Contributions to the structure theory of ω-languages

Albert Zeyer

10. Februar 2011

Inhaltsverzeichnis

1 Introduction **3**

2 Automaton **4**
2.1 Path 4
2.2 Acceptence of finite words 4

3 $*$-languages **4**
3.1 regular languages 4
3.2 piece-wise testable 5
3.3 k-locally testable 5
3.4 dot-depth-n 5
3.5 starfree 5
3.6 locally modulo testable 5
3.7 R-trivial 5
3.8 endlich / co-endlich 5
3.9 endwise testable 5

4 ω-languages **5**
4.1 Büchi automaton 5
4.2 Muller automaton 5
4.3 Rabin automaton 5
4.4 Staiger Wagner class of $\mathcal{K}$ 6

5 Operations: $*$-language K to ω-language $L_\omega(K)$ **6**
5.1 6

6 $*$-Sprachklassen **7**
6.1 regular 7
6.2 piece-wise testable 7
6.3 k-locally testable 7
6.4 dot-depth-n 7
6.5 starfree 7
6.6 locally modulo testable 7
6.7 R-trivial 7
6.8 endlich / co-endlich 7
6.9 endwise testable 7

7 ω-Sprachklassen **7**
7.1 Staiger Wagner Klasse zu $\mathcal{K}$ 7

8 Operationen: von $*$-Sprache K zu ω-Sprache $L_\omega(K)$ **7**
8.1 7

9 Lemmas **8**
9.1 piece-wise testable 8
9.2 extension of $\mathcal{L}^*(\mathrm{FO}[+1])$ 8
9.3 limit of $\mathcal{L}^*(\mathrm{FO}[<])$ 9

1 Introduction

...

2 Automaton

An **automaton** $\mathcal{A}$ on the alphabet Σ is given by a set Q of states and a subset $E \subset Q \times A \times Q$ of transitions. In most cases you also have a subset $I \subset Q$ of initial states and a subset $F \subset Q$ of final states.

We write:

$$\mathcal{A} = (Q, \Sigma, E, I, F).$$

The automaton is **finite** iff Q and Σ are finite.

The automaton is **deterministic** iff E is a set of functions $Q \times A \to \mathcal{Q}$ and there is only a single initial state.

2.1 Path

Two transitions $(p, a, q), (p', a', q') \in E$ are **consecutive** iff $q = p'$.

A **path** in the automaton $\mathcal{A}$ is a sequence of consecutive transitions, written as:

$$q_0 \to^{a_0} q_1 \to^{a_1} q_2 \dots$$

2.2 Acceptence of finite words

An automaton $\mathcal{A} = (Q, \Sigma, E, I, F)$ **accepts** a finite word $w = (a_0, a_1, ..., a_n) \in \Sigma^*$ iff there is a path $q_0 \to^{a_0} q_1 \to^{a_1} q_2 \cdots \to^{a_n} q_{n+1}$ with $q_0 \in I$ und $q_{n+1} \in F$.

The language $L^*(\mathcal{A})$ is defined as set of all words which are accepted by $\mathcal{A}$.

3 $*$-languages

The $*$-languages are all languages of words $w \in \Sigma^*$, i.e. the set of languages of finite words.

3.1 regular languages

A languages is **regular** iff an automaton accepts it.

3.2 piece-wise testable

3.3 k-locally testable

3.4 dot-depth-n

3.5 starfree

3.6 locally modulo testable

3.7 R-trivial

3.8 endlich / co-endlich

3.9 endwise testable

4 ω-languages

4.1 Büchi automaton

An automaton $\mathcal{A} = (Q, \Sigma, E, I, F)$ **Büchi-accepts** a word $\alpha = (a_0, a_1, a_2, ...) \in \Sigma^\omega$ iff there is an infinite path $q_0 \to^{a_0} q_1 \to^{a_1} q_2 \to^{a_2} q_3...$ with $q_0 \in I$ and $\{q_i | q_i \in F\}$ infinite, i.e. which reaches a state in F infinitely often.

The language $L^\omega(\mathcal{A})$ is defined as the set of all infinite words which are Büchi-accepted by $\mathcal{A}$.

