Language Operations and a Structure Theory of ω-Languages

Diploma Thesis
in Computer Science

by
Albert Zeyer

submitted to the
Faculty of Mathematics, Computer Science and Natural Science of
RWTH Aachen University

July 2012

Supervisor: Prof. Dr. Dr.h.c. Wolfgang Thomas
Second examiner: PD Dr. Christof Löding

written at the
Chair of Computer Science 7
Logic and Theory of Discrete Systems
Prof. Dr. Dr.h.c. Wolfgang Thomas

Erklärung

Hiemit versichere ich, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie Zitate kenntlich gemacht habe.

Aachen, den 28. Juni 2012

Contents

1 Introduction **4**

2 Background results on regular ω-languages **5**

2.1 Preliminaries . . . 5

2.2 The class of regular $*$-languages . . . 5

2.3 The class of regular ω-languages . . . 6

2.3.1 ω regular expressions . . . 6

2.3.2 ω-automata . . . 7

2.3.3 Language operators . . . 8

2.3.4 Logic on infinite words . . . 8

2.4 Language Operators: Transformation of $*$-languages to ω-languages . . . 9

2.5 Classification of regular ω-languages . . . 10

2.6 Towards a theory for subclasses of the regular language class . . . 12

3 General results **13**

3.1 Background . . . 13

3.2 Classification for arbitrary language classes . . . 15

3.3 Congruenced based language classes . . . 21

3.3.1 Introduction . . . 21

3.3.2 Classification . . . 22

4 Results on concrete $*$-language classes **26**

4.1 Starfree regular languages . . . 26

4.2 FO[+1] . . . 28

4.3 FO[] . . . 28

4.4 piece-wise testable . . . 28

4.5 locally testable . . . 30

4.6 locally threshold testable . . . 30

4.7 endwise testable . . . 30

4.8 local . . . 30

4.9 finite / co-finite . . . 30

4.10 dot-depth-n . . . 31

4.11 L-trivial . . . 31

4.12 R-trivial . . . 31
4.13 locally modulo testable . . . 31
4.14 context free . . . 31

5 Conclusion **32**

6 References **33**

Chapter 1

Introduction

The study of formal languages and finite-state automata theory is very old and fundamental in theoretical computer science. Regular expressions were introduced by Kleene in 1956 ([Kle56]). Research on the connection between formal languages, automata theory and mathematical logic began in the early 1960's by Büchi ([Büc60]). Good introductions into the theory are [Str94] and [Tho96].

We call languages over finite words the $*$-languages. Likewise, ω-languages are over infinite words.

The class of regular $*$-languages is probably the most well studied language class. Its expressiveness is exactly equivalent to the class of finite-state automata. For many applications, less powerful subsets of the regular $*$-languages are interesting, like star-free $*$-languages, locally testable $*$-languages, etc., as well as more powerful supersets, like context-free $*$-languages.

The research on ω-languages and their connection to finite-state automata began a bit later by Büchi [Büc62] and [Mul63]. As for the $*$-languages, the most well studied ω language class are the regular ω-languages. Good introductions into these theories are [Tho90], [Tho10], [Sta97] and [PP04].

The acceptance-condition in automata for $*$-languages is straight-forward. If we look at ω-languages, several different types of automata and their acceptance have been thought of, like Büchi-acceptance or Muller-acceptance, or E-acceptance and A-acceptance.

For all types, we can also argue with equivalent language-theoretical operators which operate on a $*$-language, like $\lim$ or ext. We will study the equivalences in more detail.

Depending on the $* \to \omega$ language operator or the ω-automaton acceptance condition, we get different ω-language classes. This was studied earlier already in detail for the class of regular $*$-languages. E.g., we get the result $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{reg}) \subsetneq \mathrm{BC}\lim\mathcal{L}^*(\mathrm{reg})$ and $\lim\cap\overline{\lim}\,\mathcal{L}^*(\mathrm{reg}) = \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{reg})$.

When we look at other $*$-language classes and the different ways to transform them into ω-languages, we can get different results. E.g., $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}(\mathrm{PT}) = \mathrm{BC}\lim\mathcal{L}(\mathrm{PT})$. This study is the main topic of this thesis.

Chapter 2

Background results on regular ω-languages

2.1 Preliminaries

We introduce some common terminonoly used in this thesis.

The set of natural numbers $1, 2, 3, \ldots$ is denoted by $\mathbb{N}$, likewise $0, 1, 2, 3, \ldots$ by $\mathbb{N}_0$.

An **alphabet** is a finite set of **symbols**. We usually denote an alphabet by Σ and its elements by $a, b, c, \ldots$. A finite sequence of elements in Σ is also called a **finite word**, often named $u, v, w, \ldots$. The set of such words, including the **empty word** ϵ, is denoted by Σ^*. Likewise, Σ^+ is the set of non-empty words. Infinite sequences over Σ are called **infinite words**, often named α, β. The set of such infinite words is denoted by Σ^ω.

A subset $L \subseteq \Sigma^*$ is called a **language** of finite words or also called a $*$**-language**. Likewise, a subset $L^\omega \subseteq \Sigma^\omega$ is called an ω**-language**.

A set $\mathcal{L}$ of $*$-languages is called a $*$**-language class**. Likewise, a set $\mathcal{L}^\omega$ of ω-languages is called a ω**-language class**.

We can **concatenate** finite words with each other and also finite words with infinite words. For languages $L_1 \subseteq \Sigma^*$, $L_2 \subseteq \Sigma^*$, $L_3^\omega \subseteq \Sigma^\omega$, we define the concatenation $L_1 \cdot L_2 := \{v \cdot w \mid v \in L_1, w \in L_2\}$ and $L_1 \cdot L_3^\omega := \{v \cdot \alpha \mid v \in L_1, \alpha \in L_3^\omega\}$. Exponentation of languages is defined naturally: For $L \subseteq \Sigma^*$, we define $L^0 := \{\epsilon\}$ and $L^{i+1} := L^i \cdot L$ for all $i \in \mathbb{N}_0$. The union of all such sets, is called the **Kleene closure** or the **Kleene star** operator, defined as $L^* := \cup_{i \in \mathbb{N}_0} L^i$. The **positive Kleene closure** is defined as $L^+ := \cup_{i \in \mathbb{N}} L^i$. The **infinite Kleene closure** is defined by $L^\omega := \{w_1 \cdot w_2 \cdot w_3 \cdots \mid w_i \in L\}$.

2.2 The class of regular $*$-languages

A **regular expression** is representing a language over an alphabet Σ. Regular expressions are defined recursively based on the ground terms $\emptyset$, ϵ and a for $a \in \Sigma$ denoting the languages $\emptyset$, $\{\epsilon\}$ and $\{a\}$. Then, if r and s are regular expressions representing $R, S \subseteq \Sigma^*$, then also $r + s$ (written also as $r|s$, $r \vee s$, $r \cup s$), rs (written also as $r \cdot s$) and r^* are regular expressions, representing $R \cup S$, $R \cdot S$ and R^*. Let $\mathcal{L}^*(\text{RE})$ be the set of languages which can be represented as regular expressions.

We extend these expression also by $r \wedge s$ (written also as $r \cap s$) and $-r$ (written also as $\neg r$), representing the language $R \cap S$ and $-R := \{w \in \Sigma^* \mid w \notin R\}$. Some basic result of the study of formal languages, as can be seen in e.g. [Str94], is the equivalence of the class of these extended regular expression languages and $\mathcal{L}^*(\text{RE})$.

A **non-deterministic finite-state automaton** $\mathcal{A}$ over an alphabet Σ is given by a finite set Q of **states** and a subset $\Delta \subseteq Q \times \Sigma^* \times Q$ of **transitions**. In most cases we also have an **initial states** $q_0 \in Q$ and a subset $F \subseteq Q$ of **final states**.

We write:

$$\mathcal{A} = (Q, \Sigma, q_0, \Delta, F).$$

The automaton is **deterministic** (a DFA) iff Δ is a function $Q \times \Sigma \to Q$. In that case, we often call the function δ.

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

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

$$q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \ldots$$

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

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

The set of $*$-languages accepted by a NFA is called $\mathcal{L}^*(\text{NFA})$. Likewise, $\mathcal{L}^*(\text{DFA})$ is the set of $*$-languages accepted by a DFA. A basic result (see for example [Str94] or [PP04]) is

$$\mathcal{L}^*(\text{DFA}) = \mathcal{L}^*(\text{NFA}) = \mathcal{L}^*(\text{RE}).$$

This class of $*$-languages is called the class of **regular $*$-languages**. We name it $\mathcal{L}^*(\text{reg})$ from now on.

2.3 The class of regular ω-languages

The class of regular ω-languages can be defined in many different ways. We will use one common definition and show some equivalent descriptions.

$$\mathcal{L}^\omega(\text{reg}) := \left\{ \bigcup_{i=1}^{n} U_i \cdot V_i^\omega \,\middle|\, U_i, V_i \in \mathcal{L}^*(\text{reg}), n \in \mathbb{N}_0 \right\}$$

This is also called the **Kleene closure**.

2.3.1 ω regular expressions

For a regular expression r representing a $*$-language $R \subseteq \Sigma^*$, we can introduce a corresponding ω regular expression r^ω which represents the ω-language R^ω. This ω regular expression can be combined with other ω regular expressions as usual and prefixed by standard regular expressions. We call all these combinations ω regular expressions.

