Über die Operation Fortsetzung bei formalen Sprachen

Robert Hartmann

24. September 2010

Inhaltsverzeichnis

1 Einleitung

In dieser Arbeit untersuchen wir die Operation Fortsetzung bei formalen Sprachen. Diese Operation wird in der Arbeit [St87] eingeführt.

Wir bezeichnen die Menge X^* als Menge aller endlichen Wörter über dem Alphabet X.
Wir bezeichnen weiterhin die Menge X^ω als Menge aller unendlichen Wörter über dem Alphabet X.
Sei ferner die Präfixrelation $\sqsubseteq$ wie üblich definiert:

Definition 1.

$$w \sqsubseteq b \Leftrightarrow w \cdot b' = b,\ \textit{für ein } b' \in X^*$$

$$\textit{pref}(L) = \{v : v \sqsubseteq w \wedge w \in L\}$$

Es wird nun der δ-Limes einer Wortmenge W^δ definiert (s. [St87, Seite X])

Definition 2.

$$W^\delta = \{\beta : \beta \in X^\omega \ \textit{und pref}(\beta) \cap W \ \textit{ist unendlich}\}$$

Die folgende Eigenschaft (13) aus [St87] ist leicht einzusehen:

$$(U \cup W)^\delta = U^\delta \cup W^\delta$$

Definition 3. *Eine Sprache nennen wir eine (σ, δ)-Teilmenge von X^* genau dann, wenn für alle $\beta \in X^\omega$ entweder $\textit{pref}(\beta) \cap W$ oder $\textit{pref}(\beta) \backslash W$ endlich ist.*

Beispiele für (σ, δ)-Teilmengen sind alle endlichen Sprachen und deren Komplemente. Weitere Beispiele sind Sprachen der Form $\textit{pref}(U)$ oder $W \cdot X^*$.
Eine Eigenschaft für diese Teilmengen ergibt sich wiefolgt :

Satz 1 (St87)**.** *Sei U eine $(\sigma, \delta) -$ Teilmenge von X^*, dann gilt:*

$$(U \cap W)^\delta = U^\delta \cap W^\delta, \qquad \textit{für alle } W \subseteq X^*$$

Nun wird die Operation “Fortsetzung“ wie in [St87] eingeführt, im nachfolgenden als $\triangleright$ bezeichnet. Die Fortsetzung eines Wortes w in eine Sprache $V \subseteq X^*$ sei definiert als:

Definition 4.

$$w \triangleright V := \textit{Min}_{\sqsubseteq}\{v : v \in V \wedge w \sqsubseteq v\} = \textit{Min}\,(w \cdot X^* \cap V)$$

Diese Operation wird wie folgt auf Sprachen ausgedehnt, dabei bezeichnen wir die Fortsetzung einer Sprache W in eine Sprache $V \subseteq X^*$ mit:

Definition 5.

$$W \triangleright V := \bigcup_{w \in W} w \triangleright V$$

Diese Operation hat nun folgende Eigenschaft bezüglich des δ-Limes: Während

$$(W \cap U)^\delta = W^\delta \cap U^\delta$$

nur für (σ, δ)-Teilemgen gilt, so gilt

$$(W \triangleright U)^\delta = W^\delta \cap U^\delta$$

für sämtliche Sprachen
Daher wird nun im Verlauf der Arbeit die Operation Fortsetzung untersucht.

2 Eigenschaften

2.1 Allgemeines

In diesem Abschnitt betrachten wir Eigenschaften der Operation der Fortsetzung von Sprachen. Hierbei werden insbesondere die Stabilität bzw. Monotonie bezüglich mengentheoretischen Operationen betrachtet.

2.2 grundlegende Beziehungen

Aus der Definition folgen direkt die Eigenschaften:

Eigenschaft 1. $\{w\} \triangleright L = \{w\}$, *wenn* $w \in L$
$W \cap L \subseteq W \triangleright L \subseteq L$

Eigenschaft 2. $(W \cup V) \triangleright L = W \triangleright L \cup V \triangleright L$

Aus Eigenschaft 1 wissen wir, dass $W \cap L \subseteq W \triangleright L$. Man kann nun die Fortsetzung von W in L aufsplitten, sodass sich folgende Beziehung ergibt:

Eigenschaft 3. $W \triangleright L = W \cap L \cup (W \backslash L) \triangleright L$

