Proseminar Automatentheorie

Thema: Berry – Sethi Algorithmus

Von Christopher Haas

Inhaltsangabe

- Allgemeines/Notation
- Ableitung regulärer Ausdruck
  - Intuitiv
  - Definition
  - Beispiele
- Das Brzozowski-Verfahren
- Markierungsverfahren (McNaughton und Yamada)
- Der Berry-Sethi Algorithmus
- Quelle

Allgemeines/Notation

- Die Syntax eines regulären Ausdruckes über einer Menge $\sum$ (wobei a ein Symbol daraus ist) sei wie folgt definiert:

- E ::= 0 (entspricht leerer Menge $\varnothing$) | 1 (entspricht ε) | a | (E + E) | (E · E) | (E)*.

- Wir bezeichnen L(E) als die durch den regulären Ausdruck beschriebene Sprache.

- L(0) entspricht leerer Menge, L(1) entspricht Menge mit nur einem Element ε. L(E + F) entspricht der Vereinigung von L(E) und L(F), L(E · F) der Konkatenation und L(E*) dem Kleene-Star.

Allgemeines/Notation

- Wir benötigen folgende Notation:

  Es gilt $\delta(E) = 1$ falls $L(E)$ den leeren String (ε) enthält, ansonsten $\delta(E) = 0$.

  $\delta(0) = 0$,

  $\delta(1) = 1$,

  $\delta(a) = 0$,

  $\delta(E + F) = \delta(E) + \delta(F)$,

  $\delta(E \cdot F) = \delta(E) \cdot \delta(E)$,

  $\delta(E^*) = 1$.

- Stelle fest: $\delta(E) \cdot F = F$ falls leerer String in $L(E)$, ansonsten $\delta(E) \cdot F = 0$.

Ableitung regulärer Ausdruck: Intuitiv

- Gegeben regulärer Ausdruck E und Symbol a.

  Die Ableitung eines regulären Ausdruckes E nach Symbol a "berechnet" einen neuen regulären Ausdruck E'. E' entspricht gerade dem regulären Ausdruck: E ohne a am Anfang.

- Beispiele:

  $a^{-1}(abb) = bb$

  $b^{-1}(abb) = 0$ (d.h. : $\emptyset$)

  $a^{-1}(aba + ab) = ba + b$

  $a^{-1}(aba)^* = ba(aba)^*$

  $a^{-1}(ab + b)^*ba = b(ab + b)^*ba$

Ableitung regulärer Ausdruck: Definition

- Gegeben ein regulärer Ausdruck E und ein Symbol a.

  Die Ableitung von E nach a

  $a^{-1} E$

  kann wie folgt rekursiv berechnet werden:

  $a^{-1} 1 = 0, \qquad a^{-1} 0 = 0,$

  $a^{-1} a = 1, \qquad a^{-1} b = 0 \quad \text{für } b \neq a,$

  $a^{-1} (E + F) = a^{-1} E + a^{-1} F,$

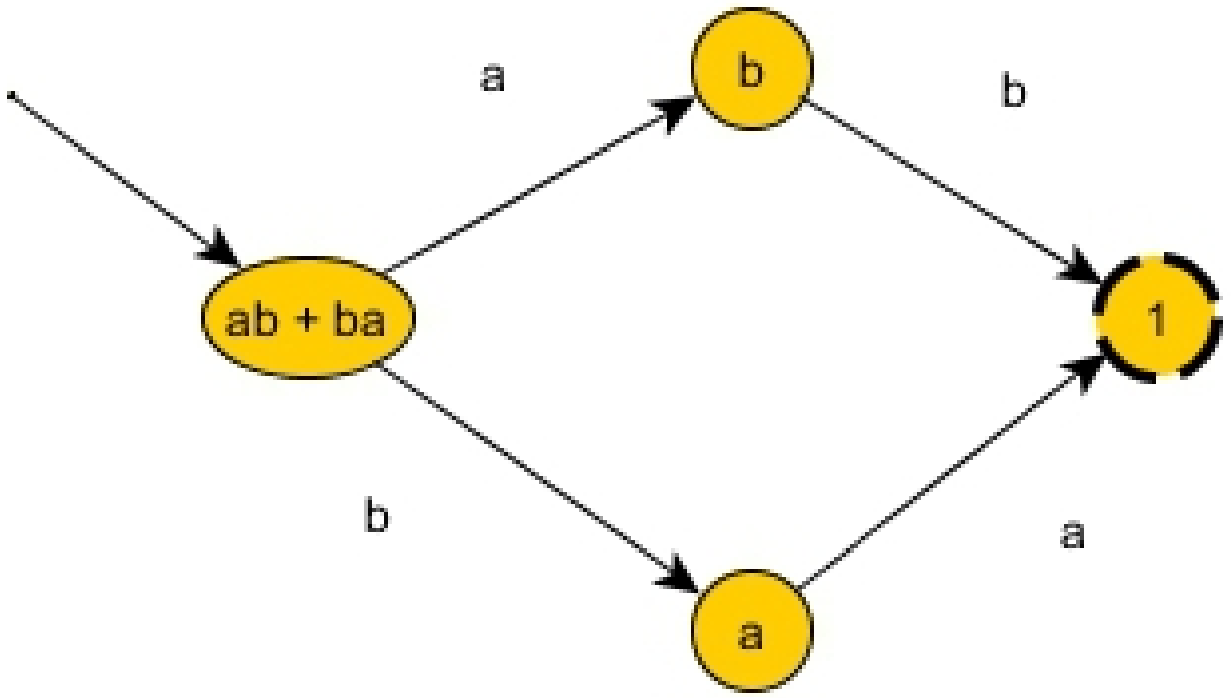
  $a^{-1} (E \cdot F) = a^{-1} E \cdot F + \delta(E) \cdot a^{-1} F,$

  $a^{-1} (E^*) = a^{-1} E \cdot E^*.$

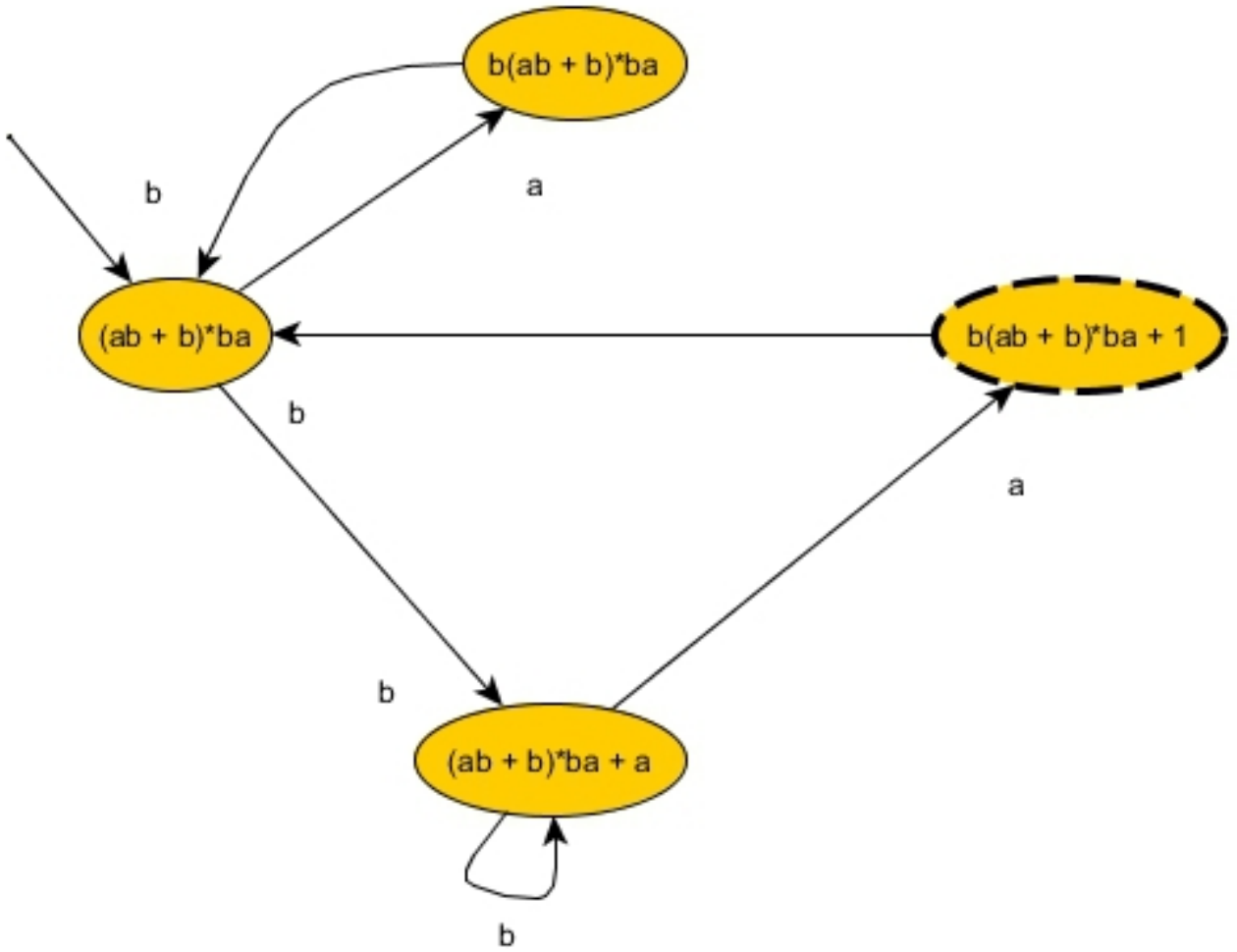
- Die Erweiterung der Ableitung von (einzelnen) Symbolen hin zu ganzen Strings w definieren wir durch:

  $\varepsilon^{-1} E = E, \qquad (wa)^{-1} E = a^{-1} (w^{-1} E).$

Bsp. 1: ab + ba



Bsp. 2: (ab + b)*ba



Das Brzozowski Verfahren

- Konstruiere einen deterministischen Automaten für einen regulären Ausdruck E wie folgt:
  - Konstruiere Startzustand. Dieser entspricht E.
  - Berechne vom Startzustand sukzessive alle Folgeableitungen bis keine neuen Ableitungen mehr existieren. Konstruiere für jeden eindeutigen Ableitungsausdruck einen Zustand.
  - Konstruiere Übergänge von Zustand p nach Zustand q durch a genau dann, wenn a^{-1} p abgeleitet q ergibt.
  - Endzustände sind alle Zustände, deren durch ihren regulären Ausdruck beschriebene Sprache den leeren String enthält (also für die gilt: $\delta(E) = 1$, wobei E der reguläre Ausdruck im Zustand ist).

Das Brzozowski Verfahren

Konstruiere Übergänge von Zustand p nach Zustand q durch a genau dann, wenn $a^{-1}p$ abgeleitet q ergibt.

- Problem: Punkt 3 benötigt Äquivalenztest auf regulären Ausdrücken.
- Extrem teuer und fällt in Klasse "PSPACE-vollständige Probleme" .

  => vermutlich exponentiell
- Dies ist gerade die Motivation für den Berry – Sethi Algorithmus.

Markierungsverfahren (McNaughton und Yamada)

- Betrachte regulären Ausdruck E, z.B.

