Language Theoretical Solutions for Church's Problem of Controller Synthesis

von
Jörg Olschewski
Matrikelnummer 242329

Diplomarbeit
im Studiengang Informatik

vorgelegt der
Fakultät für Mathematik, Informatik und Naturwissenschaften der
Rheinisch-Westfälischen Technischen Hochschule Aachen

September 2008
überarbeitete Version vom November 2008

Betreuer: Univ.-Prof. Dr. Dr.h.c. Wolfgang Thomas

angefertigt am
LEHRSTUHL FÜR INFORMATIK VII
Logik und Theorie diskreter Systeme
Univ.-Prof. Dr. Dr.h.c. Wolfgang Thomas

Contents

Chapter 1

Introduction

A frequently regarded topic in computer science is the problem of controller synthesis. When we look at an open system we can identify two actors (or agents) – a controller and the environment. On the one hand, at each point in time, an event occurs inside the environment which influences the behavior of the system. On the other hand, at each point in time, the controller can analyze the preceding behavior and take a certain action to regulate the future behavior of the system. In such a setting, each agent reacts on the actions of the other agent. This is why we also speak of a reactive system. We want to make sure that a system like that fulfills certain properties like maintenance of service or keeping safety conditions. Therefore a system specification is provided that defines these properties. It describes how this reactive system shall behave. Then the question arises: Can we construct a controller that regulates the behavior of the system such that the system specification is fulfilled?

In this thesis we concentrate on a problem which has first been posed by Church [Chu57, Chu63]. He puts this question into a theoretical framework. Church talks about a "requirement which a circuit is to satisfy" instead of a specification. He asks for finding a representation of a circuit which satisfies the requirement or to determine that a circuit like this does not exist. Today the term "circuit" has been replaced by the common notion of a finite automaton. So nowadays under Church's Problem we understand the question whether for a given system specification there is a finite automaton with output which produces a sequence of actions such that the system specification is fulfilled – and can this finite automaton be constructed (synthesized) effectively?

In many applications, one does not know the lifespan of the system in advance or one considers the system to be nonterminating. That is why we model time by the infinite set of natural numbers. We say that at each point in time, both actors take a certain action. At each point i at first the environment takes action X_i and then the controller takes action Y_i. So we can describe the whole behavior of the system by two infinite sequences of actions, namely

$$\begin{aligned} X &= X_0,\ X_1,\ X_2,\ \dots \quad \text{and} \\ Y &= Y_0,\ Y_1,\ Y_2,\ \dots \quad . \end{aligned}$$

The sequence X describes the actions issued by the environment, the sequence Y describes the actions of the controller. The system specification depicts the set of acceptable behaviors. Proceeding on the assumption that the set of possible actions is finite, the system specification can now be considered as a language of infinite words. The specification language consists of all infinite sequences of pairs (X_i, Y_i) that fulfill the system specification.

$$L = \left\{ \binom{X_0}{Y_0}\binom{X_1}{Y_1}\binom{X_2}{Y_2}\cdots \;\middle|\; (X, Y) \text{ fulfills the specification} \right\}$$

At each point in time the controller has the task to analyze the previous behavior of the system and conclude from that to the next action. Precisely, at time point i the controller knows the past actions $X_0, \dots, X_{i-1}$ and $Y_0, \dots, Y_{i-1}$ and additionally the last action of the environment X_i. With that knowledge it has to determine the next action Y_i such that the resulting infinite word is in L.

The first solution of this problem was offered by Büchi and Landweber [BL69]. They also transformed the problem statement into the terminology of infinite games. It is only natural to model the situation described above in a game theoretical context. The environment can arbitrarily adopt unexpected actions so we assume the worst case that it tries to intentionally sabotage the task of the controller. Then the alternating acting of environment and controller can be seen as a game of two competing players, the environment filling the role of Player 1 and the controller filling the role of Player 2. A play of this game is won by Player 2, if the system specification is fulfilled, i.e. if $X \times Y \in L$. Otherwise it is won by Player 1. Church's Problem then translates into the question: Is there a winning strategy for Player 2 in this game and if there is one then how to construct it?

The game described above is a slight variant of a Gale-Stewart game [GS53]. The only difference to Gale-Stewart games is due to the members of the language L. Here we take the elementwise cross product of the two sequences X and Y while in Gale-Stewart games we would take the interleaving of X and Y and obtain a word $X_0 Y_0 X_1 Y_1 \cdots$.

The solution of Büchi and Landweber is even stronger than requested. Church only asked for the controller to be synthesized. Büchi and Landweber give a symmetrical solution – for both parties, the controller and the environment. Their result is that for every specification, expressible by a finite automaton, one can determine, if there is a controller that can fulfill the specification, or if the environment can violate the specification and furthermore that there is a finite automaton operator that implements this controller respectively the environment. They also make an equivalent statement in game terminology.

> "Every finite-state game is determined. Moreover, the player who has a winning strategy in fact has one which can be executed by a finite automaton."
>
> [BL69]

Determinacy of a game is a frequently used concept in game theory. It means that there always is a winning strategy for one of the players. There are only very few artificial games which are not determined. The first one of those nondetermined games has been discovered by Gale and Stewart [GS53]. Later it has been shown by Martin [Mar75] that every Gale-Stewart game which is defined with Borel winning conditions is determined. Admittedly, the type of game we consider here is not a Gale-Stewart game, so the result by Martin is not directly applicable. But both types of games are closely related, so it seems natural (by a coding argument) that all the games of the type we consider here which are defined with Borel winning conditions are determined, too.

In 2007, the Büchi-Landweber Theorem has been refined in two independent papers. Selivanov [Sel07] showed a refinement for aperiodic languages. He stated that each game defined by an aperiodic ω-language L is determined with winning strategies that are computed by aperiodic synchronous transducers. Rabinovich and Thomas [RT07] also have shown analogues of the Büchi-Landweber Theorem. They investigated a repertory of logics and proved that each X-definable game is determined with X-definable winning strategies for X being one of the logics MSO, FO(<), FO(S), FO(<)+MOD or strictly bounded logic. In their paper they also suggest the idea to express winning conditions and strategies as (tuples of) languages.

The aim of the thesis is to pursue this approach. We want to refine the Büchi-Landweber Theorem for certain subclasses of the regular languages. We concentrate on so-called weak languages. Examples will be locally testable languages, piecewise testable languages and their counterparts of threshold countable languages.

There are several possibilities how to express the specification of the system and the implementation of the environment and the controller. Büchi and Landweber did both by finite automata – either with an accepting condition or by finite automata with output. The paper [RT07] presents both in the form of logical formulas. In this thesis we express winning conditions as ω-languages and strategies as tuples of languages of finite words.

Let us illustrate these ideas in a trivial example. The ω-language

$$L_{\mathrm{cp}} = \left\{ \alpha \in (\{0,1\}^2)^\omega \;\middle|\; \forall i\colon \alpha(i) = \binom{0}{0} \vee \alpha(i) = \binom{1}{1} \right\}$$

describes all those system behaviors where Player 2 copies the choice of Player 1. It shall serve as the winning condition of the game. The first component of each letter is chosen by Player 1, the second one by Player 2. A typical play where Player 2 wins would then look like the infinite word

$$\binom{0}{0}\binom{0}{0}\binom{1}{1}\binom{0}{0}\binom{1}{1}\binom{1}{1}\binom{1}{1}\binom{0}{0}\binom{1}{1}\binom{0}{0}\binom{1}{1}\cdots$$

and a strategy can be given by a pair of languages (T_0, T_1) where

$$T_0 = \{w\tbinom{0}{*} \mid w \in \Sigma^*\}$$
$$T_1 = \{w\tbinom{1}{*} \mid w \in \Sigma^*\}$$

The symbol $*$ here serves as a dummy symbol and will be replaced by a proper letter in the next turn of Player 2. This strategy means: if the current play prefix is of the form T_0, then pick 0 as the next action; if it is of the form T_1, then pick 1. Clearly this strategy is a winning strategy for Player 2, because the resulting play is always a word from L_{cp}.

As one can see, we have expressed both winning conditions and winning strategies as (tuples of) languages. It is important to notice that the ω-language L_{cp} can be defined by only regarding factors of length 1. The winning strategy (T_0, T_1) has the very same property. Each of the languages T_0 and T_1 can be defined by only regarding factors of length 1. We exploit the fact that such strategies can contain the dummy symbol $*$ only at the very end of each word. Then a word is in T_0, only if it contains the factor $\binom{0}{*}$. Languages that only depend on the factors of length k are called *k-locally testable languages*. We will see for example that in general k-locally testable languages are determined with k-locally testable winning strategies.

Outline

This thesis is structured as follows. In Chapter 2 the basic notions on infinite games are introduced. We discuss two different types of infinite games. Games on graphs are treated in Section 2.2. In particular the determinacy of parity games and Muller games and of their weak counterparts are recalled. The second type of infinite games is treated in Section 2.3. There we introduce Church's Problem and its associated game formally and embed it into the concept of games on graphs. In Section 2.5 the focus lies on weak languages and weak games. In Section 2.6 we remind of some basic game reduction methods to convert Muller games into parity games and weak Muller games into weak parity games. Chapter 3 covers the analysis of Church's Problem for some classes of regular languages, well-known from language theory. We approach locally testable languages, piecewise testable languages, locally threshold testable languages, piecewise threshold testable languages and some modulo counting languages. In Chapter 4, we consolidate the results from the previous chapter to gain a general result which covers most of the preceding theorems. This general result is applied to some examples afterwards. The last chapter (Chapter 5) summarizes this work and suggests possibilities for future work.

Acknowledgments

I would like to thank all people who contributed to this thesis, in particular my advisor Prof. Wolfgang Thomas for proposing this topic and for his continuous support and encouragement, Prof. Erich Grädel for his kind readiness to review this thesis, and Dr. Christof Löding for some fruitful discussions about the proof of Lemma 41.

Chapter 2

Infinite Games

In this chapter we describe two different possibilities for defining infinite two-player games. First one can describe a game by a language of all possible plays. Then one can define the plays which are won by the first player by a certain language of plays and the plays which are won by the second player by its complement. Gale and Stewart [GS53] introduced these games and they showed that well-founded Gale-Stewart games are always determined. The problem stated by Church [Chu57] relates to a slight variant of Gale-Stewart games. We will describe it in Section 2.3.

Secondly there are games on graphs. These games can be defined by two sets of nodes, each set belonging to one of the players. The nodes represent game-states and a play proceeds through these nodes, depending on the moves of the players. The winning player is then determined by certain conditions on the visited nodes. There is a vast range of such conditions. In the Section 2.2 we define games on graphs and several winning conditions that we will use in this thesis. In Section 2.5 we concentrate on "weak" winning conditions and define what is meant when we talk about "weak games". In Section 2.6 we remind of some game reductions for games on graphs. This is used to make the winning conditions for these games simpler, but it comes along with a more complex game graph. As some reductions are used in some of the proofs, we will define them there. Section 2.1 repeats the basic notions of languages.

2.1 Words and Languages

We assume that the reader is familiar with the usual notations of mathematics and language theory. Nevertheless we give a short repetition of the most important concepts. The list shall be by no means exhaustive.

For any set A let $\mathcal{P}(A)$ denote the power set of A. An *alphabet* is a finite and nonempty set of symbols. We usually denote such a set with a capital Greek letter like Σ and call its elements symbols, letters or characters.

A *(finite) word* over Σ is a tuple $w = a_1 \cdots a_n$ of letters from Σ. The *length* $|w|$ of w is n. The empty tuple is called ε and has length 0. An *infinite word* over Σ is an infinite sequence $\alpha = a_0 a_1 \cdots$ of letters from Σ and $\alpha(i) = a_i$ is the i-th letter of α. We denote with Σ^* the set of all words over Σ and with Σ^+ the set of all words over Σ of length at least 1. So $\Sigma^+ = \Sigma^* \setminus \{\varepsilon\}$. Σ^ω is the set of all infinite words over Σ. A $*$-*language* over Σ is a subset of Σ^*, an ω-*language* over Σ is a subset of Σ^ω. In the literature the above $*$-languages are usually just called languages (of finite words). We introduce a special name here, because we want to emphasize the difference between languages of finite words and languages of infinite words.

We assume that the reader is familiar with regular expressions and their semantics. We will not define them here.

2.2 Games on Graphs

Games on graphs take place on a game arena. When visualizing such games, the game arena serves as a playboard on which a single token is moved from one node to the next node along the edges between them. The nodes are divided into those belonging to Player 1 and those belonging to Player 2. When the token reaches a node belonging to a player, this player can choose arbitrarily among the adjacent graph nodes and move the token to this next node.

A game arena is therefore often (see e.g. the survey paper by Zielonka [Zie98]) defined as a tuple (Q_1, Q_2, E), where Q_1 and Q_2 are the set of nodes of the two players and E is an edge relation. In the proofs in Chapter 3 we will map (partial) plays of a game to runs of a deterministic finite automaton. It turned out that by defining game arenas directly with a transition function δ instead of an edge relation, these mappings can be handled with more ease. Therefore we define games on graphs similarly to [Tho95].

Definition 1. A *game arena* is a tuple $G = (Q_1, Q_2, \Sigma_1, \Sigma_2, \delta)$, where

- Q_1 and Q_2 are (possibly infinite) nonempty and disjoint sets of nodes (or "states"), belonging to Player 1 and Player 2, respectively; we define $Q := Q_1 \cup Q_2$
- Σ_1 and Σ_2 are alphabets, belonging to Player 1 and Player 2, respectively; we define $\Sigma := \Sigma_1 \times \Sigma_2$
- $\delta\colon (Q_1 \times \Sigma_1) \cup (Q_2 \times \Sigma_2) \to Q$ is a transition function, which assigns to each state and letter, belonging to the same player, a new state.

In the sequel, we fix the alphabets for Player 1 and Player 2 and therefore we omit both alphabets in the tuple notation of a game arena. So we write it as a tuple $G = (Q_1, Q_2, \delta)$.

When the token reaches a node $q \in Q_1$, Player 1 has to choose the next node from $\delta(q, \Sigma_1)$ and when it reaches a node $q \in Q_2$, Player 2 has to choose from $\delta(q, \Sigma_2)$. Since each vertex has a successor, the game never ends. We can witness an infinite path through the game arena which we call a play.

Definition 2. A *play* $\varrho = q_0, q_1, \ldots$ is an infinite sequence of nodes, such that for all $i \in \mathbb{N}$ there is an $a \in \Sigma$ with $\delta(q_i, a) = q_{i+1}$. A *partial play* $q_0, \ldots, q_k$ is a finite sequence of nodes, such that for all $0 \leq i < k$, there is an $a \in \Sigma$ with $\delta(q_i, a) = q_{i+1}$. For every play ϱ, two sets of nodes are defined, namely

$$\begin{aligned} \mathrm{Occ}(\varrho) &:= \big\{q \in Q \;\big|\; \exists i \in \mathbb{N} \colon \varrho(i) = q\big\} \text{ and} \\ \mathrm{Inf}(\varrho) &:= \big\{q \in Q \;\big|\; \forall i \in \mathbb{N}\ \exists j > i \colon \varrho(j) = q\big\} \end{aligned}$$

being the set of nodes occurring in ϱ and the set of nodes occurring infinitely often in ϱ, respectively.

The game defined so far still lacks a winner and a loser. For determining a winner for a certain play, we need a so-called winning condition. If the winning condition is fulfilled, Player 2 wins the play, otherwise Player 1 wins it.

There are diverse winning conditions, that we can constitute for a game arena, but we will only need a few of them. For every winning condition needed, we now define special games, using them.

Definition 3. A *Muller game* is a tuple $(G, \mathcal{F})$ where G is a game arena and $\mathcal{F} \subseteq \mathcal{P}(Q)$ is a family of subsets of Q. Player 2 wins a play ϱ of this game, if $\mathrm{Inf}(\varrho) \in \mathcal{F}$. In the other case, Player 1 wins the play.

A *weak Muller game* is a tuple $(G, \mathcal{F})$ where G is a game arena and $\mathcal{F} \subseteq \mathcal{P}(Q)$ is a family of subsets of Q. Player 2 wins a play ϱ of this game, if $\mathrm{Occ}(\varrho) \in \mathcal{F}$. In the other case, Player 1 wins the play. Weak Muller games are sometimes called *Staiger-Wagner games.*

For the parity winning conditions, we consider so-called coloring functions. That are functions $c \colon Q \to \{0, \ldots, m\}$, which assign to each node $q \in Q$ an element from a finite set of colors (here: numbers).

Definition 4. Let

$$\begin{aligned} C(\varrho) &:= \max\big\{c(q) \;\big|\; q \in \mathrm{Occ}(\varrho)\big\} \text{ and} \\ C_\omega(\varrho) &:= \max\big\{c(q) \;\big|\; q \in \mathrm{Inf}(\varrho)\big\} \end{aligned}$$

be the maximal color of nodes occurring in ϱ and the maximal color of nodes occurring infinitely often in ϱ, respectively.

A *parity game* is a tuple (G, c) where G is a game arena and $c \colon Q \to \{0, \ldots, m\}$ is a coloring function. Player 2 wins a play ϱ, if $C_\omega(\varrho)$ is even. Otherwise, Player 1 wins the play.

A *weak parity game* is a tuple (G, c) where G is a game arena and $c\colon Q \to \{0, \ldots, m\}$ is a coloring function. Player 2 wins a play ϱ, if $C(\varrho)$ is even. Otherwise, Player 1 wins the play.

All these games consist of a game arena G and a second component, which determines the winner of each play. Therefore we lapidary speak of a *winning condition* when referring to this second component. We denote such a winning condition with the Greek letter φ. Now all the games on graphs are tuples (G, φ). We will come back to this notion when talking about game reductions in the next section.

With the preceding definitions we have completely described all the games on graphs that we will use. What is still missing are the solutions of such games. We now introduce strategies and the important notion of determinacy.

Definition 5. A *strategy for Player 1* is a function f_1 that assigns to each partial play $q_0, \ldots, q_k$ which ends in a node $q_k \in Q_1$ of Player 1 a letter $a \in \Sigma_1$. A partial play $\varrho = q_0, \ldots, q_k$ is said to be *consistent with a strategy* f_1, if for all $0 \leq i < k$ with $q_i \in Q_1$ holds $\delta(q_i, f_1(q_0, \ldots, q_i)) = q_{i+1}$. A play $\varrho = q_0, q_1, \ldots$ is *consistent with a strategy* f_1, if every partial play prefixing ϱ is consistent with f_1. A strategy f_1 for Player 1 is called *winning from* q_0, if every play starting with q_0 and consistent with f_1 is won by Player 1. The set W_1 of all vertices q where there is a winning strategy for Player 1 from q is named the *winning region of Player 1*.

Analogously, a *strategy for Player 2* is a function f_2 that assigns to each partial play $q_0, \ldots, q_k$ which ends in a node $q_k \in Q_2$ of Player 2 a letter $x \in \Sigma_2$. A partial play $\varrho = q_0, \ldots, q_k$ is said to be *consistent with a strategy* f_2, if for all $0 \leq i < k$ with $q_i \in Q_2$ holds $\delta(q_i, f_2(q_0, \ldots, q_i)) = q_{i+1}$. A play $\varrho = q_0, q_1, \ldots$ is *consistent with a strategy* f_2, if every partial play prefixing ϱ is consistent with f_2. A strategy f_2 for Player 2 is called *winning from* q_0, if every play starting with q_0 and consistent with f_2 is won by Player 2. The set W_2 of all vertices q where there is a winning strategy for Player 2 from q is named the *winning region of Player 2*.

A game is called *determined*, if $W_1 \cup W_2 = Q_1 \cup Q_2$, i.e for every starting node either Player 1 or Player 2 has a winning strategy.

Note that for predefined strategies for both players, the resulting play is uniquely determined. In general, one is interested in very simple winning strategies – in particular in strategies with finite memory. There are cases, in which not even a single bit of memory is needed to provide a winning strategy. In that case, we call the strategy memoryless or positional.

Definition 6. Formally a strategy f is called *memoryless*, if $f(q_0, \ldots, q_k) = f(q'_0, \ldots, q'_l)$ for all partial plays $q_0, \ldots, q_k$ and $q'_0, \ldots, q'_l$ with $q_k = q'_l$. If a strategy is memoryless, we can write it as a function $f\colon Q \to \Sigma$, which maps only the last state to a letter instead of the whole partial play.

Parity games and weak parity games are both determined. For parity games Emerson and Jutla [EJ91] and Mostowski [Mos91] independently proved that memoryless strategies suffice. Today this is a well-known result. For a comprehensive proof see also [Zie98, GTW02].

Theorem 7. *Parity games are determined with memoryless winning strategies. Furthermore, if G is finite, then the winning regions and the winning strategies can be constructed effectively.*

The same result also holds for weak parity games. The more basic proof can be found for example in [LT00].

Theorem 8. *Weak parity games are determined with memoryless winning strategies. Furthermore, if G is finite, then the winning regions and the winning strategies can be constructed effectively.*

2.3 Church's Problem

In this section, we want to formulate Church's Problem. The formalism that we use is based on the notion of an infinite game. We therefore describe a game between two players, called Player 1 and Player 2. Player 1 and Player 2 choose letters in turn. Two letters at a time are put into a pair. The pairs are concatenated and form an infinite word. For deciding the winner, this word is tested for membership of a certain language L. Player 2 wins the game, if the word is in L, otherwise Player 1 wins the game.

Let Σ_1 denote a finite set of symbols (an alphabet) from which Player 1 can choose and let Σ_2 denote the alphabet of Player 2. The letter $* \notin \Sigma_2$ is a fixed letter, which does not occur in Σ_2. This letter serves as a placeholder.

In our view, Player 1 and Player 2 present letters alternately and append them to a common word. This common word has the form

- $P_1 := (\Sigma_1 \times \Sigma_2)^*$, if it is Player 1's turn and
- $P_2 := (\Sigma_1 \times \Sigma_2)^*(\Sigma_1 \times \{*\})$, if it is Player 2's turn.

Starting with the empty word, in each turn the word grows successively larger. When it is Player 1's turn, he takes the common word of the form P_1, chooses a letter $a \in \Sigma_1$ and appends $\binom{a}{*}$ to the common word. Then the word is of the form P_2. It is Player 2's turn. She chooses a letter $x \in \Sigma_2$ and replaces the $\binom{a}{*}$ at the very end of the word by $\binom{a}{x}$. Then again, the word is in P_1 and it is Player 1's turn, etc. Since the game under consideration is infinitely long, the length of this word is unbounded and we can identify it with an infinite word α, which we call the result of a play.

For a precise definition of the result of a play, we somehow have to express that one finite word can be continued to another infinite word. For that reason, we extend the usual prefix relation a little bit. We not only regard words from P_1 as being able to prefix an infinite word, but also words from P_2. The succeeding $*$ in a word from P_2 is ignored.

Definition 9. For $w \in P_1 \cup P_2$ and $\alpha \in (\Sigma_1 \times \Sigma_2)^\omega$ we set

$$\begin{aligned} w \sqsubseteq \alpha :\Longleftrightarrow \quad & \exists \beta \in \Sigma^\omega \colon w \cdot \beta = \alpha \\ & \vee \exists \beta \in \Sigma^\omega \ \exists u \in P_1 \ \exists a \in \Sigma_1 \ \exists x \in \Sigma_2 \colon \\ & \quad w = u \cdot \binom{a}{*} \wedge w \cdot \binom{a}{x} \cdot \beta = \alpha. \end{aligned}$$

We call $\sqsubseteq$ the *extended prefix relation.*

From now on we fix the alphabets Σ_1 and Σ_2. Therefore we can define the main object that we use throughout this thesis just by a single ω-language.

Definition 10. An *instance of Church's Problem* is an ω-language $L \subseteq (\Sigma_1 \times \Sigma_2)^\omega$. We denote the game for L with $\mathrm{Ch}(L)$. A *play* for $\mathrm{Ch}(L)$ is an infinite sequence $\varrho = w_0, w_1, \ldots$ of finite words such that

- $w_0 = \varepsilon$,
- for all even $i \in \mathbb{N}$ there is an $a \in \Sigma_1$ such that $w_{i+1} = w_i\binom{a}{*}$ and
- for all odd $i \in \mathbb{N}$ there are $u \in P_1$, $a \in \Sigma_1$ and $x \in \Sigma_2$ such that $w_i = u\binom{a}{*}$ and $w_{i+1} = u\binom{a}{x}$.

A *partial play* for $\mathrm{Ch}(L)$ is a finite sequence $w_0, \ldots, w_k$ which can be continued to a play. For each play ϱ, there is a unique infinite word $\alpha(\varrho) \in (\Sigma_1 \times \Sigma_2)^\omega$ with $\varrho(i) \sqsubseteq \alpha(\varrho)$ for all $i \in \mathbb{N}$. We call this word the *result of a play.* Player 2 wins the play ϱ, if $\alpha(\varrho) \in L$. In the other case, Player 1 wins it.

Note that there is a bijection between the set of all possible plays and $(\Sigma_1 \times \Sigma_2)^\omega$. For each possible play, there is a corresponding result of that play and for each possible result of a play there is a corresponding play.

One way to describe strategies for the players is by gathering all the words, where a certain letter shall be chosen, and put them into a set. Thus, for each letter emerges a set of nodes, happening to be a $*$-language over $(\Sigma_1 \times (\Sigma_2 \cup \{*\}))$. The language is either a subset of P_1 or a subset of P_2. We put these languages into a tuple, obtaining two tuples, one for Player 1 and one for Player 2.

