^0 = \{\varepsilon\} \cup \{a \mid (q_i, a, q_j) \in \Delta\}$ Sonst: $\{a \mid (q_i, a, q_j) \in \Delta\}$

Daraus folgt für r_{ij}^0:

Falls i = j: $r_{ij}^0 = \varepsilon + \Sigma_{(q_i,a,q_j) \in \Delta} a$ Sonst: $r_{ij}^0 = \Sigma_{(q_i,a,q_j) \in \Delta} a$

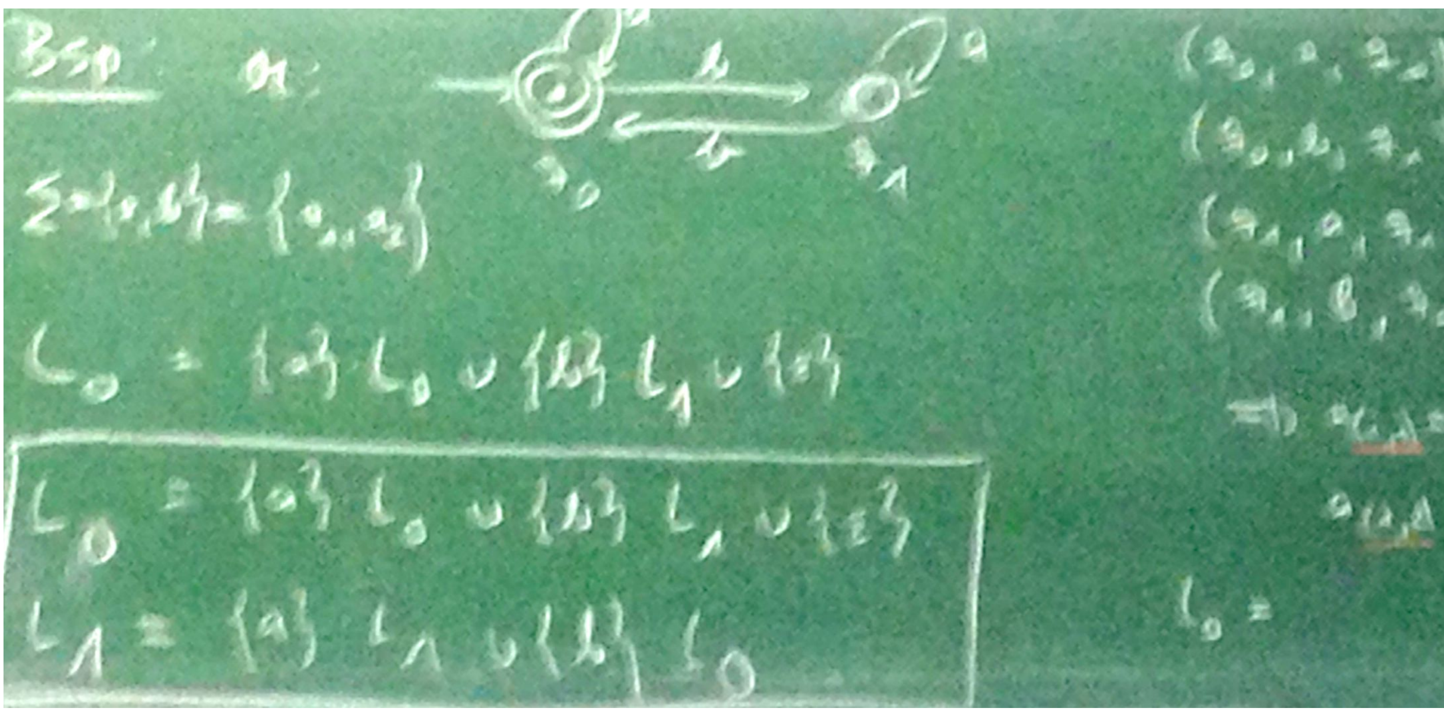
Induktionsschritt:

Es sei L_{ij}^k durch den regulären Ausdruck r_{ij}^k definierbar, also gilt:

$$\begin{aligned} w \in L_{ij}^{k+1} &\iff w \in L_{ij}^k \vee \exists m : w = w_0 w_1 ... w_m \text{ und } w_0 \in L_{ij}^k, w_1, ..., w_{m-1} \in L_{kk}^k \text{ und } w_m \in L_{Rj}^k \\ &\implies L_{ij}^{k+1} = L_{ij}^k \cup L_{ik}^k (L_{kk}^k)^* L_{kj}^k \text{ und } r_{ij}^{k+1} = r_{ij}^k + r_{ik}^k (r_{kk}^k)^* r_{kj}^k \end{aligned}$$