  $E = (a + ba)^*ab$

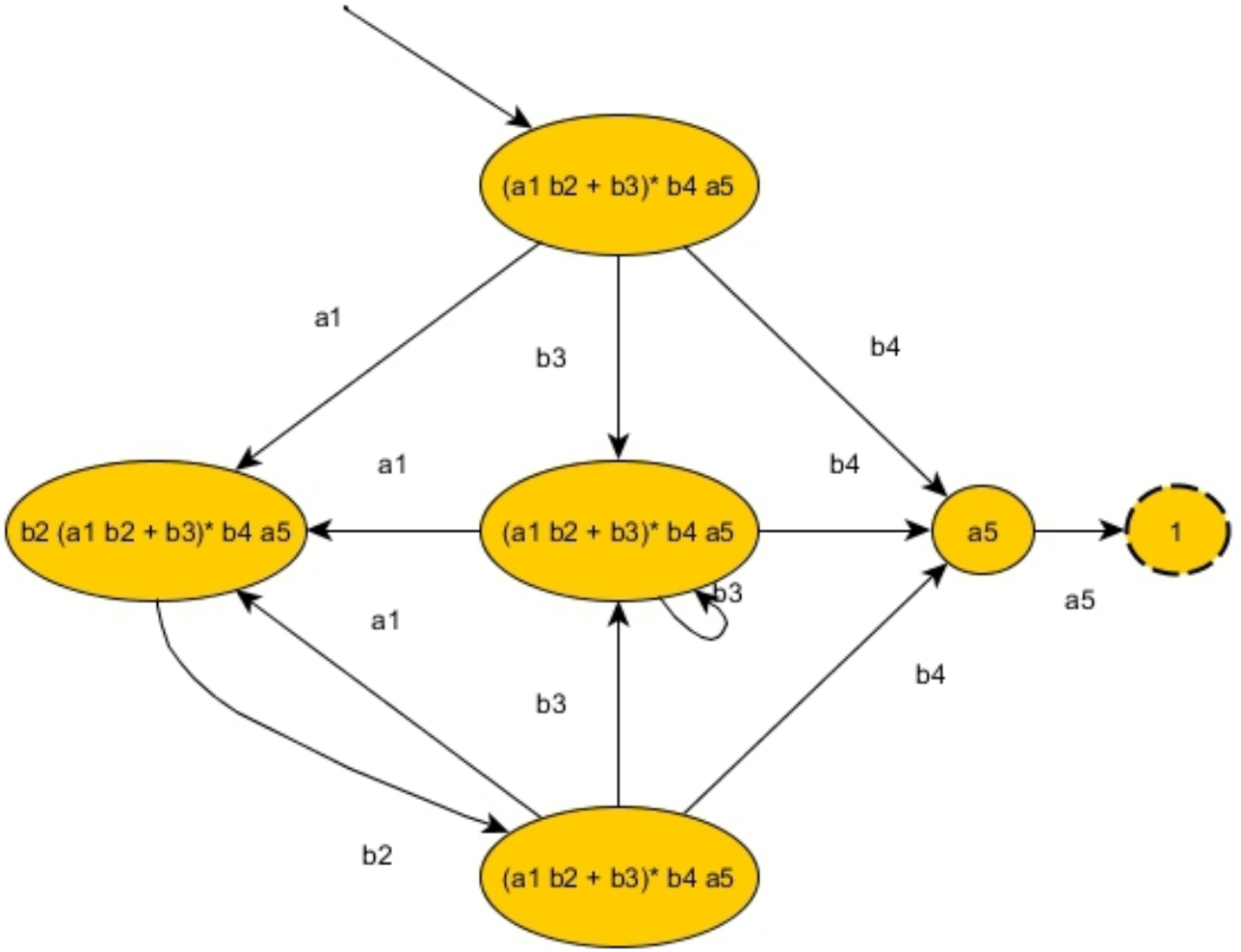
- "Markiere" E, das heißt: alle Eingabesymbole werden eindeutig.

  Im Beispiel:

  $E_{markiert} = (a_1 + b_2a_3)^*a_4b_5$.

- a_1, a_3, a_4 werden also als unterschiedliche Symbole behandelt.

Bsp. 3: $(a_1 b_2 + b_3)^* b_4 a_5$



Der Berry-Sethi Algorithmus

- Definition: Seien alle Symbole in E eindeutig. Für jedes Symbol a aus E definieren wir eine Fortsetzung von a in E als irgendeinen regulären Ausdruck der Form $wa^{-1}E \neq 0$.
- Konstruktion eines deterministischen Automaten M zu gegebenem regulärem Ausdruck E:
  - Markiere E, erhalte regulären Ausdruck E' mit nur eindeutigen Symbolen.
  - M hat Startzustand (entspricht E') und genau so viele zusätzliche Zustände wie markierte Symbole in E'.
  - Konstruiere Übergänge über a von p zu der Fortsetzung von a genau dann, wenn p für eine Forsetzung S ergibt und dieses S Strings mit a an erster Stelle beschreibt.
  - Endzustände sind genau alle Zustände für die gilt: $\delta(S) = 1$, wobei S wieder Fortsetzung von E ist.