We see that $\mathcal{L}^\omega(\text{reg})$ is closed under union (obviously), intersection and complement.

Thus, the class of languages accepted by ω regular expressions is exactly $\mathcal{L}^\omega(\text{reg})$.

2.3.2 ω-automata

A different, very common description is in terms of automata.

An automaton $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ **Büchi-accepts** an infinite word $\alpha = (a_0, a_1, a_2, ...) \in \Sigma^\omega$ iff there is an infinite run $q_0 \xrightarrow{a_0} q_1 \xrightarrow{a_1} q_2 \xrightarrow{a_2} q_3 \cdots$ in $\mathcal{A}$ with $\{i \in \mathbb{N}_0 \mid q_i \in F\}$ 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 $\mathcal{A}$. To make clear that we use the Büchi acceptance condition, we sometimes will also write $L^\omega_{\text{Büchi}}(\mathcal{A})$.

A basic result of the study of this language class is: The set of all languages accepted by a non-deterministic Büchi automaton is exactly $\mathcal{L}^\omega(\text{reg})$ (see [Tho10] or others). Deterministic Büchi automata are less powerful, e.g. they cannot recognise $(a+b)^* b^\omega$.

There are some different forms of ω-automata which differ in their acceptance condition. Noteable are the **Muller conition**, the **Rabin condition**, the **Streett condition** and the **Parity condition**. With such an acceptance condition, we call it **Muller automaton**, etc. The main theorem of deterministic ω-automata states:

- Non-deterministic Büchi automata,
- a boolean combination of deterministic Büchi automata,
- deterministic Muller automata,
- deterministic Rabin automata,
- deterministic Streett automata,
- deterministic Parity automata

all recognize the same languages. See [Tho10], [Tho96], [PP04] and others. The main part of this theorem is the **McNaughton's Theorem** which states the equivalence of non-deterministic Büchi automata and deterministic Muller automata.

Muller automata are interesting for us in the rest of this thesis. The acceptance component of a Muller automaton is a set $\mathcal{F} \subseteq 2^Q$, also called the **table** of the automaton (instead of a single set $F \subseteq Q$). A word $w \in \Sigma^\omega$ is accepted iff there is an infinite run p with $\text{Inf}(p) \in \mathcal{F}$, where $\text{Inf}(p)$ is the set of infinitely often reached states of the run p.

We write:

$$\mathcal{A} = (Q, \Sigma, q_0, \Delta, \mathcal{F}).$$

2.3.3 Language operators

Büchi acceptance is closely connected to the language operator

$$\lim(L) := \{\alpha \in \Sigma^\omega \mid \exists^\omega n\colon \alpha[0,n] \in L\}\,.$$

We define the language class operator

$$\lim(\mathcal{L}) := \{\lim(L) \mid L \in \mathcal{L}\}\,.$$

We see that $\lim(\mathcal{L}^*(\text{reg}))$ is equal to the languages accepted by deterministic Büchi automata ([Tho10]). Thus:

$$\mathrm{BC}\lim \mathcal{L}^*(\text{reg}) = \mathcal{L}^\omega(\text{reg}),$$

where BC means all boolean combinations (union, intersection, complement).

Another classification is

$$\mathcal{L}^\omega(\text{reg}) = \left\{ \bigcup_{i=0}^{n} U_i \cdot \lim V_i \;\middle|\; U_i, V_i \in \mathcal{L}^*(\text{reg}), n \in \mathbb{N}_0 \right\}.$$

2.3.4 Logic on infinite words

Let $L_2(\Sigma)$ be the set of formulas $\mathrm{MSO}(<)$ over Σ. The interpretation of such formulas over infinite words is straight-forward. In [Tho81], we can see that

$$\mathcal{L}^\omega(\text{reg}) = \{A \subseteq \Sigma^\omega \mid A \text{ definable in } L_2(\Sigma)\}\,.$$

2.4 Language Operators: Transformation of $*$-languages to ω-languages

We already introduced $\lim$. We can define a family of language operators, partly also derived from the study of $\mathcal{L}^\omega(\text{reg})$. Some of these operators operate on a single language and not on the class. Let $\mathcal{L}$ be a $*$-language class. Let $L \in \mathcal{L}$.

1. $\text{ext}(L) := \{\alpha \in \Sigma^\omega \mid \exists n \colon \alpha[0,n] \in L\} = L \cdot \Sigma^\omega$
2. $\overline{\text{ext}}(L) := \{\alpha \in \Sigma^\omega \mid \forall n \colon \alpha[0,n] \in L\}$ (also called the dual-ext)
3. $\text{BC}\,\text{ext}$
4. $\lim(L) := \{\alpha \in \Sigma^\omega \mid \forall N \colon \exists n > N \colon \alpha[0,n] \in L\} = \{\alpha \in \Sigma^\omega \mid \exists^\omega n \colon \alpha[0,n] \in L\}$
5. $\overline{\lim}(L) := \{\alpha \in \Sigma^\omega \mid \exists N \colon \forall n > N \colon \alpha[0,n] \in L\}$ (also called dual-lim)
6. $\text{BC}\,\lim$
7. $\widehat{\text{Kleene}}(\mathcal{L}) := \{\bigcup_{i=1}^n U_i \cdot V_i^\omega \mid U_i, V_i \in \mathcal{L}, n \in \mathbb{N}_0\}$
8. $\widehat{\lim}(\mathcal{L}) := \{\bigcup_{i=1}^n U_i \cdot \lim V_i \mid U_i, V_i \in \mathcal{L}, n \in \mathbb{N}_0\}$

From language operators, we get language class operators in a canonical way, e.g. $\lim(\mathcal{L}) := \{\lim L \mid L \in \mathcal{L}\}$. BC denotes always all boolean combinations (union, intersection, complement) of a language class, i.e. for $\mathcal{X} \subseteq \Sigma^\omega \cup \Sigma^*$, $\text{BC}\,\mathcal{X}$ is defined as the smallest set such that

- $\mathcal{X} \subseteq \text{BC}\,\mathcal{X}$,
- $-X \in \text{BC}\,\mathcal{X}$ for all $X \in \text{BC}\,\mathcal{X}$,
- $X_1 \cup X_2 \in \text{BC}\,\mathcal{X}$ for all $X_1, X_2 \in \text{BC}\,\mathcal{X}$,
- $X_1 \cap X_2 \in \text{BC}\,\mathcal{X}$ for all $X_1, X_2 \in \text{BC}\,\mathcal{X}$.

For ext and $\overline{\text{ext}}$, we can also introduce equivalent ω automata acceptance conditions (as in [Tho10]). Let $L \subseteq \Sigma^*$ be a regular $*$-language and $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ be an automaton which accepts exactly L. Let ρ be an infinte run in $\mathcal{A}$.

- $\mathcal{A}$ **E-accepts** ρ iff $\exists i \colon \rho(i) \in F$,
- $\mathcal{A}$ **A-accepts** ρ iff $\forall i \colon \rho(i) \in F$.

We define

$L_E^\omega(\mathcal{A}) := \{\alpha \in \Sigma^\omega \mid \alpha$ is E-accepted in $\mathcal{A}\}$,

$L_A^\omega(\mathcal{A}) := \{\alpha \in \Sigma^\omega \mid \alpha$ is A-accepted in $\mathcal{A}\}$, and we have the equalities

$$L_E^\omega(\mathcal{A}) = \mathrm{ext}(L),$$
$$L_A^\omega(\mathcal{A}) = \overline{\mathrm{ext}}(L).$$

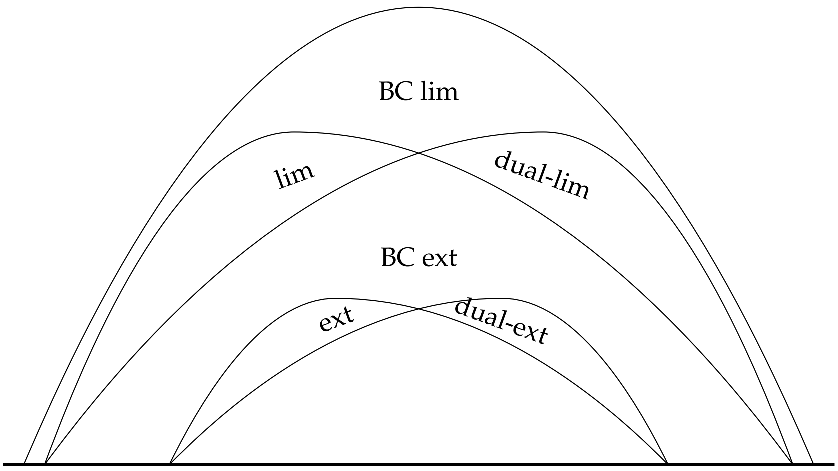
Note that $\mathcal{A}$ can be both deterministic or non-deterministic for this property (see 3.2).

