COMENIUS UNIVERSITY IN BRATISLAVA
FACULTY OF MATHEMATICS, PHYSICS AND INFORMATICS

Grammars With Energy and Finite Approximations of Languages

Master's Thesis

2014 Bc. András Varga

COMENIUS UNIVERSITY IN BRATISLAVA
FACULTY OF MATHEMATICS, PHYSICS AND INFORMATICS

Grammars With Energy and Finite Approximations of Languages

Master's Thesis

Study Program:	Computer Science
Branch of Study:	2508 Computer Science, Informatics
Department:	Department of Computer Science
Supervisor:	prof. RNDr. Branislav Rovan, PhD.

Bratislava, 2014 **Bc. András Varga**

Comenius University in Bratislava
Faculty of Mathematics, Physics and Informatics

THESIS ASSIGNMENT

Name and Surname:	Bc. András Varga
Study programme:	Computer Science (Single degree study, master II. deg., full time form)
Field of Study:	9.2.1. Computer Science, Informatics
Type of Thesis:	Diploma Thesis
Language of Thesis:	English
Secondary language:	Slovak

Title:	Grammars With Energy and Finite Approximations of Languages
Aim:	Find modifications of the definition of grammars with energy allowing one-sided finite approximations of context-free languages and explore relevant properties.
Keywords:	Context-free grammars with energy, approximations of labguages

Supervisor:	prof. RNDr. Branislav Rovan, PhD.
Department:	FMFI.KI - Department of Computer Science
Head of department:	doc. RNDr. Daniel Olejár, PhD.
Assigned:	15.11.2011

Approved: 15.11.2011 prof. RNDr. Branislav Rovan, PhD.
Guarantor of Study Programme

.....
Student

.....
Supervisor

I honorably declare that I am a sole author of this thesis. The ideas and results contained in it are based on my research using only the listed literature.

. . . . . .

Acknowledgement

I want to thank to my supervisor Branislav Rovan and also to József Bukor, Rastislav Královič and Ján Plesník for their help and constructive advices, as well as to my family standing by my side.

Abstrakt

Hlavným zámerom tejto práce je formálne popísať konečnú approximáciu nekonečného jazyka, dokázať niektoré užitočné vlastnosti našej formálnej definície a zaviesť pojem vzdialenosti pre túto konštrukciu. Prvým krokom je úprava gramatík s energiou z diplomovej práce Jánošíka, aby umožnili jednostrannú approximáciu daného jazyka. Definujeme pojem postupnosti approximácie pre daný jazyk a dokážeme niektoré vlastnosti tejto konštrukcie. Zavedieme niektoré operácie pre postupnosti approximácie a zadefinujeme silnú a slabú formu vzdialenosti medzi postupnosťami approximácie. V práci je zavedená metóda vypočítania vzdialenosti medzi dvoma postupnosťami approximácie v prípade, keď vieme ako boli tieto postupnosti vytvorené, pomocou akých operácií. Uvedieme ako súvisia gramatiky s energiou postupnosti approximácie, a definujeme triedu postupnosti approximácií generovaných pomocou gramatík s energiami. Ako poslednú vec dokážeme, že silná forma vzdialenosti za istých podmienok úzko súvisí s gramatikami s energiou.

KĽÚČOVÉ SLOVÁ: gramatika s energiou, monotonné postupnosti konečných jazykov, postupnosti approximácie, vzdialenosť medzi jazykmi

Abstract

The main goal of this thesis is to define the finite approximation of infinite languages, prove some useful properties of this formal definition, and introduce the distance measure on this construction. As the first step, we needed to modify the definition of grammars with energy from Jánošík's thesis, allowing one-sided approximations of the given language. We define the monotone sequences of finite languages, which are used as sequences of approximations for a given language. We introduce operations on these sequences, and prove some of their properties. We also define the string and weak distance measure between monotone sequences of approximations. In this thesis we show how the strong and the weak distance measures behave, when we can obtain additional information about the monotone sequences being measured. Later we show how can grammars with energy generate sequences of approximations, and we introduce classes of sequences of approximations generated by grammars with energy. Finally, we show that under specific conditions the strong distance measure is related to the strict grammars with energy.

KEYWORDS: Grammars With Energy, Monotone Sequences of Finite Languages, Sequences of Approximations, Distances on Languages

Contents

Introduction and Motivation

This thesis is dedicated to make the first step to define a distance measure between languages as well as grammars. Our intention is to generalize one of the distance measures existing between words. This new distance measure should help us to highlight the similarities and differences between the languages (or grammars) being measured.

Our solution is based on approximations of the given language by finite languages. This idea, introduced by Jánošík in Thesis [4], gives us a tool to approximate infinite languages by finite ones for context-free languages. This tool is called *grammar with energy*, which is a modification of context-free grammars.

A new structure is introduced, called *monotone sequence of finite languages*, which is a generalization of the construction created by grammars with energy, and can be used also to describe the length-based approximations, taking into account the length of words from the language. We prove the basic properties of this structure and introduce some operations on them.
Each monotone sequence of finite languages is a *sequence of approximations* for a specific language.

Later we introduce the distance measure between the monotone sequences in general as well as two specific distance measures later used in this thesis called *strong* and *weak* distance measure. We prove some basic properties of these defined distances, and the relation between chosen operations and the distance measure in general, too.

As our intention is introduce a distance measure between the grammars and languages itself, we introduce and examine new classes of monotone sequences of finite languages, similarly to the classes of languages which can be created by some *strict* grammar with energy. We introduce three basic classes based on the complexity of grammars generating the monotone sequences. These classes are *regular*, *linear*, and *context-free* sequences of approximations.

In the later chapters of this thesis we analyze properties of these classes of sequences of approximations, and show some advanced properties of the strong distance measure for these classes as well.

Chapter 1

Grammars With Energy

This chapter contains the definition and basic properties of *grammar with energy* and *strict grammar with energy* as well as some examples. The second part of this chapter contains a derivation of some useful properties associated with *strict grammars with energy*.

1.1 Definitions

Grammars with energy were introduced by Jánošík in Thesis [4]. They were used to define finite approximations of context-free languages. On the other hand, approximations in [4] may contain words, which do not belong into the language being approximated. In this thesis we present a modification of grammars with energy allowing only approximation by finite subsets of the language being approximated.

Definition 1.1.1. *A* grammar with energy *is a 4-tuple* $G = (N, \Sigma, P, \sigma)$*, where*

i) N *and* Σ *are disjoint finite sets of* non-terminal *and* terminal symbols *respectively;*

ii) P *is a finite subset of* $N \times ((N \times (0,1)) \cup \Sigma)^*$*, where* $(0,1)$ *denotes the open real interval. The elements of* P *are called* rewrite rules*;*

iii) σ *in* N *is the starting non-terminal symbol.*

After the definition of the grammar with energy itself, we define the application of rewrite rules from P. For this purpose we explain our notations first:

Notation 1.1.2. *Similarly to [4], let* $\xi^{[k]}$ *denote non-terminal symbols from the right side of the rewrite rules, where* k *is a coefficient of this non-terminal symbol. Otherwise let* ξ^e *denote non-terminal symbols in the sentential form, where* e *is a real number from* $(0,1]$*. In this case* e *is the energy level of non-terminal symbol* ξ *in the sentential form.*

Notation 1.1.3. *Similarly to [4], let $\hat{G}$ denote the underlying context-free grammar of the grammar with energy G obtained by omitting the real numbers associated with non-terminals.*

Definition 1.1.4. *The* derivation step for a grammar with energy G with minimal energy level e *written as $\Rightarrow_{G,e}$ is a relation defined on the set $V = ((N \times (0,1]) \cup \Sigma)^*$, where $(0,1]$ is a real interval, open from left, closed from right.*
$u \Rightarrow_{G,e} v \iff \exists n \in \mathbb{N}; x, y, v' \in V; v_1, v_2, \ldots, v_{n+1} \in \Sigma^*;$
$A, A_1, A_2, \ldots, A_n \in N; e_A, e_1, e_2, \ldots, e_n \in (0,1]; k_1, k_2, \ldots, k_n \in (0,1); e_A \geq e$ *such that the following statements hold:*

i) $u = xA^{e_A}y;$

ii) $v' = xv_1A_1^{e_1}v_2A_2^{e_2}\ldots v_nA_n^{e_n}v_{n+1}y;$

iii) $A \to v_1A_1^{[k_1]}v_2A_2^{[k_2]}\ldots v_nA_n^{[k_n]}v_{n+1} \in P;$

iv) For all i from $1, 2, \ldots, n$ $e_i = k_ie_A$ holds;

v) v is obtained from v' by a substitution of each A_i, where $e_i < e$, by some word from the given finite set of words $\Psi_{A_i} = \{w | w \in \Sigma^*, A_i \Rightarrow_{\hat{G}}^* w\}$. The words in Ψ_{A_i} are terminal words, which can be derived in the context-free grammar $\hat{G}$ from the non-terminal symbol A_i.

Definition 1.1.5. *Let $G = (N, \Sigma, P, \sigma)$ be a grammar with energy. The* language generated by G with a minimal energy-level (or energy-threshold) e *is*

$$L_e(G) = \{w \in \Sigma^* | \sigma^1 \Rightarrow_{G,e}^* w\}.$$

This general definition of grammars with energy is specified in the definition of strict grammars with energy.

Definition 1.1.6. *Grammar with energy G is called* strict *grammar with energy, iff*

i) Every non-terminal symbol is accessible: $\forall \xi \in N : \exists a, b \in (N \cup \Sigma)^+ : \sigma \Rightarrow^* a\xi b;$

ii) Every non-terminal symbol is productive: $\forall \xi \in N : \exists w \in \Sigma^* : \xi \Rightarrow^* w;$

iii) In $\Rightarrow_{G,e}$ part v), when $e_i < e$ the shortest word *is substituted to the sentential form , which is produced by the corresponding non-terminal symbol A_i in $\hat{G}$.*

When there are more shortest words, all of them is substituted one by one into the sentential form and each substitution creates a new derived sentential form similarly to the application of the rewriting rules in the context-free grammars.

All of these restrictions are algorithmically constructable: The first and the second condition is a part of the standard normal-form for the context-free grammars, therefore these are constructable algorithmically. The following lemma says that the third condition is algorithmically constructable as well.

Lemma 1.1.7. *Let $G = (N, \Sigma, P, \sigma)$ be a context free grammar without inaccessible and unproductive non-terminal symbols. We can algorithmically compute the set of shortest words produced by an arbitrary non-terminal symbol.*

Proof. Without loss of generality ξ in N. So we are searching for all shortest words in the context free language $G = (N, \Sigma, P, \xi)$. We prove, that for all shortest words there is a derivation with no cycle within its derivation. Proof by contradiction. Let w be one of the shortest words. Let us assume, that there is a cycle within every derivation of this shortest word. So the shortest derivation of this shortest word shall be: $\xi \Rightarrow_G^* u_1\xi_r v_1 \Rightarrow_G^* u_2\xi_r v_2 \Rightarrow_G^* w$, where ξ_r is a non-terminal symbol causing the cycle. But with omitting this cycle we obtain a word w', which is at most as long as w, because no rewrite rule can delete a terminal symbol from the sentential form. Therefore $|w| = |w'|$ can hold only, if $|u_1| = |u_2|$ and $|v_1| = |v_2|$ and because context-free grammars can not alter the terminal symbols in the sentential form there is a possible derivation, where $u_1 = u_2$ and $v_1 = v_2$. So the cycle is not producing terminal symbols. This is a contradiction, because either $|w'| < |w|$, so w is not one of the shortest words or there is a derivation shorter than the shortest derivation mentioned above. □

From the lemma above follows, that we can algorithmically obtain the set of shortest words derivable from ξ in the context-free grammar $G = (N, \Sigma, P, \sigma)$. For this we shall compute all derivation trees of grammar $G = (N, \Sigma, P, \xi)$ to the depth $|N| + 1$ and choose all shortest terminal words from there.

Now we shall show the usage of these definitions on examples. First, we can notice, that strict grammars with energy are not equivalent to the general definition. Because of the good properties of these strict grammars with energy, we shall use strict grammars with energy instead of the general definition in this paper.

Example 1.1.8. Let $G = \{N = \{\sigma, \alpha, \beta\}, \Sigma = \{a, b, c\}, P, \sigma\}$ be a strict grammar with energy with a set of rules

$$P = \left\{\sigma \to \sigma^{[0.9]}\alpha^{[0.8]} \mid \sigma^{[0.9]}\beta^{[0.8]} \mid ab \mid bc, \alpha \to a\alpha^{[0.8]}b \mid ab, \beta \to b\beta^{[0.8]}c \mid bc\right\}.$$

The shortest words produced by the non-terminal symbols are ab and bc for σ; ab and bc for α and β respectively. The following words can be derived by this grammar

with an energy threshold 0.95:

$$\sigma^1 \Rightarrow_{G,0.95} \sigma^{0.9}\alpha^{0.8} \Rightarrow_{G,0.95} abab$$
$$\sigma^1 \Rightarrow_{G,0.95} \sigma^{0.9}\alpha^{0.8} \Rightarrow_{G,0.95} bcab$$
$$\sigma^1 \Rightarrow_{G,0.95} \sigma^{0.9}\beta^{0.8} \Rightarrow_{G,0.95} abbc$$
$$\sigma^1 \Rightarrow_{G,0.95} \sigma^{0.9}\beta^{0.8} \Rightarrow_{G,0.95} bcbc$$

Word *ababbcaabb* can be derived by this grammar as well with an energy threshold 0.75. One of the possible derivation is:

$$\underline{\sigma^1} \Rightarrow_{G,0.75} \underline{\sigma^{0.9}}\alpha^{0.8} \Rightarrow_{G,0.75} \sigma^{0.81}\underline{\beta^{0.72}}\alpha^{0.8} \rightarrow \underline{\sigma^{0.81}}bc\alpha^{0.8} \Rightarrow_{G,0.75}$$
$$\Rightarrow_{G,0.75} \underline{\sigma^{0.729}\alpha^{0.64}}bc\alpha^{0.8} \rightarrow ababbc\underline{\alpha^{0.8}} \Rightarrow_{G,0.75} ababbca\underline{\alpha^{0.64}}b \rightarrow ababbcaabb$$

$\Rightarrow_{G,0.75}$ is the derivation step and $\rightarrow$ refers to the last action of the derivation step, namely the substitution of the shortest word generated by the non-terminal symbol without enough energy. The underlined non-terminals are used in the given step of derivation.

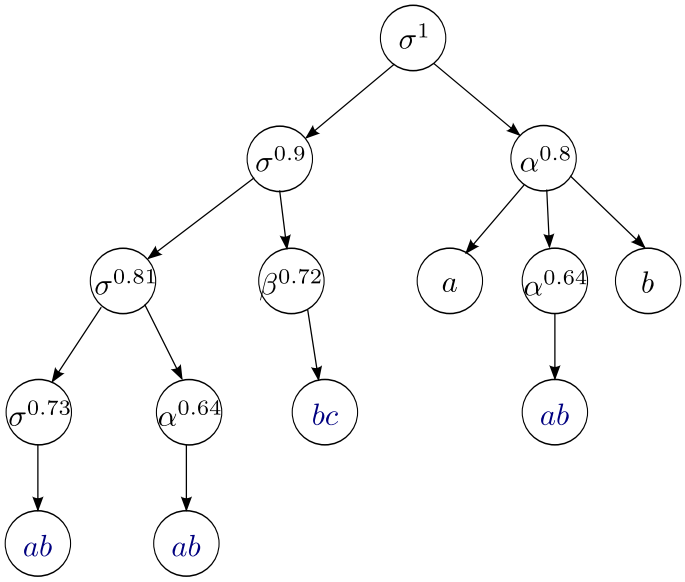

Figure 1.1: The derivation tree for the word *ababbcaabb* with energy threshold 0.75. The blue words are the shortest terminal words, which can be derived from the given non-terminal symbol. (The energy levels of the non-terminal symbols are rounded.)

1.2 Basic Properties

Theorem 1.2.1. *For every context-free language $\bar{G}$ a strict grammar with energy G exists such that:*

(i) For all $e \in (0,1) : \forall w \in L_e(G) \Rightarrow w \in L(\bar{G})$*;*

(ii) For all $w \in L(\bar{G}) \Rightarrow \exists e \in (0,1) x \in L_e(G)$.

Proof. Without loss of generality let $\bar{G}$ be a context-free grammar with only accessible and productive non-terminal symbols.

(i) The first statement holds, because every rewrite rule in G is a rewrite rule in $\bar{G}$ as well, moreover the set of shortest words is derived from the rewrite rules of $\bar{G}$ too.

(ii) This part is proved in [4], the basic idea of this proof is, that we can set such a low energy-threshold e, that the whole derivation in $\bar{G}$ becomes available in G too without reaching the energy-threshold e.

□

Theorem 1.2.2. *Let* G *be a strict grammar with energy. Let* $0 < e_1 < e_2 < 1$ *be real numbers. The following statement holds:*

$$L_{e_2}(G) \subseteq L_{e_1}(G).$$

Proof. We have to prove, that every word in $L_{e_2}(G)$ is derivable in $L_{e_1}(G)$. In other words we have to prove, that every word derivable with the higher energy threshold is derivable with a lower energy threshold too.

$$w \in L_{e_2}(G) \Rightarrow w \in L_{e_1}(G)$$

We have three cases:

(i) While deriving w with an energy threshold e_2, the substitutions (of shortest words) are not required. In this case every non-terminal symbol in every sentential form of this derivation has more energy, than e_2. Which means every non-terminal symbol has a higher energy level, than e_1 too, because $e_2 > e_1$. Therefore during the usage of rewrite rules with energy threshold e_1 there is no substitution required too. In this situation the following statement holds: $\sigma^1 \Rightarrow^*_{G,e_2} w$ implies $\sigma^1 \Rightarrow^*_{G,e_1} w$.

(ii) In this case, we have to substitute at least one non-terminal symbol while using a rewrite rule with the energy-threshold e_2. Let us divide this situation into two cases. In this case we shall assume, that when using these substitutions every non-terminal symbol has lower energy level than e_1 too. ($e_1 < e_2$). Then every substitution made with the energy-threshold e_2 is made with the energy-threshold e_1 too. So in this case every rewrite rule used in $\sigma^1 \Rightarrow^*_{G,e_2} w$ has the same effect in the derivation $\sigma^1 \Rightarrow^*_{G,e_1} w$.

(iii) In this, last case, we have to assume, that at least one non-terminal symbol is substituted, moreover this non-terminal symbol has energy between e_1 and e_2. Without loss of generality the first non-terminal symbol in derivation where the energy is between e_1 and e_2 is ξ has energy e_ξ, and it holds: $e_1 < e_\xi < e_2$. So we have the following situation $\sigma^1 \Rightarrow^*_{G,e_2} x\xi^{e_\xi}y \Rightarrow^*_{G,e_2} w$, where $x, y \in (N \times (0,1) \cup T)^*$. From the (i) and (ii) we know, that $\sigma^1 \Rightarrow^*_{G,e_1} x\xi^{e_\xi}y$ holds too. So we have to prove, that the shortest words, substituted into this sentential form are derivable with energy-threshold e_1 too.

Without loss of generality one of these shortest words is s. There are two cases now.

First, when e_1 has such a low energy, that the whole derivation $\xi \Rightarrow^*_{\hat{G}} s$ is enabled without substitutions in energy-threshold e_1. In this situation we have no problem with the derivation with energy threshold w in e_1 we can produce the same part of word w, which is substituted in the application of rewrite rule.