Definition 11. A *strategy for Player 1* is an indexed family of languages $(S_a)_{a \in \Sigma_1}$ which are pairwise disjoint and cover P_1, i.e. $\biguplus_{a \in \Sigma_1} S_a = P_1$.

A *strategy for Player 2* is an indexed family of languages $(T_x)_{x \in \Sigma_2}$ which are pairwise disjoint and cover P_2, i.e. $\biguplus_{x \in \Sigma_2} T_x = P_2$.

A play ϱ is *played consistent to* a strategy $(S_a)_{a \in \Sigma_1}$ for Player 1, if for every odd $i \in \mathbb{N}$ and $\varrho(i) = w\binom{a}{*}$ it holds that $w \in S_a$. A play ϱ is *played*

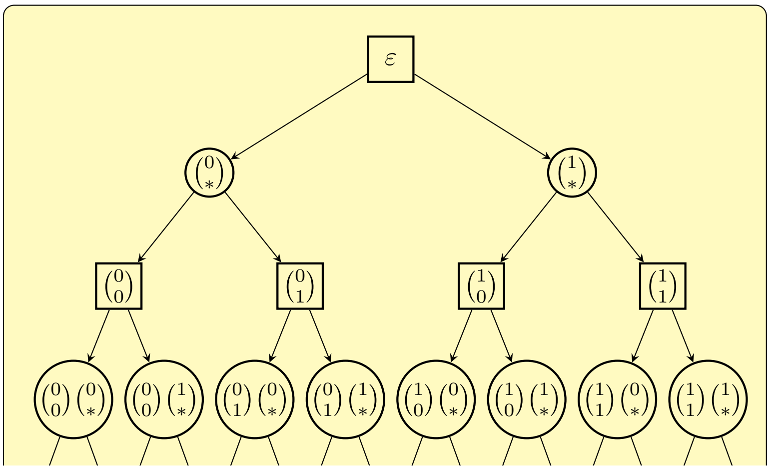

Figure 2.1: Excerpt from the game arena of a game with $\Sigma_1 = \Sigma_2 = \{0, 1\}$.

consistent to a strategy $(T_x)_{x\in\Sigma_2}$ for Player 2, if for every even $i \in \mathbb{N}$ and $\varrho(i) = w\binom{a}{x}$ it holds that $w\binom{a}{*} \in T_x$.

A strategy $(S_a)_{a\in\Sigma_1}$ for Player 1 is called *winning*, if every play played consistent to it is won by Player 1. A strategy $(T_x)_{x\in\Sigma_2}$ for Player 2 is called *winning*, if every play played consistent to it is won by Player 2.

If strategies for Player 1 and Player 2 are preset, there is a play satisfying both strategies and furthermore this play is uniquely determined.

We can also map this game to the notion of games on graphs, introduced in Section 2.2. For this, we model it by the game arena $G = (P_1, P_2, \gamma)$, where

- P_1 and P_2 are defined as above,
- $\gamma(w, a) = w\binom{a}{*}$ and
- $\gamma(w\binom{a}{*}, x) = w\binom{a}{x}$ for all $w \in P_1, a \in \Sigma_1$ and $x \in \Sigma_2$.

This game always starts with the empty word, so we declare the node $p_0 := \varepsilon$ to be the starting node for all plays on this game arena. See Figure 2.1 for a finite excerpt from a game arena.

This representation of the game arena as an infinite tree is similar to the one defined by Gale and Stewart [GS53]. Gale and Stewart also defined the set of positions of their game as the vertices of an infinite tree. However the plays are described differently. For example a play in $\mathrm{Ch}(L)$ starting with ε, $\binom{0}{*}$, $\binom{0}{1}$, $\binom{0}{1}\binom{0}{*}, \ldots$ would translate to $x_0, 0, 01, 010, \ldots$ in Gale-Stewart games. This is an important difference when it is demanded to describe winning conditions by ω-languages.

With the above definitions, we can now define what it means to solve Church's Problem. This will be the main notion used in this thesis.

Definition 12. Let $\mathcal{L}$ be a class of ω-languages and $\mathcal{K}$ be a class of $*$-languages. We say that *$\mathcal{L}$-games are determined with $\mathcal{K}$-winning strategies*, if for each $L \in \mathcal{L}$

$$\begin{aligned} &\exists (S_a)_{a\in\Sigma_1}\ (\forall a \in \Sigma_1 \colon S_a \in \mathcal{K} \wedge (S_a)_{a\in\Sigma_1} \text{ is winning for Player 1}) \\ \vee &\exists (T_x)_{x\in\Sigma_2}\ (\forall x \in \Sigma_2 \colon T_x \in \mathcal{K} \wedge (T_x)_{x\in\Sigma_2} \text{ is winning for Player 2}), \end{aligned}$$

so for each $L \in \mathcal{L}$ there is either a winning strategy $(S_a)_{a\in\Sigma_1}$ for Player 1, which is in $\mathcal{K}$ or a winning strategy $(T_x)_{x\in\Sigma_2}$ for Player 2, which is in $\mathcal{K}$.

2.4 Example

Let us look at an example of Church's Problem, which one can find in [Tho08]. Let $\Sigma_1 := \Sigma_2 := \{0, 1\}$. We set $L \subseteq \Sigma^\omega$ to be the language defined by

$$\begin{aligned} \alpha \in L \quad :\Longleftrightarrow \quad & \forall i \colon \alpha(i) \neq \tbinom{1}{0} \\ & \wedge \forall i \colon \alpha(i) = \tbinom{0}{0} \rightarrow \alpha(i+1) \neq \tbinom{0}{0} \\ & \wedge \left(\forall i\ \exists j > i \colon \alpha(j) = \tbinom{0}{1}\right) \rightarrow \left(\forall i\ \exists j > i \colon \alpha(j) = \tbinom{0}{0}\right). \end{aligned}$$

In the first row of this definition, the infix $\binom{1}{0}$ is excluded from the set of allowed infixes. If such an infix occurs in α, Player 2 looses the game. In the second row, it is simply stated, that Player 2 may not pick the letter 0 two times consecutively and the third row demands infinitely many statements of 0 from Player 2, if Player 1 states infinitely many letters 0.

We can also express the language L by a regular expression. Then

$$\begin{aligned} L = \quad & (\varepsilon + \tbinom{0}{0})[(\tbinom{0}{1} + \tbinom{1}{1})^+ \tbinom{0}{0}]^\omega \\ & + (\varepsilon + \tbinom{0}{0})[(\tbinom{0}{1} + \tbinom{1}{1})^+ \tbinom{0}{0}]^* \tbinom{1}{1}^\omega. \end{aligned}$$

Player 2 has a strategy to win this game. Whenever Player 1 provides 1, Player 2 answers with 1. When Player 1 provides 0, the answer depends on the last but one letter. If the last but one letter was $\binom{0}{0}$, then Player 2 answers with 1, else with 0.

We express this solution of the game as a tuple of languages (T_0, T_1). If the game state is in T_1, Player 2 answers with 1, else she answers with 0:

$$\begin{aligned} T_1 &:= (\Sigma_1 \times \Sigma_2)^* (\tbinom{1}{*} + \tbinom{0}{0}\tbinom{0}{*}) \\ T_0 &:= (\Sigma_1 \times \Sigma_2)^* \setminus T_1 \end{aligned}$$

Both the instance of Church's Problem L and its solution (T_0, T_1) are regular. Moreover they are expressible by some conditions over infixes and suffixes of length at most 2.

2.5 Weak Games

In Section 2.2 about games on graphs we have seen Muller games and parity games and their weak counterparts, weak Muller games and weak parity games. The weak versions do not consider infinite behavior anymore. In particular, the winning conditions of these weak games do not depend on the set $\mathrm{Inf}(\varrho)$ of states occurring infinitely often in a play.

For instances of Church's Problem there do also exist these weak counterparts. We say that $\mathrm{Ch}(L)$ describes a *weak game*, if the language L is recognized by a deterministic weak Muller automaton, also called a Staiger-Wagner automaton (cf. [SW74]).

Definition 13. A *Staiger-Wagner automaton* is a tuple $\mathfrak{A} = (Q, \Sigma, q_0, \delta, \mathcal{F})$ where

- Q is a finite set of states,
- Σ is an alphabet,
- $q_0 \in Q$ is the starting state,
- $\delta \colon Q \times \Sigma \to Q$ is the transition function and
- $\mathcal{F} \subseteq \mathcal{P}(Q)$ is a family of state sets.

For every word $\alpha \in \Sigma^\omega$ there is a unique run $\varrho = q_0, q_1, \ldots$ of $\mathfrak{A}$ on α with $\delta(q_i, \alpha(i)) = q_{i+1}$ for every $i \in \mathbb{N}$. Similar to games, we define the set

$$\mathrm{Occ}(\varrho) := \big\{ q \in Q \;\big|\; \exists i \in \mathbb{N} \colon \varrho(i) = q \big\}$$

of all occurring states in a run. An infinite word $\alpha \in \Sigma^\omega$ is accepted by $\mathfrak{A}$, if $\mathrm{Occ}(\varrho) \in \mathcal{F}$.

Most of the games that we regard in the next chapters will be weak games. The proof idea that we are going to present in Chapter 3 is based on such weak games. Therefore it is helpful to distinguish between both types of games.

Definition 14. A game $\mathrm{Ch}(L)$ is called a *weak game*, if L is recognized by a Staiger-Wagner automaton, otherwise it is called a *strong game*.

2.6 Game Reductions

The idea of a game reduction is to reduce a game with a complex winning condition to a game with a more simple winning condition but with an extended game graph. The simple game can then be solved and the resulting winning strategies can be transformed back to the complex game graph, yielding a finite-state strategy. One can for example reduce a Muller game to a parity game and thus obtain a more complex game arena, where a memoryless winning strategy exists. This memoryless winning strategy together with the finite game graph of the parity game then describe a finite-state winning strategy for the Muller game.

Definition 15. Let $G = (Q_1, Q_2, \delta)$ and $G' = (Q'_1, Q'_2, \delta')$. The game (G, φ) is *reducible* to (G', φ'), if there is a finite set S and functions $g\colon Q \to S$ and $f\colon Q \times S \to S$ with

1. $Q'_1 = Q_1 \times S$ and $Q'_2 = Q_2 \times S$
2. each play $\varrho = q_0, q_1, \ldots$ in G is translated into a play $\varrho' = q'_0, q'_1, \ldots$ in G' as follows:
   - $q'_0 = (q_0, g(q_0))$
   - $\forall q \in Q_1, s \in S, a \in \Sigma_1\colon\ \delta'((q, s), a) = (\delta(q, a), f(\delta(q, a), s))$
   - $\forall q \in Q_2, s \in S, a \in \Sigma_2\colon\ \delta'((q, s), a) = (\delta(q, a), f(\delta(q, a), s))$
3. Player 1 wins ϱ in (G, φ) iff Player 1 wins ϱ' in (G', φ').

Note that by the translation of a play ϱ into a play ϱ', described in item 2 of Definition 15, we also obtain that each partial play $q_0, \ldots, q_k$ of (G, φ) is translated into a partial play $q'_0, \ldots, q'_k$ of (G', φ').

Throughout this thesis, we will use two common game reductions. The first one is reducing a weak Muller game to a weak parity game. The second is reducing a Muller game to a parity game. In the following we will shortly describe both reductions together with the data structures they use.

The reduction of weak Muller games to weak parity games is simple. For a given game arena $G = (Q_1, Q_2, \delta)$ and weak Muller set $\mathcal{F}$, define a new game arena $G' = (Q'_1, Q'_2, \delta')$ by setting $Q'_1 = Q_1 \times \mathcal{P}(Q)$, $Q'_2 = Q_2 \times \mathcal{P}(Q)$. The first component of the new game arena G' is the same as in G. The set of already visited states is saved in the new second component, which is called the *appearance record* AR. For a partial play $q_0, \ldots, q_n$, the AR is the set $\{q_0, \ldots q_n\}$. The new transition function δ' is then defined by setting

$$\delta'((q_1, AR_1), a) := (q_2, AR_2) \text{ where } q_2 = \delta(q_1, a), AR_2 = AR_1 \cup \{q_2\}\,.$$

Colors are assigned to the new states by

$$c(q, M) := \begin{cases} 2 \cdot |M|, & \text{if } M \in \mathcal{F}; \\ 2 \cdot |M| - 1, & \text{otherwise.} \end{cases}$$

For every play ϱ over G, there is a corresponding play ϱ' over G' such that ϱ is won by Player 1 (respectively Player 2) if ϱ' is won by Player 1 (respectively Player 2). One observes, that the AR is of finite size and therefore the AR describes the finite memory which is needed for a finite-state winning strategy in weak Muller games.

Theorem 16. *Weak Muller games are determined with finite-state winning strategies. Furthermore, if G is finite, then the winning regions and the winning strategies can be constructed effectively.*

Reducing Muller games to parity games is much more complex. The AR does not suffice for this. Instead we need a structure, called the *latest appearance record* LAR. It was first introduced in [GH82]. It is basically a record, that tracks the recently visited states. An LAR is a permutation of all states where one of the states is particularly marked. So the LAR structure can be saved in $\mathrm{Perm}(Q) \times Q$, where $\mathrm{Perm}(Q)$ is the set of all possible permutations of Q. The permutation describes the order in which the states appeared in the current play. The marked state (the so-called "hit") indicates which subset of the permutation was visited recently. The hit is commonly specified by underlining the marked state in the permutation. For the empty sequence of states the LAR is $(\underline{q_0}, \ldots, q_n)$. For a sequence of states $q_0, \ldots, q_n, q_{n+1}$, the new LAR is obtained by updating the LAR of $q_0, \ldots, q_n$: move the state q_{n+1} from position j to the beginning of the permutation and underline the state at position j.

For a given game arena $G = (Q_1, Q_2, \delta)$ and Muller set $\mathcal{F}$, define a new game arena $G' = (Q'_1, Q'_2, \delta')$ by setting $Q'_1 = Q_1 \times \mathrm{Perm}(Q) \times Q$, $Q'_2 = Q_2 \times \mathrm{Perm}(Q) \times Q$. The first component of the new game arena G' is again the same as in G. The other two components describe the LAR. The new transition function δ' is then defined by setting

$$\begin{aligned}\delta'((q_1, LAR_1), a) &:= (q_2, LAR_2)\\ &\qquad \text{where } q_2 = \delta(q_1, a), LAR_2 = \mathrm{update}(LAR_1, q_2).\end{aligned}$$

Colors are assigned to the new states by

$$c(q, (q_1, \ldots, \underline{q_h}, \ldots, q_n)) := \begin{cases} 2 \cdot h, & \text{if } \{q_1, \ldots, q_h\} \in \mathcal{F}; \\ 2 \cdot h - 1, & \text{otherwise.} \end{cases}$$

For every play ϱ over G, there is a corresponding play ϱ' over G' such that ϱ is won by Player 1 (respectively Player 2) if ϱ' is won by Player 1 (respectively Player 2). The LAR is of finite size and therefore the LAR describes the finite memory which is needed for a finite-state winning strategy in Muller games. See [Tho95] for a proof of the correctness of this reduction.

Theorem 17. *Muller games are determined with finite-state winning strategies. Furthermore, if G is finite, then the winning regions and the winning strategies can be constructed effectively.*

Chapter 3

Special Regular Winning Conditions

In this chapter we develop analogues to the result of Büchi and Landweber. Therefore it is necessary to link classes of ω-languages with classes of $*$-languages. We will work with well-known examples of classes $\mathcal{K}$ of $*$-languages and associate with them certain classes $\mathcal{L}_{\mathcal{K}}$ of ω-languages.

We will then show for such pairs $(\mathcal{K}, \mathcal{L}_{\mathcal{K}})$ that

$\mathcal{L}_{\mathcal{K}}$-games are determined with $\mathcal{K}$-winning strategies.

The language classes $\mathcal{K}$ and $\mathcal{L}_{\mathcal{K}}$ will share the same name. For example we will consider $\mathcal{K}$ being the class of so-called locally testable $*$-languages and $\mathcal{L}_{\mathcal{K}}$ being the class of so-called locally testable ω-languages. However, a concrete formally defined link between $\mathcal{K}$ and $\mathcal{L}_{\mathcal{K}}$ is not acquired until Chapter 4.

This chapter is structured as follows. At first (Section 3.1) we will illustrate the proof idea for solving Church's Problem for weak games, since the proofs all follow the same scheme. Then we examine Church's Problem for locally testable games (Section 3.2) and for strongly locally testable games (Section 3.3). In the case of strong local testability, which does not describe weak games, we will see that the claim fails. Then we consider locally threshold testable games in Section 3.4 and piecewise testable games in Section 3.5. We generalize the concept of piecewise testability in Section 3.6, obtaining piecewise threshold testable languages. We will do a little excursion to combinatorics on words when examining this language class and obtain an inclusion of the hierarchy of piecewise threshold testable languages inside the hierarchy of piecewise testable languages. In Section 3.7 we proceed to languages with modulo counting quantifiers and examine two different possibilities for those quantifiers.

3.1 Proof Scheme

The proofs of this chaper closely resemble each other, because we will use repeatedly the same machinery of argumentation. Therefore we sketch the proof idea already at this point.

A family of ω-languages $\mathcal{L}$ and a family of $*$-languages $\mathcal{K}$ are given. Each element $L \in \mathcal{L}$ defines an instance of Church's Problem. We want to show that the game $\mathrm{Ch}(L)$ is determined, that means that one of the players has a winning strategy. We simulate the game $\mathrm{Ch}(L)$ by a game on a graph.

Step 1 Construct a game arena $G = (Q_1, Q_2, \delta)$ which has a *finite* set of nodes $Q = Q_1 \cup Q_2$.

Step 2 Define a mapping $h\colon P_1 \cup P_2 \to Q$, such that h respects the transition function. This mapping assigns to each partial play of $\mathrm{Ch}(L)$ a node of the finite game arena G. It holds $\delta(h(p), a) = h(\gamma(p, a))$. We call such a mapping a *homomorphism*. We need h to map each play ϱ of $\mathrm{Ch}(L)$ to a corresponding play ϱ' in the new game graph.

Step 3 Equip G from Step 1 with a winning condition φ such that

- the resulting game (G, φ) is determined,
- the winning regions and winning strategies can be computed effectively,
- the winning strategies are finite-memory strategies (or even memoryless strategies) and
- Player 1 wins a play ϱ iff Player 1 wins the corresponding play ϱ' in $\mathrm{Ch}(L)$

Then we obtain a game which is determined and the winning strategy f of the winning player (the player who wins from $q_0 = h(\varepsilon)$) can be computed effectively.

Furthermore if the strategy f is memoryless (e.g. for weak parity games and parity games), then we can also execute Step 4.

Step 4 Transfer the memoryless winning strategy f from (G, φ) back to $\mathrm{Ch}(L)$. For a strategy $(S_a)_{a\in\Sigma_1}$ for Player 1 accomplish this by first gathering all the nodes from which a certain letter is chosen:

$$Q_1^a := \{q \in Q_1 \mid f(q) = \delta(q, a)\}$$

and then building up the language S_a by taking the preimage of this set under the homomorphism h:

$$S_a := \{w \in P_1 \mid h(w) \in Q_1^a\}.$$

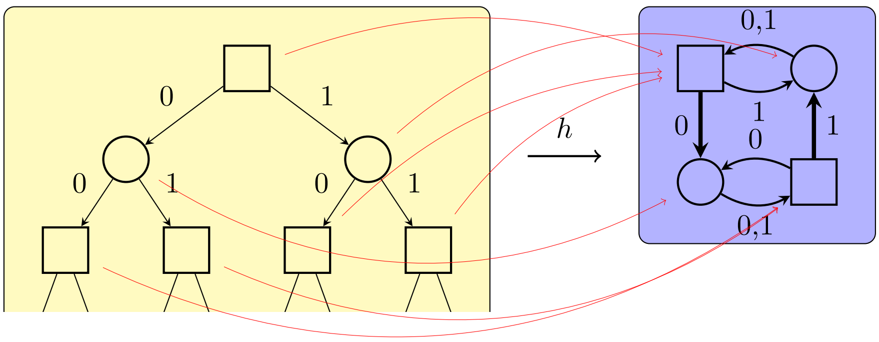

Figure 3.1: Step 3: Bold arrows indicate the memoryless winning strategy in (G, φ)

Analogously for a strategy $(T_x)_{x \in \Sigma_2}$ for Player 2 construct

$$Q_2^x := \{q \in Q_2 \mid f(q) = \delta(q, x)\} \text{ and}$$
$$T_x := \{w \in P_2 \mid h(w) \in Q_2^x\}.$$

We will do this construction such that the preimage of a node $q \in Q$ is a language from $\mathcal{K}$. Furthermore $\mathcal{K}$ will be closed under finite unions, so each set S_a (respectively T_x) will again be in $\mathcal{K}$. This proves the result that $\mathrm{Ch}(L)$ has $\mathcal{K}$-definable winning strategies.

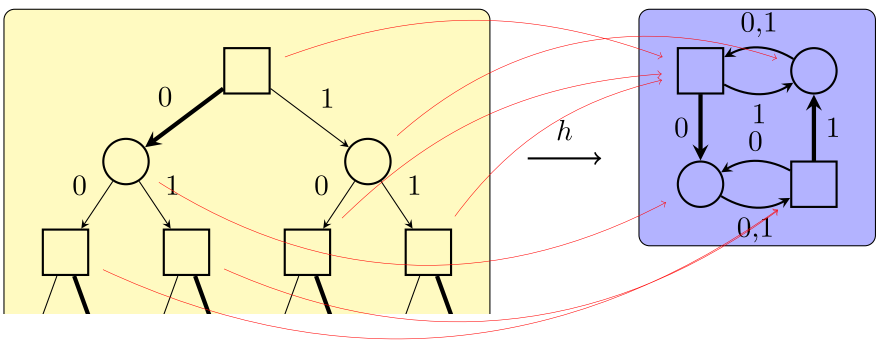

Figure 3.2: Step 4: The strategy is transfered back to $\mathrm{Ch}(L)$

In the case that the strategy is not memoryless, we may still be able to carry out a game reduction to a game with memoryless winning strategies and after that execute Step 4. With a game reduction however comes a finer division of the languages that result from the preimage. So they may not be in $\mathcal{K}$ anymore.

3.2 Locally Testable Languages

In this section, we define the classes of locally testable languages – for both $*$-languages and ω-languages. Then we give a first try of a proof, using a weak Muller game and the reduction to a weak parity game with an appearance record (cf. Section 2.6). We will see, that this method gives an inferior result, compared to a direct approach via weak parity games, which we will use in Section 3.4 in conjunction with locally threshold testable languages.

Locally Testable $*$-Languages

A $*$-language is locally testable, if one can test membership of a word to this language by means of a finite table of prefixes, suffixes and factors. This means that certain words cannot be distinguished, if their prefixes, suffixes and factors of a certain bounded length are equal. So it is obvious to collect these indistinguishable words into the same class of an equivalence relation.

Definition 18. For every word $w \in \Sigma^*$ define the sets

$$\begin{aligned}\mathrm{Infix}_k(w) &:= \big\{u \in \Sigma^* \;\big|\; \exists v_1, v_2 \in \Sigma^* \colon v_1 \cdot u \cdot v_2 = w, |u| \leq k\big\},\\ \mathrm{Prefix}_k(w) &:= \big\{u \in \Sigma^* \;\big|\; \exists v \in \Sigma^* \colon u \cdot v = w, |u| \leq k\big\},\\ \mathrm{Suffix}_k(w) &:= \big\{u \in \Sigma^* \;\big|\; \exists v \in \Sigma^* \colon v \cdot u = w, |u| \leq k\big\},\end{aligned}$$

of infixes, prefixes respectively suffixes of w of length $\leq k$.

For every $k \geq 1$, we define an equivalence relation on the words from Σ^*. Let $u, v \in \Sigma^*$. Then

$$\begin{aligned} u \sim_k v \quad :\Longleftrightarrow \quad & \mathrm{Infix}_k(u) = \mathrm{Infix}_k(v)\\ & \wedge \mathrm{Prefix}_{k-1}(u) = \mathrm{Prefix}_{k-1}(v)\\ & \wedge \mathrm{Suffix}_{k-1}(u) = \mathrm{Suffix}_{k-1}(v).\end{aligned}$$

A language $K \subseteq \Sigma^*$ is called *k-locally testable*, if it can be written as a (finite) union of classes from $\Sigma^*/{\sim_k}$:

$$K = \bigcup_{i=1}^{m} [w_i]_{\sim_k}$$

A language $K \subseteq \Sigma^*$ is called *locally testable*, if it is k-locally testable for a certain $k \geq 1$.

McNaughton [McN74] has shown that it is decidable, whether a given regular $*$-language is locally testable or not.

Example 19. For example the language of all even words, given by the regular expression $(\Sigma\Sigma)^*$, is not locally testable. One cannot distinguish between a word of even length and a word of odd length, if one merely knows the factors of both words and how they start and end. The language defined by the regular expression $(ab)^*$ is however locally testable. We know that it can only contain the factors ab and ba and must start with a and end with b.

The language $a(a+b)^*$ is also locally testable: the prefix a is mandatory; factors and suffixes do not matter at all.

Locally Testable ω-Languages

An ω-language is locally testable, if one can test membership of a word to this language by means of a finite table of prefixes and factors. Suffixes are not considered here, as infinite words don't have finite suffixes.

There are several characterizations of locally testable ω-languages, for example in the book by Perrin and Pin [PP04]. They also call these languages "prefix-factors testable".