Given these language operators, we are interested in the relations between them. For the class $\mathcal{L}^*(\mathrm{reg})$ of regular languages, we already know that

$$\mathcal{L}^\omega(\mathrm{reg}) = \widehat{\mathrm{Kleene}}(\mathcal{L}^*(\mathrm{reg})) = \mathrm{BC}\lim(\mathcal{L}^*(\mathrm{reg})) = \widehat{\lim}(\mathcal{L}^*(\mathrm{reg})).$$

2.5 Classification of regular ω-languages

Considering $\mathcal{L} := \mathcal{L}^*(\mathrm{reg})$, we get a language diagram like:



where all inclusions are strict. In more detail:

Lemma 2.1. *1* $\mathrm{ext}\,\mathcal{L} \cap \overline{\mathrm{ext}}\,\mathcal{L} \neq \emptyset$

2a. $\mathrm{ext}\,\mathcal{L} \cap \overline{\mathrm{ext}}\,\mathcal{L} \subsetneqq \mathrm{ext}\,\mathcal{L}$

2b. $\mathrm{ext}\,\mathcal{L} \cap \overline{\mathrm{ext}}\,\mathcal{L} \subsetneqq \overline{\mathrm{ext}}\,\mathcal{L}$

3. $\mathrm{ext}\,\mathcal{L} \neq \overline{\mathrm{ext}}\,\mathcal{L}$

4. $\mathrm{ext}\,\mathcal{L} \cup \overline{\mathrm{ext}}\,\mathcal{L} \subsetneqq \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}$

5. $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L} = \lim\mathcal{L} \cap \overline{\lim}\,\mathcal{L}$ *(Staiger-Wagner class)*

6a. $\lim\mathcal{L} \cap \overline{\lim}\,\mathcal{L} \subsetneqq \lim\mathcal{L}$

6b. $\lim \mathcal{L} \cap \overline{\lim} \mathcal{L} \subsetneq \overline{\lim} \mathcal{L}$

7. $\lim \mathcal{L} \neq \overline{\lim} \mathcal{L}$

8. $\lim \mathcal{L} \cup \overline{\lim} \mathcal{L} \subsetneq \mathrm{BC} \lim \mathcal{L}$

and we have the additional properties

9. $\mathrm{BC} \lim \mathcal{L} = \widehat{\mathrm{Kleene}}(\mathcal{L})$

10. $\mathrm{BC} \lim \mathcal{L} = \widehat{\lim}(\mathcal{L})$

11. $\mathrm{BC} \lim \mathcal{L} = \left\{ L^\omega_{\textit{Büchi}}(\mathcal{A}) \,\middle|\, \mathcal{A} \textit{ automaton so that } L^*(\mathcal{A}) \in \mathcal{L} \right\}$

Proof. 1 $\tilde{L}_1 := a\Sigma^\omega \in \mathrm{ext} \cap \overline{\mathrm{ext}}\, \mathcal{L}$ with $\tilde{L}_1 = \mathrm{ext}(a)$ and $\tilde{L}_1 = \overline{\mathrm{ext}}(a\Sigma^*)$. ([Tho10, prop, p.38])

2a. $\tilde{L}_{2a} := \mathrm{ext}(a^*b) = a^*b\Sigma^\omega \in \mathrm{ext}\, \mathcal{L}$. Assume some A-automaton $\mathcal{A}$ with n states accepts $\tilde{L}_{2a}$. $\mathcal{A}$ would also accept $a^n b^\omega$. I.e. the $(n+1)$th state after the run of a^n would also accept a, i.e. $\mathcal{A}$ would accept a^{n+1}. By inclusion, $\mathcal{A}$ would accept a^ω. That is a contradiction. Thus, there is no such A-automat. Thus, $\tilde{L}_{2a} \notin \overline{\mathrm{ext}}\, \mathcal{L}$.

2b. $\tilde{L}_{2b} := -\tilde{L}_{2a} \in \overline{\mathrm{ext}}\, \mathcal{L}$, $\tilde{L}_{2b} \notin \mathrm{ext}\, \mathcal{L}$.

3. Follows directly from P2a and P2b.

4. $\tilde{L}_4 := \Sigma^* a \Sigma^\omega \cap -(\Sigma^* b \Sigma^\omega)$, $\Sigma = \{a, b, c\}$. Then we have $\tilde{L}_4 \notin \mathrm{ext} \cup \overline{\mathrm{ext}}\, \mathcal{L}$, $\tilde{L}_4 \in \mathrm{BC}\, \mathrm{ext}\, \mathcal{L}$. ([Tho10, p.38])

5. A language in this class is also said to have the **obligation property**. Staiger and Wagner have introduced a **Staiger-Wagner automaton** (also called a **weak Muller automaton**; see definition 3.9) which can accept exactly this language class. This class of languages is called the **Staiger-Wagner-recognizable** languages. This is stated in theorem 3.10.

   A generic proof of the equality $\mathrm{BC}\, \mathrm{ext}\, \mathcal{L} = \lim \mathcal{L} \cap \overline{\lim} \mathcal{L}$ is given in lemma 3.11.

6a. $\tilde{L}_{6a} := \lim(\Sigma^* a) = (\Sigma^* a)^\omega$. Assume there is $L \subseteq \Sigma^*$ with $\lim(L) = -\tilde{L}_{6a}$. Let $(w_0, w_1, w_2, \dots) \in (\Sigma^*)^\mathbb{N}$ so that $w_0 \in L, w_0 a w_1 \in L, \dots, w_0 \prod_{i=0}^n a w_i \in L\ \forall n \in \mathbb{N}$. Thus, $\alpha := w_0 \prod_{i \in \mathbb{N}} a w_i \in \lim L$. But $\alpha \notin -\tilde{L}_{6a}$. That is a contradiction. Thus, $-\tilde{L}_{6a} \notin \lim \mathcal{L}$. Because $\mathcal{L}$ is closed under complement, we get $\tilde{L}_{6a} \notin \overline{\lim} \mathcal{L}$.

6b. Analog to 6a with $\tilde{L}_{6b} := -\tilde{L}_{6a}$.

7. Follows directly from 6a and 6b.

8. $\tilde{L}_8 := (\Sigma^* a)^\omega \cap -(\Sigma * b)^\omega$. Then $\tilde{L}_8 \notin \lim \cup \overline{\lim} \mathcal{L}$ but $\tilde{L}_8 \in \mathrm{BC} \lim \mathcal{L}$. ([Tho10, prop, p.38])

9.-11. This is explained already in chapter 2.3 and in more detail in [Tho10] or [Tho81, Theorem 3.1].

□

2.6 Towards a theory for subclasses of the regular language class

This was studied in detail for $\mathcal{L}^*(\mathrm{reg})$. We are now studing relations of resulting ω-language classes for different $*$-language classes.

Esp.:

- $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L} \overset{?}{\subsetneq} \mathrm{BC}\,\lim\,\mathcal{L}$

Chapter 3

General results

Let $\mathcal{L}$ be a $*$-language class. We start with some very basic results on language operators.

3.1 Background

Lemma 3.1. *Let* $L, A, B \in \mathcal{L}$.

1. $\operatorname{ext} L = L \cdot \Sigma^\omega$
2. $\operatorname{ext} L = \lim(L \cdot \Sigma^*)$
3. $\operatorname{ext} L = \overline{\lim}(L \cdot \Sigma^*)$
4. $-\lim(-L) = \overline{\lim}(L)$
5. $\overline{\lim}\, L \subseteq \lim L$
6. $\lim A \cup \lim B = \lim(A \cup B)$
7. $\overline{\lim}\, A \cup \overline{\lim}\, B \subseteq \overline{\lim}(A \cup B)$
   There is no equality in general: $A = (00)^*$, $B = (00)^*0$.

Proof. 1.-5. They all follow directly from the definition.

6.

$$\begin{aligned} & \alpha \in \lim A \cup \lim B \\ \Leftrightarrow\ & \exists^\omega n\colon \alpha[0,n] \in A \ \vee\ \exists^\omega n\colon \alpha[0,n] \in B \\ \Leftrightarrow\ & \exists^\omega n\colon \alpha[0,n] \in A \cup B \\ \Leftrightarrow\ & \alpha \in \lim A \cup B \end{aligned}$$

7.

$$\begin{aligned} & \alpha \in \overline{\lim}\, A \cup \overline{\lim}\, B \\ \Leftrightarrow\ & \exists N\colon \forall n \geq N\colon \alpha[0,n] \in A \ \vee\ \exists N\colon \forall n \geq N\colon \alpha[0,n] \in B \\ \Rightarrow\ & \exists N\colon \forall n \geq N\colon \alpha[0,n] \in A \cup B \end{aligned}$$

$\square$

For ω automata, we already know that non-determinism can be more powerful than determinism (see 2.3): The class of non-deterministic Büchi automata can accept clearly more languages than the class of determinstic Büchi automata. E.g. $L^\omega := (a+b)^* b^\omega \in \mathcal{L}^\omega(\text{reg})$ cannot be recognised by deterministic Büchi automata, i.e. $L^\omega \notin \lim \mathcal{L}^*(\text{reg})$.

Luckily, for E- and A-acceptance, this is not the case as we see below. This matches the intuition that E/A-acceptance doesn't really tell something about infinite properties of words but Büchi/Muller does. And when talking about finite words, we already know that non-deterministic and deterministic automata are equaly powerful (see chapter 2.2).