Beispiel

k = 0:



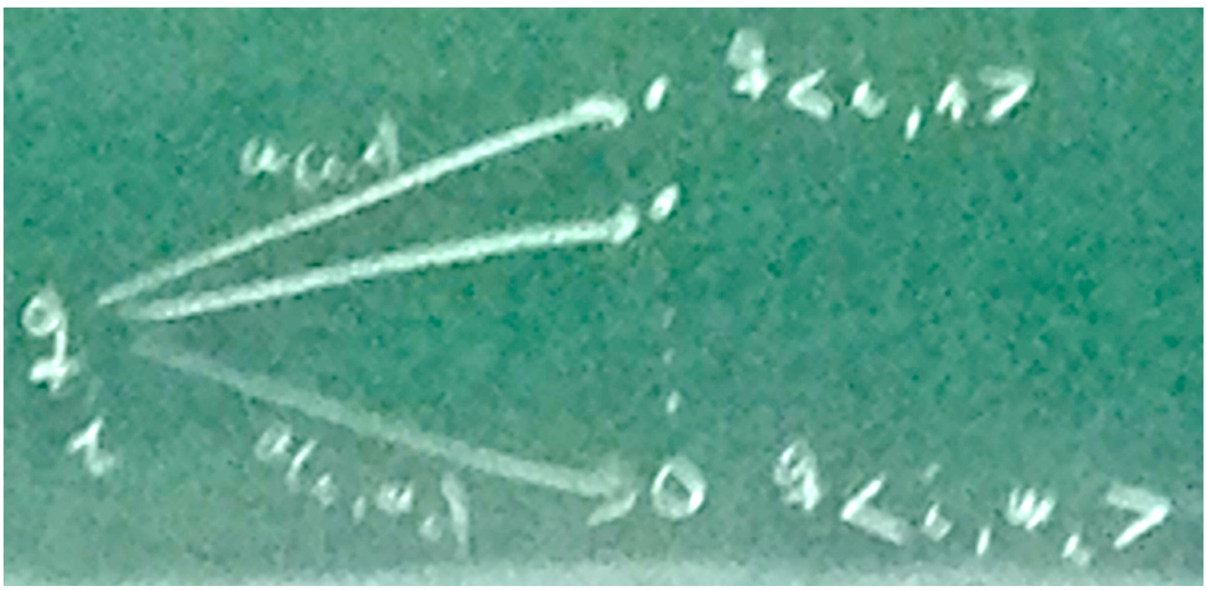
k = 1:

$$
\begin{aligned}
r_{00}^1 &= r_{00}^0 + r_{00}^0(r_{00}^0)^* r_{00}^0 = (\varepsilon+0)+(\varepsilon+0)^*+(\varepsilon+0) = 0^* \\
r_{01}^1 &= r_{01}^0 + r_{00}^0(r_{00}^0)^* r_{01}^0 = 1 + 0^*1 = 0^*1 \\
r_{02}^1 &= r_{01}^0 + r_{00}^0(r_{00}^0)^* r_{02}^0 = \emptyset \\
r_{11}^1 &= r_{11}^0 + r_{10}^0(r_{00}^0)^* r_{01}^0 = \varepsilon + \emptyset 1 = 1 \\
.... &= .... \\
r_{02}^2 &= r_{02}^1 + r_{02}^1(r_{11}^1)^* r_{12}^1 = \emptyset + 0^*1\varepsilon 1 = 0^*11 \\
r_{02}^3 &= r_{02}^2 + r_{02}^2(r_{22}^2)^* r_{22}^2 = 0^*11 + 0^*11\varepsilon = 0^*11 = r_{02} = L(a)
\end{aligned}
$$

Ziel: Elegantes Verfahren für die Konstruktion von regulären Ausdrücken

Sei $a = (Q, \Sigma, q_0, \Delta, F)$ mit $Q = \{q_0, ..., q_n\}$, $a_i = (Q, \Sigma, q_i, \Delta, F)$, $L_i = L(a_i)$ für $0 \leq i \leq n$

Nummerierung der Transiitonen aus q_i: $(q_i, a_{(i,1)}, q_{(i,1)}), ..., (q_i, a_{(i,w_i)}, q_{(i,w_i)})$



$(*)_i$:

$$L_i = \{a_{(i,1)}\} \cdot L_{(i,1)} \cup ... \cup \{a_{(i,w_i)}\} L_{(i,w_i)} \cup E_i$$