Definition 20. For every word $\alpha \in \Sigma^\omega$ define the sets

$$\begin{aligned}\mathrm{Infix}_k(\alpha) &:= \big\{u \in \Sigma^* \bigm| \exists v \in \Sigma^* \exists \beta \in \Sigma^\omega \colon v \cdot u \cdot \beta = \alpha, |u| \leq k\big\},\\ \mathrm{Prefix}_k(\alpha) &:= \big\{u \in \Sigma^* \bigm| \exists \beta \in \Sigma^\omega \colon u \cdot \beta = \alpha, |u| \leq k\big\},\end{aligned}$$

of infixes respectively prefixes of w of length $\leq k$.

For every $k \geq 1$, we define an equivalence relation on the words from Σ^ω. Let $\alpha, \beta \in \Sigma^\omega$. Then

$$\begin{aligned}\alpha \sim_k^\omega \beta \quad :\Longleftrightarrow \quad & \mathrm{Prefix}_{k-1}(\alpha) = \mathrm{Prefix}_{k-1}(\beta)\\ & \wedge \mathrm{Infix}_k(\alpha) = \mathrm{Infix}_k(\beta).\end{aligned}$$

A language $L \subseteq \Sigma^\omega$ is called *k-locally testable*, if it can be written as a finite union of classes from $\Sigma^\omega/\sim_k^\omega$:

$$L = \bigcup_{i=1}^{m} [\alpha_i]_{\sim_k^\omega}$$

A language $L \subseteq \Sigma^\omega$ is called *locally testable*, if it is k-locally testable for a certain $k \geq 1$.

Example 21. For example over the alphabet $\Sigma = \{a, b\}$ let L_1 be the language of all infinite words α with the following property. Every factor of α of length 3 contains an even number of letters a. L_1 is certainly locally testable, as it only depends on the factors of each word. The factors aab, aba, baa and bbb are allowed. All other factors of length 3 are forbidden.

Let L_2 be the language of all infinite words over Σ where eventually nothing but b appears. An ω-regular expression for this language is $(a + b)^*b^\omega$. L_2 is not locally testable. Assume it is k-locally testable. Then the word $b^k ab^\omega$ is equivalent to $(b^k a)^\omega$, because both words possess the same factors up to length k. But $b^k ab^\omega \in L$ while $(b^k a)^\omega \notin L$.

A slightly modified version of L_2 would be the ω-language L_3 defined by $(a+b)^* c^\omega$ which again is locally testable. The factors ca and cb do not occur, but c does occur.

Church's Problem

We now examine Church's Problem for locally testable languages. An analogue to the Büchi-Landweber Theorem would be, that every game that is defined by a locally testable ω-language is determined with locally testable winning strategies. This statement really holds and in fact we can even sharpen it by talking about k-locally testable games for a specific k.

Theorem 22. *For every $k \in \mathbb{N}$, k-locally testable games are determined with $(k+1)$-locally testable winning strategies.*

This is not the best result, we can get. In fact even k-locally testable winning strategies are enough for winning this game. We will see the better result as Theorem 29 in one of the next sections as a special case of locally threshold testable games. This weaker result is due to a detour over weak Muller games and the game reduction method. The reason why we still give the proof for this result is, that the proofs developed over time and in the early stages we used a detour over weak Muller games and the game reduction method. It is interesting to see that this method does not give us the sharpest result.

Proof. Let $L \subseteq (\Sigma_1 \times \Sigma_2)^\omega$ be an instance of Church's Problem and let L be k-locally testable. Then there are words $\alpha_1, \ldots, \alpha_m \in \Sigma^\omega$ with $L = \bigcup_{i=1}^m [\alpha_i]_{\sim_k^\omega}$.

We map every play of the original game to a play of another well-known game, namely a weak parity game. However, we won't do this directly, but instead take a little detour over a weak Muller game. For this we define a game arena $G = (Q_1, Q_2, \delta)$ with node sets Q_1 and Q_2 owned by Player 1 and Player 2, respectively. In every node, we memorize the last k symbols that we have seen so far. We set

$$\begin{aligned} Q_1 &:= \big\{ w \bigm| w \in P_1, |w| \leq k \big\}, \\ Q_2 &:= \big\{ w \cdot \tbinom{a}{*} \bigm| w \in P_1, |w| < k, a \in \Sigma_1 \big\} \end{aligned}$$

and $Q := Q_1 \cup Q_2$. Every (partial) play of this game starts in the designated starting node $q_0 := \varepsilon$. It then passes through some of the states with words of length strictly lesser than k, always increasing the length. Then it stays inside the set of states with words of length equal to k. We define the transition function $\delta\colon (Q_1 \times \Sigma_1) \cup (Q_2 \times \Sigma_2) \to Q$ by

$$\begin{aligned}
\delta(w, a) &:= w\tbinom{a}{*} \text{ , if } |w| < k \\
\delta(\tbinom{b}{x} w, a) &:= w\tbinom{a}{*} \text{ , if } |w| = k - 1 \\
\delta(w\tbinom{a}{*}, x) &:= w\tbinom{a}{x}.
\end{aligned}$$

The acceptance component depends on the words $\alpha_1, \ldots, \alpha_m$, which make up the language L. It consists of sets $\mathcal{F}_i$ for every α_i:

$$\mathcal{F} := \{F_1, \ldots, F_m\}$$

Precisely, Player 2 wins the game if and only if the infinite word that we read is in one of the equivalence classes $[\alpha_i]_{\sim_k^\omega}$. That is the case, iff the prefixes and infixes of our word correspond to the prefixes and infixes of α_i, thus

$$\begin{aligned}
F_i := \quad & \{w \in Q_1 \mid w \in \mathrm{Prefix}_{k-1}(\alpha_i)\} \\
\cup & \{w\tbinom{a}{*} \in Q_2 \mid \exists x \in \Sigma_2\colon w\tbinom{a}{x} \in \mathrm{Prefix}_{k-1}(\alpha_i)\} \\
\cup & \{w \in Q_1 \mid w \in \mathrm{Infix}_k(\alpha_i), |w| = k\} \\
\cup & \{w\tbinom{a}{*} \in Q_2 \mid \exists x \in \Sigma_2\colon w\tbinom{a}{x} \in \mathrm{Infix}_k(\alpha_i), |w\tbinom{a}{*}| = k\}.
\end{aligned}$$

The game $(G, \mathcal{F})$ is a weak Muller game. We know from Theorem 16 that it is determined and that both players have finite-state winning strategies. In order to transfer these strategies back to the original game, we need the content of the finite memory. So we will first map the game to a weak parity game and then transfer the (memoryless) strategies from this weak parity game back to the original game.

In order to translate partial plays in $\mathrm{Ch}(L)$ into partial plays in G, we define a function $g\colon P_1 \cup P_2 \to Q$ by letting

$$g(w) := \begin{cases} v, & \text{if } w = uv, |v| = k; \\ w, & \text{otherwise} \end{cases}$$

for each $w \in P_1 \cup P_2$. It is easy to show that g is a homomorphism. We do it for the case that $w \in P_1$, $w = u\binom{b}{x}v$ for certain $u, v \in P_1, |v| = k - 1$, $b \in \Sigma_1, x \in \Sigma_2$:

$$\begin{aligned}
\delta(g(w), a) &= \delta(g(u\tbinom{b}{x}v), a) \\
&= \delta(\tbinom{b}{x}v, a) \\
&= v\tbinom{a}{*} \\
&= g(u\tbinom{b}{x}v\tbinom{a}{*}) \\
&= g(\gamma(u\tbinom{b}{x}v, a)) \\
&= g(\gamma(w, a)).
\end{aligned}$$

The other cases work analogously.

We carry out the game reduction from this weak Muller game to a weak parity game (G', c), described in Section 2.6. We obtain a new game arena $G' = (Q'_1, Q'_2, \delta')$.

Now for every partial play in $\mathrm{Ch}(L)$ there is a partial play in G and for every partial play in G there exists a corresponding partial play in G'. So we can directly map each node of $\mathrm{Ch}(L)$ to the node in G', where this partial play ends. We call this mapping h and we show, that it is a homomorphism:

Let p_k and p_{k+1} be two states in $\mathrm{Ch}(L)$ that are connected: $\delta(p_k, a) = p_{k+1}$. Then there is a partial play $\varrho = p_0, \dots, p_k, p_{k+1}$ in $\mathrm{Ch}(L)$. By applying g to each state, we obtain a partial play $\varrho' = q_0, \dots, q_k, q_{k+1}$ in G, because g is a homomorphism. By the game reduction to a weak parity game, we obtain a partial play $\varrho'' = r_0, \dots, r_k, r_{k+1}$ in G' with $h(p_k) = r_k$ and $h(p_{k+1}) = r_{k+1}$. But this means $\delta'(h(p_k), a) = h(p_{k+1})$.

Lemma 23. *If Player 1 has a winning strategy in (G', c) from r_0, then Player 1 has a $(k+1)$-locally testable winning strategy in $Ch(L)$.*

Lemma 24. *If Player 2 has a winning strategy in (G', c) from r_0, then Player 2 has a k-locally testable winning strategy in $Ch(L)$.*

Since weak parity games are determined, either Player 1 or Player 2 has a winning strategy in (G', c) from r_0. Then one can use Lemma 23 respectively Lemma 24 to show, that the player who wins, has a $(k+1)$-locally testable winning strategy respectively a k-locally testable winning strategy (which is again $(k+1)$-locally testable).

□

Proof of Lemma 23. Let $f_1 \colon Q'_1 \to \Sigma_1$ be a memoryless winning strategy for Player 1 from q_0 in (G', c). For every $a \in \Sigma_1$ define

$$Q_1^a := \{ q \in Q_1 \mid f_1(q) = a \} \text{ and}$$
$$S_a := \{ w \in P_1 \mid h(w) \in Q_1^a \}.$$

Then clearly the sets Q_1^a are pairwise disjoint and cover Q_1, while the sets S_a are pairwise disjoint and cover P_1. So $(S_a)_{a \in \Sigma_1}$ is a strategy for Player 1.

We still have to show that $(S_a)_{a \in \Sigma_1}$ is winning for Player 1. Let ϱ be a play played according to $(S_a)_{a \in \Sigma_1}$. We want Player 1 to win this play, so we are going to show $\alpha := \alpha(\varrho) \notin L$. Like in the construction above, we let $\varrho = p_0, p_1, \dots$, $\varrho' = q_0, q_1, \dots$ and $\varrho'' = r_0, r_1, \dots$ be the sequence of states of the corresponding plays. So $p_0 = \varepsilon \in P_1$, $p_1 \in P_2$, etc. Since ϱ is played according to $(S_a)_{a \in \Sigma_1}$, for every even $i \in \mathbb{N}$ there is an $a \in \Sigma_1$ and a $w \in P_1$ such that $p_i = w, p_{i+1} = w\binom{a}{*}$. Then clearly $w \in S_a$ and by the definition of S_a and Q_1^a we obtain $h(w) \in Q_1^a$ and $f_1(h(w)) = a$. So ϱ'' is played according to f_1. But f_1 was a winning strategy for Player 1 from r_0, so Player 1 wins ϱ''. Then Player 1 also wins the corresponding play ϱ' in G, so for all $i \in \{1, \dots, m\}$ holds $F_i \neq \mathrm{Occ}(\varrho')$. Then by definition of the set F_i, the set of prefixes and factors of $\alpha(\varrho)$ is not equal to any of the sets of prefixes and factors of α_i and thus Player 1 wins ϱ in $\mathrm{Ch}(L)$. So $(S_a)_{a\in\Sigma_1}$ is indeed winning for Player 1.

For $w_1, w_2 \in P_1$, we show: if $w_1 \sim_{k+1} w_2$ then $w_1 \in S_a \Leftrightarrow w_2 \in S_a$. Let $w_1 \sim_{k+1} w_2$. Then

$$\begin{aligned} \mathrm{Infix}_k(w_1) &= \mathrm{Infix}_k(w_2), \\ \mathrm{Prefix}_{k-1}(w_1) &= \mathrm{Prefix}_{k-1}(w_2), \\ \mathrm{Suffix}_k(w_1) &= \mathrm{Suffix}_k(w_2). \end{aligned}$$

The state $h(w_1)$ has the form $h(w_1) = (v, AR)$ where v is the k-suffix of w_1 (or w_2) and AR is the appearance record of the k-suffixes seen so far, which means

$$\begin{aligned} AR = &\big\{w \in Q_1 \bigm| w \in \mathrm{Prefix}_{k-1}(w_1)\big\} \\ &\cup \big\{w\tbinom{a}{*} \in Q_2 \bigm| \exists x \in \Sigma_2 \colon w\tbinom{a}{x} \in \mathrm{Prefix}_{k-1}(w_1)\big\} \\ &\cup \big\{w \in Q_1 \bigm| w \in \mathrm{Infix}_k(w_1), |w| = k\big\} \\ &\cup \big\{w\tbinom{a}{*} \in Q_2 \bigm| \exists x \in \Sigma_2 \colon w\tbinom{a}{x} \in \mathrm{Infix}_k(w_1), |w\tbinom{a}{*}| = k\big\}. \end{aligned}$$

This set AR is the same for w_1 and w_2.

So $h(w_1) = h(w_2)$ and thus $w_1 \in S_a \Leftrightarrow w_2 \in S_a$. This means, S_a is as a union of $\sim_{k+1}$-classes $(k+1)$-locally testable for every $a \in \Sigma_1$.

So $(S_a)_{a\in\Sigma_1}$ is a $(k+1)$-locally testable winning strategy. □

Proof of Lemma 24. Let $f_2 \colon Q_2' \to \Sigma_2$ be a memoryless winning strategy for Player 2 from q_0 in (G', c). For every $x \in \Sigma_2$ define

$$\begin{aligned} Q_2^x &:= \big\{q \in Q_2 \bigm| f_2(q) = x\big\} \text{ and} \\ T_x &:= \big\{w \in P_2 \bigm| h(w) \in Q_2^x\big\}. \end{aligned}$$

Then clearly the sets Q_2^x are pairwise disjoint and cover Q_2, while the sets T_x are pairwise disjoint and cover P_2. So $(T_x)_{x\in\Sigma_2}$ is a strategy for Player 2.

We show that $(T_x)_{x\in\Sigma_2}$ is winning for Player 2. Let ϱ be a play played according to $(T_x)_{x\in\Sigma_2}$. We want Player 2 to win this play, so we are going to show $\alpha := \alpha(\varrho) \in L$. Like above, we let $\varrho = p_0, p_1, \dots,$ $\varrho' = q_0, q_1, \dots$ and $\varrho'' = r_0, r_1, \dots$ be the sequence of states of the corresponding plays. So $p_0 = \varepsilon \in P_1$, $p_1 \in P_2$, etc. For every odd $i \in \mathbb{N}$ there is an $a \in \Sigma_1$, an $x \in \Sigma_2$ and a $w \in P_1$ such that $\varrho_i = w\binom{a}{*}, \varrho_{i+1} = w\binom{a}{x}$. Since ϱ is played according to $(T_x)_{x\in\Sigma_2}$, clearly $w\binom{a}{*} \in T_x$ and by the definition of T_x and Q_2^x we obtain $h(w) \in Q_2^x$ and $f_2(h(w)) = x$. So ϱ'' is played according to

f_2. But f_2 was a winning strategy for Player 2 from r_0, so Player 2 wins ϱ''. Then Player 2 also wins the corresponding play ϱ' in G, so there exists an $i \in \{1, \ldots, m\}$ with $F_i = \mathrm{Occ}(\varrho')$. Then by definition of the set F_i, the set of prefixes and factors of $\alpha(\varrho)$ is equal to the set of prefixes and factors of α_i and thus Player 2 wins ϱ in $\mathrm{Ch}(L)$. So $(T_x)_{x \in \Sigma_2}$ is indeed winning for Player 2.

For $w_1, w_2 \in P_2$, we show: if $w_1 \sim_k w_2$ then $w_1 \in T_x \Leftrightarrow w_2 \in T_x$. Let $w_1 \sim_k w_2$. Then

$$\begin{aligned} \mathrm{Infix}_k(w_1) &= \mathrm{Infix}_k(w_2), \\ \mathrm{Prefix}_{k-1}(w_1) &= \mathrm{Prefix}_{k-1}(w_2). \end{aligned}$$

The state $h(w_1)$ has the form $h(w_1) = (v, AR)$ where v is the k-suffix of w_1 and AR is the appearance record of the k-suffixes seen so far (including all the suffixes, which end with $\binom{a}{*}$), which means

$$\begin{aligned} AR = & \big\{ w \in Q_1 \;\big|\; w \in \mathrm{Prefix}_{k-1}(w_1) \big\} \\ & \cup \big\{ w\tbinom{a}{*} \in Q_2 \;\big|\; \exists x \in \Sigma_2 \colon w\tbinom{a}{x} \in \mathrm{Prefix}_{k-1}(w_1) \big\} \\ & \cup \big\{ w \in Q_1 \;\big|\; w \in \mathrm{Infix}_k(w_1), |w| = k \big\} \\ & \cup \big\{ w\tbinom{a}{*} \in Q_2 \;\big|\; \exists x \in \Sigma_2 \colon w\tbinom{a}{x} \in \mathrm{Infix}_k(w_1), |w\tbinom{a}{*}| = k \big\}. \end{aligned}$$

So $h(w_2) = (u, AR)$ where AR is the same as for $h(w_1)$ and u is the k-suffix of w_2. Since $\mathrm{Infix}_k(w_1) = \mathrm{Infix}_k(w_2)$, we can extract the only factor from $\mathrm{Infix}_k(w_1)$ (respectively $\mathrm{Infix}_k(w_2)$), which ends in $\binom{a}{*}$ and obtain the k-suffix of w_1 (respectively w_2). So $u = v$ and thus $w_1 \in T_x \Leftrightarrow w_2 \in T_x$. This means, T_x is as a union of $\sim_k$ classes k-locally testable for every $x \in \Sigma_2$.

So $(T_x)_{x \in \Sigma_2}$ is a k-locally testable winning strategy. □

3.3 Strongly Locally Testable Languages

In this section we will slightly modify the class of locally testable languages. We want not only to reckon the set of factors that occur at all in a word, but also on the set of factors that occur infinitely often. We call those languages strongly locally testable.

One could possibly imagine that this extension of the locally testable languages does not lead to weak games anymore. Weak games must not rely on the set $\mathrm{Inf}(\varrho)$ of states occurring infinitely often, so strongly locally testable games should not be won by weak winning strategies.

Strongly Locally Testable $*$-Languages

Factors cannot occur infinitely often in a finite word. So for $*$-languages holds: a $*$-language is strongly locally testable, if it is locally testable.

Strongly Locally Testable ω-Languages

An ω-language is strongly locally testable, if one can test membership of a word to this language by means of a finite table of prefixes and factors like in the locally testable case and additionally by a finite table of infixes that occur infinitely often.

Definition 25. We let

$$\begin{aligned}\mathrm{Infix}_k^\omega(\alpha) := \big\{ w \in \Sigma^* \;\big|\; & |w| \leq k, \forall i \in \mathbb{N}\ \exists u \in \Sigma^*, \beta \in \Sigma^\omega : \\ & |u| > i, \alpha = uw\beta \big\}\end{aligned}$$

be the set of infixes of length less than or equal to k, occurring infinitely often in α.

For $\alpha, \beta \in \Sigma^\omega$ let

$$\begin{aligned}\alpha \mathrel{\widehat{\approx}_k^\omega} \beta \quad :\Longleftrightarrow \quad & \mathrm{Prefix}_{k-1}(\alpha) = \mathrm{Prefix}_{k-1}(\beta) \\ & \wedge \mathrm{Infix}_k(\alpha) = \mathrm{Infix}_k(\beta) \\ & \wedge \mathrm{Infix}_k^\omega(\alpha) = \mathrm{Infix}_k^\omega(\beta).\end{aligned}$$

A language $L \subseteq \Sigma^\omega$ is called *strongly k-locally testable*, if it can be written as a finite union of classes from $\Sigma^\omega / \widehat{\approx}_k^\omega$:

$$L = \bigcup_{i=1}^{m} [\alpha_i]_{\widehat{\approx}_k^\omega}$$

A language $L \subseteq \Sigma^\omega$ is called *strongly locally testable*, if it is strongly k-locally testable for a certain $k \in \mathbb{N}$.

Church's Problem

Proposition 26. *There are strongly locally testable games that do not have locally testable winning strategies.*

Proof. We give a strongly locally testable language, defining an instance of Church's Game. But the player winning this game will not have a locally testable winning strategy.

Let $\Sigma_1 := \{a, b\}$ and $\Sigma_2 := \{0, 1\}$. We set $L \subseteq \Sigma^\omega$ to be the language defined by

$$\begin{aligned}\alpha \in L \quad :\Longleftrightarrow \quad & \mathrm{Infix}_1(\alpha) \subseteq \left\{ \tbinom{a}{0}, \tbinom{b}{0}, \tbinom{b}{1} \right\} \\ & \wedge \tbinom{b}{0} \in \mathrm{Infix}_1^\omega(\alpha) \leftrightarrow \tbinom{b}{1} \in \mathrm{Infix}_1^\omega(\alpha).\end{aligned}$$

We basically stated only boolean combinations of conditions over the set of infixes and the set of infixes occurring infinitely often. So it should be clear that this language is strongly locally testable. However, the sceptical

Word α_i	$\text{Infix}_1(\alpha_i)$	$\text{Infix}_1^\omega(\alpha_i)$
$\alpha_1 = \left[\binom{b}{0}\binom{b}{1}\right]^\omega$	$\{\binom{b}{0}, \binom{b}{1}\}$	$\{\binom{b}{0}, \binom{b}{1}\}$
$\alpha_2 = \binom{a}{0}\left[\binom{b}{0}\binom{b}{1}\right]^\omega$	$\{\binom{a}{0}, \binom{b}{0}, \binom{b}{1}\}$	$\{\binom{b}{0}, \binom{b}{1}\}$
$\alpha_3 = \left[\binom{a}{0}\binom{b}{0}\binom{b}{1}\right]^\omega$	$\{\binom{a}{0}, \binom{b}{0}, \binom{b}{1}\}$	$\{\binom{a}{0}, \binom{b}{0}, \binom{b}{1}\}$
$\alpha_4 = \binom{a}{0}^\omega$	$\{\binom{a}{0}\}$	$\{\binom{a}{0}\}$
$\alpha_5 = \binom{b}{0}\binom{a}{0}^\omega$	$\{\binom{a}{0}, \binom{b}{0}\}$	$\{\binom{a}{0}\}$
$\alpha_6 = \binom{b}{1}\binom{a}{0}^\omega$	$\{\binom{a}{0}, \binom{b}{1}\}$	$\{\binom{a}{0}\}$
$\alpha_7 = \binom{b}{0}\binom{b}{1}\binom{a}{0}^\omega$	$\{\binom{a}{0}, \binom{b}{0}, \binom{b}{1}\}$	$\{\binom{a}{0}\}$

Table 3.1: The equivalence classes for $L = \bigcup_{i=1}^{7} [\alpha_i]_{\approx_k^\omega}$ together with their infixes, occurring finitely and infinitely often.

reader may want to inspect Table 3.1 for a detailed definition of L in the form $L = \bigcup_{i=1}^{m} [\alpha_i]_{\approx_k^\omega}$.

In the first line of the above language specification, we forbid the symbol $\binom{a}{1}$ to occur in the infinite word α. So Player 2 (if she wants to win) will never answer with a 1 when Player 1 provides an a. In the second line, we say that either both $\binom{b}{0}$ and $\binom{b}{1}$ occur infinitely often in α, or none of them. So if Player 1 states infinitely many letters b, then Player 2 needs to alternately specify 0 and 1 as his output. In fact, this is already the winning strategy for Player 2. Player 2 wins this game, if she plays as mentioned.

However, Player 2 has no locally testable winning strategy. Assume (T_0, T_1) is a locally testable winning strategy for Player 2, say it is k-locally testable. Then we let Player 1 choose $(a^{k-1}b)^\omega$ and consider $\varrho = \varrho_0, \varrho_1, \dots$ to be the (unique) play which is played according to (T_0, T_1). Since Player 2 wins this game, $\binom{b}{0}$ and $\binom{b}{1}$ both occur in α. So $\text{Infix}_1(\alpha) = \{\binom{a}{0}, \binom{b}{0}, \binom{b}{1}\}$. There is a point in time l from which onwards the set $\text{Infix}_k(\varrho_i)$ does not change anymore, so $\forall i \geq l, i$ even: $\text{Infix}_k(\varrho_i) = \text{Infix}_k(\alpha)$. Then we look at all the prefixes $w_i = u_i\binom{b}{*}$, $u_i \in P_1$ with length at least l. Each of them has the same sets $\text{Infix}_k(w_i)$, $\text{Prefix}_{k-1}(w_i)$ and $\text{Suffix}_{k-1}(w_i)$. So they all lie inside the same $\sim_k$-class.

But then either all w_i are in T_0 or they are in T_1. Assuming that they are in T_0 yields infinitely many $\binom{b}{0}$ in α but only finitely many $\binom{b}{1}$. Assuming that they are in T_1 yields finitely many $\binom{b}{0}$ in α but infinitely many $\binom{b}{1}$. Both cases imply $\alpha \notin L$, which means that Player 1 wins the game. This is a contradiction to the assumption that (T_0, T_1) is a winning strategy for Player 2. $\square$

3.4 Locally Threshold Testable Languages

The property of local testability describes languages by the presence or absence of factors. So in principle one can count these factors from "0" to "1 or more". This concept is generalized by counting the factors not alone up to 1, but up to an arbitrary threshold r and the corresponding languages are called locally threshold testable languages.