Aus der Eigenschaft 3 folgt durch Einschränkung der Mengen W bzw. L:

Folgerung 1. $W \subseteq L \rightarrow W \triangleright L = W \cap L = W$

Folgerung 2. $W \supseteq L \rightarrow W \triangleright L = W \cap L = L$

Aus Eigenschaft 2 folgt:

Eigenschaft 4. $W \triangleright L \rightarrow W \triangleright V \subseteq L \triangleright V$

Damit haben wir eine

Gleichung 2.1. $(L \cap U) \triangleright V \subseteq (L \triangleright V) \cap (U \triangleright V)$

Dabei muss nicht notwendig die Gleichheit gelten, wie das folgende Beispiel zeigt

Beispiel 1. *Es seien* $L = \{aa, bb\}, U = \{aa, b\}, V = \{aa, bb\}$.
Dann haben wir $(L \cap U) \triangleright V = \{aa\} \subset \{aa, bb\} = (L \triangleright V) \cap (U \triangleright V)$

Gleichung 2.2. $L \triangleright (U \cup V) \subseteq (L \triangleright U) \cup (L \triangleright V)$

Zum Beweis bemühen wir die Beziehung aus

Lemma 1. $Min\ (A \cup B) \subseteq Min\ A \cup Min\ B$

Beweis. Es genügt, die Eigenschaft für $L = \{w\}$ zu zeigen.
Es gilt $w \triangleright (W \cup V) = Min\ (w \cdot X^* \cap (W \cup V)) = Min\ ((w \cdot X^* \cap W) \cup (w \cdot X^* \cap V))$ und mit Lemma 1 erhalten wir:
$w \triangleright (W \cup V) \subseteq Min\ (w \cdot X^* \cap W) \cup Min\ (w \cdot X^* \cap V) = (L \triangleright W) \cup (L \triangleright V)$ $\square$

Dabei muss nicht notwendig die Gleichheit gelten, wie das folgende Beispiel zeigt.

Beispiel 2. *Es seien* $L = \{a, b\}$, $U = \{aaa\}$, $V = \{bb, aa\}$.
Dann haben wir $L \triangleright (U \cup V) = \{aa, bb\} \subset \{aa, bb, aaa\} = (L \triangleright U) \cup (L \triangleright V)$

Eine ähnliche Beziehung ergibt sich für die Operation $\cup$

Gleichung 2.3. $L \triangleright (U \cap V) \supseteq (L \triangleright U) \cap (L \triangleright V)$

Zum Beweis bemühen wir die Beziehung aus

Lemma 2. $Min\ (A \cap B) \supseteq Min\ A \cap Min\ B$

Beweis. Es genügt die Eigenschaft für $L = \{w\}$ zu zeigen.
Es gilt $w \triangleright (U \cap V) = Min\ (w \cdot X^* \cap (U \cap V)) = Min\ ((w \cdot X^* \cap U) \cap (w \cdot X^* \cap V))$
und mit Lemma 2 erhalten wir
$w \triangleright (U \cap V) \supseteq Min\ (w \cdot X^* \cap U) \cap Min\ (w \cdot X^* \cap V)$
$= (L \triangleright U) \cap (L \triangleright V)$ $\square$

Dabei muss nicht notwendig die Gleichheit gelten, wie das folgende Beispiel zeigt

Beispiel 3. *Es seien* $L = \{a, b\}$, $U = \{aaa, b, bb\}$, $V = \{bb, aaa\}$
Dann haben wir $L \triangleright (U \cap V) = \{bb, aaa\} \supset \{aaa\} = (L \triangleright U) \cap (L \triangleright V)$

2.3 Konkatenation

Bisher wurden hauptsächlich die Operationen $\cap$ sowie $\cup$ in Verbindung mit $\triangleright$ betrachtet. Für die Konkatenation ergeben sich keine Eigenschaften allgemeingültiger Natur.