On the other hand, let us have such energy level e_1, that within $\xi^{e_\xi} \Rightarrow^*_{G,e_1} s'$ substitution is used somewhere. Let us prove, that $s' = s$ by contradiction. So let us assume, that $s' \neq s$. In $\xi^{e_\xi} \Rightarrow^*_{G,e_1} s'$ the same rewrite rules, as in $\xi \Rightarrow^*_{\hat{G}} s$ are used, but there is a point, where one non-terminal symbol has less energy than e_1. In this case the shortest words are substituted, and because every rewrite rule is context-free, these substituted words are not further modified. So we know, that s' has the form $s' = x_1y_1x_1y_2 \dots x_n$, where each x_i (for $i = 1$ to n) is derived from the rewrite rules of $\hat{G}$, so these parts of words are parts of s too. On the other hand all y_i (for $i = 1$ to $n-1$) are the parts, which are substituted. So $s = x_1q_1x_1q_2 \dots x_n$. Where for all $i = 1, 2 \dots (n-1)$: q_i is derived from the rewrite rules of $\hat{G}$. Now we know, that every y_i is the shortest possible word. But y_i cannot be shorter than q_i, because that would be a contradiction: a word shorter than s derivable from ξ (for example s'). So every y_i has the same length as q_i. (The second implication holds, because of the fact, that y_i are the shortest words derivable.) In grammars we are always substituting every possible choice, so there is a situation, where for all $i = 1, 2 \dots (n-1) : q_i = y_i$. So the statement holds in this case too.

□

Chapter 2

Monotone Sequences of Finite Languages and Sequences of Approximations

In this chapter we introduce a new model for approximation of languages called *sequence of approximations*. Our motivation is to find a model, which includes all approximations by the length of the words as well as approximations generated by strict grammars with energy. The following two examples illustrate some important properties of approximations. These properties are used in the definition of sequences of approximations.

Example 2.0.3. Consider the linear language $L = \{a^n b^n | n \geq 0\}$. We can define a finite set (finite language) $L_k = \{a^i b^i | k \geq i \geq 0\}$ for every k. In this case every L_k can be considered as an approximation of language L. As an example $L_3 = \{\varepsilon, ab, aabb, aaabbb\}$, which can be considered as a beginning of the language L in some ordering.

Note that for all k $L_k \subseteq L$ and $L_k \subseteq L_{k+1}$.

Example 2.0.4. Similar languages can be defined not only for the linear languages, but for not context-free languages as well. For example $L = \{a^n b^n c^n | n \geq 0\}$ is not a context-free language. On the other hand $L_9 = \{\varepsilon, abc, aabbcc, aaabbbccc\}$ consists those words from L, which are not longer then 9 characters.

First we introduce how can we approximate a language using the length of the words from the given language. We can define the following finite sets for every language L and for every natural number n: $L_n = \{w \in L | |w| \leq n\}$. With growing n the finite language L_n is "better and better" approximation of the original language L. It is easy to see, that L_n is a finite language for every n and can not reach L. On the other hand,

as it is shown in [4] we can define the limit of approximation similarly to limes inferior or limes superior.

Another observation is, that for every pair of numbers n and m, where $n \leq m$ for the corresponding sets it holds: $L_n \subseteq L_m$.

In the Example 2.0.3 we defined such sequence of finite sets (namely L_k for every natural number k), which satisfies the conditions above. On the other hand we can see, that the conditions will hold even if we skip some of the natural numbers. For example a sequence of sets $L_1, L_3, L_4, L_6, L_8, \ldots$ satisfies the conditions above as well. In this case we expect, that $\{a^n b^n | n \geq 0\}$ will be the "limit" of these sequences.

Now consider strict grammars with energy. In the previous chapter we have already shown that $L_{e_2}(G) \subseteq L_{e_1}(G)$ for every strict grammar with energy G and for two real numbers $0 < e_1 < e_2 < 1$. This property corresponds to the last property from above. We can easily see, that $L_e(G)$ is a finite subset of $L(G)$ for every real number $0 < e < 1$ and grammar with energy G. The limit mentioned above is defined in Thesis [4] for grammars with energy. We shall define it below for sequences of approximations in general.

2.1 Definitions

Now we define the *sequence of approximations* satisfying the following three properties (mentioned above):

1. every language in the sequence of approximations is finite;
2. every language in the sequence of approximations is a superset of all previous languages;
3. there is a way to describe the "limit" of the approximation.

Definition 2.1.1. *The sequence of finite languages* $\Phi = \{L_i\}_{i=1}^{\infty}$ *is called* monotone, *iff for all* $i \geq 1 : L_i \subseteq L_{i+1}$.

Definition 2.1.2. $L_\Phi := \lim_{i \to \infty} \Phi = \lim_{i \to \infty} L_i = \bigcup_{i=1}^{\infty} L_i$ *is called a* limit *of the monotone sequence of finite languages.*[1]

Definition 2.1.3. *A monotone sequence of finite languages* Φ *is a* sequence of approximations for language L *iff* $L_\Phi = L$. *(We shall also say that* Φ *is approximating the language* L.*)*
Languages $L_i \in \Phi$ *are called* finite approximations *of* L.

[1] This limit is a simpler variant of the definition from [4], because we do not allow words not included in the language.

Lemma 2.1.4. *Let Φ be a monotone sequence of finite languages. Then for a language L the following two conditions hold iff $L = L_\Phi$:*

(i) For all $w \in L$ exists i, such that $w \in L_i$;

(ii) For all i and for all $w \in L_i$: $w \in L$.

Proof. We have to prove, that those conditions are equivalent to $\bigcup_{i=1}^{\infty} L_i$.

$L \subseteq \bigcup_{i=1}^{\infty} L_i$: From the first condition for all $w \in L$ exists such i that $w \in L_i$, thus for all $w \in L$: $w \in \bigcup_{i=1}^{\infty} L_i$.

$\bigcup_{i=1}^{\infty} L_i \subseteq L$: This part of the proof is almost identical to the part above.

The second implication ($\Longleftarrow$) is trivial from the properties of union. □

Example 2.1.5. Now we show, that the languages L_k from Example 2.0.3 indeed approximate $\{a^n b^n | n \geq 1\}$ i.e. $\Phi = \{L_k\}_{k=1}^{\infty}$ is a sequence of approximations for language $\{a^n b^n | n \geq 0\}$. Recall that L_k is defined as $L_k = \{a^n b^n | k \geq n \geq 0\}$. It is easy to see, that L_k is finite for every k. $L_k \subset L_{k+1}$ holds as well.
Now we show, that language $L = \{a^n b^n | n \geq 0\}$ is the limit of this monotone sequence of finite languages. We need to check two conditions:

(i) For all $w \in L$ exists $k \in \mathbb{N}$, where $w \in L_k$;

(ii) For all $k \in \mathbb{N}$ and for all $w \in L_k$: $w \in L$.

The second condition holds because for every natural k L_k is a subset of L. The first condition holds as well, because for every word w in L (words can not be infinite) has to be contained in the set $L_{|w|/2}$.
This is an example, that our definitions describe what we expected from them.

Example 2.1.6. One more example for a sequence of approximations. As we pointed out we can define sequences of approximations even for languages, which are not context-free. Consider the language $L = \{w | w \in \{a, b, c\}^* \wedge \#_a(w) = \#_b(w) = \#_c(w)\}$. We can construct $L_k = \{w | w \in \{a, b, c\}^* \wedge \#_a(w) = \#_b(w) = \#_c(w) \wedge |w| \leq k\}$ for every natural number k. It is easy to see that $\Phi = \{L_k\}_{k=1}^{\infty}$ is a sequence of approximations for the language L (the proof is similar to the one in the example above).

Notation 2.1.7. *Let us use the following notation: whenever there are more monotone sequences used they are distinguished by apostrophes on their notations. For example Φ, Φ' or Φ''. Finite languages from a monotone sequence are marked by the same number of apostrophes as the monotone sequence itself. So L_i is from Φ, L_i' is from Φ, L_i'' from Φ'' and so on.*

Now we define a standard construction for creating a new sequence of approximations from an existing one.

Definition 2.1.8. *Let Φ be a monotone sequence of finite languages and let $I = i_1 < i_2 < i_3 < ... < i_k < ...$ be a strictly increasing sequence of natural numbers. Then $\Phi' = \{L_{i_k}\}_{k=1}^{\infty}$ is a monotone sequence as well and Φ' is called a monotone* subsequence *of Φ. This fact is denoted by Φ^I.*

One way to see the subsequence of approximations for a particular sequence of approximations is, that we want to "speed up" the approximating process, so we skip some of the steps from the original sequence. From this point of view the definition of the sequence of approximations holds, because the definition above does not affect the finiteness of approximations nor the subset ordering.

Moreover it is easy to see, that every monotone subsequence is approximating the same language as the original monotone sequence. Every word from the limit of the original sequence is in the subsequence as well, because it appears in the original sequence. On the other hand every finite approximation in the monotone subsequence is from the original sequence, which means every word from every finite approximation is contained in the original monotone sequence.
So we can say:

Lemma 2.1.9. *Let Φ be a monotone sequence of finite languages and let Φ^I be its subsequence. Then the following statement holds:*

$$L_\Phi = L_{\Phi^I}.$$

Example 2.1.10. In Example 2.1.5 we show, that $\Phi = \{a^n b^n | k \geq n \geq 0\}_{k=1}^{\infty}$ is a sequence of approximations. $\Phi' = \{a^n b^n | 2k \geq n \geq 0\}_{k=1}^{\infty}$ is a subsequence of approximations of Φ, thus Φ' is a sequence of approximations approximating $L = \{a^n b^n | n \geq 0\}$. We can consider Φ' as a sequence of approximations, which is approximating L by the length of the words in language L, but using only the even length for approximating purposes.

Another common property is transitivity. Being a subsequence of some monotone sequence is transitive, thus if Φ' is a subsequence of Φ and Φ'' is a subsequence of Φ' implies that Φ'' is a subsequence of Φ as well. Being a subsequence is reflexive and antisymmetric as well.

Now we introduce another standard notation, which is the equivalence of two monotone sequences of finite languages.

Definition 2.1.11. *Let $\Phi = \{L_i\}_{i=1}^{\infty}$ and $\Phi' = \{L'_i\}_{i=1}^{\infty}$ be monotone sequences of finite languages. We say, that Φ and Φ' are equivalent iff from some $n_0 \in \mathbb{N}$ for every $n > n_0$ $L_n = L'_n$. The notation for this fact is $\Phi \approx \Phi'$.*

Lemma 2.1.12. *Let Φ and Φ' be equivalent monotone sequences of finite languages ($\Phi \approx \Phi'$). Then the following statement holds:*

$$L_\Phi = L_{\Phi'}.$$

The proof of this lemma is trivial. It is easy to see as well, that the relation being equivalent on the sequences of approximations is reflexive, symmetric and transitive, thus it is a relation of equivalence.

2.2 Operations Defined on Monotone Sequences of Finite Languages

Now we modify some of the usual language operations. We define the *union*, *disjoint union*, *prefix*, *suffix* operations as well as an operation which allows us to take out finite number of words from the monotone sequence. We show the definition of each operation mentioned above in that particular order, and we show, how these operations affect the limit of the approximation.

Definition 2.2.1. *The* union *of two sequences $\Phi = \{L_i\}_{i=1}^{\infty}$ and $\Phi' = \{L'_i\}_{i=1}^{\infty}$ is a sequence denoted by $\Phi \cup \Phi'$. Let $\Phi \cup \Phi'$ be a sequence of sets $\{L''_i\}_{i=1}^{\infty}$ where for each i $L''_i = L_i \cup L'_i$.*

Definition 2.2.2. *The* disjoint union *of two sequences Φ and Φ' is a sequence denoted by $\Phi \uplus \Phi'$. If L_Φ and $L_{\Phi'}$ are languages with no common symbol, then $\Phi \uplus \Phi' = \Phi \cup \Phi'$. Otherwise this operation is not defined.*

Definition 2.2.3. *Let Φ be a sequence and let w be a word. w is called as* prefix *of $\Phi' = w\Phi = \{L'_i\}_{i=1}^{\infty}$ when for every i $L'_i = w.L_i$. Similarly w is called as* suffix *of $\Phi' = \Phi w = \{L'_i\}_{i=1}^{\infty}$ when for every i $L'_i = L_i.w$.*

Definition 2.2.4. *Let Φ be a sequence and let S be finite set of words from L_Φ ($S \subset L_\Phi$). Then $\Phi' = \Phi - S$ is a sequence, where for every i $L'_i = L_i \setminus S$. The operation described above is called* finite deletion *and marked as $-$.*

Now let us show that the operations above are well defined for the monotone sequences as well.

Proposition 2.2.5. *Let Φ and Φ' be monotone sequences of a finite languages, let w be a word and let $S \subseteq L_\Phi$ be a finite set. The following statements hold:*

a) $\Phi \cup \Phi'$ *is a monotone sequence*

b) $\Phi \uplus \Phi'$ *is a monotone sequence*

c) $w\Phi$ *is a monotone sequence*

d) Φw *is a monotone sequence*

e) $\Phi - S$ *is a monotone sequence.*

Proof. Let us prove the *a*) part of the proposition the other parts of the proof are similar.

Now we have to prove, that $\Phi \cup \Phi'$ is a monotone sequence. We have to show, that every condition holds. Obviously all languages in $\Phi \cup \Phi'$ are finite, because they are unions of two finite languages. Now let us refer as $\{L_i''\}_{i=1}^{\infty}$ to $\Phi \cup \Phi'$, so $L_i'' \subseteq L_{i+1}''$ needs to be proved. As Φ and Φ' are monotone sequences $L_i \subseteq L_{i+1}$ and $L_i' \subseteq L_{i+1}'$ hold, therefore $L_i \cup L_i' \subseteq L_{i+1} \cup L_{i+1}'$, which is exactly $L_i'' \subseteq L_{i+1}''$. □

Now let us show the limit of the monotone sequence, which is a result of the operations above.

Proposition 2.2.6. *Let Φ and Φ' be monotone sequences of a finite languages, let w be a word and let $S \subseteq L_\Phi$ be a finite set. The following statements hold:*

a) $L_{\Phi \cup \Phi'} = L_\Phi \cup L_{\Phi'}$

b) $L_{\Phi \uplus \Phi'} = L_\Phi \cup L_{\Phi'}$

c) $L_{w\Phi} = wL_\Phi$

d) $L_{\Phi w} = L_\Phi w$

e) $L_{\Phi - S} = L_\Phi - S$.

Proof. Similarly to the proof above let us prove the *a*) part of the proposition as the other parts are analogical.

Every word from $L_i \cup L_i'$ is from $L_\Phi \cup L_{\Phi'}$. The second condition holds as well, because every word w in $L_\Phi \cup L_{\Phi'}$ is either from L_Φ or from $L_{\Phi'}$, so there is such i, that $L_i \cup L_i'$ contains w. □

Example 2.2.7. We showed in the Example 2.0.3 and proved afterwise, that $\Phi = \{L_k\}_{k=1}^{\infty}$ where $L_k = \{a^n b^n | k \geq n \geq 0\}$ is a sequence of approximations for language $\{a^n b^n | n \geq 0\}$. Then $c\Phi$ is a sequence of approximations as well approximating language $L = \{ca^n b^n | n \geq 0\}$.
Similarly $\Phi - \{\varepsilon, ab, aabb\}$ is approximating $L = \{a^n b^n | n \geq 3\}$.

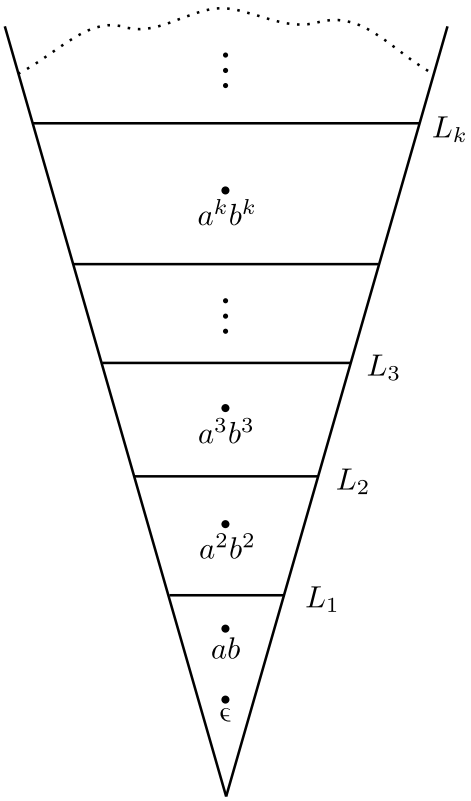

Figure 2.1: The visualization of the sequence of approximations (as a monotone sequence) from Example 2.1.5.

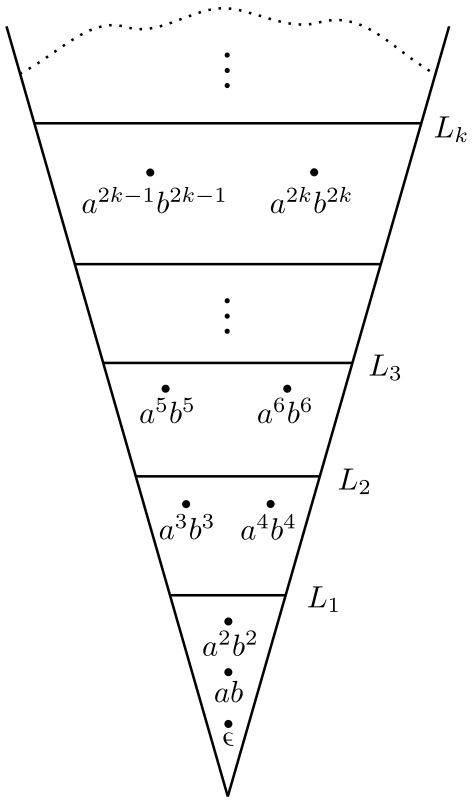

Figure 2.2: The visualization of its subsequence of approximations from Example 2.1.10.

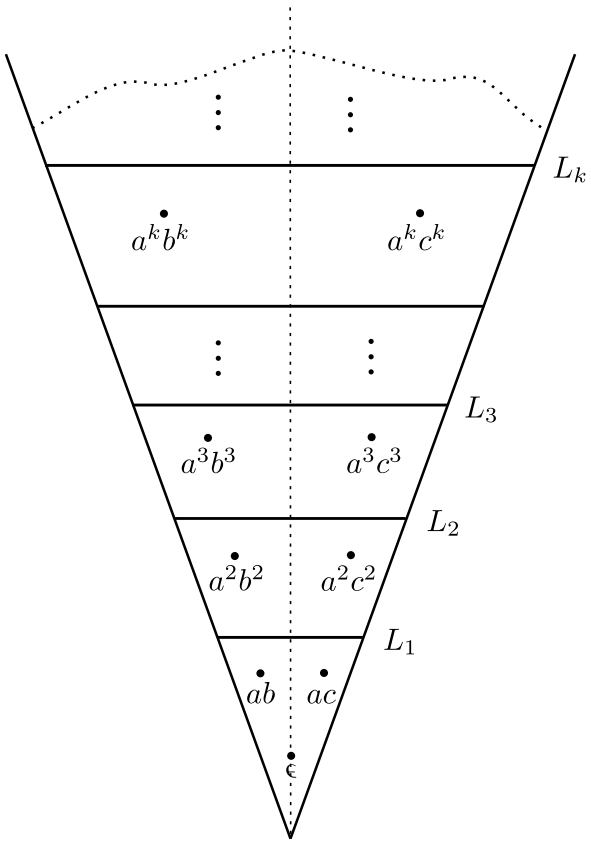

Figure 2.3: The visualization of the monotone sequence which is created as an union of two monotone sequences (one of left and one of right). The words included in both prime monotone sequences are placed on the lane.

Chapter 3

Distances in General

3.1 Distances on Words

We can distinguish various distance measures between two words (strings or character sequences). Some of them are defined and analyzed in Tomáš Kulich's master's Thesis ([7]). Let us now recall some facts about these distances on words.