It is known [Tho82] that locally threshold testable $*$-languages are exactly those languages which are definable by first order logic formulae together with the successor function $\mathrm{FO}(S)$. The same holds for locally threshold testable ω-languages.

Rabinovich and Thomas [RT07] already showed for $\mathcal{K}_{\mathrm{Ltt}}$ being the class of all $\mathrm{FO}(S)$ definable $*$-languages and $\mathcal{L}_{\mathrm{Ltt}}$ being the corresponding class of all $\mathrm{FO}(S)$ definable ω-languages, that each $\mathcal{L}_{\mathrm{Ltt}}$-definable game is determined with $\mathcal{K}_{\mathrm{Ltt}}$-definable winning strategies. So each locally threshold testable game is determined with locally threshold testable winning strategies. In the following we will generalize this result for a hierarchy of subclasses of $\mathcal{L}_{\mathrm{Ltt}}$.

Locally Threshold Testable $*$-Languages

A $*$-language is locally threshold testable, if one can test membership of a word to this language by means of a finite table of prefixes and suffixes and counting the occurring factors up to a certain threshold.

Definition 27. For $v \in \Sigma^+$ and $w \in \Sigma^*$ let

$$\mathrm{Fact}(w, v) := \left|\left\{x \in \Sigma^* \;\middle|\; \exists y \in \Sigma^* \colon w = xvy\right\}\right|$$

denote the number of times v occurs as a factor of w. The sets $\mathrm{Prefix}_k(w)$ and $\mathrm{Suffix}_k(w)$ of infixes respectively suffixes of w of length $\leq k$ are defined as in Section 3.2.

For every $k, r \geq 1$, we define an equivalence relation on the words in Σ^*. Let $w_1, w_2 \in \Sigma^*$. Then

$$\begin{aligned} w_1 \approx_r^k w_2 \quad :\Longleftrightarrow \quad & \forall v \in \Sigma^+, |v| \leq k \colon \\ & \quad \min\{\mathrm{Fact}(w_1, v), r\} = \min\{\mathrm{Fact}(w_2, v), r\} \\ & \wedge \mathrm{Prefix}_{k-1}(w_1) = \mathrm{Prefix}_{k-1}(w_2) \\ & \wedge \mathrm{Suffix}_{k-1}(w_1) = \mathrm{Suffix}_{k-1}(w_2). \end{aligned}$$

A language $K \subseteq \Sigma^*$ is called *k-locally r-threshold testable*, if it can be written as a finite union of classes from $\Sigma^*/\!\approx_r^k$:

$$K = \bigcup_{i=1}^{m} [w_i]_{\approx_r^k}$$

A language $K \subseteq \Sigma^*$ is called *locally threshold testable*, if it is k-locally r-threshold testable for certain $k, r \geq 1$.

Remark. For $r = 1$ the factors can only be counted up to one occurrence. So the k-locally 1-threshold testable languages are exactly the k-locally testable languages that we addressed in Section 3.2.

Locally Threshold Testable ω-Languages

An ω-language is locally threshold testable, if one can test membership of a word to this language by means of a finite table of prefixes and counting the occurring factors up to a certain threshold.

Wilke [Wil93] showed that it is decidable whether a given regular ω-language is locally threshold testable. In his work, locally threshold testable ω-languages are also called "finitely locally threshold testable".

Definition 28. For $v \in \Sigma^+$ and $\alpha \in \Sigma^\omega$ let

$$\mathrm{Fact}(\alpha, v) := \left|\left\{x \in \Sigma^* \;\middle|\; \exists \beta \in \Sigma^\omega \colon \alpha = xv\beta\right\}\right|$$

denote the number of times v occurs as a factor of α. Note that $\mathrm{Fact}(\alpha, v) = \infty$ is possible. The sets $\mathrm{Prefix}_k(\alpha)$ and $\mathrm{Suffix}_k(\alpha)$ of infixes respectively suffixes of α of length $\leq k$ are defined as in Section 3.2.

For every $k, r \geq 1$, we define an equivalence relation on the words in Σ^ω. Let $\alpha, \beta \in \Sigma^\omega$. Then

$$\begin{aligned} \alpha \approx_r^{k\omega} \beta \quad :\Longleftrightarrow \quad & \forall v \in \Sigma^+ |v| \leq k \colon \\ & \min\{\mathrm{Fact}(\alpha, v), r\} = \min\{\mathrm{Fact}(\beta, v), r\} \\ & \wedge \mathrm{Prefix}_{k-1}(\alpha) = \mathrm{Prefix}_{k-1}(\beta). \end{aligned}$$

A language $L \subseteq \Sigma^\omega$ is called *k-locally r-threshold testable*, if it can be written as a finite union of classes from $\Sigma^\omega/{\approx_r^{k\omega}}$:

$$L = \bigcup_{i=1}^{m} [\alpha_i]_{\approx_r^{k\omega}}$$

A language $L \subseteq \Sigma^\omega$ is called *locally threshold testable*, if it is k-locally r-threshold testable for certain $k, r \geq 1$.

Church's Problem

Theorem 29. *For every $k, r \geq 1$, k-locally r-threshold testable games are determined with k-locally r-threshold testable winning strategies.*

This theorem also refines Theorem 22. For $r = 1$, we can conclude that k-locally testable games are determined with k-locally testable winning strategies.

Proof. Let $L \subseteq (\Sigma_1 \times \Sigma_2)^\omega$ be an instance of Church's Problem and let L be k-locally r-threshold testable.

Then there are words $\alpha_1, \ldots, \alpha_m \in \Sigma^\omega$ with $L = \bigcup_{i=1}^m [\alpha_i]_{\approx_r^{k\omega}}$.

We define a new weak parity game (G, c) together with its game arena G and a coloring function c.

Only factors of length k are of interest, so let

$$M = \{v \in P_1 \mid 1 \leq |v| \leq k\}.$$

A multiset is now a function $f\colon M \to \{0, \ldots, r\}$, mapping each word v of length at most k to the number of occurrences of this word as a subword in the current partial play. With $\underline{r}^M$ we denote the set of all such multisets. For any multiset f define the cardinality of f to be $|f| = \sum_{v \in M} f(v)$ and an operator "+" with $f + \{v_1, \ldots, v_n\} := f'$ where

$$f'(v) := \begin{cases} \min\{f(v)+1, r\}, & \text{if } \exists 1 \leq i \leq n\colon v_i = v; \\ f(v), & \text{otherwise.} \end{cases}$$

$$\begin{aligned} Q_1 :=& \{(u_{\text{pre}}, f, u_{\text{suf}}) \in P_1 \times \underline{r}^M \times P_1 \mid |u_{\text{pre}}| = |u_{\text{suf}}| = k-1\} \\ &\cup \{w \in P_1 \mid |w| \leq k-1\}, \\ Q_2 :=& \{(u_{\text{pre}}, f, u_{\text{suf}}) \in P_1 \times \underline{r}^M \times P_2 \mid |u_{\text{pre}}| = k-1, |u_{\text{suf}}| = k\} \\ &\cup \{w \in P_2 \mid |w| \leq k\}. \end{aligned}$$

Then $Q := Q_1 \cup Q_2$ is the set of nodes of the game arena.

For nodes from Q_1, the transition function δ is defined by

$$\delta(w, a) := w\tbinom{a}{*} \text{ and}$$
$$\delta((u_{\text{pre}}, f, u_{\text{suf}}), a) := (u_{\text{pre}}, f, u_{\text{suf}}\tbinom{a}{*}).$$

For nodes from Q_2, it is

$$\delta(w\tbinom{a}{*}, x) := w\tbinom{a}{x} \text{ , if } |w\tbinom{a}{*}| \leq k-1$$
$$\delta(\tbinom{a}{x}w\tbinom{b}{*}, y) := (\tbinom{a}{x}w, f, w\tbinom{b}{y}) \text{ , if } |\tbinom{a}{x}w\tbinom{b}{*}| = k \geq 2$$

with $f(v) = \min\{\text{Fact}(\tbinom{a}{x}w\tbinom{b}{y}, v), r\}$,

$$\delta(\tbinom{a}{*}, x) := (\tbinom{a}{x}, f, \tbinom{a}{x}) \text{ , if } k = 1$$

with $f(v) = \min\{\text{Fact}(\tbinom{a}{x}, v), r\}$ and

$$\delta((u_{\text{pre}}, f, \tbinom{a}{x}u_{\text{suf}}\tbinom{b}{*}), y) := (u_{\text{pre}}, f', u_{\text{suf}}\tbinom{b}{y})$$

with $f' = f + \text{Suffix}_k(\tbinom{a}{x}u_{\text{suf}}\tbinom{b}{y})$.

In order to translate plays $\varrho \in (\Sigma_1 \times \Sigma_2)^\omega$ into plays $\varrho' \in Q^\omega$, we define a function $h\colon P \to Q$ by letting

$$\begin{aligned} h(w) &:= w \text{ if } |w| \leq k-1 \text{ or } |w| = k, w \in P_2 \\ h(w) &:= (u_{\text{pre}}^{k-1}, f, u_{\text{suf}}^{k-1}) \text{ if } w \in P_1, |w| \geq k \\ h(w\tbinom{a}{*}) &:= (u_{\text{pre}}^{k-1}, f, u_{\text{suf}}^{k-1}\tbinom{a}{*}) \text{ else.} \end{aligned}$$

with u_{pre}^{k-1} being the $k-1$-prefix of w, u_{suf}^{k-1} being the $k-1$-suffix of w and f counts the k-factors up to threshold r of the word w.

The function h is a homomorphism. We shall show this for the case that $w \in P_2$, $|w| \geq k+1$ and $k \geq 2$:

Let $w = u\binom{b}{y}v\binom{a}{*}$ with $|v| = k-2$ and let u_{pre} be the $k-1$-prefix of w. Then

$$\begin{aligned} \delta(h(w), x) &= \delta(h(u\tbinom{b}{y}v\tbinom{a}{*}), x) \\ &= \delta((u_{\text{pre}}, f, \tbinom{b}{y}v\tbinom{a}{*}), x) \\ &\quad \text{with } f(m) = \text{Fact}(w, m) \text{ for every } m \in M \\ &= (u_{\text{pre}}, f', v\tbinom{a}{*}) \\ &\quad \text{with } f' = f + \text{Suffix}_k(\tbinom{b}{y}v\tbinom{a}{x}) \\ &= h(u\tbinom{b}{y}v\tbinom{a}{x}) \\ &= h(\gamma(u\tbinom{b}{y}v\tbinom{a}{*}, x)) \\ &= h(\gamma(w, x)). \end{aligned}$$

The other cases work analogously.

The acceptance component depends on the words $\alpha_1, \ldots, \alpha_m$, which make up the language L. It consists of the coloring function $c\colon Q \to \big\{0, \ldots, 2r \cdot (|(\Sigma_1 \times \Sigma_2)|^{k+1} - 1)\big\}$. We map every node that is not of the form $(u_{\text{pre}}, f, u_{\text{suf}})$ to the color 0. For the other nodes we set

$$c((u_{\text{pre}}, f, u_{\text{suf}})) := \begin{cases} 2 \cdot |f|, & \text{if } \exists i \in \big\{1, \ldots, m\big\}\colon u_{\text{pre}} \text{ is prefix of } \alpha_i, \\ & \forall u \in M\colon f(u) = \min\{\text{Fact}(\alpha_i, u), r\}; \\ 2 \cdot |f| - 1, & \text{otherwise.} \end{cases}$$

We set $q_0 = h(\varepsilon)$. To show that this construction indeed gives us the desired result, we state the following lemmas.

Lemma 30. *If Player 1 has a winning strategy in (G, c) from q_0, then Player 1 has a k-locally r-threshold testable winning strategy in $Ch(L)$.*

Lemma 31. *If Player 2 has a winning strategy in (G, c) from q_0, then Player 2 has a k-locally r-threshold testable winning strategy in $Ch(L)$.*

The game (G, c) is a weak parity game. Weak parity games are determined, so one of the players has a winning strategy from q_0, which moreover is memoryless. In the case that Player 1 has a winning strategy use Lemma 30, in the other case use Lemma 31 to prove the result. □

Proof of Lemma 30. Let $f_1 \colon Q_1 \to \Sigma_1$ be a memoryless winning strategy for Player 1 from q_0 in (G, c). For every $a \in \Sigma_1$ define

$$\begin{aligned} Q_1^a &:= \{ q \in Q_1 \mid f_1(q) = a \} \text{ and} \\ S_a &:= \{ w \in P_1 \mid h(w) \in Q_1^a \}. \end{aligned}$$

Then clearly the sets Q_1^a are pairwise disjoint and cover Q_1, while the sets S_a are pairwise disjoint and cover P_1. So $(S_a)_{a \in \Sigma_1}$ is a strategy for Player 1.

We still have to show that $(S_a)_{a \in \Sigma_1}$ is winning for Player 1. Let ϱ be a play played according to $(S_a)_{a \in \Sigma_1}$. We want Player 1 to win this play, so we are going to show that $\alpha := \alpha(\varrho)$ is not in L. Let $\varrho = p_0, p_1, \dots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \dots$ in (G, c). For every even $i \in \mathbb{N}$ there is an $a \in \Sigma_1$ and a $w \in P_1$ such that $p_i = w, p_{i+1} = w\binom{a}{*}$. Since ϱ is played according to $(S_a)_{a \in \Sigma_1}$, clearly $w \in S_a$ and by the definition of S_a and Q_1^a we obtain $h(w) \in Q_1^a$ and $f_1(h(w)) = a$. So ϱ' is played according to f_1. But f_1 was a winning strategy for Player 1 from q_0, so Player 1 wins ϱ'.

Then the maximal color $C(\varrho')$ occurring in ϱ' is odd. At some point of ϱ', this color d has to occur. Let j be the smallest index with $c(q_j) = d$. Then for all following nodes, the color stays at d, because the multiset f cannot shrink during a play. But then the multiset f stays constant from j onwards and so does u_{pre}. Then u_{pre} and f also describe the prefix and the multiplicity of factors of α.

Since d is odd, there is no α_i such that u_{pre} is prefix of α_i and $\forall u \in M \colon f(u) = \min\{\text{Fact}(\alpha_i, u), r\}$. So $\alpha \notin L$.

Let us show that each S_a is k-locally r-threshold testable. For $w_1, w_2 \in P_1$, we show: if $w_1 \approx_r^k w_2$ then $w_1 \in S_a \Leftrightarrow w_2 \in S_a$. Let $w_1 \approx_r^k w_2$. Then

$$\begin{aligned} &\min\{\text{Fact}(w_1, v), r\} = \min\{\text{Fact}(w_2, v), r\} \\ \wedge\ &\text{Prefix}_{k-1}(w_1) = \text{Prefix}_{k-1}(w_2) \\ \wedge\ &\text{Suffix}_{k-1}(w_1) = \text{Suffix}_{k-1}(w_2) \end{aligned}$$

and so $h(w_1) = (u_{\text{pre}}, f, u_{\text{suf}}) = h(w_2)$ where u_{pre} is the longest word in $\text{Prefix}_{k-1}(w_1)$, u_{suf} is the longest word in $\text{Suffix}_{k-1}(w_1)$ and f is the function with the property $f(v) = \min\{\text{Fact}(w_1, v), r\}$ for every $v \in M$.

This means $w_1 \in S_a \Leftrightarrow w_2 \in S_a$ and S_a is as a union of $\approx_r^k$ classes k-locally r-threshold testable. So $(S_a)_{a \in \Sigma}$ is a k-locally r-threshold testable winning strategy. □

Proof of Lemma 31. The proof essentially follows the one of Lemma 30.

Let $f_2\colon Q_2 \to \Sigma$ be a memoryless winning strategy for Player 2 from q_0 in (G, c). For every $x \in \Sigma_2$ define

$$Q_2^x = \{q \in Q_2 \mid f_2(q) = x\} \text{ and}$$
$$T_x = \{w \in P_2 \mid h(w) \in Q_2^x\}.$$

Then clearly the sets Q_2^x are pairwise disjoint and cover Q_2, while the sets T_x are pairwise disjoint and cover P_2. So $(T_x)_{x\in\Sigma_2}$ is a strategy for Player 2.

We still have to show that $(T_x)_{x\in\Sigma_2}$ is winning for Player 2. Let ϱ be a play played according to $(T_x)_{x\in\Sigma_2}$. We want Player 2 to win this play, so we are going to show that $\alpha := \alpha(\varrho)$ is in L. Let $\varrho = p_0, p_1, \ldots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \ldots$ in (G, c). For every odd $i \in \mathbb{N}$ there is an $a \in \Sigma_1$, an $x \in \Sigma_2$ and a $w \in P_1$ such that $\varrho_i = w\binom{a}{*}$, $\varrho_{i+1} = w\binom{a}{x}$. Since ϱ is played according to $(T_x)_{x\in\Sigma_2}$, clearly $w \in T_x$ and by the definition of T_x and Q_2^x we obtain $h(w) \in Q_2^x$ and $f_2(h(w)) = x$. So ϱ' is played according to f_1. But f_2 is a winning strategy for Player 2 from q_0, so Player 2 wins ϱ'.

Then the maximal color $C(\varrho')$ occurring in ϱ' is even. At some point of ϱ', this color d has to occur. Let j be the smallest index with $c(q_j) = d$. Then for all following nodes, the color stays at d, because the multiset f cannot shrink during a play. But then the multiset f stays constant from j onwards and so does u_{pre}. Then u_{pre} and f also describe the prefix and the multiplicity of factors of α.

Since d is even, there is an α_i such that u_{pre} is prefix of α_i and $\forall u \in M\colon f(u) = \min\{\text{Fact}(\alpha_i, u), r\}$. So $\alpha \in L$.

For $w_1, w_2 \in P_2$, we show: if $w_1 \approx_r^k w_2$ then $w_1 \in T_x \Leftrightarrow w_2 \in T_x$. Let $w_1 \approx_r^k w_2$. Then

$$\begin{aligned}
&\min\{\text{Fact}(w_1, v), r\} = \min\{\text{Fact}(w_2, v), r\}\\
\wedge\ &\text{Prefix}_{k-1}(w_1) = \text{Prefix}_{k-1}(w_2)\\
\wedge\ &\text{Suffix}_{k-1}(w_1) = \text{Suffix}_{k-1}(w_2)
\end{aligned}$$

and so $h(w_1) = (u_{\text{pre}}, f, u_{\text{suf}}) = h(w_2)$ where u_{pre} is the longest word in $\text{Prefix}_{k-1}(w_1)$, u_{suf} is the longest factor of w_1 that ends with a letter $\binom{a}{*}$ and f is the function with $f(v) = \min\{\text{Fact}(w_1, v), r\}$ for every $v \in M$.

This means $w_1 \in T_x \Leftrightarrow w_2 \in T_x$ and T_x is as a union of $\approx_r^k$ classes k-locally r-threshold testable. So $(T_x)_{x\in\Sigma_2}$ is a k-locally r-threshold testable winning strategy. □

3.5 Piecewise Testable Languages

In the case of piecewise testable languages, we no longer consider factors, but so-called (scattered) subwords in order to decide whether a word belongs to the language. A subword is simply a subsequence of letters of a word such that the order of these letters is preserved. For example *abc* is a subword of *adbbece*. But *abcd* is not a subword of it.

Piecewise Testable $*$-Languages

Definition 32. A word $u = a_1 \cdots a_n \in \Sigma^*$ is called a *subword* of $w \in \Sigma^*$, if there are $w_0, \ldots, w_n \in \Sigma^*$, such that

$$w_0 a_1 w_1 \cdots w_{n-1} a_n w_n = w.$$

For $u \in \Sigma^*$ we define the set

$$\mathrm{Subwords}_k(u) := \big\{ w \in \Sigma^* \bigm| |w| \leq k, w \text{ is subword of } u \big\}$$

and for $M \subseteq \Sigma^*$ we set

$$\mathrm{Subwords}_k(M) := \bigcup_{u \in M} \mathrm{Subwords}_k(u).$$

For every $k \geq 0$ let $\sim_k \subseteq \Sigma^* \times \Sigma^*$ be the equivalence relation, defined by

$$u \sim_k v \quad :\Longleftrightarrow \quad \mathrm{Subwords}_k(u) = \mathrm{Subwords}_k(v).$$

A language $K \subseteq \Sigma^*$ is called *k-piecewise testable*, if it can be written as a finite union of classes from $\Sigma^*/{\sim_k}$:

$$K = \bigcup_{i=1}^{m} [w_i]_{\sim_k}$$

A language $K \subseteq \Sigma^*$ is called *piecewise testable*, if it is k-piecewise testable for a certain $k \in \mathbb{N}$.

Obviously for every $w \in \Sigma^*$, $a \in \Sigma$ the equation

$$\mathrm{Subwords}_k(\mathrm{Subwords}_k(w) \cdot a) = \mathrm{Subwords}_k(w \cdot a) \tag{3.1}$$

holds. We will need this equation later on.

Example 33. The language $K_1 = a(a + b)^*$ is locally testable. We have already seen that in Section 3.2. But it is not piecewise testable. Assume it is k-piecewise testable. Then the word $(ab)^{k+1}$ has the same set of subwords as $(ba)^{k+1}$. So they must be equivalent, but $(ab)^{k+1} \in K_1$ and $(ba)^{k+1} \notin K_1$. This proves that K_1 cannot be piecewise testable.

There are many different characterizations of the piecewise testable $*$-languages, most of which have been found by Simon [Sim75]. Simon's Theorem [Pin86, Sim75] is a well-known characterization of these languages by their syntactic monoids.

Piecewise Testable ω-Languages

Definition 34. A word $u = a_1 \cdots a_n \in \Sigma^*$ is called a *subword* of $\alpha \in \Sigma^\omega$, if there are $w_0, \ldots, w_{n-1} \in \Sigma^*$ and $\beta \in \Sigma^\omega$, such that

$$w_0 a_1 w_1 \cdots w_{n-1} a_n \beta = \alpha.$$

For $\alpha \in \Sigma^\omega$ we define the set

$$\mathrm{Subwords}_k(\alpha) = \{ w \in \Sigma^* \mid w \text{ is subword of } \alpha \wedge |w| \leq k \}.$$

For every $k \geq 0$ let $\sim_k^\omega \subseteq \Sigma^\omega \times \Sigma^\omega$ be the equivalence relation, defined by

$$\alpha \sim_k^\omega \beta \quad :\Longleftrightarrow \quad \mathrm{Subwords}_k(\alpha) = \mathrm{Subwords}_k(\beta).$$

A language $L \subseteq \Sigma^\omega$ is called *k-piecewise testable*, if it can be written as a finite union of classes from $\Sigma^\omega/\sim_k^\omega$:

$$L = \bigcup_{i=1}^{m} [\alpha_i]_{\sim_k^\omega}$$

A language $L \subseteq \Sigma^\omega$ is called *piecewise testable*, if it is k-piecewise testable for a certain $k \in \mathbb{N}$.

Church's Problem

Let us look at Church's Problem for piecewise testable languages.

Theorem 35. *For every $k \in \mathbb{N}$, k-piecewise testable games are determined with k-piecewise testable winning strategies.*

Proof. Let $L \subseteq (\Sigma_1 \times \Sigma_2)^\omega$ be an instance of Church's Problem and let L be k-piecewise testable.

Then there are words $\alpha_1, \ldots, \alpha_m \in \Sigma^\omega$ with $L = \bigcup_{i=1}^m [\alpha_i]_{\sim_k^\omega}$.

We define a new weak parity game (G, c) together with its game arena G and a coloring function c.

In each node of the game arena we store the set of subwords of length $\leq k$ which are already read. When a new letter a is read, then the set of subwords is extended by all subwords which result from the original subwords by appending the letter a to them.

So each node is a set of subwords of length $\leq k$. The nodes of Player 1 only store subwords with letters $\binom{a}{x}$ in them, while the nodes of Player 2 may end with a letter $\binom{a}{*}$. We define

$$Q_1 := \{ M \in \mathcal{P}(P_1) \mid \exists w \in P_1 \colon \mathrm{Subwords}_k(w) = M \} \text{ and}$$
$$Q_2 := \{ M \in \mathcal{P}(P_2) \mid \exists w \in P_2 \colon \mathrm{Subwords}_k(w) = M \}.$$

Then $Q := Q_1 \cup Q_2$ is the set of nodes of the game arena.

So for every $q \in Q$ there is a word $w_q \in P_1 \cup P_2$ with $\mathrm{Subwords}_k(w_q) = q$.

We define the transition function δ as

$$\begin{aligned} \delta\colon (Q_1 \times \Sigma_1) \cup (Q_2 \times \Sigma_2) &\to Q \\ \delta(M, a) &:= \mathrm{Subwords}_k(M \cdot \tbinom{a}{*}) \\ \delta(M, x) &:= \left\{ u \cdot \tbinom{a}{x} \in P_1 \;\middle|\; u \cdot \tbinom{a}{*} \in M \right\} \cup (M \cap P_1) \end{aligned}$$

Note that $\delta(M, x)$ effectively is M with every $*$ substituted by x.

The game arena $G = (Q_1, Q_2, \delta)$ is now complete.

In order to translate plays $\varrho \in (\Sigma_1 \times \Sigma_2)^\omega$ into plays $\varrho' \in Q^\omega$, we define a function $h\colon P_1 \cup P_2 \to Q$ by letting

$$h(w) := \mathrm{Subwords}_k(w)$$

for each $w \in P_1 \cup P_2$.