Der Berry-Sethi Algorithmus

- Stelle fest: es genügt für einen regulären Ausdruck mit n markierten Symbolen einen Automat mit n+1 Zuständen zu erstellen.

- Betrachte die Fortsetzung von a in E.

- Behauptung: So ein (Fortsetzungs-)Ausdruck muss immer existieren und alle solche Ausdrücke sind äquivalent. Die Fortsetzung von a in E beschreibt also die Äquivalenzklasse eines Ausdrucks.

- Können wir das beweisen, dann ist kein Äquivalenztest notwendig, was sich deutlich auf die Performance auswirkt.

Der Berry-Sethi Algorithmus

- Satz:

  Seien alle Symbole in E eindeutig. Betrachte (beliebiges) Symbol a aus E.

  Für jeden beliebigen String w ist $(wa)^{-1}$ E entweder 0 oder eindeutig modulo Assoziativität, Kommutativität, Idempotenz mit +.

- Beweis (per struktureller Induktion über E):
  - Falls E = 0 oder E = 1 sind (nach Definition von Ableitungen) alle Ableitungen gleich 0.
  - Falls E = a (a beliebiges Symbol) sind die Ableitungen gleich 1 (nach a abgeleitet) oder 0 (sonst).

Der Berry-Sethi Algorithmus

- Satz:

  Seien alle Symbole in E eindeutig. Betrachte (beliebiges) Symbol a aus E.

  Für jeden beliebigen String w ist $(wa)^{-1}$ E entweder 0 oder eindeutig modulo Assoziativität, Kommutativität, Idempotenz mit +.

- Beweis (per struktureller Induktion über E):
  - Fall 1: $E = E_1 + E_2$:

    $(wa)^{-1}(E_1 + E_2) = (wa)^{-1} E_1 + (wa)^{-1} E_2$ (Def. Von Ableitungen)

    $a \in E_1$: $(wa)^{-1}(E_1 + E_2) = (wa)^{-1} E_1 + 0$

    $a \in E_2$: $(wa)^{-1}(E_1 + E_2) = 0 + (wa)^{-1} E_2$

  - Analog für die Fälle $E = E_1 \cdot E_2$ und $E = E_1^*$.

Der Berry-Sethi Algorithmus

- Satz:

  Seien alle Symbole in E eindeutig. Betrachte (beliebiges) Symbol a aus E.

  Für jeden beliebigen String w ist $(wa)^{-1}$ E entweder 0 oder eindeutig modulo Assoziativität, Kommutativität, Idempotenz mit +.

- Daraus folgt also: eine Fortsetzung eines Symbols muss immer existieren (Beweis ebenfalls durch strukturelle Induktion über E) und insbesondere sind alle erhaltenen Ausdrücke äquivalent.

  => Kein Äquivalenztest notwendig!

Der Berry-Sethi Algorithmus

- Das komplette Verfahren zur Umwandlung eines regulären Ausdruckes E in einen deterministischen Automaten D lautet dann wie folgt:
  - Markiere E und erhalte $E_{markiert}$.
  - Konstruiere mit eben besprochenem Verfahren (deterministischen) Automaten M', welcher $L(E_{markiert})$ akzeptiert.
  - Innerhalb dieses Automaten wird Markierung der Symbole rückgängig gemacht und dadurch M konstruiert. M kann durch Demarkierung nicht-deterministisch werden.
  - Potenzmengenkonstruktion von M ergibt dann widerum deterministischen Automaten D, welcher E akzeptiert.

E ----------> E' -----> M' ----------> M ----------> D

Markierung Demarkierung

Quelle

- *"FROM REGULAR EXPRESSIONS TO DETERMINISTIC AUTOMATA"*

  von

  Gerard Berry und Ravi Sethi

  erschienen in

  Theoretical Computer Science 48 (1986) 117-126, North Holland

  http://www.sciencedirect.com/science/article/pii/0304397586900885/pdf?md5=cd17dad7971b346f64aa5ce38b0cabdc&pid=1-s2.0-0304397586900885-main.pdf