

Die relationale Äquivalenz von Automaten

Von HERMANN K.-G. WALTER

0. Einleitung

Die relationale Äquivalenz von Automaten ist im Gegensatz zur funktionalen Äquivalenz und der Simulation bisher wenig herangezogen worden, um das Ein-Ausgabeverhalten von Automaten zu vergleichen.

Dies hat wohl zwei Gründe

- Für einige interessante Automatenklassen (streng zusammenhängende, diagnostizierbare Automaten, etc.) kann man zeigen, daß die relationale Äquivalenz gegenüber der funktionalen Äquivalenz nichts Neues bringt.
- Im allgemeinen Fall besteht im Gegensatz zu den beiden anderen Vergleichsmethoden nur ein recht geringer Zusammenhang in der internen Struktur relational äquivalenter Automaten.

In dieser Arbeit behandeln wir folgende Fragestellungen

1. Charakterisierung der relationalen Äquivalenz, insbesondere der von P. HUMMITZSCH [5] eingeführten endlichen relationalen Äquivalenz.
2. Beziehungen zwischen funktionaler, relationaler und endlich relationaler Äquivalenz für spezielle Automatenklassen.

Insbesondere geben wir einen wesentlich einfacheren Beweis für die von P. HUMMITZSCH angegebene Charakterisierung der k -relationalen Äquivalenz von Automaten.

1. Notationen und grundlegende Definitionen

Wir stellen die wichtigsten Definitionen und Notationen zusammen. Nichtdefinierte Begriffe sind der einschlägigen Literatur (z. B. [1], [3] etc.) zu entnehmen.

Ist X eine Menge, so sei X^* das freie Monoid über X mit dem leeren Wort \square . Mit $L(w)$ bezeichnen wir die Wortlänge von $w \in X^*$. $X^+ = X^* - \{\square\}$ ist die freie Halbgruppe über X . Für jedes $k \in \mathbb{Z}_+$ sei $X^k = \{w \in X^* \mid L(w) = k\}$.

Definition 1.1. Ein Automat a ist ein 6-tupel $a = (I(a), O(a), S(a), \delta_a, \lambda_a, \text{In}(a))$, wobei $j(a)$ endliche Mengen ($j = I, O, S, \text{In}$) und $\delta_a: I(a) \times S(a) \rightarrow S(a)$ und $\lambda_a: I(a) \times S(a) \rightarrow O(a)$ Abbildungen sind. Dabei ist $\text{In}(a) \subseteq S(a)$ die Menge der Anfangszustände von a .

Wie üblich ist $I(a)$ die Menge der Eingabesymbole, $O(a)$ die der Ausgabesymbole, $S(a)$ die der Zustände, ferner ist δ_a die Transitionsfunktion und λ_a die Ausgabefunktion. Ist $\text{In}(a) = S(a)$, so ist a nichtinitialisiert.

Gemäß der synchronen und sequentiellen Arbeitsweise von Automaten setzen wir δ_a und λ_a fort auf $I(a)^* \times S(a)$ (vgl. z. B. [3]). Dann ist die von a dargestellte Abbildungsfamilie $\mathfrak{F}(a)$ gegeben durch

$$\mathfrak{F}(a) = \{\lambda_a^s \mid s \in \text{In}(a), \lambda_a^s(w) = \lambda_a(w, s) \text{ für } w \in I(a)^*\}.$$

Wir nennen nun zwei Automaten a und b *funktional äquivalent* ($a \sim b$), falls $\mathfrak{F}(a) = \mathfrak{F}(b)$ ist.

Die relationale Äquivalenz definieren wir über die *realisierte Ein-Ausgaberektion* $\mathfrak{R}(a)$. Diese ist gegeben durch

$$\mathfrak{R}(a) = \{(w, \lambda_a(w, s)) \mid s \in \text{In}(a), w \in I(a)^*\}.$$

Zwei Automaten a und b (mit $I(a) = I(b)$ und $O(a) = O(b)$) heißen dann *relational äquivalent* ($a \approx b$), falls $\mathfrak{R}(a) = \mathfrak{R}(b)$ gilt.

Offenbar gilt: $a \sim b \Rightarrow a \approx b$.

Nach HUMMITZSCH ([4]) führen wir nun zwischen „ \sim “ und „ \approx “ weitere Äquivalenzrelationen ein.