The most common group of these distance measures is called edit-distances. They are based on string operations such as insertion, deletion, substitution, etc. Edit-distances are defined as a minimal number (or minimal cost) of those operations, which is transforming one word to another. The edit-distance between two words is 0, iff the two given words are the same, otherwise the edit-distance between two words can not be 0.

The *common (or basic) edit-distance*, sometimes called as Levenshtein distance, is analyzed in Thesis [7]. The common edit-distance is defined as the minimal number of insertions, deletions and substitutions of one character. The common edit-distance between two words can be computed as follows: $d(u, v) = |u| + |v| - 2|LCS(u, v)|$, where $LCS(u, v)$ is the longest common subsequence of the words u and v.

In *Damerau–Levenshtein distance* the transposition of two adjacent characters is added to the three operations allowed by the common edit-distance. The only operation allowed by the *Hamming distance* is substitution. Therefore this distance give us the minimum number of substitutions required to alter one word to another.

We prefer to use the *relative edit-distance*, which is defined in Thesis [7], and computed as $\frac{d(u,v)}{|u|+|v|}$, where $d(u, v)$ is the common edit-distance. The relative edit-distance expresses the importance of the operations which transform one word to the other word. The basic idea is, that the same number of operations between two short words and two longer words has to be distinguished, because short words with the same number of operations will be less similar, than long words, where longer parts of the words are

similar to each other.

There are some distance measures between two strings based on sequence alignment used in bioinformatics motivated by the structures of DNA and RNA. In bioinformatics, a sequence alignment is a way of arranging sequences of DNA, RNA or proteins to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. Smith–Waterman algorithm is one of the algorithms computing the sequence alignment.

The Google search engine uses its own distance measure between two words called *normalized Google distance*. With this distance measure 0 means the two words are practically the same, whereas two independent words have distance 1. There is a possibility, that the normalized Google distance between two words is ∞, in which case the two given words never appear together. The normalized Google distance is analyzed in Paper [1].

The last distance mentioned here is the *informed distance measure*. Informed distance measure is the minimal information needed for the construction of x with a knowledge of y. The formal definition and the analysis of this distance is in [9].

3.2 Distances Between a Word and a Language

Similarly to the geometric view there is a way to define the distance between a word and a language. The distance between a word and a language is the minimal distance between the word and every element (word) of the language.

However, the geometric distance between two languages based on this does not give us information about the "similarity" of the two languages, because when one word is in both languages, the distance will be 0, even when these languages are completely different except that particular word. As an example we can see that with such approach each pair of languages would have a zero distance, which contain the empty word ε.

Chapter 4

Distances on Monotone Sequences

We strive to define a distance measure between two languages. The basic idea of the following definitions is the separation of this distance measure into three levels. The basic level is defined on *sequences of approximations of the given language*. The second level is defined on *grammars* and the highest level is defined on *languages*.

In this thesis we use the *relative edit-distances* from [7]. All definitions and theorems are based on it, unless it is mentioned otherwise.

In this thesis we define the distance measure between two sequences of approximations in general, but we are using only some particular special cases of this distance, which are mentioned below.

4.1 The Minimal Weight Maximal Matching

We base our definitions of distances between two sequences of approximations on the *minimal weight maximal matching on complete bipartite graphs*. We first recall some basic facts about the minimal weight maximal matching.

The minimal weight maximal matching is mentioned e.g. in papers [8], [2], [3], [5]. We show the definition of the corresponding objects similarly to [5].

Definition 4.1.1. *Let* $\boldsymbol{G} = (V, E)$ *be a bipartite graph*[1] *with the bipartition* (A, B) *and weight function* $w : E \to \mathbb{R}$. *The* maximum weight bipartite matching *is a matching* M *maximizing the weight of matching, given by* $w(M) = \sum_{e \in M} w(e)$.

Definition 4.1.2. *Let* $\boldsymbol{G} = (V, E)$ *be a bipartite graph with the bipartition* (A, B) *and weight function* $w : E \to \mathbb{R} \cup \{\infty\}$. *The* minimum weight perfect matching *(also called*

[1]Graphs are usually denoted by G just like grammars. Therefore, in this thesis bold characters ($\mathbf{G}$, $\mathbf{H}$, etc.) consistently refer to graphs and normal characters (G, $\hat{G}$, G_1, etc.) refer to grammars.

as minimal weight maximal matching*) in bipartite graphs is a perfect matching M minimizing* $w(M) = \sum_{e \in M} w(e)$.

The minimal weight maximal matching can be defined for any graph in general. In 1965 Edmonds introduced an algorithm, which finds the minimal weight maximal matching on every graph. This algorithm is based on linear programming, generalizing the idea of Kuhn (1955) called Hungarian Method. The Hungarian Method is based on the proof of theorem from [3], proved by Dénes Kőnig. This algorithm works only on bipartite graphs, therefore it is simpler than Edmonds' algorithm. Our definitions are based on bipartite graphs as well, so the *Hungarian Method* is sufficient for our purposes.

These algorithms are not introduced in this section, due to the fact they are described and proved in the papers above.

One of these results will be used several times in our proofs. We therefore state here a version that best fits our needs.

Lemma 4.1.3. *Let $\boldsymbol{G} = (V, E)$ be a bipartite graph with the bipartition (A, B) and weight function $w : E \to \mathbb{R}$. Without loss of generality $|A| > |B|$. Let C denote the set of unmatched vertices from A in a minimal weight maximal matching. For every edge (a, b) from this minimal weight maximal matching, where $a \in A \setminus C$, $b \in B$ $\forall c \in C : \ w(a, b) \leq w(c, b)$ holds.*

Proof. By contradiction. Let M denote the minimal weight maximal matching from the statement. Let us assume, that there is a vertex c in C and (a, b) in M, such that $w(a, b) > w(c, b)$. So we can construct another maximal matching M' as $M' := M \setminus (a, b) \cup (c, b)$. M' is a maximal matching, which has less weight than M, because we have substituted only one edge with another one with less weight. This is a contradiction. □

4.2 The Abstract Definition and its Special Cases

As it is mentioned in the preface of this chapter we define the distance measure between two monotone sequences in general. This abstract definition however is not used in this thesis, because of its complexity. The idea behind this is to allow more different distance measures between the sequences of approximations which have some similar properties.

Definition 4.2.1. *Let Φ and Φ' be monotone sequences, let $R_i \subseteq L_i \times L'_i$ be a relation on words in L_i and L'_i. Let f be a function assigning to each i and the set of pairs (u, v)*

in R_i a real number D_i from the interval $[0,1]$, where $[0,1]$ is a closed real interval. Then the distance between Φ and Φ', denoted as $D(\Phi,\Phi')$, is defined as a limit of the sequence $\{D_i\}_{i=1}^{\infty}$.
If such limit does not exist, then Φ and Φ' are incomparable.

Note that the distance between two monotone sequences (if it exists) is a real number from the interval $[0,1]$.
We now show some special cases of this definition. The *strong* and the *weak* distance defined below are the main distance measures used in this thesis. We can define some trivial distances by picking trivial functions f. For example if f is a constant function e.g. $f(X,i)=0$ or $f(X,i)=1$ the distance measure will be trivial.

D_{zero} is defined by the function $f(X,i)=0$. $D_{zero}(\Phi,\Phi')=0$ for every pair of monotone sequences.
Similarly we can define D_{one} by $f(X,i)=1$, which is a trivial distance $D_{one}(\Phi,\Phi')=1$ for every pair of monotone sequences.
We can define a distance, which is always undefined, which means all pairs of monotone sequences are incomparable according to this distance. For this purpose we can define the function f by

$$f(X,i)=\begin{cases}0 & \text{if } i \text{ is an even number}\\ 1 & \text{if } i \text{ is an odd number}\end{cases}$$

and we can denote this distance as $D_{\perp}$.

Definition 4.2.2. *Let Φ and Φ' be monotone sequences of finite languages. Let us define the* strong distance *in the following way: the relation R_i is a minimal weight maximal matching on the complete bipartite graph, where the partitions are L_i and L'_i and each edge holds the value of the distance between the two words incident to the particular edge; the function f is the average function.*
So for every i $D_i^{(S)}$ is the average value of the edges contained in the minimal weight maximal matching between L_i and L'_i.
Notation: $D_S(\Phi,\Phi')$

We define the *weak distance* similarly to the *strong*.

Definition 4.2.3. *Let Φ and Φ' be monotone sequences of finite languages. Let us define the* weak distance *in the following way: the relation R_i is a minimal weight maximal matching on the similar graph as above, but the partitions are only the previously unmatched vertexes from L_i and L'_i, this matching is combined with the previous relation (for L_{i-1} and L'_{i-1},). The function f is the average function.*
Notation: $D_W(\Phi,\Phi')$

We can see, that there are more ways to define the same distance measure. For example the undefined distance $D_\perp$ can be defined by using the reverse of the function above. For every trivial distance shown above we can set the relation as we want, so trivially D_{zero}, D_{one} and $D_\perp$ can be defined in more ways.

Notation 4.2.4. *Let $D(\Phi, \Phi')$ be ∞ if the particular distance between Φ and Φ' does not exist.*

4.3 Basic Properties of the Strong and the Weak Distance

Lemma 4.3.1. *If both the weak and the strong distance measures exist at the same time between one pair of monotone sequences, then $D_S(\Phi, \Phi') \leq D_W(\Phi, \Phi')$.*

Proof. In the ith step of the approximation there are two different maximal matchings on the same graph. One of them is partially minimal weight matching, but the second one is the minimal weight matching, so there is no matching with less weight than this one. So for each step $D_i^{(S)} \leq D_i^{(W)}$ and each sequence is convergent, so for the limits $D_S(\Phi, \Phi') \leq D_W(\Phi, \Phi')$ must hold as well. □

The following two statements are without proof, because they are following directly from the definitions, the properties of the minimal weight maximal matching and from the sandwich theorem (squeeze lemma).

Lemma 4.3.2. *The following statements hold:*

$$\exists D_W(\Phi, \Phi') \text{ and } D_W(\Phi, \Phi') = 0 \Rightarrow \exists D_S(\Phi, \Phi') \text{ and } D_S(\Phi, \Phi') = 0;$$
$$\exists D_S(\Phi, \Phi') \text{ and } D_S(\Phi, \Phi') = 1 \Rightarrow \exists D_W(\Phi, \Phi') \text{ and } D_W(\Phi, \Phi') = 1.$$

Lemma 4.3.3. *For the distances the reflexivity holds: $D_W(\Phi, \Phi)$ and $D_S(\Phi, \Phi)$ always exist and have value 0. If $D_W(\Phi, \Phi')$ or $D_S(\Phi, \Phi')$ exists, then the symmetry holds too:*

$$D_W(\Phi, \Phi') = D_W(\Phi', \Phi);$$
$$D_S(\Phi, \Phi') = D_S(\Phi', \Phi).$$

Lemma 4.3.4. *Let Φ be a monotone sequence and Φ^I its subsequence. Then $D_S(\Phi, \Phi^I)$ always exists and*

$$D_S(\Phi, \Phi^I) = 0.$$

Proof. From the definition of the subsequence we know, that for every j $L_j \subseteq L_j^I$. From it follows, that we can match every word from L_j with itself, thus creating a maximal matching with 0 weight. Better matching does not exists, so we have constructed an infinite sequence of zeros, which is of course convergent to 0. □

For the further analysis of the distances we need to define a particular sequence corresponding to $D_i^{(W)}$ and $D_i^{(S)}$, allowing us to use Cauchy-Bolzano convergence criteria. So let us define two sequences a_i and A_i for $D_i^{(W)}$ and $D_i^{(S)}$ respectively. Satisfying $D_r^{(W)} = \sum_{i=1}^{r} a_i$ and $D_r^{(S)} = \sum_{i=1}^{r} A_i$ thus $a_1 = D_1^{(W)}$ and $a_r = D_r^{(W)} - D_{r-1}^{(W)}$ for each $r \geq 2$. Similarly $A_1 = D_1^{(S)}$ and $A_r = D_r^{(S)} - D_{r-1}^{(S)}$ for each $r \geq 2$.

We must notice, that these sequences may include some positive and some negative values too. In this way we can obtain a necessary and sufficient condition for a_i and A_i to be convergent. Cauchy-Bolzano convergence criterion from [6] has to hold: The sequence $\sum_{i=1}^{\infty} a_i$ is convergent iff

$$\forall \varepsilon > 0, \exists n_0 \in \mathbb{N}, \forall n \in \mathbb{N}, n > n_0, \forall p \in \mathbb{N} : |a_{n+1} + a_{n+2} + \ldots + a_{n+p}| < \varepsilon.$$

Without the loss of generality $n_0 > 1$. Replacing each a_r by $D_r^{(W)} - D_{r-1}^{(W)}$ we obtain

$$|\sum_{j=n+1}^{n+p} (D_r^{(W)} - D_{r-1}^{(W)})| = |\sum_{j=n+1}^{n+p} D_r^{(W)} - \sum_{j=n}^{n+p-1} D_r^{(W)}| = |D_{n+p}^{(W)} - D_n^{(W)}| < \varepsilon.$$

Thus we can state the following theorem.

Theorem 4.3.5. *Sequence* $\sum_{i=1}^{\infty} a_i$ *is convergent iff*

$$\forall \varepsilon > 0, \exists n_0 \in \mathbb{N}, n_0 > 1, \forall n \in \mathbb{N}, n > n_0 \forall p \in \mathbb{N} : |D_{n+p}^{(W)} - D_n^{(W)}| < \varepsilon.$$

Similarly sequence $\sum_{i=1}^{\infty} A_i$ *is convergent iff*

$$\forall \varepsilon > 0, \exists n_0 \in \mathbb{N}, n_0 > 1, \forall n \in \mathbb{N}, n > n_0 \forall p \in \mathbb{N} : |D_{n+p}^{(S)} - D_n^{(S)}| < \varepsilon.$$

Theorem 4.3.6. *Neither strong nor weak distance measure on monotone sequences is a distance in a metric space.*

Proof. The first condition of metric space is, that $d(x, y) = 0 \iff x = y$. This condition does not hold. For example for languages $L_1 = \{a^n b^n | n \geq 0\}$ and $L_2 = \{a^n b^n \cup c | n \geq 0\}$ we can construct sequences of approximations as $\Phi' = \{a^i b^i | i < n\}_{n=1}^{\infty}$ and $\Phi'' = \{c \cup a^i b^i | i < n\}_{n=1}^{\infty}$ respectively. $D_S(\Phi', \Phi'') = D_W(\Phi', \Phi'') = D_{zero}(\Phi', \Phi'')$, but these two sequences of approximations are not equivalent, moreover the limit of these sequences are different as well.

The third condition, the triangle inequality does not hold too. In general, there are cases, when the triangle inequality holds and there are cases, when it does not. We show one example for both cases. Let Φ, Φ' and Φ'' be monotone sequences with pairwise disjoint alphabets. Then the strong and weak distances exists for all pairs of monotone sequences and are equal to 1. Therefore $2 = D_S(\Phi, \Phi') + D_S(\Phi, \Phi'') \geq D_S(\Phi'', \Phi') = 1$ and $2 = D_W(\Phi, \Phi') + D_W(\Phi, \Phi'') \geq D_W(\Phi'', \Phi') = 1$.

On the other hand let $\Phi^{IV} = \{c \cup a^n b^n | n < i\}_{i=1}^{\infty}$, $\Phi^V = \{c \cup x^n | n < i\}_{i=1}^{\infty}$ and $\Phi^{VI} = \{c\}_{i=1}^{\infty}$. Then $D_S(\Phi^{IV}, \Phi^{VI}) = D_S(\Phi^V, \Phi^{VI}) = D_W(\Phi^{IV}, \Phi^{VI}) = D_W(\Phi^V, \Phi^{VI}) = 0$ and $D_S(\Phi^{IV}, \Phi^V) = D_W(\Phi^{IV}, \Phi^V) = 1$. So $0 = D_S(\Phi^{IV}, \Phi^{VI}) + D_S(\Phi^V, \Phi^{VI}) \ngeq D_S(\Phi^{IV}, \Phi^V) = 1$ and $0 = D_W(\Phi^{IV}, \Phi^{VI}) + D_W(\Phi^V, \Phi^{VI}) \ngeq D_W(\Phi^{IV}, \Phi^V) = 1$. □

Lemma 4.3.7. *Let Φ and Φ' be equivalent monotone sequences ($\Phi \approx \Phi'$) and let Φ'' be another monotone sequence. If $D_S(\Phi, \Phi'')$ exists, then $D_S(\Phi', \Phi'')$ exists as well and $D_S(\Phi', \Phi) = D_S(\Phi, \Phi'')$.*

Proof. From the assumption and from Theorem 4.3.5 we know, that for all ε there is such n_0, for which the difference between the element of the sequence $D_i^{(S)}$ is less then epsilon for Φ and Φ''. From the definition of the equivalence we know, that there is point n_0', from where sequences Φ and Φ' are the same. It is satisfactory to choose the bigger of numbers n_0 and n_0' for all ε and we obtain, that $D_S(\Phi', \Phi'')$ always exists. Moreover the new sequence of numbers is convergent to the same distance as $D_S(\Phi, \Phi'')$. □

From the fact, that $D_S(\Phi, \Phi) = 0$ for every monotone sequences and from the lemma above it holds:

Corollary 4.3.8. *Let Φ and Φ' be equivalent monotone sequences ($\Phi \approx \Phi'$). $D_S(\Phi, \Phi')$ always exists and*

$$D_S(\Phi, \Phi') = 0.$$

4.4 Special Languages for Strong and Weak Distance

In this section we show that the strong and the weak distance measures may behave differently on the same pair monotone sequences and that sequences of approximations for finite languages are trivial in terms of the strong and weak distance measure.

Lemma 4.4.1. *For all sequences of approximations approximating finite languages exists such n_0 in $\mathbb{N}$, that for all $n > n_0 : L_n = L_\Phi$.*

Proof. Without loss of generality let $k := |L_\Phi|$, so the finite language approximated consists of k words $\{w_1, w_2, ..., w_k\}$.

For a given word w_j let r_j be the smallest index such that w_j in L_{r_j}. Clearly we can create a finite set of numbers $\{r_1, r_2, ..., r_k\}$ containing the smallest indexes for every word.

Let r_{max} be a maximal value of set $\{r_1, r_2, ..., r_k\}$, so for every $j \geq r_{max}$ $L_j = L_\Phi$. □

Theorem 4.4.2. *For all finite language L, for all sequences of approximations Φ, Φ' for L $D_W(\Phi, \Phi')$ and $D_S(\Phi, \Phi')$ exist.*

Proof. From Lemma 4.4.1 there are n_0 and n_0', from where all the sets of the sequences Φ and Φ' are the same. Without loss of generality $n_0 > n_0'$. So the following statement holds: for all $n > n_0$: $L_n = L_n' = L$. Therefore for all $n > n_0, D_n^{(W)} = D_{n-1}^{(W)}$ and $D_n^{(S)} = D_{n-1}^{(S)}$, because no word has been added to the approximations. So the condition for the convergence $|D_{n+p}^{(W)} - D_n^{(W)}| = 0 < \varepsilon$ and $|D_{n+p}^{(S)} - D_n^{(S)}| = 0 < \varepsilon$ holds. □

Lemma 4.4.3. *Let Φ and Φ' be sequence of approximations for finite languages L and L' respectively. Then $D_S(\Phi, \Phi')$ exists and equals to the average of the values of the minimal weight maximal matching on the two finite language L and L'.*

Proof. We prove, that $\{D_i^{(S)}\}_{i=1}^{\infty}$ is convergent. From Lemma 4.4.1 we know that for both sequences of approximations there is a point in the approximation where the whole language is in the finite approximation. Let this index be n_0 and n_0' for Φ and Φ' respectively. Without loss of generality let $n_0 \geq n_0'$.