The function h is a homomorphism in the sense that it respects the transition function δ. This can be seen by observing the two following equations.

$$\begin{aligned} h(\gamma(w, a)) &= h(w\tbinom{a}{*}) \\ &= \mathrm{Subwords}_k(w\tbinom{a}{*}) \\ &= \mathrm{Subwords}_k(\mathrm{Subwords}_k(w)\tbinom{a}{*}) \quad \text{(Eq. 3.1)} \\ &= \delta(\mathrm{Subwords}_k(w), a) \\ &= \delta(h(w), a) \end{aligned}$$

$$\begin{aligned} h(\gamma(w\tbinom{a}{*}, x)) &= h(w\tbinom{a}{x}) \\ &= \mathrm{Subwords}_k(w\tbinom{a}{x}) \\ &= \mathrm{Subwords}_k(w\tbinom{a}{x}) \cup \mathrm{Subwords}_k(w) \\ &= \mathrm{Subwords}_k(\mathrm{Subwords}_k(w)\tbinom{a}{x}) \cup \mathrm{Subwords}_k(w) \quad \text{(Eq. 3.1)} \\ &= \left\{ u\tbinom{a}{x} \in P_1 \;\middle|\; u\tbinom{a}{*} \in h(w\tbinom{a}{*}) \right\} \cup (h(w\tbinom{a}{*}) \cap P_1) \\ &= \delta(\mathrm{Subwords}_k(w\tbinom{a}{*}), x) \\ &= \delta(h(w\tbinom{a}{*}), x) \end{aligned}$$

We equip the game arena $G = (Q_1, Q_2, \delta)$ with a coloring function $c\colon Q \to \left\{0, \dots, 2 \cdot (|(\Sigma_1 \times \Sigma_2)|^{k+1} - 1)\right\}$ by letting

$$c(M) := \begin{cases} 0, & \text{if } M \in Q_2; \\ 2 \cdot |M|, & \text{if } \exists i \in \left\{1, \dots, m\right\}\colon M = \mathrm{Subwords}_k(\alpha_i); \\ 2 \cdot |M| - 1, & \text{otherwise} \end{cases}$$

and therefore we obtain a weak parity game. Observe that all nodes of Player 2 have color 0 and do not play any role in $C(\varrho)$. There is a designated starting node $q_0 := \{\varepsilon\} \in Q_1$ for all plays. As we know from Theorem 8, weak parity games are determined with positional winning strategies. This result can be used to prove the following two lemmas.

Lemma 36. *If Player 1 has a winning strategy in (G, c) from q_0, then Player 1 has a k-piecewise testable winning strategy in $Ch(L)$.*

Lemma 37. *If Player 2 has a winning strategy in (G, c) from q_0, then Player 2 has a k-piecewise testable winning strategy in $Ch(L)$.*

Since weak parity games are determined, either Player 1 or Player 2 has a winning strategy in (G, c) from q_0. Then one can use Lemma 36 respectively Lemma 37 to show, that the player who wins, has a k-piecewise testable winning strategy. □

Proof of Lemma 36. Let $f_1 \colon Q_1 \to Q$ be a positional winning strategy in (G, c). For every $a \in \Sigma_1$ define $Q_1^a := \{q \in Q_1 \mid f_1(q) = \delta(q, a)\}$ to be the set of nodes in which Player 1 chooses the letter a. Clearly the sets Q_1^a are pairwise disjoint for $a \in \Sigma_1$, each Q_1^a is finite and the union of these sets covers Q_1. Then for every $a \in \Sigma_1$ the set

$$S_a := \bigcup_{q \in Q_1^a} [w_q]_{\sim_k}$$

is k-piecewise testable.

Let us show that $(S_a)_{a\in\Sigma_1}$ is really a strategy for Player 1. For every $w \in P_1$ we take a look at $q' := h(w)$. Let $a \in \Sigma_1$ such that $q' \in Q_1^a$. Then $w \in [w]_{\sim_k} = [w_{q'}]_{\sim_k} \subseteq \bigcup_{q\in Q_1^a}[w_q]_{\sim_k}$. Assume $w \in P_1$ is in the sets S_a and S_b. Then $w \in [w_{q_1}]_{\sim_k} \cap [w_{q_2}]_{\sim_k}$ for $q_1 \in Q_1^a$ and $q_2 \in Q_1^b$. But then $h(w_{q_1}) = h(w_{q_2})$ and therefore $Q_1^a = Q_1^b$ and $S_a = S_b$. So $(S_a)_{a\in\Sigma_1}$ is really a strategy for Player 1.

Furthermore, for all $a \in \Sigma_1$ and all $w \in P_1$ the equivalence

$$w \in S_a \Leftrightarrow h(w) \in Q_1^a \tag{3.2}$$

holds.

So $(S_a)_{a\in\Sigma_1}$ is a k-piecewise testable strategy. We still have to show that $(S_a)_{a\in\Sigma_1}$ is winning for Player 1. Let ϱ be a play played according to $(S_a)_{a\in\Sigma_1}$. We want Player 1 to win this play, so we are going to show $\alpha := \alpha(\varrho) \notin L$. Let $\varrho_0, \varrho_1, \ldots$ be the sequence of prefixes of ϱ. So $\varrho_0 = \varepsilon \in P_1$, $\varrho_1 \in P_2$, etc. We apply h to every ϱ_i and obtain a sequence $\varrho' = h(\varrho_0), h(\varrho_1), \ldots = q_0, q_1, \ldots$ of states from Q. This sequence is a play of (G, c), because h is a homomorphism. It holds $h(\varrho_0) = q_0$.

Since ϱ is played according to $(S_a)_{a\in\Sigma_1}$, for every even $i \in \mathbb{N}$ there is an $a \in \Sigma_1$ and a $w \in P_1$ such that $\varrho_i = w, \varrho_{i+1} = w \cdot \binom{a}{*}$. Then clearly $w \in S_a$ and with Equivalence (3.2) we obtain $h(w) \in Q_1^a$. But then $f_1(h(w)) = \delta(h(w), a)$.

Then we get

$$\begin{aligned}f_1(q_i) &= f_1(h(w))\\ &= \delta(h(w), a)\\ &= h(w \cdot \tbinom{a}{*})\\ &= h(\varrho_{i+1})\\ &= q_{i+1}.\end{aligned}$$

So ϱ' is played according to f_1 and starts in q_0, however f_1 was a winning strategy for Player 1 from q_0, so Player 1 wins ϱ'. That means $d := \max(\mathrm{Occ}(c(\varrho')))$ is odd. At some point of ϱ', this color d has to occur. The set of subwords for words from P_1 can only grow as we proceed through ϱ, so for all even $i \in \mathbb{N}$: $h(\varrho_i) \subseteq h(\varrho_{i+2})$. All colors occurring before d are less than or equal to d, all colors occurring after d are d and 0 in alternation. According to the definition of c, this means that eventually the set of subwords for Player 1 does not change anymore, so there is an $l \in \mathbb{N}$, such that for all even $j \geq l$: $M := \mathrm{Subwords}_k(\varrho_j)$. Since d is odd, $\neg\exists i \in \{1, \dots, m\}\colon M = \mathrm{Subwords}_k(\alpha_i)$ holds. So $\mathrm{Subwords}_k(\alpha) = \bigcup_{j\in\mathbb{N}, j \text{ even}} \mathrm{Subwords}_k(\varrho_j) = M$ and $\forall i \in \{1, \dots, m\}\colon M \neq \mathrm{Subwords}_k(\alpha_i)$. This means $\alpha \notin L$. □

Proof of Lemma 37. Let $f_2\colon Q_2 \to Q$ be a positional winning strategy in (G, c). For every $x \in \Sigma_2$ define $Q_2^x := \{q \in Q_2 \mid f_2(q) = \delta(q, x)\}$. Please note that here Q_2^x and Q_2^y may be not disjoint for distinct $x, y \in \Sigma_2$. We make them pairwise disjoint by arbitrarily assigning their intersection to one of the sets. Then the union of all sets Q_2^x still covers Q_2 and each Q_2^x is finite. For every $x \in \Sigma_2$ the set

$$T_x := \bigcup_{q\in Q_2^x} [w_q]_{\sim_k}$$

is k-piecewise testable.

Let us show that $(T_x)_{x\in\Sigma_2}$ is really a strategy for Player 2. For every $w \in P_2$ we take a look at $q' := h(w)$. Let $x \in \Sigma_2$ such that $q' \in Q_2^x$. Then $w \in [w]_{\sim_k} = [w_{q'}]_{\sim_k} \subseteq \bigcup_{q\in Q_2^x}[w_q]_{\sim_k}$. Assume $w \in P_2$ is in the sets T_x and T_y. Then $w \in [w_{q_1}]_{\sim_k} \cap [w_{q_2}]_{\sim_k}$ for $q_1 \in Q_2^x$ and $q_2 \in Q_2^y$. But then $h(w_{q_1}) = h(w_{q_2})$ and therefore $Q_2^x = Q_2^y$ and $T_x = T_y$. So $(T_x)_{x\in\Sigma_2}$ is really a strategy for Player 2.

Furthermore, for all $x \in \Sigma_2$ and all $w \in P_2$ the equivalence

$$w \in T_x \Leftrightarrow h(w) \in Q_2^x \tag{3.3}$$

holds.

So $(T_x)_{x\in\Sigma_2}$ is a piecewise testable strategy. We still have to show that $(T_x)_{x\in\Sigma_2}$ is winning for Player 2. Let ϱ be a play played according to $(T_x)_{x\in\Sigma_2}$. We want Player 2 to win this play, so we are going to show $\alpha := \alpha(\varrho) \in L$. Let $\varrho_0, \varrho_1, \ldots$ be the sequence of prefixes of ϱ. So $\varrho_0 = \varepsilon \in P_1$, $\varrho_1 \in P_2$, etc. We apply h to every ϱ_i and obtain a sequence $\varrho' = h(\varrho_0), h(\varrho_1), \ldots = q_0, q_1, \ldots$ of states from Q. This sequence is a play of (G, c), because h is a homomorphism. It holds $h(\varrho_0) = q_0$.

Since ϱ is played according to $(T_x)_{x\in\Sigma_2}$, for every odd $i \in \mathbb{N}$ there is an $a \in \Sigma_1$, an $x \in \Sigma_2$ and a $w \in P_1$ such that $\varrho_i = w\binom{a}{*}, \varrho_{i+1} = w\binom{a}{x}$. Then clearly $w\binom{a}{*} \in T_x$ and with Equivalence (3.3) we obtain $h(w\binom{a}{*}) \in Q_2^x \Rightarrow f_2(h(w\binom{a}{*})) = \delta(h(w\binom{a}{*}), x)$.

Then we get

$$\begin{aligned} f_2(q_i) &= f_2(h(w \cdot \tbinom{a}{*})) \\ &= \delta(h(w\tbinom{a}{*}), x) \\ &= h(w\tbinom{a}{x}) \\ &= h(\varrho_{i+1}) \\ &= q_{i+1}. \end{aligned}$$

So ϱ' is played according to f_2 and starts in q_0, however f_2 was a winning strategy for Player 2 from q_0, so Player 2 wins ϱ'. That means $d := \max(\mathrm{Occ}(c(\varrho')))$ is even. At some point of ϱ', this color d has to occur. The set of subwords for words from P_1 can only grow as we proceed through ϱ, so for all even $i \in \mathbb{N}$: $h(\varrho_i) \subseteq h(\varrho_{i+2})$. All colors occurring before d are less than or equal to d, all colors occurring after d are d and 0 in alternation. According to the definition of c, this means that eventually the set of subwords for Player 1 does not change anymore, so there is an $l \in \mathbb{N}$, such that for all even $j \geq l$: $M := \mathrm{Subwords}_k(\varrho_j)$. Since d is even, $\exists i \in \{1, \ldots, m\}$: $M = \mathrm{Subwords}_k(\alpha_i)$ holds. So $\mathrm{Subwords}_k(\alpha) = \bigcup_{j\in\mathbb{N}, j \text{ even}} \mathrm{Subwords}_k(\varrho_j) = M$ and $\exists i \in \{1, \ldots, m\}$: $M = \mathrm{Subwords}_k(\alpha_i)$.

This means $\alpha \in L$. □

3.6 Piecewise Threshold Testable Languages

Piecewise Threshold Testable ∗-Languages

A ∗-language is piecewise threshold testable, if one can test membership of a word to this language by counting the occurring subwords up to a certain threshold.

Definition 38. An *occurrence* of a word $v = b_1 \cdots b_t$ of length $t \geq 1$ in another word $w = a_1 \cdots a_n$ is a t-tuple $(i_1, \ldots, i_t)$ of increasing positive integers with $i_t \leq n$ and $a_{i_j} = b_j$ for all $1 \leq j \leq t$. In this context, i_j is called a *position* of the letter a_{i_j} in the word w.

Now it is easy to specify the number of occurrences of v as a subword of w. In the literature there are many different notations for this. In [CK96] the function that maps v on this number is also referred to as the *spectrum* of w, in [Sal03] the number of occurrences of v in w is denoted $|w|_v$, and sometimes the notation $\binom{w}{v}$ can be found. We shall call it $\mathrm{Subw}(w, v)$.

Definition 39. For $v = v_1, \ldots, v_l \in \Sigma^+$ and $w \in \Sigma^*$ let

$$\mathrm{Subw}(w, v) := \left|\left\{(i_1, \ldots, i_t) \in \mathbb{N}^t \;\middle|\; (i_1, \ldots, i_t) \text{ is an occurrence of } v \text{ in } w\right\}\right|$$

denote the number of times v occurs as a subword of w.

For every $k, r \geq 1$, we define an equivalence relation on the words from Σ^*. Let $w_1, w_2 \in \Sigma^*$. Then

$$\begin{aligned} w_1 \approx_r^k w_2 \quad :\Longleftrightarrow \quad & \forall u \in \Sigma^+, |u| \leq k\colon \\ & \min\{\mathrm{Subw}(w_1, u), r\} = \min\{\mathrm{Subw}(w_2, u), r\}. \end{aligned}$$

A language $K \subseteq \Sigma^*$ is called *k-piecewise r-threshold testable*, if it can be written as a finite union of classes from $\Sigma^*/{\approx_r^k}$:

$$K = \bigcup_{i=1}^{m} [w_i]_{\approx_r^k}$$

A language $K \subseteq \Sigma^*$ is called *piecewise threshold testable*, if it is k-piecewise r-threshold testable for certain $k, r \geq 1$.

Remark. For $r = 1$ the subwords can only be counted up to one occurrence. So the k-piecewise 1-threshold testable languages are exactly the k-piecewise testable languages that we addressed in Section 3.5.

Piecewise Threshold Testable ω-Languages

An ω-language is piecewise threshold testable, if one can test membership of a word to this language by counting the occurring subwords up to a certain threshold.

Definition 40. For $v = v_1, \ldots, v_l \in \Sigma^+$ and $\alpha \in \Sigma^\omega$ let

$$\mathrm{Subw}(\alpha, v) := \left|\left\{(i_1, \ldots, i_t) \in \mathbb{N}^t \;\middle|\; (i_1, \ldots, i_t) \text{ is an occurrence of } v \text{ in } \alpha\right\}\right|$$

denote the number of times v occurs as a subword of α. Note that $\mathrm{Subw}(\alpha, v) = \infty$ is possible.

For every $k \geq 1$, we define an equivalence relation on the words from Σ^ω. Let $\alpha, \beta \in \Sigma^\omega$. Then

$$\begin{aligned} \alpha \approx_r^{k\omega} \beta \quad :\Longleftrightarrow \quad & \forall v \in \Sigma^+, |v| \leq k\colon \\ & \min\{\mathrm{Subw}(\alpha, v), r\} = \min\{\mathrm{Subw}(\alpha, v), r\}. \end{aligned}$$

A language $L \subseteq \Sigma^\omega$ is called *k-piecewise r-threshold testable*, if it can be written as a finite union of classes from $\Sigma^\omega/{\approx_r^{k\omega}}$:

$$L = \bigcup_{i=1}^{m} [\alpha_i]_{\approx_r^{k\omega}}$$

A language $L \subseteq \Sigma^\omega$ is called *piecewise threshold testable*, if it is k-piecewise r-threshold testable for certain $k, r \geq 1$.

Church's Problem

When examining piecewise threshold testable languages in the context of already known piecewise testable languages, one can strike on the idea of describing multiple occurrences of one subword by a single occurrence of another subword. For instance, instead of testing a word for two occurrences of *ab* we can test whether *aab* or *abb* occurs at all.

So one can characterize the n-fold occurrence of a subword by a single occurrence of another subword. If a word u occurs at least n times as a subword of w, then there is an "intermediate" subword of w with length at most $|w| + n - 1$ with n occurrences of u. The opposite direction also holds. We state this formally in the following lemma.

Lemma 41. *For every $w \in \Sigma^*, u \in \Sigma^+$ and $n \geq 1$ it holds*

$$\mathit{Subw}(w, u) \geq n \quad \Longleftrightarrow \quad \exists v \in \mathit{Subwords}_{|u|+n-1}(w) \colon \mathit{Subw}(v, u) \geq n.$$

Proof. The direction from right to left is easy. If v is a subword of w and u occurs n times as a subword of v, then by the transitivity of the subword relation, u occurs n times as a subword of w.

Let us now show the direction from left to right. For that, we will construct a sequence of n distinct occurrences $u_1, \ldots, u_n$ of u in w with the following property. Every occurrence u_k will have at most one position that is not in the set of all positions gathered from the occurrences $u_1, \ldots, u_{k-1}$.

Such being the case, the number of all positions in $u_1, \ldots, u_n$ will be at most $|u| + \sum_{i=2}^{n} 1 = |u| + n - 1$. Then the set of all these positions defines a new subword v of length at most $|u| + n - 1$ where all the occurrences $u_1, \ldots, u_n$ of u in w also describe valid occurrences of u in v. This suffices to prove the right statement.

Let us now construct $u_1, \ldots, u_n$.

For u_1 we take an arbitrary occurrence of u in w.

For u_{k+1} with $1 \leq k \leq n - 1$: Let U_k be the set of all occurrences of u in w which are distinct from the occurrences $u_1, \ldots, u_k$. U_k is not empty, since $\mathrm{Subw}(w, u) \geq n$.

Let us number the positions of the occurrences already used:

$$\begin{aligned} u_1 &= (i_{11}, \ldots, i_{1m}) \\ u_2 &= (i_{21}, \ldots, i_{2m}) \\ &\vdots \\ u_k &= (i_{k1}, \ldots, i_{km}) \end{aligned}$$

For every $u' = (j_1, \ldots, j_m) \in U_k$, we denote the position j_μ to be *new*, if $j_\mu \neq i_{rs}$ for all $1 \leq r \leq k$ and $1 \leq s \leq m$. The number of new positions in an occurrence $u' \in U_k$ is denoted by $\#\mathrm{new}(u')$.

We fix a $u' \in U_k$ with minimal number of new positions $\#\mathrm{new}(u')$.

If $\#\mathrm{new}(u') = 0$ or $\#\mathrm{new}(u') = 1$, we simply set $u_{k+1} := u'$. Then u_{k+1} has at most one position more than $\{u_1, \ldots, u_k\}$ and we are done.

Hence we assume that $\#\mathrm{new}(u') \geq 2$. Let j_μ be the first new position in u' and consider

$$\begin{aligned} u_1 &= (i_1, \ldots, i_\mu, i_{\mu+1}, \ldots, i_m) \text{ and} \\ u' &= (j_1, \ldots, j_\mu, j_{\mu+1}, \ldots, j_m). \end{aligned}$$

First case: $j_\mu < i_{\mu+1}$.
We set $u_{k+1} := (j_1, \ldots, j_\mu, i_{\mu+1}, \ldots, i_m)$. Then u_{k+1} is an occurrence of u in w with exactly one new position, so we are done.

Second case: $j_\mu \geq i_{\mu+1}$.
Then $i_\mu < j_{\mu+1}$. We set $u'' := (i_1, \ldots, i_\mu, j_{\mu+1}, \ldots, j_m)$. Then u'' is an occurrence of u in w with $\#\mathrm{new}(u'') = \#\mathrm{new}(u') - 1$. Furthermore, u'' has at least one new position, so $u'' \in U_k$. But this contradicts our assumption that $\#\mathrm{new}(u')$ was minimal. $\square$

For any finite set of occurrences (even in infinite words), we can always give the very first position and the very last position, enclosing all other positions between them. For subword occurrences in infinite words, we can therefore apply the above lemma and obtain an equivalent for infinite words.

Corollary 42. *For every $\alpha \in \Sigma^\omega, u \in \Sigma^+$ and $n \geq 1$ it holds*

$$\mathit{Subw}(\alpha, u) \geq n \quad \Longleftrightarrow \quad \exists v \in \mathit{Subwords}_{|u|+n-1}(\alpha) \colon \mathit{Subw}(v, u) \geq n.$$

With this characterization of the n-fold occurrence of a subword, it is easy to prove our initial proposition.

Proposition 43. *For every $k, r \geq 1$ and every k-piecewise r-threshold testable language $L \subseteq \Sigma^\omega$, L is also $(k + r - 1)$-piecewise testable.*

Proof. We show: The $\sim^\omega_{k+r-1}$-relation refines the $\approx^{k\omega}_r$-relation (from Section 3.5).

Let $\alpha, \beta \in \Sigma^\omega$ with $\alpha \not\approx_r^{k\omega} \beta$.
By definition it follows that there is a $u \in \Sigma^+$ with $|u| \leq k$ and

$$\min\{\mathrm{Subw}(\alpha, u), r\} \neq \min\{\mathrm{Subw}(\beta, u), r\}.$$

Without loss of generality, we assume

$$\min\{\mathrm{Subw}(\alpha, u), r\} < \min\{\mathrm{Subw}(\beta, u), r\}$$

and set $r' := \min\{\mathrm{Subw}(\beta, u), r\}$.

- Since $\min\{\mathrm{Subw}(\alpha, u), r\} < r' \leq r$ it follows $\mathrm{Subw}(\alpha, u) < r'$.
  With Corollary 42 we conclude

  $$\neg\exists v_1 \in \mathrm{Subwords}_{|u|+r'-1}(\alpha)\colon \mathrm{Subw}(v_1, u) \geq r'.$$

  Clearly, $|u| + r' - 1 \leq k + r - 1$ and thus

  $$\neg\exists v_1 \in \mathrm{Subwords}_{k+r-1}(\alpha)\colon \mathrm{Subw}(v_1, u) \geq r'.$$

- Since $\min\{\mathrm{Subw}(\beta, u), r\} = r'$ it follows $\mathrm{Subw}(\beta, u) \geq r'$.
  With Corollary 42 we conclude

  $$\exists v_2 \in \mathrm{Subwords}_{|u|+r'-1}(\beta)\colon \mathrm{Subw}(v_2, u) \geq r'.$$

  Clearly, $|v_2| \leq |u| + r' - 1 \leq k + r - 1$ and thus

  $$\exists v_2 \in \mathrm{Subwords}_{k+r-1}(\beta)\colon \mathrm{Subw}(v_2, u) \geq r'.$$

So there is a $v_2 \in \mathrm{Subwords}_{k+r-1}(\beta)$ which is not in $\mathrm{Subwords}_{k+r-1}(\alpha)$. This means $\alpha \not\sim_{k+r-1}^\omega \beta$. □

Remark. Reducing the subword-length to $k + r - 2$ in Proposition 43 does not work. Already for a simple case like $k = 2, r = 1$, one can certainly not decide whether a subword of length two exists by only regarding the subwords of length one.

Now we know, that piecewise threshold testable languages are always piecewise testable. Therefore the class of piecewise threshold testable ω-languages is equal to the class of piecewise testable ω-languages. So the determinacy of piecewise threshold testable games follows from the results about piecewise testable games with no additional effort.

Theorem 44. *For every $k, r \geq 1$, k-piecewise r-threshold testable games are determined with $(k + r - 1)$-piecewise testable winning strategies.*

Proof. Let $L \subseteq (\Sigma_1 \times \Sigma_2)^\omega$ be k-piecewise r-threshold testable. Then by Proposition 43 it is also $(k + r - 1)$-piecewise testable and by Theorem 35 Church's Game for L is determined with $(k+r-1)$-piecewise testable winning strategies. □

However, we also want to examine whether such winning strategies are k-piecewise r-threshold testable with the same k and r as in the premise of this theorem. This is actually the case, as the following theorem shows.

Theorem 45. *For every $k, r \geq 1$, k-piecewise r-threshold testable games are determined with k-piecewise r-threshold testable winning strategies.*

Proof. Let $L \subseteq (\Sigma_1 \times \Sigma_2)^\omega$ be an instance of Church's Problem and let L be k-piecewise r-threshold testable.

Then there are words $\alpha_1, \ldots, \alpha_m \in \Sigma^\omega$ with $L = \bigcup_{i=1}^m [\alpha_i]_{\approx_r^{k\omega}}$.

We define a new weak parity game (G, c) together with its game arena G and a coloring function c.