An automaton $\mathcal{A}$ is a Büchi automaton iff you use the Büchi-acceptence.

4.2 Muller automaton

A Muller automaton $\mathcal{A}$ is a finite, deterministic automaton with **Muller acceptence** and a set $\mathcal{T} \in 2^Q$, called the **table** of the automaton (instead of the set F). A word $w \in \Sigma^\omega$ is accepted iff there is a path p with $\mathrm{Inf}(p) \in \mathcal{T}$, where $\mathrm{Inf}(p)$ is the set of infinitely often reached states of the path p.

We write:

$$\mathcal{A} = (Q, \Sigma, E, i, \mathcal{T}).$$

4.3 Rabin automaton

A Rabin automaton is a tuple $\mathcal{A} = (Q, \Sigma, E, i, \mathcal{R})$, where (Q, Σ, E) is a deterministic automaton, i is the initial state and $\mathcal{R} = \{(L_j, U_j) | j \in J\}$ is a family of pairs of state-sets. A path p is successfull iff it starts in i and there is an index $j\ inJ$ such that p reaches U_j infinitely often and L_j only finitely often. If the automaton is finite, this is equivalent to

$$\mathrm{Inf}(p) \cap L_j = \emptyset \text{ and } \mathrm{Inf}(p) \cap U_j \neq \emptyset.$$

4.4 Staiger Wagner class of $\mathcal{K}$

5 Operations: $*$-language K to ω-language $L_\omega(K)$

5.1 ...

a) * alle Sprachen $K\dot{\Sigma}^\omega = \mathrm{ext}(K)$, $K \in \mathcal{K}$

* offene G

* Staiger Wagner Klasse http://de.wikipedia.org/wiki/Staiger-Wagner-Automat Erich Grädel, Wolfgang Thomas und Thomas Wilke (Herausgeber), Automata, Logics, and Infinite Games, LNCS 2500, 2002, Seite 20 (auf englisch) http://www.automata.rwth-aachen.de/material/skripte/areas-english.pdf - s.53

a') dual $\overline{K} = \omega$-Wörter, deren alle Präfixe in K sind

b) Sprachen $\lim \mathcal{K}$ BC Muller-erkennbare (BC: boolean closure ?)

b') von einer Stelle an alle Prefixe in K

c) Kleene-Closure

alle der Form $\cup_{i=1}^n U_i \dot{V}_i^\omega$, $U_i, V_i \in \mathcal{K}$

d) $\mathcal{K}$ nicht suffix sensitiv

$K \in \mathcal{K} \Rightarrow K\dot{\Sigma}^* \in \mathcal{K}$

Hauptfrage: Für welche $\mathcal{K}$ ergibt sich eine andere Sprache als bei $\mathcal{K} = \text{Reg}$.

6 $*$-Sprachklassen

6.1 regular

6.2 piece-wise testable

6.3 k-locally testable

6.4 dot-depth-n

6.5 starfree

6.6 locally modulo testable

6.7 R-trivial

6.8 endlich / co-endlich

6.9 endwise testable

7 ω-Sprachklassen

7.1 Staiger Wagner Klasse zu $\mathcal{K}$

8 Operationen: von $*$-Sprache K zu ω-Sprache $L_\omega(K)$

8.1 ...

a) * alle Sprachen $K\dot{\Sigma}^\omega = \text{ext}(K)$, $K \in \mathcal{K}$
* offene G
* Staiger Wagner Klasse http://de.wikipedia.org/wiki/Staiger-Wagner-Automat Erich Grädel, Wolfgang Thomas und Thomas Wilke (Herausgeber), Automata, Logics, and Infinite Games, LNCS 2500, 2002, Seite 20 (auf englisch) http://www.automata.rwth-aachen.de/material/skripte/areas-english.pdf - s.53
a') dual $\overline{K} = \omega$-Wörter, deren alle Präfixe in K sind
b) Sprachen $\lim \mathcal{K}$ BC Muller-erkennbare (BC: boolean closure ?)
b') von einer Stelle an alle Prefixe in K
c) Kleene-Closure
alle der Form $\cup_{i=1}^{n} U_i \dot{V}_i^\omega$, $U_i, V_i \in \mathcal{K}$
d) $\mathcal{K}$ nicht suffix sensitiv
$K \in \mathcal{K} \Rightarrow K\dot{\Sigma}^* \in \mathcal{K}$