So from n_0 every $D_i^{(S)}$ holds the average value of the edges in the minimal weight maximal matching between the two finite languages L_Φ and $L_{\Phi'}$. This sequence is convergent (is constant), and the limit of such sequence is the same value as every $D_i^{(S)}$ from n_0. Therefore $\{D_i^{(S)}\}_{i=1}^{\infty}$ is convergent. □

Corollary 4.4.4. *When the two sequences of approximations are approximating the same finite language (so $L_\Phi = L_{\Phi'}$) we can additionally say, that $D_S(\Phi, \Phi') = 0$, because the minimal weight maximal matching in this case is the identity matching, so all edges from the matching have value 0, therefore the average is 0 as well.*

Lemma 4.4.5. *Let Φ be a sequence of approximation approximating a finite language and Φ' approximating an infinite one. In this case $D_S(\Phi, \Phi')$ always exists.*

Proof. Similarly to the lemma above we have to prove, that $\{D_i^{(S)}\}_{i=1}^{\infty}$ is convergent. From Lemma 4.4.1 we know, that there is a point n_0, from which $L_n = L_\Phi$ for every $n > n_0$. Now we can find such $k_0 > n_0$, from where $|L_k'| > |L_\Phi|$ for every $k > k_0$. Such a k_0 has to exist, because Φ' is approximating an infinite language.

Now we prove, that from k_0 the sequence $\{D_i^{(S)}\}_{i=1}^{\infty}$ is monotone and bounded. This sequence is trivially bounded, because we are working on the real closed interval $[0, 1]$. For every $k > k_0$ the whole L_k is matched with one word from L'_k. $L'_k \subseteq L'_{k+1}$ so the words, which were used in the matching between L_k and L'_k are still in L'_{k+1} as well, thus the minimal weight maximal matching between $L_{k+1} = L_k$ and L'_{k+1} has to be at least that good as a minimal weight maximal matching between L_k and L'_k (This matching is still maximal and so the minimal weight maximal matching should have no more weight as this matching). Every maximal matching from k_0 on have the same cardinality, so $D_k^{(S)} \geq D_{k+1}^{(S)}$.

Every monotone and bounded sequence is convergent, so $\{D_i^{(S)}\}_{i=1}^{\infty}$ is convergent and $D_S(\Phi, \Phi')$ exists. □

For the weak distance measure the properties above do not have to hold:

Lemma 4.4.6. *There exist a pair of sequences of approximations Φ and Φ' for the same finite language, such as $L_\Phi = L_{\Phi'} = L$ and $D_W(\Phi, \Phi') = 1$.*

Proof. $L = \{a, b\}$, $\Phi = \{\{a\}, \{a, b\}\}$, $\Phi' = \{\{b\}, \{a, b\}\}$. So $D_1^{(W)} = 1$, because words a and b have no common subsequence. In the next step b is matched to a, because these are words, which were not used in a step 1. So the matching will be $a - b, b - a$ in the step 2. $D_2^{(W)} = 1$. From now $D_i^{(W)} = D_{i-1}^{(W)}$. So $D_W = 1$ too. □

Lemma 4.4.7. *There exist a pair of sequences of approximations Φ and Phi' for the same infinite language, such as $L_\Phi = L_{\Phi'} = L$ and $D_W(\Phi, \Phi') = 1$.*

Proof. The proof is almost identical with the one above. $L = \{a^n | n > 0\} \cup \{b^n | n > 0\}$. Φ and Φ' are sequences of approximations, where one word is added in each step. Words a^l and b^l is added into Φ and Φ' respectively, when $k = 2l$. In the opposite case b^l and a^l is added into Φ and Φ' respectively, when $k = 2l - 1$. So in each step one new edge is added to the maximal matching with a value 1. □

In this way we can even force regular languages to have significantly larger weak distance than strong distance. So it would be better to have some restrictions for these monotone sequences. Therefore we shall consider only those monotone sequences, which are generated by grammars with energies (or sometimes by the length-based approximation). These monotone sequences are examined starting in Chapter 6.

Chapter 5

Advanced Properties of Strong and Weak Distance

In this chapter we show how the strong and the weak distance measures behave, when we can obtain additional information about the monotone sequences being measured. In this chapter we show how the strong and weak distances behave under some of the operations defined for monotone sequences. Namely we consider the disjoint union for both distances, the finite deletion for the strong distance and finite suffixes and prefixes for the strong distance measure.

5.1 Strong Distance and Disjoint Union

In this section we show how the disjoint union behaves with the strong distance measure.

At first let us introduce some notation and assumptions needed in this section.

i) Let $\Phi, \Phi', \Phi'', \textit{and } \Phi'''$ be sequences of approximations approximating infinite languages.

ii) Alphabets of Φ and Φ' are disjoint with alphabets of Φ'' and Φ'''.

iii) Let $\{M_i^1\}_{i=1}^{\infty}$ and $\{M_i^2\}_{i=1}^{\infty}$ denote the matchings from the definition of the strong distance measure between L_i and L_i' and between L_i'' and L_i''' respectively. Let $\{M_i^3\}_{i=1}^{\infty}$ denote one of the *maximal cardinality matchings on unmatched vertices.*

iv) $D_S(\Phi, \Phi')$ and $D_S(\Phi'', \Phi''')$ exist and we will refer to them as *original distances* in this section.

v) Let $\{p_i\}_{i=1}^{\infty}$, $\{q_i\}_{i=1}^{\infty}$ and $\{r_i\}_{i=1}^{\infty}$ denote the sequences of the cardinalities of M_i^1, M_i^2 and M_i^3 respectively. So $p_n := |M_n^1|$, $q_n := |M_n^2|$ and $r_n := |M_n^3|$. The sequence

of the proportions of these cardinalities is convergent to a non-zero number. So $\{\frac{p_n}{q_n}\}$ is convergent to $k_1 > 0$ and $\{\frac{p_n}{r_n}\}$ is convergent to $k_2 > 0$. This condition is equivalent to $\{\frac{q_n}{p_n}\}$ is convergent to $\frac{1}{k_1} > 0$, $\{\frac{r_n}{p_n}\}$ is convergent to $\frac{1}{k_2} > 0$, $\{\frac{q_n}{r_n}\}$ is convergent to $\frac{k_2}{k_1} > 0$ and $\{\frac{r_n}{q_n}\}$ is convergent to $\frac{k_1}{k_2} > 0$.

Under these conditions we try to express $D_S(\Phi \uplus \Phi'', \Phi' \uplus \Phi''')$ using $D_S(\Phi, \Phi')$ and $D_S(\Phi'', \Phi''')$ as well as all k_i.

Now let us describe, why are these assumptions needed. The first assumption says that all the sequences of approximations are approximating infinite languages. This condition is not a real restriction, because those sequences of approximations, which are approximating finite languages are trivial in terms of the strong distance measure. So when all sequences of approximations are approximating finite languages the disjoint union of these sequences will approximate finite languages as well. On the other hand when at least one of the sequences of approximations is approximating infinite languages these sequences of approximations will dominate the other sequences approximating finite languages. Therefore the only considerable non-trivial setup is the one included in the first condition.

The second condition is enforcing the existence of the disjoint union between the pairs of monotone sequences.

The third and the fourth assumption are related. The fourth assumption is enforcing the existence of the strong distance measure between the corresponding pairs of monotone sequences.

The fifth assumption is introducing a lot of notations, however this assumption is a strict restriction, which enforces that the matchings between the pairs are growing more or less at the same rate. Whenever this condition does not hold we can assume that one of the distances ($D_S(\Phi, \Phi')$ or $D_S(\Phi'', \Phi''')$) dominates the other.

Our intention is to prove, that the strong distance measure between $\Phi \uplus \Phi''$ and $\Phi' \uplus \Phi'''$ is a weighted average of the distances between the pairs of the monotone sequences. Before we actually prove this we have to prove one lemma first which is describing the minimal weight maximal matching between $\Phi \uplus \Phi''$ and $\Phi' \uplus \Phi'''$.

Lemma 5.1.1. *Let us consider monotone sequences, which are satisfying assumptions i)-iii) above. One matching corresponding to the strong distance measures between* $L_i \cup L_i''$ *and* $L_i' \cup L_i'''$ *is* $M_i^1 \cup M_i^2 \cup M_i^3$.

Proof. We have to prove, that $M_i^1 \cup M_i^2 \cup M_i^3$ is a minimal weight maximal matching. Moreover we prove, that $M_i^1 \cup M_i^2 \cup M_i^3$ is a minimal weight maximal matching with a minimal number of edges between L_i and L_i''' and vica versa between L_i'' and L_i', let us call these edges *cross side* edges and each cross side edge by the definition holds the value 1.

The maximality of this matching is trivial (from the definition of matching M_i^3).

The minimal number of cross side edges is the minimum of $||L_i''| - |L_i||$, $||L_i'''| - |L_i'||$ or 0, because all these edges have to be covered by the maximal matching. We can see, that $M_i^1 \cup M_i^2 \cup M_i^3$ is one of those matchings, which has the minimal possible number of cross side edges. Let us prove first, that among those maximal matchings, which has the minimal possible number of cross side edges matching $M_i^1 \cup M_i^2 \cup M_i^3$ has the minimal weight.

By contradiction, let us assume, that there is another maximal matching M, with the same number of cross side edges and less weight. We have two possibilities: All unmatched vertices in the previous minimal weight maximal matchings are on the same partition after the disjoint union, thus no cross side edge is needed. In this case $M_i^1 \cup M_i^2 \cup M_i^3$ is trivially a minimal weight maximal matching, following from Lemma 4.1.3.

The second option is that the previously unmatched vertices are in the different partitions after the disjoint union, thus some cross side edges are needed. The overall weight of matching M is the sum of the weight of all cross side edges, one maximal matching on L_i and L_i' (without the vertices used by the cross side edges) and another maximal matching on L_i'' and L_i''' (without the vertices used by the cross side edges). In this case we know, that the choice of the edges used by the cross side edges does not effect the value of this edge, but still effects the maximal matchings between L_i'' and L_i''' and L_i and L_i'. Thus M can not have less weight then matching $M_i^1 \cup M_i^2 \cup M_i^3$.

Let us prove, then $M_i^1 \cup M_i^2 \cup M_i^3$ is a minimal weight maximal matching in general.

By contradiction, let us assume, that $M_i^1 \cup M_i^2 \cup M_i^3$ is not a minimal weight maximal matching. Let us call M one of the minimal weight maximal matchings[1], which have minimal number of cross side edges.[2]

From the paragraph above we know, that M has more cross side edges then needed. In this case we can do the following: Picking two cross side edges with a opposite direction[3]. We can eliminate both cross side edges by rematching these four edges. In this case either the matching will have less weight overall, which is a contradiction, or there will be a matching with the same weight as M with less cross side edges used, which is another contradiction. Therefore there are no two cross side edge with an opposite direction in matching M, but this is in a contradiction with the maximality of the matching M.

[1] M is not related to the matching used in the paragraph above

[2] There can be more then one minimal weight maximal matching and more then one with a minimal number of cross side edges.

[3] For an edge between L_i and L_i''' an edge with the opposite direction is an edge between two vertices from L_i' and L_i'' and vica versa.

Therefore $M_i^1 \cup M_i^2 \cup M_i^3$ is one of the minimal weight maximal matching on $L_i \cup L_i''$ and $L_i' \cup L_i'''$. □

Now we can prove the main theorem in this section:

Theorem 5.1.2. *Let us consider monotone sequences, which are satisfying assumptions i)-v) above. Then the disjoint union of these monotone sequences has a strong distance too, so* $D_S(\Phi \uplus \Phi'', \Phi' \uplus \Phi''')$ *exists and*

$$D_S(\Phi \uplus \Phi'', \Phi' \uplus \Phi''') = \frac{k_1 k_2 D_S(\Phi, \Phi') + k_2 D_S(\Phi'', \Phi''') + k_1}{k_1 k_2 + k_1 + k_2}.$$

Proof. We have to show, that the matching $M_i^1 \cup M_i^2 \cup M_i^3$ is convergent. From Theorem 4.3.5 it follows that the sequence of average values from the matching is convergent iff

$$\forall \varepsilon > 0 \exists n_0 \in \mathbb{N}, n_0 > 1, \forall n \in \mathbb{N}, n > n_0, \forall a \in \mathbb{N} : |D_{n+a}^{(S)} - D_n^{(S)}| < \varepsilon.$$

Now we have to express $D_n^{(S)}$ using matchings M_n^1, M_n^2 and M_i^3 (where $D_n^{(S)1}$ and $D_n^{(S)2}$ are the average values of matchings M_n^1 and M_n^2 respectively on the nth level). These two sequences are satisfying Theorem 4.3.5, because distances between the corresponding pairs of the sequences of approximations exist. So

$$\forall \varepsilon > 0, \exists n_0 \in \mathbb{N}, n_0 > 1, \forall n \in \mathbb{N}, n > n_0, \forall a \in \mathbb{N} : |D_{n+a}^{(S)1} - D_n^{(S)1}| < \varepsilon;$$

$$\forall \varepsilon > 0, \exists n_0 \in \mathbb{N}, n_0 > 1, \forall n \in \mathbb{N}, n > n_0, \forall a \in \mathbb{N} : |D_{n+a}^{(S)2} - D_n^{(S)2}| < \varepsilon.$$

From Lemma 5.1.1 $D_n^{(S)}$ can be expressed by $D_n^{(S)1}$ and $D_n^{(S)2}$ as follows: $D_n^{(S)} = \frac{p_n D_n^{(S)1} + q_n D_n^{(S)2} + r_n}{p_n + q_n + r_n}$. [4] We have to bound $|D_{n+a}^{(S)} - D_n^{(S)}|$ by ε and find such n_0, that for every n from it $|D_{n+a}^{(S)} - D_n^{(S)}| < \varepsilon$.

[4] Note, that all edges in M_i^3 has value 1.

$$\left|D^{(S)}_{n+a} - D^{(S)}_{n}\right| = \left|\frac{p_{n+a}D^{(S)1}_{n+a} + q_{n+a}D^{(S)2}_{n+a} + r_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} - \frac{p_nD^{(S)1}_n + q_nD^{(S)2}_n + r_n}{p_n+q_n+r_n}\right| =$$

$$= \left|\frac{p_{n+a}D^{(S)1}_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} + \frac{q_{n+a}D^{(S)2}_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} + \frac{r_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} + \right.$$

$$\left. - \frac{p_nD^{(S)1}_n}{p_n+q_n+r_n} - \frac{q_nD^{(S)2}_n}{p_n+q_n+r_n} - \frac{r_n}{p_n+q_n+r_n}\right| =$$

$$= \left|\frac{p_{n+a}D^{(S)1}_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} - \frac{p_nD^{(S)1}_n}{p_n+q_n+r_n} + \frac{q_{n+a}D^{(S)2}_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} + \right.$$

$$\left. - \frac{q_nD^{(S)2}_n}{p_n+q_n+r_n} + \frac{r_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} - \frac{r_n}{p_n+q_n+r_n}\right| \leq$$

$$\leq \left|\frac{p_{n+a}D^{(S)1}_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} - \frac{p_nD^{(S)1}_n}{p_n+q_n+r_n}\right| + \left|\frac{q_{n+a}D^{(S)2}_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} + \right.$$

$$\left. - \frac{q_nD^{(S)2}_n}{p_n+q_n+r_n}\right| + \left|\frac{r_{n+a}}{p_{n+a}+q_{n+a}+r_{n+a}} - \frac{r_n}{p_n+q_n+r_n}\right| =$$

$$= \left|\frac{D^{(S)1}_{n+a}}{1+\frac{q_{n+a}}{p_{n+a}}+\frac{r_{n+a}}{p_{n+a}}} - \frac{D^{(S)1}_n}{1+\frac{q_n}{p_n}+\frac{r_n}{p_n}}\right| + \left|\frac{D^{(S)2}_{n+a}}{\frac{p_{n+a}}{q_{n+a}}+1+\frac{r_{n+a}}{q_{n+a}}} + \right.$$

$$\left. - \frac{D^{(S)2}_n}{\frac{p_n}{q_n}+1+\frac{r_n}{q_n}}\right| + \left|\frac{1}{\frac{p_{n+a}}{r_{n+a}}+\frac{q_{n+a}}{r_{n+a}}+1} - \frac{1}{\frac{p_n}{r_n}+\frac{q_n}{r_n}+1}\right|$$

We can generalize the three subexpressions as follows: $\left|\frac{D_{n+a}}{1+c_{n+a}+c'_{n+a}} - \frac{D_n}{1+c_n+c'_n}\right|$, where sequence D is staying for some sequence of numbers, which are convergent and from the closed interval $[0,1]$, sequences c and c' are fractions which are convergent from condition v).

$$\left|\frac{D_{n+a}}{1+c_{n+a}+c'_{n+a}} - \frac{D_n}{1+c_n+c'_n}\right| =$$

$$= \frac{1}{1+c_{n+a}+c'_{n+a}}\left|D_{n+a}+c_nD_{n+a}+c'_nD_{n+a}-D_n-c_{n+a}D_n-c'_{n+a}D_n\right| \leq$$

$$\leq \left|D_{n+a}+c_nD_{n+a}+c'_nD_{n+a}-D_n-c_{n+a}D_n-c'_{n+a}D_n\right| =$$

$$= \left|D_{n+a}-D_n+c_n(D_{n+a}-D_n)+c'_n(D_{n+a}-D_n)-(c_{n+a}-c_n)D_n-(c'_{n+a}-c'_n)D_n\right| \leq$$

$$\leq |(D_{n+a}-D_n)(1+c_n+c'_n)| + |-(c_{n+a}-c_n)D_n| + \left|-(c'_{n-a}-c'_n)D_n\right| =$$

$$= (1+c_n+c'_n)\left|D_{n+a}-D_n\right| + D_n\left|c_{n+a}-c_n\right| + D_n\left|c'_{n+a}-c'_n\right|$$

In the last expression all three sequences used (eve thou are renamed) are convergent and the coefficients $(1+c_n+c'_n)$ and D_n are constants, which can be bounded as well. So

$\left|\frac{D_{n+a}}{1+c_{n+a}+c'_{n+a}} - \frac{D_n}{1+c_n+c'_n}\right|$ is convergent for every three setup from the original expression, so the original expression is bounded by some ε.

Now we can evaluate $D_S(\Phi \uplus \Phi'', \Phi' \uplus \Phi''')$ as follows:

$$D_S(\Phi \uplus \Phi'', \Phi' \uplus \Phi''') = \lim_{i\to\infty} D_i^{(S)} = \lim_{i\to\infty} \frac{p_i D_i^{(S)1} + q_i D_i^{(S)2} + r_i}{p_i + q_i + r_i}$$

With growing i the limits of all used sequences are becoming more and more accurate, thus

$$D_S(\Phi \uplus \Phi'', \Phi' \uplus \Phi''') = \lim_{i\to\infty} \frac{p_i D_S(\Phi, \Phi') + \frac{p_i}{k_1} D_S(\Phi'', \Phi''') + \frac{p_i}{k_2}}{p_i + \frac{p_i}{k_1} + \frac{p_i}{k_2}} =$$
$$= \frac{k_1 k_2 D_S(\Phi, \Phi') + k_2 D_S(\Phi'', \Phi''') + k_1}{k_1 k_2 + k_1 + k_2}.$$

□

First we can point out, that if one of the sequences of the fractions is divergent, then the other sequence is divergent as well. In this case there is only one sequence, which is convergent to a non-zero value. If one sequence of average values $(D_i^{(S)})$ dominates the others the distance will be equal to the distance corresponding to this dominating sequence. If two sequences of average values dominate the third one, then the distance will be the weighted average of these two distances.
We can specially point out, that if there is only a relatively small number of edges on unmatched vertices, then these vertices are not modifying the overall distance between the merged sequences of approximations.