Only subwords of length k are of interest, so let

$$M = \{v \in P_1 \mid 1 \leq |v| \leq k\}.$$

A multiset is now a function $f\colon M \to \{0, \ldots, r\}$, mapping each word v of length at most k to the number of occurrences of this word as a subword in the current partial play. With $\underline{r}^M$ we denote the set of all such multisets. For any multiset f define the cardinality of f to be $|f| = \sum_{v \in M} f(v)$ and an operator "+" with $f + \{v_1, \ldots, v_n\} := f'$ where

$$f'(v) := \begin{cases} \min\{f(v) + 1, r\}, & \text{if } \exists 1 \leq i \leq n\colon v_i = v; \\ f(v), & \text{otherwise.} \end{cases}$$

Let $Q = Q_1 \cup Q_2$ be the set of nodes of the game arena with

$$\begin{aligned} Q_1 =& \{f \mid f \in \underline{r}^M\} \\ Q_2 =& \{(f, a) \mid f \in \underline{r}^M, a \in \Sigma_1\} \end{aligned}$$

The transition function δ is defined by

$$\begin{aligned} \delta(f, a) &:= (f, a) \\ \delta((f, a), x) &:= f' \end{aligned}$$

where f' is the multiset obtained by

$$\begin{aligned} f' = f +& \{\tbinom{a}{x}\} \\ &+ \{v\tbinom{a}{x} \mid v \in M, v < k, f(v) \geq 1\} \\ &+ \{v\tbinom{a}{x} \mid v \in M, v < k, f(v) \geq 2\} \\ &\vdots \\ &+ \{v\tbinom{a}{x} \mid v \in M, v < k, f(v) \geq k - 1\} \end{aligned}$$

In order to translate plays $\varrho \in (\Sigma_1 \times \Sigma_2)^\omega$ into plays $\varrho' \in Q^\omega$, we define a function $h\colon P \to Q$ by letting

$$\begin{aligned} h(w) &:= f \\ h(w\tbinom{a}{*}) &:= (f, a) \end{aligned}$$

with $f(v) := \min\{\mathrm{Subw}(w, v), r\}$ for every $v \in M$.

The function h is a homomorphism in the sense that it respects the transition function: For every $w \in P_1$ it holds

$$\begin{aligned} h(\gamma(w, a)) &= h(w\tbinom{a}{*}) \\ &= (f, a) \\ &= \delta(f, a) \\ &= \delta(h(w), a) \end{aligned}$$

with $f(v) = \min\{\mathrm{Subw}(w, v), r\}$. For every $w\binom{a}{*} \in P_2$ it holds

$$\begin{aligned} h(\gamma(w\tbinom{a}{*}, x)) &= h(w\tbinom{a}{x}) \\ &= f' \\ &= \delta((f, a), x) \\ &= \delta(h(w\tbinom{a}{*}), x) \end{aligned}$$

with $f(v) = \min\{\mathrm{Subw}(w, v), r\}$ and $f'(v) = \min\{\mathrm{Subw}(w\binom{a}{x}, v), r\}$.

Every play starts in $q_0 := h(\varepsilon)$.

The acceptance component depends on the words $\alpha_1, \dots, \alpha_m$, which make up the language L. It consists of the coloring function $c\colon Q \to \big\{0, \dots, 2r \cdot (|(\Sigma_1 \times \Sigma_2)|^{k+1} - 1)\big\}$. We set

$$c(f) := c((f, a)) := \begin{cases} 2 \cdot |f|, & \text{if } \exists i \in \big\{1, \dots, m\big\}\colon \\ & \forall u \in M\colon f(u) = \min\{\mathrm{Subw}((, \alpha)_i, u), r\}; \\ 2 \cdot |f| - 1, & \text{otherwise.} \end{cases}$$

The game (G, c) is a weak parity game. We know from Theorem 8 that it is determined and that the winning player has a memoryless winning strategy.

Lemma 46. *If Player 1 has a winning strategy in (G, c) from q_0, then Player 1 has a k-piecewise r-threshold testable winning strategy in Ch(L).*

Lemma 47. *If Player 2 has a winning strategy in (G, c) from q_0, then Player 2 has a k-piecewise r-threshold testable winning strategy in Ch(L).*

The game (G, c) is a weak parity game. Weak parity games are determined, so one of the players has a winning strategy from q_0, which moreover is memoryless. In the case that Player 1 has a winning strategy use Lemma 46, in the other case use Lemma 47 to prove the result.

□

Proof of Lemma 46. Let $f_1\colon Q \to \Sigma$ be a memoryless winning strategy for Player 1 from q_0 in (G, c). For every $a \in \Sigma$ define

$$Q_1^a := \{q \in Q_1 \mid f_1(q) = a\} \text{ and}$$
$$S_a := \{w \in P_1 \mid h(w) \in Q_1^a\}.$$

Then clearly the sets Q_1^a are pairwise disjoint and cover Q_1, while the sets S_a are pairwise disjoint and cover P_1. So $(S_a)_{a\in\Sigma_1}$ is a strategy for Player 1.

We still have to show that $(S_a)_{a\in\Sigma_1}$ is winning for Player 1. Let ϱ be a play played according to $(S_a)_{a\in\Sigma_1}$. We want Player 1 to win this play, so we are going to show that $\alpha := \alpha(\varrho)$ is not in L. Let $\varrho = p_0, p_1, \ldots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \ldots$ in (G, c). Analogously to the proofs above, we obtain that Player 1 wins ϱ'.

Then the maximal color $C(\varrho')$ occurring in ϱ' is odd. At some point of ϱ', this color d has to occur. Let j be the smallest index with $c(q_j) = d$. Then for all following nodes, the color stays at d, because the multiset f cannot shrink during a play. But then the multiset f stays constant from j onwards. So f not only describes the multiplicity of subwords of these partial plays but also the multiplicity of subwords of α.

Since d is odd, there is no α_i with $\forall u \in M\colon f(u) = \min\{\mathrm{Subw}(\alpha_i, u), r\}$. So $\alpha \notin L$.

Let us show that each S_a is k-piecewise r-threshold testable. For $w_1, w_2 \in P_1$, we show: if $w_1 \approx_r^k w_2$ then $w_1 \in S_a \Leftrightarrow w_2 \in S_a$. Let $w_1 \approx_r^k w_2$. Then

$$\min\{\mathrm{Subw}(w_1, v), r\} = \min\{\mathrm{Subw}(w_2, v), r\}$$

and so $h(w_1) = f$ with $f(v) = \min\{\mathrm{Subw}(w_1, v), r\} = \min\{\mathrm{Subw}(w_2, v), r\}$ for every $v \in M$ and so $h(w_2) = h(w_1)$.

This means $w_1 \in S_a \Leftrightarrow w_2 \in S_a$ and S_a is as a union of $\approx_r^k$ classes k-piecewise r-threshold testable. So $(S_a)_{a\in\Sigma_1}$ is a k-piecewise r-threshold testable winning strategy. □

Proof of Lemma 47. Let $f_2\colon Q \to \Sigma$ be a memoryless winning strategy for Player 2 from q_0 in (G, c). For every $a \in \Sigma$ define

$$Q_2^x := \{q \in Q_2 \mid f_2(q) = x\} \text{ and}$$
$$T_x := \{w \in P_2 \mid h(w) \in Q_2^x\}.$$

Then clearly the sets Q_2^x are pairwise disjoint and cover Q_2, while the sets T_a are pairwise disjoint and cover P_2. So $(T_x)_{x\in\Sigma_2}$ is a strategy for Player 2.

We still have to show that $(T_x)_{x\in\Sigma_2}$ is winning for Player 2. Let ϱ be a play played according to $(T_x)_{x\in\Sigma_2}$. We want Player 2 to win this play, so we are going to show that $\alpha := \alpha(\varrho)$ is in L. Let $\varrho = p_0, p_1, \ldots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \ldots$ in (G, c). Analogously to the proofs above, we obtain that Player 2 wins ϱ'.

Then the maximal color $C(\varrho')$ occurring in ϱ' is even. At some point of ϱ', this color d has to occur. Let j be the smallest index with $c(q_j) = d$. Then for all following nodes, the color stays at d, because the multiset f cannot shrink during a play. But then the multiset f stays constant from j onwards. So f not only describes the multiplicity of subwords of these partial plays but also the multiplicity of subwords of α.

Since d is even, there is an α_i with $\forall u \in M\colon f(u) = \min\{\mathrm{Subw}(\alpha_i, u), r\}$. So $\alpha \in L$.

Let us show that each T_x is k-piecewise r-threshold testable. For $w_1, w_2 \in P_2$, we show: if $w_1 \approx_r^k w_2$ then $w_1 \in S_a \Leftrightarrow w_2 \in S_a$. Let $w_1 \approx_r^k w_2$. Then

$$\min\{\mathrm{Subw}(w_1, v), r\} = \min\{\mathrm{Subw}(w_2, v), r\}.$$

Let $w_1 = u_1\binom{a}{*}$ and $w_2 = u_2\binom{b}{*}$. Then $h(w_1) = (f, a)$ with $f(v) = \min\{\mathrm{Subw}(u_1, v), r\}$ for every $v \in M$ and $h(w_2) = (g, b)$ with $g(v) = \min\{\mathrm{Subw}(u_2, v), r\}$ for every $v \in M$. But since $\binom{a}{*}$ is the only subword of w_1 of length 1 which has a $*$ in its second component and $\binom{b}{*}$ is the only subword of w_2 of length 1 which has a $*$ in its second component, it holds $a = b$. Furthermore the letter $*$ does not appear in any of the words $v \in M$, so

$$\begin{aligned}\min\big\{\mathrm{Subw}(u_1, v), r\big\} &= \min\big\{\mathrm{Subw}(w_1, v), r\big\} \\ &= \min\big\{\mathrm{Subw}(w_2, v), r\big\} = \min\big\{\mathrm{Subw}(u_2, v), r\big\}\end{aligned}$$

for every $v \in M$ and this means, $f = g$. So $h(w_2) = h(w_1)$.

This means $w_1 \in T_x \Leftrightarrow w_2 \in T_x$ and T_x is as a union of $\approx_r^k$ classes k-piecewise r-threshold testable. So $(T_x)_{x \in \Sigma_2}$ is a k-piecewise r-threshold testable winning strategy. □

3.7 Modulo Counting Languages

The content of this section is motivated by the idea to consider logics with modulo counting quantifiers. Rabinovich and Thomas [RT07] show two result for instances of Church's Problem which are defined by logics with modulo counting quantifiers. The logics they consider are FO(<)+MOD and FO(S)+MOD.

These are first order logics together with a new quantifier $\exists^{(q,r)}$. The common semantics of this quantifier is this: $\exists^{(q,r)} x \varphi(x)$ is fulfilled, iff there are exactly n positions x in the word model that satisfy $\varphi(x)$ and for the number of positions holds $n \equiv r \pmod q$. The classes of $*$-languages which can be defined by these logics are extensively treated in the book by Straubing [Str94].

There is also a not so common semantics for the $\exists^{(q,r)}$ quantifier: The formula $\exists^{(q,r)}x\varphi(x)$ is fulfilled, iff there is a position x in the word model that satisfies $\varphi(x)$ and for this position holds $x \equiv r \pmod q$.

Modulo Counting the Number of Positions

We have already mentioned in Section 3.4 that the class of FO(S)-definable languages is equal to the class of locally threshold testable languages. Taking a look at the logic FO(S)+MOD, we see that the additional quantifier properly expands the class of definable languages. For example the language $(\Sigma\Sigma)^*$ of all even words can be expressed by counting modulo 2.

Example 48. Over the alphabet $\Sigma = \{a, b\}$ the sentence

$$\varphi = \exists^{(2,0)}x\exists^{(2,1)}yS(x,y)$$

describes the language $(\Sigma\Sigma)^*$ of all finite words of even length. Here, $S(x,y)$ means "y is the direct successor of x". We do not need the usual existential quantifier $\exists$ in this example.

However, Church's Problem for this language class has already been treated. The result obtained in [RT07] is the following.

Theorem 49. *There are FO(S)+MOD-definable games that do not have FO(S)+MOD-definable winning strategies.*

Modulo Counting the Position

Let us consider the second interpretation of the $\exists^{(q,r)}$ quantifier. The formula $\exists^{(q,r)}x\varphi(x)$ is fulfilled, iff there is a position x in the word model that satisfies $\varphi(x)$ and $x \equiv r \pmod q$.

Example 50. Over the alphabet $\Sigma = \{a, b\}$ the sentence

$$\varphi = \exists^{(2,1)}x\neg\exists yS(x,y)$$

describes the language $\Sigma\Sigma(\Sigma\Sigma)^*$ of all finite words of even length ≥ 2. Again, $S(x,y)$ means "y is the direct successor of x".

With this new semantics one can for example express that a letter b occurs at a position that is divisible by 3. Taking the logical successor relation into account, we can also express that a whole factor occurs at a position that is divisible by 3. We can even count the occurring factors up to a threshold, but we cannot say if a factor occurs infinitely often in an infinite word. This indicates that the class of ω-languages defined by this logic describes weak ω-languages.

How do we translate this semantics into the language theoretical framework, that we used in all the other cases of this chapter? This requires a bit of preliminary work. We first define a new set $\mathrm{Factors}_k^{(q,r)}(w)$ which contains all factors that begin at a position $x \equiv r \pmod q$.

Definition 51.

$$\begin{aligned}\mathrm{Factors}_k^{(q,r)}(w) := \bigl\{u \in \Sigma^* \bigm| & |u| \le k, \exists v, x \in \Sigma^*\colon \\ & |v| \equiv r \pmod q, \\ & w = vux\bigr\}\end{aligned}$$

Now we can again define a class of languages by means of an equivalence relation.

Definition 52.

$$\begin{aligned} w_1 \approx_q^k w_2 \quad :\Longleftrightarrow \quad & \forall 1 \le r < q\colon \mathrm{Factors}_k^{(q,r)}(w_1) = \mathrm{Factors}_k^{(q,r)}(w_2) \\ & \wedge |w_1| \equiv |w_2| \pmod q \\ & \wedge \mathrm{Prefix}_{k-1}(w_1) = \mathrm{Prefix}_{k-1}(w_2) \\ & \wedge \mathrm{Suffix}_{k-1}(w_1) = \mathrm{Suffix}_{k-1}(w_2). \end{aligned}$$

A language $K \subseteq \Sigma^*$ is called *k-locally positional q-modulo testable*, if it can be written as a finite union of classes from $\Sigma^*/{\approx_q^k}$:

$$K = \bigcup_{i=1}^{m} [w_i]_{\approx_q^k}$$

A language $K \subseteq \Sigma^*$ is called *locally positional modulo testable*, if it is k-locally positional q-modulo testable for certain $k, q \ge 1$.

Notice in the definition of the relation $\approx_q^k$ we require both words to have the same length modulo q and the same suffix of length $k-1$. We need this in the game $\mathrm{Ch}(L)$ for updating the set $\mathrm{Factors}_k^{(q,r)}(w_1)$. For the corresponding ω-language this information is, of course, unavailable.

Definition 53.

$$\begin{aligned} \alpha_1 \approx_q^{k\omega} \alpha_2 \quad :\Longleftrightarrow \quad & \forall 1 \le r < q\colon \mathrm{Factors}_k^{(q,r)}(\alpha_1) = \mathrm{Factors}_k^{(q,r)}(\alpha_2) \\ & \wedge \mathrm{Prefix}_{k-1}(w_1) = \mathrm{Prefix}_{k-1}(w_2). \end{aligned}$$

A language $L \subseteq \Sigma^\omega$ is called *k-locally positional q-modulo testable*, if it can be written as a finite union of classes from $\Sigma^\omega/{\approx_q^{k\omega}}$:

$$L = \bigcup_{i=1}^{m} [w_i]_{\approx_q^{k\omega}}$$

A language $L \subseteq \Sigma^\omega$ is called *locally positional modulo testable*, if it is k-locally positional q-modulo testable for certain $k, q \ge 1$.

With the $\exists^{(q,r)}$ quantifier we can also count up to some threshold. The locally positional modulo testable languages, that we defined above, do not allow this.

Example 54. Over the alphabet $\Sigma = \{a, b\}$ the sentence

$$\varphi = \exists^{(3,0)} x \exists^{(3,0)} y (x \neq y \wedge P_a(x) \wedge P_a(y))$$

describes the language of all words that contain at least 2 occurrences of the letter a at positions that are divisible by 3. For instance, this formula is satisfied by the infinite word

$$\alpha = bbbabbab^\omega.$$

But the language defined by φ is not locally positional modulo testable.

Church's Problem

Theorem 55. *For every $k, q \in \mathbb{N}$, k-locally positional q-modulo testable games are determined with k-locally positional q-modulo testable winning strategies.*

Proof. Let $L \subseteq (\Sigma_1 \times \Sigma_2)^\omega$ be an instance of Church's Problem and let L be k-locally positional q-modulo testable.

Then there are words $\alpha_1, \ldots, \alpha_m \in \Sigma^\omega$ with $L = \bigcup_{i=1}^m [\alpha_i]_{\approx_q^{k\omega}}$.

We define a new weak parity game (G, c) together with its game arena G and a coloring function c. For each pair (q, r) with $0 \leq r < q$ we have to accumulate the set $\mathrm{Factors}_k^{(q,r)}(\cdot)$ of already seen factors. This is accomplished by a mapping

$$\begin{aligned} g\colon \{0, \ldots, q-1\} &\to \mathcal{P}(M) \text{ with} \\ M &= \{v \in P_1 \mid 1 \leq |v| \leq k\}. \end{aligned}$$

With $\underline{r}$ denote the set of all integers $\{0, \ldots, r\}$ up to r. Then the set $\mathcal{P}(M)^{\underline{r}}$ of all such mappings g is finite. For any mapping g define the cardinality of g to be $|f| = \sum_{s \in \underline{q-1}} g(s)$.

Let $Q = Q_1 \cup \overline{Q_2}$ be the set of nodes of the game arena with

$$\begin{aligned} Q_1 :=& \{(u_{\mathrm{pre}}, g, u_{\mathrm{suf}}, r) \in P_1 \times \mathcal{P}(M)^{\underline{r}} \times P_1 \times \underline{q-1} \mid \\ & \qquad |u_{\mathrm{pre}}| = |u_{\mathrm{suf}}| = k-1\} \\ & \cup \{w \in P_1 \mid |w| \leq k-1\}, \\ Q_2 :=& \{(u_{\mathrm{pre}}, g, u_{\mathrm{suf}}, r) \in P_1 \times \mathcal{P}(M)^{\underline{r}} \times P_2 \times \underline{q-1} \mid \\ & \qquad |u_{\mathrm{pre}}| = k-1, |u_{\mathrm{suf}}| = k\} \\ & \cup \{w \in P_2 \mid |w| \leq k\}. \end{aligned}$$

For nodes from Q_1, the transition function δ is defined by

$$\delta(w,a) := w\binom{a}{*} \text{ and}$$
$$\delta((u_{\text{pre}}, g, u_{\text{suf}}, r), a) := (u_{\text{pre}}, g, u_{\text{suf}}\binom{a}{*}, r).$$

For nodes from Q_2, it is

$$\delta(w\binom{a}{*}, x) := w\binom{a}{x} \text{ , if } |w\binom{a}{*}| \leq k-1$$
$$\delta(\binom{a}{x}w\binom{b}{*}, y) := (\binom{a}{x}w, g, w\binom{b}{y}, 0) \text{ , if } |\binom{a}{x}w\binom{b}{*}| = k \geq 2$$

with $g(s) = \text{Prefix}_k(\binom{a}{x}w\binom{b}{y})$,

$$\delta(\binom{a}{*}, x) := (\binom{a}{x}, g, \binom{a}{x}, 0) \text{ , if } k = 1$$

with $g(s) = \begin{cases} \{\binom{a}{x}\}, & \text{if } s = 0; \\ \emptyset, & \text{otherwise;} \end{cases}$

$$\delta((u_{\text{pre}}, g, \binom{a}{x}u_{\text{suf}}\binom{b}{*}), r) := (u_{\text{pre}}, g', u_{\text{suf}}\binom{b}{y}, r')$$

with $r' = r + 1 \bmod q$ and
$g'(s) = \begin{cases} g(s) \cup \text{Prefix}_k(\binom{a}{x}u_{\text{suf}}\binom{b}{y}), & \text{if } s = r'; \\ g(s), & \text{otherwise.} \end{cases}$

In order to translate plays $\varrho \in (\Sigma_1 \times \Sigma_2)^\omega$ into plays $\varrho' \in Q^\omega$, we define a function $h\colon P \to Q$ by letting

$$h(w) := w \text{ if } |w| \leq k-1 \text{ or } |w| = k, w \in P_2$$
$$h(w) := (u_{\text{pre}}^{k-1}, g, u_{\text{suf}}^{k-1}, r) \text{ if } w \in P_1, |w| \geq k$$
$$h(w\binom{a}{*}) := (u_{\text{pre}}^{k-1}, g, u_{\text{suf}}^{k-1}\binom{a}{*}, r) \text{ else.}$$

with u_{pre}^{k-1} being the $k-1$-prefix of w, u_{suf}^{k-1} being the $k-1$-suffix of w, $r = (|w| - k) \bmod q$ and $g(s)$ contains the k-factors of all positions x with $x \equiv s \pmod q$ of the word w. Then h is a homomorphism.

The acceptance component depends on the words $\alpha_1, \ldots, \alpha_m$, which make up the language L. It consists of the coloring function $c\colon Q \to \{0, \ldots, q \cdot (|(\Sigma_1 \times \Sigma_2)|^{k+1} - 1)\}$. We map every node that is not of the form $(u_{\text{pre}}, g, u_{\text{suf}}, r)$ to the color 0. For the other nodes we set

$$c((u_{\text{pre}}, g, u_{\text{suf}}, r)) := \begin{cases} 2 \cdot |g|, & \text{if } \exists i \in \{1, \ldots, m\}\colon u_{\text{pre}} \text{ is prefix of } \alpha_i, \\ & \forall s \in \underline{q-1}\colon g(s) = \text{Factors}_k^{(q,s)}(\alpha_i); \\ 2 \cdot |g| - 1, & \text{otherwise.} \end{cases}$$

We set $q_0 = h(\varepsilon)$. To show that this construction indeed gives us the desired result, we state the following lemmas.

Lemma 56. *If Player 1 has a winning strategy in (G, c) from q_0, then Player 1 has a k-locally positional q-modulo testable winning strategy in $Ch(L)$.*

Lemma 57. *If Player 2 has a winning strategy in (G, c) from q_0, then Player 2 has a k-locally positional q-modulo testable winning strategy in $Ch(L)$.*

The game (G, c) is a weak parity game. Weak parity games are determined, so one of the players has a winning strategy from q_0, which moreover is memoryless. In the case that Player 1 has a winning strategy use Lemma 56, in the other case use Lemma 57 to prove the result. □

Proof of Lemma 56. Let $f_1 : Q_1 \to \Sigma_1$ be a memoryless winning strategy for Player 1 from q_0 in (G, c). For every $a \in \Sigma_1$ define

$$\begin{aligned} Q_1^a &:= \{q \in Q_1 \mid f_1(q) = a\} \text{ and} \\ S_a &:= \{w \in P_1 \mid h(w) \in Q_1^a\}. \end{aligned}$$

Then clearly the sets Q_1^a are pairwise disjoint and cover Q_1, while the sets S_a are pairwise disjoint and cover P_1. So $(S_a)_{a\in\Sigma_1}$ is a strategy for Player 1.

We still have to show that $(S_a)_{a\in\Sigma_1}$ is winning for Player 1. Let ϱ be a play played according to $(S_a)_{a\in\Sigma_1}$. We want Player 1 to win this play, so we are going to show that $\alpha := \alpha(\varrho)$ is not in L. Let $\varrho = p_0, p_1, \ldots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \ldots$ in (G, c). For every even $i \in \mathbb{N}$ there is an $a \in \Sigma_1$ and a $w \in P_1$ such that $p_i = w, p_{i+1} = w\binom{a}{*}$. Since ϱ is played according to $(S_a)_{a\in\Sigma_1}$, clearly $w \in S_a$ and by the definition of S_a and Q_1^a we obtain $h(w) \in Q_1^a$ and $f_1(h(w)) = a$. So ϱ' is played according to f_1. But f_1 was a winning strategy for Player 1 from q_0, so Player 1 wins ϱ'.

Then the maximal color $C(\varrho')$ occurring in ϱ' is odd. At some point of ϱ', this color d has to occur. Let j be the smallest index with $c(q_j) = d$. Then for all following nodes, the color stays at d, because the cardinality of the mapping g cannot decrease during a play. But then the mapping g stays constant from j onwards and so does u_{pre}. Then u_{pre} and g also describe the prefix of α and the sets $\text{Factors}_k^{(q,s)}(\alpha)$ for every $s \in \underline{q-1}$.

Since d is odd, there is no α_i such that u_{pre} is prefix of α_i and $\forall r \in \underline{q-1} \colon f(r) = \text{Factors}_k^{(q,r)}(\alpha_i)$. So $\alpha \notin L$.

Let us show that each S_a is k-locally positional q-modulo testable. For $w_1, w_2 \in P_1$, we show: if $w_1 \approx_q^k w_2$ then $w_1 \in S_a \Leftrightarrow w_2 \in S_a$. Let $w_1 \approx_q^k w_2$.

Then

$$\begin{aligned}
&\forall 1 \leq r < q\colon \mathrm{Factors}_k^{(q,r)}(w_1) = \mathrm{Factors}_k^{(q,r)}(w_2)\\
&\wedge |w_1| \equiv |w_2| \pmod{q}\\
&\wedge \mathrm{Prefix}_{k-1}(w_1) = \mathrm{Prefix}_{k-1}(w_2)\\
&\wedge \mathrm{Suffix}_{k-1}(w_1) = \mathrm{Suffix}_{k-1}(w_2)
\end{aligned}$$

and so $h(w_1) = (u_{\mathrm{pre}}, g, u_{\mathrm{suf}}, r) = h(w_2)$ where u_{pre} is the longest word in $\mathrm{Prefix}_{k-1}(w_1)$, u_{suf} is the longest word in $\mathrm{Suffix}_{k-1}(w_1)$, g is the function with $g(s) = \mathrm{Factors}_k^{(q,s)}(w_1)$ for every $s \in \underline{q-1}$ and $r = |w_1| - k \bmod q$.