So gibt es Sprachen, bei denen die Gleichheit in folgender Weise gegeben ist:

Beispiel 4. *Es seien* $L = \{e\} \quad , U = \{a\} \quad , V = \{b\}$*, so erhalten wir*
$L \triangleright (U \cdot V)\{ab\} = \{ab\} = (L \triangleright U) \cdot (L \triangleright V)$*, also* $L \triangleright (U \cdot V) = (L \triangleright U) \cdot (L \triangleright V)$

weiterhin können wir Sprachen angeben, sodass Teilmengenbeziehung in folgender Weise existieren:

Beispiel 5. *Es seien* $L = \{a\} \quad , U = \{aa\} \quad , V = \{b, a\}$
Dann erhalten wir $L \triangleright (U \cdot V) = \{aab, aaa\} \supset \{aaa\} = (L \triangleright U) \cdot (L \triangleright V)$*, also* $L \triangleright (U \cdot V) \supset (L \triangleright U) \cdot (L \triangleright V)$*.*

Das letzte Beispiel in diesem Abschnitt zeigt deutlich, dass keine allgemeingültigen Eigenschaften für die Operation $\cdot$ bezüglich $\triangleright$ existieren:

Beispiel 6. *Es seien* $L = \{aab, a\} \quad U = \{aa\} \quad V = \{b\}$
Dann erhalten wir $L \triangleright (U \cdot V) = \{aab\} \neq \{aa\} = (L \triangleright U) \cdot (L \triangleright V)$*, das bedeutet also, dass beide Seiten mengentheoretisch unvergleichbar sind, also* $L \triangleright (U \cdot V) \not\supset (L \triangleright U) \cdot (L \triangleright V)$*, sowie* $L \triangleright (U \cdot V) \not\subset (L \triangleright U) \cdot (L \triangleright V)$

Betrachtet man nun die Konkatenation ‘vorn‘, also die Beziehung $(L \cdot U) \triangleright V$ zu $(L \triangleright V) \cdot (U \triangleright V)$, so sieht man leicht, dass es sich auf der linken Seite der Gleichung um Wörter aus V handelt die man vergleicht mit Wörtern aus V^2.
Demnach treten hier Eigenschaften nur auf wenn V eine ganz spezielle Gestalt hat.

2.4 Gilt nicht...

Folgt direkt aus den Gleichungen in 2.1

$$L \triangleright (U \cup V) \supset (L \triangleright U) \cup (L \triangleright V)$$

$$L \triangleright (U \cap V) \subset (L \triangleright U) \cap (L \triangleright V)$$

$$L \triangleright (U \cdot V) \subset (L \triangleright U) \cdot (L \triangleright V)$$

$$(L \cap U) \triangleright V \supset (L \triangleright V) \cap (U \triangleright V)$$

3 Eigenschaften bei Sprachen spezieller Gestalt

Lemma 3. *Sei* $V \subseteq X^*\backslash W \cdot X^*$*, so gilt* $V \triangleright W \cdot X^* = V \triangleright Min\ (W)$

zu zeigen:

1. $V \triangleright W \cdot X^* \subseteq V \triangleright Min\ (W)$
2. $V \triangleright W \cdot X^* \supseteq V \triangleright Min\ (W)$

Beweis. :
zu 1. Die Inklusion $\supseteq$ folg aus $W \cdot X^* \supseteq Min\ (W)$.
zu 2. Zum Beweis der der anderen Inklusion genügt es, diese für den Fall $V = \{v\}$ zu zeigen.
Es sei nun $w \in v \triangleright W \cdot X^*$, wobei nach Vorraussetzung $v \notin W \cdot X^*$ gelte. Dann gilt für kein u, $v \sqsubseteq u \sqsubset w$ oder $u \sqsubseteq v$ die Beziehung $u \in W \cdot X^*$.
Also haben wir $w \in Min\ (W \cdot X^*) = Min\ (W)$ □

Eigenschaft 5. $pref(V) \triangleright W$

Beweis. 1. Fall: $w \in W \cap pref(V) \to w \in pref(V) \triangleright W$
2. Fall: $w \in W\backslash pref(V) \to w \in V \triangleright Min(W)$
$\to pref(V) \triangleright W = (pref(V) \cap W) \cup (pref(V) \triangleright Min\ (W))$

□

Eigenschaft 6. $W \triangleright pref(V) = W \cap pref(V)$

Beweis. Es genügt die Eigenschaft für $W = \{w\}$ zu zeigen:
Aus der Definition wissen wir, dass $w \triangleright pref(V) = Min\ (\{w\} \cdot X^* \cap pref(V))$ entspricht. Sei $w' \in (\{w\} \cdot X^* \cap pref(V))$, so sieht man leicht, dass $w \sqsubseteq w'$ und demnach $w \in pref(V)$ gelten muss. Daraus folgt sofort $Min\ (\{w\} \cdot X^* \cap pref(V)) = \{w\} \cap pref(V)$