We have to mention that our conditions are symmetrical, so our result should be symmetrical as well. And this is the case even though it is not easy to see for the first time. The weighted average in 5.1.2 is symmetrical, because the values of k_i are derived from the particular setup of the original distances. We have to add that $k_1 k_2$ is only in the formula because we had not introduced a new symbol for $\frac{k_1}{k_2}$, which is the limit of $\frac{r_n}{q_n}$.

At last we can say something about the union operation in general by omitting the ii) condition. If the distance between $\Phi \cup \Phi''$ and $\Phi' \cup \Phi'''$ exists then it is less or equal to $\frac{k_1 k_2 D_S(\Phi,\Phi') + k_2 D_S(\Phi'',\Phi''') + k_1}{k_1 k_2 + k_1 + k_2}$, so the distance derived in Theorem 5.1.2 is an upper bound for the distance measure, if exists.

In this and the following section we do not examine the situation, when the empty word is a part of all monotone sequences. In the following section we show, that we can delete some words from the monotone sequences without changing the distance between them. So we can delete the empty word from all monotone sequences, then apply the theorem above and then put the empty word back.

5.2 Weak Distance and Disjoint Union

In this section we show how does the disjoint union behaves under the weak distance measure. The idea of the proof will remain, but we have to consider, how the matching in the weak distance measure looks like. We have one problem with the matching introduced in the chapter above, namely that matching M^3 will use up some vertices, which can be matched later in the original matchings. In this case our matching can not be considered as a matching in the definition of the weak distance measure. On the other hand we can not omit M^3 in general, because without M^3 we can not build a maximal matching after the union is done.

The solution is to consider those monotone sequences, for which the matching M^3 is empty, thus all unmatched vertices on the original matchings are on the same side.

Now let us introduce the set of assumptions needed in this section. These assumptions are similar to the ones made in the previous section.

i) Let $\Phi, \Phi', \Phi'', \textit{and } \Phi'''$ be sequences of approximations approximating infinite languages.

ii) Alphabets of Φ and Φ' are disjoint with alphabets of Φ'' and Φ'''.

iii) Let $\{M_i^1\}_{i=1}^{\infty}$ and $\{M_i^2\}_{i=1}^{\infty}$ denote the matchings from the definitions of the strong distance measure between L_i and L_i' and between L_i'' and L_i''' respectively.

iv) $D_W(\Phi, \Phi')$ and $D_W(\Phi'', \Phi''')$ exist, and we will refer to them as *original distances.*

v) Let $\{p_i\}_{i=1}^{\infty}$ and $\{q_i\}_{i=1}^{\infty}$ denote the sequences of the cardinalities of M_i^1 and M_i^2 respectively. So $p_n := |M_n^1|$ and $q_n := |M_n^2|$. The sequence of the proportion of these two cardinalities is convergent to a non-zero number. So $\{\frac{p_n}{q_n}\}$ is convergent to $k_1 > 0$. This condition is equivalent to $\{\frac{q_n}{p_n}\}$ is convergent to $\frac{1}{k_1} > 0$.

Lemma 5.2.1. *Let us consider monotone sequences, which are satisfying conditions i-iii above. If $|L_i| \geq |L_i'|$ and $|L_i''| \geq |L_i'''|$ for each i, then the matching corresponding to the distance measures between $L_i \cup L_i''$ and $L_i' \cup L_i'''$ is $M_i^1 \cup M_i^2$.*

Proof. This lemma follows from Lemma 5.1.1 in the following setup: M^3 is empty and we are considering only the vertices, which were added in the last step, so we are searching for the minimal weight maximal matching between $(L_i \cup L_i'') \setminus (X_{i-1} \cup X_{i-1}'')$ and $(L_i' \cup L_i''') \setminus (X_{i-1}' \cup X_{i-1}''')$, where X_i denotes the vertices matched in step i. □

Theorem 5.2.2. *Let us consider monotone sequences, which are satisfying conditions i-v above. If $|L_i| \geq |L_i'|$ and $|L_i''| \geq |L_i'''|$ for every possible i, then the disjoint union of*

these monotone sequences has a distance two, so $D_W(\Phi \uplus \Phi'', \Phi' \uplus \Phi''')$ *exists and*

$$D_W(\Phi \uplus \Phi'', \Phi' \uplus \Phi''') = \frac{k_1 D_W(\Phi, \Phi') + D_W(\Phi'', \Phi''')}{1 + k_1}.$$

Proof. This proof is almost identical to the proof of Theorem 5.1.2. We have to show, that

$$\forall \varepsilon > 0 \exists n_0 \in \mathbb{N}, n_0 > 1, \forall n \in \mathbb{N}, n > n_0, \forall a \in \mathbb{N} : |D_{n+a}^{(W)} - D_n^{(W)}| < \varepsilon.$$

From Lemma 5.2.1 $D_n^{(W)}$ can be expressed by $D_n^{(W)1}$ and $D_n^{(W)2}$, where $D_n^{(W)1}$ and $D_n^{(W)2}$ stands for the average value of matchings M^1 and M^2 on the nth step respectively, as follows: $D_n^{(W)} = \frac{p_n D_n^{(W)1} + q_n D_n^{(W)2}}{p_n + q_n}$.

After a similar derivation to the one used in the proof of Theorem 5.1.2 we obtain:

$$\left| D_{n+a}^{(W)} - D_n^{(W)} \right| \leq \left| \frac{D_{n+a}^{(W)1}}{1 + \frac{q_{n+a}}{p_{n+a}}} - \frac{D_n^{(W)1}}{1 + \frac{q_n}{p_n}} \right| + \left| \frac{D_{n+a}^{(W)2}}{1 + \frac{p_{n+a}}{q_{n+a}}} - \frac{D_n^{(W)2}}{1 + \frac{p_n}{q_n}} \right|$$

Both expressions are in a form $\left| \frac{D_{n+a}}{1+c_{n+a}} - \frac{D_n}{1+c_n} \right|$ and can be bounded:

$$\left| \frac{D_{n+a}}{1 + c_{n+a}} - \frac{D_n}{1 + c_n} \right| \leq (1 + c_n) \left| D_{n+a} - D_n \right| + D_n \left| c_{n+a} - c_n \right|$$

Therefore $|D_{n+a}^{(W)} - D_n^{(W)}|$ is bounded as well, thus the weak distance between those two monotone sequences exists.

Moreover $D_W(\Phi \uplus \Phi'', \Phi' \uplus \Phi''') = \frac{k_1 D_W(\Phi, \Phi') + D_W(\Phi'', \Phi''')}{1 + k_1}$. □

5.3 Strong Distance and Finite Deletion

In this section we show, how does the finite deletion impact the strong distance measure. Similarly to the previous sections we consider only those sequences of approximations, which are approximating infinite languages. This is not a real restriction, because we know, that those sequences of approximations, which are approximating finite languages are trivial in terms of the strong distance measure.

Now consider a pair of sequences of approximations Φ and Φ', which are approximating infinite languages with an existing strong distance; let S and S' be finite sets of words from languages L_Φ and $L_{\Phi'}$ respectively. Our intention is to derive $D_S(\Phi - S, \Phi' - S')$ using $D_S(\Phi, \Phi')$.

The idea behind our derivation is, that a finite operation should not change a distance between two infinite objects, thus $D_S(\Phi - S, \Phi' - S') = D_S(\Phi, \Phi')$. This is equivalent to $D_S(\Phi - S, \Phi') = D_S(\Phi, \Phi')$, because from this equality and from the symmetry we can get $D_S(\Phi - S, \Phi' - S') = D_S(\Phi, \Phi' - S') = D_S(\Phi' - S', \Phi) = D_S(\Phi', \Phi) = D_S(\Phi, \Phi')$.

S is a finite set, thus we can sort all words from S into a sequence $w_1, w_2, ...w_n$. By proving that $D_S(\Phi - \{w\}, \Phi') = D_S(\Phi, \Phi')$ for every word w from L_Φ we prove $D_S(\Phi - S, \Phi') = D_S(\Phi, \Phi')$ as well, because we can delete the words w_i one by one from Φ without changing the original distance.

Lemma 5.3.1. *Let Φ and Φ' be two sequences of approximations approximating infinite languages and let w be a word w in L_Φ. If $D_S(\Phi, \Phi')$ exists then*

$$D_S(\Phi - \{w\}, \Phi') = D_S(\Phi, \Phi').$$

Proof. In this proof our intention is to show, that at every level i the average value of the minimal weight maximal matching is the same between L_i and L'_i and between $L_i - \{w\}$ and L'_i. Let us call these matching M_i and M'_i respectively.

Let s_i be a sum of all edges in a minimal weight maximal matching between L_i and L'_i. Similarly let t_i be a sum of all edges in a minimal weight maximal matching between $L_i - \{w\}$ and L'_i. Let m_i be the cardinality of the first matching[5]. From the conditions we know, that $\frac{s_i}{m_i} = D_i^{(S)}$ is convergent.

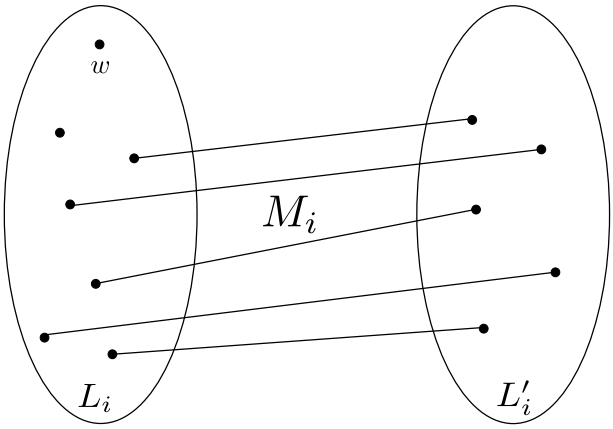

Figure 5.1: When w is not included in M_i the maximal matching remains the same.

At first, we do not have to consider those sets, where w is not from L_i, because we are not altering the original sets nor the minimal weight maximal matching.

Our aim is to bound t_i using s_i. For this we examine all possibilities, how the word w can be deleted from L_i. At first consider, that w is not matched by M_i. In this case $M_i = M'_i$, thus $t_i = s_i$.

We have two more cases, w is matched by M_i in both of them. The first case is when we are deleting from the partition, which is fully matched by M_i. Let e in M_i be the edge, with a value $v(e)$ which is incident with word w. $M_i - \{e\}$ is a maximal matching between $L_i - \{w\}$ and L'_i with a sum of weights $s_i - v(e)$. Because of the minimality of matching M'_i: $t_i \leq s_i - v(e) \leq s_i$.

[5]The cardinality of the second matching is almost the same.

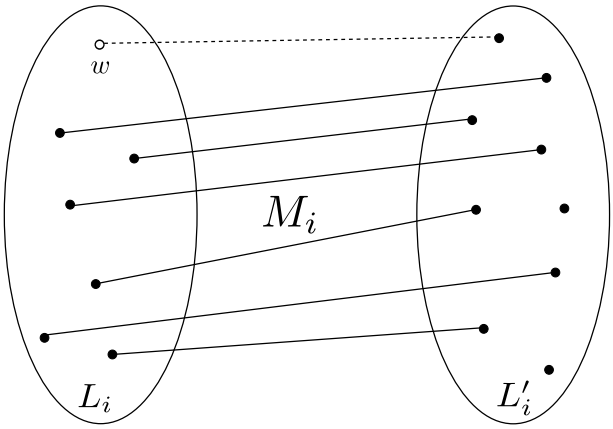

Figure 5.2: After excluding w from L_i every vertex in L_i remains matched, thus the new matching is maximal.

The second case is when w is deleted from the partition, where there are some unmatched vertices by M_i. Let one of them be w'. Let w_1 be the word, which is matched with w in M_i, let e be the edge between them with a value $v(e)$. Let e' be the edge between w' and w_1 with value $v(e')$. In this case matching $M_i - \{e\} \cup \{e'\}$ is a maximal matching between $L_i - \{w\}$ and L'_i with a sum of weights $s_i - v(e) + v(e')$. From the minimality of M'_i it follows $t_i \leq s_i - v(e) + v(e') \leq s_i + 1$.

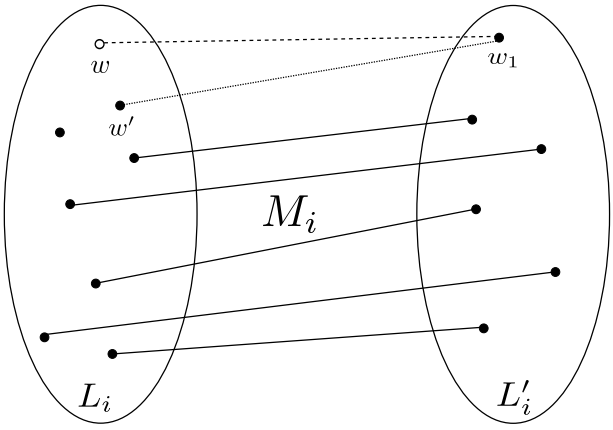

Figure 5.3: After excluding w from L_i there is one more possible edge, which should be added to create a maximal matching.

So in general $t_i \leq s_i + 1$ in all cases.

Now let us consider the opposite situation. We have the minimal weight maximal matching between $L_i - \{w\}$ and L'_i and we have a word w, which should be inserted into L_i. We have two cases. First case is, that L'_i is fully matched by M'_i, thus there is no vertex in L'_i which can be matched with w. In this case M'_i is a maximal matching between L_i and L'_i and from the minimality of M_i follows: $s_i \leq t_i$.

The second case is when there is a word in L'_i, which is unmatched by M'_i. Let w_1

be one of these words. Let e be the edge between w and w_1 with value $v(e)$. Then $M_i' \cup \{e\}$ is a maximal matching between L_i and L_i' and from the minimality of M_i it follows: $s_i \leq t_i + v(e) \leq t_i + 1$.

So in all cases $s_i \leq t_i + 1$.

So in all cases it holds: $s_i - 1 \leq t_i \leq s_i + 1$ for every i.[6] We can now divide this inequality by m_i. Note, that the sequence $\{m_i\}_{i=1}^{\infty}$ is a monotone growing sequence, so the sequence $\{\frac{1}{m_i}\}_{i=1}^{\infty}$ has limit 0.

$$\frac{s_i}{m_i} - \frac{1}{m_i} \leq \frac{t_i}{m_i} \leq \frac{s_i}{m_i} + \frac{1}{m_i}$$

$\frac{t_i}{m_i}$ is convergent to $D_S(\Phi - \{w\}, \Phi')$ as well, because matching M_i' has the same number of edges or one less edges then M_i, which does not effect the convergence. In a limit $\frac{t_i}{m_i}$ is bounded and has the same upper and lower bound, thus it is converging to this number. This number is the limit of $\frac{s_i}{m_i}$, which is $D_S(L_\Phi, L_{\Phi'})$. □

Now we introduce the main theorem of this section, without proof, because it directly follows from Lemma 5.3.1 as it is stated in the preface of this section:

Theorem 5.3.2. *Let Φ and Φ' be two sequences of approximations approximating infinite languages and let $S \subset L_\Phi$ and $S' \subset L_{\Phi'}$ be finite sets of words. If $D_S(\Phi, \Phi')$ exists then*

$$D_S(\Phi - S, \Phi' - S') = D_S(\Phi, \Phi').$$

It is not exactly shown, but this lemma and thus the theorem works in the opposite way as well. When $D_S(\Phi - S, \Phi' - S')$ exists $D_S(\Phi, \Phi')$ as well and they are equal.[7]

5.4 Strong Distance and Finite Suffix and Prefix

In this section we derive, how prefixes and suffixes affect the strong distance measure. We prove the theorem for prefixes, but the same theorem holds for suffixes as well. In this section (similarly to the sections above we consider only the setup, where both sequences of approximations are approximating infinite languages.)

We proceed step-by-step, first proving that the distance measure does not change when using any prefix by length 1. Then we can add prefix of any length to the given monotone sequences by adding the characters from the prefix one by one into the monotone sequences.

[6] We only assumed $w \in L_i$.

[7] In the lemma we bound the sequence of numbers corresponding to one distance with the sequence of numbers corresponding to the other distance and this bound is symmetric.

Now we prove, that when $D_S(\Phi, \Phi')$ exists for any two monotone sequences Φ and Φ', $D_S(a\Phi, b\Phi')$ exists as well for any character a and b, moreover $D_S(a\Phi, b\Phi') = D_S(\Phi, \Phi')$. Note that we are adding one character into both monotone sequences as prefix, thus we can add only prefixes with the same length in the general case.

First case, when a and b are different symbols.

Lemma 5.4.1. *Let Φ and Φ' be monotone sequences with an existing strong distance measure, and a and b two different symbols. Then $D_S(a\Phi, b\Phi')$ exists as well, and*

$$D_S(a\Phi, b\Phi') = D_S(\Phi, \Phi').$$

Proof. Let M_i be a minimal weight maximal matching between L_i and L'_i and let M'_i be a minimal weight maximal matching between aL_i and bL'_i.

The relative edit distance between two words (w and w') can be computed as $\frac{|w|+|w'|-2|LCS(w,w')|}{|w|+|w'|}$, where LCS is a longest common subsequence of the given words. For our purposes let $l := |w|$, $l' := |w'|$ and $c := |LCS(w, w')|$. Then the relative edit distance between w and w' is $\frac{l+l'-2c}{l+l'}$.

Using this notation we can calculate the sum of the edges in the matchings M_i and M'_i. Moreover, we can calculate the sum of the edges of the matching M_i between aL_i and bL'_i and M'_i between L_i and L'_i. The reason behind this is that there is a bijective correlation between the words from L_i and aL_i and between the words from L'_i and bL'_i. For the same reason matchings M_i and M'_i have the same cardinality.