Lemma 3.2. *The ω-language-class accepted by deterministic E-automata is equal to non-deterministic E-automata. I.e., for every non-deterministic E-automaton, we can construct an equivalent deterministic E-automaton. The same goes for A-automata.*

Proof. Let $\mathcal{A}^N$ be any non-deterministic automaton and $\mathcal{A}^D$ an ($*$-)equivalent deterministic automaton. Then:

$$\begin{aligned} & \alpha \in L_E^\omega(\mathcal{A}^N) \\ \Leftrightarrow\ & \exists n\colon \alpha[0,n] \in L(\mathcal{A}^N) \\ \Leftrightarrow\ & \exists n\colon \alpha[0,n] \in L(\mathcal{A}^D) \\ \Leftrightarrow\ & \alpha \in L_E^\omega(\mathcal{A}^D) \end{aligned}$$

$\mathcal{A}^N$ can be interpreted as an arbitary E-automata and we have shown that we get an equivalent deterministic E-automata.

For A-automata, the proof is analogue. $\square$

We are interested in relations like $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L} \overset{?}{\subsetneqq} \mathrm{BC}\lim \mathcal{L}$ or $\mathrm{ext}\,\mathcal{L} \overset{?}{\subsetneqq} \lim \mathcal{L}$. With $\mathcal{L} = \{\{a\}\}$, we realize that even $\mathrm{ext}\,\mathcal{L} \subseteq \lim \mathcal{L}$ is not true in general ($\mathrm{ext}\,\{\{a\}\} = \{a\Sigma^\omega\} \neq \emptyset = \lim \{\{a\}\}$). In lemma 3.3, we see a sufficient condition for this property, though.

We want to study all the properties we have shown for $\mathcal{L}^*(\text{reg})$ in lemma 2.1.

We will formulate some properties of interest in a general form for a $*$-language class $\mathcal{L}$ which all hold for $\mathcal{L}^*(\text{reg})$. We get some general results based on these properties in chapter 3.

Let $L, A, B \in \mathcal{L}$.

1. **Closure under suffix-insensitiveness**: $L \cdot \Sigma^* \in \mathcal{L}$

2a. **Closure under union**: $A \cup B \in \mathcal{L}$

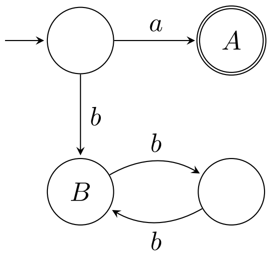
2b. **Closure under intersection**: $A \cap B \in \mathcal{L}$

3. **Closure under complementation/negation**: $-L \in \mathcal{L}$
4. **Closure under acceptance component variation**: In some proofs, e.g. in 3.11 or 3.12, we have an automaton based on some language of the language class and we do some modifications on it, e.g. we modify the acceptance component. If there is a way to stay in the language class, the class is called to be **closed under acceptance component variation**. Formally:

   There is a deterministic automaton $\mathcal{A} = (Q, \Sigma, q_0, \delta, F)$ with $L^*(\mathcal{A}) = L$ such that for all $F' \subseteq Q$, we have $L^*((Q, \Sigma, q_0, \delta, F')) \in \mathcal{L}$.

   For $\mathcal{L}^*(\text{reg})$, this property holds obviously.

   Note that we cannot just take any automaton. For $\mathcal{L}^*(\text{FO}[<])$ and the automaton below, it does not hold:

   This is a deterministic automaton for the language $\{a\} \in \mathcal{L}^*(\text{FO}[<])$. If you make B also a final state, we get the language $a + b(bb)^* \notin \mathcal{L}^*(\text{FO}[<])$.

3.2 Classification for arbitrary language classes

Lemma 3.3. *If $\mathcal{L}$ is closed under suffix-insensitiveness,*

$$\operatorname{ext} \mathcal{L} \subseteq \lim \mathcal{L} \cup \overline{\lim}\, \mathcal{L}.$$

Proof. For $L \in \mathcal{L}$, we have $\operatorname{ext} L = \lim L\Sigma^* = \overline{\lim}\, L\Sigma^*$. □

Lemma 3.4. *If we have* $\operatorname{ext} \mathcal{L} \subseteq \lim \mathcal{L}$*, then we also have*

$$\operatorname{BC} \operatorname{ext} \mathcal{L} \subseteq \operatorname{BC} \lim \mathcal{L}.$$

Proof. From $\operatorname{ext} \mathcal{L} \subseteq \lim \mathcal{L}$, it directly follows $\{- \operatorname{ext} L \mid L \in \mathcal{L}\} \subseteq \{- \lim L \mid L \in \mathcal{L}\}$. Thus, it also follows the claimed inequality. □

Lemma 3.5. *If we have* $\operatorname{ext} \mathcal{L} \subseteq \lim \mathcal{L}$ *and let $\mathcal{L}$ be closed under negation. Then we have*

$$\overline{\operatorname{ext}}\, \mathcal{L} \subseteq \overline{\lim}\, \mathcal{L}.$$

Proof. Let $L \in \mathcal{L}$. Then $\overline{\operatorname{ext}}\, L = -\operatorname{ext}(-L)$. Because of the negation closure, we also have $-L \in \mathcal{L}$ and $\operatorname{ext}(-L) \in \operatorname{ext} \mathcal{L}$. Thus $\lim(-L) \in \lim \mathcal{L}$. Thus, $-\lim(-L) \in \{-\lim A \mid A \in \mathcal{L}\} = \left\{\overline{\lim}\, A \,\middle|\, -A \in \mathcal{L}\right\}$. Because of the negation closure, this is equal to $\left\{\overline{\lim}\, A \,\middle|\, A \in \mathcal{L}\right\} = \overline{\lim}\, \mathcal{L}$. I.e. $\overline{\lim}\, L \in \overline{\lim}\, \mathcal{L}$. □

Note that we needed the negation closure in the proof. This is in contrast to 3.4, where it directly follows. We have to be careful about the difference $-\operatorname{ext}\mathcal{L} := \{-\operatorname{ext} L \mid L \in \mathcal{L}\} \neq \overline{\operatorname{ext}}\,\mathcal{L}$ (in general, if $\mathcal{L}$ is not closed under negation).

Lemma 3.6.
- *$\mathcal{L}$ closed under union $\Rightarrow \bigcup \operatorname{ext}\mathcal{L} \subseteq \operatorname{ext}\mathcal{L}$.*
- *$\mathcal{L}$ closed under intersection $\Rightarrow \bigcap \operatorname{ext}\mathcal{L} \subseteq \operatorname{ext}\mathcal{L}$.*

Proof. Let $A, B \in \mathcal{L}$. Then we have $\operatorname{ext}(A \cup B) = \operatorname{ext}(A) \cup \operatorname{ext}(B)$ and $\operatorname{ext}(A \cap B) = \operatorname{ext}(A) \cap \operatorname{ext}(B)$. □

Lemma 3.7. *Let $\mathcal{L}$ be closed under negation. Then*

$$\operatorname{ext} \cup \overline{\operatorname{ext}}\,\mathcal{L} \subsetneq \mathrm{BC}\operatorname{ext}\mathcal{L}.$$

Proof. If there is $L_\Sigma \in \mathcal{L}_\Sigma$ with $L_\Sigma \in \operatorname{ext}\mathcal{L}_\Sigma, L_\Sigma \notin \overline{\operatorname{ext}}\,\mathcal{L}_\Sigma$ and E3 (closed under negation) holds for $\mathcal{L}$:

$$\Rightarrow\ -L_\Sigma \in \overline{\operatorname{ext}}\,\mathcal{L}_\Sigma, -L_\Sigma \notin \operatorname{ext}\mathcal{L}_\Sigma$$

$$\Rightarrow\ L_{\Sigma_1} \cup -L_{\Sigma_2} \in \mathrm{BC}\operatorname{ext}\mathcal{L}_{\Sigma_1 \dot\cup \Sigma_2}$$
$$L_{\Sigma_1} \cup -L_{\Sigma_2} \notin \operatorname{ext} \cup \overline{\operatorname{ext}}\,\mathcal{L}_{\Sigma_1 \dot\cup \Sigma_2}$$

Thus, $\operatorname{ext} \cup \overline{\operatorname{ext}}\,\mathcal{L} \subsetneq \mathrm{BC}\operatorname{ext}\mathcal{L}$. □

Similarily:

Lemma 3.8. *Let $\mathcal{L}$ be closed under negation. Then*

$$\lim \cup \overline{\lim}\,\mathcal{L} \subsetneq \mathrm{BC}\lim\mathcal{L}.$$

Proof. If there is $L_\Sigma \in \lim\mathcal{L}_\Sigma, L_\Sigma \notin \overline{\lim}\,\mathcal{L}_\Sigma$ and E3 holds for $\mathcal{L}$:

$$\Rightarrow\ -L_\Sigma \in \overline{\lim}\,\mathcal{L}_\Sigma, -L_\Sigma \notin \lim\mathcal{L}_\Sigma$$

$$\Rightarrow\ L_{\Sigma_1} \cup -L_{\Sigma_2} \in \mathrm{BC}\lim\mathcal{L}_{\Sigma_1 \dot\cup \Sigma_2}$$
$$L_{\Sigma_1} \cup -L_{\Sigma_2} \notin \lim \cup \overline{\lim}\,\mathcal{L}_{\Sigma_1 \dot\cup \Sigma_2}$$

Thus, $\lim \cup \overline{\lim}\,\mathcal{L} \subsetneq \mathrm{BC}\lim\mathcal{L}$. □

Definition 3.9. A **Staiger-Wagner automaton** (also called **weak Muller automaton**) is of the same form $\mathcal{A} = (Q, \Sigma, q_0, \delta, \mathcal{F})$ with acceptance component $\mathcal{F} \subseteq 2^Q$ like a Muller automaton with the acceptance condition that a run ρ in $\mathcal{A}$ is accepting if and only if $\mathrm{Occ}(\rho) := \{q \in Q \mid q \text{ occurs in } \rho\} \in \mathcal{F}$. ([Tho10, Def.61, p.43])

Theorem 3.10. *We see that the class of Staigner-Wagner-recognized languages is exactly the class $\mathrm{BC}\operatorname{ext}\mathcal{L}^*(reg)$ and also $\lim \cap \overline{\lim}\,\mathcal{L}^*(reg)$. And thus:*

$$\mathrm{BC}\operatorname{ext}\mathcal{L}^*(reg) = \lim \cap \overline{\lim}\,\mathcal{L}^*(reg).$$

Proof. See [Tho10, Theorem 63+64, p.44]. □

We are now formulating a more general and direct proof for the $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L} = \lim \cap \overline{\lim}\,\mathcal{L}$ equality without Staiger-Wagner-automata (where some of the ideas are loosely based on [Tho10, Theorem 63+64, p.44]).