Definition 1.2. Seien a und b Automaten mit $I(a) = I(b)$ und $O(a) = O(b)$, sei ferner $k \in \mathbb{Z}_+$.

a und b sind *k-relational äquivalent* ($a \approx_k b$), falls es zu jedem $w \in I(a)^k$ und $s \in \text{In}(a)$ ein $s' \in \text{In}(b)$ gibt mit

$$\lambda_a(wv, s) = \lambda_b(wv, s')$$

für alle $v \in I(b)^*$ und umgekehrt.

Wir nennen nun zwei Automaten a und b *endlich relational äquivalent* ($a \approx^{\mathbb{E}} b$), falls es ein $k \in \mathbb{Z}_+$ mit $a \approx_k b$ gibt.

Folgende Beziehungen zwischen diesen Äquivalenzbegriffen sind offensichtlich.

Beobachtung. Seien a, b Automaten; dann gilt:

- i) $a \sim b \Leftrightarrow a \approx_0 b$;
- ii) $a \approx_k b \Rightarrow a \approx^{\mathbb{E}} b \Rightarrow a \approx b \quad (k \in \mathbb{Z}_+)$;
- iii) $a \approx_k b \Rightarrow a \approx_{k+1} b \quad (k \in \mathbb{Z}_+)$.

Die Implikationen in ii) und iii) sind nicht umkehrbar wie P. HUMMITZSCH gezeigt hat [4].

2. Beziehungen zwischen relationaler, endlich relationaler und funktionaler Äquivalenz für spezielle Automatenklassen

Wir wenden uns nun dem zweitgenannten Problembereich dieser Arbeit zu.

Hierzu ist zunächst anzuführen, daß NELSON [6] für nichtinitialisierte, streng zusammenhängende Automaten a und b gezeigt hat: $a \approx b \Rightarrow a \sim b$.

Ein ähnliches Resultat hat P. HUMMITZSCH [5] gezeigt, nämlich, daß für nicht-initialisierte, einfache Automaten a und b gilt:

$$a \approx b \Rightarrow a \sim b.$$

Wir ergänzen diese beiden Resultate

Ein Automat a heißt *irreduzibel*, falls a keinen nichttrivialen δ -Unterautomaten a' mit $I(a') = I(a)$ besitzt.

Es gilt der

Satz 2.1. *Sind a und b irreduzible nichtinitialisierte Automaten, dann gilt:*

$$a \approx^{\mathbb{E}} b \Rightarrow a \sim b.$$

Beweis. Bekanntermaßen ist ein Automat a genau dann irreduzibel, wenn a streng zusammenhängend oder fast streng zusammenhängend ist (vgl. [3]). Sind nun a und b streng zusammenhängend, so ergibt das Resultat von NELSON das gewünschte Ergebnis.

Sei nun a fast streng zusammenhängend und $a \not\approx_E b$. Sei ferner $S(a) = \{i \mid 0 \leq i \leq n\}$, wobei 0 der bei a ausgezeichnete Zustand ist. Sei nun $w \in I(a)^k$ mit $\delta_a(w, 1) \neq 0$, dann existiert $s \in S(b)$ mit

- (1) $\lambda_a(w, 1) = \lambda_b(w, s)$,
- (2) $\delta_a(w, 1) \sim \delta_b(w, s)$.

(Hierbei ist die Äquivalenz von Zuständen in üblicher Weise definiert.) Wir wählen nun für $0 \leq j \leq n$ Werte w_j mit

$$\delta_a(w_j, \delta_a(w, 1)) = j.$$

Dann gilt

$$j = \delta_a(w_j, \delta_a(w, 1)) \sim \delta_b(w_j, \delta_a(w, 1)).$$

Hieraus folgt aber die Behauptung.

Bemerkung. Man beachte in diesem Zusammenhang, daß jeder einfache Automat irreduzibel ist.

Wir studieren einige Beispiele, die die Voraussetzung „nichtinitialisiert“ beleuchten.

Beispiel 1. Betrachte a und b , beide definiert durch den Zustandsgraphen von Abb. 1 und $\text{In}(a) = \{s_1, s_3\}$, $\text{In}(b) = \{s_2, s_4\}$.