$\sum_{(w,w')\in M_i} \frac{|w|+|w'|-2|LCS(w,w')|}{|w|+|w'|} = \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'}$ is the sum of weight of matching M_i between L_i and L'_i.
$\sum_{(w,w')\in M'_i} \frac{|w|+|w'|-2|LCS(w,w')|}{|w|+|w'|} = \sum_{(w,w')\in M'_i} \frac{l+l'-2c}{l+l'}$ is the sum of weight of matching M'_i between aL_i and bL'_i, therefore from the minimality of matching M_i

$$\sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \leq \sum_{(w,w')\in M'_i} \frac{l+l'-2c}{l+l'}. \tag{5.1}$$

$\sum_{(aw,bw')\in M_i} \frac{|aw|+|bw'|-2|LCS(aw,bw')|}{|aw|+|bw'|} = \sum_{(aw,bw')\in M_i} \frac{2+l+l'-2c}{2+l+l'}$ is the sum of weight of matching M_i between aL_i and bL'_i.
$\sum_{(aw,bw')\in M'_i} \frac{|aw|+|bw'|-2|LCS(aw,bw')|}{|aw|+|bw'|} = \sum_{(aw,bw')\in M'_i} \frac{2+l+l'-2c}{2+l+l'}$ is the sum of weight of matching M'_i between aL_i and bL'_i, from the minimality of the matching M'_i

$$\sum_{(aw,bw')\in M'_i} \frac{2+l+l'-2c}{2+l+l'} \leq \sum_{(aw,bw')\in M_i} \frac{2+l+l'-2c}{2+l+l'}. \tag{5.2}$$

Now we prove, that the relative edit distance between aw and bw' is bigger then

the relative edit distance between the same w and w'.

$$\frac{2+l+l'-2c}{2+l+l'} \geq \frac{l+l'-2c}{l+l'}$$
$$2(l+l')+(l+l')^2-2c(l+l') \geq (l+l')(2+l+l')-2c(2+l+l')$$
$$-l-l' \geq -2-l-l'$$
$$0 \geq -2$$

In the matching M_i' the same edges are included in both cases (between L_i and L_i' and between aL_i and bL_i'), moreover we know, that every edge between aL_i and bL_i' holds the same or larger value as the same edge between L_i and L_i'. Thus the sum of these edges is greater or equal to the sum of the same edges between L_i and L_i'.

$$\sum_{(w,w')\in M_i'} \frac{l+l'-2c}{l+l'} \leq \sum_{(aw,bw')\in M_i'} \frac{2+l+l'-2c}{2+l+l'}. \tag{5.3}$$

From Equations 5.1, 5.2 and 5.3 we obtain

$$\sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \leq \sum_{(aw,bw')\in M_i'} \frac{2+l+l'-2c}{2+l+l'} \leq \sum_{(aw,bw')\in M_i} \frac{2+l+l'-2c}{2+l+l'} \tag{5.4}$$

All matchings have the same cardinality, therefore (from the sandwich theorem) if we prove, that matching M_i is convergent between $a\Phi$ and $b\Phi'$ and has the same limit as matching M_i between Φ and Φ' we obtain, that matching M_i' is convergent between $a\Phi$ and $b\Phi'$ as well to the same number.

$$\sum_{(aw,bw')\in M_i} \frac{2+l+l'-2c}{2+l+l'} =$$
$$= \sum_{(aw,bw')\in M_i} \frac{2}{2+l+l'} + \sum_{(aw,bw')\in M_i} \frac{l+l'-2c}{2+l+l'} =$$
$$= \sum_{(w,w')\in M_i} \frac{2}{2+l+l'} + \sum_{(w,w')\in M_i} \frac{l+l'-2c}{2+l+l'} =$$
$$= 2 \sum_{(w,w')\in M_i} \frac{1}{2+l+l'} + \sum_{(w,w')\in M_i} \frac{l+l'-2c}{2+l+l'} \frac{l+l'}{l+l'} =$$
$$= 2 \sum_{(w,w')\in M_i} \frac{1}{2+l+l'} + \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \frac{l+l'}{2+l+l'} =$$
$$= 2 \sum_{(w,w')\in M_i} \frac{1}{2+l+l'} + \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \frac{2+l+l'-2}{2+l+l'} =$$
$$= 2 \sum_{(w,w')\in M_i} \frac{1}{2+l+l'} + \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \left(1 + \frac{-2}{2+l+l'}\right) =$$
$$= 2 \sum_{(w,w')\in M_i} \frac{1}{2+l+l'} + \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} + \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \frac{-2}{2+l+l'} =$$
$$= \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} + 2 \sum_{(w,w')\in M_i} \frac{1}{2+l+l'} - 2 \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \frac{1}{2+l+l'}$$

The first sum is the sum of the minimal weight maximal matching between L_i and L'_i, thus divided by the cardinality of M_i is convergent to $D_S(\Phi, \Phi')$. We have to prove, that the other two sums divided by the cardinality of matching M_i are convergent to 0, thus the sequence of numbers derived from M_i between aL_i and bL'_i is convergent to $D_S(\Phi, \Phi')$.

So we have to prove, that $\frac{2\sum_{(w,w')\in M_i} \frac{1}{2+l+l'}}{|M_i|}$ and $\frac{2\sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \frac{1}{2+l+l'}}{|M_i|}$ is convergent to 0. Both sequence contains non-negative numbers only, thus their limit can not be negative.

Let min be the minimal value of all $2+l+l'$ within the matching M_i. Then we can bound the sequences above as follows:

$$\frac{2\sum_{(w,w')\in M_i} \frac{1}{2+l+l'}}{|M_i|} \leq \frac{2\sum_{(w,w')\in M_i} \frac{1}{min}}{|M_i|} = \frac{\frac{2}{min}\sum_{(w,w')\in M_i} 1}{|M_i|} = \frac{\frac{2}{min}|M_i|}{|M_i|} = \frac{2}{min}$$

Similarly:

$$\frac{2\sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \frac{1}{2+l+l'}}{|M_i|} \leq \frac{2\sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \frac{1}{min}}{|M_i|} = \frac{2}{min} \frac{\sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'}}{|M_i|} \leq$$
$$\leq \frac{2}{min}$$

Both sequences can be bound by $\frac{2}{min}$. Now we show, that this can be ignored. From the last subsection we know, that we can erase a finite sequence of words from the monotone sequences without altering the distance between them. $lim_{min\to\infty}\frac{2}{min} = 0$. For every value $min = 1..$ we erase all words from both monotone sequences which are shorter the $\lceil\frac{min}{2}\rceil$. This is a finite set of words for every natural min. Thus as a limit we can enforce that both sequences above are upper bounded by 0.

We can conclude, that every $D_S(a\Phi, b\Phi') = D_S(\Phi, \Phi')$. □

Now let us prove a similar lemma for the case when both prefixes are the same character. The proof itself is a bit easier and similar to the proof above.

Lemma 5.4.2. *Let Φ and Φ' be monotone sequences with an existing strong distance measure, and a be a symbol. Then $D_S(a\Phi, a\Phi')$ exists as well, and*

$$D_S(a\Phi, a\Phi') = D_S(\Phi, \Phi').$$

Proof. Similarly to the lemma above let M_i be a minimal weight maximal matching between L_i and L'_i and let M'_i be a minimal weight maximal matching between aL_i and aL'_i.

The relative edit distance between two words (w and w') can be computed as $\frac{|w|+|w'|-2|LCS(w,w')|}{|w|+|w'|}$, where LCS is a longest common subsequence of the given words. For our purposes let $l := |w|$, $l' := |w'|$ and $c := |LCS(w, w')|$. Then the relative edit distance between w and w' is $\frac{l+l'-2c}{l+l'}$.

Using this notation we can calculate the sum of the edges in the matchings M_i and M'_i. Moreover, we can calculate the sum of the edges of the matching M_i between aL_i and aL'_i and M'_i between L_i and L'_i. The reason behind this is that there is a bijective correlation between the words from L_i and aL_i and between the words from L'_i and aL'_i. For the same reason matchings M_i and M'_i have the same cardinality.

At first we prove, that the relative edit distance between aw and aw' is at most then the relative edit distance between the same w and w'.

$$\begin{aligned}
\frac{l+l'-2c}{2+l+l'} &\leq \frac{l+l'-2c}{l+l'} \\
\frac{1}{2+l+l'} &\leq \frac{1}{l+l'} \\
l+l' &\leq 2+l+l' \\
0 &\leq 2
\end{aligned}$$

From the inequality above the following inequality follows:

$$\sum_{(w,w')\in M_i'} \frac{l+l'-2c}{l+l'} \geq \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} \geq$$
$$\geq \sum_{(aw,aw')\in M_i} \frac{l+l'-2c}{2+l+l'} \geq \sum_{(aw,aw')\in M_i'} \frac{l+l'-2c}{2+l+l'}$$

Where all matchings have the same cardinality. The first and the last inequality hold because M_i' and M_i is the minimal weight maximal matching for a given sets.

Let us show now, that the average of the edges in matching M_i is convergent to the same number as the average of the edges in M_i'. The following inequality holds which can be further modified as follows.

$$\begin{aligned}
0 &\leq \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} - \sum_{(aw,aw')\in M_i'} \frac{l+l'-2c}{2+l+l'} \leq \\
&\leq \sum_{(w,w')\in M_i'} \frac{l+l'-2c}{l+l'} - \sum_{(aw,aw')\in M_i'} \frac{l+l'-2c}{2+l+l'} = \\
&= \sum_{(aw,aw')\in M_i'} \left(\frac{l+l'-2c}{l+l'} - \frac{l+l'-2c}{2+l+l'} \right) = \\
&= \sum_{(aw,aw')\in M_i'} \left(\frac{(2+l+l')(l+l'-2c) - (l+l')(l+l'-2c)}{(l+l')(2+l+l')} \right) = \\
&= \sum_{(aw,aw')\in M_i'} \left(\frac{2(l+l'-2c) + (l+l')(l+l'-2c) - (l+l')(l+l'-2c)}{(l+l')(2+l+l')} \right) = \\
&= \sum_{(aw,aw')\in M_i'} \frac{2(l+l'-2c)}{(l+l')(2+l+l')} \leq \sum_{(aw,aw')\in M_i'} \frac{2(l+l')}{(l+l')(2+l+l')} = \\
&= \sum_{(aw,aw')\in M_i'} \frac{2}{(2+l+l')} = 2 \sum_{(aw,aw')\in M_i'} \frac{1}{(2+l+l')}
\end{aligned}$$

It is shown in the lemma above that $\frac{1}{(2+l+l')}$ can be ignored, precisely $lim_{min\to\infty} \frac{2}{min} = 0$ when min is the minimal length of the words in Φ and Φ'.

Therefore for growing min we have that

$$0 \leq \sum_{(w,w')\in M_i} \frac{l+l'-2c}{l+l'} - \sum_{(aw,aw')\in M_i'} \frac{l+l'-2c}{2+l+l'} \leq 0.$$

Which means that both matchings have the same limit and both limits exist if one of them exists.

So $D_S(a\Phi, a\Phi') = D_S(\Phi, \Phi')$ holds. □

From the lemmas in this section we have the following theorem:

Theorem 5.4.3. *Let Φ and Φ' be monotone sequences with an existing strong distance measure, and a and b two symbols, Then $D_S(a\Phi, b\Phi')$ exists as well, and*

$$D_S(a\Phi, b\Phi') = D_S(\Phi, \Phi').$$

Similar theorem holds for the finite suffixes as well as we are not using the fact, that a and b are prefixes in our proof.

Theorem 5.4.4. *Let Φ and Φ' be monotone sequences with an existing strong distance measure, and a and b two symbols, Then $D_S(\Phi a, \Phi' b)$ exists as well, and*

$$D_S(\Phi a, \Phi' b) = D_S(\Phi, \Phi').$$

Chapter 6

Sequences of Approximations Generated by Strict Grammars With Energy

In this chapter consider monotone sequences generated by strict grammars with energy and prove some useful properties of these monotone sequences.

6.1 Definitions and Basic Relations

In this section we define monotone sequences generated by grammars with energy as well as sequence of approximations based on the length of words and point out the relationship between the subsequence relationship and between the generation process of these monotone sequences.

Definition 6.1.1. *Let* $\{e_i\}_{i=1}^{\infty}$ *be a monotone sequence of real numbers from the open interval* $(0, 1)$ *convergent to* 0. *Sequence* $\{e_i\}_{i=1}^{\infty}$ *with these properties will be called* sequence of threshold values.

Definition 6.1.2. *Let* G *be a strict grammar with energy and* $\{e_i\}_{i=1}^{\infty}$ *be a sequence of threshold values. The sequence of finite languages* $\Phi(G, \{e_i\}_{i=1}^{\infty}) = \{L_{e_i}(G)\}_{i=1}^{\infty}$ *is a* sequence of approximations for language $L(\hat{G})$ generated by G using threshold values $\{e_i\}_{i=1}^{\infty}$.[1]

We have to prove, that $\Phi(G, \{e_i\}_{i=1}^{\infty})$ as defined above is a monotone sequence. From Theorem 1.2.2 and from the monotonicity of sequence $\{e_i\}_{i=1}^{\infty}$ we know, that for every $i > 0$ $e_i \geq e_{i+1}$, so $L_{e_i}(G) \subseteq L_{e_{i+1}}(G)$, thus $\Phi(G, \{e_i\}_{i=1}^{\infty})$ is a monotone sequence of finite languages (each language $L_e(G)$ is finite).

[1] $\hat{G}$ is an underlying context-free grammar to G from Notation 1.1.3.

Now we show, that the limit of $\Phi(G, \{e_i\}_{i=1}^{\infty})$ is $L(\hat{G})$. We have to show, that

(i) For all w in $L(\hat{G})$ there exists such i, that w is in $L_{e_i}(G)$;

(ii) For all i and for all w in $L_{e_i}(G)$ it holds that w in $L(\hat{G})$.

Condition (ii) is trivially true, because $L_{e_i}(G) \subset L(\hat{G})$ for all i. Condition (i) holds as well, because from Theorem 1.2.1 we know, that for every word $w \in L(\hat{G})$ there exists i such that $w \in L_{e_j}(G)$ for each $j \geq i$. From Theorem 1.2.1 we know, that there exists a strict grammar with energy for every context-free grammar $\hat{G}$, which satisfies condition (i). In this case G is that strict grammar with energy as $\hat{G}$ is the underlying grammar to G. The existence of such a low threshold is maintained by the convergence of sequence $\{e_i\}_{i=1}^{\infty}$.

Definition 6.1.3. $\mathcal{F}_{CF} = \{\Phi|$ *there exist a strict grammar with energy* G *and a sequence of threshold values* $\{e_i\}_{i=1}^{\infty}$ *such, that* $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})\}$. *We call* $\mathcal{F}_{CF}$ *the* class of context-free sequences of approximations.

Similarly we can define the linear and regular classes of sequences of approximations.

Definition 6.1.4. $\mathcal{F}_{LN} = \{\Phi|$ *there exist a strict grammar with energy* G*, where* $\hat{G}$ *is a linear grammar and a sequence of threshold values* $\{e_i\}_{i=1}^{\infty}$ *such, that* $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})\}$. *We call* $\mathcal{F}_{LN}$ *the* class of linear sequences of approximations.

Definition 6.1.5. $\mathcal{F}_{R} = \{\Phi|$ *there exist a strict grammar with energy* G*, where* $\hat{G}$ *is a regular grammar and a sequence of threshold values* $\{e_i\}_{i=1}^{\infty}$ *such, that* $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})\}$. *We call* $\mathcal{F}_{R}$ *the* class of regular sequences of approximations.

Now we define those monotone sequences, which are corresponding to the approximation defined by the length of the words.

Definition 6.1.6. *Let* L *be a language and* $\{k_i\}_{i=1}^{\infty}$ *be a monotone growing sequence of natural numbers. The sequence of finite languages generated as* $\Phi(L, \{k_i\}_{i=1}^{\infty}) = \{L_{\leq k_i}\}_{i=1}^{\infty}$*, where* $L_{\leq k_i}$ *is a set of all words from* L *with a length less or equal to* k_i*, is a* sequence of approximations for language L generated by the length-based approximation using $\{k_i\}_{i=1}^{\infty}$.

$\{L_{\leq k_i}\}_{i=1}^{\infty}$ is a monotone sequence of finite languages for every language L. It is easy to see, that the limit of $\Phi(L, \{k_i\}_{i=1}^{\infty})$ is the language L.

Lemma 6.1.7. *(i) Let* G *be a grammar with energy and* $\{e_i\}_{i=1}^{\infty}$ *be a sequence of threshold values. Let* $\{e'_i\}_{i=1}^{\infty}$ *be a subsequence of* $\{e_i\}_{i=1}^{\infty}$*, thus convergent to* 0 *as well. Then* $\Phi(G, \{e'_i\}_{i=1}^{\infty})$ *is a subsequence of* $\Phi(G, \{e_i\}_{i=1}^{\infty})$.

(ii) Let L be a language and $\{k_i\}_{i=1}^{\infty}$ be a monotone growing sequence of natural numbers. Let $\{k'_i\}_{i=1}^{\infty}$ be a subsequence of $\{k_i\}_{i=1}^{\infty}$. Then $\Phi(L, \{k'_i\}_{i=1}^{\infty})$ is a subsequence of $\Phi(L, \{k_i\}_{i=1}^{\infty})$.

Proof. The proof is identical for both cases, so we provide a proof for case (i) only.

We have to prove, that every $L_{e'_i}$ is from Φ and that the order of the finite languages in Φ' is the same as in Φ. The first property directly follows from the fact, that $\{e'_i\}_{i=1}^{\infty}$ is a subsequence of $\{e_i\}_{i=1}^{\infty}$. The second property follows from the fact, that both $\{e_i\}_{i=1}^{\infty}$ and $\{e'_i\}_{i=1}^{\infty}$ are monotone, thus the order of the numbers e'_i is the same in both sequence. □

Now we define the subset of the class of context-free (regular, linear) sequences of approximations, which is used later in this thesis.

Definition 6.1.8. $\mathcal{F}_{B-CF} = \{\Phi|$ *there exist a strict grammar with energy G and a sequence of threshold values $\{e_i\}_{i=1}^{\infty}$ for which $\lim_{i\to\infty} \frac{e_i}{e_{i-1}} = 1$ such, that $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})\}$. We call $\mathcal{F}_{B-CF}$ the* class of bounded context-free sequences of approximations.

Definition 6.1.9. $\mathcal{F}_{B-R}(\mathcal{F}_{B-LN}) = \{\Phi|$ *there exist a strict grammar with energy G, where $\hat{G}$ is a regular (linear) grammar and a sequence of threshold values $\{e_i\}_{i=1}^{\infty}$ for which $\lim_{i\to\infty} \frac{e_i}{e_{i-1}} = 1$ such, that $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})\}$. We call $\mathcal{F}_{B-R}$ ($\mathcal{F}_{B-LN}$) the* class of bounded regular (linear) sequences of approximations.

The inclusions $\mathcal{F}_{B-CF} \subseteq \mathcal{F}_{CF}$, $\mathcal{F}_{B-LN} \subseteq \mathcal{F}_{LN}$ and $\mathcal{F}_{B-R} \subseteq \mathcal{F}_R$ directly follow from the definition. Clearly $\mathcal{F}_R \subseteq \mathcal{F}_{LN} \subseteq \mathcal{F}_{CF}$ and $\mathcal{F}_{B-R} \subseteq \mathcal{F}_{B-LN} \subseteq \mathcal{F}_{B-CF}$ hold as well.

Lemma 6.1.10. *For every $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})$ every shortest word generated by σ in $\hat{G}$ is in $L_1 = L_{e_1}(G)$.*

Proof. So we have to prove, that every shortest word s derived from σ in $\hat{G}$ belongs to L_1. Without loss of generality s is one of the shortest words. There are two cases:

In the first case e_1 is such that the whole derivation of s is doable from σ^1 with energy-threshold e_1 without a substitution of the shortest words.

In the second case we have to substitute at least once the shortest word derived from one non-terminal symbol. Let s be $u_1v_1u_2v_2u_3...v_nu_{n+1}$, where u_i and v_i are terminal words or ε and the parts u_i are derived with energy e_1 and v_i are the parts, where shortest words are substituted. The situation is similar to Theorem 1.2.2, that we can derive word $u_1v'_1u_2v'_2u_3...v'_nu_{n+1}$ from σ with energy-threshold e_1 we obtain that for every i $|v_i| = |v'_i|$. Because grammars are generating all possible words we have a situation that $v_i = v'_i$, thus s is generated with energy-threshold e_1. □

Lemma 6.1.11. *For every $\Phi \in \mathcal{F}_X$ or $\Phi \in \mathcal{F}_{B-X}$: $L_\Phi \in \mathscr{L}_X$, where X can be R, LN and CF.*

Proof. All proofs are identical, so we prove this lemma for CF.

Every $\Phi \in \mathcal{F}_{CF}$ or $\Phi \in \mathcal{F}_{B-CF}$ is generated by a strict grammar with energy G. The underlying grammar $\hat{G}$ is context-free by the definition of $\mathcal{F}_{CF}$ or $\mathcal{F}_{B-CF}$. From the definitions above we know, that $L_\Phi = L(\hat{G})$ and $L(G) \in \mathscr{L}_{CF}$. □

Now we show the second subset relation:

Lemma 6.1.12. *For every $L \in \mathscr{L}_X$ there exists such sequence of approximation Φ, that $\Phi \in \mathcal{F}_{B-X}$, where X can be R, LN and CF.*

Proof. For every (regular, linear, context-free) language exist a (regular, linear, context-free) grammar without unreachable and unproductive non-terminal symbols. We can add one random coefficient for each non-terminal symbol on the right side of the rewriting rules and so we create a strict (regular, linear, context-free) grammar with energy G.