Lemma 3.11. *Let $\mathcal{L}$ be closed under suffix-insensitiveness, union, negation and acceptance component variation. Then*

$$\mathrm{BC}\,\mathrm{ext}\,\mathcal{L} = \lim \cap \overline{\lim}\,\mathcal{L}.$$

Proof. First, we show $\lim \cap \overline{\lim}\,\mathcal{L} \subseteq \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}$.

Let $\tilde{L} \in \lim \cap \overline{\lim}\,\mathcal{L}$, i.e. there are deterministic automaton $\mathcal{A}$ and $\overline{\mathcal{A}}$ so that $L^{\omega}_{\text{Büchi}}(\mathcal{A}) = L^{\omega}_{\text{co-Büchi}}(\overline{\mathcal{A}}) = \tilde{L}$. Let $Q, \overline{Q}$ be the states of $\mathcal{A}, \overline{\mathcal{A}}$. Now look at the product automaton $\mathcal{A} \times \overline{\mathcal{A}} =: \overset{\times}{\mathcal{A}}$ with states $Q \times \overline{Q}$ and final states $F \times \overline{F} \subseteq Q \times \overline{Q}$. $\overset{\times}{\mathcal{A}}$ is also deterministic.

In $\overset{\times}{\mathcal{A}}$, we have

$$\begin{aligned} & \alpha \in \tilde{L} \\ \Leftrightarrow\ & \forall N\colon \exists n \geq N\colon \overset{\times}{\rho}(\alpha)[n] \in F \times \overline{Q} \\ \Leftrightarrow\ & \exists N\colon \forall n \geq N\colon \overset{\times}{\rho}(\alpha)[n] \in Q \times \overline{F} \end{aligned}$$

Look at strongly connected component (SCC) S in $\overset{\times}{\mathcal{A}}$. We have $S \cap F \times \overline{Q} \neq \emptyset$, iff S accepts. It follows that all states in S are finite states in $\overline{\mathcal{A}}$, i.e. $S \cap Q \times \overline{F} = S$.

Single $\overset{\times}{q} \in \overset{\times}{Q}$ which are not part of a SCC can be ignored. For the acceptance of infinte words, only SCCs are relevant. For S, define $S_+ := \left\{ \overset{\times}{q} \in \overset{\times}{Q} - S \,\middle|\, \overset{\times}{q} \text{ can be visited after } S \right\}$.

Then we have

$$\begin{aligned} \tilde{L} = \bigcup_{\text{SCC } S} & S \text{ will be visited} \wedge \\ & \text{all states of } S \text{ will be visited forever after some step} \wedge \\ & S_+ \text{ will not be visited.} \end{aligned}$$

S will be visited: Let S exactly be the finite states. This interpreted as an E-automaton $\mathcal{A}^E_S$ is exactly the condition.

Only the allowed states will be visited but nothing followed after S: Mark S and all states on all paths to S as finite states. This as an A-automaton $\mathcal{A}^A_S$ is exactly the condition.

A similar negated condition might be simpler: Let S_+ be exactly the finite states. Interpret this as an E-automaton $\mathcal{A}^E_{S_+}$.

Then we have

$$\begin{aligned}\tilde{L} &= \bigcup_{\text{SCC } S} L^\omega_E(\mathcal{A}^E_S) \cap L^\omega_A(\mathcal{A}^A_S) \\ &= \bigcup_{\text{SCC } S} L^\omega_E(\mathcal{A}^E_S) \cap -L^\omega_E(\mathcal{A}^E_{S_+}).\end{aligned}$$

Thus,

$$\tilde{L} \in \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\text{reg}).$$

Given the *closure under acceptance component variation*, we have $L^*(\mathcal{A}^E_S), L^*(\mathcal{A}^E_{S_+}) \in \mathcal{L}$.

Now let us show $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L} \subseteq \lim \mathcal{L}$.

With the *closure under suffix-insensitiveness*, we get $\mathrm{ext}\,\mathcal{L} \subseteq \lim \mathcal{L}$ and $\mathrm{ext}\,\mathcal{L} \subseteq \overline{\lim}\,\mathcal{L}$. I.e. $\mathrm{ext}\,\mathcal{L} \subseteq \lim \cap \overline{\lim}\,\mathcal{L}$. Let us show that $\lim \cap \overline{\lim}\,\mathcal{L}$ is closed under boolean closure.

Let $\tilde{L}_a, L_b \in \lim \cap \overline{\lim}\,\mathcal{L}$, i.e. $\exists L_{a1}, L_{a2}, L_{b1}, L_{b2} \in \mathcal{L}$: $\tilde{L}_a = \lim L_{a1} = \overline{\lim}\,L_{a2}$, $\tilde{L}_b = \lim L_{b1} = \overline{\lim}\,L_{b2}$. Let us show 1. $-\tilde{L}_a \in \lim \cap \overline{\lim}\,\mathcal{L}$, 2. $\tilde{L}_a \cup \tilde{L}_b \in \lim \cap \overline{\lim}\,\mathcal{L}$.

1. $-\tilde{L}_a = -\lim L_{a1} = \overline{\lim} -L_{a1}$, $-\tilde{L}_b = -\overline{\lim}\,L_{a2} = \lim -L_{a2}$. With the *closure under negation*, we get

   $$-\tilde{L}_a \in \lim \cap \overline{\lim}\,\mathcal{L}.$$

2. $\tilde{L}_a \cup \tilde{L}_b = \lim L_{a1} \cup \lim L_{b1} = \lim L_{a1} \cup L_{b1}$ (3.1). Thus, with *closure under union*, we have

   $$\tilde{L}_a \cup \tilde{L}_b \in \lim \mathcal{L}.$$

   The $\overline{\lim}\,\mathcal{L}$ case is harder. Let $\mathcal{A}_a$, $\mathcal{A}_b$ be deterministic automaton, so that $L^\omega_{\text{Büchi}}(\mathcal{A}_a) = L^\omega_{\text{co-Büchi}}(\mathcal{A}_a) = \tilde{L}_a$, $L^\omega_{\text{Büchi}}(\mathcal{A}_b) = L^\omega_{\text{co-Büchi}}(\mathcal{A}_b) = \tilde{L}_b$. Look at the product automaton $\mathcal{A}_a \times \mathcal{A}_b =: \overset{\times}{\mathcal{A}}$. Then we have $L^\omega_{\text{Büchi}}(\overset{\times}{\mathcal{A}}) = L^\omega_{\text{co-Büchi}}(\overset{\times}{\mathcal{A}}) = \tilde{L}_a \cup \tilde{L}_b$.
   Thus,

   $$\tilde{L}_a \cup \tilde{L}_b \in \overline{\lim}\,\mathcal{L}^*(\text{reg}).$$

   Again, given the *closure under acceptance component variation*, we have $L^*(\overset{\times}{\mathcal{A}}) \in \mathcal{L}$.

$\square$

Lemma 3.12. *Let $\mathcal{L}$ be closed under acceptance component variation. Then*

$$\mathrm{BC}\lim \mathcal{L} = \widehat{\mathrm{Kleene}}\, \mathcal{L}.$$

Proof. Let $U, V \in \mathcal{L}$. Look at the non-deterministic automaton $\mathcal{A}$ defined as:

$$\begin{tikzcd} {} \arrow[r] & U \arrow[r, "\epsilon"] & V\odot \arrow[loop above] \end{tikzcd}$$

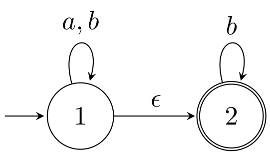
Then we have $L^{\omega}_{\text{Büchi}}(\mathcal{A}) = U \cdot V^{\omega}$.

Let us construct deterministic automata for $\mathcal{A}$ so that we can formulate 'V will be visited and not be left anymore' and 'finite states of the V-related automaton will be visited infinitely often' (or 'UV^* will be visited infinitely often').

In a constructed automaton, we must be able to tell wether we are in U or we deterministically have been in U the previous state. In a state power set construction, we can tell wether we are deterministically in U or not. If we are non-determinstic and we may be in both U or V and we get an input symbol which determines that we have been in U, we might not be able to tell from the following power set.