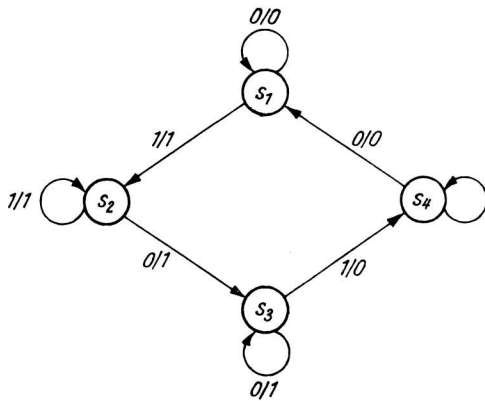


Abb. 1

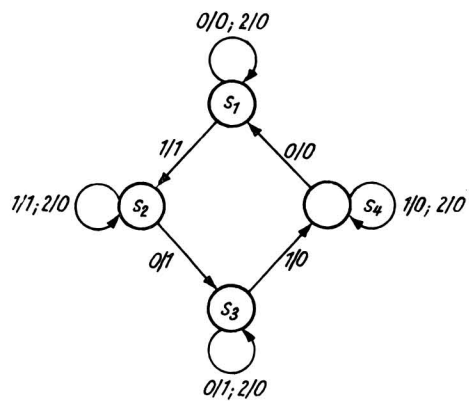


Abb. 2

Beispiel 1a. Durch Hinzunahme eines weiteren Eingangssymbols erhalten wir, daß i. allg. nicht einmal aus $a \approx_E b$ die Beziehung $a \approx b$ folgt. Dazu definieren wir a und b durch den Zustandsgraphen von Abb. 2 und $\text{In}(a) = \{s_1, s_3\}$, $\text{In}(b) = \{s_2, s_4\}$. Offenbar gilt $a \approx b$ und $a \not\approx_E b$.

Beispiel 2. Betrachte a und b , definiert durch die Zustandsgraphen von Abb. 3. Es gilt: a und b sind nichtinitialisierte fast streng zusammenhängende Automaten. Man rechnet nach: $a \approx b$, aber $a \not\approx_E b$.

Beispiel 2a. Betrachte a und b mit dem Zustandsgraphen von Abb. 4 und $\text{In}(a) = \{s_1, s_2\}$, $\text{In}(b) = \{s_1, s_4\}$. Es gilt: $a \approx_E b$, $a \not\sim b$, a, b fast streng zusammenhängend.

Beispiel 2b. Seien a und b durch den Zustandsgraphen von Abb. 4 und $\text{In}(a) = \{s_1, s_2, s_3\}$ und $\text{In}(b) = \{s_1, s_2\}$; dann gilt $a \approx b$, aber $a \not\approx_E b$.