We can create Φ as $\Phi(G, \{e_i\}_{i=1}^{\infty})$ where $e_i = \frac{1}{i}$. From the definition $\Phi \in \mathcal{F}_{B-X}$. □

From the two lemmas above and the fact that $\mathscr{R} \subset \mathscr{L}_{LN} \subset \mathscr{L}_{CF}$ we obtain that $\mathcal{F}_{B-R} \subset \mathcal{F}_{B-LN} \subset \mathcal{F}_{B-CF}$ and $\mathcal{F}_R \subset \mathcal{F}_{LN} \subset \mathcal{F}_{CF}$.

Lemma 6.1.13. *For every $\Phi \in \mathcal{F}_X$ approximating a finite language it holds $\Phi \in \mathcal{F}_{B-X}$, where X can be R, LN and CF.*

Proof. We prove the lemma for the regular sequences of approximations, the proof is almost identical for the other two cases.

$\Phi \in \mathcal{F}_R \implies \Phi(G, \{e_i\}_{i=1}^{\infty})$ and G is a strict regular grammar with energy. Let k be such a natural number, that $L_k = L_\Phi$. Let e_k be the energy threshold related to L_k. Now let us create sequence of thresholds $\{e'_i\}_{i=1}^{\infty}$ with property $\lim_{i\to\infty} \frac{e'_i}{e'_{i-1}} = 1$.

$\{\frac{1}{n}\}_{n=1}^{\infty}$ is a monotone convergent sequence of real numbers. $\lim_{n\to\infty} \frac{\frac{1}{n}}{\frac{1}{n-1}} = \lim_{n\to\infty} \frac{n-1}{n} = 1$.

Let n_0 be the first number, for which $\frac{1}{n_0} \leq e_k$.

Now we can define e'_i as follows:

$$e'_i = \begin{cases} e_i & \text{for every } i \leq k \\ \frac{1}{n_0+i-k} & \text{otherwise} \end{cases}$$

$\Phi' = \Phi(G, \{e'_i\}_{i=1}^{\infty}) \in \mathcal{F}_{B-R}$, because $\lim_{i\to\infty} \frac{e'_i}{e'_{i-1}} = 1$.

Now we show, that $\Phi = \Phi'$. For the first k finite approximations $L_i = L_{e_i}(G) = L'_i = L_{e'_i}(G) = L_{e_i}(G)$. The finite approximations in both sequences of approximations are $L_\Phi = L_{\Phi'}$ from the same j_max, thus Φ and Φ' are the same.

$\Phi \in \mathcal{F}_{B-R}$. □

Theorem 6.1.14. *For every $\Phi \in \mathcal{F}_{CF}$ approximating a finite language it holds $\Phi \in \mathcal{F}_R$.*

Proof. We have to construct a strict regular grammar with energy G and a sequence of energy-thresholds e_i, which $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})$. Without loss of generality $L = \{w_j | 1 \leq j \leq n \wedge n \geq 1\}$. Now we can create a list of first occurrences for each word. So let f be a function, which tell us the first occurrence of each word. So $w_j \notin L_l$ for $l < f(w_j)$, but $w_j \in L_{f(w_j)}$.

From Lemma 6.1.10 we know that the shortest words generated by the original starting non-terminal symbol σ' are in L_1. Because we are approximating a finite language these are the shortest words in general. Let s be one of the shortest words from L, thus a $f(s) = 1$.

Now we construct a regular grammar with energy as follows:

1. We add a "mid-rule" for every word from L: $\sigma \to \xi_j^{[\frac{1}{f(w_j)}]}$.
2. We add a "production-rule" for each word from L: $\xi_j \to w_j$.
3. We add a "escaping-rule" for each non-terminal ξ_j: $\xi_j \to s$.

We construct the sequence of thresholds as follows: We construct a set of first occurrences as $F = \{f_1 < f_2 < ... < f_m\}$. So let us define the sequence of threshold values: Let f_l be the first number in the set F bigger then i. Then $e_i = \frac{1}{f_{l-1}}$.

Now we should prove, that $L_i = L_{e_i}(G)$, where $L_i \; in \Phi$.

1. Proof of $L_i \subseteq L_{e_i}(G)$. Each word in L_i has its first occurrence prior to L_i or in L_i. So each word has its $f(w) \leq i$, so $\frac{1}{f(w)} \geq \frac{1}{i}$. Now we can do our first step in a derivation: we obtain $\xi_j^{\frac{1}{i}}$ for each ξ_j. From the fact that $\frac{1}{f(w)} \geq \frac{1}{i}$ for every word in L we obtain that the corresponding ξ_js' has enough energy to derive the words from L_i.

2. Proof of $L_{e_i}(G) \subseteq L_i$. Every word in $L_{e_i}(G)$ is either s (substituted because of the energy threshold) or a word, which has higher coefficient then e_i. $s \in L_1$ from Lemma 6.1.10, so $s \in L_i$ as well. Now the words, which have higher coefficient then e_i has a lower first occurrence then $\frac{1}{e_i} < f_l$. On the other hand f_l is the first number bigger then i which has a new first occurrence of some word, so $\frac{1}{i} \leq i$. From this we see, that each word in $L_{e_i}(G)$ has its first occurrence before L_i in Φ.

$L_i = L_{e_i}(G)$ thus our proof is complete.

□

From the theorem and lemma above we obtain:

Corollary 6.1.15. *For every $\Phi \in \mathcal{F}_{CF}$ approximating a finite language it holds $\Phi \in \mathcal{F}_{B-R}$.*

6.2 Difference Between Length-Based Approximation and Approximations Using Grammars With Energy

Now we have tools to examine what is the difference between the length-based approximation a given language and between the approximations generated by grammars with energy. The intuition is, that while approximating with grammars with energy we can highlight some part of the grammar, which is "more relevant" than other parts of the grammar. This is something which can not be done by the length-based approximation, because in this case the languages are approximated, so the additional information given by the grammar is lost.

6.2.1 Context-Free Sequences of Approximations

In this section we show that there is a context-free sequence of approximations, which can not be generated by the length of the words. Later we show one monotone sequence bounded by the length of the words included in finite approximations as well, which is approximating context-free language but can not be generated by the grammar with energy.

Let us show one example of monotone sequence of finite languages which is generated by grammar with energy and can not be generated by length-based sequences of approximations.

Example 6.2.1. Let $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})$ be a monotone sequence of finite languages generated by grammar with energy $G = (N, \Sigma, P, \sigma)$ where $N = \{\sigma, \alpha, \beta\}$ and

$$P = \left\{\sigma \to \alpha^{[0.9]} \mid \beta^{[0.8]} \mid a \mid b, \alpha \to a\alpha^{[0.9]} \mid a, \beta \to b\beta^{[0.8]} \mid b\right\}.$$

In this case $L_{e_i}(G)$ contains words of two types a^* and b^*. For every $e_i \leq 0.9$ there are longer words of type a^* then the words of type b^*. Moreover the language being approximated is $L = \{a^n \cup b^n | n \geq 1\}$, so $L_{e_i}(G)$ is not a finite language containing each word shorter then some number.

On the other hand let us show an example of a sequence of approximation which can not be generated by grammars with energy.

Example 6.2.2. Let L be the context-free language $L = \{a^n b^n c^m d^m \mid n \geq 1\}$. Now let us define the monotone sequence of finite languages as follows: $\Phi = \{L_i\}_{i=1}^{\infty}$ where $L_i = \{w | \forall w \in L \wedge |w| \leq 2i\}$.

$L_\Phi = L$ and L is a context-free language, so there is a context-free grammar $\hat{G}$ generating this language, therefore there is a grammar with energy which has this grammar as its underlying grammar. Now consider the structure of grammar $\hat{G}$. From the context-freeness of $\hat{G}$ there should be two nests of non-terminal symbols, each nest generating the $a^n b^n$ and $c^m d^m$ part respectively. These two nests can not interfere after some finite amount of steps.

So let us consider the first i in a form $i = n' + m' - 1$ which is after the finite amount of steps where the two nests can interfere. In this situation L_i contains both words $a^{n'} b^{n'} c^{m'-1} d^{m'-1}$ and $a^{n'-1} b^{n'-1} c^{m'} d^{m'}$ but does not contain word $a^{n'} b^{n'} c^{m'} d^{m'}$. Now is we consider any grammar with energy with two not interfering nests we need such energy threshold e that $L_e(G) = L_i$. On the other hand both words $a^{n'} b^{n'} c^{m'-1} d^{m'-1}$ and $a^{n'-1} b^{n'-1} c^{m'} d^{m'}$ has to be generated by G with energy threshold e and such both $a^{n'} b^{n'}$ and $c^{m'} c^{m'}$ can be generated in the respective nest with energy threshold e. Therefore $a^{n'} b^{n'} c^{m'} d^{m'} \in L_e(G)$.

In this example we abused the fact that the boundary put onto the length of the words is a global condition, but the energy threshold for the non-terminal symbols is a local condition for each nest.

6.3 Pumping Lemma

Now we prove the first lemma, the first necessary condition of the context-free sequences of approximations, similar to the Pumping Lemma for the context-free languages. From now on we refer to the Pumping Lemma for the context-free languages as *original Pumping Lemma*.

Lemma 6.3.1 (Pumping Lemma)**.** *For every Φ in $\mathcal{F}_{CF}$ exist some integer $p, q \geq 1$ such that for every j in $\mathbb{N}$, for every word w in L_j, with $|w| > p$ exist such u, v, x, y, z that the following four conditions hold:*

1. $w = uvxyz$
2. $|vxy| \leq q$
3. $|vy| \geq 1$, *and*

4. For every natural $n \geq 0$ there exists a natural number i, such that $uv^nxy^nz \in L_i$.

Proof. We prove, that all conditions hold.

We know, that word $w \in L_j \subset L_\Phi \in \mathscr{L}_{CF}$ is a word from a context-free grammar. Let p and q be the constants correlated with L_Φ from the original Pumping Lemma. Then from the original Pumping Lemma we know, that the first three conditions hold. We know, that for every $n \geq 0$ $uv^nxy^nz \in L_\Phi$. Let this word be $w_n := uv^nxy^nz$, thus every $w_n \in L_\Phi$.

From the definition of the limit of the sequence of approximations we know, that for very word w' from L_Φ there is such k, that w' in L_k. Thus for every w_n there is such a number i, that w_n in L_i, thus the fourth condition holds as well.

When $n = 1$, then $uv^nxy^nz = w$, thus from the condition, that L_j is the first occurrence of word w we obtain that $i \geq j$. □

6.4 Predecessor Lemma

The second necessary condition is from the opposite point of view to the Pumping Lemma, which means, that every "new" word in L_{i+1} has to has a "predecessor" in L_i.

Lemma 6.4.1 (Predecessor Lemma)**.** *For every $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty}) \in \mathcal{F}_{CF}$, for every word $w \in L_{i+1} \setminus L_i$, where $i > 1$ exists a word w' in L_i, such that w' is obtained by the substitution of the shortest words into the sentential form in $w'' \Rightarrow_{G,e_i} w'$ and $w'' \Rightarrow^*_{G,e_{i+1}} w$. Word w' is called a* predecessor *for word w.*

Proof. By contradiction. Let $w \in L_{i+1} \setminus L_i$ be such a word, that w has no predecessor in L_i. Let $\sigma^1 \Rightarrow_{G,e_{i+1}} v_1 \Rightarrow_{G,e_{i+1}} v_2 \Rightarrow_{G,e_{i+1}} \dots \Rightarrow_{G,e_{i+1}} v_n \Rightarrow_{G,e_{i+1}} w$ be one of the derivations of word w with energy threshold e_{i+1} corresponding to set L_{i+1}. w has no predecessor in L_i, thus when we transform this derivation[2] into energy threshold e_i there is no need for a substitution of the shortest words for any of the non-terminals in $\sigma, v_1, v_2, \cdots, v_n$, which are not substituted with the energy threshold e_{i+1} as well. But from this we obtain, that $\sigma^1 \Rightarrow_{G,e_i} v_1 \Rightarrow_{G,e_i} v_2 \Rightarrow_{G,e_i} \dots \Rightarrow_{G,e_i} v_n \Rightarrow_{G,e_i} w$ is a valid derivation as well, thus w in L_i, which is a contradiction. □

There can be more then one predecessor for each word $w \in L_{i+1} \setminus L_i$ in L_i and each word w' in L_i can be a predecessor for more then one word $w \in L_{i+1} \setminus L_i$. Moreover there can be words in L_i, which are not predecessor for any word from $L_{i+1} \setminus L_i$.

One interesting fact comes from this lemma as well: there is only a finite amount of patterns, which can be "reverted" as the derivation goes on with a smaller energy

[2]Changing the energy threshold for all steps from e_{i+1} to e_i.

threshold. These patterns are all the shortest words, which can be substituted instead of a non-terminal symbol with not enough energy.

Chapter 7

Strong Distance Measure on Context-Free Sequences of Approximations

In this chapter we show some additional properties of the strong distance measure on the class of the context-free sequences of approximations.

It is obvious that when $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})$ and $\Phi' = \Phi(G', \{e'_i\}_{i=1}^{\infty})$ are monotone sequences on *disjoint alphabets* every edge will have value 1, thus $D_S(\Phi, \Phi') = 1$. Similarly for the "almost" disjoint alphabets, with no non-trivial terminal in their intersection. Non-trivial terminals are defined below.

Terminal symbols not included in both alphabets $\Sigma_{L(\hat{G})}$ and $\Sigma_{L(\hat{G'})}$ are irrelevant in terms of the distance measure, because every such terminal symbol necessarily create an "error" between the words. Therefore these symbols can be substituted by an *error mark*: $_{\text{-}\Phi}$ for the sequence of approximations Φ. For example, instead of computing the distance between $a^k b^k$ and $a^k c^k$ we can write $a^k {}_{\text{-}\Phi}{}^k$ and $a^k {}_{\text{-}\Phi'}{}^k$.

We can split the set of terminal symbols included in both alphabets into two subsets:

- Terminal a in $\Sigma_{L(\hat{G})} \cap \Sigma_{L(\hat{G'})}$ is called *trivial*, when it occurs only in a finite number of words.
- Terminal a in $\Sigma_{L(\hat{G})} \cap \Sigma_{L(\hat{G'})}$ is called *non-trivial*.

We can see from Theorem 5.3.2 that trivial terminal symbols do not affect the strong distance measure, because we can simply delete those words from the sequences of approximations, which contain these terminals, without changing the distance (or the existence of the distance) of the two sequences of approximations.

7.1 Existence

In this section we show that the strong distance measure does exist between every pair of context-free sequences of approximations, satisfying some properties. At first, we prove that the strong distance measure exists between each pair of monotone sequences, which are generated by the same grammar with energy. In this case we prove that the strong distance between these monotone sequences is always 0.

Furthermore we prove that the strong distance measure exists between every pair of context-free sequences of approximations satisfying some property. In this case however, we can not compute the exact value of this distance.

Lemma 7.1.1. *Let G be a strict grammar with energy and $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})$, $\Phi' = \Phi(G, \{e'_i\}_{i=1}^{\infty})$ two sequences of approximations generated by the same grammar G. Then*

$$D_S(\Phi, \Phi') = 0.$$

Proof. The finite approximations are $L_i = L_{e_i}(G)$ and $L'_i = L_{e'_i}(G)$. From Theorem 1.2.2 we know, that $L_i \subseteq L'_i$ or $L'_i \subseteq L_i$ depending on the values of e_i and e'_i.[1]

We create a maximal matching with overall weight 0, thus a minimal weight maximal matching between L_i and L'_i.[2] We have no restrictions for i, thus we can use this matching for every i and create a sequence of zeros, which corresponds to the sequence from the definition of the strong distance measure. This sequence is convergent and its limit is 0.

The matching between L_i and L'_i will be the following: Without loss of generality $L_i \subseteq L'_i$. Then match each word from L_i to itself from L'_i, these edges hold value 0. The maximality of this matching is trivial.

The proof is finished, as we constructed a maximal matching with 0 total weight. □

Theorem 7.1.2. *For every pair Φ, Φ' in $\mathcal{F}_{CF}$ for which the following conditions hold $\lim_{i\to\infty} \frac{|L_i|}{|L_{i-1}|} = 1$ and $\lim_{i\to\infty} \frac{|L'_i|}{|L'_{i-1}|} = 1$: $D_S(\Phi, \Phi')$ exists.*

Proof. From Lemmas 4.4.3, 4.4.5, and from the symmetry we know that the strong distance measure exists when one of the sequences of approximations is approximating a finite language. The only case, which is not proved is, when both sequences of approximations are approximating infinite languages.

So let Φ and Φ' approximate infinite languages. We prove, that from some n_0 in $\mathbb{N}$ the average value of edges in the minimal weight maximal matching between L_i and L'_i is the same, as between L_{i+1} and L'_{i+1}.

[1] Both of these inclusions can hold at the same time as well.

[2] There is no better matching between two fractional languages, because all edge between L_i and L'_i holds a non-negative value.

We have two similar cases:

1. $|L_i| \leq |L'_i|$ and $|L_{i+1}| \leq |L'_{i+1}|$
2. $|L_i| \leq |L'_i|$ and $|L_{i+1}| \geq |L'_{i+1}|$

Because of the symmetry we can omit the case, where $|L'_i| \leq |L_i|$ and $|L'_{i+1}| \leq |L_{i+1}|$.

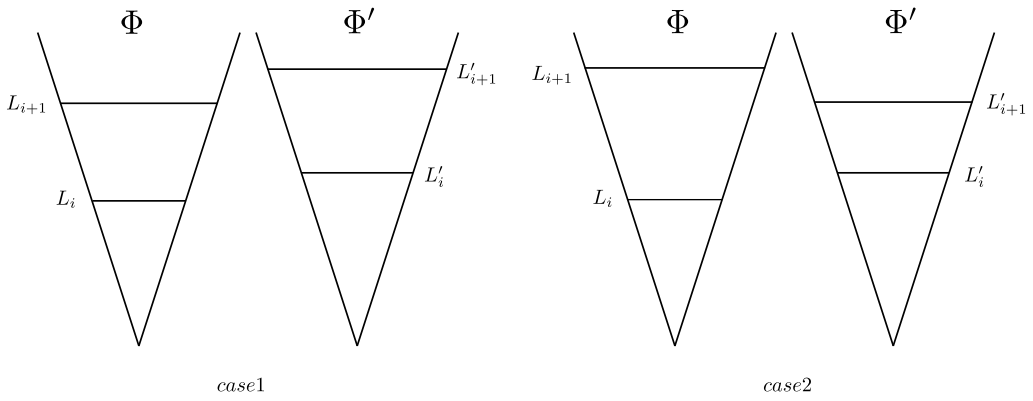

Figure 7.1: The two main case of this proof.

Let M_i be a minimal weight maximal matching between L_i and L'_i and M_{i+1} the minimal weight maximal matching between L_{i+1} and L'_{i+1}. Let s_i be a sum of edges in M_i as well as s_{i+1} be a sum of edges in M_{i+1}.

Let us bound s_{i+1} using s_i for every $i > n_0$ for both cases:

1. Let M'_{i+1} be a maximal matching created as follows: $M_i \subseteq M'_{i+1}$, for every word w in $L_{i+1} - L_i$ add a new edge for one of the previously unmatched words from L'_i. The maximality of M'_{i+1} follows from $|L_{i+1}| \leq |L'_{i+1}|$. The sum of edges in this matching is less or equal to $s_i + |L_{i+1}| - |L_i|$, because every edge added for words w in $L_{i+1} - L_i$ can possibly have value 1.

   From the minimality of matching M_{i+1}: $s_{i+1} \leq s_i + |L_{i+1}| - |L_i|$.