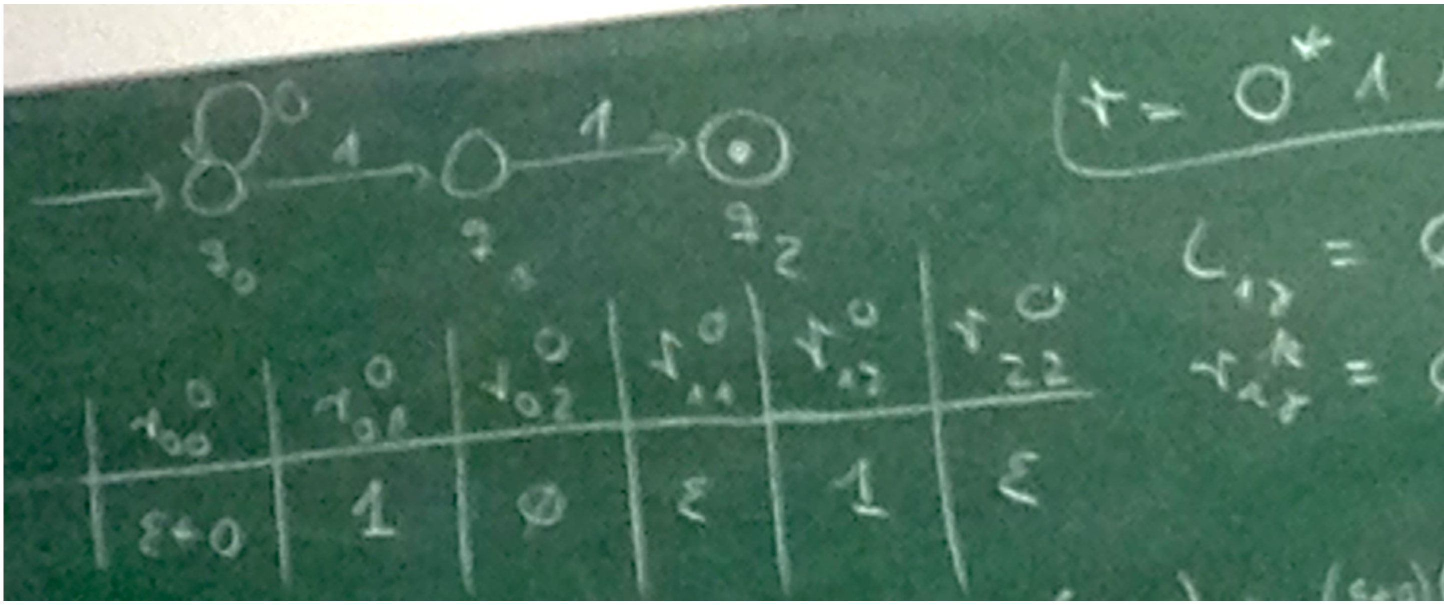
Mit

$$E_i = \begin{cases} \{\varepsilon\} & \text{falls } q_i \in F \\ \emptyset & \text{sonst} \end{cases}$$

Zur Vereinfachung: $\{a\}L_j \cup \{b\}L_j = \{a, b\}L_j$

Sei also $A_{i,j} = \{a \mid \{a\}L_j$ kommt in der Zeile $(*)_j$ vor.$\}$ Daraus folgt: $L_i = A_{i,0}L_0 \cup A_{i,1}L_1 \cup ... \cup A_{i,n}L_n \cup E_i$, wobei $A_{i,j}\tilde{} in\Sigma$.

Beispiel



Satz: Das zum NEA gehörige Gleichungssystem ist eindeutig lösbar

Das zum NEA a gehörige Gleichungssystem GS(a) $X_i = \bigcup_{j=0}^{n} A_{ij} X_j \cup E_i$ mit A_{ij}, E_i wie oben ist eindeutig lösbar und $L_0, ..., L_n$ mit $L_i = L(a_i)$ ist die Lösung.

Beweis:

$(L_0, ..., L_n)$ ist Lösung wegen der Definition von GS(a).

Eindeutigkeit:

Seien $(K_0, ..., K_n), (K_0^{'}, ..., K_n^{'})$ Lösungen von GS(a). Zeige $\forall w \in \Sigma^* : \{w\} \cap K_i = \{w\} \cap K_i^{'}$

Indktion über |w|:

IA: |w| = 0

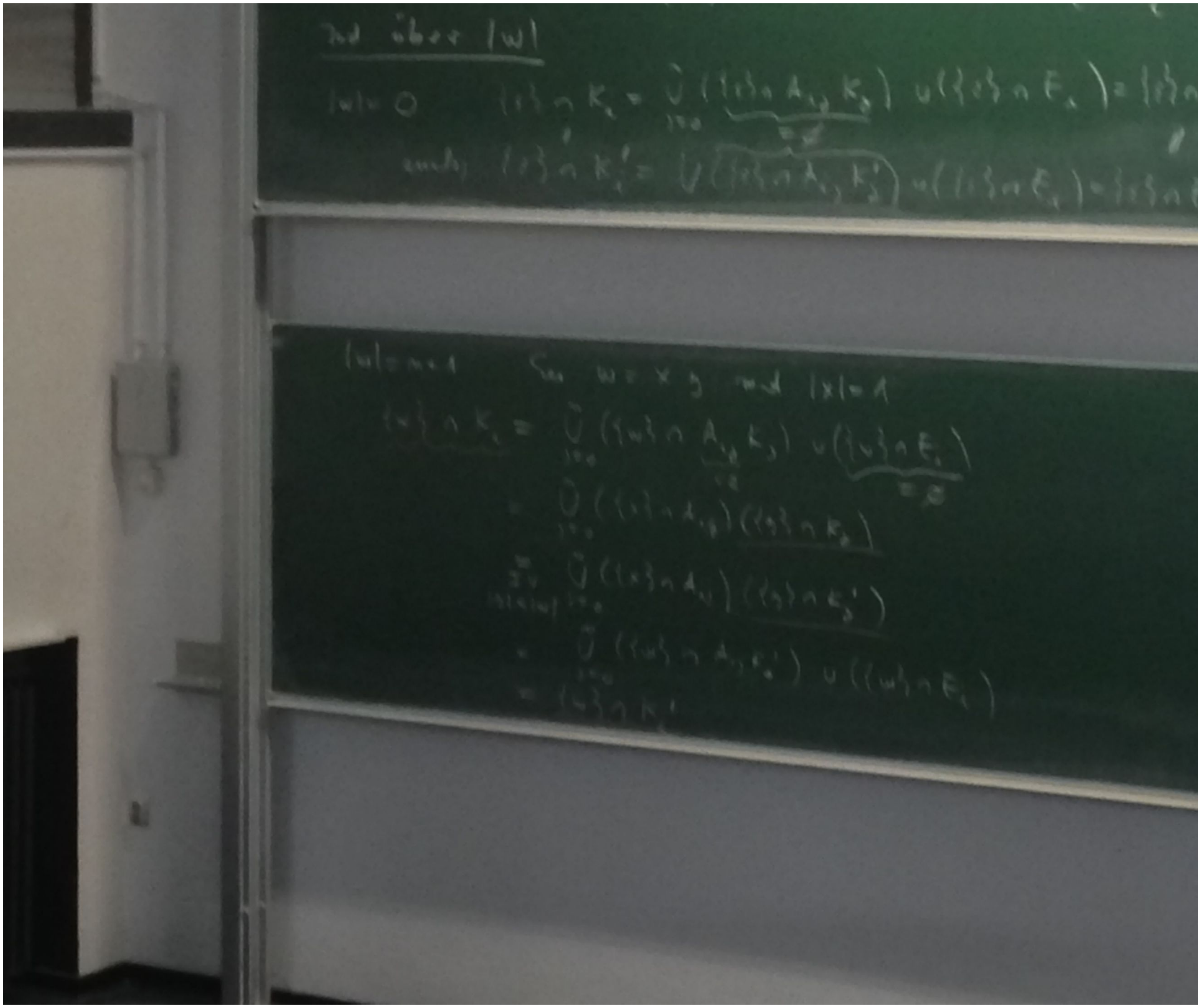
$$\begin{aligned} \{\varepsilon\} \cap K_i &= \bigcup_{j=0}^{n} (\{\varepsilon\} \cap A_{ij} K_j) \cup (\{\varepsilon\} \cap E_i)) \\ &= \{\varepsilon\} \cap E_i \end{aligned}$$

Analog für K'

IS: |w| = n + 1

Sei $w = xy$ mit |x| = 1

$$\begin{aligned} \{w\} \cap K_i &= \bigcup_{j=0}^{n} (\{w\} \cap A_{ij} K_j) \cup (\{w\} \cap E_i) \\ &= ... \\ &= \{w\} \cap K_L^{'} \end{aligned}$$



CHAPTER

TWO

DEFINITIONSVERZEICHNIS:

2.1 Definitionen

2.1.1 Bezeichnungen

Σ

Alphabet

ς

Leeres Wort

Σ^*

Menge der Wörter über Σ

$\{a, b\}^* = \{\varepsilon, a, b, ab, aa, ba, bb, ...\}$

$$\mid w \mid = \text{Länge des Wortes } w \in \Sigma^* \text{(Anzahl der Buchstaben)}$$

$$\mid w \mid_a = \text{Anzahl der a's in } w \in \Sigma^*$$

Verkettung

$u \cdot v \equiv uv$

Wortmenge / formale Sprache in $P(\Sigma^*)$

L, L'

$$L \cdot L' = \{uv \mid u \in L, v \in L'\}$$
$$L^o = \{\varsigma\}$$
$$L^{n+1} = L^n \cdot L$$
$$L^1 = L$$
$$L^* = \bigcup_{u \in \mathbb{N}} L^n$$