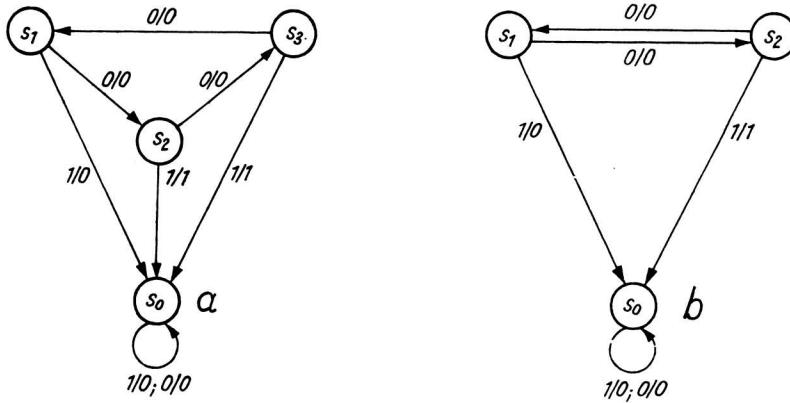


Abb. 3

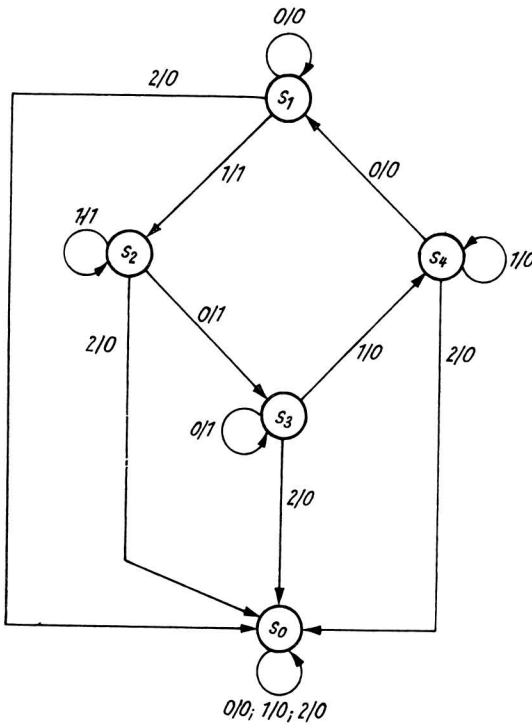


Abb. 4

Ein weiterer Typ von Automaten, der Aussagen über Zusammenhänge zwischen relationaler und funktionaler Äquivalenz zuläßt, sind die Automaten mit endlichem Gedächtnis ([2]). Es gilt:

Hilfssatz 2.1. Seien a und b Automaten mit endlichem Gedächtnis der Ordnung p ; dann gilt

$$a \approx b \Rightarrow a \approx_p b.$$

Beweis. Sei $s \in \text{In}(a)$, $w \in I(a)^p$. Da $a \approx b$ ist, existiert $s' \in \text{In}(b)$ mit

$$\lambda_a(w, s) = \lambda_b(w, s').$$

Nach einem in [2] zu findenden Hilfssatz gilt dann aber

$$\delta_a(w, s) \sim \delta_b(w, s').$$

Hieraus folgt die Behauptung

Eine wichtige Teilklasse von Automaten mit endlichem Gedächtnis sind die sog. linearen Automaten ([2]). Die beweisrelevante Eigenschaft kann verallgemeinert werden (vgl. [9]). Wir zeigen:

Hilfssatz 2.2. Sei a ein Automat mit

- (1) $\forall w \in I(a)^*, s, s' \in S(a):$
 $\lambda_a(w, s) = \lambda_a(w, s') \Rightarrow (\forall w' \in I(a)^* \ \& \ L(w') = L(w): \lambda_a(w', s) = \lambda_a(w', s'));$
- (2) $\forall w, w' \in I(a)^*, s \in S(a):$
 $\lambda_a(w, s) = \lambda_a(w', s) \Rightarrow (\forall s \in S(a): \lambda_a(w, s) = \lambda_a(w', s)),$

dann besitzt a endliches Gedächtnis.

Beweis. Wir zeigen mehr, nämlich

$$\exists p \forall w \in I(a)^p \ I(a)^*, s, s' \in S(a): \lambda_a(w, s) = \lambda_a(w, s') \Rightarrow \lambda_a^s = \lambda_a^{s'}$$

(p -Diagnostizierbarkeit ([2])).

Sei $x_0 \in I(a)$ fest, $s, s' \in S(a)$ mit

$$\begin{aligned} \lambda_a^s \neq \lambda_a^{s'} &\Leftrightarrow \exists w \in I(a)^* \quad \text{mit} \quad \lambda_a(w, s) \neq \lambda_a(w, s') \\ &\Leftrightarrow \exists k \geq 1 \quad \text{mit} \quad \lambda_a(x_0^k, s) \neq \lambda_a(x_0^k, s'). \end{aligned}$$

Betrachte nun:

$$p(s, s') = \begin{cases} 0, & \text{falls} \quad \lambda_a^s = \lambda_a^{s'}, \\ \text{Min} \{k \mid \lambda_a(x_0^k, s) \neq \lambda_a(x_0^k, s')\} & \text{sonst.} \end{cases}$$

Sei $p = \text{Max}_{s, s' \in S(a)} p(s, s')$. Betrachte nun $w \in I(a)^p \subseteq I(a)^*$; $s, s' \in S(a)$ mit $\lambda_a(w, s) = \lambda_a(w, s')$. Dann gilt

$$\lambda_a(x_0^{L(w)}, s) = \lambda_a(x_0^{L(w)}, s').$$

Da $L(w) \geq p \geq p(s, s')$, folgt (im Falle $p(s, s') > 0$)

$$\lambda_a(x_0^{p(s, s')}, s) \neq \lambda_a(x_0^{p(s, s')}, s')$$

und hieraus

$$\lambda_a(x_0^{L(w)}, s) \neq \lambda_a(x_0^{L(w)}, s').$$

Hieraus ergibt sich ein Widerspruch. Also ist $p(s, s') = 0$, d. h. $\lambda_a^s = \lambda_a^{s'}$.