Example:

Let $U = (a+b)^*$, $V = \{b\}$. I.e. $UV^{\omega} = \{\alpha \in \{a,b\}^{\omega} \mid \text{at one point in } \alpha\text{, there are only } b\text{s}\}$. The non-deterministic automaton is:



Powerset construction: The initial state is $\{1,2\}$. Then we have:

- $\{1,2\} \xrightarrow{a} \{1,2\}$
- $\{1,2\} \xrightarrow{b} \{1,2\}$

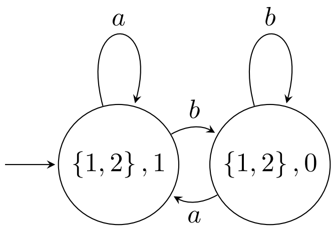
This gives the $*$-language $\{a,b\}^*$ and we cannot formulate UV^{ω} in any way from there.

In the construction, when we got the a from $\{1,2\}$, we knew that we have been deterministically in 1, i.e. in U. We loose this information. To keep it, we introduce another state flag which exactly says wether we have determined that we have been in U. Thus, we construct an automaton with the states $\mathcal{P}(Q) \times \mathbb{B}_{\text{det. been in } U}$, where Q are the states from $\mathcal{A}$.

For the example, we get the initial state $(\{1,2\}, 1)$. Then we have:

- $(\{1,2\},1) \xrightarrow{a} (\{1,2\},1)$
- $(\{1,2\},1) \xrightarrow{b} (\{1,2\},0)$
- $(\{1,2\},0) \xrightarrow{a} (\{1,2\},1)$
- $(\{1,2\},0) \xrightarrow{b} (\{1,2\},0)$

This is the automaton



When we mark all states from V and where we have not been deterministically in U as final, this as a co-Büchi automaton gives exactly the condition 'V will be visited and not be left anymore'. Let L_E be the $*$-language of this automata. Note that $L_E \neq UV^*$ in general and esp. in the example.

When we mark the final states as in the original non-deterministic automata, no matter about $\mathbb{B}_{\text{det. been in } U}$, with Büchi-acceptance, we get the condition 'UV^* will be visited infinitely often'. This is just $\lim UV^*$.

Together, we get UV^ω, i.e.:

$$\lim UV^* \cap \overline{\lim} L_E = UV^\omega$$

Given the *closure under acceptance component variation*, we have $L_E \in \mathcal{L}$. Then, it follows

$$\left\{ \bigcup_{i=1}^{n} U_i \cdot V_i^\omega \,\middle|\, U_i, V_i \in \mathcal{L} \right\} = \widehat{\text{Kleene}}\,\mathcal{L} \subseteq \text{BC}\lim \mathcal{L}.$$

We also need to show $\text{BC}\lim \mathcal{L} \subseteq \widehat{\text{Kleene}}\,\mathcal{L}$.

Show: $\lim \mathcal{L} \subseteq \widehat{\text{Kleene}}\,\mathcal{L}$.

Proof: Let $\mathcal{A}$ be a deterministic Büchi automaton for some language $\tilde{L} = L^\omega_{\text{Büchi}}(\mathcal{A}) \in \mathcal{L}$ with final states F.

For all finite states $q \in F$: If q is not part of a strongly connected component (SCC), we can ignore it. Let S be the SCC where $q \in S$. Then the set of all $\alpha \in \Sigma^\omega$ which are infinitely often in q can be described as $U_q \cdot V_q^\omega$, where U_q is the set of words so that we arrive in q and V_q is the set of words so that we get from q to q. Both sets are regular.

Thus,

$$\tilde{L} = L^{\omega}_{\text{Büchi}}(\mathcal{A}) = \bigcup_{q \in F} U_q V_q^{\omega}.$$

Obviously, the Kleene-Closure is closed under union.

TODO: Show that Kleene-Closure is closed under negation. (S306.5) (Follows with non-det Büchi complementation but a more generic proof might be useful.) □

3.3 Congruenced based language classes

3.3.1 Introduction

Definition 3.13. We define $\mathcal{L}(R)$ for an equivalence relation $R \subseteq \Sigma^* \times \Sigma^*$

$$\mathcal{L}(R) := \{L \subseteq \Sigma^* \mid L \text{ is finite union of } R\text{-equivalence-classes}\}.$$

Examples of such language classes are locally testable (LT, section 4.5), locally threshold testable (LTT, section 4.6) or piece-wise testable (PT, section 4.4) languages. At their definition, the word-relation basically tells wether a local test / piece-wise test can see a difference between two words.

If a language class $\mathcal{L}(R)$ is defined as finite union of equivalence classes of a relation $R \subseteq \Sigma^* \times \Sigma^*$ and

- the set of equivalent classes of R is finite,
- R is a congruence relation, i.e. also $(v, w) \in R \Leftrightarrow (va, wa) \in R \;\forall a \in \Sigma$

then we can construct a canonical deterministic automaton $\mathcal{A}_R$ which has $S_R := \Sigma^*/R$ as states, $\langle \epsilon \rangle_R$ is the initial state and the transitions are according to concatenation. Call this an R-automaton.

The LT, LTT and PT language classes have the above properties and thus such related canonical automaton.

The set of all such R-automata, varying in the final state set, is isomprophic to $\mathcal{L}(R)$. We have

$$\mathcal{L}(R) = \{L(\mathcal{A}_R(F)) \mid F \subseteq S_R\} =: \mathcal{L}^*(\mathcal{A}_R).$$

Definition 3.14. Analogously for ω, we get the set of R-E-automata with the ω-language-class

$$\mathcal{L}^{\omega}_E(\mathcal{A}_R) := \left\{L^{\omega}(\mathcal{A}^E_R(F)) \,\middle|\, F \subseteq S_R\right\},$$

R-Büchi-automata and

$$\mathcal{L}^\omega_{\text{Büchi}}(\mathcal{A}_R) := \left\{ L^\omega(\mathcal{A}_R^{\text{Büchi}}(F)) \;\middle|\; F \subseteq S_R \right\},$$

R-Muller-automata and

$$\mathcal{L}^\omega_{\text{Muller}}(\mathcal{A}_R) := \left\{ L^\omega(\mathcal{A}_R^{\text{Muller}}(\mathcal{F})) \;\middle|\; \mathcal{F} \subseteq 2^{S_R} \right\}.$$

Definition 3.15. For a relation R on Σ^*, there are various ways to construct a relation on Σ^ω. For now, we mainly study $R^\omega := \overline{\text{ext}}\, R$, i.e.

$$(\alpha, \beta) \in R^\omega \;:\Leftrightarrow\; \forall n\colon (\alpha[0,n], \beta[0,n]) \in R.$$

Analogously to $\mathcal{L}(R)$, define the ω-language-class

$$\mathcal{L}^\omega(R^\omega) := \{L^\omega \subseteq \Sigma^\omega \mid L^\omega \text{ is finite union of } R^\omega\text{-equivalence-classes}\}.$$

3.3.2 Classification

With this preparation, we show for some R the equalities:

- $\mathcal{L}^\omega_E(\mathcal{A}_R) = \text{ext}\,\mathcal{L}(R)$ (3.16)
- $\mathcal{L}^\omega_{\text{Büchi}}(\mathcal{A}_R) = \lim \mathcal{L}(R)$ (3.17)
- $\mathcal{L}^\omega_{\text{Muller}}(\mathcal{A}_R) = \text{BC} \lim \mathcal{L}(R)$ (3.18)
- $\mathcal{L}^\omega(R^\omega) = \text{BC}\,\text{ext}\,\mathcal{L}(R)$
- $\text{BC} \lim \mathcal{L}(R) \cap \text{ext}\,\mathcal{L}^*(\text{reg}) = \text{ext}\,\mathcal{L}(R)$
- $\text{BC} \lim \mathcal{L}(R) \cap \lim \mathcal{L}^*(\text{reg}) = \lim \mathcal{L}(R)$
- $\lim \mathcal{L}(R) \cap \overline{\lim}\, \mathcal{L}(R) = \text{BC}\,\text{ext}\,\mathcal{L}(R)$

We will see that all those equations hold for $\mathcal{L}(LT)$, $\mathcal{L}(LTT)$ and $\mathcal{L}(PT)$.