9 Lemmas

9.1 piece-wise testable

Theorem 9.1.

$$\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\text{piece-wise testable}) = \mathrm{BC}\lim \mathcal{L}^*(\text{piece-wise testable})$$

Proof. L piece-wise testable $\Leftrightarrow$ L is a boolean algebra of $\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*$

$\subseteq$: It is sufficient to show $\mathrm{ext}(\mathcal{L}^*(\text{piece-wise testable})) \subseteq \mathrm{BC}\lim \mathcal{L}^*(\text{piece-wise testable})$. By complete induction:

$$\begin{aligned}
&\mathrm{ext}(\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*) = \Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^\omega = \lim(\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*)\\
&\mathrm{ext}(\neg(\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*)) = \Sigma^\omega = \lim(\Sigma^*)\\
&\mathrm{ext}(\emptyset) = \emptyset = \lim(\emptyset)
\end{aligned}$$

It is sufficient to show negation only for such ground terms because we can always push the negation down.

$$\begin{aligned}
&\mathrm{ext}(A \cup B) = \mathrm{ext}(A) \cup \mathrm{ext}(B)\\
&\mathrm{ext}(A \cap B) = \mathrm{ext}(A) \cap \mathrm{ext}(B)
\end{aligned}$$

This makes the induction complete.

$\supseteq$: It is sufficient to show $\lim(\mathcal{L}^*(\text{piece-wise testable})) \subseteq \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\text{piece-wise testable})$.

$$\begin{aligned}
&\lim(\emptyset) = \mathrm{ext}(\emptyset),\ \lim(\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*) = \mathrm{ext}(\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*) \quad \text{(see above)}\\
&\lim(\neg(\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*)) = \left\{\alpha \in \Sigma^\omega \,\middle|\, \exists^\omega n\colon \alpha[0,n] \notin \Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*\right\}\\
&\qquad = \left\{\alpha \in \Sigma^\omega \,\middle|\, \forall n\colon \alpha[0,n] \notin \Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*\right\}\\
&\qquad = \neg\,\mathrm{ext}(\Sigma^* a_1 \Sigma^* a_2 \cdots a_n \Sigma^*)\\
&\lim(A \cup B) = \left\{\alpha \in \Sigma^\omega \,\middle|\, \exists^\omega n\colon \alpha[0,n] \in A \cup B\right\} = \lim(A) \cup \lim(B)\\
&\lim(A \cap B) = \left\{\alpha \in \Sigma^\omega \,\middle|\, \exists^\omega n\colon \alpha[0,n] \in A \cap B\right\}
\end{aligned}$$

and because A, B are piece-wise testable

$$= \left\{\alpha \in \Sigma^\omega \,\middle|\, \exists n : \forall m > n\colon \alpha[0,m] \in A \cap B\right\} = \lim(A) \cap \lim(B)$$

□

9.2 extension of $\mathcal{L}^*(\mathrm{FO}[+1])$

Theorem 9.2.

$$\mathcal{L}^\omega(\mathrm{FO}[+1]) = \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{FO}[+1])$$

Proof. From [Tho96, Theorem 4.8], we know that each formular in FO[+1] is equivalent (for both finite and infinite words) to a boolean combination of statements "sphere $\sigma \in \Sigma^+$ occurs $\geq n$ times". That statement can be expressed by a sentence of the form

$$\psi := \exists \overline{x_1} \cdots \exists \overline{x_n} \varphi(\overline{x_1}, \cdots, \overline{x_n})$$

where each $\overline{x_i}$ is a $|\sigma|$-tuple of variables and the formula φ states:

$$\bigwedge_{\substack{i,j\in\underline{n},\\ i\neq j,\\ k,l\in\underline{|\sigma|}}} x_{i,k} \neq x_{j,l} \ \wedge \ \bigwedge_{\substack{i\in\underline{n},\\ k\in\underline{|\sigma|-1}}} x_{i,k+1} = x_{i,k}+1 \ \wedge \ \bigwedge_{\substack{i\in\underline{n},\\ k\in\underline{|\sigma|}}} Q_{\sigma_k} x_{i,k}$$