Beispiel 3 ([7]). Die Tatsache, daß lineare Automaten die im Hilfssatz 2.2 genannte Eigenschaft besitzen, geht zurück auf die allgemeine Antwortformel. Diese kann noch in allgemeineren Situationen bewiesen werden.

Wir betrachten Automaten a mit der Eigenschaft, daß $I(a)$, $S(a)$ und $O(a)$ Monoide sind. In Anlehnung an die linearen Automaten schreiben wir die Verknüpfung, auch wenn diese nichtkommutativ ist, mit $+$, das neutrale Element bezeichnen wir mit 0 .

Sei $\hat{\lambda}_a$ die Ergebnisfunktion (letztes ausgegebenes Symbol). Wir setzen nun voraus, daß δ_a und λ_a Monoidhomomorphismen sind. Dann gilt stets:

$$\delta_a(x, s) = \delta_a(x, 0) + \delta_a(0, s)$$

und

$$\lambda_a(x, s) = \lambda_a(x, 0) + \lambda_a(0, s)$$

($x \in I(a)$ & $s \in S(a)$). Offenbar werden dann durch

$$A = \delta_a(-, 0), \quad B = \delta_a(0, -),$$

$$C = \lambda_a(-, 0), \quad D = \lambda_a(0, -)$$

Monoidhómomorphismen definiert.

Man zeigt nun in üblicher Weise für $w = x_1, \dots, x_r$

$$\delta_a(w, s) = Ax_r + BAx_{r-1} + \dots + BA^{r-1}x_1 + B_s^r$$

&

$$\hat{\lambda}_a(w, s) = Cx_r + DAx_{r-1} + DBAx_{r-2} + \dots + DB^{r-2}Ax_1 + DB_s^r.$$

Gelten nun in $S(a)$ und $O(a)$ beide Kürzungsregeln, so erfüllt a die Voraussetzungen des Hilfssatzes 2.2.

Wir schließen diesen Abschnitt ab mit der Feststellung ([2]), daß für lineare Automaten a und b sogar gilt:

$$a \approx b \Rightarrow a \sim b.$$

3. Charakterisierung und Entscheidbarkeit der relationalen und endlich relationalen Äquivalenz

Wir wollen die verschiedenen angesprochenen Äquivalenzbegriffe charakterisieren. Zunächst bemerken wir, daß GRAY und HARRISON [10] gezeigt haben, daß allgemein die relationale Äquivalenz von Automaten entscheidbar ist.

Wir charakterisieren die k -relationale Äquivalenz bei festem k . Hierfür geben wir einen neuen Beweis, der wesentlich kürzer ist als der HUMMITSZsche Beweis.

Im Falle $k = 0$, haben wir es mit der funktionalen Äquivalenz zu tun. Die Charakterisierung ist in diesem Fall wohlbekannt. Durch Übergang zu den „ k -ten Simultanten“ reduziert sich nach P. HUMMITSZSCH [4] das Problem auf den Fall $k = 1$.

Wir betrachten also das Problem, ob zwei gegebene Automaten a und b 1-relational äquivalent sind. Diese Frage führen wir wieder zurück auf die funktionale Äquivalenz. Hierzu betrachten wir

$$S_1 = \delta_a(I(a), S(a)).$$

Offenbar gilt: $s \in S_1 \Rightarrow s \in \text{In}(a)$.

Definiere nun \hat{a} durch

$$1) \quad I(\hat{a}) = I(a), \quad O(\hat{a}) = 2^{O(a) \times I(a)} \times O(a)$$

$$S(\hat{a}) = I(a),$$

$$\text{In}(\hat{a}) = \{s \mid \exists x_0 \in I(a), s_0 \in \text{In}(a) \quad \text{mit} \quad \delta_a(x_0, s_0) = s\}.$$

$$2) \quad \delta_{\hat{a}}(x, s) = \delta_a(x, s)$$

$$\lambda_{\hat{a}}(x, s) = (\{(\lambda_a(x_1, s_1), x_1) \mid x_1 \in I(a), s_1 \in \text{In}(a), \delta_a(x_1, s_1) = s\}, \lambda_a(x, s)).$$

Beobachtung.

$$\lambda_a(x, s) \neq (\emptyset, \lambda_a(x, s)) \Leftrightarrow s \in \text{In}(\hat{a}).$$