□

Eigenschaft 7. $V \cdot X^* \triangleright W = V \cdot X^* \cap W$

Beweis. Es genügt die Eigenschaft für ein $v \in V \cdot X^*$ zu zeigen:
Aus der Definition wissen wir, dass $v \triangleright W = Min\ (\{v\} \cdot X^* \cap W)$ entspricht. Da laut Voraussetzung $v \in V \cdot X^*$, so gilt $= Min\ (\{v\} \cdot X^* \cap W) = Min\ (\{v\} \cap W)$. Da $\{v\} \cap W$ in jedem Fall einelementig ist, wissen wir, dass $Min\ (\{v\} \cap W) = \{v\} \cap W$ gilt.

□

Eigenschaft 8. $W \triangleright V \cdot X^* = (W \triangleright Min\ (V)) \cup (W \cap V \cdot X^*)$

Beweis. W lässt sich in 2 Teile aufspliten: $W = (W \cap V \cdot X^*) \cup (W\backslash V \cdot X^*)$ Nun betrachten wir folgende 2 Fälle:

Fall a) $w \in W \cap V \cdot X^*$
Fall b) $w \in W \backslash V \cdot X^*$

zu a): $w \triangleright V \cdot X^* \rightarrow w \in (W \cap V \cdot X^*)$
zu b): Es gilt nach Vorraussetzung $\{w\} \subseteq X^* \backslash V \cdot X^*$ und mit Hilfe von Lemma 3 erhalten wir $\{w\} \triangleright V \cdot X^* = \{w\} \triangleright Min\ (V)$
Daraus folgt:

$$W \triangleright V \cdot X^* = (W \triangleright Min\ (V)) \cup (W \cap V \cdot X^*)$$

$\square$

4 Abgeschlossenheit in der CHOMSKY-Hierachie

4.1 Regularität

Seien L und W regulär, so ist auch $L \triangleright W$ regulär.

Automat $A_L = (X, Z, z_0, \delta_L, Z_f)$ akzeptiere L, Automat $A_W = (X, S, s_0, f, S_f)$ akzeptiere W. Automat A akzeptiert $L \triangleright W$,

Vorgehensweise:
A_L und A_W lesen das Wort w parallel. Falls A_L akzeptiert und wählt A nicht-deterministisch aus ob Schritt 2 aktiviert wird oder nicht.
Schritt 2: A_W liest das Wort w zu Ende, während A_L im Zustand z'_f verweilt. Sollte A_W auf diesem mehr als einmal akzeptieren, so akzeptiert A nicht indem A_W im Stoppzustand s_x stehen bleibt, ansonsten akzeptiert A.

$A = (X, Z \cup \{z'_f\} \times S \cup \{s_x\}, (z_0, s_0), \delta, \{(z'_f, s') : s' \in S_f\}), s_x \notin S$ mit

$\delta = \{((z_i, s_i), x, (z_j, s_j)) : (z_i, x, z_j) \in \delta_L \wedge f(s_i, x) = s_j\} \cup$
$\{((z_i, s_i), x, (z'_f, s_j)) : (z_i, x, z') \in \delta_L \wedge z' \in Z_f \wedge f(s_i, x) = s_j\} \cup$
$\{((z'_f, s_i), x, (z'_f, s_j)) : f(s_i, x) = s_j \wedge s_i \notin S_f\} \cup$
$\{((z'_f, s_i), x, (z'_f, s_x)) : f(s_i, x) = s_j \wedge s_i \in S_f\}$

Beweis. Der konstruierte Automat A akzeptiert nur in einem Zustand $(z'_f, s'), s' \in S_f$ Nach Konstruktion gelangt A bei Eingabe w genau dann in (z'_f, s), wenn ein Wort $l \in L$ mit $l \sqsubseteq w$ existiert. In solch einem Fall kann der Automat umschalten. Wenn dies der Fall ist, so arbeitet A weiter auf der Eingabe w wie A_W es tut. Sollte A_W nun akzeptieren und w ist noch nicht zu Ende gelesen, so wird A nach Konstruktion in einen Stoppzustand (z'_f, s_x) geleitet, in dem er nie wieder akzeptiert. A akzeptiert also nur wenn A_L akzeptiert hat (es existiert ein $l \in L \wedge l \sqsubseteq w$) und wenn für alle v' mit $l \sqsubseteq v' \sqsubset w$ gilt $v' \notin W$.

