Prof. Dr. Hans-Jörg Kreowski
Studiengang Informatik
Universität Bremen
OAS 3001
Tel.: 2956, 3697 (Sekr.), Fax: 4322
E-Mail: kreo@informatik.uni-bremen.de

Mai 2003

Theoretische Informatik 2 (SS 2003)

Wie im Extra-Material versprochen, kann ein endlicher Automat konstruiert werden, der den Durchschnitt zweier Sprachen erkennt, die selbst von endlichen Automaten erkannt werden. Es wird außerdem gezeigt, dass das Leerheitsproblem für endliche Automaten entscheidbar ist.

1 Produktautomat erkennt Durchschnitt

Seien $A_i = (Z_i, I, d_i, s_{0i}, F_i)$ für $i = 1, 2$ zwei deterministische Automaten. Dann ist der *Produktautomat* definiert durch

$$A_1 \times A_2 = (Z_1 \times Z_2, I, d, (s_{01}, s_{02}), F_1 \times F_2)$$

mit $d((s_1, s_2), x) = (d_1(s_1, x), d_2(s_2, x))$ für alle $(s_1, s_2) \in Z_1 \times Z_2$ und $x \in I$.
Der Produktautomat führt also die Zustandsübergänge der beiden Einzelautomaten parallel durch. Wie das folgende Lemma zeigt, gilt das auch für die fortgesetzte Zustandsüberführung. Da der Anfangszustand aus den beiden einzelnen Anfangszuständen und die Endzustände Paare der einzelnen Endzustände sind, ergibt sich aus dem Lemma die gewünschte Durchschnittseigenschaft.

Beobachtung: $L(A_1 \times A_2) = L(A_1) \cap L(A_2)$.

Beweis: $w \in L(A_1 \times A_2)$ gdw. nach folgendem Lemma

$$d^*((s_{01}, s_{02}), w) = (d_1^*(s_{01}, w), d_2^*(s_{02}, w)) \in F_1 \times F_2$$

gdw. $d_i^*(s_{0i}, w) \in F_i$ für $i = 1, 2$ gdw. $w \in L(A_i)$ für $i = 1, 2$.

Lemma: $d^*((s_1, s_2), w) = (d_1^*(s_1, w), d_2^*(s_2, w))$ für alle $(s_1, s_2) \in Z_1 \times Z_2$ und $w \in I^*$.

Beweis: Der Beweis wird mit vollständiger Induktion über den Aufbau von w geführt.
IA: $d^*((s_1, s_2), \lambda) = (s_1, s_2) = (d_1^*(s_1, \lambda), d_2^*(s_2, \lambda))$, wobei die Definition der fortgesetzten Zustandsüberführung verwendet wird.

IS: $d^*((s_1, s_2), wx) = d(d^*((s_1, s_2), w), x) =$
$d((d_1^*(s_1, w), d_2^*(s_2, w)), x) =$
$(d_1(d_1^*(s_1, w), x), d_2(d_2^*(s_2, w), x) =$
$(d_1^*(s_1, wx), d_2^*(s_2, wx))$.
Dabei ergeben sich die Gleichheiten in der gegebenen Reihenfolge aus der Definition der fortgesetzten Zustandsüberführung, der Induktionsvoraussetzung, der Definition der Zustandsüberführung des Produktautomaten und erneut aus der Definition der fortgesetzten Zustandsüberführung.

2 Entscheidbarkeit des Leerheitsproblems

Sei $A = (Z, I, d, s_o, F)$ ein deterministischer Automat. Sei

$$H(A) = \{s \in Z \mid d^*(s, w) \in F \text{für ein} w \in I^*\}$$

die Menge aller Zustände, die zu Endzuständen führen. Dann gilt offensichtlich:

$$L(A) \neq \emptyset \text{ gdw. } s_0 \in H(A).$$

$H(A)$ lässt sich folgermaßen iterieren:
$H_0 = F$ und $H_{i+1} = H_i \cup \{s \in Z \mid d(s, x) \in H_i$ für ein $x \in I\}$ sowie $H(A) = H_m$ für kleinstes m mit $H_{m+1} = H_m$.

Nach Definition gilt: $F = H_0 \subseteq H_1 \subseteq \cdots \subseteq H_i \subseteq H_{i+1} \subseteq \cdots \subseteq Z$. Da Z endlich ist, können nur endlich viele dieser Inklusionen echt sein, so dass es m geben muss. Dann gilt auch $H_{m+k} = H_m$ für alle $k \in \mathbb{N}$ (einfache Induktion über k). Daraus folgt die gewünschte Gleichheit. Denn mit Induktion über i ergibt sich $H_i \subseteq H(A)$:

IA: $H_0 = F \subseteq H(A)$ wegen $d^*(s, \lambda) = s \in F$ f.a. $s \in F$.
IS: $H_{i+1} = H_i \cup \{s \in Z \mid d(s, x) \in H_i$ für ein $x \in I\}$
$\subseteq H(A) \cup \{s \in Z \mid d(s, x) \in H(A)$ für ein $x \in I\}$
$= H(A) \cup \{s \in Z \mid d^*((d)s, x), w)) \in F$ für ein $w \in I^*, x \in I\}$
$= H(A) \cup \{s \in Z \mid d^*(s, xw) \in F$ für ein $xw \in I^*\}$
$\subseteq H(A) \cup H(A) = H(A)$.
Insbesondere gilt $H_m \subseteq H(A)$.

Betrachte umgekehrt $s \in H(A)$. Dann gibt es $w \in I^*$ mit $d^*(s, w) \in F$. Das impliziert $s \in H_{length(w)}$, wie eine einfache Induktion ergibt. Daraus folgt wie gewünscht: $s \in H_{length(w)} \subseteq H_m$.