Wir berechnen für $w = x_1 \cdots x_r (x_i \in I(a), 1 \leq i \leq r)$ & $s \in S_1$: $\lambda_a(x_1, \dots, x_r, s)$.

$$\begin{aligned} \lambda_a(x_1, \dots, x_r, s) &= \\ &= \{(\lambda_a(y_1, s_1), y_1) \mid y_1 \in I(a), s_1 \in \text{In}(a), \delta_a(y_1, s_1) = s\}, \lambda_a(x_1, s)\} \cdot \\ &\cdot \lambda_a(x_2 \cdots x_r, \delta_a(x_1, s)) = \\ &= \{(\lambda_a(y_1, s_1), y_1) \mid y_1 \in I(a), s_1 \in \text{In}(a), \delta_a(y_1, s_1) = s\}, \lambda_a(x_1, s)\} \cdot \dots \\ &\dots \\ &\dots \cdot \{(\lambda_a(y_i, s_i), y_i) \mid y_i \in I(a), s_i \in \text{In}(a), \delta_a(y_i, s_i) = \delta_a(x_1 \cdots x_{i-1}, s)\}, \\ &\lambda_a(x_r, \delta_a(x_1 \cdots x_{i-1}, s))\} \cdot \dots \\ &\dots \\ &\dots \cdot \{(\lambda_a(y_r, s_r), y_r) \mid y_r \in I(a), s_r \in \text{In}(a), \delta_a(y_r, s_r) = \delta_a(x_1 \cdots x_{r-1}, s)\}, \\ &\lambda_a(x_r, \delta_a(x_1 \cdots x_{r-1}, s))\} \cdot \end{aligned}$$

Setze

$$A_i = \{(\lambda_a(y_i, s_i) \mid y_i \in I(a), s_i \in \text{In}(a), \delta_a(y_i, s_i) = \delta_a(x_1 \cdots x_{i-1}, s))\}.$$

Nun gilt

$$\lambda_a(x_1 \cdots x_r, s) = (A_1, \lambda_a(x_1, s)) \cdots (A_r, \lambda_a(x_r, \delta_a(x_1 \cdots x_{r-1}, s))) \cdot$$

Hilfssatz 3.1. Seien a und b Automaten. Dann gilt: $a \approx_1 b \Leftrightarrow \hat{a} \sim \hat{b}$.

Beweis. (\Rightarrow) $a \approx_1 b \Rightarrow \forall x \in I(a), s \in \text{In}(a) \exists s' \in \text{In}(b)$: $\lambda_a^{\delta_a(x, s)} = \lambda_b^{\delta_b(x, s')}$ und umgekehrt. Sei nun $s \in \text{In}(\hat{a})$, dann existiert $x \in I(a), \hat{s} \in \text{In}(a)$ mit

$$\lambda_a^s = \lambda_a^{\delta_a(x, \hat{s})}.$$

Nun existiert s' mit $\lambda_a^{\delta_a(x, \hat{s})} = \lambda_a^{\delta_a(x, \hat{s})}$. Setze $s' = \delta_a(x, \hat{s})$. Dann gilt $\lambda_a^s = \lambda_b^{s'}$. Es bleibt für $w \in I(a)^*, s \in S(a)$ zu zeigen:

$$\begin{aligned} &\{(\lambda_a(y, \hat{s}), y) \mid y \in I(a), \hat{s} \in \text{In}(a), \delta_a(y, \hat{s}) = \delta_a(w, s)\} = \\ &= \{(\lambda_b(y, \hat{s}'), y) \mid y \in I(a), \hat{s}' \in \text{In}(a), \delta_b(y, \hat{s}') = \delta_b(w, s')\}. \end{aligned}$$

Es gilt $\lambda_a^s = \lambda_b^{s'}$. Nun gilt

$$\lambda_a^{\delta_a(w, s)} = \lambda_a^{\delta_a(y, s)} = \lambda_b^{\delta_b(y, s')} = \lambda_b^{s''}$$

für \hat{s}' geeignet, $s'' = \delta_b(x, \hat{s}')$. Aus $\lambda_a^s = \lambda_b^{s'}$ folgt $\lambda_a^{\delta_a(w, s)} = \lambda_b^{\delta_b(w, s')}$. Also $s'' = \delta_b(w, s')$, wegen der Reduziertheit. Damit ist \subseteq bewiesen. \supseteq folgt durch Vertauschung der Rolle von a und b .