Demnach akzeptiert A die Eingabe w genau dann, wenn $w \in L \triangleright W$ □

4.2 Kontextfreiheit

4.2.1 deterministisch kontextfrei

Es existieren deterministisch kontextfreie Sprachen L, W, sodass $L \triangleright W$ nicht deterministisch kontextfrei ist!

Sei $L = \{a^n b^n c^i : i, n > 0\}$ und $W = \{a^i b^n c^n : i, n > 0\}$. Sowohl L als auch W sind deterministische, kontextfreie und auch lineare Sprachen, da es je einen deterministischen Kellerautomaten gibt, der L sowie W akzeptiert und eine es lineare Grammatiken G_L

und G_W gibt, sodass $L(G_L) = L$ und $L(G_W) = W$ gilt.
Betrachtet man sich nun ein Wort $u \in L \triangleright W$ so muss u laut Definition folgende Struktur besitzen $u \in Min\ (l \cdot X^* \cap W)$ *für ein* $l \in L$. Damit sieht man leicht, dass $L \triangleright W = \{a^n b^n c^n : n > 0\}$ und $\{a^n b^n c^n : n > 0\}$ ist bekanntlich nicht kontextfrei, also auch nicht deterministisch kontextfrei

4.3 Entscheidbarkeit

4.3.1 L und W entscheidbar

Seien L und W (Turing)entscheidbar, so ist auch $L \triangleright W$ entscheidbar.

Seien die Turing Maschinen T_L und T_W.
Die Turing Maschine T entscheidet $L \triangleright W$ nach folgendem Algorithmus:

Algorithm 1 entscheide $L \triangleright W$, Input w

```
if (w ∉ W) then
  T rejects
else
  if (w ∈ L) then
    T accepts
  end if
end if
w' = w
repeat
  w' ← cut(w')
  if (w' ∈ W) then
    T rejects
  end if
  if (w' ∈ L) then
    T accepts
  end if
until (w' == e)
T rejects
```

4.3.2 L akzeptierbar, W entscheidbar

Sei w die Eingabe. Es soll w akzeptiert werden, wenn $w \in L \triangleright W$, wobei L akzeptierbar und nicht entscheidbar und W entscheidbar.
Dazu zählt man $l \in L$ auf und prüft für alle u mit $l \sqsubseteq u \sqsubset w$, ob ein $u \in W$ liegt. Sollte dies der Fall sein, darf man nicht akzeptieren. Dann wird ein weiteres l aufgezählt und der Algorithmus beginnt von vorn.

Algorithm 2 akzeptiere $L \triangleright W$, Input w

```
if (w ∉ W) then
  T rejects
end if
while true do
  zähle ein l ∈ L auf
  if |l| ≤ |w| ∧ l ∈ pref({w}) then
    v := w
    while v ≠ l do
      v := w' mit w' · x = w, x ∈ X
      if v ∈ W then
        break
      end if
    end while
    if v = l then
      T accepts
    end if
  end if
end while
```

4.3.3 L entscheidbar, W akzeptierbar

Sei L entscheidbar und W aufzählbar, so ist $L \triangleright W$ nicht notwendigerweise aufzählbar.

Beweis. Bemerkung: A ist aufzählbar, aber nicht entscheidbar

Sei $W = \{0^{n+1}\ 1^{n+1} : n \in \mathbb{N}\} \cup \{0^{n+1}\ 1 : n \in A\}$, also W ist aufzählbar und sei $L = \{0\}$.
So ist $L \triangleright W = 0 \triangleright W = \{0^{n+1}\ 1^{n+1} : n \in \mathbb{N} \wedge n \notin A\} \cup \{0^{n+1}\ 1 : n \in \mathbb{N} \wedge n \in A\}$. Angenommen $0 \triangleright W$ wäre aufzählbar, so müsste der Schnitt mit einer aufzählbaren Sprache wieder aufzählbar sein. Sei $V = \{0^n\ 1^n : n \in \mathbb{N}\}$ offensichtlich aufzählbar. Dann ergibt sich aber für $0 \triangleright W \cap V = \{0^{n+1}\ 1^{n+1} : n \in \mathbb{N} \wedge n \notin A\}$, was nicht aufzählbar ist. □

5 Schlusswort

6 Quellen und Literatur