This means $w_1 \in S_a \Leftrightarrow w_2 \in S_a$ and S_a is as a union of $\approx_q^k$ classes k-locally positional q-modulo testable. So $(S_a)_{a\in\Sigma}$ is a k-locally positional q-modulo testable winning strategy. □

Proof of Lemma 57. The proof essentially follows the one of Lemma 56.

Let $f_2\colon Q_2 \to \Sigma$ be a memoryless winning strategy for Player 2 from q_0 in (G, c). For every $x \in \Sigma_2$ define

$$\begin{aligned}
Q_2^x =& \{q \in Q_2 \mid f_2(q) = x\} \text{ and}\\
T_x =& \{w \in P_2 \mid h(w) \in Q_2^x\}.
\end{aligned}$$

Then clearly the sets Q_2^x are pairwise disjoint and cover Q_2, while the sets T_x are pairwise disjoint and cover P_2. So $(T_x)_{x\in\Sigma_2}$ is a strategy for Player 2.

We still have to show that $(T_x)_{x\in\Sigma_2}$ is winning for Player 2. Let ϱ be a play played according to $(T_x)_{x\in\Sigma_2}$. We want Player 2 to win this play, so we are going to show that $\alpha := \alpha(\varrho)$ is in L. Let $\varrho = p_0, p_1, \ldots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \ldots$ in (G, c). For every odd $i \in \mathbb{N}$ there is an $a \in \Sigma_1$, an $x \in \Sigma_2$ and a $w \in P_1$ such that $\varrho_i = w\binom{a}{*}, \varrho_{i+1} = w\binom{a}{x}$. Since ϱ is played according to $(T_x)_{x\in\Sigma_2}$, clearly $w \in T_x$ and by the definition of T_x and Q_2^x we obtain $h(w) \in Q_2^x$ and $f_2(h(w)) = x$. So ϱ' is played according to f_1. But f_2 is a winning strategy for Player 2 from q_0, so Player 2 wins ϱ'.

Then the maximal color $C(\varrho')$ occurring in ϱ' is even. At some point of ϱ', this color d has to occur. Let j be the smallest index with $c(q_j) = d$. Then for all following nodes, the color stays at d, because the cardinality of the mapping g cannot decrease during a play. But then the mapping g stays constant from j onwards and so does u_{pre}. Then u_{pre} and g also describe the prefix of α and the sets $\mathrm{Factors}_k^{(q,s)}(\alpha)$ for every $s \in \underline{q-1}$.

Since d is even, there is an α_i such that u_{pre} is prefix of α_i and $\forall r \in \underline{q-1}\colon f(r) = \mathrm{Factors}_k^{(q,r)}(\alpha_i)$. So $\alpha \in L$.

For $w_1, w_2 \in P_2$, we show: if $w_1 \approx_q^k w_2$ then $w_1 \in T_x \Leftrightarrow w_2 \in T_x$. Let

$w_1 \approx_q^k w_2$. Then

$$\begin{aligned}
&\forall 1 \leq r < q \colon \mathrm{Factors}_k^{(q,r)}(w_1) = \mathrm{Factors}_k^{(q,r)}(w_2) \\
\wedge\ & |w_1| \equiv |w_2| \pmod q \\
\wedge\ & \mathrm{Prefix}_{k-1}(w_1) = \mathrm{Prefix}_{k-1}(w_2) \\
\wedge\ & \mathrm{Suffix}_{k-1}(w_1) = \mathrm{Suffix}_{k-1}(w_2)
\end{aligned}$$

and so $h(w_1) = (u_{\mathrm{pre}}, g, u_{\mathrm{suf}}, r) = h(w_2)$ where u_{pre} is the longest word in $\mathrm{Prefix}_{k-1}(w_1)$, u_{suf} is the longest factor of w_1 that ends with a letter $\binom{a}{*}$, g is the function with $g(s) = \mathrm{Factors}_k^{(q,s)}(w)$ for $w\binom{a}{*} = w_1$ and every $s \in \underline{q-1}$ and $r = |w_1| - k \bmod q$.

This means $w_1 \in T_x \Leftrightarrow w_2 \in T_x$ and T_x is as a union of $\approx_q^k$ classes k-locally positional q-modulo testable. So $(T_x)_{x \in \Sigma_2}$ is a k-locally positional q-modulo testable winning strategy. $\square$

Chapter 4

General Result

In the last chapter, we solved Church's Problem for some well-known language classes. In this chapter, we want to generalize these results. It would be interesting to know a method that takes a class of $*$-languages $\mathcal{K}$ and gives us a class of ω-languages $\mathcal{L}_{\mathcal{K}}$, such that

$\mathcal{L}_{\mathcal{K}}$-definable games are determined with $\mathcal{K}$-definable winning strategies.

We concentrate on subclasses of the regular languages and for these subclasses, we present such a method. Based on a class of regular $*$-languages $\mathcal{K}$ with certain closure properties, we will define a new class of weak regular ω-languages $\mathcal{K}^w$ which consists of weak infinite counterparts of the languages in $\mathcal{K}$ and their Boolean combinations.

4.1 Main Theorem

The proof idea for the results of Chapter 3 involves simulating a game by a weak parity game. We have depicted this proof idea already in Section 3.1. The games from Chapter 3 for which our idea works are all weak games. But we have seen in the example of strongly locally testable games that it does not perform with strong games. Therefore we concentrate only on weak games.

Weak games are described by languages that are recognized by Staiger-Wagner automata (cf. Section 2.5). These languages can also be constructed by Boolean combinations of so-called A-recognizable languages. Those A-recognizable languages are ω-languages which can be recognized by a finite automaton in the following way. The automaton accepts all runs that always stay in a set of accepting states. Equivalently we can say that every prefix is a member of a certain regular language. This motivates the following definition.

Let again $\sqsubseteq$ be the extended prefix relation from Definition 9. For example it holds $\binom{a}{*} \sqsubseteq \binom{a}{x}\binom{a}{y}^{\omega}$.

Definition 58. For every $*$-language V let

$$V^{\Box} = \{\alpha \in \Sigma^\omega \mid \forall w \sqsubseteq \alpha \colon w \in V\}$$

and for every class of $*$-languages $\mathcal{K}$, let $\mathcal{K}^w$ be the class of *weak ω-languages over* $\mathcal{K}$ defined by

$$L \in \mathcal{K}^w \iff L \text{ is a Boolean combination of languages } V_i^{\Box} \text{ with } V_i \in \mathcal{K}.$$

Under which assumptions can we proceed from $\mathcal{K}$ to $\mathcal{K}^w$, such that $\mathcal{K}^w$-definable games are determined with $\mathcal{K}$-definable winning strategies? We propose some suitable assumptions that will suffice to make the statement true.

The Myhill-Nerode equivalence relation is commonly used in conjunction with regular $*$-languages, for example for minimizing a finite automaton. We are going to define an analogue for ω-languages. Although with this analogue it is not possible to define an arbitrary regular ω-language, it still suffices to define a weak regular ω-language.

Recall from Section 2.3 that $P_1 := (\Sigma_1 \times \Sigma_2)^*$ is the set of finite words from which Player 1 has his next turn and $P_2 := (\Sigma_1 \times \Sigma_2)^*(\Sigma_1 \times \{*\})$ is the set of finite words where it is Player 2's turn.

Definition 59. For every ω-language $L \in \Sigma^\omega$ let $\sim_L \subseteq (P_1 \times P_1) \cup (P_2 \times P_2)$ be the "Myhill-Nerode" equivalence defined for every $u, v \in P_1$ by

$$u \sim_L v \ :\iff\ \forall \beta \in \Sigma^\omega \colon (u\beta \in L \iff v\beta \in L)$$

and

$$u\tbinom{a}{*} \sim_L v\tbinom{b}{*} \ :\iff \forall x \in \Sigma_2, \beta \in \Sigma^\omega \colon \\ (u\tbinom{a}{x}\beta \in L \iff v\tbinom{b}{x}\beta \in L).$$

Definition 60. Let $\mathcal{K}$ be a class of $*$-languages. For every $L \in \mathcal{K}^w$ assume that every class $[u]_{\sim_L}$ belongs to $\mathcal{K}$. Furthermore, let $\mathcal{K}$ be closed under finite intersections and finite unions. Then we call $\mathcal{K}$ *adequate*.

If we proceed from a class $\mathcal{K}$ of adequate regular $*$-languages to a class $\mathcal{K}^w$, then $\mathcal{K}^w$-definable games are determined with $\mathcal{K}$-definable winning strategies. We will show this below by the familiar game simulation method.

Therefore it is necessary to know that the Myhill-Nerode classes do suffice for building up a game graph with which we can simulate a game $\mathrm{Ch}(L)$ for any $L \in \mathcal{L}$. This is clarified by the following theorem.

Definition 61. For a game arena $G = (Q_1, Q_2, \delta)$ define the transitive closure δ^* inductively for $q \in Q_1 \cup Q_2$ and $w \in P_1 \cup P_2$ by

$$\begin{aligned}
\delta^*(q, \varepsilon) &= q, \\
\delta^*(q, w\tbinom{a}{*}) &= \delta(\delta^*(q, w), a), \\
\delta^*(q, w\tbinom{a}{x}) &= \delta(\delta^*(q, w\tbinom{a}{*}), x).
\end{aligned}$$

Theorem 62. *Let $\mathcal{K}$ be a class of adequate regular $*$-languages. Then for every $L \in \mathcal{K}^w$ we can effectively construct a weak parity game (G, c) with game arena $G = (Q_1, Q_2, \delta)$ and a designated initial node $q_0 \in Q_1$ such that*

1. *Q_1 and Q_2 are finite subsets of $\mathcal{K}$,*
2. *for each $K_1, K_2 \in Q$ with $K_1 \neq K_2$ holds $K_1 \cap K_2 = \emptyset$,*
3. *for each $w \in P_1 \cup P_2$ holds $w \in \delta^*(q_0, w)$ and*
4. *for each play ϱ holds: (Player 2 wins ϱ in $(G, c) \iff \alpha(\varrho) \in L$).*

Proof. We prove an even stronger statement by induction, namely that we can construct such a game graph with a monotonic coloring function. We say a coloring function is *monotonic*, if in each possible play, the colors do not decrease, so for each play $\varrho \in Q^\omega$ and all $i \in \mathbb{N}$ it holds $c(\varrho(i)) \leq c(\varrho(i+1))$.

Then the claim is that for every $L \in \mathcal{K}^w$ there is a weak parity game (G, c) with game arena $G = (Q_1, Q_2, \delta)$ and a designated initial node $q_0 \in Q_1$ such that 1. – 4. are fulfilled and additionally

5. c is a monotonic coloring function.

The proof of this stronger statement is by induction on the structure of the Boolean expression.

Induction basis: $L = V^\Box$ with $V \in \mathcal{K}$
We define

- $Q_1 = P_1/{\sim_L}$, $Q_2 = P_2/{\sim_L}$,
- $\delta([w]_{\sim_L}, a) = [w\binom{a}{*}]_{\sim_L}$, $\delta([w\binom{a}{*}]_{\sim_L}, x) = [w\binom{a}{x}]_{\sim_L}$,
- $q_0 = [\varepsilon]_{\sim_L}$ and
- $c\colon Q \to \{0, 1\}$ by

$$c([w]) := \begin{cases} 0, & \text{if there is a } \beta \in \Sigma^\omega \text{ s.t. } w\beta \in V^\Box; \\ 1, & \text{otherwise;} \end{cases}$$

$$c([w\tbinom{a}{*}]) := \begin{cases} 0, & \text{if there are } x \in \Sigma_2, \beta \in \Sigma^\omega \text{ s.t. } w\binom{a}{x}\beta \in V^\Box; \\ 1, & \text{otherwise.} \end{cases}$$

This is a well-defined function: Let $[w_1]_{\sim_L} = [w_2]_{\sim_L}$ and let there be a $\beta \in \Sigma^\omega$ such that $w_1\beta \in V^\Box$. Since $w_1 \sim_L w_2$, then $w_2\beta \in V^\Box$, too. The same holds for equivalence classes $[w\binom{a}{*}]$.

Furthermore, c is monotonic. If there is an $i \in \mathbb{N}$ with $c(\varrho(i)) = 1$, then for any $w \in \varrho(i)$ (respectively $w\binom{a}{*} \in \varrho(i)$) there is no continuation β (respectively x and β) with $w\beta \in V^\Box$ (respectively $w\binom{a}{x}\beta \in V^\Box$), so all successors must also have the color $c(\varrho(i+1)) = 1$.

For any $w \in P_1$ then holds $\delta^*(q_0, w) = [w]_{\sim_L}$. So $w \in \delta^*(q_0, w)$ and for any $w\binom{a}{*} \in P_2$ holds $w\binom{a}{*} \in \delta^*(q_0, w\binom{a}{*})$.

For any $K_1, K_2 \in Q$ with $K_1 \neq K_2$ holds $K_1 \cap K_2 = \emptyset$, because $Q_1 = P_1/{\sim_L}$ is a partition of P_1, $Q_2 = P_2/{\sim_L}$ is a partition of P_2 and P_1 and P_2 are disjoint.

V is a regular $*$-language, so there is a finite automaton $\mathfrak{A}$ with $\mathfrak{A} = (Q', \Sigma, q_0', \delta', F)$, recognizing V. For two arbitrary words $u, v \in P_1$ with $\delta'^*(q_0', u) = \delta'^*(q_0', v)$ it follows that $u \sim_L v$. So Q_1 is finite. But then, Q_2 must also be finite.

Let $\alpha \in \Sigma^\omega$ and let $\varrho = q_0, q_1, \ldots$ be the unique play of (G, c) on α. Then it holds

$$\begin{aligned} \text{Player 2 wins } \alpha \text{ in } (G,c) &\iff \forall i\colon c(\varrho(i)) = 0 \\ &\iff \text{each prefix } w \sqsubseteq \alpha \text{ is in } V \\ &\iff \alpha \in V^\Box. \end{aligned}$$

So the claim holds for any $V^\Box$ with $V \in \mathcal{K}$.

Induction step:

First case: $L = L_1 \cup L_2$

By induction hypothesis there are deterministic weak parity games (G_1, c_1) with $G_1 = (Q_1^1, Q_2^1, \delta_1)$, initial node $q_0^1 \in Q_1$ and (G_2, c_2) with $G_2 = (Q_1^2, Q_2^2, \delta_2)$ and initial node $q_0^2 \in Q_1$ with the desired properties.

We construct a product game graph where the new nodes are intersections of nodes of the originating game graphs. Define (G, c) with $G = (Q_1, Q_2, \delta)$ by

- $Q_1 = \{K_1 \cap K_2 \mid K_1 \in Q_1^1, K_2 \in Q_1^2, K_1 \cap K_2 \neq \emptyset\}$,
- $Q_2 = \{K_1 \cap K_2 \mid K_1 \in Q_2^1, K_2 \in Q_2^2, K_1 \cap K_2 \neq \emptyset\}$,
- $q_0 = q_0^1 \cap q_0^2$,
- $\delta(K_1 \cap K_2, a) = K_1' \cap K_2'$ where $\delta_1(K_1, a) = K_1'$, $\delta_2(K_2, a) = K_2'$,
- $\delta(K_1 \cap K_2, x) = K_1' \cap K_2'$ where $\delta_1(K_1, x) = K_1'$, $\delta_2(K_2, x) = K_2'$ and
- $c(K_1 \cap K_2) = c_1(K_1) \cdot c_2(K_2)$.

This is indeed a well-defined deterministic weak parity game. The transition function δ is well-defined: For any state $K \in Q$ there is only one representation $K = K_1 \cap K_2$. Let $K = K_1 \cap K_2 = K_1' \cap K_2'$. Then there is a $w \in K_1 \cap K_1'$. But since $K_1 \cap K_1' = \emptyset$ for $K_1 \neq K_1'$ it must hold $K_1 = K_1'$. Analogously for K_2 and K_2'.

The set Q is again finite and each $K \in Q$ is a language from $\mathcal{K}$, because $\mathcal{K}$ is closed under finite intersections.

If the coloring functions c_1 and c_2 have codomains $\{0, \ldots, m_1\}$ and $\{0, \ldots, m_2\}$ then the new coloring function c also has a finite codomain, namely $\{0, \ldots, m_1 \cdot m_2\}$.

By induction over the length of words one can verify that $w \in \delta^*(q_0, w)$ for each $w \in P_1 \cup P_2$.

For $K_1 \cap K_2 \neq K_1' \cap K_2'$ there must hold $K_1 \neq K_1'$ or $K_2 \neq K_2'$. For the former case, it holds $K_1 \cap K_1' = \emptyset$, so $(K_1 \cap K_2) \cap (K_1' \cap K_2') = \emptyset$, too.

The coloring function c is again monotonic, because multiplication is a monotonic operator.

Let $\alpha \in \Sigma^\omega$ and let $\varrho = q_0, q_1, \ldots$ be the unique play of (G, c) on α, ϱ_1 be the unique play of (G_1, c_1) on α and ϱ_2 be the unique play of (G_2, c_2) on α. Because c_1, c_2 and c are monotonic, there is a color $d_1 \in \{0, \ldots, m_1\}$ which is finally assumed in ϱ_1, a color $d_2 \in \{0, \ldots, m_2\}$ which is finally assumed in ϱ_2 and a color $d = d_1 \cdot d_2$ which is finally assumed in ϱ. Then it holds

$$\begin{aligned}
\text{Player 2 wins } \varrho \text{ in } (G, c) &\iff d \text{ is even} \\
&\iff d_1 \text{ is even or } d_2 \text{ is even} \\
&\iff \text{Player 2 wins } \varrho \text{ in } (G_1, c_1) \\
&\qquad \text{or Player 2 wins } \varrho \text{ in } (G_1, c_1) \\
&\iff \alpha(\varrho) \in L_1 \cup L_2.
\end{aligned}$$

So the claim holds for $L_1 \cup L_2$.

Second case: $L = \Sigma^\omega \setminus L_1$

By induction hypothesis there is a deterministic weak parity game (G_1, c_1) with $G_1 = (Q_1^1, Q_2^1, \delta_1)$ and initial node $q_0^1 \in Q_1$ with the desired properties.

We preserve the game arena and just modify the coloring function. Define the new game (G_1, c) with a new coloring function c by

$$\begin{aligned}
c \colon Q &\to \{0, \ldots, m_1 + 1\} \\
c(q) &:= c_1(q) + 1.
\end{aligned}$$

Then c is again monotonic and for every play ϱ it holds

$$\begin{aligned}
\text{Player 2 wins } \varrho \text{ in } (G_1, c) &\iff \text{Player 1 wins } \varrho \text{ in } (G_1, c_1) \\
&\iff \alpha(\varrho) \in \Sigma^\omega \setminus L_1.
\end{aligned}$$

So the claim holds for $\Sigma^\omega \setminus L_1$. □

We are now able to present the general theorem.

Theorem 63. *Let $\mathcal{K}$ be a class of adequate regular $*$-languages. Then every game defined by $L \in \mathcal{K}^w$ is determined with winning strategies in $\mathcal{K}$.*

Moreover, the player who wins can be determined and the winning strategy can be constructed effectively.

Proof. Let $L \in \mathcal{K}^w$ be an instance of Church's Problem. Then by Theorem 62, there is a weak parity game (G, c) with game arena $G = (Q_1, Q_2, \delta)$ and a designated initial node $q_0 \in Q_1$ such that items 1. – 4. are fulfilled.

We define a mapping $h\colon P_1 \cup P_2 \to Q$ by

$$h(w) := \delta^*(q_0, w).$$

This mapping is a homomorphism, because for $w \in P_1$, $a \in \Sigma_1$ and $x \in \Sigma_2$ it holds

$$\begin{aligned}\delta(h(w), a) &= \delta(\delta^*(q_0, w), a)\\ &= \delta(q_0, w\tbinom{a}{*})\\ &= h(w\tbinom{a}{*})\\ &= h(\gamma(w, a))\end{aligned}$$

and

$$\begin{aligned}\delta(h(w\tbinom{a}{*}), x) &= \delta(\delta^*(q_0, w\tbinom{a}{*}), x)\\ &= \delta(q_0, w\tbinom{a}{x})\\ &= h(w\tbinom{a}{x})\\ &= h(\gamma(w\tbinom{a}{*}, x)).\end{aligned}$$

Since weak parity games are determined (Theorem 8), either Player 1 or Player 2 has a winning strategy from q_0 in (G, c).

First case: Player 1 has a winning strategy from q_0. Then let $f_1\colon Q_1 \to \Sigma_1$ be a memoryless winning strategy for Player 1 from q_0 in (G, c). For every $a \in \Sigma_1$ define

$$\begin{aligned}Q_1^a :=&\big\{q \in Q_1 \mid f_1(q) = a\big\} \text{ and}\\ S_a :=&\big\{w \in P_1 \mid h(w) \in Q_1^a\big\}.\end{aligned}$$

Then clearly the sets Q_1^a are pairwise disjoint and cover Q_1, while the sets S_a are pairwise disjoint and cover P_1. So $(S_a)_{a\in\Sigma_1}$ is a strategy for Player 1.

We now show that $(S_a)_{a\in\Sigma_1}$ is winning for Player 1. Let ϱ be a play played according to $(S_a)_{a\in\Sigma_1}$. We want Player 1 to win this play, so we are going to show $\alpha := \alpha(\varrho) \notin L$. Let $\varrho = p_0, p_1, \ldots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \ldots$ in (G, c), because h is a homomorphism. For every even $i \in \mathbb{N}$ there is an $a \in \Sigma_1$ and a $w \in P_1$ such that $p_i = w, p_{i+1} = w\binom{a}{*}$. Since ϱ is played according to $(S_a)_{a\in\Sigma_1}$, clearly $w \in S_a$ and by the definition of S_a and Q_1^a we obtain $h(w) \in Q_1^a$ and $f_1(h(w)) = a$. So ϱ' is played according to f_1. But f_1 is a winning strategy for Player 1 from q_0, so Player 1 wins ϱ'. By item 4. it follows $\alpha \notin L$ and $(S_a)_{a\in\Sigma_1}$ is indeed winning for Player 1 in the game defined by L.

Because of item 2. and 3., we can express S_a by

$$S_a = \bigcup_{K \in Q_1^a} K.$$

Since each $K \in Q_1^a$ is in $\mathcal{K}$ (item 1), S_a is a finite union of $*$-languages from $\mathcal{K}$. So $(S_a)_{a \in \Sigma_1}$ is a $\mathcal{K}$-definable winning strategy.

Second case: Player 2 has a winning strategy from q_0. Then let $f_2 \colon Q_2 \to \Sigma_2$ be a memoryless winning strategy for Player 2 from q_0 in (G, c). For every $x \in \Sigma_2$ define

$$Q_2^x := \{ q \in Q_2 \mid f_2(q) = x \} \text{ and}$$
$$T_x := \{ w \in P_2 \mid h(w) \in Q_2^x \}.$$

Then clearly the sets Q_2^x are pairwise disjoint and cover Q_2, while the sets T_a are pairwise disjoint and cover P_2. So $(T_x)_{x \in \Sigma_2}$ is a strategy for Player 2.

We show that $(T_x)_{x \in \Sigma_2}$ is winning for Player 2. Let ϱ be a play played according to $(T_x)_{x \in \Sigma_2}$. We want player 2 to win this play, so we are going to show $\alpha := \alpha(\varrho) \in L$. Let $\varrho = p_0, p_1, \ldots$ be the sequence of states of ϱ. By applying h to every state, we obtain a play $\varrho' = q_0, q_1, \ldots$ in (G, c), because h is a homomorphism. For every odd $i \in \mathbb{N}$ there is an $a \in \Sigma_1$, an $x \in \Sigma_2$ and a $w \in P_1$ such that $\varrho_i = w\binom{a}{*}$, $\varrho_{i+1} = w\binom{a}{x}$. Since ϱ is played according to $(T_x)_{x \in \Sigma_2}$, clearly $w\binom{a}{*} \in T_x$ and by the definition of T_x and Q_2^x we obtain $h(w) \in Q_2^x$ and $f_2(h(w)) = x$. So ϱ' is played according to f_2. But f_1 is a winning strategy for Player 2 from q_0, so Player 2 wins ϱ'. By item 4. it follows $\alpha \in L$ and $(T_x)_{x \in \Sigma_2}$ is indeed winning for Player 2 in the game defined by L.

Because of item 2. and 3., we can express T_x by

$$T_x = \bigcup_{K \in Q_2^x} K.$$

Since each $K \in Q_2^x$ is in $\mathcal{K}$ (item 1), T_x is a finite union of $*$-languages from $\mathcal{K}$. So $(T_x)_{x \in \Sigma_2}$ is a $\mathcal{K}$-definable winning strategy.

Since the game arena is finite, the winning strategies in the game (G, c) can be constructed effectively and so the winning strategies in $\mathrm{Ch}(L)$ can be constructed effectively, too. □

4.2 Applications and Examples

We now want to apply our general theorem to some example language classes. At first, let us consider an example with finite classes of languages.