Lemma 3.16.

$$\mathcal{L}^\omega_E(\mathcal{A}_R) = \text{ext}\,\mathcal{L}(R)$$

Proof. Let $L = \bigcup_i \langle w_i \rangle_R$, $L \in \mathcal{L}(R)$. Then

$$\begin{aligned}
& L^\omega = \text{ext}\, L \\
\Leftrightarrow\; & L^\omega = \left\{ \alpha \in \Sigma^\omega \;\middle|\; \exists n\colon \alpha[0,n] \in \bigcup_i \langle w_i \rangle_R \right\} \\
\Leftrightarrow\; & L^\omega = \{\alpha \in \Sigma^\omega \mid \exists n\colon \delta_{\mathcal{A}_R}(\alpha[0,n]) \in \{\langle w_i \rangle_R \subseteq S_R \mid i\}\} \\
\Leftrightarrow\; & L^\omega = L^\omega(\mathcal{A}_R^E(\{\langle w_i \rangle_R \subseteq S_R \mid i\}))
\end{aligned}$$

□

Lemma 3.17.

$$\mathcal{L}^{\omega}_{\textit{Büchi}}(\mathcal{A}_R) = \lim \mathcal{L}(R)$$

Proof. Let $L = \bigcup_i \langle w_i \rangle_R$, $L \in \mathcal{L}(R)$. Then

$$\begin{aligned}
&L^{\omega} = \lim L \\
\Leftrightarrow\ &L^{\omega} = \left\{ \alpha \in \Sigma^{\omega} \,\middle|\, \exists^{\infty} n \colon \alpha[0,n] \in \bigcup_i \langle w_i \rangle_R \right\} \\
\Leftrightarrow\ &L^{\omega} = \{\alpha \in \Sigma^{\omega} \mid \exists^{\infty} n \colon \delta_{\mathcal{A}_R}(\alpha[0,n]) \in \{\langle w_i \rangle_R \subseteq S_R \mid i\}\} \\
\Leftrightarrow\ &L^{\omega} = L^{\omega}(\mathcal{A}_R^{\text{Büchi}}(\{\langle w_i \rangle_R \subseteq S_R \mid i\}))
\end{aligned}$$

□

Lemma 3.18.

$$\mathcal{L}^{\omega}_{\textit{Muller}}(\mathcal{A}_R) = \mathrm{BC} \lim \mathcal{L}(R)$$

Proof. Any $L^{\omega} \in \mathrm{BC} \lim \mathcal{L}(R)$ can be described by $\mathrm{BC}\, 2^{S_R}$. $2^{2^{S_R}}$ is also finite. Thus, any $A \in \mathrm{BC}\, 2^{S_R}$ can be represented in $2^{2^{S_R}}$. This is exactly an acceptance condition in Muller. □

Lemma 3.19.

$$\mathcal{L}^{\omega}(R^{\omega}) = \mathrm{BC}\, \mathrm{ext}\, \mathcal{L}(R)$$

Proof. TODO... □

Lemma 3.20.

$$\mathrm{BC} \lim \mathcal{L}(R) \cap \mathrm{ext}\, \mathcal{L}^*(\textit{reg}) = \mathrm{ext}\, \mathcal{L}(R)$$

Proof. We have $\mathrm{ext}\, \mathcal{L}(R) \subseteq \mathrm{ext}\, \mathcal{L}^*(\text{reg})$ and $\mathrm{ext}\, \mathcal{L}(R) \subseteq \mathrm{BC} \lim \mathcal{L}(R)$. Thus, "$\supseteq$" is shown.

Now, we show "$\subseteq$". Let $L^{\omega} \in \mathrm{BC} \lim \mathcal{L}(R) \cap \mathrm{ext}\, \mathcal{L}^*(\text{reg})$. Because $L^{\omega} \in \mathrm{ext}\, \mathcal{L}^*(\text{reg})$, there is an E-automaton $\mathcal{A}^E$ which accepts L^{ω}. We can assume that $\mathcal{A}^E$ is deterministic (with 3.2).

We must find an R-E-automaton which accepts L^{ω}. We will call it the $\overline{\mathcal{A}}^M$ E-automaton and will construct it in the following.

Let $\mathcal{A}^M$ be the deterministic R-Muller-automaton for L^{ω} (according to 3.3.1 and 3.18). Without restriction, there are no final state sets in $\mathcal{A}^M$ which are not loops. Then, $\overline{\mathcal{A}}^M$ has the same states and transitions as $\mathcal{A}^M$.

Look at a final state q^E of $\mathcal{A}^E$. Without restriction, we can assume that there is no path that we can reach multiple final states at once. Let L_{q^E} be all words which reach q^E exactly once at the end.

Let $w \in L_{q^E}$. Let q be the state in $\mathcal{A}^M$ which is reached after w. Let S be the set of states in $\mathcal{A}^M$ which can be reached from q.

Then, $\mathcal{A}^M$ accepts all words in $L_q \cdot L^\omega_{q,S}$, where L_q is the set of words to q and $L^\omega_{q,S}$ is the set of words of possible infinite postfixes after q in S so that they are accepted. Any word with a prefix in L_q, which is not in $L_q \cdot L^\omega_{q,S}$, will not be accepted by $\mathcal{A}^M$ because $\mathcal{A}^M$ is deterministic. Also, because $L_{q^E} \cap L_q \neq \emptyset$ and $L_{q^E} \cdot \Sigma^\omega \subseteq L^\omega$ and $L_q \cdot L^\omega_{q,S} \subseteq L^\omega$, we get $L^\omega_{q,S} \neq \emptyset$.

Assuming $L^\omega_{q,S} \neq \Sigma^\omega$. Then we would have $L^\omega \notin \operatorname{ext} \mathcal{L}^*(\text{reg})$, which is a contradiction. I.e. $L^\omega_{q,S} = \Sigma^\omega$.

Thus, $\mathcal{A}^M$ accepts all words in $L_q \cdot \Sigma^\omega$. Mark q as a final state in $\overline{\mathcal{A}}^M$. Thus, $\overline{\mathcal{A}}^M$ E-accepts all words in $L_q \cdot \Sigma^\omega \subseteq L^\omega$.

Because we did this for all final states in $\mathcal{A}^E$, there is no $\alpha \in L^\omega$ which is not accepted by $\overline{\mathcal{A}}^M$. I.e., the R-E-automata $\overline{\mathcal{A}}^M$ accepts exactly L^ω. I.e. $L^\omega \in \operatorname{ext} \mathcal{L}(R)$. □

Lemma 3.21.

$$\mathrm{BC} \lim \mathcal{L}(R) \cap \lim \mathcal{L}^*(\mathit{reg}) = \lim \mathcal{L}(R)$$

Proof. This proof is loosely analogue to the proof in 3.20.

We have $\lim \mathcal{L}(R) \subseteq \lim \mathcal{L}^*(\text{reg})$ and $\lim \mathcal{L}(R) \subseteq \mathrm{BC} \lim \mathcal{L}(R)$. Thus, "$\supseteq$" is shown.

Now, we show "$\subseteq$". Let $L^\omega \in \mathrm{BC} \lim \mathcal{L}(R) \cap \lim \mathcal{L}^*(\text{reg})$. Because $L^\omega \in \lim \mathcal{L}^*(\text{reg})$, there is an Büchi-automaton $\mathcal{A}^B$ which accepts L^ω. We can assume that $\mathcal{A}^B$ is deterministic (with 3.2).

We must find an R-Büchi-automaton which accepts L^ω. We will call it the $\overline{\mathcal{A}}^M$ Büchi-automaton and will construct it in the following.

Let $\mathcal{A}^M$ be the deterministic R-Muller-automaton for L^ω (according to 3.3.1 and 3.18). Without restriction, there are no final state sets in $\mathcal{A}^M$ which are not loops. Then, $\overline{\mathcal{A}}^M$ has the same states and transitions as $\mathcal{A}^M$.

Look at the SCC S in $\mathcal{A}^M$. Let $q \in S$. Let $\mathcal{F}_q \subseteq 2^S$ be the set of final states in $\mathcal{A}^M$ with $q \in F$ for all $F \in \mathcal{F}_q$. Let $\mathcal{S}_q \subseteq 2^S$ be the set of loops in S which include q.

Case 1: $\mathcal{F}_q \neq \mathcal{S}_q$.

Case 2: $\mathcal{F}_q = \mathcal{S}_q$. In that case, mark q as a final state in $\overline{\mathcal{A}}^M$.

For the constructed Büchi-automaton $\overline{\mathcal{A}}^M$, we show that it accepts exactly L^ω.

Let $\alpha \in L^\omega(\overline{\mathcal{A}}^M)$. Let q be some final state in $\overline{\mathcal{A}}^M$ which is infinitely often visited by α. Then, $\mathcal{F}_q = \mathcal{S}_q$ from the construction. I.e., no matter what loops through q of the related SCC are visited infinitely often by α, it will be accepted by $\mathcal{A}^M$. Thus, $\alpha \in L^\omega$.

Let $\alpha \in L^\omega$. Then, the set of states F infinitely often visited by α in $\mathcal{A}^M$ is some final state set of the Muller-automaton $\mathcal{A}^M$. In $\mathcal{A}^B$, there is a final state $\tilde{q}$ infinitely often visited by α. Let $\alpha =: \prod_{i=1}^{\infty} w_i$ so that $\prod_{i=1}^{n} w_i$ ends up in $\tilde{q}$ in $\mathcal{A}^B$ for all $n \in \mathbb{N}$ for shortest possible w_i (i.e. we don't miss any $\tilde{q}$). Let S be the SCC in $\mathcal{A}^M$ where we finally end up with α. Then, $F \subseteq S$.

There must be a $q \in F$ so that $\mathcal{F}_q = \mathcal{S}_q$. Then, by construction of $\overline{\mathcal{A}}^M$, q is a final state in $\overline{\mathcal{A}}^M$ and thus, $\alpha \in L^\omega(\overline{\mathcal{A}}^M)$.

Let us show that there is such $q \in F$ by contradiction. I.e. assume there is no such $q \in F$. I.e. for all $q \in F$, $\mathcal{F}_q \neq \mathcal{S}_q$. Of course we have $F \in \mathcal{F}_q$ for all $q \in F$.

Let $\mathcal{P}_{\tilde{q}}$ be the set of loops in $\mathcal{A}^M$ so that all words which end up looping there infinitely would also visit $\tilde{q}$ infinitely often in $\mathcal{A}^B$. Of course, all $P \in \mathcal{P}_{\tilde{q}}$ will be final state sets in $\mathcal{A}^M$ because $\mathcal{A}^B$ would accept. Define $\mathcal{P}_{\tilde{q},S} := \{P \in \mathcal{P}_{\tilde{q}} \mid P \subseteq S\}$. No matter how much other infinte loops in S we add to α so that we still visit some loops from $\mathcal{P}_{\tilde{q},S}$ infinitely often, $\mathcal{A}^M$ and $\mathcal{A}^B$ will keep accepting. Thus, for $P \in \mathcal{P}_{\tilde{q},S}$, every $P' \supseteq P$, $P' \in \mathcal{S}_S$, we have $P' \in \mathcal{P}_{\tilde{q},S}$.