   Now let M'_i be a maximal matching created as follows: Delete from the matching M_{i+1} those edges, which have at least one incident word from $L_{i+1} - L_i$ or $L'_{i+1} - L'_i$. For every word w in L_i, which is not matched after this deletion add an edge with a previously unmatched word from L'_i. The maximality follows from $|L_i| \leq |L'_i|$ and the maximal number of such new edges is $|L'_{i+1}| - |L'_i|$ (from the assumption for big enough i this is much less then $|L_i|$).

   From the minimality of matching M_i: $s_i \leq s_{i+1} - |L'_{i+1}| + |L'_i| \leq s_{i+1}$.

   Thus $s_i \leq s_{i+1} \leq s_i + |L_{i+1}| - |L_i|$.

$$
\begin{aligned}
s_i \leq s_{i+1} &\leq s_i + |L_{i+1}| - |L_i| \\
\frac{s_i}{|L_{i+1}|} \leq \frac{s_{i+1}}{|L_{i+1}|} &\leq \frac{s_i}{|L_{i+1}|} + \frac{|L_{i+1}|}{|L_{i+1}|} - \frac{|L_i|}{|L_{i+1}|} \\
\frac{s_i}{|L_i|}\frac{|L_i|}{|L_{i+1}|} \leq \frac{s_{i+1}}{|L_{i+1}|} &\leq \frac{s_i}{|L_i|}\frac{|L_i|}{|L_{i+1}|} + 1 - \frac{|L_i|}{|L_{i+1}|} \\
D_i^{(S)}\frac{|L_i|}{|L_{i+1}|} \leq D_{i+1}^{(S)} &\leq D_i^{(S)}\frac{|L_i|}{|L_{i+1}|} + 1 - \frac{|L_i|}{|L_{i+1}|}
\end{aligned}
$$

According to the assumption from this theorem: with growing i $\frac{|L_i|}{|L_{i+1}|}$ will be more and more precisely 1. Therefore with growing i $D_i^{(S)} = D_{i+1}^{(S)}$.

2. Similarly to the first case we define matching M'_{i+1}: Add each edge from M_i and add a new edge with a previously unmatched vertex for every word w in L'_{i+1}, which is not matched by M_i. The maximality of this matching is trivial.

   From the minimality of M_{i+1}: $s_{i+1} \leq s_i - |L'_{i+1}| + |L_i|$.

   M'_i is defined as in the first case. The maximal number of edges being added after the deletion is $|L'_{i+1}| - |L'_i|$.

   From the minimality of M_i: $s_i \leq s_{i+1} - |L'_{i+1}| + |L'_i| \leq s_{i+1}$.

   Hence $s_i \leq s_{i+1} \leq s_i - |L'_{i+1}| + |L_i|$.

$$
\begin{aligned}
s_i \leq s_{i+1} &\leq s_i - |L'_{i+1}| + |L_i| \\
\frac{s_i}{|L'_{i+1}|} \leq \frac{s_{i+1}}{|L'_{i+1}|} &\leq \frac{s_i}{|L'_{i+1}|} - \frac{|L'_{i+1}|}{|L'_{i+1}|} + \frac{|L_i|}{|L'_{i+1}|} \\
\frac{s_i}{|L_i|}\frac{|L_i|}{|L'_{i+1}|} \leq D_{i+1}^{(S)} &\leq \frac{s_i}{|L_i|}\frac{|L_i|}{|L'_{i+1}|} - 1 + \frac{|L_i|}{|L'_{i+1}|} \\
D_i^{(S)}\frac{|L_i|}{|L'_{i+1}|} \leq D_{i+1}^{(S)} &\leq D_i^{(S)}\frac{|L_i|}{|L'_{i+1}|} - 1 + \frac{|L_i|}{|L'_{i+1}|} \\
D_i^{(S)}\frac{|L_i|}{|L_{i+1}|} \leq D_{i+1}^{(S)} &\leq D_i^{(S)}\frac{|L'_i|}{|L'_{i+1}|} - 1 + \frac{|L'_i|}{|L'_{i+1}|}
\end{aligned}
$$

From the conditions: with growing i $\frac{|L_i|}{|L_{i+1}|}$ and $\frac{|L'_i|}{|L'_{i+1}|}$ will be more and more precisely 1. Thus with growing i $D_i^{(S)} = D_{i+1}^{(S)}$.

So in each case with growing $i \to \infty$ $D_i^{(S)} = D_{i+1}^{(S)}$ and that is why the sequence is convergent, and $D_S(\Phi, \Phi')$ exists.

□

7.2 Strong Distance Only Depends on Grammar With Energy

Now we prove, that the strong distance measure depends only on grammars with energy, which generates context-free sequences of approximations.

To prove our main theorem in this section we need to define the concept of merging two monotone sequences into one. Informally, merging is an operation defined on two monotone sequences generated by the same strict grammar with energy, which creates a third monotone sequence using only those finite approximations, which are used in the two original monotone sequence. Moreover both original sequences are fully contained in the third, merged sequence.

Definition 7.2.1. *Let Φ and Φ' be monotone sequences generated by the same strict grammar with energy. The monotone sequence Φ'' is called a* merging *of Φ and Φ' denoted as Φ'' in $\Phi \bowtie \Phi'$ iff*

1. *For every L''_i in Φ'' at least one of the following statements hold: L''_i in Φ or L''_i in Φ'.*

2. *For every L_i in Φ (L'_i in Φ'): L_i in Φ'' (L'_i in Φ'').*

When Φ and Φ' are generated by different grammars we do not define the merging.

For every Φ'' in $\Phi \bowtie \Phi'$ Φ'' is generated by the same grammar with energy as Φ (Φ').

The easiest way of merging two monotone sequence of finite languages is to create Φ'' in $\Phi \bowtie \Phi'$ as follows:

1. step: At the beginning let $L_a := L_1$ and $L'_{a'} := L'_1$ be the "actual" finite languages. Let $\{e''_i\}_{i=1}^{\infty}$ and Φ'' be empty. Let j be the smallest natural number, that Φ'' is created up to j. So $j := 1$.

2. step: From Theorem 1.2.2 at least one of the statement holds: $L_a \subseteq L'_{a'}$ or $L'_{a'} \subseteq L_a$. Without loss of generality let the first statement be true.

3. step: Add the L_a into Φ'' at position j

4. step: Increase a by 1

5. step: Add e_a into $\{e''_i\}_{i=1}^{\infty}$ into position j

6. step: Increase j by 1

7. step: Repeat from step 2.

Then $\Phi'' = \Phi(G, \{e''_i\}_{i=1}^{\infty})$, where Φ is generated by G.

We can see, that each finite language L_i in Φ (L'_i in Φ') can be used more then once in Φ'' in $\Phi \bowtie \Phi'$.

Definition 7.2.2. *Class of context-free sequences of approximations $\mathcal{F} \subseteq \mathcal{F}_{CF}$ is closed under merging, when for every Φ, Φ' in $\mathcal{F}$ generated by the same strict grammar with energy $\Phi \bowtie \Phi' \subseteq \mathcal{F}$.*

Theorem 7.2.3. *Let G and $\bar{G}$ be strict grammars with energy. Let $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})$, $\Phi' = \Phi(G, \{e'_i\}_{i=1}^{\infty})$, $\bar{\Phi} = \Phi(\bar{G}, \{\bar{e}_i\}_{i=1}^{\infty})$ and $\bar{\Phi}' = \Phi(\bar{G}, \{\bar{e}'_i\}_{i=1}^{\infty})$ and $\Phi, \Phi', \bar{\Phi}, \bar{\Phi}' \in \mathcal{F}$, where $\mathcal{F}$ is closed under merging. If for every pair $\Phi, \Phi' \in \mathcal{F}$ exists the strong distance measure, then for these monotone sequences*

$$D_S(\Phi, \bar{\Phi}) = D_S(\Phi', \bar{\Phi}').$$

Proof. We prove this theorem by contradiction. Let $d_1 = D_S(\Phi, \bar{\Phi}) \neq D_S(\Phi', \bar{\Phi}') = d_2$. We create a pair of context-free sequences of approximations, which are strongly incomparable and are from $\mathcal{F}$. We create these two monotone sequences by using clever merging of $\Phi^{(1)}$ in $\Phi \bowtie \Phi'$ as well as $\Phi^{(2)}$ in $\bar{\Phi} \bowtie \bar{\Phi}'$.

Each infinite subsequence of a convergent sequence is convergent to the same number as the original sequence. Thus we enforce by merging that the sequence of real numbers corresponding to the strong distance measure between $\Phi^{(1)}$ and $\Phi^{(2)}$ contains an infinite subsequence of both sequences used in a computation of $D_S(\Phi, \bar{\Phi})$ and $D_S(\Phi', \bar{\Phi}')$. Therefore the sequence of real numbers corresponding to the strong distance between $\Phi^{(1)}$ and $\Phi^{(2)}$ has no limit, so $\Phi^{(1)}$ and $\Phi^{(2)}$ are strongly incomparable, which is a contradiction.

The existence of $D_S(\Phi, \bar{\Phi})$ and $D_S(\Phi', \bar{\Phi}')$ is ensured by the conditions.

Now we show, how to create $\Phi^{(1)}$ and $\Phi^{(2)}$ by simultaneous merging of Φ and Φ' as well as $\bar{\Phi}$ and $\bar{\Phi}'$.

To ensure, that $\Phi^{(1)}$ and $\Phi^{(2)}$ are strongly incomparable, there must be from time to time a pair of finite approximations within $\Phi^{(1)}$ and $\Phi^{(2)}$, which is from Φ and $\bar{\Phi}$ (Φ' and $\bar{\Phi}'$). Formally there must be two infinite growing sequence of numbers $\{k_i\}_{i=1}^{\infty}$ and $\{k'_i\}_{i=1}^{\infty}$, such that for every i $L^{(1)}_{k_i} = L_n$ and $L^{(2)}_{k_i} = \bar{L}_n$ and $L^{(1)}_{k'_i} = L'_n$ and $L^{(2)}_{k'_i} = \bar{L}'_n$ for some n.

We create such $\Phi^{(1)}$ and $\Phi^{(2)}$ by using so called *synchronizing steps* during the merging:
The input for this synchronizing step is one of the following pairs: Φ and $\bar{\Phi}$; Φ' and $\bar{\Phi}'$, that states which pairs should be synchronized during the merging process in this step.

The setup before the synchronizing step:
Let j and j' be such numbers that each finite approximation from Φ and Φ' before L_j and $L'_{j'}$ is merged into $\Phi^{(1)}$, but L_j is not in $\Phi^{(1)}$ and $L'_{j'}$ is not in $\Phi^{(1)}$. Let similarly define $\bar{j}$ and $\bar{j}'$ for $\Phi^{(2)}$. The aim of the synchronizing step:
To merge Φ and Φ' into $\Phi^{(1)}$ and $\bar{\Phi}$ and $\bar{\Phi}'$ into $\Phi^{(2)}$, that for some k and $n > j$ and $n > j'$ $L_k^{(1)} = L_n$ and $L_k^{(2)} = \bar{L}_n$ (for input Φ and $\bar{\Phi}$) and nothing more is merged.[3]

We show the algorithm for input Φ and $\bar{\Phi}$, for Φ' and $\bar{\Phi}'$ the algorithm is similar, just the indexes and corresponding sets should be exchanged.

1. step: The choice of n. Let choose such n, that $L_j \subseteq L_n$, $L'_{j'} \subseteq L_n$, $\bar{L}_{\bar{j}} \subseteq \bar{L}_n$ and $\bar{L}'_{\bar{j}'} \subseteq \bar{L}_n$ is true at the same time.

2. step: Simple merging. Merge Φ and Φ' into $\Phi^{(1)}$ and $\bar{\Phi}$ and $\bar{\Phi}'$ into $\Phi^{(2)}$ simultaneously[4] as simply as it is introduced in the preface of this section until one of the sets L_n or $\bar{L}_n$ is in $\Phi^{(1)}$ or $\Phi^{(2)}$ respectively. Then stop the merging process.

3. step: Without loss of generality let $L_n = L_{k_1}^{(1)}$. If $\bar{L}_n = L_{k_1}^{(2)}$ holds as well then the synchronizing step is done.

4. step: Without loss of generality let $L_n = L_{k_1}^{(1)}$ and $L_{k_1}^{(2)} \subseteq \bar{L}_n$. Now merge $\bar{\Phi}$ and $\bar{\Phi}'$ into $\Phi^{(2)}$ as it is introduced in the preface of this section, but when a set is added into $\Phi^{(2)}$ add L_n into $\Phi^{(1)}$, while in $\Phi^{(1)}$ there will be as many L_ns after each other as needed. Stop the merging, when $\bar{L}_n$ in $\Phi^{(2)}$.

By using the synchronizing step above at first with input Φ and $\bar{\Phi}$ then Φ' and $\bar{\Phi}'$ it is ensured, that the sequence of real numbers corresponding to the strong distance between $\Phi^{(1)}$ and $\Phi^{(2)}$ will contain one element of both sequences for $D_S(\Phi, \bar{\Phi})$ and $D_S(\Phi', \bar{\Phi}')$. Hence such merging which is created by an iteration of this synchronizing step creates such $\Phi^{(1)}$ and $\Phi^{(2)}$, what has no strong distance and this is a contradiction.

□

Corollary 7.2.4. *Let G and $\bar{G}$ be strict grammars with energy. Let $\Phi = \Phi(G, \{e_i\}_{i=1}^{\infty})$, $\Phi' = \Phi(G, \{e'_i\}_{i=1}^{\infty})$, $\bar{\Phi} = \Phi(\bar{G}, \{\bar{e}_i\}_{i=1}^{\infty})$ and $\bar{\Phi}' = \Phi(\bar{G}, \{\bar{e}'_i\}_{i=1}^{\infty})$ all four fulfilling condition $\lim_{i\to\infty} \frac{|L_i|}{|L_{i-1}|} = 1$. Then for these monotone sequences*

$$D_S(\Phi, \bar{\Phi}) = D_S(\Phi', \bar{\Phi}').$$

[3] After $L_k^{(1)}$ the merging is not finished.
[4] Add one set to each $\Phi^{(1)}$ and $\Phi^{(2)}$ at the same time.

Proof. We have to show, that when context-free sequences of approximations Φ and Φ' fulfill the condition $\lim_{i\to\infty} \frac{|L_i|}{|L_{i-1}|} = 1$, then every Φ'' in $\Phi \bowtie \Phi'$ fulfill this condition, too. $\frac{|L''_i|}{|L''_{i-1}|}$ can be bounded for every i as follows: $1 \leq \frac{|L''_i|}{|L''_{i-1}|}$. On the other hand, L''_{i-1} is L_j or L'_j for some j. Without loss of generality $L''_{i-1} = L_j$. Then $L''_i \subseteq L_{j+1}$, thus $\frac{|L''_i|}{|L''_{i-1}|} \leq \frac{|L_i|}{|L_{i-1}|}$. □

Conclusion

The main goal of this thesis was to introduce a distance measure between grammars and languages. This is a third thesis dealing with this topic, since Jánošík in Thesis [4] and Kulich in Thesis [7] have already introduced grammars with energy and the relative edit-distance between the words respectively.

Firstly, we modified the grammars with energy so they do not generate words which are not included in the language being approximated. We also introduced strict grammars with energy which were later used in this thesis and can be considered as a "normal form" of the grammars with energy.

Besides we introduced monotone sequences of finite languages, which can be considered as better and better approximations of the given language. Therefore we can call them sequences of approximations for the given language. Both of the terms are referring to the same object, and we use them to highlight one desirable property from all properties respectively. The term monotone sequence highlights the inner structure of this object, sequence of approximations for a given language, on the other hand highlights the language being approximated.
We also defined operations such as union, disjoint union of two monotone sequence a finite deletion from a given monotone sequence and prefixes.

We introduced the distance measure between two monotone sequences in general based on the relative distance measure used between words, as well as the strong and weak distances, which are the main distances examined in this thesis. Non of these distances are satisfying the properties of the metric space. This is related to the following facts: the relative edit-distance is a number between 0 and 1 both included, and we are working with infinite structures.

Later we show how can we compute a strong or weak distance measure, when we know that the monotone sequence of finite languages is created by some of the operations introduced.

As we would like to compare the grammars as well using this approach, we introduced classes of sequences of approximations. These classes were defined similarly to the classes of languages: regular, linear and context-free where the name of the class

also refers to the complexity of the grammar with energy generating the particular sequence of approximations. We introduced the bounded versions of these classes, too. A modified version of Pumping Lemma was proven for the context-free sequences of approximations, and the Predecessor Lemma was proven for the same class.

Finally, we showed that the distance measure exists between all the context-free sequences of approximations, which are not growing too fast. We also showed that under some conditions the distance measure between a pair of context-free monotone sequences is related only to the grammars generating those sequences. These last two theorems, mainly the second one can be considered as a foundation for the definition of the distance measure between grammars with energy. For this purpose we should generalize the theorem about the existence of the distance measure in the future.

This is the list of properties which are not proved in this thesis but can be considered interesting in future branch researches:

- Transitivity: $D_S(\Phi, \Phi') = d$ and $D_S(\Phi'', \Phi') = d \Rightarrow D_S(\Phi, \Phi'') = d$.

- Specially for $d = 0$ or $d = 1$.

- Cardinality of sets generated by grammars with energy with fixed coefficients and given but changing energy threshold. (The cardinality of $L_e(G)$)

- To examine some of the feasible normal forms of the context-free grammars, whether they are applicable for the grammars with energy. The elimination of the chain rules for example.

- To add the following clause to the Pumping Lemma:
  Moreover, when L_j is the first set containing w in Φ and $n > 0$, then $i \geq j$.

There are some topics which were left open in this thesis and can be considered as follow up research topics:

- The equivalence of Linear Sequences of Approximations and approximations by the length of the words - similarly to Subsection 6.2.1.

- Distances between the monotone sequences generated by different grammars with energy with the same underlying grammar.

- To define the distance measure between two regular languages.

- To define the distance measure between two context-free languages.

Bibliography

[1] Alberto J. Evangelista and Bjørn Kjos Hanssen. Google Distance Between Words. 2006. `www.math.hawaii.edu/~bjoern/Publications/Evangelista_Kjos-Hanssen.pdf`.

[2] James R. Evans and Edward Minieka. *Optimization Algorithms for Networks and Graphs*. Marcel Dekker, Inc., 270 Madison Avenue, New York, New York 10016, 1992. Second Edition, ISBN 0-8247-8602-5.

[3] András Frank. A magyar módszer és általánosításai, (The Hungarian method and its extensions; in Hungarian). Technical Report TR-2002-06, Egerváry Research Group, Pázmány P. sétány 1/C, H-1117, Budapest, 2002. ISSN 1587–4451, `http://bolyai.cs.elte.hu/egres/www/tr-02-06.html`.

[4] Juraj Jánošík. *Konečné Aproximácie Jazykov*. FMFI UK, Bratislava, 1999. Master's Thesis.

[5] Nitish Korula. Maximum Weight Matching in Bipartite Graphs. 2010. Scribe: Abner Guzmán-Rivera, `www.cs.illinois.edu/class/sp10/cs598csc/Lectures/Lecture10.pdf`.

[6] Zbyněk Kubáček. Matematická Analýza II. Lectures for course Matematická Analýza II.

[7] Tomáš Kulich. *The Distances on Words*. FMFI UK, Bratislava, 2006. Master's Thesis.

[8] Ján Plesník. *Grafové Algoritmy*. Vydavateľstvo Slovenskej Akadémie Vied, Bratislava, 1983. 1197/I-1973.

[9] Pavol Ďuriš. Výpočtová zložitosť. 2003. Lectures for course Výpočtová Zložitosť a Vypočitateľnosť.