(\Leftarrow) Sei $x \in I(a), s \in \text{In}(a)$. Betrachte $\delta_a(x, s) = s'$. Nun existiert, da $s' \in \text{In}(\hat{a})$ ist, $\hat{s}' \in \text{In}(b)$ mit

$$\lambda_a^{\hat{s}} = \lambda_b^{\hat{s}'}$$

Aus der Antwortformel folgt

$$\lambda_a^{\delta_a(x, s)} = \lambda_b^{s'}$$

Wir haben zu zeigen: $\hat{s}' = \delta_b(x, \hat{s}), \lambda_a(x, s) = \lambda_b(x, \hat{s})$ mit $\hat{s} \in \text{In}(b)$. Insbesondere gilt nun:

$$\begin{aligned} &\{(\lambda_a(y, \tilde{s}), y) \mid y \in I(a), \tilde{s} \in \text{In}(a), \delta_a(y, \tilde{s}) = s'\} = \\ &= \{(\lambda_b(y, \tilde{s}'), y) \mid y \in I(b), \tilde{s}' \in \text{In}(a), \delta_a(y, \tilde{s}') = \hat{s}'\}. \end{aligned}$$

Da $(\lambda_a(x, s), x) \in \{(\lambda_a(y, \hat{s}), y) \mid y \in I(a), \tilde{s} \in \text{In}(a), \delta_a(y, \tilde{s}) = s'\}$, folgt: $\hat{s} \in \text{In}(b)$ mit

$$\lambda_a(x, s) = \lambda_b(x, \hat{s})$$

und

$$\lambda_a^{s'} = \lambda_a^{\delta_a(x, s)} = \lambda_b^{\delta_b(x, \hat{s})} = \lambda_b^{\hat{s}'}$$

Wegen der Reduziertheit folgt: $\delta_b(x, \hat{s}) = \hat{s}'$. Hieraus erhalten wir den

Satz 3.1 (HUMMITZSCH [4]). *Sei $k \in \mathbf{Z}_+$, a und b endliche Automaten. Es ist entscheidbar, ob $a \stackrel{k}{\approx} b$ oder $a \not\stackrel{k}{\approx} b$ gilt.*

Als abschließenden Satz wollen wir beweisen, daß auch die endliche relationale Äquivalenz entscheidbar ist.

Seien a und b endliche Automaten mit $a \approx b$. Sei $S(a) = \{s_1, \dots, s_m\}$, $S(b) = \{s'_1, \dots, s'_n\}$ ($\text{card}(S(a)) = m$, $\text{card}(S(b)) = n$). Setze $K_a = m \cdot n^n$. Betrachte zu $w \in I(a)^{K_a}$ und $s \in S(a)$ die Menge $M(w, s) = \{(\delta_a(w', s), \delta_b(w', s'_1), \dots, \delta_b(w', s'_n)) \mid w' < w, L(w') = i, 0 \leq i \leq L(w)\}$.

Offenbar gilt

$$\text{card}(M(w, s)) < K_a + 1.$$

Also existieren i und j mit (o.B.d.A.) $i < j$ und

$$w_j = w_i x, \quad w = w_j v$$

mit $x, v \in I(a)^*$, $L(x) \geq 1$, $L(v) \geq 0$, sowie

$$\begin{aligned} \delta_a(w_j, s) &= \delta_a(w_i, s) \\ \delta_b(w_j, s') &= \delta_b(w_i, s') \quad (s' \in S(b)). \end{aligned}$$

Sei $K = \text{Max}(K_a, K_b)$.

Hilfssatz 3.2. *Es gilt: $a \stackrel{K}{\approx} b \Leftrightarrow a \stackrel{K+1}{\approx} b$.*

Beweis. Wir haben die Richtung \Leftarrow zu zeigen. Wir führen den Beweis indirekt.