2.1.2 Transitionssysteme, endl. Automaten

Transitionssystem

Ein Transitionssystem hat die Form $a = (Q, \Sigma, I, \Delta, F)$

- Q ist die Zustandsmenge
- Σ ist ein endliches Alphabet
- $I, F \subseteq Q$ ist die Menge der Anfangs- bzw. Endzustände
- $\Delta \subset QX\Sigma XQ$ ist die Transitionsrelation

a ist ein endliches Transitionssystem, wenn Q endlich ist.

nicht - deterministischer endlicher Automat

Ein endliches Transitionssystem heißt nicht - deterministischer endlicher Automat (NEA / $I = \{q_0\}$ für $q_0 \in Q$)

Beispiel:

$$a = (Q = \{q_0, q_1, q_2\}, \Sigma = \{0, 1\}, I = \{q_0\}), \Delta = \{(q_0, 0, q_0), (q_0, 1, q_0), (q_1, 0, q_2)\}, F = \{q_2\}$$

Notation bei NEA's:

$$a = (Q, \Sigma, q_0, \Delta, F)$$

Pfad

Ein Pfad durch a ist eine Folge $\Pi = P_0 a_1 P_1 a_2, ..., a_n P_n$ mit $(P_i, a_{i+1}, P_{i+1}) \in \Delta$ für i =0,. . . , n-1.

Die **Länge** von Π sei N und die **Beschriftung** $\beta(\Pi)$ sei $a_1, a_2, ..., a_n$.

$a : p \rightarrow^w q$ für $w \in \Sigma^*$ besagt, dass es mindestens einen Pfad Π durch a von p nach q mit Beschriftung $\beta(\Pi) = w$ gibt.

Beispiel:

$$\Pi = q_0 0 q_0 1 q_0 0 q_1 0 q_2, \text{ Beschriftung } \beta(\Pi) = 0100$$

Akzeptiert

a akzeptiert $w \in \Sigma^* \iff \exists p \in I, \exists q \in F$ mit $a : p \rightarrow^w q$

Sprache

Für NEA sei $L(a) = \{w \in \Sigma^* \mid akzeptiertw\}$ die durch a erkannte Sprache.

Beispiel: a akzeptiert w = 0 1 0 0 (da der obrige Pfad: $q_0 \in I, q_2 \in F$)

Frage: Welches Wort wird nicht akzeptiert? (z.B. w = 0)

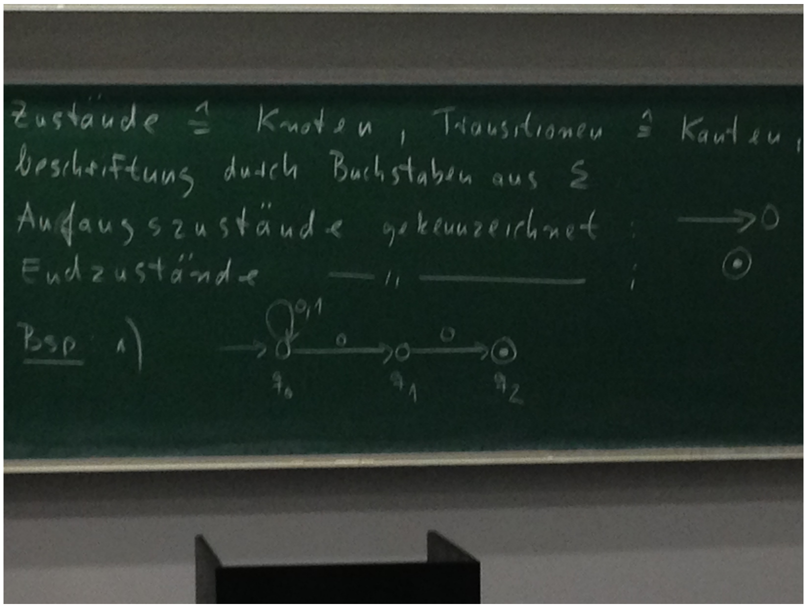
Darstellung des NEA a durch Graphen:

- Zustände $\equiv$ Knoten
- Transitionen $\equiv$ Kanten

- Kantenbeschriftung durch Buchstaben aus Σ

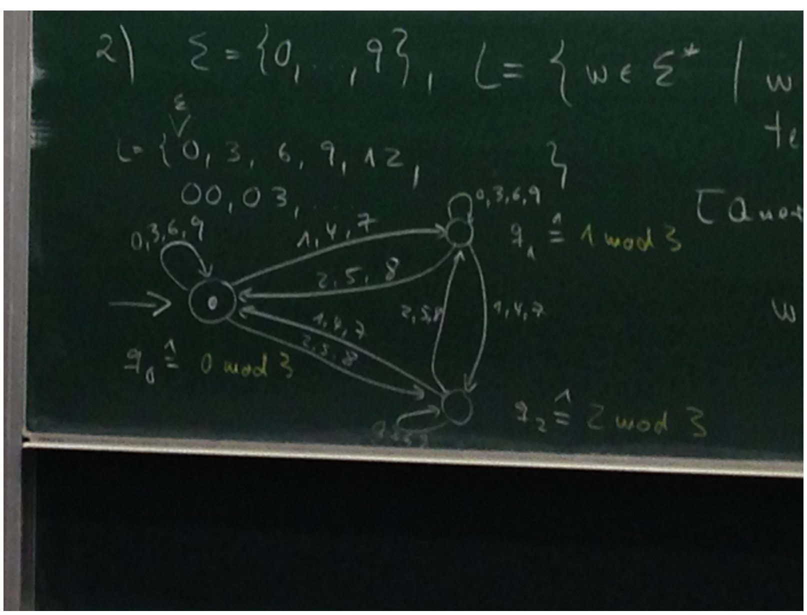
Beispiel 1:

$$\begin{aligned} L &= \{w \in \Sigma^* \mid w = w'00 \text{ für } w' \in \Sigma^*\} \\ &= \text{ Menge der Wörter, die mit 00 enden} \end{aligned}$$



Beispiel 2:

$$\begin{aligned} &\Sigma = \{0, ..., 9\}, \\ &L = \{w \in \Sigma^* \mid \text{ w stellt eine durch 3 teilbare Dezimalzahl dar } \} \\ &L = \{\varsigma, 0, 3, 6, 9, 12, ..., 00, 03, ...\} \end{aligned}$$



Deterministisch

a ist deterministisch, wenn

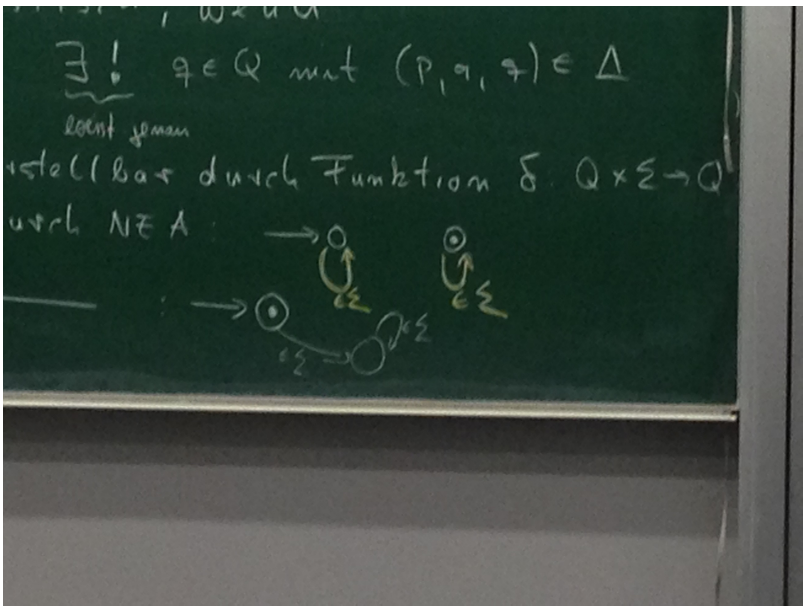
$$\forall p \in Q : \forall a \in \Sigma : \exists!(\text{existiert genau}) q \in Q mit (p, a, q) \in \Delta$$

In diesem Fall ist Δ darstellbar durch eine Funktion $\delta : QX\Sigma \to Q$

Beispiel

- L = ∅ akzeptiert durch NEA:
- L = { ς }

In dieser Reihenfolge:



Präfix

Sei $u, w \in \Sigma^*$, so heißt u Präfix von w, wenn

$$\exists v \in \Sigma^* : w = uv$$

Suffix

Sei $u, w \in \Sigma^*$, so heißt u Suffix von w, wenn

$$\exists v \in \Sigma^* : w = vu$$

Infix

Sei $u, w \in \Sigma^*$, so heißt u Infix von w, wenn

$$\exists v_1, v_2 \in \Sigma^* : w = v_1 u v_2$$

2.1.3 Produktautomat

Gegeben seien zwei NEA's $a_i = (Q_i, \Sigma, q_i, \Delta_i, F_i)$ für $i = 1, 2$, so ist der Produktautomat $a_1 X a_2$ gegeben durch:

$$(Q_1 \times Q_2, \Sigma, (q_1, q_2), \Delta, F) \text{ mit}$$
$$((p, r), a, (p', r')) \in \Delta \iff (p, a, p') \in \Delta_1 \text{ und } (r, a, r') \in \Delta_2$$
$$F = F_1 \times F_2$$

2.1.4 NEA mit Worttransitionen

Ein NEA mit Worttransitionen hat die Form $a = (Q^{\text{endl.}}, \Sigma^{\text{endl.}}, q_0, \Delta, F)$ mit $\Delta \subset Q \times \Sigma^* \times Q$ endlich.

Sonderfall: ε-NEA

ε-NEA mit $\Delta \subset Q \times (\Sigma \cup \{\varsigma\}) \times Q$

2.1.5 Deterministisch endlicher Automat (DEA)

Ein NEA $a = (Q, \Sigma, q_0, \Delta, F)$ heißt deterministisch endlicher Automat (DEA), falls zu jedem $(q, a) \in Q \times \Sigma$ genau ein q' mit $(q, a, q') \in \Delta$ existiert.

Dann ist Δ definiert durch $\delta : Q \times \Sigma \rightarrow Q$ mit

$$\delta(q, a) = q' \iff (q, a, q') \in Delta$$

2.1.6 Fortsetzung

Definition der Fortsetzung von δ mit $\delta^* : Q \times \Sigma^* \rightarrow Q$ wird definiert durch

$$\delta^*(q, \varepsilon) = q\delta^*(q, wa) = \delta(\delta^*(q, w), a) \text{ für } w \in \Sigma^*, a \in \Sigma$$

2.1.7 $\delta^*(p, w) = q \iff a : p \rightarrow^w q$

Beweis

Durch Induktion über den Aufbau der Wörter über Σ:

$\underline{w = \varepsilon}$:

$$\delta^*(p, \varepsilon) = q \iff p = q \iff a : p \rightarrow^\varepsilon q$$

$\underline{w = va \in \varepsilon}$:

$$\begin{aligned}\delta^*(p, \varepsilon) = q &\iff^{Def.\delta^*} \delta(\delta^*(p, v), a) = q \\ &\iff \exists p' : \delta^*(p, v) = p' \wedge \delta(p', a) = q \\ &\iff \exists p' : a : p \rightarrow^v p' \wedge p' \rightarrow^q \\ &\iff^{IV} p \rightarrow^{va} Q\end{aligned}$$

Bemerkung b

$L(a) = \{w \in \Sigma^* \mid \delta^*(q_0, w) \in F\}$

Schreibweise

$\delta(q, w)$ statt $\delta^*(q, w)$

2.1.8 Regulär

L ist von NEA / DEA erkennbar (L ist regulär).

2.1.9 Lemma: Ist $L \subset \Sigma^*$ von DEA erkennbar, so auch $\Sigma \backslash L$

Beweis:

Sei $a = (Q, \Sigma, q_0, \delta, F)$ ein DEA der L erkennt, so gilt:

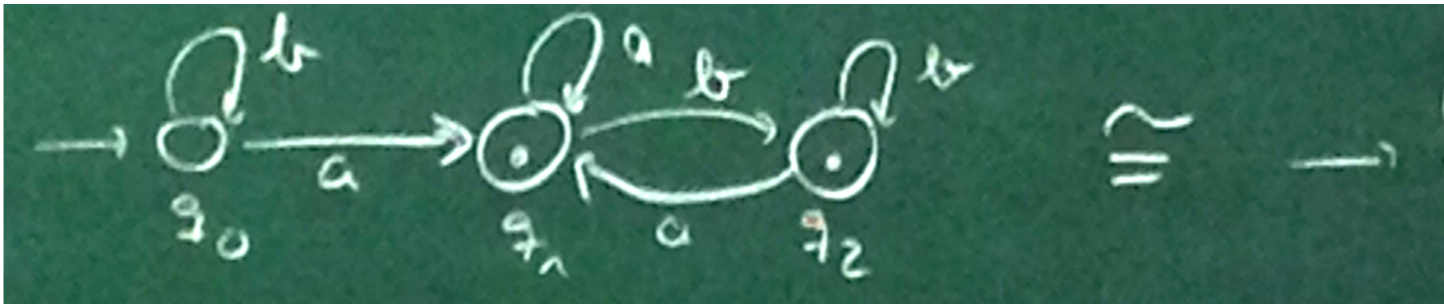
$$\begin{aligned} w \in \Sigma^* \backslash L &\iff \delta(q_0, w) \in Q \backslash L \\ &\iff a' = (Q, \Sigma, q_0, \delta, Q \backslash F) \text{ akzeptiert w.} \end{aligned}$$

Also erkennt a' $\Sigma^* \backslash L$.