For ψ, we have:

$$\alpha \models \psi \Leftrightarrow \exists n \colon \alpha[0,n] \models \psi \quad \text{for all } \alpha \in \Sigma^\omega,$$

i.e.

$$L^\omega(\psi) = \operatorname{ext} L^*(\psi).$$

Any formular in FO[+1] can be expressed as a boolean combination of ψ-like formular. With

$$L^\omega(\neg\psi) = \neg L^\omega(\psi)$$
$$L^\omega(\psi_1 \wedge \psi_2) = L^\omega(\psi_1) \cap L^\omega(\psi_2)$$
$$L^\omega(\psi_1 \vee \psi_2) = L^\omega(\psi_1) \cup L^\omega(\psi_2)$$

we get:

$$\mathcal{L}^\omega(\mathrm{FO}[+1]) = \mathrm{BC}\operatorname{ext}\mathcal{L}^*(\mathrm{FO}[+1]).$$

□

9.3 limit of $\mathcal{L}^*(\mathrm{FO}[<])$

Theorem 9.3.

$$\mathcal{L}^\omega(\mathrm{FO}[<]) = \mathrm{BC}\lim\mathcal{L}^*(\mathrm{FO}[<])$$

Proof. Let $\varphi \in \mathrm{FO}[<]$. By the [Tho81, Normal Form Theorem (4.4)] there are bounded formulas $\varphi_1(y), \cdots, \varphi_r(y), \psi_1(y), \cdots, \psi_r(y)$ such that for all $\alpha \in \Sigma^\omega$:

$$\alpha \models \varphi \Leftrightarrow \alpha \models \bigvee_{i=1}^{r} (\forall x \exists y > x \colon \varphi_i(y)) \wedge \neg(\forall x \exists y > x \colon \psi_i(y))$$

Thus:

$$\alpha \models \varphi \Leftrightarrow \bigvee_{i=1}^{r} \underbrace{(\alpha \models \forall x \exists y > x \colon \varphi_i(y))} \wedge \neg(\alpha \models \forall x \exists y > x \colon \psi_i(y))$$
$$\Leftrightarrow \forall x \exists y > x \colon \alpha[0,n] \models \varphi_i(\omega)$$
$$\Leftrightarrow \exists^\omega n \colon \alpha[0,n] \models \varphi_i(\omega)$$
$$\Leftrightarrow \alpha \in \lim L^*(\varphi_i(\omega))$$

where $\varphi_i(\omega)$ stands for φ_i with all bounds removed. I.e. we have

$$L^\omega(\varphi) = \bigcup_{i=1}^{r} \lim(L^*(\varphi_i(\omega)) \cap \neg\lim(L^*(\psi_i(\omega))),$$

and thus

$$L^\omega(\varphi) \in \mathrm{BC}\lim \mathcal{L}^*(\mathrm{FO}[<]).$$

We have prooved the $\subseteq$-direction. For $\supseteq$:

$$\begin{aligned}
&\alpha \in \lim(L^*(\varphi))\\
\Leftrightarrow\ &\exists^\omega n\colon \alpha[0,n] \models \varphi\\
\Leftrightarrow\ &\alpha \models \forall x \exists y > x\colon \varphi(y)\\
\Leftrightarrow\ &\alpha \in L^\omega(\forall \exists y > x\colon \varphi(y))
\end{aligned}$$

where $\varphi(y)$ stands for φ with all variables bounded by y. I.e.

$$\lim \mathcal{L}^*(\mathrm{FO}[<]) \subseteq \mathcal{L}^\omega(\mathrm{FO}[<]),$$

and thus also

$$\lim \mathrm{BC}\, \mathcal{L}^*(\mathrm{FO}[<]) \subseteq \mathcal{L}^\omega(\mathrm{FO}[<]).$$

Thus we have prooved the equality. $\square$

Literatur

[Tho81] Wolfgang Thomas. A combinatorial approach to the theory of omega-automata. *Information and Control*, 48(3):261–283, 1981.

[Tho96] Wolfgang Thomas. Languages, automata, and logic. In *Handbook of Formal Languages*, pages 389–455. Springer, 1996.