TODO ...

□

Lemma 3.22.

$$\lim \mathcal{L}(R) \cap \overline{\lim}\, \mathcal{L}(R) = \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}(R)$$

Proof. TODO... □

Chapter 4

Results on concrete $*$-language classes

We already showed many results for $\mathcal{L}^*(\mathrm{reg})$.

4.1 Starfree regular languages

This class is also equivalent to the set of $\mathrm{FO}[<]$-definable languages.

Theorem 4.1.

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

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:

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

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

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

Thus we have prooved the equality. □

Theorem 4.2.

$$\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{FO}[<]) \subsetneqq \mathrm{BC}\lim \mathcal{L}^*(\mathrm{FO}[<])$$

Proof. $\subseteq$: $L \subset \Sigma^\omega$ starfree $\Rightarrow$ $L\Sigma^\omega \in \lim(\mathcal{L}^*(\mathrm{FO}[<]))$ □

Proof. $\neq$:

$$\begin{aligned} & L := (\Sigma^* a)^\omega \\ \Rightarrow\ & L = \lim((\Sigma^* a)^*) \\ \Rightarrow\ & L = L^\omega(\exists^\omega x : Q_a x) \end{aligned}$$

And we have $L \notin \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{FO}[<])$. □

With 3.3, we get $\mathrm{ext}\,\mathcal{L} \subseteq \lim \mathcal{L}$.

$\tilde{L} := \lim(\Sigma^* a) = (\Sigma^* a)^\omega \in \lim \mathcal{L}$ but $\tilde{L} \notin \mathrm{ext}\,\mathcal{L}$ as shown in chapter 2.1.

- P1: $\{a\} \in \mathcal{L}$. $a\Sigma^* \in \mathcal{L}$, thus $a\Sigma^\omega = \mathrm{ext}(\{a\}) = \overline{\mathrm{ext}}\, a\Sigma^*$.
- P2a: $\tilde{L}_{2a} := \mathrm{ext}(a^*b) = a^*b\Sigma^\omega$, $a^*b \in \mathcal{L}$. Then $\tilde{L}_{2a} \notin \mathrm{ext}\,\mathcal{L}^*(\mathrm{reg}) \supseteq \mathcal{L}^*(\mathrm{FO}[<])$.
- P2b: $-\tilde{L}_{2a} := \overline{\mathrm{ext}}(-a^*b)$, $-a^*b \in \mathcal{L}$. Then $-\tilde{L}_{2a} \notin \mathrm{ext}\,\mathcal{L}$.
- P3: Follows directly from P2a and P2b.
- P4: $\tilde{L}_4 := \mathrm{ext}(\Sigma^* a) \cap \overline{\mathrm{ext}}(-\Sigma^* b) = \Sigma^* a \Sigma^\omega \cap -(\Sigma^* b \Sigma^\omega)$, whereby $\Sigma^* a \in \mathcal{L}$, $-\Sigma^* b \in \mathcal{L}$. $\tilde{L}_4 \notin \mathrm{ext} \cup \overline{\mathrm{ext}}\,\mathcal{L}^*(\mathrm{reg}) \supseteq \mathcal{L}^*(\mathrm{FO}[<])$ but $\tilde{L}_4 \in \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}$.
- P5: TODO
- P6a/P6b/P7/P8: $\Sigma^* a \in \mathcal{L}$. We can use the same arguments as for $\mathcal{L}^*(\mathrm{reg})$.
- P9: TODO
- P10: TODO

4.2 FO[+1]

Theorem 4.3.

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

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:

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

i.e.

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

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:

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

□

4.3 FO[]

4.4 piece-wise testable

Theorem 4.4.

$$\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{PT}) = \mathrm{BC}\,\lim\,\mathcal{L}^*(\mathrm{PT})$$

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 $\operatorname{ext}(\mathcal{L}^*(\mathrm{PT})) \subseteq \mathrm{BC}\lim \mathcal{L}^*(\mathrm{PT})$.

By complete induction:

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

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

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

This makes the induction complete.

$\supseteq$: It is sufficient to show $\lim(\mathcal{L}^*(\mathrm{PT})) \subseteq \mathrm{BC}\operatorname{ext} \mathcal{L}^*(\mathrm{PT})$.

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

and because A, B are piece-wise testable

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

□

For positive piece-wise testable (pos-PT) languages, we get the same result.

Theorem 4.5.

$$\mathrm{BC}\operatorname{ext} \mathcal{L}^*(\text{pos-PT}) = \mathrm{BC}\lim \mathcal{L}^*(\text{pos-PT})$$

Proof. $\subseteq$: Exactly like the proof for PT except that we leave out the negated part. $\supseteq$: Also like the proof for PT. □

We also have a relation between pos-PT and PT.

Theorem 4.6.

$$\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\text{pos-PT}) = \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{PT})$$

Proof. In the proof of $\lim \mathcal{L}^*(\mathrm{PT}) \subseteq \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{PT})$ we actually proved $\mathrm{BC}\lim \mathcal{L}^*(\mathrm{PT}) \subseteq \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\text{pos-PT})$. Similiarly we also proved $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{PT}) \subseteq \mathrm{BC}\lim \mathcal{L}^*(\text{pos-PT})$.

With 4.4 and 4.4 we get the claimed equality. □

4.5 locally testable

Theorem 4.7.

$$\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{LT}) \subsetneq \mathrm{BC}\lim \mathcal{L}^*(\mathrm{LT})$$

Proof. Let $w \in \Sigma^+$.

$$\begin{aligned}
&\mathrm{ext}(w\Sigma^*) = \lim(w\Sigma^*)\\
&\mathrm{ext}(\Sigma^* w) = \Sigma^* w \Sigma^\omega = \lim(\Sigma^* w \Sigma^*)\\
&\mathrm{ext}(\Sigma^* w \Sigma^*) = \Sigma^* w \Sigma^\omega = \lim(\Sigma^* w \Sigma^*)
\end{aligned}$$

Thus we have

$$\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{LT}) \subseteq \mathrm{BC}\lim \mathcal{L}^*(\mathrm{LT}).$$

But we also have

$$\lim(\Sigma^*) = (\Sigma^* w)^\omega \notin \mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(\mathrm{LT}).$$

□

4.6 locally threshold testable

4.7 endwise testable

- $\mathrm{BC}\,\mathrm{ext}\,\mathcal{L}^*(endwise) \neq \mathrm{BC}\lim \mathcal{L}^*(endwise)$ because $\Sigma^* a \in \mathcal{L}^*(endwise)$.
- $\mathrm{ext}(a\Sigma^* a) = a\Sigma^* a \Sigma^\omega \notin \mathrm{BC}\lim \mathcal{L}^*(endwise)$

4.8 local

4.9 finite / co-finite

- $\lim \mathcal{L}^*(finite) = \{\emptyset\}$

- $\text{ext}\,\mathcal{L}^*(finite) = \mathcal{L}^*(finite) \cdot \Sigma^\omega$
- $\lim \mathcal{L}^*(co - finite) = \{\Sigma^\omega\}$
- $\text{ext}\,\mathcal{L}^*(co - finite) = \{\Sigma^\omega\}$

4.10 dot-depth-n

4.11 L-trivial

4.12 R-trivial

4.13 locally modulo testable

4.14 context free

Chapter 5

Conclusion

Chapter 6

References

[Büc60] J. Richard Büchi. Weak second-order arithmetic and finite automata. *Z. Math. Logik und Grundl. Math.*, 6:66–92, 1960.

[Büc62] J. Richard Büchi. On a decision method in restricted second order arithmetic. In *Logic, Methodology and Philosophy of Science (Proc. 1960 Internat. Congr .)*, pages 1–11. Stanford Univ. Press, Stanford, Calif., 1962.

[Kle56] S. C. Kleene. Representation of events in nerve nets and finite automata. In *Automata studies*, pages 3–41. Princeton University Press, Princeton, N. J., 1956. Annals of mathematics studies, no. 34.

[Mul63] David E. Muller. Infinite sequences and finite machines. In *Switching Theory and Logical Design, Proc. Fourth Annual Symp. IEEE*, pages 3–16. IEEE, 1963.

[PP04] D. Perrin and J.É. Pin. *Infinite Words: Automata, Semigroups, Logic and Games*. Number Bd. 141 in Pure and Applied Mathematics. Elsevier, 2004.

[Sta97] Ludwig Staiger. ω-languages. In *Handbook of formal languages, Vol. 3*, pages 339–387. Springer, Berlin, 1997.

[Str94] Howard Straubing. *Finite automata, formal logic, and circuit complexity*. Birkhäuser Boston Inc., Boston, MA, 1994.

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

[Tho90] Wolfgang Thomas. Automata on infinite objects. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume vol. B, Formal models and semantics, pages 135–191. Elsevier, 1990.

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

[Tho10] Wolfgang Thomas. Infinite computations. Technical report, RWTH Aachen, February 2010.