Example 64. We consider the game with alphabets $\Sigma_1 = \Sigma_2 = \{0,1\}$, $\Sigma = \Sigma_1 \times \Sigma_2$ where Player 2 wins, iff there is the letter 1 anywhere in the second component of the result of a play α. So

$$L = \Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^\omega.$$

See Table 4.1 and Table 4.2 to follow the construction of the language classes $\mathcal{K}$ and $\mathcal{K}^w$. L can be found as L_2 in Table 4.2. We want to find a class of adequate regular $*$-languages $\mathcal{K}$, such that $L \in \mathcal{K}^w$ and we can apply Theorem 63.

We need every $\sim_L$-class to be in $\mathcal{K}$. This results in the languages $K_1, \ldots, K_4$ in Table 4.1. Now we take the closure of these four languages under finite unions and finite intersections. This results in the other languages K_0 and $K_5, \ldots, K_{15}$.

Looking again at $\mathcal{L}$, we see that L_1 is obtained from K_5 and L_3 is obtained from K_{15} by applying the $^\square$-operator. L_0 and L_2 are their complements. $\mathcal{L}$ is closed under Boolean combinations and each $\sim_{L_i}$-class is in $\mathcal{K}$ for every $L_i \in \mathcal{L}$. So $\mathcal{K}$ is adequate and indeed $\mathcal{L} = \mathcal{K}^w$.

So we can apply Theorem 63 and get the result that every game defined by a language $L_i \in \mathcal{K}^w$ is determined with winning strategies in $\mathcal{K}$.

In fact Player 2 has a winning strategy in $\mathrm{Ch}(L)$. For example she can choose $(T_0, T_1) = (K_4, K_2)$ or $(T_0, T_1) = (K_0, K_9)$ as winning strategies.

We can also apply Theorem 63 on some of the language classes that we examined in Chapter 3. To show that $\mathcal{L}_\mathcal{K}$-definable games are determined with $\mathcal{K}$-definable winning strategies, we have to prove that $\mathcal{K}$ is a class of adequate regular $*$-languages and that $\mathcal{L}_\mathcal{K} \subseteq \mathcal{K}^w$.

Example 65. Let $\mathcal{K}$ be the class of all locally testable $*$-languages (cf. Section 3.2). Each locally testable language is regular, so $\mathcal{K}$ is a class of regular $*$-languages. Thus, in order to apply Theorem 63 and get a result similar to Theorem 22 we have to show two claims:

1. $\mathcal{K}$ is adequate,

2. every locally testable ω-language is in $\mathcal{K}^w$ ($\mathcal{L}_\mathcal{K} \subseteq \mathcal{K}^w$).

Proof. Let us first examine whether $\mathcal{K}$ is adequate. The class of locally testable $*$-languages $\mathcal{K}$ is closed under finite intersections and finite unions. So it remains to show that for every $L \in \mathcal{K}^w$ each class $[u]_{\sim_L}$ is locally testable. Let $L \in \mathcal{K}^w$. It suffices to show that the $\sim_k$ relation from Sec-

name	regular expression	origin
K_0	$\emptyset$	$K_1 \cap K_2$
K_1	$(\binom{0}{0} + \binom{1}{0})^*$	$[\varepsilon]_L$
K_2	$(\binom{0}{0} + \binom{1}{0})^*(\binom{0}{*} + \binom{1}{*})$	$[\binom{0}{*}]_L$
K_3	$\Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*$	$[\binom{0}{1}]_L$
K_4	$\Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*(\binom{0}{*} + \binom{1}{*})$	$[\binom{0}{1}\binom{0}{*}]_L$
K_5	$(\binom{0}{0} + \binom{1}{0})^*(\varepsilon + \binom{0}{*} + \binom{1}{*})$	$K_1 \cup K_2$
K_6	Σ^*	$K_1 \cup K_3$
K_7	$(\binom{0}{0} + \binom{1}{0})^* + \Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*(\binom{0}{*} + \binom{1}{*})$	$K_1 \cup K_4$
K_8	$(\binom{0}{0} + \binom{1}{0})^*(\binom{0}{*} + \binom{1}{*}) + \Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*$	$K_2 \cup K_3$
K_9	$\Sigma^*(\binom{0}{*} + \binom{1}{*})$	$K_2 \cup K_4$
K_{10}	$\Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*(\varepsilon + \binom{0}{*} + \binom{1}{*})$	$K_3 \cup K_4$
K_{11}	$(\binom{0}{0} + \binom{1}{0})^*(\varepsilon + \binom{0}{*} + \binom{1}{*}) + \Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*$	$K_1 \cup K_2 \cup K_3$
K_{12}	$(\binom{0}{0} + \binom{1}{0})^*(\varepsilon + \binom{0}{*} + \binom{1}{*}) + \Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*(\binom{0}{*} + \binom{1}{*})$	$K_1 \cup K_2 \cup K_4$
K_{13}	$(\binom{0}{0} + \binom{1}{0})^* + \Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*(\varepsilon + \binom{0}{*} + \binom{1}{*})$	$K_1 \cup K_3 \cup K_4$
K_{14}	$(\binom{0}{0} + \binom{1}{0})^*(\binom{0}{*} + \binom{1}{*}) + \Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^*(\varepsilon + \binom{0}{*} + \binom{1}{*})$	$K_2 \cup K_3 \cup K_4$
K_{15}	$\Sigma^*(\varepsilon + \binom{0}{*} + \binom{1}{*})$	$K_1 \cup \cdots \cup K_4$

Table 4.1: $\mathcal{K} = \{K_0, \ldots, K_{15}\}$ of Example 64

name	regular expression	origin
L_0	$\emptyset$	$\overline{L_3}$
L_1	$(\binom{0}{0} + \binom{1}{0})^\omega$	$K_5^\square$
$L = L_2$	$\Sigma^*(\binom{0}{1} + \binom{1}{1})\Sigma^\omega$	$\overline{L_1}$
L_3	Σ^ω	$K_{15}^\square$

Table 4.2: $\mathcal{L} = \{L_0, \ldots, L_3\}$ of Example 64

tion 3.2 refines the $\sim_L$ relation. For any two words $u, v \in P_1$ it holds

$$\begin{aligned}
u \sim_k v \;&\Rightarrow \mathrm{Infix}_k(u) = \mathrm{Infix}_k(v) \\
&\quad \wedge \mathrm{Prefix}_{k-1}(u) = \mathrm{Prefix}_{k-1}(v) \\
&\quad \wedge \mathrm{Suffix}_{k-1}(u) = \mathrm{Suffix}_{k-1}(v) \\
&\Rightarrow \forall \beta \in \Sigma^\omega \colon \mathrm{Infix}_k(u\beta) = \mathrm{Infix}_k(v\beta) \\
&\quad \wedge \mathrm{Prefix}_{k-1}(u\beta) = \mathrm{Prefix}_{k-1}(v\beta) \\
&\Rightarrow \forall \beta \in \Sigma^\omega \colon (u\beta \in L \iff v\beta \in L) \\
&\Rightarrow u \sim_L v.
\end{aligned}$$

Analogously for $u, v \in P_2$. So any $[u]_{\sim_L}$ class is the union of $\sim_k$ classes and therefore locally testable. This implies that $\mathcal{K}$ is adequate.

We show that $\mathcal{L}_{\mathcal{K}} \subseteq \mathcal{K}^w$. Let L be a locally testable ω-language. Then there is a $k \in \mathbb{N}$ such that

$$L = \bigcup_{i=1}^{m} [\alpha_i]_{\sim_k}.$$

We look for $U_i, V_i \in \mathcal{K}$ such that

$$U_i^{\Box} \setminus V_i^{\Box} = [\alpha_i]_{\sim_k},$$

because then $L = \bigcup_{i=1}^{m}(U_i^{\Box} \setminus V_i^{\Box})$ and therefore L is a Boolean combination of languages $V^{\Box}$ with $V \in \mathcal{K}$ and as such $L \in \mathcal{K}^w$.

For every α_i we define

$$\begin{aligned}
U_i := \big\{ w \in P_1 \;\big|\; &\mathrm{Prefix}_{k-1}(w) \subseteq \mathrm{Prefix}_{k-1}(\alpha_i) \\
&\wedge \mathrm{Infix}_k(w) \subseteq \mathrm{Infix}_k(\alpha_i) \big\} \\
\cup \big\{ v\tbinom{a}{*} \in P_2 \;\big|\; &v \in P_1, a \in \Sigma_1 \big\}
\end{aligned}$$

and

$$\begin{aligned}
V_i := \big\{ w \in P_1 \;\big|\; &\mathrm{Prefix}_{k-1}(w) \neq \mathrm{Prefix}_{k-1}(\alpha_i) \\
&\vee \mathrm{Infix}_k(w) \neq \mathrm{Infix}_k(\alpha_i) \big\} \\
\cup \big\{ v\tbinom{a}{*} \in P_2 \;\big|\; &v \in P_1, a \in \Sigma_1 \big\}.
\end{aligned}$$

Both U_i and V_i are locally testable, because they depend only on the set of prefixes respectively factors of at most length k.

We still have to show $U_i^{\Box} \setminus V_i^{\Box} = [\alpha_i]_{\sim_k}$. The first direction is $U_i^{\Box} \setminus V_i^{\Box} \subseteq [\alpha_i]_{\sim_k}$. Let $\beta \in U_i^{\Box} \setminus V_i^{\Box}$. Since $\beta \notin V_i^{\Box}$, there is a $w_1 \sqsubseteq \beta$ with $w_1 \notin V_i$. From the definition of V_i it follows $w_1 \in P_1$ and

$$\begin{aligned}
&\mathrm{Prefix}_{k-1}(w_1) = \mathrm{Prefix}_{k-1}(\alpha_i) \\
&\wedge \mathrm{Infix}_k(w_1) = \mathrm{Infix}_k(\alpha_i).
\end{aligned}$$

Since $\beta \in U_i^\Box$, for all $w \sqsubseteq \beta$ it holds

$$\begin{aligned} &\mathrm{Prefix}_{k-1}(w) \subseteq \mathrm{Prefix}_{k-1}(\alpha_i) \\ &\wedge \mathrm{Infix}_k(w) \subseteq \mathrm{Infix}_k(\alpha_i). \end{aligned}$$

The set of prefixes and the set of factors can only grow as we proceed to longer prefixes of β. So for each prefix $w \sqsubseteq \beta, w \in P_1$ with w_1 being a prefix of w these sets stay constant. So

$$\begin{aligned} &\mathrm{Prefix}_{k-1}(w_1) = \mathrm{Prefix}_{k-1}(\alpha_i) = \mathrm{Prefix}_{k-1}(\beta) \\ &\wedge \mathrm{Infix}_k(w_1) = \mathrm{Infix}_k(\alpha_i) = \mathrm{Infix}_k(\beta) \end{aligned}$$

and $\beta \sim_k \alpha_i$ and therefore $\beta \in [\alpha_i]_{\sim_k}$.

We show the other direction $[\alpha_i]_{\sim_k} \subseteq U_i^\Box \setminus V_i^\Box$. Let $\beta \in [\alpha_i]_{\sim_k}$. For every prefix $w \sqsubseteq \beta$, $w \in P_1$ it holds

$$\begin{aligned} &\mathrm{Prefix}_{k-1}(w) \subseteq \mathrm{Prefix}_{k-1}(\alpha_i) \\ &\wedge \mathrm{Infix}_k(w) \subseteq \mathrm{Infix}_k(\alpha_i). \end{aligned}$$

So for every $w \sqsubseteq \beta$ it holds $w \in U_i$ and therefore $\beta \in U_i^\Box$. There is a prefix $w_2 \sqsubseteq \beta$ with

$$\begin{aligned} &\mathrm{Prefix}_{k-1}(w_2) = \mathrm{Prefix}_{k-1}(\alpha_i) \\ &\wedge \mathrm{Infix}_k(w_2) = \mathrm{Infix}_k(\alpha_i). \end{aligned}$$

So $w_2 \notin V_i$ and therefore $\beta \notin V_i^\Box$. Altogether $\beta \in U_i^\Box \setminus V_i^\Box$ and this proves our proposition that $\mathcal{L}_{\mathcal{K}} \subseteq \mathcal{K}^w$.

The other inclusion direction does not hold as we will show in the following. We exhibit an ω-language that is in $\mathcal{K}^w$ but not locally testable. Let $\Sigma_1 = \{a\}$, $\Sigma_2 = \{a, b, c\}$, $k = 1$ and let

$$\begin{aligned} V = {} & [\varepsilon]_{\sim_k} \cup [\tbinom{a}{*}]_{\sim_k} \cup [\tbinom{a}{a}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{a}{*}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{a}{b}]_{\sim_k} \\ & \cup [\tbinom{a}{a}\tbinom{a}{b}\tbinom{a}{*}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{a}{b}\tbinom{a}{c}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{a}{b}\tbinom{a}{c}\tbinom{a}{*}]_{\sim_k}. \end{aligned}$$

Then $V^\Box = \tbinom{a}{a}\tbinom{a}{a}^*\tbinom{a}{b}(\tbinom{a}{a} + \tbinom{a}{b})^*\tbinom{a}{c}(\tbinom{a}{a} + \tbinom{a}{b} + \tbinom{a}{c})^\omega$ is the language of all infinite words over $\Sigma_1 \times \Sigma_2$ starting with $\tbinom{a}{a}$ and where $\tbinom{a}{b}$ occurs before $\tbinom{a}{c}$. $V^\Box$ is not locally testable. Assume it is k-locally testable. Then the word

$$\alpha := \tbinom{a}{a}^{k-1}\tbinom{a}{b}\tbinom{a}{a}^{k-1}\tbinom{a}{c}\tbinom{a}{a}^\omega$$

is in $V^\Box$ and

$$\beta := \tbinom{a}{a}^{k-1}\tbinom{a}{c}\tbinom{a}{a}^{k-1}\tbinom{a}{b}\tbinom{a}{a}^\omega$$

has the same factors of length k and the same prefix of length $k-1$ as α. So $\alpha \sim_k \beta$ and β must be in $V^\Box$, too. But in β $\tbinom{a}{c}$ occurs before $\tbinom{a}{b}$, so this is a contradiction to the definition of $V^\Box$. Therefore $\mathcal{K}^w \not\subseteq \mathcal{L}_{\mathcal{K}}$. $\Box$

$\mathcal{K}$ and $\mathcal{K}^w$ meet the conditions of Theorem 63. Then every game defined by $L \in \mathcal{K}^w$ is determined with winning strategies in $\mathcal{K}$. This means locally testable games are determined with locally testable winning strategies.

We have seen that the length of the prefixes k grows when proceeding from $\mathcal{K}$ to $\mathcal{K}^w$. So the above example can only work, if we do not claim the parameter k to stay constant.

Example 66. Let $\mathcal{K}$ be the class of all piecewise testable $*$-languages (cf. Section 3.5). The general theorem is not applicable to $\mathcal{K}$, because $\mathcal{K}$ is not adequate. There is an $L \in \mathcal{K}^w$ and a word u such that $[u]_{\sim_L}$ is not piecewise testable.

Let $\Sigma_1 = \Sigma_2 = \{a, b\}$, $k = 1$ and let

$$\begin{aligned} V = [\varepsilon]_{\sim_k} &\cup [\tbinom{a}{*}]_{\sim_k} \cup [\tbinom{a}{a}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{b}{*}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{a}{*}]_{\sim_k} \\ &\cup [\tbinom{a}{a}\tbinom{b}{b}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{b}{b}\tbinom{a}{*}]_{\sim_k} \cup [\tbinom{a}{a}\tbinom{b}{b}\tbinom{b}{*}]_{\sim_k}. \end{aligned}$$

Then set $L = V^{\square} = \tbinom{a}{a}(\tbinom{a}{a} + \tbinom{b}{b})^{\omega}$. Compare this with Example 33. L is not piecewise testable. For $u = \tbinom{a}{a}$ the class $[u]_{\sim_L}$ is $\tbinom{a}{a}(\tbinom{a}{a} + \tbinom{b}{b})^*$, which is not piecewise testable. So $[u]_{\sim_L}$ does not belong to $\mathcal{K}$ and therefore $\mathcal{K}$ is not adequate.

The last example fails, because the $^{\square}$-operator can be used to express bounded prefixes of words, but piecewise testable languages do not distinguish between different prefixes. If we would change the definition of piecewise testable languages and also regard prefixes in it, then we expect the general theorem to work. We conclude that it also works with locally threshold testable languages, but not with modulo counting languages.

We know that piecewise testable games are determined with piecewise testable winning strategies. So an obvious idea to make Example 66 work would be to drop the requirement that every class $[u]_{\sim_L}$ belongs to $\mathcal{K}$. Instead one could demand $\mathcal{K}$ to be closed under complement, too. Let $\Sigma_1 = \Sigma_2 = \{a, b\}$ and

$$\begin{aligned} V := \varepsilon + \tbinom{a}{*} + \tbinom{a}{\Sigma_2} + \tbinom{a}{\Sigma_2}\tbinom{b}{*} + \tbinom{a}{\Sigma_2}\tbinom{b}{\Sigma_2} \\ + \tbinom{a}{\Sigma_2}\tbinom{b}{\Sigma_2}\tbinom{\Sigma_1}{*}^* + \tbinom{a}{\Sigma_2}\tbinom{b}{\Sigma_2}\tbinom{\Sigma_1}{\Sigma_2}^*. \end{aligned}$$

If we choose $\mathcal{K} = \{V, \overline{V}\}$ then $\mathcal{K}$ is closed under Boolean combinations. We get

$$V^{\square} = \tbinom{a}{\Sigma_2}\tbinom{b}{\Sigma_2}\tbinom{\Sigma_1}{\Sigma_2}^{\omega}$$

and the game defined by $L := \overline{V^{\square}}$ is won by Player 1 iff the first two letters of the first component are a and b. Clearly Player 1 has a winning strategy. He just has to choose a in the first round and b in the second. But there is no winning strategy in $\mathcal{K}$ as one can easily see.

Chapter 5

Conclusion

We investigated Church's Problem for certain subclasses of regular languages. We showed that X-definable games are determined with X-definable winning strategies, where X is the class of locally testable languages, respectively piecewise testable languages. Moreover, k-locally r-threshold testable games are determined with k-locally r-threshold testable winning strategies, and k-piecewise r-threshold testable games are determined with k-piecewise r-threshold testable winning strategies, with the same parameters k and r. We considered two possibilities for logical modulo counting quantifiers and generated the language class of locally positional modulo testable languages. For this class we showed that k-locally positional q-modulo testable games are determined with k-locally positional q-modulo testable winning strategies. Then we found a general concept to proceed from a class of regular $*$-languages $\mathcal{K}$ to a class of weak ω-languages $\mathcal{K}^w$, such that $\mathcal{K}^w$-definable games are determined with $\mathcal{K}$-definable winning strategies.

By an excursion to combinatorics on words, we found out that each piecewise threshold testable language is also piecewise testable and we used this result to state that k-piecewise r-threshold testable games are determined with $(k + r - 1)$-piecewise testable winning strategies.

A natural question which arises is whether there are other types of games for which these results hold. One could consider Gale-Stewart games where the plays are represented by linear sequences of letters instead of sequences of pairs of letters. Other examples are Banach-Mazur games (see e.g. [Mos80]) where the players pick integers or finite sequences of bits. If both players' choices are restricted to the same alphabet, then winning conditions and winning strategies can again be expressed by (tuples of) languages. Thus, we should obtain similar results for these types of games.

The Büchi-Landweber Theorem and the results of Selivanov [Sel07] and Rabinovich and Thomas [RT07] show that there are several strong games for which the result holds. The question is, whether there is also a general result for strong games: Can we conclude from a class of $*$-languages to a class of strong ω-languages?

Another possibility to extend the results of this work, is to consider not only subclasses of regular languages, but also non-regular ones. The proof idea would be the same. The only difference would consist in an infinite graph of the weak parity game. Therefore, an effective construction of the winning strategies would not follow directly anymore. However, determinacy still holds, if the winning conditions are Borel, and winning strategies that are of the same type as the winning conditions might still exist.

Considering the class of context-free languages, Walukiewicz [Wal01] showed that the result at least holds for deterministic context-free languages. It would be interesting to know whether it also holds for the general case of context-free languages or for other subclasses of the context-free languages.

Bibliography

[BL69] Julius R. Büchi and Lawrence H. Landweber, *Solving sequential conditions by finite-state strategies*, Trans. Amer. Math. Soc. **138** (1969), 295–311.

[Chu57] Alonzo Church, *Applications of recursive arithmetic to the problem of circuit synthesis*, Summaries of the Summer Institute of Symbolic Logic (Ithaca, N.Y.), vol. 1, Cornell Univ., 1957, pp. 3–50.

[Chu63] ———, *Logic, arithmetic, and automata*, Proc. Int. Congr. Math. 1962 (Djursholm, Sweden), Inst. Mittag-Leffler, 1963, pp. 23–35.

[CK96] Christian Choffrut and Juhani Karhumäki, *Combinatorics of words*, Handbook of Formal Languages, 1996.

[EJ91] E. Allen Emerson and Charanjit S. Jutla, *Tree automata, mu-calculus and determinacy (extended abstract)*, FOCS, IEEE, 1991, pp. 368–377.

[GH82] Yuri Gurevich and Leo Harrington, *Trees, automata, and games*, STOC, ACM, 1982, pp. 60–65.

[GS53] David Gale and Frank M. Stewart, *Infinite games with perfect information*, Ann. Math. Studies **28** (1953), 245–266.

[GTW02] Erich Grädel, Wolfgang Thomas, and Thomas Wilke (eds.), *Automata, logics, and infinite games*, Lecture Notes in Computer Science, vol. 2500, Springer, 2002.

[LT00] Christof Löding and Wolfgang Thomas, *Alternating automata and logics over infinite words*, IFIP TCS (Jan van Leeuwen, Osamu Watanabe, Masami Hagiya, Peter D. Mosses, and Takayasu Ito, eds.), Lecture Notes in Computer Science, vol. 1872, Springer, 2000, pp. 521–535.

[Mar75] Donald A. Martin, *Borel determinacy*, Ann. of Math. **102** (1975), no. 2, 363–371.

[McN65] Robert McNaughton, *Finite-state infinite games*, Project MAC Rep. (1965).

[McN74] ______, *Algebraic decision procedures for local testability*, Mathematical Systems Theory **8** (1974), no. 1, 60–76.

[Mos80] Yiannis N. Moschovakis, *Descriptive set theory*, North Holland, 1980.

[Mos91] Andrzej W. Mostowski, *Games with forbidden positions*, Tech. Report 78, University of Gdansk, 1991.

[Pin86] Jean-Éric Pin, *Varieties of formal languages*, North Oxford, London and Plenum, New-York, 1986.

[PP04] Dominique Perrin and Jean-Éric Pin, *Infinite words*, Elsevier, Amsterdam, 2004.

[Rab72] Michael O. Rabin, *Automata on infinite objects and Church's problem*, CBMS Regional Conference Series in Mathematics (1972), no. 13.

[RT07] Alexander Rabinovich and Wolfgang Thomas, *Logical refinements of Church's problem*, CSL (Jacques Duparc and Thomas A. Henzinger, eds.), Lecture Notes in Computer Science, vol. 4646, Springer, 2007, pp. 69–83.

[Sal03] Arto Salomaa, *Counting (scattered) subwords*, Bulletin of the EATCS **81** (2003), 165–179.

[Sel07] Victor L. Selivanov, *Fine hierarchy of regular aperiodic omega-languages*, Developments in Language Theory (Tero Harju, Juhani Karhumäki, and Arto Lepistö, eds.), Lecture Notes in Computer Science, vol. 4588, Springer, 2007, pp. 399–410.

[Sim75] Imre Simon, *Piecewise testable events*, Automata Theory and Formal Languages (H. Barkhage, ed.), Lecture Notes in Computer Science, vol. 33, Springer, 1975, pp. 214–222.

[Str94] Howard Straubing, *Finite automata, formal logic, and circuit complexity*, Birkhäuser Verlag, Basel, Switzerland, 1994.

[SW74] Ludwig Staiger and Klaus W. Wagner, *Automatentheoretische und automatenfreie charakterisierungen topologischer klassen regulärer folgenmengen*, Elektronische Informationsverarbeitung und Kybernetik **10** (1974), no. 7, 379–392.

[Tho82] Wolfgang Thomas, *Classifying regular events in symbolic logic*, J. Comput. Syst. Sci. **25** (1982), no. 3, 360–376.

[Tho95] ______, *On the synthesis of strategies in infinite games*, STACS, 1995, pp. 1–13.

[Tho08] ______, *Church's problem and a tour through automata theory*, Pillars of Computer Science (Arnon Avron, Nachum Dershowitz, and Alexander Rabinovich, eds.), Lecture Notes in Computer Science, vol. 4800, Springer, 2008, pp. 635–655.

[Wal01] Igor Walukiewicz, *Pushdown processes: Games and model-checking*, Inf. Comput. **164** (2001), no. 2, 234–263.

[Wil91] Thomas Wilke, *An Eilenberg theorem for infinity-languages*, ICALP (Javier Leach Albert, Burkhard Monien, and Mario Rodríguez-Artalejo, eds.), Lecture Notes in Computer Science, vol. 510, Springer, 1991, pp. 588–599.

[Wil93] Thomas Wilke, *Locally threshold testable languages of infinite words*, STACS'93 (P. Enjalbert, A. Finkel, and Klaus W. Wagner, eds.), Lecture Notes in Computer Science, vol. 665, Springer, 1993, pp. 607–616.

[Zie98] Wieslaw Zielonka, *Infinite games on finitely coloured graphs with applications to automata on infinite trees*, Theor. Comput. Sci. **200** (1998), no. 1–2, 135–183.