Annahme: $a \not\stackrel{K}{\approx} b$. — Dann existieren $w \in I(a)^K$ und $s \in \text{In}(a)$ mit

$$s' \in \text{In}(b) \lambda_a^{\delta_a(w, s)} \neq \lambda_b^{\delta_b(w, s')}$$

oder

$$\lambda_a(w, s) \neq \lambda_b(w, s').$$

Sei nun o.B.d.A. $K = K_a$. Betrachte $M(w, s)$. Dann existiert eine Zerlegung

$$w = w' x v \quad \text{mit} \quad L(x) \geq 1.$$

$$\delta_a(x, \delta_a(w', s)) = \delta_a(w', s). \quad (1)$$

$$\delta_b(w', s') = \delta_b(w' x, s') \quad (s' \in S(b)). \quad (2)$$

Sei $w_0 = w' x x v$. Nun existiert s' mit

$$\lambda_a^{\delta_a(w' x x v, s)} = \lambda_b^{\delta_b(w' x x v, s')}; \quad (3)$$

$$\lambda_a(w' x x v, s) = \lambda_b(w' x x v, s'). \quad (4)$$

Es gilt $\delta_a(w' x x v, s) = \delta_a(w, s)$ (nach (1)) und $\delta_b(w' x x v, s') = \delta_b(w, s')$ nach (2). Nach (4) folgt $\lambda_a(w, s) = \lambda_b(w, s')$. Hieraus erhalten wir einen Widerspruch!

Satz 3.2. *Seien a und b endliche Automaten; es ist entscheidbar, ob $a \stackrel{E}{\approx} b$ oder $a \not\stackrel{E}{\approx} b$ gilt.*

Beweis. Sei $K = \text{Max}(K_a, K_b)$. Nach obigem Hilfssatz gilt

$$a \stackrel{E}{\approx} b \Leftrightarrow a \stackrel{K}{\approx} b, \quad \text{sofern } a \approx b.$$

Mit Satz 3.1 und der Entscheidbarkeit der relationalen Äquivalenz ergibt sich dann die Behauptung.

Literatur

- [1] GÉCSEG, F., I. PEAK, Algebraic Theory of Automata. Akadémiai Kiadó, Budapest 1972.
- [2] HARRISON, M. A., Lectures on Linear Sequential Machines. Academic Press, New York 1969.
- [3] HOTZ, G., H. WALTER, Automatentheorie und formale Sprachen; II: Endliche Automaten. BI-Hochschultaschenbücherverlag, Bd. 822/822a, Mannheim 1969.
- [4] HUMMITZSCH, P., Zur schwachen Äquivalenz von endlichen Automaten. EIK 7 (1971) 8, 467–484.
- [5] HUMMITZSCH, P., Beziehungen zwischen einigen schwachen Äquivalenzen endlicher Automaten. EIK 8 (1972) 2/3, 77–86.
- [6] NELSON, R. J., Introduction to Automata. Case Institute of Technology Cleveland, Ohio; J. Wiley & Sons, Inc., New York.
- [7] REUSCH, B., private Mitteilung.
- [8] SCHMITT, F., Relationale und endlich-relationale Äquivalenz von endlichen Automaten. Diplomarbeit, Fachbereich Informatik der TH Darmstadt, 1973.
- [9] STUCKY, W., Linear realisierbare endliche Automaten. Diss., Universität des Saarlandes, Saarbrücken 1969.
- [10] GRAY, J. N., M. A. HARRISON, The Theory of Sequential Relations. Inform. and Control 9 (1966), 435–468.

Kurzfassung

In der vorliegenden Arbeit befassen wir uns mit Zusammenhängen zwischen relationaler, endlich relationaler und funktionaler Äquivalenz von Automaten. Wir geben für die Entscheidbarkeit der k -relationalen Äquivalenz einen neuen Beweis und zeigen die Entscheidbarkeit der endlich relationalen Äquivalenz.

Abstract

In this paper we discuss connections between relational, finite relational and functional equivalence of automata. We give a new proof of the decidability of the k -relational equivalence and show that the finite relational equivalence is decidable.

Резюме

В настоящей работе рассматриваются соотношения между экспериментальной (relational), конечно экспериментальной и функциональной эквивалентностями автоматов. Приводится новое доказательство для разрешимости k -экспериментальной эквивалентности и доказывается разрешимость конечно экспериментальной эквивалентности.

(Eingang: Erste Fassung am 23. 12. 1973,
vorliegende Fassung am 30. 5. 1975)

Anschrift des Verfassers:

Prof. Dr. H. K.-G. Walter
Forschungsgruppe Automatentheorie
und formale Sprachen des
Fachbereichs Informatik der
Technischen Hochschule Darmstadt
Magdalenenstr. 11
61 Darmstadt
BRD