Folgerung:

Die durch DEA's erkennbare Sprachen über Σ bilden eine boolsche Algebra (d.h abgeschlossen unter Vereinigung, Schnitt und Komplement).

2.1.10 Beschränkung auf erreichbare Zuständen



Definition

Ein Zustand q heißt erreichbar, falls $\exists w \in \Sigma^* : a : q_0 \rightarrow^w q$

2.1.11 Äquivalenzklassen

Die Äquivalenzklassen DEA $\overline{a}$ zu a sei definiert

$$\begin{aligned}
\overline{a} &= (\overline{Q}, \Sigma, \overline{q_0}, \overline{\delta}, \overline{F}) \\
\overline{Q} &= \{\overline{q} \mid q \in Q\} \\
\overline{\delta}(\overline{q}, a) &= \overline{\delta(q, a)} \\
\overline{F} &= \{\overline{q} \mid q \in F\}
\end{aligned}$$

Bemerkungen

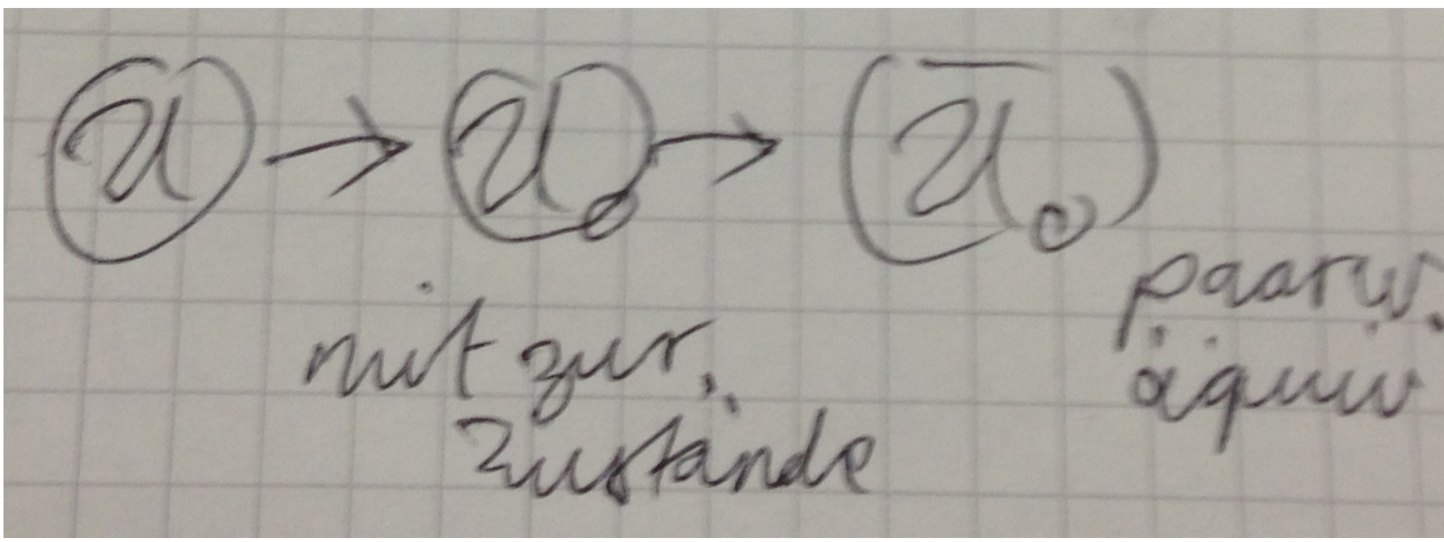
3. $\overline{\delta}(\overline{q}, w) = \overline{\delta(q, w)}$
4. $\overline{a}$ ist äquivalent zu a

2.1.12 Reduzierter DEA

Ein DEA, der nur erreichbare und paarweise nicht-äquivalente Zustände hat heißt reduziert.

Zu jedem DEA existiert ein äquivalenter reduzierter

Der DEA a ist ein Automat mit minimaler Angabe von Zuständen (der L(a) erkennt)



CHAPTER

THREE

GLIEDERUNG DER VORLESUNGEN

1. Endliche Automaten
2. Grammatiken
3. Berechenbarkeit
4. Entscheidbarkeit
5. Komplexität